



Fikri EGE

Fabrika Yerleşim Probleminde CRAFT ve Karışık  
Tam Sayılı Programlama Yaklaşımları

ÇUKUROVA ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

FABRİKA YERLEŞİM PROBLEMİNDE  
CRAFT VE KARIŞIK TAM SAYILI  
PROGRAMLAMA YAKLAŞIMLARI

FİKRİ EGE

YÜKSEK LİSANS TEZİ

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

Bu tez 12.09.1997 Tarihinde Aşağıdaki Jüri Üyeleri Tarafından Oybirliği/Oyçokluğu ile Kabul Edilmiştir.

İmza.....

Yrd. Doç. Dr. Funda YILDIRIM  
DANIŞMAN

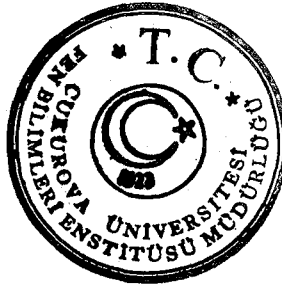
İmza.....

Prof. Dr. Aleattin SABANCI  
ÜYE

İmza.....

Yrd.Doç.Dr.Rizvan EROL  
ÜYE

Kod No: 1354



Prof. Dr. Ural DİNÇ  
Enstitü Müdürü  
İmza ve Mühür

|                                                              |     |
|--------------------------------------------------------------|-----|
| ÖZ.....                                                      | I   |
| ABSTRACT.....                                                | II  |
| TEŞEKKÜR.....                                                | III |
| ŞEKİLLER DİZİNİ.....                                         | IV  |
| ÇİZELGELER DİZİNİ.....                                       | V   |
| 1. GİRİŞ.....                                                | 1   |
| 1.1. Tesis Tasarımının Tanımı.....                           | 2   |
| 1.2. Fabrika Yerleşim Probleminin Sınıflandırılması.....     | 3   |
| 1.3. Tesis Planlamanın Amaçları.....                         | 5   |
| 1.3.1. İhtiyaçları Tanımlama.....                            | 5   |
| 1.3.1.1. Yerleşim Tasarım İşlemi.....                        | 5   |
| 1.3.1.2. Problemin Formülasyonu.....                         | 5   |
| 1.3.1.3. Problemin Analizi.....                              | 6   |
| 1.3.1.4. Alternatif Çözümlerin Araştırılması.....            | 12  |
| 1.3.1.5. Çözümün Seçilmesi.....                              | 14  |
| 1.3.2. Ürün, Proses ve Program Tasarımı.....                 | 14  |
| 1.3.3. Akış, Boşluk ve Faaliyet İlişkileri.....              | 15  |
| 1.3.4. Faaliyet İlişkileri.....                              | 15  |
| 1.3.5. Akış Prosesi.....                                     | 16  |
| 1.3.6. Akış Planlama.....                                    | 17  |
| 1.3.6.1. İş İstasyonu Belirleme.....                         | 19  |
| 1.3.6.2. Koridor Düzenleme.....                              | 21  |
| 1.4. Malzeme Taşıma.....                                     | 21  |
| 1.4.1. Malzeme Taşımanın Tanımı.....                         | 21  |
| 1.4.2. Malzeme Taşıma Sistemi Tasarımı.....                  | 22  |
| 1.5. Yerleşim.....                                           | 24  |
| 1.5.1. Fabrika Yerleşim Problemine Genel Bakış.....          | 25  |
| 1.5.2. Temel Yerleşim Türleri.....                           | 25  |
| 1.5.3. Yerleşim Prosedürü.....                               | 27  |
| 1.5.3.1. Apple'ın Tesis Yerleşim Prosedürü.....              | 28  |
| 1.5.3.2. Reed'in Tesis Yerleşim Prosedürü.....               | 28  |
| 1.5.3.3. Muher'ın Tesis Yerleşim Prosedürü.....              | 29  |
| 1.5.3.4. Algoritmik Araştırmalar.....                        | 29  |
| 1.5.4. Yeni Yerleşimler İçin İlişki Diyagramı Oluşturma..... | 30  |
| 1.6. Bilgisayar Destekli Yerleşim.....                       | 30  |
| 1.6.1. Algoritma Sınıflandırma.....                          | 30  |
| 1.6.2. BLOCPLAN.....                                         | 34  |
| 1.6.3. LOGIC.....                                            | 34  |
| 1.6.4. ALDEP.....                                            | 35  |
| 1.7. Niceliksel Tesis Planlama.....                          | 36  |
| 1.7.1. Tesis Yerleşim Modelleri.....                         | 36  |
| 1.7.2. Düzgün Doğrusal Tesis Yerleşim Problemi.....          | 36  |

|                                                    |     |
|----------------------------------------------------|-----|
| 1.7.3. Tek Tesis Minitoplam Yerleşim Problemi..... | 37  |
| 1.7.4. Tek Tesis Minimaks Yerleşim Problemi.....   | 37  |
| 1.7.5. N Tesis Yerleşim Problemi.....              | 37  |
| 2. ÖNCEKİ ÇALIŞMALAR.....                          | 39  |
| 3. MATERYAL VE METOD.....                          | 42  |
| 3.1. Materyal.....                                 | 42  |
| 3.2. Metod.....                                    | 42  |
| 3.2.1. Akış Ölçümü.....                            | 42  |
| 3.2.1.1. Niceliksel Akış Ölçümü.....               | 42  |
| 3.2.1.2. Niteliksel Akış Ölçümü.....               | 43  |
| 3.2.2. CRAFT.....                                  | 46  |
| 3.2.3. MIP.....                                    | 49  |
| 4. BULGULAR VE TARTIŞMA.....                       | 58  |
| 4.1. CRAFT Uygulaması.....                         | 58  |
| 4.2. Karışık Tam Sayılı Programlama.....           | 61  |
| 5. SONUÇLAR VE ÖNERİLER.....                       | 62  |
| 6. EKLER.....                                      | 68  |
| KAYNAKLAR.....                                     | 113 |
| ÖZGEÇMİŞ.....                                      | 116 |



**ÖZ**

**YÜKSEK LİSANS TEZİ**

**FABRİKA YERLEŞİM PROBLEMİNDE  
CRAFT VE  
KARIŞIK TAM SAYILI  
PROGRAMLAMA YAKLAŞIMLARI**

**FİKRİ EGE  
ÇUKUROVA ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**

**Danışman: Yrd. Doç. Dr. Funda YILDIRIM**

**Yıl: 1997, Sayfa: 116**

**Jüri: Yrd. Doç. Dr. Funda YILDIRIM**

**Prof. Dr. Aleattin SABANCI**

**Yrd. Doç. Dr. Rızvan EROL**

Bu çalışmada tesis tasarımı ele alınarak, bir işletmede fabrika yerleşim problemi incelenmiştir. İlk olarak konu hakkındaki literatür araştırması yapılmış, ve yerleşim probleminde öncelikle kullanılan metodlar tanıtılmış ve işletme için uygulama yapılmıştır. Bu uygulamada bilgisayar destekli algoritmalarından CRAFT algoritması ve Tam Sayılı Programlama (MIP) modelinden yararlanılmış ve sonuçlar tartışılmıştır.

**Anahtar Kelimeler: Tesis Tasarımı, CRAFT, MIP.**

## **ABSTRACT**

**MSc THESIS**

### **CRAFT AND MIP APPROACHES IN THE FACILITY LAYOUT PROBLEM**

**FIKRI EGE**

**DEPARTMENT OF INDUSTRIAL ENGINEERING  
INSTITUTE OF APPLIED SCIENCES  
UNIVERSITY OF ÇUKUROVA**

**Supervisor: Yrd. Doç. Dr. Funda YILDIRIM**

**Year: 1997, Pages: 116**

**Jury: Yrd. Doç. Dr. Funda YILDIRIM**

**Prof. Dr. Aleattin SABANCI**

**Yrd. Doç. Dr. Rızvan EROL**

In this study, the facility layout problem is investigated and applied to a real world problem. First, an extensive literature survey of methods and algorithms used to solve the layout problem is provided. Among these methods, a computer-aided layout algorithm (CRAFT) and Integer Programming approach (MIP) are utilized to solve the selected problem. Finally, comparison of the results produced by the two approaches is made to choose the best layout.

**Key Words: Facility Layout, CRAFT, MIP.**

## TEŞEKKÜR

Bu çalışmam sırasından yardımlarını ve desteklerini benden esirgemeyen danışmanım Sayın Yrd. Doç. Dr. Funda YILDIRIM' a , bölüm başkanımız Sayın Yrd. Doç. Dr. Rızvan EROL' a, Öğr. Gör. Ali KOKANGÜL ve Arş. Gör. İrfan MACİT' e teşekkürü borç bilirim.



## ŞEKİL LİSTESİ

## SAYFA

|                                                                                  |    |
|----------------------------------------------------------------------------------|----|
| 1.1. Tesis Planlama Hiyerarşisi.....                                             | 6  |
| 1.2. Yerleşim Problemlerinin Sınıflandırılması-yeni tesis durumu.....            | 7  |
| 1.3. Yerleşim Problemlerinin Sınıflandırılması-mevcut tesis durumu.....          | 8  |
| 1.4. Yerleşim Problemlerinin Sınıflandırılması-çözüm uzayı.....                  | 9  |
| 1.5. Yerleşim Problemlerinin Sınıflandırılması-yeni/mevcut tesis etkileşimi..... | 10 |
| 1.6. Yerleşim Problemlerinin Sınıflandırılması-mesafe ölçümü.....                | 11 |
| 1.7. Yerleşim Problemlerinin Sınıflandırılması-amaç.....                         | 11 |
| 1.8. Siyah Kutu Araştırması.....                                                 | 13 |
| 1.9. Hacim Çeşit Yerleşim Sınıflandırma.....                                     | 16 |
| 1.10. Akış Planlama Hiyerarşisi.....                                             | 17 |
| 1.11. Malzeme Yönetim Sistemi.....                                               | 18 |
| 1.12. İdeal Sistem Yaklaşımı.....                                                | 23 |
| 1.13. Ürün, Proses ve Program Tasarımı.....                                      | 26 |
| 4.1. İşletmenin Başlangıç Yerleşimi.....                                         | 59 |
| 5.1. İşleme İçin Alternatif Yerleşim.....                                        | 63 |
| 5.2. İşletme İçin Alternatif Yerleşim.....                                       | 64 |

**TABLO LİSTESİ****SAYFA**

|      |                                                                        |    |
|------|------------------------------------------------------------------------|----|
| 1.1  | Koridor Boşluk İhtiyaçları.....                                        | 20 |
| 1.2. | Çeşitli Akış Türleri İçin Tavsiye Edilen Koridor Boşlukları.....       | 22 |
| 3.1. | Yakınlık İlişki Değerleri.....                                         | 43 |
| 3.2  | İşletmenin Bölümler Arası Malzeme Akış Çizelgesi.....                  | 44 |
| 5.1. | Yerleşim Alternatifleri İçin Maliyetlerin Karşılaştırılması.....       | 65 |
| 5.2. | Birinci Bölümün Sabit Olduğu Durumda Maliyetler.....                   | 65 |
| 5.3. | Bölümlerin Sabit Olmaması Durumunda Maliyetlerin Karşılaştırılması.... | 66 |
| 5.4. | İki Bölümün Sabit Olduğu Durumda Maliyetlerin Karşılaştırılması.....   | 66 |
| 5.5. | CPU Zamanlarının Karşılaştırılması.....                                | 67 |



## 1. GİRİŞ

İstenilen ürünleri üretebilmek için bir fabrikanın bölümlerinin ya da araç ve gereçlerinin birbirlerine, sistemin girdi ve çıktılarına uygun olarak düzenlenmesine kısaca fabrika yerleşimi denir. Fabrika yerleşiminin temel amacı insan gücü malzeme ve donatımın verimli biçimde düzenlenmesidir. Fabrika yerleşimi binaların ve tesislerin, başlangıç tasarımlarından tezgahların yeri ve hareketine kadar olan tüm aşamaları kapsar.

Fabrika yerleşimi düzenlemelerinde dikkate alınması gereken önemli bir etken, gelecekte öngörülen genişlemelere karşı gerekli esnekliğe sahip olmasıdır. Gerek konumu ve gerekse bina yerleşimi, var olan işlemlere uyacak şekilde ileride işlemlerin genişletilmesine olanak sağlamalıdır.

Fabrika yerleşimi düzenlemelerinde dikkate alınması gereken kısıtlar vardır. Bu kısıtlara mevcut binaların konumu, ulaşım yollarına ya da kamu hizmetlerine bağlantıları ya da imar durumu vb. örnek olarak verilebilir.

Fabrika yerleşimi sanayi devrimi ile önemli bir sorun olarak gündeme gelmiştir. Sanayi devrimi öncesi küçük atelyelerde ve küçük ölçeklerde yapılan üretim, sanayi devrimi ile birlikte giderek daha büyük ölçeklerde yapılmaya başlanmıştır. Bunun doğal sonucu olarak, üretim faaliyetleri giderek artan oranda kompleks hale gelmiştir. Bu da fabrika yerleşimi problemini ortaya çıkarmıştır.

Fabrika yerleşimi, sanayi devriminin ilk yıllarından, 1950' li yıllara kadar üzerinde fazla durulmayan bir problem olmuştur. Bunun temel iki nedeni o yıllara kadar işgücünün ve arazi maliyetlerinin toplam üretim maliyeti içinde çok önemli bir yer tutmamasıdır. Bir başka deyişle, 1950' li yıllara kadar fabrika yerleşim problemi istenildiği kadar işgücü ve yer kullanılarak çözümlenmiştir. Ancak, gerek işgücü maliyetlerinin giderek artması ve toplam ürün maliyetlerinin içinde önemli bir yer tutması, gerekse de giderek kalabalıklaşan şehir merkezlerinde arazi fiyatlarının artarak ilk yatırım maliyetlerinde önemli bir yer tutması, fabrika yerleşimini önemli bir problem haline getirmiştir. Fabrika yerleşim probleminin artan önemine paralel olarak,

bu alanda ilk çalışma 1955 yılında Muther tarafından yapılmıştır. 1955 yılından günümüze kadar bu alanda bir çok çalışma yapılmıştır.

Bu çalışma da fabrika yerleşiminin orta ölçekli bir işletmeye uygulanmasını kapsamaktadır. Uygulama yapılan işletmede kompresör ve hidrofor üretilmektedir. Birinci bölümde konuyla ilgili teorik bilgiler, ikinci bölümde önceki çalışmalar, üçüncü bölümde materyal ve uygulanan metodlar ve son bölümde de sonuçlar verilerek bu sonuçlar tartışılacaktır. Uygulama yapılan işletmenin yerleşiminin yapılmasında daha sonraki bölümlerde ayrıntıları ile açıklanan CRAFT algoritması ve karışık tam sayılı programlama modelinden yararlanılmıştır. Sonuçlar ekler kısmında sunulmuştur. Bu çalışmanın asıl amacı işletmenin yeni taşınacağı yer için fabrika yerleşimini yapmak, mevcut sorunları gidererek taşıma maliyetlerini minimize etmektir.

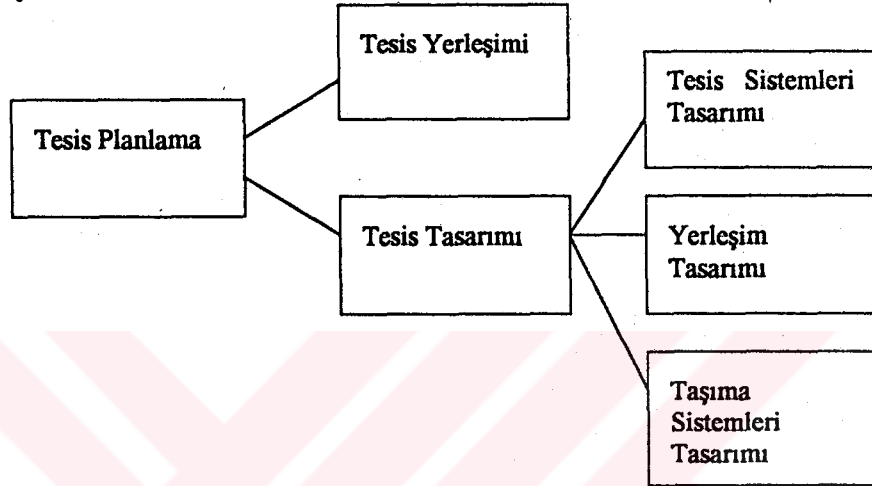
### 1.1 Tesis Tasarımının Tanımı

Tesis tasarımı birkaç disiplini kapsayan kompleks bir konudur. Mühendislik dalında inşaat, elektrik, endüstri ve makine mühendisleri tesis tasarımıyla ilgilidir. Yer tesisin müşteri, yan sanayi ve ilişkide olduğu diğer tesislerle olan yerleştirmeyi kapsar. Aynı zamanda yer özel bir alanda yerleştirmeyi ve yönlendirmeyi de içerir.

Tesisin tasarım bileşenleri tesis sistemleri, yerleşim ve taşıma sistemini içerir. Tesis sistemleri yapısal sistemler, atmosferik sistemler, ışıklandırma /elektrik /haberleşme sistemleri, güvenlik sistemleri ve sağlık sistemlerini kapsar; yerleşim bina içindeki bütün ekipman, makine ve eşyaları içerir; ve taşıma sistemi ihtiyaç duyulan tesis ilişkilerini sağlamak için gerekli mekanizmaları kapsar. Üretim tesisi için tesis sistemleri çevre (yapı ve çevredeki elemanlar), güç, ışık, gaz, ısı, havalandırma, soğutma, su ve atık ihtiyaçlarını kapsar. Yerleşim bina içindeki üretim alanlarını, üretimle ilgili veya destek alanlarını ve personel alanlarını içerir. Taşıma sistemi üretimi desteklemek için gerekli malzeme, personel, bilgi ve ekipman taşıma sistemlerini içerir.

Tesisin tasarım bileşenlerinin, tesisin amacını nasıl karşılayacağına karar vermeye tesis tasarımı denir. Bu nedenle tesis planlama, tesis yeri ve tesis tasarımı olarak iki alt gruba ayrılabilir (Şekil 1.1.). Tesis yeri makro konuları ele alırken, tesis tasarımı mikro elemanları belirler.

Yerleşim problemlerinin sınıflandırılmasında kullanılan en son kategori alternatif çözümlerin değerlendirilmesinde yararlanılan amaç fonksiyonudur. Tesis düzenlemesi durumunda bir dizi niteliksel amaç genellikle kullanılır; oysa yerleşim problemlerinin çözümünde niceliksel amaçlar genellikle bazı toplam maliyet fonksiyonlarının minimizasyonu veya tesis çiftleri arasındaki maksimum maliyetin minimizasyonudur.



Şekil 1.1. Tesis Planlama Hiyerarşisi

### 1.3 Tesis Planlamanın Amaçları

Tesis Planlamanın genel kural ve teknikleri vardır ve bunlar izlenirse etkili tesis planına ulaşılır. Tesis planlama için anlamlı bir amacı kısaca belirtmek çok güçtür. Bu nedenle, tesis planlamayı tesis yeri ve tesis tasarımı alt gruplarına bölerek bir dizi çalışılabilir amaç elde edilebilir.

Bazı tesis tasarımı amaçları;

- 1- Malzeme taşıma, malzeme kontrol ve güvenliği geliştirmede organizasyonu desteklemek,
- 2- İnsan, ekipman, boşluk, ve enerjiden etkili bir biçimde yararlanmak,
- 3- Sermaye yatırımını minimize etmek
- 4- Uygulanabilirlik ve bakımı kolaylaştırmak
- 5- Çalışanların güvenliğini ve tatminini sağlamaktır.

### 1.3.1 İhtiyaçları Tanımlama

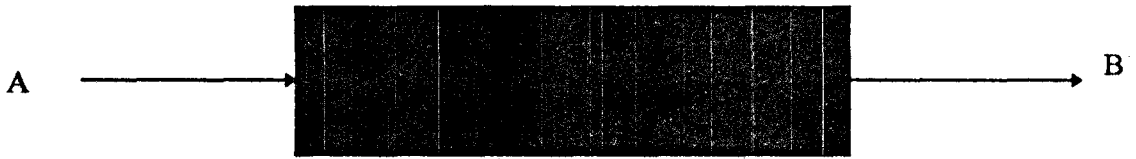
#### 1.3.1.1 Yerleşim Tasarım İşlemi

Fabrika yerleşim problemi durumunda en iyi yerleşim tasarımı araştırılır. Bir çok tasarım problemiyle karşılaşılabilceğinden, herhangi bir tasarım problemin çözümü için genel bir prosedür kullanmak önemlidir. Aşağıdaki adımlar Krick (1965) tarafından tavsiye edilmektedir:

- Problemin formülasyonu
- Problemin analiz edilmesi
- Alternatif çözümlerin aranması
- Çözümün seçilmesi
- Çözümün tayini

#### 1.3.1.2 Problemin Formülasyonu

Problemin formülasyonu siyah kutu yöntemiyle yapılabilir. Krick' in siyah kutu açıklamasına göre, meydana gelen iş durumu (A) ve istenilen iş durumu (B) vardır. Aynı zamanda A' dan B' ye giderken bir dönüşüm olması gerekir. A durumundan B durumuna dönüşümü gerçekleştirmek için birden fazla metod vardır. Problemin çözümü giriş için A, çıkış için B, bilinmeyenlere sahip siyah kutu, belirlenmemiş içerikler olarak görüntülenir.



Şekil 1.8. Siyah Kutu Araştırması (Krick, 1965)

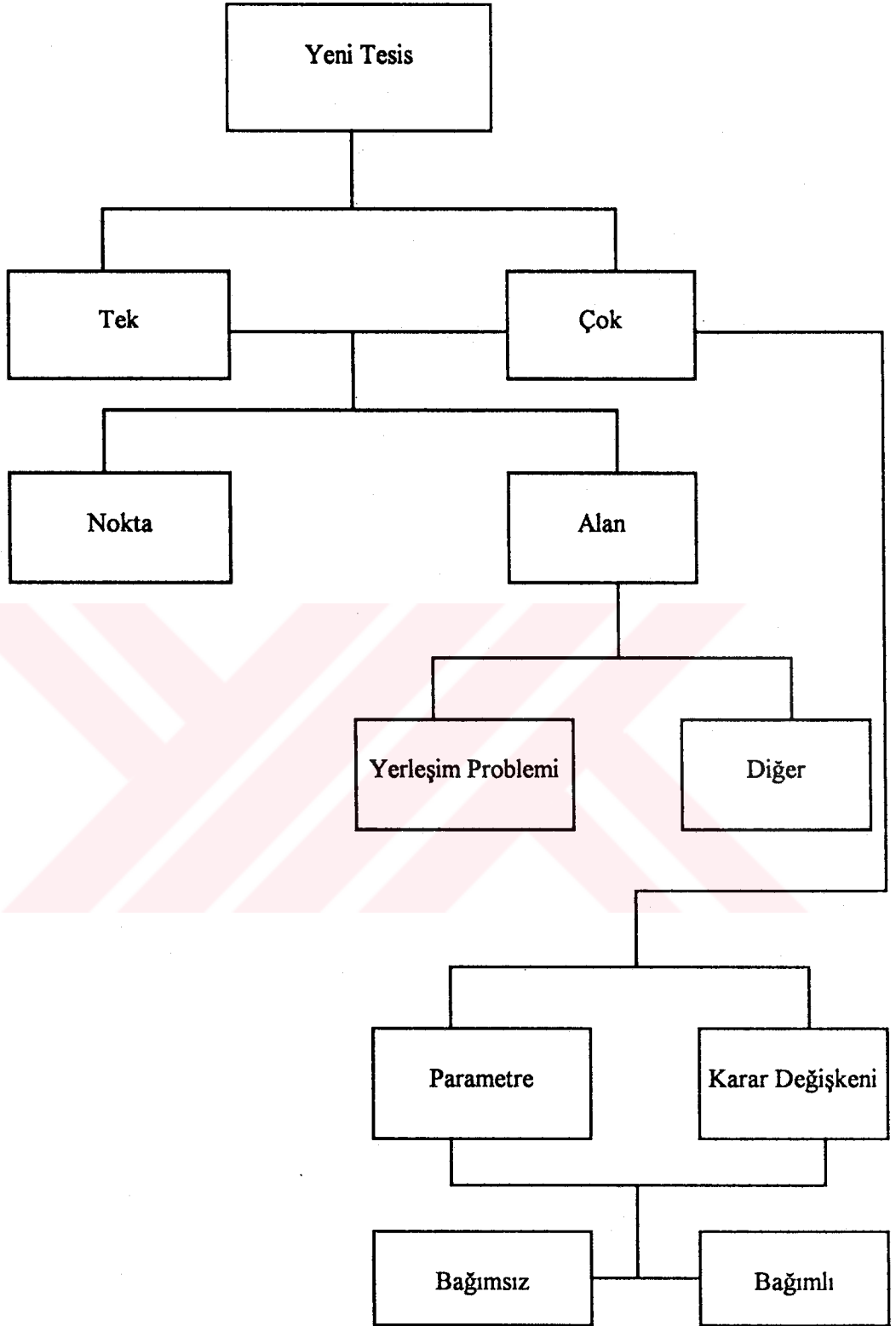
Siyah kutu araştırması (Şekil 1.8.) problemin formülasyonunda A ve B durumlarının uygun kimliğini belirlemeyi kolaylaştırır. Genellikle problemin formülasyonunu ekonomik durumun, zaman kısıtlarının ve organizasyonel sınırların izin verildiği kadar geniş tutulmasıdır. Dar formülasyon yapılarak probleme daha az sonuç bulunacak şekilde yapılan formülasyon toplam sistemde maliyetten tasarruf sağlayabilecek çözümün kaçırılması ile sonuçlanabilir.

### 1.3.1.3 Problemin Analizi

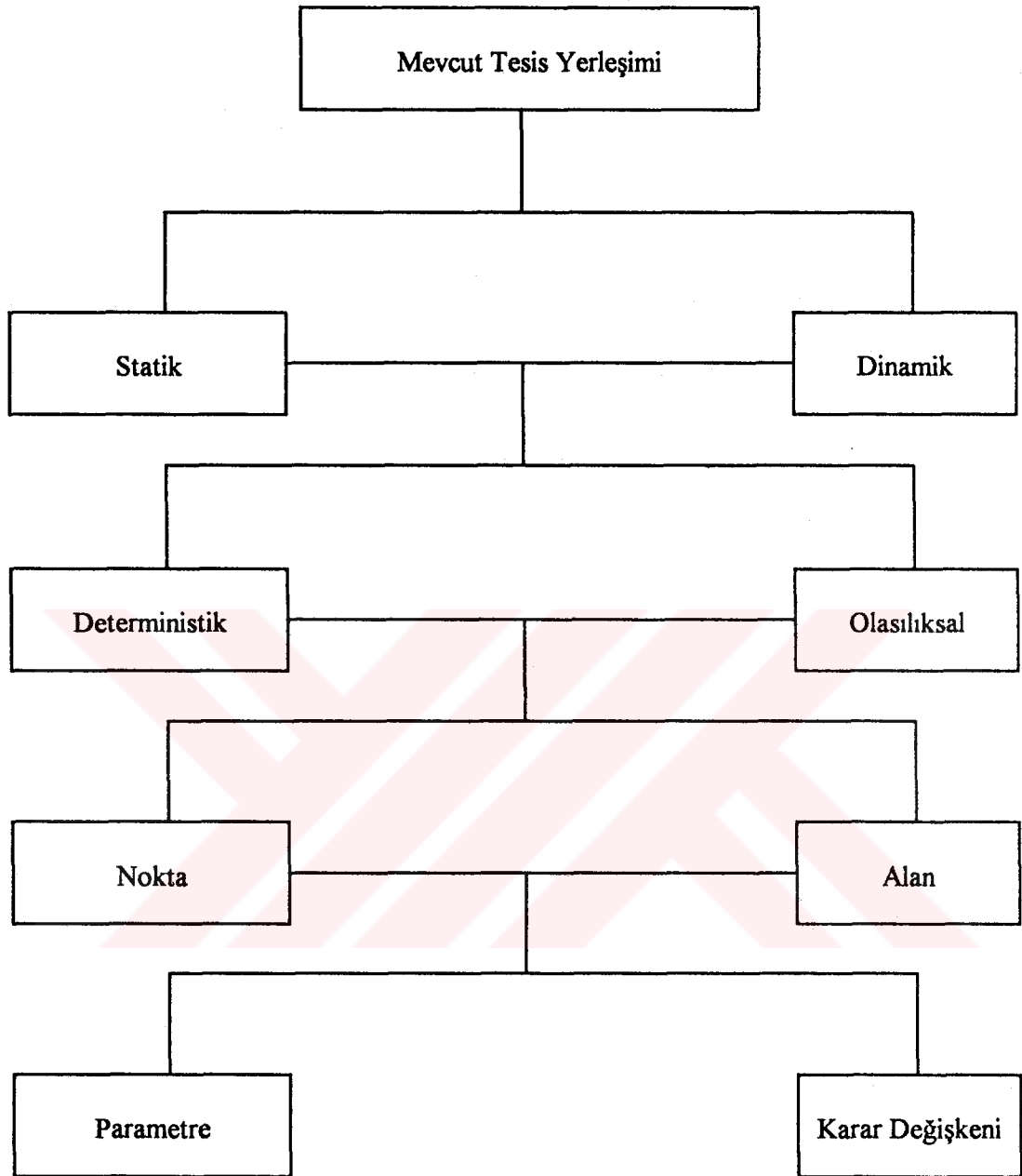
Daha önce formüle edilen problemin analizi, kısıtları kapsayan problem karakteristiklerinin nispeten daha detaylı ifade edilmesini içerir. Gerçek kısıtları fiktif kısıtlardan ayırmaya dikkat edilmelidir. Problemin analizi aynı zamanda problemlerin alternatif çözümlerinde yer alan kriterin belirlenmesini gerektirir.

Kriter seçimi aşamasındaki fabrika yerleşim modelleri normatif modellerdir. Bu nedenle ayrı bir kriter seçme sorunu göz önüne alınmalıdır. Fabrika yerleşimi konusunda en çok yapılan araştırmalardan biri alınan mesafenin bazı fonksiyonlarının minimize edilmesi üzerinedir. Bu araştırmanın seçilmesi, alınan mesafenin minimize edilmesi yolu ile malzeme taşıma maliyetlerinin minimize edilmesi temeline dayanır. Vollman ve Buffa (1966) geçmişte geliştirilen bir çok fabrika yerleşim modelinin,

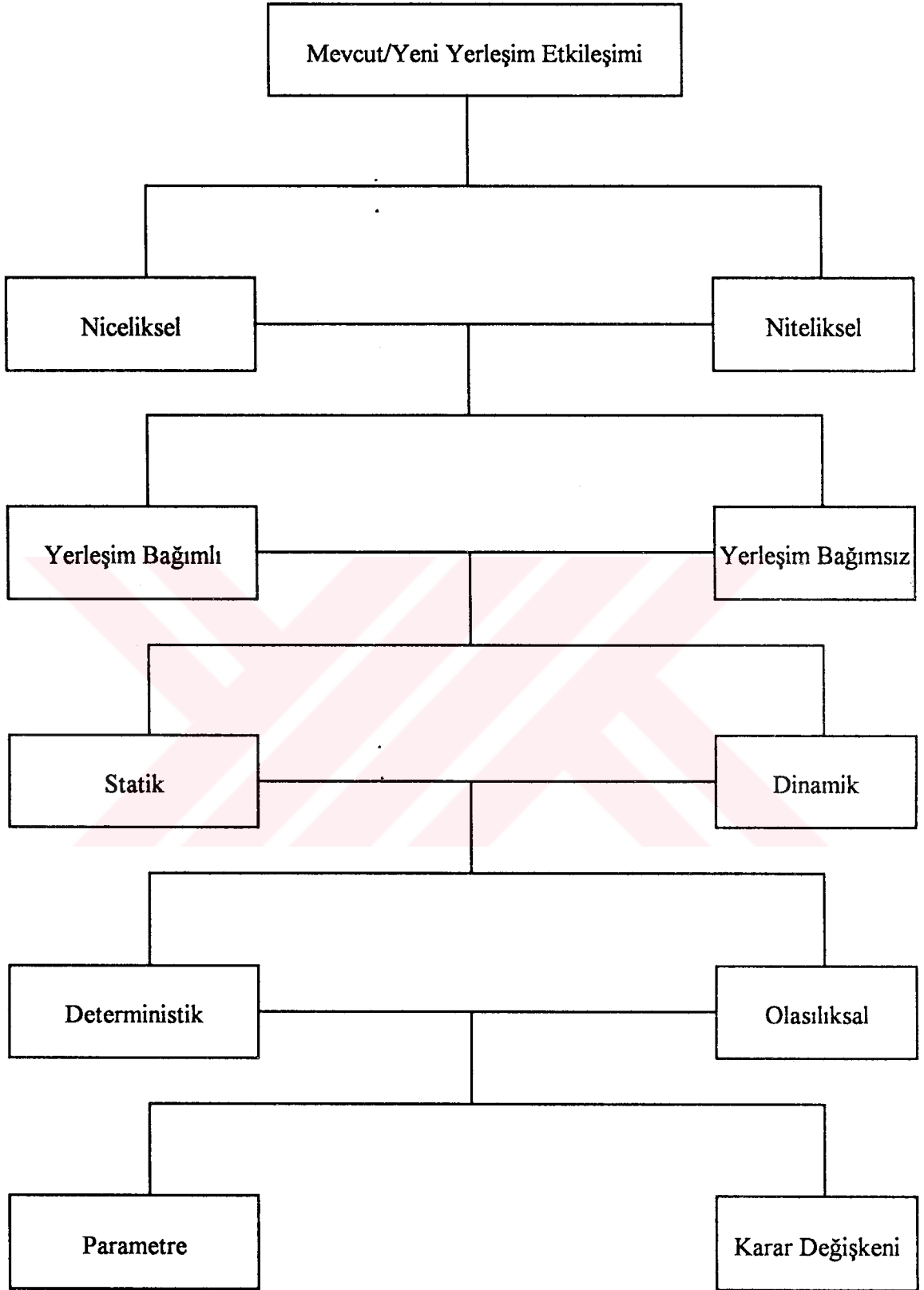
- a) Maliyet ve akış bilgilerinin tanımsal olarak bilinmeyen durumlar için mevcut olduğunu
- b) Malzeme taşıma maliyetlerinin spesifik olaylarda doğrusal, artan ve tayini mümkün olması
- c) Malzeme taşıma maliyetinin tek önemli faktör olduğu
- d) Akış bilgilerinin deterministik olduğu
- e) Fabrika yerleşimi problemiyle diğer sistem problemleri arasında etkileşim olmaması gibi tahminlere dayandığını belirtirler.



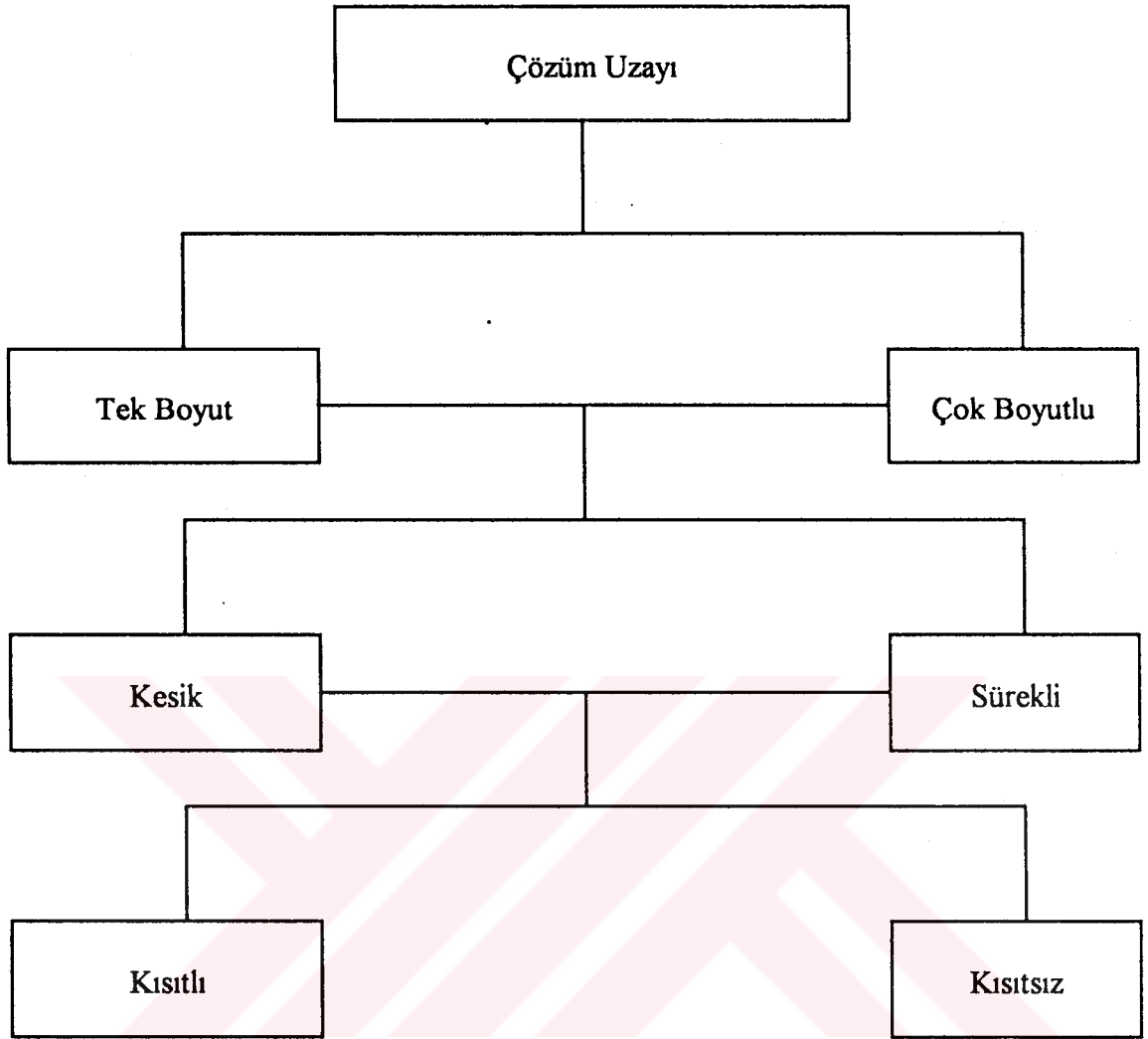
Şekil 1.2. Yerleşim Problemlerinin Sınıflandırılması - yeni tesis durumunda (Francis, R. L., White, J.A., 1974)



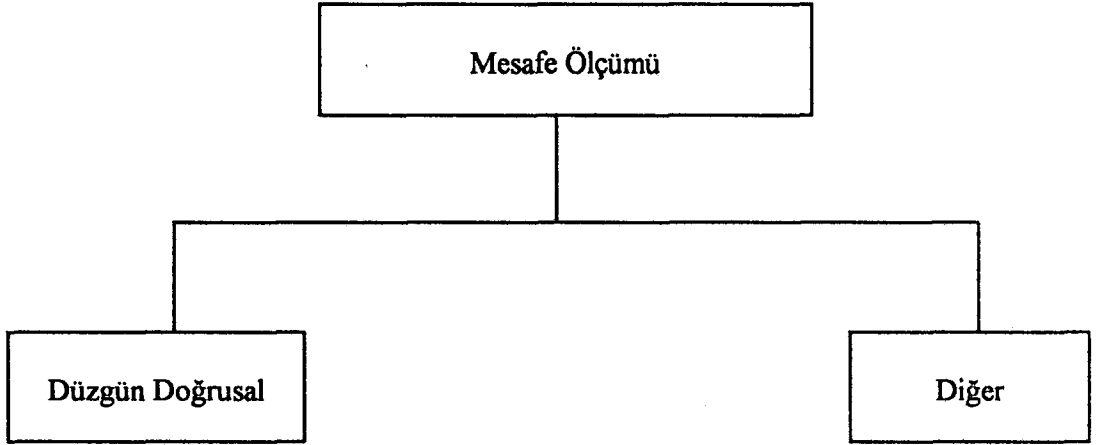
Şekil 1.3. Yerleşim Problemlerinin Sınıflandırılması - mevcut tesis durumunda (Francis, R. L., White, J.A., 1974)



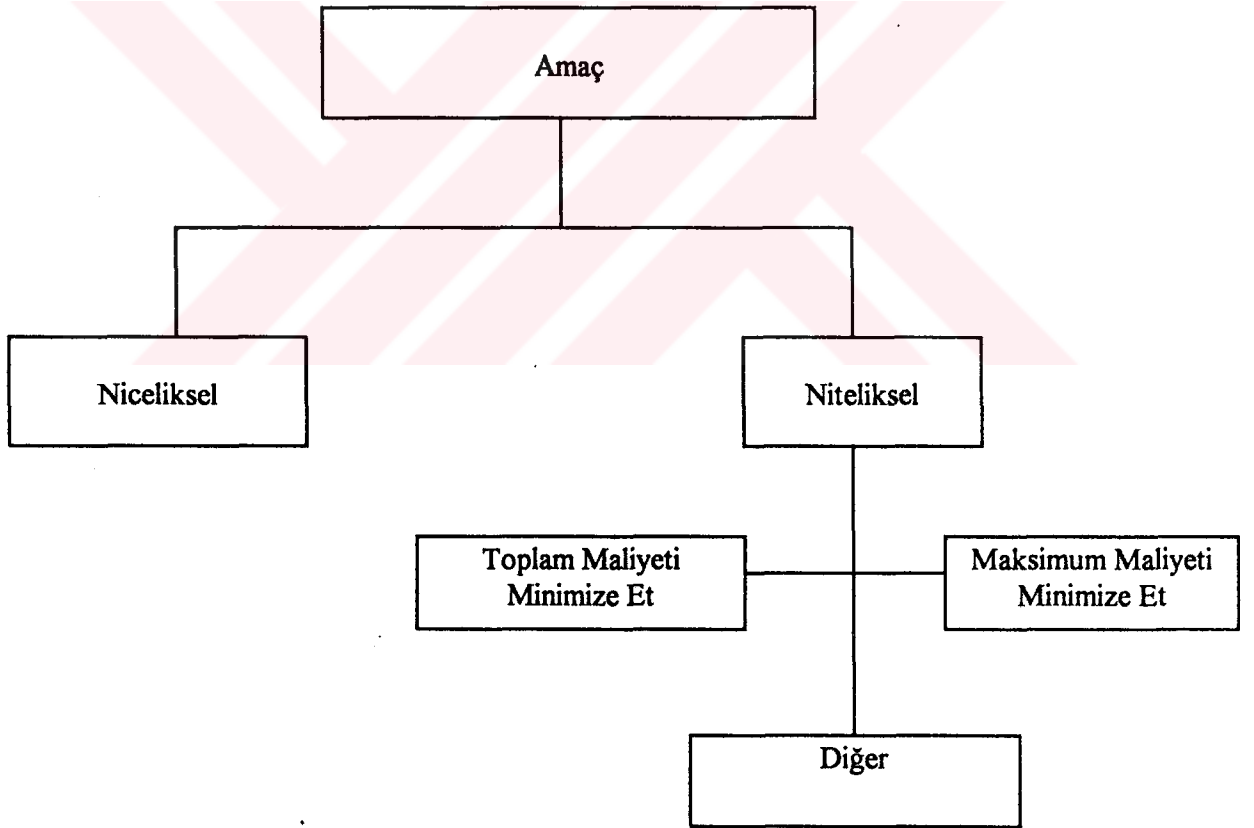
Şekil 1.4. Yerleşim Problemlerinin Sınıflandırılması - Yeni/mevcut tesis etkileşimi  
(Francis, R. L., White, J.A., 1974)



Şekil 1.5. Yerleşim Problemlerinin Sınıflandırılması - Çözüm Uzayı Karakteristikleri  
(Francis, R. L., White, J.A., 1974)



Şekil 1.6. Yerleşim Problemlerinin Sınıflandırılması - Mesafe Ölçümü (Francis, R. L., White, J.A., 1974)



Şekil 1.7. Yerleşim Problemlerinin Sınıflandırılması - Amaç (Francis, R. L., White, J.A., 1974)

Toplam alınan mesafenin minimize edilmesi yerine, maksimum alınan mesafenin minimize edilmesi istenebilir. Böyle bir çözüme minimaks yerleşim denir.

Malzeme taşıma maliyetinin tanımı, üzerinde çalışılan sistemin bileşenleri arasındaki mesafenin çeşitli maliyet fonksiyonlarını içerir. Genellikle kullanılan iki mesafe öklidyen ve düzgün doğrusal mesafedir. Koordinatları  $(x_A, y_A)$  ve  $(x_B, y_B)$  olan yerleşimin düzgün doğrusal mesafesi  $|x_A - x_B| + |y_A - y_B|$  'ye eşittir. Öklidyen mesafe ise  $[(x_A - x_B)^2 + (y_A - y_B)^2]^{1/2}$  'ye eşittir.

#### 1.3.1.4 Alternatif Çözümlerin Araştırılması

Yerleşim problemlerinde alternatif çözümlerin araştırılması, problemin formülasyonunda kullanılan siyah kutunun alternatif içeriklerinin özelliklerini kapsar. Problemin en son çözümünün genelleştirilmiş alternatiflerden gelmesinden dolayı, alternatif çözümlerin sayı, kalite ve çeşidini maksimize etmeye çalışması önerilir.

Alternatif çözümlerin araştırılmasında, uğraşı yaratıcı olmalı ve düşünce bazındaki bütün çabalar mevcut çözümden ayrılmaya yönelmelidir.

Alternatif çözümlerin değerlendirilmesinde bir dizi teknik Muther (1955) tarafından verilmiştir:

- a. Avantaj ve dezavantajların listelenmesi: Alternatif yerleşimlerin değerlendirilmesinde en kolay yol avantaj ve dezavantajların listelenmesidir.
- b. Sınıflandırma: Yerleşimde önemli olan faktörleri veya düşünceleri seç, listele ve her faktör için sayısal olarak alternatifleri sırala.
- c. Faktör analizi: Her faktöre bir sayısal ağırlık verilir ve daha sonra her alternatif faktörlere karşılık gelecek şekilde sınıflandırılır. Ağırlıklandırılmış sınıflandırmalar her alternatif için toplanır ve en iyi skoru veren alternatif seçilir.
- d. Maliyet karşılaştırma: Her alternatifte birleştirilmiş bütün maliyetler, sağladıkları tasarruf kadar belirlenir ve en son ekonomik alternatif seçilir.

Avantaj ve dezavantajları listeleme en kolay yoldur. Aynı zamanda bu belkide en az doğru olanıdır.

Sınıflandırma prosedüründe bütün alternatiflerin aynı faktörlere göre kıyaslanması özelliği vardır. Bu metodun ihmal ettikleri bazı önemli faktörlerin görülemiyebilmesi ve tercih edilen tasarımın seçiminin kolayca yapılamamasıdır. Sınıflandırma prosedürü kullanıldıktan sonra, bir tercih yapılabilecek şekilde sınıflandırmaların birleştirilmesi gerekir. Faktör analizi tekniği bu prosesi her faktöre ağırlıklar atanmasıyla kesinleştirir. Niceliksel ve niteliksel faktörler dahil edilebildiğinden ve tercih edilen çözüm kesinleştiğinden, faktör analizi alternatif yerleşimlerin belirlenmesinde sık kullanılan bir metottur.

Sınıflandırma prosedürü ve faktör analizi prosedürü her faktör için yerleşim alternatiflerinin karşılaştırılmasını içerir. Bu karşılaştırmanın mümkün olduğunca objektif yapılması önemlidir. Kullanılan bir araştırma her faktör için iki yerleşim alternatifinin tüm kombinasyonlarının sınıflandırıldığı ikili karşılaştırma tekniğidir. Örnek olarak beş yerleşim alternatifinin (A, B, C, D, E) bulunduğu ve verilen faktör için ikili karşılaştırmanın sonuçları aşağıda verilmiştir:

$$A > B, B < D$$

$$A > C, B < E$$

$$A < D, C < D$$

$$A > E, C < E$$

$$B < C, D > E$$

Burada  $X > Y'$  nin anlamı  $X'$  in sınıflandırması  $Y'$  den büyüktür. Bu karşılaştırmaları ele alınan faktör için birleştirirsek

$$D > A > E > C > B$$

olarak elde edilir.

Eğer n alternatif ve m faktör varsa

$$m \cdot n \cdot (n-1)/2$$

kadar karşılaştırma gerekir.

İkili karşılaştırma tekniği tutarsızlıkları test etmede yararlıdır.

En çok kullanılan faktörler:

a. Gelecekte genişlemenin kolay olması

b. Yerleşimin esnekliği

- c. Malzeme taşıma tesiri
- d. Boşluklardan yararlanma
- e. Emniyet ve temizlik
- f. Çalışma koşulları
- g. Deneme ve kontrolün kolaylığı
- h. Görünüş, kuruluş değeri, halkla ve toplumla ilişkiler
- i. Ekipmandan yararlanma
- j. İşletmenin organizasyon yapısıyla uyum
- k. Kapasite veya ihtiyaçları karşılama kabiliyetleri
- l. Yatırım ya da sermaye ihtiyacı
- m. Tasarruflar, ödemeler, kârlılık

#### 1.3.1.5 Çözümün Seçilmesi

Tasarım prosesinde son aşama çözümün seçimidir. Tamamlama için yerleşim planının seçiminden sonra tasarımın daha detaylı tayini gerekir. Normal olarak çözümü tayin etmek için yerleşimin bir ikonik temsili ve yazılı rapor kullanılır. Yerleşim problemini çalışma alanına bağlı olmak üzere daha detaylı çizimler (elektrik, su yolları vs.) yerleşimi tayin etmek için kullanılır.

#### 1.3.2 Ürün, Proses ve Program Tasarımı

Üretim tesisleri için tesis planlama prosesi aşağıda listelenmiştir:

1. Üretilcek ürünleri tanımla
2. Üretilcek ürünler için üretim yöntemlerini ve gereken ilgili faaliyetleri belirle
3. Bütün faaliyetler arasındaki ilişkilere karar ver
4. Bütün faaliyetler için boşluk ihtiyaçlarına karar ver
5. Alternatif tesis planlarını genelle
6. Alternatif tesis planlarını değerlendir
7. Tercih edilen tesis planını seç
8. Tesis planını tamamla
9. Tesis planını sürdür ve kur
10. Üretilcek ürünleri güncelle ve tesisin amacını yeniden tanımla

### 1.3.3 Akış Boşluk ve Faaliyet İlişkileri

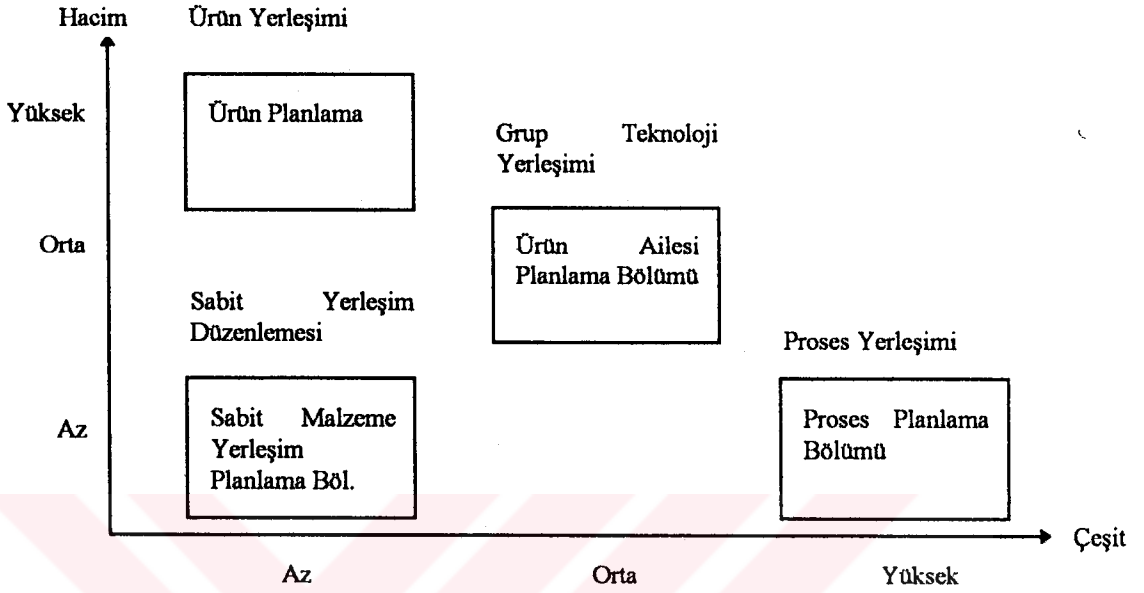
Tesisin ihtiyaçlarına karar vermede üç önemli yaklaşım akış, boşluk ve faaliyet ilişkileridir. Akış yığın boyutuna, birim yükleme boyutuna, malzeme taşıma ekipmanı ve stratejilerine, yerleşim düzenlemelerine ve bina konfigürasyonuna bağlıdır. Boşluk yığın boyutunun, depolama sisteminin, üretim ekipman türü ve boyutunun, yerleşim düzenlemelerinin, bina konfigürasyonunun, malzeme taşıma ekipmanının ve ofis, kafeterya ve dinlenme yerlerinin tasarımının fonksiyonudur. Faaliyet ilişkileri malzeme ve personel akışı, çevresel değerlendirmeler, organizasyon yapısı, sürekli gelişim metodolojisi, kontrol yayınları ve proses ihtiyaçları ile tanımlanır (Şekil 1.9).

### 1.3.4 Faaliyet İlişkileri

Faaliyet ilişkileri tesis planlama prosesinde bir çok karara temel oluşturur. Göz önüne alınan ilk ilişkiler:

1. Anlık (kısa süreli) kontrol ve ilişkilerin raporlanmasıyla bilgilendirilen organizasyon ilişkisi
2. Malzeme, insan, ekipman, bilgi ve para akışını içeren akış ilişkileri
3. Merkeziye karşı merkezi olmayan malzeme kontrolü, gerçek zamana karşı yığın envanter kontrol, ve otomasyon ve entegrasyon seviyelerini içeren kontrol ilişkileri
4. Emniyet önlemleri ve derece, gürültü, toz, nem ve dumanı içeren çevresel ilişkiler
5. Zemin yükleme, kimyasal prosesler ve özel servisler gibi yukarıdakilerin dışındaki proses ilişkileri.

Bir kaç ilişki niceliksel olarak ifade edilebilir; diğerleri niteliksel olarak ifade edilmelidir. Örneğin akış ilişkileri saatte yapılan hareket sayısı, vardiya başına hareket eden ürün sayısı, envanter için sermaye oranı, ayda işlenen doküman sayısı ve işçi ve malzeme için aylık harcama terimleriyle ifade edilir.



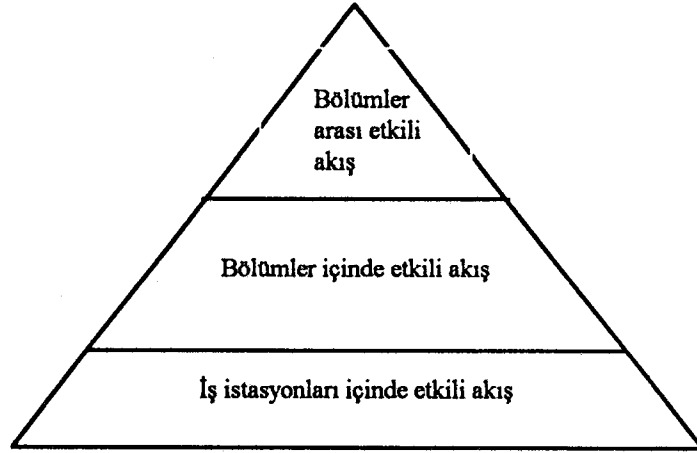
Şekil 1.9. Hacim-Çeşit Yerleşim Sınıflama

Organizasyon ilişkileri, organizasyon çizelgesiyle gösterilir. Organizasyon hakkında bilgi her zaman mevcuttur ve organizasyonun faaliyet ilişkilerinde göz önüne alınır.

#### 1.3.5 Akış prosesi

Akışın hedefi, akışın kaynakları, ve kaynakları koordine eden iletişimle tanımlanır. Akış için kaynaklar gerekli akışı sağlamak için işleme ve nakliye içerir. Eğer akış prosesi üretim tesisine malzemelerin akışını içeriyorsa, akış prosesi malzeme yönetim sistemi olarak adlandırılır. Malzeme yönetim sistemleri şirket tarafından ürünü üretebilmek için satın alınan malzeme, parça ve tedarik etmeyi içerir.

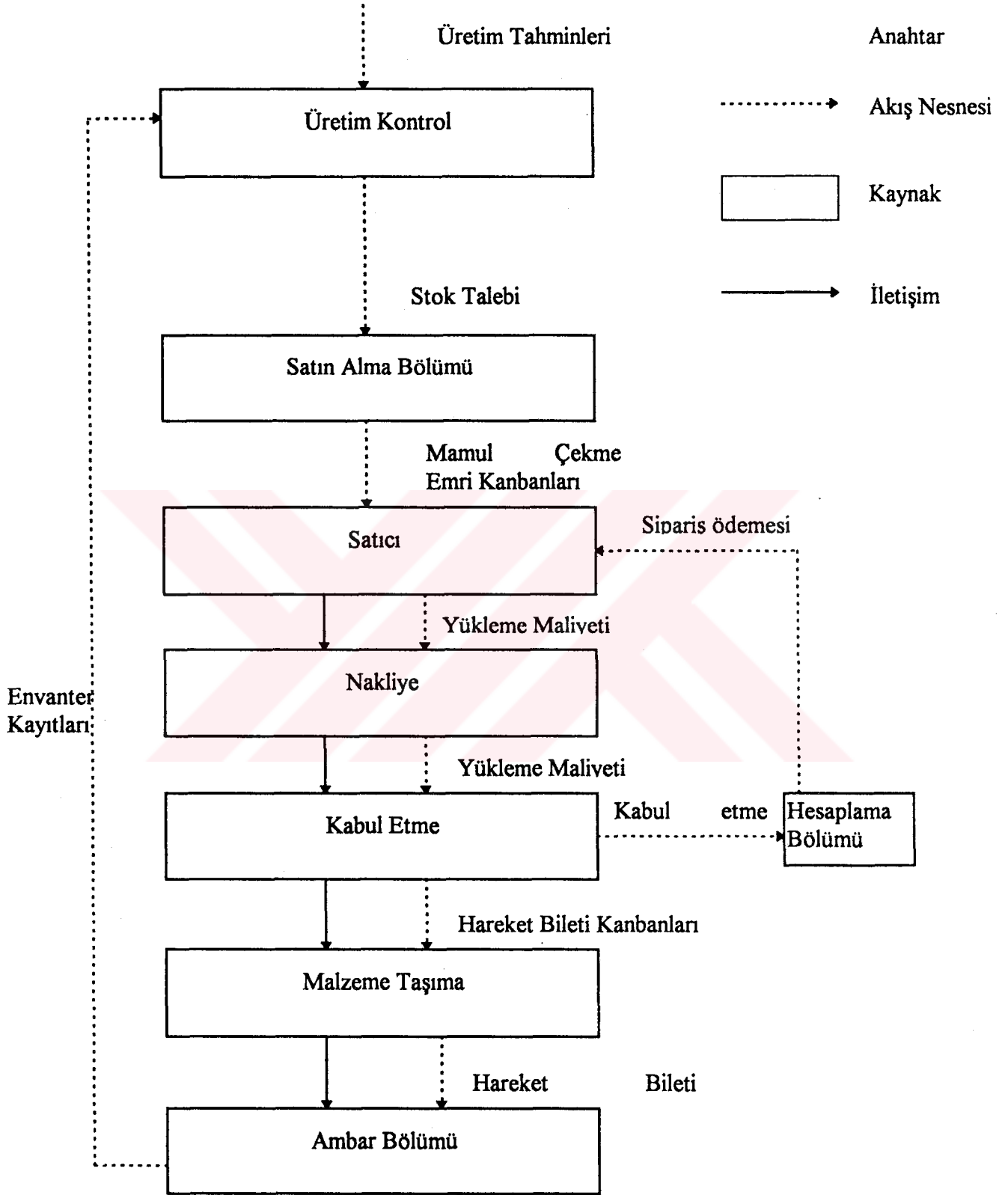
### 1.3.6 Akış Planlama



Şekil 1.10. Akış Planlama Hiyerarşisi

Malzeme akışındaki iş basitleştirme yaklaşımları:

- 1- Malzemenin, bilginin veya insanların son kullanılacağı noktaya doğrudan gitmesinin planlayarak ve aradaki aşamaları elimine ederek akışı elimine etmek
- 2- Birbirini takip eden noktalarda çoklu akışları mümkün olduğunca aza indirecek şekilde planlama
- 3- Akışları ve işlemleri malzeme, bilgi ve işlem hareketlerini proses adımıyla birleştirecek şekilde birleştirme.



Şekil 1.11 Malzeme Yönetim Sistem

## 1. Boşluk İhtiyaçları

### 1.3.6.1 İş İstasyonu Belirleme

Bir iş istasyonunda ekipman, malzeme ve personel için boşluğa ihtiyaç vardır.

Ekipman boşluğu aşağıdakileri içerir:

1. Ekipman
2. Makine hareketi
3. Makine bakımı
4. Fabrika servisleri

Ekipman boşluk ihtiyaçları makine bilgi levhasından elde edilebilir. Eğer bu yoksa, fiziksel envanter aşağıdakileri elde etmede kullanılır:

1. Makine üreticisi ve tipi
2. Makine modeli ve seri numarası
3. Zemin yükleme ihtiyacı
4. Maksimum noktada statik yükseklik
5. Maksimum dikey hareket
6. Maksimum noktada statik genişlik
7. Maksimum hareket (sola)
8. Maksimum hareket (sağa)
9. Maksimum noktada statik derinlik
10. Operatöre doğru maksimum hareket
11. Operatörden uzağa maksimum hareket
12. Bakım ihtiyaçları ve alanları
13. Fabrika servisi ihtiyaçları ve alanları

Makine hareketini de içeren zemin alanı ihtiyacı toplam genişliğin (statik genişlik + sağa ve sola maksimum hareket) \* toplam derinlik (statik derinlik + operatöre doğru veya operatörden uzağa maksimum hareket) ile elde edilebilir. Zemin alanı ihtiyacına bakım ihtiyacı ve fabrika servisi alanları eklenmelidir.

İş istasyonları için malzeme alanı ihtiyacı aşağıdakileri içerir:

1. Malzemeleri kabul etme ve depolama
2. Proses içindeki malzemeler
3. Malzemeleri yükleme ve depolama
4. Depolama ve yükleme atık ve kırıntıları
5. Aletler, teçhizat, jigler ve bakım malzemeleri

Malzemeleri kabul etme ve depolama için, prosesteki malzemeler ve depolanan ve taşınan malzemeler, taşınacak birim yükün boyutları, malzemelerin makinelerden akışı bilinmelidir. Malzeme taşıma sistemine bağlı olarak, boşluk için minimum ihtiyaç bir birim yük işlenecek parça, makinede işlenmiş bir birim yük, makinede işlenecek bir birim yük ve tamamlanmış bir birim yük için gereklidir. Ek boşluk prosesteki malzemelerin makinede yerleştirilmesi için gerekli olabilir. Atıkları ve hatalı parçaları makinadan almak için ve iş istasyonundan depoya kaldırmak için boşluk sağlanmalıdır.

İş istasyonunun boşluğu için personel alanı

1. Operatör
2. Malzeme taşıma
3. Operatör giriş ve çıkışı

Tablo 1.1. Koridor boşluk ihtiyaçlarını göstermektedir.

Tablo 1.1. Koridor-Boşluk İhtiyaçları

| EĞER EN BÜYÜK YÜK                 | KORIDOR İZİN ORANI* |
|-----------------------------------|---------------------|
| 6 ft <sup>2</sup> den büyükse     | 5-10                |
| 6-12 ft <sup>2</sup> arasındaysa  | 10-20               |
| 12-18 ft <sup>2</sup> arasındaysa | 20-30               |
| 18 ft <sup>2</sup> den büyükse    | 30-40               |

\* Ekipman, malzeme ve personel için gerekli net alanların yüzdesi olarak ifade edilir.

### *1.3.6.2 Koridor Düzenleme*

Koridorlar tesise etkili akış sağlamak amacıyla yerleştirilir. Koridorlar bölüm koridorları ve ana koridorlar olarak sınıflandırılabilir.

Koridor genişlikleri, koridorda taşınacak akışın türü ve hacmine bağlı olarak belirlenir. Akış çeşidi, koridoru kullanan personel ve ekipman göz önüne alınarak tayin edilebilir. Tablo 1.2. çeşitli akış türleri için koridor genişliklerini belirlemektedir. Ana koridorlar için koridor genişlikleri Tablo 1.2.' den elde edilebilir. Eğer belirli bir koridordan iki yönlü akış olacaksa, koridor genişliği her iki yönde belirlenmiş akış için koridor genişliklerinin toplamı olarak alınabilir.

## **1.4 Malzeme Taşıma**

### *1.4.1 Malzeme Taşımanın Tanımı*

Bir fabrikada malzeme taşıma bütün çalışanların %25, fabrika boşluğunun %55 ve üretim zamanının %87' sine neden olmaktadır. Malzeme taşıma tahminen üretilmiş ürünün toplam maliyetinin %15 ile %70' ini temsil eder. Malzeme taşıma, maliyet azaltmada ilk sırayı alır.

Malzeme taşıma aynı zamanda kalite gelişmelerinde ilk sırayı alır. Tahmini olarak taşınan bütün malzemelerin %3 ile %5' i zarar görmektedir.

Tablo 1.2. Çeşitli Akış Türleri için Tavsiye Edilen Koridor Genişlikleri

| AKIŞ TIPI                                        | KORIDOR GENİŞLİĞİ (FEET) |
|--------------------------------------------------|--------------------------|
| Traktör                                          | 12                       |
| 3 ton Forklift                                   | 11                       |
| 2 ton Forklift                                   | 10                       |
| 1 ton Forklift                                   | 9                        |
| Dar Koridor Yük Arabası                          | 6                        |
| El ile Yönetilen Platform yük arabası            | 5                        |
| Personel                                         | 3                        |
| Koridora Bir Yönden Açılan Kapılarla<br>Personel | 6                        |
| Koridora İki Yönden Açılan Kapılarla<br>Personel | 8                        |

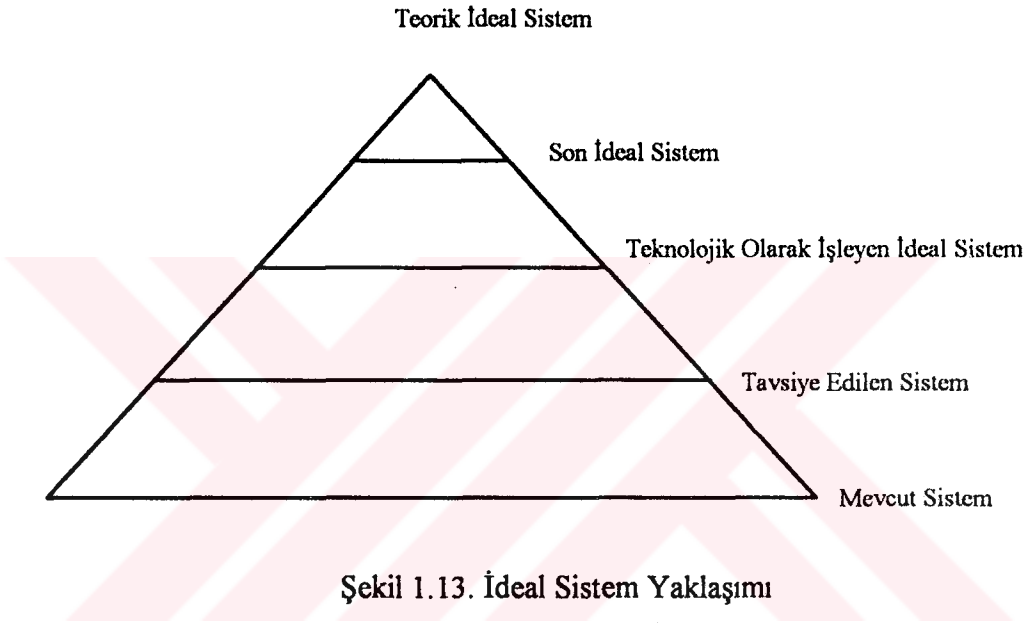
#### 1.4.2 Malzeme Taşıma Sistemi Tasarımı

Malzeme taşıma sistemi için mühendislik tasarım prosesi aşağıdaki aşamaları içerir:

1. Malzeme taşıma sistemi için amaçları ve çalışma alanını tanımla
2. Malzemeyi hareket ettirme, depolama, koruma, ve kontrol etmek için ihtiyaçları analiz et
3. Malzeme taşıma sistemi ihtiyaçlarını karşılamak üzere alternatif tasarımları genelle
4. Alternatif malzeme taşıma sistemlerini değerlendir
5. Malzemeyi hareket ettirme, depolama, koruma ve kontrol etmek için tercih edilen tasarımı seç
6. Tercih edilen tasarımı tamamla

Alternatif sistem tasarımının gelişimini göstermek için ideal sistem yaklaşımı aşağıdaki aşamaları içerir:

1. Teorik ideal sistem için amaç
2. Son ideal sistem için kavram oluşturma
3. Teknolojik olarak çalışabilir ideal sistemi tasarla
4. Tavsiye edilen sistemi kur



Teorik ideal sistem, sıfır maliyet, mükemmel kalite, emniyet tehlikeleri olmayan, boşa harcanmış boşluğa sahip olmayan ve yönetim etkisizliklerine sahip olmayan mükemmel sistemdir. Son ideal sistem, gelecekte bir noktada uygulanabilir fakat mevcut zamanda teknolojik yetersizliklerden dolayı uygulanması mümkün olmayan sistemdir. Teknolojik olarak işleyebilen ideal sistem gerekli teknolojinin bulunduğu fakat maliyetler veya diğer koşullar nedeniyle kurulması zor olan sistemdir. Tavsiye edilen sistem ise başarıyla kurulmasında engel olmayan maliyet etkili sistemdir.

Organizasyon tarafından taşınan malzemenin yolunda değişiklik yapılmasında sıkça kullanılan sebeplerden başlıcaları;

1. Maliyetleri azaltmak
2. Zararı azaltmak

3. Boşluk ve ekipman yararını arttırmak
4. Çıkışı arttırmak
5. Üretkenliği arttırmak
6. Çalışma koşullarını geliştirmek

Malzeme taşıma gelişmeleri yukarıdaki yararlarla sonuçlanmaktayken aşağıdaki maliyetlere de neden olur:

1. Sermaye ihtiyacının artması
2. Esnekliğin azalması
3. Emniyetin, bakımın ve işleme yeteneğinin azalması

Alternatiflerin kısa ve uzun vadeli değerlendirilmesi gerekir.

#### *1.4.2.1 Birim Yükler*

Malzeme taşımanın yirmi kuralından her biri değişik durumlar için taşımada ekonomiyi sağlamak için uygulanabilmesine karşılık birim yük kuralının özel önemi vardır. Birim yük kavramı birim boyut kuralından çıkarılır; birim yük bir defada taşınan veya hareket ettirilen birim olarak tanımlanır. Bazı durumlarda birim yük üretimin bir parçası olabilir; diğer durumdaysa birim yük bir çok sayıda parçadan oluşabilir.

Birim yük malzemeyi hareket ettirmede kullanılan konteynır, taşıyıcı veya desteği içerir. Birim yükün kullanılmasının temel avantajı bir defada bir çok parçanın taşınabilmesi ve hareket sayısının azaltılması, taşıma maliyetleri, yükleme ve boşaltma zamanlarının ve ürün hasarının azaltılmasıdır.

Birim yükün tasarlanmasında Apple aşağıdaki adımları önerir (Apple, J.M., 1977):

1. Birim yük kavramının uygulanabilirliğini belirle
2. Kullanılacak birim yük tipini seç
3. Potansiyel birim yüklemeye en uzak kaynağı belirle
4. Birim yük için en uzak yönü belirle
5. Birim yükün boyutunu belirle
6. Birim yükün konfigürasyonunu belirle
7. Birim yükü kurma metodunu belirle

Depolanacak parçalar ağırsa, ağırlık değerlendirmeleri hacim değerlendirmelerinden daha önemlidir. Ağırlık değerlendirmeleri yalnızca palet yapısal tasarımını bilgilendirmez, aynı zamanda taşıma ekipmanının ve ambar raflarının kapasitesini de belirler.

## 1.5 Yerleşim

### 1.5.1 Fabrika Yerleşim Problemine Genel Bakış

Fabrika yerleşim problemindeki kısıtları tanımlamak kolay değildir. Kısıtlar bir problemden diğerine değişir. Yerleşim problemleri bir çok yoldan ortaya çıkabilir ve üretim sisteminin geçerliliği üzerinde önemli etkisi vardır. Bunun için optimum fabrika yerleşim tasarımı arzulanır. Ne yazık ki, problemin boyutu o kadar büyüktür ki gerçek sistem optimizasyonu bizim kapasitemizin üstünde olur. Fabrika yerleşim problemi için araştırılan çözüm normal olarak bileşen araştırması kullanılarak tatmin edici bir çözüm bulunmasıdır. Bütün sistemi bileşenler ve alt sistemlerin birikimi olarak tanımlarız ve alt sistemler için optimum sonuçlara ulaşmaya çalışırız. Sonuçlanan sistem optimum yerine bir alt optimum çözümdür. Bileşen araştırmasının kullanılmasında sistemin bütününe zararlı bir bileşen için çözümün geliştirilmesi tehlikesi vardır. Bu olasılığı minimize etmek için ürün, proses, program ve yerleşim tasarımı kararlarını Şekil 1.11.' deki gibi koordine etmek gerekir.

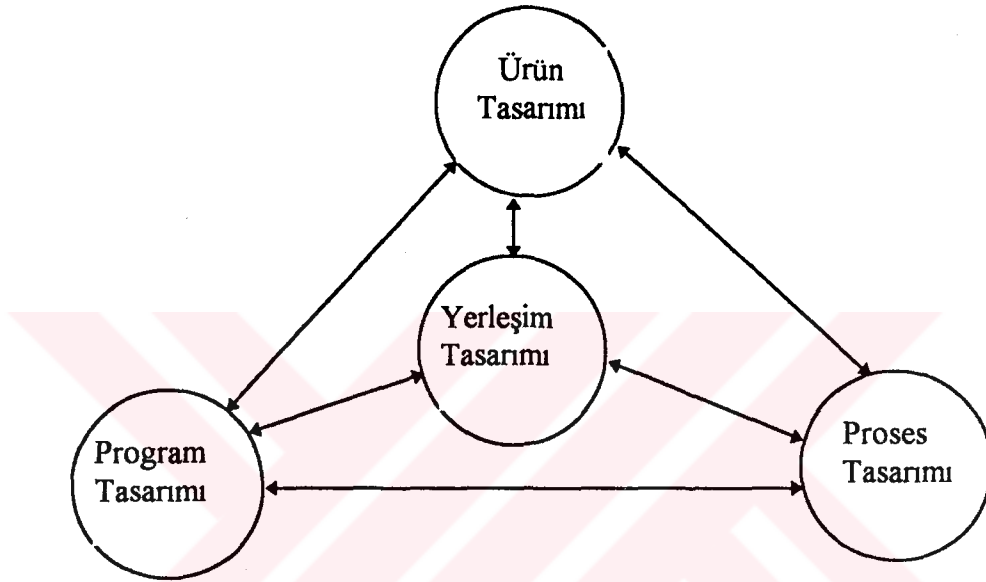
Fabrika yerleşimi aşamasındaki bazı amaçlar aşağıda verilmiştir:

- a. Ekipman yatırımını minimize etmek
- b. Toplam üretim zamanını minimize etmek
- c. Mevcut boşluklardan en etkili şekilde yararlanmak
- d. Çalışmanın emniyet, rahatlık ve uygunluğunu sağlamak
- e. Düzenleme ve işlemlerin esnekliğini korumak
- f. Malzeme taşıma maliyetini minimize etmek
- g. Malzeme taşıma ekipman türlerinin çeşitliliğini minimize etmek
- h. Üretim prosesini kolaylaştırmak
- i. Organizasyonel yapıyı kolaylaştırmak

Bunlara ek olarak analizcinin fabrika yerleşim problemini çözmesine yardımcı çeşitli amaçlar vardır. Aynı zamanda çözümde bir dizi kısıtta mevcuttur. Binanın şekli

yerleşimde bir kısıt olarak yer alabilir. Mevcut bina durumunda yerleşimin binaya uyma ihtiyacı ortaya çıkar. Yeni bina durumunda, bina alanı binanın şeklini ve yerleşimi sınırlayabilir. Yerleşimin tasarlanacağı mevcut bina durumunda, çözümde bir çok kısıt ortaya çıkar.

Yerleşim alternatiflerini genellemek, seçilen yerleşimin faaliyetler arası fiziksel ilişkileri temsil etmesi nedeniyle kritik bir adımdır.



Şekil 1.11.. Ürün, Proses, Program ve Yerleşim Arasındaki İletişim Hatları

### 1.5.2 Temel Yerleşim Türleri

Dört çeşit planlama bölümü vardır:

1. Sabit malzeme yerleşimli bölümler
2. Üretim hattı bölümleri
3. Üretim ailesi bölümleri
4. Proses bölümleri

Sabit malzeme yerleşim bölümleri diğer üçünden ayrılır. Diğer yerleşimlerde malzeme iş istasyonuna taşınırken, sabit malzeme yerleşimli bölümlerde iş istasyonu malzemeye taşınmaktadır. Bu aircraft montajında, gemi yapımında ve genellikle konstrüksiyon projelerinde kullanılır.

Ürün ailesi bölümü için yerleşim parçaları ürün aileleri oluşturmak için gruplamaya dayanır. Özdeş olmayan parçalar ortak prosesleme sırası, şekilleri, malzeme bileşenleri alet ihtiyaçları, taşıma/depolama/kontrol ihtiyaçları vb. dayanarak ailelere gruplanabilir. Ürün ailesi sahte ürün (pseudo product) olarak değerlendirilir ve sahte ürün yerleşimi geliştirilir. Sahte ürün için gerekli proses ekipmanı birlikte gruplanır ve üretim hücrelerine yerleştirilir. Sonuç yerleşimi bölüm içi akışın yüksek dereceli, bölümler arası akış ise azdır; genellikle ürün ailesi yerleşimi ve grup yerleşimi olarak adlandırılır.

Ürün hattı bölümü yerleşimi hatta üretilen parçaların proses sırasına dayanır. Malzemeler normal olarak bir iş istasyonundan takip eden istasyona akar. Yüksek hacimli envanterler için güzel, iyi planlanmış yollarla sonuçlanır. Bu tip yerleşimler ürün yerleşimleri olarak adlandırılır.

Proses bölümü için yerleşim benzer proseslerin bir araya getirilmesi ve ayrı proses bölümlerinin bölümler arası akışa bağlı olarak ilgili bölümlerle yerleştirilmesiyle elde edilir. Normal olarak, bölümler arası akış yüksek, bölüm içi akış ise azdır. Bu tip yerleşime proses yerleşimi denir ve ayrı parçalar veya grup yerleşimi oluşturamadığı faaliyet hacmi için kullanılır.

### 1.5.3 Yerleşim Prosedürü

Yerleşim planlamacılara yerleşim tasarlamada yardımcı olmak amacıyla bir dizi değişik prosedür geliştirilmiştir. Bu prosedürler iki ana kategoriye ayrılabilir: kuruluş ve geliştirme tipi. Kuruluş tipi metotları temel olarak başlangıçtan yeni bir yerleşim geliştirmeyi içerir. Geliştirme prosedürleri ise mevcut yerleşime dayanan yerleşim alternatiflerini geneller.

#### 1.5.3.1 Apple'ın Tesis Yerleşimi Prosedürü

Apple (1977) fabrika yerleşiminde aşağıdaki detaylı adım sıralamasını önermektedir:

1. Temel bilgiyi elde et
2. Temel bilgiyi analiz et
3. Verimli prosesi tasarla
4. Malzeme akış yolunu planla
5. Genel malzeme taşıma planını değerlendir
6. Ekipman ihtiyaçlarını hesapla
7. Ayrı iş istasyonlarını planla
8. Özel malzeme taşıma ekipmanını seç
9. Benzer işlem gruplarını koordine et
10. Faaliyet yakın ilişkileri tasarlamak
11. Depolama ihtiyaçlarını belirlemek
12. Servis ve yardımcı faaliyetleri planlamak
13. Boşluk ihtiyaçlarını belirlemek
14. Toplam boşlukta faaliyetlerin yerlerini tayin etmek
15. Bina tiplerini değerlendirmek
16. Ana yerleşimi kurmak
17. İlgili kişilerle yerleşimi değerlendir, kontrol et ve düzeltmek
18. Onaylamayı geçerli kılmak
19. Yerleşimi kurmak
20. Yerleşimin kuruluşunu takip etmek

### *1.5.3.2 Reed' in Fabrika Yerleşim Prosedürü*

Reed (1961) yerleşimi planlama ve hazırlama için aşağıdaki sistematik planlamayı önermektedir:

1. Üretilcek ürün veya ürünleri analiz et
2. Ürünü üretmek için gerekli prosese karar ver
3. Yerleşim planlama çizelgeleri hazırla
4. İş istasyonlarını belirle
5. Depolama alanı ihtiyaçlarını analiz et
6. Minimum koridor genişliklerini belirle
7. Ofis ihtiyaçlarını belirle
8. Personel servisleri ve tesislerini değerlendir
9. Fabrika servislerini ölç
10. Gelecekteki genişlemeleri hesapla

Reed yerleşimi planlama çizelgesine “bütün yerleşim proseslerinin en önemli tek adımı” demektedir. Aşağıdaki gibidir:

1. İşlemleri, taşımayı, depolama ve kontrolü içeren akış prosesi
2. Her işlem için standart zaman
3. Makine seçimi ve dengeleme
4. İnsan gücü seçimi ve dengeleme
5. Malzeme taşıma ihtiyaçları

### *1.5.3.3 Muther' ın Sistematik Yerleşim Planlama Prosedürü*

Giriş bilgisine ve rollerin anlaşılmasına ve faaliyetler arası ilişkilere dayanarak, malzeme akış analizi (-den -e çizelgesi) ve faaliyet ilişki çizelgesi analizi (faaliyet ilişki çizelgesi) yapılır (Muther, 1973). Analizlerin sonucunda ilişki diyagramı geliştirilir.

İlişki diyagramı faaliyetleri uzaysal olarak yerleştirilir. Yakınlıklar, faaliyet çiftleri arasındaki ilişkileri yansıtmada kullanılır. Sonraki iki adım her faaliyete atanacak boşluğun miktarını belirlemeyi içerir. Her planlama bölümü için bölüm servis ve alan ihtiyaç levhaları tamamlanmalıdır. Boşluk atamaları yapıldıktan sonra

boşlukları her planlama bölümü için geliştirilir ve boşluk boşluk-ilişki diyagramını elde etmek için ilişki diyagramına eklenir.

Değiştirme yaklaşımları ve pratik sınırlara dayanarak bir dizi yerleşim alternatifi geliştirilmiştir. Tercih edilen alternatif daha sonra belirlenir.

#### 1.5.3.4 Algoritmik Araştırmalar

Yakın bölümlerin yakınlık oranları veya malzeme akış duyarlılıklarına göre yerleştirilmesi algoritmik prosese indirgenebilir. Aşağıdaki üç metot incelenecektir:

1. İlişki diyagramlama
2. Çifte değiştirme metodu
3. Grafik tabanlı kuruluş metodu

İlişki diyagramlama sistematik yerleşim planlamanın Apple' ın önerdiği bazı düşüncelerden yararlanmış bir çeşididir. Çifte değiştirme metodu Reed tarafından verilen dolaşım çizelgesi metodu ve Buffa' nın CRAFT prosedürünü temel alır. Grafik tabanlı kuruluş metodunun kökleri ise Grafik teorisindedir.

#### 1.5.4 Yeni Yerleşimler için İlişki Diyagramı Oluşturma

İlişki diyagramı geliştirmede, birim blok her bölümü temsil etmek üzere kullanılacaktır.

1. Adım: Yerleşime başlamak için ilk bölümü seç. En çok "A" ilişkisine sahip bölüm seçilir. Eğer bir eşitlik varsa eşitlik bozma kuralı şu karar hiyerarşisinde izlenir: en büyük "E" ilişkisi, en büyük "I" ilişkisi, en az "X" ilişkisi ve son olarak rasgele seçim.

2. Adım: Yerleşime girecek 2. Bölümü seç. Seçilecek 2. bölümün ilk bölümle "A" ilişkisine sahip olması gerekir. Ek olarak, seçilmemiş bölümlerle en fazla "A" ilişkisine sahip olmalıdır. Eşitlikler, eşitlik bozma kuralıyla (1. Adımdaki) bozulur.

3. Adım: Yerleşime girmek için 3. Bölümü seç. Üçüncü bölümün seçimi, seçilmiş olan iki bölümle en yüksek ilişki ile birleştirilmiş olmasına bağlıdır.

### 1.6 Bilgisayar Destekli Yerleşim

#### 1.6.1 Algoritma Sınıflandırma

Fabrika yerleşimi alanında oluşturulan bir çok algoritma sezgisel metodlara dayanmaktadır. Heragu ve Kusiak (1991) bu algoritmaları dörde ayırır:

1. Kuruluş algoritmaları,
2. Geliştirme algoritmaları,
3. Melez algortimalar,
4. Kuramsal grafik algoritmalar.

Bir çok yerleşim algoritmaları ihtiyaç duydukları giriş bilgilerine göre sınıflandırılır. Bazı algoritmalar sadece niteliksel akış bilgisini kabul ederken, diğerleri (niceliksel) akış matrisi (-den -e çizelgesi) ile çalışırlar. Bazı algoritmalar (BLOCPLAN gibi) ilişki çizelgesi ve -den -e çizelgesini ortak kullanır; çizelgeler yerleşimi değerlendirirken her seferinde bir defada kullanılır. Şu anki eğilim hazırlamada daha fazla zaman ve çaba gerektiren fakat tamamlandığında parça akışı (veya malzeme taşıma sayısı) hakkında daha fazla bilgi veren -den -e çizelgelerinin kullanılması yönündedir. Bu akış değerleri ilişki oranlarına aktarılır ve bir çok algoritma her iki bilgiyi de kullanabilir. Tabii ki eğer ilişki çizelgesi -den -e çizelgesine aktarılırsa (yakınlık oranlarına sayısal değerler atayarak), yerleşim planlaması tarafından alınan akış değerleri sadece sıra değerlerini temsil eder.

Yerleşim algoritmaları, aynı zamanda amaç fonksiyonlarına göre de sınıflandırılabilir. İki temel amaç mevcuttur: biri akış x mesafelerin toplamını minimize etmeyi amaçlarken diğeri yakınlık skorunu maksimize etmeyim amaçlar. Genellikle uzaklık tabanlı amaç -klasik Kuadratik Atama Problemi amacı- eğer giriş bilgisi “-den -e” çizelgesi olarak veriliyorsa daha uygundur. Eğer giriş bilgisi ilişki çizelgesiye yakınlık tabanlı amaç daha uygundur.

İlk olarak uzaklık tabanlı amacı ele alalım. M bölüm sayısı olsun,  $f_{ij}$  i bölümünden j bölümüne akış (birim zamanda hareket eden birim yük sayısı ile ifade edilir) ve  $c_{ij}$  i bölümünden j bölümüne birim yükün birim mesafede hareket ettirilme maliyeti olsun. Amaç, bölümler arasında hareket için birim zamandaki maliyeti minimize etmektir. Matematiksel olarak amaç,

$$\min. \sum_{i=1}^m \sum_{j=1}^m f_{ij} c_{ij} d_{ij} \quad (1.1.)$$

burada  $d_{ij}$  i bölümünden j bölümüne uzaklıktır. Bir çok yerleşim algoritmasında  $d_{ij}$  bölüm merkezlerinden düzgün doğrusal olarak ölçülür; ayrıntılı koridor yapısına göre de ölçülebilir.

Denklemden  $c_{ij}$  değerlerinin taşıma ekipmanının kullanımından tamamen bağımsız, ve hareketin uzunluğunun doğrusal olduğu varsayımları yapılır.  $c_{ij}$  değerlerinin yukarıdaki kabulleri sağlamadığı durumlarda bütün i ve j' ler için  $c_{ij} = 1$  seçilebilir ve tesiste dolaşan toplam birim yük,  $f_{ij}$ ' nin ürünü ve  $d_{ij}$  değerleri üzerine odaklanılabilir. Bazı durumlarda  $c_{ij}$  değerlerini görece ağırlık olarak kullanarak tesiste dolaşan birim yükün ağırlık toplamını minimize etmeye çalışılabilir.

İkinci olarak birbirine yakın bölümler arasında bütün akış değerlerinin toplamı olarak hesaplanan yakınlık skorunda yakınlık tabanlı amacı inceleyelim. i ve j bölümlerinin yerleşimde birbirine yakın olması (aynı sınırı paylaşması) durumunda 0 olsun. Yakınlık skorunu maksimize etmek olan amaç fonksiyonu

$$\max. \quad z = \sum_{i=1}^m \sum_{j=1}^m f_{ij} x_{ij} \quad (1.2.)$$

Yukarıdaki denklemden elde edilen yakınlık skoru iki veya daha fazla alternatif yerleşimi kıyaslamada yardımcı olmasına rağmen ayrıntılı yerleşimin görece etkililiğini belirlemek için belli alt ve üst sınır istenir. Bu amaç için, yerleşim planlamacı aşağıdaki normalleştirilmiş yakınlık skorunu kullanabilir:

$$z = \frac{\sum_{i=1}^m \sum_{j=1}^m f_{ij} x_{ij}}{\sum_{i=1}^m \sum_{j=1}^m f_{ij}} \quad (1.3.)$$

Normalleştirilmiş yakınlık skoru (etkililik oranı olarak da bilinir) 7.2. denklemden elde edilen yakınlık skorunun tesisteki toplam akışa bölünmesiyle elde edilir. Sonuç olarak normalleştirilmiş yakınlık skoru her zaman 0 ile 1 arasındadır. Eğer normalleştirilmiş yakınlık skoru 1 ise (ararlarında pozitif akış olan) bütün bölüm çiftleri yerleşimde yakındır.

Bazı durumlarda yerleşim planlamacı negatif  $f_{ij}$  değeri atayarak i ve j bölümlerinin arasında X ilişkisi sunabilir. Kullanılacak tam negatif değer -den -e

çizelgesinde gerçek akış değerleriyle bulunur. Eğer bu gibi negatif akış değerleri kullanılacaksa, normalleştirilmiş yakınlık skoru

$$z = \frac{\sum_{(i,j),F} f_{ij} x_{ij} - \sum_{(i,j),\bar{F}} f_{ij} (1 - x_{ij})}{\sum_{(i,j),F} f_{ij} - \sum_{(i,j),\bar{F}} f_{ij} (1 - x_{ij})} \quad (1.4.)$$

F ve  $\bar{F}$  pozitif ve negatif akış değerlerine sahip bölüm çiftlerinin grubunu ifade etmektedir.

Yakınlık tabanlı amaç bir dizi algoritmada kullanılmıştır. Bu tür bir amaç kullanım yönünden kolay ve sezgisel olmasına rağmen, uzaklığı ve yakın olmayan bölümler arasındaki ayrılmayı önemsememesi nedeniyle yerleşim etkinliğinin ölçütü olamaz. Bu nedenle, aynı veya benzer yakınlık skoruna sahip fakat parça akışı cinsinden dolaşım mesafelerinin farklı olduğu iki yerleşimi kurmak mümkündür.

Bilgisayarla yerleşim algoritmaları daha sonra yerleşim sunuşundaki düzenlerine göre sınıflandırılabilir. Bir çok yerleşim algoritmaları bilgisayarın yerleşimde matris gibi manevra yapmasını ve depolamasını sağlayan ayrı sunuş kullanır. Bu tür sunuşla her bölümün ızgaralara en yakın tam sayıya yuvarlanır. Eğer ızgara boyutu çok büyükse (çok küçükse), küçük (büyük) bölümler çok az (çok fazla) ızgaraya sahip olur. Aynı zamanda ızgara boyutu yerleşimin bütün çözümünü belirler; küçük ızgara boyutu bölüm şekillerinde esnekliğe sahip daha iyi çözünürlük sunar. Bu nedenle toplam alan sabit olduğundan daha küçük ızgara boyutu hesabın ağırlığını artmasını sağlayacak daha çok sayıda ızgaralara (veya daha büyük matris) neden olur. Aynı sunuşlu algoritmalarda, tahsis edilen ızgara boyutu planlama prosesinin başında yapılması gereken önemli bir karardır.

Alternatif sunuş, temeli oluşturacak hiçbir ızgara yapısı olmayan sürekli sunuştur. Bu tür sunuş, aynı sunuşa göre teorik olarak daha çok esnekliğe sahip olmasına rağmen bilgisayara kurulması zordur. Mevcut yerleşim algoritmaları bir durum hariç (belli bölümlerin şekillerinin dikdörtgen olmayan binayı düzeltmek için ayarlama) dikdörtgen bina ve dikdörtgen bölüm şekilleriyle sınırlıdır.

Son olarak bilgisayar tabanlı yerleşim algoritmaları öncelikli fonksiyonlarına göre sınıflandırılabilir; bu, yerleşim geliştirmeye karşı kuruluş yerleşimidir. Geliştirme

algoritmalarına genellikle analizci tarafından belirlenen bir başlangıç yerleşimiyle başlanır ve yerleşimdeki değişimlerle amaç fonksiyonundaki gelişmeler gözlenir. Kuruluş algoritmaları genellikle sıfırdan yerleşimi geliştirirler. Daha sonra iki kategoriye ayrılırlar: bina boyutlarının verildiğini kabul edilenler ve verilmediği kabul edilenler. Birinci çeşit kuruluş algoritmalar işletmenin mevcut bir binaya taşınması durumunda uygundur. İkinci tür kuruluş algoritması ise “yeşil alan” uygulamaları için geçerlidir.

### 1.6.2 BLOCPLAN

BLOCPLAN Donaghey ve Pire (1990) tarafından geliştirilmiştir ve bölümlerin bantlarla düzenlenmesi yönünden MCRAFT’ a benzer. Bununla birlikte, iki algoritma arasında belirgin bazı farklılıklar vardır. BLOCPLAN -den -e çizelgesi kadar ilişki çizelgesini de akış için giriş bilgisi olarak kullanır. Yerleşim maliyeti uzaklık tabanlı amaca veya bant yakınlık tabanlı amaca göre ölçülebilir. Ayrıca BLOCPLAN’ da bant sayısı program tarafından belirlenmekte ve iki veya üç bantla sınırlandırılmaktadır. Aynı zamanda, BLOCPLAN’ da her bölüm bir bantı işgal ettiğinden, bütün bölümler dikdörtgen şeklindedir. Son olarak, MCRAFT’ tan farklı olarak BLOCPLAN sürekli temsili kullanır.

BLOCPLAN kuruluş ve geliştirme algoritması olarak kullanılabilir. Son durumda, MCRAFT’ la olduğu gibi, başlangıç yerleşimini doğru olarak elde etmek mümkün olmayabilir. Bu nedenle yerleşimdeki gelişmeler, bölüm değişimlerinde görülebilmektedir. Program hem ilişki çizelgesi hem de -den -e çizelgesini girdi olarak kabul ettiğinden, iki çizelge yerleşim değerlendirileceği zaman her seferinde bir defa kullanılır. Bu, yerleşimin iki çizelgenin bazı kombinasyonlarına göre değerlendirilemeyeceğini belirtir.

BLOCPLAN ilk olarak her bölümü iki veya üç banttan birine atar. Her bölümün bir kısmı banda atanmasından sonra, BLOCPLAN ayrılmış bant kalınlığını bölümlerin toplam alanını bina uzunluğuna bağlı banda bölerek hesaplar.

### 1.6.3 LOGIC

LOGIC (Layout Optimization with Guillotine Induced Cuts) Tam (1991) tarafından geliştirilmiştir. LOGIC’ i tanımlamada, akış için bir -den -e çizelgesinin giriş

bilgisi olarak verildiği kabulü yapılır. Aynı zamanda yerleşim maliyetinin (7.1) denkleminde gösterildiği gibi uzaklık-tabanlı amaç fonksiyonuyla ölçüldüğünü kabul ederiz. LOGIC tarafından genellenen bölümler dikdörtgen binalarda dikdörtgen bölümlerdir. Yerleşim sürekli türde sunulur.

LOGIC geliştirme algoritması olarak kullanılabilmesine rağmen burada ilk olarak kuruluş algoritması olarak sunulacaktır. LOGIC, binayı birbirini izleyen giyotin kesişleriyle küçük parçalara bölmek yoluna dayanır. Her kesiş yatay veya dikey olabilir. Eğer dikeyse, bölüm doğu veya batı tarafına atanır. Bina kalınlığının (veya onun parçasının) ve atanan bütün bölümlerin toplam alanının verilmesiyle, dikey kesişin doğu kısmı diyelim, kesimin tam x koordinatı hesaplanabilir.

LOGIC bir dizi yatay ve dikey kesişle uygulanır. Her kesişle, bölümlerin uygun alt dizileri kesimin doğu-batı veya kuzey-güney tarafına atanır.

#### 1.6.4 ALDEP

ALDEP IBM' de geliştirilmiş ve özgün olarak Seehof ve Evans tarafından sunulmuştur (Seehof, J.M., Evans, O., 1967). ALDEP bir kuruluş algoritmasıdır. ALDEP, tasarıma yerleştirme düzeni programları gibi mevdut bir yerleştirme düzenine gereksinim olmaksızın yerleşim düzenlerini tasarlar.

ALDEP' in birçok uygulama biçimleri olmasına rağmen burada rastgele seçim yöntemi ele alınacaktır. ALDEP' in rastgele seçim yöntemi, bir bölümün rastgele seçimi ve yerleştirme düzeni içine yerleştirilmesiyle bir düzen tasarımı gerçekleştirmektedir. Sonra ilişki çizelgesi incelenerek yüksek derecede yakınlık gösteren bir bölüm düzen içine yerleştirilir. Bu işlem tüm bölümler yerleştirilinceye kadar sürdürülür.

Bilinen ALDEP programının yakınlık derecelerine göre verilmiş sabit sayısal değerleri vardır. Bu değerler;

$$A=4^3 = 64$$

$$O= 4^0 = 1$$

$$E= 4^2 = 16$$

$$U= 0$$

$$I= 4^1 = 4$$

$$X= -4^5 = -1024,$$

ALDEP 63 bölüm ya da eyleme kadar olan çözümleri gerçekleştirme yeteneğindedir ve üç kata kadar yerleştirme düzenleri geliştirebilir. ALDEP için gerekli girdiler aşağıdaki gibi sıralanabilir:

- i. Her bir kat için uzunluk, genişlik ve alan gereksinimleri,
- ii. Yerleştirme düzeni basılı çıktısının ölçeği,
- iii. Yerleştirme düzenindeki bölüm sayısı
- iv. Türetilecek yerleştirme düzeni sayısı,
- v. Kabul edilebilir bir yerleştirme düzeni için istenen en küçük puan,
- vi. En küçük bölümün tercih edilmesi,
- vii. Her bir kat için kısıtlanmış bölgelerin yerleşimi ve boyutları.

## 1.7 Niceliksel Tesis Planlama

### 1.7.1 Tesis Yerleşim Modelleri

Yerleşim kararlarında çeşitli amaçlar kullanılmaktadır. Örneğin yeni tesisle mevcut tesis arasındaki ağırlıklı mesafenin toplamının minimize edilmesi (mini toplam yerleşim problemi) ve yeni tesisle mevcut tesis arasındaki maksimum mesafenin minimize edilmesi gibi amaçlar kullanılır (Francis, R. L., White, J.A., 1974).

Tesis yerleşim problemlerinde uzaklık ölçümü analitik modeli formüle etmede önemli bir elemandır. Uzaklık ölçümleri aşağıdaki gibi sınıflandırılabilir:

1. Uzaklıkların diğerlerine ortogonal (veya dikey) olduğu düzgün doğrusal: bu ölçüm şehirdeki bir çok yolun birbirine paralel veya dikey olması nedeniyle Manhattan mesafesi olarak adlandırılır.
2. Uzaklıkların iki nokta arasındaki düz bir hatta ölçüldüğü Öklidyen (veya düz hat).
3. Uzaklıkların iki nokta arasında güncel yolu çapraz kateden akış yolu.

### 1.7.2 Düzgün Doğrusal Tesis Yerleşim Problemi

Aşağıdaki notasyon kullanılır:  $X = (x,y)$  yeni tesisin yeri

$P = (a_i, b_i)$   $i$  mevcut tesisinin yeri,  $i = 1, 2, \dots, m$

$W_i =$  oransal sabit

$d(X, P_i)$   $i$  mevcut tesisile yeni tesis arasındaki mesafe

Amaç,

$$\text{minimize } f(X) = \sum_{i=1}^m W_i d(X, P_i) \quad (1.5)$$

Düzgün doğrusal modelde, uzaklık koordinatlar arası mesafenin toplamıyla ölçülür.

$$d(X, P_i) = |x - a_i| + |y - b_i| \quad (1.6.)$$

### 1.7.3 Tek Tesis Minitoplama Yerleşim Problemi

Minitoplama yerleşim problemi aşağıdaki şekilde formüle edilebilir:

$$\text{minimize } f(X) = \sum_{i=1}^m W_i |x - a_i| + \sum_{i=1}^m W_i |y - b_i| \quad (1.7.)$$

$X'$  in optimum değerini bulmak için, iki matematiksel özellik kullanılır. Yeni tesisin  $x$  koordinatı bazı mevcut tesislerin  $x$  koordinatlarıyla aynı olacaktır; ve optimum  $x$  koordinatı öyle olacaktır ki toplam ağırlığın yarısı  $x'$  in soluna veya sağına kaymayacaktır. Son durum orta durum olarak adlandırılır. Her iki özelliğe  $y$  koordinatı için kullanılır.

### 1.7.4 Tek Tesis Minimaks Yerleşim Problemi

Minimaks problemi:

$$\text{minimize } f(X) = \max \left[ (|x - a_i|, |y - b_i|), i = 1, 2, \dots, m \right] \quad (1.8.)$$

minimaks çözümü elde etmek için

$$C_1 = \text{minimum } (a_i, b_i) \quad (1.9.)$$

$$C_2 = \text{maksimum } (a_i, b_i) \quad (1.10.)$$

$$C_3 = \text{minimum } (a_i, b_i) \quad (1.11.)$$

$$C_4 = \text{maksimum } (a_i, b_i) \quad (1.12.)$$

Minimaks yerleşim probleminin optimum çözümü doğru parçasında bütün noktaların birleşim noktası,

$$(x_1^*, y_1^*) = 0.5 (C_1 - C_3, C_1 + C_3 + C_5) \quad (1.13.)$$

$$(x_2^*, y_2^*) = 0.5 (C_2 - C_4, C_2 + C_4 + C_5) \quad (1.14.)$$

$$C_5 = \max (C_2 - C_1, C_4 - C_3) \quad (1.15.)$$

maksimum uzaklık  $C_5/2$ ' ye eşittir.

### 1.7.5 n Tesis Yerleşim Problemi

$n$  tesis yerleşim probleminin genel formülü

$$\text{minimum } f(X) = \sum_{j=1}^n \sum_{k=1}^n V_{jk} d(X_j, X_k) + \sum_{j=1}^n \sum_{i=1}^m W_{ji} d(X_j, P_i) \quad (1.16.)$$

burada

$X = (x_i, y_i)$  j yeni tesisinin yeri,  $j = 1, 2, \dots, n$

$P = (a_i, b_i)$  i mevcut tesisinin yeri,  $i = 1, 2, \dots, m$

$V_{jk}$  katedilen birim mesafe başına maliyet (yeni tesis j ve k arasında)

$W_{ji} = j$  ve  $i$  arasındaki mesafe

i. Düzgün doğrusal mesafe ile

$$d(X_j, X_k) = |x_j - x_k| + |y_j - y_k| \quad (1.17.)$$

$$d(X_j, P_i) = |x_j - a_i| + |y_j - b_i| \quad (1.18.)$$

ii. Öklidyen mesafe ile

$$d(X_j, X_k) = [(x_j - x_k)^2 + (y_j - y_k)^2]^{1/2} \quad (1.19.)$$

$$d(X_j, P_i) = [(x_j - a_i)^2 + (y_j - b_i)^2]^{1/2} \quad (1.20.)$$

Terimlerin tam değerlerle dönüştürülmesiyle, n tesis düzgün doğrusal mesafe yerleşim problemi doğrusal programlama modelleriyle çözülebilir.

birimle ilişkisi en yoğun olan birimin yerleştirilmesiyle devam edilir. Lee ve Held (1967) CORELAP programını geliştirdiler. Karp ve Held (1967) fabrika yerleşim problemini dinamik programlama problemi olarak modellediler. El rayah ve Hollier (1970) üç değişik fabrika yerleşimini tartıştılar ve kuadratik atama problemi için optimal ve alt optimal algoritmaları derlediler. Pierce ve Crowston (1971) kuadratik atama probleminin çözümü için optimal algoritmayı geliştirdiler. Deisenworth ve Apple (1972) üç aşamalı tasarım olan PLANET' i geliştirdiler. Hannan ve Kurtzberg (1972) kuadratik atama problemi için algoritmalar üzerinde çalıştılar. Zoller ve Adendorf (1972) simulatör yardımı ile iki boyutlu yerleşim alternatiflerini hazırladılar. Bazaara (1975) genel yerleşim problemleri için üretimin gerçekleştirildiği alanın numaralandırılmış bloklara bölündüğü bir hesaplama yöntemi olan QSP'yi (Quadratic Set Covering) geliştirdi. Love ve Wong (1976) doğrusal programlama ve tamsayılı programlama yardımı ile bir model geliştirdiler. Burkard ve Stratman (1978), kuadratik atama problemi için alt algoritmaları dahil ettiler ve sayısal sonuçlar kullanarak Pierce ve Crowsto' un algoritmalarını test ettiler. Kaufman ve Broeckx (1978) kuadratik atama probleminin tamsayılı programlar arasında değişken ve kısıtlayıcı miktarı en az olan yeni bir doğrusal karışık tam sayılı program geliştirdiler. Ritzman et al. (1978) bina içindeki büroları yerleştiren karışık tam sayılı hedef programlama modelini formüle etti. Drezner (1980) konveks olmayan matematik programlama modelini formüle etti. Giffi ve Foulds (1984) yakınlık ilişkilerini içeren bir seti bir yerleşim diyagramına dönüştüren metodu geliştirdiler. Heragu ve Kusiak (1987) optimal ve sezgisel yaklaşımla ilgili algoritmalar geliştirmişler ve bunları çok kanallı üretim düzenlerinde uygulamışlardır. Yine Heragu ve Kusiak (1988) tek sıra yerleşimi için sezgisel bir algoritma sunmuşlardır. Urban (1989) fabrika düzenlemesi üzerine niceliksel ve niteliksel analizlerin birarada ele alındığı bir çalışma yapmıştır. Heragu ve Kusiak (1990), makina yerleşimi konusunda bilgi tabanlı bir yöntem (KBML) sunmuşlardır. Housyar (1991) yine niceliksel ve niteliksel analizleri bir arada problem çözümüne almış ve bunun için de birbirini etkileyen bir çok amaçlı yaklaşımdan yola çıkmıştır. Logendran (1991) hücrel üretimde hücre içi hareketleri göz önüne alarak hücreler arası hareketlerle birlikte düzenleme işlevine etkisini

incelemiştir. Heragu ve Kusiak (1991) fabrika yerleşim problemi için amaç fonksiyonunda kesin değerlere sahip doğrusal sürekli ve doğrusal karışık tamsayılı iki yeni model sundular.



### 3. MATERYAL VE METOD

#### 3.1 Materyal

Bu çalışmada orta ölçekli bir sanayinin tesis yerleşim problemi ele alınmıştır. Uygulama yapılan işletmede vidalı kompresör, pistonlu kompresör ve hidrofor üretilmektedir. Mevcut yerlerinin darlığı nedeniyle yeni alınan bazı makinelerin monte edilecek yer bulunamaması sonucu işletme yetkilileri yeni bir yer satın alarak fabrikanın tamamını 3 yıllık bir süre içinde yeni yere taşımak istemektedirler. Bunun için de fabrika yerleşim metotlarından yararlanılacaktır. İşletmede dokuz bölüm bulunmaktadır: Saç, Kaynak, Boya, Ambar, Talaşlı İmalat, Mekanik Montaj, Vidalı Kompresör Montaj ve Test, Pistonlu Kompresör Montaj ve Test, Hidrofor Montaj ve Test. Taşıma sistemi olarak vinç, konveyör ve elle taşıma bulunmaktadır.

#### 3.2 Metod

Burada iki metod yardımıyla uygulama yapılmıştır; CRAFT ve MIP. Bu metodlarla ilgili ayrıntılı bilgi verilecektir, fakat bu iki metotta da başlangıç bilgilerinde yer alan “den -e” çizelgesi uygulama için öncelikli olarak hazırlanmıştır (Şekil 3.1.). Ayrıca niteliksel bir yaklaşım olarak REL (ilişki) çizelgeside uygulama yapılan işletme için hazırlanmıştır (Şekil 3.2.).

##### 3.2.1 Akış Ölçümü

Akış niceliksel ve niteliksel yöntem olarak ölçülebilir. Niceliksel ölçümler saat başına parça, günlük hareket vb. içerir. Niteliksel ölçümler iki bölümün yan yana olma gereksinimi vs. içerir.

##### 3.2.1.1 Niceliksel Akış Ölçümü

Akışlar niceliksel olarak bölümler arası hareket miktarıyla ölçülür. Bunları kaydetmede en çok “-den -e” çizelgeleri kullanılır.

Bir “-den -e” çizelgesi aşağıdaki şekilde kurulur

1. Bütün akış yolunu izleyecek şekilde bölümleri satır ve sütunlara yaz
2. Tesis için eşdeğer akış hacimlerini tam olarak temsil eden bir akış ölçümü oluşturun.

3. Eğer akış hareketle ölçülecekse, -den -e çizelgesine yapılan hareket sayısı işlenebilir. Eğer çeşitli boyut, ağırlık, değer, zarar riski, şekil vs.' ye sahipse -den -e çizelgesi hareket hacimlerinin uygun ilişkileriyle temsil edilir.

4. Akış yollarına ve temsili akış ölçümlerine, akış hacimlerini -den -e çizelgesine kaydet.

#### 3.2.1.2 Niteliksel Akış Ölçümü

Akış niteliksel olarak Muther tarafından geliştirilen yakınlık ilişkileri değeriyle ölçülebilir (Tablo 3.1.).

Tablo 3.1. Yakınlık İlişki Değerleri

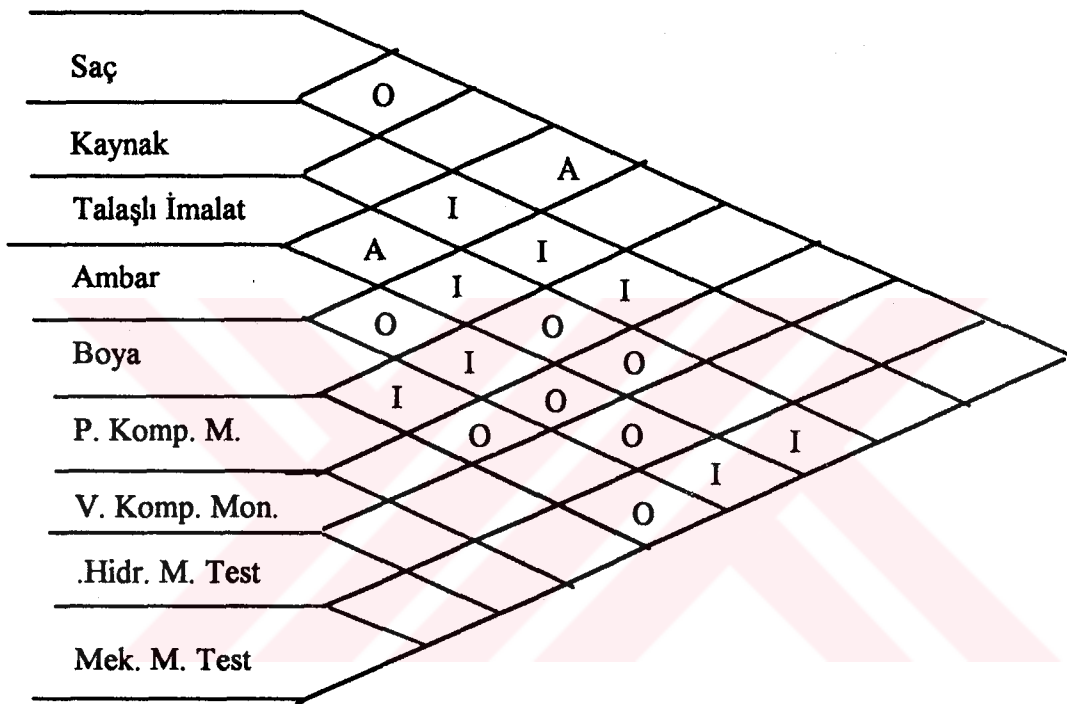
| DEĞER | YAKINLIK           |
|-------|--------------------|
| A     | Kesinlikle gerekli |
| E     | Özellikle önemli   |
| I     | Önemli             |
| O     | Sıradan            |
| U     | Gereksiz           |
| X     | İstenmeyen         |

İlişki çizelgesi aşağıdaki şekilde kurulur:

1. İlişki çizelgesine bütün bölümleri listele
2. Listedeki her bölümden insanlarla ve bütün bölümlerden sorumlu yöneticiyle görüşme yap
3. Yakınlık ilişkileri atamada kullanılan kriteri tanımla, ayrıntılı olarak yaz ve ilişki çizelgesinde ilişki değerlerinin nedenleri olarak kriteri kaydet
4. Her bölüm çifti için ilişki değerini ve değer in nedenini oluştur.
5. Herkesi ilişki çizelgesinin geliştirilmesinde katkısını ve değişiklikleri değerlendirmelerini ve tartışmalarını sağla.

| -den \ -e                            | Saç | Kaynak | Talaşlı<br>İmalat | Ambar | Boya<br>Bölümü | Pistonlu<br>Kompresör | V. Kompresör<br>Testve Montaj | Mekanik<br>Montaj ve<br>Test | Hidrofor<br>Montaj<br>ve Test |
|--------------------------------------|-----|--------|-------------------|-------|----------------|-----------------------|-------------------------------|------------------------------|-------------------------------|
| Saç                                  |     | 4      | -                 | 1     | -              | -                     | -                             | -                            | -                             |
| Kaynak                               |     |        | -                 | 5     | 5              | 5                     | -                             | -                            | -                             |
| Talaşlı<br>İmalat                    |     |        |                   | 10    | 10             | 5                     | 5                             | 10                           | -                             |
| Ambar                                | 20  | 10     | 30                |       | 2              | 10                    | 8                             | 5                            | 10                            |
| Boya                                 |     | 10     |                   | 5     |                | 10                    | 4                             |                              | 5                             |
| Pistonlu<br>Kompresör                |     |        |                   | 10    | 6              |                       |                               |                              |                               |
| V.<br>Kompresör<br>Test ve<br>Montaj |     |        |                   | 10    |                |                       |                               |                              |                               |
| Mekanik<br>Montaj ve<br>Test         |     |        |                   | 5     | 10             |                       |                               |                              |                               |
| Hidrofor<br>Montaj ve<br>Test        |     |        |                   | 5     |                |                       |                               |                              |                               |

Şekil 3.1. İşletmenin Bölümler Arası Malzeme Akış Çizelgesi (-den -e çizelgesi)



| Değer | Yakınlı            |
|-------|--------------------|
| A     | Kesinlikle Gerekli |
| E     | Özellikle önemli   |
| I     | Önemli             |
| O     | Sıradan            |
| U     | Önemsiz            |
| X     | İstenmeyen         |

**Şekil 3.2. Uygulama Yapılan İşletme İçin İlişki Çizelgesi**

### 3.2.2 CRAFT

Vollman, Armour ve Buffa (1963) tarafından sunulmuştur. CRAFT (Computerized Relative Allocation of Facilities Technique) literatürde sunulmuş etkili algoritmalarından biridir. Akış için bir -den -e çizelgesini giriş bilgisi olarak kullanır. Yerleşim maliyeti 1.1 denkleminde gösterilen uzaklık tabanlı amaç fonksiyonuyla ölçülür. Bölümler dikdörtgen şekillerle sınırlı değildir ve yerleşim ayrı şekilde sunulur.

CRAFT geliştirme tipi yerleşim algoritması olduğundan, mevcut yerleşim tesisin güncel yerleşimini başlangıç çözümü olarak sunmayla başlayabileceği gibi bir başka algoritma tarafından geliştirilmiş yerleşimi de başlangıç yerleşimi olarak sunabilir. CRAFT başlangıç çözümünde bölümlerin merkezlerini belirlemeye başlar. Daha sonra bölüm merkezleri çiftlerinin arasındaki düzgün doğrusal mesafeyi ölçer ve uzaklık matrisinde gösterir. Başlangıç yerleşimi maliyeti -den -e çizelgesinde her girişle birim matris ve uzaklık matrisinde karşılık gelen girişleri çarparak elde edilir.

CRAFT daha sonra bütün olası iki-yol veya üç-yol bölüm değişimlerini göz önüne alır ve en iyi değişimi, yerleşim maliyetinde en çok azalmayı sağlayanı seçer. Hiçbir bölüm iki yol veya üç yol değişimi sırasında aynı kalmaz.

Bir defa en iyi değişim belirlendikten sonra, CRAFT yerleşimi en iyi değişime göre günceller, ve yeni bölüm merkezini yeni yerleşim maliyetini de hesaplayarak birinci iterasyonu tamamlar. Sonraki iterasyon bir defa daha güncellenmiş yerleşimde bütün olası iki-yol veya üç-yol değişimlerini göz önüne alacak şekilde en iyi değişimi belirler. Proses yerleşim maliyetinde daha fazla azalma olmayacağı anlaşıncaya kadar devam eder. Bu mantıkla elde edilen son yerleşim hiçbir iki-yol (üç-yol) değişimi, maliyeti düşüremeyeceğinden iki-opt (üç-opt) yerleşim olarak da bilinir.

Bilgisayarların 1960'larda görece olarak yavaş olması nedeniyle, CRAFT' ın orijinal kuruluşu yukarıdaki açıklamadan az sapmıştır. Program i ve j bölümlerinin yerleşimlerini değiştirdiğinde, bölüm yerleşimlerinin güncel değişimini, yeni merkezleri ve güncel yerleşim maliyetlerini hesaplamak yerine mevcut yerleşimde i bölümünün

merkezine geçici olarak  $j$  bölümünün merkezi olarak davranarak tahmini yerleşim maliyetini hesaplamada kolaylık sağlamaktadır; bu  $i$  ve  $j$  merkezlerini değiştirir.

Yerleşim maliyetinde yukarıda açıklanan nedenden dolayı oluşan hata değiştirilecek iki bölümün boyutlarına bağlıdır. Eğer bölümler boyut olarak farklıysa, tahmini merkezler gerçek yerlerinden sapar. Tabii ki eğer bölümler eşit boyuttaysa hata oluşmaz. Sonuç olarak, yerleşim maliyetindeki güncel azalma tahminin üstünde veya altında olabilir. Bir defa tahmini yerleşim maliyetlerine göre en iyi değişim elde edildiğinde, CRAFT yeni iterasyona devam etmeden bölümlerin yerlerini değiştirir ve yeni merkezleri (ve güncel yerleşim maliyetlerini) hesaplar.

Yukarıdaki CRAFT tanımında yapılması gereken önemli bir açıklama değişimin prosedürüdür. CRAFT iki bölümün değişimini ele aldığı zaman, yukarıda açıklandığı gibi bütün olası değişimleri göz önüne almak yerine, komşu veya eşit alanlı bölümlerin değişimini ele alır. Bu kısıt rasgele değildir. Bu bölümler ayrılamaz, yerleşimde diğer bölümlerin yerleri değiştirilmeden (eğer bu bölümler eşit alanlı veya komşu değilse) iki bölümün değişimi imkansızdır. CRAFT bu mantıkla bölümlerin yerini otomatik olarak değiştiremeyeceğinden, sadece eşit alanlı veya komşu bölümlerin değişimlerini ele alır.

Eşit alanlı iki bölüm komşu olsun veya olmasın yerleşiminde diğer bölümleri değiştirmeye gerek kalmadan değiştirilebilir. Eğer iki bölüm eşit alana sahip değilse, diğer bölümleri etkilemeden bu bölümleri değiştirmek için komşuluk gerekli fakat yeterli değildir. Belli durumlarda eşit olmayan alanlı iki bölüm komşu olabilir fakat diğer bölümleri etkilemeden değiştirmek mümkün olmayabilir.

Daha iyi çözüm araştırılırken, CRAFT her iterasyonda sadece an iyi değişimi alır. Yukarıdaki araştırmada aynı zamanda geriye veya ileriye bakış yoktur. Bu nedenle CRAFT araştırma sırasında karşıladığı ilk iki veya üç optimal çözümü bitirir. Böyle bir çözüm daha çok sadece yerel optimale benzer. Bu araştırma prosedürüyle bitiş noktası (veya son çözüm) güçlü bir şekilde başlangıç noktasının (başlangıç çözümü) etkisinde olacaktır. CRAFT yüksek derecede “yol-bağımlı” sezgisel metottur ve etkili kullanılması için değişik başlangıç çözümlerinin veya değişik değişim seçeneklerinin (iki-yol, üç-yol vs.) her iterasyonda denenmesi önerilir.

CRAFT bölüm şekilleri yönünden esnekler. Bölümler ayrı olmadığı sürece herhangi bir bölüm şeklini virtüel olarak ayarlayabilir. Teorik olarak, merkezden merkeze mesafe ölçümüyle, optimum yerleşim (amaç fonksiyonu değeri sıfır olan) ortak merkezli dikdörtgenleri içerir. Tabii ki, yukarıdaki problem bazı O, U, ve L şekilli bölümlerin merkezlerinin bölümün dışında yer alması durumunda önerilir. Eğer başlangıç çözümü ortak merkezli bölümleri içermiyorsa, CRAFT böyle bir yerleşimi kurmayacaktır. Bununla birlikte, bazı bölüm şekilleri düzensiz olabilir ve amaç fonksiyonu değeri merkezden merkeze ölçüme göre tahmin edilemiyebilir.

CRAFT normal olarak dikdörtgen binalar için de kullanılabilir. Fakat sahte (dummy) bölümlerle, dikdörtgen olmayan binalar için de kullanılabilir. Sahte bölümler diğer bölümlerle akışa veya ilişkiye sahip değildir fakat yerleşim planlamacı tarafından belirlenecek belli alana ihtiyaç duyar. Sahte bölümler genellikle:

1. Düzensizliklere karşı binaları doldurmak,
2. Tesisteki engelleri ve kullanılmayan alanları temsil etmede (merdivenler, asansörler vs.),
3. Tesisteki ekstra boşluğu sunmada,
4. Son yerleşimde koridor boşluklarını değerlendirmeye yardımcı olmada kullanılır.

Sahte bölümler engelleri temsil edeceği zaman, yeri sabitlenmelidir. CRAFT kullanıcılarına herhangi bir yerleşimi sabitleme olanağı verir. Bu nitelik, özellikle mevcut tesislerde kabul etme ve yükleme gibi bölümler kadar engellerde de kullanılır.

CRAFT'ın güçlü yanlarından biri başlangıç yerleşimini kabul edilebilir doğrulukla yakalamasıdır. Bu güç temel olarak CRAFT'ın dikdörtgen olmayan binalarda dikdörtgen olmayan bölümler veya engellerin herhangi bir yere yerleştirme yeteneğinden gelir. Bununla birlikte, yüksek oranda yol bağımlı olmasına ek olarak, CRAFT'ın zayıf yanlarından biri son yerleşimde istenilen koridorların çakışmaması ve düz olması özelliğini nadiren sağlamasıdır. Bazı bölümlerin özel yerlere sabitlenmesi ve bazı durumlarda sahte bölümlerin ana koridorları temsil edecek şekilde yerleştirilmesi daha kabul edilebilir bölüm şekillerine neden olur.

### 3.2.3 MIP

Bu bölümde Karışık Tam Sayılı Programlamaya geçmeden önce programın çalışmasını daha iyi görebilmek açısından önce dal ve sınır algoritmasına kısaca değinilecektir.

#### Dal ve Sınır Algortması

Herhangi bir saf tam sayılı programlama probleminin sonlu sayıda uygun çözüme sahip olması nedeniyle, optimal çözümün bulunması için bir çeşit numaralandırılmış prosedürün kullanılması doğaldır. Ne yazık ki bu sonlu sayı çok büyük olabilir. Bu nedenle herhangi bir numaralandırılmış prosedürü uygun çözümlerin küçük bir kısmının ele alınacağı şekilde yapılandırmak akıllıca olur (Hillier, F.S., Lieberman, G.J., 1995).

Dal ve sınır tekniğinin altında yatan temel kavram bölme ve ele geçirmektir. Orjinal büyük problem doğrudan çözülemeyecek kadar zor olduğu için, küçük ve daha küçük alt problemlere, bu alt problemler ele geçirilinceye kadar bölünür. Bölme (dallanma) tüm uygun çözümler grubunun küçük ve daha küçük alt gruplara ayrılmasıyla yapılır. Ele geçirme (fathoming) kısmen alt gruplardaki en iyi sonucun nasıl iyi olabileceğini sınırlama ve eğer sınırının problem için optimal çözümü vermeyeceğini gösteriyorsa alt gruplardan vaz geçme yoluyla olur.

**Dallanma:** Eğer ikili (binary) değişkenlerle çalışılıyorsa uygun çözümler kümesini alt kümelere ayırmanın en doğru yolu bir değişkenin değerini bir alt küme için 0, diğer alt küme için 1 almaktır. Bu ağaç iterasyondan iterasyona dallara ayrılır ve algoritma için çözüm ağacı olarak adlandırılır. Dallanmanın yapıldığı değişkenlere değerlerin atandığı herhangi bir iterasyonda, değişkene dallanma değişkeni denir.

**Sınırlama:** Bu alt problemlerin herbiri için en iyi uygun çözümün nasıl olacağını belirlemek için bir sınır koymamız gerekir. Bunu yapmanın standart yolu alt problemi rahatlatmaktır. Bir çok durumda problemin rahatlatılması problemi yaratan kısıtlar değişkenlerin tam sayı olması gereken kısıtlardır. Bu nedenle en geniş kullanılan rahatlatma, bu kısıt gruplarını silen doğrusal programlama rahatlatmasıdır.

**Fathoming:** Bir alt problem ele geçirilebilir ve bu nedenle daha sonraki değerlendirmelerden çıkarılabilir. Bunun üç yolu vardır.

1. Sınır  $\leq z^*$  olursa

2. Doğrusal programlama rahatlatması uygun çözüme sahip değilse

3. Doğrusal programlama rahatlatması tam sayıysa fathom yapılır.

Dal ve Sınır Algoritmasının Özeti

Başlangıç:  $z^* = -\infty$  al. Sınırlama, fathoming ve optimalite testlerini aşağıdaki şekilde probleme uygula.

İterasyonlar için Adımlar

1. Dallanma: Mevcut alt problemlerden, en sık karşılaşılana seç. Bu alt problemlerden bir sonraki değişkeni 0 veya 1 olacak şekilde iki yeni alt problem oluştur.

2. Sınırlama: Her yeni alt problem için doğrusal programlama rahatlatmasına simpleks metodla sınır oluştur ve optimal çözümün sonucuna göre  $z$  değerini aşağı yuvarla.

3. Fathoming: Bütün yeni alt problemler için üç fathoming testini uygula ve herhangi bir testte fathom yapılan alt problemleri çıkar.

Optimalite testi: Alt problem kurulamayınca dur. Bulunan sonuç uygundur. Diğer durumda başka bir iterasyon yapmak için geri dön.

MIP Dal ve Sınır Algoritması

Başlangıç:  $z^* = -\infty$  al. Sınırlama, fathoming adımları ve optimalite testlerini aşağıdaki şekilde probleme uygula. Eğer fathom olmazsa bu problemi aşağıdaki ilk tam iterasyonu uygulamak üzere herhangi bir alt problem olarak sınıflandır.

Adımlar

1. Dallanma: Mevcut alt problemlerden, en sık ortaya çıkanı seç (bağları en büyük sınıra sahip olana göre ayır). Alt problemin LP rahatlatmaları için optimal çözümde tam sayı olmayan değerlere sahip tam sayı sınırlı değerler arasından dallanma değişkeni olarak doğal sıralı ilk değişkeni seç.  $x_j$  değişkeni  $x^*$  çözümdeki değeri olsun. Alt problemin yeni alt problemler oluşturması için  $x_j \leq [x_j^*]$  ve  $x_j \geq [x_j^*] + 1$  kısıtlayıcılarının eklenmesiyle oluşur.

2. Yeni alt problemler için simpleks metodu ve optimal çözümün sonucu olan  $z$  değerini kullanarak LP rahatlatmasına sınırları elde et.

3. Fathmonig: Yeni alt problemler için, aşağıda verilen üç fathoming testini uygula ve herhangi bir testte fathom olan alt problemi ayır.

Test 1: Sınır  $\leq z^*$ ,  $z^*$  in  $z'$  ye düşen değeri olduğu durumda.

Test 2: LP rahatlatmanın uygun çözümü yoksa

Test 3: LP rahatlatması için optimal çözüm tamsayı sınırlı değişkenlerde tam sayı değerleri alırsa (eğer bu değerden daha iyiyse yeni değeri olur ve 1. Testten yeniden başlanır).

Optimalite testi: Alt problem kalmadığı zaman dur: mevcut değeri optimaldir. Diğer durumda bir başka iterasyon kullan.

#### MIP

Dal ve sınır algoritmasına kısaca baktıktan sonra şimdi de yerleşim için bir karışık tam sayılı programlama modelini inceleyelim. Sürekli sunum kullanılarak tesis yerleşim problemi eğer bütün bölümler dikdörtgen kabul ediliyorsa Karışık Tamsayılı Programlama (MIP) problemi olarak formüle edilebilir. Dikdörtgen bölümlerle, sadece merkez ve uzunlukla (ya da kalınlıkla) bölümün yeri ve şekli tanımlanabilir. Alternatif bir amaç kullanılabilmesine rağmen, burada sunulan model  $d_{ij}$  i ve j bölümlerinin merkezleri arasındaki düzgün doğrusal mesafe olmak üzere, uzaklık tabanlı bir modeldir. Genel görüş, matematiksel programlamaya dayalı modellerin başlangıç çözümüne ihtiyaç duymaması nedeniyle kuruluş tipi yerleşim modelleri olduğudur. Bununla beraber, daha sonra görüleceği gibi, bu tip modeller aynı zamanda yerleşimi geliştirmede de kullanılabilir.

Burada gösterilen model Montreuil' in (Montreuil, 1991) sunduğu modele dayanmaktadır. Bölüm boyutlarına karar değişkenleri yerine bilinen değerlere sahip parametreler olarak davranılan benzer bir model Heragu ve Kusiak (1991) tarafından sunulmuştur. Bütün bölümlerin boyutlarının verilmiş ve sabit olması her bölümün dikdörtgen makine ayak izine göre sunulduğu makine yerleşim problemine uygundur. Böyle bir yaklaşımda yerleşim modeli güncel olarak bilinen şekillere sahip dikdörtgen şekillerin yerleşimine karar vermede iki boyutlu paketleme algoritması olarak kullanılabilir. Amaç fonksiyonundaki olası farklılıklardan ayrı olarak, bu paketleme problemi iki boyutlu ambar paketleme problemi olarak da bilinir.

Bölüm boyutlarına karar değişkenleri olarak davranarak tesis tasarımı problemi aşağıdaki gibi formüle edilebilir.

İlk olarak problem parametrelerini ele alalım:

$B_x$  : Bina uzunluğu (x koordinatı boyunca)

$B_y$  : Bina genişliği (y koordinatı boyunca)

$A_i$  : i bölümünün alanı

$L_i^l$  : i bölümünün uzunluktaki alt limiti

$L_i^u$  : i bölümünün uzunluktaki üst limiti

$W_i^l$  : i bölümünün genişlikteki alt limiti

$W_i^u$  : i bölümünün genişlikteki üst limiti

$M$  : büyük bir sayı

Daha sonra karar değişkenlerini ele alalım:

$\alpha_i$  : i bölümünün merkezinin x koordinatı

$\beta_i$  : i bölümünün merkezinin y koordinatı

$x'_i$  : i bölümünün sol (veya batı) tarafını x koordinatı

$x''_i$  : i bölümünün sağ (veya doğu) tarafının x koordinatı

$y'_i$  : i bölümün alt kısmının (veya güney) y koordinatı

$y''_i$  : i bölümünün üst (veya kuzey) kısmının y koordinatı

$z_{ij}$  : eğer i bölümü tam olarak j bölümünün doğusunda 1, diğer durumda 0

$z'_{ij}$  : eğer i bölümü tam olarak j bölümünün batısında 1, diğer durumda 0.

i bölümü sadece  $x''_j \leq x'_i$  olursa j bölümünün tam olarak doğusunda kalır.

Benzer şekilde i bölümünün j bölümünün batısında olması için sadece ve sadece  $y''_j \leq y'_i$  olmalıdır. İki bölümün x veya y koordinatında dağılması durumunda çakışmaları garanti edilmiştir. (x koordinatı için bir bölüm diğerinin doğusunda, y koordinatı içinse kuzeyinde olacaktır). Tabii ki iki bölüm hem x hem de y koordinatı boyunca olabilir.

Yukarıdaki parametre ve değişken tanımları aşağıdaki modeli oluşturur:

$$\text{minimize } z = \sum_i \sum_j f_{ij} * c_{ij} * (|a_i - a_j| + |b_i + b_j|) \quad (3.1.)$$

$$\text{s.t. } L_i^u \leq (x''_i - x'_i) \leq L_i^u \quad \text{bütün } i' \text{ ler için} \quad (3.2.)$$

$$W_i^u \leq (y''_i - y'_i) \leq W_i^u \quad \text{bütün } i' \text{ ler için} \quad (3.3.)$$

$$(x''_i - x'_i)(y''_i - y'_i) = A_i \quad \text{bütün } i' \text{ ler için} \quad (3.4.)$$

$$0 \leq x'_i \leq x''_i \leq B_x \quad \text{bütün } i' \text{ ler için} \quad (3.5.)$$

$$0 \leq y'_i \leq y''_i \leq B_y \quad \text{bütün } i' \text{ ler için} \quad (3.6.)$$

$$\alpha_i = 0.5x'_i + 0.5x''_i \quad \text{bütün } i' \text{ ler için} \quad (3.7.)$$

$$\beta_i = 0.5y'_i + y''_i \quad \text{bütün } i' \text{ ler için} \quad (3.8.)$$

$$x''_j \leq x'_i + M(1 - z_{ij}^x) \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.9.)$$

$$y''_i \leq y'_j + M(1 - z_{ji}^y) \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.10.)$$

$$z_{ij}^x + z_{ji}^x + z_{ij}^y + z_{ji}^y \geq 1 \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i < j \quad (3.11.)$$

$$\alpha_i, \beta_i \geq 0 \quad \text{bütün } i' \text{ ler için} \quad (3.12.)$$

$$x'_i, x''_i, y'_i, y''_i \geq 0 \quad \text{bütün } i' \text{ ler için} \quad (3.13.)$$

$$z_{ij}^x, z_{ij}^y \text{ 0/1 tamsayı} \quad \text{bütün } i \text{ ve } j' \text{ ler için} \quad (3.14.)$$

(3.1.)' de verilen amaç fonksiyonu daha önce (1.1)' de gösterilen uzaklık tabanlı amaçtır. (3.2.) ve (3.3.) kısıtları her bölümün uzunluk ve genişliğinin belirlenen sınırlar arasında olmasını sağlar. Her bölüm için gerekli alan modelin tek doğrusal olmayan kısıtı (3.4.)' le ifade edilir. (3.5.) ve (3.6.) kısıtları bölüm kenarlarını tanımlar ve her bölümün binanın x ve y koordinatlarında kalacak şekilde yerleştirilmesini sağlar. (3.7.) ve (3.8.) kısıtları her bölümün merkezlerinin x ve y koordinatlarını tanımlar. (3.9.) kısıtı eğer  $z_{ij}^x = 1$  ise  $x''_j \leq x'_i$  olmasını sağlar (i bölümü j bölümünün doğusunda). Eğer  $z_{ij}^x = 0$  olursa, (3.10) kısıtı  $x''_j$  küçük veya eşit  $x'_i$  olursa sağlanır. Diğer bir deyişle, (3.10.) kısıtı sadece  $z_{ij}^x = 1$  olursa aktif olur. (3.11.) kısıtı (3.10.) kısıtıyla aynı amaca hizmet eder (y yönünde). (3.12) kısıtı hiçbir bölümün doğu-batı veya kuzey-güney yönünde ayrılma durumunda çalışmayacağını belirtir. Son olarak, (3.13.) ve (3.14.) negatif olmama koşulunu sağlar.

(3.4.) kısıtına göre yukarıdaki model doğrusal olmayacak şekilde kısıtlanmıştır. Bundan başka, amaç fonksiyonu tam değer işlemcisini içerir. Doğrusal fonksiyon yoluyla (3.4.) kısıtını uygun hale getirmek için alternatif planlar uygulanmalıdır. Montreuil' in çalışmasında bölüm alanı çevresi boyunca kontrol edilir (bölümün uzunluk ve genişliğinin doğrusal fonksiyonu olan) ve kısıt (3.3.) ve (3.2.) oluşturulur.

Bütün  $i$  ve  $j$ ' ler için  $f_{ij} \geq 0$  oluşturulmasıyla, amaç fonksiyonundaki değerler her terim için pozitif ve negatif bölümler olarak değiştirilir. Eğer

$$|a_i - a_j| = a_{ij}^+ - a_{ij}^-, \text{ ve } |\beta_i - \beta_j| = \beta_{ij}^+ - \beta_{ij}^- \quad (3.15.)$$

olursa

$$|\alpha_i - \alpha_j| = \alpha_{ij}^+ + \alpha_{ij}^- \text{ ve } |\beta_i^+ - \beta_j^-| = \beta_{ij}^+ + \beta_{ij}^- \quad (3.16.)$$

burada  $\alpha_{ij}^+, \alpha_{ij}^-, \beta_{ij}^+$  ve  $\beta_{ij}^-$  bütün negatif olmayan değişkenlerdir.

(3.2.)' den (3.14) denkleminde kadar verilmiş doğrusal olmayan model aşağıdaki şekilde doğrusal MIP problemine dönüştürülür:

$$\text{minimize } z = \sum_i \sum_j f_{ij} c_{ij} (\alpha_{ij}^+ + \alpha_{ij}^- + \beta_{ij}^+ + \beta_{ij}^-) \quad (3.17.)$$

$$\text{s.t. } L_i^l \leq (x''_i - x'_i) \leq L_i^u \quad \text{bütün } i' \text{ ler için} \quad (3.18.)$$

$$W_i^u \leq (y''_i - y'_i) W_i^u \quad \text{bütün } i' \text{ ler için} \quad (3.19.)$$

$$P_i^l \leq 2(x''_i - x'_i + y''_i - y'_i) \leq P_i^u \quad \text{bütün } i' \text{ ler için} \quad (3.20.)$$

$$0 \leq x'_i \leq x''_i \leq B_x \quad \text{bütün } i' \text{ ler için} \quad (3.21.)$$

$$0 \leq y'_i \leq y''_i \leq B_y \quad \text{bütün } i' \text{ ler için} \quad (3.22.)$$

$$\alpha_i = 0.5 x'_i + 0.5 x''_i \quad \text{bütün } i' \text{ ler için} \quad (3.23.)$$

$$\beta_i = 0.5 y'_i + 0.5 y''_i \quad \text{bütün } i' \text{ ler için} \quad (3.24.)$$

$$\alpha_i + \alpha_j = \alpha_{ij}^+ - \alpha_{ij}^- \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.25.)$$

$$\beta_i - \beta_j = \beta_{ij}^+ - \beta_{ij}^- \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.26.)$$

$$x''_j \leq x'_i + M(1 - z_{ij}^x) \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.27.)$$

$$y''_j \leq y'_i + M(1 - z_{ij}^y) \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.28.)$$

$$z_{ij}^x + z_{ji}^x + z_{ij}^y + z_{ji}^y \geq 1 \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.29.)$$

$$\beta_i, \alpha_i \geq 0 \quad \text{bütün } i' \text{ ler için} \quad (3.30.)$$

$$x'_i, x''_i, y'_i, y''_i \geq 0 \quad \text{bütün } i' \text{ ler için} \quad (3.31.)$$

$$\alpha_{ij}^+, \alpha_{ij}^-, \beta_{ij}^+, \beta_{ij}^- \geq 0 \quad \text{bütün } i' \text{ ler için} \quad (3.32.)$$

$$z_{ij}^x, z_{ij}^y \text{ 0/1 tamsayı} \quad \text{bütün } i \text{ ve } j' \text{ ler için} \quad (3.33.)$$

$P_i^l$  ve  $P_i^u$  (3.20) kısıtında  $i$  bölümünün çevresinin üst ve alt limitlerini belirtmektedir.

Yukarıdaki formülasyonda, bölüm şekilleri alanlar kadar (3.18.)-(3.20.) arasındaki kısıtlarla da kontrol edilmektedir. Bazı durumlarda bütün şekillerinde uzun/kısa kenar oranında üst bir sınır konularak tam kontrol istenebilir. Eğer üst limiti  $R_i (\geq 1.0)$  olarak belirtirsek, böyle bir oran aşağıdaki kısıt eklenerek kolayca kontrol edilebilir:

$$(x''_i - x'_i) \leq R_i (y''_i - y'_i) \quad (3.34.)$$

$$(y''_i - y'_i) \leq R_i (x''_i - x'_i) \quad (3.35.)$$

Hangi kenarın daha uzun olduğuna bağlı olmak üzere, kare olmayan bölümler için yukarıdakiler en fazla tespit etme veya eşitlik olarak tutulabilir.

$X$  ilişkisinin  $i$  ve  $j$  bölümleri arasındaki  $f_{ij}$  değerini negatif seçerek verilebileceği daha önce belirtilmişti. Bu nedenle modeli doğrusallaştırmak için kullanılan dönüşüm  $f_{ij} \geq 0$  (bütün  $i$  ve  $j'$  ler için) ihtiyaç duyar. Bu ihtiyaç negatif akış kullanımına olanak vermeyeceği için yukarıdaki modelde  $X$  ilişkisi etkili bir şekilde gösterilemiyebilir. Kısıt (3.27.) ve (3.28.) değiştirilerek iki bölüm arasında ihtiyaç duyulan minimum mesafe kullanılabilir:

$$x''_j + \Delta_{ij}^x \leq x'_i + M(1 - z_{ij}^x) \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.36.)$$

$$y''_i + \Delta^y_{ij} \leq y'_i + M(1 - z^x_{ij}) \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.37.)$$

burada  $\Delta^x_{ij}$  ve  $\Delta^y_{ij}$  i ve j bölümlerinin x ve y koordinatı boyunca minimum gerekli mesafedir. Bu mesafeler ihtiyatlı kullanılmalıdır, aksi takdirde uygun çözüm sayısı çok sınırlı olabilir. Aynı zamanda, eğer i ve j bölümleri uzunluk boyunca sıralanırsa (sadece x koordinatı diyelim) aralarındaki sonuç mesafesi y koordinatı boyunca  $\Delta^y_{ij}$ ' den küçük olmalıdır.

i ve j bölümleri arasında X ilişkisi için alternatif bir yaklaşım için aşağıdaki kısıt eklenebilir:

$$\alpha^+_{ij} + \alpha^-_{ij} + \beta^+_{ij} + \beta^-_{ij} \geq \Delta_{ij} \quad (3.38.)$$

burada sol el tarafı bölümlerin merkezleri arası düzgün doğrusal mesafe ve  $\Delta_{ij}$  alt sınırdır.

Teoride, bölümleri alan ihtiyaçlarının kesin olarak sağlanması gerekmez, dikdörtgen bölümlerle optimum yerleşim (3.17.)' den (3.33)' e kadar verilen MIP modelinin çözümüyle elde edilir. Pratikte, yukarıdaki problemin tam sonucu ikili (0/1) değişken sayısının çok olması nedeniyle açık olarak bulunamaz.

Mevcut durumda genel optimizasyon kodlarıyla makul hesap zamanında çözülebilecek en büyük problem 7 veya 8 bölümdür.

Netice olarak, pratik MIP' nin kullanımı ( bölüm sayısı kolaylıkla 15 veya 18' in üstünde olabilir), optimal çözüm yerine sezgisel çözüm yoluyladır. Bir açık sezgisel yaklaşım en az alt sınır ve mevcut çözüm arasındaki fark %5 veya %10' dan az olduğu zaman dal ve sınır araştırmasını sınırlamaktır. Böyle bir yaklaşımın avantajı sonuç çözümünün optimum çözümünden %5 veya %10 yakın olmasını garantilemesidir; en önemli dezavantajı ise böyle bir çözümü elde etmenin halen çok fazla hesap zamanı çok fazla hesap zamanı almasıdır.

Bir başka sezgisel yaklaşım (36) bir tasarım iskeletini bölümler arası kuzey-güney ve batı-doğu ilişkileri için sezgisel olarak belirlenir ve (7.21)' den (7.37)' ye kadar olan denklemlerle gösterilen MIP modeli bina içindeki bölümleri paketlemek için doğrusal programlama problemi olarak çözülebilir.

Bu modelde aşağıdaki değişiklikler yapılmıştır:

$$\text{minimize } z = \sum_i \sum_j f_{ij} c_{ij} (\alpha_{ij}^+ + \alpha_{ij}^- + \beta_{ij}^+ + \beta_{ij}^-) \quad (3.17.)$$

$$\text{s.t. } L_i^1 \leq (x''_i - x'_i) \leq L_i^u \quad \text{bütün } i' \text{ ler için} \quad (3.18.)$$

$$W_i^u \leq (y''_i - y'_i) W_i^u \quad \text{bütün } i' \text{ ler için} \quad (3.19.)$$

$$P_i^1 \leq 2(x''_i - x'_i + y''_i - y'_i) \leq P_i^u \quad \text{bütün } i' \text{ ler için} \quad (3.20.)$$

$$0 \leq x'_i \leq x''_i \leq B_x \quad \text{bütün } i' \text{ ler için} \quad (3.21.)$$

$$0 \leq y'_i \leq y''_i \leq B_y \quad \text{bütün } i' \text{ ler için} \quad (3.22.)$$

$$\alpha_i = 0.5x'_i + 0.5x''_i \quad \text{bütün } i' \text{ ler için} \quad (3.23.)$$

$$\beta_i = 0.5y'_i + 0.5y''_i \quad \text{bütün } i' \text{ ler için} \quad (3.24.)$$

$$\alpha_i + \alpha_j = \alpha_{ij}^+ - \alpha_{ij}^- \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.25.)$$

$$\beta_i - \beta_j = \beta_{ij}^+ - \beta_{ij}^- \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.26.)$$

$$x''_j \leq x'_i + M(1 - z_{ij}^x) \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.27.)$$

$$y''_j \leq y'_i + M(1 - z_{ij}^y) \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i \neq j \quad (3.28.)$$

$$z_{ij}^x + z_{ji}^x + z_{ij}^y + z_{ji}^y \geq 1 \quad \text{bütün } i \text{ ve } j' \text{ ler için, } i > j \quad (3.29.)$$

$$z_{ij}^x + z_{ji}^x \leq 1 \quad \text{bütün } i \text{ ve } j' \text{ ler için} \quad (3.39)$$

$$z_{ij}^y + z_{ji}^y \leq 1 \quad \text{bütün } i \text{ ve } j' \text{ ler için} \quad (3.40)$$

$$z_{ij}^x, z_{ij}^y \text{ 0/1 tamsayı} \quad \text{bütün } i \text{ ve } j' \text{ ler için} \quad (3.33.)$$

(3.39)) kısıtlayıcısı eklenerek  $z_{ijx}$  ve  $z_{jix}$  değerlerinden sadece birinin 1 değerini alabileceği gösterilmiştir. Aynı şekilde (3.40) kısıtlayıcısı  $z_{ijy}$  ve  $z_{jiy}$  ikili değişkenlerinden sadece birinin 1 olabileceğini göstermektedir. (3.30)-(3.32) kısıtları ise Lindopro' nun değişkenleri sıfırdan alması özelliği nedeniyle çıkartılmıştır.

#### 4. BULGULAR VE TARTIŞMA

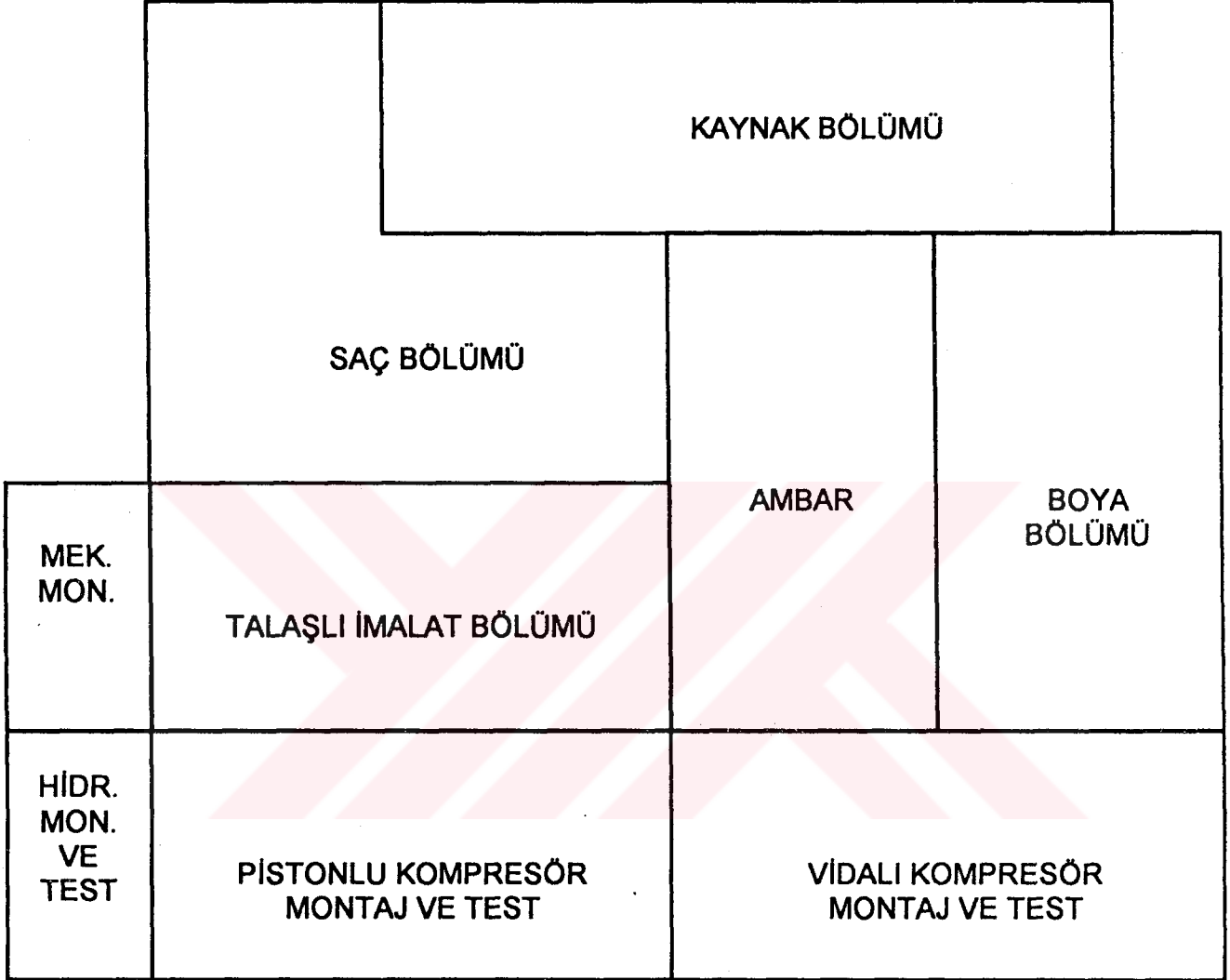
##### 4.1.1 CRAFT Uygulaması

İşletme için fabrika yerleşim araştırması yapılırken iki yöntemden yararlanılmıştır. Daha önceki bölümlerde ayrıntılı olarak bilgi verilen bu yöntemler Armour ve Buffa tarafından geliştirilmiş olan CRAFT ve Karışık Tam Sayılı Programlama (MIP) yaklaşımlarıdır.

CRAFT uygulaması sırasında çeşitli alternatifler göz önüne alınmıştır. Bunlar bazı bölümlerin sabit tutulması ve çözüm yöntemlerinin değiştirilmesidir. Bütün yaklaşımlar için birim maliyet 1 olarak kabul edilmiştir. Yerleşimdeki her bir hücre 25x25 oranındadır. Başlangıç yerleşimi programa girilmiştir. Fabrikanın başlangıç yerleşimi Şekil 4.1.' de görülmektedir.

İlk yaklaşım bütün bölümlerin yerlerinin değiştirilebileceği yönünde olmuştur. Giriş bilgilerinde hiçbir bölüm sabit olarak gösterilmemiştir. Bundan sonraki aşamada ilk olarak ikili değişim metodu denenmiştir. CRAFT ikili değişim metoduyla 9 iterasyon sonucunda ve 96.723 CPU zamanında maliyeti 3476.38 birim olarak bulmuştur. Sonuç yerleşiminde bütün bölümler yer değiştirmiştir. Değişimler Şekilde görülmektedir. İkinci yaklaşım olarak üçlü değişim yöntemi ele alınmıştır. Burada CRAFT bölümleri üçlü gruplar halinde değiştirmektedir. Burada üç iterasyon sonucunda ve 39.102 CPU zamanında elde edilen maliyet 3844.821 birimdir. Bu sonuç ikili değişim metodundaki sonuçtan yaklaşık 368 birim daha kötü bir sonuçtur, fakat hesap zamanı daha düşüktür. Bu yaklaşımda birinci ve ikinci bölümler sabit kalmış, diğer bölümlerin yerleri ise değiştirilmiştir. Üçüncü yaklaşım olarak önce ikili daha sonra üçlü değişim yöntemi seçilmiştir. Bu uygulamada elde edilen sonuçlar ise 9 iterasyon sonucunda ve 19.379 CPU zamanında maliyet 3476.38 birim bulunmuştur. Dikkat edilecek olursa maliyet ikili değişim metodundaki maliyete eşittir. Bunun nedeni CRAFT' ın ikili değişim metoduyla önce ikili sonra üçlü değişim metodu yaklaşımında aynı yerleşimi elde etmiş olmasıdır. Fakat görülebileceği gibi hesap zamanı 96.723 CPU zamanından 19.379 CPU zamanına düşmüştür. Son yaklaşımımız ise önce üçlü sonra ikili değişim yaklaşımıdır. Altı iterasyon sonucunda ve 16.7968 CPU zamanında

maliyet 3485.175 birim olarak elde edilmiştir. Bu sonuç önce ikili daha sonra üçlü değişim metoduna



Şekil 4.1. İşletmenin Başlangıç Yerleşimi

göre daha iyi fakat diğer iki yöntemle göre daha kötüdür. CPU zamanı olarak en iyi sonuç bu alternatifte ortaya çıkmıştır. Birinci araştırma sonucunda elde edilen sonuçlara bakılacak olursa en iyi sonuç ikili değişim ve önce ikili sonra üçlü değişim metodları ile elde edilmiştir. Fakat burada dikkat edilmesi gereken önemli bir noktada bölüm şekilleridir. CRAFT uygulamayı yaparken bölüm şekillerine değil taşıma maliyetlerinin minimum olmasına bakmaktadır. En düşük maliyetin elde edildiği ikili değişim ve önce ikili sonra üçlü değişim metodlarına bakılacak olursa bazı bölüm şekillerinin uygulanmasının çok güç olduğu görülebilir. Burda birinci bölüm diğer bölümlerin arasında koridor şeklinde yerleştirmiştir ki bunun uygulanması çok zordur. Sekizinci bölümde üçüncü ve birinci bölümün arasında koridor şeklinde yerleştirilmiştir. Bu şekilde bölümlerin yerleştirilmesi oldukça zordur. Bu nedenle yerleşim tasarımcısının CRAFT' la elde edilen şekiller üzerinde düzeltmeler yapması kaçınılmazdır.

İkinci araştırma ilk taşınılacağı düşünülen birinci bölümü sabit tutarak diğer bölümlerin değiştirilmesi yaklaşımıdır. Burada giriş değerleri değiştirilmemiş fakat birinci bölüm sabit alınmıştır. Diğer araştırmadaki gibi dört yöntemde bu araştırma sırasında da kullanılmıştır. Birinci yöntemimiz olan ikili değişim yöntemi sonucunda 8 iterasyon ve 11.980 CPU zamanında maliyet 3635.997 birim olarak bulunmuştur. Üçlü değişim yönteminde üç iterasyon ve 8.5625 CPU zamanında maliyet 3850.764 birim olarak bulunmuştur. Görüldüğü gibi hesap zamanı ve iterasyon sayısı oldukça düşmüş, fakat maliyet bir miktar artmıştır. Üçüncü metotta maliyet sekiz iterasyon ve 13.457 CPU zamanında yine bir önceki değer olan 3635.997 birim olarak elde edilmiştir. Yerleşim şekil bir önceki metottaki ile aynıdır. Son olarak önce üçlü sonra ikili değişim metodunda yedi iterasyon ve 15.64844 CPU zamanında maliyet 3502.765 birim olarak bulunmuştur. En düşük maliyet son yaklaşımda elde edilmiştir. Burada CPU zamanı olarak makul bir zamanda diğer yaklaşımlara göre nispeten daha iyi bölüm şekilleri elde edilmiştir.

Son olarak ilk iki taşınacak bölümün sabit kalması durumunda taşıma maliyeti araştırılmıştır. Burada da girişi değerleri diğer araştırmalardaki gibi aynı alınmış fa kat

ilk iki bölümün sabit kaldığı varsayımı kullanılmıştır. İkili değişim metoduyla yedi iterasyon ve 9.0625 CPU zamanına maliyet 3707.012 birim olarak bulunmuştur. İkinci metod olan üçlü değişim metoduyla altı iterasyon ve 14.281 CPU zamanında maliyet 3592.207 birim olarak bulunmuştur. Önce ikili sonra üçlü değişim metoduyla yedi iterasyon ve 10.66016 CPU zamanında maliyet 3707.012 birim olarak elde edilmiştir. Son metod olan önce üçlü sonra ikili değişim metoduyla altı iterasyon ve 15 .10938 CPU zamanında maliyet 3592.207 birim olarak elde edilmiştir. İlk iki bölümün sabit tutulması durumunda elde edilen sonuçlardan en iyisi üçlü değişim ve önce üçlü sonra ikili değişim metodlarıyla elde edilmiştir. İterasyon sayısı her iki metotta da altıdır, fakat CPU zamanı üçlü değişimi metodunda diğerine oranla nispeten daha düşüktür.

#### 4.1.2 Karışık Tam Sayılı Programlama

Burada bir tam sayılı programlama modeli tesis yerleşimi problemine uygulanmıştır. Model MIP bölümünde ayrıntılı olarak açıklandığı için ayrıntılarına girilmeyecektir. Uygulama Lindopro' da yapılmıştır. Bu yöntemin sağladığı avantaj amaç fonksiyonunun optimum çözümünü bulmasıdır ki burada amaç fonksiyonumuz CRAFT uygulamasında olduğu gibi taşıma maliyetlerini minimize etmektir. Amaç fonksiyonundaki değerlerden malzeme akış yoğunluğu -den -e çizelgesinden elde edilmiştir. Maliyetler CRAFT' ta olduğu gibi 1 olarak alınmıştır. Kısıtlayıcılardaki bina uzunluğu ve genişliği değerleri bilinmektedir. Burada bina uzunluğu (binanın x koordinatı) 90 metre, bina genişliği( binanın y koordinatı) ise 70 metredir. Bölüm uzunluk ve genişliklerinin alt ve üstsınırları bölümlerin mevcut uzunluk ve genişliklerine göre belirlenmiştir. Problem tanımlandıktan sonra Lindopro'ya uygun şekilde düzenlenerek bu programda optimum değer araştırılmıştır.

## 5. SONUÇLAR VE ÖNERİLER

CRAFT sezgisel yaklaşımı kullanan bir algoritmadır. Bu nedenle optimum çözümü bulmaz. Fakat optimuma yakın veya bazı durumlarda optimumdan daha iyi sonuçlar elde edilebilir. Burada yapılan araştırma sırasında üç değişik alternatif için CRAFT' ta dört metod kullanılmıştır. Bu metodlar sonucunda

a) En iyi sonuç maliyetin 3476.38 birim olarak elde edildiği hiç bir bölümün sabit olmaması durumundaki ikili değişim ve önce ikili sonra üçlü değişim metodları ile elde edilmiştir.

b) CPU zamanı olarak ikinci metod ikili değişime göre daha az zaman almıştır.

c) Daha öncede belirtildiği gibi elde edilen bölüm şekilleri, özellikle birinci ve sekizinci bölümlerin şekillerinin düzeltilmesi ve uygulanabilir şekillere dönüştürülmesi gerekmektedir.

d) Birinci bölümün sabit kalması halinde önce ikili sonra üçlü değişim metodu ile elde edilen sonuç alternatif bir sonuç olarak ele alınabilir. Buradaki bölüm şekilleri diğer sonuçlara göre daha uygulanabilir niteliktedir (Şekil 5.1.).

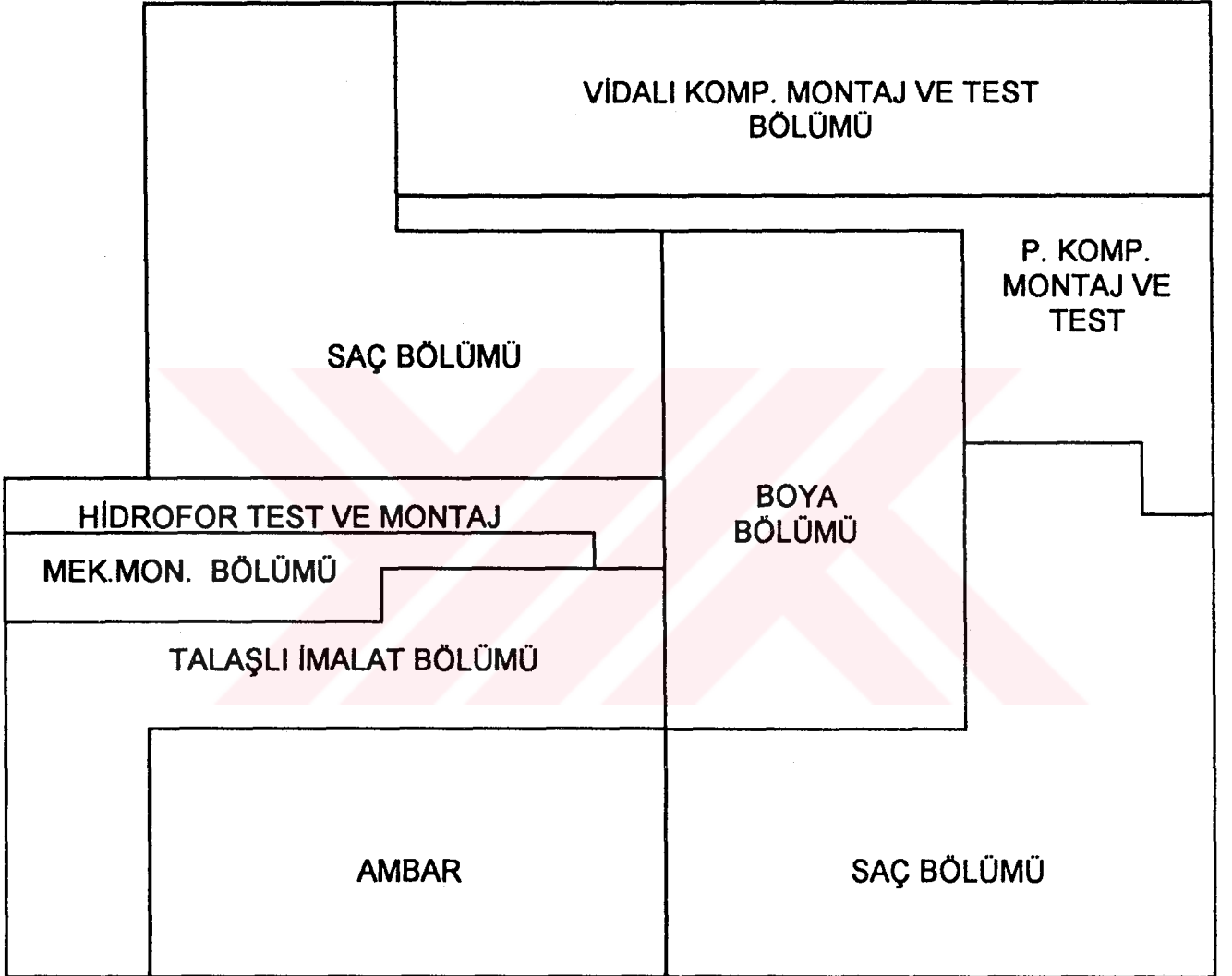
f) En kötü sonuç ise birinci bölümün sabit kalması halinde üçlü değişim yöntemi ile elde edilmiştir.

g) Genel olarak iki bölümün sabit olduğu durumda maliyetler diğer alternatiflere göre daha yüksek çıkmıştır. Fakat bu alternatifte üçlü değişim ve önce üçlü sonra ikili değişim metodları ile elde edilen çözüm bölüm şekillerinin diğer alternatiflere göre daha makul ve uygulanabilir olması nedeniyle önerilmektedir. Maliyetinin biraz daha fazla olmasına rağmen uygulanabilir bir şekil elde edilmiştir (Şekil 5.2.).

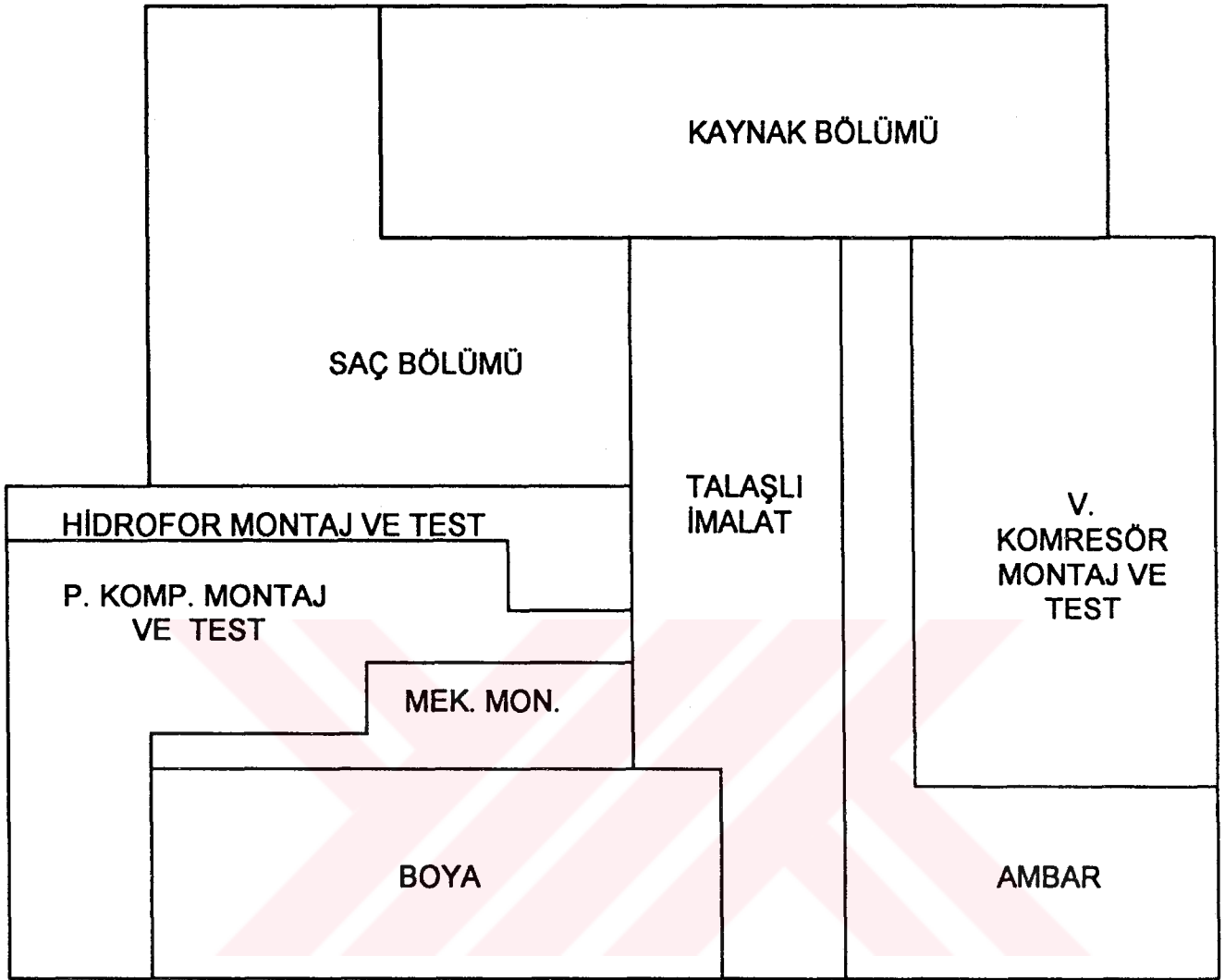
h) İşletme için optimum çözüm arayışı MIP uygulaması ile yapılmıştır. Fakat problemin boyutunun büyük olması nedeniyle bu uygulamadan sonuç alınamamıştır. Bölüm sayısının fazla olması ve buna bağlı olarak değişken sayısının fazla olması nedeniyle Lindopro' da yapılan uygulama bir haftalık süre içinde bir sonuca ulaşamamıştır.

i) Elde edilen sonuçlar maliyet ve CPU zamanlarının kıyaslanması açısından Tablo 5.1 ve 5.2' de gösterilmiştir. Burada ilk dört alternatif birinci bölümün sabit

olması durumunda ikili, üçlü, önce ikili sonra üçlü ve önce üçlü sonra ikili değişim metodları ile elde edilen yerleşimlerdir. İkinci dörtlü ise bölümlerin sabit olmaması durumunda elde edilen maliyet sonuçlarını göstermektedir. Son olarak tablodaki son dörtlü ise ilk iki bölümün sabit olması durumunda maliyetlerdir.

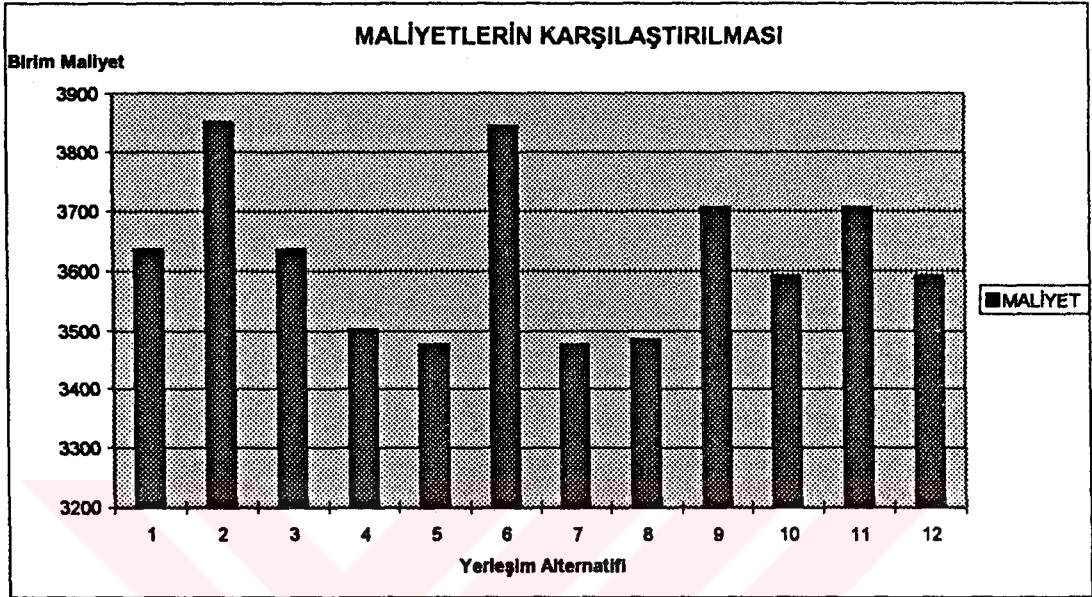


Şekil 5.1. İşletme İçin Alternatif Yerleşim



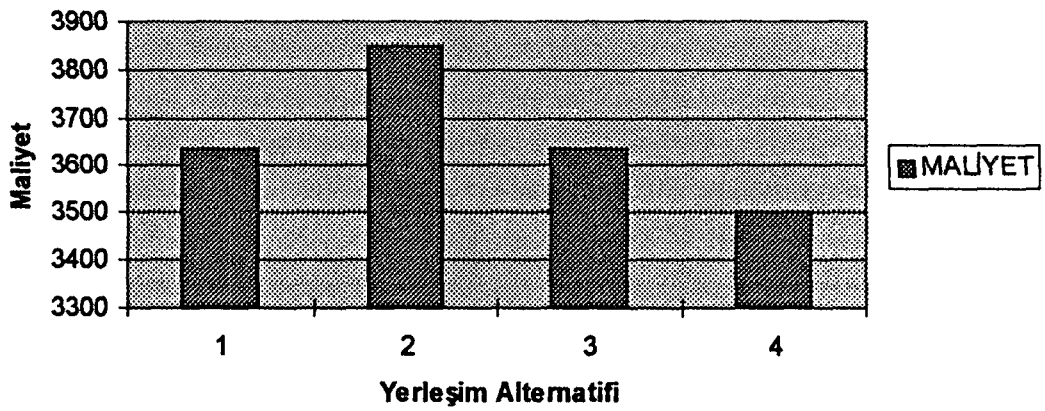
Şekil 5.2. İşletme İçin Alternatif Yerleşim

Tablo 5.1. Yerleşim Alternatifleri İçin Maliyetlerin Karşılaştırılması

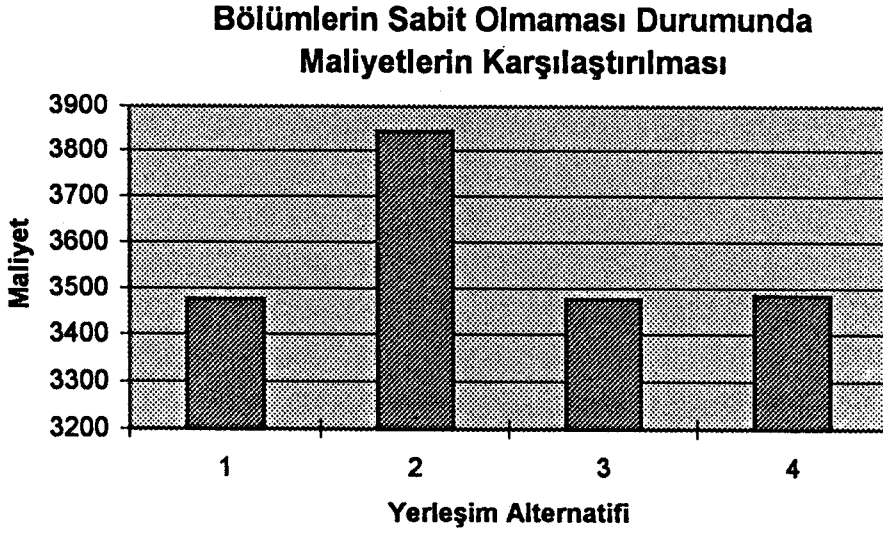


Tablo 5.2. Birinci Bölümün Sabit Olması Durumunda Maliyetlerin Karşılaştırılması

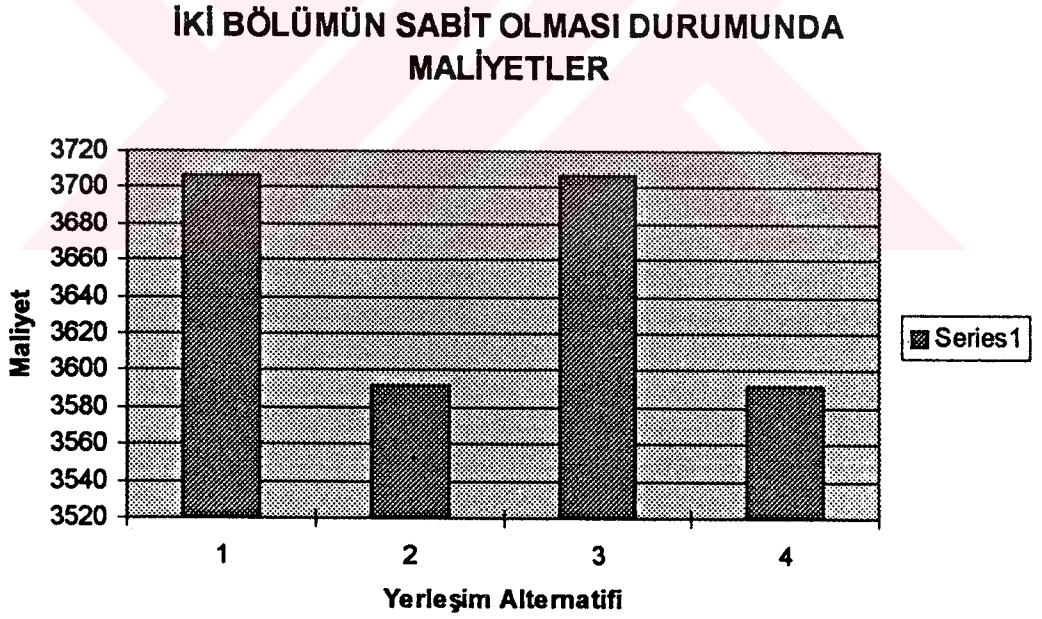
**BİRİNCİ BÖLÜMÜN SABİT OLDUĞU DURUMDA  
MALİYETLER**



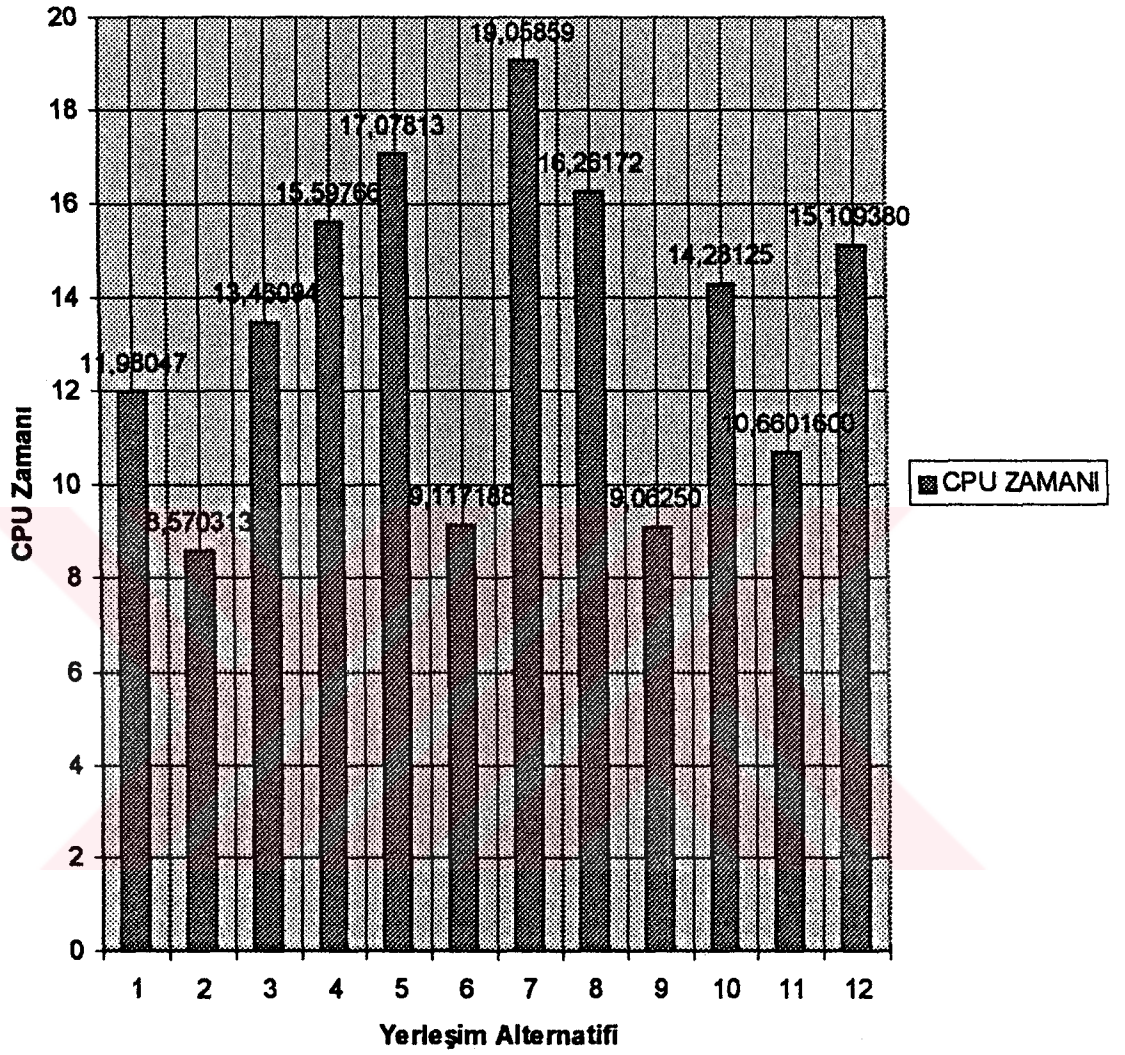
Tablo 5.3. Bölümlerin Sabit Olmaması Durumunda Maliyetlerin Karşılaştırılması



Tablo 5.4. İki Bölümün Sabit Olması Durumunda Maliyetlerin Karşılaştırılması



## CPU ZAMANLARININ KARŞILAŞTIRILMASI



**6. EKLER****EK 1**

Birinci Bölümün Sabit Kalması Durumunda Yerleşim

Kriter: Minimize

Bölüm Sayısı = 9

Başlangıç yerleşimi için satır sayısı = 28

Başlangıç yerleşimi için kolon sayısı = 36

Her bölümdeki hücre sayısı:

|       |   |   |     |             |
|-------|---|---|-----|-------------|
| Bölüm | 1 | : | 168 | Sabit       |
| Bölüm | 2 | : | 147 | Sabit değil |
| Bölüm | 3 | : | 105 | Sabit değil |
| Bölüm | 4 | : | 140 | Sabit değil |
| Bölüm | 5 | : | 126 | Sabit değil |
| Bölüm | 6 | : | 105 | Sabit değil |
| Bölüm | 7 | : | 126 | Sabit değil |
| Bölüm | 8 | : | 24  | Sabit değil |
| Bölüm | 9 | : | 21  | Sabit değil |

## Başlangıç Yerleşimi

|    | 123456789012345678901234567890123456    |                               |    |      |
|----|-----------------------------------------|-------------------------------|----|------|
| 1  | 000111111111122222222222222222222222000 |                               |    |      |
| 2  | 0 01                                    | 12                            |    | 20 0 |
| 3  | 0 01                                    | 12                            |    | 20 0 |
| 4  | 0 01                                    | 12                            |    | 20 0 |
| 5  | 0 01                                    | 12                            |    | 20 0 |
| 6  | 0 01                                    | 12                            |    | 20 0 |
| 7  | 0 01                                    | 12222222222222222222222222000 |    |      |
| 8  | 0 01                                    | 11111114444444444555555555    |    |      |
| 9  | 0 01                                    | 14                            | 45 | 5    |
| 10 | 0 01                                    | 14                            | 45 | 5    |
| 11 | 0 01                                    | 14                            | 45 | 5    |
| 12 | 0 01                                    | 14                            | 45 | 5    |
| 13 | 0 01                                    | 14                            | 45 | 5    |
| 14 | 00011111111111111114                    |                               | 45 | 5    |
| 15 | 88833333333333333334                    |                               | 45 | 5    |
| 16 | 8 83                                    | 34                            | 45 | 5    |
| 17 | 8 83                                    | 34                            | 45 | 5    |
| 18 | 8 83                                    | 34                            | 45 | 5    |
| 19 | 8 83                                    | 34                            | 45 | 5    |
| 20 | 8 83                                    | 34                            | 45 | 5    |
| 21 | 888333333333333333344444444555555555    |                               |    |      |
| 22 | 999666666666666666677777777777777777    |                               |    |      |
| 23 | 9 96                                    | 67                            |    | 7    |
| 24 | 9 96                                    | 67                            |    | 7    |
| 25 | 9 96                                    | 67                            |    | 7    |
| 26 | 9 96                                    | 67                            |    | 7    |
| 27 | 9 96                                    | 67                            |    | 7    |
| 28 | 999666666666666666677777777777777777    |                               |    |      |

## Bölümler Arası Akışlar

| -den | -e     |       |       |       |       |      |
|------|--------|-------|-------|-------|-------|------|
| 1    | 1: 0   | 2:+4  | 3: 0  | 4: 0  | 5: 0  | 6: 0 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 2    | 1: 0   | 2: 0  | 3: 0  | 4: 0  | 5:+5  | 6:+5 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 3    | 1: 0   | 2: 0  | 3: 0  | 4: 0  | 5:+10 | 6:+5 |
|      | 7:+5   | 8:+10 | 9: 0  |       |       |      |
| 4    | 1:+20  | 2:+10 | 3:+30 | 4: 0  | 5:+2  |      |
|      | 6:+10  | 7:+8  | 8:+5  | 9:+10 |       |      |
| 5    | 1:+5   | 2: 0  | 3:+10 | 4: 0  | 5: 0  |      |
|      | 6:+10  | 7:+4  | 8: 0  | 9: +5 |       |      |
| 6    | 1:+10  | 2: 0  | 3: 0  | 4: 0  | 5:+6  | 6:0  |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 7    | 1: +10 | 2: 0  | 3: 0  | 4: 0  | 5: 0  | 6: 0 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 8    | 1: +5  | 2: 0  | 3: 0  | 4: 0  | 5:+10 | 6: 0 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 9    | 1: +5  | 2: 0  | 3: 0  | 4: 0  | 5: 0  | 6: 0 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |

## Bölümler Arası Birim Maliyetler

| -den | -e   |      |      |      |      |
|------|------|------|------|------|------|
| 1    | 1:0  | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|      | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 2    | 1:+1 | 2: 0 | 3:+1 | 4:+1 | 5:+1 |
|      | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 3    | 1:+1 | 2:+1 | 3: 0 | 4:+1 | 5:+1 |
|      | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 4    | 1:+1 | 2:+1 | 3:+1 | 4: 0 | 5:+1 |
|      | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 5    | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5: 0 |
|      | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 6    | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|      | 6: 0 | 7:+1 | 8:+1 | 9:+1 |      |
| 7    | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|      | 6:+1 | 7:0  | 8:+1 | 9:+1 |      |
| 8    | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|      | 6:+1 | 7:+1 | 8: 0 | 9:+1 |      |
| 9    | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|      | 6:+1 | 7:+1 | 8:+1 | 9: 0 |      |

## Bir Bölümün Sabit Kalması Durumunda Son Yerleşim

## 1) İkili Değişim Metoduyla Son Yerleşim

```

123456789012345678901234567890123456
1 AAA111111111177777777777777777777AAA
2 A A1          17                      7A A
3 A A1          17                      7A A
4 A A1          17                      7A A
5 A A1          17                      7A A
6 A A1          177777777777777777777A A
7 A A1          122222222222222222222AAA
8 A A1          11111118333333332      2222
9 A A1          183          32          2
10 A A1         183          32          2
11 A A1         183          32          2
12 A A1         183          32          2
13 A A1         183          32          2
14 AAA1111111111111111833          32          2
15 5555555555555555599883          32          2
16 5          599883          32          2
17 5          5599883          32          2
18 5          5999883          32          2
19 5          5998883          32          2
20 5          5998 83          32          2
21 5555          59988833333333222222222
22 66655555555555 59994444444444444444444
23 6 666666666665559994          4
24 6          66666664          4
25 6          64          4
26 6          64          4
27 6          64          4
28 6666666666666666666664444444444444444

```

Toplam Maliyetin Minimum Değeri= 3635.997      Toplam  
 İterasyon Sayısı= 8 Düzgün Doğrusal Mesafe Kullanılarak  
 Toplam CPU saniyesi= 11.98047

2) Üçlü Değişim Metodu  
The Final Layout for f2

```

123456789012345678901234567890123456
1 AAA111111111122222222222222222222AAA
2 A A1 12 2A A
3 A A1 12 2A A
4 A A1 12 2A A
5 A A1 12 2A A
6 A A1 12 2A A
7 A A1 122222222222222222222222AAA
8 A A1 111111144444444447777777777
9 A A1 14 47 7
10 A A1 14 47 7
11 A A1 14 47 7
12 A A1 14 47 7
13 A A1 14 47 7
14 AAA11111111111111114 47 7
15 99999999999999999994 47 7
16 33333333333333339994 47 7
17 3 33334 47 7
18 3 34 47 7
19 3 33333333334 47 7
20 3 333333388888888884 47 7
21 3 38888888888888888444444447777777777
22 3 355555555555555555566666666666666
23 3 35 56 6
24 3 35 56 6
25 3 35 56 6
26 3 35 56 6
27 3 35 56 6
28 333555555555555555555666666666666666
Toplam Maliyetin Minimum Değeri= 3850.764 Toplam
İterasyon Sayısı= 3
Düzgün Doğrusal Mesafe Kullanılarak Toplam CPU saniyesi=
8.570313

```

3) Önce İkili Sonra Üçlü Değişim Metodu  
The Final Layout for f2

|    | 123456789012345678901234567890123456  |                               |      |
|----|---------------------------------------|-------------------------------|------|
| 1  | AAA111111111177777777777777777777AAA  |                               |      |
| 2  | A A1                                  | 17                            | 7A A |
| 3  | A A1                                  | 17                            | 7A A |
| 4  | A A1                                  | 17                            | 7A A |
| 5  | A A1                                  | 17                            | 7A A |
| 6  | A A1                                  | 17777777777777777777777777A A |      |
| 7  | A A1                                  | 122222222222222222222222AAA   |      |
| 8  | A A1                                  | 11111118333333332             | 2222 |
| 9  | A A1                                  | 183                           | 32 2 |
| 10 | A A1                                  | 183                           | 32 2 |
| 11 | A A1                                  | 183                           | 32 2 |
| 12 | A A1                                  | 183                           | 32 2 |
| 13 | A A1                                  | 183                           | 32 2 |
| 14 | AAA111111111111111833                 | 32                            | 2    |
| 15 | 555555555555555599883                 | 32                            | 2    |
| 16 | 5                                     | 599883                        | 32 2 |
| 17 | 5                                     | 5599883                       | 32 2 |
| 18 | 5                                     | 5999883                       | 32 2 |
| 19 | 5                                     | 5998883                       | 32 2 |
| 20 | 5                                     | 5998 83                       | 32 2 |
| 21 | 5555                                  | 5998883333333222222222        |      |
| 22 | 66655555555555 5999444444444444444444 |                               |      |
| 23 | 6 6666666665559994                    |                               | 4    |
| 24 | 6                                     | 6666664                       | 4    |
| 25 | 6                                     | 64                            | 4    |
| 26 | 6                                     | 64                            | 4    |
| 27 | 6                                     | 64                            | 4    |
| 28 | 6666666666666666444444444444444444    |                               |      |

Toplam Maliyetin Minimum Değeri= 3635.997 Toplam  
İterasyon Sayısı= 8 Düzgün Doğrusal Mesafe Kullanılarak  
Toplam CPU saniyesi= 13.46094

## 4) Önce Üçlü Sonra İkili Değişim Metodu

|    | 123456789012345678901234567890123456   |  |
|----|----------------------------------------|--|
| 1  | AAA11111111117777777777777777777777AAA |  |
| 2  | A A1 17 7A A                           |  |
| 3  | A A1 17 7A A                           |  |
| 4  | A A1 17 7A A                           |  |
| 5  | A A1 17 7A A                           |  |
| 6  | A A1 1777777777777777777777777A A      |  |
| 7  | A A1 1666666666666666666666666AAA      |  |
| 8  | A A1 111111155555555556 6666           |  |
| 9  | A A1 15 56 6                           |  |
| 10 | A A1 15 56 6                           |  |
| 11 | A A1 15 56 6                           |  |
| 12 | A A1 15 56 6                           |  |
| 13 | A A1 15 56 6                           |  |
| 14 | AAA1111111111111115 56 6               |  |
| 15 | 999999999999999995 56 6                |  |
| 16 | 8888888888888888995 56666666 6         |  |
| 17 | 8888888888333333335 5222222666         |  |
| 18 | 3333333333 35 52 2222                  |  |
| 19 | 3 35 52 2                              |  |
| 20 | 3 35 52 2                              |  |
| 21 | 3 33333333333333335555555552 2         |  |
| 22 | 3 34444444444444444442222222 2         |  |
| 23 | 3 34 42 2                              |  |
| 24 | 3 34 42 2                              |  |
| 25 | 3 34 42 2                              |  |
| 26 | 3 34 42 2                              |  |
| 27 | 3 34 42 2                              |  |
| 28 | 333444444444444444442222222222222222   |  |

Toplam Maliyetin Minimum Değeri= 3502.765 Toplam İterasyon Sayısı= 7

Düzgün Doğrusal Mesafe Kullanılarak Toplam CPU saniyesi= 15.59766

Bölümlerin Sabit Olmadığı Durumda Yerleşim  
Başlangıç Değerleri

Kriter: Minimize

Bölüm Sayısı= 9

Başlangıç Çözümündeki Satır Sayısı= 28

Başlangıç Çözümündeki Sütun Sayısı= 36

Her Bölümdeki Hücre Sayısı:

|        |   |     |             |
|--------|---|-----|-------------|
| Bölüm1 | : | 168 | Sabit Değil |
| Bölüm2 | : | 147 | Sabit Değil |
| Bölüm3 | : | 105 | Sabit Değil |
| Bölüm4 | : | 140 | Sabit Değil |
| Bölüm5 | : | 126 | Sabit Değil |
| Bölüm6 | : | 105 | Sabit Değil |
| Bölüm7 | : | 126 | Sabit Değil |
| Bölüm8 | : | 24  | Sabit Değil |
| Bölüm9 | : | 21  | Sabit Değil |

Toplam 962

## BAŞLANGIÇ YERLEŞİMİ

## Başlangıç Yerleşimi

|    | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1  | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 0 |
| 2  | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 |
| 3  | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 |
| 4  | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 |
| 5  | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 |
| 6  | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 |
| 7  | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 0 |
| 8  | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 |
| 9  | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 10 | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 11 | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 12 | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 13 | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 14 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 |   |   |   |   |   |   | 4 | 5 | 5 |
| 15 | 8 | 8 | 8 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 |   |   |   |   |   |   | 4 | 5 | 5 |
| 16 | 8 | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 17 | 8 | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 18 | 8 | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 19 | 8 | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 20 | 8 | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   | 4 | 5 | 5 |
| 21 | 8 | 8 | 8 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 |
| 22 | 9 | 9 | 9 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 |
| 23 | 9 | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   | 7 |
| 24 | 9 | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   | 7 |
| 25 | 9 | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   | 7 |
| 26 | 9 | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   | 7 |
| 27 | 9 | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   | 7 |
| 28 | 9 | 9 | 9 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 |

## Bölümler Arası Akışlar

| -den | -e     |       |       |       |       |      |
|------|--------|-------|-------|-------|-------|------|
| 1    | 1: 0   | 2:+4  | 3: 0  | 4: 0  | 5: 0  | 6: 0 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 2    | 1: 0   | 2: 0  | 3: 0  | 4: 0  | 5:+5  | 6:+5 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 3    | 1: 0   | 2: 0  | 3: 0  | 4: 0  | 5:+10 | 6:+5 |
|      | 7:+5   | 8:+10 | 9: 0  |       |       |      |
| 4    | 1:+20  | 2:+10 | 3:+30 | 4: 0  | 5:+2  |      |
|      | 6:+10  | 7:+8  | 8:+5  | 9:+10 |       |      |
| 5    | 1:+5   | 2: 0  | 3:+10 | 4: 0  | 5: 0  |      |
|      | 6:+10  | 7:+4  | 8: 0  | 9: +5 |       |      |
| 6    | 1:+10  | 2: 0  | 3: 0  | 4: 0  | 5:+6  | 6:0  |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 7    | 1: +10 | 2: 0  | 3: 0  | 4: 0  | 5: 0  | 6: 0 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 8    | 1: +5  | 2: 0  | 3: 0  | 4: 0  | 5:+10 | 6: 0 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |
| 9    | 1: +5  | 2: 0  | 3: 0  | 4: 0  | 5: 0  | 6: 0 |
|      | 7: 0   | 8: 0  | 9: 0  |       |       |      |

Bölümler Arası Birim Maliyetler  
-den -e

|   |      |      |      |      |      |
|---|------|------|------|------|------|
| 1 | 1:0  | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 2 | 1:+1 | 2: 0 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 3 | 1:+1 | 2:+1 | 3: 0 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 4 | 1:+1 | 2:+1 | 3:+1 | 4: 0 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 5 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5: 0 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 6 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6: 0 | 7:+1 | 8:+1 | 9:+1 |      |
| 7 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:0  | 8:+1 | 9:+1 |      |
| 8 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8: 0 | 9:+1 |      |
| 9 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9: 0 |      |

—

## Bölümlerin Sabit Olmaması Durumunda Son Yerleşim

### 1) İkili Değişim

|                                     |                                          |                                 |                            |
|-------------------------------------|------------------------------------------|---------------------------------|----------------------------|
|                                     | 123456789012345678901234567890123456     |                                 |                            |
| 1                                   | AAA2222222222277777777777777777777777AAA |                                 |                            |
| 2                                   | A A2                                     | 277                             | 7A A                       |
| 3                                   | A A2                                     | 27                              | 7A A                       |
| 4                                   | A A2                                     | 27                              | 7A A                       |
| 5                                   | A A2                                     | 27                              | 7A A                       |
| 6                                   | A A2                                     | 2777777777777777777777777777A A |                            |
| 7                                   | A A2                                     | 2711111111111111111111111111AAA |                            |
| 8                                   | A A2                                     | 222221183333333331              | 1111                       |
| 9                                   | A A2                                     | 21183                           | 31 1                       |
| 10                                  | A A2                                     | 21183                           | 31 1                       |
| 11                                  | A A2                                     | 21183                           | 31 1                       |
| 12                                  | A A2                                     | 21183                           | 31 1                       |
| 13                                  | A A2 2222222221183                       | 31                              | 1                          |
| 14                                  | AAA222221111111111833                    | 31                              | 1                          |
| 15                                  | 5555555555555555599883                   | 31                              | 1                          |
| 16                                  | 5                                        | 599883                          | 31 1                       |
| 17                                  | 5                                        | 5599883                         | 31 1                       |
| 18                                  | 5                                        | 5999883                         | 31 1                       |
| 19                                  | 5                                        | 5998883                         | 31 1                       |
| 20                                  | 5                                        | 5998 83                         | 31 1                       |
| 21                                  | 5555                                     | 599888333333331111111111        |                            |
| 22                                  | 66655555555555 599944444444444444444444  |                                 |                            |
| 23                                  | 6 66666666665559994                      |                                 | 4                          |
| 24                                  | 6 66666664                               |                                 | 4                          |
| 25                                  | 6 64                                     |                                 | 4                          |
| 26                                  | 6 64                                     |                                 | 4                          |
| 27                                  | 6 64                                     |                                 | 4                          |
| 28                                  | 66666666666666666666444444444444444444   |                                 |                            |
| Toplam Maliyetin Minimum Değeri=    |                                          | 3476.38                         | Toplam İterasyon Sayısı= 9 |
| Düzgün Doğrusal Mesafe Kullanılarak |                                          | Toplam CPU saniyesi= 17.07813   |                            |

## 2) Üçlü Değişim

The Final Layout for f2

|    | 1   | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6   |     |   |   |   |
|----|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|-----|-----|---|---|---|
| 1  | AAA | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | AAA |     |   |   |   |
| 2  | A   | A | 1 |   |   |   |   |   |   |   |   | 1 | 2 | 2 |   |   |   |   |   |   |   |   |   |   |   | 2A  | A   |   |   |   |
| 3  | A   | A | 1 |   |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   | 2A  | A   |   |   |   |
| 4  | A   | A | 1 |   |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   | 2A  | A   |   |   |   |
| 5  | A   | A | 1 |   |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   | 2A  | A   |   |   |   |
| 6  | A   | A | 1 |   |   |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   | 2A  | A   |   |   |   |
| 7  | A   | A | 1 |   |   |   |   |   |   |   |   | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2   | AAA |   |   |   |
| 8  | A   | A | 1 |   |   |   |   |   |   |   |   | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 7 | 7   | 7   | 7 | 7 |   |
| 9  | A   | A | 1 |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   |   |   |   | 4 | 7   |     | 7 |   |   |
| 10 | A   | A | 1 |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   |   |   |   | 4 | 7   |     | 7 |   |   |
| 11 | A   | A | 1 |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   |   |   |   | 4 | 7   |     | 7 |   |   |
| 12 | A   | A | 1 |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   |   |   |   | 4 | 7   |     | 7 |   |   |
| 13 | A   | A | 1 |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   |   |   |   | 4 | 7   |     | 7 |   |   |
| 14 | AAA | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 |   |   |   |   |   |   |   | 4 | 7   |     | 7 |   |   |
| 15 | 9   | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 4   |     | 7 |   |   |
| 16 | 3   | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 4 |     | 7   |   |   |   |
| 17 | 3   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 3 | 3 | 3 | 4 |   |   |   | 4 | 7   |     | 7 |   |   |
| 18 | 3   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   | 4 | 7   |     | 7 |   |   |
| 19 | 3   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 |   | 4   | 7   |   | 7 |   |
| 20 | 3   | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 4 |   |   |   |   |   |   |   | 4   | 7   |   | 7 |   |
| 21 | 3   | 3 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 7 | 7   | 7   | 7 | 7 | 7 |
| 22 | 3   | 3 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6   | 6   | 6 | 6 | 6 |
| 23 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 5 | 6   |     |   | 6 |   |
| 24 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 5 | 6   |     |   | 6 |   |
| 25 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 5 | 6   |     |   | 6 |   |
| 26 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 5 | 6   |     |   | 6 |   |
| 27 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 5 | 6   |     |   | 6 |   |
| 28 | 3   | 3 | 3 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6   | 6   | 6 | 6 | 6 |

Toplam Maliyetin Minimum Değeri= 3844.821 Toplam İterasyon Sayısı= 3

Düzgün Doğrusal Mesafe Kullanılarak Toplam CPU saniyesi= 9.117188

3) Önce İkili Sonra Üçlü Değişim  
The Final Layout for f2

|    | 123456789012345678901234567890123456   |                                 | 12345678901234567890123456 |
|----|----------------------------------------|---------------------------------|----------------------------|
| 1  | AAA22222222222777777777777777777777AAA |                                 |                            |
| 2  | A A2                                   | 277                             | 7A A                       |
| 3  | A A2                                   | 27                              | 7A A                       |
| 4  | A A2                                   | 27                              | 7A A                       |
| 5  | A A2                                   | 27                              | 7A A                       |
| 6  | A A2                                   | 2777777777777777777777777777A A |                            |
| 7  | A A2                                   | 2711111111111111111111111111AAA |                            |
| 8  | A A2                                   | 222221183333333331              | 1111                       |
| 9  | A A2                                   | 21183                           | 31 1                       |
| 10 | A A2                                   | 21183                           | 31 1                       |
| 11 | A A2                                   | 21183                           | 31 1                       |
| 12 | A A2                                   | 21183                           | 31 1                       |
| 13 | A A2                                   | 222222221183                    | 31 1                       |
| 14 | AAA222221111111111833                  | 31                              | 1                          |
| 15 | 555555555555555599883                  | 31                              | 1                          |
| 16 | 5                                      | 599883                          | 31 1                       |
| 17 | 5                                      | 5599883                         | 31 1                       |
| 18 | 5                                      | 5999883                         | 31 1                       |
| 19 | 5                                      | 5998883                         | 31 1                       |
| 20 | 5                                      | 5998 83                         | 31 1                       |
| 21 | 5555                                   | 5998883333333111111111          |                            |
| 22 | 666555555555555599944444444444444444   |                                 |                            |
| 23 | 6 66666666665559994                    |                                 | 4                          |
| 24 | 6                                      | 66666664                        | 4                          |
| 25 | 6                                      | 64                              | 4                          |
| 26 | 6                                      | 64                              | 4                          |
| 27 | 6                                      | 64                              | 4                          |
| 28 | 666666666666666666664444444444444444   |                                 |                            |

Toplam Maliyetin Minimum Değeri= 3476.38 Toplam İterasyon Sayısı= 9  
Düzgün Doğrusal Mesafe Kullanılarak Toplam CPU saniyesi= 19.05859

4) Önce Üçlü Sonra İkili Değişim  
The Final Layout for f2

|    | 1   | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6     |
|----|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|-------|
| 1  | AAA | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7AAA  |
| 2  | A   | A | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 11A A |
| 3  | A   | A | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 1A A  |
| 4  | A   | A | 7 |   |   |   |   |   |   |   | 7 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   | 1A A  |
| 5  | A   | A | 7 |   |   |   |   |   |   |   | 7 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   | 1A A  |
| 6  | A   | A | 7 |   |   |   |   |   |   |   | 7 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1A A  |
| 7  | A   | A | 7 |   |   |   |   |   |   |   | 7 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | AAA   |
| 8  | A   | A | 7 |   |   |   |   |   |   |   | 7 | 7 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 9 | 9 | 2 | 2222  |
| 9  | A   | A | 7 |   |   |   |   |   |   |   | 7 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   | 2     |
| 10 | A   | A | 7 |   |   |   |   |   |   |   | 7 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   | 2     |
| 11 | A   | A | 7 |   |   |   |   |   |   |   | 7 | 7 |   |   |   |   |   |   |   |   |   |   |   |   |   | 2     |
| 12 | A   | A | 7 |   |   |   |   |   |   |   | 7 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   | 2     |
| 13 | A   | A | 7 |   |   |   |   |   |   |   | 7 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   | 2     |
| 14 | AAA | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 1 | 1 | 1 | 4 |   |   |   |   |   |   |   |   |   |   | 2     |
| 15 | 4   | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 2     |
| 16 | 3   | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 2     |
| 17 | 3   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 2     |
| 18 | 3   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 2     |
| 19 | 3   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 2     |
| 20 | 3   | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 2     |
| 21 | 3   | 3 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 4 | 4 | 4 | 4 | 2     |
| 22 | 3   | 3 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 2     |
| 23 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 6     |
| 24 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 6     |
| 25 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 6     |
| 26 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 6     |
| 27 | 3   | 3 | 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 6     |
| 28 | 3   | 3 | 3 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6     |

Toplam Maliyetin Minimum Değeri= 3485.175 Toplam İterasyon Sayısı= 6

Düzgün Doğrusal Mesafe Kullanılarak Toplam CPU saniyesi= 16.26172

İlk İki Bölümün Sabit Olması Durumunda Yerleşim  
Başlangıç Değerleri

Kriter: Minimize

Bölüm Sayısı= 9

Başlangıç Çözümündeki Satır Sayısı= 28

Başlangıç Çözümündeki Sütun Sayısı= 36

Her Bölümdeki Hücre Sayısı:

|        |   |     |             |
|--------|---|-----|-------------|
| Bölüm1 | : | 168 | Sabit       |
| Bölüm2 | : | 140 | Sabit       |
| Bölüm3 | : | 105 | Sabit Değil |
| Bölüm4 | : | 140 | Sabit Değil |
| Bölüm5 | : | 126 | Sabit Değil |
| Bölüm6 | : | 105 | Sabit Değil |
| Bölüm7 | : | 126 | Sabit Değil |
| Bölüm8 | : | 24  | Sabit Değil |
| Bölüm9 | : | 21  | Sabit Değil |

Toplam955

## CRAFT UYGULAMASI İÇİN BAŞLANGIÇ YERLEŞİMİ

## Başlangıç Yerleşimi

|    | 1   | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 |   |
|----|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1  | 000 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2  | 0   | 0 | 0 | 1 |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 | 0 |   |   |   |   |
| 3  | 0   | 0 | 0 | 1 |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 | 0 |   |   |   |   |
| 4  | 0   | 0 | 0 | 1 |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 | 0 |   |   |   |   |
| 5  | 0   | 0 | 0 | 1 |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 | 0 |   |   |   |   |
| 6  | 0   | 0 | 0 | 1 |   |   |   |   |   |   | 1 | 2 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | 2 | 0 | 0 |   |   |   |   |
| 7  | 0   | 0 | 0 | 1 |   |   |   |   |   |   | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 0 | 0 | 0 |   |   |   |   |
| 8  | 0   | 0 | 0 | 1 |   |   |   |   |   |   | 1 | 1 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |   |   |   |   |
| 9  | 0   | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 10 | 0   | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 11 | 0   | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 12 | 0   | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 13 | 0   | 0 | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   | 1 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 14 | 0   | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 15 | 8   | 8 | 8 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 16 | 8   | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 17 | 8   | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 18 | 8   | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 19 | 8   | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 20 | 8   | 8 | 8 | 3 |   |   |   |   |   |   |   |   |   |   |   |   | 3 | 4 |   |   |   |   |   |   |   | 4 | 5 |   |   |   | 5 |   |   |   |   |   |   |
| 21 | 8   | 8 | 8 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |   |   |   |   |
| 22 | 9   | 9 | 9 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 |   |   |   |   |
| 23 | 9   | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   |   |   |   |   |   | 7 |   |   |   |   |   |   |
| 24 | 9   | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   |   |   |   |   |   | 7 |   |   |   |   |   |   |
| 25 | 9   | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   |   |   |   |   |   | 7 |   |   |   |   |   |   |
| 26 | 9   | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   |   |   |   |   |   | 7 |   |   |   |   |   |   |
| 27 | 9   | 9 | 9 | 6 |   |   |   |   |   |   |   |   |   |   |   |   | 6 | 7 |   |   |   |   |   |   |   |   |   |   |   |   | 7 |   |   |   |   |   |   |
| 28 | 9   | 9 | 9 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 |   |   |   |   |

## Bölümler Arası Akışlar

| -den | -e     |        |        |        |        |       |
|------|--------|--------|--------|--------|--------|-------|
| 1    | 1: 0   | 2: +4  | 3: 0   | 4: 0   | 5: 0   | 6: 0  |
|      | 7: 0   | 8: 0   | 9: 0   |        |        |       |
| 2    | 1: 0   | 2: 0   | 3: 0   | 4: 0   | 5: +5  | 6: +5 |
|      | 7: 0   | 8: 0   | 9: 0   |        |        |       |
| 3    | 1: 0   | 2: 0   | 3: 0   | 4: 0   | 5: +10 | 6: +5 |
|      | 7: +5  | 8: +10 | 9: 0   |        |        |       |
| 4    | 1: +20 | 2: +10 | 3: +30 | 4: 0   | 5: +2  |       |
|      | 6: +10 | 7: +8  | 8: +5  | 9: +10 |        |       |
| 5    | 1: +5  | 2: 0   | 3: +10 | 4: 0   | 5: 0   |       |
|      | 6: +10 | 7: +4  | 8: 0   | 9: +5  |        |       |
| 6    | 1: +10 | 2: 0   | 3: 0   | 4: 0   | 5: +6  | 6: 0  |
|      | 7: 0   | 8: 0   | 9: 0   |        |        |       |
| 7    | 1: +10 | 2: 0   | 3: 0   | 4: 0   | 5: 0   | 6: 0  |
|      | 7: 0   | 8: 0   | 9: 0   |        |        |       |
| 8    | 1: +5  | 2: 0   | 3: 0   | 4: 0   | 5: +10 | 6: 0  |
|      | 7: 0   | 8: 0   | 9: 0   |        |        |       |
| 9    | 1: +5  | 2: 0   | 3: 0   | 4: 0   | 5: 0   | 6: 0  |
|      | 7: 0   | 8: 0   | 9: 0   |        |        |       |

Bölümler Arası Birim Maliyetler  
-den -e

|   |      |      |      |      |      |
|---|------|------|------|------|------|
| 1 | 1:0  | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 2 | 1:+1 | 2: 0 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 3 | 1:+1 | 2:+1 | 3: 0 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 4 | 1:+1 | 2:+1 | 3:+1 | 4: 0 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 5 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5: 0 |
|   | 6:+1 | 7:+1 | 8:+1 | 9:+1 |      |
| 6 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6: 0 | 7:+1 | 8:+1 | 9:+1 |      |
| 7 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:0  | 8:+1 | 9:+1 |      |
| 8 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8: 0 | 9:+1 |      |
| 9 | 1:+1 | 2:+1 | 3:+1 | 4:+1 | 5:+1 |
|   | 6:+1 | 7:+1 | 8:+1 | 9: 0 |      |

İki Bölümün Sabit Kalması Durumunda Son Yerleşim  
1) İkili Değişim

|    | 123456789012345678901234567890123456 |
|----|--------------------------------------|
| 1  | AAA111111111122222222222222222222AAA |
| 2  | A A1 12 2A A                         |
| 3  | A A1 12 2A A                         |
| 4  | A A1 12 2A A                         |
| 5  | A A1 12 2A A                         |
| 6  | A A1 12 2A A                         |
| 7  | A A1 1222222222222222222222AAA       |
| 8  | A A1 1111111833333333777777777       |
| 9  | A A1 183 37 7                        |
| 10 | A A1 183 37 7                        |
| 11 | A A1 183 37 7                        |
| 12 | A A1 183 37 7                        |
| 13 | A A1 183 37 7                        |
| 14 | AAA1111111111111833 37 7             |
| 15 | 55555555555555599883 37 7            |
| 16 | 5 599883 37 7                        |
| 17 | 5 599883 37 7                        |
| 18 | 5 599883 37 7                        |
| 19 | 5 599883 37 7                        |
| 20 | 5 5998 83 37 7                       |
| 21 | 5555 599888333333377777777           |
| 22 | 66655555555555 59994444444444444444  |
| 23 | 6 666666666559994 4                  |
| 24 | 6 66666664 4                         |
| 25 | 6 64 4                               |
| 26 | 6 64 4                               |
| 27 | 6 64 4                               |
| 28 | 66666666666666664444444444444444     |

Toplam Maliyetin Minimum Değeri= 3707.012 Toplam  
İterasyon= 7  
Düzgün Doğrusal Mesafe Kullanılarak Toplam CPU dakikası=  
9.0625

## 2) Üçlü Değişim

|                                                         | 123456789012345678901234567890123456   |                             |       |      |
|---------------------------------------------------------|----------------------------------------|-----------------------------|-------|------|
| 1                                                       | AAA11111111112222222222222222222222AAA |                             |       |      |
| 2                                                       | A A1                                   | 12                          |       | 2A A |
| 3                                                       | A A1                                   | 12                          |       | 2A A |
| 4                                                       | A A1                                   | 12                          |       | 2A A |
| 5                                                       | A A1                                   | 12                          |       | 2A A |
| 6                                                       | A A1                                   | 12                          |       | 2A A |
| 7                                                       | A A1                                   | 122222222222222222222222AAA |       |      |
| 8                                                       | A A1                                   | 1111111333333444777777777   |       |      |
| 9                                                       | A A1                                   | 13                          | 34 47 | 7    |
| 10                                                      | A A1                                   | 13                          | 34 47 | 7    |
| 11                                                      | A A1                                   | 13                          | 34 47 | 7    |
| 12                                                      | A A1                                   | 13                          | 34 47 | 7    |
| 13                                                      | A A1                                   | 13                          | 34 47 | 7    |
| 14                                                      | AAA1111111111111113                    | 34 47                       |       | 7    |
| 15                                                      | 999999999999999993                     | 34 47                       |       | 7    |
| 16                                                      | 6666666666666669993                    | 34 47                       |       | 7    |
| 17                                                      | 6                                      | 66663                       | 34 47 | 7    |
| 18                                                      | 6                                      | 63                          | 34 47 | 7    |
| 19                                                      | 6                                      | 66666666663                 | 34 47 | 7    |
| 20                                                      | 6 66666668888888883                    | 34 47                       |       | 7    |
| 21                                                      | 6 68888888888888883333                 | 34 47777777777              |       |      |
| 22                                                      | 6 655555555555555553                   | 34 44444444444              |       |      |
| 23                                                      | 6 65                                   | 53 34                       |       | 4    |
| 24                                                      | 6 65                                   | 53 34                       |       | 4    |
| 25                                                      | 6 65                                   | 53 34                       |       | 4    |
| 26                                                      | 6 65                                   | 53 34                       |       | 4    |
| 27                                                      | 6 65                                   | 53 34                       |       | 4    |
| 28                                                      | 66655555555555555333444444444444       |                             |       |      |
| Toplam Maliyetin Minimum Değeri= 3592.207 Toplam        |                                        |                             |       |      |
| İterasyon= 6                                            |                                        |                             |       |      |
| Düzgün Doğrusal Mesafe KullanılarakToplam CPU dakikası= |                                        |                             |       |      |
| 14.28125                                                |                                        |                             |       |      |

## 3) Önce İkili Sonra Üçlü Değişim

|    | 123456789012345678901234567890123456    |                               |      |
|----|-----------------------------------------|-------------------------------|------|
| 1  | AAA11111111112222222222222222222222AAA  |                               |      |
| 2  | A A1                                    | 12                            | 2A A |
| 3  | A A1                                    | 12                            | 2A A |
| 4  | A A1                                    | 12                            | 2A A |
| 5  | A A1                                    | 12                            | 2A A |
| 6  | A A1                                    | 12                            | 2A A |
| 7  | A A1                                    | 12222222222222222222222222AAA |      |
| 8  | A A1                                    | 1111111833333333337777777777  |      |
| 9  | A A1                                    | 183                           | 37 7 |
| 10 | A A1                                    | 183                           | 37 7 |
| 11 | A A1                                    | 183                           | 37 7 |
| 12 | A A1                                    | 183                           | 37 7 |
| 13 | A A1                                    | 183                           | 37 7 |
| 14 | AAA1111111111111111833                  | 37                            | 7    |
| 15 | 555555555555555599883                   | 37                            | 7    |
| 16 | 5                                       | 599883                        | 37 7 |
| 17 | 5                                       | 5599883                       | 37 7 |
| 18 | 5                                       | 5999883                       | 37 7 |
| 19 | 5                                       | 5998883                       | 37 7 |
| 20 | 5                                       | 5998 83                       | 37 7 |
| 21 | 5555                                    | 599888333333337777777777      |      |
| 22 | 6665555555555555 5999444444444444444444 |                               |      |
| 23 | 6 66666666665559994                     |                               | 4    |
| 24 | 6 66666664                              |                               | 4    |
| 25 | 6 64                                    |                               | 4    |
| 26 | 6 64                                    |                               | 4    |
| 27 | 6 64                                    |                               | 4    |
| 28 | 666666666666666666444444444444444444    |                               |      |

Toplam Maliyetin Minimum Değeri= 3707.012 Toplam  
İterasyon= 7  
Düzgün Doğrusal Mesafe KullanılarakToplam CPU dakikası=  
10.66016

## 4) Önce Üçlü Sonra İkili Değişim

|    | 123456789012345678901234567890123456     |                                 | 123456789012345678901234567890123456 |      |
|----|------------------------------------------|---------------------------------|--------------------------------------|------|
| 1  | AAA1111111111122222222222222222222222AAA |                                 |                                      |      |
| 2  | A A1                                     | 12                              |                                      | 2A A |
| 3  | A A1                                     | 12                              |                                      | 2A A |
| 4  | A A1                                     | 12                              |                                      | 2A A |
| 5  | A A1                                     | 12                              |                                      | 2A A |
| 6  | A A1                                     | 12                              |                                      | 2A A |
| 7  | A A1                                     | 1222222222222222222222222222AAA |                                      |      |
| 8  | A A1                                     | 111111133333334447777777777     |                                      |      |
| 9  | A A1                                     | 13                              | 34 47                                | 7    |
| 10 | A A1                                     | 13                              | 34 47                                | 7    |
| 11 | A A1                                     | 13                              | 34 47                                | 7    |
| 12 | A A1                                     | 13                              | 34 47                                | 7    |
| 13 | A A1                                     | 13                              | 34 47                                | 7    |
| 14 | AAA111111111111111113                    |                                 | 34 47                                | 7    |
| 15 | 99999999999999999993                     |                                 | 34 47                                | 7    |
| 16 | 666666666666666669993                    |                                 | 34 47                                | 7    |
| 17 | 6                                        | 66663                           | 34 47                                | 7    |
| 18 | 6                                        | 63                              | 34 47                                | 7    |
| 19 | 6                                        | 66666666663                     | 34 47                                | 7    |
| 20 | 6                                        | 666666688888888883              | 34 47                                | 7    |
| 21 | 6                                        | 6888888888888888883333          | 34 47777777777                       |      |
| 22 | 6                                        | 65555555555555555553            | 34 44444444444                       |      |
| 23 | 6                                        | 65                              | 53 34                                | 4    |
| 24 | 6                                        | 65                              | 53 34                                | 4    |
| 25 | 6                                        | 65                              | 53 34                                | 4    |
| 26 | 6                                        | 65                              | 53 34                                | 4    |
| 27 | 6                                        | 65                              | 53 34                                | 4    |
| 28 | 666555555555555555555333444444444444     |                                 |                                      |      |

Toplam Maliyetin Minimum Değeri= 3592.207 Toplam İterasyon= 6

Düzgün Doğrusal Mesafe Kullanılarak Toplam CPU dakikası= 15.10938

## EK 2

## MIP MODELİNİN LINDO PROGRAMINDA FORMÜLASYONU

|                |           |             |
|----------------|-----------|-------------|
| AP: $\alpha^+$ | XR: $x''$ | ZB: $z^y$   |
| BP: $\beta^+$  | XL: $x'$  | ZD: $z^y$   |
| AN: $\alpha^-$ | YL: $y'$  | A: $\alpha$ |
| BN: $\beta^-$  | YU: $y''$ | B: $\beta$  |

MIN 5AP12+5AN12+5BP12+5BN12+  
 4AP14+4AN14+4BP14+4BN14+  
 4AP24+4AN24+4BP24+4BN24+  
 5AP26+5AN26+5BP26+5BN26+  
 10AP34+10AN34+10BP34+10BN34+  
 10AP38+10AN38+10BP38+10BN38+  
 10AP35+10AN35+10BP35+10BN35+  
 5AP37+5AN37+5BP37+5BN37+  
 5AP36+5AN36+5BN36+5BP36+  
 AP41+AN41+BP41+BN41+  
 4AP42+4AN42+4BP42+4BN42+  
 10AP43+10AN43+10BP43+10BN43+  
 5AP45+5AN45+5BP45+5BN45+  
 10AP46+10AN46+10BP46+10BN46+  
 10AP47+10AN47+10BP47+10BN47+  
 5AP48+5AN48+5BP48+5BN48+  
 5AP49+5AN49+5BP49+5BN49+  
 5AP54+5AN54+5BP54+5BN54+  
 5AP59+5AN59+5BP59+5BN59+  
 4AP57+4AN57+4BP57+4BN57+  
 10AP53+10AN53+10BP53+10BN53+  
 10AP56+10AN56+10BP56+10BN56+  
 10AP64+10AN64+10BP64+10BN64+

6AP65+6AN65+6BP65+6BN65+  
10AP75+10AN75+10BP75+10BN75+  
5AP84+5AN84+5BP84+5BN84+  
10AP85+10AN85+10BP85+10BN85+  
5AP94+5AN94+5BP94+5BN94

subject to

XR1-XL1 $\leq$  35  
XR1-XL1 $\geq$  25  
XR2-XL2 $\leq$  50  
XR2-XL2 $\geq$  40  
XR3-XL3 $\leq$  35  
XR3-XL3 $\geq$  25  
XR4-XL4 $\leq$  25  
XR4-XL4 $\geq$  20  
XR5-XL5 $\leq$  25  
XR5-XL5 $\geq$  20  
XR6-XL6 $\leq$  10  
XR6-XL6 $\geq$  5  
XR7-XL7 $\leq$  35  
XR7-XL7 $\geq$  25  
XR8-XL8 $\leq$  50  
XR8-XL8 $\geq$  40  
XR9-XL9 $\leq$  10  
XR9-XL9 $\geq$  5  
YU1-YL1 $\leq$  40  
YU1-YL1 $\geq$  35  
YU2-YL2 $\leq$  25  
YU2-YL2 $\geq$  15  
YU3-YL3 $\leq$  25  
YU3-YL3 $\geq$  20  
YU4-YL4 $\leq$  40  
YU4-YL4 $\geq$  30  
YU5-YL5 $\leq$  40

YU5-YL5>= 30

YU6-YL6<= 20

YU6-YL6>= 15

YU7-YL7<= 25

YU7-YL7>= 20

YU8-YL8<= 20

YU8-YL8>= 15

YU9-YL9<= 20

YU9-YL9>= 15

XR1<=90

XR2<=90

XR3<=90

XR4<=90

XR5<=90

XR6<=90

XR7<=90

XR8<=90

XR9<=90

XR1-XL1>=0

XR2-XL2>=0

XR3-XL3>=0

XR4-XL4>=0

XR5-XL5>=0

XR6-XL6>=0

XR7-XL7>=0

XR8-XL8>=0

XR9-XL9>=0

YU1<=70

YU2<=70

YU3<=70

YU4<=70

YU5<=70

YU6<=70

$$YU7 \leq 70$$

$$YU8 \leq 70$$

$$YU9 \leq 70$$

$$YU1 - YL1 \geq 0$$

$$YU2 - YL2 \geq 0$$

$$YU3 - YL3 \geq 0$$

$$YU4 - YL4 \geq 0$$

$$YU5 - YL5 \geq 0$$

$$YU6 - YL6 \geq 0$$

$$YU7 - YL7 \geq 0$$

$$YU8 - YL8 \geq 0$$

$$YU9 - YL9 \geq 0$$

$$2XR1 - 2XL1 + 2YU1 - 2YL1 \geq 120$$

$$2XR2 - 2XL2 + 2YU2 - 2YL2 \geq 110$$

$$2XR3 - 2XL3 + 2YU3 - 2YL3 \geq 90$$

$$2XR4 - 2XL4 + 2YU4 - 2YL4 \geq 100$$

$$2XR5 - 2XL5 + 2YU5 - 2YL5 \geq 100$$

$$2XR6 - 2XL6 + 2YU6 - 2YL6 \geq 40$$

$$2XR7 - 2XL7 + 2YU7 - 2YL7 \geq 90$$

$$2XR8 - 2XL8 + 2YU8 - 2YL8 \geq 110$$

$$2XR9 - 2XL9 + 2YU9 - 2YL9 \geq 40$$

$$2XR1 - 2XL1 + 2YU1 - 2YL1 \leq 150$$

$$2XR2 - 2XL2 + 2YU2 - 2YL2 \leq 150$$

$$2XR3 - 2XL3 + 2YU3 - 2YL3 \leq 120$$

$$2XR4 - 2XL4 + 2YU4 - 2YL4 \leq 130$$

$$2XR5 - 2XL5 + 2YU5 - 2YL5 \leq 130$$

$$2XR6 - 2XL6 + 2YU6 - 2YL6 \leq 60$$

$$2XR7 - 2XL7 + 2YU7 - 2YL7 \leq 120$$

$$2XR8 - 2XL8 + 2YU8 - 2YL8 \leq 140$$

$$2XR9 - 2XL9 + 2YU9 - 2YL9 \leq 60$$

$$A1 - 0.5XR1 - 0.5XL1 = 0$$

$$A2 - 0.5XR2 - 0.5XL2 = 0$$

$$A3 - 0.5XR3 - 0.5XL3 = 0$$

A4-0.5XR4-0.5XL4=0  
A5-0.5XR5-0.5XL5=0  
A6-0.5XR6-0.5XL6=0  
A7-0.5XR7-0.5XL7=0  
A8-0.5XR8-0.5XL8=0  
A9-0.5XR9-0.5XL9=0  
B1-0.5YU1-0.5YL1=0  
B2-0.5YU2-0.5YL2=0  
B3-0.5YU3-0.5YL3=0  
B4-0.5YU4-0.5YL4=0  
B5-0.5YU5-0.5YL5=0  
B6-0.5YU6-0.5YL6=0  
B7-0.5YU7-0.5YL7=0  
B8-0.5YU8-0.5YL8=0  
B9-0.5YU9-0.5YL9=0  
A1-A2-AP12+AN12=0  
A1-A3-AP13+AN13=0  
A1-A4-AP14+AN14=0  
A1-A5-AP15+AN15=0  
A1-A6-AP16+AN16=0  
A1-A7-AP17+AN17=0  
A1-A8-AP18+AN18=0  
A1-A9-AP19+AN19=0  
A2-A1-AP21+AN21=0  
A2-A3-AP23+AN23=0  
A2-A4-AP24+AN24=0  
A2-A5-AP25+AN25=0  
A2-A6-AP26+AN26=0  
A2-A7-AP27+AN27=0  
A2-A8-AP28+AN28=0  
A2-A9-AP29+AN29=0  
A3-A1-AP31+AN31=0  
A3-A2-AP32+AN32=0

$$A3-A4-AP34+AN34=0$$

$$A3-A5-AP35+AN35=0$$

$$A3-A6-AP36+AN36=0$$

$$A3-A7-AP37+AN37=0$$

$$A3-A8-AP38+AN38=0$$

$$A3-A9-AP39+AN39=0$$

$$A4-A1-AP41+AN41=0$$

$$A4-A2-AP42+AN42=0$$

$$A4-A3-AP43+AN43=0$$

$$A4-A5-AP45+AN45=0$$

$$A4-A6-AP46+AN46=0$$

$$A4-A7-AP47+AN47=0$$

$$A4-A8-AP48+AN48=0$$

$$A4-A9-AP49+AN49=0$$

$$A5-A1-AP51+AN51=0$$

$$A5-A2-AP52+AN52=0$$

$$A5-A3-AP53+AN53=0$$

$$A5-A4-AP54+AN54=0$$

$$A5-A6-AP56+AN56=0$$

$$A5-A7-AP57+AN57=0$$

$$A5-A8-AP58+AN58=0$$

$$A5-A9-AP59+AN59=0$$

$$A6-A1-AP61+AN61=0$$

$$A6-A2-AP62+AN62=0$$

$$A6-A3-AP63+AN63=0$$

$$A6-A4-AP64+AN64=0$$

$$A6-A5-AP65+AN65=0$$

$$A6-A7-AP67+AN67=0$$

$$A6-A8-AP68+AN68=0$$

$$A6-A9-AP69+AN69=0$$

$$A7-A1-AP71+AN71=0$$

$$A7-A2-AP72+AN72=0$$

$$A7-A3-AP73+AN73=0$$

$$A7-A4-AP74+AN74=0$$

$$A7-A5-AP75+AN75=0$$

$$A7-A6-AP76+AN76=0$$

$$A7-A8-AP78+AN78=0$$

$$A7-A9-AP79+AN79=0$$

$$A8-A1-AP81+AN81=0$$

$$A8-A2-AP82+AN82=0$$

$$A8-A3-AP83+AN83=0$$

$$A8-A4-AP84+AN84=0$$

$$A8-A5-AP85+AN85=0$$

$$A8-A6-AP86+AN86=0$$

$$A8-A7-AP87+AN87=0$$

$$A8-A9-AP89+AN89=0$$

$$A9-A1-AP91+AN91=0$$

$$A9-A2-AP92+AN92=0$$

$$A9-A3-AP93+AN93=0$$

$$A9-A4-AP94+AN94=0$$

$$A9-A5-AP95+AN95=0$$

$$A9-A6-AP96+AN96=0$$

$$A9-A7-AP97+AN97=0$$

$$A9-A8-AP98+AN98=0$$

$$B1-B2-BP12+BN12=0$$

$$B1-B3-BP13+BN13=0$$

$$B1-B4-BP14+BN14=0$$

$$B1-B5-BP15+BN15=0$$

$$B1-B6-BP16+BN16=0$$

$$B1-B7-BP17+BN17=0$$

$$B1-B8-BP18+BN18=0$$

$$B1-B9-BP19+BN19=0$$

$$B2-B1-BP21+BN21=0$$

$$B2-B3-BP23+BN23=0$$

$$B2-B4-BP24+BN24=0$$

$$B2-B5-BP25+BN25=0$$

$$B2-B6-BP26+BN26=0$$

$$B2-B7-BP27+BN27=0$$

$$B2-B8-BP28+BN28=0$$

$$B2-B9-BP29+BN29=0$$

$$B3-B1-BP31+BN31=0$$

$$B3-B2-BP32+BN32=0$$

$$B3-B4-BP34+BN34=0$$

$$B3-B5-BP35+BN35=0$$

$$B3-B6-BP36+BN36=0$$

$$B3-B7-BP37+BN37=0$$

$$B3-B8-BP38+BN38=0$$

$$B3-B9-BP39+BN39=0$$

$$B4-B1-BP41+BN41=0$$

$$B4-B2-BP42+BN42=0$$

$$B4-B3-BP43+BN43=0$$

$$B4-B5-BP45+BN45=0$$

$$B4-B6-BP46+BN46=0$$

$$B4-B7-BP47+BN47=0$$

$$B4-B8-BP48+BN48=0$$

$$B4-B9-BP49+BN49=0$$

$$B5-B1-BP51+BN51=0$$

$$B5-B2-BP52+BN52=0$$

$$B5-B3-BP53+BN53=0$$

$$B5-B4-BP54+BN54=0$$

$$B5-B6-BP56+BN56=0$$

$$B5-B7-BP57+BN57=0$$

$$B5-B8-BP58+BN58=0$$

$$B5-B9-BP59+BN59=0$$

$$B6-B1-BP61+BN61=0$$

$$B6-B2-BP62+BN62=0$$

$$B6-B3-BP63+BN63=0$$

$$B6-B4-BP64+BN64=0$$

$$B6-B5-BP65+BN65=0$$

B6-B7-BP67+BN67=0  
B6-B8-BP68+BN68=0  
B6-B9-BP69+BN69=0  
B7-B1-BP71+BN71=0  
B7-B2-BP72+BN72=0  
B7-B3-BP73+BN73=0  
B7-B4-BP74+BN74=0  
B7-B5-BP75+BN75=0  
B7-B6-BP76+BN76=0  
B7-B8-BP78+BN78=0  
B7-B9-BP79+BN79=0  
B8-B1-BP81+BN81=0  
B8-B2-BP82+BN82=0  
B8-B3-BP83+BN83=0  
B8-B4-BP84+BN84=0  
B8-B5-BP85+BN85=0  
B8-B6-BP86+BN86=0  
B8-B7-BP87+BN87=0  
B8-B9-BP89+BN89=0  
B9-B1-BP91+BN91=0  
B9-B2-BP92+BN92=0  
B9-B3-BP93+BN93=0  
B9-B4-BP94+BN94=0  
B9-B5-BP95+BN95=0  
B9-B6-BP96+BN96=0  
B9-B7-BP97+BN97=0  
B9-B8-BP98+BN98=0  
XR1-XL2+150ZD21<=150  
XR1-XL3+150ZD31<=150  
XR1-XL4+150ZD41<=150  
XR1-XL5+150ZD51<=150  
XR1-XL6+150ZD61<=150  
XR1-XL7+150ZD71<=150

XR1-XL8+150ZD81<=150  
XR1-XL9+150ZD91<=150  
XR2-XL1+150ZD12<=150  
XR2-XL3+150ZD32<=150  
XR2-XL4+150ZD42<=150  
XR2-XL5+150ZD52<=150  
XR2-XL6+150ZD62<=150  
XR2-XL7+150ZD72<=150  
XR2-XL8+150ZD82<=150  
XR2-XL9+150ZD92<=150  
XR3-XL1+150ZD13<=150  
XR3-XL2+150ZD23<=150  
XR3-XL4+150ZD43<=150  
XR3-XL5+150ZD53<=150  
XR3-XL6+150ZD63<=150  
XR3-XL7+150ZD73<=150  
XR3-XL8+150ZD83<=150  
XR3-XL9+150ZD93<=150  
XR4-XL1+150ZD14<=150  
XR4-XL2+150ZD24<=150  
XR4-XL3+150ZD34<=150  
XR4-XL5+150ZD54<=150  
XR4-XL6+150ZD64<=150  
XR4-XL7+150ZD74<=150  
XR4-XL8+150ZD84<=150  
XR4-XL9+150ZD94<=150  
XR5-XL1+150ZD15<=150  
XR5-XL2+150ZD25<=150  
XR5-XL3+150ZD35<=150  
XR5-XL4+150ZD45<=150  
XR5-XL6+150ZD65<=150  
XR5-XL7+150ZD75<=150  
XR5-XL8+150ZD85<=150

XR5-XL9+150ZD95<=150  
XR6-XL1+150ZD16<=150  
XR6-XL2+150ZD26<=150  
XR6-XL3+150ZD36<=150  
XR6-XL4+150ZD46<=150  
XR6-XL5+150ZD56<=150  
XR6-XL7+150ZD76<=150  
XR6-XL8+150ZD86<=150  
XR6-XL9+150ZD96<=150  
XR7-XL1+150ZD17<=150  
XR7-XL2+150ZD27<=150  
XR7-XL3+150ZD37<=150  
XR7-XL4+150ZD47<=150  
XR7-XL5+150ZD57<=150  
XR7-XL6+150ZD67<=150  
XR7-XL8+150ZD87<=150  
XR7-XL9+150ZD97<=150  
XR8-XL1+150ZD18<=150  
XR8-XL2+150ZD28<=150  
XR8-XL3+150ZD38<=150  
XR8-XL4+150ZD48<=150  
XR8-XL5+150ZD58<=150  
XR8-XL6+150ZD68<=150  
XR8-XL7+150ZD78<=150  
XR8-XL9+150ZD98<=150  
XR9-XL1+150ZD19<=150  
XR9-XL2+150ZD29<=150  
XR9-XL3+150ZD39<=150  
XR9-XL4+150ZD49<=150  
XR9-XL5+150ZD59<=150  
XR9-XL6+150ZD69<=150  
XR9-XL7+150ZD79<=150  
XR9-XL8+150ZD89<=150

YU1-YL2+150ZB21<=150  
YU1-YL3+150ZB31<=150  
YU1-YL4+150ZB41<=150  
YU1-YL5+150ZB51<=150  
YU1-YL6+150ZB61<=150  
YU1-YL7+150ZB71<=150  
YU1-YL8+150ZB81<=150  
YU1-YL9+150ZB91<=150  
YU2-YL1+150ZB12<=150  
YU2-YL3+150ZB32<=150  
YU2-YL4+150ZB42<=150  
YU2-YL5+150ZB52<=150  
YU2-YL6+150ZB62<=150  
YU2-YL7+150ZB72<=150  
YU2-YL8+150ZB82<=150  
YU2-YL9+150ZB92<=150  
YU3-YL1+150ZB13<=150  
YU3-YL2+150ZB23<=150  
YU3-YL4+150ZB43<=150  
YU3-YL5+150ZB53<=150  
YU3-YL6+150ZB63<=150  
YU3-YL7+150ZB73<=150  
YU3-YL8+150ZB83<=150  
YU3-YL9+150ZB93<=150  
YU4-YL1+150ZB14<=150  
YU4-YL2+150ZB24<=150  
YU4-YL3+150ZB34<=150  
YU4-YL5+150ZB54<=150  
YU4-YL6+150ZB64<=150  
YU4-YL7+150ZB74<=150  
YU4-YL8+150ZB84<=150  
YU4-YL9+150ZB94<=150  
YU5-YL1+150ZB15<=150

YU5-YL2+150ZB25<=150  
YU5-YL3+150ZB35<=150  
YU5-YL4+150ZB45<=150  
YU5-YL6+150ZB65<=150  
YU5-YL7+150ZB75<=150  
YU5-YL8+150ZB85<=150  
YU5-YL9+150ZB95<=150  
YU6-YL1+150ZB16<=150  
YU6-YL2+150ZB26<=150  
YU6-YL3+150ZB36<=150  
YU6-YL4+150ZB46<=150  
YU6-YL5+150ZB56<=150  
YU6-YL7+150ZB76<=150  
YU6-YL8+150ZB86<=150  
YU6-YL9+150ZB96<=150  
YU7-YL1+150ZB17<=150  
YU7-YL2+150ZB27<=150  
YU7-YL3+150ZB37<=150  
YU7-YL4+150ZB47<=150  
YU7-YL5+150ZB57<=150  
YU7-YL6+150ZB67<=150  
YU7-YL8+150ZB87<=150  
YU7-YL9+150ZB97<=150  
YU8-YL1+150ZB18<=150  
YU8-YL2+150ZB28<=150  
YU8-YL3+150ZB38<=150  
YU8-YL4+150ZB48<=150  
YU8-YL5+150ZB58<=150  
YU8-YL6+150ZB68<=150  
YU8-YL7+150ZB78<=150  
YU8-YL9+150ZB98<=150  
YU9-YL1+150ZB19<=150  
YU9-YL2+150ZB29<=150

YU9-YL3+150ZB39<=150  
YU9-YL4+150ZB49<=150  
YU9-YL5+150ZB59<=150  
YU9-YL6+150ZB69<=150  
YU9-YL7+150ZB79<=150  
YU9-YL8+150ZB89<=150  
ZD12+ZD21+ZB12+ZB21>=1  
ZD13+ZD31+ZB13+ZB31>=1  
ZD14+ZD41+ZB14+ZB41>=1  
ZD15+ZD51+ZB15+ZB51>=1  
ZD16+ZD61+ZB16+ZB61>=1  
ZD17+ZD71+ZB17+ZB71>=1  
ZD18+ZD81+ZB18+ZB81>=1  
ZD19+ZD91+ZB19+ZB91>=1  
ZD23+ZD32+ZB23+ZB32>=1  
ZD24+ZD42+ZB24+ZB42>=1  
ZD25+ZD52+ZB25+ZB52>=1  
ZD26+ZD62+ZB26+ZB62>=1  
ZD27+ZD72+ZB27+ZB72>=1  
ZD28+ZD82+ZB28+ZB82>=1  
ZD29+ZD92+ZB29+ZB92>=1  
ZD34+ZD43+ZB34+ZB43>=1  
ZD35+ZD53+ZB35+ZB53>=1  
ZD36+ZD63+ZB36+ZB63>=1  
ZD37+ZD73+ZB37+ZB73>=1  
ZD38+ZD83+ZB38+ZB83>=1  
ZD39+ZD93+ZB39+ZB93>=1  
ZD45+ZD54+ZB45+ZB54>=1  
ZD46+ZD64+ZB46+ZB64>=1  
ZD47+ZD74+ZB47+ZB74>=1  
ZD48+ZD84+ZB48+ZB84>=1  
ZD49+ZD94+ZB49+ZB94>=1  
ZD56+ZD65+ZB56+ZB65>=1

$$ZD57+ZD75+ZB57+ZB75 \geq 1$$

$$ZD58+ZD85+ZB58+ZB85 \geq 1$$

$$ZD59+ZD95+ZB59+ZB95 \geq 1$$

$$ZD67+ZD76+ZB67+ZB76 \geq 1$$

$$ZD68+ZD86+ZB68+ZB86 \geq 1$$

$$ZD69+ZD96+ZB69+ZB96 \geq 1$$

$$ZD78+ZD87+ZB78+ZB87 \geq 1$$

$$ZD79+ZD97+ZB79+ZB97 \geq 1$$

$$ZD89+ZD98+ZB89+ZB98 \geq 1$$

$$ZD12+ZD21 \leq 1$$

$$ZD13+ZD31 \leq 1$$

$$ZD14+ZD41 \leq 1$$

$$ZD15+ZD51 \leq 1$$

$$ZD16+ZD61 \leq 1$$

$$ZD17+ZD71 \leq 1$$

$$ZD18+ZD81 \leq 1$$

$$ZD19+ZD91 \leq 1$$

$$ZD23+ZD32 \leq 1$$

$$ZD24+ZD42 \leq 1$$

$$ZD25+ZD52 \leq 1$$

$$ZD26+ZD62 \leq 1$$

$$ZD27+ZD72 \leq 1$$

$$ZD28+ZD82 \leq 1$$

$$ZD29+ZD92 \leq 1$$

$$ZD34+ZD43 \leq 1$$

$$ZD35+ZD53 \leq 1$$

$$ZD36+ZD63 \leq 1$$

$$ZD37+ZD73 \leq 1$$

$$ZD38+ZD83 \leq 1$$

$$ZD39+ZD93 \leq 1$$

$$ZD45+ZD54 \leq 1$$

$$ZD46+ZD64 \leq 1$$

$$ZD47+ZD74 \leq 1$$

ZD48+ZD84<=1  
ZD49+ZD94<=1  
ZD56+ZD65<=1  
ZD57+ZD75<=1  
ZD58+ZD85<=1  
ZD59+ZD95<=1  
ZD67+ZD76<=1  
ZD68+ZD86<=1  
ZD69+ZD96<=1  
ZD78+ZD87<=1  
ZD79+ZD97<=1  
ZD89+ZD98<=1  
ZB12+ZB21<=1  
ZB13+ZB31<=1  
ZB14+ZB41<=1  
ZB15+ZB51<=1  
ZB16+ZB61<=1  
ZB17+ZB71<=1  
ZB18+ZB81<=1  
ZB19+ZB91<=1  
ZB23+ZB32<=1  
ZB24+ZB42<=1  
ZB25+ZB52<=1  
ZB26+ZB62<=1  
ZB27+ZB72<=1  
ZB28+ZB82<=1  
ZB29+ZB92<=1  
ZB34+ZB43<=1  
ZB35+ZB53<=1  
ZB36+ZB63<=1  
ZB37+ZB73<=1  
ZB38+ZB83<=1  
ZB39+ZB93<=1

$ZB45+ZB54 \leq 1$  $ZB46+ZB64 \leq 1$  $ZB47+ZB74 \leq 1$  $ZB48+ZB84 \leq 1$  $ZB49+ZB94 \leq 1$  $ZB56+ZB65 \leq 1$  $ZB57+ZB75 \leq 1$  $ZB58+ZB85 \leq 1$  $ZB59+ZB95 \leq 1$  $ZB67+ZB76 \leq 1$  $ZB68+ZB86 \leq 1$  $ZB69+ZB96 \leq 1$  $ZB78+ZB87 \leq 1$  $ZB79+ZB97 \leq 1$  $ZB89+ZB98 \leq 1$ 

END

integer ZD12

integer ZD13

integer ZD14

integer ZD15

integer ZD16

integer ZD17

integer ZD18

integer ZD19

integer ZD21

integer ZD23

integer ZD24

integer ZD25

integer ZD26

integer ZD27

integer ZD28

integer ZD29

integer ZD31

integer ZD32  
integer ZD34  
integer ZD35  
integer ZD36  
integer ZD37  
integer ZD38  
integer ZD39  
integer ZD41  
integer ZD42  
integer ZD43  
integer ZD45  
integer ZD46  
integer ZD47  
integer ZD48  
integer ZD49  
integer ZD51  
integer ZD52  
integer ZD53  
integer ZD54  
integer ZD56  
integer ZD57  
integer ZD58  
integer ZD59  
integer ZD61  
integer ZD62  
integer ZD63  
integer ZD64  
integer ZD65  
integer ZD67  
integer ZD68  
integer ZD69  
integer ZD71  
integer ZD72

integer ZD73  
integer ZD74  
integer ZD75  
integer ZD76  
integer ZD78  
integer ZD79  
integer ZD81  
integer ZD82  
integer ZD83  
integer ZD84  
integer ZD85  
integer ZD86  
integer ZD87  
integer ZD89  
integer ZD91  
integer ZD92  
integer ZD93  
integer ZD94  
integer ZD95  
integer ZD96  
integer ZD97  
integer ZD98  
integer ZB12  
integer ZB13  
integer ZB14  
integer ZB15  
integer ZB16  
integer ZB17  
integer ZB18  
integer ZB19  
integer ZB21  
integer ZB23  
integer ZB24

integer ZB25  
integer ZB26  
integer ZB27  
integer ZB28  
integer ZB29  
integer ZB31  
integer ZB32  
integer ZB34  
integer ZB35  
integer ZB36  
integer ZB37  
integer ZB38  
integer ZB39  
integer ZB41  
integer ZB42  
integer ZB43  
integer ZB45  
integer ZB46  
integer ZB47  
integer ZB48  
integer ZB49  
integer ZB51  
integer ZB52  
integer ZB53  
integer ZB54  
integer ZB56  
integer ZB57  
integer ZB58  
integer ZB59  
integer ZB61  
integer ZB62  
integer ZB63  
integer ZB64

integer ZB65  
integer ZB67  
integer ZB68  
integer ZB69  
integer ZB71  
integer ZB72  
integer ZB73  
integer ZB74  
integer ZB75  
integer ZB76  
integer ZB78  
integer ZB79  
integer ZB81  
integer ZB82  
integer ZB83  
integer ZB84  
integer ZB85  
integer ZB86  
integer ZB87  
integer ZB89  
integer ZB91  
integer ZB92  
integer ZB93  
integer ZB94  
integer ZB95  
integer ZB96  
integer ZB97  
integer ZB98

## KAYNAKLAR

APPLE, J. M., 1977, Plant Layout and Material Handling, 3rd Edition, John Wiley, New York.

BUFFA, E. S., ARMOUR, G. C., VOLLMAN, T. E., 1964, Allocatin Facilities with CRAFT, Harward Business Review, V. 42, No 2, S. 136-159, March/April.

BUFFA, E.S., ,1975, Basic Production Management, John Wiley.

DEMİR, H., 1987, Fabrika Yerleşim Düzeni, D.E.Ü. Yayınları, Kavram Matbaası.

DONAGHEY, C. E., PIRE, V. F., 1990, Solving the Facility LayoutProblem with BLOCPLAN, Industrial Engineering Department, University of Houston, TX.

FRANCIS, R., WHITE, J. , 1974, Facility Layout and Location, Prentice Hall Inc., New Jersey, USA.

HERAGU, S.S., KUSIAK, A., 1990, Machine Layout: An Optimization and Knowledge Based Approach, Int. J. Prod. Research, Vol. 27, No: 4.

HERAGU, S.S., KUSIAK, A, 1991., Efficient Models for the Facility Layout Problem, European J. of Operational Research, Vol. 53, S. 1.

HERAGU, S.S., KUSIAK, A., 1988, Machine Layout: In Flexible Manufacturing Systems Operations Research, Vol. 36, S. 258-268.

HILLIER, F. S. , LIEBERMAN, G.J., 1995, Introduction to Operations Research, McGraw-Hill Books.

KOUVELIS, P., KIM, M.W., Undirectional loop Network Layout Problem in Automated Manufacturing Systems, Operations Research, Vol. 40, S. 533-550, 1990.

KRICK, E. V., 1965, An Introduction to Engineering and Engineering Design, John Wiley and Sons Inc., New York.

KUSIAK, A, HERAGU, S.S. , 1987, The Facility Layout Problem, European J. of Operational Research, Vol. 29, S. 229-251.

LEVARY, R.R., KALCHIK, S. , 1985, Facilities Layout A Survey of Solution Procedures, Computer and Ind. Eng., Vol. 92, S. 141-148.

MONTREUIL, 1990, A Modeling Framework for Integrating Layout Design and Flow Network Design, Proceedings of Material Handling Research Colloquium, Hebron, KY, S. 43-58.

MUTHER, R. , 1955, Practical Plant Layout, McGraw Hill Book Company Inc. USA.

MUTHER, R. , 1976, Systematic Layout Planning, Second Edition, Cahners Books, Massachusettes, USA.

REED, R. Jr., 1961, Plant Layout: Factors, Principles and Techniques, Richard D. Irwin II.

SEEHOF, J.M., EVANS, O., 1967, Automated Layout Design Program, Journal of Industrial Engineering, Vol. 18, No. 12, S. 690-695

SEPPANEN, J.J., MOORE, J.M., Facilities Planning with Graph Theory, Management Science, Vol. 17, S. 242-253, 1970.

TAM, K. Y., 1991, A Simulated Annealing Alghorhytm For Allocating Space To Manufacturing Cells, International Journal of Production Research, Vol. 30, S. 63-87.

URBAN, T. L., Heuristic For The Dynamic Facility Layout Problem, IIE Transactions, Vol. 25, S. 57-63, 1993.

VOLLMAN, T. E., BUFFA, E. S., The Facilities Layout Problem in Perspective, Management Science, Vol. 12, No. 10, S. 450-468, 1966.

