

127093

**AKIŞ TİPİ ÇİZELGELEME PROBLEMLERİNİN GENETİK
ALGORİTMA (GA) İLE ÇÖZÜM PERFORMANSININ
ARTIRILMASINDA DENEY TASARIMI UYGULAMASI**

YÜKSEK LİSANS TEZİ
Müh. Hakan EREN
(507991071)

TC. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ

Tezin Enstitüye Verildiği Tarih : 2 Ocak 2002
Tezin Savunulduğu Tarih : 16 Ocak 2002

Tez Danışmanı : Doç.Dr. Alpaslan FIĞLALI *Alpaslan* 15.2.2002
Diğer Jüri Üyeleri Doç.Dr. Mehmet TANYAŞ (İ.T.Ü.) *Mehmet*
Y.Doç.Dr. Tijen ERTAY (İ.T.Ü.) *Tijen*

OCAK 2002

127093

ÖNSÖZ

Bu çalışma sayesinde akış tipi çizelgeleme, genetik algoritma ve deney tasarımı konularını tanıma fırsatını buldum. Çalışmada geçen teorik kısımların ve uygulamaların konuya ilgi duyan kişiler için yararlı bir kaynak olacağına inanıyorum.

Çalışmalarım sırasında özellikle deney tasarımı ve analizleri konusunu bilimsel yöntemlerle inceledim. Deney tasarım yöntemleri yeni ürün tasarımı, üretim prosesleri geliştirilmesi ve iyileştirilmesi gibi mühendislik alanları başta olmak üzere birçok disiplinde kendine uygulama alanı bulmuştur. Deney tasarımı ürünün imal edilebilirliğinin araştırılmasında ve/veya varolan ürünün performansının geliştirilmesinde ve güvenilirliğinin devamlılığının sağlanmasında, ürün maliyetinin aşağı çekilmesinde ve ürün geliştirme sürecinin kısaltılmasında etkin rol oynar. Aslında deneyler bilimsel prosesin bir parçası olarak görülebilir. Genelde, tahminler yapılan proses hakkında yapılan aktiviteler süresince öğrenilir, prostesten veri elde edebilmek için deney yapılır, yeni tahminler oluşturmak için deneyden elde edilen bilgiler kullanılır, tekrar bu tahminleri sınamak için deneyler yapılır ve böylece devam eder. Bu kadar önemli bir konu olan deney tasarımının maalesef ülkemizde hakettiği ilgiyi bulamadığını düşünüyorum.

Yüksek lisans tezimi hazırlarken benden yardımlarını esirgemeyen Sayın Doç. Dr. Alpaslan Fırlalı'ya ve bugünlere gelmemi sağlayan aileme teşekkür ederim.

Ocak, 2002

Hakan EREN

İÇİNDEKİLER

TABLO LİSTESİ	vi
ŞEKİL LİSTESİ	vii
ÖZET	viii
SUMMARY	ix
1. GİRİŞ	1
1.1. Giriş ve Çalışmanın Amacı	1
2. GENETİK ALGORİTMA	4
2.1. Giriş	4
2.2. Parametre Kodlama	6
2.2.1. Operasyona dayalı kodlama	9
2.2.2. İşe dayalı kodlama	9
2.2.3. Liste önceliğine dayalı kodlama	9
2.2.4. İş çifti ilişkisine dayalı kodlama	10
2.2.5. Öncelik kuralına dayalı kodlama	10
2.2.6. Tamamlanma zamanına dayalı kodlama	11
2.2.7. Makineye dayalı kodlama	11
2.2.8. Rassal anahtarlı kodlama	12
2.2.9. Karışık kodlama	12
2.3. Üreme Operatörü	12
2.4. Çaprazlama Operatörü	13
2.4.1. Pozisyona dayalı çaprazlama (PBX)	14
2.4.2. Sıraya dayalı çaprazlama (OBX)	14
2.4.3. Kısmi planlı çaprazlama (PMX)	15
2.4.4. Dairesel çaprazlama (CX)	16
2.4.5. Doğrusal sıralı çaprazlama (LOX)	16
2.4.6. Sıralı çaprazlama (OX)	16
2.4.7. Alt değişimli çaprazlama (SXX)	17
2.4.8. İş tabanlı sıralı çaprazlama	17
2.5. Mutasyon Operatörü	18
3. AKIŞ TİPİ ÇİZELGELEME PROBLEMLERİ	20
3.1. Giriş	20
3.2. Optimal Çözüm Koşulları Analizi	21
3.3. Genetik Algoritma Modelinin Kurulması	21
3.3.1. Kodlama	21

3.3.2. Başlangıç popülasyonu oluşturma	22
3.3.3. Üreme operatörü	22
3.3.4. Çaprazlama operatörü	23
3.3.5. Mutasyon operatörü	23
4. DENEY TASARIMI	24
4.1. Deney Kavramı	25
4.2. Deney Tasarımının Amaçları	25
4.3. Deney Tasarımlarının Tipik Uygulamaları	26
4.4. Deney Tasarım Yöntemlerinin Ortak Dili	27
4.5. Temel Prensipler	29
4.6. Deney Tasarım Aşamaları	30
4.7. Deney Tasarımının Tarihçesi	33
5. DENEY TASARIM YÖNTEMLERİNİN İNCELENMESİ	35
5.1. Deney Tasarım Yöntemleri	35
5.1.1. Klasik yöntemler	35
5.1.1.1. Bir kerede bir faktör	35
5.1.1.2. Tam faktöriyel	37
5.1.1.3. Kesirli faktöriyel	41
5.1.2. Taguchi yöntemi	42
5.1.3. Shainin yöntemi	47
5.1.4. Genetik algoritma yöntemi	48
6. İKİ SEVİYELİ KESİRLİ FAKTÖRİYEL DENEYLER VE TAGUCHİ DİZAYNLARI	49
6.1. Giriş	49
6.2. İki Seviyeli Üç Faktörlü L8 Dizaynı	49
6.3. İki Seviyeli Dört Faktörlü L16 Dizaynı	50
6.4. Kesirli Deneylerin Tasarımı ve Eşad Kavramı	51
6.5. Deney Çözünürlüğü	53
6.6. Dizayn Jeneratörünün Seçimi	55
6.7. Yarım Kesirleri Oluşturma	55
6.8. Kesirli Faktöriyellere Dönüştürme	56
6.9. Genel 2^{k-p} Kesirli Faktöriyel Tasarımlar	56
6.10. 2^{k-p} Kesirli Faktöriyellerin Analizi	59
6.11. Kesirli Faktöriyelleri Bloklama	59
6.12. Deneyde Kullanılan L8 Hesap Tablosu	61
6.13. Normal Olasılık Grafiği	63
6.14. Deneyde Kullanılan Hesap Yönteminin Özeti	64
6.15. İki Katlı Dizayn	65
6.16. Normal Olasılık Grafiğinin Kuramsal Temeli	67
6.17. Deney Tasarımı İçin Temel Basamaklar	67
6.18. Sonuç Değişkeninin Hedef Değeri	69
7. DENEY UYGULAMASI – 1	70
7.1. Giriş	70

7.2. Deney Notasyonu	70
7.3. Problemin Tanımı	70
7.3.1. Sonuç değişkeninin belirlenmesi	70
7.3.2. Faktörlerin ve seviyelerinin belirlenmesi	70
7.4. Deneyin Tasarımı	71
7.5. Deneyin Yapılışı	72
7.6. Veri Analizi	73
7.7. Deney Sonucu	75
8. DENEY UYGULAMASI – 2	76
8.1. Giriş	76
8.2. Sonuç Değişkeninin Belirlenmesi	76
8.3. Faktörlerin ve Seviyelerinin Belirlenmesi	76
8.4. Deneyin Tasarımı	77
8.5. Deneyin Yapılışı	77
8.6. Veri Analizi	78
8.7. Faktör Etkileşimleri	80
8.8. Deney Sonucu	81
9. DENEY UYGULAMASI – 3	82
9.1. Giriş	82
9.2. Sonuç Değişkeninin Belirlenmesi	82
9.3. Faktörlerin ve Seviyelerinin Belirlenmesi	82
9.4. Deneyin Tasarımı	83
9.5. Deneyin Yapılışı	83
9.6. Veri Analizi	84
9.7. Deney Sonucu	86
10. SONUÇLAR	87
KAYNAKLAR	88
EKLER	89
ÖZGEÇMİŞ	105

TABLO LİSTESİ

	<u>Sayfa No</u>
Tablo 2.1. Seçilmiş öncelik kuralları	11
Tablo 4.1. Deney tasarım aşamaları	31
Tablo 5.1. 2^7 tam faktöriyel deney için etkileşim dağılımları	39
Tablo 5.2. İki seviyeli iki faktör deney tasarım matrisi	40
Tablo 5.3. 2^3 tam faktöriyel tasarım matrisi	46
Tablo 5.4. Taguchi'nin L8 dizayn matrisi	46
Tablo 6.1. L8 dizayn matrisi	50
Tablo 6.2. L16 dizayn matrisi	51
Tablo 6.3. L8 kesirli deneylerde faktörlerin sütunlara atanması ve eşadlar ..	52
Tablo 6.4. L16 matrisinde faktörlerin sütunlara atanması	54
Tablo 6.5. 2^3 tasarım için iki yarım kesir	55
Tablo 6.6. Seçilmiş 2^{k-p} kesirli faktöriyel tasarımlar	57
Tablo 6.7. Seçilmiş 2^{k-p} faktöriyel tasarımlar için bloklar	60
Tablo 6.8. L8 hesap tablosu	62
Tablo 6.9. İki katlı dizayn matrisi	65
Tablo 6.10. İki katlı dizayn etki hesap tablosu	66
Tablo 6.11. İki katlı dizayn etki hesap tablosunda faktörlerin ana etki ve etkileşim etkisi sütunlarına atanması	66
Tablo 7.1. Kontrol edilebilir faktörler ve seviyeleri	71
Tablo 7.2. Deneyde kullanılan L8 dizayn matrisi	72
Tablo 7.3. Eşad tablosu	72
Tablo 7.4. Deney verileri	73
Tablo 7.5. L8 etki hesap tablosu	74
Tablo 8.1. Kontrol edilebilir faktörler ve seviyeleri	76
Tablo 8.2. Deneyde kullanılan L8 dizayn matrisi	77
Tablo 8.3. Deney verileri	78
Tablo 8.4. L8 etki hesap tablosu	79
Tablo 8.5. AC etkileşimine ait ortalamalar	80
Tablo 9.1. Kontrol edilebilir faktörler ve seviyeleri	82
Tablo 9.2. Deneyde kullanılan L8 dizayn matrisi	83
Tablo 9.3. Deney verileri	84
Tablo 9.4. L8 etki hesap tablosu	85
Tablo 10.1. Uygun parametre seti	87

ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1 : Kod uzayı ve çözüm uzayı	6
Şekil 2.2 : Kodlamada feasibility ve legality	7
Şekil 2.3 : Pozisyona dayalı çaprazlama	14
Şekil 2.4 : Sıraya dayalı çaprazlama	15
Şekil 2.5 : Komşu iki iş değiştirme	18
Şekil 2.6 : Keyfi iki iş değiştirme	19
Şekil 2.7 : Keyfi üç iş değiştirme	19
Şekil 2.8 : Shift değiştirme	19
Şekil 4.1 : Bir proses veya sistemin genel modeli	26
Şekil 5.1 : Bir kerede bir faktör değiştirilerek yapılan deneylerin tasarım yöntemi akış şeması	36
Şekil 6.1 : ABD ile örtüşen iki bloklü 2^{6-2} tasarımı	60
Şekil 6.2 : Normal olasılık grafik kağıdı	63
Şekil 7.1 : Normal olasılık grafiği	73
Şekil 8.1 : Normal olasılık grafiği	78
Şekil 8.2 : AC etkileşimi	80
Şekil 9.1 : Normal olasılık grafiği	84

ÖZET

Akış tipi çizelgeleme iyi bilinen zor bir kombinatoriyel optimizasyon problemidir. Son on yılda üzerinde en çok çalışılan konulardan biri olup sezgisel yöntemler ile bulunan çözümlerin genetik algoritmalar ile iyileştirilmesine çalışılmaktadır. Genetik algoritmanın akış tipi çizelgeleme problemlerinde (n iş, m makine için) bilinen sezgisel yöntemlerden daha iyi çözüm verdiği belirlenmiştir.

Bu çalışma, NP olarak bilinen akış tipi çizelgeleme problemlerinin genetik algoritma ile çözüm performansının artırılmasında deney tasarımı ile ilgilidir. Deney tasarımı ve özellikle iki seviyeli kesirli faktöriyel deneyler ile Taguchi'nin L8 ve L16 dizaynları üzerine yoğunlaşmıştır. Genetik algoritmanın optimum veya optimuma yakın çözüme ulaşma performansını etkileyen; başlangıç popülasyonu, üreme, çaprazlama ve mutasyon operatörleri ile çaprazlama ve mutasyon oranlarının belirlenmesine çalışılmıştır. Parametrelerin ayrı ayrı değerlendirilmesi ile bulunan en iyi iki parametre değeri, çok makine problemleri için iki seviyeli, altı faktörlü olarak deney tasarımı tabi tutulmuştur. Daha sonra elde edilen veriler kullanılarak her faktörün en iyi seviyeleri belirlenmiştir. Sayısal seviyelere sahip faktörler için elde edilen sonuçlar doğrultusunda seviye aralıkları daraltılarak daha hassas sonuçlar elde etmek üzere iki deney tasarımı daha gerçekleştirilmiştir. Deneyler Turbo Paskal programlama dilinde hazırlanan genetik algoritma programı ile yapılmıştır. Gerçekleştirilen üç deney tasarımı toplam 24 adet deneyden oluşur. Bu deneyler sonucunda genetik algoritmanın çözüm performansını artıracak parametre setleri önerilmeye çalışılmıştır.

SUMMARY

Scheduling jobs in a flow-shop is a well-known hard combinatorial optimization problem. It is one of the most studied topic in the last ten years and the solutions which are found by heuristic methods, are studied to be improved by genetic algorithms. It was determined that genetic algorithms give better solutions than heuristic methods at the problem of scheduling jobs in a flow-shop (for n work, m machine).

This study is about the experimental design in improving solution performance by using genetic algorithm of the problem of scheduling jobs in a flow-shop which is known as NP-complete problem. The design of experiments and especially bi-level fractional factorial experiments and Taguchi's L8 and L16 designs were covered. It was studied the parameters, such as number of initial population, reproduction operators, crossover operators, mutation operators, rate of crossover and mutation operators affected performance of genetic algorithm to achieve optimum or suboptimum solution. The best two parameters value found by evaluating the parameters individually were used in design of experiments as bi-level and six factors for multi-machine problems. Then by using the obtained data, the best levels of every parameter were determined. According to the results, two more experimental designs were realized for the parameters which have numerical levels, by reducing the level intervals, to obtain more sensitive results. Experiments were done with the genetic algorithms program made by using Turbo Pascal programming language. Three experiment designs have totally 24 experiments. As a result of experiments, parameter sets being able to improve the solution performance of genetic algorithm were tried to be proposed.

1. GİRİŞ

1.1. Giriş ve Çalışmanın Amacı

Optimizasyon problemleri, karar değişkenlerinin sürekli ve kesikli olmalarına göre iki kısımdan oluşur. Karar değişkenleri kesikli (süreksiz) olanlara kombinatoriyel optimizasyon problemleri denir.

Kombinatoriyel problemlerde, sonlu veya sayılabilir sonsuz bir kümeden, bir alt kümeye, bir nesneye, bir permütasyona ulaşmaya çalışırız. Kombinatoriyel optimizasyon problemlerinin büyük bir kısmı, sezgisel (heuristic) metotlar yardımı ile çözülebilmektedir. Diğer bir deyişle bu tip problemler için optimum değil, optimuma yakın çözümler elde edilmektedir.

Bir problemin, bilinen algoritmalar ile tatminkar sürede çözümü elde edilebiliyor ise bu problemler P (polinomial) olarak adlandırılır. Bir algoritmanın artış hızı, 2^n , polinomiye olmayan şekilde değişiyor ise (eksponansiyel artış) problem NP (non deterministic polinomial) haline dönüşmektedir. Üstel artış hızları, $n!$, n^2 , $n^{\log n}$ şeklinde olabilir. Polinomiye fonksiyonlar, teknolojik gelişmelere daha iyi uyum sağladıklarından ve kapalılık özelliğinden dolayı (bazı özel problemler birbirleri ile birleştirilebilirler) daha avantajlıdır. NP problemleri için bir polinomiye algoritma henüz geliştirilememiştir [5].

Bu çalışmada, NP olarak bilinen akış tipi çizelgeleme problemlerinin Genetik Algoritma (GA) yardımı ile çözüm performansının artırılmasında deney tasarımı yapılmıştır. Akış tipi çizelgeleme problemlerinde, birbirinden farklı, m-makine ve n-iş mevcut olup her bir iş aynı sıra ile m farklı operasyondan oluşur. Akış tipi çizelgeleme problemleri ile ilgili yapılan bu çalışmada amaç en iyi çözüme ulaşmak için gereken iterasyon sayısını minimize etmektir.

Akış tipi çizelgeleme problemleri ile ilgili ilk çalışmayı, Johnson (1954) yapmıştır. İki makine n-iş problemleri için optimum çözüm veren basit bir algoritma geliştirmiştir.

Daha sonra yapılan çalışmalarda m-makine sayısı ($m>2$) için araştırmalar yapılmıştır. Makine sayısı ($m>2$) problemleri, NP kapsamına girdiğinden bu problemler için çeşitli sezgisel yöntemler geliştirilmiştir. Bunlar; Palmer'in Eğim Dizisi Yöntemi (1965), Compbell, Dudek ve Smith (CDS) Algoritması (1970), Gupta Yöntemi (1971), Dannenbring Yöntemi (1972), Nawaz, Enscore ve Ham (NEH) Yöntemi (1983), Hundal ve Rajgopal Yöntemi (1998), Widmer ve Hertz Yöntemi (1989) ve Ho ve Chang (HC) Yöntemi'dir (1991).

Akış tipi çizelgeleme problemleri NP kapsamında olduğundan, bilgisayarların hızlarının artması ile daha etkin çizelgeleme yöntemleri geliştirilmeye çalışılmıştır. Son yıllarda, çözümü zor olan böyle optimizasyon problemleri için geliştirilen literatürdeki farklı sezgisel arama metotlarından Genetik algoritmalar (GA), Tavlama Benzetimi (SA) ve Tabu Araştırmaları (TS) başarı ile uygulanan üç tanesidir. Genetik algoritmalar biyoloji biliminden, tavlama benzetimi fizik biliminden ve tabu aramada zeka tekniklerinden ortaya çıkmış arama metotlarıdır. Bu yöntemlerden genetik algoritma, rassal arama tekniklerini kullanarak çözüm bulmaya çalışan, parametre kodlama esasına dayanan bir arama tekniğidir. Genetik algoritmaların çizelgeleme yöntemlerinde kullanımı iki farklı şekilde gerçekleştirilmiştir. Bunlar;

1. Genetik algoritma yardımı ile permütasyon tipi iş sırası bulma,
2. Genetik algoritma ile bulunan permütasyon sırasının, bilinen sezgisel yöntemler ile karşılaştırılmasıdır.

NP olarak bilinen akış tipi çizelgeleme problemlerinin GA ile çözüm kalitesi, GA'da kullanılan parametrelere bağlıdır [5]. Genetik algoritmada optimuma yakın çözüm veren parametreler, problemlerin yapısına göre değiştiğinden dolayı akış tipi çizelgeleme problemlerinde optimum veya optimuma yakın çözüme ulaşılması için bu çalışmada genetik operatörlerin belirlenmesine ilişkin deney tasarımı yapılmıştır. Deney tasarımı sayesinde hem operatör etkileşimleri görülebilmektedir hem de sistematik bir yöntem uygulanmış olmaktadır. Deneylerden elde edilen sonuçlara göre parametre setlerinin çözüm kalitesini etkilediği belirlenmiştir.

Deney, bir proses veya sistemin girdi değişkenlerinde yapılan anlamlı değişikliklerin çıktı üzerinde oluşturduğu değişim şeklini ve nedenini belirlemek amacıyla gerçekleştirilen sınama veya sınama gruplarıdır. Deney tasarımı yöntemlerinde sistemi etkileyen tüm faktörler ele alındığı için normalde sistemdeki bir faktörü

düzeltilmekle elde edilebilecek fayda yerine, birkaç faktörde küçük deęişiklikler yaparak toplamda daha çok fayda sağlanabilir. İstatistiksel olarak tasarlanan deneylerde, birçok faktörün ürün parametrelerine olan etkileri ve birbirleriyle oluşan etkileşimleri ekonomik olarak incelemek mümkündür. Deney tasarımı ve analizleri özellikle yeni ürün tasarımı, üretim proseslerinin geliştirilmesi ve süreç iyileştirilmesi gibi mühendislik alanlarında önemli bir rol oynamaktadır. Bu çalışma, NP olarak bilinen akış tipi çizelgeleme problemlerinin genetik algoritma ile çözüm performansının artırılmasında deney tasarımı ile ilgilidir. Deney tasarımı ve özellikle iki seviyeli kesirli faktöriyel deneyler ile Taguchi'nin L8 ve L16 dizaynları üzerine yoğunlaşmıştır.

2. GENETİK ALGORİTMA

2.1. Giriş

Genetik algoritma, ilk olarak 1975 yılında John Holland'ın " Doğal ve yapay sistemlerin uyumu " isimli kitabıyla ortaya çıkmıştır. GA'nın çalışma yöntemi, Darwin'in, " Doğal Seleksiyon " prensibine dayanır. Darwin, " Türlerin Kökeni " adlı yapıtında, iki varsayımı ortaya atmıştır. Bunlar:

1. Tüm organizmalar gereğinden fazla yavru meydana getirme yeteneğine sahiptirler. Bununla beraber elimine edilenler ile popülasyonda denge sağlanmaktadır.
2. Bir tür içerisindeki bireyler, kalıtsal özellikleri bakımından farklıdır.

GA, doğadaki canlıların geçirdiği evrim sürecini dikkate alır. Amaç, doğal sistemlerin uyum sağlama özelliğini dikkate alarak, yapay sistemleri tasarlamaktır. GA da tasarlanan yapay sistemlerde dikkate alınan en önemli faktör sağlamlık (robustness) dır.

Eğer yapay sistemler, doğal sistemler kadar sağlam olabilse, mevcut sistemler faaliyetlerini daha uzun zaman sürdürecekler ve pahalı olan yeniden tasarlama ve uyarlama işlemleri ortadan kalkacaktır.

Genetik algoritmaların gelişimi, John Holland'ın doktora öğrencisi olan David E. Goldberg'in, 1985 yılında hazırlamış olduğu, "Gaz boru hatlarının Genetik algoritmalar kullanılarak denetlenmesi" konusundaki tezi ile sağlanmıştır. Bu ilk uygulamadan sonra Goldberg'in 1989 yılında çıkardığı " Genetic algorithm in search optimization and machine learning ", adlı kitabı, GA'ya yeni bir boyut kazandırmıştır [5].

Genetik algoritma, rassal arama tekniklerini kullanarak çözüm bulmaya çalışan, parametre kodlama esasına dayanan bir arama tekniğidir.

Genetik algoritmayı diğer arama yöntemlerinden farklı kılan özellikler:

1. Genetik algoritma, parametre setlerinin kodları ile ilgilenir, parametrelerin kendileri ile direkt ilgilenmez.
2. Genetik algoritmanın arama alanı, yığının veya popülasyonun tamamıdır; tek nokta veya noktalarda arama yapmaz.
3. Genetik algoritmalarda, amaç fonksiyonu kullanılır, sapma değerleri veya diğer faktörler kullanılmaz.
4. Genetik algoritmaların uygulanmasında kullanılan operatörler, stokastik yöntemlere dayanır, deterministik yöntemler kullanılmaz.

Genetik algoritmalar, her zaman, asıl optimum olmasa da, optimuma yakın çözümleri garanti ederler. Bir problemin GA ile çözümünde takip edilecek işlem adımları:

1. Arama uzayındaki bütün muhtemel çözümler dizi olarak kodlanır. Bu diziyi (kromozomu) oluşturan her bir elemana gen denir. Her bir dizi, arama uzayında belirli bir bölgeye karşılık gelir.
2. Genellikle rassal bir çözüm seti seçilir ve başlangıç popülasyonu olarak kabul edilir.
3. Her bir dizi için bir uygunluk değeri hesaplanır. Bulunan uygunluk değerleri, dizilerin çözüm kalitesini gösterir.
4. Bir grup dizi (kromozom) belirli bir olasılık değerine göre rassal olarak seçilip üreme işlemi gerçekleştirilir.
5. Üreme işleminde, çeşitli genetik operatörler kullanılabilir.
6. Yeni bireylerin uygunluk değerleri hesaplanarak, çaprazlama ve mutasyon işlemlerine tabi tutulur.
7. Önceden belirlenen nesil sayısı boyunca yukarıdaki işlemler devam ettirilir.
8. İterasyon, nesil sayısına ulaşınca işlem bitirilir. Uygunluk değeri en yüksek olan dizi seçilir.

Genetik algoritmalarda kullanılan temel kavramlar aşağıdaki şekilde sıralanabilir [5].

Gen: Kalıtsal molekülde bulunan ve organizmanın karakterlerinin tayininde rol oynayan kalıtsal birimlere denir. Yapay sistemlerde gen, kendi başına anlamlı bilgi

taşıyan en küçük birim olarak alınır. Uygulama problemimizde (iş çizelgeleme) her bir gen bir işi temsil eder.

Kromozom: Birden fazla genin bir araya gelerek oluşturduğu diziye denir. Kromozomlar, alternatif aday çözümleri gösterirler.

Plan – Katar (Schemata): Belirli pozisyonlarda, uygun olan alt dizileri (kromozomları) tanımlamak için kullanılan sayı katarlarına veya planlarına denir.

Örneğin, iki tabanlı (0 ve 1) kodlama sisteminde beş adet genden oluşan bir kromozom dizisinde her bir genin yerine, (0, 1, *) gibi üç farklı kod kullanılabileceğinden, $L = 5$ için, $3 \times 3 \times 3 \times 3 \times 3 = 243$ farklı alt dizi tanımlanabilir.

Genel olarak iki tabanlı kodlama sistemlerinde, plan (katar) sayısı,

$$\text{Plan sayısı} = (k+1)^L \quad \text{dir.} \quad (2.1)$$

L = Dizideki rakam (gen) sayısı

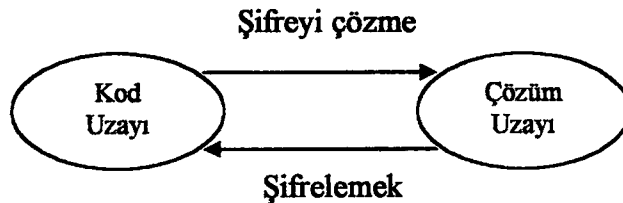
k = Kodlama sistemindeki karakter sayısı (0, 1 gibi)

Populasyon: Kromozomlardan oluşan topluluğa denir. Populasyon geçerli alternatif çözüm kümesidir. Populasyondaki birey sayısı (kromozom) genelde sabit tutulur. GA'da populasyondaki birey sayısı ile ilgili genel bir kural yoktur. Populasyondaki kromozom sayısı arttıkça çözüme ulaşma süresi (iterasyon sayısı) azalır. Problemin büyüklüğüne bağlı olarak populasyon büyüklüğü (10 ile 100) arasında seçilebilir.

2.2. Parametre Kodlama

Genetik algoritmaların temel özelliği, çözüm uzayı ve kod uzayında çalışmasıdır.

Kod ve çözüm uzayı arasındaki etkileşim Şekil 2.1'de verilmiştir.

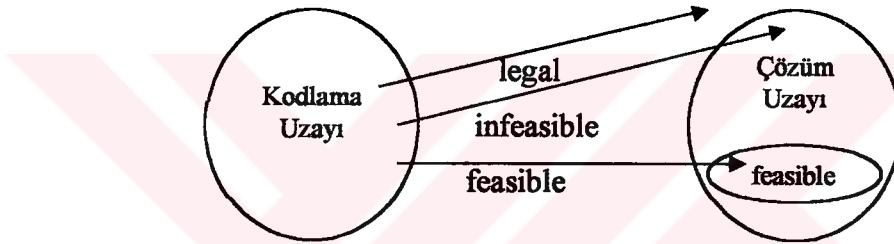


Şekil 2.1. Kod uzayı ve çözüm uzayı

John Holland'ın çalışmalarında genelde iki tabanlı kodlama sistemi kullanılmıştır. Problemlerin başarılı bir şekilde çözümü için en kısa kodlama şekillerinin kullanılması gerekir.

Son on yıl boyunca değişik kodlama teknikleri geliştirilmiştir. Çözümün kromozomlara nasıl kodlanacağı GA'da anahtar niteliğinde bir çalışmadır. Kromozomların kodlanması ve şifre çözümü aşamasında, üç kritik hususa dikkat edilmesi gerekir. Bunlar:

1. Kromozomun uygunluğu (feasibility) kontrol edilir.
2. Kromozomun belirlenen şartları sağlayıp sağlamadığı (legality) kontrol edilir.
3. Kodlama haritasında tek olduğu kontrol edilir. Kromozomun uygunluğu ile şartları sağlayıp sağlamadığı işlemleri Şekil 2.2'de gösterilmiştir.



Şekil 2.2. Kodlamada feasibility ve legality

Şartları sağlamayan (illegal) bir kromozom, çözüm uzayındaki değerleri şifreleyemez, böyle bir kromozom, evrime uğrayamaz.

GA ile kurulacak model, akış tipi çizelge problemi olduğu için bunu konu ile ilgili geliştirilen kodlama yöntemlerine değinilecektir.

İkili kodlama ile beraber kullanılan kodlama yöntemlerinden biri harfli kodlamadır. Bu kodlamada her bir kromozoma bir harf verilir, kromozomdaki genler de ikili tabanlı kodlanırlar, eğer kromozom sayısı 26'dan fazla olursa artan kromozomlar için, 1, 2, 3... gibi rakamlar kullanılır.

Çok parametrelili kodlama da iki kodlama tabanına göre kullanılır. Bu yöntemde uygunluk değeri en küçük olan kromozoma U_{min} , en büyük olana U_{max} , değerleri verilir. $U_1, U_2, \dots, U_n$ olarak kodlanırlar.

Akış tipi çizelgeleme problemlerinde iki tip dizi kodlama yöntemi kullanılır. Bunlar:

1. Zayıf kodlama
2. Genel dizi kodlama

Zayıf dizi kodlamada ayrıştırıcı semboller kullanılır. Genel dizi kodlamada ise birbirinin tekrarı olan (0 – 1) kullanılır. Zayıf kodlamada dizi içerisinde tekrarlı genlere izin verilmez; her kodda en fazla bir tane vardır.

John Holland tarafından geliştirilen ve yaygın olarak kullanılan plan (sayı katarları) son yıllarda çizelgeleme problemlerinde kullanılmamaktadır.

Çizelgeleme problemleri ile ilgili olarak son beş yılda, yukarıdaki yöntemlerden farklı olarak 9 çeşit temsil yöntemi geliştirilmiştir. Bunlar:

1. Operasyona dayalı kodlama
2. İşe dayalı kodlama
3. Liste önceliğine dayalı kodlama
4. İş çifti ilişkisine dayalı kodlama
5. Öncelik kuralına dayalı kodlama
6. Tamamlanma zamanına dayalı kodlama
7. Makineye dayalı kodlama
8. Rassal anahtarlı kodlama
9. Karışık kodlama

İki temel kodlama yaklaşımları içinde yukarıdaki kodlamalar sınıflandırılabilir.

- Direkt yaklaşım
- En direkt yaklaşım

Operasyona dayalı kodlama, işe dayalı kodlama, iş çifti ilişkisine dayalı kodlama, tamamlama zamanına dayalı kodlama ve rassal anahtarlı kodlama direkt yaklaşım sınıfına aittir. Tercih edilen dizine dayalı kodlama ve öncelik kuralına dayalı kodlama ise en direkt yaklaşım sınıfına aittir [5].

2.2.1. Operasyona Dayalı Kodlama

Operasyonların dizilimi olarak bu kodlama bir çizelgeleme şifreler. Her gen bir operasyon için kullanılır. Her operasyonu isimlendirmenin bir yolu doğal sayıları kullanmaktır. GSP (Gezgin Satıcı Problemi) için permütasyon kodlamaya benzer. Maalesef öncelik kısıtlamalarının oluşumundan dolayı uygun çizelgeleri tanımlayan bu sayıların permütasyonları mümkün değildir. m makine ve n iş probleminde kromozomlar $n \times m$ gen içerir. Her iş tam olarak m kere kromozomda görünür ve her gen işin somut operasyonlarını göstermez. Genellikle uygun çizelgeleme sağlayan kromozomların her bir permütasyonunu görmek uygundur. Aşağıdaki kod çözücü prosedürü ile bir kromozomdan bir çizelgelemenin şifresi çözülür [5].

1. İlk olarak kromozomları, operasyonların sıralamasının dizisine göre değiştirir.
2. Daha sonra bir geçişli sezgisel temelli liste tarafından çizelgelemeyi oluşturur. Listedeki ilk operasyon ilk çizelgelenir, daha sonra ikinci operasyon ve diğerleri devam eder. Uygun makine operasyon gerekleri için mevcut en uygun işlem zamanı proses altındaki her operasyona ayrılır. Operasyonlar çizelgelenene kadar işlem tekrar edilir.
3. Prosedür tarafından oluşturulan çizelgeleme aktif çizelgeleme olabilmesini garantileyebilir.

2.2.2. İşe Dayalı Kodlama

Bu kodlama, n iş listesinden oluşur ve işlerin dizilimine göre çizelgeleme oluşturulur. İşlerin dizilimi verildiği için listedeki ilk işin tüm operasyonları ilk çizelgelenir. Sonra listedeki ikinci işin operasyonları düşünülür ve ikinci işin tüm operasyonları çizelgelenir. Uygun dizilimde düşünülen listedeki işlerin her biri ile proses tekrar ettirilir [5].

2.2.3. Liste Önceliğine Dayalı Kodlama

n iş, m makine için çizelgeleme problemi, m alt kromozomdan oluşturulur. Her bir makine için n uzunluğuyla sembolize edilen her alt kromozom bir dizidir.

Bir operasyonu tanımlayan her sembol, ilişkili makine üzerinde işlem gördürmelidir. Alt kromozomlar makine üzerindeki operasyon dizilimlerini göstermez, öncelik

listesinde yer alır. Her makine kendine ait öncelik listesine sahiptir. Makine önünde, kuyruktaki beklemelerinin durumunu analiz etmek için, simülasyonla kromozomlardan asıl çizelge çıkarılır. Eğer çizelge düzenlemek için öncelik listeleri kullanmak gerekli ise, öncelik listesinde ilk görünen operasyon seçilmelidir [5].

2.2.4. İş Çifti İlişisine Dayalı Kodlama

Nakaza ve Yamada bir çizelgeleme kodlamak için ikilik düzendeki matrisi kullanmışlardır. Bu matris, makinelere uygun işlerin çift ilişkisi önceliğine göre düzenlenmiş bir matristir [5].

İş çizelgeleme kodlamaları içinde en kompleks olanı iş çifti ilişkisine dayanan kodlama yöntemidir.

2.2.5. Öncelik Kuralına Dayalı Kodlama

Öncelik kural tabanlı genetik algoritmayı ilk geliştiren Dorndort ve Pesch'dir. n iş makine problemi için $n \times m$ giriş, $(P_1, P_2, \dots, P_m)$ dizilimli bir kromozom seti geliştirilmiştir.

$P_{st} = t$ çizelgelenmiş operasyonları içeren parçalı çizelgeleme

$S_t =$ Verilen P_{st} 'ye uygun t iterasyonunda çizelgeleme operasyonlarının kümesi

$G_i =$ Operasyon $i \in S_t$ 'de başlanabilen en erken zaman

$\phi_i =$ Operasyon $i \in S_t$ 'de tanımlanabilen en erken zaman

$C_t = t$ iterasyonunda karıştırılmış operasyonlar kümesi

Kromozom $(P_1, P_2, \dots, P_m)$ 'den uygun dizi elde etme prosedürü [5]:

Prosedür: Öncelik kuralına dayalı kodlama için dizi oluşturma

Adım 1: $t = 1$ 'de sabit parçalı çizelgeleme olarak P_{st} ile başla. S_t öncelikleri olmayan operasyonları içerir.

Adım 2: $\phi_i^* = \min i \in S_t [\phi_i]$ düzende ve ϕ_i^* üzerinde gerçekleştirilebilen makine m^* eğer örnek bir makine oluşumundan daha fazla ise rassal seçim tarafından düğüm kırılır.

Adım 3: m^* makineyi gerçekleştiren $i \in S_t$ ile $G_i < \phi_i^*$ operasyonlarını içeren karıştırılmış set C_t 'yi oluşturur. Öncelik kural P_t tarafından C_t 'den bir operasyon seç

ve mümkün olduğunca erken bu operasyonu P_{st} 'ye ekle. Bundan dolayı yeni bir parçalı P_{st+1} çizelgeleme oluşturur. Eğer öncelik kuralı P_t 'ye göre birden fazla operasyon oluşturulur ise rassal seçim tarafından bağlantısı koparılır.

Adım 4: S_t 'den seçilmiş operasyonların kaydırmasıyla P_{st+1} 'i güncelleştirir ve S_t operasyonunun direkt olasılığını ekle.

Adım 5: Tam bir çizelgeleme oluşturulana kadar Adım 2'ye dön.

Tablo 2.1. Seçilmiş öncelik kuralları

No	Kural	Tanım
1	SPT	En kısa işlem zamanlı bir operasyon seç
2	LPT	En uzun işlem zamanlı bir operasyon seç
3	MWR	En çok toplam işlem zamanı kalan iş için operasyon seç
4	LWR	En az toplam işlem zamanı kalan iş için operasyon seç

2.2.6. Tamamlanma Zamanına Dayalı Kodlama

Operasyonların tamamlanma zamanlarının sıralanmış listesi bir kromozomdur. Genetik operatörler için, uygun olmayan örnek kodlamayı bilmek kolaydır. Çünkü kodlama illegal bir çizelgeleme oluşturur [5].

2.2.7. Makineye Dayalı Kodlama

Makineye dayalı genetik algoritmayı Dorndorf ve Pesch önermiştir. Makinelerin sırasına göre bir kromozom kodlanır ve sıralamaya dayalı olarak sezgisel darboğazı önleme ile bir çizelgeleme oluşturulur.

Sezgisel darboğazı önleme Adams tarafından önerilmiştir. Atölye çizelgeleme problemi için şu ana kadar tüm sezgisel metotlar içinde bilinen en güçlü yöntemdir. Makinelerin sıralanması birer birer yapılır. Henüz sıraya alınmamış makineler darboğaz olarak kabul edilir. Kromozomlar makine sıralama listesini oluşturur. Genetik algoritma makine sıralarının bulunmasında kullanılır [5].

Henüz sıralanmış makinelerin kümesi m ise kromozom $[m_1, m_2, \dots, m_m]$ olur. Çalışan kromozomdan bir çizelgeleme prosedürü çıkarma:

Prosedür: Makineye dayalı kodlama için bir çizelgeleme oluşturma.

Adım 1: $M_0 \leftarrow \phi$, $i \leftarrow 1$ ve kromozom $[m_1, m_2, \dots, m_m]$

Adım 2: Makine m_i 'yi optimal bir şekilde sırala. Set $M_0 \leftarrow M_0 \cup [M_i]$ 'i güncelleştirir.

Adım 3: Diğer düzenlenmiş sıralamaları muhafaza ederken her kritik makine $m_i \leftarrow M_0$ 'ın dizilimini yeniden optimize et.

Adım 4: $i \leftarrow i+1$ yap. Eğer $i > m$ ise durdur; diğer durumlarda adım 2'ye git.

2.2.8. Rassal Anahtarlı Kodlama

Rassal anahtarlı kodlama, Bean tarafından geliştirilmiştir. Rassal sayı ile beraber rassal anahtar kodlama bir çözümü şifreler. Çözümün şifresini çözmek için sıralı anahtar olarak bu değerler kullanılır. n iş, m makine çizelgeleme problemi için her gen iki parçadan oluşturulur. Set'teki tamsayı $[1, 2, \dots, m]$ ve $(0, 1)$ 'den rassal olarak oluşturulan bir parçasıdır. Rassal anahtarların tamsayı bölümü, bu iş için makine düzenleme olarak yorumlanır. Her makinede iş sıralama kesirli kısımların sınıflandırılmasıyla sağlanır [5].

2.2.9. Karışık Kodlama

Bir çizelgeleme probleminde sonsuz sayıda çizelge mevcut olduğundan bunlar üç sınıflandırmaya tabi tutulabilir;

1. Aktif çizelge
2. Yarı aktif çizelge
3. Gecikmesiz çizelge

Optimal çizelge, aktif çizelge setlerinden oluşur. Gecikmesiz çizelge aktif çizelgeden daha küçüktür ve optimum çözümü garanti etmez. Bütün kodlama teknikleri, iş çizelgeleme problemlerine aktif bir yapı kazandırır [5].

Bahsedilen toplam dokuz çeşit kodlama sisteminin farklı çizelgeleme problemi tiplerine göre avantaj ve dezavantajları, hangi tip problemlerde hangi kodlama sistemini kullanmanın avantajlı olacağı henüz tam olarak belirlenmemiştir [5].

2.3. Üreme Operatörü

Bu prosede herbir dizi (kromozom) amaç fonksiyonuna göre kopyalanır, gelecek nesilde daha iyi döl verecek diziler seçilir. Üreme bir nevi yapay seleksiyon işlemidir.

Mevcut populasyon içinde en iyi olanlar, rulet çemberi yöntemine göre seçilir. Her bir dizinin amaç değeri rulet çemberine yerleştirilir, dizi sayısı kadar çember çevrilir. Her çevirme işleminde karşılık gelen kromozomlar bir sonraki nesilde döl vermek için seçilir. Seçilen kromozomlar gen havuzunda biriktirilir. Aynı diziden gen havuzunda birden fazla mevcut olabilir. Seçim işleminden sonra çaprazlama yapılır.

Genetik algoritmalarda Elitist stratejisi kullanıldığında başlangıç populasyonu rassal olarak seçilir. Daha sonraki seçim işlemlerinde, Elitist stratejisine göre, her nesilde en iyi çıkan kromozomlar sonraki nesillere değişiklik yapılmadan aktarılır.

Üreme operatörü çalıştırıldığında yapılacak işlem adımları aşağıda kısaca özetlenmiştir [5]. Mevcut popülasyondaki (başlangıç popülasyonunda olabilir) dizileri (kromozomları) POP sembolü ile, üreme işlemi sonucunda elde edilen yeni popülasyonu da NEWPOP sembolü ile gösterelim.

1. Başlangıçta popülasyondaki dizi sayısı NPOP olsun.
2. Rulet çemberi yöntemine göre popülasyondaki dizilerin uygunluk değeri toplamı NSUM olsun.
3. Bir değerinden NPOP değerine kadar sıra ile sayıları artırıp 4. Adıma geç.
4. Bir değeri ile NSUM değerleri arasında rassal sayı üretilip bu sayılara karşılık gelen uygunluk değerine göre kromozomları oluştur.
5. Popülasyondaki yüksek değerli dizileri yeni popülasyon (NEWPOP) içinde göster.

2.4. Çaprazlama Operatörü

Çaprazlama, iki ebeveynin dizisinden yeni nesil oluşturmak için gerçekleştirilen bir süreçtir. Çaprazlama da iki ayrı işlem yapılır [5]. Bu işlemler,

1. Gen havuzunda biriktirilen kromozomların rassal seçimi,
2. Seçilen kromozom çiftlerinin bilinen çaprazlama operatörleri ile işleme tabi tutulmasıdır.

Çaprazlama operatörünün tasarımında iki temel düşünce hakimdir. Bu düşünceler,

1. Çaprazlama işlemi ile diziler üzerinde çok az değişiklik yapmak ve bütün varyasyonlarda iki noktalı çaprazlama kullanmak,

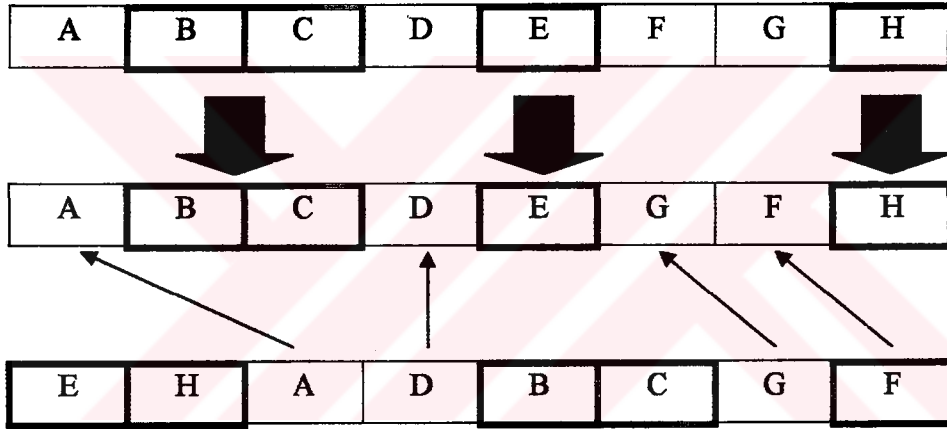
2. Çaprazlama işlemi ile diziler üzerinde çok büyük değişiklikler gerçekleştirmek ve farklı çaprazlama metotlarını uygulamaktır.

Klasik genetik algoritmalarda genelde bir noktalı çaprazlama operatörü kullanılır.

Genetik algortmada kullanılan çaprazlama operatörleri detaylı olarak aşağıda anlatılmıştır.

2.4.1. Pozisyona Dayalı Çaprazlama (PBX)

Şekil 2.3'te görüldüğü gibi rassal olarak seçilmiş pozisyondaki işler bir ebeveynden çocuğa kalıtsallaştırılır. Diğer ebeveyndeki sıradaki işlerin görünümündeki sırada diğer işler yerleştirilir. İlk olarak pozisyondaki sayılar $[1, n]$ rassal tamsayılar şeklinde düzenlendi. Daha sonra bu pozisyonlar rassal olarak seçildi. Her pozisyonun çaprazlama olasılığı 0.5'dir [5].



Şekil 2.3. Pozisyona dayalı çaprazlama

2.4.2. Sıraya Dayalı Çaprazlama (OBX)

Bu yöntemde bir grup nokta rasgele seçilir. Birinci kromozomun seçilen noktalara karşılık gelen karakterleri aynen yerlerini korur. İkinci kromozomun seçilen noktalara ait karakterleri birinci kromozomun aynı noktalarındaki karakterlerinin önüne getirilir. Geriye kalan boş pozisyonlara ikinci kromozomdan aktarılan yeni karakterler de göz önünde bulundurularak ilk kromozomun kullanılmayan karakterleri tarafından sıra ile (soldan sağa) yerleştirilerek yeni bir kromozom elde edilir. Bu tür çaprazlama kromozomu oluşturan karakterlerin sayı ve sıralarının önem taşıdığı durumlarda kullanılır [5]. Bu çaprazlama işlemine ait birer çaprazlama örneği aşağıda Şekil 2.4'de verilmiştir.

2.4.4. Dairesel Çaprazlama (CX)

Dairesel çaprazlama yöntemi Davis, Goldberg ve Lingle tarafından geliştirilmiş bir yöntemdir. İşlem adımları örnek üzerinde gösterilmiştir [5].

İki ayrı ebeveyn (kromozom) C ve D olarak kodlanmış olsun.

C = 9 8 2 1 7 4 5 10 6 3

D = 1 2 3 4 5 6 7 8 9 10

Çaprazlama haritası yerine C kromozomundan (ilk bireyden) en sondaki değer seçilir.

C' = 9 - - - - - - - - -

C bireyinden 9 değerini seçtiğimizde, 9'un karşılığı D bireyinde 1 olmaktadır; C'de 1 geni yerine yazılır. 1'in karşılığı D'de 4 olmaktadır; 4 geni yerine yazılır.

C' = 9 - - 1 - 4 - - 6 -

Kalan boşluklara diğer genler yazılır. Bu durumda yeni kromozomlarımız,

C' = 9 2 3 1 5 4 7 8 6 10

D' = 1 8 2 4 7 6 5 10 9 3 bulunur.

2.4.5. Doğrusal Sıralı Çaprazlama (LOX)

Falkenauer ve Bouffouix tarafından geliştirilmiştir. İş çizelgeleme problemlerinde dairesel çaprazlama kullanılamadığı için dairesel çaprazlamanın bir varyantı olan doğrusal sıralı çaprazlama geliştirilmiştir [5]. İşlem adımları şunlardır:

1. Mevcut populasyon içerisinde rassal olarak iki ebeveyn seç.
2. Seçilen bu iki dizi (kromozom) üzerinde rassal olarak iki alt dizi seç.
3. P1 dizisinden seçilen alt diziyi kromozomdan kopar boş kalan yerlere H yaz. Benzer şekilde P2 dizisinde de aynı işlemleri gerçekleştir.
4. Birinci alt diziyi P2'ye ve ikinci alt diziyi P1'e yerleştir.

2.4.6. Sıralı Çaprazlama (OX)

Bu yöntem Davis, Goldberg ve Lingle tarafından geliştirilmiştir. İşlem adımları örnek üzerinde gösterilmiştir [5].

$$A = 9 \ 8 \ 4 \mid 5 \ 6 \ 7 \mid 1 \ 3 \ 2 \ 10$$

$$B = 8 \ 7 \ 1 \mid 2 \ 3 \ 10 \mid 9 \ 5 \ 4 \ 6$$

Sıralı çaprazlama yöntemine göre 5, 6, 7 genleri yerine 2, 3, 10 genleri atanır ve A kromozomunda daha önce 2, 3 ve 10 bulunan yerlere H yazılır. Buna göre,

$$B = 8 \ H \ 1 \mid 2 \ 3 \ 10 \mid 9 \ H \ 4 \ H \text{ yeni kromozomlarımız,}$$

$$A' = 5 \ 6 \ 7 \mid 2 \ 3 \ 10 \mid 1 \ 9 \ 8 \ 4$$

$$B' = 2 \ 3 \ 10 \mid 5 \ 6 \ 7 \mid 9 \ 4 \ 8 \ 1$$

2.4.7. Alt Değişimli Çaprazlama (SXX)

Koobayashi, Brady ve Mühlenbein tarafından farklı zamanlarda yapılan çalışmalar ile gezgin satıcı problemi için geliştirilmiş bir çaprazlama yöntemidir. Bir iş sıra matrisinde, n iş, m makine problemi için n x m matris kodu yazılabilir. Bu koddaki her bir gen bir iş operasyonuna karşılık gelmektedir. Operasyon şeklinde kodlanan çizelgeleme problemlerinde iki işlem adımı ile SXX çaprazlama metodu kullanılır [5]. Bunlar:

1. Her bir dizideki genleri makine bazında tanımla.
2. Dizideki her bir geni makine makine olarak değiştirip yeni nesiller oluştur.

2.4.8. İş Tabanlı Sıralı Çaprazlama

One ve arkadaşları tarafından geliştirilen bu yöntem, alt değişimli çaprazlama metodunun bir varyantıdır. Bu metodun geliştirilmesinin nedeni, iş çizelgelemede, bütün işlerin mutlaka çizelgede yer alması zorunluluğundan kaynaklanmaktadır. İş tabanlı sıralı çaprazlama, iş sıra matrisinin kodlanması şeklindedir. Aşağıdaki işlem adımlarından oluşur [5].

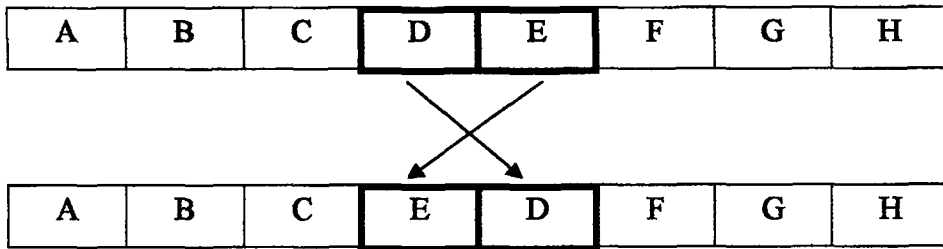
1. Her bir makineden geçen iş setlerini, kromozomlar üzerinde belirle.
2. Belirlenen işleri, ilk döl (çocuk) için makine makine olarak kromozomlardan (ebeveynlerden) kopyala, aynı işlemleri ikinci çocuk için gerçekleştir.
3. Seçilmeyen işleri de dikkate alarak ilk çocuğu (kromozomu) bütün genleri ile oluştur. Benzer şekilde diğer çocuklar içinde aynı işlemleri gerçekleştir.

2.5. Mutasyon Operatörü

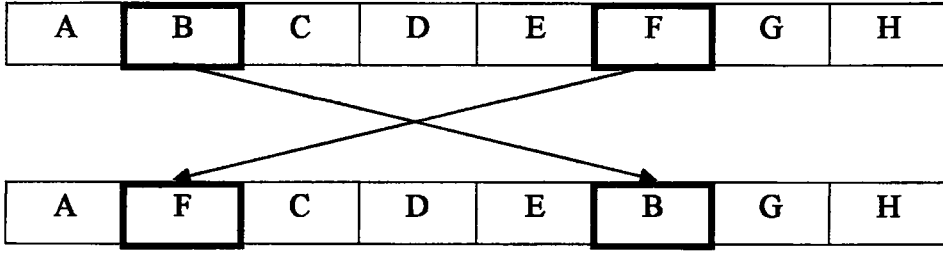
Genetik algoritmalarda önemli rol oynayan faktörlerden biri de mutasyon operatörüdür. Mutasyon frekansının belirlenmesi önemli bir faktördür. Doğal popülasyonlarda mutasyon oranı çok düşük olduğundan genetik algortmada da genelde düşük seçilir.

Son on yılda çok çeşitli mutasyon yöntemleri geliştirilmiştir [5]. İş çizelgeleme ile ilgili geliştirilen yöntemler:

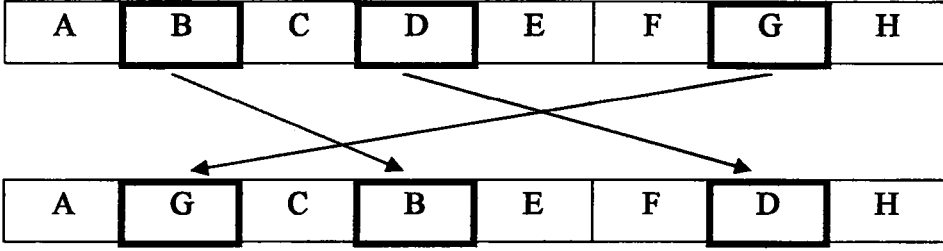
1. Ters Mutasyon: Bir kromozomda rassal olarak iki pozisyon seçilir. Bu iki pozisyonadaki alt diziler ters çevrilir.
2. Komşu İki İş Değiştirme: Şekil 2.5’de gösterildiği gibi iki komşu iş değiştirilir. Değiştirilen iki komşu iş rassal olarak seçilir.
3. Keyfi İki İş Değiştirme: Şekil 2.6’da gösterildiği gibi keyfi seçilen iki iş değiştirilir. Değiştirilen iki iş rassal olarak seçilir ve keyfidir. Özel bir durum olarak değiştirilen iki komşu işi bu mutasyon içerir.
4. Keyfi Üç İş Değiştirme: Şekil 2.7’de gösterildiği gibi keyfi olarak seçilen üç iş keyfi olarak değiştirilir. Mutasyondan sonra seçilebilen işlerin sırası rassal olarak belirlenir. Üstteki iki mutasyon operatörünün özel bir durumunu bu mutasyon içerir.
5. Shift (Araya Sokarak) Değiştirme: Bu mutasyonda Şekil 2.8’de gösterildiği gibi bir operasyondaki bir iş kaydırılır ve diğer bir pozisyona koyulur. İki operasyonda rassal olarak seçilir. Komşu iki iş değiştirmenin özel bir durumunu bu mutasyon içerir ve keyfi üç iş değiştirmeyele bir kesişime sahiptir.



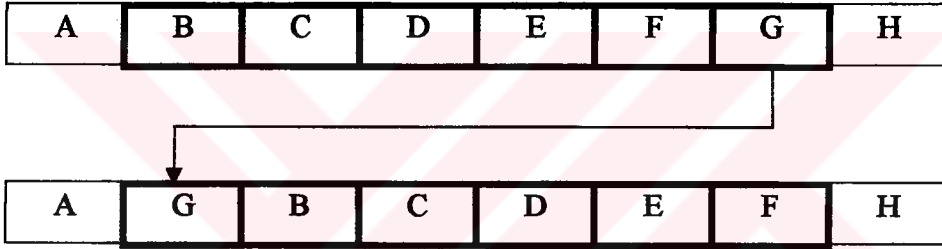
Şekil 2.5. Komşu iki iş değiştirme



Şekil 2.6. Keyfi iki iş değiştirme



Şekil 2.7. Keyfi üç iş değiştirme



Şekil 2.8. Shift değiştirme

3. AKIŞ TİPİ ÇİZELGELEME PROBLEMLERİ

3.1. Giriş

Akış tipi çizelgeleme, birbirinden farklı m makine ve n iş bulunduğu, her bir işin m operasyondan oluştuğu, her bir operasyonun farklı makinelerde yapıldığı ve bütün işlerin operasyonlarının aynı sıra ile yapıldığı problemlere denir.

Akış tipi çizelgeleme probleminde amaç fonksiyonu, toplam akış süresinin ağırlığını ve en son işin tamamlanma zamanını (makespan) minimize etmektir. Akış tipi iki makine çizelgelemede, F toplam akış zamanını ve F' ortalama akış zamanını minimum yapmak NP problemleri kapsamına girer.

Ignall ve Schrage, 1965 yılında ilk olarak, dal-sınır tekniği ile akış tipi çizelgeleme problemlerini çözmeye çalışmıştır. Hoogwen farklı bir dal-sınır prosedürü geliştirmiştir. Bu çalışmalarda sadece bir tek kriter ($C_{\max}$, F , F') düşünülmüştür. Karar verebilmek için çoğu zaman iki veya daha fazla kriterin göz önüne alınması gerekir. Bu yüzden çok kriterli iki makine akış problemi gündeme gelmiştir. Selen, Hott ve Wilson, çok kriterli iki makine akış çizelgelemesi için tamsayılı programlama modelini geliştirmiştir. Bununla toplam akış süresi ve en son işin tamamlanma süresini minimize etmeye çalışmıştır.

İş çizelgelemede kullanılan yöntemler:

1. Matematik programlama
2. Öncelik kurallarına göre sıralama
3. Simulasyon tabanlı metotlar
4. Yapay zeka tabanlı metotlar
5. Sezgisel yöntemler
6. Çok temsilli paradigmlar (Multi-agent paradigm)
7. Dağıtım yapay zeka üretim çizelgeleme (DAI yazılımları)
8. Üretim düzgünleştirme ve çizelgeleme operatörü yöntemi

Çizelgeleme optimizasyonunda matematik programlama kullanmak zordur; çünkü hesaplama süresi çok uzun olmaktadır, bu yüzden optimum çözümü bulmak zordur. Simulasyon tabanlı çizelgeleme programları esnek olduğundan son yıllarda tercih edilmektedir [5].

3.2. Optimal Çözüm Koşulları Analizi

Problem çözülürken aşağıdaki kabuller yapılmıştır [5].

1. Tezgah hazırlık süreleri bilinmektedir ve işlem sürelerinin içerisinde yer almaktadır.
2. Makinelerin önceden rezervasyon yapılmasına izin verilmez. Her bir operasyon önce başlar, ikinci operasyonun başlayabilmesi için, birincinin tamamlanması gerekir.
3. Çizelgeleme periyodu boyunca makineler sabit, mevcut yerinde ve kapasiteleri aynıdır.
4. Bir iş aynı anda birden fazla makinede işlenemez.
5. Makine bir anda bir işi işleyebilir; birden fazla iş işleyemez.
6. Bir işe ait operasyon bölünemez, bir makinede tamamlanması gerekir.

3.3. Genetik Algoritma Modelinin Kurulması

Genetik algoritmanın, iş çizelgelemede kullanılmasında temel iki yaklaşım benimsenmiştir [5].

1. Genetik algoritma kullanılarak uygun permütasyon geliştirme
2. Geliştirilen permütasyona uygun çözüm elde etmek için sezgisel metotları oluşturma

3.3.1. Kodlama

Yapının kodlanması, genetik operatörlerinin uygulanışını etkiler. Permütasyon çizelgeleme problemleri için, yapının temel kodlanması, işlerin dizilimini basitleştirir.

Uygulamada iş çizelgeleme problemi için temel kodlama kullanılmıştır. Kromozom dizisi üzerinde bulunan her bir rakam bir işi temsil etmektedir.

Örneğin, 8 işe sahip bir akış tipi probleminde 1, 2, 3, 4, 5, 6, 7, 8 rakamları sıra ile işleri temsil etmektedir [5].

3.3.2. Başlangıç Populasyonu Oluşturma

Genel olarak ilk populasyon birkaç yolla oluşturulabilir. Genetik algoritmanın birçok uygulamalarında rassal olarak oluşturulmaktadır. İlk populasyonun farklı çözüm kümelerinden oluşturulması daha etkili araştırmayı sağlar. İlk populasyon ortalama uygunluk değerini yükseltir ve hesaplama zamanını (nesil sayısını) düşürür. Bu nedenlerden dolayı farklı prosedürlerle ilk populasyonu oluşturmak, populasyon ortalama uygunluk değerini yükseltecek ve bizleri kısa sürede çözüme ulaştıracaktır [5].

Test işlemlerinde kullanılan çizelgeleme problemlerinin işlem süreleri, üniform dağılıma göre 1-99 aralığında rassal olarak oluşturulmuştur.

Populasyon büyüklüğü literatürdeki çalışmalara benzer şekilde 10, 20, 30, 40 ve 50 kromozom şeklinde rassal olarak belirlenen değerlerden aralık daraltmasına uygun olacak şekilde 20 ve 40 kromozom olarak seçilmiştir.

3.3.3. Üreme Operatörü

Genetik algoritmanın yürütülmesinde düşünülecek diğer önemli bir faktör seçim olasılığıdır. Genel tanıma göre bireyin seçim olasılığı, populasyonda bireyin performans ölçümünde projelendirilmelidir [5]. Mevcut problemde üreme yöntemi olarak, beş ayrı üreme operatörü üzerinde daha önceden yapılan deneyler sonucundan [5] en uygun bulunan kısmi yapay seçim ile akış zamanlı rulet çemberi yöntemi kullanılmıştır.

1. Toplam akış zamanına (makespan) göre rulet çemberi yöntemi: Bu yöntemde, populasyondaki her bireyin seçim olasılığı prosedürü aşağıdadır.

- a) Populasyonda her birey için toplam akış zamanı hesapla
- b) Populasyondaki toplam maksimum akış zamanı F_{max} 'ı bul
- c) Her bireyin toplam akış zamanı ve F_{max} arasındaki farka eşit olan her bireyin uygunluk değerini hesapla
- d) Uygunluk değerine dayanan her bireyin seçim olasılığını hesapla (Bu değer, bireylerin uygunluk değerlerinin, populasyondaki her bireyin

uygunluk deęerinin toplamına bölünmesine eşit olan her bireyin seçim olasılığıdır.)

2. Kısmı yapay seçim yöntemi: Bu yöntemde son iki nesilden elde edilen kromozomlar amaç fonksiyonuna göre büyükten küçüğe sıralanır. Bu sıralama içinde en kötü sonuca sahip (en büyük makespan deęeri olan) kromozomlar tamamen ters çevrilerek gen havuzuna atılır, en iyi çözüm kromozomlarından alınır.

3.3.4. Çaprazlama Operatörü

Çaprazlama operatörü genetik algoritmanın temel işlemcisidir. Genetik algoritmanın performansı ona baęlıdır. Akış tipi çizelgelemede genellikle bir ve iki noktalı çaprazlama yöntemi kullanılır [5].

Çaprazlama operatörü olarak, ikinci bölümde belirlenen çaprazlama operatörlerinden altı tanesi (pozisyona dayalı çaprazlama, sıraya dayalı çaprazlama, kısmi planlı çaprazlama, dairesel çaprazlama, doğrusal sıralı çaprazlama, sıralı çaprazlama) üzerinde daha önceden yapılan deneyler sonucundan [5] en uygun bulunan sıralı çaprazlama (OX) ile pozisyona dayalı çaprazlama (PBX) kullanılmıştır.

Alt deęişimli ve iş tabanlı çaprazlama yöntemleri, atölye tipi çizelgeleme problemlerinde kullanılmakta olup; akış tipi çizelgelemeye uygun deęildir.

Çaprazlama oranı, %0-%100 aralığında aralık daraltmasına uygun olacak şekilde %25 ile %75 seçilmiştir.

3.3.5. Mutasyon Operatörü

Mutasyon operatörü olarak, ikinci bölümde belirlenen beş ayrı mutasyon operatörü arasında yapılan deneyler sonucunda [5] en uygun bulunan ters mutasyon ile keyfi üç iş deęiştirme mutasyon yöntemleri kullanılmış ve mutasyon oranı da %0-%100 aralığında aralık daraltmasına uygun olacak şekilde %25 ile %75 seçilmiştir.

4. DENEY TASARIMI

4.1. Deney Kavramı

Deneyler, herhangi bir proses veya sistem ile ilgili yeni bulgular keşfetmek amacıyla, pek çok alanda yapılmaktadır. Herbir deney, istatistik literatüründe, aslında bir sınamadır. Başka bir deyişle deney, bir proses veya sistemin girdi değişkenlerinde yapılan anlamlı değişikliklerin, çıktı üzerinde oluşturduğu değişim şeklini ve nedenini belirlemek amacıyla gerçekleştirilen sınama veya sınama gruplarıdır.

Deney tasarım ve analizleri, özellikle yeni ürün tasarımı, üretim prosesleri geliştirilmesi ve iyileştirilmesi gibi mühendislik alanlarında önemli rol oynamaktadır. Bu çalışmalarda amaç, dayanıklı tasarım elde etmektir. Burada dayanıklı, kontrol edilemeyen faktörlere, örneğin nem, toz, ısı gibi çevre koşullarına, müşteri kullanımında farklı uygulamalara ve malzemelerdeki farklılıklara karşı duyarsız, yani onlardan etkilenmeyen, ürün ve proses anlamında kullanılmaktadır.

Bir deneye örnek olarak, bir metalurji mühendisinin bir alüminyum alaşımının iki farklı sertleştirilmesi prosesinin (yağ ile soğutma ve su ile soğutma), etkilerini incelemesini ele alalım [6]. Burada deneyin amacı, hangi soğutma prosesinin alaşım üzerinde maksimum sertleştirme sağladığının belirlenmesidir. Mühendis, herbir soğutma işlemi için belli sayıda alaşım örneğini işleme tabi tutmaya ve sonra sertlik ortalamasını ölçmeye karar verir. Herbir soğutma prosesi için kullanılan örneklerin ortalama sertlikleri, hangi prosesin daha iyi olduğunu karar vermede kullanılır.

Bu basit deneyi göz önüne alarak, şu önemli sorular göz önünde bulundurulmalıdır:

1. Bu iki çözelti, ilgi alanına giren tek soğutma yöntemleri mi?
2. Bu deneyde sertliği etkileyen, incelenmesi veya kontrol edilmesi gerekli başka faktör(ler) var mı?
3. Herbir soğutma çözelti için kaç adet alaşım örnek kullanılmalı?

4. Soğutma çözümlerine örnekler ne şekilde dağıtılmalı ve veriler hangi sıra ile toplanılmalı?
5. Hangi veri analiz metodu kullanılmalı?
6. İki soğutma yöntemi için gözlemlenen ortalama sertlik arasında çıkan hangi fark önemli sayılacak?

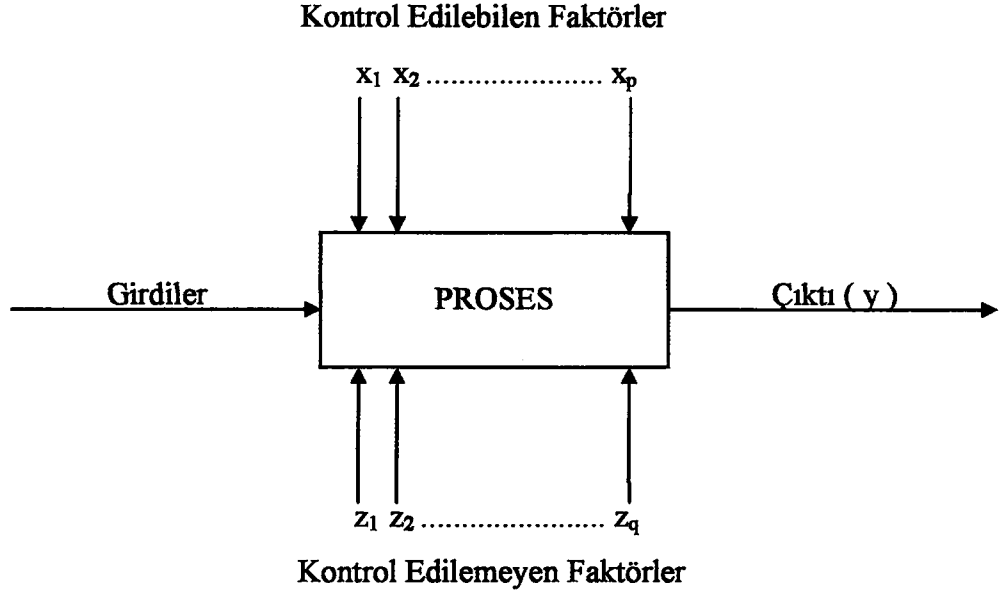
Bütün deneylerde, sonuçlar ve çıkarımlar, verilerin nasıl toplandığı ile çok yakından ilgilidir. Bu noktayı açıklamak için, yukarıda bahsedilen örnekteki metalurji mühendisinin yağ ile soğutmadaki örnekleri bir ısıdan, su ile soğutma örneklerini diğer bir ısıdan aldığını farz edelim. Ortalama sertlikler karşılaştırıldığında, mühendisin, gözlenen farkın ne kadarının soğutma yöntem farklılığından, ne kadarının da ısılar arasındaki farktan sonuçlandığını söylemesi güçtür. Bu nedenle, veri toplama metodu, analizi olumsuz yönde etkilemiştir.

4.2. Deney Tasarımının Amaçları

Deney tasarımı ürünün imal edilebilirliğinin araştırılmasında ve/veya varolan ürünün performansının geliştirilmesinde ve güvenilirliğinin devamlılığının sağlanmasında ürün maliyetinin aşağı çekilmesinde ve ürün geliştirme sürecinin kısaltılmasında etkin rol oynar [13].

Kalitenin ve maliyetin ortak düşmanı sistem ve/veya ürün özelliklerinde görülen değişimlerdir. Değişkenlik değişik kalite araç ve teknikleri ile belirlenebilir. Deney tasarım yöntemlerinin amacı değişkenliği azaltarak veya değişkenliğin azaltılmasının mümkün olmadığı veya çok yüksek maliyet gerektirdiği durumlarda ise sistemi/ürünü bu değişkenlere karşı güçlendirerek yüksek kalite ve düşük maliyeti sağlamaktır.

Genel olarak, deneyler proseslerin ve sistemlerin performanslarını anlamada kullanılır. Bu proses veya sistem, Şekil 4.1'deki modelle temsil edilebilir [6]. Prosesi makina, insan, metot kombinasyonları veya girdileri gözlemlenebilir çıktıya dönüştüren kaynaklar olarak görebiliriz. Proses değişkenleri $x_1, x_2, \dots, x_p$ gibi kontrol edilebilir ve $z_1, z_2, \dots, z_q$ gibi kontrol edilemeyen değişkenler olmak üzere iki bölüme ayrılır.



Şekil 4.1. Bir proses veya sistemin genel modeli

Deney tasarımının amaçları şöyle sıralanabilir:

1. y çıktısında en fazla hangi değişkenlerin etkili olduğunu belirleme
2. y çıktısını istenilen optimum seviyede tutmak amacıyla etkili x değişkenlerini hangi seviyelerde tutmak gerektiğini belirleme
3. y çıktısının değişirliğini minimumda tutmak amacıyla etkili x değişkenlerini hangi seviyelerde tutmak gerektiğini belirleme
4. $z_1, z_2, \dots, z_q$ gibi kontrol edilemeyen değişkenlerin tesirlerini minimumda tutmak amacıyla, etkili x değişkenlerini hangi seviyelerde tutmak gerektiğini belirleme

Deneyler genelde pek çok faktörü içermektedir. Çoğunlukla deney amacı, bu faktörlerin çıktı üzerindeki etkilerini belirlemektir. Bir deneyi planlama ve yürütme yöntemine “**deney stratejisi**” adı verilir. Deney stratejileri bir sonraki bölümde ayrıntılı olarak anlatılacaktır.

4.3. Deney Tasarımlarının Tipik Uygulamaları

Deney tasarım yöntemleri birçok disiplinde kendine uygulama alanı bulmuştur. Aslında deneyler bilimsel prosesin bir parçası olarak görülebilir. Genelde, tahminler yapılan proses hakkında yapılan aktiviteler süresince öğrenilir, prostesten veri elde edebilmek için deney yapılır, yeni tahminler oluşturmak için deneyden elde edilen

bilgiler kullanılır, tekrar bu tahminleri sınamak için deneyler yapılır ve böylece devam eder.

Deney tasarımları mühendislik dünyasında üretim prosesleri performanslarını iyileştirmede önemli bir role sahiptir. Ayrıca, yeni prosesler geliştirilmesinde de yoğun uygulamaları vardır. Proses geliştirilmesinde, deney tasarım tekniklerinin uygulanması şu sonuçları doğurabilir:

1. İyileştirilmiş proses çıktısı
2. Azaltılmış değişkenlik ve istenilen özelliklere daha yakın uygunluk
3. Azaltılmış geliştirme zamanı
4. Azaltılmış maliyetler

Deney tasarım yöntemleri, yeni ürünlerin geliştirildiği ve mevcut ürünlerin iyileştirildiği mühendislik tasarım aktivitelerinde de önemli rol oynar. Mühendislik tasarımı alanında deney tasarım uygulamalarına şunları gösterebiliriz:

1. Temel tasarım konfigürasyonlarının karşılaştırılması ve değerlendirilmesi
2. Malzeme alternatiflerinin değerlendirilmesi
3. Ürünün değişik ortam koşullarından etkilenmemesini sağlayacak tasarım parametrelerinin seçilmesi ve böylece ürünün dayanıklı olmasının sağlanması
4. Ürün performansı üzerinde etkisi olan tasarım parametrelerinin belirlenmesi

Bu alanlarda kullanılan deney tasarımları, daha kolay üretilen ürünler, performansı ve güvenilirliği iyileştirilen ürünler, azaltılan ürün maliyetleri, kısaltılan ürün tasarım ve geliştirme aşamaları süreleri ile sonuçlanabilir.

4.4. Deney Tasarım Yöntemlerinin Ortak Dili

Deney tasarım yöntemlerinde problem ister yeni ürün veya üretim sisteminin tasarlanması ister varolanların geliştirilmesine yönelik çalışmalar olsun ortak bir dilin kullanılması hem deney tasarımını hem de istatistiksel analizleri kolaylaştırmaktadır. Ayrıca bazı deney tasarımlarında deneyi tasarlayan ve sonuçların analizini gerçekleştirenlerin farklı kişiler olması bu durumu zorunlu hale getirmektedir.

Gerek bilinçsiz olarak tasarlanmış deneylerde ve gerekse sistematik deney tasarımlarında kullanılan terimlerin anlaşılabilirliğinin sağlanması, deney tasarımlarının verimliliğini artırmaktadır. Araştırma geliştirme çalışmalarında önemli olan sistematik ve düzenli olarak çalışmaktır. Bu yüzden herşeyden önce deneylerin hangi değişkenlerle ve hangi şartlarda yapılacağını ve sonuçta hangi değişkenin ölçülmesinin gerekli olduğunu bilmek gerekmektedir.

Deney tasarım yöntemlerinde deney matrisleri farklı olmasına rağmen kullanılan ortak dil genelde aynıdır. Ancak tasarımcıların bazı kavramlar üzerinde daha hassasiyetle durmaları deney oluşum felsefesinde temel ayrımları ortaya çıkarmaktadır.

Deneylerde kullanılan ortak kavramlar aşağıdaki şekilde sıralanabilir.

Sonuç Değişken (Y): Deneylerde gözlemlenen, ölçülen değer yani deneyin çıktısıdır. Sonuç değişkeni bazen bağımlı değişken, bazen proses performansının ölçümü bazen ise bir kalite karakteristiğidir. Deneyler bir veya daha fazla çıkış değişkenine yani kalite karakteristiğine sahip olabilirler [7].

Faktör (x): Kontrol edilebilen parametrelere faktör denir [10]. Bağımsız veya güncel değişken olarak da adlandırılır [7]. Deneylerde kontrollü olarak sonuç değişkeni üzerinde etkilerini gözlemlemek için devamlı değiştirilirler. Faktörler kantitatif veya kalitatif olabilirler [7]. Kantitatif faktörler sürekli skalada ölçülen faktörler olup sıcaklık örnek olarak verilebilir. Kalitatif faktörler ise sürekli skalada ölçülemezler örnek olarak hammadde tedarikçileri verilebilir [7]. Kalitatif faktörler için değer aralıkları, nasıl ölçülmesi gerektiği ve deney esnasında kontrol edilebilecek seviyeler önceden belirlenmelidir [10].

Gürültü Faktörü (z): Bazen geri plan değişkeni veya bloklama değişkeni olarak da adlandırılır. Bu değişkenler; deneylerde sonuç değişkeni etkilerler ama faktör olarak ele alınmazlar. Tipik geri plan değişkenleri zaman, operatör, kalıptaki boşluk ve donanım türü olarak sayılabilir [7].

Deney Birimi: Deneyde kullanılan malzemelerin en küçük miktarıdır. Örnek olarak yığınlar, malzemenin bir kilogramı, bir insan veya 10 m² yenilebilir. Diğer bir ifade ile malzemenin imalat aşamasında veya serviste bir denemedeki miktarıdır [10].

Seviye: Deneyde kantitatif veya kalitatif faktörlerin özel seçenekleri veya verilen değerleridir [7].

İşlem/Deneme/Çalışma: Deney tasarım terminolojisinde faktör seviyelerinin sonuç değişken üzerine etki eden belirli bileşimlerine denir [10]. Kimyasal bileşiğin kazancına etki eden faktörlerin fırın sıcaklığı ve hammaddenin satıcıları olduğu varsayılırsa, buna bağlı olarak bu iki faktörün değişik seviyelerinin ele alındığı kombinasyonların her biri bir denemedir.

Etki: Faktörün bir seviyeden öteki seviyeye değişmesi durumunda sonuç değişken üzerinde görülen değişimdir [7].

Etkileşim: Etkilerin lineer olmaması durumudur [7]. Diğer bir deyişle bir faktörün, sonuç değişkenine olan etkisi, diğer bir faktörün hangi değerde bulunduğuna bağlı ise, iki faktör arasında etkileşim var denir [2].

Özellikle sonuç değişkenler, faktörler ve geri plan değişkenleri arasındaki ayırım deney planlamada oldukça önemlidir. Faktörler ve geri plan değişkenleri sebep olarak ve sonuç değişkenleri ise etki olarak ele alınırsa bu ayırım kolaylaşır [7].

4.5. Temel Prensipler

Deney tasarımlarının istatistiksel analizi, deney için toplanan verilerin istatistiksel metotlarla analiz edilmesi, amaca uygun geçerli sonuçların, çıkarılması demektir. Deneysel hatalar sonucu oluşan veriler, ancak istatistiksel yöntemler ile anlamlı bir şekilde analiz edilebilir. Bu nedenle bir deneysel problem ancak iki bölümde incelenir: Deney tasarımı ve istatistiksel veri analizleri. Öngörülen tasarıma bağlı kalınarak analiz yapıldığından dolayı, bu iki konu birbirleriyle oldukça yakından ilgilidir. Bu çalışmada iki bölüm de gözlemlenebilir.

Bir deney tasarımının üç temel prensibi vardır: Tekrarlama, rassallık ve bloklama.

- **Tekrarlama :** Tekrarlama, temel deneyin tekrarlanması anlamına gelir. Örneğin, alüminyum alaşımının soğutulması örneğinde, bir örneğin yağ ile soğutulması ve bir örneğin su ile soğutulması ile bir tekrar yapılmış olunur. Böylece, her bir soğutma yöntemi için beşer adet örnek kullanıldığında, beş adet tekrar yapılmış olur. İki önemli özelliği vardır: Birincisi tekrar ile, deney sahibi, deneysel hataları tahmin etme şansını yakalar. Hata tahmini, toplanan veriler arasındaki

farklılıkların istatistiksel olarak önemli olup olmadığını belirlemede temel noktadır. İkinci olarak, veriler hakkında daha kesin bilginin elde edilmesini sağlar. Bunun anlamı şudur: r tekrar sayısı arttıkça, örnek ortalamasının varyansı azalır; daha kesin sonuç elde edilir.

- **Rassallık** : Rassallıkla kastedilen, deney bireysel koşullarında hem deneysel materyallerin, hem de sıralamanın rassal olarak dağıtılmasıdır. İstatistiksel metotlar, gözlemlerin ve hataların bağımsız dağılmış rassal değişkenler olmasını gerektirir. Rassallık, çoğunlukla bu varsayımı geçerli kılar. Deneyleri doğru şekilde rassallaştırmak ile, var olabilecek dışsal faktörlerin deney sonuçlarına etkilerini de yok etmiş olur. Örneğin, alüminyum sertleştirilmesi deneyinde, örneklerin farklı kalınlıklarda olduklarını ve soğutma aracı üzerinde kalınlığın etkisi olduğunu varsayalım. Eğer, yağ ile soğutmada kullanılan örneklerin tümü, su ile soğutmada kullanılan örneklerden daha kalınsa, bir soğutma yöntemini diğerine tercih ederken yanlıya düşmemek olası değildir. Oysa örnekleri soğutma araçlarına rassal dağıtma, bu problemi yok edecektir.
- **Bloklama** : Bloklama, deneyin kesinliğini artırmak amacıyla kullanılan tekniktir. Bir deneyde, deneysel malzeme bütününden farklılık gösteren ancak birbiri içinde homojen dağılmış verilere sahip olan deneysel malzeme parçası, bir blok oluşturur.

4.6. Deney Tasarım Aşamaları

Deney tasarım ve analiz aşamalarında istatistik yaklaşımını kullanabilmek için, üzerinde çalışılacak konunun açıkça ifade edilmesi ve anlaşılması, verilerin toplanma şekli ve bu verilerin nasıl analiz edileceğinin belirlenmesi gerekmektedir. Deney tasarım aşamaları, Tablo 4.1’de özetlenmiştir. Herbir maddenin temel noktaları aşağıda özetlenmiştir.

Tablo 4.1. Deney tasarım aşamaları

-
1. Problem seçme ve tanımlama
 2. Faktörleri, seviyelerini ve aralıklarını seçme
 3. Sonuç değişkeninin seçilmesi
 4. Deney tasarımını seçme
 5. Deneyi gerçekleştirme
 6. Verilerin istatistik analizlerini yapma
 7. Sonuç ve öneriler
-

1. Problem seçme ve tanımlama: Deney gerektirecek bir problemi fark etmek pratikte çoğunlukla kolay değildir. Bir deney gerçekleşeceği zaman, amaç ile ilgili tüm detaylar göz önünde bulundurulmalıdır. Probleme net bir çözüm getirmek için problemin iyi belirlenmesi ve anlaşılması gerekmektedir.

2. Faktörleri, seviyelerini ve aralıklarını seçme: (2. ve 3. adımlar eşzamanlı veya ters sırada yapılabilir.) Deneyde değiştirilecek faktörler seçilmeli, bu faktörlerin değişim aralıkları ve denemelerde hangi seviyelerde tutulacağı belirlenmelidir. Bu faktörlerin istenilen seviyelerde nasıl kontrol altında tutulacağı ve nasıl ölçülmeleri gerektiği de üzerinde düşünülmesi gerekli konulardan biridir. Bunun için de proses bilgisi gerekli olacaktır. Bu proses bilgisi, pratik deneyim ve teorik bilgilerin kombinasyonlarından oluşabilir.

Deneyin amacı faktör inceleme olduğu zaman, faktör seviyelerini az sayıda tutmakta fayda vardır. Genelde iki seviye seçme, bu tür çalışmalarda iyi sonuçlar vermektedir. Aralık seçme de önemli bir konudur. Faktör incelemede ilgilenilen aralığın geniş seçilmesi yine verimli sonuçlar çıkaracaktır. Daha sonra, hangi değişkenlerin önemli olduğunu ve hangi seviyelerde iyi sonuçlar verdiğini öğrendikçe, ilgilenilen aralık daha çok daraltılabilir.

3. Sonuç değişkeninin seçilmesi: Sonuç değişkeninin seçilmesinde dikkat edilecek nokta, üzerinde çalışılan proses hakkında gerekli bilgiyi sağlamasıdır. Birden fazla sonuç değişkeni seçilmesi de mümkündür.

1, 2, ve 3. adımlar deney öncesi planlama adı altında toparlanırlar.

4. Deney tasarımı seçme: Yukarıdaki deney öncesi planlama aktiviteleri doğru bir şekilde yapıldıktan sonra, bu aşama daha kolay hale gelir. Örnek sayısının seçilmesi, herbir deney koşulunun deneme sırasının ve tekrarlanma sayısının belirlenmesi, blok veya daha başka kısıtlayıcı şartların bulunup bulunmadığının saptanması gibi kriterler, deney tasarımı oluşturur.

Bu adımı destekleyecek istatistiksel paket programları da mevcuttur. Bu programa; faktör sayıları, seviyeler, aralık (minimum değer-maksimum değer) girilir ve istenilen tasarıma göre analizler yapılır.

Tasarımı seçerken, deney amaçlarını da göz önünde tutmak gereklidir. Hangi faktörlerin çıktıda farklılık oluşturduğu ve ne derecede etkili olduklarını, bu amaç çerçevesinde inceleriz. Bu durumda hangi faktörlerin farkı oluşturduğu ve çıktı üzerinde ne derece etkili olduğunu bulmak isteriz. Bizim için benzerliğin/eşitliğin önemli olduğu durumlar da ortaya çıkabilir. Örneğin, A standart ve B daha maliyetli yöntem olmak üzere, iki farklı üretim yönteminin karşılaştırılması ile ilgilenilebilir. Bu durumda, iki yöntemin çıktı üzerinde fark oluşturmadığını ispatlamaya yönelik bir deney de olabilir.

5. Deneyi gerçekleştirme: Deneyi gerçekleştirirken, herşeyin planlandığı gibi yapıldığından emin olmak için prosesi takip etmek çok önemlidir. Deney prosedüründeki hatalar deney geçerliliğini yok edecektir.

6. Verilerin istatistiksel analizlerini yapma: Veri analizlerini destekleyecek pekçok yazılım paketleri bulunmaktadır, adım 4'te de deney tasarımı seçiminde bu paketlerden faydalanılmaktadır. Özellikle basit grafiksel metotları, sapma/hata analizleri ve model uygunluk kontrolleri önemli analiz teknikleridir. Hata analizleri model uygunluk sınamaları da kullanılan önemli analizlerdendir.

Unutulmamalıdır ki istatistiksel yöntem, faktör veya faktörlerin etkilerini kanıtlamaz. Sadece sonuçların güvenilirliği veya geçerliliği üzerine yol göstericidir. Yerinde kullanımında, istatistiksel yöntem hiçbir şeyi ispatlamaya yönelmez, ancak oluşan farklılıkları ölçmemize ve belirlenen bir güven aralığı atamamıza yarar. İstatistiksel yöntemlerin en önemli avantajı, karar verme sürecine nesnellik katmasıdır.

7. Sonuç ve öneriler: Veri analizlerinden sonra, deney sahibi, sonuçlara göre çıkarımlar yapmalı, nasıl hareket edileceğini belirlemelidir. Bu aşamada grafiksel gösterimlerden sıkça faydalanılmaktadır.

Tüm bu proses boyunca, unutulmamalıdır ki deney öğrenme sürecinin bir parçasıdır. Sistem ile ilgili hipotezler formüle edilir, bu hipotezlerin geçerliliğini araştırmak için deneyler yapılır, çıkan sonuçlara göre yeni hipotezler formüle edilir ve böylece devam eder. Görülmektedir ki, deney iteratif bir çalışmadır. Çalışmanın başlangıcında çok büyük, kapsamlı, tek bir deney yapmak hatalı bir davranıştır. Başarılı bir deney için, ilgili önemli faktörleri, bu faktörlerin değişeceği aralığı, doğru seviye sayılarını ve bu değişkenlerin doğru ölçüm ünitelerini bilmek gerekmektedir. Genellikle, bu soruların cevabı tam olarak başlangıçta bilinmez, ancak deney prosesi süresince öğrenilir. Deneyde ilerledikçe, bazı değişkenler ilave edilir, bazıları çıkarılır, bazı faktörler için aralık değiştirilir veya farklı sonuç değişkeni üzerinde durulmaya başlanılır. Genel bir kural olarak, mevcut kaynakların %25'inden fazlası ilk deney için kullanılmaz. Böylece deney süresince, diğer aşamalar içinde yeterli kaynak kalınması sağlanmış olunur.

4.7. Deney Tasarımının Tarihçesi

Deney tasarımı, 1920'lerde, istatistik biliminin babası sayılan İngiliz istatistikçi Sir Ronald Fisher tarafından, tarım alanında araştırmalar yaparken bulunmuş ve geliştirilmiştir. Fisher, ayrıca, deney verilerinin analizi için bugün klasik sayılan “varyans analizi” (ANOVA) yöntemini de geliştirmiştir. Yöntem, kısa bir süre içinde, Amerika'da tarım sektöründe üretimin geliştirilmesi için yoğun olarak uygulanmış ve Amerika'nın bu alanda dünya lider konumuna gelmesine büyük katkıda bulunmuştur. Tarım alanında, çeşitli gübre ve dozları ile iklim koşullarının ve sulama seviyelerinin çeşitli ürünlere olan etkilerini belirlemek üzere uygulanmıştır. Deneysel tasarım literatüründe adı önemle vurgulanan diğer bilim adamları arasında F. Yates, R. C. Bose, O. Kempthorne, W. O. Cochran, R. H. Myers, J. S. Hunter, W. G. Hunter ve G. E. P. Box sayılabilir.

Deney tasarımı daha sonra kimya ve ilaç sektörlerinde de uygulanmış olmasına rağmen, imalat sektöründeki uygulamaları, 1970'lere kadar son derece kısıtlı kalmıştır. Amerika'da imalat sektörü, 1980'lerin başında, deney tasarımını Japon

kalitesinin nedenlerini araştırırken yeniden keşfetmiştir. Deney tasarımı, o tarihlerde Japonya’da profesör Genechi Taguchi’nin önderliğinde yoğun ve etkili olarak uygulanmaktaydı. Taguchi, deney tasarımına kuramsal yenilikler getirmemiştir. Ancak, üretimdeki uygulamalarda yenilikler yapmış ve başarılı uygulamalarla yöntemin imalat sektöründe kabul görmesini sağlamıştır [2].



5. DENEY TASARIM YÖNTEMLERİNİN İNCELENMESİ

5.1. Deney Tasarım Yöntemleri

Deneylerin tasarımında araştırılan sisteme etki eden faktörlerin optimum seviyelerini belirlemek ortak amaç olmasına rağmen uygulama biçimlerindeki farklılık sonuçta değişik yöntemlerin ortaya çıkmasına neden olmaktadır. Uygulama şekillerine göre deney tasarım yöntemleri,

1. Klasik Yöntemler
2. Taguchi Yöntemi
3. Shainin Yöntemi
4. Genetik Algoritma Yöntemi olarak dört grupta incelenebilir.

5.1.1. Klasik Yöntemler

Klasik deney tasarımlarında ele alınan parametreler kontrol edilebilen faktörlerdir. Klasik deney tasarım yöntemleri de uygulamada kendi aralarında üçe ayrılır. Bunlar:

- 1) Bir Kerede Bir Faktör
- 2) Tam Faktöriyel
- 3) Kesirli Faktöriyel

Klasik deney tasarım yöntemleri arasındaki bu ayrımın sebebi faktörler arası etkileşimlerin deney matrisinde ele alınıp alınmaması esasına dayanır.

5.1.1.1. Bir Kerede Bir Faktör

Klasik deney tasarım yöntemleri arasında yer alan bir kerede bir faktör yöntemi deney tasarım yöntemleri içerisinde en eski olan ve çok sık uygulanan yöntemdir.

Bir kerede bir faktör değiştirilerek yapılan deneysel tasarım, diğer tasarım yöntemlerinin en basit olanıdır. Bu yaklaşım genelde bilimsel çalışmalarda adım adım her faktörün proses üzerine etkisini belirlemek için kullanılır (Şekil 5.1) [14].



Şekil 5.1. Bir kerede bir faktör değiştirilerek yapılan deneylerin tasarım yöntemi akış şeması

Bu metod, her bir faktör için bir başlangıç noktası veya temel seviye seçmekle başlar ve kendi dışındaki faktörleri temel seviyelerinde sabit tutarak her bir faktörü kendi aralığı üzerinde değiştirerek devam eder.

Deneyssel analizlerde genelde birden çok faktör ele alınır. Gerçekçi problemlerde önemli faktörler diğerlerinden izole edilirler. Birden fazla faktör varsa bu durumda deneylerin bir kerede bir değişken değiştirilerek yapılması zordur [15].

Bir kerede bir faktör değiştirilerek yapılan deneylerin dezavantajlarından biri mali verimliliğinin olmaması diğeri de faktörler arası etkileşimlerin belirlenememesidir. Etkileşimler bazı faktörlerin seviyelerinden etkilenen belirli faktörler ve sonuç değişken arasındaki ilişkinin doğal sonuçları olarak tanımlanırlar [13].

Etkileşimler faktörlerin sonuç fonksiyonu üzerine bileşik ilişkileri olarak gösterilir. Bu tür etkiler çok faktörlü (multifaktör) tasarımlarda ele alınırlar, çünkü gerçek dünya olaylarına daha yakındırlar. Bir şirket içinde değişik bölümlerde (faktör A) yeni motivasyon eğitim programlarının memurların (faktör B) verimliliği üzerine etkisinin fazla olması A ve B yani bölümler ve memurlar arasındaki etkileşime örnek oluşturmaktadır [10].

Deney tasarımında son derece önemli bir konu olan etkileşimi biraz daha açıklamak için günlük hayattan bir örnek verelim. Çeşitli faktörlerin bir insanın uyanıklık derecesine olan etkilerine bakalım. Zinde bir insanın uyanıklık derecesine % 100 diyelim. Şunları varsayalım: Bir kadeh içki uyanıklık derecesini % 90'a indiriyor ve bir doz sakinleştirici ilaç (sinir ilacı) uyanıklık derecesini % 80'e indiriyor. Doktorun tavsiyesinin aksine, kişi sakinleştirici ilacı bir kadeh içkiyle alıyor ve derin bir uykuya dalıyor (uyanıklık % 0'a düşüyor). Bu örnek iki faktör (içki ve ilaç) arasında önemli bir etkileşim olduğunu gösteriyor. Etkileşim olmasaydı, ilacın içkiyle alınması sonucunda, uyanıklık iki etkinin toplamı kadar azalacak, yani yalnızca % 70'e inecekti [2].

Etkileşim grafik olarak sonuç değişken değerinin dış hatlarının eksenlerle (ilgili faktörlerle) olan uyumu değerlendirilerek yapılır. Dış hatların koordinat eksenlerine paralel olması etkileşim olmadığını gösterirken, dış hatların yönünün eksenlere göre dönmüş olması etkileşimin varlığını gösterir [10].

Bir kerede bir faktör değiştirilerek yapılan deneylerde bir işlemde yalnızca bir faktör değiştirildiği için deney sonunda o faktörün etkisi direkt olarak gözlenir. Bunun sonucu olarak deneylerin analizi aşamasında sadece faktörün sonuca olan etkisi grafikte gösterilir.

Deneylerin analizi esnasında eğer seviyelendirilen faktör, kantitatif yani sürekli skalada ölçülebilen karakterde ise bu durumda interpolasyonla aslında deneyde o faktör için ele alınmayan seviyelerinde sonuca olan etkisi hakkında bilgi sahibi olunabilir [10]. Kalitatif faktörlerde bu tür interpolasyon mümkün değildir.

5.1.1.2. Tam Faktöriyel

İki veya daha fazla seviyeli ve iki veya daha çok sayıda faktörün tüm kombinasyonlarının ele alındığı deney tasarım yöntemine tam faktöriyel tasarım adı verilir.

Tam faktöriyel deney tasarım yöntemlerinde deneylerin tam tekrarı için faktörlerin tüm mümkün bileşimleri gösterilir. Deneme sayısı faktörlerin tümünün seviyelerinin çarpımına eşittir. Örneğin bir televizyon üretiminde televizyon montaj süresi sonuç değişkeni olarak ele alınırsa operatörün ustalığı (6 seviye), ısılandırma (3 seviye) ve eğitim programı (4 seviye) etkili faktörlerdir. Bu durumda deneme sayısı $6 \times 4 \times 3 = 72$ 'dir. Bu durumda faktörlerin sayısı veya seviyeleri arttığında deneme sayısı artar [10].

Tam faktöriyel deney tasarımının avantajı bir kerede bir faktör değiştirilerek yapılan deney tasarımında mümkün olmayan faktörler arası etkileşimleri hesaplama olasılığı vermesidir [10].

Faktörler bağımsız ve etkileşim yoksa çok az deney temel etkileri aynı kararlılıkla göstermek için yeterlidir. Faktöriyel deneyler hangi faktörün önemli olduğunu da gösterir [10].

Tam faktöriyel deney tasarımı tüm mevcut bileşimlerin bulunmasını gerektirir. Bir kerede bir faktör değiştirilerek yapılan deney tasarımının aksine tüm deneydeki

faktörlere göre deney tipi devamlı olarak değişir ve etkileşimlerin etkilerinin değerlendirilmesine olanak verir [3].

Tam faktöriyel deney tasarımının en büyük dezavantajı deneylerin boyutunun üstel olarak faktör sayısı ve seviyeleri ile artmasıdır. Deneylerin tümünün yapılış maliyeti çok yüksektir. Bu durumda tüm işlemlerin tam tekrarının yapılmasına izin verecek yeterince büyük blok bulma olasılığı da önemlidir. Diğer bir dezavantaj ise etkileşimlerin yüksek seviyede olanlarının açıklanmasındaki zorluktur [13].

Deney şartlarındaki beklenmeyen değişimler faktörlerin etkilerinin hesaplanmasını zorlaştırır. Bu etkilerin azaltılması için deneylerin rassallık, tekrarlama ve bloklama gibi genel özelliklerinin kullanılması gereklidir. Böylece deneyi yapan kişinin ön yargıları veya deney ortamının hata potansiyeli azaltılır [3].

Tam faktöriyel deneyler genelde rassal tam bloklanmış tasarımlar şeklinde yürütülür. Rassal tam bloklanmış yapıyı kullanmak için blok etkileri ve faktör etkileri arasında etkileşim olmaması şartı aranır. Bloklamanın ilk amacı dış değişiklikleri kontrol etmektir. Bloklar ve faktörler etkileşmiyorsa bloklar diğer bir faktör gibi değerlendirilir ve tam rassal tasarımın kullanılması daha anlamlı olur [3].

Rassal blok tasarımları deneysel birimlerinin çift olarak ele alındığı deneylerde kullanılır. Bu tür deneylerde her çift, blok faktörlerin iki seviyeli olarak kullanıldığı bloklardır [3].

Tam faktöriyel deneylerde amaç, seviyelendirilmiş her faktörün mümkün olan tüm kombinasyonlarını denemektir. Eğer 3 tane faktör var ve bunlar 2 seviyeli ise gerçekleştirilmesi gereken deneme sayısı $2^3 = 8$ 'dir. Deneylerde belirlenmiş olan faktörlerin sayısı ve seviyeleri artmasına bağlı olarak deneme sayısı da üstel olarak artar.

Parametrelerin seviyeleri bilinmiyor buna rağmen parametrelerin tüm olası etkileşimleri belirlenmek isteniyorsa tam faktöriyel deney tasarlanmalıdır. Bu tür deneyler tasarlanmadan önce parametreler indirgenmelidir. Çünkü 2 seviyeli 4 parametre için işlem sayısı 16 iken 2 seviyeli 5 parametre için 32'dir [4].

Tam faktöriyel deneylerde faktörlerin temel etkilerinin yanı sıra tüm mümkün etkileşimlerinin de yer alması hem zaman hem de maliyet yönünden araştırmalara büyük yük getirmektedir. Durumu, iki seviyeli yedi faktör için açıklayalım [2].

k = faktör sayısı ve n = deneme sayısı olmak üzere $k = 7$ için tam faktöriyel deney yapılırsa gereken deneme sayısı $n = 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 2^7 = 128$ 'dir. Bu durumda, $n-1 = 127$ farklı etki hesap edilebilir. Bu 127 etkinin 7 tanesi ana etki, geri kalan 120 tanesi etkileşimdir.

Binom katsayısı formülü genel olarak şöyle verilir:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad (5.1)$$

Şimdi, binom katsayısı kullanılarak 120 etkileşimin dağılımı hesaplınsın (Tablo 5.1).

Tablo 5.1. 2^7 tam faktöriyel deney için etkileşim dağılımları

Etkileşim	Adet
2'li	$\binom{7}{2} = 7! / (2! 5!) = 21$
3'lü	$\binom{7}{3} = 7! / (3! 4!) = 35$
4'lü	$\binom{7}{4} = 7! / (4! 3!) = 35$
5'li	$\binom{7}{5} = 7! / (5! 2!) = 21$
6'lı	$\binom{7}{6} = 7! / (6! 1!) = 7$
7'li	$\binom{7}{7} = 7! / (7! 0!) = 1$
Toplam	120

Özet olarak, yedi faktör içeren tam faktöriyel deneyde 128 deneme yapılır. Hesap edilen 127 etkinin ancak 7 tanesi ana etki, geri kalan 120 tanesi etkileşimdir. Böyle bir deneyde, toplam maliyetin büyük bölümünün etkileşimler için harcandığı açıktır [2].

Tam faktöriyel deney tasarım yönteminin uygulanması için gerekli prosedür bir kerede bir faktör değiştirilerek yapılan basit deney tasarımlarından oldukça farklıdır. Prosedürde yer alan adımlar sırasıyla:

1. Ön hazırlık safhası: İncelenecek faktörlerin seçilmesi ve seviyelerinin belirlenmesi bu aşamada yapılır. Belirlenen seviyelerde düşük olan negatif (-) işareti ile veya (1) rakamı ile ve yüksek olan seviye pozitif (+) işareti ile veya (2) rakamı ile gösterilir [4].
2. Tam faktöriyel deney matrisinin oluşturulması: Faktör sayısına bağlı olarak tüm mümkün etkileşimler gözönüne alınarak gerçekleştirilir [4].

İki seviyeli iki faktör için mümkün bileşimler Tablo 5.2’de gösterilmiştir [4].

Tablo 5.2. İki seviyeli iki faktör deney tasarım matrisi

Deneme	A	B	AB
1	-	-	+
2	-	+	-
3	+	-	-
4	+	+	+

İki seviyeli iki faktör ele alınırsa (A ve B) matrisin birinci sütununa A ve ikinci sütununa B faktörü yerleştirilir ve aralarında tek etkileşim AB olduğu için üçüncü sütunda yer alır. daha sonra AB etkileşiminin işaretleri A ve B sütunlarındaki işaretler çarpılarak belirlenir [4].

3. İşlem sırası: Rassallaştırma yapılarak işlem sıraları bulunur ve bulunan bu sıraya göre sonuçları ile birlikte işlem kombinasyonu satırlarına taşınır [4].
4. Tüm işlemler bittikten sonra tekrar deneyleri yapılır [4].
5. Deney matrisinin oluşumu ve deneylerin yürütülmesi aşamaları sonucunda deneyin analizi aşamasına geçilir. Deneylerin analizinde amaç tek tek faktörlerin ve etkileşimlerinin sonuç değişken üzerine katkılarını belirlemek ve en uygun faktör kombinasyonunu seçmektir [4].

Tam faktöriyel deneylerde basit etki analizi yanısıra varyans analizi (ANOVA) tekniğinin uygulanması da mümkündür.

Basit etki hesabında; A faktörünün etkisi; faktörün yüksek seviyesinde elde edilen değerlerin ortalamasından düşük seviyesinde elde edilen değerlerin ortalaması

çıkartılarak bulunur. Aynı işlem B ve AB için tekrarlanarak onların etkileri hesaplanabilir.

Varyans analizi yardımıyla yapılan hesaplamalar biraz daha karışık ama daha iyi sonuç verir.

Deney sonuçları sadece grafik olarak veya sadece tek faktörlerin etkisi olarak elde edilebilir. Buna göre değişik faktörlerin anlamını belirlemek için varyans analizi uygulanır [4].

Varyans analizinde kullanılan tanımlar genelde aynı olmasına rağmen deneyde kullanılan faktörlerin sayısı ve seviyelerine bağlı olarak uygulamaları değişir. Tam faktöriyel deneylerde iki veya daha fazla seviyeli ve iki veya daha fazla faktör kullanıldığı için iki koşullu varyans analizi (two way ANOVA) ve çok koşullu varyans analizi (Multi ANOVA) kullanılır.

5.1.1.3. Kesirli Faktöriyel

Tam faktöriyelden farklı olarak kesirli faktöriyel deney tasarımı yönteminin ortaya çıkış sebeplerinin başında maliyet ve zaman kavramları gelmektedir. Tam faktöriyel olarak tasarlanan deneylerde maksimum bilgi kazancı elde edilirken deneylerin yürütülmesi sırasında geçen zaman ve harcanan para da maksimum düzeydedir. Bu şartlar bilgi ve maliyet arasındaki dengeyi en iyi şekilde oluşturmak ve bunun devamını sağlamak zorunluluğunu ortaya çıkarır. Ayrıca klasik faktöriyel tasarımlarda gerek deneyin oluşturulmasında ve gerekse analizinde uzman personel gereklidir.

Kesirli faktöriyel deneylerde amaç işlem sayısını azaltmaktır. Bu da incelenen faktörlerden ödün vermeden incelenen etkileşimlerin sayısını azaltarak sağlanabilir. Diğer yandan imalat sektöründe uygulanan deneylerde ikili etkileşimden daha yüksek derecede (üçlü, dördü ve daha yüksek) önemli etkileşim çok ender gözlenmektedir. Pratik uygulamalarda, genellikle, ana faktörlerin bir bölümü ile ancak bazı ikili etkileşimlerin etkisi önemli bulunmaktadır. Bu nedenle uygulamalarda ikilinin üzerindeki etkileşimleri ihmal etmek (yok varsaymak) mümkündür. Tam faktöriyel kısmında verilen yedi faktörlü deneyde 120 etkileşimden 99'u ikilinin üzerinde etkileşimdir. Bunların ihmal edilmesiyle deneyin boyutunu küçültmek ve böylece deneyin maliyetinde önemli derecede ekonomi sağlamak mümkündür [2].

Kesirli faktöriyel deneyleri kullanmaktaki amaç, ikili ve daha yüksek dereceli etkileşimleri ihmal ederek, deneyleri daha ucuza mal etmektir. Kesirli faktöriyel deneyler, tam faktöriyel deneyler için gereken deneme sayısı n 'nin belli bir kesirini (oranını) içerir. Örneğin, yedi faktörlü ve $n = 128$ deneme gerektiren deney, kesirli deney olarak tasarımlanırsa, $1/2$ kesir için $n = 64$, $1/4$ için $n = 32$, $1/8$ için $n = 16$ ve en aşırı durumda, $1/16$ için ise $n = 8$ deneme gerektirir. Bu kesirli deneylerin her birinde, $n-1$ sayıda etki hesap edilebilir [2].

Faktör sayısı $k = 7$ için en küçük $1/16$ kesirli deney tasarlanabilir. Böyle bir deneyde $n-1 = 8-1 = 7$ etki hesap edilebilir. Faktör sayısı da 7 olduğu için, bu kesirli deneyde etkileşim hesap etmek mümkün değildir. Bu tür kesirli deneylere, deney tasarım literatüründe, aşırı satüre (doymuş) deney denmektedir. Yalnızca ana etkilerin hesabını mümkün kılmasına karşın, bu deneyin maliyeti, $n = 128$ 'lik deneyin maliyetinin ancak $1/16$ 'sıdır [2].

Deneylerin sanayi sektöründe yoğun olarak uygulanmamış olmasının bir nedeni, tam faktöriyel deneylerin yüksek maliyetidir. Kesirli faktöriyel deneyler, maliyet engelinin aşılmasına büyük ölçüde yardımcı olmaktadır [2].

Kesirli deneyler konusu bir sonraki bölümde ayrıntılı olarak anlatılacaktır.

5.1.2. Taguchi Yöntemi

Dünyadaki ekonomik, teknolojik, kültürel gelişmelere ve bunun sonucu olarak pazar yapısında ve insan yaşantısında meydana gelen değişimler tüketici istek ve beklentilerine doğrudan yansımaktadır. Bilinçli ve seçici tüketici toplumun oluşumu ve buna bağlı olarak satıcı pazarından alıcı pazarına dönüşüm, üreticileri pazarda yer alabilme ve sürekli gelişme faaliyetlerine yönlendirmektedir. Şirketlerin veya üretici firmaların son yıllarda önemle üzerinde durdukları kalite kavramı bu değişimlere paralel olarak yenilenmiş ve müşteri mutluluğu esasına bağlı olarak değişmiştir.

II. Dünya Savaşı sonrasında üzerinde önemle durulmaya başlanılan kalite 1964'de Juran tarafından "spesifikasyonlara uygunluk" olarak tanımlanırken, artan istekler ve azalan hammadde kaynakları gözönüne alınarak günümüzde Taguchi tarafından ürün ve prosesteki değişkenliklerin azaltılarak hedef değere yaklaştırılması ve buna bağlı olarak tüketici gereksinimlerinin en optimum seviyede karşılanması olarak yorumlanmıştır. Kalite kavramında günümüze kadar gelen değişimler bu amaca ulaşmaya yönelik olarak teknik ve spesifikasyonları beraberinde getirmiştir.

Şirketlerde kalite uygulamaları başlangıçta kalite kontrol adı altında daha dar kapsamlı iken günümüzde ISO 9000 çerçevesinde tüm sistemi içeren Kalite Güvencesi Sistemi haline gelmiş ve tüm işletme faaliyetlerini kapsayan geniş bir boyut kazanmıştır.

Rekabetin ağır bastığı günümüz koşullarında ürün tasarımı ve üretiminde yüksek kalite yanında zamanında teslimat, servis güvencesi ve düşük maliyet talep edilmektedir. Bütün bu istekler üreticiyi yeni teknikler uygulama ve geliştirmeye zorlamaktadır. Düşük maliyetle yüksek kaliteye ulaşma yolunda tasarım aşamasından üretim aşamasına kadar etkili birçok yöntem mevcuttur. Bu yöntemlerden tasarım aşamasında uygulanan Taguchi Metodu sistem, ürün karakteristikleri, proses parametreleri ve toleransların tasarımının gerçekleştirildiği çevrim dışı kalite kontrol ve çevrim içi kalite kontrol ve deneylerin tasarımı çalışmalarından oluşur.

Kalite anlayışında ve tekniklerinde görülen yenilenme kalite kavramının durağan olmadığını, müşteri beklentileri ve ihtiyaçlarına göre ürün tasarımından imalata kadar uzanarak değiştiğini göstermektedir.

Taguchi metodu, ekonomik ve teknolojik koşullara bağlı olarak değişen tüketici gereksinimlerini karşılamak ve topluluklar oluşturarak rekabeti hızlandıran dünya pazarlarında kalıcı ve büyük pay alabilmek için ürün ve prosesdeki değişkenleri azaltarak belirlenen hedef değere ulaşma yolunda yapılan çalışmaları içerir.

Taguchi performansın, kalitenin ve maliyetin optimizasyonunu sağlayan metottur [11].

Temelde müşteri istekleri doğrultusunda değişikliklerin azaltılarak maliyetin düşürülmesini hedef alan bu tekniğin uygulanmasıyla geleneksel kalite anlayışında yer alan spesifikasyonlara uygunluk yerini hedef değere ulaşma çabasına bırakmıştır.

Taguchi'ye göre kalite ürün müşteriye ulaştıktan sonra ortaya çıkan kayıptır. Her zaman için müşteri istek ve ihtiyaçlarının ilk sırayı aldığı Taguchi anlayışında kalitenin yüksekliği üretilen ürünün fonksiyonlarını istenilen şekilde yerine getirmesi ile orantılıdır aksi takdirde kalite bir kayıptır. Bu kayıp ancak proses ve ürün değişkenlikleri azaltılıp ürün karakteristiklerini hedef değere yaklaştırarak azaltılır.

Taguchi metodu ürün ve/veya proses parametrelerinin ve toleranslarının tasarımını, kalite kayıp fonksiyonunu, çevrim içi ve çevrim dışı kalite kontrol olgularını,

ortogonal stnları kullanarak oluřturulan deneylerin tasarımı ve lm sistemlerini deęerlendirmek iin uygulanan metodolojileri kapsar.

Taguchi'nin kalite felsefesi yedi temele dayanır [8].

1. retilmiř rnn kalitesi topluma verdięi kayıpla llr.
2. Rekabeti kalite ortamında iř hayatında tutunabilmek iin devamlı kalite geliřimi ve dřk maliyet řarttır.
3. Devamlı kalite geliřtirme programı rn performans karakteristiklerinin hedef deęerler etrafında azaltılması ile saęlanır.
4. rn performans deęiřiminin sebep olduęu kayıplar performans karakteristiklerinin hedef deęerden sapmasının karesi ile orantılıdır. (Kayıp Fonksiyonu)
5. retilmiř rn kalitesi ve maliyeti rnn mhendislik tasarımı ve imalat prosesi tarafından kontrol edilir. (evrim İi ve evrim Dıřı Kalite Kontrol)
6. Performans karakteristikleri zerinde rn parametrelerinin lineer olmayan etkisi ortadan kaldırılarak rn performans deęiřimlerini azaltmak gereklidir.
7. Performans deęiřimini azaltan rn veya proses parametreleri gruplarının kombinasyonlarını belirlemek iin istatistiksel deneyler yapılır.

Taguchi'ye gre kaliteyi saęlamak iin yapılan faaliyetler iki blme ayrılmıřtır [2].

1. **evrim Dıřı Kalite Kontrol:** evrim dıřı kalite kontrol, pazar arařtırması ile rn ve retim prosesinin geliřtirilmesi sırasında gerekleřtirilen kalite faaliyetlerini iermektedir. Bu faaliyetler rne doęrudan mdahaleler yerine, retimin bařlamasından nce gerekleřtirilen tasarım alıřmalarıdır.
2. **evrim İi Kalite Kontrol:** evrim ii kalite kontrol rnn imalatı sırasındaki ve imalat sonrası, rneęin servis sırasındaki, kalite faaliyetlerini kapsar. İstatistiksel proses kontrol ve eřitli muayeneler, evrim ii kalite kontrol faaliyetlerindendir.

Deney tasarımı Taguchi'nin kalite sisteminde evrim dıřı kalite kontrol iinde yer almaktadır. (Japonlar, "kontrol" kelimesini olduęundan daha geniř anlamda kullanmaktadırlar [2]. rneęin, Japonya'da toplam kalite ynetimine, toplam kalite kontrol denmektedir.)

Taguchi, çevrim dışı kalite kontrolü,

1. Ürün tasarımı
2. Proses tasarımı

açısından ikiye ayırıyor. Kalite sağlama aşaması olarak, hem ürün tasarımı hem de proses tasarımı için üç kalite aşaması tanımlıyor. Bunlar;

1. Sistem tasarımı
2. Parametre tasarımı
3. Tolerans tasarımı aşamalarıdır.

Taguchi'ye göre, ürünün kalitesini iyileştirmede en belirleyici çalışmaların yapılabileceği aşama, hem ürün hem de proses tasarımı için parametre tasarımı aşamasıdır. Ürün parametre tasarımı, ürün parametrelerinin, malzeme (çelik, lastik, kağıt, plastik vb.) formülasyon değerlerinin, çeşitli boyutlarının, yüzey özellikleri gibi, optimal değerlerinin belirlenmesi anlamına gelmektedir. Parametre tasarımında amaç üründe ortaya çıkabilecek farklılığı (varyasyonu) minimuma indirerek, ürünün hem imalat hem de hayat boyu maliyetini azaltmaktır [2].

Proses parametre tasarımı, kontrol edilebilen imalat proses parametreleri (hat hızı gibi çeşitli hızlar, fırın sıcaklığı gibi çeşitli sıcaklıklar, çeşitli basınçlar ve çeşitli süreler) için optimal seviye ve ayarların belirlenmesi anlamında kullanılmaktadır. Her iki parametre tasarımında da amaç, üründe ve proseste varyasyon (hedef değerlerden farklılık, yani kalitesizlik) yaratan ve kontrol edilemeyen faktörlere karşı kontrol edilebilen faktörlerin (parametrelerin) değerlerini optimal seçerek, ürün ve prosesteki varyasyonu en aza indirmektir. Taguchi bu amaçla yapılan ürün ve proses tasarımına robüst tasarım demektedir. Burada robüst, kontrol edilemeyen faktörlere, örneğin nem, toz, ısı gibi çevre koşullarına, müşteri kullanımındaki farklı uygulamalara ve malzemedeki farklılıklara karşı duyarsız, yani onlardan etkilenmeyen, ürün ve proses anlamında kullanılmaktadır [2].

Ürün ve proses parametre tasarım aşamalarında optimal değerlerinin belirlenmesi ve optimal ayarlarının yapılması gereken çok sayıda faktör vardır. Üstelik bu faktörlerin birçoğu birbirleriyle etkileşim durumundadır. Bu kontrol edilebilen ve edilemeyen faktörlerin ürün ve ürünün performansına olan etkilerinin belirlenebilmesi için en etkin yöntem istatistiksel deney tasarım yöntemidir. Deney tasarımı aracılığıyla

birçok faktörün ürün üzerindeki etkisini ekonomik olarak (düşük maliyetle) belirlemek ve varyasyon yaratan faktörlere karşı önlemleri tasarım aşamasında almak mümkün olmaktadır. Dolayısıyla Taguchi'nin çevrim dışı kalite kontrol sistemi içerisinde en önemli kalite sağlama yöntemi deney tasarımıdır [2].

Taguchi'nin deney tasarım yöntemi ile faktöriyel deney tasarım yöntemleri arasında temel bir fark yoktur. Aralarındaki fark notasyondan kaynaklanır [1]. Üç faktörlü ve her faktörün seviye sayısı iki olan bir deney için Tablo 5.3'te tam faktöriyel tasarım matrisi ve Tablo 5.4'te de Taguchi'nin L8 dizayn matrisi verilmektedir [1].

Tablo 5.3. 2^3 tam faktöriyel tasarım matrisi

A	B	C	AB	AC	BC	ABC
-1	-1	-1	1	1	1	-1
1	-1	-1	-1	-1	1	1
-1	1	-1	-1	1	-1	1
1	1	-1	1	-1	-1	-1
-1	-1	1	1	-1	-1	1
1	-1	1	-1	1	-1	-1
-1	1	1	-1	-1	1	-1
1	1	1	1	1	1	1

Tablo 5.4. Taguchi'nin L8 dizayn matrisi

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

Tablo 5.3 ile Tablo 5.4'te verilen dizaynlar temel olarak aynıdır. Tek fark notasyon ve sıralamadır. Her iki dizaynın da deneme sayısı sekizdir. Matrisler yedi ortogonal

sütundan oluşmaktadır. Her iki dizaynın da üç sütunu ana etki, üç sütunu ikili etkileşim ve bir sütunu da üçlü etkileşimi temsil etmektedir [1].

Bir sonraki bölümde pratikte en çok uygulanan Taguchi'nin L8 ve L16 dizayn matrisleri ayrıntılı olarak anlatılacaktır.

5.1.3. Shainin Yöntemi

Deneylerin tasarımında ilk amaç parametrelerinin seçiminin en doğru şekilde gerçekleştirilmesidir. Deney tasarım yöntemlerindeki çeşitliliğin sebeplerinden birisi deney matrislerini oluştururken kullanılan kabuller ve yöntemlerken diğer bir sebep de parametrelerin seçimi konusunda ortaya çıkan değişik yaklaşımlardır.

Faktörlerin belirlenmesi aşamasında elde edilen büyük miktarda parametre, parametrelerin seçimi safhasında Homing-in yöntemleri ile azaltılır [4].

Homing-in yöntemlerinde belirlenen kritere göre etkili faktörler seçilir. Özel Homing-in yönteminin seçimi değişik kriterlere bağlıdır. Deney tasarımı çalışmalarında deney takımı tarafından çok sayıda parametre belirlenebilir. Bu parametreler deney takımı tarafından indirgenir. Deney takımı parametrelerin azaltılması konusunda belli kritere göre düzenlenmiş Homing-in yöntemleri kullanılır [4].

Homing-in yöntemleri deneyi tasarlayan kişilerin kriterlerine göre belirlenmekle birlikte temelde iki ayrı grup altında ele alınır.

1. Shainin'e göre Homing-in yöntemleri
2. Parametrelerin ağırlıklı seçimi olarak incelenir.

Homing-in uygulamasında kullanılan Shainin yöntemleri sırasıyla;

1. Çoklu analiz tabloları
2. Bileşen araştırma
3. Çiftli karşılaştırmadır.

Deneylerin tasarımında ikinci amaç değişimi azaltmaktır. Bu konuda her tasarım yöntemi değişik prosedürler uygulayarak prosesi ve/veya ürünü optimize etmeye çalışırlar. Shainin parametrelerin optimizasyonu konusuna yaklaşımı diğer yöntemlerden oldukça farklıdır. Shainin deneylerin tasarımında kullanmak üzere teşhis ve değişimin azaltılması konusunda bir takım araçlar geliştirmiştir. Bu

araçların kullanılması sıfıra yakın değişimin sağlanabileceğini ve sıfır hataya ulaşılabilceğini savunmuştur [9].

Shainin'in geliştirmiş olduđu bu araçlar oldukça basit, mantıklı, istatistiksel olarak güçlü ve sonuçların doğruluđu % 100-1000'ler mertebesinde [9].

Taguchi ve Shainin metotları klasik deney metotları ile karşılaştırıldığında aşağıdaki temel noktalar ortaya konulabilir.

- Klasik deney metotları proseslerin sistematik ve öngörülen hedeflere yönelik olarak araştırılmasında çok geniş olanaklar sağlarlar. Bu noktada özellikle yıllardır başarı ile uygulanan faktöriyel deneyler büyük anlam taşımaktadırlar. Bu yöntemler verimli sonuçlar üretme konusunda başarılı olmakla birlikte belirli bir temel bilgiyi ve harcamayı gerektirirler. Deney sayısının azaltılması konusundaki imkanlar oldukça sınırlıdır. Çünkü hata interpolasyonlarında öngörülebilecek riskleri kabul edilebilir seviyede tutmak hedefdir.
- Diğer taraftan Taguchi ve Shainin yöntemleri yakından incelendiğinde bu yöntemlerin öngördüğü basitleştirmenin ancak belirli sınır koşullar altında geçerli olduğunu söylemek mümkündür. Bu sınır koşullar ciddi farklılıklar gösterirler ve bizi yanlış deney sonuçlarına götürebilirler. Bu nedenle ancak iyi bir iletişim ve temel bilgiye sahip olunması durumunda verimli kullanımları söz konusudur.

5.1.4. Genetik Algoritma Yöntemi

Deneylerin tasarımı alanında genetik algoritma özel bir yere sahiptir. Bu tekniğin temeli teknik problemlerin biyolojik kurallar yardımıyla optimize edilerek genetik problem haline dönüştürülmesine dayanır [4].

Bu yöntemin avantajı evrensel olarak uygulanabilirliği ve öngereksinimlerin yer almamasıdır. Deneylerin değerlendirilmesi ve otomatize edilmesi kolaydır. Bu yöntem yüksek olasılıklar sağlarken mutlak optimum bulunabilir. Böylece proses/ürünün tam potansiyeli tanımlanır. Bu tasarım tekniğinde öngereklere ihtiyaç duyulmasa da parametre sayısına bağlı olarak deney sayısının artması nedeniyle parametre azaltma yöntemlerinin uygulanması tavsiye edilir [4].

Genetik algoritma yöntemi uygulamasında maliyet gözönüne alınırsa deney sayısının ve parametrelerin ağırlıklarının belirlenmesi konusunda tecrübeli olmayı gerektirir [4].

6. İKİ SEVİYELİ KESİRLİ FAKTÖRİYEL DENEYLER VE TAGUCHİ DİZAYNLARI

6.1. Giriş

2^k faktöriyel tasarımda faktör sayısı arttıkça, tam bir tekrar için gerekli deneme sayısı oldukça fazla artar. Örneğin 2^6 tasarımının tam bir tekrarı, 64 deneme gerektirir. Bu tasarımda 63 serbestlik derecesinden yalnızca 6 tanesi esas etkiyle, yalnızca 15 tanesi ikili etkileşim ile ilgilidir. Geriye kalan 42 serbestlik derecesi ise üçlü ve daha yüksek etkileşimler ile ilgilidir.

Eğer deney sahibi belli yüksek dereceden etkileşimlerin önemsiz olduğunu varsayabiliyorsa esas etki ve düşük dereceli etkileşimlerle ilgili bilgiler, tam faktöriyel deneyin sadece bir kesimi yürütülerek elde edilebilir. Bu kesirli faktöriyel tasarımlar, ürün tasarımı, proses tasarımı ve proses iyileştirme gibi alanlarda çok kullanılmaktadır.

Kesirli faktöriyel tasarımların kullarımdaki başarıısı, üç ana fikre dayanmaktadır:

1. Farklı pek çok değişken olduğunda, sistem ya da proses, bazı esas etkiler ve düşük dereceden etkileşimler tarafından yönlendirilir.
2. Kesirli deneyler, önemli etkilerin bir ait kümesi olarak geniş tasarımlar haline dönüşebilir.
3. Faktörler ve ilgilenilen etkileşimlerle ilgili etkileri tahmin etmede daha büyük tasarımlar elde etmek için, iki veya daha fazla kesirli faktöriyellerin denemeleri birleştirilebilir.

6.2. İki Seviyeli Üç Faktörlü L8 Dizaynı

İki seviyeli bir deneyde faktör sayısı $k=3$ ise, tam eşlendirme durumunda tüm kombinezonların sayısı $2 \times 2 \times 2 = 8$ 'e eşittir. Bu 8 kombinezonu içeren matrise L8 dizayn matrisi denir. Bu matris Tablo 6.1'de verilmektedir. L8 dizayn matrisinin 7 sütunu vardır. Bunlardan ilk üçü, faktörlerin seviye kombinezonlarını temsil eder.

Diğer sütunlar, faktörler arası etkileşimleri temsil etmektedir. Bu sütunların “eksi” ve “artı” işaretleri, ilk üç sütunun işaretlerinin çarpımı olarak elde edilmiştir. Örneğin, AB etkileşim sütununun işaretleri, A ve B sütunlarının işaretlerinin çarpımı sonucu bulunmuştur [2].

İki seviyeli deneylerde, seviyelerin birine alt seviye, diğerine üst seviye denmektedir. Genellikle, alt seviyeler “-”, üst seviyeler “+” işaretiyle temsil edilir. Bazen, “-” ve “+” işaretlerinin yerine, sırasıyla, “1” ve “2” sayıları kullanılır. Örneğin, Tablo 6.1’de “-” ve “+” işaretlerinin yerine “1” ve “2” sayıları konabilir [2].

Tablo 6.1. L8 dizayn matrisi

Standart Sıra	ETKİLER						
	A	B	C	AB	AC	BC	ABC
1	-	-	-	+	+	+	-
2	-	-	+	+	-	-	+
3	-	+	-	-	+	-	+
4	-	+	+	-	-	+	-
5	+	-	-	-	-	+	+
6	+	-	+	-	+	-	-
7	+	+	-	+	-	-	-
8	+	+	+	+	+	+	+
Sütun No	1	2	3	4	5	6	7

6.3. İki Seviyeli Dört Faktörlü L16 Dizaynı

İki seviyeli bir deneyde faktör sayısı $k=4$ ise, tam eşlendirme durumunda tüm kombinezonların sayısı $2 \times 2 \times 2 \times 2 = 16$ 'ya eşittir. Bu 16 kombinezonu içeren matrise L16 dizayn matrisi denir. L16 dizayn matrisinin 15 sütunu vardır. Bunlardan ilk dördü, faktörlerin ana etkilerini, geri kalanları, faktörler arası etkileşimleri temsil eder ve bu etkilerin hesabı için kullanılır. L16 dizayn matrisi Tablo 6.2’de verilmektedir. L16 dizayn matrisi, pratikte en çok uygulanan dizaynlardan biridir. Bunun nedeni, kesirli deneylerde, 15 faktöre kadar faktörün bir arada incelenmesini mümkün kılmasıdır. Özetle L16 dizaynı, 4 ile 15 arasındaki sayıda faktör için kullanılabilir [2].

Tablo 6.2. L16 dizayn matrisi

Standart Sıra	ETKILER														
	A	B	C	D	AB	AC	AD	BC	BD	CD	ABC	ABD	ACD	BCD	ABCD
1	-	-	-	-	+	+	+	+	+	+	-	-	-	-	+
2	-	-	-	+	+	+	-	+	-	-	-	+	+	+	-
3	-	-	+	-	+	-	+	-	+	-	+	-	+	+	-
4	-	-	+	+	+	-	-	-	-	+	+	+	-	-	+
5	-	+	-	-	-	+	+	-	-	+	+	+	-	+	-
6	-	+	-	+	-	+	-	-	+	-	+	-	+	-	+
7	-	+	+	-	-	-	+	+	-	-	-	+	+	-	+
8	-	+	+	+	-	-	-	+	+	+	-	-	-	+	-
9	+	-	-	-	-	-	-	+	+	+	+	+	+	-	-
10	+	-	-	+	-	-	+	+	-	-	+	-	-	+	+
11	+	-	+	-	-	+	-	-	+	-	-	+	-	+	+
12	+	-	+	+	-	+	+	-	-	+	-	-	+	-	-
13	+	+	-	-	+	-	-	-	-	+	-	-	+	+	+
14	+	+	-	+	+	-	+	-	+	-	-	+	-	-	-
15	+	+	+	-	+	+	-	+	-	-	+	-	-	-	-
16	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Sütun No	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

6.4. Kesirli Deneylerin Tasarımı ve Eşad Kavramı

L8 ve L16 dizayn matrisleri kesirli deneylerin tasarımı için kullanıldığı zaman, bu matrislerin etkileşim sütunlarına diğer faktörleri atamak (yerleştirmek) gerekir. Örneğin, L8 dizayn matrisi 3 ile 7 arasındaki sayıda faktör için kullanılabilir. Faktör sayısı üç olduğu zaman deney tam eşlendirmelidir ve faktörler matrisin ilk üç sütununa atanır. Faktör sayısı üçten fazla (4 ile 7 arasında) ise deney kesirlidir [2].

Tablo 6.3’de faktör sayısına göre faktörlerin L8 matrisine nasıl atanacakları gösterilmiştir. Atanan faktör sayısına (3’ten 7’ye kadar) göre, tabloda ilk sırada faktörler, atandıkları sütunlara göre yer almıştır. Faktör atanmayan etkileşim sütunları, ilk sırada, orijinal adlarını korumaktadır. Daha sonraki sıralar sütunların eşadlarını (mahlas veya İng. alias), yani diğer adlarını belirtmektedir [2].

Tablo 6.3. L8 kesirli deneylerde faktörlerin sütunlara atanması ve eşadlar

Faktör Sayısı	Sütun No						
	1	2	3	4	5	6	7
3	A	B	C	AB	AC	BC	ABC
4	A	B	C	AB CD	AC BD	AD BC	D
5	A BD CE	B AD	C AE	D AB	E AC	BC DE	BE CD
6	A BD CE	B AD CF	C AE BF	D AB EF	E AC DF	F BC DE	AF BE CD
7	A BD CE FG	B AD CF EG	C AE BF DG	D AB EF CG	E AC BG DF	F AG BC DE	G AF BE CD

Kesirli deneylerde eşad durumu ortaya şöyle çıkar. Eşad durumu $k = 4$ faktör için incelensin. Tablo 6.3'e göre, dördüncü faktör D, orijinal adı ABC olan sütun 7'ye atanır. Bu durumda, sütun 7'nin D ve ABC olmak üzere iki adı vardır: $D = ABC$

Bu eşitliğe dizayn jeneratörü denir. Dizayn jeneratörü denkleminin her iki tarafının faktörlerin herbiriyle çarpılmasıyla diğer sütunların eşadları da elde edilir. Eşadların bulunmasına ilişkin örnekler verilmeden önce dizayn matrislerinin bazı özelliklerine değinilecektir [2].

1. Tablo 6.1 ve 6.2'deki L8 ve L16 matrislerinin herhangi bir sütununun kendisiyle çarpımı, I = etkisiz sütunu denen, tamamen "artı" işaretlerden oluşan sütuna eşittir. Örneğin,

$$A \times A = C \times C = AB \times AB = BC \times BC = I$$

2. Ayrıca, herhangi bir sütunun I ile çarpımı kendisine eşittir. Örneğin,

$$A \times I = I \times A = A$$

$$I \times BC = BC \times I = BC$$

Dizayn jeneratörü denklemini faktörlerle tek tek çarparsak sütunların eşadları elde edilir. Jeneratör denklemini A ile çarpılırsa,

$$AD = AABC = IBC = BC$$

Yukarıda, $A \times A = I$; dolayısıyla altıncı sütunun eşadları,

$$BC = AD$$

denklemini ile verilmektedir. Benzer biçimde, dizayn jeneratörü denklemini B ile çarpılınca,

$$BD = BABC = ABBC = AIC = AC$$

bulunur ve sütun 5 için eşadlar elde edilir:

$$AC = BD$$

Son olarak, jeneratör denklemini C ile çarpılınca,

$$CD = CABC = ABI = AB$$

bulunur ve sütun 4'ün eşadları elde edilir,

$$AB = CD$$

Tablo 6.3'de $k=4$ satırında bu sonuçlar gösterilmiştir. Gerçekte, ilk üç sütunun 3'lü etkileşim olarak da eşadları vardır. Ancak 2'li etkileşimden daha yüksek dereceli etkileşimler yok varsayıldığı için daha yüksek derecedeki eşadlar Tablo 6.3'de gösterilmemiştir. Tablo 6.3'de k 'nın diğer değerleri için eşadlar yukarıda açıklandığı biçimde elde edilmiştir [2].

6.5. Deney Çözünürlüğü

Kesirli deneyler, ekonomik olmalarına karşın, atanan faktör sayısı arttıkça etkileri ayırt etme yeteneklerini bir ölçüde kaybederler. Deneylerin etkileri ayırt edebilme özelliklerine deney tasarımı dilinde deney çözünürlüğü (deney rezolüsyonu) denir. Kesirli deneyler çözünürlük derecelerine göre sınıflandırılır. Deney çözünürlükleri Romen rakamları ile gösterilir [6]. Aşağıda, önemli üç deney çözünürlüğü tanımlanmıştır [2].

1. **Çözünürlük V Deneyleri:** Bu deneylerde, hiçbir ana etki veya ikili etkileşim, başka ana etki veya ikili etkileşimle eşad oluşturmaz.

2. **Çözünürlük IV Deneyleri:** Bu deneylerde, hiçbir ana etki başka bir ana etki ile veya ikili etkileşimle eşad oluşturmaz. Ancak, ikili etkileşimler birbirleriyle eşad oluşturabilirler. Tablo 6.4.'de $k=6$, $k=7$ ve $k=8$ için verilen deneylerin çözünürlüğü IV'dir.
3. **Çözünürlük III Deneyleri:** Bu deneylerde, hiçbir ana etki başka bir ana etkiyle eşad oluşturmaz. Ancak, ana etkiler ikili etkileşimlerle eşad oluşturabilir. Tablo 6.4'de faktör sayısı 9 ve daha fazla olan deneylerin çözünürlüğü III'dür [2].

Tablo 6.4. L16 matrisinde faktörlerin sütunlara atanması

Faktör Sayısı	SÜTUN NO														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
4	A	B	C	D	AB	AC	AD	BC	BD	CD	ABC	ABD	ACD	BCD	ABCD
5	A	B	C	D	AB	AC	AD	BC	BD	CD	DE	CE	BE	AE	E
6	A	B	C	D	AB	AC	AD	BC	BD	CD	E	F	ACD	BCD	CF
					CE	BE	BF	AE	AF	EF					DE
					DF										
7	A	B	C	D	AB	AC	AD	AE	AF	AG	E	F	G	BCD	BG
					CE	BE	BF	BC	BD	CD					CF
					DF	DG	CG	FG	EG	EF					DE
8	A	B	C	D	AB	AC	AD	AE	AF	AG	E	F	G	H	AH
					CE	BE	BF	BC	BD	CD					BG
					DF	DG	CG	DH	CH	EF					CF
					GH	FH	EH	FG	EG	BH					DE
9	A	B	C	D	AB	AC	AD	AE	AF	AG	E	F	G	H	I
	HI	GI	FI	EI	CE	BE	BF	BC	BD	CD	DI	CI	BI	AI	AH
					DF	DG	CG	DH	CH	EF					BG
					GH	FH	EH	FG	EG	BH					CF
															DE
10	A	B	C	D	J	AC	AD	AE	AF	AG	E	F	G	H	I
	BJ	AJ	EJ	EI	AB	BE	BF	BC	BD	BH	CJ	CI	BI	AI	AH
	HI	GI	FI	FJ	CE	DG	CG	DH	CH	CD	DI	DJ	HJ	GJ	BG
					DF	FH	EH	FG	EG	EF					CF
					GH					IJ					DE
11	A	B	C	D	J	K	AD	AE	AF	AG	E	F	G	H	I
	BJ	AJ	AK	EI	AB	AC	BF	BC	BD	BH	BK	CI	BI	AI	AH
	C	EK	EJ	FJ	CE	BE	CG	DH	CH	CD	CJ	DJ	DK	FK	BG
	K	GI	FI	GK	DF	DG	EH	FG	EG	EF	DI	HK	HJ	GJ	CF
	HI				GH	FH		JK	IK	IJ					DE

Kesirli deneylerin tasarımında Tablo 6.3 ve Tablo 6.4'ün önemi çok büyüktür. Bu tablolar, kesirli deneylerin tasarımında en önemli basamak olan, faktörlerin sütunlara atanmasında yardımcı olur. Örneğin, belli bir etkileşimin ölçülmesi isteniyorsa, bu etkileşimin bulunduğu sütuna faktör atanmaması gerekir. Diğer bir alternatif de,

faktörlere ad (A, B, C gibi) verirken, dikkatli davranıp ölçülmesi istenen etkileşimin faktör atanmayacak bir sütunda oluşmasını sağlamaktır [2].

Deneyin kesir derecesine göre, kesirli tasarımlara mümkün en yüksek çözünürlüğü uygulamak isteriz. Çözünürlük ne kadar yüksek olursa, hangi etkileşimlerin göz ardı edilebilir olduğu ile ilgili varsayımların sınırlayıcılığı azalır [12].

6.6. Dizayn Jeneratörünün Seçimi

Dizayn jeneratörünün seçimi keyfi değildir. Bu seçim; faktör sayısı ve kullanılacak tasarım çözünürlüğüne bağlı olarak daha iyi eşad yapılar üretecek uygun deneme sayısını sağlayan tasarımın oluşturulmasına bağlıdır [13].

6.7. Yarım Kesirleri Oluşturma

En yüksek çözünürlüklü 2^k tasarımın yarım kesrini oluştururken, ilk önce tam bir 2^{k-1} faktöriyel denemelerinden oluşan temel tasarım yazılır. Daha sonra k. faktörü, artı ve eksilerini en yüksek dereceden etkileşimin ABC...(K-1) artı ve eksi işaretleri ile tanımlayarak ekleriz. Böylece 2^{3-1} kesirli deney, temel tasarım olarak 2^2 tasarımı yazarak ve daha sonra faktör C'yi AB etkileşimine eşitleyerek ekleriz [3]. Bu yaklaşım, Tablo 6.5'de görülmektedir. Dikkat edilirse temel tasarım, doğru deneme sayısı içermekte, ancak sütunlardan biri eksik kalmaktadır. Jeneratör $I = ABC...K$, kayıp K sütunu için çözülür, böylece $K = ABC...(K-1)$, k. faktörün seviyelerinde kullanılmak üzere herbir satır için artı ve eksi işaretlerin çarpımını sunar.

Herhangi bir etkileşim etkisi, k. faktör sütununu üretmede kullanılabilir. Ancak, ABC...(K-1)'den farklı bir etki kullanmak, mümkün en yüksek çözünürlüklü tasarım oluşturmayacaktır.

Tablo 6.5. 2^3 tasarım için iki yarım kesir

Deneme	Tam 2^2 Tasarım (Temel Tasarım)		2^{3-1} , I = ABC			2^{3-1} , I = -ABC		
	A	B	A	B	C = AB	A	B	C = -AB
1	-	-	-	-	+	-	-	-
2	+	-	+	-	-	+	-	+
3	-	+	-	+	-	-	+	+
4	+	+	+	+	+	+	+	-

6.8. Kesirleri Faktöriyellere Dönüştürme

R çözünürlüklü herhangi bir kesirli faktöriyel tasarım, R-1 faktörün herhangi bir alt kümesi için tam bir faktöriyel tasarım (tekrarlı veya tekrarsız) içermektedir. Bu tanım, çok önemli ve kullanışlıdır. Örneğin deneyde çeşitli faktörler varsa ve bu faktörlerden R-1 kadarının önemli olduğuna inanılıyorsa, o halde R çözünürlüklü kesirli deney, uygun bir tasarım olacaktır. Eğer deney sahibi bu inancında haklı çıkıyorsa, R çözünürlüklü kesirli faktöriyel tasarım, R-1 önemli etkinin bulunduğu tam faktöriyel tasarıma dönüşecektir.

6.9. Genel 2^{k-p} Kesirli Faktöriyel Tasarımlar

2^{k-p} adet deneme içeren 2^k kesirli faktöriyel tasarımlara $1/2^p$ kesirli 2^k tasarım denir, ya da daha basitçe 2^{k-p} kesirli faktöriyel tasarım denir. p jeneratör sayısını ve $1/2^p$ tasarımın kesrini belirtir. Böylece $1/4$ kesir 2^{k-2} kesirli tasarım olarak adlandırılır ve $1/8$ 2^{k-3} , $1/16$ 2^{k-4} vs [13]. Bu tasarımlar p adet bağımsız jeneratörün seçilmesini öngörürler. p jeneratörden oluşan deney için belirleyici ilişki seçilir. Bir kesirli faktöriyel için belirleyici ilişki etkisiz sütun I'ya eşit olan tüm sütunlar kümesidir.

Eşad yapı, herbir etki sütununun belirleyici ilişki ile çarpılması ile bulunur. Jeneratör seçerken, ilgilenilen etkilerin birbirleriyle eşad olmamasına önem verilir. Herbir etkinin 2^{p-1} adet eşadı vardır. Faktör sayısı k'nın büyük değerleri için, yüksek dereceden etkileşimlerin (örneğin üçüncü, dördüncü veya daha yüksek) göz ardı edilebilir olduğu varsayılır, ve bu da eşad yapıyı oldukça basitleştirir.

2^{k-p} kesirli faktöriyel deney için p jeneratör öyle seçilmelidir ki, mümkün en iyi eşad yapı elde edilsin. Jeneratörleri seçmede mantıklı yol, 2^{k-p} tasarımın mümkün en yüksek çözünürlüğe sahip olmasıdır. Tablo 6.6, faktör sayısı $k \leq 10$ ve deneme sayısı $n \leq 128$ için 2^{k-p} kesirli faktöriyel deney seçimlerini göstermektedir. Bu tabloda önerilen jeneratörler, ilgili tasarımlar için mümkün en yüksek çözünürlüğü sağlamaktadır. Bu tabloda verilen eşad yapılar esas etkiler, iki-faktör ve üç-faktör etkileşimlerine odaklanmaktadır.

Tablo 6.6. Seçilmiş 2^{k-p} kesirli faktöriyel tasarımlar

Faktör Sayısı, k	Kesir	Deneme Sayısı	Dizayn Jeneratörleri
3	2^{3-1}	4	$C = \pm AB$
4	2^{4-1}	8	$D = \pm ABC$
5	2^{5-1} 2^{5-2}	16 8	$E = \pm ABCD$ $D = \pm AB$ $E = \pm AC$
6	2^{6-1} 2^{6-2} 2^{6-3}	32 16 8	$F = \pm ABCDE$ $E = \pm ABC$ $F = \pm BCD$ $D = \pm AB$ $E = \pm AC$ $F = \pm BC$
7	2^{7-1} 2^{7-2} 2^{7-3} 2^{7-4}	64 32 16 8	$G = \pm ABCDEF$ $F = \pm ABCD$ $G = \pm ABDE$ $E = \pm ABC$ $F = \pm BCD$ $G = \pm ACD$ $D = \pm AB$ $E = \pm AC$ $F = \pm BC$ $G = \pm ABC$
8	2^{8-2} 2^{8-3} 2^{8-4}	64 32 16	$G = \pm ABCD$ $H = \pm ABEF$ $F = \pm ABC$ $G = \pm ABD$ $H = \pm BCDE$ $E = \pm BCD$ $F = \pm ACD$ $G = \pm ABC$ $H = \pm ABD$

Tablo 6.6. (devamı)

9	2^{9-2}	128	H = $\pm$ ACDFG J = $\pm$ BCEFG
	2^{9-3}	64	G = $\pm$ ABCD H = $\pm$ ACEF J = $\pm$ CDEF
	2^{9-4}	32	F = $\pm$ BCDE G = $\pm$ ACDE H = $\pm$ ABDE J = $\pm$ ABCE
	2^{9-5}	16	E = $\pm$ ABC F = $\pm$ BCD G = $\pm$ ACD H = $\pm$ ABD J = $\pm$ ABCD
10	2^{10-3}	128	H = $\pm$ ABCG J = $\pm$ ACDE K = $\pm$ ACDF
	2^{10-4}	64	G = $\pm$ BCDF H = $\pm$ ACDF J = $\pm$ ABDE K = $\pm$ ABCE
	2^{10-5}	32	F = $\pm$ ABCD G = $\pm$ ABCE H = $\pm$ ABDE J = $\pm$ ACDE K = $\pm$ BCDE
	2^{10-6}	16	E = $\pm$ ABC F = $\pm$ BCD G = $\pm$ ACD H = $\pm$ ABD J = $\pm$ ABCD K = $\pm$ AB

6.10. 2^{k-p} Kesirli Faktöriyelerin Analizi

2^{k-p} kesirli faktöriyelerin analizinde kullanılmak üzere çeşitli bilgisayar programları mevcuttur. Tasarım, şu temeller üzerinde şekillenmiştir:

$$l_i = \frac{2 (\text{Kontrast}_i)}{n} = \frac{\text{Kontrast}_i}{(n / 2)} \quad (6.1)$$

i. etkinin tahmini yukarıdaki formülle bulunur. Kontrast_i , i. sütundaki artı ve eksi işaretlerini kullanarak bulunur. $n = 2^{k-p}$ toplam deneme sayısıdır. 2^{k-p} tasarımı sadece $2^{k-p}-1$ etkinin (ve eşadlarının) tahminine imkan tanır.

Genel olarak, AB...K etkisi için kontrast belirlemeyi aşağıdaki gibi yaparız:

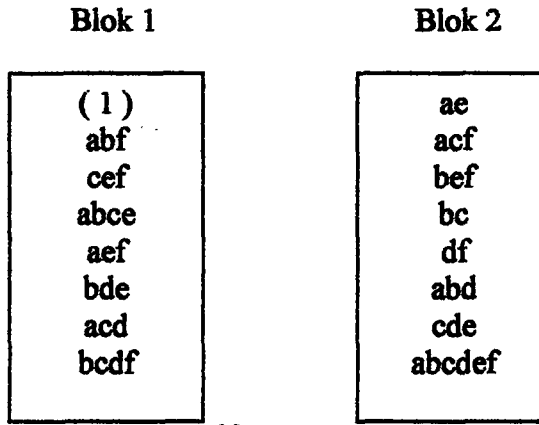
$$\text{Kontrast}_{AB...K} = (a \pm 1) (b \pm 1) ... (k \pm 1) \quad (6.2)$$

Denklem 6.2 açılırsa, çarpım sonucu çıkan "1", (I) ile değiştirilir. Herbir parantez içindeki işaret, eğer o faktör dahil edilmişse eksi, dahil edilmemişse artı olur.

6.11. Kesirli Faktöriyeleri Bloklama

Bazı durumlarda, bir kesirli faktöriyel deneyin gerektirdiği tüm denemelerin hepsi homojen bir yapıda gerçekleşmeyebilir. Bu durumda, kesirli faktöriyel bloklara bölünebilir. Tablo 6.7, çeşitli kesirli faktöriyel tasarımlar için bloklama düzenlemeleri önermektedir. Bu tasarımlar için minimum blok büyüklüğü, sekiz denemedir.

Örnek olarak $I = ABCE = BCDF = ADEF$ belirleyici ilişkiye sahip 2^{6-2} kesirli deneyi göz önüne alalım. Tasarımı herbiri sekiz deneme kombinasyonuna sahip iki blokta yürüttüğümüzü farzedelim. Bloklarla örtüşen etkileşimi seçerken, eşad yapıyı incelediğimizde, sadece üç-faktörlü etkileşimlerden oluşan iki eşad kümesi olduğunu gözlemleriz. Tablo 6.7 ABD'yi (ve eşadlarını) bloklarla örtüşen olarak önermektedir. Buna göre Şekil 6.1'de görülen iki blok oluşur. Dikkat edilirse esas blok, ABD sütunda artı işaretlere sahip deneme kombinasyonlarını içermektedir.



Şekil 6.1. ABD ile örtüşen iki bloklu 2^{6-2} tasarımı

Tablo 6.7. Seçilmiş 2^{k-p} kesirli faktöriyel tasarımlar için bloklar

Faktör Sayısı k	Kesir	Deneme Sayısı	Bloklar
3	2^{3-1}	4	-
4	2^{4-1}	8	-
5	2^{5-1}	16	8'lik 2 blok : AB = CDE
	2^{5-2}	8	-
6	2^{6-1}	32	8'lik 4 blok : BC ; ABF = CDE ; ACF = BDE 16'luk 2 blok : ABF = CDE
	2^{6-2}	16	8'lik 2 blok : ABD = CDE = ACF = BEF
	2^{6-3}	8	-
	2^{7-1}	64	16'luk 4 blok : ABC ; CEF ; CDG 32'lik 2 blok : ACE = AFG 8'lik 4 blok : ACE = AFG ; BCE = BFG ; AB = CDF = DEG 16'luk 2 blok : ACE = AFG 8'lik 2 blok : ABD = CDE = ACF = BEF = BCG = AEF = DFG
7	2^{7-2}	32	
	2^{7-3}	16	
	2^{7-4}	8	-

Tablo 6.7. (devamı)

8	2^{8-2}	64	16'lık 4 blok : CDE = FGH ; ACF ; BDH
	2^{8-3}	32	32'lik 2 blok : CDE = FGH 8'lik 4 blok : ABE = CEF = DEG ; ABF = CFH = DGH ; EH = BCD = ADF = ACG = BFG
	2^{8-4}	16	16'lık 2 blok : ABE = CEF = DEG 8'lik 2 blok : AB = EF = CG = DH
9	2^{9-3}	64	16'lık 4 blok : CFG ; AGJ = BEF = DGH ; ADE = EHJ
	2^{9-4}	32	32'lik 2 blok : CFG 8'lik 4 blok : AEF = BEG = CEH = DEJ ; AB = FG = DEH = CEJ ; CD = BEF = AEG = HJ
	2^{9-5}	16	16'lık 2 blok : AEF = BEG = CEH = DEJ 8'lik 2 blok : AB = CE = FG = DH
10	2^{10-4}	64	16'lık 4 blok : AGJ = CEF = BHJ ; AGK = DEF = BHK ; CD = BFG = AFH = JK
	2^{10-5}	32	32'lik 2 blok : AGJ = CEF = BHJ 8'lik 4 blok : AK = EF = DG = CH = BJ ; AJ = CDE = CFG = DFG = EGH = BK ; AB = CDF = CEG = DEH = FGH = JK
	2^{10-6}	16	16'lık 2 blok : AK = EF = DG = CH = BJ 8'lik 2 blok : AG = CD = BF = EH = JK

6.12. Deneyde Kullanılan L8 Hesap Tablosu

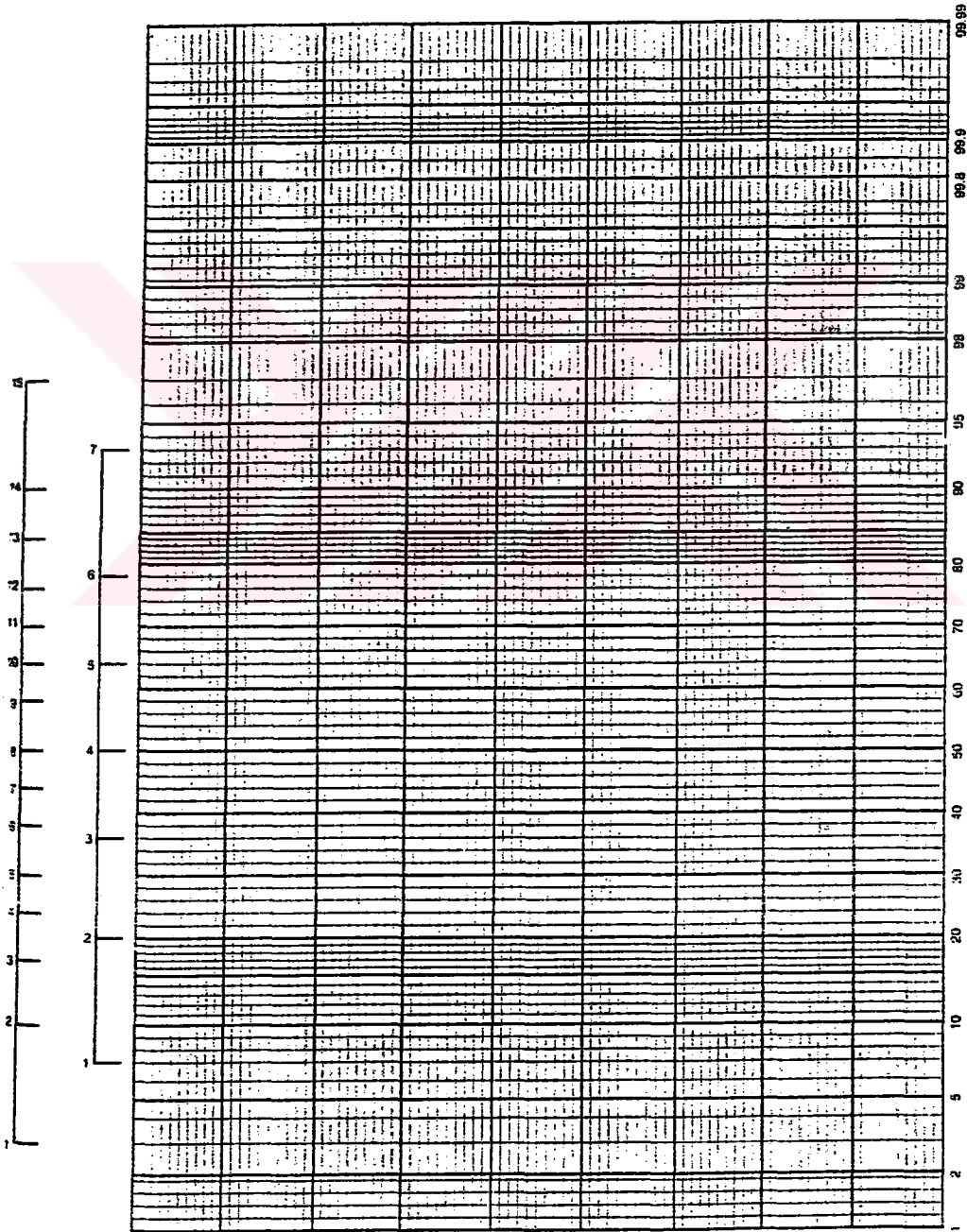
Deney tasarımı eğitiminin son yıllarda yaygınlaşması sonucu, fabrika ortamında hesaplamayı kolaylaştırmak amacıyla bazı tablolar geliştirildi. Bu tablolar arasında, L8 dizayn matrisi için kullanılacak hesap tablosu, boş olarak Tablo 6.8'de verilmektedir [2].

Tablo 6.8. L8 hesap tablosu

Rassal sıra	Standart sıra	Deneme değeri	1. Sütun		2. Sütun		3. Sütun		4. Sütun		5. Sütun		6. Sütun		7. Sütun	
			1	2	1	2	1	2	1	2	1	2	1	2	1	2
	1															
	2															
	3															
	4															
	5															
	6															
	7															
	8															
TOPLAM																
SAYI			4	4	4	4	4	4	4	4	4	4	4	4	4	4
ORTALAMA																
ETKİ																
SIRA																

6.13. Normal Olasılık Grafiği

Etkilerden hangilerinin istatistiksel açıdan önemli (statistically significant) olduğuna karar vermek için normal olasılık grafiği yöntemi kullanılır. Boş bir normal olasılık grafik (NOG) kağıdı Şekil 6.2’de verilmiştir. Kullanıcıya kolaylık sağlaması için normal olasılık grafiğinin dikey eksen L8 ve L16 deneyleri için ölçeklendirilmiştir [2]. L8 deneylerinde 7, L16 deneylerinde 15 etki değeri sıraya konacak ve o sayıda nokta grafiğe işlenecektir.



Şekil 6.2. Normal olasılık grafik kağıdı

6.14. Deneyde Kullanılan Hesap Yönteminin Özeti

1. Deneyin uygulaması bitirilip veriler elde edildikten sonra, veriyi L8 (veya L16) hesap tablosunun deneme değeri sütununa giriniz.
2. Deneme değeri sütunundaki her değeri, aynı satırdaki boş hanelere işleyiniz.
3. Her sütunun toplamını alarak “toplam” satırına işleyiniz.
4. Toplamları, kendi sütunlarındaki “sayılara” bölerek ortalamaları bulunuz ve “ortalama” satırına işleyiniz.
5. Her etki sütunundaki seviye 2 ortalamasından seviye 1 ortalamasını çıkartarak etki değerini bulunuz ve “etki” satırına işleyiniz.
6. Etki değerlerini cebirsel sıraya koyunuz; en eksi değere “1”, en artı değere “7” (veya 15) sıra numarasını veriniz. (Eğer etki değerleri eşit ise ortalama sıra numarası kullanma yöntemi uygulanabilir. Bu yönteme göre herhangi iki sütunun etki değerleri aynı ise ve bunlar x. ve y. sırada ise bunlara sıra numarası olarak x ve y’nin ortalaması $(x + y) / 2$ değeri verilir.) Sıra numaralarını “sıra” satırına giriniz.
7. Sıra numaraları için dikey eksen, etki değerleri için yatay eksen kullanarak normal olasılık grafiğini oluşturunuz. Noktaları grafiğe işleyiniz.
8. Noktaların sifıra yakınlık bakımından en küçük yaklaşık yarısını kullanarak, bu noktalara en yakın geçecek biçimde bir doğru çiziniz. (Doğrunun çiziminde ve önemli etkilerin belirlenmesinde her zaman bir miktar öznellik rol oynamaktadır.)
9. Noktaların doğruya göre konumlarına bakarak, önemli etkileri belirleyiniz. Önemli etkilere karşı düşen noktalar, yukarıda doğrunun sağında, aşağıda doğrunun solunda yer alır. Bu etkilerin değeri, ya aşırı artı, ya da aşırı eksi değerlerdir.
10. Belirlediğiniz etkiler arasında “etkileşimler” varsa, etkileşim grafikleri çizerek etkileşimleri yorumlayınız. (Etkileşim grafiğini çizmek için önce etkileşimi oluşturan faktörler seviyelerine göre gruplara ayrılır ve her grubun ortalaması bulunur. Ortalama değerler etkileşim grafiğini çizmek için kullanılır. Sonuç değişkeni Y’nin ortalama değerleri, etkileşimi oluşturan faktörlerden herhangi birinin seviyelerine göre grafiğe işlenir ve diğer faktörün seviyelerine göre

etkileşim doğruları çizilir. Eğer doğrular paralel değil ise etkileşim önemli aksi durumda önemsizdir. Buna ilaveten, doğrular zıt eğilimli ise etkileşimin değeri negatif aksi durumda pozitiftir.)

Not: İstatistiksel açıdan, uygulama sırasının standart sırada değil, rassal olarak belirlenmesi gerekir. Uygulama sırasının rassallaştırılmasının amacı kontrol edilemeyen, ancak deney sırasında kendiliğinden değişen faktörlerin sonuç değişkeni Y'ye olası etkilerini elimine etmektir. Uygulamalarda mümkün oldukça deneyin uygulama sırasını rassallaştırmaya çalışılmalıdır [2].

6.15. İki Katlı Dizayn

L8 dizaynı faktör sayısının 3'ten fazla olduğu durumlarda her zaman kesin sonuç vermez. Bazen deney üzerinde gerçekte önemli bir etkisi olmayan faktör, deney sonrasında çizilen normal olasılık grafiğine göre etkili gözükabilir. Böyle durumlarda karışıklığı önlemek için yeni bir L8 deneyi yapılır. Bu deney ilk L8 deneyi ile birleştirilir. İki deneyin birleştirilmesi ile meydana gelen bu yeni forma iki katlı dizayn denir [1].

Her iki L8 deneyinin de çözünürlük dereceleri üç iken ikisinin birleştirilmesi ile meydana gelen yeni deneyin çözünürlük derecesi dördür. Oluşturulan yeni L8 dizayn matrisi ile ilk L8 dizayn matrisi Tablo 6.9'da verilmektedir. Blok sütununda 1 rakamı olan matris ilk L8 dizayn matrisi, blok sütununda 2 rakamı olan matris ise ikinci L8 dizayn matrisidir [1].

Tablo 6.9. İki katlı dizayn matrisi

Standart Sıra	Sütun No							
	Blok	1	2	3	4	5	6	7
1	1	1	1	1	2	2	2	1
2	1	1	1	2	2	1	1	2
3	1	1	2	1	1	2	1	2
4	1	1	2	2	1	1	2	1
5	1	2	1	1	1	1	2	2
6	1	2	1	2	1	2	1	1
7	1	2	2	1	2	1	1	1
8	1	2	2	2	2	2	2	2
9	2	2	2	2	1	1	1	2
10	2	2	2	1	1	2	2	1
11	2	2	1	2	2	1	2	1
12	2	2	1	1	2	2	1	2
13	2	1	2	2	2	2	1	1
14	2	1	2	1	2	1	2	2
15	2	1	1	2	1	2	2	2
16	2	1	1	1	1	1	1	1

Tablo 6.9'dan da görüldüğü gibi ikinci L8 dizayn matrisi, ilk L8 dizayn matrisindeki 1'lerin yerine 2, 2'lerin yerine ise 1 yazarak oluşturulmuştur [1].

Kullanılan hesap yöntemindeki tek fark L8 hesap tablosundan kaynaklanır. Tablo 6.8'de verilen L8 hesap tablosunda 1. sütun yeni hesap tablosunda 2. sütun, 2. sütun ise 1. sütun olur. Her iki deney için toplam 16 etki bulunur. Bu etkiler Tablo 6.10'da verilen iki katlı dizayn etki hesap tablosunun ilgili sütunlarına (deney 1 etkisi sütunu ve deney 2 etkisi sütunu) yazılır. Ana etki ve etkileşim etkisi sütunları ise Tablo 6.10'da verilen formüller kullanılarak doldurulur [1].

Tablo 6.10. İki katlı dizayn etki hesap tablosu

Sütun	Deney 1 Etkisi E_1	Deney 2 Etkisi E_2	Ana Etki $(E_1 + E_2) / 2$	Etkileşim Etkisi $(E_1 - E_2) / 2$
Y				
1				
2				
3				
4				
5				
6				
7				

Y sütunu için E_1 ve E_2 değerleri L8 hesap tablolarındaki ortalama Y değerleridir.

Tablo 6.10'da yer alan ana etki ve etkileşim etkisi sütunlarına beş, altı ve yedi faktörlü durumlarda hangi faktörlerin atanacağı Tablo 6.11'de verilmiştir [1].

Tablo 6.11. İki katlı dizayn etki hesap tablosunda faktörlerin ana etki ve etkileşim etkisi sütunlarına atanması

Sütun	Faktör Sayısı					
	5		6		7	
	Etki Sütunu		Etki Sütunu		Etki Sütunu	
	Ana	Etkileşim	Ana	Etkileşim	Ana	Etkileşim
Y	ortalama	blok	ortalama	blok	ortalama	blok
1	A	BD = CE	A	BD = CE	A	BD = CE = FG
2	B	AD	B	AD = CF	B	AD = CF = EG
3	C	AE	C	AE = BF	C	AE = BF = DG
4	D	AB	D	AB = EF	D	AB = CG = EF
5	E	AC	E	AC = DF	E	AC = BG = DF
6	-	BC = DE	F	BC = DE	F	AG = BC = DE
7	-	BE = CD	-	AF = BE = CD	G	AF = BE = CD

6.16. Normal Olasılık Grafiğinin Kuramsal Temeli

Önemli etkileri normal olasılık grafiği aracılığıyla saptama yöntemini Daniel önermiştir. Deney tasarımında temel varsayım, gözlenen Y değerlerinin olasılık dağılımının normal olduğudur. Bu nedenle, hesap tablolarında her sütun için hesaplanan etki değerlerinin dağılımı da normal varsayılmaktadır. Önemli etkiler için normal dağılımın ortalaması sıfırdan farklı, eksi veya artı bir değer olacaktır. Önemsiz etkiler için ise gözlenen yalnızca ortam gürültüsüdür; ortalama sıfırdır.

Daniel'in önerdiği yöntemle göre, sıfır ortalamalı normal dağılımdan gelen denemeler, sıraya konduktan sonra, normal olasılık kağıdına işlenirse, elde edilen noktaların bir doğru üzerinde yer almaları gerekir. Bu doğrunun eğimi, normal dağılımın standart sapmasıyla orantılıdır. Önemli, yani ortalaması sıfırdan farklı etkilere karşı düşen noktalar doğrunun üzerinde yer almazlar.

Normal olasılık grafiği oluşturulurken, etki değerleri yatay eksene işlenir. Olasılık ekseni olan dikey eksene, etkilerin sıra numarasına karşı düşen olasılık değeri işlenir. Olasılık değerinin formülü $(i - 0.5) / m$ şeklindedir. Formülde, i = sıra numarası, m = toplam etki sayısı'dır.

L8 ve L16 dizaynlarında toplam etki sırasıyla 7 ve 15'tir. Deneyde kullanılan normal olasılık grafik kağıtlarında L8 ve L16 dizaynları için olasılık değerleri önceden dikey eksene işaretlenmiştir [2].

6.17. Deney Tasarımı İçin Temel Basamaklar

Deney tasarımı çalışmaları, toplam kalite yönetimi çerçevesinde yapılmalıdır. Özellikle, sonuçları birden fazla bölümü etkileyecek deney tasarımı çalışmaları için bir takım oluşturmak yerinde olur. Deney tasarımı takımı, genel çalışma yöntemi olarak Deming'in "Planla $\Rightarrow$ Yap $\Rightarrow$ Doğrula $\Rightarrow$ Karar Ver" çemberini benimser. Takım üyelerinin, "yedi basit araç" gibi sürekli geliştirme yöntemleri konusunda eğitim görmüş olmalarının, çalışmanın etkinliğine ve doğruluğuna katkısı büyük olacaktır.

Aşağıda deney tasarımı uygulaması için temel basamaklar verilmektedir [2]:

1. Kalite problemini tanımlayınız. Bunun için öncelikle, deney sırasında çıktı olarak ölçülecek ve ürünün kalite özelliklerini temsil eden sonuç değişkeni veya

değişkenlerini tanımlamak gerekir. Sonuç değişkenlerinin ölçülebilir (sayısal yada nicel) olması gerekir. Ürünün, boya ve yüzey parlaklığı gibi kozmetik yada nitel özelliklerini temsil eden değişkenler için, birkaç kişilik bir jüri oluşturarak ürünün puanlaması önerilir.

2. Sonuç değişken(ler)ini etkilemesi muhtemel tüm faktörlerin bir listesini çıkarınız. Bu listenin tam ve doğru bir şekilde oluşturulabilmesi için ürüne ve sürece ilişkin bir iş akış çizelgesi (flow chart) hazırlanması ve ilgili kişi ve bölümlerin görüşlerinin alınması çok yararlı olur. Ayrıca belirlenen faktörlerin birbirleriyle olan ilişkilerinin net olarak görülebilmesi ve sınıflandırılabilmesi için bir balık kılçığı çizelgesi oluşturulabilir.
3. Yukarıda belirlediğiniz faktörlerden hangilerinin deneyde inceleneceğine karar veriniz. Deneyin içereceği faktörlerin kontrol edilebilir, yani değerleri değiştirilebilir olması gerekir. Kontrol edilebilen faktörlerin değerleri birbirlerinden bağımsız olarak değiştirilebilmelidir. Deneye dahil edilmeyen faktörlerin, neden dahil edilmediklerini belirtiniz.
4. Basamak 3'te belirlenen faktörlerden hangilerinin arasında etkileşimlerin olası olduğuna ve deneyde inceleneceğine, takım olarak karar veriniz.
5. Deneye dahil edilen faktör ve incelenecek etkileşim sayısına bakarak, L8 veya L16 gibi bir deney dizaynı seçiniz.
6. Her faktör için seviye değerleri belirleyiniz. Alt ve üst seviye arasındaki aralık çok küçük olursa, faktörün etkisi görülmeyebilir. Aralık çok büyük olursa, faktör deneye hakim duruma gelir ve diğer faktörlerin etkilerinin saptanması güçleşebilir. Deneye dahil edilmeyen faktörlerin değerleri deney sırasında sabit kalacaktır.
7. Faktörleri kesirli deneyler için verilen eşad tablolarını kullanarak, L8 veya L16 dizayn matrisinin sütunlarına atayınız.
8. Faktörlerin atandığı dizayn matrisini kullanarak, uygulanacak faktör seviye kombinezonlarını belirleyiniz ve reçete tablosunu oluşturunuz.
9. Bir rassallaştırma yöntemi (torbadan sayı çekme gibi) aracılığı ile deneyin uygulama sırasını, mümkün olduğunca rassallaştırınız.

10. Deneyin uygulama planını (personel seçimi, zaman, ölçme yöntemi gibi) yapınız. Basamak 8 ve 9'da belirlediğiniz faktör seviye kombinezonlarını ve sırayı açıklayarak uygulama grubuna veriniz. (Büyük bir olasılıkla, uygulama grubunda takım üyelerinin bir bölümü yer alacaktır.)
11. Deneyi uygulayarak sonuç değişkeni Y'nin değerlerini alınız.
12. Hesap tablolarını doldurunuz ve etki değerlerini hesaplayınız.
13. NOG'u (normal olasılık grafiğini) oluşturarak hangi etkilerin istatistiksel açıdan önemli olduğuna karar veriniz.
14. Basamak 13'te önemli etkileşimler bulmuşsanız, etkileşim grafiklerini çizerek bunları yorumlayınız.
15. Önemli etki ve etkileşimlere bağlı olarak, optimal ürün ve proses seviyelerini belirleyiniz. Sonuç değişkeni Y'nin beklenen değerini hesaplayınız. (Sonuç değişkeni Y'nin beklenen değeri, deney sırasında gözlenmiş olan Y'nin ortalama değerine, önemli etki ve etkileşimlerin değerlerinin yarısı eklenerek yada çıkarılarak bulunur. Önemli etki ve etkileşimlerin optimal seviyeleri alt seviye ise bu etki ve etkileşimlerin değerlerinin yarısı çıkartılır. Aksi durumda değerlerin yarısı eklenir.)
16. Basamak 15'te belirlenen seviyelerde deneyler uygulayarak sonuçları doğrulayınız.

6.18. Sonuç Değişkeninin Hedef Değeri

Taguchi, uygulamadaki problemleri hedefin türüne göre üçe ayırmıştır [2].

1. **En Küçük – En İyi:** Bu tür problemlerde sonuç değişkeni Y'nin hedef değeri sıfırdır.
2. **En Büyük – En İyi:** Bu durumda Y'nin hedef değeri sonsuzdur.
3. **Hedef Değer – En İyi:** Bu tür problemlerde Y için belli bir hedef değer (örneğin ürün boyutları gibi) verilmiştir.

7. DENEY UYGULAMASI - 1

7.1. Giriş

Bu uygulamada, akış tipi çizelgeleme problemlerinin genetik algoritma ile çözüm performansının artırılmasında deney tasarımı yapılmıştır. Öncelikle sonuç değişkeni ve bu değişken üzerinde etkili olduğu düşünülen faktörler ve seviyeleri belirlenmiştir. Deney tasarım edildikten sonra deney gerçekleştirilmiştir. Daha sonra elde edilen veriler kullanılarak her faktörün en iyi seviyeleri belirlenmiştir.

7.2. Deney Notasyonu

Sonuç değişkeni Y ile temsil edilir. Deney tasarımı notasyonuna göre, sonuç değişkenini etkilemesi olası faktörler alfabenin büyük harfleriyle ($A, B, C, \dots$) temsil edilir. Diğer yandan, faktörlerin seviye sayıları küçük harflerle ($a = A$ 'nın seviye sayısı, $b = B$ 'nin seviye sayısı gibi) gösterilir. Ayrıca, $k =$ faktör sayısı, $n =$ deneme sayısı ve $r =$ tekrar sayısı olarak tanımlanır.

7.3. Problemin Tanımı

Bu deney tasarımında, akış tipi çizelgeleme problemlerinin genetik algoritma ile çözüm performansının artırılması göz önüne alınmıştır.

7.3.1. Sonuç Değişkeninin Belirlenmesi

Bu deney için sonuç değişkeni olarak iterasyon sayısı seçilmiştir. Sonuç değişkeni göz önüne alındığında amaç en iyi çözüme ulaşmak için gereken iterasyon sayısını minimuma getirmektir.

7.3.2. Faktörlerin ve Seviyelerinin Belirlenmesi

İterasyon sayısını etkileyen ana faktörler aşağıda sıralanmıştır:

1. Başlangıç popülasyonu (A)

2. Üreme operatörü (B)
3. Çaprazlama operatörü (C)
4. Mutasyon operatörü (D)
5. Çaprazlama oranı (E)
6. Mutasyon oranı (F)

Herbir faktör ikişer seviyeye sahiptir ve bu seviyeler Tablo 7.1’de gösterilmiştir.

Tablo 7.1. Kontrol edilebilir faktörler ve seviyeleri

	Başlangıç Popülasyonu	Üreme Operatörü	Çaprazlama Operatörü	Mutasyon Operatörü	Çaprazlama Oranı	Mutasyon Oranı
Seviye 1	20	Kısmi Yapay Seçim	Sıralı Çaprazlama (OX)	Ters Mutasyon	25	25
Seviye 2	40	Akış Zamanlı Rulet Çemberi	Pozisyona Dayalı Çaprazlama (PBX)	Keyfi Üç İş Değiştirme	75	75

7.4. Deneyin Tasarımı

Bu deneyde iterasyon sayısını etkileyen toplam altı faktör bulunmaktadır. Her faktör ikişer seviyeye sahiptir. 2^k faktöriyel tasarımda faktör sayısı arttıkça tam bir tekrar için gerekli deneme sayısı da büyük ölçüde artmaktadır. 2^6 faktöriyel tasarımın tam bir tekrarı için 64 deneme gereklidir. Bu tasarımda hesap edilen 63 etkinin 6 tanesi ana etki ve 15 tanesi ikili etkileşimdir. Geri kalan 42 etki üçlü veya daha yüksek dereceli etkileşimdir.

Bu deney için ikili etkileşimden daha yüksek dereceli etkileşimlerin göz ardı edilebilir olduğu öğrenilmiştir. Ayrıca eldeki kaynakların tam bir tekrar için yetersiz kalacağı vurgulanmıştır. Bu nedenle tam bir 2^6 faktöriyel tasarım yerine, bu tasarımın 1/8 kesrinin yürütülmesine karar verilmiştir. Bu da $2^{6-3} = 2^3$ deneme yapılması anlamına gelmektedir.

Deneyde herbir denemede 30 tekrar yapılmıştır. L8 dizayn matrisi kullanılmıştır. Bu matris tablo 7.2’de verilmektedir.

Tablo 7.2. Deneyde kullanılan L8 dizayn matrisi

Standart Sıra	ETKİLER						
	A	B	C	D	E	F	—
1	1	1	1	2	2	2	1
2	1	1	2	2	1	1	2
3	1	2	1	1	2	1	2
4	1	2	2	1	1	2	1
5	2	1	1	1	1	2	2
6	2	1	2	1	2	1	1
7	2	2	1	2	1	1	1
8	2	2	2	2	2	2	2
Sütun No	1	2	3	4	5	6	7

Deney kesirli olduğu için eşad durumu vardır. Tablo 7.3’de her faktörün eşadları gösterilmiştir.

Tablo 7.3. Eşad tablosu

Faktör Sayısı	Sütun No						
	1	2	3	4	5	6	7
6	A	B	C	D	E	F	AF
	BD	AD	AE	AB	AC	BC	BE
	CE	CF	BF	EF	DF	DE	CD

İkili etkileşimden daha yüksek dereceli etkileşimler yok varsayıldığı için daha yüksek derecedeki eşadlar Tablo 7.3’te gösterilmemiştir.

7.5. Deneyin Yapılışı

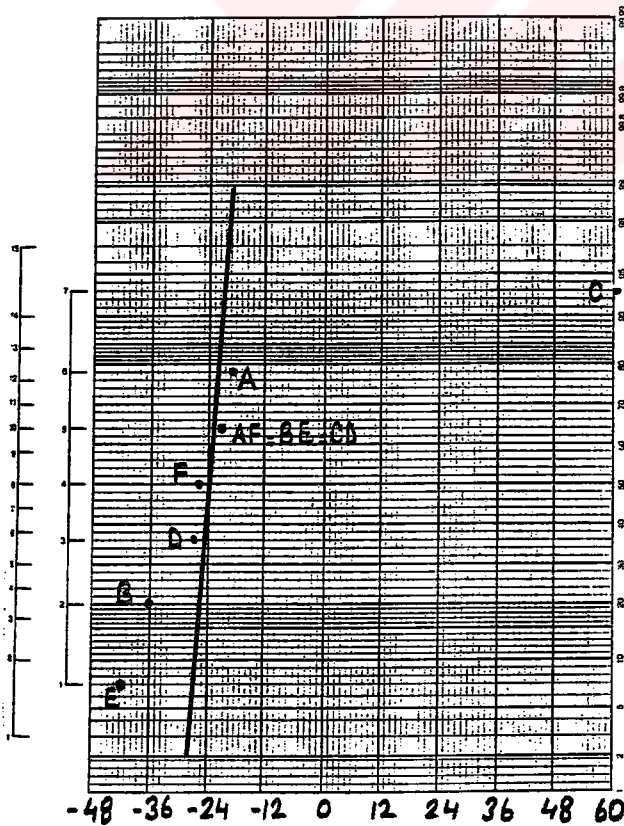
Deneyler Turbo Paskal programlama dilinde hazırlanan Genetik algoritma programı yardımı ile bilgisayarda gerçekleştirilmiştir. Elde edilen veriler ayrıntılı olarak bilgisayar çıktıları şeklinde ekte verilmiştir. Deneyde elde edilen veriler Tablo 7.4’te özetlenmiştir.

Tablo 7.4. Deney verileri

Deneme	A	B	C	D	E	F	İterasyon Ortalaması
1	1	1	1	2	2	2	40.37
2	1	1	2	2	1	1	148.73
3	1	2	1	1	2	1	36.97
4	1	2	2	1	1	2	135.27
5	2	1	1	1	1	2	66.80
6	2	1	2	1	2	1	135.70
7	2	2	1	2	1	1	51.47
8	2	2	2	2	2	2	23.87

7.6. Veri Analizi

Deneyde verilerin analizi L8 etki hesap tablosu yardımıyla yapılmıştır. Tablo 7.4'te belirtilen değerler, L8 etki hesap tablosuna işlenir ve etkiler hesaplanarak sıraya konur. Sonuçlar Tablo 7.5'te verilmiştir. Daha sonra, önemli etkilerin belirlenmesi için normal olasılık grafiği oluşturulur. NOG, Şekil 7.1 olarak verilmiştir.



Şekil 7.1. Normal olasılık grafiği

Tablo 7.5. L8 etki hesap tablosu

Standart sıra	Deneme değeri	1. Sütun		2. Sütun		3. Sütun		4. Sütun		5. Sütun		6. Sütun		7. Sütun	
		A		B		C		D		E		F		AF = BE = CD	
		1	2	1	2	1	2	1	2	1	2	1	2	1	2
1	40.37	40.37		40.37		40.37		40.37		40.37		40.37		40.37	
2	148.73	148.73		148.73		148.73		148.73		148.73		148.73		148.73	
3	36.97	36.97		36.97		36.97		36.97		36.97		36.97		36.97	
4	135.27	135.27		135.27		135.27		135.27		135.27		135.27		135.27	
5	66.80	66.80		66.80		66.80		66.80		66.80		66.80		66.80	
6	135.70	135.70		135.70		135.70		135.70		135.70		135.70		135.70	
7	51.47	51.47		51.47		51.47		51.47		51.47		51.47		51.47	
8	23.87	23.87		23.87		23.87		23.87		23.87		23.87		23.87	
TOPLAM	639.18	361.34	277.84	391.60	247.58	195.61	443.57	374.74	264.44	402.27	236.91	372.87	266.31	362.81	276.37
SAYI	8	4	4	4	4	4	4	4	4	4	4	4	4	4	4
ORTALAMA	79.90	90.34	69.46	97.90	61.90	48.90	110.89	93.69	66.11	100.57	59.23	93.22	66.58	90.70	69.09
ETKİ		-20.88		-36.00		61.99		-27.58		-41.34		-26.64		-21.61	
SIRA		6		2		7		3		1		4		5	

Şekil 7.1 sonuç değişkeni Y açısından B, C ve E faktörlerinin önemli olduğunu göstermektedir.

Herhangi bir faktörün etkisi eşadından dolayı önemli gözükmemektedir yani herhangi bir etkileşimin etkisi ile karışmamaktadır. Bu yüzden iki katlı dizaynı uygulamaya gerek yoktur.

7.7. Deney Sonucu

B (üreme operatörü), C (çaprazlama operatörü) ve E (çaprazlama oranı) faktörlerinin etki değerleri diğerlerinden daha yüksektir. Bu yüzden iterasyon sayısına etkileri çok daha fazla olmaktadır. Problemimiz “en küçük – en iyi” türünde bir problem olduğundan, bu faktörlerin seviyeleri Y’nin ortalama değerini azaltacak biçimde seçilmelidir. Bu değerler şöyledir:

$B_2 = \text{Akış Zamanlı Rulet Çemberi}$, $C_1 = \text{Sıralı Çaprazlama}$, $E_2 = 75$

Diğer yandan, A, D ve F faktörleri Y açısından önemli değildir. Bu faktörlerin seviyeleri bir sonraki bölümdeki deney uygulaması için, amacımız olan en iyi çözüme ulaşmak için gereken iterasyon sayısını azaltmak doğrultusunda, şu şekilde seçilmelidir:

$A_2 = 40$, $D_2 = \text{Keyfi Üç İş Değiştirme}$, $F_2 = 75$

8. DENEY UYGULAMASI – 2

8.1. Giriş

Bu uygulama, bir önceki bölümdeki uygulamanın devamıdır. Bu bölümde bir önceki bölümde elde edilen sonuçlar kullanılarak yeni bir deney tasarlanmıştır. Yeni deneyde bir önceki bölümdeki sayısal seviyelere sahip faktörler kullanılmıştır. Sonuç değişkeni değişmemiştir ama sayısal seviyelere sahip faktörler için bir önceki bölümde elde edilen sonuçlar doğrultusunda seviye aralıkları daraltılmıştır. Bu uygulamanın amacı daha hassas sonuçlar elde etmektir. Deney tasarım edildikten sonra deney gerçekleştirilmiştir. Daha sonra elde edilen veriler kullanılarak her faktörün en iyi seviyeleri belirlenmiştir.

8.2. Sonuç Değişkeninin Belirlenmesi

Bu deney için sonuç değişkeni olarak iterasyon sayısı seçilmiştir. Sonuç değişkeni göz önüne alındığında amaç en iyi çözüme ulaşmak için gereken iterasyon sayısını minimuma getirmektir.

8.3. Faktörlerin ve Seviyelerinin Belirlenmesi

Üç faktör kontrol edilebilir faktör olarak tanımlanmıştır. Bu faktörler başlangıç popülasyonu (A) , çaprazlama oranı (B) ve mutasyon oranıdır (C).

Herbir faktör ikişer seviyeye sahiptir ve bu seviyeler Tablo 8.1’de gösterilmiştir.

Tablo 8.1. Kontrol edilebilir faktörler ve seviyeleri

	Başlangıç Popülasyonu	Çaprazlama Oranı	Mutasyon Oranı
Seviye 1	30	50	50
Seviye 2	50	100	100

8.4. Deneyin Tasarımı

Bu deneyde iterasyon sayısını etkileyen toplam üç faktör bulunmaktadır. Her faktör ikişer seviyeye sahiptir.

Deneyde herbir denemede 30 tekrar yapılmıştır. L8 dizayn matrisi kullanılmıştır. Bu matris tablo 8.2’de verilmektedir. Deney üç faktöre ve L8 dizayn matrisine sahip olduğu için tam faktöriyeldir.

Tablo 8.2. Deneyde kullanılan L8 dizayn matrisi

Standart Sıra	ETKİLER						
	A	B	C	AB	AC	BC	ABC
1	1	1	1	2	2	2	1
2	1	1	2	2	1	1	2
3	1	2	1	1	2	1	2
4	1	2	2	1	1	2	1
5	2	1	1	1	1	2	2
6	2	1	2	1	2	1	1
7	2	2	1	2	1	1	1
8	2	2	2	2	2	2	2
Sütun No	1	2	3	4	5	6	7

8.5. Deneyin Yapılışı

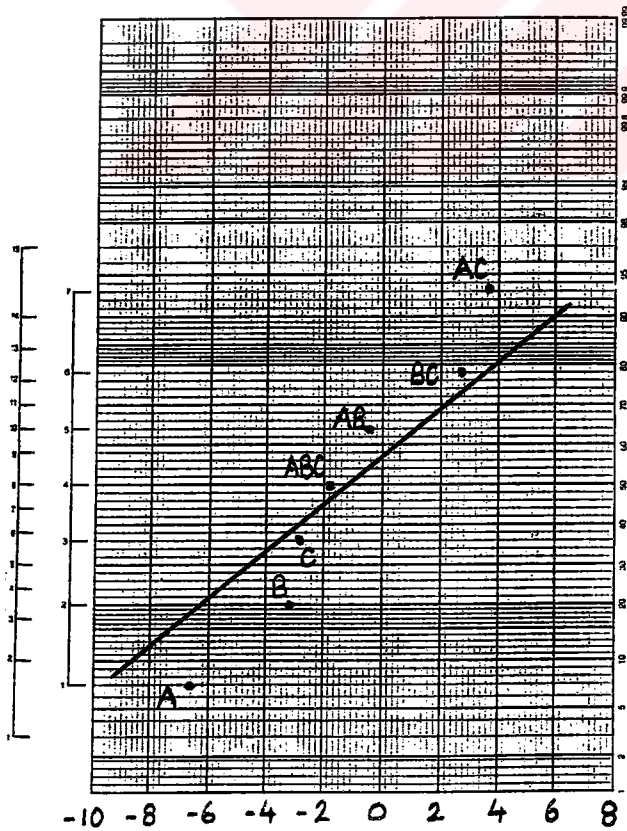
Deneyler Turbo Paskal programlama dilinde hazırlanan Genetik algoritma programı yardımı ile bilgisayarda gerçekleştirilmiştir. Elde edilen veriler ayrıntılı olarak bilgisayar çıktıları şeklinde ekte verilmiştir. Deneyde elde edilen veriler Tablo 8.3’te özetlenmiştir.

Tablo 8.3. Deney verileri

Deneme	A	B	C	İterasyon Ortalaması
1	1	1	1	22.80
2	1	1	2	11.57
3	1	2	1	15.27
4	1	2	2	13.27
5	2	1	1	10.93
6	2	1	2	10.77
7	2	2	1	6.47
8	2	2	2	8.30

8.6. Veri Analizi

Deneyde verilerin analizi L8 etki hesap tablosu yardımıyla yapılmıştır. Tablo 8.3'te belirtilen değerler, L8 etki hesap tablosuna işlenir ve etkiler hesaplanarak sıraya konur. Sonuçlar Tablo 8.4'te verilmiştir. Daha sonra, önemli etkilerin belirlenmesi için normal olasılık grafiği oluşturulur. NOG, Şekil 8.1 olarak verilmiştir.



Şekil 8.1. Normal olasılık grafiği

Tablo 8.4. L8 etki hesap tablosu

Standart sıra	Deneme değeri	1. Sütun		2. Sütun		3. Sütun		4. Sütun		5. Sütun		6. Sütun		7. Sütun	
		A		B		C		AB		AC		BC		ABC	
		1	2	1	2	1	2	1	2	1	2	1	2	1	2
1	22.80	22.80		22.80		22.80		22.80		22.80		22.80		22.80	
2	11.57	11.57		11.57		11.57		11.57		11.57		11.57		11.57	
3	15.27	15.27		15.27		15.27		15.27		15.27		15.27		15.27	
4	13.27	13.27		13.27		13.27		13.27		13.27		13.27		13.27	
5	10.93	10.93		10.93		10.93		10.93		10.93		10.93		10.93	
6	10.77	10.77		10.77		10.77		10.77		10.77		10.77		10.77	
7	6.47	6.47		6.47		6.47		6.47		6.47		6.47		6.47	
8	8.30	8.30		8.30		8.30		8.30		8.30		8.30		8.30	
TOPLAM	99.38	62.91	36.47	56.07	43.31	55.47	43.91	50.24	49.14	42.24	57.14	44.08	55.30	53.31	46.07
SAYI	8	4	4	4	4	4	4	4	4	4	4	4	4	4	4
ORTALAMA	12.42	15.73	9.12	14.02	10.83	13.87	10.98	12.56	12.29	10.56	14.29	11.02	13.83	13.33	11.52
ETKİ		-6.61		-3.19		-2.89		-0.27		3.73		2.81		-1.81	
SIRA		1		2		3		5		7		6		4	

Şekil 8.1'deki normal olasılık grafiğine göre, ana etki olarak A ve B, etkileşim olarak da AC etkileşimi önemlidir.

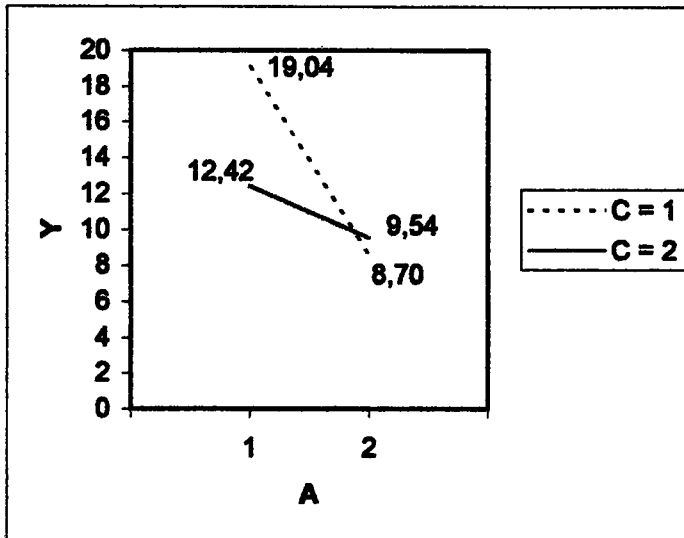
AC etkileşiminin önemliliğinin saptanmış olması, bu etkileşimi oluşturan A ve C faktörlerinin birlikte dikkate alınmasını gerektirmektedir. Sonuç değişkeni Y'nin optimize edilebilmesi için etkileşimin yorumunun yapılarak Y'yi nasıl etkilediğinin belirlenmesi gerekir. Etkileşimin yorumu etkileşim grafiği aracılığıyla yapılır.

8.7. Faktör Etkileşimleri

Aşağıdaki Tablo 8.5'te sırasıyla AC etkileşimi için faktörlerin seviyelerine göre ortalamalar hesaplanmıştır. Tabloda ortalamalar altı çizili olarak verilmektedir. Bu tablodan yararlanılarak etkileşim grafik haline getirilmiştir.

Tablo 8.5. AC etkileşimine ait ortalamalar

A	C	
	1	2
1	22.80 ; 15.27 <u>19.04</u>	11.57 ; 13.27 <u>12.42</u>
2	10.93 ; 6.47 <u>8.70</u>	10.77 ; 8.30 <u>9.54</u>



Şekil 8.2. AC etkileşimi

Şekil 8.2’den görüldüğü üzere doğrular birbirlerine paralel değildir. Bu nedenle AC etkileşimi önemlidir. Ayrıca doğrular aynı eğilimli oldukları için AC etkileşimi pozitifdir. Nitekim Tablo 8.4’te de AC etkileşiminin değeri artı (+3.73) olarak hesaplanmıştır. Şekil 8.2’ye göre ortalamaların üçü arasında (A_1C_2 , A_2C_1 ve A_2C_2) istatistiksel olarak önemli bir fark yoktur. Etkileşim ortalamalarından, $A_1C_1 = 19.04$, diğerlerinden önemli ölçüde büyüktür. Dolayısıyla amacımız sonuç değişkeni Y’nin değerini azaltmak olduğu için $A=1$, $C=1$ kombinasyonundan kaçınmak gerekir.

8.8. Deney Sonucu

AC etkileşiminin analizine göre $A=1$, $C=1$ kombinasyonundan kaçınılması gerekir. A faktörünün ana etkisi dikkate alındığında ise sonuç değişkeni Y’nin değerini azaltmak için bu faktörün ikinci seviyesi tercih edilir. A faktörü ikinci seviyesinde tutulduğu sürece AC etkileşiminin analizine göre C faktörü her iki seviyesinde de bulunabilir. Çünkü Şekil 8.2’den de görüldüğü gibi $A_2C_1 = 8.7$ ve $A_2C_2 = 9.54$ değerleri arasında sonuç değişkeni açısından istatistiksel olarak bir fark yoktur. Diğer önemli faktör olan B’nin ana etkisi dikkate alındığında ise sonuç değişkeni Y’nin değerini azaltmak için bu faktörün ikinci seviyesi tercih edilir. Tablo 8.4’teki etki değerlerine göre C faktörü Y açısından A ve B kadar önemli değildir. Kontrolü gerekmeyen faktörler uygulamada seviye 1 ile seviye 2 değeri arasında herhangi bir değere ayarlanabilir. Sonuç değişkeni olan iterasyon sayısını azaltmak amacıyla bir sonraki bölümdeki uygulama için C faktörünün, Tablo 8.4’teki ana etkisi göz önüne alınarak, ikinci seviyesi tercih edilmiştir.

Tablo 8.1’deki seviyelerin optimal değerleri bakımından bu deneyin sonucu şöyle özetlenebilir:

$$A_2 = 50, B_2 = 100, C_2 = 100$$

9. DENEY UYGULAMASI – 3

9.1. Giriş

Bu uygulama, bir önceki bölümdeki uygulamanın devamıdır. Bu bölümde bir önceki bölümde elde edilen sonuçlar kullanılarak yeni bir deney tasarlanmıştır. Sonuç değişkeni değişmemiştir ama faktörlerin seviyeleri bir önceki bölümde elde edilen sonuçlar doğrultusunda değişmiştir. Bu uygulamanın amacı daha hassas sonuçlar elde etmektir. Deney tasarım edildikten sonra deney gerçekleştirilmiştir. Daha sonra elde edilen veriler kullanılarak her faktörün en iyi seviyeleri belirlenmiştir.

9.2. Sonuç Değişkeninin Belirlenmesi

Bu deney için sonuç değişkeni olarak iterasyon sayısı seçilmiştir. Sonuç değişkeni göz önüne alındığında amaç en iyi çözüme ulaşmak için gereken iterasyon sayısını minimuma getirmektir.

9.3. Faktörlerin ve Seviyelerinin Belirlenmesi

Üç faktör kontrol edilebilir faktör olarak tanımlanmıştır. Bu faktörler başlangıç popülasyonu (A) , çaprazlama oranı (B) ve mutasyon oranıdır (C).

Herbir faktör ikişer seviyeye sahiptir ve bu seviyeler Tablo 9.1’de gösterilmiştir.

Tablo 9.1. Kontrol edilebilir faktörler ve seviyeleri

	Başlangıç Popülasyonu	Çaprazlama Oranı	Mutasyon Oranı
Seviye 1	40	75	75
Seviye 2	60	100	100

9.4. Deneyin Tasarımı

Bu deneyde iterasyon sayısını etkileyen toplam üç faktör bulunmaktadır. Her faktör ikişer seviyeye sahiptir.

Deneyde herbir denemede 30 tekrar yapılmıştır. L8 dizayn matrisi kullanılmıştır. Bu matris tablo 9.2’de verilmektedir. Deney üç faktöre ve L8 dizayn matrisine sahip olduğu için tam faktöriyeldir.

Tablo 9.2. Deneyde kullanılan L8 dizayn matrisi

Standart Sıra	ETKİLER						
	A	B	C	AB	AC	BC	ABC
1	1	1	1	2	2	2	1
2	1	1	2	2	1	1	2
3	1	2	1	1	2	1	2
4	1	2	2	1	1	2	1
5	2	1	1	1	1	2	2
6	2	1	2	1	2	1	1
7	2	2	1	2	1	1	1
8	2	2	2	2	2	2	2
Sütun No	1	2	3	4	5	6	7

9.5. Deneyin Yapılışı

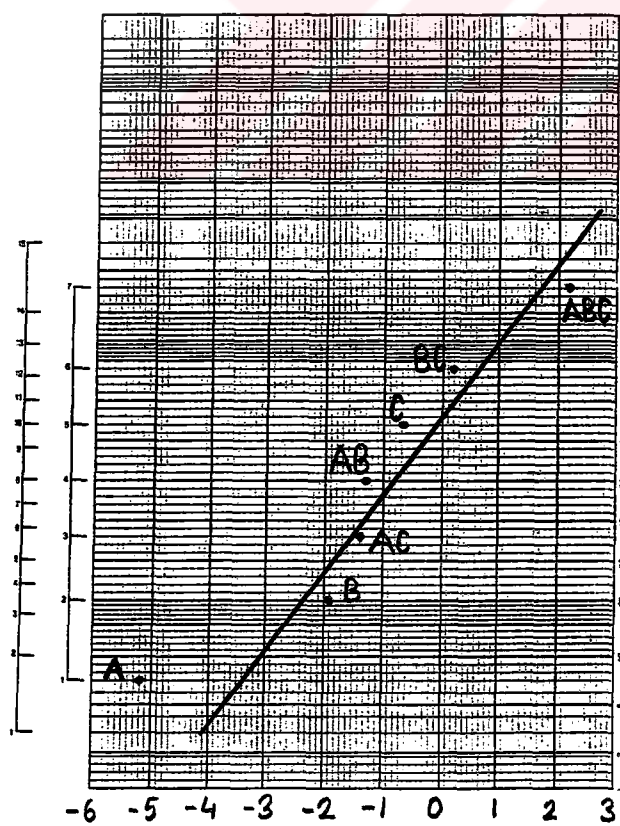
Deneyler Turbo Paskal programlama dilinde hazırlanan Genetik algoritma programı yardımı ile bilgisayarda gerçekleştirilmiştir. Elde edilen veriler ayrıntılı olarak bilgisayar çıktıları şeklinde ekte verilmiştir. Deneyde elde edilen veriler Tablo 9.3’te özetlenmiştir.

Tablo 9.3. Deney verileri

Deneme	A	B	C	İterasyon Ortalaması
1	1	1	1	10.57
2	1	1	2	13.30
3	1	2	1	12.00
4	1	2	2	10.73
5	2	1	1	10.30
6	2	1	2	5.83
7	2	2	1	4.60
8	2	2	2	4.97

9.6. Veri Analizi

Deneyde verilerin analizi L8 etki hesap tablosu yardımıyla yapılmıştır. Tablo 9.3'te belirtilen değerler, L8 etki hesap tablosuna işlenir ve etkiler hesaplanarak sıraya konur. Sonuçlar Tablo 9.4'te verilmiştir. Daha sonra, önemli etkilerin belirlenmesi için normal olasılık grafiği oluşturulur. NOG, Şekil 9.1 olarak verilmiştir.



Şekil 9.1. Normal olasılık grafiği

Tablo 9.4. L8 etki hesap tablosu

Standart sıra	Deneme değeri	1. Sütun		2. Sütun		3. Sütun		4. Sütun		5. Sütun		6. Sütun		7. Sütun	
		A		B		C		AB		AC		BC		ABC	
		1	2	1	2	1	2	1	2	1	2	1	2	1	2
1	10.57	10.57		10.57		10.57		10.57		10.57		10.57		10.57	
2	13.30	13.30		13.30		13.30		13.30		13.30		13.30		13.30	
3	12.00	12.00		12.00		12.00		12.00		12.00		12.00		12.00	
4	10.73	10.73		10.73		10.73		10.73		10.73		10.73		10.73	
5	10.30	10.30		10.30		10.30		10.30		10.30		10.30		10.30	
6	5.83	5.83		5.83		5.83		5.83		5.83		5.83		5.83	
7	4.60	4.60		4.60		4.60		4.60		4.60		4.60		4.60	
8	4.97	4.97		4.97		4.97		4.97		4.97		4.97		4.97	
TOPLAM	72.30	46.60	25.70	40.00	32.30	37.47	34.83	38.86	33.44	38.93	33.37	35.73	36.57	31.73	40.57
SAYI	8	4	4	4	4	4	4	4	4	4	4	4	4	4	4
ORTALAMA	9.04	11.65	6.43	10.00	8.08	9.37	8.71	9.72	8.36	9.73	8.34	8.93	9.14	7.93	10.14
ETKİ		-5.22		-1.92		-0.66		-1.36		-1.39		0.21		2.21	
SIRA		1		2		5		4		3		6		7	

Şekil 9.1 sonuç değişkeni Y açısından sadece A faktörünün önemli olduğunu göstermektedir.

Herhangi bir etkileşim önemli bulunmamıştır. Bunu için etkileşim grafiği çizmeye gerek yoktur.

9.7. Deney Sonucu

A (başlangıç popülasyonu) faktörünün etki değeri diğer etki değerlerinden daha yüksektir. Bu yüzden iterasyon sayısına etkisi çok daha fazla olmaktadır. Problemimiz “en küçük – en iyi” türünde bir problem olduğundan, bu faktörün seviyesi Y’nin ortalama değerini azaltacak biçimde seçilmelidir. Bu değer şöyledir:

$$A_2 = 60$$

Diğer yandan, B ve C faktörleri Y açısından A kadar önemli değildir. Kontrolü gerekmeyen bu faktörler uygulamada seviye 1 ile seviye 2 değeri arasında herhangi bir değere ayarlanabilir. Her iki faktör için seviye değerlerinin ortalamasını kullanabiliriz. Bu değerler şöyledir:

$$B_2 = 87.5, C_2 = 87.5$$

10. SONUÇLAR

NP olarak bilinen akış tipi çizelgeleme problemlerinin genetik algoritma ile çözüm kalitesi, genetik algortmada kullanılan parametrelere bağlıdır. Genetik algortmada optimuma yakın çözüm veren parametreler problemlerin yapısına göre farklılık gösterir. Her problemde aynı parametre seti kullanılamaz.

Optimum çözümleri bulunamayan, n-iş, m-makine ($m > 2$) akış tipi çizelgeleme problemlerinin rassal arama teknikli olan genetik algoritma ile çözüm performansının artırılmasında etkin parametrelerin belirlenmesine ilişkin uygulanan deney tasarımı yönteminden elde edilen sonuçlara göre, genetik algoritmanın performansını etkileyen temel faktör çaprazlama yöntemidir. Çok sayıda optimum veya optimuma yakın çözümü bulunan çizelgeleme problemlerinde çaprazlama operatörünün etkisi, optimum veya optimuma yakın çözüm kümesi küçük olan problemlerden daha azdır. Genetik algoritmanın performansını etkileyen ikinci faktörün çaprazlama oranı ve üçüncü faktörün de üreme yöntemi olduğu belirlenmiştir.

İşlem sürelerinin [1-99] arasında seçildiği durumlarda akış tipi çizelgeleme problemlerinin genetik algoritma ile çözümünde Tablo 10.1'deki parametrelerin kullanılmasının iyi sonuç vereceği belirlenmiştir. Başka şartlar için deneyler yeniden yapılmalıdır.

Tablo 10.1. Uygun parametre seti

Parametreler	Başlangıç Popülasyonu	Üreme Yöntemi	Çaprazlama Yöntemi	Mutasyon Yöntemi	Çaprazlama Oranı	Mutasyon Oranı
Uygun olanlar	60	Akış Zamanlı R.Ç.	Sıralı Çaprazlama (OX)	Keyfi Üç iş değiştirme	%87.5	%87.5

KAYNAKLAR

- [1] Lochner, R. H. and Matar, J. E., 1990. Designing for Quality, Chapman and Hall, London.
- [2] Şirvancı, M., 1997. Kalite İçin Deney Tasarımı “ Taguchi Yaklaşımı “, Literatür Yayınları, İstanbul.
- [3] Gunst, R. F. and Mason, R. L., 1991. How to Construct Fractional Factorial Experiments, ASQC Quality Press, Milwaukee.
- [4] Graessel, B. and Zeidler, P., 1993. Using Quality Function Deployment to Improve Customer Service, Quality Progress.
- [5] Engin, O., 2000. Akış Tipi Çizelgeleme Problemlerinin Genetik Algoritma İle Çözüm Kalitesinin Araştırılmasında Parametre Optimizasyonu, Doktora Tezi, İ.T.Ü. Fen Bilimleri Enstitüsü, İstanbul.
- [6] Montgomery, D. C., 1996. Design and Analysis of Experiments, John Wiley, New York.
- [7] Ronald, D., Thomas, W. N. and Liroyd, P. P., 1991. Improving Quality Through Planned Experimentation, Mc Graw-Hill.
- [8] Nackar, R. N., 1986. Taguchi's Quality Philosophy: Analysis and Comentary, Quality Progress.
- [9] Bhote, K. R., 1991. World Class Quality, American Management Association.
- [10] Mitra, A., 1993. Fundamentals of Quality Control and Improvement, Macmillian Publishing Company.
- [11] Tam, S. C., Noor, Y. M.M. and Yang, L. J., 1993. Maximizing the Output Power of CO₂ Laser Using the Taguchi Technique, *IEEE Journal of Quantom Electronics*, 29, No. 1.
- [12] Hamada, M. and Wu, C. F., 1992. Analysis of Designed Experiments with Complex Aliasing, *Journal of Quality Technology*, 24, 130-137.
- [13] Montgomery, C. D., 1985. Statistical Quality Control, John Wiley & Sons Inc.
- [14] Lanche, A. E., 1994. Quality by Design, American Supplier Institute Press.
- [15] Hakes, C., 1992. Total Quality Management, Chapman & Hall.

EKLER

İŞLEM SÜRELERİ

Makine	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
M1	64	73	8	85	64	21	59	68	6	25	50	40	97	12	29
M2	96	21	43	31	32	40	40	90	40	57	42	27	32	64	94

DENEY UYGULAMASI - 1

Deneme 1:

- 1.Problem = The best result is 755 at 33. Generation
- 2.Problem = The best result is 755 at 180. Generation
- 3.Problem = The best result is 755 at 196. Generation
- 4.Problem = The best result is 755 at 13. Generation
- 5.Problem = The best result is 755 at 1. Generation
- 6.Problem = The best result is 755 at 17. Generation
- 7.Problem = The best result is 755 at 22. Generation
- 8.Problem = The best result is 755 at 36. Generation
- 9.Problem = The best result is 755 at 15. Generation
- 10.Problem = The best result is 755 at 62. Generation
- 11.Problem = The best result is 755 at 1. Generation
- 12.Problem = The best result is 755 at 55. Generation
- 13.Problem = The best result is 755 at 46. Generation
- 14.Problem = The best result is 755 at 10. Generation
- 15.Problem = The best result is 755 at 1. Generation
- 16.Problem = The best result is 755 at 89. Generation
- 17.Problem = The best result is 755 at 65. Generation
- 18.Problem = The best result is 755 at 9. Generation
- 19.Problem = The best result is 755 at 115. Generation
- 20.Problem = The best result is 755 at 14. Generation
- 21.Problem = The best result is 755 at 83. Generation
- 22.Problem = The best result is 755 at 13. Generation
- 23.Problem = The best result is 755 at 2. Generation
- 24.Problem = The best result is 755 at 10. Generation
- 25.Problem = The best result is 755 at 1. Generation
- 26.Problem = The best result is 755 at 40. Generation
- 27.Problem = The best result is 755 at 15. Generation
- 28.Problem = The best result is 755 at 17. Generation
- 29.Problem = The best result is 755 at 33. Generation
- 30.Problem = The best result is 755 at 17. Generation

Deneme 2:

- 1.Problem = The best result is 755 at 145. Generation
- 2.Problem = The best result is 755 at 558. Generation
- 3.Problem = The best result is 755 at 162. Generation
- 4.Problem = The best result is 755 at 24. Generation
- 5.Problem = The best result is 755 at 45. Generation
- 6.Problem = The best result is 755 at 36. Generation
- 7.Problem = The best result is 755 at 19. Generation
- 8.Problem = The best result is 755 at 48. Generation
- 9.Problem = The best result is 755 at 467. Generation
- 10.Problem = The best result is 755 at 386. Generation
- 11.Problem = The best result is 755 at 284. Generation
- 12.Problem = The best result is 755 at 29. Generation
- 13.Problem = The best result is 755 at 110. Generation
- 14.Problem = The best result is 755 at 57. Generation
- 15.Problem = The best result is 755 at 174. Generation
- 16.Problem = The best result is 755 at 129. Generation
- 17.Problem = The best result is 755 at 22. Generation
- 18.Problem = The best result is 755 at 355. Generation
- 19.Problem = The best result is 755 at 203. Generation
- 20.Problem = The best result is 755 at 173. Generation
- 21.Problem = The best result is 755 at 77. Generation
- 22.Problem = The best result is 755 at 211. Generation
- 23.Problem = The best result is 755 at 146. Generation
- 24.Problem = The best result is 755 at 1. Generation
- 25.Problem = The best result is 755 at 186. Generation
- 26.Problem = The best result is 755 at 42. Generation
- 27.Problem = The best result is 755 at 208. Generation
- 28.Problem = The best result is 755 at 57. Generation
- 29.Problem = The best result is 755 at 104. Generation
- 30.Problem = The best result is 755 at 4. Generation

Deneme 3:

- 1.Problem = The best result is 755 at 71. Generation
- 2.Problem = The best result is 755 at 10. Generation
- 3.Problem = The best result is 755 at 13. Generation
- 4.Problem = The best result is 755 at 23. Generation
- 5.Problem = The best result is 755 at 16. Generation
- 6.Problem = The best result is 755 at 70. Generation
- 7.Problem = The best result is 755 at 41. Generation
- 8.Problem = The best result is 755 at 12. Generation
- 9.Problem = The best result is 755 at 86. Generation
- 10.Problem = The best result is 755 at 45. Generation
- 11.Problem = The best result is 755 at 116. Generation
- 12.Problem = The best result is 755 at 19. Generation
- 13.Problem = The best result is 755 at 13. Generation
- 14.Problem = The best result is 755 at 12. Generation
- 15.Problem = The best result is 755 at 28. Generation
- 16.Problem = The best result is 755 at 16. Generation
- 17.Problem = The best result is 755 at 85. Generation

- 18.Problem = The best result is 755 at 4. Generation
- 19.Problem = The best result is 755 at 37. Generation
- 20.Problem = The best result is 755 at 43. Generation
- 21.Problem = The best result is 755 at 17. Generation
- 22.Problem = The best result is 755 at 20. Generation
- 23.Problem = The best result is 755 at 42. Generation
- 24.Problem = The best result is 755 at 8. Generation
- 25.Problem = The best result is 755 at 2. Generation
- 26.Problem = The best result is 755 at 34. Generation
- 27.Problem = The best result is 755 at 93. Generation
- 28.Problem = The best result is 755 at 95. Generation
- 29.Problem = The best result is 755 at 3. Generation
- 30.Problem = The best result is 755 at 35. Generation

Deneme 4:

- 1.Problem = The best result is 755 at 113. Generation
- 2.Problem = The best result is 755 at 79. Generation
- 3.Problem = The best result is 755 at 150. Generation
- 4.Problem = The best result is 755 at 174. Generation
- 5.Problem = The best result is 755 at 198. Generation
- 6.Problem = The best result is 755 at 186. Generation
- 7.Problem = The best result is 755 at 218. Generation
- 8.Problem = The best result is 755 at 462. Generation
- 9.Problem = The best result is 755 at 51. Generation
- 10.Problem = The best result is 755 at 80. Generation
- 11.Problem = The best result is 755 at 168. Generation
- 12.Problem = The best result is 755 at 99. Generation
- 13.Problem = The best result is 755 at 250. Generation
- 14.Problem = The best result is 755 at 214. Generation
- 15.Problem = The best result is 755 at 162. Generation
- 16.Problem = The best result is 755 at 1. Generation
- 17.Problem = The best result is 755 at 184. Generation
- 18.Problem = The best result is 755 at 104. Generation
- 19.Problem = The best result is 755 at 42. Generation
- 20.Problem = The best result is 755 at 16. Generation
- 21.Problem = The best result is 755 at 68. Generation
- 22.Problem = The best result is 755 at 301. Generation
- 23.Problem = The best result is 755 at 56. Generation
- 24.Problem = The best result is 755 at 140. Generation
- 25.Problem = The best result is 755 at 113. Generation
- 26.Problem = The best result is 755 at 63. Generation
- 27.Problem = The best result is 755 at 24. Generation
- 28.Problem = The best result is 755 at 35. Generation
- 29.Problem = The best result is 755 at 165. Generation
- 30.Problem = The best result is 755 at 142. Generation

Deneme 5:

- 1.Problem = The best result is 755 at 193. Generation
- 2.Problem = The best result is 755 at 145. Generation
- 3.Problem = The best result is 755 at 22. Generation

4.Problem = The best result is 755 at 28. Generation
5.Problem = The best result is 755 at 8. Generation
6.Problem = The best result is 755 at 42. Generation
7.Problem = The best result is 755 at 1. Generation
8.Problem = The best result is 755 at 208. Generation
9.Problem = The best result is 755 at 1. Generation
10.Problem = The best result is 755 at 308. Generation
11.Problem = The best result is 755 at 125. Generation
12.Problem = The best result is 755 at 55. Generation
13.Problem = The best result is 755 at 1. Generation
14.Problem = The best result is 755 at 11. Generation
15.Problem = The best result is 755 at 82. Generation
16.Problem = The best result is 755 at 1. Generation
17.Problem = The best result is 755 at 13. Generation
18.Problem = The best result is 755 at 107. Generation
19.Problem = The best result is 755 at 1. Generation
20.Problem = The best result is 755 at 99. Generation
21.Problem = The best result is 755 at 33. Generation
22.Problem = The best result is 755 at 56. Generation
23.Problem = The best result is 755 at 33. Generation
24.Problem = The best result is 755 at 2. Generation
25.Problem = The best result is 755 at 39. Generation
26.Problem = The best result is 755 at 48. Generation
27.Problem = The best result is 755 at 124. Generation
28.Problem = The best result is 755 at 64. Generation
29.Problem = The best result is 755 at 153. Generation
30.Problem = The best result is 755 at 1. Generation

Deneme 6:

1.Problem = The best result is 755 at 553. Generation
2.Problem = The best result is 755 at 102. Generation
3.Problem = The best result is 755 at 1. Generation
4.Problem = The best result is 755 at 263. Generation
5.Problem = The best result is 755 at 1. Generation
6.Problem = The best result is 755 at 50. Generation
7.Problem = The best result is 755 at 126. Generation
8.Problem = The best result is 755 at 1. Generation
9.Problem = The best result is 755 at 199. Generation
10.Problem = The best result is 755 at 5. Generation
11.Problem = The best result is 755 at 7. Generation
12.Problem = The best result is 755 at 485. Generation
13.Problem = The best result is 755 at 209. Generation
14.Problem = The best result is 755 at 188. Generation
15.Problem = The best result is 755 at 174. Generation
16.Problem = The best result is 755 at 1. Generation
17.Problem = The best result is 755 at 152. Generation
18.Problem = The best result is 755 at 1. Generation
19.Problem = The best result is 755 at 128. Generation
20.Problem = The best result is 755 at 1. Generation
21.Problem = The best result is 755 at 12. Generation

- 22.Problem = The best result is 755 at 79. Generation
- 23.Problem = The best result is 755 at 305. Generation
- 24.Problem = The best result is 755 at 302. Generation
- 25.Problem = The best result is 755 at 38. Generation
- 26.Problem = The best result is 755 at 136. Generation
- 27.Problem = The best result is 755 at 38. Generation
- 28.Problem = The best result is 755 at 1. Generation
- 29.Problem = The best result is 755 at 211. Generation
- 30.Problem = The best result is 755 at 302. Generation

Deneme 7:

- 1.Problem = The best result is 755 at 40. Generation
- 2.Problem = The best result is 755 at 6. Generation
- 3.Problem = The best result is 755 at 3. Generation
- 4.Problem = The best result is 755 at 38. Generation
- 5.Problem = The best result is 755 at 28. Generation
- 6.Problem = The best result is 755 at 1. Generation
- 7.Problem = The best result is 755 at 12. Generation
- 8.Problem = The best result is 755 at 100. Generation
- 9.Problem = The best result is 755 at 1. Generation
- 10.Problem = The best result is 755 at 126. Generation
- 11.Problem = The best result is 755 at 15. Generation
- 12.Problem = The best result is 755 at 123. Generation
- 13.Problem = The best result is 755 at 15. Generation
- 14.Problem = The best result is 755 at 4. Generation
- 15.Problem = The best result is 755 at 6. Generation
- 16.Problem = The best result is 755 at 65. Generation
- 17.Problem = The best result is 755 at 20. Generation
- 18.Problem = The best result is 755 at 64. Generation
- 19.Problem = The best result is 755 at 30. Generation
- 20.Problem = The best result is 755 at 16. Generation
- 21.Problem = The best result is 755 at 11. Generation
- 22.Problem = The best result is 755 at 20. Generation
- 23.Problem = The best result is 755 at 7. Generation
- 24.Problem = The best result is 755 at 109. Generation
- 25.Problem = The best result is 755 at 196. Generation
- 26.Problem = The best result is 755 at 108. Generation
- 27.Problem = The best result is 755 at 39. Generation
- 28.Problem = The best result is 755 at 50. Generation
- 29.Problem = The best result is 755 at 23. Generation
- 30.Problem = The best result is 755 at 268. Generation

Deneme 8:

- 1.Problem = The best result is 755 at 67. Generation
- 2.Problem = The best result is 755 at 7. Generation
- 3.Problem = The best result is 755 at 6. Generation
- 4.Problem = The best result is 755 at 6. Generation
- 5.Problem = The best result is 755 at 1. Generation
- 6.Problem = The best result is 755 at 7. Generation
- 7.Problem = The best result is 755 at 6. Generation

- 8.Problem = The best result is 755 at 10. Generation
- 9.Problem = The best result is 755 at 2. Generation
- 10.Problem = The best result is 755 at 1. Generation
- 11.Problem = The best result is 755 at 135. Generation
- 12.Problem = The best result is 755 at 3. Generation
- 13.Problem = The best result is 755 at 92. Generation
- 14.Problem = The best result is 755 at 15. Generation
- 15.Problem = The best result is 755 at 2. Generation
- 16.Problem = The best result is 755 at 32. Generation
- 17.Problem = The best result is 755 at 9. Generation
- 18.Problem = The best result is 755 at 53. Generation
- 19.Problem = The best result is 755 at 25. Generation
- 20.Problem = The best result is 755 at 17. Generation
- 21.Problem = The best result is 755 at 32. Generation
- 22.Problem = The best result is 755 at 10. Generation
- 23.Problem = The best result is 755 at 36. Generation
- 24.Problem = The best result is 755 at 11. Generation
- 25.Problem = The best result is 755 at 25. Generation
- 26.Problem = The best result is 755 at 46. Generation
- 27.Problem = The best result is 755 at 6. Generation
- 28.Problem = The best result is 755 at 35. Generation
- 29.Problem = The best result is 755 at 13. Generation
- 30.Problem = The best result is 755 at 6. Generation

DENEY UYGULAMASI – 2

Deneme 1:

- 1.Problem = The best result is 755 at 2. Generation
- 2.Problem = The best result is 755 at 47. Generation
- 3.Problem = The best result is 755 at 9. Generation
- 4.Problem = The best result is 755 at 49. Generation
- 5.Problem = The best result is 755 at 12. Generation
- 6.Problem = The best result is 755 at 22. Generation
- 7.Problem = The best result is 755 at 3. Generation
- 8.Problem = The best result is 755 at 1. Generation
- 9.Problem = The best result is 755 at 15. Generation
- 10.Problem = The best result is 755 at 19. Generation
- 11.Problem = The best result is 755 at 22. Generation
- 12.Problem = The best result is 755 at 21. Generation
- 13.Problem = The best result is 755 at 67. Generation
- 14.Problem = The best result is 755 at 14. Generation
- 15.Problem = The best result is 755 at 7. Generation
- 16.Problem = The best result is 755 at 6. Generation
- 17.Problem = The best result is 755 at 38. Generation
- 18.Problem = The best result is 755 at 31. Generation
- 19.Problem = The best result is 755 at 1. Generation
- 20.Problem = The best result is 755 at 32. Generation
- 21.Problem = The best result is 755 at 22. Generation
- 22.Problem = The best result is 755 at 32. Generation

- 23.Problem = The best result is 755 at 48. Generation
- 24.Problem = The best result is 755 at 30. Generation
- 25.Problem = The best result is 755 at 31. Generation
- 26.Problem = The best result is 755 at 31. Generation
- 27.Problem = The best result is 755 at 9. Generation
- 28.Problem = The best result is 755 at 26. Generation
- 29.Problem = The best result is 755 at 28. Generation
- 30.Problem = The best result is 755 at 19. Generation

Deneme 2:

- 1.Problem = The best result is 755 at 23. Generation
- 2.Problem = The best result is 755 at 33. Generation
- 3.Problem = The best result is 755 at 5. Generation
- 4.Problem = The best result is 755 at 1. Generation
- 5.Problem = The best result is 755 at 3. Generation
- 6.Problem = The best result is 755 at 3. Generation
- 7.Problem = The best result is 755 at 5. Generation
- 8.Problem = The best result is 755 at 5. Generation
- 9.Problem = The best result is 755 at 17. Generation
- 10.Problem = The best result is 755 at 4. Generation
- 11.Problem = The best result is 755 at 33. Generation
- 12.Problem = The best result is 755 at 14. Generation
- 13.Problem = The best result is 755 at 4. Generation
- 14.Problem = The best result is 755 at 5. Generation
- 15.Problem = The best result is 755 at 17. Generation
- 16.Problem = The best result is 755 at 3. Generation
- 17.Problem = The best result is 755 at 1. Generation
- 18.Problem = The best result is 755 at 1. Generation
- 19.Problem = The best result is 755 at 9. Generation
- 20.Problem = The best result is 755 at 3. Generation
- 21.Problem = The best result is 755 at 30. Generation
- 22.Problem = The best result is 755 at 28. Generation
- 23.Problem = The best result is 755 at 8. Generation
- 24.Problem = The best result is 755 at 7. Generation
- 25.Problem = The best result is 755 at 18. Generation
- 26.Problem = The best result is 755 at 7. Generation
- 27.Problem = The best result is 755 at 14. Generation
- 28.Problem = The best result is 755 at 11. Generation
- 29.Problem = The best result is 755 at 11. Generation
- 30.Problem = The best result is 755 at 24. Generation

Deneme 3:

- 1.Problem = The best result is 755 at 24. Generation
- 2.Problem = The best result is 755 at 3. Generation
- 3.Problem = The best result is 755 at 8. Generation
- 4.Problem = The best result is 755 at 13. Generation
- 5.Problem = The best result is 755 at 4. Generation
- 6.Problem = The best result is 755 at 50. Generation
- 7.Problem = The best result is 755 at 10. Generation
- 8.Problem = The best result is 755 at 41. Generation

9.Problem = The best result is 755 at 10. Generation
10.Problem = The best result is 755 at 3. Generation
11.Problem = The best result is 755 at 3. Generation
12.Problem = The best result is 755 at 27. Generation
13.Problem = The best result is 755 at 13. Generation
14.Problem = The best result is 755 at 2. Generation
15.Problem = The best result is 755 at 8. Generation
16.Problem = The best result is 755 at 38. Generation
17.Problem = The best result is 755 at 13. Generation
18.Problem = The best result is 755 at 29. Generation
19.Problem = The best result is 755 at 59. Generation
20.Problem = The best result is 755 at 1. Generation
21.Problem = The best result is 755 at 3. Generation
22.Problem = The best result is 755 at 4. Generation
23.Problem = The best result is 755 at 7. Generation
24.Problem = The best result is 755 at 9. Generation
25.Problem = The best result is 755 at 24. Generation
26.Problem = The best result is 755 at 11. Generation
27.Problem = The best result is 755 at 34. Generation
28.Problem = The best result is 755 at 1. Generation
29.Problem = The best result is 755 at 5. Generation
30.Problem = The best result is 755 at 1. Generation

Deneme 4:

1.Problem = The best result is 755 at 39. Generation
2.Problem = The best result is 755 at 45. Generation
3.Problem = The best result is 755 at 7. Generation
4.Problem = The best result is 755 at 7. Generation
5.Problem = The best result is 755 at 1. Generation
6.Problem = The best result is 755 at 1. Generation
7.Problem = The best result is 755 at 5. Generation
8.Problem = The best result is 755 at 16. Generation
9.Problem = The best result is 755 at 1. Generation
10.Problem = The best result is 755 at 33. Generation
11.Problem = The best result is 755 at 15. Generation
12.Problem = The best result is 755 at 2. Generation
13.Problem = The best result is 755 at 2. Generation
14.Problem = The best result is 755 at 4. Generation
15.Problem = The best result is 755 at 32. Generation
16.Problem = The best result is 755 at 12. Generation
17.Problem = The best result is 755 at 7. Generation
18.Problem = The best result is 755 at 3. Generation
19.Problem = The best result is 755 at 17. Generation
20.Problem = The best result is 755 at 5. Generation
21.Problem = The best result is 755 at 30. Generation
22.Problem = The best result is 755 at 6. Generation
23.Problem = The best result is 755 at 34. Generation
24.Problem = The best result is 755 at 6. Generation
25.Problem = The best result is 755 at 13. Generation
26.Problem = The best result is 755 at 21. Generation

- 27.Problem = The best result is 755 at 14. Generation
- 28.Problem = The best result is 755 at 9. Generation
- 29.Problem = The best result is 755 at 2. Generation
- 30.Problem = The best result is 755 at 9. Generation

Deneme 5:

- 1.Problem = The best result is 755 at 1. Generation
- 2.Problem = The best result is 755 at 54. Generation
- 3.Problem = The best result is 755 at 8. Generation
- 4.Problem = The best result is 755 at 1. Generation
- 5.Problem = The best result is 755 at 9. Generation
- 6.Problem = The best result is 755 at 2. Generation
- 7.Problem = The best result is 755 at 27. Generation
- 8.Problem = The best result is 755 at 23. Generation
- 9.Problem = The best result is 755 at 11. Generation
- 10.Problem = The best result is 755 at 7. Generation
- 11.Problem = The best result is 755 at 4. Generation
- 12.Problem = The best result is 755 at 1. Generation
- 13.Problem = The best result is 755 at 10. Generation
- 14.Problem = The best result is 755 at 1. Generation
- 15.Problem = The best result is 755 at 11. Generation
- 16.Problem = The best result is 755 at 11. Generation
- 17.Problem = The best result is 755 at 3. Generation
- 18.Problem = The best result is 755 at 5. Generation
- 19.Problem = The best result is 755 at 6. Generation
- 20.Problem = The best result is 755 at 21. Generation
- 21.Problem = The best result is 755 at 47. Generation
- 22.Problem = The best result is 755 at 1. Generation
- 23.Problem = The best result is 755 at 20. Generation
- 24.Problem = The best result is 755 at 2. Generation
- 25.Problem = The best result is 755 at 5. Generation
- 26.Problem = The best result is 755 at 20. Generation
- 27.Problem = The best result is 755 at 2. Generation
- 28.Problem = The best result is 755 at 10. Generation
- 29.Problem = The best result is 755 at 4. Generation
- 30.Problem = The best result is 755 at 1. Generation

Deneme 6:

- 1.Problem = The best result is 755 at 2. Generation
- 2.Problem = The best result is 755 at 28. Generation
- 3.Problem = The best result is 755 at 20. Generation
- 4.Problem = The best result is 755 at 2. Generation
- 5.Problem = The best result is 755 at 15. Generation
- 6.Problem = The best result is 755 at 11. Generation
- 7.Problem = The best result is 755 at 19. Generation
- 8.Problem = The best result is 755 at 7. Generation
- 9.Problem = The best result is 755 at 8. Generation
- 10.Problem = The best result is 755 at 29. Generation
- 11.Problem = The best result is 755 at 1. Generation
- 12.Problem = The best result is 755 at 1. Generation

- 13.Problem = The best result is 755 at 1. Generation
- 14.Problem = The best result is 755 at 1. Generation
- 15.Problem = The best result is 755 at 8. Generation
- 16.Problem = The best result is 755 at 1. Generation
- 17.Problem = The best result is 755 at 10. Generation
- 18.Problem = The best result is 755 at 5. Generation
- 19.Problem = The best result is 755 at 1. Generation
- 20.Problem = The best result is 755 at 1. Generation
- 21.Problem = The best result is 755 at 11. Generation
- 22.Problem = The best result is 755 at 26. Generation
- 23.Problem = The best result is 755 at 6. Generation
- 24.Problem = The best result is 755 at 32. Generation
- 25.Problem = The best result is 755 at 8. Generation
- 26.Problem = The best result is 755 at 22. Generation
- 27.Problem = The best result is 755 at 7. Generation
- 28.Problem = The best result is 755 at 15. Generation
- 29.Problem = The best result is 755 at 8. Generation
- 30.Problem = The best result is 755 at 17. Generation

Deneme 7:

- 1.Problem = The best result is 755 at 2. Generation
- 2.Problem = The best result is 755 at 11. Generation
- 3.Problem = The best result is 755 at 8. Generation
- 4.Problem = The best result is 755 at 6. Generation
- 5.Problem = The best result is 755 at 10. Generation
- 6.Problem = The best result is 755 at 12. Generation
- 7.Problem = The best result is 755 at 5. Generation
- 8.Problem = The best result is 755 at 1. Generation
- 9.Problem = The best result is 755 at 3. Generation
- 10.Problem = The best result is 755 at 10. Generation
- 11.Problem = The best result is 755 at 3. Generation
- 12.Problem = The best result is 755 at 2. Generation
- 13.Problem = The best result is 755 at 12. Generation
- 14.Problem = The best result is 755 at 11. Generation
- 15.Problem = The best result is 755 at 1. Generation
- 16.Problem = The best result is 755 at 8. Generation
- 17.Problem = The best result is 755 at 13. Generation
- 18.Problem = The best result is 755 at 6. Generation
- 19.Problem = The best result is 755 at 1. Generation
- 20.Problem = The best result is 755 at 11. Generation
- 21.Problem = The best result is 755 at 22. Generation
- 22.Problem = The best result is 755 at 13. Generation
- 23.Problem = The best result is 755 at 3. Generation
- 24.Problem = The best result is 755 at 5. Generation
- 25.Problem = The best result is 755 at 1. Generation
- 26.Problem = The best result is 755 at 3. Generation
- 27.Problem = The best result is 755 at 1. Generation
- 28.Problem = The best result is 755 at 7. Generation
- 29.Problem = The best result is 755 at 2. Generation
- 30.Problem = The best result is 755 at 1. Generation

Deneme 8:

- 1.Problem = The best result is 755 at 20. Generation
- 2.Problem = The best result is 755 at 20. Generation
- 3.Problem = The best result is 755 at 25. Generation
- 4.Problem = The best result is 755 at 1. Generation
- 5.Problem = The best result is 755 at 13. Generation
- 6.Problem = The best result is 755 at 6. Generation
- 7.Problem = The best result is 755 at 7. Generation
- 8.Problem = The best result is 755 at 1. Generation
- 9.Problem = The best result is 755 at 17. Generation
- 10.Problem = The best result is 755 at 2. Generation
- 11.Problem = The best result is 755 at 14. Generation
- 12.Problem = The best result is 755 at 10. Generation
- 13.Problem = The best result is 755 at 2. Generation
- 14.Problem = The best result is 755 at 15. Generation
- 15.Problem = The best result is 755 at 6. Generation
- 16.Problem = The best result is 755 at 5. Generation
- 17.Problem = The best result is 755 at 7. Generation
- 18.Problem = The best result is 755 at 8. Generation
- 19.Problem = The best result is 755 at 3. Generation
- 20.Problem = The best result is 755 at 19. Generation
- 21.Problem = The best result is 755 at 3. Generation
- 22.Problem = The best result is 755 at 8. Generation
- 23.Problem = The best result is 755 at 1. Generation
- 24.Problem = The best result is 755 at 1. Generation
- 25.Problem = The best result is 755 at 6. Generation
- 26.Problem = The best result is 755 at 7. Generation
- 27.Problem = The best result is 755 at 3. Generation
- 28.Problem = The best result is 755 at 4. Generation
- 29.Problem = The best result is 755 at 1. Generation
- 30.Problem = The best result is 755 at 14. Generation

DENEY UYGULAMASI – 3

Deneme 1:

- 1.Problem = The best result is 755 at 26. Generation
- 2.Problem = The best result is 755 at 7. Generation
- 3.Problem = The best result is 755 at 7. Generation
- 4.Problem = The best result is 755 at 2. Generation
- 5.Problem = The best result is 755 at 1. Generation
- 6.Problem = The best result is 755 at 10. Generation
- 7.Problem = The best result is 755 at 5. Generation
- 8.Problem = The best result is 755 at 5. Generation
- 9.Problem = The best result is 755 at 13. Generation
- 10.Problem = The best result is 755 at 21. Generation
- 11.Problem = The best result is 755 at 42. Generation
- 12.Problem = The best result is 755 at 7. Generation
- 13.Problem = The best result is 755 at 6. Generation
- 14.Problem = The best result is 755 at 5. Generation

- 15.Problem = The best result is 755 at 11. Generation
- 16.Problem = The best result is 755 at 4. Generation
- 17.Problem = The best result is 755 at 1. Generation
- 18.Problem = The best result is 755 at 25. Generation
- 19.Problem = The best result is 755 at 1. Generation
- 20.Problem = The best result is 755 at 1. Generation
- 21.Problem = The best result is 755 at 38. Generation
- 22.Problem = The best result is 755 at 4. Generation
- 23.Problem = The best result is 755 at 18. Generation
- 24.Problem = The best result is 755 at 4. Generation
- 25.Problem = The best result is 755 at 10. Generation
- 26.Problem = The best result is 755 at 7. Generation
- 27.Problem = The best result is 755 at 21. Generation
- 28.Problem = The best result is 755 at 5. Generation
- 29.Problem = The best result is 755 at 2. Generation
- 30.Problem = The best result is 755 at 8. Generation

Deneme 2:

- 1.Problem = The best result is 755 at 1. Generation
- 2.Problem = The best result is 755 at 9. Generation
- 3.Problem = The best result is 755 at 1. Generation
- 4.Problem = The best result is 755 at 25. Generation
- 5.Problem = The best result is 755 at 2. Generation
- 6.Problem = The best result is 755 at 1. Generation
- 7.Problem = The best result is 755 at 1. Generation
- 8.Problem = The best result is 755 at 25. Generation
- 9.Problem = The best result is 755 at 10. Generation
- 10.Problem = The best result is 755 at 20. Generation
- 11.Problem = The best result is 755 at 3. Generation
- 12.Problem = The best result is 755 at 18. Generation
- 13.Problem = The best result is 755 at 2. Generation
- 14.Problem = The best result is 755 at 7. Generation
- 15.Problem = The best result is 755 at 25. Generation
- 16.Problem = The best result is 755 at 14. Generation
- 17.Problem = The best result is 755 at 1. Generation
- 18.Problem = The best result is 755 at 2. Generation
- 19.Problem = The best result is 755 at 54. Generation
- 20.Problem = The best result is 755 at 3. Generation
- 21.Problem = The best result is 755 at 46. Generation
- 22.Problem = The best result is 755 at 8. Generation
- 23.Problem = The best result is 755 at 1. Generation
- 24.Problem = The best result is 755 at 11. Generation
- 25.Problem = The best result is 755 at 13. Generation
- 26.Problem = The best result is 755 at 3. Generation
- 27.Problem = The best result is 755 at 6. Generation
- 28.Problem = The best result is 755 at 26. Generation
- 29.Problem = The best result is 755 at 16. Generation
- 30.Problem = The best result is 755 at 45. Generation

Deneme 3:

- 1.Problem = The best result is 755 at 8. Generation
- 2.Problem = The best result is 755 at 20. Generation
- 3.Problem = The best result is 755 at 5. Generation
- 4.Problem = The best result is 755 at 2. Generation
- 5.Problem = The best result is 755 at 8. Generation
- 6.Problem = The best result is 755 at 16. Generation
- 7.Problem = The best result is 755 at 24. Generation
- 8.Problem = The best result is 755 at 1. Generation
- 9.Problem = The best result is 755 at 21. Generation
- 10.Problem = The best result is 755 at 30. Generation
- 11.Problem = The best result is 755 at 5. Generation
- 12.Problem = The best result is 755 at 9. Generation
- 13.Problem = The best result is 755 at 20. Generation
- 14.Problem = The best result is 755 at 26. Generation
- 15.Problem = The best result is 755 at 32. Generation
- 16.Problem = The best result is 755 at 3. Generation
- 17.Problem = The best result is 755 at 18. Generation
- 18.Problem = The best result is 755 at 15. Generation
- 19.Problem = The best result is 755 at 4. Generation
- 20.Problem = The best result is 755 at 19. Generation
- 21.Problem = The best result is 755 at 10. Generation
- 22.Problem = The best result is 755 at 5. Generation
- 23.Problem = The best result is 755 at 4. Generation
- 24.Problem = The best result is 755 at 8. Generation
- 25.Problem = The best result is 755 at 13. Generation
- 26.Problem = The best result is 755 at 6. Generation
- 27.Problem = The best result is 755 at 3. Generation
- 28.Problem = The best result is 755 at 10. Generation
- 29.Problem = The best result is 755 at 3. Generation
- 30.Problem = The best result is 755 at 12. Generation

Deneme 4:

- 1.Problem = The best result is 755 at 8. Generation
- 2.Problem = The best result is 755 at 18. Generation
- 3.Problem = The best result is 755 at 5. Generation
- 4.Problem = The best result is 755 at 37. Generation
- 5.Problem = The best result is 755 at 4. Generation
- 6.Problem = The best result is 755 at 35. Generation
- 7.Problem = The best result is 755 at 1. Generation
- 8.Problem = The best result is 755 at 28. Generation
- 9.Problem = The best result is 755 at 4. Generation
- 10.Problem = The best result is 755 at 12. Generation
- 11.Problem = The best result is 755 at 2. Generation
- 12.Problem = The best result is 755 at 1. Generation
- 13.Problem = The best result is 755 at 5. Generation
- 14.Problem = The best result is 755 at 6. Generation
- 15.Problem = The best result is 755 at 15. Generation
- 16.Problem = The best result is 755 at 1. Generation
- 17.Problem = The best result is 755 at 16. Generation

- 18.Problem = The best result is 755 at 6. Generation
- 19.Problem = The best result is 755 at 34. Generation
- 20.Problem = The best result is 755 at 4. Generation
- 21.Problem = The best result is 755 at 5. Generation
- 22.Problem = The best result is 755 at 1. Generation
- 23.Problem = The best result is 755 at 10. Generation
- 24.Problem = The best result is 755 at 11. Generation
- 25.Problem = The best result is 755 at 27. Generation
- 26.Problem = The best result is 755 at 2. Generation
- 27.Problem = The best result is 755 at 8. Generation
- 28.Problem = The best result is 755 at 8. Generation
- 29.Problem = The best result is 755 at 6. Generation
- 30.Problem = The best result is 755 at 2. Generation

Deneme 5:

- 1.Problem = The best result is 755 at 17. Generation
- 2.Problem = The best result is 755 at 13. Generation
- 3.Problem = The best result is 755 at 39. Generation
- 4.Problem = The best result is 755 at 17. Generation
- 5.Problem = The best result is 755 at 19. Generation
- 6.Problem = The best result is 755 at 3. Generation
- 7.Problem = The best result is 755 at 5. Generation
- 8.Problem = The best result is 755 at 17. Generation
- 9.Problem = The best result is 755 at 12. Generation
- 10.Problem = The best result is 755 at 1. Generation
- 11.Problem = The best result is 755 at 12. Generation
- 12.Problem = The best result is 755 at 14. Generation
- 13.Problem = The best result is 755 at 1. Generation
- 14.Problem = The best result is 755 at 3. Generation
- 15.Problem = The best result is 755 at 1. Generation
- 16.Problem = The best result is 755 at 12. Generation
- 17.Problem = The best result is 755 at 4. Generation
- 18.Problem = The best result is 755 at 5. Generation
- 19.Problem = The best result is 755 at 6. Generation
- 20.Problem = The best result is 755 at 18. Generation
- 21.Problem = The best result is 755 at 3. Generation
- 22.Problem = The best result is 755 at 10. Generation
- 23.Problem = The best result is 755 at 15. Generation
- 24.Problem = The best result is 755 at 4. Generation
- 25.Problem = The best result is 755 at 1. Generation
- 26.Problem = The best result is 755 at 26. Generation
- 27.Problem = The best result is 755 at 15. Generation
- 28.Problem = The best result is 755 at 1. Generation
- 29.Problem = The best result is 755 at 5. Generation
- 30.Problem = The best result is 755 at 10. Generation

Deneme 6:

- 1.Problem = The best result is 755 at 1. Generation
- 2.Problem = The best result is 755 at 3. Generation
- 3.Problem = The best result is 755 at 7. Generation

- 4.Problem = The best result is 755 at 6. Generation
- 5.Problem = The best result is 755 at 7. Generation
- 6.Problem = The best result is 755 at 4. Generation
- 7.Problem = The best result is 755 at 1. Generation
- 8.Problem = The best result is 755 at 3. Generation
- 9.Problem = The best result is 755 at 13. Generation
- 10.Problem = The best result is 755 at 1. Generation
- 11.Problem = The best result is 755 at 7. Generation
- 12.Problem = The best result is 755 at 4. Generation
- 13.Problem = The best result is 755 at 15. Generation
- 14.Problem = The best result is 755 at 11. Generation
- 15.Problem = The best result is 755 at 23. Generation
- 16.Problem = The best result is 755 at 7. Generation
- 17.Problem = The best result is 755 at 1. Generation
- 18.Problem = The best result is 755 at 6. Generation
- 19.Problem = The best result is 755 at 7. Generation
- 20.Problem = The best result is 755 at 2. Generation
- 21.Problem = The best result is 755 at 3. Generation
- 22.Problem = The best result is 755 at 2. Generation
- 23.Problem = The best result is 755 at 10. Generation
- 24.Problem = The best result is 755 at 2. Generation
- 25.Problem = The best result is 755 at 1. Generation
- 26.Problem = The best result is 755 at 7. Generation
- 27.Problem = The best result is 755 at 1. Generation
- 28.Problem = The best result is 755 at 2. Generation
- 29.Problem = The best result is 755 at 17. Generation
- 30.Problem = The best result is 755 at 1. Generation

Deneme 7:

- 1.Problem = The best result is 755 at 2. Generation
- 2.Problem = The best result is 755 at 16. Generation
- 3.Problem = The best result is 755 at 1. Generation
- 4.Problem = The best result is 755 at 6. Generation
- 5.Problem = The best result is 755 at 6. Generation
- 6.Problem = The best result is 755 at 1. Generation
- 7.Problem = The best result is 755 at 1. Generation
- 8.Problem = The best result is 755 at 1. Generation
- 9.Problem = The best result is 755 at 3. Generation
- 10.Problem = The best result is 755 at 9. Generation
- 11.Problem = The best result is 755 at 1. Generation
- 12.Problem = The best result is 755 at 1. Generation
- 13.Problem = The best result is 755 at 1. Generation
- 14.Problem = The best result is 755 at 1. Generation
- 15.Problem = The best result is 755 at 15. Generation
- 16.Problem = The best result is 755 at 1. Generation
- 17.Problem = The best result is 755 at 3. Generation
- 18.Problem = The best result is 755 at 2. Generation
- 19.Problem = The best result is 755 at 3. Generation
- 20.Problem = The best result is 755 at 1. Generation
- 21.Problem = The best result is 755 at 2. Generation

- 22.Problem = The best result is 755 at 6. Generation
- 23.Problem = The best result is 755 at 3. Generation
- 24.Problem = The best result is 755 at 6. Generation
- 25.Problem = The best result is 755 at 3. Generation
- 26.Problem = The best result is 755 at 28. Generation
- 27.Problem = The best result is 755 at 5. Generation
- 28.Problem = The best result is 755 at 1. Generation
- 29.Problem = The best result is 755 at 4. Generation
- 30.Problem = The best result is 755 at 5. Generation

Deneme 8:

- 1.Problem = The best result is 755 at 2. Generation
- 2.Problem = The best result is 755 at 1. Generation
- 3.Problem = The best result is 755 at 11. Generation
- 4.Problem = The best result is 755 at 1. Generation
- 5.Problem = The best result is 755 at 1. Generation
- 6.Problem = The best result is 755 at 1. Generation
- 7.Problem = The best result is 755 at 4. Generation
- 8.Problem = The best result is 755 at 13. Generation
- 9.Problem = The best result is 755 at 7. Generation
- 10.Problem = The best result is 755 at 1. Generation
- 11.Problem = The best result is 755 at 4. Generation
- 12.Problem = The best result is 755 at 3. Generation
- 13.Problem = The best result is 755 at 6. Generation
- 14.Problem = The best result is 755 at 13. Generation
- 15.Problem = The best result is 755 at 3. Generation
- 16.Problem = The best result is 755 at 1. Generation
- 17.Problem = The best result is 755 at 15. Generation
- 18.Problem = The best result is 755 at 2. Generation
- 19.Problem = The best result is 755 at 3. Generation
- 20.Problem = The best result is 755 at 2. Generation
- 21.Problem = The best result is 755 at 8. Generation
- 22.Problem = The best result is 755 at 2. Generation
- 23.Problem = The best result is 755 at 10. Generation
- 24.Problem = The best result is 755 at 1. Generation
- 25.Problem = The best result is 755 at 4. Generation
- 26.Problem = The best result is 755 at 8. Generation
- 27.Problem = The best result is 755 at 4. Generation
- 28.Problem = The best result is 755 at 9. Generation
- 29.Problem = The best result is 755 at 7. Generation
- 30.Problem = The best result is 755 at 2. Generation

ÖZGEÇMİŞ

İstanbul'da 1978 yılında doğdu. İlköğrenimini Muhsin Adil Binal İlkokulu'nda, orta ve lise eğitimini Hüseyin Avni Sözen Anadolu Lisesi'nde tamamladı. 1995 yılında Elektrik Mühendisliği bölümüne girdi ve 1999 yılında İ.T.Ü. Elektrik-Elektronik Fakültesi Elektrik Mühendisliği Bölümü'nden iyi derece ile mezun oldu. Aynı yıl İ.T.Ü. Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı Mühendislik Yönetimi Programı'nda yüksek lisansa başladı ve 2002 yılında pekiyi derece ile yüksek lisans programını tamamladı.



**Y.C. YÜKSEKÖĞRETİM KURULU
DOKÜMAN TAYIN MERKEZİ**