

**BULANIK UYARLAMALI PARÇACIK SÜRÜ EN İYİLEME YÖNTEM İLE
KESİR DERECELİ DENETLEYİCİLERİN KAOS TABANLI HABERLEŞME
SİSTEMLERİNE UYGULANMASI**

Özkan ATAN

**Doktora Tezi
Elektrik-Elektronik Mühendisliği
Danışman: Doç. Dr. Mustafa TÜRK
MART-2014**

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

**BULANIK UYARLAMALI PARÇACIK SÜRÜ EN İYİLEME YÖNTEM İLE
KESİR DERECELİ DENETLEYİCİLERİN KAOS TABANLI HABERLEŞME
SİSTEMLERİNE UYGULANMASI**

DOKTORA TEZİ

Özkan ATAN

(08113202)

Tezin Enstitüye Verildiği Tarih : 04. 03.2014

Tezin Savunulduğu Tarih : 27.03.2014

Tez Danışmanı : Doç. Dr. Mustafa TÜRK (F.Ü.)

Diğer Jüri Üyeleri : Prof. Dr. Yakup DEMİR (F.Ü.)

Prof. Dr. Mehmet KAYA (F.Ü.)

Doç. Dr. S. Ethem HAMAMCI (İnönü Ü.)

Doç. Dr. Arif GÜLTEN (F.Ü.)

Mart-2014

ÖNSÖZ

Son yıllarda, arařtırmacıların ilgisini çeken kesir dereceli sistemler üzerine yapılan arařtırmalar çok yönlü olarak sürmektedir. Bu çalıřmalar iki ana başlıkta sıralanabilir. Birincisi, kesir dereceli sistemlerin matematiksel analizi; ikincisi ise kesir dereceli sistemlerin mühendislikteki uygulamalarıdır. Matematiksel analizi üzerine yapılan çalıřmalar yüzlerce yıldır sürmesine rağmen, mühendislik alanındaki uygulamalar bu yüzyılın başlarında gerçekleştirilmiş ve her geçen yıl farklı alanlardaki başarımları görülmektedir. Şüphesiz kesir dereceli sistemlerin ilk uygulandığı ve başarımları birçok çalıřmada ortaya konan alan, denetim sistemleridir. Bu tez çalıřmasında da kesir dereceli bir denetleyicinin lineer olmayan bir sistem üzerindeki başarımlarını incelemek ve yine bu sisteme sağladığı avantajları göstererek ileride yapılacak çalıřmalara kaynak oluşturmak amaçlanmıştır.

Bu çalıřmam beni yönlendiren, kendisiyle çalıřmaktan büyük bir memnuniyet duyduğum danışman hocam Sayın Doç.Dr. Mustafa TÜRK'e teşekkürlerimi sunarım.

Ayrıca yardımlarını esirgemeyen ve beni yönlendiren hocam Sayın Yrd. Doç. Dr. Remzi TUNTAŞ'a, çalıřmalarından istifade ettiğim ve kesir dereceli sistemlere olan ilgimin artmasında büyük payı olan Sayın Doç. Dr. Serdar Ethem HAMAMCI hocama, devre modellerinin oluşturulmasında desteklerini esirgemeyen Prof. Dr. Diyi CHEN'e, her zaman beni destekleyen mesai arkadaşlarıma, çalıřmalarım dolayısıyla bazen ihmal ettiğim ama beni anlayışla karşılayan Eşim ve Aileme teşekkür ederim.

Özkan ATAN

ELAZIĞ – 2014

İÇİNDEKİLER

Sayfa No

ÖNSÖZ	I
İÇİNDEKİLER	II
ŞEKİLLER LİSTESİ	V
TABLolar LİSTESİ	IX
SİMGELER LİSTESİ	X
KISALTMALAR LİSTESİ.....	XI
ÖZET	XII
SUMMARY.....	XIII
1. GİRİŞ.....	1
1.1. Kesir Dereceli Sistemler	1
1.2. Kaotik Sistemler ve Uygulamaları	2
1.3. Tezin Amacı	3
1.4. Tezin Kapsamı	4
2. KESİR DERECELİ SİSTEMLER	6
2.1. Kesir Dereceli Sistemlerin Tarihçesi.....	6
2.2. Kesir Dereceli Sistemler için Çözüm Yöntemleri	6
2.2.1. Caputo Yöntemi.....	7
2.2.2. Grünwald-Letnikov Yöntemi	7
2.2.3. Riemann-Liouville Yöntemi.....	9
2.2.4. Kesir Dereceli Sistemlerde Laplace Dönüşüm Yöntemi	9
2.3. Kesir Dereceli Sistemlerin Devre Modelleri	13
2.3.1. Diyi Chen Modeli	14
2.3.2. Dorčák Modeli	15
2.3.3. Xiang-Rong Modeli	15
2.4. Kesir Dereceli Denetleyiciler ve KDPID Tipi Denetleyiciler	18
2.5. KDPID Tipi Denetleyiciler Yapısı	18
2.6. KD Denetleyicilerin Parametrelerinin Akıllı Sistemlerle Belirlenmesi	19
3. KESİR DERECELİ KAOTİK SİSTEMLER VE KAOTİK SENKRONİZASYON	21
3.1. Kaotik Sistemler	21
3.2. Kesir Dereceli Kaotik Sistemler	23

3.2.1.	Kesir Dereceli Sistemlerde Kaosun Varlığı	23
3.3.	Kaos Senkronizasyon	27
3.4.	Kesir Dereceli Van der Pol Sistemi	28
3.4.1.	Kesir Dereceli Van der Pol Osilatörünün Simulink Modeli.....	29
3.4.2.	Kesir Dereceli Van der Pol Osilatörün Devre Modeli.....	32
3.5.	Kesir Dereceli Lorenz Sistemi.....	35
3.5.1.	Kesir Dereceli Lorenz Sisteminin Simulink Modeli	35
3.5.2.	Kesir Dereceli Lorenz Sisteminin Devre Modeli	36
3.6.	Kesir Dereceli Sprott Osilatör	39
3.6.1.	Kesir Dereceli Sprott Osilatörünün Simulink Modeli.....	40
3.6.2.	Kesir Dereceli Sprott Osilatörünün Devre Modeli	42
3.7.	Kesir Dereceli Van der Pol Osilatörünün Senkronizasyonu	44
3.7.1.	Kesir Dereceli Van der Pol Osilatörünün Senkronizasyonu için Simulink Modeli	45
3.7.2.	Kesir Dereceli Van der Pol Osilatörünün Senkronizasyonu için Devre Modeli ...	47
3.8.	Kesir Dereceli Lorenz Sisteminin Senkronizasyonu.....	48
3.8.1.	Kesir Dereceli Lorenz Sisteminin Senkronizasyonu için Simulink Modeli.....	49
3.8.2.	Kesir Dereceli Lorenz Sisteminin Senkronizasyonu için Elektronik Devre Modeli.	50
3.9.	Kesir Dereceli Sprott Sisteminin Senkronizasyonu	52
3.9.1.	Kesir Dereceli Sprott Sisteminin Senkronizasyonu için Simulink Modeli	52
3.9.2.	Kesir Dereceli Sprott Sisteminin Senkronizasyonu için devre modeli	53
3.10.	Sonuç	55
4.	PARÇACIK SÜRÜ ENİYİLEMESİ.....	56
4.1.	Klasik PSO Yöntemi	56
4.2.	Bulanık Uyarlamalı PSO (FAPSO).....	57
4.2.1.	Klasik PSO’da Katsayıların Etkisi ve Hesaplanması	59
4.2.2.	Bulanıklaştırma.....	59
4.2.3.	Durulaştırma	61
4.2.4.	Uyarlama Elemanının Hesaplanması	62
4.3.	KDPID Tipi Denetleyicinin Parametre ve Derecelerinin FAPSO ile Eniyileme ..	64
4.3.1.	Denetleyici Parametrelerinin PSO Yöntemiyle Eniyilemesi.....	64
4.3.2.	Denetleyici Parametrelerinin FAPSO Yöntemiyle Eniyilemesi.....	64
4.4.	Sonuç	66

5.	KAOTİK HABERLEŞME SİSTEMLERİ.....	67
5.1.	Kaotik Analog Modülasyon Yöntemi	67
5.1.1.	Kaotik Maskeleye Yöntemi	67
5.1.2.	Kaotik Modülasyon	68
5.2.	Kaotik Sayısal Modülasyon Yöntemleri	69
5.2.1.	Kaotik Maskeli Faz Kaydırmalı Anahtarlama (K-PSK)	69
5.2.2.	Kaotik Açma-Kapama Modülasyonu (COOK).....	71
5.2.3.	Kaotik Kaydırmalı Anahtarlama (CSK).....	72
5.2.4.	Farksal Kaotik Kaydırmalı Anahtarlama (DCSK)	74
5.2.5.	Korelasyon Gecikmeli Kaydırmalı Anahtarlama (CDSK).....	75
5.3.	KDPID Denetleyicisinin Kesir Dereceli Kaotik Haberleşme Sistemlerine Uygulanması.....	77
5.3.1.	Kaotik Maskeli Analog Haberleşme	77
5.3.2.	Kaotik Sayısal Haberleşme.....	84
5.4.	Sonuç	89
6.	TARTIŞMA, SONUÇ VE ÖNERİLER.....	90
6.1.	Elde Edilen Sonuçlar ve Tartışma	90
6.2.	Gelecekte Yapılabilecek Çalışmalar	92
	KAYNAKLAR	93
	ÖZGEÇMİŞ	104

ŞEKİLLER LİSTESİ

Sayfa No

Şekil 2.1. Farklı çözüm yaklaşımlarına göre $f(t)=t$ fonksiyonunun 0.5 dereceden integrali	12
Şekil 2.2. Crone yaklaşımına göre birim basamak fonksiyonunun farklı KD'ye sahip türev işlemi [51].....	12
Şekil 2.3. Crone yaklaşımına göre birim basamak fonksiyonunun farklı KD'ye sahip integral işlemi	13
Şekil 2.4. Diyi Chen v.d. tarafından oluşturulan KD devre modeli.....	14
Şekil 2.5. Ľubomír Dorčák v.d. tarafından oluşturulan devre modeli	15
Şekil 2.6. Chen Xiang-Rong v.d. tarafından oluşturulan devre modeli.....	16
Şekil 2.7. $n=3$ boyutunda bir devre modeli.....	16
Şekil 2. 8. Klasik P, PI, PD ve PID denetleyicisinin tanımlı olduğu noktalar.....	18
Şekil 2. 9. PI^λ , PD^μ veya $PI^\lambda D^\mu$ denetleyicisinin tanımlı olduğu alan	19
Şekil 3. 1. x işaretinin değişimi $[x_0, y_0, z_0]=[1, 0, 0]$	21
Şekil 3. 2. x işaretinin frekans spektrumu.....	22
Şekil 3.3. $[x_0, y_0, z_0]=[1 0 0]$ ve $[x_0, y_0, z_0]=[1.0001 0 0]$ başlangıç durumlarına göre x1 işaretinin değişimi	22
Şekil 3. 4. $q=0.9$ için oluşan kaotik çeki.....	25
Şekil 3.5. KD çoklu hiperbolik çeker sistemine ait kaotik çeki.....	26
Şekil 3.6. Senkronizasyon hatasının (e_1, e_2 ve e_3) değişimi.....	28
Şekil 3.7. Kesir dereceli Van der Pol osilatörünün Simulink® modeli	29
Şekil 3.8. Kesir dereceli Van der Pol osilatörünün limit çevrimi	30
Şekil 3.9. Farklı kesir derecesine sahip Van der Pol osilatörünün ($q=0.7, 0.8, 0.9$ için) limit çevrimi.....	30
Şekil 3.10. $q =0.7$ için kesir dereceli Van der Pol osilatörünün frekans spektrumu.....	31
Şekil 3.11. $q =0.8$ için kesir dereceli Van der Pol osilatörünün frekans spektrumu.....	31
Şekil 3.12. $q=0.9$ için kesir dereceli Van der Pol osilatörünün frekans spektrumu.....	32
Şekil 3.13. KD Van der Pol kaotik devre modeli	32
Şekil 3.14. KD Van der Pol osilatörünün x değişimi (yatay eksen:Zaman:50ms/kare, düşey eksen x:2/kare).....	33

Şekil 3.15. KD Van der Pol osilatörünün y değişimi (yatay eksen:Zaman 50ms/kare, düşey eksen y 2V/kare)	34
Şekil 3.16. KD Van der Pol osilatörünün limit çevrimi (yatay eksen:x 2V/ kare, düşey eksen y 2V/ kare)	34
Şekil 3.17. Kesir dereceli Lorenz osilatörünün Simulink® ile oluşturulan modeli	35
Şekil 3.18. Kesir dereceli Lorenz osilatörünün kaotik çekisi	36
Şekil 3.19. Kesir dereceli Lorenz osilatörünün frekans spektrumu	36
Şekil 3.20. KD Lorenz sisteminin elektronik devre şeması.....	37
Şekil 3.21. x'in değişim eğrisi (yatay eksen:Zaman:1 ms/kare, düşey eksen x:500mV/kare)	38
Şekil 3.22. y'in değişim eğrisi (yatay eksen:Zaman:1 ms/kare düşey eksen y:500mV/kare)	38
Şekil 3.23. z'in değişim eğrisi (yatay eksen:Zaman:1 ms/kare, düşey eksen z:1V/kare).....	39
Şekil 3.24. KD Lorenz kaotik çekisi (yatay eksen:x:500mV/kare, düşey eksen y:500mV/kare).....	39
Şekil 3. 25. KD Sprot kaotik model.....	40
Şekil 3.26. Sprot kaotik sistemin oluşan kaotik çekisi	41
Şekil 3.27. Sprot sisteminin frekans spektrumu	41
Şekil 3.28. KD Sprot kaotik sisteminin elektronik devre modeli	42
Şekil 3.29. x'in değişim eğrisi (yatay eksen: Zaman 20ms/kare, düşey eksen x 500mV/kare).....	43
Şekil 3.30. y'in değişim eğrisi (yatay eksen:Zaman 1 ms/kare, düşey eksen y 500mV/kare)	43
Şekil 3.31. z'in değişim eğrisi (yatay eksen:Zaman 1 ms/kare, düşey eksen z 500mV/kare)	44
Şekil 3.32. KD Sprot kaotik sisteminin kaotik çekisi (yatay eksen: z 500mV/kare, düşey eksen z 500mV/kare)	44
Şekil 3. 33. Van der Pol sisteminin senkronizasyonu için Simulink modeli	45
Şekil 3.34. Van der Pol sisteminin PID ile senkronizasyon denetimi	46
Şekil 3.35. Van der Pol kaotik sisteminin KDPID ile senkronizasyon denetimi	46
Şekil 3.36. KD Van der Pol sisteminin KDPID ile senkronizasyon kontrol devresi.....	47

Şekil 3.37. KDPID ile kontrol edilen iki farklı Van der Pol devresinin ürettiği kaotik sinyallerin şekilleri (yatay eksen:Zaman 50ms/kare, düşey eksenler x1 ve x2 500mV/kare).....	48
Şekil 3.38. KD Lorenz kaotik sisteminin denetleyicisinin Simulink modeli	49
Şekil 3.39. PID ile senkronizasyon kontrolü yapılan iki KD Lorenz kaotik sistemlerinin çıkışları	49
Şekil 3.40. KDPID ile senkronizasyon kontrolü yapılan iki KD Lorenz kaotik sistemlerinin çıkışları	50
Şekil 3.41. PID ile Lorenz kaotik sisteminin senkronizasyon kontrolü(yatay eksen:Zaman 50ms/kare, düşey eksenler x1 ve x2 500mV/kare).....	51
Şekil 3.42. KDPID ile Lorenz kaotik sisteminin senkronizasyon kontrolü (yatay eksen:Zaman 50ms/kare, düşey eksenler x1 ve x2 500mV/kare)	51
Şekil 3.43. KD Sprot kaotik sisteminin senkronizasyonu için Simulink modeli	52
Şekil 3.44. PID tipi denetleyiciyle senkronizasyon denetimi yapılan iki kaotik sistemin çıkışı	53
Şekil 3.45. KDPID tipi denetleyiciyle senkronizasyon denetimi yapılan iki kaotik sistemin çıkışı	53
Şekil 3.46. PID devre simülasyonundan elde edilen çıkış değişimleri (yatay eksen:Zaman 50ms/kare, düşey eksenler x1 ve x2 500mV/kare).....	54
Şekil 3.47. KDPID devre simülasyonundan elde edilen çıkış değişimleri (yatay eksen:Zaman 500 µs/kare, düşey eksenler x1 ve x2 500mV/kare)	54
Şekil 4.1. Klasik PSO'ya ait akış diyagramı.....	58
Şekil 4.2. d_g girişe ait üyelik fonksiyonu.....	60
Şekil 4.3. C_1 'ye ait üyelik fonksiyonu	60
Şekil 4.4. C_2 'ye ait üyelik fonksiyonu	61
Şekil 4.5. FAPSO yöntemi için akış diyagramı	63
Şekil 4.6. FAPSO yakınsama eğrisi.....	65
Şekil 4.7. Denetleyici kazanç parametreleri ve derecelerinin değişimi	66
Şekil 5.1. Kaotik maskeleyme blok diyagramı	68
Şekil 5.2. K-PSK yöntemine ait blok diyagramı	69
Şekil 5.3. a) $b(t)= 0011101100$ için veri işareti b) modüle edilmiş işaret	70
Şekil 5.4. Evre uyum demodülasyon yöntemine ait blok diyagramı.....	71
Şekil 5.5. COOK modülatörüne ait blok diyagramı	71
Şekil 5.6. COOK için evre uyumsuz demodülatör	72

Şekil 5.7. CSK Modülatörüne ait blok diyagramı	73
Şekil 5.8. Korelatör temelli CSK demodülatör	73
Şekil 5.9. (a) DCSK Modülatör ve (b) demodülatör	75
Şekil 5.10. a) CDSK modülatör, b) demodülatör	76
Şekil 5.11. KDPID ile yapılan denetleyiciye ait Simulink modeli	77
Şekil 5.12. Maskeleyenme bloğunun içyapısı	78
Şekil 5.13. Kaotik sinyallerin oluşturduğu kaotik çeki.....	78
Şekil 5.14. a) Veri işareti, b) maskelenmiş veri işareti	79
Şekil 5.15. Kanal modeli	80
Şekil 5.16. Alıcı girişindeki maskelenmiş gürültülü veri	80
Şekil 5.17. Maske kaldırma için kullanılan Simulink modeli	81
Şekil 5.18. Denetleyici modeli.....	81
Şekil 5.19. Klasik PID denetleyici kullanılan sistemde oluşan senkronizasyon hatası	82
Şekil 5.20. Klasik PID denetleyici kullanılan sistemde oluşan veri hatası.....	82
Şekil 5.21. KDPID denetleyici kullanılan sistemde oluşan senkronizasyon hatası.....	83
Şekil 5.22. KDPID denetleyici kullanılan sistemde oluşan veri hatası	83
Şekil 5.23. Kaotik maskeli sayısal haberleşme sisteminin Simulink modeli	84
Şekil 5.24. Bilgi işareti	84
Şekil 5.25. PSK ile modüle edilmiş işaret	85
Şekil 5.26. Gürültülü maskelenmiş veri	86
Şekil 5.27. Klasik PID denetleyicisi kullanıldığında oluşan senkronizasyon hatası.....	86
Şekil 5.28. KDPID denetleyicisi kullanıldığında oluşan senkronizasyon hatası.....	87
Şekil 5.29. Maske kaldırma işleminden sonra PSK taşıyıcısının değişimi.....	87
Şekil 5.30. Gönderilen veri ile demodülasyonda kullanılan integratörün çıkışı.....	88
Şekil 5.31. BER grafiği.....	88

TABLolar LİSTESİ

	<u>Sayfa No</u>
Tablo 2.1. KD integratörünün transfer fonksiyonu [52]	14
Tablo 2.2. KD devre modeline ait direnç değerleri [55]	17
Tablo 2.3. KD devre modeline ait kondansatör değerleri [55]	17
Tablo 3.1. KD Van der Pol kaotik devre elemanlarının değeri ve modeli	33
Tablo 3.2. KD Lorenz kaotik devre elemanlarının değeri ve modeli	37
Tablo 3.3. KD Sprott kaotik devresine ait elemanlarının değeri ve modeli	42
Tablo 3.4. Önerilen KDPID denetleyicisine ait devre elemanlarının modeli ve değeri	48
Tablo 4.1. PSO optimizasyon adımları	64
Tablo 4.2. FAPSO optimizasyon adımları	65
Tablo 4.3. KDPID denetleyicisinin parametre ve dereceleri	66
Tablo 5.1. SNR'ye göre BER'in değışimi	88
Tablo 6.1. Tasarlanan denetleyici ile denetlenen senkronizasyon sonucu elde edilen verilerin literatürle karşılaştırılması	92

SİMGELER LİSTESİ

C_1 ve C_2	: İvmelenme sabitleri
$e(t)$: Senkronizasyon hatası
G_{best}	: Küresel en iyi sonuç
K_d	: Türev denetleyici sabiti
K_i	: İntegral denetleyici sabiti
K_p	: Oransal denetleyici sabiti
P_{best}	: Önceki en iyi sonu
ω	: Atalet ağırlığı
ω_{max} ve ω_{min}	: Atalet ağırlığının en büyük ve en küçük değerleri
μ	: Kesir dereceli denetleyicinin türev derecesi
λ	: Kesir dereceli denetleyicinin integral derecesi

KISALTMALAR LİSTESİ

- COOK** : Kaotik Açma-Kapama Anahtarlama
- CSK** : Kaotik Kaydırmalı Anahtarlama
- CDSK** : Korelasyon Gecikmeli Kaydırmalı Anahtarlama
- DCSK** : Farksal Kaotik Kaydırmalı Anahtarlama
- FAPSO** : Bulanık Uyarlamalı Parçacık Sürü Eniyilemesi
- KD** : Kesir Dereceli
- KDD** : Kesir Dereceli Denetleyiciler
- KDS** : Kesir Dereceli Sistemler
- KDPID** : Kesir Dereceli PID
- K-PSK** : Kaotik Maskeli Faz Kaydırmalı Anahtarlama
- PSO** : Parçacık Sürü Eniyilemesi
- PID** : Oransal Integral ve Türev Tipi Denetleyici

ÖZET

Kesir dereceli modelleme günümüzde sistemlerin modellenmesinde klasik yöntemlerden daha gerçekçi bir yöntemdir. Özellikle, kontrol mühendisliği alanında sistemlerin kesir dereceli olarak modellenmesi ve tasarımı klasik tam sayı dereceli modellemelerden daha faydalı sonuçlar elde edilmesine olanak sağlamaktadır. Bu çalışmada, son yıllarda adından sıkça söz edilen kesir dereceli PID (KDPID) denetleyicisinin, parametre eniyilemesi yapılarak yine kesir dereceli kaotik sistemlerin senkronizasyon denetiminde başarımı incelenmiştir. Ayrıca, KDPID denetleyicisinin ve kesir dereceli (KD) kaotik sistemlerinin devre modelleri oluşturulmuş, bu denetleyici ve kaotik devrelerin uygulanabilirliği gösterilmiştir. Denetleyicinin parametre ve kesir dereceleri, bulanık uyarlamalı PSO yöntemiyle belirlenerek denetleyicinin performansı artırılmıştır. KD kaotik sistemlerin senkronizasyonunun özellikle haberleşme sistemlerinde kullanılabilir olduğu, literatürdeki birçok araştırmada görülmektedir. Ancak, senkronizasyon hızı bu sistemler için oldukça önemlidir. Çünkü senkronizasyon ne kadar hızlı olursa, veri kaybı da o kadar az olur. Bu nedenle KDPID ile yapılan senkronizasyon kontrolü hem dış etmenlere karşı daha dayanıklı olup, hem de senkronizasyon süresi oldukça kısa olan bir denetim imkanı sağlamaktadır. Bu sayede haberleşmede SNR/BER oranına göre gürültü dayanımı açısından oldukça iyi bir performans sağlandığı görülmektedir. Bunun yanında senkronizasyona ulaşma süresine göre seçilen bit süresi kısa olması nedeniyle veri iletim hızı artırılabilir.

Anahtar Kelimeler: Kesir dereceli kaotik sistemler ve senkronizasyonu, kesir dereceli PID, Bulanık uyarlamalı parçacık sürü eniyilemesi (FAPSO), kaotik maskeleme.

SUMMARY

APPLICATION OF FRACTIONAL ORDER CONTROLLERS TO CHAOS BASED COMMUNICATION SYSTEMS BY FUZZY ADAPTIVE PARTICLE SWARM OPTIMIZATION

Fractional order modeling is nowadays a more realistic method than conventional methods in the system modeling. Particularly in the field of control engineering, systems modelling and designing with fractional order provides more useful results than conventional integer order modellings. In this thesis, the success of the fractional order PID(KDPID) controller, which was cited frequently in recent years, has been investigated on the control of the synchronization of fractional order chaotic systems by optimizing parameters. Besides, the circuit models of the KDPID controller and KD chaotic systems have been constituted, and it has been showed that these controllers and chaotic systems are applicable. The performance of controller has been increased by determining parameter and fractional order of the controller with fuzzy adaptive PSO method. In the literature, it has been seen that KD chaotic systems synchronization are applicable especially in communication systems. But synchronization speed is quite important for these studies. Because the quicker the synchronization, the less data lose. Therefore, synchronization control with KDPID method is both more robust against external factors and provides less synchronization time. Thus, according to SNR/BER rate the high communication noise performance is obtained. Besides, data rate can be increased because of the short bit time chosen according to arrival time to synchronization.

Keywords: Fractional Order Chaotic Systems and Synchronization, Fractional Order PID, Fuzzy Adaptive Particle Swarm Optimization (FAPSO), Chaotic Masking.

1. GİRİŞ

1.1. Kesir Dereceli Sistemler

Kesir Dereceli Sistemler (KDS), sistem derecesi herhangi bir reel sayı olan sistemlerdir. KDS ilk olarak 1695 yılında Leibniz'in L'Hospital'e göndermiş olduğu mektupla ortaya çıkmıştır. Leibniz mektubunda;

“tam dereceli türev için yapılan genellemeleri, tam dereceli olmayan sistemler için de yapabilir miyiz? Yani Sistem derecesi $\frac{1}{2}$ olduğunda ne olur?”

Sorusu üzerine L'Hospital;

“Bu bir paradokstur ve ileride yararlı sonuçları olacaktır”

cevabıyla karşılık vermiş ve bu mektup sonrası KDS üzerine bir çok araştırmacı tarafından teorik çalışmalar yapılmıştır. Bu çalışmaların birçoğu KDS'lerin çözüm yaklaşımlarının geliştirilmesi üzerine olmuştur [1-3]. Literatürde hala güncelliğini koruyan çözüm yaklaşımları Caputo, Liouville-Riemann, Grünwald-Letnikov gibi bilim insanlarının yaptığı çalışmalardır. Sonraki yıllarda yapılan çalışmalarda sistemlerin kesir dereceli (KD) olarak modellenmesinin daha başarılı sonuçlar verdiği gözlenmiş daha sonra fizik ve mühendislik alanında ilk uygulama örnekleri görülmeye başlanmıştır. Uygulama alanındaki ilk çalışmalar 1900'lü yılların ikinci yarısından itibaren başlamış ve genellikle denetim sistemleri üzerinde yoğunlaşmıştır [4-8]. Bu alanda öncü olarak kabul edilen Manabe, KD'li integralin denetim sistemleri üzerinde etkisi incelemiş [9] ve 90'lı yılların sonunda ise Podlubny tarafından kesir dereceli PID (KDPID) denetleyicisi üzerine çalışmalar yapmıştır [8]. KDS'lerin kararlılığı, KDPID denetleyicilerinin parametrelerinin belirlenmesi ve performansının değerlendirilmesi gibi konular ise literatürde güncelliğini korumaktadır [10-20].

Matignon tarafından yapılan çalışmada KDS'lerin kararlılık bölgesinin klasik sistemlere göre daha geniş olduğu vurgulanmış ve sistem derecesinin üçün altında olması durumunda kaotik davranış sergilediğini göstermiştir [21-22].

KD'li denetleyiciler klasik denetim sistemlerinden daha fazla parametre barındırdığı için hesaplanmada daha karmaşık teknikler kullanılır. Bu nedenle son yıllarda bu parametrelerin belirlenmesinde akıllı sistem temelli yaklaşımlar kullanılmaktadır [16-19].

1.2. Kaotik Sistemler ve Uygulamaları

Bilindiği üzere kaotik sistemler, düzensiz yapıya sahip gibi gözüken ancak kendi içinde düzeni olan sistemlerdir. Genellikle gürültü ile karıştırılan kaos, ilk olarak Eduard N. Lorenz tarafından hava tahmini için ortaya koyduğu sistemle ortaya çıkmıştır [23]. Sonraki yıllarda farklı kaotik sistemlerin varlığı üzerine araştırmalar sürmüştür ve farklı kaotik yapılar ortaya konmuştur. Literatürde Lorenz, Sprott, Chua v.d. gibi birçok kaotik sistem mevcuttur. Bu sistemler ekonomi, kimya, meteoroloji, biyoloji gibi alanlarda benzer davranış sergilediği, haberleşme sistemlerinde de kullanımının yararlı olabileceği görülmüştür. Kaotik sistemler başlangıç durumuna aşırı duyarlı sistemler olup, farklı başlangıç şartlarında farklı sinyaller üretmekte ve periyodik olmayan davranışlar sergilemektedir. Kaotik sistemler lineer olmayan davranışa sahip sistemlerdir. Doğada var olan sistemlerin birçoğu lineer olmayan davranışa sahip olduğundan, birçok yapı kaotik davranış sergilemektedir.

Günümüzde KD'li matematiğin gelişmesiyle KD kaotik sistemlerin modellenmesi, kontrolü ve senkronizasyonu üzerine çalışmaların sayısı artmıştır. KD'li kaotik sistemler içinde KD'li ifade bulunduran ve kaotik davranış sergileyen sistemler olarak bilinmektedir. Literatürde daha önceden tanımlanmış kaotik sistemlerin KD'li olarak yeniden modellenmesinin yanı sıra bu sistemlerin senkronizasyonu ve denetimi üzerine çalışmalar yapılmaktadır [24-28].

Devreler ve sistemler alanında ise bu kaotik sistemlerin devre modelleri, iki özdeş kaotik sistemin senkronizasyonu ve denetimi üzerine çalışmalar yapılmıştır [29]. Son yıllarda farklı kaotik devre modellerinin oluşturulması üzerine çalışmalar yapılmaktadır [30]. Kaotik sistemler, Haberleşme alanında bilgi güvenliği için kullanılması, yayılı spektruma sahip sistemlerin uygulanması gibi alanlarda karşımıza çıkmaktadır [31-33].

Kaotik sistemlerin en yaygın kullanıldığı yerlerden biri haberleşme sistemleridir. Haberleşme sistemlerinde kaosun kullanılması 90'lı yılların başında gerçekleşmiş ve bu

alandaki gelişimini sürdürmektedir [34-35]. Kaotik haberleşme sistemleri kaotik maskeleye, kaotik modülasyon ve kaotik şifreli sistemler olarak üç farklı yapıya sahiptir.

Kaotik maskeleye yönteminde, veri kaotik sistemle toplanarak haberleşme kanalına gönderilir [36]. Bu tip sistemlerde alıcı ve vericide bulunan kaotik sistemlerin senkronize olması gerekmektedir. Kaotik haberleşme sistemleri maskeleye yönteminin dışında kaotik modülasyon, kaotik şifreli sistemler ve kaotik sayısal haberleşme sistemleri olarak farklı yönlerde gelişimini sürdürmüştür. Kaotik sayısal haberleşme sistemleri ise kaotik açma-kapama (COOK) [37], kaotik kaydırmalı anahtarlama (CSK) [38], farksal kaotik kaydırmalı anahtarlama (DCSK) [39], korelasyon gecikmeli kaotik kaydırmalı anahtarlama (CDSK) [40] başlıkları altında sıralanabilir. Bunun yanında kaotik maskeli faz kaydırmalı anahtarlama yöntemi de literatürde kullanılmaktadır [41].

1.3. Tezin Amacı

Bu tezde, KD denetleyici ile yine KD temelli iki kaotik osilatörün senkronizasyonu gerçekleştirilerek sistemin performansı değerlendirilecektir. Bu KD'li kaotik osilatörlerin, kaotik maskeli haberleşme sistemi üzerinde sağladığı avantajlar incelenerek, gerçek bir sistemde denetleyicinin başarımı değerlendirilecektir.

Denetleyicinin parametrelerinin belirlenmesi için literatürdeki çalışmalardan farklı olarak, hızlı bir eniyileme yöntemi olan bulanık uyarlamalı parçacık sürü eniyileme (FAPSO) yöntemi kullanıldı. Bu eniyileme yönteminde klasik eniyileme yöntemlerinden farklı olarak eniyileme algoritmasında bulunan parametreler, bulanık mantık yaklaşımına göre uyarlamalı olarak değişmektedir. Bu şekilde eniyilemenin hızlı bir şekilde gerçekleşeceği düşünülmektedir.

Tezin amacı kapsamında yapılanlar kısaca özetlenecek olursa;

- i) KD'li iki kaotik osilatör ve bu osilatörlerde kaosun var olabileceğini ispatlamak için Matignon yönteminin kullanılması,
- ii) Kaotik osilatörlerin senkronizasyonu için KDD tasarlanıp iki osilatörün senkronizasyonunun sağlanması,
- iii) KD kaotik sistemlerin ve KDD devre modelinin oluşturulmaları,

- iv) Denetleyici parametrelerinin ayarlanmasında, literatürde yeni bir yaklaşım olan FAPSO için gerekli algoritmanın oluşturması,
 - v) Denetleyicinin performansını farklı denetleyicilerle karşılaştırıp başarımının incelenmesi,
 - vi) Bu sistemin kaotik maskeli haberleşme sistemine uygulanarak, elde edilen sonuçların literatürde yapılan veya kullanılan denetim yöntemleriyle karşılaştırılması,
- olarak verilebilir.

Bu işlemleri gerçekleştirmekteki amaç, dayanıklı bir senkronizasyon denetimi sağlamanın yanında, bunun kaotik senkronizasyon temelli haberleşme sistemi gibi doğrusal olmayan sistemlerde başarıyı arttırmaktır.

1.4. Tezin Kapsamı

Bu tez çalışmasında, KD'li kaotik sistemler, KDPID tipi denetleyiciler, bu denetleyicilerin parametrelerinin eniyilemesinde kullanılan eniyileme yöntemi olan FAPSO ve KD'li kaotik sistemlerin ve KDPID denetleyicisinin uygulandığı kaotik haberleşme sistemleri incelenmiştir. Tezin giriş kısmından sonraki bölümleri aşağıdaki gibi planlanmıştır.

2 Bölüm: Kesir Dereceli Sistemler

Bu bölümde KD'li sistemlerin çözüm yöntemleri olan Caputo, Grünwald-Letnikov, Riemann-Liouville tanımları yapılarak, KDS'lerin Laplace dönüşüm yöntemine değinilmiştir. Ayrıca KD'li sistemlerin devre modelleri incelenmiştir.

3. Bölüm: Kesir Dereceli Kaotik Sistemler ve Senkronizasyon

Bu bölümde, kaotik sistemlerin yapısı, Lorenz, Sprott ve limit çevrim davranışı gösteren Van der Pol osilatör yapılarına değinilerek bu sistemlerin KD'li olarak modellenmesi anlatılmıştır. Ayrıca, bu sistemlerin senkronizasyon denetimi, klasik PID ve KDPID yöntemiyle yapılarak başarımları incelenmiştir. Hem Simulink hem de devre modelleri elde edilerek yapılan çalışmanın doğruluğu karşılaştırılmıştır.

4. Bölüm: Bulanık Uyarlamalı Parçacık Sürü Eniyileme Yöntemi

Bu bölümde, önce klasik PSO yöntemi anlatılarak, bu yönteme yeni yaklaşımlar üzerinde durulmuştur sonrasında literatürde yeni bir yöntem olan FAPSO yönteminin avantajları ve KDPID denetleyicisinin parametre ve derecelerinin eniyilmesi gibi adımlar anlatılmıştır.

5. Bölüm: Kaotik Haberleşme Sistemleri

Bu bölümde, kaotik maskeli analog ve sayısal haberleşme sistemlerinin yapıları verilmiştir. FAPSO yöntemi ile parametre ve dereceleri eniylenen KDPID ve klasik PID denetleyicisiyle yapılan kaotik haberleşme sisteminin performansı test edilmiştir.

6. Bölüm: Tartışma, Sonuç ve Öneriler

Bu bölümde, elde edilen bulgular değerlendirilmiş literatürde yapılan çalışmalarla elde edilen sonuçlar karşılaştırmalı olarak verilmiştir. Son olarak, sonuç ve öneriler sunularak gelecekte yapılabilecek çalışmalar sıralanmıştır.

2. KESİR DERECELİ SİSTEMLER

2.1. Kesir Dereceli Sistemlerin Tarihçesi

KDS, derecesi tamsayıdan farklı herhangi bir reel sayı olan sistemler olarak bilinir [1-2]. Bu sistemlerin varlığı 17. yy'da Alman filozof ve matematikçi olan Leibniz'in, Fransız asıllı matematikçi L'Hopital'a yazdığı mektupla ortaya çıktığı bilinmektedir. Sonraki yıllarda 1730 yılında Euler çalışmasını bu yönde yaparken, 1772'de yılında Lagrange, farklı operatörler için dolaylı olarak KD'li hesap yöntemi geliştirmiştir. 1812 yılında Laplace, 1819 yılında Lacroix, 1822'de Fourier bu alanda çalışmalarıyla KD'li hesaplama farklı çözüm yöntemleri gerçekleştirilmiştir. 1832 yılında Liouville, tanımı problemlerin çözümü için uygulanmıştır. 1867 yılında ise Grünwald, kesirli operatör üzerinde çalışmalar yapmıştır. Riemann, kesirli integral teorisini geliştirmiş ve bu yöntem Liouville yöntemiyle birlikte Riemann-Liouville çözüm yaklaşımı olarak bilinmektedir. Letnikov'un 1872 yılına kadar yaptığı çalışmalarla kesirli hesaplama farklı yaklaşım sunmuştur. Yine Letnikov'un yaptığı bu çalışmalar, Grünwald'ın çalışmalarıyla birlikte literatürde Grünwald-Letnikov çözüm yöntemi olarak bilinir. 1900-1990 yılları arası yine birçok bilim adamı kesirli hesaplama yöntemi üzerinde çalışmalar yapmıştır [2].

KD hesaplama yöntemleri bu şekilde gelişirken, KDS'lerin uygulamaları ancak 1990'lı yıllara doğru ortaya çıkmıştır. Gerçek sistemlerin tamsayı değil ancak kesirli olabileceği ortaya konulmaya başlanmıştır [3]. KDS uygulamaları ilk olarak Wang ve Podlubny tarafından gerçekleştirilmiş ve bu yıllardan sonra KDS'ler birçok farklı alanda uygulamaları görülmektedir [8, 4, 42]. Günümüzde de KDS uygulamaları güncelliğini korumakla birlikte, yapılan çalışmalar da her geçen gün artmaktadır.

2.2. Kesir Dereceli Sistemler için Çözüm Yöntemleri

KDS'lerin tarihçesinde belirtildiği gibi birçok bilim adamının bu KDS çözüm yöntemleri konusunda çalışması bulunmaktadır. KD sistemlerin hesaplanmasında bazı özel

fonksiyonlar kullanılmaktadır. Bunlardan en temel olan Gamma fonksiyonu ve Mittag-Leffler fonksiyonudur. Gamma fonksiyonu faktöriyel işleminin tüm reel sayılar için tanımlanmış hali olarak bilinir ve bu fonksiyon Denklem (2.1)'de ifade edildiği gibidir.

$$\Gamma(\alpha) = \begin{cases} \int_0^{\infty} t^{\alpha-1} e^{-t} dt, & \alpha \in \mathbb{R} \\ (\alpha-1)!, & \alpha \in \mathbb{N} \end{cases} \quad (2.1)$$

Bir diğer fonksiyon olan Mittag-Leffler fonksiyonu da Denklem (2.2)'de verilmiştir.

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + \beta)}, \quad \alpha > 0, \beta > 0 \quad (2.2)$$

Literatürde özellikle gamma fonksiyonundan yararlanılarak yapılan Caputo, Grünwald-Letnikov, Riemann-Liouville ve Laplace dönüşümü gibi KDS'lerin çözüm yaklaşımları mevcuttur [43-44].

2.2.1. Caputo Yöntemi

KDS'lerin analizi için kullanılan yöntemlerden biri Caputo yöntemidir. İlk olarak İtalyan matematikçi Caputo tarafından 1967 yılında ortaya konan bu yöntem, Denklem (2.3)'de verilmiştir.

$$D^{\alpha} f(t) = \begin{cases} \frac{1}{\Gamma(n-\alpha)} \int \frac{f^{(n)}(\tau)}{t-\tau} d\tau, & n-1 < \alpha < n \\ \frac{d^n}{dt^n} f(t), & \alpha = n \in \mathbb{N} \end{cases} \quad (2.3)$$

Caputo denklemi olarak bilinen bu ifade de α KD'yi ifade ederken, n ise α 'dan büyük en küçük tamsayıyı ifade etmektedir.

2.2.2. Grünwald-Letnikov Yöntemi

Bu yöntem, klasik türev yaklaşımının ayrık düzlemde KDS'e uyarlanmış halidir. Klasik türevde, 1. dereceden türev;

$$\frac{d}{dt} f(t) = f'(t) = \lim_{h \rightarrow 0} \frac{f(t) - f(t-h)}{h} \quad (2.4)$$

2. dereceden türev için;

$$\begin{aligned}
f''(t) &= \lim_{h \rightarrow 0} \frac{f'(t) - f'(t-h)}{h} \\
&= \lim_{h \rightarrow 0} \frac{1}{h} \left[\frac{f(t) - f(t-h)}{h} - \frac{f(t-h) - f(t-2h)}{h} \right] \\
&= \lim_{h \rightarrow 0} \frac{f(t) - 2f(t-h) + f(t-2h)}{h^2}
\end{aligned} \tag{2.5}$$

3. dereceden türev için;

$$\begin{aligned}
f'''(t) &= \lim_{h \rightarrow 0} \frac{f''(t) - f''(t-h)}{h} \\
&= \lim_{h \rightarrow 0} \frac{1}{h} \left[\frac{f(t) - 2f(t-h) + f(t-2h)}{h^2} - \frac{f(t-h) - 2f(t-2h) + f(t-3h)}{h^2} \right] \\
&= \lim_{h \rightarrow 0} \frac{f(t) - 3f(t-h) + 3f(t-2h) - f(t-3h)}{h^3}
\end{aligned} \tag{2.6}$$

n. dereceden türev, Denklem (2.4)'deki gibi tanımlanır.

$$\frac{d^n}{dt^n} f(t) = f^{(n)}(t) = \lim_{h \rightarrow 0} \frac{1}{h^n} \sum_{j=0}^n (-1)^j \binom{n}{j} f(t-jh) \tag{2.7}$$

Burada verilen $\binom{n}{j}$ ifadesi n'nin j'li kombinasyonunu ifade eder. Kombinasyon Denklem (2.8)'deki gibi ifade edilir.

$$\binom{n}{j} = \frac{n!}{(n-j)!} \tag{2.8}$$

Burada n ve j'nin reel olması durumunda faktöriyel işleminin tüm reel sayılar için genellenmiş fonksiyonu olan *gamma* fonksiyonu kullanılır.

Gamma fonksiyonu Denklem (2.9)'daki gibi tanımlanır.

$$\Gamma(z) = \int_0^{\infty} e^{-t} t^{z-1} dt \tag{2.9}$$

Kombinasyonun reel sayılarda tanımı Denklem (2.10)'de verilmiştir.

$$\binom{\alpha}{j} = \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \quad \alpha \in R \quad (2.10)$$

Denklem (2.7) ve (2.10)'den yararlanarak klasik türev yöntemi, KD türev işlemi için de Denklem (11)'deki gibi genellenmiş olur [3].

$$D_t^\alpha f(t) = \frac{d^\alpha}{dt^\alpha} f(t) = f^{(\alpha)}(t) = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{j=0}^{\infty} (-1)^j \left(\frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \right) f(t-jh) \quad (2.11)$$

2.2.3. Riemann-Liouville Yöntemi

Bu yönteme göre KD'li türev ve integral işlemi, Cauchy formülünün reel sayılara göre genellenmesiyle elde edilir [45]. Cauchy formülü Denklem (2.12)'de verilmiştir.

$$\int_a^x dx_1 \int_a^{x_1} dx_2 \int_a^{x_2} dx_3 \dots \int_a^{x_{n-1}} f(x_n) dx_n = \frac{1}{(n-1)!} \int_a^x \frac{f(t)}{(x-t)^{1-n}} dt \quad (2.12)$$

Burada, Gamma ifadesini kullanarak Denklem (2.9)'u düzenlersek Denklem (2.13) elde edilir.

$$\int_a^x dx_1 \int_a^{x_1} dx_2 \int_a^{x_2} dx_3 \dots \int_a^{x_{n-1}} f(x_n) dx_n = \frac{1}{\Gamma(n)} \int_a^x \frac{f(t)}{(x-t)^{1-n}} dt \quad (2.13)$$

Bu denklemi KDS'ler için genelleyecek olursak Denklem (2.14)'de verilen ifade elde edilir.

$$I_a^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_a^x \frac{f(x)}{(t-x)^{1-\alpha}} dx \quad (2.14)$$

2.2.4. Kesir Dereceli Sistemlerde Laplace Dönüşüm Yöntemi

KDS'lerin analizinde kullanılan bir diğer yöntem de Laplace dönüşüm yöntemidir [45]. Laplace dönüşüm yönteminde, KD türev işlemi Denklem (2.15)'de verilmiştir.

$$L\{D^q f(t)\} = L\left\{\frac{d^q f(t)}{dt^q}\right\} = s^q L\{f(t)\} - \sum_{k=0}^{n-1} s^k \left[\frac{d^{q-1-k} f(t)}{dt^{q-1-k}} \right]_{t=0} \quad (2.15)$$

Başlangıç durumu sıfır olduğunda ifade Denklem (2.16)'da verildiği gibi olur.

$$L\{D^q f(t)\} = L\left\{\frac{d^q f(t)}{dt^q}\right\} = s^q L\{f(t)\} \quad (2.16)$$

Laplace dönüşüm yönteminde integral işlemi, eğer başlangıç değeri sıfır alınırsa Denklem (2.17) gibi olur.

$$L\{I^q f(t)\} = L\left\{\frac{d^{-q} f(t)}{dt^{-q}}\right\} = s^{-q} L\{f(t)\} \quad (2.17)$$

KDS'nin Laplace dönüştürüldükten sonraki hali olan s^q 'nin hesaplanmasının zorluğu nedeniyle literatürde bu dönüşüm için çeşitli yaklaşıklık yöntemleri vardır. Bunlar Crone, Matsuda, Carlson, Tustin gibi yöntemler literatürde kullanılan yaklaşımlardandır [44, 46-47].

Crone Yaklaşımı: Fransızca tamsayı olmayan dayanıklı kontrol (Commande Robuste d'Ordre Non Entier) kelimelerinin baş harflerinin kısaltması olarak adlandırılan bu yöntem, literatürde en yaygın kullanılan yöntemlerden biridir. Crone yaklaşımına ait matematiksel ifadeler Denklem (2.18)'de verilmiştir.

$$C^q(s) = s^q \approx C_0 \left(\frac{1 + \frac{s}{\omega_l}}{1 + \frac{s}{\omega_h}} \right)^q \quad (2.18)$$

$$a \prec \omega_l \prec \omega_h$$

$$0 \prec C_0$$

Burada ω_l ve ω_h yaklaşımın geçerli olduğu frekans aralığının üst ve alt limitlerini, C_0 ise pozitif kazanç değerini göstermektedir.

Carlson Yaklaşımı: Carlson ve Halijak'ın 1964 yılında KDS için önerdiği yöntemdir. s^q dönüşümü Denklem (2.19) gibi transfer fonksiyonuna çevrilir [48].

$$C_i(s) = s^q = C_{i-1}(s) \left(\frac{(r-m)(C_{i-1}(s))^2 + (r+m)s}{(r-m)(C_{i-1}(s))^2 + (r+m)s} \right) \quad (2.19)$$

$$r = \frac{1}{q}, m = \frac{r}{2}, C_0(s) = 1$$

Matsuda Yöntemi: Bu yöntem irrasyonel fonksiyon yaklaşımı temellidir [49]. Matsuda yönteminde Denklem (2.20)'deki eşitliklerinden yararlanılarak sistem transfer fonksiyonuna dönüştürülür.

$$C(s) = a_0 + \frac{s-s_0}{a_1} + \frac{s-s_1}{a_2} + \frac{s-s_2}{a_3} + \dots \quad (2.20)$$

$$a_i = v_i(s_i), \quad v_0(s) = H(s) \quad v_{i+1}(s) = \frac{s-s_i}{v_i(s) - a_i}$$

Carlson, Crone ve Matsuda gibi yöntemler sürekli sistemler için olup bunların dışında ayrık sistem yaklaşımı temelli, çözüm yöntemleri bulunur. Tustin, Simpson gibi yöntemler bunların en yaygın olarak kullanılanlarıdır [47-50].

Tustin Yöntemi: Bu yöntemde s Laplace operatörünün yerine Denklem (2.21)'deki gibi $(\omega.z^{-1})$ ifadesi konularak sistem ayrık sisteme dönüştürülür.

$$(s)^{\pm q} = (\omega(z^{-1}))^{\pm q} = \left(\frac{2}{T} \frac{1-z^{-1}}{1+z^{-1}} \right)^{\pm q} \quad (2.21)$$

Burada $0 < q < 1$ olup çözüm için Denklem (2.22)'daki ifadelerinden yararlanılarak hesaplanır [49].

$$s^q = \left(\frac{2}{T} \right)^q \lim_{n \rightarrow \infty} \frac{A_n(z^{-1}, q)}{A_n(z^{-1}, q)}, \quad q \in [-1, 1]$$

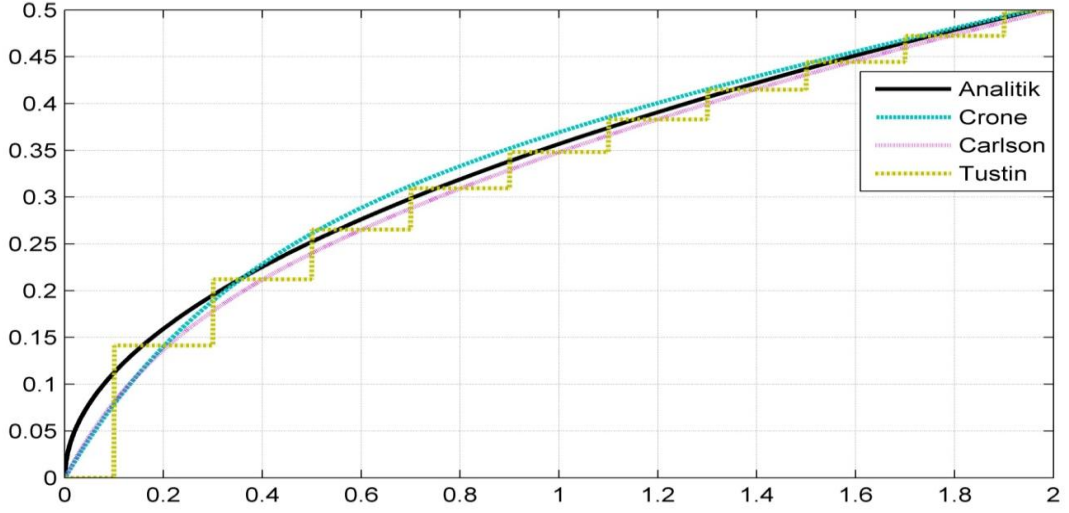
$$A_n = A_{n-1}(z^{-1}, q) + c_n z^n A_{n-1}(z^{-1}, q) \quad (2.22)$$

$$A_0(z^{-1}, q) = 1, \quad c_n = \begin{cases} \frac{q}{n}, & n; \text{tek} \\ 0, & n; \text{çift} \end{cases}$$

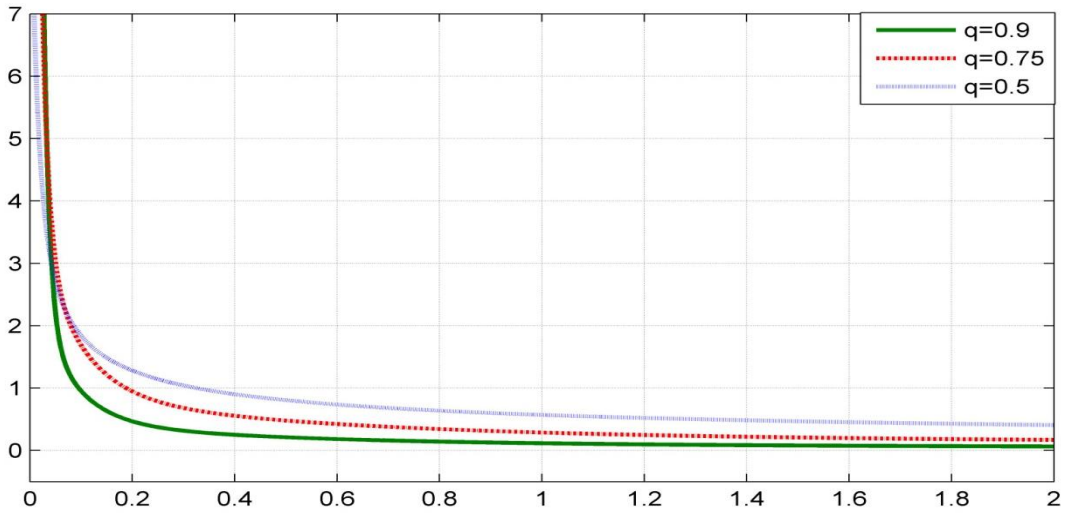
Simpson Yöntemi: Bu yöntemde de s ifadesinin z dönüşümü yapılarak Taylor serisine açılır. Dönüşüm, Denklem (2.23)'de verilen ifade yardımıyla yapılır [50].

$$C(s) = s^q = \left(\frac{3}{T} \frac{(1-z^{-1})(1+z^{-1})}{1+4z^{-1}+z^{-2}} \right)^q \quad (2.23)$$

Sürekli ve ayrık zaman için geliştirilen bu yöntemler, $f(t)=t$ fonksiyonunun 0.5 dereceden integralinin analitik çözümü Crone, Carlson ve Tustin gibi yöntemlerine göre analiz edilerek Şekil 2.1’de gösterilmiştir. Bu şekilden de görüldüğü gibi Carlson, Crone ve Tustin yöntemi analitik çözüm yaklaşımına yakın sonuç göstermektedir. Tustin yöntemi ayrık sistem olduğu için hatanın periyoda göre farklılık gösterdiği görülmektedir.

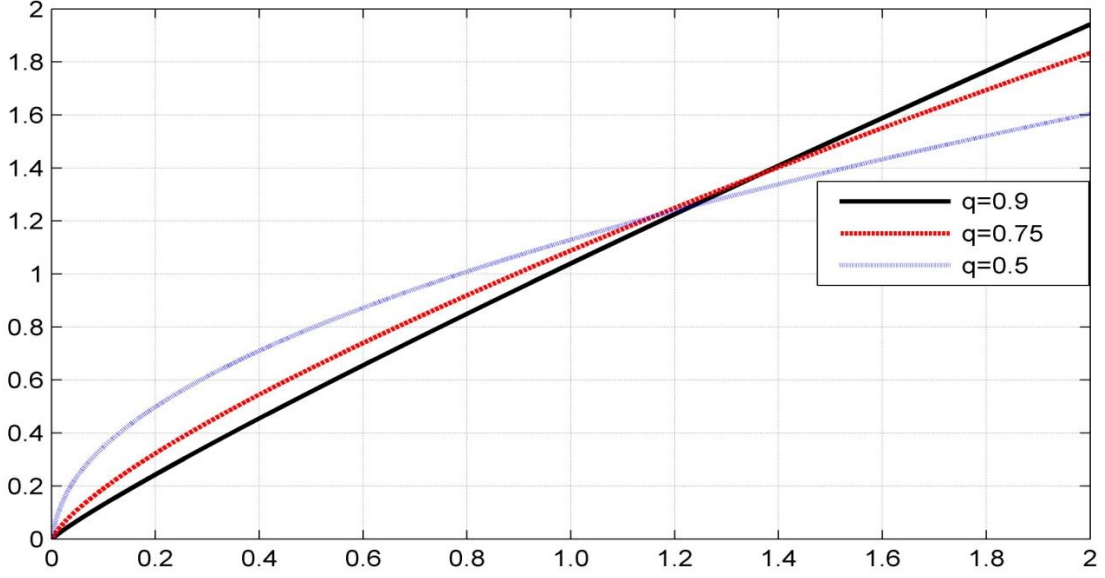


Şekil 2.1. Farklı çözüm yaklaşımlarına göre $f(t)=t$ fonksiyonunun 0.5 dereceden integrali



Şekil 2.2. Crone yaklaşımına göre birim basamak fonksiyonunun farklı KD'ye sahip türev işlemi [51]

Crone yöntemine göre farklı KD'ne sahip türev ve integral işlemlerinin sonucu, Şekil 2.2 ve Şekil 2.3'de gösterilmiştir [51]. Bu şekillerden farklı KD'ye sahip ifadelerin birim basamak cevaplarının etkileri görülmektedir.



Şekil 2.3. Crone yaklaşımına göre birim basamak fonksiyonunun farklı KD'ye sahip integral işlemi

2.3. Kesir Dereceli Sistemlerin Devre Modelleri

KD sistemlerin devre modellerinin oluşturulması için birçok bilim adamı, Ahmed Wajidi v.d. tarafından oluşturulan KD integratörün transfer fonksiyonu modelini referans alır. Kullanılan dönüşüm modeli, Tablo 2.1'de verilmiş olup, farklı KD'ye sahip bir integratörlerin transfer fonksiyonu modeli gösterilmektedir [52]. KD devre modeli için literatürde farklı yaklaşımlar mevcuttur. Bunlar;

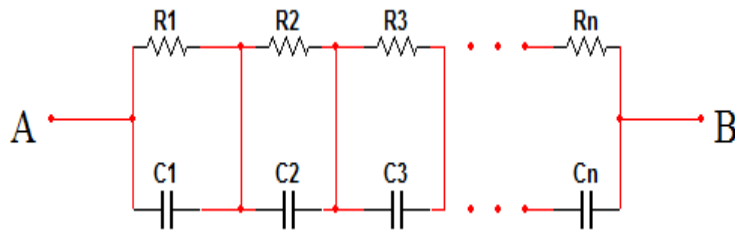
- Diyi Chen v.d. oluşturduğu model
- Dorčák v.d. tarafından oluşturulan model
- Chen Xiang-Rong'ın oluşturduğu model

Tablo 2.1. KD integratörünün transfer fonksiyonu [52]

q	G(s)
0.1	$\frac{1584.8932(s+0.1668)(s+27.83)}{(s+0.1)(s+16.68)(s+2783)}$
0.2	$\frac{79.4328(s+0.05623)(s+1)(s+17.78)}{(s+0.03162)(s+0.5623)(s+10)(s+177.8)}$
0.3	$\frac{39.8107(s+0.0416)(s+0.3728)(s+3.34)(s+29.94)}{(s+0.02154)(s+0.1931)(s+1.73)(s+15.51)(s+138.9)}$
0.4	$\frac{35.4813(s+0.03831)(s+0.261)(s+1.778)(s+12.12)(s+82.54)}{(s+0.01778)(s+0.1212)(s+0.8254)(s+5.623)(s+3831)(s+261)}$
0.5	$\frac{15.8489(s+0.03981)(s+0.2512)(s+1.585)(s+10)(s+63.1)}{(s+0.01585)(s+0.1)(s+0.631)(s+3.981)(s+25.12)(s+158.5)}$
0.6	$\frac{10.7978(s+0.04642)(s+0.3162)(s+2.154)(s+14.68)(s+100)}{(s+0.01468)(s+0.1)(s+0.6813)(s+4.642)(s+31.62)(s+215.4)}$
0.7	$\frac{9.3633(s+0.06449)(s+0.578)(s+5.179)(s+46.42)(s+416)}{(s+0.01389)(s+0.1245)(s+1.116)(s+10)(s+89.62)(s+803.1)}$
0.8	$\frac{5.3088(s+0.1334)(s+2.371)(s+42.17)(s+749.9)}{(s+0.01334)(s+0.2371)(s+4.217)(s+74.99)(s+1334)}$
0.9	$\frac{2.2675(s+1.292)(s+215.4)}{(s+0.01292)(s+2.154)(s+359.4)}$

2.3.1. Diyi Chen Modeli

KD devre modelinde ilk olarak Diyi Chen v.d. tarafından ortaya konan bu yöntem Şekil 2.4’de gösterildiği gibi, paralel bağlı direnç ve kondansatör grubunun birbirlerine seri bağlanmasıyla oluşturulmuştur. Burada devreye ait direnç ve kapasite değerleri, kesir derecesine göre değişir. Bu modele ait matematiksel ifade Denklem (2.24)’de verilmiştir [53].



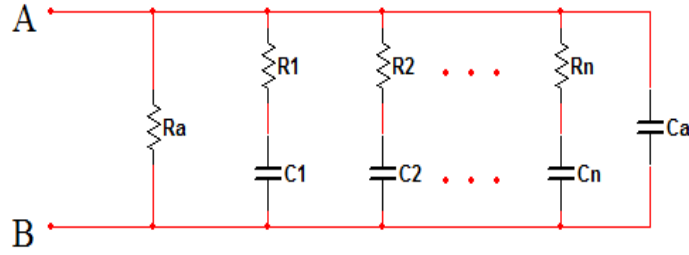
Şekil 2.4. Diyi Chen v.d. tarafından oluşturulan KD devre modeli

$$G(s) = \sum_{n=1}^{\infty} \frac{R_n}{1 + sR_n C_n} \quad (2.24)$$

2.3.2. Dorčák Modeli

Bu modelde birbirine seri bağı direnç ve kondansatör ikili gruplar halinde birbirine paralel bağlanır, ek olarak bir direnç ve bir kondansatör de bu yapıya paralel bağlanır. Bu yöntemle göre devre Şekil 2.5’de gösterilmiştir [54]. Dorčák modeline ait matematiksel ifade Denklem (2.25)’de verilmiştir.

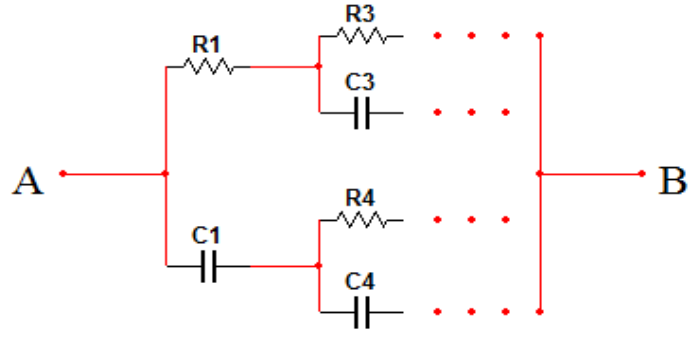
$$G(s) = \frac{1}{\frac{1 + sR_a C_a}{R_a} + \sum_{n=1}^{\infty} \frac{sC_n}{1 + sR_n C_n}} \quad (2.25)$$



Şekil 2.5. Lubomír Dorčák v.d. tarafından oluşturulan devre modeli

2.3.3. Xiang-Rong Modeli

Bu model, direnç ve kondansatörlerin birbirine Şekil 2.6’da gösterildiği gibi bağlanmasıyla oluşturulmuştur. İlk olarak Xian Rong v.d. tarafından oluşturulan bu modele ait matematiksel ifade Denklem (2.26)’da verilmiştir [55].



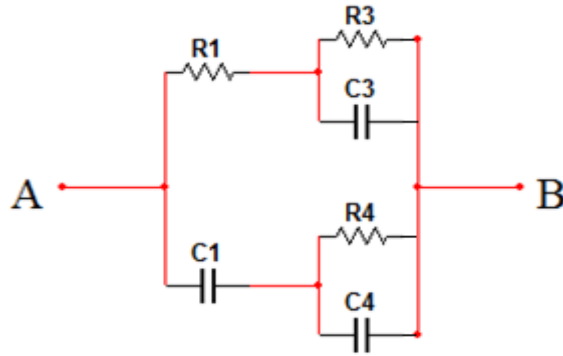
Şekil 2.6. Chen Xiang-Rong v.d. tarafından oluşturulan devre modeli

$$G(s) = \frac{s(R_1 Z_1 C_1 + Z_1 Z_2 C_1) + R_1 + Z_1}{sC_1(R_1 + Z_1 + Z_2) + 1}$$

$$Z_1 = \frac{s(R_2 Z_3 C_2 + Z_3 Z_4 C_2) + R_2 + Z_3}{sC_2(R_2 + Z_3 + Z_4) + 1}, \quad (2.26)$$

$$Z_2 = \frac{s(R_3 Z_5 C_3 + Z_5 Z_6 C_3) + R_3 + Z_5}{sC_3(R_3 + Z_5 + Z_6) + 1}, \dots$$

Bu devre modelini Şekil 2.7’de verildiği gibi kabul edersek, devrenin matematiksel ifadesi Denklem (2.27)’de verildiği gibi olur.



Şekil 2.7. n=3 boyutunda bir devre modeli

$$G(s) = \frac{C_0 \left(\frac{C_1 + C_3}{C_1 C_3} \right) \left(s + \frac{R_1 + R_2}{C_2 R_1 R_2} \right) \left(s + \frac{1}{(C_1 + C_3) R_3} \right)}{C_0 s^3 + C_0 \left(\frac{R_1 + R_2}{R_1 C_2 R_2} + \frac{1}{R_3 C_3} + \frac{C_1 + C_3}{R_1 C_1 C_3} \right) s^2 + \dots} \quad (2.27)$$

$$+ C_0 \left(\frac{R_1 + R_2}{R_1 R_2 R_3 C_2 C_3} + \frac{1}{R_1 R_3 C_1 C_3} + \frac{C_1 + C_3}{R_1 R_2 C_1 C_2 C_3} \right) s + C_0 \frac{1}{R_1 R_2 R_3 C_1 C_2 C_3}$$

Bu devrenin KD'ye göre, direnç ve kondansatör değerleri değişir, bu Tablo 2.2 ve Tablo 2.3'de gösterilmiştir.

Tablo 2.2. KD devre modeline ait direnç değerleri [55]

q	R ₁ (Ω)	R ₂ (Ω)	R ₃ (Ω)	R ₄ (Ω)	R ₅ (Ω)	R ₆ (Ω)
0.1	951.7k	633.2k	1.363M			
0.2	545.4k	755.3k	3.307M	1.211M		
0.3	347.6k	1296k	4.854M	2.339M	219.6k	
0.4	184.5k	429.4k	2.08M	5.698M	781.9k	1.961M
0.5	174.9k	441.7k	2.48M	9.41M	695.7k	2.354M
0.6	82.91k	320.2k	2.594M	15.45M	543.6k	2.853M
0.7	21.69k	120.5k	2.016M	24.99M	280.6k	3.247M
0.8	8.586k	92.69k	6.325M	39.69M	490k	
0.9	1.55M	61.54M	526k			

Tablo 2.3. KD devre modeline ait kondansatör değerleri [55]

q	C ₁ (μF)	C ₂ (μF)	C ₃ (nF)	C ₄ (μF)	C ₅ (nF)	C ₆ (μF)
0.1	0.02572	15.77	0.6468			
0.2	5.366	0.1814	12.62	1.56		
0.3	4.927	0.5417	25.25	2.752	309.8	
0.4	5.023	0.4836	28.34	0.1456	142.8	4.854
0.5	3.793	0.5827	64.16	0.1751	229.2	4.441
0.6	2.741	0.5489	95.85	0.1692	244.2	3.247
0.7	1.917	0.4657	11.31	0.15	196.4	2.019
0.8	0.9503	0.6139	23.49	0.2337	239.1	
0.9	0.7346	0.5221	1103			

2.4. Kesir Dereceli Denetleyiciler ve KDPID Tipi Denetleyiciler

KD sistemlerin mühendislik alanında ilk uygulaması denetleyici sistemlerinde olmuştur [2]. Bu sistemlerde bulunan türev ve integral operatörü klasik türev ve integral operatöründen ziyade KD türev ve integral temellidir. Literatürde KD kayan kip denetleyiciler [56-57] ve KDPID denetleyicileri üzerinde çalışmalar yapılmaktadır [8, 16, 51]. KDD olarak en yaygın olarak çalışılan yöntem, KDPID denetleyicileridir.

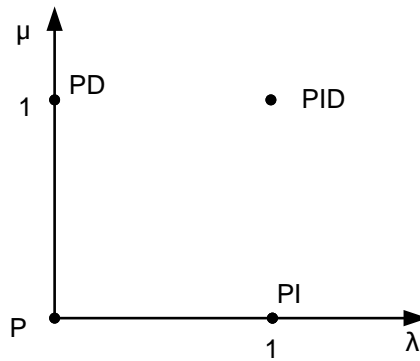
2.5. KDPID Tipi Denetleyiciler Yapısı

KDPID ilk olarak Podlubny tarafından ortaya konulmasından bugüne kadar, bu alanda birçok çalışma yapılmıştır. Bu çalışmalar genelde denetleyicinin gerçekleştirilmesi, KD çözüm yaklaşımlarına göre denetleyicinin analizi, kararlılığı, parametre ayarı gibi konular üzerinde yoğunlaşmıştır [58-65].

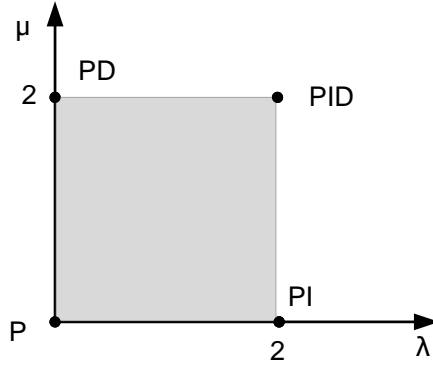
KDPID denetleyiciler hatanın oransal değeri, hatanın KD integralinin değeri ve yine hatanın KD türev değerinin toplamı (Denklem (2.28)) prensibine göre çıkış üretir [8].

$$C(s) = \frac{U(s)}{E(s)} = Kp + \frac{1}{T_i} s^{-\lambda} + T_d s^\mu = Kp + K_i s^{-\lambda} + T_d s^\mu \quad (2.28)$$

Transfer fonksiyonunda bulunan katsayılarla KD denetleyicilerinin, PI^λ , PD^μ veya $PI^\lambda D^\mu$ gibi kombinasyonları elde edilebilir. Buradaki transfer fonksiyonu, Şekil 2.8'de gösterildiği gibi (klasik PID için) noktalarda tanımlı değil, Şekil 2.9'da gösterildiği gibi yüzey üzerinde tanımlıdır.



Şekil 2.8. Klasik P, PI, PD ve PID denetleyicisinin tanımlı olduğu noktalar



Şekil 2.9. PI^λ , PD^μ veya $PI^\lambda D^\mu$ denetleyicisinin tanımlı olduğu alan

KDPID denetleyicisinin Denklem (2.28)'de verilen transfer fonksiyonu düzenlenirse;

$$C(s) = \frac{T_d s^{\lambda+\mu} + K_p s^\mu + K_i}{s^\lambda} \quad (2.29)$$

formülü elde edilir.

2.6. KD Denetleyicilerin Parametrelerinin Akıllı Sistemlerle Belirlenmesi

Klasik PID yönteminde denetleyici parametreleri olan K_p , K_i ve K_d 'nin belirlenmesi için literatürde birçok farklı çalışma bulunmaktadır. Ancak KDPID tipi denetleyicilerinin parametre ve derecelerinin belirlenmesi için literatürde klasik PID yönteminden farklı olarak akıllı sistemler temelli çalışmalar bulunmaktadır [16-20]. Klasik PID yönteminde üç parametre belirlenirken, KDPID yönteminde bu üç parametrenin yanında ayrıca türev ve integral derecelerinin de belirlenmesi gereklidir [16]. Bu çalışmalardan birinde Denklem (2.30)'da verilen $G(s)$ kontrol nesnesinin denetiminde eniyi parametreleri ve derecelerini belirlemek için genetik algoritma yönteminden yararlanmıştır [66].

$$G(s) = \frac{400}{s^2 + 50s} \quad (2.30)$$

PSO yöntemiyle 2006 yılında yapılan eniyilemesinde elde edilen sonuçlar genetik algoritmayla karşılaştırıldığında PSO yönteminin daha iyi sonuçlar aldığı görülmüştür [67].

KD sistemler için kullanılan bir başka akıllı yöntem de farksal evrim yaklaşımıdır. Biswas v.d. tarafından yapılan bu çalışmada Denklem (2.31)'de verilen dört farklı denetleyici nesnesi üzerinde uygulanmış ve başarılı sonuçlar elde edildiği gözlenmiştir [18].

$$\begin{aligned}
 G_1(s) &= \frac{0.01}{(0.01s + 0.1)(0.5s + 1) + 0.0001} \\
 G_2(s) &= \frac{400}{s^2 + 50s} \\
 G_3(s) &= \frac{1}{0.8s^{2.2} + 0.5s^{0.9} + 1} \\
 G_4(s) &= \frac{1}{0.9s^{0.3} + 0.6s^{0.8} + 1}
 \end{aligned} \tag{2.31}$$

Lee & Chang tarafından 2010 yılında yapılan yine akıllı yöntemlerden yararlanılarak evrimsel hesaplama ve genetik algoritma yaklaşımının birleşimi olan bir başka çalışma yapılmıştır. Yazar çalışmasının sonuçlarını Cao v.d. yaptığı çalışma ile karşılaştırmıştır [63]. Sonuçlar parametrelerin bulunmasında oldukça iyi performans gösterdiği gözlenmiştir.

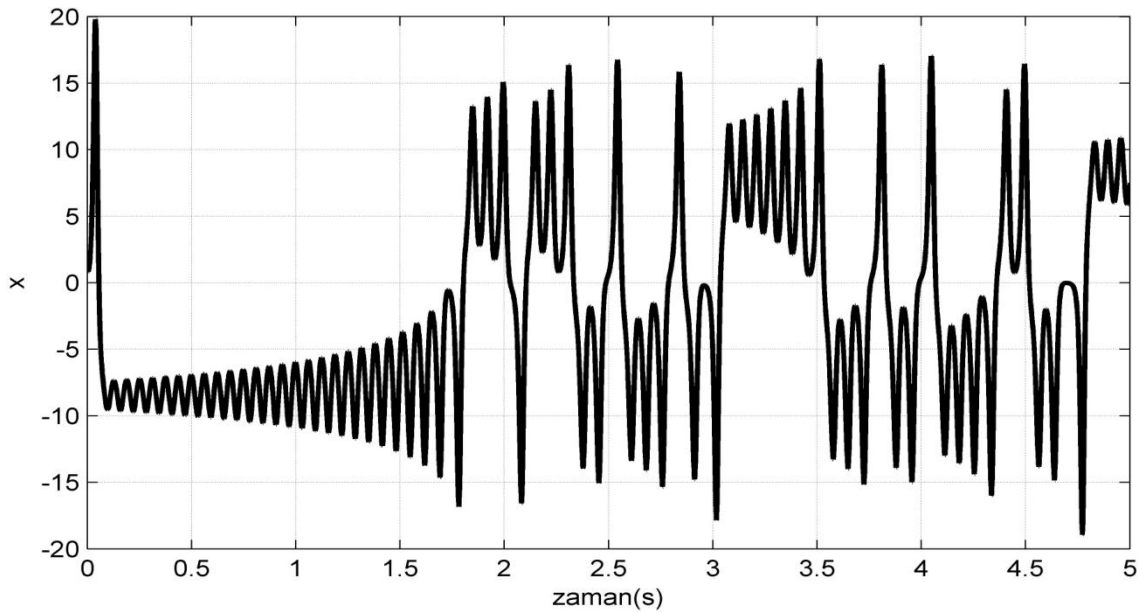
3. KESİR DERECELİ KAOTİK SİSTEMLER VE KAOTİK SENKRONİZASYON

3.1. Kaotik Sistemler

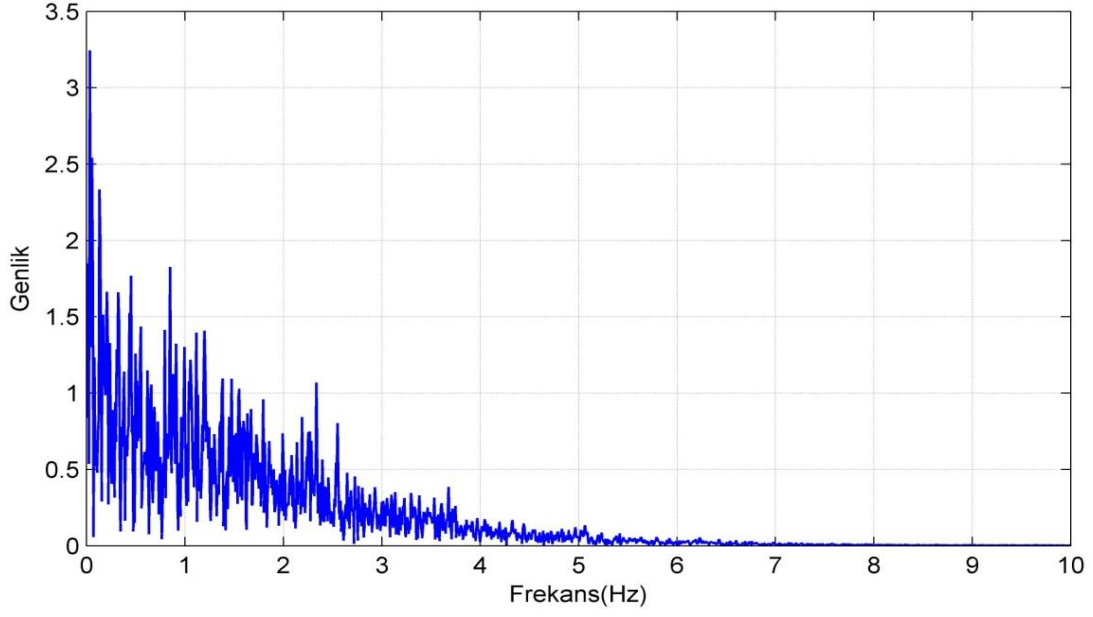
Kaos, ilk olarak antik Yunan filozofları tarafından evrenin kaos üzerine kurulduğu düşüncesiyle üzerine ortaya çıktığı bilinmektedir [68]. Gerçek anlamda varlığı, yüzyıllar sonra modern bilim tarafından ortaya konan, kaos başlangıç koşullarına aşırı bağımlı sistemler olarak bilinmektedir [69].

Kaotik sistemlerin diferansiyel denklem modeli, ilk olarak Lorenz tarafından doğada hava olaylarını modellemesinde kullanılmak üzere oluşturmuştur [23]. Lorenz'in hava olaylarının tahmini için bulduğu sistemin matematiksel modeli, Denklem (3.1)'de verilmiştir. Burada, $\sigma=10$, $r=8/3$ $b=28$ ve başlangıç şartı $(1, 0, 0)$ olarak alındığında oluşan kaotik işaret ve işaretin frekans spektrumu Şekil 3.1 ve Şekil 3.2'de gösterildiği gibidir.

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= -xz + rx - y \\ \dot{z} &= xy + bx\end{aligned}\tag{3.1}$$

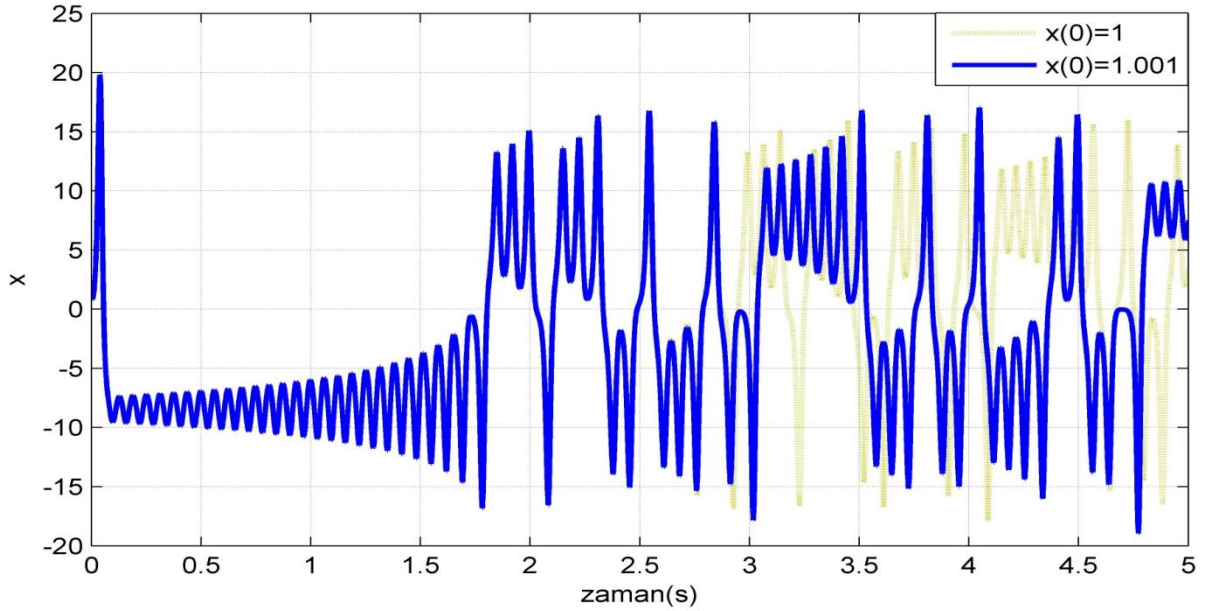


Şekil 3.1. x işaretinin değişimi $[x_0, y_0, z_0]=[1, 0, 0]$



Şekil 3.2. x işaretinin frekans spektrumu

Farklı başlangıç durumlarda kaotik işaretin değişimi incelendiğinde, işaretlerin farklı olduğu Şekil 3.3’de görülür.



Şekil 3.3. $[x_0, y_0, z_0]=[1 \ 0 \ 0]$ ve $[x_0, y_0, z_0]=[1.0001 \ 0 \ 0]$ başlangıç durumlarına göre x_1 işaretinin değişimi

3.2. Kesir Dereceli Kaotik Sistemler

KD sistemlerde kaosu varlığı üzerine ilk çalışmalar 2000'li yılların başında gerçekleştirilmiştir. Daha önceden kaosu varlığı ispatlanmış sistemlerde bu yıldan sonra KD olması durumunda kaosu varlığı ve bu kaotik sistemlerin senkronizasyonu üzerinde çalışmalar yapılmıştır [70-73].

Literatürde KD Chua, KD Volta, KD Rössler, KD Lü, KD Chen, KD Duffing, ve KD Van der Pol sistemlerinde kaosu varlığı, modellenmesi ve simülasyonu üzerine çalışmalar yapılmıştır [74-81]. Bu sistemlerin senkronizasyonu ve kontrolü araştırmacıların çalıştığı güncel konulardandır.

3.2.1. Kesir Dereceli Sistemlerde Kaosu Varlığı

Kaosu varlığı var olabilmesi için sistemin toplam derecesi üç olması gerekir ancak Matignon teoremine göre bu değer için altında olması durumunda da kaos oluşabileceği gözlenmiştir.

Teorem 3.1: KD'li bir sistem, Denklem (3.2)'de verilen şartı sağlarsa sistem kararlıdır [21-22].

$$|\arg(\text{eig}(J))| > \alpha \frac{\pi}{2} \quad (3.2)$$

Bu teoremden bulunan J sistemin Jakobian matrisi ve α ise kesir derecesinin değerini vermektedir. Bu teoreme göre kaosu oluşabilmesi için sistemin kararsız bölgede veya denge noktasında bulunması gereklidir [3, 82]. Yani kaosu oluşabilmesi için Denklem (3.3)'de verilen şartı sağlaması gereklidir.

$$|\arg(\lambda_i)| \leq \alpha \frac{\pi}{2} \quad (3.3)$$

Örnek 3.1.

KD Lorenz sistemine ait matematiksel ifade Denklem (3.4)'de verildiği şekildedir.

$$\begin{aligned}
\frac{d^q x(t)}{dt^q} &= \sigma [y(t) - x(t)], \\
\frac{d^q y(t)}{dt^q} &= x(t) [\rho - z(t)] - y(t), \\
\frac{d^q z(t)}{dt^q} &= x(t)y(t) - \beta z(t),
\end{aligned} \tag{3.4}$$

Burada $\sigma=10$, $\beta=8/3$ ve $\rho=28$ olarak alınmıştır. Sistemin Jakobian matrisi Denklem (3.5)'de verilmiştir.

$$J(A) = \begin{bmatrix} -10 & 10 & 0 \\ 28 - z^* & -1 & -x^* \\ y^* & x^* & -8/3 \end{bmatrix} \tag{3.5}$$

Bu sistemin üç denge noktası vardır. Bunlar Denklem (3.6)'da verildiği gibidir.

$$E^* = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 6\sqrt{2} & 6\sqrt{2} & 27 \\ -6\sqrt{2} & -6\sqrt{2} & 27 \end{bmatrix} \tag{3.6}$$

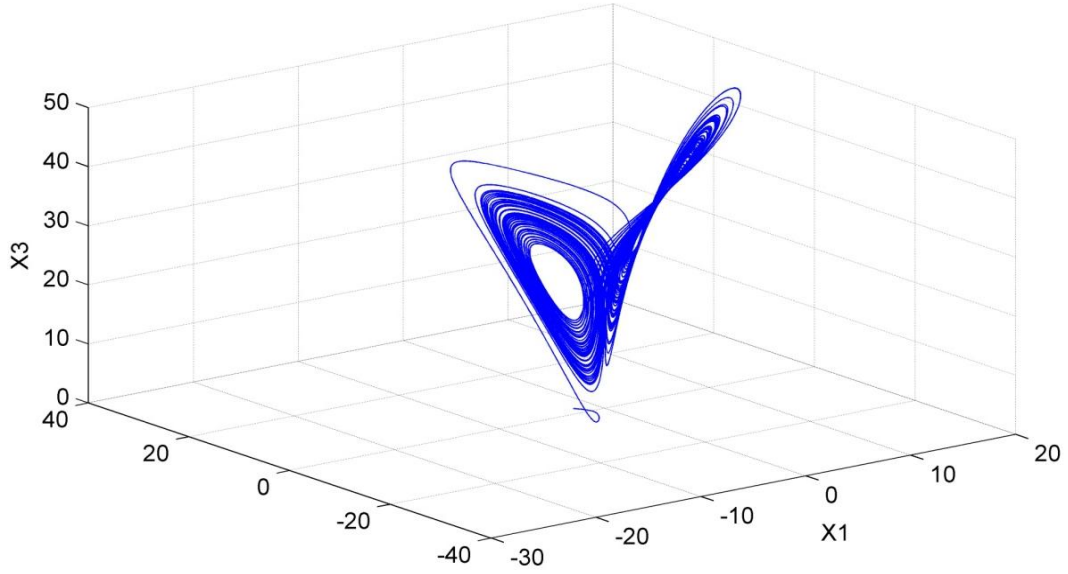
Denklem (3.5) ve Denklem (3.6)'da verilen ifadelerle göre sistemin özdeğerleri ise Denklem (3.7)'de verilmiştir.

$$\lambda = \begin{bmatrix} -13.8545 \\ 0.09339 + i10.1944 \\ 0.09339 - i10.1944 \end{bmatrix} \tag{3.7}$$

şeklinde. Sistemin, Teorem (3.1)'e göre KD hesaplanırsa Denklem (3.8) eşitsizliği elde edilir.

$$0.378 \leq q \tag{3.8}$$

Denklem (3.8) eşitsizliğinden de görüldüğü gibi sistemin kaos davranışı sergilemesi için eşitsizliğini sağlamalıdır. Burada kaosun varlığı $q=0.9$ için Şekil 3.4'de gösterilmiştir.



Şekil 3.4. $q=0.9$ için oluşan kaotik çeki

Örnek 3.2

KD'li çoklu hiperbolik çeker sistemine ait matematiksel ifade Denklem (3.9)'da verilmiştir [82-83].

$$\begin{aligned}
 \frac{d^q x}{dt^q} &= y \\
 \frac{d^q y}{dt^q} &= z \\
 \frac{d^q z}{dt^q} &= ax + by + cz - \alpha \frac{\tau}{2} \left[(r-s) + \sum_{i=-r}^s \tanh(x+i\tau) \right]
 \end{aligned} \tag{3.9}$$

Bu ifadede sabit değerler $a=b=c=\alpha=-0.65$, $r=s=3$ ve $\tau=100$ alınırsa sistemin matris şekli Denklem (3.10)'da verilmiştir.

$$A = \begin{bmatrix} 0 & y & 0 \\ 0 & 0 & z \\ -0.65x + 32.5f(x) & -0.65 & -0.65 \end{bmatrix} \tag{3.10}$$

$$f(x) = \sum_{i=-1}^1 \tanh(x+i\tau)$$

Bu ifadeden Denklem (3.10)'un Jakobian matrisi ($\tau=100$ için) Denklem (3.11) elde edilir.

$$J(A) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -0.65 + 32.5f'(x) & -0.65 & -0.65 \end{bmatrix} \quad (3.11)$$

$$f'(x) = 3 - \tanh^2(x-100) - \tanh^2(x+100)$$

Denklem (3.11)'in denge noktası Denklem (3.12)'deki gibi olarak elde edilir.

$$E_0 = [250 \ 0 \ 0] \quad (3.12)$$

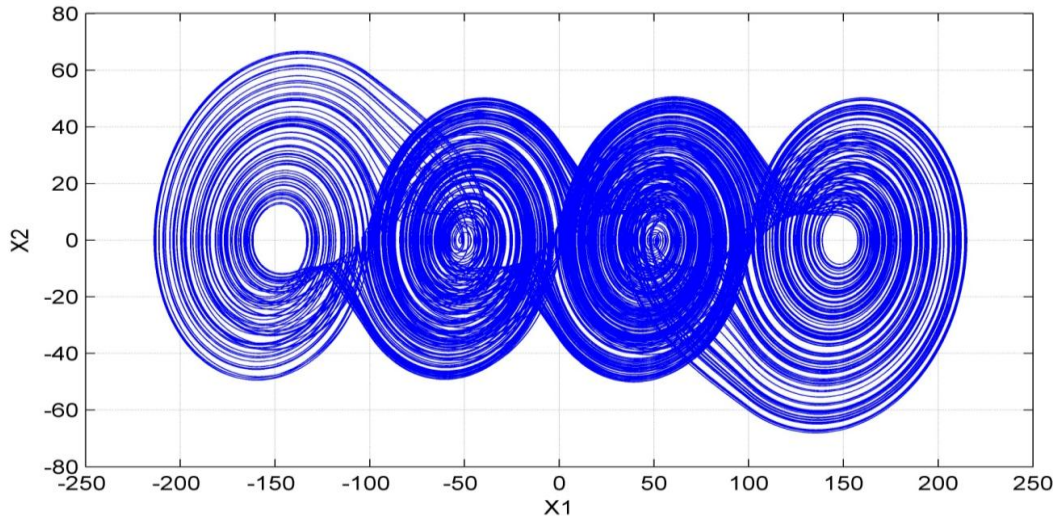
Bu denge noktalarına karşılık gelen özdeğerleri Denklem (3.13) elde edilir.

$$\lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \begin{bmatrix} 2.9037 \\ -1.7768 + i2.7949 \\ -1.7768 - i2.7949 \end{bmatrix} \quad (3.13)$$

Teorem 3.1'den sistemin kaos üretmesi için gerekli şartı hesaplırsak Denklem (3.14)'de verilen sonucu elde ederiz.

$$0.6395 \leq q \quad (3.14)$$

Burada $q=0.7$ olarak alınmıştır. Bu değerlere göre oluşan kaotik çeki Şekil 3.5'de gösterilmiştir.



Şekil 3.5. KD çoklu hiperbolik çeker sistemine ait kaotik çeki

3.3. Kaos Senkronizasyon

Farklı başlangıç koşullarına sahip özdeş iki kaotik sistemin aynı çıkış üretmeleri için senkronizasyon kontrolü gereklidir. İlk olarak Pecora and Carroll [84-85], tarafından gerçekleştirilen senkronizasyon işlemi sonraki yıllarda senkronizasyon üzerine yapılan farklı çalışmalarla sürmüştür. Literatürde bulunan diğer yöntemler; genelleştirilmiş senkronizasyon [86-89], tam senkronizasyon [90], lag senkronizasyon [91], faz senkronizasyonu [92], ve anti faz senkronizasyonu [93] gibi yöntemlerdir.

Senkronizasyon için n boyutlu Denklem (3.15)'de verilen doğrusal olmayan bir sistemi göz önüne alalım $u(x) = (u_1(x(t, x_0), \dots, x_k(x(t, x_0))))$ dir.

$$\begin{aligned} \dot{x} &= f(x) & \dot{y} &= g[y, u(x)] \\ x, y &\in R^n \end{aligned} \quad (3.15)$$

Burada senkronizasyon hatası Denklem (3.16)'da verilmiştir. Burada y ve y_H kaotik iki fonksiyondur.

$$e = \lim_{t \rightarrow \infty} \|y(t, y_0) - y_H(t, x_0)\| = 0 \quad (3.16)$$

Örnek 3.3:

Denklem (3.17)'de verilen Lorenz kaotik sisteminin Denklem (3.18)'deki sistem ile senkronize olması için Denklem (3.18)'e "u" denetleyici girişi eklenmiştir.

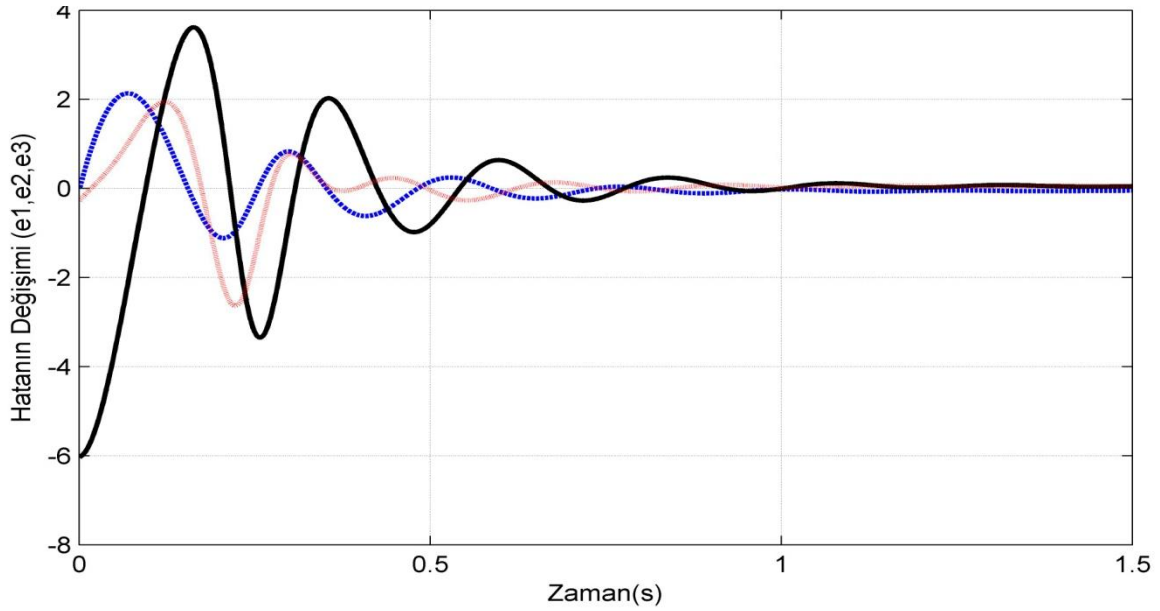
$$\begin{aligned} \dot{x}_1 &= 10(y_1 - x_1) \\ \dot{y}_1 &= -x_1 z_1 + 28x_1 - y_1 \\ \dot{z}_1 &= x_1 y_1 - \frac{8}{3} x_1 \end{aligned} \quad (3.17)$$

$$\begin{aligned} \dot{x}_2 &= 10(y_2 - x_2) \\ \dot{y}_2 &= -x_2 z_2 + 28x_2 - y_2 + u \\ \dot{z}_2 &= x_2 y_2 - \frac{8}{3} x_2 \end{aligned} \quad (3.18)$$

Burada "u" denetleyici girişinin değeri senkronizasyon hatasına göre değişir [76]. Senkronizasyon hatası Denklem (3.19)'da verildiği gibi hesaplanır.

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} x_1 - x_2 \\ y_1 - y_2 \\ z_1 - z_2 \end{bmatrix} \quad (3.19)$$

Denklem (3.16) ve (3.17)'de verilen ifadelerle göre senkronizasyon hatasının deęiřimi Őekil 3.6'da gsterilmiřtir. Burada klasik PID yntemiyle yapılan denetim sonucunda, Lorenz kaos osilatrnn deęiřkenleri olan x, y ve z deęiřkenlerinin hatasının (master ve slave kaotik sistemleri arasında ki farkın) zamanla azaldıęı gzlenmektedir. Ancak senkronizasyonda sistemin mmkn olduęunca hızlı bir Őekilde senkronize olması istenir.



Őekil 3. 6. Senkronizasyon hatasının (e_1 , e_2 ve e_3) deęiřimi

3.4. Kesir Dereceli Van der Pol Sistemi

İlk olarak Alman elektrik mhendisi ve fizikisi olan Balthasar Van der Pol tarafından keřfedilen bu osilatr sonrakı yıllarda fizik biyologlar tarafından model olarak kullanılmıřtır. Van der Pol osilatrne ait matematiksel ifade, Denklem (3.20)'de verilmiřtir [94].

$$\ddot{x} + \alpha(x^2 - 1)\dot{x} + x = 0 \quad (3.20)$$

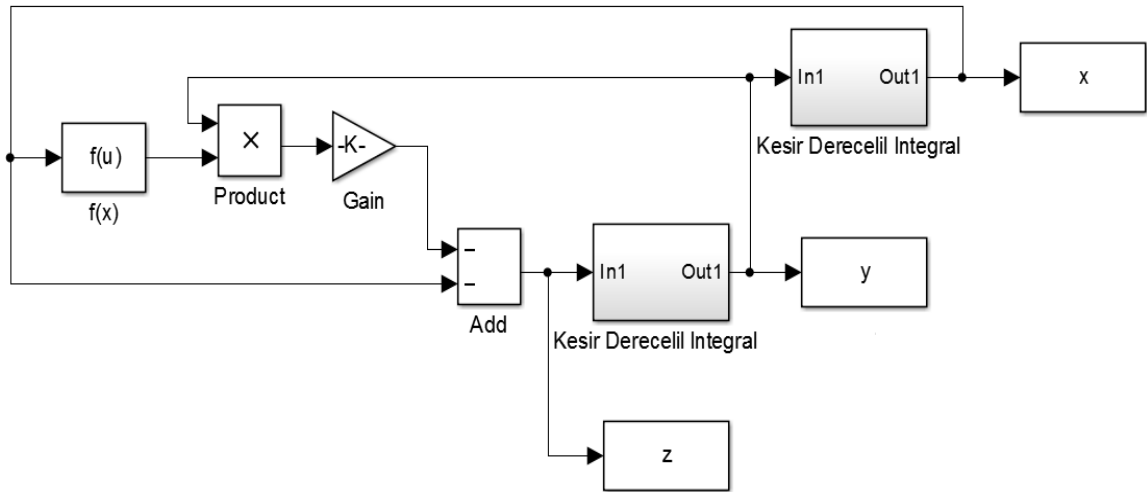
Burada, klasik türev işleminin yerine KD türev kullanılarak KD Van der Pol kaotik modeli oluşturulabilir.

3.4.1. Kesir Dereceli Van der Pol Osilatörünün Simulink Modeli

KD Van der Pol sistemi klasik Van der Pol dan farklı olarak sistem derecesi herhangi bir reel sayı olabilir. Van der Pol sistemine ait eşitlikte (Denklem (3.20)'de), x 'in derecesini herhangi reel bir sayı kabul edilir. Yani KD Van der Pol osilatörüne ait ifade, Denklem (3.21)'de görüldüğü gibidir [95].

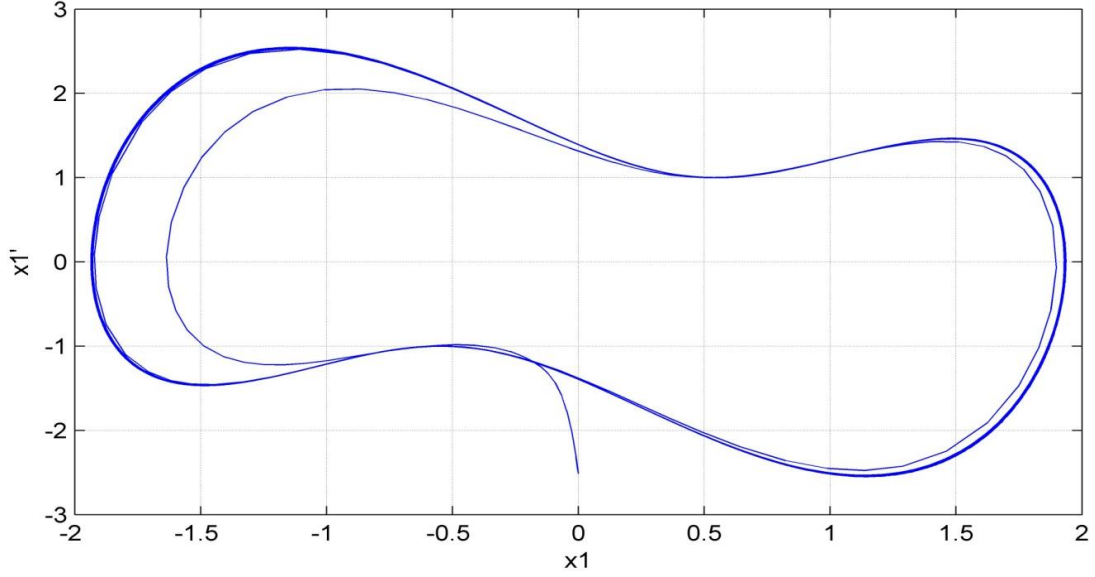
$$\begin{aligned} \frac{d^q x}{dx^q} &= y \\ \frac{d^q y}{dx^q} &= z \\ yz + f(x)y + x &= 0 \\ f(x) &= \alpha(x^2 - 1), \quad 0 < q < 1, \quad q \in R \end{aligned} \quad (3.21)$$

KD Van der Pol osilatörünün Simulink modeli, Şekil 3.7'de görülmektedir. Burada KD'li ifade, Crone yaklaşımı kullanılarak oluşturulmuştur.

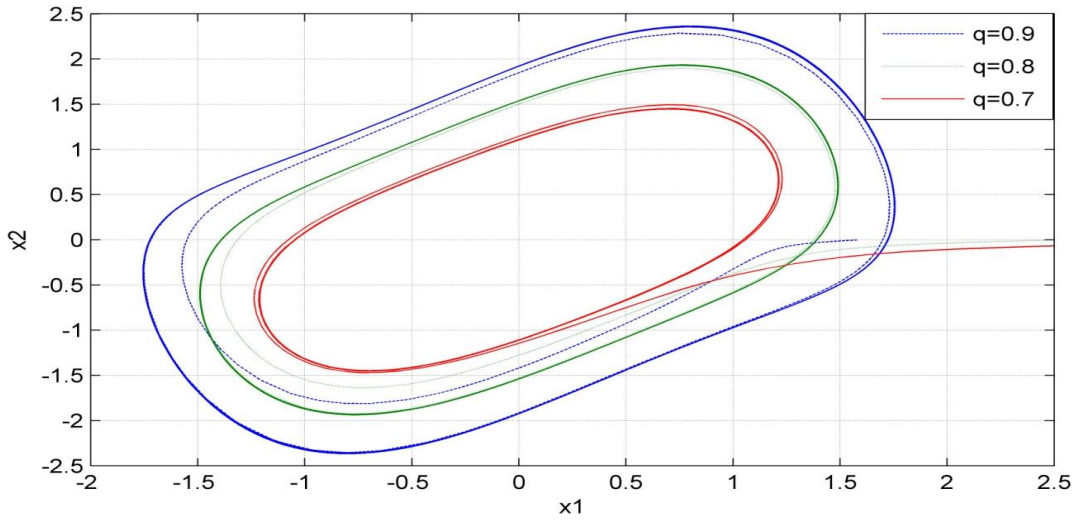


Şekil 3.7. Kesir dereceli Van der Pol osilatörünün Simulink® modeli

KD'li Van der Pol kaotik osilatörünün Simulink® modelinden elde edilen veriler ile sistemin limit çevrim davranışını, Şekil 3.8'de görüldüğü gibidir. Burada, kesir derecesi (q) arttırdığımız da kaotik çeki genliğinin arttığı Şekil 3.9'da gözlemlenmektedir.

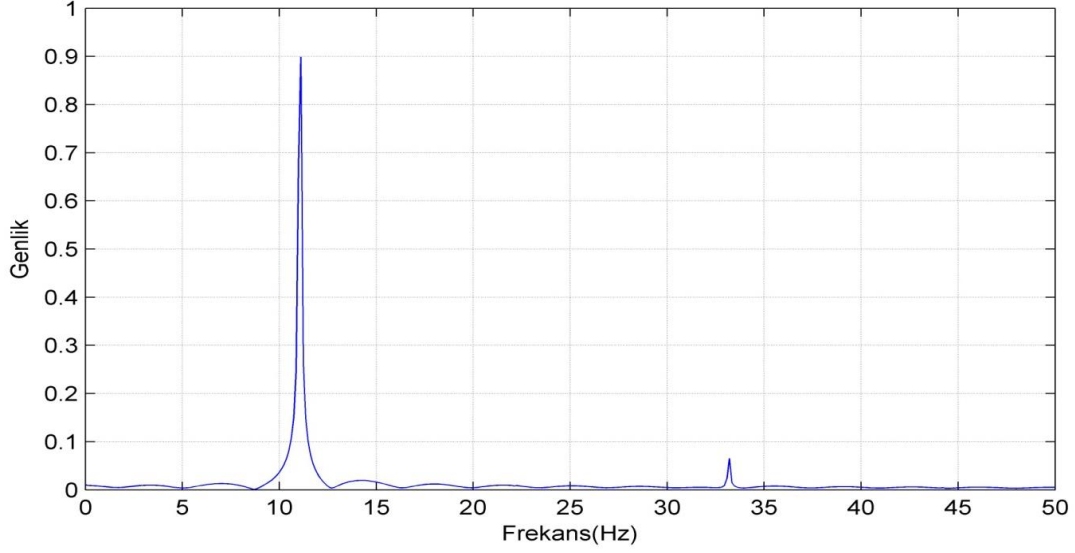


Şekil 3.8. Kesir dereceli Van der Pol osilatörünün limit çevrimi

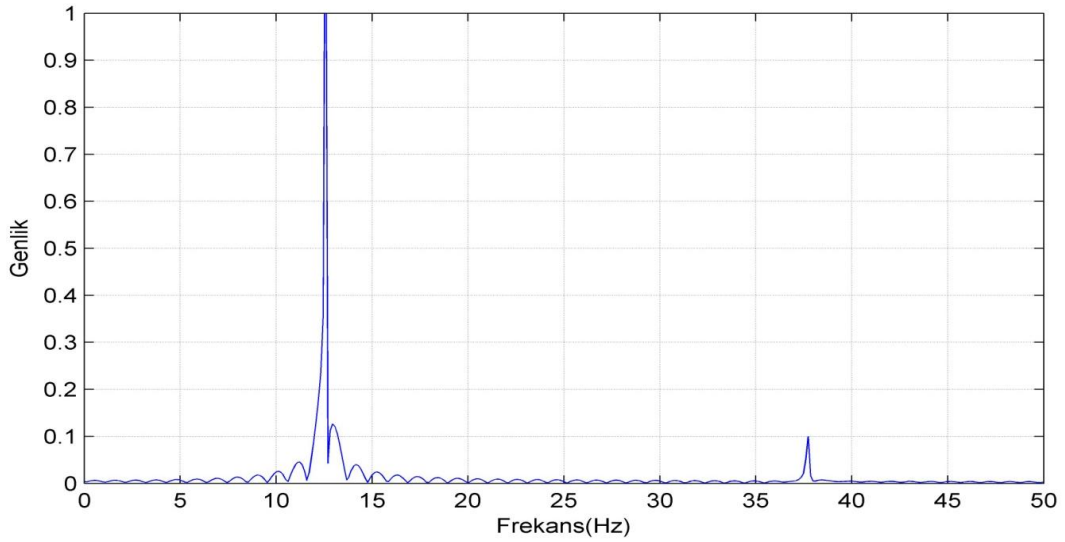


Şekil 3.9. Farklı kesir derecesine sahip Van der Pol osilatörünün ($q=0.7, 0.8, 0.9$ için) limit çevrimi

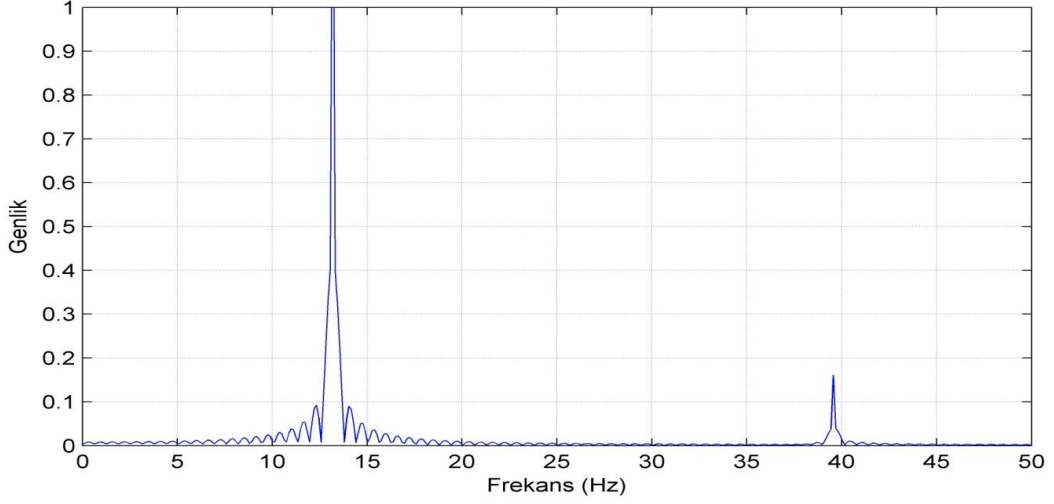
Van der Pol kaotik osilatörünün kesir derecesi deđiřtiđinde, sinyalin frekans bileřenleri de deđiřmektedir. Bu frekans bileřenleri ve genliklerini $q=0.7$ için Őekil 3.10'da, $q=0.8$ için Őekil 3.11'de ve $q=0.9$ için Őekil 3.12'de gōsterilmiřtir. Bu frekans bileřenleri gōrōldōđü gibi farklı kesir derecesine sahip sistemlerin frekans bileřenleri deđiřmektedir.



Őekil 3.10. $q = 0.7$ için kesir dereceli Van der Pol osilatörünün frekans spektrumu



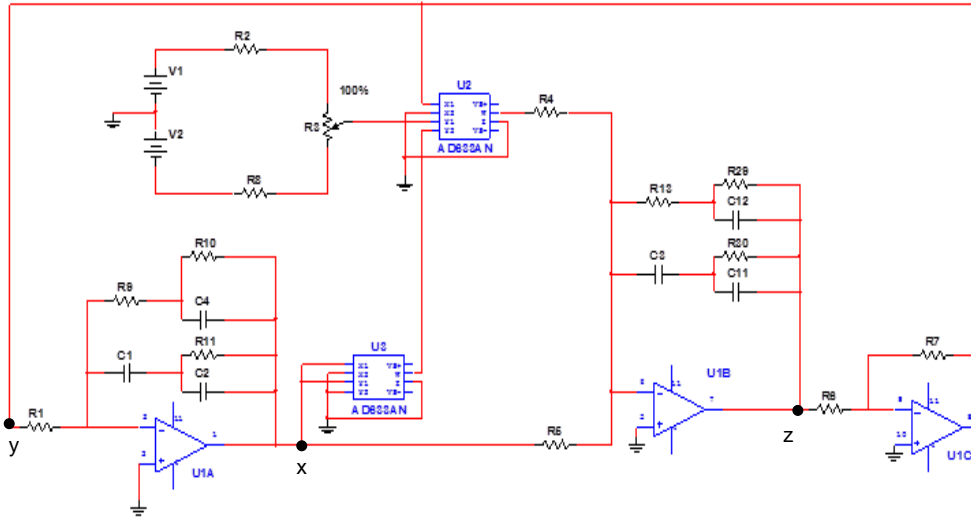
Őekil 3.11. $q = 0.8$ için kesir dereceli Van der Pol osilatörünün frekans spektrumu



Şekil 3.12. $q=0.9$ için kesir dereceli Van der Pol osilatörünün frekans spektrumu

3.4.2. Kesir Dereceli Van der Pol Osilatörün Devre Modeli

KD Van der Pol osilatörüne ait matematiksel ifadeden yola çıkarak ve Multisim 11.0 programı kullanılarak oluşturulan devre şeması Şekil 3.13’de ve bu devreye ait elemanların değerleri Tablo 3.1’de gösterilmektedir. Devrede klasik integratör modeli yerine KD’li devre modeli kullanılmıştır. Bunun için literatürde Chen Xiang-Rong v.d. tarafından oluşturulan modeli kullanılarak KD Van der Pol devresi oluşturulmuştur [96-101]

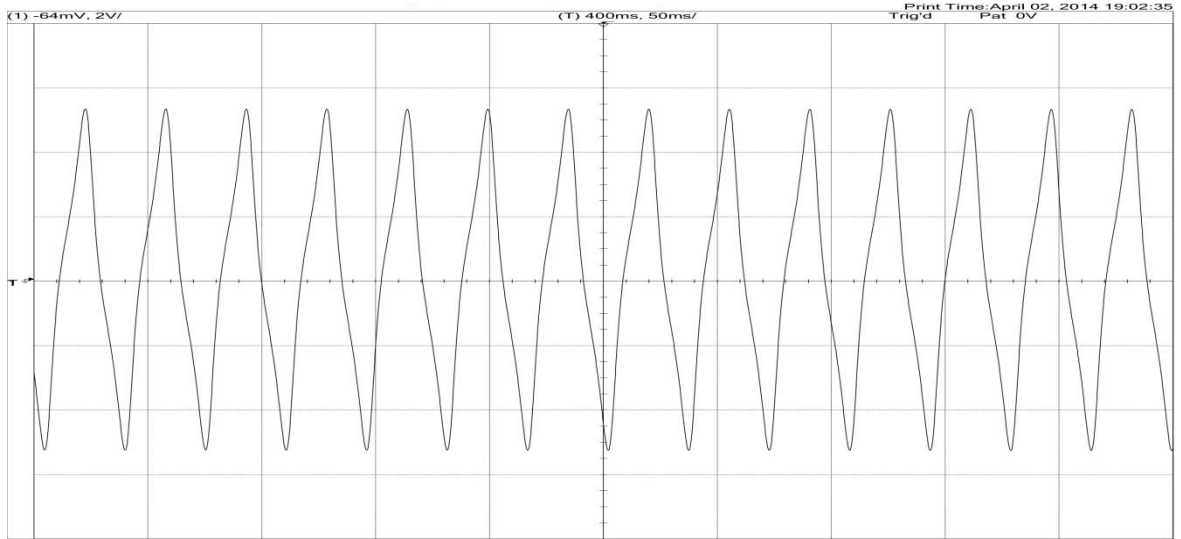


Şekil 3.13. KD Van der Pol kaotik devre modeli

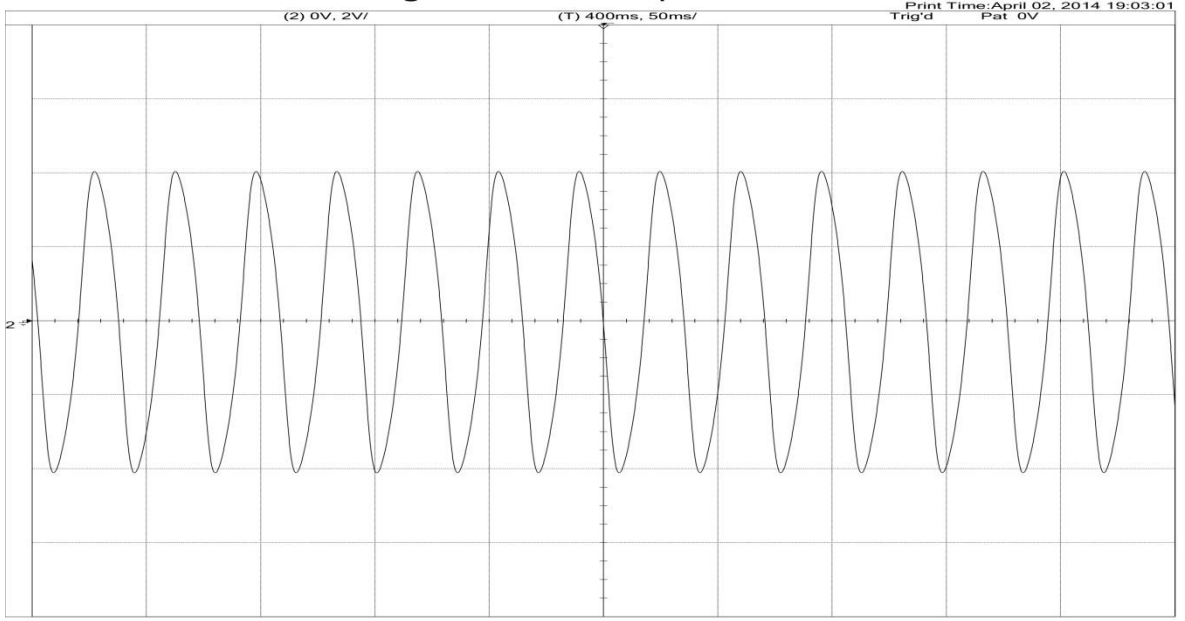
Tablo 3.1. KD Van der Pol kaotik devre elemanlarının değeri ve modeli

Devre Elemanları	Değeri veya Modeli
U_1, U_2	TL084A CN
U_2, U_3	AD633A N
R_2, R_8	470k Ω
R_1, R_5, R_6, R_7	10k Ω
R_4	1k Ω
R_3	100k Ω
R_9, R_{13}	1.55M Ω
R_{10}, R_{29}	61.54M Ω
R_{11}, R_{30}	526k Ω
C_1, C_3	7346nF
C_4, C_{12}	522.1nF
C_2, C_{11}	1.103 μ F

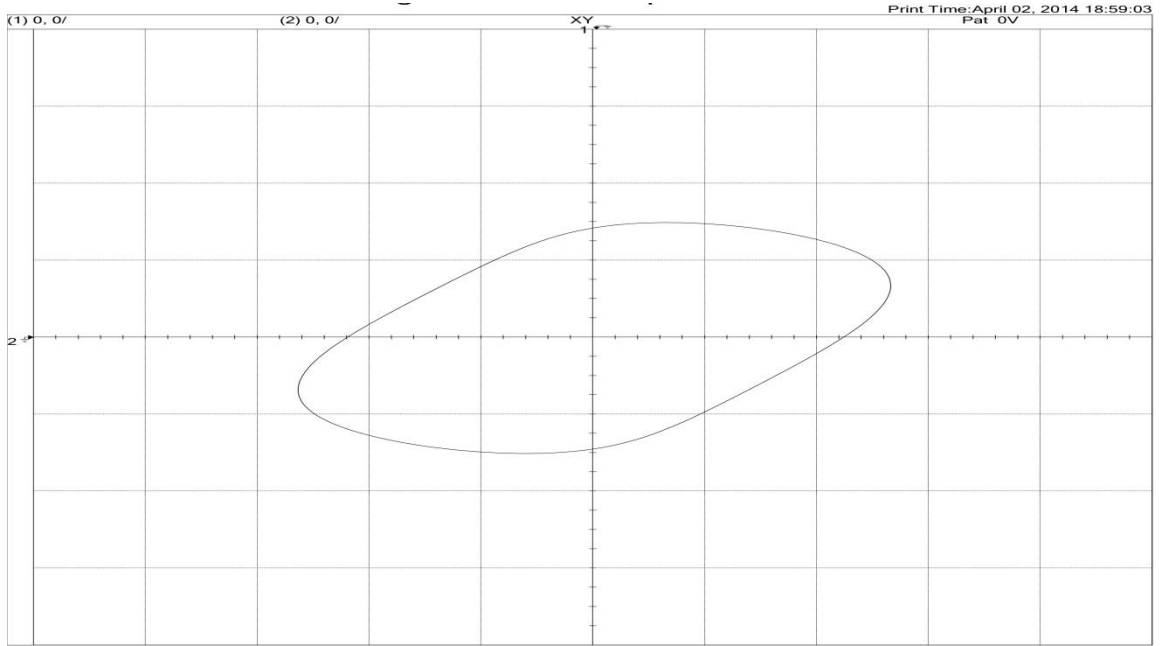
KD Van der Pol devre modelinin çıkışından elde edilen x, y ve z sinyallerinin grafikleri Şekil 3.14 ve 3.15’de gösterilmektedir. Ayrıca KD Van der Pol osilatörünün oluşan limit çevrim davranışı Şekil 3.16’da görülmektedir.



Şekil 3.14. KD Van der Pol osilatörünün x değişimi (yatay eksen:Zaman:50ms/kare, düşey eksen x:2/kare)



Şekil 3.15. KD Van der Pol osilatörünün y değişimi (yatay eksen:Zaman 50ms/kare, düşey eksen y 2V/kare)



Şekil 3.16. KD Van der Pol osilatörünün limit çevrimi (yatay eksen:x 2V/ kare, düşey eksen y 2V/ kare)

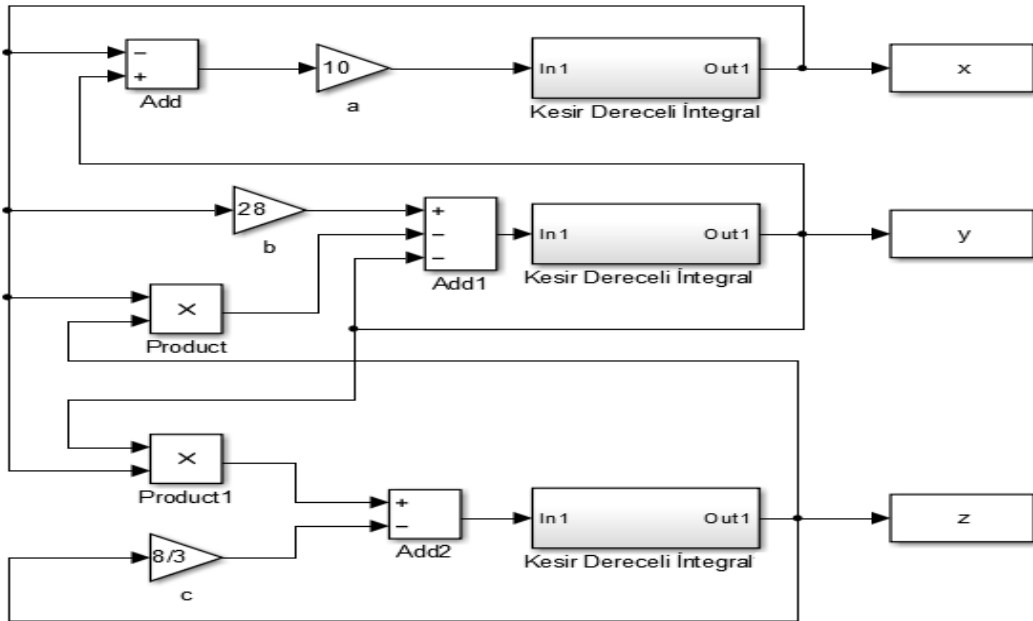
3.5. Kesir Dereceli Lorenz Sistemi

Lorenz'in hava olaylarını açıklamak için bulduğu kaotik sistemin, sonraki yıllarda KD yapısı üzerine çalışmalar yapılmıştır. Lorenz sisteminin KD'li matematiksel modeli Denklem (3.22)'de verilmiştir.

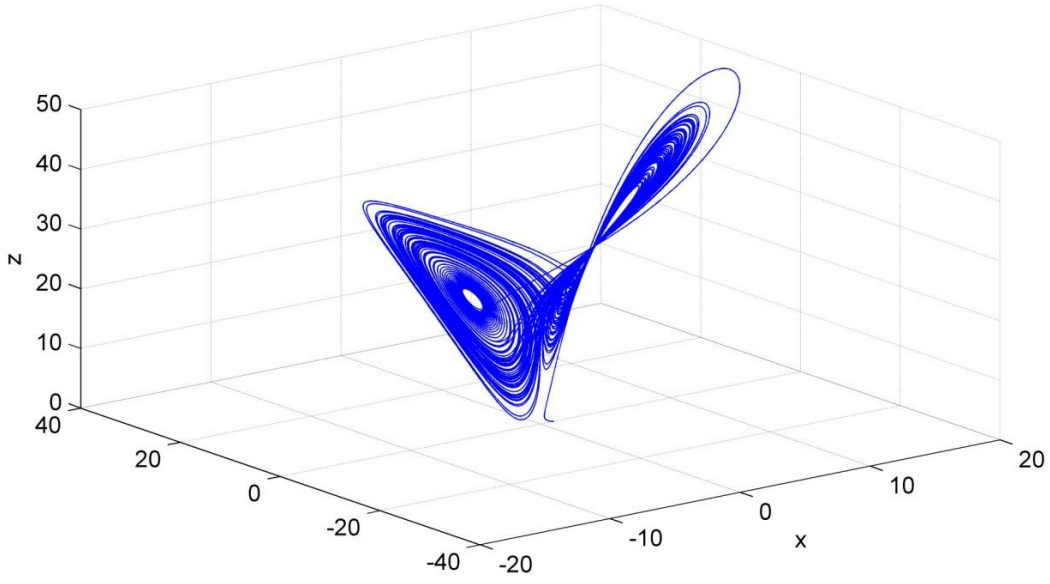
$$\begin{aligned}\frac{d^{0.9}x}{dt^{0.9}} &= 10(y-x) \\ \frac{d^{0.9}y}{dt^{0.9}} &= -xz + 28x - y \\ \frac{d^{0.9}z}{dt^{0.9}} &= xy - 2.666z\end{aligned}\quad (3.22)$$

3.5.1. Kesir Dereceli Lorenz Sisteminin Simulink Modeli

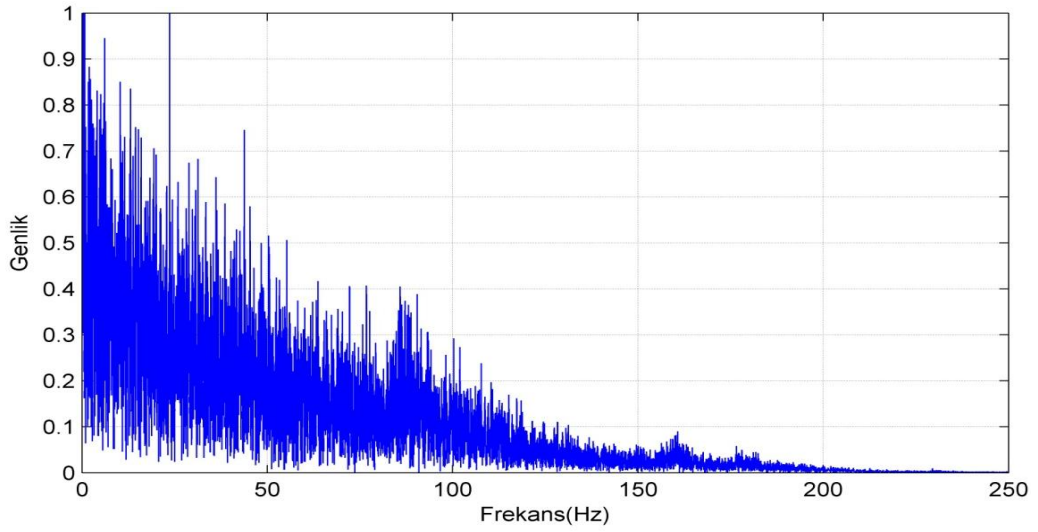
Lorenz'in matematiksel ifadesine göre oluşturulan Simulink® modeli Şekil 3.17'de gösterilmektedir. Modelin çıkışından elde edilen çıkış sinyali ve çıkış sinyalinin frekans spektrumu Şekil 3.18 ve Şekil 3.19'da gösterilmektedir. Bu iki şekilde de (Şekil 3.18 ve Şekil 3.19'da) kaotik davranışın görüldüğü gözlenmektedir.



Şekil 3.17. Kesir dereceli Lorenz osilatörünün Simulink® ile oluşturulan modeli



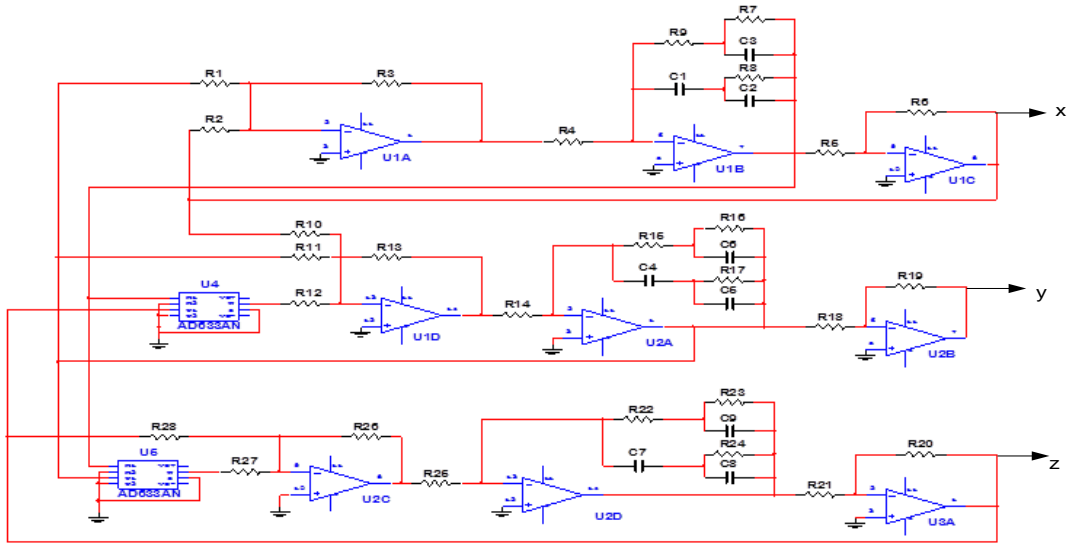
Şekil 3.18. Kesir dereceli Lorenz osilatörünün kaotik çekisi



Şekil 3.19. Kesir dereceli Lorenz osilatörünün frekans spektrumu

3.5.2. Kesir Dereceli Lorenz Sisteminin Devre Modeli

KD Lorenz sisteminin matematiksel modelinden yola çıkarak oluşturulan elektronik devre şeması, Şekil 3.20'de ve bu devreye ait elemanların değerleri Tablo 3.2'de gösterilmektedir.

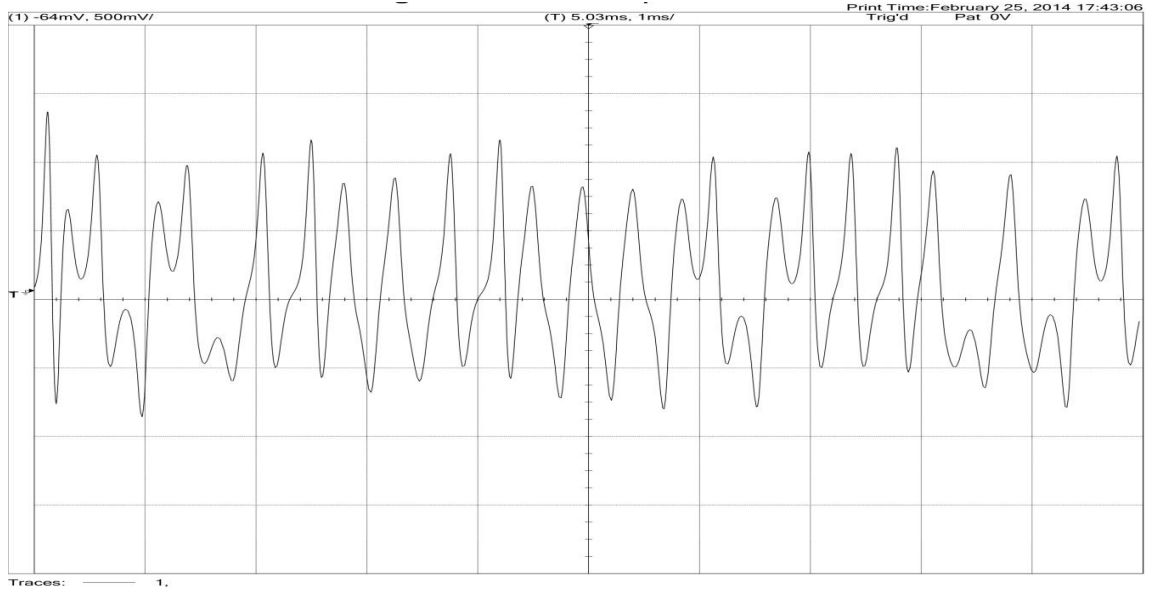


Şekil 3.20. KD Lorenz sisteminin elektronik devre şeması

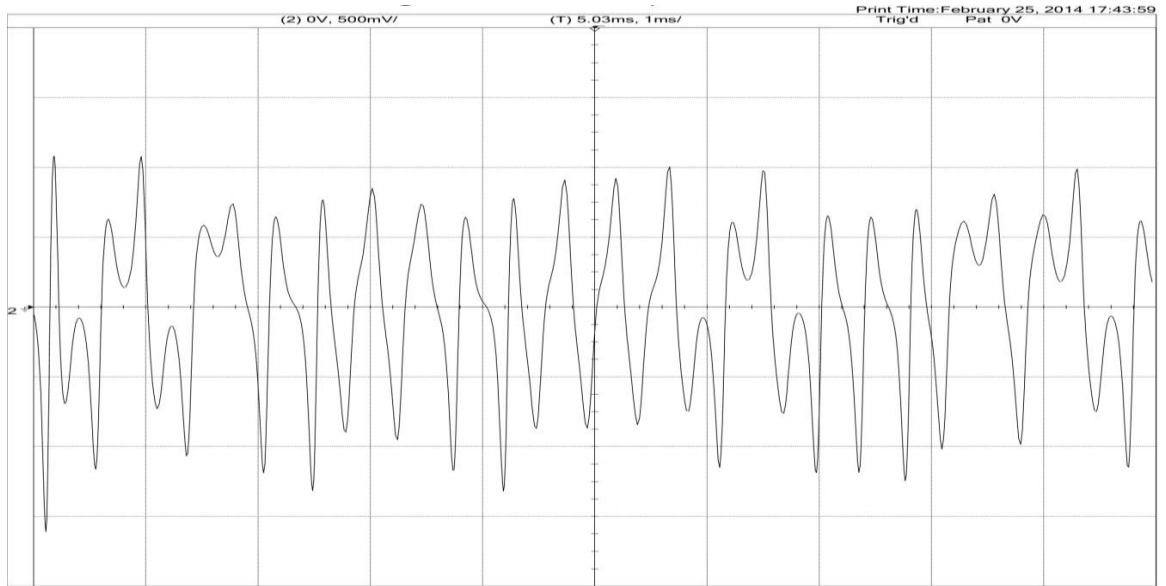
Tablo 3.2. KD Lorenz kaotik devre elemanlarının değeri ve modeli

Devre Elemanları	Değeri veya Modeli
U_1, U_2	TL084A CN
U_4, U_6	AD633A N
$R_1, R_2, R_{13}, R_{26}, R_5, R_6, R_{19}, R_{18}, R_{20}, R_{21}$	10k Ω
R_3	35k Ω
R_{27}	250 Ω
R_{28}	33.3k Ω
R_{12}	1k Ω
R_{11}	3.5k Ω
R_4, R_{14}, R_{25}	100k Ω
R_9, R_{15}, R_{22}	1.55M Ω
R_7, R_{16}, R_{23}	14.3k Ω
R_8, R_{17}, R_{24}	526k Ω
C_1, C_4, C_7	7346nF
C_3, C_6, C_9	522.1nF
C_2, C_5, C_8	1.103 μ F

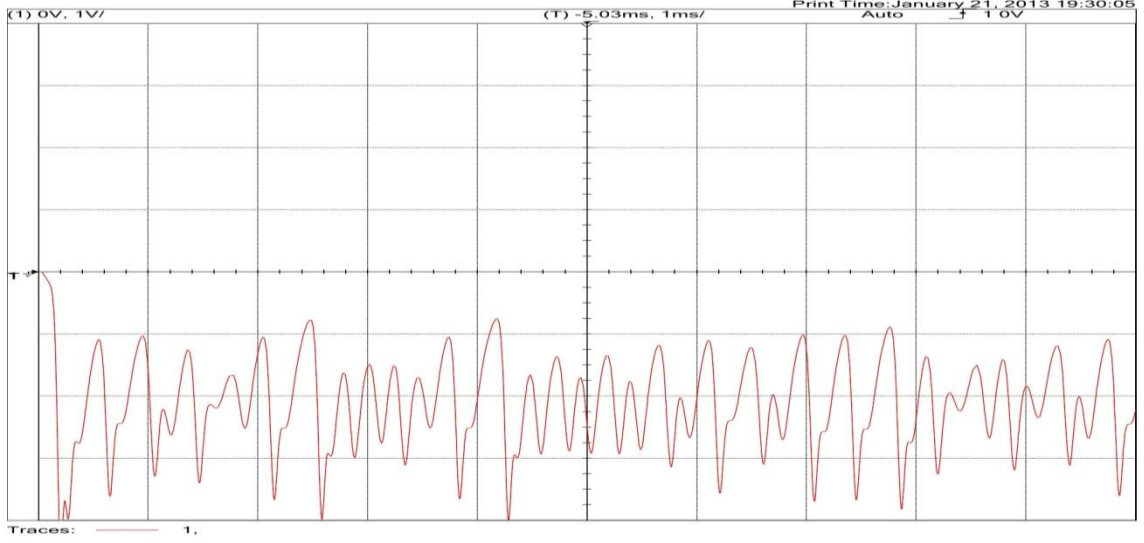
Devre çıkışından elde edilen sinyaller ve kaotik çeki Şekil 3.21-3.24'de gösterilmektedir. Bu şekillerden de görüldüğü gibi seçilen KD ve parametrelerin kaotik davranış gösterdiği görülmektedir. Osiloskop çıkışında osilatör çıkışında yaklaşık $\pm 1V$ aralığında değiştiği görülmektedir.



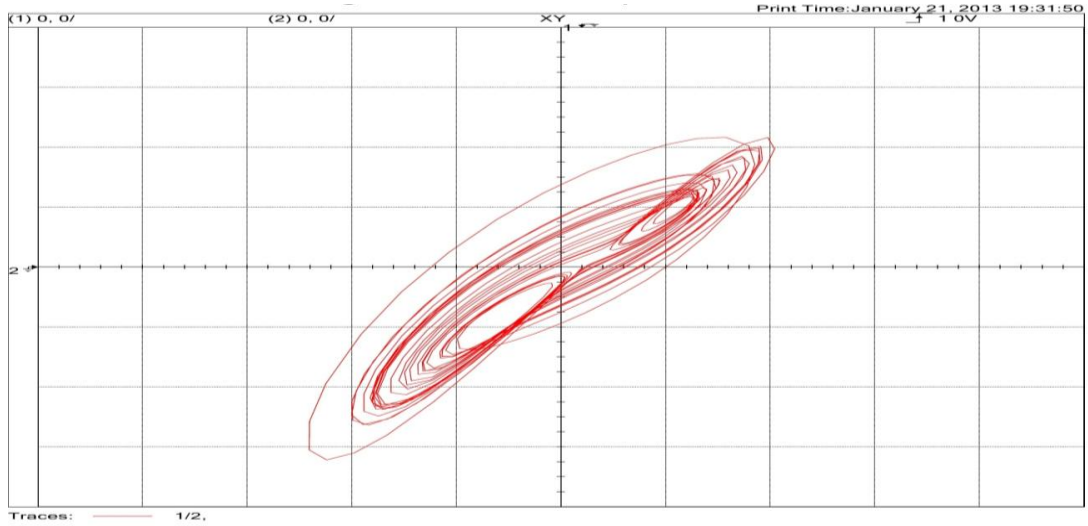
Şekil 3.21. x' 'in deęişim eęrisi (yatay eksen:Zaman:1 ms/kare, dűsey eksen x :500mV/kare)



Şekil 3.22. y' 'in deęişim eęrisi (yatay eksen:Zaman:1 ms/kare dűsey eksen y :500mV/kare)



Şekil 3.23. z' in değişim eğrisi (yatay eksen:Zaman:1 ms/kare, düşey eksen z :1V/kare)



Şekil 3.24. KD Lorenz kaotik çekisi (yatay eksen: x :500mV/kare, düşey eksen y :500mV/kare)

3.6. Kesir Dereceli Sprot Osilatör

Sprot kaotik osilatöre ait matematiksel ifade Denklem (3.23)'de verilmiştir.

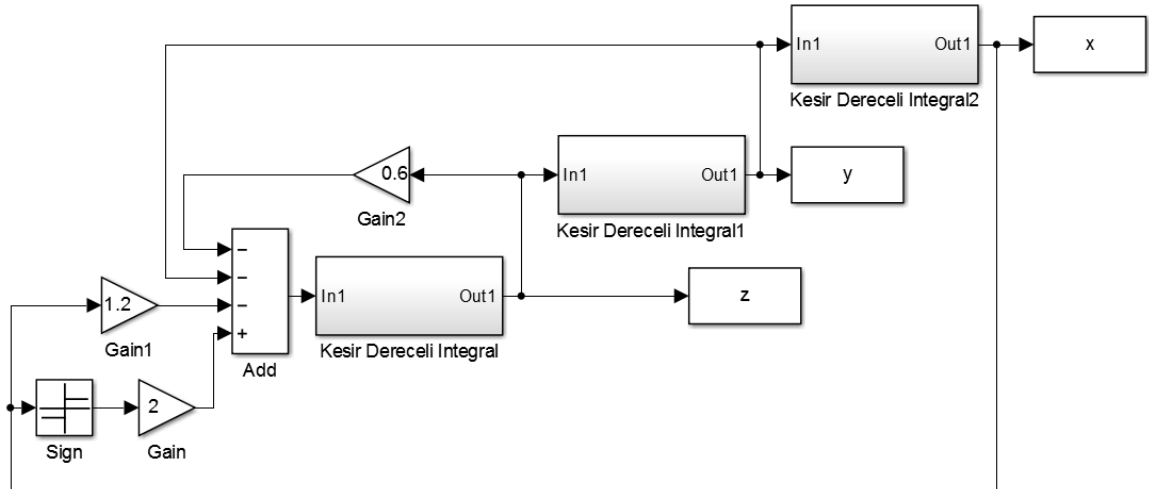
$$\begin{aligned}\frac{dx}{dt} &= y \\ \frac{dy}{dt} &= z \\ \frac{dz}{dt} &= ax - y - bz + csig(x)\end{aligned}\tag{3.23}$$

Bu modelden yola çıkarak oluşturulan KD Sprot sistemine ait matematiksel model ise Denklem (3.24)'de gösterildiği gibidir.

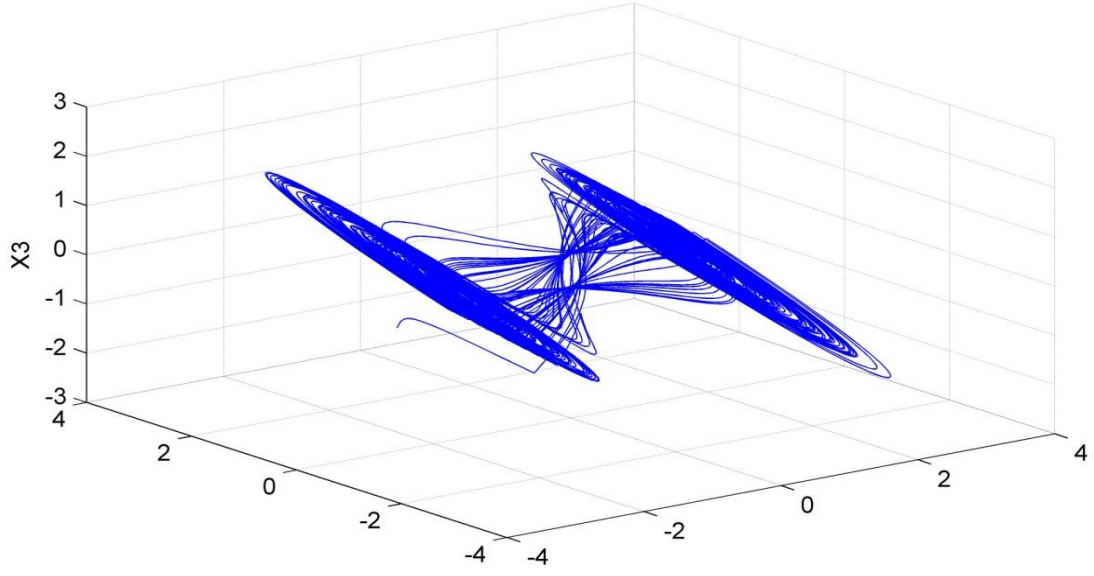
$$\begin{aligned}\frac{d^q x}{dt^q} &= y \\ \frac{d^q y}{dt^q} &= z \\ \frac{d^q z}{dt^q} &= ax - y - bz + csig(x)\end{aligned}\tag{3.24}$$

3.6.1. Kesir Dereceli Sprot Osilatörünün Simulink Modeli

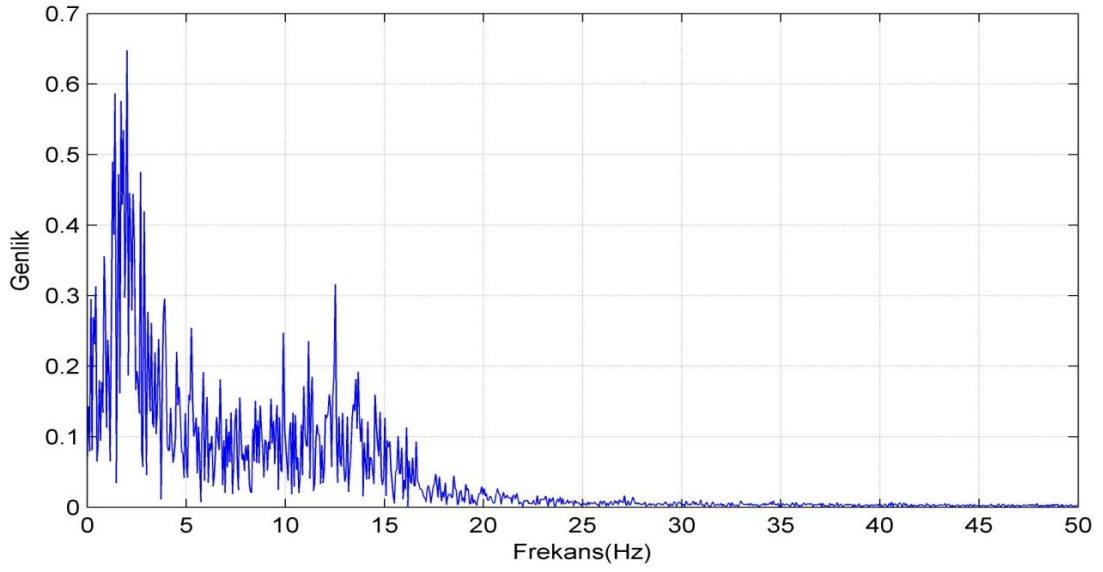
KD sprott kaotik sisteminin Simulink® modeli Şekil 3.25'de gösterilmektedir. KD Sprott modelinin (a=-1.2, b=-1, c=-0.6, d=2 ve q=0.9 olarak alınmış) oluşturduğu kaotik çeki Şekil 3.26'da ve x sinyalinin frekans spektrumu Şekil 3.27'de gösterilmektedir.



Şekil 3.25. KD Sprott kaotik model



Şekil 3.26. Sprot kaotik sistemin oluşan kaotik çekisi

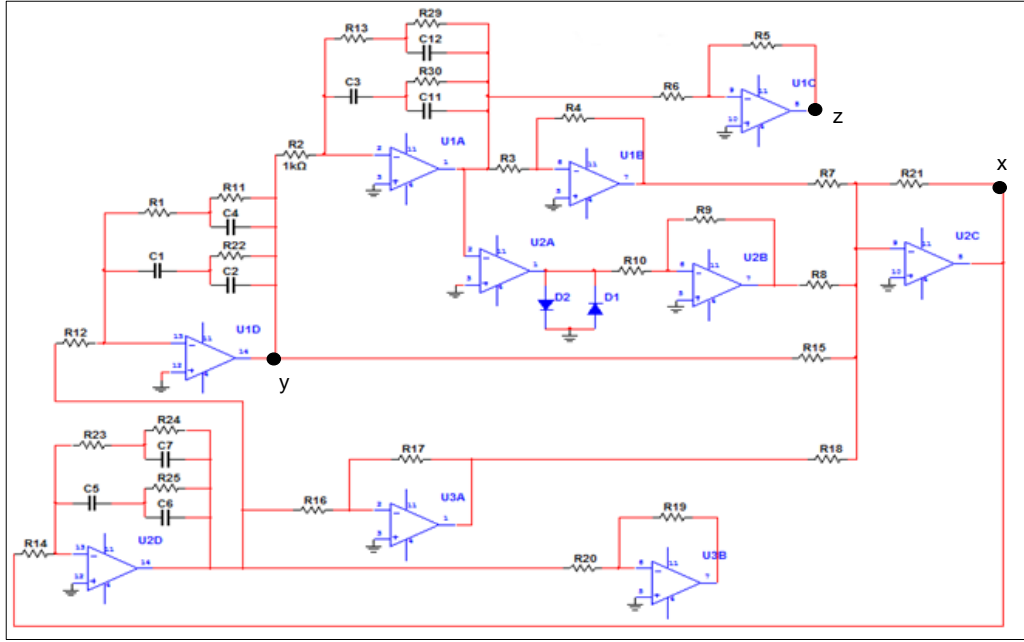


Şekil 3.27. Sprot sisteminin frekans spektrumu

Kaotik çekinin oluştuğu Şekil 3.26 ve Şekil 3.27'deki frekans spektrumlarından da görüldüğü gibi seçilen KD değerinde kaos oluşmuştur.

3.6.2. Kesir Dereceli Sprot Osilatörünün Devre Modeli

Sistemin matematiksel modelinden yola çıkarak oluşturulan devre modeli, Şekil 3.28'de ve bu devreye ait elemanların değerleri Tablo 3.3'de gösterilmiştir. Devre çıkışından elde edilen sinyaller sırasıyla Şekil 3.29-3.31'de gösterilmektedir. Devrenin oluşturduğu kaotik çeki Şekil 3.32'de gösterilmektedir.

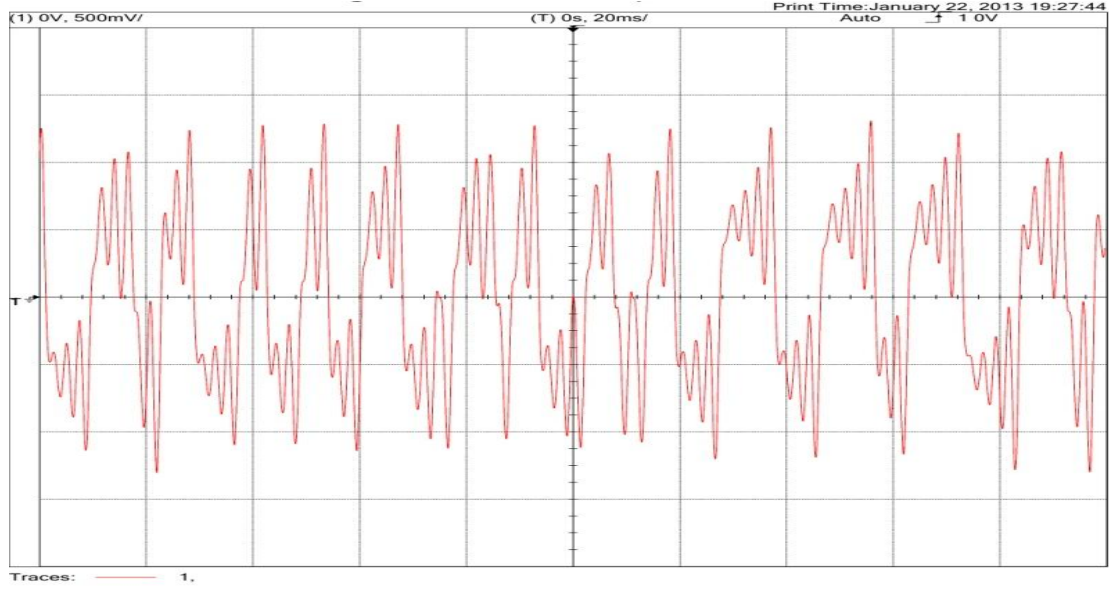


Şekil 3.28. KD Sprot kaotik sisteminin elektronik devre modeli

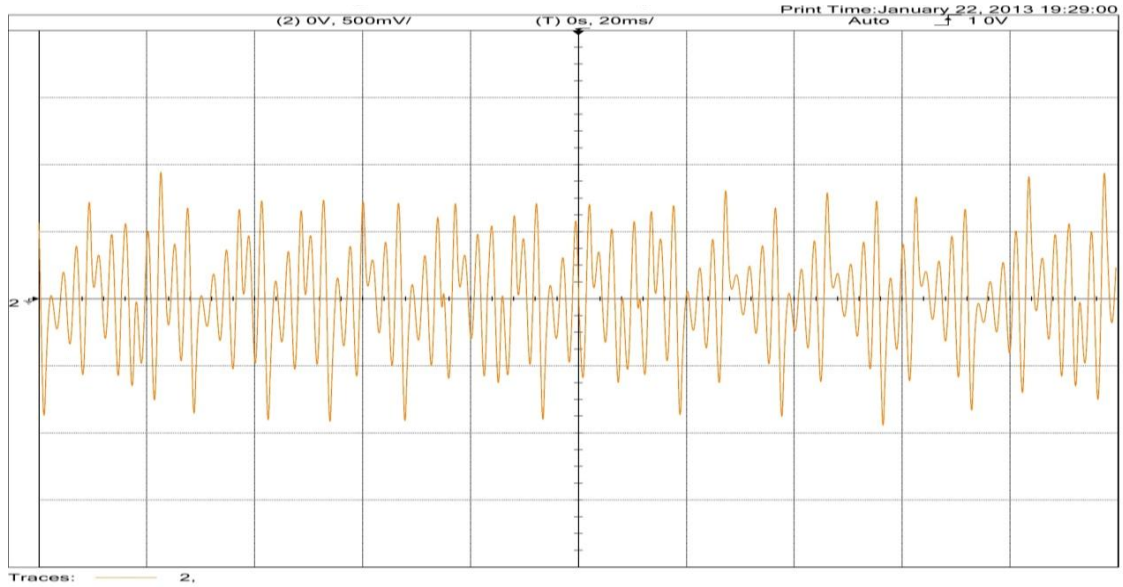
Tablo 3.3. KD Sprot kaotik devresine ait elemanlarının değeri ve modeli

Devre Elemanları	Değeri veya Modeli
U_1, U_2, U_3	TL084A CN
D_1, D_2	1N4148
$R_7, R_8, R_{15}, R_{18}, R_{21}, R_{3,}, R_5, R_6, R_9, R_{10}, R_{16}, R_{19}, R_{20},$	10k Ω
R_4	12k Ω
R_{17}	6k Ω
R_2, R_{12}, R_{14}	1k Ω
R_1, R_{13}, R_{23}	1.55M Ω
R_{11}, R_{24}, R_{29}	14.3k Ω
R_{22}, R_{25}, R_{30}	526k Ω
C_1, C_3, C_5	7346nF
C_4, C_7, C_{12}	522.1nF
C_2, C_6, C_{11}	1.103 μ F

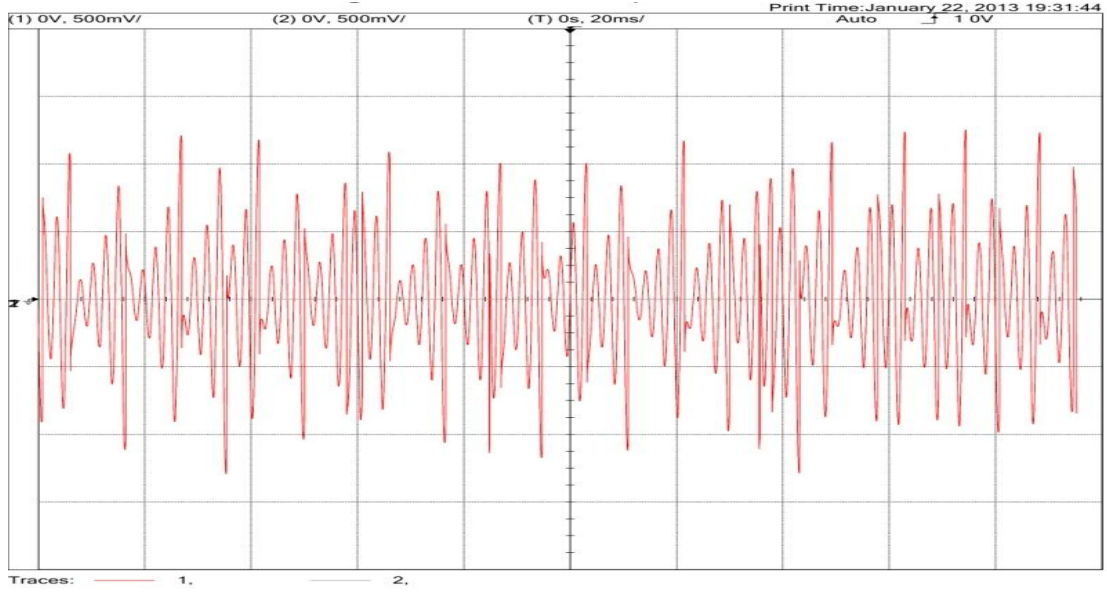
KD Sprott kaotik osilatörünün çıkışları sırasıyla Şekil 3.29-3.31'den de görüldüğü gibi genliğin ± 1 aralığında değiştiği ve kaotik davranış sergilediği açıkça gözlenmektedir. Yine x ve y değişimi birlikte çizdirildiğinde kaotik çeki Şekil 3.32'dan da kaotik davranış açıkça gözlenmektedir.



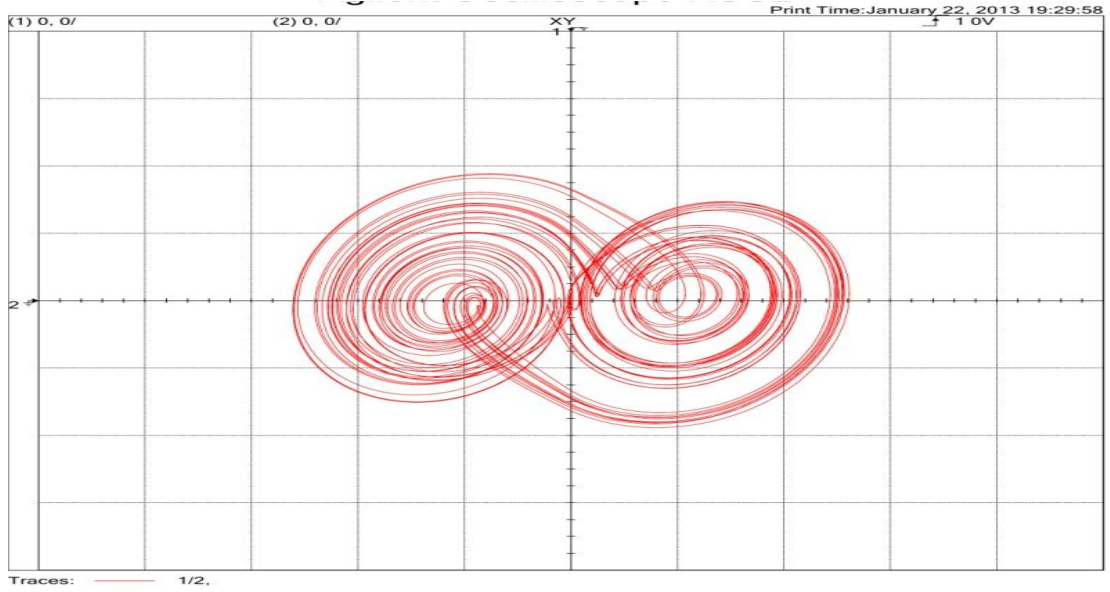
Şekil 3.29. x'in değişim eğrisi (yatay eksen: Zaman 20ms/kare, dikey eksen x 500mV/kare)



Şekil 3.30. y'in değişim eğrisi (yatay eksen: Zaman 1 ms/kare, dikey eksen y 500mV/kare)



Şekil 3.31. z' 'in deęişim eęrisi (yatay eksen:Zaman 1 ms/kare, dűşey eksen z 500mV/kare)



Şekil 3.32. KD Sprott kaotik sisteminin kaotik çekisi (yatay eksen: z 500mV/kare, dűşey eksen z 500mV/kare)

3.7. Kesir Dereceli Van der Pol Osilatörünün Senkronizasyonu

Bu bölümde KD Van der Pol kaotik sisteminin senkronizasyonu için Simulink ve elektronik devre modeli oluşturulmuştur. Senkronizasyon denetimi için daha öncede

bahsettiğimiz klasik PID ve KDPID olarak iki farklı denetim yöntemi uyguladık. Denetleyici parametreleri ve dereceleri FAPSO yöntemiyle hesaplanmıştır.

3.7.1. Kesir Dereceli Van der Pol Osilatörünün Senkronizasyonu için Simulink Modeli

KD Van der Pol sisteminin senkronizasyonunda, alıcı tarafta bulunan kaotik sistemde denetleyici girişi bulunmaktadır. Denetleyici girişinin bulunduğu kaotik sistemin matematiksel ifadesi Denklem (3.25)'de gösterilmektedir.

$$x^{(1+\lambda)} + \alpha(x^2 - 1)x^{(\lambda)} + x + u = 0$$

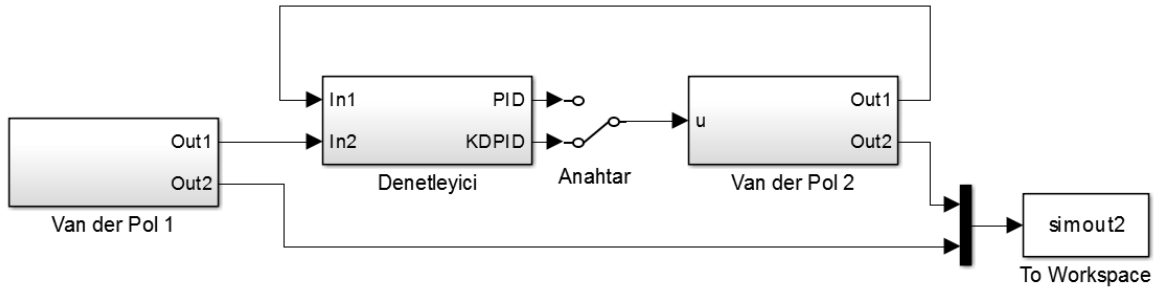
$$0 < \lambda < 1$$

$$\lambda \in R$$
(3.25)

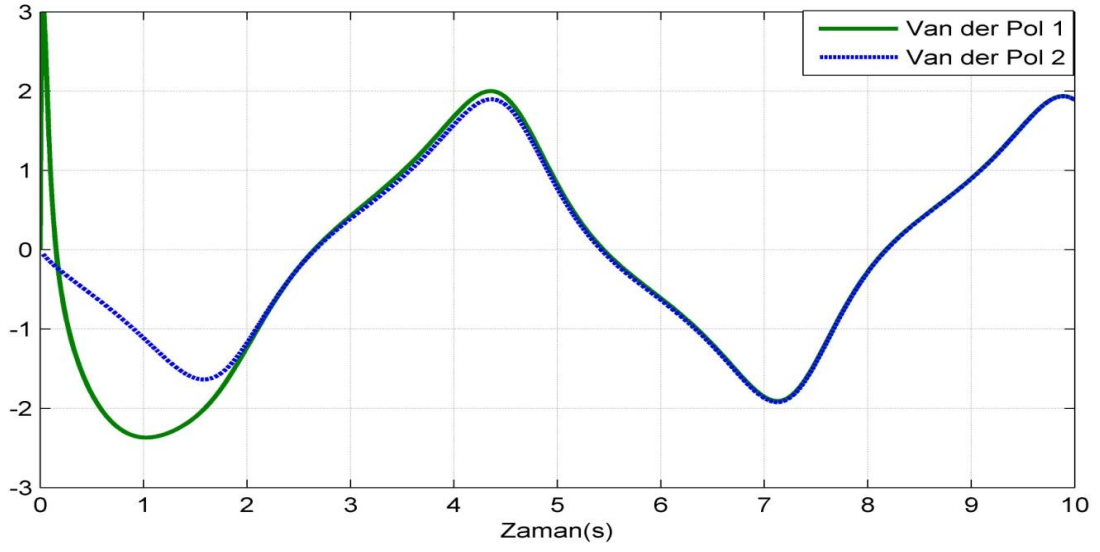
Burada “u” denetleyici girişi olup, senkronize hatasına göre çıkış üreten herhangi bir denetleyici tarafından denetlenir.

Van der Pol sisteminin senkronizasyon denetimi için kullanılan Simulink modeli Şekil 3.33’de gösterilmiştir.

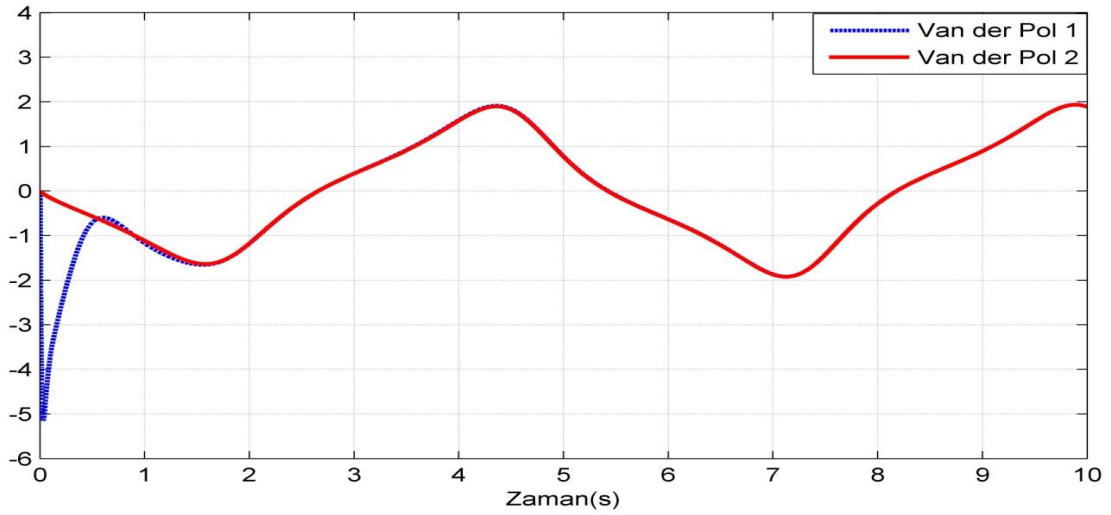
PID ve KDPID sistemine ait sistem çıkışlar sırasıyla Şekil 3.34 ve Şekil 3.35’de gösterilmektedir.



Şekil 3.33. Van der Pol sisteminin senkronizasyonu için Simulink modeli



Şekil 3.34. Van der Pol sisteminin PID ile senkronizasyon denetimi



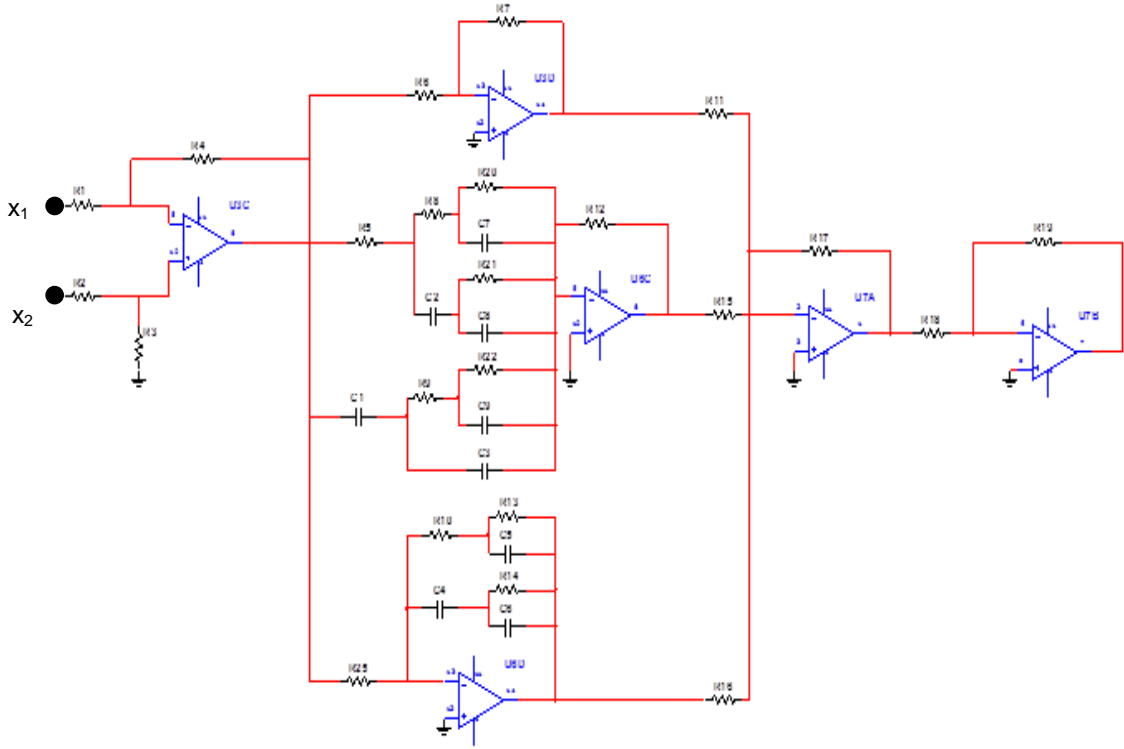
Şekil 3.35. Van der Pol kaotik sisteminin KDPID ile senkronizasyon denetimi

Klasik PID denetleyicisiyle yapılan denetim işleminde Şekil 3.34'den de görüldüğü gibi senkronizasyon süresi yaklaşık olarak 2 s olarak gerçekleşmiştir. Ayrıca aşma miktarının da fazla olduğu gözlenmektedir. Sistemin 5. s'de master ve slave osilatörleri arasında genlik farkı gözlenmektedir.

Eniyilenen KDPID denetleyicisiyle yapılan Van der Pol osilatörünün senkronizasyon denetiminde ise senkronizasyon süresi 0.5 s'den biraz fazla olduğu gözlenmiştir. Ayrıca aşma miktarı açısından da iyi bir sonuç verdiği Şekil 3.35'da gözlenmektedir.

3.7.2. Kesir Dereceli Van der Pol Osilatörünün Senkronizasyonu için Devre Modeli

Van der Pol kaotik sisteminin, Multisim 11 devre simülasyon programıyla analizi için kurulan devre modeli Şekil 3.36'da gösterilmiştir. Tasarlanan bu denetleyiciye ait devre elemanlarının modeli ve değeri Tablo 3.4'de verilmiştir.

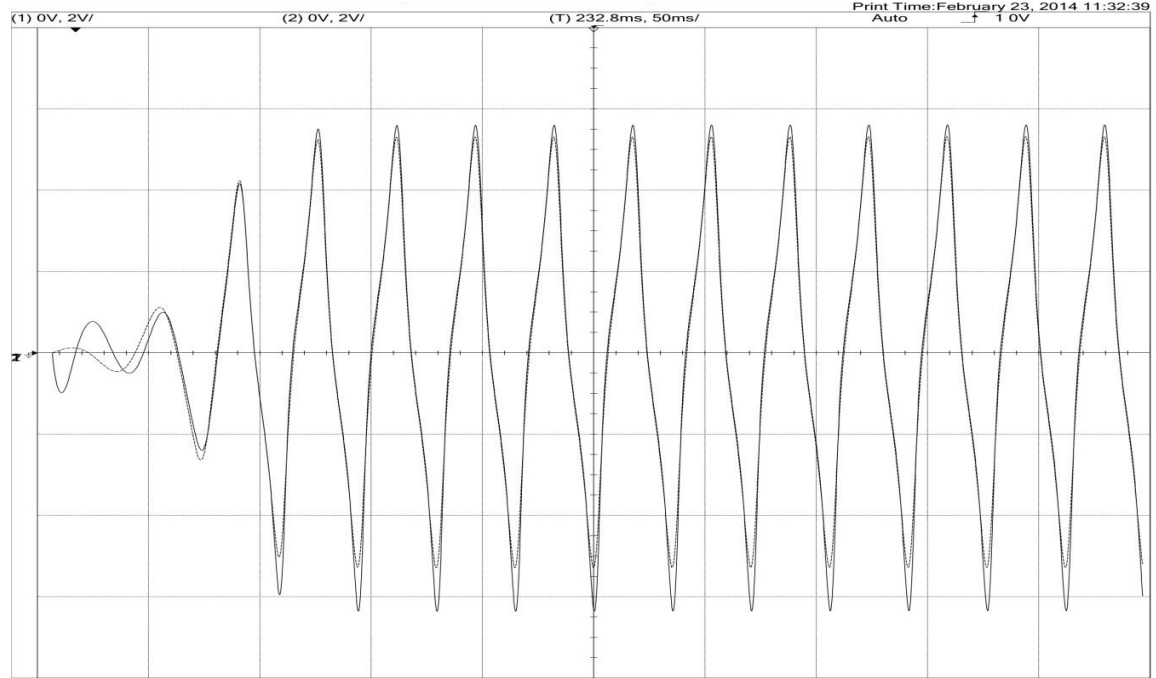


Şekil 3.36. KD Van der Pol sisteminin KDPID ile senkronizasyon kontrol devresi

Master ve slave kaotik iki devrenin ürettiği sinyallerin şekilleri Şekil 3.37'de verilmiştir. Çıkış işaretlerinin genliği yaklaşık $\pm 5.5V$ arasında değiştiği gözlenmektedir. Ayrıca yaklaşık 50ms sürede senkronize gerçekleştiği ve senkronizasyonun simülasyon süresi boyunca bozulmadığı görülmektedir.

Tablo 3.4. Önerilen KDPID denetleyicisine ait devre elemanlarının modeli ve değeri

Devre Elemanı	Değeri veya Modeli	Devre Elemanı	Değeri veya Modeli
U ₃ , U ₆ , U ₇	TL084ACN	C ₁	3.793μF
D ₁ , D ₂	1N4148	C ₂	582.7μF
R ₁ , R ₂ , R ₃ , R ₄ , R ₆ , R ₁₁ , R ₁₇ , R ₁₈ , R ₁₉ , R ₁₀ , R ₁₆ ,	10kΩ	C ₃	64.16nF
R ₁₂ , R ₂₅	1kΩ	C ₇	175.1 μF
R ₅	174.9kΩ	C ₈	229.2nF
R ₈	441.7kΩ	C ₉	4.441μF
R ₉	2.48MΩ	C ₄	25.72μF
R ₂₀	9.41MΩ	C ₅	15.77μF
R ₂₁	695.7kΩ	C ₆	646.8μF
R ₂₂	2.354MΩ	R ₁₄	1.363M
R ₁₀	951.7k		
R ₁₃	633.2		



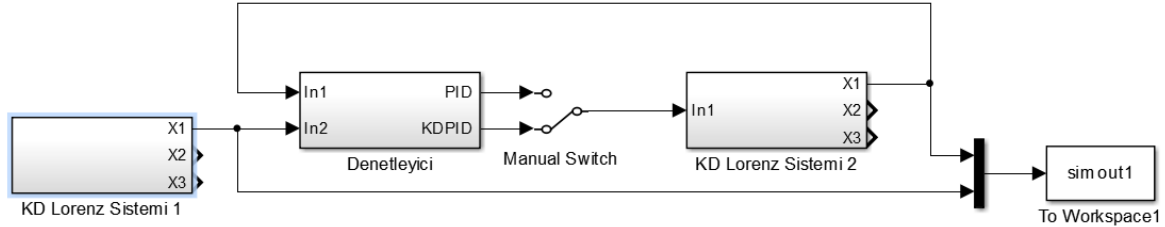
Şekil 3.37. KDPID ile kontrol edilen iki farklı Van der Pol devresinin ürettiği kaotik sinyallerin şekilleri (yatay eksen:Zaman 50ms/kare, düşey eksenler x_1 ve x_2 500mV/kare)

3.8. Kesir Dereceli Lorenz Sisteminin Senkronizasyonu

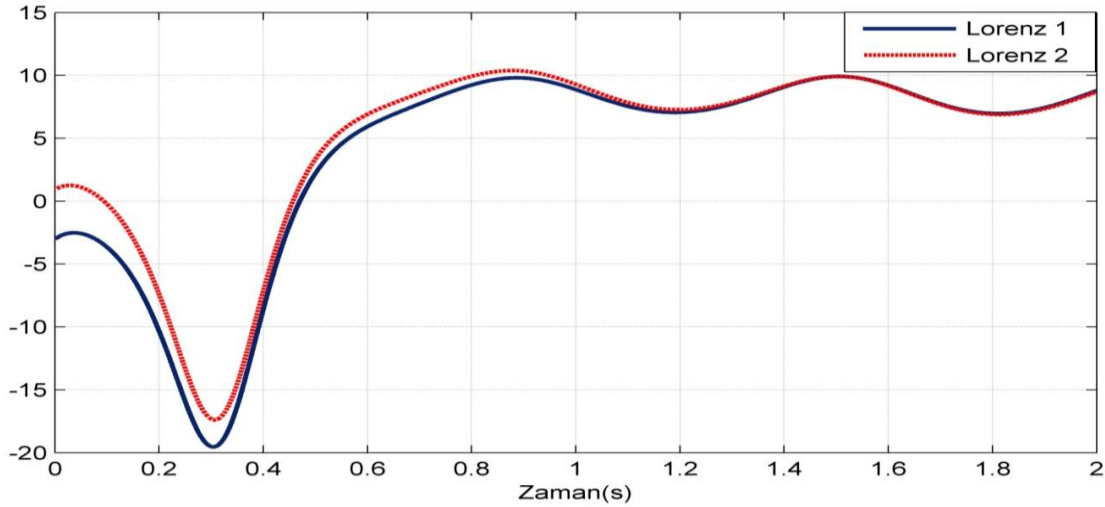
Bu bölümde bahsettiğimiz klasik PID ve KDPID yöntemlerinin etkilerini görmek için Simulink modeli ve Multisim 11.0'da elektronik devre modelleri oluşturulmuştur.

3.8.1. Kesir Dereceli Lorenz Sisteminin Senkronizasyonu için Simulink Modeli

KD dereceli lorenz sisteminin senkronizasyonu için Simulink modeli, Şekil 3.38’de gösterilmektedir. PID ve KDPID denetleyicisiyle yapılan denetim sonucu iki kaotik sinyalin değişimi, sırasıyla Şekil 3.39 ve Şekil 3.40’da gösterilmektedir.

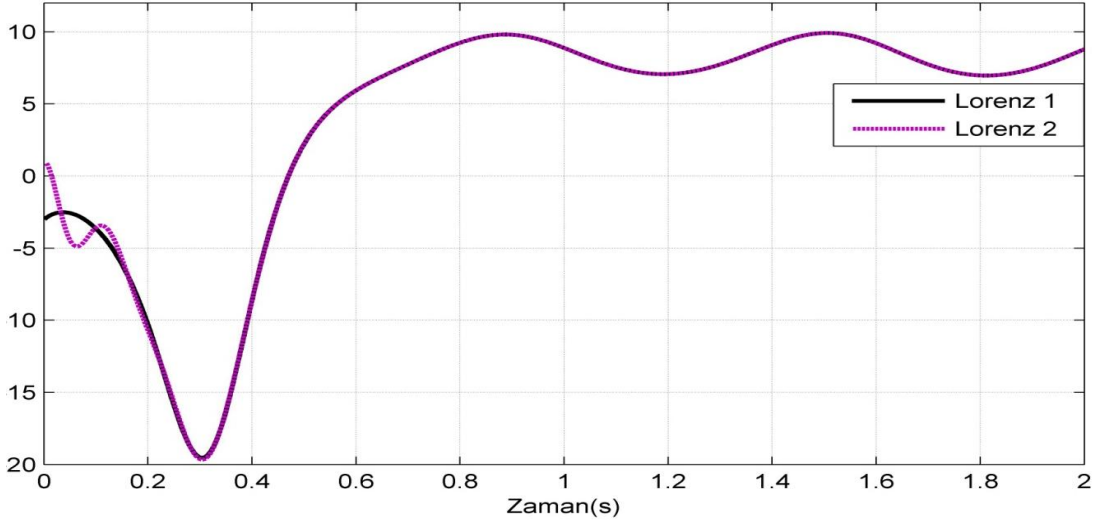


Şekil 3.38. KD Lorenz kaotik sisteminin denetleyicisinin Simulink modeli



Şekil 3.39. PID ile senkronizasyon kontrolü yapılan iki KD Lorenz kaotik sistemlerinin çıkışları

Klasik PID denetleyicisinde Şekil 3.39’den da görüldüğü gibi senkronizasyon hatasının yaklaşık 0.4 s’de kabul edilir seviyeye düştüğü gözlenmektedir. Ancak 0.5 s ile 0.8 s arasında bu hatanın tekrardan arttığı ve yaklaşık 1 s sonra sifira düştüğü görülmektedir.



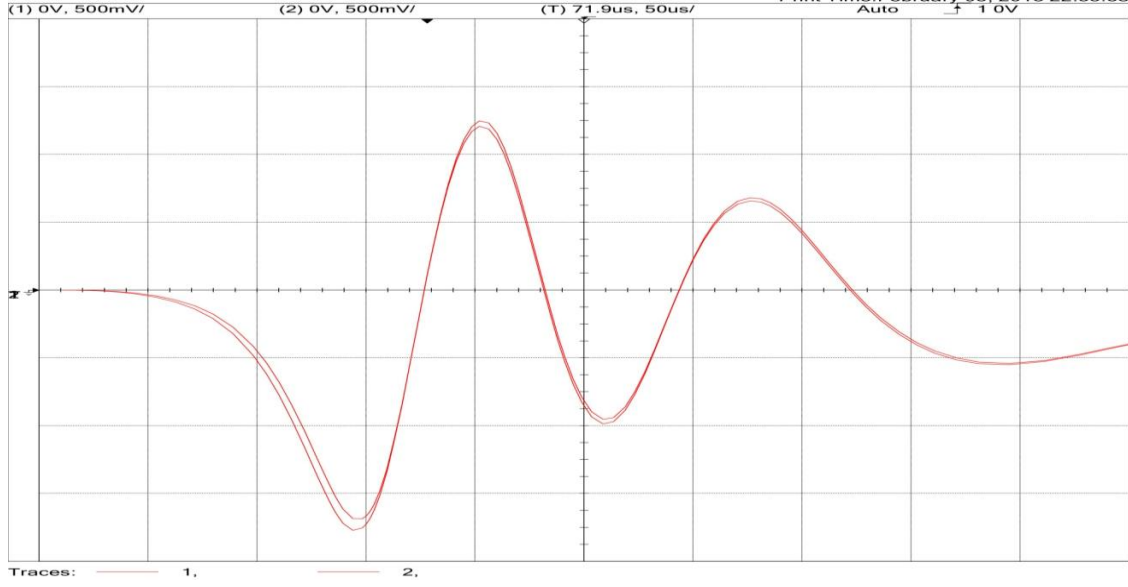
Şekil 3.40. KDPID ile senkronizasyon kontrolü yapılan iki KD Lorenz kaotik sistemlerinin çıkışları

KDPID denetleyicisinde ise senkronizasyonun yaklaşık 0.15 s'de gerçekleştiği Şekil 3.40'da gözlenmektedir. Ayrıca senkronizasyonun ilerleyen zaman içerisinde bozulmadığı da bu şekilde görülmektedir.

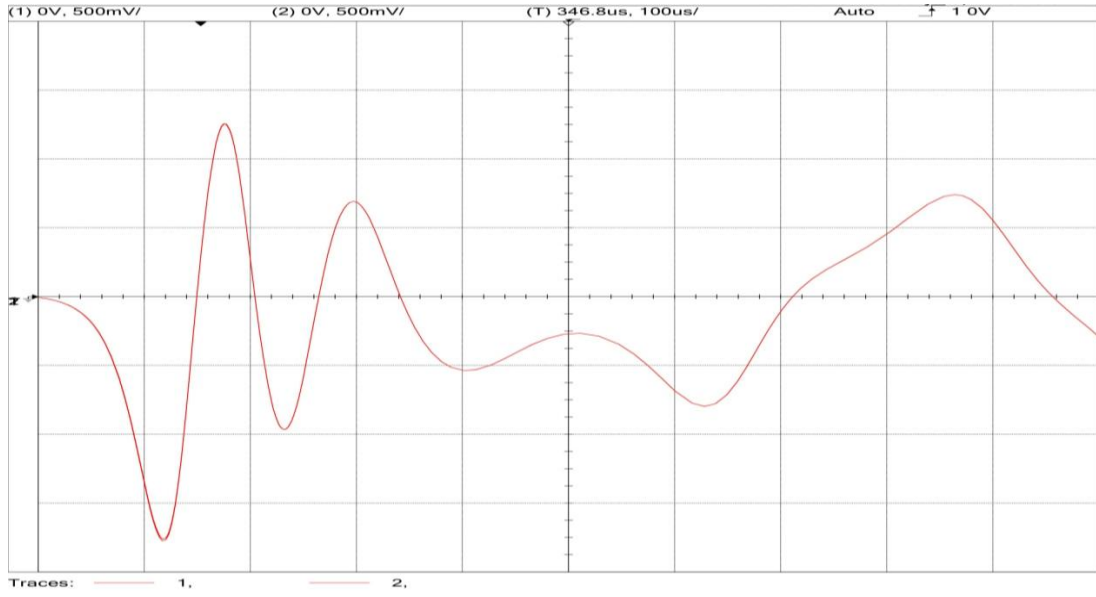
3.8.2. Kesir Dereceli Lorenz Sisteminin Senkronizasyonu için Elektronik Devre Modeli

KD Lorenz kaotik sisteminin elektronik modeli Şekil 3.20'de gösterildiği gibidir. İki kaotik sistemin aynı çıkış üretmesi için tasarlanan KDPID denetleyicisinin devre modeli, Şekil 7.30'da gösterilen devre modeli ve Tablo 3.4'de verilen devre elemanları kullanılmıştır.

Bu iki kaotik sistemin çıkışında elde edilen grafikler, Şekil 3.41 ve Şekil 3.42'de gösterilmiştir. Burada sırasıyla klasik PID ve KDPID tipi denetleyici kullanılmıştır. Bu iki denetleyiciyle yapılan senkronizasyonun süreleri karşılaştırıldığında, klasik PID yönteminde senkronizasyonun 0.15 ms'de gerçekleştiği görülürken, KDPID yönteminde ise neredeyse ilk andan itibaren senkronizasyonu koruduğu gözlenmektedir.



Şekil 3.41. PID ile Lorenz kaotik sisteminin senkronizasyon kontrolü (yatay eksen:Zaman 50ms/kare, düşey eksenler x_1 ve x_2 500mV/kare)



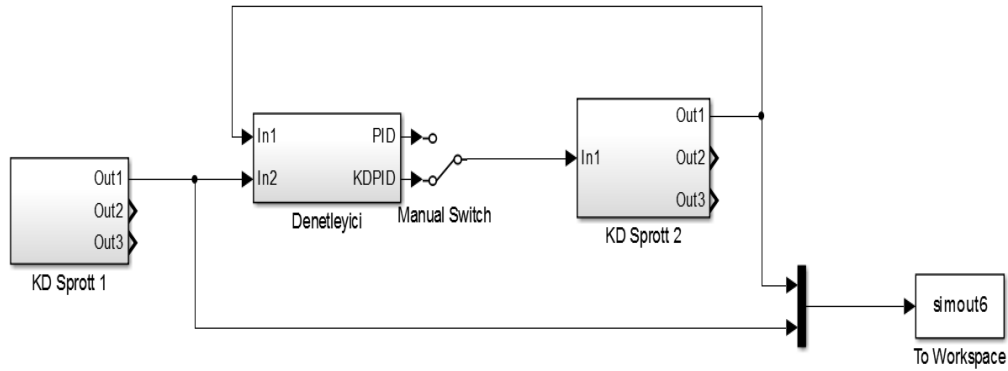
Şekil 3.42. KDPID ile Lorenz kaotik sisteminin senkronizasyon kontrolü (yatay eksen:Zaman 50ms/kare, düşey eksenler x_1 ve x_2 500mV/kare)

3.9. Kesir Dereceli Sprot Systeminin Senkronizasyonu

Literatürde bulunan bir klasik Sprot osilatörünün, bu kısımda Simulink modeli ve Multisim 11.0'da elektronik devre modelleri oluşturulmuş ve bu sistemlerin senkronizasyon denetiminde klasik PID ve KDPID denetleyicisinin etkisi incelenmiştir.

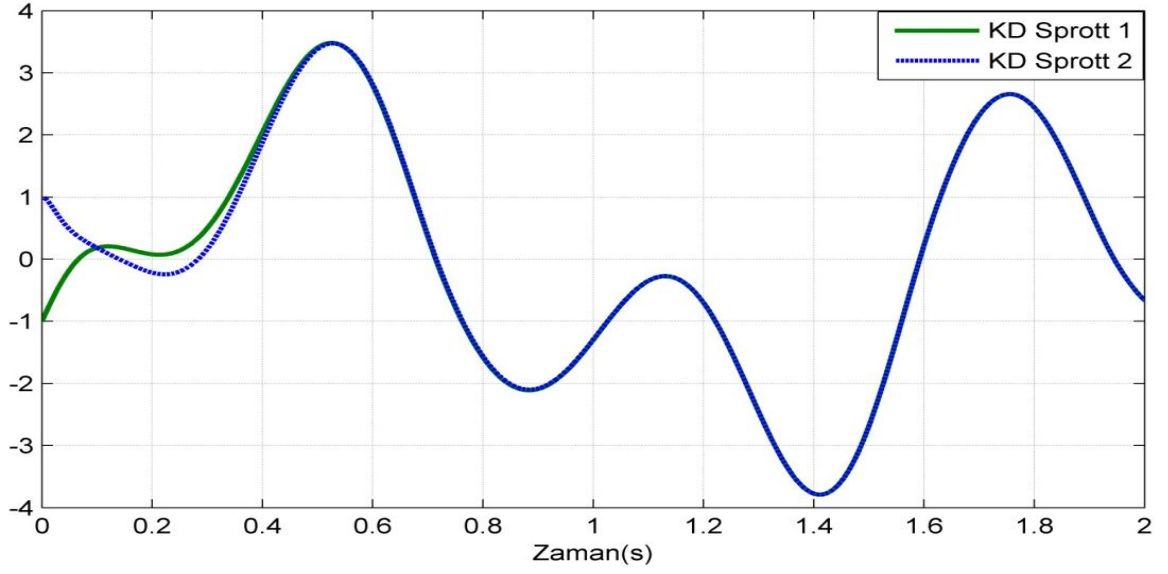
3.9.1. Kesir Dereceli Sprot Systeminin Senkronizasyonu için Simulink Modeli

KD Sprot kaotik sisteminin senkronizasyonu simülasyonu için Simulink modeli Şekil 3.43'de gösterilmiştir. Denetleyicilerin performansını sınamak için klasik PID ve KDPID ile ayrı ayrı denetleme yapılmıştır. Klasik PID ve KDPID ile elde edilen sonuçlar sırasıyla, Şekil 3.44 ve Şekil 3.45'de gösterilmiştir.

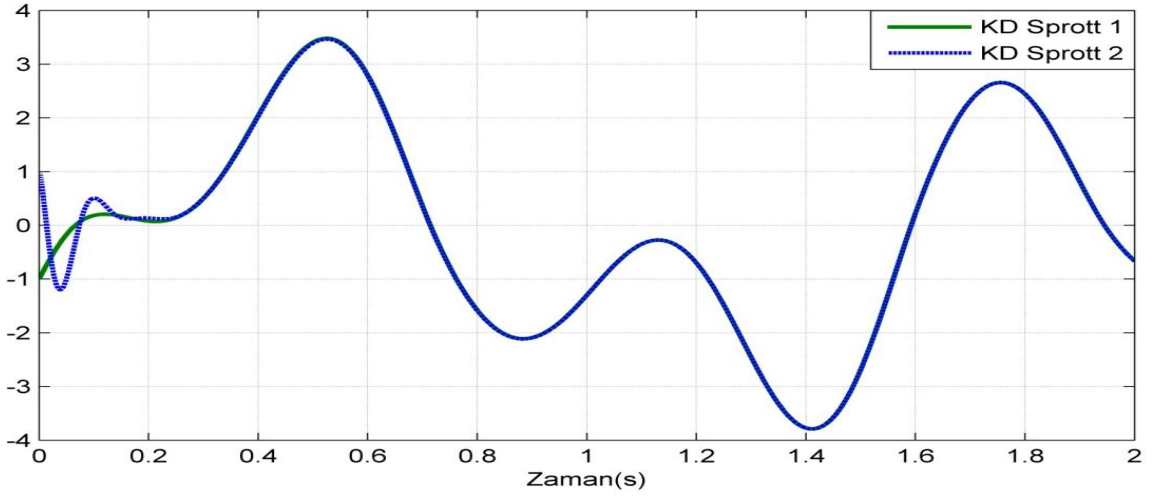


Şekil 3.43. KD Sprot kaotik sisteminin senkronizasyonu için Simulink modeli

KD Sprot master ve slave kaotik osilatörlerinin senkronizasyonu için ilk olarak uygulanan klasik PID yönteminde (Şekil 3.44'de) senkronizasyon yaklaşık 0.38s'de gerçekleştiği görülmektedir. Önerilen yöntem olan KDPID ile yapılan senkronizasyon denetiminde ise senkronizasyon süresinin yaklaşık 0.18s olduğu görülmektedir. Ayrıca her iki denetimde de simülasyon süresince senkronizasyonun bozulmadığı görülmektedir.



Şekil 3.44. PID tipi denetleyiciyle senkronizasyon denetimi yapılan iki kaotik sistemin çıkışı



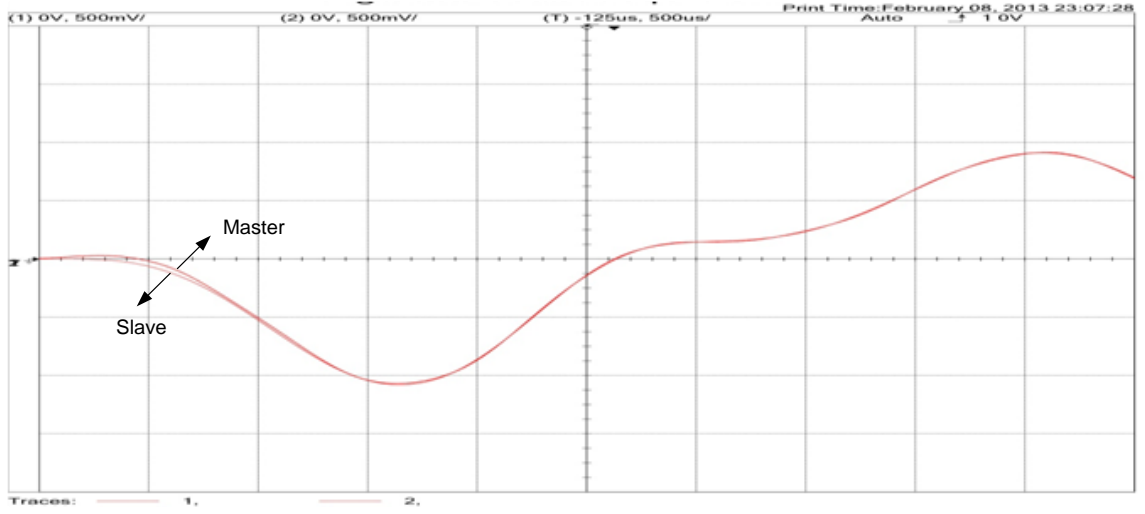
Şekil 3.45. KDPID tipi denetleyiciyle senkronizasyon denetimi yapılan iki kaotik sistemin çıkışı

3.9.2. Kesir Dereceli Sprott Sisteminin Senkronizasyonu için devre modeli

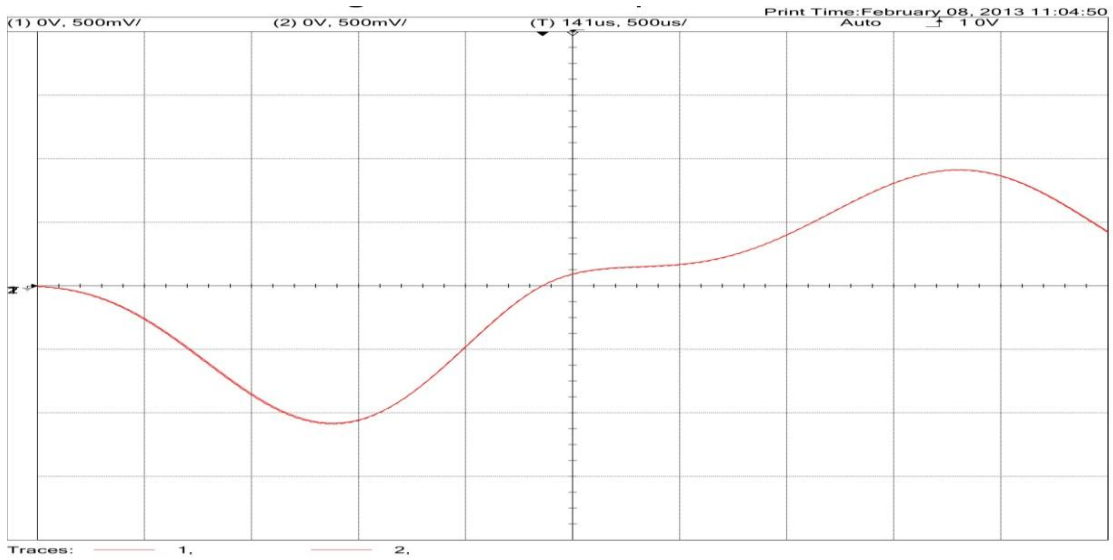
KD'li Sprott kaotik sistemin devre modeli Şekil 3.28'de gösterilmektedir. Buradan yola çıkarak KDPID ile oluşturulan senkronizasyon denetleyici modeli Şekil 3.36'da ve devre elemanları Tablo 3.4'de gösterilmektedir. Devre simülasyonundan elde edilen çıkış

değişimleri, PID için Şekil 3.46’da ve KDPID için elde edilen çıkış şekilleri Şekil 3.47’de gösterilmektedir.

Klasik PID ile yapılan denetimde senkronizasyonun tam anlamıyla 0.6ms’de gerçekleştiği gözlenirken (Şekil 3.46), KDPID ile yapılan denetimde bu süre neredeyse sıfır olduğu görülmektedir (Şekil 3.47).



Şekil 3.46. PID devre simülasyonundan elde edilen çıkış değişimleri (yatay eksen:Zaman 50ms/kare, düşey eksenler x_1 ve x_2 500mV/kare)



Şekil 3.47. KDPID devre simülasyonundan elde edilen çıkış değişimleri (yatay eksen:Zaman 500µs/kare, düşey eksenler x_1 ve x_2 500mV/kare)

3.10. Sonuç

Bu kısımda KD kaotik sistemlerden Lorenz, ve Sprott tipi kaotik sistemlerin Simulink modelleri oluşturularak kaosu oluşturduğu gözlenmiştir. Seçilen kaotik sistemlerin matematiksel modellerinden yararlanılarak devre modelleri oluşturulmuştur. Bu modellerin yanında literatürde bulunmayan KD Van der Pol devre modelinde de limit çevrim davranışı gözlenmiştir. Bu açıdan literatürde bu yaklaşım ile oluşturulan ilk devre modelidir. Ayrıca KD'li kaotik devre modellerinin senkronizasyonunu sağlamak amacıyla bu çalışmada, tasarlanan KDPID devre modelinin de senkronizasyon süresini kısalttığı gözlenmektedir. KDPID devresinin katsayı ve derecelerinin hesaplanması literatürde bu denetleyiciler için dezavantaj olarak bilinmektedir. Bu çalışmada parametrelerin ve derecelerinin belirlenmesi için FAPSO yöntemi kullanılmıştır.

4. PARÇACIK SÜRÜ ENİYİLEMESİ

Kelime anlamı, verilen duruma uygun en iyi sonuç elde etme yöntemi olarak bilinen eniyileme, karmaşık problemlerin çözümüne yardımcı olmaktadır. Literatürde eniyileme yöntemlerinin birçoğu akıllı yöntemlere dayanmaktadır. Bu yöntemler aşağıdaki gibi sıralanabilir

- Genetik algoritmalar
- Karınca Koloni
- Parçacık Sürü Eniyileme (PSO)

İlk olarak, Kennedy ve Eberhart tarafından bulunan PSO yöntemi, kuşların yiyecek araması temeline dayanır [102]. Son yıllarda mühendisliğin birçok alanındaki zorlu problemlerin çözümünde kullanılan PSO yöntemi, popülasyon temelli evrimsel bir hesaplama yöntemidir [103]. Araştırmacılar, PSO yöntemini geliştirmek için yaptıkları çalışmalar sonucunda farklı PSO yöntemleri oluşturmuşlardır. Bunları aşağıdaki gibi sıralayabiliriz.

- Uyarlamalı PSO [104],
- Bulanık Uyarlamalı PSO [105-107],
- Kaotik PSO [108]

4.1. Klasik PSO Yöntemi

İlk olarak Kennedy ve Eberhart tarafından 1990'ların ortalarında kuşların yiyecek arama davranışlarının algoritmaya dönüştürülmesi sonucunda bulunan bir yöntemdir [102]. Bu yöntemin en önemli avantajı ise; karmaşık sistemlerin çözümü için kullanılan basit bir yöntem olmasıdır. PSO yöntemi, eniyileme yaparken, sürünün üyelerinin konum ve hızlarını iyi deneyimlere göre yeniden belirlemektedir.

PSO algoritması, D boyutlu arama uzayında hareket eden parçacıkların bir kümesini kapsar ve tüm olası çözümleri içerir. Arama uzayında, P_{best} 'i.' parçacığına kadar bulunmuş

en iyi pozisyonu ifade ederken; g_{best} sürü tarafından keşfedilen küresel en iyi pozisyonudur. PSO yöntemine ait matematiksel ifade Denklem (4.1)'de verilmiştir,

$$\begin{aligned} V_i^{d+1} &= V_i^d + C_1 \cdot rand_1^d \cdot (Pbest_i^d - X_i^d) + C_2 \cdot rand_2^d \cdot (g_{best}^d - X_i^d) \\ X_i^{d+1} &= X_i^d + V_i^d \end{aligned} \quad (4.1)$$

Burada $rand_1$ ve $rand_2$ [0, 1] aralığından seçilmiş rastgele sayıları ifade ederken, C_1 ve C_2 ivmelenme sabitleridir. Araştırmacılar daha sonra bu ifadeyi Denklem (4.2)'deki gibi düzenlemişlerdir.

$$V_i^{d+1} = \omega \cdot V_i^d + c_1 \cdot rand_1^d \cdot (Pbest_i^d - X_i^d) + c_2 \cdot rand_2^d \cdot (g_{best}^d - X_i^d) \quad (4.2)$$

ω ise atalet ağırlığıdır ve Denklem (4.3)'teki gibi hesaplanır.

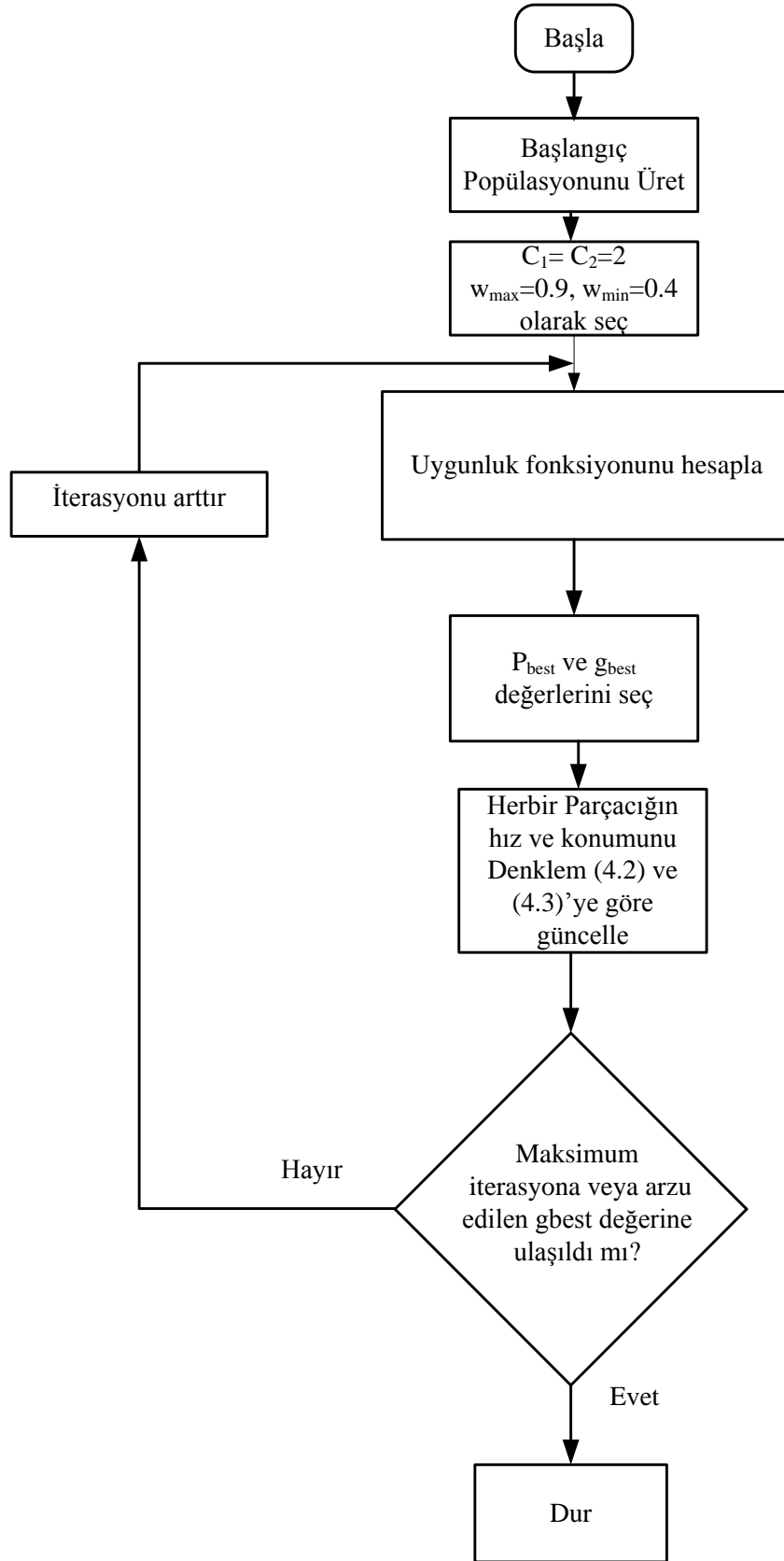
$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{i_{max}} \cdot k \quad (4.3)$$

Burada ω_{max} ve ω_{min} değerleri atalet ağırlığının sınır değerleri olup; k ise iterasyon sayısıdır. Klasik PSO'ya ait akış diyagramı Şekil 4.1'de gösterilmiştir. $\omega_{max}=0.9$, $\omega_{min}=0.4$ ve $C_1=C_2=2$ şeklinde ampirik olarak alınır.

4.2. Bulanık Uyarlamalı PSO (FAPSO)

Klasik PSO yönteminde C_1 ve C_2 sabit olarak alınırken, ω değeri ise iterasyona bağlı olarak değişir. Yapılan araştırmalar bu değerlerin uyarlamalı olarak hesaplanmasıyla daha iyi sonuç alınacağını ortaya koymuştur [104-107, 109-110]. FAPSO yöntemi klasik PSO'da kullanılan bazı sabit değerlerin her aşamada tekrardan hesaplanıp daha sonra klasik PSO'ya uygulanmasıyla oluşturulur. Klasik PSO'da bulunan bu değerlerin hesaplanması için aşağıda verilen basamaklar sırasıyla izlenir.

- Klasik PSO'da katsayıların hesaplanması
- Bulanıklaştırma işlemi
- Durulaştırma
- Uyarlama elemanın hesaplanması
- PSO sistemine uygulanması



Şekil 4.1. Klasik PSO'ya ait akış diyagramı

4.2.1. Klasik PSO'da Katsayıların Etkisi ve Hesaplanması

PSO yönteminde, C_1 değeri parçacığı en iyi pozisyona çekerken, C_2 değeri sürüyü küresel en iyi konuma çeker. Bu nedenle, FAPSO yönteminde, C_1 ve C_2 değerleri dört farklı durum için farklı değerler alır. Bunlar; *keşif durumu*, *yararlı durumu*, *yakınsama durum* ve *dışlama durumu*dur.

Keşif Durumunda: C_1 değeri arttırılırken, C_2 değeri azaltılır.

Yararlı Durumda: C_1 değeri biraz arttırılırken, C_2 değeri biraz azaltılır.

Yakınsama Durumu: C_1 ve C_2 değerleri arttırılır.

Dışlama Durumunda: C_1 değeri azaltılırken, C_2 değeri arttırılır [104].

Sistemin hangi durumda olduğuna karar verebilmek için kullanılan bulanık mantık uygulamasıyla oldukça başarılı sonuçlar elde edilmektedir. Bulanık mantıkla durum kararı için Denklem (4.4)'de verilen ifade kullanılmıştır.

$$dg^k = \frac{g(g_{best}^{k-1}) - g_{min}}{g_{max} - g_{min}} \quad (4.4)$$

Burada verilen g_{min} ve g_{max} değerleri sırasıyla; küresel en iyi çözümün minimum ve maksimum değerleri olup g_{best}^{k-1} ise, k iterasyonundan önceki küresel en iyi sonuçtur.

4.2.2. Bulanıklaştırma

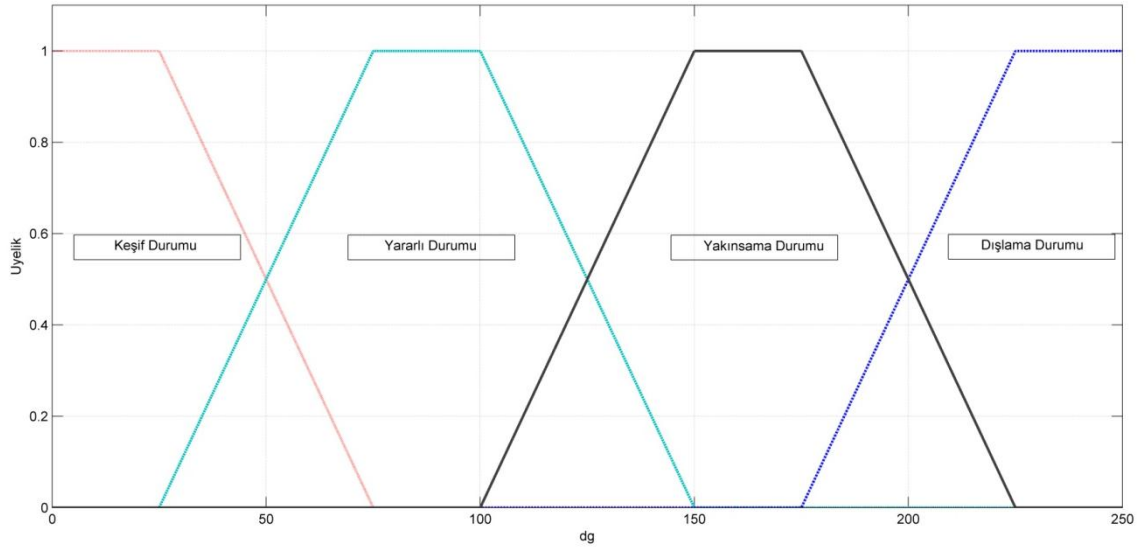
Bulanık sistem, tek girişli ve iki çıkışlı sistem olup, çıkışta C_1 ve C_2 değerlerini verir. Bulanıklaştırma işlemindeki için C_1 ve C_2 değerlerine ait üyelik fonksiyonları, Şekil 4.3 ve Şekil 4.4'de gösterilmektedir. Başlangıç durumunda C_1 değeri büyük ve C_2 değeri küçük seçilir. Sisteme ait kural tabanı aşağıda verilmiştir. Bulanık sistem, d_g 'ye göre, sistemin durumuna karar verebilmek için Şekil 4.2'de gösterilen üyelik fonksiyonundan yararlanır.

Kural 1: Eğer d_g yakınsama durumundaysa, C_1 ve C_2 değerleri biraz arttırılır.

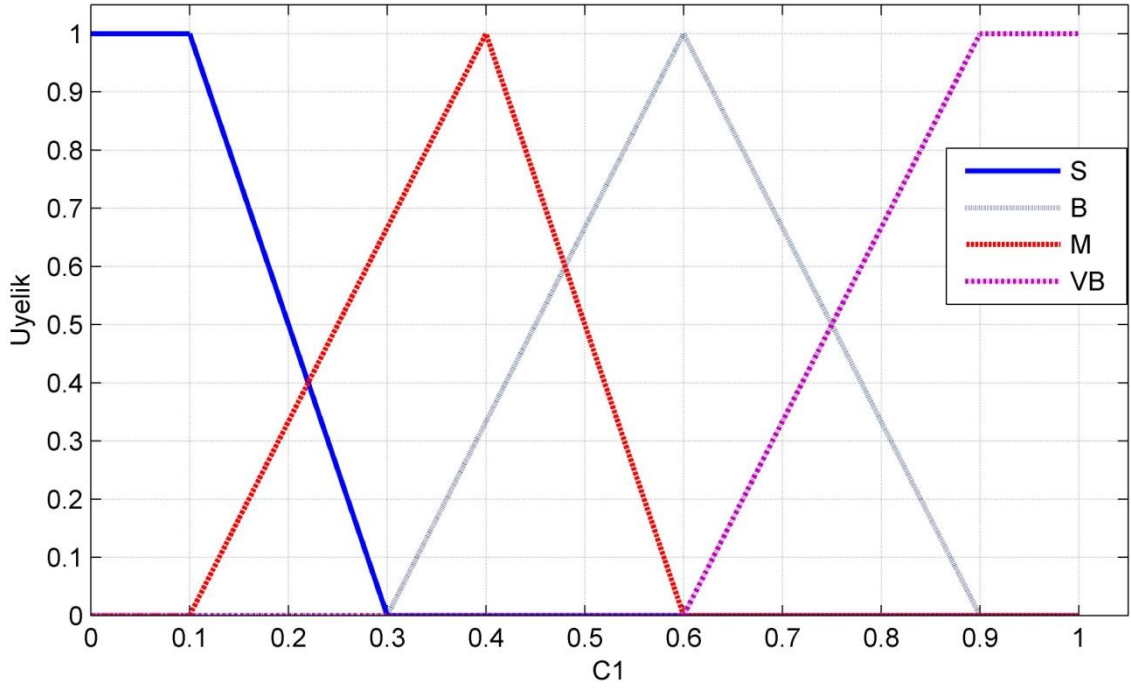
Kural 2: Eğer d_g yararlı durumdaysa, C_1 biraz arttırılırken C_2 biraz azaltılır.

Kural 3: Eğer d_g keşif durumundaysa, C_1 değeri arttırılırken C_2 değeri azaltılır.

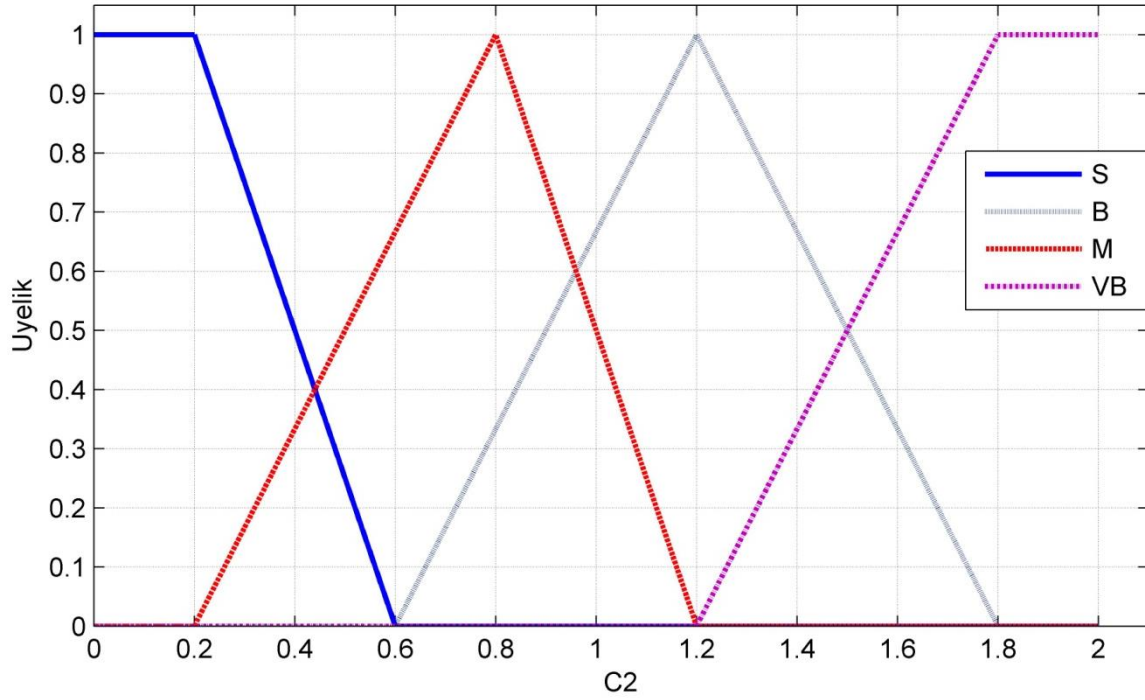
Kural 4: Eğer d_g dışlama durumundaysa, C_1 değeri azaltılırken, C_2 değeri arttırılır.



Şekil 4.2. d_g girişe ait üyelik fonksiyonu



Şekil 4.3. C_1 'e ait üyelik fonksiyonu



Şekil 4.4. C₂'ye ait üyelik fonksiyonu

4.2.3. Durulaştırma

Durulaştırma işlemi için literatürde birçok çalışma bulunmaktadır. Bunlardan bazıılarını aşağıdaki gibi sıralayabiliriz [111].

- Maksimum üyelik yöntemi
- Ağırlık merkezi yöntemi
- Ağırlık ortalamaları yöntemi

Maksimum üyelik yöntemi: En büyük üyelik fonksiyonuna göre çıkış veren bir yöntemdir. Hızlı bir yöntem olarak bilinen bu yöntemde ait matematiksel ifade Denklem (4.5)'de verilmiştir. Burada x^* durulaştırma çıkışı olarak tanımlanır.

$$x^* = \mu(x') \geq \mu(x) \quad x \in X \quad (4.5)$$

Ağırlık Merkezi Yöntemi: Sugeno tarafından geliştirilmiş ve en yaygın kullanılan yöntemlerden biridir. Bu yöntemin tek dezavantajı, karmaşık üyelik fonksiyonları için durulaştırma hesaplanması zordur [112]. Bu yöntemde ait matematiksel ifade Denklem (4.6)'da verilmektedir.

$$x^* = \frac{\int \mu(x).xdx}{\int \mu(x)dx} \quad (4.6)$$

Ağırlık Ortalaması Yöntemi: Sistemde kayıtlı olan kurala tablosunun her bir çıkışında bulunan ağırlıkların ortalamasının alınmasıyla elde edilir. Bu yöntemle ait matematiksel ifade Denklem (4.7)'de verilmiştir. Burada m_i her bir kurala ait çıkış üyeliği olup, w_i ise bu kurala ait ağırlık değeridir.

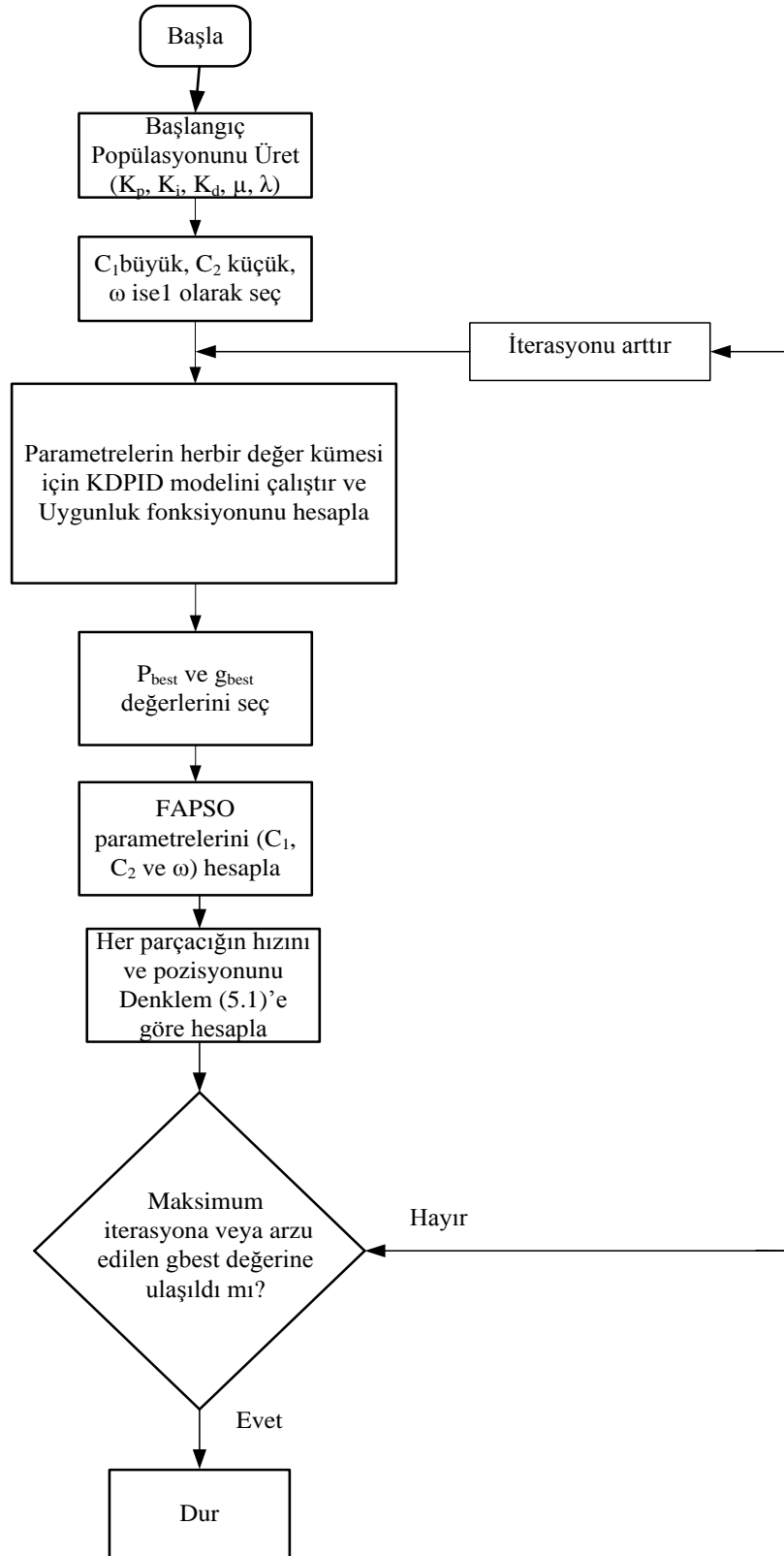
$$x^* = \frac{\sum_{i=1}^n m_i w_i}{\sum_{i=1}^n m_i} \quad (4.7)$$

4.2.4. Uyarlama Elemanının Hesaplanması

FAPSO yöntemindeki ω değeri, Denklem (4.8)'de verildiği gibi uyarlamalı olarak değişir. σ ve ε değerleri sabit olup sırasıyla 1.5 ve -2.6 gibi değerler alır [1055].

$$\omega(dg) = \frac{1}{1 + \sigma.e^{\varepsilon.dg}} \quad (4.8)$$

FAPSO yöntemine ait akış diyagramı Şekil 4.5'de gösterilmiştir.



Şekil 4.5. FAPSO yöntemi için akış diyagramı

4.3. KDPID Tipi Denetleyicinin Parametre ve Derecelerinin FAPSO ile Eniyileme

Bu bölümde, beşinci bölümde ele alındığı gibi kuş ve balık sürülerinin davranış temelli olan PSO yönteminin yeni bir yaklaşımı olan FAPSO kullanılarak, KDPID denetleyicisinin parametre optimizasyonuna uygulanmıştır.

4.3.1. Denetleyici Parametrelerinin PSO Yöntemiyle Eniyilemesi

KDPID denetleyicilerinin parametrelerinin eniyi değerlerini bulmak için burada klasik PSO yöntemi kullanılabilir. Klasik PSO yöntemiyle KDPID parametrelerinin belirlenmesi için adımlar Tablo 4.1 'de verilmiştir.

Tablo 4.1. PSO eniyileme adımları

1. Başlangıç K_p , K_d , K_i , λ ve μ değerlerini verilen sınırlar içinde üret
2. $C_1=C_2=2$, $\omega_{max}=0.9$ ve $\omega_{min}=0.4$ olarak al
3. KDPID parametre seti için denetleyici modelini çalıştır
4. Uygunluk fonksiyonunu, P_{best} ve g_{best} değerlerini hesapla
5. Her bir parametre için hız ve konum bilgisini hesapla
6. Parametre değerlerini ayarla ve modeli çalıştır
7. g_{best} değerini hesapla, arzu edilen değere ulaşıldıysa veya maksimum iterasyona ulaşıldı mı? Ulaşıldıysa eniyilemeyi bitir-Ulaşılmadıysa 2. adıma geç
8. Eniyilemeyi bitir.

Klasik PSO yönteminde (Tablo 4.1'de) 2. adımda bulunan C_1 , C_2 ve ω değeri sabit olarak alınır. C_1 , C_2 ve ω değeri ampirik olarak seçilmiştir. Bu değerlerin iterasyona göre değiştirilmesi halinde, daha sağlıklı sonuç elde edildiği literatürde ki çalışmalarda ispatlanmıştır [104-107].

4.3.2. Denetleyici Parametrelerinin FAPSO Yöntemiyle Eniyilemesi

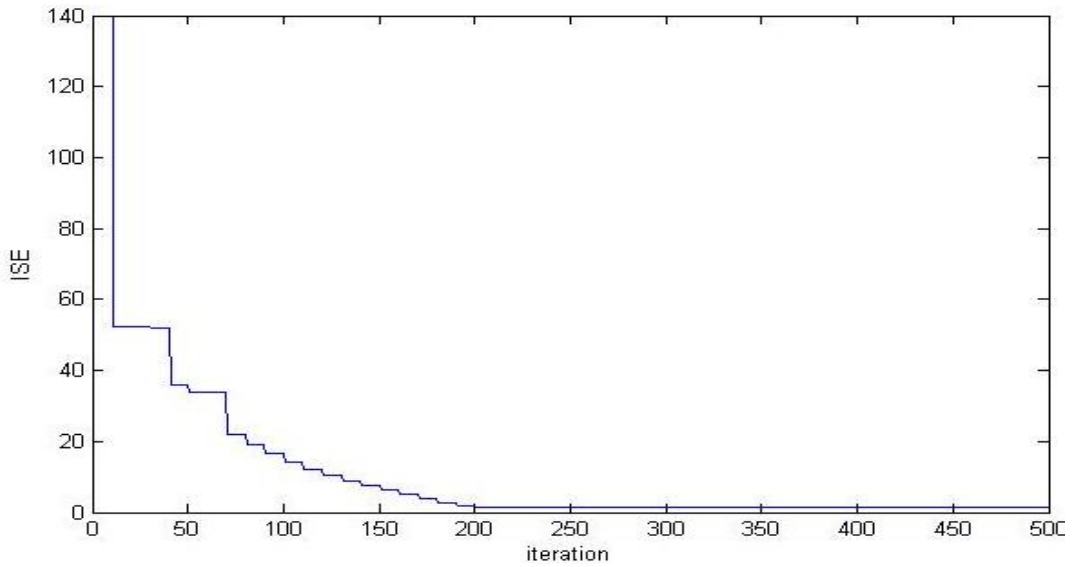
Bu çalışmada kullanılan KDPID denetleyicisinin parametrelerinin FAPSO yöntemiyle bulunabilmesi için C_1 , C_2 ve ω değerlerinin değişken olması gereklidir [103]. Bunun için sistem basamakları Tablo 4.2'de gösterilmektedir.

Buradan yola çıkarak yapılan eniyileme işleminde yakınsama eğrisi Şekil 4.6'da gösterilmektedir. Bu yakınsama eğrisinden görüldüğü üzere yakınsama yaklaşık 200.

iterasyonda tamamlanmıştır. Bu da beş parametrenin eniyilemesinde, literatürde daha az parametre için (klasik PID de üç parametre için yapılır) yapılan çalışmalarla karşılaştırıldığında iyi bir sonuç verdiği görülmektedir [113].

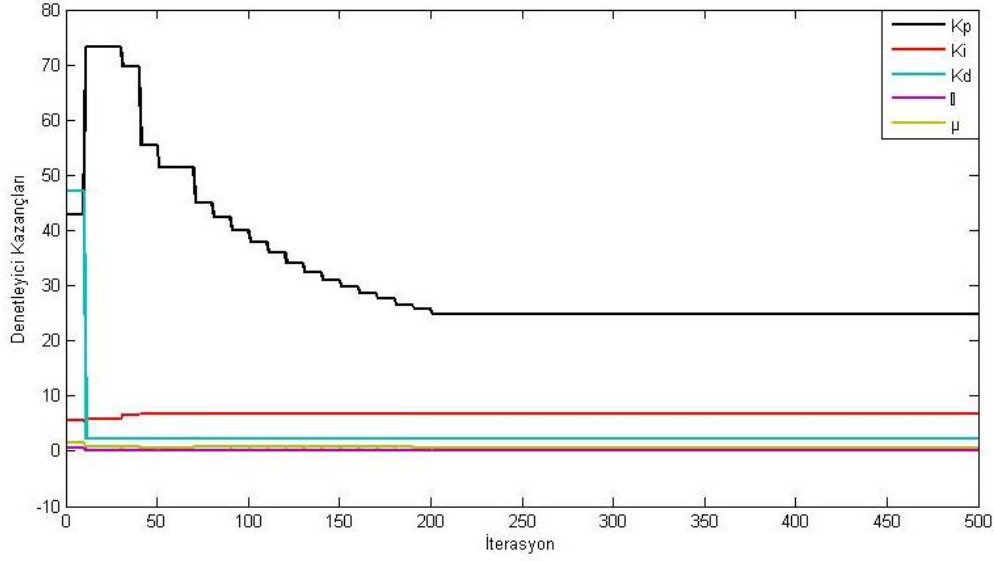
Tablo 4.2. FAPSO eniyileme adımları

1. Başlangıç K_p , K_d , K_i , λ ve μ değerlerini verilen sınırlar içinde üret
2. C_1 değerini büyük C_2 değerini küçük ve ω değerini yaklaşık 1 olarak al
3. KDPID parametre seti için denetleyici modelini çalıştır
4. Uygunluk fonksiyonunu, P_{best} ve g_{best} değerlerini hesapla
5. Her bir parametre için hız ve konum bilgisini hesapla
6. C_1 ve C_2 değerlerini bulanık kural tablosuna ve Şekil 4.3 ve Şekil 4.4’de verilen üyelik fonksiyonuna göre hesapla
7. ω değerini Denklem (4.8)’de verilen ifadeye göre uyarlamalı olarak hesapla
8. Parametre değerlerini ayarla ve modeli çalıştır
9. g_{best} değerini hesapla, arzu edilen değere ulaşıldıysa veya maksimum iterasyona ulaşıldı mı? Ulaşıldıysa eniyilemeyi bitir-Ulaşılmadıysa 2. adıma geç
10. Eniyilemeyi bitir.



Şekil 4.6. FAPSO yakınsama eğrisi

KDPID denetleyicisinin parametrelerinin değişimi Şekil 4.7’de gösterilmektedir. Bu parametrelerin alacağı eniyi değer aşağıdaki Tablo 4.3’de verilmiştir.



Şekil 4.7. Denetleyici kazanç parametreleri ve derecelerinin değişimi

Tablo 4.3. KDPID denetleyicisinin parametre ve dereceleri

K_p	24.8977
K_i	6.8200
K_d	2.2173
λ	0.1157
μ	0.5128

4.4. Sonuç

Bu bölümde KD kaotik sistemlerin (Lorenz, Van der Pol ve Sprot) senkronizasyon denetiminde kullanılan KDPID' nin parametre ve derecelerinin belirlenmesi için FAPSO yöntemi kullanılmıştır. Bu yöntem literatürde birçok farklı sistemler üzerinde olumlu sonuç verdiği gözlenmiştir [105-107]. Benzer şekilde KDPID denetleyicisinin parametrelerinin belirlenmesi için kullanılan eniyileme yönteminin avantaj sağladığı gözlenmektedir. Elde edilen sonuçlarda toplam karesel hatanın yaklaşık 200. iterasyonda minimum olduğu ve bu noktadaki KDPID' nin parametre ve derecelerinin değerleri Tablo 4.3'deki değerler olduğu görülmektedir. Yakınsama eğrisi incelendiğinde literatürdeki çalışmalara nazaran hızlı bir senkronizasyon gerçekleştiği gözlenmiştir.

5. KAOTİK HABERLEŞME SİSTEMLERİ

Kaosun geniş bantlı doğası nedeniyle taşıyıcı olarak, yayılı spektruma sahip haberleşme sistemlerinde kullanılır [114]. Kaosun haberleşme sistemlerinde kullanımı bazı avantajlara sahiptir. Bunlar geniş bantlı kaos işaretleri basit devrelerle üretildiği için donanım olarak gerçekleştirilmesi basittir ve kaosun periyodik olmayan işaretlere sahip olması nedeniyle mesaj güvenliği açısından avantaj sağlar [115].

Kaos temelli haberleşme sistemleri üzerinde son yıllarda birçok çalışma yapılmıştır. Bu yöntemleri üç başlık altında toparlanabilir. Bunlar; kaotik analog modülasyon yöntemi, kaotik sayısal haberleşme yöntemi ve dizi yayılı haberleşme sistemi.

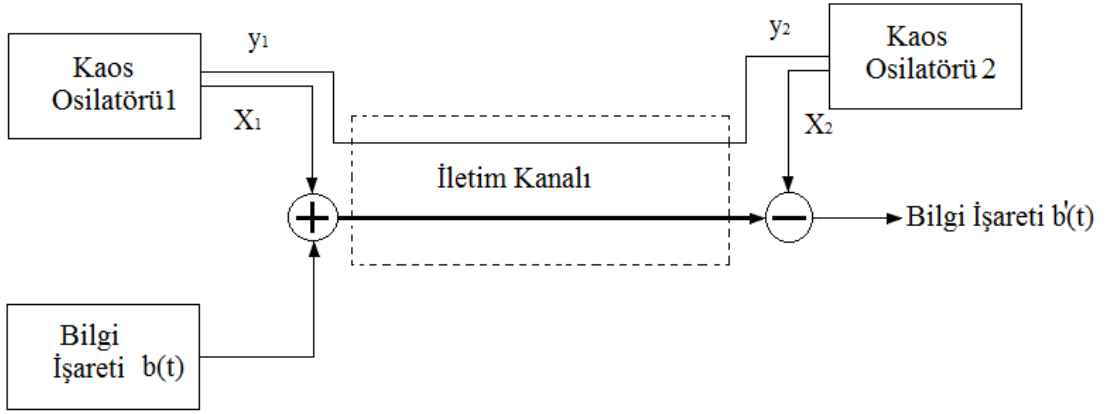
5.1. Kaotik Analog Modülasyon Yöntemi

Analog işaretlerin kaos tabanlı bir sistemle iletilmesi yöntemi 90'lı yılların başlarında gerçekleştirilmiştir. Temel olarak kaotik analog modülasyon yöntemi ikiye ayrılır. Bunlar;

- Kaotik Maskeleyme
- Kaotik Modülasyon

5.1.1. Kaotik Maskeleyme Yöntemi

Bilgi işareti, kaotik bir işaretle toplanarak iletişim kanalına aktarılması yöntemine kaotik maskeleyme yöntemi denir. Senkronizasyon temelli bir modülasyon olan kaotik maskeleymede alıcıda bulunan kaotik sistem ile alıcıda bulunan kaotik osilatör senkronize olmalıdır [115-116]. Aksi takdirde işaretin tekrardan elde edilmesi mümkün olmadığı için kaotik maskeleyme bir açıdan güvenli haberleşme sağlamaktadır. Maskeleyme yönteminin blok şeması Şekil 5.1'de gösterilmektedir. Buradan da görüldüğü gibi kaos osilatörünün bir değişkeni doğrudan hatta verilirken diğer bir değişken bilgi işaretiyle toplanarak başka bir haberleşme kanalına verilir. Kanaldan alınan bu iki veriden başta ki kaosun senkronizasyonu için kullanılır.



Şekil 5. 1. Kaotik maskeleye blok diyagramı

Maskelemiş verinin yeniden elde edilebilmesi için senkronizasyon kontrolü yapılması gereklidir. Burada senkronizasyon kontrolünün performansı ve dayanıklılığı haberleşme sisteminin kalitesini doğrudan etkilemektedir.

5.1.2. Kaotik Modülasyon

Kaotik modülasyon yönteminde, kaotik haritanın parametrelerine bilgi eklenerek toplanır. Kaotik harita sürekli değişerek bilgi işaretinin güvenliğini sağlar. Bu yöntemin en önemli avantajı senkronizasyon gerektirmemesidir. Demodülasyon için üç temel yaklaşım vardır;

- Uyarlamalı filtrelerle lineer demodülasyon yöntemi [116-117]
- İncersiyon yaklaşımı [118]
- Yapay sinir ağlarıyla demodülasyon [119-120]

Bu yöntemlerden basit olanı incersiyon yöntemidir ama bu yöntemin gürültü dayanımı açısından performansı sınırlıdır. Yapay sinir ağları bu açıdan diğer iki yöntemle göre daha avantajlıdır. Gürültü dayanımı açısından var olan yöntemler yeterli performans göstermedikleri için kullanımı yaygınlaşmamıştır.

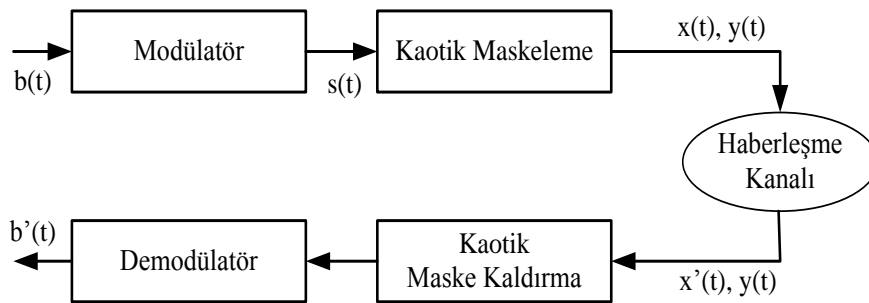
5.2. Kaotik Sayısal Modülasyon Yöntemleri

Kaotik sayısal modülasyon yöntemi, kaotik analog modülasyon yöntemleriyle karşılaştırıldığında kullanımı daha pratiktir. Ayrıca bu yöntemlerin başarımı, diğer sayısal haberleşme yöntemlerine göre performans açısından çeşitli üstünlüğe sahiptir. Bu nedenle, son yıllarda kaos temelli sayısal haberleşme sistemleri üzerine birçok çalışma yapılmaktadır. Literatürde var olan bazı kaotik sayısal haberleşme yöntemleri aşağıdaki gibi sıralanabilir.

- Kaotik açma-kapama anahtarlanması (COOK) [37]
- Kaotik kaydırmalı anahtarlama (CSK) [38]
- Farksal kaotik kaydırmalı anahtarlama (DCSK) [39]
- Korelasyon gecikmeli kaotik kaydırmalı anahtarlama (CDSK) [40]
- Kaotik maskeli faz kaydırmalı anahtarlama (K-PSK) [41]

5.2.1. Kaotik Maskeli Faz Kaydırmalı Anahtarlama (K-PSK)

Klasik faz kaydırmalı anahtarlama yönteminde taşıyıcının kaotik olarak maskelenmesi mantığına dayanır. Klasik modülasyon yöntemlerinden farklı olarak, analog kaotik maskeli haberleşme yöntemine benzeyen bu yöntem basit olması en önemli avantajıdır. Yönteme ait blok diyagramı Şekil 5.2'de verilmiştir. Bu şekilde veri modülatör kaotik maskeleme işleminden sonra haberleşme kanalına verilir tabi aynı zamanda kaotik değişkenlerden biride senkronizasyon denetimi için farklı bir kanala aktarılır.

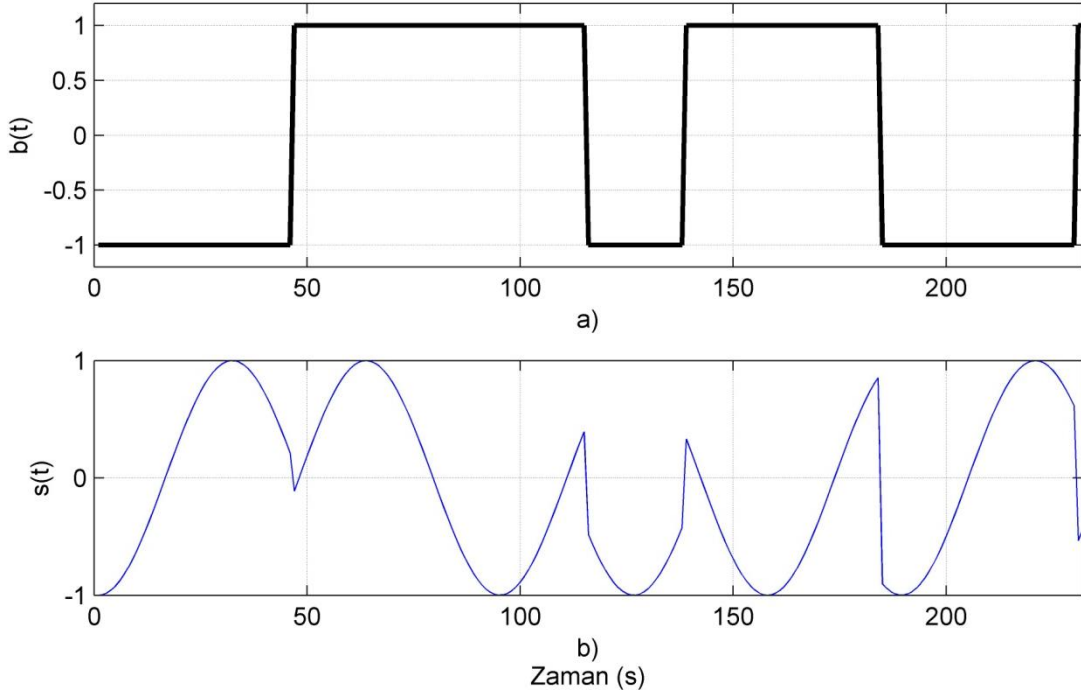


Şekil 5.2. K-PSK yöntemine ait blok diyagramı

Modülatör: Yönteme ait matematiksel ifade Denklem (5.1)'de verilmiştir [41].

$$s(t) = \begin{cases} \cos \frac{2\pi}{T} t & \text{eğer; } b(k) = 1 \\ -\cos \frac{2\pi}{T} t & \text{eğer; } b(k) = 0 \end{cases} \quad (5.1)$$

$b(t) = 0011101100$ için veri işareti ve modüle edilmiş işaret Şekil 5.3'de gösterilmiştir.



Şekil 5.3. a) $b(t) = 0011101100$ için veri işareti b) modüle edilmiş işaret

Kaotik Maskeleye: kaotik maskeleye yönteminde veri kaotik değişkenlerden biri ile toplanarak iletim kanalına aktarılır. Maskeleye işlemi Denklem (5.2)'de verilen ifadeye göre yapılır.

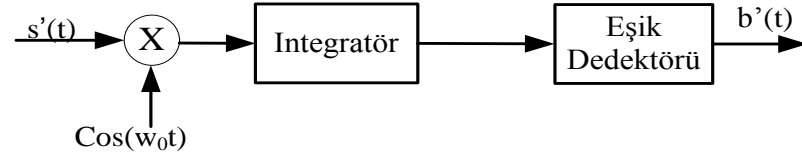
$$\phi(t) = x(t) + s(t) \quad (5.2)$$

Kaotik maskeli olarak kanala aktarılan verinin tekrardan elde edilebilmesi için kaotik değişkenlerden bir diğeri de iletim kanalına aktarılır.

Maske Kaldırma: maske kaldırma işleminde; alıcı ile senkronize olmuş kaotik osilatörün değişkenlerinden biri, iletim kanalından gelen bilgi işaretinin saklı olduğu değişkenden çıkarılır. Bu sisteme ait matematiksel ifade Denklem (5.3)'de görülmektedir.

$$\begin{aligned} s'(t) &= \phi'(t) - x'(t) \\ x'(t) &= x(t) \end{aligned} \quad (5.3)$$

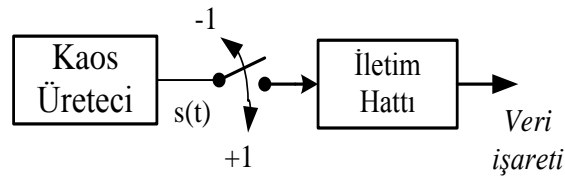
Demodülasyon: Demodülasyon yönteminde iki yaklaşım mevcuttur. Bunlardan biri senkronizasyon temelli evre uyumlu ve evre uyumsuz demodülasyondur. Evre uyumlu demodülasyon yöntemine ait blok diyagramı Şekil 5.4'de gösterilmektedir [41].



Şekil 5.4. Evre uyum demodülasyon yöntemine ait blok diyagramı

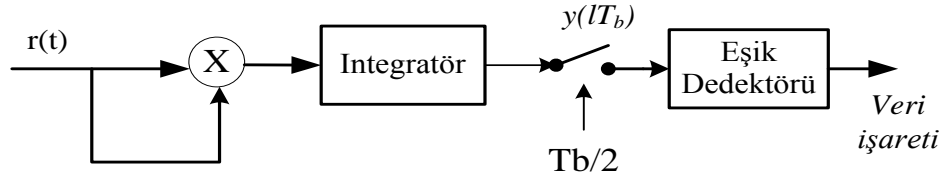
5.2.2. Kaotik Açma-Kapama Modülasyonu (COOK)

Modülasyon: Kaotik açma kapama modülasyona ait blok diyagramı Şekil 5.5'de gösterilmektedir. İşaret +1 olduğu zaman anahtar kapanır ve kaotik işaret gönderilir, eğer işaret -1 olursa anahtar açılır ve herhangi bir işaret gönderilmez [37].



Şekil 5.5. COOK modülatörüne ait blok diyagramı

Demodülasyon: Demodülasyon için kullanılan evre uyumsuz demodülasyon yöntemi bit enerji tahminine dayalı demodülasyonu yöntemidir. Demodülasyona ait blok diyagramı Şekil 5.6'da gösterilmektedir [37].



Şekil 5.6. COOK için evre uyumsuz demodülatör

Blok diyagramında $y(IT_b)$ ifadesinin Denklem (5.4)'de verildiği gibi olduğu görülmektedir.

$$y(IT_b) = \int_{(l-1)T_b}^{lT_b} r^2(t) dt \quad (5.4)$$

Burada gürültü ihmal edilirse Denklem (5.5)'de verilen ifade elde edilir.

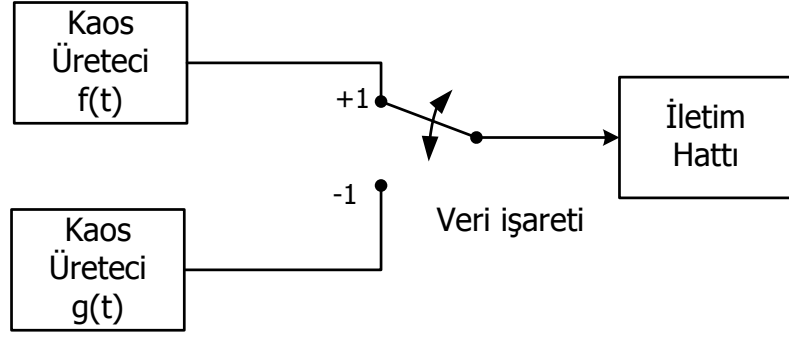
$$y(IT_b) = \int_{(l-1)T_b}^{lT_b} s^2(t) dt \quad (5.5)$$

Bu ifade eşik girişine uygulandığında detektörün çıkışında Denklem (5.6)'da verilen ifadeye göre karar verilir.

$$\text{Alınan veri} = \begin{cases} +1 & \int_{(l-1)T_b}^{lT_b} s^2(t) dt \\ -1 & 0 \end{cases} \quad (5.6)$$

5.2.3. Kaotik Kaydırmalı Anahtarlama (CSK)

Modülasyon: Kaos kaydırmalı anahtarlama (CSK) yönteminde iki kaotik osilatör arasında verinin +1 olması durumunda f kaotik osilatör, -1 olması durumunda g kaotik osilatör veri gönderir. CSK yöntemine ait blok diyagramı Şekil 5.7'de gösterilmiştir [38].

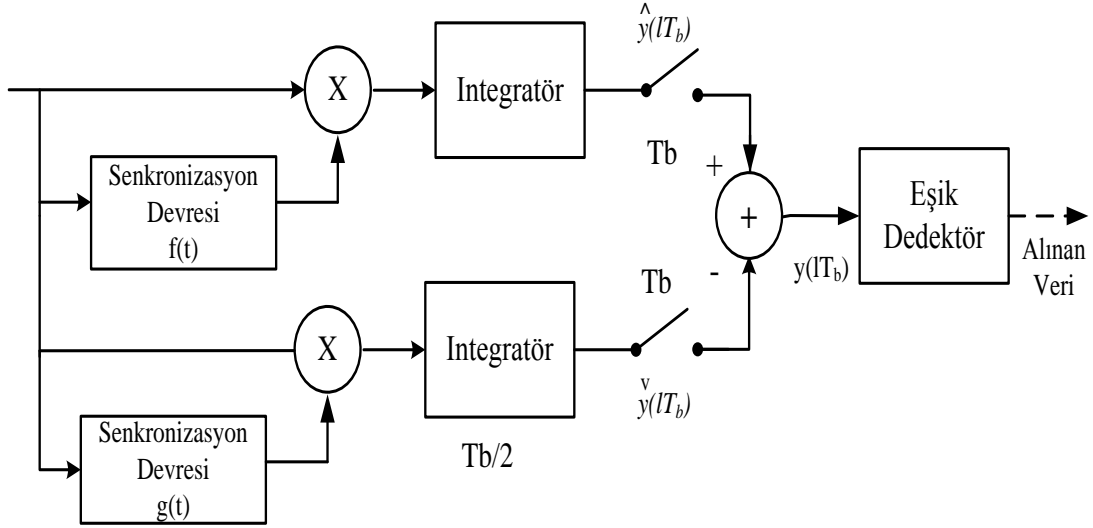


Şekil 5.7. CSK Modulatörüne ait blok diyagramı

Modülatöre ait matematiksel ifade Denklem (5.7)'de verilmiştir.

$$s(t) = \begin{cases} f(t) & +1 \\ g(t) & -1 \end{cases} \quad (5.7)$$

Demodülasyon: literatürde iki farklı yöntem vardır. Bunlar senkronizasyon hatasına göre ve korelatör tipi demodülatördür. Korelatör tipi demodülatöre ait blok diyagramı Şekil 5.8'de gösterilmektedir.



Şekil 5.8. Korelatör temelli CSK demodülatör

CSK demodülatörüne ait Şekil 5.8'de verilen blok diyagramından da görüldüğü gibi matematiksel olarak ifade edersek Denklem (5.8) elde edilir.

$$\hat{y}(lT_b) = \int_{(l-1)T_b}^{lT_b} r(t)f(t).dt \quad (5.8)$$

$$\check{y}(lT_b) = \int_{(l-1)T_b}^{lT_b} r(t)g(t).dt$$

Burada $y(lT_b)$ ise Denklem (5.9)'da verilmiştir.

$$y(lT_b) = \hat{y}(lT_b) - \check{y}(lT_b) \quad (5.9)$$

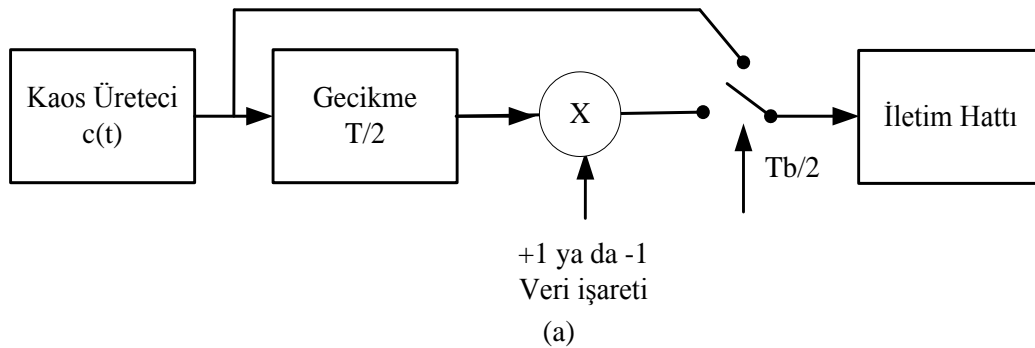
5.2.4. Farksal Kaotik Kaydırmalı Anahtarlama (DCSK)

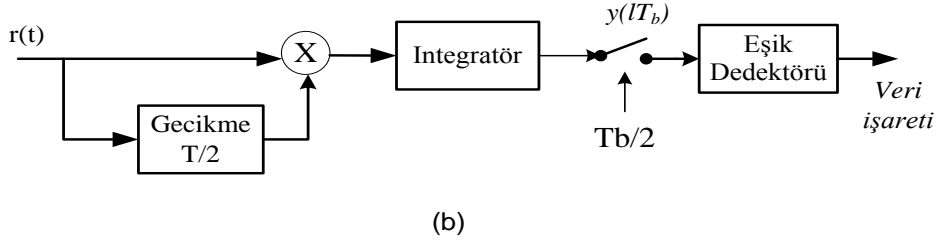
CSK yönteminde bit enerjisine göre tahmin yapılırken, gürültü nedeniyle birçok hata oluşabilir. Bu sorunu gidermek için farksal kaotik kaydırma mantığına dayanan DCSK yöntemi üzerinde çalışmalar sürmüştür [37-39, 41]. DCSK modülatör ve demodülatöre ait blok diyagramı Şekil 5.9'da gösterilmektedir.

Demodülatör çıkışında ki ifade Denklem (5.10)'da verilmiştir.

$$y(lT_b) = \int_{(l-1/2)T_b}^{lT_b} r(t)r(t-T/2)dt \quad (5.10)$$

Gürültüsüz ortamda $r(t)=s(t)$ olduğuna göre Denklem (5.11)'de integratör girişi verilmiştir. Denklem (5.12)'de ise karar eşik dedektörünün çıkışı görülmektedir.





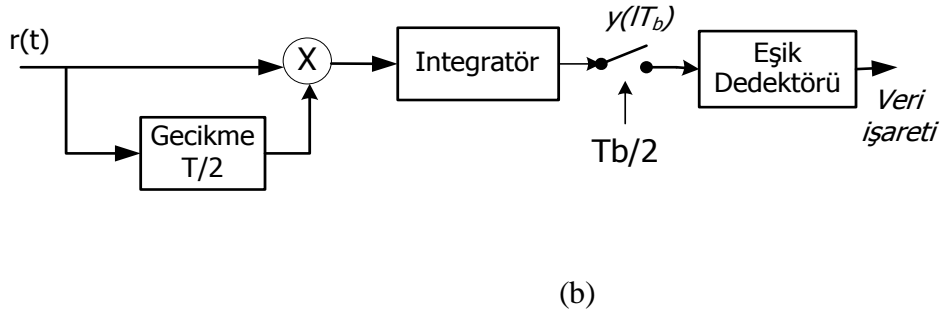
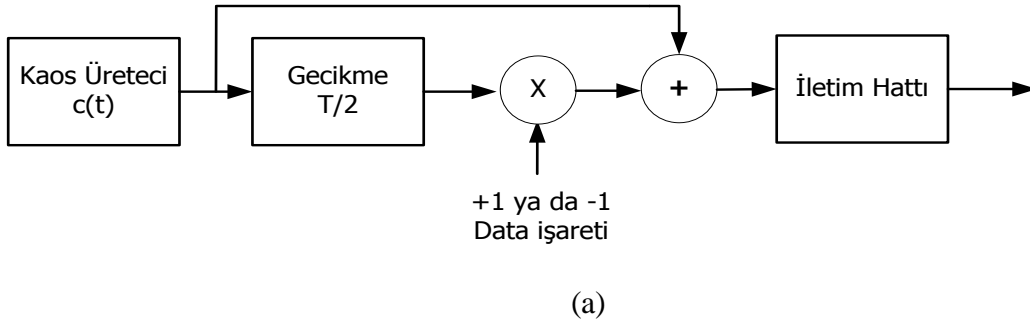
Şekil 5.9. (a) DCSK Modülör ve (b) demodülör

$$y(lT_b) = \int s(t)s(t - T_b / 2)dt \quad (5.11)$$

$$y(lT_b) = \begin{cases} \int_{(l-1)T_b}^{(l-1/2)T_b} c^2(t)dt & +1 \text{ gönderildiğinde} \\ - \int_{(l-1)T_b}^{(l-1/2)T_b} c^2(t)dt & -1 \text{ gönderildiğinde} \end{cases} \quad (5.12)$$

5.2.5. Korelasyon Gecikmeli Kaydırmalı Anahtarlama (CDSK)

DCSK haberleşmesinde, sembol süresinin yarısında bilgi taşınmadığı için verimli bir yöntem değildir. Bu nedenle Sushchik tarafından korelasyon gecikmeli kaydırmalı anahtarlama (CDSK) yöntemi geliştirilmiştir [40]. CDSK yöntemine ait blok diyagramı Şekil 5.10'da gösterilmiştir.



Şekil 5.10. (a) CDSK modülatör, (b) demodülatör

CDSK yönteminde, şekilden de görüldüğü gibi kaotik veri ile gecikmeye uğratılmış değerinin veri işaretiyle ilişkilendirilerek iletim kanalına verilir. Sisteme ait matematiksel ifade Denklem (5.13)'teki gibidir.

$$s(t) = \begin{cases} c(t) + c(t - \tau) & +1 \\ c(t) - c(t - \tau) & -1 \end{cases} \quad (5.13)$$

Demodülatördeki integratör çıkışı Denklem (5.14)'de verilmiştir.

$$y(IT_b) = \int_{(l-1)T_b}^{IT_b} r(t)r(t - \tau)dt \quad (5.14)$$

Gürültüsüz ortamda integratör çıkışı Denklem (5.15)'de verildiği gibidir.

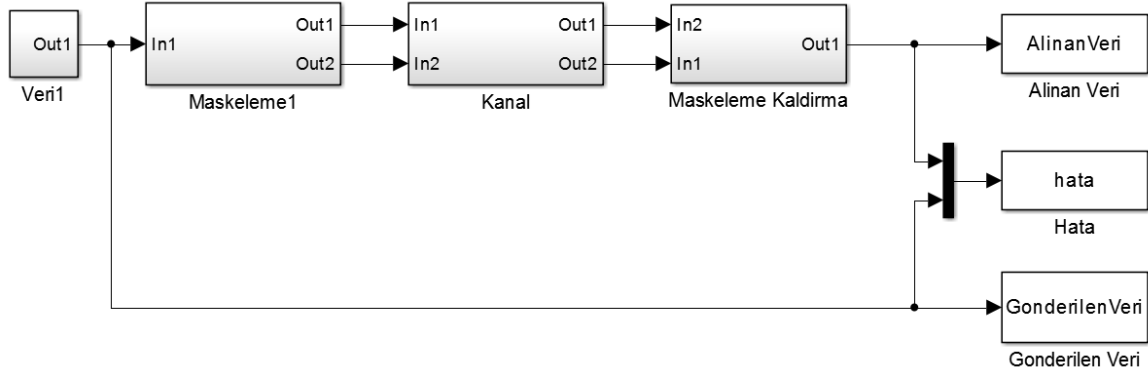
$$y(IT_b) = \int_{(l-1)T_b}^{IT_b} s(t)s(t - \tau)dt = \int_{(l-1)T_b}^{IT_b} [c(t) + c(t - \tau)][c(t - \tau) \pm c(t - 2\tau)]dt \quad (5.15)$$

5.3. KDPID Denetleyicisinin Kesir Dereceli Kaotik Haberleşme Sistemlerine Uygulanması

Literatürde birçok farklı kaotik haberleşme yöntemi bulunmaktadır. Bunlar analog ve sayısal yöntemler olarak ikiye ayrılmaktadır. Analog yöntemlerin başında kaotik maskeleme yöntemi gelmektedir. Sayısal yöntemler, analog yöntemlere nazaran daha çeşitlidirler. K-PSK, COOK, CSK, DCSK, CDSK ve CDSK yöntemleri kaotik sayısal modülasyon yöntemlerindedir [37-41, 94-101, 121-124].

5.3.1. Kaotik Maskeli Analog Haberleşme

Literatürde uzun yıllardır kullanılan bu yöntem, senkronizasyon temelli olup denetleyici performansı ve kanal gürültüsünden doğrudan etkilenir [125-127]. Bu nedenle denetleyicinin gürültü performansı yüksek olmalıdır. Çalışmada hızlı ve dayanıklı bir senkronizasyon yöntemi olan KDPID yöntemi kullanılarak senkronizasyon denetimi yapılmıştır. KDPID ile yapılan denetleyiciye ait Simulink modeli Şekil 5.11'de gösterilmektedir.

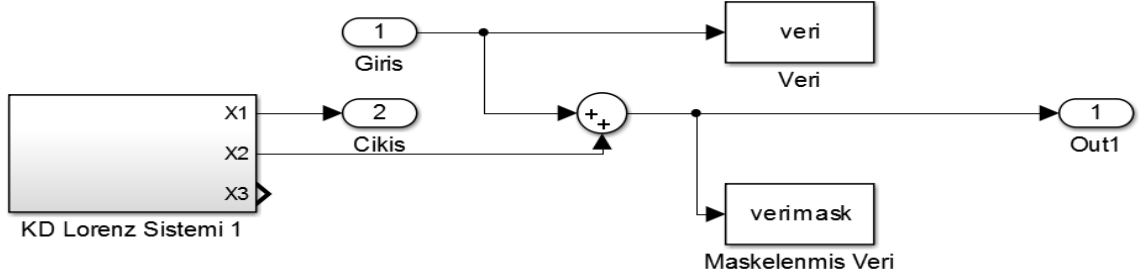


Şekil 5. 11. KDPID ile yapılan denetleyiciye ait Simulink modeli

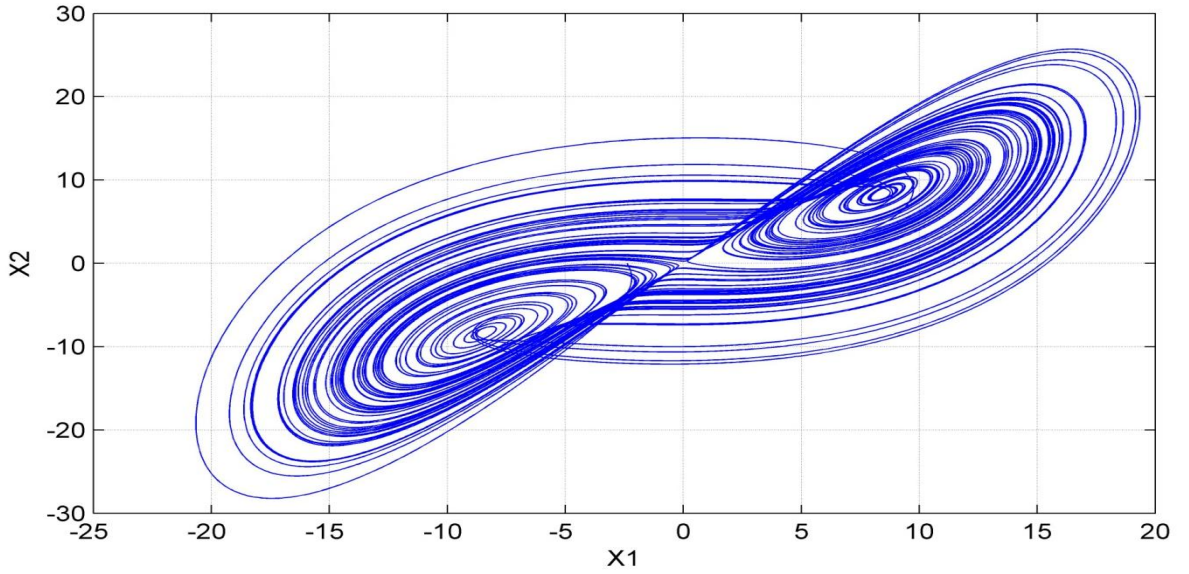
Modelden görüldüğü gibi bu haberleşme sisteminde veri; maskeleme, kanal, maske kaldırma işlemi olmak üzere üç aşamadan geçerek tekrardan oluşur.

Maskeleme: Kaotik maskeli haberleşme sisteminde Şekil 5.11'de görülen şekilde maskeleme bloğunun iç yapısı Şekil 5.12'de gösterilmektedir. Burada Lorenz kaotik

sistemi kullanılmıştır. Kaotik sinyallerden biri (x_2), bilgi işaretiyle toplanarak haberleşme kanalına aktarılırken diğer çıkış (x_1) doğrudan kanala aktarılmıştır. Kaotik sinyallerin oluşturduğu kaotik çeki Şekil 5.13’de ve kaotik sinyal ile veri işaretinin toplamı Şekil 5.14’de gösterilmektedir.

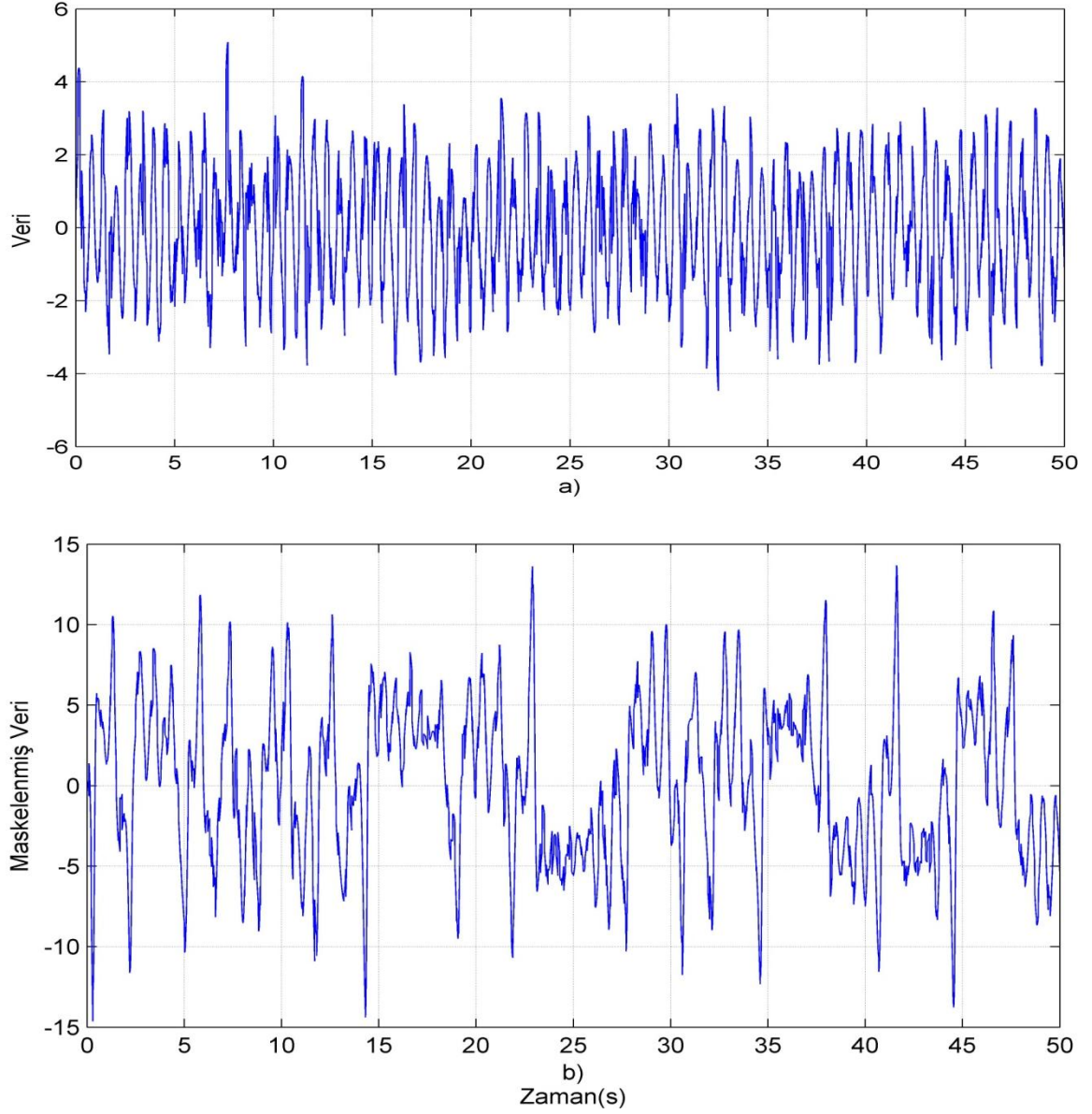


Şekil 5.12. Maskeleye bloğunun içyapısı



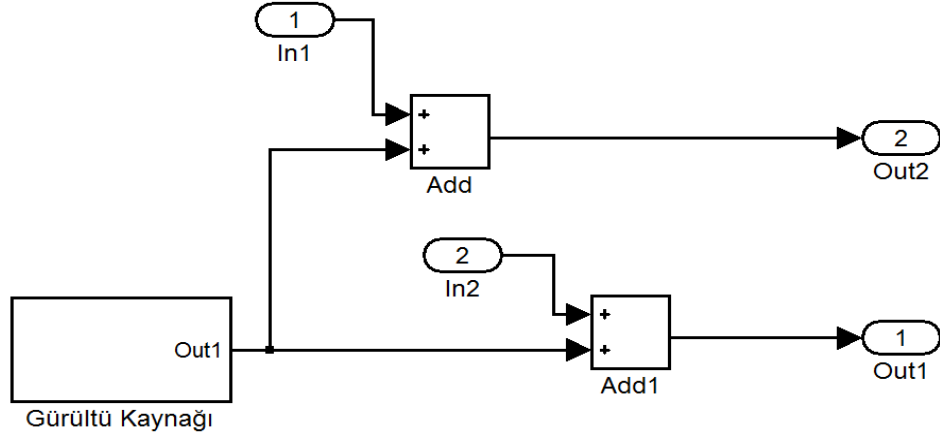
Şekil 5.13. Kaotik sinyallerin oluşturduğu kaotik çeki

Bu her iki şekilden veri kaotik sinyal içine gizlendiği görülmektedir. Yani verinin tekrardan elde edilebilmesi için bu kaotik sinyalin karşı tarafta tekrardan oluşturulması gerekmektedir.



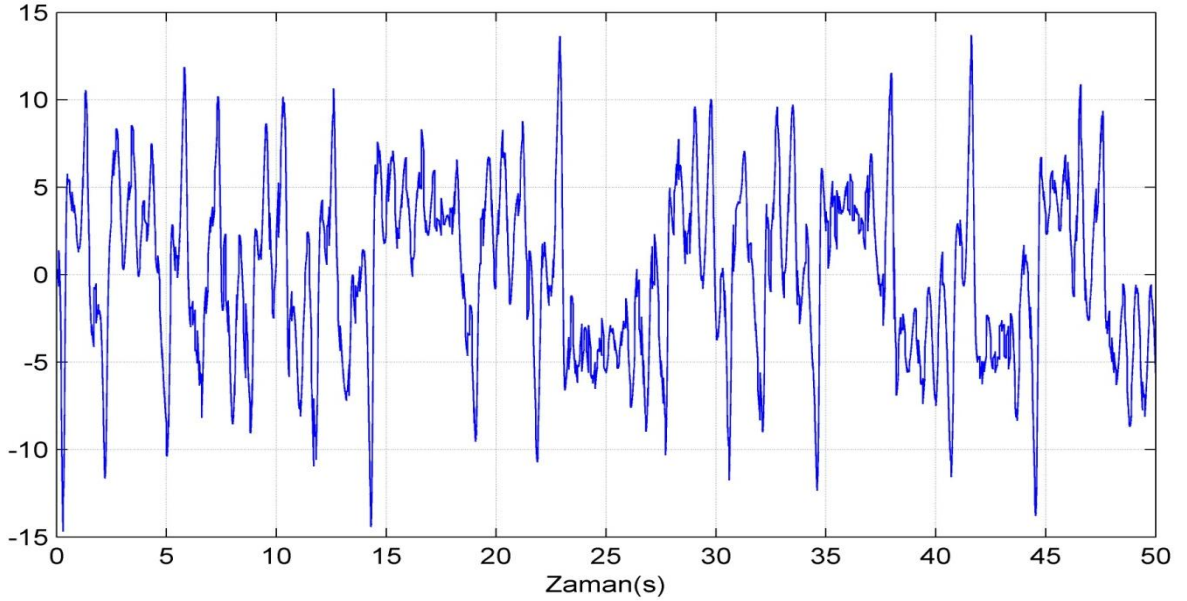
Şekil 5.14. (a) Veri işareti, (b) maskelenmiş veri işareti

Haberleşme Kanalı: Haberleşme kanalı, kaotik işaretlerden birini ve maskelenmiş bilgi işaretini alıcıdan vericiye ileten iki hattan oluşan yapıdır. Haberleşme kanalına gürültü eklenerek daha gerçekçi kanal modeli oluşturulmuştur. Kanal yapısına ait Şekil 5.11’de gösterilen bloğun içyapısı Şekil 5.15’de gösterilmiştir. Bu kanal modeli, kanala gürültü eklememize ve denetleyicinin gürültü performansı görmemizi sağlamaktadır.



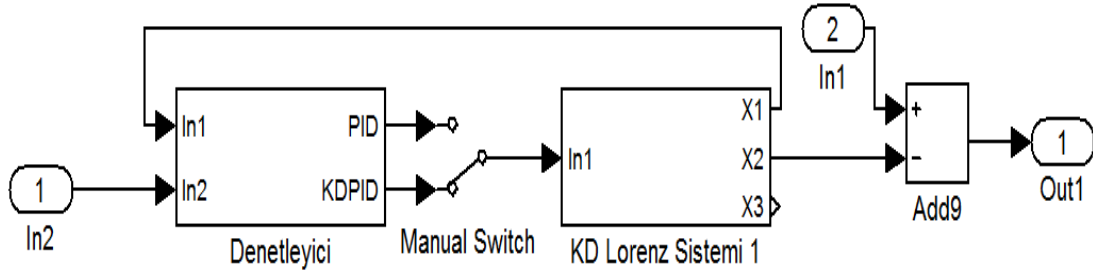
Şekil 5. 15. Kanal modeli

Kanala giren maskelenmiş verinin gürültü eklendikten sonra (alıcının girişinde) oluşan işaretin değişimi Şekil 5.16’da gösterilmiştir.



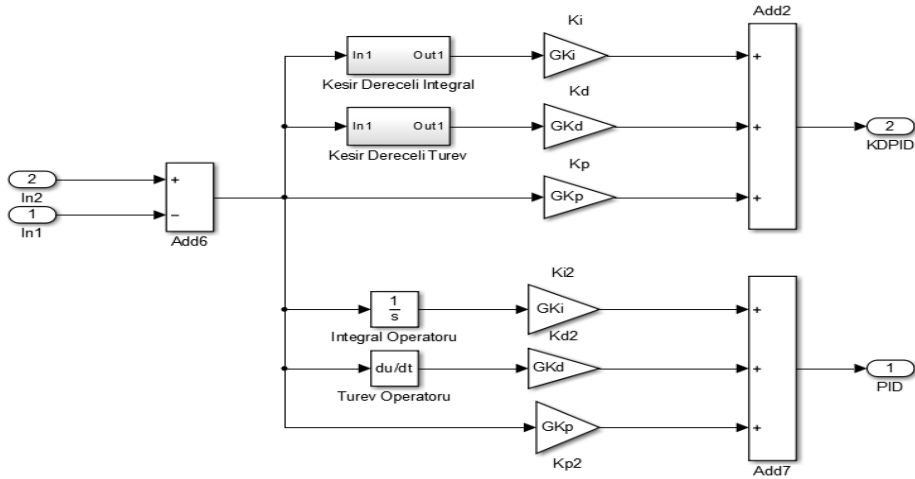
Şekil 5.16. Alıcı girişindeki maskelenmiş gürültülü veri

Maske Kaldırma: Kaotik maskelenmiş verinin maskesinin kaldırılması için alıcı ve vericide bulunan kaotik osilatörler senkronize olması gerekmektedir. Bunun için KDPID ve klasik PID yöntemleri ayrı ayrı uygulanmıştır. Burada amaç, senkronizasyon süresi ve veri kaybı açısından karşılaştırma yapabilmektir. Maske Kaldırma işlemi için kullanılan blok Şekil 5.17’de gösterilmektedir.

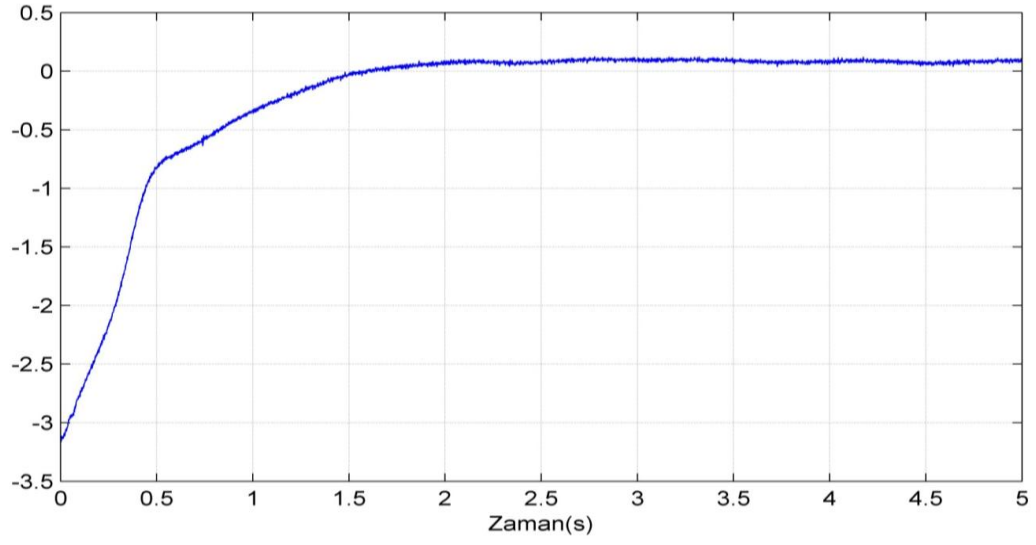


Şekil 5.17. Maske kaldırma için kullanılan Simulink modeli

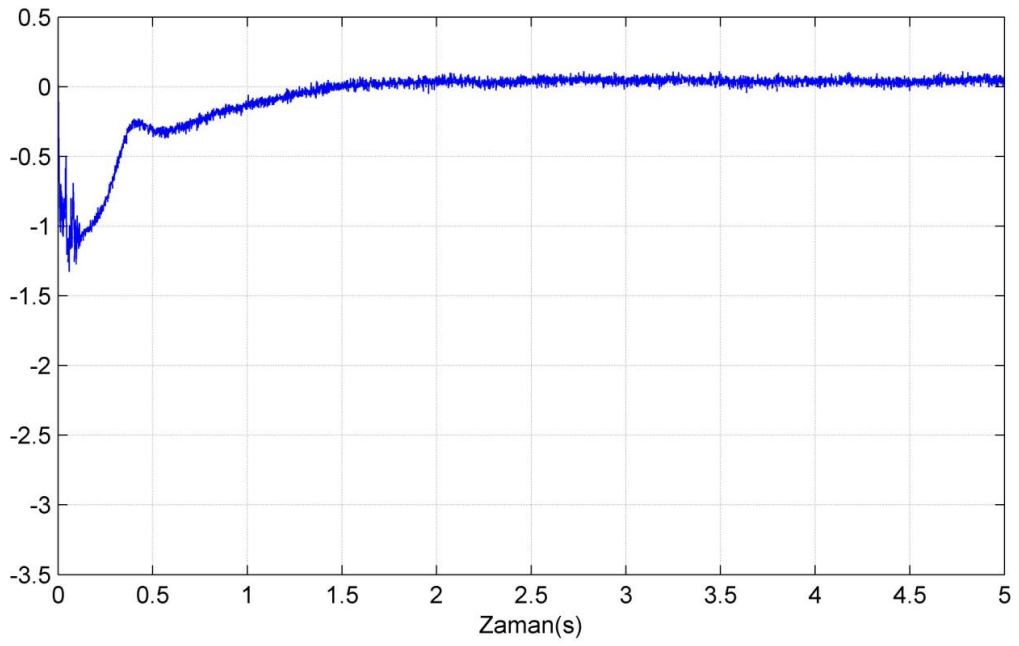
Modelde bulunan denetleyicinin iç yapısı Şekil 5.18’de gösterilmektedir. Model çalıştırıldığında ve PID denetleyicisi kullanıldığında senkronizasyon hatasının değişimi, Şekil 5.19’da gösterilmektedir. Gönderilen ve alınan veri arasındaki farkın zamana göre değişimi Şekil 5.20’de gösterilmektedir.



Şekil 5.18. Denetleyici modeli



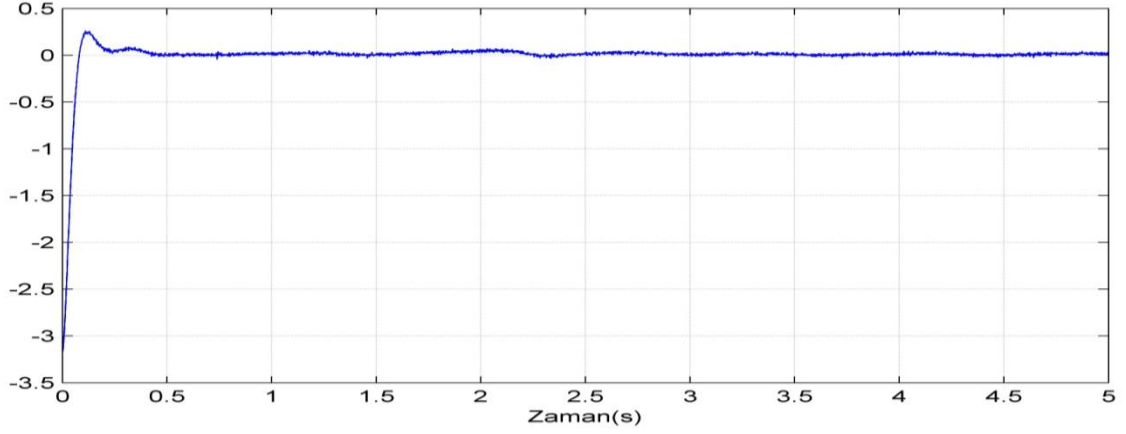
Şekil 5.19. Klasik PID denetleyici kullanılan sistemde oluşan senkronizasyon hatası



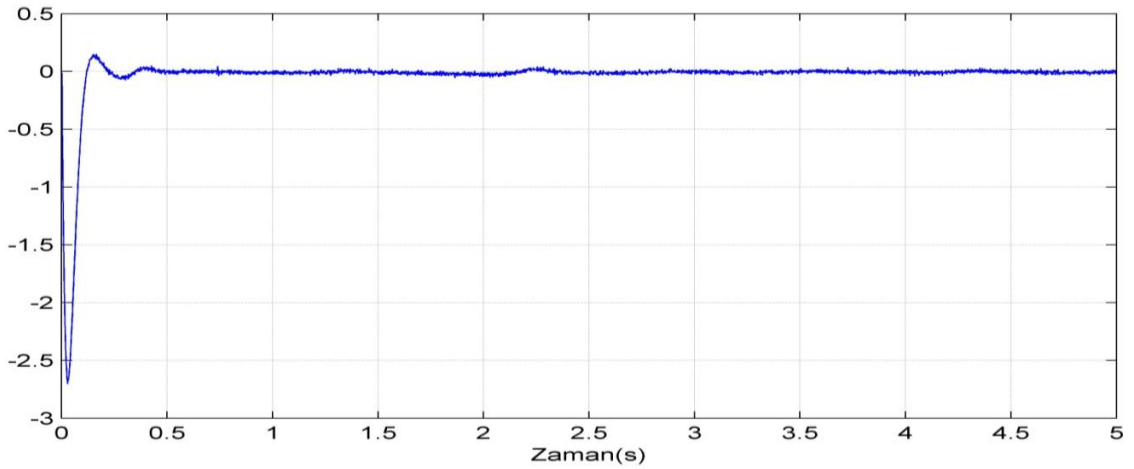
Şekil 5.20. Klasik PID denetleyici kullanılan sistemde oluşan veri hatası

Klasik PID denetleyicisinde Şekil 5.19'dan görüldüğü gibi senkronizasyon hatası yaklaşık 1.5s sonra sıfır olduğu gözlenmektedir. Ayrıca senkronizasyon süresindeki bu artış veri hatası üzerinde de etkisi Şekil 5.20'de görülmektedir. Veri hatası yaklaşık 1.5s sonra sıfır olduğu gözlenmektedir. Yani senkronizasyon hatası aynı zamanda veri hatasını doğrudan etkilediği de açıkça görülmektedir.

KDPID denetleyicisiyle yapılan senkronizasyon denetimli haberleşme sisteminde senkronizasyon hatasının yaklaşık 0.4s'de sıfır olduğu Şekil 5.21'den görülürken, benzer şekilde veri hatasının da yaklaşık 0.4s'de sıfır olduğu Şekil 5.22'de görülmektedir.



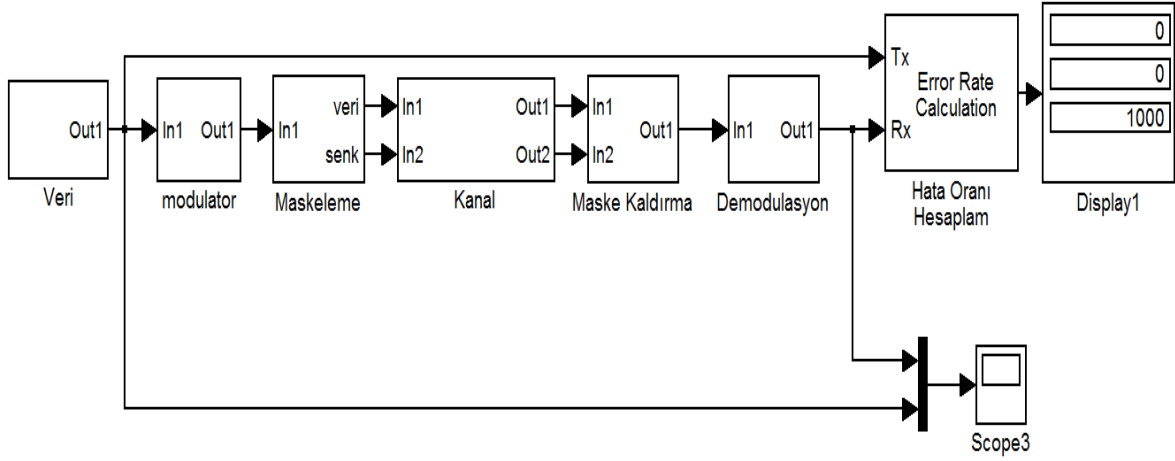
Şekil 5.21. KDPID denetleyici kullanılan sistemde oluşan senkronizasyon hatası



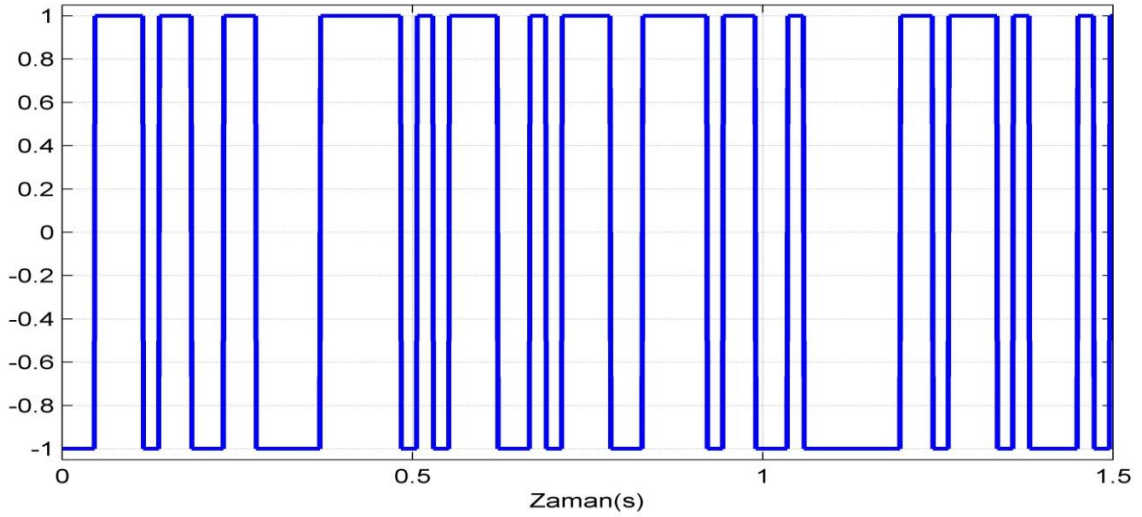
Şekil 5.22. KDPID denetleyici kullanılan sistemde oluşan veri hatası

5.3.2. Kaotik Sayısal Haberleşme

Kaotik sayısal maskeleyme yönteminde PSK ile modüle edilmiş işaret maskelenerek iletim kanalına gönderilmesi prensibine dayanır [41]. Bu prensibe göre oluşturulan Simulink modeli Şekil 5.23’de gösterilmektedir. Model görüldüğü gibi yedi bloktan oluşmuştur. Bunlar; veri, modülatör, maskeleyme, kanal, maske kaldırma, demodülasyon ve hata oranı hesaplama bloklarıdır.



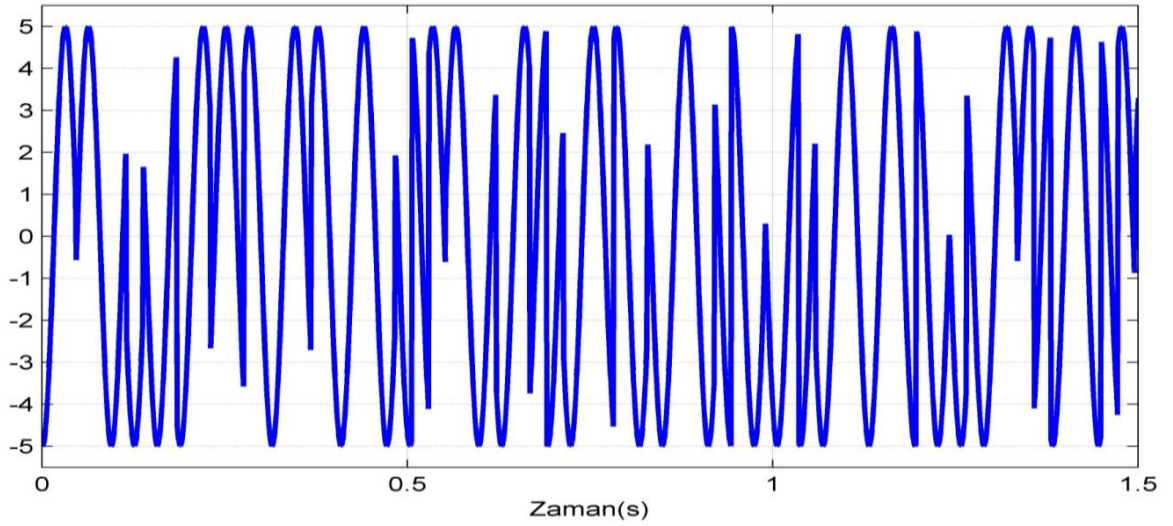
Şekil 5.23. Kaotik maskeli sayısal haberleşme sisteminin Simulink modeli



Şekil 5.24. Bilgi işareti

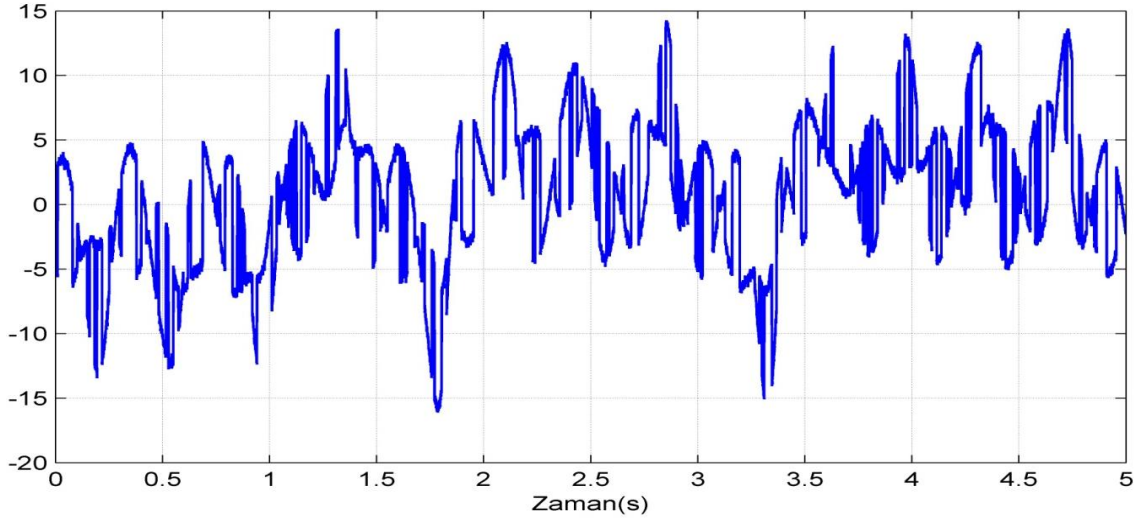
Veri: Sayısal haberleşme sisteminde iletmek istenen sayısal veri 1 ve 0' lardan oluşur. PSK yönteminde gönderilmek istenen verinin 0 olması durumunda -1 sinyali gönderilir. Böylelikle verinin fazı 180° kaydırılmış olur. Blok içinde veri rastgele sinyal üreten bir kaynak ve bir signum fonksiyon alıcıdan oluşmaktadır. PSK ile modüle edilmiş bilgi işareti Şekil 5.24'de gösterilmektedir.

Modülatör: Modülatör bloğunda 1 ve -1 şeklinde gelen veri, sinüzoidal sinyal ile çarpılarak modüle faz kaydırma işlemi elde edilir. PSK ile modüle edilmiş işaret, Şekil 5.25'de gösterilmiştir. Burada verinin her değişiminde taşıyıcı dalganın fazı 180° değiştiği gözlenmektedir.



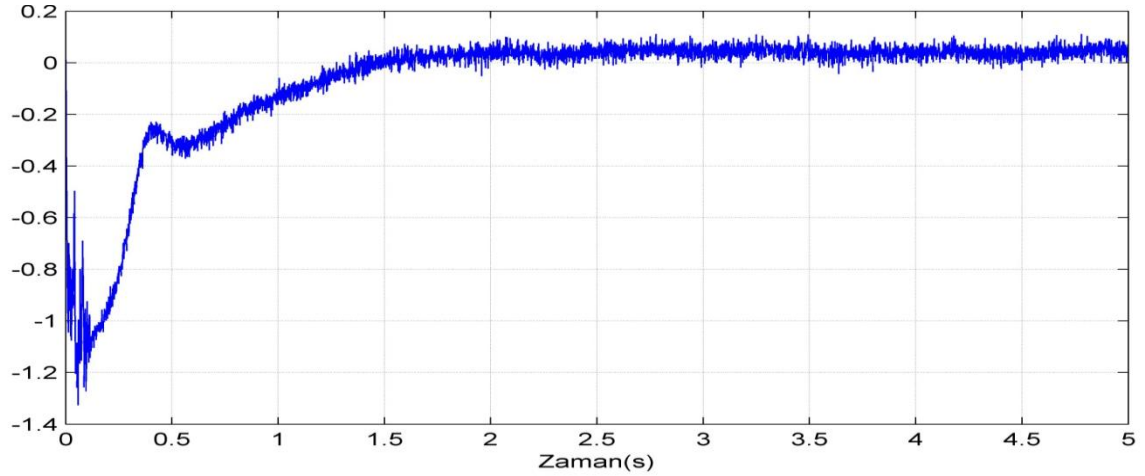
Şekil 5.25. PSK ile modüle edilmiş işaret

Kanal: Haberleşme kanalı iki farklı veriyi taşıyan çift hatlı bir haberleşme kanalı kullanılmıştır. Modelin performansının test edilebilmesi için Gauss gürültü kaynağı eklenmiştir. Kanalın çıkışında gürültülü maskelenmiş veri, Şekil 5.26'da gösterilmiştir.

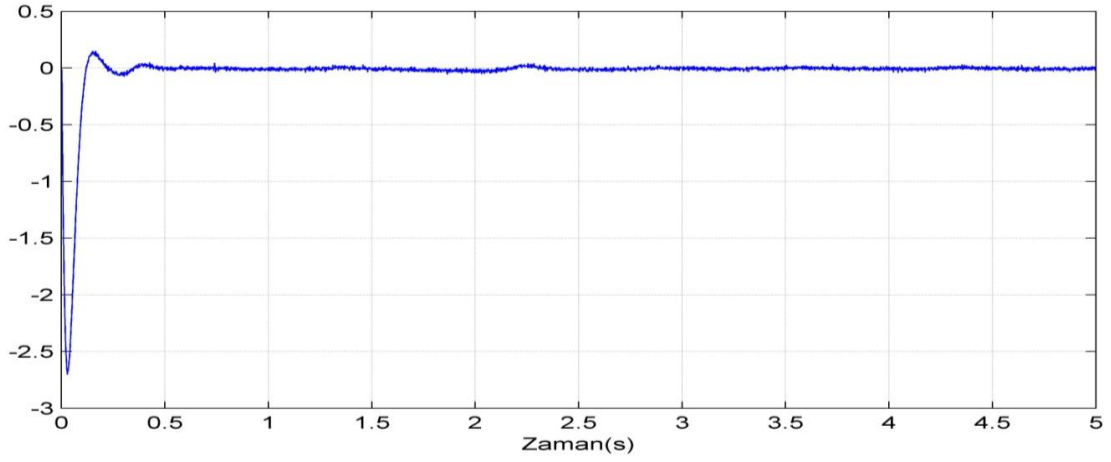


Şekil 5.26. Gürültülü maskelenmiş veri

Maske Kaldırma: Maske kaldırma işleminde analog yöntemdeki gibi KDPID ve klasik PID kullanılarak farklı gürültü seviyelerinde denetleyicinin performansı değerlendirilmiştir. Klasik PID ve KDPID denetleyicilerinin kullanıldığı durumda oluşan senkronizasyon hatası, sırasıyla Şekil 5.27 ve Şekil 5.28’de gösterilmektedir.

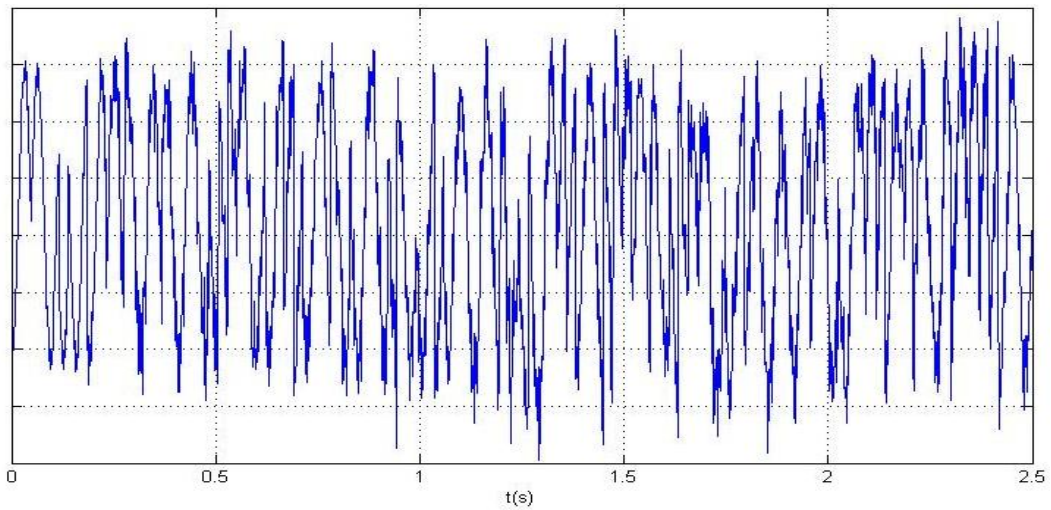


Şekil 5.27. Klasik PID denetleyicisi kullanıldığında oluşan senkronizasyon hatası

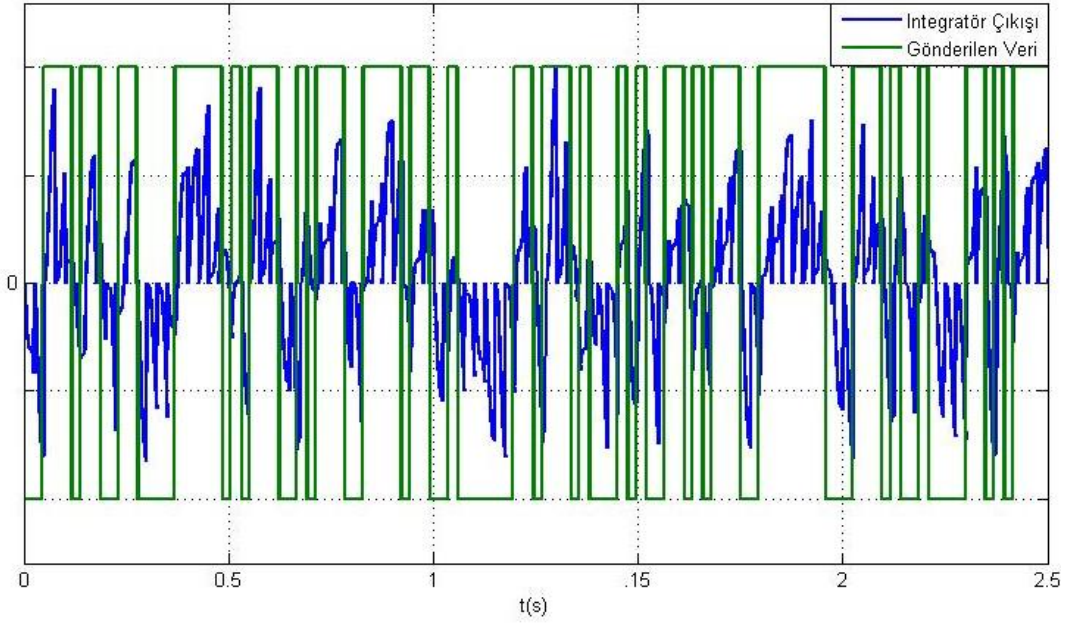


Şekil 5.28. KDPID denetleyicisi kullanıldığında oluşan senkronizasyon hatası

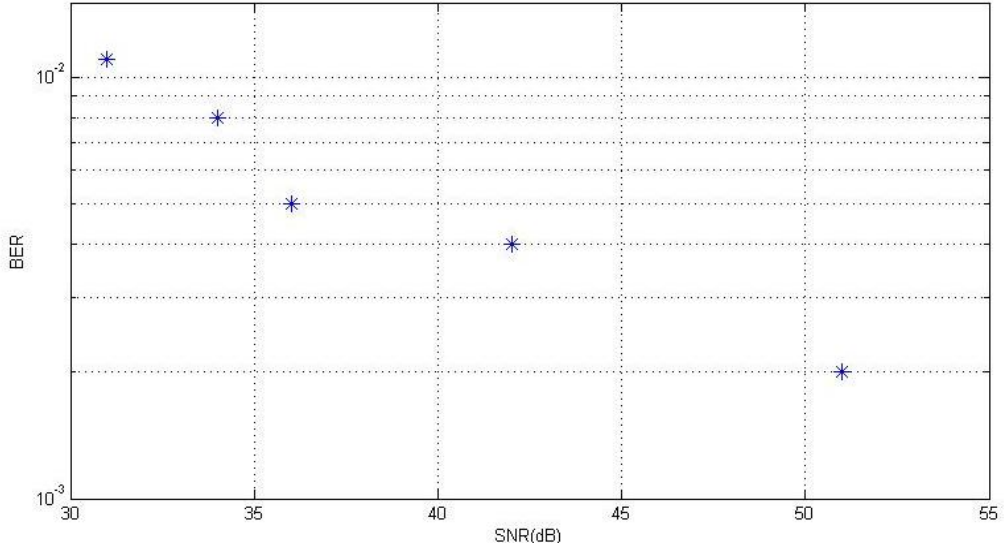
Demodülasyon: Sistemde son olarak, PSK modülasyon yöntemine göre gönderilen veri demodüle edilmiştir. Haberleşme kanalından gelen modüleli ve maskeli veri maskesi kaldırıldıktan sonra Şekil 5.29'da gösterilmektedir. Maskesi kaldırılan veri taşıyıcı ile bir multiplex yardımıyla çarpıldıktan sonra integratöre aktarılmaktadır. Bu integratörün çıkışı ve gönderilen veri aynı eksen üzerinde Şekil 5.30'de gösterilmektedir. Veri demodüle etmek için modelde gürültü performansı için sinyal/gürültü oranına (SNR) göre bit/hata oranı (BER) elde edilmiştir. Bu veriler Tablo 5.1'de gösterilmiştir. Tablo 5.1'den elde edilen verilere göre BER grafiği Şekil 5.31'de gibidir.



Şekil 5.29. Maske kaldırma işleminden sonra PSK taşıyıcısının değişimi



Şekil 5.30. Gönderilen veri ile demodülasyonda kullanılan integratörün çıkışı



Şekil 5.31. BER grafiği

Tablo 5.1. SNR' ye göre BER' in değişimi

SNR	BER(%)
38	1.1
47.5	0.8
63.3	0.5
126	0.4
380	0.2

5.4. Sonuç

Daha önceki bölümde tasarlanan KDPID denetleyicisi ile gerçekleştirilen senkronizasyon denetimi, bu bölümde kaotik haberleşme sistemine uygulanmıştır. Tasarlanan modelin senkronizasyon süresi kısaltması, haberleşme sistemlerindeki veri kaybı için önemli bir göstergedir. Ayrıca kaotik maskeli haberleşme sistemine uygulanan bu modelin gürültü karşısında performansının iyi olduğunu da bu bölümde görmekteyiz. Tasarlanan KDPID ve KD kaotik sistemlerle gerçekleştirilen analog haberleşme ve sayısal haberleşme yöntemlerini, literatürde bulunan benzer çalışmalarla [114, 122-125] karşılaştırdığımızda senkronizasyon süresinden kaynaklanan veri kaybını ve gürültüden kaynaklanan bozulmaları azalttığı açıkça görülmektedir.

6. TARTIŞMA, SONUÇ VE ÖNERİLER

6.1. Elde Edilen Sonuçlar ve Tartışma

Son yıllarda KDS üzerine yapılan araştırmalar artmaktadır. Özellikle KD kaotik devre modellerinin gerçekleştirilmesi ve senkronizasyon kontrolü, KD'ye sahip sistemler ile ilgili çalışmaların başında gelmektedir. Bu tez çalışması, KD yapıya sahip olan KDPID, KD kaotik sistemler ve KDPID denetleyicisinin parametre eniyileme üzerinde yoğunlaşmıştır. Ayrıca KDPID denetleyicilerinin kaotik sistemlerinin senkronizasyonunda literatürdeki çalışmalardan farklı olarak, senkronizasyon süresini kısaltacak bir tasarımın üzerinde durulmuştur. Bu çalışmalardan elde edilen sonuçlar aşağıda sıralanmıştır:

- KD kaos modelleri olan Van der Pol, Sprott, Lorenz osilatör modelleri oluşturularak kaos gerçekleştirilmiş ve bu osilatörlerin oluşturduğu sinyallerin frekans bileşenleri elde edilmiştir. Bu sonuçlara göre elde edilen çıkış işaretleri literatürde bulunan benzer çalışmalarla karşılaştırılmış ve sonuçların örtüştüğü gözlenmiştir. KD'nin değişiminin, Van der Pol osilatörünün sinyal genliğinin değişmesine neden olduğu ve KD'nin 1'e yaklaştırılmasıyla frekans bileşenlerinin değiştiği gözlenmiştir.

- Oluşturulan KD Van der Pol kaotik devre modelinin çıkışından elde edilen sinyallerin, kaotik davranış sergilediği gözlenmiştir. KD'ye sahip bu devre modeli kullanılarak daha farklı yapılar oluşturulabilir. Ancak bu yöntemde istenen değerlere sahip kapasite ve direnç elemanlarının standart dışı olması, devrenin amatör olarak gerçekleştirilmesini zorlaştırır. Bir başka KD kaotik model olan, KD Lorenz'in elektronik devre modeli oluşturulmuştur. Elde edilen sonuçlara bakılarak kaosun tekrar olduğu gözlenmiştir. Ayrıca senkronizasyon kontrolü KDPID ile gerçekleştirilerek başarıyı yüksek olduğu gözlenmektedir.

- Bir başka kaotik yöntem olan Sprott'un simulink modeli ve elektronik devre modeli oluşturulmuş, bu modelin kaotik davranış gösterdiği gözlenmiştir. Bu üç farklı KD'ye sahip kaotik sistemlerden elde edilen sonuçlara göre; Lorenz'in daha yayılı bir frekans spektrumuna sahip olduğu anlaşılmaktadır.

- KD kaotik sistemlerin senkronizasyonu için KDPID denetleyicisi tasarlanmış ve literatürde bulunan klasik PID yöntemi ile karşılaştırılmıştır. Karşılaştırma sonucunda; benzer parametreye sahip klasik PID yönteminin, KDPID denetleyicisine göre, senkronizasyon süresini önemli ölçüde azalttığı açık bir şekilde gözlenmiştir. Senkronizasyon kontrolü üç farklı kaotik yöntem için uygulanmış olup, her biri için ayrı ayrı karşılaştırma yapılmıştır. Elde edilen başarımların, her üç kaotik sistem için de oldukça iyi olduğu gözlenmiştir.

- Senkronizasyon hatasının değişimi, literatürdeki [28, 128-129] benzer çalışmalarla karşılaştırıldığında bu çalışmanın performansının oldukça iyi olduğu görülmektedir.

- KDPID yöntemi, klasik PID yönteminden farklı olarak fazladan iki bilinmeyen parametreye sahiptir. Bunun belirlenmesi için, literatürdeki var olan akıllı sistemler kullanılmaktadır. Bu çalışmada klasik PSO yönteminden farklı olan, FAPSO yöntemi kullanıldı. Bu yöntem daha az iterasyonda eniyi sonucun bulunmasına yardımcı olmaktadır. Böylelikle KDPID denetleyicisinin parametreler ve derecelerini bulmasını sağlayarak verimli bir denetim işlemi gerçekleştirmesini sağlar.

- FAPSO yöntemiyle elde edilen katsayılar göre tasarlanan KDPID denetleyici devresi, yine KD olarak tasarlanan kaos devresinin senkronizasyon denetimindeki başarımının oldukça iyi olduğu görülmektedir.

- Literatürde kaotik haberleşme yöntemlerini K-PSK, COOK, CSK, DCSK, CDSK olarak sayabiliriz. Kaotik senkronizasyon denetleyici temelli olan, K-PSK yöntemi bu çalışmada kullanılmıştır. Kaotik maskeleyme yöntemi senkronizasyon denetimine ihtiyaç duyar ve bu denetleyicide, senkronizasyona ulaşma süresi mümkün olduğunca kısa olması gerekmektedir. Ayrıca denetleyicinin gürültü dayanımı yüksek olması gerekir. Bu nedenle kaotik maskeleyme yönteminde, KDPID tipi bir denetleyicinin kullanımının veri kaybını azaltması açısından oldukça iyi sonuç verdiği gözlenmektedir. Literatürdeki çalışmalarla karşılaştırıldığında [41], senkronizasyon süresinin kısa olmasının başlangıçta yaşanan veri kaybını azalttığı görülmektedir.

Yapılan çalışmayı kısaca özetleyecek olursak; literatürde bulunan kaotik sistemlerin KD olarak tasarlandıklarında da kaotik davranış gösterdiği gözlenmiştir. Bu kaotik sistemlerin devre modelleri oluşturulduğunda tekrar kaotik davranış sergilediği gözlenmiştir. KD kaotik devre modelleri ve KDPID ile senkronizasyonu açısından literatürde yapılan ilk çalışmadır. Klasik denetleyicilerden farklı olarak KDPID denetleyicisi kullanılmıştır. Bu denetleyicinin parametre ve derece gibi bilinmeyen

değerleri, FAPSO yöntemiyle eniyileme edilerek kaos senkronizasyonun denetiminde oldukça iyi bir performans sergilediği görülmektedir. Bu yöntemle gerçekleştirilen çalışmamızın, K-PSK yönteminde veri kaybının az olması açısından literatürde yapılan çalışmalardan [114] daha iyi olduğu açıkça görülmektedir. Elde edilen sonuçların literatürde yapılan bazı çalışmalarla karşılaştırması Tablo 6.1’de gösterilmiştir.

Tablo 6. 1. Tasarlanan denetleyici ile denetlenen senkronizasyon sonucu elde edilen verilerin literatürle karşılaştırılması

	Kaotik Model	Denetim Yöntemi	Senkronizasyon süresi (s)	Alınan Sinyal süresi (s)
Önerilen model	KD Lorenz	KDPID	0.38	0.4
Kaynak [104]**	Sprott	Evrimsel Programlama Temelli PID	~ 2	~ 2
Kaynak [105]**	Gyros	Yumuşak Öğrenmeli PID	~0.8	Verilmemiş
Kaynak [105]	Lorenz	Genişletilmiş Kalman Filtresi *	<0.4 >0.13	0.43
Kaynak [105]	KD Lorenz	Genişletilmiş Kalman Filtresi *	<0.72 >0.4	1.611
* durum tahmini yapılmış **Veriler kaynak [104] and [105]'de bulunan şekillerden yaklaşık olarak alınmıştır.				

6.2. Gelecekte Yapılabilecek Çalışmalar

- KD sistemlerle ilgili yapılan devre modelleme çalışmalarında, standartlara uygun direnç ve kapasite elemanlarıyla gerçekleştirilebilecek devre modelleri elde edilebilir ve bu devreler gerçek sistemlere uygulanabilir.
- KD’ye sahip farklı denetleyici modelleri de yine bu yöntemle gerçekleştirilerek, gerçek zamanlı sistemler üzerindeki performansı değerlendirilebilir.
- Farklı kaotik haberleşme sistemlerinde (CSK COOK DCSK CDSK gibi) KD kaotik devre modelleri kullanılarak, IEEE 802.11 standartlarında bir kaotik haberleşme sistemi gerçekleştirilebilir.
- Parametreleri FAPSO tarafından eniyileme yapılarak gerçek zamanlı farklı bir sistemin (lineer olmayan sistemler veya yüklerin) KDPID veya farklı bir KD denetleyici ile denetimindeki başarısı arttırılabilir.

KAYNAKLAR

- [1] **Oldham, K.B., and Spanier, J.**, 1974. The Fractional Calculus, Academic Press, New York.
- [2] **Miller, K.S., and Ross, B.**, 1993. An Introduction to the Fractional Calculus and Fractional Differential Equations, John Wiley & Sons. Inc., New York.
- [3] **Petras, I.**, 2011. Fractional order nonlinear systems: modeling, analysis and simulation, Springer.
- [4] **Manabe, S.** 1960. The non-integer integral and its application to control systems, *ETJ of Japan*, **6**, 1961, 83-87.
- [5] **Axtell, M. and Bise, M.E.**, 1990. Fractional calculus application in control systems, NAECON 1990 Aerospace and Electronics Conference, Dayton, Ohio, USA May 1990, 563-566.
- [6] **Dorcak, L.**, 1994. Numerical Models for the simulation of the Fractional Order Control Systems, *Slovak Academy of Science Institute of Experimental Physics*, **3**, 8-17.
- [7] **Pudlobny, I.**, 1994. Fractional Order Systems and Fractional Order Controllers, *Slovak Academy of Science Institute of Experimental Physics*, **4**, 28-34.
- [8] **Podlubny, I.**, 1999. Fractional-order systems and $PI^{\lambda}D^{\mu}$ controllers, *IEEE Transactions on Automatic Control*, **44**, 208-214.
- [9] **Manabe, S.**, 1960. The non-integer integral and its application to control systems, *ETJ of Japan*, **6**, 3/4, 1961, 83-87.
- [10] **Hamamci, S.E.**, 2007. An algorithm for stabilization of fractional-order time-delay systems using fractional-order PID controllers, *IEEE Transactions on Automatic Control*, **52**, 1964–1969.
- [11] **Çelik, V., and Demir, Y.**, 2010. Effects on the chaotic system of fractional order PI^{α} controller, *Nonlinear Dynamics*, **59**, 143-159
- [12] **Luo, Y., Chen, YQ., Wang, CY. and Pi, YG.**, 2010. Tuning fractional order proportional integral controllers for fractional order systems, *Journal of Process Control*, **20**, 823–832.

- [13] **Podlubny, I., Petraš, I., Vinagre, B. M., O'leary, P. and Dorčák, L.**, 2002. Analogue realizations of fractional-order controllers. *Nonlinear dynamics*, 29, 281-296.
- [14] **Monje, C.A., Vinagre, B.M., Feliu, V. and Chen, YQ.**, 2008. Tuning and auto-tuning of fractional order controllers for industry applications, *Control Engineering. Practice*, 16, 798–812.
- [15] **Sharma, R., Rana, K.P.S. and Kumar, V.**, 2014. Performance analysis of fractional order fuzzy PID controllers applied to a robotic manipulator, *Expert Systems with Applications*, 41, 4274–4289.
- [16] **Tang Y., Cui M., Hua C., Li L. and Yang Y.**, 2012. Optimum design of fractional order $PI^{\lambda}D^{\mu}$ controller for AVR system using chaotic ant swarm, *Expert Systems with Applications*, 39, 6887-6896.
- [17] **Das S., Pan I., Das S. and Gupta A.**, 2012. Improved model reduction and tuning of fractional-order image controllers for analytical rule extraction with genetic programming, *ISA Transactions*, 51, 237-261.
- [18] **Biswas, A., Das, S., Abraham, A. and Dasgupta, S.**, 2009. Design of fractional-order $PI^{\lambda}D^{\mu}$ controllers with an improved differential evolution. *Engineering Applications of Artificial Intelligence*, 22, 343–350.
- [19] **Karimi, M., Zamani, M., Sadati, N. and Parniani, M.**, 2009. An optimal fractional order controller for an AVR system using particle swarm optimization algorithm, *Control Engineering Practice*, 17, 1380–1387.
- [20] **Valério D. and Costa J.S.**, 2006. Tuning of fractional PID controllers with Ziegler–Nichols type rules, *Signal Processing*, 86, 2771–2784.
- [21] **Matignon, D.**, 1996. Stability result on fractional differential equations with applications to control processing, *IMACS-SMC Proceedings*, Lille, France, July 1996, 963–968.
- [22] **Matignon D.**, 1998. Stability properties for generalized fractional differential systems, *Proceeding of Fractional Differential Systems: Models, Methods and Applications*, 5, 145–158.
- [23] **Lorenz, E.**, 1963. Deterministic nonperiodic flow, *Journal of the Atmospheric Sciences*, 20, 130–141.

- [24] **Barbosa, R.S., Machado, J.A.T., Vinagre, B.M., Calderón, A.J.**, 2007. Analysis of the Van der Pol Oscillator Containing Derivatives of Fractional Order, *Journal of Vibration and Control*, **13**, 1291-1301.
- [25] **Yu, Y., Li, H. X., Wang, S., & Yu, J.**, 2009. Dynamic analysis of a fractional-order Lorenz chaotic system, *Chaos, Solitons & Fractals*, **42**, 1181-1189.
- [26] **Li, C., Chen, G.**, 2004. Chaos and hyperchaos in the fractional-order Rössler equations, *Physica A: Statistical Mechanics and its Applications*, **341**, 55-61.
- [27] **Li, C., Chen, G.**, 2004. Chaos in the fractional order Chen system and its control, *Chaos, Solitons & Fractals*, **22**, 549-554.
- [28] **Chen, D., Zhang, R., Sprott, J. C., Chen, H., Ma, X.** 2012. Synchronization between integer-order chaotic systems and a class of fractional-order chaotic systems via sliding mode control, *Chaos: An Interdisciplinary Journal of Nonlinear Science*, **22**, 023130.
- [29] **Chua, L.O.**, 1984. Nonlinear circuits, *IEEE Transactions on Circuits and Systems*, **31**, 69-87.
- [30] **Sundarapandian, V. and Pehlivan, I.** 2012. Analysis, control, synchronization, and circuit design of a novel chaotic system" ,*Mathematical and Computer Modelling*, **55**, 1904-1915.
- [31] **Morgül, O. and Fekib, M.**, 1999. A chaotic masking scheme by using synchronized chaotic systems, *Physics Letters A*, **251**, 169-176.
- [32] **Pehlivan, I. and Uyaroğlu, Y.**, 2007. Rikitake attractor and it's synchronization application for secure communication systems, *Journal of Applied Sciences*, **7**, 232-236.
- [33] **Cao, H., Zhang, R. and Yan, F.**, 2013. Spread spectrum communication and its circuit implementation using fractional-order chaotic system via a single driving variable, *Communications in Nonlinear Science and Numerical Simulation*, **18**, 341-350.
- [34] **Türk, M., Oğraş, H.** 2011. Classification of chaos-based digital modulation techniques using wavelet neural networks and performance comparison of wavelet families, *Expert Systems with Applications*, **38**, 2557-2565.
- [35] **Yang, T.** 2004. A survey of chaotic secure communication systems. *International Journal of Computational Cognition*, **2**, 81-130

- [36] **Liao, T.L., Huang, N.S.** 1999. An observer-based approach for chaotic synchronization with applications to secure communications, *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, **46**, 1144-1150.
- [37] **Uchida, A., Yoshimori, S., Shinozuka, M., Ogawa, T. and Kannari F.**, 2001. Chaotic on off keying for secure communications, *Optics Letters*, **26**, 866-868.
- [38] **Dedieu, H., Kennedy, M.P. and Hasler, M.**, 1993. Chaos shift keying: modulation and demodulation of a chaotic carrier using self-synchronizing Chua's circuits, *IEEE Transactions on Circuits and Systems-II*, **40**, 637-642.
- [39] **Kolombán, G., Vizvari, B., Schwarz, W. and Abel, A.**, 1996. Differential chaos shift keying: A robust coding for chaos communications, 4th International Specialist Workshop on Nonlinear Dynamics of Electronics Systems (NDES'96), Seville, Spain, June, 87-92.
- [40] **Tam, W.M., Lau, F.C.M. and Tse, C.K.**, 2006. Generalized Correlation-Delay-Shift-Keying Scheme for Noncoherent Chaos-Based Communication Systems, *IEEE Transactions on Circuits and Systems—I*, **53**, 712-722.
- [41] **Chang, W.D.**, 2009. Digital secure communication via chaotic systems, *Digital Signal Processing*, **19**, 693-699.
- [42] **Wang, J.C.**, 1987. Realizations of generalized Warburg impedance with RC ladder networks and transmission lines, *Journal of the Electrochemical Society*, **134**, 1915-1920.
- [43] **Caputo, M.**, 1967. Linear model of dissipation whose Q is almost frequency independent - II, *The Geophysical Journal of the Royal Astronomical Society*, **13**, 529-539.
- [44] **Vinagre, B., Podlubny, I., Hernandez, A. and Feliu, V.**, 2000. Some approximations of fractional order operators used in control theory and applications, *Fractional Calculus and Applied Analysis*, **3**, 231-248.
- [45] **Samko, S.G., Kilbas, A.A., and Marichev, O.I.**, 1993. Fractional Integrals and Derivatives: Theory and Applications, Gordon and Breach, Amsterdam.
- [46] **Valerio, D., and Costa, J.S.**, 2005. Time-Domain implementation of fractional order controllers, *IEEE Proceedings Control Theory and Applications*, **152**, 539-552.

- [47] **Aoun, M., Malti, R., Levron, F., and Oustaloup, A.**, 2003. Numerical simulation of fractional systems, ASME 2003 Design Engineering Technical Conference, Chicago, Illinois, USA, September 2003, 745-752.
- [48] **Carlson, G. and Halijak, C.**, 1964. Approximation of fractional capacitors $(1/s)^{(1/n)}$ by a regular Newton process, *IEEE Transactions on Circuit Theory*, **11**, 210–213.
- [49] **Vinagre, B.M., Chen Y.Q. and Petras I.**, 2003. Two direct Tustin discretization methods for fractional-order differentiator/integrator, *Journal of the Franklin Institute*, **340**, 349–362.
- [50] **Ford, N.J. and Simpson, A.C.**, 2001. The numerical solution of fractional differential equations: speed versus accuracy, *Numerical Algorithms*, **26**, 333-346.
- [51] **Atan, Ö., Turk, M., ve Tuntaş, R.**, 2009. Serbest uyartımlı bir Dc motorun $PI^{\lambda}D^{\mu}$ hız Denetimi ve kesir dereceli denetleyicilerin performans analizi, *Sakarya Üniversitesi Fen Bilimleri Dergisi*, **13**, 34-41.
- [52] **Wajdi M.A. and Sprott J.C.**, 2003. Chaos in fractional-order autonomous nonlinear systems, *Chaos, Solitons and Fractals*, **16**, 339–351.
- [53] **Chen, D., Liu, C., Wu, C., Liu, Y., Ma, X. and You, Y.**, 2012. A new fractional-order chaotic system and its synchronization with circuit simulation, *Circuits Systems and Signal Processing*, **31**, 1599–1613.
- [54] **Dorcak, L.** 2012. Comparison of the electronic realization of the fractional-order system and its model, 13th International Carpathian Control Conference (ICCC), High Tatras, Slovakia, May 2012, 119-124.
- [55] **Xiang-Rong, C., Chong-Xin, L. and Qiang W. F.**, 2008. Circuit realization of the fractional-order unified chaotic system, *Chinese Physics B*, **17**, 1664-1669.
- [56] **Zhang, B. Pi, Y. and Luo, Y.**, 2012. Fractional order sliding-mode control based on parameters auto-tuning for velocity control of permanent magnet synchronous motor, *ISA Transactions*, **51**, 649-656.
- [57] **Efe, M.Ö.**, 2008. Fractional Fuzzy Adaptive Sliding-Mode Control of a 2-DOF Direct-Drive Robot Arm. *IEEE Transactions on Systems, Man, and Cybernetics, PartB: Cybernetics*, **38**, 1561–1570.

- [58] **Podlubny, I., Petráš, I., Vinagre, B.M., O’leary, P. and Dorcák, L.**, 2002. Analogue Realizations of Fractional-Order Controllers, *Nonlinear Dynamics*, **29**, 281–296.
- [59] **Ramiro, S., Barbosa, J.A., Machado, T. and Jesus, I.S.**, 2008. On the fractional PID control of a laboratory servo system, *Proceedings of the 17th World Congress the International Federation of Automatic Control*, Seoul, Korea, July, 2008, 15273-15278.
- [60] **Petráš, I. and Vinagre, M. B.**, 2002. Practical application of digital fractional-order controller to temperature control, *Acta Montanistica Slovaca Ročník*, **7**, 131-137.
- [61] **Hamamci, S.E.**, 2008. Stabilization using fractional-order PI and PID controllers, *Nonlinear Dynamics*, **51**, 329–343.
- [62] **Zamani, M., Ghartemani, M.K., Sadati, N. and Parniani, M.**, 2009. Design of a fractional order PID controller for an AVR using particle swarm optimization, *Control Engineering Practice*, **17**, 1380–1387.
- [63] **Lee, C.H. and Chang, F.K.**, 2010. Fractional-order PID controller optimization via improved electromagnetism-like algorithm, *Expert Systems with Applications*, **37**, 8871–8878.
- [64] **Luoa, Y., Chenc, Y.Q., Wangc, C.Y. and Pi, Y.G.**, 2010. Tuning fractional order proportional integral controllers for fractional order systems, *Journal of Process Control*, **20**, 823–831.
- [65] **Hwanga, C. and Cheng, Y.C.**, 2006. A numerical algorithm for stability testing of fractional delay systems, *Automatica*, **42**, 825–831.
- [66] **Cao, J.Y., Liang, J. and Cao, B.G.** 2005. Optimization of fractional order PID controllers based on genetic algorithms, *In Proceedings of The Fourth International Conference on Machine Learning and Cybernetics*, Guangzhou, China, August 2005, 5686–5689.
- [67] **Cao, J.Y. and Cao, B.G.**, 2006. Design of fractional order controller based on particle swarm optimization, *International Journal of Control, Automation, and Systems*, **4**, 775–781.
- [68] **Stavroulakis, P.**, 2006. Chaos Applications in Telecommunications, Taylor and Francis, New York.

- [69] **Sprott, J.C.**, 2003. *Chaos and Time-Series Analysis*, Oxford University Press, Oxford.
- [70] **Tavazoei, M.S. and Haeri, M.**, 2008, Synchronization of chaotic fractional-order systems via active sliding mode controller, *Physica A: Statistical Mechanics and its Applications*, **387**, 57-70.
- [71] **Hegazi, A.S. Ahmed, E. and Matouk, A.E.**, 2013. On chaos control and synchronization of the commensurate fractional order Liu system, *Communications in Nonlinear Science and Numerical Simulation*, **18**, 1193-1202.
- [72] **Matouk, A.E.**, 2011. Chaos, feedback control and synchronization of a fractional-order modified Autonomous Van der Pol–Duffing circuit, *Communications in Nonlinear Science and Numerical Simulation*, **16**, 975-986.
- [73] **Wang, S., Yu, Y., We, G.**, 2014. Hybrid projective synchronization of time-delayed fractional order chaotic systems, *Nonlinear Analysis: Hybrid Systems*, **11**, 129-138.
- [74] **Petráš, I.**, 2008. A note on the fractional-order Chua’s system, *Chaos, Solitons and Fractals*, **38**, 140–147.
- [75] **Petráš, I.**, 2009. Chaos in the fractional-order Volta’s system: modeling and simulation, *Nonlinear Dynamics*, **57**, 157–170.
- [76] **Li, C. and Chen, G.**, 2004. Chaos and hyperchaos in the fractional-order Rössler equations, *Physica A*, **341**, 55–61.
- [77] **Deng, W.H. and Li, C.P.**, 2005. Chaos synchronization of the fractional Lü system, *Physica A*, **353**, 61–72.
- [78] **Lu, J.G. and Chen, G.**, 2006. A note on the fractional-order Chen system, *Chaos Solitons and Fractals*, **27**, 685–688.
- [79] **Ge, Z.M. and Ou, C.Y.**, 2007. Chaos in a fractional order modified Duffing system, *Chaos Solitons and Fractals*, **34**, 262–291.
- [80] **Ge, Z.M. and Hsu, M.Y.**, 2007. Chaos in a generalized van der Pol system and in its fractional order system, *Chaos Solitons and Fractals*, **33**, 1711–1745.
- [81] **Chen, J. H. and Chen, W.C.**, 2008. Chaotic dynamics of the fractionally damped van der Polequation, *Chaos Solitons and Fractals*, **35**, 188–198.
- [82] **Atan, Ö., Turk, M. ve Tuntaş, R.**, 2012. Kesir dereceli denetleyici parametrelerinin bulanık adaptif parçacık sürü optimizasyon yöntemi ile

belirlenmesi, *Otomatik Kontrol Ulusal Toplantısı, TOK-2012*, 11-13 Ekim 2012, Niğde, s.445-450.

- [83] **Xu, F. and Yu, P.**, 2010. Chaos control and chaos synchronization for multi-scroll chaotic attractors generated using hyperbolic functions, *Journal of Mathematical Analysis and Applications*, **362**, 252–274.
- [84] **Pecora, L.M. and Carroll, T.L.**, 1990. Synchronization in Chaotic Systems, *Physical Review Letters*, **64**, 821-823
- [85] **Pecora, L.M. and Carroll, T.L.**, 1991. Driving systems with chaotic signals, *Physical Review A*, **44**, 2374-2383
- [86] **Li, G.H.**, 2007. Generalized projective synchronization between Lorenz system and Chens system, *Chaos Solitons and Fractals*, **32**, 1454-1458.
- [87] **Kocarev, L. and Parlitz, U.**, 1996. Generalized Synchronization, Predictability, and Equivalence of Unidirectionally Coupled Dynamical Systems, *Physical Review Letters*, **76**, 1816-1819.
- [88] **Yang, J.Z. and Hu, G.**, 2007. Three types of generalized synchronization, *Physics Letters A*, **361**, 332-335.
- [89] **Pourkargar, B.D. and Shahrokhi, M.**, 2011. Optimal fuzzy synchronization of generalized lorenz chaotic systems, *The Journal of Mathematics and Computer Science*, **2**, 27-36.
- [90] **Yu, H. and Liu, Y.**, 2003. Chaotic synchronization based on stability criterion of linear systems, *Physics Letters A*, **314**, 292–298.
- [91] **Rosenblum, M.G., Pikovsky, A.S. and Kurths, J.**, 1997. From Phase to Lag Synchronization in Coupled Chaotic Oscillators, *Physical Review Letters*, **78**, 4193-4196.
- [92] **Park, E.H., Zaks, M.A. and Kurths, J.**, 1999. Phase synchronization in the forced Lorenz system, *Physical Review E*, **60**, 6627-6638.
- [93] **Liu, W.**, 2006. Antiphase synchronization in coupled chaotic oscillators, *Physical Review E*, **73**, 057203_1-057203_4.
- [94] **Kennedy, M.P. and Chua, L.**, 1986. Van der pol and chaos, *IEEE Transactions on Circuits and Systems*, **33**, 974-980.
- [95] **Barbosa, S.R., Machado, J.A.T, Vinagre, B.M. and Calderon, A.J.**, 2007. Analysis of the Van der Pol oscillator containing derivatives of fractional order, *Journal of Vibration and Control*, **13**, 1291-1301.

- [96] **Li, C.G. and Chen, G.R.**, 2004. Chaos in the fractional order Chen system and its control, *Chaos, Solitons and Fractals*, **22**, 549-554
- [97] **Grigorenko, I. and Grigorenko, E.**, 2003. Chaotic dynamics of the fractional Lorenz system, *Physical Review Letters*, 91 034101
- [98] **Lu, J.G.**, 2006. Chaotic dynamics of the fractional-order Lü system and its synchronization, *Physics Letters A*, **354**, 305-311.
- [99] **Wang, F.Q. and Liu, C.X.**, 2006. Hyperchaos evolved from the Liu chaotic system, *Chinese Physics*, **15**, 963-968.
- [100] **Wu, X.J., Li, J. and Chen, G.R.**, 2008. Chaos in the fractional order unified system and its synchronization, *Journal of the Franklin Institute*, **345**, 392-401.
- [101] **Chen, X.R., Liu, C.X. and Wang, F.Q.**, 2008. Circuit realization of the fractional-order unified chaotic system, *Chinese Physics B*, 17, 1664-1669.
- [102] **Kennedy, J. and Eberhart, R.**, 1995. Particle swarm optimization, IEEE International Conference on Neural Networks, Washington, DC, USA, Aralık 1995, 1942–1948.
- [103] **Cao, J.Y. and Cao, B.G.**, 2006. Design of fractional order controller based on particle swarm optimization, *International Journal of Control, Automation, and Systems*, **4**, 775-781.
- [104] **Zhan, Z.H., Zhang, J., Li, Y. and Chung, H.S.H.**, 2009. Adaptive particle swarm optimization, *IEEE Transactions on Systems Man and Cybernetics Part B*, **39**, 1362-1381.
- [105] **Saber, A.Y., Senjyu, T., Yona, A. and Funabashi, T.**, 2007. Unit commitment computation by fuzzy adaptive particle swarm optimization, *IET Generation, Transmission & Distribution*, **1**, 456–465.
- [106] **Zhang, W. and Liu, Y.**, 2008. Multi-objective reactive power and voltage control based on fuzzy optimization strategy and fuzzy adaptive particle swarm, *International Journal of Electrical Power & Energy Systems*, **30**, 525–532.
- [107] **Niknam, T., Mojarad, H.D. and Meymand, H.Z.**, 2011. Non-smooth economic dispatch computation by fuzzy and self adaptive particle swarm optimization, *Applied Soft Computing*, **11**, 2805-2817.
- [108] **Alataş, B., Akın, E. and Özer, A.B.**, 2009. Chaos Embedded Particle Swarm Optimization Algorithms, *Chaos, Solitons & Fractals*, **40**, 1715-1734.

- [109] **Wang, Y., Li, B., Weise, T., Wang, J., Yuan, B. and Tian, Q.,** 2011. Self-adaptive learning based particle swarm optimization, *Information Sciences*, **181**, 4515-4538.
- [110] **Juang, Y.T., Tung, S.L. and Chiu, H.C.,** 2011. Adaptive fuzzy particle swarm optimization for global optimization of multimodal functions, *Information Sciences*, **181**, 4539-4549.
- [111] **Ross T. J.** 2004. Fuzzy Logic with Engineering Applications, Wiley, UK.
- [112] **Harris J.,** 2006. Fuzzy Logic Applications in Engineering Science, Springer, Netherlands.
- [113] **Solihin, M.I., Tack, L.F., Kean, M.L.,** 2011. Tuning of PID Controller Using Particle Swarm Optimization (PSO), *International Journal on Advanced Science, Engineering and Information Technology*, **1**, 458-461.
- [114] **Tam, W.M., Lau, F.C.M. and Tse, C.K.,** 2006. Digital Communication with Chaos, Elsevier, UK.
- [115] **Alvarez, G., Montoya, F., Pastor, G. and Romera, M.,** 1999. Chaotic cryptosystems, *International Journal of Bifurcation and Chaos*, **9**, 332–338.
- [116] **Elmirghani, J.M.H. and Cryan, R.A.,** 1994. New chaotic based communication technique with multiuser provision, *Electronics Letters*, **30**, 1206–1207.
- [117] **Miller, S.H., Elmirghani, J.M.H. and Cryan, R.A.,** 1995. Efficient chaotic-driven echo path modelling, *Electronics Letters*, **31**, 429–430.
- [118] **Leung, H. and Lam, J.,** 1997. Design of demodulator for the chaotic modulation communication system, *IEEE Transactions on Circuits and Systems Part I*, **44**, 262–267.
- [119] **Chow, T.W.S., Feng, J.C. and Ng, K.T.,** 2000. An adaptive demodulator for the chaotic modulation communication system with RBF neural network, *IEEE Transactions on Circuits and Systems Part I*, **47**, 902–909.
- [120] **Feng, J.C. and Tse C.K.,** 2001, An on-line adaptive chaotic demodulator based on radial-basis-function neural Networks, *Physical Review E*, **63**, 026202-1-10.
- [121] **Chiou, J.S., Tsai, S.H. and Liu, M.T.,** 2012. A PSO-based adaptive fuzzy PID-controllers, *Simulation Modelling Practice and Theory*, **26**, 49-59.
- [122] **Chen, HC., Chang, JF., Yan, JJ. and Liao TL.,** 2008. EP-based PID control design for chaotic synchronization with application in secure communication, *Expert Systems with Applications*, **34**, 1169-1177.

- [123] **Kiani-B, A., Fallahi, K., Pariz, N. and Leung, H.**, 2009. A chaotic secure communication scheme using fractional chaotic systems based on an extended fractional kalman filter, *Communications in Nonlinear Science and Numerical Simulation*, **14**, 863-879.
- [124] **Türk, M. and Oğraş, H.**, 2010. Recognition of multi-scroll chaotic attractors using wavelet-based neural network and performance comparison of wavelet families, *Expert Systems with Applications*, **37**, 8667–8672.
- [125] **Zaher, A.A. and Abu-Rezq, A.**, 2011. On the design of chaos-based secure communication systems, *Communications in Nonlinear Science and Numerical Simulation*, **16**, 3721-3737.
- [126] **Cuomo, K. and Oppenheim, A.**, 1993. Circuit implementation of synchronized chaos with applications to communications, *Physical Review Letters*, **71**, 65–68.
- [127] **Short, K.**, 1994. Steps toward unmasking secure communications, *International Journal of Bifurcation and Chaos*, **4**, 959–977.
- [128] **Chen, D., Zhang, R., Ma, X. and Liu, S.**, 2012. Chaotic synchronization and anti-synchronization for a novel class of multiple chaotic systems via a sliding mode control scheme, *Nonlinear Dynamics*, **69**, 35-55.
- [129] **Chen, D., Liu, C., Wu, C., Liu, Y., Ma, X. and You, Y.**, 2012. A new fractional-order chaotic system and its synchronization with circuit simulation, *Circuits, Systems, and Signal Processing*, **31**, 1599-1613.

ÖZGEÇMİŞ

1983- yılında Van'da dünyaya gelen Özkan ATAN, ilk ve orta eğitimini Van'da tamamladı. Lise eğitimini yine Van'da Endüstri Meslek Lisesinde dereceyle bitirdi. 2000 yılında Fırat Üniversitesi Teknik Eğitim Fakültesi Elektrik Öğretmenliği bölümüne girip 2004 yılında buradan mezun oldu. Mezun olduktan sonra bir yıl süreyle Van TEDAŞ'da çalıştı. 2005 yılında Yüzüncü Yıl Üniversitesi Elektrik Elektronik Mühendisliğinde yüksek lisans eğitimine başlayıp, 2008 yılında "Fırçasız DC motorların modellenmesi ve PWM ile hız kontrolü" adlı tez çalışmasıyla yüksek lisans öğrenimini tamamladı. Aynı yıl Yüzüncü Yıl Üniversitesi Erciş Meslek Yüksek Okulunda öğretim görevlisi olarak çalışmaya başladı ve 2008 yılının Güz döneminde Fırat Üniversitesi Elektrik Elektronik Mühendisliğinde Doktora öğrenimine başladı. Halen Yüzüncü Yıl Üniversitesinde çalışmaya devam etmektedir.