

# Bulanık-PID Kontrolör Parametrelerinin Diferansiyel Gelişim Algoritması ile En Uygunlaması

Ahmet CİHAN<sup>1</sup>, Cihan KARAKUZU<sup>2</sup>

<sup>1</sup> Kocaeli Üniversitesi, Endüstri Mühendisliği Bölümü

<sup>2</sup> Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği  
Bölümü

[ahmet.can.cihan@gmail.com](mailto:ahmet.can.cihan@gmail.com), [cihankk@kocaeli.edu.tr](mailto:cihankk@kocaeli.edu.tr)

## Özet:

Bu çalışmada, bulanık parametrelerin belirlenmesinde karşılaşılan sorunların üstesinden gelebilmek için yapay zeka teknikleri içerisinde yer alan diferansiyel gelişim algoritması yaklaşımının kullanılabilirliği gösterilmiştir. Doğrusal olmayan bir modele sahip ters sarkaç kontrol problemi için bir bulanık kontrolör tasarlanmış, diferansiyel gelişim algoritması yardımı ile de üçgensel olarak seçilen bulanık üyelik fonksiyonlarına ait parametrelerin etkin bir biçimde belirlenebileceği gösterilmiştir.

## 1. Giriş

Son yirmi yılda bilgisayar teknolojisindeki gelişmelerden sonra matematiksel çözüm yöntemlerinde karşılaşılan zorluklar nedeniyle metasezgisel adıyla bilinen ve en uygun çözümü bulması kesin olmamasına rağmen uygun çözüm sunan algoritmalar geliştirilmiştir. Ayrıca insanların düşünce biçimine benzer biçimde sözel olarak ifade edilen modellerin daha matematiksel olarak ifade edilip modelinin kurulmasına ve modelin çözümüne olanak sağlayan bulanık mantık yöntemleri geliştirilmiştir. Bulanık mantık yöntemlerinin en önemli özellikleri ise doğrusal olmayan sistemler için matematiksel bir modele gereksinimlerinin olmamasıdır. Ancak matematiksel modele gereksinim duymamalarının yanında bulanık sistem parametrelerinin ve kural tabanının belirlenmesi için her ne kadar istatistik bilimine dayalı yöntemler kullanılabilir de doğruluğu kesin bir yöntem geliştirilmemiştir.

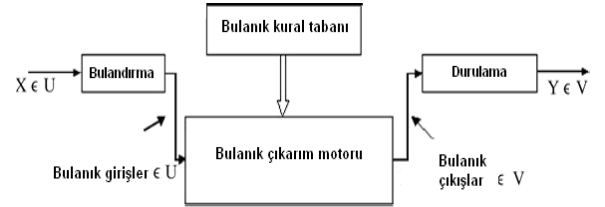
Kullanılan bulanık kontrol sistemlerinde genellikle kullanıcı deneyimlerine dayalı olan üyelik fonksiyonu parametreleri, bulandırma, kural tabanı, durulama parametreleri kullanılmaktadır. Bu durum bilimsel yöntemlerden uzak olması yanında, maliyet ve sistem gereksinimleri açısından da büyük problemlerde sorun teşkil etmektedir. Bu sebeple bu parametrelerin otomatik belirlenmesi son zamanlarda ilgi çeken konu olmuştur. Örneğim [3]'de bu çalışmada da kullanılan ters sarkaç sisteminin kontrolü için genetik algoritma tabanlı bulanık kontrolör tasarımı yapılmıştır. Yine metasezgisel bir yöntem olan PSO ile [4]'de kaotik sistem kontrolü için Sugeno yapılı bulanık-PD kontrolör parametre en iyilemesi yapılmıştır. [5] karınca koloni ile ve [6] Tabu arama algoritmaları ile bulanık kontrolör parametre belirleme

çalışmalarına birer örnektir.

Bu çalışmada bulanık kontrol sistemlerinin üyelik fonksiyonu parametrelerinin belirlenmesi için diferansiyel gelişim algoritmasının kullanımı üzerinde durulmuş ve doğrusal olmayan bir sistem olan ters sarkaç problemi için bulanık mantık kontrolcüsünün üyelik fonksiyonlarına ait parametrelerin değerlerinin bulunması üzerine uygulama yapılmıştır.

## 2. Bulanık Kontrolör Yapısı

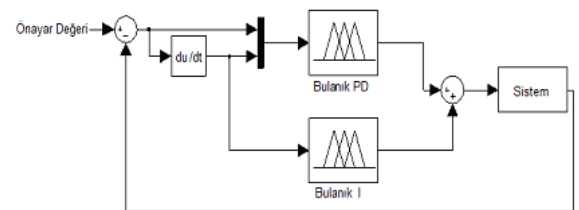
Bulanık mantık Lotfi A. Zadeh tarafından geliştirilmiştir. Bulanık mantık çıkarımları bulanık kümeler ile çalışmaktadır. Klasik kümeler sadece iki değerli üyelik değerlerini kullanırken, bulanık kümeler ara değerleri kısmi üyelik kavramı olarak kullanarak genişletmektedir. Bulanık kümeler en etkin olarak sözel ifadelerin sayısallaştırılmasında yaygın biçimde kullanılır. Bu özelliği bulanık mantık çıkarımlarının daha etkin olmasını da sağlar. Basit bir bulanık çıkarım mekanizmasının işleyişi Şekil 1 ile gösterilmiştir.



Şekil 1: Bulanık çıkarım mekanizması

Bulandırma için genel olarak üçgen üyelik fonksiyonları, yamuk üyelik fonksiyonları, sigmoid üyelik fonksiyonları, daha istatistiksel olarak düşünülebilecek gauss üyelik fonksiyonları gibi üyelik fonksiyonları, bulanık çıkarım motoru olarak mamdani, sugeno ve benzeri bulanık çıkarım motor modelleri, durulama yöntemi olarak ise ağırlık merkezi, en yüksek üyelik derecesi, kenarortayı gibi model ve yöntemler mevcuttur.

Özellikle doğrusal olmayan sistem modelleme ve kontrolünde yaşanan zorlukları aşmada bulanık mantık başarılı olmuştur. Bu sebeple kontrol mühendisliği sahasında da çok çeşitli kullanım alanları bulunmuştur. Bulanık kontrolörler kontrol edilecek sisteme göre çok farklı girişlere ve sistemi kontrol etmekte kullanılan çok farklı çıkışlara sahip olabilirler. Basit bir bulanık-PID kontrolör yapısı Şekil 2 ile gösterilmiştir. Bu kontrolörde iki giriş değişkeni kullanılmakta, sistemi kontrol için de bir kontrol çıkışı üretilmektedir. Giriş olarak hata ve hatanın türevi kullanılmıştır. Farklı sistemler için farklı durumlar oluşabileceğinden yeterli sayıda ve uygun kontrol girişleri belirlenmelidir.



Şekil 2: Bulanık-PID kontrol sistemi

### 3. Diferansiyel gelişim algoritması

Diferansiyel gelişim algoritması [1] son zamanlarda popüler olan populasyon tabanlı en uygunlama algoritmalarından birisidir. Gerçek değerli parametrelere sahip problemlerin çözümünde küresel en uygunlamada kullanılmak üzere tasarlanmış basit bir algoritmadır. Çözüm uzayının çok büyük olduğu durumlarda veya matematiksel yöntemlerin çok uzun zamanda çözüm bulabildiği problemlerde kullanılması daha uygundur. Sezgisel bir algoritma olduğundan en uygun parametreleri bulması kesin değildir. Ancak yeterli seviyelerde çözümler bulması mümkündür. Bulunan çözüm kabul edilebilir seviyede sonuçlar vermekte ise mühendislik sahasında uygulamaya geçirilebilir.

Diferansiyel gelişim algoritmasının pseudo kodu tablo 1 ile verilmiştir. Ayrıca algoritmanın adımları aşağıda kısaca verilmiştir:

1. Algoritma kontrol parametrelerinin belirlenmesi. (D,  $G_{max}$ , NP, F, CR)
2. Başlangıç populasyonunun oluşturulması.
  - NP büyüklüğündeki populasyonun i. bireyi için D parametre sayısı büyüklüğünde vektörel gösterim oluştur.
3. Durdurma kriteri sağlanana kadar mutasyon, rekombinasyon ve seleksiyon.
4. Mutasyon ve rekombinasyon
  - $r_1, r_2, r_3$  değerlerinin rassal olarak (1..NP) tamsayı sınırında birbirlerine ve i değerine eşit olmayacak biçimde seç.
  - $J_{rand}$  rassal olarak (1..D) aralığında tamsayı seç.
  - Her  $j < D$  için aday bireyleri belirle.
  - Eğer  $rand_j [0,1) < CR$  v  $j = j_{rand}$  ise  
 $U_{j,i,G+1} = X_{j,r1,G} + F * (X_{j,r2,G} - X_{j,r3,G})$
  - Diğer durumlarda  $U_{j,i,G} = X_{j,i,G}$
5. Seleksiyon.
  - Eğer  $f(U_{i,G+1}) \leq f(X_{i,G})$  ise  $X_{i,G+1} = U_{i,G+1}$
  - Diğer durumlarda  $X_{i,G+1} = X_{i,G}$
6. Bitir.

Burada; NP populasyon büyüklüğünü ( $\geq 4$  koşulu vardır), F (0,1+) aralığında ölçekleme faktörü olan kontrol parametresini, CR [0,1] aralığında rassal kontrol parametresini, G kaçınıcı nesilde olduğunu gösteren parametreyi, Gmax oluşturulacak en fazla nesil sayısını (durdurma kriteri), U çözüm için aday vektörünü,  $r_1 r_2 r_3$  [1, NP] aralığında rassal seçilmiş tamsayılar ve f en küçüklenecek ölçüt fonksiyonunu temsil etmektedir.

NP adet D boyutlu çözüm vektörü oluşturularak işe başlayan algoritma öncelikle her çözüm vektörü için uygunluk değeri hesaplar. Bir vektörü geliştirmek amacı ile mutasyon ve rekombinasyon için öncelikle hem birbirinden hem de geliştirilecek vektörden farklı üç adet çözüm vektörü rassal olarak seçilir. Vektörün belirttiği parametrenin yeni oluşturulacak olan aday vektörde değişip değişmeyeceği rassal  $rand_j$  değerinin kontrol parametresinden (CR) büyük olup olmasına bağlıdır. Eğer kontrol parametresinden büyük bir sayı rassal olarak üretilmişse parametre değiştirilmez ve aday çözüm vektöründe parametre geliştirilen vektörden alınır. Eğer değiştirilecek ise yeni parametre oluşturulurken seçilen vektörlerden iki tanesi arasındaki değiştirilen parametre için

fark hesaplanır. Bu fark ölçekleme faktörü adı verilen 0-1 aralığında değer alan bir parametre ile çarpılır. Sonrasında da rassal seçilmiş olan üçüncü vektörün parametresine eklenir. Her bir vektör için bu işlemler yapılarak aday vektör populasyonu oluşturulur. Bir sonraki nesile geçecek olan çözümler için mevcut her bir vektör ile o vektörün oluşturulmuş olan aday vektörü arasında uygunluk değerleri karşılaştırılarak daha iyi vektörün bir sonraki nesile geçmesi sağlanır. Bu adımlar durdurma kriteri sağlanıncaya kadar tekrar edilir.

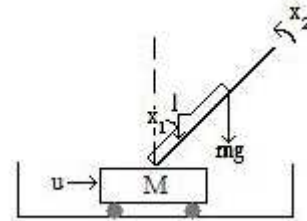
Tablo : Diferansiyel gelişim algoritmasının Psuedo kodu

```
1- Rassal çözümler üret
2- Çözüm vektörlerinin uygunluk değerlerini hesapla.
3- Durdurma kriteri sağlanana kadar tekrarla:
  for i = 1:NP
    Rassal J,  $r_1, r_2$  ve  $r_3$  tamsayılarını üret
    for j = 1:gen sayısı
      Rassal x sayısı üret
      Eğer ( $x \leq CR$ ) veya ( $j = J$ ) ise yap:
         $u(aday)_j = (u(r_1)_j - u(r_2)_j) * F + u(r_3)_j$ 
      Diğer durumda yap:
         $u(aday)_j = u(i)_j$ 
      Son(Eğer)
    Son(j=1:gen sayısı)
    Son(i=1:NP)
  F(u(aday)) uygunluk değerlerini hesapla
  Eğer ( $F(u(aday)) < F(u(i))$ ) ise yap:
    U(i) = U(aday)
  Diğer durumda yap:
    U(i) = U(i)
  Son(Eğer)
  Son(Tekrarla)
4- En uygun çözümü seç.
```

### 4. Uygulama

Yukarıda tanımlanan algoritma ile bulanık kontrolör parametrelerinin belirlenmesi göstermek amacıyla ters sarkaç sisteminin kontrolü seçilmiştir. Ana amaç en uygun kontrol başarımını veren bulanık kontrolör elde etmektir. Kontrol için uygulamada Mamdani bulanık çıkarım mekanizması kullanılacaktır

#### 4.1. Ters sarkaç sistemi



Şekil 3. Ters sarkaç problemi

Kullanılacak olan ters sarkaç sistemi Şekil 3 ile gösterilmiştir. Bu problemde bir yük arabası ve bir sarkaç bulunmaktadır. Yük arabası sağ ve sol yönde yatay olarak yatay eksen boyunca hareket edebilmektedir. Aracın yatay eksen üzerindeki hareketi uygulanan u kuvveti ve sarkaç ağırlığı gibi

sistem parametrelerine büyük ölçüde bağlıdır. Konum sistem çözülürken göz önüne alınmamıştır. Sistemin dinamik denklemleri aşağıda verilen denklemler ile tanımlanmaktadır. Ters sarkaç sistemi kapalı döngüye sahip bir sistem olarak modellenebilir. Ancak bunun için gürültü fonksiyonunun bilinmesi gerekmektedir. Bu modelde sistemde gürültünün olmadığı varsayılmaktadır. Ayrıca sistemin 0 derece noktasının sarkacın yatay eksene dik durduğu konum olduğu kabul edilir. Saat yönünün tersi pozitif kabul edilmektedir.

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \ddot{x}_2 &= \ddot{\theta} = \dot{\omega} = \alpha = \frac{g \sin \theta - \frac{m_p l \dot{\theta}^2 \cos \theta \sin \theta}{m_c + m_p} + \frac{\cos \theta}{m_c + m_p} u}{l \left( \frac{4}{3} - \frac{m_p \cos^2 \theta}{m_c + m_p} \right)} \end{aligned}$$

Bu denklemlerdeki parametreler:

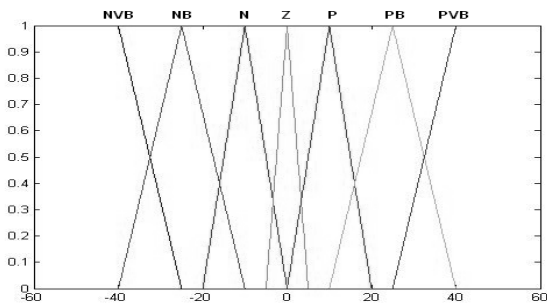
- $m_c$ : Yük arabasının ağırlığı.
- $m_p$ : Sarkacın ağırlığı.
- $g$ : Yerçekimi ivmesi.
- $u$ : Sisteme dışarıdan uygulanan kontrol kuvveti.
- $l$ : Sarkacın yarı uzunluğu.
- $x$ : dikey eksen ile sarkaç kolu arasındaki açı.

Ters sarkaç sistem modeli adi diferansiyel denklem sistemidir. Sistemin çözümü için bu çalışmada Runge-Kutta sayısal analiz yöntemi kullanılmıştır.

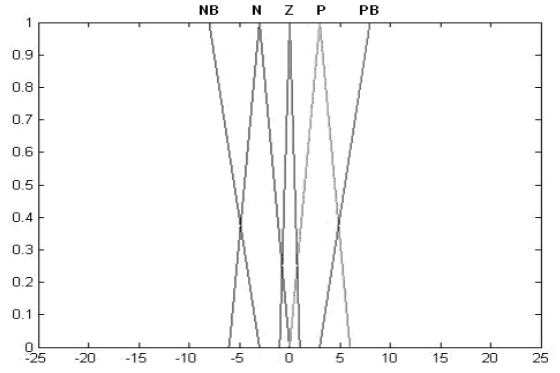
#### 4.2. Başlangıç üyelik fonksiyonlarının belirlenmesi

Sistemi kontrol etmek için kullanılacak olan açı değeri hatası için beş adet üçgensel, iki adet yamuk üyelik fonksiyonu kullanılmıştır. Kullanılacak olan diğer giriş değeri olan açının türevinin hatası için üç adet üçgensel, iki adet yamuk üyelik fonksiyonu (ÜF) kullanılmıştır. Diferansiyel gelişim algoritmasında başlangıç popülasyonu oluşturulurken Üflerin parametreleri ilk ÜFler parametrelerinin 0.5 ile 2 arasında sayılar ile çarpılmasıyla belirlenmiştir. Böylece uygun sayılabilecek bir başlangıç popülasyonu elde edilmiştir. İlk ÜFler olarak [2]'de kullanılan Üfler kullanılmıştır. Bu ÜFler hata girişi için Şekil 4 ve hatanın türevi girişi için Şekil 5'de gösterilmiştir.

Kural tabanı ve çıkış ÜFleri diferansiyel gelişim algoritması tarafından en uygun hale getirilmemiş, sabit kabul edilmiştir. Sabit varsayılan kural tabanı Tablo 2'de, çıkış Üfleri de Şekil 6'da verilmiştir.



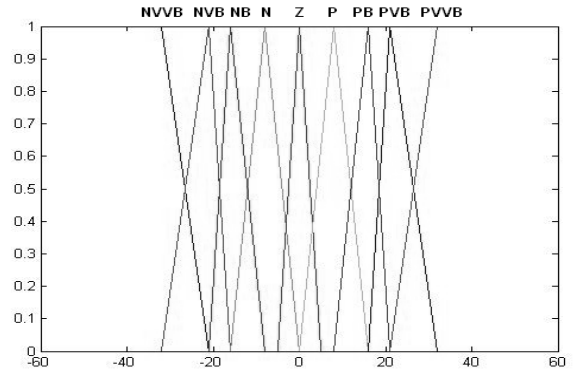
Şekil 4: Bulanık-PD' de kullanılacak hata girişi için başlangıç üyelik fonksiyonları



Şekil 5: Bulanık-PD' de kullanılacak hatanın türevi girişi için başlangıç üyelik fonksiyonları

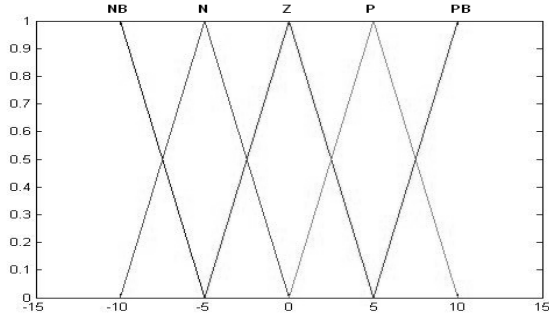
Tablo 2: Bulanık-PD için kural tablosu

X1	X2				
	PB	P	Z	N	NB
PVB	PVVB	PVVB	PVB	PB	P
PB	PVVB	PVB	PB	P	Z
P	PVB	PB	P	Z	N
Z	PB	P	Z	N	NB
N	P	Z	N	NB	NVB
NB	Z	N	NB	NVB	NVVB
NVB	N	NB	NVVB	NVVB	NVVB

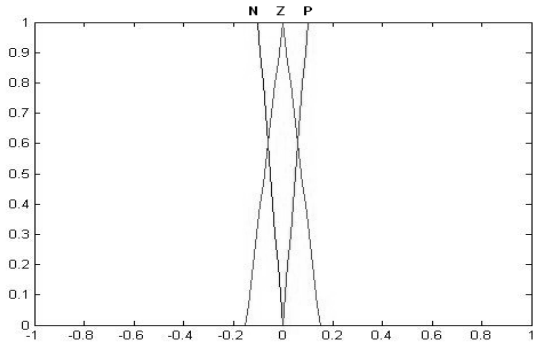


Şekil 6: Bulanık-PD için sabit tutulan çıkış üyelik fonksiyonları

Şekil 2'den de görüleceği üzere Bulanık-PD'nin yanında kalıcı durum hatasını en aza indirebilmek amacı ile ek olarak bulanık integratör (Bulanık-I) de sisteme dahil edilmiştir. Bu birim bulanık-PD'ye ek bir destek sağlamak için tasarlandığından uygulayabileceği ek kuvvet katkısı daha sınırlı tutulmuştur. Bulanık-I'nın giriş üyelik fonksiyonları Şekil 7'de gösterilmiştir. Bu birim girişine uygulanan hata değerine göre bir ek kuvvet çıktısı üretir. Bu mekanizma için de çıkış üyelik fonksiyonları da Şekil 8'de gösterilmiştir. Bulanık integratörün kural tabanı da Tablo 3'deki sabit tutulmuştur.



Şekil 7: Bulanık-I için başlangıç giriş üyelik fonksiyonları



Şekil 8: Bulanık-I için çıkış üyelik fonksiyonları

Tablo 3: Bulanık-I için kural tablosu

e	NB	N	Z	P	PB
u <sub>1</sub>	N	N	Z	P	P

### 4.3. Sistemin parametreleri, karşılaşılan zorluklar ve çözümleri

Başlangıç açısı 30 derece, başlangıç ivmesi 2 derece/saniye,  $m_c = 1$  kg,  $m_p = 0.1$  kg,  $l = 1$  m,  $g = 9.81$  m/s<sup>2</sup> başlangıç parametreleri ile sistem çözümü yapılmıştır. Sistemde her yinelemede 10 saniye için benzetim uygulanmış ve sistemin oturma zamanının, ortalama hatasının, kalıcı durum hatasının ve yaptığı en büyük aşım miktarının en küçük olması amaçlanmıştır. Oturma değeri hesaplanırken otuma bandında sistemin 1 derece sapma yapmasına müsaade edilmiştir. Yukarıda tanımlanan amaca ulaşmak için parametrelere ağırlık verilerek aşağıdaki amaç fonksiyonu tanımlanmıştır.

$$\text{Amaç değeri} = 2 * \text{en büyük aşım miktarı} + \text{oturma zamanı} + 3 * \text{kalıcı durum hatası} + 3 * \text{ortalama hata miktarı}$$

Karşılaşılan en büyük sorun ise üçgen veya yamuk ifade etmesi gereken ÜF parametrelerinin algoritma tarafından değiştirilmesi sonucu uygun bir ifadenin oluşmamasıdır. Yani üçgen üyelik ifade etmesi gereken bir çözüm yapısı üçgen ÜF ifade etmemektedir. Bu durumda, üyelik fonksiyonu parametrelerini üçgen veya yamuk ifade edecek biçime getirmek için ÜF parametreleri uygun değerlerle sınırlandırılmıştır. Algoritma ile ayarlanan 6 adet yamuk

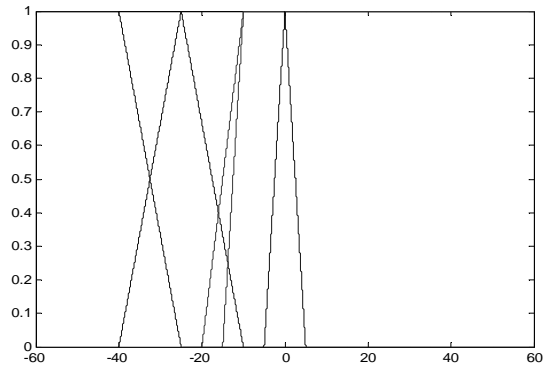
biçimli ÜF için 24 parametrenin yamuk ÜF ifade edecek biçime getirilmesi sağlanmış, 11 adet üçgen biçimli ÜF için 33 parametrenin de üçgen biçimli ÜF ifade etmesi sağlanmıştır. Bu parametrelerle ÜFlerin gereken şekilleri almalarını sağlamak için kuantalama kullanılmıştır. Basit bir örnek Şekil 9 ile gösterilmiştir.

-60	-60	-40	-25	-40	-25	-10	-20	-10	-15	5	0	5
-60	-60	-40	-25	-40	-25	-10	-20	-10	-10	5	0	5

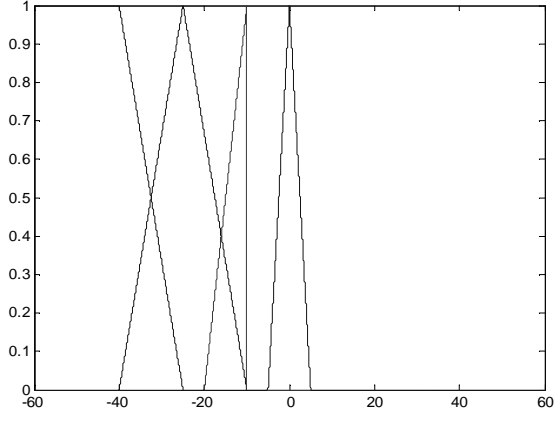
Şekil 9: Kuantalama işleminin vektörel gösterimi

Şekil 9' da görülen 13 adet parametre ile ifade edilen vektörün bir yamuk ve üç adet üçgen üyelik fonksiyonu temsil eden bir yapıda olduğu varsayılmaktadır. Bu vektörün üçüncü alt vektörünün (koyu boyalı olan) bir üçgen üyelik fonksiyonunu temsil etmediği Şekil 10 üzerinde grafiksel olarak görülmektedir. Bu nedenle çözüm değerini üçgen üyelik fonksiyonu biçimine getirmek gerekmektedir. Bunun için alt vektörü bir dik üçgen haline getirmek yeterli olacaktır. Üçgene ait alt kenarın ikinci noktası olan -15 değeri tepe noktası olan -10 değerine doğru kuantalanmıştır. Böylece alt vektörün [-20 -10 -10] değerini alması ve bulanık üçgen fonksiyonu ifade etmesi sağlanmıştır. Kuantalamadan sonra oluşan üyelik vektörleri ise Şekil 11 ile gösterilmiştir.

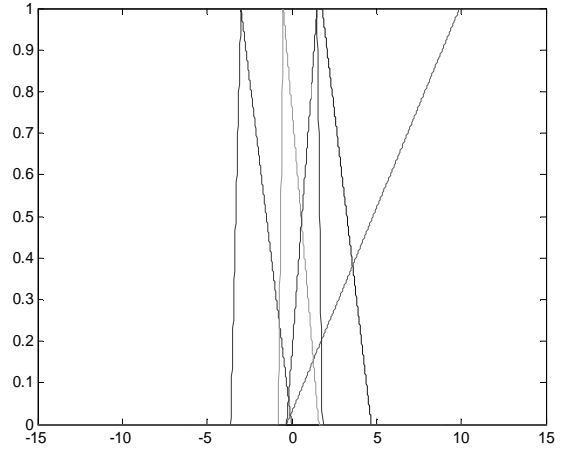
Uygulamada sistem 300 yineleme için çalıştırılmış ve kontrol başarımında iyileşme sağlanmıştır. Sistem ilk olarak mevcut parametreler ile çalıştırıldığında Şekil 12 ile gösterilen cevap elde edilmiştir. 300 yineleme sonunda bulunan parametrelerin tanımladığı ÜFler hata girişi için Şekil 13'de, hatanın değişimi girişi için Şekil 14'de ve hatanın integrali bulanık kontrolörünün girişi için Şekil 15'te gösterilmiştir. Bu şekillerle verilen ÜFler ile çalışan sistemin cevabı da Şekil 16'da verilmiştir. Şekil 12 ve Şekil 16'da verilen sistem kontrol başarımları incelendiğinde algoritma ile elde edilen iyileşme açıkça ortaya çıkmaktadır.



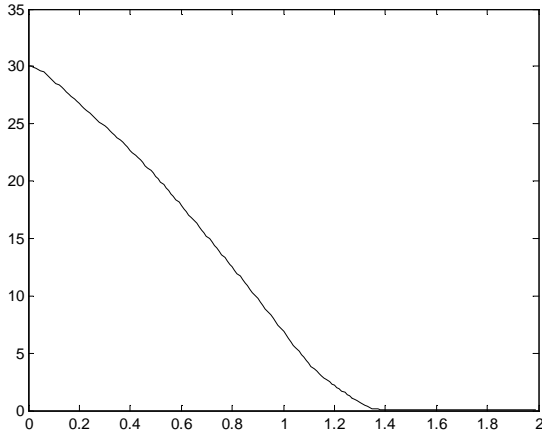
Şekil 10: Kuantalamadan önce üyelik fonksiyonları



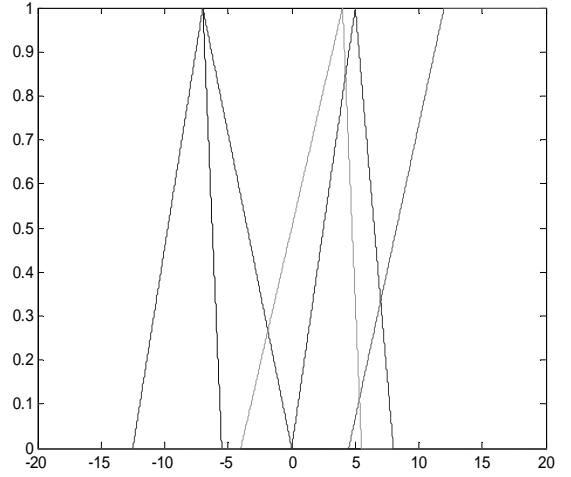
Şekil 10: Kuantalamadan sonra üyelik fonksiyonları



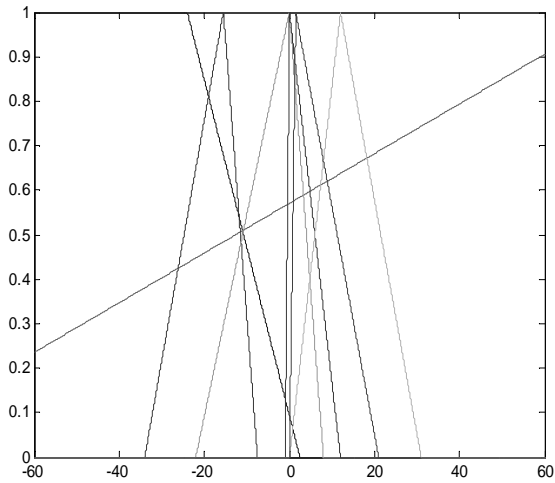
Şekil 14: Parametre en uygunlaması sonunda hatanın türevi girişi için bulunan üyelik fonksiyonları



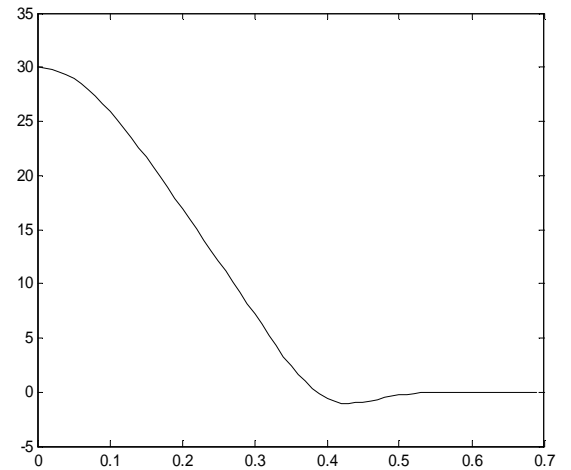
Şekil 12: Parametre en uygunlaması yapılmadan önce sistemin cevabı



Şekil 15: Parametre en uygunlaması sonunda bulanık integratör girişi için bulunan üyelik fonksiyonları



Şekil 13: Parametre en uygunlaması sonunda hatası girişi için bulunan üyelik fonksiyonları



Şekil 16: Parametre en uygunlaması yapıldıktan sonra sistemin cevabı

## 5. Sonuç ve Öneriler

Diferansiyel gelişim algoritması sürekli paramerler söz konusu olduğunda en uygunlamada kullanılan metodlardan birisidir. Çalışmada yöntemin iyi sonuçlar ürettiği görülmektedir. Daha farklı hedef değerleri ile de sistem test edilmiş ve kalıcı durum hatalarının oluşabildiği görülmüştür. Dolayısıyla sistem için gerekli olan hedef değer önceden belirlenmeli ve bu hedef değer için yöntem çalıştırılmalıdır. Bu sayede hedef değerde en kısa zamanda sistemin dengeye geldiği görülecek ve kalıcı durum hatası ise en aza inecektir. Çalışmada çıkış üyelik fonksiyonları parametreleri en uygunlaması yapılmamış olmasına karşın tüm sistemin parametrelerinin en uygulanması ile daha da iyi sonuçlar elde edilebilir. Ayrıca farklı metasezgisel yöntemlerin kullanımları literatürde mevcuttur [3-6]. Bu çalışma mevcut çalışmalara alternatif olarak diferansiyel gelişim algoritmasını sunmuştur.

## 6. Kaynaklar

- [1] Storn R., "Differential Evolution: A Simple and Efficient Heuristic Strategy for Global Optimization over Continuous Spaces", *Journal of Global Optimization*, vol. 11, Dordrecht, pp. 341-359, 1997.
- [2] Becerikli Y., Çelik B.K. , "Fuzzy Control of Inverted Pendulum and Concept of Stability Using Java Application", *Mathematical and Computer Modelling*, vol. 46, pp. 24-37, 2007.
- [3] Wong C.C., Her S.M., "A Self-Generating Method for Fuzzy System Design", *Fuzzy Sets and Systems*, vol. 103, pp. 13-25, 1999.
- [4] Karakuzu C., "Fuzzy Controller Training Using Particle Swarm Optimization for Nonlinear System Control", *ISA Transactions*, vol. 47, pp. 229-239, 2008.
- [5] Baojiang Z., Shiyong L. , "Ant Colony Optimization Algorithm and Its Application to Neuro-Fuzzy Controller Design", *Journal of Systems Engineering and Electronics*, vol. 18, no. 3, pp.603-610, 2007.
- [6] Karaboğa D., Bağış A., Haktanır T., "Controlling Spillway Gates of Dams by Using Fuzzy Logic Controller with Optimum Rule Number", *Applied Soft Computing*, vol. 8, pp. 232-238, 2008.