



**T.C.
ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**2021-2022 BAHAR YARIYILI
BULANIK MANTIK (BLM2536) DERSİ
BULANIK SİSTEM TASARIMI ÖDEV RAPORU**

Hazırlayanlar:

Tuğba Taluy 19291085

Erencan Tekin 19290273

Betül Öğmen 19290263

Dersi Veren Kişi:

Araştırma Görevlisi Murat OSMANOĞLU

Özet

Bu ödevde amacımız 2021-2022 bahar yarıyılı süresince almış olduğumuz “bulanık mantık” dersinin en önemli kazanımlarından biri olan bulanık mantık sistemi tasarlamaktır. Bu tasarımı yaparken hem bildiğimiz bir programlama dilini hem de bulanık mantık hesaplamalarında kullanılan matematiksel yaklaşımları uyguladık. Yani bir diğer deyişle tüm dönem boyunca öğrendiğimiz matematiğin, matematiksel yaklaşımların kodlanarak bilgisayar ortamına geçirilmesi ve mantıklı sonuçlar elde etmemiz temel amacımızdır diyebiliriz.

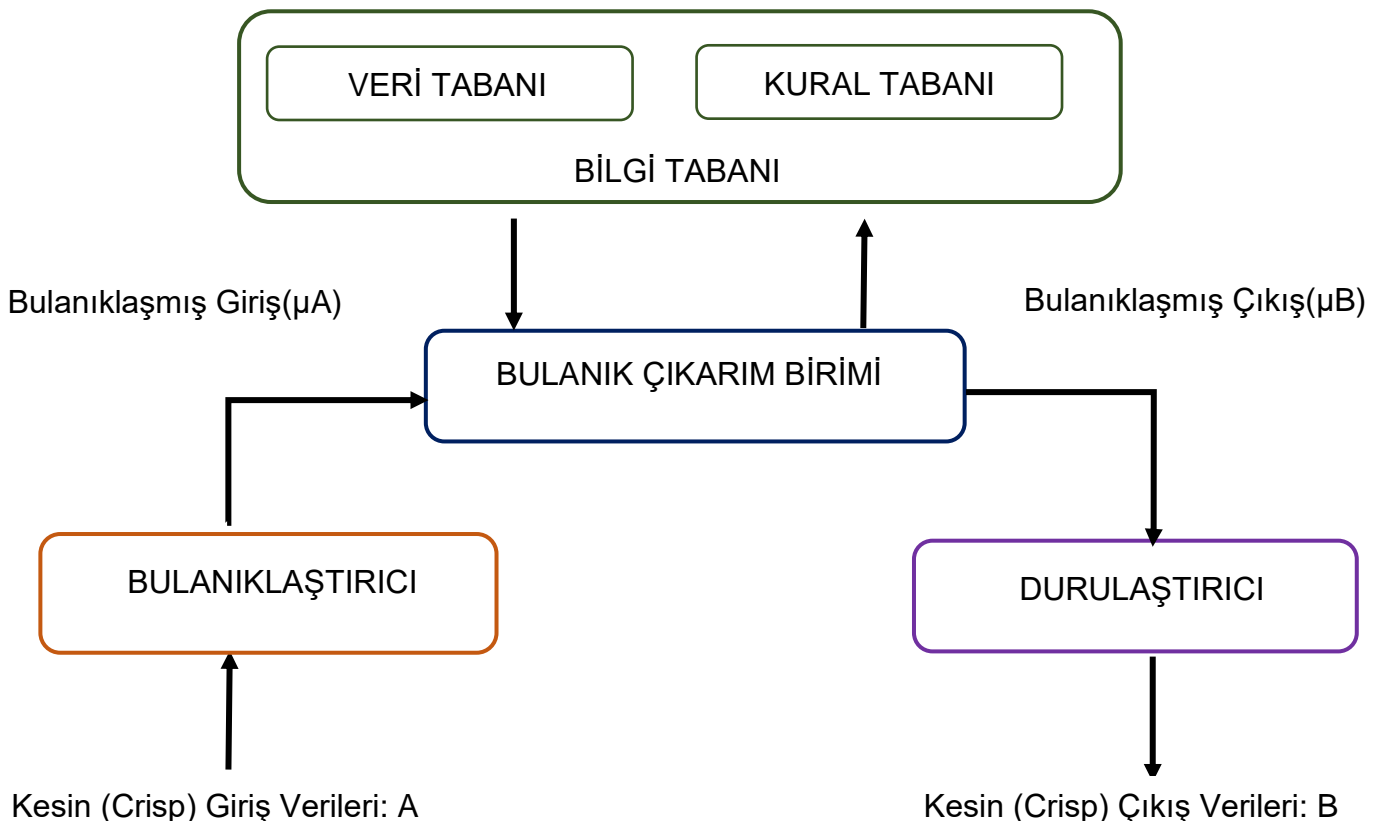
Bir bulanık mantık sistemi bulanıklaştırıcı (fuzzifier), çıkarım (inference), bilgi tabanı (knowledge base) ve durulaştırıcı (defuzzifier) olmak üzere 4 temel parçadan oluşur.

Bulanık mantık sistemleri gerçeğe daha yakın matematiksel sonuçlar ve bu sonuçlardan yola çıkarak daha gerçekçi yorumlar yapmamıza, eğer bu sonuç bir sistemin çalışma prensibini belirleyecekse sistemin daha verimli çalışmasına olanak sağlar.

GİRİŞ

Bulanık kümeler ve bu kümenin elemanlarının üyelik değer fonksiyonlarını kullanarak anlamlı çıktılar elde edebildiğimiz kontrol sistemlerine bulanık mantık sistemleri denmektedir. Her ne kadar elde edilen sonuç sistemin kendi içinde karar verebilmesinde kullanılsa da bulanık mantık sistemleri kendi kendilerine öğrenme eylemini gerçekleştirmezler yani elde edilen sonuç üzerinden herhangi bir öğrenme modeli oluşturma durumu söz konusu değildir. Diğer bir deyişle bulanık mantık sistemleri anlamlı çıktı oluşturmaya sağlar ancak bu çıktıyı kullanarak kendini eğitebilme yeteneğine sahip değildir.

Bir bulanık sistemin parçaları aşağıdaki gibi şematize edilebilir:



- ✓ Bulanık mantık sistemlerinde verinin işlenişi A noktasından B noktasına doğru ilerler. Giriş verilerinin üzerinde herhangi bir işlem yapılmak zorunda olmamakla beraber, bulanık kümeler, dilsel değişkenler ve bunlara bağlı olarak üyelik değer fonksiyonları bulanık sistem çalıştırılmadan önce tanımlanmış olmalıdır.

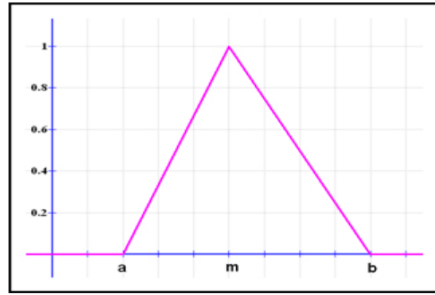
Şimdi sistem elemanlarını kısaca tanıyalım:

Bulanıklaştırıcı (Fuzzifier):

Bulanıklaştırma (fuzzification) işleminin yapıldığı sistem elemanıdır. Bulanıklaştırma ise bilgi tabanında bulunan bilgilerin yardımıyla girdi olarak girilen verinin üyelik değeri $[0,1]$ değer aralığında olan bulanık bir girdiye dönüştürüldüğü basamaktır. Bir diğer deyişle elimizdeki kurallar dahilinde kesin bir değeri, bir üyelik değer fonksiyonuna dönüştürmektir.

1.Üçgensel Üyelik Değer Fonksiyonu:

$$\mu_A(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{m-a}, & a < x \leq m \\ \frac{b-x}{b-m}, & m < x < b \\ 0, & x \geq b \end{cases}$$



Üçgensel üyelik değer fonksiyonları temelde $a \leq m \leq b$ şeklinde 3 sayı ve bu 3 sayı arasındaki tüm

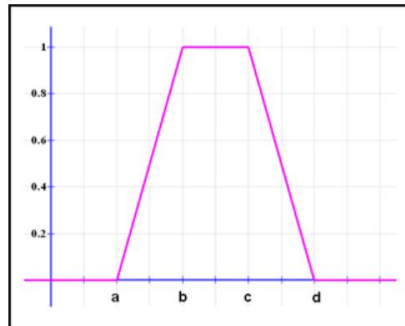
reel sayıların $[0,1]$ değer aralığına sahip üyelik değerlerinden oluşan parçalı fonksiyonlardır. Diğer bir deyişle;

Tanım Kümesi : $[-\infty, \infty] \longrightarrow$ Görüntü Kümesi: $[0,1]$

M sayısı üçgensel sayının yükseklik değerinde görüntüye sahiptir (üyelik değeri 1'dir). Üçgensel üyelik değer fonksiyonu min sol tarafında artan eğimli bir doğrudan, sağ tarafında azalan eğimli bir doğrunun ve bu değerler harici $y=0$ doğrusunun birleşiminden oluşur.

2. Yamuk Üyelik Değer Fonksiyonu

$$\mu_A(x) = \begin{cases} 0, & (x < a) \text{ or } (x > d) \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \end{cases}$$



Yamuk üyelik değer fonksiyonları temelde $a \leq b \leq c \leq d$ şeklinde 4 sayı ve bu 4 sayı arasındaki tüm reel sayıların

$[0,1]$ değer aralığına sahip üyelik değerlerinden oluşan parçalı fonksiyonlardır.

a ve d'nin görüntüleri 0, $[b,c]$ kapalı aralığındaki tüm reel sayılar için görüntü değeri üyelik değer fonksiyonunun yüksekliğidir. $[a,b]$ kapalı değer aralığında artan eğime sahip bir doğrudur. $[b,c]$ aralığı $y=1$ doğrusudur, $[c,d]$ aralığı azalan eğime sahip bir

doğrudur, bu tanım aralıkları $y=0$ değerini alır ve bu değerler yamuk üyelik değer fonksiyonunu oluşturur.

Bulanık Çıkarım Birimi (Fuzzy Inference):

Bilgi tabanıyla ortaklaşa çalışarak (özellikle kural tabanı kısmını kullanarak) bulanıklaştırıcının ürettiği değerden kurallara bağlı olarak mantıklı sonuçlar elde edilen aşamadır. Kurallardan yararlanarak birtakım mantıksal işlemler sonucunda değer elde edilme işlemine çıkarım adı verilir. Çıkarımdan raporun ilerleyen kısımlarında daha ayrıntılı olarak bahsedeceğiz.

Bilgi Tabanı (Knowledge Base):

Bulanık kümeler, kümelere ilişkin bilgiler, kümelerin birbirleriyle olan ilişkileri ve bu ilişkilerin sonuçlarının bulunduğu sistem parçasıdır. Burada yer alan bilgiler doğrultusunda çıkarımın nasıl yapılacağı belirlenir. Bilgi tabanı veri tabanı ve kural tabanı olmak üzere 2 kısımdan oluşur:

- Veri tabanında giriş değerlerine ait üyelik değer fonksiyonları, dilsel değişkenler ve çıkışa ait üyelik değer fonksiyonları bulunur.
- Kural tabanında ise giriş değerlerinin dilsel değişkenler şeklinde ifade edilmiş sonuçlarına göre elde edilmesi beklenen çıkış değerleri (kurallar) yer alır.

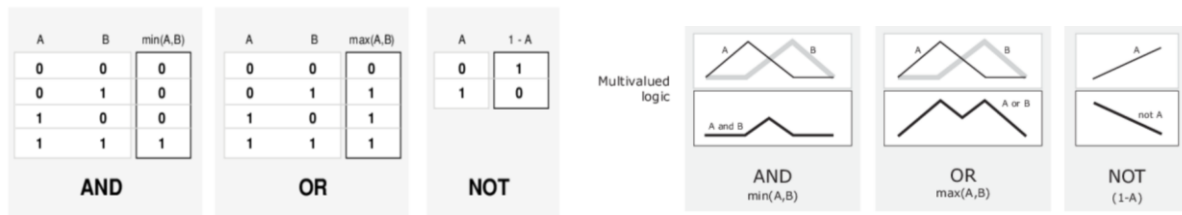
Durulaştırıcı (Defuzzifier):

Durulaştırma (defuzzification) işleminin yapıldığı sistem parçasıdır. Durulaştırma işlemi çıkarımın sonucunda elde edilen bulanık değerlerin problemin sonucunda daha anlamlı bir çözüm elde etmek için yapılan kırılma ve ölçeklendirme işlemlerini kapsar. Yani elimizde var olan sonucu problemin bizden ne istediğine bağlı olarak, kesinleştirmemizi sağlar. Basitçe elimizde var olan bulanık sonucun kesin (crisp) hale getirilmesi demektir.

Çıkarım:

Elimizde olan bilgileri, belirli kurallar dahilinde yeni bilgiler elde etmek amacıyla işlemeye çıkarım adı verilir.

Bir bulanık mantık sisteminde bir kümeye ait birden çok dilsel değişken ve birden çok bulanık küme kullanılarak sonuç elde etmek durumunda olabiliriz. Böyle bir durumda bu değerleri birbirine bağlayan mantıksal ifadelerle diğer bir deyimle “kurallara” ihtiyaç duyarız. Bu kuralları oluştururken klasik mantıkta kullandığımız bağlaçların bulanık mantıktaki hallerini kullanırız.



Şekil 1 Klasik mantık işlemlerinin bulanık mantıktaki karşılıkları

Kurallar Nasıl Sunulur?

Bulanık mantık sistemlerinde çıkarımların kurallar dahilinde yapıldığını söylemiştik. Bu kuralların mantıksal olarak sisteme sunulması gereklidir ve şu şekilde sunulurlar:

EĞER — O HALDE (IF — THEN)
Öncül Sonuç

Eğer hava yağmurluysa yanımıza şemsiye almak iyi bir fikirdir.

Eğer yemek lezzetliyse bahşiş yüksektir.

Eğer çalışan hevesli ve/veya donanımlıysa maaşı yüksektir.

Çıkarım Yöntemleri:

Bulanık mantık sistemlerinde birçok çıkarım yöntemi bulunmakla beraber en çok kullanılanları Mamdani ve Sugeno çıkarım yöntemleridir.

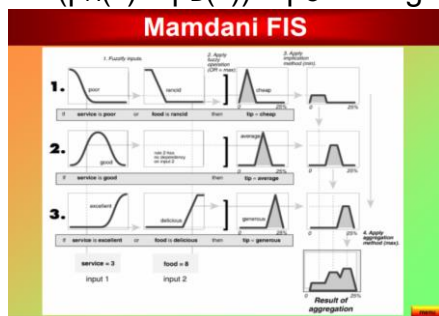
Mamdani Çıkarım Yöntemi

Bulanık mantık sistemlerinde en çok kullanılan çıkarım yöntemidir. Kullanılan matematiksel yaklaşımlar insan mantığına daha kolay yattığından çok tercih edilir. Ayrıca elde edilen sonucu yorumlamak çok daha rahattır. 1975 yılında Londra Üniversitesi'nde çalışan matematik ve bilgisayar bilimci İbrahim Mamdani tarafından geliştirilmiştir.

Mamdani yönteminde giriş ve çıkış değerleri bulanık değerlerdir.

Belirlenen kurallara göre üyelik değerleri hesaplanır ve kuralda geçen mantıksal bağlaçlara göre min veya max işlemi uygulanır. Eğer bağlaç “or/veya” ise değerlerin max değerleri, “and/ve” ise değerlerin min değerleri alınır. A ve B giriş C çıkış değeri olsun:

- $(\mu_A(x) \wedge \mu_B(x)) \vee \mu_C$ ile değer durulaştırılır.



2.Larsen Çıkarım Yöntemi

Mamdani ile birlikte en çok kullanılan çıkarım yöntemlerinden biridir. Mamdaniden farklı olarak giriş değerlerini üyelik değerlerinin minimumu değil, değerlerin cebirsel çarpımları alınır ve çıkış değeri ile maksimumları alınır.

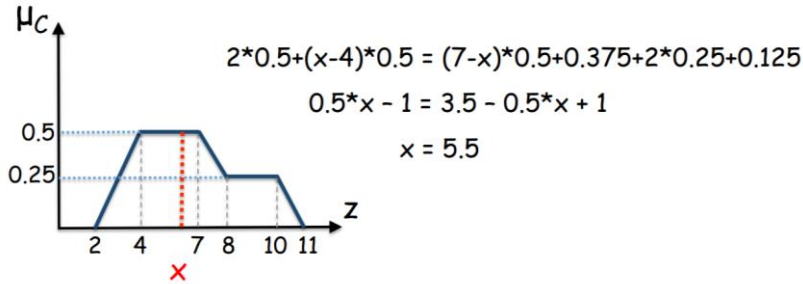
- $(\mu_A(x) \cdot \mu_B(x)) \vee \mu_C$ ile değer durulaştırılır.

Durulaştırma Yöntemleri:

Elde edilen bulanık değerin kesin bir değere dönüştürülmesi işlemidir.

1. İkiye Bölme (Bisector) Yöntemi:

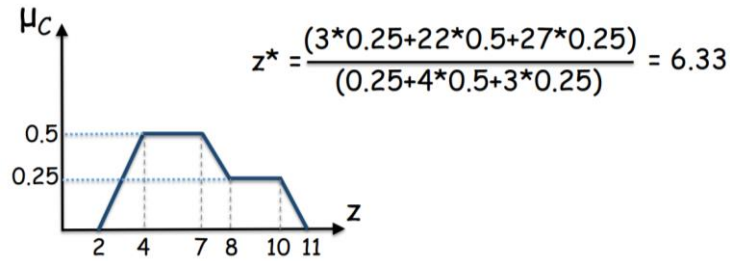
Kuralların uygulanışından sonra elde edilen grafikte grafiğin altında kalan alanı iki eşit parçaya ayıran dikey eksendeki çizginin geçtiği x değerini sonuç kabul eden durulaştırma yöntemidir.



2. Alanların Merkezi-Ağırlık Merkezi (Center of Gravity) Yöntemi:

Kuralların uygulanışından sonra elde edilen üyelik değerinin girdi değeriyle çarpımlarının toplamının üyelik değerlerinin oranıyla kesin sonuç elde edilen yöntemdir. Eğer fonksiyon sürekli ise integral alınır. Ayırık kesitlerden oluşuyor ise toplam formülü

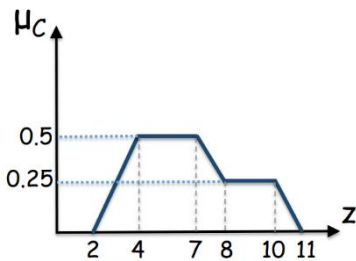
$$z^* = (\sum \mu_C(z_i) \cdot z_i) / (\sum \mu_C(z_i))$$



kullanılabilir.

3. Maximumun Ortası (Mean Of Maxima) Yöntemi:

Maksimumun ortası, en yüksek üyelik değerine sahip alan değerinin ortancasını döndürür.



Bu grafiğin en yüksek üyelik değerine sahip iki değeri var: 4 ve 7.

MOM ile durulaştırmanın sonucu:

$$(4+7)/2=5.5 \text{ olur.}$$

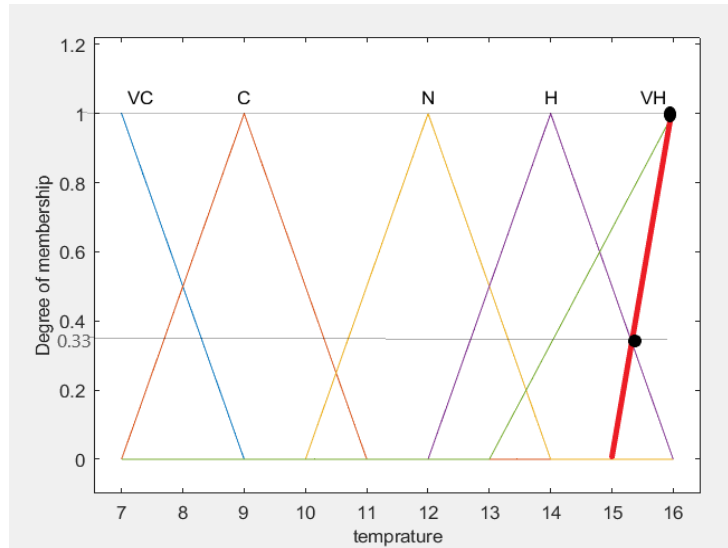
YAPI

Bulanık mantık sistemini oluştururken oluşturduğumuz algoritma aşağıda verilmiştir:

1. Sistemi oluştur.
2. Girdileri (2 girişten oluşan bir sistem tasarlanacak) oluştur.
3. Girdilere ait dilsel değişkenleri tanımla.
4. Çıktıyı (tek çıktıdan oluşan bir sistem tasarlanacak) oluştur.
5. Çıktılara ait dilsel değişkenleri tanımla.
6. Kuralları tanımla.
7. Kuralları sisteme tanı.
8. Giriş değerlerini al.
9. Kurallar dahilinde durulaştır (defuzzification) ve sonucu elde et.

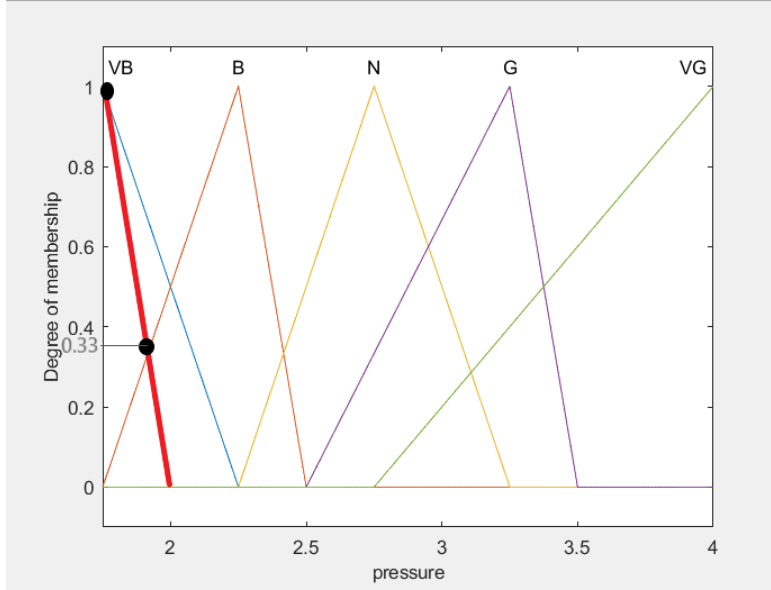
Sistemde Kullanılan Dilsel Değişkenler:

SICAKLIK	BASINÇ	CO ₂ SEVİYESİ
VC = Very Cold	VB = Very Bad	VB = Very Bad
C = Cold	B = Bad	B = Bad
N = Normal	N = Normal	N = Normal
H = Hot	G = Good	G = Good
VH = Very Hot	VG = Very Good	VG = Very Good



Elimizdeki sıcaklık ve basınç değerlerine göre bir gazlı içecekteki karbondioksit yüzdesini bulmak için mamdani yöntemini kullandık. Eğer Sıcaklık girdisini 16 ve Basınç girdisini 1.75 olarak alırsak; Sıcaklık grafiğinde girdi üçgensel sayıya dönüştürülerek çizilir. Sayı, 'H' üyelik fonksiyonunu 0.33' de, 'VH' üyelik fonksiyonunu 1.0' de keser. Basınç grafiğinde de girdi üçgensel olarak çizersek,

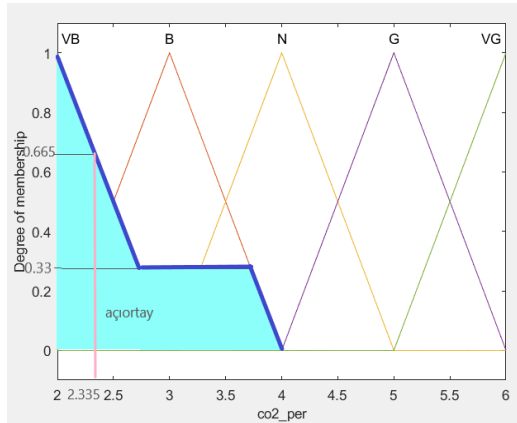
sayı; 'VB' üyelik fonksiyonunu 1.0 'da, 'B' üyelik fonksiyonunu 0.33' de keser. Kural matrisinde; Sıcaklık 'H' ve Basınç 'VB' ise Karbondioksit yüzdesi 'B' olduğu için 0.33 ve 1.0 değerlerinin minimumu (grafikte minimum alanı), karbondioksit yüzdesi grafiğinde 'B' üyelik fonksiyonunda alınır.



Sıcaklık 'VH' ve Basınç 'VB' ise Karbondioksit yüzdesi 'VB' olduğu için 1.0 ve 1.0 değerlerinin minimumu (grafikte minimumum alanı), karbondioksit yüzdesi grafiğinde 'VB' üyelik fonksiyonunda alınır. Sıcaklık 'H' ve Basınç 'B' ise Karbondioksit yüzdesi 'B' olduğu için 0.33 ve 0.33 değerlerinin minimumu (grafikte minimumum alanı), karbondioksit yüzdesi grafiğinde 'B' üyelik değeri

fonksiyonunda alınır. Sıcaklık 'VH' ve Basınç 'B' ise Karbondioksit yüzdesi 'B' olduğu için 0.33 ve 1.0 değerlerinin minimumu (grafikte minimumum alanı), karbondioksit yüzdesi grafiğinde 'B' üyelik fonksiyonunda alınır.

Bu sonuçların da minimumu alındıktan sonra oluşan şeklin (genelde iki yamuğun birleşimi şeklinde görünür) açığortayı bulunur. Açığortayın kestiği karbondioksit yüzdesi, girdilere göre gerekli olan karbondioksit yüzdesini vermektedir.



Rule Matrix	0.33 1.0				
	Verycold	Cold	Normal	Hot	Veryhot
1.0 Verybad	Normal	Bad	Bad	Bad	Verybad
0.3 Bad	Normal	Good	Normal	Bad	Bad
	Normal	Good	Good	Normal	Normal
	Good	Verygood	Good	Good	Normal
	Verygood	Verygood	Verygood	Good	Good

$\mu_{\text{Verycold}}(1.6) = 0$ $\mu_{\text{cold}}(1.6) = 0$ $\mu_{\text{normal}}(1.6) = 0$ $\mu_{\text{hot}}(1.6) = 0.33$ $\mu_{\text{veryhot}}(1.6) = 1.0$
 $\mu_{\text{verybad}}(1.7) = 1.0$ $\mu_{\text{bad}}(1.7) = 0.3$ $\mu_{\text{normal}}(1.7) = 0$ $\mu_{\text{good}}(1.7) = 0$ $\mu_{\text{verygood}}(1.7) = 0$

Üyelik değeri fonksiyonu 0 olmayan değerleri kural matrisindeki kurallara uyguluyoruz.

Sıcaklık		
Hot	VeryHot	
Verybad	Bad	Verybad
Bad	Bad	Bad

$\Rightarrow \begin{matrix} 1.0 & 0.33 & 1.0 \\ 0.33 & 0.33 & 0.33 \end{matrix}$

$\max_{(\text{Bad})} (0.33, 0.33, 0.33) = 0.33$
 $\max_{(\text{verybad})} (1.0) = 1.0$

(Basınç)
 $(1 + 0.33) / 2 = 0.665$ Açığortayın yamuğu kestiği noktanın y değeri
 verybad grafiğinin formülü; $y = -x + 3$ olduğu için x değeri (yani karbondioksit yüzdesi değeri)
 2.335 olur.

Sistemde Kullanılan Kurallar:

1. Sıcaklık VC ve basınç VB ise CO₂ seviyesi N'dir.
2. Sıcaklık VC ve basınç B ise CO₂ seviyesi N'dir.
3. Sıcaklık VC ve basınç N ise CO₂ seviyesi G'dir.
4. Sıcaklık VC ve basınç G ise CO₂ seviyesi VG'dir.
5. Sıcaklık VC ve basınç VG ise CO₂ seviyesi VG'dir.
6. Sıcaklık C ve basınç VB ise CO₂ seviyesi B'dir.
7. Sıcaklık C ve basınç B ise CO₂ seviyesi G'dir.
8. Sıcaklık C ve basınç N ise CO₂ seviyesi G'dir.
9. Sıcaklık C ve basınç G ise CO₂ seviyesi G'dir.
10. Sıcaklık C ve basınç VG ise CO₂ seviyesi VG'dir.
11. Sıcaklık N ve basınç VB ise CO₂ seviyesi B'dir.
12. Sıcaklık N ve basınç B ise CO₂ seviyesi N'dir.
13. Sıcaklık N ve basınç N ise CO₂ seviyesi N'dir.
14. Sıcaklık N ve basınç G ise CO₂ seviyesi G'dir.
15. Sıcaklık N ve basınç VG ise CO₂ seviyesi VG'dir.
16. Sıcaklık H ve basınç VB ise CO₂ seviyesi B'dir.
17. Sıcaklık H ve basınç B ise CO₂ seviyesi B'dir.
18. Sıcaklık H ve basınç N ise CO₂ seviyesi N'dir.
19. Sıcaklık H ve basınç G ise CO₂ seviyesi N'dir.
20. Sıcaklık H ve basınç VG ise CO₂ seviyesi G'dir.
21. Sıcaklık VH ve basınç VB ise CO₂ seviyesi VB'dir.
22. Sıcaklık VH ve basınç B ise CO₂ seviyesi B'dir.
23. Sıcaklık VH ve basınç N ise CO₂ seviyesi N'dir.
24. Sıcaklık VH ve basınç G ise CO₂ seviyesi N'dir.
25. Sıcaklık VH ve basınç VG ise CO₂ seviyesi G'dir.

Sistemi tasarlarken Matlab'i kullandık.

Şimdi sistemin implementasyonuna göz atalım:

Bulanık mantık sisteminde çalışacağımız için bu sistemi tanımlamamız gerekiyor. mamfis fonksiyonu ('mamdani fuzzy inference system') mamdani yöntemi için gerekli olan hazırlıkları tamamlar.

```
fis = mamfis('Name', 'fuzz_system');
```

Sıcaklık ve Basınç isimli iki inputu addInput fonksiyonu ile tanımladık. Bu fonksiyonun 1. argümanına hangi bulanık sistemde çalışacağımızı, 2. argümanına tanımladığımız inputun hangi aralıkta değerler alacağını, 3. ve 4. argümanına grafiklerimizin isimlerini girdik.

```
fis = addInput(fis,[7 16], 'Name', "temprature");  
fis = addInput(fis,[1.75 4.00], 'Name' , 'pressure');
```

Outputu tanımlamak için benzer şekilde addOutput fonksiyonunu kullanırız.

```
fis = addOutput(fis,[2 6], 'Name', "co2_per");
```

Bulanık çıkarım sistemindeki üçgensel değer olan üyelik değer fonksiyonlarını 'fismf' fonksiyonu ile tanımladık, 1. argüman sayının tipini, 2. argüman sayının değerlerini, 3. ve 4. argüman sayının neyi temsil ettiği dilsel değişkeni tutar.

```
mf1 = fismf("trimf",[2 2 3], 'Name', "VB");  
mf2 = fismf("trimf",[2 3 4], 'Name', "B");  
mf3 = fismf("trimf",[3 4 5], 'Name', "N");  
mf4 = fismf("trimf",[4 5 6], 'Name', "G");  
mf5 = fismf("trimf",[5 6 6], 'Name', "VG");
```

Üyelik değer fonksiyonlarını (üçgensel sayıları) Output'a tanımlamak için Outputs(1).MembershipFunctions fonksiyonunu kullanabiliriz. (Eğer iki outputumuz olsaydı ikinci outputu (2) olarak tanımlayacaktık. Tek outputumuz olduğu için sadece (1)'i kullandık.

```
fis.Outputs(1).MembershipFunctions = [mf1 mf2 mf3 mf4 mf5];
```

Sıcaklık ve Basınç grafiklerine dilsel değişkenlerin sayısal karşılıklarını eklemek için üçgensel sayılar tanımladık. addMF fonksiyonunda 2. argüman sayıyı hangi grafiğe eklediğimizi, 3. argüman tipini , 4. argüman sayının değerini, 5. ve 6. argümanlar sayının neyi temsil ettiğini tutar.

```
fis = addMF(fis,"temprature","trimf",[7 7 9], 'Name', "VC");  
fis = addMF(fis,"temprature","trimf",[7 9 11], 'Name', "C");  
fis = addMF(fis,"temprature","trimf",[10 12 14], 'Name', "N");  
fis = addMF(fis,"temprature","trimf",[12 14 16], 'Name', "H");  
fis = addMF(fis,"temprature","trimf",[13 16 16], 'Name', "VH");  
fis = addMF(fis,"pressure","trimf",[1.75 1.75  
2.25], 'Name', 'VB');  
fis = addMF(fis,"pressure","trimf",[1.75 2.25  
2.5], 'Name', 'B');  
fis = addMF(fis,"pressure","trimf",[2.25 2.75  
3.25], 'Name', 'N');  
fis = addMF(fis,"pressure","trimf",[2.5 3.25 3.5], 'Name', 'G');
```

```
fis = addMF(fis, "pressure", 'trimf', [2.75 4 4], 'Name', 'VG');
```

Kuralları bir kural listesinde aşağıdaki gibi tanımlayabiliriz.

- Her bir satırın birinci elemanı tanımladığımız ilk inputun indexi ,
- İkinci elemanı tanımladığımız 2. inputun indexi,
- Üçüncü elemanı tanımladığımız outputun indexi,
- Dördüncü elemanı kuralın önem ağırlığı (1 ile 2 arasında değerler alır),
- Beşinci elemanı birinci ve ikinci eleman arasındaki ilişkiyi belirler.

Örneğin "İlk input(Verycold) ve ikinci input (Normal) olursa output(Good) olur" kuralı 've' bağlacı içerdiği için beşinci eleman '1' dir, 'veya' bağlacı için 2 kullanılır. Rule matrisinde AND operatörünü kullanırız çünkü mamdani çıkarım yönteminde iki input ve bir output olan sistemlerde inputların minimumunu alırız yani 'AND' operatörünü kullanırız.

```
ruleList = [1 1 3 1 1;
```

```
1 2 3 1 1;
```

```
1 3 4 1 1;
```

```
1 4 5 1 1;
```

```
1 5 5 1 1;
```

```
2 1 2 1 1;
```

```
2 2 4 1 1;
```

```
2 3 4 1 1;
```

```
2 4 4 1 1;
```

```
2 5 5 1 1;
```

```
3 1 2 1 1;
```

```
3 2 3 1 1;
```

```
3 3 3 1 1;
```

```
3 4 4 1 1;
```

```
3 5 5 1 1;
```

```
4 1 2 1 1;
```

```
4 2 2 1 1;
```

```
4 3 3 1 1;
```

```
4 4 3 1 1;
```

```
4 5 4 1 1;
```

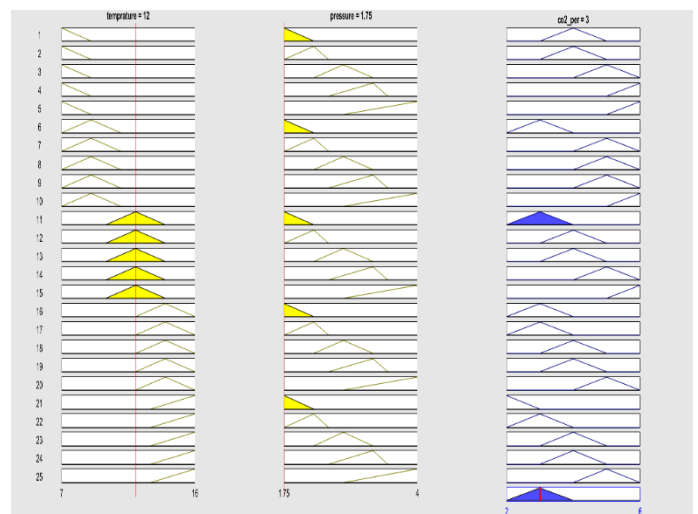
```
5 1 1 1 1;
```

```
5 2 2 1 1;
```

```
5 3 3 1 1;
```

```
5 4 3 1 1;
```

```
5 5 4 1 1];
```

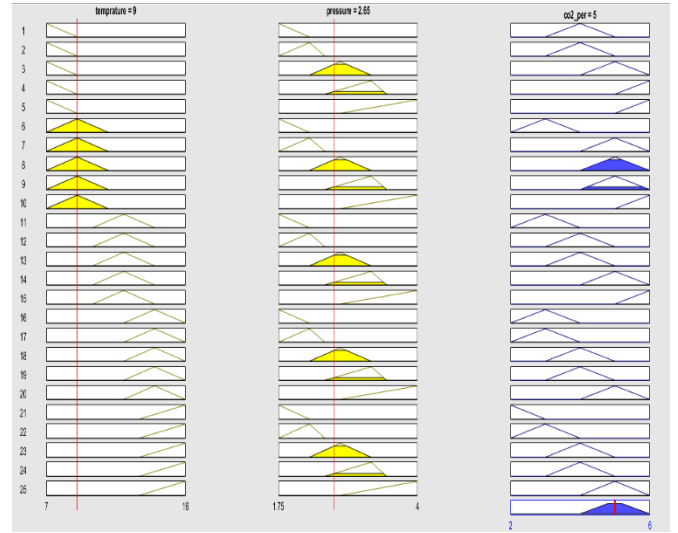


2) [9 2.65] sonuç 5.0000

```

fuzzydeneme x
% Kuralları bulanık mantık sisteminin
fis = addRule(fis,ruleList);
% Sisteme gireceğiniz input değerler
evalfis(fis,[9, 2.65]);
% Bulanık sistemi görüntülemek ister
% görüntülemek isterseniz plotenf fonks
%plotenf(fis);
%plotenf(fis,'input',1) plotenf(fis,'
%plotenf(fis,'output',1);
% Oluşturduğumuz bulanık sistemde i
% görmek için ruleview fonksiyonu k
% ruleview(fis);

```

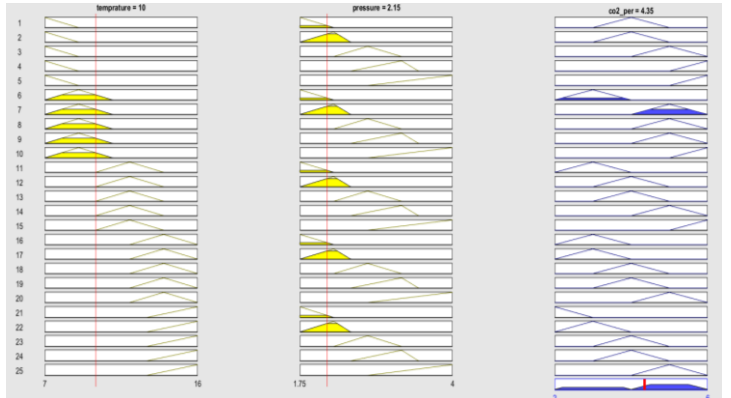


3) [10 2.15] sonuç 4.3511

```

fuzzydeneme x
% Kuralları bulanık mantık sisteminin
fis = addRule(fis,ruleList);
% Sisteme gireceğiniz input değerler
evalfis(fis,[10, 2.15]);
% Bulanık sistemi görüntülemek ister
% görüntülemek isterseniz plotenf fonks
%plotenf(fis);
%plotenf(fis,'input',1) plotenf(fis,'
%plotenf(fis,'output',1);
% Oluşturduğumuz bulanık sistemde i
% görmek için ruleview fonksiyonu k
% ruleview(fis);

```



KAYNAKÇA

Kwang Hee Lee, First Course on Fuzzy Theory and Applications

George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic Theory and Applications

https://www-liphy.univ-grenoble-alpes.fr/pagesperso/bahram/biblio/Zadeh_FuzzySetTheory_1965.pdf

<https://www.massey.ac.nz/~nhreys/MASSEY/159741/Lectures/Lec2012-3-159741-FuzzyLogic-v.2.pdf>

<https://www.tech-wonders.com/2010/08/membership-functions-and-its-features.html>

https://www.omerfarukbay.com.tr/wp-content/uploads/2022/01/BMD-8_BulanıkDenetleyiciModelleri.pdf

<https://www.sciencedirect.com/topic/s/engineering/fuzzification>

<https://www.mathworks.com/help/fuzzy/foundations-of-fuzzy-logic.html#bp78170-5>

<https://medium.com/deep-learning-turkiye/python-ile-bulan%C4%B1k-mant%C4%B1k-modellemesi-74459dc27308>

<https://medium.com/deep-learning-turkiye/python-ile-bulan%C4%B1k-mant%C4%B1k-modellemesi-74459dc27308>

Fuzzy Inference Ders Slaytı

<https://ahmetatasoglu98.medium.com/bulan%C4%B1k-mant%C4%B1k-3-bulan%C4%B1k-kurallar-ve-%C3%A7%C4%B1kar%C4%B1m-8f9d411080c>

<https://ahmetatasoglu98.medium.com/bulan%C4%B1k-mant%C4%B1k-4-bulan%C4%B1k-sistem-uygulamas%C4%B1-f346d35af51>