

**T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**UÇAKLARIN TERMİNAL KAPILARINA ATANMASI PROBLEMİNİN FARKLI
YÖNTEMLERLE ÇÖZÜMÜ VE UYGULAMASI**

ŞEFİKA ARSLAN

**YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI
ENDÜSTRİ MÜHENDİSLİĞİ PROGRAMI**

**DANIŞMAN
PROF. DR. HÜSEYİN BAŞLIGİL**

İSTANBUL, 2011

**T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**UÇAKLARIN TERMİNAL KAPILARINA ATANMASI PROBLEMİNİN FARKLI
YÖNTEMLERLE ÇÖZÜMÜ VE UYGULAMASI**

ŞEFİKA ARSLAN

**YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI
ENDÜSTRİ MÜHENDİSLİĞİ PROGRAMI**

**DANIŞMAN
PROF. DR. HÜSEYİN BAŞLIGİL**

İSTANBUL, 2011

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

UÇAKLARIN TERMİNAL KAPILARINA ATANMASI PROBLEMİNİN FARKLI
YÖNTEMLERLE ÇÖZÜMÜ VE UYGULAMASI

Şefika ARSLAN tarafından hazırlanan tez çalışması 01.12.2011 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı Endüstri Mühendisliği Programı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Prof. Dr. Hüseyin BAŞLIGİL
Yıldız Teknik Üniversitesi

Jüri Üyeleri

Prof. Dr. Hüseyin BAŞLIGİL
Yıldız Teknik Üniversitesi

Prof. Dr. Hüseyin YILMAZ
Yıldız Teknik Üniversitesi

Doç. Dr. Ali Fuat GÜNERİ
Yıldız Teknik Üniversitesi

ÖNSÖZ

Tez çalışmam sırasında bilimsel katkıları ile bana yardımcı ve yol gösterici olan tez danışmanım Sayın Prof. Dr. Hüseyin BAŞLIGİL' e; uçakların terminal kapılarına atanması ve genetik algoritma konularında geniş bilgi ve deneyimlerini benimle paylaşan ve benden desteğini esirgemeyen Sayın Prof. Dr. Ahmet BOLAT'a; çalışmada çok değerli katkıları olan Dr. Zeliha AKÇA'ya ve bugüne kadar gelebilmemi sağlayan tüm hocalarıma teşekkürü borç bilirim.

Ayrıca her zaman yanımda olan ve özellikle lisans ve yüksek lisans eğitimim ve tüm hayatım boyunca maddi ve manevi her türlü desteği veren annem ve babama ve tüm aileme teşekkürlerimi ve şükranlarımı sunar; ömrüm boyunca kendilerine minnet duyacağımı içtenlikle belirtirim.

Ve son olarak hayatımda sevgi ve şefkati ile yeni bir sayfa açan, varlığını her zaman hissettiren, sabır ve desteğini esirgemeyen hayat arkadaşşıma teşekkür ediyorum, iyi ki varsın...

Aralık, 2011

Şefika ARSLAN

İÇİNDEKİLER

	Sayfa
KISALTMA LİSTESİ.....	vi
ŞEKİL LİSTESİ.....	vii
ÇİZELGE LİSTESİ	viii
ÖZET	ix
ABSTRACT.....	xi
BÖLÜM 1	1
GİRİŞ.....	1
1.1 Literatür Özeti	2
1.2 Tezin Amacı	4
1.3 Hipotez	4
BÖLÜM 2	5
UÇAKLARIN TERMİNAL KAPILARINA ATANMASI PROBLEMİ	5
2.1 Uçakların Terminal Kapılarına Atanması Probleminin Tanımı	5
2.2 Uçakların Terminal Kapılarına Atanması Probleminin Formülasyonu	7
2.3 Havaalanı Kapı Ataması Problemi Çözüm Yöntemleri	9
2.3.1 Matematiksel Programlama Teknikleri	9
2.3.2 Simülasyon ve Kural Tabanlı Uzman Sistemler	10
2.4 Havaalanı Kapı Ataması Problemiyle İlgili Yapılmış Çalışmalar	10
BÖLÜM 3	20
UÇAKLARIN TERMİNAL KAPILARINA ATANMASI PROBLEMİNE BİR ÇÖZÜM	
YAKLAŞIMI ÖNERİSİ.....	20
3.1 Doyumsuz (Greedy) Algoritma Aşaması	23
3.2 Genetik Algoritma	25
3.2.1 Genetik Algoritma'nın Tanımı	25
3.2.2 Genetik Algoritmanın Adımları.....	26
3.2.2.1 Genetik Kodlama	27
3.2.2.2 Amacın Tanımlanması	28
3.2.2.3 Başlangıç Popülasyonun Yaratılması	28
3.2.2.4 Yeniden Üreme (Seleksiyon).....	28
3.2.2.5 Çaprazlama	30

3.2.2.6 Dönüşüm (Mutasyon)	32
3.2.2.7 Bitiş Ölçütlerine Ulaşılması	32
3.2.2.8 En iyi Sonucun Gösterilmesi	33
3.2.3 Yeni Bir Değiştirme Stratejili Genetik Algoritma	33
3.2.3.1 Yeni değiştirme stratejisi	33
3.2.3.2 Önerilen Genetik Algoritmanın Tanımı	34
BÖLÜM 4	37
ÖNERİLEN ÇÖZÜM YAKLAŞIMININ UYGULANMASI	37
4.1 Veriler	37
4.1.1 Rastgele Veriler	37
4.1.2 Gerçek Veriler	38
4.2 Rastgele Verilerle Uygulama	39
4.3 Gerçek Verilerle Uygulama	44
BÖLÜM 5	52
SONUÇ VE ÖNERİLER	52
KAYNAKLAR	55
EK-A	59
MATLAB'DE PROGRAM EKRAN GÖRÜNTÜSÜ	59
EK-B	60
ÖRNEK RASTGELE UÇUŞ VERİLERİ	60
EK-C	62
RASTGELE VERİLERİN KAPI UZAKLIK MATRİSLERİ	62
EK-D	64
RASTGELE VERİLERİN YOLCU AKIŞ MATRİSLERİNİN BİR KISMI	64
EK-E	65
SEÇİLEN ÖRNEKLERE YAPILAN GA DENEMELERİNİN SONUÇLARI	65
EK-F	66
UÇAK TİPİ VE KAPILARIN UYUMLULUK MATRİSİ	66
EK-G	67
1 GÜNLÜK DIŞ HAT UÇUŞ ÇİZELGESİNİN BİR KISMI	67
EK-H	68
UÇUŞ-YOLCU SAYISI MATRİSİNİN BİR KISMI	68
EK-I	69
AHL DIŞ HAT KAPILARININ UZAKLIK MATRİSİ	69
ÖZGEÇMİŞ	70

KISALTMA LİSTESİ

AUTKAP	Aşılmış Havaalanı Kapı Ataması Problemi
AHL	Atatürk Havalimanı
CHIP	Constraint Handling in Prolog-A Constraint Logic Programming Language
CKSA	Chiang Kai-Shek Havaalanı
ÇAHKAP	Çok Amaçlı Havaalanı Kapı Atama Problemi
DD	Domestic Domestic Leg
DI	Domestic International Leg
FCFS	First Come First Serve
GA	Genetik Algoritma
GM	Gri Diferansiyel Denklem
GRFRTFD	Gate Reassignment Framework For Real Time Flight Delays
HKYAP	Havaalanı Kapılara Yeniden Atama Problemi
ID	International Domestic Leg
II	International International Leg
KAP	Kuadratik Atama Problemi
PTB	Pareto Tavlama Benzetimi
QAPLIB	Quadratic Assignment Problem Library
RACES	Ramp Activity Control Expert System
RGA	Replacement Genetik Algoritma
SHKA	Sağlam Havaalanı Kapı Ataması
SKYAM	Stokastik Kapı Yeniden Atama Modeli
TTYA	Taiwan Taoyuan Havaalanı
UBSHM	Ulaştırma Bakanlığı Sivil Havacılık Müdürlüğü
UTKAP	Uçakların Terminal Kapılarına Atanması Problemi

ŞEKİL LİSTESİ

	Sayfa
Şekil 2.1 2002-2009 yılları arasında Türkiye havayolu istatistikleri [43]	6
Şekil 3.1 Genetik algoritmanın adımları	27
Şekil 3.2 Tek noktalı çaprazlama	31
Şekil 3.3 İki noktalı çaprazlama	31
Şekil 3.4 Uniform çaprazlama.....	32
Şekil 3.5 Mutasyon örneği.....	32
Şekil 3.6 Kromozom gösterimi	35
Şekil 3.7 Yeni genetik algorithmada çaprazlama şeması.....	36
Şekil 4.1 Örneklerin kullanım oranlarına göre ortalama iyileşme yüzdeleri	43
Şekil 4.2 Tüm örneklerin iyileşme yüzdeleri.....	44
Şekil 4.3 Yöntemlerin atama oranlarının grafiksel gösterimi.....	48
Şekil 4.4 RGA'nın minimum uygunluk değeri	50
Şekil 4.5 Elde edilen minimum maliyet değerleri.....	51

ÇİZELGE LİSTESİ

	Sayfa
Çizelge 3.1 Örnek uçuş çizelgesi.....	24
Çizelge 3.2 Uçuş kalkış zamanlarına göre uçuşların küçükten büyüğe sıralı hali.....	24
Çizelge 3.3 Doyumsuz algoritma ile kapılara atanan uçuşlar.....	26
Çizelge 3.4 FCFS kuralına göre atanan uçuşlar.....	25
Çizelge 3.5 Genetik algoritma ile ilgili tanımlar.....	26
Çizelge 4.1 Rastgele verilerin doyumumsuz algoritma-fcfs atanma oranları.....	40
Çizelge 4.2 Uygulanan genetik algoritma parametre değerleri	41
Çizelge 4.3 Seçilen örneklerle yapılan denemelerin özeti.....	42
Çizelge 4.4 Doyumsuz algoritma ile yapılan atamaların kapılarda gösterimi	42
Çizelge 4.5 Mevcut atamaların kapılardaki gösterimi	42
Çizelge 4.6 Yöntemlerin atama oranları.....	48
Çizelge 4.7 Elde edilen toplam maliyet değerleri.....	49

UÇAKLARIN TERMİNAL KAPILARINA ATANMASI PROBLEMİNİN FARKLI YÖNTEMLERLE ÇÖZÜMÜ VE UYGULAMASI

Şefika ARSLAN

Endüstri Mühendisliği Anabilim Dalı

Endüstri Mühendisliği

Yüksek Lisans Tezi

Tez Danışmanı: Prof. Dr. Hüseyin BAŞLIGİL

Uçakların terminal kapılarına atanması problemi havaalanı performans ölçütü olan yolcu memnuniyetini doğrudan etkileyen havaalanı işletmesinin içerdiği en önemli konulardan biridir. Havaalanı kapı ataması işlemi, uçakların çeşitli ölçütler göz önünde bulundurularak kapılara (köprülere) tahsis edilmesi işlemidir.

Kapıların verimli kullanımı ve yolcu memnuniyeti üzerinde büyük etkiye sahip olan kapı ataması işlemi yapılırken, yolcu yürüme mesafesi/süresi, bagajlara erişim mesafesi/süresi, uçuşlara ait zaman çizelgeleri, uçak-kapı boyutlarının uyumu gibi pek çok ölçüt dikkate alınmaktadır.

Günümüzde kalite kavramının gittikçe önem kazanması, beklentileri yerine getirirken maliyetlerin (kantitatif ve kalitatif) minimize edilmesine yönelik çalışmaların artmasına neden olmuştur.

Atama problemlerinin çözümünde, analitik yöntemlerin maliyeti yüksek çözümler sunması sebebiyle sezgisel yöntemlerin kullanılması yaygınlık kazanmıştır. Bu bağlamda havaalanı kapı ataması problemine iki aşamalı bir çözüm yaklaşımı geliştirilmiştir. İlk adımda (eğer varsa) uçuş sayısının kapı sayısını aştığı durum elimine edilerek uçuşlar kapılara yerleştirilmiş, ikinci adımda farklı parametrelerle genetik algoritma meta sezgiseli kullanılarak yolcu yürüme mesafesi en aza indirilmeye çalışılmıştır.

Önerilen çözüm yaklaşımı Atatürk Havalimanı'ndaki kapı atama problemini çözmek için uygulanmıştır. Gerçekleştirilen ve adımları ayrıntılı olarak tez kapsamında ifade edilen metodun önce rastgele verilere ve sonra AHL' deki verilere uygulanması sonucunda yolcu yürüme mesafesinin en aza indirilmesi sağlanmıştır.

Anahtar Kelimeler: Havaalanı kapı ataması problemi, atama, doyumsuz algoritma, genetik algoritma, optimizasyon

**DIFFERENT SOLUTION APPROACHES FOR AIRPORT GATE ASSIGNMENT
PROBLEM AND ITS APPLICATION**

Şefika ARSLAN

Department of Industrial Engineering
MSc. Thesis

Advisor: Prof. Dr. Hüseyin BAŞLIGİL

Airport gate assignment problem is one of the most important issues contained by airport operations that directly affect passenger satisfaction which is a measure of airport performance. Airport gate assignment is the process of allocating the aircraft to the gates (bridges) by taking into account various criteria.

During the gate assignment process that has major influence on efficient usage of the gates and passenger satisfaction, many criteria are taken into account such as passenger walking distance/time, luggage access distance/ time, flight schedules, aircraft-gate size conformity.

Nowadays the increasing importance of quality concept has led to increase the studies for minimizing (quantitative and qualitative) costs while fulfilling the expectations.

While solving the assignment problem, because of the high cost solutions obtained from analytical methods, usage of heuristic methods has thrived. In this context a two phase solution approach has developed for the airport gate assignment problem. In the first step the situation where the number of flights exceeds the number of gates has eliminated. In the second step total passenger walking distance is tried to minimize using genetic algorithm metaheuristic with different parameters.

The proposed solution approach has been applied to solve gate assignment problem at Atatürk Airport. As a result of the implementation of method whose steps have been

denoted within the context of the thesis to random data and data obtained from AHL, passenger walking distance minimization is provided.

Key words: Airport gate assignment problem, assignment, greedy algorithm, genetic algorithm, optimization

BÖLÜM 1

GİRİŞ

Havayolu taşımacılığındaki trafik yoğunluğunun artması, talebin karşılanabilmesi için havaalanlarında sürekli bir gelişimi zorunlu kılmaktadır. Gelişim, fiziksel anlamda sınırlı bir biçimde gerçekleştirilebileceğinden havaalanlarında işletmenin iyileştirilmesi konusuna ağırlık vermek daha anlamlı bir hale gelmiştir.

Havaalanı işletmesinin içerdiği en önemli konulardan biri, havaalanı performans ölçütü olan yolcu memnuniyetini doğrudan etkileyen havaalanı kapı ataması işlemidir. Havaalanı kapı ataması işlemi, uçakların çeşitli ölçütler göz önünde bulundurularak kapılara (köprülere) tahsis edilmesi işlemidir. Kapıların verimli kullanımı ve yolcu memnuniyeti üzerinde büyük etkiye sahip olan kapı ataması işlemi yapılırken, yolcu yürüme mesafesi/süresi, bagajlara erişim mesafesi/süresi, uçuşlara ait zaman çizelgeleri, uçak-kapı boyutlarının uyumu gibi pek çok ölçüt dikkate alınmaktadır.

Günümüzde kalite kavramının gittikçe önem kazanması, beklentileri yerine getirirken maliyetlerin (kantitatif ve kalitatif) minimize edilmesine yönelik çalışmaların artmasına neden olmuştur.

Bu çalışmada amacı kapılara atanmamış uçuşları ve toplam yürüme mesafesini en aza indirmek olan UTKAP (Uçakların terminal kapılarına atanması problemi) üzerine çalışılmıştır. Tanımladığımız problem için iki aşamalı bir çözüm yaklaşımı önerdik. İlk aşamada kapılara atanmayan uçuşları en aza indirecek bir doyumsuz (greedy) algoritma kullandık. İkinci kısımda toplam yürüme mesafesini en aza indirmek için çeşitli genetik algoritma yaklaşımları kullandık.

Bolat tarafından [21] de verilen rastgele veriler kullanılarak ve bu verilere dayalı havaalanı-yolcu verileri türetilerek iki aşamalı çözüm önerimizi uyguladık ve yöntemi test ettik. Daha sonra iki aşamalı çözüm önerimizi Atatürk Havalimanı'ndaki kapı atama problemini çözmek için uyguladık. AHL' den alınan 1 günlük gerçek uçuş verileri, 23 dış hat kapısı ve 58 açık park alanı, AHL terminal yapısı ve ortalama yolcu sayısı bilgileri göz önüne alınarak hesaplamalar yapıldı.

1.1 Literatür Özeti

Havaalanı kapı ataması problemi birçok araştırmacı tarafından incelenmiştir. Braaksma ve Shortreed [1] de terminal içi seyahati en aza indirmek için bir tasarım süreciyle birleşmiş nicel yöntemler kullandı. Babic vd. [2] de UTKAP'a daha hızlı bir şekilde optimal bir çözüm bulmak için geliştirilmiş dal sınır algoritması kullandılar. Bunun aksine transfer yolcularını göz önüne alan [3] te Mangoubi ve Mathaisel problemi çözmek için doğrusal programlama gevşetmesi ve doyumsuz (greedy) sezgiseli kullandılar. Bihr [4] te UTKAP'da gelişleri sabit olan uçuşlar için en az yürüme mesafesini bulmak için ikili doğrusal programlama kullandı. Yukarıdaki iki çalışmanın aksine bu UTKAP bir atama problemi olarak ele alınarak daha kolay formüle edildi. Wirasinghe ve Bandara [5] te ve Wirasinghe vd. [6] da terminal içi seyahati en aza indirmek için havaalanı terminal tasarım sürecine gecikme maliyetlerini de dâhil ettiler. Robuste [7] de çalışmayı yolcuların ve bagajların transferini içeren merkez terminaline genişletti. Obata [8] de bağlantılı yolculuk yapan yolcuların gelen uçuş kapısından gidecek uçuşun kapısına yürüme mesafelerini en aza indirmeyi amaçladı. Xu ve Bailey aynı amaca sahip [9] da problemi KAP olarak ele aldılar ve 0-1 tamsayılı problem olarak tekrar formüle ettiler. Haghani ve Chen [10] da çok zaman aralıklı kapı atama problemini, tamsayılı programlama ile formüle ettiler.

Bazı modeller stokastik uçuş gecikmelerini (erken veya geç gelişler ve geç kalkışları içeren) dikkate alarak statik kapı atamasının performansını arttırmayı denediler. Örneğin, Hassounah ve Steuart [11] de planlı tampon sürelerinin uçuş programının dakikliğini arttırabileceğini gösterdiler. Yan ve Chang [12] de ve Yan ve Huo [13] de statik kapı ataması problemlerinde stokastik uçuş gecikmelerini absorbe etmek için aynı kapiya atanan iki art arta uçak arasında sabit tampon süre kullandılar. Yan ve

Chang [12] de çoklu emtialı ağ akış modeli geliştirdiler. Yan vd. [14] te stokastik uçuş gecikmelerinin sadece statik kapı atamaları ([12], [13]) üzerine etkilerini analiz edebilen değil, aynı zamanda esnek tampon sürelerini ve gerçek zamanlı kapı atama kurallarını değerlendirebilen bir simülasyon çatısı önerdiler. Yan ve Tang [15] te problemi stokastik uçuş gecikmeleri altında etkin bir şekilde entegre çözebilen bir çerçeve içine gömülü sezgisel bir yaklaşım geliştirdiler.

Ding vd. [16], [17] ve [18] de uçuş sayısının mevcut kapı sayısını aştığı UTKAP üzerine çalıştılar. Son zamanlarda bazı yazarlar UTKAP'ın dinamik karakterini dikkate almaya çalıştı. Hem Dorndorf vd. [19] da hem de Drexl ve Nikulin [20] de başlangıçta Ding vd. tarafından [16], [17] ve [18] de önerilen çok amaçlı UTKAP'ı ele aldılar. Sağlam programları amaçlayan bir yaklaşım gerçekleştirmek için yapılan ilk girişimlerden biri Bolat tarafından [21], [22] ve [23] te gerçekleştirildi ve yazar giriş verilerindeki küçük değişikliklerden etkilenmeyecek sağlam program elde etmek için kapıların olabildiğince düzgün kullanılmasını önerdi. Diğer bir sağlam havaalanı kapı atamasını da Lim ve Wang [24] te önerdiler. Ayrıca Yang ve Hu [25] te kapı atama sağlamlığını artırmak amacıyla her uçuşun varış ve kalkış saatini tahmin eden ve sonra tahmin sonuçlarına göre kapılara uçuşları atayan bir strateji önerdiler. Bolat'ın [21] ve [22] deki amacına benzer şekilde Zheng vd. [26] da tüm boş zamanların varyansının en aza indirilmesini önerdiler. Yukarıdaki çalışmalara benzer amaçlar (toplam yürüme mesafesini en aza indirmek ve sağlam kapı ataması) içeren [27], [28] da Wei ve Liu çok amaçlı modeller sundular. Bu iki çalışmaya benzer şekilde Hu ve Di Paolo [29], [30] de UTKAP'a genetik algoritma tabanlı çözümler önerdiler.

Literatürde kapılara yeniden atama problemini çözmek için de bazı yaklaşımlar sunulmuştur. Gu ve Chung [31] de minimum ekstra gecikme zamanlı çizelgeleri en az deneyimli kapı yöneticileri tarafından üretilen çözümler kadar verimli bir şekilde hesaplayan etkili bir genetik algoritma önerdiler. Tang [32] de (ihlal kavramını dahil eden) kapı yeniden atamaları için geçici kapı sıkıntılarının ve stokastik uçuş gecikmelerinin olduğu bir model (SKYAM) geliştirdi. Yan vd. [33] te güçlü bir temel ağ yapısına sahip kapı yeniden atama modeli geliştirdiler. Benzer şekilde Tang vd. [34] te çeşitli uçuş gecikme bilgileri doğrultusunda, tekrarlanan kapı yeniden atamalarını ele

almak için tasarlanmış bir kapı yeniden atama çerçevesi, sistematik bir bilgisayar aracı, geliştirdiler.

İkinci temel araştırma akımı simülasyon ve kural tabanlı uzman sistemlerin inşası üzerine yoğunlaşmaktadır. Birçok yazar kural tabanlı uzman sistem tasarımı konusuna odaklandılar [35–38]. Uzman sistemler yer kontrolörlerinden elde edilen bilgiye dayanarak atamaları üretmek için üretim kurallarını kullanır. Hamzwawi [35] te uçuşların kapılara atanmasını simüle etmek için ve kapı kullanımıyla ilgili belirli kuralların etkilerini değerlendirmek için kural tabanlı bir sistem ortaya koydu. Gosling [36] da Denver Stapleton havaalanının önemli bir merkezinde bir kapı ataması için uygulanmış uzman bir sistem ortaya koydu. Muthukrishnan ve Srihari [37] de UTKAP'ı çözmek için benzer bir yaklaşım kullandılar ve ayrıca duyarlılık analizinin nasıl uygulanacağını açıkladılar. Cheng [39], [40], [41] ve [42] de çoklu amaçlı kısmi paralel atamalar sağlamak için matematiksel programlama tekniklerinin bilgi tabanlı bir kapı atama sistem içine entegrasyonunu ortaya koydu. Optimizasyon ve kural temelli yaklaşımların her ikisi de simülasyon analizleri ile kombine edildi [35], [39].

1.2 Tezin Amacı

Bu tez çalışmasının amacı, havaalanı kapı ataması problemi için aprona atanan uçuşları ve toplam yolcu yürüme mesafesini en aza indirmeye yönelik bir çözüm yaklaşımı önermektir. Tez çalışmasında önerilen çözüm yaklaşımının uçuş sayısının kapı sayısını aştığı durumdaki statik kapı atama problemlerine uygulanabilecek şekilde esnek bir yapıya sahip olması ve etkili ve verimli sonuçlar vermesi hedeflenmektedir.

1.3 Hipotez

Bu tez çalışmasının başlangıç noktası, havaalanı kapı ataması problemi için ortaya konulacak çözüm yaklaşımının problemin yapısı nedeni ile karmaşık olması düşüncesidir. Bu bağlamda matematiksel programlama tekniklerinden doyumsuz algoritma ve genetik algoritmanın yetenekleri bir arada kullanıldığında literatüre daha net ve anlamlı sonuçlar üretecek bir çözüm yaklaşımı kazandırılabilir.

UÇAKLARIN TERMİNAL KAPILARINA ATANMASI PROBLEMİ

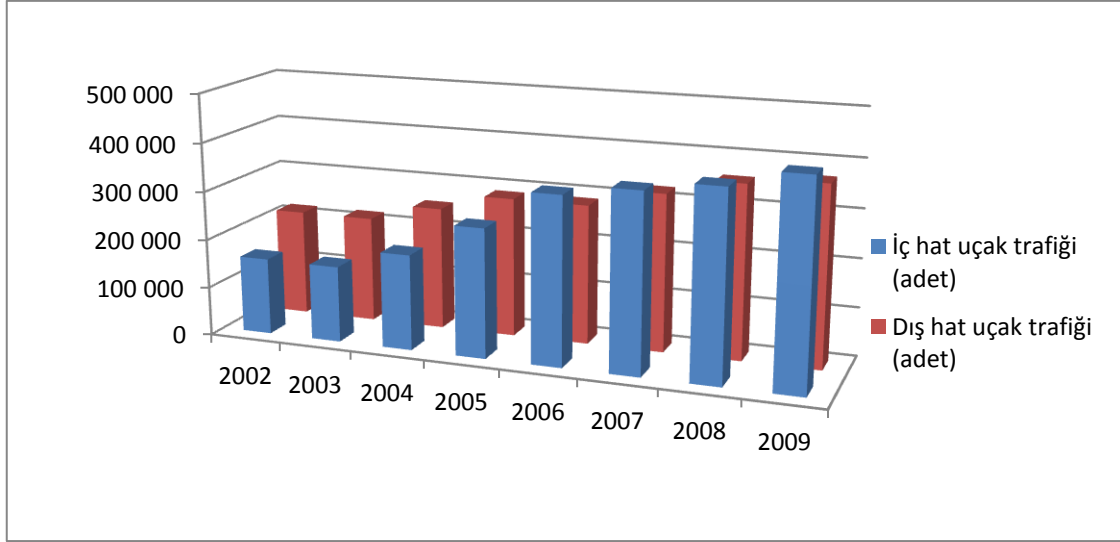
2.1 Uçakların Terminal Kapılarına Atanması Probleminin Tanımı

Havaalanında geliş ve gidiş zamanları belli olan uçakların uygun kapılara atanmasına uçakların terminal kapılarına atanması problemi denir (UTKAP). UTKAP çizelgeleme problemi ve atama probleminin bir kombinasyonudur. Uçuşlar için geliş ve gidiş zamanları UTKAP'ın çizelgeleme kısmını oluştururken hangi uçuşun hangi kapıya atanması gerektiği sorusu da UTKAP'ın atama kısmını oluşturur.

Uçuş kapı çizelgelemesinin temel amacı çizelgelenmiş uçuşlar için havaalanına gelen yada havaalanından giden yolculara gereksiz rahatsızlık vermeyen düzgün uçuş operasyonları oluşturacak şekilde optimum kapı atamasını bulmaktır. Birçok araştırmacı havayolu işletmeciliğinde önemli olan operasyon maliyetlerini en aza indirmeye ya da geliri en üst seviyeye çıkarmaya öncelik vermiştir. Uçuş-kapı atamasının etkili yapılması operasyon maliyeti ve müşteri memnuniyeti açısından son derece önemlidir. Bu nedenle uçuş kapı çizelgelemesi optimum uçuş kapı atama politikası ile çeşitli ortak amaçları yansıtır şekilde uyumludur. Bazı karakteristik ve ortak amaçlar aşağıdaki gibidir:

1. Gelen ya da giden yolcuların toplam yürüme mesafesinin en aza indirilmesi.
2. Transfer yolcularının kapılar arasındaki toplam yürüme mesafesinin en aza indirilmesi.
3. Aprona atanan (Kapılara atanamayan) uçuş sayısının en aza indirilmesi.
4. Kapıların kullanım oranlarının en üst seviyeye çıkarılması.
5. Belli uçakların belli kapılara atanma seçeneklerinin en üst seviyeye çıkarılması.

İstenen birçok amaca aynı anda hizmet edecek optimum uçuş kapı atamaları yapmak son derece zordur. Son yıllarda hava trafik yoğunluğunun artışıyla birlikte operasyon maliyetlerini düşürmek, müşteri memnuniyetini arttırmak, karışıklıkları en aza indirmek gibi amaçlar göz önüne alındığında kapıların etkili kullanım ihtiyacı çok önemli bir hal almıştır.



Şekil 2.1 2002-2009 yılları arasında Türkiye havayolu istatistikleri [43]

Uygulamada havaalanı yönetimi iki çeşit atama problemiyle karşılaşır: Planlama seviyesinde atama ve gerçek zamanlı yeniden atama. Havaalanı operasyonundaki her bir havayolu için verilen uçuş çizelgesi göz önüne alınarak havaalanı yönetiminin hangi uçağın hangi kapıya atanması gerektiğini genelde 12 yada 24 saati baz alarak tanımlaması gerekir. Bu yüzden uçuş kapı atamasının planlanması 12 ya da 24 saat önceden yapılmalıdır. Bu UTKAP için planlama seviyesinde atamadır. Genelde uçuş çizelgelerinin havayolu şirketleri tarafından birkaç günlük olacak şekilde güncellenmesine rağmen havaalanı operasyonunda farklı havayolu şirketlerinin uçuşları olacağı için, sıkça uçuş çizelgesi değişiklikleri yaşanmaktadır. Bu nedenle kapı atamasının aylık ya da haftalık planlanması yerine, genelde havaalanı yönetimi günlük bir plan oluşturur. Hazırlanması için yeterli zaman kalması için atamanın planlanması gece yarısı gibi yoğun olmayan saatlerde çizelgelenir ve gerçekte büyük bir havaalanı için zaman alıcı olan bu kapı atama programının çalışması için daha çok zaman kalır.

Günlük uçuş kapı ataması planlamasında bir yolcunun havaalanında ayrılış kapısına, bagaj teslim alanına ya da bağlantılı uçuşa gitmesi için yürümesi gereken uzaklık havaalanı terminalinde sabit uçak kapılarının etkili kullanımı için önemli bir kriterdir. Havaalanı yönetiminin yolculara daha iyi hizmet sağlamak amacıyla özellikle kısa bir zaman aralığında başka bir uçuşa geçmesi gereken transfer yolcularının yolcu yürüme mesafelerini düşürerek uçakları kapılara ataması gerekmektedir. Havaalanında yoğun saatler esnasında, belli bir zaman aralığında kapı sayısından fazla uçuş sayısı olduğunda tüm uçaklar için belli bir kapı olmamasıyla sonuçlanan uç bir durum söz konusu olabilir. Böyle bir durumda uçaklardan bazıları açık park alanlarına atanmak zorunda kalabilir. Bundan dolayı yolcular havaalanı terminaline sabit kapılardaki köprülerden geçerek gitmek yerine servislerle gitme rahatsızlığıyla yüzleşmek zorunda kalacaklardır. Havaalanı yönetimi pratikte böyle durumların olmasından kaçınmaktadır [44].

2.2 Uçakların Terminal Kapılarına Atanması Probleminin Formülasyonu

Havaalanı kapı ataması işleminde gelen/giden uçakların hizmet vermek veya almak için park ettikleri yerlerin seçimi büyük önem taşımaktadır. Bir optimizasyon problemi olan kapı ataması işleminde pek çok amaç fonksiyonu kullanılabilir. Bunlardan bazıları yolcu yürüme mesafesinin, kapalı kapı sayısının, kapıların boş kaldığı sürelerin en aza indirilmesi olarak ifade edilebilir. Tek bir amaç fonksiyonunun en küçüklenmesi veya en büyüklenmesi söz konusu olabileceği gibi, amaç olarak birden fazla fonksiyonun ele alındığı (çoklu amaç fonksiyonu içeren) problemler de bulunmaktadır.

Ding vd. [16] da kapı ataması problemi için geliştirdikleri modelde aşağıdaki notasyonları kullanılmışlardır:

N: havaalanına gelen veya havaalanından giden uçuşların kümesi

M: havaalanındaki uygun kapı kümesi

n: toplam uçuş sayısı

m: toplam kapı sayısı

a_i : i uçuşunun geliş zamanı ($1 \leq i \leq n$)

d_i : i uçuşunun kalkış zamanı ($1 \leq i \leq n$)

$w_{k,l}$: k kapısından l kapısına giden yolcuların yürüme mesafesi ($1 \leq k, l \leq m$)

$f_{i,j}$: i uçuşundan j uçuşuna transfer yapan yolcu sayısı ($1 \leq i, j \leq n$)

Burada, $n > m$ durumu kapasite üstü bir durumu yani uçuş sayısının kapı sayısından fazla olması durumunu ifade eder. Ek olarak, iki hayali kapı kullanılacaktır. Kapı 0, havaalanının giriş/çıkış kapısını temsil etmektedir. Kapı $m+1$ ise uçakların uygun kapı olmadığı zaman yönlendirildiği apronu (açık park alanını) ifade eder. Buna göre, $w_{k,0}$ k kapısı ile havaalanının giriş/çıkış kapısı arasındaki yürüme mesafesini, $f_{i,0}$ i uçuşu ile havaalanına gelen yolcu sayısını ve $w_{m+1,k}$ ise apron ve k kapısı arasındaki yürüme mesafesini (bu mesafe genelde kapılar arasındaki mesafelerden daha fazladır) ifade eder.

$y_{i,k}$ ikili değişkeni, eğer i uçuşu k kapısına atanmışsa 1 değerini alır ($0 \leq k \leq m+1$), diğer durumda ise 0 değerini alır.

Bu koşullarda;

$$y_{i,k} = y_{j,k} = 1 \quad (k \neq m+1) \text{ ifadesi } a_i > d_j \text{ veya } a_j > d_i \quad (1 \leq i, j \leq n) \quad (1.1)$$

durumunu belirtmektedir. Bu koşul aynı anda iki farklı uçuşun aynı kapıya atanmasına engel olmaktadır.

Modeldeki amaçlar, aprona atanan uçuş sayısını ve toplam yürüme mesafesini minimize etmektir. Toplam yürüme mesafesi üç bileşenden oluşmaktadır: transfer yolcularının yürüme mesafesi, havaalanına gelen yolcuların yürüme mesafesi ve havaalanından gidecek olan yolcuların yürüme mesafesi.

Havaalanı kapı ataması problemi aşağıdaki şekilde formüle edilmiştir:

$$\min \sum_{i=1}^n y_{i,m+1} \quad (1.2)$$

$$\min \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^{m+1} \sum_{l=1}^{m+1} f_{i,j} w_{k,l} y_{i,k} y_{j,l} + \sum_{i=1}^n f_{0,i} w_{0,i} + \sum_{i=1}^n f_{i,0} w_{i,0} \quad (1.3)$$

Aşağıdaki kısıtlar altında

$$\sum_{k=1}^{m+1} y_{i,k} = 1 (1 \leq i \leq n) \quad (1.4)$$

$$a_i \leq d_i (1 \leq i \leq n) \quad (1.5)$$

$$y_{i,k} y_{j,k} (d_j - a_i) (d_i - a_j) \leq 0 (1 \leq i, j \leq n, k \neq m + 1) \quad (1.6)$$

$$y_{i,k} \in \{1, 0\} (1 \leq i \leq n, 1 \leq k \leq m + 1) \quad (1.7)$$

(1.4) eşitliği her bir uçuşun sadece ve sadece bir kapıya veya aprona atanabileceğini belirtir. (1.5) eşitliği her uçuşun kalkış zamanının geliş zamanından büyük olduğunu ifade eder. Eşitlik (1.6) eğer iki uçuş aynı kapıya atanmışsa bu uçuşların zaman çizelgelerinin aynı zamana denk gelmemesi gerektiğini belirtir. Eşitlik (1.7) ise (1.1) eşitliğinde ifade edilen ve aynı kapıya aynı anda birden fazla uçağın atanmasını önlemek amacını taşır [16].

2.3 Havaalanı Kapı Ataması Problemi Çözüm Yöntemleri

Uçuş-kapı atama işlemi için geliştirilmiş iki ana araştırma akımı bulunmaktadır. İlki “Matematiksel Programlama Teknikleri”, ikincisi ise “Simülasyon ve Kural Temelli Uzman Sistemler”’dir [19].

2.3.1 Matematiksel Programlama Teknikleri

Kapı atama probleminin çözümüne yönelik olarak öncelikle kesin sonuç veren analitik modeller kullanılmıştır. Bu modellerin problemin gittikçe karmaşık bir hal alması nedeniyle çeşitli sezgisel yöntemler geliştirilmiştir.

- **Kesin Sonuç Veren Algoritmalar:** Atama probleminde kullanılan analitik yöntemlerden bazıları dinamik programlama, kesme yüzeyi ve dal-sınır teknikleridir. Bu algoritmalar iyi sonuçlar vermekle birlikte çok zaman alan ve sınırlı büyüklükteki problemler için etkili yöntemlerdir.
- **Sezgisel Algoritmalar:** Sezgisel algoritmalar, herhangi bir amacı gerçekleştirmek veya hedefe varmak için çeşitli alternatif hareketlerden etkili olanlara karar vermek amacıyla tanımlanan kriterler veya bilgisayar metotlarıdır. Atama problemlerinde sık

kullanılan sezgisel yöntemler; ısıt işlem, genetik algoritma, karınca kolonisi optimizasyonu ve tabu araştırmasıdır.

2.3.2 Simülasyon ve Kural Tabanlı Uzman Sistemler

Klasik araştırma tekniklerini kullanan yaklaşımlarda çok sayıda bilgi ve performans kriteri sebebi ile oluşan zorluklar, alternatif olarak çalışmaları kural tabanlı uzman sistem modellerine yöneltmiştir. Yer kontrolcülerinden alınan bilgilere göre oluşturulan uzman sistem, atamaları oluşturmak için üretim kuralları tanımlar. Dikkate alınması gereken faktör sayısı büyüktür, bu nedenle, en önemli konu tüm kuralları tanımlamak, önem sırasına göre listelemektir. Pratik bir bakış açısına göre, matematiksel programlama tekniklerinin kullanımını sağlayan bir uzman sistem geliştirmek daha önemlidir. Bu tip bir bileşke, kapı çizelgeleme sisteminin beklenen esneklik özelliğini oluşturmayı sağlayabilir [45].

2.4 Havaalanı Kapı Ataması Problemiyle İlgili Yapılmış Çalışmalar

Uçuş-kapı atama işlemi için geliştirilmiş iki ana araştırma akımı bulunmaktadır. İlki “Matematiksel Programlama Teknikleri”, ikincisi ise “Simülasyon ve Kural Temelli Uzman Sistemler” dir [5].

Araştırmamıza Matematiksel Programlama Teknikleriyle başlayacağız;

Terminal içi seyahati en aza indirmek için bir tasarım süreciyle birleşmiş nicel yöntemler kullanmak için ilk çabalardan biri Braaksma ve Shortreed tarafından [1] de gösterilmiştir.

Wirasinghe vd. [6] da bu alandaki çalışmayı genişletmiş ve yürüme mesafesini hesaplamak için çoklu iskeleli (multi pier) terminal geometrisini de içeren bir yöntem önermişlerdir. Ayrıca çalışmalarında bir havaalanı için yolcu yürüme mesafesine dayalı en uygun geometriyi bulmak için bir yöntem sundular. Robuste [7] de çalışmayı yolcuların ve bagajların transferini içeren merkez terminaline genişletti. Bandara ve Wirasinghe [5] te Wirasinghe vd.’nin [6] daki çalışmalarını takip etti ve sonucunu genelleştirmek için çalışmalarına daha çok terminal tipleri, merkez transferleri, eşit olmayan iskeleler vb. ekleyerek ön havaalanı terminal tasarımı için bazı kurallar

önerdiler. Belirli bir trafik hacmi ve yolcu kompozisyonuna göre Bandara ve Wirasinghe [5] te havaalanı terminalleri uygunluğunu değerlendirmek için yöntemler önerdiler ve terminal içi seyahati en aza indirmek için havaalanı terminal tasarım sürecine gecikme maliyetlerini de dahil etmişlerdir.

Birçok çalışma, havaalanı terminali için ön tasarım süreci optimizasyonuna dayalıyken, Babic vd.'nin [2] de ve Mangoubi ve Mathaisel'in [3] te yaptığı gibi diğer araştırmacılar UTKAP'ı çözmek için yöneylem araştırması ve tamsayılı programlama modellemesi yaklaşımları ile çalıştılar. Böyle yaklaşımlarda belirli bir havaalanı terminali düzeni göz önüne alındığında, toplam yolcu yürüme mesafesi varış ve ayrılış yolcu hacmine, transfer yolcularının hacmine, kapıdan kapıya uzaklığa, kapı bagaj teslim alanı arası uzaklığa, giriş kapı arası uzaklığa ve uçuş kapı atamalarına dayanmaktadır. Problemin modellenmesinde, uçağın kapıya atanması ile ilgili maliyet kapılar, kontrol noktası ve bagaj teslim alanı gibi önemli tesisler arası mesafelere bağlı olduğu kadar bu tesisler arası ilişkilere de bağlıdır. Babic vd. [2] de kapı atama problemini doğrusal 0-1 tamsayılı programlama modeli olarak formüle ettiler. Uçak kapı atamasına optimal bir çözüm bulmak için transfer yolcularının göz önüne alınmadığı bir dal-sınır algoritması kullandılar. Mangoubi ve Mathaisel [3] te Babic vd. [2] deki problemi çözmek için doğrusal programlama gevşetmesi ve doyumsuz (greedy) sezgiseli kullandılar. Transfer yolcularının yürüme mesafesini düzgün dağılımlı kapıdan kapıya transfer modeline dayalı olarak değerlendirdiler. Ayrıca problemi lineerleştirilmiş ikinci dereceden bir kapı atama modeli olarak formüle ettiler. Ancak çok sayıda değişkenler ve kısıtlar içeren lineerleştirilmiş ikinci dereceden bir tamsayılı programlama modelini çözmenin zor ve zaman alıcı olacağına dikkat çektiler. Bu yüzden önerilen model üzerinde daha fazla çalışmadılar.

Bihr'in [4] te amacı ilgili transfer kapıları arasındaki transfer yolcu yürüme mesafelerini en aza indirmektir. UTKAP'da gelişleri sabit olan uçuşlar için en az yürüme mesafesini bulmak için ikili doğrusal programlama kullandı. Yukarıdaki iki çalışmanın aksine bu UTKAP bir atama problemi olarak ele alınarak daha kolay formüle edildi. Bu çalışmaların sonuçları kesin adımlarla yürüme mesafesinde önemli ilerlemenin elde edilebilir olduğunu gösterdi.

Genellikle 0-1 tamsayı (doğrusal veya kuadratik) programlar, karma tamsayılı programlar veya ağ akış problemleri olarak formüle edilmiş bir dizi analitik kapı uçuş atama modelleri geliştirilmiştir [8], [10], [21], [22], [23].

Sağlam programları amaçlayan bir yaklaşım gerçekleştirmek için yapılan ilk girişimlerden biri Bolat tarafından [21], [22] ve [23] te gerçekleştirilmiştir. Bolat'ın [21] ve [22] deki amaçları toplam yolcu bekleme zamanının, toplam yolcu yürüme mesafesinin, kapılara atanamayan uçuşların en aza indirilmesi ve kapıların kullanıldığı zaman aralıklarının en üst seviyeye çıkarılması ve kapıların boş kalma zamanlarının düzgün dağılması ya da bu amaçların kombinasyonu şeklindedir. Ancak, yukarıda adı geçen tüm modellerin planlanan kapı atamalarını sonuçlandırmak için önerilen uçuş planı sabittir ve gerçek günlük operasyonlarda ortaya çıkan ortak stokastik uçuş gecikmeleri ihmal edilmiştir. Bolat [21] ve [22] de matematiksel modeller ve sezgisel ve optimal prosedürleri kullanarak sağlamlık için kapılardaki boş zamanları olabildiğince düzgün dağıtmayı önerdi. Bolat [23] te UTKAP' a boş zaman periyotlarının asgari dağıldığı çözümler sunmak için matematiksel modeller ve sezgisel ve optimal prosedürler önerdi. Probleme özel bilgi kullanmak için bir genetik algoritma da önerildi. Büyük boyutlu ve gerçek problemler üzerindeki deneyler optimum veya optimuma yakın birçok çözümün verimli bir şekilde elde edilebileceğini gösterdi.

Bolat'ın [21] ve [22] deki amacına benzer şekilde Zheng vd. [26] da tüm boş zamanların varyansının en aza indirilmesini önermişlerdir. Modeli daha rahat çözmek için amaç fonksiyonunun en aza indirmesinin yanı sıra bir ilk çözüm sağlaması için uçuşları ayırabilen bir meta sezgisel yöntem ortaya konulmuştur. Daha sonra UTKAP'ı en iyilemek için bir tabu araştırma algoritması kullanmışlardır. Deneyler gerçek hayattan bir dizi test verileri kullanılarak yapılmış ve sonuçlar kullanılan algoritmanın ve modelin önceki bir metotla kıyaslandığında etkili olduğunu göstermiştir.

Diğer bir sağlam havaalanı kapı atamasını da Lim ve Wang [24] te önerdiler. SHKA uçuşların gerçek zamanlı varış ve kalkış saati hakkında önceden herhangi bir bilgiye sahip olmadan denetimsiz tahmin fonksiyonlarını ortaya koyarak ikili bir programlama modeline dönüşen stokastik programlama modelidir. Bu tür problemleri çözmek için

yazarlar tabu arama ve yerel arama yöntemleri birleştirerek melez meta-sezgisel arama geliştirdiler.

Ayrıca Yang ve Hu [25] de kapı atama sağlamlığını artırmak amacıyla her uçuşun varış ve kalkış saatini tahmin eden ve sonra tahmin sonuçlarına göre kapılara uçuşları atayan bir strateji önerdiler. Geleneksel GM (1,1) modeli geliştirmek için hareketli bir operatör sunulmuş ve gelişmiş GM(1,1) modeli uçuşun geliş zamanını ve kapıyı meşgul ettiği zamanı iyi tahmin etmiştir. Daha sonra 0-1 Programlama modeli önerilmiştir. Amacı havaalanı terminal alanı içinde yolcu yürüme mesafelerini en aza indirmek olan model için sezgisel bir algoritma tasarlanmıştır. Test sonuçları tahmin sonuçlarına dayalı kapı atamasının uçuş planına dayalı olan atamadan sağlamlık ve yolcu memnuniyeti açısından daha iyi sonuç verdiğini ortaya koymuştur.

Obata [8] de bağlantılı yolculuk yapan yolcuların gelen uçuş kapısından gidecek uçuşun kapısına yürüme mesafelerini en aza indirmeyi amaçladığı çalışmada kapı atama problemini bir NP-zor kuadratik atama problemi (KAP) olarak formüle etti. O zamandan beri, birçok araştırmacı bu problemi inceledi ve uçuş kapı atama problemlerini çözmek için değişik teknikler önerdiler [9], [10], [17].

Bağlantılı yolculuk yapan yolcuların bağlantı sürelerinin minimize edilmesinin amaçlandığı çalışmaları [9] da Xu ve Bailey kuadratik olarak ifade ettikleri problemi doğrusal amaç fonksiyonu ve doğrusal kısıt fonksiyonları olan 0-1 tamsayılı problem olarak tekrar formüle etmişlerdir. Basit bir tabu araştırması sezgiseli tasarlanarak problem çözülmüştür. Sonuç olarak oluşturulan tabu araştırması sezgiseli yolcu bağlantı sürelerinde uygulanan mevcut yöntemle göre kazanç sağlamıştır.

Haghani ve Chen [10] da çok zaman aralıklı kapı atama problemini, tamsayılı programlama ile formüle etmişlerdir. Problemin çözümü için dal-sınır tekniği kullanılmış ve optimum çözümler elde edilmiştir. Sunulan yeni bir sezgisel yaklaşımla elde edilen sonuçlar ve optimal çözümler karşılaştırılmıştır. Sezgisel yaklaşımla bulunan sonuçlar şimdiye kadar elde edilmiş en iyi sonuçlarla karşılaştırıldığında kapı atama problemine etkin bir şekilde çözüm sağladığı belirlenmiştir.

Ding vd. [16] ve [17] de uçuş sayısının mevcut kapı sayısını aştığı UTKAP üzerine çalışmışlar ve Xu ve Bailey'in [9] daki çalışmasından daha iyi yeni bir komşuluk arama tekniği geliştirmişlerdir. Toplam yürüme mesafesinin ve açık park alanına park eden uçak sayısının minimize edilmesinin amaçlandığı [16] da Ding vd. problemin çözümü için doyumsuz (greedy) algoritma kullanmışlardır. Ardından yeni bir komşuluk arama tekniğine sahip tabu araştırması algoritması geliştirilmiş, elde edilen sonuçlar Ding vd.'nin [18] de tabu araştırma algoritmasıyla buldukları sonuçlarla karşılaştırılmıştır. Yeni yöntemin çok daha iyi sonuçlar verdiği belirlenmiştir. Ding vd. [17] de yaptıkları bir diğer çalışmada biri ısıtma işlemi, diğeri ısıtma işlemi ile tabu araştırması yaklaşımlarının bir arada kullanıldığı iki algoritma geliştirmişlerdir. Sonuçlara göre ısıtma işlemi algoritması önceden çalışılmış olan tabu araştırması algoritması kadar iyi sonuç vermezken, tabu araştırması ve ısıtma işlemi algoritmalarının bir arada kullanıldığı algoritma her iki yönetime göre daha iyi sonuçlara ulaşılmasını sağlamıştır.

Hem Dorndorf vd. [19] da hem de Drexler ve Nikulin [20] daki çalışmalarında başlangıçta Ding vd. tarafından [16] ve [18] de önerilen çok amaçlı UTKAP'ı ele aldılar. Dorndorf vd. [19] da terminal kapılarına atanmayan uçuş sayılarını ve toplam yolcu yürüme mesafesini en aza indirmeyi göz önünde bulundurmanın yanı sıra çekme(towing) işlemlerinin ve referans kapısı atama sapmalarının en aza indirilmesi ve buna ek olarak atama tercihlerinin en üst seviyeye çıkarılmasını da içeren iki amaç fonksiyonu formüle ettiler. Drexler ve Nikulin [20] de aynı amaçları kullanarak UTKAP'ı KAP olarak modellediler ve problemi etkin ve verimli bir şekilde çözmek için bir PTB sezgiseli önerdiler.

Yukarıdaki çalışmalara benzer amaçlar (toplam yürüme mesafesini en aza indirmek ve sağlam kapı ataması) içeren Wei ve Liu [27], [28] de çok amaçlı modeller sundular. Bu iki çalışmaya benzer şekilde Hu ve Di Paolo [29], [30] da UTKAP'a genetik algoritma tabanlı çözümler önermişlerdir. Wei ve Liu [27] de hem yolcular hem de havaalanını göz önüne alarak toplam yürüme mesafesini en aza indirmek ve sağlam kapı atamasını (boş zaman periyotlarının dağılımını en aza indirmeyi) amaçlamışlardır. UTKAP'ı çözmek için tabu yöntemini ve meta sezgiseli entegre eden bir genetik algoritma tasarlanmıştır. Çok amaçlı problemi daha rahat çözmek için ikinci amaç tabu yöntemi ile bir kısıta dönüştürülmüştür. Gerçek zaman verileri hesaplama sonuçları modelin

doğru olduğunu ve algoritmalarının verimli olduğunu göstermiştir. Aynı amaçlara sahip Wei ve Liu [28] de ise uçuş kapı atamasının boşta zamanları bulanık değişkenler olarak kabul ettiler ve bu değişkenlerin üyelik dereceleri atamanın sağlamlığına olan etkiyi ifade etmek için kullanıldı. İki amacı bir amaca dönüştürmek için üyelik derecesini ayarlama fonksiyonu ortaya koyuldu ve NP-zor problemi optimize etmek için değiştirilmiş genetik algoritma kullanıldı. Simülasyon sonuçları önerilen bulanık yöntemin fizibilite ve etkinliğini gösterdi. Hu ve Di Paolo [29] da havaalanı terminalinde çok amaçlı havaalanı kapı atama problemini çözmek için tek tip bir çaprazlama ile etkili GA tasarımını hedeflediler. Mevcut GA' larda UTKAP için sıkça kullanılan uçakların kapılardaki kuyruktaki ilgili duruşları yerine yeni GA kromozomları oluşturmak için uçaklar arasındaki ilgili duruşlar kullanıldı. Sonra evrim süreci içinde çeşitlilik ve yakınsama arasında iyi bir denge tutmada etkili olan yüksek verimli tek tip çaprazlama operatörü tasarlanmış ve yeni GA' nın avantajları ayrıntılı simülasyon testleriyle kanıtlanmıştır. Önceki çalışmalarından sonra Hu ve Di Paolo [30] da melez bir GA planı ortaya koydu. Orijinal UTKAP kapılarda durmak için bekleyen uçakların parametrelili bir uzayda noktalar olarak yansıtıldığı bir ön problem haline dönüştürüldü ve kapılarda bazı uçak kuyrukları oluşturmak için ilgili uzaysal parametrelerin tüm uçak noktalarını birleştirmek için girdi olarak kullanıldığı doğal dalgalanma-yayılma olgusundan esinlenilmiş bir deterministik model geliştirildi. Önerilen dalgalanma-yayılmalı GA'nın UTKAP üzerine etkinliği simülasyon sonuçlarıyla ortaya konuldu.

Bazı modeller stokastik uçuş gecikmelerini (erken veya geç gelişler ve geç kalkışları içeren) dikkate alarak statik kapı atamasının performansını arttırmayı denediler. Örneğin, Hassounah ve Steuart [11] de planlı tampon sürelerinin uçuş programının dakiklığını arttırabileceğini gösterdiler. Yan ve Chang [12] de ve Yan ve Huo [13] te statik kapı ataması problemlerinde stokastik uçuş gecikmelerini absorbe etmek için aynı kapiya atanan iki art arta uçak arasında sabit tampon süre kullanmışlardır. Yan ve Chang [12] de çoklu emtialı (multi commodity) ağ akış modeli geliştirdiler. Ayrıca Mangoubi ve Mathaisel'in [3] teki çalışmalarıyla karşılaştırıldığında çok daha etkili olduğu görünen alt gradient optimizasyonu ile lagrange gevşetme metodu kullandılar ve UTKAP'ı çözmek için bazı sezgisellere de yer verdiler. Yan ve Huo [13] te uçak pozisyonu tahsisi için iki amaçlı 0-1 tamsayılı programlama modeli formüle ettiler. İlk

amaçla yolcu yürüme süresini en aza indirmeye çalışırken ikinci amaçla yolcu bekleme sürelerini en aza indirmeyi hedeflediler.

Yan ve Tang [15] te stokastik uçuş gecikmelerini göz önünde bulunduran havaalanı kapısı atamaları için sezgisel bir yaklaşım geliştirdiler. Problemi stokastik uçuş gecikmeleri altında etkin bir şekilde entegre çözebilen bir çerçeve içine gömülü sezgisel bir yaklaşım geliştirdiler. Bu çerçeveyi değerlendirmek için Tayvan'daki bir uluslararası havalimanı operasyonu verileri kullanılarak sayısal testler yapıldı ve bu çerçevenin stokastik uçuş gecikme durumlarında manüel atama süreci üzerinde önemli iyileştirmelere imkan verdiği görüldü.

Gelen bir uçağın kalkışı geciktiğinde havaalanı kapılarına yeniden atama problemi (HKYAP) ortaya çıkar. Eğer gecikme atanan kapıya sonradan gelen uçakların gelişini geciktirecek kadar önemli ise yönetimin ekstra gecikme sürelerini en aza indirmek için kapı atamasını değiştirmesi gerekir. Aşağıdaki çalışmalar HKYAP' ni çözmek için bazı yaklaşımlar sunmaktadırlar.

Gu ve Chung [31] de minimum ekstra gecikme zamanlı çizelgeleri en az deneyimli kapı yöneticileri tarafından üretilen çözümler kadar verimli bir şekilde hesaplayan etkili bir genetik algoritma önerdiler.

Tang [32] de (ihlal kavramını dahil eden) kapı yeniden atamaları için geçici kapı sıkıntılarının ve stokastik uçuş gecikmelerinin olduğu bir model (SKYAM) geliştirdi. Geçmişte bu konulardaki çalışmalarda bir eksiklik olmuş ve en iyi bilindiği üzere önerilen model teorik ve pratik bakış açısından kapı yeniden atama problemlerine temel olabilecek ilk çalışmalardan biridir. TTY Havaalanı operasyonlarına dayanan sayısal test sonuçlarının özellikleri ve elde edilen bulgular tartışıldığında planlamacılar ve operatörler için yararlı bir başvuru malzemesi olarak hizmet verebileceği sonucuna varılmıştır.

Yan vd. [33] te havaalanı yetkililerinin geçici havaalanı kapalılıklarından sonraki uçuşları kapılara daha verimli ve etkin bir şekilde yeniden atamalarına yardım etmek için bir kapı yeniden atama modeli geliştirdiler. Model, ilgili operasyon kısıtlarına bağlı toplam kapı değişiklikleri sayısını en aza indirmek amacı ile tasarlanmıştır. Model güçlü bir temel ağ yapısına sahiptir ve verimli bir şekilde ağ tabanlı algoritmalar kullanılarak

özöllebilir (örneğın ağ simpleks yöntemi ile CPLEX' te alıřtırılan dal ve sınır algoritması). Ağ modelini deęerlendirmek için CKS Havaalanı operasyon verilerini temel alan bir vaka alıřması yapıldı. Test sonuçları ağ modelinin mevcut manüel yönteme kıyasla önemli bir gelişme sunduęunu gösterdi. Özellikle beř büyük ölçekli problem örneğinden elde edilen test sonuçları önerilen modelin büyük ölçekli sorunların özümündeki etkinlięini göstermektedir.

Tang vd. [34] te eřitli uuř gecikme bilgileri doęrultusunda, tekrarlanan kapı yeniden atamalarını ele almak için tasarlanmış bir kapı yeniden atama erevesi, sistematik bir bilgisayar aracı, geliřtirdiler. Pratik süreç gereklilięi nedeniyle yeniden işleme ve yeniden atama evreleri CPLEX 10.0 matematiksel programlama özücüsü ile birleşmiş C bilgisayar dili kullanılarak birbirine bağlanmışır. Gerçek zamanlı uuř gecikmeleri için kapılara yeniden atama erevesini (GRFRTFD) deęerlendirmek için Tayvan'daki bir uluslararası havalimanının operasyonlarından elde edilen verilerin sayısal testleri yapıldı ve sonuçlar yöntemin zaman alıcı olmadığını ve gerçek zamanlı operasyonlara uygulanabileceğini gösterdi.

Klasik yöneylem araştırması tekniklerini kullanan geleneksel yaklaşımlar belirsiz bilgi ve birden fazla performans kriterleri ile zorluk yařarken ve gerçek zamanlı operasyonların destek ihtiyaçlarına iyi uyum sağlayamazken ikinci temel araştırma akımı simölasyon ve kural tabanlı uzman sistemlerin inřası üzerine yoğunlaşmaktadır [36]. Alternatif olarak birçok yazar kural tabanlı uzman sistem tasarımı konusuna odaklandılar. Bir uzman sistem yer kontrolörlerinden elde edilen bilgiye dayanarak atamaları üretmek için üretim kurallarını kullanır.

Pratik açıdan bakıldığında, matematiksel programlama tekniklerini (dal ve sınır, dinamik programlama, yerel arama) içeren basit uzman sistemler geliřtirmek daha önemlidir. Böyle bir birleşme, istenen esneklik özellięine sahip bir kapı planlaması sistemi oluşturmak için yardımcı olacaktır. Optimizasyon ve kural temelli yaklaşımların her ikisi de simölasyon analizleri ile kombine edilebilir.

Hamzwawi [35] te uuřların kapılara atanmasını simüle etmek için ve kapı kullanımıyla ilgili belirli kuralların etkilerini deęerlendirmek için kural tabanlı bir sistem ortaya koydu. Cheng [39] da ok amaçlı kısmi paralel atamaları sağlamak için matematiksel

programlama tekniklerinin bilgi tabanlı bir kapı atama sistemine entegrasyonunu ortaya koydu. Bu bir yandan kriterlerin önem sırasını değiştirmek için havalimanı otoritesine daha fazla kontrol verirken diğer yandan uygulanabilir bir çözüm aramayı amaçlayan matematiksel bir modelin kullanıldığı süreçtir. Cheng benzer şekilde [40] ve [41] de de kural tabanlı sistemler ortaya koymuştur. Cheng [40] ta bilgi tabanlı teknolojilerin bir simülatör içinde daha derin yapılandırılabilirdiği ve simülasyonun tamamen bir bilgi tabanlı sisteme gömülü olduğu bir çeşit yeni model üretmek için nesne yönelimli simülasyonu ve bilgi tabanlı sistemleri bir araya getirmeye çalışmıştır. Cheng [41] de ağdaki tüm düğümlere olan en uzun yolu hesaplanmak suretiyle yapılan daha etkili bir simülasyon için ağ tabanlı bir metot önermiş ve benzer metotlarla karşılaştırıldığında ortalama simülasyon performans değerlerinden çok daha iyi sonuçlar verdiğini ortaya koymuştur.

Bilindiği gibi bir uzman sistem yapılandırılmış bilgi-tabanlı bir sistemdir. Uçak kapı atama prosedürü havaalanındaki deneyimli uzmanlar tarafından yapılan mantık temelli kapı atama prosedürünün simüle edilmesidir. Optimal bir çözüm bulmak yerine, karmaşık büyük ölçekli problemler için uygulanabilir ve tatmin edici bir çözüm bulmak için kullanılır. Gosling [36] da, Muthukrishnan ve Srihari [37] de ve Jo vd. [38] de uzman sistemler kullanarak kapı atama problemi üzerine çalıştılar. Gosling [36] da Denver Stapleton havaalanının önemli bir merkezinde bir kapı ataması için uygulanmış uzman bir sistem ortaya koymuştur. Muthukrishnan ve Srihari [37] de UTKAP'ı çözmek için benzer bir yaklaşım kullanmış ve ayrıca duyarlılık analizinin nasıl uygulanacağını açıklamışlardır.

Jo vd. [38] de CHIP'de rampa aktivitesi koordinasyonu uzman sistemini (RACES) geliştirdi. RACES uçak kapı planlama problemini bir uzmanın problem çözme prosedürüne benzer yöntemler kullanarak çözebilen, sistem planlama yaptığı zaman bir etki alanı filtreleme tekniği ile arama alanını daraltabilen bir sistem olarak geliştirildi. RACES değişimli bir sezgisel çizelgeleme yöntemi kullanarak kullanıcı odaklı optimal bir program üretti. Sistemin doğruluğunu test etmek için gerçek bir havayolu şirketinin günlük operasyonel verilerine yöntem uygulandı, sonuçlar analiz edildi ve alan uzmanları tarafından onaylandı. Ayrıca, bu sistem, Seul Kimpo havaalanında Korean Air tarafından uygulanmıştır.

Simülasyona dayalı diğer bir çalışma da Yan vd. tarafından sunulan [14] tür. Çoğu araştırma statik kapı atama performansını geliştirmeye odaklanırken simülasyonun, statik kapı ataması ve atamayı etkileyebilecek faktörler arasındaki ilişkiyi analiz etmek için kullanılabileceğini göstermişlerdir. Yan vd. [14] te havaalanı yöneticilerine/işletmecilerine yardımcı olacak bir simülasyon çatısı sunmuşlardır. Bu yöntemle rastgele uçuş gecikmelerinin statik kapı atamaları üzerindeki etkileri incelenmiştir ve rastgele tampon süreler ile gerçek zamanlı kapı atama kuralları değerlendirilmiştir. Tayvan'daki Chiang Kai-Shek Havaalanına ait verilerin kullanıldığı deneysel çalışmada oluşturulan simülasyon çatısının diğer havaalanları tarafından kendi özelliklerine bağlı olarak kullanılabileceği belirtilmiştir.

UÇAKLARIN TERMİNAL KAPILARINA ATANMASI PROBLEMİNE BİR ÇÖZÜM YAKLAŞIMI ÖNERİSİ

Bu çalışmada amacı kapılara atanmamış uçuşları ve kapılar arası toplam yürüme mesafesini (transfer yolcuların yürüme mesafesi ve gelen ve giden yolcuların giriş/çıkışlara yürüme mesafelerini) en aza indirmek olan, tüm uçuşların kapılara atanamayacağı durumu ifade eden (over constrained airport gate assignment problem) aşımış havaalanı kapı ataması problemi (AUTKAP) üzerine çalışılmıştır. Kapılara atanması gerçekleştirilemeyen uçaklar apron adı verilen açık park alanlarına yönlendirilecek ve yolcular otobüslerle terminal alanına taşınacaklardır.

Problem ikili bir kuadratik programlama problemi olarak formüle edildi. Tanımladığımız problem için iki aşamalı bir çözüm yaklaşımı önerdik. İlk aşamada kapılara atanmayan uçuşları en aza indirecek bir doyumsuz (greedy) algoritma kullandık [16]. Böylece uçuş sayısının kapı sayısını aştığı durumu ortadan kaldırmış olduk. İkinci kısımda toplam yürüme mesafesini en aza indirmek için Wu ve Ji'nin [53] teki genetik algoritma metasezgisel yaklaşımından faydalanıldı ve oldukça başarılı sonuçlar veren genetik algoritma parametrelerinin özel bir kuadratik atama problemi olan UTKAP'da ne kadar etkili olacağı araştırıldı.

Ayrıca uçuşların kapılara atanması işleminde test verileri için tüm kapıların tüm uçuşlara hizmet verebilecek yapıda olduğunu, gerçek veriler için ise bu durumun minimizasyon aşamasında geçerli olacağını ve ilk atama kısmı için mümkün olmadığını

kabul ettik. Son olarak tüm uçuşların geliş-gidişlerinin çizelgelere uygun olduğunu kabul ettik.

Durma zamanlarından ve varış ve kalkış saatleri arasındaki diğer tampon sürelerden kaynaklanan ikincil kısıtlar, uçuş varış ve kalkış sürelerinin genişletilmesiyle kolayca çözülebileceği için modelde göze alınmamalarına karar verildi. Gerçek verilerde uçuş sürelerinin ayarlanmasıyla tampon süreler de modele dahil edildi.

AUTKAP'ı tanımlamak için 2. bölümde yer verdiğimiz Ding vd.'nin [16] daki modelini dikkate alarak önerdiğimiz modelin farklılıklarını açıklayalım. Doyumsuz (greedy) atama ve GA kullanarak yolcu yürüme mesafesinin en küçüklenmesi aşamalarını birbirinden bağımsız olarak düşündük ve ilk aşamada aprona atanan uçuşları en aza indirdik. Böylelikle uçuşların kapıları aştığı aşımış durum egale edildiği için ikinci amaç fonksiyonunda sadece kapılara atanan uçuşları değerlendirmeye almış olduk. Ding vd.'nin [16] daki modelinde yer alan $m+1$. kapı olarak kabul edilen apron bizim modelimizde ikinci amaç fonksiyonunda yer almayacaktır. Bu değişiklik sonucunda iki durum ortaya çıkacaktır.

1. Ding vd. [16] da yolcu yürüme mesafesini minimize etmek için doymusuz (greedy) algoritmadan sonra aprona atanan uçakları tekrar değerlendirmeye almakta ve apron-kapı arası uçak değişimin öngörmektedir. Önerdiğimiz modelde apron-kapı arası uçuş değişimi yapılmayacaktır ve toplam yolcu yürüme mesafesi sadece kapılara atanan uçuşlar arasındaki değişimlerle en aza indirilecektir.

2. Modelde aprondan kapılara olan yürüme mesafeleri dikkate alınmayacaktır çünkü aprona atanan uçuşlardaki yolcuların terminale otobüslerle taşındığı kabul edilecektir.

Bu değişiklikler sonucunda problemi iki bağımsız model halinde ele aldık. İlk modelden elde edilen atanmış uçuşlar ikinci modeldeki uçuşları oluşturacak ve böylece kapılara atanmış uçuşlar ele alınacak, aprona atanan uçuşlar ele alınmayacaktır.

Böylelikle ilk amaç fonksiyonu aynı kalacak;

N: havaalanına gelen veya havaalanından giden uçuşların kümesi.

M: havaalanındaki uygun kapı kümesi.

n : toplam uçuş sayısı.

m : toplam kapı sayısı.

a_i : i uçuşunun geliş zamanı. ($1 \leq i \leq n$)

d_i : i uçuşunun kalkış zamanı. ($1 \leq i \leq n$)

$w_{k,l}$: k kapısından l kapısına giden yolcuların yürüme mesafesi. ($1 \leq k, l \leq m$)

$f_{i,j}$: i uçuşundan j uçuşuna transfer yapan yolcu sayısı. ($1 \leq i, j \leq n$)

$y_{i,k}$: Eğer i uçuşu k kapısına atanmışsa 1 değerini alır ($0 \leq k \leq m+1$), diğer durumda ise 0 değerini alır.

$$\min \sum_{i=1}^n y_{i,m+1} \quad (3.1)$$

Şu kısıtlar altında;

$$\sum_{k=1}^{m+1} y_{i,k} = 1 \quad (1 \leq i \leq n) \quad (3.2)$$

$$a_i \leq d_i \quad (1 \leq i \leq n) \quad (3.3)$$

$$y_{i,k} y_{j,k} (d_j - a_i) (d_i - a_j) \leq 0 \quad (1 \leq i, j \leq n, k \neq m+1) \quad (3.4)$$

$$y_{i,k} \in \{1,0\} \quad (1 \leq i \leq n, 1 \leq k \leq m+1) \quad (3.5)$$

Modelin ikinci amaç fonksiyonu şu şekilde hesaplanacaktır;

n =toplam uçuş sayısı

a =aprona atanan toplam uçuş sayısı

t =kapılara atanan toplam uçuş sayısı olmak üzere,

$t=(n-a)$ şeklinde ifade edilir.

a_i : i uçuşunun geliş zamanı. ($1 \leq i \leq t$)

d_i : i uçuşunun kalkış zamanı. ($1 \leq i \leq t$)

$w_{k,l}$: k kapısından l kapısına giden yolcuların yürüme mesafesi. ($1 \leq k, l \leq m$)

$f_{i,j}$: i uçuşundan j uçuşuna transfer yapan yolcu sayısı. ($1 \leq i, j \leq t$)

$$\min \sum_{i=1}^t \sum_{j=1}^t \sum_{k=1}^m \sum_{l=1}^m f_{i,j} w_{k,l} y_{i,k} y_{j,l} + \sum_{i=1}^t f_{0,i} w_{0,i} + \sum_{i=1}^t f_{i,0} w_{i,0} \quad (3.6)$$

Aşağıdaki kısıtlar altında:

$$\sum_{k=1}^m y_{i,k} = 1 \quad (1 \leq i \leq t) \quad (3.7)$$

$$a_i \leq d_i \quad (1 \leq i \leq t) \quad (3.8)$$

$$y_{i,k} y_{j,k} (d_j - a_i) (d_i - a_j) \leq 0 \quad (1 \leq i, j \leq t, k \neq m+1) \quad (3.9)$$

$$y_{i,k} \in \{0,1\} \quad (1 \leq i \leq t, 1 \leq k \leq m) \quad (3.10)$$

3.1 Doyumsuz (Greedy) Algoritma Aşaması

Doyumsuz algoritmanın ana fikri şudur:

Tüm uçuşları ayrılış zamanlarına göre sıraladıktan sonra (artan sıralama), uçuşlar kapılara birer birer atanır. Eğer uygun kapı yoksa uçuş aprona atanır.

Algoritmanın yapısı şu şekildedir:

1. Uçuşları ayrılış zamanlarına (d_i) göre sıralayın ($1 \leq i \leq n$)

g_k ($1 \leq k \leq m$) k kapısının uygun olma zamanı (k kapısından ayrılan son uçuşun ayrılış zamanı) olsun.

2. Tüm k' lar için $g_k = -1$ ayarlayın.

3. Her i uçuşu için;

(a) bir k kapısı bulun öyle ki $g_k < a_i$ olsun ve g_k maksimum olsun.

(b) eğer böyle bir k kapısı varsa uçuş i'yi k kapısına atayın. $g_k = d_i$ olarak güncelleyin.

(c) eğer böyle bir k kapısı yoksa i uçuşunu aprona atayın.

4. Sonuçları elde edin.

Doyumsuz (Greedy) algoritmanın optimum sonuçlar verdiği Ding vd. tarafından [16] da ortaya konmuştur.

Bu yöntemin anlaşılabilmesi için 8 uçuş,2 kapılı bir örnek üzerinden yöntemi açıklayalım. Çizelge 3.1’de görüldüğü üzere, örnek uygulamada 8 uçuş ve bu uçuşların geliş ve gidiş zamanları geliş zamanlarına göre sıralı halde verilmiştir.

Çizelge 3.1 Örnek uçuş çizelgesi

Uçuşlar	Geliş Zamanı (dk)	Gidiş zamanı (dk)
1	3	7
2	4	8
3	3	12
4	7	10
5	7	16
6	8	14
7	11	18
8	9	14

Çizelge 3.2 Uçuş kalkış zamanlarına göre uçuşların küçükten büyüğe sıralanmış hali

Uçuşlar	Geliş Zamanı (dk)	Gidiş zamanı (dk)
1	3	7
2	4	8
4	7	10
3	3	12
6	8	14
8	9	14
5	7	16
7	11	18

Çizelge 3.2’ de uçuşların ayrılış zamanlarına göre sıralandığı hali görülmektedir. Bu sıralamaya göre uçuşların atanmasının bitmiş hali Çizelge 3.3’ te Gantt diyagramı şeklinde gösterilmiştir. Bu örnekte 3,5 ve 6 nolu uçuşlar aprona atanmıştır.

Çizelge 3.3 Doyumsuz algoritma ile kapılara atanan uçuşlar

	Zaman																	
Kapı	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1				1				4				7						
2					2				8									

Aynı uçuşların FCFS (First Come First Serve) yani ilk gelen ilk hizmet alır kuralına göre atandığında oluşan durum Çizelge 3.4’ teki gibidir. Bu kural uçuşların geliş saatlerine göre sıralanarak atanmasını ifade eder. Atanamayan uçuşlar 2,5,6, ve 8’dir. Doyumsuz algoritmanın daha başarılı atama yaptığı anlaşılmaktadır.

Çizelge 3.4 FCFS kuralına göre atanan uçuşlar

	Zaman																	
Kapı	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1				1				4				7						
2					3													

3.2 Genetik Algoritma

Önceki bölümde görüldüğü gibi UTKAP veya türevleri için yapılmış çalışmaların çoğunda problem 0-1 tamsayı programlama problemi olarak ya da kural tabanlı bir sistem bağlamında formüle edilmiştir. Biz bu çalışmada NP-zor UTKAP için bir metasezgisel yaklaşıma yer vereceğiz.

3.2.1 Genetik Algoritma’nın Tanımı

Genetik Algoritma, canlı sistemlerdeki genetik şifre mantığı kullanılarak sezgisel olarak en iyi çözüm veya en iyi çözüme yakın olabilecek bir sonuç bulmayı hedefler. Bu mantık, doğal seçime yani güçlü bireyin hayatta kalma olasılığının yüksek olmasına dayanır. Bu yöntemle evrim sonucu hayatta kalan birey en iyi sonuç olarak alınır [46].

GA sezgisel bir yöntemdir. Doğadaki bireylerin birleşerek çocukları oluşturması ve bu çocuklardan güçlü olanın hayatta kalması GA’ nın temelini oluşturur [47]. GA, bir bireyler topluluğuyla başlayarak bu bireylerden çocuklar elde eder. Bu bireylerin özellikleri ve sayısı rassal olarak belirlenir. Bu bireyler de rassal olarak eşleştirilerek

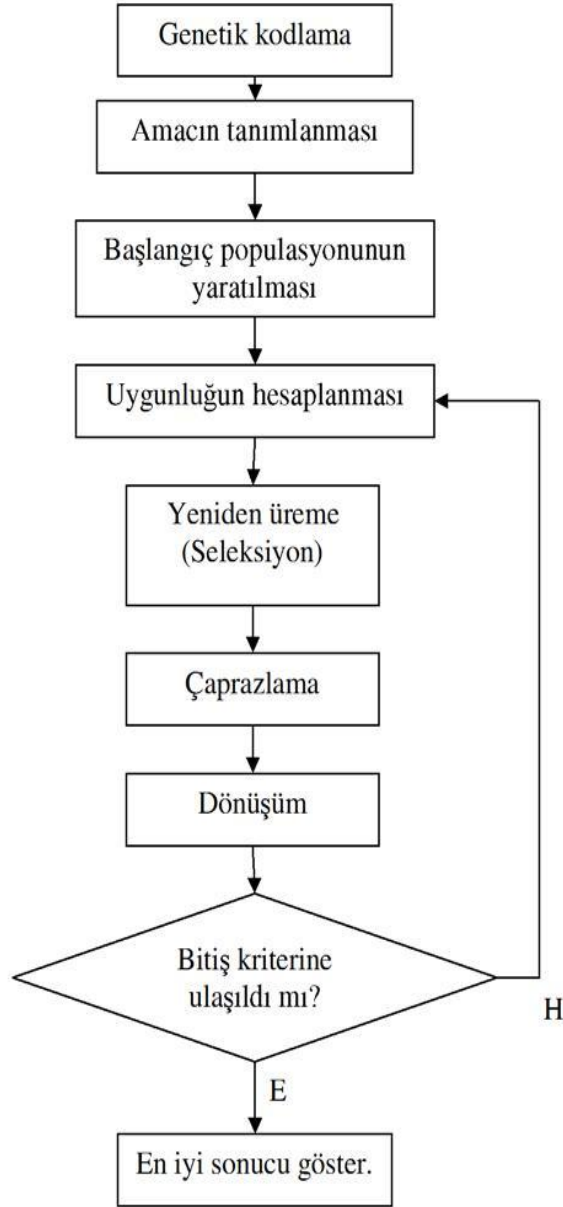
ebeveyn olurlar. Bu ebeveynlerden elde edilen çocuklar seleksiyon, çaprazlama ve dönüşüm işlemlerine maruz kalırlar. Bu işlemlerin tekrarlanması ile evrim gerçekleştirilir. Genetik algoritmanın bitirilmesi bitiş ölçütlerinin gerçekleşip gerçekleşmediğine bağlıdır. Aşağıdaki tabloda Genetik Algoritma ile ilgili başlıca terimler yer almaktadır.

Çizelge 3.5 Genetik algoritma ile ilgili tanımlar

Terim	GA Karşılığı	Tanım
Gen	Karakter özelliği	Bit olarak da tanımlanır. Bireyin karakterini dolayısıyla da değerini belirleyen öğedir.
Kromozom	Birey	Dizi olarak tanımlanır. Genlerin birleşerek oluşturduğu dizidir. Kromozomlar belli bir kodlama sistemiyle oluşturulurlar. Aday çözümleri gösterir.
Genotip	Bireyin gen yapısı	Aday çözüm olan kromozomların içindeki belirli gen gruplarına genotip denir.
Fenotip	Deşifre edilmiş yapı	Alternatif çözüm kümesidir. Genotipin deşifre edilerek asıl değer belirlenmiş halidir.
Allel	Özellik değeri	Genlerin kaç değişik değer alabiliyorsa, bu değerlerin kümesidir.
Lokus	Karakterin yeri	Genin kromozomdaki pozisyonu (yeri)dir.
Popülasyon	Aday çözümler topluluğu	Kromozomların bir araya gelerek oluşturduğu çözüm topluluğudur. Genellikle algoritma boyunca sabit tutulur. Bu sayı çoğunlukla 10 ile 100 arasında değişir.
Uygunluk Fonksiyonu	Amaç fonksiyonu	Bireyin hayatta kalma durumunu yani uyumunu gösteren değerdir. Yüksek değerler bireyin hayatta kalma olasılığının daha yüksek olduğunu belirtir.
Seleksiyon	Seçim	Popülasyon içinden hayatta kalabilen canlıların seçilmesidir. Seçim için uygulanan yöntemler açıklanacaktır.

3.2.2 Genetik Algoritmanın Adımları

Herhangi bir problemin çözümünde kullanılan basit bir GA'nın işleyişi Şekil 3.1' de verilmiştir [47].



Şekil 3.1 Genetik algoritmanın adımları

3.2.2.1 Genetik Kodlama

Genetik kodlama işlemi fenotipin genotipe dönüştürülme işlemidir. Bir popülasyon kromozom topluluğu, kromozom ise genler topluluğu olarak açıklanır. Kromozomun yapısı ise aşağıda verilmiştir.

$$S = S_1 S_2 \dots S_j \dots S_n$$

Burada s_i ($i = 1, 2, \dots, n$) genleri göstermektedir. i değeri genin lokusudur. Herhangi bir i geninin alabileceği değişik değerler allel olarak adlandırılır. Bir kromozomun uzunluğu

(n) genellikle sabit tutulur [48]. Literatürde en yaygın kullanılan kodlama ikili düzende kodlamadır. Bu kodlamada, her dizi 0 ve 1 değerlerinden oluşmaktadır. İkili düzende kodlama çok sık kullanılan bir kodlama olmasına rağmen bazı sakıncaları vardır. Örneğin, çok değişkenli bir fonksiyonun en iyilenmesi için, değişkenlerin alt ve üst sınırlarına bağlı olarak elde edilen dizi uzunluğu çok büyük olabilmektedir. Aynı zamanda, gezgin satıcı, çizelgeleme, karesel atama gibi kombinatoriyal en iyileme problemlerinde, ikili düzende kodlama, arama uzayını tam olarak temsil edememektedir. Bu nedenle literatürde, gerçel sayı kodlama, permütasyon ve ağaç gösterimi kodlama türleri de çok sık kullanılmaktadır [49].

3.2.2.2 Amacın Tanımlanması

Elimizde çözüm kümesi olduğunda bu çözümlerin ne kadar iyi olduğunu belirlemek gerekir. Bunun için amaç fonksiyonu tanımlanmalıdır. Genetik Algoritma'nın bir avantajı da amaç fonksiyonunu formüle etmenin zor olduğu problemlere kullanım kolaylığı getirmesidir.

3.2.2.3 Başlangıç Popülasyonunun Yaratılması

Bu adım ile algortmada kullanılacak bireyler (ebeveynler) oluşturulur. Burada yapılacak seçim başlangıç popülasyonundaki birey sayısıdır. Az sayıda birey algortmada kullanılabilecek birçok değer in islenmeden atlanmasına neden olur. Bunun nedeni çaprazlamanın az sayıda birey arasında gerçekleşmesi ve çeşitliliğin daha az olmasıdır. Çok sayıda birey ise algortmanın yavaşlamasına neden olur. Burada her döngüde (nesil) daha çok birey incelenir. Popülasyon sayısı belli bir değeri geçtikten sonra daha fazla arttırmak algortmanın verimliliği azalmaktadır. Her problem tipi için en iyi çözümün başlangıç popülasyon sayısı farklıdır. Nesiller boyunca popülasyon sayısı genellikle korunur [47], [50].

3.2.2.4 Yeniden Üreme (Seleksiyon)

Yeniden üreme operatörü, doğadaki doğal seleksiyon işleminin GA tarafından uygulanmasıdır. Bu işlemle güçlü bireylerin hayatta kalması ve güçlü çocuklar oluşturması hedeflenir [48]. Burada öncelikle çocukları oluşturacak bireylerin eşleşmesi

gerekmektedir. Üreme için bireylerin seçilmesinin en basit yolu ise rassal seçimdir [46]. Bireyler seçildikten sonra yeniden üreme işlemine geçilir. Üreme için birçok yöntem bulunmaktadır.

Yeniden Üreme Yöntemleri

- Rassal Üreme: Var olan ebeveynlerden popülasyon sayısı kadar rassal seçim yapılır.
- Rulet Çarkı Yöntemi: En sık kullanılan yöntemlerden biridir. Bu yöntemle uygunluk değeri yüksek olan güçlü bireyin dogma olasılığının yüksek olması sağlanır.
- Beklenen Değer Yöntemi: Rulet çarkından daha hassas bir yöntem olarak kabul edilir. Bu yöntemde daha önce geliş olasılıkları hesaplanmış olan bireylerin beklenen değerleri bulunur. Beklenen değer, olasılıkla birey sayısının çarpımına eşittir.
- Boltzmann Yöntemi: Bu yöntem olasılıklar arasında orantı sağlamak için kullanılmaktadır ve hesaplama sonucu çıkan olasılıklar doğrusal olarak değil ivmeli olarak artmaktadır.
- Sıralı Seçim Yöntemi: Rulet çarkı yönteminde çok yüksek olasılıklı bir kromozoma karşı çeşitlilik yaratmak için kullanılır. Bu yöntemde kromozomlar uygunluk değerine göre azalacak şekilde sıralanır. Bu sıralamaya göre olasılıklar belirlenir ve rassal sayı atışı yapılarak yeni nesilde hangi bireylerin olacağı belirlenir [47].
- Turnuva Yöntemi: Bu yöntemde kromozomlar bir eşleşme havuzuna atarak eşleştirilirler. Bu eşleşmelerden galip çıkan birey güçlü birey olarak tanımlanır ve bir sonraki nesle geçmesi istenir. Bunun için bir p olasılığı belirlenir. p değeri $[0.5 - 1]$ aralığında seçilir. Turnuva yönteminde en iyi
 1. birey diğer nesle p olasılığıyla geçer.
 2. birey diğer nesle $p(1-p)$ olasılığıyla geçer.
 3. birey diğer nesle $p(1-p)^2$ olasılığıyla geçer...
 - n. birey diğer nesle $p(1-p)^{n-1}$ olasılığıyla geçer. Ayrıca, 2 turnuvadan fazlasını kazanan birey diğer nesle aktarılır [51].
- Elitizm Yöntemi: Bu yöntem en güçlü bireyin bir sonraki nesilde yer almasını garantilemek ve nesilleri güçlendirmek için uygulanır. Çaprazlama ve dönüşüm

sonrasında en güçlü birey yeni nesil arasında yer almamışsa, yeni nesildeki en güçsüz birey öldürülerek yerine bir önceki neslin en güçlü bireyi yerleştirilir [47].

- Denge Durumu Seçim Yöntemi: Bu yöntemde popülasyonlar arasından belirlenen birey(ler) diğer tüm nesillere aktarılır [52].

3.2.2.5 Çaprazlama

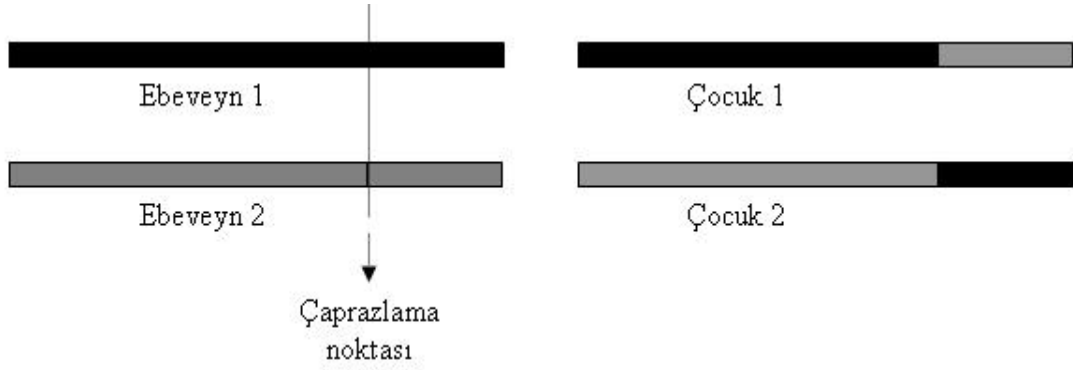
Çaprazlama genetik algoritma süresince kullanılan en önemli operatördür ve amacı popülasyonda olmayan bireyleri yaratarak bireyler arasındaki çeşitliliğin arttırılmasıdır [47 - 48]. Bu işlem ebeveynden çocuk oluşturma sürecinde yer alır. Problem çözümleri için birçok çaprazlama yöntemi üretilmiştir [48]. Genelde tüm çaprazlama operatörlerinde, iki birey bir çiftleşme havuzundan alınarak karşılıklı gen değiştirmesiyle gerçekleşir [52].

Çaprazlama operatörünün en önemli özelliği çaprazlama olasılığı (pc)' dir. Bu olasılık popülasyon içindeki herhangi bir bireyin çaprazlanma olasılığıdır. Bir popülasyon içindeki bireylerin beklenen çaprazlanma sayısı çaprazlama olasılığı ile popülasyondaki birey sayısının çarpılmasıyla elde edilir. Çaprazlama olasılığı genellikle 0.6 ile 1.0 arasında belirlenir. Her nesil için çaprazlama olasılığı değişebilir. Çaprazlama olasılığının küçük olması sonuca yavaş ulaşılmasına, büyük olması en iyi çözüm noktasının kaçırılmasına neden olabilir [47].

Genetik Algoritmada Kullanılan Çaprazlama Yöntemleri

En çok kullanılan çaprazlama yöntemleri aşağıda verilmiştir:

- Tek noktalı Çaprazlama: Çaprazlama noktası, dizi uzunluğuna eşit ya da daha küçük rassal bir sayı üretilerek belirlenir. Kromozomda, üretilen rassal sayıdan önce gelen bölümler değiştirilmeden korunurken, bu noktadan sonraki bölümler yer değiştirilerek yeni iki dizi elde edilir. Bir noktalı çaprazlama aşağıdaki gibi şematize edilebilir [50].



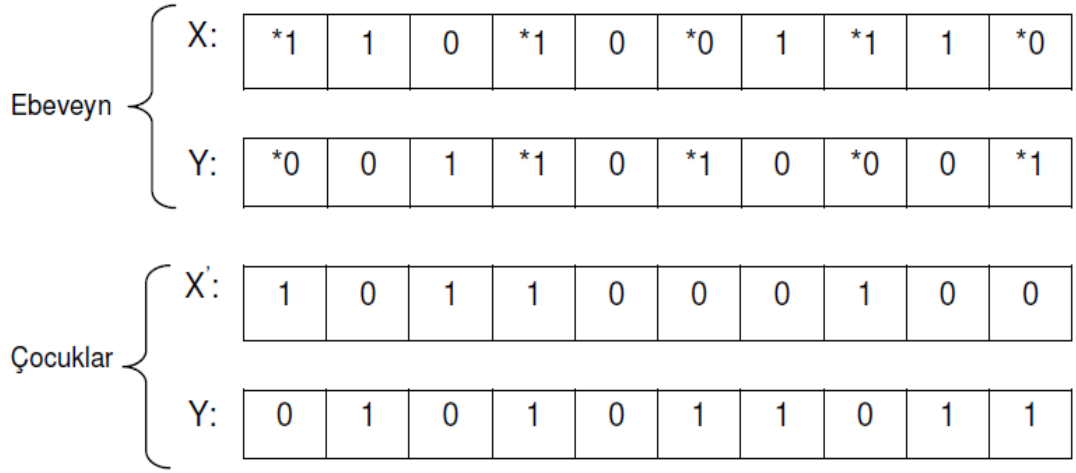
Şekil 3.2 Tek noktalı çaprazlama

- İki noktalı çaprazlama: İki noktalı çaprazlama operatöründe, diziler arasındaki gen değişimi, dizilerin rassal olarak seçilen ve birbiri ile aynı olmayan iki noktası arasında kalan bölümleri arasında gerçekleşir. 2 noktalı çaprazlama aşağıdaki gibi şematize edilir [50].



Şekil 3.3 İki noktalı çaprazlama

- Uniform Çaprazlama: İlk ebeveynin her geni 0.5 olasılıkla, ikinci ebeveynin karşı gelen geni ile yer değiştirilir. Şekil 3.4 'te uniform çaprazlamaya örnek verilmiştir. Bu örnekte, her gen için 0 ile 1 arasında üretilen rassal sayılar 0.2, 0.7, 0.9, 0.4, 0.6, 0.1, 0.8, 0.3, 0.7, 0.1'dir. Bir gen için üretilen olasılık değeri, 0.5' den az ise birinci çocuk birinci ebeveyninden kendisine karşılık gelen geni alırken, ikinci çocuk ise ikinci ebeveyninden kendisine karşılık gelen geni alır. Üretilen olasılık değeri 0,5'den az olan genler örnekte * ile işaretlenmiştir [50].



Şekil 3.4 Uniform çaprazlama

3.2.2.6 Dönüşüm (Mutasyon)

Mutasyon ise çaprazlamadan sonra üretilen her bir çocuğa uygulanan genetik operatördür. Rassal olarak seçilen bir gen, 0.001 gibi küçük bir olasılıkla değiştirilir. Mutasyon operatörü, arama uzayında taranmamış hiçbir noktanın kalmaması olasılığının gerçekleşmesine yardımcı olmak ve küçük oranda rassal arama sağlamak görevini üstlenmektedir. Şekil 3.5' te verilen örnekte, A dizisinin 3. geni mutasyona uğratarak A' dizisi elde edilmiştir [52].

A	1	1	1	1	0	0	0	0
A'	1	1	0	1	0	0	0	0

Şekil 3.5 Mutasyon örneği

3.2.2.7 Bitiş Ölçütlerine Ulaşılması

Genetik Algoritma'nın döngülerinin bitirilmesi (nesillerin sonlandırılması) için gerekli ölçütler farklı tanımlanabilir.

Bazı ölçütler şunlardır:

1. Belli bir nesil sayısı bitirildikten sonra döngüler durdurulur ve nesiller boyunca kimin en çok uygunluk gösterdiğine bakılır [46].

2. Evrim sonucunda uygunluk fonksiyonundaki iyileşme belli bir değerin altına düştüğünde döngüler bitirilir [46].
3. Evrim için belirli bir süre verilerek, bu süre bittiğinde evrim durdurulabilir [47].
4. Uygunluk fonksiyonu için belirli bir değer verilerek, bu değeri geçen ilk döngüde evrim bitirilebilir [46].

3.2.2.8 En iyi Sonucun Gösterilmesi

Bitiş ölçütlerine ulaşıldıktan sonra, en iyi değer en iyi çözüm kabul edilerek bu sonuç problemin çözümü olarak sunulur.

3.2.3 Yeni Bir Değiştirme Stratejili Genetik Algoritma

UTKAP' a uygulanacak yeni yer değiştirme stratejili genetik algoritma temelde Wu ve Ji tarafından [53] de önerilmiştir. Bu çalışmada ayrıca seçmeli olarak bir çaprazlama sezgiseli de önermişlerdir. Çalışmalarında bu stratejiyi NP-zor kuadratik atama problemine uygulamışlar ve algoritmanın performansını test etmek için QAPLIB' deki örnekler denenerek ve literatürdeki sonuçlarla karşılaştırılarak, genetik algoritmanın performansının çaprazlama sezgiseliyle birlikte de tek de gelecek vaat edici olduğuna karar vermişlerdir. Ayrıca bu genetik algoritmanın genel yapıda olduğunu yani birçok kombinatoryal probleme uygulanabileceğini de belirtmişlerdir. Bu öneriden yola çıkarak yeni yer değiştirme stratejili genetik algoritmanın bu stratejisinin bir kuadratik atama problemi olan UTKAP' a uygulanabileceğini ve UTKAP' la ilgili literatürdeki çalışmalar da göz önüne alındığında bu uygulamanın ilk olacağını söyleyebiliriz.

3.2.3.1 Yeni Değiştirme Stratejisi

Değiştirme stratejisi, üremeden sonraki nesil üyelerinin ve son nesil üyelerinin nasıl seçileceğini tanımlayan bir seçim stratejisidir. Özellikle çok-kısıtlı problemler için çok önemlidir çünkü arama alanı boyunca algoritma aramasına yol gösterir ve bu yüzden algoritma performansını etkiler. Literatürde en yaygın kullanılan değiştirme stratejisi sabit-durum (steady state) değiştirme stratejisidir. Her nesilde genetik operatörlerin yürütülmesi için bireyler seçilir. Her yeni çocuk popülasyonun en kötü üyesi ile kıyaslanacaktır. Eğer çocuk en kötü üyeden daha iyiyse onun yerini alır.

Bu çalışmada algoritmanın global arama yeteneğini geliştirmeyi hedefleyen yeni bir değiştirme stratejisi önerilmiştir. İki farklı değiştirme politikası olan en kötüyle değiştirme politikası ve ebeveynle değiştirme politikasını içermektedir. Ebeveynle değiştirme politikası her nesilde bir kez yapılır. Her yeni çocuk sadece ebeveyniyle kıyaslanır. Eğer çocuğun uygunluk değeri ebeveyninden iyiye ebeveynin yerini çocuk alır. Buna karşılık, en kötüyle değiştirme politikası her belirli miktar nesilde bir kez alınır. Her yeni çocuk mevcut popülasyondaki en kötü kromozomla kıyaslanır. Eğer çocuğun uygunluk değeri daha iyiye en kötü kromozomun yerini çocuk alır. En kötüyle değiştirme politikasının frekansı ayarlanarak arama sürecinin yakınsaması ve popülasyon çeşitliliği iyi bir şekilde dengelenebilir ve popülasyondan çıkarılmadan önce her kromozomun potansiyelinden iyi bir şekilde faydalanılabilir.

Yeni değiştirme stratejisine dayalı genetik algoritmanın adımları aşağıdaki gibi tanımlanabilir. Şemada, en kötüyle değiştirme politikası her T_c nesilde bir uygulanmaktadır. (T_c en kötüyle değiştirme politikasında periyodu temsil eder.)

başla

ilk popülasyonu yarat ;

her nesil için, tekrarla

bireyleri eşleşme havuzuna rastgele seç;

çocukları oluşturmak için genetik operatörleri uygula ;

$i \times T_c$ nesilde değilse

her yeni çocuğu ebeveyniyle kıyasla ve kötü olanı mevcut popülasyondan çıkar ;

aksi halde

her yeni çocuğu mevcut popülasyondaki en kötü kromozomla kıyasla ve kötü olanı mevcut popülasyondan çıkar, herhangi bir bitiş kriterine denk gelinceye kadar

bitir ;

3.2.3.2 Önerilen Genetik Algoritmanın Tanımı

Aşağıda KAP için önerilen GA açıklanmıştır.

- **Kromozom Gösterimi**

Permütasyon gösterimi tercih edilmiştir.

j	1	2	3	4	5			N-2	N-1	N
$\pi(j)$	3	6	M	2	3	.	.	5	1	4

Şekil 3.6 Kromozom gösterimi

N genden oluşan tamsayı bir dizi şekildeki gibi gösterilmiştir. j nci gendeki değer j nci uçuşa atanan kapı numarasını ifade eder. (Örneğin; 1. uçuşa 3. kapı, 2. uçuşa 6. kapı, 3. uçuşa M. kapı gibi.)

- **Amaç ve Uygunluk Fonksiyonları**

Genetik algoritmanın amacı, amaç fonksiyonunda (eşitlik 3.6) açıklandığı gibi toplam maliyeti en aza indirmektir.

C: Toplam maliyeti

f : Kromozom uygunluk fonksiyonu

C_i : i kromozomunun amaç değeri

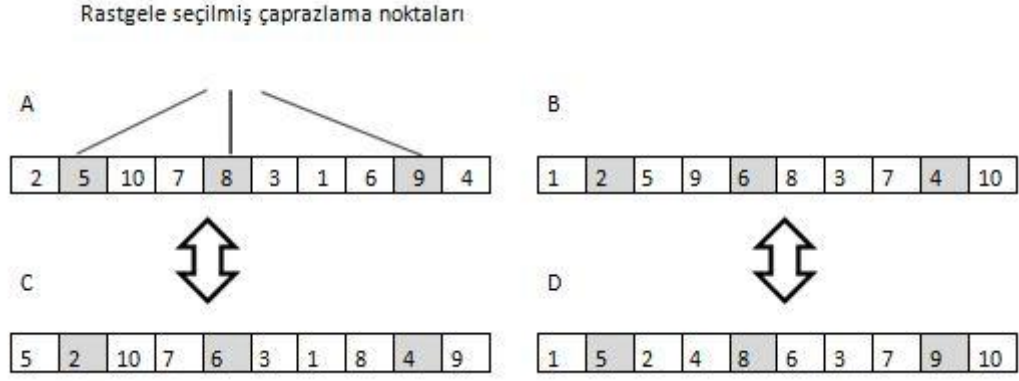
f_i : i kromozomunun uygunluk fonksiyonu olmak üzere;

Genetik algoritmadaki kromozomun uygunluk fonksiyonu aşağıdaki gibi tanımlanır:

$$f_i = C_i \quad (3.11)$$

- **Çaprazlama**

Uniform çaprazlama tercih edilmiştir. Bu çaprazlama süreci şekil 3.7' de gösterilmiştir. çaprazlamadan sonra A ve B bireyleri C ve D bireylerini oluşturmaktadır. Birbirlerine benzer oldukları için A' yı C' nin doğrudan ebeveyni olarak tanımlayabiliriz. Benzer şekilde B de D'nin doğrudan ebeveynidir.



Şekil 3.7 Yeni genetik alıgoritmada çaprazlama şeması

Her nesilde, her kromozom rastgele başka bir kromozom ile eşleştirilmiştir. Çaprazlama operatörü tüm kromozom çiftlerine uygulanmıştır ve bu da çaprazlama oranının 1 olduğu anlamına gelir.

Ön testler sayesinde, mutasyon operatörünün GA' nın problemi çözme performansı üzerinde çok fazla bir etkisi olmadığı tespit edildi. Yani önerilen GA için mutasyon operatörü kullanma kararı kullanıcıya bağlıdır.

- **En iyi Sonucun Gösterilmesi**

Belli bir nesil sayısı bitirildikten sonra döngüler durduruldu ve nesiller boyunca kimin en çok uygunluk gösterdiğine bakıldı. Örnek büyüklüklerine uygun sayıdaki nesil seçildi.

ÖNERİLEN ÇÖZÜM YAKLAŞIMININ UYGULANMASI

Önerilen çözüm yaklaşımının ilk aşamasının amacı açık park alanına atanan uçuşları en aza indirecek şekilde atamaları yapmak ve mevcut atama durumuyla karşılaştırdıktan sonra genetik algoritmanın ilk çözüm kümesini oluşturmaları amacıyla kapılara atanmış uçuşları elde etmektir. İkinci aşamada da elde ettiğimiz sonuçlar genetik algoritmanın ilk çözümleri olarak kabul edilerek farklı parametrelere sahip genetik algoritma yaklaşımlarıyla toplam yürüme mesafesi en aza indirilmeye çalışılmıştır. İyileştirme yapılmadan ve yapıldıktan sonraki maliyetlerin karşılaştırılmasının yanı sıra yeni yer değiştirme stratejili genetik algoritma ve diğer genetik algoritma yaklaşımlarının sonuçları da birbirleriyle karşılaştırılmıştır.

Prosedürlerin performansını rastgele veriler ve AHL' den alınan veriler üzerinden değerlendirdik. Tüm algoritmalar MATLAB R2008b'de kodlandı. Ayrıca MATLAB Genetik Algoritma optimizasyon araç kutusundan faydalanıldı. Çok sayıdaki denemenin etkili bir şekilde yapılabilmesi için MATLAB' de probleme özel bir arayüz geliştirildi. EK A' da ekran görüntüsü yer almaktadır.

4.1 Veriler

4.1.1 Rastgele Veriler

Bolat'ın oluşturduğu ve [21], [22] ve [23] te kullandığı rastgele uçuş çizelgelerini kullandık. Bu verilerde bir zaman birimi 5 dakika olarak kabul edilmiş ve yerde kalma süreleri 15 ve 36 birim zaman arasında düzenli değişmiştir. Problemler Küçük-5 kapılı, orta-10 kapılı, büyük-20 kapılı olmak üzere boyutlarına göre sınıflandırılmıştır. Her

kapıdaki ortalama uçuş sayısı 5.15'tir. Boş zaman aralığını ve süre uzunluğunu ayarlayarak kullanım yüzdesi kontrol edilebilir. Boyutlarına göre ayrılmış her problem de kendi içinde kullanım yüzdelerine göre yüksek kullanım, normal kullanım ve düşük kullanım şeklinde ayrılmışlardır. Yüksek kullanımlı problemlerde süre uzunluğu 150 birim zaman ve boş zamanlar 0 ile 6 birim zaman arasında değişir. Normal kullanımlı olanlarda süre uzunluğu 199 birim zaman ve boş zamanlar 7 ile 15 birim zaman arasında değişir. Düşük kullanımlı problemlerde süre uzunluğu 288 birim zaman (24 saate karşılık gelir) ve boş zamanlar 16 ile 35 birim zaman arasında değişir. Sonuç olarak 9 durum ve her durum için de 8 set veri olmak üzere toplam 72 örnek uçuş çizelgesi vardır. Bu verilerin bir kısmı EK B' de verilmiştir.

Ayrıca bu verilere ikinci amaç fonksiyonumuzu uygulayabilmek için seçtiğimiz belli sayıdaki veriye uygun olacak şekilde rastgele kapı uzaklık matrisleri ve yolcu akış matrisleri tanımladık. Kapı uzaklık matrisleri 5 kapılı örnek tek taraflı terminal, 10 ve 20 kapılı örnek çift taraflı terminal yapısında ve her birim uzaklık 1 metre olarak kabul edildi. Yolcu verileri bir uçağın kapasitesinin 50 ile 300 arasında değiştiği varsayımı altında, girişten gelen ya da çıkışa giden yolcu sayıları 30-60 arasında rastgele, uçuştan uçuşa geçen yolcu sayıları da 0-15 arasında rastgele ayarlandı. Uçuş çizelgelerindeki geliş gidiş saatleri göz önünde bulundurularak birbirine yolcu transfer edemeyen uçuşlar arası yolcu akış değerleri 0 olarak ayarlandı. Bu verilerin bir kısmı EK C ve EK D' de verilmiştir.

4.1.2 Gerçek Veriler

Verilerde 34 uçak tipi, 9 iç hat kapısı, 23 dış hat kapısı ve 58 açık park alanı mevcuttur. Önerilen metot tek tip terminal yapısına uygun olduğu yani dış hat ve iç hat uçuşlarını ayıran bir algoritmaya sahip olmadığı için uluslar arası yolculara öncelik vermek amacıyla çalışmamızda 352 uçuş içeren bir günlük tüm uçuş çizelgesini dikkate almak yerine bu çizelgenin dış hatlar terminalini ilgilendiren 184 uçuş dikkate alınacaktır. Verilerin bir kısmı EK G' de verilmiştir. Kapılar homojen değildir yani her uçak tipi her kapıya atanamaz. Uçak-kapı uyumluluk matrisi EK F' de verilmiştir ve 0 uyumsuzluk durumunu 1 ise uyumluluk durumunu temsil eder.

Uçuş çizelgesinin gösterimini ve hesaplamalarımızı kolaylaştırmak için her bir zaman birimini 5 dakika kabul ettik. Bu da 1 saatin 12 zaman birimine eşit olduğu anlamına gelmektedir. Yani bir kapının günlük kapasitesi $(1 \text{ gün} * 24 \text{ saat} * 1 \text{ kapı} * 60 \text{ dakika} / 5 \text{ dakika})$ 288 birim zamandır. Atatürk Havalimanında atama yapılırken uçuşlar arasında en az 15 dakika tampon süre bulundurma zorunluluğu vardır. Test verilerinde göz önüne almadığımız bu durumu gerçek verilerde göz önüne alabilmek için programa tampon süreleri de dâhil ettik.

Geçmiş verilere dayanarak AHL' de dış hat uçuşlarının ortalama 200 yolcuya sahip olduğunu ve transfer olan yolcu sayılarının da 0-7 arasında rastgele değiştiğini kabul ettik. Bu değerleri ve gelen/ayrılan/transfer olan yolcuları içeren uçuş-yolcu sayısı matrisinin bir kısmı EK H' de verilmiştir. Son olarak yürüme mesafesini hesaplayabilmek için gerekli olan kapı uzaklık matrisini AHL' nin krokisinden birim uzaklıklar şeklinde elde ettik. 10 metre=1 birim olarak gösterilmiştir. Bu matris EK I' da verilmiştir.

4.2 Rastgele Verilerle Uygulama

Öncelikle 72 adet uçuş çizelgesinin hepsi doyumsuz metotla ve ilk gelen ilk hizmet alır (FCFS) metoduyla çözülerek uçuşların kapılara ve açık park alanına atanma oranları karşılaştırılmıştır. Atama oranları iki metot için de aynı ve başarılı bulunmuştur çünkü açık park alanına atama yapmaya gerek kalmamış, tüm uçuşlar kapılara atanmıştır. Buradan da anlaşılmaktadır ki daha yoğun iniş-kalkış saatlerine sahip uçuş çizelgelerinde etkisini daha iyi göstermesi beklenen doyumsuz algoritma, bu tip düzenli verilerde en az FCFS metodu kadar başarılı olmuştur. Sonuçlar Çizelge 4.1 de verilmiştir.

Çizelge 4.1 Rastgele verilerin doyumsuz algoritma-FCFS atanma oranları

Problem	Kapı Sayısı	Uçuş Sayısı	Doyumsuz	FCFS	Problem	Kapı Sayısı	Uçuş Sayısı	Doyumsuz	FCFS	Problem	Kapı Sayısı	Uçuş Sayısı	Doyumsuz	FCFS
DSH1	5	25	100%	100%	DMH1	10	54	100%	100%	DLH1	20	104	100%	100%
DSH2	5	27	100%	100%	DMH2	10	52	100%	100%	DLH2	20	104	100%	100%
DSH3	5	28	100%	100%	DMH3	10	52	100%	100%	DLH3	20	109	100%	100%
DSH4	5	26	100%	100%	DMH4	10	56	100%	100%	DLH4	20	108	100%	100%
DSH5	5	27	100%	100%	DMH5	10	54	100%	100%	DLH5	20	104	100%	100%
DSH6	5	27	100%	100%	DMH6	10	53	100%	100%	DLH6	20	112	100%	100%
DSH7	5	27	100%	100%	DMH7	10	53	100%	100%	DLH7	20	109	100%	100%
DSH8	5	29	100%	100%	DMH8	10	53	100%	100%	DLH8	20	106	100%	100%
DSN1	5	26	100%	100%	DMN1	10	51	100%	100%	DLN1	20	106	100%	100%
DSN2	5	25	100%	100%	DMN2	10	52	100%	100%	DLN2	20	106	100%	100%
DSN3	5	27	100%	100%	DMN3	10	51	100%	100%	DLN3	20	105	100%	100%
DSN4	5	26	100%	100%	DMN4	10	53	100%	100%	DLN4	20	109	100%	100%
DSN5	5	26	100%	100%	DMN5	10	51	100%	100%	DLN5	20	103	100%	100%
DSN6	5	26	100%	100%	DMN6	10	54	100%	100%	DLN6	20	105	100%	100%
DSN7	5	25	100%	100%	DMN7	10	53	100%	100%	DLN7	20	105	100%	100%
DSN8	5	25	100%	100%	DMN8	10	50	100%	100%	DLN8	20	104	100%	100%
DSU1	5	25	100%	100%	DMU1	10	51	100%	100%	DLU1	20	102	100%	100%
DSU2	5	25	100%	100%	DMU2	10	51	100%	100%	DLU2	20	106	100%	100%
DSU3	5	25	100%	100%	DMU3	10	52	100%	100%	DLU3	20	104	100%	100%
DSU4	5	26	100%	100%	DMU4	10	53	100%	100%	DLU4	20	105	100%	100%
DSU5	5	26	100%	100%	DMU5	10	53	100%	100%	DLU5	20	102	100%	100%
DSU6	5	27	100%	100%	DMU6	10	51	100%	100%	DLU6	20	104	100%	100%
DSU7	5	25	100%	100%	DMU7	10	51	100%	100%	DLU7	20	103	100%	100%
DSU8	5	26	100%	100%	DMU8	10	50	100%	100%	DLU8	20	105	100%	100%

İkinci amacımız olan toplam yolcu yürüme mesafesinin en aza indirilebilmesi için küçük/orta/büyük rastgele verilerden bu verilerin yüksek /normal /düşük kullanımlı olanlarından eşit sayılarda seçilerek 18 örnek uçuş çizelgesi alınmıştır. Bu verilere paralel şekilde oluşturduğumuz rastgele kapı uzaklık matrisleri ve gelen/ayrılan/transfer olan yolcu sayısı matrisleri kullanılmıştır. Doyumsuz algoritmayla yapılan atama ve GA1 ve GA2 adı verdiğimiz farklı parametrelili genetik algoritmalar ve yeni değiştirme stratejili genetik algoritma (RGA) ile yapılan atamalar sonucunda elde

edilen amaç fonksiyonu değerleri 18 örnekte her GA için 5 er kez olmak üzere toplamda $18 \times 3 \times 5 = 270$ deneme sonucunda üretilmiştir.

GA1’ deki değerler UTKAP’ da daha önce kullanılmış genel genetik algoritma parametrelerini, RGA’ daki değerler Wu ve Ji’ nin [53] te kullandıkları genetik algoritma parametrelerini ve GA2’ deki değerler RGA’ nın yeni değiştirme stratejisinin etkisini test edebilmek amacıyla sonraki nesil seçimi dışında aynı değerleri içeren genetik algoritma parametrelerini temsil etmektedir. Ayrıca tüm genetik algoritma örnekleri için popülasyon sayıları değerlendirmeyi etkilememesi açısından en ideal değerlerde ve aynı kabul edilmiştir. Çizelge 4.2’ de farklı genetik algoritmaların farklı parametreleri görülmektedir.

Çizelge 4.2 Uygulanan genetik algoritma parametre değerleri

	GA1	GA2	RGA
Populasyon Sayısı	100	100	100
Ebeveyn Seçimi	Rank	Random	Random
Çaprazlama Çeşidi	Tek Noktalı	Uniform	Uniform
Çaprazlama Oranı	0.8	1	1
Mutasyon Çeşidi	Tek Kapı Değiş	Karma	Karma
Mutasyon Oranı	0.2	0.05	0.05
Sonraki Nesil Seçimi	Rulet	En İyiler Seçimi	Yeni değiştirme
Maksimum İterasyon	200(S)/300(M)/500(L)	200(S)/300(M)/500(L)	200(S)/300(M)/500(L)

Programın her çalışması örnek büyüklüğüne bağlı olarak farklı hesaplama zamanına sahiptir ve ortalama 60-1000 saniye sürmüştür. Yapılan 270 denemenin toplam program çalışma süresi ortalama 16200-27000 saniye olarak elde edilmiştir. Kişisel bilgisayar ortamında çalıştırıldığı göz önünde bulundurulduğunda CPU kullanım oranları %70-80 civarında olmuştur.

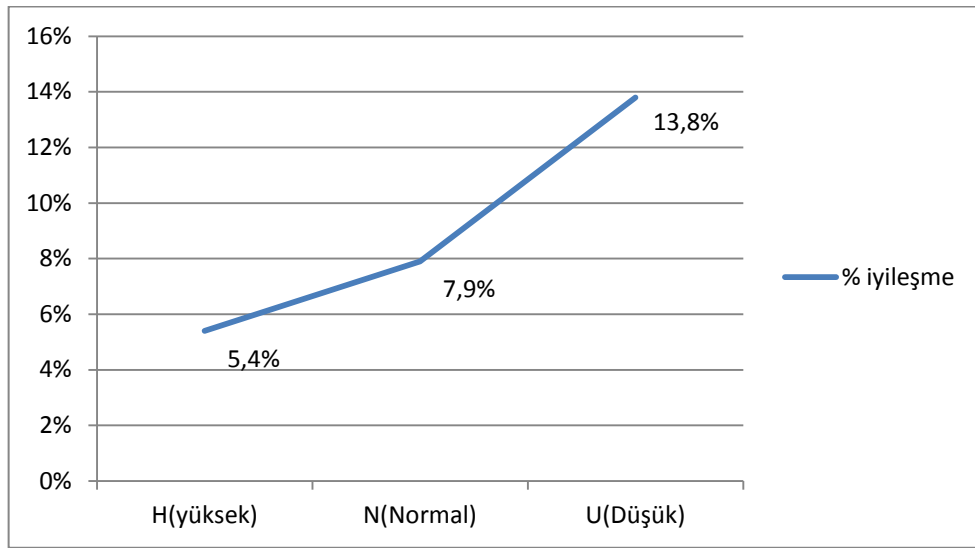
Seçilen örneklerle yapılan denemelerin özeti Çizelge 4.3’ te yer almaktadır. İlk durum sütunu herhangi bir iyileştirme yapmaksızın doyumsuz algoritmanın atamasının sonucunda oluşan toplam yolcu yürüme maliyetlerini gösterirken, min. maliyet sütunu genetik algoritma ile bulunan en iyi değeri, GA sütunu da bu değer hangi GA tarafından elde edildiğini temsil etmektedir. Seçilen örneklere yapılan GA denemelerinin süreleri ve sonuçları EK E’ de verilmiştir.

Çizelge 4.3 Seçilen örneklerle yapılan denemelerin özeti

Örnekler	İlk Durum(birim*kişi)	min.maliyet(birim*kişi)	GA	% iyileşme
DSH1	115440	103390	RGA	10,4
DSH4	145240	141860	RGA	2,3
DSN2	90430	84290	RGA	6,8
DSN6	118300	111680	RGA	5,6
DSU3	110930	98600	RGA	11,1
DSU5	122050	104250	RGA	14,6
ortalama				8,5
DMH4	550410	512390	RGA	6,9
DMH7	509780	488180	RGA	4,2
DMN1	473450	427730	RGA	9,7
DMN4	479160	446760	RGA	6,8
DMU1	451720	396130	GA2	12,3
DMU6	427620	392980	GA2	8,1
ortalama				8,0
DLH6	3749580	3598910	RGA	4,0
DLH8	3427870	3281340	GA2	4,3
DLN2	3296210	2976860	RGA	9,7
DLN4	3519250	3204710	GA2	8,9
DLU3	3088270	2600000	RGA	15,8
DLU5	3264820	2591120	GA2	20,6
ortalama				10,6
toplam iyileşme				9,0 (%)

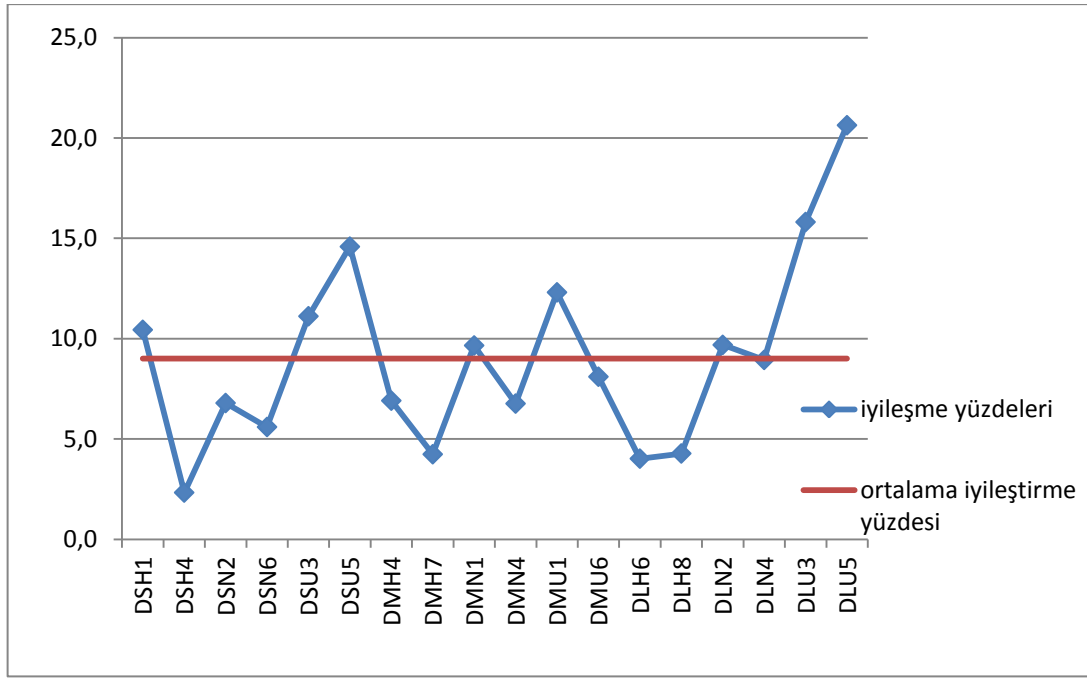
Çizelge 4.3' ten anlaşıldığı üzere GA1 ve GA2' ye göre daha başarılı bulunan önerdiğimiz RGA (Yeni değiştirme stratejili genetik algoritma) 18 örneğin 13 tanesinde minimum maliyeti vermiştir. 5 örnekte de GA2 minimum maliyete ulaşmıştır. GA2 ve RGA' nın yer değiştirme stratejileri dışında aynı parametrelere sahip olmalarını göz önünde bulundurduğumuzda seçilen parametrelerin probleme iyi uyum sağladığını anlamak mümkündür. Anlaşıldığı üzere önceki UTKAP çalışmalarında sıkça kullanan GA değerlerinin kullanıldığı GA1 diğer GA' lara göre iyi sonuçlar verememiştir. Örnek büyüklüklerine, kullanım oranlarına, rastgele yolcu sayısı ve yürüme mesafesi verilerine bağlı olarak her örneğin ilk maliyete göre iyileşme oranları farklı çıkmıştır. Örnek büyüklükleri açısından değerlendirildiğinde küçük örnekler için ortalama % 8.5, orta

büyükölükteki örnekler için ortalama % 8, büyük örnekler için ortalama % 10.6 olmuştur. Örnek büyüklüğü değışimine paralel bir iyileşme söz konusu olmamakla birlikte örnek kullanım oranlarına göre yaptığımız değerdendirme sonucunda Şekil 4.1’ deki sonuçlar elde edilerek örnek kullanım oranlarıyla yüzde iyileşme arasında bir ilişki bulunmuştur. Buna göre kullanım oranı düştükçe yani uçuş çizelgelerinde süre uzunluğu ve boş zamanlar arttıkça genetik algoritma daha iyi sonuçlar vermiştir.



Şekil 4.1 Örneklerin kullanım oranlarına göre ortalama iyileşme yüzdeleri

Şekil 4.2’ de tüm örnekleri ele aldığımızda tüm oranlar ve ortalama %9 olarak elde edilmiş olan toplam iyileşme oranı açıkça gözükmemektedir.



Şekil 4.2 Tüm örneklerin iyileşme yüzdeleri

Rastgele örnek verilerde toplamda ortalama %9 iyileştirme yapılmış ve önerilen yöntemin farklı büyüklüklere ve farklı kullanım oranlarına sahip bu örnekler üzerinde etkili sonuçlar verdiği belirlenmiştir.

4.3 Gerçek Verilerle Uygulama

32 milyonu aşan iç ve dış hat yolcu sayısı ile Avrupa'da sekizinci, dünyada 37'nci en büyük havalimanı olan Atatürk Havalimanında uçuşlar 15 ila 20 yıllık deneyime sahip planlamacılar tarafından manuel olarak kapılara atanır. Bu uçuşlar atanırken 3 temel kuralı göz önünde bulundurulur:

1. Bir uçağı orijinal atandığı konumundan gerekmedikçe hareket ettirmeyin.
2. Herhangi bir uçak hareket gerekiyorsa, kalkış müşterilerine öncelik verin. (Buradan kalkış müşterilerinin memnuniyetine daha çok önem verildiği anlaşılabılır.)
3. İlk giren ilk çıkar (FCFS) yöntemi kullanılarak uçuşları kapılara atayın.

Mevcut yöntemde havaalanındaki planlamacılar en yüksek önceliği dış hatlar terminalinde II bacağına, iç hatlar terminalinde DD bacağına verirler. 2. kurala dayanarak da dış hatlar terminalinde DI bacağı ID bacağından daha önceliklidir ve tersi durum da iç hatlar terminalinde geçerlidir.

Ayrıca yurtdışı hattında hizmet veren uçaklara hem ülkenin prestiji açısından hem de yerel kuruluşların rekabet koşulları dikkate alınarak dış hat kapılarında yer verilmektedir veya 3 saatten fazla konaklama yapacak uçaklar açık park alanına (yine uçak boyutları dikkate alınarak) yönlendirilmektedir. Bu süreyi aşan gecikmelerin meydana getirdiği aksaklıklar görevli kişinin ani olarak verdiği kararlarla yönetilmektedir.

Atama yapılırken AHL' nin 15 dk'lık tampon süre kuralı dikkate alınmıştır. Kapıların her uçak tipine uygun olmaması durumu da göz önünde bulundurarak kapı uygunluk matrisi kullanılmıştır. Uçak tipi ve kapıların uygunluk matrisi EK F de verilmiştir.

Önerilen metodun uçuş tiplerini (uluslararası uçuşlar-iç uçuşlar) ayırma amacı olmadığı için çalışmamızda uluslar arası uçuşlara öncelik vermek için sadece dış hatlar terminalini dikkate aldık. Uzmanların dış hatlara yaptıkları atamalar ve açık park alanına atadıkları (dış hatlar terminaline atanması öncelikli olarak amaçlanan) uluslar arası uçuşları doyumsuz algoritma yöntemiyle kapılara atadık.

Birinci gün 00.00'da ilk gelen uçuştan 1.gün 23.59 da gelen uçuşa kadarki veriler bir günlük uçuş olarak kabul edilmiştir. Bu durumda çizelge ikinci güne de geçmektedir. Ayrıca geliş ve ayrılış süreleri arasında büyük fark olan uçuşlar da uluslar arası kurallar gereği kapıları meşgul etmemeleri açısından açık park alanlarına yönlendirilmiştir. Hem mevcut uygulamada hem de önerdiğimiz yöntemde bu durum geçerlidir. EK G' de bir kısmı verilen dış hat uçuş çizelgesinde ayrıca uzmanların yaptıkları atamalar ve uçuşların geliş-gidiş saatlerinin birim zaman haline dönüştürdüğümüz hali de verilmiştir.

Bir günlük uçuş çizelgesindeki 184 uçuş doyumsuz algoritma ile kapılara ve açık park alanlarına atanmıştır. Bu atamalar Çizelge 4.4' te kapılara atanmış şekilde gösterilmiştir. Doyumsuz algoritma uçuş çizelgesinin yanında uçuş tipi-kapı uyumsuzluk matrisini de dikkate alarak gerçekleştirdiği atamalar sonucunda elde edilen çizelgede görüldüğü gibi 184 uçuştan 175 uçuş kapılara, 9 uçuş da açık park alanına atanmıştır.

Çizelge 4.4 Doyumsuz algoritma ile yapılan atamaların kapılarda gösterimi

kapılar	uçuşlar									
1	5	18	28	49	66	86	98	147		
2	6	19	52	78	105	124	149	165		
3	1	22	44	60	108	131	166	176		
4	7	21	61	128	139	153	164			
5	8	25	83	130	144	169				
6	14	23	47	63	82	118	184			
7	15	24	29	94	121	143	162	173		
8	2	31	53	76	90	111	129	151	180	
9	10	33	72	89	103	132	174			
10	11	34	48	69	85	97	154			
11	12	51	67	81	93	113	122	142	155	178
12	32	57	106	119	127	137	146	163		
13	35	64	80	96	115	135	140	156	170	183
14	43	56	75	88	104	159				
15	39	59	87	109	157	171	182			
16	9	26	73	102	158	172	177			
17	16	27	79	100	133	175				
18	17	36	50	68	84	95	112	152	179	
19	40	58	77	92						
20	3	30	55	107	120	134	145	168		
21	45	62	74	99	114	125	138	150	181	
22	37	54	71	91	110	123	141	160		
23	41	70	101	117	126	136	148			
A1	4									
A2	13	116								
A3	20	161								
A4	38									
A5	42									
A6	46									
A7	65									

Ayrıca aynı durumda uzmanların yapmış oldukları mevcut atamalar da Çizelge 4.5’ te kapılara atanmış şekilde gösterilmiştir.

Çizelge 4.5 Mevcut atamaların kapılardaki gösterimi

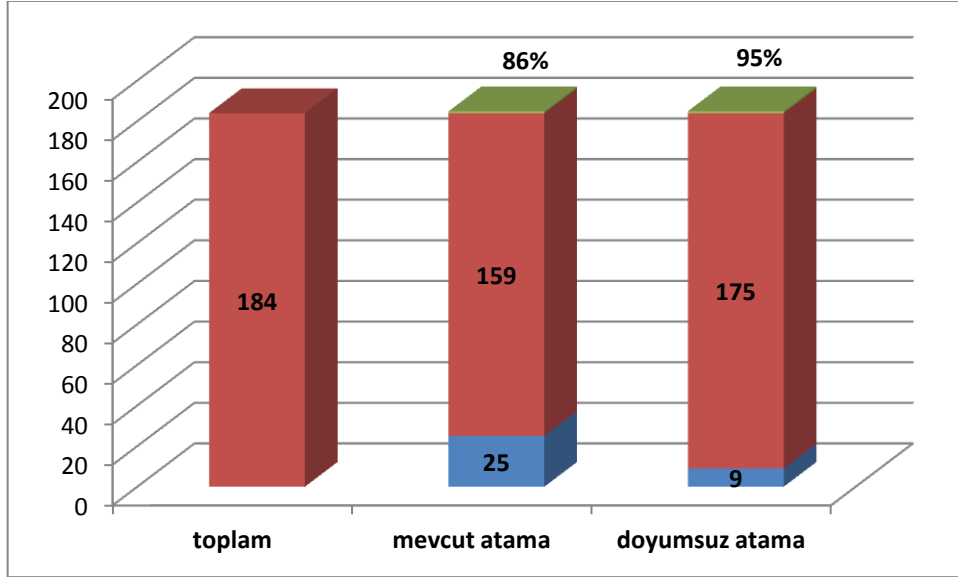
kapılar	uçuşlar									
1	23	44	64	94	147					
2	22	54	89	113						
3	3	33	45	62	76	91	110	119	141	163
4	27	67	85	105	130	145	168			
5	25	51	77	101	121	152	173			
6	26	47	63	83	146	162	175			
7	34	50	78	111	136	155	172			
8	17	40	70	96	118	171				
9	29	48	66	108	125	153	166	184		
10	9	35	56	81	100	124	170			
11	41	69	86	116	142	164				
12	14	28	75	97	114	132	182			
13	58	88	106	122	137	165				
14	5	21	52	107	131	183				
15	10	32	60	80	95	115	150			
16	18	53	73	98	117	129	148	179		
17	31	57	84	103	120	140	157	176		
18	19	46	79	109	135	149	167	178		
19	13	71	90	112	133					
20	15	30	55	74	92	134	151	180		
21	8	24	59	99	126	139	160	174		
22	11	37	68	87	104	128	144	169		
23	7	38	72	93	123	158				
A1	6	39								
A2	43	156	177							
A3	36									
A4	49	102								
A5	42									
A6	143	161								
A7	138	159								
A8	65									
A9	4									
A10	16									
A12	61	127	181							
A13	2									
A14	82									
A16	1									
A20	20									
A22	12									
A26	154									

Çizelge 4.5’ te görüldüğü gibi 184 uçuştan 159 uçuş kapılara, 25 uçuş da açık park alanına atanmıştır.

Doyumsuz algoritma atamasının mevcut durumda uzmanların yapmış oldukları atamalarla karşılaştırılması Çizelge 4.6’ da gösterilmiştir.

Çizelge 4.6 Yöntemlerin atama oranları

	Mevcut Atama Yöntemi	Doyumsuz Metot
Toplam Uçuş Sayısı	184	184
Kapılar Atana Uçuş Sayısı	159	175
Aprona Atanan Uçuş Sayısı	25	9
Atama Yüzdeleri	86%	95%



Şekil 4.3 Yöntemlerin atama oranlarının grafiksel gösterimi

Yapılan değerlendirmeler sonucunda ilk amaç olan aprona atanan uçuşların en aza indirilmesine yönelik önerilen doyumsuz algoritma uçuşların %95' ini kapılara atayarak mevcut yöntemin %86'lık atama oranından daha başarılı atamalar elde etmiştir.

Önerilen yöntemin ikinci kısmı olan toplam yolcu yürüme mesafesi minimizasyon aşamasına geçmeden önce bazı kabuller yapılmalıdır. Önerdiğimiz metodun apron-kapı arası uçuş değişimi yapmadığı ve aprona atanan uçuşlardaki yolcuların servislerle terminale geçtiklerinden yürümeyecekleri düşüncesiyle aprona atanan uçuşların maliyetlerini hesaba katmaması için ve mevcut yöntemin atamasıyla ortaya çıkan toplam yolcu yürüme mesafesi (toplam maliyet) ile GA yaklaşımıyla ortaya çıkan toplam maliyetin aprona atanan uçuşların farkından doğacak maliyetten etkilenmeden karşılaştırılabilmesi için GA' nın ilk çözümünü uzmanların mevcut atamaları olarak kabul edeceğiz. AHL' de toplam yolcu yürüme mesafesinin en aza indirilmesine yönelik

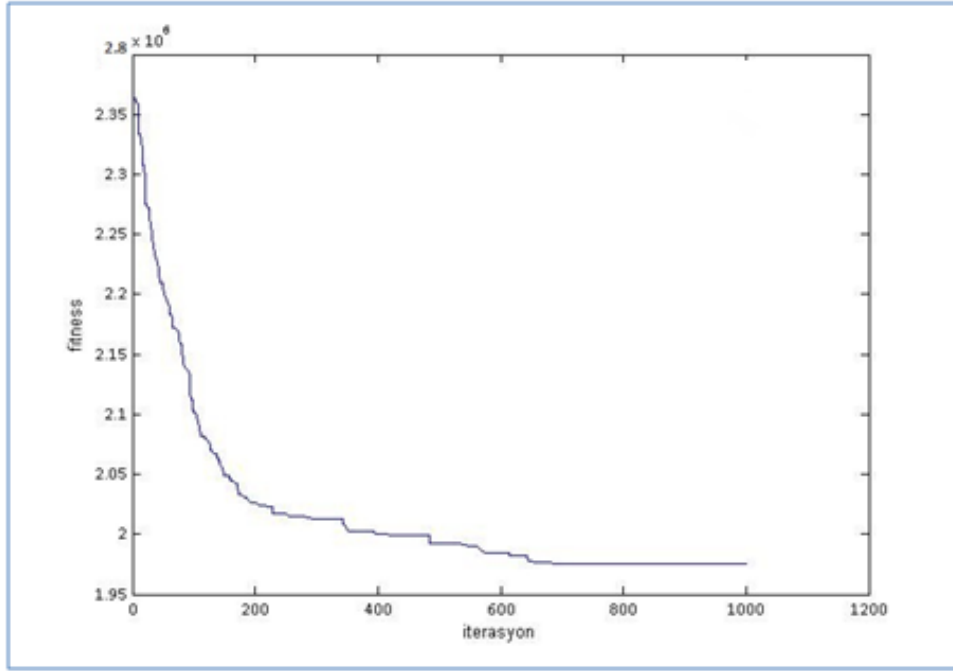
mevcut bir çalışma olmadığını göz önünde bulundurursak bu kabul çözüm önerimiz olan genetik algoritma tabanlı yaklaşımın mevcut durumu büyük oranda iyileştirmesi beklentisini değiştirmeyecek aksine aynı uçuşlar üzerinden elde edilecek iyileştirme başarısını görmek adına mevcut atamaların nasıl daha etkili atanabileceğini ortaya koyacaktır. Bu durumda ilk çözüm kümesi uzmanların atadığı uçuşlar olan 159 uçuşa eşit olacaktır.

Örnek büyüklüğü ve genetik algoritmanın çalışma prensibi göz önüne alındığında uçuş-kapı uyumsuzluğu durumu modelin kısıtlarını gereğinden fazla arttırarak genetik algoritmanın verimini düşüreceği için dikkate alınmayacaktır ki dış hatlar terminali kapılarının çizelgedeki 159 uçuşun uçuş tiplerinin yoğunluğuna hizmet edebilecek uygunlukta olması bu kabulün isabetli olduğuna işaret etmektedir.

Amacımız için örnek verilere uygulanan aynı genetik algoritmaları kullandık. GA1’deki değerler UTKAP’da daha önce kullanılmış genel genetik algoritma parametrelerini, RGA’deki değerler Wu ve Ji’ nin [53] te kullandıkları genetik algoritma parametrelerini ve GA3’teki değerler RGA’nın yeni değiştirme stratejisinin etkisini test edebilmek amacıyla sonraki nesil seçimi dışında aynı değerleri içeren genetik algoritma parametrelerini temsil etmektedir. Gerçek uygulamayı problem büyüklüğü dikkate alınarak ilk popülasyon sayısı 150, maksimum popülasyon sayısı da 1000 olarak ayarlanmıştır. Her GA çeşidi için 5 er kez olmak üzere 15 deneme yapılmıştır. Uygulamalar sonucunda elde edilen değerler Çizelge 4.7’de görülmektedir.

Çizelge 4.7 Elde edilen toplam maliyet değerleri

Denemeler	İlk durum birim*kişi	cpu zamanı(s)	GA1 birim*kişi	cpu zamanı(s)	GA2 birim*kişi	cpu zamanı(s)	RGA birim*kişi
1	2889470	1045	2068110	1113	2000130	1265	1994670
2		1019	2065100	1152	2024960	1194	2054000
3		1072	2092010	1203	2007045	1491	2019000
4		1034	2091465	1249	1994160	1480	1975575
5		1038	2062165	1145	2017825	1468	2006100
Ortalama fit			2075770		2008824		2009869
Min. fit			2062165		1994160		1975575
% iyileşme			28,6		31		31,6

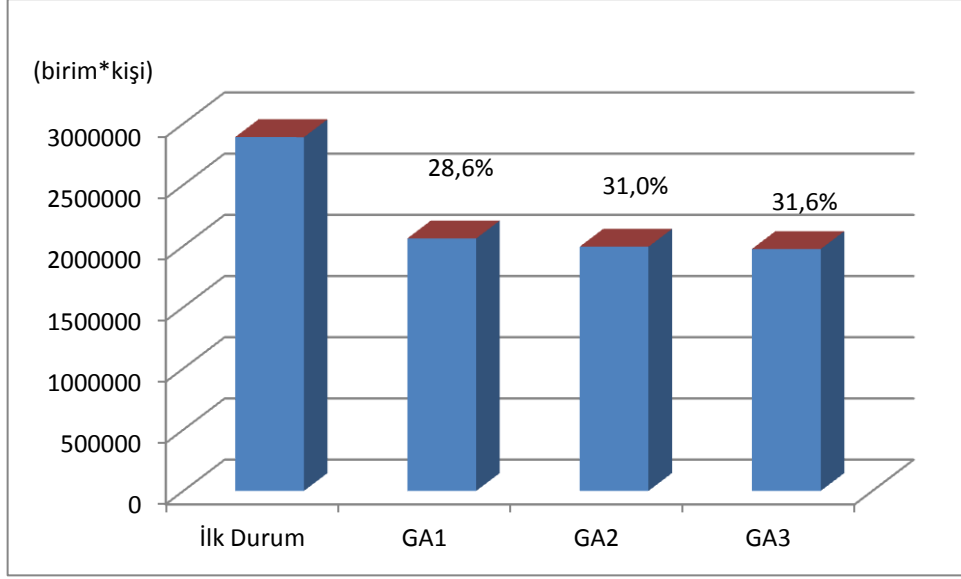


Şekil 4.4 RGA'nın minimum uygunluk değeri

Çizelge 4.7' den anlaşıldığı üzere GA1 ve GA2' ye göre daha başarılı bulunan önerdiğimiz RGA (Yeni değiştirme stratejili genetik algoritma) minimum maliyeti vermiştir. Ayrıca Şekil 4.4' e bakıldığında RGA' nın minimum uygunluk değerine ulaşırken uygunluk değerinin iterasyon sayısına göre değişimi görülmektedir.

Ayrıca programın her çalışması ortalama 1000-1500 saniye sürmüştür. Yapılan 15 denemenin toplam program çalışma süresi ortalama 15000-22500 saniye olarak elde edilmiştir. Kişisel bilgisayar ortamında çalıştırıldığı göz önünde bulundurulduğunda CPU kullanım oranları %80-90 civarında olmuştur.

GA2 ve RGA' nın yer değiştirme stratejileri dışında aynı parametrelere sahip olmalarını göz önünde bulundurduğumuzda birbirine yakın sonuçlar verdiklerinden seçilen parametrelerin gerçek verilere iyi uyum sağladığını anlamak mümkündür. Anlaşıldığı üzere önceki UTKAP çalışmalarında sıkça kullanan GA değerlerinin kullanıldığı GA1 diğer GA' lara göre iyi sonuçlar verememekle birlikte mevcut durumdan % 28,6' lık bir iyileşme sağlamıştır. GA2 ise % 31' lik iyileştirme ile ikinci en iyi çözümü vermiştir.



Şekil 4.5 Elde edilen minimum maliyet değerleri

RGA mevcut durumda % 31,6 iyileştirme yapmış, toplam yolcu yürüme mesafesini 2889470 birim*kişi' den 1975575 birim*kişi' ye düşürmüştür.

Ayrıca mevcut çözümde yolcu başına düşen ortalama yürüme mesafesi 51,1 birim iken önerilen çözüm yaklaşımıyla bu değer 34,9 birime düşürülmüştür.

SONUÇ VE ÖNERİLER

Tez kapsamında amacı kapılara atanmamış uçuşları ve toplam yürüme mesafesini en aza indirmek olan UTKAP üzerine çalışılmış ve iki aşamalı bir çözüm yaklaşımı önerilmiştir. Ortaya konulan çözüm yaklaşımı Atatürk Havalimanı'nın dış hatlar terminalinde kapı atama problemini çözmek için uygulanmıştır.

Uçakların terminal kapılarına atanması probleminin yapısı nedeni ile karmaşık olması durumu ve problem boyutu büyüdükçe çözüme zor ulaşılması durumu göz önünde bulundurulduğunda analitik yöntemlerin maliyeti yüksek çözümler sunması kaçınılmazdır. Bu bağlamda matematiksel programlama tekniklerinden doyumsuz algoritma ve sezgisel çözüm yöntemlerinden genetik algoritmanın yeteneklerinin bir arada kullanılması uygun bulunmuştur.

İlk adımda (eğer varsa) uçuş sayısının kapı sayısını aştığı durum elimine edilerek uçuşlar kapılara yerleştirilmiş, ikinci adımda farklı parametrelerle genetik algoritma meta sezgiseli kullanılarak yolcu yürüme mesafesi en aza indirilmeye çalışılmıştır.

Rastgele veriler kullanılarak ve bu verilere dayalı havaalanı-yolcu verileri türetilerek iki aşamalı çözüm önerimizi uyguladık ve yöntemi test ettik. Birinci aşamada atama oranlarının karşılaştırılması açısından doyumsuz algoritma en az FCFS metodu kadar başarılı olmuştur. İkinci aşamada üç farklı genetik algoritma yöntemi denenmiş önceki UTKAP çalışmalarında sıkça kullanan GA değerlerinin kullanıldığı GA1 diğer GA' lara göre iyi sonuçlar verememiştir. GA1 ve GA2' ye göre daha başarılı bulunan önerdiğimiz

RGA 18 örneğin 13 tanesinde minimum maliyeti vermiştir. 5 örnekte de GA2 minimum maliyete ulaşmıştır. GA2 ve RGA' nın yer değiştirme stratejileri dışında aynı parametrelere sahip olmalarını göz önünde bulundurduğumuzda seçilen parametrelerin probleme iyi uyum sağladığını anlamak mümkündür. Bu verilerde ilk duruma göre ortalama %9 iyileştirme yapılmış ve önerilen yöntemin farklı büyüklüklere ve farklı kullanım oranlarına sahip bu örnekler üzerinde etkili sonuçlar verdiği belirlenmiştir.

Uygulama kapsamında AHL' nin dış hatlar terminalinde yapılan değerlendirmeler sonucunda 184 uçuş, 23 kapı, 58 açık park alanına ve kapıların uçuşlara uyumlu olup olmamasına bağlı olarak ilk aşamada aprona atanan uçuşların en aza indirilmesine yönelik önerdiğimiz doyumsuz metotla 175 uçuş kapılara 9 uçuş aprona, uzmanların metoduyla da 159 uçuşu kapılara 25 uçuşu aprona atanmıştır. Uluslar arası uçuşlara kalite ve ülke prestiji açısından verilen önem göz önünde bulundurulduğunda önceki yöntemle aprona atanmış 16 uluslar arası uçuşun önerilen metodla kapılara atanması sağlanmıştır. Doyumsuz algoritma uçuşların %95' ini kapılara atayarak mevcut yöntemin %86'lık atama oranından daha başarılı atamalar elde etmiştir.

İkinci aşama olan toplam yolcu yürüme mesafesi minimizasyon aşamasında yapılan bazı kabullerle birlikte ilk çözüm kümesi uzmanların kapılara atadığı 159 uçuş olarak alınmış ve GA1, GA2 ve RGA (Yeni değiştirme stratejili genetik algoritma)' dan elde edilen maliyetler ilk çözüm kümesinden elde edilen maliyetle karşılaştırılarak GA1 ve GA2' ye göre daha başarılı bulunan RGA minimum maliyeti vermiştir. RGA mevcut durumda % 31,6 iyileştirme yapmış, toplam yolcu yürüme mesafesini 2889470 birimden 1975575 birime düşürmüştür. Ayrıca mevcut çözümde yolcu başına düşen ortalama yürüme mesafesi 51,1 birim iken önerilen çözüm yaklaşımıyla bu değer 34,9 birime düşürülmüştür.

Sonuç olarak UTKAP' da daha önce denenmemiş genel kuadratik atama probleminde etkili sonuçlar vermiş yeni değiştirme stratejili genetik algoritma metasezgiseli ilk kez bu çalışmada kullanılmış ve mevcut durumu büyük oranda iyileştirerek başarılı bulunmuştur.

Ayrıca tez kapsamında yapılan UTKAP için güncel çalışmaları içeren geniş literatür çalışması araştırmacılar için bir kaynak olma niteliğindedir.

Tez kapsamında geliştirilebilecek bazı hususlar ve öneriler de mevcuttur. Yapılan 15 denemenin toplam program çalışma süresinin ortalama 15000-22500 saniye olarak elde edilmiş olmasını ve CPU kullanım oranlarının %80-90 değerlerinde değişmesini göz önünde bulundurduğumuzda, önerdiğimiz yaklaşımda programın kişisel bilgisayarda çalışıyor olması hesaplama hızının düşük olmasına neden olduğu için yeterli kapasiteye sahip bir sistem kullanıldığında uçuş gecikmelerini de modele dâhil edecek bir yaklaşım rötar yapan uçuşların yapılan kapı planlamasını etkilememesi açısından etkili olacaktır.

Çalışmamızda daha önce UTKAP' da denenmemiş genel kuadratik atama probleminde etkili sonuçlar vermiş yeni değiştirme stratejili genetik algoritma meta sezgiselini kullandık. Tek başına da etkili sonuçlar veren bu genetik algoritma yaklaşımı başka meta sezgisellerle birleştirilerek daha etkili sonuçlar da elde edilebilir.

Çalışmamızda dış hatlar terminalini dikkate alarak aprona atanan II bacaklarını öncelikli oldukları için kapılara tekrar atadık. Ancak uçuşların tüm II, DD, ID, DI bacakları göz önünde bulunduran bir ön program geliştirilerek uçuşlara başka atama seçenekleri eklenebilir ve sonrasında önerilen çözüm yöntemi esnek yapıda olduğu için bu programa entegre edilebilir.

Ayrıca önerilen çözüm yaklaşımına UTKAP' da genelde kullanılan diğer amaç fonksiyonları da eklenerek çözüm genişletilebilir.

Son olarak belirtilmelidir ki Atatürk Havalimanı'nda veri toplama zorluğundan dolayı ancak bir günlük uçuş planıyla çalışılmak zorunda kalındı. Daha kapsamlı bir çalışma ile önerilen metoddaki iyileştirme oranı daha iyi olabileceğinden havaalanının karakteristik özellikleri ortaya çıkarılabilecektir.

KAYNAKLAR

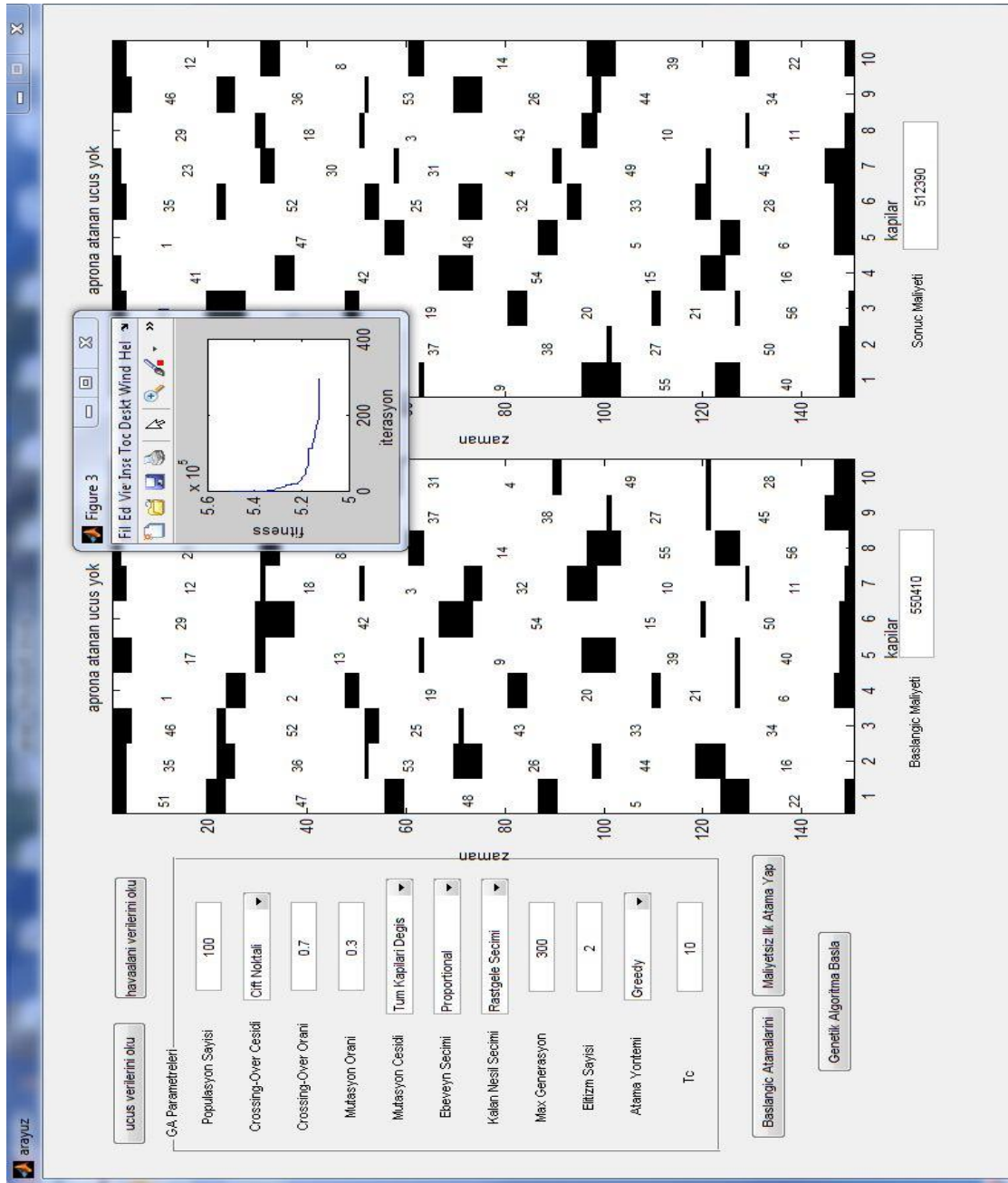
-
- [1] Braaksma, J.P. ve Shortreed, J.H., (1971). "Improving airport gate usage with critical path method", *Transportation Engineering Journal of ASCE* 97(TE2), 187-203.
 - [2] Babic, O., Teodorovic, C., ve Tasic, V., (1984). "Aircraft stand assignment to minimize walking", *J. Trans Engineering*, 110: 55-66.
 - [3] Mangoubi, R. S. ve Mathaisel, D. F. X., (1985). "Optimizing gate assignments at airport terminals", *Transportation Science*, 19:173-88.
 - [4] Bihr, R.A., (1990). "Assigning arriving flights at an airport to the available Gates", *Journal of the Operational Research Society*, 50:23-34.
 - [5] Bandara, S. ve Wirasignhe, S.C., (1992). "Walking distance minimization for airport terminal configurations", *Transportation Research-A* 26(1), 59-74.
 - [6] Wirasinghe, S.C., Bandara, S., ve Vandebona, U., (1987). "Airport terminal geometries for minimal walking distances", *Proceedings of the 10th Symposium on Transportation and Traffic Theory*, eds N.H. Gartner and N.H.M. Wilson pp. 483-502.
 - [7] Robuste, F., (1991). "Centralized hub-terminal geometric concepts. I: walking distance", *Journal of Transportation Engineering*, 117: 2.
 - [8] Obata, T., (1979). "The quadratic assignment problem: evaluation of exact and heuristic algorithms", *Technical report TRS-7901*, Rensselaer Polytechnic Institute, Troy, New York.
 - [9] Xu, J. ve Bailey, G., (2001). "The airport gate assignment problem: Mathematical model and a tabu search algorithm", In *Proceedings of the 34th Hawaii International Conference on System Sciences*, 2001, Island of Maui, Hawaii, USA.
 - [10] Haghani, A. ve Chen, M., (1998). "Optimizing gate assignments at airport terminals", *Transportation Research, A* 32, 437-454.
 - [11] Hassounah, M. ve Steuart, G., (1993). "Demand for aircraft gates", *Transportation Research*, 1423:26-33.
 - [12] Yan, S. ve Chang, C., (1998). "A network model for gate assignment", *Journal of Advanced Transportation*, 32:176-89.

- [13] Yan, S. ve Huo, M. C., (2001). "Optimization of multiple objective gate assignments", *Transportation Research*, 35(5):413-432.
- [14] Yan, S., Shieh, C. Y. ve Chen, M., (2002). "A simulation framework for evaluating airport gate assignments", *Transportation Research*, 36:885–98.
- [15] Yan, S. ve Tang, C. H., (2007). "A heuristic approach for airport gate assignments for stochastic flight delays", *European Journal of Operational Research*, 180(2):547.
- [16] Ding, H., Lim, A., Rodrigues, B. ve Zhu, Y., (2004). "New heuristics for the overconstrained flight to gate assignments", *Journal of the Operational Research Society*, 55:760–768.
- [17] Ding, H., Lim, A., Rodrigues, B. ve Zhu, Y., (2005). "The over-constrained airport gate assignment problem", *Computers & Operations Research*, 32:1867–1880.
- [18] Ding, H., Lim, A., Rodrigues, B. ve Zhu, Y., (2004). "Aircraft and gate scheduling optimization at airports", In *Proceedings of the 37th Hawaii International Conference on System Sciences*, 2004, Big Island, Hawaii, USA.
- [19] Dorndorf, U., Drexl, A., Nikulin, Y., ve Pesch, E., (2007). "Flight gate scheduling: State-of-the-art and recent developments", *Omega*, 35: 326–334.
- [20] Drexl, A. ve Nikulin, Y., (2008). "Multicriteria airport gate assignment and Pareto simulated annealing", *IIE Transactions (Institute of Industrial Engineers)*, 40(4):385-397.
- [21] Bolat, A., (1999). "Assigning arriving flights at an airport to the available gates", *Journal of the Operational Research Society*, 50: 23–34.
- [22] Bolat, A., (2000). "Procedures for providing robust gate assignments for arriving aircraft", *European Journal of Operations Research*, 120:63–80.
- [23] Bolat, A., (2001) "Models and a genetic algorithm for static aircraftgate assignment problem", *Journal of the Operational Research Society*, 52:1107–20.
- [24] Lim, A. ve Wang, F., (2005). "Robust airport gate assignment", *Tools with Artificial Intelligence*, 17th IEEE International Conference on (1082-3409), (0-7695-2488-5):81-81.
- [25] Yang, S. ve Hu, M., (2009). "Airport gate assignment based on improved GM(1,1) model", *Grey Systems and Intelligent Services*, IEEE International Conference on (978-1-4244-4914-9):407-407.
- [26] Zheng, P., Hu, S. ve Zhang, C., (2010). "Airport Gate Assignments Model and Algorithm", *Computer Science and Information Technology (ICCSIT) 3rd IEEE International Conference on* (978-1-4244-5537-9), 2:461-461
- [27] Wei, D. ve Liu C., (2007). "Optimizing Gate Assignment at Airport Based on Genetic-Tabu Algorithm", *Automation and Logistics*, IEEE International Conference on (978-1-4244-1531-1):1140-1140, 2007, Jinan, China.

- [28] Wei, D. ve Liu C., (2009). "Fuzzy model and optimization for airport gate assignment problem", IEEE International Conference on Intelligent Computing and Intelligent Systems, 2:828-832, 2009, Jinan, China.
- [29] Hu, X. B. ve Di Paolo, E., (2007). "An Efficient Genetic Algorithm with Uniform Crossover for the Multi-Objective Airport Gate Assignment Problem", IEEE Congress on Evolutionary Computation.
- [30] Hu, X. B. ve Di Paolo, E., (2009). "A ripple-spreading Genetic Algorithm for the airport Gate Assignment Problem", Evolutionary Computation, IEEE Congress on (978-1-4244-2958-5):1864-1864, 2009.
- [31] Gu, Y. B. ve Chung, C. A., (1999). "Genetic algorithm approach to aircraft gate reassignment problem", Journal of Transportation Engineering, (0733-947X) 125(5):384-389.
- [32] Tang, C. H., (2009). "Real-time gate assignments under temporary gate shortages and stochastic flight delays", IEEE/INFORMS International Conference on Service Operations, Logistics and Informatics, 266-271, SOLI 2009.
- [33] Yan, S., Chen, C. Y. ve Tang, C. H., (2009). "Airport gate reassignment following temporary airport closures", Transportmetrica, 5(1):25-41.
- [34] Tang, C. H., Yan, S. ve Hou, Y.Z., (2010). "A gate reassignment framework for real time flight delays", Journal of Operations Research, 8,(3):299-318.
- [35] Hamzwawi, S., (1986). "Management and planning of airport gate capacity: a microcomputer-based gate assignment simulation model", Transportation Planning and Technology, 11:189-202.
- [36] Gosling, G., (1990). "Design of an expert system for aircraft gate assignment", Transportation Research, 24:59-69.
- [37] Muthukrishnan, R., ve Srihari, K., (1991). "An expert system methodology for aircraft-gate assignmen", Computers and Industrial Engineering, 21:101-105.
- [38] Jo, G. S., Jung, J. J. ve Yang, C. Y., (1997). "Expert system for scheduling in an airline gate allocation", Expert Systems with Applications, 13(4):275.
- [39] Cheng, Y., (1997). "A knowledge-based airport gate assignment system integrated with mathematical programming", Computers and Industrial Engineering, 32:837-852.
- [40] Cheng, Y., (1998). "Network-based simulation of aircraft at Gates in airport terminals", Journal of Transportation Engineering, 124:188-96.
- [41] Cheng, Y., (1998). "Rule-based reactive model for the simulation of aircraft on airport gates", Knowledge-Based Systems (0950-7051), 10,(4):225-236.
- [42] Cheng, Y., (1998). "Solving push-out conflivts in apron taxiways of airports by a network-based simulation", Computers and Industrial Engineering, 34:351-69.
- [43] T.C Ulaştırma Bakanlığı Sivil Havacılık Genel Müdürlüğü, 2002-2009 yılları arasında Türkiye havayolu istatistikleri, <http://web.shgm.gov.tr/>, Mayıs 2011.

- [44] Maharjan, B., (2010). Flight Gate Assignment and Proactive Flight Gate Reassignment Optimization for Hub and Spoke Airline Operations, M.Sc Dissertation, Texas Tech University Graduate Faculty, Texas.
- [45] Karaboğa, D., (2004). Yapay Zekâ Optimizasyon Algoritmaları, Atlas Yayın Dağıtım, İstanbul.
- [46] Man, K.F., Tang, K.S. ve Kwong, S., (1999). Genetic Algorithms, Springer Publishing.
- [47] Altay, A., (2007). Genetik Algoritma ve Bir Uygulama, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul
- [48] Sakawa, M., (2002). Genetic Algorithms and Fuzzy Multiobjective Optimization, Kluwer Academic Publishers.
- [49] Michalewicz, Z., (1992). Genetic Algorithms and Data Structures, Evolution Programs, Springer-Verlag, 3rd edition, Berlin, Heidelberg.
- [50] Goldberg, D. E., (1989). Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Publishing Company.
- [51] Miller, B. L. ve Goldberg, D. E., (1995). Genetic Algorithms, Tournament Selection and the Effect of Noise, Illinois, ABD.
- [52] Deb, K., (2001). Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons Ltd., England.
- [53] Wu, Y. ve Ji, P., (2007). "Solving the Quadratic Assignment Problems by a Genetic Algorithm with a New Replacement Strategy", World Academy of Science, Engineering and Technology, 30.
- [54] MATLAB, Documentation and Support, <http://www.mathworks.com/products/matlab/>, Haziran 2011.

MATLAB'DE PROGRAM EKRAN GÖRÜNTÜSÜ



ÖRNEK RASTGELE UÇUŞ VERİLERİ

DSH1-TOTAL NUMBER OF GATES:5			DMH1-TOTAL NUMBER OF GATES:10			DLH1-TOTAL NUMBER OF GATES:20		
TOTAL NUMBER OF FLIGHTS:25			TOTAL NUMBER OF FLIGHTS:54			TOTAL NUMBER OF FLIGHTS:104		
THE LENGTH OF THE HORIZON:150.0			THE LENGTH OF THE HORIZON:150.0			THE LENGTH OF THE HORIZON:150.0		
Flight	Arrival T.	Dep. T.	Flight	Arrival T.	Dep. T.	Flight	Arrival T.	Dep. T.
1	4.0	34.0	1	0.0	20.0	1	1.0	33.0
2	37.0	52.0	2	24.0	59.0	2	33.0	52.0
3	53.0	86.0	3	63.0	93.0	3	53.0	86.0
4	87.0	120.0	4	95.0	121.0	4	91.0	124.0
5	124.0	144.0	5	126.0	150.0	5	128.0	150.0
6	5.0	29.0	6	2.0	32.0	6	4.0	31.0
7	34.0	59.0	7	32.0	61.0	7	31.0	66.0
8	63.0	89.0	8	61.0	81.0	8	69.0	90.0
9	91.0	117.0	9	85.0	116.0	9	95.0	110.0
10	122.0	145.0	10	120.0	147.0	10	111.0	144.0
11	3.0	23.0	11	0.0	17.0	11	3.0	31.0
12	28.0	47.0	12	17.0	40.0	12	36.0	57.0
13	47.0	70.0	13	40.0	59.0	13	57.0	86.0
14	75.0	109.0	14	63.0	89.0	14	91.0	125.0
15	114.0	147.0	15	89.0	113.0	15	129.0	149.0
16	1.0	26.0	16	118.0	149.0	16	0.0	31.0
17	26.0	51.0	17	3.0	27.0	17	33.0	68.0
18	53.0	84.0	18	32.0	67.0	18	72.0	93.0
19	86.0	121.0	19	69.0	104.0	19	97.0	117.0
20	123.0	148.0	20	105.0	120.0	20	117.0	145.0
21	3.0	31.0	21	123.0	147.0	21	4.0	25.0
22	31.0	58.0	22	3.0	29.0	22	29.0	63.0
23	60.0	84.0	23	31.0	56.0	23	66.0	96.0
24	85.0	117.0	24	57.0	82.0	24	99.0	114.0
25	119.0	148.0	25	82.0	115.0	25	117.0	146.0
			26	119.0	149.0	26	0.0	18.0
			27	5.0	37.0	27	23.0	54.0
			28	40.0	60.0	28	57.0	78.0
			29	61.0	78.0	29	81.0	104.0
			30	79.0	103.0	30	107.0	125.0
			31	103.0	125.0	31	129.0	147.0
			32	125.0	149.0	32	2.0	21.0
			33	0.0	15.0	33	24.0	55.0
			34	17.0	43.0	34	58.0	77.0
			35	44.0	71.0	35	79.0	102.0
			36	72.0	107.0	36	104.0	119.0
			37	107.0	132.0	37	121.0	144.0
			38	133.0	150.0	38	4.0	33.0
			39	3.0	37.0	39	37.0	52.0
			40	39.0	54.0	40	54.0	81.0
			41	58.0	74.0	41	81.0	101.0
			42	74.0	100.0	42	101.0	118.0
			43	105.0	121.0	43	123.0	150.0
			44	122.0	150.0	44	3.0	37.0
			45	4.0	34.0	45	37.0	70.0
			46	35.0	54.0	46	70.0	96.0
			47	56.0	88.0	47	98.0	116.0
			48	93.0	123.0	48	119.0	146.0
			49	123.0	150.0	49	4.0	39.0
			50	2.0	34.0	50	39.0	66.0
			51	36.0	53.0	51	69.0	84.0
			52	54.0	84.0	52	89.0	124.0
			53	88.0	116.0	53	128.0	145.0
			54	118.0	144.0	54	4.0	33.0

55	37.0	65.0
56	68.0	101.0
57	104.0	120.0
58	124.0	148.0
59	1.0	32.0
60	32.0	65.0
61	70.0	87.0
62	91.0	125.0
63	128.0	147.0
64	4.0	25.0
65	26.0	42.0
66	42.0	60.0
67	64.0	99.0
68	102.0	119.0
69	123.0	149.0
70	0.0	30.0
71	32.0	61.0
72	65.0	92.0
73	97.0	119.0
74	121.0	147.0
75	3.0	28.0
76	29.0	64.0
77	64.0	80.0
78	82.0	110.0
79	111.0	146.0
80	0.0	15.0
81	15.0	43.0
82	47.0	80.0
83	82.0	116.0
84	116.0	150.0
85	4.0	25.0
86	28.0	53.0
87	54.0	83.0
88	83.0	113.0
89	117.0	144.0
90	3.0	31.0
91	34.0	68.0
92	69.0	92.0
93	96.0	113.0
94	114.0	146.0
95	5.0	36.0
96	39.0	69.0
97	70.0	94.0
98	96.0	124.0
99	125.0	148.0
100	2.0	35.0
101	39.0	74.0
102	78.0	102.0
103	102.0	119.0
104	121.0	150.0

RASTGELE VERİLERİN KAPI UZAKLIK MATRİSLERİ

C-1 5 Kapılı Uzaklık Matrisi

kapi/kapi mesafe(birim)	0	1	2	3	4	5
0	0	10	20	30	40	50
1	10	0	10	20	30	40
2	20	10	0	10	20	30
3	30	20	10	0	10	20
4	40	30	20	10	0	10
5	50	40	30	20	10	0

C-2 10 Kapılı Uzaklık Matrisi

kapi/kapi mesafe(birim)	0	1	2	3	4	5	6	7	8	9	10
0	0	10	20	30	40	50	10	20	30	40	50
1	10	0	10	20	30	40	20	30	40	50	60
2	20	10	0	10	20	30	30	20	30	40	50
3	30	20	10	0	10	20	40	30	20	30	40
4	40	30	20	10	0	10	50	40	30	20	30
5	50	40	30	20	10	0	60	50	40	30	20
6	10	20	30	40	50	60	0	10	20	30	40
7	20	30	20	30	40	50	10	0	10	20	30
8	30	40	30	20	30	40	20	10	0	10	20
9	40	50	40	30	20	30	30	20	10	0	10
10	50	60	50	40	30	20	40	30	20	10	0

C-3 20 Kapılı Uzaklık Matrisi

Kapı/Kapı (birim)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0	10	20	30	40	50	60	70	80	90	100	10	20	30	40	50	60	70	80	90	100
1	10	0	10	20	30	40	50	60	70	80	90	20	30	40	50	60	70	80	90	100	110
2	20	10	0	10	20	30	40	50	60	70	80	30	20	30	40	50	60	70	80	90	100
3	30	20	10	0	10	20	30	40	50	60	70	40	30	20	30	40	50	60	70	80	90
4	40	30	20	10	0	10	20	30	40	50	60	50	40	30	20	30	40	50	60	70	80
5	50	40	30	20	10	0	10	20	30	40	50	60	50	40	30	20	30	40	50	60	70
6	60	50	40	30	20	10	0	10	20	30	40	70	60	50	40	30	20	30	40	50	60
7	70	60	50	40	30	20	10	0	10	20	30	80	70	60	50	40	30	20	30	40	50
8	80	70	60	50	40	30	20	10	0	10	20	90	80	70	60	50	40	30	20	30	40
9	90	80	70	60	50	40	30	20	10	0	10	100	90	80	70	60	50	40	30	20	30
10	100	90	80	70	60	50	40	30	20	10	0	110	100	90	80	70	60	50	40	30	20
11	10	20	30	40	50	60	70	80	90	100	110	0	110	100	90	80	70	60	50	40	30
12	20	30	20	30	40	50	60	70	80	90	100	110	0	110	100	90	80	70	60	50	40
13	30	40	30	20	30	40	50	60	70	80	90	100	110	0	110	100	90	80	70	60	50
14	40	50	40	30	20	30	40	50	60	70	80	90	100	110	0	110	100	90	80	70	60
15	50	60	50	40	30	20	30	40	50	60	70	80	90	100	110	0	110	100	90	80	70
16	60	70	60	50	40	30	20	30	40	50	60	70	80	90	100	110	0	110	100	90	80
17	70	80	70	60	50	40	30	20	30	40	50	60	70	80	90	100	110	0	110	100	90
18	80	90	80	70	60	50	40	30	20	30	40	50	60	70	80	90	100	110	0	110	100
19	90	100	90	80	70	60	50	40	30	20	30	40	50	60	70	80	90	100	110	0	110
20	100	110	100	90	80	70	60	50	40	30	20	30	40	50	60	70	80	90	100	110	0

RASTGELE VERİLERİN YOLCU AKIŞ MATRİSLERİNİN BİR KISMI

Fij(kişi)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
0	0	30	55	40	35	50	50	53	45	36	56	40	51	35	38	47	42	48	32	54	58	56	41	41	44	55
1	39	0	2	15	15	4	1	4	9	2	13	6	5	1	10	11	3	9	7	9	5	5	0	12	3	5
2	38	11	0	13	15	7	7	1	14	14	5	15	6	10	7	11	1	0	11	13	11	2	3	11	1	7
3	34	10	5	0	9	4	8	10	14	2	6	14	13	5	4	9	7	13	1	6	12	11	12	2	0	12
4	33	6	8	15	0	11	7	6	15	13	13	3	7	8	12	13	1	15	11	5	6	6	8	8	2	2
5	30	7	14	15	5	0	6	15	12	2	5	1	8	3	6	12	11	10	10	2	13	12	9	5	0	9
6	35	0	0	3	13	10	0	13	1	4	10	10	10	4	14	12	14	13	4	8	9	12	2	11	13	9
7	50	0	0	15	14	7	3	0	5	4	10	3	3	4	11	11	1	13	5	3	12	12	8	13	9	8
8	52	0	0	0	13	0	11	0	0	4	2	6	8	14	1	4	11	0	11	10	4	0	4	15	11	4
9	30	0	0	0	0	0	12	5	15	0	4	10	12	4	9	1	13	3	4	9	1	4	5	8	5	3
10	33	0	0	0	0	0	10	14	15	5	0	12	3	12	8	12	8	15	11	15	5	10	15	7	13	14
11	30	0	0	0	0	0	0	0	12	3	0	0	2	1	13	7	3	3	10	11	10	5	4	11	1	4
12	48	0	0	0	0	0	0	0	13	9	0	3	0	11	15	11	2	2	13	10	4	6	10	8	1	8
13	35	0	0	0	0	0	0	0	0	0	0	7	15	0	7	12	15	9	10	15	6	14	9	5	7	8
14	40	0	0	0	0	0	0	0	0	0	0	7	5	6	0	6	6	7	7	5	4	10	15	14	2	12
15	56	0	0	0	0	0	0	0	0	0	0	11	10	9	8	0	8	7	13	7	14	11	8	2	3	6
16	48	0	0	0	0	0	0	0	0	0	0	0	0	0	14	1	0	12	15	13	9	14	8	1	5	1
17	55	0	0	0	0	0	0	0	0	0	0	0	0	0	9	12	6	0	13	14	3	3	8	3	5	13
18	51	0	0	0	0	0	0	0	0	0	0	0	0	0	13	2	8	10	0	10	1	0	15	8	10	11
19	35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	10	12	11	0	15	3	12	4	12	4
20	42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	4	2	0	0	13	1	15	3	15
21	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	1	0	3	0	6	1	3	4
22	43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	5	0	12	13	13
23	54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	2	3	0	0	7

SEÇİLEN ÖRNEKLERE YAPILAN GA DENEMELERİNİN SONUÇLARI

Örnekler		cpu time(s)	GA1 (birim*kişi)	cpu time(s)	GA2 (birim*kişi)	cpu time(s)	RGa (birim*kişi)
DSH1	min fit.	76	104120	179	103570	76	103390
	ortalama	150	105174	154	103680	68	103462
DSH4	min fit.	161	141860	121	141860	83	141860
	ortalama	198	142152	158	141860	187	141860
DSN2	min fit.	149	84630	153	84570	155	84290
	ortalama	134	85054	66	84668	119	84550
DSN6	min fit.	74	111680	192	111680	85	111680
	ortalama	122	112130	117	111824	72	111824
DSU3	min fit.	159	98990	152	98600	92	98600
	ortalama	157	99162	178	98654	172	98640
DSU5	min fit.	140	104740	63	104470	64	104470
	ortalama	88	105074	191	104928	116	104728
DMH4	min fit.	383	523900	266	515950	319	515020
	ortalama	490	524968	280	516662	370	516088
DMH7	min fit.	323	493790	355	490180	405	489000
	ortalama	429	495350	322	491102	417	490622
DMN1	min fit.	385	436790	335	430730	426	430570
	ortalama	488	439110	480	432206	273	431270
DMN4	min fit.	395	452520	496	447400	393	447190
	ortalama	498	454340	424	448152	357	447600
DMU1	min fit.	473	406250	358	396130	252	400040
	ortalama	448	410034	436	398982	227	401076
DMU6	min fit.	219	400090	271	394830	409	395950
	ortalama	208	402864	423	396636	352	396504
DLH6	min fit.	536	3679230	982	3612190	793	3598910
	ortalama	631	3693828	852	3619544	680	3619368
DLH8	min fit.	605	3370510	956	3281340	808	3290980
	ortalama	634	3380294	684	3292284	916	3307528
DLN2	min fit.	910	3070740	978	2977980	929	2976860
	ortalama	812	3095920	667	2986264	682	2992192
DLN4	min fit.	762	3251920	549	3204710	856	3232980
	ortalama	671	3279814	577	3214966	613	3247614
DLU3	min fit.	792	2668110	686	2607880	913	2600000
	ortalama	644	2680936	709	2623452	780	2639590
DLU5	min fit.	626	2629230	522	2591120	611	2619300
	ortalama	800	2651704	506	2592940	978	2623504

UÇAK TİPİ VE KAPILARIN UYUMLULUK MATRİSİ

TYPE	G101	G102	G104	G105	G106	G107	G108	G110	G111	G201	G202	G203	G204	G205	G206	G207	G208	G209	G210	G211	G212	G213	G214	G215	G216	G217	G218	G219	G220	G221	G222	G223	
SB2000	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
MD	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0	0	1	1	0	1	1	1
CRJ	0	1	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	1	0	0	1	1	1	1	0	1	1	1	1	0	
ATR	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	0	1	0	0	0	0	0	0	0
32P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
31Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
777	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	1	1	1	1	0	0	0	0
767	0	0	0	0	0	1	1	0	0	0	1	1	1	1	1	0	1	0	0	0	1	0	1	0	1	1	1	1	1	0	0	0	1
757	1	0	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1
755	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
747	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	1	1	1	1	0	1	0	1	1	0	0	0	1
743	0	0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
738	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
737	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
735	1	1	1	1	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0
734	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
733	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
732	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
727	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
720	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
442	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
350	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
340	1	0	1	1	1	0	1	0	1	0	0	0	1	1	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1
330	1	1	1	1	1	0	1	0	0	0	1	0	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1
321	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
320	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
319	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
314	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
310	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0	0	1	0	0	0	1	1	0	1	1	0	0
300	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
195	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
154	0	0	0	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	1	1	1	1	0	0	0
86	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

1 GÜNLÜK DIŞ HAT UÇUŞ ÇİZELGESİNİN BİR KISMI

Uç uş	Arr date(S)	Dep date(S)	Arr Time(S)	Dep Time(S)	ArrTime(birim zaman)	Dep Time(birim zaman)	Arr Gate	Subty pe	Airli ne	LE G
1	1	1	00:00	04:10	0	50	16	738	TK	II
2	1	1	00:00	05:30	0	66	13	321	TK	II
3	1	1	00:00	05:45	0	69	203	319	ZZ	II
4	1	2	00:00	21:00	0	540	9	300	ZZ	II
5	1	1	00:30	03:00	6	36	214	310	TK	ID
6	1	1	01:30	03:00	18	36	1	319	ZZ	II
7	1	1	01:45	05:00	21	60	223	321	OH	D
8	1	1	02:15	05:00	27	60	221	321	TK	D
9	1	1	03:00	05:30	36	66	210	320	TK	II
10	1	1	03:00	05:45	36	69	215	738	TK	II
11	1	1	03:00	06:00	36	72	222	MD	OHY	D
12	1	1	03:00	07:38	36	92	22	320	ZZ	II
13	1	1	03:00	09:30	36	114	219	MD	ZZ	II
14	1	1	03:05	05:00	37	60	212	MD	OHY	D
15	1	1	03:15	05:15	39	63	220	320	ZZ	II
16	1	1	03:15	05:30	39	66	10	738	TK	II
17	1	1	03:45	06:00	45	72	208	777	ZZ	II
18	1	1	04:00	05:45	48	69	216	738	TK	II
19	1	1	04:00	07:05	48	85	218	734	TK	D
20	1	1	04:30	12:00	54	144	20	738	TK	II
21	1	1	05:15	07:00	63	84	214	320	TK	ID
22	1	1	05:15	07:15	63	87	202	320	TK	D
23	1	1	05:25	07:10	65	86	201	738	TK	II
24	1	1	05:30	05:45	66	69	221	738	TK	ID
25	1	1	05:30	07:00	66	84	205	738	TK	D
26	1	1	05:43	07:28	69	90	206	321	ZZ	II
27	1	1	05:45	07:30	69	90	204	154	ZZ	II
28	1	1	06:00	07:30	72	90	212	319	TK	D
29	1	1	06:00	07:30	72	90	209	734	TK	D
30	1	1	06:00	07:45	72	93	220	314	TK	ID
31	1	1	06:00	07:45	72	93	217	320	TK	ID
32	1	1	06:00	08:00	72	96	215	738	TK	DI
33	1	1	06:20	07:15	76	87	203	442	ZZ	II
34	1	1	06:30	07:45	78	93	207	320	TK	ID
35	1	1	06:30	08:00	78	96	210	MD	ZZ	II
36	1	1	06:45	07:45	81	93	3	738	ZZ	II
37	1	1	06:45	08:00	81	96	222	320	ZZ	II

UÇUŞ-YOLCU SAYISI MATRİSİNİN BİR KISMI

fij matrisi(kişi)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
0	0	30	12	10	24	28	17	14	18	26	22	13	15	19	29	27	12	20	26	23	23	20	14	18
1	15	0	3	0	0	2	3	3	4	4	0	2	4	0	2	2	7	6	4	2	3	7	5	6
2	10	0	0	6	2	1	0	4	6	4	0	5	2	7	7	6	5	7	7	0	6	4	4	2
3	17	3	1	0	3	4	7	4	4	3	0	0	5	3	2	7	3	7	1	1	0	2	6	1
4	18	0	7	1	0	1	5	3	2	0	7	7	7	2	3	1	4	5	4	3	1	7	4	6
5	23	7	6	3	2	0	7	7	7	4	2	3	2	7	6	1	2	3	6	7	1	5	3	1
6	20	6	4	1	1	5	0	2	0	5	5	0	7	3	6	0	7	4	7	6	2	0	6	6
7	11	3	5	4	2	6	5	0	1	1	4	6	1	3	4	2	5	5	4	7	0	6	5	2
8	26	2	2	5	6	6	6	0	0	5	0	1	1	6	7	7	7	5	2	7	4	5	5	7
9	19	0	0	1	6	1	7	3	4	0	5	3	2	0	6	0	4	7	7	3	6	0	1	4
10	16	2	0	5	2	3	5	7	2	6	0	5	4	4	5	0	5	1	4	5	7	1	2	3
11	17	7	0	6	6	2	7	2	1	1	0	0	7	0	0	1	4	4	2	5	1	3	7	4
12	12	0	0	6	2	1	6	1	6	3	5	5	0	4	5	4	1	1	1	6	6	1	5	1
13	19	0	0	0	2	2	1	4	7	4	7	7	4	0	0	7	5	3	7	5	7	4	0	0
14	18	6	0	0	0	2	6	3	1	0	2	3	4	5	0	1	0	1	0	7	4	2	1	3
15	25	5	0	0	0	4	6	7	6	0	5	5	0	0	7	0	2	4	0	5	0	0	7	7
16	15	0	0	0	0	5	7	1	3	0	0	4	2	5	6	0	0	6	4	3	2	1	1	1
17	16	5	0	0	0	3	3	5	2	0	0	0	1	1	6	2	2	0	4	7	5	6	7	0
18	25	2	0	0	0	0	3	2	4	0	0	6	2	0	6	2	4	7	0	3	1	3	1	0
19	30	5	0	0	0	0	1	2	2	0	0	6	6	4	2	5	2	7	5	0	7	3	0	3
20	22	2	0	0	0	0	1	5	6	0	0	1	2	3	4	3	7	5	0	6	0	7	4	6
21	10	0	0	0	0	0	0	0	5	0	0	6	0	0	6	1	3	0	6	3	3	0	5	5
22	19	0	0	0	0	0	0	6	7	0	0	0	0	2	3	4	6	0	3	0	1	5	0	1
23	23	0	0	0	0	0	0	0	0	0	0	7	0	3	1	1	4	0	1	6	4	4	2	0
24	14	0	0	0	0	0	0	7	6	0	0	7	0	3	2	0	6	0	4	2	4	1	3	3
25	29	0	0	0	0	0	0	6	1	0	0	6	0	6	4	6	1	0	1	3	3	2	6	3
26	20	0	0	0	0	0	0	0	4	0	0	0	0	0	6	2	6	0	7	5	0	7	3	4

AHL DIŞ HAT KAPILARININ UZAKLIK MATRİSİ

Wkl(Birim)	0	G201	G202	G203	G204	G205	G206	G207	G208	G209	G210	G211	G212	G213	G214	G215	G216	G217	G218	G219	G220	G221	G222	G223
0 giriş	0	45	35	25	35	50	60	60	60	60	55	45	35	30	25	20	20	20	20	25	30	40	50	60
G201	40	0	10	20	30	55	65	65	65	65	60	50	40	50	50	55	60	65	70	75	80	90	100	110
G202	30	10	0	10	20	45	55	55	55	55	50	40	30	40	40	45	50	55	60	65	70	80	90	100
G203	20	20	10	0	10	35	45	45	45	45	40	30	20	30	30	35	40	45	50	55	60	70	80	90
G204	30	30	20	10	0	25	35	35	35	35	30	25	10	20	30	40	50	55	60	65	70	80	90	100
G205	40	55	45	35	25	0	10	10	10	10	5	10	20	30	40	50	60	70	80	90	100	110	120	130
G206	50	65	55	45	35	10	0	5	10	15	10	20	30	40	50	60	70	80	90	100	110	120	130	140
G207	50	65	55	45	35	10	5	0	5	10	10	20	30	40	50	60	70	80	90	100	110	120	130	140
G208	50	65	55	45	35	10	10	5	0	5	10	20	30	40	50	60	70	80	90	100	110	120	130	140
G209	50	65	55	45	35	10	15	10	5	0	10	20	30	40	50	60	70	80	90	100	110	120	130	140
G210	45	60	50	40	30	5	10	10	10	10	0	10	20	30	40	50	60	70	80	90	100	110	120	130
G211	35	50	40	30	25	10	20	20	20	20	10	0	10	20	30	40	50	60	70	80	90	100	110	120
G212	25	40	30	20	10	20	30	30	30	30	20	10	0	10	20	30	40	50	60	70	80	90	100	110
G213	25	50	40	30	20	30	40	40	40	40	30	20	10	0	10	20	30	40	50	60	70	80	90	100
G214	25	50	40	30	30	40	50	50	50	50	40	30	20	10	0	10	20	30	40	50	60	70	80	90
G215	25	55	45	35	40	50	60	60	60	60	50	40	30	20	10	0	10	20	30	40	50	60	70	80
G216	25	60	50	40	50	60	70	70	70	70	60	50	40	30	20	10	0	10	20	30	40	50	60	70
G217	25	65	55	45	55	70	80	80	80	80	70	60	50	40	30	20	10	0	10	20	30	40	50	60
G218	25	70	60	50	60	80	90	90	90	90	80	70	60	50	40	30	20	10	0	10	20	30	40	50
G219	25	75	65	55	65	90	100	100	100	100	90	80	70	60	50	40	30	20	10	0	10	20	30	40
G220	25	80	70	60	70	100	110	110	110	110	100	90	80	70	60	50	40	30	20	10	0	10	20	30
G221	35	90	80	70	80	110	120	120	120	120	110	100	90	80	70	60	50	40	30	20	10	0	10	20
G222	45	100	90	80	90	120	130	130	130	130	120	110	100	90	80	70	60	50	40	30	20	10	0	10
G223	55	110	100	90	100	130	140	140	140	140	130	120	110	100	90	80	70	60	50	40	30	20	10	0

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Şefika ARSLAN
Doğum Tarihi ve Yeri : 21.01.1986 - İstanbul
Yabancı Dili : İngilizce
E-posta : sefikaarslan86@gmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Yüksek Lisans	Endüstri Müh.	Yıldız Teknik Üniversitesi	2011
Lisans	Makine Müh.	T.C Dumlupınar Üniversitesi	2009
Lisans	Endüstri Müh.	T.C Dumlupınar Üniversitesi	2008
Lise	Fen Bilimleri	Çetinkaya Anadolu Lisesi	2003