

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

**BULANIK MANTIK MODEL GELİŞTİRME
ORTAMLARININ KARŞILAŞTIRILMASI
VE ÖRNEK BİR UYGULAMA**

YÜKSEK LİSANS TEZİ

Tek. Öğr. Barış TAÇYILDIZ

Enstitü Anabilim Dalı : ELEKTRONİK VE BİLGİSAYAR EĞİTİMİ

Tez Danışmanı : Doç. Dr. Ahmet ZENGİN

Mart 2013

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BULANIK MANTIK MODEL GELİŞTİRME
ORTAMLARININ KARŞILAŞTIRILMASI
VE ÖRNEK BİR UYGULAMA

YÜKSEK LİSANS TEZİ

Tek. Öğr. Barış TAÇYILDIZ

Enstitü Anabilim Dalı : ELEKTRONİK VE BİLGİSAYAR EĞİTİMİ

Bu tez 30 / 01 / 2013 tarihinde aşağıdaki jüri tarafından Oybirliği ile kabul edilmiştir.

Prof.Dr. Abdullah Feri Koştu
.....
Jüri Başkanı

Doc.Dr. Ahmet Zengin
.....
Üye

Doç.Dr. Dr. Bülent Alobanğlı
.....
Üye

TEŐEKKÜR

Yüksek lisans tezimi hazırlarken, çok değerli fikirleri ve yol göstericiliđi ile bana büyük katkılar sađlayan, çalışmalarımnda desteklerini esirgemeyen, tez çalışmasının her aşamasında bilgisinden istifade ettiđim çok saygıdeđer hocam Doç. Dr. Ahmet ZENGİN' e teşekkürü bir borç bilirim

Çalışmalarım sırasında bana çok önemli destek, bilgi ve yardımda bulunan, çok değerli arkadaşım Yrd. Doç. Dr. Fatih ÇELİK' e ve aileme sonsuz teşekkür ederim.

İÇİNDEKİLER

TEŞEKKÜR.....	ii
İÇİNDEKİLER.....	iii
SİMGELER VE KISALTMALAR LİSTESİ.....	vii
ŞEKİLLER LİSTESİ.....	viii
TABLolar LİSTESİ.....	xi
ÖZET.....	xii
SUMMARY.....	xiii
BÖLÜM 1. GİRİŞ.....	1
1.1. Çalışmanın Amacı.....	2
1.2. Literatür Taraması.....	3
1.3. Çalışmanın Bilime Katkısı.....	6
1.4. Kullanılan Yöntem ve Teknikler.....	8
BÖLÜM 2. BULANIK SİSTEMLER VE EĞİTİMİ.....	10
2.1. Bulanık Mantık.....	10
2.2. Bulanık Kümeler ve Üyelik Fonksiyonları.....	12
2.3. Bulanık Mantık Eğitimi.....	16
2.3.1. Örnek bir bulanık mantık dersi ders içeriği.....	17
2.4. Bulanık Mantık Simülatörleri.....	19
BÖLÜM 3. BULANIK MANTIK MODEL GELİŞTİRME ORTAMLARI... ..	21
3.1. FuzzyTECH Model Geliştirme Ortamı.....	21
3.1.1. FuzzyTECH Programının Genel Görünümü.....	22
3.1.2 FuzzyTech Programının Özellikleri.....	24
3.1.3 FuzzyTECH Bulanık Mantık Tasarım Sihirbazı.....	29

3.1.4. FuzzyTECH ile Örnek Bir Uygulama:Kamyon Hız Kontrolü	30
3.1.5. FuzzyTECH Sürümleri.....	33
3.2. LabVIEW Bulanık Mantık Model Geliştirme Ortamı.....	33
3.2.1. LabVIEW PID-Fuzzy Logic Araç Kutusu.....	34
3.2.2. LabVIEW programı ile örnek bir uygulama: Sera Kontrolü...	37
3.2.3. LabVIEW programının diğer özellikleri.....	40
3.3. MATLAB FIS Bulanık Mantık Model Geliştirme Ortamı.....	42
3.3.1. Matlab FIS (Bulanık Arayüz Sistemi) editörleri.....	43
3.3.1.1. Bulanık çıkarım sistemi (FIS) editörü.....	44
3.3.1.2. Üyelik fonksiyon editörü.....	46
3.3.1.3. Kural editörü.....	46
3.3.1.4. Kural görüntüleyici.....	47
3.3.1.5. Yüzey görüntüleyici.....	47
3.3.2. Matlab ile örnek bir uygulama: Model Uçak Hız Kontrolü....	48
3.3.3 Matlab FLT' da Bulunan Üyelik Fonksiyon Tipleri.....	51
3.3.4 MATLAB programının sürümleri.....	53

BÖLÜM 4.

BİLGİSAYAR SİSTEMLERİNİN SOĞUTULMASI İÇİN BULANIK MANTIK SOĞUTUCU TASARIMI.....	54
4.1. Sunucu Bilgisayarlar.....	54
4.2. Sunucu Bilgisayar Sistemlerinin Soğutulması İşlemi.....	55
4.3. Doğru Soğutmanın Önemi.....	56
4.4. Bulanık Mantıkla Çalışan Soğutucu Tasarımı.....	57
4.5. Tasarlanan Sistemin FuzzyTECH Programı ile Simülasyonu.....	58
4.5.1. Ortam sıcaklığı giriş değişkeni.....	58
4.5.2. Hava akışı giriş değişkeni.....	59
4.5.3. İşlemci kullanımı giriş değişkeni.....	59
4.5.4. Sunucu yoğunluk değişkeni.....	60
4.5.5. Bulanık fan çıkış sıcaklık değişkeni.....	61
4.5.6. Kural tabanı oluşturma.....	61
4.5.7. Simülasyon sonuçlarını görüntüleme.....	62
4.6. Tasarlanan Sistemin Labview Programı ile Simülasyonu.....	63

4.6.1. Değişkenlerin Labview’ da tasarlanması.....	63
4.6.2. Simülasyon sonuçlarını görüntüleme.....	65
4.7. Tasarlanan Sistemin Matlab Programı ile Simülasyonu.....	67
4.7.1. Değişkenlerin Matlab FIS Editor’de tasarlanması.....	67
4.7.2. Simülasyon sonuçlarını görüntüleme.....	70
BÖLÜM 5.	
BULANIK MANTIK MODEL GELİŞTİRME ORTAMLARI VE SİMÜLASYON SONUÇLARININ KARŞILAŞTIRILMASI.....	72
5.1. Kurulum Kriterine Göre Programların Karşılaştırılması.....	72
5.2. Kullanım Kolaylığı Kriterine Göre Programların Karşılaştırılması.	73
5.3 Erişilebilirlik Kriterine Göre Programların Karşılaştırılması.....	75
5.4 Görşellik Kriterine Göre Programların Karşılaştırılması.....	75
5.5 Performans Kriterine Göre Programların Karşılaştırılması.....	76
5.6. Sistem Kaynakları Kriterine Göre Programların Karşılaştırılması...	78
5.7. Destek Kriterine Göre Programların Karşılaştırılması.....	79
5.8. Eğitim Ortamlarında Kullanım Durumu Kriterine Göre Programların Karşılaştırılması.....	81
5.9. Dokümantasyon Kriterine Göre Programların Karşılaştırılması.....	82
5.10. Nesneye Yönelik Olma Durumu Kriterine Göre Programların Karşılaştırılması.....	83
5.11 Model Kütüphanelerinin Genişliği Kriterine Göre Programların Karşılaştırılması.....	84
5.12 Sonuçların analiz edilme kolaylığı Kriterine Göre Programların Karşılaştırılması.....	85
5.13. Genişletilebilirlik Kriterine Göre Programların Karşılaştırılması..	87
5.14 GUI oluşturma kriterine göre programların karşılaştırılması.....	88
5.15 Yeniden Kullanılabilirlik Kriterine Göre Programların Karşılaştırılması.....	89
BÖLÜM 6.	
SONUÇLAR VE DEĞERLENDİRME.....	90

KAYNAKLAR.....	93
EKLER.....	98
ÖZGEÇMİŞ.....	99

SİMGELER VE KISALTMALAR LİSTESİ

PID	: Oransal Integral Türevsel Denetleyici
TAI	: Türk Havacılık ve Uzay Sanayii A.Ş.
PIC	: Çevrebirim Arayüz Denetleyicisi
BUS	: Bulanik Uzman Sistemi
FLC	: Bulanik Mantık Kontrolörü
DC	: Doğru Akım
SAÜSEM	: Sakarya Üniversitesi Sürekli Eğitim Merkezi
YSA	: Yapay Sinir Ağları
MCU	: Mikrokontrolör Ünitesi
MS	: Microsoft
IEC	: Uluslararası Elektroteknik Komisyonu
mph	: mil/saat
VI	: LabVIEW sanal cihazları
GUI	: Grafiksel Kullanıcı Arayüzü
FIS	: Bulanık Çıkarım Sistemi
Σ	: Kartezyen Toplam
$\mu_A(x)$: Üyelik Fonksiyonu
α	: Kesim Katsayısı

ŞEKİLLER LİSTESİ

Şekil 2.1.	Sayıların komşuluğunun analitik düzlemde gösterimi.....	13
Şekil 2.2.	$A = (-5,-1,1)$ kümesinin komşuluğu.....	15
Şekil 2.3.	$A = (0,1,1,0)$ kümesinin komşuluğu.....	16
Şekil 3.1.	FuzzyTECH programının genel ekran görüntüsü.....	25
Şekil 3.2.	FuzzyTECH programında dilsel değişkenlerin ifade edilmesi.....	25
Şekil 3.3.	FuzzyTECH programında kural editör penceresi.....	26
Şekil 3.4.	FuzzyTECH programında 3 boyutlu çizim ekranı.....	27
Şekil 3.5.	FuzzyTECH programıyla yapılmış bir simülasyon çalışması.....	27
Şekil 3.6.	FuzzyTECH programı raporlama ekranı.....	28
Şekil 3.7.	FuzzyTECH bulanık mantık sihirbazı görünümü.....	29
Şekil 3.8.	Tasarlanan sistemin mekanik modeli.....	30
Şekil 3.9.	Tasarlanan sistemin Proje Editörü görünümü.....	31
Şekil 3.10.	Tasarlanan sistemin simülasyon anındaki görüntüsü.....	32
Şekil 3.11.	Sistemin gerçek dünyaya uygulanmasının şematik gösterimi.....	32
Şekil 3.12.	LabVIEW dilsel değişken penceresi.....	34
Şekil 3.13.	LabVIEW terim düzenleme penceresi.....	35
Şekil 3.14.	LabVIEW kural penceresi.....	35
Şekil 3.15.	Sistemin test anındaki görüntüsü.....	36
Şekil 3.16.	Sistemin sıcaklık giriş ve elektronik çatı çıkış değişkenlerinin belirlenmesi.....	37
Şekil 3.17.	Sistemin nem giriş ve sulama çıkış değişkenlerinin belirlenmesi..	38
Şekil 3.18.	Sistemin kural penceresindeki görünümü.....	39
Şekil 3.19.	Sistemin grafik kullanıcı arayüzü.....	40
Şekil 3.20.	LabVIEW kontrol dizayn ve simülasyon araçları.....	40
Şekil 3.21.	LabVIEW bulanık mantık araçları.....	41
Şekil 3.22.	MATLAB bulanık mantık blok yapısı.....	44

Şekil 3.23.	MATLAB kural editör penceresi.....	45
Şekil 3.24.	MATLAB FIS editör penceresi.....	45
Şekil 3.25.	MATLAB üyelik fonksiyon editörü penceresi.....	46
Şekil 3.26.	MATLAB kural editörü penceresi.....	46
Şekil 3.27.	MATLAB kural görüntüleyici penceresi.....	47
Şekil 3.28.	MATLAB yüzey görüntüleyici penceresi.....	47
Şekil 3.29.	Tasarlanan sistemin MATLAB üyelik fonksiyon editör penceresi	48
Şekil 3.30.	Tasarlanan sistemin giriş ve çıkış değişkenleri.....	49
Şekil 3.31.	Tasarlanan sistemin kuralları.....	49
Şekil 3.32.	Tasarlanan sistemin çalıştırılması ve çalışma anındaki görüntüsü.	50
Şekil 3.33.	MATLAB programında üçgensel üyelik fonksiyonları.....	51
Şekil 3.34.	MATLAB programında Gauss üyelik fonksiyonları.....	52
Şekil 3.35.	MATLAB programında sigmoidal üyelik fonksiyonları.....	52
Şekil 3.36.	MATLAB programında polinomal üyelik fonksiyonları.....	52
Şekil 4.1.	FuzzyTECH programında ortam sıcaklığı giriş değişkeninin tanımlanması.....	59
Şekil 4.2.	FuzzyTECH programında hava döngüsü giriş değişkeninin tanımlanması.....	59
Şekil 4.3.	FuzzyTECH programında CPU kullanımı giriş değişkeninin tanımlanması.....	60
Şekil 4.4.	FuzzyTECH programında sunucu yoğunluğu giriş değişkeninin tanımlanması.....	60
Şekil 4.5.	FuzzyTECH programında bulanık fan çıkış değişkeninin tanımlanması.....	61
Şekil 4.6.	FuzzyTECH programında kural tabanı oluşturulması.....	62
Şekil 4.7.	FuzzyTECH programında sistemin ısıt için ayarlanması.....	62
Şekil 4.8.	FuzzyTECH programında sistemin çok soğut için ayarlanması...	63
Şekil 4.9.	FuzzyTECH ile tasarlanan sistemin 3 boyutlu görünümü.....	63
Şekil 4.10.	LabVIEW programında CPU kullanımı giriş ve çıkış değişkenlerinin tasarlanması.....	64
Şekil 4.11.	LabVIEW programında ortam sıcaklığı giriş değişkeninin tasarlanması.....	64
Şekil 4.12.	LabVIEW programında sunucu yoğunluğu giriş değişkeninin	

	tasarlanması.....	64
Şekil 4.13.	LabVIEW 'da hava döngüsü giriş değişkeninin tasarlanması.....	65
Şekil 4.14.	LabVIEW programında oluşturulan kural tabanı.....	65
Şekil 4.15.	LabVIEW programında birinci simülasyon sonuçları.....	66
Şekil 4.16.	LabVIEW programında ikinci simülasyon sonuçları.....	66
Şekil 4.17.	MATLAB programında CPU kullanımı giriş değişkeninin tanımlanması.....	67
Şekil 4.18.	MATLAB programında ortam sıcaklığı giriş değişkeninin tanımlanması.....	68
Şekil 4.19.	MATLAB programında sunucu yoğunluğu giriş değişkeninin tanımlanması.....	68
Şekil 4.20.	MATLAB 'da hava döngüsü giriş değişkeninin tanımlanması.....	69
Şekil 4.21.	MATLAB programında kural tablosunun tasarlanması.....	69
Şekil 4.22.	MATLAB programında birinci simülasyon sonuçları.....	70
Şekil 4.23.	MATLAB programında ikinci simülasyon sonuçları.....	70
Şekil 4.24.	MATLAB programında sistemin yüzey görünümü.....	71
Şekil 5.1.	İncelenen programların görsellikleri.....	75
Şekil 5.2.	MATLAB performans testi çubuk grafiği.....	77
Şekil 5.3.	LabVIEW performans sonuç grafiği.....	77
Şekil 5.4.	İşlem- geçen süre kriterine göre performans sonuç grafiği.....	78
Şekil 5.5.	Sistem kaynaklarının kullanımı.....	78
Şekil 5.6.	Programların aynı anda çalışması sırasında sistem durumu.....	79
Şekil 5.7.	MATLAB analiz ekranı.....	86
Şekil 5.8.	LabVIEW analiz ekranı.....	86
Şekil 5.9.	MATLAB kullanılarak hazırlanmış GUI.....	88
Şekil 5.10.	LabVIEW kullanılarak hazırlanmış GUI.....	88

TABLULAR LİSTESİ

Tablo 3.1.	Değişik ortamlar için FuzzyTECH programının verdiği cevaplar.	23
Tablo 3.2.	FuzzyTECH programının sürümleri.....	33
Tablo 3.3.	LabVIEW programının sürümleri.....	42
Tablo 3.4.	MATLAB programının sürümleri.....	53
Tablo 4.1.	İncelenen sunucunun özellikleri.....	55
Tablo 4.2.	Giriş ve çıkış değişkenleri.....	57
Tablo 6.1.	İncelen programların değişik kriterlere göre durumu.....	91

ÖZET

Anahtar kelimeler: Bulanık Mantık Eğitimi, FuzzyTECH, LabVIEW, MATLAB, Soğutma Sistemi

Bulanık mantık, düşünen makinelere doğru bir çabanın olduğu dünyamızda gittikçe değeri artan bir düşünme biçimidir. İnsanların günlük hayatta kullandıkları çok yönlü düşünme ve karar alma mekanizmalarını makinelere uyarlamak üzerine yapılan bulanık mantık çalışmaları gün geçtikçe artmaktadır. Eğitim ortamlarında da bulanık mantık üzerine verilen ders ve yapılan çalışmalara ilginin arttığı görülmektedir.

Bu tez çalışmasında bulanık mantık eğitiminin ülkemizdeki durumu incelenmiştir. Bulanık mantık eğitimi verilirken kullanılan programlar, karşılaşılan sorunlar ve çözüm yolları üzerinde durulmuştur.

Ülkemizde yaygın kullanılan bulanık mantık simülasyon araçları 15 kritere göre karşılaştırılmış ve değerlendirme yapılarak sonuçlar elde edilmiştir. Elde edilen sonuçlar ışığında değişik ortamlar için hangi bulanık mantık aracının kullanılması gerektiği ile ilgili sonuçlar elde edilmiştir.

COMPARISON OF THE FUZZY LOGIC MODEL DEVELOPMENT ENVIRONMENTS AND A CASE STUDY

SUMMARY

Key Words: Fuzzy Logic Education, FuzzyTECH, LabVIEW, MATLAB, Cooling System

Fuzzy Logic is an effort towards thinking machines so it has an increasing value of our world. Today, researches on fuzzy logic are increasing based on efforts to inject human's intelligence and mindset to machines. Fuzzy Logic technology helps on adapting decision-making mechanisms to the self-organizing systems. On the other hand, educational activities including Fuzzy Logic are becoming important day by day.

In this thesis, state of the fuzzy logic discipline is investigated. Software, challenges and solutions are considered. Fuzzy logic simulation tools that are widely used in our country are compared with fifteen criteria and results are presented. Finally, the suggestions are made for the educational environments about which program may be used.

BÖLÜM 1. GİRİŞ

Bulanık Mantık insanların her gün kullandığı ve davranışlarının yorumlandığı yapıya ulaşılmasını sağlayan matematiksel bir disiplindir. İnsanlar günlük hayatta; tam olarak tanımlanmamış ve nümerik olmayan dilsel niteleyiciler (soğuk, biraz soğuk, ılık, sıcak, çok sıcak vb. gibi) kullanarak kararlar verir ve problemlerini çözerler [1].

Bulanık mantık kullanıldığı her alanda klasik mantığa göre daha hassas sonuçlar verebilen bir kontrol yöntemidir. Bulanık mantıkta “kesin ve sabit” gibi kavramlar yerini üyelik dereceleriyle ifade edilen doğruluk derecelerine bırakır. Bu doğruluk dereceleri tamamen doğru ve tamamen yanlış değerleri arasında yer alır. Belli bir derecenin üstü kesin doğru ya da belli bir derecenin altı kesinlikle yanlıştır denilmez. Bulanık mantık kullanılarak tasarlanan denetleyicilerde günlük yaşamda kullanılan dilsel değişkenler kullanılması modelleme işleminde kolaylık sağlar [2].

Bulanık mantık denetleyici tasarımı yapılırken giriş ve çıkış değişkenleri arasında dilsel olarak ifade edilen bir kural tabanı oluşturulması gerekir. Yani 1 ve 0’lardan oluşan sayısal dünyayı, dilsel değişkenlerle ifade edilen bir forma dönüştürmek gereklidir. Bunu yapabilmek için de öncelikle programlama dilleri ya da paket programlar kullanılarak bilgisayarda simülasyonun gerçekleştirilmesi gereklidir [3].

Bulanık küme teorisinin, üyelikten üye olmamaya dereceli geçişi ifade etmesindeki yeteneği, belirsizlikleri tanımlamada önemli bir yere sahiptir. Bulanık küme teorisi; belirsizliğin ölçülmesinde güçlü ve anlamlı araçlar sunmasına ek olarak, dilimizde ifade edilen belirsiz kavramların anlamlı bir biçimde temsilini sağlar. Ancak Aristo mantığı üzerinde temellenen klasik küme teorisi, verilen bir alana ait bütün bireyleri incelenen özelliğe göre, kümeye ait olan elemanlar ve ait olmayanlar olmak üzere ele almaktadır. Kümeye üye ve üye olmayan elemanlar arasında kesin ve belirsiz olmayan bir ayırım vardır. Dilimizde ifade edilen ve üzerinde çalışılan çoğu

sınıflandırma kavramı, bu türde bir karakter de değildir. Örneğin; uzun boylu insanlar kümesi, pahalı otomobiller kümesi, yakın sürüş mesafesi, güvenilir kar araçları, birden çok büyük sayıların oluşturduğu küme gibi kavramlar klasik küme teorisinin öngördüğü şekilde incelenemezler [2].

Bulanık mantık (Fuzzy Logic) kavramı ilk kez 1965 yılında California Berkeley Üniversitesinden Prof. Lutfi A. Zadeh'in bu konu üzerinde ilk makalelerini yayınlamasıyla duyuldu. O tarihten sonra önemi gittikçe artarak günümüze kadar gelen bulanık mantık, belirsizliklerin anlatımı ve belirsizliklerle çalışılabilmesi için kurulmuş katı bir matematik düzen olarak tanımlanabilir [4]. Bilindiği gibi istatistik ve olasılık kuramında, belirsizliklerle değil kesin ifadelerle çalışılır ancak insanın yaşadığı ortam daha çok belirsizliklerle doludur. Bu yüzden insanoğlunun sonuç çıkarabilme yeteneğini anlayabilmek için belirsizliklerle çalışmak gereklidir [5].

1.1. Çalışmanın Amacı

Bulanık mantık kullanılarak sistem tasarlamak için yapılması gereken ilk işlem tasarlanacak sistemin standart bir işlemcisi olan bir bilgisayar üzerinde yüklenen yazılım tarafından kullanıma hazır hale getirilmesidir. Bu amaçla bulanık mantık tasarımlarında programlama dili olarak C# ve paket program olarak da FuzzyTECH ve MATLAB Fuzzy Logic Toolbox sıklıkla kullanılmaktadır [6].

Bu tez çalışmasında amaç örnek bir sunucu odası ortamı için istenen optimum şartların; FuzzyTECH, MATLAB Fuzzy Logic Toolbox ve ülkemizde sıklıkla kullanılmamasına rağmen üstün özellikleri olan LabVIEW PID and Fuzzy Logic Toolkit programları ile tasarlanan bulanık uzman sistem tarafından değerlendirilerek, gerektiğinde sunucu odası klima sisteminin çıkış değerlerini belirlemek ve bu üç programı çeşitli yönleriyle karşılaştırmaktır. Bunun için sıcaklık, nem ve sunucu yoğunluğu giriş değişkenleri olarak alınmış ve bu veriler, üç programda tasarlanan bulanık uzman sistem tarafından değerlendirilerek klimaların soğutma veya ısıtma yapmak üzere sistem tarafından devreye sokulacağı bir sunucu odası klima sistemi simülasyonu yapılması amaçlanmıştır.

Aynı zamanda FuzzyTECH, Matlab ve Labview programları tüm yönleriyle tanıtılmış ve bu programlar kullanılarak bulanık sistemlerin nasıl tasarlanacağı, kural veri tabanı oluşturma, grafiksel arayüzleri ve sonuç değişkenleri ortak ve farklı yönleriyle karşılaştırılmıştır.

1.2. Literatür Taraması

Yapılan literatür taramasında bulanık mantıkla ilgili çok sayıda çalışma bulunmuştur. Buna rağmen bulanık mantık kullanılarak klima sistemi kontrolü ile ilgili sadece üç adet çalışma bulunmuştur. Kontrol sistemi tasarım ve simülasyonunda bazı çalışmalarda FuzzyTECH programı tek başına kullanılmış, bazılarında ise Matlab Fuzzy Toolbox kullanıldığı görülmüştür. Ancak yapılan literatür taramasında Labview programı ile bulanık mantık sistem tasarımına rastlanmamıştır. Aynı zamanda bu üç programın karşılaştırmasını içeren bir çalışma da bulunamamıştır.

Bulanık mantık konusunda yapılan araştırmaların Japonya'da oldukça fazla olduğu görülmektedir. Özellikle Bulanık İşlem Kontrolörü (FPS) olarak isimlendirilen özel amaçlı bulanık mantık mikroişlemci çipinin üretilmesine çalışılmaktadır. Bu teknoloji fotoğraf makineleri, çamaşır makineleri, klimalar ve otomatik iletim hatları gibi uygulamalarda kullanılmaktadır. Bundan başka uzay araştırmaları ve havacılık endüstrisinde de kullanılmaktadır. TAI (Türk Havacılık ve Uzay Sanayii A.Ş.)'de araştırma gelişme kısmında bulanık mantık konusunda çalışmalar yapılmaktadır [7].

Bulanık mantık ile klima sistemi kontrolü üzerine yapılan araştırmalarda ErKaymaz H. ve Çayroğlu İ. (2010) tarafından yazılan “Bulanık mantık ve PIC kullanılarak bir klima sisteminin kontrolü” başlıklı bir çalışma bulunmaktadır. Bu çalışmalarında, bulanık mantık sistemi ile PIC programlanarak, bir klima sisteminin kontrolü gerçekleştirilmiştir. Sistem bulunduğu ortamın sıcaklığını 19 – 23 °C arasında tutmaktadır [8]. Giriş değişkenleri olarak nem ve sıcaklık değerleri dış ortamdan SHT11 sensörü ile alınmakta ve bulanık mantık sistemi ile programlanmış olan PIC 16F876'ya aktarılmaktadır. Çıktı değerlerine bağlı olarak ısıtıcı yada soğutucu fanlar gerekli iklimlendirmeyi sağlayacak şekilde çalıştırılmaktadır. Yaptıkları çalışmada MATLAB programını kullanmışlardır.

Özerk A. ve Sinecen M. (2004), klima sistem kontrolünün bulanık mantık ile modellenmesi üzerine bir çalışma yapmışlardır [3]. Bu çalışmada, matematiksel modelinin oluşturulmasının zor ve karmaşık olduğu klima sisteminin bulanık mantık ile kontrolü incelenmiştir. Yapılan kontrolün sonuçlarının incelenmesi için Fuzzy Logic Toolbox simülasyon paket programı kullanılmıştır. Literatür, bulanık mantığın, kullanıldığı uygulamaların birçoğunda önemli bir yöntem olduğu fikrinde birleşir. Ayrıca, klasik mantık ile oluşturulan karmaşık kontrol sistemlerini daha basite indirgeyerek daha ekonomik bir yöntem olduğunu göstermiştir.

Soy H. (2006) yaptığı tez çalışmasında ev otomasyon sistemi bulanık kontrolör kullanılarak denetlemeye çalışmıştır [10]. Belirtilen amaca erişmek için uygun bulanık kurallar, üyelik fonksiyonları, bulanıklaştırma ve durulaştırma işlemlerini yapan birimler belirlemiş ve sonunda bütün bunlar gömülü kontrolör üzerindeki yazılımla gerçekleştirmiştir. Proje ev içinde konforlu iklim şartları sağlamak için sıcaklık kontrol sisteminin tipik bir uygulamasıdır. Kontrolörün çıkışları bina içi sıcaklık, bina dışı sıcaklık ve nispi nem üzerine kurulmuştur. Sistem mikro kontrolör içine gömülü bulunmakla birlikte sensörler, kompresör ve hava devir daim fanı arasında arabirim görevi üstlenir. Mikroçip PIC18F452 çok sayıda güçlü özelliğe sahip olması sebebiyle gömülü kontrolör olarak tahsis edilmiştir. Mikro kontrolör için kod C programlama dilinde yazılmıştır. Sistemin çalışmasını basitçe göstermek için kullanıcı dostu bir arabirim tasarlanmıştır.

Etik N. (2007) tarafından hazırlanan tez çalışmasında, ameliyathane için gerekli ortam şartlarının sağlanması için bulanık uzman sistem (BUS) kontrollü bir tasarım yapılmıştır. Hedeflenen amaç için gerçek ameliyathaneler incelenmiş, bu ameliyathanelerde uygulanan sistemler araştırılarak daha verimli, konforlu ve güvenilir bir sistem olup olmadığı araştırılmıştır. Ayrıca BUS ile bir ameliyathanenin nasıl kontrol edilebileceği, avantaj ve dezavantajları araştırılmıştır. Yukarıda sayılan avantajları sağlayabileceği teorik olarak görülen BUS için bir ameliyathane prototipi yapılarak uygun konfigürasyon tasarlanmıştır. Bu tasarım için ortam ısısı, partikül miktarı, nem değeri ve oksijen miktarı giriş parametreleri olarak; temiz hava girişi ve egzoz için fan devirleri çıkış parametresi olarak seçilmiştir. Bir uzman yardımıyla uygun dilsel ifadeler ve bu ifadelerin üyelik fonksiyonları belirlenmiştir. Sistemde

kullanılması gereken sensörler gruplandırılmış, bu sensör bilgilerinin tasarlanan bir arabirim ile bilgisayara aktarımı sağlanmıştır. Sistem için C# ile bir ara yüz yazılarak veriler anlık olarak sisteme aktarılmıştır. Sistem prototip için en uygun kontrolü sağlayıp sağlamadığı ameliyat simüle edilerek kontrol edilmiştir. Bu denemelerde sensörlerden alınan giriş parametre değerleri ve BUS'ın hesapladığı çıkış parametre değerleri bilgisayara kayıt edilip, verilerin geriye dönük olarak kontrol edilebilmesi sağlanmıştır. Sonuç olarak çalışmada, BUS ile yapılan ameliyathane prototip kontrolünde iyi sonuçların alındığı gözlemlenmiştir. Yapılan analizler sonucunda BUS ile yapılan kontrolün daha ekonomik, konforlu, güvenilir ve kararlı bir kontrol sağladığı ve bunun gerçek ameliyathanelerde uygulanabilirliği görülmüştür [9].

FuzzyTECH programının kullanıldığı bir çalışmada ise Karadeniz S. (2007) bulanık kontrolün temeli, gelişimi, bulanık mantık kontrolünün genel yapısı üzerinde durarak, çoklu kompresör sistemlerinde klasik PID kontrol algoritmasından çok farklı olarak bulanık mantık kontrol uygulama tasarım çalışmasını gerçekleştirmiştir. Dizayn için gerekli olan bulanık mantık temelli yazılım geliştirme araçlarından FuzzyTECH ve C programlarını öğrenerek sistem tasarımını gerçekleştirmiştir. Bu çalışmada 4 adet çoklu kompresör sisteminin bulanık mantık temelinde basınç kontrolünün nasıl yapılacağı oluşturulan bir simülatör sistemi üzerinde gösterilmiştir [11].

MATLAB ve FuzzyTECH programlarının birlikte kullanıldığı bir çalışmaya ise Bolat Y. (2006) Matlab-SIMULINK + PIC tabanlı bulanık mantık denetleyici tasarımı ve gerçek zamanlı sıcaklık kontrolü uygulaması isimli bir çalışmada rastlanmaktadır. Bu çalışmada FuzzyTECH programı, bulanık kazanç ayarlı PI denetleyicinin PIC mikro denetleyici yazılımına gömülebilmesinde C kodu üretmek için kullanılmış ve üretilen C kodu üzerinde gerekli düzenlemeler yapılmıştır. Daha sonra da Matlab-SIMULINK kullanılarak simülasyon gerçekleştirilmiştir [12].

LabVIEW ile bulanık mantık tasarımına Thepsatorn, P. (2006) "DC Motor Speed Control using Fuzzy Logic based on LabVIEW" isimli çalışmasında karşılaşılmıştır. Bu çalışmasında Thepsatorn, LabVIEW (Laboratory Virtual Instrument Engineering Workbench) programına dayalı Bulanık Mantık Kontrolör (FLC) kullanarak

uyarımli DC motor hız kontrolü uygulaması yapmıştır. Bu çalışmasında kullanıcı dostu arayüzü olduğunu ifade ettiği Labview programının DC motor uygulamasını hazırlarken ne kadar uygun olduğunu göstermiştir [13].

N. A. Muhamad, ve S.A.M. Ali (2005) yaptıkları çalışmalarının amacı transformatör arızalarının tespiti ve durumunun izlenmesi için bulanık mantık kontrolör ile Labview programını kullanmaktır. Çalışmalarında kural veritabanı oluşturmuş ve simülasyonunu doğru bir şekilde gerçekleştirmişlerdir [14].

Sunucu odalarının iklimlendirilmesiyle ilgili bir çalışma ise Akın G. ve Ketenci S. tarafından “Sistem Odaları İklimlendirme Sistemleri” ismiyle 2010 tarihinde gerçekleştirilmiştir. Bu çalışmalarında sistem odalarında en ideal sıcaklık ve nem değerlerinin hangi değerler arasında olması gerektiğinden, sistem odası iklimlendirme tiplerinden ve monitör yazılımlarından bahsetmişlerdir. Sonuç olarak; kabin tipi sunucu sistemlerinin artması ve daha küçük alanlara daha çok sunucu sığdırılması, aynı alanda daha çok ısının açığa çıkmasına sebep olduğu görülmüştür. Ayrıca sunucu bilgisayarların yoğun bir şekilde çalışması sonucu bant genişliklerinin ve işlemci hızlarının artması bu ısının daha da artmasına sebep olmaktadır. Klasik sistem olarak adlandırılabilir sistemler halen daha yaygın olarak kullanılmasına rağmen uzun vadede enerji anlamında verimsiz kalmaktadırlar. Yeni kurulacak sistem odalarında diğer metotlarında incelenmesi, eğer belirli bir hacmin üstünde bir merkez olacaksa analizin yapılarak, tasarımda dikkate alınması yerinde olacaktır. Ayrıca kurulacak sistemin mutlaka yedekli bir yapıda olması ve aktif çözüm sağlayacak monitör amaçlı donanım ve yazılım çözümlerine sahip olması tasarım aşamasında göz önünde bulundurulmalıdır [15].

1.3. Çalışmanın Bilime Katkısı

Yapılan çalışmanın bilime olan katkısı maddeler halinde yazılmak istenirse şöyledir:

- 1- Bulanık mantık alanında ülkemizde yapılan çalışmalar bulunsa da yeterli olduğu söylenemez. Özellikle Japonya gibi ülkelerle karşılaştırıldığında

ülkemizde bulanık mantıkla ilgili yapılan her çalışmanın bilime katkısı olacağı açıktır.

- 2- Sunucu odası klima sistemlerinin otomasyonu ve uygun kontrol sistemlerinin tasarımı ile iklimlendirme sistemlerinde %20'ye varan oranlarda enerji tasarrufu sağlanabilir. Bu da tüm ülkede harcanan enerjinin yaklaşık olarak % 8'i kadardır. Bu oran iklimlendirme sistemlerinin kontrolünün ne kadar önemli olduğunu göstermektedir.
- 3- Yapılan literatür çalışmasında da görüldüğü üzere bulanık mantık kontrolör tasarımında kullanılan program sayısı sınırlıdır. Ülkemizde özellikle Matlab-Simulink ve az olarak da FuzzyTECH programı kullanılmaktadır. Labview programını açıklayıcı ve bulanık mantıkta kullanılması üzerine çok az sayıda makale bulunmuştur. Kaldı ki bu üç programı bir arada kullanıp sonuçlarını karşılaştıran herhangi bir araştırma bulunmamaktadır.
- 4- Bilimsel çalışmalarda tutarlılık sağlanması önemlidir. Bu çalışmada aynı giriş değişkenleri verildiğinde FuzzyTECH, MATLAB ve LabVIEW programlarıyla elde edilen çıkış değerleri, simülasyon sonuçları ve grafikleri 15 kritere göre karşılaştırılmış ve tutarlı sonuçlar elde edilmiştir.
- 5- Yapılan tez çalışması özellikle sunucu odalarının klima sisteminin simülasyonu olsa da her türlü bulanık mantık çalışmasında kullanılabilir ve üç programın verdiği sonuçlar karşılaştırılarak tam bir kontrol sağlanabilir.
- 6- Eğitim açısından da tez çalışmasının bilime katkısı bulunmaktadır. Eğitim kurumlarımızda bulanık mantık konusu anlatılırken MATLAB programı sıklıkla işlenmektedir. Ancak bulanık mantık denetleyici tasarım aşamasında sadece MATLAB programının kullanılmadığı, dünyada yaygın olarak FuzzyTECH ve LABVIEW Fuzzy Toolkit programların da kullanıldığı vurgulanmaktadır. Bu programların eğitim ortamlarına sağlayacakları katkılar da tez çalışmasının içinde bulunmaktadır.

1.4. Kullanılan Yöntem ve Teknikler

Tez çalışmamızda amaç örnek bir sunucu odası için bulanık mantık denetleyici simülasyonunu Matlab, FuzzyTECH ve Labview programlarını kullanarak yapmak ve sonuçlarını karşılaştırarak değerlendirme yapmaktır. Bu amaçla kullanılan yöntem ve teknikler şunlardır:

- 1- İlk olarak bulanık mantık konusunda araştırma yapılmıştır. Bu araştırmalarda bulanık mantığın ne olduğu ve uygulama alanları üzerine makale, araştırma, yazı ve tezler incelenmiştir. Bu araştırmalarda bulanık mantığın avantaj ve dezavantajları araştırılmış ve yapılan uygulamalar incelenmiştir. Bulanık mantığın ülkemizdeki uygulama alanları incelenmiş, hangi alanlarda daha yaygın olarak kullanıldığı ve hangi alanlarda ihtiyaç bulunduğu üzerinde durulmuştur.
- 2- Sakarya Üniversitesi Sürekli Eğitim Merkezi (SAÜSEM) içerisinde bulunan sunucu odası incelenmiş ve görevlilerden sunucuların yüksek sıcaklıklardan nasıl korunduğuyla alakalı bilgi alınmıştır. Çalışmada giriş değişkenleri olarak kullanılacak olan oda sıcaklıklarının değerleri, nem miktarı ve sunucu yoğunluğuyla ilgili veriler kendilerinden alınmıştır. Kullanılan klimaların özellikleri ve hangi şekilde kullanıldıkları incelenmiş ve notlar alınmıştır.
- 3- Simülasyon aşamasında ilk olarak kullanılacak program olan FuzzyTECH programı kurulmuş ve her yönüyle incelenmiştir. İnceleme sonuçları ayrı bir bölümde detaylı olarak açıklanmaktadır. FuzzyTECH programı ile simülasyon gerçekleştirilmiş ve giriş değişkenlerine karşılık elde edilen çıkış değerleri bulunmuştur.
- 4- Simülasyon aşamasında ikinci olarak kullanılacak program olan Matlab-Simulink programı kurulmuş ve her yönüyle incelenmiştir. İnceleme sonuçları ayrı bir bölümde detaylı olarak açıklanmaktadır. Matlab-Simulink programı ile simülasyon gerçekleştirilmiş ve giriş değişkenlerine karşılık elde edilen çıkış değerleri bulunmuştur.

- 5- Son olarak LabVIEW Fuzzy Logic Toolkit programı kurulmuş ve her yönüyle incelenmiştir. İnceleme sonuçları ayrı bir bölümde detaylı olarak açıklanmaktadır. Labview Fuzzy Logic Toolkit programı ile simülasyon gerçekleştirilmiş ve giriş değişkenlerine karşılık elde edilen çıkış değerleri bulunmuştur.
- 6- Üç program tüm yönleriyle karşılaştırılmış ve simülasyon sırasında giriş değerlerine karşılık elde edilen çıkış değerlerinin tutarlı olduğu görülmüştür. Aynı zamanda üç program kullandıkları bellek miktarı, işlemci kullanım yüzdesi gibi kriterlere göre de karşılaştırılmış ve sonuçlar belirtilmiştir.
- 7- Son olarak elde edilen sonuçlara göre programların birbirlerine olan üstünlük ve eksik tarafları belirtilmiş, bulanık mantık denetleyici tasarımında yol göstermesi adına yorumlar yapılmıştır.

BÖLÜM 2. BULANIK SİSTEMLER VE EĞİTİMİ

2.1. Bulanık Mantık

Bulanık mantık belki de farkında olmadan insanlar tarafından en çok kullanılan kontrol sistemidir. Örneğin biraz sıcak, çok genç, fazla kilolu gibi ifadeler aslında bulanık bir şekilde ifade edilmektedir, çünkü bulanık mantıkta sadece sıcak ve soğuk, sadece genç ve yaşlı ya da sadece kilolu ve zayıf şeklinde iki yapılı bir kontrolden daha fazlası vardır. Bulanık mantık klasik mantığın kalıplarının dışına çıkılması fikriyle doğmuştur. Gerçek dünyadaki her şeyin matematikteki belli kalıplara uyması gerekmediğini tezini ilk olarak Zadeh ortaya atmıştır [16]. Ona göre gerçek hayatta kesin çizgiler yoktur ve her şey 1 ve 0 gibi var-yok, evet-hayır, doğru-yanlış gibi net kalıplara hapsedilmemelidir. Zadeh'e göre belli üyelik dereceleri verilse her şeyden biraz, belli oranda var olma şeklinde bir kontrol sağlanabilir ve bu da gerçek hayata daha uygun bir yaklaşımdır.

Klasik mantık: sıcak, soğuk – genç, yaşlı – kilolu, zayıf

Bulanık mantık: biraz sıcak, biraz soğuk, ılık, çok sıcak, çok genç, orta yaşlı, çok yaşlı, az kilolu, çok zayıf, aşırı kilolu

Bulanık mantık kuramı, 1965 yılında, Azeri asıllı Amerikalı sibernetikçi Lutfi A. Zadeh tarafından ilk olarak ortaya atılmıştır. Bulanık mantık iki değerli mantık kuramlarına bir alternatif oluşturmuş, sibernetik ve onunla doğrudan ilgili bilimlerde akıl almaz derecede hızlı gelişmeleri tetiklemiş, sibernetik ve yapay zekâ çalışmalarını hızlandırmıştır. Halen süren ömrüne beş büyük bilimsel kuram sığdıran Lutfi A. Zadeh mantık, sibernetik, bilgisayar, yapay zekâ ve otomatik makinelerle dair ortaya koyduğu kuramlar ve keşfettiği kanunlarla 20. ve 21. yüzyıl teknoloji devriminde önemli pay sahibi olmuştur. Çünkü sibernetik, insan gibi hareket eden

makineler yapma girişimi olarak daha fazla otomasyon ve makineleşmeye imkân sunmaktadır [17].

Bulanık mantık ortaya atıldığı ülke olan Amerika'da çok da yaygınlaşma fırsatı bulamamıştır. Bunun nedenleri olarak klasik mantık kavramının ve bilimin kesin doğruluğunun şüphe götürmez gerçek olduğu düşüncesidir. Kullanılan bulanık (fuzzy) kelimesi de çokları tarafından anlaşılmaz ve gereksiz olarak bulunmuştur. Ancak sistemlerin daha esnek kontrol ve dizaynına olanak veren bulanık mantık kavramı doğduğu ülkeden çok uzaklarda Japonya'da sahiplenilmiş ve üzerine çok sayıda araştırma yapılmıştır. Günümüzde bulanık mantığın en yaygın bir kontrol yapısı olarak kullanıldığı ülke Japonya'dır. Japonya'da deprem çalışmalarında, hızlı tren çalışmalarında ve benzeri çok sayıda çalışmada bulanık mantık kullanılmaktadır. Bulanık mantık günümüzde yaygın olarak kullanılmaktadır. Kullanıldığı alanlar daha çok kontrol sistemleri dizaynında, bilgi işlem süreçlerinde, mühendislik alanlarında, yüz tanıma gibi üst düzey kontrol isteyen çalışmalarda kullanılmaktadır.

Lutfi A. Zadeh: “Bulanık mantık her şeyin, doğrunun da, bir derece meselesi olduğu insani akıl yürütme için bir modeldir. Temelde, sözcükle hesaplama anlamı sunmaktadır” demiştir [16].

Bulanık mantık gerçek dünyaya klasik mantığa göre daha yakın olduğundan, bulanık mantık doğrusal olmayan denetim için kullanılabilecek yeni ve alternatif bir yaklaşım olarak görülmektedir. Sistemlerin doğrusal olmayan karakteristikleri kurallar, üyelik fonksiyonları ve sonuca varma işlemi ile temsil edilir. Bulanık mantık yaklaşımının kullanılmasıyla sistem performansı artar, uygulama basitleşir ve maliyet azalır. Gerçek sisteme daha yakın olan, daha doğal bir kural tabanı kullanarak doğrusal olmayan denetim alışlagelmiş yöntemlere göre daha iyi biçimde gerçekleştirilebilir. Bu durumda sistem performansı mükemmel bir şekilde iyileştirilip daha etkili ve duyarlı bir denetim elde edilebilir.

Çoğu kontrol uygulamaları çok girişli olup çok sayıda parametrenin tasarlanıp, ayarlanmasını gerektirirler. Bu da uygulamayı zorlaştırıcı ve zaman alıcı bir işlemdir. Oysa bulanık mantık tabanlı bir denetleyicinin kuralları, doğrusal olmayan

özellikleri de dikkate alarak, çok sayıdaki girişi tekli if then sözel ifadeleriyle birleştirip uygulamayı basitleştirir. Ayrıca, bulanık mantık kullanılarak, çıkış büyüklüğü VE (AND) gibi işlemcilerle birbirine bağlanmış iki veya daha fazla girişin bir fonksiyonu olarak ifade edilebilir. Girişlerle çıkış arasındaki bu ilişki kurallardan oluşan bir tablo ile de gösterilebilir. Bu kural tablosu, el altı tablolarla karıştırılmamalıdır. Her bir giriş değişkeni için gerekli tanım sayısına bağımlı olan el altı tablolarına karşın, bulanık yaklaşım önemli ölçüde az giriş gerektirir. Ayrıca kurallar daha kolay geliştirilip, programa daha basitçe birleştirilebilir ve daha kolay ayarlanabilirler [18].

2.2. Bulanık Kümeler ve Üyelik Fonksiyonları

Bulanık mantık, sayıların iç içe ve birbiriyle ilişkili olması felsefesine dayanır. Eğer karar verme sürecinde bir durum bir sayıyla ifade ediliyorsa, söz konusu durumun kabul edilirliliği o sayının gerçekleşmesiyle sağlanacaktır. Ancak söz konusu sayıya yakın sayılar karar sürecinin bir parçası olarak algılanmayacaktır. Oysa belirli bir güven katsayısında bu sayıların farklı popülasyonların üyeleri olduğunu öne sürmek de istatistiksel açıdan yanlış olacaktır. Örneğin bir tezgâhta işlenen bir parçanın sıcaklığının 39 °C' ye ulaşması, tezgâhın bakım sürecini başlatan bir durumsa belki de sıcaklığın 36 °C' ye ulaşması da aynı bakım sürecinin başlaması için bir ön şart olarak kabul edilebilir. Bu durumda aynı temel amaca hizmet eden sayıların komşuluğundan söz etmek mümkündür. Yaralıoğlu K. bunu şu şekilde anlatmaktadır [19]:

Eğer $A \in R \in (-\infty, +\infty)$, da, söz konusu kümenin bir elemanı ise $\mu_A(x)$ üyelik fonksiyonu $R \rightarrow [0,1]$ aralığında oluşur. Başka bir deyişle A kümesi $A = [a_1, a_3]$ aralığında ise genel olarak $\mu_A(x)$ üyelik fonksiyonu (2.1) formülüyle gösterilebilir.

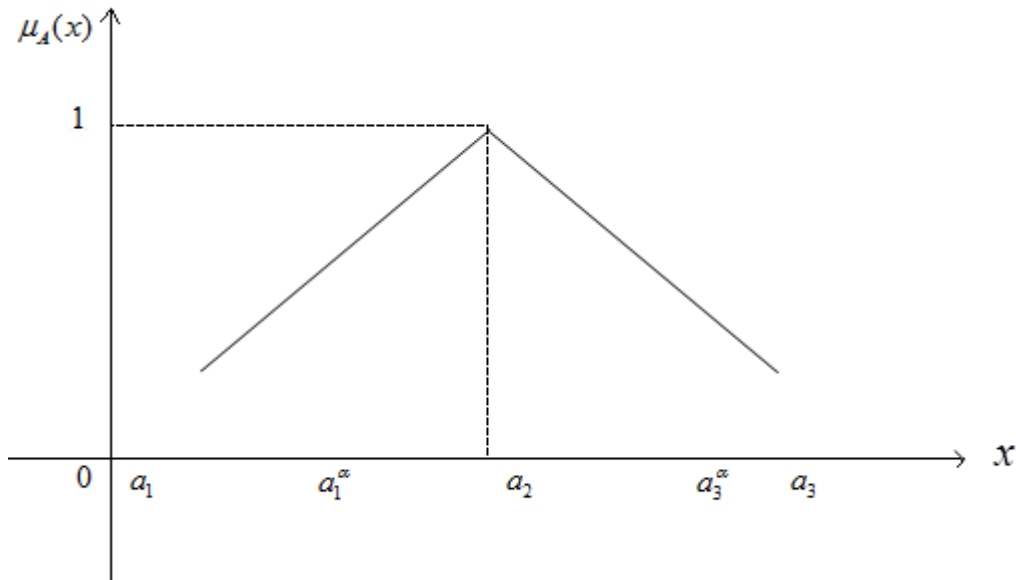
$$\mu_A(x) = \begin{cases} 0, & x < a_1 \\ 1, & a_1 \leq x \leq a_3 \\ 0 & x > a_3 \end{cases} \quad (2.1)$$

Üyelik fonksiyonları genellikle, üçgensel üyelik fonksiyonları ve yamuk üyelik fonksiyonları olmak üzere iki başlık altında incelenmektedir.

$\mu_A(x)$ üçgensel üyelik fonksiyonu, (2.2) formülünde tanımlanmıştır [20].

$$\mu_A(x) = \begin{cases} 0, & x < a_1 \\ \frac{x - a_1}{a_2 - a_1}, & a_1 \leq x \leq a_2 \\ \frac{a_3 - x}{a_3 - a_2}, & a_2 \leq x \leq a_3 \\ 0, & x > a_3 \end{cases} \quad (2.2)$$

(2.2) formülüne göre küme, $A = (a_1, a_2, a_3)$ olmalıdır. Burada a_2 normal değerli üyelik olarak tanımlanabilir. Bulanık Mantık bu noktada bir α katsayısına bağlı olarak a_2 ' ye yakın değerlerin, bu değere yüklenen anlam ile temsil edileceğini varsaymaktadır. Diğer bir deyişle a_2 ' deki belirsizlik, varsayılacak ya da dağılıma göre bulunabilecek bir α katsayısı ile tolere edilebilir. Söz konusu komşuluk Şekil 2.1' de gösterilmiştir.



Şekil 2.1. Sayıların komşuluğunun analitik düzlemde gösterimi [21].

α değeri bulanık mantık terminolojisinde kesim katsayısı olarak adlandırılır. a_1^α ve a_3^α sayıları ise a_2 normal değerinin komşuluğunu oluşturan aralığın alt ve üst sınır değerleridir. Diğer bir deyişle a_1^α ve a_3^α aralığındaki tüm sayılar a_2 normal değeri ile aynı anlama sahiptir. a_1^α ve a_3^α değerleri (2.3) ve (2.4) formülleri yardımıyla bulunabilir [22].

$$\frac{a_1^\alpha - a_1}{a_2 - a_1} = \alpha \quad (2.3)$$

$$\frac{a_3 - a_3^\alpha}{a_3 - a_2} = \alpha \quad (2.4)$$

(2.3) ve (2.4) formüllerinden $\forall \alpha \in [0,1]$ için $A_\alpha = [a_1^\alpha, a_3^\alpha]$ aralığı oluşturulabilir. a_1^α ve a_3^α değerleri (2.5) ve (2.6) formüllerinde gösterilmiştir.

$$a_1^\alpha = \alpha(a_2 - a_1) + a_1 \quad (2.5)$$

$$a_3^\alpha = a_3 - (a_3 - a_2)\alpha \quad (2.6)$$

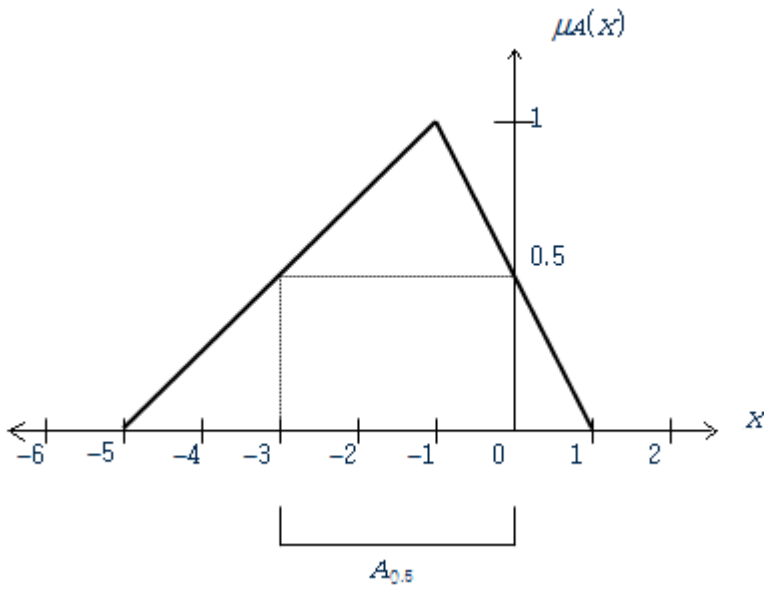
Örneğin üçgensel bulanık mantık sayılarına ilişkin küme $A = (-5, -1, 1)$ ise bu durumda (2.2) formülünden üyelik fonksiyonu,

$$\mu_A(x) = \begin{cases} 0, & x < -5 \\ \frac{x+5}{4}, & -5 \leq x \leq -1 \\ \frac{1-x}{2}, & -1 \leq x \leq 1 \\ 0, & x > 1 \end{cases} \quad (2.7)$$

olarak bulunur. Eğer karar verici α kesim katsayısını 0,5 olarak saptamışsa -1 normal değerinin komşuları (2.5) ve (2.6) formüllerinden $a_1^{0,5} = -3$ ve $a_3^{0,5} = 0$

olarak bulunacaktır. Diğer bir deyişle -1 normal değeri ile aynı anlam düzeyinde bulunan sayılar kümesi $[-3,0]$ aralığıdır. Söz konusu ilişki Şekil 2.2' de gösterilmiştir.

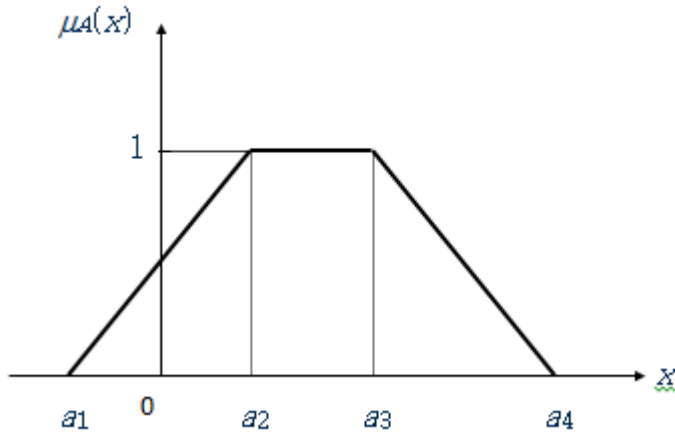
Eğer bulanık mantık sayılarına ilişkin kümede normal kabul edilen iki değer varsa diğer bir deyişle küme, $A = (a_1, a_2, a_3, a_4)$ şeklinde 4 belirleyici değerden oluşuyorsa bu durumda üyelik fonksiyonu yamuk üyelik fonksiyonu tipinde oluşacaktır. Yamuk üyelik fonksiyonu (2.8) formülünde gösterilmiştir.



Şekil 2.2 $A = (-5, -1, 1)$ kümesinin komşuluğu

$$\mu_A(x) = \begin{cases} 0, & x < a_1 \\ \frac{x - a_1}{a_2 - a_1}, & a_1 \leq x \leq a_2 \\ 1, & a_2 \leq x \leq a_3 \\ \frac{a_4 - x}{a_4 - a_3}, & a_3 \leq x \leq a_4 \\ 0, & x > a_4 \end{cases} \quad (2.8)$$

Söz konusu komşuluk Şekil 2.3' deki gibi oluşacaktır.



Şekil 2.3. $A = (0,1,1,0)$ kümesinin komşuluğu

2.3. Bulanık Mantık Eğitimi

Bulanık mantık aslında insanın doğasında vardır. İnsanlar ilkçağlardan itibaren bu konuları tartışmışlar ancak sistem haline getirilmesi bu yüzyılda gerçekleşmiştir. Aristo mantığına yapılan eleştiriler ve insanların esnek düşünebilmesi bulanık mantığın ortaya çıkıp yaygınlaşmasını sağlamıştır. Artık bulanık mantık tüm bilimsel alanlarda kullanıma açıktır. Sadece kontrol sistemleri tasarımında değil tıp, psikoloji, sosyoloji gibi beşeri bilimlerde de bulanık mantık yaklaşımı kabul görmekte ve ders olarak işlenmektedir.

Ülkemizde bulanık mantık bir ders olarak sadece üniversitelerde verilmektedir. Lise ve ortaokullarda ise bulanık mantıkla ilgili bir konu yoktur. Ancak bulanık mantığın sağladığı esnekliğin okullarımızda öğretilmesi ve uygulanması eğitimde başarının artmasına sebep olacaktır. Klasik eğitim sisteminde sadece öğretene konumunda olan öğretmen ve öğrenene konumunda olan öğrenci vardır ve bunlar keskin çizgilerle birbirinden ayrılmıştır. Bulanık mantık yaklaşımı kullanılsa öğretmen ve öğrenci kavramlarının yerini bir belirli bir yüzdelle öğrenen kavramı alacağı düşünülebilir. Çünkü öğretmen de öğrenci de birlikte belirli yüzdelle öğrenmektedir. Öğretmenler için öğretirken öğrenmek ve öğrenciler için de esnek bir yapıda konuyu öğrenmeye çalışmak vardır.

Yapılan arařtırmalarda bulanık mantık ders olarak verilmekle beraber daha çok yüksek lisans ve doktora tez konusu olarak görölmektedir. Lisans bölümlerinde verilen bulanık mantık dersleri de daha ziyade teorik olarak kalmakta ve pratikte pek kullanım alanı işlenmemektedir. Bulanık mantık lisans derslerinde bulanık mantık sistem tasarımı yapılırken de en fazla Matlab- Simulink üzerinde çalışılmakta diğeri programlar çoğru üniversitelerde ya hiç kullanılmamakta ya da sadece ismen geçmektedir.

Bulanık mantık derslerinde genel olarak işlenen konular şunlardır: Bulanıklık kavramı, bulanık kümeler, bulanık üyelik fonksiyonları, bulanık kümelerin özellikleri, temel bulanık işlemler, bulanık ilişkiler ve ilişkilendirme. Belirsizliğin bulanık modeli: Bulanık kümeleme ve paylaşırma. Bulanık kural tabanlı sistemler ve bulanık karar verme. Fiziksel sistemlerin modellenme ve kontrollerine kısa bir bakış. Bulanık mantık denetleyicilerin tasarlanması ve simülasyonu. Değişik bulanık mantık uygulama örnekleri konuları ders olarak işlenmektedir.

Esnek hesaplama ve yapay zeka tekniklerindeki hızlı gelişmeler son yıllarda bu yöntemlerin farklı alanlarda geniş kullanım alanı bulmasına yol açmıştır. Özellikle esnek hesaplama tekniklerinden en etkini olan bulanık mantık çeşitli bilim dallarına uygulanmış ve başarılı sonuçlar alınmıştır. Belirsizlik ortamında karar vermeyi ve sistem modellemeyi başarıyla gerçekleştiren bulanık tekniklerin Eğitim Bilimlerinde de etkinliğini artırması öngörölebilecektir. Sosyal olayların bulanık karakteri, eğitim-öğretim süreçlerinde gerçekleştirilen ölçümlerin belirli ölçülerde belirsizlik içermesi, uzman görüşlerin önemi ve eğitim teknolojilerindeki gelişmeler uygulamaların artması için zemin oluşturmaktadır [23].

2.3.1. Örnek bir bulanık mantık dersi ders içeriği

Sakarya Üniversitesi'nde bulanık mantık eğitimi lisans, yüksek lisans ve doktora seviyesinde verilmektedir. Lisans seviyesinde verilen bulanık mantık dersine örnek olarak Bilgisayar Sistemleri Öğretmenliği Bulanık Mantık dersinin içeriği şu şekildedir [24].

- a) Bulanık Kümeler. Üyelik fonksiyonları,
- b) Bulanık İşlemler. T-norm ve N-norm işlemler,
- c) Bulanık kurallar. Bulanıklaştırma, berraklaştırma, bulanık çıkarım,
- d) Mamdani bulanık çıkarım,
- e) Mamdani bulanık çıkarım uygulamaları,
- f) Sugeno bulanık çıkarım ve uygulama örneği,
- g) Matlab ortamında bulanık mantık uygulamaları,
- h) Beynin yapısı. Yapay sinir.
- i) Perceptron,
- j) Çok katmanlı sinir ağları,
- k) Öğrenme.
- l) Geri yayılım algoritması,
- m) Geri yayılımda momentum katsayısı,
- n) Matlab ortamında YSA uygulama örnekleri.

Sakarya Üniversitesi'nde yüksek lisans ve doktora seviyesinde verilen bulanık mantık dersine örnek olarak Bilgisayar ve Bilişim Mühendisliği Bulanık Sistemler ve Uygulamaları dersinin içeriği ise şu şekildedir:

- a) Neden bulanık mantık,
- b) Bulanık Kümeler. Üyelik fonksiyonları,
- c) Bulanık İşlemler. T-norm ve S-norm işlemler,
- d) Bulanık işlenenler. Genişleme prensibi,
- e) Bulanık kurallar. Bulanıklaştırma, berraklaştırma, bulanık çıkarım,
- f) Mamdani bulanık çıkarım,
- g) Mamdani bulanık çıkarım uygulamaları,
- h) Sugeno bulanık çıkarım,
- i) Sugeno bulanık çıkarım uygulamaları,
- j) Matlab Fuzzy tool. Matlab ortamında bulanık sistemler uygulamaları,
- k) Anfis ve Anfis uygulama örnekleri,
- l) Matlab Anfis tool. Matlab ortamında Anfis uygulamaları,
- m) Öğrenci uygulamaları ve makale çalışmaları,
- n) Öğrenci uygulamaları ve makale çalışmaları.

Görüldüğü üzere bulanık mantık eğitimi lisans, yüksek lisans ve doktora seviyelerinde ders olarak bulunmaktadır. Ders içerikleri incelendiğinde simülasyon için kullanılan programın MATLAB programı olduğu görülmektedir.

Bulanık mantık simülasyonlarında sadece MATLAB değil FuzzyTECH ve LabVIEW programlarının da kullanılması faydalı olacaktır. Simülasyonun farklı programlarda yapılabilmesi bu programların sağladığı üstünlük ve avantajları kullanabilmeyi sağlayacaktır. Bu şekilde daha çok kullanıcıya hitap etmek ve daha karmaşık sistem tasarımları mümkün olacaktır.

2.4. Bulanık Mantık Simülatörleri

Simülasyon diğer adıyla benzetim; teorik ya da fiziksel gerçek bir sistemin, bilgisayar ortamında modellendikten sonra bu model ile sistemin işletilmesi amacıyla yönelik olarak, sistemin davranışını anlayabilmek veya değişik stratejileri değerlendirebilmek için deneyler yürütülmesi, bu sistemlerin özelliklerini ve davranışlarını bilgisayar aracılığıyla değerlendiren bir tekniktir [25]. Simülasyon ne zaman kullanılır, sorusuna şu cevaplar verilebilir:

- 1- Sistem deney yapmaya uygun değilse,
- 2- Sistem henüz tasarım aşamasındaysa,
- 3- Sistem/Problem karmaşıksa,
- 4- Sistemin davranışı analiz edilecekse ve
- 5- Bilgisayar mevcutsa ve eğitim amacıyla kullanılır.

“ İyi bir simülasyon nasıl olmalıdır? ” sorusuna şu cevaplar verilebilir:

- 1- Kullanıcı tarafından kolayca anlaşılmalıdır.
- 2- Amaç veya hedef yönlü olmalıdır.
- 3- Kullanıcı tarafından denetimi ve işletilmesi kolay olmalıdır.
- 4- Tam olmalıdır.
- 5- Model değişikliği ve güncelleştirilmesi için kolaylıkla uyarlanabilir olmalıdır.

6- Evrimsel olmalıdır; yani basit bir şekilde başlayıp giderek karmaşıklaşmalıdır [25].

Bilgisayar tabanlı yazılım ile bulanık mantık sistemine entegre etmek istenirse eğer; süreç otomasyonu, ölçüm ve simülasyon yazılımları (FISPro, XFuzzy, Matlab, LabVIEW, WinCC, JFuzzyLogic, AwiFuzz, Citect, VisSim, ..) ve standart yazılımlar (Excel, VisualBasic, VisualC++, Access, ..) kullanılabilir [26].

Simülasyon sürecinde en çok Matlab FIS (MATLAB for Fuzzy Inference Systems), Inform FuzzyTECH ve Labview Fuzzy Toolkit kullanılmaktadır.

BÖLÜM 3. BULANIK MANTIK MODEL GELİŞTİRME ORTAMLARI

Bulanık mantık uygulamalarında simülasyon yapmadan önce model geliştirme ortamlarının incelenmesi gereklidir. Bu şekilde hangi programın istenen amaca yönelik kullanılabileceği, avantaj ve dezavantajları bilinmiş olur ve sistem kurulurken buna göre kurulur. Bu da tabii ki zaman ve iş gücünün gereksiz yere kullanılmasını engellemiş olur. Bu tez çerçevesinde incelenecek olan programlar Almanya merkezli Inform firması tarafından geliştirilen FuzzyTECH, ABD merkezli The MathWorks, Inc. firması tarafından geliştirilen Matlab (Matrix Laboratory) ve National Instruments Corporation (NI) tarafından geliştirilmiş olan LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) programları en yaygın kullanılan bulanık mantık araçlarıdır.

3.1. FuzzyTECH Model Geliştirme Ortamı

FuzzyTECH bulanık mantık tabanlı kontrol ve karar verme sistemleri tasarım ve geliştirilmesinde dünyanın önde gelen yazılım araçlarından birisidir. FuzzyTECH programı piyasaya çıktığı ilk günden itibaren kendisini değişik alanlarda ispat etmiştir. FuzzyTECH programı ile çözülemeyen herhangi bir bulanık mantık probleminin başka bir yazılımla yada yaklaşımla da çözülemeyeceği söylenebilir. Bugüne kadar 3800 civarında bulanık mantık projesi FuzzyTECH programı kullanılarak geliştirilmiştir [27]. Son kullanıcı olarak FuzzyTECH programı kullanan kişilerin herhangi bir programlama dili bilmesine yada bulanık mantığın tüm konularını anlamış olmasına gerek yoktur, çünkü FuzzyTECH içerisinde çok geniş bir yardım kütüphanesi ve destek fonksiyonları vardır. Bu vesile ile FuzzyTECH programı çeşitli donanım ortamlarında çalışabilmektedir. Inform firması çok sayıda büyük şirketle de bulanık mantık çalışmaları yapmaktadır.

3.1.1. FuzzyTECH programının genel görünümü

Inform firması tarafından kodlanan FuzzyTECH programı bulanık mantık işlemleri yapmak için programlanmış, tamamıyla bulanık mantık geliştirme ortamıdır. Grafik tabanlı editörü ve analiz araçları sayesinde etkileyici sistem tasarımları, bu sistemlerin optimizasyonu ve doğrulamaları gerçekleştirilebilmektedir. Bir tuşa basılmasıyla birlikte FuzzyTECH programı bulanık mantık tasarımlarını yüksek gerçeklikteki simülasyonlar yardımıyla gösterebilmektedir. Bunun sonucunda ise özel donanımına olan ihtiyaç en aza inmektedir. FuzzyTECH tüm Windows işletim sistemi ortamlarında çalışabilmektedir.

Çoğu teknik uygulamalarda geliştirme ortamı olarak kullanılan bir bilgisayar ve bulanık mantık sisteminin çalıştırılacağı bir platform vardır. FuzzyTECH programı sadece geliştirme ortamı olan bilgisayar için tasarlanmıştır. Sistem gereksinimi olarak Windows 7, Windows Vista, Windows XP, Windows 2000, Windows NT 4.0 yada Windows 98 hatta Windows 95, IE 5.01 yada daha yükseği, 30 MB boş disk alanı ve 16 MB bellek yeterlidir.

FuzzyTECH geliştirme ortamı olarak sadece kişisel bilgisayarları kullanmasına rağmen geniş yelpazede bulanık sisteminin çalıştırılacağı platformları da destekler. Örneğin “Benim bulanık mantık uygulamam bir PC / Workstation üzerinde çalışacak. Benim uygulama alanım kontrol mühendisliği, sensör sinyal işleme veya teknik veri analizidir. “ diyen bir kullanıcı için standart sürüm olarak FuzzyTECH Professional Edition diğer yazılım sistemleri için tüm tasarım araçları, bulanık teknolojileri ve ara yüzleri sağlar. “Ben gerçek zamanlı olarak sistemimi optimize etmek istiyorum.” diyen bir kullanıcı içinse FuzzyTECH Professional Edition, tüm FuzzyTECH IA-XX sürümleri ve bazı FuzzyTECH MCU-XX sürümleri (RTRCD desteği olan) tavsiye edilmektedir. Farklı kriterlere göre FuzzyTech sürümleri karşılaştırılmak istenirse ilk önce hangi kriterlere göre karşılaştırılacağı belirlenmelidir. Aşağıdaki kriterler kullanılırsa;

Kriter 1: Bulanık mantık pozisyon kontrolörü (7 kural, Her biri 3 terim içeren 2 giriş değişkeni, 5 terimli 1 çıkış değişkeni, Kural ağırlıkları = 1),

Kriter 2: Bulanık PI kontrolörü (20 kural, Her biri 5 terim içeren 2 giriş değişkeni, 5 terimli 1 çıkış değişkeni, Kural ağırlıkları = 1),

Kriter 3: Bulanık PD kontrolörü (20 kural, Her biri 5 terim içeren 2 giriş değişkeni, 5 terimli 1 çıkış değişkeni, çeşitli ağırlıkta kurallar),

Kriter 4: ABS için parametre hesaplayıcı (80 kural, 3/4/6 terim içeren 3 giriş değişkeni, 5 terimli 1 çıkış değişkeni, çeşitli ağırlıkta kurallar),

Kriter 5: Araçlar arası dinamik kontrol (500 kural, Her biri 7 terim içeren 8 giriş değişkeni, 7 terimli 4 çıkış değişkeni, çeşitli ağırlıkta kurallar).

Verilen girişler ve kurallar sonucunda elde edilen çıkışların karşılaştırılması Tablo 3.1’de değişik ortamlar için verilmiştir.

Tablo 3.1 Değişik ortamlar için FuzzyTECH programının verdiği cevaplar [27].

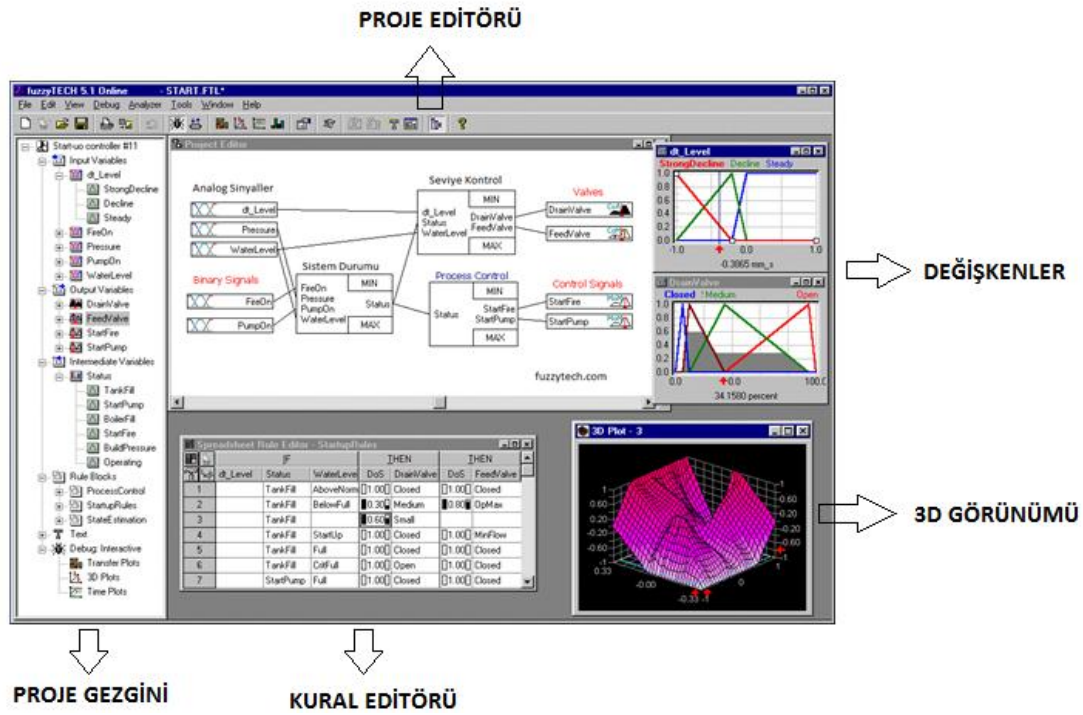
Kriterler	Kriter 1	Kriter 2	Kriter 3	Kriter 4	Kriter 5
İşlemci/Kontrolör	7 Kural 2 Giriş / 1 Çıkış	20 Kural 2 Giriş / 1 Çıkış	20 Kural 2 Giriş / 1 Çıkış	80 Kural 3 Giriş / 1 Çıkış	500 Kural 8 Giriş / 4 Çıkış
PC: fuzzyTECH Professional Edition, C kod oluşturucu, 16 bit çözünürlük, test cihazı 80486SX, 33 MHz, MS-C 8.00	0.05/0.07 ms 0.73 KB OBJ 39 Byte RAM	0.05/0.08 ms 0.82 KB OBJ 47 Byte RAM	0.05/0.09 ms 0.89 KB OBJ 47 Byte RAM	0.09/0.12 ms 1.27 KB OBJ 53 Byte RAM	0.35/0.45 ms 3.5 KB OBJ 185 Byte RAM
80Cx96: fuzzyTECH MCU-96 Edition, Assembly kod üreticisi, 16 bit çözünürlük, test cihazı 80C196KD, dahili ROM, 20 MHz	0.19/0.23 ms 0.63 KB ROM 50 Byte RAM	0.24/0.33 ms 0.69 KB ROM 58 Byte RAM	0.26/0.38 ms 0.78 KB ROM 58 Byte RAM	0.44/0.64 ms 1.17 KB ROM 66 Byte RAM	1.87/2.45 ms 3.47 KB ROM 214 Byte RAM
80C16x: fuzzyTECH MCU-166 Edition, Assembly kod üreticisi, 16 bit çözünürlük, test cihazı 80C166, dahili ROM, 20 MHz	0.05/0.06 ms 0.44 KB ROM 30 Byte RAM	0.06/0.08 ms 0.56 KB ROM 30 Byte RAM	0.06/0.09 ms 0.61 KB ROM 30 Byte RAM	0.10/0.14 ms 0.99 KB ROM 36 Byte RAM	0.43/0.56 ms 3.2 KB ROM 194 B RAM
68HC11: fuzzyTECH MCU-11/12 Edition, Assembly kod üreticisi, 16 bit çözünürlük, test cihazı 68HC11, harici ROM, 8 MHz	1.1 ms/1.3 ms 0.64 KB ROM 31 Byte RAM	1.4ms /1.9 ms 0.71 KB ROM 35 Byte RAM	1.4 ms/1.9 ms 0.75 KB ROM 35 Byte RAM	2.6 ms/3.5 ms 1.1 KB ROM 39 Byte RAM	11 ms/14 ms 3.1 KB ROM 113 Byte RAM
68HC12: fuzzyTECH MCU-11/12 Edition, Assembly kod üreticisi, 16 bit çözünürlük, test cihazı 68HC12, harici ROM, 16 MHz	0.09ms/0.09ms 0.34 KB ROM 26 Byte RAM	0.14ms/0.15ms 0.42 KB ROM 30 Byte RAM	0.16ms/0.17ms 0.54 KB ROM 30 Byte RAM	0.40ms/0.41ms 1.4 KB ROM 34 Byte RAM	2.1 ms/2.2 ms 8.8 KB ROM 108 Byte RAM

FuzzyTECH programı 1990 yılından başlayarak çok sayıda uygulamada diğer geliştirme araçlarından daha başarılı sonuçlar elde etmiştir. FuzzyTECH bu sayede dünyanın önde gelen çok sayıda büyük şirketiyle bulanık mantık çalışmaları yapmıştır. Bu şirketlerden bazıları şunlardır: AEG, Audi, Blaupunkt GmbH, Intel Corporation, Siemens, Toshiba, Volkswagen.

3.1.2. FuzzyTech programının özellikleri

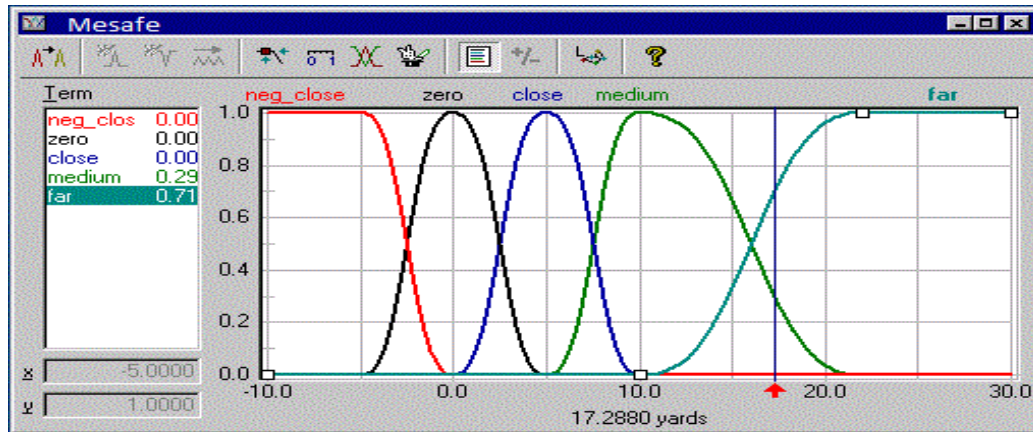
1. FuzzyTECH programı teknik uygulamalar için özelleştirilmiş bir programdır.
2. FuzzyTECH tasarımlara bulanık mantığın avantajlarını yerleştirmeyi sağlar.
3. FuzzyTECH diğer geliştirme ortamlarından daha başarılı bir şekilde bulanık mantık uygulamaları yapmayı sağlar.
4. FuzzyTECH kontrol sistemi üreticileri tarafından diğer bulanık mantık araçlarından daha fazla lisans almıştır.
5. FuzzyTECH tasarımın gerçekleştirileceği hedef sistemi piyasada bulunan diğer bulanık mantık araçlarından daha fazla destekler.
6. FuzzyTECH karmaşık ve hızlı sistemleri diğer bulanık mantık araçlarından daha fazla destekler.
7. FuzzyTECH programının bulanık tasarım sihirbazı çözümümüze daha kolay ulaşmamızı sağlar.
8. FuzzyTECH dosya indirme sayfasından animasyonlu simülasyon örnekleri indirip test etmeye imkan vermektedir.

FuzzyTECH programı son nesil Windows ve MS Office programlarının arayüzlerine benzer bir arayüz sunmaktadır. Programın üst kısmında başlık çubuğu, menü çubuğu ve araç çubukları bulunmaktadır. Windows gezgini Windows üzerinde nasıl kullanıcılara bilgisayarlarında olan dosyaları gösteriyorsa, FuzzyTECH programında bulunan dallanma yapısı da aynı şekilde kullanıcılara çalıştıkları dosyaların tamamına erişmelerini sağlar. Editör ve Analiz pencereleri grafiksel bir bulanık mantık sisteminin her bir bileşeninin tasarımına olanak sağlar. Analiz aşamasında ekran görüntüsü değişmektedir. Aynı anda hem analiz sonuçları hem de tasarlanan sistem görülebilmektedir. FuzzyTECH programının genel ekran görüntüsü Şekil 3.1'de görüldüğü gibidir.



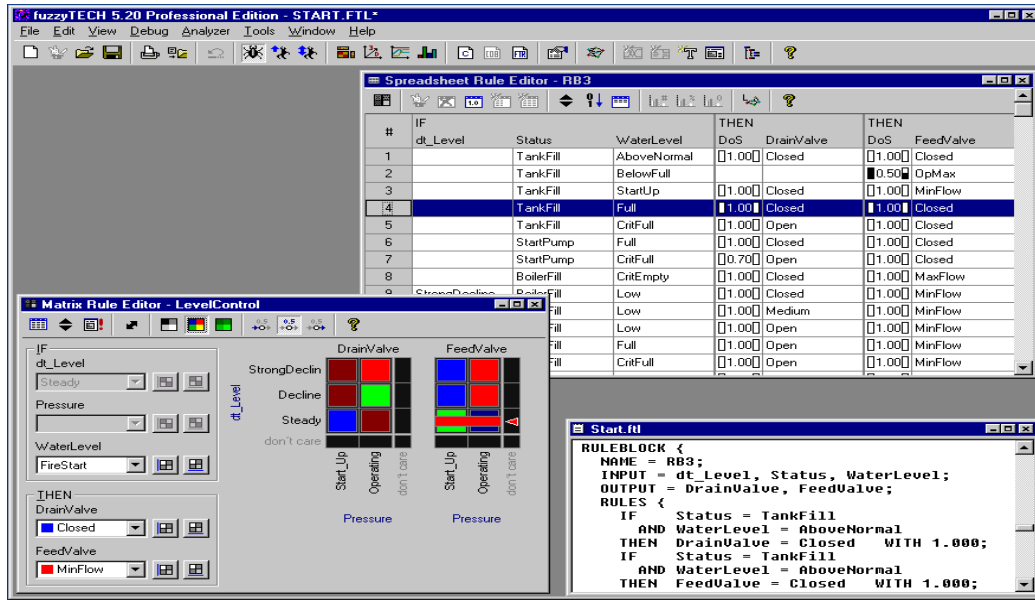
Şekil 3.1 FuzzyTECH programının genel ekran görüntüsü

Dilsel değişkenler bir bulanık mantık sisteminin can damarlarıdır. FuzzyTech programının dilsel değişken editörü sayesinde değişik tipteki üyelik fonksiyonları kolayca fare yardımıyla çizilebilir. Hata ayıklama modunda dilsel değişken editörü bulanıklaştırmayı da görselleştirmektedir. İnteraktif hata ayıklama modunda ise kırmızı fare işaretçisi hareket ettirilerek girişlere karşı verilen çıkış değerleri kolayca görülmektedir. Şekil 3.2’de FuzzyTECH programında dilsel değişkenlerin nasıl gösterildiği görülmektedir.



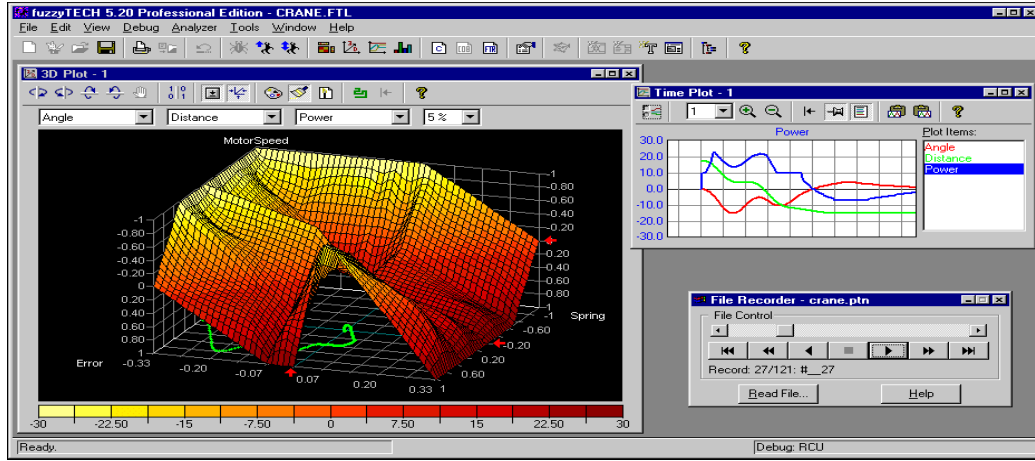
Şekil 3.2. FuzzyTECH programında dilsel değişkenlerin ifade edilmesi

Bulanık mantık kuralları sistemin giriş koşullarına nasıl erişeceğini kararlaştırır. Bu da çözümü ve sistemi kontrol etmek için uygun stratejiler geliştirme imkanı sağlar. Farklı uygulamalar için farklı kural blok editörlerinin olması FuzzyTECH programının sunduğu en iyi hizmetlerdendir. Çalışma kağıdı kural editörü tabloya yakın stilde kuralları takip etmenizi sağlar. Tüm kural tabanlarını tanımlamak için yapılması gereken işaretlemek ve tıklamaktır. Otomatik olarak kurallar işlenir. Uzman kullanıcıların çoğu karışık kural blokları için matris kural editörünü tercih ederler. Bulanık teknoloji dili içerisinde kuralları girmek en rahat yoldur. Kuralları girmenin bir diğer yolu ise her zaman diğer editörleri kullanmaktır. Şekil 3.3'de tüm kural editörlerini görmek mümkündür.



Şekil 3.3. FuzzyTECH programında kural editör penceresi

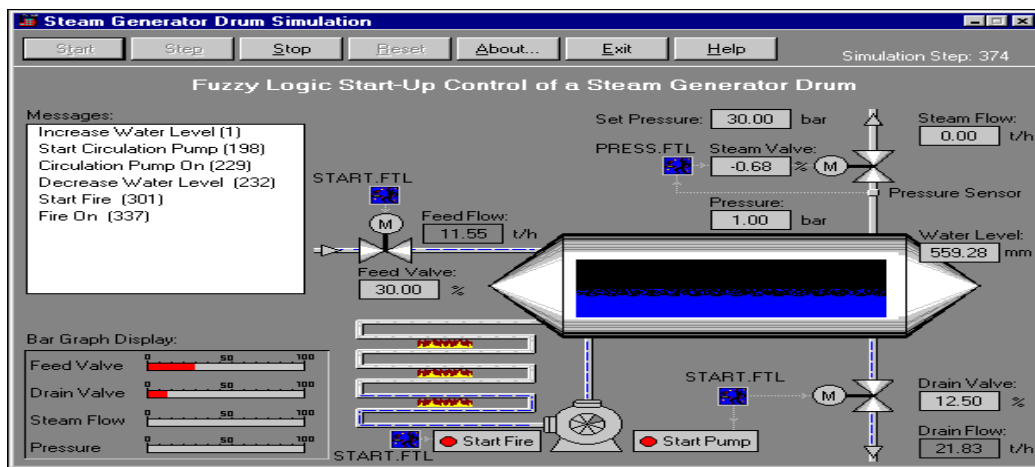
FuzzyTECH programının zengin analiz araçları görsel simülasyon yazılımının standartlarının oluşturmuştur. 2 ve 3 boyutlu analiz araçları bulanık mantık kontrol sisteminin veya bileşenlerinin herhangi bir yolla transfer karakteristiklerinin analiz edilmesini sağlar. Sistemdeki her türlü değişim analiz sonuçlarına aynen yansıtılmaktadır. Bu şekilde çözüme interaktif bir şekilde ulaşmak mümkün olmaktadır. Zamana göre çizim çözümleri sayesinde bulanık mantık kontrolörün işlem süreci esnasında nasıl çalıştığı anlaşılabilir ve sistem optimizasyonu gerçekleştirilebilir. Şekil 3.4 FuzzyTECH programında bulunan 3 boyutlu çizim ekranını göstermektedir.



Şekil 3.4. FuzzyTECH programında 3 boyutlu çizim ekranı

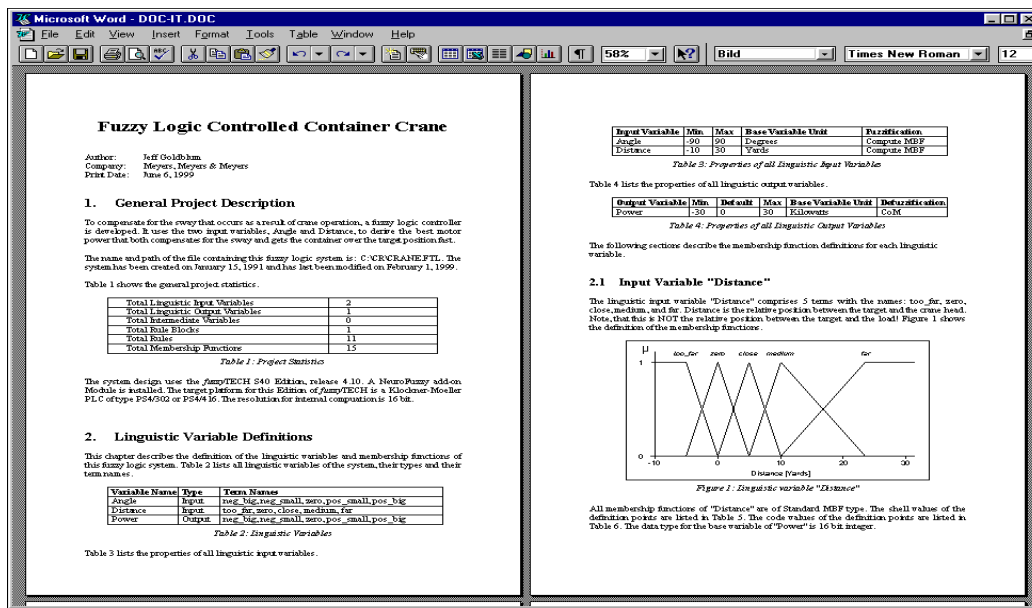
FuzzyTECH programı tasarım esnasında ihtiyaç duyulan bulanık mantık sistemiyle ilgili tüm simülasyon özelliklerini Şekil 3.4'te de görüldüğü gibi içinde bulundurmasıyla birlikte Matlab Simulink ve diğer bulanık mantık tasarım programlarıyla uyumlu bir şekilde çalışabilmektedir.

Simülasyon esnasında FuzzyTECH 'in tüm editörleri ve analiz araçlarını kullanmak mümkün olmaktadır. Bu da sistemin çalışma anında düzenlenebileceği anlamına gelir. Tüm FuzzyTECH versiyonlarını demolarıyla beraber web sayfası üzerinden indirmek ve kullanmak mümkündür. Bu demolar içerisinde bulunan gerçek yaşam süreçleriyle ilgili uygulamalar da kullanıma açıktır. Şekil 3.5 örnek bir simülasyon anındaki ekran görüntüsünü göstermektedir.



Şekil 3.5. FuzzyTECH programıyla yapılmış bir simülasyon çalışması

FuzzyTECH programının yapısal tasarım yaklaşımı, belge oluşturma özelliği ve değişiklik yapmayı sağlayan kontrol sistemiyle beraber tamamlanmıştır. Belge oluşturma özelliği sayesinde oluşan zengin metin biçimli (RTF) dosyaları standart kelime işlemcileri ile görebilmek, yazdırabilmek ve düzenlemek mümkündür. Gözden geçirme sistemi, işlemi daha eski duruma getirebilme ve belge halinde kullanabilmeyi sağlar. Belgelendirme yapısının tamamı IEC 1131-7 bulanık mantık geliştirme standartları içerisinde yer almaktadır. FuzzyTECH programında raporlama ekranı vardır ve bu ekran Şekil 3.6'da görülmektedir.



Şekil 3.6. FuzzyTECH programı raporlama ekranı

FuzzyTECH programı Japonya'da, Avrupa ve Amerika'da gerçekleştirilen çok sayıda bulanık mantık uygulamalarının temelini oluşturmaktadır. Aynı zamanda FuzzyTECH programı Inform GmbH tarafından sürekli olarak geliştirilmeye devam edilmektedir. Eklentileri sayesinde bulanık mantık tasarımlarıyla birlikte sinirsel ağ teknolojilerini de destekler. FuzzyTECH programının tamamı grafiksel olan editör arayüzü basitçe işaretleyerek ve tıklayarak tüm sistemin belirtilmesine olanak verir.

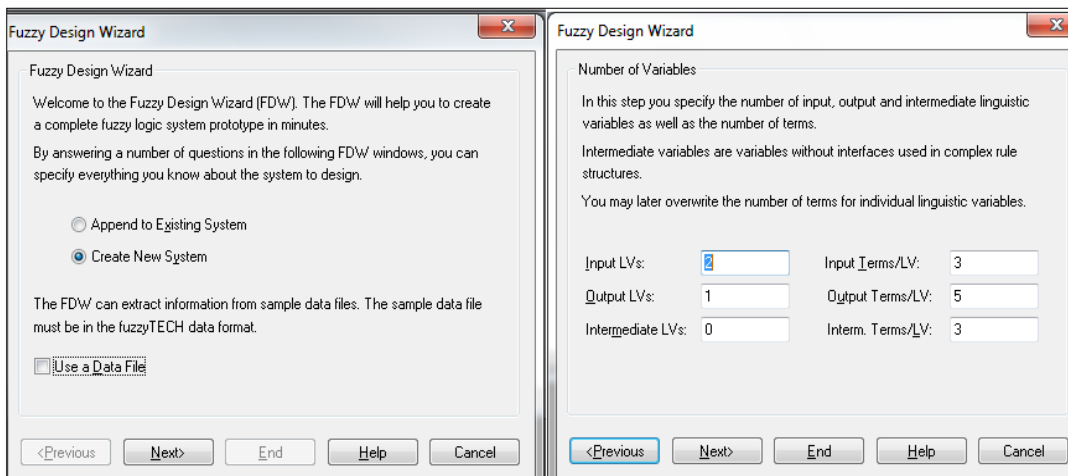
FuzzyTECH programının kod oluşturucu kısmı sayesinde C ve Assembly kodları oluşturmak ve bunu diğer yazılımlarla entegre etmek mümkündür. FuzzyTECH oluşturulan sistemin dokümantasyonunu gerçekleştirebilme ve gözden geçirebilme özellikleri sayesinde ISO 9001 standartlarını destekler.

3.1.3 FuzzyTECH Bulanık Mantık Tasarım Sihirbazı

FuzzyTECH programının bulanık kontrol sistemleri geliştirmekte kullanılan standart özelliklerinin yanı sıra programda bulunan Bulanık Tasarım Sihirbazı yardımıyla şunlar yapılabilmektedir:

- Hızlı uygulama tasarımları,
- İlk örneği oluşturma desteği,
- IEC ve ISO bulanık mantık yaklaşım standartlarını oluşturma sürecinde rehberlik etme,
- FuzzyTECH programının kullanan yeni kullanıcılara dakikalar içerisinde bulanık mantık kontrol sisteminin bir örneğini oluşturabilme,
- Uzman tasarımcılar, tasarım süreçlerini çabuklaştırmakta ve modernize edebilmektedirler.

Bulanık mantık sihirbazı ilk olarak yeni bir sistem mi kurulacağı yoksa var olan bir sistem üzerine mi çalışma yapılacağını sorar. Daha sonra giriş ve çıkış değişkenlerinin sayısını ve terimlerinin kaç tane olacağını ister. Giriş ve çıkış değişkenlerinin tanımlanmasından sonra çıkış değişkeninin bulanıklaştırma yöntemini sorar. Tüm bu bilgileri aldıktan sonra kural veritabanının tanımlanmasını, kaç tane kural bütünü olacağını ister ve sistemi oluşturur. Şekil 3.7 tasarım sihirbazının değişik anlardaki görüntüsünü vermektedir.



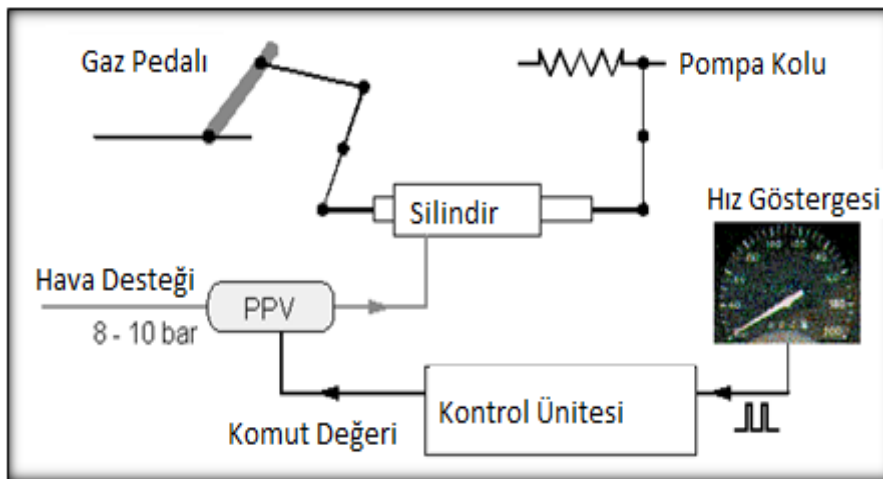
Şekil 3.7 FuzzyTECH bulanık mantık sihirbazı görünümü

3.1.4 FuzzyTECH ile Örnek Bir Uygulama: Kamyon Hız Kontrolü

Avrupa'da 12 ton azami yük taşıyan kamyonların Avrupa yasalarına uygun olarak maksimum 53.3 mph (86 km / h) hız sınırlayan bir hız sınırlayıcısı ile donatılmış olması gerekmektedir. Bu çalışmada hız sınırlayıcı gibi bir elektro-pnömatik tasarım üzerinde durulmuştur [27]. Bu tasarım, bir pnömatik silindir mekanik yakıt pompası kolu vasıtasıyla gaz kelebeği açılma açısını sınırlar. Silindir basıncı bir darbe orantılı elektro valf tarafından kontrol edilir. Bu valf gerçek hıza göre hızı ayarlamak için mikroişlemci kullanan bir elektronik kontrol ünitesine bağlanır. Aynı hız sınırlayıcı cihaz farklı davranışlar sergileyen farklı kamyon kullanıcılarının değişik davranışları olabileceğinden dolayı bu kontrol problemi için bir algoritma tasarımı, zor olduğu kanıtlanmış bir çalışmadır. Zaten bulanık mantıktaki en büyük problem de budur.

Buna ek olarak, bir kamyonun dinamik davranışı kamyonun tamamen dolu ya da boş olup olmadığına bağlı olarak da değişir. PID kontrol gibi geleneksel kontrol algoritmaları, kontrol altında işleminin bir doğrusal modeli yoksa kullanılamaz.

Kamyonun matematiksel modelini kullanarak bir çözüm üretmek birincisi çok zahmetli bir kurulum ve ikincisi düşük maliyetli bir 8 bit mikroişlemci için gereksiz bir hesaplama çabasıdır. Dolayısıyla kontrol tasarım algoritması için bulanık mantık kullanmak daha akıllıcadır. Sistemin mekanik modeli Şekil 3.8'de verilmektedir.

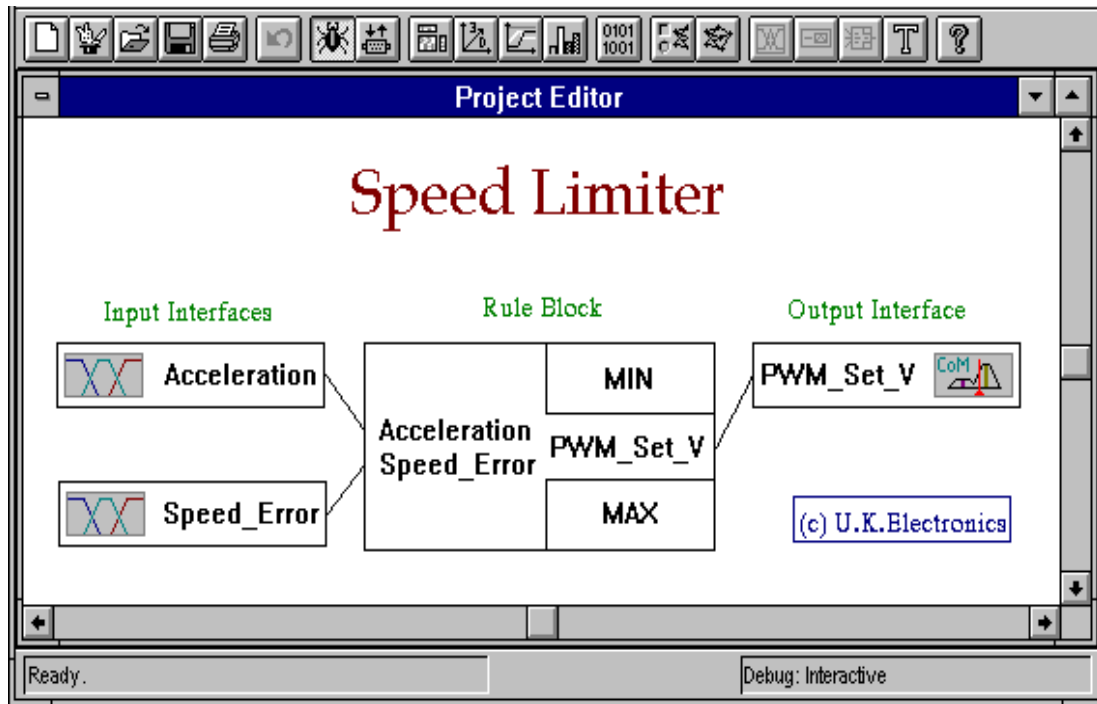


Şekil 3.8. Tasarlanan sistemin mekanik modeli

Sistemin mekanik modeli tasarlandıktan sonra sistem için elektronik devre kartı tasarlanmıştır. Elektronik kontrol ünitesi, karışık dijital ve analog bir devre olarak

tasarlanmıştır. Kilometre sinyal işleme, bulanık mantık kontrol algoritması ve teşhis fonksiyonları bir PIC 8-bit mikrodenetleyici (MCU) ile hesaplanır. MCU harici bir EEPROM'u kamyon ve hız, maksimum hız, ve tanı değişkenlerinin parametrelerini saklamak için harici kullanır. Tüm bunlardan sonra ise bulanık mantık denetleyici FuzzyTECH programı kullanılarak tasarlanmıştır.

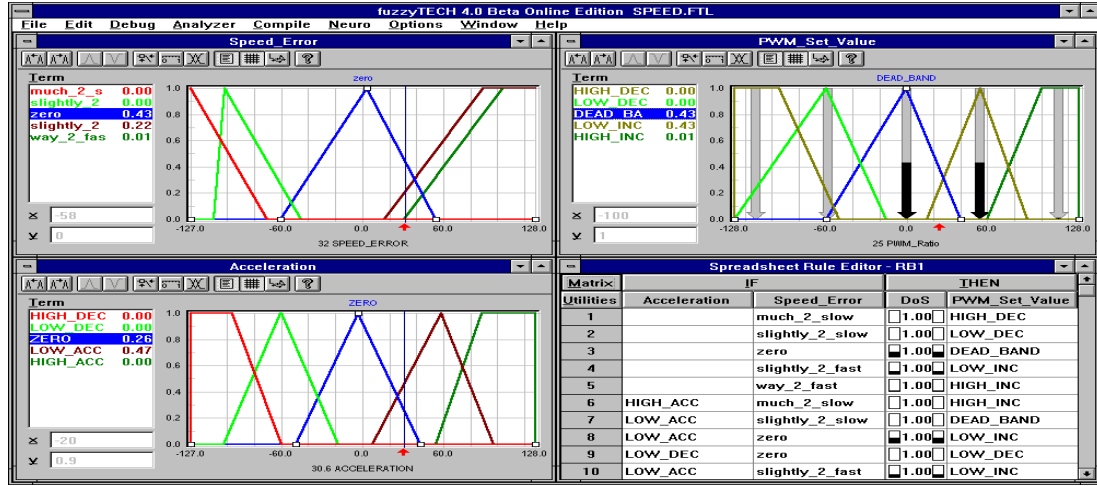
Bulanık mantık çok parametrelili ve doğrusal olmayan kontrol problemlerinin çözümünde yenilikler sunan bir teknolojidir. Kontrol stratejisi tanımlamak için matematik modelden ziyade insan deneyimi ve deneysel sonuçları kullanır. Sonuç olarak genellikle daha hızlı kontrol tasarım teknikleri sunar. Aynı zamanda mikro denetleyiciler üzerindeki bulanık mantık uygulamaları daha verimli olmaktadır. Tasarlanan bulanık mantık hız kontrolörünün yapısını gösteren FuzzyTECH Proje Editörü yapısı Şekil 3.9'da verilmiştir:



Şekil 3.9. Tasarlanan sistemin Proje Editörü görünümü

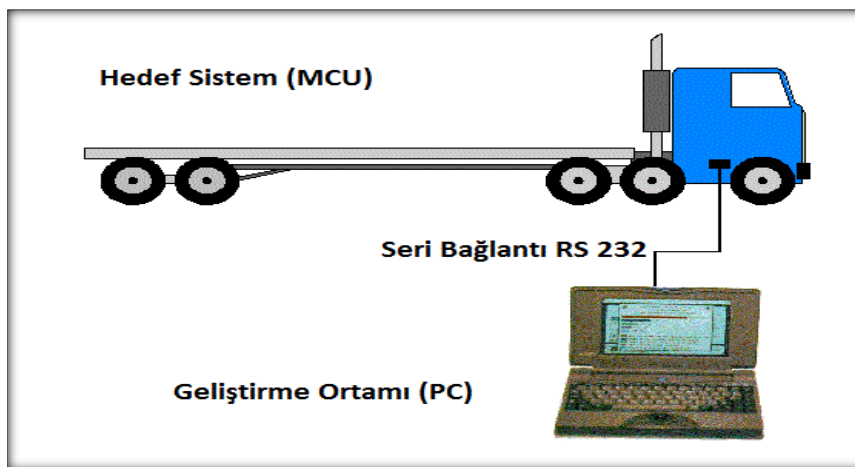
Şekil 3.10 bulanık mantık sistemini daha ayrıntılı olarak göstermektedir. Her dilsel değişken bir değişkenin düzenleyici penceresinde görüntülenir ve kuralları Tablolama Kural Editor penceresinde görüntülenir. Her dilsel değişken beş (5) kayıt

ve üyelik fonksiyonları içerir. Bunlar 12 bulanık mantık kurallarına bağlıdır. Tasarlanan sistemin simülasyon anındaki görüntüsü Şekil 3.10'daki gibidir.



Şekil 3.10. Tasarlanan sistemin simülasyon anındaki görüntüsü

Bulanık mantık sistemi için verilen tüm kurallar tasarımcıya verilen durum için en iyi çıkış değerini tanımlamasına olanak sağlar. Bu durum giriş değişkenleri kombinasyonu ile açıklanmaktadır. Böyle bir kontrol tasarımcı tarafından tanımlandıktan sonra sistem performansını test etmek mümkündür. FuzzyTECH programının hedef donanım olmadan bulanık mantık sistemini simüle edebilmesi sayesinde sistem başarılı bir şekilde bilgisayar üzerinde optimize edilmiştir. Bununla birlikte sistemin nihai optimizasyonu ve doğrulanması gerçek bir kamyon üzerinde olmuştur. Sistem Şekil 3.11'deki gibi gerçek dünyaya uygulanmıştır.



Şekil 3.11. Sistemin gerçek dünyaya uygulanmasının şematik gösterimi

3.1.5. FuzzyTECH sürümleri

FuzzyTECH programı sürekli olarak güncellenmekte, piyasanın ihtiyaçlarına cevap verecek şekilde geliştirilmektedir. FuzzyTECH programının sürümleri Tablo 3.2’de görüldüğü gibidir:

Tablo 3.2 FuzzyTECH programının sürümleri

Sürüm Adı	Sürüm Tarihi
FuzzyTECH 6.0	12 Temmuz 2012
FuzzyTECH 5.81	8 Eylül 2010
FuzzyTECH 5.72	12 Şubat 2009
FuzzyTECH 5.61	12 Şubat 2007
FuzzyTECH 5.60	22 Şubat 2006
FuzzyTECH 5.55	31 Mart 2006

3.2. LabVIEW Bulank Mantık Model Geliştirme Ortamı

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) görsel programlama yapmak için National Instruments firması tarafından ilk olarak 1986 yılında geliştirilmiş olan bir sistem tasarım ve geliştirme ortamıdır.

1986'dan beri mühendisler ve bilim adamlarınca sinyal işleme konularında sıklıkla kullanılmaktadır. Geleneksel metin tabanlı (text based) programlama dillerinden farklı olarak görsel bir programlama dilidir. Program geliştirirken printf(“hello world”/n); gibi yazı dili kullanmak yerine akış şeması çizilir. Sanal cihaz felsefesinde LabVIEW programları ön panel ve blok çizelgesi olmak üzere iki kısımdan oluşur. Front panel, kullanıcılar tarafından görünen arabirimdir. Bu arabirimin arkasında şemalardan oluşan blok çizelgesi bulunur.

Gerçek zamanlı, FPGA, Mobile, Embedded, Datalogging and Supervisory Control, Vision Development, Motion Control, PID-Fuzzy Logic, Sound and Vibration, Statechart, Simulation gibi ek modülleri bulunmaktadır [28]. Bunlardan PID ve

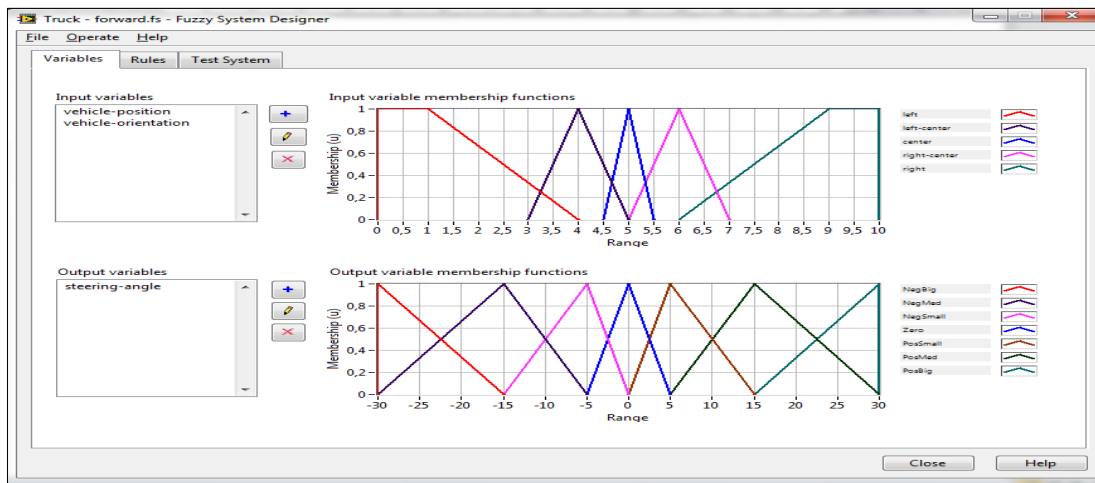
Fuzzy Logic Toolkit aracı PID kontrolörler ve “bulanık mantık uygulamaları geliştirmek için özelleştirilmiştir.

3.2.1. LabVIEW PID-Fuzzy Logic Araç Kutusu

Labview PID ve Fuzzy Logic Toolkit aracı bulanık mantık uygulamaları geliştirmede iki şekilde kullanılabilir. Birincisi Fuzzy System Designer kullanarak ya da Fuzzy Logic VI'lar yardımıyla. Labview programları veya VI (sanal cihazlar)'lar gerçek olmayan belgeler olarak adlandırılır. Çünkü görünüşleri ve çalışmaları osiloskop ve milimetreler gibi gerçek fiziki belgeleri taklit ederler. Bir VI'nın 3 bileşeni vardır:

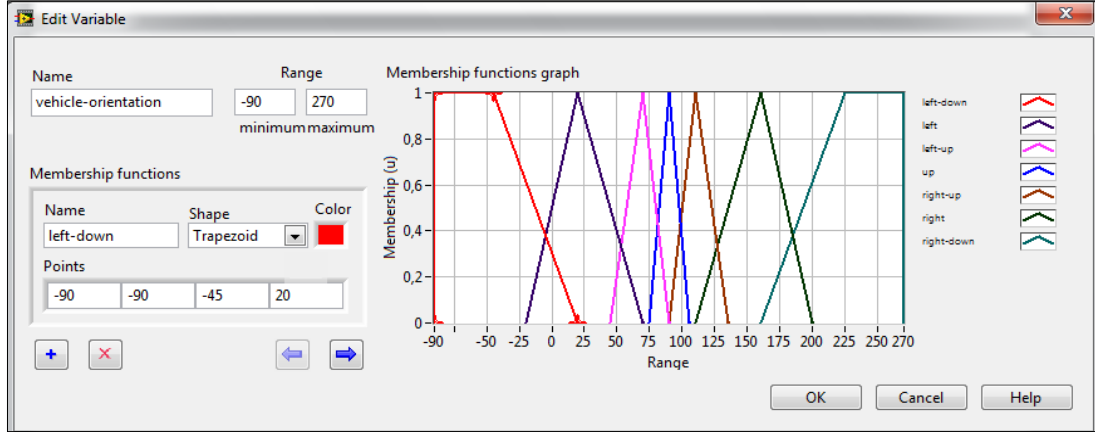
1. Ön panel – Kullanıcı arayüzü gibi hizmet verir.
2. Blok diyagramı – VI fonksiyonlarının tanımlanmasından oluşan kodların grafik kaynaklarını oluşturur.
3. Simge ve bağlantı penceresi – Bir VI' yi başka bir VI içerisinde kullanabildiğin için VI olarak tanımlanır. Diğer VI içinde kullanılan VI, subVI olarak adlandırılır. Bu subVI'larda metin tabanlı programlama dillerindeki alt rutine benzer [28].

Labview programı açıldıktan sonra Tools menüsünden Control Design and Simulation menüsünden Fuzzy System Designer tıklanır ve Fuzzy System Designer açılır. Fuzzy ve PID Toolkit'in Labview programına eklenmiş olması gereklidir. Bulanık mantık tasarım sayfası açıldıktan sonra değişkenlerin girileceği sayfa gelir.



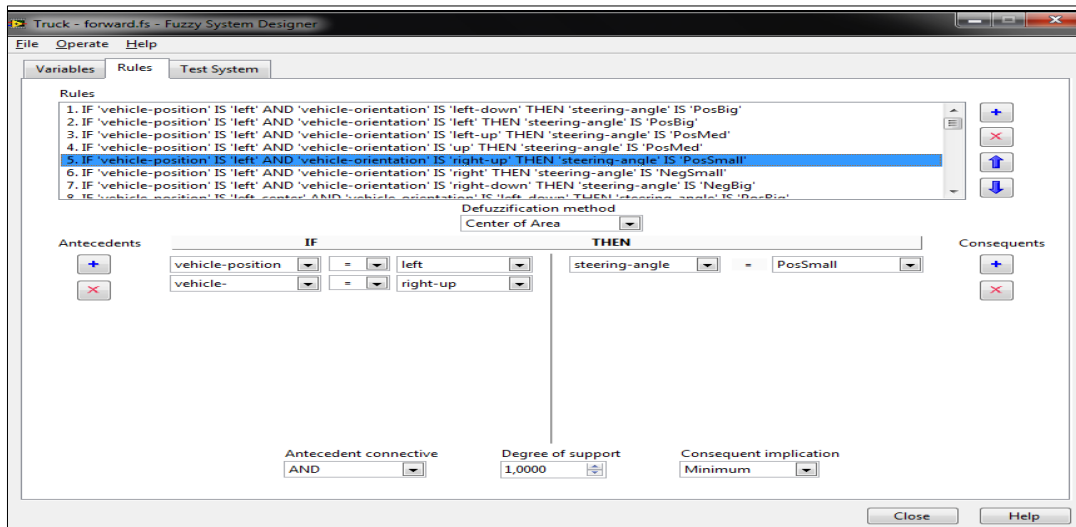
Şekil 3.12. LabVIEW dilsel değişken penceresi

Şekil 3.12’de görülen dilsel değişken penceresinde giriş ve çıkış değişkenleri tanımlanır. Eğer bir değişkenin üyelik fonksiyonları görüntülenmek ve değiştirilmek istenirse değişken ismi üzerinde çift tıklanması gereklidir. Çift tıklandığında Edit Variable penceresi açılır ve değişken değerleri bu kısımdan değiştirilebilir. Şekil 3.13’de terim düzenleme penceresi görülmektedir.



Şekil 3.13. LabVIEW terim düzenleme penceresi

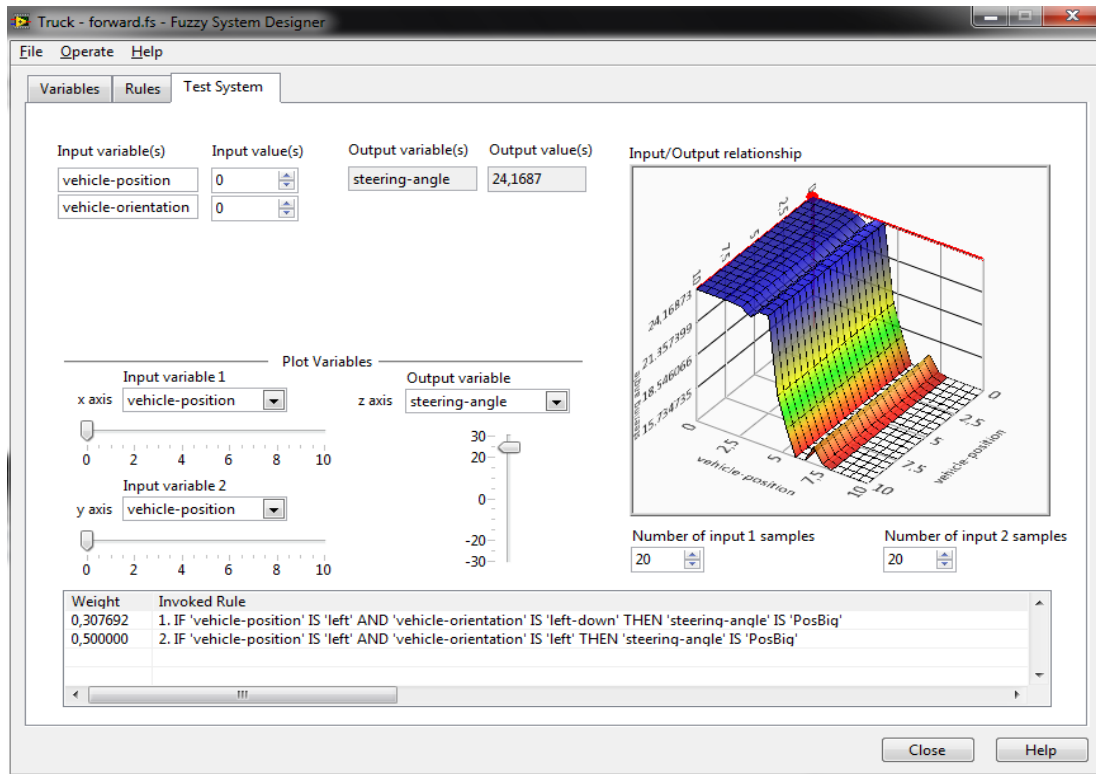
Giriş ve çıkış değişkenleri tanımlandıktan sonra Rules sekmesinden kural veritabanı oluşturulur. Kurallar kelimelerle tanımlanır. Giriş ve çıkış dilsel değişkenleri arasındaki ilişki yine dilsel olarak tanımlanmalıdır. Bulanık sistemdeki kural tabanı giriş değişkenlerine bağlı olarak çıkış değişkenlerini tanımlamaktır. LabVIEW programı kural penceresi Şekil 3.14’deki gibidir.



Şekil 3.14. LabVIEW kural penceresi

Bulanık sistem için kural tabanı oluşturulduktan sonra yapılması gereken bulanık kontrolörün sistem için durulaştırma işlemini nasıl gerçekleştireceğini belirtmektir. Durulaştırma işlemi çıkış dilsel değişkenlerinin üyelik derecelerinin sayısal değerlere çevrilmesi işlemidir. Bu işlem de gerçekleştirildikten sonra bulanık sistem tamamlanmış olur. Ancak sistemin otomatikleştirilmek istenen kontrol yapısına entegre edilmeden önce nasıl çalıştığı test edilebilir.

Bulanık sistemin giriş ve çıkış değerleri arasındaki ilişki, kural tabanının onaylanması için test edilebilir. Bunun için Test System sekmesi kullanılır. Şekil 3.15 LabVIEW ile hazırlanmış sistemin test anındaki görüntüsünü vermektedir.

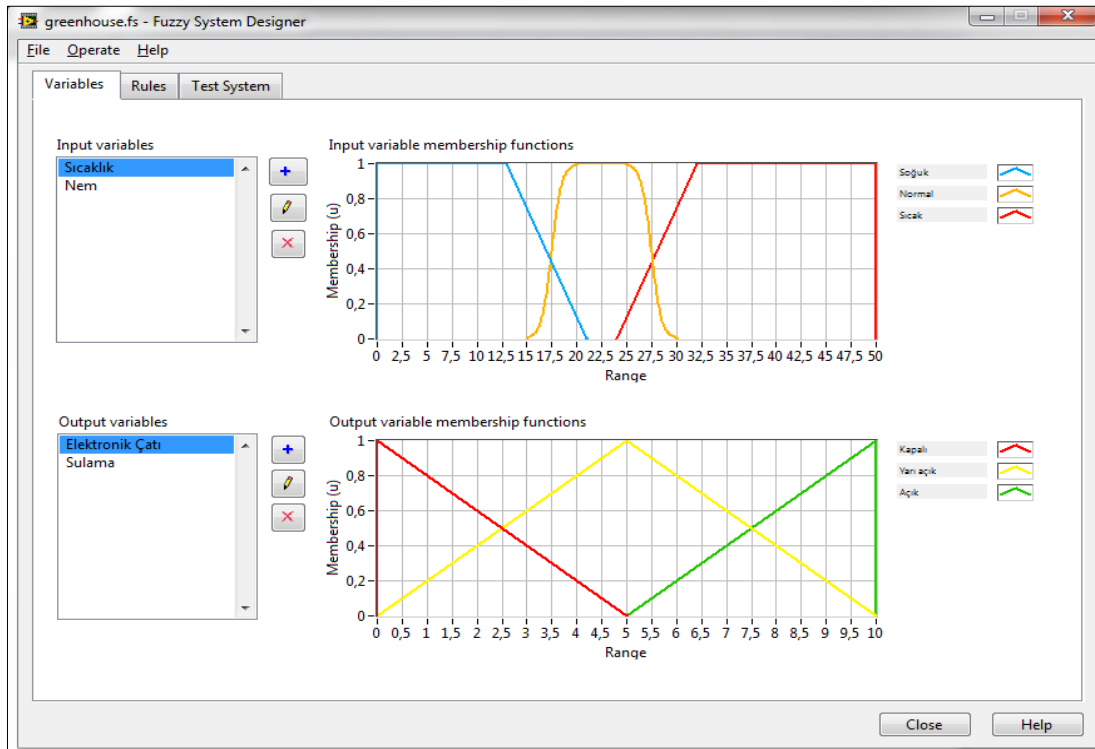


Şekil 3.15. Sistemin test anındaki görüntüsü

Bu kısımda giriş değişkenlerinin değerleri klavyeden girilerek ya da fare ile seçilerek değiştirilebilir. Giriş değişkenlerinin değişmesiyle beraber çıkış değişkenlerinin ve giriş ile çıkış arasındaki ilişkiyi gösteren üç boyutlu şeklin de değiştiği görülür. Invoked Rules kısmında bulanık sistemin hangi kural yada kuralları çağırdığı görülür.

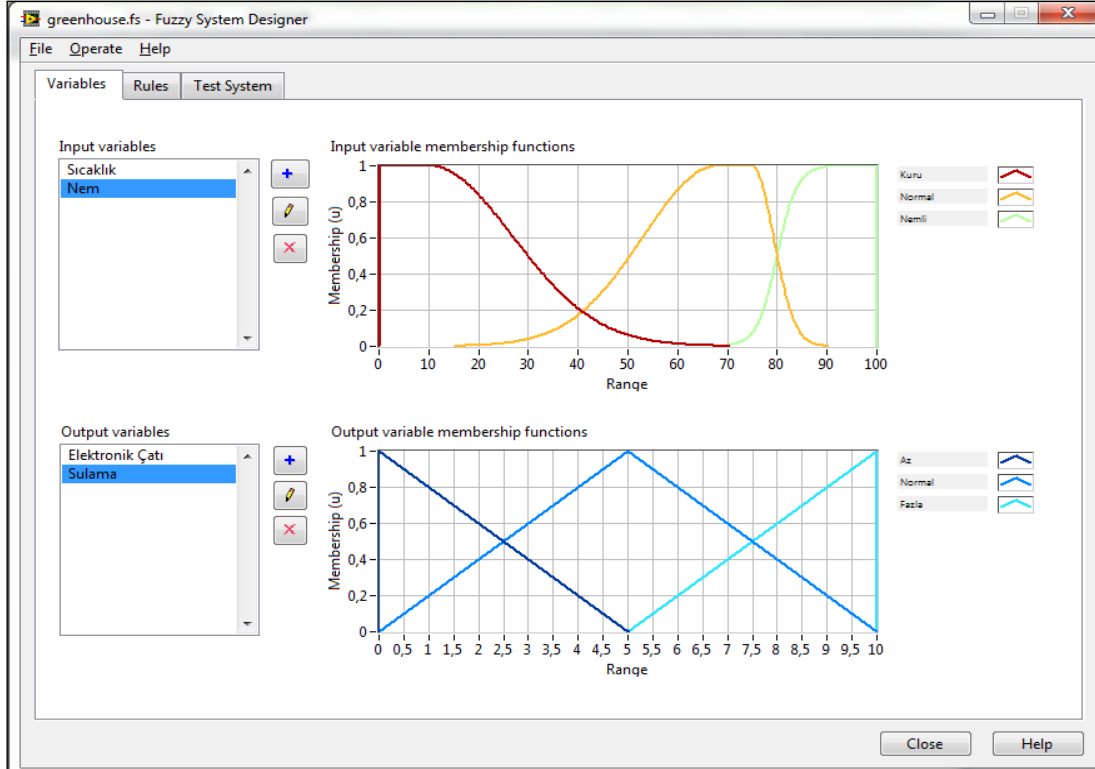
3.2.2. LabVIEW programı ile örnek bir uygulama: Sera Kontrolü

Seralar, bitkilerin yetişmesine uygun şartların sağlanması amacı ile çevre şartları kontrol edilebilen veya düzenlenebilen cam, plastik, fiberglas gibi ışığı geçiren materyallerle örtülü yapı elemanlarıdır [28]. Seralarda doğal olan koşullarla beraber sağlanan koşullar da vardır ve bunlara göre bitkilerin verimliliği artırılabilir. Seralarda bulanık mantık uygulaması gerçekleştirilirken giriş değişkenleri olarak sıcaklık ve nem alınabilir. Verilen bu sıcaklık ve nem değerlerine göre yapay koşullarda iyileştirme yapılabilir. Bu yapay koşullar çatının açılıp açılmaması ve sulamanın olup olmayacağıdır. Bu uygulamada da giriş olarak sıcaklık ve nem; çıkış olarak ise elektronik çatının açılma durumu ve sulamanın ne kadar olacağıdır. Oluşturulan sistemin ilk giriş değişkeni sıcaklık değeridir. Sıcaklık değişkeni 0 ile 50 arasında değer alabilir. Terimleri: Soğuk, Normal ve Sıcaktır. Çıkış değişkenlerinin ilki olan elektronik çatı değişkeni ise Kapalı, Yarı Açık ve Açık değerlerini alır. Sistemin giriş ve çıkış değişkenleri belirlenmiş ve Şekil 3.16'da görüldüğü gibi girilmiştir.



Şekil 3.16. Sistemin sıcaklık giriş ve elektronik çatı çıkış değişkenlerinin belirlenmesi

İkinci giriş değişkeni Nemdir ve 0 dan 100 e kadar değer alabilir. Kuru, Normal ve Nemli dilsel değişkenleri ile ifade edilir. Çıkış değişkeni ise Sulamadır ve Az, Normal ve Fazla değişkenlerini alır. Nem giriş ve Sulama çıkış değişkenleri Şekil 3.17’deki gibi girilmiştir.



Şekil 3.17. Sistemin nem giriş ve sulama çıkış değişkenlerinin belirlenmesi

Oluşturulan sistemin kuralları ise şu şekildedir:

Eğer SICAKLIK “Soğuk” ve NEM “Kuru” ise ELEKTONİK ÇATI “Kapalı”
SULAMA “Normal”;

Eğer SICAKLIK “Soğuk” ve NEM “Normal” ise ELEKTONİK ÇATI “Kapalı”
SULAMA “Normal”;

Eğer SICAKLIK “Soğuk” ve NEM “Nemli” ise ELEKTONİK ÇATI “Kapalı”
SULAMA “Normal”;

Eğer SICAKLIK “Normal” ve NEM “Kuru” ise ELEKTONİK ÇATI “Yarı Açık”
SULAMA “Normal”;

Eğer SICAKLIK “Normal” ve NEM “Normal” ise ELEKTONİKÇATI “Yarı Açık”
SULAMA “Normal”;

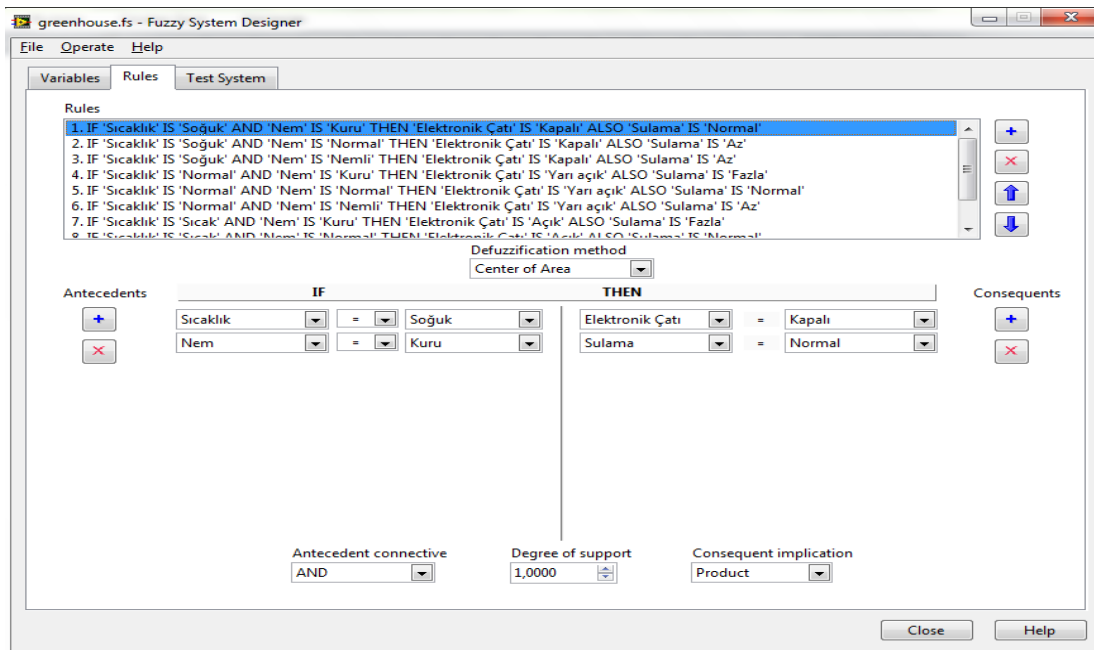
Eğer SICAKLIK “Normal” ve NEM “Nemli” ise ELEKTONİK ÇATI “Yarı Açık”
SULAMA “Normal”;

Eğer SICAKLIK “Sıcak” ve NEM “Kuru” ise ELEKTONİK ÇATI “Açık”
SULAMA “Normal”;

Eğer SICAKLIK “Sıcak” ve NEM “Normal” ise ELEKTONİK ÇATI “Açık”
SULAMA “Normal”;

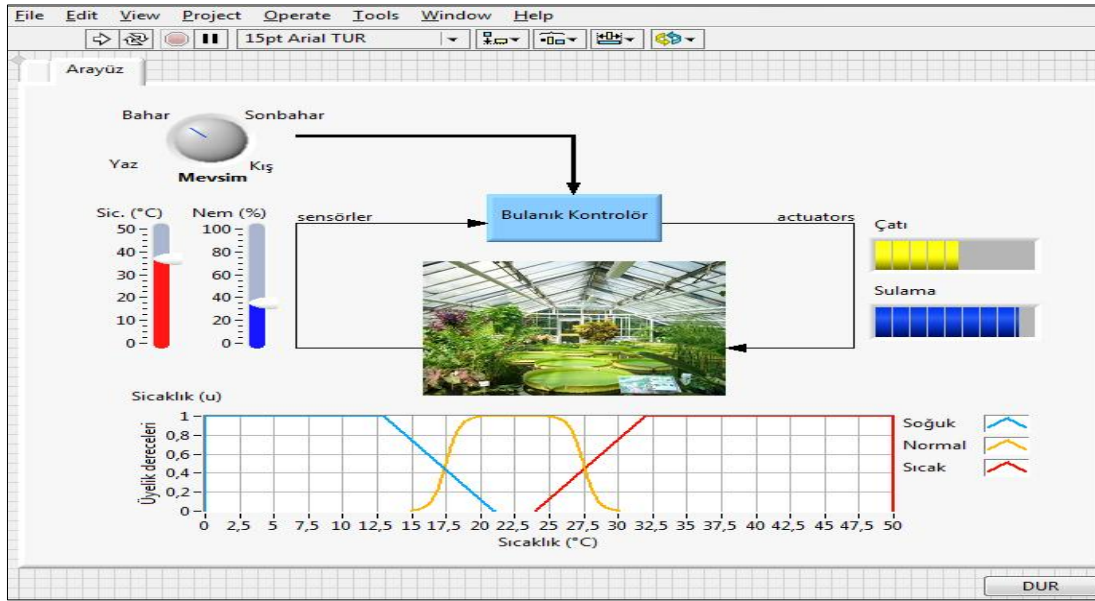
Eğer SICAKLIK “Sıcak” ve NEM “Nemli” ise ELEKTONİK ÇATI “Açık”
SULAMA “Normal”;

Oluşturulan bu kuralların kural penceresindeki görünümü Şekil 3.18’deki gibidir.



Şekil 3.18. Sistemin kural penceresindeki görünümü

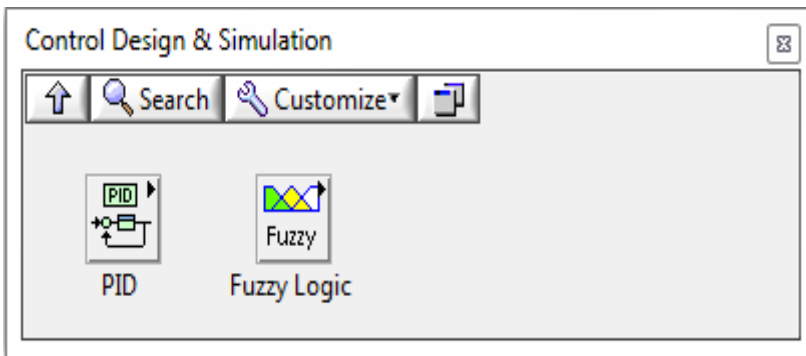
Sistemin test anındaki arayüz görüntüsü ise Şekil 3.19’da görüldüğü gibidir.



Şekil 3.19. Sistemin grafik kullanıcı arayüzü

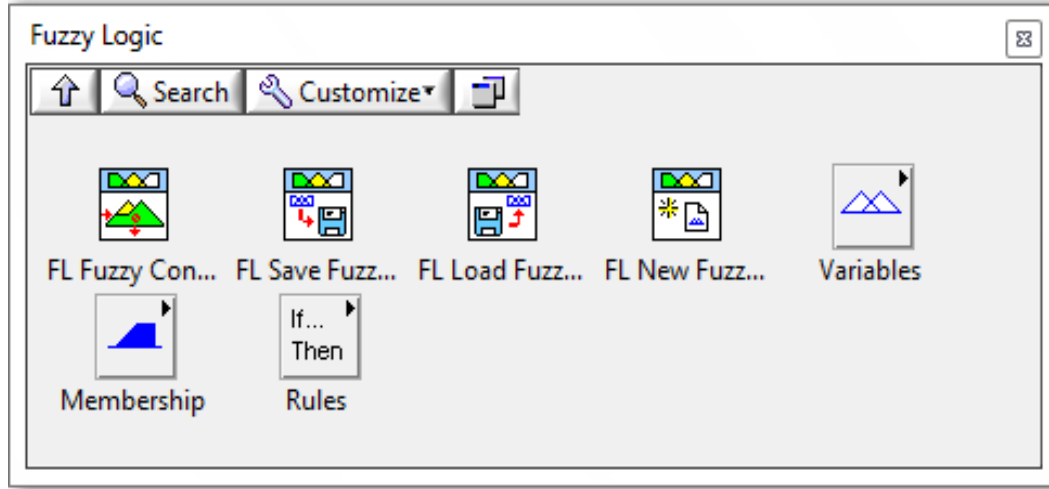
3.2.3. LabVIEW programının diğer özellikleri

LabVIEW programı VI denilen alt programlardan oluşur. LabVIEW programını diğer programlardan ayıran en büyük özelliği gerçek bir cihaz kullanılıyor hissi veren önyüz tasarımlarıdır. LabVIEW programına PID ve Fuzzy Toolkit yüklendikten sonra blok diyagram üzerinde bulanık mantık araçlarını kullanabilmek için Control Design & Simulation menüsünden bulanık mantık araçlarını seçmek gereklidir. Bu araçlar Şekil 3.20’de görüldüğü gibi eklenmekte ve buradan seçilebilmektedir.



Şekil 3.20. LabVIEW kontrol dizayn ve simülasyon araçları

Buradan Fuzzy Logic simgesi tıklandığında alt seçenekler gelir. (Şekil 3.21)



Şekil 3.21. LabVIEW bulanık mantık araçları

Bunlar blok diyagrama yerleştirilerek bulanık sistemin kullanıcı arayüzü tasarlanır. LabVIEW programının üstün özelliklerinden birisi de var olan örnek sistemleri değiştirilebilmektedir. Çok sayıda örnek uygulaması olan LabVIEW programı üstün yardım menüleri sayesinde görsel programlama konusunda başarılı olmaktadır. Bulanık mantık sistem tasarımı için özelleştirilmiş olmasa da LabVIEW, bulanık mantık sistemleri tasarlamak ve kullanmak konusunda başarılı bir programdır.

Windows için NI LabVIEW PID & Fuzzy Logic Toolkit ile şunlar yapılabilir:

- PI, PD ve PID kontrol algoritmaları LabVIEW uygulamaları içerisine entegre edilebilir.
- Fuzzy System Designer ve Fuzzy Logic VI lar kullanılarak bulanık sistemlerin tasarım, adaptasyon ve kontrolü gerçekleştirilebilir.
- Kontrol performansı artırmak için farklı algoritmalara dayalı online ve offline kazançlar sağlanır.

İlk olarak Nisan 1983'te başlayan LabVIEW çalışmaları çok sayıda yenilik ve sürümlerle günümüze kadar ulaşmıştır. LabVIEW programının sürümleri Tablo 3.3'de verilmiştir.

Tablo 3.3. LabVIEW programının sürümleri

İsim- Versiyon	Tarih
LabVIEW projesinin başlangıcı	Nisan 1983
LabVIEW 1.0 (Macintosh için)	Ekim 1986
LabVIEW 2.0	Ocak 1990
LabVIEW 3.0 (Çok platformlu)	Temmuz 1993
LabVIEW 4.0	Nisan 1996
LabVIEW 4.1	Mayıs 1997
LabVIEW 5.0	Şubat 1998
LabVIEW RT (Gerçek Zamanlı)	Mayıs 1999
LabVIEW 6.0 (6i)	Temmuz 2000
LabVIEW 7.0 (Ekspres)	Nisan 2003
LabVIEW 8.0	Eylül 2005
LabVIEW 2009 (32 ve 64-bit)	Ağustos 2009
LabVIEW 2010	Ağustos 2010
LabVIEW 2011	Haziran 2011
LabVIEW 2012	Ağustos 2012

3.3. MATLAB FIS Bulanık Mantık Model Geliştirme Ortamı

Matlab Mathworks firması tarafından geliştirilmiş olan dördüncü nesil bir programlama dilidir. Matlab kelimesi Matrix Laboratory kelimelerinin kısaltmasıdır ve matris laboratuvarı anlamına gelir. Matlab'ın simülasyon aracı Simulink'tir. Matlab ile bulanık mantık işlemlerini yapabilmek için Matlab içerisine Fuzzy Logic Toolbox'ın yüklenmesi gereklidir [29].

Fuzzy Logic Toolbox, bulanık mantık tabanlı sistemler için Matlab fonksiyonları, grafik araçları ve analiz, dizayn ve simülasyon yapılmasına olanak veren Simulink bloklarını Matlab içerisine ekler. Bulanık mantıkla sonuç elde etmek için tasarım yapılırken her aşamada rehberlik eden uygulamaları vardır. Fonksiyonları yaygın

olarak kullanılan bulanık kümeleme ve adaptif sinirsel bulanık çalışmaları gibi çok sayıda metotları destekler.

Matlab Fuzzy Logic Toolbox, karmaşık sistem davranışlarının basit mantık kurallarıyla modellenmesi ve daha sonra bu kuralların uygulanması için bir bulanık çıkarım sistemi oluşturulmasına izin verir. Aynı zamanda araç kutusunu bağımsız bir bulanık mantık aracı olarak da kullanmak mümkündür. Matlab 'da alternatif olarak Simulink içerisinde bulanık çıkarım blokları kullanılabilir ve bulanık sistem dinamik sistemin kapsamlı bir modeli olarak simüle edilebilir.

Matlab Fuzzy Logic Toolbox ile şunlar yapılabilir:

- a) Bulanık çıkarım sistemleri için özelleştirilmiş GUI 'ler (Graphics User Interface – Grafik Kullanıcı Arayüz) oluşturma, bunları izleme ve sonuçları analiz etme.
- b) Bulanık çıkarım sistemleri meydana getirmek için üyelik fonksiyonları oluşturma.
- c) AND, OR ve NOT mantıksal ve kullanıcı tanımlı kuralları destekleme
- d) Standart Mamdani ve Sugeno tipteki bulanık çıkarım sistemleri oluşturabilme
- e) Simulink model içerisine bulanık çıkarım sistemini ekleme
- f) Başka yerlerde kullanılmak istenirse eğer C kodları üretme veya kendi çalışabilen .exe uzantılı bulanık sistemler oluşturma

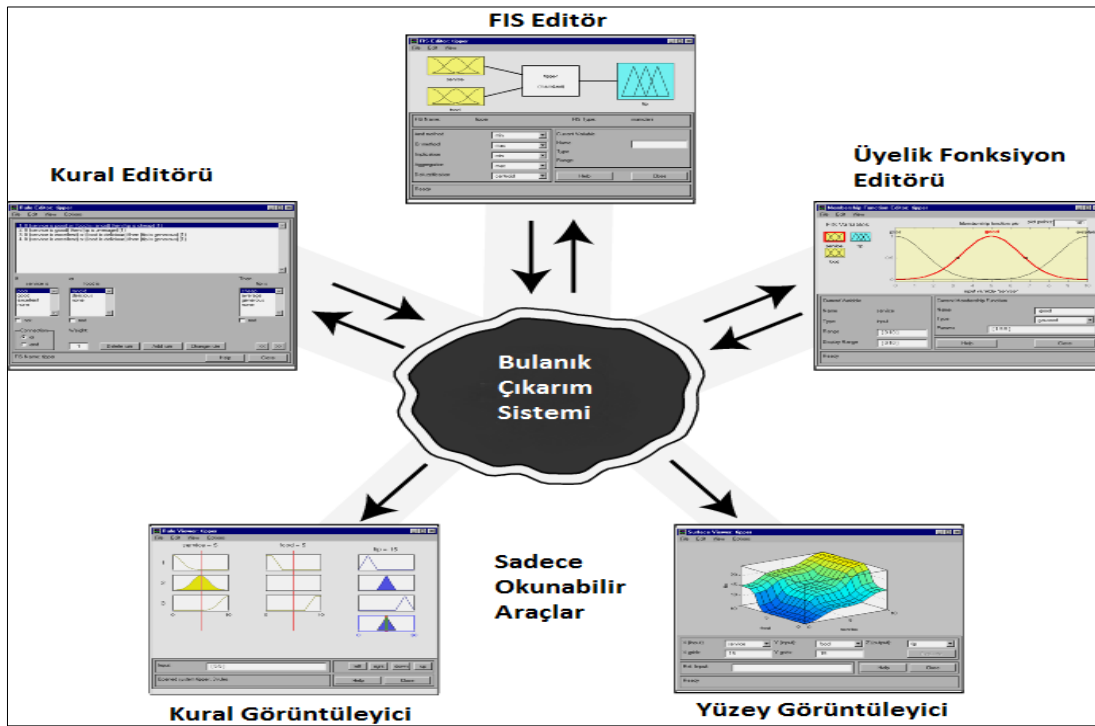
3.3.1. Matlab FIS (Bulanık Çıkarım Sistemi) editörleri

Matlab programında bulanık mantık tasarımında GUI 'ler kullanılır. Bu GUI 'ler aracılığı ile klasik bulanık sistem geliştirme ve model tanımlamak mümkündür. Bulanık çıkarım sistemleri analiz ve geliştirmek, sinirsel bulanık çıkarım sistemleri oluşturmak ve bulanık kümeleme yapmak için Fuzzy Logic Toolbox kullanılır. Bu Toolbox 'ta bulunan bileşenler şunlardır:

- 1- FIS Editörü (Fuzzy Inference System – Bulanık Çıkarım Sistemi)
- 2- Üyelik Fonksiyon Editörü (Membership Function Editor)
- 3- Kural Editörü (Rule Editor)
- 4- Kural Görüntüleyici (Rule Viewer)
- 5- Yüzey Görüntüleyici (Surface Viewer)

Tüm bu açıklanan GUI' ler birbirine dinamik olarak bağlıdır. FIS üzerinde yapılan herhangi bir değişiklik aynı anda tüm GUI' leri değiştirir ve bu değişim aynı anda görülebilir. Örneğin eğer Üyelik Fonksiyon Editörü'nde üyelik fonksiyonlarının isimleri değiştirilirse bu değişiklik aynı anda Kural Editörü penceresinde de görülür. İstenen sayıda FIS sistemi için aynı anda çok sayıda editör yada tüm editörler açık olabilir.

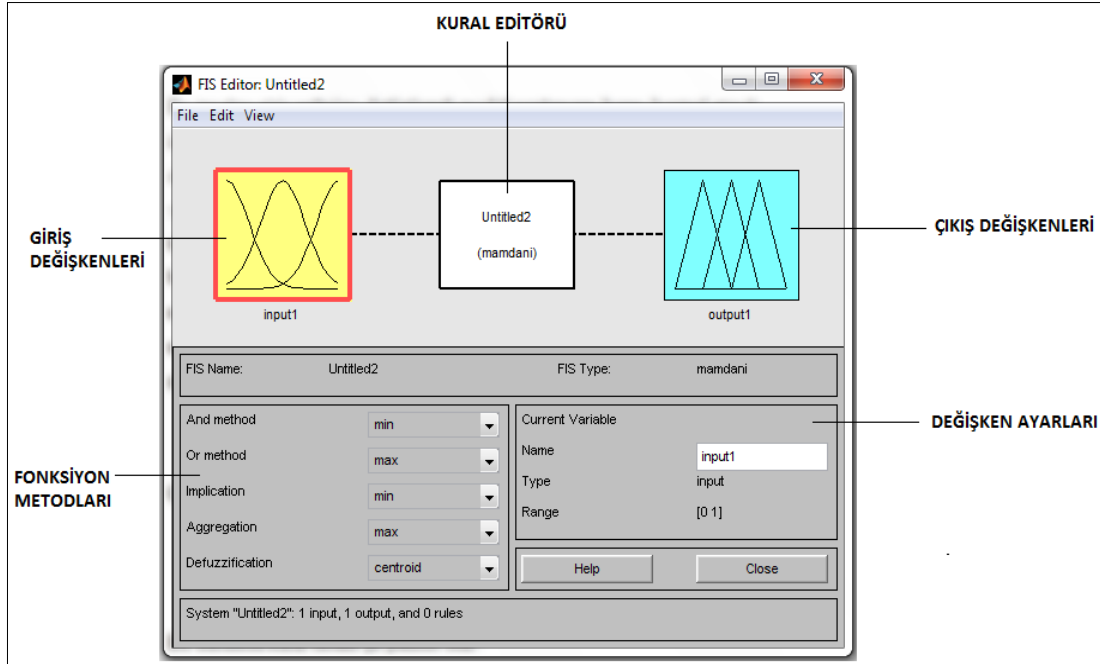
Matlab programında bulanık mantık sisteminin blok yapısı Şekil 3.22'de gösterilmiştir:



Şekil 3.22. MATLAB bulanık mantık blok yapısı

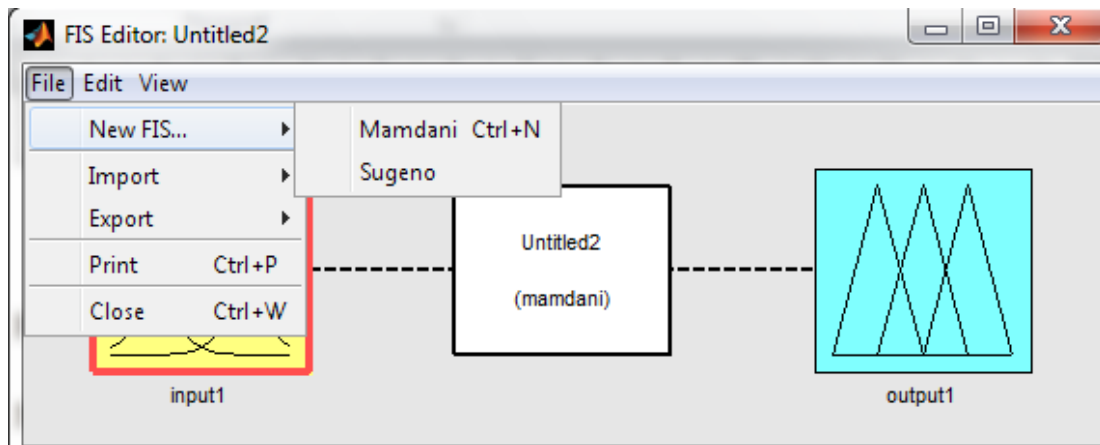
3.3.1.1. Bulanık çıkarım sistemi (FIS) editörü

Sistem için üst düzey sorunları gideren sistemdir. “Kaç tane giriş ve çıkış değişkeni olacak?” , “Bunların isimleri nelerdir?” gibi soruların yanıtını verir. Fuzzy Logic Toolbox yazılımı giriş sayısını sınırlandırmaz. Ancak girişlerin sayısı tasarımın yapıldığı cihazın belleğiyle sınırlandırılabilir. Eğer girişler çok büyükse yada üyelik fonksiyonlarının sayısı çok fazlaysa, o zaman da diğer GUI araçlarıyla FIS'i analiz etmek zor olabilir. Şekil 3.23 MATLAB kural editör penceresini göstermektedir.



Şekil 3.23. MATLAB kural editör penceresi

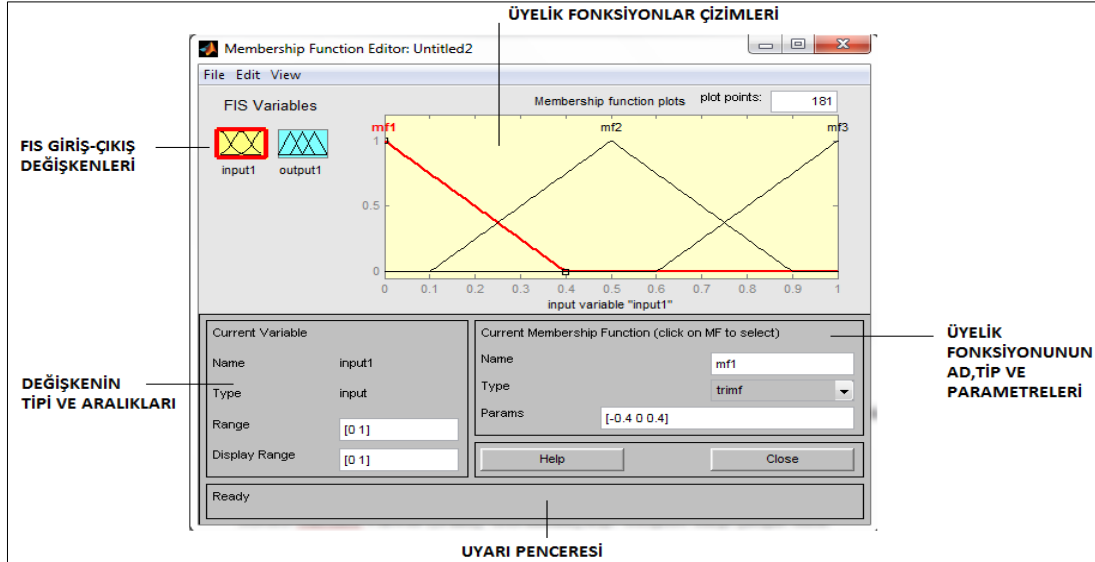
FIS Editor penceresinde 3 tane menü vardır. File, Edit ve View Menüleri. File menüsü altında New FIS seçeneği vardır ve bu seçenek seçildiğinde Sugeno ve Mamdani seçenekleri gelir. Bunun nedeni Matlab'da iki farklı modelleme yapısı olmasındandır. Bunların farkı ise Sugeno girilen girişlere göre sabit yada lineer bir çıkış üretirken, Mamdani metodunda çıkışlar nonlineer fonksiyonlar olabilir. Mamdani sonucu, ağır hesaplamalarla gerçeğe daha yakın bir şekilde oluşturur [30]. MATLAB FIS Editör Penceresi Şekil 3.24'de görüldüğü gibidir.



Şekil 3.24. MATLAB FIS editör penceresi

3.3.1.2. Üyelik fonksiyon editörü

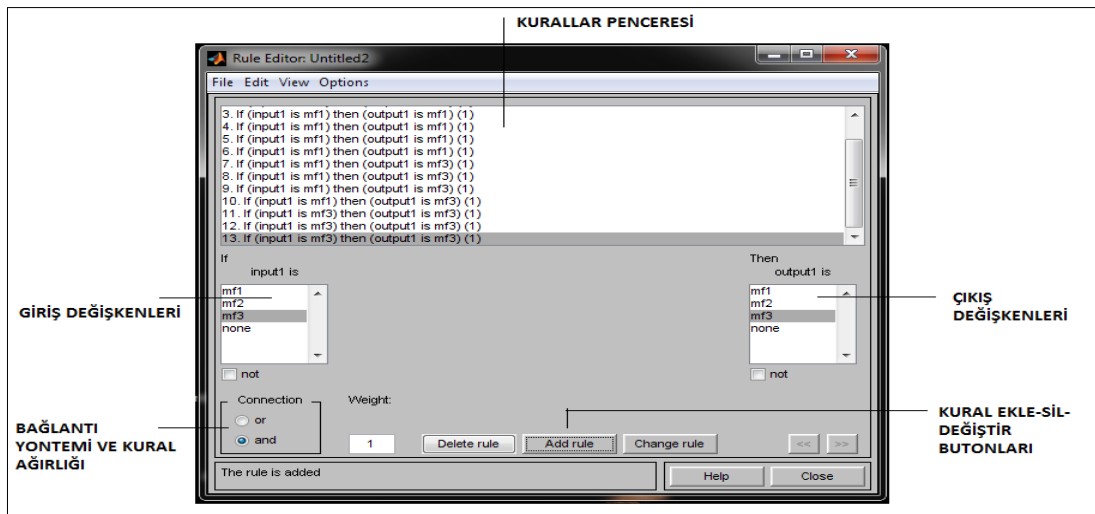
Her bir değişken ile bağlantılı tüm üyelik fonksiyonlarının şekillerini tanımlar. Üyelik fonksiyon editörünün yapısı Şekil 3.25’de görülmektedir.



Şekil 3.25 MATLAB üyelik fonksiyon editörü penceresi

3.3.1.3. Kural editörü

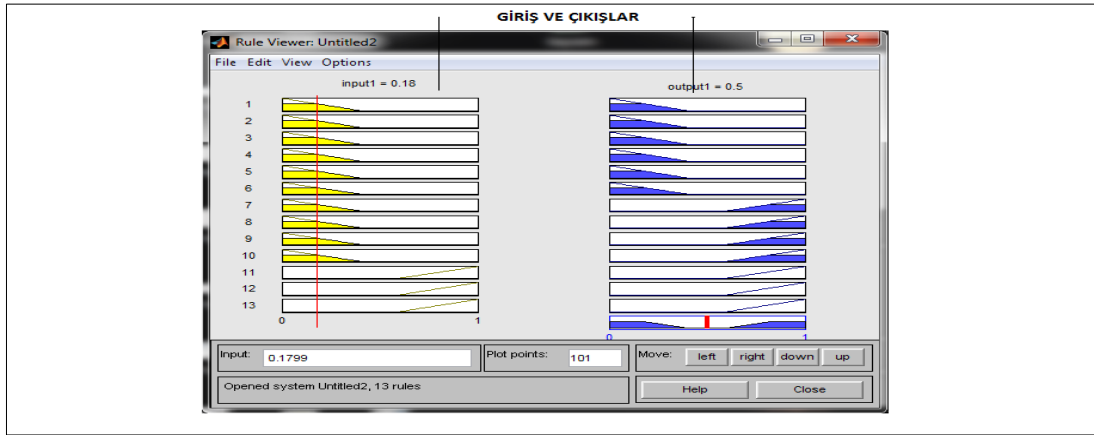
Sistemin davranışını tanımlayan kuralların listesini düzenlemek için kullanılır. MATLAB kural editör penceresi Şekil 3.26’da görülmektedir.



Şekil 3.26. MATLAB kural editörü penceresi

3.3.1.4. Kural görüntüleyici

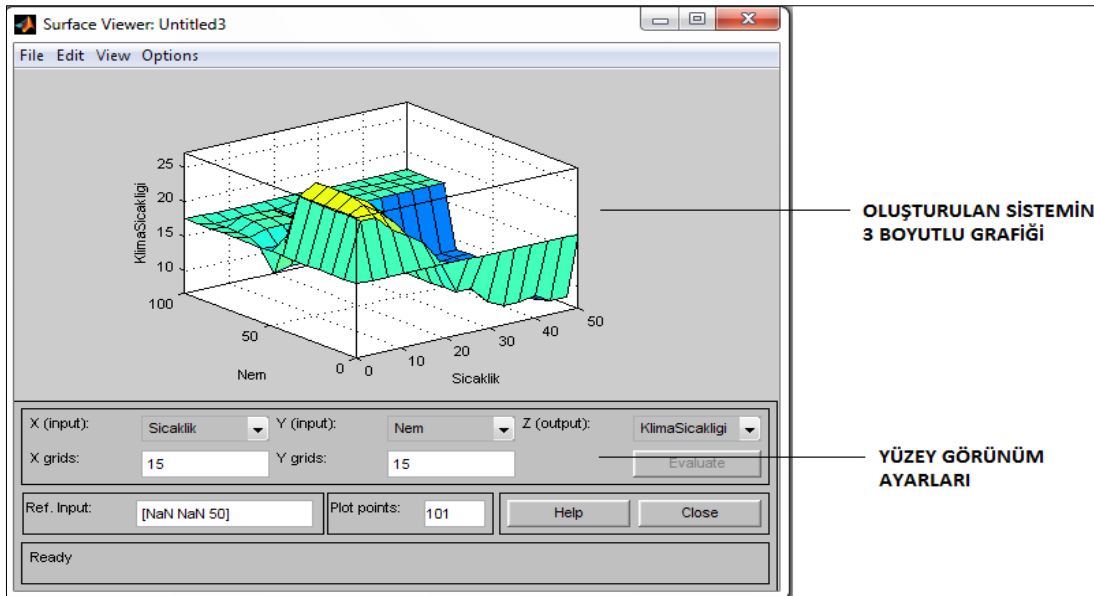
Bulanık çıkarım diyagramını görmek için kullanılır. Bu araç sayesinde hangi kuralların aktif olduğu ya da sonuçları etkileyen üyelik fonksiyonları görülebilir. Kural görüntüleyici penceresi Şekil 3.27’de görüldüğü gibidir.



Şekil 3.27 MATLAB kural görüntüleyici penceresi

3.3.1.5 Yüzey görüntüleyici

Girişlere karşı çıkışın nasıl bağlantıda olduğunu görmeye yarayan araçtır. Sistemin yüzey haritasını çıkarır. Şekil 3.28 yüzey görüntüleyici penceresini göstermektedir.



Şekil 3.28 MATLAB yüzey görüntüleyici penceresi

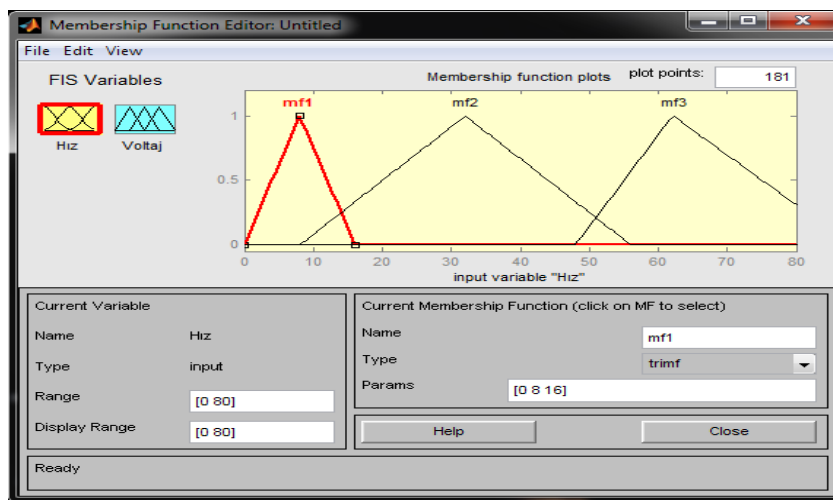
3.3.2. Matlab ile örnek bir uygulama: Model Uçak Hız Kontrolü

Bu örnekte giriş voltajını değiştirerek model uçağımızın hızını kontrol etmek istiyoruz. Uçağın hızı için bir ayar noktası belirlenir. Eğer herhangi bir nedenden dolayı model uçak daha hızlı çalışırsa giriş voltajını azaltarak uçağın hızını yavaşlatmak gerekir. Çünkü yüksek hızda model uçağın kontrolü kaybedilebilir ve böylece bir yere çarpmasına sebep olunabilir. Bu durumda da eğer uçak başta belirlenen ayar hızından daha düşük bir hıza ulaşırsa, giriş voltajı artırılmalıdır ki model uçağın hızı ayar noktasına ulaşsın. Çünkü bu durumda da hız çok düştüğünden model uçağımızın yere düşme riski olabilir. Böyle bir problem için giriş değerleri şöyle olmalıdır: Yavaş, Doğru, Hızlı. Çıkış değerlerinin dilsel ifadesi ise şu şekildedir: Az voltaj (Yavaşlaması için), Değişiklik yok ve Yüksek Voltaj (Hızını artırmak için)

Bu durumda kural tabanı şu şekilde olur:

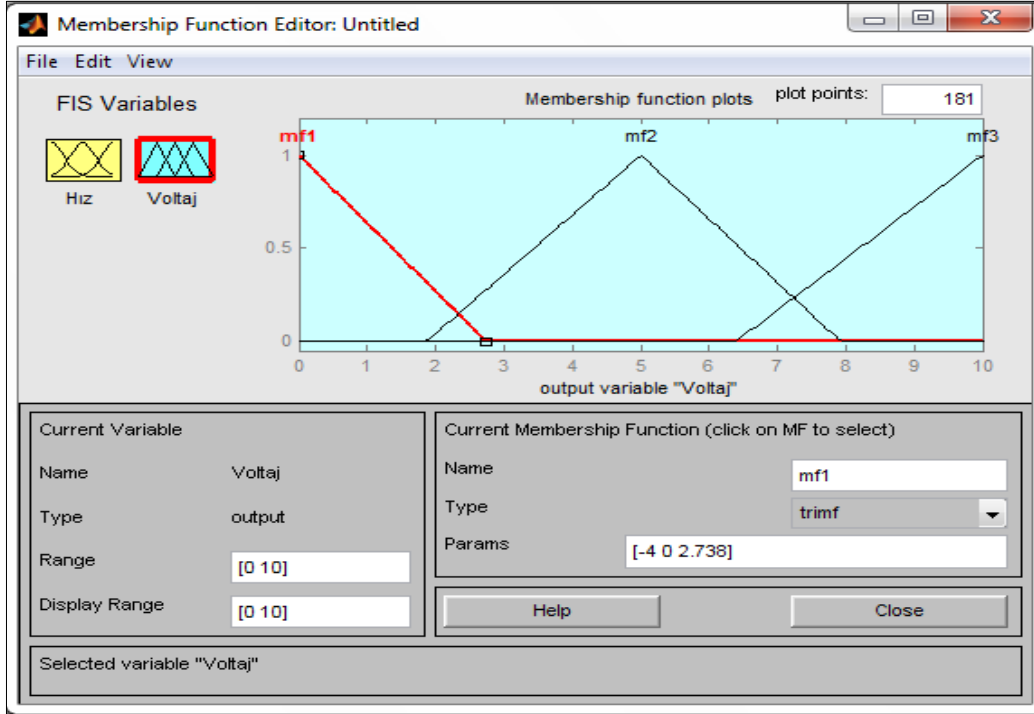
- 1- Eğer uçak çok yavaşsa daha fazla voltaj ver
- 2- Eğer uçak doğru hızdaysa değişiklik yapma
- 3- Eğer uçak hızlıysa daha az voltaj ver

Böyle bir örnek için Matlab editörüne ilk olarak “fuzzy” komutu yazılarak FIS Editor açılır. Membership Function Editor’ de oluşturulan sistemin giriş değişkeni Hız Şekil 3.29’da görüldüğü gibi tanımlanmıştır.



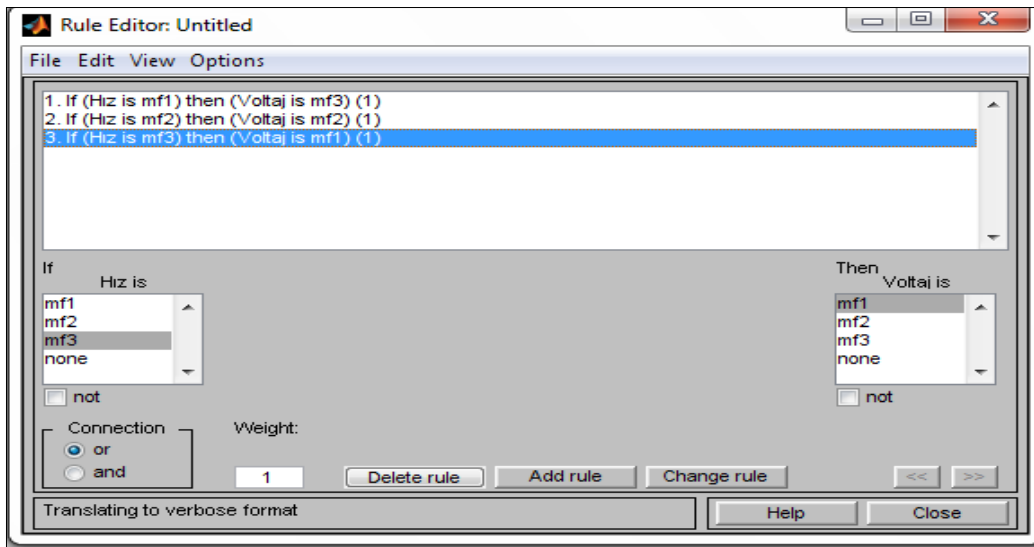
Şekil 3.29. Tasarlanan sistemin MATLAB üyelik fonksiyon editör penceresi

Çıkış üyelik fonksiyonu olan Voltaj değişkeni ise Şekil 3.30'da görüldüğü gibi tanımlanmıştır.



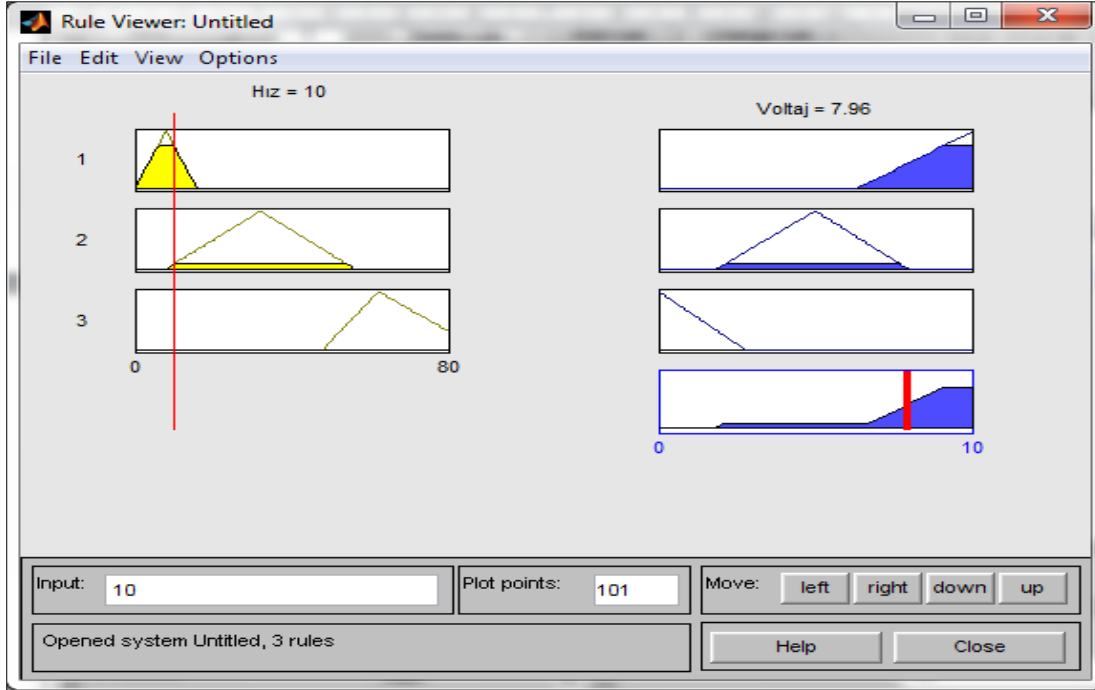
Şekil 3.30. Tasarlanan sistemin giriş ve çıkış değişkenleri

Giriş ve çıkış değişkenleri belirlendikten sonraki aşama ilk başta belirlenen kuralları Rule Editor kısmında yazmaktır. Bu da Şekil 3.31'deki gibi gerçekleşir:



Şekil 3.31. Tasarlanan sistemin kuralları

Oluşturulan sistem için sonuçlar değerlendirilmek istenirse düşük hız için yüksek voltaj ve yüksek hız için düşük voltaj verilmesi gerektiği görülür. Sistemin simülasyon anındaki görüntüsü Şekil 3.32’deki gibidir.



Şekil 3.32. Tasarlanan sistemin çalıştırılması ve çalışma anındaki görüntüsü

Burada örneğin “Hız:10 değerini aldığı anda voltaj 7.96 olmalıdır. Hız 52 değerini aldığı anda voltaj 3,38 olmalıdır.” sonuçları alınır.

Çalışma tamamlandıktan sonra diske kaydedilebilir. Bulanık sistem diske kaydedilirken uzantısı .fis olacak şekilde kaydedilir ve bu dosya ASCII metin dosyası şeklindedir. Bu metin dosyası düzenlenebilir, değiştirilebilir ve anlaşılması kolaydır. Matlab görsel bir programlama dili olduğu kadar yapısal bir programlama dili özelliğini de taşır. Matlab’ da komut satırı kullanılarak da bulanık mantık uygulamaları yapmak mümkündür. Yukarıda yaptığımız örneği model.fis olarak kaydetmiş olalım. Bu dosyayı sistemden yüklemek için Matlab komut satırına

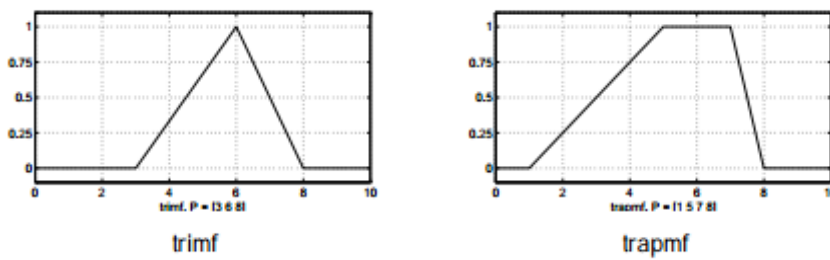
>> a = readfis('model.fis'); yazılması yeterlidir. Eğer dosya metin halinde görüntülenmek isteniyorsa bu durumda komut satırına,

>> type model.fis yazılarak dosya metin dosyayı halinde açılır. En çok kullanılan komutlar şunlardır:

- a) fuzzy(a): FIS Editor açmak için kullanılır.
- b) mfedit(a): Membership Function Editor açmak için kullanılır.
- c) ruleedit(a): Rule Editor açmak için kullanılır.
- d) ruleview(a): Inference Viewer açmak için kullanılır.
- e) surfview(a): Surface Viewer açmak için kullanılır.

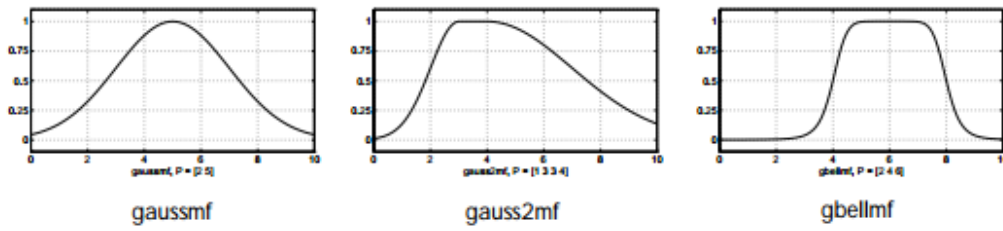
3.3.3. Matlab FLT' da Bulunan Üyelik Fonksiyon Tipleri

Matlab FLT (Fuzzy Logic Toolbox) içerisinde yerleşik 11 tane üyelik fonksiyonu vardır. Bu fonksiyonlar lineer fonksiyon türleri, Gauss dağılım fonksiyonları, sigmoid eğri, kuadratik ve kübik polinom eğrileri şeklindedir. En basit üyelik fonksiyonları düz çizgiler kullanılarak yapılır. Bunlar üçgen üyelik fonksiyonları yada trapez şeklinde olabilmektedir. Üçgensel olan üyelik fonksiyonlarının kod isimleri trimf, trapez şeklinde olanları ise trapmf 'dir. Şekil 3.33'de üçgensel üyelik fonksiyonları görülmektedir.



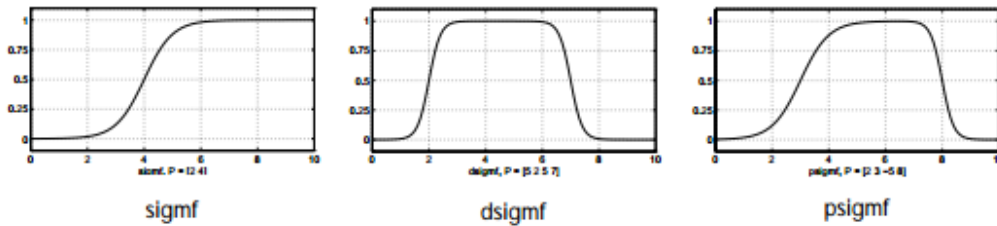
Şekil 3.33. MATLAB programında üçgensel üyelik fonksiyonları

Gauss dağılım eğrisi üzerine inşa edilen iki üyelik fonksiyonu vardır. Bunlar basit Gauss eğrisi ve iki farklı Gauss eğrisi arasında iki yönlü bileşik eğriler. Bunların kod isimleri gaussmf ve gauss2mf dir. Bir diğer Gauss eğrisi tipi çan üyelik fonksiyonu şeklinde ve üç parametre tarafından belirtilir. Kod ismi gbellmf dir. Şekil 3.34'de bu üç fonksiyon görülmektedir.



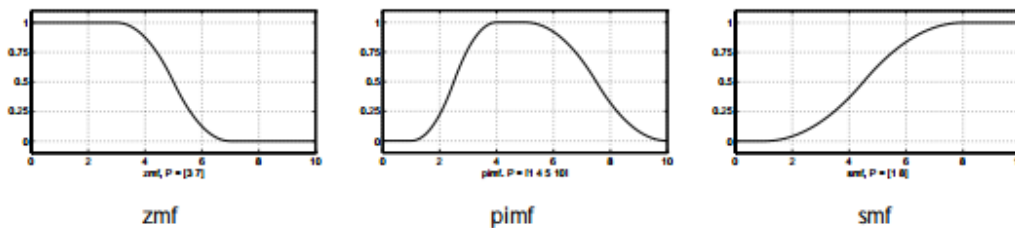
Şekil 3.34. MATLAB programında Gauss üyelik fonksiyonları

Gauss üyelik fonksiyonları ve çan üyelik fonksiyonları bazı uygulamalarda önemli işlev gören asimetrik üyelik fonksiyonlarını belirtebilir. Bu nedenle sol veya sağ tarafı açık olan sigmoid üyelik fonksiyonları tanımlanmıştır. Bu sigmoidal fonksiyonlar `sigmf`, `dsigmf` ve `psigmf` isimlerini alır. MATLAB programında sigmoidal üyelik fonksiyonları Şekil 3.35’de görüldüğü gibidir.



Şekil 3.35. MATLAB programında sigmoidal üyelik fonksiyonları

Polinom tabanlı üyelik eğri fonksiyonları Toolbox içerisinde yüküdür. Oluşturduğu şekilden dolayı isim alan üç tane fonksiyon vardır. Bunlar Z, S ve Pi eğrileridir. Zmf fonksiyonu sola açık polinomal eğri oluşturur. Smf sağa açık, Pmf ise ortasında bir artış ve iki ucunda sıfır olan bir fonksiyondur. Bu fonksiyonların görünümü Şekil 3.36’da görüldüğü gibidir.



Şekil 3.36. MATLAB programında polinomal üyelik fonksiyonları

3.3.4. MATLAB programının sürümleri

MATLAB programı çok yaygın kullanıldığı için sürümleri de fazladır. 1984 yılında ilk sürüm piyasaya çıkmıştır ve sürekli olarak yeni sürümler çıkmaktadır. Tablo 3.4’de görüldüğü gibi MATLAB programının çok sayıda sürümü bulunmaktadır.

Tablo 3.4. MATLAB programının sürümleri

Sürüm	Yılı	Sürüm Özellikleri
MATLAB 1.0	1984	İlk sürüm.
MATLAB 2	1986	Güncelleştirilmiş ikinci sürüm.
MATLAB 3	1987	
MATLAB 4	1992	Yeni kütüphanelerle zenginleştirme çalışmaları yapılmıştır.
MATLAB 5.0	1996	
MATLAB 6.0	2000	Bundled Java machine(JVM)'nin ilk sürümü.
MATLAB 7	2004	Windows 2000 için son sürüm.
MATLAB 7.5	2007	PowerPC, Mac ve Windows 2000 için son sürüm. Windows Vista için lisans sunucu desteği
MATLAB 7.8	2009	32-bit ve 64-bit Microsoft Windows 7 için ilk sürüm.
MATLAB 7.9	2009	Intel 64-bit Mac için ilk sürüm. SPARC ve Solaris için son sürüm.
MATLAB 7.10	2010	Intel 32-bit Mac için son sürüm.
MATLAB 8.1	2013	Son sürüm.

BÖLÜM 4. SUNUCU ODALARININ SOĞUTULMASI İÇİN BULANIK MANTIK TABANLI SOĞUTUCU TASARIMI

Bilgisayarların ısınması bilgisayar kullanıcılarının karşı karşıya kaldığı en önemli sorunlardan birisidir. Bu ısınmanın nedenleri farklılık göstermekle birlikte en çok görülen sebepleri bilgisayarın yüksek performansta çalıştırılması, uygun olmayan yüzeylerde kullanılması ve soğutma sisteminin yetersiz oluşudur. Bilgisayarlarda ısı üreten en önemli parçalar işlemci ve ekran kartıdır.

Tüm bilgisayar sistemi elektrikle çalışır ve bazı bilgisayar parçaları diğerlerinden daha fazla elektrik (watt- güç) tüketir. Direnç, iki elin birbirine hızlıca sürtülmesi gibi ısı üretir. Elektrik, devrelerinden ve kablolardan geçerken doğal olarak dirençle karşılaşır. LED ışığı gibi bazı parçaların çalışması için çok az elektrik gereklidir ve bu nedenle bunların ısı üretimi göz ardı edilebilecek düzeydedir. Diğer Parçalar, yani, Merkezi İşlem Birimi (CPU-İşlemci) ve Grafik İşlemci Birimi (GPU – Ekran Kartı) daha fazla elektrik kullanır. Ayrıca bu parçaların kullandığı elektrik miktarı çalışan uygulamaların türüne ve sayısına ya da yapılan hesaplamalara bağlı olarak oldukça değişiklik gösterebilir. Yüksek görsellikli oyunlar bilgisayarların aşırı ısınmasına sebep olabilir ve bazı büyük programlar da bilgisayarların ısınmasına sebep olabilir. Doğal olarak bu işlemler önemli miktarlarda ısı üretir [31].

4.1. Sunucu Bilgisayarlar

Sunucu bilgisayarlar (server) bir ağ ortamında diğer bilgisayarların erişimine açık olan, onlara istedikleri bilgi ve dosyaları alma imkânı sunan bilgisayar birimleridir. Bir sunucu bilgisayara bağlanan bilgisayara da istemci (client) denir. Sunucu bilgisayarlar değişik amaçlarla kullanılabilir. Örneğin posta sunucusu, web sitelerini barındıran web sunucusu, online oyun imkanı sunan oyun sunucuları gibi sunucu çeşitleri vardır.

Tez çalışmasında Sakarya Üniversitesi Sürekli Eğitim Merkezi (SAUSEM) sunucu odası ve sunucu bilgisayarları incelenmiştir. Örnek olarak alınan sunucu bilgisayarın özellikleri Tablo 4.1'deki gibidir:

Tablo 4.1. İncelenen sunucunun özellikleri

ÖZELLİK ADI	DEĞER
İşlemci sayısı	2
Kullanılabilir işlemci çekirdekleri	4
Bellek yuvaları	48 DIMM yuvası
Bellek türü	PC3-10600R veya PC3-10600E RDIMM'ler veya UDIMM'ler DDR3
Genişletme yuvaları	6
Ağ denetleyicisi	(4) 10GbE NC532i Flex-10 Çok İşlevli 8 Bağlantı Noktası
Sürücü açıklaması	(4) SFF SAS
Depolama denetleyicisi	(2) Smart Array P410i RAID
Form factor (tamamen yapılandırılmış)	4 (c7000); 2 (c3000)
Altyapı yönetimi	iLO Gelişmiş (iLO 3) özellikli Insight Control
Bellek, maksimum	768 GB

4.2. Sunucu Bilgisayar Sistemlerinin Soğutulması İşlemi

Yapılan araştırmalar 2010 yılı itibarıyla sunucuların ABD'deki enerji tüketiminin %2,5'inden sorumlu olduklarını ortaya çıkarmıştır. %2.5'lik başka bir pay da sunucuları soğutmak için kullanılan soğutma sistemlerine ayrılmıştır. 2010 yılında yapılan araştırmalara göre, eğer enerji tüketimleri böyle devam ederse 2020 yılına kadar sunucuların dünya genelinde kullandığı enerji oranı, hava ulaşımına ayrılan enerji oranından daha fazla olacaktır [32]. Bu da doğru soğutma sistemi kullanmanın ne kadar önemli olduğunu göstermektedir.

Bilgisayar da aynı arabaların motoru gibi hızlandıkça daha fazla ısınır. Sunucu bilgisayarlar ise daha fazla işlev ve güç kullanımı istediklerinden dolayı klasik soğutma yöntemleri sunucular ne kadar ısınırsa, soğutma maliyetini de o kadar yükseltiyor. Bu da aynı zamanda kesintilere ve hata oluşumlarına yol açmaktadır. 10 KW güç tüketen bir kabinin ürettiği sıcaklığı düşürmek içinde 10 KW soğutma kapasitesine ihtiyaç duyulmaktadır.

Günümüzde en yaygın olarak kullanılan soğutma yöntemi fanlarla yapılan hava soğutma yöntemidir. Isının hava veya suyun dolaştırılması sonucunda ısınan bileşenlerden uzaklaştırılması şeklinde yapılan soğutmaya aktif soğutma denir. Havayla soğutmada kullanılan araç fanlardır. Fanlar işlemci, ekran kartı gibi ısınan elemanların üzerine yerleştirilerek ısının dışarı atılmasına yardımcı olur. Bir diğer soğutma yöntemi ise su ile soğutmadır. Suyla soğutma daha iyi soğutma performansı ve sessiz çalışma özelliğinin yanında detaylı bir kurulum ve yüksek maliyet ister.

4.3. Doğru Soğutmanın Önemi

Tez çalışmasında üniversite sunucu odası bilgisayarları incelenmiştir. Sakarya Üniversitesi CAWIS (Kampüs Otomasyon Sistemi), Uzaktan Eğitim Sistemi gibi çok sayıda büyük projeyi barındıran bir üniversitedir. Bu nedenle sunucu bilgisayarların kesintisiz çalışması büyük önem arz eder. Sistem sağlıklı çalışmazsa yada aşırı ısınma olursa beraberinde maddi ve manevi büyük zararlar görülebilir. Örneğin; yüksek sıcaklıkta sürekli çalışan cihazların ömürleri azalır ve bu nedenle yeni sunucu cihazları almak gerekir. Bu da üniversitenin büyük zarara girmesi demektir. Ayrıca ısınma sonucunda sistemde yavaşlamalar olur ve bu da üniversiteden bilgi almak isteyen yada kurum içerisinde bilgi akışı yapmak isteyen herkesi olumsuz etkiler. Tüm internet alt yapısı bu durumdan olumsuz etkilenir.

Sunucu cihazları işlemci belli bir sıcaklığa geldiğinde otomatik kapanacak şekilde programlanmıştır. Sistemin merkezinde bulunan bir sunucu bilgisayarın aşırı ısınmadan dolayı kapanması yedek sunucu kullanımını zorunlu hale getirir. Bu da hem maliyet olarak hem de yer sıkıntısından dolayı olumsuz bir durumdur. Tüm bu nedenler göstermektedir ki doğru soğutmanın önemi büyüktür ve mutlaka kullanılmalıdır.

4.4. Bulanık Mantıkla Çalışan Isıtıcı - Soğutucu Tasarımı

Klasik soğutucular sadece sistemin içerisindeki değişkenlerle ilgilenir. Yani işlemcinin ısınmasıyla, HDD'nin daha hızlı dönmesiyle yada oyun oynanırken grafik işlemcisinin daha fazla çalışmasıyla. Bilgisayarın dışındaki ortamla ilgileri yoktur. Bilgisayar içerisindeki şartlar ev kullanıcısı için yeterli olabilir fakat çok önemli olan bir üniversite sunucu odasında bulunan süper bilgisayarlar için daha fazla kriter göz önünde bulundurulmalıdır. Tasarlayacağımız bulanık soğutma sistemi için Tablo 4.2'deki giriş ve çıkış değişkenlerini alır.

Tablo 4.2. Giriş ve çıkış değerleri

Bilgisayarın dışından sensörlerle alınan giriş değişkenleri	Ortam Sıcaklığı, Hava akışının değeri
Bilgisayarın içinden alınan giriş değişkenleri	İşlemci Kullanımı, Sunucu Yoğunluk Bilgisi
Çıkış değişkenleri	Bulanık Fan Sıcaklığı

Bu 4 değişkenin kullanılma nedenleri ise şöyle açıklanabilir:

Ortam Sıcaklığı: Ortam sıcaklığı bilgisayarların, özellikle de sunucu bilgisayarların hızını etkiler. Yaz aylarında ısınma nedeniyle bilgisayar hızlarında yavaşlamalar meydana gelir. Bu nedenle oda sıcaklığının değeri bir giriş olarak alınmalıdır. Sunucu odaları için 16-24 °C arası bir sıcaklık tavsiye edilmektedir. Daha yüksek yada düşük sıcaklıklar sunucu bilgisayarların performansını olumsuz etkilemektedir. [33].

Hava akışı (Airflow) : Hava akışı da sunucu odaları için önemli bir kriterdir. Hava akışı sensörler yardımıyla alınır ve 0 dan 100'e ölçeklendirilir. Eğer hava akışı < 20 ise sakın hava, 100 e yaklaştıkça ise yüksek hava akışı vardır denir. Hava akışının olması istenen bir durumdur.

İşlemci Kullanımı: İşlemci ne kadar hızlanırsa kullandığı elektrik miktarı da o kadar artar ve bu da sunucuların ısınmasına neden olur. Bu yüzden işlemci hızı önemli bir kriterdir.

Sunucu Yoğunluk Bilgisi: Sakarya Üniversitesi sunucularında yapılan bir çalışmada hangi saatler arasında daha siteye girişin daha fazla olduğu dolayısıyla da sunucuların daha hızlı çalıştığı incelenmiştir. Bu incelemeye dayanarak sunucu yoğunluk bilgisi tanımlanıp giriş değişkeni olarak kullanılacaktır.

Çıkış değişkenleri olarak da Bulanık Fan Sıcaklığı alınacaktır. Bulanık fan bilgisayarların en çok ısınan parçaları olan CPU ve GPU' nun ve tüm sistemin olması gerekli olan sıcaklık değerini belirleyecektir. Bu sonuçlara göre bulanık mantık denetleyicili bir sistem tasarımı yapılacaktır. Sistem simülasyonu FuzzyTECH, LabVIEW FLT ve Matlab FIS Editor'de gerçekleştirilip sonuçları karşılaştırılacaktır.

4.5. Tasarlanan Sistemin FuzzyTECH Programı ile Simülasyonu

Önceki bölümlerde de işlendiği üzere FuzzyTECH sadece bulanık mantık uygulamaları gerçekleştirmek üzere yazılmış bir programdır. İlk olarak giriş ve çıkış değişkenlerinin bulanıklaştırılması gerekmektedir. Bunu yapmak için FuzzyTECH programı başlatıldıktan sonra hangi giriş ve çıkış değişkenlerinin bulanıklaştırılacağı belirlenmelidir.

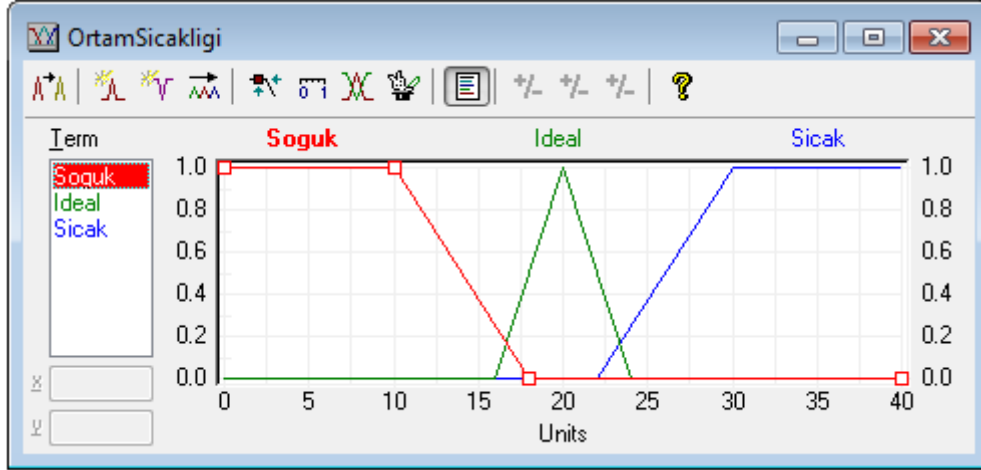
Bulanıklaştırma işlemi giriş değişkenleri ve çıkış değişkenlerini dilsel ifadelerle gösterme biçimidir.

4.5.1. Ortam sıcaklığı giriş değişkeni

Oda sıcaklığı giriş değişkeni için şu ifadeler kullanılmıştır:

Soğuk (0,17°C), İdeal (16,24°C), Sıcak (22,40°C)

Oda sıcaklığı FuzzyTECH değişken editöründe üyelik fonksiyonu olarak Şekil 4.1'de görüldüğü gibi tanımlanmıştır. Şekil üzerinde de görülebileceği gibi üyelik fonksiyonları daha iyi anlaşılabilmesi için farklı renklerde gösterilmektedir. Soğuk üyelik fonksiyonu kırmızı renkle, ideal üyelik fonksiyonu yeşil renkte ve sıcak üyelik fonksiyonu da mavi renkle gösterilmektedir.

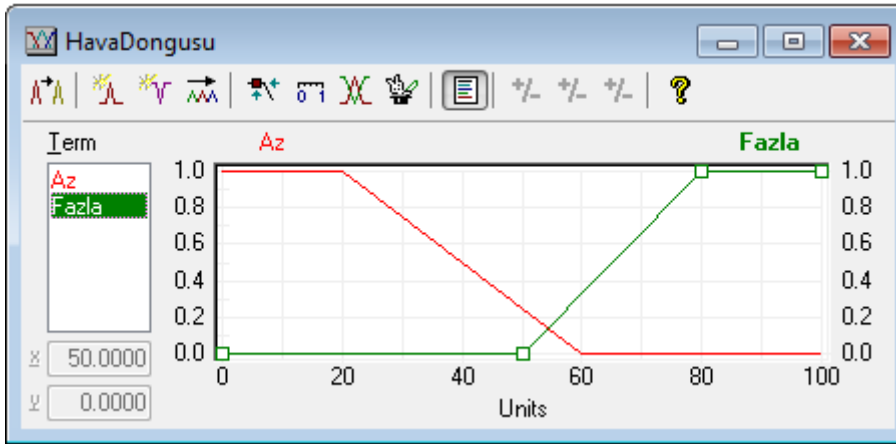


Şekil 4.1. FuzzyTECH programında ortam sıcaklığı giriş değişkeninin tanımlanması

4.5.2. Hava akışı giriş değişkeni

Hava akışı giriş değişkeni için şu ifadeler kullanılmıştır:

Az (%0-60), Fazla (%40-100) Hava akışı değişkeni FuzzyTECH değişken editöründe Şekil 4.2’de görüldüğü gibi tanımlanmıştır:

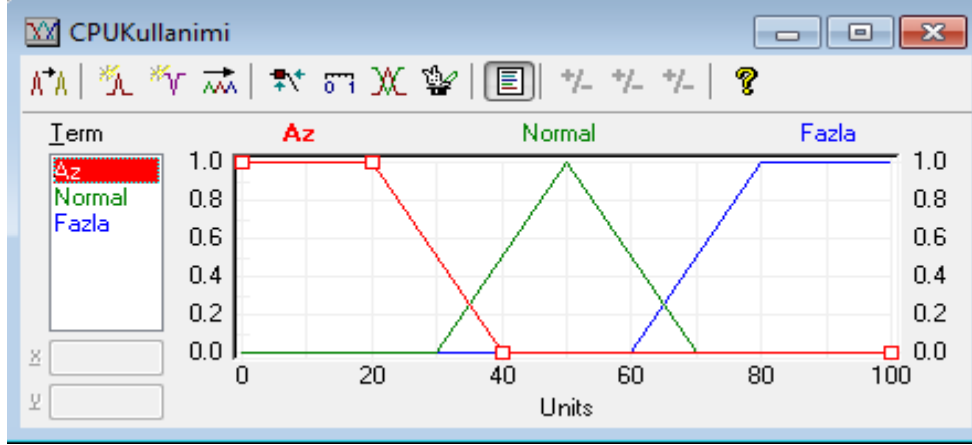


Şekil 4.2 FuzzyTECH programında hava döngüsü giriş değişkeninin tanımlanması

4.5.3. İşlemci kullanımı giriş değişkeni

İşlemci kullanımı giriş değişkeni için şu ifadeler kullanılmıştır:

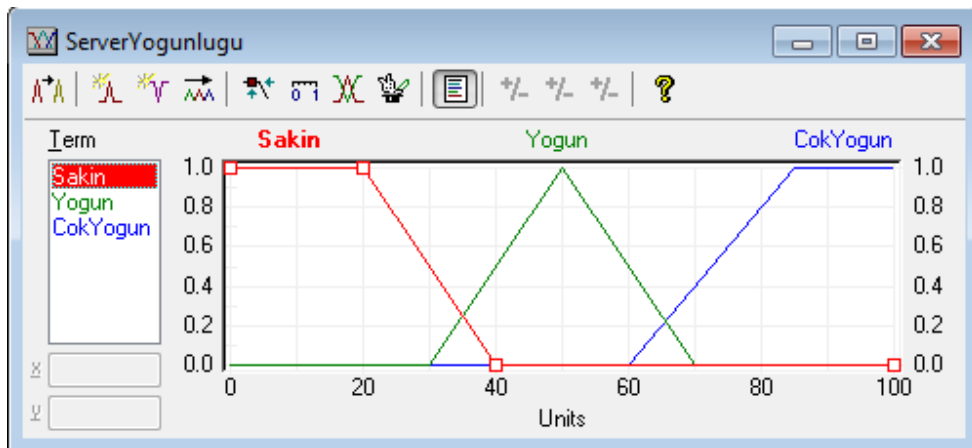
Az (%0-40), Normal (%30-70), Fazla (%60-100) İşlemci kullanımı değişkeni FuzzyTECH değişken editöründe Şekil 4.3’de görüldüğü gibi tanımlanmıştır:



Şekil 4.3. FuzzyTECH programında CPU kullanımı giriş değişkeninin tanımlanması

4.5.4. Sunucu yoğunluk değişkeni

Sunucu yoğunluğu ısınma için önemli bir nedendir. Sunucu yoğunluk üyelik fonksiyonu terimleri şunlardır: Sakin (%0-40), Yoğun (%30-70), Çok Yoğun (%60-100). Şekil 4.4’de sunucu yoğunluğu giriş değişkeninin tanımlandıktan sonraki görüntüsü görülmektedir.

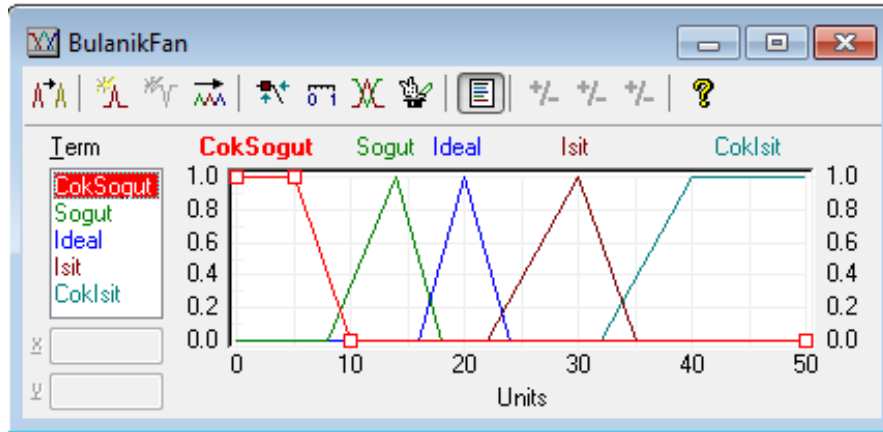


Şekil 4.4. FuzzyTECH programında sunucu yoğunluğu giriş değişkeninin tanımlanması

4.5.5. Bulanık fan çıkış sıcaklık değişkeni

Tasarlanan sistemdeki bulanık fan sistemi ideal sıcaklık koşulları arasında (16-24 °C) tutmak için sadece soğutma değil gerekirse ısıtma da yapabilecek şekildedir. Bu nedenle çıkış değişkeni içerisinde ısıt terimi de bulunmalıdır. Sıcaklık değişkeninin terimleri şunlardır:

Çok Soğut (0-10 °C), Soğut (7-18 °C), İdeal (16-24 °C), Isıt (22-30 °C), Çok Isıt (28-40 °C). Şekil 4.5’de çıkış değişkeninin tanımlanması gösterilmektedir.



Şekil 4.5. FuzzyTECH programında bulanık fan çıkış değişkeninin tanımlanması

4.5.6. Kural tabanı oluşturma

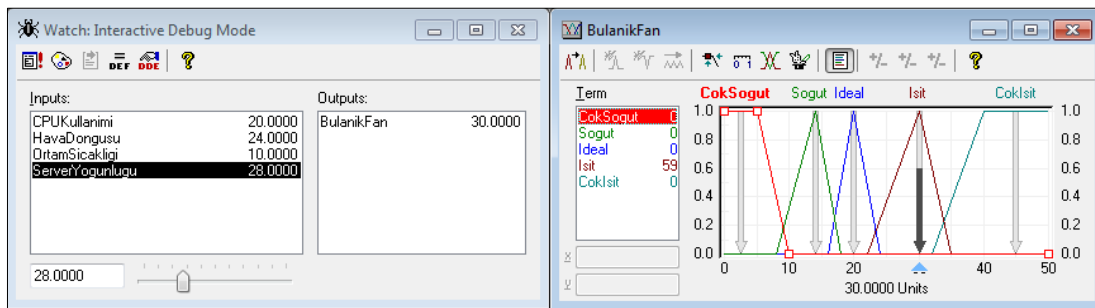
Sistemin düzgün çalışabilmesi için gerekli olan kurallar kural tabanında belirlenir. Sistemi tam olarak tasarlamak için üyelik fonksiyonu terim sayılarını çarpmak gereklidir. Hazırlanan bu sistemi ifade etmek için 54 (2*3*3*3) tane kural yazılabilir. Tabi bu şekilde en hassas kontrol sağlanabilir. Yazılan kurallar FuzzyTECH Rule Editor’de girilmiştir. 54 tane kural Şekil 4.6’da görüldüğü gibi girildikten sonra kural editöründe görülmektedir.

Name	If	Operators	Then	With	Cc
B1.G1.R3	OrtamSicakligi.Soguk,ServerYogunlugu.Yogun,HavaDongusu.Fazla,CPUKullanimi.Az	=>	BulanikFan.Ideal	100	
B1.G1.R4	OrtamSicakligi.Soguk,ServerYogunlugu.Sakin,HavaDongusu.Az,CPUKullanimi.Az	=>	BulanikFan.Jsit	100	
B1.G1.R5	OrtamSicakligi.Soguk,ServerYogunlugu.CokYogun,HavaDongusu.Fazla,CPUKullanimi.Az	=>	BulanikFan.Ideal	100	
B1.G1.R6	OrtamSicakligi.Soguk,ServerYogunlugu.CokYogun,HavaDongusu.Az,CPUKullanimi.Az	=>	BulanikFan.Ideal	100	
B1.G1.R7	OrtamSicakligi.Ideal,ServerYogunlugu.Yogun,HavaDongusu.Az,CPUKullanimi.Az	=>	BulanikFan.Ideal	100	
B1.G1.R8	OrtamSicakligi.Ideal,ServerYogunlugu.Sakin,HavaDongusu.Fazla,CPUKullanimi.Az	=>	BulanikFan.Ideal	100	
B1.G1.R9	OrtamSicakligi.Ideal,ServerYogunlugu.Sakin,HavaDongusu.Az,CPUKullanimi.Az	=>	BulanikFan.Ideal	100	
B1.G1.R10	OrtamSicakligi.Ideal,ServerYogunlugu.Yogun,HavaDongusu.Fazla,CPUKullanimi.Az	=>	BulanikFan.Sogut	100	
B1.G1.R11	OrtamSicakligi.Ideal,ServerYogunlugu.CokYogun,HavaDongusu.Az,CPUKullanimi.Az	=>	BulanikFan.Sogut	100	
B1.G1.R12	OrtamSicakligi.Ideal,ServerYogunlugu.CokYogun,HavaDongusu.Fazla,CPUKullanimi.Az	=>	BulanikFan.Jsit	100	
B1.G1.R13	OrtamSicakligi.Sicak,ServerYogunlugu.Sakin,HavaDongusu.Az,CPUKullanimi.Az	=>	BulanikFan.Ideal	100	
B1.G1.R14	OrtamSicakligi.Sicak,ServerYogunlugu.Sakin,HavaDongusu.Fazla,CPUKullanimi.Az	=>	BulanikFan.Ideal	100	
B1.G1.R15	OrtamSicakligi.Sicak,ServerYogunlugu.Yogun,HavaDongusu.Az,CPUKullanimi.Az	=>	BulanikFan.Sogut	100	
B1.G1.R16	OrtamSicakligi.Sicak,ServerYogunlugu.Yogun,HavaDongusu.Fazla,CPUKullanimi.Az	=>	BulanikFan.Sogut	100	
B1.G1.R17	OrtamSicakligi.Sicak,ServerYogunlugu.CokYogun,HavaDongusu.Az,CPUKullanimi.Az	=>	BulanikFan.CokSogut	100	
B1.G1.R18	OrtamSicakligi.Sicak,ServerYogunlugu.CokYogun,HavaDongusu.Fazla,CPUKullanimi.Az	=>	BulanikFan.CokSogut	100	
B1.G1.R19	OrtamSicakligi.Soguk,ServerYogunlugu.Sakin,HavaDongusu.Az,CPUKullanimi.Normal	=>	BulanikFan.Jsit	100	
B1.G1.R20	OrtamSicakligi.Soguk,ServerYogunlugu.Sakin,HavaDongusu.Fazla,CPUKullanimi.Normal	=>	BulanikFan.Jsit	100	
B1.G1.R21	OrtamSicakligi.Soguk,ServerYogunlugu.Yogun,HavaDongusu.Az,CPUKullanimi.Normal	=>	BulanikFan.Jsit	100	
B1.G1.R22	OrtamSicakligi.Soguk,ServerYogunlugu.Yogun,HavaDongusu.Fazla,CPUKullanimi.Normal	=>	BulanikFan.Ideal	100	
B1.G1.R23	OrtamSicakligi.Soguk,ServerYogunlugu.CokYogun,HavaDongusu.Az,CPUKullanimi.Normal	=>	BulanikFan.Ideal	100	
B1.G1.R24	OrtamSicakligi.Soguk,ServerYogunlugu.CokYogun,HavaDongusu.Fazla,CPUKullanimi.Normal	=>	BulanikFan.Ideal	100	
B1.G1.R25	OrtamSicakligi.Ideal,ServerYogunlugu.Sakin,HavaDongusu.Az,CPUKullanimi.Normal	=>	BulanikFan.Ideal	100	

Şekil 4.6. FuzzyTECH programında kural tabanı oluşturulması

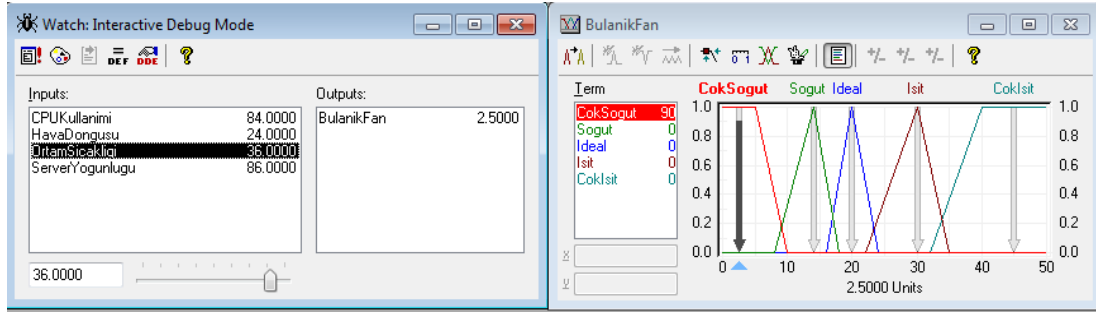
4.5.7. Simülasyon sonuçlarını görüntüleme

Birinci simülasyon kriterleri olarak CPU Kullanımı %20, Hava Döngüsü %20, Ortam Sıcaklığı 10°C, Sunucu Yoğunluğu %28 alınır Şekil 4.7’de görüldüğü gibi Bulanık Fan 30 derece yani “Isıt” olarak bulunmuştur.



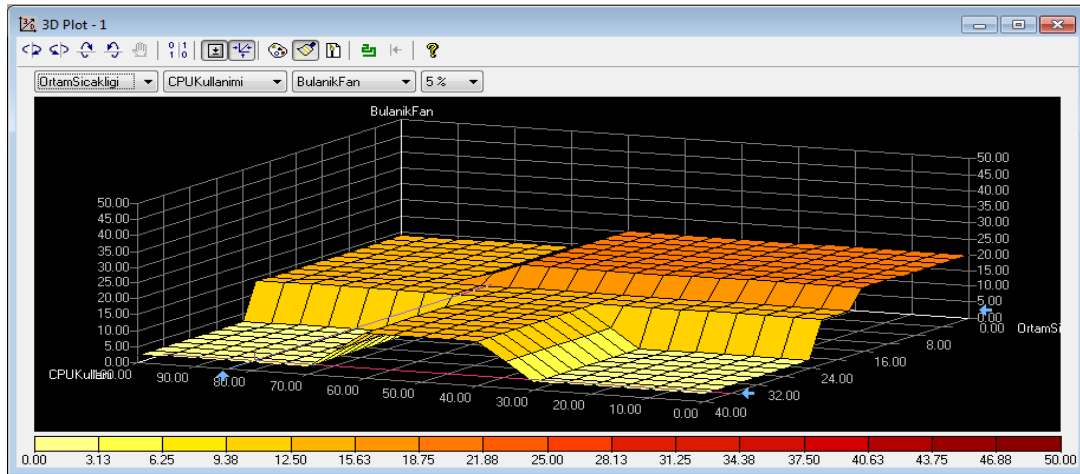
Şekil 4.7. FuzzyTECH programında sistemin ısıt için ayarlanması

İkinci simülasyon kriterleri olarak CPU Kullanımı %84, Hava Döngüsü %24, Ortam Sıcaklığı 36°C, Sunucu Yoğunluğu %86 alınır Şekil 4.8’de görüldüğü gibi Bulanık Fan 2.5 biriminde yani “Çok Soğut” olarak bulunmuştur.



Şekil 4.8 FuzzyTECH programında sistemin çok soğut için ayarlanması

Sistemin 3 boyutlu görüntüsü ise Şekil 4.9’da görülmektedir. Burada giriş değişkenleriyle çıkış değişkenlerinin arasındaki ilişki daha net görülmektedir.



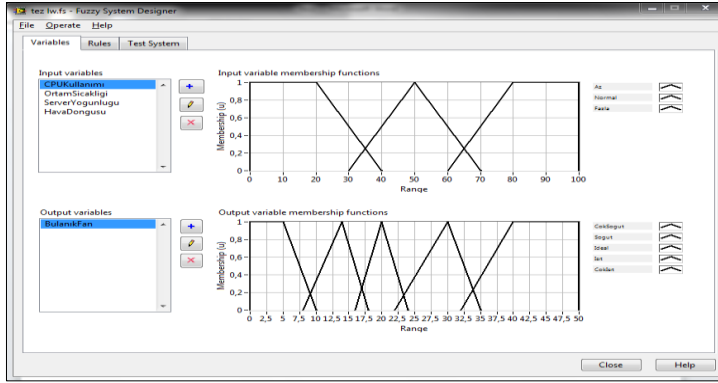
Şekil 4.9 FuzzyTECH programında tasarlanan sistemin 3 boyutlu görünümü

4.6. Tasarlanan Sistemin Labview Programı ile Simülasyonu

Bulanık fan tasarımı yapılırken kullanılan programlardan birisi de Labview programı olmuştur. Giriş ve çıkış değişkenlerinin gösterimi daha önceki bölümlerde tek tek verildiği için burada hepsinin değeri tekrar edilmeyecektir.

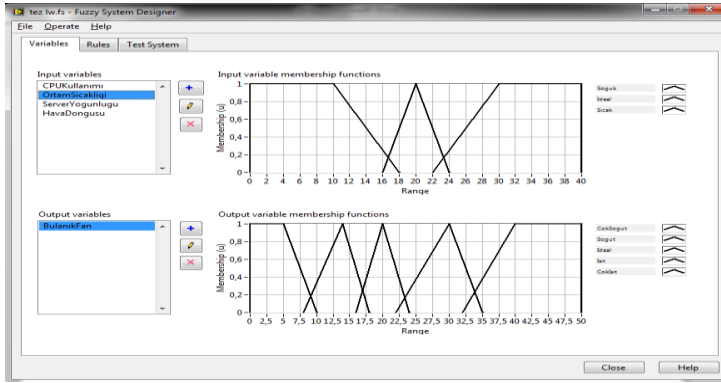
4.6.1. Değişkenlerin Labview’ da tasarlanması

CPU Kullanımı giriş değişkenininin Labview programındaki ifadesi Şekil 4.10’da görüldüğü gibidir.



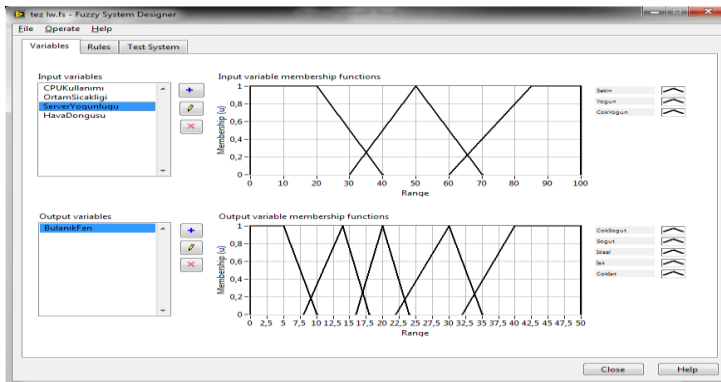
Şekil 4.10. LabVIEW programında CPU kullanımı giriş ve çıkış değişkenlerinin tasarlanması

Ortam Sıcaklığı giriş değişkeninin ve Bulanık Fan çıkış değerlerinin Labview programında üyelik fonksiyonları Şekil 4.11’de görülmektedir:



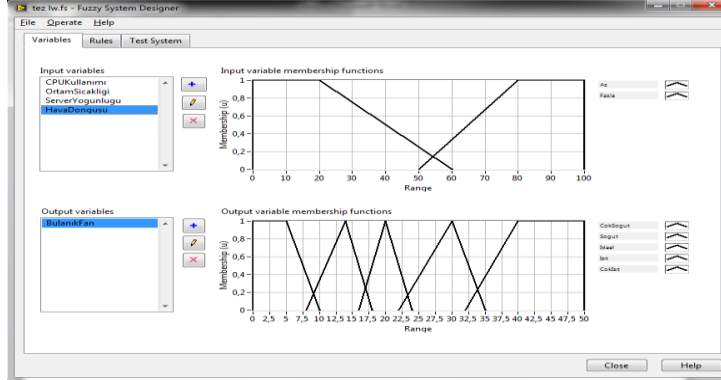
Şekil 4.11. LabVIEW programında ortam sıcaklığı giriş değişkeninin tasarlanması

Sunucu Yoğunluğu değişkeninin giriş üyelik fonksiyonları Şekil 4.12’deki gibidir:



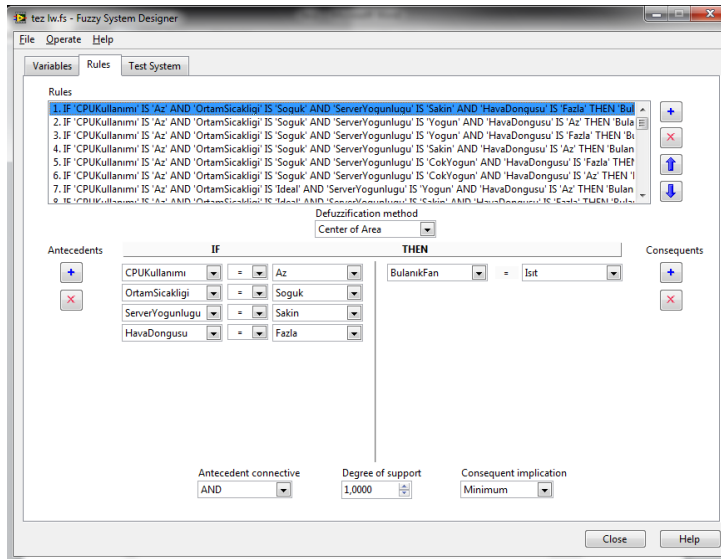
Şekil 4.12. LabVIEW programında sunucu yoğunluğu giriş değişkeninin tasarlanması

Hava döngüsü giriş değişkeni üyelik fonksiyonlarının Labview programındaki ifadesi Şekil 4.13’de görülmektedir.



Şekil 4.13. LabVIEW ‘da hava döngüsü giriş değişkeninin tasarlanması

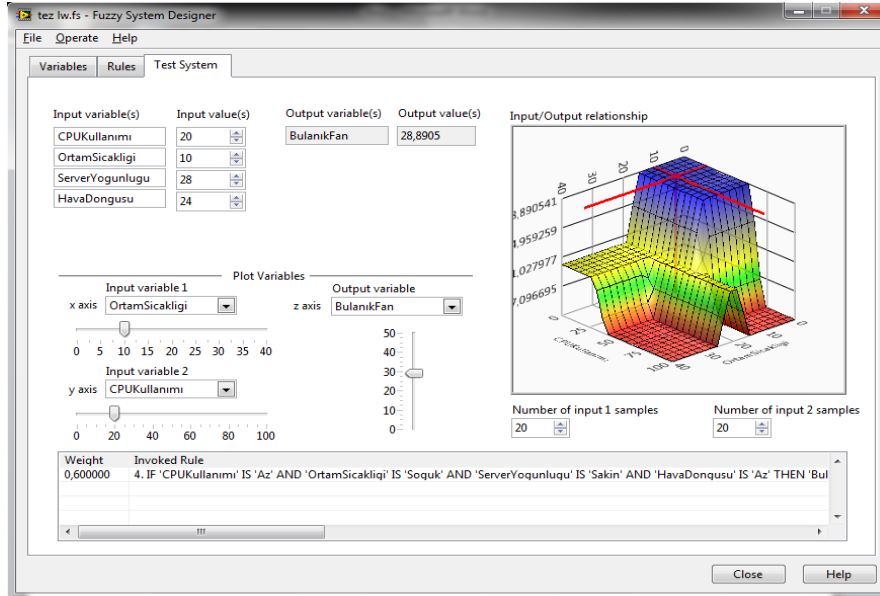
Labview programında Kural Tablosu hazırlanmıştır ve bu kural tablosu Şekil 4.14’de görüldüğü gibi olmuştur.



Şekil 4.14. LabVIEW programında oluşturulan kural tabanı

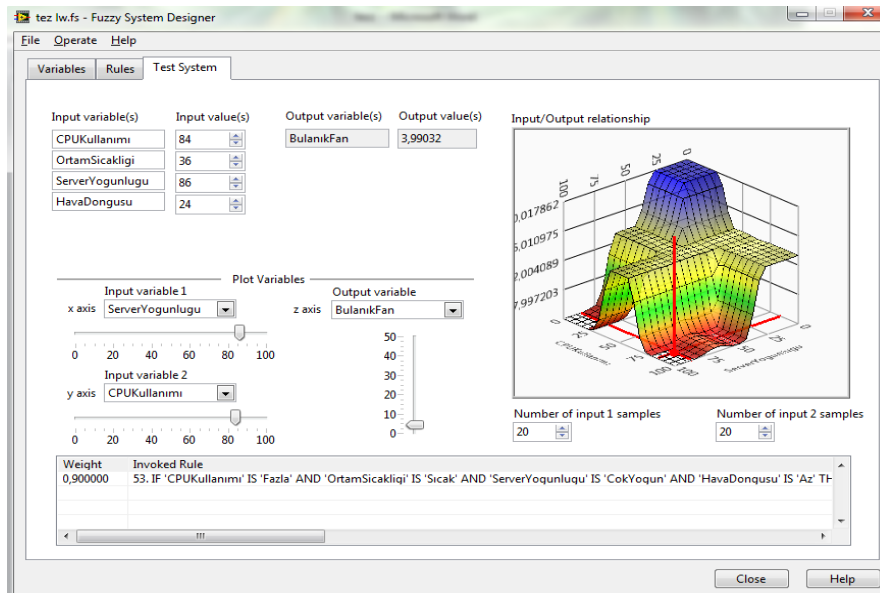
4.6.2. Simülasyon sonuçlarını görüntüleme

Birinci simülasyon kriterleri olarak CPU Kullanımı %20, Hava Döngüsü %20, Ortam Sıcaklığı 10°C, Sunucu Yoğunluğu %28 alınır Şekil 4.15’de görüldüğü gibi Bulanık Fan 28,9 biriminde yani “Isıt” olarak bulunmuştur.



Şekil 4.15. LabVIEW programında birinci simülasyon sonuçları

İkinci simülasyon kriterleri olarak CPU Kullanımı %84, Hava Döngüsü %24, Ortam Sıcaklığı 36°C, Sunucu Yoğunluğu %86 alınır Şekil 4.16’da görüldüğü gibi Bulanık Fan 3.9 biriminde yani “Çok Soğuk” olarak bulunmuştur.



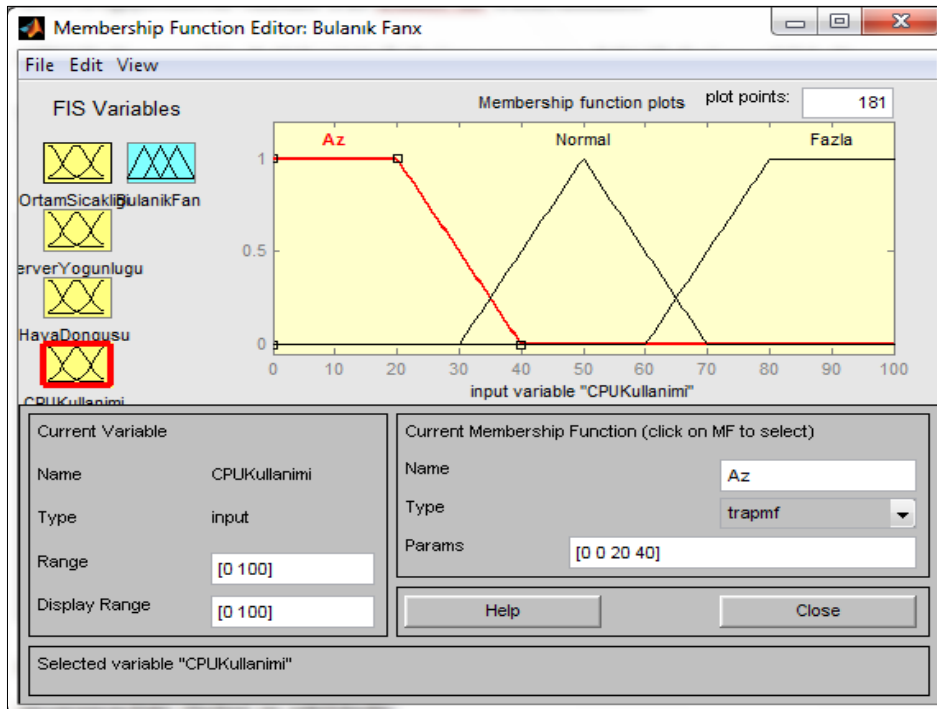
Şekil 4.16. LabVIEW programında ikinci simülasyon sonuçları

4.7. Tasarlanan Sistemin Matlab Programı ile Simülasyonu

Bulanık fan tasarımı yapılırken kullanılan programlardan birisi de Matlab programı olmuştur. Giriş ve çıkış değişkenlerinin gösterimi daha önceki bölümlerde verildiği için burada hepsinin değeri tekrar edilmeyecektir. Sonraki bölümde üç programda da yapılan simülasyon işlemi ve sonuçları değişik kriterlere göre karşılaştırılacaktır.

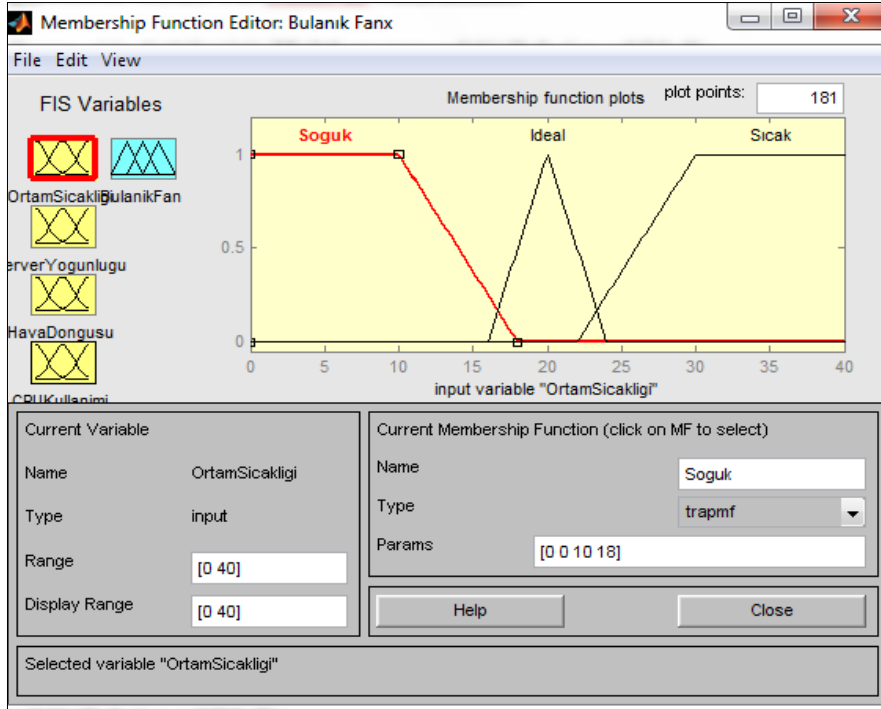
4.7.1. Değişkenlerin Matlab FIS Editor'de tasarlanması

CPU Kullanımı giriş değişkeni Matlab programında tanımlanmıştır. Üyelik fonksiyonları Şekil 4.17'de görülmektedir. Burada giriş CPU Kullanımı giriş değişkeninin değerleri görülmektedir.



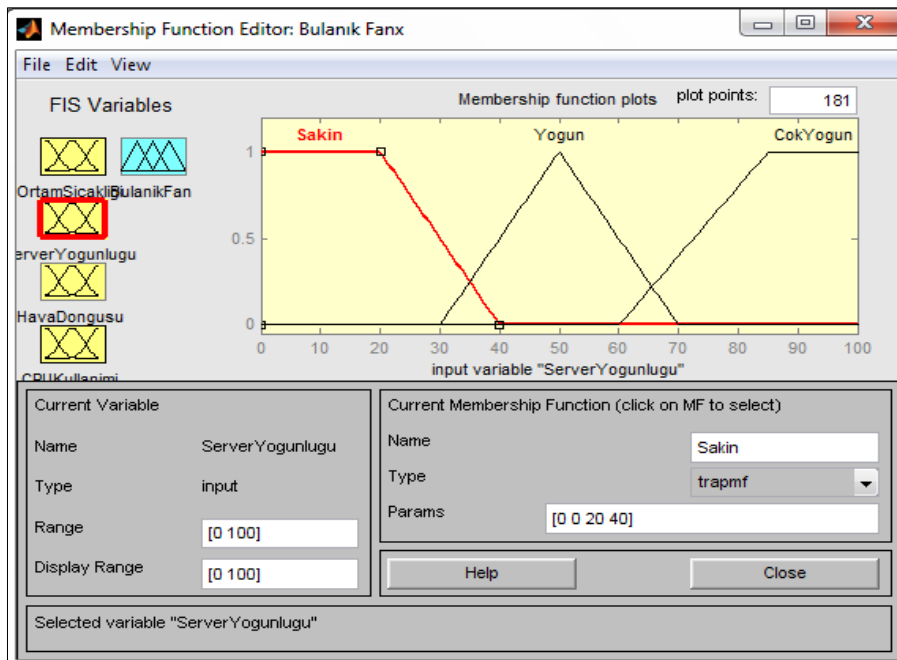
Şekil 4.17. MATLAB programında CPU kullanımı giriş değişkeninin tanımlanması

Ortam Sıcaklığı giriş değişkeninin ve Bulanık Fan çıkış değerlerinin Matlab programındaki ifadesi Şekil 4.18'de görüldüğü gibidir. Bu şekilde giriş değişkeni ve üyelik fonksiyonlarının parametreleri görülmektedir.



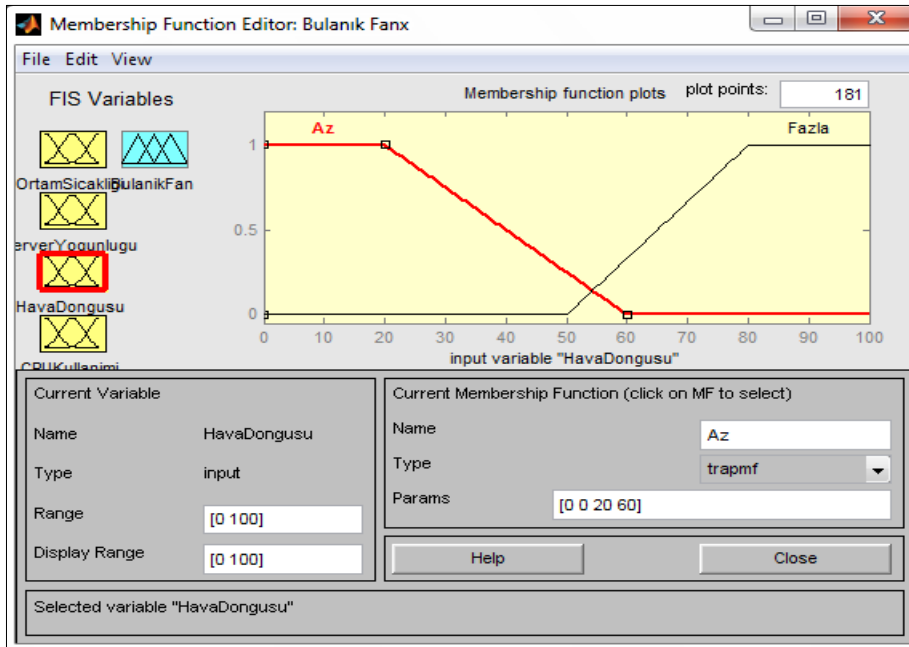
Şekil 4.18. MATLAB programında ortam sıcaklığı giriş değişkeninin tanımlanması

Sunucu Yoğunluğu giriş değişkeninin Matlab programındaki ifadesi Şekil 4.19'da görülmektedir. Şekilde görüldüğü gibi üyelik fonksiyonları Sakin, Yoğun ve Çok Yoğun şeklinde alınmıştır.



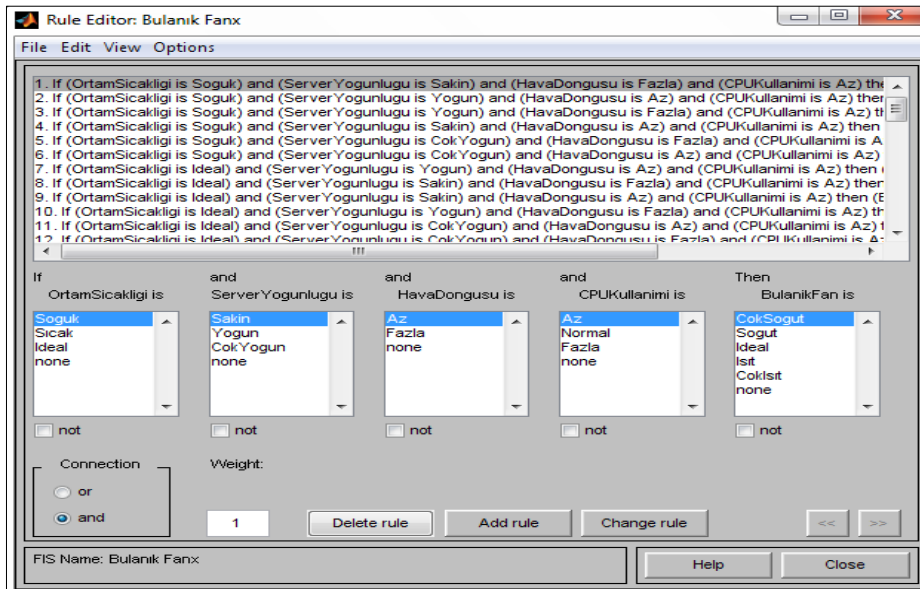
Şekil 4.19. MATLAB programında sunucu yoğunluğu giriş değişkeninin tanımlanması

Hava döngüsü giriş değişkeninin Matlab programındaki ifadesi Şekil 4.20'deki gibidir. Bu şekilde üyelik fonksiyonlarının hangi değerler arasında olduğu görülmektedir.



Şekil 4.20. MATLAB 'da hava döngüsü giriş değişkeninin tanımlanması

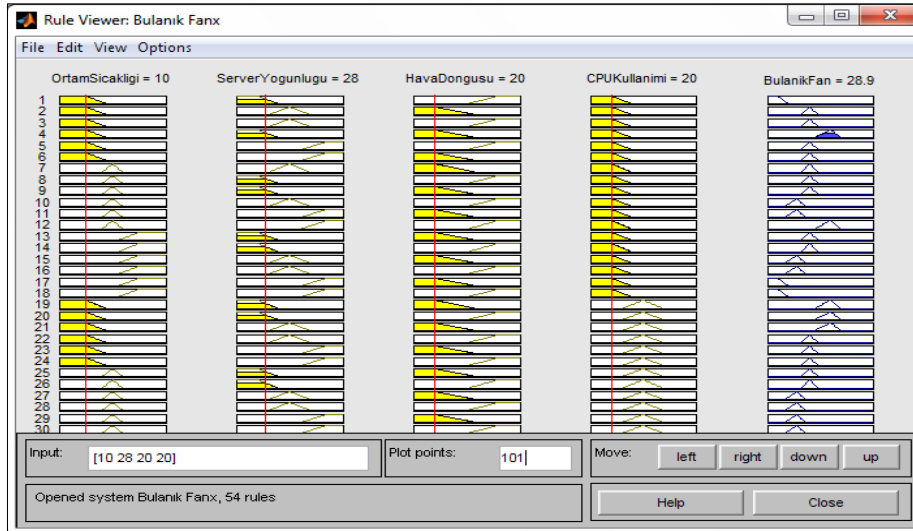
Kural Tablosu MATLAB ekranında Şekil 4.21'deki şekilde olmaktadır.



Şekil 4.21. MATLAB programında kural tablosunun tasarlanması

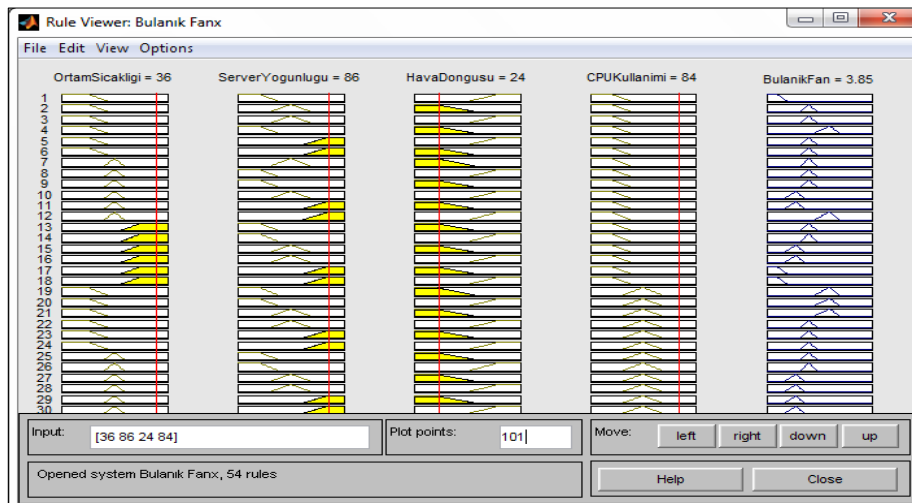
4.7.2. Simülasyon sonuçlarını görüntüleme

Birinci simülasyon kriterleri olarak CPU Kullanımı %20, Hava Döngüsü %20, Ortam Sıcaklığı 10°C, Sunucu Yoğunluğu %28 alınır Şekil 4.22’de görüldüğü gibi Bulanık Fan 28,9 biriminde yani “Isıt” olarak bulunmuştur.



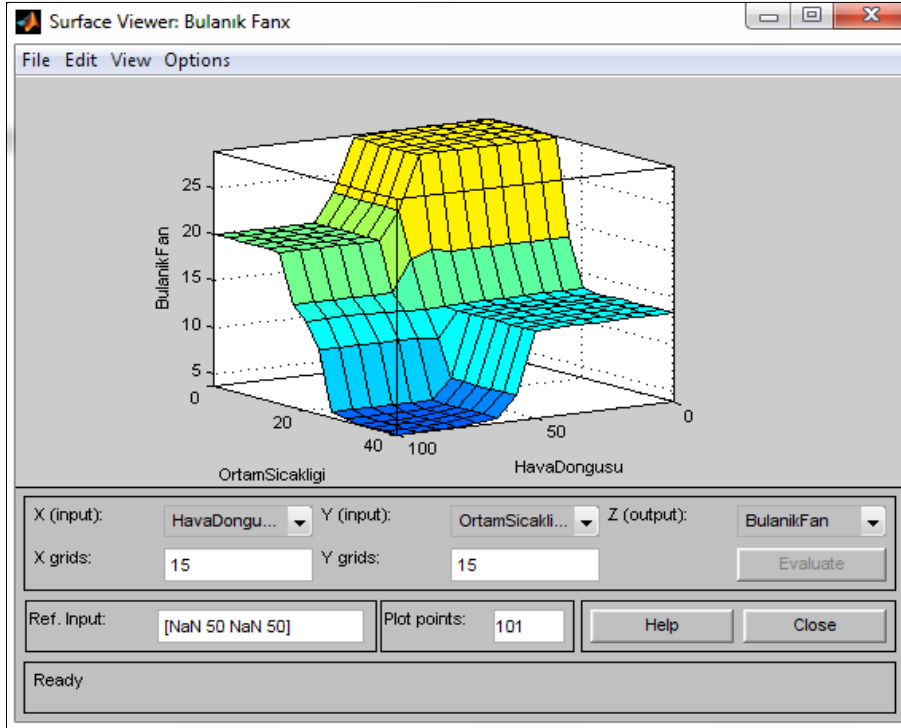
Şekil 4.22. MATLAB programında birinci simülasyon sonuçları

İkinci simülasyon kriterleri olarak CPU Kullanımı %84, Hava Döngüsü %24, Ortam Sıcaklığı 36°C, Sunucu Yoğunluğu %86 alınır Şekil 4.23’de görüldüğü gibi Bulanık Fan 3.85 biriminde yani “Çok Soğuk” olarak bulunmuştur.



Şekil 4.23. MATLAB programında ikinci simülasyon sonuçları

Sistemin Yüzey Görüntüleyici 'deki 3 boyutlu görüntüsü ise Şekil 4.24'de görülmektedir. Bu şekilde giriş değişkenlerine karşı çıkış değişkeninin aldığı tüm değerler aynı anda görülebilmektedir.



Şekil 4.24. MATLAB programında sistemin yüzey görünümü

BÖLÜM 5. BULANIK MANTIK MODEL GELİŞTİRME ORTAMLARININ VE SİMÜLASYON SONUÇLARININ KARŞILAŞTIRILMASI

Simülasyon Matlab, Labview ve FuzzyTECH programları ile yapılmıştır. Bu bölümde bulanık mantık tasarımında kullanılabilen bu programların değişik kriterlere göre karşılaştırılması yapılacaktır. Karşılaştırmada kullanılacak kriterler şunlardır:

1. Kurulum,
2. Kullanım kolaylığı,
3. Erişilebilirlik,
4. Görsellik,
5. Performans,
6. Sistem kaynaklarını kullanma,
7. Destek,
8. Eğitim ortamlarında kullanım durumu,
9. Dokümantasyon,
10. Nesneye yönelik olma durumu,
11. Model kütüphanelerinin genişliği,
12. Sonuçların analiz edilme kolaylığı,
13. Genişletilebilirlik ve yeniden kullanılabilirlik ve
14. GUI oluşturma kriterine göre programların karşılaştırılması.

5.1. Kurulum Kriterine Göre Programların Karşılaştırılması

MATLAB ve LabVIEW programları sadece bulanık mantık sistemi tasarlamak için yazılmış programlar değildir. Her türlü mühendislik uygulamaları yapmaya olanak sağlayan geliştirme platformlarıdır. Tasarım yapılırken kullanılan programlardan sadece FuzzyTECH sadece bulanık mantık tasarımı yapmak için özelleştirilmiş bir programdır. Bu yüzden kurulum yapılırken FuzzyTECH programına herhangi bir

eklenmesi yüklemeye gerek yoktur. Ancak MATLAB ve LabVIEW programlarında bulanık mantık işlemleri yapabilmek için Fuzzy Logic Toolkit denilen bulanık mantık araçlarını yüklemek gereklidir. Bu açıdan bakıldığında sadece bulanık mantık tasarımı yapacak olan bir tasarımcı için FuzzyTECH programı tek başına yeterli olmaktadır.

Maliyet noktasından bakıldığında ise FuzzyTECH 6.02 yani son sürümü sadece 29 \$'a almak mümkündür. Ancak LabVIEW programı 3,432 €'dur ve bulanık mantık aracı yüklenmek istenirse ilave 1,099 € vermek gerekmektedir. MATLAB programı öğrenci sürümünü bulanık mantık araç kutusuyla birlikte 118 \$'a almak mümkündür. Tasarım sürecinde FuzzyTECH 6.02, Matlab R2009a, LabVIEW 2011 programları kullanılmıştır. Minimum kurulum gereksinimleri olarak incelenirse:

FuzzyTECH: Windows 7,XP, Windows 2000, Windows NT 4.0 veya Windows 98-95, IE 5.01 ve üstü 30 MB disk alanı ve 16 MB RAM.

LabVIEW Programı: Pentium III/Celeron 866 MHz veya eşiti işlemci, 256 MB Ram, Ekran Çözünürlüğü: 1024 x 768 piksel, İşletim sistemi: Windows 7/Vista (32-bit veya 64-bit), Windows XP SP3 (32-bit), Windows Server 2003 R2 (32-bit), Windows Server 2008 R2 (64-bit), Disk Alanı: 353 MB.

MATLAB Programı: Microsoft Windows 7,XP, 95, Windows 98, Windows Millennium Edition, Windows NT 4.0, Windows 2000. Pentium, Pentium Pro, Pentium II, Pentium III, Pentium IV* yada AMD Athlon tabanlı bilgisayar. 64 MB RAM minimum, 128 MB RAM

Sadece MATLAB için 120 GB, online yardım dosyaları için 260 MB. 8-bit grafik kartı. Microsoft Visual C/C++ versiyon 5.0 yada 6.0 gereklidir.

5.2. Kullanım Kolaylığı Kriterine Göre Programların Karşılaştırılması

MATLAB programının avantajları olarak güçlü grafiksel altyapısı ve kullanım kolaylığı olarak düşünülmektedir. MATLAB'da hazırlanan programların derlenmekten ziyade yorumlandığı söylenebilir. MATLAB içerisinde bulunan özellikler ve sonradan eklenebilen özelliklerle kullanım kolaylığı sağlar. Bulanık

mantık tasarım olarak düşünüldüğünde MATLAB ile bulanık mantık denetleyici tasarımı yapmak FuzzyTECH ve LabVIEW programlarına göre daha zor olmaktadır. Bunun nedenleri olarak ise Kural Editöründe kural yazmak diğerlerine göre daha zordur. Aynı zamanda fazla sayıda giriş değişkeni olduğu zaman girişler ekrana sığmamakta bu da kural yazmayı zorlaştırmaktadır.

LabVIEW programı güçlü ve esnek programlama dilleri arasındadır. LabVIEW etkili, güçlü ve kolay kullanımlı grafik geliştirme ortamı sağlar. LabVIEW test, ölçüm, otomasyon ve kontrol uygulamaları oluşturmak için ideal bir araçtır. Bir bilim adamı, mühendis veya teknisyen, LabVIEW programlama dili ile uygun bir arayüz ortamı bulur, böylece hızlı bir şekilde ürün tasarımlarını gerçekleştirebilir.

LabVIEW programının tam özellikli bir programlama dili olması ve sağlam bir temel üzerine inşa edilmiş olduğu göz önüne alındığında, herhangi bir noktada kullanıcı programı tamamen özelleştirilebilir, güçlü programlama elemanları ve yapılarından yararlanabilir [28].

LabVIEW ile bulanık mantık tasarımı yapılırken yukarıda ifade edilen kullanım özelliklerinden faydalanılmıştır. LabVIEW üzerinde çalışılan programlar içerisinde bulanık mantık tasarımı olarak en görsel ve kullanım kolaylığı sağlayan program olmuştur. Bu açıdan bakıldığında LabVIEW ile bulanık mantık tasarımı yapmak ve simüle etmek diğer programlardan daha kolay ve anlaşılır demek mümkündür. Kural editörü ile tüm kuralları programın kendisi oluşturabilir (7 girişe kadar). Bu da kullanıcıya tüm kuralları yazmadan sadece çıkış değişkenini değiştirerek tasarım yapma kolaylığı sağlar.

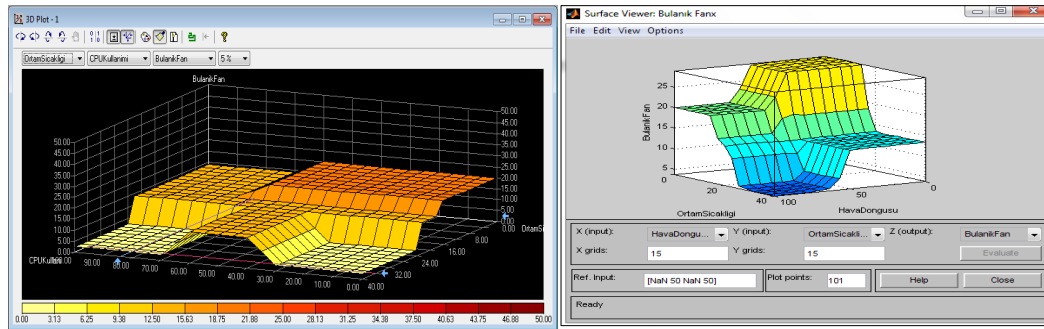
FuzzyTECH programı bulanık mantık çalışmaları için özelleştirilmiş bir program olması açısından özel bir konumda bulunmaktadır. Bulanık mantık tasarım sihirbazı kullanıcıya sadece giriş ve çıkış değişkenlerinin sayısını ve isimlerini vererek kısa zamanda bir sistem tasarımı yapmaya olanak sağlar. FuzzyTECH programında kural yazmak da son derece kolaydır. Ancak giriş değişkenleri oluştururken değişkenlerin değerlerini belirlemek ve bunu grafik üzerinde ayarlamak LabVIEW programına göre daha zordur denebilir. Programla ilgili yeterli örnek uygulama olmaması da bir zorluk olarak düşünülebilir.

5.3. Erişilebilirlik Kriterine Göre Programların Karşılaştırılması

Erişilebilirlik bilgisayar programlarından birinde oluşturulan dosyanın başka programlarda da kullanılmasıyla ilgilidir. Matlab ile oluşturulan bulanık mantık dosyalarının uzantısı .fis, LabVIEW programında yazılan dosyaların uzantısı .fs ve FuzzyTECH programında oluşturulan dosyaların uzantısı .ftl olmaktadır. Bu uzantılara sahip dosyalar sadece kendi programlarında çalışabilmektedir. Bu açıdan programların bulanık mantık dosyaları anlamında erişilebilir olduklarını söylemek mümkün değildir. MATLAB ile diğer programlarda oluşturulan dosyalar açılmak istendiğinde program hata vermemekle beraber kilitlenmektedir.

5.4. Görsellik Kriterine Göre Programların Karşılaştırılması

Görsellik incelenen programların tamamı için önemli bir argümandır. Yapılan bulanık mantık tasarımları uygulamaya geçmeden önce simüle edilmektedir. Bu açıdan programların görsellikleri yapılacak olan bulanık mantık denetleyici tasarımlarında gerçeğe en yakın sonuç üretmeyi kolaylaştırmaktadır. Görsellik olarak incelendiğinde programların hepsinde 3 boyutlu grafiklerle simülasyon sonuçlarını inceleme imkanı verdiği görülmüştür. İncelenen programların görsellikleri Şekil 5.1'de görüldüğü gibidir.



Şekil 5.1. İncelenen programların görsellikleri

Bir sıralama yapmak gerekirse gerçeğe en yakın görsellik sağlayan program LabVIEW programıdır. LabVIEW ön panel uygulaması sayesinde cihazları sanki gerçek hayatta kullanıyormuş izlenimi verecek şekilde tasarımlar yapmaya imkan sağlar. MATLAB bulanık mantık görselliği açısından yüzey görünüm aracı ile 3 boyutlu görünüm sağlar ancak bu LabVIEW programındaki kadar ileri değildir.

FuzzyTECH programı ise görsellik açısından bu iki programın gerisinde kalmaktadır.

5.5. Performans Kriterine Göre Programların Karşılaştırılması

Bir programın performansı çeşitli faktörlerden etkilenir. Bu faktörler, üzerinde çalışılan sistemin yapılandırılması, donanım özellikleri, kod mimarisi ve algoritma verimliliği gibi ele alınabilir. Bulanık mantık tasarımı açısından bakıldığında tasarlanan sistem için MATLAB ve LabVIEW programlarının performansları arasında bir fark görülmemiştir. İkisinin de sistem performansı güçlüdür. Bulanık mantık sistem tasarımında harcanan süre dikkate alınırsa en iyi performans LabVIEW programından alınmıştır. Daha sonra MATLAB ve FuzzyTECH programlarından alınmıştır. Tasarım yapılan sistem aynı olduğundan donanım özellikleri arasında bir fark yoktur. Elde edilen sonuçlara bakıldığında FuzzyTECH programıyla elde edilen sonucun diğer iki programda elde edilen sonuca göre daha yuvarlak bir değer verdiği görülmüştür. LabVIEW ve MATLAB aynı hassaslıkta sonuç vermiştir.

MATLAB için yapılan özel bir performans testinde ise 6 farklı görev için MATLAB'ın bu işlemleri gerçekleştirme süreleri belirlenmiştir. Bu süreler kullanılan bilgisayar ve diğer bazı bilgisayar türleri için ayrı ayrı belirlenmiştir. Bunun için MATLAB kod satırında yazılan kod şu şekildedir:

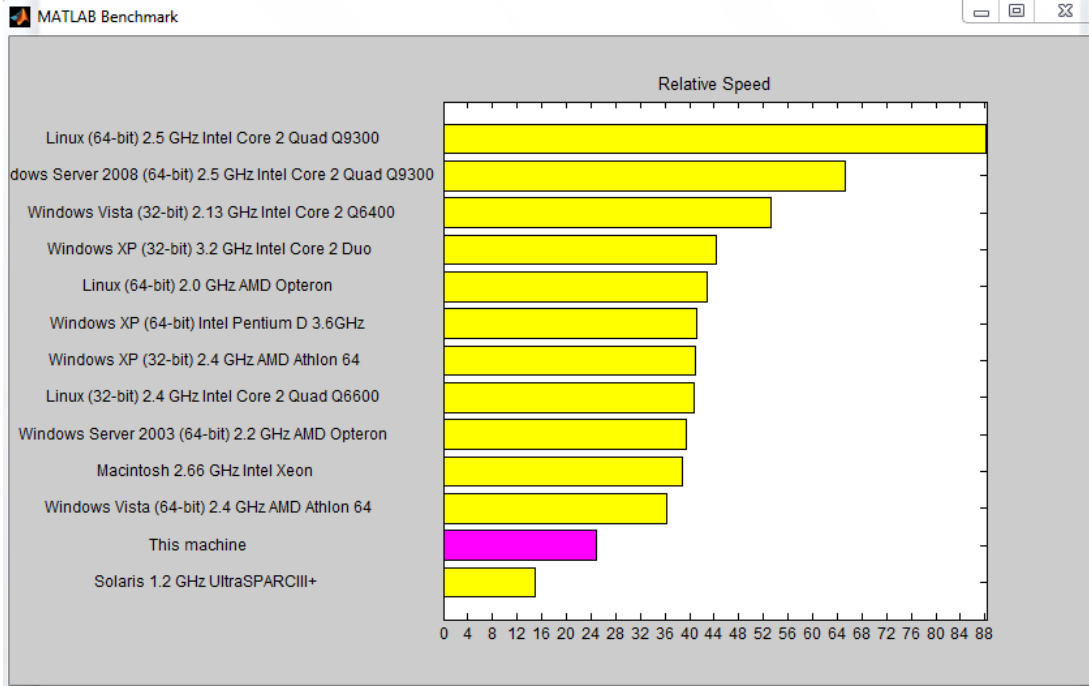
```
>> bench
```

```
>> bench(N)
```

```
>> bench(0)
```

```
>> t = bench(N)
```

Değişik bilgisayar konfigürasyonları ile yapılan deneyler sonucunda elde edilen performans değerleri Şekil 5.1'de görülmektedir. Sonuçlar incelendiğinde en iyi performansın Linux işletim sistemi altında Intel Core 2 Quad Q9300 işlemcisi ile alındığı görülmektedir.



Şekil 5.2. MATLAB performans testi çubuk grafiği

LabVIEW programının değişik işlem ve işlemciler için verdiği işlem süresi incelenmiş ve Şekil 5.2'deki sonuçlar alınmıştır.

Multicore Analysis and Sparse Matrix Toolkit VIs Using 16 threads vs. LabVIEW Advanced Analysis Libraries VIs on 16-core Xeon 64 bit CPU

Test Case (matrix size)	10	12	15	19	23	29	36	45	57	71	88	110	137	171	213	266	332	414	516	643	802	1000
A x B (CDB)	1.23	1.37	1.34	1.43	0.82	1.36	2.22	3.13	4.83	6.22	8.4	8.47	9.92	10.76	12.19	12.84	12.5	12.18	13.13	12.38	13.98	13.58
A x B (DBL)	0.84	1.2	1.17	1.21	1.33	1.51	0.78	1.25	2.13	2.99	4.38	5.58	6.94	7.81	9.13	11.7	10.42	10.6	11.09	14.02	14.02	14.12
A x Vector (CDB)	0.96	0.94	0.95	0.97	1.01	0.97	1	1.02	0.98	1.43	1.32	1.58	2.07	2.76	3.2	4	4.91	4.63	5.73	6.18	6.34	6.66
A x Vector (DBL)	1.93	1.89	1.6	1.51	1.47	1.38	1.5	1.31	1.26	1.05	1.04	1.35	1.17	2.28	2.74	3.41	4.24	5.31	6.11	6.51	6.84	7.14
Inverse Matrix (CDB)	0.96	1.03	1.07	0.7	0.63	0.89	0.77	0.71	0.73	1.14	1.28	1.27	1.9	2.69	3.31	4.27	5.13	5.65	7.02	7.74	10.15	8.43
Inverse Matrix (DBL)	0.76	0.82	0.8	0.77	0.84	0.88	0.96	0.98	1.03	0.52	0.77	0.91	1.26	1.54	2.11	2.43	3.11	3.83	5.2	6.43	6.78	7.71
Solve Linear Equations (single right hand) (CDB)	1.04	1.05	1.05	1.15	1.27	1.18	1.16	1.11	1.16	1.3	1.4	1.54	1.82	2.04	2.27	2.78	3.48	4.12	5.19	5.84	7.02	8.5
Solve Linear Equations (single right hand) (DBL)	0.85	0.87	0.83	0.66	0.75	0.85	0.84	0.9	0.9	0.81	0.91	0.98	0.96	0.95	0.96	0.99	1.12	1.23	1.38	1.53	1.69	1.97
SVD Decomposition (CDB)	0.96	0.95	0.91	0.92	0.96	0.9	0.97	1.08	1.09	1.11	1.22	1.24	1.26	1.65	1.8	2.14	2.54	3	3.43	3.93	4.68	4.93
SVD Decomposition (DBL)	0.94	0.97	0.9	0.77	0.82	0.89	0.93	0.9	1.03	1.13	1.15	1.16	1.21	1.31	1.41	1.53	1.72	1.93	2.14	2.43	2.75	3.15

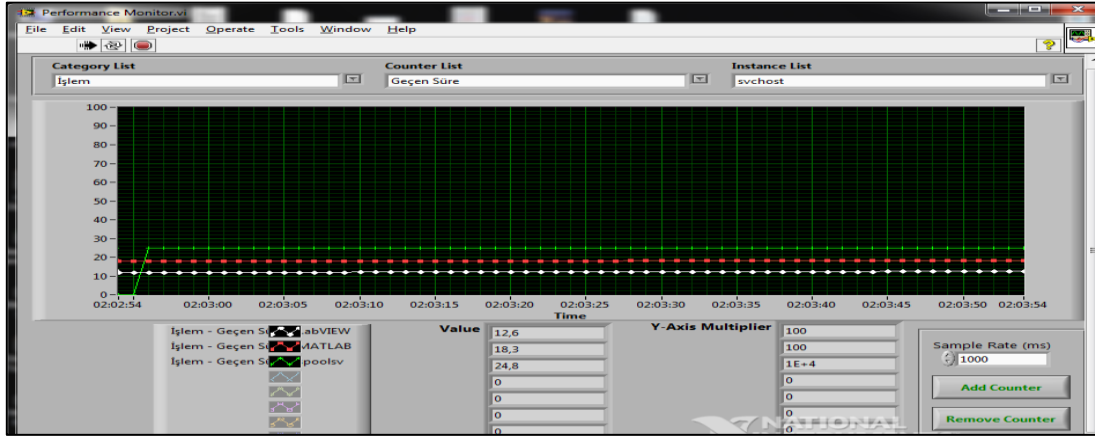
Test Case (FFT size)	32	64	128	256	512	1024	2048	4096	8192	16384	32768	65536	131072	262144	524288	1048576
1D FFT, Power of 2 (CDB)	1.61	1.73	2.51	3.62	4.03	4.18	2.22	3.08	2.74	3.5	6.85	15.62	15.1	16.97	21.97	15.24
1D FFT, Power of 2 (DBL)	1.32	1.59	2	2.59	2.91	3.22	3.39	2.82	3.03	3.25	3.38	4.32	4.17	4.89	5.45	6.21
1D Inverse FFT, Power of 2 (CDB)	1.63	2.17	2.6	3.6	4.18	3.7	2.57	2.58	2.69	3.57	4.79	13.58	14.47	18.66	18.13	19.96
1D Inverse FFT, Power of 2 (DBL)	1.79	2.12	2.37	3.1	3.58	4.06	4.46	3.44	3.63	3.65	3.49	3.61	4.22	4.38	5.17	6.55

Test Case (FFT size in each dimension)	16	32	64	128	256	512	1024
2D FFT, Power of 2 (CDB)	2.38	3.34	7.72	9.04	19.67	31.96	30.95
2D FFT, Power of 2 (DBL)	2.21	2.86	2.99	5.8	8.05	17.5	17.89
2D Inverse FFT, Power of 2 (CDB)	2.63	3.61	6.72	10.08	21.86	33.41	33.29
2D Inverse FFT, Power of 2 (DBL)	2.51	3.81	6.95	14.23	26.82	36.19	50.78

Test Case (signal length)	100	144	225	361	529	841	1296	2025	3249	5041	7744	12100	18789	29241	45389	70756	110224	171396	266256	413449	643204	1000000
1D Convolution (CDB)	2.73	3.34	2.79	3.05	3.07	3.37	1.44	2.16	2.03	2.6	2.29	2.84	2.78	3.03	3.85	3.76	3.59	4.15	4.19	4.47	4.03	4.11
1D Convolution (DBL)	2.79	3.24	3.02	1.82	3.15	1.97	3.11	2.98	1.55	1.48	2.04	1.92	2.11	2.34	1.5	2.52	2.61	2.49	1.45	2.18	2.49	2.3

Şekil 5.3. LabVIEW performans sonuç grafiği

LabVIEW programı kullanılarak yapılan bir performans testinde ise İşlem – Geçen Süre kriterine Şekil 5.4’deki sonuçlar alınmıştır:



Şekil 5.4. İşlem- geçen süre kriterine göre performans sonuç grafiği

Performansı en iyi olan program olarak LabVIEW belirlenmiştir. Daha sonrasında MATLAB ve FuzzyTECH programları gelmektedir.

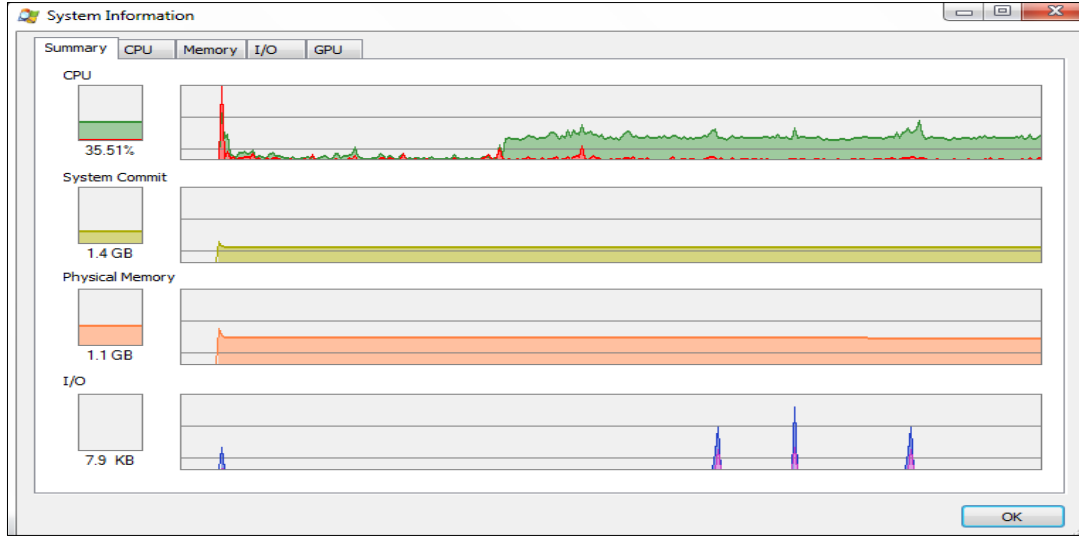
5.6. Sistem Kaynakları Kriterine Göre Programların Karşılaştırılması

Tez çalışmaları sırasında kullanılan programlar MATLAB, FuzzyTECH ve LabVIEW aynı anda açıldığında sistem kaynaklarını aynı oranda kullanmamaktadır. Process Explorer programı ile yapılan test esnasında sistem kaynaklarını en çok kullanan program olarak MATLAB görülmüştür. Daha sonra sırasıyla LabVIEW ve FuzzyTECH programları gelmektedir. Bu durum Şekil 5.4’de işlemci kullanımı kısmında görülmektedir.

Process	PID	CPU	Private Bytes	Working Set	Description	Company Name
csrss.exe	484	0.18	2.164 K	8.640 K		
winlogon.exe	716		1.696 K	1.388 K		
explorer.exe	1560	4.45	38.540 K	38.096 K	Windows Gezini	Microsoft Corporation
mssecss.exe	2660		3.972 K	2.276 K	Microsoft Security Client Use...	Microsoft Corporation
DTSShellHlp.exe	3912		3.556 K	3.252 K	DAEMON Tools Shell Extens...	DT Soft Ltd
wcmmon.exe	4344	< 0.01	1.452 K	1.008 K		
WINWORD.EXE	2208	0.02	42.136 K	60.488 K	Microsoft Word	Microsoft Corporation
taskmgr.exe	5160	0.46	2.920 K	9.884 K	Windows Görev Yöneticisi	Microsoft Corporation
fuzzyTECH.exe	7312	0.08	2.052 K	11.380 K	fuzzyTECH	INFORM GmbH
LabVIEW.exe	4744	0.05	69.596 K	91.984 K	LabVIEW 11.0 Development...	National Instruments Corpo...
MATLAB.exe	2036	0.33	185.868 K	187.488 K	MATLAB	The MathWorks Inc.
process-explorer-windows-xp-1.e...	6588	26.60	12.736 K	21.084 K	Sysinternals Process Explorer	Sysinternals - www.sysinter...

Şekil 5.5. Sistem kaynaklarının kullanımı

MATLAB 185 MB RAM kullanırken, LabVIEW 69 MB ve FuzzyTECH programını ise sadece 2 MB RAM kullanmaktadır. Programların üçü aynı anda açık olduğunda sistemin genel durumu Şekil 5.5’de görülmektedir.



Şekil 5.6. Programların aynı anda çalışması sırasında sistem durumu

5.7. Destek Kriterine Göre Programların Karşılaştırılması

Destek kriterini üç şekilde incelemek mümkündür. Programın yardım menülerinin ve örneklerinin yeterli olması, yardıma ulaşabilmek amacıyla oluşturulan internet gruplarının olması ve online yardım şeklinde yardıma ulaşılabilmesi. İncelenen programların hepsinde yardım menüsünün olduğu görülmüştür.

MATLAB hem yeni başlayanlara hem de uzmanlaşmış olanlara çok pratik bir yardım da sağlamaktadır. Özel bir komut hakkında bilgi edinmek için ekrandan "help + komut adı" komutu girilir; burada "komut adı" ilgilendiğiniz MATLAB fonksiyonunu temsil eden komutlardır. Kullanıcı, belirli bir konuda hangi komutları kullanabileceğini bilmek isterse "lookfor kavram adı" komutunu girmesi yeter. Örneğin, "lookfor sinus" yazılıp enter tuşuna basıldığında; MATLAB her biri için kısa açıklamalarıyla birlikte sinus'e ilişkin tüm komutları sıralar. Şayet aranan bilgi bir fonksiyon ise "help sin" komutunun girilmesi aşağıdaki pencerede gösterildiği gibi ilgili komuta ait ayrıntılı bir yardımla neticelenir [34].

>> lookfor sinus

FLATPLRS McBryde-Thomas Flat-Polar Sinusoidal Pseudocylindrical
Projection

MODSINE Tissot Modified Sinusoidal Pseudocylindrical Projection

SINUSOID Sinusoidal Pseudocylindrical Projection

Tüm yardım konuları MATLAB arama yolunda bir klasör ismiyle bağlantılıdır. Bir fonksiyon ile ilgili yardım alınmak istenirse help fonksiyon_adı şeklinde yazmak yeterlidir. Aynı zamanda MATLAB ile oluşturulan programların içerisine de MATLAB yardımı eklemek mümkündür. Teknik destek konusunda da MATLAB yeterli hizmeti sunmaktadır. Forumlar konusunda da MATLAB için açılmış çok sayıda forum sitesi ve cevap bulmak mümkündür.

LabVIEW programı içerisinde bulunan yardım sistemi LabVIEW programlama kavramı, LabVIEW kullanmak için adım adım komutlar ve LabVIEW VI'ler için fonksiyonlar, menüler, araçları barındıran bilgileri içerir. LabVIEW programı içerisinde bulunan tüm araçlar, modüller yada sürücüler için ayrı ayrı yardım menüleri bulunmaktadır. Tüm araçlar için pdf dosyası şeklinde kullanıcı el kitapları bulunmaktadır. Herhangi bir simge üzerine gelindiğinde o simge ile ilgili yardım menüsü ve örnek uygulama bulmak mümkündür. LabVIEW için açılmış çok sayıda forum sitesi vardır. Aynı zamanda kendi web sitesi üzerinden de forum hizmeti vermektedir. Teknik destek anlamında da LabVIEW üstün özellikler göstermektedir. Tüm bu açılardan bakıldığında LabVIEW programının en önemli özelliklerinden birisi olarak üstün yardım ve destek sistemi gösterilebilir.

FuzzyTECH programı içerisinde bulunan yardım menüsü kullanılarak bulanık mantık hakkında doyurucu bilgilere ulaşmak mümkündür. Aynı zamanda bulanık mantık örnek uygulamaları sayesinde yeni başlayanlar için kolaylık sağlanmaktadır. Forum yardımı olarak FuzzyTECH çok kullanılan bir program olmadığı için çok güçlüdür demek mümkün değildir. Teknik destek anlamında da FuzzyTECH diğer iki programda olduğu gibi web sitesi üzerinden teknik destek vermektedir. Sağlanan form aracılığı ile sorular kısa sürede cevaplanmaktadır. Destek kriteri değişik açılardan ele alındığında göre en az güçlü program olarak FuzzyTECH görünmektedir.

5.8. Eğitim Ortamlarında Kullanım Durumu Kriterine Göre Programların Karşılaştırılması

Herhangi bir sistemin benzetiminin oluşturulması için değişik paket programlar ya da programlama dilleri kullanılmaktadır. Bunlar arasında en önemli programlama paketlerinden birisi de MATLAB'dır. MATLAB, dünyada 2500'ü aşkın üniversite tarafından kullanılan en önemli ve yaygın yazılımlardan birisidir [35]. Son derece hızlı matematik işlemleri yürütebilmesi, gelişmiş araç kutuları, 2 ya da 3 boyutlu grafik oluşturmadaki başarısından dolayı, birçok araştırmacı ve teknik eleman, geliştirdikleri uygulamalar için MATLAB programından faydalanmaktadır. MATLAB bazı üniversitelerde başlı başına bir ders olarak anlatılmaktadır. MATLAB'ın 13. sürümü ile birlikte yeni bir araç kutusu tanıtılmıştır. MATLAB Web Sunucu (MWS) 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. [36]

MATLAB uzaktan eğitim ortamlarında da sıklıkla anlatılmaktadır. Örneğin Yıldız Teknik Üniversitesi 8 haftalık bir MATLAB kursunu uzaktan eğitim yoluyla vermektedir. [37]

LabVIEW programı MATLAB kadar yaygın olmamakla beraber eğitim ortamlarında kullanılmaktadır. Örnek olarak Sakarya Üniversitesi Elektronik Mühendisliği, İnönü Üniversitesi Biyomedikal Mühendisliği, ODTÜ Fizik, Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği gibi bölümlerde LabVIEW programı ders uygulama yazılımı olarak kullanılmaktadır. LabVIEW üzerine çalıştaylar yapılmakta, program daha geniş kitlelere yayılmaya çalışılmaktadır.

FuzzyTECH programı Bulanık Mantık, Bulanık Mantığa Giriş derslerinde MATLAB FIS Editor ile birlikte yazılım olarak kullanılabilir. Örneğin Yaşar Üniversitesi Endüstri Mühendisliği Bölümü'nde Bulanık Mantığa Giriş dersi şu şekilde tanıtılmaktadır:

- IENG 467 – Bulanık Mantığa Giriş (2, 2, 3, 4)

Bu ders bulanık kümeler ve bulanık mantığı tanıtarak ve endüstri mühendisliği alanındaki uygulamaları incelemeyi hedeflemektedir. Bu ders bulanık modellemenin gerekliliği, bulanık kümelerin ve bulanık sayıların tanımı ve matematiksel gösterimi, bulanık aritmetik, durulaştırma yöntemleri, Boolean mantık ve bulanık mantık, bulanık çıkarım yöntemleri, bulanık uzman sistemler ve bulanık karar verme gibi konuları içermektedir. Bu derste MATLAB, Fuzzy Logic Tool-Box, Neural Network Tool-Box ve FuzzyTech yazılımları kullanılacaktır.

FuzzyTECH programı için online eğitim yada uzaktan eğitim yoluyla herhangi bir kurs, seminer bulunamamıştır.

Görüldüğü üzere en yaygın kullanım alanı MATLAB programına aittir. LabVIEW yaygınlaşmakta ve gün geçtikçe kullanımı artmaktadır. FuzzyTECH programı ise sadece bulanık mantık üzerine özelleştirilmiş olduğundan bu ders için ek yazılım olarak görülmekte ve bu şekilde kullanılmaktadır.

5.9. Dokümantasyon Kriterine Göre Programların Karşılaştırılması

Ülkemizde genel olarak bakıldığında MATLAB, LabVIEW ve FuzzyTECH programlarıyla ilgili son derece geniş kapsamlı, içerik olarak her türlü ihtiyacı karşılayabilecek kapasitede dokümanların varlığından söz edilememektedir. Bunların nedenleri olarak programların üniversite ortamlarının dışında pek kullanılmaması olarak düşünülmektedir. Kaynakların çoğunluğu çeviri dokümanlar şeklinde görülmektedir. Ancak yine de karşılaştırma yapılırsa en çok doküman MATLAB programı ile ilgilidir.

MATLAB programı çok sayıda bilimsel çalışmada kullanılması, ders olarak okutulması ve geniş bir kullanım alanı bulmasından dolayı doküman bakımından diğer programlara oranla daha zengindir denilebilir. Kitap olarak düşünüldüğünde de MATLAB ile ilgili yazılmış kitapların sayısının LabVIEW ve FuzzyTECH programına göre oldukça fazla olduğu görülmektedir. Örneğin kitapyurdu.com sitesinde aratıldığında MATLAB ile ilgili 59 tane basılı kitap bulunmaktadır.

LabVIEW programı ile ilgili dili İngilizce olan 4 adet kitap bulunmuştur. FuzzyTECH programı ile ilgili kitap bulunamamıştır. Bu da göstermektedir ki; en çok Türkçe doküman kaynağı MATLAB programındadır.

Belge oluşturma açısından bakıldığında üç programın da belge oluşturma özelliği vardır. Bu özellikler programlara göre şu isimleri almaktadır:

MATLAB: Report Generator, FuzzyTECH: Documentation Generator, LabVIEW: Documentation

MATLAB programında doküman oluşturmak için report komutu kullanılır. MATLAB' da rapor oluşturmak için kullanılan komutlar şunlardır:

MATLAB Fonksiyonları

<u>report</u>	Rapor kurulum dosyasından yeni bir rapor oluşturur.
<u>rptconvert</u>	DocBook XML dosyalarını desteklenen doküman türlerine çevirir

Rapor Oluşturucu Bileşenleri

<u>Stop Report Generation</u>	Rapor oluşturmaya durdurur.
<u>Time/Date Stamp</u>	Rapora tarih ve saat bilgisi ekler.

LabVIEW programında her bir alt program için içerik yardımı oluşturulabilir. Bunu yapmak için Properties diyalog kutusundan Documentation menüsünü seçmek gereklidir. Aynı zamanda HTML tabanlı yardım dosyalarını da oluşturmak mümkündür. FuzzyTECH programında belge oluşturucu bileşeni vardır. Bu bileşen yardımıyla üzerinde çalışılan sistem zengin metin belgesi haline dönüştürülerek dokümantasyona yapılabilir.

5.10. Nesneye Yönelik Olma Durumu Kriterine Göre Programların Karşılaştırılması

Nesneye yönelik programlama, veri ve veriyle bağlantılı metotları nesne diye adlandırılan mantıksal yapılar içerisinde birleştirme yaklaşımıdır. Bu yaklaşım

yazılımın daha karmaşık bir yapıda olma yeteneğini artırır ve bu sayede büyük uygulamalar ve veri yapıları ile çalışmaya imkân sağlanmış olur.

MATLAB programının nesne tabanlı programlama kapasitesi karmaşık teknik hesaplama uygulamalarını C++, C# ve Java'dan daha hızlı yapmaya imkân sağlar. MATLAB programı kullanarak sınıflar tanımlanabilir ve standart nesne tabanlı tasarım uygulamaları gerçekleştirilebilir. MATLAB nesneye yönelik tasarım programlama özelliği şunları içerir:

- a) Özellik, metot ve olayları tanımlayabilecek yapıda sınıf tanımlama dosyaları
- b) Bağlantılı listelerde olduğu gibi veri yapıları oluşturma yeteneği
- c) Nesne özellik değiştirdiğinde anlık görüntüleme özelliği

Bu özellikleri sayesinde MATLAB programı ile nesneye yönelik tasarım yapılabildiği görülmektedir.

LabVIEW programı da nesneye yönelik tasarımı desteklemektedir. LabVIEW programının amacı olarak programlama konusunda çok iyi olmasa bile program ile çalışanlara kolayca program yazabileceği bir arayüz oluşturmaktır. C++ ile LabVIEW programlarında nesneye yönelik tasarım yapılması açısından karşılaştırıldığında LabVIEW programının daha üstün olduğu görülmektedir. Bunun en önemli nedenlerinden birisi LabVIEW programında tüm nesneler için sınıflar mevcuttur. C++ da böyle bir durum yoktur.

FuzzyTECH programı da nesneye yönelik kullanıcı arayüzü olan ve bulanık mantık programlama yapabilen bir programdır. Sonuç olarak incelenen üç program da nesneye yönelik program özelliğini destekler denilebilmektedir.

5.11. Model Kütüphanelerinin Genişliği Kriterine Göre Programların Karşılaştırılması

MATLAB,- Optimizasyon Kütüphanesi (Optimization Toolbox), Denetim Sistem Kütüphanesi (Control System Toolbox), Yapay Sinir Ağları Kütüphanesi (Neural Network Toolbox), Bulanık Mantık Kütüphanesi (Fuzzy Logic Toolbox) gibi giderek artan sayıda özelleştirilmiş kütüphaneleri içeren ve hala gelişmekte olan bir paket

programdır. Bu kütüphaneler kullanıcılara uzmanlık alanlarındaki uygulamalarının her birinde geniş imkânlar sağlamaktadır [35].

LabVIEW, programlama işleri esnasında kullanıcılara yardımcı olabilecek ve hafıza işgal etmeden kullanılabilen çok geniş bir fonksiyon ve alt program kütüphanelerine sahiptir. Geleneksel programlama dillerinde karşılaşılabileceğiniz gizli programlama problemleri ile LabVIEW’de daha az karşılaşılır. LabVIEW, Data Acquisition (DAQ) ve General Purpose Interface Bus (GPIB) için kod kütüphaneleri gibi spesifik kütüphane uygulamalarını ve seri aygıt kontrolü, veri analizi, veri sunumu, veri depolama ve internet üzerinden haberleşme gibi farklı uygulamaları da içerir. Analiz kütüphanesi; sinyal üretme, sinyal işleme filtreleri, Windows istatistikleri ve regresyonları, lineer cebir ve array aritmetiği gibi çok yönlü ve kullanışlı fonksiyonları bünyesinde barındırmaktadır [38].

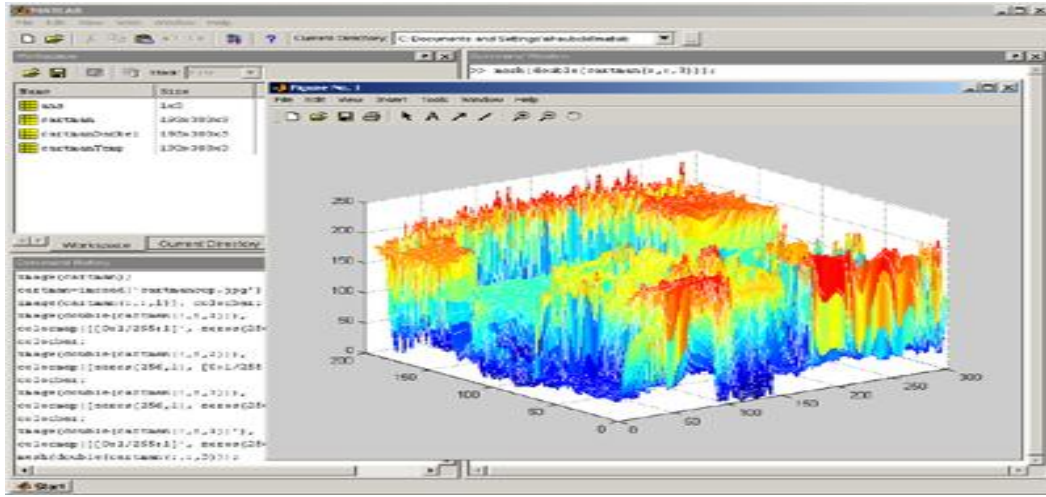
FuzzyTECH, bulanık algoritma uygulamaları için farklı model kütüphaneleri sunmaktadır. Bunlar FuzzyTECH Runtime DLL, FuzzyTECH Runtime Control, fuzzy TECH C Runtime Library, FuzzyTECH Assembler Runtime Library, FuzzyTECH COBOL Runtime Library... gibidir.

Tüm bu açılardan baktığımızda kapsam olarak daha büyük olan programlardan LabVIEW ve MATLAB programlarının model kütüphanelerinin de son derece geniş kapsamlı olduğu görülmektedir. FuzzyTECH programı ise bulanık mantık uygulamaları için gerekli olan kütüphane dosyalarını içermektedir. [39]

5.12. Sonuçların Analiz Edilme Kolaylığı Kriterine Göre Programların Karşılaştırılması

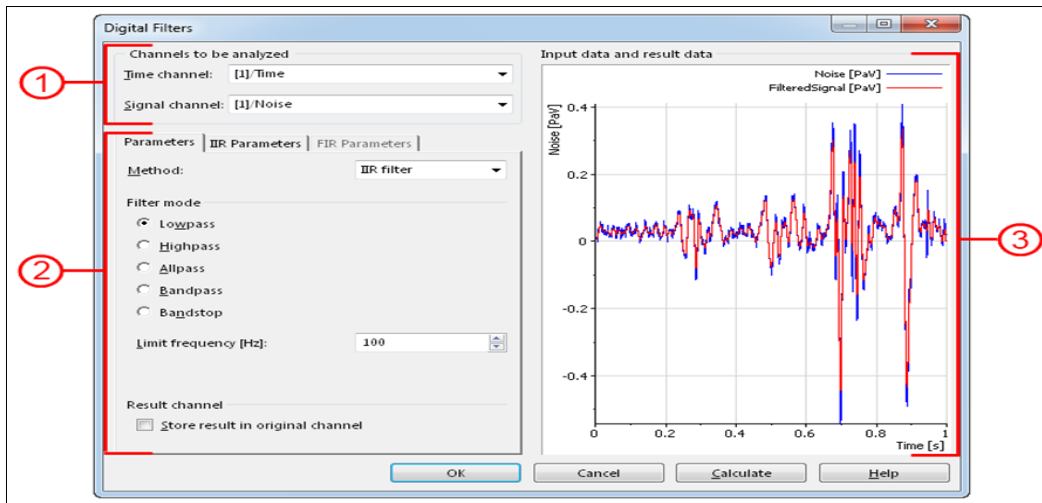
MATLAB verileri elde etmek, analiz etmek ve görselleştirmek için çeşitli araçlar bulundurmaktadır. MATLAB programı ile elde edilen sonuçlar MATLAB kodları şeklinde rapor haline getirilebilir ve yayınlanabilir. MATLAB’ da bulunan 2 ve 3 boyutlu çizim fonksiyonları olabildiğince görsellik sağlamakta, verilerin analizini kolaylaştırmaktadır. Bu fonksiyonlar kullanılarak veri görselleştirilebilir ve sonuçlarla bağlantı kolayca kurulabilir. Çizimler interaktif yada yapısal olarak özelleştirilebilir.

MATLAB’ da sonuçları analiz etmek için çeşitli renkler kullanılır. Bu renkler üç boyutlu görünüm esnasında kullanıcıya kolaylık sağlar ve analiz işlemini kolaylaştırır. Görüldüğü üzere sonuçları analiz etme kolaylığı açısından MATLAB üstün özellikleri olan bir programdır. Şekil 5.6 MATLAB analiz ekranını göstermektedir.



Şekil 5.7. MATLAB analiz ekranı

LabVIEW programı sonuçları analiz ederken eş zamanlı analiz yapılmasına olanak verir. LabVIEW aynı zamanda yüzlerce fonksiyon yardımıyla sonuçlar üzerinde raporlar oluşturmaya ve elde edilen sonuçlarla ilgili bilgi edinmeye olanak sağlar. LabVIEW programında sonuç analizi Şekil 5.7’de görüldüğü gibi yapılmaktadır:



Şekil 5.8. LabVIEW analiz ekranı

Birinci adımda analiz edilecek kanallar belirlenmelidir. İkinci adımda parametreler girilerek eş zamanlı sonuç analizi yapılabilmektedir. Üçüncü adımda ise giriş verileri ve sonuç verileri görülmektedir.

FuzzyTECH programında sonuçları analiz etmek için Analyzer menüsü bulunmaktadır. Bu menünün altında çizim seçenekleri vardır ve bu çizimler ile sonuçları analiz etmek mümkündür.

5.13. Genişletilebilirlik Kriterine Göre Programların Karşılaştırılması

MATLAB programının en önemli özelliklerinden birisi de genişletilebilir olmasıdır. MATLAB programı araştırma, tasarım, işaret işleme sistemleri için mühendislik uygulamaları gibi çok sayıda uygulamayı kullanıcıların kendilerinin yapabilmesini sağlar. Basitçe dosyalar kopyalanır, eğer isteniyorsa düzenlenir ve örneğin araç kutusunu genişletmek için yeni fonksiyonlar oluşturulabilir.

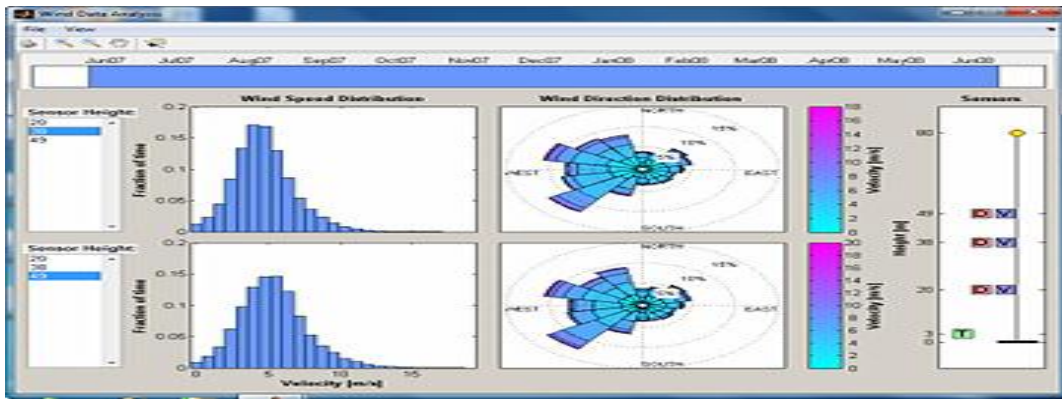
MATLAB programında bulunan program parçalarının çoğunun görüntülenme ve düzenlenmeye açık olduğunu söyleyebiliriz. Olmayan bir program parçası varsa eğer MATLAB bunu programcının oluşturmasına olanak sağlar. Tüm bunlar dikkate alındığında MATLAB programının sınırsız bir genişletilebilirliğe sahip olmadığı söylenebilir. Hatta fonksiyon ve program parçalarını makine üzerine kodlayarak da genişletme yapmak mümkündür [40].

LabVIEW programı da genişletilebilir bir programdır. LabVIEW programı ile oluşturulan alt programlar ve alt sistemler basitçe başka programlara eklenebilir. LabVIEW programının genişletilebilir veri depolama yapıları da vardır. Tüm bu bilgiler ışığında LabVIEW programının tamamen genişletilebilir ve özelleştirilebilir olduğunu söylemek mümkündür.

FuzzyTECH programı için de genişletilebilir olduğu söylenebilir. FuzzyTECH programı ile yapılan programlar C kodlarına dönüştürülebilir. Bu kodlar vasıtasıyla bulanık mantık denetleyici içerisine entegre edilebilir. Bu da yapılan uygulamaların gerçek dünya şartlarında çalışabilmesini sağlamaktadır.

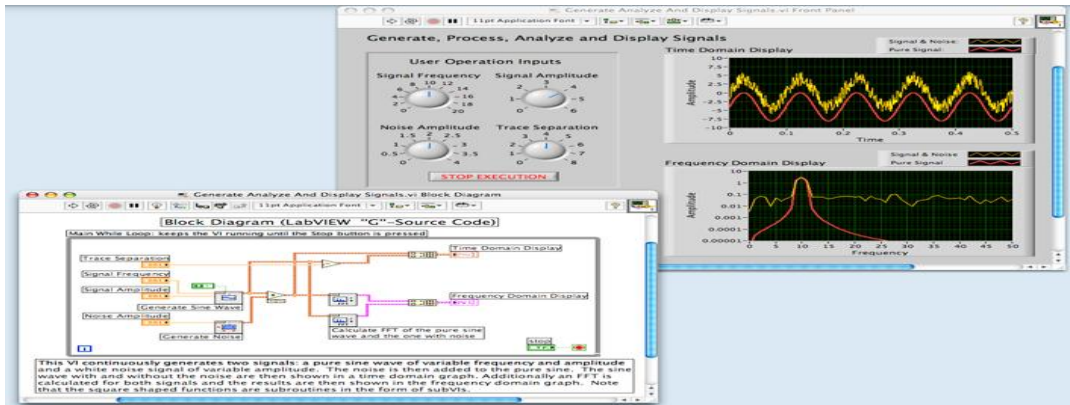
5.14. GUI oluşturma kriterine göre programların karşılaştırılması

Kullanıcı arayüzü (Graphics User Interface - GUI) kullanıcıya görsellik sağlayan bir işlevdir. Kullanıcı arayüzü sayesinde komut ezberleme ihtiyacı olmadan programı kullanmak mümkündür. GUI tasarımı FuzzyTECH programı içerisinde yoktur. MATLAB ve LabVIEW programları ile GUI tasarımı yapılabilir. MATLAB içerisinde bulunan GUIDE (GUI development environment – GUI geliştirme ortamı) aracı ile GUI tasarımı yapılabilir. MATLAB ile grafiksel olarak GUI tasarımı yapılır. MATLAB ile hazırlanmış örnek bir kullanıcı arayüzü Şekil 5.7’deki gibidir:



Şekil 5.9. MATLAB kullanılarak hazırlanmış GUI

LabVIEW programı teknik olarak GUI’lerle çalışır. LabVIEW ’da bulunan ön panel boş olsa bile bir GUI işlevi görmektedir. Grafik yaklaşımı sayesinde programcılar zaten aşına oldukları laboratuvar donanımlarını sanal temsilleriyle sürükleyip bırakarak programlarını oluştururlar. Şekil 5.8 örnek bir GUI göstermektedir.



Şekil 5.10. LabVIEW kullanılarak hazırlanmış GUI

5.15. Yeniden Kullanılabilirlik Kriterine Göre Programların Karşılaştırılması

İncelenen programlardan MATLAB, LabVIEW ve FuzzyTECH programları grafiksel nesne tabanlı programlardır. Grafiksel nesne tabanlı programlar yönetilme, ölçeklendirme ve yeniden kullanılabilme özelliklerine sahiptir.

Tüm nesnelerin kendilerine ait verileri vardır, böylece aynı sınıfta bulunan nesneler birbirini etkilemezler. Global değişkenlerin aksine paylaşılan bu veriler, çoklu nesneleri desteklemek için diske kaydedilme ve yeniden alt program yazılma gereği duymazlar. Bileşen içerisinde bulunan tüm veriler bileşeni kullanan kullanıcıları direkt olarak etkilemezler. Böylece tek bir bileşen çok sayıda uygulamada yeniden kullanılabilir. Uyumluluk sorunu olmamaktadır.

Yeniden kullanılabilirlik kriteri için LabVIEW ve MATLAB programlarının çok güçlü olduğu söylenebilir. FuzzyTECH programının sadece bulanık mantık işlemlerinde kullanılması ve içeriğinin çok güçlü olmaması gelecek uyumluluk sorunlarının doğmasına sebep olabilir.

BÖLÜM 6. SONUÇLAR VE DEĞERLENDİRME

Tez çalışması bulanık mantık üzerine yoğunlaştırılmış olsa da MATLAB, FuzzyTECH ve LabVIEW programları tüm yönleriyle detaylı bir şekilde incelenmiştir. İncelenen programlardan en çok kullanılanı MATLAB programıdır ve bunun nedenleri üzerinde durulmuştur. Önceki bölümde yapılan karşılaştırma sonuçları da gösteriyor ki en yaygın kullanım alanı olan, eğitim ortamlarında da en yaygın şekilde kullanılan program MATLAB programıdır. Ancak çalışma göstermiştir ki MATLAB alternatifsiz değildir ve özellikle bulanık mantık uygulamalarında aynı sonuç başka programlarla da alınabilir. Sadece bulanık mantık denetleyici tasarımı yapacak bir uzamanın maliyet esasını da düşünerek FuzzyTECH programına yönelmesi doğru olabilir.

LabVIEW programı özellikle elektronik mühendislik uygulamalarında kendisine yaygın bir şekilde kullanım alanı bulmaktadır. Kod yazma güçlüğü ortadan kaldırmış olması LabVIEW programının kolay kullanımı açısından yaygınlaşmasına sebep olarak gösterilebilir. Aynı zamanda yardım ve destek noktasında çok geniş çaplı bir olanak sunması da programın kullanımını artırmıştır. Ancak yine maliyet açısından bakıldığında incelenen programlar arasında en yüksek maliyet LabVIEW programındadır. Sadece bulanık mantık üzerine çalışacak bir uzmanın bu maliyeti göze alması durumunda çok görsel ve kullanışlı bir bulanık mantık tasarım ekranıyla karşılaşacağı görülmüştür.

FuzzyTECH programı sadece bulanık mantık tasarımı yapmak için kullanılan bir programdır. Bu programda bulanık mantık tasarımı yapmak kolaydır ve sonuçlar kolayca analiz edilebilir. Bu açıdan bakıldığında FuzzyTECH programının da özellikle bulanık mantık tasarımcıları arasında popüler olduğu söylenmektedir. Bu olumluluklara rağmen yeterli materyalin olmaması, eğitim ortamlarında neredeyse

hiç kullanılmıyor olması olumsuzluklardır ve bunlar da programa olan ilginin az olmasının sonuçlarındandır.

Programlar tüm yönleriyle karşılaştırılmıştır. 15 farklı kritere göre yapılan karşılaştırmada elde edilen sonuçlar Tablo 6.1’de verilmiştir.

Tablo 6.1 İncelen simülatörlerin değişik kriterlere göre durumu

KRİTER	MATLAB	LABVIEW	FUZZYTECH
Kurulum	ORTA	ORTA	GÜÇLÜ
Kullanım kolaylığı	ORTA	GÜÇLÜ	ÇOK GÜÇLÜ
Erişilebilirlik	GÜÇLÜ	GÜÇLÜ	ZAYIF
Görsellik	GÜÇLÜ	ÇOK GÜÇLÜ	ORTA
Performans	GÜÇLÜ	GÜÇLÜ	GÜÇLÜ
Sistem kaynaklarını kullanma	ORTA	ORTA	ÇOK GÜÇLÜ
Destek	ÇOK GÜÇLÜ	GÜÇLÜ	ORTA
Eğitim ortamlarında kullanım durumu	ÇOK GÜÇLÜ	GÜÇLÜ	ÇOK ZAYIF
Dokümantasyon	ÇOK GÜÇLÜ	ÇOK GÜÇLÜ	ORTA
Nesneye yönelik olma durumu	ÇOK GÜÇLÜ	GÜÇLÜ	ORTA
Model kütüphanelerinin genişliği	ÇOK GÜÇLÜ	ÇOK GÜÇLÜ	ÇOK ZAYIF
Sonuçların analiz edilme kolaylığı	GÜÇLÜ	GÜÇLÜ	ORTA
Geniştirilebilirlik ve yeniden kullanılabilirlik	GÜÇLÜ	GÜÇLÜ	GÜÇLÜ
GUI oluşturma	GÜÇLÜ	GÜÇLÜ	ZAYIF

Bulanık mantık tasarımı özelinde programlar arasında karşılaştırma yapılırken kullanılan ölçekler ÇOK GÜÇLÜ, GÜÇLÜ, ORTA, ZAYIF ve ÇOK ZAYIF’ tır. Kriterler puanlanarak bir sonuç elde edilmek istenmiş ve dilsel ifadeler aşağıdaki gibi rakamsal olarak ifade edilmiştir.

ÇOK GÜÇLÜ: 5, GÜÇLÜ: 4, ORTA: 3, ZAYIF: 2, ÇOK ZAYIF: 1

Araştırma sonucunda göz önünde bulundurulmuş kriterler rakamsal olarak ifade edildiğinde MATLAB programının 58, LabVIEW programının 57, FuzzyTECH programının 43 puan değeri aldığı görülmektedir. Bu sonuçlara bakarak bulanık mantık tasarımı yapılırken MATLAB ve LabVIEW programlarının yaklaşık olarak aynı ağırlıkta, FuzzyTECH programının ise bu iki programdan daha zayıf olduğu görülmektedir.

Bu sonuçlar bulanık mantık tasarımı yapılırken sadece MATLAB programının kullanılmasının bir zorunluluk olmadığını göstermektedir. Sadece bulanık mantık tasarımı yapılacaksa FuzzyTECH programını kullanmak da yeterli olmaktadır. LabVIEW programı bulunan bir eğitim ortamında da bulanık mantık tasarımı yapmak için ayrıca başka program yüklemek yerine LabVIEW bulanık mantık araçlarını yüklemek yeterli olmaktadır. Elde edilen sonuçlar MATLAB programının bulanık mantık tasarımlarında en güçlü program olduğunu göstermektedir. Bu açıdan bakıldığında eğitim ortamlarında çoğunlukla MATLAB programının neden kullanıldığı açıklanabilir.

Sonuç olarak sadece bulanık mantık tasarımı yapılacak ve maliyet düşünülüyorsa FuzzyTECH programı tek başına yeterli olmaktadır. MATLAB üzerine çalışılan bir ortamda MATLAB FIS Editör yüklenerek bulanık mantık tasarım ve simülasyonları gerçekleştirilebilir. Görsellik isteniyorsa ve LabVIEW programı ile çalışılıyorsa LabVIEW bulanık mantık araçları yüklenerek tüm bulanık mantık işlemleri sorunsuz bir şekilde gerçekleştirilebilir.

Ülkemizdeki bulanık mantık çalışmalarına destek olmak amacıyla tez çalışması sürecinde <http://www.bulanikmantik.com> web sayfası kurulmuş ve bilgi paylaşımı hedeflenmiştir. Bu web sitesinin ilerleyen zamanlarda daha da geliştirilmesine çalışılacaktır.

KAYNAKLAR

- [1] DERELİ T., Toplam Kalite Yönteminin Işığında Yapay Zekanın Endüstriyel Problemlerin Çözümünde Kullanımı Semineri, Gaziantep Üniversitesi Mühendislik Fakültesi, Gaziantep, 2000.
- [2] SARI M., MURAT Y. Ş., KIRABALI M., Bulanık Modelleme Yaklaşımı ve Uygulamaları, Dumlupınar Üniversitesi Fen Bilimleri Enstitüsü Dergisi, Sayı 9, Sayfa 2.
- [3] ÖZERK A., SİNECEN M. Klima Sistem Kontrolünün Bulanık Mantık ile Modellemesi, Pamukkale Üniversitesi Mühendislik Fakültesi Mühendislik Bilimleri Dergisi Cilt 10, Sayı 3, Sayfa 353-358 (2004).
- [4] ALTINTAŞ E., Bulanık Mantık, <http://www.yapay-zeka.org/modules/icontent/index.php?page=33>, Erişim Tarihi: 20.07.2012.
- [5] GÜÇLÜ S., Bulanık Mantık, <http://suleymanguclu.blogspot.com/2011/05/bulank-mantk.html>, Erişim Tarihi: 21.08.2012.
- [6] GÜN A., Bulanık Mantık Tasarım İlkeleri, <http://bulanikmantik.com>, Erişim Tarihi: 29.09.2012,
- [7] POLATLI A., Bulanık Mantık, <http://www.makaleler.com/bilim-makaleleri/bulanik-mantik.htm>, Erişim Tarihi: 11.11.2012.
- [8] ERKAYMAZ H., ÇAYIROĞLU İ., Bulanık Mantık ve PIC Kullanarak Bir Klima Sisteminin Kontrolü. Mehmet Akif Ersoy Üniversitesi Fen Bilimleri Enstitüsü Dergisi, Cilt 1, Sayı 2 (2010).
- [9] ETİK N., Ameliyathane Klima Kontrol Sistemleri İçin Bulanık Uzman Sistem Tasarımı (Y. Lisans Tezi), 2007, Selçuk Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Bilgisayar Sistemleri Eğitimi Anabilim Dalı, Sayfa 15.
- [10] SOY H., Gömülü Sistem Üzerinden Bulanık Kontrol ile Ev Otomasyonunu Gerçekleştirilmesi (Y. Lisans Tezi), 2006, Selçuk Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Bilgisayar Sistemleri Eğitimi Anabilim Dalı, Sayfa 2.

- [11] KARADENİZ S., Çoklu Kompresör Sisteminin Bulanık Mantık ile Kontrolü (Yüksek Lisans Tezi), Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Elektrik Elektronik Mühendisliği Anabilim Dalı, Sayfa 3.
- [12] BOLAT Y, Matlab-Simuink + Pic Tabanlı Bulanık Mantık Denetleyici Tasarımı ve Gerçek Zamanlı Sıcaklık Kontrolü Uygulaması (Yüksek Lisans Tezi), Marmara Üniversitesi Fen Bilimleri Enstitüsü, 2006, Sayfa 2.
- [13] NAKORNNAYOK A., TIPSUWANPORN V., TEANTHONG T., DC Motor Speed Control using Fuzzy Logic based on LabVIEW, Department of Electronics Engineering, Srinakharinwirot University, Nakornnayok, 2006.
- [14] MUHAMAD N. A., ALI S.A.M., LabVIEW with Fuzzy Logic Controller Simulation Panel for Condition Monitoring of Oil and Dry Type Transformer, International Journal of Mathematical and Computer Sciences 1:4 2005.
- [15] AKIN G., KETENCİ S., Sistem Odaları İklimlendirme Sistemleri, Trakya Üniversitesi Fen Bilimleri Enstitüsü / Bilgisayar Mühendisliği Bölümü 2010, Sayfa 15.
- [16] ZADEH, L. A., Commercialism and Human Values, Azerbaijan International, Spring 1998 (6.1).
- [17] IŞIKLI Ş., Bulanık Mantık ve Bulanık Teknolojiler, Ankara Üniversitesi Dil ve Tarih-Coğrafya Fakültesi Felsefe Bölümü Dergisi, Cilt: 19, Sayfa: 105-126.
- [18] ALTAŞ H., Bulanıklık Kavramı. Enerji Elektrik Elektromekanik-3, Bileşim Yayıncılık, 1999, s.80-85.
- [19] YARALIOĞLU K., Bulanık Mantık, www.deu.edu.tr/userweb/k.yaralioglu/dosyalar/bul_man.doc, Erişim Tarihi: 10.05.2012.
- [20] TRIANTAPHYLLOU E., Multi-Criteria Decision Making Methodologies: A Comparative Study, volume 44 of Applied Optimization. Kluwer Academic, Dordrecht, 2000.
- [21] LOOTSMA F.A., Fuzzy Logic for Planning and Decision Making, Springer, 2012.
- [22] TERANO T., ASAI K., SUGENO M., Applied Fuzzy Systems, AP Professional, Michigan University, 1994.
- [23] TÛTMEZ B., Bulanık Mantık ve Eğitim Bilimlerinde Kullanılabilirliği, Eğitim Dergisi, Sayı: 18, Şubat 2008.

- [24] Sakarya Üniversitesi - Eğitim Öğretim Bilgi Sistemi, http://www.ebs.sakarya.edu.tr/ebs_2012, Erişim Tarihi: 18.11.2012.
- [25] Simülasyon nedir?, <http://simulasyon.nedir.com/#ixzz2C5axe3z1>, Erişim Tarihi: 10.09.2012.
- [26] Fuzzy Control Language, http://en.wikipedia.org/wiki/Fuzzy_Control_Language, Erişim Tarihi: 02.01.2013.
- [27] fuzzyTECH, <http://www.fuzzytech.com>, Erişim Tarihi: 30.08.2011.
- [28] LabVIEW System Design Software, <http://www.ni.com/labview>, Erişim Tarihi: 20.04.2012.
- [29] MATLAB, The Language of Technical Computing, www.mathworks.com/products/matlab, Erişim Tarihi: 01.11.2012.
- [30] BİRTANE S., YILMAZ U., Matlab Fuzzy Logic Toolbox İnceleme ve GUI Uygulaması Tasarımı, Web Project Sistemi Dinamik Web Tabanlı Proje Yayınlama Sistemi, 2009.
- [31] SEÇGİN A., Bilgisayar neden ısınır, nasıl soğutulur?, <http://bilgi-n.com/318/bilgisayar-neden-isinin-nasil-sogutulur>, Erişim Tarihi: 24 Ekim 2011.
- [32] Sunucu (bilişim), [http://tr.wikipedia.org/wiki/Sunucu_\(bilişim\)](http://tr.wikipedia.org/wiki/Sunucu_(bilişim)), Erişim Tarihi: 20 Nisan 2012.
- [33] Veri merkezi, http://tr.wikipedia.org/wiki/Veri_merkezi, Erişim Tarihi: 11 Mart 2012.
- [34] AYDIN H., Matlab ile Kontrol Sistemlerinin İncelenmesi, Marmara Üniversitesi (Bitirme Tezi), 2003, Sayfa 11.
- [35] URAN, S., JEZERNIK, K., “MATLAB Web Server and M-file Application”, 12th International Power Electronics and Motion Control Conference, Portoroz, Slovenia, 2088-2092, Ağustos 2006.
- [36] IRMAK E., E-Öğrenme Ortamları için Matlab Web Sunucu Kullanımı, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, Cilt 23, No 2, 495-506, 2008.
- [37] MATLAB EĞİTİMİ, Yıldız Teknik Üniversitesi, http://www.sem.yildiz.edu.tr/kisisel_egitim_detay.php?pID=24&eID=141, Erişim Tarihi: 20.10.2012.

- [38] Herkes İin LabVIEW ISBN 978-605-364-301-2, Do. Dr. Ahmet Mavi Eylül 2012.
- [39] User's Manual for all FuzzyTECH 5.3 Editions, <http://www.fuzzytech.com>, Eriřim Tarihi: 08.09.2012.
- [40] Mansoo R., Image Processing in Java, <http://tr.scribd.com/doc/55625177/Thesis>, Eriřim Tarihi: 23.11.2012.

EKLER

EK 1. BULANIKMANTIK.COM WEB SİTESİ

Ülkemizde bulanık mantık çalışmalarını desteklemek amacıyla kurulmuştur. Sitenin içeriği bulanık mantık üzerine çalışan herkesin görüş alışverişinde bulunabileceği bir platform olarak tasarlanmıştır. Tez çalışmasının devamında yapılacak olan çalışmaların internet ortamında da bulunması gerektiği düşüncesinden hareketle Kasım 2012 tarihinde yayınlanmaya başlanmıştır. Sitenin amacı ülkemizde bulanık mantık çalışmalarına destek olacak her türlü imkanı sağlamaktır.

ÖZGEÇMİŞ

Barış TAÇYILDIZ, 20.02.1981 tarihinde Ankara' da doğdu. İlk, orta ve lise eğitimini Ankara'da tamamladı. 2000 yılında başladığı Kocaeli Üniversitesi Bilgisayar Öğretmenliği bölümünü 2004 yılında bitirdi. 2004 yılında Milli Eğitim Bakanlığı'na bağlı devlet okullarında öğretmenliğe başladı. Halen İMKB Sakarya Teknik ve Endüstri Meslek Lisesi'nde Bilişim Teknolojileri alanı öğretmenliği yapmaktadır.