

**T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

WEB TABANLI AKILLI ÖĞRETİM TEKNİĞİ GELİŞTİRME

**DOKTORA TEZİ
Müzeyyen BULUT ÖZEK
(04113205)**

**Tezin Enstitüye Verildiği Tarih: 11 Ağustos 2010
Tezin Savunulduğu Tarih: 07 Eylül 2010**

**Tez Danışmanı: Yrd. Doç. Dr. Ahmet ORHAN
Diğer Jüri Üyeleri: Prof. Dr. Hanifi GÜLDEMİR
Prof. Dr. Erhan AKIN
Doç. Dr. Serdar ETHEM HAMAMCI
Yrd. Doç. Dr. Arif GÜLTEN**

AĞUSTOS, 2010

**T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

WEB TABANLI AKILLI ÖĞRETİM TEKNİĞİ GELİŞTİRME

**DOKTORA TEZİ
Müzeyyen BULUT ÖZEK
(04113205)**

**Anabilim Dalı: Elektrik-Elektronik Mühendisliği
Programı : Elektrik Makinaları**

**Tez Danışmanı: Yrd. Doç. Dr. Ahmet ORHAN
Prof. Dr. Z. Hakan AKPOLAT**

Tezin Enstitüye Verildiği Tarih: 11 Ağustos 2010

AĞUSTOS, 2010

ÖNSÖZ

Doktora tez çalışmamın her aşamasında ilgi ve yardımlarını esirgemeyen danışman hocalarım Sayın Yrd. Doç. Dr. Ahmet ORHAN'a, Sayın Prof. Dr. Z. Hakan AKPOLAT'a, ders notlarından yararlanma imkanı sağlayan Sayın Prof. Dr. Muammer GÖKBULUT'a, bugüne kadar maddi ve manevi imkânlarıyla beni destekleyen eşim Cebeli ÖZEK'e ve desteğini ve yardımını esirgemeyen Resul DAŞ'a teşekkürü bir borç bilirim.

Müzeyyen BULUT ÖZEK
ELAĞIĞ-2010

İÇİNDEKİLER

Sayfa No

ÖNSÖZ.....	II
İÇİNDEKİLER.....	III
ÖZET.....	V
SUMMARY.....	VI
ŞEKİLLER LİSTESİ.....	VII
TABLOLAR LİSTESİ.....	IX
SEMBOLLER LİSTESİ.....	X
1. GİRİŞ.....	1
1.1 Tez Çalışmasının Amacı.....	2
1.2 Literatür Taraması ve Değerlendirmesi.....	2
1.3 Tezin Organizasyonu ve Katkıları.....	5
2. UZMAN SİSTEMLER.....	6
2.1 Uzman Sistemlerin Yararları Ve Kullanım Alanları.....	8
2.2 Uzman Sistemlerin Yapısı.....	12
3. UZMAN SİSTEM TASARIM YÖNTEMLERİ.....	15
3.1 Uzman Sistemlerin Tasarımında Heuristik Yöntemin Kullanılması.....	17
3.2 Kurallara Dayalı Uzman Sistemlerin Tasarlanması.....	18
3.2.1 Geriye Zincirleme.....	18
3.2.2 İleriye Zincirleme.....	19
3.3 Olasılık Teorisi, Bulanık Mantık ve Yapay Sinir Ağları Yöntemi ile Uzman sistemlerin Tasarlanması.....	21
3.3.1 Olasılık Teorisine Dayanan Uzman Sistem Tasarlanması.....	21
3.3.1.1 Bayes Olasılığı.....	23
3.3.2 Bulanık Mantık Ve Bilgi Sistemlerinde Kullanılması.....	24
3.3.2.1 Bulanık Mantık Kavramları.....	25
3.3.2.2 Bulanık Uzman Sistemler.....	28
3.3.3 Uzman Sistemler ve Yapay Sinir Ağları.....	32
3.3.3.1 Uzman Sistemler ile Yapay Sinir Ağlarının Karşılaştırılması.....	33
3.3.3.2 Uzman Sinir Ağları.....	34
3.3.4 Bulanık Sinir Ağları.....	38
3.4 Uzman Sistem Tasarımında Kullanılan Diller, Kabuklar Ve Araçlar.....	40
4. GELİŞTİRİLEN WEB TABANLI UZMAN SİSTEMİN GENEL YAPISI VE TASARIMI.....	43
4.1 Geliştirilen Sistemin Genel Amacı.....	43
4.2 Geliştirilen Sistemin Mimari Yapısı.....	44
4.2.1 Bilgi Anabirimi.....	45
4.2.2 Çıkarım Mekanizması.....	47
4.2.3 Kullanıcı Arabirimi.....	48
4.2.4 Öğrenci Modelleme Birimi.....	48
4.3 Geliştirilen Sistemde Kullanılan Yazılımlar ve Teknolojiler.....	55
4.3.1 Fuzzy2 Toolbox.....	55

4.3.1.1	Tip-2 Bulanık Mantık.....	56
4.3.2	MATLAB Web Sunucu.....	60
4.3.2.1	Uygulama Gereksinimleri.....	61
4.3.2.2	MWS Bileşenleri.....	62
4.3.3	PHP, MYSQL.....	64
4.4	Geliştirilen Yazılımın Modülleri.....	64
4.4.1	Sisteme Giriş Modülü.....	65
4.4.2	Sistemin İşleyişi.....	65
4.4.3	Bireysel Raporlar.....	70
4.4.4	Öğrenme Stili Belirleme Testi.....	71
4.4.5	Sanal Deneyler.....	72
4.4.6	Forum.....	73
4.4.7	Zeki Oyunlar.....	74
4.5	Geliştirilen Yazılımın Öğrenciler Üzerindeki Uygulamaları.....	75
4.5.1	Uygulama ve Sonuçlar.....	75
4.5.2	Uygulama Başarısının Değerlendirilmesi.....	76
5.	SONUÇ VE DEĞERLENDİRME.....	79
5.1	Sonuçların Değerlendirilmesi.....	80
5.2	Öneriler.....	82
KAYNAKLAR.....		84
ÖZGEÇMİŞ.....		93

ÖZET

Geçmişten günümüze sürekli olarak gelişen ve değişen bilişim teknolojileri, günlük hayatın her alanının önemli ve vazgeçilmez bir parçası haline gelmiştir. Bilgisayarlar eğitim alanında da çeşitli amaçlarla kullanılmaktadır. Eğitim alanında bilgisayarların en yaygın kullanım şekli internettir. Bilgisayarı ve interneti bir araya getiren öğrencilerin öğrenme sürecinde daha etkin olmasını sağlayan eğitimsel amaçlara uygun olarak hazırlanan sistemler, web tabanlı eğitim sistemleri olarak bilinmektedir.

Kişisel uyarlanabilir öğrenmeyi hedefleyen web tabanlı akıllı öğretim sistemleri, bu hedefi gerçekleştirmek için öğrencinin konu hakkında ne kadar bilgisi olduğunu ve başarılı olmak için ne gibi hedefleri olduğunu öğrenmeye ihtiyaç duyar. Bu nedenle sistem, öğrenme stilini belirlemek için öğrenciyi gözlemler. Ancak, öğrenci çeşitli şekillerde akıl yürütebilir ve düşünme tarzını açıkça göstermeyebilir. Bu yüzden öğrenci modeli, büyük oranda belirsizlik içerir.

Bu tez çalışmasında geliştirilen yazılımda, öğrenme belirsizliklerini en aza indirgeyip verimli ve etkili bir akıllı öğretim sistemi oluşturmak amacıyla yapay zeka tekniklerinden tip-2 bulanık mantık kullanılmıştır. Yazılım, Type-2 Fuzzy Logic Toolbox ve öğrenci-matlab iletişimi sağlayan Matlab Web Sunucu ortamında geliştirilmiştir.

Bulanık mantığın yeni bir açılımı olan tip-2 bulanık mantık, belirsizlikleri modelleyip etkisini azaltabilir. Belirsizlikler kaybolduğunda da tip-2 bulanık mantık tip-1 bulanık mantığa indirgenebilir.

Geliştirilen yazılımın başarısını değerlendirmek amacıyla Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi bölümündeki öğrenciler üzerinde bir uygulama çalışması yapılmıştır. Çalışmadan elde edilen sonuçlara göre hazırlanan sistemin, temel bir kontrol sistemleri dersinde uygulanan geleneksel yöntemlere göre öğrenmede daha etkili olduğu tesbit edilmiştir.

Anahtar Kelimeler: Web Tabanlı Akıllı Öğretim Sistemi, Yapay Zeka, Belirsizlik, Tip-2 Bulanık Mantık

SUMMARY

Development Of A Web Based Intelligent Tutoring Technique

Informatics technologies, having constantly developed and changed from past up to present, have become an important and indispensable part in every area of daily life. Computers are also utilized in education for various purposes. One of the most common usages of computers in education is the internet. The systems which bring computer and the internet together and make students be more effective in the learning process and which are prepared in accordance with educational purposes are known as web based educational systems.

When a web based intelligent tutoring system aims to provide a personal adaptive learning opportunity, it needs to know what knowledge the student has and what goals the student is currently trying to achieve. Therefore, system observes student to predict his/her learning style while working with an intelligent tutor. However, student can follow several paths of reasoning and he/she usually does not show his/her reasoning explicitly. Therefore, student model involves a high level of uncertainty

In order to minimize learning uncertainties and create a productive and effective intelligent tutoring system, type-2 fuzzy logic, one of the artificial intelligence techniques, is used in the system developed in this study. The software have been designed on Type-2 Fuzzy Logic Toolbox and Matlab Web Server environment which communicate with students and matlab.

The expanded Fuzzy Logic, Type-2 Fuzzy Logic, is able to handle uncertainties because it can model them and minimize their effects. And, if all uncertainties disappear, Type-2 Fuzzy Logic can be reduced to Type-1 Fuzzy Logic.

An application made on the junior students in the Department of Electronics and Computer Education of Technical Education Faculty at Firat University in order to evaluate the success of developed software. According to the results obtained from the application, prepared system was determined to be more effective in learning than traditional methods applied in a basic Control System Course.

Keywords: Web-Based Intelligent Tutoring Systems, Artificial Intelligence, Uncertainty, Type-2 Fuzzy Logic

ŞEKİLLER LİSTESİ

Sayfa No

Şekil 2.1.	Uzman Sistem Elemanları ve Bilgi Akışı.....	12
Şekil 3.1.	Geriye zincirleme	19
Şekil 3.2.	İleriye zincirleme	19
Şekil 3.3.	"Boyut" dilsel değişkeni için üyelik fonksiyonları.....	25
Şekil 3.4.	Bir sinirin modeli.....	32
Şekil 3.5.	Çok katmanlı YSA modeli.....	33
Şekil 3.6.	YSA ile US'in fonksiyonel birleştirilmesi..	35
Şekil 3.7.	YSA ile US'in yapısal birleşmesi.	36
Şekil 3.8.	Kurala dayalı, sinir ağı... ..	37
Şekil 3.9.	Kurala dayalı, uzman sinir ağına kuralların eklenmesi.....	38
Şekil 3.10.	Bulanık uzman ağı modeli.....	40
Şekil 4.1.	Geliştirilen sistemin mimari yapısı.....	44
Şekil 4.2.	Konu planı.....	45
Şekil 4.3.	Uyarı mesajı.....	46
Şekil 4.4.	Sistemin Fuzzy2 Toolbox'da hazırlanmış çıkarım mekanizması.....	47
Şekil 4.5.	Kaplama Öğrenci Modeli.....	49
Şekil 4.6.	Felder Silverman öğrenme stilleri modeli.....	50
Şekil 4.7.	Fuzzy2 toolbox ana penceresi.....	55
Şekil 4.8.	Tip-1 üyelik fonksiyonları.....	57
Şekil 4.9.	Tip-2 üyelik fonksiyonu.....	58
Şekil 4.10.	Tip-2 Bulanık Mantık Sistem Yapısı.....	59
Şekil 4.11.	Basit bir MWS uygulaması için ağ yapısı.....	60
Şekil 4.12.	İş yükünün yoğun olduğu MWS uygulamaları için ağ yapısı	61
Şekil 4.13.	MWS modeli.....	62
Şekil 4.14.	matweb.conf ayar dosyası.....	63
Şekil 4.15.	t2den.m dosyası.....	64
Şekil 4.16.	Sisteme giriş ekranı.....	65
Şekil 4.17.	Geliştirilen sistemin ana sayfası.....	66

Şekil 4.18.	Sisteme giriş ve ön stil testi bölümlerinin akış şeması.....	66
Şekil 4.19.	Sistem işleyişinin genel akış şeması.....	67
Şekil 4.20.	Konular sayfası.....	68
Şekil 4.21.	Kullanıcı arabiriminden örnek bir konu anlatım sayfası.....	69
Şekil 4.22.	Ardışık anlama yeteneğine sahip öğrenciler için kavram haritası örneği...	69
Şekil 4.23.	Motor hız kontrolü sanal deneyinin parametre değiştirme sayfası.....	70
Şekil 4.24.	Öğrenci raporları	71
Şekil 4.25.	Öğrenme stili ön testi	71
Şekil 4.26.	Sanal deneyler sayfası	72
Şekil 4.27.	Motor hız kontrolü sanal deneyinin parametre değiştirme sayfası.....	73
Şekil 4.28.	Forum sayfası	74
Şekil 4.29.	Zeki oyunlar sayfası	74

TABLolar LİSTESİ

Sayfa No

Tablo 3.1. Bulanık giriş değişkenlerinin aldığı değerler.....	28
Tablo 3.3. US ve YSA'nın Karşılık Gelen Elemanları.....	34
Tablo 3.4. US'/er ve YSA Arasındaki Farklılıklar.....	34
Tablo 4.1 Kural tablosundan bir bölüm.....	47
Tablo 4.2 Felder Silverman Öğrenme Stillerine ait baskın özellikler.....	51
Tablo 4.3 bilgi_durumu tablosuna ait alanlar.....	52
Tablo 4.4 kisiselbilgi tablosuna ait alanlar.....	53
Tablo 4.5 konuderecesi tablosuna ait alanlar.....	53
Tablo 4.6 sistemparmetre tablosuna ait alanlar.....	53
Tablo 4.7 zaman tablosuna ait alanlar.....	54
Tablo 4.8. Kontrol ve deney grubu öntest independent-samples t-testi sonuçları.....	77
Tablo 4.9. Kontrol grubu öntest ve sontest ortalama puanlarının paired-samples t-testi sonuçları.....	77
Tablo 4.10. Deney grubu öntest ve sontest ortalama puanlarının paired-samples t-testi sonuçları.....	77
Tablo 4.11. Sontest puanlarının gruplara göre betimsel istatistikleri.....	78
Tablo 4.12. Önteste göre düzeltilmiş sontest puanlarının yönteme göre ANCOVA sonuçları.....	78

SEMBOLLER LİSTESİ

$\tilde{A}$	:	$\tilde{A}$ tip-2 bulanık kümesi
AÖS	:	Akıllı Öğretim Sistemi
$\tilde{B}$	:	$\tilde{B}$ tip-2 bulanık kümesi
BSA	:	Bulanık Sinir Ağları
BUS	:	Bulanık Uzman Sistem
C_A	:	A tip-1 bulanık kümesinin ağırlık merkezi
$C_{\tilde{A}}$	:	$\tilde{A}$ tip-2 bulanık kümesinin ağırlık merkezi
f^{cg}	:	İkinci derece üyelik fonksiyonunun ağırlık merkezi
MWS	:	Matlab Web Server
US	:	Uzman Sistem
VRML	:	Virtual Reality Modeling Language
$Y_c(x)$	:	Ağırlık merkezi yöntemini kullanan tip-indirgeyicinin çıkışı
$Y_a(x)$	:	Merkezlerin toplamı yöntemini kullanan tip-indirgeyicinin çıkışı
$Y_h(x)$	:	Yükseklik yöntemini kullanan tip-indirgeyicinin çıkışı
$Y_{mh}(x)$	:	Değiştirilmiş yükseklik yöntemini kullanan tip-indirgeyicinin çıkışı
$Y_{cos}(x)$	:	Kümelerin merkezi yöntemini kullanan tip-indirgeyicinin çıkışı
$Y_{TSK,2}(x)$	:	Sugeno sistemlerin çıkışı
YSA	:	Yapay Sinir Ağları
YZ	:	Yapay Zeka
μ	:	Üyelik fonksiyonu
$\bar{\mu}$	:	Üst üyelik fonksiyonu
$\underline{\mu}$	:	Alt üyelik fonksiyonu
*	:	t-norm işlemi
$\neg$	:	Olumsuzluk (negation) işlemi
Π	:	Buluşma (meet) işlemi
$\sqcup$	:	Katılma (join) işlemi

1.GİRİŞ

Bilişim teknolojisindeki hızlı gelişmeler bilginin paylaşımını ve erişimini oldukça kolaylaştırmıştır. Bilgisayarlar artık hayatımızın vazgeçilmez bir parçası haline gelmiştir. Her alanda kullanılan bilgisayarlardan eğitim alanında da çeşitli amaçlarla yararlanıldığı görülmektedir. Eğitim alanında bilgisayarların en yaygın kullanım şekillerinden biri internettir. Bilgisayarı ve interneti bir araya getiren, öğrencilerin öğrenme sürecinde daha etkin olmasını sağlayan, eğitimsel amaçlara uygun olarak hazırlanan sistemler, web tabanlı eğitim sistemleri olarak bilinmektedir. Bu sistemlerin eğitimsel açıdan en büyük faydası, sınıf ortamından bağımsız ve bireysel öğretime uygun olmalarıdır. Bununla birlikte öğretim koşulları ve çevre düzenlemesi yapılırken öğrencilerin bireysel farklılıkları ve öğrenme yeteneklerine uygun bir düzenleme yapma ihtiyacı doğmuştur. Bu özellik dikkate alınarak yapay zeka ve bilgisayar teknolojilerinin kullanılması ile Akıllı Öğretim Sistemleri (AÖS) (Intelligent Tutoring Systems) oluşturulmuştur.

Sürekli gelişen AÖS, yapıları ve amaçlarına göre pek çok araştırmacının ilgilendiği bir alan olmuştur [1]. “İleri Öğrenme Teknolojileri (Advanced Learning Technologies)” başlığı altında incelenen bu uzman sistemler; neyi öğreteceğini, kime öğreteceğini ve nasıl öğreteceğini bilerek yapay zeka ortak oluşumunda yer alan tekniklerden yararlanarak tasarlanmıştır.

Klasik Bilgisayar Destekli Öğretim sistemlerinden farklı olarak bu uzman istemler, öğrenciye bilgi düzeyine uygun öğrenme ortamı sunarak öğrencinin daha kolay, hızlı ve kalıcı öğrenmesini sağlamayı amaçlar. Carnegie-Mellon Üniversitesi’nde yapılan bir araştırmaya göre bu sistemlerin öğrenme kalitesini %43 arttırdığı ve öğrenme süresini %30 düşürdüğü tespit edilmiştir [2]. Ancak bu sistemler, öğretim için iyi sunulmuş bir çözüm olmakla beraber gerçekleşmesi zaman alan karmaşık yapılardır. Planlama ve gerçekleştirme sürecinde yapay zeka tekniklerinin, bilgisayar teknolojilerinin ve öğretim teknolojilerinin bir arada kullanılması gereklidir. Sistemi oluşturmadaki temel problem, öğrencinin bilgi seviyesine uygun kişisel bir gezinme desteğinin nasıl verileceğidir. Diğer bir deyişle, bu tip öğretim sistemlerinin temel problemi; öğrencinin bilgi seviyesine uygun olarak sistemin alan bilgisinin nasıl özelleştirileceğidir.

1.1. Tez Çalışmasının Amacı

İnsanlar farklı yollarla öğrenirler. Bazıları bilgiyi dinleyerek kavrarken diğerleri okuyarak, mantık yürüterek veya pratik yaparak öğrenir. Öğrenmenin bu farklı yolları, “öğrenme stilleri” olarak adlandırılmış ve sayısız farklı biçimde araştırılarak modellenmiştir. Kişiler farklı öğrenme konularına yönelik olarak farklı yaklaşımlar izleyebilirler ve gösterdikleri bu yaklaşımlar kişisel öğrenme stilini belirler. Öğrencilerin başarısını etkileyen en önemli etken, onların beklentileri ya da yaklaşımları ile onlara sunulan yöntemler arasındaki örtüşme ile yakından ilişkilidir. Bu çalışmada öğrenciye özgü içerik sağlayan ve kendini sürekli güncelleyebilen bir akıllı öğretim sistemi oluşturulmuştur. Çalışmanın temel amacı, kişisel öğrenme stilleri araştırılarak oluşturulan akıllı öğretim sistemi ile başarı oranını yükseltmek ve yeni araştırmalara ışık tutmaktır.

1.2 Literatür Taraması ve Değerlendirmesi

Literatürde, akıllı öğretim sistemleri konusunda birçok çalışma yapıldığı ve farklı yaklaşımlar sunulduğu görülmektedir. Carbonell’in 1970 yılındaki bir çalışması olan SCHOLAR akıllı öğretim alanında ilk öncü sayılabilecek programdır. SCHOLAR’a dahil edilen bazı önemli özellikler;

- Olguların, içeriklerin ve yordamların oluşturduğu bir ağ yapısı içerisinde karmaşık ancak iyi tanımlanmış bir veritabanı yapısı,
 - Bilgi-tanımlama kelimeleri içeren bir ağ ve çok-dallı ağaç yapısında olaylar,
 - Sokratal öğretim diyalogu,
 - Cevaplar için sonuç çıkarım stratejileri anlamsal ağın içeriğinden bağımsız olması
- şeklinde sıralanabilir. SCHOLAR, öğrencilere Güney Amerika coğrafyası’nı öğretmeyi amaçlayan kimi zaman öğrencinin programa kimi zaman ise programın öğrenciye sorular sorduğu bir uzman sistemdir [3]. SCHOLAR’ın uzman bilgisi modülü, anlamsal bir ağ formunda Güney Amerika coğrafyasını barındırmaktadır. Bu yapı, sorulara izin vermeyi cevapların saklanması gerek duymadan sağlayan yapısından dolayı sistemi bellek problemlerinden kurtarmaktadır.

Bu gelişimlerin yanı sıra, SCHOLAR basit olarak bu alanda ilk sistemlerden biri olduğu için belirli bir takım dezavantaj ve güçsüz yönleri sahiptir. Bu dezavantajlar, öğretim stratejisinin ve dil işleme yeteneğinin kısıtlı olmasında kendilerini göstermektedir.

SCHOLAR'ın eksikleri olsa da akıllı öğretim sistemleri dizaynı ve uygulamaları için yaşamsal pek çok yenilik getirmiştir.

BUGGY, matematikte öğrencinin doğru olmayan yaklaşımlarını ortaya çıkarmak için geliştirilmiş olup, öğrenciye neden yanlış yaptığını anlatabilme kapasitesine sahiptir [4]. Deneyler, herhangi bir problemin çözülmesinden ziyade yanlış yaklaşım modellerinin oluşturulmasının daha zor olduğunu göstermiştir. Aynı zamanda sistem, öğretmene öğrencilerin hatalarını tespit etme imkanı tanır.

PROUST, programcılığa yeni başlayanlar tarafından yazılan Pascal programlarındaki yazımsal olmayan hataları bulmak üzere tasarlanan bilgi tabanlı bir sistem olup, her çeşit hatayı bulur [5]. Buna ek olarak, hataların nasıl ortadan kaldırılacağını belirler. Sentezle-analiz yaklaşımını kullanarak programları analiz eder. Daha önce belirlenmiş olan program gereksinimlerini yerine getirebilecek metotları bulmak için inceler. Daha sonra sistem, mümkün olan her metodu programcının metoduyla karşılaştırır ve bu programcılık bilgisine gereksinim gösterir. Tecrübesiz programcılarca yazılmış olan kod içerisindeki pek çok hatayı da bulmayı becerir. Bu uzman sistem üzerine kurulabilecek diğer bir mantıklı adım da, sadece öğrencinin hatalarını düzelterken değil, aynı zamanda gereksinim duyulduğunda öğrenciye pratik kazandırmak için gerekli örnekleri sağlayabilecek bir otomatikleştirilmiş programlama kursu olmasıdır.

WEST, öğrencilere “How the West was won?” (“Batı nasıl kazanıldı?”) adı verilen, bir oyun oynatılarak öğretimin gerçekleştirilmesini amaçlayan bir çalıştırıcı programın ilk örneklerinden biridir [6]. Oyun, her bir oyuncu tarafından üç anahtar sayı üzerinde dört işlemden yararlanılarak değişiklik yapılmasına dayanmaktadır. Bu sistem, akıllı öğretim sisteminin farklı modüllerinin çok farklı sistemlerde kullanılabileceğini mükemmel bir şekilde vurgulayan bir programdır. Bu yüzden, ortaya koyduğu resmi olmayan öğretim atmosferine bağlı olarak bir öğreticiden çok “çalıştırıcı” olarak adlandırılmaktadır.

WUMPUS'de WEST gibi öğrencinin; mantık, ihmal teorisini, karar kabul etme teorisi ve geometri gibi matematik alanlarını öğrenmesi için oyun kurallarını kullanır [7]. Sistem; uzman, psikolog, öğrenci modeli, ve eğitimci olmak üzere dört modülden meydana gelir. Bu sistemin öğretim sürecinde faydalı sonuçlar verdiği gözlenmiştir.

Turan öğrenci modeli oluştururken Stereotip Öğrenci Modelini tercih etmiştir [8]. Bu modelde, öğrencileri farklı seviyelere ayırarak seviyelerine uygun içerik ve soruların verilmesini sağlamıştır. Öğrencinin ön test ile elde ettiği sonucun yanında o derste aldığı

notların ortalamasını öğrencinin seviyesini belirlemede kullanmıştır. Bu sayede öğrencinin seviyesini belirlemedeki eksiklikleri gidermeye çalışmıştır.

Büğrü öğrenci modeli oluşturmada Bayesian modeli ile örtüşme modelini kullanmıştır [9]. Sunulacak etkinlik seçiminde ise Bayesian Karar Ağları kullanımını benimsemiştir.

Öncü hazırladığı çalışmada kural tabanlı uzman sistem yaklaşımı ile öğretim değerlendirme sistemi geliştirmiştir. Geliştirdiği bu sistem için model ve program tasarlamıştır [10, 11].

Doğan çalışmasında, akıllı öğretim sisteminden alınan verilerde veri madenciliği tekniklerinden tek ve çok boyutlu birliktelik kuralı analizi, kümeleme analizinin yapılmasını ve kümeleme analizi sonuçlarının görselleştirilmesini sağlayarak bu alana katkı sağlamıştır. [12].

Kaya geliştirdiği ExcelTUTOR yazılımı ile bilgisayar destekli öğretimin dezavantajlarına karşı kişiselleştirilmiş acil geribildirimler, isteğe bağlı ipuçları ve açıklamalar sunmaktadır [13]. Öğrencinin kendi hızında öğrenmesine olanak sağlayan bu yazılım, öğrenciye tekrar ve alıştırma yapma olanağı sunan, öğrenci ve öğretmene öğrenci performansıyla ilgili değerlendirme ve öneriler sunabilen bir yapıya sahiptir.

Luan çalışmasında, WebCT yazılım aracı tarafından üretilen Web kullanıcı erişim kütük bilgilerinin ne tür önışlemlerden geçirildiği ve makine öğrenme algoritmalarından biri olan ikili karar ağacı kullanılarak öğrenci başarımlarını tahmin etmeye yönelik modelin nasıl kurduğunu anlatmaktadır [14]. Sonuçta geliştirilen sistem sayesinde bir öğrencinin bir haftalık WebCT materyali üzerindeki hareketlerine bakılarak dönem sonundaki başarı notunu %70 olasılıkla tahmin etmek mümkün hale gelmektedir.

Yoshikawa ve diğerleri üniversite öğrencilerinin elektrik devrelerini öğrenmelerine yardımcı olmak amacıyla bir uzman öğretim sistemi geliştirmiştir [15]. Öğrenci modeli oluştururken hata ve kaplama modellerinden yararlanmıştır.

Negnevitsky güç sistemlerinde hata analici için bir sistem geliştirmiştir [16]. Geliştirdiği uzman sistemi test etme ve oluşturma aşamasında Leonardo Uzman sistem Kabuğunu kullanmıştır.

Brusilovsky ve diğerleri öğrenciye sunulan bağlantıları düzenlerken doğrudan rehberlik, bağlantı sıralaması, bağlantı gizlemesi, bağlantı notu ve harita adaptasyonu tekniklerini kullanmıştır.[17-19].

Gürbüz kavramsal modelleme yapılabilen ve kullanıcıyı yönlendirebilen yapay zekaya dayalı kavramsal model geliştiren bir yazılım geliştirmiştir [20]. In-CM olarak adlandırdığı kavramsal model geliştirme aracını uzman sistem olarak tasarlamıştır.

Dağ Visual Prolog programlama dilini kullanarak hem öğrenci hem de öğretmen arabirimlerine sahip birden fazla ders için kullanılabilecek akıllı öğretim sisteminin alt yapısını oluşturmuştur [21].

Yapılan literatür araştırması sonucunda web tabanlı akıllı öğretim sistemlerinde öğrenci modeli oluşturulurken veri madenciliği, bayes ağları, bulanık mantık vb. yapay zeka yöntemlerinin kullanıldığı görülmüştür [22-40]. Bu yöntemler içerisinde yer almayan Tip-2 Bulanık Mantığın öğrenci modeli oluşturulurken meydana gelen belirsizlikleri modellemede etkin olacağı beklenmektedir.

1.3 Tezin Organizasyonu ve Katkıları

Tez çalışmasının birinci bölümünde, teze genel bir bakış açısı kazandırılmaya yönelik temel bilgiler verilmiştir. Diğer bölümlerin organizasyonu ile birlikte tezde yapılan katkılar aşağıda sunulmaktadır:

Bölüm 2’de, uzman sistem ve yapısı konusu ayrıntılı olarak incelenip, çalışma ve uygulama alanları üzerinde geniş bir şekilde durulmuştur

Bölüm 3’de, literatürde geçen uzman sistem tasarım yöntemleri açıklanmıştır.

Bölüm 4’de, Kontrol Sistemleri dersi için geliştirilen web tabanlı uzman sistemin yapısı, bileşenleri, sistemde kullanılan yazılım ve teknolojiler detaylı olarak verilmektedir. Geliştirilen sistemin başarısı, Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümü öğrencileri üzerinde uygulanarak test edilmiştir. Elde edilen sonuçlar, ayrıntılı olarak irdelenmiştir.

Bölüm 5’de, tezin sonuçları irdelenmiş ve orijinal katkılar vurgulanmıştır. Ayrıca ileriye dönük uygulama alanları ve öneriler tartışılmıştır.

2. UZMAN SİSTEMLER

Uzman sistemler, insan tarafından yapılan işlerin bilgisayarlara daha iyi nasıl yaptırılacağına araştırmasını yapan yapay zeka programlama tekniklerinin bir dalıdır [41]. Genellikle bilgi ve çıkarıma dayanan bu bilgisayar programları, konusunda uzmanlaşmış insanların üstlendiği zor bir görevi gerçekleştirmek için oluşturulmuşlardır. Uzman bir insan belli bir alanda yeterli bilgiye, uzman sistem de belli bir alanla ilgili bilgilerden oluşan veri tabanına sahiptir. Uzman sistemler, uzman insanlar gibi sahip oldukları bilgilere dayanarak mantıksal çıkarım yaparak sonuca ulaşırlar. Bir uzman sistemin asıl gücü, algoritma ve belirli sonuca varma metotlarını kullanırken içerdiği bilgidir.

Uzman sistem, veri işleme sürecinden bilgi işleme sürecine geçiş olarak ifade edilebilir. Veri işleme sürecinde veri tabanı bir algoritmaya bağlı olarak işlenirken, bilgi işleme sürecinde bilgi tabanı herhangi bir algoritmaya bağlı kalmaz. Başka bir deyişle, algoritmalarla sonuç çıkarma mekanizmaları (inference engine) yer değiştirmiştir. Yani:

Geleneksel programlar => Algoritma + Veritabanı;

Uzman Sistemler => Çıkarım Mekanizması + Bilgi tabanı.

Çeşitli kurum ve kişilerce uzman sistemlerin farklı tanımlarının yapılması bir kavram kargaşasına neden olmaktadır. Tanımlamalarda en sık yapılan hatalardan biri uzman sistemler ile bilgi tabanlı sistemlerin karıştırılmasıdır. Bilgi tabanlı sistemler, bilgisayara girilmiş bilgi yardımıyla ve akıl yürütme işlemiyle zor problemleri çözerler. Ancak bu sistemler, uzman sistemlerden çözdükleri problemlerin küçük boyutlu ve daha sınırlı olması yönüyle ayrılırlar. Bir başka deyişle, uzman sistemler gerçek uzmanlığı gerektiren karmaşık bilgileri içerirler. Kitap, dergi gibi dokümanlarda bulunmayan bu tür bilgiler, bireylerin ancak uzman oluncaya kadar yıllar süren deneyimleriyle elde edilir. Oysaki bilgi tabanlı sistemler, yalnızca yayınlarda bulunan bilgilerle oluşturulur [42]. Uzman sistemler ile karıştırılan başka bir sistem de karar destek sistemleridir. Uzman sistemler belli bir problemle ilgili kesin kararı, sonucu verirler. Halbuki karar destek sistemleri kesin kararı veremezler. Yalnızca kararın verilmesine yardımcı olacak bilgiyi üretirler. Ayrıca uzman sistemler karar destek sistemlerinin çoğunlukla yaptığı gibi verilen girişleri kabul etmek yerine sorular sorarak gerek duyduğu bilgiyi alma yoluna giderler. Ancak üretilen her uzman sistem başarılı olamaz ve yapılan çalışmaların boşa gitmemesi için bu tür uzman

sistem denemeleri bir kenara atılmak yerine karar destek sistemi olarak kullanılabilir. Uzman sistemlerin sözü edilen sistemlerden en önemli farkı da verdikleri kararları ve nedenlerini açıklayabilmeleri, kesin olmayan eksik bilgileri değerlendirebilmeleridir [42].

Uzman sistemler bir uzmanın yapabileceği çeşitli aktiviteleri yerine getirebilmek amacı ile hazırlanmıştır. Bu aktiviteler hastalık teşhisinden bir molekülün yapısının ortaya çıkarılmasına kadar geniş bir yelpazeyi oluşturur. Geleneksel programların tersine uzman sistem programlarında kesin ve net algoritmalar kullanılmaz. Daha çok sezgisel ve kıyaslama gibi yöntemler kullanılır. Bir uzman sistem hazırlanması iki veya daha çok yıl alabilir.

Uzman sistemlerin tasarımı karmaşık ve çok zaman alan bir iştir. Bir uzman sistemin tasarlanması genelde bir grup çalışması gerektirmektedir. Uzman sistem tasarımcılarına göre, bir uzman sistemi tasarlayan grupta en az iki bilgi mühendisi (program için uzman kişiden gereken bilgileri alan ve programcıya ileten kişi), uzman ve programcı bulunmalıdır. Üretim kurallarına dayalı bir uzman sistem prototipi tasarlamak, bilgi tabanında 50'den 100'e kadar kural olduğunda bir kaç aydan 1-2 yıla kadar sürmektedir. Kuralların sayısı 1000 ile 1500 arasında olduğunda, prototipin sanayi veya uygun alanda kullanılabilir duruma getirilmesi 2 yıldan 4 yıla kadar süre alabilmektedir. Tasarlanmış bir uzman sistemin geliştirilerek daha geniş alanda kullanılması ve kuralların sayısının 2000 ile 3000 arasında olduğu durumda, sistemin ticari ürüne dönüştürülmesi ortalama 3 ile 6 yıl zaman alabilmektedir [42]. Böylesine uzun hazırlama ve programlama süresine rağmen uzman sistemler hem araştırma objesi olarak, hem de ticari bir araç olarak yaygın kullanılmaktadır.

Uzman sistemler 70'li yıllardan beri değişik alanlarda başarıyla uygulanmaktadır. Bunlara örnek olarak MYCIN ve INTERNIST (tıbbi teşhislerde), DENDRAL ve SECS (kimyasal analizler için), XCON (bilgisayar sistemlerinin konfigürasyonunu belirlemek için) vb. verilebilir. Uzman sistemlerin ilk prototipleri olarak GBS (General Problem Solver 1969), SIR (Semantic International Retrieval 1968), MACSYMA (1968) vb. programları gösterilebilir [43].

Uzman sistem programlarının büyük çoğunluğu ABD'de tasarlanmakta ve uygulanmaktadır. Gün geçtikçe diğer ülkelerde de uzman sistemlerin geliştirilmesi yaygınlaşmaya başlamıştır. Günümüzde dünyanın hemen hemen tüm gelişmiş ülkelerinde olduğu gibi ülkemizde de uzman sistemler geliştirilerek kullanılmaktadır.

Genelde uzman sistemler statik ve dinamik olmak üzere ikiye ayrılırlar. Şimdiye kadar tasarlanmış olan uzman sistemlerin büyük çoğunluğu statik uzman sistemlerdir. Böyle uzman sistemlerin bilgi tabanı sistemin çalışması boyunca değişmemektedir. Dinamik uzman sistemlerde ise bilgi tabanı sistemin çalışması süresince çeşitli kaynaklardan gelen bilgilere göre değişebilmektedir. Dinamik bir uzman sistemin tasarlanması oldukça zor bir iştir [41].

2.1. Uzman Sistemlerin Yararları Ve Kullanım Alanları

Uzman sistemler kullanıcılara birçok alanda önemli faydalar sağlarlar. Bu faydalar;

1. **Uzmanlığın Yaygınlaştırılması:** Yeterli sayıda uzman olmayan alanlarda uzman sistem oluşturmak sıkıntıları gidermektedir. Özellikle uzmanların emeklilikten dolayı ve diğer nedenlerden ortaya çıkan işten ayrılmaların yol açtığı sorunlar azaltılmaktadır. Burada bir uzman sistemin kolayca kopyalanarak diğer uygun alanlarda kullanılması olanağı da uzman sistemlerin yaygınlaştırılabilmesi nedenidir. Oysa yeni bir uzmanın yetiştirilmesi pahalı ve zaman alan bir süreçtir.
2. **Ara Sonuçları Gösterebilme Yeteneği:** Bazen yönetilen sistemi daha iyi anlamak ve yönetmek için önce ara sonuçları görmek faydalı olmaktadır. Bir uzman sistem buna imkan vermektedir.
3. **Süreklilik:** Uzman kişilerden farklı olarak uzman sistemlerde unutkanlık yoktur ve o sürekli aynı kapasite ile kullanılabilir.
4. **Maliyet Tasarrufu:** Uzman Sistem kullanımı ile karşılaştırıldığında insanların incelemeleri daha pahalı görülmektedir. Böylece kullanıcı başına düşen uzmanlık maliyeti azalmış olur.
5. **Kalite Düzeyinin Yükselmesi:** Uzman sistemler kararlı ve birbiriyle tutarlı önerilerde bulunurlar. Kararları hızlı "düşünerek" ve acele etmeden verirler. Hata oranların çok düşük olması ve tutarlı karar verebilmesi kalitenin yükselmesine getirmektedir.
6. **Çalışılmayan Sürenin (down-time) Azalması:** Pek çok uzman sistem arızaların teşhisinde ve tamir önerileri vermede kullanılmaktadır. Bu fonksiyonun uzman sistemlerle yapılması araçların boş durmasını önemli ölçüde azaltmaktadır.
7. **Tam ve Kesin Olmayan Bilgiyle Çalışabilme Özelliği:** Diğer bilgisayar programlarından farklı olarak uzman sistemler aynı insan uzmanlar gibi soru sorma aşamasında aldıkları cevapların "bilmiyorum", "emin değilim" gibi olması halinde

de sonuç üretebilirler. Hatta kullanıcının olasılıklı cevapları karşısında olasılıklı önerilerde bulunabilirler. Belirsiz ortamlarda çalışabilmek için özel olarak geliştirilmiş yöntemler mevcuttur.

8. **Üretim Artışı:** Uzman sistemler insan uzmanlardan çok daha hızlı olduğundan dolayı az zamanda daha fazla üretim yapabilme imkanı sağlamakta yani verimliliği artırmaktadır.
9. **Eğitim Verme:** Açıklama yapma özelliği yeterli düzeyde olmayan kullanıcılara, karşılaştıkları sorunların çözümleri ile ilgili bilgiler vererek onların eğitimini sağlamaktadır. Bu özellikle öğrencilerin eğitilmesi açısından önemlidir.
10. **Sağlıklı Öneri Üretimi:** Bazı uzmanlar ne kadar yetenekli olurlarsa olsunlar zaman kısıtlı olduğunda telaşa kapılmakta ve sağlıklı değerlendirme yapma özelliklerini yitirmektedirler. Böylece yanlış sonuçlara varma ihtimalleri artmaktadır. Bir bilgisayar programı için böyle bir durum söz konusu değildir, yani uzman sistemler problem alanlarında çözüme ulaşmak için verimli bir şekilde araştırma yeteneğine sahip olmaktadır.
11. **Güvenilirlik:** Uzman sistemler bütün detayları inceledikleri için güvenilir sistemlerdir. Ayrıca yorulma, hastalanma, mazeret ileri sürerek işe gelmeme gibi problemler çıkarmazlar ve 24 saat çalışırlar. Greve gitmezler, amirlerin arkasından konuşmazlar.
12. **Geleceğin Fabrikalarının Gelişmesine Katkı:** Uzman sistemler yakın gelecekte ulaşılması beklenen otomasyona geçmiş fabrikaların yazılım yönü açısından önemli yapı taşlarından biri olarak görülmektedir. İnsansız fabrikalar ancak insan uzmanların yerini alacak uzman sistemler ile kurulabilecektir. Okunabilirliği açısından sistemin veri tabanı önemli bir husus arz etmektedir. Bir uzman sistem devamlı olarak gereken belgeleri üretebilmekte ve böylece karar verme sürecine bir başka yönden de katkıda bulunmaktadır.
13. **Sonuçlardan Yararlanma Kabiliyeti:** Sistemin çalışması sonucu elde edilen çeşitli fırsatları değerlendirecek bilgiden yararlanılabilmektedir. Bundan başka uzman sistem sistemdeki olabilecek tüm ilişkileri gözden geçirebilmektedir. İlişkilerin sayısı çok büyük olduğunda, uzman bu işlemi gerçekleştirirken zorlanmaktadır.

ana başlıkları altında toplanabilir [41, 42]. Uzman sistemlerin yararları için daha birçok madde eklenebilir. Ancak tüm bu olumlu özelliklerin yanında uzman sistemlerin çıkardığı

sorunlar ve kullanım kısıtlamaları da vardır. İleriye yönelik büyük umutlar beslenmesine karşın bugünkü uzman sistem teknolojisi ve metotlarıyla yalnızca karmaşık olmayan uygulamalarda başarı sağlanmaktadır. Bazı uzman sistemler araştırma aracı olarak kullanılırken, bazıları önemli iş ve endüstri alanlarında kullanılmaktadır. Yabancı para değerlerinin takibi ve tahmini, yatırım danışmanlığı, kredi yönetimi ve müşteri değerlendirme, faiz karşılığında ödünç para alma işlemlerini onaylama, sigorta risklerini değerlendirme ve yatırım fırsatlarını değerlendirme gibi alanlarda uzman sistem kullanımı yaygınlaşmaktadır. En yaygın uygulama alanları;

1. **Arıza Teşhisleri ve Tamir Önerileri:** Gözlemler yaparak çeşitli makine ve sistem arızalarının nedenlerini ortaya çıkarma ve bunların giderilmesi için önerilerde bulunma.
2. **Planlama:** Özellikle proje ve işlem planlama konularında çok verimli uygulamalar. Çoğu uzman planlama sistemleri kural temelli sistemlerdir.
3. **Yorumlama:** Algılayıcılardan gelen değerlerden çıkarım yapılarak bir sonuca varılması.
4. **Kontrol:** Çeşitli üretim işlemlerinde operasyon parametrelerinin öngörülen sınırlar içinde tutulması amacıyla bu parametrelere etki yapan faktörlerin değişimlerinin kontrol altına alınması ve bu değerlerin istatistiklerinin derlenmesi.
5. **Tasarı:** Sınırlı şartları dikkate alarak çizimler üretme.

şeklinde sıralanabilir [7].

Uzman sistemlerin kullanım alanlarının genişletilmesi ve uzmanlık derecelerinin yükseltilmesi yolunda bazı engeller bulunmaktadır. Bu engellerin aşılması yolunda sürdürülen araştırmalar büyük umutlar vaat etmesine karşın henüz arzu edilen düzeye ulaşamamıştır. Engelleri;

1. **Uzmanların Bilgilerinin Elde Edilmesindeki Güçlükler:** Bu, uzman sistem tasarımında en güç olan nedenlerden en önde gelenidir. Uzmanlar sahip oldukları bilgileri çeşitli psikolojik sebepler nedeniyle vermeye yanaşmamaktadırlar. Bu sebepler işini kaybetme korkusu ve yeni teknolojiye olan şüphe olarak gösterilebilir.
2. **Uzmanlık Bilgisinin Olmaması:** Bazı konularda yetişmiş uzmanlara ulaşamamaktadır ve uzmanların çoğunun kendi bilgisini bilgi mühendisine aktarmak için zamanı olmamaktadır.

3. **Yaratıcılık:** Uzman bir kişi özellikle alışlagelmemiş durumlarda bir uzman sistemden daha yaratıcı olabilmektedir.
4. **Ortak Terminoloji Olmaması:** Pek çok uzmanın kullandığı kelimeler taşıdığı anlam bakımından uzmana özgü değildir. Başka bir deyişle uzmanların bazı mesleki kelimelere yükledikleri anlamlar, o kelimelerin bilinen genel anlamlarından farklıdır. Bu durum yanlış anlamalara neden olmaktadır.
5. **Uzman Sistemlerin Dar Alanlara Yönelik Olmaları:** Uzman sistemlerden verim alınması henüz yalnızca dar alanlarda kalmakla mümkündür. Hatta bazen bu alan çok dar olmak zorundadır. Bu alandan az da olsa dışarı çıkılması sistemin çalışmasını zorlaştırmaktadır.
6. **Üst Yönetimlerin Tutuculuğu:** Uzman sistem yazılımlarının yönetici ve patronlara kabul ettirilmesi zor olmaktadır. Burada yöneticilerin yeni teknolojiye şüphe ile bakması ve fazladan para harcamak istememeleri rol oynamaktadır. Aynı işi yapan uzman olduğu durumda neden yeni bir teknolojinin (programın) geliştirilmesine ihtiyaç olduğunun anlaşılması zor olabilir.
7. **Öğrenme:** İnsan uzman ortam değişimine çabuk uygunluk sağlamaktadır. Uzman sistemin bunu hemen yapması zordur. Yani öğrenen bir uzman sistemin tasarlanması da başlı başına bir problemdir.
8. **Aynı Konuda Uzmanların Farklı Bakış Açıklarına Sahip Olmaları:** Her uzmanın konusuna yaklaşımı doğru olmakla birlikte farklı da olabilmektedir. Yalnız bir uzmandan elde edilen bilgilere dayalı uzman sistem, sadece o uzmanın probleme olan bakış açısını sergileyecektir. Ancak farklı bir uzmanın problemin çözümüne daha ekonomik ve/veya daha doğru bir tutum sergileyebileceği unutulmamalıdır.
9. **Uzmanların Objektif ve Bağımsız Denetim Mekanizmalarının Olmaması:** Pek çok uzman kişisel yöntemler ile çalışmaktadır ve onlar bu yöntemleri açıklamakta zorluk çekmektedirler. Bu durum tasarımcılarda tam doğrulukla çalışan uzman sistem oluşturmalarının mümkün olamayacağı düşüncesine neden olmaktadır.
10. **Maliyet Yüksekliği ve Zaman Fazlalığı:** Uzman sistemlerin oluşturulması yüksek maliyetli olup uzun sürmektedir. Maliyetler çok geniş aralıkta farklı tutarlarda gerçekleşmekte, süreler ise 3 yıl ile 20 yıl arasında değişmektedir.

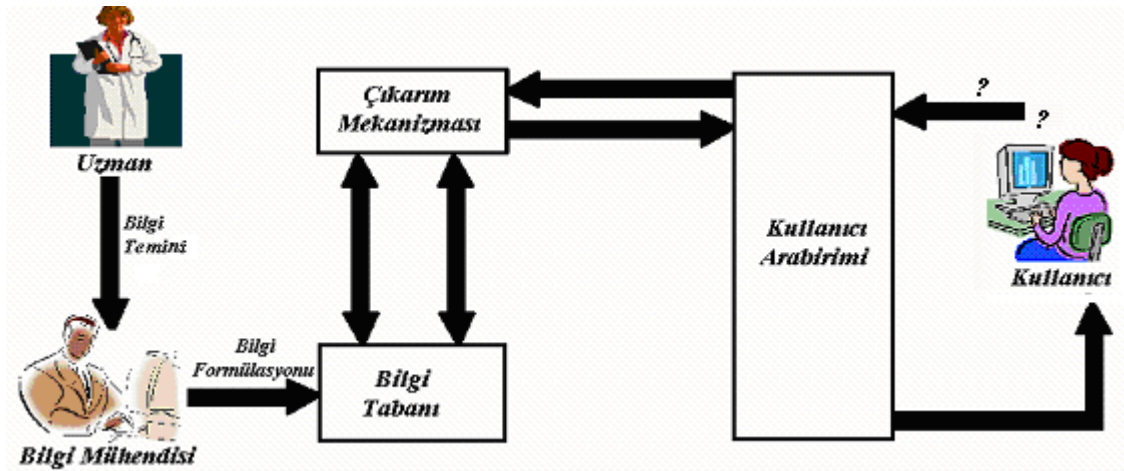
başlıkları altında incelenebilir [7, 44].

Uzman sistemlerin kullanımı ve sağladıkları yararlar bu sınırlamalara rağmen giderek artmaktadır. Özellikle yapay sinir ağları, bulanık mantık, genetik algoritmalar vb gibi yöntemleri kullanmasıyla uzman sistemler daha önceleri geçerli olan bir dizi kısıtlamaları ortadan kaldırmaktadır. Örneğin, yapay sinir ağı kullanan bir uzman sistem esnek öğrenme olanağına, bulanık mantık kullanan bir uzman sistem ise bilgi yetersizliği durumunda doğru sonuçlara varabilme olanağına sahiptir.

Her tasarlanan uzman sistem yardımıyla daha çok bilgi bilgisayar ortamına girilmiş olacaktır. Dolayısıyla bu bilgiler faydalı olacak şekilde analiz edilebileceklerdir.

2.2. Uzman Sistemlerin Yapısı

Uzman sistemler, uzman destek sistemi vererek kararsız problemleri ele alabilir ve çözüm getirebilir. Yapıları gereği, veri tabanı arabirimi ile karar destek sistemleri aynı zamanda kullanabilirler. Şekil 2.1’de görüldüğü gibi mimari bakış açısından birbirinden bağımsız ama etkileşimli dört elemanı vardır. 1) Bilginin temin edilmesi, 2) Bilgi tabanı (knowledge base), 3) Çıkarım mekanizması (inference engine), 4) Kullanıcı arabirimi (user interface) [43].



Şekil 2.1. Uzman sistem elemanları ve bilgi akışı

- a) Bilginin temin edilmesi: Uzman sistemin uzmanlık alanı ile ilgili bilgilerin toplanması, derlenmesi ve bilgisayarın anlayacağı şekle dönüştürülmesi çalışmalarını kapsar.

- b) Bilgi tabanı: Uzman sistemin uzmanlık alanı ile ilgili bilgilerin toplandığı yerdir. Bilgiler genellikle kurallar (Eğer...ise O ZAMAN... şeklinde), bilgi çatıları (çerçeveleri), bilgi sınıfları ve prosedürlerden oluşur. Bu bilgiler ilgili uzmanlık alanı hakkında uzmanların bildiği ve belirlediği gerçeklere dayanmaktadır.
- c) Çıkarım Mekanizması: Bilgi tabanında bulunan bilgileri arayan filtreleyen, yorumlayan ve sonuç çıkaran yani; çözüm üreten bir mekanizmadır. Genel olarak iki türlü çıkarım vardır.
1. İleri doğru zincirleme: Bu durumda ilgili problem hakkındaki gerçeklerden hareket edilerek sonuca gidilir.
 2. Geri doğru zincirleme: Bu durumda ise bir sonuç ele alınarak geriye doğru, o sonucu destekleyen gerçekler var mıdır? sorusunun cevabı aranır.
- d) Kullanıcı arabirimi: Uzman sistemi kullanan kişiler ile uzman sistemin iletişimini sağlar. Problemlere üretilen sonuçların nasıl üretildiği ve niçin o sonuçlara varıldığını açıklar. Uzman sistemin bir uzman gibi görülmesi bu arabirim ve açıklama yeteneğinin güçlü olmasına bağlıdır.

Kullanıcı, uzman olmayan herhangi biridir ve sistemin içyapısı hakkında bilgisi olması gerekmez. O yalnızca sonucunu almak istediği soru veya önermeleri sisteme verir ve bunlara karşılık cevap veya cevaplar alır. Bir uzman sistemin başarılılığının ölçüsü; doğru mantıklar sonuçlar verebilmesinin yanında, bu sonuçlara ulaşabilmesi için kullanıcının sistemin içyapısı hakkında ne kadar bilgili olmasının zorunlu olduğudur. Diğer bir deyişle, kullanımda olan bir uzman sistemde kullanıcı ile sistem arasındaki fark ne kadar büyükse, bu uzman sistem bir o kadar da başarılıdır.

Şekil 2.1'den görüldüğü gibi uzman sistemin elemanlarından birisi de uzmanın kendisidir. Bir uzman sistemi geliştirmek için en az bir uzmana ihtiyaç vardır. Bilgi mühendisi bilgi temini yöntemlerini kullanarak uzmandan bilgileri elde eder. Topladığı bilgileri derleyerek gereksiz bilgileri uzmanın yardımıyla ayıklar ve bilgileri bilgisayarın anlayacağı biçime getirerek bilgi tabanına koyar. Kullanıcı uzman sisteme bir soru sorduğunda, çıkarım mekanizması bilgi tabanını arayarak sorulan sorunun cevabını arar. İlgili bilgileri belirleyip probleme çözüm ürettikten sonra kullanıcı arabirimi aracılığı ile kullanıcıya sorusunun cevabı verilir. Bazı durumlarda problemleri çözecek bilgiler bilgi tabanında bulunmayabilir. Bu durumda bilgi tabanına yeni bilgi eklenmesi ve güncellenmesi olasıdır. Zaman içinde bilgi tabanında yeterli bilgi toplamak söz konusu

olabilir. Bilginin çoğalması ise çıkarım mekanizmasının problemlere çözümler üretme gücünü artırmaktadır [43].

Uzman sistemlerin gelişimi için, bilgiyi saklayabilen ve belirli formatlarda açıklayabilen insanlara ihtiyaç vardır. Çünkü bir çok uzman program yazamaz. Bilgi genellikle saklı ve kodlanmamış şekilde uzman insandadır. İşte bu bilgilerin kodlanabilir hale gelmesi için bilgi mühendisliği metodolojisine ihtiyaç duyulur.

Klasik programlamada, yazılım ile bilgi tabanı yada veriler aynı ortamda olduğundan, program içersinde değişiklik gerektiğinde programın yeniden yazılması gerekmektedir. Buna karşılık, uzman sistemlerde yukarda belirtilen modüller birbirlerinden bağımsız olduklarından bilgi tabanında istenilen değişiklik yapılarak aynı program başka amaçlar için de kullanılabilirler.

3. UZMAN SİSTEM TASARIM YÖNTEMLERİ

Bir uzman sistem tasarımına başlamadan önce, çözümü aranan problem için uzman sistemlerin en uygun yöntem olup olmadığına karar vermek gerekir. Örneğin geleneksel programlama ile çözüm bulunabilen bir problem için uzman sistemlere başvurmak gereksizdir. Bu konuda bir karara varabilmek için, göz önünde bulundurulması gereken bazı kriterler mevcuttur.

1. Problem, yordamsal programlama ile çözülebilir mi?

Eğer bu sorunun cevabı “Evet” ise, bu durumda uzman sistemlerle uğraşmanın en iyi çözüm olmadığı açıktır. Örneğin arıza yapan bir cihazdaki hatayı tespit etmek için, bir arızaya yol açan bütün nedenler bilindikten sonra yapılması gereken tek şey, sadece bir hata ve çözüm tablosuna bakmak yeterli olur. Uzman sistemler daha çok algoritmik çözümü olmayan, eldeki verileri kullanarak sonuç çıkarmak gereken problemler için uygun olur.

2. Uygulama yapılacak alanın sınırları kesin olarak belli mi?

Tasarlanacak uzman sistemin bilmesi gerekenler ve yeteneklerinin sınırları tam olarak belirli olmalıdır. Örneğin baş ağrılarına teşhis koymak için kullanılan bir uzman sistem düşünelim. Böyle bir uzman sistem için bir tıp doktorunun bilgisine başvurmak gerekir. Fakat baş ağrılarının nedenlerini çok ayrıntılı bir biçimde araştırmak, tavsiyede bulunmak için nörologun da bilgisine ihtiyaç duyulur. Daha ayrıntıya girildikçe biyokimya, kimya, moleküler biyofizik vs. gibi pek çok alan ile ilgili bilginin de bilgi tabanına eklenmesi gerekir. İşte böyle durumlara düşmemek için, uzman sistemin alanının mutlaka bir şekilde sınırlandırılması gerekir. Aksi takdirde uzman sistem çok karmaşık hale gelir.

3. Uzman sisteme ihtiyaç var mı?

Pek çok uzmanın bulunduğu bir alanda uzman sistem hazırlamak pek de mantıklı olmaz. Çünkü böyle bir uzman sistemin dayanak noktası az bulunan uzman bilgisinden ziyade, çok sayıda uzmandan rahatlıkla alınabilecek bilgiye dayanır.

4. İşbirliği yapılabilecek en az bir uzman kişi mevcut mu?

Tasarlanan projeye ilgi duyan en az bir uzman kişinin bulunması gereklidir. Çünkü herkes, bilgisinin doğruluğunun test edilmesini kabul etmeyebilir. Bir projeye çok fazla sayıda uzman kişinin katılması da bazen sakıncalı olabilir. Bir problemin çözümü için iki uzman farklı testler ve çözümler önerebilir. Bazen, farklı sonuçlara da varabilirler.

5. Uzman kiři, bilgisini anlařılabilir derecede aıklayabiliyor mu?

Bir uzman sistem hazırlanırken, bilgisine başvuru olan uzman kiřinin kendisine net bir biimde ifade edebilmesi ok onemlidir. Tasarım yapan mhendis, uzman kiřinin sylediklerini net bir řekilde anlayamazsa, aldıęı bilgiyi bilgisayar koduna dnřtrmesi ok zor olur [7]. Uzman sistemin tasarımına karar verildikten sonra izlenmesi gereken adımlar řu řekilde sıralanmalıdır [42]:

1. Problemin zmnn gerek amacının belirlenmesi,
2. Mmkn olduęu kadar problemin znn kavranması,
3. Problemleri ayırıştırarak, alt amaların ortaya ıkarılmasının saęlanması,
4. Disiplin alanının zel niteliklerinin belirlenmesi,
5. Seilmiş disiplin alanında uzmanlařmış bir uzmanın bulunması ve bilgiye dayalı sistemin tasarlanmasına yardım etmesinin saęlanması,
6. Uzmanla birlikte birkaç uygulamalı problemin zmne iřtirak edilmesi, uzmanın uyguladıęı yntemlerin belirlenmesi ve bu yntemlerin ayrıntılı olarak tasvir edilmesi,
7. Sistemin oluřturulması iin gereken program aralarının seilmesi,
8. Uzmanla birlikte zlen problemlerin rneklerini ieren bilgi sisteminin laboratuvar prototipinin kurulması,
9. Bilgi tabanının oluřturulması, disiplin alanı nesnelerinin arasındaki iliřkiler ve trlerinin belirlenmesi, nesnelerin sınıflara blnmesi, bilgi tabanı uzmanının disiplin alanının kuruluřu hakkındaki tasvirlerine uygun olarak yapılařtırılması,
10. Bilgi tabanının bytlmesi zerine gereken dng sayısının yapılmasının saęlanması. Bu esnada eklenen bilgilerin eliřkili olmasının yoklanmasına ve tespit edilen uygunsuzluęun giderilmesi iin modifikasyonunun saęlanması,
11. Gerekli program ve donanım aralarının belirlenmesi ve elde edilmesi,
12. Sistem belgelerinin iřlenilip hazırlanması,
13. İlk adımdan bařlayarak sınırlı olsa da dzgn alıřan bir sistemin kurulmasına alıřılmalı, bunun iin tasarımda modler bir yapı kullanılmalıdır.

3.1. Uzman Sistemlerin Tasarımında Sezgisel Yöntemin Kullanılması

Uzman sistemler bir dizi pratik problemleri çözmek için kullanılır. Uzman sistemlerde çözüme ulaşmak için, geleneksel programlama yöntemlerinin dışına çıkılır. Kısaca söylemek gerekirse uzman sistemler, çözülmesi zor olan ve bilgi talep eden problemlerin çözülmesinde kullanılırlar. Uzman sistemler, analiz yapabilen, teşhis koyabilen, tavsiye ve danışma verebilen bilgisayar programları sınıfına girerler. Bunlar adeta insandan bir uzmanın incelemesini talep eden problemlerin çözümüne yönelmiştir. Sıradan bir bilgisayar programından farklı olarak uzman sistemler, sınırlı bir disiplin alanında tündengelim akıl yürütme ile problemleri çözerler. Bu sistemler yapılaştırılmamış ve az belirlenmiş problemlerin çözümünü bulmaya imkan verirler. Yapılaştırılmamış yeterli bilgi ve zamanın olmaması gibi tam analiz yapılmasına imkan vermediği durumlarda sezgisel yöntemler kullanılır [42].

E. Feigenbaum'e göre, uzman sistemler, çok karmaşık olduğu için bilirkişi katılmadan çözülemeyen problemleri, bilgi tabanı ve çıkarım mekanizması kullanarak çözen akıllı bilgisayar programlarıdır.

Başka bir tanıma göre uzman sistemler, belli bir alandaki bilgiye ve deneyime dayanan verilerin bilgisayarda, o alanda istenen bir problem için akıllı danışma yapabilecek ve karar verebilecek şekilde bir program halinde gerçekleştirilmesidir [44].

Uzman sistemlerin arzu edilen bir niteliği, onun akıl yürütme sürecini anlaşılabilir bir şekilde açıklayabilmesidir. Bu nitelik formel kurallara dayanarak, programlama yoluyla elde edilir.

Uzman sistemlerde sezgisel yöntemler oldukça kullanılmaktadır. Örnek olarak [42]; Nehir kenarında (yatağında) bir miktar petrol artığı kirlenmesi tespit edilmiştir. Buna nehir kenarına sıralanmış fabrikalardan birinin veya birkaçının kaçak olarak akıttığı petrolün sebep olduğu sanılmaktadır. Amaç, hangi fabrikanın bu petrolü akıttığını bulmak ve nehirin kirlenmesini önlemektir. Böyle bir uzman sistem her şeyden önce hesaplama yapabilmelidir. Bunun için, petrolün suda çözülme süresince nehrin akma yönü ve hızı bilinmelidir. Bu bilgiler toplandıktan ve petrolün nehre karıştığı yer yaklaşık olarak belirlendikten sonra özel kurallar kullanılıp suçlu fabrika bulunabilir.

H1 kuralı;

Eğer hesaplanmış yerin 1 km çevresinde bir fabrika varsa **O Halde** nehre petrol karıştıran fabrika odur.

H2 kuralı:

Eğer hesaplanmış yerin 1 km çevresinde birkaç fabrika varsa **O Halde** nehre petrol karıştıran fabrika, petrolü en çok kullanan fabrikadır.

Bu çeşit kurallar sezgisel olarak adlandırılır. Bunların temel özellikleri alışılmış, kabul edilmiş bilgilere göre değil uzmanın pratik bilgilerine dayanarak biçimlendirilirler. Amaca ulaşırken önce sadeleştirmenin yapılması, bakılan varyantların sayısını azaltmaya imkan vererek düşünmenin verimini artırır. Uzman sistemlerde sezgisel kuralları kullanan sadeleştirme mekanizmasına sezgisel arama mekanizması denir. Sezgisel mekanizma, suyu kirleten suçlu adayını belirledikten sonra, ileri sürdüğü bu hipotezi yoklamak için başka kurallar kullanılabilir. Önce fabrikaların petrol taşıma sistemlerindeki en hassas yerlere bakılacaktır. Bunun için aşağıdaki sezgisel kurallar kullanılabilir;

Eğer 14 sahasındaki petrol kemerinin basıncı nehirdeki suyun basıncından az ise **O Halde** 14 nolu sahayı yoklamak gerekir. Böyle bir kural yalnız 14 nolu sahada yerleşmiş bu fabrikayı iyi tanıyan bir insan tarafından formlaştırılabilir. Yani, bu kural, o konuda bilirkişi olan bir insanın pratiğine dayandığından dolayı sezgiseldir.

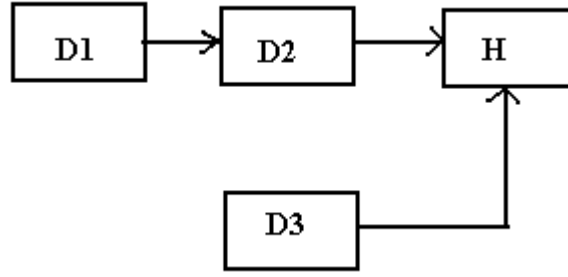
3.2. Kurallara Dayalı Uzman Sistemlerin Tasarlanması

Uzman sistem programlar genel olarak "Muhakeme Etme"; yani eldeki verilere göre en uygun durumu belirleme esasına göre çalışırlar. Genellikle bilgi tabanındaki tüm kuralların muhakeme edilmesi Geriye ve İleriye Zincirleme olmak üzere iki teknikle gerçekleştirilir [45].

3.2.1. Geriye Zincirleme

Muhakeme ünitesi; problemi çözerken kuralın en sonu olan sonuç (THEN. . .) cümlesi ile başlar ve şart (IF. . .) cümleleri tatbik edilerek çözüm bulunur. Yani bu tür zincirleme tümdengelim ilkesini temel olarak alır ve sonuç kısmını sağlayacak bütün kuralları tek tek inceler.

Geriye zincirleme yönteminde, önceden bir hipotezin (H) gerçek olduğu varsayılır. Bu hipotezi kanıtlayacak Şekil 3.1’de görülen deliller (D1, D2 ve D3) aranır. Burada H hipotezini kanıtlamak için D2 veya D3 delillerini sorgulamak gerekir. Fakat D2 de D1’in sorgulanmasını istemektedir. D3’ü sorgulamakla ise D1 ve D2’ye ulaşmadan H hipotezi kanıtlanabilir.

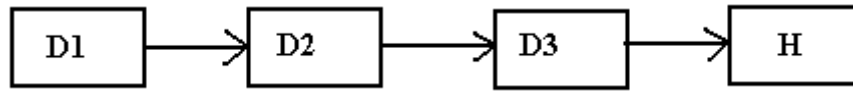


Şekil 3.1. Geriye zincirleme

Geriye doğru zincirlemenin, Genişlik öncelikli ve Derinlik öncelikli olmak üzere iki şekli vardır. Genişlik öncelikli geriye doğru zincirleme, o anda eldeki amaca çözüm bulmak için tüm kuralların sonuç kısmını kontrol eder. Çözüm bulamazsa kuralların şart kısımlarına bakar. Derinlik öncelikli geriye doğru zincirleme ise, eldeki amaca çözüm bulmak için ilgili bir kural bulur ve bu kuralın önce şart kısmına bakar. Bu kuralın şart kısmı sonuca götürmezse başka bir kural arar [45].

3.2.2. İleriye Zincirleme

Muhakeme ünitesi, problemin en başından başlayarak (IF cümlesinden) sonuç kısmına (THEN. . .) ulaşmasıdır. Bu yöntem tümevarım mantığı ile çalışır. İleriye zincirleme yönteminde eğer D1 delili kanıtlanırsa D2'ye geçilir. Bu da kanıtlanırsa Şekil 3.2'de görüldüğü gibi D3 'e geçilir. Bu delilin de kanıtlandığı durumda H hipotezinin doğru olduğu kanaatine varılır.



Şekil 3.2. İleriye zincirleme

Geriye doğru zincirleme tekniği kadar basit olmasa da ileriye doğru zincirleme yönteminin Prolog dilinde ifade edilmesi de kolaydır. Herhangi bir disiplin alanında optimal bir uzman sistem tasarlanması için bu sistem, uygun disiplin alanına yönelmelidir. İleriye doğru zincirleme yöntemi, bir kurala ait bir koşul cümlesiyle başlayan ve eylem kurallarını harekete geçiren ve ileriye doğru kurallar zinciriyle çalışan işlerde kullanılır. Bu tür zincirleme sırasında çıkarım mekanizması, "EĞER" koşullarının bilgi tabanındaki diğer

kurallarla olan uyumunu araştırır. Bu işlemlerle yeni değerler oluşturulduktan sonra, eylem kuralları uygulanır.

İleriye doğru zincirleme tekniği, başlangıç bilgilerinden ileriye doğru çıkarım yaparak ilerler. Kullanıcılar, bu teknikte istedikleri düzende istedikleri kadar veri sağlarlar. Ancak kurallar, yeterli veri olduğu zaman çalışmaya başlar. Yeni gelen verilerle kuralların bütün koşulları yada diğer kuralların sonuçları eşleştiği zaman, sistem işlemeye başlar. Bu durumda bir problemin sonucunu veren bir kuralın bulunması ilkesi kullanılır. Örnek olarak, hareket halinde olan bir arabanın motoru aşırı ısınır. Bunun sonucunu aşağıdaki kurallarla belirlemek mümkündür [42].

Kural 1: Eğer motor çok ısınmışsa **O Halde** motor durur.

Kural 2: Eğer motor durursa **O Halde** para harcanmasına ve eve geç dönülmesine sebep olur.

Motorun ısınmasının fazla para harcanmasına ve eve geç dönülmesine sebep olduğu sonucuna nasıl varıldı? Bu sonucu elde etmek için, ileriye doğru zincirleme akıl yürütülmesi kullanılır. Başlangıç noktasına motorun çok ısınması durumu getirilir. Sonra 1. kuralın şart kısmı Eğer çalışır. Durum bu şarta uygun geldiği için (motor ısınmıştır), kuralın O Halde kısmı eşlenerek motorun çalışmayacağı sonucuna varılır. Akıl yürütme zinciri devam eder. 2. kuralın Eğer kısmındaki iddianın doğruluğu 1. kuraldan belirlenmişti (motor durur). Böylece para harcanması ve eve geç dönülmesi gibi sonuçlar bu iki durum sonunda elde edilir.

Tasvir edilen bu ardışıklık, kuralın EĞER kısmı doğru ise o zaman kuralın O HALDE kısmı yerine getirilir ki bu ileriye doğru zincirleme olarak adlandırılır. Böylece başlangıç noktasında verilen kurallar yürütülerek sonuca ulaşılır. İleriye doğru akıl yürütmeyi gerçekleştiren program, baktığımız örnekteki verilmiş durum için kullanıcının sorularına, ileriye doğru akıl yürütme prensiplerini ve bilgi tabanını kullanarak cevap vermelidir. Başka bir deyişle program, kullanıcıdan oluşmuş durum hakkındaki verileri (örneğin motorun ısınması), bilgi tabanının adını (örneğin, "arabanın teknik problemleri") almalı ve bilgi tabanındaki enformasyonu gözden geçirmelidir. Sonra da, ileriye doğru akıl yürütmeyi kullanarak motorun çalışıp çalışmayacağı veya tamir için para harcanması gerektiği ve eve geç dönüleceği sonuçlarını bulmalıdır.

Arabanın çalışmadığı varsayılırsa; bu nedenden hareket ederek arabanın çalışmaması koşullarını belirlemek gerekir. İleriye doğru akıl yürütmede bu koşullar bellidir (motor hararet yapmıştı). Fakat bu koşulların getireceği sonuçlar belli değildir. Problem, mümkün

olabilecek sonucu tahmin etmektir. Geriye doğru akıl yürütmede ise sonuç bellidir ve bu sonucu oluşturan nedenlerin bulunması gerekir. Problemin çözümü için kurallar gereklidir.

Kural 1. Eğer araba çalışmıyorsa **Ve** akü bitmişse **O Halde** marş motoruna elektrik gelmemektedir.

Kural 2. Eğer marş motoruna elektrik gelmiyorsa **O Halde** araba hareket edemez.

Arabanın hareket edememe koşullarını bulmak için geriye doğru akıl yürütme kullanılarak 2. kuralın O Halde kısmındaki gerçek sonucuna, bu kuralın Eğer kısmında bulunan "Marş motoruna elektrik gelmediği" koşulu ile ulaşılır. Akıl yürütme işine devam edilir. Çünkü marş motoruna neden elektrik gelmediği belirlenmelidir. Bu sorunun cevabı 1. kuraldadır. 1. kuralın O Halde kısmındaki iddia doğru olduğu için (marş motoruna elektrik gelmemektedir), onun Eğer kısmı da doğrudur. Yani araba çalışmıyor ve akü bitmiştir.

Bazen kuralda doğru olan iddiaya uygun olarak koşullu kısım bulunmuyorsa, o zaman uzmanlara müracaat ederek ek bilgiler almak gereklidir. Yani, uzman sistemdeki bütün kuralların koşul kısmındaki iddialar yanlış ise, o zaman işleme, problemi çözmeye yardım eden mantıklı çıkarımlar eklenmelidir.

Görüldüğü gibi zincirleme, burada bir dizi kuralın mantıklı ilişkilerinin prosedürüdür. Geriye doğru zincirleme, ele alınan olaydan geriye, onu oluşturan kaynaklara doğru akıl yürütme ile devam etmektir.

3.3. Olasılık Teorisi, Bulanık Mantık Ve Yapay Sinir Ağları Yöntemi İle Uzman Sistemlerin Tasarlanması

Uzman sistem tasarımında olasılık teorisi, bulanık mantık, yapay sinir ağları gibi yapay zeka teknikleri de kullanılmaktadır.

3.3.1. Olasılık Teorisine Dayanan Uzman Sistem Tasarlanması

Sezgisel kurallar, uzmanlar tarafından belirlenen kurallardır. Bir çok sezgisel kuralın temeli belli olayların ortaya çıkmasına bağlıdır. Bu olayların ortaya çıkma olasılığını yalnız uzman görebilir veya hesaplayabilir. Başka bir deyişle, bir olayın meydana gelmesi ihtimali hakkında uzmanın belirli bir fikri olmaktadır. Aslında tecrübeye de dayalı olsa bir olasılık hesabı söz konusu olunca mutlaka istatistiksel verilere ihtiyaç duyulur. Örneğin, bir doktor belli bulguları değerlendirerek bir teşhise karar verirken daha önce gördüğü bu

tip hastaları istatistiksel veri olarak değerlendirmekte ve koyduğu teşhisin güvenilirliğini daha önce doğru olarak koyduğu teşhislerine dayandırmaktadır. Aynı bulgularla daha önce karşısına çıkan hastalar hekimin deneyimini artırmakta ve doğru teşhis koyma ihtimalini kuvvetlendirmektedir. Bir başka örnek vermek gerekirse; petrol arama şirketleri yaptıkları gözlem deney ve incelemeler neticesinde petrol yatağının bulunduğu ve orada bir petrol kuyusu kazılması gerektiğine karar verebilir. Fakat petrolün olması bir ihtimal olacaktır ve kuyuda petrol çıkmayabilir.

İstatistiksel çözümlere dayanan ve koşullu olasılık teorisi içinde yer alan yöntem Bayes tarafından geliştirilmiştir. Bu yöntem "önce bir olay ortaya çıktığı için herhangi bir olay oluşacaktır" iddiasına dayanır. Bazı olayları belirlenmiş kurallar yardımıyla her zaman tanımlamak mümkün olmayabilir. Örneğin bir insanın ateşi 37 dereceden fazla ama 38 dereceden az ise hafif bir soğuk algınlığı geçirdiği, daha yüksek ateşler için ise ciddi bir hastalığının olduğu söylenebilir. Fakat insanlar yüksek, az, çok, kolay vb. gibi kelimeleri her zaman aynı miktarı anlatacak şekilde kullanmadıklarından sözlerin manasını, bir "kesinlik" anlatacak şekilde ifade etmek de mümkün olmayabilir. Çoğu insan olasılık hesabını yanlış anlamıştır. Örneğin, 100 kişiden 95'i para atıldığı zaman neden %50 ihtimalle yazı geleceği sorusuna "Para ya yazı yada tura gelmek zorundadır. Bu nedenle de yazı veya tura gelmesi ihtimali %50'dir" şeklinde cevap verir. Hilesiz bir paranın yazı ve tura gelmekten başka şansı yoktur. Ama ağırlık merkezi ayarlanmış bir para %90'lar seviyesinde tura gelebilir. Gerçek ihtimal hesabı bir olasılık kümesine dayanmalıdır. Yeterli miktarda yapılan temiz denemeler sonucunda yazı gelme oranı, tura gelme oranına eşit ve bu da %50'dir. Deneme yapılmadan bir olasılıktan bahsedilmesi, insanı tamamen yanlış neticelere götürür. Zaten tecrübeler de bir çeşit denemedir. İşte bu nedenle deneyimi fazla olan uzmanlarla çalışmak daha doğru neticeler elde etmek için gereklidir.

Belirsizliğin işlenmesi YZ'nın temel konularından biridir. Doğru ve iyi karar vermede kullanılan belirsizliklerin olasılık yöntemleri ile (Bayes, Dempster-Shafer vb. yöntemler) işlenmesi [44] nolu çalışmada anlatılmıştır. Diğer taraftan yaklaşık karar verme yöntemleri (bulanık mantık, yapay sinir ağları vb.) son yıllarda yaygın kullanılmaya ve geliştirilmeye başlanmıştır.

3.3.1.1. Bayes Olasılığı

Bayesian Olasılık Kuramı, matematiksel istatistik kuramının bir dalıdır. Bu kuram; belirsizlik taşıyan herhangi bir durumun modelinin oluşturularak, bu durumla ilgili evrensel doğrular ve gerçekçi gözlemler ışığında belli sonuçlar üretilmesini sağlar.

Olasılık kuramında en önemli kavramlardan birisi de 'Koşullu Olasılık' dır. $P(X = x | Y = y) = r$ koşullu olasılığının sözlü ifadesi : "Eğer $Y = y$ doğru ise, $X = x$ olma olasılığı r 'dir" şeklinde söylenir [44].

X ve Y 'nin alabileceği değerlerin her kombinasyonu için koşullu olasılıkları belirleyen tabloya koşullu olasılık dağılımı adı verilir ve $P(X|Y)$ ile ifade edilir. Koşullu olasılık, 'Çarpım Kuralı' nı belirlemede önemli bir yer tutar. Çarpım kuralı, iki olayın birden oluşma olasılığını tanımlar ve $P(A \cap B)$ ile ifade edilir.

Bu durumda,

$$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A) \text{ 'dır.}$$

Literatürde $P(A \cap B)$ kısaca $P(A, B)$ ile gösterilir.

$P(A, B)$, 'Kesişme Olasılık Dağılımı' olarak adlandırılır. Tıpkı koşullu olasılık dağılımı gibi, $P(X, Y)$ 'de X ve Y 'nin alabileceği değerlerin her kombinasyonu için kesişme olasılıklarını belirler. En genel anlamda 'n' adet değişkenin kesişme olasılık dağılımı, çarpım kuralını uygulayarak özyineli biçimde aşağıdaki gibi gösterilir:

$$\begin{aligned} P(X_1, X_2, \dots, X_n) &= P(X_1 | X_2, \dots, X_n) P(X_2, \dots, X_n) \\ &= P(X_1 | X_2, \dots, X_n) P(X_2 | X_3, \dots, X_n) P(X_3, \dots, X_n) \\ &= P(X_1 | X_2, \dots, X_n) P(X_2 | X_3, \dots, X_n) \dots P(X_{n-1} | X_n) P(X_n) \end{aligned}$$

Bu eşitliğin yeniden düzenlenmesi ile ünlü 'Bayesian Kuramı' elde edilir:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Bayesian Kuramı belirsizlik taşıyan durumlarda karar vermede çok kullanışlıdır.

Eşitlikte yer alan terimlerin anlamları:

A : Belirsizlik taşıyan önerme,

B : Delil (Evidence),

$P(A|B)$: A 'nın B delilinden sonraki olasılığı,

$P(A)$: A 'nın B delilinden önceki olasılığı,

$P(B|A)$: B delilinin A olayının gerçekleşmesi için oluşma ihtimalinin en büyük olasılık değeri.

$1 / P(B)$: Bir normalizasyon etmenidir ve hesaba katılmayabilir.

Bu durumda Bayesian Kuramı şu duruma dönüşmüş olur:

$$P(A|B) = P(B|A) P(A)$$

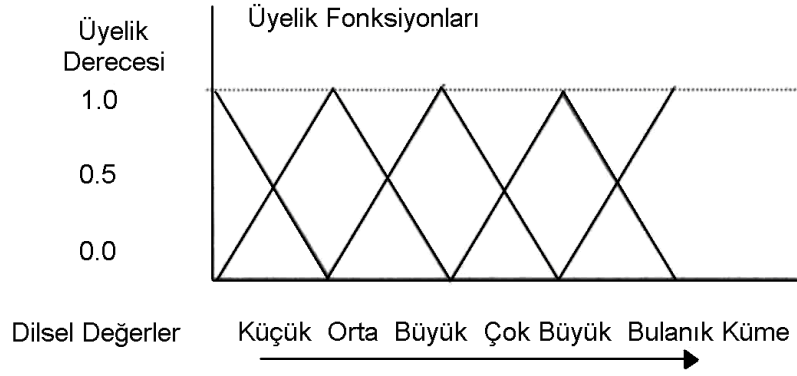
Kuramın bu biçimi, hesaplamaların daha etkin ve karmaşıklıklardan uzak yapılabilmesini sağlar.

3.3.2. Bulanık Mantık Ve Bilgi Sistemlerinde Kullanılması

Birbirinden keskin farklılık gösteren iki durumla tasvir edilen Boole mantığı, bir takım durumlarda kullanılamamaktadır. Tam ve kesin olmayan bilgilere dayanarak tutarlı ve doğru kararlar vermeyi sağlayan düşünme ve karar verme mekanizması bulanık mantık olarak adlandırılır. 1965 yılında California Üniversitesinden (Berkeley) Azeri kökenli Ali Asker Lütvi-Zade (Lotfi Zadeh) tarafından ortaya atılan bu mantık şimdi birçok alanlarda kullanılmaya başlamıştır [46].

Boole mantığı yaklaşımında herhangi bir önerme ya doğrudur yada yanlıştır, arada üçüncü bir durum söz konusu değildir. Eğer bu durum küme kavramına uygulanacak olursa, elemanlar A kümesine ait olanlar (1) ve ait olmayanlar (0) şeklinde iki birbirinden ayrık gruba ayrılırlar. Bulanık mantık yaklaşımında bir kümeye üye olanlar ve olmayanlar şeklinde bir kesin sınıflandırma mevcut değildir.

Bulanık mantıkta örneğin "boyut" gibi bir dilsel değişken "küçük", "orta" veya "büyük" gibi birkaç dilsel değer alabilmektedir. Her bir dilsel değere üyelik fonksiyonuna bağlı bir bulanık küme gibi bakılır. Bu üyelik fonksiyonu üçgen şeklinde, çan biçiminde veya diğer bir biçimde olabilir. "Boyut" dilsel değişkeni için üyelik fonksiyonları şekil 3.3 'de gösterilmektedir. Sayısal değer verildiğinde her bir dilsel kategori için ona karşılık gelen üyelik fonksiyonundan o değer için üyelik derecesi bulunabilir. Tersine, eğer her bir bulanık kümedeki üyelik fonksiyonları verilmiş ise onun sayısal değerleri de hesaplanabilir (durulaştırma).



Şekil 3.3. "Boyut" dilsel değişkeni için üyelik fonksiyonları

3.3.2.1. Bulanık Mantık Kavramları

Bulanık kümelerde, söz konusu evrenin elemanlarının bir A bulanık kümesine ait olma derecelerini temsil etmek amacıyla üyelik fonksiyonları belirlenir. Bulanık kümeler, üyelik fonksiyonları ile tanımlanırlar. Bu fonksiyonlar, elemanlara $[0,1]$ kapalı aralığında gerçel değerler atayarak elemanların A bulanık kümesi ile temsil edilen kavrama ne derece uygun olduklarını veya A bulanık kümesi ile temsil edilen özellikleri ne derecede taşıdıklarını gösterirler. Bir A bulanık kümesinin üyelik fonksiyonu şu şekilde gösterilir:

$$m_A(x) \rightarrow [0,1], x \in X$$

Burada x , X evrensel kümesinden alınan elemanları, $m_A(x)$ ise x elemanının A bulanık kümesine üyelik derecesini göstermektedir. $m_A(x)$ 'in değeri aşağıdaki gibi hesaplanır:

$$m_A(x) = \begin{cases} 1, & \text{eger } x \in A \\ 0, & \text{eger } x \notin A \end{cases}$$

$X = \{X_1, X_2, \dots, x_n\}$ sonlu kümesi için A bulanık kümesi şöyle hesaplanır:

$$A = m_A(x_1) / x_1 + m_A(x_2) / x_2 + \dots + m_A(x_n) / x_n = \sum_{i=1}^n m_A(x_i) / x_i$$

X sonsuz olduğu durumda:

$$A = \int_x m_A(x) / x$$

Bulanık küme teorisinde de, klasik küme teorisinde olduğu gibi kümeler üzerinde tanımlanmış birleşim, kesişim ve tümleme işlemleri vardır. Bu işlemlerin

gerçekleştirilmesi için bir takım fonksiyonlar tanımlanır. Bir fonksiyonun bir bulanık kesişim, bir bulanık birleşim veya bir bulanık tümleme fonksiyonu olabilmesi için bir takım aksiyonları sağlaması gerekmektedir.

Bulanık küme teorisinin standart işlemleri ($\forall x \in X$ için) [46]:

Bulanık Tümleme:

$$m_{\bar{A}}(x) = 1 - m_A(x)$$

Bulanık Birleşim veya Max:

$$m_{A \cup B}(x) = \max(m_A(x), m_B(x))$$

Bulanık Kesişim veya Min:

$$m_{A \cap B}(x) = \min(m_A(x), m_B(x))$$

Cebri Çarpım:

$$m_{A \bullet B}(x) = m_A(x) \bullet m_B(x)$$

Her üç işlem için aksiyomatik iskeleti oluşturan aksiyomlar mevcuttur.

Sınırlı Çarpım:

$$m_{A \bullet B}(x) = \min(m_A(x), m_B(x))$$

Kesin (Drastic) Çarpım:

$$m_{A \cap B}(x) = \begin{cases} \min(m_A(x), m_B(x)), & \text{eğer } m_A(x) = 1 \text{ veya } m_B(x) = 1 \\ 0, & \text{diğer durumlarda} \end{cases}$$

Cebri Toplam:

$$m_{A+B}(x) = m_A(x) + m_B(x) - m_A(x) \bullet m_B(x)$$

Sınırlı Toplam:

$$m_{A \cup B}(x) = \min(1, m_A(x) + m_B(x))$$

Kesin (Drastic) Toplam:

$$m_{A \in B}(x) = \begin{cases} \max(m_A(x), m_B(x)), & \text{eğer } m_A(x) = 0 \text{ veya } m_B(x) = 0 \\ 1, & \text{diğer durumlarda} \end{cases}$$

Sınırlı Fark:

$$m_{A|-B}(x) = \max(0, m_A(x) - m_B(x))$$

Simetrik Fark:

$$m_{A \vee \bar{B}}(x) = |m_A(x) - m_B(x)|$$

Bulanık kümeler üzerinde ikili işlemlerden başka n'li işlemler de tanımlanabilir. Bulanık kesişim ve bulanık birleşim fonksiyonlarında asosyatiflik özelliği kullanılarak argüman sayısı n'e çıkartılıp, n'li işlemler haline dönüştürülebilirler.

Bulanık Kümenin Gücü:

$$|A| = \sum m_A(x)$$

α -Kesim:

$$A^\alpha = m_A(x) \geq \alpha$$

A bulanık kümesinin n dereceden üssü:

$$m_A^n(x) \geq [m_A(x)]^n, \quad \forall n \in R^+, \text{ burada } R^+ \text{ pozitif gerçel sayılar kümesidir.}$$

A'nın Sıklaştırılması (Consentration):

$A = ((x, m_A(x)))$ bulanık küme olsun. Bu durumda

$$Con_n A = (x; [m_A(x)]^n)$$

A'nın Genişletilmesi (Dilation):

$$Dil_n A = (x; \sqrt[n]{m_A(x)})$$

Sıklaştırma ve genişletme işlemlerinin bulanık kümelerin çeşitli dönüştürmelerde kullanılması örnekleri aşağıda gösterilmektedir:

$$A = \int m_A(x) / x$$

$$\text{Çok } A = \int [m_A(x)]^2 / x$$

$$\text{Çok çok } A = \int [m_A(x)]^4 / x$$

$$\text{Az veya çok } A = \int \sqrt{m_A(x)} / x$$

$$\text{AZ } A = \int \sqrt[4]{m_A(x)} / x$$

$$\text{Değil } A = \int 1 - m_A(x) / x$$

$$\text{Çok Değil } A = \int 1 - [m_A(x)]^2 / x$$

Elemanlarla üyelik dereceleri arasındaki bölüm işareti ve toplama işareti gerçek kullanımındaki anlamında değildir. Sadece elemanlar arası ilişkileri ve elemanlarla üyelik derecesi arasındaki ilişkiyi ortaya koymak için kullanılırlar. Sayılamayan kümeler için

kullanılan integral işareti de yine kümenin süreksizliğini göstermek için kullanılır.

3.3.2.2. Bulanık Uzman Sistemler

Bulanık Uzman Sistem (BUS), veriler üzerinde akıl yürütme yapabilmek için Boole mantığı yerine, fonksiyon ve kuralların bulanık üyelik özelliklerini kullanan bir US'tir. Aşağıda BUS'lerdeki kurallara bir örnek verilmektedir:

Eğer X düşük ve Y yüksek ise **O Halde** Z ortadır.

Burada X ve Y giriş değişkenleri veya bilinen veri değerlerinin adlarıdır;

Z çıkış değişkeni veya değeri hesaplanması istenen verinin adıdır; düşük - X üzerinde belirlenmiş üyelik fonksiyonu (bulanık altküme); yüksek - Y üzerinde belirlenmiş üyelik fonksiyonu; orta - Z üzerinde belirlenmiş üyelik fonksiyonudur. Kuralın varsayım (antecedent, eğer) kısmı bu kuralın ne derecede uygulanabileceğini, çıkarım (consequent, o halde) kısmı ise bir veya birden çok çıkış, değişkenlerinin her birine üyelik fonksiyonu tahsis edilmesini tanımlamaktadır. Bir kural birden çok hüküm çıkarmaya da olanak tanıyabilmektedir.

Genel olarak BUS'lerde çıkarım süreci üç veya dört adımdan oluşmaktadır [42]:

1. **Bulanıklaştırma:** Her bir kural varsayımının doğruluk derecesini belirlemek için gerçek değerlere uygulanmış giriş değişkenleri üzerinde üyelik fonksiyonlarının belirlenmesi;

2. **Çıkarım:** Her bir kuralın varsayım kısmı için doğru değerlerin hesaplanması ve bu değerlerin her kuralın çıkarım kısmına uygulanması. Bir bulanık altkümede olan bu sonuçlar her kuraldaki her çıkış değişkenine atanırlar. Genelde yalnız Min veya Çarpım işlemleri bir çıkarım kural olarak kullanılır. Min çıkarımda çıkış üyelik fonksiyonu, kural varsayımının hesaplanmış doğruluk derecesine uygun ağırlıkların kesişmesi ile elde edilir (bulanık VE). Çarpım çıkarımda çıkış üyelik fonksiyonu, kural varsayımının hesaplanmış doğruluk derecesi ile ölçeklenir.

3. **Bileşim:** Her bir çıkış değişkenine atanmış bulanık alt kümelerinin tümünün, her bir çıkış değişkeni için bir tane bulanık altküme oluşturulması için birleştirilmesi. Bu amaçla en çok Max veya Toplam fonksiyonları kullanılır. Max bileşim, birleştirilmiş çıkış bulanık altkümeleri çıkarım kuralları değişkenlerine atanmış bulanık altkümelerin maksimumlarını alarak elde edilir (bulanık VEYA). Toplam bileşim, birleştirilmiş çıkış

bulanık altkümeleri çıkarım kuralları değişkenlerine atanmış bulanık altkümelerin toplamalarını almakla elde edilir.

4. **Durulaştırma:** Bulanık çıkış kümesinin kesin (crisp) sayılara dönüştürülmek istendiği zaman yapılan işlem ele alınır. 30'dan fazla durulaştırma yöntemi mevcuttur. Bunlardan en çok bilinenleri Merkezi (CENTROID) ve MAXIMUM yöntemleridir. CENTROID yönteminde çıkış değişkeninin kesin değeri, bulanık değer için üyelik fonksiyonunun ağırlık merkezinin değişken değerini bulmakla hesaplanır. MAXIMUM yönteminde bulanık kümede maksimum doğruluk değeri olan değişken değerlerinden biri, çıkış değişkeninin kesin değeri olarak seçilir.

Bulanık mantık kullanılarak kurallar oluşturmak ve yukarıda saydıklarımızı uygulamak için bir örnek verelim. X,Y ve Z değişkenlerinin [0, 10] arasında değer aldıklarını ve üyelik fonksiyonlarının aşağıdaki gibi hesaplanırsa:

$$\text{Düşük (t)} = 1 - (t/10)$$

$$\text{Yüksek (t)} = t/10.$$

Kurallar aşağıdaki gibi olsun:

Kural 1. Eğer X düşük Ve Y düşük ise O Halde Z yüksektir.

Kural 2. Eğer X düşük Ve Y yüksek ise O Halde Z düşüktür.

Kural 3. Eğer X yüksek Ve Y düşük ise O Halde Z düşüktür.

Kural 4. Eğer X yüksek Ve Y yüksek ise O Halde Z yüksektir.

Bu kurallardan görüldüğü gibi, giriş ve çıkış değişkenlerine (X: Y ve Z) somut bir değer değil, bulanık küme değerleri (düşük veya yüksek) atanmaktadır. Bu örnekte dikkat edilmesi gereken bazı hususlar var:

1. Tüm t'ler için düşük (t) + yüksek (t) = 1.0.

2. t'nin bir değerinde düşük (t) değeri maksimum ise bu değerde yüksek (t) değeri minimum (ve tersine) olmaktadır.

3. Aynı üyelik fonksiyonları bütün değişkenler için kullanılmaktadır.

1. ve 2. hususları geneldir, ama bu hususlar istenmeyebilir. 3. hususu genel değildir, ama istenmeyebilir.

Bulanıklaştırma sürecinde, giriş değişkenleri üzerinde belirlenmiş üyelik fonksiyonları, onların gerçek değerlerine uygulanır. Bu, her bir kural varsayımı için doğruluk derecesini belirlemeye olanak tanır.

Herhangi bir kural varsayımı için doğruluk derecesi α olarak işaretlenir. Eğer kural varsayımı sıfırdan farklı doğruluk derecesine sahip ise, kuralın o halde kısmi işletilir denilir.

$$\alpha_1 = \min(\text{düşük}(X), \text{düşük}(Y)); \alpha_2 = \min(\text{düşük}(X), \text{yüksek}(Y));$$

$$\alpha_3 = \min(\text{yüksek}(X), \text{düşük}(Y)); \alpha_4 = \min(\text{yüksek}(X), \text{yüksek}(Y)).$$

Tablo 3.1 'de değişkenlerin aldığı değerler gösterilmektedir. Burada X ve Y'ye 0 ile 10 arasında dört değer verilmektedir: 0,0; 3,2; 6,1 ve 10. Bu durumlarda, X ve Y'nin 0,0 değerlerinde düşük (X) ve düşük (Y) değerleri maksimum, yani 1 olacak, yüksek (X) ve yüksek (Y) ise minimum, yani 0 değerleri alacaktır. Çıkarım alt sürecinde her kuralın varsayımı için doğruluk değeri hesaplanır ve her kuralın çıkarım kısmına uygulanır ($\max \alpha_1, \alpha_2, \alpha_3, \alpha_4$). Bir bulanık altkümedeki bu sonuçlar her kuraldaki çıkış değişkenine atanır. Min (Bulanık Kesişim) ve Çarpım, iki çıkarım yöntemi veya çıkarım kuralıdır. Örneğin, X=0,0 ve Y=3,2 olduğunda Kural 1'e bakalım. Tablo 3.1 'den de görüldüğü gibi varsayılan doğruluk derecesi 0,68'dir (α_1). Bu kural için Min çıkarım Z'ye atanacak. Bu değer aşağıdaki gibi hesaplanır:

$$\text{Kural 1 } (Z) = \begin{cases} Z/10 & \text{eğer } Z \leq 6,8 \\ 0,68 & \text{eğer } Z > 6,8 \end{cases}$$

Aynı koşullarda çarpım çıkarım böyle belirlenir:

$$\text{Kural 1 } (Z) = 0,68 * \text{yüksek}(Z) = 0,068 * Z.$$

Tablo 3.1. Bulanık giriş değişkenlerinin aldığı değerler

X	Y	Düşük (X)	Yüksek (X)	Düşük (Y)	Yüksek (Y)	$\alpha 1$	$\alpha 2$	$\alpha 3$	$\alpha 4$
0.0	0.0	1.0	0.0	1.0	0.0	1.0	0.0	0.0	0.0
0.0	3.2	1.0	0.0	0.68	0.32	0.68	0.32	0.0	0.0
0.0	6.1	1.0	0.0	0.39	0.61	0.39	0.61	0.0	0.0
0.0	10	1.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0
3.2	0.0	0.68	0.32	1.0	0.0	0.68	0.0	0.32	0.0
6.1	0.0	0.39	0.61	1.0	0.0	0.39	0.0	0.61	0.0
10	0.0	0.0	1.0	1.0	0.0	0.0	0.0	1.0	0.0
3.2	3.1	0.68	0.32	0.69	0.31	0.68	0.31	0.32	0.31
3.2	3.3	0.68	0.32	0.67	0.33	0.67	0.33	0.32	0.32
10	10	0.0	1.0	0.0	1.0	0.0	0.0	0.0	1.0

Min (Bulanık Kesişim) çıkarımı Z'ye aşağıdaki dört bulanık altkümei atamaktadır:

$$\text{Kural 1 } (Z) = \begin{cases} Z/10 & \text{eğer } Z \leq 6.8 \\ 0.68 & \text{eğer } Z > 6.8 \end{cases}$$

$$\text{Kural 2 } (Z) = \begin{cases} 0.32 & \text{eğer } Z \leq 6.8 \\ 1 - Z/10 & \text{eğer } Z > 6.8 \end{cases}$$

$$\text{Kural 3 } (Z) = 0.0$$

$$\text{Kural 4 } (Z) = 0.0$$

Max bileşim aşağıdaki bulanık alt kümenin sonucu elde edilir:

$$\text{Bulanık}(Z) = \begin{cases} 0.32 & \text{eğer } Z \leq 3.2 \\ Z/10 & \text{eğer } 3.2 \leq Z \leq 6.8 \\ 0.68 & \text{eğer } Z \geq 6.8 \end{cases}$$

Çarpım çıkarım Zye aşağıdaki dört bulanık alt kümeleri atayacaktır.

$$\text{Kural 1 } (Z) = 0.068 * Z$$

$$\text{Kural 2 } (Z) = 0.32 - 0.032 * Z$$

$$\text{Kural 3 } (Z) = 0.0$$

$$\text{Kural 4 } (Z) = 0.0$$

Bulanık altkümede toplam bileşim sonucu aşağıdaki gibi olacaktır:

$$\text{Bulanık } (Z) = 0.32 + 0.036 * Z$$

Durulaştırma için CENTROID yöntemini kullanalım. CENTROID yönteminde

$f(x)$ momentini hesaplamak için $\int x f(x) dx$ hesaplanır. $f(x)$ alanını hesaplamak için $\int f(x) dx$ hesaplanır. Bu durumda entegral $[0;10]$ arasında $(0.32 + 0.36 * Z) dZ$ olarak hesaplanılır:

$$0.32 * 10 + 0.018 * 100 = 3.2 + 1.8 = 5.0.$$

$[0;10]$ arasındaki entegral momenti

$$(0.32 * Z + 0.036 * Z * Z) dZ = 0.16 * 10 * 10 + 0.012 * 10 * 10 * 10 = 16 + 12 = 28 \text{ olarak hesaplanılır.}$$

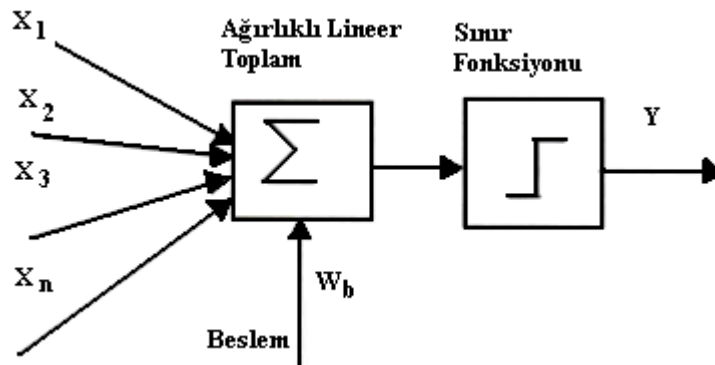
Sonuç olarak CENTROID yöntemi ile $28/5 = 5.6$ değeri elde edilir. BUS'ler en çok lineer ve lineer olmayan kontrolde, desen tanımda, maliye sistemlerinde, veri analizinde vb. kullanılmaktadır.

3.3.3. Uzman Sistemler Ve Yapay Sinir Ağları

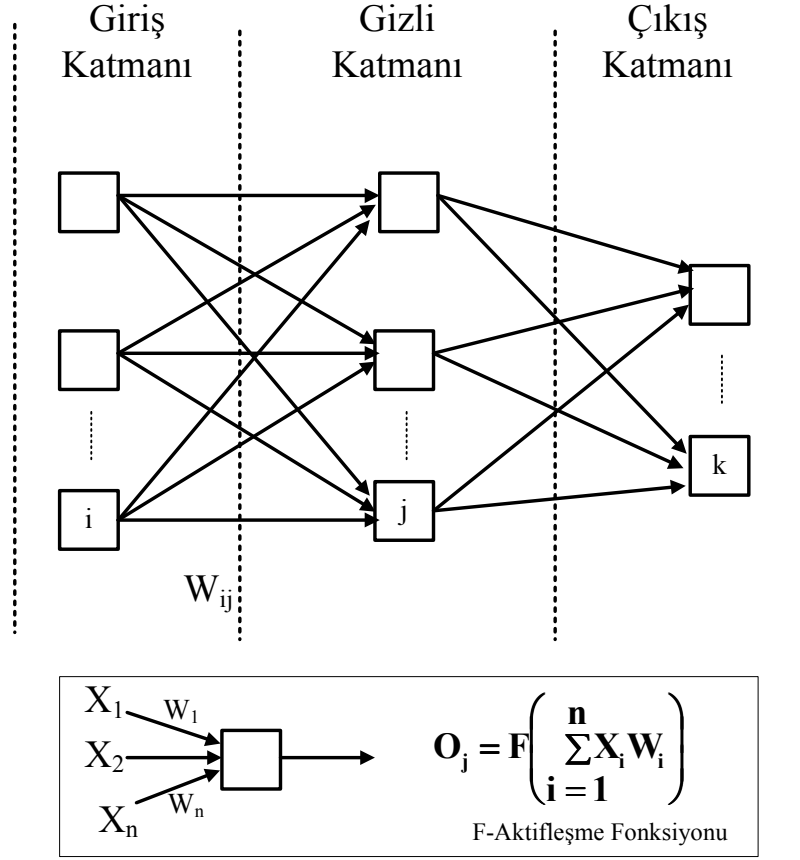
Son yıllarda Yumuşak Hesaplama yöntemleri gelişmekte olup geleneksel Yapay Zeka yöntemlerinin birbiri ile çeşitli kombinasyonları kullanmakla daha yüksek performanslı bilgi işlem sistemlerinin elde edildiği görülmektedir. Yukarıda baktığımız BUS'leri buna örnektir. Yapay Zeka'nın diğer bir alt dalı olan yapay sinir ağları ile US'lerin birleştirilerek sinir ağı uzman sistemler tasarlamak suretiyle US'lerin bazı eksikliklerini gidermek mümkündür. Bu bölümde böyle bir sistemin nasıl tasarlanacağı anlatılacaktır. Ama bundan önce YSA'nın ne olduğu hakkında kısa bir bilgi verilmesi, sonra YSA ile US'lerin kısa bir kıyaslaması yapılması yerinde olur.

Bir bilgisayar programı gibi ele alınabilen YSA girişleri ve çıkışı belli olan bir sistemde, girişler değiştiğinde sistemin önceden belirlenmiş bir çıkışı tanımasını sağlamaktadır. Böyle problemler desen, örnek, karakter ve benzeri tanıma işlemlerinde yaygın olarak karşılaşılmaktadır. YSA'nın temelini insandaki sinir ağının elektrik modeli teşkil etmektedir. Yani bir sinir ağı girişlerinden dahil olan veriler her girişin ağırlığı ile çarpılır ve sonra toplanır. Bu toplam değer, belirli bir sınır değerini aşarsa bu durumda bu toplam çıkışa iletilir. Aksi durumda çıkışa sıfır değeri iletilir. Burada problem, sonunda istenilen çıkış elde edilmesi için girişlerin ağırlıklarının düzenlenmesidir. Girişlerin ağırlıklarının düzenlenmesi bir çok yöntemle yapılmaktadır.

Şekil 3.4'de basit bir sinir ağının modeli gösterilmektedir. Burada X_i ağırlık giriş değerleri, W_i uygun girişin ağırlığı, giriş değerlerinin toplanmasını gösteren lineer toplam ve ise sınır fonksiyonunu göstermektedir. Genelde YSA'da çok katmanlı bir model kullanılır. Bu tür bir modelde giriş katmanı, ara veya gizli katman ve çıkış katmanı mevcuttur. Şekil 3.5'de üç katmanlı bir YSA modeli gösterilmektedir [43, 47].



Şekil 3.4. Bir sinirin modeli



Şekil 3.5. Çok katmanlı YSA modeli

3.3.3.1. Uzman Sistemler ile Yapay Sinir Ağlarının Karşılaştırılması

YSA'da hem dağıtılmış, hem de sayısal bilgiler işlenebilmektedir. Diğer taraftan US'lerde işlenen bilgiler belli derecede kesin ve açık olmak zorundadır. YSA'da ise kullanılan bilgiler saklı olmaktadır, Bilgi işlem yöntemi US'lerde temel olarak seridir ve karmaşıktır, yani hem sayı, hem de sembolik işlem yapılabilir. Yalnız son yıllarda US'lerde paralel işlem yöntemleri geliştirilmeye başlanmıştır. YSA'da ise bilgi işlem yöntemi olarak paralel işlem kullanılabilir ve yalnızca sayılar üzerinde işlem yapılabilir.

Tablo 3.2'de US'lerin ve YSA'nın benzerlikleri. Tablo 3.3'te ise farklılıkları gösterilmektedir. Bu tablolardan da görüldüğü gibi, bir US ile YSA arasındaki temel farklardan biri öğrenme olanağı ile ilgilidir. US'lerde öğrenme ya yapılamamakta ya zor yapılmaktadır [42]. YSA ise bu olay üzerinde kurulmaktadır. Bu yöntemler çoğu zaman

birbirini tamamlayıcı özelliklere sahiptirler. Bu yüzden son yıllarda her iki yöntemi birleştirerek, her birisinin zayıf taraflarını giderecek sistemlerin tasarlanması yaygınlaşmaktadır.

Tablo 3.2. US ve YSA'nın Karşılık Gelen Elemanları

Uzman Sistemler	Yapay Sinir Ağları
Şartlar ve Sonuçlar	İşlem elemanları (düğüm)ler
Kurallar	Bağlantılar
Belirsizlik değerleri	Ağırlık değerleri
Gerçekleri toplama	Toplama Fonksiyonu
Kuralları çalıştırma	Çıkış fonksiyonu

Tablo 3.3. US'ler ve YSA Arasındaki Farklılıklar

Parametre	Uzman Sistemler	Yapay Sinir Ağları
Kullanıcı Arabirimi	Mevcuttur	Mevcut değildir
Açıklama Yeteneği	Mevcuttur	Mevcut değildir
Uzman kişi	İhtiyacı vardır	İhtiyacı yoktur
Örnekler	İhtiyacı yoktur	İhtiyacı vardır
Problemin boyutu çoğaldığında bakım	Zorlaşır	Değişmemektedir
Oluşturulması	Kurallar, hiyerarşik yapılar vs. geliştirmekle	Eğitilmekle
Bilgi işlemi	Seri	Paralel
Tam değerlerin belli olması	Şarttır	Şart değildir

3.3.3.2. Uzman Sinir Ağları

YSA ile US'ler birlikte kullanılarak oluşturulan sistem, uzman sinir ağları veya kısaca uzman ağlar diye adlandırılır. Böyle bir sistem kurulması için bazı ön şartlar taşınmalıdır. Uzman ağların performansının hem YSA'dan hem de US'in performanslarından üstün olması gerekmektedir. Özellikle YSA'daki öğrenme yeteneğinin

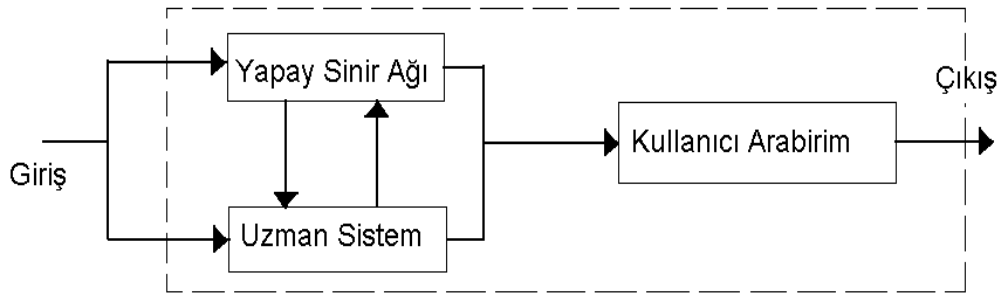
kullanılabilirliği çok önemlidir. Ayrıca YSA'nın paralellik özelliği US'e eklendiğinde oluşan uzman ağı daha hızlı çalışabilecektir. Diğer taraftan bir US'in sahip olduğu kullanıcı arabirimi ve açıklama alt sistemi, YSA'nın daha açık bir bilgi yapısına ve kullanıcı ile daha etkin bir diyaloga sahip olmalarına imkan sağlayacaktır.

Uzman ağlarda bilgi tabanındaki gereksiz kurallar belirlenerek silinebilmekte ve gereken yeni kurallar eklenebilmektedir. Fazla ve birbirleriyle zıt olan kurallarda ortaya çıkarılabilmektedir.

Uzman ağların tasarlanması için iki yaklaşım mevcuttur.

- Fonksiyonel bütünleşme
- Yapısal bütünleşme

Uzman ağlar fonksiyonel bütünleşme yöntemi ile tasarlandığında Şekil 3.6'daki gibi YSA ve US iki ayrı eleman olarak görülür. Çözülmesi istenen büyük bir problem daha küçük alt problemlere bölünür ve her bir alt problem, ya US yada YSA tarafından işlenir. Bunların arasında bilgi alış verişi olanağı mevcuttur. Genelde sistemin kontrolü US kısmında olur.



Şekil 3.6. YSA ile US'in fonksiyonel birleştirilmesi

Yapısal birleştirme yaklaşımında YSA, Şekil 3.7'de görüldüğü gibi US'in içeriğine gömülmektedir. US'in fonksiyonlarının tamamı veya bazıları YSA tarafından yapılmakta ve kurallar ağı yapısı içinde kodlanmaktadır. Kuralların koşul ve eylemleri işlem elemanları. Belirsizlik katsayıları ise ağırlık değerleri olarak ele alınır. Dolayısıyla US'deki çıkarım mekanizması yerine YSA kullanılmaktadır. Ayrıca YSA bilgi tabanı görevini de üzerine almaktadır.

Bir kuralın genel yapısı;

Eğer A_1 ve $A_{21}, \dots$, **O Halde** B

şeklinde ise, A_i koşulu, B sonuç veya çıkarımı (consequent). A_i -lerin mantıksal çarpımı kuralın varsayımını (antecedent) göstermektedir. Kurallar tabanının YSA'nın içeriğine nasıl gömüldüğünü görmek için bir örnek inceleyelim. Aşağıdaki kuralların verildiğini varsayalım:

Kural 1: **Eğer** A ve B ise, **O Halde** X.

Kural 2: **Eğer** B ve C ise, **O Halde** Y.

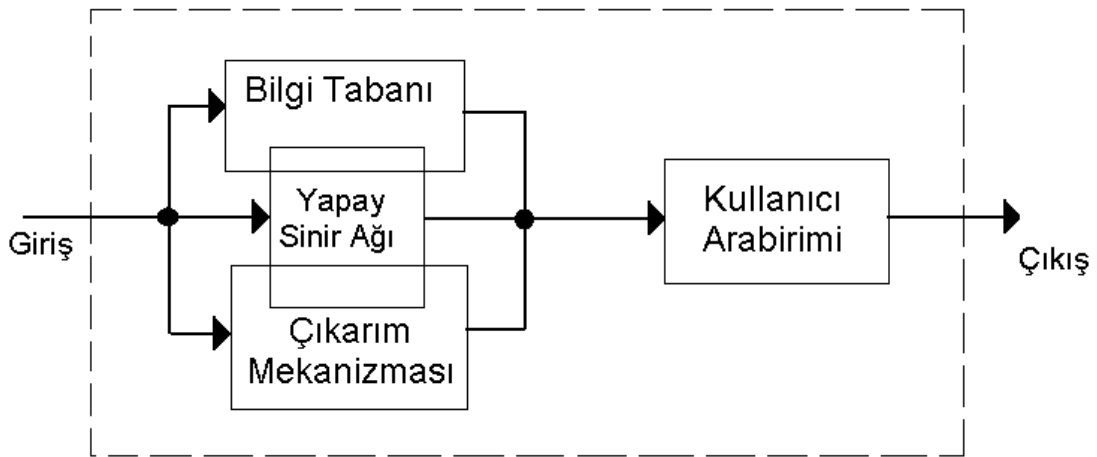
Kural 3: **Eğer** A ve C ise, **O Halde** X.

Kural 4: **Eğer** C ve D ve E ise, **O Halde** Z.

Eğer kurallar bu şekilde verilmemişse, onları benzer şekle getirmek gerekmektedir.

Örneğin Kural 1 ve Kural 3 birleştirilerek şu şekilde de gösterilebilir:

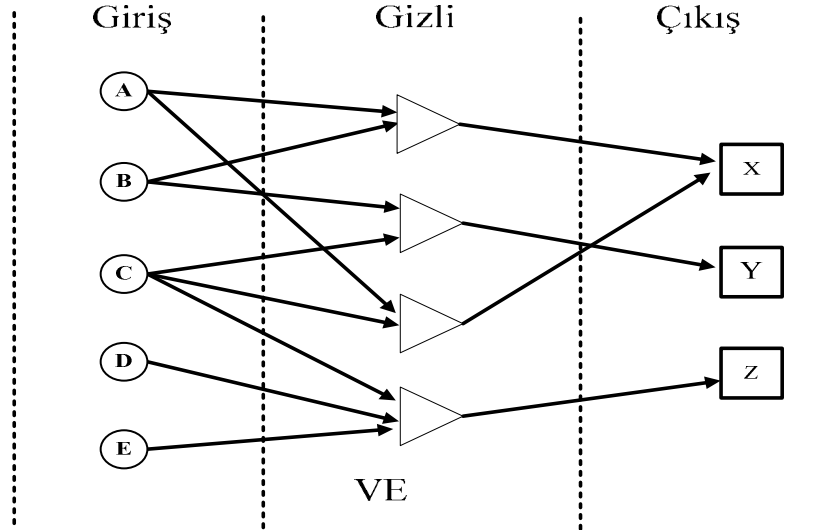
Eğer (A ve B) veya (A ve C) ise, **O Halde** X.



Şekil 3.7. YSA ile US'in yapısal birleşmesi

Daha önceden de belirtildiği gibi, her bir kuralın varsayım (koşul) ve çıkarım (sonuç) kısımları mevcut olmaktadır. Uzman ağlarda kuralların varsayım kısmı gizli katmana, her bir koşul belli bir nitelik veya kavram düğümüne ve sonuç kısmı ise belirli bir kavram düğümüne karşılık gelmektedir. Her bir koşul düğümü, sonuç düğümü ile bağlı olan varsayım düğümüne bağlı bulunmaktadır. Böyle bir yapıda bir kural, düğümler arasındaki ağırlık değerine uygun gelmektedir. Burada gizli katman, kuralın varsayım kısmındaki mantıksal VE işlemlerini temsil etmektedir. Gizli katman bazen VE birimi olarak da adlandırılır. Yukarıda gösterilen 4 kural için anlatılan yöntemle tasarlanmış uzman ağı Şekil 3.8'de gösterilmektedir.

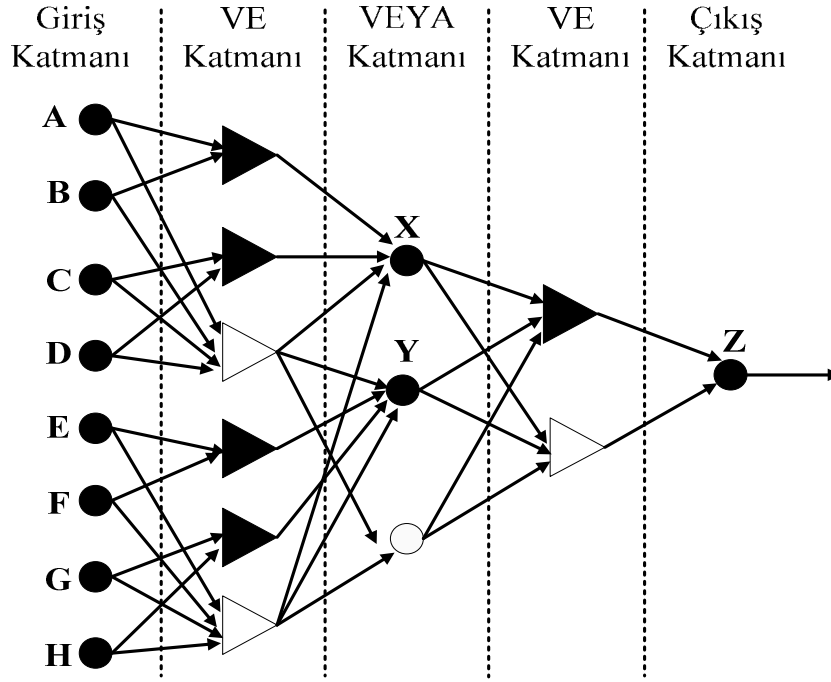
Görüldüğü gibi, kurallara dayalı bir uzman ağı oluşturmak zor değildir ve kurallar çok seviyeli bir hiyerarşi oluşturabildiğinden dolayı katmanların sayısı da çoğalabilir. Burada temel problem, öğrenmenin nasıl yapılacağıdır. Bunun yapılabilmesi için ise ağ ilişkilerinin, özellikle giriş katmanındakilerin başlangıç ağırlıkları verilmelidir. Bir çok durumda bu ağırlıklar -1 ve 1 ile sınırlandırılır.



Şekil 3.8. Kurala dayalı sinir ağı

Giriş katman için kuralların sayısı P ise her bir giriş ilişkisi için ağırlık değeri $1/p$ olarak kabul edilebilir. Bu, kurallardaki varsayımların meydana gelmesi olanaklarının eşitliğine dayanır. Tabii ki eğer tasarımcı herhangi bir kuraldaki varsayımın diğerlerine göre daha ağırlıklı olduğunu biliyorsa, bu ağırlıklar eşit olmayabilir. Burada bir diğer problem de, eğitme zamanı yeni kuralların eklenebilmesi imkanındır. Yeni kurallar eklendiğinde (bu kurallar mevcut olmayan kurallar da olabilir) VE katmanından sonraki katmanlardaki tüm düğümlerle ilişkilendirilmelidir. Yeni eklenen kurallar için başlangıç ağırlık değeri daha zayıf kabul edilmelidir. Genelde, eğer eklenen kuralların sayısı q ise bu kurallar için başlangıç ağırlık değeri $-1/q$ olarak kabul edilir. Şekil 3.9'da aşağıdaki kurallar için bir uzman sinir ağı gösterilmiştir.

Şekil 3.9'da görüldüğü gibi ağ çok katmanlıdır ve bu katmanlardan ikisi VE ve biri VEYA katmanıdır. Şekilde siyahla gösterilmiş elemanlar mevcut kuralları; içi boş elemanlar ise eklenen kuralları; aynı zamanda sürekli hatlar mevcut olan ilişkileri; kesit hatlar ise sonradan eklenen ilişkileri göstermektedir.



Şekil 3.9. Kurala dayalı, uzman sinir ağına kuralların eklenmesi

Kural tabanı:

Eğer A ve B ise, O halde X

Eğer C ve D ise, O halde X

Eğer E ve F ise, O halde Y

Eğer G ve H ise, O halde Y

Eğer X ve Y ise, O halde Z.

Kurala bir güvenilirlik değeri verilmişse (örneğin, MYCIN'de olduğu gibi) bu değer kuralın başlangıç ağırlık değeri göstericisi olabilir [42, 43, 47].

3.3.4. Bulanık Sinir Ağları

Bulanık sinir ağlarında (BSA), bulanık fikirler sinir ağları ile birleştirilmiştir. Bulanık sinir ağları, bulanık sonuçlandırmanın sinir ağı prensiplerine uygulandığı sistemlerdir. Bu sistemler yalın haldeki bulanık denetleyicilerden daha iyi şekilde çalışırlar. Bu tip bulanık denetleyiciler, bir problem için üyelik fonksiyonlarının düzenlenmesi ve bulanık kuralların tanımlanmasının otomatik olarak yapılabilmesi bakımından pek çok avantaja sahiptirler. BSA'nın topolojisi sebepsellik veya kurala dayalı yaklaşım olabilir.

Genelde bulanık kuralla dayalı sinir ağı, giriş ve çıkış katmanlarından başka üç katmana daha sahip olurlar. Bunlar; giriş bulanık katmanı VE katmanı çıkış bulanık katmanı olarak adlandırılırlar. Giriş değerleri bulanık küme birimine iletilir ve iletilen değer aktifleşme seviyesi olarak üyelik derecesine dönüştürülür. VE katmanı, ona dahil olan üyelik derecelerinin minimumunu bulacak ve çıkış bulanık katmanı VE katmanından dahil olan değerlerin ya maksimumunu, ya da toplamını bularak çıkışa iletecektir. Tabii ki burada VE katmanı bulanık kurallara uygun olarak oluşturulmaktadır. Bulanık kuralları aşağıda verilmiş olan bulanık uzman ağı şekil 3.10'da gösterilmektedir [42, 43, 47].

Bulanık kurallar (burada X_1, X_3, Y_1, Y_2, Z_1 ve Z_3 küçük, orta, büyük vb. gibi dilsel değişkenlerdir):

Eğer $X=X_1$ ve $Y=Y_1$ ise, **O Halde** $Z=Z_1$.

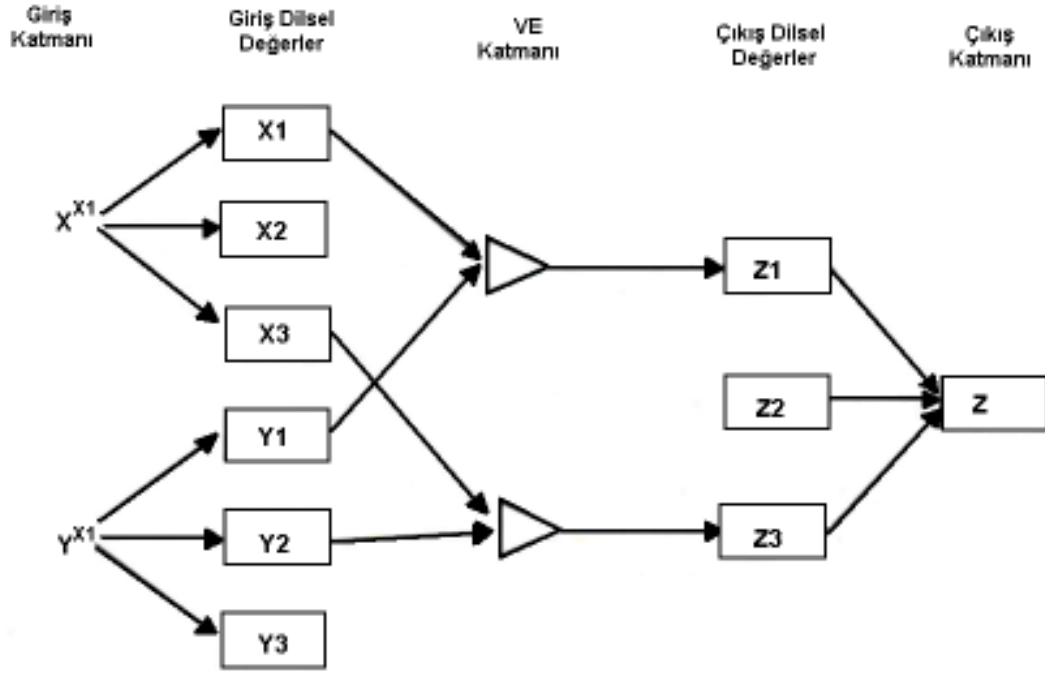
Eğer $X=X_3$ ve $Y=Y_2$ ise, **O Halde** $Z=Z_3$.

Böylece bulanık uzman ağının her bir katmanında aktifleşme fonksiyonu aşağıdaki gibi hesaplanır:

- > Giriş katmanı: aktifleşme fonksiyonu yoktur.
- > Giriş bulanık katmanda: üyelik fonksiyonu olarak hesaplanır.
- > VE katmanında: MIN fonksiyonu olarak hesaplanır.
- > Çıkış bulanık katmanı: MAX veya başka bir fonksiyon olarak hesaplanılabilir.
- > Çıkış katmanı: $F(a) = a / \sum_i X_i \delta_{ji}$ (burada X_i çıkış bulanık katmanından çıkış

katmanına gelen değer. δ_{ji} bulanık kümenin i. elemanından çıkış katmanının j. elemanı arasındaki ilişkinin genişliğidir.

Bulanık uzman ağlar, ağırlıkları düzenlenerek eğitilirler. Yalnızca üyelik fonksiyonları değil, ilişki ağırlıkları da değiştirilebilir. Burada geriye yayılma ve diğer eğitim yöntemleri kullanılabilir.



Şekil 3.10. Bulanık uzman ağı modeli

3.4. Uzman Sistem Tasarımında Kullanılan Diller, Kabuklar Ve Araçlar

Bir problemin tanımlanmasında temel olarak yapılması gereken şey problemin en iyi şekilde nasıl modellendirileceğine karar vermektir. Bazen en iyi ikilemleri seçmeye yardımcı olmak için daha önceden edinilmiş deneyimler mevcuttur. Deneyimler, ücret bordrosu hazırlarken geleneksel yordamsal (prosedürel) programlamanın kullanılmamasını tavsiye etmektedir. Hazır ticari programlar mevcut ise bunların kullanılmasının tercih sebebi olması yine deneyimlerin bir tavsiyesidir. Bir ikilem seçerken kullanılan genel rehber ilk olarak en geleneksel olanı düşünmekle başlar ki bu da geleneksel programlamadır. Bunun sebebi ise geleneksel programlama üzerinde bugüne kadar çok durulması ve ticari olarak çok geniş ölçekte ve alanda hazır yazılımların yada paketlerin bulunmasıdır. Eğer bir problem geleneksel programlama ile etkin bir şekilde çözülmiyorsa o zaman yapılması gereken şey geleneksel olmayan bir paradigma olan Yapay Zeka'ya doğru yönelmektir.

Uzman sistemler yapay zekanın bir dalı olmasına rağmen, uzman sistemler için özel olarak geliştirilen ve kullanılan dillerin çoğu yapay zeka için kullanılanlardan oldukça farklıdır. LISP yapay zeka problemleri programlaması için kullanılırken uzman sistemler

için geliştirilen PROLOG oldukça farklı yapıda olup yapay zeka için etkin bir şekilde kullanılması söz konusu değildir. Benzer şekilde IPL-II, SAIL, CONNIVER, KRL ve SMALTALK yapay zeka için kullanılan diller olup uzman sistemlerde geniş bir kullanım alanı bulmayıp sadece araştırmalarda kullanılmaktadır [7, 42, 44].

Bir uzman sistem dili yüksek seviyeli bir dil olmalı ve C ve LISP gibi programlama dillerinden oldukça farklı olmalıdır. Çünkü uzman sistemin doğası bu dillerin yalnızca ve yalnızca uzman sistemlerin yazımında kullanılmasını uygun kılmakta olup genel amaçlı programlama için etkin ve verimli kılmamaktadır. Birçok durumda uzman sistemlerin yazılımda uzman sistem dilinden geçici olarak çıkıp bir fonksiyonu yordamsal yada yapısal bir dilde yazmak ve sonra uzman sistem diline geri dönmek gereği görülmektedir.

Uzman sistemler dilleri ve yordamsal diller arasındaki ilk fonksiyonel fark, sunumun odaklanması ile ilgilidir. Yordamsal diller verinin sunumu için esnek ve güçlü teknikler üzerinde odaklaşır. Örneğin, diziler, kayıtlar, bağlı listeler, yığınlar, kuyruklar ve ağaçlar gibi veri yapıları çok kolayca yaratılır ve ustalıkla yönetilir. Modula 2 ve Ada gibi programlama dilleri modüller ve paketler gibi yapıları kullanarak veri soyutlaması için kapsülleme imkânını sağlar. Bu ise bir programı oluşturmak için operatörler ve kontrol ifadeleri olarak bilinen metotlar ile gerçekleştirilen soyutlamanın bir katmanını oluşturur. Veri ve verileri yöneten metotların içi içe dokunduğu yada örüldüğü görülür. Uzman sistemlerde ise buna zıt olarak bilginin sunumunda esnek ve güçlü yollar mevcuttur. Uzman sistem paradigması soyutlamanın iki katmanına izin verir: veri ve bilgi soyutlaması, uzman sistemler veriyi ve veri üzerinde işlem yapacak olan metotları birbirinden ayırır. Buna örnek olarak kural tabanlı bir uzman sistem dilinde kuralların (bilgi soyutlaması) ve gerçeklerin (veri soyutlaması) ayrılması gösterilebilir [42].

Uzman sistem tasarımında kullanılan kavramlar:

Dil: Özel yazım kuralına sahip komutların çeviricisidir. Bir uzman sistem dili ayrıca dilin ifadelerini yada komut zincirini çalıştıracak çıkarım mekanizmasını da sağlar. Uygulamaya bağlı olarak çıkarım mekanizması ileriye doğru, geriye doğru yada her iki yöne doğru zincirleme yapabilir. Bu tanımlamalar ışığında PROLOG bir uzman sistem dili olup LISP bu kavrama girmemektedir.

Araç: Uygulama programlarının geliştirilmesinin, bakımının sağlamanın ve tesliminin gerçekleştirilmesinin yardımıyla kullanılan programlama dili ve faydalı programların hepsine birden araçlar denilmektedir. Faydalı programlar metin ve grafik editörleri, hata ayıklayıcıları, dosya yönetimi ve hatta kod üreticileri içerebilirler.

Kabuk: Kabuklar hazır hale getirilmiş, çıkarım mekanizması ve bilgi saklama özellikleri ile donatılmış sistemler olup sadece alan bilgisi olmayan içi boş uzman sistemlerdir. Bir sonuç mekanizması ile donatılmışlardır ve spesifik bir formata göre bilgi girişine sağlarlar. Diğer bazı özellikleri, hypertext yazma araçları, kullanıcı için kullanımı kolay arabirim tasarımı, listeleri, stringleri ve objeleri işleyebilme, harici programlarla ve veritabanları ile anlaşabilme şeklinde sıralanabilir. Kabuklar, programlama dillerine benzerler, fakat programlama dilleri gibi birçok uygulama ile ilişkili olamazlar. İlişkili oldukları uygulama açısından sınırlıdır. Ayrıca kullanıcının kendisinin özel çıkarım mekanizması geliştirmesine imkan veren daha gelişmiş sistemler de vardır. Buna örnek olarak EMYCIN (boş MYCIN) kabuğu verilebilir. Bu kabuk MYCIN uzman sisteminden tıbbi bilginin arındırılmasıyla yapılmıştır. Tıbbi bilgi çıkarılarak EMYCIN yaratılmış ve bu kabuk geriye doğru zincirlemeyi kullanan diğer başvuru sistemler hakkında bilgiyi içerecek şekilde kullanılabilir haldedir. Böylelikle MYCIN yazılımına ait olan çıkarım mekanizması ve kullanıcı ara yüzü yeniden kullanılır hale gelmiştir. şeklinde tanımlanabilirler [42].

Günümüzde piyasada oldukça çok sayıda yazılım araçları ve kabuklar mevcuttur. Bunların büyük çoğunluğu ticari amaçla geliştirilmiş, az bir kısmı ise eğitim amacıyla da kullanılabilir. Bir uzman sistem tasarımcısının bunları iyi bilmesi, onlar arasından kendi işine yarayanı seçmesi önemlidir.

4. GELİŞTİRİLEN WEB TABANLI UZMAN SİSTEMİN GENEL YAPISI VE TASARIMI

Bu bölümde, geliştirilen web tabanlı akıllı öğretim sistemine ait tüm bilgiler detaylı olarak verilmektedir. Bu bağlamda sistemin yapısı, sistemi oluşturan bileşenler, sistemde kullanılan modeller, teknolojiler, yazılımlar anlatılmaktadır.

Geliştirilen akıllı öğretim sisteminde konu alanı olarak, Fırat Üniversitesi, Teknik Eğitim Fakültesi, Elektronik-Bilgisayar Eğitimi Bölümü, Elektronik Öğretmenliği 5.döneminde yer alan “Kontrol Sistemleri I” dersi belirlenmiştir. Geliştirilen akıllı öğretim sistemi, üzerinde MATLAB, MWS ve bir web sunucu yazılımının yüklü olduğu bir sunucu üzerine yüklenerek yerel bir ağ üzerinde de çalışmaktadır. Sistemdeki bileşenlerin çalışmasında uygulama aracı olarak Macromedia Dreamweaver 8 içerisinde yer alan PHP uygulamaları geliştirme programı kullanılmıştır.

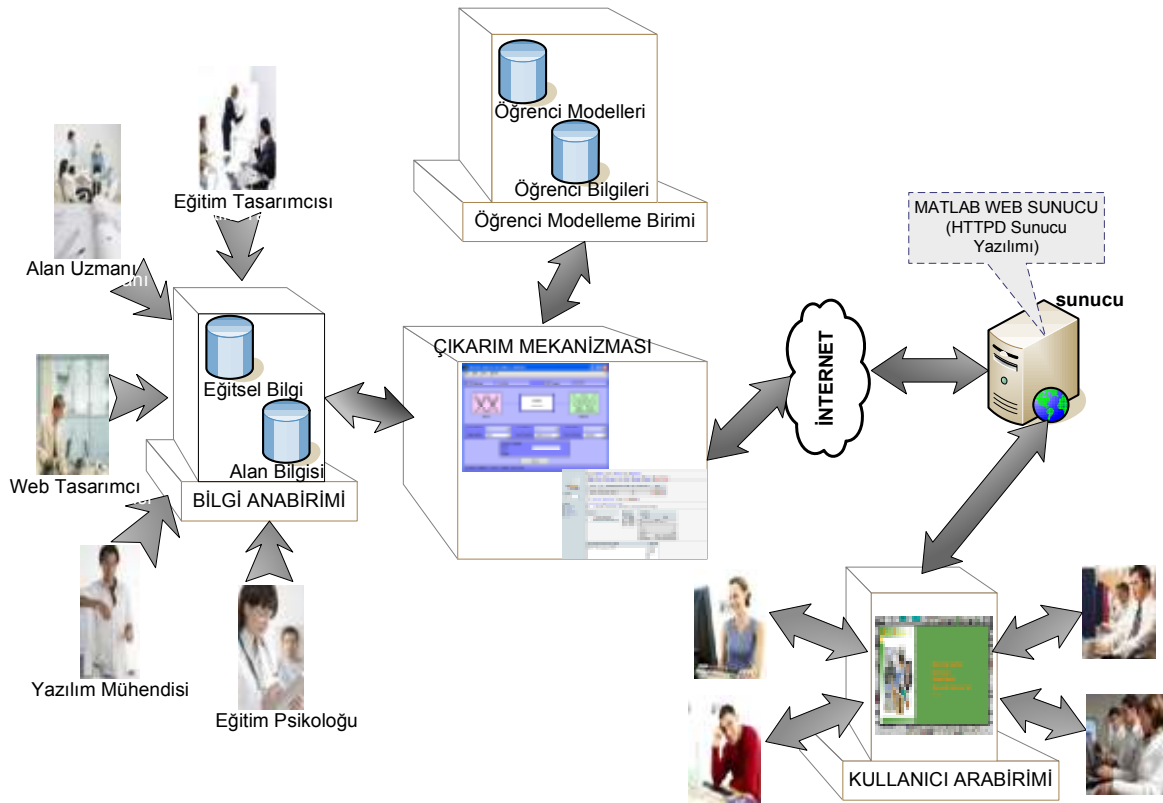
4.1 Geliştirilen Sistemin Genel Amacı

Geliştirilen sistemde amaç, öğrencinin özelliklerini yansıtan bir öğrenci modeli oluşturarak, bu model doğrultusunda öğrencilere kişiselleştirilmiş seçenekleri otomatik olarak sunmaktır. Böylece öğrencinin ihtiyaçlarına cevap veren, öğrenciye en uygun içeriği sunan ve öğrencinin seviyesini çok daha iyi ölçen bir sistem oluşturulabilmektedir. Sistem elde ettiği verileri değerlendirerek sürekli kendini güncelleyebilmeli, böylece zamanla öğrenci profilinde meydana gelecek değişikliklere uyum sağlayabilmelidir. Bu nedenle tasarlanan web tabanlı akıllı eğitim sisteminin görevini eksiksiz olarak yerine getirebilmesi için, sistemin öğrenciyi tanıması gerekir. Bu her öğrenci için sistemde bir öğrenci modelinin olmasını gerektirir. Geliştirilen öğrenci modelinde uzun dönemli ve kısa dönemli olmak üzere iki farklı tür yer alır. Uzun dönemli öğrenci modelinde, öğrencinin hemen değişmeyen, uzun süre geçerliliğini koruyan bilgi düzeyi, amaç ve öğrenme metodu gibi bilgileri tutulur. Kısa dönemli öğrenci modelinde, öğrencinin sadece bir oturum süresince geçerli olan bilgileri tutulur. Bunlar öğrencinin sürekli takip edilen davranışlarına ilişkin bilgilerdir. Örneğin bir problem çözümü sırasında seçtiği yöntem, hata yaptıysa bu hataya sebep olabilecek bilgi eksikliğinin yada hatalı algılamının ne olduğu, bir konuya harcadığı zaman gibi bilgiler kısa dönemli öğrenci modelinde yer alır [48, 49].

Sistem, öğrenci modelindeki öğrenciye ait bilgileri değerlendirerek alan bilgisinden gerekli olan içerik bilgisini ve soruları alarak kullanıcıya sunar. Kullanıcıdan alınan bilgilerle alan modelindeki bilgiler değerlendirilerek öğrenci modelinin güncellemesi yapılır ve böylece sistemin kendini geliştirmesi sağlanmış olur.

4.2 Geliştirilen Sistemin Mimari Yapısı

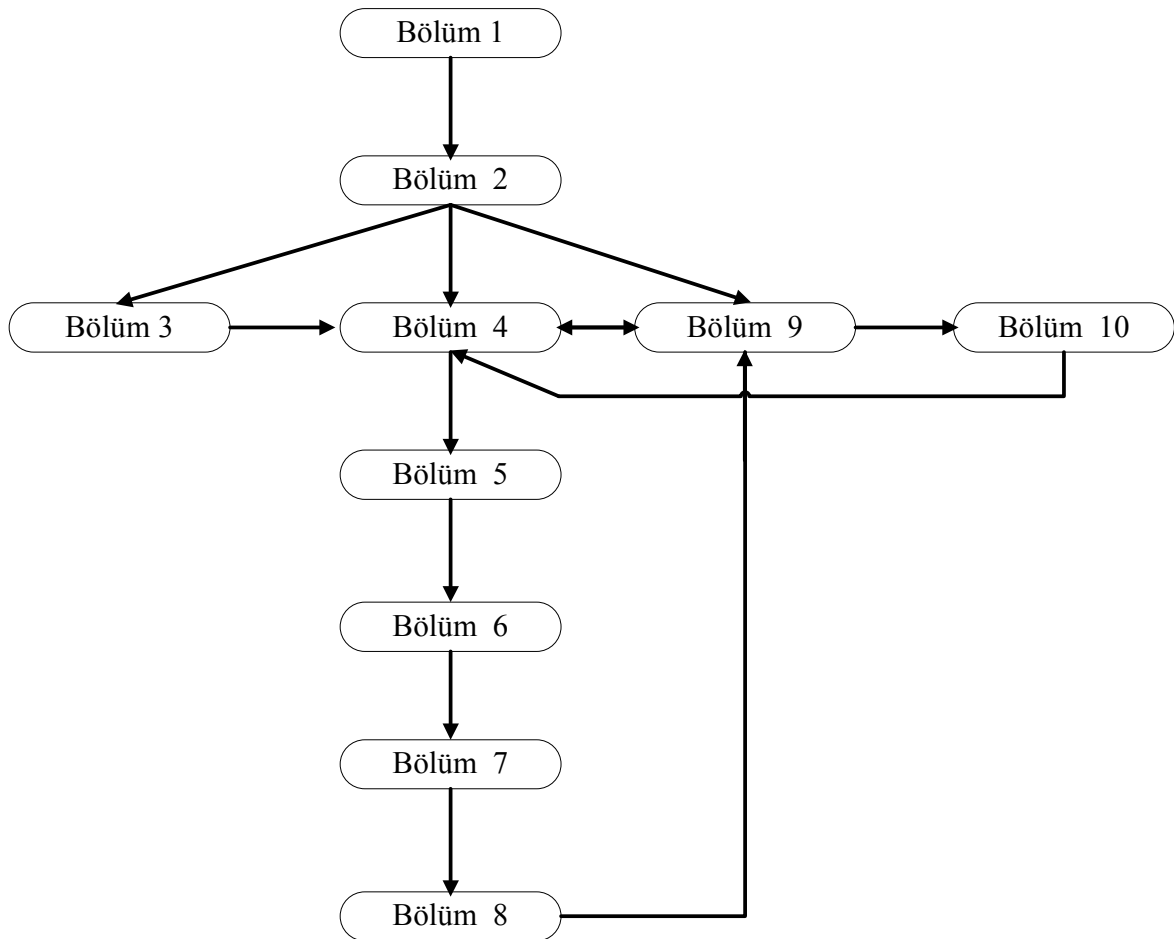
Uyarlanabilir bir öğretim sistemi olarak tasarlanan sistem; öğrencinin bilmesi gereken konuları içeren bilgi ana birimi, çıkarım mekanizması, öğrenciyle iletişim kurulmasını sağlayan kullanıcı arabirimi ve öğrenci modelleme biriminden oluşmaktadır. Sistemi oluşturan bileşenlerden kullanıcı ara biriminin tasarımı ve kodlaması için Macromedia Dreamweaver 8 yazılımı PHP teknolojisiyle birlikte kullanılmıştır. Veri tabanı olarak Mysql, web sunucusu olarak Apache Web Server tercih edilmiştir.



Şekil 4.1. Geliştirilen sistemin mimari yapısı

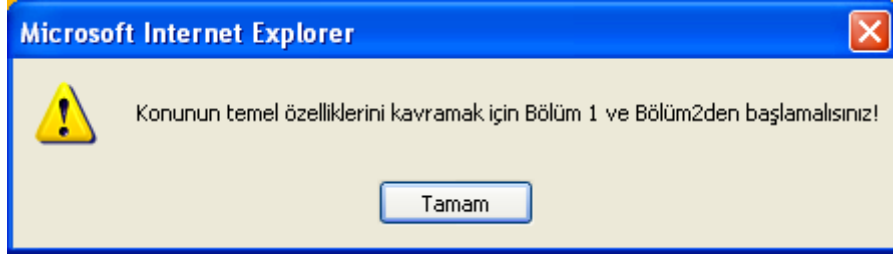
4.2.1. Bilgi Anabirimi

Bilgi anabirimi, öğrenciye sunulacak olan konu ile ilgili kuralları, bilgileri, soruları ve çözümleri içeren bölümdür. Sistemde herhangi bir alan bilgisine ait öğretim programının oluşturulması için alan bilgisinin modellenmesi gerekmektedir. Bu amaçla sistemin bilgi anabirimi modeli oluşturulur. Bilgi anabirimi modelinin alan bilgisi parçası oluşturulurken; alan bilgisi kontrol dersi olarak düşünülmüştür. Alan bilgisi, bir ders için oluşturulan bölümler, bölümlere ait kavramlar, bölüm sonu testleri olarak anlamlı bilgi parçaları halinde temsil edilmiştir. Bu anlamlı bilgi parçaları kullanılarak Şekil 4.2'deki gibi bir konu planı [50] çıkarılmıştır. Hazırlanan bu konu planında amaç öğrencinin dersi daha iyi kavramasını sağlamaktır.



Şekil 4.2. Konu planı

Şekil 4.2'ye göre öğrenciler Kontrol Sistemleri I dersindeki Bölüm 1 ve Bölüm 2'deki konuları öğrenmeden diğer konulara geçiş yapamamaktadır ve ilgili konuyu öğrenmek için şekildeki ok sırasını takip etmek zorundadır. Öğrencilerin yanlış sırayı takip etmesi halinde sistem Şekil 4.3'deki gibi uyarı verip doğru yolu gösterir.



Şekil 4.3. Uyarı mesajı

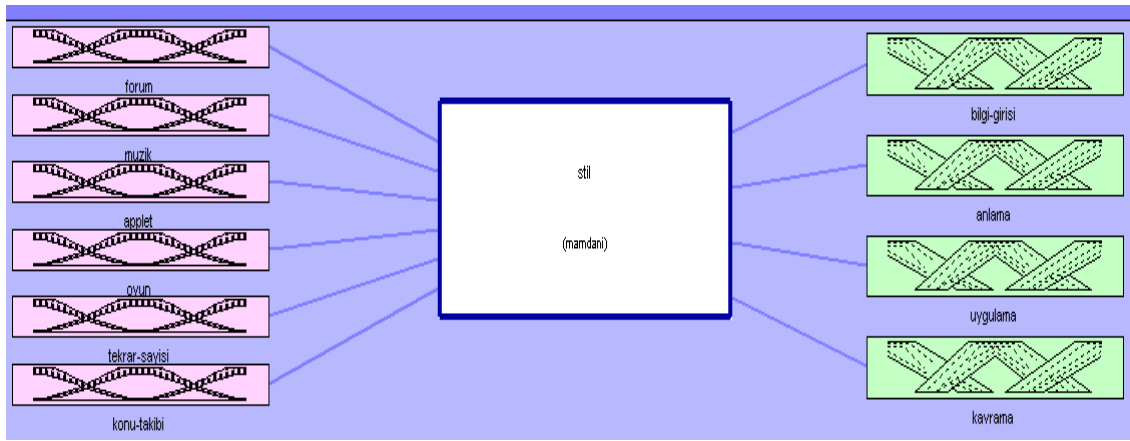
Alandaki her konu Dreamweaver tasarım ortamında hazırlanan .php uzantısına sahip dosyalar içerisinde metin ve şekiller yardımıyla açıklanmıştır. Öğrenci sisteme bağlandığında, öncelikle sahip olduğu öğrenci numarası ile veritabanı kontrol edilerek, en son hangi konuyu öğrenmeye çalıştığı belirlenmekte ve öğrenciye bulunduğu durumla ilgili rehberlik yapılmaktadır. Her bir konunun sonunda, konuyla ilgili çoktan seçmeli soruların öğrenci tarafından yanıtlanması gerekmektedir.

Tasarlanan sistemde, konulara ait sayfaların öğrenci tarafından ziyaret edilip edilmediğinin belirlenmesi ve doğru bir şekilde öğrencinin karşısına gelmesi için, öğrenci sisteme bağlanarak oturum açtıktan sonra, PHP nesnelerinden Oturum nesnesi kullanılarak kontrol yapılmaktadır. Oturum nesnesi, verilerin oturum boyunca saklanmasını sağlamaktadır. Oturum sayesinde kullanıcıya ait veriler sayfalar arasında kullanılmaktadır. Bir oturum, web tarayıcısının açılıp kapanmasına kadar geçen süredir. Öğrenci sisteme bağlandığı andan itibaren, sadece ona ait özel değişkenler kullanılarak, bulunduğu konu içerisindeki ziyaret sırasını ve hangi sayfaları ziyaret ettiğini belirlemek mümkün olmaktadır. Bu şekilde, öğrenciye tasarımcının belirlediği sayfa yönlendirmesi yapılmaktadır.

Bilgi ana birimindeki eğitsel bilgi öğrenme taktiklerine karar verme bilgisini içerir. Eğitsel bilgi öğrenme stratejilerine ve metotlarına göre belirlenen pedagojik kararlardır.

4.2.2. Çıkarım Mekanizması

Kural tabanı sistemde tanımlanan kurallar dizisini kullanarak bilgi ana biriminden anlamlı sonuçlar çıkarmakla sorumludur. Öğrenciye uygun ders içeriği oluşturabilen kurallar listesi halinde oluşturulan kural tabanı fuzzy2 toolbox [51-53] ortamında yazılmıştır.



Şekil 4.4. Sistemin Fuzzy2 toolbox’da hazırlanmış çıkarım mekanizması

Şekil 4.4’de görüldüğü gibi Fuzzy2 toolbox’da tasarlanan sistemin çıkarım mekanizması 6 giriş ve 4 çıkıştan oluşmuştur. Sistem girişleri olarak Tablo 4.1’de görülen forum, muzik, applet oyun, tekrar-sayisi, konu-takibi gibi öğrencinin sistem kullanım parametreleri; çıkışları olarak da Felder-Silverman’ın öğrenme stilleri modelinden hareketle bilgi-girisi, anlama, uygulama, kavrama olarak seçilmiştir.

Tablo 4.1. Kural tablosundan bir bölüm

1. If (forum is az) and (muzik is az) and (applet is az) and (oyun is az) and (tekrar-sayisi is kotu) and (konu-takibi is kotu) then (bilgi-girisi is sozel)(anlama is sirali)(uygulama is dusunsel)(kavrama is sezgisel)
2. If (forum is orta) and (muzik is orta) and (applet is orta) and (oyun is cok) and (tekrar-sayisi is orta) and (konu-takibi is enough) then (bilgi-girisi is gorsel)(anlama is global)(uygulama is deneysel)(kavrama is duyusal)
3. If (forum is cok) and (muzik is az) and (applet is az) and (oyun is az) and (tekrar-sayisi is kotu) and (konu-takibi is kotu) then (bilgi-girisi is sozel)(anlama is sirali)(uygulama is deneysel)(kavrama is sezgisel)
4. If (forum is cok) and (muzik is az) and (applet is az) and (oyun is orta) and (tekrar-sayisi is iyi) and (konu-takibi is kotu) then (bilgi-girisi is sozel)(anlama is sirali)(uygulama is deneysel)(kavrama is sezgisel)

Tablo 4.1'in devamı

5. If (forum is çok) and (muzik is az) and (applet is az) and (oyun is az) and (tekrar-sayisi is iyi) and (konu-takibi is kötü) then (bilgi-girisi is sozel)(anlama is sirali)(uygulama is deneysel) (kavrama is sezgisel)
6. If (forum is az) and (muzik is az) and (applet is az) and (oyun is az) and (tekrar-sayisi is iyi) and (konu-takibi is iyi) then (bilgi-girisi is sozel)(anlama is global)(uygulama is deneysel) (kavrama is duysal)
7. If (forum is az) and (muzik is az) and (applet is az) and (oyun is az) and (tekrar-sayisi is kötü) and (konu-takibi is iyi) then (bilgi-girisi is sozel)(anlama is global)(uygulama is dusunsel)(kavrama is sezgisel)
.....

4.2.3. Kullanıcı Arabirimi

Akıllı öğretim sistemi ile kullanıcıların etkileşimini sağlayan kullanıcı arabiriminin tasarlanmasındaki amaç, hazırlanan sistemin görsel nesnelerle temsil edilmesidir. Bu amaç doğrultusunda, kullanıcı arabiriminin sundukları ile öğrencinin zihnindekilerin birebir örtüşmesi gerekir. Ancak bu şekilde sağlıklı bir iletişim kurulabilir.

Geliştirilen sistemde kullanıcı arabiriminin tasarımında sarı zemin üzerine siyah metin rengi ile kolay takip edilebilir bir tasarım oluşturulmuştur. Önemli kavramlar üzerinde dikkat çekilmek istendiğinde parlak renkler tercih edilmiştir.

4.2.4. Öğrenci Modelleme Birimi

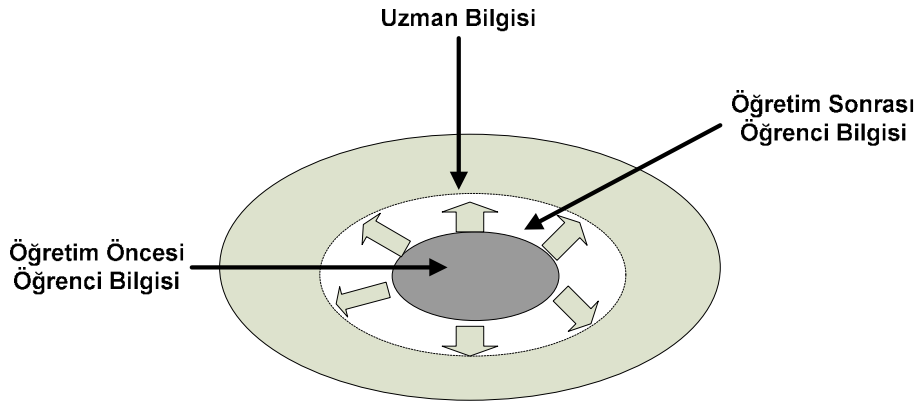
Bir akıllı öğretim sistemini oluşturan tüm bileşenler, birbirlerini destekleyip tamamladıkları için çok önemlidirler. Özellikle öğrenci modelleme bileşeni içlerinde en önemli olanıdır. Çünkü öğrenci modeli kaliteli değilse, sistem öğrenciyi yeterince tanıyamaz ve diğer birimlerin işlevleri ne kadar kaliteli olursa olsun, alınan kararların tatmin edici olmamasına sebep olur. Bununla beraber öğrenci modelleme bileşeni akıllı öğretim sistemlerinde en çok engelle sahip olan alandır. Öğrenci davranışlarından bilgi çıkarabilme probleminin sonucunda bazı engeller bulunmaktadır. Bu engeller;

- Akıllı öğretim sistemi ortamındaki büyük miktardaki belirsizlikler,
- Öğrenciden elde edilen bilgilerin sağlam dayanağı olmayan çelişkili bilgilerle örtüşmesi,
- Her öğrencinin diğerinden farklı ve kendine ait bir öğrenme kapasitesi olması,

- Her öğrencinin farklı öğrenme tekniklerini kullanması,
- Öğrencinin sınırlı konu alanlarında modellenebilmesi,
- Akıllı Öğretim Sistemi'nin öğrenme sorununun olması,

şeklinde sıralanabilir. Öğrenci modeli olmadan, AÖS olarak tanımlanan bir öğretim sistemi öğretme işlevini gerçekleştirememektedir. Bu nedenle belirtilen olası sorunlar ve kullanıcı modelinin sahip olması gereken işlevsel yetenekler göz önüne alınarak uygun bir modelleme yöntemiyle öğrenci modeli oluşturulmalıdır.

Tasarlanan sistemde, öğrenci bilgisinin modellenmesi için kaplama öğrenci modeli kullanılmaktadır. Kaplama öğrenci modelinde Şekil 4.5'de görüldüğü gibi, öğrenci bilgisi uzman bilgisinin bir alt kümesi olarak görülmektedir. Bu modelde, konu içinde öğrencinin, uzmana göre ne kadar ilerleyebildiği gösterilmekte ve alanda uzmanın bilmesi gereken her ögenin düzeyi belirlenmektedir. Ayrıca, kaplama öğrenci modelinde öğrencinin bilgisi, uzman bilgisi ile karşılaştırılarak öğrenci bilgisinin uzman modeli içindeki düzeyi de belirlenmektedir. Böylece öğrencinin neleri bildiği ve neleri bilmesi gerektiğine karar verilir.

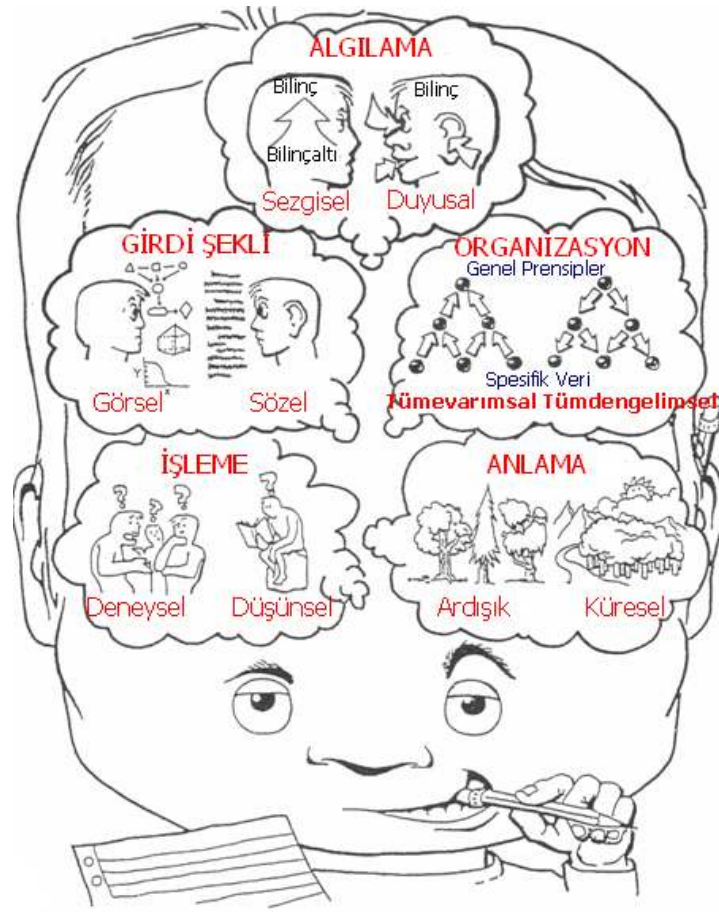


Şekil 4.5. Kaplama Öğrenci Modeli

Hazırlanan sistemde öğrenci modelleme birimi oluşturulurken öğrencilerin kişisel farklılıklarını merkeze alan bir anlayış benimsenmiştir. Bu anlayışla yola çıkıldığında öğrenenlerin bilgiyi algılama, işleme, düzenleme, problem çözme, ürün ortaya koyma, güdülenme şekillerindeki farklılıkların göz önüne alınması ve bunlara bağlı olarak öğrenme – öğretme süreçlerinin tasarlanması gerektiği bilgisine ulaşılmıştır. Öğrenme stilleri olarak bilinen bu bilgi “bireylerin bilgiyi alma, tutma ve işleme sürecindeki

karakteristik güçlölük ve tercihler” [54-63] olarak tanımlanmaktadır ve Şekil 4.6’da görüldüğü gibi birbirinden bağımsız beş boyut içermektedir. Bu boyutların her biri öğrenenin farklı alanlardaki tercih ve eğilimlerini ortaya koymaktadır. Bu alanlar;

- a) duyuşal – sezgisel
- b) görsel – sözel
- c) deneysel – düşünsel
- d) tümevarımsal – tümdengelimsel
- e) ardışık-küresel şeklindedir.



Şekil 4.6. Felder Silverman öğrenme stilleri modeli [64]

Tablo 4.2. Felder Silverman öğrenme stillerine ait baskın özellikler

Öğrenme Stili Modelleri		Baskın Özellikleri
Algılama	<i>Sezgisel</i>	• Belli bir sıra ve yöntem gerektiren işleri rahat yaparlar. • Somut bilgileri, olguları tercih ederler. • Somut ve deney gibi belirli bir işlem basamağına dayanan etkinlikler ilgilerini daha çok çeker. • Detayları çalışmakta ve bulmakta sabırlıdır. • Sembolleri (yazı, rakam, denklem vb) algılamak ve anlamakta zorlanabilirler.
	<i>Duyusal</i>	• Soyut bilgileri (kavram, kural, ilke, kuram, matematiksel ifade vb) tercih ederler. • Detayları çalışmak ve bulmak konularında sabırsızdır. • Daha hızlı çalışma eğilimindedirler, daha ilerlemecilerdir ama dikkatsiz olabilirler. • Öğrenme sürecinde, olasılıkları ve ilişkileri keşfetmeyi, hayal gücünü kullanmayı severler. • Sembolleri algılamak ve anlamakta daha rahattırlar ancak zamana dayalı testlerde başarısız olabilirler.
Girdi Şekli	<i>Görsel</i>	• En çok ne gördüklerini hatırlarlar. Görsel verilerden (resim, diyagram, gösteri, şema, harita vb.) sözel verilere göre (yazılı ve sözlü açıklamalar) daha çok bilgi edinirler. • Sözlü uyarıcıları görselleştirmeye yönelik öğrenme stratejilerini kullanmaya yatkındırlar.
	<i>Sözel</i>	• En çok yazılı ve sözlü uyarıcıları ve söylediklerini hatırlarlar. Tartışmaları, sözel açıklamaları görsel gösterimlere tercih ederler ve bir şeyi başkalarına açıklayarak en etkili biçimde öğrenirler. • Okuma ve tekrardan başka bilgiyi işleme stratejilerini pek bilmediklerinden dolayı çok öğrenme stratejilerinde çeşitlilik gösteremezler.
Organizasyon	<i>Tümevarımsal</i>	• Gözlemlerden başlar. • Sonuç çıkarır, açıklar. • Uzun süre zihinde tutmak ve aktarmak için daha uygundur.
	<i>Tümdengelimsel</i>	• Prensiplerden başlar • Mantık yürütür, türetir • Daha çok bilginin kısa süre zihinde tutulması için daha uygundur
İşleme	<i>DeneySEL</i>	• “Deneyelim ve nasıl çalıştığını görelim” cümlesi bu tip öğrenenleri tanımlar. • Grup çalışmalarını tercih ederler. • Bilgiyi en iyi tartışarak, uygulayarak, deneyerek ya da başkalarına açıklayarak alır ve anlarlar. Etkileşimi tercih ederler.
	<i>Düşünsel</i>	• “İlk önce onun hakkında düşünelim” cümlesi bu tip öğrenenleri tanımlar. • Yalnız başına ya da en fazla bir kişiyle yapılan çalışmaları tercih ederler. • Bilgiyi, öncelikli olarak hakkında düşünerek alır ve anlarlar. Bilginin ne anlama geldiği, kendisine ne çağırırdı, olası uygulamalarının ne olduğu hakkında düşünmek isterler.

Tablo 4.2'nin devamı

<i>Anlama</i>	<i>Ardışık</i>	• Her adımın mantıksal olarak bir başka adımı takip ettiği doğrusal adımlar şeklinde anlama eğilimindedirler. • Bilgiyi kısmen anlasalar bile onunla bir şeyler yapabilirler. • Problem çözerken doğrusal bir sorgulama sürecini takip ederler. (İstenen ne? Verilen ne? Hangi işlem basamaklarıyla bu problemi çözebilirim?)
	<i>Küresel</i>	• Büyük sıçramalar şeklinde öğrenme eğilimindedirler. Bu süreç içerisinde bilgiyi, ilişkileri görmeden küme, yığın biçiminde bütünsel olarak alırlar ve ansızın ilişkileri keşfederler. • Problem çözerken aniden çözümü bulabilirler ama bu çözüme nasıl ulaştıkları konusunda açıklama yapamayabilirler. • Konunun detaylarını anlamadan önce, ele alınana konunun daha önce öğrenilen konularla nasıl bir ilişki içinde olduğunu öğrenmek ya da bulmak isterler.

Sistemde kullanıcının öğrenme stilini belirlemek amacıyla bir ön test yapılmaktadır. Ön stil testi sonucuna göre, öğrenciye konu sunumu yapılmaktadır. Ancak, öğrencinin ön teste doğru cevaplar veremeyebileceği düşünülerek ve sistemdeki öğrenci davranışları referans alınarak sürekli öğrenci modeli güncellenmektedir.

Öğrenci modelinin oluşturulması için her öğrencinin bilgileri veritabanındaki tablolara kayıt edilmektedir. Bu amaçla, öğrenci veritabanında bilgi durumu, kisisel bilgileri, sistem parametre, konu derecesi, zaman tabloları gibi bilgiler bulunmaktadır. Bu tablolara ait alanlar aşağıdaki tablolardan incelenebilir. Öğrencinin sistemdeki her işleminde bu tablolar yenilenmektedir.

Tablo 4.3. bilgi_durumu tablosuna ait alanlar

<u>Alan Adı</u>	<u>Açıklama</u>
numara	Öğrenci numarası
ogrenmestili	Seviye belirleme testine göre belirlenmiş öğrenme stili
guncel_ogrenme_stili1	Adapte edilmiş öğrenme stili1
guncel_ogrenme_stili2	Adapte edilmiş öğrenme stili2
guncel_ogrenme_stili3	Adapte edilmiş öğrenme stili3
guncel_ogrenme_stili4	Adapte edilmiş öğrenme stili4

Tablo 4.4. kisiselbilgi tablosuna ait alanlar

<u>Alan Adı</u>	<u>Açıklama</u>
numara	Öğrenci numarası
adi	Öğrenci Adı
soyadi	Öğrenci Soyadı
email	Öğrenci mail adresi
sifre	Öğrenci şifresi

Tablo 4.5. konuderecesi tablosuna ait alanlar

<u>Alan Adı</u>	<u>Açıklama</u>
numara	Öğrenci numarası
konuadi	Konunun ayrıntılı ismi
derecesi	Konuyu öğrenme düzeyi

Tablo 4.6. sistemparmetre tablosuna ait alanlar

<u>Alan Adı</u>	<u>Açıklama</u>
numara	Öğrenci numarası
giris	Öğrencinin sisteme giriş tarihi
cikis	Öğrenci sistemden çıkış tarihi
sayac	Sayfa sayacı
forum	Öğrenciye ait Forum sayacı
ses	Öğrenciye ait ses sayacı
applet	Öğrenciye ait applet sayacı
oyun	Öğrenciye ait oyun sayacı
konutakibi	Öğrenci konu planını takip etme bilgisi
bol1cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 1 de çözdüğü doğru soru sayısı
bol2cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 2 de çözdüğü doğru soru sayısı
bol3cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 3 de çözdüğü doğru soru sayısı
bol4cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 4 de çözdüğü doğru soru sayısı
bol5cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 5 de çözdüğü doğru soru sayısı
bol6cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 6 da çözdüğü doğru soru sayısı
bol7cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 7 de çözdüğü doğru soru sayısı
bol8cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 8 de çözdüğü doğru soru sayısı

Tablo 4.6'nin devamı

bol9cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 9 da çözdüğü doğru soru sayısı
bol10cozdugu_dogru_soru_sayisi	Öğrencinin bölüm 10 da çözdüğü doğru soru sayısı
bol1ziyaret_edilen_sayfa_sayisi	Öğrencinin 1.bölümde ziyaret ettiği sayfa sayısı
bol2ziyaret_edilen_sayfa_sayisi	Öğrencinin 2.bölümde ziyaret ettiği sayfa sayısı
bol3ziyaret_edilen_sayfa_sayisi	Öğrencinin 3.bölümde ziyaret ettiği sayfa sayısı
bol4ziyaret_edilen_sayfa_sayisi	Öğrencinin 4.bölümde ziyaret ettiği sayfa sayısı
bol5ziyaret_edilen_sayfa_sayisi	Öğrencinin 5.bölümde ziyaret ettiği sayfa sayısı
bol6ziyaret_edilen_sayfa_sayisi	Öğrencinin 6.bölümde ziyaret ettiği sayfa sayısı
bol7ziyaret_edilen_sayfa_sayisi	Öğrencinin 7.bölümde ziyaret ettiği sayfa sayısı
bol8ziyaret_edilen_sayfa_sayisi	Öğrencinin 8.bölümde ziyaret ettiği sayfa sayısı
bol9ziyaret_edilen_sayfa_sayisi	Öğrencinin 9.bölümde ziyaret ettiği sayfa sayısı
bol10ziyaret_edilen_sayfa_sayisi	Öğrencinin 10.bölümde ziyaret ettiği sayfa sayısı
bol1cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 1 de çözdüğü yanlış soru sayısı
bol2cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 2 de çözdüğü yanlış soru sayısı
bol3cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 3 de çözdüğü yanlış soru sayısı
bol4cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 4 de çözdüğü yanlış soru sayısı
bol5cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 5 de çözdüğü yanlış soru sayısı
bol6cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 6 da çözdüğü yanlış soru sayısı
bol7cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 7 de çözdüğü yanlış soru sayısı
bol8cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 8 de çözdüğü yanlış soru sayısı
bol9cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 9 da çözdüğü yanlış soru sayısı
bol10cozdugu_yanlis_soru_sayisi	Öğrencinin bölüm 10 da çözdüğü yanlış soru sayısı

Tablo 4.7. zaman tablosuna ait alanlar

<u>Alan Adı</u>	<u>Açıklama</u>
oturum	Oturum numarası
numara	Öğrenci numarası
sisgiris	Öğrencinin sisteme giriş zamanı
siscikis	Öğrenci sistemden çıkış zamanı
zamanfarki	Öğrencinin sistemde kalma süresi

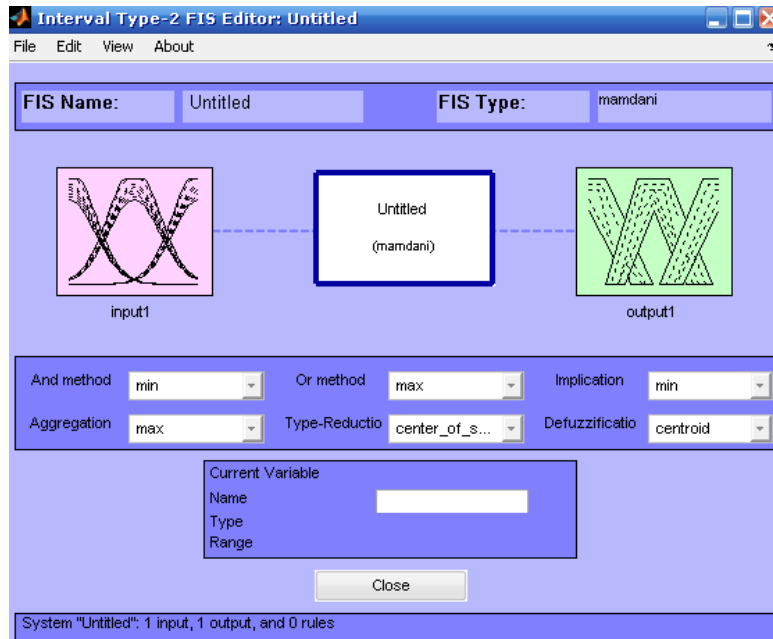
4.3 Geliştirilen Sistemde Kullanılan Yazılımlar ve Teknolojiler

Geliştirilen sistemde dinamik web programlama dillerinden PHP ve çok kanallı, çok kullanıcı, hızlı ve sağlam bir veri tabanı yönetim sistemi olan MYSQL kullanılmıştır. Sistem internet üzerinden erişimi desteklediği için herhangi bir tarayıcı kullanılarak çalıştırılabilmektedir.

Sistemin web sunucu yazılımı için internetteki en yaygın yazılım olan Apache web sunucusu kurulmuştur. MATLAB uygulamalarının oluşturulması ve hazırlanan uygulamaların internet üzerinden erişiminin sağlanabilmesi için MATLAB'ın 13. sürümü ile birlikte tanıtılan MATLAB Web Sunucu araç kutusu seçilmiştir.

4.3.1 Fuzzy2 Toolbox

Fuzzy2 toolbox, şekil 4.7'de görüldüğü gibi tip-2 bulanık mantık tabanlı sistemlerin tasarımı için m-dosyaları ile oluşturulmuş fonksiyon topluluğudur. Bu alet-kutusu ile singleton tip2 mamdani, tip-1 non-singleton tip-2 mamdani, tip-2 non-singleton tip-2 mamdani ve tip-2 sugeno bulanık mantık sistemler tasarlanabilir, grafiksel ortamda hazırlanan bu sistemler SIMULINK ortamında kullanılabilir [53].



Şekil 4.7. Fuzzy2 toolbox ana penceresi

4.3.1.1 Tip-2 Bulanık Mantık

Bilindiği gibi istatistikte ve olasılık kuramında, belirsizliklerle değil kesinliklerle çalışılır ama insanın yaşadığı ortam daha çok belirsizliklerle doludur. Bu nedenle insanoğlunun sonuç çıkarabilme yeteneğini anlayabilmek için belirsizliklerle çalışmak gereklidir.

Bulanık mantık sistemlerde, kelimelerin farklı insanlar için farklı anlamlarının olması, bilginin aynı görüşte olmayan uzman grubundan alınması, bulanık mantık sistemleri harekete geçiren ölçümlerin gürültülü olması gibi nedenlerden dolayı belirsizlikler vardır [65-67].

Bulanık mantığın yeni bir açılımı olan bir tip-2 bulanık mantık üyelik fonksiyonunun bulanık olmasından ve üçüncü boyutundan dolayı belirsizlikleri modelleyip etkisini azaltabilir. Belirsizlikler kaybolduğunda da tip-2 bulanık mantık tip-1 bulanık mantığa indirgenebilir. Ancak üçüncü boyuttan dolayı çizimin zorlaşması, hesaplamaların daha karmaşık olması gibi nedenlerden tip-2 bulanık kümeleri anlamak tip-1 bulanık kümeleri anlamaktan daha zordur [68-87].

Tip-2 Üyelik Fonksiyonları: Bulanık mantık sistemlerde üyelik fonksiyonları, giriş ve çıkışlarla ilişkili olarak kuralların varsayım (antecedent) ve sonuç (consequent) kısımlarında yer alır. Bulanık kümeleri tanımlayan üyelik fonksiyonları tip-1 ve tip-2 olmak üzere ikiye ayrılabilir. Belirsizliği daha iyi ifade edebilme gücü olan tip-2 bulanık mantık için tanımlanan üyelik fonksiyonları, klasik bulanık mantık olarak bilinen tip-1 üyelik fonksiyonlarından yola çıkılarak incelenecektir.

Tek değişkenli A tip-1 bulanık kümesi

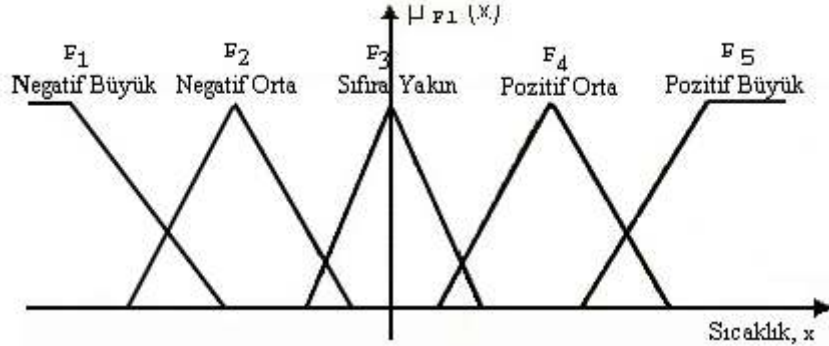
$$A = \{(x, \mu_A(x)) \mid \forall x \in X\} \quad (4-1)$$

olarak gösterilir.

Tüm $x \in X$ 'ler için tip-1 üyelik fonksiyonu $\mu_A(x)$, 0 ile 1 arasında olmalıdır. Tip-1 üyelik fonksiyonu iki boyutludur.

Üyelik fonksiyonundaki dilsel etiketlerin seçimi serbesttir. Örneğin Şekil 4.8'de sıcaklık için beş tane dilsel etiket tanımlanmıştır (Negatif Büyük, Negatif Orta, Sıfıra Yakın, Pozitif Orta, Pozitif Büyük). Ayrıca üçgen şeklinde üyelik fonksiyonları kullanılmıştır. Üyelik fonksiyonlarını çizerken üçgenlerin merkezlerinin nerede olacağı, taban genişliklerinin ne kadar olacağı, ne kadar üst üste binecekleri gibi kararlar önemlidir.

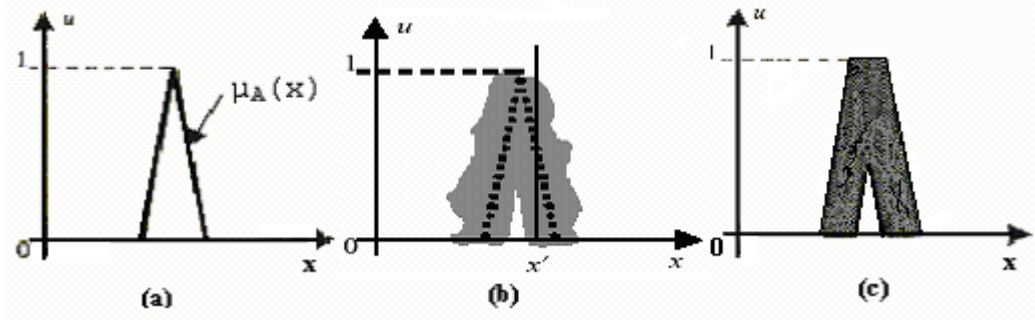
Tip-2 bulanık küme ve üyelik fonksiyonlarının belirsizliği modelleme özelliğinden dolayı bu kararların çoğuna ihtiyaç duyulmamaktadır.



Şekil 4.8. Tip-1 üyelik fonksiyonları

Elemanların üyeliği 0 veya 1 olarak belirlenemediğinde tip-1 bulanık kümeler kullanılır. Benzer şekilde koşullar çok bulanıksa yani üyelik derecesini $[0, 1]$ arasında belirlemede problem yaşıyorsa, tip-2 bulanık kümeler kullanılır. (Bu fikir ilk kez 1975 yılında Zadeh tarafından ileri sürülmüştür) [65]. Ancak bu görüş tip-2 kümeleri kullanmak için olağanüstü bulanık koşullara sahip olunması gerektiği anlamında değildir. Eğer belirsizliğin tam değeri belirlenemiyorsa (üyelik derecesi net olarak belirlenemiyorsa), tip-1 yerine tip-2 bulanık kümeleri kullanmak daha doğru olacaktır. Bu şekilde düşünüldüğünde, belirsizliği sonlu tipte bir bulanık kümenin temsil edemeyeceği söylenebilir. Bu durumda, belirsizliği tamamen göstermek için tip- ∞ bulanık kümesini kullanma ihtiyacı duyulur. Ancak pratikte bu mümkün değildir. Bu yüzden bazı sonlu tip bulanık kümeler kullanılır.

Şekil 4.9 (a)'daki tip-1 üyelik fonksiyonu üçgenin solundan ve sağından çeşitli miktarlarda noktaları kaydırılarak bulanıklaştırılmış olsun. Şekil 4.9 (b)'ye bakılırsa x 'de dikey çizginin bulanık kısımla kesiştiği noktalarda üyelik fonksiyonu birden fazla değer almıştır. Bu değerler aynı ağırlığa sahip olmak zorunda değildir ve bu noktalara bir genlik dağılımı uygulanabilir. Bu bütün $x \in X$ için yapılırsa tip-2 bulanık kümeleri tanımlayan üç boyutlu üyelik fonksiyonu yani tip-2 üyelik fonksiyonu elde edilir.



Şekil 4.9. (a) Tip-1 üyelik fonksiyonu (b) Bulanıklaştırılmış tip-1 üyelik fonksiyonu (c) FOU(Belirsizliğin Ayak İzi)

Bir tip-2 bulanık kümesi $\tilde{A}$ ile gösterilir ve $x \in X$, $u \in J_x \subseteq [0,1]$ olmak üzere tip-2 üyelik fonksiyonu $\mu_{\tilde{A}}(x,u)$ şeklinde tanımlanır, yani

$$\tilde{A} = \{((x,u), \mu_{\tilde{A}}(x,u)) \mid \forall x \in X, \forall u \in J_x \subseteq [0,1] \} \quad 0 \leq \mu_{\tilde{A}}(x,u) \leq 1 \quad (4-2)$$

olarak ifade edilir.

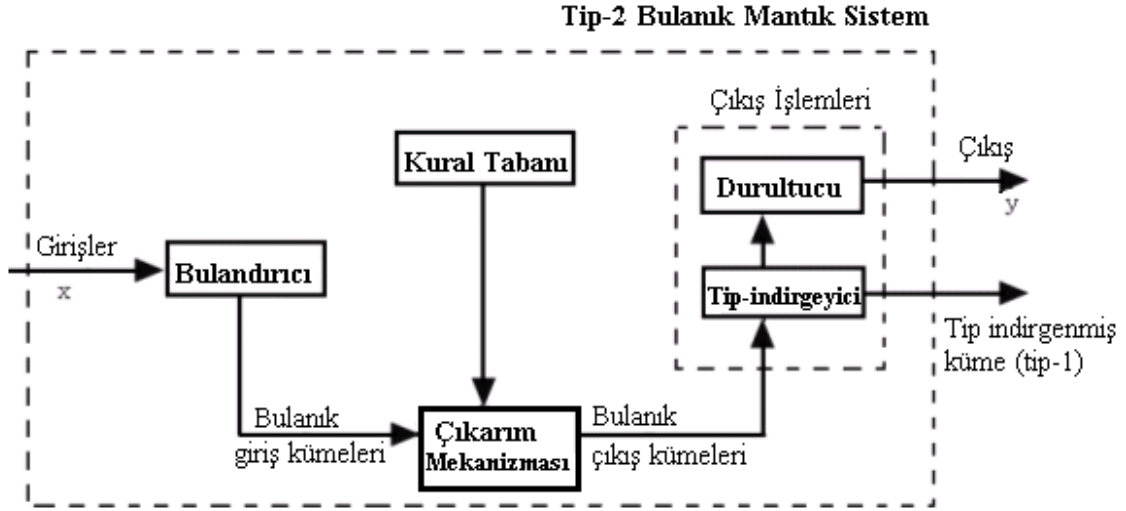
$\tilde{A}$ ayrıca

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x,u) / (x,u) \quad J_x \subseteq [0,1] \quad (4-3)$$

şeklinde de gösterilebilir.

Denklem (4-3)'teki $\int \int$ x ve u'nun birleşimini gösterir.

Tip-2 Bulanık Mantık Sistemler: Tip-2 bulanık mantık sistemin yapısı Şekil 4.10'da gösterilmiştir. Şekilden de görüldüğü gibi tip-2 bulanık mantık sistem, tip-1 bulanık mantık sisteme çok benzemektedir. Tip-2 bulanık mantık sistemin yapısal farklılığı çıkış bloğundadır. Tip-1 bulanık mantık sistemin çıkış bloğunda sadece durultucu bulunurken, tip-2 bulanık mantık sistemin çıkış bloğunda tip-indirgeyici ve durultucu birlikte yer almaktadır [85-87].



Şekil 4.10. Tip-2 bulanık mantık sistem yapısı

Bulanık mantığın ilkeleri, tip-1’den tip-2’ye geçerken değişme göstermemiş olup tip- n için de değişmesi beklenmemektedir [65]. Yüksek tip sayısı bulanıklığın yükseklik derecesini göstermektedir. Yüksek tip, üyelik fonksiyonunun yapısını değiştirdiği için üyelik fonksiyonlarına bağlı işlemler değişecektir. Bulanık mantığın basit ilkeleri, üyelik fonksiyonu yapısından bağımsız olduğu için ilkelerde değişiklik olmayacaktır.

Bu tezde anlatılan tip-2 bulanık mantık sistemlerde, kurallardaki tüm varsayım (antecedent) ve sonuç (consequent) kümeleri tip-2 olarak alınmıştır. Ancak pratikte böyle bir zorunluluk yoktur. Bir bulanık mantık sistemin tip-2 olması için kuraldaki varsayım ve sonuç kümelerinden birinin tip-2 olması yeterlidir.

Tip-1’de kural yapısı:

$$R^l : IF \ x_1 \text{ is } F_1^l \text{ and } x_2 \text{ is } F_2^l \text{ and...and } x_p \text{ is } F_p^l \text{ THEN } y \text{ is } G^l \quad l=1,..,M \quad (4-4)$$

şeklindedir.

Tip-2’de kural yapısı tip-1 ile aynı kalmaktadır. Sadece kuraldaki bir veya daha fazla küme tip-2 olmaktadır. P tane girişi olan $x_1 \in X_1, ..., x_p \in X_p$, tek çıkışlı, M tane kurala sahip tip-2 bulanık mantık sistemde l . kural

$$R^l : IF \ x_1 \text{ is } \tilde{F}_1^l \text{ and } x_2 \text{ is } \tilde{F}_2^l \text{ and...and } x_p \text{ is } \tilde{F}_p^l \text{ THEN } y \text{ is } \tilde{G}^l \quad l = 1, ..., M \quad (4-5)$$

şeklinde ifade edilir.

Bu kural $X_1 \times ... \times X_p$ giriş uzayı ve Y çıkış uzayı arasındaki ilişkiyi göstermektedir.

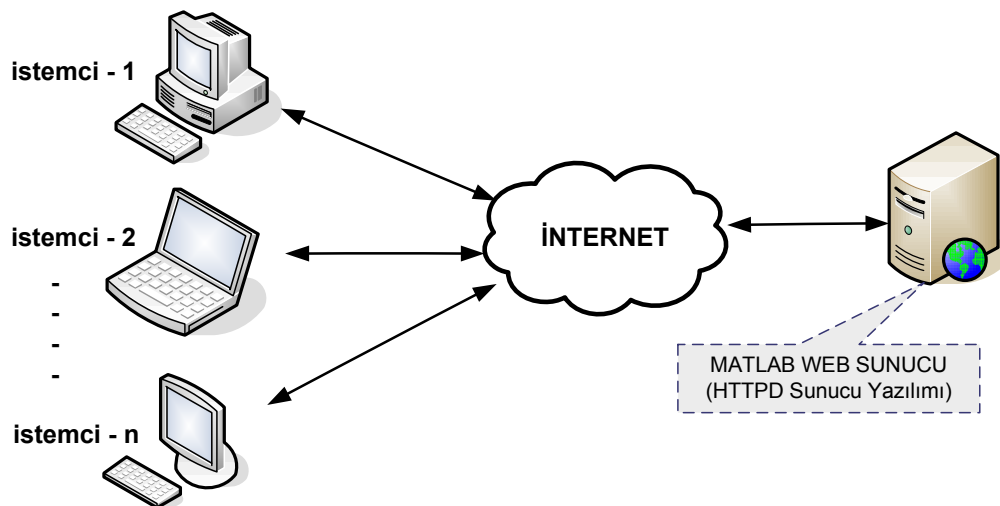
Tip-2 bulanık mantık sistemde çıkarım mekanizmasının işleyişi, tip-1 bulanık sistemle paralellik göstermektedir. Tip-1 bulanık mantık sistemde çıkarım mekanizması,

bulandırıcının çıkışlarını (üyelik derecelerini) ve kural tabanını kullanarak bir bulanık küme oluşturur. Bu bulanık küme, durultucu tarafından denetleyicinin çıkışını hesaplamak için kullanılacaktır. Kuraldaki varsayımlar t-norm ile bağlıdır. Giriş kümelerindeki üyelik dereceleri çıkış kümesindeki üyelik dereceleri ile sup-star yöntemi kullanılarak birleştirilir. Benzer şekilde tip-2 bulanık mantık sistemde de üyelik dereceleri, genişletilmiş sup-star yöntemi kullanılarak birleştirilmektedir.

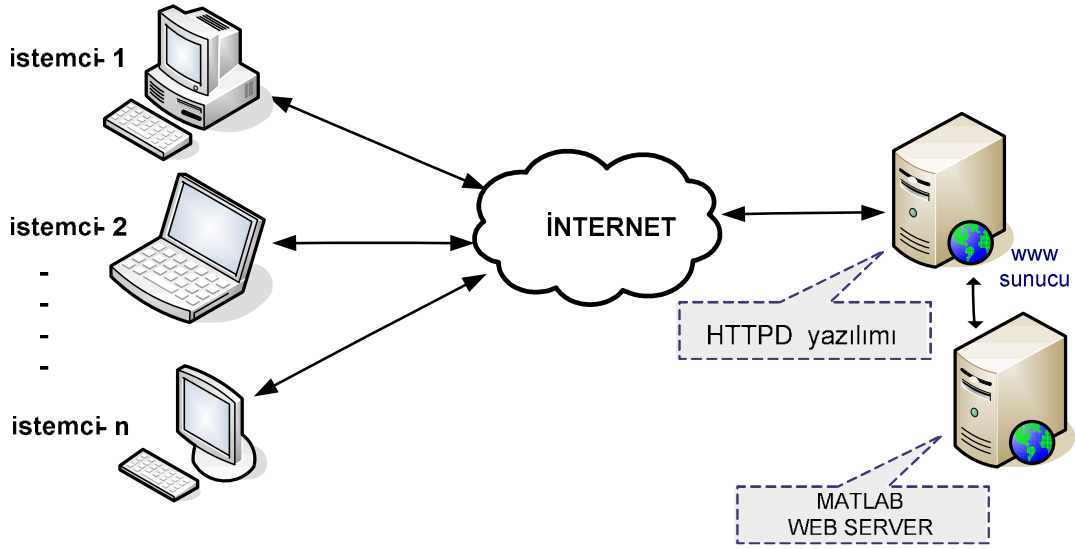
1.3.2 MATLAB Web Sunucu

MATLAB'ın 13. sürümü ile birlikte yeni bir araç kutusu tanıtılmıştır. MATLAB Web Sunucu olarak isimlendirilen bu araç kutusu sayesinde, geliştirilen MATLAB tabanlı benzetimlere ve uygulamalara internet üzerinden erişebilmek, bunları çalıştırabilmek ve sonuçları yine web tabanlı olarak görüntüleyebilmek mümkün olmaktadır [88].

MATLAB Web Sunucusu, MATLAB programında işlenmek üzere internet üzerinden veri alabilen ve bu verileri işledikten sonra elde edilen sonuçları yine bir internet tarayıcısında gösterebilen MATLAB uygulamaları geliştirmeyi sağlamaktadır. İstemci sistem ile MATLAB arasındaki veri alışverişi TCP/IP protokolüne uygun olarak gerçekleştirilmektedir. Şekil 4.11'de görüldüğü gibi, en basit bir yapılanmada; bir istemci bilgisayar ve bunun üzerinde çalışan bir internet tarayıcı, üzerinde MATLAB, MWS ve bir web sunucu yazılımının yüklü olduğu bir sunucu bilgisayar olmalıdır. İş yükünün yoğun olduğu durumlarda ise, şekil 4.12'de verilen yapı kullanılabilir.



Şekil 4.11. Basit bir MWS uygulaması için ağ yapısı

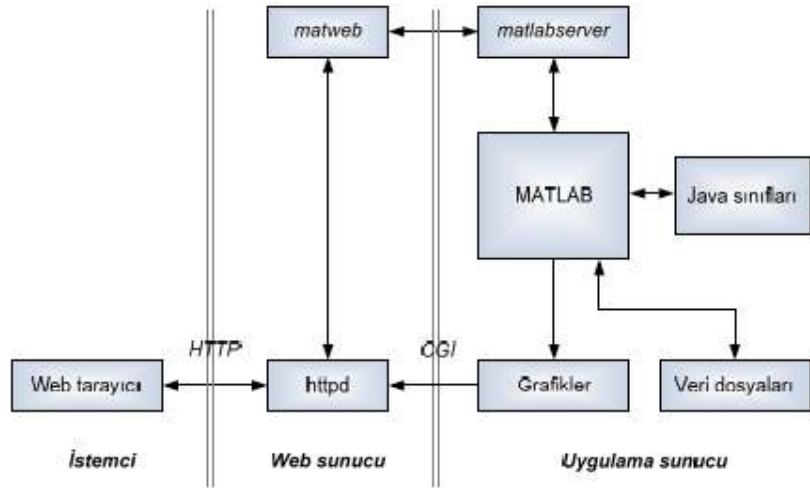


Şekil 4.12. İş yükünün yoğun olduğu MWS uygulamaları için ağ yapısı

4.3.2.1 Uygulama Gereksinimleri

MWS, MATLAB programı ile uygulama geliştirme için duyulan yazılım ve donanım gereksinimleri dışında fazladan bir donanıma ya da yazılıma ihtiyaç duymamaktadır. Uygulama geliştirilecek olan bilgisayara TCP/IP ağ yazılımı yüklenmiş durumda olmalıdır. MWS tarafından girişleri almak ve çıkışları sunmak işlemlerinde kullanılacak olan tarayıcı programda bilgisayarda kurulmuş olmalıdır.

MATLAB programının çalıştığı bilgisayara ya da bu bilgisayara ağ üzerinden bağlı başka bir bilgisayara web sunucu yazılımı yüklenilmelidir. Yüklenen web sunucu CGI programlarını çalıştırabilmelidir. Bu yazılımı yüklemek için değişik kaynaklar kullanılabilir. MWS, Apache, Microsoft Internet Information Server veya Netscape Enterprise Server gibi farklı birçok sunucu yazılımı altında çalışabilmektedir. MWS uygulamalarının çalışma düzenini gösteren genel bir blok model yapısı Şekil 4.13’de gösterilmiştir [88].



Şekil 4.13. MWS modeli

4.3.2.2 MWS Bileşenleri

MATLAB Web Sunucusu, MATLAB uygulamalarının oluşturulması ve yapılan uygulamaların internet üzerinden erişiminin sağlanabilmesi için gerekli olan bir sunucu sistemidir. Bu sunucu sistemi kendi içerisinde birçok bileşenden oluşmaktadır.

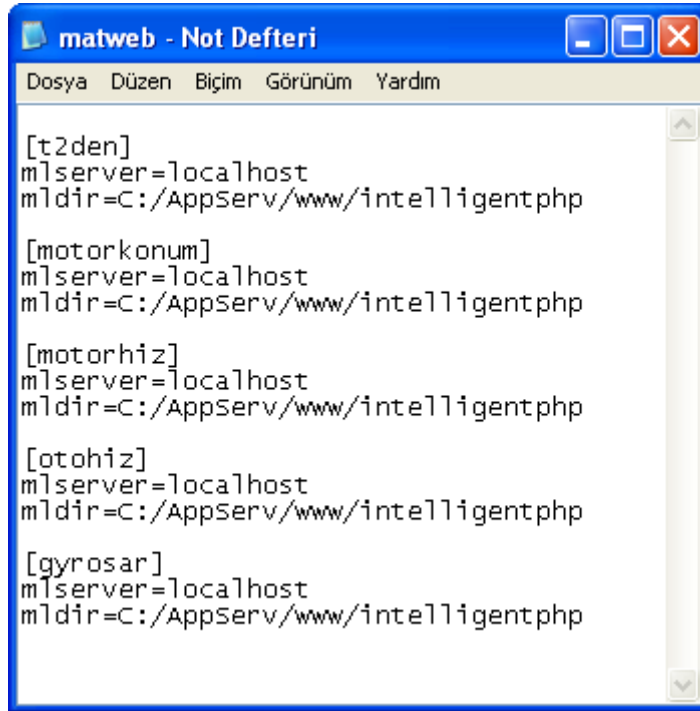
- *matlabserver*: Web uygulaması ile MATLAB arasındaki iletişimi yönetir. Çok yönlü bir TCP/IP sunucusudur. HTML dosyası içerisindeki *mlmfile* şeklinde isimlendirilmiş gizli bir alanda belirtilen MATLAB m-dosyasını çalıştırır. *matlabwserver.conf* dosyasında gerekli değişiklikler yapılarak MWS'nin hangi TCP/IP portunu dinleyeceği ve aynı anda kaç adet uygulamanın eş zamanlı çalıştırılabileceği ayarlanabilir.
- *matweb*: *matlabserver*'in bir TCP/IP istemcisidir. HTML dosyasından verileri elde etmek ve bunları *matlabserver*'e iletmek amacıyla CGI sistemini kullanmaktadır.
- *matweb.m*: Web uygulaması tarafından belirtilen M-dosyasını çalıştırmaktadır.
- *matweb.conf*: *matweb* ile *matlabserver* arasındaki iletişimi yapılandırmayı sağlayan dosyadır. Uygulamalar bu dosya içerisinde listelenmelidir.
- *hosts.conf*: Ek güvenlik sağlamak amacıyla isteğe bağlı olarak kullanılan bir konfigürasyon dosyasıdır. Eğer bu dosya kullanılırsa, MWS'ye sadece bu dosyada tanımlanan bilgisayar yada bilgisayarlar bağlanabilir.

MWS uygulamaları, M-dosyalarının, HTML dosyalarının ve grafiklerinin bir bileşimidir. MWS uygulamaları gerçekleştirmek için MATLAB programcılık bilgisi ve

temel düzeyde HTML bilgisine sahip olmak yeterlidir. Uygulama geliştirme aşağıdaki basit aşamalardan oluşur [88]:

- a. Kullanıcılardan internet vasıtasıyla alınacak bilgileri ve elde edilen sonuçları tutacak HTML dosyaları hazırlanır. Bunun için basit bir metin editörü kullanılabileceği gibi, birçok web sayfası tasarım programı (Microsoft FrontPage, Macromedia Dreamweaver vb.) kullanılabilir.
- b. Uygulama ismi ve gerekli yapılandırma ayarları `matweb.conf` dosyası üzerinde hazırlanır.
- c. Aşağıdaki işlemleri yürüten bir MATLAB *m-dosyası* yazılır.
 - HTML giriş dosyasından verilerin alınması
 - Veriler üzerinde istenen işlemlerin yapılarak, sonuç ve grafiklerin üretilmesi
 - Çıkış verilerinin MATLAB dosyası biçiminde hazırlanması
 - Çıkış verilerinin, HTML çıkış dosyasına aktarılması için *htmlrep* fonksiyonunun işletilmesi

Tasarlanan sistemde de bu işlemleri yerine getiren Şekil 4.14'deki `matweb.conf` ve Şekil 4.15'deki `t2den.m` dosyaları oluşturulmuştur.



Şekil 4.14. `matweb.conf` ayar dosyası

```
1 function retstr = t2den(outfile)
2 retstr = char(' ');
3 mden1=urlread('file:///C:\AppServ\www\intelligentphp\dosya.txt');
4 mden=str2num(mden1);
5 msize = [1 2];
6 outstruct.msize = msize;
7 outstruct.mden = mden;
8 outstruct.msquare = ruleview2guisiz(mden);
9 clear fide;
10 y = outstruct.msquare;
11 fide = fopen('\AppServ\www\intelligentphp\result.txt','w');
12 fprintf(fide,'%0.5g ',y);
13 fclose(fide);
14 d = sum(outstruct.msquare,1);
15 outstruct.msum = d(1,1);
16 templatefile = which('t2den2.php');
17 if (nargin == 1)
18     retstr = htmlrep(outstruct, templatefile);
19 elseif (nargin == 2)
20     retstr = htmlrep(outstruct, templatefile, outfile);
21 end
```

Şekil 4.15. t2den.m dosyası

4.3.3 PHP, MYSQL

Basitlik, veri tabanlarıyla mükemmel iletişim ve platformdan bağımsızlık gibi kullanım nedenlerinden dolayı geliştirilen sistemde web’de PHP ile çalışılmıştır. Veri tabanı ise Apache ve PHP ile beraber web-veritabanı uygulamalarında çok yaygın kullanılan, çok kullanıcıli hızlı ve sağlam bir veri tabanı yönetim sistemi olan MYSQL ile oluşturulmuştur.

4.4 Geliştirilen Yazılımın Modülleri

Geliştirilen yazılım sisteme giriş, konular, bireysel raporlar, öğrenme stili belirleme testi, sanal deneyler, forum ve zeki oyunlar olmak üzere yedi modülden oluşmaktadır.

4.4.1 Sisteme Giriş Modülü

Öğrenci sisteme kullanıcı adı ve şifre ile girmektedir. Sisteme giriş esnasında öğrenciye ait ad, soyad, numara gibi sistemin sonradan kullanacağı birçok bilgi elde edilir. Elde edilen bu bilgiler veritabanı tablolarında tutulur ve yapılacak sorgulamalarda bu tablolar kullanılır. Sisteme giriş ekranı Şekil 4.16’da görülmektedir.

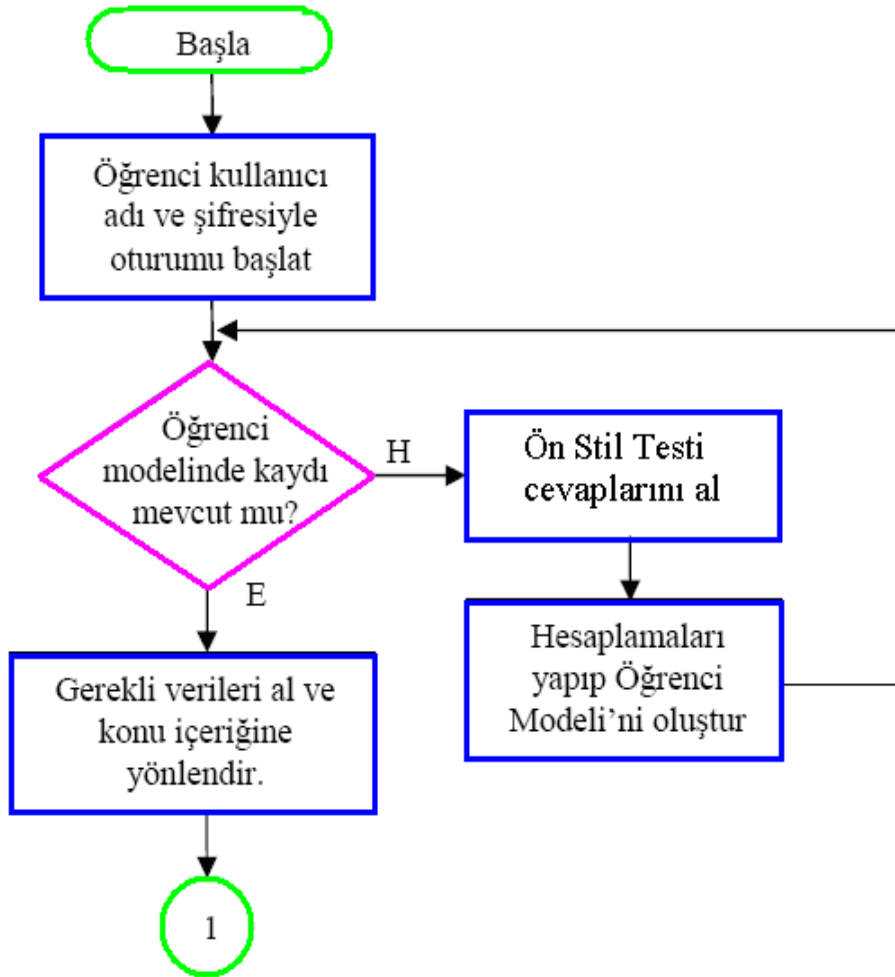
Şekil 4.16. Sisteme giriş ekranı

4.4.2 Sistemin İşleyişi

Öğrenci sisteme giriş yaptıktan sonra Şekil 4.17’de görülen ana menüde yer alan linklerle Akıllı Öğretim Sistemi uygulaması başlamaktadır. Sistem Şekil 4.18’deki akış şemasından görüldüğü gibi önce öğrencinin öğrenci modelinde kaydı olup olmadığını kontrol etmektedir. Eğer öğrenci modelinde kaydı varsa bu öğrenci daha önceden giriş yapmış ve en azından ön stil testini çözmüş demektir. Öğrencinin mevcut bilgileri alınarak konu içeriğinin sunulduğu sayfalara yönlendirilir. Eğer bilgileri yoksa ki bu öğrencinin sisteme ilk kez girdiğini gösterir, ön stil testi sayfasına yönlendirilir.

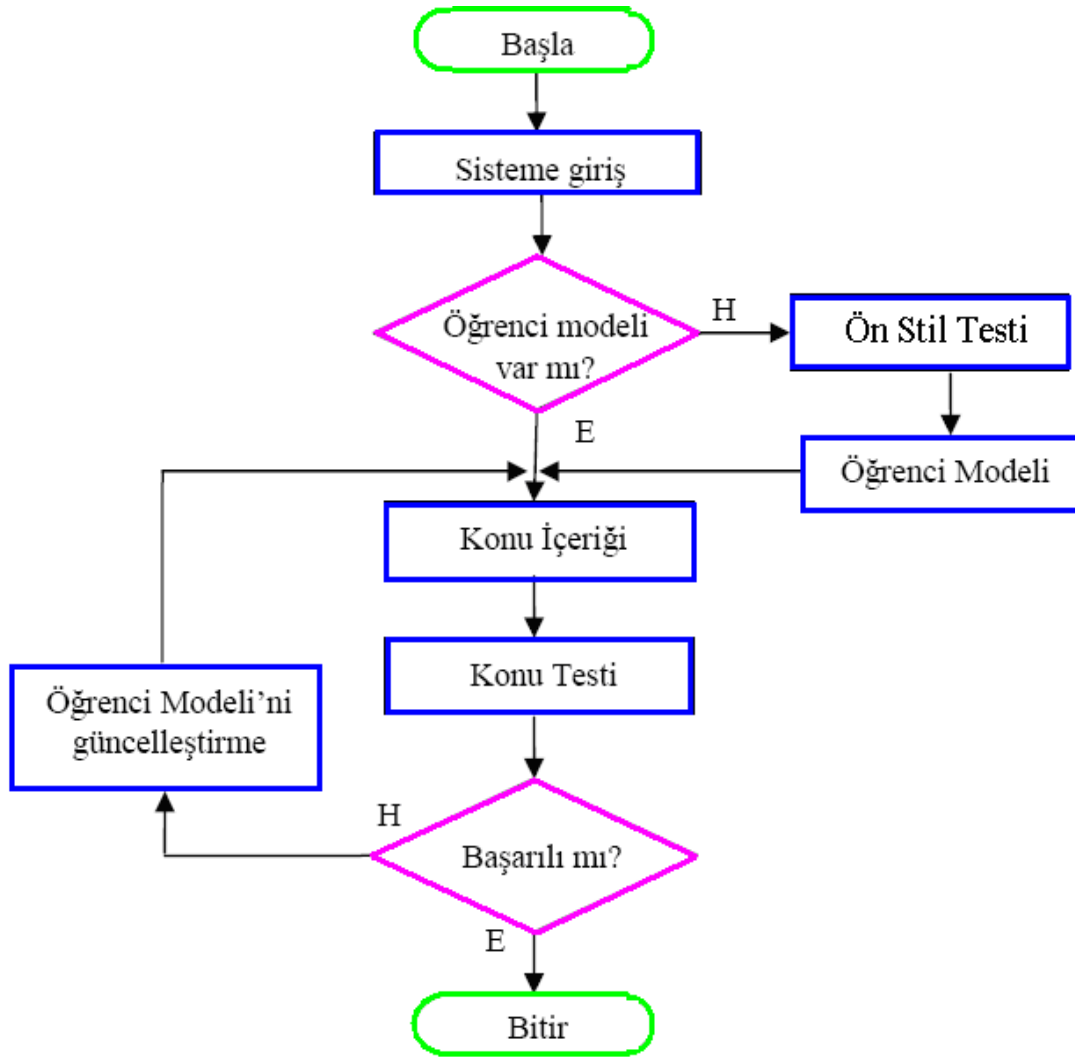


Şekil 4.17. Geliştirilen sistemin ana sayfası



Şekil 4.18. Sisteme giriş ve ön stil testi bölümlerinin akış şeması

Sisteme daha önceden girilmiş olan ön stil testi soruları tüm öğrencilere aynen sorulur. Öğrencilerin başlangıç stillerini tespitite ön stil testinin yeri büyük olduğundan tüm öğrencilere aynı soruları sorarak stillerini belirlemek yerinde olacaktır. Ön stil testini çözdükten sonra öğrencinin stili belirlenir. Elde edilen sonuç öğrenci modeline kaydedilir. Böylelikle öğrenci modelinin ilk bölümü oluşturulmuş olunur. Hesaplamalar yapıp ilgili alanlara kaydı yapıldıktan sonra tekrar girişe yönlendirilerek öğrenci modelindeki bilgilerin alınması ve öğrencinin konu içeriği sayfasına yönlendirilmesi sağlanır. Öğrenci sisteme ilk girdiğinde ön stil testi karşısına çıkarken, sonraki girişlerinde konu içeriği sayfaları ekrana gelmektedir. Yapılan tüm işlemler arka planda öğrencinin haberi olmadan gerçekleşmektedir.



Şekil 4.19. Sistem işleyişinin genel akış şeması

Şekil 4.19 akıllı öğretim sistem işleyişinin genel akış şemasında görüldüğü gibi sisteme giriş yapan öğrencinin ön stil testi ile öğrenci modeli oluşturulmaktadır. Konu içeriğini aldıktan sonra konu testini cevaplandıran öğrencinin öğrenci modeli başarısız olduğu konulara göre güncellenir ve başarısız olduğu konuları incelemek ve sorularını cevaplamak için konu içeriği sayfalarına yönlendirilir. Öğrenci tüm konulardan başarılı olana kadar, sistem konu içeriğini [89] sunmakla ve sorularını [90,91] sormakla görevlidir.



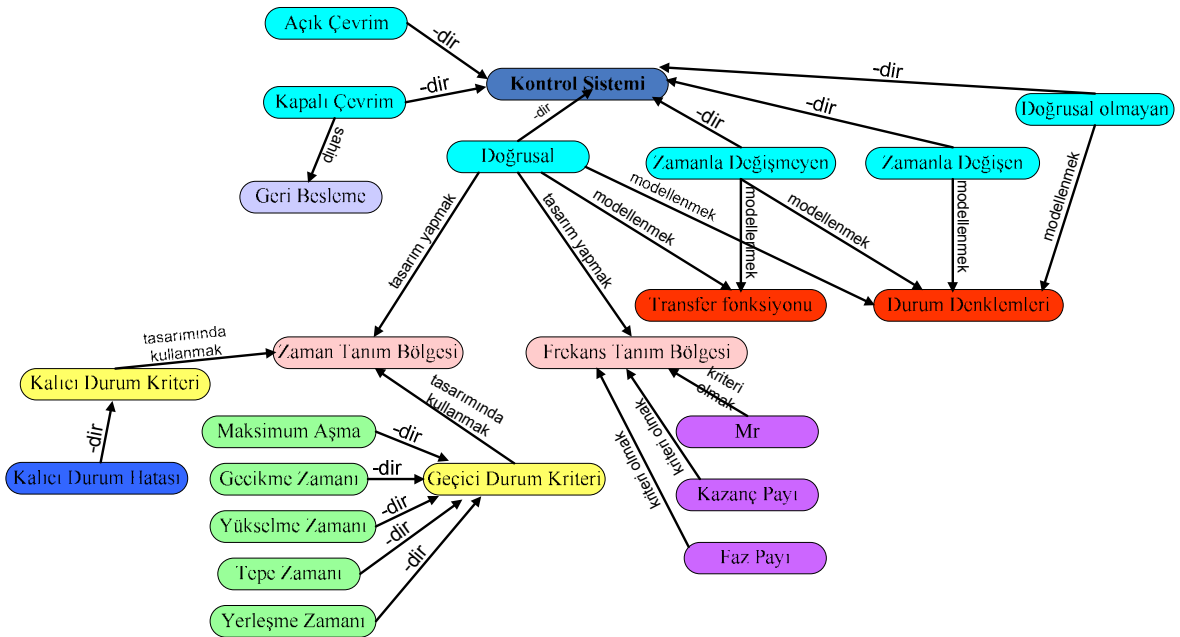
Şekil 4.20. Konular sayfası

Öğrencinin ünite testleri sonunda verdiği cevaplar Şekil 4.20'deki gibi analiz edilerek her konu için başarıları değerlendirilir. Eğer öğrenci çok düşük bir sonuç elde ettiyse ■, orta düzeyde bir sonuç elde ettiyse ■, iyi düzeyde bir sonuç elde ettiyse ■ simgesiyle gösterilir. ■ simgesi öğrencinin o ünite ile ilgili testi çözmediğini gösterir.

Geliştirilen sistem, öğrencilere konu sunumu yaparken öğrencilerin öğrenme stillerini referans alır. Şekil 4.21'de görüldüğü gibi görsel öğrenciler için şema, sözel öğrenciler için dersin sesli sunumu, deneysel öğrenciler için uygulamalar linki eklenmiştir. Her adımın mantıksal olarak bir başka adımı takip ettiği doğrusal adımlar şeklinde anlama eğiliminde olan ardışık öğrenciler için Şekil 4.22'deki kavram haritası çizilmiştir.



Şekil 4.21. Kullanıcı arabiriminden örnek bir konu anlatım sayfası



Şekil 4.22. Ardışık anlama yeteneğine sahip öğrenciler için kavram haritası örneği

Sistem bazı durumlarda dışa yönelimli öğrencileri motive etmek amacıyla bir takım uygulamalar sunmaktadır. Şekil 4.23’de örnek bir motivasyon uygulaması sunulmaktadır.

EN BÜYÜK ENGEL İNSANIN İÇİNDEKİ ENGELDİR.

Sirklerde eğitilen yavru fillerin ayaklarına, sağlam bir kazığa bağlanmış bir zincir takılır. Uzun bir müddet bu şekilde bakılıp beslenen yavru fil artık büyüüp çok güçlü hale geldiği halde, ayağına çok ince ve kolay koparılabilecek bir ip bağlandığı halde asla onu koparıp kurtulmayı düşünmez, düşünemez. Çünkü aklında hala tutsak olduğu inancı vardır. Birçok insanda, filler gibi, bir kez içine düştükleri çarktan bir türlü kurtulamazlar. En büyük engel insanın içindeki engeldir.



Şekil 4.23. Motivasyon uygulaması

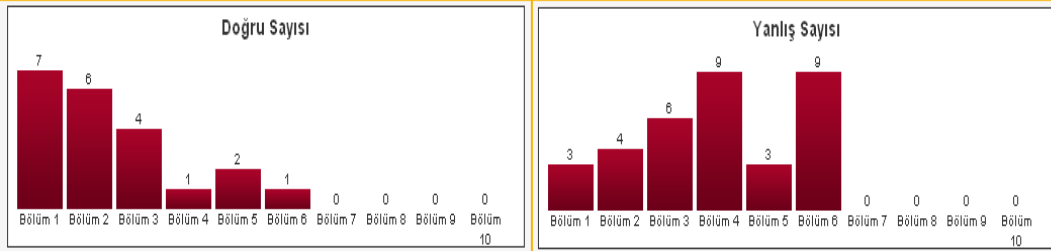
4.4.3 Bireysel Raporlar

Öğrenci isterse ana sayfadaki bireysel raporlar linkinden Şekil 4.24’de görüldüğü gibi başarı durumu ve kullanım istatistikleri ile ilgili bilgi alabilir. Bu bölümün geliştirilmesindeki amaç, öğrenciye başarı durumu ile ilgili rehberlik etmek ve hazırlanan grafiklerle konuya olan ilgisini artırmaktır.

Öğrenci Raporları

Adı	Soyadı	Bölüm Adı	Durum	Rehberlik
bihter	bulut	Dinamik Sistemlerin Matematiksel Modellenmesi	Kötü	Çok çalışmalısınız.
bihter	bulut	Denetleyiciler ve Kompansatörler	Çok Kötü	Çok fazla eksisiniz var. Derse olan ilgi ve isteginizi artırmalısınız.
bihter	bulut	Denetim Sistemlerinin Matematiksel Temelleri	Orta	Biraz daha gayret etmelisiniz.
bihter	bulut	Denetim Sistemlerine Giriş: Temel Kavramlar	İyi	Daha iyi olabilirsiniz.

oturum	Adı	Soyadı	Sisteme Giriş	Sistemden Çıkış	Sisteme Kalma Süresi
1	bihter	bulut	2009-06-10 14:23:17	2009-06-10 15:23:19	1 SAAT 0 DAKİKA 2 SANİYE
2	bihter	bulut	2009-06-10 16:25:13	2009-06-10 18:25:16	2 SAAT 0 DAKİKA 3 SANİYE
3	bihter	bulut	2009-06-10 20:29:22	2009-06-10 20:32:29	0 SAAT 3 DAKİKA 7 SANİYE



Şekil 4.24. Öğrenci raporları

4.4.4 Öğrenme Stili Belirleme Testi

Öğrencinin öğrenme stilini belirlemek amacıyla Şekil 4.25’de görüldüğü gibi öğrenciye genel sorular sorulmaktadır [92]. Bu sorulara verilen cevaplara göre, sistem öğrencinin öğrenme stili hakkında ön bilgi sahibi olmaktadır.

STİ testi - Microsoft Internet Explorer

Dosya Düzen Görünüm Sekülerler Araçlar Yardım

Geri İleri Duraklatma Yenileme Sil Kütüphane Arama Sık Kullanılanlar

Adres <http://localhost/intelligentphp/ogrenmestili.html> Git Başlıklar

NOT: Sistemin öğrenme stilinize uygun davranışlarınıza ilişkin sorulara doğru yanıtı veriniz.

1)Yapacak daha iyi bir şey bulamadığınız birkaç dakikanız var, aşağıdakilerden hangisini yapmayı tercih edersiniz?

- ☐ Boşluğa bakmayı ya da bir şeyler karalamayı.
- ☐ Kendi kendinize veya birileriyle konuşmak.
- ☐ Okumak için bir şeyler araştırmak.
- ☐ Bir şeylerle uğraşmak, odadaki bir şeyleri düzenlemek yerini değiştirmek gibi.

2)'BAĞIMLI' ya da 'BAĞIMSIZ' hangisinin doğru olduğundan emin değilsiniz. Bu durumda,

- ☐ Sözünce bakarsınız.
- ☐ Kelimeyi gözünüzün önüne getirerek doğru seçeneği bulmaya çalışırsınız.
- ☐ Kelimeyi içinizden seslendirirsiniz.
- ☐ Kelimeleri birer kağıda yazıp katedikten sonra birini seçersiniz.

3)En iyi arkadaşınız için, sürpriz bir doğum günü partisi hazırlamak istiyorsunuz. Nasıl yaparsınız?

- ☐ Diğer arkadaşlarınızla konuşup, fikir alış veriş yaparak.
- ☐ Yapacaklarınız ve alacaklarınızın bir liste halinde belirtebilirsiniz.
- ☐ Parti için yapacaklarınız beyrinizde canlandırarak.
- ☐ Arkadaşlarınızı davet edip, nasıl gelirse ona razı olmak.

Şekil 4.25. Öğrenme stili ön testi

4.4.5 Sanal Deneyler

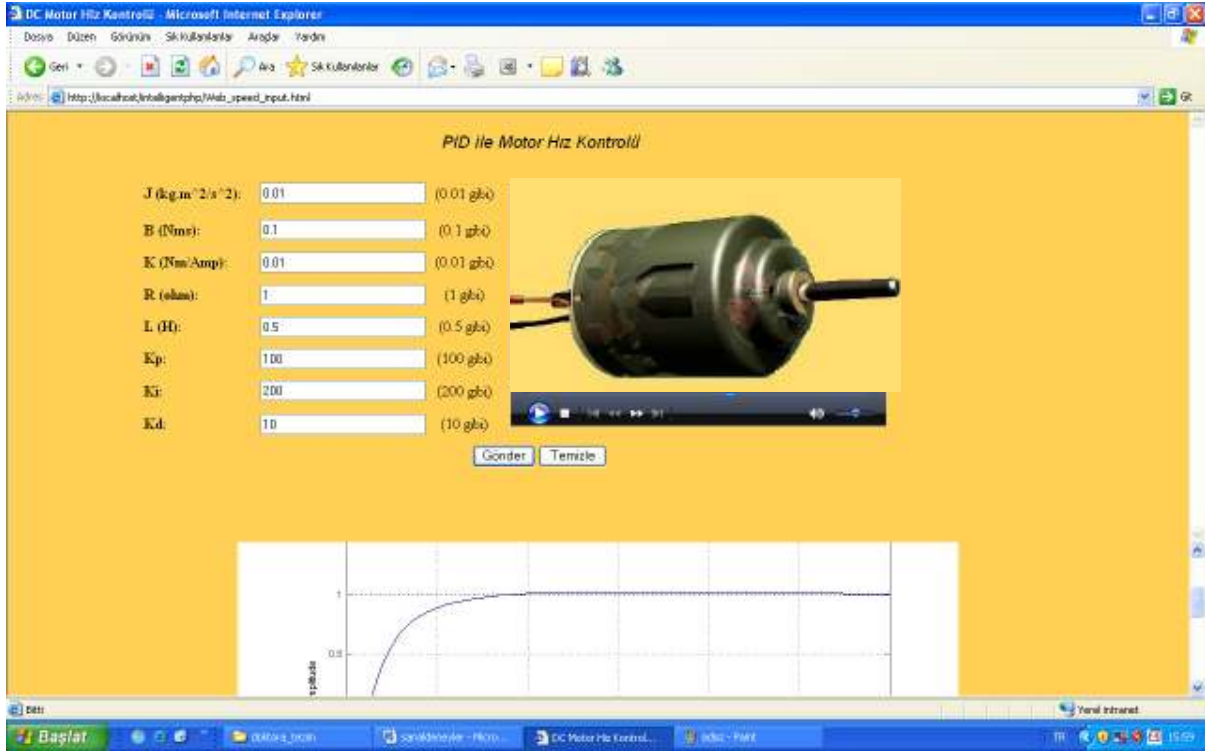
Sanal deneyler bölümüne; Şekil 4.26’da görüldüğü gibi top çubuk sistemi, DC motor hız ve konum kontrolü, araç hız kontrolü, tank sıvı seviyesi, jiro-sarkaç uygulamaları gibi deneysel çalışmalar konulmuştur [93]. Bu bölüm ile öğrencilere eşzamansız deney yapabilme, kendi kendine öğrenebilme, geribildirim yolu ile kendini değerlendirebilme gibi imkânlar sunulmaktadır.



Şekil 4.26. Sanal deneyler sayfası

Deneyler, internette etkileşimli üç boyutlu ortamlar oluşturmakta kullanılan Virtual Reality Modeling Language (VRML) dili kullanılarak hazırlanmıştır. VRML, internet üzerinden 3 boyutlu içerik yayınlamak amacıyla, özel kodlama kuralları çerçevesinde geliştirilen “.wrl” uzantılı dosyaların, web tarayıcılarının yardımıyla yorumlayabilecekleri şekilde geliştirilmesini sağlayan bir dildir [94]. Bu dosyaların bilgisayarda görüntülenebilmesi için bir VRML oynatıcısı eklenti yazılımı kurulu olmalıdır. Yaygın olarak kullanılan eklenti Cosmo Player eklentisidir.

Şekil 6.27’de görüldüğü gibi, internet üzerinden deneylere kullanıcıların erişimi sağlanmış, modellerde bulunan parametreleri değiştirebilme ve bu değişim sonucunu grafiklerle izleyebilme imkânı oluşturulmuştur. Modellerde yer alan durum denklemlerinin çözümü MATLAB yazılımı ortamında yaptırılmıştır. Bu amaçla MATLAB yazılımının bir bileşeni olan WebServer kullanılmıştır.



Şekil 4.27. Motor hız kontrolü sanal deneyinin parametre değiştirme sayfası

4.4.6 Forum

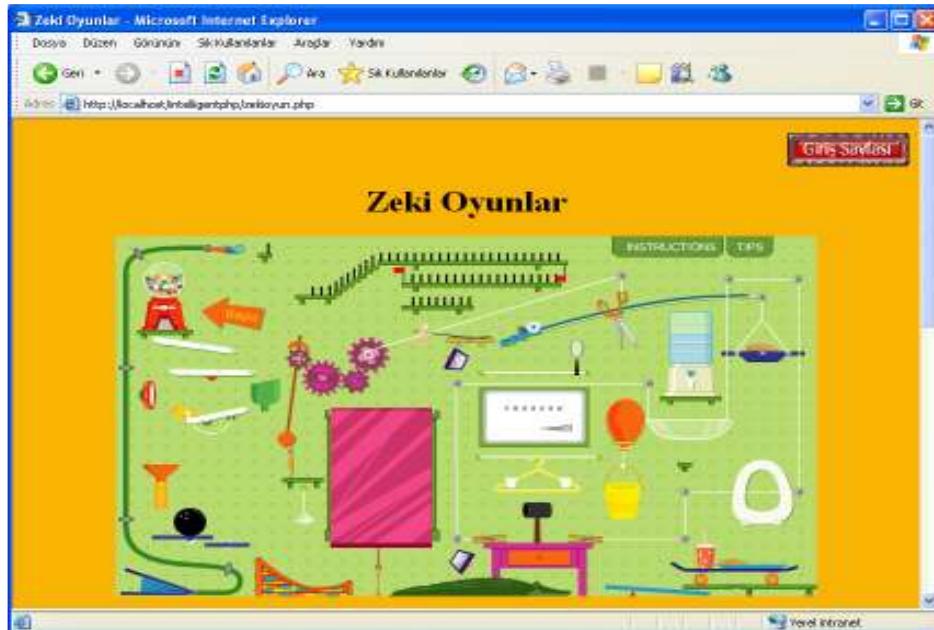
Öğrenciler, anasayfadaki forum linkinden giriş yaptıkları Şekil 4.28’deki bu platformda bilmediği ve merak ettiği sorularını sorup cevap alabilirler, bilinen bir konuyu başlık haline getirip forumdaki öğrencilerin görüşlerini öğrenebilirler. Geliştirilen sisteme forum, bilgiyi en iyi tartışarak, başkalarına açıklayarak alıp ve anlayan deneysel öğrenciler için dahil edilmiştir.



Şekil 4.28. Forum sayfası

4.4.7 Zeki Oyunlar

Geliştirilen sistemde yer alan zeki oyunlar linki, öğrencileri motive etmek ve küresel anlama yeteneğine sahip öğrencilere yardımcı olmak amacıyla eklenmiştir. Şekil 4.29’da görülen zeki oyunlar büyük sıçramalar şeklinde öğrenme eğiliminde olan bu öğrencilerin ani karar verme, doğru ve hızlı düşünme, hızlı hareket etme gelişimine katkıda bulunurlar.



Şekil 4.29. Zeki oyunlar sayfası

4.5 Geliştirilen Yazılımın Öğrenciler Üzerindeki Uygulamaları

Geliştirilen web tabanlı akıllı öğretim sistemi, Fırat Üniversitesi, Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Bölümü, Elektronik Öğretmenliği 3.sınıf öğrencilerine uygulanmıştır. Kontrol Sistemleri I dersi müfredatı, uygulama konusu olarak seçilmiştir. Sistem kullanımı sonrasında öğrencilerin başarısında artış olup olmadığı incelenmiştir.

4.5.1 Uygulama ve Sonuçlar

Bu tez çalışmasında, öğrenme sürecinde öğrenciye rehberlik yapan, öğrencinin hazır bulunuşluk düzeyini belirleyen ve öğrenciyi tanılamaya uygun yönergeler sunarak bireysel farklılıklarına göre öğrenmesini sağlayan web tabanlı bir uzman sistem (akıllı öğretim sistemi) geliştirilmiştir. Tasarlanan akıllı öğretim sistemi (AÖS); öğrenci bilgisinin düzeyini temsil eden öğrenci modelleme birimi, öğrencinin bilmesi gereken konuları içeren ve bu doğrultuda pedagojik kararların alınmasını sağlayan bilgi anabirimi, öğrenciyle iletişim kurulmasını sağlayan kullanıcı arabirimi, sahip olduğu veriler ve imkanlar dahilinde sonuçlara erişme sırasında kullanıldığı mantık sürecini sağlayan çıkarım mekanizmasından oluşmaktadır.

Geliştirilen yazılımda, öğrencinin yapısını belirlemek en önemli bölüm olmuştur. Öğrenci modeli oluşturulurken Felder-Silverman öğrenme stilleri modeli kullanılmıştır. Öğrencilerin algılama, girdi şekli, organizasyon, işleme ve anlama yeteneklerine bağlı olarak parametreler oluşturulmuştur. Elde edilen bu parametreler değerlendirilerek öğrenci performansını belirleyen ve elde ettiği sonuçları yorumlayan bir uzman sistem tasarlanmıştır.

Sistemde kullanılan parametrelerin klasik eğitim ortamında öğreticiler tarafından öğrenci değerlendirmede kullanılan verilerden elde edilmesine özen gösterilmiştir. Değerlendirmede çok etkin olmasına rağmen doğru olarak elde edilmesi güç olan parametreler sistemde kullanılmamıştır.

Bu çalışmanın asıl önem taşıyan yanı, yapay zeka tekniklerinden tip-2 bulanık mantığın tasarıma dahil edilmesidir. Tip-2 bulanık mantık öğrenci modellemede, öğrencinin hareketleri, bilgisi ve bilişsel yetenekler gibi kesin olmayan bilgileri

yakalamadaki yeteneğinden dolayı tercih edilmiştir. Elde edilen sonuçlar, yapay zeka tekniklerinin, bir çok alanda olduğu gibi eğitim alanında da öğrenme-öğretme sürecinin etkili ve verimli bir şekilde gerçekleşmesinde alternatif bir yaklaşım olarak kullanılabileceğini açık bir şekilde göstermektedir.

4.5.2 Uygulama Başarısının Değerlendirilmesi

Gerçek dünya durumlarına yapay zekâ tekniklerini uygulayan uyarlanabilir bir sistemin gerçekten çalışıp çalışmadığına karar vermek diğer bir deyişle, etkililiği, verimliliği ve kullanıcı memnuniyeti sağlayıp sağlamadığını belirlemek amacıyla değerlendirilmesi gereklidir. Bu amaçla, öntest-sontest kontrol gruplu araştırma modeli kullanılarak deneysel bir çalışma yapılmıştır. Deneysel araştırma modelinin gereği olarak Kontrol Sistemleri-I dersini alan Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümü Elektronik Öğretmenliği 3.sınıf öğrencilerinden 18'şer kişilik kontrol grubu ve deney grubu oluşturulmuştur. Kontrol grubuna geleneksel yöntemlerle, deney grubuna ise hazırlanan sistemle konular öğretilmiştir. Gruplara ön testler uygulandıktan sonra uygulamalara başlanmıştır. Uygulamalar tamamlandıktan bir süre sonra gruplara başarı testi son test olarak uygulanmıştır.

Deneysel çalışma sonunda deney grubundan elde edilen veriler, geleneksel öğretim yöntemlerinin uygulandığı kontrol grubundan elde edilen verilerle karşılaştırılarak gerekli analizler SPSS 15.0 paket programı kullanılarak yapılmıştır. Bu analizler içinde bağımsız gruplar için t-testi eşleştirilmiş gruplar için t testi ve tek faktörlü kovaryans analizi (ancova) kullanılmıştır.

Deney ve kontrol grubu öğrencilerinin ön testten aldıkları puanların bağımsız t-testi analiz sonuçlarına göre deney grubu öğrencilerinin ortalaması 52.5, kontrol grubu öğrencilerinin ortalaması ise 45.28 olarak hesaplanmıştır. Akademik başarı seviyeleri farklı öğrencilerden oluşan gruplara ön test olarak uygulanan başarı testinden aldıkları puanlar arasında anlamlı bir farklılık olup olmadığını belirlemek için yapılan t- testi sonuçlarına göre p değeri anlamlılık düzeyi 0,05 alındığında, öğrencilerin bilgi düzeyleri arasında istatistiksel olarak anlamlı bir farkın olmadığı görülmektedir. Bu bilgi Tablo 4.8'de görüldüğü gibi kontrol grubu ile deney grubunu oluşturan öğrencilerin düzeylerinin birbirine yakın olduğunu göstermektedir.

Tablo 4.8. Kontrol ve Deney Grubu öntest independent-samples t-testi sonuçları

Gruplar	N	X	SS	Sd	t	p
Kontrol Grubu	18	45.28	8.13	34	2.08	.045
Deney Grubu	18	52.50	12.23			

Tablo 4.9 ve 4.10’da ise deney grubu ve kontrol grubuna uygulanan son test sonuçlarına göre gruplar arasında $p < 0,01$ olduğundan grupların son testten aldıkları puanlar arasında istatistiksel olarak anlamlı bir fark olduğu görülmektedir. Deney grubunun aritmetik ortalaması 63.388, kontrol grubunun aritmetik ortalaması 51.833 olarak hesap edilmiştir. Deney grubu ile kontrol grubu arasında görülen fark deney grubunun lehinedir. Deney grubu ile kontrol grubu arasında anlamlı farklılık olduğu saptanmıştır. Yani, deney grubu öğrencileri, kontrol grubu öğrencilerine göre daha fazla başarılıdır.

Tablo 4.9. Kontrol grubu öntest ve sontest ortalama puanlarının paired-samples t-testi sonuçları

Ölçüm	N	X	SS	Sd	t	p
kontrol grubu öntest sonucu	18	45.278	8.13	17	5.15	.000
kontrol grubu sontest sonucu	18	51.833	8.82			

Tablo 4.10. Deney grubu öntest ve sontest ortalama puanlarının paired-samples t-testi sonuçları

Ölçüm	N	X	SS	Sd	t	p
deney grubu öntest sonucu	18	52.500	12.277	17	8.47	.000
deney grubu sontest sonucu	18	63.388	13.456			

Tablo 4.9 ve 4.10’da görüldüğü gibi, kontrol grubunun ön test puan ortalaması 45.278, son test puan ortalaması ise 51.833; deney grubuna uygulanan ön test puan ortalaması 52.5, son test puan ortalaması da 63.388 olarak bulunmuştur. Bu analiz sonuçlarından anlaşıldığı gibi, deney grubunun öntestteki başarı düzeyi, kontrol grubunun öntestteki başarı düzeyine göre daha yüksektir.

Tablo 4.11. Sontest puanlarının gruplara göre betimsel istatistikleri

Gruplar	N	Ortalama	Düzeltilmiş Ortalama
Kontrol Grubu	18	51.83	55.30
Deney Grubu	18	63.39	59.92

Kovaryans analizi, ön testte eşitlenmemiş olan kontrol ve deney gruplarının sontest puan ortalamalarının birbirinden anlamlı bir şekilde farklılık gösterip göstermediğini test eder. Tablo 4.11’de görüldüğü gibi kovaryans analizi ile kontrol grubunun düzeltilmiş sontest puan ortalaması 55.30, deney grubunun düzeltilmiş sontest puan ortalaması ise 59.92 olarak hesaplanmıştır. Başlangıçtaki öntest puanları kovariat olarak alınmak suretiyle, son test puanlarının kontrol ve deney gruplarındaki farklılığı, kovaryans analiz tekniği kullanılarak analiz edilmiştir.

Tablo 4.12. Önteste göre düzeltilmiş sontest puanlarının yönteme göre ANCOVA sonuçları

Varyansın Kaynağı	Kareler Toplamı	sd	Kareler Ortalaması	F	p
Öntest	3406.086	1	3406.086	112.774	.000
Yöntem	169.888	1	169.888	5.625	.024
Hata	996.692	33	30.203		
Toplam	5604.556	35			

Tablo 4.12’deki kovaryans analizi sonuçlarına göre, deney ve kontrol grubunun düzeltilmiş sontest puan ortalamaları arasındaki fark anlamlı bulunmuştur ($p<0.05$). Tasarlanan uzman sistem kullanılarak öğretimin gerçekleştirildiği deney grubunun ortalama puanı 63.388 ile geleneksel yöntemle öğretimin gerçekleştirildiği kontrol grubunun puan ortalamaları 51.833 arasında önemli bir fark olduğu görülmektedir. Bu sonuca göre hazırlanan sistemin, laboratuvar ortamında uygulanan geleneksel yöntemlere göre öğrenmede daha etkili olduğu belirlenmiştir.

5. SONUÇ VE DEĞERLENDİRME

Günümüzde internet, insanlar için geniş bir bilgi kaynağı haline gelmiştir. Bu bağlamda, pek çok eğitim organizasyonu Web teknolojilerini kullanmakta ve eğitim içeriklerini dünya ile paylaşmaktadır. Web teknolojileri, eğitim teknolojileri için gelişmiş bir uygulama platformu olmuştur. Sadece ilköğretim, lise ve üniversite gibi eğitim kurumları için değil, ticari kurumsal kimliğe sahip tüm organizasyonlar hizmet içi ve mesleki gelişim eğitimlerinin sunulmasında Web teknolojileri artık etkin bir platformdur [95-98]. E-Öğrenme, bu platformun en önemli uygulama alanlarından biridir. E-öğrenme ile sunulan eğitimlerden hem öğrencinin başarısı hem de öğrenci başarısının kısa sürede gerçekleşmesi istenilen asıl hedeftir. Kişisel başarının artmasındaki temel referans noktası, kişilere bireysel farklılıklarını gözeterek eğitim sunmaktır. Bireylerin farklı kişilik özellikleri taşımaları, farklı öğrenme biçimlerine sahip olmaları, bilgiyi farklı şekillerde işlemeleri, farklı bilgi kaynaklarını kullanmayı tercih etmeleri, aynı ortamı kullanırken öğrenme gereksinimlerinin de farklılaşmasına neden olmaktadır. Bu noktada, uyarlanır ve zeki özelliklere sahip e-öğrenme sistemleri alanındaki araştırmalar değer kazanmaktadır.

Bu tez çalışmasında, öğrenciler arasındaki farklılıkları gözetken ve her bir öğrencinin öğrenmedeki kişisel gereksinimlerine göre farklı öğrenme imkânları sunan bir sistem geliştirilmiştir. Web tabanlı öğretim yapan bu sistem ile dersi alan öğrencilerin öğrenme, kavrama, değerlendirme ve sonuçlandırma aşamaları belirli bir algoritma ile gerçekleşmektedir. Algoritma adımları,

1. Dersi alan öğrenci öncelikle kendi hesabı ile sisteme dâhil olduktan sonra bir ön teste tabi tutulur ve öğrenci modeli oluşturulur.
2. Öğrencinin konular hakkındaki ön bilgisi ölçülerek, her bir öğrenciye çalışabileceği uygun konular önerilmektedir. Önerilen konularla ilgili öğrencinin maksimum düzeyde tatmin olabilmesi açısından konuya ilişkin kapsamlı bilgilere yönlendirme yapılmaktadır. Bu tür uyarlamalar, öğrencilerin öğrenme sürecini destekleyerek, bilgi aktarımını artırabileceği gibi, öğrencilerin gezinme sürecini kolaylaştırması ve kendilerine göre kişiselleştirilmiş seçenekler içermesinden dolayı motivasyonlarını artıracaktır.

3. Konu içeriğini aldıktan sonra konu testini cevaplandıran öğrencinin öğrenci modeli başarısız olduğu konulara göre güncellenir ve başarısız olduğu konuları incelemek ve sorularını cevaplamak için konu içeriği sayfalarına yönlendirilir.
 4. Öğrenci tüm konulardan başarılı olana kadar sistem, konu içeriğini sunmakla ve sorularını sormakla görevlidir.
- şeklinde sıralanabilir.

5.1. Sonuçların Değerlendirilmesi

Bu tez çalışmasında, öğrenciler arasındaki farklılıkları gözeten ve her bir öğrencinin öğrenmedeki kişisel gereksinimlerine göre farklı öğrenme imkânları sunan web tabanlı bir uzman sistem geliştirilmiştir. Yapılan araştırma ve incelemelerde, gerçekleştirilen uygulamalar neticesinde elde edilen sonuçlar ve gözlemler ortaya konulmuştur.

1. Akıllı öğretim sistemleri gelişime açık ve geleceğin öğretim sistemleridir. Bu sistemlerde yapay zekâ tekniklerini kullanarak içeriği şekillendirmenin başarı düzeyini yükselttiği görülmüştür.
2. Literatürde yapay zeka tekniklerini kullanarak web tabanlı uzman sistem geliştirilmesine yönelik çalışmalar bulunmaktadır. Ancak sistemin çıkarım mekanizmasında fonksiyon ve kuralların tip-2 bulanık üyelik özellikleri ilk defa kullanılmıştır.
3. Yapılan değerlendirmeler sonucunda deney grubunun öntest ve sontest puan ortalamaları t-testi ile istatistiksel olarak analiz edilmiş, analiz sonucunda öntest ve sontest puan ortalamaları arasında anlamlı bir fark olmadığı sonucuna ulaşılmıştır. Buna göre sistemin öğrenmeyi artırdığı belirlenmiştir. Ayrıca deney grubu ve kontrol grubunun düzeltilmiş sontest puan ortalamaları arasındaki farkın kovaryans analizi sonucu anlamlı olduğu görülmüştür. Buna göre hazırlanan sistem kullanılarak öğretimin gerçekleştirildiği deney grubunun öğrenme düzeyi, geleneksel yöntemlerle öğretimin gerçekleştirildiği kontrol grubunun öğrenme düzeyine göre daha yüksektir. Bu sonuca göre kontrol sistemleri dersi için hazırlanan web tabanlı uzman sistem geleneksel yöntemlere göre öğrenmede daha etkilidir.
4. Geliştirilen uygulama sayesinde öğrenciler sistemde istedikleri hızda ilerleyebilir, anlamadıkları konuları tekrar edebilirler. Aynı zamanda, hazırlanan ön test ile

- öğrencinin bilgisi ölçülebilir ve eğitimin sadece ihtiyacı olan kısmını alması sağlanabilir. Öğrenci her bölümü tamamladığında bilgisini ölçer, eğer başarılı olursa yeni bölüme geçip devam eder, başarılı olmazsa aynı bölümü tekrar edebilir. Böylece öğrencilere bireyselleştirilmiş bir eğitim imkânı sunulmaktadır.
5. Öğrenci sisteme internet bağlantısı olan her bilgisayardan erişebilmekte ve istediği yerde bırakıp, daha sonra kaldığı yerden devam edebilmektedir. Başarılı olana kadar içeriği alıp konu testini çözebilmektedir. Bu sayede öğrenme hızı düşük olan öğrencilerin de ihtiyacı karşılanabilmektedir.
 6. Öğrencinin bireysel özelliklerinin eğitimci personel tarafından belirlenmesi, önemli ölçüde hem zaman hem de finansman gerektiren bir durumdur. Geliştirilen sistemde bu özelliklerin belirlenmesi için kullanılan çıkarım mekanizması sayesinde eğitimci personel yerine bilgisayar sistemlerinin kullanılması ile zaman problemi aşıp maddi imkânlar zorlanmamıştır.
 7. Öğrenci bireysel özellikleri belirlenirken bilgisayar kullanıldığı için insanlarda var olan objektif değerlendirmeden uzaklaşma, değişik zamanlarda ve içinde bulundukları farklı durumlarda farklı yorumlar yapma gibi problemler söz konusu değildir. Bu durumda ise bilgisayarların insanlar gibi düşünememesi ve yorum yapamaması düşünceleri akla gelmektedir. Ancak günümüz teknolojilerinden yapay zeka ile bu düşünceler açığa kavuşacaktır. Tasarlanan sistemde kullanılan tip-2 bulanık mantık sayesinde öğrencinin hangi öğrenme stilinde olduğu kesin ifadelerle değil, birçok öğrenme stiline belirli oranlarda sahip olabileceği düşünülerek belirlenmiştir [99, 100]. Böylece bir yapay zeka türü olan tip-2 bulanık mantık algoritmaları ile öğrenme stilleri kesin çizgiler arasına sıkışmaktan çıkıp daha esnek sınırlar içerisinde ortaya çıkacaktır.
 8. Geliştirilen sistemde tip-2 bulanık mantık kullanımının önemli bir avantajı da, öğrenci gözlemleri sırasında oluşan belirsizlikleri iyi modelleyerek elde edilebilecek verinin güvenilirliği artırmaktır. Ancak aynı amaçla kullanılan Bayes ve Dempster Shafer teorileri sadece verilerin sayısal olarak ifade edildiği durumlarda kullanışlıdır.
 9. Özellikle öğretmen–öğrenci etkileşiminin kolayca sağlanamayacağı kalabalık sınıflarda öğrenciler hakkında değerlendirmeye yönelik ayrıntılı bilgi elde edilmesi

oldukça zordur. Ancak hazırlanan sistemle öğrenci hakkında yeterince bilgi edinilmekte ve bilgiler ışığında dönüt verilmektedir.

10. Öğrenciye her an değerlendirilme sürecinde olduğu hissi verilerek dikkatinin canlı tutulması sağlanmıştır.

5.2 Öneriler

1. Uyarlanabilir bir öğrenme sisteminde gerçekleştirilen uyarlama türü (içerik/gezinme), kullanılan uyarlama tekniği (bağlantı üretme/bağlantıları açıklama/ ek açıklamalar vb.), bu tekniklerin uygulanış biçimi, bu tekniklerin geliştirilmesinde temel alınan özellikler (görevle ilgililik/ ön bilgilere uygunluk vb.) ile modelleme ve sonuç çıkarma mekanizmaları (elle oluşturulmuş kurallar/bulanık mantık/bayes ağlar) farklılaştıkça ortamın kullanışlılığı/etkililiği değişmektedir. Bu nedenle yukarıda listelenen her bir boyuta ilişkin yapılacak çalışmalar alana katkı sağlayacaktır.[101, 102].
2. Öğrencinin değerlendirilmesinde gerçeğe yakın sonuçlar alınabilmesi için doğru-yanlış, eşleştirmeli, boşluk doldurmalı gibi web ortamında yapılması mümkün olabilecek farklı sınav çeşitleri de kullanılabilir [103-112].
3. Web kullanıcı erişim kütüklerinin Web kullanım madenciliği ile analiz edilmesi sonucunda öğrenci davranışları ve materyaller hakkında birçok istatistiksel bilgi çıkarımları yapılabilmektedir [113-115]. Bu bilgi çıkarımları sisteme dahil edilerek daha kaliteli bir öğrenci modeli ve daha kaliteli bir AÖS geliştirilebilir.
4. Öğrenci modeli oluşturulurken öğrenci hakkında daha fazla bilgi edinebilmek için ön stil testi ile birlikte zeka testi, kişilik testi gibi psikolojik testlerden de yararlanılabilir.
5. Geliştirilen sistem sınırlı sayıda öğrenciye uygulanmıştır. Daha geniş kapsamlı bir uygulamanın yapılmasında, daha çok geri bildirimin sağlanması açısından yarar vardır.
6. İyi tanımlanmış ve ilişkilendirilmiş olan bilgilerin ve servislerin web ortamında kolay bir şekilde bilgisayarlar tarafından okunabilir ve bilgisayarlar tarafından anlaşılabilir olmasını sağlayacak standartların ve teknolojilerin geliştirilmesi olarak

tanımlanan semantik web, e-öğrenme uygulamalarının gerçekleştirilmesi için çok uygun bir platformdur [116,117].

7. Geliştirilen sistemde öğrenci modeli oluşturulurken, göz önünde bulundurulan bilgi düzeyi, öğrenme metodu, bir bilgi ögesine karşılık gösterdiği davranış ve bir bilgi ögesi için harcadığı zaman gibi giriş parametrelerine yenileri eklenebilir.

KAYNAKLAR

- [1] **Önder, H. H.**, 2001. Yapay Zeka Programlama Teknikleri Ve Bilgisayar Destekli Eğitim, Uluslararası Eğitim Teknolojileri Sempozyumu Bildirileri, Sakarya Üniversitesi Eğitim Fakültesi.
- [2] **Doğan, N., Kubat B.**, 2008. Akıllı Öğretim Sistemleri için Yeni Bir Bileşen: Düzenleyici Modül, Bilişim Teknolojileri Dergisi, cilt 1, sayı2.
- [3] **O'Shea, T.**, 1983. Learning and Teaching with Computers: Artificial Intelligence in Education, Hervaster Pres. First ed. Brighton, 4.
- [4] **Brown, J. S., Burton R. R., Bell, A. G.**, 1975. SOPHIE: A Step Toward Creating a Reactive Learning Environment, International Journal of Man-Machine Studies, 7, 675-696.
- [5] **Brown, J. S., Burton R. R.**, 1978. Diagnostic Models for Procedural Bugs in Basic Mathematical skills, Cognitive Science, 2, 2, 155-192.
- [6] **Brown, J. S., Burton R. R.**, 1976. A Tutoring and Student Modelling Pradigm for Gaming Environments, Computer Science and Education, 8, 19, 236-246.
- [7] **Aydın, Y. S.**, 2000. Visual Prolog İle Programlama-Yapay Zeka ve Uzman Sistemler, Sistem Yayıncılık No:253, İstanbul.
- [8] **Turan, F.**, 2007. Stereotip öğrenci modeli kullanarak zeki öğretim sistemi tasarımı, *Yüksek Lisans Tezi*, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- [9] **Büçrö, E., Ö.**, 2003. Web-tabanlı akıllı eğitimde uyarlanıır içerik sunumu sisteminin Bayesian ağı yaklaşımı ile tasarımı ve gerçekleştirilmesi, *Yüksek Mühendislik Tezi*, Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, Ankara.
- [10] **Öncü, A., Varol, H. S.**, 2005. Web Tabanlı Uzaktan Öğretimin Verimliliğini Artırma Yaklaşımları- Yapay Zeka Teknolojilerinin Kullanımı, *1. Uluslararası Mesleki ve Teknik Eğitim Teknolojileri Kongresi*.
- [11] **Öncü, A.**, 2006. Uzman sistem yaklaşımı ile web tabanlı öğretim değerlendirme sisteminin geliştirilmesi, *Doktora Tezi*, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- [12] **Doğan, B.**, 2006. Zeki Öğretim Sistemlerinde Veri Madenciliği Kullanılması. *Doktora Tezi*, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.

- [13] **Kaya, S.**, 2005. Microsoft Excel Öğretimi İçin Zeki Öğretim sistemi, *Yüksek Lisans Tezi*, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara.
- [14] **Luan, J.**, 2002. Data Mining and Its Applications in Higher Education. New Directions for Institutional Research, 2002, 113, 17–36.
- [15] **Yoshikawa, A., Shintani, M. and Ohba, Y.**, 1992. Intelligent Tutoring System for Electric Circuit Exercising, IEEE Transactions On Education, vol. 35, no. 3.
- [16] **Negnevitsky, M.**, 1996. Application of an intelligent tutoring system in electrical engineering education, IEEE.
- [17] Brusilovsky P., Adaptive and Intelligent Technologies for Web-based Education, <http://www2.sis.pitt.edu/~peterb/papers/KI-review.pdf>, Eylül 2006.
- [18] **Brusilovsky, P., Eklund, J., Schwarz, E.**, 1998. Web-based Educations for All: A Tool for Development Adaptive Courseware, *In Proceedings of the Seventh International World Wide Web Conference*.
- [19] **Peylo, C., Brusilovsky, P., Ritter, S., Rollinger, C., Stern, M. K. and Weber, G.**, 2000. Adaptive and Intelligent Web-Based Education Systems, International Journal of Artificial Intelligence in Education, 11, 1050-1059.
- [20] **Gürbüz, A.**, 2006. Zeki Kavramsal Model Geliştirme Aracı, *Doktora Tezi*, Fen Bilimleri Enstitüsü, Marmara Üniversitesi, İstanbul.
- [21] **Dağ, F. and Erkan, K.**, 2004. Prolog Tabanlı Zeki Öğretim Sistemi, Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, özel sayı, cilt 10, 47-55.
- [22] **Tamer, T.**, 2002, Yapay Zeka Programlama Tekniklerinin Bilgisayar Destekli Eğitimde Kullanımına İlişkin Bir Model, *Yüksek Lisans Tezi*, Gazi Üniversitesi, Fen Bilimleri Enstitüsü. Ankara.
- [23] **Yavuz, U., Karaman S.**, 2003. Öğretim Yazılımlarının Zekileştirilmesi: Bir Değerlendirme Makinesi, Çukurova Üniversitesi Akademik Bilişim 2003, Adana.
- [24] **Özdemir, B.**, 2000. Development of an intelligent agent for distance learning, *Yüksek Lisans Tezi*, Ortadoğu Teknik Üniversitesi Fen Bilimleri Enstitüsü, Ankara.
- [25] **Tseng, J., Wu, C.**, 2007. An expert system approach to improving stability and reliability of web service, *Expert Systems with Applications*, 33, 379–388.

- [26] **Granic, A., Stankov, S. and Glavinic, V.,** 2000. User Interface Aspects of an Intelligent Tutoring System, 22nd Int. Conf. Information Technology Interfaces.
- [27] **Nascimento, D., Domínguez, A., Silva, A.,** 2005. Developing a Teaching Supporting Tool based on Electronic Portfolio, Agents and Intelligent Tutoring System, Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies.
- [28] **Piramuthu, S.,** 2005. Knowledge-Based Web-Enabled Agents and Intelligent Tutoring System, IEEE Transactions On Education, vol. 48, no. 4.
- [29] **Frasson, C. and Aimeur, E.,** 1998. Designing a multi-strategic ITS for Training in Industry, Elsevier Science Computers in Industry, 37, 153-167.
- [30] **Stottler, R., Vinkavich M.,** 2000. Tactical Action Officer Intelligent Tutoring System (Tao Its), Proceedings of the Industry/Interservice, Training, Simulation & Education Conference (I/ITSEC 2000).
- [31] **Cheung, B., Hui, L., Zhang, J. and Yiu, S. M.,** 2003. SmartTutor: An intelligent tutoring system in web based adult education, The Journal of System and Software, 68, 11-25.
- [32] **Hsieh, S., Hsieh, P., Zhang, D.,** 2003. Web-Based Simulations And Intelligent Tutoring System For Programmable Logic Controller, 33rd ASEE/IEEE Frontiers in Education Conference.
- [33] **Keleş, A., Ocak R., Keleş, A., Gülcü, A.,** 2009. ZOSMAT: Web-based intelligent tutoring system for teaching-learning process, Expert Systems with Applications, 36(2), 1229-1239.
- [34] **Rosic, M., Stankov, S., Glavinic, V.,** 2000. Intelligent Tutoring Systems for Asynchronous Distance Education, 10th Mediterranean Electrotechnical Conference, MEleCon 2000, Vol. I.
- [35] **Butz, C. J., Hua, S., Maguire, R. B.,** 2004. A Web-based Intelligent Tutoring System for Computer Programming, Proceedings of the IEEE/ACIS International Conference on Web Intelligence.
- [36] **Wijekumar, K., K., Spielvogel, J.,** 2006. Web-Based Intelligent Tutoring Systems in K-12 Settings, Proceedings of the 5th IEEE/ACIS International Conference on Computer and Information Science and 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse.
- [37] **Schwartz, A., R., Ilani, Z.,** 2006. An Intelligent Tutoring System for Efficient Usage of Databases, International Conference on Information Technology: Research and Education.

- [38] **Tucho, R., Sierra, J. M., Fernandez, J. E., Vijande, R. and Moris, G.,** 2003. Expert Tutoring system for teaching mechanical engineering, Expert System with Applications, 24, 415-424.
- [39] **El-Khouly, M. M., Far, B. H. and Koono, Z.,** 2000. Expert tutoring system for teaching computer programming languages, Expert System with Applications, 18, 27-32.
- [40] **Sağıroğlu, Ş., Çolak, İ. ve Kahraman, H. T.,** 2008. Geleneksel Web Tabanlı Öğretim Sistemlerinden Uyarlanır Öğretim Sistemine Geçiş: Uhös İçin Tasarım Yaklaşımlarının İncelenmesi, Gazi Üniv. Müh. Mim. Fak. Der., Cilt 23, No 4, 837-852.
- [41] **Doğan, A.,** 2002. Yapay Zeka, Kariyer Yayıncılık, İstanbul.
- [42] **Allahverdi, N.,** 2002. Uzman Sistemler - Bir Yapay Zeka Uygulaması, Atlas Yayınları No:12, İstanbul.
- [43] **Öztemel, E.,** 2003. Yapay sinir Ağları, Papatya Yayıncılık, İstanbul.
- [44] **Nabiyev, V. V.,** 2005. Yapay Zeka, Seçkin Yayıncılık, Ankara.
- [45] **Winstanley, G.,** 1991. Artificial Intelligence in Engineering, New York.
- [46] **Dubois, D. and Prade, H.,** 1980. Fuzzy Sets and Systems: Theory and Applications, Academic Press, NY.
- [47] **Elmas, Ç.,** 2003. Yapay sinir Ağları, Seçkin Yayıncılık, Ankara.
- [48] **Moundridou, M., Virvou, M.,** 2003. Analysis and design of a Web-based authoring tool generating intelligent tutoring systems, Computers & Education, 40, 157-181.
- [49] **Yong, Z., Zhijing, L.,** 2003. A Model of Web Oriented Intelligent Tutoring System for Distance Education, Proceedings of the Fifth International Conference on Computational Intelligence and Multimedia Applications.
- [50] **Hatzilygeroudis, I., Prentzas, J.,** 2004. Using a hybrid rule-based approach in developing an intelligent tutoring system with knowledge acquisition and update capabilities, Expert System with Applications, 26, 477-492.
- [51] **Castro, J. R., Castillo, O. and Martinez, L. G.,** 2007. Interval Type-2 Fuzzy Logic Toolbox, Engineering Letters, 15, 1.
- [52] **Karnik, N. N., Liang, Q., Mendel, J. M.,** 2001. Type-2 Fuzzy Logic Software. Available from: <<http://sipi.usc.edu/mendel/software/>>.

- [53] **Bulut., M.**, 2004. Tip-2 Bulanık Mantık Sistemlerin Benzetimi için Yazılım Geliştirme, *Yüksek Lisans Tezi*, Fırat Üniversitesi Fen Bilimleri Enstitüsü, Elazığ.
- [54] **Felder, R. M., Silverman, L. K.**, 1998. Learning and teaching styles in engineering education, *Engineering Education*, 78(7), 674-681.
- [55] <http://www.learning-styles-online.com> Discover your Learning Styles - Graphically!, 12 Ocak 2009.
- [56] McMillan, N. and Keller, C., 2009.
<http://www.accd.edu/sac/history/keller/ACCDitg/SSLS.html> Learning Styles, 12 Ocak 2009.
- [57] **Felder, R. M. and Brent, R.**, 2005. Understanding Student Differences. *J. Engr. Education*, 94(1), 57-72 .
- [58] **Litzinger, T. A., Lee, S. H., Wise, J. C. and Felder, R. M.**, 2007. A Psychometric Study of the Index of Learning Styles, *J. Engr. Education*, 96(4), 309-319 .
- [59] **Felder R. M. and Spurlin, J.E.**, 2005. Applications, Reliability, and Validity of the Index of Learning Styles, *Intl. Journal of Engineering Education*, 21(1), 103-112 (2005).
- [60] **Kapadia R. J.**, 2008. Teaching and Learning Styles in Engineering Education, 38th ASEE/IEEE Frontiers in Education Conference.
- [61] **Felder, R. M., Felder G.N., and Dietz, E.J.**, 2002. The Effects of Personality Type on Engineering Student Performance and Attitudes. *J. Engr. Education*, 91(1), 3-17.
- [62] **Felder, R. M.**, 1996. Matters of Style. *ASEE Prism*, 6(4), 18-23.
- [63] **Veznedaroğlu, R. L., Özgür, A. O.**, 2005. Öğrenme Stilleri: Tanımlamalar, Modeller ve İşlevleri, *İlköğretim-Online*, 4(2), 1-16.
- [64] Ersolmaz, B., T., www.mm.anadolu.edu.tr/./EtkinOgretme_Eskisehir_180108.ppt, 2008. Mühendislik Eğitiminde Etkin Öğretme, 12 Ocak 2009.
- [65] **Mendel, J. M.**, 2001. Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice-Hall, NJ, USA.
- [66] **Passino K. M. and Yurkovich, S.**, 1998, Fuzzy Control. Addison-Wesley, USA.
- [67] **Klir, G. J. and Folger, T. A.**, 1988. Fuzzy Sets, Uncertainty, and Information, Prentice Hall, Englewood Cliffs, NJ.

- [68] **Karnik, N. N., Mendel, J. M.**, 2001. Operations on type-2 fuzzy sets, *International Journal on Fuzzy Sets and Systems* 122 (2) 327–348.
- [69] **Dubois, D. and Prade, H.**, 1980, *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, NY.
- [70] **Sepulveda, R., Castillo, O., Melin, P., Rodriguez-Diaz, A., Montiel, O.**, 2007. Experimental study of intelligent controllers under uncertainty using type-1 and type-2 fuzzy logic *Information Sciences* 177, 2023–2048
- [71] **Mendel, M. J.**, 2007, Advances in type-2 fuzzy sets and systems, *Information Sciences* 177, 84–110
- [72] **Coupland, S., John, R. I.**, 2004, A new and efficient method for the type-2 meet operation, in: *Proceedings of IEEE FUZZ Conference*, Budapest, Hungary, July, 959–964.
- [73] **Ahmed, M., Muzaffar, Z.**, 2009. Handling imprecision and uncertainty in software development effort prediction: A type-2 fuzzy logic based framework *Information and Software Technology*, 51, 640–654.
- [74] **Walker, C., Elbert A.**, 2009. Walker Sets with type-2 operations *International Journal of Approximate Reasoning*, 50, 63–71.
- [75] **Walker, C., Walker, E.**, 2005. The algebra of fuzzy truth values, *Fuzzy Sets and Systems* 149, 309–347.
- [76] **Walker, C., Walker, E.**, 2002. Algebraic properties of type-2 fuzzy sets, in: Nguyen Hoang Phuong, Hung T. Nguyen, Nguyen Cat Ho, Pratit Santiprabhob (Eds.), *Proceedings of the Third International Conference on Intelligent Technologies and Third Vietnam–Japan Symposium on Fuzzy Systems and Applications*, Hanoi, Vietnam, December 3–5, , p. 9.
- [77] **Walker, C., Walker, E.**, 2003. Algebraic structures on fuzzy sets of type-2, in: Guoqing Chen et al. (Eds.), *Proceedings of the International Conference on Fuzzy Information Processing*, vol. 1, pp. 97–100.
- [78] **Yanga, M, Lin, D.**, 2009. On similarity and inclusion measures between type-2 fuzzy sets with an application to clustering *Computers and Mathematics with Applications* 57, 896-907.
- [79] **Mendel, J. M.**, 2004. Computing Derivatives in Interval Type-2 Fuzzy Logic Systems, *IEEE Transactions on Fuzzy Systems*, 12(1), 84–98.

- [80] **Wu, D., Mendel, J.M.**, 2007. Uncertainty measures for interval type-2 fuzzy sets Information Sciences 177, 5378–5393.
- [81] **Wu, D., Mendel, J.M.**, 2008. A vector similarity measure for linguistic approximation: Interval type-2 and type-1 fuzzy sets Information Sciences 178, 381–402.
- [82] **Lee, C. H., Hong, J. L., Lin, Y.C., Lai, W. Y.**, 2003. Type-2 fuzzy neural network systems and learning, International Journal of Computational Cognition 1 4, 79–90.
- [83] **John, R. I. and Mendel, J. M.**, 2002. Tip-2 Fuzzy Sets Made Simple, IEEE Trans. on Fuzzy Systems, 10, 2.
- [84] **Mizumoto, M. and Tanaka, K.**, 1976. Some Properties of Fuzzy Sets of Type-2, Information and Control, 31, 312-340.
- [85] **Karnik, N. N. and Mendel, J. M.**, May. 1998. Introduction to Type-2 Fuzzy Logic Systems, in Proc. 1998 IEEE FUZZ Conf., pp. 915-920, Anchorage, AK.
- [86] **Karnik, N. N.**, 1998. Type-2 Fuzzy Logic Systems, Ph.D. Dissertation, University of Southern California, Los Angeles, CA.
- [87] **Karnik, N. N., Liang, Q. and Mendel, J. M.**, 1999. Type-2 Fuzzy Logic Systems, IEEE Trans. Fuzzy Syst., 7, 643-658.
- [88] **Irmak, E.**, 2008. E-Öğrenme Ortamları İçin Matlab Web Sunucu Kullanımı, Gazi Üniv. Müh. Mim. Fak. Dergisi, Cilt 23, No 2, 495-506.
- [89] **Gökbulut, M.**, 2009, Kontrol Sistemleri Ders Notları, 12 Ocak 2009.
- [90] Nyack, C., 2006. Introduction to Control, <http://controlcan.homestead.com/files/idxpages.htm>, 12 Ocak 2009.
- [91] **Kuo, B.**, Automatic Control Systems, 2002. (Türkçe), Çeviren: Bir, A., Literatür Yayınları.
- [92] <http://www.acilveilkyardim.com/egitim/egitimogretim.htm>, 12 Ocak 2009.
- [93] <http://virtual.cvut.cz>, 15 Nisan 2010.
- [94] <http://yzgrafik.ege.edu.tr/~tekrei/dosyalar/sunum/Web3DStandartlar>, 15 Nisan 2010.
- [95] <http://www.imsglobal.org>, 12 Ocak 2009.
- [96] <http://ieeeltsc.org>, 12 Ocak 2009.

- [97] <http://www.adlnet.gov/Pages/Default.aspx>, 12 Ocak 2009.
- [98] <http://www.ariadne-eu.org>, 12 Ocak 2009.
- [99] **Xu, D., Wang, H., Su, K.**, 2002. Intelligent Student Profiling with Fuzzy Models, Proceedings of the 35th Hawaii International Conference on System Sciences.
- [100] **Chen, C. M., Duh, L. J.**, 2008. Personalized web-based tutoring system based on fuzzy item response theory, *Expert System with Applications*, 34, 4, 2298-2315.
- [101] **Mayo, M. J.**, 2001. Bayesian Student Modelling and Decision-Theoretic Selection of Tutorial Actions in Intelligent Tutoring Systems, University Of Canterbury Doctor Of Philosophy In Computer Science, 255p.
- [102] **Mitchell, T. M.**, 1997. "Machine Learning", McGraw-Hill, New York, USA, 231-236.
- [103] **Martin, B., I.**, 2001. Intelligent Tutoring Systems: The Practical Implementation Of Constraint-Based Modelling, University Of Canterbury, *Doctor Of Philosophy In Computer Science*.
- [104] **Baghaei, N.**, 2007. A Collaborative Constraint-Based Intelligent System For Learning Objectoriented Analysis And Design Using Uml, University Of Canterbury *Doctor Of Philosophy*, In Computer Science.
- [105] **Csizmadia, V.**, 2003. Constructing an Authoring Tool for Intelligent Tutoring Systems with Hierarchical Domain Models, Degree of Master of Science in Computer Science, Worcester Polytechnic Institute, 81p.
- [106] **Anohina, A.**, 2007. Advances in Intelligent Tutoring Systems: Problem-solving Modes and Model of Hints, *International Journal of Computers, Communications & Control*, Vol. II, No. 1, pp. 48-55.
- [107] **Chen, C.**, 2008. Intelligent web-based learning system with personalized learning path guidance, *Computers & Education*, 51, 2, 787-814.
- [108] **Canales, A., Pena, A., Peredo, R., Sossa, H., Gutierrez, A.**, 2007. Adaptive and intelligent web based education system: Towards an integral architecture and framework, *Expert Systems with Applications*, 33, 1076-1089.
- [109] **Sarrafzadeh, A Alexander, S Dadgostar, F., Fan, C., Bigdeli, A.**, 2008. "How do you know that I don't understand?" A look at the future of intelligent tutoring systems, *Computers in Human Behavior*, 24, 4, 1342-1363.
- [110] **Zhang, J., Ren, F., Chen, T.**, 2005. Intelligent Tutoring Systems: Research Status and Its Development in China, *Proceeding of NLP-KE'05*.

- [111] **Simic, an G., Devedzic, V.**, 2003. Building intelligent system using modern Internet Technologies, Expert System with Applications, 25, 231-246.
- [112] **Kodaganallur, V., Weitz, R., Rosenthal, D.**, 2006. Tools for Building Intelligent Tutoring Systems, Proceedings of the 35th Hawaii International Conference on System Sciences.
- [113] **Daş, R., Türkoğlu, İ., Poyraz, M.**, 2007. Web Kayıt Dosyalarından İlginç Örüntülerin Keşfedilmesi, Fırat Üniversitesi Fen ve Mühendislik Bilimleri Dergisi, 19, 4, 493-503.
- [114] **Daş, R., Türkoğlu, İ., Poyraz, M.**, 2008. Analyzing of the user access logs of a website using web usage mining method: Example of Fırat University, e-Journal of New World Sciences Academy (NWSA), Natural and Applied Sciences, 3, 2, 310-320.
- [115] **Daş, R., Türkoğlu, İ.** 2009. Creating meaningful data from web logs for improving the impressiveness of a website by using path analysis method, Expert Systems with Applications, 36, 3, 6635-6644.
- [116] **Şendağ, S.**, 2008. Web’de Yeni Eğilimler: Öğrenme Ortamlarına Entegrasyonu, 8. Uluslararası Eğitim Teknolojileri Konferansı, Mayıs 2008, Eskişehir.
- [117] **Dağ, F., Erkan, K.**, 2007. E-Öğrenme ve Anlamsal Web, Ulusal Teknik Eğitim, Mühendislik ve Eğitim Bilimleri Genç Araştırmacılar Sempozyumu (UMES 2007), 21-23 Haziran 2007, Kocaeli Üniversitesi, KOCAELİ, Cilt 1, 146-150.

ÖZGEÇMİŞ

1980 yılında Elazığ'da doğdu. İlk, orta ve lise öğrenimini Elazığ'da tamamladıktan sonra 1998 yılında Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik Bilgisayar Eğitimi Bilgisayar Öğretmenliği Bölümü'nü kazandı. 2002 yılında bu bölümden mezun olup Fırat Üniversitesi Elektronik Bilgisayar Eğitimi Kontrol Eğitimi Bilim Dalı'nda yüksek lisans eğitimine, 2004 yılında Elektrik Elektronik Mühendisliği Elektrik Makinaları Bilim Dalı'nda doktora başladı.

2002 yılından beri Elazığ İlköğretim okulunda bilgisayar öğretmeni olarak görev yapmaktadır. Evli ve bir çocuk annesidir.