KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

DOKTORA TEZİ

AKIŞ TİPİ ÇİZELGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN VERİ MADENCİLİĞİ TABANLI BİR MODEL ÖNERİSİ

BURCU ÖZCAN

KOCAELİ 2013

KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

DOKTORA TEZİ

AKIŞ TİPİ ÇİZELGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN VERİ MADENCİLİĞİ TABANLI BİR MODEL ÖNERİSİ

BURCU ÖZCAN

Prof.Dr.Alpaslan FIĞLALI,
Danışman, Kocaeli Üniv.
Yrd.Doç.Dr. Gülşen AKMAN
Jüri Üyesi, Kocaeli Üniv.
Doç.Dr.Orhan ENGİN
Jüri Üyesi, Selçuk Üniv.
Yrd.Doç.Dr.Gülşen AYDIN KESKİN,
Jüri Üyesi, Kocaeli Üniv.
Yrd.Doç.Dr.Alparslan Serhat DEMİR,
Jüri Üyesi, Sakarya Üniv.



Tezin Savunulduğu Tarih: 17.06.2013

ÖNSÖZ ve TEŞEKKÜR

Çizelgeleme, üretim ve hizmet sektöründe süreçlerin zaman ve maliyet etkin yürütülmesi için önemli rol oynayan bir karar verme sürecidir. Çeşitli amaçlar için, belli kısıtlar altında işlerin veya görevlerin atanmasını sağlar.

Bu süreç tamamlandığında her ürünün/parçanın üretimine hangi tarihte başlanacağı, ne kadar üretileceği, işlemlerin hangi bölümlerde ve tezgâhlarda yapılacağı, parça ve ürünlerin işlenme öncelikleri ile işlerin bitiş zamanları ayrıntılı olarak belirlenmiş olur. Ancak çizelgeleme problemi zorluk açısından NP-zor sınıfında yer almaktadır. Küçük problemlerde kesin çözüm bulunabilmekte ancak problem boyutu büyüdükçe optimum çözüm zorlaşmaktadır. Bu nedenle bu tez kapsamında çözüm kalitesinin artırılmasına yönelik bir model önerisi sunulmaktadır.

Bu çalışmada rota birleştirme algoritması veri madenciliği teknikleri kullanılarak bir model önerisi getirilmiştir ve veri madenciliği tekniği literatürdeki alışılmış biçiminin dışında kullanılarak büyük boyutlu problemlerin çözümü mümkün olmuştur.

Çalışmanın ortaya çıkması sürecinde yardım ve desteğini hiçbir zaman esirgemeyen, değerli hocalarım Prof.Dr.Alpaslan FIĞLALI ve Doç.Dr.Mesut YAVUZ'a çok teşekkür ederim. Ayrıca beni destekleyen canım aileme de sonsuz minnet duygularımı sunarım.

Nisan- 2013 Burcu ÖZCAN

İÇİNDEKİLER

ÖNSÖZ ve TEŞEKKÜR i
İÇİNDEKİLER ii
ŞEKİLLER DİZİNİ iii
TABLOLAR DİZİNİ v
SİMGELER DİZİNİ ve KISALTMALAR vii
ÖZET ix
ABSTRACT x
GİRİŞ 1
1. İŞ SIRALAMA ve ÇİZELGELEME 4
1.1. Çizelgeleme Problemlerine Giriş 4
1.2. İş Sıralama ve Çizelgeleme Problemlerinin Sınıflandırılması 5
1.2.1. Olası makine ortamında belirtilen α alanları 6
1.2.2. Olası makine ortamında β alanında kullanılan semboller 10
1.2.3. Olası makine ortamında γ alanında kullanılan semboller 11
1.2.4. Bazı problemlerin tanımsal gösterimi (α | β | γ) 13
1.2.5. Temel önceliklendirme kuralları 14
1.3. Algoritmanın Karmaşıklığı 14
1.4. Akış Tipi Çizelgeleme Problemleri için Yapılan çalışmalar 15
2. VERİ MADENCİLİĞİ TABANLI OPTİMİZASYON ÇALIŞMALARI 17
2.1. Büyük Ölçekli Problemlerin Çözümünde Veri Madenciliği Uygulaması 32
3. DENEY TASARIMI VE DENEY TASARIMI YÖNTEMLERİ 40
3.1. Deney Tasarım Yöntemleri 43
3.1.1. Bir kerede bir faktör 44
3.1.2. İstatistiksel deney tasarımı 44
3.1.3. Tam faktöriyel deney tasarımı 44
3.1.4. Kesirli faktöriyel deney tasarımı 44
3.1.4.1. Taguchi yöntemi 44
3.2. Taguchi Deneysel Tasarım Metodunun Prosedürleri 47
3.2.1. Çözülecek problemin belirlenmesi 47
3.2.2. Faktörlerin ve seviyelerin belirlenmesi 47
3.2.3. Ortogonal diziler ve seçimleri 48
3.2.4. Faktörlerin kolonlara atanması 49
3.2.5. Deneylerin gerçekleştirilmesi ve verilerin toplanması 49
3.2.6. Verilerin analiz edilmesi ve optimum seviye seçimi 50
4. ROTA BİRLEŞTİRME ALGORİTMASI 51
5. AKIŞ TİPİ PROBLEMLER İÇİN MODEL ÖNERİSİ 57
6. SONUÇLAR ve ÖNERİLER 96
KAYNAKLAR 98
KİŞİSEL YAYINLAR ve ESERLER 105
ÖZGEÇMİŞ 108

ŞEKİLLER DİZİNİ

Şekil 1.1. Problem tiplerinin gösterimi 5
Şekil 1.2. Akış tipi çizelgelemede M adet seri makine 7
Şekil 1.3. Çok işlemcili esnek akış tipi çizelgeleme 8
Şekil 1.4. Atölye tipi çizelgeleme 9
Şekil 1.5. Öncelik kısıtları 10
Şekil 1.6. Geç kalma süresi, pozitif geç kalma, teslim edilen iş sayısı grafikleri 12
Şekil 1.7. Algoritmanın karmaşıklığı 15
Şekil 2.1. Atölye tipi iş sıralama problemi için elde edilen optimum diziliş 19
Şekil 2.2. İşlem zamanı ve sıklıkları 20
Şekil 2.3. Arta kalan işlem süresinin sıklığı ve sınıfları 20
Şekil 2.4. Arta kalan işlem süresinin sıklığı ve sınıfları 22
Şekil 2.5. Ortalama makine yükü ve her makine için toplam yük 27
Şekil 2.6. Arta kalan işlem sürelerin sıklıkları 28
Şekil 2.7. Elde edilen kurallar 28
Şekil 2.8. On işli tek makine problemi için karar ağacı 30
Şekil 2.9. Beş iş iki makine akış tipi problemi için karar ağacı 31
Şekil 2.10. En iyi iş sırasını belirleme aşamaları 33
Şekil 2.11. İşlem süresi genellemesi 36
Şekil 2.12. Makine yüklemesi genellemesi 37
Şekil 2.13. Arta kalan işlem süresinin genellemesi 37
Şekil 2.14. Elde edilen kurallar seti 39
Şekil 3.1. Performans karakteristiği, ürün parametreleri, gürültü faktörleri 41
Şekil 3.2. İki seviye üç faktör için tüm olası kombinasyonları 44
Şekil 3.3. Kesirli deney tasarımı 45
Şekil 4.1. Rota birleştirme algoritma kodu 55
Şekil 5.1. Model önerisinin akış şeması 58
Şekil 5.2. S/G oranlarına göre parametre seviyeleri 61
Şekil 5.3. Rota birleştirme algoritmasının uygulanması 62
Şekil 5.4. Rota birleştirme algoritma sonrası elde edilen iş sıraları 63
Şekil 5.5. Z dağılımı 70
Şekil 5.6. Kritik iş sıralarının konumlandırılması 74
Şekil 5.7. Rota birleştirme algoritması 76
Şekil 5.8. Rota birleştirme ve veri madenciliği algoritmasının birleştirilmesi 77
Şekil 5.9. S/G oranlarına göre parametre seviyeleri (20x5) 79
Şekil 5.10. Elde edilen değerlerin karşılaştırılması (20x5) 80
Şekil 5.11. S/G oranlarına göre parametre seviyeleri (20x10) 82
Şekil 5.12. Elde edilen değerlerin karşılaştırılması (20x10) 83
Şekil 5.13. S/G oranlarına göre parametre seviyeleri (20x20) 85
Şekil 5.14. Elde edilen değerlerin karşılaştırılması (20x20) 86
Şekil 5.15. S/G oranlarına göre parametre seviyeleri (50x5) 88
Şekil 5.16. Elde edilen değerlerin karşılaştırılması (50x5) 89
Şekil 5.17. S/G oranlarına göre parametre seviyeleri (50x10) 91
Şekil 5.18. Elde edilen değerlerin karşılaştırılması (50x10) 92

Şekil 5.19. S/G oranlarına göre parametre seviyeleri (100x10) 94
Şekil 5.20. Elde edilen değerlerin karşılaştırılması (100x10) 95

TABLOLAR DİZİNİ

Tablo 1.1. Paralel m adet özdeş makine örneği 6
Tablo 1.2. Paralel m adet farklı işleme hızlarındaki makine örneği 6
Tablo 1.3. Paralel m adet özdeş olmayan makine örneği 7
Tablo 1.4. Amaç fonksiyonları ve formülleri 13
Tablo 2.1. 6 iş 6 makine atölye tipi iş sıralama problemi 19
Tablo 2.2. Arta kalan işlem süreleri 21
Tablo 2.3. Altı makine altı işli atölye tipi iş sıralama problemi için kurallar 22
Tablo 2.4. Tahminlenen kurallara göre işgücü düzenlenmesi 24
Tablo 2.5. Sekiz iş dört makine problemi için veriler 25
Tablo 2.6. Sekiz iş dört makine problemi için feromon miktarları 25
Tablo 2.7. Optimal farklı iş sıraları 26
Tablo 2.8. Her makine için toplam iş yükü 26
Tablo 2.9. Arta kalan işlem süreleri 27
Tablo 2.10. İşlem süresi ve operasyon sınıfları 28
Tablo 2.11. Beş iş iki makine problemi için değerlendirme tablosu 31
Tablo 2.12. Veri madenciliği yazılımları ve teknikleri 34
Tablo 2.13. Karar ağacı araçlarının karşılaştırılması 35
Tablo 2.14. Sınırlar ve kodlar 36
Tablo 2.15. Özelliklerin detaylı sınıflanması 38
Tablo 2.16. Sınıf ve veri dosyası 39
Tablo 3.1. Faktörlerin seviyeleri ve isimlendirilmesi 45
Tablo 3.2. Kesirli deney tasarımı için desenler 46
Tablo 3.3. Eşit seviyeli deney tasarımları 47
Tablo 3.4. Karma seviyedeki tasarımlar 48
Tablo 3.5. Beş faktör üç seviyeli ortogonal dizi tablosu 49
Tablo 4.1. Rota birleştirme algoritması için örnek çözüm 56
Tablo 5.1. Deney tasarımı için belirlenen faktörler ve seviyeler 59
Tablo 5.2. Taguchi deney tasarımı L27 (3^5) 60
Tablo 5.3. Problem büyüklüklerine göre elde edilen en iyi parametre seviyeleri 61
Tablo 5.4. Rassal oluşturulmuş grup ortalamasının bulunması 64
Tablo 5.5. Grup ortalama tablosu 65
Tablo 5.6. Rassal çözüm için konumlara göre ortalamaların bulunması 65
Tablo 5.7. Standart sapma tablosu (Örnek) 67
Tablo 5.8. Standart sapma tablosu 68
Tablo 5.9. Rassal çözüm için konumlara göre standart sapmaların bulunması 69
Tablo 5.10. Z Tablosu 71
Tablo 5.11. Rassal çözümler için Z değerinin bulunması 71
Tablo 5.12. Optimum Çözüm Alternatifleri 73
Tablo 5.13. Konuma göre atama 75
Tablo 5.14. Deney tasarımı için belirlenen faktörler ve seviyeler (20x5) 78
Tablo 5.15. L27 (3^5) Taguchi tasarımı (20x5) 78
Tablo 5.16. Elde Edilen Sonuçlar (20x5) 79

Tablo 5.17. Deney tasarımı için belirlenen faktörler ve seviyeler (20x10) 81
Tablo 5.18. L27 (3^5) Taguchi tasarımı (20x10) 81
Tablo 5.19. Elde Edilen Sonuçlar (20x10)..... 82
Tablo 5.20. Deney tasarımı için belirlenen faktörler ve seviyeler (20x20) 83
Tablo 5.21. L27 (3^5) Taguchi tasarımı (20x20) 84
Tablo 5.22. Elde Edilen Sonuçlar (20x20)..... 85
Tablo 5.23. Deney tasarımı için belirlenen faktörler ve seviyeler (50x5) 86
Tablo 5.24. L27 (3^5) Taguchi tasarımı (50x5) 87
Tablo 5.25. Elde Edilen Sonuçlar (50x5)..... 88
Tablo 5.26. Deney tasarımı için belirlenen faktörler ve seviyeler (50x10) 89
Tablo 5.27. L27 (3^5) Taguchi tasarımı (50x10) 90
Tablo 5.28. Elde Edilen Sonuçlar (50x10)..... 91
Tablo 5.29. Deney tasarımı için belirlenen faktörler ve seviyeler (100x10) 92
Tablo 5.30. L27 (3^5) Taguchi tasarımı (100x10) 93
Tablo 5.31. Elde Edilen Sonuçlar (100x10)..... 94

SİMGELER DİZİNİ ve KISALTMALAR

C_j : Completion Time (j İşinin Tamamlanma Süresi)
C_{max} : Maksimum Tamamlanma Süresi
D_j : Due date (Teslim Zamanı)
E_{max} : Maksimum Erken Tamamlanma
FFc : Flexible Flow Shop (Esnek Akış Tipi)
J_m : Job Shop (Atölye Tipi)
L_j : Lateness (Geç kalma süresi)
L_{max} : Maximum Lateness (Maksimum Pozitif Geç kalma Süresi)
O_m : Open Shop (Açık Atölye Tipi)
P_{ij} : Processing Time (İşlem Zamanı)
R_{ij} : Release Date (Serbest Zaman)
S_{jk} : Sequence Dependent Setup Times (Sıraya Bağlı Hazırlık Süresi)
S_{jo} : j İşinden Sonra Temizlik Zamanı
T_j : Tardiness (Pozitif Geç Kalma Süresi)
U_j : Geç Teslim Edilen İş Sayısı
V_{ij} : J İşinin i Makinesindeki Hızı
W_j : Job Weight (İşin Ağırlığı)
w_jT_j : Total Weighted Tardiness (Toplam Ağırlıklı Gecikme)
w_jU_j : Weighted Number Of Tardy Jobs (Ağırlıklı Geç Kalan İş Sayısı)

Kısaltmalar

brkdwn : Breakdowns (Arıza)
CDS : Campbell, Dudek and Smith Sezgiseli
ÇİEATÇ : Çok İşlemcili Esnek Akış Tipi Çizelgeleme
EATÇ : Esnek Akış Tipi Çizelgeleme
EDD : Earliest Due Date (En Erken Teslim Zamanı)
FIFO : First in First Out (İlk Giren İlk Çıkar)
FJc : Flexible Job Shop (Esnek İş Atölyesi)
Fm : Flow Shop (AkışTipi)
FMLS : Job Families (İş Ailesi)
GRASP : Greedy randomized adaptive search procedure (Aç gözlü-rastgele adaptif arama yöntemi
LPT : : Longest Processing Time (En Uzun İşlem Süreli İşlem Zamanı)
MS : Minimum Slack (En Düşük Serbestlik)
NEH : Nawaz, Enscore ve Ham Sezgiseli
NP : Non Polynomial Time (Polinomsal zamanda çözülemeyen)
nwt : No-wait (Beklemesiz)

PFSP : Permutation Flow Shop Problem (Permütasyon Tipi Akış Problemi)
prec : Precedence Constraints (Öncelik Kısıtları)
prmu : Permutation (Permütasyon)
rcrc : Recirculation (Yeniden Dolaşım)
SIRO : Service In Random Order (Rassal seçim)
SPT : Shortest Processing Time (En Kısa İşlem Süreli İşlem Zamanı)
SST : Shortest Setup Time (En Kısa Hazırlık Süreli İşlem Süresi)
VM : Veri Madenciliği

AKIŞ TİPİ ÇİZELGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN VERİ MADENCİLİĞİ TABANLI BİR MODEL ÖNERİSİ

ÖZET

Üretim faaliyetlerinin zamanında, etkin ve düşük maliyetle yapılabilmesi için nerede ve ne zaman gerçekleştirileceğine karar vermeyi sağlayan çizelgeleme konusu büyük önem arz etmektedir. Çizelgeleme işi küçük boyutlu problemlerde makul sürelerde ve iyi çözüm kalitesi elde edilebilecek şekilde yapılmaktadır, ancak problem boyutu büyüdüğünde çözüm süresi ve kalitesi açısından olumsuzluklar görülmektedir.

Bu çalışmada çizelgeleme problemleri içersinde önemli bir yer tutan akış tipi çizelgeleme problemlerinin çözümüne yönelik bir algoritma önerilmektedir.

NP-zor sınıfında yer alan akış tipi çizelgeleme problemlerinin çözümünde veri madenciliği ve rota birleştirme algoritması birlikte kullanılmıştır. Veri madenciliği ile çözüm uzayında global arama yapılırken, rota birleştirme algoritması ile yerel aramanın yapılması hedeflenmiştir. Veri madenciliği çok sayıda veri içeren bir küme içerisinde gömülü olan bilgilerin çıkarılmasında kullanılan bir yöntemdir. Rota birleştirme algoritması ise bir başlangıç çözümünü bir hedef çözümüne dönüştürmek için ikili değiştirmeler yaparak ilerleyen ve bu süreçte elde edilen en iyi çözümü başlangıç noktası olarak atayarak tekrar eden bir algoritmadır. Modelin etkinliği Taillard'ın akış tipi çizelgeleme problemleri üzerinde test edilmiştir.

Çalışılan problemlerde iş sayısı 20 ile 100 arasında değişmektedir, her bir problem büyüklüğü için modelin parametrelerinin en uygun seviyelerinin belirlenmesi amacıyla deney tasarımı yapılmıştır. Beş faktör üç seviyeli L27 düzeneği oluşturulmuştur ve bu seviyelere göre çok sayıda farklı optimum çözüm elde edilmiştir. Bu çalışmayla veri madenciliği tekniği literatürdeki alışılmış biçiminin dışında kullanılmış, önemli bir matematik altyapısı gerektirmeksizin büyük boyutlu problemlerin çözümü mümkün olmuştur. Test problemlerinin çözümleriyle elde edilen sonuçlar, önerilen yöntemin diğer metasezgisel yöntemlerle rekabet edebilecek düzeyde olduğunu göstermektedir.

Anahtar Kelimeler: Akış Tipi Çizelgeleme, Rota Birleştirme Algoritması, Sezgisel Yöntem, Veri Madenciliği.

DATA MINING BASED MODEL FOR FLOWSHOP SCHEDULING PROBLEMS

ABSTRACT

The scheduling theme which determines where and when the manufacturing actions will be performed is important to perform the activities in time, efficiently and cost-effectively. Scheduling is performed for the small-size problems to be able to get the best solution in a reasonable period of time. However, when the size of problem is increased, some disadvantages of the solution in terms of time and quality are observed.

In this paper, an algorithm is proposed for the solution of the flow shop scheduling which holds an important place in scheduling problems.

The path relinking algorithm and data mining are used together for the solution of the flow shop scheduling problems which take a place in the NP class: while it is being performed a global research in the solution space by using the Data Mining, it is aimed to perform a local research by using the pathrelinking algorithm. Data mining is a method which is used to get the embedded info in a cluster that includes many info. Pathrelinking is an algorithm which advances by making binary displacements in order to convert the initial solution to the guiding solution and which repeats by assinging the best obtained solution within this process as to the starting point. The efficiency of the model was tested on Taillard's flow shop scheduling problems.

The number of job in the problems changes between 20 and 100, design of experiment was used to define the optimal level of the model parameters for the each problem size. Five factors three level L27 orthaganal array was prepared and various different optimum solutions were obtained in these level. In this study, data mining method was used in unusual formats within the literature, it has been possible to solve the large-size problem without the considerable mathematical backround. The obtained results show that the proposed method is in the competitiveness level with the other metaheuristic methods.

Keywords: Flow Shop Scheduling, Path Relinking, Heuristic, Data Mining.

GİRİŞ

Hızla gelişen teknolojik donanımlara rağmen günümüzde hala NP grubundaki büyük ölçekli problemler için optimum çözümün elde edilmesi çok uzun zaman almaktadır. Endüstri Mühendisliği alanından örnek vermek gerekirse; gezgin satıcı problemi, çizelgeleme problemleri, kuadratik atama problemi vb. problemlerin en iyi çözümlerini bulabilmek için günümüze kadar farklı yaklaşımlar içeren optimizasyon modelleri ve çeşitli yöntemler geliştirilmiştir. Geliştirilen bu modeller temel olarak üç grupta incelenebilir:

1. Dal-sınır algoritması, doğrusal programlama modelleri, dinamik programlama gibi kesin çözümü hedefleyen matematik programlama temelli modeller.
2. Probleme özgü geliştirilen, optimum çözümü garanti etmeyen ancak makul sürelerde optimuma yakın çözümler verebilen sezgisel modeller.
3. Genel amaçlı olarak kullanılabilen, makul çözüm sürelerinde optimuma yakın sonuçlar veren tavlama benzetimi, tabu arama, yapay sinir ağları, genetik algoritma vb. yapay zekâ tabanlı metasezgisel modeller.

Birinci grupta yer alan modellerle optimum çözümün elde edilmesi garanti edilebilmekle beraber genel olarak çözüm süresi problem büyüklüğü ile üssel olarak artmakta ve üssel çözüm süresi gerekmektedir. Problem boyutu büyüdükçe optimum çözümün elde edilebilmesi için gereken zaman kabul edilebilir sınırların çok üzerine çıkmaktadır. Sezgisel yöntemlerin zayıf tarafı probleme özgü olarak geliştirilmeleri ve farklı problem türleri için uygulanamamalarıdır. Bahsedilen bu sakıncaların üstesinden gelebilmek amacıyla son dönemlerde metasezgisel yöntemler optimizasyon alanında yaygın olarak kullanılmaktadır. Metasezgisel yöntemlerin önemli bir avantajı da önemli bir matematik altyapısı gerektirmeksizin çözüme ulaşabilmeleridir.

Metasezgisellerin başarılı çözümler verebilmesi için en önemli koşul global arama ve yerel arama mekanizmalarının uyumlu ve dengeli biçimde çalışabilmesi; bunun yanı sıra metasezgiselin içsel parametrelerinin deney tasarımı vb. yöntemlerle uygun şekilde belirlenmesidir. Metasezgisellerin uygun olmayan arama mekanizmaları veya uygun olmayan parametrelerin seçilmiş olması nedeniyle yerel optimumlara yakınsaması önemli bir problemdir.

Bu çalışmada metasezgisellerin yukarıda sayılan problemlerini içermeyecek, benzer çözüm kalitesine rekabet edebilir sürelerde ulaşacak ve matematik altyapısı açısından da basit bir model geliştirilmek amacıyla yola çıkılmıştır.

Veri Madenciliği (VM), büyük ölçekli veriler içinde saklı kalmış anlamlı bilgiye ulaşmak için, sahip olunan verileri analiz ederek, yorumlama sürecidir. Veri madenciliğinin optimizasyon alanındaki uygulamasında kısa süre içerisinde rassal olarak üretilen çok sayıdaki çözüm içerisinden seçilen en iyi çözümlerin içerdiği saklı ve anlamlı bilgi yardımıyla optimum veya optimuma yakın çözümlere ulaşılmaya çalışılmaktadır. Veri madenciliği yaklaşımı ile bulunan en iyi çözümlere yerel arama uygulanarak çözüm performansının arttırılması da bir strateji olarak değerlendirilmektedir.

Üretim planlama alanındaki konulardan biri olan, zorluk açısından NP zor grubunda yer alan, iş sıralama çizelgeleme problemi incelenecektir. Küçük problemlerde kesin çözüm yöntemleri (dal sınır gibi matematiksel yöntemler) kısa zamanda optimum çözümü bulabilirken, pratik hayatta daha çok karşılaştığımız büyük problemler için ise bu matematiksel yöntemlerin uygulanması uzun zaman almaktadır. Büyük boyutlu problemler için uygulanabilecek seçenekler arasında CDS, Johnson, Moore, NEH gibi algoritmaların yanı sıra SPT, FIFO, EDD vb. öncelik kuralları bulunmaktadır. Çizelgeleme problemlerinde son dönemlerde yaygın olarak kullanılan diğer bir seçenek ise genetik algoritmalar, tavlama benzetimi, karınca kolonileri optimizasyonu, yapay bağışıklık sistemi gibi metasezgisel yöntemlerdir.

Bu çalışmada rassal olarak çok sayıda iş sırası üretilecek, belirlenecek amaç fonksiyonu açısından en iyi değere sahip belirli sayıda iş sırası seçilecek ve bu çözümler arasındaki ilişkilerin belirlenmesi amacıyla veri madenciliği yöntemi kullanılacaktır. Veri madenciliği ile geliştirilen algoritmanın rekabet edebilir bir

seçenek haline gelebilmesi için rota birleştirme algoritması uygulanacaktır. Önerilen modelin iç parametrelerinin optimizasyonu amacıyla da deney tasarımı yöntemi kullanılacaktır.

Çalışmanın birinci bölümünde iş sıralama ve çizelgeleme kavramları, notasyonları ve bu alandaki problemler incelenmiştir.

İkinci bölümde ise, veri madenciliği yönteminin optimizasyon alanında hangi problem tiplerine uygulandığı ve elde edilen sonuçlar özetlenmiştir. Literatürdeki çalışmalardan yöntemin küçük boyutlu problemlere uygulandığı görülmüş, Bölüm 2.1'de literatürde, yöntemin büyük boyutlu problemlerdeki performansı araştırılmıştır.

Önerilen modelde çok sayıda parametrenin olması nedeniyle parametre optimizasyonu yapılması gerekmektedir. Bu amaçla kullanılan Taguchi deney tasarımı konusunda yapılan çalışmalar, deney tasarım yöntemleri, Taguchi deney tasarımı yönteminin aşamaları üçüncü bölümde özetlenmiştir.

Dördüncü bölümde önerilen modelin çözüm kalitesini artırmak amacıyla kullanılan rota birleştirme algoritmasıyla ilgili literatür çalışmaları özetlenmiş ve algoritma anlatılmıştır.

Beşinci bölümde önerilen modelin performansı Taillard'ın akış tipi çizelgeleme problemleri üzerinde test edilmiştir. Çalışılan problemlerde iş sayısı 20 ile 100 arasında değişmektedir ve NP zor grubunda yer aldıkları bilinmektedir. Bu problemler için şu ana kadar elde edilmiş en iyi sonuçlar ile model tarafından elde edilen en iyi sonuçlar birbiriyle kıyaslanmıştır.

1. İŞ SIRALAMA ve ÇİZELGELEME

Bu bölümde, iş sıralama ve çizelgelemenin önemi, iş sıralama sınıflandırılması, algoritmanın karmaşıklığı ve akış tipi çizelgeleme problemleri için yapılan çalışmalar özetlenmiştir.

1.1. Çizelgeleme Problemlerine Giriş

Çizelgeleme, üretim ve hizmet sektöründe önemli rol oynayan bir karar verme prosesi olup tedarik ve üretimde, taşıma ve dağıtımda, bilgi işleme ve iletişim gibi pek çok alanda kullanılmaktadır. Günümüz rekabet ortamında işletmelerin siparişlerini zamanında teslim etmeleri açısından, etkin sıralama ve çizelgeleme kritik öneme sahiptir. Aksi takdirde işletme için prestij kaybı söz konusu olmaktadır. Bu bağlamda, kıt kaynakların belirlenmesi ve kaynakların bu doğrultuda atanması gerekmektedir. Buradaki kıt kaynaklar bir atölyedeki tezgâhlar, bir hava alanındaki pistler, bir inşaattaki çalışanlar veya bir bilgisayardaki işlem üniteleri olabilir. Görevler ise üretimdeki işlemler, havaalanındaki iniş ve kalkışlar, inşaat projesindeki safhalar veya çalıştırılması gereken bilgisayar programları olabilir. Her bir görev bir öncelik seviyesine, mümkün en erken başlama zamanına veya en geç tamamlanma zamanına sahip olabilir. Bunun yanı sıra performans kriterleri de çeşitli biçimlerde ortaya çıkabilmektedir. Örneğin tüm görevlerin yapılması için gereken zamanın veya geciken görev sayısının en aza indirilmesi vb. olabilir (Pinedo, 1999).

Hazırlık işlemleri ile ilgili problemler literatürde iki sınıfta ele alınmıştır. Birincisinde, hazırlıklar sadece işlem görecek işe bağlı olup sıra-bağımsız hazırlık zamanı olarak ifade edilmektedir. Diğerinde ise hazırlık, hem o anda işlem görecek işe hem de bir önceki işe bağlıdır. Bu durum ise sıra-bağımlı hazırlık zamanı olarak ifade edilmektedir (Allahverdi ve diğ., 1999). Sıra-bağımlı hazırlık zamanı uygulamaları ile ilgili örnek olarak matbaa endüstrisi verilebilir. Burada makinenin temizlenmesi ve hazırlanması, en son kullanılan mürekkep rengine, kâğıdın boyutuna ve özelliğine bağlıdır. Ayrıca kimya, ilaç, metal endüstrilerinde de sıra-bağımlı hazırlık zamanı uygulamalarına sıkça rastlanmaktadır (Yang ve Liao, 1999).

Tüm iş sıralama ve çizelgeleme problemlerinde, makinelerin ve işlerin sonlu olduğu farz edilir. İşlerin sayıları n ile, makineler ise m ile gösterilir. Genellikle i alt indisi; bir makineyi, j alt indisi; işi ifade eder. Eğer iş çok sayıda işlem adımları veya operasyonlardan oluşuyorsa, makine i de j işinin işlenme adımı (i,j) ikilisi şeklinde gösterilir.

İşleme zamanı: p_{ij} makine i de, j işinin işleme zamanını gösterir. Eğer j işinin işleme zamanı, makineye bağlı değilse veya j işi sadece verilen bir makinede işleniyorsa alt indis i çıkarılır.

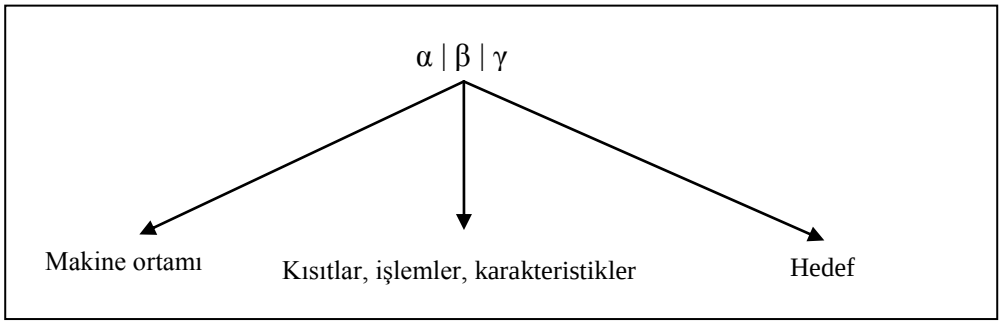
Geliş zamanı: r_j, j işinin geliş zamanı, hazır olma zamanını gösterir. j işinin işleme başlayabileceği en erken zamandır.

Teslim zamanı: d_j, j işinin teslim zamanı, tamamlanma zamanını veya sevkiyatın teslim zamanını gösterir. İşin tamamlanması, teslim zamanından sonraya bırakılırsa, ceza oluşur. Eğer teslim süresi zamanında karşılanırsa, gösterimi $\bar{d}_j$ şeklinde olur.

Ağırlık: w_j, j işinin ağırlığını gösterir ve j işinin sistemdeki diğer işler arasındaki nispi önemini gösteren öncelik faktörüdür. Örneğin, bu ağırlık sistemdeki işlerin gerçek maliyetlerini gösterebilir. Bu maliyet elde bulundurma veya stok maliyeti olabilir. Bu, işe yüklenen değeri de gösterebilir.

1.2. İş Sıralama ve Çizelgeleme Problemlerinin Sınıflandırılması

Bir çizelgeleme problemi Şekil 1.1'deki gibi ifade edilebilir ($\alpha \mid \beta \mid \gamma$). α alanı makine ortamını, β alanı işlem karakteristikleri ve kısıtlarını gösterir. γ alanı optimize edilecek hedefi belirtir.



Şekil 1.1. Problem tiplerinin gösterimi

1.2.1. Olası makine ortamında belirtilen α alanları

1. Tek makine (1): Tek makine, makine ortamında mümkün olan en kolay ve diğer tüm karışık makine ortamının özel bir durumudur.

2. Paralel m adet özdeş makine (P_m): Paralel m tane özdeş makine vardır. j işi tek işlem gerektirir ve m adet makinenin herhangi birisinde veya verilen alt kümeye ait herhangi bir makinede işlenebilir. Tablo 1.1'de görüldüğü üzere, makine i'nin hızı v_i ile gösterilir.

Tablo 1.1. Paralel m adet özdeş makine örneği

	Makine 1	Vi	Makine 2	Vi
İş A	5	1	5	1
İş B	3	1	3	1

3. Farklı işleme hızlarındaki paralel m adet makine (Q_m): Farklı hızlarda m adet paralel makine vardır. Tablo 1.2'de farklı işleme hızlarına sahip makine ortamı bulunmaktadır. Makine i de j işinin işlem zamanı P_{ij}, P_j/v_i ye eşittir. Bu ortam, üniform makineler diye adlandırılacaktır. Tüm makinelerin hızları aynı ise (mesela $v_i=1$ her i için ve $p_{ij}=p_j$), durum bir önceki ortama benzeyecektir.

Tablo 1.2. Paralel m adet farklı işleme hızlarındaki makine örneği

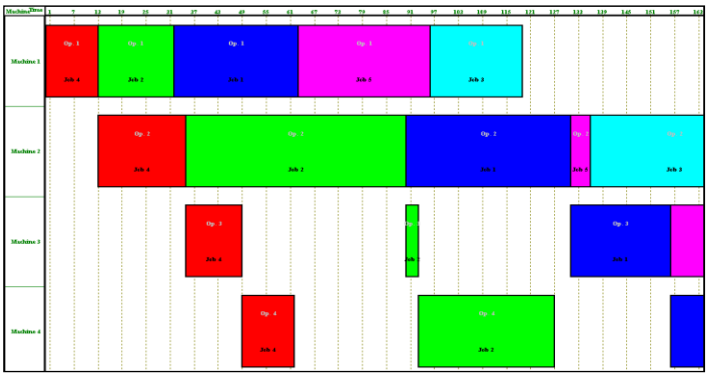
	Makine 1	Vi	Makine 2	Vi
İş A	5	0.8	5	0.6
İş B	3	0.8	3	0.6

4. Paralel m adet özdeş olmayan makine (R_m): Bu ortam bir önceki ortamın genelleştirilmiş halidir. Paralel m sayıda farklı makine vardır. Tablo 1.3'de özdeş olmayan farklı işleme hızlarına sahip makine ortamı bulunmaktadır. Makine i, j işini v_{ij} hızıyla işleyebilir. Makine i de j işinin işlem zamanı P_{ij}, P_j/v_{ij} ye eşittir. Eğer makinelerin hızları, işlerden bağımsızsa (mesela her i ve j için $v_{ij}=v_i$), durum bir önceki ortama benzeyecektir.

Tablo 1.3. Paralel m adet özdeş olmayan makine örneği

	Makine 1	V_i	Makine 2	V_i
İş A	5	0.8	5	0.7
İş B	3	0.6	3	0.9

5. m adet seri makine (akış tipi, F_m): m adet seri makine vardır. Her bir iş herhangi bir m makinesinde işlenebilir. Şekil 1.2'de görüldüğü üzere tüm işler aynı rotayı takip etmek zorundadır (Örneğin işler ilk olarak makine 1 de sonra makine 2 de işlem görmelidir). İş, ilk makinede tamamlandıktan sonra, sonraki makinede kuyruğa girer. Genellikle tüm kuyruktaki işlerin ilk giren ilk çıkar (FIFO) disiplini altında işleme gireceği farzedilir. Bu yüzden kuyrukta bekleme esnasında, bir iş kuyrukta bekleyen diğer işi geçemez. Eğer FIFO disiplini etkinse, permütasyon akış tipi olarak adlandırılır ve β alanı permütasyon içerir.

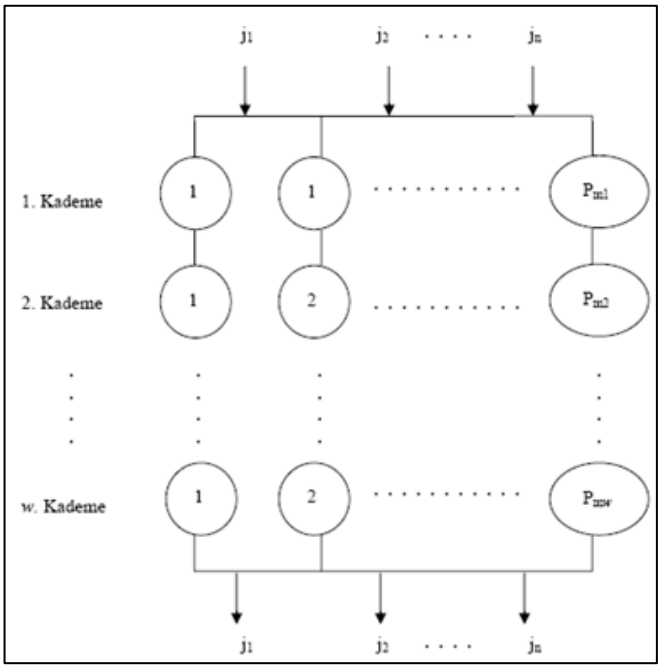


Şekil 1.2. Akış tipi çizelgelemede M adet seri makine

6. Esnek akış tipi (FFc): Klasik akış tipi ve paralel makine problemlerinin genelleştirmesi şeklindedir. Her aşamada birbirine paralel çok sayıda özdeş makine seri halde bulunmaktadır. Her iş ilk olarak aşama birde, sonra aşama ikide ve diğerlerinde işlenmek zorundadır. İşlerin önceliği yoktur. Bir aşamadaki kuyruk FIFO disiplinine göre işlenebilir ya da işlenmeyebilir. Her makinede belli bir anda sadece bir işlem görebilir.

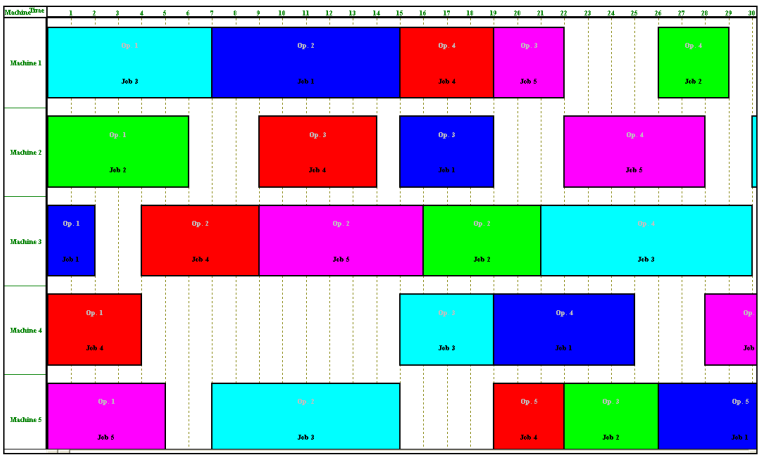
Şekil 1.3'de bir esnek akış tipi sistemin yapısı verilmiştir. Makineler w tane seri kademeye yerleştirilmişlerdir. W=(1, 2… w) olmak üzere bir w kademesinde, bir veya daha fazla eş makine bulunmaktadır. j=1,2,.....,n olmak üzere bir j işi, önce 1. kademede sonra 2. kademede ve son olarak w. kademede işlem görür. Her j işi aynı

zamanda sadece bir makinede ve her bir kademedeki makinelerden herhangi birinde işlem görür. İşlem gören her j işi, $O=O_{1J},\ldots,O_{mJ}$ olmak üzere bir operasyon zinciri oluşturur. Farklı kademelerde j işinin m_w makine için bir O_{mj} operasyonunda p_{1j},p_{2j}, ...,p_{wj} olmak üzere p_{wj} işlem süreleri vardır. Her makine aynı zamanda en fazla bir işi işleyebilir. Bir operasyona sadece ondan önceki operasyon tamamlandıktan sonra başlanabilir. Tüm işlerin ve tüm makinelerin çizelgeleme süresince her zaman hazır olduğu varsayılmaktadır. Amaç, işlerin tamamlanma zamanını minimize edecek çizelgeyi bulmaktır (Öztürk, 2007; Oğuz ve Ercan, 2005).



Şekil 1.3. Çok işlemcili esnek akış tipi çizelgeleme

7. Atölye tipi (J_m): m makineli bir iş atölyesinde her bir iş, önceden tayin edilen rotaya sahiptir. Her bir işin her makineye en az bir kere uğrayacağı iş atölyeleri ve her makineye birden fazla uğrayacağı iş atölyeleri arasında bir ayrım vardır. Şekil 1.4'de atölye tipi çizelgeleme problemine ilişkin gantt şeması verilmektedir.



Şekil 1.4. Atölye tipi çizelgeleme

8. Esnek atölye tipi (FJc): Atölye tipine benzemektedir ancak her aşamada c sayıda özdeş makine vardır. Esnek iş atölyesi, çok sayıda paralel makinenin bulunduğu, atölye tipinin (J_m) genelleştirilmiş halidir. Her bir iş merkezinde çok sayıda özdeş makine bulunmaktadır.

9. Açık atölye modelleri (Om): m sayıda makine vardır. Her iş her bir m makinesinde işlenmek zorundadır. Bu işleme zamanlarından bazıları sıfır olabilir. Çeşitli işler farklı rotalara sahip olabilir, rota kısıtı yoktur. İşlerin rotasının esnek bir sırada gerçekleştirilebilmesi söz konusudur.

Açık atölye çizelgeleme problemi, J_1,J_2, . . . ,J_n den oluşan işler kümesi M_1,M_2,...,M_m den oluşan makineler kümesinde işlenmek zorundadır. Bir makinede işin işlenmesi operasyon anlamına gelir ve kuyruk önemsizdir. Tüm operasyonlardaki işlem zamanlarının verildiği farz edilir. Her makine bir seferde en çok bir iş ve her iş bir seferde en çok bir kere işlenebilir (Pinedo, 2002).

Açık atölye tipi çizelgeleme problemiyle çok sayıda endüstriyel alanda karşılaşılmaktadır. Örneğin uzmanlaşmış iş istasyonları olan hava araçları hangarlarında bir uçak motorunda ve elektrik devresinde tamir gerekebilir. Bu iki görev herhangi bir sıra olmadan gerçekleştirilebilir fakat bu iki görevi aynı uçak üzerinde eş zamanlı gerçekleştirmek mümkün değildir. Diğer açık atölye uygulama alanları otomobil tamiri, kalite kontrol merkezleri, yarı iletken imalatı, öğretmen sınıf ataması, sınav çizelgeleme ve uydu iletişimidir (Andresen ve diğ., 2008).

1.2.2. Olası makine ortamında β alanında kullanılan semboller

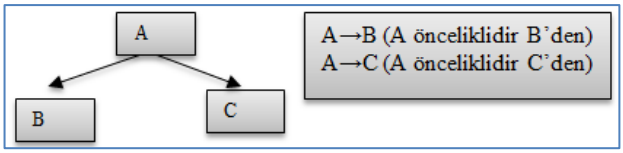
İşlem kısıtları ve şartları β alanında belirlenir.

1. Geliş zamanı (r_j): Eğer bu sembol β alanında gösteriliyorsa, j işi geliş zamanından önce işleme başlayamaz. Eğer bu sembol β alanında gösterilmiyorsa, j işinin işlemi herhangi bir zamanda başlayabilir.

2. Sıraya bağlı hazırlık süreleri (s_{jk}): s_{jk}, j ve k işleri arasındaki sıraya bağlı hazırlık sürelerini belirtir. Eğer k işi kuyrukta ilk sırada ise, s_{ik} k işi için hazırlık süresini gösterir ve eğer j işi kuyrukta sonda ise, s_{jo} j işinden sonra temizlik zamanıdır (s_{ok} ve s_{jo} sıfır olabilir). Eğer j işi ve k işi arasındaki hazırlık süresi makineye bağlı ise, alt indis i içerir (örneğin s_{ijk}). Eğer β alanında hiç s_{jk} yoksa tüm hazırlık süreleri sıfır veya kuyruğun bağımsız olduğu varsayılır.

3. Bölünebilme (prmp): İşin tamamlanabilmesi için makinede sonuna kadar kalmasına gerek yoktur. Herhangi bir anda işin işlenmesi durdurularak makineye farklı bir iş yerleştirilebilir. İşlemi yarıda kesilen iş tekrar makineye konulduğunda sadece kalan süre kadar işlem görür. Bu durum, önceliği daha yüksek bir iş söz konusu olduğunda gerçekleşir.

4. Öncelik kısıtları (prec): Öncelik kısıtları tek makine veya paralel makine ortamında ortaya çıkabilir. Diğer işin işleme başlaması izin verilmeden önce bir veya birden fazla iş tamamlanmak zorundadır. Öncelik kısıtlarının çeşitli özel durumları vardır. Eğer her bir iş en çok bir tane kendinden önce ve sonra gelen işe sahipse bu kısıt zincirler olarak adlandırılır. Eğer β alanında prec yoksa işler öncelik kısıtlarına tabi değildir. Şekil 1.5'deki örnekte, B ve C işinden önce A işinin gelmesi gerekmektedir, A işi bitmeden B ve C işi başlayamaz.



Şekil 1.5. Öncelik kısıtları

5. Arızalar (brkdwn): Makine duruşları/arızaları makinelerin sürekli kullanılabilir olmadığını gösterir. Bir makinenin kullanılabilir olmayan periyodunda tamir edildiği

varsayılır. Eğer paralel özdeş birtakım makineler varsa, herhangi bir zamanda kullanılabilir makine sayısı zamanın bir fonksiyonudur.

6. Makine atama kısıtları (Mj): β alanında M_j sembolünün bulunması paralel özdeş makine bulunduğunu ifade eder. M_j bulunduğunda, her m makinesi j işini işleme kapasitesine sahip olmayabilir. M_j seti j işini işleyebilen makine setini gösterir. Eğer β alanı M_j içermezse, j işi herhangi m makinesinde işlenebilir.

7. Permütasyon (seri makineler üzerindeki iş sırasının değişmemesi, prmu): Akış tipi atölye ortamında ortaya çıkan, ilk giren ilk çıkar prensibine göre kuyruk disiplinini ifade eder.

8. Bloklanma (block): Bu durum akış tipi atölyede ortaya çıkabilir. Birbirini takip eden iki makine arasındaki kuyruk dolu ya da sınırlı bir kapasiteye sahip olduğunda, önceki makinenin işlemini bitirdiği işi sonraki makineye gönderemeyeceğini ifade eder.

9. Bekleme yok (nwt): Beklemenin olmaması akış tipi atölyelerde ortaya çıkan ayrı bir durumdur. Birbirini izleyen iki makine arasında beklemeye izin yoktur. Örneğin çelik plakalı hadde makinesinde kurumanın gerçekleşmesi engellemek için beklemeye izin verilmez. Bu kısıtlamanın olduğu modelde de kuyruk disiplini FIFO'dur.

10. Yeniden dolaşım (recrc): Atölye tipi veya esnek atölye tipinde görülür. Yeniden dolaşım, bir iş bir makineyi veya iş istasyonunu birden fazla dolaştığında ortaya çıkar.

1.2.3. Olası makine ortamında γ alanında kullanılan semboller

1. Maksimum tamamlanma süresi (C_{max}): Sistemdeki son işin bitiş zamanıdır. Bitiş zamanın minimum olması makinelerin kullanımının yüksek olduğu anlamına gelir.

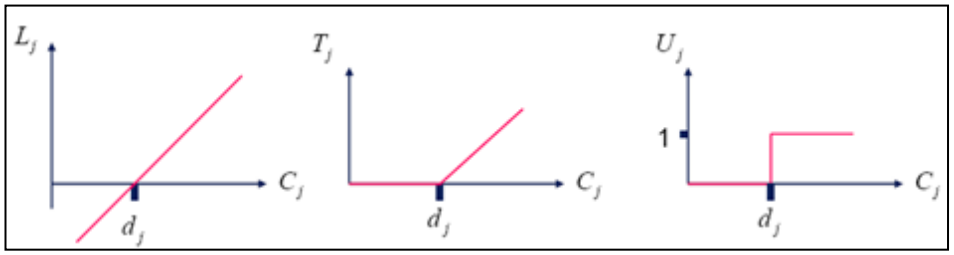
2. Maksimum pozitif geç kalma süresi (L_{max}): L_{max}, $\max(L_1,....,L_n)$ olarak tanımlanır. L_{max} ihlal edilen bitiş tarihinin en kötüsünü ölçer. j işinin tamamlanma süresi, C_j; geç kalma süresi, L_j; pozitif geç kalma süresi, T_j olarak gösterilir. Geç kalma süresi, pozitif geç kalma süresi, geç teslim edilen iş sayısı aşağıdaki Eşitlik (1.1), (1.2) ve (1.3)'deki gibidir.

$$L_j = C_j - d_j \quad (1.1)$$

$$T_j = maks\{C_j - d_j\} = maks\{L_j, 0\} \quad (1.2)$$

$$Uj = \begin{cases} 1, Cj > dj \ ise \\ 0, aksi\ halde \end{cases} \quad (1.3)$$

Şekil 1.6'da geç kalma süresi, pozitif geç kalma, teslim edilen iş sayısının grafikleri gösterilmektedir.



Şekil 1.6. Geç kalma süresi, pozitif geç kalma, teslim edilen iş sayısının grafikleri

3. Toplam ağırlıklı tamamlanma zamanı ($\sum W_j C_j$): n işin ağırlıklı tamamlanma zamanlarının toplamı, elde bulundurma veya stok maliyetlerini gösterir. Toplam tamamlanma süresi literatürde akış zamanı olarak belirtilir. Toplam ağırlıklı bitiş zamanı ağırlıklı akış zamanı olarak adlandırılır. Eşitlik (1.4)'de toplam ağırlıklı akış zamanı gösterilmiştir.

$$(\sum W_j (C_j - r_j)) = \sum W_j C_j - \sum W_j r_j \quad (1.4)$$

4. Azalan toplam ağırlıklı tamamlanma zamanı: $(\sum W_j (1 - e^{-rC_j}))$ ifadesi, daha öncekinden daha genel maliyet fonksiyonudur. Maliyetler, her birim zamanda r hızında azalmaktadır ($0<r<1$). Yani, eğer j işi t zamanda tamamlanmazsa, bu dönemde $[t, t+d_t]$ ek maliyet $w_j r e^{-rt} dt$ meydana gelecektir. Eğer j işi t zamanda tamamlanırsa, toplam maliyet $w_j (1 - e^{-rt})$'dir. r nin değeri genelikle sıfıra yakın 0,1 ya da %10 dur.

5. Toplam ağırlıklı pozitif gecikme: $(\sum W_j T_j)$ ifadesi genel maliyet fonksiyonudur.

6. Geç teslim edilmiş iş sayısının ağırlığı $(\sum W_j U_j)$: Geç teslim edilmiş işlerin sayısının ölçümü literatürde yaygın kullanım alanı bulamamaktadır, ancak ölçümü basit olmasından dolayı pratikte bir hedef olarak belirlenebilir.

Tablo 1.4'de amaç fonksiyonları özetlenmiştir (Qiana ve diğ., 2009).

Tablo 1.4. Amaç fonksiyonları ve formülleri

1	Maksimum tamamlanma süresi	$C_{max}(\pi) = max_j C_j(\pi)$
2	Ortalama(ağırlıklı) tamamlanma süresi	$\overline{C}(\pi) = \frac{1}{n}\sum_j^n \lambda_j C_j(\pi)$
3	Maksimum gecikme	$T_{max}(\pi) = max_j\, T_j(\pi)$
4	Ortalama (ağırlıklı) gecikme	$\overline{T}(\pi) = \frac{1}{n}\sum_{j=1}^n \lambda_j T_j(\pi)$
5	Toplam makine boş zamanı	$I_{sum}(\pi) = \sum_{i=1}^m I_i(\pi)$
6	Maksimum erken tamamlanma	$E_{max}(\pi) = max_j\, E_j(\pi)$
7	Ortalama (ağırlıklı) erken tamamlanma	$\overline{E}(\pi) = \frac{1}{n}\sum_{j=1}^n \lambda_j E_j(\pi)$
8	Maksimum akış zamanı	$F_{max}(\pi) = max_j(C_j(\pi) - S_{1,\pi(j)})$
9	Ortalama (ağırlıklı) akış zamanı	$\overline{F}(\pi) = \frac{1}{n}\sum_{j=1}^n \lambda_j(C_j(\pi) - S_{1,\pi(j)})$
10	Geciken İş sayısı	$N_T(\pi) = \sum_{j=1}^m U_j(\pi)$

1.2.4. Bazı problemlerin tanımsal gösterimi (α | β | γ)

$FF_c \mid r_j \mid \Sigma\, w_j T_j$: İşlerin geliş zamanı ve teslim tarihlerinin tanımlı olduğu esnek akış tipi çizelgeleme ortamını belirtmektedir. Amaç fonksiyonu, toplam ağırlıklı pozitif gecikmenin mininizasyonudur.

$1 \mid s_{jk} \mid C_{max}$: Sıra bağımlı hazırlık sürelerinin tanımlı olduğu tek makineli çizelgeleme ortamını belirtmektedir. Amaç fonksiyonu tamamlanma zamanının minimizasyonudur.

$P_m \mid r_j, Mj \mid \Sigma\, w_j T_j$: İşlerin geliş zamanı ve teslim zamanlarının tanımlı olduğu paralel m makine çizelgeleme ortamında her işin yalnızca belli bir makine grubunda

işlenebilir olduğunu belirtmektedir. Amaç fonksiyonu toplam ağırlıklı pozitif gecikme zamanının minimizasyonudur.

1.2.5. Temel önceliklendirme kuralları

Bir makine boş kaldığında, önünde bekleyen işler arasından önceliği en yüksek olan işi seçmek için çeşitli kurallar uygulanmaktadır. Bunlar;

1. Statik kurallar: İş ve/veya makine verisine bağlı, zamandan bağımsızdır.
2. Dinamik kurallar: Zamana bağlıdır.
3. Rassal seçim (SIRO): Herhangi bir öncelik kuralı yoktur, rassal seçim vardır.
4. İlk gelen ilk hizmet alır: Bekleme sürelerinin değişkenliğini azaltır.
5. Teslim zamanı en erken olan işin en önce işlenmesi (EDD): En büyük geç kalmayı en küçükler.
6. En düşük serbestliği olan işin en erken işlenmesi (MS)
7. En kısa işlem süreli işin en erken işlenmesi (SPT): Toplam akış süresini en küçükler.
8. En uzun işlem süreli işin en erken işlenmesi (LPT): Paralel makinalarda yük dengeleme söz konusu olduğu durumlarda kullanılır.
9. En kısa hazırlık süreli işin en erken işlenmesi (SST)
10. En az esnek işin en erken işlenmesi: İşlerin paralel makinalarda işleme kısıtı bulunması halinde kullanılır.
11. Kritik yol kuralı: Öncelik kısıtları varsa kullanılır.
12. En fazla artçılı bulunan işin en erken işlenmesi

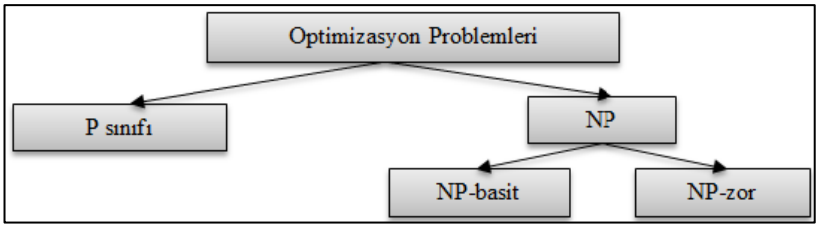
1.3. Algoritmanın Karmaşıklığı

Algoritmanın karmaşıklığını üçe ayırmak mümkündür. Bunlar;

1. Polinom zamanla çözülebilen P sınıfı problemler: Bu problemlerin polinom zamanda optimal olarak çözülmesi mümkündür.
2. NP-basit problemler: Polinom zamanda çözüm algoritması bulunamayan problemler vardır. Bu tür problemlerin cevabı tahmin edilebiliyorsa tahminin doğruluğunu sınamak için veri büyüklüğüne polinom mertebesinde bağımlı sürelerde çalışabilecek algoritmalar bulunmaktadır.

3. NP zor problemler: NP içerisinde P'ye asla taşınamayacak problemler NP zor problem sınıfında yer almaktadır. Çizelgeleme problemi de , bu sınıf içerisindedir ve NP zor olarak adlandırılır (Çetin, 2007).

Şekil 1.7'de optimizasyon problemlerinin zorluk açısından sınıflandırılması gösterilmiştir.



Şekil 1.7. Algoritmanın karmaşıklığı

1.4. Akış Tipi Çizelgeleme Problemleri için Yapılan çalışmalar

Akış tipi çizelgelemede hem toplam akış zamanı hem de tamamlanma zamanını minimize etmenin NP zor sınıfında bir problem olduğu Garey ve diğ. (1976) tarafından kanıtlamıştır. Bu yüzden optimum sonucu bulmaya çalışmak yerine sezgisel ve metasezgisel yöntemlerle makul sürelerde, yüksek kalitede çözümlere odaklanılmıştır. Tamamlanma süresini minimize etmek için Palmer (1965), Campbell ve diğ. (1970), Dannenbring (1977), Nawaz ve diğ. (1983) tarafından sezgisel metotlar önerilmiştir. Bu metotlar yapıcı metotlardır. Bu metotların içinde en iyi model Nawaz ve diğ. (1983) tarafından önerilen NEH algoritması olmasına rağmen, bu algoritmadan elde edilen ortalama sapma %5 in üstündedir. Son yıllarda akış tipi çizelgeleme probleminde tamamlanma süresinin minimizasyonuna yönelik çeşitli meta sezgisel metotlar geliştirilmiştir. Bunlar tavlama benzetimi (Ogub ve Simith, 1990; Ishubuchi ve diğ., 1995), tabu arama (Taillard, 1990; Nowicki ve Smutnicki, 1996; Grabowski ve Wodecki, 2004), genetik algoritma (Reeves, 1995) ve hibrit metasezgisellerdir (Reeves ve Yamada, 1998; Wang ve Zheng, 2003). Metasezgisel metotlar, sezgisel metotlardan genel olarak daha iyi çözüm kalitesi sunmaktadır ve kesin metotlara göre çok daha kısa zamanda sonuç vermektedir. Sezgisel metotlar üç gruba ayrılmaktadır: Yapıcı sezgiseller (Wang ve diğ., 1997), geliştirici sezgiseller (Ho, 1995), ve metasezgiseller. Rajendran ve Ziegler (1997), her bir işi sırasıyla aralara yerleştirerek yapıcı sezgisel algoritmayı iyileştirmeye çalışmıştır. Woo ve Yim (1998) önerdikleri yapıcı sezgisel algoritmayı Rajendran ve

Ziegler (1997) ve NEH algoritmasıyla karşılaştırmışlar ve önerdikleri algoritmanın bu iki algoritmaya göre daha iyi sonuçlar verdiğini ortaya koymuşlardır.

Liu ve Reeves (2001) lokal arama metodlarını da içeren çok sayıda yapıcı sezgiseli birleştiren bir sezgisel önermiştir. Elde ettikleri bu yapıcı sezgiselin daha iyi sonuç verdiğini belirtmişlerdir. Allahverdi ve Aldowaisan (2002) küçük modifikasyonlarla (ikili yer değiştirmeler ile) performansın artırılabileceğini ortaya koymuştur. Framinan ve diğ. (2005) toplam akış tipi çizelgeleme problemleri için metasezgisel yöntemlere genel bir bakış sunmuştur. Ek olarak diğer sezgisellere göre daha üstün metasezgisel önermişlerdir. Ele aldığı toplam akış tipi çizelgeleme problemi için metasezgisellerin yapıcı ve geliştirici sezgisellere göre daha iyi sonuç verdiğini göstermiştir. Genel olarak metasezgiseller sezgisellerden daha iyi sonuç verdiğini ancak daha fazla hesap zamanları söz konusu olduğunu belirtmişlerdir.

Reeves ve Yamada (1998) çalışmasında lokal arama ile genetik algoritmayı birleştirerek toplam akış süresini minimize etmektedir. Kullandıkları algoritma iyi sonuçlar vermektedir ancak hesaplama zamanı uzundur. Rajendran ve Ziegler (2004) toplam akış süresini ve tamamlanma süresini azaltmak için M-MMAS and PACO olmak üzere iki karınca koloni optimizasyon metodu önermiştir. Metodlarının performansını ölçmek için, Liu ve Reeves (2001) algoritması ile karşılaştırmışlar ve Taillard'ın kıyaslama problemlerinde daha iyi sonuçlar elde etmişlerdir. Rajendran ve Ziegler (2005) iki karınca kolonisi algoritmasını geliştirerek daha kaliteli çözümler elde etmişlerdir. Tasgetiren ve diğ. (2007) parçaçık sürü optimizasyon modeli önermiştir ve bu metod ile Liu ve Reeves (2001), ve Rajendran ve Ziegler (2005) tarafından geliştirilen algoritmalardan çok daha iyi sonuçlar elde etmişlerdir.

2. VERİ MADENCİLİĞİ TABANLI OPTİMİZASYON ÇALIŞMALARI

Veri madenciliği tekniklerini işlevlerine göre üç temel grupta toplanır: Bunlar; sınıflama, kümeleme ve birliktelik kuralları ve sıralı örüntülerdir.

Sınıflama: Sınıflama, verilerin önceden belirlenen çıktılara uygun olarak ayrılmasını sağlamaktadır. Çıktılar, önceden bilindiği için sınıflama, veri kümesini denetimli olarak öğrenir. Örneğin; finans alanında çalışan A şirketi; müşterisinin yeni bir yatırım yapma konusunda değerlendirme yapmaktadır. Finans şirketi daha önceden benzer bir ürün satmıştır ve geçmiş verilere dayanarak hangi müşterilerin önceki teklife cevap verdiğini bilmektedir. Müşterilerin özellikleri belirlenerek, satış faaliyetlerini daha etkin bir şekilde gerçekleştirmek amaçlanmaktadır. Müşterinin bu teklife cevabı olarak evet / hayır şeklinde bir alan bulunmaktadır. Bu alana hedef yada bağımlı değişken denilmektedir. Müşterilerin çeşitli özelliklerinin (yaş, cinsiyet,....) hedef değişkeni üzerinde etkisini belirlemek amaçtır. Bu değişken bağımsız ya da tahminci değişkendir. Sınıflamada kullanılan çok çeşitli yöntemler bulunmaktadır. Bunlardan bazıları karar Ağaçları, Yapay Sinir Ağları, Genetik Algoritmalar, K-En Yakın Komsu, Bellek Tabanlı Yöntemler, Regresyondur (Akpınar, 2000).

Kümeleme: Verilerdeki benzerliklere göre verilerin gruplandırılmasını sağlayan bir tekniktir. Kümeleme tekniği çoğunlukla başka uygulamalar için bir ilk işlem olarak kullanılır (Tantuğ, 2002). Çoğunlukla K-ortalamalar algoritması ya da Kohonen şebekesi gibi istatistiksel yöntemler kullanılmaktadır. Bu yöntemlerde süreçler aynıdır. İlk olarak her kayıt var olan kümelerle karşılaştırılır ve her kayıt kendisine en yakın kümeye atanır ve bu kümeyi tanımlayan değerleri değiştirir. Her defasında kayıtlar yeniden atanır ve küme merkezleri ayarlanır (Hui ve Jha, 2000).

Birliktelik kuralları ve ardışık zamanlı örüntüler: Burada diğerlerinden birbirini ayıran özelliğin zaman kavramın uygulamada olmasıdır. Belli bir dönem boyunca nesneler arasındaki birlikteliklerin incelenmesi ardışık zamanlı örüntü olarak adlandırılır (Goebel ve Gruenwald, 1999).

Birliktelik kuralları çok çeşitli alanlarda uygulanmaktadır. Birliktelik kuralları aynı işlem içinde çoğunlukla beraber görülen nesneleri içeren kurallardır. Birliktelik kuralları pazar sepeti çözümlemesinde kullanılmaktadır. Bu çözümlemede, nesneler müşteriler tarafından satın alınan ürünlerdir ve bir işlem (kayıt) ise birçok nesneyi içinde bulunduran tek bir satın almadır. Pazar sepeti çözümlemesinde sıklıkla beraber alınan nesneler üzerine çalışılır. Bulunan kurallar ile nesnelerin birbiri ile nasıl ilişkili olduğu bilgisine ulaşılır (Gönülol, 2009).

Önerilen modelde kullanılan veri madenciliği tekniği sınıflama grubunda değerlendirilebilir. Burada örneğin 20 işin çizelgeleneceği bir problemde 20 farklı pozisyonu temsil eden 20 sınıf bulunacaktır. Bu pozisyonlar hedef yada bağımlı değişken olurken, pozisyonlara atanacak işler ise bağımsız değişken olmaktadır.

Literatürde veri madenciliğini çizelgeleme alanında kullanan optimizasyon çalışmalarını kısaca özetlemek gerekirse:

Koonce ve diğ. (1997) atölye tipi iş sıralama çizelgeleme problemi üzerinde çalışmışlardır. Bu çalışmada, genetik algoritmayla bulunan optimum iş sıralarından işlem, süreç ve yük göz önüne alınarak kurallar belirlenmiştir. Sonuç ve öneriler kısımında veri madenciliği yönteminin karmaşık sistemleri açıklamada yardımcı olarak kullanılacağına değinilmiştir.

Koonce ve Tsai (2000) çalışmalarında çok sayıda atölye tipi iş sıralama probleminden bilgi çıkartımı için veri madenciliği algoritmasını kullanmıştır. Bu çalışmada, atölye tipi iş sıralama problemleri olarak Muth ve Thompson (1963) problemleri kullanılmıştır. Tablo 2.1'de bulunan ilk rakam makineyi, ikinci rakam süreyi göstermektedir. Örneğin birinci işin birinci makinede ilk operasyon olarak yapılması durumunda işlem süresi üçtür. Aynı şekilde birinci işin ilk makinede ikinci operasyon olarak işlem gördüğündeki süresi 1'dir. Gösterimi (1,3) şeklindedir.

Tablo 2.1. 6 iş 6 makine atölye tipi iş sıralama problemi

İş	**Operasyon**					
	1	2	3	4	5	6
1	3,1	1,3	2,6	4,7	6,3	5,6
2	2,8	3,5	5,10	6,10	1,10	4,4
3	3,5	4,4	6,8	1,9	2,1	5,7
4	2,5	1,5	3,5	4,3	5,8	6,9
5	3,9	2,3	5,5	6,4	1,3	4,1
6	2,3	4,3	6,9	1,10	5,4	3,1

Genetik algoritmalar doğal seçim ve genetiğin mekanizmalarını temel alan stokastik arama algoritmalarıdır. Şekil 2.1'de görüldüğü üzere 36 lık bir dizi şeklinde genetik algoritma kullanılarak kodlanmıştır. Her bir eleman bir işi karşılamaktadır ve algoritma 1000 kere çalıştırılmıştır. Populasyon büyüklüğü 500'dür ve başlangıç populasyonu rassal olarak oluşturulmuştur. Mutasyon oranı ise %20 olarak belirlenmiştir.

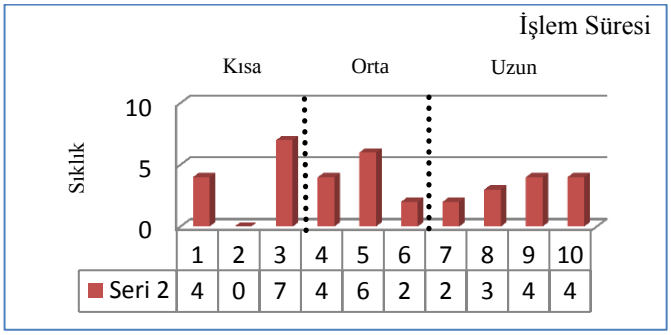
1	3	4	2	6	1	5	4	6	3	4	1	3	6	2	5	5	4	2	4	3	6	2	3	1	6	
1	1	1	1	1	2	1	2	2	2	3	3	3	3	2	2	3	4	3	5	4	4	4	5	4	5	...

Şekil 2.1. Atölye tipi iş sıralama problemi için elde edilen optimum diziliş

Şekil 2.1'de ilk satırdakiler işleri, ikinci satırdakiler operasyonları göstermektedir. İlk işin ilk operasyonda işlendiğini daha sonra sırasıyla 3. ,4. ,2. ve 6. işlerin takip ettiği, daha sonra ikinci operasyonda birinci işin başladığı görülmektedir. Bu akış ikinci işin altıncı operasyonda işlem görmesiyle sonlanmaktadır. Bu çalışma, genetik algoritma tarafından üretilen verilerdeki desenleri bulmak ve kurallar seti geliştirmek için veri madenciliği yöntemine başvurmaktadır. Elde edilen bu optimal ya da optimala yakın çözümler arasında işlemler ve sıralı durumların karakteristikleri arasında benzer ilişkiler bulunabilir. Bu ilişkiler içersinden tanımlanan kurallar benzer problemler üzerinde genetik algoritmanın performansını artırabilir ve benzer problemler için basit öncelik kuralları ile daha çok üstünlük sağlayabilir. Çalışmada kuralların ortaya çıkması için aşağıdaki faktörler üzerinde durulmuştur.

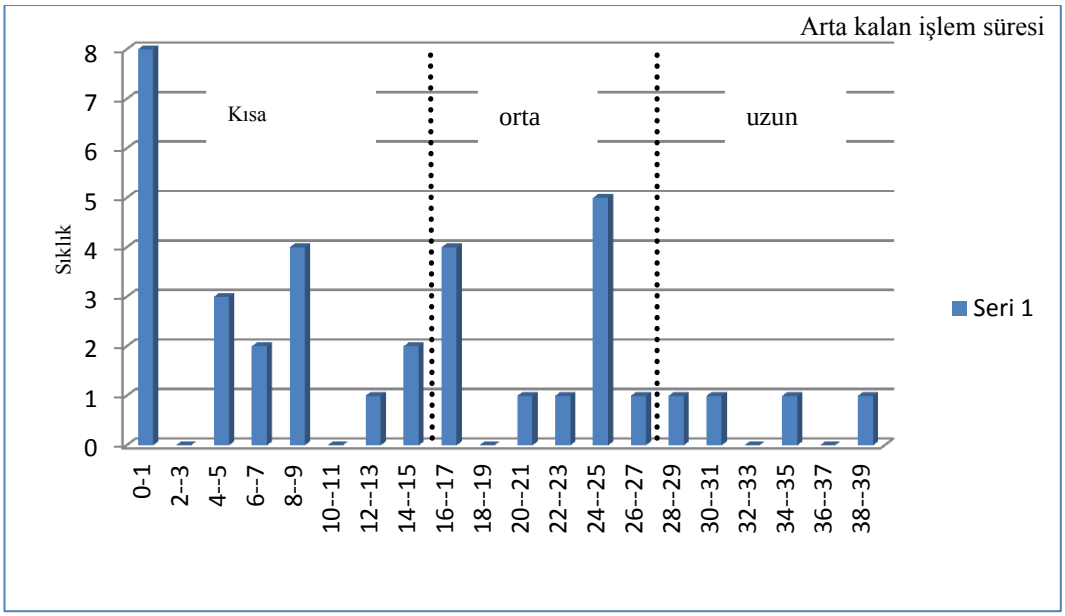
İşlem (Operasyon): İşin sırasını göstermektedir ve altı sınıfa ayrılmıştır. Operasyon 1 ilk, 2. ve 3. orta; 4. ve 5. operasyonlar son ve altıncı operasyon en son operasyon olarak sınıflandırılmıştır.

İşlem zamanı ve arta kalan işlem zamanı: İşlem zamanı, söz konusu işin işleme zamanını ifade eder. Şekil 2.2'de işlerin işlem zamanları ve sıklıkları görülmektedir. Örneğin işlem süresi 1 ile 3 arasındakiler kısa; 4 ile 6 arasındakiler orta; 7 ile 10 arasındakiler uzun işlem süreli işler olmak üzere üçe ayrılmıştır.



Şekil 2.2. İşlem zamanı ve sıklıkları

Şekil 2.3'de görüldüğü üzere arta kalan işlem süresi kısa, orta ve uzun olmak üzere üçe ayrılmıştır.



Şekil 2.3. Arta kalan işlem süresinin sıklığı ve sınıfları

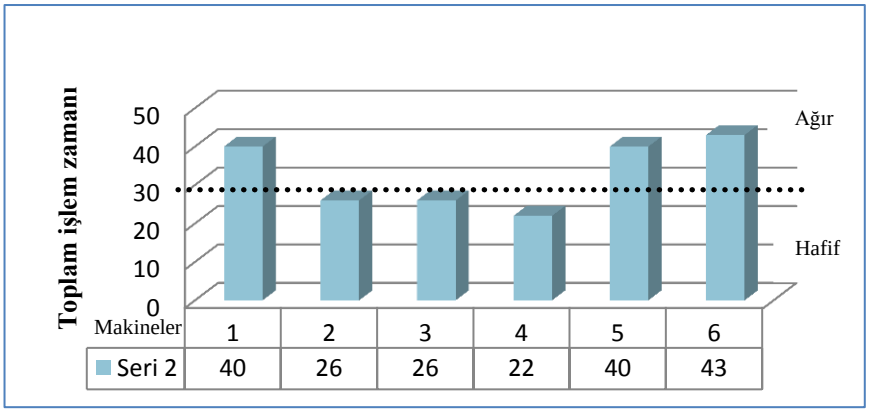
Tablo 2.2'de, her bir iş için her operasyonda gerçekleşen toplam işlem süreleri hesaplanmış ve bu toplam iş sürelerinden operasyon sırasıyla işlem süreleri

çıkarılarak arta kalan işlem süreleri belirlenmiştir. Kısaca, arta kalan işlem zamanı kümülâtif işlem süresini gösterir.

Tablo 2.2. Arta kalan işlem süreleri

İş	**Operasyon**	**Makine**	**İşlem süresi**	**Arta kalan işlem süresi**	
1	1	3	1	26-1=25	25
1	2	1	3	25-3=22	22
1	3	2	6	22-6=16	16
1	4	4	7	16-7=9	9
1	5	6	3	9-3=6	6
1	6	5	6	6-6=0	0
Toplam			**26**		
2	1	2	8	47-8=39	39
2	2	3	5	39-5=34	34
2	3	5	10	34-10=24	24
2	4	6	10	24-10=14	14
2	5	1	10	14-10=4	4
2	6	4	4	4-4=0	0
Toplam			**47**		
	…	…	…	…	
	…	…	…	…	
Toplam				…	
6	1	2	3	30-3=27	27
6	2	4	3	27-3=24	24
6	3	6	9	24-9=15	15
6	4	1	10	15-10=5	5
6	5	5	4	5-4=1	1
6	6	3	1	1-1=0	0
Toplam			**30**		

Makine yükleri: Her bir makinedeki toplam işleme zamanı hesaplanmıştır. Toplam işleme zamanı ilk makine için 40, ikinci ve üçüncü makine için 26, dördüncü makine için 22, beşinci makine için 40 ve altıncı makine için 43 olarak hesaplanmıştır. Makineler toplam işlem zamanına göre iki sınıfa ayrılmıştır. Toplam işlem zamanı 30'un üzerinde olan makineler yükü ağır sınıfına ve toplam işlem zamanı 30'un altında olan makineler ise yükü hafif sınıfına dahil edilmiştir. Şekil 2.4'de makine yükü ağır ve hafif olan makineler gösterilmektedir.



Şekil 2.4. Arta kalan işlem süresinin sıklığı ve sınıfları

Tablo 2.3'de görüldüğü üzere, 6x6 atölye tipi sıralama problemi için, 264 optimal çözüm sırasından 24 tane farklı kural üretilmiştir.

Tablo 2.3. Altı makine altı işli atölye tipi iş sıralama problemi için kurallar

Kural No	Operasyon	Proses	Makine Yüklemesi	Arta Kalan Süre	0	1	2	3	4	5
1	İlk	Uzun	Hafif	Uzun	1					
2	İlk	Uzun	Hafif	Orta	0.0379	0.7841	0.1780			
3	İlk	Orta	Hafif	Uzun	0.8769	0.1231				
4	İlk	Kısa	Hafif	Uzun	0.4432	0.5530	0.0038			
5	İlk	Kısa	Hafif	Orta	0.9773	0.0227				
6	Orta	Uzun	Ağır	Orta	0.0429	0.3902	0.5101	0.0568		
7	Orta	Orta	Ağır	Orta	0.0303	0.6629	0.3068			
8	Orta	Orta	Ağır	Kısa		0.0038	0.3220	0.6591	0.0152	
9	Orta	Orta	Hafif	Uzun	0.5114	0.4848	0.0038			
10	Orta	Orta	Hafif	Orta	0.2235	0.2285	0.3914	0.1503	0.0063	
11	Orta	Kısa	Ağır	Orta	0.4242	0.5227	0.0492	0.0038		
12	Orta	Kısa	Hafif	Orta	0.0227	0.6250	0.3523			
13	Orta	Kısa	Hafif	Kısa		0.1856	0.7311	0.0833		
14	Son	Uzun	Ağır	Orta			0.1818	0.7727	0.0455	
15	Son	Uzun	Ağır	Kısa		0.0133	0.1998	0.4119	0.3589	0.0161
16	Son	Uzun	Hafif	Kısa			0.0227	0.5189	0.4545	0.0038
17	Son	Orta	Ağır	Kısa				0.1458	0.6932	0.1610
18	Son	Kısa	Ağır	Kısa				0.0246	0.5663	0.4091
19	Son	Kısa	Hafif	Orta			0.2538	0.7083	0.0379	
20	Son	Kısa	Hafif	Kısa			0.0909	0.5720	0.3182	0.0189
21	Enson	Uzun	Ağır	Kısa				0.0341	0.3277	0.6383
22	Enson	Orta	Ağır	Kısa					0.0492	0.9508
23	Enson	Orta	Hafif	Kısa				0.0038	0.3371	0.6591
24	Enson	Kısa	Hafif	Kısa					0.0568	0.9432

Örneğin kural 1 için; eğer (operasyon =ilk) ∧ (proses(x)=uzun) ∧ (makine(x)= hafif) ∧ (arta kalan süre(x)=uzun) ise öncelik (x)=0, ağırlık=1. Elde edilen kurallara göre sonuçların en kısa işlem süresi sezgiselinden (SPT) daha iyi sonuçlar verdiği görülmüştür. SPT sezgiseli %20 hata ile, veri madenciliği tabanlı model ise %14 hata ile çalışmıştır. Ancak öğrenilen kurallar genetik algoritmanın performansıyla karşılaştırıldığında iyi sonuçlar vermemektedir.

Santos ve diğ. (2006) çalışmalarında araç rotalama problemi gibi bilinen bir problem üzerinde, algoritmadaki performansı artırmak için bazı alternatifler sunmuşlardır. Genetik algoritma (GA), lokal arama ile GA, veri madenciliği modulunu içeren GA ve lokal arama ve veri madenciliği içeren GA olmak üzere dört algoritma geliştirilmiştir. Veri madenciliği ve lokal arama çözüm kalitesini artırmak amacıyla kullanılmıştır. Elit çözüm kümelerinde sık tekrar eden yakın çözümleri bulmak amacıyla veri madenciliğinde apriori algoritması uygulanmıştır. Deneysel sonuçlar, veri madenciliği ve lokal arama modüllü genetik algoritmanın daha iyi sonuçlar verdiğini göstermiştir.

Ceran (2006) çalışmasında, maksimum tamamlanma zamanının minimizasyonuna yönelik esnek akış tipi çizelgeleme problemlerinin çözümü için genetik algoritma ve bu çözümler arasındaki ilişkileri ortaya çıkarmak için veri madenciligini kullanmıştır. Veri madenciliginde gerekli bilgi için genetik algoritmalar uygulanarak elde edilen iş sıraları ve C_{max} değerleri kullanılmıştır. İşler üç bölüme ayrılmıştır. İşlerin zorluk derecesine göre öncelik vermek amacıyla her bölümdeki işlerin ortalama süreleri hesaplanmıştır. Hesaplanan ortalamalar karşılaştırılıp, en küçük olana 1, sonrakine 2 ve en büyügüne 3 değeri atanmıştır. Ayrıca elde edilen kromozom dizilişleri dikkate alınarak C_{max} değerleri en küçük olana 1, %3 lük fazlalığı olanlara 2, %5 fazlalığı olanlara 3 değeri verilmiştir. Analiz sonucunda her tip problem türüne uygun, genetik algoritmadan elde edilen çözümlere göre çeşitli kurallar elde edilmiştir. Bu kurallardan C_{max}'ı minimize eden ve en çok tekrarlanan kurala göre, sıralamanın en kısa ortalamalı iş grubundan uzun süreliye mi yoksa uzun ortalamalı iş grubundan en kısa süreliye doğru mu olduğu konusunda çıkarımlar yapılmıştır. Önceden bilinmeyen ancak potansiyel olarak kullanışlı olan bu kurallardan elde edilecek sonuçların başka tekniklerle hibritlenerek iyi çözümler elde edilebilmesinin mümkün olduğu ifade edilmiştir.

Yang ve diğ. (2009) hastanelerin acil bölümündeki işgücü planlamasına yönelik bir çalışma yapmışlardır. Veri madenciliğinde kullanılan karar ağacını günlere göre (gece-gündüz-tatil olup olmadığı da göz önünde bulundurularak) işgücü tahmininde kullanmışlardır. Model kurulduktan ve veriler analiz edildikten sonra, işgücü sayısı belirlenmiş ve zaman dilimlerine göre işgücü planlanmıştır. Tablo 2.4'de veri madenciliği algoritmasında kullanılan karar ağaçları ile elde edilen kurallara göre işgücü düzenlemesi görülmektedir.

Tablo 2.4. Tahminlenen kurallara göre işgücü düzenlenmesi

NO	KURAL	İŞGÜCÜ
1	Eğer Hafta=1 ve Vardiya= Gündüz Vardiyası ise	7 Kişi
2	Eğer Hafta=2 ve Vardiya= Gündüz Vardiyası ise	6 Kişi
3	Eğer Hafta=3 ve Vardiya= Gündüz Vardiyası ise	6 Kişi
4	Eğer Hafta=4 ve Vardiya= Gündüz Vardiyası ise	6 Kişi
5	Eğer Hafta=5 ve Vardiya= Gündüz Vardiyası ise	7 Kişi
6	Eğer Hafta=6 ve Vardiya= Gündüz Vardiyası ise	7 Kişi
7	Eğer Hafta=7 ve Vardiya= Gündüz Vardiyası ise	5 Kişi
8	Eğer Hafta=1 ve Vardiya= Gece Vardiyası ise	5 Kişi
9	Eğer Hafta=2 ve Vardiya= Gece Vardiyası ise	5 Kişi
10	Eğer Hafta=3 ve Vardiya= Gece Vardiyası ise	5 Kişi
11	Eğer Hafta=4 ve Vardiya= Gece Vardiyası ise	5 Kişi
12	Eğer Hafta=5 ve Vardiya= Gece Vardiyası ise	5 Kişi
13	Eğer Hafta=6 ve Vardiya= Gece Vardiyası ise	5 Kişi
14	Eğer Hafta=7 ve Vardiya= Gece Vardiyası ise	6 Kişi
15	Eğer Kural 14 ve Ulusal tatil varsa	Birden Fazla

Kumar ve Rao (2009) karınca kolonisi algoritması ve yeni çözümler oluşturmak amacıyla genetik algoritma operatörleri (çaprazlama ve mutasyon) kullanarak bloklu akış tipi çizelgeleme problemine çözüm getirmişlerdir. Sekiz iş dört makine akış tipi çizelgeleme problemini ele almışlardır. Tablo 2.5'deki verileri kullanarak karınca kolonisi algoritmasını uygulamışlardır. Her bir pozisyon için feramon miktarları hesaplanarak optimal sıra elde edilmiştir.

Tablo 2.5. Sekiz iş dört makine problemi için veriler

	M1	**M2**	**M3**	**M4**
J1	10	20	5	30
J2	15	8	12	10
J3	20	7	9	5
J4	13	7	17	10
J5	8	3	16	7
J6	6	9	22	7
J7	7	5	15	12
J8	14	13	6	4

Tablo 2.6'da her pozisyon için, karınca kolonisi algoritması ile elde edilen feromon miktarları gösterilmektedir.

Tablo 2.6. Sekiz iş dört makine problemi için feromon miktarları

	Her pozisyon için feromon miktarları							
iş	P1	P2	P3	P4	P5	P6	P7	P8
1	0.00566385	0.0337077	**0.171359**	0.00515377	0.0113062	0.00580029	0.00832053	0.0051537
2	0.00582081	0.00588679	0.00572423	0.00823283	0.00566385	0.0404619	**0.16911**	0.0055642
3	0.00691272	0.00768597	0.0143749	0.0401588	**0.354684**	0.00734112	0.00588679	0.128636
4	0.00515377	**0.126699**	0.00515377	0.00515377	0.0058497	0.0516028	0.0416981	0.0051537
5	**0.199369**	0.00515377	0.00515377	0.00549667	0.00515377	0.00778495	0.00515377	0.013199
6	0.00515377	0.016125	0.0337077	**0.169428**	0.00618129	0.00515377	0.00515377	0.0055616
7	0.0128565	0.0460527	0.00553477	0.00768729	0.0410516	**0.121881**	0.00515377	0.0062475
8	0.00553477	0.00515377	0.00545694	0.00515377	0.0013579	0.00643934	0.00598799	**0.076948**

Tablo 2.6'da görülen P1, P2, P3, P4, P5, P6, P7, P8 değerleri her pozisyon için feromon miktarını göstermektedir. Yüksek feromon miktarları iş sıralarını belirlemekte kullanılır. İlk pozisyona 5.iş, ikinci pozisyona 4.iş, üçüncü pozisyona ilk iş, dördüncü pozisyona 6.iş, beşinci pozisyona 3. iş, altıncı pozisyona 7. iş, yedinci pozisyona ikinci iş, son konuma ise sekizinci işin atanması gerekmektedir. Elde edilen optimal iş sırası 5,4,1,6,3,7,2,8 olarak belirlenmiştir. Yeni optimal yada optimuma yakın iş sıraları elde etmek amacı ile genetik algoritmadaki çaprazlama

operatörleri kullanılmıştır. Tablo 2.7'de elde edilen bu optimal iş sırasına alternatif sıralar gösterilmiştir.

Tablo 2.7. Optimal farklı iş sıraları

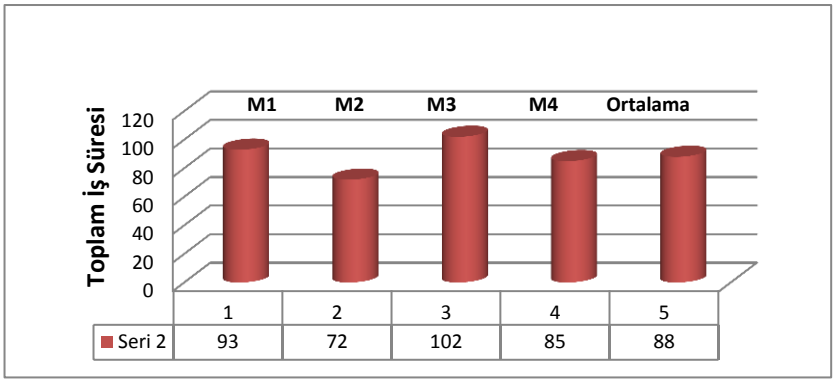
No	**Optimal İş Sıraları**							
1	5	4	1	6	3	7	2	8
2	5	2	4	1	6	3	7	8
3	7	1	5	6	4	2	8	3
4	7	5	6	4	2	8	1	3

Elde edilen optimum sonuçlardan Koonce ve diğ. (1997) ve Koonce ve Tsai (2000)'nın çalışmalarında olduğu gibi aynı parametrelere göre (makine yüklemesi, işlem zamanı, arta kalan işlem zamanı) kurallar türetilmiştir.

Tablo 2.8. Her makine için toplam iş yükü

	M1	**M2**	**M3**	**M4**
J1	10	20	5	30
J2	15	8	12	10
J3	20	7	9	5
J4	13	7	17	10
J5	8	3	16	7
J6	6	9	22	7
J7	7	5	15	12
J8	14	13	6	4
Toplam	**93**	**72**	**102**	**85**

Tablo 2.8'de görüldüğü üzere makine yüklemesi için her bir makinede işlem gören tüm işlerin süreleri toplanarak her bir makineye düşen yük bulunmuştur. Makine 1 için toplam iş süresi 93, makine 2 için 72, makine 3 için 102, makine 4 için 85 olarak hesaplanmıştır. Tüm makinelerin yük ortalaması ise 88 olarak hesaplanmıştır. Bu ortalamanın altında kalan işlem süreleri hafif, üstünde kalan işlem süreleri ise ağır olarak iki sınıfa ayrılmıştır. Şekil 2.5'de her makine için elde edilen yükler gösterilmiştir.



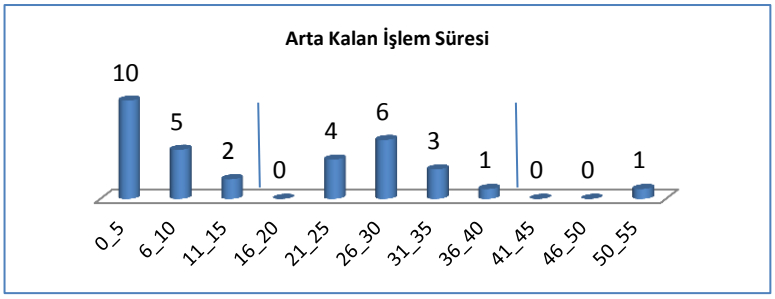
Şekil 2.5. Ortalama makine yükü ve her makine için toplam yük

Tablo 2.9'da görüldüğü üzere her bir makinedeki toplam iş süresinden, sırasıyla her bir operasyon çıkarılarak arta kalan işlem süresi hesaplanmıştır.

Tablo 2.9. Arta kalan işlem süreleri

İş	Operasyon	Makine	İşlem süresi	Arta Kalan İşlem süresi
1	1	1	10	65-10=55
1	2	2	20	55-20=35
1	3	3	5	35-5=30
1	4	4	30	30-30=0
Toplam				65
2	1	1	15	45-15=30
2	2	2	8	30-8=22
2	3	3	12	22-12=10
2	4	4	10	10-10=0
Toplam				45
	…	…	…	…
	…	…	…	…
Toplam				…
8	1	1	19	42-19=23
8	2	2	13	23-13=10
8	3	3	6	10-6=4
8	4	4	4	4-4=0
Toplam				42

Şekil 2.6'da arta kalan işlem sürelerinin sıklıkları gösterilmektedir. Arta kalan işlem süresi 0 ile 15 arasındakiler kısa; 16 ile 40 arasındakiler orta; 41 ile 55 arasındakilerde uzun olmak üzere üç sınıfa ayrılmıştır.



Şekil 2.6. Arta kalan işlem sürelerin sıklıkları

İşlerin işlem süreleri 4 ile 30 arasında değişmektedir. İşlem süresi 0-8 arası kısa; 9-18 arası orta; 19-30 arası uzun olmak üzere üç sınıfa ayrılmıştır. Birinci operasyon/makine için ilk, ikinci ve üçüncü operasyon /makine için orta, dördüncü operasyon için son olmak üzere üç sınıfa ayrılmıştır. Tablo 2.10'da işlem süresi ve operasyon sınıfları gösterilmiştir.

Tablo 2.10. İşlem süresi ve operasyon sınıfları

İşlem süresi		**Operasyon/makine**	
0-8	Kısa	1	İlk
9-18	Orta	2-3	Orta
19-30	Uzun	4	Son

Şekil 2.7'de gösterildiği üzere, belirlenen tüm kriterler sınıflara ayrıldıktan sonra see5 yazılımıyla kurallar belirlenmiştir.

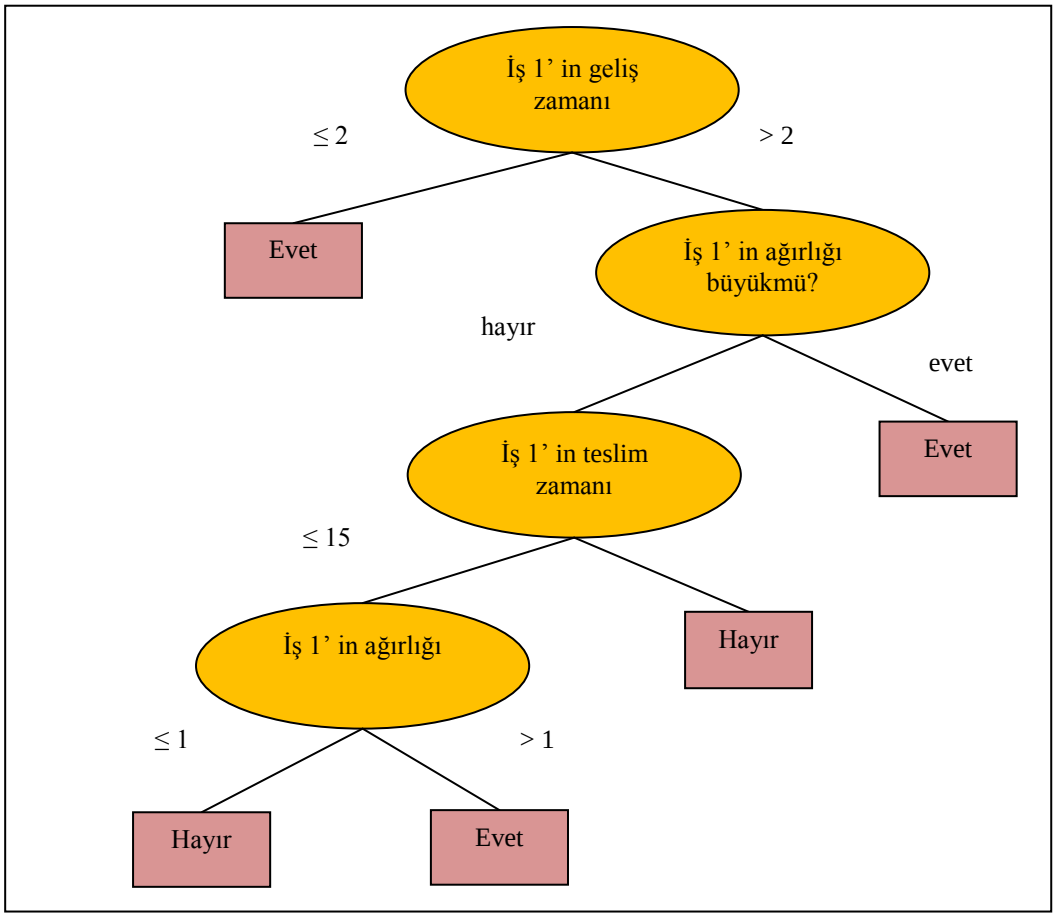
Kural 1: (93/7, lift 3.7)	**Kural 2:** (48, lift 3.8)	**Kural 7:** (45/6, lift 3.5)
İşlem süresi ≤ uzun	İşlem süresi ≤ uzun	İşlem süresi ≤ uzun
Kalan işlem süresi≤uzun	Kalan işlem süresi≤orta	Kalan işlem süresi≤kısa
Makine yüklemesi≤hafif	Makine yüklemesi≤hafif	Makine yüklemesi≤hafif
iş>3	iş≤1	iş>1
iş≤5	sınıf orta [0.980]	iş≤2
sınıf ilk [0.916]	...	sınıf son [0.851]

Şekil 2.7. Elde edilen kurallar

Bu kurallar başka sekiz iş dört makine akış tipi problemlerine uygulanmıştır. Çalışmada, karınca kolonisi algoritmasının veri madenciliği tabanlı algoritmaya göre çok daha başarılı sonuçlar verdiğinden ve optimum çözümden %7,04 sapmayla çalıştığından söz edilmiştir.

Gönülol (2009) çalışmasında veri madenciliği tabanlı yeni bir sezgisel yaklaşım ile, birçok alanda yaygın olarak kullanılan Gezgin Satıcı Probleminin (GSP) çözüm performansını araştırmıştır. Çok sayıda rassal çözüm ile eşik değer, kural ilişki matrisi oluşturmuş ve bu kurallara göre turu belirlemiştir. Eşik değer, üretilen rassal çözümlerin ortalamasıdır. Kural ilişki matrisi ise, problem büyüklüğü kadar satır ve sütundan oluşmaktadır. Bu matriste her turda, tüm bağlı olan şehirler için bir oran hesaplanmıştır. Bu oran, tur uzunluğunun eşik değerden çıkartılarak tur uzunluğuna bölünmesiyle bulunur. Her tura ait bu oran hesaplanıp tüm bağlı olan şehirlere bu oran atanır. Diğer tüm turların oranları bu matriste bağlı olan şehirlere eklenerek toplanır. Tur oluştururken, kural ilişki matrisinde yer alan şehirlerarasında ki en büyük oranlı iki şehir birbirine bağlanır. Daha sonra 2. en büyük orana sahip şehirler birbirine bağlanarak, tüm şehirler bitene kadar bu işleme devam edilir ve tur oluşturulur. Bu yöntemin performansının belirlenmesi amacıyla 15 test problemi kullanılmıştır. Elde edilen sonuçlar, ilgili problemin optimum sonuçları ile karşılaştırılmıştır. Geliştirilen veri madenciliği yaklaşımının, 300 şehirden az şehirli tüm gezgin satıcı problemlerinde iyi sonuçlar verdiği görülmüştür.

Olafsoon ve Li (2010) on işli tek makine problemini; geliş zamanı, teslim zamanı, işlem süresi, işlerin ağırlığı gibi kriterlerden yararlanarak, hem veri madenciliğinde kullanılan klasik karar ağacı ile hem de geliştirilmiş karar ağacı yapısı ile çözmüşlerdir. Sonuç olarak, klasik karar ağacı yaklaşımından daha iyi sonuçlar elde etmişlerdir. Şekil 2.8'de elde edilen karar ağacı yapısı gösterilmiştir.



Şekil 2.8. On işli tek makine problemi için karar ağacı

Thabtah ve Cowling (2010) personel çizelgeleme problemi için çözüm uzayı oluşturmuşlardır. Eğer bir problem birden fazla sezgisel çözümüne sahipse, hangi yöntemin daha başarılı olacağına karar vermek için bir sezgisel yöntem daha uygulamışlar ve her bir iterasyon içerisinde C4.5 algoritmasının da bulunduğu farklı sınıflandırma algoritmaları kullanarak, bu algoritmaları birbirleriyle karşılaştırmışlardır.

Zare ve Fakhizoal (2011) esnek akış tipi çizelgeleme probleminde toplam gecikme zamanını minimize etmeyi amaçlamaktadır. İyi sonuçlar bulmayı kolaylaştırmak amacıyla genetik algoritmayla veri madenciliğini birleştirmiştir. Parametre olarak, Koonce ve diğ., (1997) ile Koonce ve Tsai, (2000) çalışmalarına atıf yaparak operasyon süresi, arta kalan işlem süre, makinedeki yükleme ve teslim süresi parametrelerini kullanmışlardır. Diğer çalışmalardan farklı olarak çözümü geliştirmek amacıyla bulanık mantık yaklaşımı kullanılmıştır. Genetik algoritma veri madenciliği ve bulanık mantık yaklaşımının birleşiminden oluşan hibrit modelin, NP

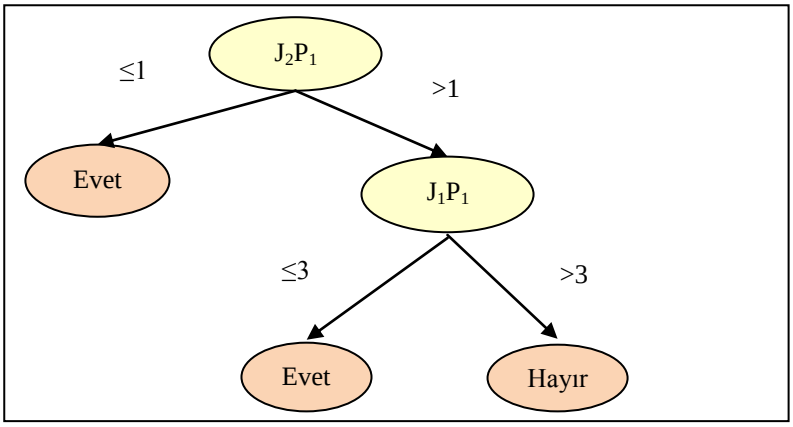
zor problemi olan esnek akış tipi çizelgeleme problemine iyi sonuçlar verdiği görülmüştür.

Balasundaram ve diğ. (2012) iki makine beş işli akış tipi çizelgeleme problemlerinde veri madenciliğinde karar ağaçlarını kullanarak kurallar üretmişlerdir. İşler sırasıyla ikili olarak karşılaştırılmıştır. 1. iş den sonra 2. iş geldiği durumdaki tamamlanma süresi ile 2. işten sonra 1. işin gelmesi durumundaki tamamlanma süresi hesaplanarak, en düşük tamamlanma süresine sahip iş başa alınmış, bu durum kalan işler için de yapılmış ve Tablo 2.11'de buna göre oluşturulmuş kural tablosu gösterilmiştir.

Tablo 2.11. Beş iş iki makine problemi için değerlendirme tablosu

NO	İŞ-1	J_1P_1	J_1P_2	İŞ-2	J_2P_1	J_2P_2	İŞ-1 İLK ÇİZELGELENİR
1	1	3	6	2	5	2	EVET
2	1	3	6	3	1	2	HAYIR
3	1	3	6	4	6	6	EVET
4	1	3	6	5	7	5	EVET
5	2	5	2	3	1	2	HAYIR
6	2	5	2	4	6	6	HAYIR
7	2	5	2	5	7	5	HAYIR
8	3	1	2	4	6	6	EVET
9	3	1	2	5	7	5	EVET
10	4	6	6	7	7	5	EVET

Şekil 2.9'de görüldüğü üzere karar ağacı yapısı oluşturularak kurallar oluşturulmuş ve bu yeni yaklaşımı Johnson ve genetik algoritmayla karşılaştırmışlardır. Üç algoritma da optimum sonucu vermiştir.



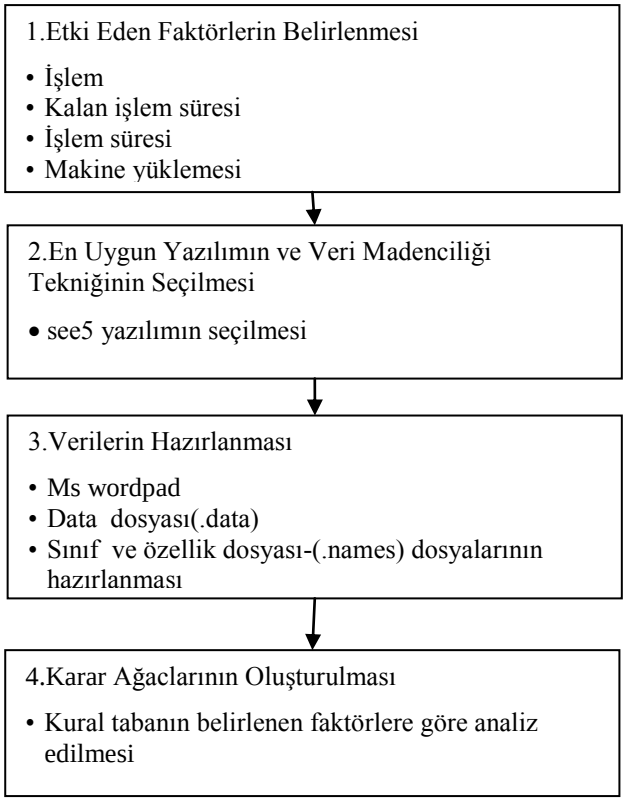
Şekil 2.9. Beş iş iki makine akış tipi problemi için karar ağacı

Literatürdeki çalışmaları özetlemek gerekirse, veri madenciliği ya küçük problemler üzerine uygulanmış ya da çözüm kalitesini geliştirmek amacıyla araç olarak kullanılmıştır. Hata oranı olarak altı iş altı makineli atölye tipi çizelgeleme probleminde %14 lük, sekiz iş dört makineli atölye tipi çizelgeleme probleminde ise %7,04 lük hatayla çalıştığı görülmüştür. Bölüm 2.1'de literatürdeki kriterlere göre Taillard'ın yirmi iş beş makine akış tipi çizelgeleme problemi için hata oranları hesaplanmıştır.

2.1. Büyük Ölçekli Problemlerin Çözümünde Veri Madenciliği Uygulaması

Veri madenciliğinde çizelgeleme alanında yapılan çalışmalar dikkate alındığında, yöntemin küçük boyutlu problemlerin çözümünde ya da metasezgisel yöntemlerden elde edilen çözümlerin kalitesinin artırılması amacıyla kullanılmış olduğu görülmüştür. Büyük boyutlu problemlerdeki çözüm kalitesinin belirlenmesi amacıyla Koonce (1997); Koonce (2000); Kumar ve Rao (2009) tarafından kullanılan kriterlerden yararlanılarak yirmi iş beş makine akış tipi çizelgeleme problemi çözülmüştür.

İlk olarak iş sırasına etki edecek faktörler belirlenerek iş sırası problemi için en uygun veri madenciliği tekniği ve yazılımı seçilmiştir. Veri tabanı iş sırasına etki edebilecek faktörlere göre hazırlanmıştır. Son olarak veriler karar ağacı ile analiz edilmiştir. Bu kurallar aynı büyüklükteki problemler için denenmiştir ve en iyi iş sıraları tahmin edilmiştir. Bu aşamalar Şekil 2.10'da gösterilmiştir.



Şekil 2.10. En iyi iş sırasını belirleme aşamaları

1. Etki Eden Faktörlerin Belirlenmesi: Literatür araştırması sonucunda iş sıralama problemine etki eden faktörler belirlenmiştir. Bu faktörler işlem, işlem süresi, arta kalan işlem süresi ve makine yüklemesidir.

2. En Uygun Yazılımın ve Veri Madenciliği Tekniğinin Seçilmesi: Veri madenciliği yazılımları ve teknikleri gözden geçirilmiştir. Bazı veri madenciliği araçları ve web adresleri Tablo 2.12'de gösterilmiştir.

Tablo 2.12. Veri madenciliği yazılımları ve teknikleri

Yazılım	Teknikler	Web adresi
ART	Karar Ağaçları	http://www.salford-systems.com
BayesiaLab	Sinir Ağları	http://www.bayesia.com
Clementine	Karar ağaçları,sinir ağları	http://www.spss.com/clementine/
Ctree	Karar Ağaçları	http://www.geocities.com/adotsaha/CTree/CTreeinexcel.html
Cubist 2.01	Numerik modeller	http://www.rulequest.com
Data Miner	Tahminleme, sınıflandırma,karar ağaçları	http://www.dataminer.com/quick.html
DTREG	Karar Ağaçları ve Regresyon Ağaçları	http://www.dtreg.com/
Evolver 4.0 Professional	Genetik Algoritma	http://www.sciencedowload.com/
GritBot 1.06	Veri Anomali	http://www.rulequest.com
Intellegent Miner	Karar Ağaçları	http://www.306ibm.com/software/data/iminer
Magnum Opus 2.0.1	Birliktelik Kuralları	http://www.rulequest.com
NeuNet Pro 2.3	Sinir Ağları	http://www.cormactech.com
Neuralware	Sinir Ağları	http://www.neuralware.com/products.jsp
Poly Analyst	Sinir ağları,karar ağaçları,sınıflandırma,doğrusal olmayan tahminleme	http://www.megaputer.com/
See5/c5.0	Sınıfların yapılandırılması	http://www.rulequest.com
Visirex	Kural çıkarımı ve karar ağaçları	http://www.cormactech.com
XLMiner	Sinir Ağları	http://www.xlminer.net/

Problemin yapısı göz önünde bulundurulduğunda, sınıflandırma konusunda diğerlerinden daha avantajlı yazılımın seçilmesinde fayda vardır. Bu bağlamda See5/C5.0 karar ağaçları konusunda (C4.5 ve C5) daha avantajlı olduğu Tablo 2.13'de görülmektedir. Ayrıca çizelgeleme alanında yapılan çalışmaların çoğunda see5 yazılımı kullanılmıştır. Bu nedenlerden dolayı çalışmada see5 yazılımı kullanılmıştır.

Tablo 2.13. Karar ağacı araçlarının karşılaştırılması

KARAR AĞAÇLARI	CART	C5 VEYA C4.5	CHAID	DİĞERLERİ	ÖNCELİKLER	SINIFLANDIRMA MALİYETLERİ	EKSİK VERİLER	BUDAMA YETENEĞİ	GÖRSELLİK
CLEMENTİNE		√				√	√	√	√-
DARWİN	√				√	√	√		
ENTERPRİCE MİNER	√	√-	√		√+	√	√	√	√
GAİNSMARTS	√		√	√			√		√
İNTELLİGENT MINER				√			√		√
MINESET	√		√			√	√	√	√
MODEL1	√		√				√-		√
MODELQUES		√-					√	√	
CART	√+				√	√	√		√
SCENARİO				√			√		
S-PLUS	√						√	√	√
SEE5		√+				√	√	√	

3. Verilerin Hazırlama Aşamaları: Veri hazırlama işlemi üç aşamada incelenebilir. İlk aşama problem sınırlarının belirlenmesidir.

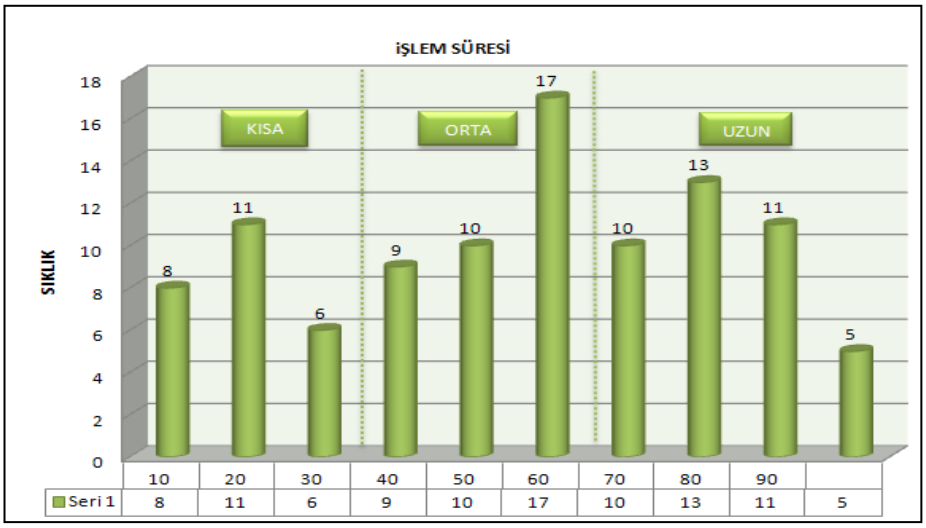
Problemin Sınırlarının Belirlenmesi: İlk olarak problemin sınırları belirlenmiştir. Problem tipinin ne olduğu ve buna bağlı olarak makine iş ve operasyonların belirlenmesi gerekir.

Verilerin Sınırlarını Çizmek ve Kodlamak: Tablo 2.14'de görüldüğü üzere analiz için sınırların çizilmesi gerekir. Yirmi iş beş makine akış tipi çizelgeleme problemi için veri sınıfları oluşturulmuştur.

Tablo 2.14. Sınırlar ve kodlar

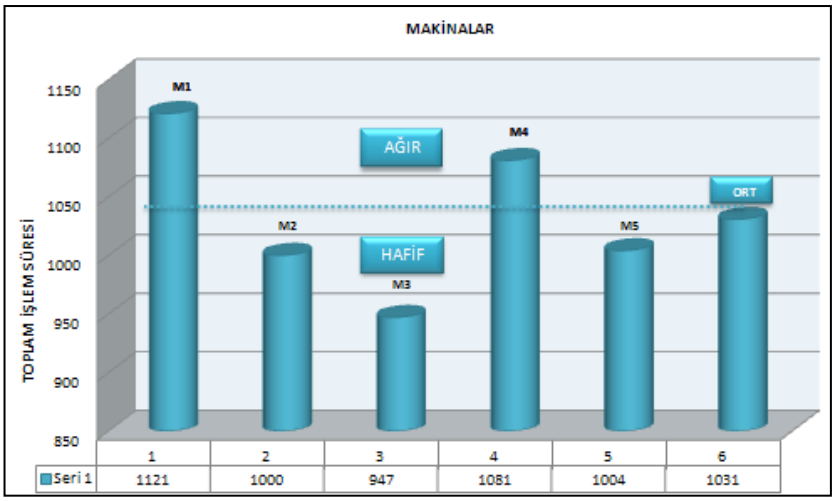
Faktörler	Sınırlar	Kodlar
İşlem	1	İlk
	2-3	Orta
	4	Son
	5	En son
İşlem Süresi	İşlem süresi<=30	Kısa
	30<işlem süresi<60	Orta
	İşlem süresi>=60	Uzun
Kalan İşlem Süresi	Kalan işlem süresi<=90	Kısa
	90<kalan işlem süresi<=180	Orta
	Kalan işlem süresi>180	Uzun
Makine Yüklemesi	Top. iş yük.< Ortalamadan(1030)	Hafif
	Top.iş yük >=Ortalamadan(1030)	Ağır

İşlem süresi çizelgeleme problemler için kullanılan kriterlerden biridir. İşlem süresi 0 ile 99 arasında değişmektedir. Şekil 2.11'da görüldüğü üzere, işlem süresi kısa, orta ve uzun olmak üzere üç sınıfa ayrılmıştır.



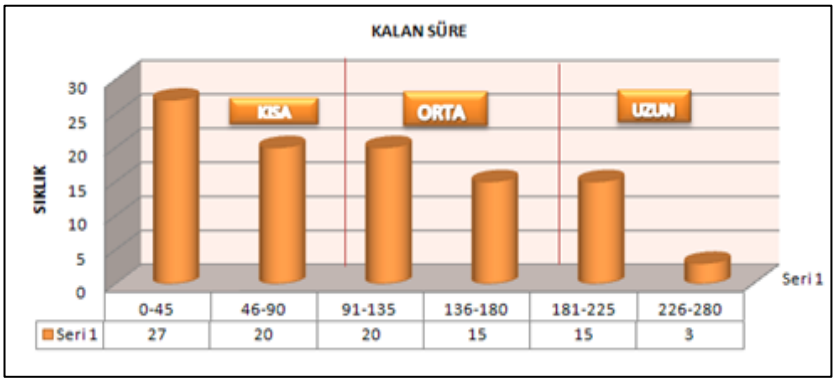
Şekil 2.11. İşlem süresi genellemesi

Makine yüklemesinde, her bir makinenin toplam işlem süresi hesaplanarak iş yükü bakımından ağır olup olmadığı belirlenmiştir. Makinelerin iş yüklemesi sırasıyla 1121, 1000, 947, 1081, 1004 olarak hesaplanmıştır. Tüm makinelerin iş yüklerinin ortalaması 1031'dir. Şekil 2.12'de görüldüğü üzere, ortalamanın altındaki iş süreleri hafif üstündeki iş süreleri ise ağır olarak belirlenmiştir.



Şekil 2.12. Makine yüklemesi genellemesi

Şekil 2.13'de arta kalan işlem sürelerinin genelleştirilmesi görülmektedir. Eğer arta kalan işlem süresi 90'dan küçükse kısa, 90 ile 180 arasında ise orta, 180'den büyük ise işlem süreleri uzun olarak sınıflanmıştır.



Şekil 2.13. Arta kalan işlem süresinin genellemesi

Tablo 2.15'de dört farklı parametreye göre yapılan sınıflandırma ayrıntılı bir biçimde gösterilmektedir. 20 tane öncelik sınıfı bulunmaktadır. Toplam 400 veri girişi yapılmıştır.

Tablo 2.15. Özelliklerin detaylı sınıflanması

İş	İşlem	Makine	İşlem süresi	Kalan işlem süresi	İşlem	İşlem süresi	Kalan işlem süresi	Makina yükleme	Öncelik
1	1	1	54	219	ilk	orta	uzun	ağır	S16
1	2	2	79	140	orta	uzun	orta	hafif	S16
1	3	3	16	124	orta	kısa	orta	hafif	S16
1	4	4	66	58	son	uzun	kısa	ağır	S16
2	1	1	83	206	ilk	uzun	uzun	ağır	S17
2	2	2	3	203	orta	kısa	uzun	hafif	S17
2	3	3	89	114	orta	uzun	orta	hafif	S17
3	1	1	15	111	ilk	kısa	orta	ağır	S5
….	….	….	….	….	….	….	….	….	….
….	….	….	….	….	….	….	….	….	….
19	3	3	77	119	orta	uzun	orta	hafif	S13
19	4	4	51	68	son	orta	kısa	ağır	S13
19	5	5	68	0	enson	uzun	kısa	hafif	S13
20	1	1	94	176	ilk	uzun	orta	ağır	S19
20	2	2	77	99	orta	uzun	orta	hafif	S19
20	3	3	40	59	orta	orta	kısa	hafif	S19
20	4	4	31	28	son	orta	kısa	ağır	S19
20	5	5	28	0	enson	kısa	kısa	hafif	S19

Verinin Hazırlanması: See5 yazılımında karar ağacı uygulaması için iki dosya gereklidir. İlk gerekli olan dosya, sınıfların ve açıklamaların bulunduğu, uzantısı .names olan isim dosyasıdır. Veri girilmeden önce dosyaismi.names adı altında oluşturulacak dosyaya alan adları ve herbir alanın alabileceği değer birimlerinin girilmesi gerekir. Bu değer birimlerinden biri continuous tanımlaması, karşılığına denk gelen ismin yada alanın alabileceği değerin sayısal değerler olduğunu göstermektedir ve f, t değerleri ise doğru/yanlış tanımlamasına karşılık gelmektedir. Zaman ataması, zamanı tutmaktadır ve HH:MM:SS formatında yazılmaktadır (Silahtaroğlu, 2008). İkinci dosya ise sadece verilerin bulunduğu, uzantısı .data olan veri dosyasıdır. Tablo 2.16'da dosya türlerinden örnekler gösterilmiştir. Değerler arasına (,) kullanılarak veriler gösterilmektedir.

Tablo 2.16. Sınıf ve veri dosyası

Sınıf ve Özellik Dosyası	Veri Dosyası
sira. | the target attribute operasyon: ilk, orta, son, enson. birimsure: kısa, orta, uzun. kalansure: kısa, orta, uzun. makineyuk: ağır, hafif. sira: S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13,S14, S15, S16, S17, S18, S19, S20.	ilk,orta,uzun,ağır,S16 orta,uzun,orta,hafif,S17 orta,kısa,orta,hafif,S9 son,uzun,kısa,ağır,S1 enson,orta,kısa,hafif,S8 …..

Şekil 2.14'de görüldüğü üzere, See5 yazılımı ile elde edilen kurallar aşağıdaki gibidir.

```
Kural 1: (24/20, lift 3.8)
        birimsure = orta
        kalansure = orta
        makineyuku = hafif
        ->  sınıf S1  [0.192]

Kural 2: (24/20, lift 3.8)
        operasyon = enson
        birimsure = uzun
        ->  sınıf S1  [0.192]

Kural 3: (100/92, lift 1.8)
        birimsure = kısa
        ->  sınıf S1  [0.088]

Kural 4: (144/132, lift 1.8)
        birimsure = o
        ->  class S2  [0.089]

Kural 5: (12/9, lift 5.7)
        birimsure = kısa
        kalansure = uzun
        makineyuku = ağır
        ->  sınıf S6  [0.286]

Kural 6: (40/32, lift 4.3)
        birimsure = kısa
        kalansure = orta
        makineyuku = hafif
        ->  sınıf S7  [0.214]

Kural 7: (96/87, lift 2.0)
        birimsure = orta
        kalansure = kısa
        ->  sınıf S9  [0.102]

Kural 8: (4, lift 16.7)
        birimsure = orta
        kalansure = uzun
        makineyuku = hafif
        ->  sınıf S12  [0.833]

Kural 9: (156/140, lift 2.2)
        birimsure = uzun
        ->  sınıf S13  [0.108]

Kural 10: (16/12, lift 5.6)
        birimsure = orta
        kalansure = uzun
        makineyuku = ağır
        ->  sınıf S15  [0.278]

Kural 11: (8/4, lift 10.0)
        birimsure = kısa
        kalansure = uzun
        makineyuku = hafif
        ->  sınıf S17  [0.500]

Kural 12: (8/4, lift 10.0)
        operasyon = ilk
        birimsure = uzun
        kalansure = orta
        ->  sınıf S19  [0.500]

Kural 13: (8/4, lift 10.0)
        operasyon = orta
        birimsure = uzun
        kalansure = kısa
        ->  sınıf S20  [0.500]

Kural 14: (32/24, lift 5.3)
        birimsure = kısa
        kalansure = kısa
        ->  sınıf S20  [0.265]
```

Şekil 2.14. Elde edilen kurallar seti

See5 yazılımı ile 14 adet kural üretilmiştir. 400 verinin 322 tanesi hatalı sınıflanmış olup %80.5 hata payıyla çalışmaktadır. Özellik kullanım oranı olarak %100 birim süre, %62 arta kalan süre, %26 makine yükü, %10 operasyon kulanılmıştır.

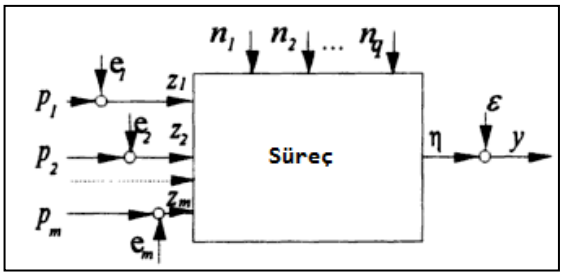
Özetle büyük problemlerdeki hata oranı, küçük problemlere göre daha fazladır. Elde edilen sonuçlar, veri madenciliği algoritmasının küçük boyuttaki problemler için sezgisellerden daha iyi performans göstermesine karşın meta sezgisellere göre düşük performansta olduğu ve konunun gelişmeye açık bir alan olduğu (Koonce, 1997; Koonce, 2000; Santos, 2006) görüşünü desteklemektedir.

3. DENEY TASARIMI VE DENEY TASARIMI YÖNTEMLERİ

İstatistiksel olarak kalite iyileştirme ve deney tasarımı, birçok işletme için önem verilen konulardır. Deney tasarımı (DT), verimliliği artırmak, ürün ve süreç geliştirmek için kullanılan güçlü bir yaklaşımdır (Jiju ve Preece, 2002). Deney tasarımı; bağımlı faktördeki değişikliğin nedeni olarak ele alınan bağımsız faktörün etkilerinin ölçülmesi işlemini yürütmek için kullanılan tekniktir (Çömlekçi, 2003). Deney tasarımı, bir prosesteki girdi faktörleri üzerine istenilen değişikliklerin sistematik bir şekilde yapılmasıyla, cevap değişkeni üzerindeki değişkenliğin gözlenmesi ve yorumlanması olarak da tanımlanabilir (Soylak, 2000). Taguchi deney tasarımı aracılığıyla ile proses iyileştirme ve ürün geliştirmenin yanında bu yöntemin optimum faktör seviyelerinin belirlenmesinde kullanılan bir tekniktir. Özellikle çizelgeleme alanında içsel parametrelerin deney tasarımı ile belirlenmesiyle daha kaliteli sonuçlar elde edilmektedir.

Şekil 3.1' de ürün ya da sürecin performans karakteristikleri gösterilmiştir. (örneğin gerilme, viskozite, çıkış voltaj gibi..). p; ürün ya da süreçteki parametreleri, e; ürün ya da süreçteki hataları, η; ürünün /sürecin parametrelerinin deterministik fonksiyonunu ifade etmektedir (örneğin sıcaklık,basınç,konsantrasyon,giriş voltajı, gibi). n; gürültü faktörü olup, seri üretimde (ortam sıcaklığı, nem, ürün kullanımı esnasındaki titreşim) kontrol edilmesi mümkün olmayabilir, ε; rassal etkileri göstermektedir (Vuchkov ve Boyadjieva, 2001). Son olarak, ürün için performans karakteristiğinin gerçek değeri Eşitlik (3.1)' deki gibidir.

$$y = \eta(p_1 + e_1, p_2 + e_2, ... p_m + e_m, n_1, n_2, ..., n_q) + \varepsilon \qquad (3.1)$$



Şekil 3.1. Performans karakteristiği, ürün parametreleri, gürültü faktörleri

1920 li yılların başlarında Ronald Fisher çalışmalarına tarım alanında istatistiksel deney tasarımı ile başlamış, rassallık, replikasyon ve bloklama konularında temel prensipler geliştirmiştir. Box ve Wilson 1951 yıllında istatistiksel olarak tarımsal ve endüstriyel uygulamalar arasında farkları ortaya koymuş ve Fisherin çalışmalarını biraz daha ileri götürmüştür. 1970 yıllarında Genichi Taguchi tarafından deney tasarımı alanındaki çalışmalar daha ileri boyutta incelenmiştir (Wilcox, 2007). Günümüzde de deney tasarımı ile ilgili çok sayıda kullanım alanı olduğu görülmektedir. Bu çalışmada deney tasarımı, parametre optimizasyonu amacıyla kullanıldığından aşağıda, bu alandaki çalışmalar özetlenmiştir. İlk çalışmalardan biri Cavory ve diğ. (2001) tarafından yapılmıştır. Her bir makinada önleyici bakım çizelgesinin optimize edilmesi için genetik algoritma kullanılmıştır, Mutasyon, başlangıç değerleri (rassal ya da lokal optimum sonuçlar) ve seçim mekanizması (rulet ya da turnuva seçim mekanizmaları) olmak üzere 3 parametre için karma desen kullanılarak taguchi analizi yapılmıştır.

Cheng ve Chang (2007) çalışmalarında çoğu yazarın genetik algoritmada kullandıkları parametreleri belirlerken literatürdeki çalışmalardan yararlandığından veya deneme yanılma yoluyla zaman kaybettiğinden ve optimal çözüme ulaşamadıklarından bahsetmişlerdir. Sonuç olarak, optimum parametre seviyelerinin belirlenmesinde taguchi yönteminin kullanılması gerektiğini savunmuşlardır. Genetik algoritma parametreleri olarak da; popülasyon, başlangıç çözümü (rassal, NEH, NEH-Modi); seçim mekanizması (rulet-turnuva-sıra seçim), çaprazlama ve mutasyonu belirlemişlerdir. Bunun sonucunda da en önemli parametrenin başlangıç çözümü olduğunu ortaya koymuşlardır.

Naderi ve diğ. (2009) tarafından sıra bağımlı hazırlık ve taşıma süresine sahip hibrit akış tipi çizelgeleme problemi için tavlama benzetimi algoritması geliştirilmiş ve toplam tamamlanma ve gecikme süresinin minimize edilmesi amaçlanmıştır. Kontrol edilebilir faktörler olarak; başlangıç çözümü, başlangıç sıcaklığı, her bir sıcaklıktaki komşuluk arama, başlangıç ve bitiş sıcaklıkları arasındaki iterasyon sayısı, komşuluk arama yapısı belirlenmiş ve bu parametreler taguchi yöntemiyle analiz edilmiştir. Yine aynı yazar tarafından sıra bağımlı atölye tipi çizelgeleme problemlerine yönelik, yeni operatörlere sahip tavlama benzetimi algoritması geliştirilmiştir. Bu farklı tip operatörlerin ve parametrelerin analizinde yine Taguchi metodu kullanılmıştır (Naderi ve diğ., 2010a). Aynı yıl yapılan bu çalışmada tek makine çizelgeleme problemi ele alınmıştır. Diğer çalışmalardan farklı olan yönü iki farklı grup iş olmasıdır. İlk grup işin amaç fonksiyonu, toplam ağırlıklı tamamlanma süresinin minimizasyonu, ikinci grubun amaç fonksiyonu ise maksimum gecikmenin minimizasyonudur. Genetik algoritma, hibrit kanguru ve tavlama benzetimi algoritmaları kullanılmış ve parametre optimizasyonu aracı olarak da taguchi yöntemi kullanılmıştır (Naderi ve diğ., 2010b).

Delavar ve diğ. (2010) üretim ve hava taşımacılığının koordineli olarak çizelgelenmesi için iki farklı genetik algoritma yaklaşımı kullanarak minimum maliyet hedeflemişlerdir. İki model için de belirlenen parametreler taşımacılıkta başlangıç çözümü, taşımacılıkta populasyon büyüklüğü, yeniden taşıma yüzdesi, taşımacılıkta mutasyon olasılığı, üretimde yeniden üretim yüzdesi, üretimde mutasyon olasılığı, üretimde başlangıç çözümüdür. Soltani ve Sadjadi (2010) tarafından çapraz sevkiyat sistemlerindeki çizelgeleme problemi ele alınmış, toplam akış süresinin minimize edilmesi amaçlanmıştır. İki farklı hibrit metot kullanılmış (hibrit tavlama benzetimi ve hibrit değişken komşuluk arama), iki yöntem için ayrı ayrı taguchi analiziyle parametre optimizasyonu yapılmıştır. Aynı konu bir de Arabani (2011) tarafından çalışılmış olup çapraz sevkiyat optimizasyonu için genetik algoritma (GA), parçacık sürü optimizasyonu (PSO), karınca kolonileri (ACO), diferansiyel gelişim (DE) algoritması ve tabu arama (TS) yöntemleri ve bu yöntemlerin parametre optimizasyonunda taguchi metodu kullanılmıştır. Parametre olarak, GA için; popülasyon büyüklüğü, mutasyon, olasılık, seçim mekanizması (rulet ve turnuva seçim mekanizmaları), PSO için; popülasyon büyüklüğü, kalıtım,

maksimum ve minimum atalet ağırlığı, ACO için; popülasyon büyüklüğü, buharlaşma oranı, q ve β sabiti ve başlangıç feromonu belirlenmiştir. Naderi ve diğ. (2011), paralel makineli açık atölye tipi çizelgeleme problemi için memetik algoritma uygulamıştır. Bu çalışmadaki amaç, atölyedeki etkinliği artırabilmek için kısıt olan süreçleri azaltmak ya da elimine etmektir. Memetik algoritmadaki çaprazlama operatörü, mutasyon operatörü, mutasyon olasılığı ve popülasyon büyüklüğü parametreleri Taguchi metodu kullanılarak optimize edilmiştir. Safari ve diğ. (2011) tarafından makine arızası ve önleyici bakım kısıtı altında akış tipi çizelgeleme problemi için hibrit model önerilmiştir. Genetik ve tavlama benzetimi yöntemleri tabanlı modelin parametreleri başlangıç çözümü (rassal, NEH), başlangıç sıcaklığı, soğutma oranı, çaprazlama oranı, mutasyon oranı, populasyon büyüklüğü, tamamlanma kriteri ve komşuluk yapısı olmak üzere sekiz tanedir. Bu parametrelerin optimizasyonu için taguchi yöntemi kullanılmıştır.

Sadjedi ve diğ. (2011) tarafından işgücü taleplerinin belirsizliği altında, yeni karma tam sayılı doğrusal olmayan stokastik personel çizelgeleme problemi ele alınmıştır. Simülasyon tekniği ve genetik algoritma ile çözüm sunulmuştur. Bu çalışmada da parametre optimizasyonu için taguchi yöntemi kullanılmıştır. Bu parametreler: populasyon büyüklüğü, mutasyon ve çaprazlama oranı, çaprazlama operatörü, mutasyon ve çaprazlama operatörü tipidir.

Wang ve diğ. (2011) tarafından esnek atölye tipi çizelgeleme problemi için iki populasyonlu dağıtım algoritması kullanılmış, maksimum tamamlanma süresinin kısaltılması amaçlanmıştır. Bu algoritmanın etkinliğini artırabilmek için taguchi yöntemiyle parametre optimizasyonu yapılmıştır. Her bir problem için nesil sayısı (Gen=10 x is sayısı(n) x makine sayısı(m)) ve populasyon büyüklüğü sayısı (P=n x m) belirlenmiştir. Diğer parametreler olan seçilme oranını güncelleyen η değeri, öğrenme oranları α ve β ile bölünmeyi sağlayan her parametre için en uygun seviyeler belirlenmiştir.

3.1. Deney Tasarım Yöntemleri

Deney tasarımından beklenen minimum zamanda optimum faktör seviyelerinin elde edilmesidir. Bu yüzden eş zamanlı analizin mümkün olmadığı geleneksel yöntemlerin yerine istatistiksel deney tasarımları yapılmalıdır. Bunlar;

3.1.1. Bir kerede bir faktör

Geleneksel yöntem ya da klasik yöntem olarak adlandırılır. Her defasında bir parametre değiştirilir. Temel seviye seçilir ve seçilen parametreler kendi aralarında değiştirilir ve etkiler analiz edilir.

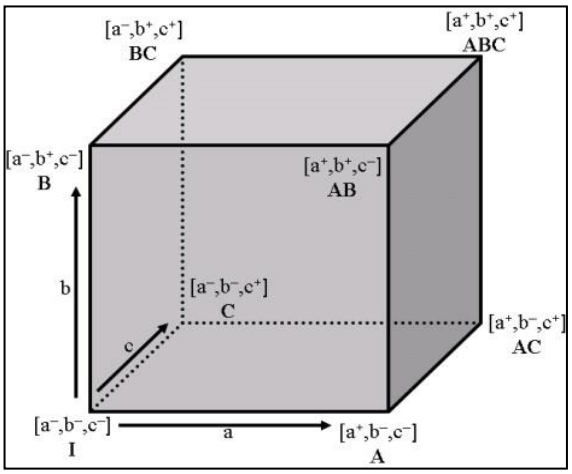
3.1.2. İstatistiksel deney tasarımı

Geleneksel yöntemlerin yetersizliği, istatistiksel deney tasarım yöntemleriyle giderilmiştir. Bu yöntemleri üçe ayırabiliriz.

3.1.3. Tam faktöriyel deney tasarımı

Tüm kombinasyonların yapıldığı deney tasarımı yöntemidir (Sas, 2008). Bu yüzden maliyetli bir yaklaşımdır. Örneğin 5 parametreli bir deneyde her bir parametre 3 seviyeye sahipse, tek replikasyon için toplam 3x3x3x3x3 =243, 5 replikasyon için 1215 adet deney yapılması gerekmektedir.

Şekil 3.2 de görüldüğü üzere üç faktörlü (a,b ve c) her birinde iki seviye olan toplam $2\times2\times2 = 2^3 = 8$ kombinasyona sahip tasarım küp şeklinde gösterilmiştir. Genel olarak, n faktör ve m seviyeli tam faktöriyel deneyler mn kombinasyon sayısına sahiptir. Eğer her bir faktör $i \in 1,2,\cdots,n$ mi seviyeye sahipse, kombinasyon sayısı $\prod_{i=1}^{n} m_i$ dir (Louviere, 2000).



Şekil 3.2. İki seviye üç faktör için tüm olası kombinasyonları

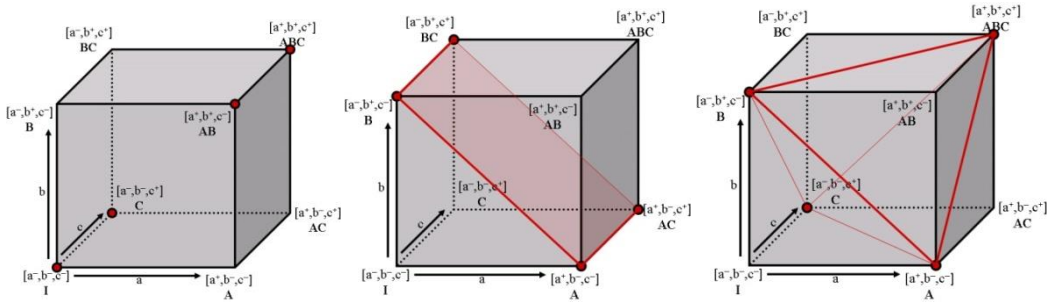
3.1.4. Kesirli faktöriyel deney tasarımı

Birçok uygulamada faktör sayısı ve/veya seviyelerinin sayısına bağlı kaynak kısıtı nedeniyle tam faktöriyel tasarımı yapmak mümkün olamamaktadır. Böyle bir durumda, tam faktöriyel deney yerine kesirli deney tasarımına yönlenilir. Ancak kesirli deney tasarımı ile analiz yaparken hangi alt kümenin seçilmesi gerektiğine karar verilmelidir. Çünkü yüksek mertebeden etkileşimler ihmal edilmemelidir. Örneğin 7 parametreli ve 2'şer seviyeli bir deney tam faktöriyel olarak yapıldığında $2^7 = 128$ deney yapılması gerekir yani yüksek mertebeden etkileşimler ihmal edilerek 64 deney ya da kesiri 32 deneyle de sonuca ulaşılabilir. Deney sayısını kesirli olarak azaltmak tamamen araştırmacıların elindedir. Tablo 3.1'de iki seviye üç faktörden oluşan bir deney için kullanılan seviyeler gösterilmiştir.

Tablo 3.1. Faktörlerin seviyeleri ve isimlendirilmesi

A	B	C	İsim	A	B	C	İsim
-	-	-	I	+	-	-	A
-	-	+	C	+	-	+	AC
-	+	-	B	+	+	-	AB
-	+	+	BC	+	+	+	ABC

İki seviye üç faktörden oluşan bir deney tasarımında, : I-C-AB-ABC; A-B-AC-BC; A-B-C-ABC olmak üzere üç farklı desen söz konusudur (Wu ve Hamada, 2000). Şekil 3.3'de üç farklı desen gösterilmiştir.



Şekil 3.3. Kesirli deney tasarımı

I-C-AB-ABC deney düzeneği seçildiğinde, A, B ve C faktörlerinin ilk seviyeleri (I); A ve B faktörünün ilk seviyeleri, C faktörünün ikinci seviyesi (C); A ve B faktörünün ikinci seviyeleri, C faktörünün ilk seviyesi (AB); A, B ve C faktörlerinin ikinci seviyesi (ABC) olmak üzere dört farklı kombinasyonla deney tasarımı

oluşturulmalıdır. İkinci deney düzeneği için A faktörü için ikinci seviye, B ve C faktörleri için ilk seviye (A); A ve C faktörü için ilk seviyeler, B faktörü için ikinci seviye (B); A ve C faktörleri için ikinci seviyeler, B faktörü için ilk seviye (AC); A faktörü için ilk seviye, B ve C faktörü için ikinci seviye seçilmelidir (BC). Son düzenekte ise, A faktörünün ikinci seviyesi, diğer faktörlerin ilk seviyeleri (A); B faktörünün ikinci seviyesi, diğer faktörlerin ilk seviyeleri (B); C faktörünün ikinci seviyesi diğer faktörlerin ilk seviyeleri (C); tüm faktörlerin ikinci seviyeleri (ABC) olmak üzere dört kombinasyon oluşturulur. Tablo 3.2'de kesirli deney tasarımı için kullanılan desenler görülmektedir.

Tablo 3.2. Kesirli deney tasarımı için desenler

I.Desen					II.Desen				III.Desen			
	A	B	C	İsim	A	B	C	İsim	A	B	C	İsim
1	-	-	-	I	+	-	-	A	+	-	-	A
2	-	-	+	C	-	+	-	B	-	+	-	B
3	+	+	-	AB	+	-	+	AC	-	-	+	C
4	+	+	+	ABC	-	+	+	BC	+	+	+	ABC

3.1.4.1. Taguchi yöntemi

Taguchi yönteminin parametre, sistem ve tolerans olmak üzere üç aşaması vardır (Krishnaiah ve Shahabudeen, 2012). Sistem tasarımı bir ürünün üretiminde gerekli olan bilimsel uygulama ve mühendislik bilgisi içerir. Parametre tasarımı kalite özelliklerinin iyileştirilmesi amacıyla en uygun işlem değerlerini bulmak için kullanılır. Tolerans tasarımı, parametre tasarımında belirlenen optimum ayarların analizinde ve belirlenmesinde kullanılır (Özçelik ve diğ., 2010).

Taguchi yöntemindeki ortogonal dizilerle, zaman ve maliyetler azaltılır. Süreç değişkenliğinin ölçüsü olan S/G oranı, yanıt değişkenin hem ortalaması hem de varyansı ile ilgili ölçümü söz konusudur (Tong ve diğ., 2004). S/G oranı üç şekildedir: Elde edilmesi istenen yanıt değişkeninin maksimum olması isteniyorsa en büyük en iyinin, hedeflenen bir seviyede olması isteniyorsa nominal en iyinin, minimum olması isteniyorsa, minimum en iyinin seçilmesi gerekir. S/G oranına ek olarak, ANOVA ile süreç parametrelerinin etkileri belirlenebilir. Böylece optimum seviyeler ve en önemli faktörler belirlenmiş olacaktır. Taguchi deney tasarımı ile

deney sayısını azaltarak kısa sürede sonuca ulaşılabilmesi ayrıca S/G değeriyle varyasyonu da ortadan kaldırarak daha gürbüz bir tasarım söz konusu olmaktadır.

3.2. Taguchi Deneysel Tasarım Metodunun Prosedürleri

Taguchi deneysel tasarım metodunda izlenecek temel aşamalar aşağıdaki gibidir.

3.2.1. Çözülecek problemin belirlenmesi

İlk aşama, problemin detaylı bir şekilde ele alınmasıdır. Problemin iyi bir şekilde analizi deneyin modellenmesi için önemlidir. Süreçte en önemli kısıt olduğuna inanılan problemden başlamak ve hedefe odaklanmak kısa süre içersinde verimliliği artıracaktır.

3.2.2. Faktörlerin ve seviyelerin belirlenmesi

Performans karakteristiğini etkileyeceğine inanılan faktörler çok sayıda uzman kişi tarafından belirlenmelidir. Faktörlerin doğru ve eksiksiz olarak belirlenmesinde beyin fırtınası ve/veya balık kılçığı diyagramları kullanılabilir.

Tablo 3.3'de tek seviyeli taguchi tasarımları kullanılmaktadır. Tablo içindeki sayılar her tasarım için minimum ve maksimum faktör sayılarını göstermektedir. Örneğin L8 (2^7) tasarımı iki ile yedi arasında değişen faktör sayısına sahip iki seviyeli desenler için geçerlidir. L16 (4^5) tasarımı da iki ile beş arasında faktör sayısına sahip ve her bir faktör seviyesinin 4 olduğu durumlarda geçerlidir.

Tablo 3.3. Eşit seviyeli deney tasarımları

Tasarımlar	2	3	4	5
L4 (2^3)	2-3			
L8 (2^7)	2-7			
L9 (3^4)		2-4		
L12 (2^11)	2-11			
L16 (2^15)	2-15			
L16 (4^5)			2-5	
L25 (5^6)				2-6
L27 (3^13)		2-13		
L32 (2^31)	2-31			

Karma Desenler: Tablo 3.4'de seviyeleri (2-4),(2-3),(2-8),(3-6) seviyeli tasarımlar özetlenmiştir. Tablo içindeki hücrelerdeki sayılar herbir seviye için maksimum ve minimum faktör sayılarını göstermektedir. Örneğin 2 ve 4 seviyeden oluşan karışık tasarımlar için, L8 (2^4, 4^1) tasarımı bir ile dört arasında değişen iki seviyeli ve dört seviyeli bir faktörden oluşmaktadır. İki ve üç seviyeli tasarımlar için, L18 (2^1, 3^7) tasarımı bir ile yedi arasında değişen üç seviyeli ve iki seviyeli bir faktörden oluşmaktadır. İki ve sekiz seviyeli tasarımlar için, L16 (2^8, 8^1) tasarımı bir ile sekiz arasında değişen iki seviyeli ve sekiz seviyeli bir faktörden oluşmaktadır. Üç ve altı seviyeli tasarımlar için, L18 (3^6, 6^1) tasarımı bir ile altı arasında değişen üç seviyeli ve altı seviyeli bir faktörden oluşmaktadır.

Tablo 3.4. Karma seviyedeki tasarımlar

	Seviyeler		**Seviyeler**		
Tasarımlar	**2**	**4**	**Tasarımlar**	**2**	**3**
L8 (2^4, 4^1)	1-4	1	L18 (2^1, 3^7)	1	1-7
L16 (2^12, 4^1)	2-12	1	L36 (2^11, 3^12)	1-11	2-12
L16 (2^9, 4^2)	1-9	2	L36 (2^3, 3^13)	1-3	13
L16 (2^6, 4^3)	1-6	3	L54 (2^1, 3^25)	1	3-25
L16 (2^3, 4^4)	1-3	4	**Tasarımlar**	**2**	**8**
L32 (2^1, 4^9)	1	2-9	L16 (2^8, 8^1)	1-8	1
			Tasarımlar	**3**	**6**
			L18 (3^6, 6^1)	1-6	1

3.2.3. Ortogonal diziler ve seçimleri

Ortogonal dizilerle çok sayıda faktör minimum deneme yapılarak optimize edilmeye çalışılmıştır. Böylece kısa süre içerisinde minimum maliyetle faktör seviyelerini tek tek değiştirmek yerine eş zamanlı değiştirebilme imkânı vardır. Örneğin üç seviyeli beş faktörlü tam faktöriyel tasarım için 243 deneme yapılırken ortogonal diziyle sadece 27 deneme yapılacaktır. Tablo 3.5'de beş faktör üç seviyeli ortogonal dizi verilmiştir.

Tablo 3.5. Beş faktör üç seviyeli ortogonal dizi tablosu

Deneme No	Faktör 1	Faktör 2	Faktör 3	Faktör 4	Faktör 5
1	1	1	1	1	1
2	1	1	1	1	2
3	1	1	1	1	3
4	1	2	2	2	1
5	1	2	2	2	2
6	1	2	2	2	3
7	1	3	3	3	1
8	1	3	3	3	2
9	1	3	3	3	3
10	2	1	2	3	1
11	2	1	2	3	2
12	2	1	2	3	3
13	2	2	3	1	1
14	2	2	3	1	2
15	2	2	3	1	3
16	2	3	1	2	1
17	2	3	1	2	2
18	2	3	1	2	3
19	3	1	3	2	1
20	3	1	3	2	2
21	3	1	3	2	3
22	3	2	1	3	1
23	3	2	1	3	2
24	3	2	1	3	3
25	3	3	2	1	1
26	3	3	2	1	2
27	3	3	2	1	3

3.2.4. Faktörlerin kolonlara atanması

Bir deney tasarım matrisi tüm plandaki her bir deneyi tanımlayan deneysel şartların bir özetidir. Faktörler ortogonal dizi sütunlarına atanırken öncelikle seviye durumuna ve faktörler arasındaki etkileşim olup olmadığına bakılır. Eğer faktörler birbirinden bağımsız ise her hangi bir faktör herhangi bir sütuna atanabilir. Eğer faktörler arasında tespit edilmiş bir etkileşim var ise, o zaman belli kurallar gereği atama sütunlara zorunlu olarak yapılır.

3.2.5. Deneylerin gerçekleştirilmesi ve verilerin toplanması

Ortogonal dizisine uygun olarak deneyler gerçekleştirilir. Her bir deneme için kaç replikasyon yapılması gerektiğinede karar verilmelidir. Veriler toplandıktan sonra S/G gürültü oranları hesaplanmalıdır.

En küçük En iyi: Kalite değişkeni Y'nin, hedef değeri sıfırdır. S/G oranı aşağıdaki Eşitlik (3.2)'deki gibi hesaplanmaktadır.

$$S/G = log(\Sigma Y^2/n) \tag{3.2}$$

En Büyük-En iyi: Bu durumda Y'nin hedef değeri sonsuzdur ve sinyal/ gürültü oranı Eşitlik (3.3)'deki gibi hesaplanmaktadır.

$$S/G = 10\,log[\Sigma(1/Y^2)/n] \tag{3.3}$$

Hedef Değer-En iyi: Bu tür problemlerde, Y için belli bir hedef değer (örneğin, ürün boyutları gibi) verilmiştir. Bu durumda Eşitlik (3.4)'deki gibi hesaplanmaktadır.

$$S/G = 10\,log(\overline{Y^2}/S^2) \tag{3.4}$$

Bu üç problemde de amaç S/G oranını maksimize etmektir.

3.2.6. Verilerin analiz edilmesi ve optimum seviye seçimi

Elde edilen gözlem değerlerinin analizinde, varyans analizi, faktör etkilerinin grafiksel gösterimi metodu, hesap tablosu metodu gibi yöntemler kullanılmaktadır. Seviyeler arasındaki S/G oran değişimi, faktörlerin önem derecesini belirler.

4. ROTA BİRLEŞTİRME ALGORİTMASI

Rota birleştirme algoritması popülasyon tabanlı bir meta sezgiseldir. Algoritma her iterasyonda bilinen çözümlerden aday çözümler oluşturur ve bir sonraki iterasyona en uygun çözümleri elinde tutarak bir çözüm kümesi oluşturur. Rota birleştirme algoritmasında kaynak çözüm ve hedef çözüm olmak üzere iki başlangıç çözüm vardır. İkili yerdeğişim, ileri veya geriye kaydırma komşuluklar ile başlangıç çözüm hedef çözüme benzemektedir (Alpay, 2008). Her iterasyonda kaynak hedefe daha çok yaklaşmaktadır. Bu iterasyonlar esnasında patika üzerindeki uygun çözümler elde tutulur. Algoritma, kaynak hedefe eşitlendiğinde son bulur (Yavuz ve diğ., 2006).

Rota birleştirme algoritması ile ilgili çok sayıda makale mevcut olup aşağıda özetlenmiştir.

Aiex ve diğ. (2003) tarafından atölye tipi iş sıralama çizelgeleme probleminde tamamlanma süresinin minimizasyonuna yönelik bir çalışma yapılmıştır. Rota birleştirme ve açgözlü rastgele adaptif arama yöntemi (Grasp) algoritması kullanılmış ve algoritmanın yaklaşık çözümler elde ettiği görülmüştür. Zhang ve Lai (2006) tarafından depo yerleşim problemi için genetik algoritma ve rota birleştirme algoritması birlikte kullanılmıştır.

Yagiura ve diğ. (2006) makine atama, tedarik zinciri gibi gerçek dünya problemlerinden oluşan genelleştirilmiş atama problemi için rota birleştirme algoritmasını kullanmışlardır. Yamashita ve diğ. (2006) dağıtık arama algoritmasının proje çizelgeleme problemi çözümünde başarılı olduğunu ileri sürmektedirler. Dağıtık arama yöntemlerinin rota birleştirme algoritması ile kombine edilmesi sayesinde, komşu çözümlerden daha iyi çözümler elde edilebileceğini savunmuşlardır.

Huang ve Lim (2006) üç indeksli atama problemi için kendi sezgisel aramalarıyla genetik algoritmayı birleştirmişler ve çeşitli makalelerde yer alan sonuçlarla karşılaştırmışlardır. Gupta ve Smith (2006) sıra bağımlı hazırlık süreli tek makine problemlerinde toplam gecikmenin minimizasyonu için grasp ve komşuluk arama mekanizmalarını (değişken komşuluk arama ve rota birleştirme) kullanmışlar; tavlama benzetimi, genetik ve karınca kolonisi algoritmalarıyla karşılaştırmışlardır.

Keskin ve Uster (2007) tarafından lojistik problemlerinden olan çoklu ürün üretim/dağıtım, sistem tasarım problemi için metasezgisel yaklaşımlar ortaya konmuştur. Dağıtık arama sezgiseli rota birleştirme algoritmasıyla kombine edilerek daha kaliteli çözümler elde edilmiştir. Boudia ve diğ. (2007) tarafından üretim dağıtım problemi araştırılmıştır. Buradaki amaç üretim hazırlık stok seviyeleri ve dağıtım maliyetlerinin minimize edilmesidir. Tam sayılı doğrusal model ve büyük problemlerin çözülememesinden dolayı meta sezgisel algoritmalar kullanılmıştır. 90 rassal örnek üretilmiştir ve bu örnekler rota birleştirme ve grasp algoritmasıyla çözülmüştür. Tek başına grasp algoritması daha iyi sonuçlar vermiştir.

Jaeggi ve diğ. (2008) tarafından rota birleştirme algoritması ile tabu arama algoritması birleştirilerek çok amaçlı optimizasyon problemlerine çözüm getirilmiş ve bu çözüm bu konuda en etkin çözümler veren çok amaçlı genetik algoritma ile karşılaştırılmıştır. Valdes ve diğ. (2008)'nin çalışmasında kısmen yenilenebilir kaynaklar altında proje çizelgeleme problemi irdelenmiştir. Kısmen yenilenebilir olması bu problem tipini gerçek hayata uygun hale getirmekte ancak klasik modele uymadığından dolayı da çözümü zorlaştırmaktadır. Bu problem çözümü için grasp ve rota birleştirme algoritması birlikte uygulanmış ve sonuçlar analiz edilmiştir. Diaz ve diğ. (2008) iki kriterli montaj sıralama problemi için grasp ve rota birleştirme algoritması kullanmışlardır. Ranjbar ve diğ. (2008) tarafından minimum maliyetle, taahhüt edilen teslim tarihinde projeyi bitirebilmek amacıyla çizelgeleme yapılmıştır. Bu amaçla, genetik algoritma ve rota birleştirme algoritmaları kullanılmış, rota birleştirme algoritmasının etkin bir algoritma olduğu savunulmuştur.

Ranconi ve Henriques (2009) bloklu akış tipi çizelgeleme problemine çözüm getirmişler, toplam gecikme minimizasyonunu hedeflemişlerdir. Bu çalışmada grasp tabanlı metasezgisel önerilmiş ve rota birleştirme algoritmasıyla sonuçlar

geliştirilmiştir. Ayrıca sonuçlar NEH algoritmasıyla karşılaştırılmıştır. Jaszkiewicz ve Zielniewicz (2009) tarafından gezgin satıcı problemindeki her bağa maliyet yüklenerek iki kriterli amaç fonksiyonu için rota birleştirme algoritması ve etkili lokal aramalı memetik algoritma ortaya konulmuştur. Wang ve Tang, (2009) tek makine toplam ağırlıklı gecikme fonksiyonunun minimizasyonu problemi için populasyon tabanlı değişken komşuluk arama algoritması önermişlerdir. Bu çalışmada çeşitliliği artıracak mekanizma eklenerek klasik değişken komşuluk arama algoritmasından farklı bir yöntem oluşturulmuştur. İkinci olarak rota birleştirme algoritmasına tabu arama eklenerek yoğunlaşma artırılmıştır. Literatürdeki test problemleri üzerinde test edilerek optimal ve iyi çözümler elde edilmiştir.

Nascimento ve diğ. (2010) tarafından bağımsız birden çok işletme, birden çok süreç ve birden çok parça ve bu işletmeler arasında akışın olduğu kapasite kısıtlı sipariş büyüklüğü problemi irdelenmiş, rota birleştirme ile grasp algoritması tabanlı bir model kullanılarak çözüm elde edilmiştir. Önerilen algoritmanın diğer sezgisellere göre üstünlük sağladığı ifade edilmiştir. İstatistiksel testlerle sonuçlar analiz edilmiştir. Li ve diğ. (2010) türdeş olmayan araçlarla rotalama problemleri için, sabit ve değişken maliyetleri minimize etmeyi amaçlamışlar ve bu problem için, adaptif hafıza tabanlı bir meta sezgisel algoritma kullanmışlardır. Bu algoritmayı rota birleştirme algoritması ile birleştirerek karşılaştırmışlardır. Sonuçta rota birleştirme ile hibrit hale getirilen algoritmanın daha başarılı sonuçlar verdiği görülmüştür.

Liu ve Liu (2011)'nin çalışmasında permütasyon tipi akış tipi çizelgeleme problemindeki başlangıç çözümlerinin kalite ve farklılığının artırılması amacıyla NEH tabanlı grasp algoritması kullanılmış ve bu algoritmaya ek olarak lokal arama mekanizmaları eklenmiştir (yer değiştirme ve rota birleştirme). Bu hibrit algoritma Taillard problemlerinde denenmiş ve sonuçlar analiz edilmiştir. Villegas ve diğ. (2011)'de kamyon ve treyler rotalama problemini, rota birleştirme algoritması değişken komşu arama, grasp algoritması olmak üzere üç farklı metasezgisel yöntemi birleştirerek çözmüşlerdir. Bu hibrit yöntem çok sayıda iyi çözümler ortaya koymuştur. Usberti ve diğ., (2011) çöp toplama, posta dağıtım, okul servis aracı rotalama, sayaç okuma gibi çeşitli alanlarda çok sayıda uygulama alanı bulunan kapasite kısıtlı bağ rotalama problemi için rota birleştirme ve grasp algoritması kullanmıştır.

Nguyen ve diğ. (2012) yine lojistik problemlerinden olan iki kademeli tesis yeri rotalama problemi (LRP-2E) üzerine çalışmıştır. Bu problem tipinde iki ana depo ve çoklu ara depolar mevcuttur. Ana depolardan ara depolara, ara depolardan müşterilere ve ana depolardan direk müşteriye olmak üzere üç tür bağlantı söz konusudur. Çözüm için grasp algoritması, LP (learning Process) ve rota birleştirme algoritmalarıyla birleştirilerek test edilmiştir. Basit sezgisel algoritmalara göre daha başarılı sonuçlar elde edilmiştir. Valdes ve diğ. (2012) iki ve üç boyutlu kutulama problemi için ilk olarak grasp algoritması önermiş, çözümler elde edildikten sonra statik, dinamik ve evrimsel olmak üzere rota birleştirme algoritması üç farklı şekilde kombine edilerek çözüm aramışlardır. Nasiri ve Kianfar (2012) tarafından atölye tipi iş sıralama problemi için, rota birleştirme ve tabu aramadan oluşan hibrit bir algoritma önerilmiş, bu algoritma kıyaslama problemleri üzerinde test edilmiş ve etkinliği ortaya konmuştur. Nguyen ve diğ. (2012) tarafından iki kademeli tesis yeri rotalama problemi için tabu arama ve rota birleştirme algoritması kullanılmıştır. 1-5 ana depo, 8-20 ara depo ve 20-200 arasında değişen müşteri sayısına sahip test problemleri denenmiş, önerilen algoritmanın etkin bir algoritma olduğu ifade edilmiştir.

Ferreira ve diğ. (2012) genelleştirilmiş minimum yayılan ağaç problemi için grasp algoritmasının çok sayıda versiyonunu uygulamış ve bu algoritmaya ek olarak rota birleştirme algoritması ve yerel aramalar katıldığında daha iyi sonuçlar elde edilebileceğini savunmuşlardır. Costa ve diğ. (2012) toplam akış süresinin minimizasyonu için parçacık sürü algoritması kullanmış, arama uzayında daha iyi çözümler bulabilmek için VNS ve rota birleştirme algoritmasıyla birleştirilerek hibrit bir algoritma oluşturmuşlar. Bu algoritma Taillard'ın problemleriyle test edilmiştir.

Şekil 4.1'de minimizasyon problemi için s_i den s_g ye tek geçişin olduğu rota birleştirme algoritması detaylı bir şekilde anlatılmıştır. 1. ve 2. satırda başlangıç ve hedef çözümler belirlenmiştir. 3. ve 4. satırda s_i ve s_g arasındaki fark (Δ) hesaplanmıştır. 6.satırdan 15.satıra kadar s_i'nin s_g'ye dönüşünceye kadar olan aşamaları vardır. Her iterasyonda başlangıç çözümünde (S_i) konumlar değişerek ($s_i \oplus m$) en düşük maliyetli (iş sıralama çizelgeleme problemi için en düşük tamamlanma süresine sahip) çözüm elde edilmeye çalışılır. Bu çözüm herbir mxn ci

pozisyondaki yerdeğişimi sonucunda güncellenir ve elde edilen çözüm hafızada tutulur. Başlangıç çözüm hedef çözümle aynı olduğunda algoritma son bulur.

```
   Prosedür Rota birleştirme algoritması;
1  Si←Başlangıççözüm;
2  Sg←Hedefçözüm;
3  Δ←Farkı_Hesapla(si,sg);
4  EnküçükmaliyetçözümüPR←min{f(si),f(sg)};
5  EniyiçözümPR←argmin{f(si),f(sg)};
6  while |Δ|>1 do
7          m*← argmin{f(si ⊕ m),mϵΔ};
8          Δ ← Δ \ m*;
9          InterSol← si ⊕ m*;
10         if f(InterSol) < EnküçükmaliyetçözümüPR
11            EnküçükmaliyetçözümüPR← f(InterSol);
12            EniyiçözümPR← InterSol;
13         end if;
14            si← InterSol;
15 end_while;
16 return   EniyiçözümPR;
end
```

Şekil 4.1. Rota birleştirme algoritma kodu

Örneğin, A başlangıç çözüm, B ise hedef çözüm olarak belirlensin. İlk aşamada 1. ve 8. işler yer değiştirmiş, İkinci aşamada 2. iş ile 7. iş yerdeğiştirmiştir. Giderek başlangıç çözüm hedef çözüme benzemiştir. Son aşamada başlangıç çözüm hedef çözümle aynı olmuş ve algoritma bitmiştir. U, A çözümünün kopyası olarak tutulmaktadır. İterasyon sırasında minimum maliyetli (iş sıralama çizelgeleme problemi için tamamlanma süresi minimum olan) çözüm var ise hafızada tutulur. Bu çözüm her iterasyonda güncellenir.

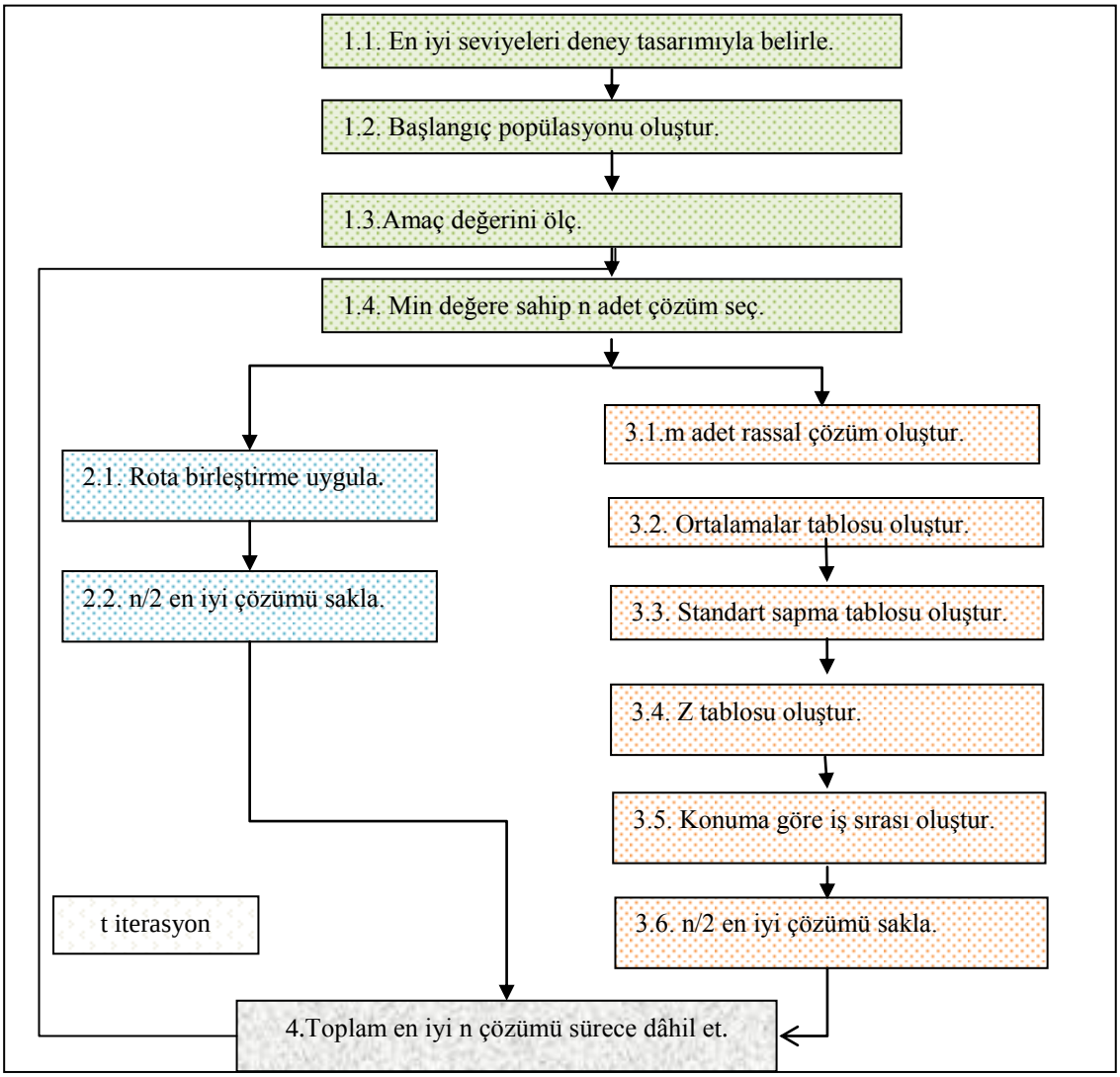
Tablo 4.1. Rota birleştirme algoritması için örnek çözüm

A: [1 2 3 4 5 6 7 8]→hedef C_{max}=1448
B: [8 7 6 5 4 3 2 1]→kaynak C_{max}=1473 → B:1473 A: 1448 U:1473
[***1*** 7 6 5 4 3 2 ***8***] Δ(U,B)=9, yerdeğiştir 1 ve 8.→ C_{max}=1420 → B:1420 A: 1448 U:1420
[8 2 6 5 4 3 7 1] Δ(U,B)=8, yerdeğiştir 2 ve 7. → C_{max}=1403* →B:1403 A: 1448 U:1403
[8 7 ***3*** 5 4 ***6*** 2 1] Δ(U,B)=7, yerdeğiştir 3 ve 6. → C_{max}=1412 →B:1412 A: 1448 U:1403
[8 7 6 ***4*** ***5*** 3 2 1] Δ(U,B)=6, yerdeğiştir 4 ve 5. → C_{max}=1415 →B:1415 A: 1448 U:1403
[*1* 2 6 5 4 3 7 *8*] Δ(U,B)=5, yerdeğiştir 1 ve 8. → C_{max}=1401* →B:1401 A: 1448 U:1401
[8 2 ***3*** 5 4 ***6*** 7 1] Δ(U,B)=4, yerdeğiştir 1 ve 8. → C_{max}=1407 →B:1407 A: 1448 U:1401
[8 2 6 ***4*** ***5*** 3 7 1] Δ(U,B)=3, yerdeğiştir 4 ve 5. → C_{max}=1409 →B:1409 A: 1448 U:1401
[1 2 ***3*** 5 4 ***6*** 7 8] Δ(U,B)=2, yerdeğiştir 3 ve 6. → C_{max}=1398 →B:1398 A: 1448 U:1398
[1 2 6 *4* *5* 3 7 8] Δ(U,B)=1, yerdeğiştir 4 ve 5. → C_{max}=1396* →B:1396 A: 1448 U:1396
[1 2 3 4 5 6 7 8] →Δ(U,B)=0 kaynak=hedef

5. AKIŞ TİPİ PROBLEMLER İÇİN MODEL ÖNERİSİ

Bu modelde, büyük ölçekli veriler içinde saklı kalmış anlamlı bilgiye ulaşmak için, sahip olunan verileri analiz ederek, yorumlamayı sağlayan Veri Madenciliği (V.M) yöntemi kullanılacaktır. Ayrıca en iyi çözümün elde edilmesi amacıyla veri madenciliği ile geliştirilen algoritmanın, rekabet edebilir bir seçenek haline gelebilmesi için rota birleştirme algoritması kullanılmış, böylece çözüm kalitesi artırılmıştır.

Bölüm 2'de veri madenciliğinin optimizasyon alanında kullanım alanları araştırılmış ve bu alanda üç temel çalışmanın mevcut olduğu görülmüştür. Literatürde yer alan çalışmaların sezgisellerden daha iyi performans göstermesine karşın meta sezgisellere göre düşük performansta oldukları ve konunun gelişmeye açık bir alan olduğu yazarlarca belirtilmiştir (Koonce, 1997; Koonce, 2000; Santos, 2006). Bu amaçla büyük boyutlu problemlere çözüm bulabilmesi ve elde edilen sonuçların diğer metasezgisellerle rekabet edebilir hale gelmesi için yeni bir model önerisi geliştirilmiştir. Model önerisi için matlab 7.0.1 programında istenilen sayıda rassal çözümler oluşturulup bu rassal iş sıralarının pozisyonlara göre ortalamaları, standart sapmaları, z değerleri hesaplanmıştır. Şekil 5.1'de önerilen model için oluşturulan algoritma gösterilmiştir.



Şekil 5.1. Model önerisinin akış şeması

Önerilen Modelin Algoritması: Önerilen model akış tipi çizelgeleme problemlerinin optimizasyonu için veri madenciliği ve rota birleştirme algoritmalarını hibrit olarak kullanmaktadır. Veri madenciliği ile çözüm uzayında global arama yapılırken, rota birleştirme algoritması ile yerel aramanın yapılması hedeflenmiştir. Modelin etkinliği Taillard'ın akış tipi çizelgeleme problemleri üzerinde test edilmiştir. Veri madenciliği yöntemi ile rassal olarak üretilen çok sayıda çözüm kullanılarak her bir işin bellirli bir pozisyona yerleştirilmesi durumunda elde edilen C_{max} değerlerinin ortalama değerleri, tüm çözümlerin C_{max} değerlerinden elde edilen ortalama ile karşılaştırılarak belirli bir işin belirli bir pozisyona yerleştirilmesinin uygunluğuna ilişkin kararlar verilmektedir. Rota birleştirme algoritması ise veri madenciliğinden gelen uygun çözümler üzerinden ilerleyerek yerel arama yapmaktadır.

1.1. En iyi Seviyeleri Deney Tasarımıyla Belirle: Önerilen modelde ilk adım deney tasarımıdır. Taillardın 20 iş 5 makine, 20 iş 10 makine, 20 iş 20 makine, 50 iş 5 makine, 50 iş 10 makine, 100 iş 10 makine problemlerinin ilkine deney tasarımı uygulanmış ve elde edilen en iyi seviyelere göre diğer problemler de çözülmüştür. Bu parametreler 1.2. ile tanımlanan adımdaki başlangıç populasyon büyüklüğü, 3.1. tanımlanan adımdaki ortalamalar tablosu büyüklüğü, 3.2. tanımlanan adımdaki konum sayısı, 2.1. tanımlanan adımdaki rota birleştirme sayısı ve iterasyon sayısı olmak üzere beş faktör belirlenmiştir. Herbir faktör için 3 seviye bulunmaktadır. L27 deseni kullanılmış böylece daha kısa süre içinde çözüm elde edilmiştir. Tablo 5.1'de deney tasarımı için belirlenen faktörler ve seviyeler gösterilmiştir. Başlangıç populasyonu, konum, rota birleştirmedeki çözüm sayısı, iterasyon sayısı her bir problem büyüklüğü için aynı seviyeler kullanılmış olup, ortalamalar tablosu için belirlenen bu seviyeler değişiklik göstermektedir. Problem büyüdükçe sıfıra bölme hatalarının oluşmaması için belirlenen bu seviyelerdeki değerler artırılmıştır.

Tablo 5.1. Deney tasarımı için belirlenen faktörler ve seviyeler

	Seviye1	**Seviye2**	**Seviye3**
Başlangıç populasyonu	500	2500	5000
Ortalamalar tablosu			
20	500	2500	5000
50	2500 (500x5)	12500	25000
100	5000 (500x10)	25000	50000
Konum	3	5	8
Rota birleştirme	50	100	150
İterasyon	10	25	50

Tablo 5.2.'de beş faktör ve üç seviye için 27 (L27) farklı kombinasyon oluşturulmuştur. Tüm kombinasyonlar için (tam faktöriyel deney) 243 adet deneme yapılması gerekirken, kısmı faktöriyel deney tasarımı ile 27 adet deney gerçekleştirilmiştir. Böylece daha kısa süre içersinde çözüm elde edilmiştir.

Deneysel sonuçlardan sinyal-gürültü (S/G) oranı hesaplanmıştır. En düşük değerleri elde edebilmek için en küçük en iyi karakteristiği kullanılmıştır. Analizler için bu karakteristiğini tanımlayan aşağıdaki Eşitlik (5.1) kullanılmıştır:

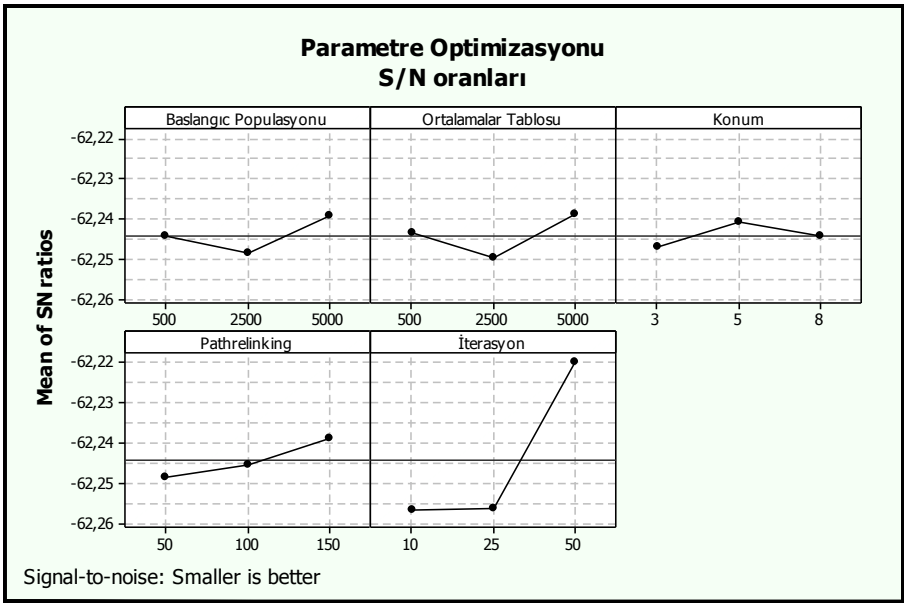
$$\eta = 10xLog10(sum(Y\text{^}2)/n) \quad (5.1)$$

Eşitlik (5.1)'de "η" en küçük en iyi durumu için S/G oranını; "y_i" deneyde elde edilen değeri; *"n"* ise tekrar sayısını ifade etmektedir. Tai 01 problemi için amaç fonksiyonu minimum tamamlanma süresi olduğu için en küçük en iyi karakteristiği kullanılarak, L27 deseni ile deney tasarımı yapılmıştır. Elde edilen sonuçlar Tablo 5.2'de gösterilmiştir. Farklı büyüklükteki diğer problemler için de benzer süreç tekrarlanmış olup parametreler ve seviyeleri, S/G oranları, Tablo 5.15, 5.18, 5.21, 5.24, 5.27, 5.30 da, parametrelerin en iyi seviyeleri ise Şekil 5.8, 5.10, 5.12, 5.14, 5.16, 5.18'de görülmektedir.

Tablo 5.2. Taguchi deney tasarımı L27 (3^5)

Deney No	Başlangıç Popülâsyonu	Ortalamalar Tablosu	Konum	Rota birleştirme	İterasyon	S/G
1	500	500	3	50	10	-62,2588
2	500	500	3	50	25	-62,2588
3	500	500	3	50	50	-62,2348
4	500	2500	5	100	10	-62,2521
5	500	2500	5	100	25	-62,2588
6	500	2500	5	100	50	-62,2335
7	500	5000	8	150	10	-62,2575
8	500	5000	8	150	25	-62,2588
9	500	5000	8	150	50	-62,1850
10	2500	500	5	150	10	-62,2588
11	2500	500	5	150	25	-62,2387
12	2500	500	5	150	50	-62,2214
13	2500	2500	8	50	10	-62,2588
14	2500	2500	8	50	25	-62,2588
15	2500	2500	8	50	50	-62,2588
16	2500	5000	3	100	10	-62,2588
17	2500	5000	3	100	25	-62,2575
18	2500	5000	3	100	50	-62,2267
19	5000	500	8	100	10	-62,2454
20	5000	500	8	100	25	-62,2588
21	5000	500	8	100	50	-62,2160
22	5000	2500	3	150	10	-62,2588
23	5000	2500	3	150	25	-62,2588
24	5000	2500	3	150	50	-62,2093
25	5000	5000	5	50	10	-62,2588
26	5000	5000	5	50	25	-62,2548
27	5000	5000	5	50	50	-62,1918

Aynı problem için en uygun faktör seviyeleri belirlenmiş ve en önemli faktörün iterasyon sayısı olduğu görülmüştür. Şekil 5.2'de beş faktör için en uygun seviyeler gösterilmiştir.



Şekil 5.2. S/G oranlarına göre parametre seviyeleri

Başlangıç populasyonu (A) III. seviye, ortalamalar tablosu (B) III. seviye, konum (C) II. seviye, rota birleştirme (D) III. seviye ve iterasyon sayısı (E) içinde III. seviye en uygun seviyeler olarak belirlenmiştir (A_3,B_3,C_2,D_3,E_3). Tai01 için izlenen süreç diğer problemlere de uygulanmıştır. Her bir problem büyüklüğü için beşer deneme yapılmıştır ve Tablo 5.3'de görüldüğü üzere en iyi faktör seviyeleri belirlenmiştir.

Tablo 5.3. Problem büyüklüklerine göre elde edilen en iyi parametre seviyeleri

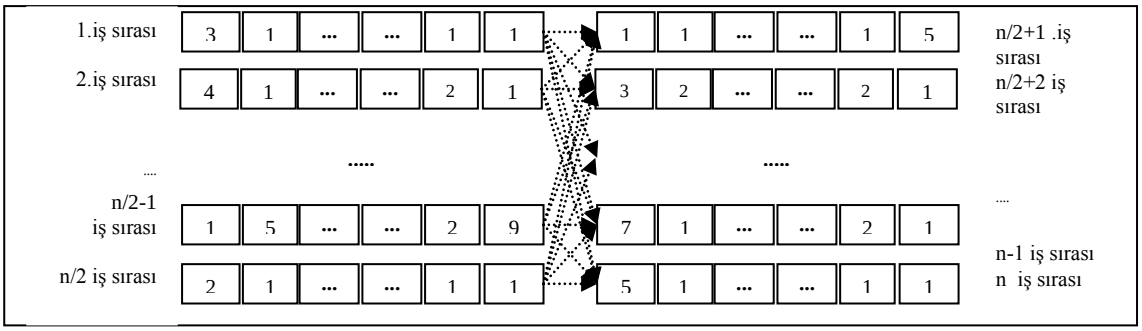
Problem Büyüklüğü	Başlangıç Populasyonu	Ortalamalar tablosu	Konum	Rota birleştirme	İterasyon	Zaman (san.)
20x5	5000	5000	5	150	50	36
20x10	2500	5000	3	150	50	52
20x20	2500	2500	3	100	50	72
50x5	500	12500	8	150	50	112
50x10	5000	12500	5	150	50	124
100x10	5000	25000	8	150	50	189

1.2. Başlangıç populasyonu oluştur: Başlangıç populasyonu belirlenen seviyelere göre rassal olarak oluşturulmuştur. Başlangıç populasyonun büyüklüğü ise akış tipi çizelgeleme probleminin boyutuna göre 20 ile 100 arasında değişmektedir.

1.3. Amaç değerini ölç: Oluşturulan başlangıç popülasyonunda yer alan iş sıralarının tamamlanma süreleri ölçülmüştür.

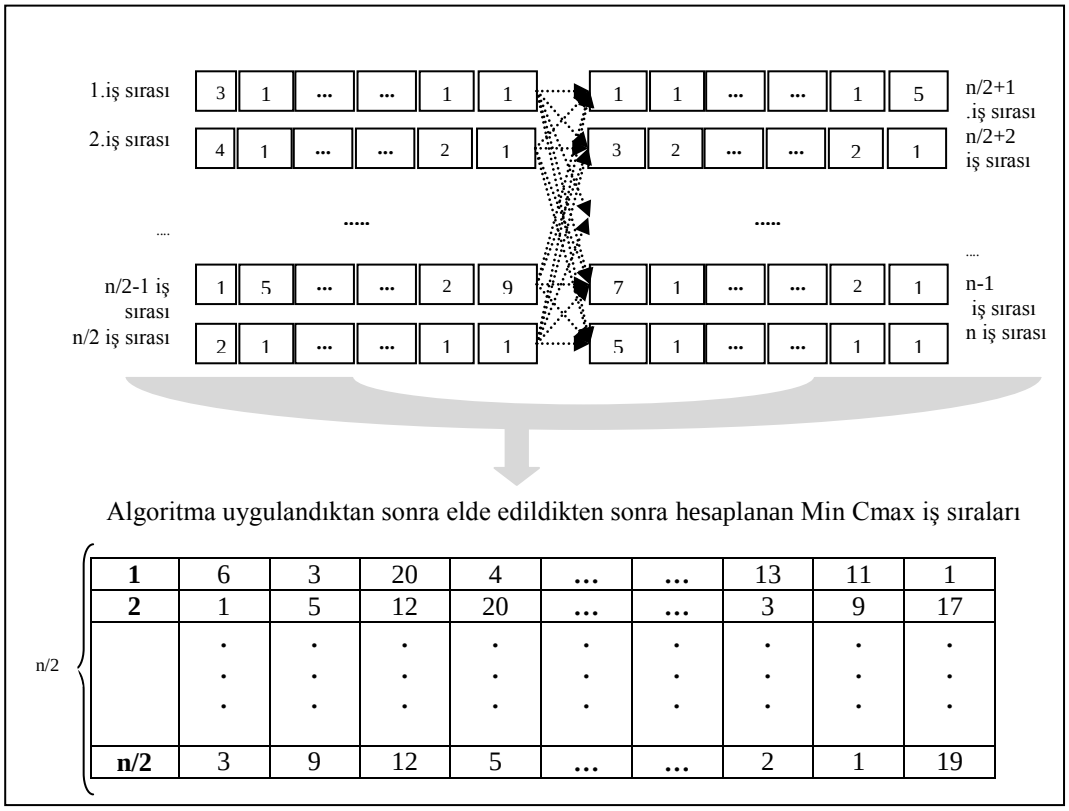
1.4. Min değere sahip n adet çözüm seç: K adet rassal çözüm oluşturulup amaç değerleri ölçüldükten sonra min C_{max}'a sahip n adet iyi çözüm seçilmiştir.

2.1. Rota birleştirme algoritmasını uygula: Daha kaliteli çözümler elde etmek amacıyla elde edilen n adet iyi çözüme rota birleştirme algoritması uygulanmıştır. n/2 çözüm kaynak çözüm, n/2 çözüm hedef çözüm olarak atanmıştır. Şekil 5.3'de rota birleştirme algoritmasının uygulanmasına ilişkin bir örnek verilmiştir.



Şekil 5.3. Rota birleştirme algoritmasının uygulanması

2.2. n/2 en iyi çözüm elde et: Rota birleştirme algoritması uygulandıktan sonra 2.1. aşamasında elde edilen en iyi n/2 çözüm veri madenciliği ile birleştirilmek üzere saklanmıştır. Şekil 5.4'de rota birleştirme algoritması uygulandıktan sonra elde edilen iş sıraları görülmektedir.



1	6	3	20	4	…	…	13	11	1
2	1	5	12	20	…	…	3	9	17
	.	.	.	.	.	.	.	.	.
	.	.	.	.	.	.	.	.	.
	.	.	.	.	.	.	.	.	.
n/2	3	9	12	5	…	…	2	1	19

n/2

Şekil 5.4. Rota birleştirme algoritma sonrası elde edilen iş sıraları

3.1. m adet rassal çözüm oluştur: m adet rassal çözüm oluşturulmuştur. Rassal çözüm sayısı problem büyüklüğüne bağlı olarak farklılık göstermektedir.

3.2. Ortalamalar tablosu oluştur: Her bir iş sırasının tamamlanma süresi belirlendikten sonra konumlara göre ortalamalar tablosu oluşturulmuştur. Her bir işin bulundukları konumlara göre tamamlanma sürelerinin ortalamaları hesaplanmıştır. Örneğin 20. işin ilk konuma gelmesi durumunda, ortalama Tablo 5.4'deki gibi hesaplanmaktadır. 2.000.000 rassal iş oluşturulmuş ve 20. işin ilk konuma gelmesi rassal 2.000.000 iş sırasından n_{20k1} tanesinde gerçekleşmiştir. Bu n_{20k1} tane rassal işin tamamlanma süreleri hesaplanmış ve bu sürelerin ortalamaları alınmıştır.

Tablo 5.4. Rassal oluşturulmuş grup ortalamasının bulunması

	Cmax	**1**	**2**	**3**	**4**	**…**	**…**	**18**	**19**	**20**
1	**1550**	20	3	5	4	**…**	**…**	13	11	1
2	1611	1	5	12	20	**…**	**…**	3	9	17
3	**1590**	20	3	4	12	**…**	**…**	11	10	5
4	1589	4	9	12	11	**…**	**…**	2	7	20
5	**1600**	20	2	1	14	**…**	**…**	5	10	9
6	**1708**	2	3	15	10	**...**	**...**	12	9	16
	. .	. .	. .	. .	. .	. .	. .	. .	. .	. .
1999.998	1594	3	9	12	5	**…**	**…**	2	1	19
1999.999	1603	19	12	20	3	**…**	**…**	7	3	11
2000.000	**1569**	20	19	1	3	**…**	**…**	2	11	17

20.işin 1.konuma gelmesi ⇐ $= (1550+1590+1600+\ldots+1569)/n_{20k1}$

	k1	**k2**	**…**	**k19**	**k20**
n1	$=(1611+.....)/n_{1k1}$	$=(.....)/n_{1k2}$	**…**	$=(1594+...)/n_{1k19}$	$=(1550+...)/n_{1k20}$
n2	$= (1708+....) / n_{2k1}$	$=1600/ n_{2k2}$	**…**	$=(.....)/n_{2k19}$	$=(.....)/n_{2k20}$
n3	$=(1594+....)/ n_{3k1}$	$=(1550+1708+...)/n_{3k2}$	**…**	$=(1603+...)/n_{3k19}$	$=(.....)/n_{3k20}$
n4	$=(1589+....)/ n_{4k1}$	$=(.....)/n_{4k2}$	**…**	$=(.....)/n_{4k19}$	$=(.....)/n4_{k20}$
	. . .	. . .	. . .	. . .	. . .
n19	$=(1603+...)/n_{19k1}$	$=(1569+...)/n_{19k2}$	**…**	$=(.....)/n_{19k19}$	$=(.....)/n_{19k20}$
n20	$=(1550+1600+...+1569)/n_{20k1}$	$=(.....)/n_{20k2}$	**…**	$=(.....)/n_{20k19}$	$=(.....)/n_{20k20}$

Bu işlemler tai01 problemi için rassal 2.000.000 çözüm için oluşturulmuş, grup ortalamaları, standart sapmaları ve z değerleri hesaplanmıştır. Tablo 5.6, Tablo 5.8 ve Tablo 5.10'de elde edilen sonuçlar gösterilmiştir.

Tablo 5.5'de ise ortalamalar tablosu için kullanılan formüller gösterilmiştir.

Tablo 5.5. Grup ortalama tablosu

	k1	k2	…	k20
n1	$\bar{C}_{n1k1} = (C_{n1k1})_1+(C_{n1k1})_2+\ldots+(C_{n1k1})_{n1k1}/n_{1k1}$	$\bar{C}_{n1k2}= (C_{n1k2})_1+(C_{n1k2})_2+\ldots+(C_{n1k2})_{n1k2}/n_{1k2}$	…	$\bar{C}_{n1k20} = (C_{n1k20})_1+(C_{n1k20})_2+\ldots+(C_{n1k1})_{n1k20}/n_{1k20}$
n2	$\bar{C}_{n2k1} =(C_{n2k1})_1+(C_{n2k1})_2+\ldots+(C_{n2k1})_{n2k1}/n_{2k1}$	$\bar{C}_{n2k2} =(C_{n2k2})_1+(C_{n2k2})_2+\ldots+(C_{n2k2})_{n1k2}/n_{2k2}$	…	$\bar{C}_{n2k20} =(C_{n2k20})_1+(C_{n2k20})_2+\ldots+(C_{n2k20})_{n2k20}/n_{2k20}$
n3	$\bar{C}_{n3k1} =(C_{n3k1})_1+(C_{n3k1})_2+\ldots+(C_{n3k1})_{n3k1}/n_{3k1}$	$\bar{C}_{n3k2} =(C_{n3k2})_1+(C_{n3k2})_2+\ldots+(C_{n3k2})_{n3k2}/n_{3k2}$	…	$\bar{C}_{n3k20} =(C_{n3k20})_1+(C_{n3k20})_2+\ldots+(C_{n3k20})_{n3k20}/n_{3k20}$
n4	$\bar{C}_{n4k1} =(C_{n4k1})_1+(C_{n4k1})_2+\ldots+(C_{n4k1})_{n4k1}/n_{4k1}$	$\bar{C}_{n4k2} =(C_{n4k2})_1+(C_{n4k2})_2+\ldots+(C_{n4k2})_{n4k2}/n_{4k2}$	…	$\bar{C}_{n4k20} =(C_{n4k20})_1+(C_{n4k20})_2+\ldots+(C_{n4k20})_{n4k20}/n_{4k20}$
	…	…	…	…
n19	$\bar{C}_{n19k1} =(C_{n19k1})_1+(C_{n19k1})_2+\ldots+(C_{n19k1})_{n19k1}/n_{19k1}$	$\bar{C}_{n19k2} =(C_{n19k2})_1+(C_{n19k2})_2+\ldots+(C_{n2k2})_{n1k2}/n_{19k2}$	…	$\bar{C}_{n19k20} =(C_{n2k20})_{19}+(C_{n2k20})_2+\ldots+(C_{n2k20})_{n19k120}/n_{19k20}$
n20	$\bar{C}_{n20k1} =(C_{n20k1})_1+(C_{n20k1})_2+\ldots+(C_{n20k1})_{n20k1}/n_{20k1}$	$\bar{C}_{n20k2} =(C_{n20k2})_1+(C_{n20k2})_2+\ldots+(C_{n20k2})_{n1k2}/n_{20k2}$	…	$\bar{C}_{n20k20} =(C_{n2k20})_1+(C_{n2k20})_2+\ldots+(C_{n2k20})_{n20k20}/n_{20k20}$

Tablo 5.6. Rassal çözüm için konumlara göre ortalamaların bulunması

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16	k17	k18	k19	k20
n1	1511,38	1512,00	1513,33	1513,59	1514,50	1514,93	1515,22	1515,96	1516,02	1516,21	1516,39	1516,47	1516,85	1517,33	1516,61	1516,51	1516,80	1516,08	1516,25	1520,52
n2	1528,70	1525,37	1523,00	1521,20	1520,01	1518,84	1517,45	1516,13	1515,02	1514,08	1513,57	1513,00	1512,00	1511,11	1510,65	1510,31	1509,77	1510,12	1509,85	1512,72
n3	1507,67	1508,19	1509,23	1510,54	1511,45	1512,95	1514,08	1515,10	1515,71	1517,38	1517,79	1519,00	1519,82	1520,05	1520,47	1520,65	1520,60	1519,58	1518,50	1514,27
n4	1514,04	1513,07	1513,63	1514,15	1514,08	1513,95	1514,36	1514,09	1514,08	1513,99	1514,08	1514,33	1514,39	1514,56	1514,92	1515,26	1516,41	1517,93	1521,64	1529,95
n5	1516,91	1514,67	1514,50	1514,74	1515,40	1516,03	1516,84	1516,78	1517,47	1517,87	1517,54	1517,59	1517,73	1517,17	1516,44	1515,34	1514,17	1512,63	1510,91	1512,24
n6	1493,99	1498,28	1501,48	1504,73	1507,25	1509,81	1512,53	1515,39	1518,02	1520,32	1522,00	1523,62	1525,11	1525,75	1526,35	1526,19	1525,32	1523,86	1519,95	1512,90
n7	1514,20	1517,09	1520,27	1521,46	1522,62	1522,69	1522,64	1521,84	1521,22	1519,83	1518,27	1517,12	1516,18	1513,70	1512,41	1510,05	1507,89	1505,37	1504,01	1504,16
n8	1507,88	1509,82	1511,33	1512,43	1513,11	1514,21	1515,14	1515,58	1516,35	1517,38	1518,11	1518,69	1519,00	1518,87	1518,74	1518,39	1518,17	1517,38	1516,24	1515,99
n9	1501,68	1501,01	1500,71	1501,42	1502,11	1502,91	1504,67	1506,07	1508,29	1510,35	1512,65	1515,26	1517,82	1520,69	1523,80	1527,38	1531,47	1535,25	1540,25	1549,29
n10	1532,50	1529,00	1526,65	1525,44	1524,48	1523,58	1522,66	1522,49	1521,46	1520,54	1519,30	1517,71	1516,17	1514,21	1511,83	1509,57	1505,79	1501,97	1495,12	1472,29
n11	1517,77	1514,40	1511,25	1508,52	1506,14	1504,73	1504,07	1504,05	1504,42	1505,34	1506,85	1508,99	1511,75	1514,46	1518,57	1522,72	1527,33	1532,62	1539,73	1549,41
n12	1545,06	1540,28	1536,21	1532,32	1528,50	1524,38	1520,13	1516,35	1522,49	1508,61	1505,10	1502,25	1499,61	1498,77	1498,59	1512,19	1502,27	1506,85	1513,44	1499,40
n13	1493,04	1504,31	1509,83	1513,02	1516,33	1518,88	1521,14	1522,69	1524,04	1524,90	1525,75	1525,53	1525,27	1524,16	1522,52	1520,31	1516,88	1512,19	1504,60	1487,18
n14	1500,04	1506,15	1509,00	1511,78	1513,14	1514,73	1515,53	1516,93	1517,70	1518,24	1518,74	1518,99	1519,27	1519,47	1519,67	1519,22	1518,69	1517,87	1517,44	1520,31
n15	1487,77	1493,88	1497,65	1501,00	1503,80	1507,34	1510,18	1512,92	1515,45	1517,53	1520,52	1522,60	1524,20	1526,50	1527,31	1528,60	1528,84	1528,66	1528,47	1530,14
n16	1529,13	1525,00	1521,42	1518,15	1515,33	1512,87	1509,99	1508,32	1506,62	1505,51	1504,82	1504,67	1505,34	1507,15	1509,56	1513,03	1517,88	1524,11	1531,67	1542,16
n17	1508,40	1506,57	1506,32	1506,25	1506,54	1506,91	1508,17	1508,75	1509,91	1511,19	1513,07	1514,73	1516,35	1518,68	1521,12	1523,37	1525,69	1528,87	1532,31	1539,79
n18	1530,17	1527,76	1527,19	1527,00	1526,81	1526,34	1525,56	1525,48	1524,39	1523,16	1521,73	1519,80	1517,11	1514,81	1511,42	1507,81	1502,67	1496,88	1488,64	1468,18
n19	1522,01	1519,48	1516,97	1515,18	1514,28	1513,19	1512,41	1511,61	1511,35	1511,13	1511,59	1511,34	1512,41	1512,83	1513,91	1515,19	1516,94	1519,29	1522,49	1529,35
n20	1550,63	1546,50	1542,92	1540,21	1537,11	1533,72	1530,24	1526,40	1523,24	1519,18	1515,01	1511,17	1506,59	1502,54	1498,08	1493,66	1489,31	1485,23	1481,61	1479,35
Genel ortalama																				1515.65

3.3. Standart sapma tablosu oluştur: z tablosunun hesaplanabilmesi için öncelikle standart sapmaların hesaplanması gerekir. Standart sapmanın hesaplanabilmesi için de grup ortalamalar tablosu kullanılmaktadır. Örneğin 1. işin 1. konumda olma durumu için, 1. işin 1. konuma geldiğinde elde edilen tamamlanma zamanlarının her biri, 1. işin 1. konuma gelen tamamlanma zamanlarının ortalamasından çıkartılır, çıkanların kareleri alınır, 1. işin 1. konuma gelme sıklığına bölünür ve karekökü alınır.

Tablo 5.7. Standart sapma tablosu (örnek)

	k1	k2	…	k20
n1	$=\sqrt{\left((1611-1511.38)+(\ldots.-1511.38)\right)^2}$ /n1k1	$=\sqrt{(\ldots-1512.00)^2}$ /n1k2	…	$=\sqrt{(1550-1520.52)+(\ldots-1520.52)}$ /n1k20
n2	$=\sqrt{\left((1708-1528.70)+(\ldots.-1528.70)\right)^2}$ /n2k1	$=\sqrt{\left((1600-1525.37)+(\ldots.-1525.37)\right)^2}$ /n2k2	…	$=\sqrt{(\ldots-1512.72)^2}$ /n2k20
n3	$=\sqrt{\left((1594-1507.67)+(\ldots.-1507.67)\right)^2}$ /n3k1	$=\sqrt{\left((1550-1508.19)+(1708-1508.19)+(\ldots-1508.19)\right)^2}$ /n3k2	…	$=\sqrt{(\ldots-1514.27)^2}$ /n3k20
n4	$=\sqrt{\left((1589-1514.04)+(\ldots.-1514.04)\right)^2}$ /n4k1	$=\sqrt{(\ldots-1513.07)^2}$ /n4k2	…	$=\sqrt{(\ldots-1529.95)^2}$ /n4k20
	. . .	. . .	. . .	. . .
n19	$=\sqrt{\left((1603-1522.01)+(\ldots.-1522.01)\right)^2}$ /n19k1	$=\sqrt{\left((1569-1519.48)+(\ldots.-1519.48)\right)^2}$ /n19k2	…	$=\sqrt{(\ldots-1529.35)^2}$ /n19k20
n20	$=\sqrt{\left((1550-1550.63)+(1600-1550.63)+(1569-1550.63)+(\ldots.-1550.63)\right)^2}$ /n20k1	$=\sqrt{(\ldots-1546.50)^2}$ /n20k2	…	$=\sqrt{(\ldots-1479.35)^2}$ /n20k20

Tablo 5.8. Standart sapma tablosu

	k1		k20
n1	$\sigma_{1k1} = \sqrt{\left(\sum_{l=1}^{n_{1k1}}\left(\left(C_{n_{1k1}}\right)_{j=1}^{n1k1} - \frac{1}{n_{1k1}}\sum_{i=1}^{n_{1k1}}(C_{1k1})_i\right)^2 . \frac{1}{n_{1k1}}\right)}$	..	$\sigma_{1k20} = \sqrt{\left(\sum_{l=1}^{n_{1k2}0}\left(\left(C_{n_{1k20}}\right)_{j=1}^{n1k20} - \frac{1}{n_{1k20}}\sum_{i=1}^{n_{1k2}}(C_{1k20})_i\right)^2 . \frac{1}{n_{1k20}}\right)}$
n2	$\sigma_{2k1} = \sqrt{\left(\sum_{l=1}^{n_{2k1}}\left(\left(C_{n_{2k1}}\right)_{j=1}^{n2k1} - \frac{1}{n_{2k1}}\sum_{i=1}^{n_{2k1}}(C_{2k1})_i\right)^2 . \frac{1}{n_{2k1}}\right)}$	..	$\sigma_{2k20} = \sqrt{\left(\sum_{l=1}^{n_{2k20}}\left(\left(C_{n_{2k20}}\right)_{j=1}^{n2k20} - \frac{1}{n_{2k20}}\sum_{i=1}^{n_{2k20}}(C_{2k20})_i\right)^2 . \frac{1}{n_{2k20}}\right)}$
n3	$\sigma_{3k1} = \sqrt{\left(\sum_{l=1}^{n_{3k1}}\left(\left(C_{n_{3k1}}\right)_{j=1}^{n3k1} - \frac{1}{n_{3k1}}\sum_{i=1}^{n_{3k1}}(C_{3k1})_i\right)^2 . \frac{1}{n_{3k1}}\right)}$	..	$\sigma_{3k20} = \sqrt{\left(\sum_{l=1}^{n_{3k20}}\left(\left(C_{n_{3k20}}\right)_{j=1}^{n3k20} - \frac{1}{n_{3k3}}\sum_{i=1}^{n_{3k3}}(C_{3k20})_i\right)^2 . \frac{1}{n_{3k20}}\right)}$
.		.	
n19	$\sigma_{19k1} = \sqrt{\left(\sum_{l=1}^{n_{19k1}}\left(\left(C_{n_{19k1}}\right)_{j=1}^{n19k1} - \frac{1}{n_{19k1}}\sum_{i=1}^{n_{19k1}}(C_{19k1})_i\right)^2 . \frac{1}{n_{19k1}}\right)}$	..	$\sigma_{19k20} = \sqrt{\left(\sum_{l=1}^{n_{19k20}}\left(\left(C_{n_{19k20}}\right)_{j=1}^{n19k20} - \frac{1}{n_{19k20}}\sum_{i=1}^{n_{20k1}}(C_{19k20})_i\right)^2 . \frac{1}{n_{19k2}}\right)}$
n20	$\sigma_{20k1} = \sqrt{\left(\sum_{l=1}^{n_{20k1}}\left(\left(C_{n_{20k1}}\right)_{j=1}^{n20k1} - \frac{1}{n_{20k1}}\sum_{i=1}^{n_{20k1}}(C_{20k1})_i\right)^2 . \frac{1}{n_{20k1}}\right)}$	..	$\sigma_{20k20} = \sqrt{\left(\sum_{l=1}^{n_{20k20}}\left(\left(C_{n_{20k20}}\right)_{j=1}^{n20k1} - \frac{1}{n_{20k20}}\sum_{i=1}^{n_{20k20}}(C_{20k20})_i\right)^2 . \frac{1}{n_{20k1}}\right)}$

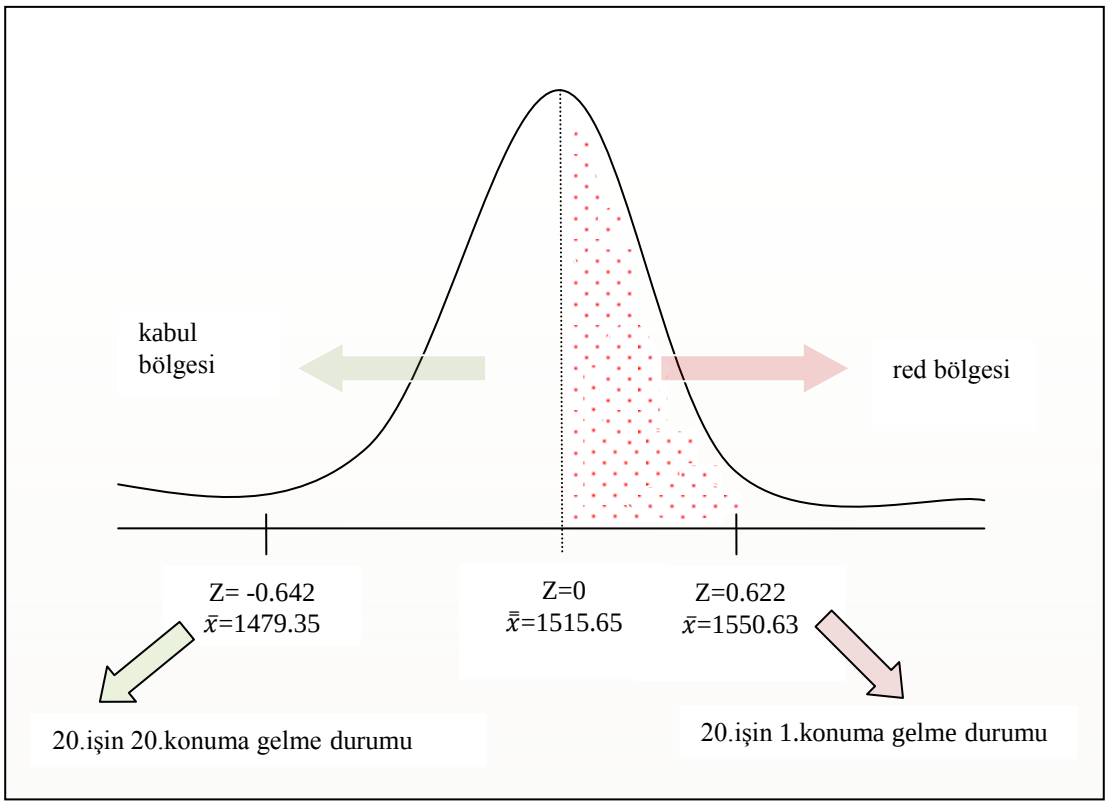
Tablo 5.9. Rassal çözüm için konumlara göre standart sapmaların bulunması

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16	k17	k18	k19	k20
n1	58,31	59,32	60,00	60,48	61,14	61,43	61,76	61,50	61,33	61,32	60,65	60,74	60,56	60,08	59,51	59,36	58,97	58,85	58,17	55,49
n2	60,70	60,50	60,05	59,66	59,52	58,99	58,75	58,93	59,17	59,40	59,72	60,04	60,52	60,93	60,99	61,17	60,89	60,35	58,81	55,74
n3	59,19	58,87	58,53	58,58	58,63	58,94	59,31	59,65	60,08	60,74	61,11	61,43	61,67	61,49	61,30	60,89	60,16	59,38	58,65	58,14
n4	57,69	59,24	60,47	61,35	61,89	62,11	62,50	62,14	61,81	61,21	60,75	60,16	59,53	59,14	58,36	58,48	58,34	58,69	58,35	54,84
n5	58,37	58,62	58,95	58,58	58,67	59,19	59,77	60,43	60,77	61,20	61,80	62,20	62,13	62,48	62,26	61,73	60,87	59,94	57,56	53,14
n6	56,57	56,92	57,51	58,17	58,57	58,95	59,68	60,24	60,96	61,30	61,46	61,77	61,83	61,49	60,99	60,52	59,58	58,44	56,66	51,00
n7	54,10	57,01	58,71	60,45	62,03	62,97	63,85	64,23	64,30	63,74	63,22	62,12	60,83	58,98	57,65	56,20	54,81	54,33	55,26	56,63
n8	58,56	59,20	59,56	60,15	60,02	60,56	60,80	60,89	61,08	61,28	60,86	60,81	60,74	60,72	60,23	59,93	59,32	59,00	58,23	56,17
n9	58,14	57,55	57,47	56,83	56,66	56,88	56,78	56,97	57,59	57,94	58,46	59,34	59,49	59,90	60,51	60,38	60,55	59,72	58,51	55,41
n10	57,64	58,34	58,11	57,59	57,17	56,95	57,00	57,14	57,51	57,67	58,17	58,63	59,13	59,79	60,61	60,89	61,17	61,01	60,12	54,26
n11	58,46	59,36	59,00	58,68	57,39	56,84	55,84	55,21	54,91	55,76	56,49	57,67	59,19	60,55	62,16	62,81	63,44	62,45	60,83	55,47
n12	57,71	58,95	59,68	60,00	60,30	59,52	58,99	58,40	59,03	56,73	55,53	54,64	54,55	54,89	56,27	57,59	59,82	61,52	61,67	57,98
n13	54,86	57,02	58,44	59,64	60,89	62,03	62,36	62,92	63,19	63,11	63,23	62,14	61,08	60,00	58,04	56,51	54,80	52,80	52,06	53,14
n14	56,84	58,41	59,87	60,67	61,56	62,36	62,63	62,83	63,01	62,33	61,96	61,34	60,51	59,34	58,42	57,58	56,82	56,34	56,25	55,52
n15	56,17	56,60	56,82	57,32	58,13	58,94	59,62	60,62	60,97	61,20	61,44	61,33	61,07	60,41	59,48	58,63	57,63	56,66	55,57	52,41
n16	60,25	60,89	61,11	60,66	60,08	59,14	58,13	57,06	56,50	55,78	55,80	55,99	56,85	57,94	59,41	60,93	62,06	62,72	61,91	58,46
n17	59,37	59,47	59,02	58,61	58,29	58,06	58,21	57,98	57,97	58,51	58,57	59,10	59,65	60,23	60,97	61,02	61,35	60,82	59,95	56,96
n18	54,14	55,57	56,84	57,76	58,39	58,85	59,34	59,90	59,82	60,21	60,24	60,02	59,67	59,32	58,92	58,03	57,31	56,45	55,98	51,54
n19	60,35	60,76	60,31	59,80	59,35	58,89	58,85	58,53	58,60	58,77	59,33	59,63	60,11	60,97	61,15	61,36	61,47	61,29	59,81	56,83
n20	56,20	57,64	57,99	58,71	58,38	58,36	57,94	57,71	56,77	56,08	55,32	54,45	53,41	52,63	52,24	52,05	52,27	53,58	55,45	56,54

3.4. Z tablosu oluştur: Z değerlerinin hesaplanması için ortalama ve standart sapmanın hesaplanması gerekir. (İşlerin ve konumların önemlerinin belirlenmesi amacıyla z değerleri hesaplanmıştır.)

z=(grup ortalaması- tüm çözümlerin ortalaması)/grup standart sapması (5.2)

z değerlerinin + değer alması o işin belirlenen konuma geldiğinde tüm çözümlerin ortalamasından daha büyük bir değer alması anlamına gelir. Bu değer arttıkça ortalamadan daha da uzaklaşılmış olur. Örneğin 20. işin ilk pozisyona gelmesi durumunda z değeri 0.622 olmaktadır. Grup ortalaması 1550.63, rassal oluşturulan iş sıralarının genel ortalaması ise 1515.65'dir. Bu durumda 20. işin 1. konuma gelmemesi gerekir (Şekil 5.5).



Şekil 5.5. Z dağılımı

Tablo 5.10. Z Tablosu

	k1	k2	…	k20
n1	$Z_{1k1} = (\bar{C}_{n1k1}\text{-}(\sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n))/\sigma_{1k1}$	$Z_{1k2} = (\bar{C}_{n1k2} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{1k2}$	. . .	$Z_{1k20} = (\bar{C}_{n1k20} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{1k20}$
n2	$Z_{2k1} = (\bar{C}_{n2k1} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{2k1}$	$Z_{2k2} = (\bar{C}_{n1k2} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{1k2}$	. . .	$Z_{2k20} = (\bar{C}_{n1k20} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{1k20}$
n3	$Z_{3k1} = (\bar{C}_{n3k1} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{3k1}$	$Z_{1k2} = (\bar{C}_{n3k2} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{1k2}$	. . .	$Z_{3k20} = (\bar{C}_{n3k20} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{1k20}$
n4	$Z_{4k1} = (\bar{C}_{n4k1} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{4k1}$	$Z_{4k2} = (\bar{C}_{n4k2} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{4k2}$	. . .	$Z_{4k20} = (\bar{C}_{n4k20} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{4k20}$
.	. .	. .	. .	. .
n19	$Z_{19k1} = (\bar{C}_{n19k1} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{19k1}$	$Z_{19k2} = (\bar{C}_{n19k2} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{19k2}$	. . .	$Z_{19k20} = (\bar{C}_{n19k20} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{19k20}$
n20	$Z_{20k1} = (\bar{C}_{n20k1} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{20k1}$	$Z_{20k2} = (\bar{C}_{n20k2} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{20k2}$	. .	$Z_{20k20} = (\bar{C}_{n20k20} - \sum_{j=1}^{20}\sum_{i=1}^{20} C_{nikj}/n)/\sigma_{20k20}$

Tablo 5.11. Rassal çözümler için Z değerinin bulunması

	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16	k17	k18	k19	k20
n1	***-0,073***	***-0,062***	***-0,039***	***-0,034***	***-0,019***	***-0,012***	***-0,007***	0,005	0,006	0,009	0,012	0,013	0,020	0,028	0,016	0,015	0,020	0,007	0,010	0,088
n2	0,215	0,161	0,122	0,093	0,073	0,054	0,031	0,008	***-0,011***	***-0,026***	***-0,035***	***-0,044***	***-0,060***	***-0,074***	***-0,082***	***-0,087***	***-0,097***	***-0,092***	***-0,099***	***-0,052***
n3	***-0,135***	***-0,127***	***-0,110***	***-0,087***	***-0,072***	***-0,046***	***-0,026***	***-0,009***	0,001	0,028	0,035	0,055	0,068	0,072	0,079	0,082	0,082	0,066	0,049	***-0,024***
n4	***-0,028***	***-0,043***	***-0,033***	***-0,024***	***-0,025***	***-0,027***	***-0,021***	***-0,025***	***-0,025***	***-0,027***	***-0,026***	***-0,022***	***-0,021***	***-0,018***	***-0,013***	***-0,007***	0,013	0,039	0,103	0,261
n5	0,022	***-0,017***	***-0,020***	***-0,016***	***-0,004***	0,007	0,020	0,019	0,030	0,036	0,031	0,031	0,033	0,024	0,013	***-0,005***	***-0,024***	***-0,050***	***-0,082***	***-0,064***
n6	***-0,383***	***-0,305***	***-0,246***	***-0,188***	***-0,143***	***-0,099***	***-0,052***	***-0,004***	0,039	0,076	0,103	0,129	0,153	0,164	0,175	0,174	0,162	0,140	0,076	***-0,054***
n7	***-0,027***	0,025	0,079	0,096	0,112	0,112	0,110	0,096	0,087	0,066	0,042	0,024	0,009	***-0,033***	***-0,056***	***-0,100***	***-0,141***	***-0,189***	***-0,211***	***-0,203***
n8	***-0,133***	***-0,099***	***-0,072***	***-0,053***	***-0,042***	***-0,024***	***-0,008***	***-0,001***	0,011	0,028	0,040	0,050	0,055	0,053	0,051	0,046	0,043	0,029	0,010	0,006
n9	***-0,240***	***-0,254***	***-0,260***	***-0,250***	***-0,239***	***-0,224***	***-0,193***	***-0,168***	***-0,128***	***-0,091***	***-0,051***	***-0,007***	0,037	0,084	0,135	0,194	0,261	0,328	0,421	0,607
n10	0,292	0,229	0,189	0,170	0,155	0,139	0,123	0,120	0,101	0,085	0,063	0,035	0,009	***-0,024***	***-0,063***	***-0,100***	***-0,161***	***-0,224***	***-0,342***	***-0,799***
n11	0,036	***-0,021***	***-0,074***	***-0,121***	***-0,166***	***-0,192***	***-0,207***	***-0,210***	***-0,204***	***-0,185***	***-0,156***	***-0,115***	***-0,066***	***-0,020***	0,047	0,113	0,184	0,272	0,396	0,609
n12	0,510	0,418	0,344	0,278	0,213	0,147	0,076	0,012	0,116	***-0,124***	***-0,190***	***-0,245***	***-0,294***	***-0,308***	***-0,303***	***-0,060***	***-0,224***	***-0,143***	***-0,036***	***-0,280***
n13	***-0,412***	***-0,199***	***-0,100***	***-0,044***	0,011	0,052	0,088	0,112	0,133	0,147	0,160	0,159	0,157	0,142	0,118	0,083	0,022	***-0,065***	***-0,212***	***-0,536***
n14	***-0,275***	***-0,163***	***-0,111***	***-0,064***	***-0,041***	***-0,015***	***-0,002***	0,020	0,033	0,042	0,050	0,054	0,060	0,064	0,069	0,062	0,054	0,039	0,032	0,084
n15	***-0,496***	***-0,385***	***-0,317***	***-0,255***	***-0,204***	***-0,141***	***-0,092***	***-0,045***	***-0,003***	0,031	0,079	0,113	0,140	0,180	0,196	0,221	0,229	0,230	0,231	0,276
n16	0,224	0,154	0,095	0,041	***-0,005***	***-0,047***	***-0,097***	***-0,128***	***-0,160***	***-0,182***	***-0,194***	***-0,196***	***-0,181***	***-0,147***	***-0,103***	***-0,043***	0,036	0,135	0,259	0,454
n17	***-0,122***	***-0,153***	***-0,158***	***-0,160***	***-0,156***	***-0,151***	***-0,129***	***-0,119***	***-0,099***	***-0,076***	***-0,044***	***-0,016***	0,012	0,050	0,090	0,126	0,164	0,217	0,278	0,424
n18	0,268	0,218	0,203	0,197	0,191	0,182	0,167	0,164	0,146	0,125	0,101	0,069	0,024	***-0,014***	***-0,072***	***-0,135***	***-0,226***	***-0,332***	***-0,482***	***-0,921***
n19	0,105	0,063	0,022	***-0,008***	***-0,023***	***-0,042***	***-0,055***	***-0,069***	***-0,073***	***-0,077***	***-0,068***	***-0,072***	***-0,054***	***-0,046***	***-0,028***	***-0,007***	0,021	0,059	0,114	0,241
n20	0,622	0,535	0,470	0,418	0,368	0,310	0,252	0,186	0,134	0,063	***-0,012***	***-0,082***	***-0,170***	***-0,249***	***-0,336***	***-0,422***	***-0,504***	***-0,568***	***-0,614***	***-0,642***

3.5. Konuma göre iş sırası oluşturma: İlk olarak işlerin önem sırası ortaya konulmaktadır. İşler standart sapma değerlerine göre sıralanır. Böylece konumlara göre tamamlanma sürelerinin değişkenlikleri ölçülür. Örneğin Tablo 5.13'deki görüldüğü üzere 20. işin diğer işlere göre, daha büyük standart sapmaya (0.4086 değeri) sahip olduğu görülmektedir. Bu durum, 20.işin konumlara göre büyük değişkenlik gösterdiğini ifade eder. Yirminci iş ilk konumda 0.622 değerini alırken, yirminci konumda -0.642 değerini almıştır. Kısaca yirminci iş ilk konuma geldiğinde, genel ortalamadan (1515.65) uzaklaşmakta, son konumlara geldiği takdirde ortalamadan daha küçük değerler almaktadır, bu yüzden yirminci iş son konumlardan birine yerleştirilmelidir.

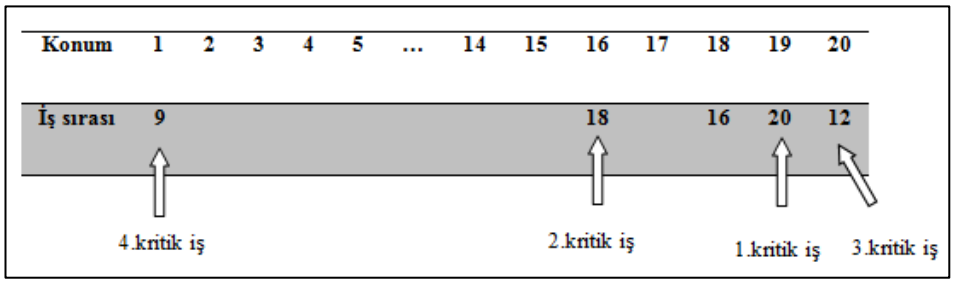
Tablo 5.12. Optimum Çözüm Alternatifleri

		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**C_{max}**
Optimum çözümler	**1**	3	17	15	8	9	6	5	14	16	7	11	13	18	19	1	4	2	10	20	12	1278
	2	9	17	15	8	3	6	14	16	5	7	11	13	19	1	18	4	2	10	20	12	1278
	3	9	17	15	8	3	6	16	14	5	7	11	13	19	1	18	4	2	10	20	12	1278
	4	9	17	15	8	3	14	6	16	5	7	11	13	19	1	18	4	2	10	20	12	1278
	5	9	17	15	8	3	14	16	6	5	7	11	13	19	1	18	4	2	10	20	12	1278
	6	9	17	15	8	3	6	14	16	5	7	11	13	1	19	18	4	2	10	20	12	1278
	7	9	17	15	8	3	6	16	14	5	7	11	13	1	19	18	4	2	10	20	12	1278
	8	9	17	15	8	3	14	6	16	5	7	11	13	1	19	18	4	2	10	20	12	1278
	9	9	17	15	8	3	14	16	6	5	7	11	13	1	19	18	4	2	10	20	12	1278
	10	3	17	15	8	9	6	14	16	5	7	11	13	19	1	18	4	2	10	20	12	1278
	11	3	17	15	8	9	6	16	14	5	7	11	13	19	1	18	4	2	10	20	12	1278
	12	3	17	15	8	9	14	6	16	5	7	11	13	19	1	18	4	2	10	20	12	1278
	13	3	17	15	8	9	14	16	6	5	7	11	13	19	1	18	4	2	10	20	12	1278
	14	3	17	15	8	9	6	14	16	5	7	11	13	1	19	18	4	2	10	20	12	1278
	15	3	17	15	8	9	6	16	14	5	7	11	13	1	19	18	4	2	10	20	12	1278
	16	3	17	15	8	9	14	6	16	5	7	11	13	1	19	18	4	2	10	20	12	1278
	17	3	17	15	8	9	6	5	14	16	7	11	13	18	19	1	4	2	10	20	12	1278
	18	3	17	15	8	9	14	16	6	5	7	11	13	1	19	18	4	2	10	20	12	1278
	19	3	8	17	15	9	6	14	16	5	7	11	13	1	19	18	4	2	10	20	12	1278
	20	9	15	17	8	3	14	6	16	5	7	11	13	19	1	18	4	2	10	20	12	1278
	21	9	15	8	17	3	14	16	6	5	7	11	13	19	1	18	4	2	10	20	12	1278
	22	9	8	17	15	3	6	14	16	5	7	11	13	1	19	18	4	2	10	20	12	1278
	23	9	17	15	8	3	6	14	16	5	7	11	13	19	1	18	4	2	10	20	12	1278
	24	9	15	17	8	3	14	6	16	5	7	11	13	19	1	18	4	2	10	20	12	1278
	25	9	15	8	17	3	14	16	6	5	7	11	13	19	1	18	4	2	10	20	12	1278
	26	9	8	17	15	3	6	14	16	5	7	11	13	1	19	18	4	2	10	20	12	1278
	27	9	8	17	15	3	6	14	16	5	7	11	13	1	19	18	4	2	10	20	12	1278
	28	3	17	15	8	9	6	16	14	5	11	13	7	19	1	18	4	2	10	20	12	1278
	29	3	17	15	8	9	14	6	16	5	13	11	7	19	1	18	4	2	10	20	12	1278
	30	3	17	15	8	9	14	6	16	11	13	5	7	1	19	18	4	2	10	20	12	1278
	31	3	17	15	8	9	14	6	16	7	11	5	13	1	19	18	4	2	10	20	12	1278

Tablo 5.12'de görüldüğü üzere Tai 01 problemi için önerilen algoritma ile çok sayıda optimum çözüm bulunmuştur ve yirminci işin sürekli son konumlarda yer aldığı görülmüştür. Aynı şekilde dokuzuncu iş, elde edilen optimum iş sıralarında ilk konumlarda ve Z tablosunda kritiklik açısından dördüncü sırada yer almaktadır.

Tüm işlerin kritiklikleri standart sapmaya göre belirlenip, sıralama yapıldıktan sonra, konumlara göre yerleştirme yapılır.

En kritik işten başlanarak deney tasarımında belirlenen seviyelere göre konumlara işler atanır. Örneğin Tablo 5.13'de görüldüğü üzere beş konum seçilmiştir. En kritik işten başlanacaktır (20. iş) ve en düşük z değerine sahip beş konumdan birine rassal bir atama yapılacaktır. Örnekte yirminci iş ondokuzuncu konuma atanmıştır ve ondokuzuncu konuma artık başka iş atanmayacaktır. İlk kritik iş tamamlandıktan sonra, ikinci kritik işe geçilecektir (18. iş). Aynı şekilde z değeri en küçük olan beş konumdan birisi rassal olarak seçilecektir. Örnekte, onsekizinci iş onaltıncı konuma atanmıştır. Bu durum standart sapma değeri en küçük olan işe kadar devam etmektedir. Şekil 5.6'da kritik iş sıralarının konumlandırılmasına ilişkin bir örnek verilmiştir.



Şekil 5.6. Kritik iş sıralarının konumlandırılması

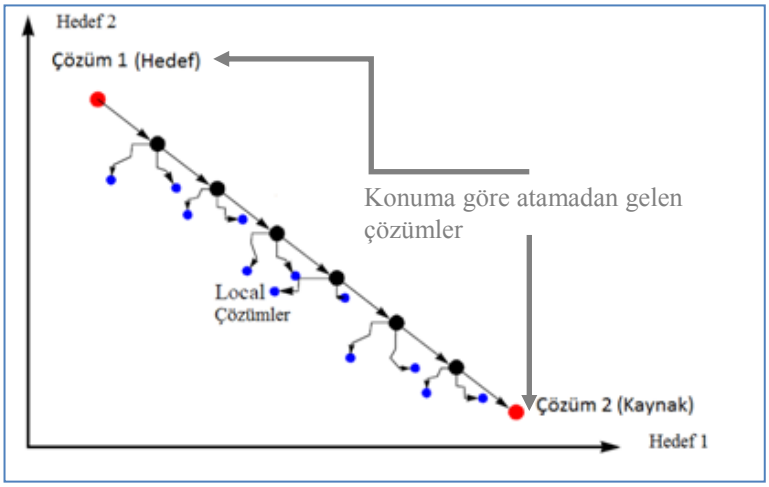
Özet olarak, en büyük standart sapmaya sahip iş ten başlayarak z değeri en küçük olan k adet konuma işler yerleştirilir. Tüm işleri kritikliklerine göre yerleştirdikten sonra elde edilen iş sırası saklanarak uygunluk değeri hesaplanır.

Tablo 5.13. Konuma göre atama

sıra	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	ss
5	0,022	*-0,017*	*-0,020*	*-0,016*	*-0,004*	0,007	0,020	0,019	0,030	0,036	0,031	0,031	0,033	0,024	0,013	*-0,005*	*-0,024*	*-0,050*	*-0,082*	*-0,064*	0,0342
1	*-0,073*	*-0,062*	*-0,039*	*-0,034*	*-0,019*	*-0,012*	*-0,007*	0,005	0,006	0,009	0,012	0,013	0,020	0,028	0,016	0,015	0,020	0,007	0,010	0,088	0,0346
8	*-0,133*	*-0,099*	*-0,072*	*-0,053*	*-0,042*	*-0,024*	*-0,008*	*-0,001*	0,011	0,028	0,040	0,050	0,055	0,053	0,051	0,046	0,043	0,029	0,010	0,006	0,0544
4	*-0,028*	*-0,043*	*-0,033*	*-0,024*	*-0,025*	*-0,027*	*-0,021*	*-0,025*	*-0,025*	*-0,027*	*-0,026*	*-0,022*	*-0,021*	*-0,018*	*-0,013*	*-0,007*	0,013	0,039	0,103	0,261	0,0690
3	*-0,135*	*-0,127*	*-0,110*	*-0,087*	*-0,072*	*-0,046*	*-0,026*	*-0,009*	0,001	0,028	0,035	0,055	0,068	0,072	0,079	0,082	0,082	0,066	0,049	*-0,024*	0,0738
19	0,105	0,063	0,022	*-0,008*	*-0,023*	*-0,042*	*-0,055*	*-0,069*	*-0,073*	*-0,077*	*-0,068*	*-0,072*	*-0,054*	*-0,046*	*-0,028*	*-0,007*	0,021	0,059	0,114	0,241	0,0821
14	*-0,275*	*-0,163*	*-0,111*	*-0,064*	*-0,041*	*-0,015*	*-0,002*	0,020	0,033	0,042	0,050	0,054	0,060	0,064	0,069	0,062	0,054	0,039	0,032	0,084	0,0911
2	0,215	0,161	0,122	0,093	0,073	0,054	0,031	0,008	*-0,011*	*-0,026*	*-0,035*	*-0,044*	*-0,060*	*-0,074*	*-0,082*	*-0,087*	*-0,097*	*-0,092*	*-0,099*	*-0,052*	0,0925
7	*-0,027*	0,025	0,079	0,096	0,112	0,112	0,110	0,096	0,087	0,066	0,042	0,024	0,009	*-0,033*	*-0,056*	*-0,100*	*-0,141*	*-0,189*	*-0,211*	*-0,203*	0,1106
17	*-0,122*	*-0,153*	*-0,158*	*-0,160*	*-0,156*	*-0,151*	*-0,129*	*-0,119*	*-0,099*	*-0,076*	*-0,044*	*-0,016*	0,012	0,050	0,090	0,126	0,164	0,217	0,278	0,424	0,1689
6	*-0,383*	*-0,305*	*-0,246*	*-0,188*	*-0,143*	*-0,099*	*-0,052*	*-0,004*	0,039	0,076	0,103	0,129	0,153	0,164	0,175	0,174	0,162	0,140	0,076	*-0,054*	0,1731
16	0,224	0,154	0,095	0,041	*-0,005*	*-0,047*	*-0,097*	*-0,128*	*-0,160*	*-0,182*	*-0,194*	*-0,196*	*-0,181*	*-0,147*	*-0,103*	*-0,043*	0,036	0,135	0,259	0,454	0,1785
13	*-0,412*	*-0,199*	*-0,100*	*-0,044*	0,011	0,052	0,088	0,112	0,133	0,147	0,160	0,159	0,157	0,142	0,118	0,083	0,022	*-0,065*	*-0,212*	*-0,536*	0,1967
11	0,036	*-0,021*	*-0,074*	*-0,121*	*-0,166*	*-0,192*	*-0,207*	*-0,210*	*-0,204*	*-0,185*	*-0,156*	*-0,115*	*-0,066*	*-0,020*	0,047	0,113	0,184	0,272	0,396	0,609	0,2226
15	*-0,496*	*-0,385*	*-0,317*	*-0,255*	*-0,204*	*-0,141*	*-0,092*	*-0,045*	*-0,003*	0,031	0,079	0,113	0,140	0,180	0,196	0,221	0,229	0,230	0,231	0,276	0,2324
10	0,292	0,229	0,189	0,170	0,155	0,139	0,123	0,120	0,101	0,085	0,063	0,035	0,009	*-0,024*	*-0,063*	*-0,100*	*-0,161*	*-0,224*	*-0,342*	*-0,799*	0,2454
9	*-0,240*	*-0,254*	*-0,260*	*-0,250*	*-0,239*	*-0,224*	*-0,193*	*-0,168*	*-0,128*	*-0,091*	*-0,051*	*-0,007*	0,037	0,084	0,135	0,194	0,261	0,328	0,421	0,607	0,2553
12	0,510	0,418	0,344	0,278	0,213	0,147	0,076	0,012	0,116	*-0,124*	*-0,190*	*-0,245*	*-0,294*	*-0,308*	*-0,303*	*-0,060*	*-0,224*	*-0,143*	*-0,036*	*-0,280*	0,2570
18	0,268	0,218	0,203	0,197	0,191	0,182	0,167	0,164	0,146	0,125	0,101	0,069	0,024	*-0,014*	*-0,072*	*-0,135*	*-0,226*	*-0,332*	*-0,482*	*-0,921*	0,2940
20	0,622	0,535	0,470	0,418	0,368	0,310	0,252	0,186	0,134	0,063	*-0,012*	*-0,082*	*-0,170*	*-0,249*	*-0,336*	*-0,422*	*-0,504*	*-0,568*	*-0,614*	*-0,642*	0,4086

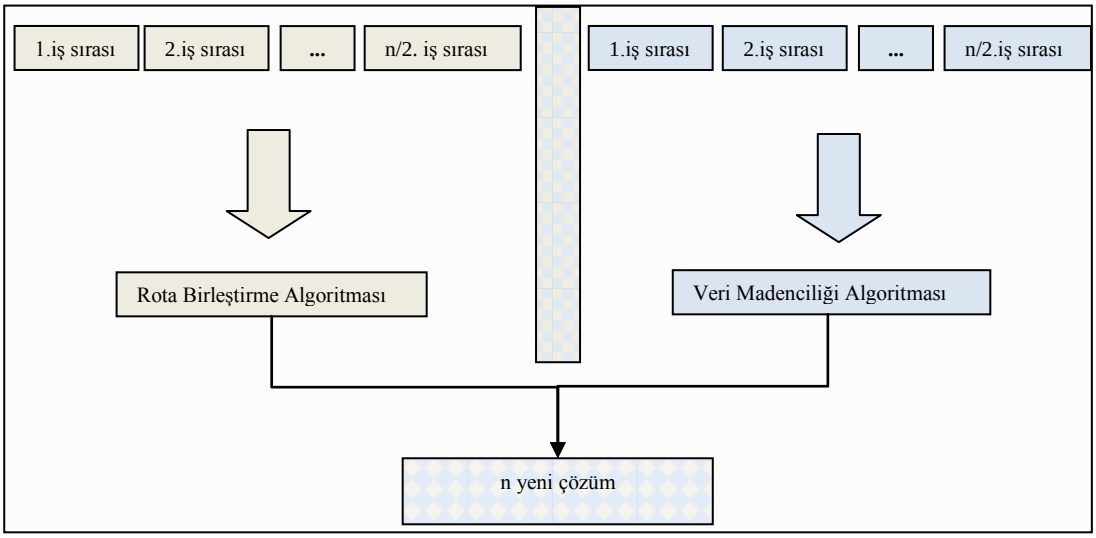
3.6. n/2 En İyi Çözüm Elde Etme: Konuma göre iş sırasından elde edilen en iyi n/2 çözüm, rota birleştirme algoritması ile birleştirmek üzere saklanır.

Rota birleştirme yöntemi lokal aramanın genelleştirilmiş halidir. Çözümler kaynak çözüm ve başlangıç çözüm ismiyle adlandırılmaktadır. Adından da açıkça anlaşılacağı üzere başlangıç çözüm yöntemin başlangıç noktasını, hedef çözüm ise ulaşılması istenen çözümü ifade etmektedir. Şekil 5.7'de rota birleştirme algoritması gösterilmektedir.



Şekil 5.7. Rota birleştirme algoritması

4.Toplam en iyi n çözümün sürece dâhil edilmesi: 3.5'de elde edilen n/2 çözüm ve 2.2.de elde edilen n/2 çözüm birleştirilerek 1.4'e tekrar gönderilir. Algoritma tekrar çevrime girer. Belirlenen iterasyon sayısı kadar döngü gerçekleşir. Şekil 5.8'de rota birleştirme ve veri madenciliği algoritmasının birleştirilmesiyle elde edilen n adet çözüm görülmektedir.



Şekil 5.8. Rota birleştirme ve veri madenciliği algoritmasının birleştirilmesi

Elde Edilen Sonuçlar: Önerilen algoritmanın performansını gözlemlemek amacıyla Taillardın akış tipi problemleri seçilmiş, iş sayısı 20 ile 100 arasında değişen problemler çözülmüştür. Her bir problem grubu için modelin parametrelerinin en uygun seviyelerinin belirlenmesi amacıyla deney tasarımı yapılmıştır. Amaç fonksiyonu minimum tamamlanma süresi olduğu için en küçük en iyi karakteristiği kullanılmıştır. Başlangıç populasyon büyüklüğü, ortalamalar tablosu büyüklüğü, konum sayısı, rota birleştirme sayısı ve iterasyon olmak üzere olmak üzere beş faktör belirlenmiştir. Herbir faktör için üç seviye bulunmaktadır. Beş faktör, üç seviyeli L27 ortagonal dizisi oluşturulmuş ve bu seviyelere göre çok sayıda farklı optimum çözüm elde edilmiştir. Her bir problem büyüklüğü için başlangıç populasyonu, konum, rota birleştirme ve iterasyon seviyeleri aynı olup, ortalamalar tablosu için problem büyüklüklükleri artıkça belirlenen seviye değerleri artmaktadır. Farklı problem büyüklükleri göz önünde tutularak elde edilen parametre seviye kombinasyonu aynı büyüklüğe sahip diğer problemler üzerinde denenmektedir.

20 iş 5 makina problemleri için çözüm performansları: 20 iş 5 makine problemleri için belirlenen seviyeler Tablo 5.14'deki gibidir. Ortalamalar tablosu için seviyeler 1. seviye için 500, 2. seviye için 2500, 3. seviye için 5000 olarak belirlenmiştir.

Tablo 5.14. Deney tasarımı için belirlenen faktörler ve seviyeler (20x5)

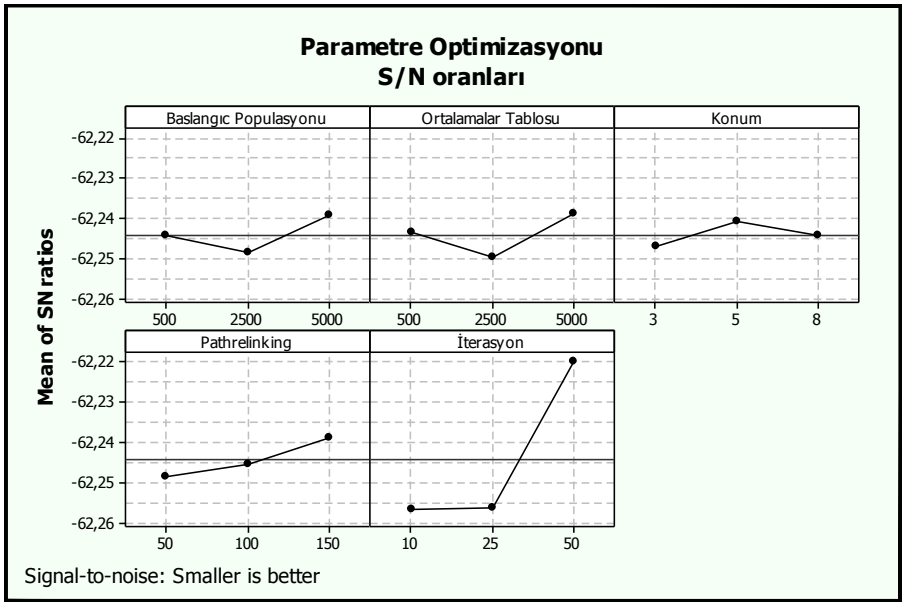
	Seviye1	**Seviye2**	**Seviye3**
Başlangıç populasyonu	500	2500	5000
Ortalamalar tablosu	500	2500	5000
Konum	3	5	8
Rota birleştirme	50	100	150
İterasyon	10	25	50

Elde edilen L27 desenine ilişkin kombinasyonlar ve sinyal gürültü değerleri Tablo 5.15'de gösterilmiştir.

Tablo 5.15. L27 (3^5) Taguchi tasarımı (20x5)

Deney No	Başlangıç Popülâsyon (A)	Ortalamalar Tablosu(B)	Konum (C)	Rota birleştirme (D)	İterasyon (E)	S/G
1	500	500	3	50	10	-62,2588
2	500	500	3	50	25	-62,2588
3	500	500	3	50	50	-62,2348
4	500	2500	5	100	10	-62,2521
5	500	2500	5	100	25	-62,2588
6	500	2500	5	100	50	-62,2335
7	500	5000	8	150	10	-62,2575
8	500	5000	8	150	25	-62,2588
9	500	5000	8	150	50	-62,1850
10	2500	500	5	150	10	-62,2588
11	2500	500	5	150	25	-62,2387
12	2500	500	5	150	50	-62,2214
13	2500	2500	8	50	10	-62,2588
14	2500	2500	8	50	25	-62,2588
15	2500	2500	8	50	50	-62,2588
16	2500	5000	3	100	10	-62,2588
17	2500	5000	3	100	25	-62,2575
18	2500	5000	3	100	50	-62,2267
19	5000	500	8	100	10	-62,2454
20	5000	500	8	100	25	-62,2588
21	5000	500	8	100	50	-62,2160
22	5000	2500	3	150	10	-62,2588
23	5000	2500	3	150	25	-62,2588
24	5000	2500	3	150	50	-62,2093
25	5000	5000	5	50	10	-62,2588
26	5000	5000	5	50	25	-62,2548
27	5000	5000	5	50	50	-62,1918

Sinyal gürültü oranlarına göre hesaplanan parametre seviye optimizasyon değerleri Şekil 5.9'da gösterilmektedir. Başlangıç populasyonu 3. seviye, ortalamalar tablosu 3. seviye, konum için 2. seviye, rota birleştirme için 3. seviye, iterasyon sayısı için 3. seviye minimum en iyi değerini vermektedir ($A_3B_3C_2D_3E_3$).



Şekil 5.9. S/G oranlarına göre parametre seviyeleri (20x5)

$A_3B_3C_2D_3E_3$ kombinasyonuna göre diğer 20x5'lik problemler için de denenmiş ve elde edilen sonuçlar Tablo 5.16'da gösterilmiştir.

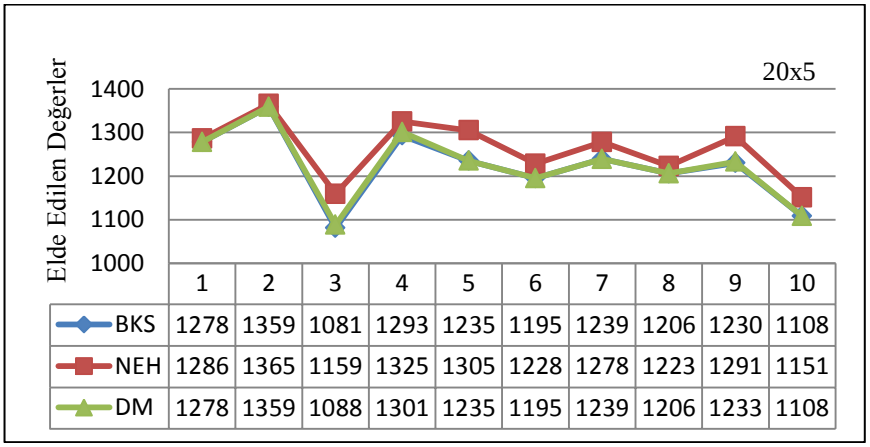
Tablo 5.16. Elde Edilen Sonuçlar (20x5)

	PS	BKS	LB	NEH	DM+PR	Sapma
1	20X5	1278	1278	1286	**1278**	**0,00%**
2	20X5	1359	1359	1365	**1359**	**0,00%**
3	20X5	1081	1081	1159	1088	0,65%
4	20X5	1293	1293	1325	1301	0,62%
5	20X5	1235	1235	1305	**1235**	**0,00%**
6	20X5	1195	1195	1228	**1195**	**0,00%**
7	20X5	1239	1239	1278	**1239**	**0,00%**
8	20X5	1206	1206	1223	**1206**	**0,00%**
9	20X5	1230	1230	1291	1233	0,24%
10	20X5	1108	1108	1151	**1108**	**0,00%**

Tablo 5.16'da görüldüğü üzere 20x5'lik 10 adet problemden yedisi için optimum sonucu bulmuştur. İlk problemde minimum tamamlanma süresi için 22 farklı iş sırasıyla optimum çözüm olan 1278 değeri bulunmuştur.

İkinci problem için 475 farklı iş sırasına sahip 1359 değerini bulmuştur. Sekizinci problem için 10 farklı kombinasyonda 1206 değerini bulmuştur. On problem için de üç farklı kombinasyon bulunmuştur. Böylece algoritma sadece tek bir optimum çözüm değil farklı optimum çözümler de bulabilmektedir.

Optimum sonucun elde edildiği yedi problemin dışındaki diğer üç problemde minimum tamamlanma süresi için elde edilen sonuçların literatürdeki en iyi problemlerle olan farkı %0,65 den düşüktür. Şekil 5.10'da NEH algoritması ile elde edilen sonuçlar ilk iki problem için optimum çözüme yakın olsa da, diğer problemlerde optimum çözümden oldukça uzaktır.



	1	2	3	4	5	6	7	8	9	10
BKS	1278	1359	1081	1293	1235	1195	1239	1206	1230	1108
NEH	1286	1365	1159	1325	1305	1228	1278	1223	1291	1151
DM	1278	1359	1088	1301	1235	1195	1239	1206	1233	1108

Şekil 5.10. Elde edilen değerlerin karşılaştırılması (20x5)

20 iş 10 makina problemleri için çözüm performansları: 20 iş 10 makine problemleri için belirlenen seviyeler Tablo 5.17'deki gibidir. Ortalamalar tablosu için 1. seviye için 500, 2. seviye için 2500, 3. seviye için 5000 olarak belirlenmiştir.

Tablo 5.17. Deney tasarımı için belirlenen faktörler ve seviyeler (20x10)

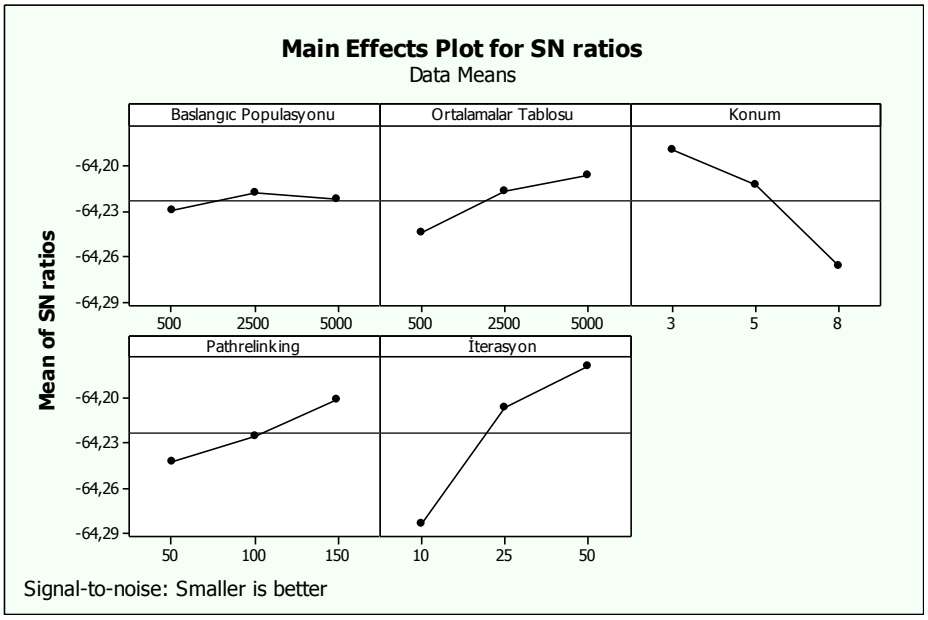
	Seviye1	**Seviye2**	**Seviye3**
Başlangıç populasyonu	500	2500	5000
Ortalamar tablosu	500	2500	5000
Konum	3	5	8
Rota birleştirme	50	100	150
İterasyon	10	25	50

Elde edilen L27 desenine ilişkin kombinasyonlar ve sinyal gürültü değerleri Tablo 5.18'de gösterilmiştir.

Tablo 5.18. L27 (3^5) Taguchi tasarımı (20x10)

Deney No	Başlangıç Popülâsyon	Ortalamalar Tablosu	Konum	Rota birleştirme	İterasyon	S/G
1	500	500	3	50	10	-64,2811
2	500	500	3	50	25	-64,2428
3	500	500	3	50	50	-64,1893
4	500	2500	5	100	10	-64,2523
5	500	2500	5	100	25	-64,2139
6	500	2500	5	100	50	-64,1817
7	500	5000	8	150	10	-64,3033
8	500	5000	8	150	25	-64,2107
9	500	5000	8	150	50	-64,1914
10	2500	500	5	150	10	-64,2811
11	2500	500	5	150	25	-64,2065
12	2500	500	5	150	50	-64,1365
13	2500	2500	8	50	10	-64,3677
14	2500	2500	8	50	25	-64,2331
15	2500	2500	8	50	50	-64,2238
16	2500	5000	3	100	10	-64,1979
17	2500	5000	3	100	25	-64,1559
18	2500	5000	3	100	50	-64,1592
19	5000	500	8	100	10	-64,3730
20	5000	500	8	100	25	-64,2814
21	5000	500	8	100	50	-64,2139
22	5000	2500	3	150	10	-64,2310
23	5000	2500	3	150	25	-64,1214
24	5000	2500	3	150	50	-64,1333
25	5000	5000	5	50	10	-64,2621
26	5000	5000	5	50	25	-64,1946
27	5000	5000	5	50	50	-64,1872

Sinyal gürültü oranlarına göre hesaplanan parametre seviye optimizasyon değerleri Şekil 5.11'de gösterilmektedir. Başlangıç populasyonu 2. seviye, ortalamalar tablosu 3. seviye, konum için 1. seviye, rota birleştirme 3. seviye, iterasyon sayısı için 3. seviye minimum en iyi değerini vermektedir ($A_2B_3C_1D_3E_3$).



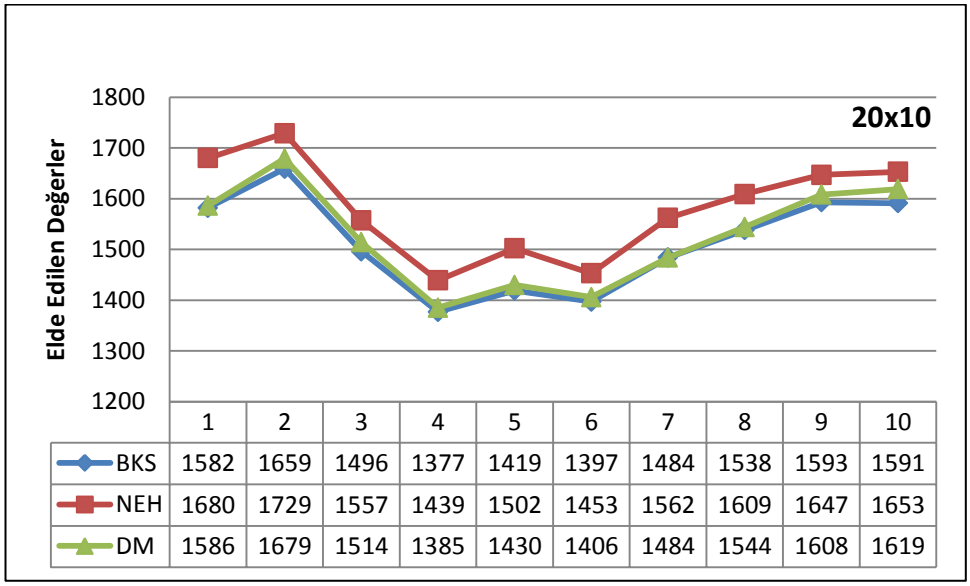
Şekil 5.11. S/G oranlarına göre parametre seviyeleri (20x10)

$A_2B_3C_1D_3E_3$ kombinasyonuna göre diğer 20x10'luk problemler için de denenmiş ve elde edilen sonuçlar Tablo 5.19'da gösterilmiştir.

Tablo 5.19. Elde Edilen Sonuçlar (20x10)

	PS	BKS	LB	NEH	DM+PR	Sapma
1	20X10	1582	1582	1680	1586	0,25%
2	20X10	1659	1659	1729	1679	1,21%
3	20X10	1496	1496	1557	1514	1,20%
4	20X10	1377	1377	1439	1385	0,58%
5	20X10	1419	1419	1502	1430	0,78%
6	20X10	1397	1397	1453	1406	0,64%
7	20X10	1484	1484	1562	**1484**	**0,00%**
8	20X10	1538	1538	1609	1544	0,39%
9	20X10	1593	1593	1647	1608	0,94%
10	20X10	1591	1591	1653	1619	1,76%

20x10 problem boyutundaki yedinci problem için optimum değer veren dört farklı sıralama elde edilmiştir. Şekil 5.12'de de görüldüğü üzere NEH algoritması tüm 20x10 problemler için optimum çözümden oldukça uzaktır.



	1	2	3	4	5	6	7	8	9	10
BKS	1582	1659	1496	1377	1419	1397	1484	1538	1593	1591
NEH	1680	1729	1557	1439	1502	1453	1562	1609	1647	1653
DM	1586	1679	1514	1385	1430	1406	1484	1544	1608	1619

Şekil 5.12. Elde edilen değerlerin karşılaştırılması (20x10)

20 iş 20 makina problemleri için çözüm performansları: 20 iş 20 makine problemleri için belirlenen seviyeler Tablo 5.20'deki gibidir. Ortalamalar tablosu için seviyeler 20x5, 20x10 problemlerde olduğu gibi, 1. seviye için 500, 2. seviye için 2500, 3. seviye için 5000 olarak belirlenmiştir.

Tablo 5.20. Deney tasarımı için belirlenen faktörler ve seviyeler (20x20)

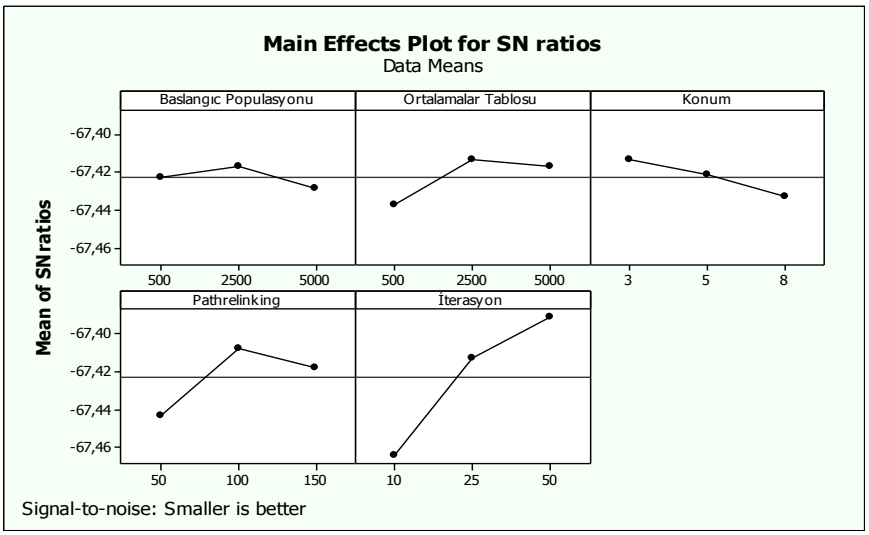
	Seviye1	**Seviye2**	**Seviye3**
Başlangıç populasyonu	500	2500	5000
Ortalamalar tablosu	500	2500	5000
Konum	3	5	8
Rota birleştirme	50	100	150
İterasyon	10	25	50

Elde edilen L27 desenine ilişkin kombinasyonlar ve sinyal gürültü değerleri Tablo 5.21'de gösterilmiştir.

Tablo 5.21. L27 (3^5) Taguchi tasarımı (20x20)

Deney No	Başlangıç Popülâsyon	Ortalamalar Tablosu	Konum	Rota birleştirme	İterasyon	S/G
1	500	500	3	50	10	-67,4775
2	500	500	3	50	25	-67,4584
3	500	500	3	50	50	-67,4112
4	500	2500	5	100	10	-67,4451
5	500	2500	5	100	25	-67,3829
6	500	2500	5	100	50	-67,3636
7	500	5000	8	150	10	-67,4730
8	500	5000	8	150	25	-67,4089
9	500	5000	8	150	50	-67,3852
10	2500	500	5	150	10	-67,4744
11	2500	500	5	150	25	-67,3985
12	2500	500	5	150	50	-67,4036
13	2500	2500	8	50	10	-67,4673
14	2500	2500	8	50	25	-67,4230
15	2500	2500	8	50	50	-67,4243
16	2500	5000	3	100	10	-67,4007
17	2500	5000	3	100	25	-67,3858
18	2500	5000	3	100	50	-67,3747
19	5000	500	8	100	10	-67,4826
20	5000	500	8	100	25	-67,4488
21	5000	500	8	100	50	-67,3829
22	5000	2500	3	150	10	-67,4664
23	5000	2500	3	150	25	-67,3747
24	5000	2500	3	150	50	-67,3756
25	5000	5000	5	50	10	-67,4914
26	5000	5000	5	50	25	-67,4347
27	5000	5000	5	50	50	-67,4015

Sinyal gürültü oranlarına göre hesaplanan parametre seviye optimizasyon değerleri Şekil 5.13'de gösterilmektedir. Başlangıç populasyonu 2. seviye, ortalamalar tablosu 2. seviye, konum için 1. seviye, rota birleştirme 2. seviye, iterasyon sayısı için 3. seviye minimum en iyi değerini vermektedir ($A_2B_2C_1D_2E_3$).



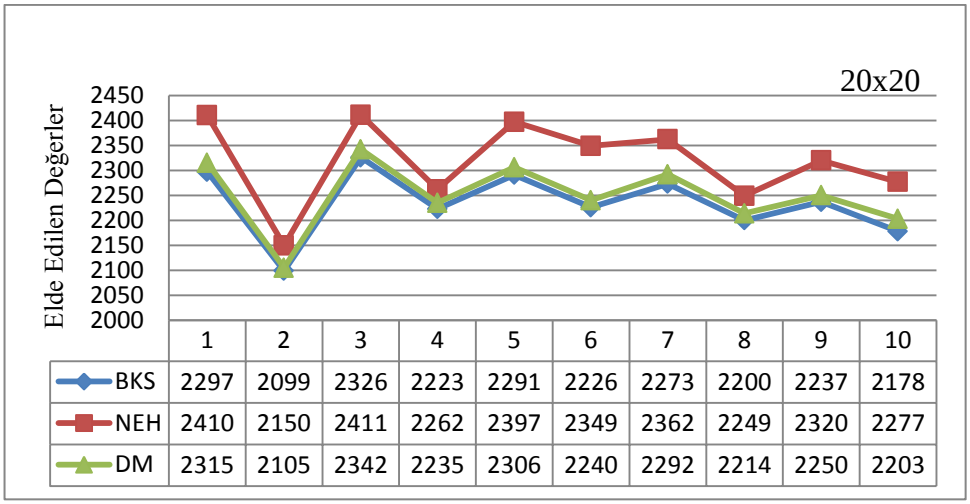
Şekil 5.13. S/G oranlarına göre parametre seviyeleri (20x20)

Tablo 5.22'de görüldüğü üzere 20x20'lik 10 adet problemden dokuzu için optimum sonuca çok yakın sonuçlar elde edilmiştir. Optimum sonuca en yakın değer %0.29, en uzak değer ise %1.15'dir.

Tablo 5.22. Elde Edilen Sonuçlar (20x20)

	PS	BKS	LB	NEH	DM+PR	Sapma
1	20X20	2297	2297	2410	2315	0,78%
2	20X20	2099	2099	2150	2105	0,29%
3	20X20	2326	2326	2411	2342	0,69%
4	20X20	2223	2223	2262	2235	0,54%
5	20X20	2291	2291	2397	2306	0,65%
6	20X20	2226	2226	2349	2240	0,63%
7	20X20	2273	2273	2362	2292	0,84%
8	20X20	2200	2200	2249	2214	0,64%
9	20X20	2237	2237	2320	2250	0,58%
10	20X20	2178	2178	2277	2203	1,15%

Şekil 5.14'de görüldüğü üzere NEH algoritması ile elde edilen çözümler optimum çözümden oldukça uzaktır.



Şekil 5.14. Elde edilen değerlerin karşılaştırılması (20x20)

50 iş 5 makine problemleri için çözüm performansları: 50 iş 5 makine problemleri için belirlenen seviyeler Tablo 5.23'deki gibidir. Ortalamalar tablosu için seviyeler 1. seviye için 2500, 2. seviye için 12500, 3. seviye için 25000 olarak belirlenmiştir.

Tablo 5.23. Deney tasarımı için belirlenen faktörler ve seviyeler (50x5)

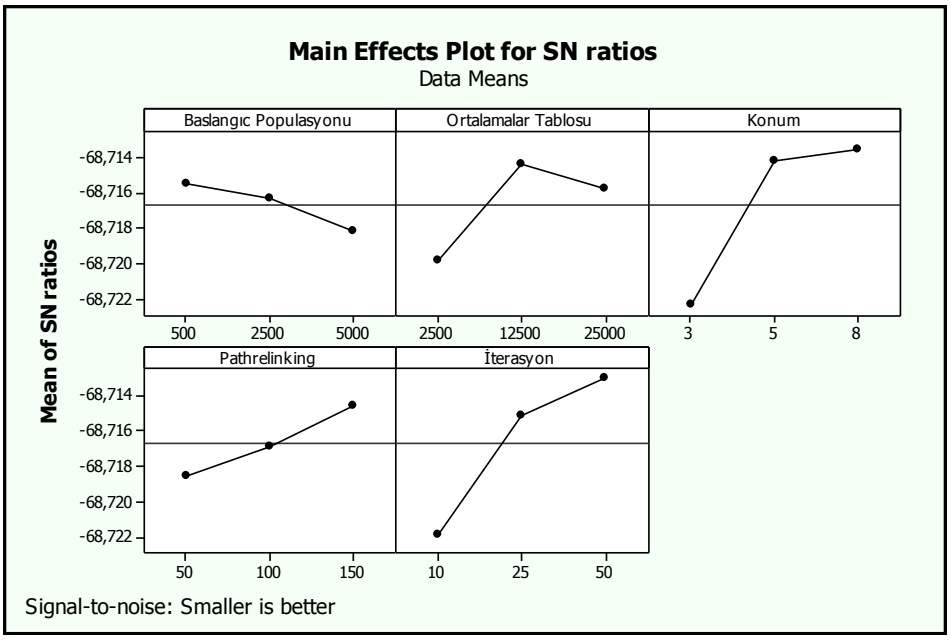
	Seviye1	**Seviye2**	**Seviye3**
Başlangıç populasyonu	500	2500	5000
Ortalamalar tablosu	2500	12500	25000
Konum	3	5	8
Rota birleştirme	50	100	150
İterasyon	10	25	50

Elde edilen L27 desenine ilişkin kombinasyonlar ve sinyal gürültü değerleri Tablo 5.24'de gösterilmiştir.

Tablo 5.24. L27 (3^5) Taguchi tasarımı (50x5)

Deney No	Başlangıç Popülâsyon	Ortalamalar Tablosu	Konum	Rota birleştirme	İterasyon	S/G
1	500	2500	3	50	500	-68,7354
2	500	2500	3	50	500	-68,7214
3	500	2500	3	50	500	-68,7220
4	500	12500	5	100	500	-68,7150
5	500	12500	5	100	500	-68,7105
6	500	12500	5	100	500	-68,7073
7	500	25000	8	150	500	-68,7169
8	500	25000	8	150	500	-68,7073
9	500	25000	8	150	500	-68,7041
10	2500	2500	5	150	2500	-68,7239
11	2500	2500	5	150	2500	-68,7105
12	2500	2500	5	150	2500	-68,7105
13	2500	12500	8	50	2500	-68,7207
14	2500	12500	8	50	2500	-68,7105
15	2500	12500	8	50	2500	-68,7073
16	2500	25000	3	100	2500	-68,7296
17	2500	25000	3	100	2500	-68,7194
18	2500	25000	3	100	2500	-68,7150
19	5000	2500	8	100	5000	-68,7175
20	5000	2500	8	100	5000	-68,7175
21	5000	2500	8	100	5000	-68,7201
22	5000	12500	3	150	5000	-68,7226
23	5000	12500	3	150	5000	-68,7182
24	5000	12500	3	150	5000	-68,7175
25	5000	25000	5	50	5000	-68,7156
26	5000	25000	5	50	5000	-68,7207
27	5000	25000	5	50	5000	-68,7137

Sinyal gürültü oranlarına göre hesaplanan parametre seviye optimizasyon değerleri Şekil 5.15'de gösterilmektedir. Başlangıç populasyonu 1. seviye, ortalamalar tablosu 2. seviye, konum için 3. seviye, rota birleştirme 3. seviye, iterasyon sayısı için 3. seviye minimum en iyi değerini vermektedir ($A_1B_2C_3D_3E_3$).



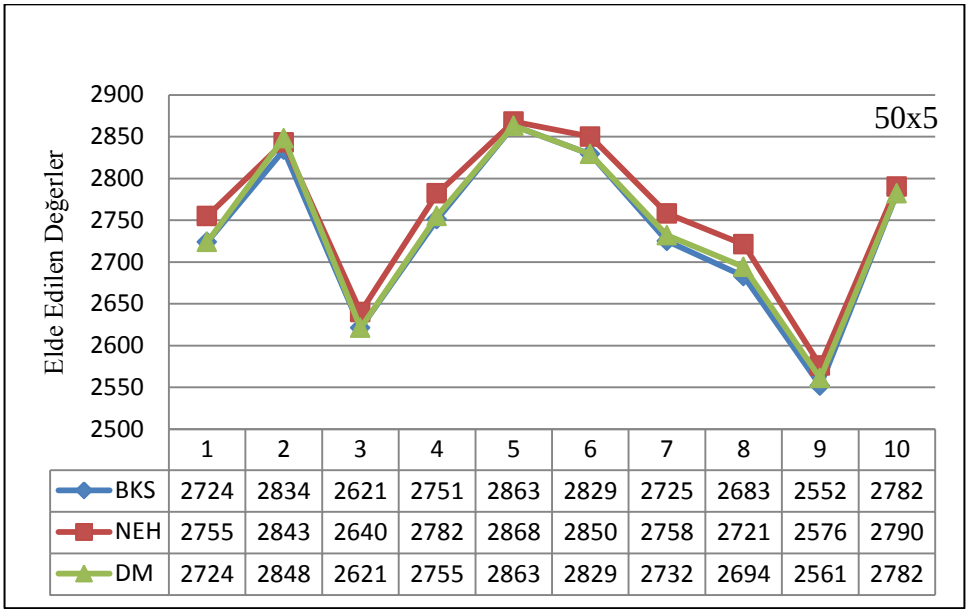
Şekil 5.15. S/G oranlarına göre parametre seviyeleri (50x5)

Tablo 5.25'de görüldüğü üzere 50x5'lik on problemin beşinde optimum sonucu bulunmuştur. İlk problem için 89 farklı iş sırası ile optimum çözüm olan 2724 değerini, son problem için ise 135 farklı iş sırası bulunmuştur. Optimum sonucun elde edildiği beş problemin dışındaki diğer problemlerde minimum tamamlanma süresi için elde edilen sonuçların literatürdeki en iyi problemlerle olan farkı %0,5 den düşüktür.

Tablo 5.25. Elde Edilen Sonuçlar (50x5)

	PS	BKS	LB	NEH	DM+PR	Sapma
1	50X5	2724	2724	2755	**2724**	**0,00%**
2	50X5	2834	2834	2843	2848	0,49%
3	50X5	2621	2621	2640	**2621**	**0,00%**
4	50X5	2751	2751	2782	2755	0,15%
5	50X5	2863	2863	2868	**2863**	**0,00%**
6	50X5	2829	2829	2850	**2829**	**0,00%**
7	50X5	2725	2725	2758	2732	0,26%
8	50X5	2683	2683	2721	2694	0,41%
9	50X5	2552	2552	2576	2561	0,35%
10	50X5	2782	2782	2790	2782	**0,00%**

Şekil 5.16'da NEH algoritması ikinci ve beşinci problem için optimum çözüme yakın olduğu görülse de, diğer problemler için optimum çözümden uzaktır.



	1	2	3	4	5	6	7	8	9	10
BKS	2724	2834	2621	2751	2863	2829	2725	2683	2552	2782
NEH	2755	2843	2640	2782	2868	2850	2758	2721	2576	2790
DM	2724	2848	2621	2755	2863	2829	2732	2694	2561	2782

Şekil 5.16. Elde edilen değerlerin karşılaştırılması (50x5)

50 iş 10 makina problemleri için çözüm performansları: 50 iş 10 makine problemleri için belirlenen seviyeler Tablo 5.26'daki gibidir. Ortalamalar tablosu için seviyeler 1. seviye için 2500, 2. seviye için 12500, 3. seviye için 25000 olarak belirlenmiştir.

Tablo 5.26. Deney tasarımı için belirlenen faktörler ve seviyeler (50x10)

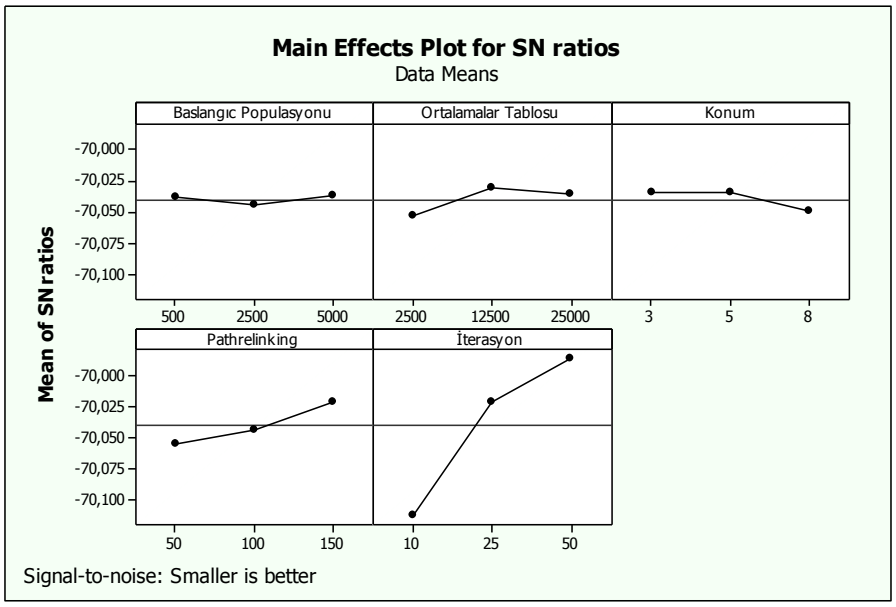
	Seviye1	**Seviye2**	**Seviye3**
Başlangıç populasyonu	500	2500	5000
Ortalamalar tablosu	2500	12500	25000
Konum	3	5	8
Rota birleştirme	50	100	150
İterasyon	10	25	50

Elde edilen L27 desenine ilişkin kombinasyonlar ve sinyal gürültü değerleri Tablo 5.27'de gösterilmiştir.

Tablo 5.27. L27 (3^5) Taguchi tasarımı (50x10)

Deney No	Başlangıç Popülâsyon	Ortalamalar Tablosu	Konum	Rota birleştirme	İterasyon	S/G
1	500	2500	3	50	10	-70,1323
2	500	2500	3	50	25	-70,0546
3	500	2500	3	50	50	-69,9999
4	500	12500	5	100	10	-70,0917
5	500	12500	5	100	25	-70,0229
6	500	12500	5	100	50	-69,9668
7	500	25000	8	150	10	-70,1133
8	500	25000	8	150	25	-69,9939
9	500	25000	8	150	50	-69,9680
10	2500	2500	5	150	10	-70,1112
11	2500	2500	5	150	25	-70,0219
12	2500	2500	5	150	50	-69,9724
13	2500	12500	8	50	10	-70,1018
14	2500	12500	8	50	25	-70,0682
15	2500	12500	8	50	50	-70,0097
16	2500	25000	3	100	10	-70,1128
17	2500	25000	3	100	25	-70,0147
18	2500	25000	3	100	50	-69,9900
19	5000	2500	8	100	10	-70,1625
20	5000	2500	8	100	25	-70,0060
21	5000	2500	8	100	50	-70,0235
22	5000	12500	3	150	10	-70,0590
23	5000	12500	3	150	25	-69,9868
24	5000	12500	3	150	50	-69,9690
25	5000	25000	5	50	10	-70,1254
26	5000	25000	5	50	25	-70,0218
27	5000	25000	5	50	50	-69,9817

Sinyal gürültü oranlarına göre hesaplanan parametre seviye optimizasyon değerleri Şekil 5.17'de gösterilmektedir. Başlangıç populasyonu 3. seviye, ortalamalar tablosu 2. seviye, konum için 2. seviye, rota birleştirme 3. seviye, iterasyon sayısı için 3. seviye minimum en iyi değerini vermektedir ($A_3B_2C_2D_3E_3$).



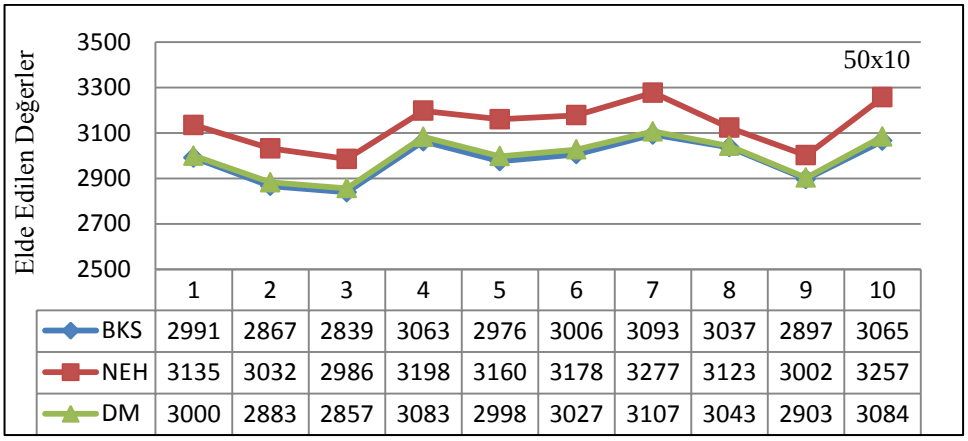
Şekil 5.17. S/G oranlarına göre parametre seviyeleri (50x10)

$A_3B_2C_2D_3E_3$ kombinasyonuna göre diğer 50x10'luk problemler için de denenmiş ve elde edilen sonuçlar Tablo 5.28'de gösterilmiştir.

Tablo 5.28. Elde Edilen Sonuçlar (50x10)

	PS	BKS	LB	NEH	DM+PR	Sapma
1	50X10	2991	2991	3135	3000	0,30%
2	50X10	2867	2867	3032	2883	0,56%
3	50X10	2839	2839	2986	2857	0,63%
4	50X10	3063	3063	3198	3083	0,65%
5	50X10	2976	2976	3160	2998	0,74%
6	50X10	3006	3006	3178	3027	0,69%
7	50X10	3093	3093	3277	3107	0,45%
8	50X10	3037	3037	3123	3043	0,20%
9	50X10	2897	2897	3002	2903	0,21%
10	50X10	3065	3065	3257	3084	0,62%

Tablo 5.28'de gözüldüğü üzere 50x10'luk tüm problemler için optimum çözüme çok yakındır. Minimum tamamlanma süresi için elde edilen sonuçların literatürdeki en iyi problemlerle olan farkı %0.20 ile %0.75 arasındadır. Şekil 5.18'de de görüldüğü üzere NEH algoritması tüm 50x10 luk problemler için optimum çözümden oldukça uzaktır.



Şekil 5.18. Elde edilen değerlerin karşılaştırılması (50x10)

100 iş 10 makine problemleri için çözüm performansları: 100 iş 10 makine problemleri için belirlenen seviyeler Tablo 5.29'daki gibidir. Ortalamalar tablosu için seviyeler 1. seviye için 5000, 2. seviye için 25000, 3. seviye için 50000 olarak belirlenmiştir.

Tablo 5.29. Deney tasarımı için belirlenen faktörler ve seviyeler (100x10)

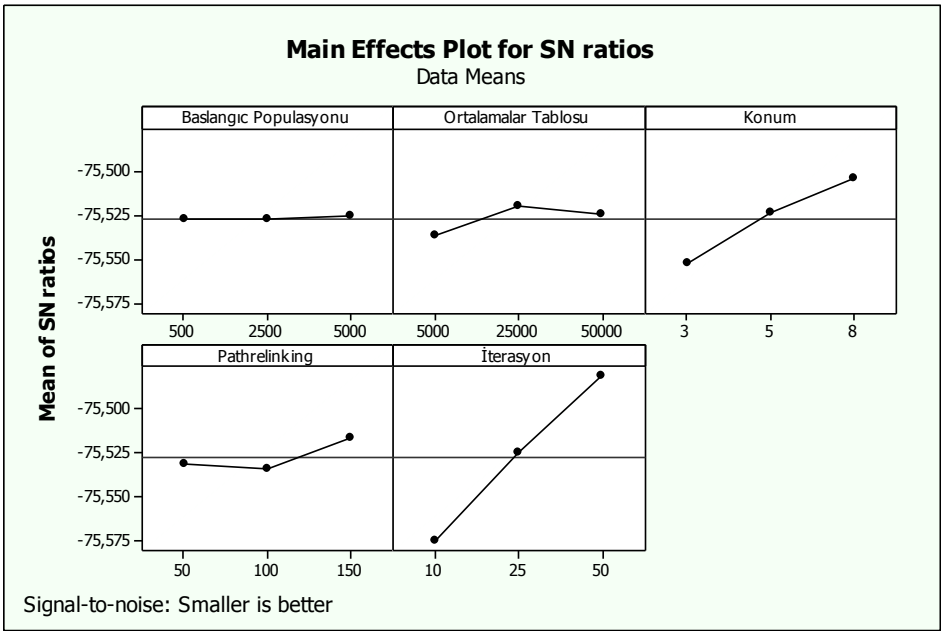
	Seviye1	**Seviye2**	**Seviye3**
Başlangıç populasyonu	500	2500	5000
Ortalamalar tablosu	5000	25000	50000
Konum	3	5	8
Rota birleştirme	50	100	150
İterasyon	10	25	50

Elde edilen L27 desenine ilişkin kombinasyonlar ve sinyal gürültü değerleri Tablo 5.30'da gösterilmiştir.

Tablo 5.30. L27 (3^5) Taguchi tasarımı (100x10)

Deney No	Başlangıç Popülâsyon	Ortalamalar Tablosu	Konum	Rota birleştirme	İterasyon	S/G
1	500	5000	3	50	10	-75,5946
2	500	5000	3	50	25	-75,5943
3	500	5000	3	50	50	-75,5143
4	500	25000	5	100	10	-75,5769
5	500	25000	5	100	25	-75,5003
6	500	25000	5	100	50	-75,4951
7	500	50000	8	150	10	-75,5442
8	500	50000	8	150	25	-75,4904
9	500	50000	8	150	50	-75,4400
10	2500	5000	5	150	10	-75,5795
11	2500	5000	5	150	25	-75,5088
12	2500	5000	5	150	50	-75,4822
13	2500	25000	8	50	10	-75,5480
14	2500	25000	8	50	25	-75,5120
15	2500	25000	8	50	50	-75,4469
16	2500	50000	3	100	10	-75,6191
17	2500	50000	3	100	25	-75,5477
18	2500	50000	3	100	50	-75,5064
19	5000	5000	8	100	10	-75,5544
20	5000	5000	8	100	25	-75,5169
21	5000	5000	8	100	50	-75,4901
22	5000	25000	3	150	10	-75,5715
23	5000	25000	3	150	25	-75,5246
24	5000	25000	3	150	50	-75,5033
25	5000	50000	5	50	10	-75,5879
26	5000	50000	5	50	25	-75,5306
27	5000	50000	5	50	50	-75,4559

Sinyal gürültü oranlarına göre hesaplanan parametre seviye optimizasyon değerleri Şekil 5.19'da gösterilmektedir. Başlangıç populasyonu 3. seviye, ortalamalar tablosu 2. seviye, konum için 3. seviye, rota birleştirme 3. seviye, iterasyon sayısı için 3. seviye minimum en iyi değerini vermektedir ($A_3B_2C_3D_3E_3$).



Şekil 5.19. S/G oranlarına göre parametre seviyeleri (100x10)

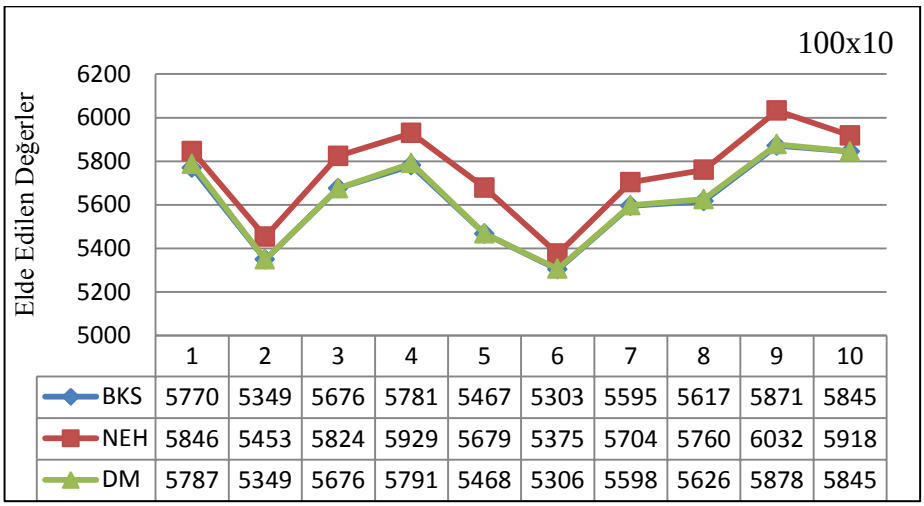
$A_3B_2C_3D_3E_3$ kombinasyonuna göre diğer 100x10'luk problemler için de denenmiş ve elde edilen sonuçlar Tablo 5.31'de gösterilmiştir.

Tablo 5.31. Elde Edilen Sonuçlar (100x10)

	PS	BKS	LB	NEH	DM+PR	Sapma
1	100X10	5770	5770	5846	5787	0,29%
2	100X10	5349	5349	5453	5349	**0,00%**
3	100X10	5676	5676	5824	5676	**0,00%**
4	100X10	5781	5781	5929	5791	0,17%
5	100X10	5467	5467	5679	5468	0,02%
6	100X10	5303	5303	5375	5306	0,06%
7	100X10	5595	5595	5704	5598	0,05%
8	100X10	5617	5617	5760	5626	0,16%
9	100X10	5871	5871	6032	5878	0,12%
10	100X10	5845	5845	5918	5845	**0,00%**

100x10'luk üç problem için optimum değer elde edilmiştir. İkinci ve üçüncü problemler için iki farklı iş sırasına sahip optimum çözüm, son problem için on farklı iş sırası ile optimum sonuç bulunmuştur. Beş problem için sapma optimum çözüme çok yakındır. En uzak sapma %3'ü geçmemektedir.

Şekil 5.20'de görüldüğü üzere NEH algoritması tüm 100x10'luk problemler için karşılaştırılamayacak kadar optimum çözümden oldukça uzaktır.



	1	2	3	4	5	6	7	8	9	10
BKS	5770	5349	5676	5781	5467	5303	5595	5617	5871	5845
NEH	5846	5453	5824	5929	5679	5375	5704	5760	6032	5918
DM	5787	5349	5676	5791	5468	5306	5598	5626	5878	5845

Şekil 5.20. Elde edilen değerlerin karşılaştırılması (100x10)

SONUÇLAR ve ÖNERİLER

Çizelgeleme problemleri NP grubunda yer alan optimizasyon problemidir. En iyi çözümleri elde edebilmek için çok sayıda farklı yaklaşım geliştirilmiştir. Bu yöntemler matematiksel tabanlı ve matematiksel tabanlı olmayan yöntemler olmak üzere; matematiksel tabanlı olmayan yöntemler de sezgisel yöntemler ve meta sezgisel yöntemler olmak üzere iki gruba ayrılmaktadır. Matematiksel tabanlı olmayan yöntemler küçük problemlerde optimum çözüm bulmalarına rağmen uzun çözüm sürelerine sahiptir. Sezgisel metotlar ise probleme özgüdürler. Meta sezgisel yöntemlerin de yerel optimum çözümlere yakınsamaları mümkün olmaktadır.

Literatürde veri madenciliğini kullanarak çizelgeleme problemlerinin optimizasyonuna yönelik üç temel çalışma yer almaktadır. Kısaca özetlemek gerekirse: Koonce D. A. ve Tsai S. C. (2000) ve Koonce D. ve diğ., (1997) çalışmalarında atölye tipi iş sıralama probleminde en iyi sırayı bulabilmek için dört faktör üzerinde durmuşlar ve bu faktörleri sınıflandırarak kurallar oluşturmuşlardır. Santos (2006) çalışmasında araç rotalama problemi gibi bilinen bir problem üzerinde gelişim algoritmalarındaki performansı artırmak için bazı alternatifler sunmuştur. Literatürde yer alan çalışmaların sezgisellerden daha iyi performans göstermesine karşın meta sezgisellere göre düşük performansta oldukları ve konunun gelişmeye açık bir alan olduğu yukarıda adı geçen yazarlarca belirtilmiştir.

Bu amaçla, çalışmada çizelgeleme problemleri içerisinde önemli bir yer tutan akış tipi çizelgeleme problemlerinin çözümüne yönelik veri madenciliği yöntemini içeren bir algoritma önerilmektedir. Elde edilen çözümlerin kalitesinin artırılması amacı ile rota birleştirme algoritması kullanılarak lokal arama yapılmıştır. Modelin etkinliği Taillard'ın akış tipi çizelgeleme problemleri üzerinde test edilmiştir.

Çalışılan problemlerde iş sayısı 20 ile 100 arasında değişmektedir, her bir problem büyüklüğü için modelin parametrelerinin en uygun seviyelerinin belirlenmesi amacıyla deney tasarımı yapılmıştır.

Bu parametreler; başlangıç populasyon büyüklüğü, ortalamalar tablosu büyüklüğü, konum sayısı, rota birleştirme sayısı ve iterasyon sayısıdır. Toplam beş faktör ve herbir faktör için de üç seviye belirlenerek L27 düzeneği oluşturulmuştur.

Taillard'ın herbiri on farklı problemden oluşan 20 iş 5 makine, 20 iş 10 makine, 20 iş 20 makine, 50 iş 5 makine, 50 iş 10 makine, 100 iş 10 makine problemleri ele alınarak, her gruptaki ilk probleme deney tasarımı uygulanmıştır. Deney tasarımı ile belirlenen en iyi seviye kombinasyonu problem büyüklüklerine göre değişikenlik gösterebilmektedir. Örneğin 20x5'lik ve 50x10'luk problemlerde konum sayısının büyüklüğü 5, 20x10'luk ve 20x20'lik problemler için konum sayısının büyüklüğü 3, 50x5'lik problemler ve 100x10'luk problemler için bu sayı sekiz olarak hesaplanmıştır. Elde edilen en iyi seviyelere göre diğer problemler de çözülmüş ve çok sayıda farklı optimum çözüm elde edilmiştir. Bu çalışmada veri madenciliği tekniği literatürdeki alışılmış biçiminin dışında kullanılmış, önemli bir matematik altyapısı gerektirmeksizin büyük boyutlu problemlerin çözümünü mümkün kılmıştır.

Önerilen model ile çözülen problemlerin %27'sinde optimum sonuç elde edilirken, %93'ünde %1'den daha küçük sapmayla çalışmıştır. Özetle, elde edilen sonuçlar sayesinde, önerilen yöntemin diğer metasezgisel yöntemlerle rekabet edebilecek düzeyde olduğu anlaşılmaktadır.

KAYNAKLAR

Aiex R., Binato S., Resende M., Parallel GRASP with path-relinking for job shop scheduling, *Parallel Computing*, 2003, **29**, 393-430.

Allahverdi A., Aldowaisan T., New heuristics to minimize total completion time in m-machine flowshops, *International Journal of Production Economics*, 2002, **77**, 71-83.

Allahverdi A., Gupta J. N. D., Aldowaisan T., A review of scheduling research involving set-up consideration, *International Journal of Management Science*, 1999, **27**, 219-239.

Alpay Ş., Tam zamanlı montaj hatlarında çok amaçlı karışık model sıralama için rota birleştirmeli açgözlü rassallaştırılmış uyarlamalı arama yordamı, *Eskişehir Osmangazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 2008, **11**, 153-170.

Andresen M., Bräsel H., Mörig M., Tusch J., Werner F., Willenius P., Simulated annealing and genetic algorithms for minimizing mean flow time in an open shop, *Mathematical and Computer Modelling*, 2008, **48**, 1279-1293.

Arabani B., Ghomi F., Zandieh M., Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage, *Expert Systems with Applications*, 2011, **38**, 1964-1979.

Boudia M., Louly M., Prins C., A reactive GRASP and path relinking for a combined production–distribution problem, *Computers & Operations Research*, 2007, **34**, 3402-3419.

Campbell H. G., Dudek R. A., Smith M. L., A heuristic algorithm form the n job, m machine sequence problem, *Management Science*, 1970, **16**, 630-637.

Cavory G., Dupas R., Goncalves G., A genetic approach to the scheduling of preventive maintenance tasks on a single product manufacturing production line, *Int. J. Production Economics*, 2001, **74**, 135-146.

Ceran G., Esnek akış tipi çizelgeleme problemlerinin veri madenciligi ve genetik algoritma kullanilarak çözülmesi, Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya, 2006

Cheng B., Chang C., A study on flowshop scheduling problem combining Taguchi experimental design and genetic algorithm, *Expert Systems with Applications*, 2007, **32**, 415-421.

Costa W., Goldbarg M., Goldbarg E., Hybridizing VNS and path-relinking on a particle swarm framework to minimize total flowtime, *Expert Systems with Applications*, 2012, **39**, 13118-13126.

Çetin M., Gezgin satıcı örnek problemlerinin optimum sonuçlarının grid aracılığı ile hesaplanması, Yüksek Lisans Tezi, Balıkesir Üniversitesi, Fen Bilimleri Enstitüsü, Balıkesir, 2007.

Çömlekçi N., *Deney tasarımı ilke ve teknikleri*, Alfa yayınları, İstanbul, 2003.

Dannenbring D. G., An evaluation of flow shop sequencing heuristics, *Management Science,* 1977, **23**, 1174-1182.

Delavar R., Keshteli M., Zavardehi M., Genetic algorithms for coordinated scheduling of production and air transportation, *Expert Systems with Applications*, 2010, **37**, 8255-8266.

Díaz B., Carbajal S., Gupta S., A path-relinking approach for a bi-criteria disassembly sequencing problem, *Computers & Operations Research*, 2008, **35**, 3989-3997.

Ferreira C., Ochi L., Parada V., Uchoa E., A GRASP-based approach to the generalized minimum spanning tree problem, *Expert Systems with Applications*, 2012, **39**, 3526-3536.

Framinan J. M., Leisten R., Usano R., Comparison of heuristics for minimisation in permutation flowshops, *Computers and Operations Research,* 2005, **32**, 1237-1254.

Goebel M., Gruenwald, L., A Survey of Data Mining and Knowledge Discovery Software Tools, *Association for Computing Machinery Explorations Newsletter*, 1999, **1**, 20-33.

Gönülol S., Gezgin satıcı problemi için veri madenciliği tabanli sezgisel bir yaklaşim, Yüksek Lisans Tezi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Kocaeli, 2009.

Grabowski J., Wodecki M., A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion, *Computers and Operational Research*, 2004, **31**, 1891-1909.

Gupta R., Smith J., Algorithms for single machine total tardiness scheduling with sequence dependent setups, *European Journal of Operational Research,* 2006, **175**, 722-739.

Akpınar H., Veri tabanlarında bilgi keşfi ve veri madenciliği, *İstanbul Üniversitesi İşletme Fakültesi Dergisi*, 2000, **29**, 1-22.

Ho J. C., Flowshop sequencing with mean flow time objective, *European Journal of Operational Research*, 1995, **81**, 571-578.

Huang G., Lim A., A hybrid genetic algorithm for the three index assignment problem, *European Journal of Operational Research*, 2006, **172**, 249-257.

Hui S., Jha G., Application data mining for customer service support, *Information and Management*, 2000, **38**, 1-13.

Ishubuchi M., Masaki S., Tanaka H., Modified simulated annealing for the flow shop sequencing problems, *European Journal of Operational Research,* 1995, **81**, 388-398.

Jaeggi D., Parks G., Kipouros T., Clarkson P., The development of a multi-objective Tabu Search algorithm for continuous optimisation problems, *European Journal of Operational Research*, 2008, **185**, 1192-1212.

Jaszkiewicz A., Zielniewicz P., Pareto memetic algorithm with path relinking for bi-objective traveling salesperson problem, *European Journal of Operational Research,* 2009, **193**, 885-890.

Jiju A., Preece D., *Understanding, managing and implementing quality frameworks, techniques and cases*, *routledge*, Taylor&Francis Group, London, 2002.

Keskin B., Uster H., Meta-heuristic approaches with memory and evolution for a multi-product production distribution system design problem, *European Journal of Operational Research*, 2007, **182**, 663-682.

Koonce D., Fang C. H. and Shi-Chi Tsai S. C., A data mining tool for learning from manufacturıng systems, *Computers Industrial Engineering,* 1997, **33,** 27-30.

Koonce D. A, Tsai S. C., Using data mining to find patterns in genetic algorithm solutions to a job shop Schedule, *Computers & Industrial Engineering,* 2000, **38**, 361-374.

Krishnaiah K., Shahabudeen P., *Applied design of experiments and Taguchi metods*, Learning Private Limited, New Delhi, 2012.

Kumar S., Rao C. S. P, Application of antcolony, genetic algorithm and data mining-based techniques for scheduling, *Robotics and Computer-Integrated Manufacturing*, 2009, **25**, 901-908.

Li X., Tian P., Aneja Y., An adaptive memory programming metaheuristic for the heterogeneous fixed fleet vehicle routing problem, *Transportation Research Part E*, 2010, **46**, 1111-1127.

Liu J., Reeves C. R., Constructive and composite heuristic solutions to the P==PCi scheduling problem, *European Journal of Operational Research*, 2001, **132**, 439-452.

Liu Y. F, Liu S. Y., A hybrid discrete artificial bee colony algorithm for permutation flowshop scheduling problem, *Applied Soft Computing*, 2011, DOI: 10.1016/j.asoc.2011.10.024.

Louviere J., Hensher D., Swait D., *Stated choice methods: Analysis and application*, United Kingdom University Press, Cambridge, 2000.

Muth J. F., Thompson G. L, *Industrial scheduling*, Prentice Hall., New Jersey, 1963.

Naderi B., Zandieh M., Balagh K., Roshanaei V., An improved simulated annealing for hybrid flowshops with sequence-dependent setup and transportation times to minimize total completion time and total tardiness, *Expert Systems with Applications*, 2009, **36**, 9625-9633.

Naderi B., Ghomia F., Aminnayeri M., A high performing metaheuristic for job shop scheduling with sequence-dependent setup times, *Applied Soft Computing*, 2010, **10**, 703-710.

Naderi B., Ghomia F., Aminnayeri M., Zandieh M., Scheduling open shops with parallel machines to minimize total completion time, *Journal of Computational and Applied Mathematics*, 2011, **235**, 1275-1287.

Naderi B., Ghomia F., Aminnayeri M., Zandieh M., Scheduling open shops with parallel machines to minimize total completion time, *Expert Systems with Applications*, 2010, **37**, 5951-5959.

Nascimento M., Resende M., Toledo F., GRASP heuristic with path-relinking for the multi-plant capacitated lot sizing problem, *European Journal of Operational Research*, 2010, **200**, 747-754.

Nasiri M., Kianfar F. A., GES/TS algorithm for the job shop scheduling, *Computers & Industrial Engineering*, 2012, **62**, 946-952.

Nawaz M., Enscore E., Ham I., A heuristic algorithm for the m-Machine, n-Job flowshop sequencing problem, *International Journal of Management Science*, 1983, **11**, 91-95.

Nguyen P., Prins C., Prodhon C., Solving the two-echelon location routing problem by a GRASP reinforced by a learning process and path relinking, *European Journal of Operational Research*, 2012, **216**, 113-126.

Nguyen V., Prins C., Prodhon C., A multi-start iterated local search with tabu list and rota birleştirme for the two-echelon location-routing problem, *Engineering Applications of Artificial Intelligence*, 2012, **25**, 56-71.

Nowicki E., Smutnicki C., A fast tabu search algorithm for the permutation flow-shop problem, *European Journal of Operational Research*, 1996, **91**, 160-175.

Ogub F. A., Simith D. K., Simulated annealing for the permutation flowshop problem, *International Journal of Management Science*, 1990, **19**, 64-67.

Oğuz C., Ercan F., A Genetic algorithm for hybrid flow shop scheduling with multiprocessor tasks, *Journal of Scheduling*, 2005, **8**, 323-351.

Olafsson S., Li X., Learning effective new single machine dispatching rules from optimal scheduling data, *Int. J. Production Economics*, 2010, **128**, 118-126.

Ozcelik B., Ozbay A., Demirbas E., Influence of injection parameters and mold materials on mechanical properties of ABS in plastic injection molding, *International Communications in Heat and Mass Transfer*, 2010, **37**, 1359-1365.

Öztürk R. E., Esnek akiş tipi ve çok işlemcili esnek akiş tipi çizelgeleme problemlerinin paralel doyumsuz algoritma ile çözümü, Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstütüsü, Konya, 2007.

Palmer D., Sequencing jobs through a multi-stage process in the minimumtotal time a quick method of obtaining a near optimum. *Operations Research Quarterly*, 1965, **16**, 101-107.

Pinedo M., *Scheduling theory, algortihms and systems*, 2th ed., Prentice Hall, New York, 2002.

Qiana B., Wang L., Huang D., Wang W.L., Wang X., An effective hybrid DE-based algorithm for multi objective flow shop scheduling with limited buffers, *Computers & Operations Research*, 2009, **36**, 209-233.

Rajendran C., Ziegler H., An efficient heuristic for scheduling in a flowshop to minimize total weighted flowtime of jobs, *European Journal of Operational Research*, 1997, **103**, 129-138.

Rajendran C., Ziegler H., Ant-colony algorithms for permutation flowshop scheduling to minimize makespan total flowtime of jobs, *European Journal of Operational Research*, 2004, **155**, 426-438.

Rajendran, C., Ziegler, H., Two ant-colony algorithms for minimizing total flowtime in permutation flowshops, *Computer and Industrial Engineering*, 2005, **48**, 789-797.

Ranjbar M., Kianfar F., Shadrokh S., Solving the resource availability cost problem in project scheduling by path relinking and genetic algorithm, *Applied Mathematics and Computation*, 2008, **196**, 879-888.

Reeves C. R., A genetic algorithm for flowshop sequencing, *Computers and Operational Research*, 1995, **22**, 5-13.

Reeves C. R., Yamada, T., Genetic algorithm, path relinking and the flowshop sequencing problem, *Evolutionary Computation*, 1998, **6**, 45-60.

Ronconi D., Henriques L., Some heuristic algorithms for total tardiness minimization in a flowshop with blocking, *International Journal of Management Science*, 2009, **37**, 272-281.

Sadjadi S., Soltani R., Izadkhah M., Saberian F., Darayi F., A new nonlinear stochastic staff scheduling model, *Scientia Iranica*, 2011, **18**, 699-710.

Safari E., Sadjadi S., A hybrid method for flowshops scheduling with condition-based maintenance constraint and machines breakdown, *Expert Systems with Applications*, 2011, **38**, 2020-2029.

Santos H. G., Ochi L. S., Marinho E. H., Drummond L.M.A., Combining an evolutionary algorithm with data mining to solve a single-vehicle routing problem, *Neurocomputing*, 2006, **70**, 70-77.

SAS, *8 Design of experiments guide*, SAS Publishing, United States, 2008.

Silahtaroğlu G., *Kavram ve algoritmalarıyla temel veri madenciliği*, Papatya Yayıncılık, İstanbul, 2008.

Soltani R., Sadjadi S., Scheduling trucks in cross-docking systems: A robust metaheuristics approach, *Transportation Research*, 2010, **46**, 650-666.

Soylak M., Kalite gelistirmede deneysel tasarım ve taguchi yöntemi, Yüksek Lisans Tezi, Erciyes Üniversitesi, Fen bilimleri Enstitüsü, Kayseri, 2000.

Taillard E., Some efficient heuristic methods for the flow shop sequencing problem. *European Journal of Operations Research*, 1990, **47**, 65-74.

Tasgetiren F., Liang Y. C., Sevkli M., Gencyilmaz G., A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem, *European Journal of Operational Research*, 2007, **177**, 1930-1947.

Thabtah F., Cowling P., Mining the data from a hyperheuristic approach using associative classification, *Expert Systems with Applications*, 2008, **34**, 1093-1101.

Tong I., Wang H., Chen C., Chen T., Dynamic multiple responses by ideal solution analysis, *European Journal of Operational Research*, 2004, **156**, 433-444.

URL-1: http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/ (Ziyaret tarihi: 20.04.2013).

Usberti F., França P. M, França L. M., GRASP with evolutionary path-relinking for the capacitated arc routing problem, Computers & Operations Research, 2011, DOI: 10.1016/j.cor.2011.10.014.

Valdes R., Crespo E., Tamarit J., Villa F., GRASP and path relinking for project scheduling under partially renewable resources, *European Journal of Operational Research*, 2008, **189**, 1153-1170.

Valdes R. A., Parreno F., Tamarit J., A GRASP / Path relinking algorithm for two and three-dimensional multiple bin-size bin packing problems, Computers & Operations Research, 2012, DOI: 10.1016/j.cor.2012.03.016.

Villegas J. G, Prins C., Prodhon C., Medaglia A.,Velasco N., A GRASP with evolutionary path relinking for the truck and trailer routing problem, *Computers & Operations Research*, 2011, **38**, 1319-1334.

Vuchkov I., Boyadjieva L., *Quality improvement with design of experiments a response surface approach*, Kluwer Academic Publishers, USA 2001.

Wang C., Chu C., Proth J. M., Heuristic approaches for n=m=F==PCi scheduling problems, *European Journal of Operational Research*, 1997, **96**, 636-644.

Wang L., Wang S., Xu Y., Zhou G. , Liu M., A bi-population based estimation of distribution algorithm for the flexible job-shop scheduling problem, *Computers & Industrial Engineering*, 2012, **62**, 917-926.

Wang L., Zheng D. Z., An effective hybrid heuristic for flow shop scheduling, *International Journal of Advanced Manufacturing Technology*, 2003, **21**, 38-44.

Wang X., Tang L., A population-based variable neighborhood search for the single machine total weighted tardiness problem, *Computers & Operations Research*, 2009, **36**, 2105-2110.

Wilcox D. R, *Optimizing engineering design of a pneumatic oxygen valve with mathematical modeling and design of experiments*, ProQuest Information and Learning Company, United States, 2007.

Woo D. S., Yim H. S., A heuristic algorithm for mean flowtime objective in flowshop scheduling, *Computers and Operational Research*, 1998, **25**, 175-182.

Wu, C. F. J., Hamada M., *Experiments: Planning, analysis and parameter design optimization*, John Wiley and Sons, New York, 2000.

Yagiura M., Ibaraki T. B., Glover F. A., Path relinking approach with ejection chains for the generalized assignment problem, *European Journal of Operational Research*, 2006, **169**, 548-569.

Yamashita D., Armentano A., Laguna M., Scatter search for project scheduling with resource availability cost, *European Journal of Operational Research*, 2006, **169**, 623-637.

Yang W. H., Liao C. J., Survey of scheduling research involving setup times, *International Journal of Systems Science*, 1999, **30**, 143-155.

Yavuz M., Akcalı E., Tufekci S., A Hybrid Meta-Heuristic for the Batching Problem in Just-In-Time Flow Shops, *Journal of Mathematical Modelling and Algorithms*, 2006, **5**, 371-393.

Zare H. K., Fakhrzad M. B., Solving flexible flow-shop problem with a hybrid genetic algorithm and data mining: A fuzzy approach, *Expert Systems with Applications*, 2011, **38**, 7609-7615.

Zhang G. Q., Lai K., Combining path relinking and genetic algorithms for the multiple-level warehouse layout problem, *European Journal of Operational Research*, 2006, **169**, 413-425.

KİŞİSEL YAYINLAR VE ESERLER

[1] Özkan C., Akman G., **Özcan B.,** Effecting Factors Of Customer Satisfaction, *International Journal of Industrial Engineering - Theory, Applications and Practice*, 2010, **17**, 287-289.

[2] Akman G., **Özcan B.,** Hatipoğlu T., Fuzzy multi criteria decision making approach to innovative strategies based on miles and snow typology, Journal of Intelligent Manufacturing, DOI: 10.1007/s10845-013-0820-4.

[3] **Özcan B.**, Polimer ve polimer kompozitlerin enjeksiyonla üretim parametrelerinin Taguchi deney tasarımı yöntemi ile optimizasyonu, *Plastik Dergisi*, 2012, **122**, 126-134.

[4] Akman G., **Özcan B.**, A fuzzy qualıty functıon deployment (QFD) approach to determine customer needs for driving mirror, *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, 2011, **10**, 1-21.

[5] Hatipoğlu T., Boran S., **Özcan B.**, Fığlalı A., Yapay sinir ağı yaklaşımıyla çinko kalınlığının tahminlenmesi, Sakarya Üniversitesi Fen Bilimleri Dergisi, DOI: 10.5505/saufbe.21939.

[6] Akman G., **Özcan B**., Evaluatıon of skin wastes in leather industry by design of experiment, *10th International Constructive and Tecnological Design Optimization Conference*, Bacău, Romania, 23-25 May 2013.

[7] **Özcan B.,** Yuluğkural Y., Aras Ö., Determining customer requirements for advanced product qualıty plannıng (APQP) in automotive sector, *26th International Euro/Informs Conference*, Roma, Italy, 1-4 July 2013.

[8] **Özcan B.**, Yuluğkural Y., Aras Ö., Multi-Criteria Decision Support System for the Selection of Basketball Players, *26th International Euro/Informs Conference*, Roma, Italy, 1-4 July 2013.

[9] Kocabaş M., **Özcan B.**, Yulugkural Y., A hybrid multi-criterıa decision-making model for evaluating job satisfaction of white collar employees, *14. International Smyposium on Econometrics and Operation Research*, Sarajova, Bosnia Herzegovina, 24-28 May 2013.

[10] Yörükoğlu H., **Özcan B.**, Özkale C., Çelik C.,The analysis of the risks of renewable energy resources by using fuzzy fmea technıque, *14.International Smyposium on Econometrics and Operation Research*, Sarajova, Bosnia Herzegovina, 24-28 May 2013.

[11] Çoban O., Bora Ö., **Özcan B.**, Sınmazçelik T., Process optimization by design of experiments: Micro compounding and ınjection molding of PPS composit, *International Conference on Advances in Materials and Processing Technologies,* İstanbul, Turkey, 13-16 July 2011.

[12] Turan H., **Özcan B.**, Akman G., Kumru P., Process improvement study to approach of theory of constraints in banking sector, *International Symposium on Computing in Science and Engineering*, İzmir, Turkey, 3-5 June 2010.

[13] **Özcan B.**, Kurubaş Ö., Fığlalı N., Karakuzu C., Material selection by fuzzy analitic hierarchy process (AHP) in product design, *Internatıonal Conference on Sustainable Value Creatıon,* 12-14 November 2008.

[14] **Özcan B.**,Hatipoğlu T., Kılıçoğullari P., Using value stream mapping method in production process: an application in manufacturing sector, *Internatıonal Conference on Sustaınable Value Creatıon,* İzmir, Turkey, 12-14 November 2008.

[15] **Özcan B.**, Karataş M., Akman G., Using thinking processes in quality control, *International Symposium on Intelligent Manufacturing Systems*, Sakarya, Turkey, 14-17 October 2008.

[16] Yılmaz D., Akman G., **Özcan B.**, A planned maintence application at manufacuring firm by means of reliability analysis, *International Conference on Modelling and Simulation,* Konya, Turkey, 28-30 August 2006.

[17] Kılıçoğulları P., **Özcan B.**, Ertuğ B., Bir akaryakıt istasyonu seçiminde electre yönteminin kullanılması, *Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi,* Ankara, Türkiye, 22-24 Haziran 2009.

[18] Kumru P., **Özcan B.**, Yener F., Kobilerde yönetim organizasyon problemleri ve Kocaeli bölgesi'nde bir uygulama, *Üretim Araştırmaları Sempozyumu*, Eskişehir, Türkiye, 15-17 Ekim 2009.

[19] Karakurt İ., Kılıçoğulları P., **Özcan B.**, Analitik hiyerarşi yöntemi kullanılarak kobilerdeki yönetim etkinliğinin belirlenmesi, *Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi,* İstanbul, Türkiye, 30 Haziran-2 Temmuz 2008.

[20] Karakurt İ., Kılıçoğulları P., **Özcan B.**, Performans değerlendirme ve performansa dayalı ücretlendirme, *Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi,* İstanbul, Türkiye, 30 Haziran-2 Temmuz 2008.

[21] Gündüz U., Kılıçoğulları P., **Özcan B.**, Süreç yönetimi ve bir üretim sektöründe süreç iyileştirme çalışması, *Üretim Araştırmaları Sempozyumu,* İstanbul, Türkiye, 24-25 Ekim 2008.

[22] Yoluç V., Çalışkan S., Kılıçoğulları P., **Özcan B.**, Kalite güvence sistemleri ve Kocaeli il özel idaresin'de kalite yönetim sistemi iyileştirme çalışması, *Üretim Araştırmaları Sempozyumu*, İstanbul, Türkiye, 24-25 Ekim 2008.

[23] Kılıçoğulları P., **Özcan B.**, Toplam kalite yönetiminde, iş süreçleri yönetimi, ford otosan kaynak atölyesinde istatistiksel ergonomik iş süreçleri iyileştirme modellemesi, *İstatistik Kongresi,* Antalya, Türkiye, 20-24 Mayıs 2007.

[24] Kılıçoğulları P., **Özcan B.**, Toplam kalite yönetimde,müşteri memnuniyetinin istatiksel olarak ölçülmesi ve hizmet sektöründe uygulaması, *Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi*, İzmir, Türkiye, 2-4 Temmuz 2007.

[25] Yoluç V., Çalışkan S., Kılıçoğulları P., **Özcan B.**, Modern Yönetim Anlayışında Süreç Yönetimi ve Konstrüksiyon Üretimi Yapan Bir İşletmede Süreç Yönetiminde 5S Uygulaması, *Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi*, İzmir, 2-4 Temmuz 2007.

[26] Kılıçoğulları P., **Özcan B.**, Tekstil Sektöründe Faaliyet Gösteren Firmada Süreç İyileştirme Uygulaması ve Müşteri Memnuniyeti", *Endüstri- İşletme Mühendisliği Kurultayı*, Bursa, Türkiye, 9-10 Kasım 2007.

[27] Kılıçoğulları P., **Özcan B.**, Üretim Sektöründe Faaliyet Gösteren Bir Işletmede En Iyi Iş Sirasinin Tesbiti Ve Çizelgelenmesi, *Üretim Araştırmaları Sempozyomu,* Ankara, Türkiye, 15-17 Kasım 2007.

[28] Kılıçoğulları P., **Özcan B.**, Bir akaryakıt dolum tesisinde, iş sağlığı iş güvenliği, teknik emniyet ve ergonomik tedbirlerin incelenerek çevre yönetim sistemine entegrasyonu, *Ergonomi Kongresi*, Kayseri, Türkiye, 6-8 Aralık 2007.

[29] Kılıçoğulları P., **Özcan B.**, İş süreçleri yönetimi, Ford Otosan'da ergonomik iş süreçleri iyileştirme çalişmasi, *Ergonomi Kongresi,* Kayseri, Türkiye, 6-8 Aralık 2007.

[30] İzgi M., **Özcan B.**, Aladağ Z.,Yuluğkuğral Y., Kurumsal Karne Performans Göstergelerinin Ağırlıklandırılmasında Analitik Serim Sürecinin Kullanımı, *Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi*, Kocaeli, Türkiye, 3-5 Temmuz 2006.

[31] **Özcan B**., Erseven M., Fığlalı A., Yapay Bağışıklık Sisteminde Başlangıç Populasyonunun Çözüm Üzerine Etkisi, *Yöneylem Araştırması ve Endüstri Mühendisliği Kongresi*, Kocaeli, Türkiye, 3-5 Temmuz 2006.

[32] **Özcan B.**, Turan H., Şube Performans Ölçümünde Veri Zarflama Analizinin Kullanılması, *Üretim Araştırmaları Sempozyumu*, İzmir, Türkiye, 26-27 Eylül 2012.

ÖZGEÇMİŞ

1983'de Bolu'da doğdu. İlk, orta ve lise öğrenimini doğduğu şehirde tamamladı. 2001 yılında girdiği Kocaeli Üniversitesi Endüstri Mühendisliği Bölümü'nden 2005 yılında mezun oldu. 2005 yılında Kocaeli Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Bölümü'nde yüksek lisansa başladı. 2007 yılında yüksek lisans programını tamamlayarak aynı yıl Kocaeli Üniversitesi Endüstri Mühendisliği A.B.D'da doktora programına başladı. Halen Kocaeli Üniversitesi Mühendislik Fakültesinde Araştırma Görevlisi olarak görev yapmaktadır.