

**BULANIK AHS YÖNTEMİ İLE FEN BİLİMLERİ
ENSTİTÜLERİ İÇİN MASTER ÖĞRENCİSİ SEÇİMİ**

Ali Kürşat ERÜMİT

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR EĞİTİMİ**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

MART 2007

ANKARA

Ali Kürşat ERÜMİT tarafından hazırlanan BULANIK AHS YÖNTEMİ İLE FEN BİLİMLERİ ENSTİTÜLERİ İÇİN MASTER ÖĞRENCİSİ SEÇİMİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. Benian TEKİNDAL
Tez Yöneticisi

Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Eğitimi Anabilim Dalında Yüksek lisans tezi olarak kabul edilmiştir.

Başkan : Prof. Dr. Ömer Faruk BAY

Üye : Doç. Dr. Şeref SAĞIROĞLU

Üye : Yrd. Doç. Dr. Benian TEKİNDAL

Tarih : 22/03/2007

Bu tez, Gazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygundur.

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Ali Kürşat ERÜMİT

**BULANIK AHS YÖNTEMİ İLE FEN BİLİMLERİ ESTİTÜLERİ İÇİN
MASTER ÖĞRENCİSİ SEÇİMİ
(Yüksek Lisans Tezi)**

Ali Kürşat ERÜMİT

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
Mart 2007**

ÖZET

Bu tezde, yüksek lisans öğrencisi seçimi problemi ele alınmıştır. Bu karar probleminin çözümünde mevcut durumun yanı sıra çok kriterli karar yöntemleri olan AHS ve Bulanık AHS algoritmaları da uygulanmıştır. Öğrenci seçimi bir süreçtir ve bu süreçte mülakat önemli bir yer tutmaktadır. Değerlendirme sırasında Bulanık AHS yönteminin kullanımı klasik yöntemlerin aksine, insana özgü karşılaştırma kriterleri olan dilsel ifadelerin de kullanılabilmesini sağlamıştır. Örnek çapı 2000 aday olarak belirlenmiştir. Çalışmada simülasyon yöntemi ile tesadüfi olarak her üç metotda da aday puanları üretilmiş, üretilen bu puanların Z değerleri ve rankları hesaplanmıştır. Daha sonrada her bir adaya ait rank değerlerine sınıf içi korelasyon analizi ve korelasyon analizi teknikleri uygulanarak, kullanılan yöntemlerin öğrenci seçimi işleminde karşılaştırılma sonuçları ve değerlendirmeler verilmiştir.

Bilim Kodu : 702.3.006
Anahtar Kelimeler : Analitik hiyerarşi süreci (ahs), bulanık ahs, yüksek lisans
Sayfa Adedi : 147
Tez Yöneticisi : Yrd. Doç. Dr. Benian TEKİNDAL

**THE ELECTION OF MASTER STUDENTS WITH FUZZY AHP METHOD
FOR INSTITUTE OF SCIENCE AND TECNOLOGY**

(M.Sc. Thesis)

Ali Kürşat ERÜMİT

**GAZİ UNIVERSITY
INSTITUTE OF SCIENCE AND TECNOLOGY**

March 2007

ABSTRACT

In this thesis, the election problem of master students is handled. Beside current situation, AHP and Fuzzy AHP algorithms which are multi criterion determination techniques are also applied to solve this determination problem. Electing students is a period and interview has an important place in this period. While evaluation, on the contrary of classical techniques, using fuzzy AHP technique provides an opportunity to use special linguistic expressions which are comparison criterions belong to humanbeing. Sample range is determined as 2000 studens. In this work, Randomize Marks is generated for each three method by simulation method, Z values and it's ranks are computed for this generated marks. After that, intraclass correlation analysis and correlation analysis techniques are applied to rank values of each student and then comparised results of used methods in the election process of student and evaluations are given.

Science Cod : 702.3.006
Key Words : Analytic hierarchy process (ahs), fuzzy ahs, master
Page Count : 147
Thesis Manager : Asist. Prof. Dr. Benian TEKİNDAL

TEŞEKKÜR

Yüksek lisans öğrenimim ve tez çalışmam süresince değerli yardım ve katkılarıyla beni yönlendiren, eğiten sayın Hocam, Danışmanım Yard. Doç. Dr. Benian TEKİNDAL'a ve kıymetli vakitlerini ayırarak yardımlarını ve desteklerini esirgemeyen Prof. Dr. Fikret GÜRBÜZ'e, ayrıca yardımlarından dolayı Özgür KOŞKAN'a, Okay ÇAMOĞLU'na ve manevi destekleriyle beni hiçbir zaman yalnız bırakmayan çok değerli Aileme ve Semra FİŞ'e teşekkürü bir borç bilirim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT	v
TEŞEKKÜR	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ	ix
ŞEKİLLERİN LİSTESİ	xii
SİMGELER VE KISALTMALAR	xiii
1. GİRİŞ	1
2. KURAMSAL TEMELLER VE KAYNAK ARAŞTIRMASI	5
2.1. Bulanık Mantığın Öğrenci Seçimindeki Yeri	5
2.2. Mülakat	5
2.3. Bulanık Mantık	7
2.3.1. Bulanık mantığın tarihçesi	7
2.3.2. Bulanık kümeler	10
2.3.3. Bulanık sayılar	15
2.3.4. Bulanık kümelerdeki işlemler	16
2.3.5. Bulanık kontrol sistemlerinin temel yapısı	19
2.3.6. Bulanık mantık uygulama örnekleri	25
2.4. Öğrenci Seçim Yöntemleri	27
2.4.1. Analitik hiyerarşi süreci (AHS)	27
2.4.2. Bulanık AHS (Fuzzy Analytic Hierarchy Process - FAHP)	34
3. MATERYAL VE METOD	45

Sayfa

3.1. Uygulanan Mülakat Sınavı ve Esasları	47
3.2. Geçerli Yöntem ile Yüksek Lisans Öğrencisi Seçimi	48
3.3. AHS Yöntemi ile Yüksek Lisans Öğrencisi Seçimi	49
3.4. Bulanık AHS Yöntemi ile Yüksek Lisans Öğrencisi Seçimi	58
4. UYGULAMA	75
5. SONUÇ	81
KAYNAKLAR	84
EKLER	89
EK-1 “Ortalama.for” isimli simülasyon programının kodları	90
EK-2 “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları	114
ÖZGEÇMİŞ	147

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. AHS değerlendirme ölçeği.....	31
Çizelge 2.2. Üçgen bulanık sayıların üyelik fonksiyonlarının üçlü tanımlanması	35
Çizelge 3.1. Histogram grafiğinde kullanılan değerler	45
Çizelge 3.2. Aday değerlendirme formu	47
Çizelge 3.3. Birinci karar vericinin adaylara verdiği puanlar	48
Çizelge 3.4. İkinci karar vericinin adaylara verdiği puanlar	48
Çizelge 3.5. Üçüncü karar vericinin adaylara verdiği puanlar	49
Çizelge 3.6. Karar vericilerin adaylara verdiği puanların ortalamaları.....	49
Çizelge 3.7. Kriterlere ait ikili kıyaslama matrisi	50
Çizelge 3.8. İkili kıyaslama sütunlar toplamı	50
Çizelge 3.9. Kriter ağırlıkları matrisi	51
Çizelge 3.10. Genel görünüş kriterine göre ikili karşılaştırma	52
Çizelge 3.11. GG kriterine göre aday ağırlıkları matrisi.....	52
Çizelge 3.12. Anlama ve anlatma yeteneği göre ikili karşılaştırma.....	53
Çizelge 3.13. Anlama ve anlatma yeteneği kriterine göre aday ağırlıkları matrisi.....	53
Çizelge 3.14. Aile ve sosyal durum kriterine göre ikili karşılaştırma.....	54
Çizelge 3.15. Aile ve sosyal durum kriterine göre aday ağırlıkları matrisi	54
Çizelge 3.16. Psikolojik yapı kriterine göre ikili karşılaştırma.....	55
Çizelge 3.17. Psikolojik yapı kriterine göre aday ağırlıkları matrisi	55
Çizelge 3.18. Bilimsel yeterlilik kriterine göre ikili karşılaştırma.....	56

Çizelge	Sayfa
Çizelge 3.19. Bilimsel yeterlilik kriterine göre aday ağırlıkları matrisi	56
Çizelge 3.20. Birleştirilmiş karar matrisi	57
Çizelge 3.21. Kriterler için karar vericinin verdiği puanlar	58
Çizelge 3.22. Adaylar için karar vericinin verdiği puanlar	59
Çizelge 3.23. Kriter puanlarına ait üyelik fonksiyonlarının oluşturulması.....	60
Çizelge 3.24. Karar vericinin verdiği puanlara göre kriterlerin aitlik kümeleri ve dereceleri	60
Çizelge 3.25. Kriterlere ait bulanık girişlerin karşılaştırılmalarına ait kural tabanı	61
Çizelge 3.26. Kriterlere ait bulanık girişlerin karşılaştırılması	63
Çizelge 3.27. Aday puanlarına ait üyelik fonksiyonlarının oluşturulması.....	63
Çizelge 3.28. Karar vericinin verdiği puanlara göre adayların aitlik kümeleri.....	64
Çizelge 3.29. Adaylara ait bulanık girişlerin karşılaştırılmalarına ait kural tabanı	64
Çizelge 3.30. Anlama anlatma yeteneği kriteri için adaylara ait bulanık girişlerin karşılaştırılması.....	67
Çizelge 3.31. Genel görünüş kriteri için adaylara ait bulanık girişlerin karşılaştırılması	67
Çizelge 3.32. Psikolojik yapı kriteri için adaylara ait bulanık girişlerin karşılaştırılması	67
Çizelge 3.33. Ailevi ve sosyal durum kriteri için adaylara ait bulanık girişlerin karşılaştırılması	67
Çizelge 3.34. Bilimsel yeterlilik kriteri için adaylara ait bulanık girişlerin karşılaştırılması	68
Çizelge 3.35. Dilsel ifade skalası	69
Çizelge 3.36. Dilsel ifade skalasına göre kriterlerin bulanık değerlendirilmesi matrisi.....	69

Çizelge	Sayfa
Çizelge 3.37. Anlama ve anlatma yeteneği kriterine göre bulanık değerlendirme matrisi	72
Çizelge 3.38. Genel görünüş kriterine göre bulanık değerlendirme matrisi	72
Çizelge 3.39. Psikolojik yapısı kriterine göre bulanık değerlendirme matrisi	73
Çizelge 3.40. Aile ve sosyal durum kriterine göre bulanık değerlendirme matrisi	73
Çizelge 3.41. Bilimsel yeterliliği kriterine göre bulanık değerlendirme matrisi	74
Çizelge 3.42. Tüm adaylar için hesaplanan tüm kriter ağırlıkları	74
Çizelge 4.1. Sınıf içi korelasyon katsayısının kabul edilebilir seviyeleri	78

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. 5 ile 8 arasındaki reel sayıları kapsayan küme.....	13
Şekil 2.2. Genç insanlar kümesine(B) değişik yaş gruplarının üyelik durumu.....	14
Şekil 2.3. Bir üçgen bulanık sayı, $\tilde{M}$	15
Şekil 2.4. 5 ile 8 arasında bulanık bir değer	17
Şekil 2.5. 4 civarında bulanık bir değer	17
Şekil 2.6. A ve B bulanık kümelerinin kesişimi	18
Şekil 2.7. A ve B bulanık kümelerinin birleşimi.....	18
Şekil 2.8. Bulanık bir kümenin tersi.....	19
Şekil 2.9. Bulanık mantık denetleyicinin (FLC) temel yapısı.....	20
Şekil 2.10. Bulanık kural tabanlı çıkarım sistem yapısı.....	23
Şekil 2.11. Basit bir AHS modeli.....	29
Şekil 2.12. Üçgen bulanık sayılar	35
Şekil 2.13. M_1 ve M_2 'nin kesişimi	40
Şekil 3.1. Histogram grafiği	46
Şekil 3.2. Kriter puanlarına ait üçgen üyelik kümeleri ve üyelik fonksiyonu dereceleri	59
Şekil 3.3. Aday puanlarına ait üçgen üyelik kümeleri ve üyelik fonksiyonu dereceleri	63

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur

Simge	Açıklama
$\tilde{A}$	A bulanık kümesi
c_i	Tutarsızlık endeks değeri
cr	Tutarlılık oranı
icc	Sınıf içi korelasyon katsayısı
$\tilde{m}$	Bir üçgen bulanık sayı
ot	Objektif kriter ağırlığı
r_i	Rastsal endeks değeri
s_i	Sentetik boyut analiz değeri
sup	Asgari üst sınır
v	Olabilirlik derecesi
w'	Ağırlık vektörü
μ_a	Klasik küme üyelik fonksiyonu
$\mu_{\tilde{a}}$	Bulanık küme üyelik fonksiyonu
λ_{max}	Maksimum özdeğer

Kısaltmalar	Açıklama
FBE	Fen Bilimleri Enstitüsü
AHP	Analytic Hierarchy Process
AHS	Analitik Hiyerarşi Süreci
BAHS	Bulanık Analitik Hiyerarşi Süreci
FLC	Bulanık Mantık Denetleyicisi

1. GİRİŞ

Personel seçim işlemi tüm kişi, kurum ve şirketler için sürekli karşılaşılan bir problemdir. Her türlü karar probleminde bir seçim yapılması söz konusu olabilir. Bu nedenle bir memurun alacağı araba için alternatifler arasından yapacağı seçim işlemi de, bir şirket için eleman ihtiyacını karşılamak amacıyla, şirkete başvuran adaylar arasından yapılacak seçim işlemi de, Fen Bilimleri Enstitüleri için yüksek lisans öğrencisi seçimi de aynı algoritmalarla gerçekleştirilebilir.

Öğrenci seçim işlemi, ilgili problemin giderilmesi amacıyla gerekli öğeler için alternatifler arasından problemi giderecek en iyi alternatife karar verilmesidir [1]. Başka bir anlatımla, öğrenci seçim prosedürünün amacı denk olmayan özelliklere sahip adayların derece derece elenmesidir. Seçim işlemi her türlü alanda gerçekleştirildiği için çok önemli ve özen gösterilmesi gereken bir konudur [2].

Fen Bilimleri Enstitüsü için yüksek lisans öğrencisi seçim işlemi öğrenci ihtiyacı ile başlayan ve seçilen kişilerin kabulü ile son bulan bir süreci kapsar. Fen Bilimleri Enstitüsü için yüksek lisans öğrencisi seçim işleminde karşılaşılan süreçte bilgiye dayanan çeşitli test ve yazılı sınavlar uygulanarak karar verilebilmektedir. Ancak en yaygın yöntem olarak mülakat ön plana çıkmaktadır [3].

Mülakat, bir yüksek lisans programı için başvuran adaylar arasından, enstitü için en uygun olan, en fazla uyum sağlayabilecek ve ihtiyaçlara en fazla cevap verebilecek adayın seçimi için kullanılan, aday ve enstitü tarafından adayları değerlendirmekle görevlendirilmiş kişi veya kişilerin yüz yüze görüşmek ve konuşmaları suretiyle gerçekleştirilen ve öğrenci seçimi için aday değerlendirme amacıyla kullanılan bir yöntemdir [4].

Mülakat sürecinde karar vericilerin insan olması ve değerlendirme işleminin konunun uzmanları tarafından hissi ve tecrübelere dayalı olarak yapılıyor olması nesnelliğin kaybolmasına neden olmaktadır. Bu nedenle öğrenci seçim işleminin daha doğru yapılabilmesi ve hata riskini en aza indirebilmek amacı ile klasik

yöntemlerden farklı çeşitli yöntemler geliştirilmesine yönelik çalışmalar sürmektedir [5].

Bu çalışmada öğrenci seçimi için çeşitli yaklaşımlardan bahsedilecektir. Bahsedilecek bu yaklaşımlar konuya özel olmamakla beraber algoritmanın uyarlanabilirliğinde görülen esneklik ve konunun genellenebilmesi sayesinde seçim amaçlı bu yöntemlerin, çalışma içerisinde kullanılabilmesi sağlanmıştır.

Fen Bilimleri Enstitüleri için yüksek lisans öğrencisi seçim işlemini simüle etmek amacı ile yapılan simülasyonda kullanılan algoritmalar literatürde sıkça yer almaları, kullanım alanlarının geniş olması, güncel ve ihtiyaçlara uygun gelişimlere açık olmaları nedeni ile kullanılan klasik yöntemin algoritması, Analitik Hiyerarşi Süreci Algoritması ve Bulanık Analitik Hiyerarşi Süreci algoritmasıdır.

Çalışmanın amacı Klasik Yöntem ile çok kriterli karar verme yöntemlerinden olan Analitik Hiyerarşi Süreci (AHS) ve Bulanık Analitik Hiyerarşi Süreci (BAHS) yöntemlerini fen bilimleri enstitüleri için yüksek lisans düzeyinde öğrenci seçimi problemine bir simülasyon yardımı ile uygulayıp, 2000 aday arasından ayrı ayrı üç yöntem tarafından seçilecek ilk 100 bireyin üzerinde, yöntemlerin seçim benzerliklerini ölçmektir.

Seçim sırasında karşılaşılan öznel değerlendirme kriterleri seçim işleminin bulanıklaşmasına neden olmaktadır. Seçim işlemi sırasında klasik yöntemin kullanılmasında karşılaşılan en büyük sorun bu bulanıklığı yani insana özgü hissi değerlendirmeyi çözüme katamamasıdır. Son yıllarda çok kriterli seçim, karar problemlerinde sıkça kullanılan yöntemlerden biri AHS yöntemidir. Bu yöntemde izlenilen yol ise; öncelikle problemi oluşturan aşamaların, alt amaçlardan ana hedefe doğru bir hiyerarşi içerisinde yeniden yorumlanması, oluşturulan hiyerarşi içerisinde kriterlerin ikili mukayeseler ile öncelik vektörlerinin oluşturulması, adayların yine ikili mukayeseler ile ağırlıklarının bulunması ve öncelik vektörleri ve aday ağırlıkları ile sentezlenme aşamalarını ihtiva eder. Bununla beraber AHS yöntemi de bulanıklığı tam olarak yansıtamamaktadır bu amaçla bulanık küme teorisinin kullanıldığı bir

yöntem düşünülmüştür. Bulanık küme teorisi belirsizliğin matematiksel bir formda modellenerek hesaplamalara dahil edilmesi amacıyla geliştirilmiş bir teoridir [6]. Bu teorinin kullanıldığı bir yöntem olarak Bulanık AHS yönteminin son yıllarda yaygın olarak kullanıldığı görülmektedir.

Tezin ilk bölümünde öncelikle yüksek lisans öğrencisi seçimi işleminin öznelliğinden dolayı değerlendirme işleminde bulanık yöntemlerin kullanımına imkan veren mülakat işleminden bahsedilmiştir.

Tezin ikinci bölümünde bulanık mantığa değinilerek tarihçesinden, bulanık küme ve bulanık sayı teorilerinden, bulanık kümelerde gerçekleştirilen işlemlerden, bir bulanık kontrol sistemini oluşturan bölümlerden bahsedilmiştir.

Tezin üçüncü bölümünde yüksek lisans öğrencisi seçimi için simülasyonda kullanacağımız yöntemlerden bahsedilmiştir. Bu konu altında çok kriterli karar verme yöntemleri olan AHS ve Bulanık AHS yöntemleri anlatılmış ve yöntemlerin uygulanmasında kullanılan algoritmalar açıklanmıştır.

Tezin dördüncü bölümünde ise öncelikle Gazi Üniversitesi Fen Bilimleri Enstitüsünde yüksek lisans öğrencisi seçimi ile ilgili uygulanan mevcut duruma göre elde edilmiş olan mülakat notlarının değerlendirilmesi yapılmış ve algoritmaların işletilmesi sonucu elde edilecek değerlerin karşılaştırılması için izlenecek istatistiksel yöntemler belirlenmiştir. Daha sonra matematiksel bir model oluşturabilmek için bir aday değerlendirme formu hazırlanmıştır. Oluşturulan aday değerlendirme formu kullanılarak 5 adayın ve 3 karar vericinin olduğunu varsaydığımız bir yüksek lisans öğrencisi seçimi işlemi, her üç yöntemlede yapılmıştır. Daha sonra ise halen kullanımda olan yönteme göre, AHS yöntemine göre ve Bulanık AHS yöntemine göre simülasyon işletilerek üç yöntemlede öğrenci seçimi işlemi yapılmış ve elde edilen veriler, belirlenmiş istatistiksel yöntemler yardımı ile karşılaştırılarak yöntemlerin benzerliklerinin ölçülmesi sağlanmıştır.

Tezin sonuç bölümünde ise elde edilen karşılaştırma sonuçları yorumlanarak üç yöntemin öğrenci seçme işlemini en doğru ve gerçekçi yapma yönünden birbirine göre durumları belirlenmiş ve uygulanan bu yöntemler eleştirilerek üstün ve eksik yönleri açıklanmıştır.

2. KURAMSAL TEMELLER VE KAYNAK ARAŞTIRMASI

2.1. Bulanık Mantığın Öğrenci Seçimindeki Yeri

Bulanık mantık, özünde insana özgü esnek karar verme özelliğini makinelere kazandırmayı amaçlayan bir yöntemdir. Klasik matematikte yalnızca 0-1 kesinliği vardır, oysaki günlük hayatta insanlar arasında sadece bu kesinlik yoktur, verdiğimiz kararlarda yalnızca siyah yada beyazı değil ikisi arasında kalan grinin çok farklı tonlarını da kullanırız. Bu nedenle çok farklı alanlarda olduğu gibi yüksek lisans düzeyinde öğrenci seçiminde de bulanık mantık algoritmalarından faydalanılabilir.

Yüksek lisans düzeyinde öğrenci seçiminde karşılaşılan mülakat süreci karar vericilerin adayları değerlendirmeleri amacıyla yüz yüze yapılan bir görüşmedir ve eğer karar vericilerin sayısı birden fazlaysa karar vericiler arasında farklı sonuçlara neden olabilecek insana özgü ve kesin olmayan değerlendirme özelliklerinin devreye girdiği bir süreçtir.

İşte bu nedenle bulanık mantık mülakat sürecinde bize çok yardımcı olacak bir yöntem olarak karşımıza çıkmaktadır.

2.2. Mülakat

Mülakat temel olarak buluşma, görüşme anlamına gelmektedir [5]. Mülakat örneğin yüksek lisans öğrenimi amacıyla bir üniversitenin ilgili enstitüsüne başvuran adaylar arasından, enstitü için en uygun olan, en fazla uyum sağlayabilecek ve öğrenimin gerek duyduğu ihtiyaçlara en fazla cevap verebilecek adayın seçimi için kullanılan ve aday ile enstitü tarafından adayları değerlendirmekle görevlendirilmiş kişi veya kişilerin yüz yüze görüşmek ve konuşmaları suretiyle gerçekleştirilen ve öğrenci seçimi için aday değerlendirme amacıyla kullanılan bir yöntemdir [7].

Sürecin tüm aşamaları uygulandığında ise mülakat, enstitüye alınacak kişi ile enstitüye alınma kararında etkili olacak kişi ya da kişileri karşı karşıya getirilmesi ve

geniş bir karşılıklı görüşme ortamı yaratılması nedeniyle, öğrenci seçiminde nihai kararın verilmesi için en önemli ve etkili aşamayı oluşturur. Birçok üniversite, bir adayın ne kadar başarılı olacağını tahmin etmenin en güzel yolunun o adayla yüz yüze görüşmek olduğuna inanır. Bu güven, görüşmeyi yapan kişide görüşme sırasında uyanan sezgilere dayanır [2].

Mülakatta ana amaç, adayın bireysel özellikler ile yüksek lisans öğrenimi yapmak istediği alanla ilgili çalışma düzenine ne ölçüde uyabileceğinin saptanmasıdır. Bu bakımdan mülakatta, adayın kişilik özellikleri ve nitelikleri hakkında yeterli bilginin sağlanmasına çalışılır.

Adayın bizzat görüp konuşmak suretiyle onun kişiliği ve yetenekleri hakkında daha iyi fikir alınmasını sağlayan mülakatın değişik türleri vardır [2].

Mülakat türleri şunlardır:

Planlı mülakat: Önceden hazırlanmış belirli soruların bulunduğu bir form kullanılarak yapılır. Formdaki sorular bir uzmanlar grubu tarafından alanın ve enstitünün ihtiyaçlarına göre hazırlanır. Sorular görüşme sırasında her adaya aynen sorulur ve alınan cevaplar mülakat formuna kaydedilir. Bu yaklaşım esneklikten uzaktır. Ancak, adaylar arasında karşılaştırma yapmak daha kolay ve objektif olur. Fakat sorular standart olduğu için adayların özel ve yaratıcı yönleri analiz edilemeyebilir [8, 9].

Plansız mülakat: Adaya sorulacak sorular önceden belirlenmez. Bu mülakatta görüşmeciler, duruma göre sorulması uygun olan soruları yeri geldikçe yöneltme yoluyla adayı konuşturarak bilgi alırlar. Sorular, alanın kapsamına göre değişir. Görüşmeyi yapanlar maksatlı olarak soracakları sorularla adayın konuşma biçimini, olaylara yaklaşımını, kendisini veya fikirlerini savunma biçimini saptamaya çalışırlar. Bu yöntem ancak bu konuda yetişmiş kişiler tarafından uygulanırsa yararlı olur. Yöntemde, mülakatçı gerekli yönlendirmeyi yapıp, daha çok dinlemede kalmalıdır. Bu yöntem ile elde edilen bilgilerin geçerliliği pek yüksek değildir. Çünkü adaylardan her birine sorulan sorular farklıdır ve adaylar arası karşılaştırma

standart olmaz. Seçim tümüyle görüşmecinin aday hakkındaki kanaatine bağlı kalır. Başka bir deyişle, görüşmeci adaylar arasından zihnine uygun olanları bulur [8, 9].

Karma mülakat: Mülakatı biraz daha tutarlı hale getirmek amacıyla planlı ve plansız mülakatı birlikte uygulandığı karma mülakat yöntemini tercih etmektedirler. Karma mülakat öncesi hazırlanan sorular mülakatın başlangıcında adaylara sorulmakta; görüşmenin seyrine göre mülakatı yapanlar gerekli görürlerse, aldıkları cevaplara göre ilave sorular sorarak plansız mülakat yöntemine yönelmektedirler. Bu yöntemde, plansız görüşme tekniklerinden de yararlanarak mülakatın derinleştirilmesi ve adaylar hakkında biraz daha geniş boyutlu bilgi elde edilmesi mümkün olmaktadır [8, 9].

Sorun Çözmeye Dönük Mülakat: Adaylara alanlarına uygun bir sorun verilir ve onların bu konudaki görüşleri alınır. Adayların cevapları ile doğru çözüm yaklaşımları birlikte değerlendirilir. Adaylar, cevaplarının tutarlılığı ölçüsünde başarılı sayılır. Bu yöntemde adaylara çözmeleri için verilen sorunların çok iyi saptanması gerekir. Çünkü her adayın her konuya ilgi duyması beklenemez. Bu mülakattan elde edilecek sonuçların geçerliliği, adaylara verilen sorunlar ile ileride karşılaşabilecekleri konuların uyumlu olmasına bağlıdır [8, 9].

Yoğunlaştırılmış Mülakat: Adayı belirli konularda derinlemesine analiz edecek soruların ağırlıklı olduğu görüşme türüdür. Neden sahip olduğu eğitimi yaptığı, neden yüksek lisans eğitimine başvurduğu, neden mevcut hobilerine sahip olduğu... gibi [8, 9].

2.3. Bulanık Mantık

2.3.1. Bulanık mantığın tarihçesi

Bulanık mantık (Fuzzy Logic) kavramı ilk kez 1965 yılında California Berkeley Üniversitesinden Prof. Lotfi A.Zadeh'in bu konu üzerinde ilk makalelerini yayınlamasıyla duyuldu. O tarihten sonra önemi gittikçe artarak günümüze kadar

gelen bulanık mantık, belirsizliklerin anlatımı ve belirsizliklerle çalışılabilmesi için kurulmuş bir matematik düzen olarak tanımlanabilir. Bilindiği gibi istatistikte ve olasılık kuramında, belirsizliklerle değil kesinliklerle çalışılır. Ama insanın yaşadığı ortam daha çok belirsizliklerle doludur. Bu yüzden insanoğlunun sonuç çıkarabilme yeteneğini anlayabilmek için belirsizliklerle çalışmak gereklidir [10].

Klasik mantıkta matematiksel bir sonuç çıkarma veya karar verme, sıfır veya birlerle ifade edilen kesin yanlış ve doğru önermeler veya var-yok gibi kesin gerçeklerle yapılır. Fakat bu şekildeki ikili mantık gerçek dünyanın belirsizliğini ifade etmede yetersizdir. Gerçek hayatta çözümler genelde kısmen doğru veya belli bir olasılıkla doğru veya yanlış şeklinde değerlendirilir. Bulanık mantık, adından anlaşılacağı gibi mantık kurallarının esnek veya bulanık bir şekilde uygulanmasını sağlar. Klasik mantıktaki doğru ve yanlış, var ve yok ya da 1 ve 0 gibi ani geçişleri yumuşatmaktadır. Bu iki değer arasında da değerler bulunduğu, insan hayatına veya kararlarına bakıldığında hemen hemen hiçbir şeyin kesinlikle doğru veya kesinlikle yanlış olmadığı göz önüne alınırsa böyle bir yaklaşımın farklılığı daha açık ortaya çıkmaktadır [10]. Örnek olarak “ $1+1=2$ ” kesin bir gerçektir ve doğruluk değeri birdir. Fakat “otlar yeşildir” ifadesi tam doğru olmamakla birlikte yanlış da değildir ve biz bunu klasik mantıkta tam olarak ifade edemeyiz. İnsan beyni ise “açık yeşil”, “serin hava” veya “yüksek hız” gibi matematiksel olarak kesin olmayan belirsiz ya da değer yargıları içeren “bulanık dilsel niteleyicileri” kullanarak sağduyulu kararlar verebilir ve sonuç çıkartabilir.

Bulanık mantık, temel olarak yaklaşıklık ve kesin olmama gibi insan düşüncesinde ve doğada var olan belirsizliği kullanmaktadır ve kesin doğru veya kesin yanlış yerine doğal derece doğru ya da yanlışlık belirten tanımlar kullanır. Böylelikle bulanık mantıkta, günlük konuşmalarda kullanılan belirsiz ifadeler tanımlanabilmekte ve bu dilsel niteleyiciler kullanılarak insan benzeri sonuç çıkarma işlemleri gerçekleştirilebilmektedir. Bu sayede bulanık mantık, bilgisayarlara dolayısıyla kontrol sistemlerine insan düşünme ve karar verebilme yeteneğinin kazandırılmasında başarılı bir yol sağlamaktadır [11].

1920'li yıllarda Polonya'lı mantıkçı Jan Łukasiewicz, önermelerin sadece bir veya sıfır doğruluk değeri alabildiği klasik mantıktan farklı olarak önermelerin bir ve sıfır arasında da kesirli doğruluk değeri alabildiği “çok değerli” mantık ilkelerini oluşturdu. 1937'de ise kuantum felsefecisi Max Black yayımlanan bir makalesinde liste ya da nesnelerden oluşan kümelere “çok değerli mantığı” uygulayarak ilk bulanık küme eğrilerini çizdi. Daha sonra 1965'de Lotfi A. Zadeh bu alana adını veren “Bulanık Kümeler” adlı çığır açıcı yazısını yayınladı. Zadeh bu makalede, bir kümenin tüm elemanlarına Łukasiewicz'in mantığını uygulayarak bulanık kümeler için eksiksiz bir cebir geliştirdi. Buna rağmen, Ebrahim H. Mamdani'nin bir buhar makinesi için bulanık mantıkla çalışan bir kontrol sistemi gerçekleştirdiği 1970'li yılların ortalarına dek bulanık kümeler kullanım alanı bulamadı. Ancak bu tarihten itibaren, “bulanık mantık” terimi, bulanık kümeler yardımıyla akıl yürüten herhangi bir matematik ya da bilgisayar sistemini ifade etmektedir [12].

Bulanık mantığın uygulama alanları kontrol sistemleriyle sınırlı değildir. Geliştirilen son teoremler bulanık mantığın ilke olarak, ister mühendislik, ister biyoloji ve hatta isterse ekonomi olsun, her türlü alanda sürekli sistemleri modellemek için kullanılabileceğini göstermektedir. Çoğu alanda bulanık mantık temelli sağduyulu modellerin standart matematiksel modellerden daha yararlı ya da daha kesin sonuçlar verdiği görülmektedir [10].

Fuzzy kuramının merkez kavramı fuzzy kümeleridir. Küme kavramı kulağa biraz matematiksel gelebilir ama anlaşılması kolaydır. Örneğin “orta yaş” kavramını inceleyecek olursak, bu kavramın sınırlarının kişiden kişiye değişiklik gösterdiğini görürüz. Kesin sınırlar söz konusu olmadığı için kavramı matematiksel olarak da kolayca formüle edemeyiz. Ama genel olarak 35 ile 55 yaşları orta yaşlılık sınırları olarak düşünülebilir. Bu kavramı grafik olarak ifade etmek istediğimizde karşımıza bir eğri çıkacaktır. Bu eğriye “aitlik eğrisi” adı verilir ve kavram içinde hangi değerin hangi ağırlıkta olduğunu gösterir [12].

Bir fuzzy kümesi kendi aitlik fonksiyonu ile açık olarak temsil edilebilir. Aitlik fonksiyonu 0 ile 1 arasındaki her değeri alabilir. Böyle bir aitlik fonksiyonu ile

“kesinlikle ait” veya “kesinlikle ait değil” arasında istenilen incelikte ayarlama yapmak mümkündür.

Bulanık mantık ile matematik arasındaki temel fark bilinen anlamda matematiğin sadece aşırı uç değerlerine izin vermesidir. Klasik matematiksel yöntemlerle karmaşık sistemleri modellemek ve kontrol etmek işte bu yüzden zordur, çünkü veriler tam olmalıdır. Bulanık mantık kişiyi bu zorunluluktan kurtarır ve daha niteliksel bir tanımlama olanağı sağlar. Bir kişi için 38,5 yaşında demektense sadece orta yaşlı demek birçok uygulama için yeterli bir veridir. Böylece azımsanamayacak ölçüde bir bilgi indirgenmesi söz konusu olacak ve matematiksel bir tanımlama yerine daha kolay anlaşılabilen niteliksel bir tanımlama yapılabilecektir [11].

Bulanık mantıkta fuzzy kümeleri kadar önemli bir diğer kavramda linguistik değişken kavramıdır. Linguistik değişken “sıcak” veya “soğuk” gibi kelimeler ve ifadelerle tanımlanabilen değişkenlerdir. Bir linguistik değişkenin değerleri fuzzy kümeleri ile ifade edilir. Örneğin oda sıcaklığı linguistik değişken için “sıcak”, “soğuk” ve “çok sıcak” ifadelerini alabilir. Bu üç ifadenin her biri ayrı ayrı fuzzy kümeleri ile modellenir [9].

Bulanık mantığın uygulama alanları çok geniştir. Sağladığı en büyük fayda ise “insana özgü tecrübe ile öğrenme” olayının kolayca modellenebilmesi ve belirsiz kavramların bile matematiksel olarak ifade edilebilmesine olanak tanımasıdır. Bu nedenle lineer olmayan sistemlere yaklaşım yapabilmek için özellikle uygundur [12].

2.3.2. Bulanık kümeler

Klasik mantık ile bulanık mantık arasındaki farkı en açık şekilde iki mantık sisteminin temelini oluşturan, kullandıkları küme teorilerini incelemekte görebiliriz. Klasik ikili mantığın dayandığı standart küme teorisinde, herhangi bir nesne bir kümeye ya aittir ya da değildir. Bunun ortası yoktur. Yani bir nesnenin bir kümeye aitlik derecesini gösteren “üyelik derecesi” ya “birdir” ya da “sıfırdır”. Örneğin: Beş sayısı kesinlikle tek sayılar kümesine aittir ve asla bu kümenin tümleyeni olan çift

sayılar kümesine ait değildir. Ayrıca bu birbirinin tümleyeni olan iki kümeden hiç birine ait olmaması da söz konusu değildir. Bu ilke hem klasik mantığın yapısını korumakta hem de bir nesnenin aynı anda hem bir şey olup hem de o şey olmaması çelişkisini ortadan kaldırmaktadır [11].

Bulanık mantığın kullanıldığı “bulanık” ya da başka bir deyişle “çok değerli” kümelerde ise, nesneler bulanık bir kümeye ve bu kümenin tümleyeni olan kümeye aynı anda sonsuz farklı derecelerde ait olabilir. Buradaki tek sınırlama bu iki üyelik derecesinin toplamının bir olması gerektiğidir. Örneğin eğer hava, %20 serin ise %80 serin değil olmalıdır. Bulanık mantık bu yolla, klasik mantığı geçersizleştiren “havanın %100 serin iken %100 serin değil” ifadesinde olduğu gibi, çift çelişkiden kaçınır. Bunun yerine nesneler birbirlerinin tamamlayıcısı olan iki bulanık kümeye, hava %20 serin iken %80 serin-değil olması gibi aynı anda kısmen aittir. Ve bu da dilsel (linguistik) niteleyicilerde olduğu gibi kısmi çelişkilere sebep olur. Bulanık kümeler nesnelerin %100 üye olup-olmadığı özel durumlar için klasik kümenin özelliklerini taşır ve klasik küme teorisi işlemleri uygulanabilir. Önemli bir nokta da bulanık kümedeki üyelik derecelerinin olasılık yüzdeleriyle aynı şey olmadığıdır [12].

Olasılığı gösteren sayılar bir şeyin olup olmayacağının ölçütüdür. Bulanık sayıların üyelik dereceleri ise bir olayın ne dereceye kadar olduğunu, bir koşulun ne dereceye kadar gerçekleştiğini gösterir; sabah hava %30 olasılıkla serin olacak önermesi sabah havanın serin olma ihtimalini gösterir; sabah hava %30 serindi veya hava %30 serin gibi ifadeler ise geçmişte veya o andaki havanın serinlik derecesini göstermektedir. Bu konudaki ikinci bir örnek de şu şekilde olabilir: X bütün sıvılardan oluşan bir küme olsun. X'in bir alt kümesi olan Y kümesi ise içilmesi zararsız olan sıvıların kümesi olsun. Farzedelim ki elimizde A ve B olmak üzere etiketi kapatılmış dolu iki şişe var. Bize verilen bilgi ise A şişesindeki sıvının Y kümesine üyelik derecesinin %91 ve B şişesindeki sıvının Y kümesine üye olma olasılığının %91 olduğudur. Eğer bu iki şişeden birini içmek zorunda kalırsak A şişesinin tehlikesi B'ye göre daha azdır. Çünkü B şişesi onda bir olasılıkla zararlı bir sıvıyla doludur ve B şişesini içersek ani bir ölümle karşılaşabiliriz. Fakat A şişesine ait %91'lik üyelik derecesi

bize şişedeki sıvının zararsız sıvılara örneğin saf suya %91 oranında benzediğini ve asla hidroklorik asit (HCl) gibi zararlı bir sıvı olmadığını ifade eder. Bu yüzden bu sıvının bize verebileceği zarar sınırlıdır ve ölüm tehlikesi yoktur. Hatta bulanık bir su olarak düşünülebilir. Şimdi etiketleri açıp olasılık ve üyelik derecesinin durumunu tekrar inceleyelim. Etiketler açıldığında A şişesinde çamurlu su B şişesinde ise hidroklorik asit(HCl) çıktığını farzedelim. Buna göre A şişesindeki çamurlu suyun Y kümesine üyelik değeri yine %91'dir; fakat B şişesinin Y kümesine üye olma olasılığı sıfırdır. Sonuç olarak üyelik derecesi, bir nesnenin herhangi bir kümeye ne derece üye olduğunu, ne derece benzediğini veya bir olayın bir şartın ne derece var olduğunu gösterir ve aldığı değer sabittir. Buna karşılık olasılık ise bir olay gerçekleşmeden önce olup olmayacağı veya ne derece olabileceği hakkında bilgi verir ve zamana bağlı olarak duruma göre değişebilir. Üyelik derecesi daha çok bir olayın benzerlik derecesini ifade ederken, olasılık olma sıklığını gösterir [13,14].

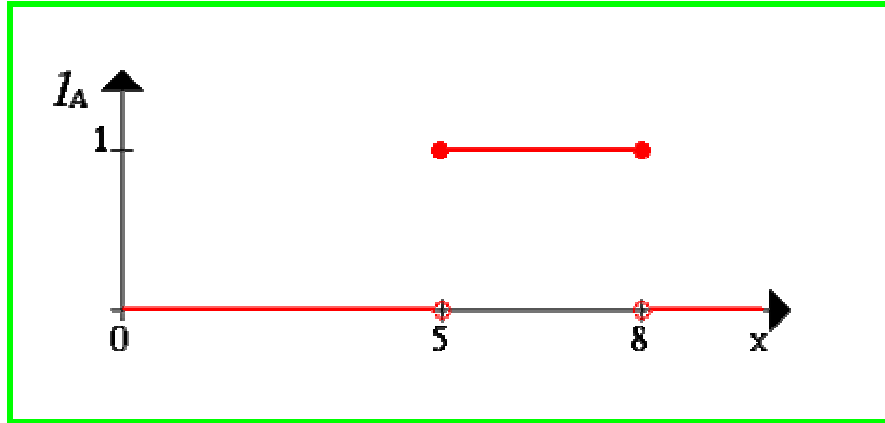
Klasik küme ile bulanık küme arasındaki farkı daha iyi anlamak için bir örnek üzerinde gösterelim.

Örnek ;

Aşağıda Evrensel kümede 0 ila 10 arasındaki tüm reel sayıları kapsayan bir X kümesi üzerinde 5 ile 8 arasında reel sayılardan oluşan bir A altkümesi tanımlanmıştır [9].

$$A = [5,8]$$

A alt kümesini X kümesi üzerinde elemanlarının A kümesinde bulunup bulunmamasına göre değişecek şekilde 1 ve 0 ile gösterilmiştir. Bu tanımlama sonucu oluşacak şekil aşağıdaki gibi olacaktır;



Şekil 2.1. 5 ile 8 arasındaki reel sayıları kapsayan küme [9]

$\mu_A(r)$ üyelik fonksiyonu Şekil 2.1’de görüldüğü gibi yalnızca iki farklı değer alabilir ve tüm reel sayıları “0” ve “1” sayılarına dönüştürür. Bu dönüşüm r reel sayısının A kümesine üye olup olmadığını (var-yok) ifade eder. Yani, eğer $\mu_A(p)=1$ ise “ p ” sayısı A kümesinin üyesidir ve $\mu_A(p)=0$ ise “ p ” sayısı A kümesinin üyesi değildir. Klasik kümede üyelik durumunu gösteren üyelik fonksiyonu ikili mantıkta bir önermenin doğru olup olmadığını ifade eder [9].

Klasik mantığın bazı durumlarda içine düştüğü tanımlama eksikliğine bir örnek daha verelim [9].

Örnek ;

Aşağıda genç insanlar kümesinin biçimsel gösterimi verilmiştir;

$$B = \{\text{genç insanlar kümesi}\}$$

B kümesinin keskin küme gösterimi

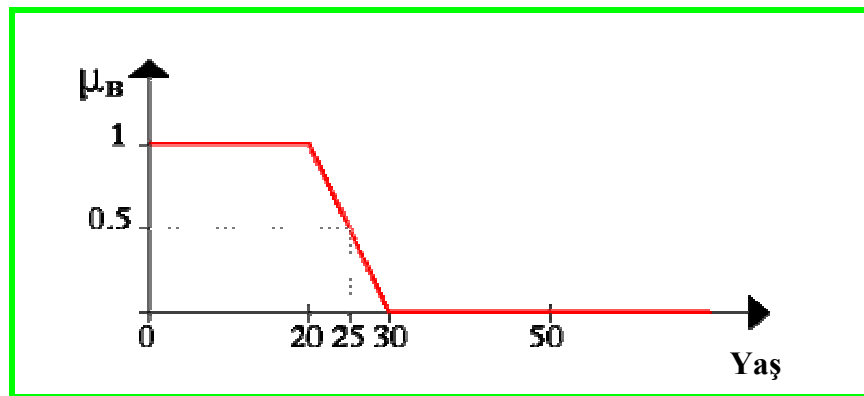
$$B = [0,20] \text{ şeklinde olacaktır.}$$

Burada yaş kavramı genellikle 0'dan başladığı için kümenin alt değeri için 0 değeri verilmiştir. Ancak üst değer için sınır değer kabul etmek oldukça zordur. Burada üst sınır olarak 20 yaş kabul edilmiştir.

Niçin bir insan 20. yaş gününde gençken ertesi gün genç olarak kabul edilmiyor? Bunun yapısal bir problem olduğu açıktır zira kümenin üst değeri farklı bir değere de taşınsa aynı soruyla tekrar karşılaşılacağı muhakkaktır.

B kümesini oluşturmanın daha tabi ve kolay yolu genç ve genç olmayan keskin ayrımını biraz gevşetmek olacaktır. Bunu EVET, O genç insanlar kümesinin bir elemanıdır veya HAYIR, O genç insanlar kümesinin bir elemanı değildir gibi keskin cümleler yerine daha esnek olan “O kısmen genç insanlar kümesine aittir.” veya “O genç insanlar kümesine üye sayılamaz.” gibi cümleler kullanmak suretiyle gerçekleştireceğiz [9].

İlk örneğin gösteriminde kümede bulunan elemanlar ordinat ekseninde 1 ile kümenin elemanı olmayanlar ise 0 ile belirtilmiştir. Bu gösterim gayet kolaydır. İkinci örneğimizde bahsettiğimiz ve keskinliği azaltmak için oluşturduğumuz cümlelerin grafik olarak izahı ise aşağıdaki gibidir. Bu grafikte de bir önceki grafikte olduğu gibi ordinat ekseninde 1 değerine karşılık gelen apsisteki eleman B kümesinin elemanı, 0 değerine karşılık gelen eleman ise B kümesinin elemanı değildir. Grafikteki diğer değerler ise B kümesine aşamalı üyelik durumunu göstermektedir.

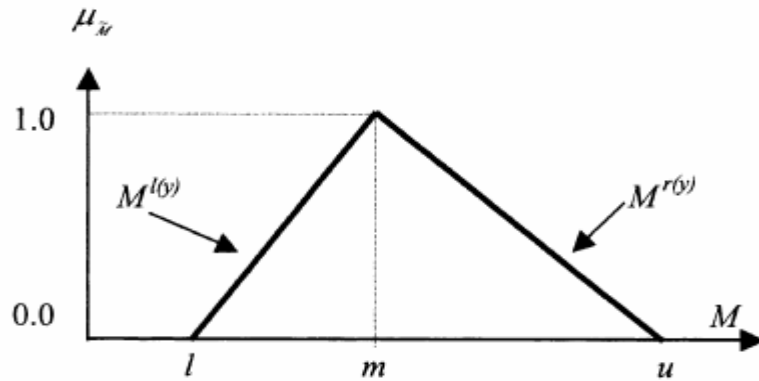


Şekil 2.2. Genç insanlar kümesine(B) değişik yaş gruplarının üyelik durumu [9]

Yukarıda da görüldüğü gibi keskin küme mantığındaki yapısal problem ortadan kalkmıştır çünkü 25 yaşındaki biri hala %50 genç olarak değerlendirilebilecektir.

2.3.3. Bulanık sayılar

İnsan düşüncesinin bilinmezliği ile ilgili Zadeh 1965 yılında ilk fuzzy set (bulanık küme) teorisini tanıttı. Bu teori belirsizlikten dolayı bilinmezliğin rasyonelliğine doğrudur. Bulanık küme teorisinin en önemli katkısı belirsiz veriyi ifade etme kabiliyetidir. Bu teori ayrıca matematiksel operatörlerin kullanımına ve programlamayı bulanık tanım kümesine uygulamaya izin verir. Bir bulanık küme, içerisindeki her üyenin derecesi 0 ile 1 arasında değişen aralıkta tayin edilen bir üyelik fonksiyonu tarafından tanımlanır [15].



Şekil 2.3. Bir üçgen bulanık sayı, $\tilde{M}$ [15]

Bir sembol bulanık kümeyi ifade ediyorsa üzerinde $\sim$ işareti yer alır. Bir üçgen bulanık sayı (A triangular fuzzy number (TFN)), Şekil 2.3’de $\tilde{M}$ ile gösterilmiştir. Bir üçgen bulanık sayı basitçe $(l|m,u)$ veya (l,m,u) şeklinde gösterilir. l,m ve u bir üçgensel sayıyı oluşturan parametrelerdir. Bunlar sırasıyla, en az olası değer, en olası değer ve en geniş olası değerdir.

Her bir üçgensel bulanık sayının sağa ve sola doğru lineer doğrusal bir gösterimi vardır. Öyle ki üyelik fonksiyonları şu şekilde tanımlanır.

$$\mu(x | \tilde{M}) = \begin{cases} 0, & x < l, \\ (x-l)/(m-l), & l \leq x \leq m, \\ (u-x)/(u-m), & m \leq x \leq u, \\ 0, & x > u. \end{cases} \quad (2.1)$$

Bir bulanık sayı üyenin her sağ ve sol gösterimine karşılık gelen şekliyle her zaman verilebilir.

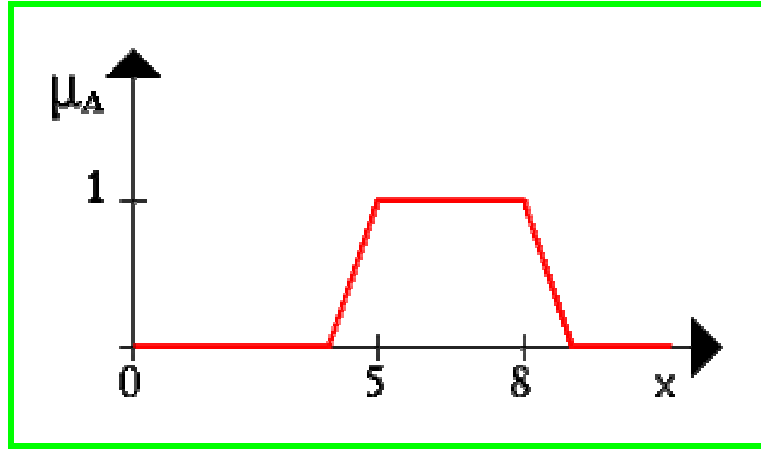
$$\begin{aligned} \tilde{M} &= (M^{l(y)}, M^{r(y)}) \\ &= (l + (m-l)y, u + (m-u)y), \quad y \in [0,1], \end{aligned} \quad (2.2)$$

$l(y)$ ve $r(y)$ bir bulanık sayının sol yan ve sağ yan gösterimidir. Literatürde bulanık sayıların sıralanması için birçok yöntem geliştirilmiştir. Bu metotlar farklı sonuçlar verebilir ve çoğu metot karmaşık matematiksel işlemler gerektiren yorumlama işlemlerinde can sıkıcıdır [16, 17].

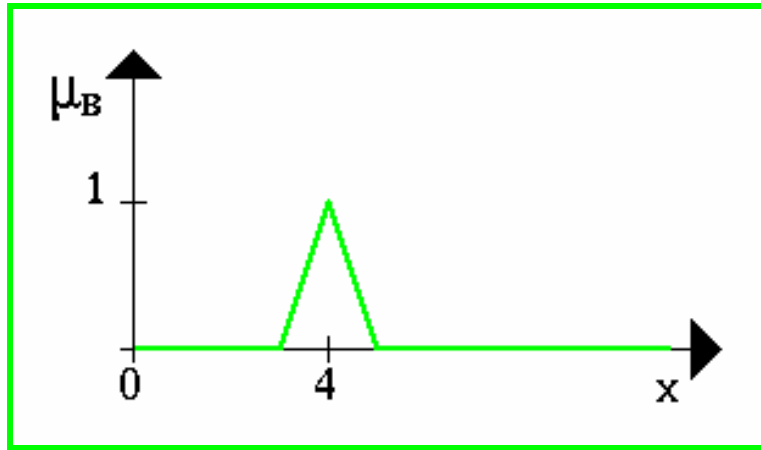
2.3.4. Bulanık kümelerdeki işlemler

Bulanık kümelerde de keskin küme işlemlerinde olduğu gibi kesişme, birleşme ve ters eleman özellikleri uygulanabilir. Bulanık mantıkla ilgili ilk makalesinde L. A. Zadeh iki bulanık kümenin kesişimi için minimum operatörünü birleşimi için de maksimum operatörünü önermişti. Bu yaklaşımın üyelik değerlerinin 0 ya da 1 olması durumunda keskin kümelerdeki kesişim ve birleşim işlemleriyle birebir örtüştüğü kolayca görülebilir [15].

Örnek olarak A 5 ile 8 arasında bulanık bir aralık olsun. B ise değeri 4 civarında olan bulanık bir değer olsun. Bu duruma uyan grafikler aşağıda verilmiştir [9].

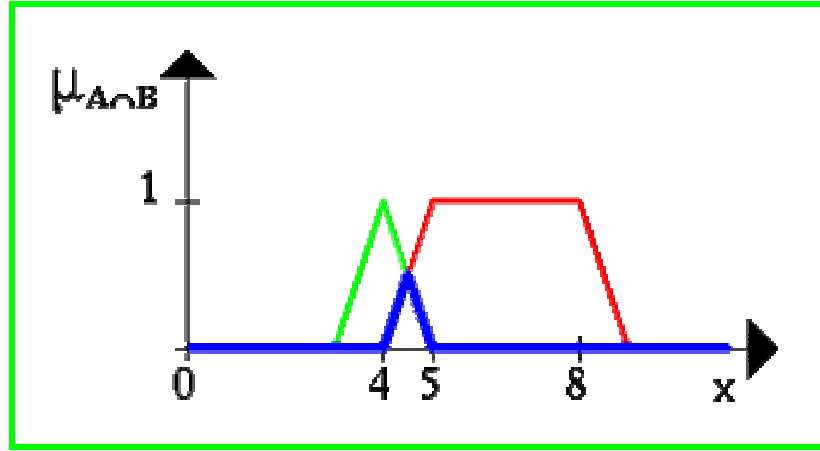


Şekil 2.4. 5 ile 8 arasında bulanık bir değer [9]



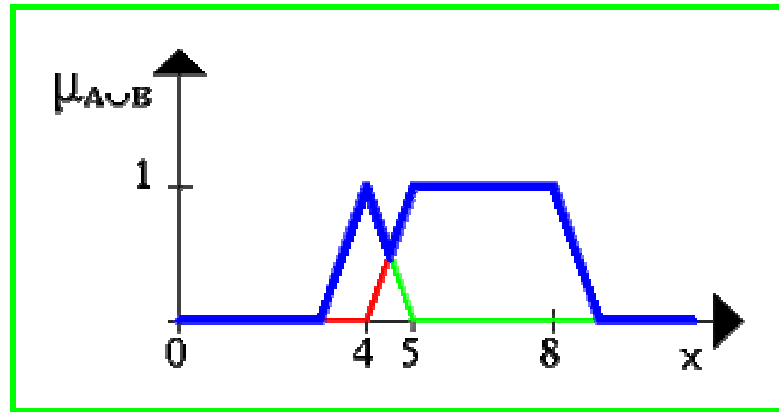
Şekil 2.5. 4 civarında bulanık bir değer [9]

Aşağıdaki grafik değeri 5 ile 8 arasında VE 4 civarında olan bulanık kümeyi göstermektedir. Burada VE işlemi iki kümenin kesişimi işlemine karşılık gelmektedir. Mavi çizgi iki kümenin kesişmesinin sonucunu göstermektedir.



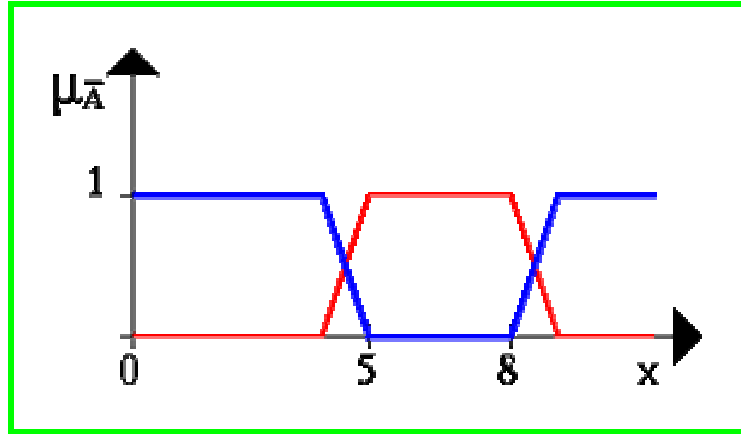
Şekil 2.6. A ve B bulanık kümelerinin kesişimi [9]

Yine aşağıdaki grafik değeri 5 ile 8 arasında VEYA 4 civarında olan bulanık kümeyi göstermektedir. Burada VEYA işlemi iki kümenin birleşimi işlemine karşılık gelmektedir. Mavi çizgi iki bulanık kümenin birleşmesinin sonucunu ifade etmektedir.



Şekil 2.7. A ve B bulanık kümelerinin birleşimi [9]

Aşağıdaki grafik bulanık kümelerin ters elemanının gösterimine bir örnektir. Mavi çizgi A bulanık kümesinin tersini göstermektedir.



Şekil 2.8. Bulanık bir kümenin tersi [9]

2.3.5. Bulanık kontrol sistemlerinin temel yapısı

Klasik kontrol sistem tasarımındaki ilk adım kontrol edilecek düzeneğin transfer fonksiyonunun tam olarak elde edilmesidir. Başka bir deyişle matematiksel modelinin oluşturulmasıdır. Ayrıca uygun ve etkin bir kontrol sağlamak için sistem parametrelerinin zamanla değişmemesi istenir. Fakat uygulamadaki sistemlerin pek çoğu bilinmeyen parametrelere veya karmaşık ve lineer olmayan karakteristiklere sahiptirler. Gerçi sistem parametrelerinin değiştiği veya sistemde lineerlikten sapma olduğu durumlar için adaptif kontrol yöntemleri gerçekleştirilmiştir; fakat bu tip kontrol sistemleri genellikle karmaşık olmaları ve hesaplamalarda uzun zaman almaları sebebiyle gerçek zaman uygulamalarında sorunlar çıkarmaktadır [9].

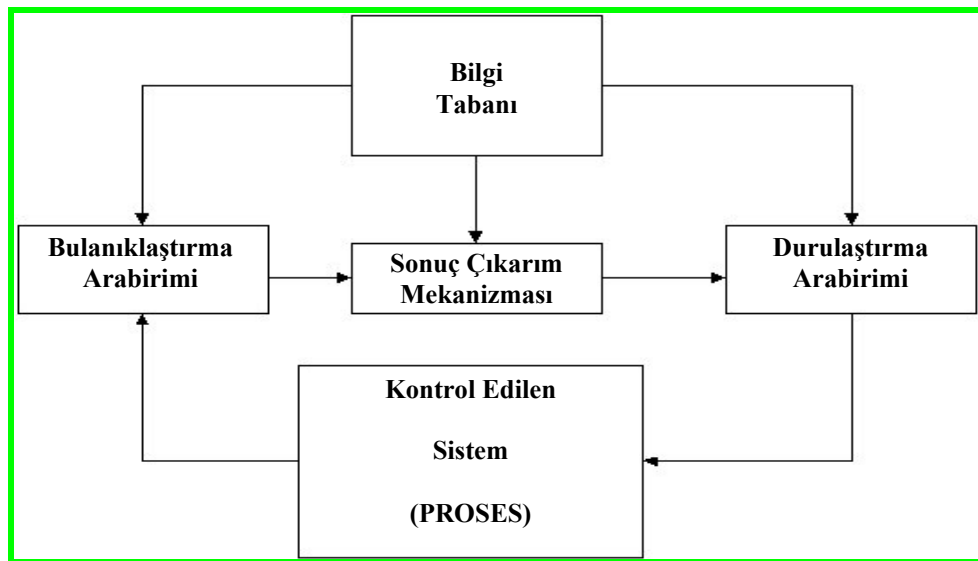
Böyle uygulamalarda, uygulama alanındaki uzman kişilerin bilgisi klasik kontrol sistemlerinden daha faydalı ve etkili olabilmektedir. Bu yüzden insan düşünme yeteneğini ve bilgisini kontrol sisteminin içine sokabilecek bir kontrol yöntemi olan Bulanık Mantık Denetleyicisi (Fuzzy Logic Controller, FLC) iyi bir çözüm olabilmektedir. Bulanık mantığa dayanan FLC sistemleri geleneksel ve matematiksel sistemlerin aksine insan düşüncenin ruhuna, ifade etme şekline yani diline daha yakındır. Bu sayede FLC sistemleri bir uzmanın kişisel ve mantıksal bilgi ve becerilerini nümerik hesaplamalar kullanarak otomatik bir kontrol stratejisine çevirebilir [9].

Bu uzman bilgisi genellikle kurallar şeklinde ifade edilmekte ve kuralların değerlendirilmesi için ihtiyaç duyulan oldukça yoğun hesaplamalar bir sayısal sinyal işlemcisi veya şuan mevcut olan birçok özel amaçlı bulanık mantık yongalarından birini kullanarak hızlandırılabilir [18].

Bir FLC tasarımı, bilinmesi gereken temel faktörler şunlardır ;

1. Gerçek giriş ve çıkışlar ve bunların evrensel kümeleri, yani her bir değişkenin alması muhtemel değerler aralığı.
2. Giriş ve çıkış değişkenlerinin ölçekleme faktörleri.
3. Her bir giriş ve çıkış değişkenleri için bulanık değerlerin kurulmasında kullanılacak bulanık üyelik fonksiyonları.
4. Bulanık kontrol kuralları tabanı [18].

Temel olarak FLC sistemleri Şekil 2.9'da da görüldüğü gibi dört temel arabirimden oluşur ;



Şekil 2.9. Bulanık mantık denetleyicinin (FLC) temel yapısı [9]

- 1) Bulanıklaştırma arabirimi,
- 2) Bilgi tabanı,
- 3) Sonuç çıkarım mekanizması,
- 4) Ve durulaştırma arabirimi.

Şimdi bu temel birimleri sırasıyla inceleyelim.

Bulanıklaştırma arabirimi

FLC sistemleri kesin olmayan dilsel ifadelerle gösterilen uygulama alanına ait bilgileri kullanmaktadır. Başka bir açıdan FLC'nin dayandığı bulanık küme teorisinde işlemler bulanık değerler üzerinde yapılmaktadır. Bu sebeple dış ortamdan ölçülen reel giriş değişkenlerini bulanık değişkenlere çeviren bulanıklaştırma FLC'de önemli bir rol oynar ve ilk adımı teşkil eder.

Bu arabirimde gerçekleştirilen temel işlemler şunlardır: Kontrolü yapılan sistemden giriş değişkenlerinin gerçek zamanda ölçümlerinin alınması, eğer gerekliyse giriş değişkenlerini sabit bir sayıyla çarpmak ya da bölmek gibi işlemlerle ölçeklendirme yapılması, yine gerekliyse ve isteğe bağlı olarak seçilen dilsel uzayın kesikli hale dönüştürülmesi, ölçümü yapılan reel giriş değişkenlerinin her birisini o değişkene ait söylem uzayına göre dilsel değişkenlere dönüştürmek ve bunlara ait üyelik değerlerini bulmak sayılabilir.

Sonuç olarak dilsel niteleyicisi ve üyelik değeri elde edilen her bir reel giriş değişkeni bulanık değişkenlere veya başka bir deyişle bulanık sayılara dönüştürülmüştür.

Her bir değişken için söylem uzayı değişkenin alabileceği maksimum ve minimum değerler arasını kapsar. Sonlu sınırlara sahip bu söylem uzayları sonlu sayıda dilsel değişkenlere ayrılır. Bu bölme işleminde kullanılan fonksiyonlara ise dilsel değişkenler kümesinin üyelik fonksiyonu denir. En çok kullanılan üyelik fonksiyonları üçgen, yamuk ve üstel fonksiyonlardır.

Bulanıklaştırma arabirimini oluşturmada üyelik fonksiyonunu belirlemek ve söylem uzayını bölgelere ayırmak için matematiksel bir metot yoktur ve sayısız şekillerde gerçekleştirilebilir. Bu ise kontrol sistemine esneklik vermesiyle beraber tasarım işlemlerini biraz daha zorlaştırmaktadır. Ve bu arabirim, kontrol sistemini büyük ölçüde etkilediği için dikkatli bir şekilde hazırlanması gerekir.

Üyelik fonksiyonlarının birbirleriyle kesişmesi ve iki farklı dilsel değişkene ait üyelik değerlerinin toplamının 1'den büyük olması bulanık kümelerde sıkça görülür. Uygulamaların genelinde görülen yapı ise ikiden fazla üyelik fonksiyonunun kesişmemesi ve üyelik fonksiyonlarının üçgen veya yamuktan oluşmasıdır. Üstel ve polinom ifadelerin hesaplanması ve sınırlandırılması ise pratikte daha zor ve zaman alıcıdır. Ayrıca bulanık üyelik fonksiyonlarının ağırlık merkezine göre simetrik olması işlemleri kolaylaştırdığı halde böyle olmadığı uygulamalarda görülebilmektedir [12].

Bilgi tabanı

Temel olarak uygulama sahasına ait uzman bilgisini ve kontrol hedeflerini içeren bilgi tabanı FLC'nin dilsel kavramları kullanabilmesini sağlayan arabirimdir. Bulanık kontrol sistem tasarımı klasik kontrol sistemlerinde olduğu gibi analitik bir metot olmaması bu arabirimin oluşturulmasını güçleştirmektedir. Veri tabanı ve kural tabanı olmak üzere iki kısımdan oluşur:

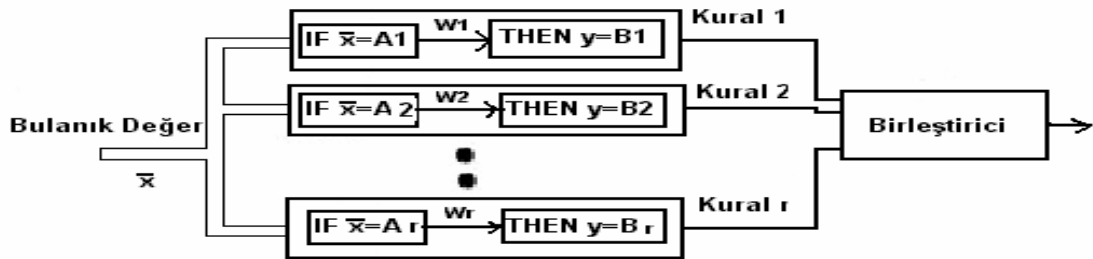
Veri tabanı

Kısaca dilsel kavramları (bulanık değişkenleri) FLC sisteminde kullanabilmek için gerekli tanımları içerir. Başlıca giriş, çıkış değişkenlerine ait dilsel uzayların tanımlarını, üyelik fonksiyonlarını, eğer sistemde değişkenler normalize edilmişse ve kesikli hale dönüştürülmüşse bu işlemlerle ilgili bilgileri ve FLC'de kullanılan bulanık işlemlerinin tanımlarını kapsar [6].

Kural tabanı

Uygulama alanındaki uzman kişilerin kontrol hedeflerini ve kontrol sırasında takip ettikleri yöntemleri karakterize etmeyi sağlayan dilsel değişkenlerden oluşturulan kurallar kümesidir. Oluşturulan bu kurallara bulanık kurallar denir [19]. Dilsel kural tabanı, sistemde dilsel-bulanık giriş değişkenlerinden hangilerinin çıkışa verileceğinin ifade edildiği bölümdür. Genellikle kurallar şart cümlelerinden (EĞER $X=A$ İSE $Y=B$) oluşur. Kuralların sayısı ve doğruluğu sistemin performansını etkileyen en önemli faktörlerdir [19]. Sistemin kural tabanını oluşturmak için bugüne kadar değişik yollar kullanılmıştır.

- Bunlardan birisi uygulanacak sistemi iyi tanıyan bir uzman bilgisi ile sistem giriş-çıkış değişkenleri ve kontrol kuralları belirlenir. Kurallar giriş çıkış değişkenlerinin dilsel ifadelerinden oluştuğundan bu işlem uzmanın kendi kontrol stratejisinin kural tabanına aktarımının en kolay ve güvenli yoludur [8].
- Diğer bir yol ise daha önceden operatörün kontrol yönteminin taklit edilmesidir. Bazı endüstriyel sistemlerde modellenemeyen ve ancak bir operatör yardımıyla kontrol edilebilen süreçlerin; operatörün izlenmesiyle, yaptığı işlemlerin (bilinçli yada bilinçsiz) şart cümlelerine (EĞER .. İSE) dönüştürerek kural tabanının elde edilmesidir [12].



Şekil 2.10. Bulanık kural tabanlı çıkarım sistem yapısı [9]

- Kural tabanıyla ilgili üçüncü yol ise sistemin bulanık modellenmesidir. Burada bulanık durum değişkenleri ve bulanık eliminasyon ile optimum bir bulanık denetleyici temel hedeftir.

- Sistemin giriş çıkış fonksiyonu veya verileri belirli ise öğrenmeye dayalı bir sistem, kural tabanı oluşturmak için kullanılabilir. Öğrenme yöntemi, sistem kurulmadan önce uygulanarak sabit bir kural tabanı oluşturulur. Daha sonra sistem performansının ölçülmesiyle kurallarda değişiklik yapılabilir. Bu yöntemin en önemli ve gerekli kısmı sisteme ait istenen (optimum) kontrol sürecinin giriş-çıkış verilerinin elimizde bulunmasıdır. Kural tabanı oluşturulduktan sonra gerekiyorsa her kurala belirli bir ağırlık vermek veya gereksiz (kullanılmayan) kuralların atılması, işlemleri hızlandırmak için gerekli olabilmektedir.

Kontrol kurallarının oluşturulması, sistemin bulanık durumları, bulanık tanımlama gibi çok geniş alanları kapsayan ve belirli bir sistematik yaklaşımın bulunmadığı bulanık modelleme üzerinde çalışmalar devam etmektedir [12].

Sonuç çıkarım mekanizması

FLC'nin en önemli arabirimidir. Bilgi tabanında tanımlanan bulanık kavram ve kuralları kullanarak giriş değişkenlerinin durumuna göre uzman bir kişiden beklenen kontrol hareketlerini dilsel ifade olarak üretir. Çıkışta dilsel değişkeni ve üyelik fonksiyonu belli bulanık sayılar elde edilir. Çıkış değişkenlerine ait üyelik fonksiyonu bulunurken çeşitli yöntemler kullanılmaktadır. Literatürde en çok geçen yöntemler minimum korelasyonu ve çarpım korelasyonu olarak sayılabilir. Bu yöntemler kullanılırken ilk önce, dilsel çıkış değişkenlerinin üretilmesine sebep olan giriş değişkenlerine ait üyelik dereceleri isteğe bağlı olarak minimum(VE) veya maksimum(VEYA) işlemine tabi tutulur ve çıkış üyelik fonksiyonlarının etkinlik derecesi (ω_i) diyebileceğimiz tek bir üyelik derecesi elde edilir. Genelde etkinlik derecesini elde etmede literatürde minimum işlemi kullanılmıştır. Bu işlem üretilen her bir çıkış için tekrarlanır ve elde edilen bu etkinlik dereceleri bilgi tabanı konusunda bahsedilen metotlardan birinde kullanılarak çıkış dilsel değişkenleri ve

üyelik fonksiyonları oluşturulur. Bu işlemler sonucu tek bir bulanık sayının elde edilmesi gerekmez ve genelde de birden fazladır. Bu FLC'nin en belirgin özelliğidir [9].

Durulaştırma arabirimi

Bulanık çıkarımın sonucu bulanık bir kümedir. Bu sonucun tekrar sisteme uygulanması için giriş değeri gibi sayısal değere dönüştürülmesi gerekir. Bu işlem durulama olarak adlandırılır. Durulama birimi karar verme biriminden gelen bulanık bir bilgiden bulanık olmayan ve uygulamada kullanılacak gerçek değerlerin elde edilmesini sağlar.

Durulama işleminde değişik yöntemler kullanılmaktadır. Önce her kural için üyelik değerlerinden oluşan değer ve sonuç kural tespit edilir. Daha sonra en uygun yöntem seçilerek durulama yapılır. En çok kullanılan yöntemler maksimum üyelik yöntemi, ağırlık merkezi yöntemi, ağırlık ortalaması yöntemi, mean-max üyelik yöntemi [20].

2.3.6. Bulanık mantık uygulama örnekleri

Bulanık mantık sistemlerinin ilk uygulama alanları çimento sanayi ve su arıtma sistemleri olmuştur. Daha sonra buhar türbini, nükleer reaktör, asansör ve vinç denetimi gibi değişik alanlarda da bulanık mantıktan yararlanılmıştır.

Özellikle Japonya'da çok geniş uygulama alanları bulmuştur. 1988 yılında Japonya'da Hitachi firmasının Sendai'deki metro için gerçekleştirdiği bulanık mantık sistemi çok başarılı olmuş ve bu olaydan sonra Japonya'da Bulanık Mantık Sistemleri çok rağbet görmüştür.

1990'da bulanık mantık fotoğraf makinelerinden ev aletlerine ve hatta borsaya kadar çok değişik alanlarda kullanılmıştır. Günümüzde, bulanık mantık uygulamalarına yönelik yazılım ve donanımlar piyasada hazır şekilde kullanıcılara sunulmuştur. Hatta bulanık mikroişlemciler dahi satışa sunulmuştur. Panasonic firmasının

tasarladığı video kayıt cihazının elle tutulması durumunda çekim arasındaki sarsıntıların ortadan kaldırılması, Subaru ve Nissan firmalarının birlikte gerçekleştirdikleri otomobil vites sisteminde araba kullanılış stiline ve motor yüküne uygun dişli oranının seçimi, yine Nissan tarafından gerçekleştirilen ABS fren sistemleri, Fujitsu, Toshiba, Hitachi ve Mitsubishi'nin büyük asansör denetim sistemleri, Matsushita firmasının dizayn ettiği çamaşır makinesinde çamaşırların kirliliğine, ağırlığına ve kumaşın cinsine göre yıkama programı seçme bulanık mantığının kullanıldığı ilginç örneklerdir.

Ayrıca Tokyo Institute of Technology'den M.Sugeno tarafından bir model helikopteri ses komutlarıyla hareket ettiren ve hatta pilotlar için bile güç olan aracı havada sabit tutmayı başaran karmaşık bir bulanık kontrol sistemi tasarlanmıştır. Bunun yanı sıra son zamanlarda lineer olmayan, cevap gecikmeli ve parametreleri zamanla değişen biyolojik süreçlerin bulanık kontrolü ve modellenmesi yapılan başarılı çalışmalar arasındadır [21, 14].

Aşağıda bulanık mantık uygulamalarının kullanılmaya başlandığı alanlar belirtilmiştir. Burada çeşitli alanlardaki örnekler kısa bir liste halinde verilmiştir.

- *Biyoloji ve Tıp Biliminde:* Bulanık mantık tabanlı teşhis sistemleri, kanser araştırmaları, bulanık mantık tabanlı protez cihazlarının işlenmesi, bulanık mantık tabanlı hareket düzensizliklerinin analizinde.
- *Yönetim ve Karar Desteklerinde:* Bulanık mantık tabanlı fabrika yeri seçimi, bulanık mantık yardımcı askeri karar yapımlarında, bulanık mantık tabanlı market satış stratejilerinin belirlenmesi.
- *Ekonomi ve Finans:* Kompleks satış sistemlerinin bulanık modellenmesinde, bulanık mantık tabanlı ticaret sistemlerinde, bulanık mantık tabanlı maliyet – fayda analizlerinde, bulanık mantık tabanlı yatırım değerlendirmesinde.

- *Çevre Bilimi:* Bulanık mantık tabanlı hava tahmininde, bulanık mantık tabanlı su kalite kontrolünde.
- *Mühendislik ve Bilgisayar Bilimlerinde:* Bulanık veri tabanı sistemlerinde, bulanık mantık tabanlı deprem tahminlerinde, bulanık mantık tabanlı nükleer işletmelerin otomasyonunun kontrolünde, bulanık mantık tabanlı bilgisayar ağları dizaynında, bulanık mantık tabanlı mimari dizaynların değerlendirilmesinde, bulanık mantık kontrol sistemlerinde.
- *Araştırma Çalışmalarında:* Bulanık mantık tabanlı planlama ve modelleme, bulanık mantık tabanlı kaynak ayrışımında.
- *Kalıp Tanıma ve Sınıflandırılması:* Bulanık mantık tabanlı konuşmaların tanınmasında, bulanık mantık tabanlı el yazısı tanımlarında, bulanık mantık tabanlı yüz karakterlerinin tanınmasında, bulanık mantık tabanlı askeri komut analizlerinde, bulanık resim araştırmasında.
- *Psikoloji:* Bulanık mantık tabanlı insan davranışları analizinde, bulanık mantık tabanlı suç işleme ve önleme sebeplerinin araştırılmasında.
- *Güvenilirlik ve Kalite Kontrolü:* Bulanık mantık tabanlı hata tahmininde, üretim hattının izlenmesi ve denetimi [9].

2.4. Öğrenci Seçim Yöntemleri

2.4.1. Analitik hiyerarşi süreci (AHS)

Analitik Hiyerarşi Süreci (AHS), 1977 yılında Thomas L. Saaty tarafından geliştirilen çok ölçütlü karar verme tekniklerinden biridir. AHS karar almada, grup veya bireyin önceliklerini de dikkate alan, nitel ve nicel değişkenleri bir arada değerlendiren matematiksel bir yöntemdir [22]. AHS, sayısallaştırılabilen somut

veya soyut kriterleri karşılaştırarak ölçen ve kriterlerin birbirlerine göre önceliklerini hesaplayarak önem sıralarını belirleyen bir yaklaşımdır [23]. Ayrıca AHS gruplara ve bireylere, karar verme sürecindeki nitel ve nicel faktörleri birleştirme olanağı veren güçlü ve kolay anlaşılır bir yöntem bilimdir [4,24]. Karar verme sürecinde karar vericilerin/uzmanların deneyim ve bilgilerine önem verilmesi gerektiğini savunur. AHS, gerçek hayatta verilmesi gereken karmaşık ve çok amaçlı kararları etkileyecek kriterler kümesini ve bu kriterlerin verilecek karardaki göreceli önemlerini uzmanların değerlendirmelerine dayanarak belirler. Böylece sistematik bir yaklaşımla sayısal performans ölçümleri, subjektif değerlendirmeler ile birleştirilerek sağlıklı sonuçlar elde edilir [23].

AHS her sorun için amaç, kriter, olası alt kriter seviyeleri ve seçeneklerden oluşan hiyerarşik bir model kullanır [25]. Karışık, anlaşılması güç veya yapılaşmamış sorunlar için genel bir yöntemdir. Karar verilen problem için AHS yönteminin uygulanması 3 aşamada gerçekleştirilir [24].

1. *Aşama:* Karar verilen problemin hiyerarşik yapısının oluşturulması (ayrıştırma)
2. *Aşama:* İkili karşılaştırma
3. *Aşama:* Ağırlık ve tutarlılık hesaplamaları

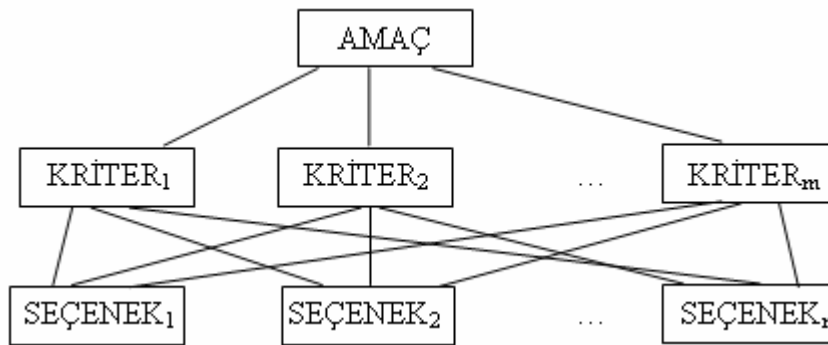
Hiyerarşinin tüm parçaları birbirleri ile ilgilidir ve bir faktördeki değişimin diğer faktörleri nasıl etkilediği kolayca görülebilir. AHS'nin hiyerarşik yapısındaki bu esneklik ve etkinlik karar vericiye karar sürecinde çok yardımcı olur. Kararları bu yapıda kurarak; birçok veri türü biraraya getirilebilir, performans seviyelerindeki farklılıklar birbirine uygun hale getirilebilir ve farklı gözüken nesneler arasında karşılaştırma yapılabilir [25].

Bilişsel psikoloji alanında yapılan deneysel çalışmalar insanların bilişsel yeteneklerinin yüksek miktarda bilgi karşısında zayıf düştüğünü göstermiştir. Bilişsel olarak aşırı yüklenen kişiler sorunun tamamı ile uğraşmak yerine sezgisel yöntemlerle sorunu küçük parçalara ayırıp büyük olasılıkla baskın olmayan çözümler bulmaktadırlar [23]. Bu konuda Miller "Sihirli yedi artı eksi iki rakamı: Bilgi işleme

kapasitemiz üzerindeki sınırlar" isimli ünlü makalesinde aynı anda uğraşılabilir, beyin tarafından farkı gözetilebilecek ve kısa dönem hafızada işlenebilecek öğe sayısının üst sınırının 7 olduğunu bunun bazı kişilerde 5'e düşerken en fazla 9'a çıkabileceğini belirtmiştir. Bu yüzden insanlar karmaşık sorunlarla karşılaştıklarında söz konusu sorunu daha iyi anlayabilmek için sorunu bileşenlerine ayırmalı ve bu bileşenleri hiyerarşik bir şekilde düzenlemelidirler. Diğer bir deyişle karar verme sorunun olabildiğince ayrıntılı olarak ortaya konması ve daha sonra hiyerarşi olarak adlandırılan ve her biri bir dizi öğeden oluşan katmanlar halinde incelenmesi gerekir [26].

Örneğin yerleşmek için bir şehir seçmek istenildiğinde seçenekler belli kriterlere göre karşılaştırılır. Bu kriterler şehrin kişinin yakınlarının yaşadığı şehre olan uzaklığı, yaşam maliyeti, iklimi, eğitim olanakları, yaşam kalitesi olabilir. Söz konusu kriterler, örneğin yaşam kalitesi; ulaşım, kültür ve eğlence etkinlikleri gibi alt kriterlere ayrılabilir ve sorun çözümü hiyerarşik bir yapıya kavuşturulabilir [27].

AHS modelinde hiyerarşinin en üstünde bir amaç; bu amacın altında sırasıyla kriterler, alt-kriterler ve seçenekler vardır. Şekil 2.11'de Saaty (1980)'den yararlanılarak oluşturulan hiyerarşik yapı modeli görülmektedir [28].



Şekil 2.11. Basit bir AHS modeli [28]

Hiyerarşi kullanımı karışık sistemlerle ilgilenmek için etkin bir yoldur. Hem sistem organizasyonuna olanak verdiğinden yapısal olarak hem de sistem içi bilgi kontrolü ve iletişimine olanak verdiğinden fonksiyonel olarak etkindir [29].

Çok amaçlı karar verilirken en temel sorun, değerlendirilen seçenekler için bir çok kriter göz önünde bulundurularak ağırlık, önem veya üstünlük belirlemektir. AHS; bir hiyerarşideki bu tür tercihlerin belirlenmesi yöntemidir [30].

Sorun, hiyerarşik bir modele oturtulduktan sonra hiyerarşiyi oluşturan öğelerin görelî üstünlükleri hesaplanır. Karar verici, bir düzeydeki öğelerin, hiyerarşide hemen bir üst düzeyde yer alan öğeler açısından görelî önemlerini saptayacak şekilde Çizelge 2.1’de görülen değer ve tanımlara dayalı bir puanlama yapar ve ikili karşılaştırmalar matrisi oluşturulur.

Görelî önemlerin belirlenmesi için gerekli matematiksel hesaplar aslında ikili karşılaştırmalar matrislerinin en büyük özvektörünün bulunmasından ibarettir. Herhangi bir düzeydeki ikili karşılaştırma matrisinin özvektörünü hesaplamak için, her sütundaki elemanları normalize edip oluşan normalize matrisin her satırındaki elemanların ortalaması bulunur.

Eğer ikili karşılaştırmalar matrisi yerine seçeneklerin ölçüte göre nicel performans değerleri kullanılacaksa, özvektörü hesaplamak için söz konusu performans değerlerinden oluşan vektörü normalize etmek yeterlidir.

Sonuçta, bir seçeneğin bir üst düzey öğeye göre görelî önemi, söz konusu üst düzey öğenin bir üst düzey açısından görelî önemi ... vs.’nin çarpılması ve bu işlemin en üst düzey olan amaç düzeyine kadar sürdürülmesi sonucu; hiyerarşinin en alt düzeyinde yer alan seçeneklerin toplam görelî üstünlükleri, amacı gerçekleştirme açısından, bulunabilir. Toplam görelî üstünlüklere göre seçenekler en iyiden en kötüye sıralanarak bir tam ön sıralama elde edilir [31].

Çizelge 2.1. AHS değerlendirme ölçeği [32]

Sayısal değer	Tanım
1	Öğeler eşit önemde veya aralarında kayıtsız kalınıyor
3	1. öğe 2.'ye göre biraz daha önemli veya biraz daha tercih ediliyor
5	1. öğe 2.'ye göre fazla önemli veya fazla tercih ediliyor
7	1. öğe 2.'ye göre çok fazla önemli veya çok fazla tercih ediliyor
9	1. öğe 2.'ye göre aşırı derecede önemli veya aşırı derecede tercih ediliyor
2.4.6.8	Ara değerler

AHS yönteminde; Adayların bir kritere göre ikili karşılaştırmaları söz konusudur ve ikili karşılaştırmaları sonucu adaylar arasında o kritere göre performans değerlerini belirten bir önem derecesi elde edilir. Aynı şekilde kriterler arasında da ana amaca yönelik yapılan ikili karşılaştırmalar sonucu bir önem derecesi elde edilir. AHS'nin, problem çözümünde hiyerarşik olarak ayrıştırılabilir bir yapısının olması ve kullandığı değer fonksiyonlarının toplamsal olarak birleştirilebilmesi ile Çok Ölçütlü Değer Teorisine dayanan yöntemlere benzediği ileri sürülür. Ancak AHS yönteminde karar vericinin tercihini belirlemede tüm adayları birbiriyle karşılaştırması ve bunun sonucunda adayların birbirlerine üstünlüklerinin kesin sayılar yerine oranlarla tanımlanması ve tutarsızlığa belirli bir oranda izin verilmesi gibi farklar olduğu söylenebilir [30]. AHS, düşünce ve yargıda tutarlılığı göz önünde bulundurmaya gerektirir fakat tercihler arasında tutarlılık bir ölçüde ihlal edilebilir:

- Öğelerin ikili karşılaştırmaları sırasında geçişgenlik olmayabilir. Örneğin herhangi bir kritere göre, karar verici, a_i seçeneğini a_j seçeneğine ve a_j seçeneğini ise a_k seçeneğine tercih ederken a_k 'yi de a_i 'ye tercih edebilir [33].
- Tercihlerin yoğunluklarına ilişkin sayısal bir tutarsızlık olabilir. Örneğin a_i , a_j 'ye üç kez daha fazla ve a_j , a_k 'ye iki kez daha fazla tercih ediliyor iken a_i , a_k 'ye göre altı kez daha fazla tercih edilmeyebilir.

Aslında bu şekilde karşılaştırmalara dayalı bir değerlendirme sırasında mükemmel bir tutarlılığa erişmek hemen hemen imkansızdır. Bir karar modelinin etkinliği

irdelenirken modelin kullanımı sonucunda verilen kararın tutarsızlığının ilgili sorun açısından ne denli kötü olduğu araştırılmalıdır. AHS, incelenen sorun için tutarlılık varsayımından sayısal olarak sapma derecesi ile ilgilenir. Sayısal tutarlılık için bu gibi durumlarda kurulan hiyerarşik modelin ikili karşılaştırmalar matrislerine ait tutarsızlık oranlarının %10'dan büyük olmaması gerekir [24].

Dikkat edilmesi gereken diğer bir özellik ise o düzeyle ilgili tüm öğelerin hiyerarşiye dahil edilmesi ve aynı düzeydeki öğelerin birbirinden bağımsız olmaları gerekliliğidir. Eğer öğeler arasında karşılıklı ilişkiler varsa birbirleriyle birleştirilmeli veya bir tanesi devre dışı bırakılmalıdır [24].

Son olarak; aynı düzeye ait bir öğenin diğer bir öğeye sonsuz düzeyde tercih edilemeyeceği söylenebilir. Tüm tercihler 1-9 ölçeğinde ifade edilmelidir.

Betimsel modelleme tarzına sahip ve oran ölçeği kullanan AHS anlaşılır ve kullanımı kolay bir yöntemdir. Yöntemin karşısında ve yanında olanların uzun tartışmalarına neden olmuştur [34, 24]. Yöntemi eleştirenler dezavantaj olarak seçeneklerden birinin değerlendirme dışı bırakılması ya da yeni bir seçeneğin değerlendirmeye alınması durumunda seçeneklerin sıralamasının değişebilmesini (fakat gerçek hayatta karar vericiler bu şekilde davranabilir), nominal ölçekte alınan bilginin oran ölçeği olarak kullanılmasını (örneğin ikili karşılaştırmalar matrisi oluşturulurken karar vericinin a 'yı b 'ye "fazla tercih etmesi" durumunda 5 kullanılıyor fakat sanki karar verici $w_a/w_b=5$ demiş gibi işlem yapılıyor) göstermişlerdir [35].

AHS ile ilgili çalışmalar

Barbarosoğlu ve Yazgaç (1997), Türk Elektrik Endüstrisi Şirketler'nde uyguladıkları AHS modellerinde tedarikçi seçimi problemini ele almışlardır [8].

Rosenbloom (1997), makalesinde, kompleks karar verme problemlerinin ele alınmasında AHS'nin popüler ve pratik bir araç olduğunu belirtmiştir. Metodun karar verme alternatifleri arasında derecelendirmeyi sağladığını ve girdi verilerinin ikili

karşılaştırmalarının rastsal değişkenlermiş gibi ele alınmasını önermiş ve bunun alternatifler arasındaki farkların istatistiksel olarak anlamlı olup olmadığına karar vermeyi sağlayacağını belirtmiştir [8].

Davies(1994), çok amaçlı karar verme modellerinde AHS kullanımı ile grup kararlarındaki komplekslik, görecelik ve grup uzlaşma azlığının üstesinden gelinebileceğini ifade etmiş ve grup kararlarını yönetmede AHS'yi kullanarak bir şirket için ajans seçimi üzerine uygulamıştır [8].

Badiru (1991), imalat teknolojilerinin ekonomik analizi için çok ölçütlü bir levha modeli geliştirmiştir. Bu modelde, AHS, Fayda Modeli ve Sistem Değeri Modeli için etkileşimli makro modeller bulunmaktadır. Söz konusu model, Lotus 1-2-3 makro dili ile geliştirilmiş ve aynı anda beş imalat teknolojisi alternatifi ile beş ölçüt ele alınabilmektedir [8].

Saaty (1990), Orta gelirli bir ailenin satın alınacak ev problemine AHS'yi uygulamıştır. Ailenin hedefi, kendileri için tatmin edici bir ev satın almaktır. Üç aday ev vardır ve ailenin bir evde aradıkları özellikleri sekiz ölçüt altında toplamıştır. En üst düzeyde “evin tatmin ediciliği” hedefi vardır. İkinci düzey, hedefe katkıda bulunan sekiz ölçüttür ki bunlar; evin büyüklüğü, odalarının sayısı, ulaşımı, kaç yıllık olduğu, bahçe alanı, modern olanakları, genel koşulları ve ekonomik koşulları olanaklarıdır. Hedefe göre ikinci düzeydeki ölçütlerle ilgili yargılar ailenin üyeleri tarafından verilmiştir ve bu yargılar sonucunda hedefe göre “ekonomik koşullar” aile için en önemli ölçüt olarak bulunmuştur [8].

Dyer (1990), makalesinde AHS'nin eleştirisini yaparken, AHS ile alternatifler için belirlenen sıranın yeni bir alternatifin ilavesi ile değişebileceğini belirtmiştir. Öte yandan BELTON (1986), çok ölçütlü karar verme modellerini karşılaştırmış ve AHS ile basit çok ölçütlü değer fonksiyonlarının karar analizinde en etkin modeller olduğu sonucuna varmıştır [8].

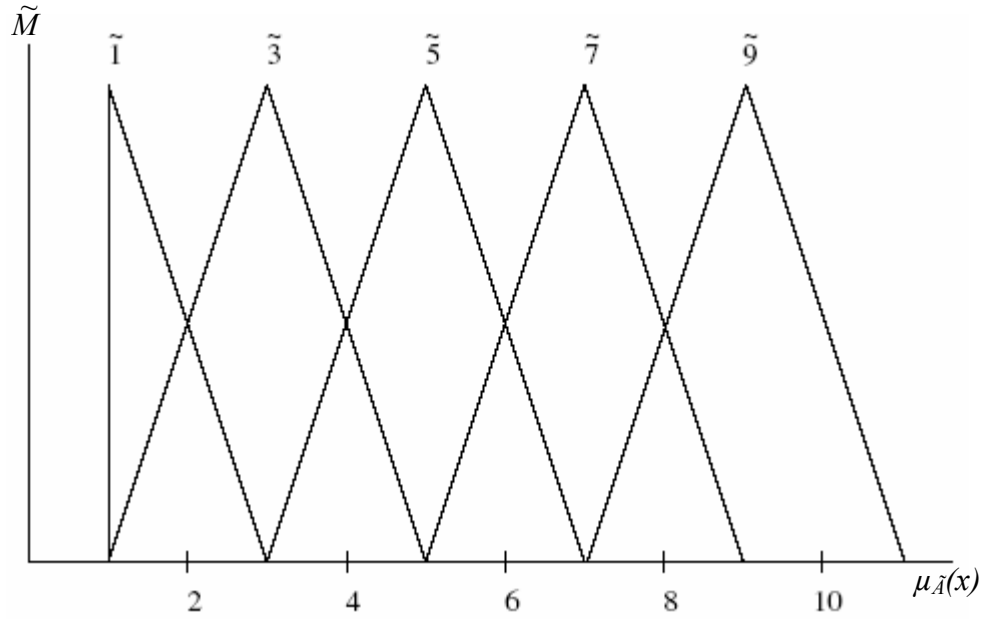
Saaty (1987), liseyi bitiren bir öğrencinin üniversite seçim problemine AHS'yi uygulamıştır. Öğrenci beş üniversiteye kabul edilmiştir ve bu üniversitelerden birini tercih etmek zorundadır. Hedefi “en iyi üniversite” yi seçmektir ve bir üniversitede aradığı özellikleri dört ölçüt altında toplanmıştır [8].

2.4.2. Bulanık AHS (Fuzzy Analytic Hierarchy Process - FAHP)

Saaty'nin AHS modelinde, bulanıklık direk olarak gösterilmez ancak probleme karar verirken ikili karşılaştırma kullanılan biçimlenmiş ters karar matrisleri tarafından bulanıklık modellenir [36]. Bununla beraber, kesin sayılar ikili karşılaştırmada yeterli bulanıklığı içermezler. Bulanık AHS bulanık mantık ve dilsel değişkenleri AHS süreci ile birlikte kullanmayı sağlayan bir karar destek tekniğidir [37, 38]. AHS karar matrislerinde ikili karşılaştırma gösteriminin üstün bir yönteminin kullanımıyla fen bilimleri enstitüleri için yüksek lisans düzeyinde öğrenci seçim probleminde öznelliğin ve belirsizliğin üstesinden gelinebilir.

Bulanık AHS için birçok algoritma mevcuttur. Aşağıda öncelikle Cheng ve Mon [13] aralık aritmetiği metodu ve daha sonrada Chang'ın bulanık boyut analiz metodu adımlarıyla anlatılmıştır [21].

Cheng ve Mon 1994 yılında bulanık AHS için üçgensel bulanık sayıları, bölümleri ve aralıklı aritmetiği kullanan basit bir genel algoritma önermişlerdir. Bu metot da Saaty (1987)'nin önerdiği sınıflandırma ölçeğini geliştirmek için bulanık sayılar kullanılır [39, 7].



Şekil 2.12. Üçgen bulanık sayılar [39]

Bu algoritmada karar matrislerini kurmak için $\tilde{1}$ ile $\tilde{9}$ arasında üçgensel bulanık sayılar kullanılır. Bu sayılar için üyelik fonksiyonları $\mu_{\tilde{A}}(x)$ üçlü olarak tanımlanır. (a_1, a_2, a_3) Çizelge 2.2'deki sayıların Şekil 2.12'de gösterimi gibi. Bulanık öz vektörlerin çözümü için aralık aritmetiği kullanılır. Algoritmanın adımları aşağıda özetlenmiştir.

Çizelge 2.2. Üçgen bulanık sayıların üyelik fonksiyonlarının üçlü tanımlanması

Bulanık sayı	Üyelik fonksiyonu $\mu_{\tilde{A}}(x)$
$\tilde{1}$	$(1, 1, 3)$
$\tilde{x}$	$(x-2, x, x+2)$, burada $x=3, 5, 7, 9$

ADIM 1 : Karar matrislerindeki kriterlerin göreceli zorlukları üçgensel bulanık sayılar $\tilde{a}_{ij}$ kullanılarak karşılaştırılır. Şöyle ki ;

$$\tilde{A} = \begin{bmatrix} 1 & \tilde{a}_{12} & \tilde{a}_{13} & \cdots & \tilde{a}_{1(n-1)} & \tilde{a}_{1n} \\ \frac{1}{\tilde{a}_{12}} & 1 & \tilde{a}_{23} & \cdots & \tilde{a}_{2(n-1)} & \tilde{a}_{2n} \\ \vdots & \vdots & \ddots & \cdots & \vdots & \vdots \\ \frac{1}{\tilde{a}_{1(n-1)}} & \frac{1}{\tilde{a}_{2(n-1)}} & \frac{1}{\tilde{a}_{3(n-1)}} & \cdots & 1 & \tilde{a}_{n(n-1)} \\ \frac{1}{\tilde{a}_{1n}} & \frac{1}{\tilde{a}_{2n}} & \frac{1}{\tilde{a}_{3n}} & \cdots & \frac{1}{\tilde{a}_{n(n-1)}} & 1 \end{bmatrix} \quad (2.3)$$

Burada, $\tilde{a}_{ij}$ şu şekilde tanımlanır.

$$\tilde{a}_{ij} \in \{\tilde{1}, \tilde{3}, \tilde{5}, \tilde{7}, \tilde{9}, \tilde{1}^{-1}, \tilde{3}^{-1}, \tilde{5}^{-1}, \tilde{7}^{-1}, \tilde{9}^{-1}\}$$

ADIM 2 : $\tilde{A}_{\tilde{x}} = \tilde{\lambda}_{\tilde{x}}$ (2.4) bulanık sayı çözümünde $\tilde{\lambda}_{\tilde{x}}$ bulanık öz değeri temsil eder, ayrıca burada $\tilde{x}$, 0'dan farklı olmak üzere bulanık sayılar içeren bulanık vektördür. Buna göre Eş. 2.4'deki formül aşağıdakine denktir.

$$(\tilde{a}_{i1} \otimes \tilde{x}_1) \oplus \dots \oplus (\tilde{a}_{in} \otimes \tilde{x}_n) = \tilde{\lambda} \otimes \tilde{x}_i; 1 \leq i \leq n \quad (2.5)$$

Burada $\otimes$ ve $\oplus$ sırasıyla bulanık çarpma ve bulanık toplama işlemlerini ifade eder [11, 23].

ADIM 3 : Aralıklı aritmetik ve bölümler kullanılarak bulanık çarpma ve toplama işlemleri uygulanır. Güven aralığı tanımıyla, üçgen bulanık sayılar alternatif olarak karakterize edilebilir [14].

Örneğin: Bulanık $\tilde{F}$ sayısının (a_1, a_2, a_3) biçiminde üçlü hale getirilmesi şöyledir.

$$\forall \alpha \in [0,1], \tilde{F}_{\alpha} = [a_1^{\alpha}, a_3^{\alpha}] = [a_1 + (a_2 - a_1)\alpha, a_3 - (a_3 - a_2)\alpha] \quad (2.6)$$

Burada $[a_1^{\alpha}, a_3^{\alpha}]$ üçgen bulanık sayılar için kesimlerdir.

Kaufmann ve Gupta (1991) yılında yaptıkları çalışmayla, $\tilde{P}=(1/a_3, 1/a_2, 1/a_3)$ değeri üçgen bulanık sayıların tersi fonksiyonun değerine yaklaşır sonucuna varmışlardır [40].

$\tilde{F}$ fonksiyonu $\tilde{F} = (a_1, a_2, a_3)$ 'dür ve tersi fonksiyon $\tilde{F}^{-1}$ ile sembolize edilir. Yani $\tilde{F}^{-1}$ fonksiyonu için α kesimleri aşağıdaki gibidir.

$$\tilde{F}_\alpha^{-1} = \left(\frac{1}{a_3 - (a_3 - a_2)\alpha}, \frac{1}{a_1 + (a_2 - a_1)\alpha} \right) \quad (2.7)$$

Eş. 2.3 ve Eş. 2.4 'deki işlemde $0 < \alpha \leq 1$ ve tüm i ve j değerleri için

$$\tilde{a}_{ij}^\alpha = [a_{ij1}^\alpha, a_{iju}^\alpha], \quad \tilde{x}_i^\alpha = [x_{i1}^\alpha, x_{iu}^\alpha], \quad \tilde{\lambda}^\alpha = [\tilde{\lambda}_1^\alpha, \tilde{\lambda}_u^\alpha] \quad (2.8)$$

tanımlanır.

Ve Eş. 2.8 'deki işlem Eş. 2.5'de yerine konularak

$$[a_{i11}^\alpha x_{11}^\alpha, a_{i1u}^\alpha x_{1u}^\alpha] \oplus \dots \oplus [a_{in1}^\alpha x_{n1}^\alpha, a_{inu}^\alpha x_{nu}^\alpha] = [\lambda_1^\alpha x_{i1}^\alpha, \lambda_u^\alpha x_{iu}^\alpha] \quad (2.9)$$

veya

$$a_{i11}^\alpha x_{11}^\alpha + \dots + a_{in1}^\alpha x_{n1}^\alpha = \lambda_1^\alpha x_{i1}^\alpha, \quad a_{i1u}^\alpha x_{1u}^\alpha + \dots + a_{inu}^\alpha x_{nu}^\alpha = \lambda_u^\alpha x_{iu}^\alpha \quad (2.10)$$

elde edilir.

ADIM 4 : Karar vericilerin tatmin olma dereceleri a_{ij}^α şöyle tanımlanır.

$$a_{ij}^\alpha = \lambda a_{iju}^\alpha + (1 - \lambda) a_{ij1}^\alpha \quad \forall \lambda \in [0,1] \quad (2.11)$$

Burada λ iyimserlik indeksidir. Yani karar matrisleri şu şekilde olur.

$$\tilde{A} = \begin{bmatrix} 1 & \tilde{a}_{12} = \lambda a_{12u}^\alpha + (1-\lambda)a_{12l}^\alpha & \cdots & \tilde{a}_{1n} = \lambda a_{1nu}^\alpha + (1-\lambda)a_{1nl}^\alpha \\ \frac{1}{\tilde{a}_{12} = \lambda a_{12u}^\alpha + (1-\lambda)a_{12l}^\alpha} & 1 & \cdots & \tilde{a}_{2n} = \lambda a_{2nu}^\alpha + (1-\lambda)a_{2nl}^\alpha \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{\tilde{a}_{1n} = \lambda a_{1nu}^\alpha + (1-\lambda)a_{1nl}^\alpha} & \frac{1}{\tilde{a}_{2n} = \lambda a_{2nu}^\alpha + (1-\lambda)a_{2nl}^\alpha} & \cdots & 1 \end{bmatrix} \quad (2.12)$$

$\alpha = 0.1, 0.2, \dots, 1$ 'e karşılık gelen öz vektör ve λ 'nın sabit değeri hesaplanır. Bu öz vektörler özellik ağırlıklarının toplamının sıralamasını sunarlar.

Sabit λ değerleri ile tüm α kesimleri için özellik ağırlığının tanımı

$[W^{(\alpha, \lambda)}_j = \{w^{(\alpha, \lambda)}_j, \forall \alpha \in [0, 1], \}$, bütün özellikleri için $j=1, 2, \dots, m$] şeklindedir.

Yukarıdaki tanımlamayı kullanarak çeşitli seçim özelliklerinin tercih edilme düzeylerini ölçecek bir vektör tanımlanabilir.

$$W_{Bileşil_j}^\lambda = \frac{\sum_\alpha w_j^{(\alpha, \lambda)} \alpha}{\sum_\alpha \alpha}, \quad j = 1, \dots, m; \quad (2.13)$$

Chang'ın bulanık boyut analiz metodu ise şu şekildedir. X, bir hedef kümesi

$X = \{x_1, x_2, \dots, x_n\}$ ve

U, bir amaç kümesi

$U = \{u_1, u_2, \dots, u_m\}$ olsun.

Chang'ın boyut analizi metoduna göre her bir hedef alınır ve her bir boyut analizi (g_i) sırasıyla uygulanır. Bu şekilde, aşağıda gösterildiği gibi, her bir hedef için m boyut analizi değeri elde edilir.

$$M_{g_i}^1, M_{g_i}^2, \dots, M_{g_i}^m, \quad i = 1, 2, \dots, n, \quad (2.14)$$

Burada belirtilen tüm $M_{g_i}^j$ ($j = 1, 2, \dots, m$) parametreleri l , m ve u olan üçgensel bulanık sayılardır. Daha önce bulanık sayılar konusunda anlatıldığı gibi bunlar sırasıyla, en az olası değer, en olası değer ve en geniş olası değerdir. Bir üçgensel bulanık sayı (TFN) (l, m, u) olarak ifade edilir. Aşağıda Chang'ın boyut analizi metodu adımlarıyla verilmiştir [21].

ADIM 1 : i . hedefe göre bulanık sentetik boyut değeri aşağıdaki gibi tanımlanabilir;

$$S_i = \sum_{j=1}^m M_{g_i}^j \otimes \left[\sum_{i=1}^n \sum_{j=1}^m M_{g_i}^j \right]^{-1} \quad (2.15)$$

$\sum_{j=1}^m M_{g_i}^j$, 'i elde etmek için aşağıdaki şekilde tanımlı bir matris için m boyut analiz değerlerinin ilave bulanık işlemleri uygulanır.

$$\sum_{j=1}^m M_{g_i}^j = \left(\sum_{j=1}^m l_j, \sum_{j=1}^m m_j, \sum_{j=1}^m u_j \right) \text{ ve}$$

$\left[\sum_{i=1}^n \sum_{j=1}^m M_{g_i}^j \right]^{-1}$ 'i bulmak için ters vektörün hesaplanmasından sonra ve

$$\sum_{i=1}^n \sum_{j=1}^m M_{g_i}^j = \left(\sum_{j=1}^m l_j, \sum_{j=1}^m m_j, \sum_{j=1}^m u_j \right) \text{ ifadesinden faydalananak } M_{g_i}^j \text{ (} j = 1, 2, \dots, m \text{)}$$

değerlerinin bulanık ilave operasyonları gerçekleştirilir ve aşağıdaki eşitlik elde edilir.

$$\left[\sum_{i=1}^n \sum_{j=1}^m M_{g_i}^j \right]^{-1} = \left(\frac{1}{\sum_{i=1}^n u_i}, \frac{1}{\sum_{i=1}^n m_i}, \frac{1}{\sum_{i=1}^n l_i} \right) \quad (2.16)$$

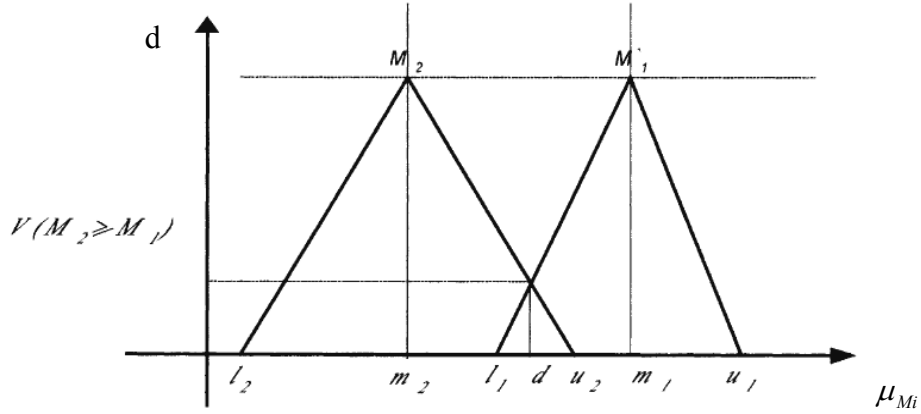
ADIM 2 : $M_2 = (l_2, m_2, u_2) \geq M_1 = (l_1, m_1, u_1)$ ifadesinin olabilirlik derecesi şöyle tanımlanır.

$$V(M_2 \geq M_1) = \sup_{y \geq x} [\min(\mu_{M_1}(x), \mu_{M_2}(y))] \quad (2.17)$$

Bu denklemin anlamı μ_{M_1} ve μ_{M_2} arasındaki bulanık bağıntının güçsüz olanlarından bir küme oluştur ve bu güçsüzlerin en güçlüsünü seç demektir. Kısaca sup-min ilişkisi kötülerin en iyisi anlamına gelir [12]. Bu eşitlik aşağıdaki gibide gösterilebilir.

$$V(M_2 \geq M_1) = \mu_{M_2}(d) = \begin{cases} 1, & \text{eğer } m_2 \geq m_1 \\ 0, & \text{eğer } l_1 \geq u_2 \\ \frac{l_1 - u_2}{(m_2 - u_2) - (m_1 - l_1)}, & \text{değilse} \end{cases} \quad (2.18)$$

Yukarıda geçen d , μ_{M_1} ve μ_{M_2} arasındaki en yüksek kesişim noktası olan D 'nin ordinatıdır. Şekil 2.13'de M_1 ve M_2 'nin kesişimi görülmektedir M_1 ve M_2 'yi karşılaştırmak için hem $V(M_1 \geq M_2)$ hem de $V(M_2 \geq M_1)$ değerlerini bilmek gerekir.



Şekil 2.13. M_1 ve M_2 'nin kesişimi [21]

ADIM 3: Bir bulanık konveks sayı M 'nin, k konveks bulanık sayıdan M_i ($i = 1, 2, \dots, k$) daha büyük olması için olabilirlik derecesi şöyle tanımlanır.

$$V(M \geq M_1, M_2, \dots, M_k) = V[(M \geq M_1) \text{ ve } (M \geq M_2) \text{ ve } \dots, \text{ ve } (M \geq M_k)] \min V(M \geq M_i), i=1, 2, \dots, k$$

varsayılan

$$d^n(A_i) = \min V(S_i \geq S_k)$$

$k = 1, 2, \dots, n$ için $k \neq i$. ağırlık vektörü aşağıdaki gibi verilebilir.

$$W' = (d(A_1), d(A_2), \dots, d(A_n))^T \quad A_i (i=1, 2, \dots, n) \text{ olmak üzere } A_i, n \text{ elemanlıdır.}$$

ADIM 4 : Normalizasyon işlemi ile normalize edilen ağırlık vektörü aşağıdaki gibi elde edilir.

$W = (d(A_1), d(A_2), \dots, d(A_n))^T$ burada W bulanık olmayan bir sayıdır ve yapılan işlemler sonucu elde edilen W sayısı bize seçim işlemine tabi tutulan adaylara ait nihai puanları vermektedir.

Bulanık AHS ile ilgili yapılan çalışmalar

Farklı yazarlar tarafından önerilen çok çeşitli bulanık AHS metotları vardır. Bu metotlar alternatif seçimlere sistematik yaklaşımlar metodu ve hiyerarşik yapı analizi ve bulanık küme teorisi kavramlarını kullanarak problem doğrulama metodudur [38].

İnsanlar genellikle karar verirken aralıklı yargıları sabit değer yargılarından daha uygun bulmaktadırlar. Bu, karşılaştırma işleminin doğasındaki bulanıklıktan kaynaklanmaktadır. Ancak aynı bulanık yaklaşım bilgisayarlar tarafından sağlanamamaktadır. Bu amaçla geliştirilen bulanık AHS teorisi ile ilgili yapılan çalışmalar aşağıda sıralanmıştır.

Laarhoven ve Pedrycz tarafından 1983 yılında yapılmıştır, bu çalışma üçgensel üyelik fonksiyonları ile tanımlanmış bulanık oranları karşılaştırır [41].

Bunckley 1985 yılında, ikizkenar yamuk üyelik fonksiyonlarını kullanarak karşılaştırma oranlarının bulanık önceliklerine karar vermiştir [42].

Stam ve arkadaşları 1996 yılında AHS'de yaklaşık olarak öncelik sınıflandırılmasını veya karar vermeyi sağlayan yapay zeka tekniklerini geliştirdiler. Stam ve arkadaşları kesin olmayan ve bulanık oran-ölçek öncelikli yargılarıyla alternatif çoklu kriter karar problemleri çözümlemesi için ileri besleme sinirsel ağ formülasyonunun güçlü bir araç olduğu sonucuna varmıştır [43].

Chang 1996 yılında, Bulanık AHS'nin ikili karşılaştırma skalası için üçgensel bulanık sayılar ve ikili karşılaştırmanın yapay (synthetic) boyut değerleri için boyut analiz metodunu kullanarak Bulanık AHS için yeni bir yaklaşım geliştirmiştir [44].

Cheng 1997 yılında, üyelik fonksiyonlarının dereceli değerlerine dayalı Bulanık AHS'yi kullanarak savaş gemilerine ait taktiksel kurşun sistemlerinin değerlendirilmesinde yeni bir algoritma önermiştir [45].

Weck ve arkadaşları 1997 yılında, klasik AHS'ye bulanık mantığı matematiksel olarak ilave ederek, farklı üretim çevirim alternatiflerinin değerlendirilmesine yönelik metot sunmuştur. Burada değerlendirilen her üretim çevirimi, bir bulanık küme olarak düşünülmüştür. Her bulanık kümenin yüzey ağırlık merkezi oluşturularak durulaştırma sağlanır ve araştırılan alternatif üretim çevirimleri ana hedef kümesi baz alınarak öncelik sırasına göre sıralanmıştır [46].

Kahraman 1998 yılında, AHS'den ve bulanık ağırlıklı değerlendirmeden ağırlıkları bulmak için bulanık nesnel ve öznel bir metot kullanmıştır [47].

Deng 1999 yılında, kolay ve anlaşılır yapıdaki nitel çok kriterli analiz problemleri için bir bulanık yaklaşımı üzerinde durmuştur [34].

Lee ve arkadaşları 1999 yılında, AHS'nin temelindeki düşünceleri gözden geçirmiştir. Bu düşüncelere dayanarak "karşılaştırma aralığı" kavramını ortaya

koymuřtur. Karřılařtırma sũrecinin bulanık yapısını uyumlařtırmak iin stokastik optimizasyona dayalı bir metodoloji nermiřtir [17].

Cheng ve arkadařları 1999 yılında, szel deęiřken aralıęına dayalı AHS ile silah sistemlerini deęerlendirmede yeni bir metot geliřtirmiřtir [48].

Zhu ve arkadařları 1999 yılında, Bulanık AHS'nin uygulamaları ve boyut analiz metodu hakkında bir makale yayınlamıřtır [49].

Chan ve arkadařları 2000 yılında kararsızlık ierisinde maddi ve manevi yararları lmek iin bir teknoloji seim algoritması sundu. Chan ve arkadařları bulanık kũme teorisini ekonomik deęerlendirme ve hiyerarřık yapısal analiz ũzerinde uygulamıřlardır. Kũmelendirilmiř hiyerarřı ile her alternatif teknolojinin aęırlıęı bulunur. Bu yntemin adı da bulanık uygun indeks'tir. Bulanık uygun indeks ile deęerlendirmek istenen eřitli teknolojiler tercih edilen sıraya gre sıralanarak aralarından en uygun teknoloji bulunur. Chan ve arkadařlarının yaptıęı bu uygulama ile ekonomik deęerlendirme aısından bir bulanık nakit akıřı analizi gerekleřtirilmiřtir [39].

Chan ve arkadařları yine 2000 yılında, ok kriterli karar verme tekniklerini ve simũlasyonu kullanan esnek imalat sistemlerinin otomatik tasarımı iin bũtũnleřik bir yaklařımdan sz etmiřtir. Tasarım sũreci, simũlasyon metotlarını kullanan alternatif tasarımların oluřturulması ve test edilmesi alıřmalarını ierir. AHS'ye dayalı en uygun tasarımın seilmesi, esnek imalat sistemi simũlasyon modellerinden elde edilen ıktının analizi iin kullanılır. Esnek imalat sistemleri tasarım sũrecini desteklemek iin akıllı aralar(uzman sistemler, bulanık sistemler, sinir aęları vb.) geliřtirilir. Akıllı karar destek sũreci ve esnek imalat sistemleri otomatik tasarım sũrecinin tam entegrasyonu iin "Aktif X" teknięi kullanılır [50].

Leung ve Cao 2000 yılında verilen kriter aęırlıklarında meydana gelebilecek sapma toleransını dũřunerek bir bulanık tutarlılık tanımı nermiřlerdir. Bu greceli nemin bulanık oranı ki bu oran belirli bir sapmayı hesaba katar, yerel nceliklerin ũyelik

değerlerindeki kısıtlamalar gibi formüle edilir. Aslında büyüme prensipleri ile bulanık yerel ve genel ağırlıklara karar verilir. Maksimum-minimum grup sıralama metodu uygulamasıyla genel ağırlıkların temelinde alternatifler sıralanır [51].

Kuo ve arkadaşları 2002 yılında, yeni bir tesisin yer seçim problemi için bir karar destek sistemi geliştirmiştir. Bu sistemin ilk elemanı Bulanık AHS süreci için hiyerarşik yapının oluşturulmasıdır. Devamında, ilk olarak Bulanık AHS için boyut analiz metodu verilmiş ve daha sonra metot katering firması seçim problemine uygulanmıştır [16].

Kahraman, Cebeci ve Ruan ise 2004 yılında, katering servis şirketleri arasından çok ölçütlü karar verme problemini ele almıştır. Çalışmada, Türkiye'deki bir duruma bulanık AHS uygulanmıştır [38].

3. MATERYAL VE METOD

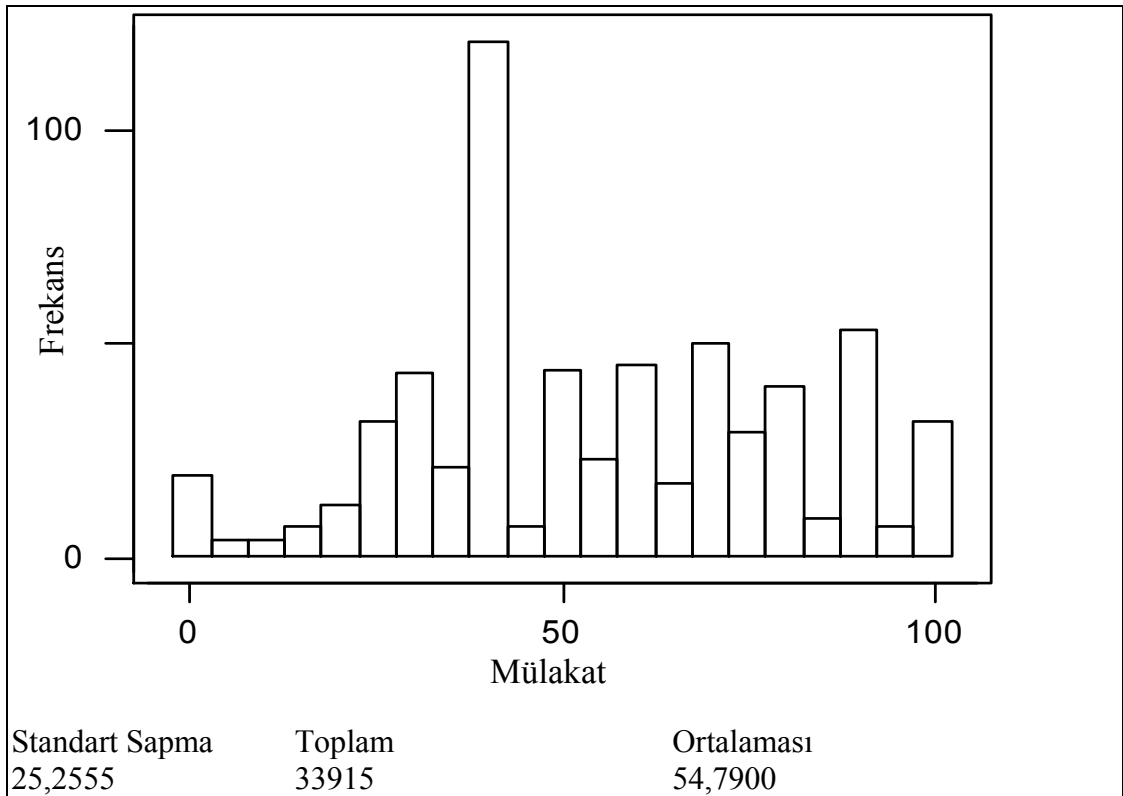
Bu bölümde, kullanılan klasik yöntemin ve çok kriterli karar verme yöntemlerinden AHS ve Bulanık AHS yöntemlerinin algoritmaları fen bilimleri enstitüleri için yüksek lisans öğrencisi seçimi problemine uygulanarak incelenmiştir.

Öncelikle yöntemlerin karşılaştırılması işleminde adaylara ait gerçek verileri kullanarak adayların aldığı puanların dağılımına bakılmıştır.

Minitab istatistiksel paket programında 2004-2005 eğitim öğretim yılında Gazi Üniversitesi Fen Bilimleri Enstitüsü'ne yüksek lisans düzeyinde eğitim için başvuran 619 adayın mülakat notları girilmiş ve SPSS paket programında Histogram grafiği oluşturulmuştur.

Çizelge 3.1. Histogram grafiğinde kullanılan değerler

Orta nokta	Sayı	Frekans
0	20	****
10	9	**
20	24	*****
30	72	*****
40	139	*****
50	54	*****
60	65	*****
70	68	*****
80	67	*****
90	63	*****
100	38	*****



Şekil 3.1. Histogram grafiği

Elde edilen sonuç değerlendirildiğinde tam bir normal dağılım oluşturmadığı tespit edilmiştir. Şekil 3.1’de görülen grafik ile elde edilen sonuç bize adayları değerlendirmek amacıyla verilen puanların normal dağılıma uymadığını göstermektedir.

Daha sonra bu dağılıma uygun değerleri rastgele üreterek, klasik yöntem, AHS yöntemi ve BAHS yönteminin algoritmalarına uygulayan ve maksimum 2000 aday arasından istenilen sayıdaki adayı seçen bir simülasyon Digital Fortran 5.0 yazılım geliştirme programında hazırlanmıştır. 2000 aday olmasının sebebi, simülasyon programının işletilmesi sırasında bilgisayar belleğinin işlem karmaşasından dolayı zorlanması ve işlemlerin yavaş gerçekleştirilmesidir. Ayrıca bu aday sayısı gerçek hayattaki örnek çapına kıyasla yeterli bir miktardır. Simülasyonun işletilmesi sonucu 3 yönetime göre elde edilen sonuçlar Minitab 13 ve SPSS 9.05 istatistiksel veri analizi programlarında karşılaştırılmıştır. Aşağıda kullanılan simülasyon ve istatistiksel yöntemlerden ayrıca bahsedilmiştir.

Simülasyonun amacı yüksek miktarlardaki veri yığınlarını işleyerek daha geniş bir örnekte AHS ve BAHS algoritmalarını uygulayabilmektir. Böylelikle daha gerçek ve geçerli neticelere ulaşılarak kullanılan klasik yöntemin yeterli olup olmadığının, diğer yöntemlerin seçim işleminde gösterdikleri başarı miktarının ve 3 yöntemin birbirinden ne derece farklı seçim yaptığının, tespit edilmesi amaçlanmaktadır.

Bu amaçla, halen kullanımda olan yöntem, AHS ve Bulanık AHS yöntemleri için örnek birer uygulama sunulmuştur. Simülasyonda 2000 aday için rastgele üretilen puanlar algoritmalarda kullanılmıştır ancak işlem karmaşasına neden olmadan anlaşılabilirliği arttırmak amacıyla burada yalnızca 5 birey ve 3 karar verici kullanılarak Klasik Yöntem, AHS ve BAHS yöntemleri uygulanacaktır. Ancak yöntemlerin değerlendirilmesi işleminde, kullanılan simülasyondan elde edilen sonuçlar değerlendirilecektir.

3.1. Uygulanan Mülakat Sınavı ve Esasları

Çok kriterli karar verme yöntemlerinden hem AHS’de ve hem de bulanık AHS’de matematiksel bir modelin oluşturulmasının zorunluluğundan dolayı örnekte uygulanmak üzere değerlendirme amacı ile bir aday değerlendirme formu hazırlanmıştır. Aşağıda örnek olarak verilen aday değerlendirme çizelgesinde konulara göre alınacak puanlar toplamının 100 üzerinden en az 50 olması gerekmektedir.

Çizelge 3.2. Aday değerlendirme formu [8]

Değerlendirme Kriterleri	Değerlendirme Puanı	Aldığı Puan
Genel Görünüşü	20	
Anlama ve Anlatma Yeteneği	20	
Aile ve Sosyal Durumu	20	
Psikolojik Yapısı	20	
Bilimsel Yeterliliği	20	
TOPLAM	100	

3.2. Geçerli Yöntem ile Yüksek Lisans Öğrencisi Seçimi

Örneğimizde FBE'ye başvuran ve mülakat aşamasına gelen 5 adayın, 3 karar verici tarafından değerlendirildiği düşünölsün ve değerlendirme sonucu ilk 2 aday için kontenjan bulunsun. Aşağıdaki çizelgede karar vericiler tarafından verildiği varsayılan puanlar ve ortalamaları gösterilmektedir. (Genel Görünüşü (GG), Anlama ve Anlatma Yeteneği (AAY), Aile ve Sosyal Durumu (ASD), Psikolojik Yapısı (PY), Bilimsel Yeterliliği (BY))

Çizelge 3.3. Birinci karar vericinin adaylara verdiği puanlar

KRİTERLER	ADAYLAR				
	1	2	3	4	5
Genel Görünüşü	20	19	12	17	14
Anlama ve Anlatma Yeteneği	18	15	17	19	19
Aile ve Sosyal Durumu	15	20	18	13	17
Psikolojik Yapısı	15	18	14	20	17
Bilimsel Yeterliliği	17	15	17	16	16
TOPLAM	85	87	78	85	83

Çizelge 3.4. İkinci karar vericinin adaylara verdiği puanlar

KRİTERLER	ADAYLAR				
	1	2	3	4	5
Genel Görünüşü	19	18	15	16	15
Anlama ve Anlatma Yeteneği	19	14	17	18	19
Aile ve Sosyal Durumu	16	19	18	14	16
Psikolojik Yapısı	16	19	15	19	16
Bilimsel Yeterliliği	18	15	17	16	15
TOPLAM	88	85	82	83	81

Çizelge 3.5. Üçüncü karar vericinin adaylara verdiği puanlar

KRİTERLER	ADAYLAR				
	1	2	3	4	5
Genel Görünüşü	20	20	17	18	17
Anlama ve Anlatma Yeteneği	17	13	18	15	18
Aile ve Sosyal Durumu	18	20	18	18	18
Psikolojik Yapısı	18	20	17	18	17
Bilimsel Yeterliliği	18	17	16	17	16
TOPLAM	91	90	86	86	86

Çizelge 3.6. Karar vericilerin adaylara verdiği puanların ortalamaları

	ADAYLAR				
	1	2	3	4	5
ORTALAMA	88	87	82	85	83

Yukarıda ortalamaların verildiği çizelge sıralandığında oluşan sıralama aşağıdaki gibi olmaktadır.

1 – 2 – 4 – 5 – 3

Bu sıralamaya göre ilk iki aday olan 1. ve 2. adayların seçilmesi uygun olmaktadır.

3.3. AHS Yöntemi ile Yüksek Lisans Öğrencisi Seçimi

Öncelikle AHS için ilk aşama olan AHS değerlendirme ölçeğine göre kriterlerin ikili karşılaştırılarak birbirlerine göre önem derecelerinin ve kriter ağırlıklarının hesaplanması gerekmektedir. Daha sonra hesaplanan bu ağırlıkların tutarlılık oranları hesaplanarak geçerlilikleri test edilecektir. Bu amaçla ilk olarak karar vericilerin

AHS değerlendirme ölçeğine göre 1'den 9'a kadar kriterleri değerlendirmeleri gerekmektedir [8].

Çizelge 3.7. Kriterlere ait ikili kıyaslama matrisi

Kriterler	GG	AAV	ASD	PY	BY
GG	1	1/2	4	3	1/5
AAV	2	1	5	5	2
ASD	1/4	1/5	1	1	1/5
PY	1/3	1/5	1	1	1/4
BY	5	1/2	5	4	1

Çizelge 3.8. İkili kıyaslama sütunlar toplamı

Kriterler	GG	AAV	ASD	PY	BY
Sütunlar Toplamı	8,5833	2,4	16	14	3,65

Çizelge 3.7'de kriterlere ait ikili kıyaslamaların sonuçları bir matris haline getirilmiş ve Çizelge 3.8'de ise ikili kıyaslamalar sonucu oluşan matrisin sütun toplamaları görülmektedir. Çizelge 3.7'de oluşturulan ikili kıyaslama matrisinde en sol sütunda bulunan kriterin en üst sütunda bulunan kritere olan üstünlüğü gösterilmiştir. Birbirinin aynı kriterlerin karşılaştırılmasında fark olmayacağı için karşılaştırma 1 olarak derecelendirilmiştir.

Çizelge 3.9'daki matriste ise ikili kıyaslama matrisindeki değerler sütun toplamına bölünmüş, satır toplamaları alınarak normalizasyon yapılmıştır. Böylelikle ağırlıkları yani öncelikleri bulunmuştur.

Çizelge 3.9 Kriter ağırlıkları matrisi

Kriterler	GG	AAV	ASD	PY	BY	Kriter Ağırlıkları
GG	0,1165	0,2083	0,2500	0,2143	0,0548	0,1688
AAV	0,2330	0,4167	0,3125	0,3571	0,5479	0,3735
ASD	0,0291	0,0833	0,0625	0,0714	0,0548	0,0602
PY	0,0388	0,0833	0,0625	0,0714	0,0685	0,0649
BY	0,5825	0,2083	0,3125	0,2857	0,2740	0,3326

Yukarıdaki çizelgede hesaplanan kriter ağırlıkları gösterilmektedir. Kriter ağırlıkları bulunduğundan sonra yapılması gereken tutarlılık oranının hesaplanmasıdır.

Tutarlılık oranı (C.R.), tutarsızlık endeks değerinin(C.I.) rassal endeks değerine(R.I.) oranı olarak tanımlanmaktadır.

$$C.R. = C.I. / R.I. \quad (3.1)$$

Burada λ_{maks} maksimum özdeğeri, n ise aday sayısını temsil etmektedir. Tutarlılık oranının bulunabilmesi için öncelikle ikili kıyaslama matrisi ile kriter ağırlıkları matrisi çarpılmalıdır.

$$\begin{bmatrix} 1 & 1/2 & 4 & 3 & 1/5 \\ 2 & 1 & 5 & 5 & 2 \\ 1/4 & 1/5 & 1 & 1 & 1/5 \\ 1/3 & 1/5 & 1 & 1 & 1/4 \\ 5 & 1/2 & 5 & 4 & 1 \end{bmatrix} \begin{bmatrix} 0,1688 \\ 0,3735 \\ 0,0602 \\ 0,0649 \\ 0,3326 \end{bmatrix} = \begin{bmatrix} 0,8577 \\ 2,0020 \\ 0,3086 \\ 0,3393 \\ 1,9241 \end{bmatrix}$$

$$\lambda_{maks} = \frac{1}{5} * \left(\frac{0,8577}{0,1688} + \frac{2,0020}{0,3735} + \frac{0,3086}{0,0602} + \frac{0,3393}{0,0649} + \frac{1,9241}{0,3326} \right) = 5,3152$$

$$C.I. = (5,3152 - 5) / 4 = 0,0788$$

Bulunan 0,0788 değeri tutarlılık sınırı olarak kabul edilen %10 yani 0,1'den küçük olduğu için kriterlere ait ağırlıklar uygundur denilebilir. Bu aşamadan sonraki adım karar vericilerin her bir kriter için adayların ikili karşılaştırmalarını yapmalarıdır.

Genel görünüş

Çizelge 3.10 genel görünüş kriterine göre adayların ikili karşılaştırmalarını gösteren matristir.

Çizelge 3.10. Genel görünüş kriterine göre ikili karşılaştırma

	A1	A2	A3	A4	A5
A1	1,0000	0,5000	2,0000	0,3333	0,5000
A2	2,0000	1,0000	2,0000	0,3333	0,5000
A3	0,5000	0,5000	1,0000	0,3333	0,5000
A4	3,0000	3,0000	3,0000	1,0000	2,0000
A5	2,0000	2,0000	2,0000	0,5000	1,0000
Sütunlar Toplamı	8,5000	7,0000	10,0000	2,5000	4,5000

Çizelge 3.11. GG kriterine göre aday ağırlıkları matrisi

	A1	A2	A3	A4	A5	Aday Ağırlıkları
A1	0,1176	0,0714	0,2000	0,1333	0,1111	0,1267
A2	0,2353	0,1429	0,2000	0,1333	0,1111	0,1645
A3	0,0588	0,0714	0,1000	0,1333	0,1111	0,0949
A4	0,3529	0,4286	0,3000	0,4000	0,4444	0,3852
A5	0,2353	0,2857	0,2000	0,2000	0,2222	0,2286

Kriterler içinde yapılan tutarlılık oranı hesaplanması burada da yapılarak elde edilen aday ağırlıklarının tutarlılığı ölçülür. Aynı formül uygulanarak tutarlılık oranı hesaplanmış ve 0,0369 olarak bulunmuştur ve %10 (0,1)'den küçük olduğu için tutarsızlık kabul edilebilir.

Anlama ve anlatma yeteneği

Çizelge 3.12 anlama ve anlatma yeteneği kriterine göre adayların ikili karşılaştırmalarını gösteren matristir.

Çizelge 3.12. Anlama ve anlatma yeteneği göre ikili karşılaştırma

	A1	A2	A3	A4	A5
A1	1,0000	0,2000	0,2500	0,3333	1,0000
A2	5,0000	1,0000	2,0000	3,0000	5,0000
A3	4,0000	0,5000	1,0000	0,3333	3,0000
A4	3,0000	0,3333	3,0000	1,0000	4,0000
A5	1,0000	0,2000	0,3333	0,2500	1,0000
Sütunlar Toplamı	14,0000	2,2333	6,5833	4,9167	14,0000

Çizelge 3.13. Anlama ve anlatma yeteneği kriterine göre aday ağırlıkları matrisi

	A1	A2	A3	A4	A5	Aday Ağırlıkları
A1	0,0714	0,0896	0,0380	0,0678	0,0714	0,0676
A2	0,3571	0,4478	0,3038	0,6102	0,3571	0,4152
A3	0,2857	0,2239	0,1519	0,0678	0,2143	0,1887
A4	0,2143	0,1493	0,4557	0,2034	0,2857	0,2617
A5	0,0714	0,0896	0,0506	0,0508	0,0714	0,0668

Kriterler içinde yapılan tutarlılık oranı hesaplanması burada da yapılarak elde edilen aday ağırlıklarının tutarlılığı ölçülür. Aynı formül uygulanarak tutarlılık oranı hesaplanmış ve 0,0672 olarak bulunmuştur ve %10 (0,1)'den küçük olduğu için tutarsızlık kabul edilebilir.

Aile ve sosyal durum

Çizelge 3.14 aile ve sosyal durum kriterine göre adayların ikili karşılaştırılmalarını gösteren matristir.

Çizelge 3.14. Aile ve sosyal durum kriterine göre ikili karşılaştırma

	A1	A2	A3	A4	A5
A1	1,0000	0,5000	2,0000	0,3333	0,5000
A2	2,0000	1,0000	3,0000	0,3333	0,5000
A3	0,5000	0,3333	1,0000	0,2500	0,3333
A4	3,0000	3,0000	4,0000	1,0000	3,0000
A5	2,0000	2,0000	3,0000	0,3333	1,0000
Sütunlar Toplamı	8,5000	6,8333	13,0000	2,2500	5,3333

Çizelge 3.15. Aile ve sosyal durum kriterine göre aday ağırlıkları matrisi

	A1	A2	A3	A4	A5	Aday Ağırlıkları
A1	0,1176	0,0732	0,1538	0,1481	0,0938	0,1173
A2	0,2353	0,1463	0,2308	0,1481	0,0938	0,1709
A3	0,0588	0,0488	0,0769	0,1111	0,0625	0,0716
A4	0,3529	0,4390	0,3077	0,4444	0,5625	0,4213
A5	0,2353	0,2927	0,2308	0,1481	0,1875	0,2189

Kriterler içinde yapılan tutarlılık oranı hesaplanması burada da yapılarak elde edilen aday ağırlıklarının tutarlılığı ölçülür. Aynı formül uygulanarak tutarlılık oranı hesaplanmış ve 0,0411 olarak bulunmuştur ve %10 (0,1)'den küçük olduğu için tutarsızlık kabul edilebilir.

Psikolojik yapı

Çizelge 3.16 psikolojik yapı kriterine göre adayların ikili karşılaştırılmalarını gösteren matristir.

Çizelge 3.16. Psikolojik yapı kriterine göre ikili karşılaştırma

	A1	A2	A3	A4	A5
A1	1,0000	0,5000	2,0000	0,3333	0,5000
A2	2,0000	1,0000	3,0000	0,3333	2,0000
A3	0,5000	0,3333	1,0000	0,2500	0,5000
A4	3,0000	3,0000	4,0000	1,0000	3,0000
A5	2,0000	0,5000	2,0000	0,3333	1,0000
Sütunlar Toplamı	8,5000	5,3333	12,0000	2,2500	7,0000

Çizelge 3.17. Psikolojik yapı kriterine göre aday ağırlıkları matrisi

	A1	A2	A3	A4	A5	Aday Ağırlıkları
A1	0,1176	0,0938	0,1667	0,1481	0,0714	0,1195
A2	0,2353	0,1875	0,2500	0,1481	0,2857	0,2213
A3	0,0588	0,0625	0,0833	0,1111	0,0714	0,0774
A4	0,3529	0,5625	0,3333	0,4444	0,4286	0,4244
A5	0,2353	0,0938	0,1667	0,1481	0,1429	0,1573

Kriterler içinde yapılan tutarlılık oranı hesaplanması burada da yapılarak elde edilen aday ağırlıklarının tutarlılığı ölçülür. Aynı formül uygulanarak tutarlılık oranı hesaplanmış ve 0,0361 olarak bulunmuştur ve %10 (0,1)'den küçük olduğu için tutarsızlık kabul edilebilir.

Bilimsel yeterlilik

Çizelge 3.18 bilimsel yeterlilik kriterine göre adayların ikili karşılaştırılmalarını gösteren matristir.

Çizelge 3.18. Bilimsel yeterlilik kriterine göre ikili karşılaştırma

	A1	A2	A3	A4	A5
A1	1,0000	0,5000	0,3333	0,3333	2,0000
A2	2,0000	1,0000	0,5000	0,3333	2,0000
A3	3,0000	2,0000	1,0000	1,0000	3,0000
A4	3,0000	3,0000	1,0000	1,0000	2,0000
A5	0,5000	0,5000	0,3333	0,5000	1,0000
Sütunlar Toplamı	9,5000	7,0000	3,1667	3,1667	10,0000

Çizelge 3.19. Bilimsel yeterlilik kriterine göre aday ağırlıkları matrisi

	A1	A2	A3	A4	A5	Aday Ağırlıkları
A1	0,1053	0,0714	0,1053	0,1053	0,2000	0,1174
A2	0,2105	0,1429	0,1579	0,1053	0,2000	0,1633
A3	0,3158	0,2857	0,3158	0,3158	0,3000	0,3066
A4	0,3158	0,4286	0,3158	0,3158	0,2000	0,3152
A5	0,0526	0,0714	0,1053	0,1579	0,1000	0,0974

Kriterler içinde yapılan tutarlılık oranı hesaplanması burada da yapılarak elde edilen aday ağırlıklarının tutarlılığı ölçülür. Aynı formül uygulanarak tutarlılık oranı hesaplanmış ve 0,0450 olarak bulunmuştur ve %10 (0,1)'den küçük olduğu için tutarsızlık kabul edilebilir.

Birleştirilmiş karar matrisinin oluşturulması

Birleştirilmiş karar matrisi, kriter ağırlıkları bulunduktan ve her bir kritere göre adayların ikili karşılaştırmaları işlemi yapıldıktan sonra elde edilen bu sonuçların bir çizelgede toplanması ve her bir adayın aldığı ağırlığın hesaplanması gerekmektedir.

Her bir adayın ağırlığının hesaplanması için kriter ağırlığı ile adayın o kritere göre aldığı değerlendirme puanı çarpılır ve tüm çarpım sonuçları toplanır. Bu işlem aşağıdaki formülle sembolize edilir.

$$OT_i = \sum_{j=1}^p [T_{ij} / (\sum_{i=1}^m T_{ij})] / p \quad (3.2)$$

Çizelge 3.20. Birleştirilmiş karar matrisi

Kriterler	Kriter Ağırlıkları	ADAYLAR				
		Aday 1	Aday 2	Aday 3	Aday 4	Aday 5
GG	0,1688	0,1267	0,1645	0,0949	0,3852	0,2286
AAY	0,3735	0,0676	0,4152	0,1887	0,2617	0,0668
ASD	0,0602	0,1173	0,1709	0,0716	0,4213	0,2189
PY	0,0649	0,1195	0,2213	0,0774	0,4244	0,1573
BY	0,3326	0,1174	0,1633	0,3066	0,3152	0,0974
TOPLAM		0,189573	0,21705	0,18809	0,187348	0,152119

Tüm adayların toplam ağırlıkları sıraya konularak en yüksek ağırlıkta olan iki aday seçilir.

Aday 2, Aday 1, Aday 3, Aday 4, Aday 5

Bu durumda en yüksek ağırlığa sahip olan 2. ve 1. adayların seçilmesi uygun olacaktır.

3.4. Bulanık AHS Yöntemi ile Yüksek Lisans Öğrencisi Seçimi

Çalışmada hem kriterler için hem de adaylar için ayrı ayrı bulanık denetleyici tasarımı yapılmıştır. Kriterler için bulanıklaştırma biriminin giriş bilgisini, karar vericinin kriterler için verdiği 0-20 aralığındaki puanlar, adaylar için bulanıklaştırma biriminin giriş bilgisini ise yine karar vericinin adaylar için verdiği 0-20 aralığındaki puanlar oluşturmuştur.

Aşağıda kriterler için karar vericinin verdiği 0-20 aralığındaki puanlar verilmiştir.

Çizelge 3.21. Kriterler için karar vericinin verdiği puanlar

KRİTERLER	PUANLAR
Anlama Anlatma Yeteneği	16
Genel Görünüş	8
Psikolojik Yapı	12
Ailevi ve Sosyal Durum	7
Bilimsel Yeterlilik	20

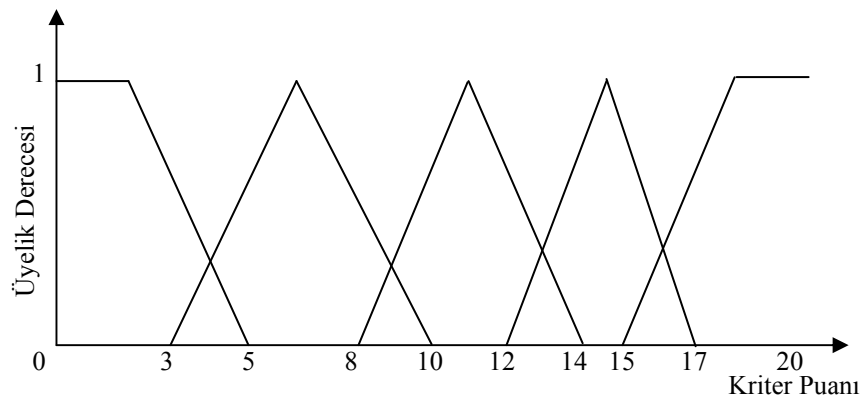
Aşağıda her bir aday için her bir kriterden karar vericinin verdiği 0-20 aralığındaki puanlar verilmiştir.

Çizelge 3.22. Adaylar için karar vericinin verdiği puanlar

Aday Kriter	Aday 1	Aday 2	Aday 3	Aday 4	Aday 5
AAY	13	10	19	14	12
GG	17	13	12	20	18
PS	20	17	16	12	17
ASD	20	16	16	20	19
BY	12	18	10	17	16

Kriterler için bulanıklaştırma birimi

Bulanıklaştırma, sistemden alınan denetim giriş bilgilerini dilsel niteleyiciler olan sembolik değerlere dönüştürme işlemidir. Üyelik işlevinden faydalanılarak giriş bilgilerinin ait olduğu bulanık kümeyi/kümeleri ve üyelik derecesini tespit edip, girilen sayısal değere küçük, en küçük gibi dilsel değişken değerler atar. Sistemin verimli çalışmasını sağlamak amacıyla değişik şekillerde (üçgen, yamuk, çan eğrisi... vs.) bulanık kümeler seçilebilir [20]. Çalışmada üçgen üyelik kümesi tercih edilmiştir. 0-20 arası kriter puanlarına göre Şekil 3.2'deki üyelikler oluşturulmuştur. Üçgen şeklindeki üyelik kümelerinin üyelik derecelerinin belirlenmesinde aşağıdaki eşitlik kullanılmıştır. Bu üyelikler ve fonksiyonlar kullanılarak kriterlerin alacağı değerlere göre hangi üyelik kümesinin üyesi olacağı Çizelge 3.23'de görülebilir.



Şekil 3.2. Kriter puanlarına ait üçgen üyelik kümeleri ve üyelik fonksiyonu dereceleri

Çizelge 3.23. Kriter puanlarına ait üyelik fonksiyonlarının oluşturulması

Üyelik Kümeleri	Fonksiyon	Değer Aralığı
ÖNEMSİZ	$\mu(x) = 1$	$x \leq 2$
	$\mu(x) = (5-x) / (5-2)$	$2 \leq x \leq 5$
ZAYIF	$\mu(x) = (x-3) / (6,5-3)$	$3 \leq x \leq 6,5$
	$\mu(x) = (10-x) / (10-6,5)$	$6,5 \leq x \leq 10$
ÖNEMLİ	$\mu(x) = (x-8) / (11-8)$	$8 \leq x \leq 11$
	$\mu(x) = (14-x) / (14-11)$	$11 \leq x \leq 14$
ÇOK ÖNEMLİ	$\mu(x) = (x-12) / (14,5-12)$	$12 \leq x \leq 14,5$
	$\mu(x) = (17-x) / (17-14,5)$	$14,5 \leq x \leq 17$
MÜKEMMEL	$\mu(x) = (x-15) / (18-15)$	$15 \leq x \leq 18$
	$\mu(x) = 1$	$x \geq 18$

Çizelge 3.24. Karar vericinin verdiği puanlara göre kriterlerin aitlik kümeleri ve dereceleri

Kriterler	Puanlar	Üyelik Kümesi	Üyelik Derecesi
Anlama Anlatma Yeteneği	16	Çok Önemli	%40
Genel Görünüş	8	Zayıf	%57,1428
Psikolojik Yapı	12	Önemli	%66,67
Ailevi ve Sosyal Durum	7	Zayıf	%85,7143
Bilimsel Yeterlilik	20	Mükemmel	%100

Kriterlere ait kural tabanı ve veri tabanı

Kural tabanı, kriterlere verilecek girişlerin karşılaştırılmaları sonucu oluşacak bulanık çıkışların belirlenmesi amacıyla oluşturulmuş kuralların dizisidir. Bu kural tabanı uzman denetiminde hazırlanmış ve beş adet dilsel çıkış bilgisi kullanılmıştır. Bunlar ; Denk, zayıf, oldukça güçlü, çok güçlü, mükemmel'dir. Aşağıda kriterler için oluşturulan kural tabanına ait çizelge görülmektedir.

Çizelge 3.25. Kriterlere ait bulanık girişlerin karşılaştırılmalarına ait kural tabanı

Kriter i \ Kriter j	Önemsiz	Zayıf	Önemli	Çok Önemli	Mükemmel
Önemsiz	Denk	1/Zayıf	1/O.Güçlü	1/Ç.Güçlü	1/Mükemmel
Zayıf	Zayıf	Denk	1/Zayıf	1/O.Güçlü	1/Ç.Güçlü
Önemli	O.Güçlü	Zayıf	Denk	1/Zayıf	1/O.Güçlü
Çok Önemli	Ç. Güçlü	O.Güçlü	Zayıf	Denk	1/Zayıf
Mükemmel	Mükemmel	Ç. Güçlü	O.Güçlü	Zayıf	Denk

Bir bulanık denetim kuralı “EĞER $x=A$ ve $y=B$ ise O HALDE $z=C$ ” bulanık içerme işlevi ile gösterilen A ve B U uzayında, C ise V uzayında tanımlanmış bulanık kümelerdir. Yukarıdaki çizelgeye göre girişler ve çıkışlar arasında aşağıdaki gibi bir bağıntı kurulabilir.

if Kriter_i = Önemsiz ve Kriter_j = Önemsiz ise Sonuç = Denk,

if Kriter_i = Önemsiz ve Kriter_j = Zayıf ise Sonuç = 1/Zayıf,

if Kriter_i = Önemsiz ve Kriter_j = Önemli ise Sonuç = 1/O.Güçlü,

if Kriter_i = Önemsiz ve Kriter_j = Çok Önemli ise Sonuç = 1/Ç.Güçlü,

if Kriter_i = Önemsiz ve Kriter_j = Mükemmel ise Sonuç = 1/Mükemmel,

if Kriter_i = Zayıf ve Kriter_j = Önemsiz ise Sonuç = Zayıf,

if Kriter_i = Zayıf ve Kriter_j = Zayıf ise Sonuç = Denk,

if Kriter_i = Zayıf ve Kriter_j = Önemli ise Sonuç = 1/Zayıf,

if $Kriter_i = Zayıf$ ve $Kriter_j = Çok \text{ Önemli}$ ise $Sonuç = 1/O.Güçlü$,

if $Kriter_i = Zayıf$ ve $Kriter_j = Mükemmel$ ise $Sonuç = 1/Ç.Güçlü$,

if $Kriter_i = Önemli$ ve $Kriter_j = Önemsiz$ ise $Sonuç = O.Güçlü$,

if $Kriter_i = Önemli$ ve $Kriter_j = Zayıf$ ise $Sonuç = Zayıf$,

if $Kriter_i = Önemli$ ve $Kriter_j = Önemli$ ise $Sonuç = Denk$,

if $Kriter_i = Önemli$ ve $Kriter_j = Çok \text{ Önemli}$ ise $Sonuç = 1/Zayıf$,

if $Kriter_i = Önemli$ ve $Kriter_j = Mükemmel$ ise $Sonuç = 1/O.Güçlü$,

if $Kriter_i = Çok \text{ Önemli}$ ve $Kriter_j = Önemsiz$ ise $Sonuç = Ç. Güçlü$,

if $Kriter_i = Çok \text{ Önemli}$ ve $Kriter_j = Zayıf$ ise $Sonuç = O.Güçlü$,

if $Kriter_i = Çok \text{ Önemli}$ ve $Kriter_j = Önemli$ ise $Sonuç = Zayıf$,

if $Kriter_i = Çok \text{ Önemli}$ ve $Kriter_j = Çok \text{ Önemli}$ ise $Sonuç = Denk$,

if $Kriter_i = Çok \text{ Önemli}$ ve $Kriter_j = Mükemmel$ ise $Sonuç = 1/Zayıf$,

if $Kriter_i = Mükemmel$ ve $Kriter_j = Önemsiz$ ise $Sonuç = Mükemmel$,

if $Kriter_i = Mükemmel$ ve $Kriter_j = Zayıf$ ise $Sonuç = Ç. Güçlü$,

if $Kriter_i = Mükemmel$ ve $Kriter_j = Önemli$ ise $Sonuç = O.Güçlü$,

if $Kriter_i = Mükemmel$ ve $Kriter_j = Çok \text{ Önemli}$ ise $Sonuç = Zayıf$,

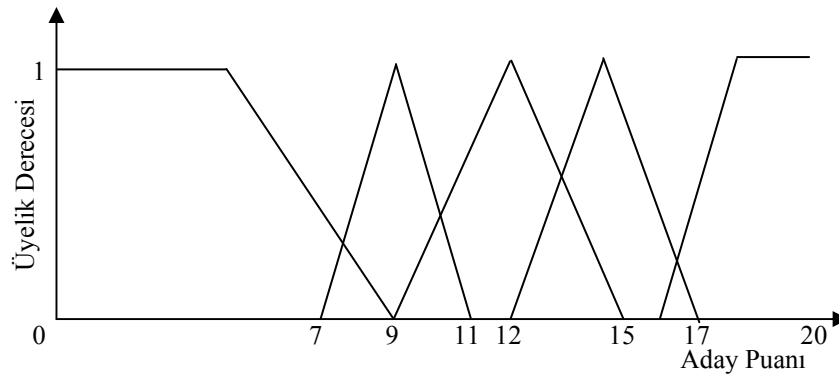
if $Kriter_i = Mükemmel$ ve $Kriter_j = Mükemmel$ ise $Sonuç = Denk$,

Çizelge 3.26. Kriterlere ait bulanık girişlerin karşılaştırılması

	AAY	GG	PY	ASD	BY
AAY	Denk	O.Güçlü	Zayıf	O.Güçlü	1/Zayıf
GG	O.Güçlü	Denk	1/Zayıf	Denk	Ç.Güçlü
PY	1/Zayıf	Zayıf	Denk	Zayıf	1/O.Güçlü
ASD	1/O.Güçlü	Denk	1/Zayıf	Denk	1/Ç.Güçlü
BY	Zayıf	Ç.Güçlü	O.Güçlü	Ç.Güçlü	Denk

Adaylar için bulanıklaştırma birimi

0-20 arası kriter puanlarına göre Şekil 3.3'deki üyelikler oluşturulmuştur. Üçgen şeklindeki üyelik kümelerinin üyelik derecelerinin belirlenmesinde aşağıdaki eşitlik kullanılmıştır. Bu üyelikler ve fonksiyonlar kullanılarak adayların alacağı değerlere göre hangi üyelik kümesinin üyesi olacağı Çizelge 3.27'de görülebilir.



Şekil 3.3. Aday puanlarına ait üçgen üyelik kümeleri ve üyelik fonksiyonu dereceleri

Çizelge 3.27. Aday puanlarına ait üyelik fonksiyonlarının oluşturulması

Üyelik Kümeleri	Fonksiyon	Değer Aralığı
KÖTÜ	$\mu(x) = 1$	$x \leq 4,5$
	$\mu(x) = (9-x) / (9-4,5)$	$4,5 \leq x \leq 9$
GEÇER	$\mu(x) = (x-7) / (9-7)$	$7 \leq x \leq 9$
	$\mu(x) = (11-x) / (11-9)$	$9 \leq x \leq 11$
ORTA	$\mu(x) = (x-9) / (12-9)$	$9 \leq x \leq 12$
	$\mu(x) = (15-x) / (15-12)$	$12 \leq x \leq 15$
İYİ	$\mu(x) = (x-12) / (14,5-12)$	$12 \leq x \leq 14,5$
	$\mu(x) = (17-x) / (17-14,5)$	$14,5 \leq x \leq 17$
ÇOK İYİ	$\mu(x) = (x-16) / (18-16)$	$16 \leq x \leq 18$
	$\mu(x) = 1$	$x \geq 18$

Çizelge 3.28. Karar vericinin verdiği puanlara göre adayların aitlik kümeleri

Aday Kriter	Aday 1	Aday 2	Aday 3	Aday 4	Aday 5
AAY	Orta	Geçer	Mükemmel	İyi	Orta
GG	Mükemmel	Orta	Orta	Mükemmel	Mükemmel
PY	Mükemmel	Mükemmel	İyi	Orta	Mükemmel
ASD	Mükemmel	İyi	İyi	Mükemmel	Mükemmel
BY	Orta	Mükemmel	Geçer	Mükemmel	İyi

Adaylara ait kural tabanı ve veri tabanı

Kural tabanı, adaylara verilecek girişlerin karşılaştırılmaları sonucu oluşacak bulanık çıkışların belirlenmesi amacıyla oluşturulmuş kuralların dizisidir. Bu kural tabanı uzman denetiminde hazırlanmış ve beş adet dilsel çıkış bilgisi kullanılmıştır. Bunlar ; Denk, zayıf, oldukça güçlü, çok güçlü, mükemmel'dir. Aşağıda adaylar için oluşturulan kural tabanına ait çizelge görülmektedir.

Çizelge 3.29. Adaylara ait bulanık girişlerin karşılaştırılmalarına ait kural tabanı

Aday j \ Aday i	Kötü	Geçer	Orta	İyi	Çok İyi
Kötü	Denk	1/Zayıf	1/O.Güçlü	1/Ç.Güçlü	1/Mükemmel
Geçer	Zayıf	Denk	1/Zayıf	1/O.Güçlü	1/Ç.Güçlü
Orta	O.Güçlü	Zayıf	Denk	1/Zayıf	1/O.Güçlü
İyi	Ç. Güçlü	O.Güçlü	Zayıf	Denk	1/Zayıf
Çok İyi	Mükemmel	Ç. Güçlü	O.Güçlü	Zayıf	Denk

Bir bulanık denetim kuralı “EĞER $x=A$ ve $y=B$ ise O HALDE $z=C$ ” bulanık içermeye işlevi ile gösterilen A ve B U uzayında, C ise V uzayında tanımlanmış bulanık kümelerdir. Yukarıda ki çizelgeye göre girişler ve çıkışlar arasında aşağıdaki gibi bir bağıntı kurulabilir.

if $A_{day_i} = \text{Kötü}$ ve $A_{day_j} = \text{Kötü}$ ise Sonuç = Denk,

if $A_{day_i} = \text{Kötü}$ ve $A_{day_j} = \text{Geçer}$ ise Sonuç = 1/Zayıf,

if $A_{day_i} = \text{Kötü}$ ve $A_{day_j} = \text{Orta}$ ise Sonuç = 1/O.Güçlü,

if $A_{day_i} = \text{Kötü}$ ve $A_{day_j} = \text{İyi}$ ise Sonuç = 1/Ç.Güçlü,

if $A_{day_i} = \text{Kötü}$ ve $A_{day_j} = \text{Çok İyi}$ ise Sonuç = 1/Mükemmel,

if $A_{day_i} = \text{Geçer}$ ve $A_{day_j} = \text{Kötü}$ ise Sonuç = Zayıf,

if $A_{day_i} = \text{Geçer}$ ve $A_{day_j} = \text{Geçer}$ ise Sonuç = Denk,

if $A_{day_i} = \text{Geçer}$ ve $A_{day_j} = \text{Orta}$ ise Sonuç = 1/Zayıf,

if $A_{day_i} = \text{Geçer}$ ve $A_{day_j} = \text{İyi}$ ise Sonuç = 1/O.Güçlü,

if Aday_i = Geçer ve Aday_j = Çok İyi ise Sonuç = 1/Ç.Güçlü,

if Aday_i = Orta ve Aday_j = Kötü ise Sonuç = O.Güçlü,

if Aday_i = Orta ve Aday_j = Geçer ise Sonuç = Zayıf,

if Aday_i = Orta ve Aday_j = Orta ise Sonuç = Denk,

if Aday_i = Orta ve Aday_j = İyi ise Sonuç = 1/Zayıf,

if Aday_i = Orta ve Aday_j = Çok İyi ise Sonuç = 1/O.Güçlü,

if Aday_i = İyi ve Aday_j = Kötü ise Sonuç = Ç. Güçlü,

if Aday_i = İyi ve Aday_j = Geçer ise Sonuç = O.Güçlü,

if Aday_i = İyi ve Aday_j = Orta ise Sonuç = Zayıf,

if Aday_i = İyi ve Aday_j = İyi ise Sonuç = Denk,

if Aday_i = İyi ve Aday_j = Çok İyi ise Sonuç = 1/Zayıf,

if Aday_i = Çok İyi ve Aday_j = Kötü ise Sonuç = Mükemmel,

if Aday_i = Çok İyi ve Aday_j = Geçer ise Sonuç = Ç. Güçlü,

if Aday_i = Çok İyi ve Aday_j = Orta ise Sonuç = O.Güçlü,

if Aday_i = Çok İyi ve Aday_j = İyi ise Sonuç = Zayıf,

if Aday_i = Çok İyi ve Aday_j = Çok İyi ise Sonuç = Denk,

Çizelge 3.30. Anlama anlatma yeteneği kriteri için adaylara ait bulanık girişlerin karşılaştırılması

AAY	Aday 1	Aday 2	Aday 3	Aday 4	Aday 5
Aday 1	Denk	Zayıf	1/O.Güçlü	1/Zayıf	Denk
Aday 2	1/Zayıf	Denk	1/Ç.Güçlü	1/O.Güçlü	1/Zayıf
Aday 3	O.Güçlü	Ç.Güçlü	Denk	Zayıf	O.Güçlü
Aday 4	Zayıf	O.Güçlü	1/Zayıf	Denk	Zayıf
Aday 5	Denk	Zayıf	1/O.Güçlü	1/Zayıf	Denk

Çizelge 3.31. Genel görünüş kriteri için adaylara ait bulanık girişlerin karşılaştırılması

GG	Aday 1	Aday 2	Aday 3	Aday 4	Aday 5
Aday 1	Denk	O.Güçlü	O.Güçlü	Denk	Denk
Aday 2	1/O.Güçlü	Denk	Denk	1/O.Güçlü	1/O.Güçlü
Aday 3	1/O.Güçlü	Denk	Denk	1/O.Güçlü	1/O.Güçlü
Aday 4	Denk	O.Güçlü	O.Güçlü	Denk	Denk
Aday 5	Denk	O.Güçlü	O.Güçlü	Denk	Denk

Çizelge 3.32. Psikolojik yapı kriteri için adaylara ait bulanık girişlerin karşılaştırılması

PY	Aday 1	Aday 2	Aday 3	Aday 4	Aday 5
Aday 1	Denk	Denk	Zayıf	O.Güçlü	Denk
Aday 2	Denk	Denk	Zayıf	O.Güçlü	Denk
Aday 3	1/Zayıf	1/Zayıf	Denk	Zayıf	1/Zayıf
Aday 4	1/O.Güçlü	1/O.Güçlü	1/Zayıf	Denk	1/O.Güçlü
Aday 5	Denk	Denk	Zayıf	O.Güçlü	Denk

Çizelge 3.33. Ailevi ve sosyal durum kriteri için adaylara ait bulanık girişlerin karşılaştırılması

ASD	Aday 1	Aday 2	Aday 3	Aday 4	Aday 5
Aday 1	Denk	Zayıf	Zayıf	Denk	Denk
Aday 2	1/Zayıf	Denk	Denk	1/Zayıf	1/Zayıf
Aday 3	1/Zayıf	Denk	Denk	1/Zayıf	1/Zayıf
Aday 4	Denk	Zayıf	Zayıf	Denk	Denk
Aday 5	Denk	Zayıf	Zayıf	Denk	Denk

Çizelge 3.34. Bilimsel yeterlilik kriteri için adaylara ait bulanık girişlerin karşılaştırılması

BY	Aday 1	Aday 2	Aday 3	Aday 4	Aday 5
Aday 1	Denk	1/O.Güçlü	Zayıf	1/O.Güçlü	1/Zayıf
Aday 2	O.Güçlü	Denk	Ç.Güçlü	Denk	Zayıf
Aday 3	1/Zayıf	1/Ç.Güçlü	Denk	1/Ç.Güçlü	1/O.Güçlü
Aday 4	O.Güçlü	Denk	Ç.Güçlü	Denk	Zayıf
Aday 5	Zayıf	1/Zayıf	O.Güçlü	1/Zayıf	Denk

Hem kriterler için hem de adaylar için karar verici tarafından verilen puanların ait oldukları küme ve aitlik dereceleri belirlendikten sonra hem kriterlerin hem de adayların karşılaştırılması yukarıda belirtilen kural tabanlarına uygun olarak yapılmıştır.

Karşılaştırma sonucu çıkan dilsel değişkenlere karşılık gelen bulanık üçgensel sayıların oluşturduğu matrise boyut analizi uygulanarak her bir kriterin tüm karşılaştırma işlemi içerisindeki ağırlığı belirlenir. Böylece her bir kriter için tek bir bulanık üçgensel değer elde edilir. Elde edilen kriter ağırlıklarına sup-min bileşke bağıntısı uygulanarak kriterlerin birbirlerine minimum üstünlükleri belirlenir ve daha sonrada bu değerler normalize edilerek nihai ağırlıkları oluşur. Aynı işlemler adaylar içinde uygulanarak her bir adayın her bir kriterden nihai ağırlığı oluşur. Elde edilen kriter ağırlıkları ve aday ağırlıkları birleştirilmiş karar matrisi üzerinde çarpılarak her bir adayın toplam puanı bulunur.

Aşağıdaki çizelgede karar vericinin hem kriterleri hem de adayları karşılaştırmak için kullanacağı dilsel ifadelerin tanımı ve önem derecelerini gösteren skala verilmiştir. Bu skalaya göre hem kriterlerin hem de adayların karşılaştırılmaları sonucu oluşan dilsel değişkenlere karşılık gelen bulanık üçgensel sayılar belirlenir.

Çizelge 3.35. Dilsel ifade skalası

Dilsel İfade	Mükemmel (Absolute)	Çok Güçlü (Very Strong)	Oldukça Güçlü (Fairly Strong)	Zayıf (Weak)	Denk (Equal)
Bulanık Sayı	(3.5,4.25,5)	(2.5,3.25,4)	(1.5,2.25,3)	(0.5,1.25,2)	(1, 1, 1)

Dilsel ifade skalasının kullanıldığı bir başka çalışma olan Özkan Bali'ye ait yüksek lisans tezinde, skalada kullanılan bulanık sayıların literatürde geçen bulanık sayı olma durumunu sağlamamış olabileceği görülmüştür [8].

Aşağıda karar vericilerin dilsel ifade skalası ile yaptıkları kriter karşılaştırmasının bulanık değerlendirme matrisi verilmiştir.

Çizelge 3.36. Dilsel ifade skalasına göre kriterlerin bulanık değerlendirilmesi matrisi

Kriterler	AAY	GG	PY	ASD	BY
AAY	(1, 1, 1)	(1.5,2.25,3)	(0.5, 1.25, 2)	(1.5, 2.25, 3)	(0.5, 0.8, 2)
GG	(0.33,0.44,0.66)	(1, 1, 1)	(0.5, 0.8, 2)	(1, 1, 1)	(0.25, 0.31, 0.4)
PY	(0.5, 0.8, 2)	(0.5,1.25,2)	(1, 1, 1)	(0.5, 1.25, 2)	(0.33,0.44,0.66)
ASD	(0.33,0.44,0.66)	(1, 1, 1)	(0.5, 0.8, 2)	(1, 1, 1)	(0.25, 0.31, 0.4)
BY	(0.5, 1.25, 2)	(2.5,3.25,4)	(1.5, 2.25, 3)	(2.5, 3.25, 4)	(1, 1, 1)

ADIM 1 : Bulanık değerlendirme matrisinden ikili karşılaştırmanın sentetik boyut değerleri aşağıdaki gibi elde edilmiştir.

$$\begin{aligned}
S_{AAY} &= (5 ; 7,55 ; 11) & * & (1/42,78 ; 1/30,39 ; 1/21,99) = (0,1169 ; 0,2484 ; 0,5002) \\
S_{GG} &= (3,08 ; 3,55 ; 5,06) & * & (1/42,78 ; 1/30,39 ; 1/21,99) = (0,0720 ; 0,1168 ; 0,2301) \\
S_{PY} &= (2,83 ; 4,74 ; 7,66) & * & (1/42,78 ; 1/30,39 ; 1/21,99) = (0,0661 ; 0,1560 ; 0,3483) \\
S_{ASD} &= (3,08 ; 3,55 ; 5,06) & * & (1/42,78 ; 1/30,39 ; 1/21,99) = (0,0720 ; 0,1168 ; 0,2301) \\
S_{BY} &= (8 ; 11 ; 14) & * & (1/42,78 ; 1/30,39 ; 1/21,99) = (0,1870 ; 0,3620 ; 0,6366)
\end{aligned}$$

ADIM 2 : $M_2 = (l_2, m_2, u_2) \geq M_1 = (l_1, m_1, u_1)$ ifadesinin olabilirlik derecesi aşağıdaki fonksiyona göre tanımlanır.

$$V(M_2 \geq M_1) = \mu_{M_2}(d) = \begin{cases} 1, & \text{eğer } m_2 \geq m_1 \\ 0, & \text{eğer } l_1 \geq u_2 \\ \frac{l_1 - u_2}{(m_2 - u_2) - (m_1 - l_1)}, & \text{değilse} \end{cases}$$

$$\begin{aligned}
V(S_{AAY} \geq S_{GG}) &= 1 & V(S_{AAY} \geq S_{PY}) &= 1 \\
V(S_{AAY} \geq S_{ASD}) &= 1 & V(S_{AAY} \geq S_{BY}) &= 0,7340 \\
V(S_{GG} \geq S_{AAY}) &= 0,4624 & V(S_{GG} \geq S_{PY}) &= 0,8072 \\
V(S_{GG} \geq S_{ASD}) &= 1 & V(S_{GG} \geq S_{BY}) &= 0,1495 \\
V(S_{PY} \geq S_{AAY}) &= 0,7145 & V(S_{PY} \geq S_{GG}) &= 1 \\
V(S_{PY} \geq S_{ASD}) &= 1 & V(S_{PY} \geq S_{BY}) &= 0,4392 \\
V(S_{ASD} \geq S_{AAY}) &= 0,4624 & V(S_{ASD} \geq S_{GG}) &= 1 \\
V(S_{ASD} \geq S_{PY}) &= 0,8072 & V(S_{ASD} \geq S_{BY}) &= 0,1495
\end{aligned}$$

$$V(S_{BY} \geq S_{GG}) = 1$$

$$V(S_{BY} \geq S_{AAV}) = 1$$

$$V(S_{BY} \geq S_{ASD}) = 1$$

$$V(S_{BY} \geq S_{PY}) = 1$$

ADIM 3 :

$$V(M \geq M_1, M_2, \dots, M_k) = V[(M \geq M_1) \text{ ve } (M \geq M_2) \text{ ve } \dots \text{ ve } (M \geq M_k)] \text{ min}$$

$$V(M \geq M_i), i=1, 2, \dots, k$$

varsayılan

$$d(A_i) = \min V(S_i \geq S_k)$$

$k = 1, 2, \dots, n$ için $k \neq i$. ağırlık vektörü aşağıdaki gibi verilebilir.

$$W' = (d(A_1), d(A_2), \dots, d(A_n))^T$$

A_i ($i=1, 2, \dots, n$) olmak üzere A_i , n

elementlidir.

Bu formüle göre $W' = (0,7340 ; 0,1495 ; 0,4392 ; 0,1495 ; 1)$ olarak bulunur.

ADIM 4 : Normalizasyon işlemi ile normalize edilen ağırlık vektörü aşağıdaki gibi elde edilir. Bu işlem için minimum karşılaştırma değerleri toplanır ve her bir değer toplama bölünür. Daha sonra kriterlere başlangıçta verilen ham puanlar toplanır ve her bir kriterin aldığı puan bu toplama bölünür. İki değer toplandıktan sonra elde edilen değerlerden en büyüğü belirlenerek tüm puanlar bu değere bölünür. Sonuçta en büyüğü her zaman 1 olan normalize edilmiş puanlar oluşur.

$$W = (0,7630 ; 0,2597 ; 0,5086 ; 0,2377 ; 1)$$

Her bir kriter için adayların karşılaştırılmasında da aynı işlemler gerçekleştirilecektir. Bu nedenle işlemin basamakları ve hesaplamalar diğer kriterler için verilmeden sadece bulanık değerlendirme matrisleri ve kriter ağırlıkları gösterilmiştir.

Anlama ve anlatma yeteneği

Aşağıda Anlama ve Anlatma Yeteneği kriterine göre bulanık değerlendirme matrisi ve hesaplamalar sonucu kriter ağırlıkları verilmiştir.

Çizelge 3.37. Anlama ve anlatma yeteneği kriterine göre bulanık değerlendirme matrisi

Adaylar	A1	A2	A3	A4	A5
A1	(1,1,1)	(0.5,1.15,2)	(0.33,0.44,0.66)	(0.5,1.25,2)	(1,1,1)
A2	(0.5,0.8,2)	(1,1,1)	(0.25,0.31,0.4)	(0.33,0.44,0.66)	(0.5,0.8,2)
A3	(1.5,2.25,3)	(2.5,3.25,4)	(1,1,1)	(0.5,1.25,2)	(1.5,2.25,3)
A4	(0.5,1.25,2)	(1.5,2.25,3)	(0.5,0.8,2)	(1,1,1)	(0.5,1.25,2)
A5	(1,1,1)	(0.5,1.25,2)	(0.33,0.44,0.66)	(0.5,0.8,2)	(1,1,1)

$$W = (0,5638 ; 0,4407 ; 1 ; 0,7351 ; 0,54)$$

Genel görünüş

Çizelge 3.38 ‘de Genel Görünüş kriterine göre bulanık değerlendirme matrisi verilmiştir.

Çizelge 3.38. Genel görünüş kriterine göre bulanık değerlendirme matrisi

Adaylar	A1	A2	A3	A4	A5
A1	(1,1,1)	(1.5,2.25,3)	(1.5,2.25,3)	(1,1,1)	(1,1,1)
A2	(0.33,0.44,0.66)	(1,1,1)	(1,1,1)	(0.33,0.44,0.66)	(0.33,0.44,0.66)
A3	(0.33,0.44,0.66)	(1,1,1)	(1,1,1)	(0.33,0.44,0.66)	(0.33,0.44,0.66)
A4	(1,1,1)	(1.5,2.25,3)	(1.5,2.25,3)	(1,1,1)	(1,1,1)
A5	(1,1,1)	(1.5,2.25,3)	(1.5,2.25,3)	(1,1,1)	(1,1,1)

$$W = (0,9357 ; 0,2786 ; 0,2571 ; 1 ; 0,9571)$$

Psikolojik yapısı

Aşağıda Psikolojik Yapısı kriterine göre bulanık değerlendirme matrisi ve hesaplamalar sonucu kriter ağırlıkları verilmiştir.

Çizelge 3.39. Psikolojik yapısı kriterine göre bulanık değerlendirme matrisi

ADAYLAR	A1	A2	A3	A4	A5
A1	(1,1,1)	(1,1,1)	(0.5,1.25,2)	(1.5,2.25,3)	(1,1,1)
A2	(1,1,1)	(1,1,1)	(0.5,1.25,2)	(1.5,2.25,3)	(1,1,1)
A3	(0.5,0.8,2)	(0.5,0.8,2)	(1,1,1)	(0.5,1.25,2)	(0.5,0.8,2)
A4	(0.33,0.44,0.66)	(0.33,0.44,0.66)	(0.5,0.8,2)	(1,1,1)	(0.33,0.44,0.66)
A5	(1,1,1)	(1,1,1)	(0.5,1.25,2)	(1.5,2.25,3)	(1,1,1)

$$W = (1 ; 0,9233 ; 0,8094 ; 0,5381 ; 0,9233)$$

Aile ve sosyal durumu

Aşağıda Aile ve Sosyal Durum kriterine göre bulanık değerlendirme matrisi ve hesaplamalar sonucu kriter ağırlıkları verilmiştir.

Çizelge 3.40. Aile ve sosyal durum kriterine göre bulanık değerlendirme matrisi

ADAYLAR	A1	A2	A3	A4	A5
A1	(1,1,1)	(0.5,1.25,2)	(0.5,1.25,2)	(1,1,1)	(1,1,1)
A2	(0.5,0.8,2)	(1,1,1)	(1,1,1)	(0.5,0.8,2)	(0.5,0.8,2)
A3	(0.5,0.8,2)	(1,1,1)	(1,1,1)	(0.5,0.8,2)	(0.5,0.8,2)
A4	(1,1,1)	(0.5,1.25,2)	(0.5,1.25,2)	(1,1,1)	(1,1,1)
A5	(1,1,1)	(0.5,1.25,2)	(0.5,1.25,2)	(1,1,1)	(1,1,1)

$$W = (1 ; 0,8381 ; 0,8381 ; 1 ; 0,9744)$$

Bilimsel yeterliliği

Aşağıda Bilimsel Yeterliliği kriterine göre bulanık değerlendirme matrisi ve hesaplamalar sonucu kriter ağırlıkları verilmiştir.

Çizelge 3.41. Bilimsel yeterliliği kriterine göre bulanık değerlendirme matrisi

ADAYLAR	A1	A2	A3	A4	A5
A1	(1,1,1)	(0.33,0.44,0.66)	(0.5,1.25,2)	(0.33,0.44,0.66)	(0.5,0.8,2)
A2	(1.5,2.25,3)	(1,1,1)	(2.5,3.25,4)	(1,1,1)	(0.5,1.25,2)
A3	(0.5,0.8,2)	(0.25,0.31,0.4)	(1,1,1)	(0.25,0.31,0.4)	(0.33,0.44,0.66)
A4	(1.5,2.25,3)	(1,1,1)	(2.5,3.25,4)	(1,1,1)	(0.5,1.25,2)
A5	(0.5,1.25,2)	(0.5,0.8,2)	(1.5,2.25,3)	(0.5,0.8,2)	(1,1,1)

$$W = (0,5553 ; 1 ; 0,3676 ; 0,9745 ; 0,8281)$$

Tüm adaylar için hesaplanan tüm kriter ağırlıkları Çizelge 3.42 ‘de gösterilmiştir.

Çizelge 3.42. Tüm adaylar için hesaplanan tüm kriter ağırlıkları

	Kriterler				
	Anlama ve Anlatma Yeteneği	Genel Görünüş	Psikolojik Yapısı	Aile ve sosyal Durum	Bilimsel Yeterlilik
Ağırlık	0,7630	0,2597	0,5086	0,2377	1
Aday 1	0,5638	0,9357	1	1	0,5553
Aday 2	0,4407	0,2786	0,9233	0,8381	1
Aday 3	1	0,2571	0,8094	0,8381	0,3676
Aday 4	0,7351	1	0,5381	1	0,9745
Aday 5	0,54	0,9571	0,9233	0,9744	0,8281

Adayların kriterlere göre aldığı toplam puanlar aşağıda sıralanmıştır.

$$W_1 = 0,7630 * 0,5638 + 0,2597 * 0,9357 + 0,5086 * 1 + 0,2377 * 1 + 1 * 0,5553$$

$$W_2 = 0,7630 * 0,4407 + 0,2597 * 0,2786 + 0,5086 * 0,9233 + 0,2377 * 0,8381 + 1 * 1$$

$$W_3 = 0,7630 * 1 + 0,2597 * 0,2571 + 0,5086 * 0,8094 + 0,2377 * 0,8381 + 1 * 0,3676$$

$$W_4 = 0,7630 * 0,7351 + 0,2597 * 1 + 0,5086 * 0,5381 + 0,2377 * 1 + 1 * 0,9745$$

$$W_5 = 0,7630 * 0,54 + 0,2597 * 0,9571 + 0,5086 * 0,9233 + 0,2377 * 0,9744 + 1 * 0,8281$$

Elde edilen değerlerden en büyüğü belirlenerek tüm puanlar bu değere bölünür. Sonuçta en büyüğü her zaman 1 olan normalize edilmiş puanlar oluşur.

$$W_1 = 0,8565 \quad W_2 = 0,901 \quad W_3 = 0,7842 \quad W_4 = 1 \quad W_5 = 0,9496$$

Adayların ağırlıklarına bakıldığında aday sıralaması ;

4, 5, 2, 1, 3 şeklinde olmaktadır. Bulanık AHS yöntemi ile yaptığımız değerlendirme sonucu 5 aday arasından en yüksek dereceye sahip 2 aday olan 4. ve 5. adayın seçilmesi uygun olacaktır.

4. UYGULAMA

Daha öncede bahsedildiği gibi uygulama aşamasında istatistiksel olarak daha gerçekçi sonuçlar alabilmek amacı ile popülasyondan alınan örnekteki aday sayısı arttırılarak 2000 olarak belirlenmiş ve 2000 adayın Klasik yöntem, AHS yöntemi ve BAHS yöntemlerine göre değerlendirilme işlemlerinde kolaylık sağlamak ve işlemi tekrar tekrar simüle edebilmek amacıyla da bir simülasyon geliştirilmiştir. Simülasyon geliştirmek amacıyla matematiksel hesaplama işlemlerinde gösterdiği performans ve ondalıklı sayılar üzerindeki işlem kabiliyetinden dolayı Digital Fortran 5.0 yazılım geliştirme programı tercih edilmiştir. Ayrıca görsel özelliklerinden dolayı BAHS yöntemine ait algoritmayı simüle etmek amacıyla Delphi'de bir simülasyon programı geliştirilmiştir.

Hazırlanan simülasyonda, öncelikle kriterlere ait, karar vericilerin 0-20 aralığında belirlediği puanlar girilmektedir. Girilen bu kriter puanları önce AHS için 1-9 ölçeğine çevrilmekte daha sonrada AHS ve BAHS algoritmalarında işlenerek kriter ağırlıkları hesaplanmaktadır. Bu hesaplama işleminden sonra her bir aday için tesadüfi olarak 0-20 aralığında 5 adet sayı üretilmektedir. 5 adet değer üretmesinin sebebi aday değerlendirme ölçütü olarak 5 adet kriterin olmasıdır. Üretilen değerlerin 0-20 aralığında olması ise 5 adet kriterden elde edilecek maksimum değer 100 olmasıdır. Yani toplamda her bir aday için Klasik Yöntem'e uygun olarak 0-100 aralığında bir puan oluşmaktadır. Daha sonra Klasik Yöntem'e uygun olarak üretilen bu sayılar AHS yönteminin aralığına dönüştürülerek ortak değerler üzerinde işlem imkanı sağlanmaktadır.

Her bir aday için her üç yöntemde de puanlar oluşturulduktan sonra, Klasik Yöntem'in, AHS'nin ve BAHS'nin kendi algoritmalarına uygun olarak puanlar işlenmekte ve her bir aday için Klasik Yöntem'de, AHS yönteminde ve BAHS yönteminde birer puan oluşmaktadır. Ayrıca üç yöntemde göre elde edilen değerler üzerinde ortak değerlendirme yapabilmek amacıyla puanlar Z değerlerine çevrilerek Rank'ları alınmaktadır.

Sonuçta her bir aday için elde edilen her bir y nteme g re ayrı ayrı puanları, bu puanların Z deęerleri ve Rankları “Orta.txt” isimli text tipinde bir dosyada saklanmaktadır.

EK-1’de kodları verilen sim lasyon programı iřletilerek elde edilen sonu lar SPSS 9.05 ve M N TAB 13 istatistiksel veri analizi paket programları yardımıyla ařaęıda bahsedilecek analizlerle deęerlendirilerek 3 y ntemin birbirinden ne derece farklı se im yaptığı, tespit edilmektedir.

Sim lasyondan elde edilen    y nteme ait deęerleri deęerlendirme amacıyla  ncelikle SPSS 9.05’de Sınıf i i Korelasyon analizi kullanılmıřtır.

Sınıf i i korelasyon analizi (Intraclass Korelasyon Analizi (ICC))

Sınıf i i korelasyon katsayısı olarak adlandırılır ve iki veya daha fazla  l me y nteminin karřılařtırılmasında kullanılan pop ler y ntemlerden birisidir. Sınıf i i (intraclass) s zc ę , aynı deęiřkenin tekrarlı  l mlerini analiz etmek gibi bir kavram i erirken, sınıflar arası (interclass) s zc ę  iki farklı deęiřkenin analizi i in uygun bir kavramdır (korelasyon katsayısı gibi). Korelasyon katsayısı, daha  ok farklı  l m birimiyle  l lm ř iki veri setinin iliřki miktarının  l m nde kullanılır. Bundan dolayı, sınıflar arası analize uygundur ve veri iki farklı  l m birimiyle  l ld ę  i in ortalamalardaki deęiřikliklerle ilgilenmez. S rekli deęiřkenler i in  l m uyumu analizlerinde arařtırmacılar tarafından en  ok yapılan hatalardan birisi bu uyumu test etmek i in eřleřtirilmiř (paired) t testinin veya Pearson korelasyon katsayısının kullanılmasıdır. Genellikle  l m uyumu y ksek olduęunda bu iki test ile de uyumun iyi olduęuna iliřkin bir bilgi edinilebilmektedir. Ancak bu iki testin  l m uyumu analizlerinde kullanılması hatalıdır. Sınıf i i korelasyon katsayısı karmařık tasarımlar i in geliřtirilmiřtir. Sınıf i i korelasyon katsayısının kabul edilebilir seviyeleri  izelge 4.1’deki gibi verilebilir [52].

Çizelge 4.1. Sınıf içi korelasyon katsayısının kabul edilebilir seviyeleri

Sınıf içi korelasyon değeri	Kabul edilebilir seviye
<0.70	Uyumsuz
0.70-0.84	İyi (kabul edilebilir)
0.84-0.94	Yüksek
0.94-1	Mükemmel

Bu korelasyon katsayısının kabul edilebilir seviyeleri verilerin özelliğine ve kullanıldığı alana göre farklılık göstermektedir. Örneğin, fizyolojik veriler üzerinde bir çalışma yapılıyor ise bu durumda sınıf içi korelasyon katsayısının 0.90'nın üzerinde olması kabul edilebilir seviyesinin “yüksek”, 0.80 ile 0.89 arasında olması “orta”, ve 0.80'nin altında olması “şüpheli” olarak nitelendirilir. Birden fazla değerlendirici arasında bir uyum olduğunu söyleyebilmek için sınıf içi korelasyon katsayısının en az 0.70 olması gerekmektedir. Sınıf içi korelasyon katsayısı 0.95 ile 1.00 arasında bir değer alıyorsa değerlendiriciler arasındaki uyumun “mükemmel” olduğunu, 0.85 ile 0.94 arasında bir değer alıyorsa değerlendiriciler arasındaki uyumun “yüksek” olduğunu, korelasyon değerinin 0.70'in altında olması durumunda değerlendiricilerin birbirleriyle hiçbir şekilde uyumlu olmadığını söyleyebiliriz.

ICC (sınıf içi korelasyon katsayısı);

$$\rho = \frac{\sigma_p^2}{\sigma_p^2 + \sigma_w^2}$$

ile tanımlanır [22].

Sınıf içi korelasyon analizini gerçekleştirmek için SPSS'de Analyz → Scale → Realibility seçeneği seçilir.

Sınıf içi korelasyon analizi sonuçları

Klasik Yöntem, AHS, BAHS yöntemlerinin 3'ünün rankları birbiriyle karşılaştırılmıştır.

Sınıf İçi Korelasyon Katsayısı = 0,9058

Yukarıda Klasik Yöntem, AHS yöntemi ve BAHS yöntemlerine ait Rank değerleri karşılaştırılmıştır. Sonuç olarak 3 rank değeri %90,58 oranında benzerdir denir. Bunun anlamı yapılan sıralama sonucunda adayların 3 yöneme ait dizilişleri birbirine %90,58 oranında benzemektedir.

Klasik Yöntem ile AHS yöntemlerinin rankları birbiriyle karşılaştırılmıştır.

Sınıf İçi Korelasyon Katsayısı = 0,9114

Yukarıda Klasik Yöntem ile AHS yöntemine ait Rank değerleri karşılaştırılmıştır. Sonuç olarak 2 rank değeri %91,14 oranında benzerdir denir. Bunun anlamı yapılan sıralama sonucunda adayların 2 yöneme ait dizilişleri birbirine %91,14 oranında benzemektedir.

Klasik Yöntem ile BAHS yöntemlerinin rankları birbiriyle karşılaştırılmıştır

Sınıf İçi Korelasyon Katsayısı = 0,8439

Yukarıda Klasik Yöntem ile BAHS yöntemine ait Rank değerleri karşılaştırılmıştır. Sonuç olarak 2 rank değeri %84,39 oranında benzerdir denir. Bunun anlamı yapılan sıralama sonucunda adayların 2 yöneme ait dizilişleri birbirine %84,39 oranında benzemektedir.

AHS yöntemi ile BAHS yöntemlerinin rankları birbiriyle karşılaştırılmıştır

Sınıf İçi Korelasyon Katsayısı = 0,9621

Yukarıda AHS yöntemi ile BAHS yöntemine ait Rank değerleri karşılaştırılmıştır. Sonuç olarak 2 rank değeri %96,21 oranında benzerdir denir. Bunun anlamı yapılan

sıralama sonucunda adayların 2 yönteme ait dizilişleri birbirine %96,21 oranında benzemektedir.

Sınıf içi korelasyon işleminin sonucu olarak her üç yöntemde birbiriyle büyük oranda benzerdir. Özellikle 2 ve 3 ncü yöntemler (ahs-bahs) birbiriyle daha yakın sonuçlar vermektedir.

Daha sonra başka bir analiz yöntemi olan Korelasyon Analizi uygulanmıştır.

İki yada daha çok değişken arasındaki ilişki düzeyini, yönünü ve önemliliğini belirlemeye yardım eden yönteme Korelasyon Analizi adı verilmektedir.

Minitab'de Korelasyon Analizi Stat → Basic Statistics → Correlation seçeneği seçilerek yapılır.

Korelasyon analizi

Klasik yöntem ile AHS yöntemlerinin Z değerleri arasındaki korelasyon katsayısı $r=0,919$ yani %91,9 düzeyinde önemli pozitif bir ilişki bulunmaktadır. Bu ilişki %5 yanılma %95 doğruluk ile istatistiki olarak önemlidir.

Klasik yöntem ile BAHS yöntemlerinin Z değerleri arasındaki korelasyon katsayısı $r=0,857$ yani %85,7 düzeyinde önemli pozitif bir ilişki bulunmaktadır. Bu ilişki %5 yanılma %95 doğruluk ile istatistiki olarak önemlidir.

AHS yöntemi ile BAHS yöntemlerinin Z değerleri arasındaki korelasyon katsayısı $r=0,965$ yani %96,5 düzeyinde önemli pozitif bir ilişki bulunmaktadır. Bu ilişki %5 yanılma %95 doğruluk ile istatistiki olarak önemlidir.

5. SONUÇ

Öncelikle Gazi Üniversitesi Fen Bilimleri Enstitüsüne 2004-2005 eğitim öğretim yılında yüksek lisans düzeyinde eğitim başvurusunda bulunan 619 adet adaya ait mülakat notları alınarak nasıl bir dağılım oluşturduğu belirlenmiştir. Yapılan değerlendirmede oluşan eğrinin tam bir normal dağılım oluşturmadığı ve belirli puanlarda yığılmalar olduğu gözlenmiştir. Bu değerlendirme hazırlanacak simülasyonda oluşturulacak random sayılar için kaynak olmuştur.

Klasik yöntem, AHS yöntemi ve BAHS yöntemlerine ait algoritmaları karşılaştırmak amacıyla oluşturulan simülasyonda önce klasik yönteme göre oluşturulan sayılar AHS yönteminin 1-9 skalasına dönüştürülmüş daha sonra her bir yöntemin algoritmalarında işlenerek neticede her üç yöntemde de adaylar için birer puan elde edilmiştir.

Elde edilen bu puanlar farklı skalalarda olduğundan ortak bir değerlendirme yapabilmek amacıyla ortalaması 1 ve varyansı 0 olan Z değerlerine çevrilmiştir. Elde edilen Z değerleri üzerinde adayların rankları hesaplanarak adayların elde ettikleri puanlara göre kaçınıcı sırada oldukları belirlenmiştir. Simülasyon bize her bir adaya ait her bir yöntemin kendi skalalarına uygun puanları, bu puanların Z değerlerini ve ranklarını oluşturmamızı sağlamıştır.

Simülasyondan elde ettiğimiz değerleri SPSS 9.05 ve Minitab 13 istatistiksel veri analizi paket programlarında çeşitli istatistiksel analizlerden geçirerek üç yöntemin 2000 bireylik bir örnekten seçeceği ilk 100 birey arasındaki benzerlik elde edilmiştir. Bu benzerlik üç yöntemin birbirine göre durumlarını ortaya koymaktadır.

Uygulama kısmında da belirtildiği gibi, yapılan Sınıf içi korelasyon testi sonucunda 3 yöntemin yaptıkları aday seçimi arasında %90,58'lik, Klasik Yöntem ile AHS yöntemi arasında %91,14'lük, Klasik Yöntem ile BAHS yöntemi arasında %84,39'luk ve AHS yöntemi ile BAHS yöntemi arasında %96,21'lik önemli bir benzerlik bulunmaktadır. Buradan da görüleceği gibi BAHS yöntemi diğer yöntemler ile daha

küçük bir benzerlik katsayısına sahiptir. Bundan dolayı BAHS ile AHS yöntemleri arasındaki benzerliğin daha önemli olduğunu söyleyebiliriz.

Sınıf içi korelasyon testi sonucu elde edilen veriler Çizelge 4.1’de verilen sınıf içi korelasyon katsayılarının seviyeleri ile mukayese edilmiş her 3 yönteminde kabul edilebilir seviye açısından kabul edilebilir olduğu, Klasik yöntemle BAHS yöntemi arasında ve Klasik yöntemle AHS yöntemi arasında yüksek, AHS yöntemi ile BAHS yöntemi arasında mükemmel seviyede bir uyum olduğu görülmüştür.

Daha sonra gerçekleştirilen Korelasyon testinde de benzer bir netice ortaya çıkarak Klasik Yöntemle AHS yöntemi arasında %91,9’luk, Klasik yöntemle BAHS yöntemi arasında %85,7’lik ve AHS yöntemiyle BAHS yöntemi arasında ise %96,5’lik bir ilişki katsayısı ortaya çıkmaktadır.

Dikkat edileceği gibi her iki test sonucunda da elde edilen sonuçlardaki benzerlik önemli düzeydedir ve AHS yöntemiyle BAHS yöntemleri arasındaki benzerlik daha da fazla ortaya çıkmaktadır.

Sonuç olarak 3 adet farklı seçim yöntemi ve örnek uygulamaları verilmiş ve elde edilen seçim sonuçları sunulmuştur. Sunulan bu yöntemlerden klasik yöntem de; değerlendirme esnasında sayıların insana özgü dilsel karşılaştırma ifadelerini yeterince yansıtamamasından kaynaklanan bir karşılaştırma eksikliğinin olduğu ve değerlendirmede güçlük yaşandığı görülmektedir. Ancak bunun yanında uygulamadaki kolaylık ve işlem karmaşasının olmaması da artı bir yönüdür. Fakat adaylar arasında ikili kıyaslamalar esnasındaki yeterince esnek olamama ve gerçek durumu tam olarak yansıtamama sorunlarının olduğu göz ardı edilmemelidir.

Diğer bir yöntem olan AHS yöntemi ise çokça tercih edilen bir yöntemdir. Bu yöntemin seçim esnasında sağladığı avantaj ise, kriterlere ait ağırlıklar hesap edilmekte ve her bir kriter, ağırlığı nispetinde karşılaştırma üzerinde etki yapmaktadır. Bu yolla öznellik sağlanmaya çalışılmaktadır. Bunun yanında AHS yönteminin uygulanabilmesi için bir uzman görüşü gerekliliği, bu nedenle de her

türlü probleme uyarlanamaması, karar vericilerin kriter değerlendirmesi sırasında yapacakları yanlışlıklar da eksiklikleridir. Ayrıca kriter veya aday sayıları arttıkça AHS yönteminin kullanımı oldukça zorlaşmaktadır. Bunun nedeni karar verici tarafından yapılması gereken ikili karşılaştırmaların çok fazlalaşmasıdır.

Bulanık AHS’de ise AHS’de karşılaşılan işlem fazlalığı ve aday ya da kriter sayısının artmasıyla yoğunlaşan bir hesap işlemi yoktur. Bunun nedeni karşılaştırmaların dilsel ifadelerle gerçekleştirilmesidir ve bu gerçektende işlem yükünü önemli bir ölçüde hafifletmektedir. Ayrıca karşılaştırma sonuçlarını ifade etmede de daha başarılı bir yöntemdir. Fakat Bulanık AHS yönteminde AHS yönteminde var olan tutarlılığın ölçülmesi işleminin olmaması Bulanık AHS için eksikliklerdir. Bulanık sınırlar doğru tespit edilmezse hatalı sonuçlar çıkabilir. Ancak insana özgü hissi karşılaştırma ve değerlendirme kolaylığını, güvenilirliğini sağladığı için geliştirilmeye açık bu yöntem ilerleyen yıllarda çok daha fazla alanda kullanılabilecektir.

KAYNAKLAR

1. Öztürk, Z., “İşletmelerde Personel Seçim Yöntemleri ve Psikoteknik”, **Gazi Üniversitesi Yayınları**, Ankara, 4-6, 19, 20, 27-35 (1995).
2. Gülen, A., “Örgütsel etkinlik açısından personel seçimi ve işe almada başarı”, Yüksek Lisans Tezi, **Gazi Üniversitesi Sosyal Bilimler Enstitüsü**, Ankara, 112-115, 135-138 (1995).
3. Mathis, R. L. ve Jackson, J. H., “Personel Human Resource Management 6nd ed.”, **West Publishing Company**, USA, 212,213 (1991).
4. Erdoğan, İ., “İşletmelerde Personel Seçimi ve Başarı Değerlendirme Teknikleri”, İstanbul Üniversitesi İşletme Fakültesi Yayınları, **Küre Ajans**, İstanbul, 248: 45,49-50,65 (1991).
5. Jewell, L. N. ve Siegel, M., “Contemporary Industrial/Organization Psychology 2nd ed.”, **West Publishing Company**, USA, 123,230 (1990).
6. Serhadlıoğlu, G., “Bulanık AHP ve Electre III yöntemlerinin personel seçimi problemine uygulanması”, Yüksek Lisans Tezi, **Gazi Üniversitesi Fen Bilimleri Enstitüsü**, Ankara, 7-10, 16-20, 16-25 (2004).
7. Stoner, J. A. F. ve Freeman R. E., “Management 4nd ed.”, **Prentice-Hall International Editions**, USA, 536 (1986).
8. Bali, Ö., “AHP, Bulanık AHP ve Bulanık Mantık’la Karaharp Okuluna Öğretim Elemanı Seçimi”, Yüksek Lisans Tezi, **Gazi Üniversitesi Fen Bilimleri Enstitüsü**, Ankara, 24-28, 43-46, 49-53 (2004).
9. Bešli, N., “Fuzzy (Belirsiz) Kontrol Sistemlerinin Tasarımı”, Yüksek Lisans Tezi, **Uludağ Üniversitesi Fen Bilimleri Enstitüsü**, Bursa, 1-7, 14-19 (1994).
10. Sağıroğlu, Ş., Beşdok, E. ve Erler, M., “Mühendislikte Yapay Zeka Uygulamaları 1: Yapay Sinir Ağları”, **Ufuk Yayıncılık**, Kayseri, 13-14 (2003).
11. Teke, S., “Hedef Programlama ve Bulanık Hedef Programlama Tekniklerinin Karşılaştırılması, Üretim Planlama Problemine Bir Uygulama”, Yüksek Lisans Tezi, **Gazi Üniversitesi Fen Bilimleri Enstitüsü**, Ankara, 44-50 (1998).
12. Güney, Z., “Eğitim Amaçlı Bulanık Mantık Denetleyicisi Tasarımı”, Yüksek Lisans Tezi, **Gazi Üniversitesi Fen Bilimleri Enstitüsü**, Ankara, 4-14, 30-34, 35 (2002).
13. Cheng, C. H. ve Mon, D. L., “Evaluating weapon system by Analytic Hierarchy Process based on fuzzy sets”, **Fuzzy Sets and Systems**, 63: 1-10 (1994).

14. Cheng, C. H. ve Mon, D. L., “Fuzzy systems reliability analysis by interval confidence”, *Fuzzy Sets and Systems*, 56: 29-35 (1993).
15. Zadeh, L., “Fuzzy sets”, *Information Control*, 8: 338–353 (1965).
16. Kuo, R.J., Chi, S.C. ve Kao, S.S., “A decision support system for selecting convenience store location through integration of fuzzy AHP and artificial neural network”, *Computers in Industry*, 47: 199–214 (2002).
17. Lee, M., Pham, H. ve Zhang, X., “A methodology for priority setting with application to software development process”, *European Journal of Operational Research*, 118: 375–389 (1999).
18. Bay, Ö. F., “Anahtarlamalı Relüktans Motorun Bulanık Mantık Tabanlı Modellenmesi ve Kontrolü”, Doktora Tezi, *Erciyes Üniversitesi Fen Bilimleri Enstitüsü*, Kayseri, 36 (1996).
19. Bezdek, C. J. ve Pall S. K., “Fuzzy Models For Pattern Recognition”, *IEEE Press*, 1-26 (1992)
20. Elmas, Ç., “Bulanık Mantık Denetleyiciler (Kuram, Uygulama, Sinirsel Bulanık Mantık)”, *Seçkin Yayıncılık*, Ankara, 91-97 (2003).
21. Chang, D.-Y., “Extent Analysis and Synthetic Decision, Optimization Techniques and Applications”, *World Scientific*, 1: 352 (1992).
22. Dağdeviren M., Akay D. ve Kurt M. “İş değerlendirme sürecinde analitik hiyerarşi prosesi ve uygulaması”, *Gazi Üniversitesi Müh. Mim. Fak. Der.*, 19(2): 131-138 (2004).
23. Tektaş, A. ve Hortaçsu, A., “Karar vermede etkinliği artıran bir yöntem: Analitik hiyerarşi süreci ve mağaza seçimine uygulanması”, *18. Yıl*, 52-61 (2003)
24. Eren T. ve Dağdeviren M., “Tedarikçi firma seçiminde analitik hiyerarşi prosesi ve 0-1 hedef programlama yöntemlerinin kullanılması”, *Gazi Üniversitesi Müh. Mim. Fak. Der.*, 16(2): 41-52 (2001).
25. Hasgöl, F. ve Koparal C., “Bilgi teknolojilerinin değişim kararlarında analitik hiyerarşi süreci kullanımı”, *YA/EM'2004 - Yöneylem Araştırması/Endüstri Mühendisliği - XXIV Ulusal Kongresi*, Gaziantep – Adana, 20-23 (2004).
26. Kahraman, C., Ruan, D. Ve Doğan, İ., “Fuzzy group decision making for facility location selection”, *Information Sciences*, 157: 135-153 (2003).

27. Cengiz, T. ve Çelem H., “Kırsal kalkınmada analitik hiyerarşi süreci (ahs) yönteminin kullanımı”, *Kafkas Üniversitesi Artvin Orman Fakültesi Dergisi*, 4 (1-2): 144-153 (2003).
28. Saaty, T.L., “The analytic hierarchy process : Planning, Priority Setting, Resource Allocation,”, *Mc Graw Hill Inc.*, New York, 226-238 (1980).
29. Rouyandegh, B. D., “DEA/AHP Sıralı Metodu İle İran Amir Kabir Üniversitesi Fakültelerinin Etkinlik Ölçümü”, Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 14-20 (2004).
30. Kuruüzüm, A. Ve Atsan, N., “Analitik hiyerarşi yöntemi ve işletmecilik alanındaki uygulamaları ”, *Akdeniz Üniversitesi İ.İ.B.F. Dergisi*, 1: 83-105 (2001).
31. Sekreter M. S., Akyüz G. ve Çetin E.İ., ”Şirketlerin derecelendirilmesine ilişkin bir model önerisi: gıda sektörüne yönelik bir uygulama”, *Akdeniz Üniversitesi İ.İ.B.F. Dergisi*, 8: 83-105 (2004).
32. Saaty, T.L., “Decision making new information, ranking and structure”, *Mathematical Modelling*, 8: 1-15,55 (1987).
33. İç, Y.T. ve Yurdakul M., “Analitik hiyerarşi süreci (ahs) yöntemini kullanan bir kredi değerlendirme sistemi”, *Gazi Üniversitesi Müh. Mim. Fak. Der.*, 15(1): 1-14 (2000).
34. Deng, H., “Multicriteria analysis with fuzzy pairwise comparison”, *International Journal of Approximate Reasoning*, 21: 215–231 (1999).
35. Karakaya, K., Yuluğkural, Y. ve Aladağ, Z., “İstanbul boğazı’ndan gemilerin emniyetli geçişinin analitik hiyerarşi prosesi (ahp) kullanarak analizi”, *YA/EM’2004 - Yöneylem Araştırması/Endüstri Mühendisliği - XXIV Ulusal Kongresi*, Gaziantep – Adana, 47-52 (2004).
36. Saaty, T.L., “How to make a decision : the analytic hierarchy process”, *Interface*, 24: 330-342 (1994).
37. Kahraman, C., Ruan, D. ve Tolga, E., “Capital budgeting techniques using discounted fuzzy versus probabilistic cash flows”, *Information Sciences*, 42: 57–76 (2002).
38. Kahraman, C., Cebeci, U. Ve Ruan, D., “Multi-attribute comparison of catering service companies using fuzzy AHP: the case of Turkey”, *Int. J. Production Economics*, 87: 171-184 (2004).

39. Chan, F.T.S., Chan, M.H. ve Tang, N.K.H., "Evaluation methodologies for technology selection", *Journal of Materials Processing Technology*, 107: 330–337 (2000).
40. Kaufmann, A. ve Gupta, M. M., "Introduction to fuzzy arithmetic theory and application", *Van Nostrand Reinhold*, New York, 86-88 (1991).
41. Van Laarhoven, P.J.M. ve Pedrycz, W., "A fuzzy extension of Saaty's priority theory", *Fuzzy Sets and Systems*, 11: 229–241 (1983).
42. Buckley, J.J., "Fuzzy hierarchical analysis", *Fuzzy Sets and Systems*, 17: 233–247 (1985).
43. Stam, A., Minghe, S. ve Haines, M., "Artificial neural network representations for hierarchical preference structures", *Computers and Operations Research*, 23: 1191–1201 (1996).
44. Chang, D.-Y., "Applications of the extent analysis method on fuzzy AHP", *European Journal of Operational Research*, 95: 649-655 (1996).
45. Cheng, C.-H., "Evaluating naval tactical missile systems by fuzzy AHP based on the grade value of membership function", *European Journal of Operational Research*, 96: 343-350 (1997).
46. Weck, M., Klocke, F., Schell, H. Ve Rüanauver, E., "Evaluating alternative production cycles using the extended fuzzy AHP method", *European Journal of Operational Research*, 100: 351-366 (1997).
47. Kahraman, C., Ulukan, Z. ve Tolga, E., "A fuzzy weighted evaluation method using objective and subjective measures", *Proceedings of the International ICSC Symposium on Engineering of Intelligent Systems (EIS'98)*, University of La Laguna Tenerife, Spain, 57–63 (1998).
48. Cheng, C.-H., Yang, K.-L. ve Hwang, C.-L., "Evaluating attack helicopters by AHP based on linguistic variable weight", *European Journal of Operation al Research*, 116: 423–435 (1999).
49. Zhu, K.-J., Jing, Y. ve Chang, D.-Y., "A discussion of extent analysis method and applications of fuzzy AHP", *European Journal of Operational Research*, 116: 450–456 (1999).
50. Chan, F.T.S., Jiang, B. ve Tang, N.K.H., "The development of intelligent decision support tools to aid the design offlexible manufacturing systems", *International Journal of Production Economics*, 65: 73–84 (2000).
51. Leung, L.C. ve Cao, D., "On consistency and ranking of alternatives in fuzzy AHP", *European Journal of Operational Research*, 124: 102–113 (2000).

52. Geyik, P., Uludağ, A., Karabulut, E. ve Saraçbaşı, O.,” Yaşam sürelerindeki farklılıkların gini katsayısı ile incelenmesi”, *VIII. Ulusal Biyoistatistik Kongresi*, Bursa,347-348 (2005).

EKLER

EK-1 “Ortalama.for” isimli simülasyon programının kodları

```

use numerical_libraries
implicit none
integer i,n,adaysay,j,k,tuteleman,eleman(2000),elemanahp(2000)
real ort,std,carp
real tut
real kriter(5),fkriter(5)
real sattop,ahptoplam(2000)
real aday(5,2000),tutoran(2000),max,tsizlikoran
real z(2000,2000),ahpg,fahpg,ahpadayr(2000,2000)
real suttop(2000),fahpadayr(2000,2000),ahp(2000,2000)
real ahpk(5,5),kritera(5),suttopk(5)
integer ahpaday(2000,2000),fahpaday(2000,2000)
integer klasik(2000),fahp(2000,2000),nout,x(2000)
real LK(2000,2000),MK(2000,2000),UK(2000,2000),SUK(2000)
real STLK(2000),STMK(2000),STUK(2000),SLK(2000),SMK(2000),
real TOPLK,TOPMK,TOPUK,ek(2000),V,topek,WK(5)
real LA(2000,2000),MA(2000,2000),UA(2000,2000)
real STLA(5,2000),STMA(5,2000),STUA(5,2000),SLA(2000),SMA(2000)
real TOPLA,TOPMA,TOPUA,WA(5,2000),SUA(2000)
real fahppuan(2000),topklas
real topk,topahp,topfahp
real karetopk,karetopahp,karetopfahp
real SXX,SXAHP,SXFAHP,ORTK,ORTAHP,ORTFAHP
real ZK(2000),Zahp(2000),zfahp(2000)
real scorek(2000),scoreahp(2000),scorefahp(2000),klasikr(2000)
real k_ahp,k_fahp,ahp_fahp,k_ahp_fahp,tfkriter,tfahpr(5),eb,d
real k_ahpe(2000),k_fahpe(2000),ahp_fahpe(2000),k_ahp_fahpe(2000)
real ykahp,ykfahp,yahpf,ytumu
real sk(2000),at(2000),za(2000),sa(2000),fa(2000),zf(2000),sfa(2000)
integer sira(2000),ds

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```
integer ks(1000),nt,uyelikk(5),uyelika(2000,5),fark
real mk1,mk2,mk3,mk4,mk5
real fahpr(2000,5)
integer secim
open (unit=1,FILE='orta')
call umach (2,nout)
print *,' Ornekteki eleman sayisini giriniz ==> '
read *,n
print *,'Kriterler için 0-20 aralığında puanlar girilecektir'
print *,'ANLAMA ANLATMA YETENEĞİ KRİTERİNİN ÖNEMİNE AİT
PUAN = '
read *,fkriter(1)
kriter(1)=fkriter(1)*9/20
print *,'GENEL GÖRÜNÜŞ KRİTERİNİN ÖNEMİNE AİT PUAN = '
read *,fkriter(2)
kriter(2)=fkriter(2)*9/20
print *,'PSİKOLOJİK YAPI KRİTERİNİN ÖNEMİNE AİT PUAN = '
read *,fkriter(3)
kriter(3)=fkriter(3)*9/20
print *,'AİLEVİ SOSYAL DURUM KRİTERİNİN ÖNEMİNE AİT PUAN ='
read *,fkriter(4)
kriter(4)=fkriter(4)*9/20
print *,'BİLİMSEL YETERLİLİK KRİTERİNİN ÖNEMİNE AİT PUAN = '
read *,fkriter(5)
kriter(5)=fkriter(5)*9/20
print *,'k1= ',kriter(1),'k2= ',kriter(2),'k3= ',kriter(3)
print *,'k4= ',kriter(4),'k5= ',kriter(5)
tfkriter=fkriter(1)+fkriter(2)+fkriter(3)+fkriter(4)+fkriter(5)
print *,'Adayların Kriter Puanları Rastgelemleri Girilsin 1-Evet/2-Hayır'
read *,secim
```


EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

if (secim==1) then
print *, 'Random Sayılar Uretiliyor'
do i=1,5,1          ! 5 kriter için aynı işlemler gerçekleştirilecek
call rnund(n,20,x)  !x dizisi 0-20 aralığında random üretilen n kadar değeri
                    tutmak içindir
do j=1,n,1          !n kadar aday için üretilen değerlerin dönüşümü sağlanıyor
z(j,i)=x(j)
ahpg=z(j,i)*9/20
ahpaday(j,i)=ahpg
fahpr(j,i)=x(j)
tfahpr(i)=tfahpr(i)+fahpr(j,i)
if (ahpg>=8.5) then
ahpaday(j,i)=9
else if (ahpg>=7.5 .and. ahpg<=8) then
ahpaday(j,i)=8
else if (ahpg>=6.5 .and. ahpg<=7) then
ahpaday(j,i)=7
else if (ahpg>=5.5 .and. ahpg<=6) then
ahpaday(j,i)=6
else if (ahpg>=4.5 .and. ahpg<=5) then
ahpaday(j,i)=5
else if (ahpg>=3.5 .and. ahpg<=4) then
ahpaday(j,i)=4
else if (ahpg>=2.5 .and. ahpg<=3) then
ahpaday(j,i)=3
else if (ahpg>=1.5 .and. ahpg<=2) then
ahpaday(j,i)=2
else if (ahpg<=1) then
ahpaday(j,i)=1
end if

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

end do
end do
end if
if (secim==2) then
do i=1,5,1      ! 5 kriter için aynı işlemler gerçekleştirilecek
do j=1,n,1
print *,i,'. Kriterden ',j,'. adayın puanını giriniz = '
read *,z(j,i)
ahpg=z(j,i)*9/20
ahpaday(j,i)=ahpg
fahpr(j,i)=z(j,i)
tfahpr(i)=tfahpr(i)+fahpr(j,i)
if (ahpg>=8.5) then
ahpaday(j,i)=9
else if (ahpg>=7.5 .and. ahpg<=8) then
ahpaday(j,i)=8

else if (ahpg>=6.5 .and. ahpg<=7) then
ahpaday(j,i)=7
else if (ahpg>=5.5 .and. ahpg<=6) then
ahpaday(j,i)=6
else if (ahpg>=4.5 .and. ahpg<=5) then
ahpaday(j,i)=5
else if (ahpg>=3.5 .and. ahpg<=4) then
ahpaday(j,i)=4
else if (ahpg>=2.5 .and. ahpg<=3) then
ahpaday(j,i)=3
else if (ahpg>=1.5 .and. ahpg<=2) then
ahpaday(j,i)=2

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

        else if (ahpg<=1) then
            ahpaday(j,i)=1
        end if
    end do
end do
end if
do i=1,5,1
    do j=1,n,1
        ahpadayr(j,i)=ahpaday(j,i)
    end do
end do
end do
print *, 'Uretilen sayilarin toplamı aliniyor'
do i=1,n,1
do j=1,5,1
    klasik(i)=klasik(i)+z(i,j)
        klasikr(i)=klasik(i)
    end do
    eleman(i)=i
end do
CALL RANKS (n,klasikr,0.0,0,0,scorek)

!*****
! *, '***** AHP YÖNTEMİYLE SEÇİM *****'
! ÖNCE KRİTER AĞIRLIKLARINI HESAPLAYALIM
do i=1,5,1
suttopk(i)=0
    do j=1,5,1
        ahpk(j,i)=kriter(j)/kriter(i)
        suttopk(i)=suttopk(i)+ahpk(j,i)
    end do

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

        end do
        do i=1,5,1
            do j=1,5,1
                kriteria(i)=kriteria(i)+(ahpk(i,j)/suttopk(j))
            end do
        kriteria(i)=kriteria(i)/5
        end do
        do k=1,5,1
            do i=1,n,1
                suttop(i)=0
                do j=1,n,1
                    ahp(j,i)=ahpadayr(j,k)/ahpadayr(i,k)
                    suttop(i)=suttop(i)+ahp(j,i)
                end do
            end do
            do i=1,n,1
                do j=1,n,1
                    aday(k,i)=aday(k,i)+(ahp(i,j)/suttop(j))
                end do
            end do
            aday(k,i)=aday(k,i)/n
        end do
        end do !k döngüsünün sonu

        do i=1,n,1
            do k=1,5,1
                ahptoplam(i)=ahptoplam(i)+(aday(k,i)*kriteria(k))
            end do
        end do

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

CALL RANKS (n,ahptoplam,0.0,0,0,scoreahp)

!*****

!*****BULANIK AHP YÖNTEMİYLE

SEÇİM*****'

!ÖNCELİKLE KRİTERLERİN GİRİŞ ÖNEM DEĞERLERİ 0-20
ARALIĞINDA ALINMIŞ VE BULANIKLAŞTIRILMIŞTIR.

!BU İŞLEM İÇİN VERİLEN ÜÇGENSEL ÜYELİK KÜMELERİNE AİT
FONKSİYONLAR KULLANILMIŞTIR.

do i=1,5,1

print *,fkriter(i)

end do

!Kriterlere ait puanlar alındıktan sonra bunların kümelerine göre üyelik
dereceleri belirlenmeli

!*****

do i=1,5,1

mk1=0

mk2=0

mk3=0

mk4=0

mk5=0

! Önemsiz üyelik kümesi için fonksiyon

if (fkriter(i)<=2) then

Mk1=1

uyelikk(i)=1

end if

if (fkriter(i)>2 .and. fkriter(i)<5) then

Mk1=((5-fkriter(i))/3)

uyelikk(i)=1

end if

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

!Zayıf üyelik kümesi için fonksiyon

```

if (fkriter(i)>=3 .and. fkriter(i)<=6.5) then
    mk2=((fkriter(i)-3)/3.5)
    if (mk1<=mk2) then
        uyelikk(i)=2
    end if
end if

```

```

if (fkriter(i)>6.5 .and. fkriter(i)<=10) then
    mk2=((10-fkriter(i))/3.5)
    uyelikk(i)=2
end if

```

!Önemli üyelik kümesi için fonksiyon

```

if (fkriter(i)>=8 .and. fkriter(i)<=11) then
    mk3=((fkriter(i)-8)/3)
    if (mk2<=mk3) then
        uyelikk(i)=3
    end if
end if

if (fkriter(i)>11 .and. fkriter(i)<=14) then
    mk3=((14-fkriter(i))/3)
    uyelikk(i)=3
end if

```

!Çok Önemli üyelik kümesi için fonksiyon

```

if (fkriter(i)>=12 .and. fkriter(i)<=14.5) then
    mk4=((fkriter(i)-12)/2.5)
    if (mk3<=mk4) then

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```
uyelikk(i)=4
```

```
    end if
```

```
end if
```

```
    if (fkriter(i)>14.5 .and. fkriter(i)<=17) then
```

```
mk4=((17-fkriter(i))/2.5)
```

```
uyelikk(i)=4
```

```
    end if
```

!Mükemmel üyelik kümesi için fonksiyon

```
if (fkriter(i)>=15 .and. fkriter(i)<=18) then
```

```
mk5=((fkriter(i)-15)/3)
```

```
if (mk4<=mk5) then
```

```
uyelikk(i)=5
```

```
    end if
```

```
end if
```

```
if (fkriter(i)>18) then
```

```
mk5=1
```

```
uyelikk(i)=5
```

```
end if
```

```
print *,i,' . kriterin üye olduğu küme = ',uyelikk(i)
```

```
print *, '*****ÜYELİK DERECELERİ*****'
```

```
print *, '\ mk1 =',mk1,' mk2=',mk2,' mk3=',mk3,' mk4=',mk4,' mk5=',mk5
```

```
read *
```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

end do

! BÖYLELİKLE KRİTERLER İÇİN SİSTEMİN VERİ TABANIDA
OLUŞTURULMUŞ OLDU

!*****

!KRİTERLER İÇİN KURAL TABANI VE KARAR MEKANİZMASI
AŞAĞIDADIR

!ŞİMDİ	SIRA	KURAL
TABANINDA*****		

! AŞAĞIDA KURAL TABANI VE BU KURALLARA UYGUN OLARAK
İŞLEMLERİ YAPAN KARAR MEKANİZMASI GÖRÜLMEKTEDİR

do i=1,5,1

do j=1,5,1

fark=uyelikk(i)-uyelikk(j)

if (fark==0) then

LK(i,j)=1

MK(i,j)=1

UK(i,j)=1

end if

if (fark==1) then

LK(i,j)=0.5

MK(i,j)=1.25

UK(i,j)=2

end if

if (fark==2) then

LK(i,j)=1.5

MK(i,j)=2.25

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

UK(i,j)=3

end if

if (fark==3) then

LK(i,j)=2.5

MK(i,j)=3.25

UK(i,j)=4

end if

if (fark==4) then

LK(i,j)=3.5

MK(i,j)=4.25

UK(i,j)=5

end if

if (fark==-1) then

LK(i,j)=0.5

MK(i,j)=0.8

UK(i,j)=2

end if

if (fark==-2) then

LK(i,j)=0.33

MK(i,j)=0.44

UK(i,j)=0.66

end if

if (fark==-3) then

LK(i,j)=0.25

MK(i,j)=0.31

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

        UK(i,j)=0.4
        end if

        if (fark== -4) then
            LK(i,j)=0.2
            MK(i,j)=0.24
            UK(i,j)=0.29
        end if

! BOYUT ANALİZİNİN YAPILDIĞI KISIM
!*****
        STLK(i)=STLK(i)+LK(i,j)
        STMK(i)=STMK(i)+MK(i,j)
        STUK(i)=STUK(i)+UK(i,j)
        TOPLK=TOPLK+LK(i,j)
        TOPMK=TOPMK+MK(i,j)
        TOPUK=TOPUK+UK(i,j)
    end do
end do
do i=1,5,1

        SLK(i)=STLK(i)*(1/TOPUK)
        SMK(i)=STMK(i)*(1/TOPMK)
        SUK(i)=STUK(i)*(1/TOPLK)

    end do

!*****
*****

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

!KRİTERLERİN BİRBİRİNE GÖRE MİNİMUM ÜSTÜNLÜKLERİNİN
HESAPLANDIĞI KISIM

```

!*****
*****
topek=0
do i=1,5,1

    ek(i)=2
    do j=1,5,1

        if (i==j) then
            continue
        end if

        if (SMK(i)>=SMK(j)) then
V=1
        else if (SLK(j)>=SUK(i)) then
            V=0
        else
            V=(SLK(j)-SUK(i))/((SMK(i)-SUK(i))-(SMK(j)-SLK(j)))
        end if

        if (V<ek(i)) then
            ek(i)=V
        end if

    end do

!*****DURULAMA
*****

print *, ' V değerlerinin en küçükleri ==> ',ek(i)

```

BİRİMİ

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```
topek=topek+ek(i)
```

```
end do
```

```
eb=0
```

```
do i=1,5,1
```

```
WK(i)=(ek(i)/topek)+(fkriter(i)/tfkriter)
```

```
if (wk(i)>eb) then
```

```
eb=WK(i)
```

```
end if
```

```
end do
```

```
do i=1,5,1
```

```
WK(i)=WK(i)/eb
```

```
print *, 'W değerleri ==> ',WK(i)
```

```
!read *
```

```
end do
```

!DAHA SONRA KRİTERLER İÇİN YAPILAN BULANIKLAŞTIRMA VE
ÜYELİK DERECELERİNİN HESAPLANMASI İŞLEMİ ADAYLAR

!İÇİNDE YAPILIR VE KÜMELERE ÜYELİK DERECELERİ BULUNUR

! her bir kriterden herbir adayın kümesini ve o kümeye üyelik derecesini
bulmalıyız

```
do i=1,5,1
```

```
do j=1,n,1
```

```
mk1=0
```

```
mk2=0
```

```
mk3=0
```

```
mk4=0
```

```
mk5=0
```

! KÖTÜ üyelik kümesiiçin fonksiyon

```
if (fahpr(j,i)<=4.5) then
```

```
Mk1=1
```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

uyelika(j,i)=1
    end if
    if (fahpr(j,i)>4.5 .and. fahpr(j,i)<9) then
Mk1=((9-fahpr(j,i))/4.5)
        uyelika(j,i)=1
    end if
    !GEÇER üyelik kümesi için fonksiyon

if (fahpr(j,i)>=7 .and. fahpr(j,i)<=9) then
    mk2=((fahpr(j,i)-7)/2)
    if (mk1<=mk2) then
        uyelika(j,i)=2
    end if
end if

    if (fahpr(j,i)>9 .and. fahpr(j,i)<=11) then
        mk2=((11-fahpr(j,i))/2)
        uyelika(j,i)=2
    end if
    !ORTA üyelik kümesi için fonksiyon
if (fahpr(j,i)>=9 .and. fahpr(j,i)<=12) then
    mk3=((fahpr(j,i)-9)/3)
    if (mk2<=mk3) then
        uyelika(j,i)=3
    end if
end if

    if (fahpr(j,i)>12 .and. fahpr(j,i)<=15) then
        mk3=((15-fahpr(j,i))/3)
        uyelika(j,i)=3
    end if
end if

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

!İYİ üyelik kümesi için fonksiyon
if (fahpr(j,i)>=12 .and. fahpr(j,i)<=14.5) then
    mk4=((fahpr(j,i)-12)/2.5)
    if (mk3<=mk4) then
        uyelika(j,i)=4
    end if
end if

    if (fahpr(j,i)>14.5 .and. fahpr(j,i)<=17) then
        mk4=((17-fahpr(j,i))/2.5)
        uyelika(j,i)=4
    end if

!Mükemmel üyelik kümesi için fonksiyon
if (fahpr(j,i)>=16 .and. fahpr(j,i)<=18) then
    mk5=((fahpr(j,i)-16)/2)
    if (mk4<=mk5) then
        uyelika(j,i)=5
    end if
end if

if (fahpr(j,i)>18) then
    mk5=1
    uyelika(j,i)=5
end if

print *,i,' . kriterden ',j,' . adayın random puanı = ',fahpr(j,i)
print *,j,' . adayın ',i,' . kriterde üye olduğu küme = ',uyelika(j,i)
print *, '*****ÜYELİK DERECELERİ*****'
print *, ' mk1 =',mk1,' mk2=',mk2,' mk3=',mk3,' mk4=',mk4,' mk5=',mk5
!read *
end do
end do

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

! BÖYLELİKLE ADAYLAR İÇİN SİSTEMİN VERİ TABANIDA
OLUŞTURULMUŞ OLDU

!*****

!ADAYLAR İÇİN KURAL TABANI VE KARAR MEKANİZMASI
AŞAĞIDIR

!ŞİMDİ	SIRA	KURAL
TABANINDA*****		

! AŞAĞIDA KURAL TABANI VE BU KURALLARA UYGUN OLARAK
İŞLEMLERİ YAPAN KARAR MEKANİZMASI GÖRÜLMEKTEDİR

do k=1,5,1

TOPLA=0

TOPMA=0

TOPUA=0

do i=1,n,1

do j=1,n,1

fark=uyelika(i,k)-uyelika(j,k)

if (fark==0) then

LA(i,j)=1

MA(i,j)=1

UA(i,j)=1

end if

if (fark==1) then

LA(i,j)=0.5

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

MA(i,j)=1.25

UA(i,j)=2

end if

if (fark==2) then

LA(i,j)=1.5

MA(i,j)=2.25

UA(i,j)=3

end if

if (fark==3) then

LA(i,j)=2.5

MA(i,j)=3.25

UA(i,j)=4

end if

if (fark==4) then

LA(i,j)=3.5

MA(i,j)=4.25

UA(i,j)=5

end if

if (fark==-1) then

LA(i,j)=0.5

MA(i,j)=0.8

UA(i,j)=2

end if

if (fark==-2) then

LA(i,j)=0.33

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

        MA(i,j)=0.44
        UA(i,j)=0.66
    end if

    if (fark== -3) then
        LA(i,j)=0.25
        MA(i,j)=0.31
        UA(i,j)=0.4
    end if
if (fark== -4) then
    LA(i,j)=0.2
    MA(i,j)=0.24
    UA(i,j)=0.29
end if

! BOYUT ANALİZİNİN YAPILDIĞI KISIM
!*****

    STLA(k,i)=STLA(k,i)+LA(i,j)
    STMA(k,i)=STMA(k,i)+MA(i,j)
    STUA(k,i)=STUA(k,i)+UA(i,j)
    TOPLA=TOPLA+LA(i,j)
    TOPMA=TOPMA+MA(i,j)
    TOPUA=TOPUA+UA(i,j)

end do

end do
do i=1,n,1

    SLA(i)=STLA(k,i)*(1/TOPUA)
    SMA(i)=STMA(k,i)*(1/TOPMA)

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

SUA(i)=STUA(k,i)*(1/TOPLA)

end do

!*****
*****

!ADAYLARIN BİRBİRİNE GÖRE MİNİMUM ÜSTÜNLÜKLERİNİN
HESAPLANDIĞI KISIM
!*****
*****

topek=0
do i=1,n,1

ek(i)=2
do j=1,n,1

if (i==j) then
continue
end if

if (SMA(i)>=SMA(j)) then
V=1
else if (SLA(j)>=SUA(i)) then
V=0
else
V=(SLA(j)-SUA(i))/((SMA(i)-SUA(i))-(SMA(j)-SLA(j)))
end if

if (V<ek(i)) then

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

        ek(i)=V
    end if
end do
!*****DURULAMA
*****
print *, ' V değerlerinin en küçükleri ==> ',ek(i)
!read *
        topek=topek+ek(i)
    end do
    eb=0

do i=1,n,1

    WA(k,i)=(ek(i)/topek)+(fahpr(i,k)/tfahpr(k))
        if (WA(k,i)>eb) then
            eb=WA(k,i)
        end if
    end do

do i=1,n,1

    WA(k,i)=WA(k,i)/eb
        print *, 'W değerleri ==> ',WA(k,i)
        !read *
    end do
end do ! K DÖNGÜSÜNÜN SONU
*****
!*****
*****

```

BİRİMİ

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```
eb=0
```

```
do i=1,n,1
```

```
do j=1,5,1
```

```
FAHPPUAN(i)=FAHPPUAN(i)+(WK(j)*WA(j,i))
```

```
end do
```

```
if (fahppuan(i)>eb) then
```

```
eb=fahppuan(i)
```

```
end if
```

```
end do
```

```
do i=1,n,1
```

```
fahppuan(i)=fahppuan(i)/eb
```

```
print *, 'BAHP yöntemine göre ', i, ' . adayın puanı==> ', fahppuan(i)
```

```
sira(i)=i
```

```
end do
```

```
CALL RANKS (n,fahppuan,0.0,0,0,scorefahp)
```

```
topk=0
```

```
topahp=0
```

```
topfahp=0
```

```
karetopk=0
```

```
karetopahp=0
```

```
karetopfahp=0
```

```
do i=1,n,1
```

```
topk=topk+klasik(i)
```

```
topahp=topahp+ahptoplama(i)
```

```
topfahp=topfahp+fahppuan(i)
```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```

karetopk=karetopk+(klasik(i)*klasik(i))
    karetopahp=karetopahp+(ahptoplam(i)*ahptoplam(i))
    karetopfahp=karetopfahp+(fahppuan(i)+fahppuan(i))
end do

```

```

SXX=sqrt((karetopk-((topk*topk)/n))/(n-1))
SXAHP=sqrt((karetopahp-((topahp*topahp)/n))/(n-1))
SXFAHP=sqrt((karetopfahp-((topfahp*topfahp)/n))/(n-1))

```

```

ortk=topk/n
ortahp=topahp/n

```

```

ortfahp=topfahp/n
do i=1,n,1

```

```

    ZK(i)=(klasik(i)-ortk)/SXX
    ZAHP(i)=(ahptoplam(i)-ortahp)/SXAHP
    ZFAHP(i)=(fahppuan(i)-ortfahp)/SXFAHP

```

```

end do
do i=1,n,1

```

```

    ks(i)=klasik(i)
    sk(i)=scorek(i)
    at(i)=ahptoplam(i)
    za(i)=zahp(i)
    sa(i)=scoreahp(i)

```

```

    fa(i)=fahppuan(i)
    zf(i)=zfahp(i)

```

EK-1(Devam) “Ortalama.for” isimli simülasyon programının kodları

```
sfa(i)=scorefahp(i)
```

```

write(nt,10) ks(i),ZK(i),sk(i),at(i),za(i),sa(i),fa(i),zf(i),sfa(i)
10  format(/,i3,f8.5,f5.1,'||',f8.5,f8.5,f5.1,'||',f8.5,f8.5,f5.1)
write(1,20)ks(i),ZK(i),sk(i),at(i),za(i),sa(i),fa(i),zf(i),sfa(i)
20  format(i4,f8.3,f8.1,f8.5,f8.3,f8.1,f8.5,f8.3,f8.1)
end do
do i=1,n-1,1
do j=i+1,n,1

if(fahppuan(i)<fahppuan(j)) then
d=fahppuan(i)
fahppuan(i)=fahppuan(j)
fahppuan(j)=d

ds=sira(i)
sira(i)=sira(j)
sira(j)=ds
end if
end do
end do

do i=1,n,1
print *, 'BAHP yöntemine göre ',i, ' . adayın puanı==> ',fahppuan(i)
end do

end

```

EK-2 “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

Şekil 2.1. Kriterlere ait puaların girişi ve karşılaştırılması işlemleri penceresi

	AAY	GG	PY	ASD	BY

Şekil 2.2. Adaylara ait puaların girişi ve karşılaştırılması işlemleri penceresi

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Grids, DBGrids, StdCtrls, ComCtrls, Buttons, DBMask, DBCtrls;
type
  TForm1 = class(TForm)
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    TabSheet2: TTabSheet;
    GroupBox1: TGroupBox;
    GroupBox2: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Memo1: TMemo;
    GroupBox3: TGroupBox;
    GroupBox4: TGroupBox;
    GroupBox5: TGroupBox;
    Label6: TLabel;
    Edit6: TEdit;
    GroupBox6: TGroupBox;
    Label7: TLabel;
    Label8: TLabel;
    Label9: TLabel;
    Label10: TLabel;
    Label11: TLabel;
  end;

```


EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

BitBtn1: TBitBtn;
    BitBtn2: TBitBtn;
    BitBtn4: TBitBtn;
    BitBtn6: TBitBtn;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Button1: TButton;
    StringGrid1: TStringGrid;
    BitBtn8: TBitBtn;
    Memo2: TMemo;
    Edit7: TEdit;
    Edit8: TEdit;
    Edit9: TEdit;
    Edit10: TEdit;
    Edit11: TEdit;
    Memo3: TMemo;
    Memo4: TMemo;
    SaveDialog1: TSaveDialog;
    Edit13: TEdit;
    Label12: TLabel;
    BitBtn3: TBitBtn;
    procedure Memo1Click(Sender: TObject);
    procedure Edit6Exit(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure BitBtn2Click(Sender: TObject);
    procedure BitBtn4Click(Sender: TObject);

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

procedure FormCreate(Sender: TObject);
    procedure BitBtn6Click(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure Edit1KeyPress(Sender: TObject; var Key: Char);
    procedure Edit2KeyPress(Sender: TObject; var Key: Char);
    procedure Edit3KeyPress(Sender: TObject; var Key: Char);
    procedure Edit4KeyPress(Sender: TObject; var Key: Char);
    procedure Edit5KeyPress(Sender: TObject; var Key: Char);
    procedure BitBtn8Click(Sender: TObject);
    procedure Edit7KeyPress(Sender: TObject; var Key: Char);
    procedure Edit8KeyPress(Sender: TObject; var Key: Char);
    procedure Edit9KeyPress(Sender: TObject; var Key: Char);
    procedure Edit10KeyPress(Sender: TObject; var Key: Char);
    procedure Edit11KeyPress(Sender: TObject; var Key: Char);
    procedure Edit6KeyPress(Sender: TObject; var Key: Char);
    procedure Edit13KeyPress(Sender: TObject; var Key: Char);
    procedure BitBtn3Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
var
    Form1: TForm1;
    fkriter:array[1..5] of real;
    fahpr:array[1..2000,1..5] of real;
    adsay,denemeno,adayno:integer;
    WK:array[1..5] of real;
    WA:array[1..5,1..2000] of real;

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

FAHPPUAN:array[1..200]of real;
adayads:array[1..200] of string;
implementation
{$R *.dfm}
procedure TForm1.Memo1Click(Sender: TObject);
var
i,j,fark:integer;
mk1,mk2,mk3,mk4,mk5:real;
uyelikk:array[1..5] of integer;
lk,mk,uk:array[1..5,1..5] of real;
STLK,STMK,STUK:array[1..5] of real;
TOPLK,TOPMK,TOPUK:real;
SLK,SMK,SUK,ek:array[1..5] of real;
topek,V,topkriter,eb:real;
begin
//Kriterlere ait puanlar alındıktan sonra bunların kümelerine göre üyelik dereceleri
belirlenmeli
//*****
*****

memo1.Color:=clSkyBlue;
fkriter[1]:=strtofloat(edit7.text);
fkriter[2]:=strtofloat(edit8.text);
fkriter[3]:=strtofloat(edit9.text);
fkriter[4]:=strtofloat(edit10.text);
fkriter[5]:=strtofloat(edit11.text);
topkriter:=fkriter[1]+fkriter[2]+fkriter[3]+fkriter[4]+fkriter[5];
for i:=1 to 5 do
begin
mk1:=0;mk2:=0;mk3:=0;mk4:=0;mk5:=0;
// Önemsiz üyelik kümesi için fonksiyon

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

if (fkriter[i]<=2) then
  begin
    Mk1:=1;
    uyelikk[i]:=1;
  end;
if (fkriter[i]>=2) and (fkriter[i]<=5) then
  begin
    Mk1:=((5-fkriter[i])/3);
    uyelikk[i]:=1;
  end;
//Zayıf üyelik kümesi için fonksiyon
if (fkriter[i]>=3) and (fkriter[i]<=6.5) then
  begin
    mk2:=((fkriter[i]-3)/3.5);
    if (mk1<=mk2) then
      uyelikk[i]:=2;
    end;
if (fkriter[i]>=6.5) and (fkriter[i]<=10) then
  begin
    mk2:=((10-fkriter[i])/3.5);
    uyelikk[i]:=2;
  end;
//Önemli üyelik kümesi için fonksiyon
if (fkriter[i]>=8) and (fkriter[i]<=11) then
  begin
    mk3:=((fkriter[i]-8)/3);
    if (mk2<=mk3) then
      uyelikk[i]:=3;
    end;

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

if (fkriter[i]>=11) and (fkriter[i]<=14) then
  begin
    mk3:=((14-fkriter[i])/3);
    uyelikk[i]:=3;
  end;
//Çok Önemli üyelik kümesi için fonksiyon
if (fkriter[i]>=12) and (fkriter[i]<=14.5) then
  begin
    mk4:=((fkriter[i]-12)/2.5);
    if (mk3<=mk4) then
      uyelikk[i]:=4;
    end;
if (fkriter[i]>=14.5) and (fkriter[i]<=17) then
  begin
    mk4:=((17-fkriter[i])/2.5);
    uyelikk[i]:=4;
  end;
//Mükemmel üyelik kümesi için fonksiyon
if (fkriter[i]>=15) and (fkriter[i]<=18) then
  begin
    mk5:=((fkriter[i]-15)/3);
    if (mk4<=mk5) then
      uyelikk[i]:=5;
    end;
if (fkriter[i]>=18) then
  begin
    mk5:=1;
    uyelikk[i]:=5;
  end;

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

if uyelikk[i]=1 then
begin
memo3.Lines.Add(inttostr(i)+' . Kriterinin Önem Kümesi ==> ÖNEMSİZ dir ve
Üyelik Dereceleri');
memo3.Lines.Add('Önemsiz Kümesine = % '+floattostr(mk1*100)+' ve Zayıf
Kümesine = % '+floattostr(mk2*100)+' dir. ');
end
else if uyelikk[i]=2 then
begin
memo3.Lines.Add(inttostr(i)+' . Kriterin Önem Kümesi ==> ZAYIF dir ve
Üyelik Dereceleri');
memo3.Lines.Add('Önemsiz Kümesine = % '+floattostr(mk1*100)+' Zayıf
Kümesine = % '+floattostr(mk2*100)+' Önemli Kümesine = %
'+floattostr(mk3*100)+' dir. ');
end
else if uyelikk[i]=3 then
begin
memo3.Lines.Add(inttostr(i)+' . Kriterin Önem Kümesi ==> ÖNEMLİ dir ve
Üyelik Dereceleri');
memo3.Lines.Add('Zayıf Kümesine = % '+floattostr(mk2*100)+' Önemli
Kümesine = % '+floattostr(mk3*100)+' Çok önemli Kümesine = %
'+floattostr(mk4*100)+' dir. ');
end
else if uyelikk[i]=4 then
begin
memo3.Lines.Add(inttostr(i)+' . Kriterin Önem Kümesi ==> ÇOK ÖNEMLİ dir
ve Üyelik Dereceleri');

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

memo3.Lines.Add('Önemli Kümesine = % '+floattostr(mk3*100)+' Çokönemli
Kümesine = % '+floattostr(mk4*100)+' Mükemmel Kümesine = %
'+floattostr(mk5*100)+' dir. ');
end
else if uyelikk[i]=5 then
begin
memo3.Lines.Add(inttostr(i)+' . Kriterin Önem Kümesi ==> MÜKEMMEL dir
ve Üyelik Dereceleri');
memo3.Lines.Add('Çokönemli Kümesine = % '+floattostr(mk4*100)+'
Mükemmel Kümesine = % '+floattostr(mk5*100)+' dir. ');
end;
end;
{
! BÖYLELİKLE KRİTERLER İÇİN SİSTEMİN VERİ TABANIDA
OLUŞTURULMUŞ OLDU
*****
*****
}
{
KRİTERLER İÇİN KURAL TABANI VE KARAR MEKANİZMASI
AŞAĞIDIR
ŞİMDİ SIRA KURAL TABANINDA*****
AŞAĞIDA KURAL TABANI VE BU KURALLARA UYGUN OLARAK
İŞLEMLERİ YAPAN KARAR MEKANİZMASI GÖRÜLMEKTEDİR
}
for i:=1 to 5 do
for j:=1 to 5 do
begin
fark:=uyelikk[i]-uyelikk[j];
if (fark=0) then

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
begin
    LK[i,j]:=1;
    MK[i,j]:=1;
    UK[i,j]:=1;
end
    else if (fark=1) then
begin
    LK[i,j]:=0.5;
    MK[i,j]:=1.25;
    UK[i,j]:=2;
end
    else if (fark=2) then
begin
    LK[i,j]:=1.5;
    MK[i,j]:=2.25;
    UK[i,j]:=3;
end
    else if (fark=3) then
begin
    LK[i,j]:=2.5;
    MK[i,j]:=3.25;
    UK[i,j]:=4;
end
    else if (fark=4) then
begin
    LK[i,j]:=3.5;
    MK[i,j]:=4.25;
    UK[i,j]:=5;
end
```


EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

    else if (fark=-1) then
begin
    LK[i,j]:=0.5;
    MK[i,j]:=0.8;
    UK[i,j]:=2;
end
    else if (fark=-2) then
begin
    LK[i,j]:=0.33;
    MK[i,j]:=0.44;
    UK[i,j]:=0.66;
end
    else if (fark=-3) then
begin
    LK[i,j]:=0.25;
    MK[i,j]:=0.31;
    UK[i,j]:=0.4;
end
    else if (fark=-4) then
begin
    LK[i,j]:=0.2;
    MK[i,j]:=0.24;
    UK[i,j]:=0.29;
end;

```

{ ! yukarıda görülen kısım FLC nin kural tabanını oluşturmaktadır.
!bilgisayarda işlem karmaşası oluşturmamak için şartlar direk yazılmak yerine

! aynı işlemi yapan daha az karmaşık bir şart yapısı tercih edilmiştir

! bu yolla işlemlerin daha hızlı yapılarak neticelenmesi amaçlanmıştır.

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

!Kural yapısıda oluşturulduktan sonra, artık giriş kısmında bulanıklaştırılan sayılar

!aitlik kümeleri ve aitlik dereceleri belirlendikten sonra kural tabanına uygun olarak karşılaştırılarak

!elde etmek istediğimiz netice oluşmaktadır.

!Karar biriminde BAHS nin algoritması gereğince önce kural tabanına uygun olarak karşılaştırmalar yapılacak sonra

!boyut analizi yapılacak ve daha sonra minimum üstünlük dereceleri hesaplanarak son olarakta

!durulama kısmında ağırlık ortalaması yöntemiyle sonuç gerçek değerlere dönüştürülecektir.

! BOYUT ANALİZİNİN YAPILDIĞI KISIM

!***** }

STLK[i]:=STLK[i]+LK[i,j];

STMK[i]:=STMK[i]+MK[i,j];

STUK[i]:=STUK[i]+UK[i,j];

TOPLK:=TOPLK+LK[i,j];

TOPMK:=TOPMK+MK[i,j];

TOPUK:=TOPUK+UK[i,j];

//memo3.Lines.Add(inttostr(i)+'Kriterle '+inttostr(j)+' Kriterin Karşılaştırması =
['+floattostr(LK[i,j])+', '+floattostr(MK[i,j])+', '+floattostr(UK[i,j])+']');

end;

for i:=1 to 5 do

begin

SLK[i]:=STLK[i]*(1/TOPUK);

SMK[i]:=STMK[i]*(1/TOPMK);

SUK[i]:=STUK[i]*(1/TOPLK);

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
//          memo3.Lines.Add(inttostr(i)+'Kriterden          BA=
('+floattostr(stlk[i])+','+floattostr(stmk[i])+','+floattostr(stuk[i])+')+'*'+(1/'+floattostr
(topuk)+'1/'+floattostr(topmk)+'1/'+floattostr(toplk)+')');
    end;
{
    !*****
*****
    !KRİTERLERİN BİRBİRİNE GÖRE MİNİMUM ÜSTÜNLÜKLERİNİN
HESAPLANDIĞI KISIM
    !*****
***** }
    topek:=0;
    for i:=1 to 5 do
        begin
            ek[i]:=1;
            for j:=1 to 5 do
                begin
                    if (i=j) then
                        continue;
                    if (SMK[i]>=SMK[j]) then
                        V:=1
                    else if (SLK[j]>=SUK[i]) then
                        V:=0
                    else
                        V:=(SLK[j]-SUK[i])/((SMK[i]-SUK[i])-(SMK[j]-SLK[j]));
                // memo3.Lines.Add(inttostr(i)+' kriterle '+inttostr(j)+' kriterin V si =
'+floattostr(v));
```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

        if (V<ek[i]) then
            ek[i]:=V;
        end;
    {
        !*****
*****
        !*****DURULAMA                                BİRİMİ
***** }
        topek:=topek+ek[i];
        end;
        eb:=0;
        for i:=1 to 5 do
            begin
                showmessage('eklenen      miktar      =      '+floattostr(fkriter[i])+'      /
'+floattostr(topkriter)+' = '+floattostr(fkriter[i]/topkriter));
                WK[i]:=(ek[i]/topek)+(fkriter[i]/topkriter);
                if eb<WK[i] then
                    eb:=WK[i];
                //showmessage('kriter      puanı      =      '+floattostr(fkriter[i])+'      topkriter=
'+floattostr(topkriter)+' enbüyük değer =' +floattostr(eb));
                end;
                for i:=1 to 5 do
                    begin
                        WK[i]:=WK[i]/eb;
                        //      memo2.Lines.Add('EnKüçük = '+floattostr(ek[i])+' En küçüklerin toplamı
= '+floattostr(topek));
                        memo3.Lines.Add("");
                        if i=1 then

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

        memo3.Lines.Add('Anlama Anlatma Yeteneği Kriterinin Ağırlığı ==>
'+floattostr(WK[i]));
        if i=2 then
            memo3.Lines.Add('Genel Görünüş Kriterinin Ağırlığı ==>
'+floattostr(WK[i]));
            if i=3 then
                memo3.Lines.Add('Psikolojik Yapı Kriterinin Ağırlığı ==>
'+floattostr(WK[i]));
            if i=4 then
                memo3.Lines.Add('Ailevi ve Sosyal Durum Kriterinin Ağırlığı ==>
'+floattostr(WK[i]));
            if i=5 then
                memo3.Lines.Add('Bilimsel Yeterlilik Kriterinin Ağırlığı ==>
'+floattostr(WK[i]));
        end;
//BU DURUMDA KRİTERLERİN ÖNEM DERECELİRİ 1 DEN AŞAĞIYA
DOĞRU YENİDEN DEĞERLENDİRİLDİ
//*****
**
end;
procedure TForm1.Edit6Exit(Sender: TObject);
begin
    adsay:=strtoint(edit6.Text);
    StringGrid1.RowCount:=adsay+1;
end;
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
    adayads[adayno]:=edit13.Text;
    fahpr[adayno,1]:=strtofloat(edit1.text);

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
fahpr[adayno,2]:=strtofloat(edit2.text);
fahpr[adayno,3]:=strtofloat(edit3.text);
fahpr[adayno,4]:=strtofloat(edit4.text);
fahpr[adayno,5]:=strtofloat(edit5.text);
StringGrid1.Cells[1,adayno]:=edit1.text;
StringGrid1.Cells[2,adayno]:=edit2.text;
StringGrid1.Cells[3,adayno]:=edit3.text;
StringGrid1.Cells[4,adayno]:=edit4.text;
StringGrid1.Cells[5,adayno]:=edit5.text;
StringGrid1.Cells[0,adayno]:=edit13.Text;
inc(adayno);
edit1.Text:="";
edit2.Text:="";
edit3.Text:="";
edit4.Text:="";
edit5.Text:="";
edit13.Text:="";
edit13.SetFocus;
GroupBox6.Caption:=inttostr(adayno)+' Adayın Puanları';
if (adayno<adsay) then
bitbtn1.Caption:=inttostr(adayno+1)+' ADAYA GEÇ';
if (adayno=(adsay+1)) then
begin
GroupBox6.Caption:='Karşılaştırmaları Yaptır Butonuna Tıklayınız';
bitbtn1.Caption:='Karşılaştırmaları Yaptır Butonuna Tıklayınız';
bitbtn1.Enabled:=false;
bitbtn6.Enabled:=true;
bitbtn6.SetFocus;
end;
```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

if (adayno=adsay) then
  bitbtn1.Caption:='ADAY PUANI GİRİŞİNİ BİTİR';
end;
procedure TForm1.BitBtn2Click(Sender: TObject);
begin
  edit7.Text:="";
  edit8.Text:="";
  edit9.Text:="";
  edit10.Text:="";
  edit11.Text:="";
  memo3.Lines.Clear;
  edit7.SetFocus;
end;
procedure TForm1.BitBtn4Click(Sender: TObject);
begin
  Application.Terminate;
end;
procedure TForm1.FormCreate(Sender: TObject);
begin
  adayno:=1;
  StringGrid1.Cells[1,0]:='AAY';
  StringGrid1.Cells[2,0]:='GG';
  StringGrid1.Cells[3,0]:='PY';
  StringGrid1.Cells[4,0]:='ASD';
  StringGrid1.Cells[5,0]:='BY';
end;
procedure TForm1.BitBtn6Click(Sender: TObject);
var
  LA,MA,UA:array [1..200,1..200] of real;

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
SLA,SMA,SUA,ek:array [1..200] of real;
STLA,STMA,STUA:array [1..5,1..200] of real;
TOPLA,TOPMA,TOPUA,topek:real;
i,j,k,fark:integer;
mk1,mk2,mk3,mk4,mk5,V:real;
uyelika:array[1..200,1..5] of integer;
tut,eb:real;
tutads:string;
sirano:array[1..200] of integer;
topahppuan:array[1..5] of real;
begin
{DAHA SONRA KRİTERLER İÇİN YAPILAN BULANIKLAŞTIRMA VE
ÜYELİK DERECELERİNİN HESAPLANMASI İŞLEMİ ADAYLAR
!İÇİNDE YAPILIR VE KÜMELERE ÜYELİK DERECELERİ BULUNUR
! her bir kriterden herbir adayın kümesini ve o kümeye üyelik derecesini
bulmalıyız
}
for i:=1 TO 5 do
  for j:=1 to adsay do
    begin
      mk1:=0;
      mk2:=0;
      mk3:=0;
      mk4:=0;
      mk5:=0;
      topahppuan[i]:=topahppuan[i]+fahpr[j,i];
      // KÖTÜ üyelik kümesi için fonksiyon
      if (fahpr[j,i]<=4.5) then
        begin
```


EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

Mk1:=1;
    uyelika[j,i]:=1;
end;
if (fahpr[j,i]>=4.5) and (fahpr[j,i]<=9) then
begin
Mk1:=((9-fahpr[j,i])/4.5);
    uyelika[j,i]:=1;
end;
//GEÇER üyelik kümesi için fonksiyon
    if (fahpr[j,i]>=7) and (fahpr[j,i]<=9) then
begin
        mk2:=((fahpr[j,i]-7)/2);
        if (mk1<=mk2) then
            uyelika[j,i]:=2;
        end;
        if (fahpr[j,i]>=9) and (fahpr[j,i]<=11) then
begin
            mk2:=((11-fahpr[j,i])/2);
            uyelika[j,i]:=2;
        end;
//ORTA üyelik kümesi için fonksiyon
        if (fahpr[j,i]>=9) and (fahpr[j,i]<=12) then
begin
            mk3:=((fahpr[j,i]-9)/3);
            if (mk2<=mk3) then
                uyelika[j,i]:=3;
            end;
            if (fahpr[j,i]>=12) and (fahpr[j,i]<=15) then
begin

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

mk3:=((15-fahpr[j,i])/3);
    uyelika[j,i]:=3;
end;
//İYİ üyelik kümesi için fonksiyon
    if (fahpr[j,i]>=12) and (fahpr[j,i]<=14.5) then
begin
    mk4:=((fahpr[j,i]-12)/2.5);
    if (mk3<=mk4) then
        uyelika[j,i]:=4;
    end;
    if (fahpr[j,i]>=14.5) and (fahpr[j,i]<=17) then
begin
    mk4:=((17-fahpr[j,i])/2.5);
    uyelika[j,i]:=4;
    end;
//Mükemmel üyelik kümesi için fonksiyon
    if (fahpr[j,i]>=16) and (fahpr[j,i]<=18) then
begin
    mk5:=((fahpr[j,i]-16)/2);
    if (mk4<=mk5) then
        uyelika[j,i]:=5;
    end;
    if (fahpr[j,i]>=18) then
begin
    mk5:=1;
    uyelika[j,i]:=5;
    end;
if uyelika[j,i]=1 then
begin

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

memo2.Lines.Add(inttostr(j)+' Adayın '+inttostr(i)+' . Kriterde Önem Kümesi ==>
KÖTÜ dür ve Üyelik Dereceleri');
    memo2.Lines.Add('KÖTÜ Kümesine = % '+floattostr(mk1*100)+' ve GEÇER
Kümesine = % '+floattostr(mk2*100)+' dir. ');
    end
    else if uyelika[j,i]=2 then
    begin
        memo2.Lines.Add(inttostr(j)+' Adayın '+inttostr(i)+' . Kriterin Önem Kümesi
==> GEÇER dir ve Üyelik Dereceleri');
        memo2.Lines.Add('KÖTÜ Kümesine = % '+floattostr(mk1*100)+' GEÇER
Kümesine = % '+floattostr(mk2*100)+' ORTA Kümesine = %
'+floattostr(mk3*100)+' dir. ');
        end
        else if uyelika[j,i]=3 then
        begin
            memo2.Lines.Add(inttostr(j)+' Adayın '+inttostr(i)+' . Kriterin Önem Kümesi
==> ORTA dır ve Üyelik Dereceleri');
            memo2.Lines.Add('GEÇER Kümesine = % '+floattostr(mk2*100)+' ORTA
Kümesine = % '+floattostr(mk3*100)+' İYİ Kümesine = % '+floattostr(mk4*100)+'
dir. ');
            end
            else if uyelika[j,i]=4 then
            begin
                memo2.Lines.Add(inttostr(j)+' Adayın '+inttostr(i)+' . Kriterin Önem Kümesi
==> İYİ dir ve Üyelik Dereceleri');
                memo2.Lines.Add('ORTA Kümesine = % '+floattostr(mk3*100)+' İYİ Kümesine
= % '+floattostr(mk4*100)+' Mükemmel Kümesine = % '+floattostr(mk5*100)+'
dir. ');
                end

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
else if uyelika[j,i]=5 then
```

```
begin
```

```
memo2.Lines.Add(inttostr(j)+' Adayın '+inttostr(i)' . Kriterin Önem Kümesi  
==> MÜKEMMEL dir ve Üyelik Dereceleri');
```

```
memo2.Lines.Add('İYİ Kümesine = % '+floattostr(mk4*100)+' Mükemmel  
Kümesine = % '+floattostr(mk5*100)+' dir.');
```

```
end;
```

```
end;
```

```
{      ! BÖYLELİKLE ADAYLAR İÇİN SİSTEMİN VERİ TABANIDA  
OLUŞTURULMUŞ OLDU
```

```
!*****
```

```
*****
```

```
!ADAYLAR İÇİN KURAL TABANI VE KARAR MEKANİZMASI  
AŞAĞIDIR
```

```
!ŞİMDİ
```

```
SIRA
```

```
KURAL
```

```
TABANINDA*****
```

```
! AŞAĞIDA KURAL TABANI VE BU KURALLARA UYGUN OLARAK  
İŞLEMLERİ YAPAN KARAR MEKANİZMASI GÖRÜLMEKTEDİR
```

```
}
```

```
for k:=1 to 5 do
```

```
begin
```

```
TOPLA:=0;
```

```
TOPMA:=0;
```

```
TOPUA:=0;
```

```
for i:=1 to adsay do
```

```
for j:=1 to adsay do
```

```
begin
```

```
fark:=uyelika[i,k]-uyelika[j,k];
```

```
if (fark=0) then
```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
begin
    LA[i,j]:=1;
    MA[i,j]:=1;
    UA[i,j]:=1;
end;
if (fark=1) then
begin
    LA[i,j]:=0.5;
    MA[i,j]:=1.25;
    UA[i,j]:=2;
end;
if (fark=2) then
begin
    LA[i,j]:=1.5;
    MA[i,j]:=2.25;
    UA[i,j]:=3;
end;
if (fark=3) then
begin
    LA[i,j]:=2.5;
    MA[i,j]:=3.25;
    UA[i,j]:=4;
end;
if (fark=4) then
begin
    LA[i,j]:=3.5;
    MA[i,j]:=4.25;
    UA[i,j]:=5;
end;
```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

    if (fark=-1) then
begin
    LA[i,j]:=0.5;
    MA[i,j]:=0.8;
    UA[i,j]:=2;
    end;
    if (fark=-2) then
begin
    LA[i,j]:=0.33;
    MA[i,j]:=0.44;
    UA[i,j]:=0.66;
    end;
    if (fark=-3) then
begin
    LA[i,j]:=0.25;
    MA[i,j]:=0.31;
    UA[i,j]:=0.4;
    end;
    if (fark=-4) then
begin
    LA[i,j]:=0.2;
    MA[i,j]:=0.24;
    UA[i,j]:=0.29;
    end;

```

{ ! yukarıda görülen kısım FLC nin kural tabanını oluşturmaktadır.
!bilgisayarda işlem karmaşası oluşturmamak için şartlar direk yazılmak yerine

! aynı işlemi yapan daha az karmaşık bir şart yapısı tercih edilmiştir
! bu yolla işlemlerin daha hızlı yapılarak neticelenmesi amaçlanmıştır.

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

!Kural yapısıda oluşturulduktan sonra, artık giriş kısmında bulanıklaştırılan sayılar

!aitlik kümeleri ve aitlik dereceleri belirlendikten sonra kural tabanına uygun olarak karşılaştırılarak

!elde etmek istediğimiz netice oluşmaktadır.

!Karar biriminde BAHS nin algoritması gereğince önce kural tabanına uygun olarak karşılaştırmalar yapılacak sonra

!boyut analizi yapılacak ve daha sonra minimum üstünlük dereceleri hesaplanarak son olarakta

!durulama kısmında ağırlık ortalaması yöntemiyle sonuç gerçek değerlere dönüştürülecektir.

! BOYUT ANALİZİNİN YAPILDIĞI KISIM

!***** }

STLA[k,i]:=STLA[k,i]+LA[i,j];

STMA[k,i]:=STMA[k,i]+MA[i,j];

STUA[k,i]:=STUA[k,i]+UA[i,j];

TOPLA:=TOPLA+LA[i,j];

TOPMA:=TOPMA+MA[i,j];

TOPUA:=TOPUA+UA[i,j];

// memo2.Lines.Add(inttostr(k)+'Kriterden '+inttostr(i)+' Adayla '+inttostr(j)+'
Adayın Karşılaştırması = [' +floattostr(LA[i,j])+' , '+floattostr(MA[i,j])+' ,
' +floattostr(UA[i,j])+'']);

end; //j döngüsü sonu

for i:=1 to adsay do

begin

SLA[i]:=STLA[k,i]*(1/TOPUA);

SMA[i]:=STMA[k,i]*(1/TOPMA);

SUA[i]:=STUA[k,i]*(1/TOPLA);

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
//                                memo2.Lines.Add(inttostr(i)+'Adayın                                BA=
('+floattostr(stla[k,i])+',' +floattostr(stma[k,i])+',' +floattostr(stua[k,i])+')+'*'+(1/' +floa
ttostr(topua)+',1/' +floattostr(topma)+',1/' +floattostr(topla)+'))');
    end;
{
    !*****
*****
    !ADAYLARIN BİRBİRİNE GÖRE MİNİMUM ÜSTÜNLÜKLERİNİN
HESAPLANDIĞI KISIM
    !*****
***** }
    topek:=0;
for i:=1 to adsay do
begin
    ek[i]:=1;
    for j:=1 to adsay do
    begin
        if (i=j) then
            continue;
        if (SMA[i]>=SMA[j]) then
            V:=1
        else if (SLA[j]>=SUA[i]) then
            V:=0
        else
            V:=(SLA[j]-SUA[i])/((SMA[i]-SUA[i])-(SMA[j]-SLA[j]));
        if (V<ek[i]) then
            ek[i]:=V;
    end;
end;
```


EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
{
    !*****
*****
    !*****DURULAMA BİRİMİ
*****

    print *, ' V değerlerinin en küçükleri ==> ', ek(i)
    !read *   }
    topek:=toppek+ek[i];
end;
eb:=0;
for i:=1 to adsay do
    begin
        WA[k,i]:=(ek[i]/toppek)+(fahpr[i,k]/topahppuan[k]);
        if WA[k,i]>eb then
            eb:=WA[k,i];
        end;
    for i:=1 to adsay do
        begin
            WA[k,i]:=WA[k,i]/eb;
            memo2.Lines.Add("");
            if k=1 then
                memo2.Lines.Add('Anlama Anlatma Yeteneği Kriterinden '+inttostr(i)+'
Adayın Ağırlığı ==> '+floattostr(WA[k,i]));
            if k=2 then
                memo2.Lines.Add('Genel Görünüş Kriterinin Ağırlığı '+inttostr(i)+' Adayın
Ağırlığı ==> '+floattostr(WA[k,i]));
            if k=3 then
                memo2.Lines.Add('Psikolojik Yapı Kriterinin Ağırlığı '+inttostr(i)+' Adayın
Ağırlığı ==> '+floattostr(WA[k,i]));
```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

if k=4 then
    memo2.Lines.Add('Ailevi ve Sosyal Durum Kriterinin Ağırlığı
'+inttostr(i)+' Adayın Ağırlığı ==> '+floattostr(WA[k,i]));
    if k=5 then
        memo2.Lines.Add('Bilimsel Yeterlilik Kriterinin Ağırlığı '+inttostr(i)+'
Adayın Ağırlığı ==> '+floattostr(WA[k,i]));
    end;
//showmessage('KRİTER NO = '+inttostr(k));
end;// ! K DÖNGÜSÜNÜN SONU*****
//
!*****
*****
eb:=0;
memo2.Lines.Add("");
memo2.Lines.Add('*****');
);
memo2.Lines.Add('*****ADAYLARIN NİHAİ
PUANLARI*****');
memo2.Lines.Add('*****');
);
for i:=1 to adsay do
begin
for j:=1 to 5 do
FAHPPUAN[i]:=FAHPPUAN[i]+(WK[j]*WA[j,i]);
if FAHPPUAN[i]>eb then
eb:=FAHPPUAN[i];
end;

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

        for i:=1 to adsay do
        begin
            FAHPPUAN[i]:=FAHPPUAN[i]/eb;
            memo2.Lines.Add("");
            memo2.Lines.Add('BAHS yöntemine göre '+inttostr(i)+' . adayın puanı==>
'+floattostr(fahppuan[i]));
        end;
        for i:=1 to adsay-1 do
        for j:=i+1 to adsay do
            if fahppuan[i]<fahppuan[j] then
            begin
                tut:=fahppuan[i];
                fahppuan[i]:=fahppuan[j];
                fahppuan[j]:=tut;

                tutads:=adayads[i];
                adayads[i]:=adayads[j];
                adayads[j]:=tutads;
                //sirano[i]:=j;
                //sirano[j]:=i;
            end;
            memo2.Lines.Add("");
        for i:=1 to adsay do
            memo2.Lines.Add(adayads[i]+' BAHS yöntemine göre '+inttostr(i)+' . olmuştur
            adayın puanı==> '+floattostr(fahppuan[i])+' dir');
            bitbtn6.Enabled:=false;
        end;

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

procedure TForm1.Button1Click(Sender: TObject);
var
i,j:integer;
begin
StringGrid1.RowCount:=6;
for i:=1 to adsay do
begin
StringGrid1.Rows[i].Clear;
fahppuan[i]:=0;
end;
for i:=1 to adsay do
for j:=1 to 5 do
fahpr[i,j]:=0;
adayno:=1;
edit1.Text:="";
edit2.Text:="";
edit3.Text:="";
edit4.Text:="";
edit5.Text:="";
edit6.Text:="";
bitbtn1.Enabled:=true;
bitbtn6.Enabled:=true;
memo2.Lines.Clear;
bitbtn1.Caption:='2 . ADAYA GEÇ';
edit6.SetFocus;
end;
procedure TForm1.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

edit2.SetFocus;
end;
procedure TForm1.Edit2KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
edit3.SetFocus;
end;
procedure TForm1.Edit3KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
edit4.SetFocus;
end;
procedure TForm1.Edit4KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
edit5.SetFocus;
end;
procedure TForm1.Edit5KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
bitbtn1.SetFocus;
end;
procedure TForm1.BitBtn8Click(Sender: TObject);
begin
// if SaveDialog1.Execute then
//dbmemo1.Lines.SaveToFile('c:\BAHP_Prog\deneme.doc');
memo4.Lines.add('*****');
memo4.Lines.add('*****KRİTER AĞIRLIKLARI*****');
memo4.Lines.add('*****');

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```

memo4.Lines.Add(memo3.Text);
memo4.Lines.add('*****');
memo4.Lines.add('*****ADAY AĞIRLIKLARI*****');
memo4.Lines.add('*****');
memo4.Lines.Add(memo2.Text);
if SaveDialog1.Execute then
memo4.Lines.SaveToFile(SaveDialog1.FileName);
end;
procedure TForm1.Edit7KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
edit8.SetFocus;
end;
procedure TForm1.Edit8KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
edit9.SetFocus;
end;
procedure TForm1.Edit9KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
edit10.SetFocus;
end;
procedure TForm1.Edit10KeyPress(Sender: TObject; var Key: Char);
begin
if key=#13 then
edit11.SetFocus;
end;

```

EK-2(Devam) “Bahs.dpr” isimli simülasyon programının ekran görüntüleri ve kodları

```
procedure TForm1.Edit11KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then
    memo1.SetFocus;
  end;
procedure TForm1.Edit6KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then
    edit13.SetFocus;
  end;
procedure TForm1.Edit13KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then
    edit1.SetFocus;
  end;
procedure TForm1.BitBtn3Click(Sender: TObject);
begin
  Application.Terminate;
end;
end.
```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : ERÜMİT, Ali Kürşat
 Uyuğu : T.C.
 Doğum tarihi ve yeri : 27.12.1982 Samsun
 Medeni hali : Bekar
 Telefon : 0 (362) 437 18 09
 Faks : 0 (362) 230 09 98
 e-mail : kursaterumit@gmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Lisans	Gazi Üniversitesi/Bilgisayar Eğitimi Bölümü	2003
Lise	Atakum Endüstri Meslek Lisesi	1999

İş Deneyimi

Yıl	Yer	Görev
2001-2003	EKOSAN Bilgisayar	Bilgisayar Teknisyeni
2003-2004	İlyasköy Türk-iş İlköğretim Okulu	Bilgisayar Öğretmeni
2004-	Samsun Kız Meslek Lisesi	Bilgisayar Öğretmeni

Yabancı Dil

İngilizce

Hobiler

Bilgisayar Teknolojileri, basketbol