

ÖZET
YÜKSEK LİSANS TEZİ

BULANIK MANTIK PROBLEMLERİ İÇİN TÜRKÇE GÖRSEL
BİR ARAYÜZ TASARIMI

AGÂH TUĞRUL KORUCU
SELÇUK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRONİK VE BİLGİSAYAR SİSTEMLERİ
EĞİTİMİ ANA BİLİM DALI

Danışman : Prof. Dr. Novruz ALLAHVERDİ
2007, 131 Sayfa

Juri : Prof. Dr. Novruz ALLAHVERDİ
Prof. Dr. Ahmet ARSLAN
Doç. Dr. Hakan IŞIK

Çalışmada bulanık kontrol problemleri için dili Türkçe olan görsel bir arayüz tasarlanması problemi ele alınmaktadır. Böyle bir arayüzün çeşitli bulanıklaştırma, çıkarım ve durulaştırma yöntemlerinin seçilebilmesi imkanının olmalısı, giriş ve çıkış değerlerinin bulanıklık derecelerinin grafiksel görünümünün elde edilebilmesi, kullanım dilinin tamamen Türkçe olması karşımıza bir amaç olarak konulmuştur.

Bu problemi çözebilmek için önce mevcut arayüzler incelenmiş ve bunların avantaj ve dezavantajları belirlenmiştir. Bu arayüzlerin Türkçe olmaması, kısıtlı sayıda bulanıklaştırma, çıkarım ve durulaştırma yöntemlerinin kullanılabilmesi ve bazı yazılımların görsel olmaması gibi dezavantajları ortadan kaldırmak için görsel programlama dilinde yazılmış bir arayüzün tasarlanması öngörülmüştür.

Yapılan inceleme ve çalışma sonucunda kullanıcının belirlediği giriş ve çıkış değerlerini alıp bulanıklaştıran, yine bulanık kuralların girilmesi sürecini görsel olarak yapan, uygun çıkarım mekanizmasını seçmeye imkan veren, kuralları ateşleyen ve çıkış parametrelerinin hangi yöntemle durulaştırılacağını belirlemeye olanak tanıyan görsel bir arayüz tasarlanmıştır. Tasarlanmış olan sistem daha önce yapılmış olan bulanık kontrol problemleri üzerinde test edilmiştir. Bu testler tasarlanmış olan yazılımın uygun sonuçlar verdiğini göstermektedir. Bu sistem araştırmacıya kendi problemini çözerken daha çok bulanıklaştırma, çıkarım ve durulaştırma yöntemleri seçme olanakları sağlamakta ve bununla daha hassas sonuçlara varmaya olanak tanımaktadır.

Anahtar kelimeleri: bulanık mantık, bulanık kontrol, bulanıklaştırma, bulanık kurallar, çıkarım mekanizması, durulaştırma, bulanık uzman sistem, bulanık arayüz, arayüz.

ABSTRACT

M.S. Thesis

DESIGN OF A TURKISH VISUAL INTERFACE FOR FUZZY LOGIC PROBLEMS

AGÂH TUĞRUL KORUCU
SELÇUK UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
COMPUTER SYSTEM EDUCATION BRANCH

Advisor : Prof. Dr. Novruz ALLAHVERDI
2007, 131 pages

Jury : Prof. Dr. Novruz ALLAHVERDI
Prof. Dr. Ahmet ARSLAN
Doç. Dr. Hakan Işık

The problems of visual interface projection for fuzzy control problems, whose language is Turkish, is concerned in this work. Interface like this must be possibility of various fuzzyfication, choosing inference and defuzzyfication method.

Besides, it's entrance and exit value of fuzzyness ranks of graphical view can be obtained and being it's using language completely Turkish is given us as a goal.

To solve this problem, first of all existing interfaces are examined and their advantages and disadvantages are determined. To remove disadvantages like interfaces being not Turkish, restricted fuzzyfication, wrong inference and defuzzyfications methods and some softwares being not visual, projection of interface which is coded at visual programming language is envisaged.

At the end of examine and study, a visual interface which is fuzzyficate entrance and exit rank determined by user, make process of entering fuzzy rules as a visual, enable choosing suitable inference mechanism, activate rules and enable exit parameters defuzzyfication with which methods is projected. The system which is projected, tested on

fuzzy control problems being made at first. This tests shows that projected software give suitable results. This system enable more fuzzyfication when solving his/her problems to investigator and give him/her possibility of choosing inference and defuzzyfication methods and it enable to take more sensible results with this possibilities.

Key words: Fuzzy Logic, Fuzzy Control, Fuzzfication, Fuzzy Rules, Defuzzyfication, Fuzzy Expert System, Fuzzy Interface, Interface.

1. GİRİŞ

Gün geçtikçe artan daha hassas cihazlar ve artan ihtiyaçlar bir problemin çözülmüş olan sonuçlarının yorumunu daha kesinden daha da genel bir ifadeye doğru götürmektedir. Bilgisayarları da bu kararları vermede kullanmak için ve sistem otomasyonlarının tüm kullanımlarını bilgisayarlara vermek için bilgisayarların gelen giriş verilerini yorumlayıp bu giriş deki verilerin hassasiyetlerine göre karar vererek sonuçları çıkış biriminden otomasyona göndermektedir. İşte bu alanda ihtiyaçları gittikçe artan yapay sinir ağı, bulanık mantık, yapay zekâ, bulanık sinir ağı ve benzer yöntemlerin ortaya çıkarılmasına neden olmuştur.

Sonuçta insan ya da uzman gibi düşünen ve giriş verilerini yorumlayıp çıkış ünitesine gönderen bir bilgisayar sistemini ortaya konulması hedeflenmektedir.

Bulanık mantık bulanık küme teorisine dayanır. Bulanık mantığa Bulanık kümeleri ve bulanık bağıntıları kullanan sonsuz değerli mantık demek bile mümkündür. Bulanık mantığın doğruluk tabloları ve çıkarım kuralları belirsizlik içerir, doğru ve yanlışla yüklenen anlamlara olduğu kadar, bu anlamları güçlendirmek ya da zayıflatmakta kullanılan niceleyicilere yüklenen anlamlara da bağlıdır (Baykal, 2004).

Günümüzde birçok alanda yapay zekâ, uzman sistem, bulanık mantık, bulanık sinir ağı teknolojileri ile yapılmış sistemler mevcuttur.

Bu çalışmanın birinci bölümünde bulanık mantığın tanımı, kim tarafından bulunduğu, nasıl geliştiği verilmiş ve bulanık mantığın alt bölümleri anlatılmıştır. İkinci bölümde şu anda internette bulunan, kitaplarda bulunan ve çalışmalarda kullanılmış otuzu aşkın bulanık kontrol programı incelenmiştir. Üçüncü bölümde, incelenen programların avantaj ve dezavantajları araştırılmıştır. Dördüncü bölümde sonuç ve öneriler verilmektedir. Beşinci bölüm yararlanılan kaynakların listesini içermektedir.

Çalışmanın amacı; hali hazırda var olan bulanık kontrol işlemlerini yapabilen bilgisayar programlarının incelenerek, bu programların avantaj ve

dezavantajlarının ortaya çıkarılması ve bulanık kontrol problemleri için dili Türkçe olan görsel bir arayüz tasarlanması problemi olarak belirlenmiş, bulanık mantık problemleri için Türkçe görsel bir arayüz tasarlanmıştır. Bu çalışma sonucunda oluşturulan bu arayüzün diğer arayüzlere göre daha çok sayıda bulanıklaştırma yöntemi, daha çok sayıda çıkarım mekanizması, daha çok sayıda durulaştırma yöntemlerinin seçebilmesi ve daha çok sayıda giriş ve çıkış parametresi belirleme imkanı olmuş, giriş ve çıkış değerlerinin, dilsel ifade parametrelerinin grafiksel görünimleri elde edilebilmiş, bulanık kurallar daha çok sayıda ve daha kolay girilebilmiş, kullanım dili tamamen Türkçe olarak yeni bir bulanık arayüz oluşturulmuştur.

1.1 Tezin Organizasyonu

Bu çalışmanın birinci bölümünde konuya giriş yapılmış, çalışmanın genel olarak ne olduğu ve ne amaçla üretildiği anlatılmıştır. İkinci bölümde bulanık mantık kısa olarak anlatılmış ve bulanıklaştırma yöntemleri, çıkarım metotları, durulaştırma yöntemleri biraz daha ayrıntılı olarak anlatılmıştır. Üçüncü bölümde literatür taraması yapılarak şu anda var olan programların avantaj ve dezavantajları karşılaştırılmış, bulanık bir yazılım için nelere ihtiyaç olduğuna karar verilmiştir. Dördüncü bölümde yapılan literatür taraması sonucu olarak şimdiye kadar yapılmış olan programların avantaj ve dezavantajları tablosu oluşturulmuştur. Beşinci bölümde tasarlanmış olan arayüz ve kullanımı anlatılmıştır. Arayüz anlatılırken kullanıcının arayüzü çalıştırdığında karşısına gelecek ekranları adım adım izleyip bu ekranlar üzerinde neler yapması gerektiği gösterilmektedir. Altıncı bölümde arayüz test edilmiş, gösterilmiş ve sonuçları verilmiştir. Bu testler yapılırken daha önce yapılmış ve sonuçları denenmiş sonuçlardan faydalanılmıştır. Yedinci bölümde çalışma sonucunda biriken bilgi birikimi ile yapılmış olan sonuç ve öneriler bölümü

vardır. Sekizinci bölümde tezin oluşumunda ve arayüzün yazımında kullanılan kaynaklar ve literatürlere ulaşmak için kaynakça bölümü vardır.

1.2 Çalışmanın Amaç ve Önemi

Bu çalışmaya otuzu aşkın bulanık kontrol ve bulanık mantık problemleri için hazırlanmış olan arayüzlerin incelenmesi ile başlanılmıştır. Bu oluşturulmuş arayüzler incelenirken, onların avantaj ve dezavantajları ortaya çıkarılmıştır. Bu arayüzlerin bulanıklaştırma yöntemlerinin sınırlı oluşu, çıkarım yöntemlerinin az sayıda olması, durulaştırma yöntemlerinin seçme imkanlarının az olması, kurallarının istenilen şekilde belirlenememesi ve arayüze eklenememesi, giriş ve çıkış parametrelerinin sınırlı sayıda olması, dilsel ifadelerin sınırlı sayıda olması, görselliklerinin sınırlı olması ve asıl önemlisi de Türkçe görsel bir arayüzün olmaması karşımıza çıkmaktadır. Bu arayüzlerin incelemeleri sonucunda bu özellikler dezavantajlar ortaya konulmuş ve bu çalışma ile bu dezavantajları minimuma indirmek amaçlanmıştır.

Tasarlanmış olan arayüzün önemli özellikleri aşağıda gösterilmektedir:

- Seçilebilen durulaştırma yöntemlerinin sayısı - 10;
- Seçilebilen bulanıklaştırma yöntemlerinin sayısı – 4 (üçgen, S-vare, ters S-vare ve trapezoidal);
- Seçilebilen çıkarım mekanizması sayısı – 7;
- Dilin Türkçe olması;
- Görsellik (giriş ve çıkış parametrelerinin bulanıklaştırılmış, durulaştırılmış grafikleri);
- Giriş parametrelerinin sayısı – 10;
- Çıkış parametrelerinin sayısı – 10.

2. Bulanık Mantık

2.1 Bulanık Mantığa Giriş

Bulanık mantık (Fuzzy Logic) kavramı ilk kez 1965 yılında California Berkeley Üniversitesinden Prof. Lotfi A.Zadeh'in bu konu üzerinde ilk makalelerini yayınlamasıyla duyuldu. O tarihten sonra önemi gittikçe artarak günümüze kadar gelen bulanık mantık, belirsizliklerin anlatımı ve belirsizliklerle çalışılabilmesi için kurulmuş katı bir matematik düzen olarak tanımlanabilir. Bilindiği gibi istatistikte ve olasılık kuramında, belirsizliklerle değil kesinliklerle çalışılır ama insanın yaşadığı ortam daha çok belirsizliklerle doludur. Bu yüzden insanoğlunun sonuç çıkarabilme yeteneğini anlayabilmek için belirsizliklerle çalışmak gereklidir (Allahverdi, 2006).

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

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

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

Bulanık mantığın uygulama alanları çok geniştir. Sağladığı en büyük fayda ise "insana özgü tecrübe ile öğrenme" olayının kolayca modellenebilmesi ve belirsiz kavramların bile matematiksel olarak ifade edilebilmesine olanak tanınmasıdır. Bu nedenle lineer olmayan sistemlere yaklaşım yapabilmek için özellikle uygundur. Bulanık mantık konusunda yapılan araştırmalar Japonya'da oldukça fazladır. Özellikle fuzzy process controller 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'de araştırma gelişme kısmında bulanık mantık konusunda çalışmalar yapılmaktadır. Yine bir başka uygulama olarak otomatik cıvatalamaların değerlendirilmesinde bulanık mantık kullanılmaktadır. Bulanık mantık yardımıyla cıvatalama kalitesi belirlenmekte, cıvatalama tekniği alanında bilgili olmayan kişiler açısından konu şeffaf hale getirilmektedir. Burada bir uzmanın değerlendirme sınırlarına erişilmekte ve hatta bu uzmanın değerlendirme sınırları geçilmektedir.

Bir olay veya bir sistem miktarı karakteristiklerle sunulmayınca bu olayın veya sistemin iyi anlaşıldığı sayılamaz olması modern bilimin temel prensiplerinden biridir. Bu açıdan bakıldığında bilimsel bilginin özünü teşkil eden bileşenlerin çoğuna, bu bileşenlerin davranışları hakkında miktarı enformasyon almağa olanak tanıyan ve çeşitli sistemlerin matematik modellerini oluşturmak için gereken prensip ve yöntemler toplusu gibi bakılabilir.

Bilgisayarların geniş kullanılmağa başlanması ile insan bilgisinin birçok alanında miktarı yöntemler daha hızlı yaygınlaşmağa başladı. Burada bilgisayarların hiç şüphesiz, davranışları mekanik, fizik, kimya ve elektromagnetizma kanunları ile belirlenen mekanistik sistemlere uygulanışı çok efektif olmuştur ve olmaktadır. Maalesef, aynı şeyi humanistik (insancıl) sistemler hakkında söyleyememekteyiz

(Humanistik sistemler, davranışına insanın muhakemelerinin, çevreyi algılamasının veya emosyonlarının etki yaptığı sistemlerdir. Örneğin; ekonomik, politik, hukuk, eğitim vb. sistemleri).

Bu sistemler şimdiye kadar matematik analize ve bilgisayar modellemeye büyük direnç göstermişlerdir.

Humanistik sistemlere bilgisayarların uygulanmasındaki başarısızlık, hesaplamalardaki yüksek hassasiyet isteğinin ve bunun yüksek hızla yapılması isteğinin uyuşmazlığından ileri geldiği bazı bilim adamları tarafından gösterilmektedir. Diğer bir deyimle, sistemin karmaşıklığı ve bu karmaşıklığı analiz etmek için kullanılan hassasiyet ters orantılıdır. Buradan, humanistik sistemlerin davranışı hakkında önemli sonuçlar alabilmek için hesaplamalardaki yüksek hassasiyet ve kesinlikten kaçınmak gerektiği sonucuna varılabilir. Bu yüzden humanistik hesaplamalarda çok da kesin olmayan kendi tabiatı itibarı ile tahmini olan diğer yöntemlerin kullanılmasına da yol vermek gerekmektedir.

Çok büyük karmaşıklık karşısında hassasiyeti kurban ederken, değerleri sayılar değil, sözler veya cümleler olan dilsel değişkenleri kullanmak imkânının öğrenilmesi bu durumda tabiidir. Sayısal değişkenleri değil dilsel değişkenleri kullanmak bu değişkenlerin daha somut olması ile ilgilidir. Örneğin; “Ahmet uzun boyludur” ifadesi Ahmet’in boyu 1m 85cm’dir ifadesinden daha az somuttur. Bu durumda **uzun** kelimesi **boy uzunluğu** değişkeninin bir dilsel değeri olarak ele alınabilir. Başka bir örneğe bakalım. “Ahmet gençtir” ve “Ahmet’in yaşı 25’tir” ifadelerinde **gençtir** dilsel değerdir. Bu değer ikinci ifadedeki 25 sayısı ile aynı rolü oynamaktadır. Aynı şeyleri **çok genç, genç olmayan, çok çok genç, çok da genç olmayan** vs. dilsel değerleri hakkında da söylenebilir. Bu durum da bu dilsel değerlerin de arkasında net olmayan bir sayısal değer mevcuttur.

Sayısal değişken değerleri grafiksel olarak bir düzlemde noktalar yardımı ile gösterilir, ama dilsel değerler düzlemde bulanık sınırları olan sahalar şeklinde gösterilir. Değişkenleri bu yöntemle sunmanın, yani noktalarla (veya hatlarla) değil, sahalarla sunmanın yardımı ile olay yaklaşık bir sunulma aracı bulmuş olur. Böylece, dilsel değişkenlerin hesaplanmalarına bir olanak tanınır ve bu geniş kullanım bulur.

Hesaplamaların böyle yapılabileceği ilk defa 1965 yılında Azeri kökenli ABD’ li bilim adamı Lotfi Ali Asker-Zadeh tarafından yayınlanmıştır. “Bulanık

Kümeler” olarak adlanan bu makalede o, matematiğin, dil ve insan zekâsını ilişkilendirebileceğini göstermiş ve bunun için **bulanık kümeler teorisini** teklif etmiştir. Zadeh birçok kavramın dilsel olarak geleneksel matematiğe göre daha iyi belirlenebildiğini ve bulanık mantığın ve onun bulanık kümelerdeki ifadelerinin gerçek hayatın daha iyi modelini oluşturduğunu göstermiştir.

Bulanık mantık teorisini ilk defa 1972 yılında İngiltere’de Ebrahim Mamdani, bir buhar makinesi için kontroller tasarlayarak kullandı. Bundan sonra Danimarka’ da çimento sanayisindeki uygulama bu yöntemin avantajlarını gösterdi. Bundan sonra bulanık mantığın en çok uygulandığı ülke Japonya oldu. Japon bilim adamları ve mühendisleri bulanık mantığı metroda, otomatik tren kontrolü, hisse senedi portföyü, asansör vs. birçok alanda kullanmışlar ve bundan büyük ekonomik kazançlar elde etmişler. Bugün Japonya’da bulanık mantık kullanılmayan beyaz eşya çeşidi yoktur (Allahverdi, 2006).

2.1.1 Bulanık Teorinin Avantajları

İnsan düşünme tarzına yakın olması,
Uygulanışının matematiksel modele ihtiyaç duymaması,
Yazılımın basit olması dolayısıyla ucuza mal olması.

2.1.2 Bulanık Teorinin Dezavantajları

Uygulamada kullanılan kuralların oluşturulmasının bulanika bağlılığı,
Üyelik fonksiyonlarının deneme – yanılma yolu ile bulunmasından dolayı uzun zaman alabilmesi,

Kararlılık analizinin yapılışının zorluğu (benzeşim yapılabilir) (Allahverdi, 2006). Tablo 2.1’de insan boy uzunluklarına ait tablo verilmektedir.

Tablo 2.1 İnsan boy uzunlukları, uzun adam örneği

İsim	Boy uzunluğu, cm	Üyelik dereceleri	
		Kesin	Bulanık
Chris	208	1	1.00
Mark	205	1	1.00
John	198	1	0.98
Tom	181	1	0.82
David	179	0	0.78
Mike	172	0	0.24
Bob	167	0	0.15
Steven	158	0	0.06
Bill	155	0	0.01
Peter	152	0	0.00

2.2 Bulanık Küme Kavramı

Elemanları x olan bir X evrensel (universal) küme düşünelim. Bu elemanların $A \subset X$ alt kümesine aitliği, yani bu altkümelerin elemanı olup olmadığı X in $\{0,1\}$ 'de olan **karakteristik fonksiyonu** olarak belirlenir. Yani

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

Bu teoride nesneler bir kümeye ne kadar aittir. Aitlik derecelendirilmiştir. Kümeye aitlik üyelik dereceleri (membership grades) ile verilir.

Örnek: Meyveler kümesini ele alalım. Elma bu kümeye ait olduğu için $\mu_{\text{meyve}}(\text{elma})=1$ ve kavun ise bir sebze olduğundan dolayı bu kümeye ait değildir ($\mu_{\text{meyve}}(\text{kavun}) = 0$).

İki değerle değerlendirilen bu tür kümeler **kesin (grips)** kümeler olarak adlandırılmaktadır. Geleneksel bilgisayarlar iki değerli kümeyi kullanarak ikili mantıktan geniş yararlanmaktadır.

Fakat gerçek hayatta bir nesnenin bu veya diğer bir kümeye aitliği tam kesinlik göstermeyebilir. Örneğin masanın üzerinde bir tabak elma olduğunu düşünelim. Bu durumda tabaktakiler elmalar mı? Sorusuna –evet, armutlar mı? Sorusuna –hayır cevabı verilecektir. Yine tabaktaki elmaların arasında bir tane armut olduğunu varsayalım. O zaman aynı sorulara kesin küme teorisi açısından nasıl cevap verileceği açık değildir. Tabaktakiler elma mı? –sorusuna “belki tam değil”, “çoğunluğu elmadır”, “bir tanesi armut, diğerleri elmadır” gibi cevaplar alınabilir.

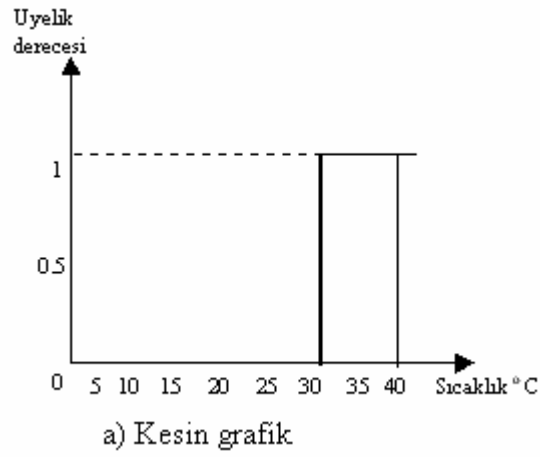
Yine tabakta yarısı elma yarısı armut olduğunu varsayalım. Bu durumda aynı sorulara “bir kısmı” , “yarısı” vs. gibi cevaplar verilebilir. Kesin küme teorisi açısından bu cevaplar mümkün değildir. Çünkü bu teoride ya “evet” (hepsi elmadır), ya da “hayır” (hiçbirisi elma değildir), cevapları mümkündür.

İşte bulanık küme bu noktada işe yarıyor. Bu teoride nesneler bir kümeye kısmen ait olabilirler. Bu aitlik **üyelik derecesi** (membership degree) ile belirlenir. Bulanık kümelerde üyelik derecesi karakteristik fonksiyonun genelleştirilmesi ile ölçülür ve üyelik fonksiyonu (membership function) olarak adlandırılır. Burada $\{0,1\}$ kümesi yerine $[0,1]$ arası kullanılır ve bu durumda üyelik fonksiyonu böyle belirlenir.

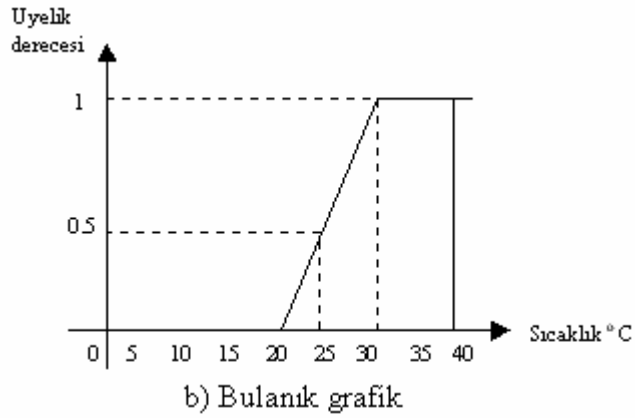
$$\mu_A(x): x \rightarrow [0,1] \text{ yani } 0 \leq \mu_A(x) \leq 1$$

Burada $\mu_A(x) = 0$ olması x 'in A 'ya ait olmadığını (A 'nın elemanı olmadığını), $\mu_A(x) = 1$ olduğunda ise x 'in A 'nın tam üyesi olduğunu göstermektedir. $\mu_A(x) = 0.5$ değeri bulanık A kümesinin **geçiş noktası** (crossover point) adlanır.

Böylece, klasik küme teorisinde, herhangi bir nesne bir kümeye ya aittir, yada ait değildir. Bulanık kümelerde ise elemanlar bu kümelere kısmen ait olabilmektedirler. Örneğin kesin küme teorisine göre 32°C'de hava sıcak, 31.5°C'de sıcak sayılmamaktadır (Şekil 2.1). Bulanık küme teorisinde ise 32°C sıcaklık, sıcaklık kümesinde maksimum üyelik derecesine, sahiptir (Şekil 2.2). 25°C klasik küme kavramına göre sıcak sayılmıyor.



Şekil 2.1 Sıcaklık için kesin küme kavramı



Şekil 2.2 Sıcaklık için bulanık küme kavramı

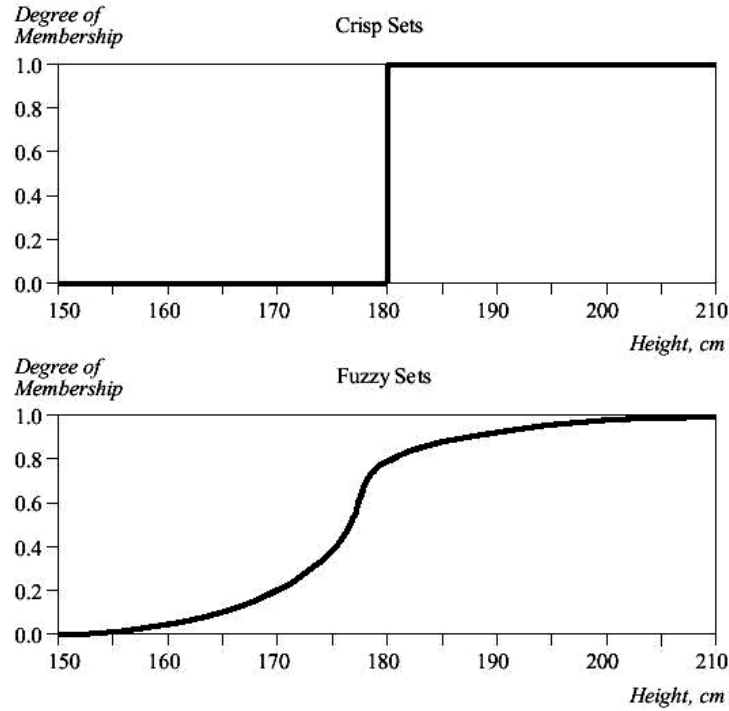
Fakat bulanık küme kavramına göre bu değerin sıcaklık kümesine üyeliği 0,5'dir (Şekil 2.2), yani 25°C tam sıcak değil ama soğuk da değildir.

Göründüğü gibi bulanık kümelerde kümenin bir elemanı bu kümeye kısmen ait olabilmektedir. Bu durum dünyayı daha gerçekçi olarak ifade etmektedir, çünkü gerçek dünya yalnızca evet veya hayır, beyaz veya siyah, doğru veya yanlış, açık

veya kapalı vs. gibi kavramlardan oluşmamakta dolayısıyla kavramların daha çok çeşit derecelerini içermektedir.

Örnek: Uzun ve kısa kavramlarını ele alalım. Kesin mantıkta 1,80 m ve üstü boyu olan bir insana uzun, 1,60 m ve altı boy uzunluğu olan birisine ise kısa denilmektedir. Bu durumda 1,20 m ve 2,20 m gibi boy uzunlukları için de aynı kısa ve uzun kavramlarını kullanacağız. Oysa bir insan bu boy uzunlukları için uygun olarak çok çok kısa ve çok çok uzun demektedir. Kesin mantıkta aynı zamanda 1,60 m ve 1,80 m boy uzunlukları arasında olan bir insan için kısa veya uzun boy açısından ne denileceği de açık değildir. Bu yüzden aslında boy kesin kümesi bir tane sınır değeri, örneğin, 1,70 m kabul etmekte ve değerlendirmeleri ona göre yapmaktadır.

Şekil 2.3’de klasik ve bulanık küme örnekleri görülmektedir;



Şekil 2.3 Boy uzunluklarının kesin ve bulanık kümeler olarak ifade edilmesi

Aşağıdaki ifadeler bulanık kavramlar içermektedirler:

Hava **çok** sıcaktır.

Utku **çok** uzun boyludur.

Saçlarım **biraz** uzun.

2.3 Bulanık Kümeler Üzerinde İşlemler

Genelde bir kesin (geleneksel) kümenin, $\{0,1\}$ üyelik fonksiyonlu bir bulanık kümenin özel hali olduğu söylenebilir. Bulanık küme teorisi kesin kümelerden daha çok uygulama bulmaktadır, çünkü bu teori insanın subjektif bir fikrini ifade etmek için daha uygundur. Bu yüzden, bu teorenin hayata, bilime, tekniğe ve onlarca diğer alanlara uygulanmasının nasıl yapıldığının öğrenilmesi ve incelenmesi gerekmektedir.

Bulanık kümeler üzerinde işlemler, basit tanımlamalar yardımı ile kesin kümeler üzerinde yapılan işlemlere benzer şekilde yapılır. Bu işlemleri örneklerle açıklayalım.

Örnek: $X=[0,+40]$ aralığında soğuk, serin ve sıcak olarak adlandırılabilen bulanık sıcaklık kümelerini ele alalım.

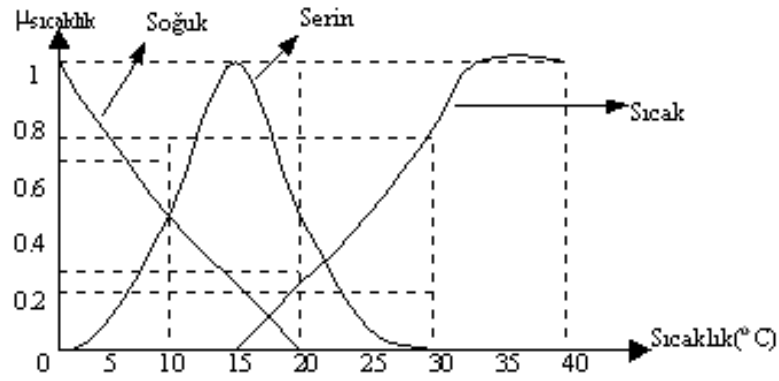
Sıcaklık 0°C ile 40°C

Soğuk = $\{1/0, 0.8/5, 0.5/10, 0.2/15, 0/20\}$

Serin = $\{0/0, 0.1/5, 0.5/10, 1/15, 0.5/20, 0.1/25, 0/30\}$

Sıcak = $\{0/15, 0.2/20, 0.5/25, 0.8/30, 1/35, 1/40\}$

Bu kümelerin grafiksel görünümü Şekil 2.4’de gösterilmektedir.



Şekil 2.4 Soğuk, Serin ve Sıcak bulanık kümelerinin grafiksel görünümü

2.4 Bulanık Kurallar

2.4.1 Dilsel Değişkenler

Bulanık kümelerin yapay zeka alanında önemli uygulamalarından biri dilsel hesaplamalardır. Burada amaç kesin rakamların yerine tabii dildeki ifadeleri kullanarak mantıkta yapıldığı gibi hesaplamaların yapılabilmesidir. “Bu gün hava çok sıcaktır” cümlesinde “Bu günün hava sıcaklığı” bir değişken ve “çok” onun değeri olarak ele alınabilir. “Hava sıcaklığı” değişkeni rakam olarak 25⁰C, 30⁰C vs. değerler aldığında bu değişkeni matematiksel olarak işlemek için iyi yöntemler mevcuttur. Fakat değişkenin değerini rakam olarak değil de kelime (“çok”) olarak ele aldığımızda bu değişkeni işlemek için klasik matematiksel bir teori mevcut değildir. Böyle bir yöntemi sağlamak için dilsel değişken kavramından kullanılır. Kabaca, eğer değişkenin değeri olarak tabii dilde kullanılan kelimeler ele alınırsa bu

değişkene **dilsel değişken** denmektedir. Aşağıdaki Tablo 2.2’de bazı dilsel değişkenler ve bunların tipik değerleri gösterilmektedir.

Tablo 2.2 Bazı Dilsel Değişkenler ve Bunların Tipik Değerleri

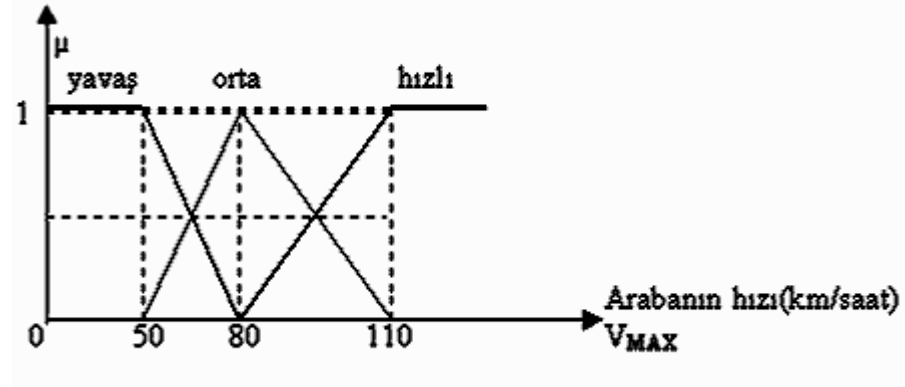
Dilsel Değişkenler	Tipik Değerler
Sıcaklık	Az, Normal, Çok
Sayı	Birkaç, Az, Çok
Yaş	Bebek, Genç, Yaşlı
Renk	Beyaz, Kara, Kırmızı, Mavi, Yeşil
Hız	Yavaş, Orta, Hızlı

Örnek: Eğer çok sıcak ise, o halde biraz soğuk ekle.

Eğer elma kırmızı ise, o halde o yetişmiştir.

Dilsel değişkenlerin değerini ifade edebilmek için çoğu zamanlar birden fazla kelime kullanılabilir: hızlı, hızlı değil, çok hızlı ve orta hızlı gibi. Genelde dilsel değişkenin değerini $x = x_1, x_2, \dots, x_n$ terimi ile ifade edersek, bu değer x_1, x_2, \dots, x_n atomik terimlerinin konketanasyonu (concatenation) ile elde edilir. Atomik terimler üç grup olarak sınıflandırılırlar:

İlk (başlangıç - primary) terimler. Bu terimler bulanık kümelerin seviyeleridir. Şekil 2.5’de bir arabanın hızını dilsel değişken olarak ele alınması ve onun “yavaş”, “orta”, ve “hızlı” değerleri gösterilmektedir.



Şekil 2.5 Bir arabanın hızı dilsel değişkeninin “yavaş”, “orta”, ve “hızlı” bulanık kümelerden değer alması.

“Değil” tümleyeni ve “VE” ve “VEYA” ilişkileri içeren terimler.

Çitler (Hedges-engeller) – “çok”, “az” ,”azçok” vs gibi değerleri içeren terimler

Şekil 2.5’de görüldüğü gibi dilsel değişken olan “arabanın hızı” değişkeni “yavaş”, “orta” ve “hızlı” değerleri almaktadır. Ama bu değerlerden başka “arabanın hızı” “çok yavaş”, “azçok yavaş”, “çok hızlı”, “az hızlı” vs gibi değerlerde alabilir. Bu değerleri ifade edebilmek için kullanılan terimler çitler denmektedir.

Eğer F bir bulanık değer ise:

Tablo 2.3 F bulanık değerinin karşılıkları

Çok F	$CON(F) = F^2$
Çok Çok F	F^4
Azçok F	$DIL(F) = F^{0.5}$
Artı F (Plus F)	$F^{1.25}$
F Değil	$1-F$
Çok F Değil	$1-CON(F)$
Az F	$F^{0.25}$

Dilsel deęişken bulanık küme veya kurallarla ifade edilebilen sentaks ve semantik deęerlere sahip olmalıdırlar. Sentaks kural iyi biçimlenmiş ifadeleri $T(L)$ şeklinde gösterebilir.

Örneęin: $T(\text{Sıcaklık}) = \{\text{Sıcak}, \text{Çok sıcak}, \text{Çok çok sıcak}, \dots\}$ ise bu terim-küme rekürsiv olarak türetilmektedir.

$$T^{i+1} = \{\text{Sıcak}\} \cap \{\text{Çok } T^i\}$$

Örneęin:

$$T^0 = \emptyset \text{ (boş küme)}$$

$$T^1 = \{\text{Sıcak}\}$$

$$T^2 = \{\text{Sıcak}, \text{Çok sıcak}\}$$

$$T^3 = \{\text{Sıcak}, \text{Çok sıcak}, \text{Çok çok sıcak}\}$$

2.4.2 Bulanık Eęer - O Halde Kuralları ve İmplikasyonları

Bulanık sistemlerde ve bulanık kontrol sistemlerinde insan bilgisi büyük çoğunlukla EęER-O HALDE (IF-THEN) bulanık kuralları ile sunulmaktadır. Bulanık EęER-O HALDE kuralı

EęER < bulanık söylem >, O HALDE < bulanık söylem > şeklinde koşullu cümledir (söylem-proposition).

Bulanık söylem iki tür olmaktadır: atomik ve bileşik (compound).

Atomik bulanık söylem;

“ x A dır (x ise A) ”

Şeklin de olmaktadır. Burada x dilsel deęişken, A ise dilsel deęerdir. A x 'in fiziksel alanında belirlenmiş bir bulanık kümedir.

Bileşik bulanık söylem;

“VE”, “VEYA” ve “DEĞİL” ilişkilerini kullanan atomik bulanık kompozisyonudur ve bu ilişkiler uygun olarak bulanık kesişme, birleşme ve tümlemeyi ifade etmektedirler.

Örneğin eğer x arabanın hızı ise o zaman aşağıdakilerin t bulanık söylem olduğu söylenebilir. (Burada S -yavaş (slow), M -orta (middle), ve F -hızlı (fast) bulanık kümeleri göstermektedir):

x S' tir.

x M' dir.

x F' tir.

x S' tir veya x M değildir

x S değildir ve x F değildir

(x S tir ve x F değildir) veya x M' dir.

Burada son üç söylem bileşik bulanık iddialardır. Bir bileşik söylemin içeriğindeki atomik iddialar özerktirler ve aynı bir söylemdeki x çeşitli değerler alabilmektedir. Yani bileşik söylemdeki dilsel değişkenler genelde aynı olmayabilirler. Örneğin x arabasının hızı ve $y = x'$ arabasının ivmesi (acceleration) ise ve eğer biz ivme için (L) bulanık kümesini belirlemek istersek, o halde aşağıdaki bulanık söylemi elde ederiz.

x F' tir ve y L' dir.

Buradan bileşik bulanık söyleme bir bulanık bağıntı gibi bakılabildiği görülmektedir. Böyle bir bulanık bağıntının üyelik fonksiyonlarını belirlemek için neler yapılması gerektiğine bakalım.

“VE” ilişkisi için bulanık kesişme kullanılmaktadır. x ve y 'nin U ve V 'nin fiziksel alanlarında dilsel değerler olduğunu ve A ve B 'nin uygun olarak U ve V 'de bulanık kümeler olduğunu varsayarsak o zaman

x A' dır ve y B' dir

Bileşik bulanık söylemi $U \times V$ de

$$\mu_{A \cap B}(x, y) = t [\mu_A(x), \mu_B(y)]$$

üyelik fonksiyonlu $A \cap B^{-I}$ bulanık bağıntısı olarak yorumlanabilir. t -norm olarak min kullanırsak;

$$\mu_{A \cap B}(x, y) = \min[\mu_A(x), \mu_B(y)]$$

“VEYA” ilişkisi için bulanık birleşme kullanılmaktadır. Yani

x A 'dır veya y B 'dir

bileşik bulanık söylemi $U \times V$ 'de

$$\mu_{A \cup B}(x,y) = s[\mu_A(x), \mu_B(y)]$$

olarak ele alınır. Burada $s:[0,1] \times [0,1] \rightarrow [0,1]$ herhangi bir s -normudur.

Örneğin s -norm olarak max kullanırsak

$$\mu_{A \cup B}(x,y) = \max[\mu_A(x), \mu_B(y)]$$

“DEĞİL” ilişkisi için bulanık tümlleme kullanılmaktadır.

Aşağıda bulanık implikasyonları ve bu implikasyonlara ait açıklamalar gösterilmektedir.

- **Dienes – Rescher implikasyonu:**

$$\mu_{Q_D}(x,y) = \max[1 - \mu_{FP_1}(x), \mu_{FP_2}(y)]$$

- **Lukasiewicz implikasyonu:**

$$\mu_{Q_L}(x,y) = \min[1, 1 - \mu_{FP_1}(x) + \mu_{FP_2}(y)]$$

- **Zadeh implikasyonu :**

$$\mu_{Q_Z}(x,y) = \max\{\min[\mu_{FP_1}(x), \mu_{FP_2}(y)], 1 - \mu_{FP_1}(x)\}$$

- **Gödel implikasyonu :**

$$\mu_{Q_G}(x,y) = \begin{cases} 1 & \text{eger } \mu_{FP_1}(x) \leq \mu_{FP_2}(y) \\ \mu_{FP_2}(y) & \text{diger durumlarda} \end{cases}$$

- **Mamdani implikasyonu :**

$$\mu_{Q_{MM}}(x,y) = \min[\mu_{FP_1}(x), \mu_{FP_2}(y)] \quad \text{veya}$$

$$\mu_{Q_{MP}}(x,y) = \mu_{FP_1}(x) \cdot \mu_{FP_2}(y)$$

- **Goguen implikasyonu :**

$$\mu_{Q_{Gog}}(x, y) = \begin{cases} 1 & \text{eger } \mu_{FR_1}(x) = 0 \\ \min(1, \mu_{FR_2}(y) / \mu_{FR_1}(x)) & \text{diger durumlarda} \end{cases}$$

- **Reichenbach implikasyonu:**

$$\mu_{Q_{Rei}}(x, y) = 1 - \mu_{FP_1}(x) + \mu_{FP_1}(x) \cdot \mu_{FP_2}(y)$$

- **Larsen implikasyonu :**

$$\mu_{Q_{Lar}}(x, y) = \mu_{FP_1}(x) \times \mu_{FP_2}(y)$$

- **Aliev – Tserkovniy 1 implikasyonu:**

$$\mu_{Q_{AR1}}(x, y) = \begin{cases} 1 - \mu_{FR_1}(x) & \text{eger } \mu_{FR_1}(x) < \mu_{FR_2}(y) \\ 1 & \text{eger } \mu_{FR_1}(x) = \mu_{FR_2}(y) \\ \mu_{FR_2}(y) & \text{eger } \mu_{FR_1}(x) > \mu_{FR_2}(y) \end{cases}$$

- **Aliev – Tserkovniy 2 implikasyonu:**

$$\mu_{Q_{AR2}}(x, y) = \begin{cases} 1, & \text{eger } \mu_{FR_1} \leq \mu_{FR_2}(y) \\ \min[1 - \mu_{FR_1}(x), \mu_{FR_2}(y)] & \text{eger } \mu_{FR_1} > \mu_{FR_2}(y) \end{cases}$$

- **Aliev – Tserkovniy 3 implikasyonu:**

$$\mu_{Q_{AR3}}(x, y) = \begin{cases} 1, & \text{eger } \mu_{FR_1}(x) \leq \alpha \mu_{FR_2}(y) \\ \frac{\mu_{FR_2}(y)}{\mu_{FR_1}(x) + (1 - \mu_{FR_2}(y))}, & \text{eger } \mu_{FR_1}(x) > \mu_{FR_2}(y) \end{cases}$$

2.5 Bulanık Sistemler

2.5.1 Bulanık Sistem Nedir?

Bulanık kümeler ve bulanık mantık teorisinin en etkin uygulama alanı kontrol sistemleridir. Geleneksel kontrol sistemleri bulanık teorisinin yardımıyla bulanık kontrol sistemlerine dönüştürülebilir ve böyle sistemlerin uygulanması birçok avantajlar elde etmeğe olanak verir.

Genelde, bulanık sistemler bilgiye dayalı veya kurala dayalı sistemlerdir. Yani bir bulanık sistemin temelinde “Eğer - O halde” kuralları durmaktadır. Örneğin, aşağıdaki kuralı bulanık sistemin bir kuralı hesap etmek yerinde olur.

Eğer sıcaklık soğuk ve basınç düşük ise, o halde sıcak su supabını orta pozitifte tut ve soğuk su supabının durumunu değişme.

Burada soğuk, düşük, orta pozitif gibi dilsel değerler kullanılır ve bu dilsel değerlerin uygun üyelik dereceleri mevcuttur.

Bir bulanık sistem tasarlanmasına karar verildikten sonra ilk yapılacak iş Eğer o-halde kurallar toplusunu elde etmektir. Bu kurallar çoğu zaman uzmandan yararlanılarak toplanılır.

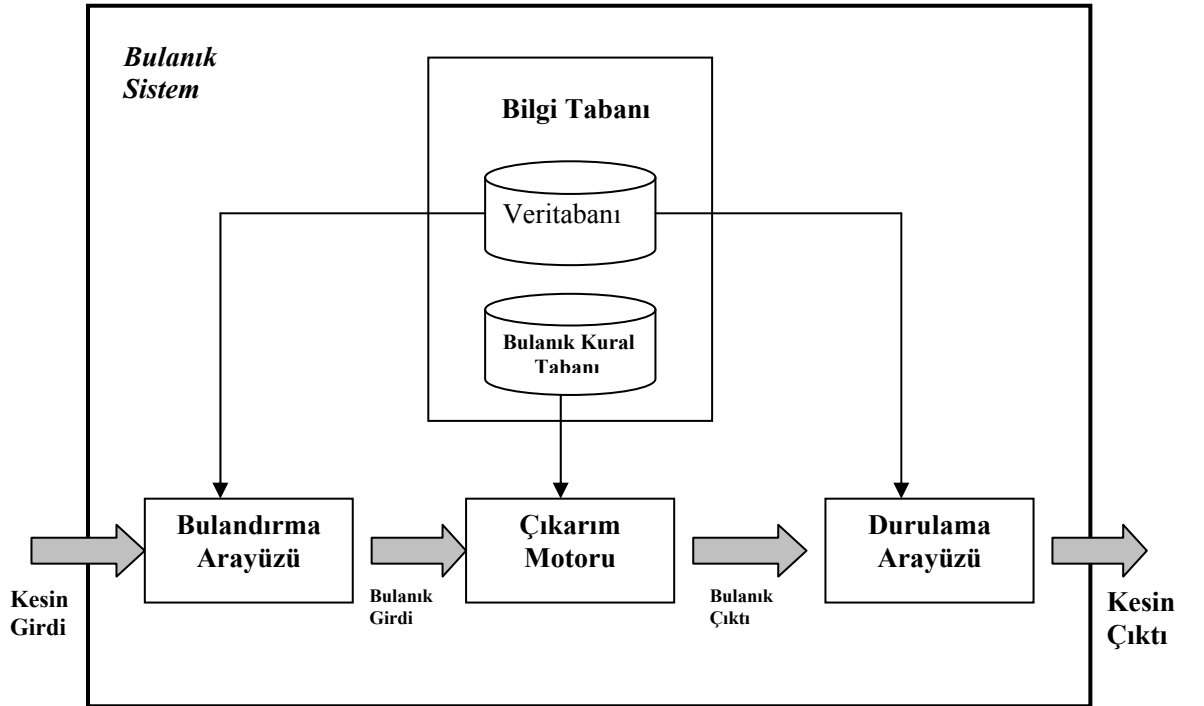
Literatürde genelde üç tip bulanık sistemden söz edilmektedir. 1) Şekil 2.6’ de temiz (pure) bulanık sistemler; 2) Şekil 2.7’ de bulanıklaştırıcı ve durulaştırıcı sistemler; 3) Şekil 2.8’ de Takaği – Sugeno – Kang (TSK) bulanık sistemlerdir [Wang, 1997; Allahverdi, 2006].

Temiz bulanık sistemlerde sistemin giriş ve çıkışları bulanıktır. Bulanık çıkarım mekanizması (Fuzzy Inference Engine) bulanık girişlere uygun kuralları bulanık kurallar tabanından (Fuzzy Rule Base) alarak imal edilir ve vardığı sonuçta bulanık olur (Şekil 2.6).



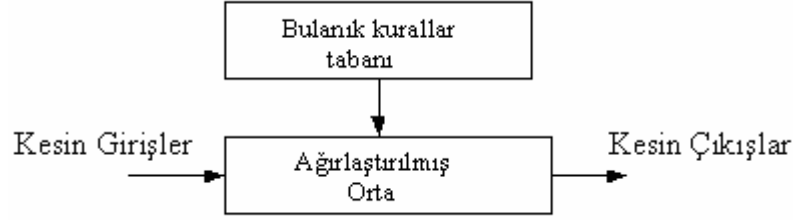
Şekil 2.6 Temiz bulanık sistemlerin temel şeması

Temiz bulanık sistemlerde giriş ve çıkış değerleri dilsel olarak kullanılmaktadır, hâlbuki gerçek sistemlerde bu değerler kesindir, bulanık değildir. Bu dezavantajı kaldırmak için sisteme girişteki kesin değerleri bulanık değerlere dönüştüren **bulanıklaştırıcı** (fuzzitier) ve çıkıştaki bulanık değerleri kesin değerlere dönüştüren **durulaştırıcıyı** (defuzzitier) uygun olarak sistemin girişine ve çıkışına ilave edilirler. Böyle sistem bulanıklaştırıcılı ve durulaştırıcılı sistem adlanmaktadır (Şekil 2.7).



Şekil 2.7 Bulanıklaştırıcılı ve durulaştırıcılı sistemler

Bulanık sistemlerin TSK modelinde sistemin giriş ve çıkış kesin değerlerdir. Bu modelde bulanık çıkarım mekanizması yerine **ağırlaştırılmış orta** (weighted average) kullanılır (Şekil 2.8). Burada ağırlaştırılmış orta Eğer – O halde kuralının o halde kısmında çoğu zaman bir matematik formül kullanılır.



Şekil 2.8 Bulanık sistemin TSK modeli

Böyle bir kullanım ise bulanık mantığın çeşitli prensiplerini uygulamaya imkân vermemektedir. Bu yüzden TSK modelinin uygulama alanı kısıtlıdır.

Bulanık teori, bulanık sistemlerde, özellikle de otomatik kontrol sistemlerinde insan bilgisine dayanan dilsel bir kontrol strateji uygulamak için kullanılır. Bulanık kontrol sistemleri tasarlarken sırasıyla hedef, bilgi tabanını oluşturan bulanık kontrol kuralları belirlenir ve bulanıklaştırma ve durulaştırma yapılır.

2.5.2 Bulanık Çıkarım Mekanizması

Bulanık çıkarım mekanizması bulanık kurallar tabanında öylesine bir bulanık Eğer - O halde kuralları oluşturuyor ki, bu kurallar U 'dan olan A' bulanık kümesini V 'den olan B' bulanık kümesine dönüştürür. Daha önce biz bulanık Eğer - O halde kuralının giriş - çıkış çarpımı $U \times V$ uzayında bir bulanık bağıntı gibi yorumlandığından konuşmuştuk. Eğer bulanık kurallar tabanı yalnız bir tane

kuraldan oluşuyor ise, O halde genelleştirilmiş Modus Ponens A' den B' 'ye doğru geçişi açıklamaktadır. Fakat kuralların sayısı pratikte birden çok olduğu için bu kurallardan nasıl çıkarım elde etmenin iyi yolu var: (1) Kompozisyona dayalı çıkarım; (2) Kişisel kurala dayalı çıkarım.

2.5.2.1 Kompozisyona Dayalı Çıkarım

Bu yöntemde bulanık kurallar tabanındaki tüm kurallar $U \times V$ 'de tek (single) bulanık bağıntı şeklinde düzenlenir. Sonradan bu bağıntı tek bir bulanık Eğer - O halde kuralı olarak ele alınır. Kurallar kümesinin ne anlama geldiğini kestirmekle yaklaşık lojik operatörler kullanılır.

Kurallar kümesinin hangi anlam taşıdığının anlaşılması için iki birbirine zıt görüş mevcuttur. Birinci görüşe göre kurallar özerk koşullu ifadeler olarak ele alınır. Bu durumda kuralları düzenlemek için uygun operatör birleşmedir (union). İkinci görüşe göre kurallar öylesine bir güçlü koşullu çift ifadelerdir ki, bu kuralların koşulları tüm kurallar kümesinin bir çarpışmasını (impact) sağlıyor. Bu durumda kesişme (üntersection) operatörü kullanılır. Bazı durumlarda (örneğin, Gödel amplifikasyonunda) bu görüş kullanılmaktadır.

(1.1) ifadesini kullanarak bu çıkarım şemalarına daha detaylı bakalım.

Ku' $U \times V$ 'de bir bulanık bağıntı oluşturduğunu varsayalım yani

$$K_M^i = A_1^i x_1 \dots x_n A_n^i \rightarrow B^i \quad (1.1)$$

Bu amplifikasyon için

$$\mu_{A_1^i \dots A_n^i}(x_1, \dots, x_n) = \mu_{A_1^i}(x_1) * \dots * \mu_{A_n^i}(x_n) \quad (1.2)$$

ve $*$ herhangi t -norm operatörüdür. İmplikasyon (\rightarrow) çeşitli yollarla belirlenebilmektedir. Eğer kurallar kümesine yukarıda bakılan birinci görüşü uygularsak (1.1) deki M tane kural $U \times V$ de birtek Q_M bulanık bağıntı şeklinde ele alınabilir.

$$Q_M = \bigcup_{i=1}^M K_{U^{(i)}} \quad (1.3)$$

Bu ifade **Mamdani kombinasyonu** adlanır. (1.3) ifadesi S-norm operatörü ile böyle yazılabilir.

$$\mu_{Q_M}(x, y) = \mu_{K_{U^{(1)}}}(x, y) \dot{+} \dots \dot{+} \mu_{K_{U^{(M)}}}(x, y)$$

Burada $\dot{+}$ S-norm operatörüdür.

Kurallar kümesine ikinci görüş uygulandığında UxV 'de bulanık bağıntı Q_G böyle yazılabilir:

$$Q_G = \bigcap_{i=1}^M K_{U^{(i)}} \quad (1.4)$$

veya eşdeğer olarak

$$\mu_{Q_G}(x, y) = \mu_{K_{U^{(1)}}}(x, y) * \dots * \mu_{K_{U^{(M)}}}(x, y)$$

Burada $*$ - t- norm operatörünü gösterir ve bu ifade **Gödel kombinasyonu** adlanır.

2.5.2.2 Kişisel Kurala Dayalı Çıkarım

Bu yöntemde bulanık kural tabanındaki her bir kural bir tane çıkış bulanık kümesi belirlemektedir ve tüm bulanık çıkarım mekanizması M tane kişisel bulanık kümelerin bir kombinasyonudur. Bu kombinasyon birleşme ve kesişme ile elde edilebilir.

Burada önce (1.1) ifadesi için (1.2) ifadesi kullanılarak üyelik derecesi bulunur. Sonra $A'_1 x \dots x A'_n$ 'e bir bulanık söylem (BS_1) gibi, B_l 'e ise diğer bir bulanık söylem (BS_2) gibi bakarak (Eğer $\langle BS_1 \rangle$, O halde $\langle BS_2 \rangle$) $\mu_{KU^{(i)}}(x_1, \dots, x_n) = \mu_{A'_1 x \dots x A'_n \rightarrow B_l}(x_1, \dots, x_n, y)$ belirlenir.

Burada Dienes-rescher, lukasiewicz, zadeh, vs. implikasyon hesaplama yöntemlerinden biri kullanılabilir.

Daha sonra her bir $K_{U^{(i)}}$ için U 'da verilmiş bir $A^{l'}$ 'ya uygun çıkış V 'de B_l bulanık kümesi hesaplanır. Bu hesaplama genelleştirilmiş modus ponense dayanır.

$$\mu_{B_i}(y) = \sup_{x \in U} t[\mu_{A_i}(x), \mu_{B_i}(x, y)] \quad (x = 1, 2, \dots, M)$$

Son adım olarak bulanık çıkarım mekanizmasının çıkışı $\{B'_1, \dots, B'_M\}$ bulanık kümesinin birleşmesi ile

$$\mu_B(y) = \mu_{B_1}(y) \dot{+} \dots \dot{+} \mu_{B_M}(y)$$

veya kesişmesi ile hesaplanır.

$$\mu_B(y) = \mu_{B_1}(y) * \dots * \mu_{B_M}(y)$$

Örneğin (1) çarpım çıkarım mekanizması kullanılırsa

$$\mu_B(y) = \max_{i=1}^M \left[\sup_{x \in U} \left(\mu_{A_i}(x) \prod_{i=1}^n \mu_{A_i}(x_i) \mu_{B_i}(y) \right) \right]$$

(2) minimum çıkarım mekanizması yardımıyla

$$\mu_B(y) = \max_{i=1}^M \left[\sup_{x \in U} \min(\mu_{A_i}(x), \mu_{A_i}(x_1), \dots, \mu_{A_i}(x_n), \mu_{B_i}(y)) \right]$$

(3) Zadeh çıkarım mekanizması yardımıyla

$$\mu_B(y) = \min_{i=1}^M \left\{ \sup_{x \in U} \min \left[\mu_{A_i}(x), \max(\min(\mu_{A_i}(x_1), \dots, \mu_{A_i}(x_n), \mu_{B_i}(y)), l - \min_{i=1}^n(x_i)) \right] \right\}$$

$A1$, $B1$ ve $C1$ sırasıyla $i=1,2$, $e \in E$, $f \in F$ ve $g \in G$ için E, F ve G 'de tanımlanmak üzere iki tane bulanık denetim kuralını düşünelim;

$R1$: Eğer $e A1$ ve $f B1$ ise $g C1$ dir.

$R2$: Eğer $e A2$ ve $f B2$ ise $g C2$ dir.

Girdiler genellikle alıcılarla ölçülür ve kesin değerdedir. Bazı olgularda girdi verisini bulanık kümelerle döndürmeye elverişli olabilirler. Bununla beraber genelde, kesin bir değer bir bulanık tekillik olarak işlenebilir.

1. Tekillik girdisi: Eğer girdi tekillik değeri olarak verilirse, ilk ve ikinci kuralların $a1$ ve $a2$ eşleşme dereceleri (ateşleme şiddetleri) $\mu a1(e_0)$ ve $\mu b1(f_0)$ kullanıcının girdiği veri (e_0 ve f_0) ile kural tabanındaki veri (Ai ve Bi) arasındaki kısmi eşleşme dereceleri olmak üzere;

$$a1 = \mu A1(e_0) \wedge \mu B1(f_0) \text{ ve } a2 = \mu A2(e_0) \wedge \mu B2(f_0) \text{ olarak açıklanabilir.}$$

2. Bulanık girdi: Eğer girdi A_2 ve B' bulanık kümeleri olarak verilirse, kuralların a_i eşleşme dereceleri, $i=1,2,\dots$ için ;

$$a_i = \text{EK}[\text{EB}(\mu_{A'}(e) \wedge \mu_{A_i}(e)), \text{EB}(\mu_{B'}(f) \wedge \mu_{B_i}(f))] \text{ olacaktır.}$$

Eşleşme derecelerinin kartezyen çarpımda EK işlemcisi ile elde edilmektedir. Bu bağıntılar, aşağıdaki dört çıkarım yönteminde anahtar rol oynamaktadır.

2.5.2.2.1 Mamdani Çıkarım Yöntemi

Bu yöntem EK işlemci R_c 'yi bulanık içerme olarak EB-EK işlemcisinin bileşke olarak kullanır. $i=1,2,\dots,n$ ($e \in E$, $f \in F$ ve $g \in G$ için);

R_i eğer $e \in A$, ve B , ise $g \in C$ 'dir.

Şeklin de bir kural tabanı verildiğini kabul edelim.

Öyleyse $R_i = A_i \text{ ve } B_i \rightarrow C_i$;

$$\mu_{R_i} = \mu(A_i \text{ ve } B_i) \rightarrow C_i(e, f, g)$$

ile tanımlanır;

1. Girdi veri tekil olduğu zaman; $e = e_0, f = f_0$;

$$\mu_{C_i}(g) = [\mu_{A_i}(e_0) \text{ ve } \mu_{B_i}(f_0)] \rightarrow \mu_{C_i}(g)$$

Mamdani yönetimi bulanık içerme (koşul önermesi) (\rightarrow) için EK (en küçükleştirme kümesi) işlemcisini kullanır. Tekillik girdisi için, C'_i , A_i ve B_i 'nin en küçük eşleme derecesi ile belirlenir. Girdiler bulanık tekillikler ise, yani $A' = e_0$, $B' = f_0$

ise, α_i eşleşme derecesi $\mu_{A_i}(e_0)$ ve $\mu_{B_i}(f_0)$ arasında en küçük değerdir önermesinden, $a_i = \mu_{A_i}(e_0) \wedge \mu_{B_i}(f_0)$ olduğundan $\mu_{C'_i}(g) = a_i \wedge \mu_{C_i}(g)$ olur.

C' toplam sonucu bireysel C'_i sonuçların toplamıdır. C' çıkarım sonucu bireysel sonuçlardan elde edilen C'_i sonucunu toplamıdır önermesinden biliyoruz ki, çıkartılan sonuç C' nin μ_C üyelik fonksiyonun bireysel denetim kurallarından türetilen toplu sonuç olarak verildiğini ve böylece iki kural $R1$ ve $R2$ varken,

$$\mu_{C'}(g) = \mu_{C'}^1 \vee \mu_{C'}^2$$

$$\mu_{C'}(g) = [a1 \wedge \mu_{C'}^1(g)] \vee [a2 \wedge \mu_{C'}^2(g)] \text{ olur.}$$

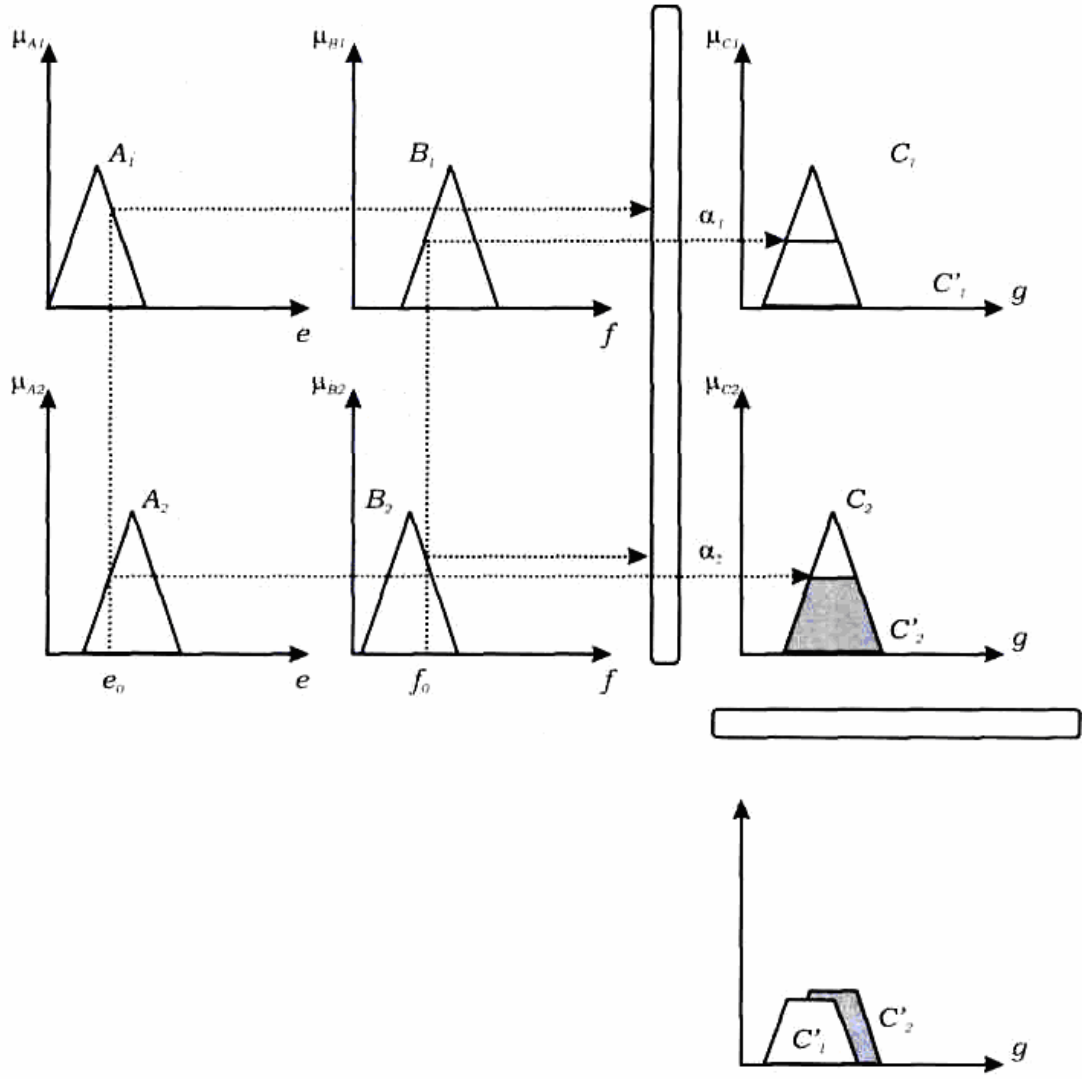
Girdiler tekillik olarak verildiğinde bu Mamdani bulanık çıkarım süreci, aşağıdaki şekildeki gibi temsil edilir.

Böylece genelde, C' toplam sonucu bireysel C'_i sonuçların toplamıdır. C' çıkarım sonucu bireysel sonuçlardan elde edilen C'_i sonucunu toplamıdır önermesinden biliyoruz ki;

$$\mu_{C'}(g) = \bigvee_{i=1}^n [\alpha_i \wedge \mu_{C_i}(g)] = \bigvee_{i=1}^n \mu_{C'_i}(g)$$

$$C' = \bigcup_{i=1}^n C'_i$$

elde edilmiş olur. Şekil 2.9.1'de Mamdani çıkarım yönteminin çıkarım mekanizmasının nasıl çalıştığının başlangıç şeklini göstermektedir.



Şekil 2.9.1 Mamdani çıkarım yöntemi başlangıç gösterimi

2. Girdi veri A' ve B' bulanık kümeleri olduğu zaman; bulanık girdi için, C'_i A' ve A_i ile B' ve B_i 'in en küçük eşleme derecesi ile belirlenir. Bulanık A' ve B' olarak girdiler verilirse, eşleşme derecesi α_i A' ve A_i ile B' ve B_i arasında en küçük aracılığıyla belirlenir önermesinden;

$$\alpha_i = \text{EK}_e[\text{EB}_e(\mu_{A'}(e) \wedge \mu_{A_i}(e)), \text{EB}_f(\mu_{B'}(f) \wedge \mu_{B_i}(f))]$$

$$\mu_{C'_i}(g) = \alpha_i \wedge \mu_{C_i}(g) \text{ dir.}$$

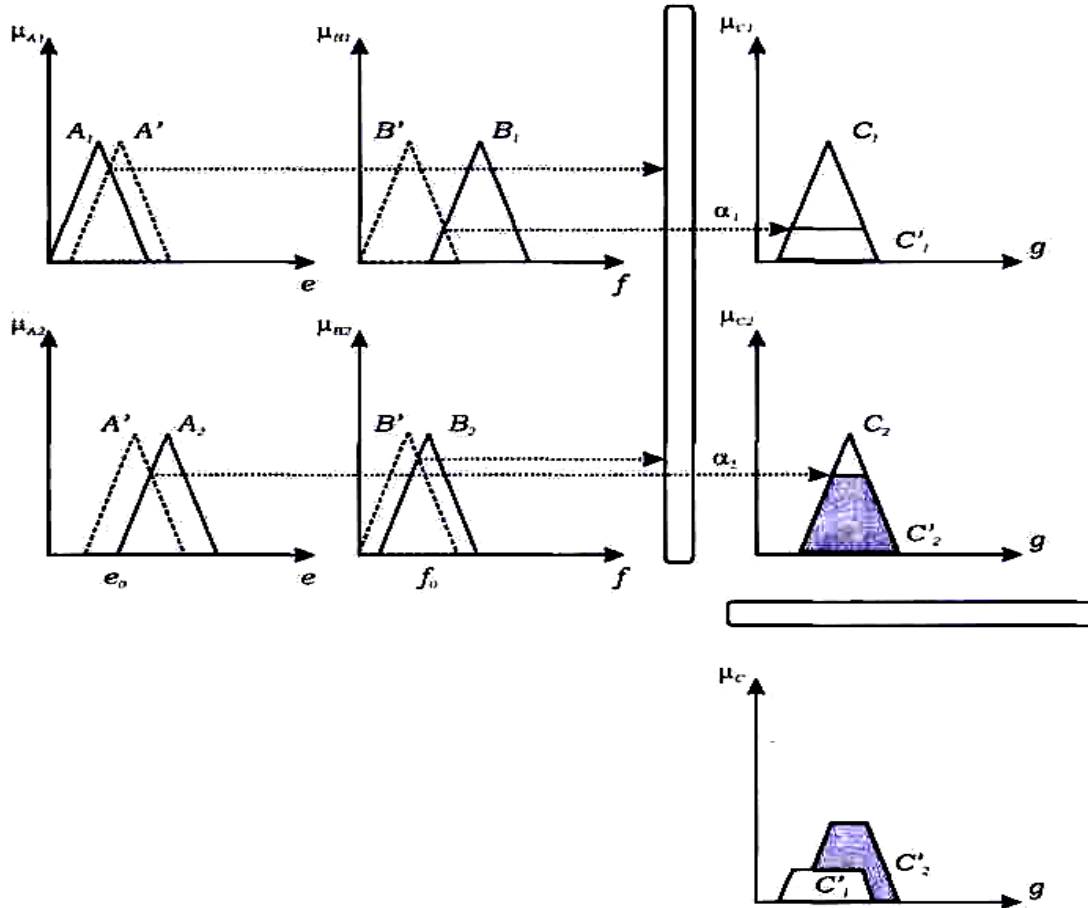
Olduğun da;

C' toplam sonucu bireysel C'_i sonuçların toplamıdır. C' çıkarım sonucu bireysel sonuçlardan elde edilen C'_i sonucunu toplamıdır önermesinden, toplu sonuçlara ulaşabiliriz;

$$\mu_{C'}(g) = \bigvee_{i=1}^n [\alpha_i \wedge \mu_{C_i}(g)] = \bigvee_{i=1}^n \mu_{C'_i}(g)$$

$$C' = \bigcup_{i=1}^n C'_i$$

elde edilmiş olur. Şekil 2.9.2 Mamdani çıkarım yöntemi sonuç gösteriminin şeklidir.



Şekil 2.9.2 Mamdani çıkarım yöntemi sonuç gösterimi

C sonucu, bulanık bir kümedir, bundan dolayı deterministik bir denetim elde etmek istersek, durulaştırma yöntemi kullanırız.

2.5.2.2.2 Larsen Çıkarım Yöntemi

Bu yöntem bulanık içerme için Rp çarpım işlemcisini, bileşke için de EB - çarpım işlemcisini kullanır. $i=1,2,\dots,n$ olmak üzere R_i : Eğer $e A_i$ ve $f B_i$ ise $g C_i$ dir.

Kural tabanı için;

$R_i = (A_i \text{ ve } B_i) \rightarrow C_i$; $\mu_{R_i} = \mu_{(A_i \text{ ve } B_i) \Rightarrow C_i}(e, f, g)$ ile tanımlanır.

1. Tekillik girdi verisi $e=e_0$, $f=f_0$ olarak verilince, Tekillik girdisi için, C'_i , A_i ve B_i 'nin en küçük eşleme derecesi ile belirlenir. Girdiler bulanık tekillikler ise, yani $A'=e_0$, $B'=f_0$ ise, α_i eşleşme derecesi $\mu_{A_i}(e_0)$ ve $\mu_{B_i}(f_0)$ arasında en küçük değerdir önermesinden;

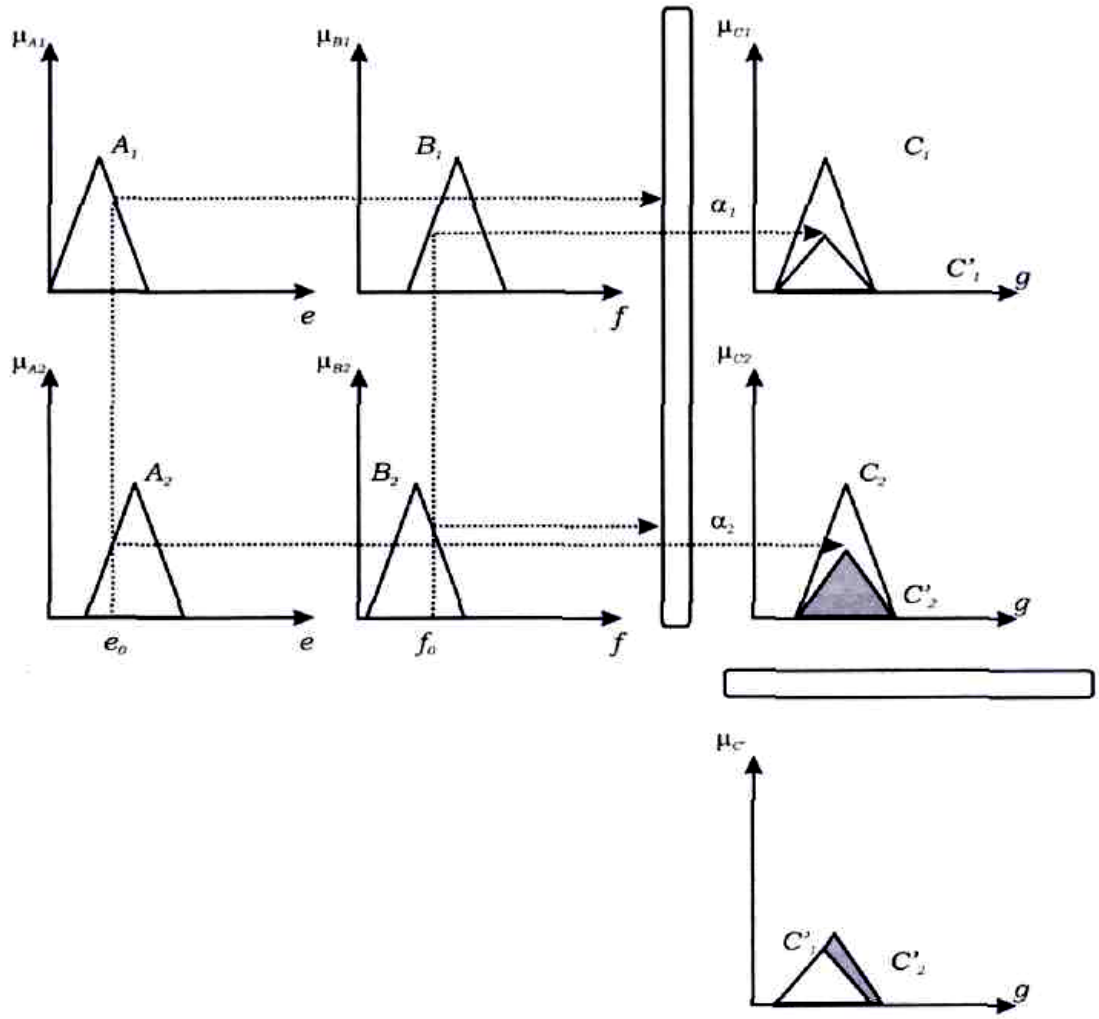
$$\mu_{C'_i}(g) = [\mu_{A_i}(e_0) \text{ ve } \mu_{B_i}(f_0)] \rightarrow \mu_{C_i}(g)$$

$$\mu_{C'_i}(g) = [\mu_{A_i}(e_0) \wedge \mu_{B_i}(f_0)] \bullet \mu_{C_i}(g)$$

$$\alpha_i = \mu_{A_i}(e_0) \wedge \mu_{B_i}(f_0) \text{ olduğunda } \mu_{C'_i}(g) = \alpha_i \bullet \mu_{C_i}(g)$$

sonucu karşımıza çıkmaktadır.

Şekil 2.10.1 Larsen çıkarım yönteminin çıkarım mekanizmasının nasıl çalıştığının başlangıç şeklini göstermektedir



Şekil 2.10.1 Larsen çıkarım yöntemi başlangıç gösterimi

C' toplam sonucu bireysel C'_i sonuçların toplamıdır. C' çıkarım sonucu bireysel sonuçlardan elde edilen C'_i sonucunu toplamıdır önermesinden;

$$\mu_{C'}(g) = \bigvee_{i=1}^n [\alpha_i \bullet \mu_{C_i}(g)] = \bigvee_{i=1}^n \mu_{C_i}(g)$$

$$C' = \bigcup_{i=1}^n C'_i$$

elde edilmiş olur.

2) Girdi verisi A' ve B' bulanık kümeleri olarak verilince, bulanık girdi için, $C'_i A'$ ve A_i ile B' ve B_i ' in en küçük eşleme derecesi ile belirlenir. Bulanık A' ve B' olarak girdiler verilirse, eşleşme derecesi α_i A' ve A_i ile B' ve B_i arasında en küçük aracılığıyla belirlenir önermesinden biliyoruz ki;

$$\alpha_i = \underset{e}{EK}[EB(\mu_{A'}(e) \wedge \mu_{A_i}(e)), \underset{f}{EB}(\mu_{B'}(f) \wedge \mu_{B_i}(f))] \text{ olduğundan;}$$

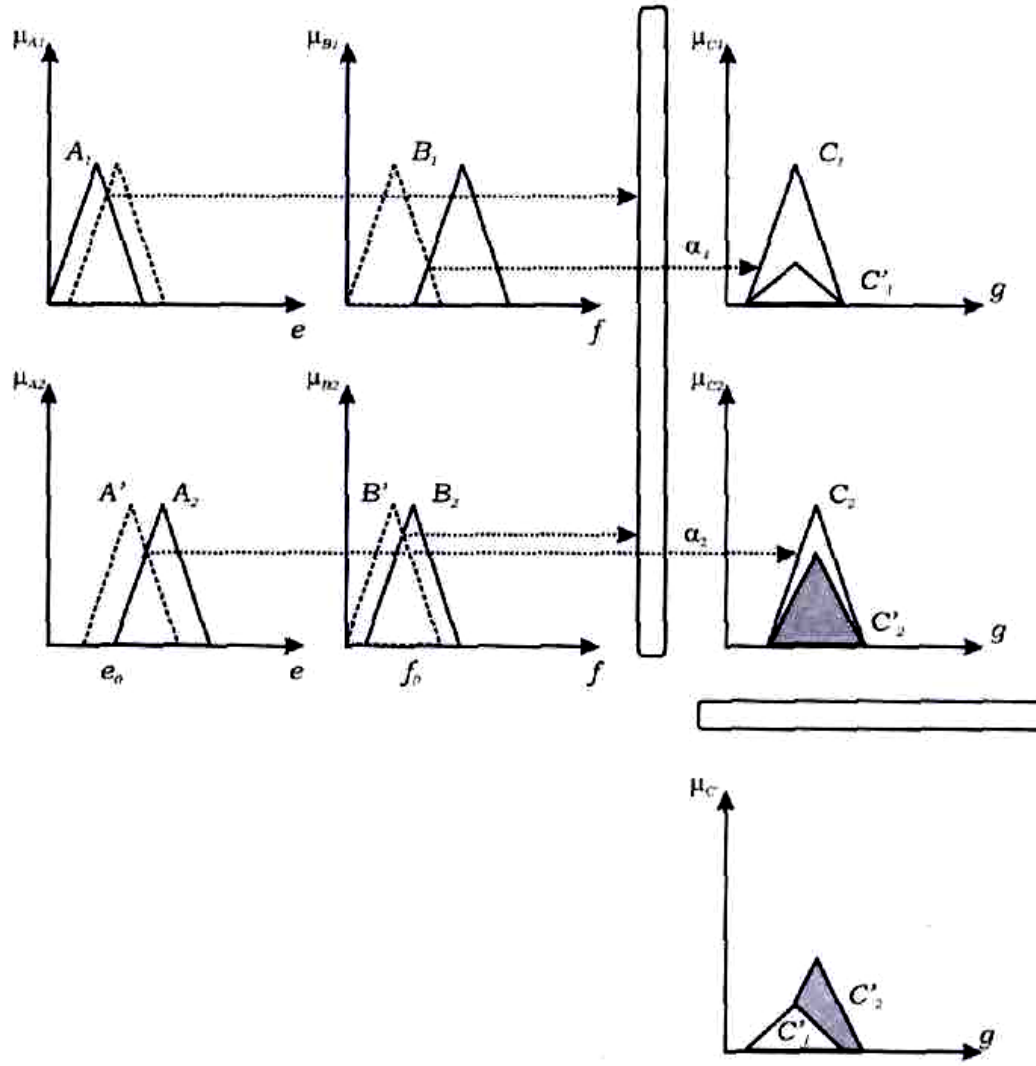
$$\mu_{C_i}(g) = \alpha_i \bullet \mu_{A_i}(g) \text{ olur.}$$

C' toplam sonucu bireysel C'_i sonuçların toplamıdır. C' çıkarım sonucu bireysel sonuçlardan elde edilen C'_i sonucunu toplamıdır önermesinden biliyoruz ki;

$$\mu_{C'}(g) = \bigvee_{i=1}^n [\alpha_i \bullet \mu_{C_i}(g)] = \bigvee_{i=1}^n \mu_{C_i}(g)$$

$$C' = \bigcup_{i=1}^n C'_i$$

elde edilmiş olur. Şekil 2.10.2 Larsen çıkarım yöntemi çıkarım mekanizması sonuç gösteriminin göstermektedir.



Şekil 2.10.2 Larsen çıkarım yöntemi sonuç gösterimi

2.5.2.2.3 Tsukamoto Çıkarım Yöntemi

Bu yöntemde her bulanık kümenin monotonik üyelik fonksiyonu olan bulanık küme tarafından temsil edilir. Kural tabanı şu şekildedir.

$\mu_{C_i}(g)$ monotonik bir fonksiyon iken,

R_i : Eğer $e A_i$ ve $f B_i$ ise $g C_i$ dir. $i=1,2,\dots,n$.

Sonuç olarak, her kuralın çıkarımı çıktı kuralın eşleşme derecesi (ateşleme şiddeti) kullanılarak kesin bir değer olarak tanımlanır. Toplam çıktı her kuralın çıktısının ağırlıklı ortalaması alınarak elde edilir.

C_i kümesinin monotonik üyelik fonksiyonu $Q(g)$ olduğu ve o 'nun, i 'inci kuralın eşleşme derecesi olduğunu kabul edelim.

1. Tekil girdi için (e_0, f_0) ; $\alpha_i = \mu_{A_i}(e_0) \wedge \mu_{B_i}(f_0)$

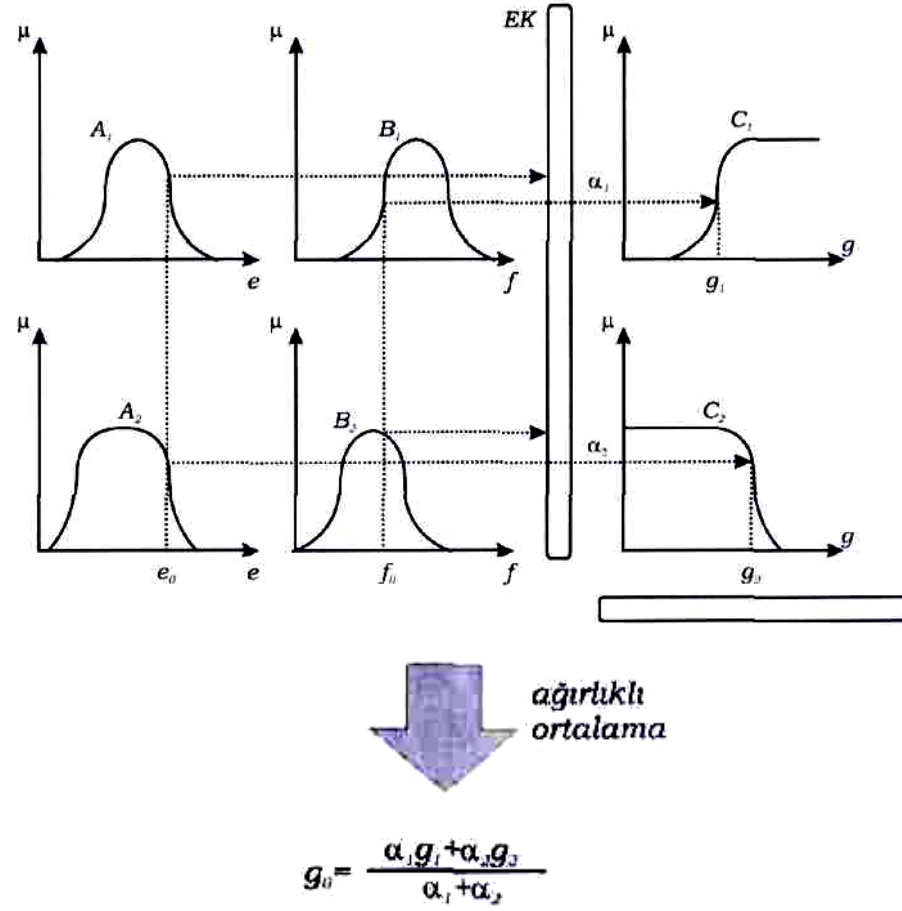
2. Bulanık küme girdisi için (A', B')

$$\alpha_i = \text{EK}_e [\text{EB}(\mu_{A'}(e) \wedge \mu_{A_i}(e)), \text{EB}(\mu_{B'}(f) \wedge \mu_{B_i}(f))]$$

olduğundan ;

Sonra i 'inci kuralın sonucu elde edilir: $g_i = \mu_{C_i}^{-1}(\alpha_i)$;

Şekil 2.11 Tsukamoto çıkarım yöntemi, ağırlıklı ortalama yönteminin şeklini ve formülünü göstermektedir.



Şekil 2.11 Tsukamoto çıkarım yöntemi, ağırlıklı ortalama yöntemi

İki kural olduğu zaman, sonuç ağırlıklı ortalamadan aşağıdaki şekilde türetilir;

$$g_0 = \frac{\alpha_1 g_1 + \alpha_2 g_2}{\alpha_1 + \alpha_2}$$

Her kural kesin bir sonuç çıkaracağından, Tsukamoto bulanık modeli ağırlıklı ortalama yöntemi ile her kuralın çıktısını toplar. Bundan dolayı zaman alıcı durulaştırmadan uzak durmayı sağlar.

2.5.2.2.4 Takagi-Sugeno-Kang (TSK) Yöntemi

Bu modelde tipik bir bulanık kural şu şekildedir;

Eğer $e \in A$ ve $f \in B$ ise $g=f(e,f)$ dir.

$g=f(e,f)$ sonucundaki kesin fonksiyon iken öncüldeki bulanık kümeler A ve B 'dir. Genellikle $f(e,f)$ e ve f girdi değerlerinde bir polinomdur ve bundan dolayı bu yöntem girdiler tekil değerler olarak verilince çalışır.

2.5.3 Bulanıklaştırıcı

(1) Gauss bulanıklaştırıcısı: Bu tür bulanıklaştırıcı $x^* \in U$ 'yu aşağıdaki gauss üyelik fonksiyonuna sahip olan bir $A' \in U$ bulanık kümesine gömülür.

$$\mu_{A'}(x) = \bar{e} \left(\frac{x_1 - x_{1^*}}{a_1} \right)^2 * \dots * \bar{e} \left(\frac{x_n - x_{n^*}}{a_n} \right)^2 \quad (1.6)$$

Bura da a_i pozitif bir parametre ve $*$ - çoğunlukla cebirsel çarpım veya min olarak kullanılan t -normdur.

(2) Üçgen (triangular) bulanıklaştırıcı:

Bu tür bulanıklaştırıcı $X' \in U$ 'yi üyelik fonksiyonu aşağıdaki gibi belirlenen $A' \in U$ bulanık kümesine gömülür.

$$\mu_{A'}(x) = \begin{cases} \left(1 - \frac{|x_1 - x_{1^*}|}{b_1} \right) * \dots * \left(1 - \frac{|x_n - x_{n^*}|}{b_n} \right) \\ 0, \text{ diğer durumlarda} \end{cases} \quad (1.7)$$

Burada bir pozitif parametre ve * - çoğunlukla cebirsel çarpım veya min olarak kullanılan t - normdur. Sonuç olarak,

Bir üçgen üyelik fonksiyonu a_1 , a_2 ve a_3 olarak üç parametre ile tanımlanır.

$$\mu_A(x; a_1, a_2, a_3) = \begin{cases} a_1 \leq x \leq a_2 \text{ ise } (x-a_1)/(a_2-a_1) \\ a_2 \leq x \leq a_3 \text{ ise } (a_3-x)/(a_3-a_2) \\ x > a_3 \text{ veya } x < a_1 \text{ ise } 0 \end{cases}$$

(3) Yamuk (trapezoid) bulanıklaştırıcı:

Bir yamuk üyelik fonksiyonu a_1 , a_2 , a_3 ve a_4 olarak dört parametre ile tanımlanır.

Aslında üçgen üyelik fonksiyonu yamuk üyelik fonksiyonunun özel bir durumudur.

$$\mu_A(x; a_1, a_2, a_3, a_4) = \begin{cases} a_1 \leq x \leq a_2 \text{ ise } (x-a_1)/(a_2-a_1) \\ a_2 \leq x \leq a_3 \text{ ise } 1 \\ a_3 \leq x \leq a_4 \text{ ise } (a_4-x)/(a_4-a_3) \\ x > a_4 \text{ veya } x < a_1 \text{ ise } 0 \end{cases}$$

Formüllerinin basit oluşu ve bilgi işlemsel etkinlikleri açısından hem üçgen hem de yamuk üyelik fonksiyonları çeşitli bulanık mantık uygulamalarında oldukça sık kullanılan fonksiyonlardır.

(1.6)-(1.7) ifadelerinin tümünün $\mu_A(x^*) = 1$ koşulunu sağlamakta olduğu, yani bulanıklaştırıcılar için öngörülen birinci kriteri sağlamakta olduğu görülmektedir. Singleton bulanıklaştırıcının bulanık çıkarım mekanizmasının

gerektirdiği hesaplamaların basitleştirdiğini göstermek mümkündür. Diğer taraftan eğer

$Ku^{(i)}$: EĞER $x_1 A_1^i$ ve...ve $x_n A_n^i$ ise O HALDE $y B^i$ 'dir. Şeklinde olan A_i^i bulanık kümesi gauss veya üçgen üyelik fonksiyonuna sahip ise o halde bu gauss ve üçgen bulanıklaştırıcı hem de girişteki görüldüğü bastırmakta olanak tanımakta, ama singleton bulanıklaştırıcı bunu yapmamaktadır.

2.5.4 Durulaştırıcı

Pek çok pratik uygulamada, denetim komutu kesin bir değer olarak verilir. Bundan dolayı, bulanık çıkarım sonucunu durulamak gerekir. Durulama, elde edilmiş bir bulanık denetim etkinliğinde olasılık dağılımını en iyi gösteren, bulanık olmayan denetim etkinliği elde etme sürecidir. Ancak iyi bir durulama stratejisi seçmek için sistematik bir işlem yoktur ve bundan dolayı uygulamanın özelliklerini dikkate alan bir yöntem seçilmesi gerekir.

Bulandırma ve durulama birbirlerinin bütünleyicisi gibi görünsede, ters fonksiyonlar değildir.

Bulanık değerlerin kesin değerlere dönüştürülmesi yani bulanık $B' \subset V$ kümesinden kesin bir $y' \in V$ noktasına gömülme süreci **durulaştırıcı** adlanır. Burada $V \subset R$ bulanık çıkarım mekanizmasının çıkışıdır. B' özel yollarla elde edildiğinden dolayı ona en iyi denk gelen noktayı seçmek için birkaç yöntem vardır. Durulaştırıcı için de aşağıdaki üç Kriter istenmektedir:

(1) Akla yakınlık (plausibility). y^* noktasının B' nü temsil etmesi sezgisel (intuitive) yolla belirlenir. Örneğin bu nokta yaklaşık olarak B' desteğinin ortasında yerleşebilir ve ya B' üyeliğinin en yüksek derecesini alabilir.

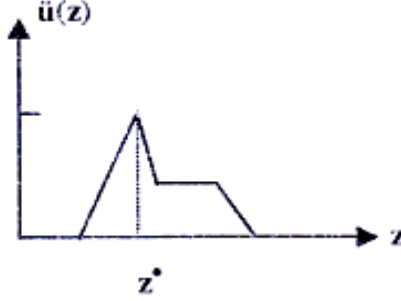
(2) Hesaplama basitliği. Bu kriter gerçek zaman ölçeğinde çalışan kontrol sistemleri için çok önemlidir.

(3) Devamlılık (continuity) B' 'deki küçük bir değişiklik y^* 'de büyük değişikliğe neden olmamalıdır.

2.5.4.1 Durulaştırma Yöntemleri

2.5.4.1.1 Ağırlık merkezci durulaştırıcı (Centroid yöntemi)

Bunun diğer bir adı da **alan merkezi** yöntemidir. Durulaştırma işlemlerinde, belki de en yaygın olarak kullanılan işlem budur. Şekil 2.12’de gösterilmiş olan bu durulaştırmanın matematik işlemi aşağıdaki denklem vasıtası ile yapılır.

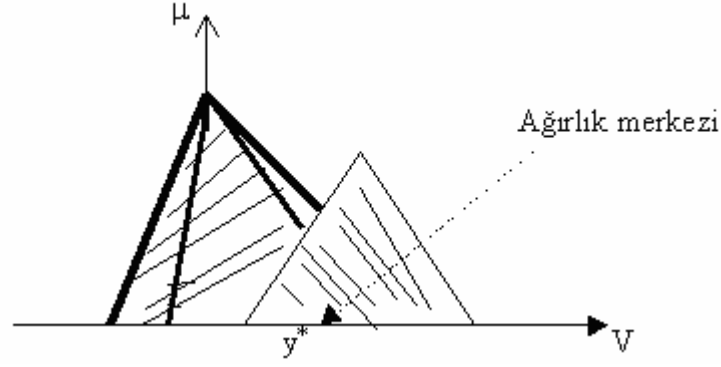
$$z^* = \frac{\int \ddot{u}_c(z) \cdot z dz}{\int \ddot{u}_c(z) dz}$$


Şekil 2.12 Centroid yöntemi ile durulaştırma

Bu tip durulaştırıcı y^* noktasını B' ’nin üyelik fonksiyonu ile örten alanın ortası olarak belirtir, yani biraz daha açarsak:

$$y^* = \frac{\int y \mu_B(y) dy}{\int \mu_B(y) dy}$$

burada; \int geleneksel integral sembolüdür. Şekil 2.13’de bu işlem grafiksel olarak gösterilmektedir.



Şekil 2.13 Ağırlık merkezli durulaştırıcının grafiksel sunumu

Bazen B 'deki üyelik değerleri çok küçük olan $y \in V$ 'leri atma gerekir. Bu indisli ağırlık merkezli durulaştırıcı yardımı ile yapılır.

$$Y^* = \frac{\int_{V_\alpha} y \mu_B(y) dy}{\int_{V_\alpha} \mu_B(y) dy}$$

$$V_\alpha = \{y \in V \mid \mu_B(y) \geq \alpha\}$$

α - bir sabittir.

Ağırlık merkezli durulaştırıcının avantajı onun sezgisel akla yakınlığındandır. Dezavantajı hesaplanmasının yoğun olmasıdır. Taktik olarak $\mu_B(y)$ üyelik fonksiyonu muntazam değildir (irregular) ve ona göre de yukarıda ki integralleri hesaplamak zordur.

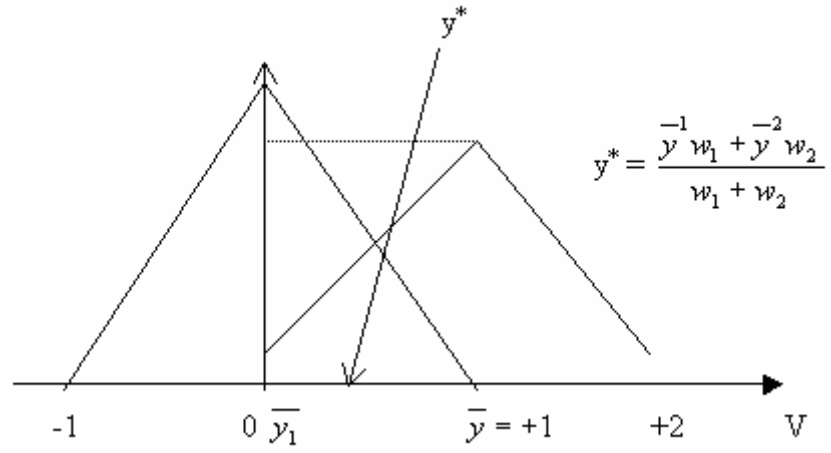
2.5.4.1.2 Ortamın Maksimumu Durulaştırıcı

Bu tür durulaştırıcı bazen merkezi orta da adlanır. B ' bulanık kümesinin M yani bulanık kümenin birleşmesi veya kesişmesi olduğundan dolayı ağırlık merkezli durulaştırıcı ifadesinin iyi aproksimasyonu (yaklaştırılması) M tane bulanık kümenin merkezlerinin ağırlıklı ortalamasıdır. Burada ağırlıklar uygun bulanık kümelerin yüksekliği olarak götürürler. (NOT: Bulanık kümenin yüksekliği (height)

kümedeki en maximum değer alan üyelik derecesidir). Özellikle y^{-1} 'in 1^{+} 'e bulanık küme ve \mathcal{W}^e 'inde onun yüksekliğini olduğunu varsayarsak, o zaman merkezi orta bulanıklaştırıcı y^* böyle belirlenir:

$$Y^* = \frac{\sum_{e=1}^M y^{-i} \mathcal{W}_e}{\sum_{e=1}^M \mathcal{W}_e} \quad (2.9)$$

Şekil 2.14'de $M=2$ için bu işlem grafiksel olarak görüntülenmektedir:



Şekil 2.14 Merkezi orta durulaştırıcının grafiksel sunumu

Bu yöntem en çok kullanılan yöntemlerden biridir. Avantajı hesaplanmanın kolay olması ve sezgisel olarak akla yakın olmasıdır. y^{-i} ve w_e 'deki küçük değişiklikler y^* 'de de küçük değişiklikler getirmesi bu yöntemin diğer avantajıdır.

2.5.4.1.3 Maksimum Durulařtırıcı

Bu yöntem V 'de $\mu_{B'}(y)$ maksimum değeri olan bir y^* noktasını seçmeğ e olanak tanır.

$$\text{Hgt}(B') = \{y \in V \mid \mu_{B'}(y) = \sup_{y \in V} \mu_{B'}(y)\}$$

olsun.

Burada $\text{Hgt}(B')$ - B' bulanık kümesinin yüksekliğ i olup V 'deki $\mu_{B'}(y)$ 'lerin maksimum değ er ald ığ ı tüm noktalar kümesidir. Maksimum durulařtırıcı y^* , $\text{hgt}(B')$ deki belirli bir eleman olarak ařağ ıdaki gibi hesaplanmaktadır:

$y^* = \text{hgt}(B')$ deki herhangi bir nokta.

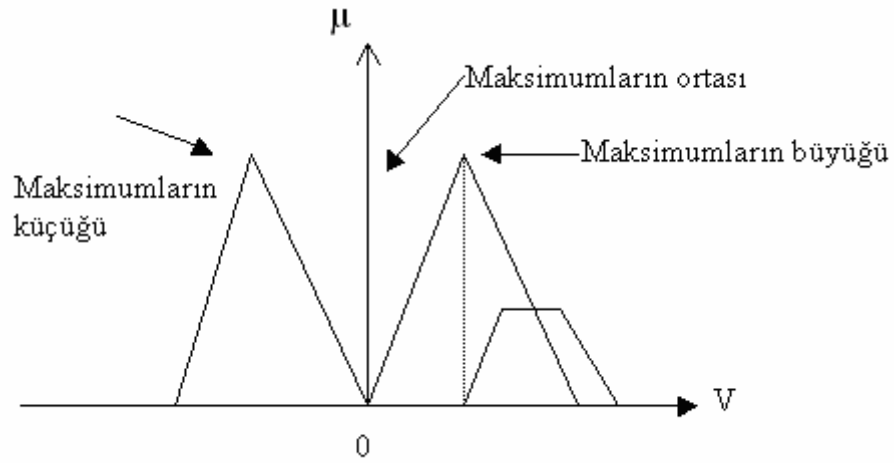
Eğ er $\text{hgt}(B')$ yalnız bir noktadan ibaret ise bu durumda y^* yegane bir yolla belirlenir. Eğ er $\text{hgt}(B')$ birden fazla noktaya sahip ise bu noktalarda herhangi biri veya maksimumlardan küç üğ ü büyüğ ü veya ortası g ö t ü r ü l e b i l i r : Ş e k i l 2.15'de bu durumlar grafiksel olarak g ö s t e r i l m e k t e d i r .

Maksimum Durulařtırıcı kendi arasında 3'e ayrılır. Bunlar;

Maksimumun küç üğ ü durulařtırıcı (som)

Maksimumun büyüğ ü durulařtırıcı (lom)

Maksimumun ortalaması durulařtırıcı (mom)

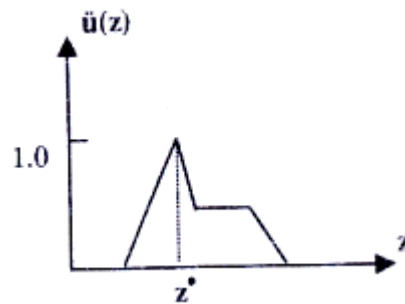


Şekil 2.15 Maksimum durulaştırıcının grafiksel sunumu

Maksimum durulaştırıcı sezgisel akla yakın ve hesaplama açısından basittir. Ama B' deki küçük değişiklikler y^* 'de büyük değişiklikler getirir.

2.5.4.1.4 En büyük üyelik ilkesi

Bunun diğer bir adı da **yükseklik** yöntemidir. Kullanılabilmesi için tepeleri olan çıkarım bulanık kümelerine gerek vardır. Şekil 2.16'de gösterilen bu durulaştırma işleminin aritmetik notasyon şeklinde gösterimi



Şekil 2.16 En büyük üyelik derecesi durulaştırması

2.5.4.1.5 Ağırlıklı ortalama yöntemi

Bunun kullanılabilmesi için simetrik üyelik fonksiyonunun bulunması gereklidir. İşlemler matematik olarak

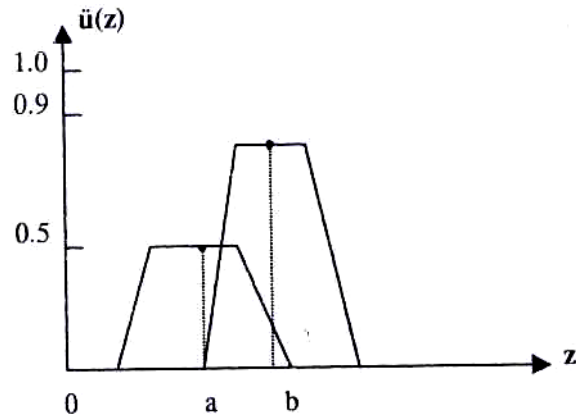
$$z^* = \frac{\sum \bar{u}_c(z) \cdot \bar{z}}{\sum \bar{u}_c(z)}$$

Şeklinde yapılır. Burada işareti cebir anlamında toplamayı gösterir. Bu durulaştırma işlemi Şekil 2.17' de gösterilmiştir. Böylece çıkışı oluşturan bulanık kümelerin üyelik fonksiyonlarının her biri sahip oldukları en büyük üyelik derecesi değeri ile çarpılarak ağırlıklı ortalamaları alınır.

Misal olarak Şekil 2.17'deki iki bulanık kümenin ağırlıklı ortalaması (durulaştırılmış değer)

$$z^* = \frac{a(0.6) + b(0.9)}{0.6 + 0.9} \text{ olarak bulunur.}$$

Bu durulaştırma işlemi sadece simetrik olan üyelik fonksiyonları için geçerli olduğundan, a ve b değerleri temsil ettikleri şekillerin ortalamalarıdır.



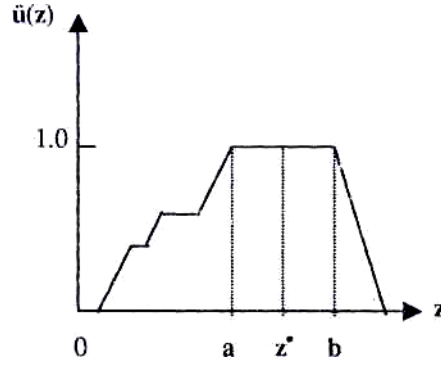
Şekil 2.17 Ağırlıklı ortalama yöntemi durulaştırması

2.5.4.1.6 Ortalama en büyük üyelik

Bu yöntem aynı zamanda en büyüklerin ortası diye de bilinir. Bu bakından birinci durulaştırma ilkesine, çok yakındır. Ancak, en büyük üyeliğin konumu tekil olmayabilir. Bunun anlamı üyelik fonksiyonunda en büyük üyelik derecesine sahip olan, $\mu A(z) = 1$, bir nokta yerine plato gibi düzlük kısmı da bulunabilir. Şekil 2.18’de durulaştırma işlemi gösterilmiş olan bu yönteme göre durulaştırılmış değer;

$$z^* = \frac{a+b}{2}$$

Olarak bulunur. Buradaki a ve b değerleri şekilde gösterilmiştir.



Şekil 2.18 Ortalama en büyük üyelik durulaştırılması

Diğer bir formül olarak, ayrık bir evren varlığında denetim etkinliği, Z_j üyelik fonksiyonunun en büyüğe ulaştığı denetim etkinliği, k bu şekildeki denetim etkinliklerinin sayısı olmak üzere;

$$z_0 = \sum_{j=1}^k \frac{z_j}{k}$$

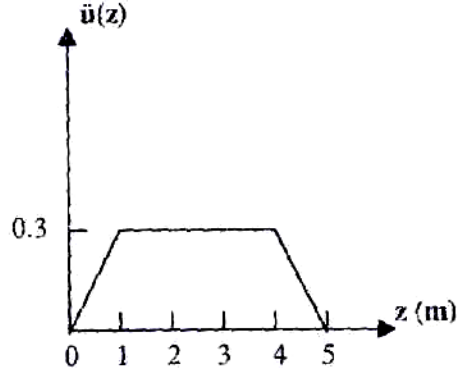
Olarak açıklanabilir.

2.5.4.1.7 Toplamların merkezi

Kullanılan durulaştırma işlemleri arasında en hızlı olanı bu yöntemdir. Bu yöntemde iki bulanık kümenin birleşimi yerine onların cebirsel toplamaları kullanılır. Bunun bir mahzuru örtüşen kısımların iki defa toplama girmesidir. Durulaştırılmış değer;

$$z^* = \frac{\int z \sum_{k=1}^n \tilde{u}_\zeta(z) dz}{\int \sum_{k=1}^n \tilde{u}_\zeta(z) dz}$$

olarak hesap edilebilir. Bir bakıma, bu hesaplama tarzı, ağırlıklı ortalama durulaştırmasına benzer. Ancak toplamların merkezi yönteminde ağırlıklar ilgili üyelik fonksiyonlarının alanlarıdır. Ortalama ağırlıklar yönteminde ise bu üyelik derecesidir. Toplamların merkezi ile durulaştırma işlemleri Şekil 2.19'da gösterilmiştir.



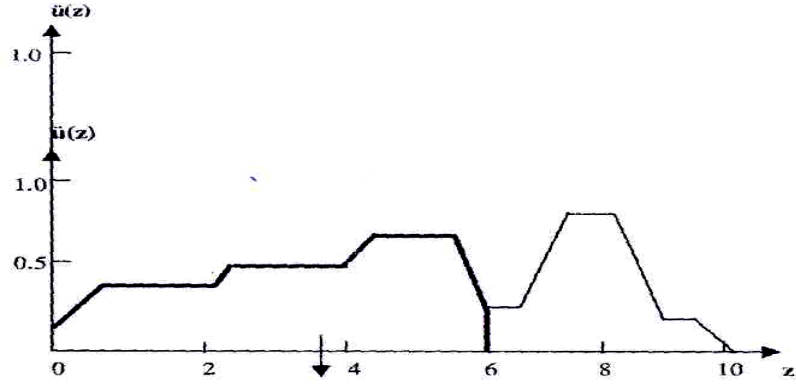
Şekil 2.19 Toplamların merkezi durulaştırması.

2.5.4.1.8 En büyük alanın merkezi

Eğer çıkış bulanık kümesi en azından iki tane dış bükey alt bulanık kümeyi içeriyor ise, dış bükey bulanık kümelerin en büyük alanlısının ağırlık merkezi durulaştırma işleminde kullanılır. Şekil 2.20’de gösterilen durulaştırma işleminin matematik hesaplaması

$$z^* = \frac{\int \tilde{u}_{eb\check{C}}(z)zdz}{\int \tilde{u}_{eb\check{C}}(z)dz}$$

eşitliğine göre yapılır. Burada $eb\check{C}(z)$ en büyük alanlı dış bükey bulanık kümenin hakim olduğu ait bölgeyi gösterir. Bu şart tüm çıkarım bulanık kümesinin dış bükey olmadığı zaman kullanılır, ama tüm çıkarımın dış bükey olması durumunda centroid yöntemi ile elde edilenin aynısıdır.



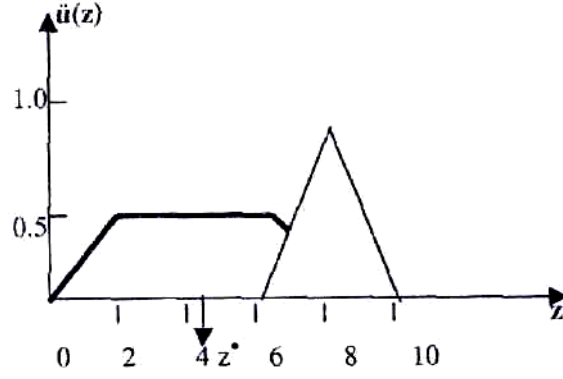
Şekil 2.20 En büyük alan merkezi ile durulaştırma

2.5.4.1.9 En büyük ilk veya son üyelik derecesi

Bu yöntem de, tüm çıktıların birleşimi olarak ortaya çıkan bulanık kümede en büyük üyelik derecesine sahip olan en küçük (veya en büyük) bulanık küme değerini seçmek esasına dayanır. Hesaplamaların vereceği z için aşağıdaki denklemler geçerlidir. Önce bulanık küme çıkarımı, B , birleşiminde en büyük, yükseklik, y_{eb} , tesbit edilir.

$$y_{eb}(B) \approx EB [\bar{u}_B(z)]$$

Bundan sonra birinci en büyük değer, z^* , bulunur. Bu yöntemin bir diğer seçeneği ise ilk yerine son en büyük bulanık küme değerinin, z , bulunmasıdır. Bu durumlar Şekil 2.21' de gösterilmiştir.



Şekil 2.21 İlk ve son en büyük üyelik dereceleri ile durulaştırma

Sonuç olarak bilgi tabanları çok sayıda bilgiden oluşur. Çıkarım yapabilmek için bilginin sunumu önemli bir konudur. Sunum yöntemlerine baktığımızda en yaygın olanı eğer - o halde kural tipidir.

Kural, bir içermeye (koşul eklemi, ise) olarak yorumlanır ve öncül (eğer bölümü) ve soncul (O halde bölümü) içerir. Eğer bir kural yukarıdaki şekilde verilmişse, " $x \text{ } a' \text{ dır}$ " şeklinde bir olgu var demektir. Buradan biz " $y \text{ } b' \text{ dir}$ " şeklinde yeni bir sonuç çıkarabiliriz.

Bilgi kuralları şeklinde sunulabilir. Kurallar iki değerli mantık sisteminde siyah-beyaz sonuçlar içerir. Bulanık mantık bu tikanıklığı aşmamızı sağlayan bir yaklaşım sunmaktadır. Bulanık bilgi tabanlı sistemlerde veriler üzerinde çıkarım ve akıl yürütme yapabilmek için iki değerli mantık yerine fonksiyon ve kuralların bulanık üyelik özellikleri kullanılır. Genel olarak bu sistemlerde sözel değişkenler kullanılır.

Bu sistemlerde çalışma 3-4 adımda gerçekleşir. Önce giriş değişkenleri üzerinde üyelik fonksiyonları belirlenir. Bu aşama bulandırmadır. Çıkarım aşamasında her bir kuralın öncül bölümü için doğru değerler hesaplanır ve bu değerler sonuç bölümüne uygulanır. Sonuçlar bulanık alt kümede olup her kuraldaki her çıkış değişkenine atanır. Her bir çıkış değişkenine atanmış bulanık alt kümelerin tamamı her bir çıkış değişkeni için bir tane bulanık alt küme oluşturacak şekilde birleştirilir. En sonunda da -eğer gerekiyorsa- bulanık çıkış kümesi kesin sayılara

dönüştürülmek üzere durulanır. Bu aşamaların hepsi için kullanılabilen çeşitli yaklaşımlar ve farklı işlemler tanımlanmıştır.

1979 da Zadeh yaklaşık akıl yürütme teorisine giriş yapmıştır. Bu teori belirsizlik ve bulanıklık karşısında akıl yürütme için güçlü bir yapı sağlamıştır. Bu kuramın merkezinde önermelerin temsilinde değişken değerleri olarak bulanık kümelerin atanmış ifadelerin sunumu yer alır. Bulanık kümeleri içeren bir kural ve bir olgu varsa, genelleştirilmiş modus ponens (GMP) ve genelleştirilmiş modus tollens (GMT) olarak iki tip akıl yürütme kullanabiliriz.

Bulanık içirme çıkarımı Zadeh tarafından ileri sürülen yaklaşık akü yürütme için bileşkesel çıkarım kuralı temeline dayanır. Bulanık akü yürütmede içirme bağıntısı olan $R(x,y)$ belirlenmesi ve bileşke işlemcisinin seçimi olarak iki konu bulunmaktadır. Genişletme ilkesi, matematiksel deyimlerin kesin alanını bulanık alana genişletmeyi sağlayan genel bir işlem olarak bulanık küme teorisinin temel kavramlarındanıdır.

Bulanık kümelerin bileşkesi kartezyen çarpımları ile elde edilir. Bulanık kümelerde çarpım uzayı için, bulanık tümel evetleme (ve) ve bulanık tikel evetleme (veya) işlemcileri kullanılabilir. Farklı çarpım uzaylarında bulanık bağıntılar, bileşke işlemi ile birleştirilebilir. Bulanık bağıntılar için farklı bileşke işlemleri önerilmiştir. Bunlardan iyi bilinen iki tanesi EB-EK bileşkesi ile EB-çarpım bileşkesidir.

Sözel olarak bir çıktı elde edilmesi gerekiyorsa, verilen bulanık kümeye en yakın sözel terim bulunmalıdır. En yakını bulmak için bulanık kümeler arasında mesafe ölçüm teknikleri kullanılabilir ve bu işlem sözel yaklaşıklştırma olarak tanımlanır. Sözel yaklaşıklştırmada, verilen bulanık kümeden en az uzaklıkta bir sözel terim bulunmaya çalışılır.

Bir bilgi sistemini modellediğimiz zaman, sıklıkla bulanık kural tabanı şeklinde temsil edilir. Bulanık kural tabanı, bulanık eğer-o halde kurallarını içerir. Çoğu olguda, bulanık kural tabanlı bulanık akü yürütme, bir düzey ileri veri sürümü çıkarımı (GMP, Genelleştirilmiş Modus Ponens)'e dayalıdır. Hiçbir bilgi tabanlı sistem tek kuraldan oluşmaz. Pek çok kuralın bileşkesi ve çıkarım mekanizmaları sayesinde bütüncül olarak ortaya konmuştur. Mamdani, Larsen, Tsukamoto, Takagi-Sugeno-Kang yaygın çıkarım yöntemleridir.

Pek çok pratik uygulamada, denetim komutu kesin bir değer olarak verilir. Bundan dolayı, bulanık çıkarım sonucunu durulamak gerekir. Durulama, elde edilmiş bir bulanık denetim etkinliğinde olasılık dağılımını en iyi gösteren, bulanık olmayan denetim etkinliği elde etme sürecidir. 30'dan fazla durulama yöntemi olmakla beraber bulanık denetleme teorisinde sıklıkla kullanılan dört durulama yöntemi bulunmaktadır. Bunlar en büyüklerin ortası, ağırlık merkezi yöntemi ile hesaplama, ortalamaların merkezi ve alan merkezi yöntemidir.

Durulaştırma sonucunda elde edilen tek değer, eldeki verilerin ışığı altında soruna iyi denilebilecek cevap vermesi beklenir.

2.6 Bulanık Mantık ve Bulanık Kontrol Sistemlerinin Kullanıldığı Yerler

Hidroelektrik güç üniteleri için kullanılan Baraj kapılarının otomatik kontrolü (*Tokio Electric Pow.*)

Stok kontrol değerlendirmesi için bir uzman sistem (*Yamaichi, Hitachi*)

Klima sistemlerinde istenmeyen ısı iniş çıkışlarının önlenmesi

Araba motorlarının etkili ve kararlı kontrolü (*Nissan*)

Otomobiller için "Cruise-control" (*Nissan, Subaru*)

Dökümanların arşivleme sistemi (*Mitsubishi Elec.*)

Depremlerin önceden bilinmesi için Tahmin Sistemi (*Inst. of Seismology Bureau of Metrology, Japan*)

İlaç teknolojileri: Kanser teşhisi (*Kawasaki Medical Scholl*)

Cep bilgisayarlarında el yazısı algılama teknolojisi (*Sony*)

Video Kameralarda hareketin algılanması (*Canon, minolta*)

El yazısı ve ses tanımlama (*CSK, Hitachi, Hosai univ., Ricoh*)

Helikopterler için uçuş desteği (*Sugeno*)

Çelik sanayinde makine hızı ve ısısının kontrolü (*Kawasaki Steel, New-Nippon Steel, NKK*)

Raylı metro sistemlerinde sürüş rahatlığı, duruş mesafesinin kesinliği ve ekonomikliğin geliştirilmesi (*Hitachi*)

Otomobiller için gelişmiş yakıt tüketimi (*NOK, Nippon Denki Tools*)

Assansör Denetimi; Yolcu trafiğini değerlendirir, böylece bekleme zamanını azaltır. (*Fujitec, Toshiba, Mitsubishi, Hitachi*)

Otomatik Şanzıman; Motor yükü, sürüş tarzı ve yol şartlarına göre uygun vites seçme durumudur. (*Honda, Nissan, Subaru*)

Fotokopi Makinesi; Resim yoğunluğu, sıcaklık ve nem oranına göre dram voltajını ayarlamak. (*Canon*)

Fotoğraf Makinesi; Birkaç obje olması durumunda en iyi fokusu ve aydınlatmayı belirler. (*Canon, Minolta Ekranda*)

Bulaşık Makinesi; Bulaşıkların miktarı ve kirlilik derecelerine göre yıkanma ve parlatma stratejilerini ve süresini belirlemek. (*Matsushita*)

Çamaşır Makinesi; Çamaşırın kirliliğine, ağırlığına, kumaş cinsine göre yıkama stratejilerini belirler. (*Matsushita, Daewoo*)

Elektrikli Süpürge; Yerin durumunu ve kirliliğini sezer ve motor gücünü uygun bir şekilde ayarlar. (*Matsushita*)

Mikro Dalga Fırın; Enerji sarfiyatını ve pişirme stratejilerini belirler. (*Hitachi*)

Televizyon; Her bir çerçeve için renk ve dağılımı ayarlamak ve odanın durumuna göre sesi stabilize eder. (*Gold Star, Sony*)

Kamera; Cihazın elle tutulması nedeni ile çekim sırasında oluşan sarsıntıları ortadan kaldırır. (*Panasonic, Canon*)

Tercüman; Kelimeleri tanıyıp tercüme etmek. (*Epson*)

Hisse Senedi Alışverişi; Makro ve Mikro Ekonomik verilere göre Japon Hisse Senedi piyasasını idare eder. (*Yamaichi*)

Klima Cihazı; Ortam koşullarına göre en iyi çalışma durumunu sezer, odaya birisi girerse soğutmayı artırır. (*Mitsubishi*)

Sendai Metro Sistemi; Hızlanma ve yavaşlamayı ayarlayarak rahat bir yolculuk sağlamanın yanısıra durma pozisyonunu iyi ayarlar, güçten tasarruf sağlar. (*Hitachi*)

ABS Fren Sistemi; Tekerleklerin kilitlenmeden frenlenmesini sağlar.
(Nissan)

Çimento Sanayi; Değirmende ısı ve oksijen oranı denetimi yapar.
(Mitsubishi Chem)

2.7 Bulanık Kontrolün Avantajları

1. Klasik mantığın temelinde ihtimal hesapları yatar. Bir olayın olabileceği ihtimalini bu mantıkla çözmeye çalışır. Sonuç evet ya da hayır ile sınırlıdır. Ancak bulanık mantık bundan tamamen farklıdır ve cevap olarak olasılığı değil ne kadar olduğunu verir. Dolayısıyla, alacağınız cevap evet ya da hayırla beraber bunların ara değerlerini de içerir. Örneğin bir endüstriyel denetim sisteminde ani sıcaklık değişimleri yerine yumuşak geçişlerle denetim sağlanır ve istenen ani değerler kullanılabilir. Böylelikle hem denetim kalitesi artırılmış hem de enerji tasarrufu sağlanmış olur.

2. Fuzzy'nin uygulama alanları çok geniştir. Sağladığı en büyük yararlar ise 'insana özgü tecrübe ile öğrenme olayının kolay modellenmesi ve belirsiz kavramların bile matematiksel olarak ifade edilebilmesine imkan tanımasıdır. Böylece Fuzzy lineer olmayan sistemlere yaklaşım yapabilmek için özellikle uygundur.

3. Fuzzy sisteminin işleme karmaşıklığı incelenen sistem karakteristiğinin doğrusal olup olmamasına bağlı değildir. Örneğin on kurallı bir Fuzzy sisteminin hızı her zaman aynıdır; fuzzy karakteristik eğrisi doğrusal, kübik, parabol, sinüs fonksiyonu veya eksponansiyel hız bakımından hiç bir şey fark etmez. Fuzzy programları tamsayı aritmetiği ile çalıştıkları için (8 bit,16 bit, 32 bit, kesinlikte) çıkarım algoritmaları çok büyük zorluklarla karşılaşmadan programlanabilir. Bu su demektir: karmaşık, doğrusal olmayan bir fonksiyon bile kayan noktalı hesap birimi

olmayan bir hesaplayıcı ile bulanık mantık kullanılarak gerçekleştirilebilir ve çok hızlı hesaplanır.

4. Fuzzy kontrol sistemi yardımıyla çok sayıda basit ya da karmaşık kontrol sistemlerinin problemsiz gerçekleştirilebileceği gibi mikrokontrolör ve PLC gerektiren sistemlerinde yapılması mümkündür.

5. Sözlü kumanda sistemlerinin (makine, tesisat, otomobil telefonu, ev cihazları, bilgisayar) dokunmadan çalışmasını sağlamak amacıyla fuzzy kullanılabilir. Uygulama türüne göre 10 ile 100 arasında sözlü komutu (ortam şartlarına rağmen) %90 - 95 doğrulukla algılayıp cihazları çalıştırma özelliğine sahiptir.

3. BULANIK MANTIK YAZILIMLARI

3.1 Bulanık Mantık İçin Yazılım Araçları

3.1.1 Giriş

Bu bölümde, bulanık sistemler, sinirsel ağlar ve sinirsel-bulanık sistemler için yazılım araçları anlatılmıştır. Seçim, aşağıdaki kategorilere bölünerek yapılabilir: grafik kullanıcı arayüzlü yürütülebilir dosyalar, grafik kullanıcı arayüzü olmayan yürütülebilir dosyalar ve C, C++, Java, MATLAB dosyalarında kaynak kodları vb. Yazılım ayrıca ticari, paylaşılan ve kamu alanlarına da bölünebilir. Bunlar, eğitici ve üretime yönelik araçlar olarak da kategorize edilebilir.

Burada tanıtılan yazılımların objektifleri farklıdır. Eğitim amaçlı araçlar, bulanık mantık ve sinirsel ağların temel kavramlarını gösterir ve basit sistemlerin operasyonlarının görselleştirilmesine yardım eder. Üretim için olanlar, sistem tasarımı ve analize yöneliktir. Burada hepsi anlatılmıştır çünkü eğitici bilgiler ekler ve okuyucuya, minimum zaman ve para yatırımıyla çok sayıda yazılım araçlı ilk el deneyimine dayalı deneme ve yargılama olanağı verirler.

Yazılım seçme kriteri aşağıdaki şekilde özetlenebilir:

Bulanık sistemler, sinirsel ağlar ve her ikisine olan ilgi.

İngilizce versiyonunun olması.

Dökümantasyonun olması veya yazılımın sezgisel kullanılabilmesi veya her ikisi.

Yazılımın ya ücretsiz ya da eğitici demosunun olması

Yazılımın ve dökümantasyonun web’de bulunabilmesi.

Araçlar, alfabetik sırayla düzenlenmiştir, bulanık ve sinirsel kaynak kodlarını kullanıcının anlaması için doğal olarak; Matlab, C, Java, Perl vb. hakkında bazı bilgileri bilmesi gerekmektedir. FuzzyTECH, FIDE ve Neuro Solutions gibi ticari üretim araçları, hem eğitim hem de büyük ölçekli projelerin tasarımı için yararlı olabilir.

3.1.2 Genel Bakış

3.1.2.1 C ve C++ Kaynak Kodları

3.1.2.1.1 The ART Gallery

<http://cns-web.bu.edu/pub/laliden/WWW/nnet.frame.html>

The ART Gallery, Lars Liden tarafından, ART stili sinirsel ağların çoğunun tamamlanması için diğer kodlarla kullanılmak üzere geliştirilmiş, ücretsiz bir prosedür seti. Visual Basic gibi Windows ortamı için bir dinamik bağlantılı kütüphane derlemesinde olduğu gibi, Unix ve DOS platformlarının her ikisinde de çağrılan C prosedürünü destekler.

3.1.2.1.2 The Binary Hopfield Net

<http://www.geocities.com/ResearchTriangle/facility/7620>

Site, Matt Hill tarafından geliştirilen bir java uygulaması sağlar. Hopfield ağına çalışmasını gösterir ve kamu malı yazılımı olarak kaynak kodu verilmiştir.

Dikdörtgen içindeki karelere tıklayarak fareyle bir örüntüye girebilir. Ağ, örüntüyü depolar; yaklaşık 33 kadar rasgele örüntü depolanabilir. Bazı örüntüler depolandıktan sonra tanımak için yeni bir örüntüye (depolanmış örüntülerden biri ama önemsiz derecede bozuk olan) girmeye çalışılabilir ve sonra ağ yerleşimi izlenir.

3.1.2.1.3 Neural Networks at your Fingertips

<http://www.geocities.com/CapeCanaveral/1624/>

Site, Karsten Kutza tarafından geliştirilen, en popüler sinirsel ağ mimarilerinin sekiz tanesi için yazılım simülatörleri sağlamakta. Ücretsiz olarak indirilebilen yazılımlar şunlardır;

ADALINE, BPN (Geri yayımlı Ağ - Backpropagation Network), HOPFIELD, BAM (Doğrultusuz Çağrışımlı Bellek-Bidirectional Associative Memory), BOLTZMAN (Boltzmann Makinası), CPN (Sayıcı yayılım Ağı-Counterpropagation Network), SOM (Kendi Kendini Organize Eden Eşlem- Self-Organizing Map), ART 1.

3.1.2.1.4 Pittnet

<http://www.pitt.edu/Aesmith/postscript/guide.pdf>

Pittnet, Brian Carnahan ve Alice Smith (University of Pittsburg) tarafından eğitim amaçlı tasarlanmış bir sinirsel ağ programıdır. Derlenmiş C++ kaynak kodu, derlenmemiş C++ kaynak kodu ve Kullanıcı Rehberi indirilebilir. Yazılım aşağıdaki sinirsel ağ paradigmalarına sahiptir: Geri yayılım, Kohonen kendinden organize, ART1 ve radyal havuz fonksiyonu RBF. Girdi ve çıktılar Pittnet ASCII yazı dosyalarıdır.

3.1.2.1.5 UNFuzzy

<http://ohm.ingsala.unal.edu.co/ogduarte/Software.htm>

Oscar Duarte (Universidad Nacional de Colombia) tarafından geliştirilmiş bir programdır. İngilizce ve İspanyolca dahil olmak üzere çeşitli dillerde mevcuttur.

3.1.2.1.6 MS-DOS NN Freeware

<http://www.simtel.net/pub/msdos/neur1net/>

Bu site MS-DOS altında çalışan bedava indirilebilir çok sayıda sinirsel ağ programına bağlantılar ve açıklamalar içermekte. Bazı programlar:

Ann110; Çok seçenekli NN eğitim programı

Bam; bir Doğrultusuz Çağrışimli Bellek simülasyonu

Bpnn132; Uyumlu öğrenme oranlı bir geriyayılım sinirsel ağı

Bps100; Bir geriyayılım simülatörü

Brain12; Bir geriyayılım sinirsel ağ simülatörü

Et; Bir perceptron simülatörü

Nerves; C++ kaynak kodlu Sinirsel Sistem Konstrüksiyon takımı

Neurfuzz; NN ve bulanık makine eğitme ve kaynak kodu üretme programı

NLmos; Perceptron, geri, yayılım ve kendinden organize eşlemler için eğitici bir simülasyon

Nnut1101; Bir C- kaynak kod kütüphanesi ve sinirsel ağlar için başvuru makalesi.

3.1.2.1.7 StarFLIP

<http://www.dbai.tuwien.ac.at/proj/StarFLIP>

Database and Expert Systems Group, Institute of Information Systems, Vienna University of Technology tarafından, bulanık kısıtlamalarla birleşimsel problemler için tasarlanmış bir C++ kütüphanelerinin yeniden kullanılabilir yineleyici optimizasyon ailesi. StarFLIP++ aşağıdaki tabakalı alt kütüphaneler dahil edilerek geliştirilmekte:

Bulanık mantık çıkarım işlemci kütüphanesi, FLIP++

Bulanık kısıtlar kütüphanesi, ConFLIP++

Dinamik kısıt üretim kütüphanesi, DynaFLIP++

Alan bilgi tanıtım kütüphanesi, DomFLIP++çeşitli buluşsal optimizasyon kütüphanesi, OptiFLIP++

Kullanıcı arayüzü, InterFLIP++

HTML dökümantasyonu, DocuFLIP++

Bilgi değişim tutarlılık denetçisi kütüphanesi, CheckFLIP++

Tüm kütüphane seti için versiyon kontrol ve test çevresi, TestFLIP++

Simülasyon takımı, SimFLIP++

Tepkisel optimize, ReaFLIP++

Bulanık üyelik fonksiyonlarının otomatik ayarlanmasına izin veren sinirsel ağ genişlemesi, NeuroFLIP++

3.1.2.2 Analog Devre Tasarımı için Bulanık Mantık Tabanlı bir Koleksiyon

<http://www.gte.us.es/usr/chavez/collect.pdf>

Chavez ve L. G. Franquelo tarafından yazılmış ve IEEE Micro, Ağustos 1996' da yayınlanmış bir makale. Analog tasarım için bulanık mantık tabanlı takımlar koleksiyonu üç ana bölümde tanıtılmış bunlar; topoloji seçimi, modelleme ve optimizasyon ve test etme.

3.1.2.3 EasyNN

<http://www.easynn.com/easynnbased.html>

EasyNN, Stephan Wolstenholme tarafından sunulan paylaşılabilen bir program. Yazı dosyalarından veya gridlerden çok tabakalı sinirsel ağlar üretebilmekte. Ağlar daha sonra eğitilebilir, onaylanabilir ve sorgulanabilir. Ağ diyagramları, grafikler, girdi/çıkı verisi ve tüm ağ ayrıntıları gösterilebilir ve basılabilir. Ağ öğrenirken, sinirler (düğümler) eklenebilir veya silinebilir. Yazılımın adım adım çalışma notuna, örnek dosyalara ve yardım dosyalarına sahip.

3.1.2.4 Fuzzy Control Manager (FCM)

<http://www.transfertechn.de/www/soft e.htm>

Bu, her karmaşıklıkta bulanık sistemleri grafik olarak tanımlamak için kullanılabilen bir Windows programı. Konuyla ilgili bütün veri, geliştirilirken, hata giderilirken veya optimize edilirken görüntülenebilir. Sistemin yapısında, değişken veya tanımlı kural sayısında bir sınır yoktur. Üyelik fonksiyonları, sınırsız sayıda büküm noktası (veya istenirse çıktılar için singletonlar) ile tanımlanabilir. Bulanık kontrolcü inşasını tamamlama kodları kullanılan FCM versiyonuna bağlı olarak, C dilinde çevirici veya ikili üretilir. FCM versiyonları şunlardır.

FCM-3000; FP-3000 adaptör kartında çalışır ve FP-3000 işlemcisiyle uygun geçiş gerçekleştiren özel C kodu üretir

FCM-SOFT; FP-3000 işlemci içeren VMEbus kartının online çalışmasını destekler

FCM-TEAM; çok analoglu ve dijital I/O hatlarına sahip mikrokontrolcü kartını destekler

FCM-SPS; FP-3000 içeren bir programlanabilir kontrolcünün bulanık modülünü doğrudan programlamak için kullanılır

FCM-Modicon; NEC 78K0, 78K3, V serisi mikrokontrolcüler için kod üreten AEG_Modicon FCM-13HD-CINT' nin bulanık programlanmasını destekler

FCM-13HD-75X; NEC 75X ailesinin mikrokontrolcleri iin kod retir

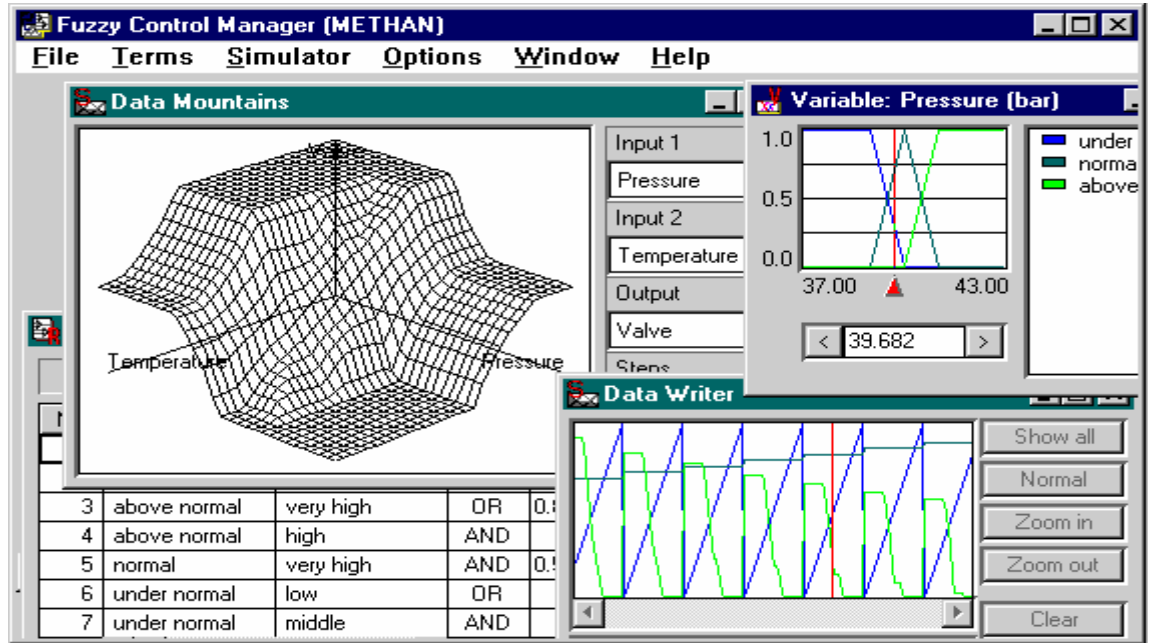
FCM-8051; 8051 ailesinin mikrokontrolcleri iin kod retir

FCM-PX-1500; Panasonic MN1500 mikrokontrolc ailesi iin kod retir

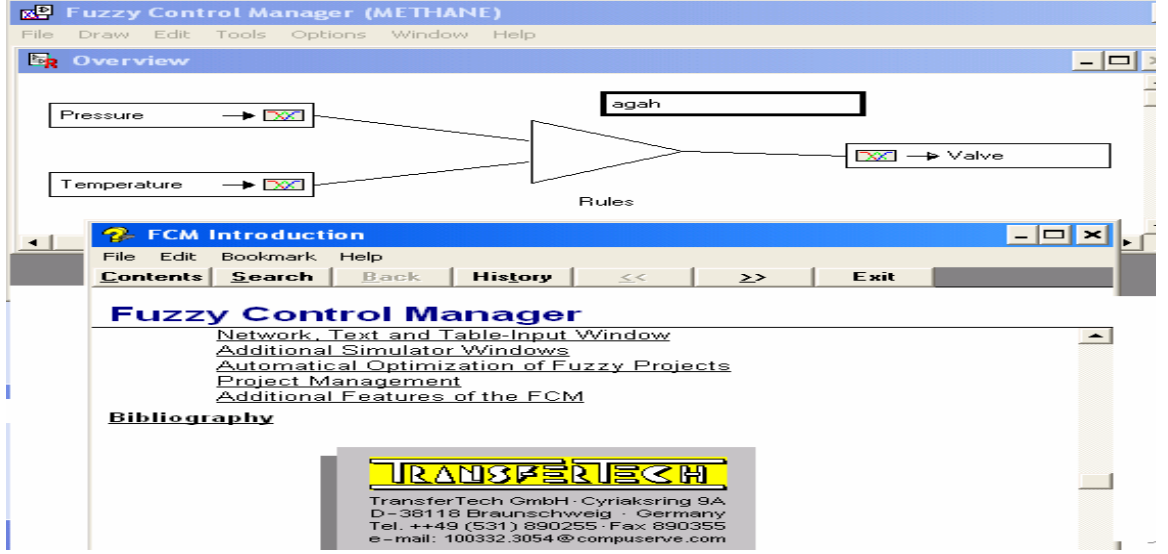
FCM-HC11; 68HC mikrokontrolcler iin kod retir

FCM-8051; 8051 ailesinin mikrokontrolcleri iin kod retir

rnek dosyalarıyla bir demo indirilebilir ve ekran grntlerine Web zerinde ulařılabilir. řekil 3.1.1 'de ve řekil 3.1.2' de Fuzzy Control Manager ekran grntleri verilmektedir.



řekil 3.1.1 Fuzzy Control Manager Arayz Ekranı

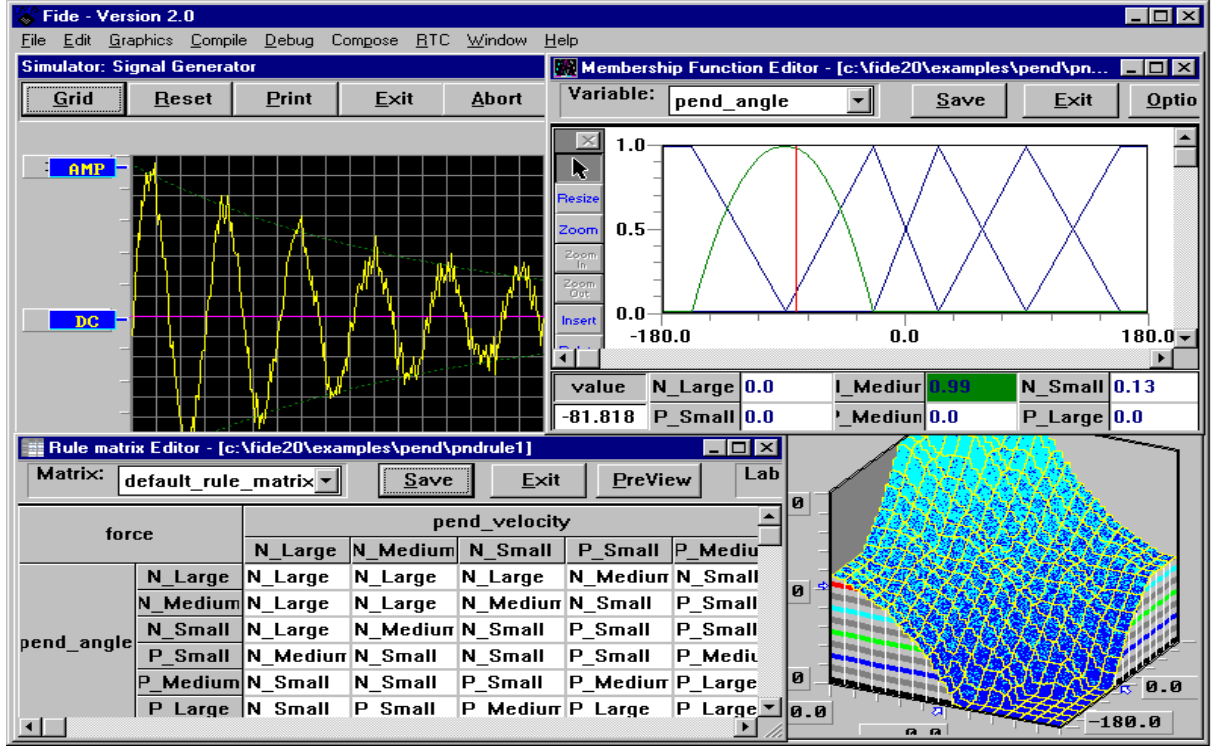


Şekil 3.1.2 Fuzzy Control Manager Yardım Ekranı

3.1.2.5 Fuzzy Inference Development Environmet (FIDE)

<http://www.aptronix.com/fide/>

FIDE, Aptronix'in (Santa Clara, USA) bulanık mantık sistemleri için gelişme ortamıdır. Yazılım Java, ANSI, C MATLAB M-File dillerinde otomatik olarak bulanık algoritmalar üretir ve çeşitli mikrokontrolcüler için kodları birleştirir. Çiper şunları destekler: Motorola: 68HC05, 6805, 68HC08, 68HC11, 68HC12, 68HC33x, Intel 80C196 ve 80C296 mimarileri, Siemens: SAE81C99A ve Omron Electronics: FP-3000, FP-5000. FIDE' nin ücretsiz demosu ve örnek dosyaları indirilebilir. Demo, bulanık mantık modellerinin yaratılması ve simülasyonuna izin verir ve bulanık çıkarım işleminin öğrenilmesine yardım eder. Demo eğitim amaçlı olarak ve tüm versiyonunun ön izlemesi olarak yararlıdır. Şüphesiz üretim ortamı için değildir, tümleyici ve kod üreticisi bulunmamaktadır. Şekil 3.2'de FIDE'nin arayüz ana ekranı gösterilmektedir.



Şekil 3.2 FIDE Arayüz Ekranı

3.1.2.6 Fuzzy Logic Development Environment (FLDE)

<http://www.flde.com/flde/index.htm>

FLDE, karmaşık gömülü kontrol uygulamaları geliştirmek için bir yazılım aracıdır. Kontrolcü davranışının doğal dil özelliklerinden hedef donanım mimarileri için C kodu üretir. İki örnek sunulmuştur.

İlki bulanık mantıkla gömülü otomotiv uygulamaları inşasıdır: Stylianos S. Sbyrakis (Syndesis Ltd. Atina) tarafından NiCd Batarya Şarjı. İkincisi Ter çevrilmiş sarkaç bulanık kontrol.

3.1.2.7 Fuzzy CLIPS

http://ai.iit.nrc.ca/IR_public/fuzzy/fuzzyClips/fuzzyCLIPSIndex.html

FuzzyCLIPS, NASA' nın CLIPS (C –Dili Entegre Üretim Sistemi- C- LAnguage Integrated production System) uzman sisteminin genişletilmiştir. Integrated reasoning Group of the Institute for Information Technology of the National Research Council of Canada tarafından geliştirilmiştir.

FuzzyCLIPS, bulanık ve normal terimlerin, uzman bir sistemin kuralları ve olguları içine serbestçe karışmasına izin vererek tam, bulanık ve kombine muhakeme ile ilgilenebilir. Ticari olmayan kullanım için ücretsiz indirilebilir, ama ticari kullanımda lisans gereklidir.

3.1.2.8 Fuzzy Control Demo, (FCD)

<http://www.cseecwvu.edu/~esazonov/loadswayindex.htm>

Eduard Sazanov (West Virginia University, USA) tarafından sunulan interaktif bir örnek. Bir yükleme vincinde bulanık mantığın kullanımı gösterilmekte. Gösterimde vinci kontrol eden bulanık kurallar herhangi bir anda değiştirilebilir. Üç dilsel değişken tanımlanmış: açı, uzaklık ve güç. Dilsel değişkenler ve üyelik fonksiyonları interaktif olarak değiştirilemez. Demonun yazarı, sadece tek hareket yönünde çalışan FuzzyTech örneğinin veya karmaşık sallanma karşıtı kuralları kullanan Omron örneğinin aksine, verilen model yük sallanmasını gidiş yönüne yakın uzaklıkta gideren çok basit iki kural çalıştırır. Yükün her asılımı için, vincin vargeli genliği azaltmak için asma yönüne taşınır. Yük sallanmasını giderme olmadan sistemin davranışını gözlemek için settteki son iki kural kaldırılır.

3.1.2.9 FuzzyCOPE

<http://kel.otago.ac.nz/software/FuzzyCOPE3/>

FuzzyCOPE, Knowledge Engineering Laboratory, KEL (University of Otago) tarafından sağlanan bir yazılım. Hibrit bağlantılı bir yazılım ortamı. Sistemin işlevselliği için grafik kullanıcı arayüzüne sahip. Sistemin çekirdeğini şekillendiren DLL (dinamik Link Kütüphaneleri- Dynamic Link Libraries) sunulan programlama kütüphanelerinin kullanımıyla Windows tabanlı zeki sistemlerin geliştirilmesinde kullanılabilir. Komut satır araçları DOS bilgi sisteminden bağlantılı yapıların yaratılması, eğitilmesi ve onaylanmasına izin verir. Yazılım yukarıdaki verilen adresten ücretsiz olarak indirilebilir.

3.1.2.10 Fuzzy Decision Tree, (FID)

<http://www.cs.umis.edu/~janikow/fid/>

FID, Cezary Janikow (Mathematics and computer Science, UMLS St. Louis) tarafından geliştirilmiş bir program. Bulanık veriden, bulanık mantık tabanlı karar ağacı üretir. Ağaç daha sonra çeşitli çıkarım yöntemleri kullanarak bilinmeyen sınıflandırma ile veriyi sınıflandırmak için kullanılabilir. Kullanılan düşünceler kısmen “Fuzzy Decision Trees: Issues and Methods” IEEE Transactions on Systems, Man, and Cybernetics, 1998’ de yayınlanan makalede açıklanmıştır.

3.1.2.11 FuzzGen

<http://www.programmersheaven.com/zone22/cat167/1244.htm>

Bu ücretsiz ve belki de ilk bulanık mantık kod üreticisi. Kullanıcıya tüm bulanık karar alma algoritmalarını bir arada sunar ve karar işlemini tamamlayacak Pascal, Basic veya C/C++ dillerinde kod üretmesine olanak verir. Yazılım sezgisel olarak kullanılabilir ama bir yardım dosyası da bulunmaktadır.

3.1.2.12 Fuzzy Logic Control in VHDL

<http://www.ee.uaiberta.ca/~elliott/ee552/studentAppNotes/2000f/chdl/fuzzyController/>

Steve Dillen ve Farrah Rashid tarafından yazılmış ve Duncan Eliot (University of Alberta, Canada) tarafından sunulmuş bir bildiri. Makale HTML formatında ve kaynak kodlarına ve yazılımla ilgili daha geniş bilgiye ulaşmak için linkler var.

3.1.2.13. FuzzyTECH

<http://www.fuzzytech.com/>

fuzzyTECH, bulanık mantık ve sinirsel-bulanık sistemler için yazılım araçları geliştirme ailesi. Yazılım İngilizce ve Almanca dillerini desteklemektedir.

Ayrıca, İngilizce, Almanca ve Mandarin Çince dillerinde dökümantasyon mevcuttur. Özel amaçlı *fuzzyTECH* aileleri şunları içerir;

fuzzyTECH Editions: *fuzzyTECH* Professional ve *fuzzyTECH* online basımları dahil olmak üzere Genel Amaçlı Donanım.

Gömülü Kontrol için *fuzzyTECH* MCU:

* *fuzzyTECH* MCU-HC05/08; tüm Motorola 68HC05xx ve 68HC08xxleri destekler

* *fuzzyTECH* MCU-ST6; tüm STMicroelectroncd ST6 mikrokontrolcülerini destekler

* *fuzzyTECH* MCU_Mp; tüm mikroçiplerin mikrokontrolcülerini destekler

* *fuzzyTECH* MCU-51; tüm 8051 ve 80251 mikrokontrolcülerini destekler

* *fuzzyTECH* MCU-374xx; tüm 8 bitli Mitsubishi mikrokontrolcülerini destekler.

* *fuzzyTECH* MCU-HC11/12; tüm Motorola 68HC11xx ve 68HC12xx ailesi mikrokontrolcülerini destekler.

* *fuzzyTECH* MCU-96, tüm Intel MCS96 mikrokontrolcülerini destekler

* *fuzzyTECH* MCU-320; Texas Instruments dijital sinyal işlemcilerini destekler.

fuzzyTECH IA, Siemens SIMATIC programlanabilir Mantık Kontrolcülerini S5 için *fuzzyTECH* IA-S5/7 dahil olmak üzere Endüstriyel Otomasyon için uygulama örnekleri C. Von Altrok tarafından çeşitli raporlarda anlatılmıştır.

Bunlar:

Fuzzy Logic in Automotive Engineering, Circuit Cellar, 1997, www.circuitcellar.com/pastissues/articles/misc/88constantin.pdf adresinden ulaşılabilir.

Practical fuzzy Logic Design, 1996 www.circuitcellar.com/pastissues/articles/misc/75constantin1.pdf

A Fuzzy Logic thermostat, Circuit Cellar, 1996 www.circuitcellar.com/pastissues/articles/misc/75constantin2.pdf

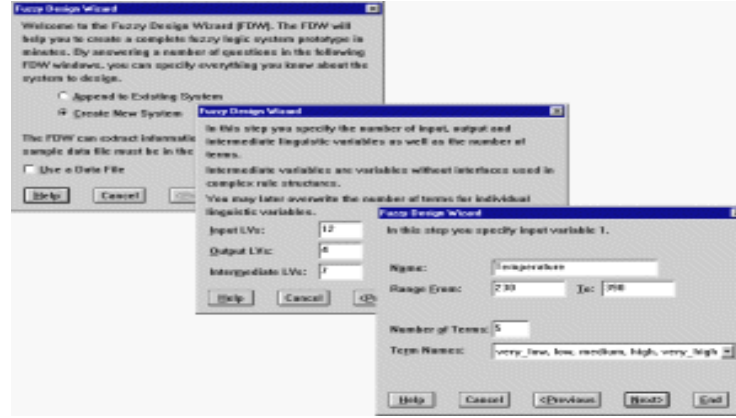
Bir *fuzzyTECH* teknik uygulamalar kütüphanesi de online olarak bulunmaktadır. Bu aşağıdaki konu başlıklarında raporlar içerir:

Pratik Tasarım, Endüstriyel Otomasyon, Kompleks Soğutma Sistemleri, AC İndüksiyon Motoru, Su Arıtma Sistemi, Kamyon Hız Sınırlayıcısı, Otomotiv Mühendisliği, ABS, Uçak Uçuş Yörüngesi, Nükleer Füzyon, Motorola 68HC12 MCU, Trafik Kontrolü, Bulanık Mantık Standardizasyonu, Losar sistemler ve Bulanık Mantık Tasarımı: Metodoloji, Standartlar ve Araçlar.

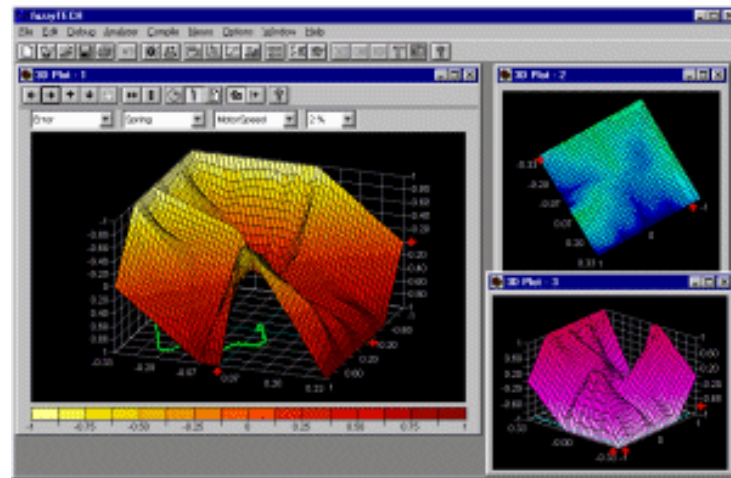
fuzzyTECH uygulamalarıyla ilgili daha fazla bilgi, tartışmalar ve resimlere şu adresten ulaşılabilir:

<http://www.neurotech.com.sg/html/archive/supplearchivefiles/fuzzy.html>

Şekil 3.3.1 ve Şekil 3.3.2' de *fuzzyTECH*'in windows ortamında çalışan arayüzünün ekran görüntüleri verilmektedir.



Şekil 3.3.1 Bulanık Mantık Editörü (Fuzzy Logic Design, *fuzzyTECH*)



Şekil 3.3.2 Bulanık Mantık Editörü Fuzzy Logic Design, *fuzzyTECH*

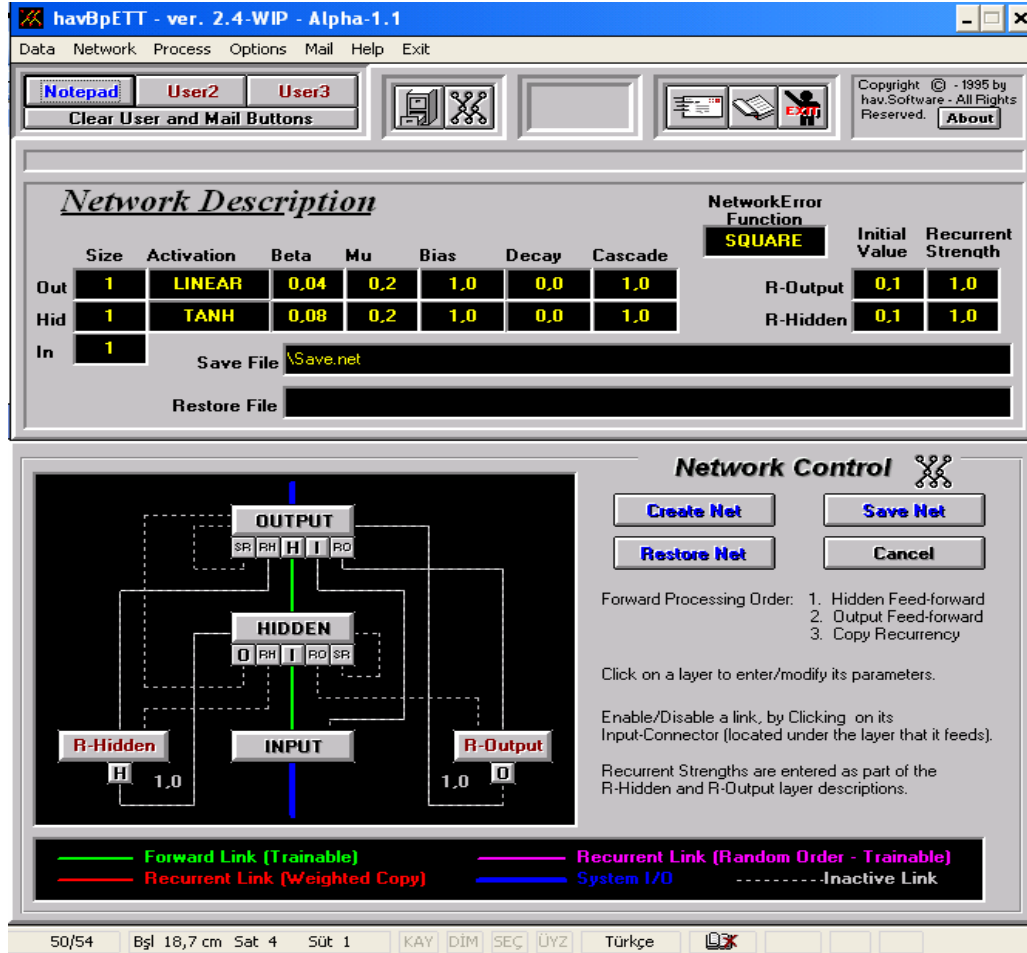
3.1.2.14 HavBpETT

<http://www.hav.com/default.html>

HavBpETT, havBpETT++C++ sinirsel ağ sınıf kütüphanesinin DLL versiyonunun kullanımını gösterir. Demo kullanıcıya basit ileri beslemeli, yinelenen veya ardışık sinirsel ağların açıklanması, eğitilmesi ve saklanması olanağını verir. İndirilebilir demolar tam işlevsel simülatörle aynıdır ancak ağ saklama fonksiyonu yoktur. havBpETT demosunun ana kullanıcı arayüzünün ekran görüntüleri Web sitesinde bulunmaktadır. Şekil 3.4.1 ve Şekil 3.4.2’de ver.2.4-WIP-ALPHA HavBpeTT’nin ekran görüntüleri verilmiştir.



Şekil 3.4.1 HavBpeTT Arayüz Ana Ekranı



Şekil 3.4.2 HavBpeTT Arayüz Ekranı

3.1.2.15 Hyper Logic Demos

<http://www.hyperlogic.com/demos.html>

Bu program, bazıları Windows bazıları da Ms-dos altında çalışan demo programları sunmakta. Ana yazılımı CUBICALC. Program demosun da gösterildiği gibi açılabilen ve yoklanabilen çeşitli örnek dosyalara sahip. Problemlere ayrıntılı olarak açıklanmış ve bir daire içinde çalışan simülasyonunu ve ekran görüntüsü üçüncü bölümde gösterilmektedir.

Program aynı zamanda C altprogramlarının OWL sinirsel Ağ kütüphanesinin nasıl çalıştığını açıklayan bir Ms-dos uygulamasının çalışmasında sağlamaktadır. Geri yayılım ve rekabetçi öğrenme kuralının kullanıldığı örnekler de içermektedir. Örnek program için kaynak kodu da sunulmuştur. Bulanık kontrol örneklerine ek olarak, bazı sinirsel ağ işlemleri ve bulanık mantık uygulamalarını gösteren çeşitli Ms-dos ve Windows dosyaları da sunulmuştur.

3.1.2.16 Jan's Fuzzy System, (JFL)

<http://inet.uni2.dk/~jemor/jfs.htm>

Jan Mortensen (Danimarka) tarafından anlatılan JFL programlama dilinin bir gelişme oramı. Ortam da, Jfl programlarını tümleyen çalıştıran, geliştiren ve dönüştüren araçlara sahip. JFL, bulanık mantık gibi geleneksel programlama dillerinden özellikleri ve makine öğrenimini birleştiriyor. Machine learning uygulamalı dersi için kullanılabilir bir program. derlenmiş bir Jfs programı, bir Dll dosyasından yada C programlarından komut satırlarıyla uygulamaya geçilebilir. Javascripte dönüştürülmüş bir programla HTML dosyasına çevrilebiliyor. Ayrıca kaynak kod dosyasına da dönüştürülebiliyor ve C programlarına da eklenebiliyor.

3.1.2.17 Java Related

<http://joone.sourceforge.net/>

Joone, sinirsel ağı yaratmak, eğitmek ve test etmek için ücretsiz bir sinirsel ağ iskeleti. Java teknolojileri tabanlı. Joon NN, varolan ve geliştirilecek olan bütün uygulamaların odağı olan merkezi bir makine tarafından birleştirilir.

Yapay sinir Ağı ve Yapay yaşam için program apletleri;

Akio Utsugi tarafından sunulan bir java uygulamacılığı koleksiyonu. Uygulamalar aşağıdaki gibi çeşitli kategorilerde sınıflandırılmıştır.

Rekabetçi öğrenme

- Vektör niteleyicisi
- Gauss karışık modeli
- Çoklunominal karışık model
- Çeşitli kendinden organize eşlemeler
- İnteraktif Som
- Kohonen özellik
- Bayesian kendinden organize eşlemeler
- Değişik türdeki rekabetçi öğrenme ile topoloji reformasyon

arasında karşılaştırma.

Geri yayılım öğrenme

- Fonksiyon yaklaşımı öğrenme
- Yapay sinirsel öğrenme üzerine interaktif notlar
- Yapay sinirsel ağ el yazısı tanıma
- Optic karakter tanıma

Kısıtlı Tatmin ve optimizasyon için sinirsel ağlar

- Hopfield ağı
- 8 – Kraliçe problemi
- Boltzman makinesi
- Örüntünün yeniden inşası
- Hopfield ağı ile içerik adresleme belleği

3.1.2.18 Labview

<http://www.ni.com/downloads>

Labview, National Instrument'in bir grafiksel uygulama geliştirme programı. **Labview** programları virtüel aletler olarak adlandırılır çünkü onların görünüşü ve işlermi güncel aletlere benzer.

Arge simülasyon news, Aegesim, Avrupa'da modelleme ve simülasyon bilgisinin yayılması için altyapı ve yönetim sunan, kar amacı gütmeyen bir çalışma grubudur. Viyana teknoloji Üniversitesi'nin simülasyon bölümünde bulunuyor. Simülasyon News Europe 'da çeşitli karşılaştırmalardan olan "iki tank sisteminin bulanık kontrolü" konulu bir karşılaştırması mevcuttur.

3.1.2.19 Lingusitic Fuzzy Logic Controller, (LFLC)

<http://ac030.osu.cz/irafm/lflc/lflc.html>

Lingusitic Fuzzy Logic Controller, Çek cumhuriyetindeki lojik modeller araştırma ve uygulama merkezinde sunulmuştur. Yazılım iki farklı ana çıkarım yöntemine sahiptir: Bir tanesi IF/Then kuralları yorumuna dayalı dilsel açıklanmış mantıksal içermeler ve diğeri bilinmeyen bir fonksiyonun interpolasyonu olan standart maks-t kuralı.

3.1.2.20 Matlab

[http:// www.etse.urv.es/Aoller/fuzzy/fuzzy logic.htm#4](http://www.etse.urv.es/Aoller/fuzzy/fuzzy%20logic.htm#4)

[http:// www.control.hut.fi/Kurssit/AS-74.115/Material/fuzzy2.pdf](http://www.control.hut.fi/Kurssit/AS-74.115/Material/fuzzy2.pdf)

[http:// www.mathworks.com](http://www.mathworks.com)

MATLAB, bilimsel ve teknik hesaplama için ticari olarak bulunabilen interaktif ortam ve programlama dili. Ayrıca öğrenciler için indirimli bir versiyon da bulunmakta. C veya C++ ile yazıldığından çok daha kısa zamanda pek çok sayısal problemin çözümüne olanak tanımakta. Matlab, kullanıcıya yeniden kullanabileceği kendi araçlarını inşa etmesine izin veriyor. Kullanıcı kendi özel fonksiyonlarını ve programlarını Matlab ortamında çalıştırabilir. Bu tür dosyalar M files olarak bilinir. Selçuk üniversitesi Bulanık mantık uygulamaları dersinde de bu tür bir özel yazılım tasarlanmış ve bu ders içeriğinde ayrıntılı olarak kullanılmaktadır. Belirli bir problem sınıfından söz eden Matlab fonksiyonları, bir araç kutusu kavramı oluşturacak şekilde birlikte gruplanabilir. Bu bağlamda Matlab'deki araç kutuları özel bir iş için tasarlanmış bir M files'ların butonları ve özel tasarlanmış işlerin kullanım alanları olmuş olacaktır.

Matlab'deki araç kutularının bir kaçına açıklamalarla birlikte bakarsak;

İletişim araç kutusu: İletişim sistemlerinin tasarımı ve analizi için

Kontrol Sistemi araç kutusu: Geri beslemeli kontrol sistemlerinin tasarımı ve analizi için

Eğri uydurma araç kutusu: Model uydurma ve analizinde

Veri kazanma araç kutusu: Takılabilir veri kazanım kartlarından veri kazanımı ve dışarı gönderimi için

Veri tabanı araç kutusu: Bağlantılı veri tabanı ile veri değişimi için

Veri belslem araç kutusu: Veri servis sağlayıcılarından gerçek zamanlı finansal veri kazanımı için

Genişletilmiş matematik sembolü araç kutusu: Matematik sembollerini ve çeşitli aritmetik işlemleri kullanarak hesaplamalar yapmak için

Filtre tasarım araç kutusu: kayar noktalı ve sabit noktalı ileri filtreli tasarım ve analizinde

Fuzzy logic araç kutusu: bulanık mantıklı sistemlerin tasarım ve analizini desteklemek için. Geliştirme, araştırma, tasarım, simülasyon ve gerçek zamanlı tamamlama dahil olmak üzere işlemin tüm aşamalarını destekler. kullanıcıya bulanık çıkarım sisteminin basamaklarında rehberlik edecek sezgisel ortam sağlamak için grafiksel kullanıcı arayüzü kullanır. Bulanık kümeleme ve uyumlu sinirsel –bulanık öğrenme gibi pek çok bulanık mantık yöntemleri için fonksiyonlar sunulmuştur.

Resim işleme araç kutusu: resim işleme, analiz ve algoritması geliştirmek için

Alet kontrol araç kutusu: test ve ölçü aletlerinin kontrolü ve iletişimi için

LMI kontrol araç kutusu: karmaşık optimizasyon teknikleri kullanarak ağır iş kontrolcülerinin tasarımı için

Eşleme araç kutusu: coğrafi tabanlı bilginin analizi ve gösterimi için

Model öngörücü kontrol araç kutusu:

Model bazlı kalibrasyon araç kutusu: karmaşık güç zinciri sistemlerini kalibre etmek için

μ -analizi ve sentezi araç kutusu: model belirsizliği olan sistemler için çok değişkenli geri beslemeli kontrolcülerin tasarımında

Sinirsel ağlar araç kutusu: Sinirsel ağların tasarımı, tamamlama, görselleştirme ve simülasyonu için. Grafik kullanıcı arayüzün de olduğu gibi, pek çok ağ paradigmalarına geniş destek sağlar. Olasılıkla Sinirsel Ağ kullanarak Sınıflandırma, radyal temelli ağlarla fonksiyon yaklaşımı gibi elemanlar ve açıklamalar bulunmaktadır.

Optimizasyon araç kutusu: standart ve büyük ölçekli optimizasyon problemlerinin çözümü için

Kısmi diferansiyel eşitliği araç kutusu: kısmi diferansiyel eşitliklerinin analizi ve çözümü için

Robust kontrol araç kutusu: çok değişkenli geri beslemeli kontrol sistemlerinin tasarımı için

Sinyal işleme araç kutusu: sinyal işleme, analiz ve algoritma geliştirimi için

İstatistik araç kutusu: istatistik algoritmaları ve olasılık modelleri uygulamak için

Simgesel matematik araç kutusu: simgesel matematik ve tam değişken aritmetiği kullanarak hesaplama yapmak için

Sistem tanımlama araç kutusu: ölçülen girdi/çıktı verisinden doğrusal dinamik modeller yaratmak için

Virtüel gerçeklik araç kutusu: üç boyutlarda Simulink sistemlerini canlandırmak ve görselleştirmek için

Dalgacık araç kutusu: dalgacık tekniklerini kullanarak sinyal ve resimlerin analizi, baskısı ve gürültü giderme için

Son çıkan Matlab sürümlerinde pek çok ticari araç kutusu bulunmaktadır. Buna ek olarak ve en önemlisi, en iyi araştırmacı ve yazılımcılar tarafından geliştirilmiş ve kamu alanına konulmuş araç kutuları ve M-files ' lerin olmasıdır. Ayrıca, Matlab e benzeyen ve kamu alanlarının hizmetine sunulmuş birçok yazılım ve program tasarlanmıştır. Bunlara başlıklar halinde bakarsak;

Gnu Octave: Bu dil Matlab'e çok benzeyen ama bedava olan yüksek düzeyde bir dildir.

<http://octave.sourceforge.net> adresinden indirilebilir.

NNSYSID araç kutusu: Doğrusal olmayan dinamik sistemlerin sinirsel ağ tabanlı tanıtması için bir Matlab araç kutusudur. Bu tool da çok tabanlı perceptronların eğitim ve değerlendirilmesi için M fileler ve Mex filer birleştirilmiştir. Bu araç kutusunun bir başka özelliğide Sinirsel ağ ve sistem tanıtma araç kutularından tamamen bağımsızdır.

Matlab için Som araç kutusu: Som da Matlab programı için ücretsiz bir pakettir. Som araç kutusu Helsinki Üniversitesinde oluşturulmuştur. Araç kutusunda; ana özellikler, ana kullanımlar, görselleştirme, veri analizi, bu araç kutusunda rekabetçi ve işbirlikçi öğrenme mantıklarının kullanıldığı gözükmemektedir.

Rekabetçi öğrenme: Bir veri vektörüne en benzeyen prototip vektör düzenlenir, böylece ona daha çok benzer hale gelir.

İşbirlikçi öğrenme: Sadece en benzer prototip vektör değil aynı zamanda onun eşlem üzerindeki komşuları da veri vektörüne doğru hareket eder.

Bulanık üyelik optimizasyonu için M-files dosyaları: Bu program Dan Simon tarafından geliştirilmiştir. Program çeşitli M fileslarını kullanılır hale getirir. İnişi eğimi ve kalmna filtrelerini kullanarak bulanık mantık üyelik fonksiyonunu optimizasyonunu gösteririler. Bu M – Files larla ilgili görev- otomatik hız kontrol sistemidir. M-fileslar sıkıştırılmış haldeki dosyalar sayesinde indirilebilir siteden.

Vhicle control: Bulanık bir araç hız kontrol programıdır.

Fuzzcalc. M: Bulanıklığı giderilmiş çıktıların karşılığı olan bulanık girdileri hesaplayan genel amaçlı bir bulanık mantık motoru. Bu dosya aynı zamanda çıktıların türevlerini üyelik fonksiyonu parametrelerine uygun olarak hesaplayabilir. Bu yordam, üçgen üyelik fonksiyonlarını kullanır.

FuzzInit. M: Bir bulanık mantık sistemi için kural tabanı ve üyelik fonksiyonlarını başlatır.

Gradecalc. M: Üçgen bir bulanık kümedeki bir sayının üyelik derecesini hesaplayan genel amaçlı bir yordamdır.

PlotMem. M: Dosyada depolanmış üyelik parametreleri temelinde üçgen üyelik fonksiyonlarını çizer.

VehicleGrad. M: Bulanık bir hız kontrol sistemini iniş eğimi kullanarak optimize eder.

VehicleKalman. M: Bulanık bir hız kontrol sistemini Kalman filtresi kullanarak optimize eder.

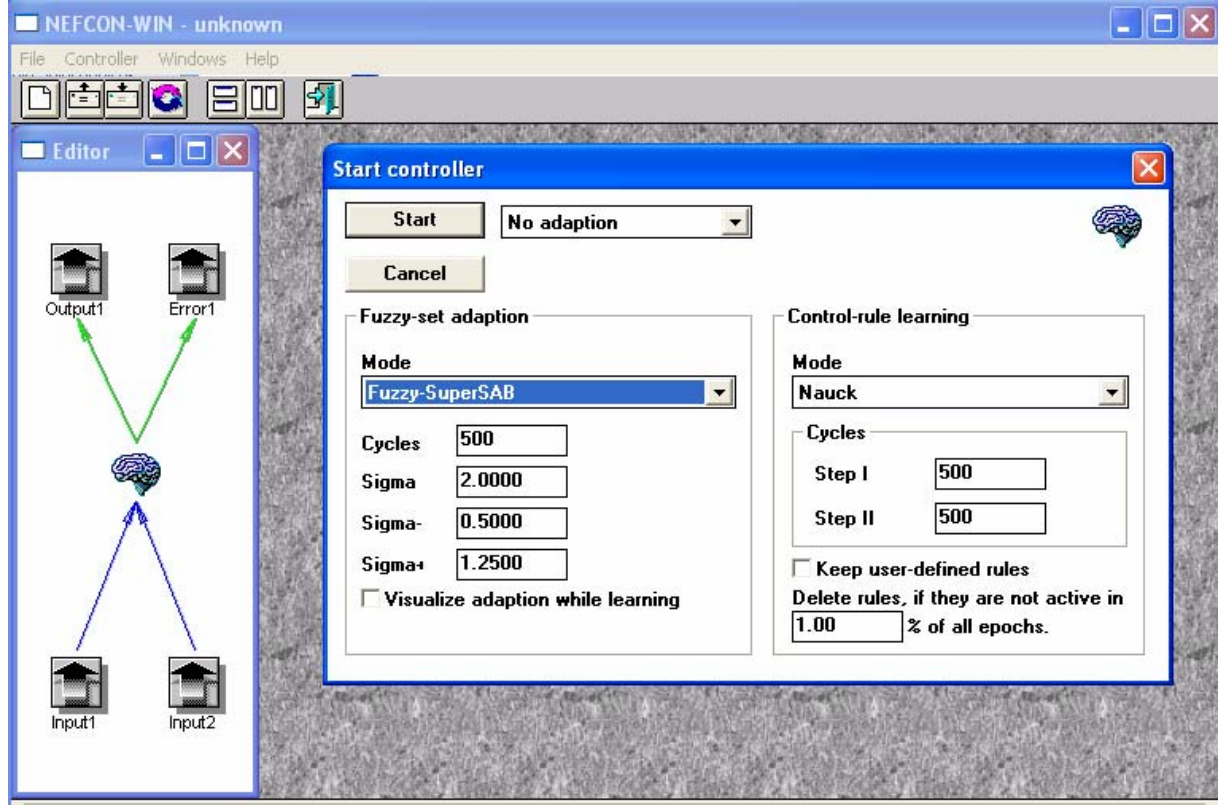
Dmatrix. M: Optimizasyon bazı normal kısıtlamalara gerçekleştirildiğinde kullanılan bir matris yaratan yardımcı bir yordamdır.

Engineering Applications of Soft Computing: Departman of Proccess Engineering, Folyamatmernoki Tazszek tarafından yapılmıştır. Yine diğerlerinde olduğu gibi çeşitli M filesleri kullanmaktadır ve tanımlanmaktadır.

Type-2 Fuzzy Logic Software: Niles N. Karnik tarafından sağlanan ücretsiz M files koleksiyonu. Koleksiyon dört bölümden oluşmaktadır; genel tip -2 bulanık mantık sistemleri, aralık tip-2 bulanık mantık sistemleri, tip-1 bulanık mantık sistemleri ve new tip-indirgeme.

Matlab için NEFCON: NEFCON, bulanık bir perceptron kullanan sinirsel bulanık kontrolcü tabanlı bir sinirsel bulanık kontrolcü tamamlaması. Matlab / Simulink ortamı gerektiriyor. NEFCON, öğrenme algoritması tavsiyesiyle bulanık

kümeleri ve kuralları öğrenebiliyor. Model, Braunschweig Teknik Üniversitesi Bulanık sistemler Araştırma grubu tarafından geliştirilmiştir. Yazılım ticari olmayan kurumlar için ücretsizdir. Şekil 3.5’de Nefcon - WIN yazılımının ekran görüntüsü verilmektedir.



Şekil 3.5 Nefcon - WIN Arayüzü Ekranı

Matlab Software Tool for neuro-Fuzzy Identification and data analysis: Bu yazılım gianluca bontempi ve Mauro Bitattiri tarafından geliştirildi. Yazılım, bir n eğitim kümesinin çıktı girdi örneği temelinde bulanık bir mimariyi eğitir. Bu yazılım ücretsiz olarak kullanılabilir.

3.1.2.21 Motorola fuzzy Logic Engine for all series

[http://www.programmersheaven.com/zone 5/cat26/1288.htm](http://www.programmersheaven.com/zone%205/cat26/1288.htm),
<http://faculty.petra.ac.id/resmana/private/fuzzy>

Bu yazılımla ilgili her hangibi bir bilgiye ayrıntılı olarak ulaşamadık; Ancak aşağıdaki web adresinden bazı firma bilgilerine ulaşılabilir.

3.1.2.22 Neurofuzzy classification, (Nefclass)

<http://fuzzy.cs.uni-magdeburg.de/nefclass/nefclass.html>

Nefclass, Detlef Nauck tarafından sağlanan bir yazılım. Bilimsel ve kişisel kullanım için ücretsiz. Yazılım sinirsel bulanık modellere veri analizinde kullanılması planlanmış. Denetimli öğrenmeyle bulanık kümeleri ve bulanık kuralları öğrenebilir. Bulanık bir sınıflandırma sistemini, bulanık sınıflandırma kurallarını artımlı bir şekilde öğrenen ve basit buluşsal yöntemler kullanarak bulanık kümeleri öğrenen şeklinde gösterebilir.

3.1.2.23 Neural networkrs and fuzzy Systems Software

<http://fuzzy.cs.uni-magdeburg.de/software.html>

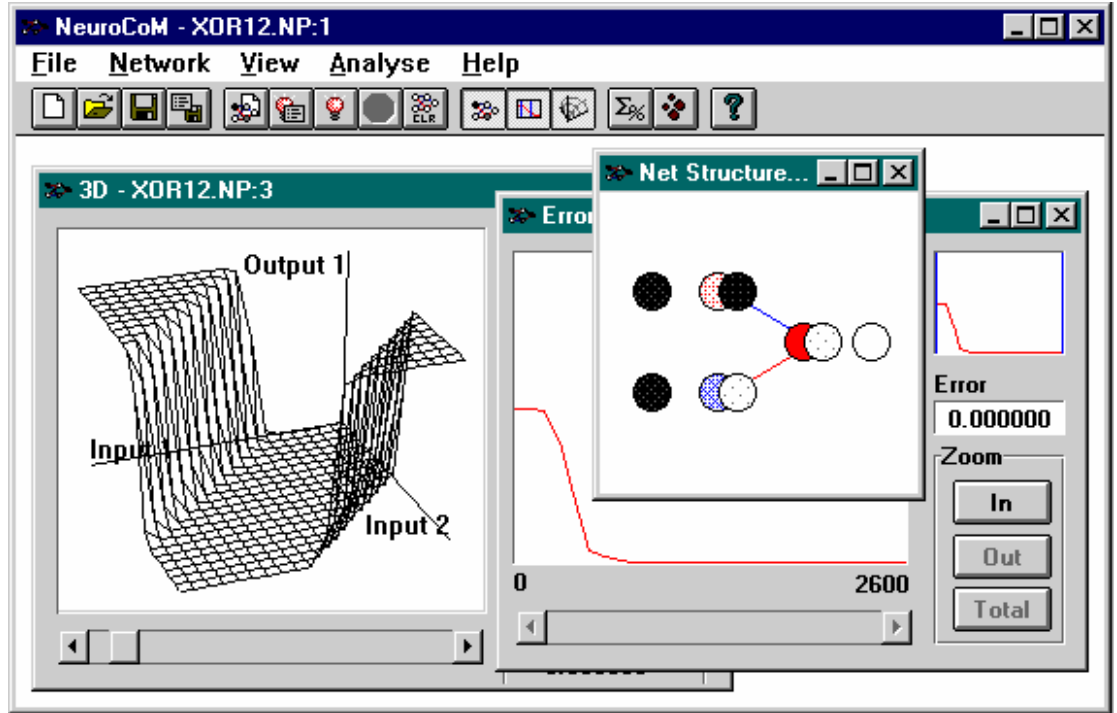
Research group on neural Networks and Fuzzy Systems at the institute of knowledge Processing and Language Engineering, Faculty of Computer Science, University of Magdeburg, tarafından geliştirilen çeşitli yazılım programlarına indirmek için alanlar sağlıyor. Sinirsel bulanık kontrol, sinirsel bulanık fonksiyon

yaklaşımı, bulanık kümeleme, Sinirsel bulanık veri analizi, çok tabakalı perceptron eğitim görselleştirilmesi, öğrenme vektörü miktarı görselleştirilmesi ve kendinden organize eğitim görselleştirilmesi, XLMP adındadır. Yazılım basit bir perceptronun iki mantıksal ve iki basit gerçek değer fonksiyonları için eğitim işleminin görselleştirilmesine yardım eder.

3.1.2.24 Neuro Designer

<http://www.neurodesigner.com>

Neuro Designer, sinirsel ağ uygulamaları için kapsamlı bir java tabanlı program ailesi. Demo versionu [www. NeuroDesigner.com](http://www.NeuroDesigner.com) adresinden temin edilebilir.Demo versiyonu sınırlı bir şekildedir. Örneğin; ağ 10 girdi ve 25 çıktı sinirden veya 4 gizli sinire sahip bir gizli tabakadan fazla olamaz.yazılımın önemli yetenekleri arasında; varolan sistemlere sinirsel ağlar gömmek ve çeşitli kaynaklardan veri getirmek sayılabilir. Şekil 3.6’da NeuroCom’un ekran görüntüsü verilmektedir.



Şekil 3.6 NeuroCom Arayüz Ekranı

3.1.2.25 Neuro Solitions

<http://www.nd.com>

Neuro Solitions, neuro dimension tarafından geliştirilen bir grafiksel sinirsel ağ geliştirme aracı. Web sistesinde ücretsiz deneme versiyonuna ve ekran görüntüleriyle yazılımın tamamına ulaşabiliriz. Bu yazılım çeşitli ağ mimarilerini destekler;

- CANFIS Network (SOM)
- Çok tabakalı perceptron (MLP)
- Kendinden orfganize eşlem ağı (SOM)
- Genelleştirilmiş ileri belsem ağı
- Genelleştirilmiş yinelenen ağ

Öğrenen vektör miktarı ölçümü(LVQ)

Yazılımda kullanıcı kendi isteği ile 2 adet sinirsel ağ üretebiliyor bu sinirsel ağları iki adet farklı sihirbaz ile üretebiliyor. Bunlar;

Neural Expert: tasarımın özelliklerinin sınıflandırma, Öngörü, fonksiyon yaklaşımı veya kümeleme gibi kullanıcının ilgilendiği probleminin tipinde toplandığı durum

Neural Builder: tasarım özelliklerinin kullanıcının inşa etmek istediği spesifik sinirsel ağ mimarisi üzerinde toplandığı durum.

Yazılım girdiler/çıktılar, hatalar, gradyanlar, ağırlıklar v.s. bütün iç ağ parametrelerine ve veriye gerçek zamanlı geçiş sağlayan araştırmalara izin vermekte

Aynı zamanda uygulamacıya yazılım, öğrenme hızları, ağırlıkları, gizli sinir sayıları gibi özellikleri optimize etme olanağı sağlamakta. Neuro Solutions profesyonel ve geliştirici basamakları tasarlanan sinirsel ağ için otomatik C++ kaynak kodları üremektedir.

3.1.2.26 NICO Artifical Neural Network Toolkit

<http://www.speech.kth.se/NICO/index.html>

NICO(Neural Inference Computation - Sinirsel çıkarımlı hesaplama), araç kutusu Nikko Storm tarafından geliştirildi. Yapay sinir ağlarının yapımı ve geri yayımlı öğrenme algoritması ile eğitim için genel amaçlı bir araç kutusudur. Sinirler, gruplara organize edilir ve grup hiyerarşik bir yapıdır; gruplar alt gruplara veya diğer nesnelere üye olarak sahip olabilirler. Bu, gelişigüzel bağlantı yapısına sahip çok tabakalı ağların belirlenmesi ve modüler ağların inşası için kolaylaştırır. Ağların grafiksel olarak gösterilmesi yazılımda desteklenmemektedir.

3.1.2.27 Rigel corporation

<http://www.rigelcorp.com/8051soft.htm>

Bu firmaya ait olan yazılımda; entegre uygulama yazılım geliştirme sistemi olan Reads51 içermektedir. Bu; MCS-51 dilinde yazmaya, derlemeye, birleştirmeye, hata gidermeye, indirmeye ve uygulamaların çalıştırılmasına olanak tanır. Menüler çalıştırılan ortamda, C- derleyicisi, göreceli çevirici, editör, çip simülatörü, bağlayıcı/yerleştirici, dil çeviricisi hata ayıklayıcı, ana sistem-kart sistemleri içerir. 8051 ailesi mikrokontrolcüler kullanan bulanık mantık uygulamaları için bir paket olan Fuzzy logic LabPac hakkında çalışmalarda yapılabilmektedir.

3.1.2.28 Stuttgart Neural Network Simulator, (SNNS)

<http://www-ra.informatik.uni-tuebingen.de/SNNS>

IPVR University of Stuttgart tarafından geliştirilmiş Unix iş istasyonlarında sinirsel ağlar, ç,n yazılım simülatörü. Başlıca iki bileşeni vardır: C dilinde yazılmış bir çekirdek simülatörü ve bir grafik kullanıcı ara yüzü.

Çekirdek simülatörü, sinirsel ağların iç veri yapıları üzerinde çalışır ve tüm öğrenme ve tekrar çağırma işlemlerini gerçekleştirir. Özel uygulamalarda gömülü C program: olarak diğer parçalar olmadan da kullanılabilir. Ağ mimarisi ve öğrenme prosedürleri; geri yayılım, sayıcı yayılım, genelleştirilmiş Radyal havuz fonksiyonları, ART1, ART2 ARTMAP ve daha fazlası tarafından desteklenir.

3.1.2.29 WinNN

<http://www.geocities.com/SiliconValley/Lab/9052/winnn.htm>

İleri beslemeli, çok tabanlı NN tamamlayabilir ve eğitim ve çeşitli sinir fonksiyonları için düzeltilmiş hızlı geri yazılım kullanır. Geniş online yardımı ve çeşitli sinir fonksiyonları vardır. Yazılım bazı örnek dosyalara da sahiptir. Demo sürümünü, yukarıda verilen web adresinden indirebilirsiniz.

3.1.2.30 XFuzzy

<http://www.imse.cnm.es/Xfuzzy/>

<http://sources.redhat.com/cygwin>

<http://sources.redhat.com/cygwin/xfree>

XFuzzy, bulanık mantık tabanlı sistemlerin tasarımı, onayı ve sentezi için bir ortam. Windows işletim sisteminde bu yazılım şu programlarla derlenebilir ve yürütülebilir; Cywin ve XFree86.

3.1.2.31 Fuzzy Sets and Systems Links

<http://www.abo.fi/~rfuller/fyzs.html>

Robert fuller tarafından hazırlanmış bir yazılım çeşidi. Personel home pages of Fuzzy Researchers, Who is Who in Fuzzy Database, fuzzy-mail archives, Fuzzy logic sources of information, fuzzy logic at Webopedia, Computational, Conferences and workshops on fuzzy Systems, Professional Organisations, Journals & Handbooks, Research Groups.

3.1.2.32 Bulanık Mantık Denetleyici Simülatörü

[http:// bmd.atlas.org](http://bmd.atlas.org)

Bulanık mantık teorisini ve farklı uygulama alanlarını anlatan oldukça fazla sayıda kitap ve makale yayınlanmış olmasına rağmen bulanık mantığın uygulanabilirliğiyle ilgili somut örneklere rastlamak ne yazık ki pek mümkün değildir. Buna bağlı olarak oluşturulan simülatör, arayüz ve animasyonların teorik bilginin daha iyi anlaşılmasındaki önemi ortaya koymaktadır ve teorik bilginin anlaşılmasını kolaylaştırmaktadır. Burada tanıtılmaya çalışılan bulanık mantık denetleyici simülatörü (BMDS) bir su tankındaki su seviyesini kontrol etmek için geliştirilmiş olup, bulanık mantık tabanlı denetleyicinin çalışma sırasındaki yapısal özelliklerinin ve gerçekleştirilen işlemlerin izlenmesine olanak vermektedir. Kullanıcı dostu bir arayüze sahip olan simülatör; üçgen, yamuk ve gaussian tipi üyelik fonksiyonlarını tercihe göre kullanarak su tankındaki su seviyesini ayarlamaktadır. Kullanıcı hem bulanık kurallarını hem de üyelik fonksiyonlarının tanım aralıklarını istediği gibi değiştirebilme şansına sahiptir. Bulanık denetleyicinin çalışma safhaları ve elemanları kullanıcıya sunulmakta, konunun görsel olarak daha kolay anlaşılmasını sağlamaktadır.

Hazırlanan bu simülatör gaussian bulanıklaştırma yönteminin olması yönünden avantajlı ancak Türkçe arayüz olmaması, çıkarım ve durulaştırma yöntemlerinin az olması yönünden dezavantajlara sahiptir.

3.2 Programların Avantaj Ve Dezavantaj Tablosu

Bu bölümde literatür taraması yapılarak programlar incelenmiştir. İncelenen bulanık yazılımların görselliği, kullanım dilleri, kullandıkları bulanıklaştırma yöntemleri, çıkarım yöntemleri, durulaştırma yöntemleri, bulanık ifadelerin yazılıma aktarılırken sınırlılıkları ve kolaylıkları, bulanık kuralların yazılıma girilirken sınırlılıkları ve kolaylıkları, hangi ortamda üretildikleri, hangi yazılım dilleriyle üretildikleri, ücretli yada ücretsiz oluşları ve kullanıcının bu arayüzleri nerelerden elde edip kullanabilecekleri gibi farklı noktalardan literatür taraması yapılmıştır.

Son olarakta incelenen literatürdeki bu yazılımların genel durumlarına ait ortak özellikleri ve ayrıcalıklı özellikleri Tablo 3.1’de ayrıntılı olarak verilmiştir.

Verilen bu tabloya göre yapılan incelemelerde şu ana kadar var olan bulanık arayüzlerin birçoğunda; Bulanıklaştırma yöntemlerinin sınırlı olması, çıkarım mekanizmalarının az sayıda olması, durulaştırma yöntemlerinin az sayıda olması, giriş ve çıkış parametrelerinin girişindeki sınırlılıklar olmasıdır.

Tablo 3.1 aşağıdaki gibi verilmiştir;

Tablo 3.1 Programların Avantaj ve Dezavantaj Tablosu

Program Adı	Üreten Firma Kişi	Yazılımı Oluşturulduğu Dil	Program Dili	Çalıştığı Ortam	Programın Amacı
The Art Gallery	Lars Liden	C ve C++ prosedürleri	İngilizce	Windows, dos, unix	Uygulama
Binary Hopfield Net	Matt Hill	java,cprosedürleri	İngilizce	Windows(görsel)	Kullanıcı
Neural Networks on your Fingertips	Karstan kutza	C ve C++ prosedürleri	İngilizce	Görsel	Uygulama
Pittnet	Brian Carnahan ve Alice Smith (Universty of Pitsburg)	C ++ Kaynaklı Kodlu	İngilizce	Görsel	Uygulama
UNFuzzy	Oscar Duarte (Universidad Nacional de colombia)	C ve C++ prosedürleri	İspanyolca-İngilizce-Fransızca vs..	Görsel	Uygulama
DOS NN freeware		C++ kaynak kodlu	İngilizce	MS-DOS	Uygulama
StarFLIP	Datebase and Expert system group Institute of Information system Vienna Universty of Technology	C++ Kütüphanesi start FLIP++Kütüphanesi	Almanca-İngilizce	Görsel	Uygulama
EasyNN	Stephan Walsten Holme	C Prosedürleri	İngilizce	Grafikle,Görsellik	Uygulama
M, Fuzzy Control Manager	-	C dilinde	-	Windows	Deneyim ve
E, Fuzzy Inference Development Environmet	Santa Clara USA Aptranix Comp. Den	Java,ANSI,C,NATLAB-M FILE	İngilizce	Windows(görsel)	Ücretli piy
DE, Fuzzy Logic Development Environment	Stylionas S. Sbyrakis (Syndesis LTD.Atina)	C kodu üretir	Yunanca - İngilizce	Görsel	Uygulama
Fuzzy CLIPS	NASA	C dili entegre üretimi	İngilizce	Görsel	Ticari Tica
Fuzzy Control Demo	West Virginia Universty(USA)	-	İngilizce	Görsel	Deneyim
FuzzyCOPE	Knowledge Engineering laboratory (kel) Universty of otago	DLL 'ler var	İngilizce	windows, Dos' a bağlantı yapılabiliyor	Uygulama

Fuzzy Decision Tree, FID	Cezary Janikow (mathematics and computer Science ,UMLS st. Lowis)	karar ağaçları	İngilizce	Görsel	
FuzzGen	(İlk bulanık kod üreticisi)	Pascal,Basic C/C++ dillerinde kod üretir	İngilizce	Görsel ve metinsel	
Fuzzy Logic Control in VHDL	Steve Dillen ve Farrah Rashid,Duncan Eliot (universty of Alberta,Canada)sundu	HTML formatında	İngilizce	Metinsel	
fuzzyTECH	Datebase and Expert system group Institute of Information system Vienna Universty of Technology	dLL 'ler var	İngilizce ve Almanca, dökümanları mandarin Çincesinde, almanca ve ingilizcede var.	Grafiksel	
HavBpETT	Neosoft(Hav Software)	HowBpett,HowBpett++, C++ DLL versiyon	İngilizce	Görsel(windows)	Deneyim
SuperLogic Demos	Cubicalc	Cubicalc(anayazılım)C, OWL kütüphaneleri	İngilizce	Windows+MS-DOS	Deneyim
Fuzzy System, JFL	Jan mortensen(Danimarka)	JFL prog. dili C ve DLL ile Harekete Geçirilebilir, Java script-HTML	Flemance, İngilizce	Görsel	
Related Joon NN	Joone NN	Java	-	Görsel	
LabVIEW	National Instrument (Viyand Tech.Uni.)	Java	-	Grafiksel,görsel	Üniversite ve kurumsal
Adaptive Fuzzy Logic Controller , LFLC	Çek cumhuriyeti lojik madde merkezi araştırma Ve uygulama merkezi	-	-	-	
Matlab	-	C,C++matlab	İngilizce	Görsel, metinsel grafiksel, komutlu	Deneyim
NEFCLASS, Neurofuzzy Classification	Nefclass-Devlet Nauck tarafından yazılmıştır	—	İngilizce	Görsel ve metinsel (komutlu)	Bilimsel kullanımlar
Neural Networks And Systems Software	Research group on neural Networks and Fuzzy system at the institu of knowledge processing and language engineering, faculty of computer science, Universtiy of Magdeburg	—	—	—	
NeuroDesigner	Neuro designer	Java tabanlı	İngilizce	Görsel	

Neuro Solutions	Neuro dimension tarafından	C ++ Kaynaklı Kodlu	İngilizce	Görsel, grafiksel	Ücretsiz
ICO Artificial Neural Network Toolkit	Nikko storm	--	--	Grafiksel, metinsel, destek yok	
Figel Corporation	Figel corporation	C derleyicisi	İngilizce	Görsel	Ücretli
Stuttgart Neural Network Simulator (SNNs)	IPVR University of Stutugard	C dilinde yazılmış	Almanca-İngilizce	Grafik kullanıcı arayüzü, Unix tabanlı	
WinNN	Sillicon Valley Labrotory	--	İngilizce	--	Ücretli
Xfuzzy	Cywin Corp.	--	--	Windows	
Link Set and System Links	Robert Fuller tarafından	--	İngilizce	Görsel	
Transfer Tech	Transfer Tech, Neurocom	C, C++ kaynak kodlu	Almanca-İngilizce	Görsel	
Adwin	Adwin Corp.	C++, Matlab M-Files	Japonca	Görsel	Ücretli
Win Version 1.0 - 32 bit	Dr. Detlef Nauck (Tecnical university of Braunscevig), Matthias webber	C, C++ kaynak kodlu prosedürleri kullanıyor	Almanca-İngilizce	Windows (görsel)	Ücretli
Yapay Zeka Mantık Kontrolleyici Simülatörü	İsmail H. ALTAŞ Karadeniz Teknik Üniversitesi	Görsel Arayüz, Java tabanlı	İngilizce	Windows (görsel)	Ücretsiz

4. TASARLANAN ARAYÜZ ve KULLANIMI

4.1 Tasarlanacak Olan Arayüze Konulan Talepler

Görsel olması

Bulanıklaştırma yöntemlerinin çok sayıda olması ve seçilebilmesi;

Çıkarım yöntemlerinin çok sayıda olması ve seçilebilmesi;

Durulaştırma yöntemlerinin çok sayıda olması ve seçilebilmesi;

Bulanık kuralların daha rahat girilmesi;

Ekran kullanımının kolaylığı.

Bu talepleri karşılayan program için Borland Delphi Enterprise 7.0 görsel programlama dili seçilmiştir. Delphi programlama dili daha görsel, daha esnek tabanlı olması, grafik hazırlama için yazılımda özel kodlarının olması ve veri tabanı bağlantılarında daha hızlı ve etkin çalışabilmesi açısından güçlü bir görsel yazılımdır.

Bulanık mantık problemleri için Türkçe görsel bir arayüz tasarım programını Delphi 7.0 Enterprise programlama dilinde yazılmış olup, programlama yapılırken verilere ve sonuçlara daha hızlı ulaşmak için bir Delphi komponenti olan kbmtable (kbm memory table) kullanılmıştır. Ayrıca veri tabanı olarak Paradox kullanılmıştır ve verilerin sorgulanması SQL kodlama ile yapılmıştır. Programın kurulum arayüzünün hazırlanmasında ve düzenlenmesinde yine Delphi 7.0 Enterprise programı içerisinde paket olarak bulunan installshield wizard programı kullanılmıştır.

4.2 Kullanıcı Program Giriş Ekranı

Arayüzün ayrıntılı görüntüleri ve anlatımları aşağıdaki şekillerdeki gibidir. Şekil 4.1’de kullanıcının programa girmesi için gerekli olan kullanıcı adı ve kullanıcı şifresini girmesi gerekli olan ekranı göstermektedir. Bu ekranda hatalı bir kullanıcı adı veya hatalı bir kullanıcı şifresi girilirse program kullanıcıya hata mesajı vermektedir.

Şekil 4.1 Kullanıcı giriş ekranı

Şekil 4.1’de SQL Query ile Paradoxdaki veri tabanı sorgulanmıştır. Daha önceden yazılımcı tarafından girilen kullanıcı hesaplar kullanıcı adı ve kullanıcı şifresi doğrulandıktan sonra kullanıcı programa giriş yapabilmektedir. Eğer ki yazılan kullanıcı adı yanlışsa “Kullanıcı bulunamadı” ya da kullanıcı adı doğru olup da kullanıcı şifresi hatalı ise “Kullanıcı şifresi hatalı” olarak bir hata mesajı arayüz tarafından kullanıcıya iletilmiştir. Eğer kullanıcı adı ve kullanıcı şifresi doğruysa kullanıcı tamam butonuna tıklayarak programa giriş yapabilmektedir. Ancak kullanıcı programa girmek istemiyorsa vazgeç butonuna tıklayarak programdan çıkış yapabilmektedir.

4.3 Arayüz Ana Ekranı

Kullanıcı, kullanıcı adını ve kullanıcı şifresini tam olarak doğru bir şekilde girip tamam butonuna tıkladıktan sonra ana ekrana giriş yapabilmektedir. Ana ekran Şekil 4.2’de ayrıntılı olarak gösterilmektedir.

Şekil 4.2 Program parametrelerini belirleme ekranı (Form1)

Şekil 4.2’deki ekranda kullanıcı giriş parametresinin değerini girip (max 10 tane) giriş parametreleri belirleyebilmekte, çıkış parametresinin değerini girip (max 10 tane) çıkış parametreleri belirleyebilmektedir. Kullanıcı çıkarım mekanizmasını ve durulaştırma yöntemini de bu ekranda seçebilmektedir. Kullanıcı bu işlemleri şu şekilde yapabilir.

Kullanıcı Şekil 4.2’de bulunan editboxlara giriş ve çıkış parametrelerinin sayısı ne ise bu değerleri belirleyip arayüze girebilmektedir. Giriş ve çıkış

parametrelerinin deęerleri girildikten sonra “OLUŐTUR” butonuna tıklayarak giriő ve ıkıő parametrelerinin her birinin programa girilmesi iin birer alan oluőturulabilmektedir. Bu aılan alanlara giriő ve ıkıő parametrelerinin adları ve deęerleri girebilmektedir. Kullanıcı bu deęerler girildięinde Őekil 4.3’de ki grnty elde ediyor. Ayrıca bu ekranda yedi farklı yntemden oluőan ıkarım mekanizması seimi ve on farklı yntemden oluőan durulaőtırma yntemlerini belirleyip karőtısına gelecek olan yeni ekranda daha nceden belirledięi giriő ve ıkıő parametrelerini ve bu parametrelerinin ayrıntılarını girebilmektedir. Kullanıcı Őekil 4.4’de ki ekranı amak iin yani parametrelerin ayrıntılarını ve deęerlerini girmek iin her bir parametrenin adı zerine farenin imleci ile gelip ift tıklaması gerekiyor.

Őekil 4.3 Parametre belirleme rnek ekranı

4.4 Dilsel İfadeleri ve Dilsel İfadelerin Aralıklarını Belirleme Ekranı

Kullanıcı Şekil 4.4’de ana ekrandaki belirlediğimiz giriş ya da çıkış parametrelerinin dilsel ifadelerini ve bu dilsel ifadelere ait aralıkları belirleyebilmektedir. Bu aralıklar ve dilsel ifadeleri girerken program bize hiçbir sınır tanımamaktadır. Şekil 4.4’ün üst kısmındaki parametre adını, kullanıcının ana ekrandan girdiği parametrelerin adlarından otomatik olarak almaktadır. Dbgrid’deki parametrelerde dilsel ifade, ifadeye ait değerler, program tarafından otomatik olarak eklenen durum değişkeni ve bu girilen değerlerin programın başında belirlenen hangi giriş ya da çıkış ifadesinden olduğunu belirleyen değer ifadesi vardır. Bu ekranda ki dilsel ifade ve değerleri tam olarak dbgrid’e girdikten sonra “Grafik ekranı” butonuna tıklayarak grafik ekranına ulaşmaktadır. Grafik ekranı alanına geçtiğinde Şekil 4.5’deki görüntüyü elde etmektedir.

ifade	min	tepe1	tepe2	max	DURUM
Az	0	0	0	4	4 GİRİŞ
Çok	2	4	4	8	8 GİRİŞ
Çokçok	6	12	50	50	50 GİRİŞ

Grafik Ekranı

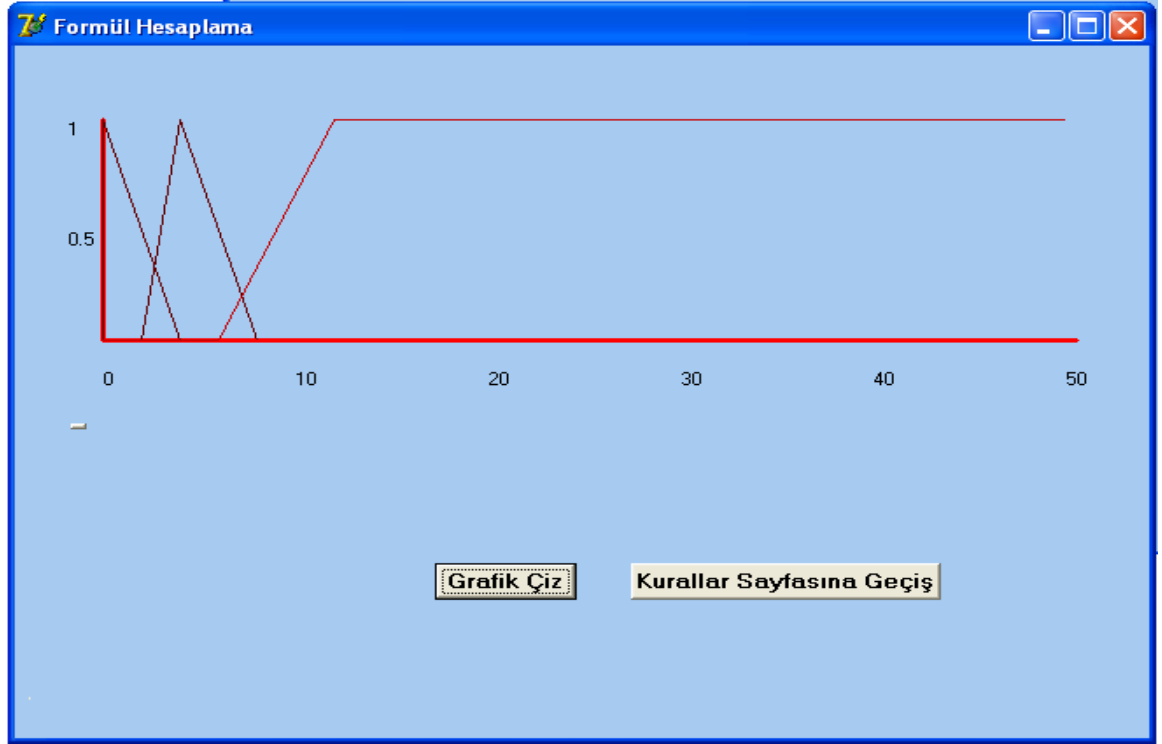
Şekil 4.4 Dilsel ifade ve aralık giriş ekranı

4.5 Bulanıklaştırma ve Grafik Oluşturma Ekranı

Kullanıcı Şekil 4.4'deki ekran görüntüsünde olan grafik ekranı butonuna tıkladığında Şekil 4.5'deki ekran karşısına gelmektedir. Açılan Şekil 4.5 ekranındaki “Grafik Çiz” butonuna tıkladığında Şekil 4.4'deki belirlenen parametrelere göre grafik oluşturulmaktadır. Grafik alanı dikey ekseninde $[0,1]$ arasında yatay ekseninde kullanıcının Şekil 4.4'de girmiş olduğu en son parametrenin değerine göre oran alır ve grafiği gerek trapezoidal (yamuk) gerekse de triangular (üçgen) olarak arayüz tarafından çizim yapılır. Aynı zaman da ekran penceresi tam ekran konumuna alındığında grafik alanı, ekranı tam ekran moduna göre arayüz ekranını oranlayıp yine aynı şekilde düzgün bir şekilde grafiği oluşturmaktadır yada güncellemektedir.

Bulanıklaştırma işlemi kullanıcının Şekil 4.4'deki belirlediği parametrelere göre arayüzün üçgensel yada trapezoidal olarak yapılmaktadır. Grafiklerde bu yöntemle göre otomatik olarak Şekil 4.5'de kullanıcının “Grafik Çiz” butonuna tıklamasıyla çizilmektedir.

Kullanıcı her bir giriş ve çıkış parametresi için aynı işlemi tekrarlamalıdır. Ekranları değiştirdikçe girdiğimiz giriş, çıkış verileri, bu verilere ait parametrelerini ve grafik sonuçlarını program bir memory tableda program sonlandırılana kadar tutmaktadır.



Şekil 4.5 Formül hesaplama ve grafik çizim ekranı

4.6 Bulanık Kuralları Belirleme Ekranı

“Kurallar sayfasına geçiş” butonuna tıkladığımızda Şekil 4.6 ve Şekil 4.6’nın devamı olan Şekil 4.7’deki ekranları karşımıza çıkarmaktadır. Burada her bir giriş, çıkış değişkeni için dilsel ifadelerin girileceği yer ve kıyaslamaları oluşturulmuştur.

Kural no, kural numaralarının; değer1, değer2 ..., giriş değerlerinin dilsel ifadelerinin, kiy1, kiy2 ..., kıyaslama alanlarının; cik1, cik2 ... ise çıkış değerlerinin dilsel ifadelerinin kurallar alanına girildikleri alanlardır.

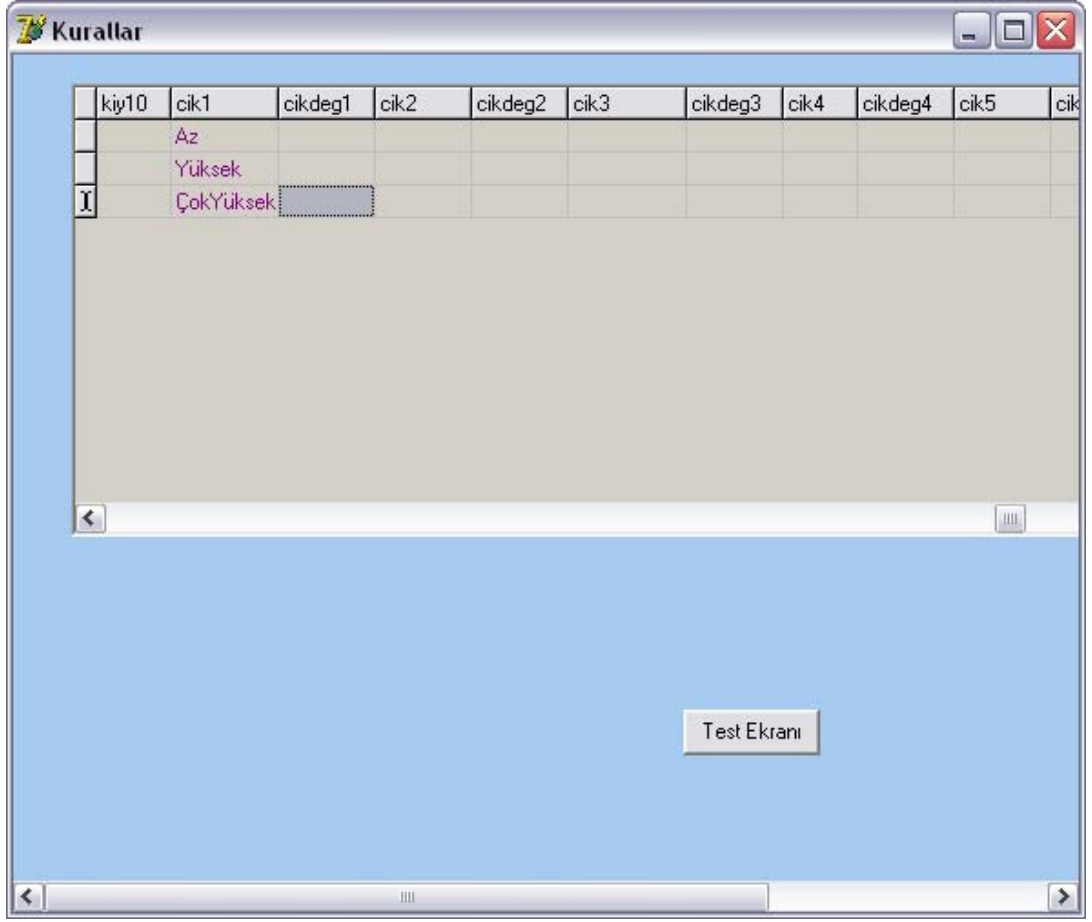
Bu ekrandaki kurallarda girildikten sonra kullanıcı “Test ekranı” yazan butona tıklayıp programa ait testlerin yapıldığı alana geçiş yapabilmektedir.

Kurallar

	kuralno	deg1	kiy1	deg2	kiy2	deg3	kiy3	deg4	kiy4	deg5	kiy5
	1	Az	ve	Orta							
	2	Çok	ve	Orta							
*	3	Çokçok	ve	Orta							

Test Ekranı

Şekil 4.6 Giriş değerleri için kuralları belirleme ekranı



Şekil 4.7 Çıkış değerleri için kuralları belirleme ekranı

Şekil 4.6'daki ekranda kıyaslama alanları için (kiy1, kiy2, ...) ve, veya gibi lojiksel ifadeleri kullanıcı arayüze giriş yapabilmektedir.

4.7 Test Ekranı

The screenshot shows a window titled 'Test' with a light blue background. At the top left, there is a small icon and the title 'Test'. Below this, there are two input fields labeled 'PSA' and 'SD'. The 'PSA' field contains the value '2' and the 'SD' field contains the value '5'. Below these fields, there are two lines of text. The first line says 'Çıkarım mekanizması olarak Larsen Max-Dot Yöntemi seçtiniz.' and the second line says 'Durulaştırma yöntemi olarak Ağırlık merkezci durulaştırıcı (Centroid) seçtiniz.' Below these lines, there is a button labeled 'SONUÇ'. At the bottom of the window, there is a line of text that says 'SL Miktarı 3,451 olarak bulunmuştur'.

Şekil 4.8 Sonuçları test etme ekranı

Test ekranı Şekil 4.8’de gösterildiği gibidir. Burada programın açılış ekranı olan ve Şekil 4.3’de gösterilen ekranda gösterilen ekrandaki girilen giriş değerlerini, çıkarım mekanizmasını ve durulaştırma yöntemini Şekil 4.8’de gösterilen alanlara sistem kendisi aktarmaktadır. Kullanıcı giriş değerlerini girdikten sonra “SONUÇ” butonuna tıkladığında; Bulanık arayüzün bize verdiği sonuçları aşağıdaki sonuç alanında kullanıcı görebilmektedir. Kullanıcı Şekil 4.8’deki alanları yenilemeye ihtiyaç duymadan tekrar girilen yeni giriş değerlerinin de sistem sonuçları kullanıcının sonuç butonuna tıklandıktan sonra karşısına çıkmaktadır.

5. TASARLANMIŞ ARAYÜZÜN TEST EDİLMESİ

Bu arayüzün test edilmesi için daha önceden tasarlanmış ve sonuçlandırılmış bulanık kontrol sistemlerinin sonuçları kullanılmıştır. Arayüzün test edilmesi için gereken veriler [SARITAŞ, 2003; Allahverdi, 2007]'den alınmıştır.

5.1 Kronik Bağırsak Rahatsızlığında Kullanılan Salazopryne İlaç Miktarı için Tasarlanan Bulanık Uzman Sistemin Test Edilmesi

Bu çalışmalardan ilkinde Kronik bağırsak rahatsızlığında kullanılan salazopryne ilaç miktarı için bir bulanık uzman sistem tasarımı yapılmıştır. Çalışma özet olarak şöyledir;

Tıpta bilinen hastalıkların tümü için birçok ilaç firmaları tarafından ilaçlar üretilmiştir. Üretilen tüm ilaçlar için tanıtım broşürleri ve kullanım talimatları hem ilaç kutuları içersine konulmuş hem de doktorlara bilgi verilerek eğitilmişlerdir. Yeni üretilen ilaçların da kullanım alanı ya da alanları hakkında doktorlar bilgilendirilmektedir.

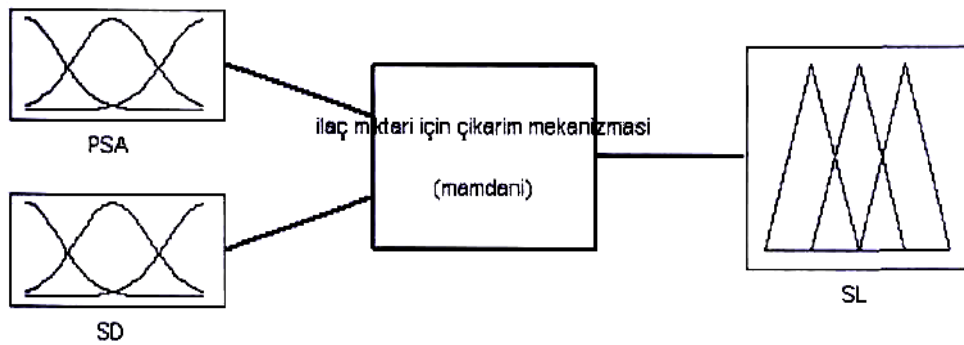
Doktorlar tarafından hastaya tedavi için verilen ilaç miktarı genelde kullanım talimatına göre verilmektedir. Ancak ilaçların birçoğunun miktarı, kan ölçümlerinin sonuçları göstermektedir. Uzman bir doktor deneyimlerinden hastasına günlük hangi miktarda ilaç alması gerektiğini söylemektedir. Bu ise çoğu zaman tam kesin miktar olamamaktadır. Bu bir belirsizlik içermektedir. Kesin miktarı belirlemede, deneyimli uzman doktor bilgilerinden faydalanılarak zeki bir bilgisayar yazılımı faydalı olabilmektedir. Bu sebeple doktorların hastaya alması gereken ilaç

miktarını tam olarak belirtmesinde yardımcı olacak kronik bağırsak rahatsızlığı tedavisinde kullanılan Salazopyrine ilaç miktarı için örnek bir BUS geliştirilmiştir.

Bu hastalık (kronik bağırsak rahatsızlığı) belirteçlerinden; kanda prostat kanser belirteci (Prostate Spesific Antijen - PSA (x)) ile kandaki hücrelerin çökme hızı (Sedimentasyon - SD (y)) durumuna göre uygulanacak tedavide verilecek olan ilacın (Salazopyrine - SL (z)) miktarları dilsel olarak Tablo 5.2'de belirtilmiştir. Ayrıca Şekil 5.1'de geliştirilen BUS genel çalışma şeması verilmiştir.

Tablo 5.1 Salazopyrine Miktarı için Dilsel İfade Tablosu

		PSA (x)		
		Düşük	Orta	Yüksek
SD(y)	Düşük	Az ver	Az ver	Çok ver
	Orta	Orta ver	Orta ver	Çok ver
	Yüksek	Orta ver	Çok ver	Çok çok ver



Şekil 5.1 Geliştirilen BUS Genel Şeması

Bu çalışmanın sonuçları şu şekildedir;

Değişik Giriş Değerleri ile Bulanık Uzman Sistemde elde edilen SL miktarları Tablo 5.2’de ayrıntılı olarak gösterilmiştir. Yapılmış olan bu çalışmada vardığı sonuç olarak kullanılan durulaştırma yöntemlerinden en iyi sonuç vereninin *centroid* yöntemi olduğu kanaatine varılmıştır (Sarıtaş, 2003).

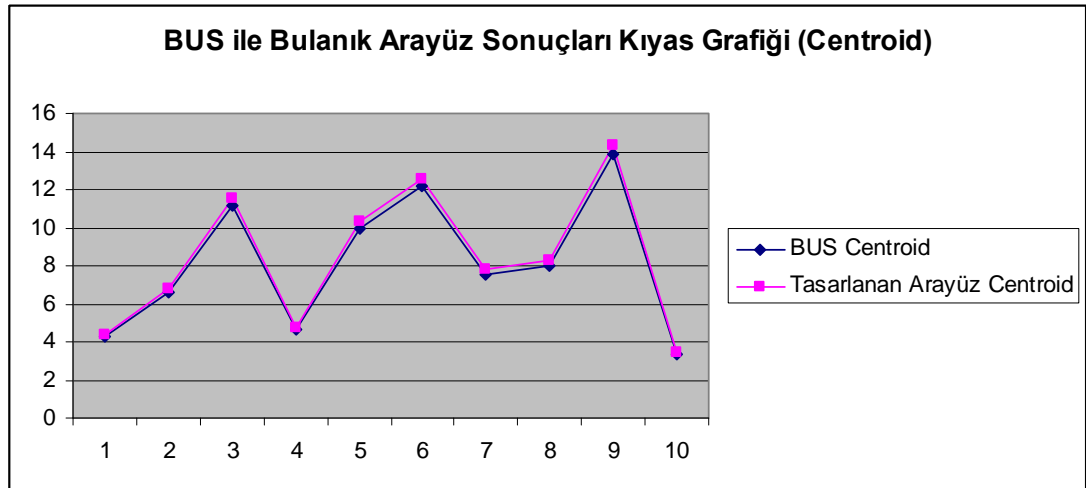
Tablo 5.2 Kronik bağırsak rahatsızlığında kullanılan salazopryne ilaç miktarı için bir bulanık uzman sistem tasarımı Matlab programı sonuçları

Sıra No	GİRİŞ DEĞİŞKENLERİ		ÇIKIŞ DEĞİŞKENİ (SL ng)				
	PSA ng/ml	SDms	Bulanık Uzman Sistem (BUS)				
			<i>Centroid</i>	<i>Bisector</i>	<i>Mom</i>	<i>Som</i>	<i>Lom</i>
1	1	20	4,27	4	1,84	3,68	0
2	3	50	6,61	6,24	6	6,88	5,12
3	7	80	11,2	12,5	14,6	16	13,1
4	2	25	4,61	4,64	6	8	4
5	8	25	10	9,92	10	11,7	8,32
6	8	75	12,2	12,6	14,4	16	12,8
7	5	50	7,55	6,88	6	6,88	5,12
8	2	70	8	8	8	12	4
9	15	90	13,9	14,1	14,1	14,1	14,1
10	0,4	10	3,34	2,72	1,44	2,88	0

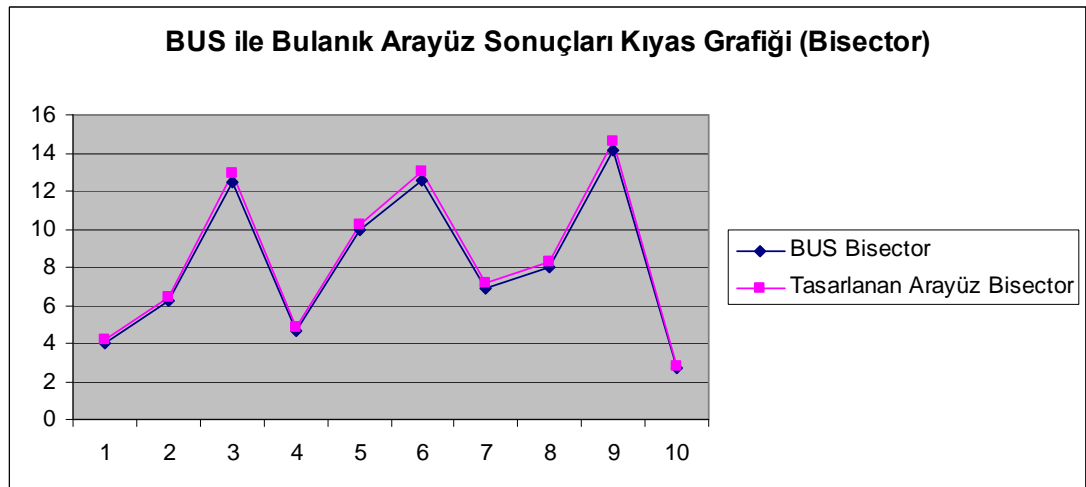
Kronik bağırsak rahatsızlığında kullanılan salazopryne ilaç miktarı için yapılan test sonuçlarını test etmek için Tablo 5.2’deki sonuçları oluşturduğumuz arayüzdede denediğimizde kaşımıza sonuç olarak Tablo 5.3’deki sonuçlar çıkmıştır.

**Tablo 5.3 Kronik Bağırsak Rahatsızlığında Kullanılan Salazopryne İlaç
Miktarı için Oluşturulan Arayüzün ve Bulanık Kontrol Arayüzün Test
Sonuçları**

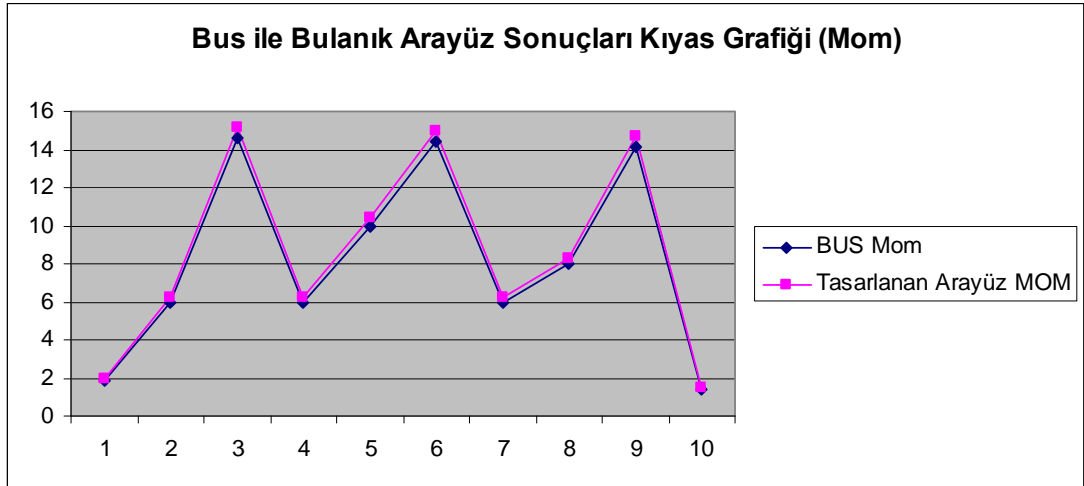
Sıra No	Giriş değişkenleri		ÇIKIŞ DEĞİŞKENİ (SL ng)					ÇIKIŞ DEĞİŞKENİ (SL ng)					
	PSA ng/ml	SDms	Bulanık Uzman Sistem (BUS) Matlab Sonuçları					Tasarlanan Bulanık Kontrol					
			<i>Centroid</i>	<i>Bisector</i>	<i>Mom</i>	<i>Som</i>	<i>Lom</i>	<i>Centroid</i>	<i>Bisector</i>	<i>Maksimum durulaştırıcı (MOM)</i>	<i>En büyük üyelik ilkesi (SOM)</i>	<i>Ağırlıklı ortalama yöntemi</i>	<i>...</i>
1	1	20	4,27	4	1,84	3,68	0	4,40	4,14	1,91	3,75	4,48	
2	3	50	6,61	6,24	6	6,88	5,12	6,81	6,46	6,24	7,02	6,94	
3	7	80	11,2	12,5	14,6	16	13,1	11,54	12,94	15,18	16,32	11,76	
4	2	25	4,61	4,64	6	8	4	4,75	4,80	6,24	8,16	4,84	
5	8	25	10	9,92	10	11,7	8,32	10,30	10,27	10,40	11,93	10,50	
6	8	75	12,2	12,6	14,4	16	12,8	12,57	13,04	14,98	16,32	12,81	
7	5	50	7,55	6,88	6	6,88	5,12	7,78	7,12	6,24	7,02	7,93	
8	2	70	8	8	8	12	4	8,24	8,28	8,32	12,24	8,40	
9	15	90	13,9	14,1	14,1	14,1	14,1	14,32	14,59	14,66	14,38	14,60	
10	0,4	10	3,34	2,72	1,44	2,88	0	3,44	2,82	1,50	2,94	3,51	



Şekil 5.2 Salazopryne ilaç miktarı için tasarlanan Bus ile Bulanık Arayüz Sonuçları Centroid yöntemi kıyas grafiği



Şekil 5.3 Salazopryne ilaç miktarı için tasarlanan Bus ile Bulanık Arayüz Sonuçları Bisector yöntemi kıyas grafiği



Şekil 5.4 Salazopryne ilaç miktarı için tasarlanan Bus ile Bulanık Arayüz Sonuçları Mom yöntemi kıyas grafiği

Tablo 5.3’deki test sonuçlarına ve ayrıca Şekil 5.2, Şekil5.3 ve Şekil 5.4’deki grafiklere baktığımızda kronik bağırsak rahatsızlığında kullanılan salazopryne ilaç miktarı için oluşturulan BUS sonuçları ve BUS sonuçlarının oluşturulan Bulanık Kontrol Arayüz programında yapılan testlerinin sonucunda yapılan karşılaştırmalara göre en iyi ve sonuca en yakın durulaştırma yönteminin “centroid” yöntemi olduğu anlaşılmaktadır.

5.2 Prostat Kanseri Riski Tespiti için Tasarlanan Bulanık Uzman Sistemin Test Edilmesi

Bu çalışmaların ikincisinde; Prostat kanseri riski tespiti için bir bulanık uzman sistem tasarımı oluşturulmuştur. Uzman doktorlar rektal muayene, PSA taraması, Ultrasonografi ve biyopsi gibi taramalarla prostat kanserlerini teşhis ve tedavi edebilmektedirler.

Prostat kanserinin tanısında bilinen en iyi tümör belirleyicisi serum prostat spesifik antijenidir (PSA). Ancak, hastalık spesifik değil, organ spesifiktir. Bu nedenle özellikle yanlış pozitif sonuçlar nedeniyle pek çok hastadan gereksiz biyopsiler alınmaktadır. Serum PSA düzeyi 2-20 ng/ml arasında ve rektal muayenesi normal olan hastalarda PSA dansitesi, yaşa bağlı PSA ve serbest PSA oranı ile PSA'nın klinik yararlılığı ile biyopsi sayısını ne ölçüde azaltabileceği önemlidir. Bu nedenle deneyimli uzman bir hekimin bilgileri ile zeki bir bilgisayar yazılım oluşturarak hastanın daha erken kanser riski teşhisi, gereksiz biyopsiler ile bu biyopsilerin maliyeti ve komplikasyonlarından ek risk almadan korunabilmesi için doktorlara ve tıp eğitimi gören öğrencilerin uzmanlaşmasına yardımcı olacak bir bulanık uzman sistem geliştirilmiştir. Geliştirilen BUS'un çalışma şeması Şekil 5.5'de verilmiştir.

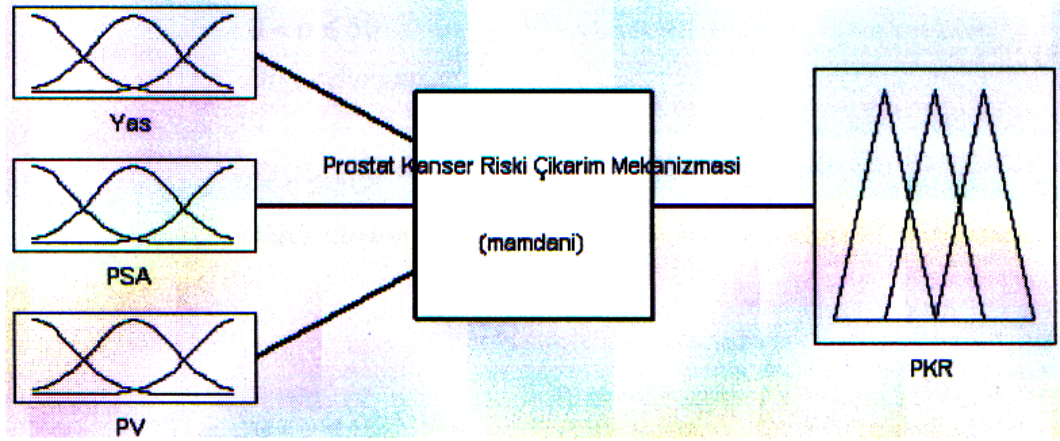
Bu çalışmanın sonuçları Matlab programında çıkarılmıştır. Bu sonuçları kendi oluşturduğum arayüzde deneyerek oluşturduğumuz program test edilmiş olup sonuçlarda Tablo 5.4'de ayrıntılı olarak verilmiştir. Çalışmanın BUS'daki centroid ve bisector sonuçları ve tasarlanan arayüz sonuçları Şekil 5.6 ve Şekil 5.7'de ayrıntılı olarak verilmiştir.

Tablo 5.4 Prostat Kanseri Riski Tespiti için Oluşturulan BUS Arayüzün ve Bulanık Kontrol Arayüzün Test Sonuçları

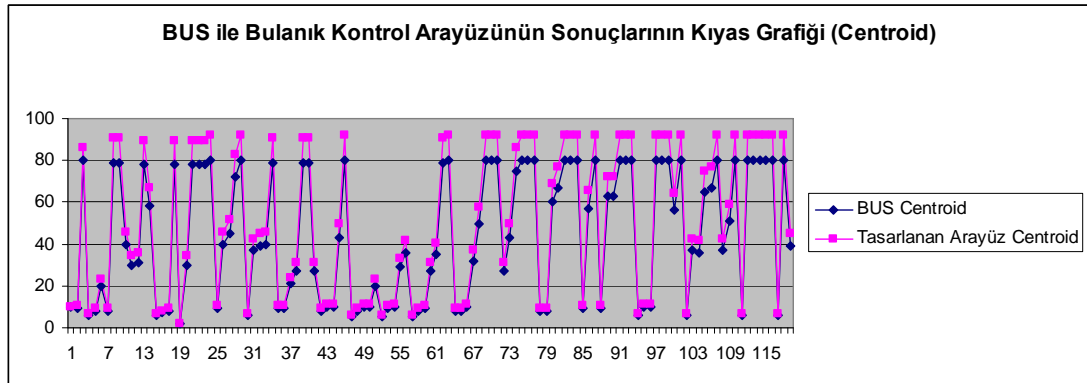
Number of Patients	Input Variables of FES and Online Calculator			FPSA	Result of Biopsy	Risk ratio FPSA/PSA		Risk ratio of online calculator	Output Variable of FES BUS (Centroid) Risk ratio	Bulanık Kontrol Arayüzü			
	Old	PSA	PV			FPSA/PSA	Predict			Centroid	Bisector	Maksimum durulaştırıcı	...
1	44	7,6	38	0,8	Negative	0,11	Yes	12	10	10	12	13	
2	51	6,76	15	0,28	Positive	0,04	Yes	5	9	10	10	5	
3	51	44	83	14	Positive	0,32	No	61	80	88	92	64	
4	53	4,5	39	0,85	Negative	0,19	No	13	6	7	7	14	
5	53	5,83	25	0,4	Negative	0,07	Yes	7	8	9	9	7	
6	53	8,34	25	0,62	Negative	0,07	Yes	10	20	22	23	11	
7	54	5,62	28	0,84	Negative	0,15	Yes	13	8	9	9	14	
8	54	17,3	90	4,75	Negative	0,27	No	38	79	87	91	40	
9	54	17,3	45	1,54	Positive	0,09	Yes	19	79	87	91	20	
10	55	10,51	54	2,36	Negative	0,22	No	26	40	44	46	27	
11	56	8,9	26	3,04	Negative	0,34	No	30	30	33	35	32	
12	56	9,05	39	0,77	Positive	0,09	Yes	12	31	34	36	13	
13	56	16	146	1,35	Negative	0,08	Yes	18	78	86	90	19	
14	57	12,56	52	8,27	Negative	0,66	No	50	58	64	67	53	
15	58	4,48	67,5	0,72	Negative	0,16	No	11	6	7	7	12	
16	58	4,62	48	0,51	Negative	0,11	Yes	9	7	8	8	9	
17	58	5,2	58	1,22	Negative	0,23	No	16	8	9	9	17	
18	58	16,39	27	15,09	Negative	0,92	No	62	78	86	90	65	
19	59	0,28	168	0,12	Negative	0,43	No	3	2	2	2	3	
20	59	8,36	55	0,63	Positive	0,08	Yes	10	30	33	35	11	
21	59	18,2	77	3,23	Negative	0,18	No	31	78	86	90	33	
22	59	19,48	79	4,87	Positive	0,25	No	39	78	86	90	41	
23	59	22,51	42	1,58	Negative	0,07	Yes	20	78	86	90	21	
24	59	22,65	66	2,45	Negative	0,11	Yes	26	80	88	92	27	
25	60	6,58	65	0,97	Negative	0,15	Yes	14	9	10	10	15	
26	60	10,6	30	1,78	Positive	0,17	No	21	40	44	46	22	
27	60	11,45	46	2,23	Negative	0,19	No	25	45	50	52	26	
28	60	14,79	38	1,02	Positive	0,07	Yes	62	72	79	83	65	

29	60	15,51	35	3,26	Negative	0,21	No	31	80	88	92	33
30	61	4,6	37	0,5	Negative	0,11	Yes	8	6	7	7	8
31	61	10,33	62	2,62	Negative	0,25	No	27	37	41	43	28
32	61	10,36	35	2,05	Negative	0,2	No	23	39	43	45	24
33	61	10,59	56	1,8	Positive	0,17	No	21	40	44	46	22
34	61	18,3	62	1,28	Positive	0,07	Yes	66	79	87	91	69
35	62	6,12	52	1,48	Negative	0,24	No	19	9	10	10	20
36	62	6,2	25	0,27	Positive	0,04	Yes	44	9	10	10	46
37	62	8,37	43	0,94	Negative	0,11	Yes	14	21	23	24	15
38	62	8,79	45	0,96	Positive	0,11	Yes	14	27	30	31	15
39	62	20	53	1,04	Positive	0,05	Yes	68	79	87	91	71
40	62	51,74	29	3,52	Positive	0,07	Yes	33	79	87	91	35
41	63	8,8	31	1,98	Positive	0,23	No	23	27	30	31	24
42	64	5,7	36	1,7	Negative	0,3	No	21	8	9	9	22
43	64	6,96	45	0,64	Negative	0,09	Yes	10	10	11	12	11
44	64	8	40	0,6	Positive	0,08	Yes	49	10	11	12	51
45	64	11,08	26	1,12	Negative	0,1	Yes	15	43	47	49	16
46	64	16,28	21	1,13	Positive	0,07	Yes	16	80	88	92	17
47	65	4,39	30	0,95	Negative	0,22	No	14	5	6	6	15
48	65	5,15	47	0,81	Negative	0,16	No	12	8	9	9	13
49	65	7,61	23	0,44	Positive	0,06	Yes	48	10	11	12	50
50	65	7,82	75	1,78	Negative	0,23	No	21	10	11	12	22
51	65	8,33	32	1,21	Positive	0,15	Yes	50	20	22	23	53
52	66	4,38	33	1,03	Negative	0,24	No	15	5	6	6	16
53	66	6,72	61	0,93	Positive	0,14	Yes	13	9	10	10	14
54	66	7,65	89	1,81	Negative	0,24	No	22	10	11	12	23
55	66	9	74	1,7	Positive	0,19	No	21	29	32	33	22
56	66	9,86	49	2,35	Negative	0,24	No	26	36	40	41	27
57	67	4,39	28	0,04	Negative	0,01	Yes	2	5	6	6	2
58	67	5,65	24	0,58	Positive	0,1	Yes	42	8	9	9	44
59	67	6,24	65	1,37	Negative	0,22	No	18	9	10	10	19
60	67	8,2	36	1,67	Positive	0,2	No	20	27	30	31	21
61	67	9,68	41	0,72	Positive	0,07	Yes	53	35	39	40	56
62	67	15,93	69	0,97	Positive	0,06	Yes	63	79	87	91	66
63	67	28	47	4,2	Positive	0,15	No	36	80	88	92	38
64	68	5,09	47	0,12	Negative	0,02	Yes	3	8	9	9	3
65	68	5,51	45	0,62	Negative	0,11	Yes	10	8	9	9	11
66	68	7,2	33	0,26	Positive	0,04	Yes	47	10	11	12	49
67	68	9,25	91	0,33	Positive	0,04	Yes	52	32	35	37	55
68	68	12,1	61	1,95	Negative	0,16	No	23	50	55	58	24
69	68	23,7	109	2,38	Positive	0,1	Yes	26	80	88	92	27
70	68	140	117	20	Positive	0,14	Yes	68	80	88	92	71
71	68	140	54	4,6	Positive	0,03	Yes	38	80	88	92	40
72	69	8,8	34	0,79	Positive	0,09	Yes	12	27	30	31	13
73	69	11,06	38	3,3	Negative	0,3	No	31	43	47	49	33
74	69	15,31	74	4,68	Positive	0,31	No	38	75	83	86	40

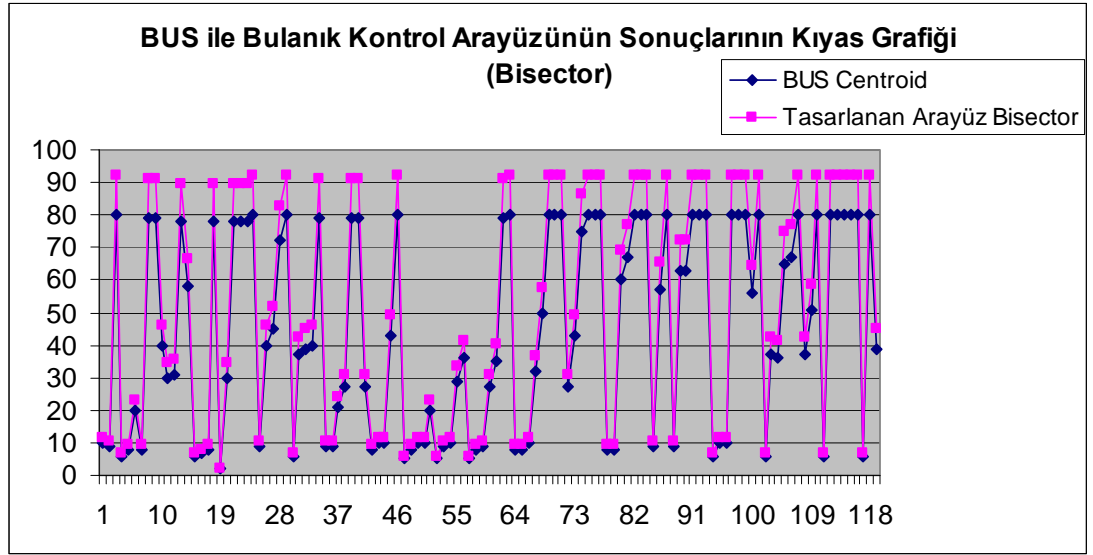
75	69	61	46	6,06	Negative	0,1	Yes	43	80	88	92	45
76	69	70,56	45	4,25	Positive	0,06	Yes	36	80	88	92	38
77	69	146	29	10,7	Positive	0,07	Yes	54	80	88	92	57
78	70	5,39	120	1,03	Negative	0,19	No	15	8	9	9	16
79	70	5,39	42	0,69	Negative	0,13	Yes	11	8	9	9	12
80	70	13	40	2,01	Negative	0,15	No	23	60	66	69	24
81	70	13,95	119	1,92	Negative	0,14	Yes	22	67	74	77	23
82	70	19,2	44	1,94	Positive	0,1	Yes	23	80	88	92	24
83	70	21,94	29	1,56	Positive	0,07	Yes	19	80	88	92	20
84	70	27,7	63	2,49	Negative	0,09	Yes	26	80	88	92	27
85	71	6,08	48	1,3	Positive	0,21	No	43	9	10	10	45
86	71	12,64	50	1,01	Positive	0,08	Yes	59	57	63	66	62
87	71	22	57	2,64	Positive	0,12	Yes	27	80	88	92	28
88	72	6,64	32	1,82	Negative	0,27	No	22	9	10	10	23
89	72	13,31	33	0,51	Positive	0,04	Yes	60	63	69	72	63
90	72	13,31	33	0,5	Positive	0,04	Yes	60	63	69	72	63
91	72	20	48	1,58	Positive	0,08	Yes	20	80	88	92	21
92	72	46	36	4,92	Positive	0,11	Yes	39	80	88	92	41
93	72	77	48	6,4	Positive	0,08	Yes	44	80	88	92	46
94	73	4,65	41	1,95	Negative	0,42	No	23	6	7	7	24
95	73	7,25	19	0,4	Negative	0,06	Yes	47	10	11	12	49
96	73	7,6	74	2,38	Positive	0,31	No	26	10	11	12	27
97	73	19	90	1,3	Positive	0,07	Yes	67	80	88	92	70
98	73	29,52	91	2,9	Negative	0,1	Yes	29	80	88	92	30
99	73	47,4	87	7,53	Positive	0,16	No	48	80	88	92	50
100	74	12,52	27	1,48	Negative	0,12	Yes	19	56	62	64	20
101	74	150	54	25	Positive	0,17	No	72	80	88	92	76
102	75	4,61	16	0,81	Positive	0,18	No	12	6	7	7	13
103	75	10	34	0,76	Positive	0,08	Yes	12	37	41	43	13
104	76	9,81	56	3,67	Negative	0,37	No	33	36	40	41	35
105	76	13,61	61	2,71	Positive	0,2	No	28	65	72	75	29
106	76	13,83	54	2,76	Positive	0,2	No	28	67	74	77	29
107	76	21	86	1,14	Positive	0,05	Yes	69	80	88	92	72
108	77	10	60	0,6	Positive	0,06	Yes	10	37	41	43	11
109	77	12,05	28	3,26	Positive	0,27	No	31	51	56	59	33
110	77	56	51	4,11	Positive	0,07	Yes	36	80	88	92	38
111	78	4,5	180	0,92	Negative	0,2	No	13	6	7	7	14
112	78	26,1	46	2,25	Negative	0,09	Yes	25	80	88	92	26
113	78	26,13	235	2,16	Negative	0,08	Yes	24	80	86	92	25
114	78	31,6	57	2,8	Negative	0,09	Yes	28	80	87	92	29
115	79	17,1	41	1,3	Negative	0,08	Yes	17	80	84	92	18
116	80	69,51	28	20	Positive	0,29	No	68	80	88	92	71
117	81	4,5	28	0,97	Positive	0,22	No	14	6	7	7	15
118	81	68,36	52	24,11	Positive	0,35	No	71	80	88	92	75
119	88	10,4	32	0,78	Positive	0,08	Yes	12	39	43	45	14



Şekil 5.5 Prostat kanser riski tespiti için geliştirilen bulanık uzman sistem



Şekil 5.6 Kanseri Riski için tasarlanan Matlab ile Bulanık Arayüz Sonuçları Centroid yönetimi kıyas grafiği (Centroid)

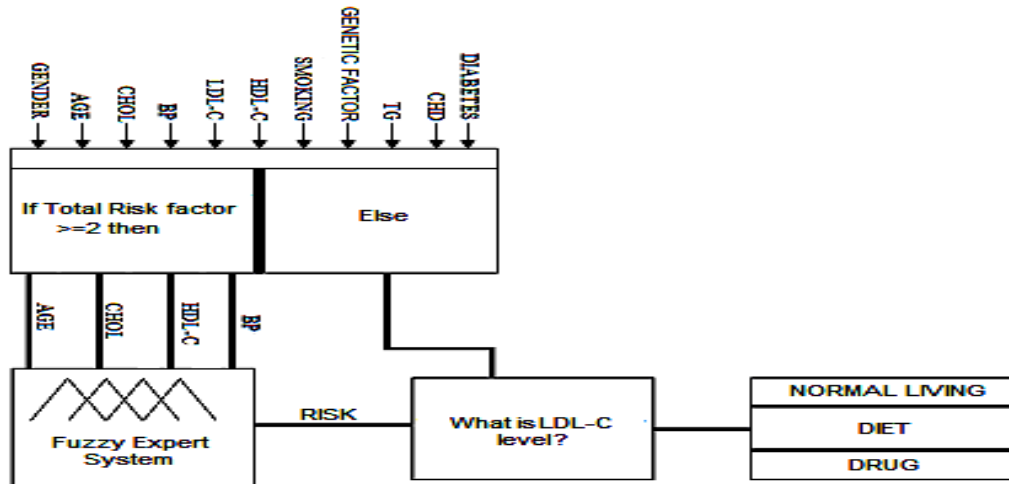


Şekil 5.7 Kanser Riski için tasarlanan Matlab ile Bulanık Arayüz Sonuçları Bisector yönetmi kıyas grafiği (Bisector)

Tablo 5.4'deki test sonuçlarına ve ayrıca Şekil 5.6 ve Şekil 5.7'deki grafiklere baktığımızda prostat kanser riski tespiti için oluşturulan BUS sonuçları ve BUS sonuçlarının oluşturulan Bulanık Kontrol Arayüz programında yapılan testlerinin sonucunda yapılan karşılaştırmalara göre en iyi ve sonuca en yakın durulaştırma yönteminin “centroid” yöntemi olduğu anlaşılmaktadır.

5.3 Koroner Kalp Hastalığı Riskini Saptamak için Tasarlanan Bulanık Uzman Sistemin Test Edilmesi

Bu çalışmaların üçüncüsü de ise; Koroner kalp hastalığı riskini saptamak için geliştirilen bulanık uzman sistemde hastaların cinsiyet, yaş, kolesterol, diyabet, sigara kullanıp kullanmama durumlarına bakılarak koroner kalp hastalığı riskleri ortaya konulmaktadır. Geliştirilen BUS'un çalışma şeması Şekil 5.8'de gösterilmiştir.

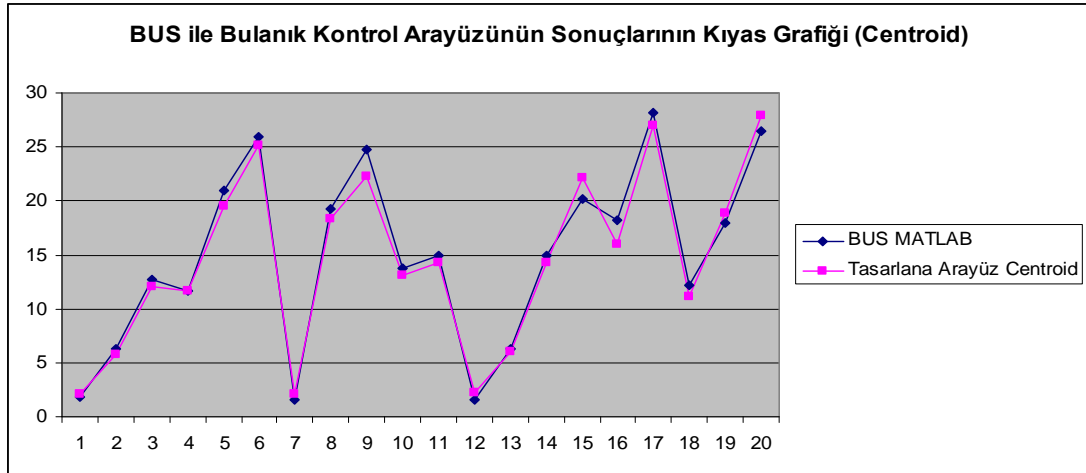


Şekil 5.8 Koroner kalp hastalığı riskini saptamak için geliştirilen bulanık uzman sistemin yapısı

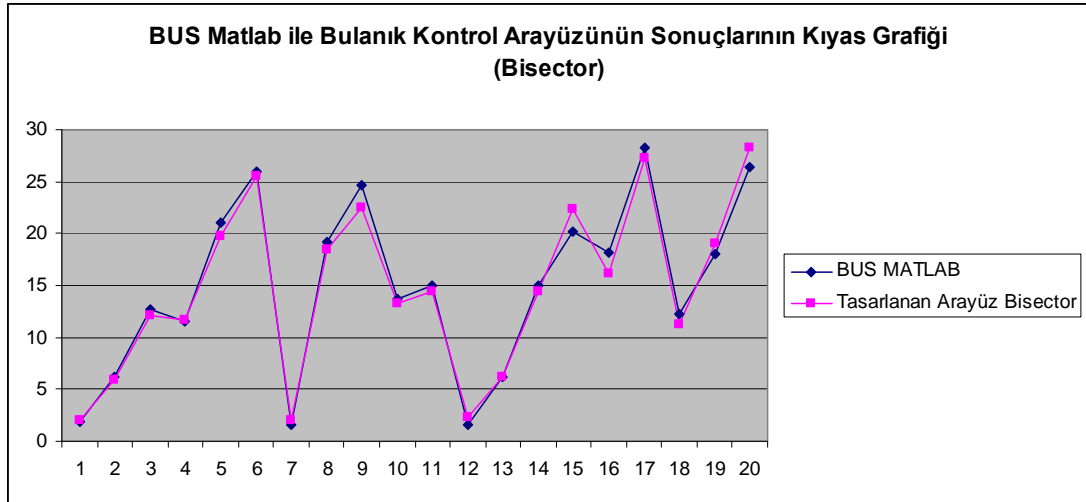
Tablo 5.5 Koroner Kalp Hastalığı Riskini Saptamak için Geliştirilen Bulanık Uzman Sistem Sonuçları ve Tasarlanan Bulanık Kontrol Arayüzün Test Sonuçları (Çıkarım Yöntemi Max-Min)

					Men		Tasarlanan Bulanık				
					No smoking						
Patient Number	AGE	CHOL	HDL-C	BP	MATLAB ile	Visual basic ile	Centroid	Biosector	Maksimum durulaştırıcı	En büyük üyelik ilkesi	Ağırlıklı ortalama
1	30	180	37	160	1,84	2,17	2,06	2,08	1,97	2,00	1,93
2	35	190	45	145	6,27	6,1	5,80	5,86	5,55	5,61	5,43
3	48	260	33	120	12,7	12,68	12,05	12,17	11,54	11,67	11,29
4	57	300	67	110	11,6	12,21	11,60	11,72	11,11	11,23	10,87
5	65	250	54	170	21	20,51	19,48	19,69	18,66	18,87	18,25
6	75	290	25	135	26	26,54	25,21	25,48	24,15	24,42	23,62
7	30	160	49	160	1,54	2,17	2,06	2,08	1,97	2,00	1,93
8	40	310	33	140	19,2	19,25	18,29	18,48	17,52	17,71	17,13
9	55	300	26	200	24,7	23,42	22,25	22,48	21,31	21,55	20,84
10	60	230	39	110	13,7	13,82	13,13	13,27	12,58	12,71	12,30
11	70	210	45	130	15	15	14,25	14,40	13,65	13,80	13,35
12	30	240	50	150	1,54	2,38	2,26	2,28	2,17	2,19	2,12
13	35	180	65	160	6,27	6,4	6,08	6,14	5,82	5,89	5,70
14	45	300	47	155	15	15	14,25	14,40	13,65	13,80	13,35
15	55	300	49	160	20,2	23,34	22,17	22,41	21,24	21,47	20,77
16	65	250	41	140	18,2	16,79	15,95	16,12	15,28	15,45	14,94
17	70	260	38	190	28,2	28,34	26,92	27,21	25,79	26,07	25,22
18	44	210	37	180	12,2	11,79	11,20	11,32	10,73	10,85	10,49
19	55	150	30	200	18	19,9	18,91	19,10	18,11	18,31	17,71
20	66	150	26	200	26,4	29,38	27,91	28,20	26,74	27,03	26,15

Tablo 5.5’de oluşturulan bulanık uzman sistemin yapısını ve oluşturduğumuz uzman sistemin girilen giriş değerlerine verdiği sonuçları gösterilmektedir. Ayrıca, çalışmanın BUS’da ki centroid ve bisector sonuçları ve tasarlanan arayüzün karşılaştırmalı sonuçları Şekil 5.9 ve Şekil 5.10’da ayrıntılı olarak verilmiştir.



Şekil 5.9 Koroner kalp hastalığı riskini saptamak için geliştirilen Matlab Toolbox Bus ile Bulanık Kontrol Arayüzün sonuçları kıyaslama grafiği (Centroid)

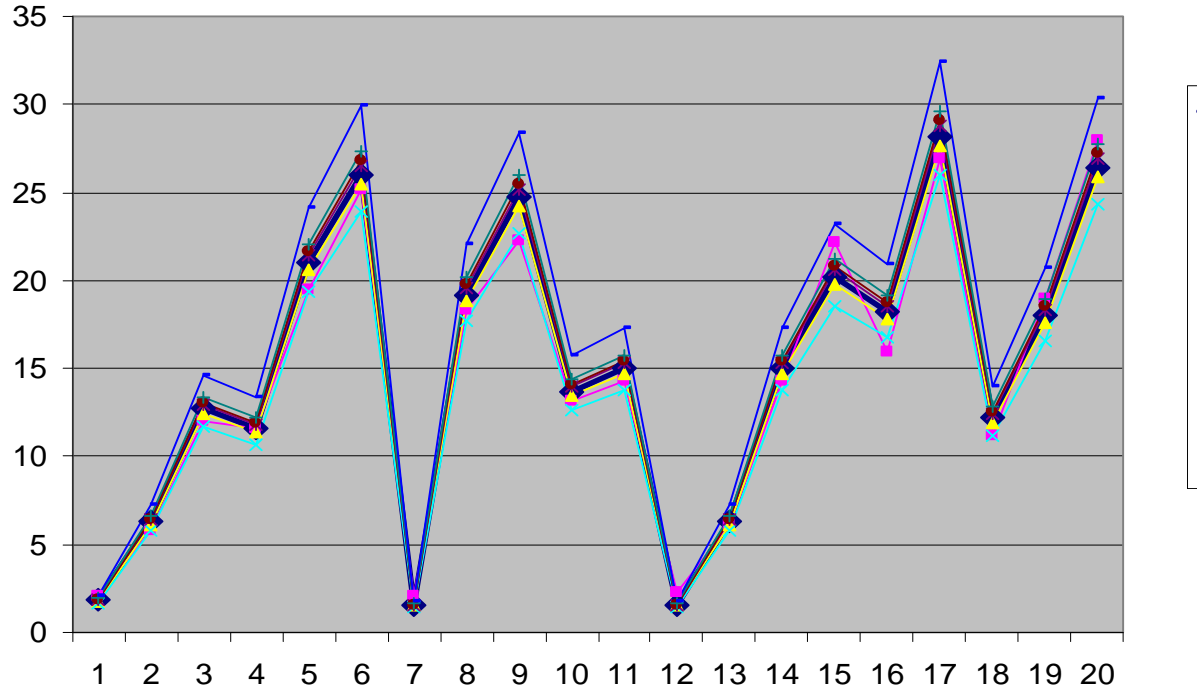


Şekil 5.10 Koroner kalp hastalığı riskini saptamak için geliştirilen Matlab Toolbox Bus ile Bulanık Kontrol Arayüzün sonuçları kıyaslama grafiği (Bisector)

Tablo 5.7 Koroner Kalp Hastalığı Riskini Saptamak için Geliştirilen Bulanık Uzman Sistem Sonuçları ve Tasarlanan Bulanık Kontrol Arayüzün Test Sonuçları (Durulaştırma Yöntemi olarak Bisector Seçilmiştir)

					Men						
					No smoking						
Patient Number	AGE	CHOL	HDL-C	BP	BUS MATLAB	Visual basic ile	Centroid	Tasarlanan Arayüz Bisector	Maksimum durulaştırıcı	En büyük üyelik ilkesi	Ağırlıklı ortalama yöntemi
1	30	180	37	160	1,84	2,17	2,06	2,08	1,97	2,00	1,93
2	35	190	45	145	6,27	6,1	5,80	5,86	5,55	5,61	5,43
3	48	260	33	120	12,7	12,68	12,05	12,17	11,54	11,67	11,29
4	57	300	67	110	11,6	12,21	11,60	11,72	11,11	11,23	10,87
5	65	250	54	170	21	20,51	19,48	19,69	18,66	18,87	18,25
6	75	290	25	135	26	26,54	25,21	25,48	24,15	24,42	23,62
7	30	160	49	160	1,54	2,17	2,06	2,08	1,97	2,00	1,93
8	40	310	33	140	19,2	19,25	18,29	18,48	17,52	17,71	17,13
9	55	300	26	200	24,7	23,42	22,25	22,48	21,31	21,55	20,84
10	60	230	39	110	13,7	13,82	13,13	13,27	12,58	12,71	12,30
11	70	210	45	130	15	15	14,25	14,40	13,65	13,80	13,35
12	30	240	50	150	1,54	2,38	2,26	2,28	2,17	2,19	2,12
13	35	180	65	160	6,27	6,4	6,08	6,14	5,82	5,89	5,70
14	45	300	47	155	15	15	14,25	14,40	13,65	13,80	13,35
15	55	300	49	160	20,2	23,34	22,17	22,41	21,24	21,47	20,77
16	65	250	41	140	18,2	16,79	15,95	16,12	15,28	15,45	14,94
17	70	260	38	190	28,2	28,34	26,92	27,21	25,79	26,07	25,22
18	44	210	37	180	12,2	11,79	11,20	11,32	10,73	10,85	10,49
19	55	150	30	200	18	19,9	18,91	19,10	18,11	18,31	17,71
20	66	150	26	200	26,4	29,38	27,91	28,20	26,74	27,03	26,15

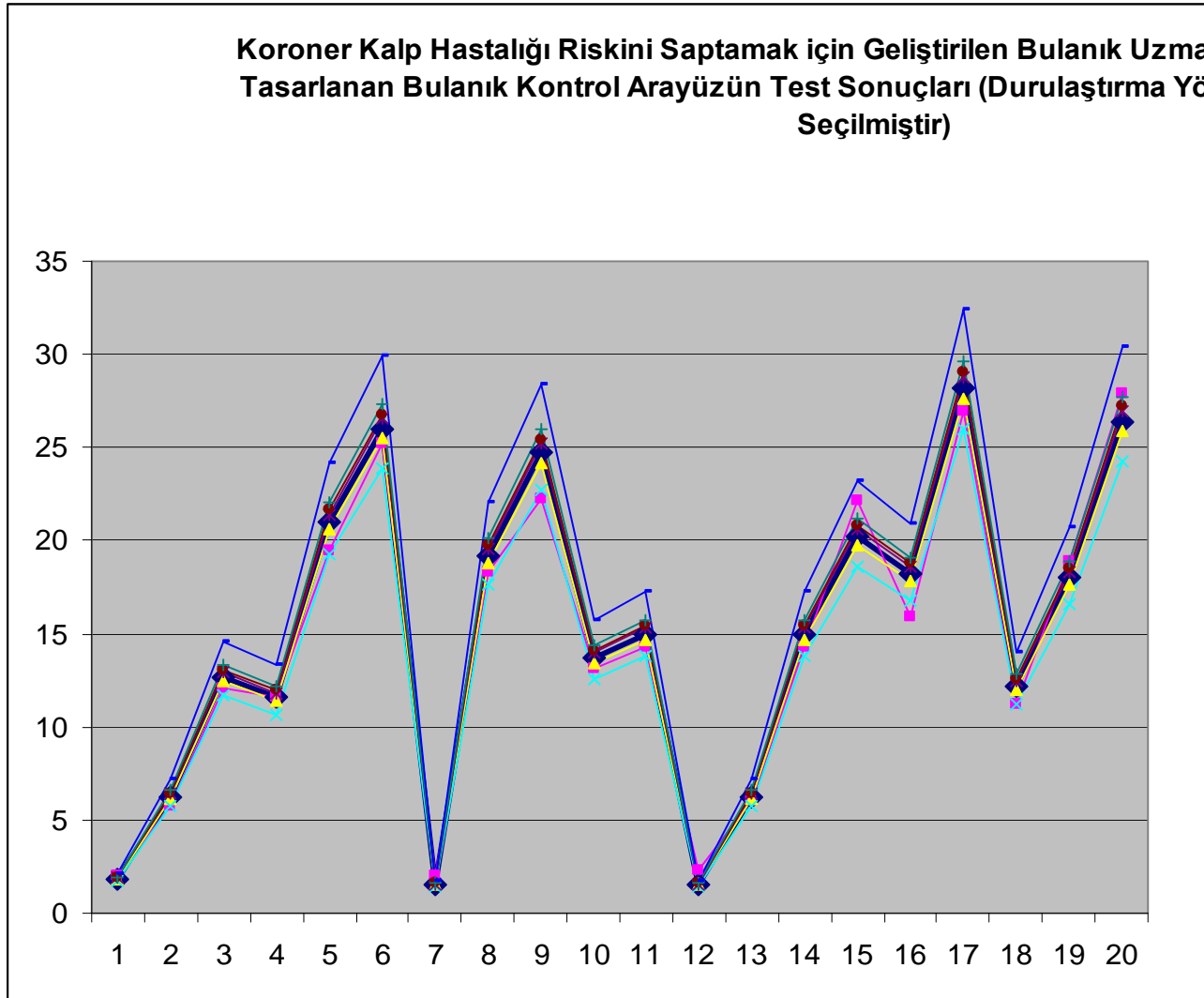
**Koroner Kalp Hastalığı Riskini Saptamak için Geliştirilen Bulanık Uzman
Tasarlanan Bulanık Kontrol Arayüzün Test Sonuçları (Durulaştırma Yöntemleri Seçilmiştir)**



**Şekil 5.12 Koroner Kalp Hastalığı Riskini Saptamak için Geliştirilen BUS
Sonuçları ve Tasarlanan Bulanık Kontrol Arayüzün Farklı Çıkarım
Yöntemlerine Test Sonuçları Grafiği**

Tablo 5.6 Koroner Kalp Hastalığı Riskini Saptamak için Geliştirilen Bulanık Uzman Sistem Sonuçları ve Tasarlanan Bulanık Kontrol Arayüzün Test Sonuçları (Durulaştırma Yöntemi olarak Centroid seçilmiştir)

					Men		Tasarlanan Bulanık Kontrol Arayüzün Sonuçları						
					No smoking								
Patient Number	AGE	CHOL	HDL-C	BP	BUS MATLAB Fuzzy Toolbox ile	Visual basic ile	(Max – Min) Mamdani	(Max-Dot) Larsen	Tsukamoto	Takagi - Sugeno - Kang (TSK)	Lukasiewicz	Zadeh	Dienes - Resher
1	30	180	37	160	1,84	2,17	2,06	1,80	1,69	1,88	1,90	1,93	2,12
2	35	190	45	145	6,27	6,1	5,80	6,14	5,77	6,40	6,46	6,58	7,21
3	48	260	33	120	12,7	12,68	12,05	12,45	11,68	12,95	13,08	13,34	14,61
4	57	300	67	110	11,6	12,21	11,60	11,37	10,67	11,83	11,95	12,18	13,34
5	65	250	54	170	21	20,51	19,48	20,58	19,32	21,42	21,63	22,05	24,15
6	75	290	25	135	26	26,54	25,21	25,48	23,92	26,52	26,78	27,30	29,90
7	30	160	49	160	1,54	2,17	2,06	1,51	1,42	1,57	1,59	1,62	1,77
8	40	310	33	140	19,2	19,25	18,29	18,82	17,66	19,58	19,78	20,16	22,08
9	55	300	26	200	24,7	23,42	22,25	24,21	22,72	25,19	25,44	25,94	28,41
10	60	230	39	110	13,7	13,82	13,13	13,43	12,60	13,97	14,11	14,39	15,76
11	70	210	45	130	15	15	14,25	14,70	13,80	15,30	15,45	15,75	17,25
12	30	240	50	150	1,54	2,38	2,26	1,51	1,42	1,57	1,59	1,62	1,77
13	35	180	65	160	6,27	6,4	6,08	6,14	5,77	6,40	6,46	6,58	7,21
14	45	300	47	155	15	15	14,25	14,70	13,80	15,30	15,45	15,75	17,25
15	55	300	49	160	20,2	23,34	22,17	19,80	18,58	20,60	20,81	21,21	23,23
16	65	250	41	140	18,2	16,79	15,95	17,84	16,74	18,56	18,75	19,11	20,93
17	70	260	38	190	28,2	28,34	26,92	27,64	25,94	28,76	29,05	29,61	32,43
18	44	210	37	180	12,2	11,79	11,20	11,96	11,22	12,44	12,57	12,81	14,03
19	55	150	30	200	18	19,9	18,91	17,64	16,56	18,36	18,54	18,90	20,70
20	66	150	26	200	26,4	29,38	27,91	25,87	24,29	26,93	27,19	27,72	30,36



**Şekil 5.11 Koroner Kalp Hastalığı Riskini Saptamak için Geliştirilen BUS
Sonuçları ve Tasarlanan Bulanık Kontrol Arayüzün Farklı Çıkarım
Yöntemlerine Test Sonuçları Grafiği**

Tablo 5.5'deki test sonuçlarına ve ayrıca Şekil 5.9 ve Şekil 5.10'daki grafiklere baktığımızda koroner kalp hastalığı riskini saptamak için oluşturulan BUS sonuçları ve BUS sonuçlarının oluşturulan Bulanık Kontrol Arayüz programında yapılan testlerinin sonucunda yapılan karşılaştırmalara göre en iyi ve sonuca en yakın durulaştırma yönteminin “centroid” yöntemi olduğu anlaşılmaktadır.

Koroner kalp hastalığı riskini saptamak için geliştirilen Matlab Bus sonuçlarını Bulanık kontrol arayüzünde var olan tüm çıkarım mekanizmaları; durulaştırma yöntemlerinden centroid ve biosector yöntemleri ayrı ayrı seçilerek

sonular alınmıřtır. Bu sonular Tablo 5.6 ve Tablo 5.7’de ayrıntılı olarak gsterilmiřtir. Tablo 5.6 ve Tablo 5.7’de gsterilen sonulara ve řekil 5.11 ve řekil 5.12’de verilen grafiklere gre ıkarılan sonuta geeėe yakın en iyi sonucu yakalamak iin ıkarım mekanizması olarak Mamdani (Max-Min), durulařtırma yntemi olarak Centroid yntemi kullanılması sonucuna ulařılmıřtır.

6. SONUÇ VE ÖNERİLER

Bu çalışmada, bulanık mantık ve bulanık kontrolün temel bilgileri, şu ana kadar yazılmış olan otuzu aşkın bulanık kontrol programının incelemesi ve arayüzleri verilmiş; bu verilmiş olan programlardaki avantaj ve dezavantajlar göz önünde bulundurulmuş bulanık mantık problemleri için yeni bir arayüz tasarlanmış ve tasarlanan bu arayüz ayrıntılı olarak anlatılmıştır. Bulanık mantığın uygulama alanları birçok alanda kendini göstermektedir ve günümüz de artık birçok sistem bu şekilde kontrol edilip sonuca ulaşmaktadır. Belirsizlik ve karmaşıklığın olduğu her alanda sonuca ulaşmak için bulanık kontrol sistemleri kendilerini göstermektedir.

Bu çalışma günümüzde kullanılan programların avantaj – dezavantaj yönlerinden değerlendirilip daha iyiye doğru nasıl bir yolla gidileceği konusunda fikir vermekte ve çözüm sunmaktadır. Çünkü daha önce hazırlamış olan programlarda birçok eksiklik göze çarpmaktadır; bunlar, bulanıklaştırma algoritmalarının az ya da sınırlı oluşu, durulaştırma algoritmalarının az olması, çıkarım metotlarının az olması, yazılımın Türkçe olmaması ya da kullanılan arayüz dilinin görsel ya da metinsel olması gibi eksikliklerdir.

Bulanık mantık problemleri için oluşturduğumuz Türkçe görsel arayüzde bulanıklaştırma, çıkarım mekanizması, durulaştırma yöntemleri diğer oluşturulan arayüzlere göre oldukça fazladır. Çıkarım mekanizması yöntemleri olarak, Mamdani Max-Min yöntemi, Larsen Max-Dot yöntemi, Tsukamoto yöntemi, Takagi - Sugeno - Kang (TSK) yöntemi, Lukasiewicz yöntemi, Zadeh yöntemi, Dienes – Resher yöntemlerini bulmak mümkündür. Durulaştırma yöntemleri olarakta, ağırlık merkezci durulaştırıcı (Centroid), ortamın maksimumu durulaştırıcı (Bisector), maksimum durulaştırıcı, en büyük üyelik ilkesi, ağırlıklı ortalama yöntemi, ortalama en büyük üyelik, toplamların merkezi, en büyük alanın merkezi, en büyük ilk veya son üyelik derecesi olarak belirlenmiş ve arayüze dahil edilmiştir. Yapılan testlerin sonuçlarında seçilen yöntemlere göre yaklaşık olarak %5 ile %10 değerleri arasında sonuç değerlerinin birbirleri arasında bir fark olduğu saptanmıştır. Kaynaklanan bu

farkın yazılımdaki yapılan hesaplamaların sonucu yuvarlamasındaki algoritmada olduğu tahmin edilmektedir.

Bu çalışmanın sonucunda programların ne tür problemlere çözüm ürettiği, ne tür işlemler yaptığı, hangi görüntüler ile sunulduğunun birbirleri arasında ne tür farklılıklar olduğu verilerine ulaşılmış ve bulunan eksiklikler yapılan bu arayüzle giderilmeye çalışılmıştır. Her şeyden önemlisi de daha geniş bulanıklaştırma, durulaştırma, çıkarım mekanizma algoritmalarına sahip ve Türkçe bir arayüz tasarlanmıştır.

7. KAYNAKÇA

Allahverdi, N., 2002. Uzman Sistemler. Bir Yapay Zeka Uygulaması. Atlas Yayınevi, İstanbul. 248 sayfa.

Allahverdi, N., 2006. Bulanık mantık ve sistemler. Ders notları. <http://farabi.selcuk.edu.tr/egitim/bulanik/>, Ocak 2006.

Novruz ALLAHVERDI, Serhat TORUN, Ismail SARITAS; Design of a fuzzy expert system for determination of coronary heart disease risk; Accepted by International Conference on Computer Systems and Technologies, 13-16 June, 2007, Russe, Bulgaria.

Novruz Allahverdi, Ismail Saritaş, Ilker Ali Özkan, Mustafa Argindogan, An Example of Determination of Medicine Dose in the Treatment by Fuzzy Method, International Conference on Computer Systems and Technologies CompSysTech'06, pp IIIA.1.1 – 10, 2006

I.Saritas, I.A.Ozkan, Novruz ALLAHVERDI, Mustafa.Argindogan Determination of the drug dose by fuzzy expert system in treatment of chronic intestine inflammation; Accepted, Journal of Intelligent Manufacturing, 2008.

Nafize, B., Timur, B., 2004. Bulanık Mantık. İlke ve Temelleri, Bıçakçılar Kitabevi, Ankara. 413 sayfa.

İbrahim, A.M., Çervatoğlu, N., 2004. Endüstriye Dönük Uygulamalı: Gömülü Sistemlerde Bulanık Mantık, Bileşim Yayınevi, İstanbul. 192 sayfa.

Şen, Z., 2001. Bulanık (Fuzzy) Mantık ve Modelleme İlkeleri, Bilge Kültür Sanat Yapım Yayıncılık, İstanbul. 152 sayfa.

Zeydin, P.,2003. Borland Delphi 7 Uygulama Geliştirme Rehberi 34 Konuda Delphi 7. Türkmen Kitabevi, İstanbul. 943 sayfa.

M. A. Akcayol, A. Cetin, ve C. Elmas, “An Educational Tool for Fuzzy Logic-Controlled BDCM”, IEEE Transactions on Educations, Vol. 45, No. 1, 2002

H. Aydar ve I. H. Atlas, “Eğitim Amaçlı Bulanık Mantık Denetleyici Simülatörü”, AYSU 2006 Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu, 2006, İstanbul

Balkan, E.,2005. Borland Delphi 7.0. Seçkin Yayıncılık, Ankara. 919 sayfa.

<http://www.dbai.tuwien.ac.at/marchives/fuzzy-mail96/0255.html> (Şubat, 2006)

<http://sipi.usc.edu/~mendel/software/> (Şubat, 2006)

<http://www.fuzzytech.com/> (Şubat, 2006)

http://www.business.com/directory/retail_and_consumer_services/fuzzy_logic_software_corporation/ (Şubat, 2006)

<http://www.rigelcorp.com/fuzzy.htm> (Şubat, 2006)

<http://www.rigelcorp.com/fuzzy.htm> (asemmbly dilinde yazılmış) (Şubat, 2006)

<http://www.megaputer.com/> (Text Mining with TextAnalyst and Web Mining with WebAnalyst WebAnalyst powers the backbone of web based analytics) (Şubat, 2006)

http://www.wisegEEK.com/what_is_fuzzy_logic.htm?referrer=adwords_campaign=fuzzylogic_ad=025541&_search_kw=fuzzy_logic (Şubat, 2006)

<http://www.knowledgestorm.com/search/keyword/FuzzyLogicSoftware/GAWOCT/FuzzyLogicSoftware/> (Şubat, 2006)

<http://www.mathworks.com/> (Şubat, 2006)

<http://www.fuzzy-logic.com/ch3.htm> (Şubat, 2006)

http://www.fuzzytech.com/e/e_a_eet.html (Şubat, 2006)

<http://www.amazon.com/exec/obidos/tg/detail/-/0671875353glance80k> (Şubat, 2006)

<http://www.cs.cofc.edu/~manaris/ai-education-repository/fuzzy-resources.html> (Nisan, 2006)

<http://www.octave.org/octave-lists/archive/helpoctave.1999/msg00419.html> (Nisan, 2006)

<http://www.pricepatternprediction.com/linksource/ai00018.htm> (Nisan, 2006)

<http://www.octave.org/octave-lists/archive/helpoctave.1999/msg00429.html> (Nisan, 2006)

<http://www.mathworks.com/matlabcentral/fileexchange/loadFile.doobjectId=2191> (Nisan, 2006)

<http://www.support-vector.ws/html/downloads.html> (Nisan, 2006)

<http://www.support-vector.ws/html/downloads.html> (Nisan, 2006)

<http://www.geocities.com/capecanaveral/1624/> (Nisan, 2006)

<http://www.dbai.tuwien.ac.at/proj/StartFLIP> (Nisan, 2006)

http://www.transfertech.de/www/soft_e.htm (Nisan, 2006)

<http://www.easynn.com/easynnbases.html> (Nisan, 2006)

<http://inet.uni2.dk/~jemor/wdload.htm> (Nisan, 2006)

<http://www.hav.com/default.html> (Nisan, 2006)

<http://www.hyperlogic.com/demos.html> (Nisan, 2006)

<http://www.ni.com/downloads/> (Nisan, 2006)

<http://www.aist.go.jp/NIBH/~b0616/Lab/links.html> (Mayıs, 2007)

http://prdownloads.sourceforge.net/joone/JooneEditor1.2.1_Win_NoVM.exe/download (Mayıs, 2007)

<http://ac030.osu.cz/irafm/lflc/lflc.html> (Mayıs, 2007)

<http://www.yapayzeka.org> (Mayıs, 2007)

NEFCLASS, <http://fuzzy.cs.uni-mag-deburg.de/nefclass/> (Mayıs, 2007)

FLEB, <http://www.imse.cnm.es/Xfuzzy/Fleb/Fleb.htm> (Mayıs, 2007)

Xfuzzy, http://www.imse.cnm.es/Xfuzzy/Xfuzzy_2.1/index.htm (Mayıs, 2007)