

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**YENİ BİR KAOTİK KARIŞTIRICI
TASARIMI VE UYGULAMASI**

YÜKSEK LİSANS TEZİ

Ersin KURT

Enstitü Anabilim Dalı : MEKATRONİK MÜHENDİSLİĞİ

Tez Danışmanı : Doç. Dr. İhsan PEHLİVAN

Aralık 2017

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ


YENİ BİR KAOTİK KARIŞTIRICI
TASARIMI VE UYGULAMASI

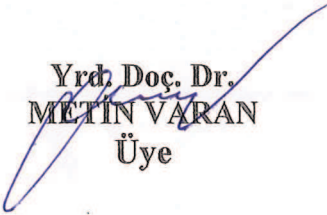
YÜKSEK LİSANS TEZİ

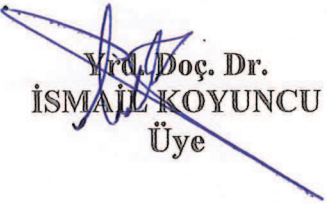
Ersin KURT

Enstitü Anabilim Dalı : MEKATRONİK MÜHENDİSLİĞİ

Bu tez 27.12.2017 tarihinde aşağıdaki jüri tarafından oybirliği / ~~oyçokluğu~~ ile kabul edilmiştir.


Doç. Dr.
İHSAN PEHLİVAN
Jüri Başkanı


Yrd. Doç. Dr.
METİN VARAN
Üye


Yrd. Doç. Dr.
İSMAIL KOYUNCU
Üye

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Ersin KURT

27.12.2017



TEŐEKKÜR

Yüksek lisans eğitimin boyunca değerli bilgi ve deneyimlerinden yararlandığım, her konuda bilgi ve desteğini almaktan çekinmediğim, araştırmanın planlanmasından yazılmasına kadar tüm aşamalarında yardımlarını esirgemeyen, teşvik eden, aynı titizlikte beni yönlendiren değerli danışman hocam Doç. Dr. İhsan PEHLİVAN'a teşekkürlerimi sunarım.

Laboratuvar olanakları konusunda anlayış ve yardımlarını esirgemeyen Sakarya Üniversitesi Mekatronik Mühendisliği Bölüm Başkanı Prof. Dr. Osman ELDOĞAN'a ve bilgi ve deneyimlerinden yararlandığım sayın hocam Doç. Dr. İhsan PEHLİVAN'a teşekkür ederim.

Ayrıca bu çalışmanın maddi açıdan desteklenmesine olanak sağlayan Sakarya Üniversitesi Bilimsel Araştırma Projeleri (BAP) Komisyon Başkanlığına (Proje No: 2017-50-01-024) teşekkür ederim.

İÇİNDEKİLER

TEŞEKKÜR	i
İÇİNDEKİLER	ii
SİMGELER VE KISALTMALAR LİSTESİ	v
ŞEKİLLER LİSTESİ	vi
TABLolar LİSTESİ	x
ÖZET	xii
SUMMARY	xiii

BÖLÜM 1.

GİRİŞ	1
1.1. Karıştırıcılar	1
1.1.1. Karıştırmaya etki eden faktörler	2
1.1.1.1. Karıştırma sıcaklığı	2
1.1.1.2. Karıştırma süresi	3
1.1.1.3. Karıştırıcı tipi	4
1.1.1.4. Devir sayısı	4
1.1.2. Karıştırıcı çeşitleri	5
1.1.2.1. Türbin tip karıştırıcı	5
1.1.2.2. Pervaneli karıştırıcı	6
1.1.2.3. Uçkur tip karıştırıcı	7
1.1.2.4. Yoğurma makinası	7
1.1.2.5. Kuru karıştırıcı	8
1.1.2.6. Döner karıştırıcı	8
1.2. Kaos Bilimi ve Kaotik Sistemler	9
1.3. Delta Robot	10

BÖLÜM 2.	
LİTERATÜR ÖZETİ	11
BÖLÜM 3.	
MATERYAL VE YÖNTEM	14
3.1. DC Motorlar	14
3.1.1. Yön	14
3.1.2. Hız	14
3.1.3. Gerilim	14
3.1.4. Akım	15
3.1.5. Güç	15
3.1.6. Tork	15
3.2. Servo Motorlar	15
3.3. Arduino	16
3.3.1. Arduino uno r3' ün kısımları	17
3.3.2. Arduino uno teknik özellikleri	18
3.4. PWM ile DC Motor Kontrol Devresi	18
3.5. Ma871 Dijital Brix Refraktometre	19
BÖLÜM 4.	
KAOTİK DELTA ROBOTUN TASARLANMASI.....	20
4.1. Yazılımın Hazırlanması	21
4.2. Delta Robot MATLAB Programı	24
4.3. Yörüngesel Dağılım Oranının Hesaplanması	30
BÖLÜM 5.	
ARAŞTIRMA BULGULARI	32
5.1. Kaotik Karıştırma Deneyleri	32
5.2. Seçilen Kaotik Sistemler	38
5.2.1. Sprott A sistemi	38
5.2.2. Pehlivan-Wei sistemi	38
5.2.3. Aizawa sistemi	38

5.2.4. Guckenheimer-Holmes sistemi	39
5.2.5. Lu-Chen sistemi	39
5.2.6. Hadley sistemi	39
5.2.7. Halvorsen sistemi	40
5.3. MATLAB Karşılaştırmaları	40
5.3.1. Sprott A sistemi MATLAB karşılaştırmaları	41
5.3.2. Pehlivan-Wei sistemi MATLAB karşılaştırmaları	44
5.3.3. Aizawa sistemi MATLAB karşılaştırmaları	46
5.3.4. Guckenheimer-Holmes sistemi MATLAB karşılaştırmaları	49
5.3.5. Lu-Chen sistemi MATLAB karşılaştırmaları	51
5.3.6. Hadley sistemi MATLAB karşılaştırmaları	54
5.3.7. Halvorsen sistemi MATLAB karşılaştırmaları	56
5.4. Karıştırma Tabloları	59
BÖLÜM 6.	
SONUÇLAR VE ÖNERİLER	66
KAYNAKLAR	68
ÖZ GEÇMİŞ	72

SİMGELER VE KISALTMALAR LİSTESİ

3D	: 3 Boyutlu
AC	: Alternatif akım
BRIX	: Sulu çözeltiler içindeki % çözünür kuru madde miktarı
DC	: Doğru akım
DATA	: Servo motorun veri ucu
HO	: Homojenlik oranı
ICSP	: Seri port programlayıcı
LDR	: Lineer delta robot
PWM	: Sinyal genişlik modülasyonu
PCDW	: Önerilen küp şeklindeki çalışma alanı
Pos_X-Y-Z	: Delta robot 3 boyutlu koordinat sistemi
RAD	: Robot açı değerleri
RC	: Uzaktan kumanda
RK45	: 4. ve 5. Dereceden runga kutta
RX	: Veri alma
TX	: Veri iletme
USB	: Evrensel seri veriyolu
X-Y-Z	: 3 Boyutlu koordinat sistemi
YDO	: Yörüngesel dağılım oranı

ŞEKİLLER LİSTESİ

Şekil 1.1. İki farklı toz kümesi	1
Şekil 1.2. İdeal karışmış toz kümesi	1
Şekil 1.3. Rastgele (random) karışmış toz kümesi	2
Şekil 1.4. Isıtıcı (caket) reaktör	3
Şekil 1.5. Sıcaklık kontrol cihazı	3
Şekil 1.6. Karıştırma süresinin saat veya kronometre ile kontrolü	4
Şekil 1.7. Zaman rölesi	4
Şekil 1.8. Türbin tip karıştırıcının reaktör içindeki konumu	5
Şekil 1.9. Kanatçık tipleri	6
Şekil 1.10. Sıvı merkezkaç kuvvetinin etkisi ve sıvı hareket katmanları	6
Şekil 1.11. Akım kırıcı kanatçıklar	7
Şekil 1.12. Uçkur tip karıştırıcı	7
Şekil 1.13. Yoğurma makinesi	8
Şekil 1.14. Kuru karıştırıcı	8
Şekil 1.15. Döner karıştırıcı	9
Şekil 1.16. Delta robot	10
Şekil 3.1. Servo motor	15
Şekil 3.2. PWM sinyalleri	16
Şekil 3.3. Arduino Uno R3' ün kısımları	17
Şekil 3.4. PWM ile DC motor kontrol devresi	19
Şekil 3.5. Ma871 dijital brix refaktometre	19
Şekil 4.1. Delta robotun tasarım aşamaları	20
Şekil 4.2. Delta robotun devre şeması	21
Şekil 4.3. Tasarlanan delta robot	21
Şekil 4.4. Delta robot yazılımı blok diyagramı	22

Şekil 4.5. Delta robotun her bir bölgedeki çalışma aralıkları	23
Şekil 4.6. Delta robotun hareket ettiği bölgeler	23
Şekil 5.1. Brix ölçerin saf su ile kalibrasyonunun yapılması	33
Şekil 5.2. Programda Dairesel hareketin seçilmesi	33
Şekil 5.3. Delta robotun çalıştırılması.....	34
Şekil 5.4. Dairesel hareketin 120. saniyedeki geçtiği bölgeler ve yüzdeleri	34
Şekil 5.5. (a) Dairesel hareket x-y-z grafiği (b) Delta robot dairese hareket servo x-y-z grafiği	35
Şekil 5.6. (a) 3 Boyutlu dairese hareket (b) 3 Boyutlu delta robot dairese hareket	36
Şekil 5.7. (a) Dairesel hareket x-y faz portresi (b) Delta robot dairese hareket servo x-y faz portresi	36
Şekil 5.8. (a) Dairesel hareket x-z faz portresi (b) Delta robot dairese hareket servo x-z faz portresi	36
Şekil 5.9. (a) Dairesel hareket y-z faz portresi (b) Delta robot dairese hareket servo y-z faz portresi	37
Şekil 5.10. Dairesel hareketile 120 saniye karıştırılan çözeltinin şeker oranının brix ölçer ile ölçülmesi	37
Şekil 5.11. (a) Sprott-A sistemi x-y-z grafiği (b) Delta robot Sprott-A sistemi servo x-y-z grafiği	41
Şekil 5.12. (a) 3 Boyutlu Sprott-A sistemi (b) 3 Boyutlu delta robot Sprott-A sistemi	41
Şekil 5.13. (a) Sprott-A sistemi x-y faz portresi (b) Delta robot Sprott-A sistemi servo x-y faz portresi	42
Şekil 5.14. (a) Sprott-A sistemi x-z faz portresi (b) Delta robot Sprott-A sistemi servo x-z faz portresi	42
Şekil 5.15. (a) Sprott-A sistemi y-z faz portresi (b) Delta robot Sprott-A sistemi servo y-z faz portresi	43
Şekil 5.16. (a) Pehlivan-Wei sistemi x-y-z grafiği (b) Delta robot Pehlivan-Wei sistemi servo x-y-z grafiği	43
Şekil 5.17. (a) 3 Boyutlu Pehlivan-Wei sistemi (b) 3 Boyutlu delta robot Pehlivan-Wei sistemi	44

Şekil 5.18. (a) Pehlivan-Wei sistemi x-y faz portresi (b) Delta robot Pehlivan-Wei sistemi servo x-y faz portresi	44
Şekil 5.19. (a) Pehlivan-Wei sistemi x-z faz portresi (b) Delta robot Pehlivan-Wei sistemi servo x-z faz portresi	45
Şekil 5.20. (a) Pehlivan-Wei sistemi y-z faz portresi (b) Delta robot Pehlivan-Wei sistemi servo y-z faz portresi	45
Şekil 5.21. (a) Aizawa sistemi x-y-z grafiği (b) Delta robot Aizawa sistemi servo x-y-z grafiği	46
Şekil 5.22. (a) 3 Boyutlu Aizawa sistemi (b) 3 Boyutlu delta robot Aizawa sistemi	46
Şekil 5.23. (a) Aizawa sistemi x-y faz portresi (b) Delta robot Aizawa sistemi servo x-y faz portresi	47
Şekil 5.24. (a) Aizawa sistemi x-z faz portresi (b) Delta robot Aizawa sistemi servo x-z faz portresi	47
Şekil 5.25. (a) Aizawa sistemi y-z faz portresi (b) Delta robot Aizawa sistemi servo y-z faz portresi	48
Şekil 5.26. (a) Guckenheimer-Holmes sistemi x-y-z grafiği (b) Delta robot Guckenheimer-Holmes sistemi servo x-y-z grafiği	48
Şekil 5.27. (a) 3 Boyutlu Guckenheimer-Holmes sistemi (b) 3 Boyutlu delta robot Guckenheimer-Holmes sistemi	49
Şekil 5.28. (a) Guckenheimer-Holmes sistemi x-y faz portresi (b) Delta robot Guckenheimer-Holmes sistemi servo x-y faz portresi	49
Şekil 5.29. (a) Guckenheimer-Holmes sistemi x-z faz portresi (b) Delta robot Guckenheimer-Holmes sistemi servo x-z faz portresi	50
Şekil 5.30. (a) Guckenheimer-Holmes sistemi y-z faz portresi (b) Delta robot Guckenheimer-Holmes sistemi servo y-z faz portresi	50
Şekil 5.31. (a) Lu-Chen sistemi x-y-z grafiği (b) Delta robot Lu-Chen sistemi servo x-y-z grafiği	51
Şekil 5.32. (a) 3 Boyutlu Lu-Chen sistemi (b) 3 Boyutlu delta robot Lu-Chen sistemi	51
Şekil 5.33. (a) Lu-Chen sistemi x-y faz portresi (b) Delta robot Lu-Chen sistemi servo x-y faz portresi	52

Şekil 5.34. (a) Lu-Chen sistemi x-z faz portresi(b) Delta robot Lu-Chen sistemi servo x-z faz portresi	52
Şekil 5.35. (a) Lu-Chen sistemi y-z faz portresi (b) Delta robot Lu-Chen sistemi servo y-z faz portresi	53
Şekil 5.36. (a) Hadley sistemi x-y-z grafiği (b) Delta robot Hadley sistemi servo x-y-z grafiği	53
Şekil 5.37. (a) 3 Boyutlu Hadley sistemi (b) 3 Boyutlu delta robot Hadley sistemi	54
Şekil 5.38. (a) Hadley sistemi x-y faz portresi (b) Delta robot Hadley sistemi servo x-y faz portresi	54
Şekil 5.39. (a) Hadley sistemi x-z faz portresi (b) Delta robot Hadley sistemi servo x-z faz portresi	55
Şekil 5.40. (a) Hadley sistemi y-z faz portresi (b) Delta robot Hadley sistemi servo y-z faz portresi	55
Şekil 5.41. (a) Halvorsen sistemi x-y-z grafiği (b) Delta robot Halvorsen sistemi servo x-y-z grafiği	56
Şekil 5.42. (a) 3 Boyutlu Halvorsen sistemi (b) 3 Boyutlu delta robot Halvorsen sistemi	56
Şekil 5.43. (a) Halvorsen sistemi x-y faz portresi (b) Delta robot Halvorsen sistemi servo x-y faz portresi	57
Şekil 5.44. (a) Halvorsen sistemi x-z faz portresi (b) Delta robot Halvorsen sistemi servo x-z faz portresi	57
Şekil 5.45. (a) Halvorsen sistemi y-z faz portresi (b) Delta robot Halvorsen sistemi servo y-z faz portresi	58

TABLolar LİSTESİ

Tablo 5.1. Sabit karıştırıcının şekerli su ölçüm verileri	59
Tablo 5.2. Dairesel hareketli karıştırıcının şekerli su yörünge verileri	59
Tablo 5.3. Dairesel hareketli karıştırıcının şekerli su ölçüm verileri	60
Tablo 5.4. Sprot A sistemli tabanlı kaotik karıştırıcının şekerli su yörünge verileri	60
Tablo 5.5. Sprot A sistemli tabanlı kaotik karıştırıcının şekerli su ölçüm verileri	60
Tablo 5.6. Pehlivan-Wei sistemli tabanlı kaotik karıştırıcının şekerli su yörünge verileri	61
Tablo 5.7. Pehlivan-Wei sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri	61
Tablo 5.8. Aizawa sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri	61
Tablo 5.9. Aizawa sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri	62
Tablo 5.10. Guckenheimer-Holmes sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri	62
Tablo 5.11. Guckenheimer-Holmes sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri	62
Tablo 5.12. Lu-Chen sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri	63
Tablo 5.13. Lu-Chen sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri	63
Tablo 5.14. Hadley sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri	63

Tablo 5.15. Hadley sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri	64
Tablo 5.16. Halvorsen sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri	64
Tablo 5.17. Halvorsen sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri	64
Tablo 5.18. Halvorsen sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri	65
Tablo 5.19. Halvorsen sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri	65

ÖZET

Anahtar kelimeler: Delta robot, kaotik sistem, arduino, kaotik karıştırıcı

Bu çalışmada Arduino ve MATLAB tabanlı yeni bir Kaotik karıştırıcı tasarlanarak katı-sıvı karışımı üzerinde farklı kaotik sistemler ile uygulamaları gerçekleştirilmiştir. Bu kapsamda öncelikle, literatür taraması yapılarak kaotik karıştırma işlemi için farklı kaotik denklem sistemlerinin araştırılması, birbirleriyle karşılaştırılması, uygun olanların tespit edilmesi işlemleri yapılmıştır.

Daha sonra tüm tasarımları özgün olan, 3 Eksenli, 3 Servo Motorlu, Motor Sürücü kartlı, Arduinio Uno işlemcisi içeren, 30x30x80 cm boyutlarında, 2 kg ağırlığında olan standart Delta robotun benzeri bir robot geliştirilmiştir. Sonraki aşamada, MATLAB ortamında seçilen kaotik sistemler RK45 sayısal çözüm algoritması ile çözdürülerek, integer formatındaki sonuçlar USB kablosu aracılığı Arduino Uno işlemcisi üzerine aktarılmıştır.

Tezde geliştirilen yazılım sayesinde; Arduinio Uno işlemcisi, MATLAB'tan alınan kaotik verileri konum bilgisi olarak değerlendirmekte, tasarlanan robotun X-Y-Z eksenlerinde, gezecek şekilde kontrol edilmesi sağlanmaktadır.

Son aşamada, katı-sıvı karışım tipi ve farklı kaotik sistemlerin kullanıldığı, Yörüngesel Dağılım Oranı (YDO), Homojenlik Oranı (HO) ve karıştırma süreleri parametrelerinin ölçüldüğü deneysel çalışmalar yapılmıştır. En son olarak tezdeki tüm simülasyon ve deneysel sonuçlar tablolar halinde birbirleri ile karşılaştırılmıştır. Tablolardan görüldüğü gibi verilen YDO ve HO parametrelerine göre seçilen kaotik sistemler sabit ve dairesel hareketli karıştırıcılara göre daha yüksek performans elde etmiştir. İçlerinde en iyi performansa sahip sistemin ise Sprott A sistemi olduğu görülmüştür.

Ayrıca gerçekleştirilen bu kaotik karıştırıcı ile sıvı-sıvı ve katı-katı maddelerin karıştırılmasına ilişkin bir dizi deneysel çalışma da yapılabilmektedir.

Tezde sunulan bu prototipin, daha kısa sürede daha homojen karışımların elde edilebilmesi için yeni kaotik karıştırıcı sistemlerin ve algoritmalarının geliştirilebilmesi amacına hizmet edeceği düşünülmektedir.

A NEW CHAOTIC MIXER DESIGN AND APPLICATION

SUMMARY

Keywords: Delta robot, chaotic system, arduino, chaotic mixer

In this study, a new chaotic mixer based on Arduino and MATLAB was designed and its applications were implemented with different chaotic systems on solid-liquid mixture. In this context, firstly literature search was carried out for the different chaotic equations systems for chaotic mixing and compared them with each other to determine the appropriate ones.

Then, a robot similar to the standard Delta robot with 3 axes, 3 servo motors, a motor driver card, an Arduino Uno processor, 30x30x80 cm size, 2 kg weight, which is unique in all designs, has been developed. In the next step, the selected chaotic systems in the MATLAB environment were solved with the RK45 numerical solution algorithm, and the results in the integer format are transferred via the USB cable to the Arduino Uno processor.

Thanks to the software developed in the thesis; The Arduino Uno processor evaluates the chaotic data received from MATLAB as position information and controls the designed robot in XYZ axes.

In the last stage, experimental studies have been carried out in which the solid-liquid mixture type and different chaotic systems are used, the Orbital Dispersion Ratio (ODR), the Homogeneity Ratio (HR) and the mixing time parameters are measured. Finally, all simulations and experimental results in the thesis are compared with each other in tabular form. The chaotic systems selected according to the ODR and HR parameters as given in the tables have achieved higher performance than the fixed and circular moving mixers and it was seen that the system with the best performance is Sprott A system.

In addition, a series of experimental studies on the mixing of liquid-liquid and solid-solid materials with this chaotic mixer can be performed.

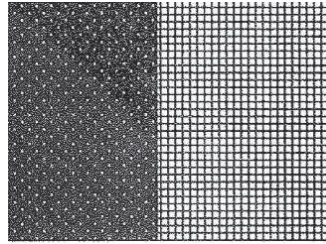
This prototype presented in the thesis is supposed to serve the purpose of developing new chaotic mixer systems and algorithms so that more homogenous mixtures can be obtained in a shorter time.

BÖLÜM 1. GİRİŞ

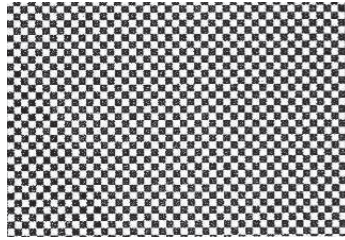
1.1. Karıştırıcılar

Farklı en az iki maddenin birleştirilmesine karıştırma, bu olayı gerçekleştiren cihazlara da karıştırıcı (mikser) denir. Karıştırılması istenilen maddeler katı, sıvı, gaz veya bunların türevleri şeklindedir. Karıştırmada en çok üzerinde durulan mevzular karıştırma işleminin süresi ve homojenliğidir [1]. Karışımın homojenliği ne kadar yüksek ve karıştırıcının çalışma süresi ne kadar az ise o karıştırıcının performansı o kadar yüksektir. Peki ideal bir karışım nasıldır?

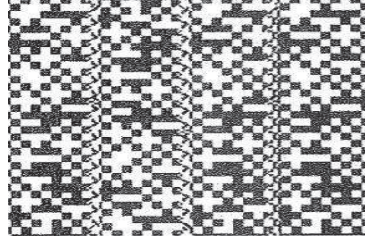
Bunu bir örnekle açıklamaya çalışalım: Elimizde aynı miktar ve büyüklükte iki farklı toz kümesinin olduğu düşünelim. Bu toz kümeleri karıştırılmadan önce Şekil 1.1.'de görüldüğü gibidir.



Şekil 1.1. İki farklı toz kümesi [2]



Şekil 1.2. İdeal karışmış toz kümesi [2]



Şekil 1.3. Rastgele (random) toz kümesi [2]

Şekil 1.2.'deki karışımda, her bir toz taneciği diğer toz taneciğine dokunur. Fakat bu karışım gerçekte elde edilemez. Gerçekte elde edebileceğimiz Şekil 1.3.'te de görüldüğü gibi rastgele (random) karışımdır. Bu karışımın elde edilmesinde toz taneciklerinin şekli ve sayısı da karıştırma işi için önemli bir etkidir. Karıştırmanın performansını belirlerken, karışacak maddelerin birbirleri arasındaki eş dağılım derecesi, temel örnekler göz önüne alınıp karşılaştırıldığı zaman kolayca açıklanabilmektedir [2].

Son yıllarda hemen hemen her yerde karıştırıcıları görmekteyiz. Evlerimizde mutfaklarda, fırınlarda hamur yoğurma makinelerinde, inşaatlarda, fabrikalarda ve daha bir çok alanda karıştırıcıları kullanmaktayız. Dolayısıyla bu kadar çok kullanılan bir makinenin geliştirilmesi ve daha verimli olarak tasarlanıp kullanılması imalat amaçlı veya üretim safhasında bu makineleri kullanan firmalar için yüksek önem arz etmektedir [3, 4].

1.1.1. Karıştırmaya etki eden faktörler

1.1.1.1. Karıştırma sıcaklığı

Sıcaklık ile maddelerin çözünürlüğü doğru orantılıdır. Diğer bir deyişle sıcaklık arttıkça maddeler daha hızlı karışır. Bu sebeple yüksek viskozitesi olan sıvıların viskozitelerini aşağı indirmek için Şekil 1.4.'te görüldüğü gibi caket ile ısıtılabilir. Böylece karışım süresi kısalmış olup güçten de tasarruf edilmektedir.

Burada dikkat edilmesi gereken en önemli husus verilecek ısının maddelerin ve karıştırma neticesinde oluşacak yeni maddenin yapısını bozmayacak bir sıcaklıkta olması gerekmektedir. Maddelerin arasındaki kimyasal reaksiyonlarla da ısı açığa çıkabilmektedir. Bu yüzden Şekil 1.5.'deki gibi bir cihaz ile sıcaklık kontrol edilmelidir.



Şekil 1.4. Isıtıcı (caket) reaktör [6]



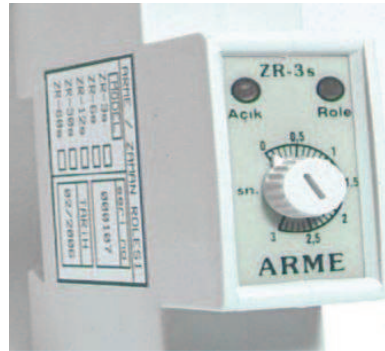
Şekil 1.5. Sıcaklık kontrol cihazı [6]

1.1.1.2. Karıştırma süresi

Yukarıda da belirtildiği gibi karıştırmada en kritik öneme sahip konulardan birisi kuşkusuz karışımın süresidir. Karıştırıcının karışımı istenilen homojenlik düzeyinde ne kadar sürede karıştırabildiğini belirlemek için bir takım testler yapılması gerekmektedir. Küçük işletmelerde ve fabrikalarda Şekil 1.6.'daki gibi kronometre veya Şekil 1.7.'deki gibi zaman röleleri kullanılmaktadır.



Şekil 1.6. Karıştırma süresinin saat veya kronometre ile kontrolü [6]



Şekil 1.7. Zaman rölesi [6]

1.1.1.3. Karıştırıcı tipi

Karıştırıcı tipi seçilirken belli başlı bir kuralı olmamaktadır. Daha çok tecrübeden yararlanılmaktadır. En kolay ve homojen yapılabilen karışımlar sıvı-sıvı karışımlarıdır. Katı ve gaz olan karışımları karıştırmak zor olduğu gibi sarf edilen güç yüksek olmasına rağmen homojenlik oranları düşük olmaktadır. Bu sebeple eğer karışım maddelerimiz sıvı değil de katı ve gaz ise bunları sıvılaştırmak işimizi kolaylaştırmaktadır.

1.1.1.4. Devir sayısı

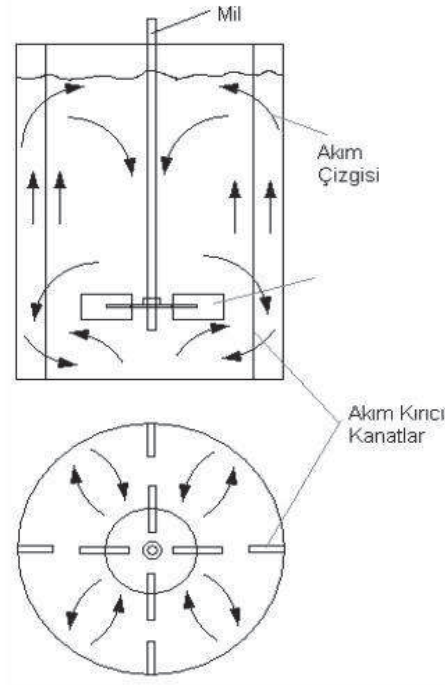
Devir sayısı, karıştırıcının 1 dakika süresince kendi etrafındaki dönüş sayısıdır. İsteddiğimiz yoğunluktaki ve sürede karışımı elde edebilmemiz için karıştırıcının belirli bir devir sayısında dönmesi gerekir. Düşük devir sayısı karışımın istenilen sürede ve homojenlikte olmamasının yanında fazladan güç harcayarak maliyet

kayıplarına neden olmaktadır. Yüksek devir sayısında ise karışımın kabın dışına savrulmasına veya taşmasına sebep olur. Bu yüzden karıştırıcıyı karışım için en uygun devir sayısında çalıştırılması gerekmektedir. Son yıllarda gelişen teknoloji sayesinde artık karıştırıcıların devir sayısı kontrol edilebilmektedir. Bu faktörler de göz önünde bulundurularak ve karıştırılacak maddelerin özellikleri de dikkate alınarak en uygun karıştırıcı tipi tespit edilir.

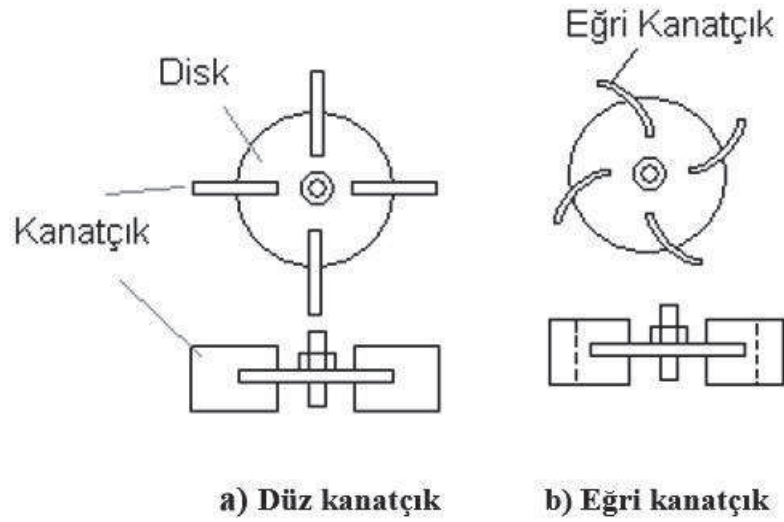
1.1.2. Karıştırıcı çeşitleri

1.1.2.1. Türbin tip karıştırıcı

Genel olarak bu karıştırıcılar sıvı-sıvı karışımlarında kullanılmaktadır. Şekil 1.8.'deki gibi motorun ucundaki mile bağlı plakaya dikey kanatlar eklenmesiyle oluşan karıştırıcı tipidir. Şekil 1.9.'da düz ve eğri kanatçıklı tipleri gösterilmektedir.



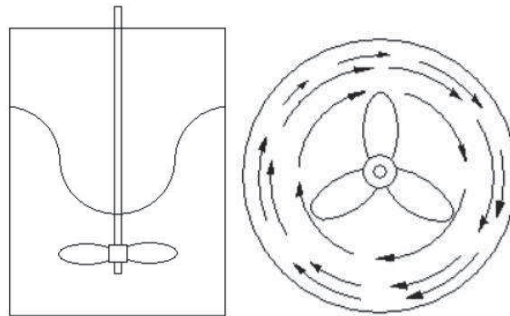
Şekil 1.8. Türbin tip karıştırıcının reaktör içindeki konumu [6]



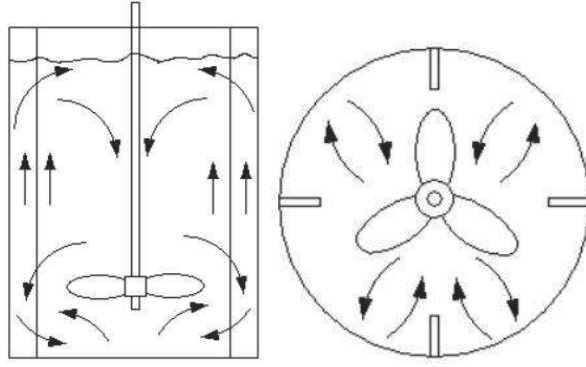
Şekil 1.9. Kanatçık tipleri [6]

1.1.2.2. Pervaneli karıştırıcı

Özellikle gemilerde kullanılan pervaneler, Şekil 1.10.'da görüldüğü gibi döndüğünde oluşan merkez kaç kuvveti ile sıvı kenarlara doğru taşmaktadır. Bunu önlemek için Şekil 1.11.'de gösterilen akım kırıcı kanatçıklar kabın kenarlarına yerleştirmiştir. Kanatçıklar sayesinde sıvı katmanlar yerine türbülanslı olarak hareket etmektedir.



Şekil 1.10. Sıvı merkezkaç kuvvetinin etkisi ve sıvı hareket katmanları [6]



Şekil 1.11. Akım kırıcı kanatçıklar [6]

1.1.2.3. Uçkur tip karıştırıcı

Uçkur tip karıştırıcıların devir sayıları diğerlerine göre daha düşük olduğundan karıştırma süresi de daha uzundur. Şekil 1.12.'de gösterilen bu karıştırıcıyı daha çok yavaş karıştırılması istenilen sıvılarda kullanılır.



Şekil 1.12. Uçkur tip karıştırıcı [6]

1.1.2.4. Yoğurma makinesi

Çok yoğun viskoziteli maddelerin karıştırılmasında yoğurma makineleri sıklıkla kullanılır. Şekil 1.13.'de de görüldüğü gibi üstü açık silindirik kaplı bir karıştırıcıdır.



Şekil 1.13. Yoğurma makinesi [6]

1.1.2.5. Kuru karıştırıcı

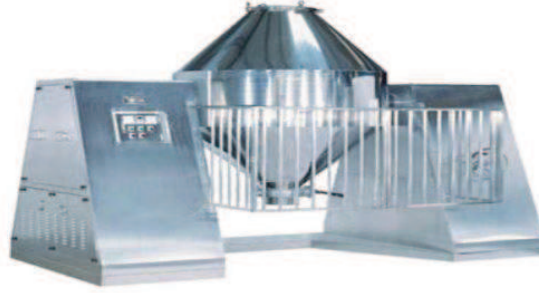
Genel olarak tozların karıştırılmasında kullanılmaktadır. Şekil 1.14.'de gösterildi gibi içerisinde iki spiral vardır ve farklı yönlerde dönmektedir. Böylelikle maddelerin farklı yönlerde hareket etmesi sağlanarak daha iyi bir karıştırma işi yapılmış olur.



Şekil 1.14. Kuru Karıştırıcı [6]

1.1.2.6. Döner karıştırıcı

Tozların karıştırılmasında kullanılan bir başka karıştırıcı tipi de döner karıştırıcıdır. Şekil 1.15.'de gösterilen döner karıştırıcının içinde kepçe şekilli veya oluk biçiminde kanatçıklar bulunmaktadır. Döner karıştırıcının dışında ise tabanları birbirine bağlı iki koniden oluştuğu görülmektedir [5, 6].



Şekil 1.15. Döner Karıştırıcı [6]

1.2. Kaos Bilimi ve Kaotik Sistemler

Kaos, en kısa tanımla, düzensizliğin düzeni şeklinde tanımlanan, doğrusal olmayan olayları açıklamaya yarayan bir bilim dalıdır [7]. Rastgele gibi görünmesine rağmen dikkatli incelendiğinde bu yapıyı rastgelelikten ayıran birçok önemli özelliği vardır. Başlangıç şartlarına hassas bağımlılığı, zaman boyutundaki düzensiz yapısı, birbirinden farklı ve sınırsız periyodik salınımları barındırması, geniş bir güç spektrumu olması, parçalı (fraktal) bir limit kümesinin olması, sınırlı bir alanda frekansı ve genliği tespit edilemeyecek kadar değişken işaretler içermesi önemli özelliklerinden birkaçıdır. Bu sebeplerle dinamik sistemlerde bilinen en karmaşık kararlı hal yapısı “kaos” dur.

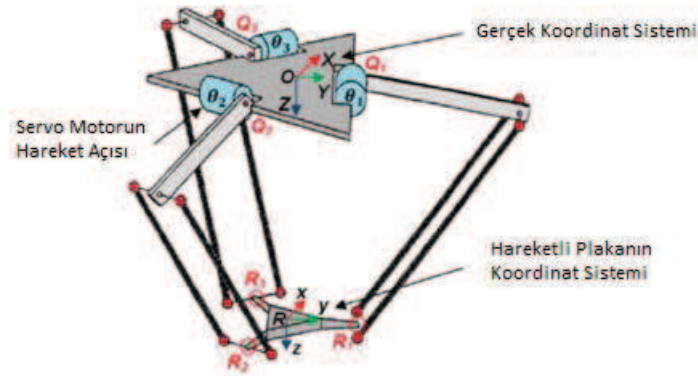
Kaos bilimi, 1963 yılında Edward Lorenz’in öncülüğünde gelişmeye başlamıştır [8]. Daha sonra Rössler, Chua gibi araştırmacılar ile hızlı ilerlemeler kaydetmiş [9, 10] ve günümüzde de halen fizik ve matematik gibi temel alanlarda çok geniş bir şekilde incelenerek gelişmesine devam etmektedir. Kaos, kendini birçok farklı yerde göstermektedir. Hava durumu tahminlerinde, sigara dumanının ortamdaki yayılışında, yıllara göre nüfus artışının incelenmesinde, evrendeki lineer olmayan birçok yapıdan tek hücreli organizmalara kadar birçok konuda kullanılmıştır.

Elektroniksel yapı olarak bir kaotik sistem incelendiğinde, kaotik denklem sistemlerini oluşturan devreler topluluğunun çoğunlukla integral ve türev alıcı devreler olduğunu görülür. İlk basit kaotik devreyi Chua yapmıştır [11]. Belli bir süre sonra bu devre kaos üretici olarak çeşitli alanlarda kullanılmıştır. Son

zamanlarda ise birçok alanda kullanılan farklı özelliklerde sayısal filtreler, osilatörler, güç devreleri, kapasitör devreleri gibi pratik elektronik devreler yapılmıştır [12].

1.3. Delta Robot

Son zamanlarda üretimde ve daha birçok alanda yüksek hızda tutup yerleştirme ve paketlenme sistemlerine talep artmaktadır. Günümüzde çok sayıda mekanik sistemlerden yüksek hassasiyetli hareket performansı istendiğinden ötürü seri manipülatöre göre paralel manipülatörler daha hassas konumlama ve daha sağlam olduğundan paralel manipülatörler oldukça önemli bir hale gelmiştir. Paralel manipülatörlerden olan delta robot, Şekil 1.16.'daki gibi sabit bir plakaya monte edilmiş 3 motor ve bu motorların her birine eklenmiş plakalar ile çalışma alanında istenilen (x, y, z) konumunda hareket yapılabilmektedir.



Şekil 1.16. Delta Robot [13]

Sabit plaka üzerindeki motorlar simetrik olarak yerleştirilmiş olup her bir motor ile kendisine bağlanan kol kontrol edilmiştir. Bu sebeple sabit ve hareketli düzlem sürekli paralel kalırlar. Dolayısıyla delta robot 3 eksenli olarak hareket eder [13, 14]. Bu yüksek lisans tezinde, yukarıdaki kapsamda bahsedilen, karıştırma-karıştırıcı tip ve yöntemleri, kaotik sistemler, delta robot, MATLAB ve Arduinio ortamları kullanılarak yeni nesil bir kaotik karıştırıcının tasarım ve gerçekleştirilmesi amaçlanmaktadır.

BÖLÜM 2. LİTERATÜR ÖZETİ

Prof. J. T. Banhero ve W. L. Bodger [15] çalışmalarında, maddelerin katı, sıvı ve gaz hallerindeki karıştırılma teorisini araştırmıştır.

Bernd Otto Biesecker [16] araştırmalarında, sıvıları karıştırıcılar kullanarak gazlanmasını, bu gazlanmadan oluşan akım şekillerini ve deneme sistemlerini gözlemlemiştir.

Vauck Müller [17] yapmış olduğu çalışmalarında, karışımın derecesi, karıştırma süresi, karıştırma yöntemleri ve karıştırıcıların standartlaştırılması için çalışmalar yapmıştır.

H. J. Henzler [18] araştırmasında, devamlı çalışan karıştırıcıların homojenliğini gözlemlemiştir.

N. İlten, tez çalışmasında karıştırıcıları, karışım yöntemlerini, karışım olaylarını ve karıştırıcı güç hesaplarını incelemiştir [19].

B. A. Can, tez çalışmasında silindirik bir kap içerisinde aksenel olarak ileri-geri hareket eden ve kendi eksenini etrafında belirli bir devirde dönen disk ile birbirinden farklı iki akışkanın homojen olarak karışmasını modellemiştir [20].

1963 yılında, M.I.T. bilimcisi E. N. Lorenz hava durumunu tahmin edebilmek için akışkan ısı-yayınımı üzerinde araştırma yaparken düzensiz salınımlar gözlemledi ve bunu modelleştirdi [21].

Li ve Yorke [22] 1975 yılında, bu davranış modelini anlatmak için “kaos” terimini kullandı. Deneysel olarak kaostun ilk gözlemleri, non otonom ve harici bir kaynakla sürülen nonlinear elektronik osilatör devrelerinde olmuştur.

Bu osilatör devreleri; Van der Pol ve Van der Mark [23] ile Kennedy ve Chua [24] tarafından yapılan AC bir kaynakla çalışan neon lamba osilatörüdür. Her ne kadar literatürde çok fazla otonom kaotik devre yapılmış olsa da en çok üzerinde durulan ve kaotik yapısı en iyi bilinen otonom sistemler Chua, Rösler ve Lorenz sistemidir [25].

Clavel 1988’de Lausanne Federal Polytechnic Institute (EPFL)’de delta robotu tasarlamıştır. Bu robotu 3D uzayda öteleme hareketleri gerçekleştirmek için yapmıştır [26].

2004 yılında ise Zsombor-Murray çalışmalarında delta robotun ileri ve ters kinematığını, merkezi sabit bir daire ve merkezi hareketli bir kürenin kesişim noktalarını analitiksel olarak modelleyip, bu hesapları bilgisayarda yapabilen yapabilen bir programı yazmışlardır [27].

İ. H. Şanlıtürk, tez çalışmasında görerek işlem yapabilen Delta robotun tasarımını yapmış ve performans karakteristiklerini araştırmıştır [28].

Qiaoling Yuan, Shiming Ji, Zhongfei Wang, Guan Wang, Yuehua Wan and Li Zhan, çalışmalarında, önerilen küp şeklindeki çalışma alanını (PCDW) elde etmek için lineer bir delta robotun (LDR) optimal tasarım yöntemini önermişlerdir [29].

Xin-Jun Liu, Jinsong Wang, Kun-Ku Oh ve Jongwon Kim, çalışmalarında bir delta robotunun küresel eklemlerinde istenilen çalışma alanını ve salınım aralığını göz önüne alarak yeni bir tasarım yöntemi sunmuşlardır [30].

K. T. Chau, Shuang Ye, Yuan Gao, J. H. Chen çalışmalarında karıştırıcı olarak geri beslemeli bir DC motor kullanılmıştır. DC motorun hızı kaotik olarak

ayarlanmaktadır ve karıştırma işlemi zaman gecikmeli olarak yapılmaktadır. Kaotik karıştırma sonuçları sabit hızda yapılan karıştırma sonuçlarıyla karşılaştırılmıştır. Yapılan deneyde asit-baz nötrleşme tepkimesi değerlendirilmiştir [31].

S. Ye ve K. T. Chau, çalışmalarında karıştırıcı olarak bir DC motor kullanılmıştır. DC motorun hızı kararsızlaştırma yöntemi ile kaotik olarak ayarlanmaktadır. Kaotik karıştırma sonuçları sabit hızda yapılan karıştırma sonuçlarıyla karşılaştırılmıştır. Yapılan deneyde asit-baz nötrleşme tepkimesi değerlendirilmiştir [32].

S. Ye ve K. T. Chau, çalışmalarında karıştırıcı olarak zaman gecikmeli geri beslemeli bir DC motor kullanılmışlardır. Kaotik karıştırma sonuçları sabit hızda yapılan karıştırma sonuçlarıyla karşılaştırılmıştır. Yapılan deneyde asit-baz nötrleşme tepkimesi değerlendirilmiştir [33].

M. A. Murtadha, M. Abdurrahman, A. I. Korman, bir sıvı karıştırıcının kaotik kontrolü üzerine çalışmışlardır. Bu çalışmalarında su - tuz karışımı 30 saniye süreyle karıştırılmıştır. Sonuçlar konsantre ölçümleri yapılarak değerlendirilmiştir [34].

Z. Zhang ve G. Chen, çalışmalarında karıştırıcı kanatları sabit tutularak karışımın içinde bulunduğu kap bir dc motor tarafından döndürülen bir plakaya sabitlenmiştir. Motorun hızı Chua devresi kullanılarak kaotik olarak ayarlanmıştır. Yapılan deneyde su-şeker karışımı değerlendirilmiştir [35].

Kavur, A. E., Demiroglu, S., Seydibeyoglu, M. Ö., Baser, Ö., Güzelis, C., Sahin, S, çalışmalarında grafen nanoplateletlerini karıştırmak için kaotik sistem tabanlı bir delta robot tasarlamışlardır [36].

BÖLÜM 3. MATERYAL VE YÖNTEM

3.1. DC Motorlar

Robotikte en çok kullanılan motor çeşididir. Çok sayıda değişik şekilleri olmakla birlikte ucuz ve güçlü olması tercih edilmesinin ana sebepleridir. Hemen hemen her alanda DC motorlar direkt yada dişli kutulu olarak kullanılmaktadır. Aşağıda DC motorların özellikleri açıklanmaktadır.

3.1.1. Yön

DC motorların yönünü değiştirmek için uçlarına düşen gerilimin polarmalandırmasını değiştirmek gerekir. Diğer bir deyişle motordan geçen akımın yönü değiştirildiğinde motorun yönü de terslenmektedir.

3.1.2. Hız

DC motorun 1 dk.'daki devir sayısı o motorun hızıdır. Birimi rpm (rotations per minute) olarak gösterilir. DC motorda hız voltaj ile doğru, yükte ters orantılı olarak değişir.

3.1.3. Gerilim

DC motorların gerilimleri 1,5 ile 48 V arasında olabilmektedir. Her bir DC motor için ayarlanan gerilim değeri o motorun en yüksek akım, güç ve hızda çalıştığı değerlerdir. Bu sebeple DC motor kullanacağımız sistemde bu gerilim değerine göre motorumuzu seçmemiz gerekmektedir.

3.1.4. Akım

Bir DC motorda akım yükü doğru orantılı olarak artış gösterir. Bu sebeple yükün motorun kaldırabileceği maksimum akımı geçmemesi gerekmektedir. Aksi takdirde motor kısa devre olup yanabilir. DC motor çalışma akımı genellikle 50 ile 2000 mA arasında olmaktadır.

3.1.5. Güç

Akım ve gerilimin çarpımına güç denir. Mekanik ve robotik sistemlerde ise sıklıkla motorun kuvveti tork cinsinden gücüne bakılır.

3.1.6. Tork

Motorun dönme momentine tork denir. Bir motorun torku ne kadar yüksekse o kadar güçlüdür. Bir motorda tork hızına ve bağlıdır. Bir motorun torku hızına, motor shaftının yarı çapına, motora bağlanan dişli kutularına, motorun mekaniksel ve elektriksel karakteristiklerine bağlıdır. Torkun yüksek olmasının istenildiği durumlardan dişli kutuları (redüktör) ile hız azaltılarak güç artırılır [37].

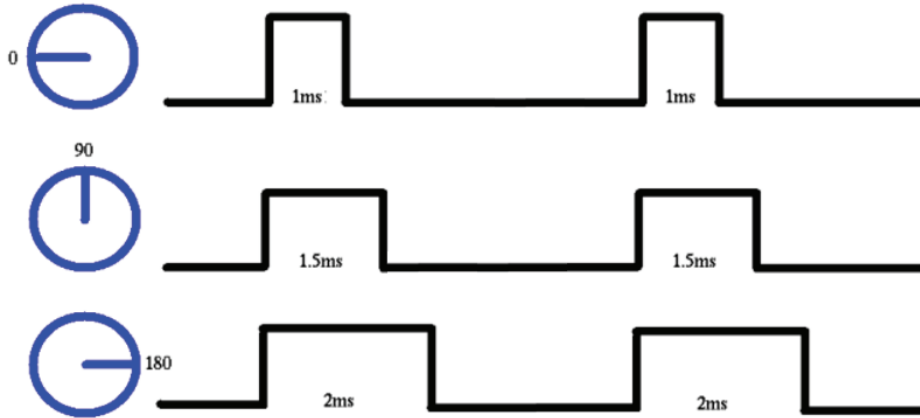
3.2. Servo Motorlar

Servo motorlar, DC motorlardan farklı olarak istenilen pozisyonda sabit kalan motorlardır. Günümüzde daha çok RC (radio controlled, uzaktan kumandalı) robotik ve ulaşım alanlarında kullanılır.



Şekil 3.1. Servo motor

Genel olarak 0° - 180° arasında PWM sinyal ile çalışırlar. PWM, servo motorun data ucuna uygulanan sinyalin 5V seviyesinde kaldığı sürenin (duty cycle) periyodik olarak değiştirilmesine denir. Örneğin analog servo motorlar 50 Hz yani 20 ms'lik periyotlarla çalışır. 20 ms'lik periyoda 1 ms ile 2 ms arasında 5V ile beslenerek, servo motora 0° - 180° arasında istenilen açı verilebilir [38, 39].



Şekil 3.2. PWM sinyalleri [39]

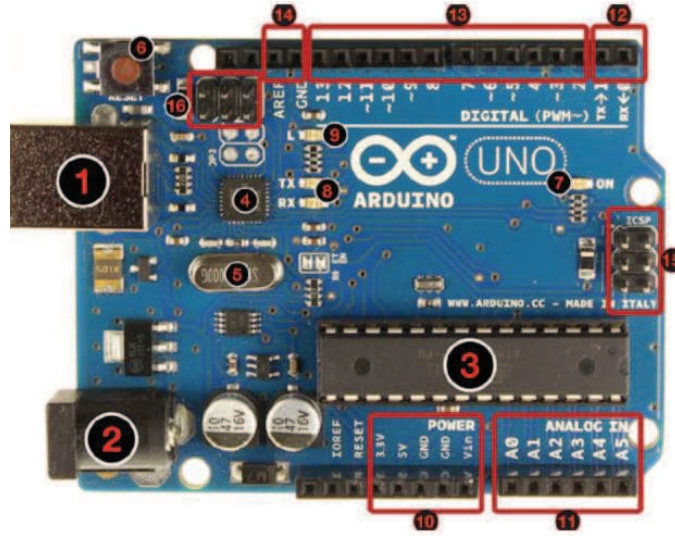
3.3. Arduino

Arduino Uno, işlemci olarak ATmega 328 entegresini seçerken birçok pin ile usb ve adaptör girişlerini de bünyesinde barındırır. Usb girişi ile bilgisayardan kolayca veri atıp okunabilmekte ve haberleşmesini sağlayabilmektedir. Adaptör ile veya 9 V 'luk pil kullanarak Arduino 'yu çalıştırabildiğimiz gibi usb ile bilgisayara bağladığımızda da başka kaynak kullanmadan da Arduino çalıştırılabilir. Arduino Uno üzerinde 3 adet led bulunmaktadır. Bunlar RX, TX ve 13. pin ledleridir. RX ve TX seri haberleşmede 13. pini ise üzerine herhangi bir led takmadan tek ledli programlarda kullanılabilir.

Arduino, açık kaynak kodlu Processing/Wiring dili ile temel I / O işlemlerini yapan bir platformdur. Yazılımını internet üzerinden Windows, Linux, Mac OS X gibi ilgili işletim sisteminize ücretsiz olarak yüklenebilir. Yazılımın kullanımı oldukça kolay

olup C ve C++ dillerine benzemektedir. Bu tezde Arduino UNO nun en son modeli olan Arduino Uno R3 'ü kullanılmıştır. Kendinden önceki tüm modellerin özelliklerine sahip olmakla birlikte daha az hafıza ile daha hızlı veri aktarımı sağlamaktadır. Şekil 3.3.'de Arduino Uno R3 'ün kısımları görülmektedir.

3.3.1. Arduino uno r3 'ün kısımları



Şekil 3.3. Arduino Uno R3' ün kısımları [40]

1. USB jakı
2. Güç jakı (7-12 V DC)
3. ATmega328 mikro denetleyicisi
4. Haberleşme entegresi
5. 16 MHz kristal
6. Reset butonu
7. Güç ledi
8. TX ve RX ledleri
9. Led
10. Besleme pinleri
11. Analog girişler
12. TX veRX pinleri
13. Dijital giriş ve çıkış pinleri (~ işaretli PWM çıkışı olarak da kullanılabilir.)

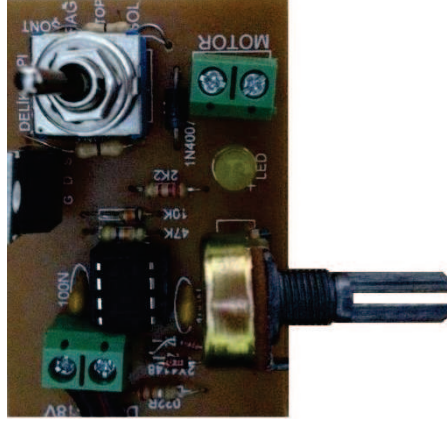
14. Toprak ve AREF pinleri
15. ATmega328 için ICSP
16. USB için ICSP

3.3.2. Arduino uno teknik özellikleri

Arduino Uno mikro denetleyici olarak ATmega328 entegresini kullanmaktadır. Çalışma gerilimi DC 5 V 'tur. Besleme gerilimi 7 ile 12 V arasında olmasının yanında 6 ile 20 V sınır değerleridir. Dijital I / O pinleri 14 tane olup 6 tanesi PWM çıkışa olanak sağlar. Analog giriş pini ise 6 tanedir. I / O pinleri DC 40 mA çıkış verebilmektedir. 3,3 V pini ise 50 mA çıkış verir. Flash hafızası 32 Kb, saat frekansı ise 16 MHz dir [40].

3.4. PWM ile DC Motor Kontrol Devresi

Genel olarak bir transistörün beyzine potansiyometre bağlayarak DC motorun hızını kontrol edildiğinde motorun ısınmasından ötürü oluşacak güç kaybı çok fazla olmaktadır. Bu yüzden son zamanlarda transistör ve beyzine bağlı potansiyometre yerine PWM kontrollü motor sürücü devreleri kullanılmaya başlanmıştır. Şekil 3.4.'te PWM kontrollü DC motor kontrol devresi görülmektedir. Devrede kullanılan potansiyometre ile 555 entegresinin ürettiği darbe genişliği değiştirilmektedir. Motorun hızı, darbenin genişliği arttıkça artıp, azaldıkça azalmaktadır. Kontrol devresi 4,5 ile 18 V arasında çalışmakta ve maksimum 4 A akım çekebilmektedir [41].



Şekil 3.4. PWM ile DC motor kontrol devresi

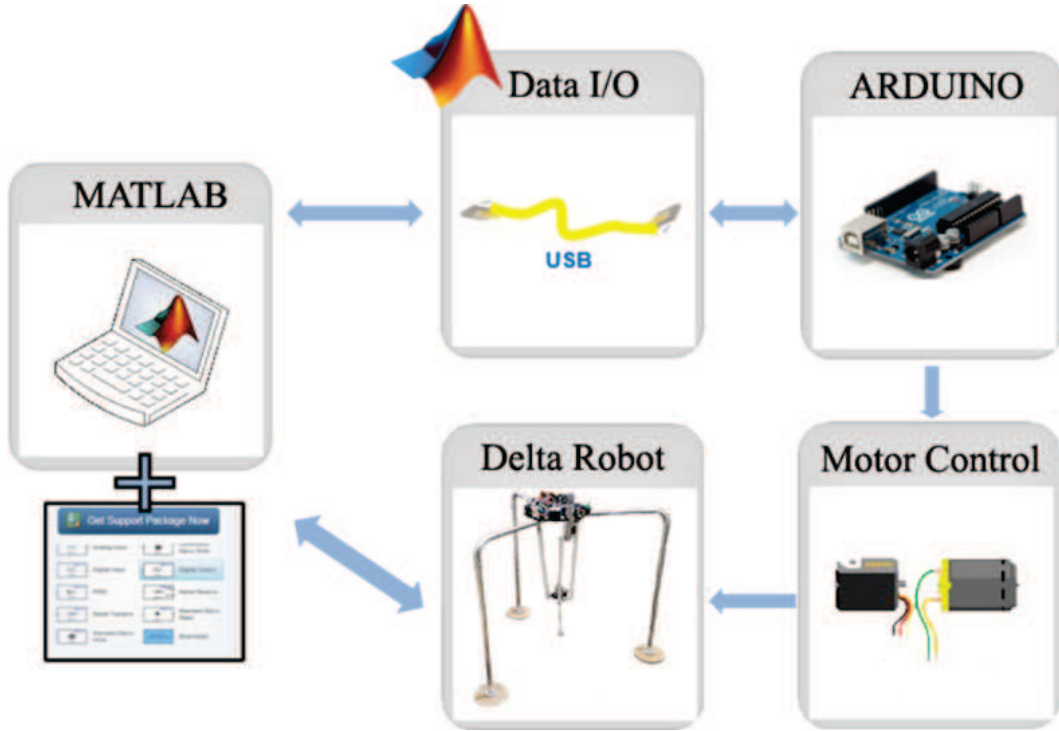
3.5. Ma871 Dijital Brix Refraktometre

Ma871 , sulu çözeltiler içindeki % çözünür kuru madde miktarını diğer bir deyişle brix belirlemek için kırılma indisi ölçümü kullanan bir optik alettir. Yöntem hem basit hem de hızlıdır. Numuneler, deiyonize veya damıtılmış su ile basit bir kullanıcı kalibrasyonundan sonra ölçülür. Birkaç saniye içinde cihaz numunenin kırılma indisini ölçer ve %Brix konsantrasyon birimine dönüştürür [42].



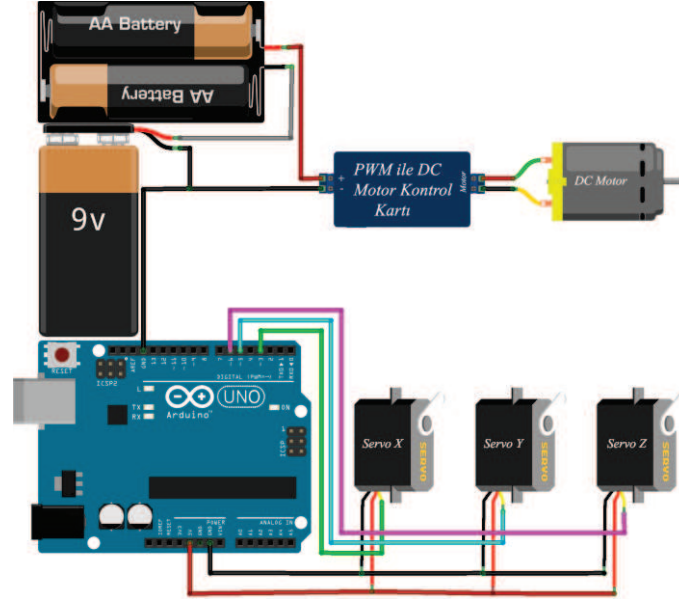
Şekil 3.5. Ma871 dijital brix refraktometri

BÖLÜM 4. KAOTİK DELTA ROBOTUN TASARLANMASI



Şekil 4.1. Delta robotun tasarım aşamaları

Öncelikle Şekil 4.1.'de de görüldüğü gibi MATLAB ile Arduino'yu haberleştirebilmek için gerekli olan Arduino yazılımı, MATLAB'ın kendi sitesi olan MathWorks'ten indirilip kurulur. Bu yazılımın yardımıyla MATLAB'ta yazılan program delta robot üzerindeki Arduino'ya yüklenir. Arduino ise bu program ile servo motorlara belirtilen iterasyon sayısına göre xyz koordinatları olarak atar. Aynı anda Şekil 4.2.'deki PWM ile DC motor kontrol kartı üzerindeki potansiyetre ile istenilen hızda ve de yönde DC motor çalıştırılır. Son olarak delta robotun izlediği xyz koordinatları MATLAB'a gönderilerek hesaplanan sistem ile gerçekleştirilen sistemin karşılaştırılması sağlanır. Tasarlanan kaotik delta robot Şekil 4.3.'de gösterilmiştir.



Şekil 4.2. Delta robotun devre şeması



Şekil 4.3. Tasarlanan delta robot

4.1. Yazılımın Hazırlanması

Programın başında kullanılacak kaotik sistem, iterasyon sayısı ve servo motor hızı seçilir. Seçilen sistem RK45 ile MATLAB'ta çözdürülür. Şekil 4.4.'deki blok diyagramda da gösterildiği gibi MATLAB'a kendi sitesinden yüklenen Arduinio paketi ile servo motorlara bağlı Arduino'nun ilişkili bitleri Matlab ile eşleştirilir.



Şekil 4.4. Delta robot yazılımı blok diyagramı

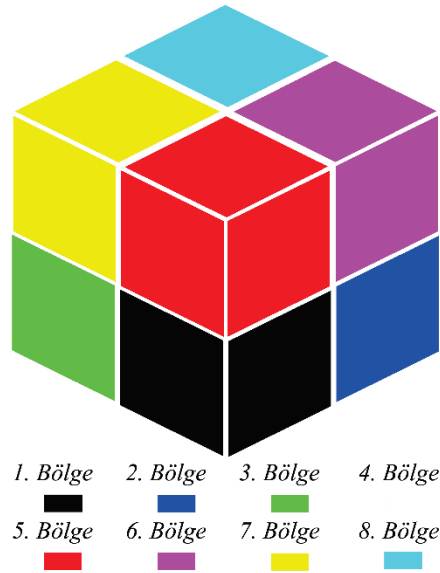
Servo motorun katalog bilgilerindeki min ve max PWM değerleri MATLAB' a girilir. Hangi bölgeden kaç nokta geçtiğini sayacak sayaçlar sıfırlanır. Seçilen kaotik sistemdeki veriler ile oluşturulan 3 sütunlu x matrisinin en büyük ve en küçük değerleri bulunarak arasındaki fark ile delta robotumuzun mekanik özelliklerinden dolayı en fazla 144° hareket edebildiği için $144'$ e bölünmesiyle çarpım değeri bulunur. x matrisindeki sütunlardan en küçük değer çıkartılıp çarpım ile çarpılarak 0° ile 144° arası skala işlemi yapılmış olur. Bu skalalama işlemi ile karıştırıcı delta robotun her bir ekseninde 0° ile 144° arası hareket edebilmesi için gerekli olan Robot Açık Değerleri (RAD) elde edilmiş olur [1, 4].

Seçmiş olduğumuz iterasyonun adım sayısını ise x matrisinin herhangi bir sütununu size komutuyla sayarak buluruz. Seçilen motor yönüne göre DC motor çalıştırılır. İterasyon sayısına göre hesaplanan adım sayısı kadar X , Y ve Z dizileri servo motorların veri girişleri olan pos_X , pos_Y ve pos_Z ' ye aktarılır. Kaotik sistem 8 bölgeye eşit şekilde bölünür. Delta robot Şekil 4.5.'deki bölgelerin hangisinden geçerse oradaki sayaç bir artar. Bu işlem adım sayısı kadar tekrar eder.

1.bölge	$0 \leq \text{pos_X} < 72,$	$0 < \text{pos_Y} \leq 72,$	$0 < \text{pos_Z} \leq 72$
2.bölge	$72 \leq \text{pos_X} < 144,$	$0 < \text{pos_Y} \leq 72,$	$0 < \text{pos_Z} \leq 72$
3.bölge	$0 \leq \text{pos_X} < 72,$	$72 < \text{pos_Y} \leq 144,$	$0 < \text{pos_Z} \leq 72$
4.bölge	$72 \leq \text{pos_X} < 144,$	$72 < \text{pos_Y} \leq 144,$	$0 < \text{pos_Z} \leq 72$
5.bölge	$0 \leq \text{pos_X} < 72,$	$0 < \text{pos_Y} \leq 72,$	$72 < \text{pos_Z} \leq 144$
6.bölge	$72 \leq \text{pos_X} < 144,$	$0 < \text{pos_Y} \leq 72,$	$72 < \text{pos_Z} \leq 144$
7.bölge	$0 \leq \text{pos_X} < 72,$	$72 < \text{pos_Y} \leq 144,$	$72 < \text{pos_Z} \leq 144$
8.bölge	$72 \leq \text{pos_X} < 144,$	$72 < \text{pos_Y} \leq 144,$	$72 < \text{pos_Z} \leq 144$

Şekil 4.5. Delta robotun her bir bölgedeki çalışma aralıkları

Hız da belirtildiği kadar adım 1 s 'de işlenir. Servo motorlara son X, Y ve Z koordinatı yüklendikten sonra DC motoru durdurur. Matlab' ta servo motorların delta robotun her bir bölgesinden kaç kez geçildiğini ve toplam kaç nokta olduğunu gösterip bölgeleri % olarak yazdırır. RK45 ile çözülen sistem ile delta robottaki sistemi iterasyon sayısına göre ve de 3 boyutlu olarak çizdirerek karşılaştırır. Delta robottaki her bir bölgeyi 0° - 144° arası Şekil 4.6.'daki gibi 8 eşit kübe bölerek bölgeleri gösterir.



Şekil 4.6. Delta robotun hareket ettiği bölgeler

4.2. Delta Robot MATLAB Programı

```
fprintf('\n 1. Dairesel Hareket\n 2. Sprott-A \n 3. Pehlivan-Wei \n 4. Aizawa \n
5. Guckenheimer-Holmes \n 6. Lu_Chen \n 7. Hadley \n 8. Halvorsen\n');
```

```
Sistem= input('\nKaotik Sistemin numarasını giriniz: ','s');
```

```
Sure= input('İterasyon sayısını giriniz(1~1000): ');
```

```
Hiz= input('Servo Motor Hızını giriniz(1~50): '); % Analog servo olduğu için hız
50 Hz Dijital servo olursa 300 Hz olur.
```

```
switch(Sistem)
```

```
    case '1'
```

```
[t,x] = ode45(@Dairesel,[0 Sure],[0.5; 0.5]);
```

```
x(:,3)= 0,5;
```

```
Sistem = 'Dairesel Hareket';
```

```
    case '2'
```

```
[t,x] = ode45(@Sprot_A,[0 Sure],[0; 0.5; 0]);
```

```
Sistem = 'Sprott-A';
```

```
    case '3'
```

```
[t,x] = ode45(@PWCS,[0 Sure],[-4; 1; -4]);
```

```
Sistem = 'Pehlivan-Wei';
```

```
    case '4'
```

```
[t,x] = ode45(@Aizawa,[0 Sure],[0.1; 0; 0]);
```

```
Sistem = 'Aizawa';
```

```
    case '5'
```

```
[t,x] = ode45(@Guckenheimer_Holmes,[0 Sure],[1; -1; 1]);
```

```
Sistem = 'Guckenheimer-Holmes';
```



```

    case '6'
[t,x] = ode45(@Lu_Chen,[0 Sure],[-3; 0; 3]);
Sistem = 'Lu-Chen';

    case '7'
[t,x] = ode45(@Hadley,[0 Sure],[0; 0; 1.3]);
Sistem = 'Hadley';

    case '8'
[t,x] = ode45(@Halvorsen,[0 Sure],[-5; 0; 0]);
Sistem = 'Halvorsen';

    otherwise
Sistem= input('Kaotik Sistemin numarasını giriniz: ','s');
end

a = arduino('com19', 'uno');
s_X = servo(a, 'D3');
s_Y = servo(a, 'D5');
s_Z = servo(a, 'D6');

clear s_X s_Y s_Z;

s_X = servo(a, 'D3', 'MinPulseDuration', 900*10^-6, 'MaxPulseDuration',
2100*10^-6);
s_Y = servo(a, 'D5', 'MinPulseDuration', 900*10^-6, 'MaxPulseDuration',
2100*10^-6);
s_Z = servo(a, 'D6', 'MinPulseDuration', 900*10^-6, 'MaxPulseDuration',
2100*10^-6);

```

```
sayac1=0; sayac2=0; sayac3=0; sayac4=0; sayac5=0; sayac6=0; sayac7=0; sayac8=0;
```

```
enbuyuk=max(max(x));
enkucuk=min(min(x));
carpim= 1/(enbuyuk-enkucuk);
X=carpim*(x(:,1)-enkucuk);
Y=carpim*(x(:,2)-enkucuk);
Z=carpim*(x(:,3)-enkucuk);
T=size(x(:,1));
```

```
for i=1:T
```

```
    fprintf('%d. adım\n',i);
    writePosition(s_X, X(i));
    pos_X(i,1)= round(readPosition(s_X)*144.00);
    fprintf('Servo motor X %d derece\n', pos_X(i,1));
    writePosition(s_Y, Y(i));
    pos_Y(i,1)= round(readPosition(s_Y)*144.00);
    fprintf('Servo motor Y %d derece\n', pos_Y(i,1));
    writePosition(s_Z, Z(i));
    pos_Z(i,1)= round(readPosition(s_Z)*144.00);
    fprintf('Servo motor Z %d derece\n', pos_Z(i,1));
```

```
        if (((pos_X(i) >= 0) && (pos_X(i) < 72)) && ((pos_Y(i) >= 0) &&
(pos_Y(i) < 72)) && ((pos_Z(i) >= 0) && (pos_Z(i) < 72)))
```

```
            sayac1=sayac1+1;
```

```
        elseif (((pos_X(i) >= 72) && (pos_X(i) <= 144)) && ((pos_Y(i) >= 0) &&
(pos_Y(i) < 72)) && ((pos_Z(i) >= 0) && (pos_Z(i) < 72)))
```

```
            sayac2=sayac2+1;
```

```
        elseif (((pos_X(i) >= 0) && (pos_X(i) < 72)) && ((pos_Y(i) >= 72) &&
(pos_Y(i) <= 144)) && ((pos_Z(i) >= 0) && (pos_Z(i) < 72)))
```

```
            sayac3=sayac3+1;
```

```

elseif (((pos_X(i) >= 72) && (pos_X(i) <= 144)) && ((pos_Y(i) >= 72) &&
(pos_Y(i) <= 144)) && ((pos_Z(i) >= 0) && (pos_Z(i) < 72)))
    sayac4=sayac4+1;
elseif (((pos_X(i) >= 0) && (pos_X(i) < 72)) && ((pos_Y(i) >= 0) &&
(pos_Y(i) < 72)) && ((pos_Z(i) >= 72) && (pos_Z(i) <= 144)))
    sayac5=sayac5+1;
elseif (((pos_X(i) >= 72) && (pos_X(i) <= 144)) && ((pos_Y(i) >= 0) &&
(pos_Y(i) < 72)) && ((pos_Z(i) >= 72) && (pos_Z(i) <= 144)))
    sayac6=sayac6+1;
elseif (((pos_X(i) >= 0) && (pos_X(i) < 72)) && ((pos_Y(i) >= 72) &&
(pos_Y(i) <= 144)) && ((pos_Z(i) >= 72) && (pos_Z(i) <= 144)))
    sayac7=sayac7+1;
else (((pos_X(i) >= 72) && (pos_X(i) <= 144)) && ((pos_Y(i) >= 72) &&
(pos_Y(i) <= 144)) && ((pos_Z(i) >= 72) && (pos_Z(i) <= 144)));
    sayac8=sayac8+1;
end
pause(1/Hiz);

```

```

sayaclar=sayac1+sayac2+sayac3+sayac4+sayac5+sayac6+sayac7+sayac8;

```

```

bolge1=round(((sayac1/sayaclar)*100),1);

```

```

bolge2=round(((sayac2/sayaclar)*100),1);

```

```

bolge3=round(((sayac3/sayaclar)*100),1);

```

```

bolge4=round(((sayac4/sayaclar)*100),1);

```

```

bolge5=round(((sayac5/sayaclar)*100),1);

```

```

bolge6=round(((sayac6/sayaclar)*100),1);

```

```

bolge7=round(((sayac7/sayaclar)*100),1);

```

```

bolge8=100-(bolge1+bolge2+bolge3+bolge4+bolge5+bolge6+bolge7);

```

```

bolgeler=bolge1+bolge2+bolge3+bolge4+bolge5+bolge6+bolge7+bolge8;

```

```

fprintf('1. Bölgeden geçen nokta sayısı= %d\n',sayac1);
fprintf('2. Bölgeden geçen nokta sayısı= %d\n',sayac2);
fprintf('3. Bölgeden geçen nokta sayısı= %d\n',sayac3);
fprintf('4. Bölgeden geçen nokta sayısı= %d\n',sayac4);
fprintf('5. Bölgeden geçen nokta sayısı= %d\n',sayac5);
fprintf('6. Bölgeden geçen nokta sayısı= %d\n',sayac6);
fprintf('7. Bölgeden geçen nokta sayısı= %d\n',sayac7);
fprintf('8. Bölgeden geçen nokta sayısı= %d\n',sayac8);
fprintf('Bölgelerden geçen toplam nokta sayısı = %d\n',sayaclar);

```

```

fprintf('1. Bölgeden yüzde olarak geçme oranı= %.1f\n',bolge1);
fprintf('2. Bölgeden yüzde olarak geçme oranı= %.1f\n',bolge2);
fprintf('3. Bölgeden yüzde olarak geçme oranı= %.1f\n',bolge3);
fprintf('4. Bölgeden yüzde olarak geçme oranı= %.1f\n',bolge4);
fprintf('5. Bölgeden yüzde olarak geçme oranı= %.1f\n',bolge5);
fprintf('6. Bölgeden yüzde olarak geçme oranı= %.1f\n',bolge6);
fprintf('7. Bölgeden yüzde olarak geçme oranı= %.1f\n',bolge7);
fprintf('8. Bölgeden yüzde olarak geçme oranı= %.1f\n',bolge8);
fprintf('Bölgelerden geçme oranlarının yüzde olarak toplamı = %d\n',bolgeler);

```

```

enbuyuk_x=max(pos_X);
enkucuk_x=min(pos_X);

```

```

enbuyuk_y=max(pos_Y);
enkucuk_y=min(pos_Y);

```

```

enbuyuk_z=max(pos_Z);
enkucuk_z=min(pos_Z);

```

```

figure
subplot(2,1,1);
plot(t,x(:,1),t,x(:,2),t,x(:,3));

```

```

title([Sistem,' Kaotik Sistemi']);
xlabel('Zaman');
ylabel('Genlik');
legend('X','Y','Z')
subplot(2,1,2);
plot(t,pos_X,t,pos_Y,t,pos_Z);
title(['Delta Robot ',Sistem,' Kaotik Sistemi']);
xlabel('Zaman');
ylabel('Derece');
legend('Servo X','Servo Y','Servo Z')

```

```

figure
subplot(2,1,1);
plot3(x(:,1),x(:,2),x(:,3));
title([Sistem,' Sistemi X-Y-Z Grafigi']);
xlabel('X');
ylabel('Y');
zlabel('Z');
grid on

```

```

subplot(2,1,2);
plot3(pos_X,pos_Y,pos_Z);
plotcube([72 72 72],[ 0 0 0],0.2,[0 0 0]); %1.bölge
plotcube([72 72 72],[72 0 0],0.2,[0 0 1]); %2.bölge
plotcube([72 72 72],[ 0 72 0],0.2,[0 1 0]); %3.bölge
plotcube([72 72 72],[72 72 0],0.2,[0 1 1]); %4.bölge
plotcube([72 72 72],[ 0 0 72],0.2,[1 0 0]); %5.bölge
plotcube([72 72 72],[72 0 72],0.2,[1 0 1]); %6.bölge
plotcube([72 72 72],[ 0 72 72],0.2,[1 1 0]); %7.bölge
plotcube([72 72 72],[72 72 72],0.2,[1 1 1]); %8.bölge
% 1.[x,y,z nin artış değerleri],2.[başl değerler],0~1 küpün saydamlığı,3.[kübün renk kodu]

```

```

title(['Delta Robot ',Sistem,' Kaotik Sistemi Servo X-Y-Z Grafiği']);
xlabel('Servo X');
ylabel('Servo Y');
zlabel('Servo Z');
grid on

```

4.3. Yörüngesel Dağılım Oranının Hesaplanması

Sistemlerin yörünge dağılım performanslarını karşılaştırabilmek için; delta robotun 8 farklı bölgeden geçiş yüzdelerini kullanarak, eş dağılım derecesinin yani Yörüngesel Dağılım Oranını (YDO)'nın hesaplanması gerekmektedir.

YDO hesabı için bir formülü oluşturabilmek adına ilk olarak, 8 farklı bölgedeki yüzde geçiş oranlarının (Denklem 4.1)'de yerine konulması gerektiği varsayılmıştır.

$$\begin{aligned}
&|12,5 - 1.\text{Bölge}| + |12,5 - 2.\text{Bölge}| + |12,5 - 3.\text{Bölge}| + |12,5 - 4.\text{Bölge}| + \\
&|12,5 - 5.\text{Bölge}| + |12,5 - 6.\text{Bölge}| + |12,5 - 7.\text{Bölge}| + |12,5 - 8.\text{Bölge}|
\end{aligned} \tag{4.1}$$

YDO'su %100 olan bir sistemin, her bir bölgeden geçiş oranının da %12,5 olması gerekir. Böyle bir sisteme ait %12,5 değerindeki veriler (Denklem 4.1)'de yerine yerleştirildiğinde 0 değeri elde edilmesine karşılık, YDO'su %0 olan (dolaşım yüzdesinin %100 ünün tek bir bölgede toplanması ve diğer bölgelerin %0 olduğu) bir sistemdeki veriler (Denklem 4.1)'de yerine konulduğunda 175 değeri elde edilir. Bu iki durum (Denklem 4.2)'de görülmektedir.

$$\begin{aligned}
&|12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| + \\
\text{Maksimum } &|12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| = 0
\end{aligned} \tag{4.2}$$

$$\begin{aligned}
\text{Minimum } &|12,5 - 100| + |12,5 - 0| + |12,5 - 0| + |12,5 - 0| + \\
&|12,5 - 0| + |12,5 - 0| + |12,5 - 0| + |12,5 - 0| = 175
\end{aligned}$$

%100 YDO değerini hesaplayabilmek için, elde edilen 0 değerinin 100 değerine; %0 YDO değerini hesaplayabilmek için, elde edilen 175 değerinin ise 0 değerine skala edilmesi gerekmektedir. Bu skala işleminin yapılmasına izin veren ve YDO hesabında kullanılan formül (Denklem 4.3)'de görülmektedir. (Denklem 4.3)'ün minimum (%0) ve maximum (%100) YDO hesabı için kullanılmasını ifade eden örnek hesaplamalar (Denklem 4.4)'de görülmektedir.

$$YDO = 100 - \left(\frac{\left(|12,5 - 1.Bölge| + |12,5 - 2.Bölge| + |12,5 - 3.Bölge| + |12,5 - 4.Bölge| + |12,5 - 5.Bölge| + |12,5 - 6.Bölge| + |12,5 - 7.Bölge| + |12,5 - 8.Bölge| \right)}{1,75} \right) \quad (4.3)$$

$$YDO_{\min} = 100 - \left(\frac{\left(|12,5 - 100| + |12,5 - 0| + |12,5 - 0| + |12,5 - 0| + |12,5 - 0| + |12,5 - 0| + |12,5 - 0| + |12,5 - 0| \right)}{1,75} \right) = 0 \quad (4.4)$$

$$YDO_{\max} = 100 - \left(\frac{\left(|12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| + |12,5 - 12,5| \right)}{1,75} \right) = 100$$

Doğal olarak; yörüngelerinin her bir bölgeden geçiş oranı homojen olan (yani YDO'su %100 olan) durum ve yörüngeleri tek bir bölgede toplanan (yani YDO'su %0 olan) durum arasında kalan tüm durumlarda, YDO orantılı bir şekilde (Denklem 4.3) kullanılarak hesaplanabilmektedir [1, 4].

Seçilen kaotik sistemin, her bir iterasyonun da delta robotun 8 bölgesinden kaç kez geçtiği, bu geçişlerin toplamı, ortalamaları, yüzdeleri, YDO ve HO oranları tabloların oluşturulması kısmında ayrıntılı olarak gösterilmiştir.

BÖLÜM 5. ARAŞTIRMA BULGULARI

5.1. Kaotik Karıştırma Deneyleri

Yapılacak olan deneylerde çözelti olarak su ve şeker kullanılmıştır. Delta robotun yapısı ve karıştırma kapasitesi hesaplanarak her deney için 20 litre çözelti hazırlanmıştır. Her çözelti için 19,5 litre su ve 0,5 litre şeker kullanılmıştır. Deneyler her sistem için 10, 30, 60, 120, 180, 240 ve 300 saniye aralıklarındaki çözeltilerin şekerlilik oranlarına göre test edilmiştir. Aşağıda örnek olarak delta robotun, yazmış olduğumuz MATLAB programı ile çözeltiyi karıştırmak üzere seçilen dairesel hareket ile 120 s boyunca karıştırılan şekerli su deneyinin yapım aşamaları gösterilmiştir.

Öncelikle Şekil 5.1.'de görüldüğü gibi brix ölçerin su ile kalibrasyonu yapılır. Sonra bilgisayar ile delta robot arasındaki usb bağlantısı yapılarak MATLAB üzerinden “Kaotik_Sistem_Delta_Robot_8” isimli program çalıştırılır. Şekil 5.2.'deki gibi dairesel hareket ve diğer parametreler seçilerek Şekil 5.3.'deki gibi delta robot çalıştırılır. 120 s sonra programı durdurduğumuzda Şekil 5.4.'deki dairesel hareketin 120 s boyunca geçtiği bölgeler ve yüzdeleri bulunmuştur. Daha sonra MATLAB programında gerçek sistem ile delta robotun birbiriyle karşılaştırmalı sonuçlarını Şekil 5.5. – 5.9.'daki gibi elde ederiz. Son olarak karıştırmadan birkaç damla alarak brix ölçere damlattığımızda Şekil 5.10.'da görülen çözeltinin Homojenlik Oranını (HO) bulmuş oluruz. Bulunan bu değerler programda belirlediğimiz her sistem için aynı zaman aralığında ve aynı iterasyon ile servo motor hızlarında tekrar edilerek MATLAB karşılaştırmaları ve tablolar aşağıdaki gibi oluşturulmuştur.



Şekil 5.1. Brix ölçerin saf su ile kalibrasyonunun yapılması

```

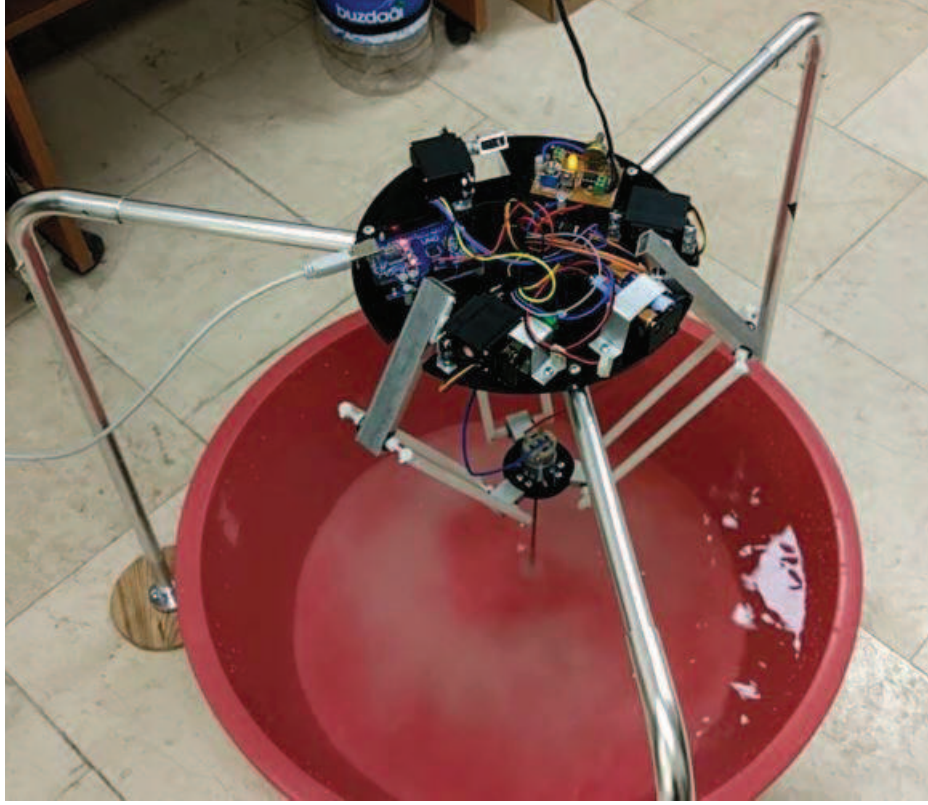
Command Window
>> Kaotik_Sistem_Delta_Robot_8

1. Dairesel Hareket
2. Sprott-A
3. Pehlivan-Wei
4. Aizawa
5. Guckenheimer-Holmes
6. Lu_Chen
7. Hadley
8. Halvorsen

Kaotik Sistemin numarasını giriniz: 1
İterasyon sayısını giriniz(1~1000): 1000
fx Servo Motor Hızını giriniz(1~50): 50

```

Şekil 5.2. Programda dairesel hareketin seçilmesi



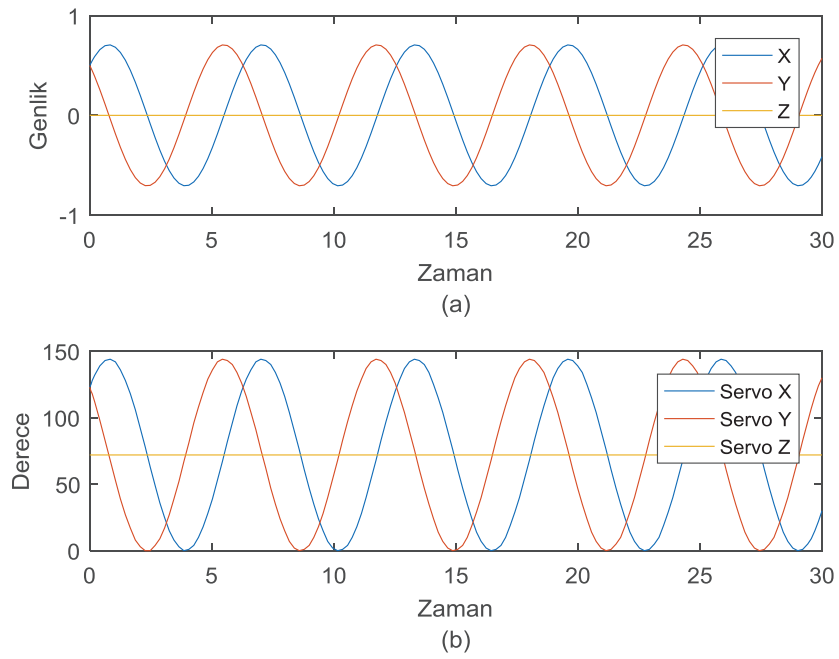
Şekil 5.3. Delta robotun çalıştırılması

```

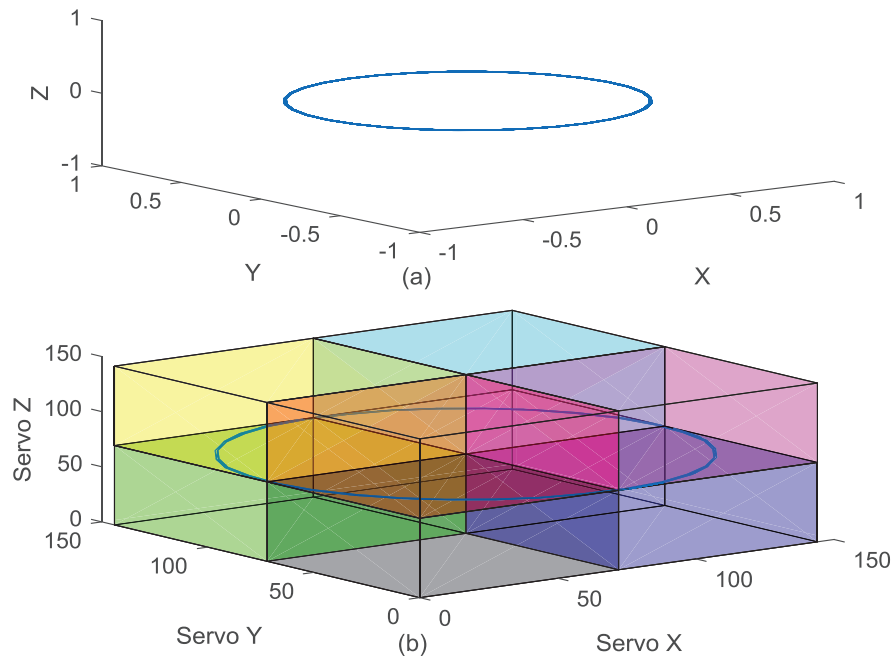
Command Window
1. Bölgeden geçen nokta sayısı= 0
2. Bölgeden geçen nokta sayısı= 0
3. Bölgeden geçen nokta sayısı= 0
4. Bölgeden geçen nokta sayısı= 0
5. Bölgeden geçen nokta sayısı= 360
6. Bölgeden geçen nokta sayısı= 360
7. Bölgeden geçen nokta sayısı= 360
8. Bölgeden geçen nokta sayısı= 365
Toplam nokta sayısı = 1445
1. Bölgeden yüzde olarak geçme oranı= 0.0
2. Bölgeden yüzde olarak geçme oranı= 0.0
3. Bölgeden yüzde olarak geçme oranı= 0.0
4. Bölgeden yüzde olarak geçme oranı= 0.0
5. Bölgeden yüzde olarak geçme oranı= 24.9
6. Bölgeden yüzde olarak geçme oranı= 24.9
7. Bölgeden yüzde olarak geçme oranı= 24.9
8. Bölgeden yüzde olarak geçme oranı= 25.3
Toplam yüzde = 100
fx >>

```

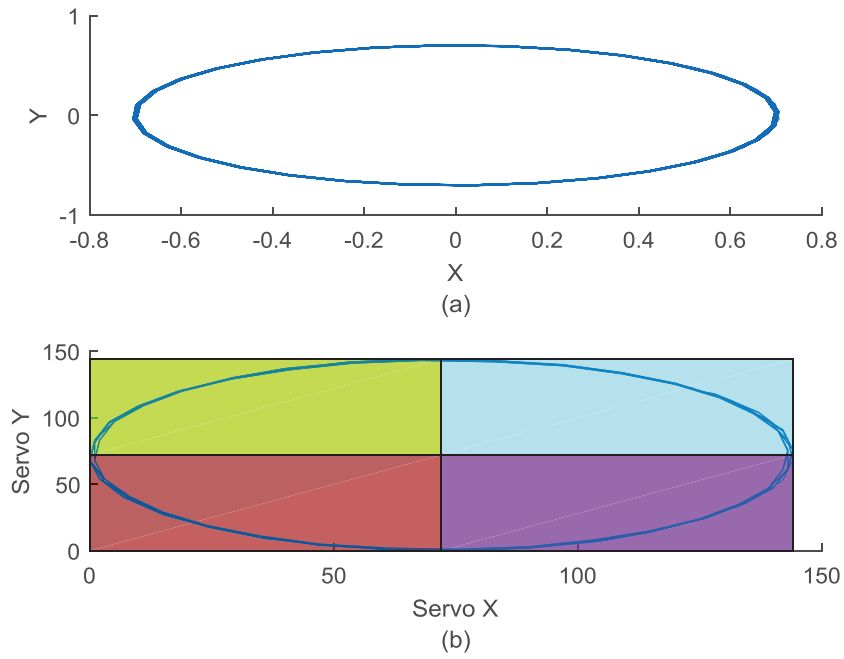
Şekil 5.4. Dairesel hareketin 120. saniyedeki geçtiği bölgeler ve yüzdeleri



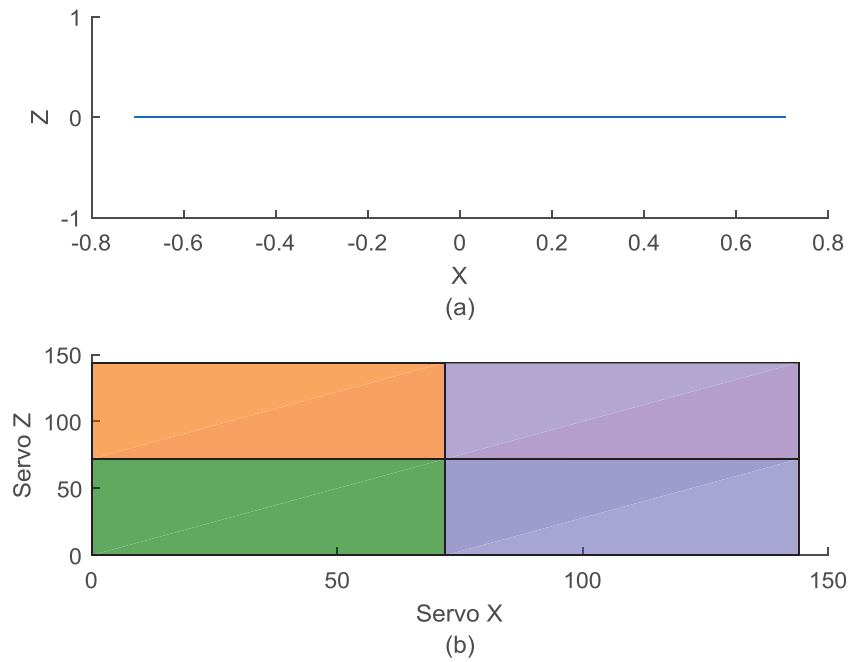
Şekil 5.5. (a) Dairesel hareket x-y-z grafiği (b) Delta robot daireysel hareket servo x-y-z grafiği



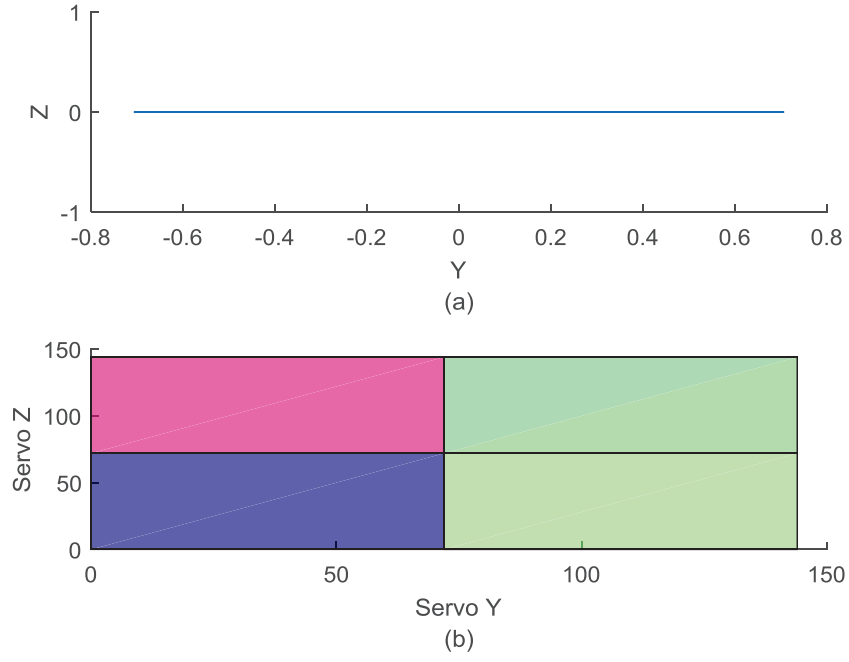
Şekil 5.6. (a) 3 Boyutlu daireysel hareket (b) 3 Boyutlu delta robot daireysel hareket



Şekil 5.7. (a) Dairesel hareket x-y faz portresi (b) Delta robot dairesel hareket servo x-y faz portresi



Şekil 5.8. (a) Dairesel hareket x-z faz portresi (b) Delta robot dairesel hareket servo x-z faz portresi



Şekil 5.9. (a) Dairesel hareket y-z faz portresi (b) Delta robot dairesel hareket servo y-z faz portresi



Şekil 5.10. Dairesel hareket ile 120 s karıştırılan çözeltinin şeker oranının brix ölçer ile ölçülmesi

5.2. Seçilen Kaotik Sistemler

Tasarlanan delta robot tabanlı yeni kaotik karıştırıcıda kullanılabilme potansiyellerini araştırmak amacıyla, örnekleme yöntemiyle aşağıdaki 7 farklı kaotik sistem seçilerek performans karşılaştırmaları yapılmıştır. Seçilen örnek kaotik sistemlerin diferansiyel denklemleri aşağıda sırasıyla verilmiştir.

5.2.1. Sprott A sistemi

Sprott A kaotik çekicinin diferansiyel denklemleri (Denklem 5.1) 'de gösterilmekte olup başlangıç şartları $x(0)=0$, $y(0)=0.5$ ve $z(0)=0$ 'dır.

$$\begin{aligned}\dot{x} &= y \\ \dot{y} &= -x + y \cdot z \\ \dot{z} &= 1 - y^2\end{aligned}\tag{5.1}$$

5.2.2. Pehlivan-Wei sistemi

Pehlivan-Wei kaotik çekicinin diferansiyel denklemleri (Denklem 5.2) 'de gösterilmekte olup başlangıç şartları $x(0)=-4$, $y(0)=1$ ve $z(0)=-4$ 'tür.

$$\begin{aligned}\dot{x} &= y - y \cdot z \\ \dot{y} &= y + y \cdot z - 2 \cdot x \\ \dot{z} &= 2 - x \cdot y - y^2\end{aligned}\tag{5.2}$$

5.2.3. Aizawa sistemi

Aizawa kaotik çekicinin diferansiyel denklemleri (Denklem 5.3) 'de gösterilmekte olup başlangıç şartları $x(0)=0.1$, $y(0)=0$ ve $z(0)=0$ 'dır.

$$\begin{aligned}
\dot{x} &= (z-0.7) \cdot x - 3.5 \cdot y \\
\dot{y} &= 3.5 \cdot x + (z-0.7) \cdot y \\
\dot{z} &= 0.6 + 0.95 \cdot z - \frac{z^3}{3} - (x^2 + y^2) \cdot (1 + 0.25 \cdot z) + 0.1 \cdot z \cdot x^3
\end{aligned} \tag{5.3}$$

5.2.4. Guckenheimer-Holmes sistemi

Guckenheimer-Holmes kaotik çekicinin diferansiyel denklemleri (Denklem 5.4) 'de gösterilmekte olup başlangıç şartları $x(0)=1$, $y(0)=-1$ ve $z(0)=1$ 'dir.

$$\begin{aligned}
\dot{x} &= 0.4 \cdot x - 20.25 \cdot y + 3 \cdot z \cdot x + 1.6 \cdot z \cdot (x^2 + y^2) \\
\dot{y} &= 0.4 \cdot y + 20.25 \cdot x + 3 \cdot z \cdot y \\
\dot{z} &= 1.7 - z^2 - 0.44 \cdot (x^2 + y^2) - 0.4 \cdot z^3
\end{aligned} \tag{5.4}$$

5.2.5. Lu-Chen sistemi

Lu-Chen kaotik çekicinin diferansiyel denklemleri (Denklem 5.5) 'de gösterilmekte olup başlangıç şartları $x(0)=-3$, $y(0)=0$ ve $z(0)=3$ 'tür.

$$\begin{aligned}
\dot{x} &= 5 \cdot x - y \cdot z \\
\dot{y} &= -10 \cdot y + x \cdot z \\
\dot{z} &= -3.4 \cdot z + x \cdot y
\end{aligned} \tag{5.5}$$

5.2.6. Hadley sistemi

Hadley kaotik çekicinin diferansiyel denklemleri (Denklem 5.6) 'de gösterilmekte olup başlangıç şartları $x(0)=0$, $y(0)=0$ ve $z(0)=1.3$ 'tür.

$$\begin{aligned}
\dot{x} &= -y^2 - z^2 - 0.25 \cdot x + 2 \\
\dot{y} &= x \cdot y - 4 \cdot x \cdot z - y + 1 \\
\dot{z} &= 4 \cdot x \cdot y + x \cdot z - z
\end{aligned} \tag{5.6}$$

5.2.7. Halvorsen sistemi

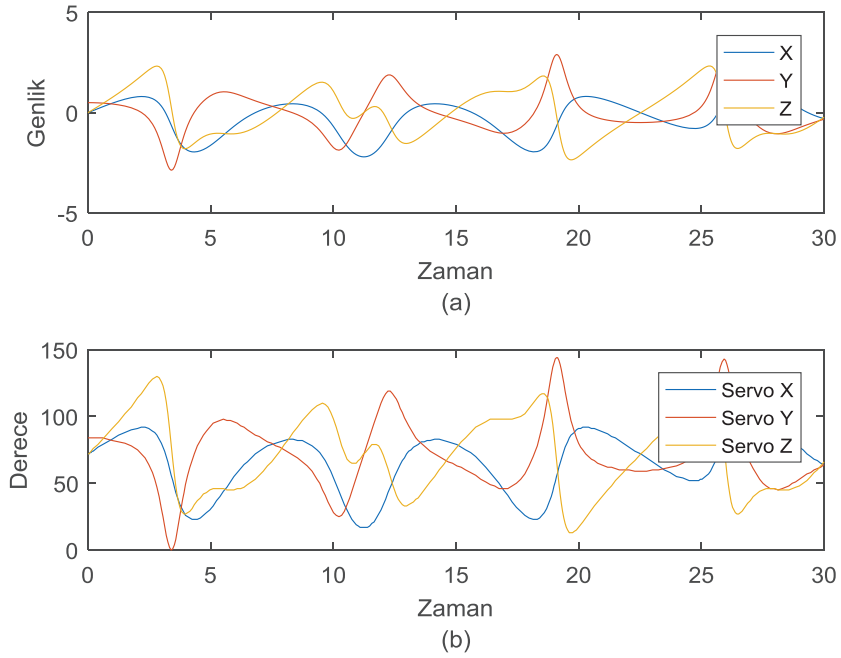
Halvorsen kaotik çekicinin diferansiyel denklemleri (Denklem 5.7) 'de gösterilmekte olup başlangıç şartları $x(0)=-5$, $y(0)=0$ ve $z(0)=0$ 'dır.

$$\begin{aligned}\dot{x} &= -1.27 \cdot x - 4 \cdot y - 4 \cdot z - y^2 \\ \dot{y} &= -1.27 \cdot y - 4 \cdot z - 4 \cdot x - z^2 \\ \dot{z} &= -1.27 \cdot z - 4 \cdot x - 4 \cdot y - x^2\end{aligned}\tag{5.7}$$

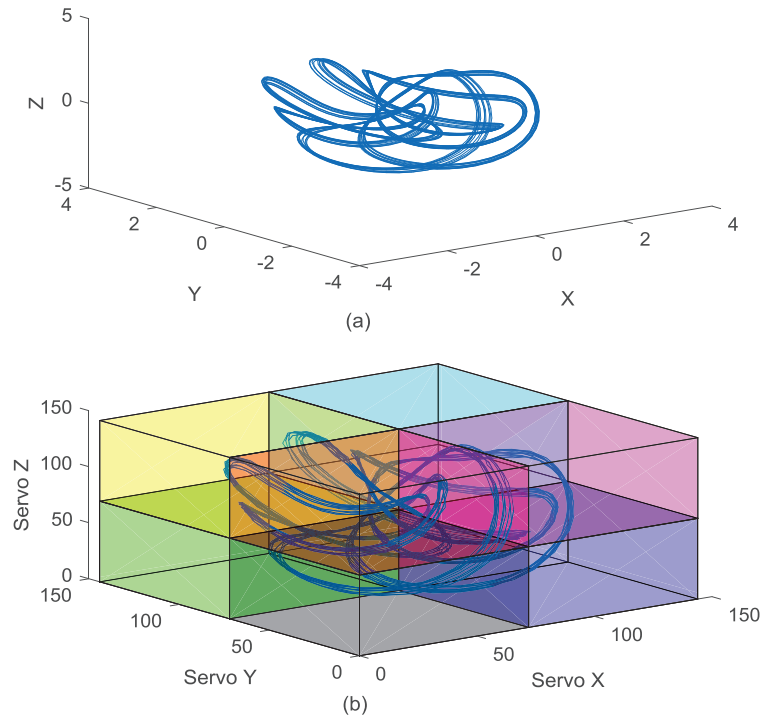
5.3. MATLAB Karşılaştırmaları

Bölüm 5.1.'deki deney her bir sistem için belirlenen zaman aralıklarında tekrar edilerek, şekil 5.11 ile 5.15 arasında Sprott A sisteminin gerçek ve delta robot karşılaştırmaları, şekil 5.16 ile 5.20 arasında Pehlivan-Wei sisteminin gerçek ve delta robot karşılaştırmaları, şekil 5.21 ile 5.25 arasında Aizawa sisteminin gerçek ve delta robot karşılaştırmaları, şekil 5.26 ile 5.30 arasında Guckenheimer-Holmes sisteminin gerçek ve delta robot karşılaştırmaları, şekil 5.31 ile 5.35 arasında Lu-Chen sisteminin gerçek ve delta robot karşılaştırmaları, şekil 5.36 ile 5.40 arasında Hadley sisteminin gerçek ve delta robot karşılaştırmaları, şekil 5.41 ile 5.45 arasında Halvorsen sisteminin gerçek ve delta robot karşılaştırmaları, aşağıdaki gibi elde edilmiştir.

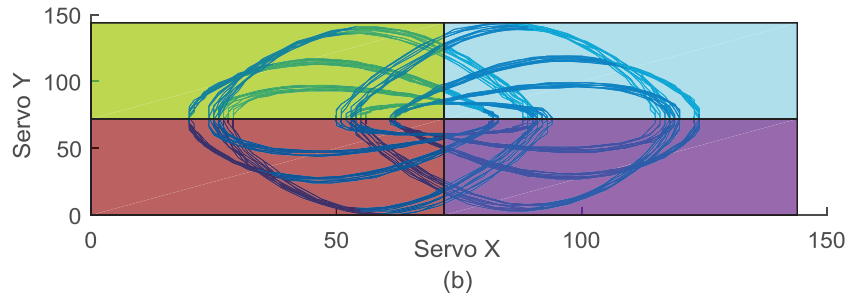
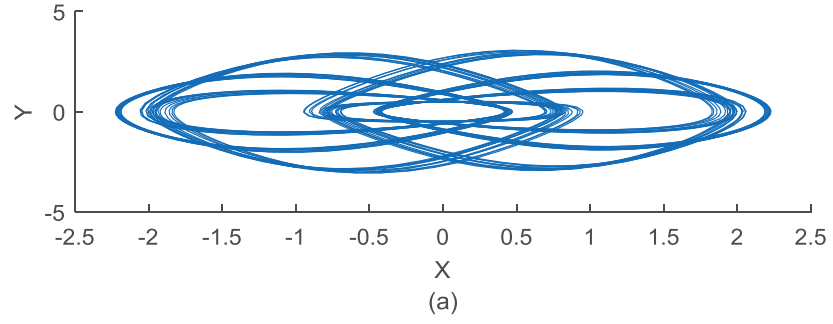
5.3.1. Sprott A sistemi MATLAB karşılaştırmaları



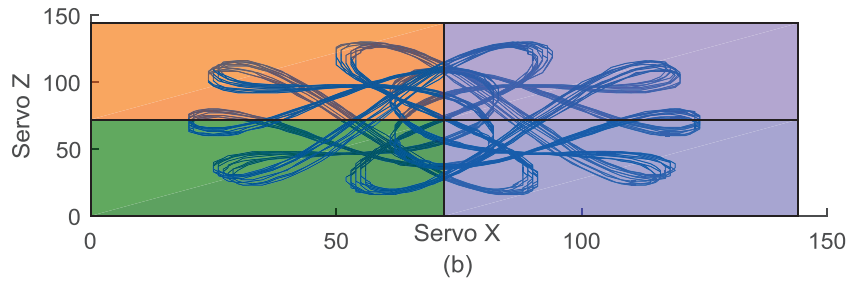
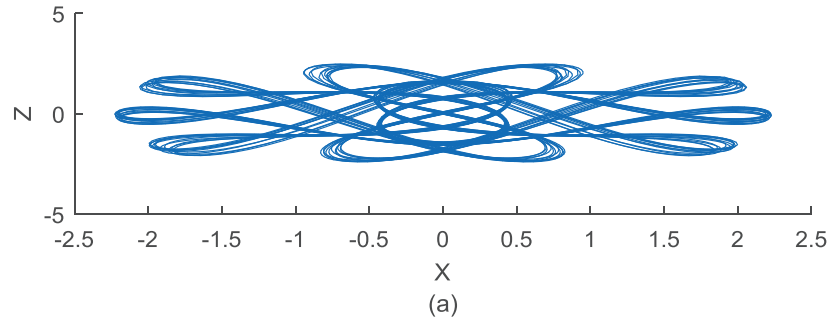
Şekil 5.11. (a) Sprott-A sistemi x-y-z grafiği (b) Delta robot Sprott-A sistemi servo x-y-z grafiği



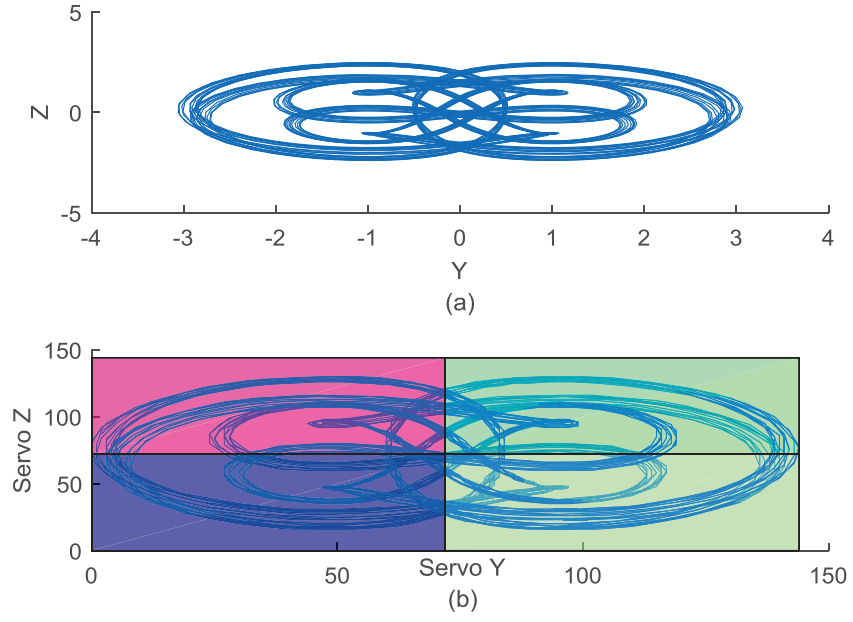
Şekil 5.12. (a) 3 Boyutlu Sprott-A sistemi (b) 3 Boyutlu delta robot Sprott-A sistemi



Şekil 5.13. (a) Sprott-A sistemi x-y faz portresi (b) Delta robot Sprott-A sistemi servo x-y faz portresi

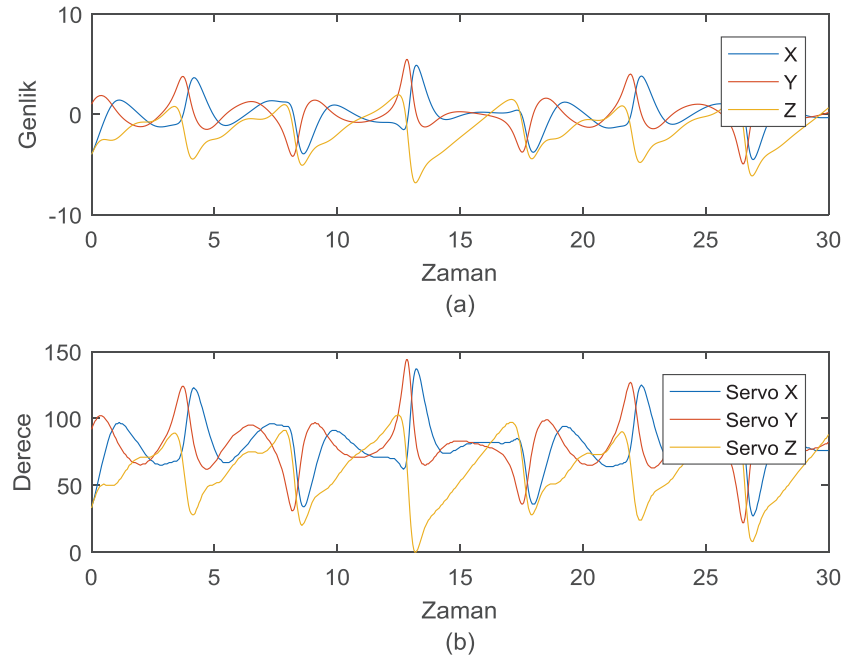


Şekil 5.14. (a) Sprott-A sistemi x-z faz portresi (b) Delta robot Sprott-A sistemi servo x-z faz portresi

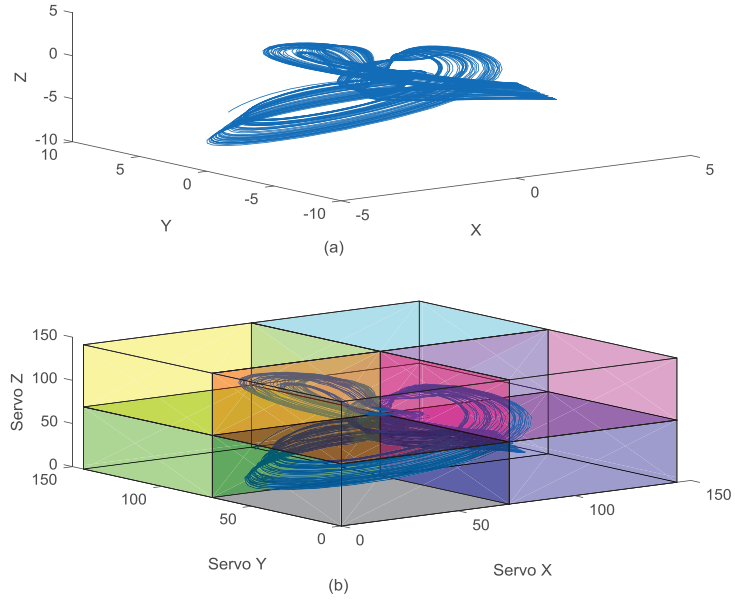


Şekil 5.15. (a) Sprott-A sistemi y-z faz portresi (b) Delta robot Sprott-A sistemi servo y-z faz portresi

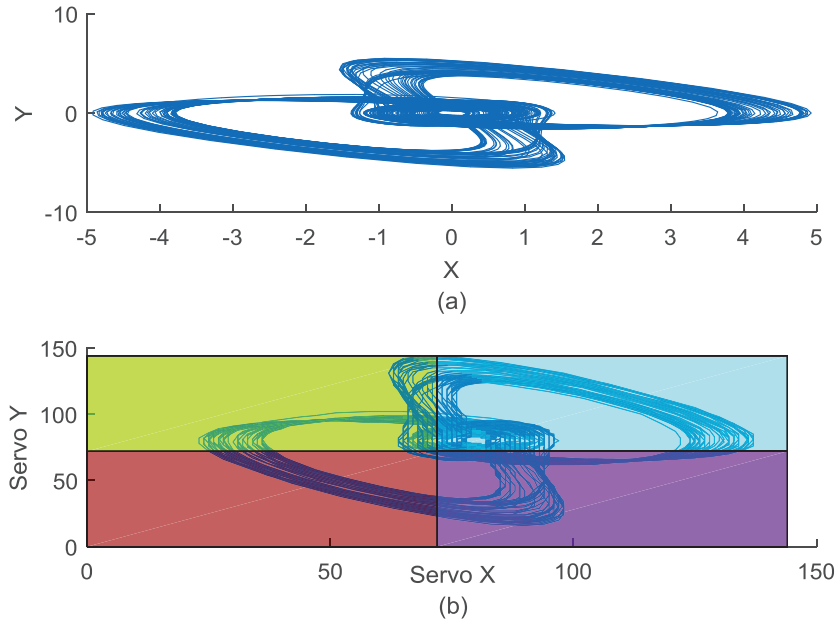
5.3.2. Pehlivan-Wei sistemi MATLAB karşılaştırmaları



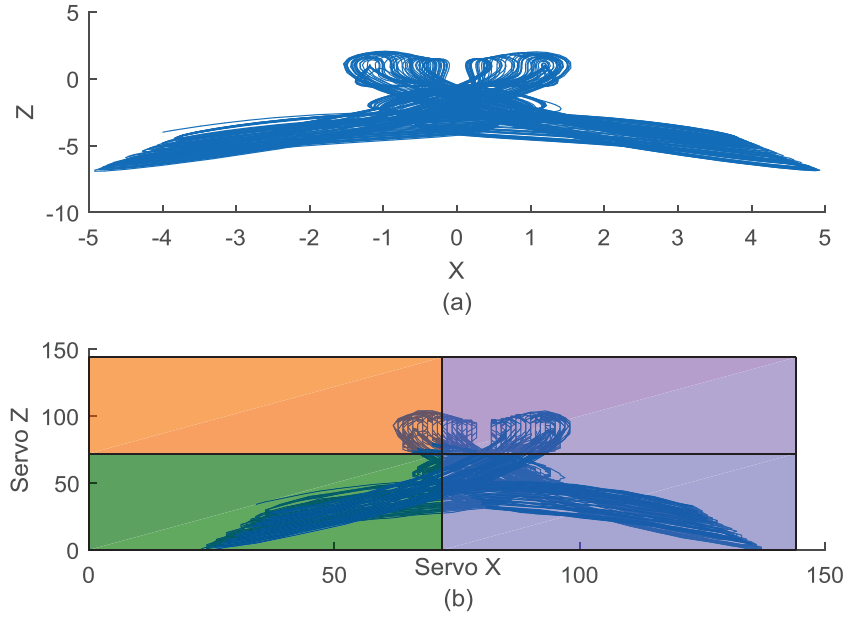
Şekil 5.16. (a) Pehlivan-Wei sistemi x-y-z grafiği (b) Delta robot Pehlivan-Wei sistemi servo x-y-z grafiği



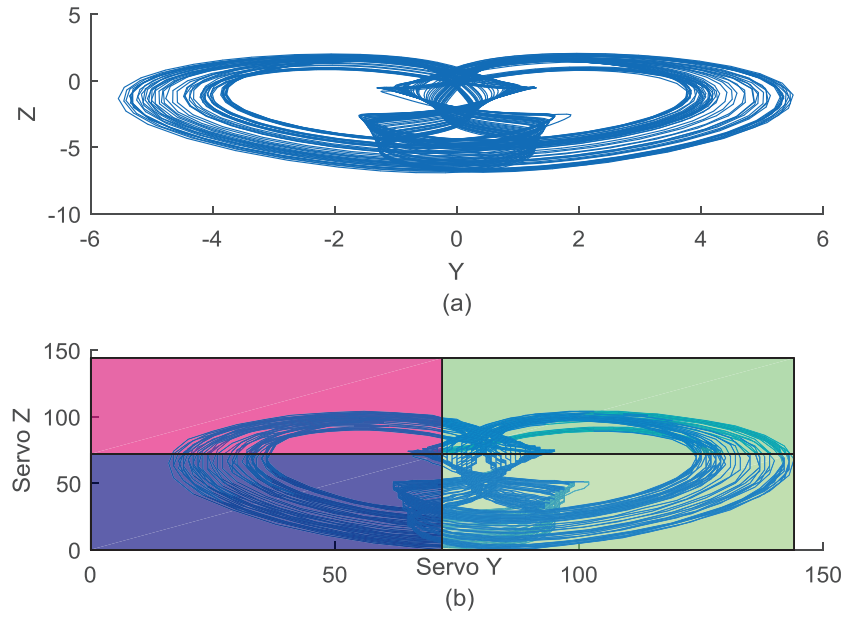
Şekil 5.17. (a) 3 Boyutlu Pehlivan-Wei sistemi (b) 3 Boyutlu delta robot Pehlivan-Wei sistemi



Şekil 5.18. (a) Pehlivan-Wei sistemi x-y faz portresi (b) Delta robot Pehlivan-Wei sistemi servo x-y faz portresi

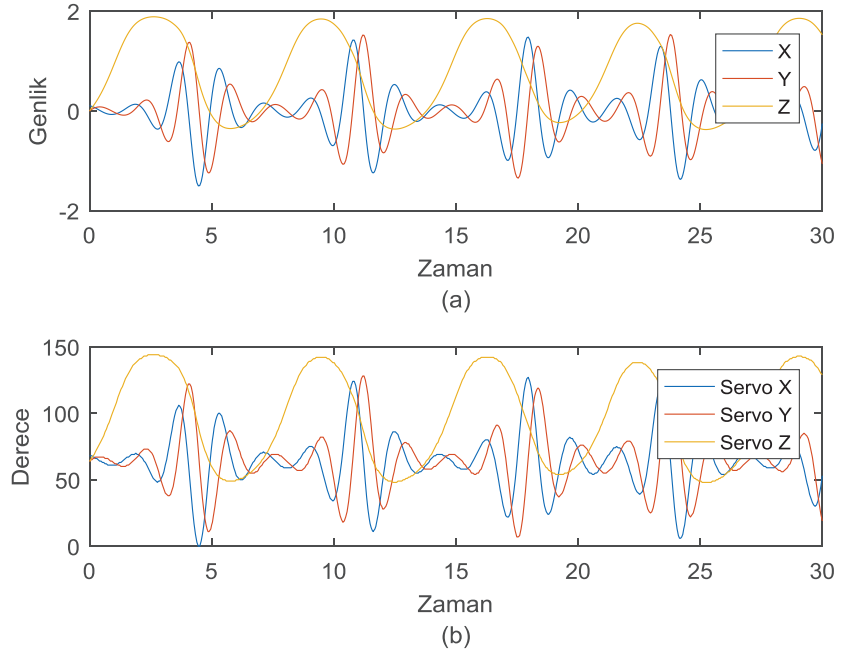


Şekil 5.19. (a) Pehlivan-Wei sistemi x-z faz portresi (b) Delta robot Pehlivan-Wei sistemi servo x-z faz portresi

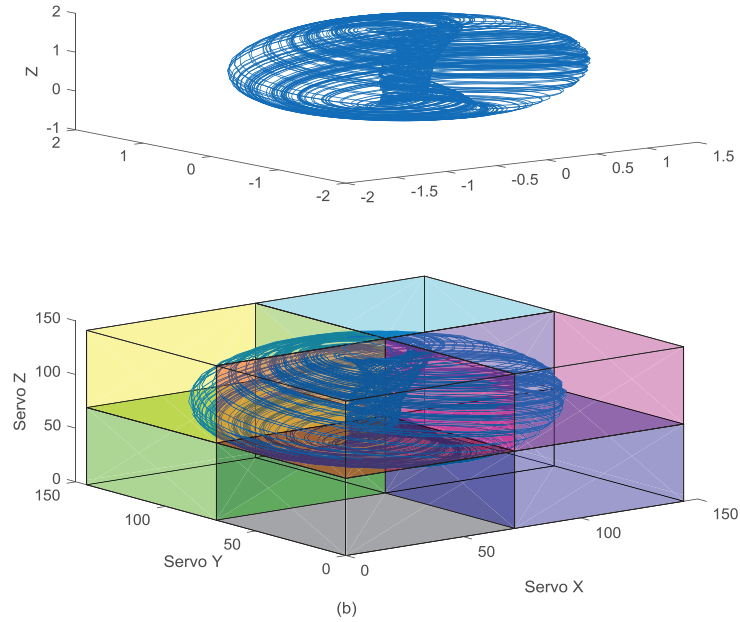


Şekil 5.20. (a) Pehlivan-Wei sistemi y-z faz portresi (b) Delta robot Pehlivan-Wei sistemi servo y-z faz portresi

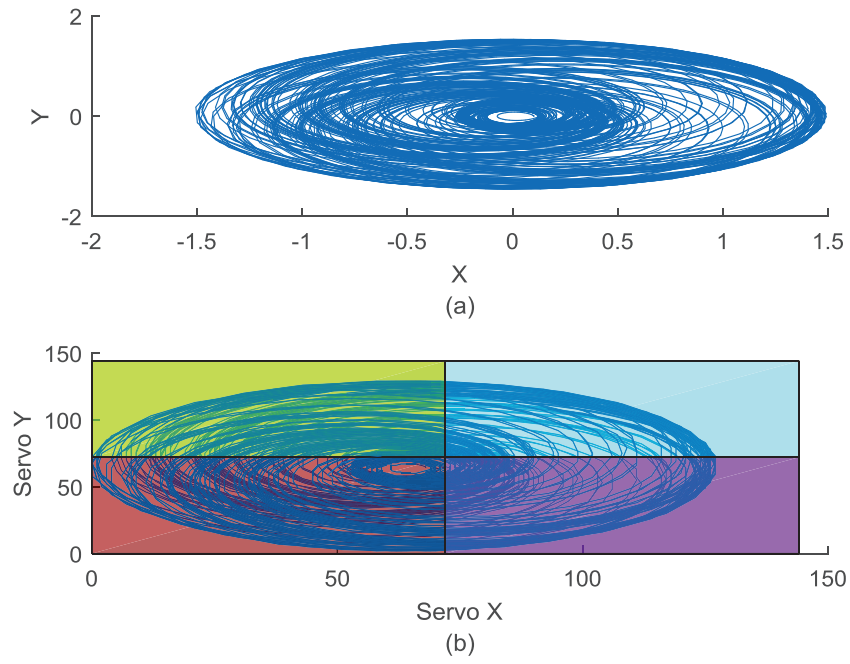
5.3.3. Aizawa sistemi MATLAB karşılaştırmaları



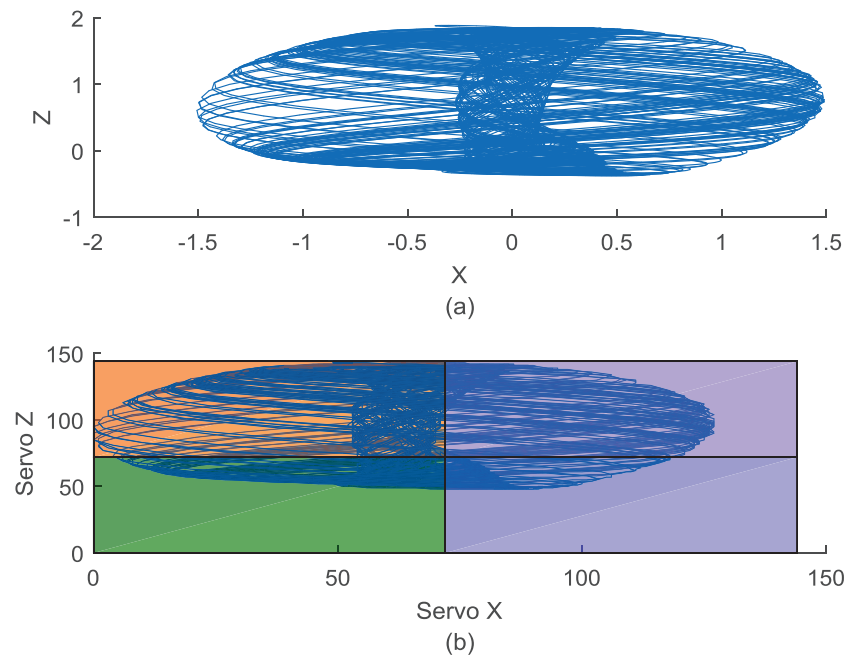
Şekil 5.21. (a) Aizawa sistemi x-y-z grafiği (b) Delta robot Aizawa sistemi servo x-y-z grafiği



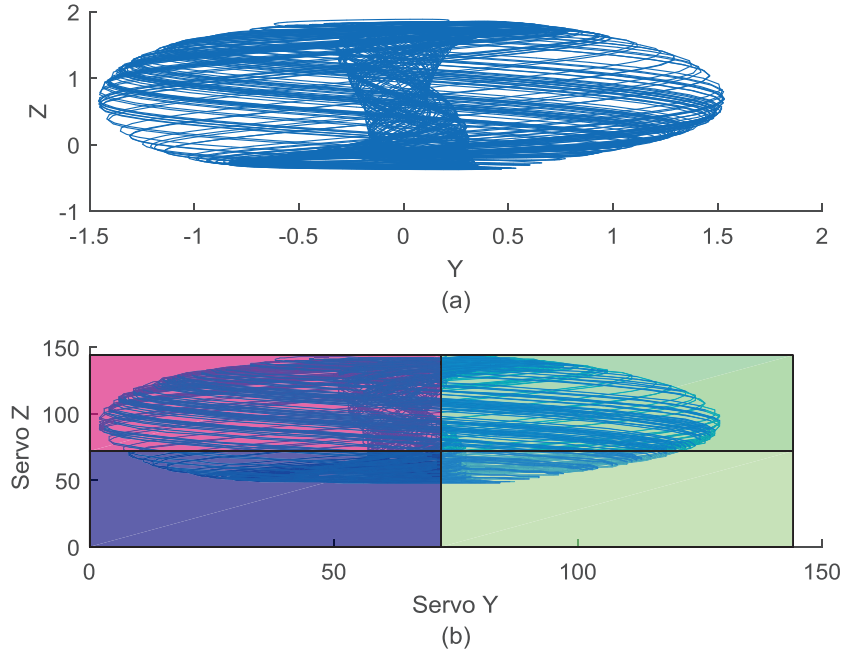
Şekil 5.22. (a) 3 Boyutlu Aizawa sistemi (b) 3 Boyutlu delta robot Aizawa sistemi



Şekil 5.23. (a) Aizawa sistemi x-y faz portresi (b) Delta robot Aizawa sistemi servo x-y faz portresi

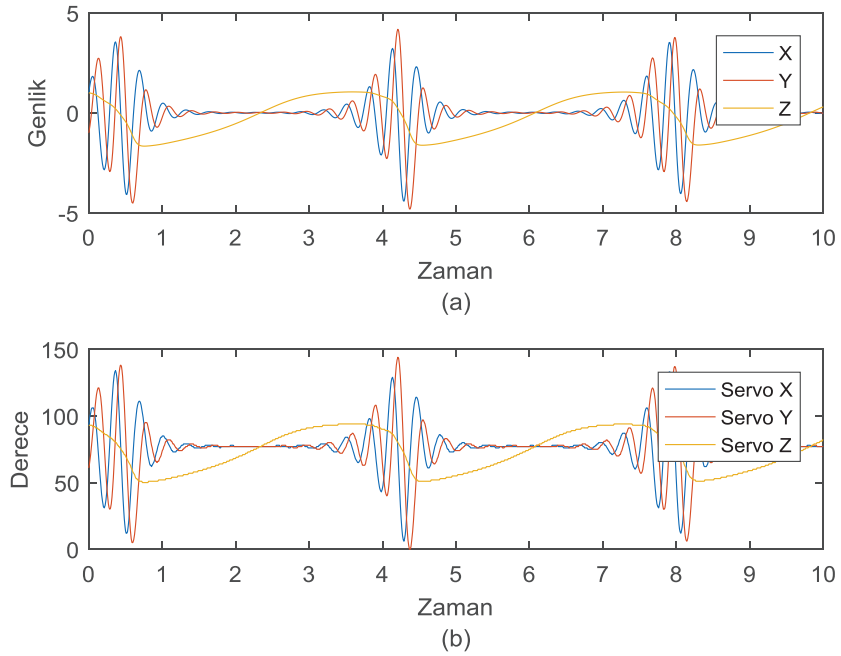


Şekil 5.24. (a) Aizawa sistemi x-z faz portresi (b) Delta robot Aizawa sistemi servo x-z faz portresi

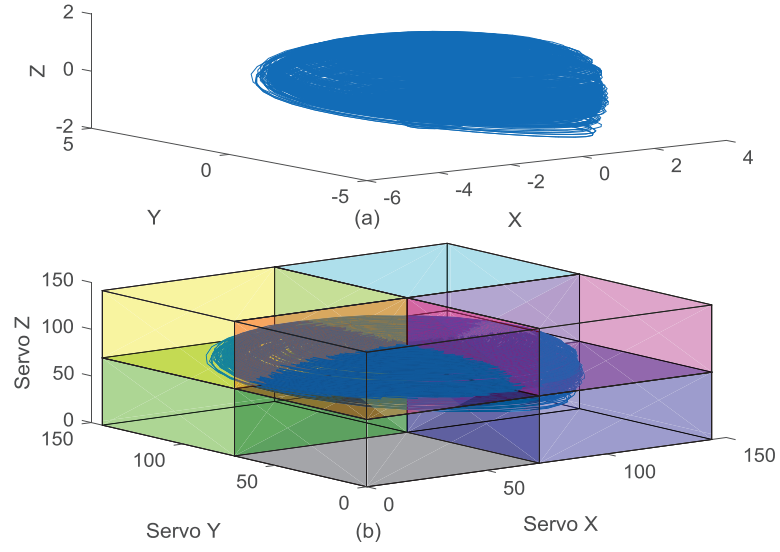


Şekil 5.25. (a) Aizawa sistemi y-z faz portresi (b) Delta robot Aizawa sistemi servo y-z faz portresi

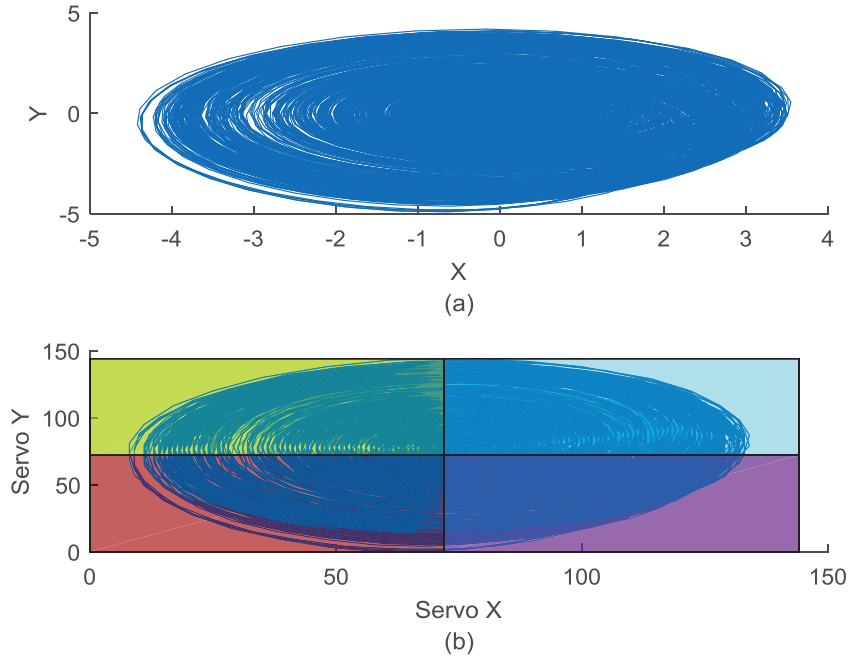
5.3.4. Guckenheimer-Holmes sistemi MATLAB karşılaştırmaları



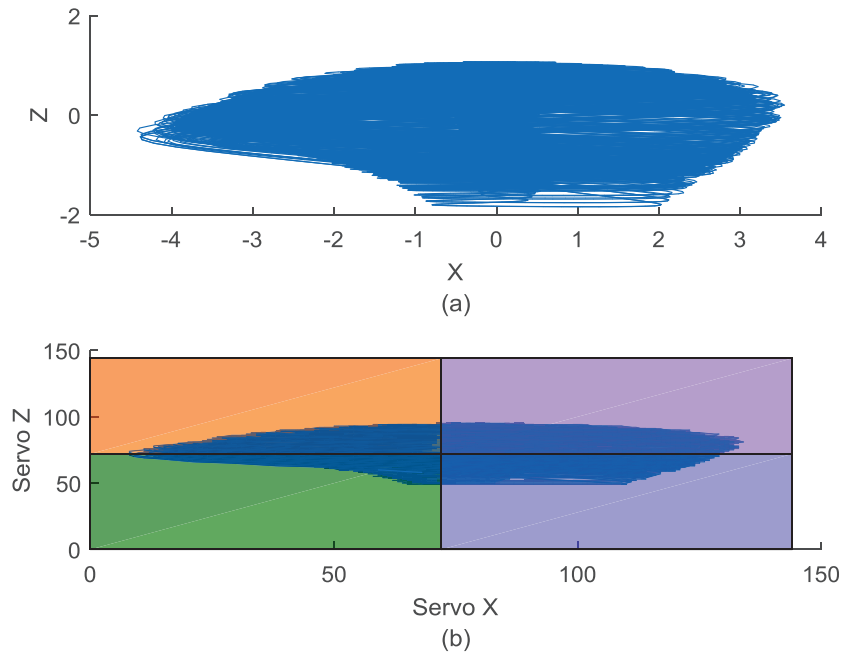
Şekil 5.26. (a) Guckenheimer-Holmes sistemi x-y-z grafiği (b) Delta robot Guckenheimer-Holmes sistemi servo x-y-z grafiği



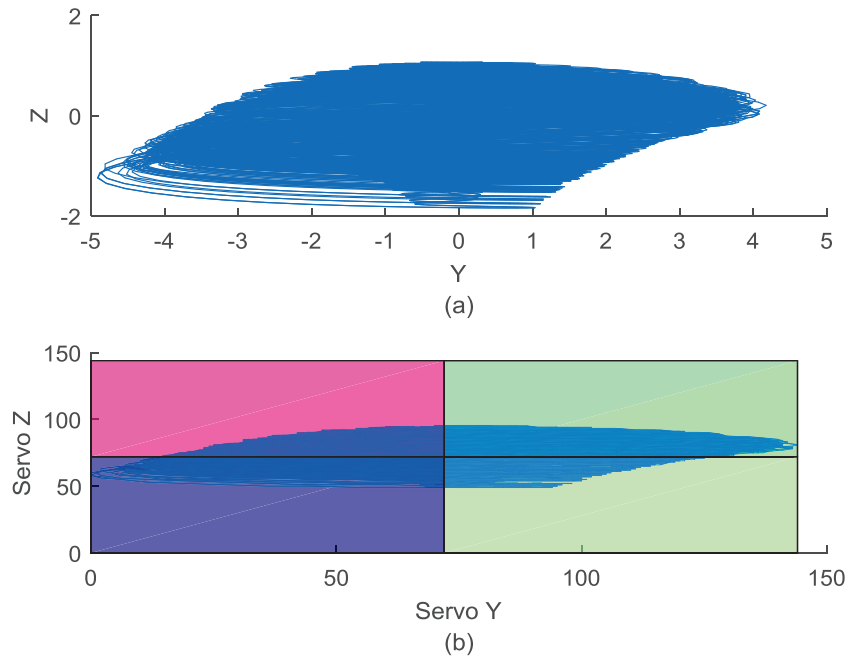
Şekil 5.27. (a) 3 Boyutlu Guckenheimer-Holmes sistemi (b) 3 Boyutlu delta robot Guckenheimer-Holmes sistemi



Şekil 5.28. (a) Guckenheimer-Holmes sistemi x-y faz portresi (b) Delta robot Guckenheimer-Holmes sistemi servo x-y faz portresi

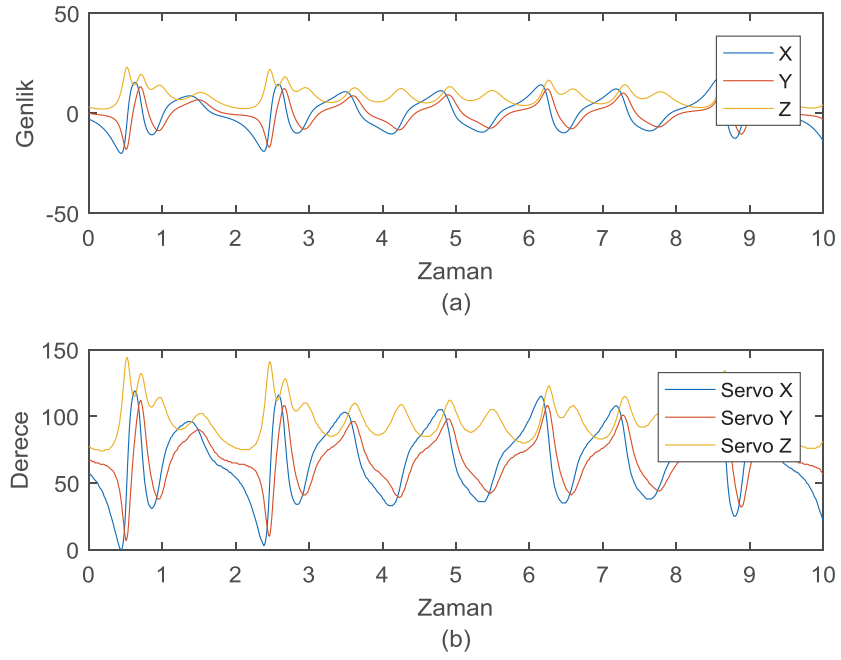


Şekil 5.29. (a) Guckenheimer-Holmes sistemi x-z faz portresi (b) Delta robot Guckenheimer-Holmes sistemi servo x-z faz portresi

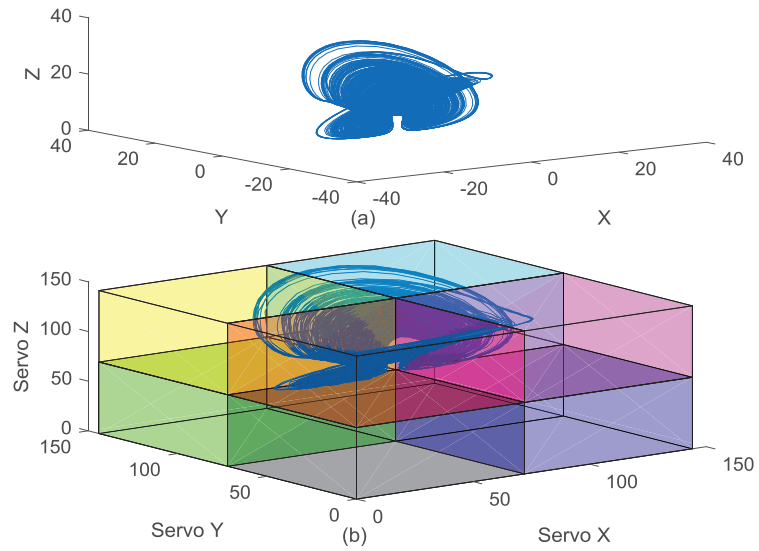


Şekil 5.30. (a) Guckenheimer-Holmes sistemi y-z faz portresi (b) Delta robot Guckenheimer-Holmes sistemi servo y-z faz portresi

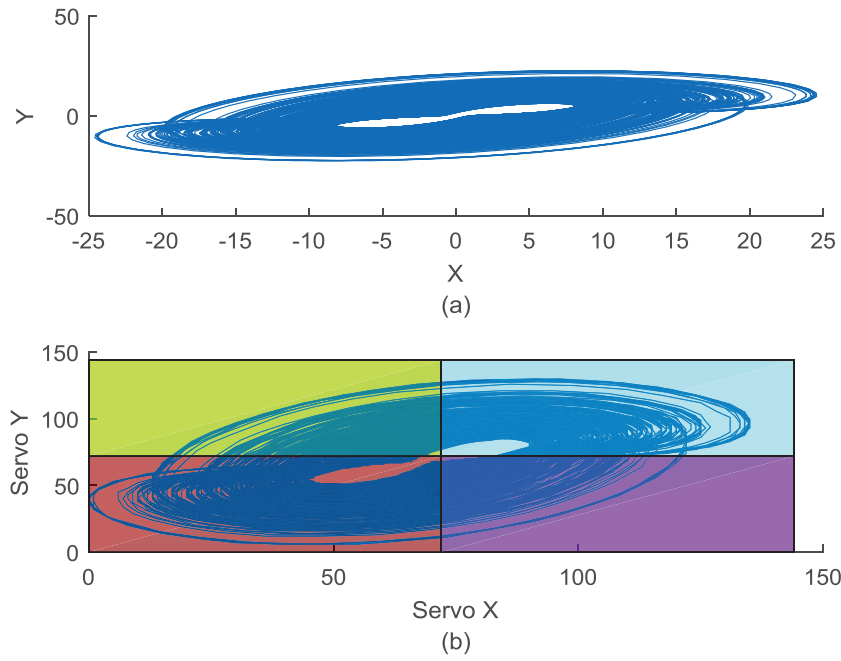
5.3.5. Lu-Chen sistemi MATLAB karşılaştırmaları



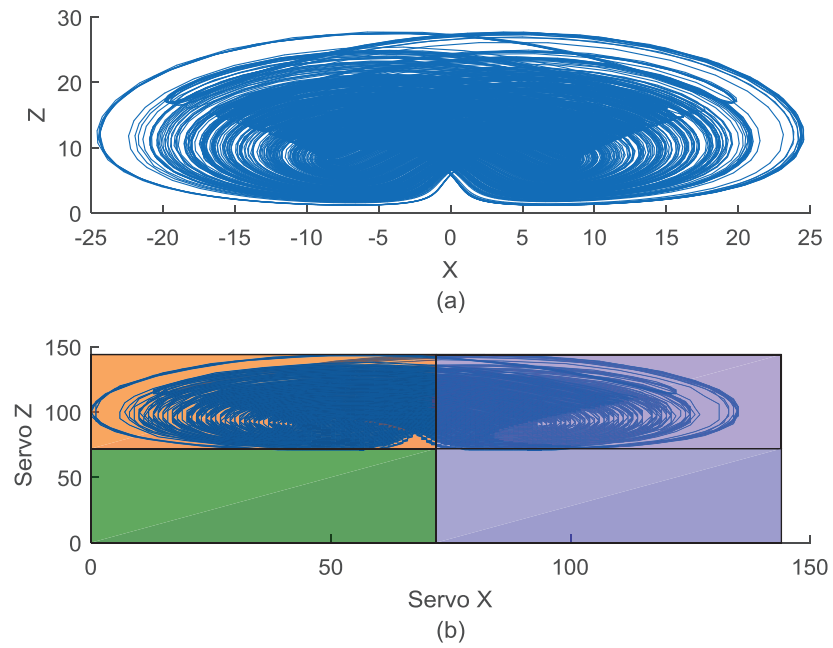
Şekil 5.31. (a) Lu-Chen sistemi x-y-z grafiği (b) Delta robot Lu-Chen sistemi servo x-y-z grafiği



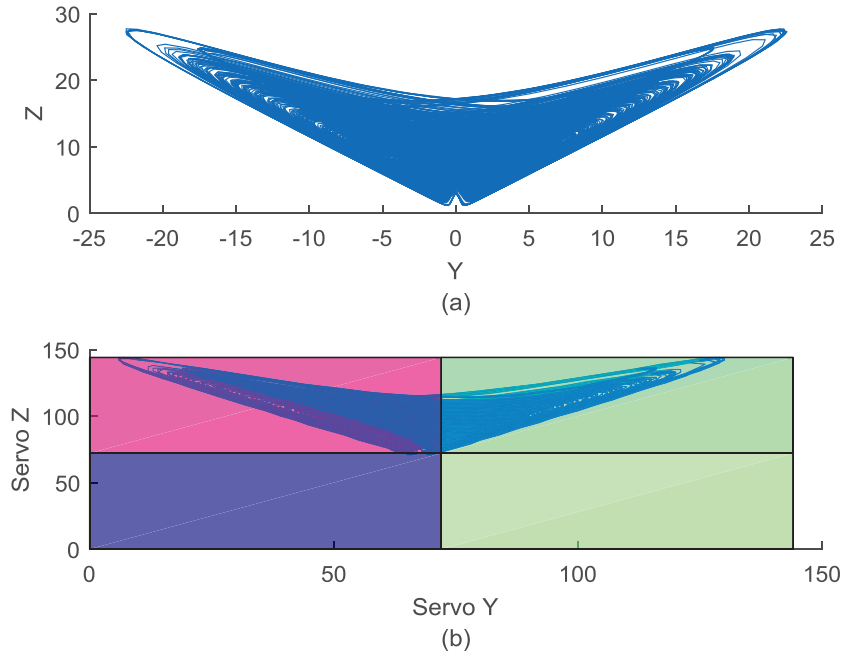
Şekil 5.32. (a) 3 Boyutlu Lu-Chen sistemi (b) 3 Boyutlu delta robot Lu-Chen sistemi



Şekil 5.33. (a) Lu-Chen sistemi x-y faz portresi (b) Delta robot Lu-Chen sistemi servo x-y faz portresi

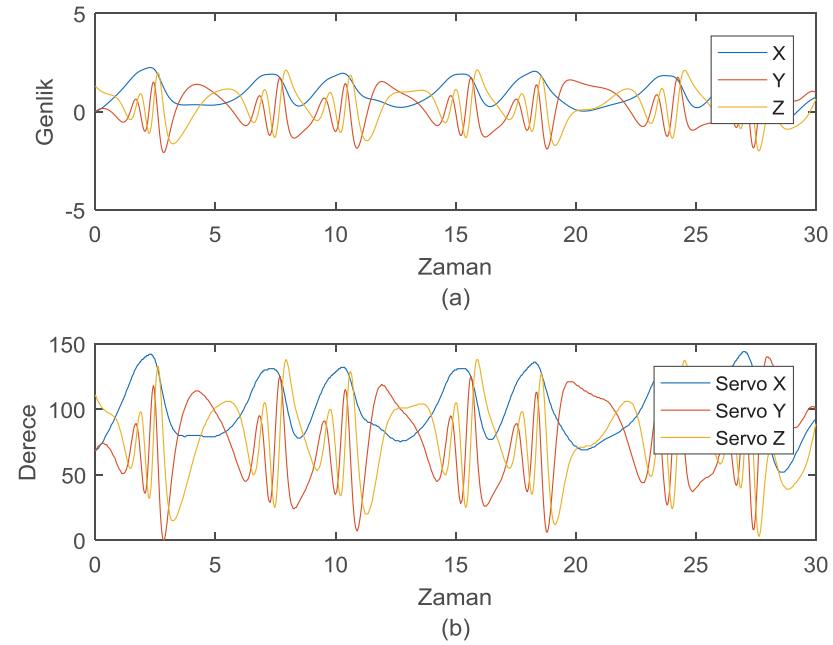


Şekil 5.34. (a) Lu-Chen sistemi x-z faz portresi (b) Delta robot Lu-Chen sistemi servo x-z faz portresi

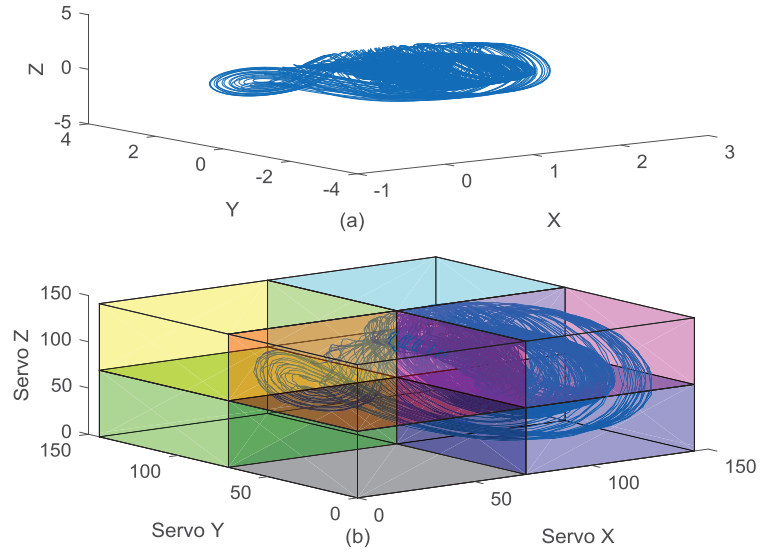


Şekil 5.35. (a) Lu-Chen sistemi y-z faz portresi (b) Delta robot Lu-Chen sistemi servo y-z faz portresi

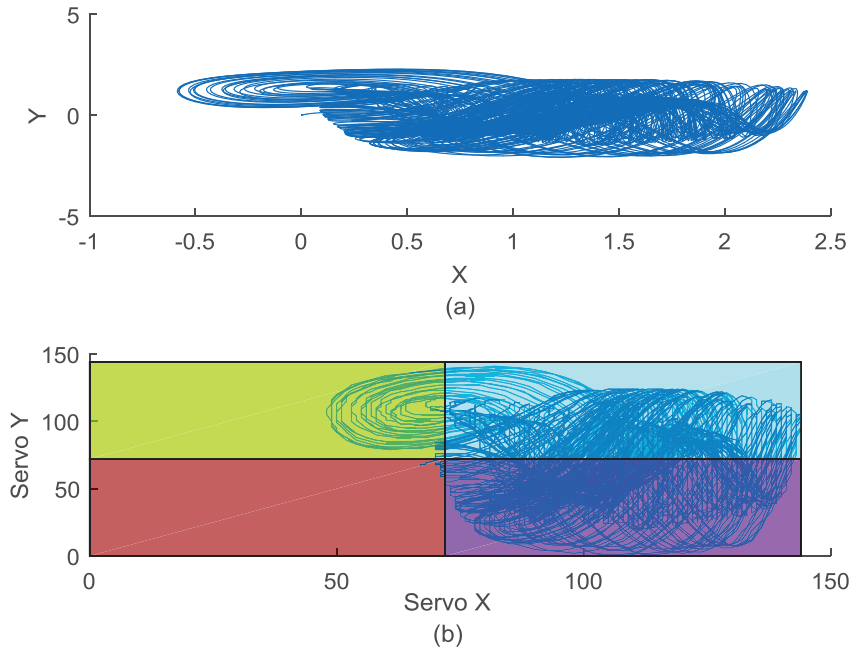
5.3.6. Hadley sistemi MATLAB karşılaştırmaları



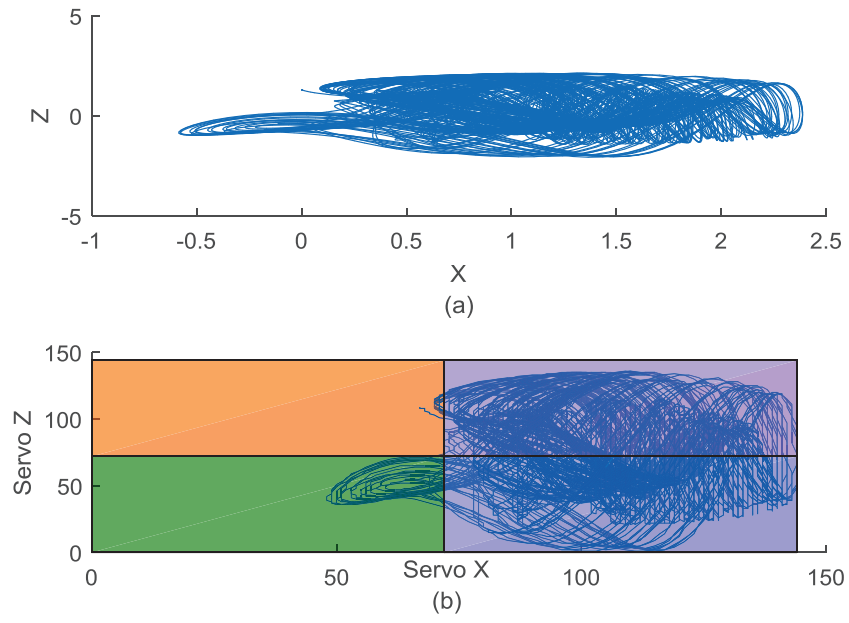
Şekil 5.36. (a) Hadley sistemi x-y-z grafiği (b) Delta robot Hadley sistemi servo x-y-z grafiği



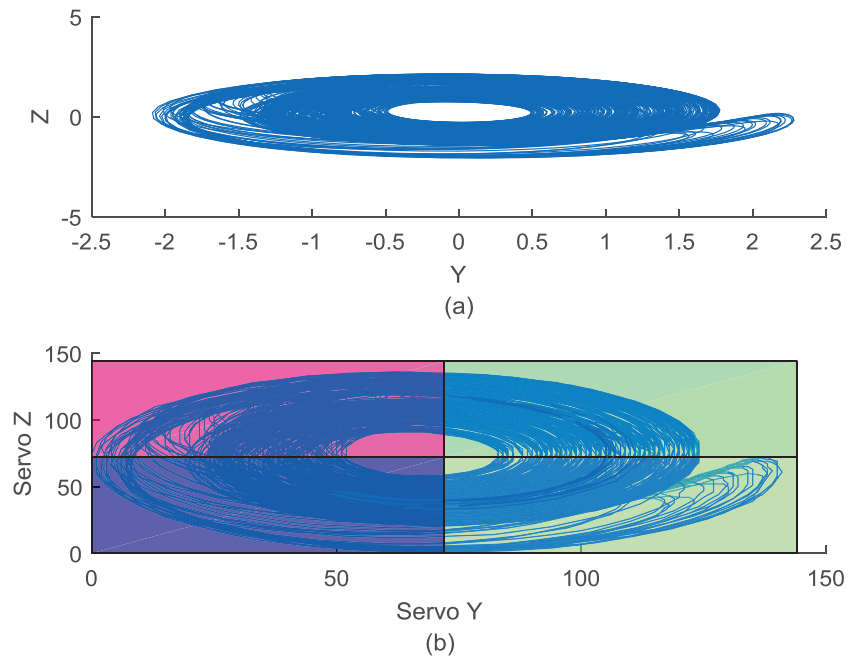
Şekil 5.37. (a) 3 Boyutlu Hadley sistemi (b) 3 Boyutlu delta robot Hadley sistemi



Şekil 5.38. (a) Hadley sistemi x-y faz portresi (b) Delta robot Hadley sistemi servo x-y faz portresi

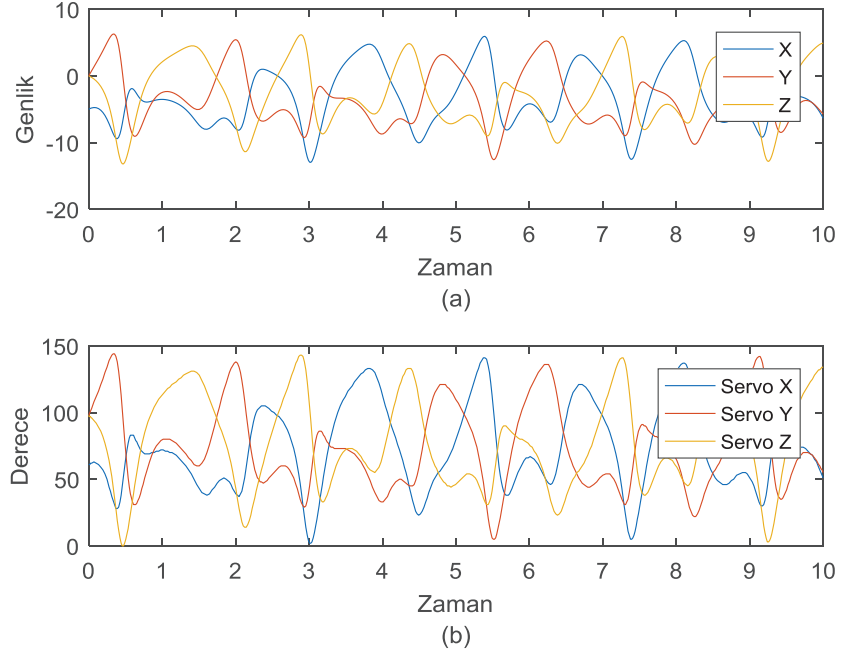


Şekil 5.39. (a) Hadley sistemi x-z faz portresi (b) Delta robot Hadley sistemi servo x-z faz portresi

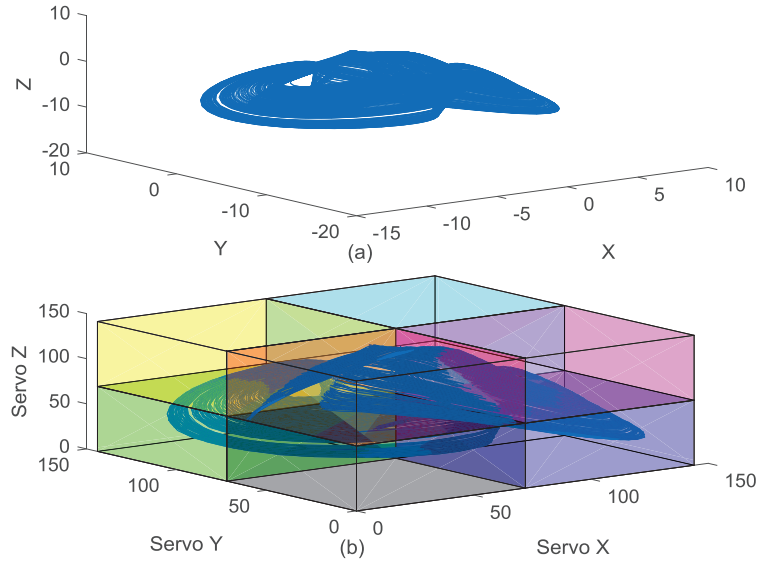


Şekil 5.40. (a) Hadley sistemi y-z faz portresi (b) Delta robot Hadley sistemi servo y-z faz portresi

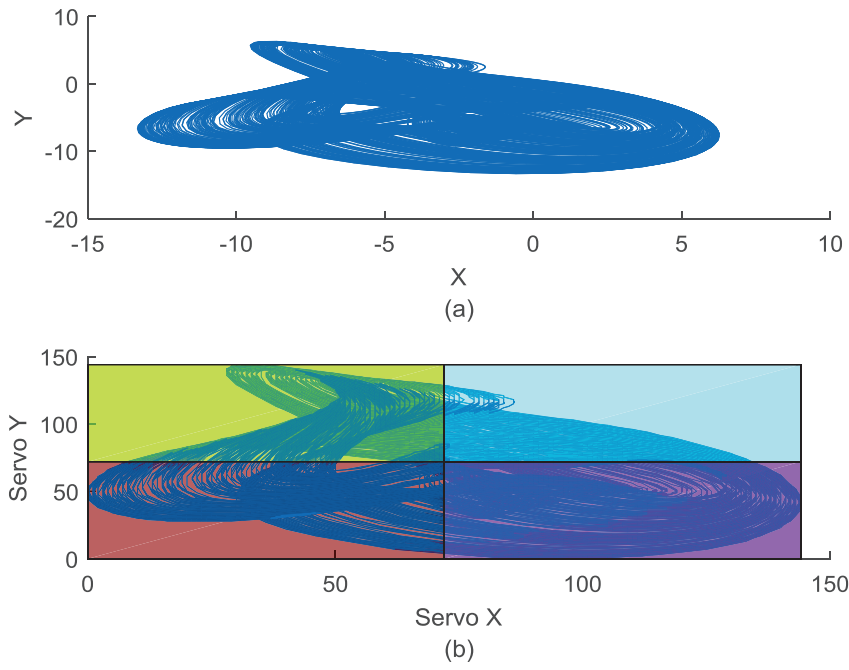
5.3.7. Halvorsen sistemi MATLAB karşılaştırmaları



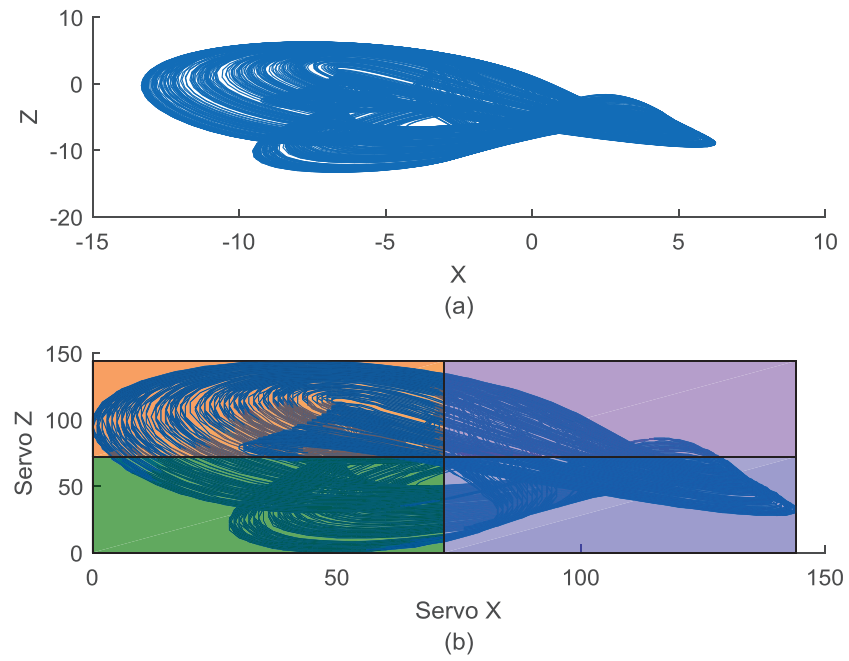
Şekil 5.41. (a) Halvorsen sistemi x-y-z grafiği (b) Delta robot Halvorsen sistemi servo x-y-z grafiği



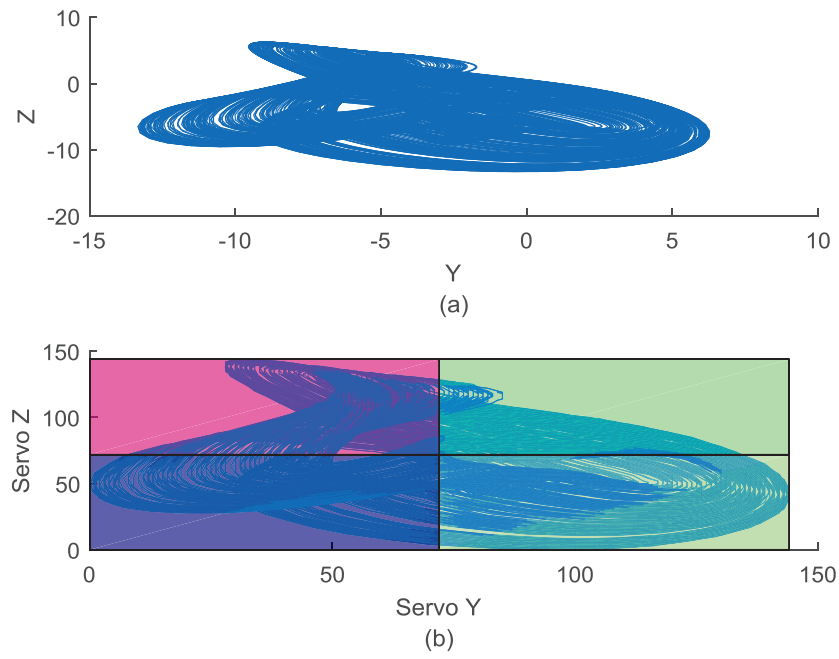
Şekil 5.42. (a) 3 Boyutlu Halvorsen sistemi (b) 3 Boyutlu delta robot Halvorsen sistemi



Şekil 5.43. (a) Halvorsen sistemi x-y faz portresi (b) Delta robot Halvorsen sistemi servo x-y faz portresi



Şekil 5.44. (a) Halvorsen sistemi x-z faz portresi (b) Delta robot Halvorsen sistemi servo x-z faz portresi



Şekil 5.45. (a) Halvorsen sistemi $y-z$ faz portresi (b) Delta robot Halvorsen sistemi servo $y-z$ faz portresi

5.4. Karıştırma Tabloları

Yapılan her bir deneyden elde edilen YDO ve HO değerleri her bir sistem için ayrı ayrı Tablo 5.1. - 5.17. arasında işlenmiştir. Tablo 5.18.'de ise önceki tablolar göz önüne alınıp YDO değerlerine göre sistemlere puanlar verilerek sıralanmıştır. Tablo 5.19.'da ise şeker-su karışımının HO oranlarına göre sistemlere puanlar verilerek sıralanmıştır.

Tablo 5.1. Sabit karıştırıcının şekerli su ölçüm verileri

Süre	Ölçülen Şeker	Homojenlik
	Oranı	Oranı
00:10	0,4	16,0
00:30	0,7	28,0
01:00	0,9	36,0
02:00	1,1	44,0
03:00	1,2	48,0
04:00	1,3	52,0
05:00	1,4	56,0
Ortalama	1,0	40,0

* Şeker-Su Karışımında 19 500 gram su ile 500 gram şeker kullanıldı. Bu sebeple en fazla Ölçülen Şeker Oranı: $500 / (19\ 500 + 500) = 0,025$ (Yani %2,5) olur.

Tablo 5.2. Dairesel hareketli karıştırıcının şekerli su yörünge verileri

Süre	1.	2.	3.	4.	5.	6.	7.	8.	Toplam
	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	
00:10	0	0	0	0	32	31	32	31	126
00:30	0	0	0	0	88	88	92	95	363
01:00	0	0	0	0	179	176	184	183	722
02:00	0	0	0	0	360	360	360	365	1445
03:00	0	0	0	0	544	544	544	544	2176
04:00	0	0	0	0	720	720	720	724	2884
05:00	0	0	0	0	897	896	904	903	3600
Ortalama	0	0	0	0	402,86	402,14	405,14	406,43	1616,57

Tablo 5.3. Dairesel hareketli karıştırıcının şekerli su ölçüm verileri

Süre	% 1. Bölge	% 2. Bölge	% 3. Bölge	% 4. Bölge	% 5. Bölge	% 6. Bölge	% 7. Bölge	% 8. Bölge	% Toplam	Yörüngesel Dağılım Oranı	Ölçülen Şeker Oranı	Homojenlik Oranı
00:10	0	0	0	0	25,40	24,60	25,40	24,60	100,0	42,9	0,3	12,0
00:30	0	0	0	0	24,20	24,20	25,30	26,30	100,0	42,9	0,7	28,0
01:00	0	0	0	0	24,80	24,40	25,50	25,30	100,0	42,9	0,8	32,0
02:00	0	0	0	0	24,90	24,90	24,90	25,30	100,0	42,9	1,2	48,0
03:00	0	0	0	0	25,00	25,00	25,00	25,00	100,0	42,9	1,4	56,0
04:00	0	0	0	0	24,98	24,98	24,98	25,06	100,0	42,9	1,5	60,0
05:00	0	0	0	0	24,90	24,90	25,10	25,10	100,0	42,9	1,7	68,0
Ortalama	0	0	0	0	24,80	24,80	25,20	25,20	100,0	42,9	1,1	43,4

Tablo 5.4. Sprot A sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri

Süre	1. Bölge	2. Bölge	3. Bölge	4. Bölge	5. Bölge	6. Bölge	7. Bölge	8. Bölge	Toplam
00:10	13	0	18	6	4	10	0	34	85
00:30	26	34	33	54	39	24	41	49	300
01:00	71	79	84	96	75	63	79	80	627
02:00	171	154	189	173	146	143	147	160	1283
03:00	258	260	284	267	206	213	211	244	1943
04:00	345	364	364	367	281	275	292	312	2600
05:00	429	461	456	470	357	350	372	380	3275
Ortalama	187,57	193,14	204	204,71	158,29	154	163,14	179,86	1444,71

Tablo 5.5. Sprot A sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri

Süre	% 1. Bölge	% 2. Bölge	% 3. Bölge	% 4. Bölge	% 5. Bölge	% 6. Bölge	% 7. Bölge	% 8. Bölge	% Toplam	Yörüngesel Dağılım Oranı	Ölçülen Şeker Oranı	Homojenlik Oranı
00:10	15,3	0	21,2	7,1	4,7	11,8	0	39,9	100,0	55,5	0,3	12,0
00:30	8,7	11,3	11,0	18,0	13,0	8,0	13,7	16,3	100,0	87,4	0,7	28,0
01:00	11,3	12,6	13,4	15,3	12,0	10,0	12,6	12,8	100,0	95,2	1,1	44,0
02:00	13,3	12,0	14,7	13,5	11,4	11,1	11,5	12,5	100,0	95,4	1,3	52,0
03:00	13,3	13,4	14,6	13,7	10,6	11,0	10,9	12,5	100,0	94,3	1,4	56,0
04:00	13,3	14,0	14,0	14,1	10,8	10,6	11,2	12,0	100,0	93,8	1,7	68,0
05:00	13,1	14,1	13,9	14,4	10,9	10,7	11,4	11,5	100,0	93,7	1,8	72,0
Ortalama	12,6	11,1	14,7	13,7	10,5	10,5	10,2	16,8	100,0	87,9	1,2	47,4

Tablo 5.6. Pehlivan-Wei sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri

Süre	1. Bölge	2. Bölge	3. Bölge	4. Bölge	5. Bölge	6. Bölge	7. Bölge	8. Bölge	Toplam
00:10	2	12	12	24	1	0	13	1	65
00:30	14	50	46	105	5	34	45	30	329
01:00	31	86	102	195	5	66	47	112	644
02:00	60	167	202	408	9	132	88	235	1301
03:00	89	249	284	649	15	212	148	333	1979
04:00	118	328	375	872	21	286	214	440	2654
05:00	147	422	462	1115	25	357	279	531	3338
Ortalama	65,86	187,71	211,86	481,14	11,57	155,29	119,14	240,29	1472,86

Tablo 5.7. Pehlivan-Wei sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri

Süre	% 1. Bölge	% 2. Bölge	% 3. Bölge	% 4. Bölge	% 5. Bölge	% 6. Bölge	% 7. Bölge	% 8. Bölge	% Toplam	Yörüngesel Dağılım Oranı	Ölçülen Şeker Oranı	Homojenlik Oranı
	00:10	3,1	18,5	18,5	36,9	1,5	0,0	20,0	1,5	100,0	49,8	0,4
00:30	4,3	15,2	14,0	31,9	1,5	10,3	13,7	9,1	100,0	71,7	0,9	36,0
01:00	4,8	13,4	15,8	30,3	0,8	10,2	7,3	17,4	100,0	69,3	1,1	44,0
02:00	4,6	12,8	15,5	31,4	0,7	10,1	6,8	18,1	100,0	68,2	1,2	48,0
03:00	4,5	12,6	14,4	32,8	0,8	10,7	7,5	16,7	100,0	69,7	1,3	52,0
04:00	4,4	12,4	14,1	32,9	0,8	10,8	8,1	16,5	100,0	70,3	1,4	56,0
05:00	4,4	12,6	13,8	33,4	0,7	10,7	8,4	16,0	100,0	70,5	1,5	60,0
Ortalama	4,3	13,9	15,2	32,8	1,0	9,0	10,3	13,6	100,0	67,1	1,1	44,6

Tablo 5.8. Aizawa sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri

Süre	1. Bölge	2. Bölge	3. Bölge	4. Bölge	5. Bölge	6. Bölge	7. Bölge	8. Bölge	Toplam
00:10	27	9	0	0	52	7	13	7	115
00:30	79	22	19	9	94	20	28	15	286
01:00	128	42	51	16	193	63	67	42	602
02:00	234	102	121	53	368	143	144	93	1258
03:00	351	162	205	81	531	221	217	144	1912
04:00	472	214	272	99	735	296	290	192	2570
05:00	604	276	347	126	897	374	362	240	3226
Ortalama	270,71	118,14	145	54,86	410	160,57	160,14	104,71	1424,14

Tablo 5.9. Aizawa sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri

Süre	% 1.	% 2.	% 3.	% 4.	% 5.	% 6.	% 7.	% 8.	%	Yörüngesel	Ölçülen	Homojenlik
	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Toplam	Dağılım Oranı	Şeker Oranı	Oranı
00:10	23,5	7,8	0	0	45,2	6,1	11,3	6,1	100,0	50,1	0,3	12,0
00:30	27,6	7,7	6,6	3,1	32,9	7,0	9,8	5,3	100,0	59,4	0,7	28,0
01:00	21,3	7,0	8,5	2,7	32,1	10,5	11,1	6,8	100,0	67,5	0,9	36,0
02:00	18,6	8,1	9,6	4,2	29,3	11,4	11,4	7,4	100,0	73,8	1,1	44,0
03:00	18,4	8,5	10,7	4,2	27,8	11,6	11,3	7,5	100,0	75,8	1,4	56,0
04:00	18,4	8,3	10,6	3,9	28,6	11,5	11,3	7,4	100,0	74,9	1,5	60,0
05:00	18,7	8,6	10,8	3,9	27,8	11,6	11,2	7,4	100,0	75,4	1,6	64,0
Ortalama	20,9	8,0	8,1	3,1	32,0	10,0	11,1	6,8	100,0	68,1	1,1	42,9

Tablo 5.10. Guckenheimer-Holmes sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri

Süre	1.	2.	3.	4.	5.	6.	7.	8.	Toplam
	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	
00:10	13	17	12	21	6	13	13	22	117
00:30	13	17	12	128	6	13	13	134	336
01:00	25	33	21	252	15	29	34	251	660
02:00	50	66	53	454	33	71	76	517	1320
03:00	70	119	96	571	64	118	134	805	1977
04:00	103	196	158	727	102	174	209	970	2639
05:00	128	241	208	826	138	251	281	1222	3295
Ortalama	57,43	98,43	80	425,57	52	95,57	108,57	560,14	1477,71

Tablo 5.11. Guckenheimer-Holmes sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri

Süre	% 1.	% 2.	% 3.	% 4.	% 5.	% 6.	% 7.	% 8.	%	Yörüngesel	Ölçülen	Homojenlik
	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Toplam	Dağılım Oranı	Şeker Oranı	Oranı
00:10	11,1	14,5	10,3	17,9	5,1	11,1	11,1	18,9	100,0	84,2	0,3	12,0
00:30	3,9	5,1	3,6	38,1	1,8	3,9	3,9	39,7	100,0	39,7	0,5	20,0
01:00	3,8	5,0	3,2	38,2	2,3	4,4	5,2	37,9	100,0	41,6	0,9	36,0
02:00	3,8	5,0	4,0	34,4	2,5	5,4	5,8	39,1	100,0	44,6	1,1	44,0
03:00	3,5	6,0	4,9	28,9	3,2	6,0	6,8	40,7	100,0	49,0	1,4	56,0
04:00	3,9	7,4	6,0	27,5	3,9	6,6	7,9	36,8	100,0	55,1	1,5	60,0
05:00	3,9	7,3	6,3	25,1	4,2	7,6	8,5	37,1	100,0	57,5	1,6	64,0
Ortalama	4,8	7,2	5,5	30,0	3,3	6,4	7,0	35,7	100,0	53,1	1,0	41,7

Tablo 5.12. Lu-Chen sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri

Süre	1. Bölge	2. Bölge	3. Bölge	4. Bölge	5. Bölge	6. Bölge	7. Bölge	8. Bölge	Toplam
00:10	0	0	0	0	71	16	6	30	123
00:30	0	0	0	0	189	55	29	93	366
01:00	0	0	0	0	315	112	65	204	696
02:00	0	0	0	0	621	215	128	401	1365
03:00	0	11	0	0	907	349	204	572	2043
04:00	0	11	0	0	1257	438	271	742	2719
05:00	0	11	0	0	1577	548	334	935	3405
Ortalama	0	4,71	0	0	705,29	247,57	148,14	425,29	1531

Tablo 5.13. Lu-Chen sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri

Süre	% 1. Bölge	% 2. Bölge	% 3. Bölge	% 4. Bölge	% 5. Bölge	% 6. Bölge	% 7. Bölge	% 8. Bölge	% Toplam	Yörüngesel Dağılım Oranı	Ölçülen Şeker Oranı	Homojenlik Oranı
00:10	0	0	0	0	57,7	13,0	4,9	24,4	100,0	34,2	0,2	8,0
00:30	0	0	0	0	51,6	15,0	7,9	25,5	100,0	37,6	0,6	24,0
01:00	0	0	0	0	45,3	16,1	9,3	29,3	100,0	39,2	0,7	28,0
02:00	0	0	0	0	45,5	15,8	9,4	29,3	100,0	39,3	1,0	40,0
03:00	0	0	0	0	44,4	17,1	10,0	28,0	100,0	40,6	1,2	48,0
04:00	0	0	0	0	46,2	16,1	10,0	27,3	100,0	40,5	1,4	56,0
05:00	0	0	0	0	46,3	16,1	9,8	27,5	100,0	40,1	1,5	60,0
Ortalama	0	0	0	0	48,1	15,6	8,8	27,3	100,0	38,8	0,9	37,7

Tablo 5.14. Hadley sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri

Süre	1. Bölge	2. Bölge	3. Bölge	4. Bölge	5. Bölge	6. Bölge	7. Bölge	8. Bölge	Toplam
00:10	0	31	0	19	21	30	1	17	119
00:30	0	77	0	65	21	92	1	88	344
01:00	0	155	14	132	21	195	3	158	678
02:00	0	305	39	265	21	390	9	318	1347
03:00	0	445	95	403	27	576	9	451	2006
04:00	0	607	119	535	27	768	15	604	2675
05:00	0	770	119	662	27	996	15	755	3344
Ortalama	0	341,43	55,14	297,29	23,57	435,29	7,57	341,57	1501,86

Tablo 5.15. Hadley sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri

Süre	% 1.	% 2.	% 3.	% 4.	% 5.	% 6.	% 7.	% 8.	%	Yörüngesel Dağılım Oranı	Ölçülen Şeker Oranı	Homojenlik Oranı
	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Toplam			
00:10	0,0	26,1	0,0	16,0	17,6	25,2	0,8	14,3	100,0	58,1	0,2	8,0
00:30	0,0	22,4	0,0	18,9	6,1	26,7	0,3	25,6	100,0	50,2	0,5	20,0
01:00	0,0	22,9	2,1	19,5	3,1	28,8	0,4	23,2	100,0	49,3	1,1	44,0
02:00	0,0	22,6	2,9	19,7	1,6	29,0	0,7	23,5	100,0	48,8	1,3	52,0
03:00	0,0	22,2	4,7	20,1	1,3	28,7	0,4	22,6	100,0	50,2	1,4	56,0
04:00	0,0	22,7	4,4	20,0	1,0	28,7	0,6	22,6	100,0	49,7	1,5	60,0
05:00	0,0	23,0	3,6	19,8	0,8	29,8	0,4	22,6	100,0	48,3	1,6	64,0
Ortalama	0,0	23,1	2,5	19,1	4,5	28,1	0,5	22,1	100,0	50,6	1,1	43,4

Tablo 5.16. Halvorsen sistemi tabanlı kaotik karıştırıcının şekerli su yörünge verileri

Süre	1.	2.	3.	4.	5.	6.	7.	8.	Toplam
	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	
00:10	1	15	26	2	16	1	51	4	116
00:30	1	61	72	25	69	35	69	7	339
01:00	2	138	143	67	134	81	101	10	676
02:00	6	270	284	125	279	153	194	27	1338
03:00	9	408	421	213	425	220	271	37	2004
04:00	12	551	563	299	562	281	354	45	2667
05:00	15	689	708	367	703	386	417	52	3337
Ortalama	6,57	304,57	316,71	156,86	312,57	165,29	208,14	26	1496,71

Tablo 5.17. Halvorsen sistemi tabanlı kaotik karıştırıcının şekerli su ölçüm verileri

Süre	% 1.	% 2.	% 3.	% 4.	% 5.	% 6.	% 7.	% 8.	%	Yörüngesel Dağılım Oranı	Ölçülen Şeker Oranı	Homojenlik Oranı
	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Bölge	Toplam			
00:10	0,9	12,9	22,4	1,7	13,8	0,9	44,0	3,4	100,0	50,7	0,2	8,0
00:30	0,3	18,0	21,2	7,4	20,4	10,3	20,4	2,0	100,0	65,7	0,6	24,0
01:00	0,3	20,4	21,2	9,9	19,8	12,0	14,9	1,5	100,0	69,9	0,7	28,0
02:00	0,4	20,2	21,2	9,3	20,9	11,4	14,5	2,1	100,0	69,4	0,9	36,0
03:00	0,4	20,4	21,0	10,6	21,2	11,0	13,5	1,9	100,0	70,2	1,1	44,0
04:00	0,4	20,7	21,1	11,2	21,1	10,5	13,3	1,7	100,0	70,1	1,3	52,0
05:00	0,4	20,6	21,2	11,0	21,1	11,6	12,5	1,6	100,0	71,0	1,4	56,0
Ortalama	0,4	19,0	21,3	8,7	19,8	9,7	19,0	2,0	100,0	66,7	0,9	35,4

Tablo 5.18. Sistemlerin yörüngesel dağılım oranlarına göre karşılaştırılması

Sistemler	Çalışma Süreleri							Toplam
	00:10	00:30	01:00	02:00	03:00	04:00	05:00	Puan
Sabit	1	1	1	1	1	1	1	7
Dairesel Hareket	3	4	4	3	3	3	3	23
Sprott-A	7	9	9	9	9	9	9	61
Pehlivan-Wei	4	8	7	6	6	7	6	44
Aizawa	5	6	6	8	8	8	8	49
Guckenheimer-Holmes	9	3	3	4	4	5	5	33
Lu-Chen	2	2	2	2	2	2	2	14
Hadley	8	5	5	5	5	4	4	36
Halvorsen	6	7	8	7	7	6	7	48

Tablo 5.19. Sistemlerin şeker-su karışımının homojenlik oranlarına göre karşılaştırılması

Sistemler	Çalışma Süreleri							Toplam
	00:10	00:30	01:00	02:00	03:00	04:00	05:00	Puan
Sabit	3	3	3	3	2	1	1	16
Dairesel Hareket	2	3	2	4	4	3	4	22
Sprott-A	2	3	5	5	4	5	5	29
Pehlivan-Wei	3	5	5	4	3	2	2	24
Aizawa	2	3	3	3	4	3	3	21
Guckenheimer-Holmes	2	1	3	3	4	3	3	19
Lu-Chen	1	2	1	2	2	2	2	12
Hadley	1	1	5	5	4	3	3	22
Halvorsen	1	2	1	1	1	1	1	8

BÖLÜM 6. SONUÇLAR VE ÖNERİLER

Bu çalışmada Arduino ve MATLAB tabanlı yeni bir Kaotik karıştırıcı tasarlanarak katı-sıvı karışımı üzerinde farklı kaotik sistemler ile uygulamaları gerçekleştirilmiştir.

Bu kapsamda öncelikle, literatür taraması yapılarak kaotik karıştırma işlemi için farklı kaotik denklem sistemlerinin araştırılması, birbirleriyle karşılaştırılması, uygun olanların tespit edilmesi işlemleri yapılmıştır. Bu adımdan sonra Sprott A, Pehlivan-Wei, Aizawa, Guckenheimer-Holmes, Lu-Chen, Hadley ve Halvorsen sistemleri seçilmiştir.

Daha sonra tüm tasarımları özgün olan, 3 Eksenli, 3 Servo Motorlu, Motor Sürücü kartlı, Ardunio Uno işlemcisi içeren, 30x30x80 cm boyutlarında, 2 kg ağırlığında olan standart Delta robotun benzeri bir robot geliştirilmiştir. Sonraki aşamada, MATLAB ortamında seçilen kaotik sistemler RK45 sayısal çözüm algoritması ile çözdürülerek, integer formatındaki sonuçlar USB kablosu aracılığı Arduino Uno işlemcisi üzerine aktarılmıştır. Arduino bu verileri delta robotun hareket açısı 0 ile 144 derece arası olduğu için skalama işlemi yaparak her bir servo motora RAD atamaktadır.

Tezde geliştirilen yazılım sayesinde; Ardunio Uno işlemcisi, RAD 'ni konum bilgisi olarak değerlendirmekte, tasarlanan delta robotun servo motorlarının X-Y-Z eksenlerinde, gelecek şekilde kontrol edilmesi sağlanmaktadır.

Son aşamada, katı-sıvı karışım tipi ve farklı kaotik sistemlerin kullanıldığı, Yörüngesel Dağılım Oranı (YDO), Homojenlik Oranı (HO) ve karıştırma süreleri parametrelerinin ölçüldüğü deneysel çalışmalar yapılmıştır. En son olarak tezdeki

tüm simülasyon ve deneysel sonuçlar tablolar halinde birbirleri ile karşılaştırılmıştır. Tablolardan görüldüğü gibi verilen YDO ve HO parametrelerine göre seçilen kaotik sistemler sabit ve dairesel hareketli karıştırıcılara göre daha yüksek performans elde etmiştir. İçlerinde en iyi performansa sahip sistemin ise Sprott A sistemi olduğu görülmüştür.

Elde edilen bu veriler ışığında genel olarak kaotik sistemler sabit ve dairesel hareketli karıştırıcıdan iyi sonuçlar vermekle birlikte farklı karışımlar için farklı kaotik sistemler ile daha yüksek sonuçlar elde edilebilmektedir.

Deneylerde kullanılan ölçüm cihazına göre farklı karışımlarda da kaotik sistemler denenebilir. Örneğin, renk ölçüm cihazı kullanarak bir boya karışımı ile bilinen karıştırma yöntemleri ve kaotik sistemler karşılaştırılabilir.

Ayrıca gerçekleştirilen bu kaotik karıştırıcı ile sıvı-sıvı ve katı-katı maddelerin karıştırılmasına ilişkin bir dizi deneysel çalışma da yapılabilmektedir.

Tezde sunulan bu prototipin, daha kısa sürede daha homojen karışımların elde edilebilmesi için yeni kaotik karıştırıcı sistemlerin ve algoritmalarının geliştirilebilmesi amacına hizmet edeceği düşünülmektedir.

KAYNAKLAR

- [1] Kurt, E., Kalaycı, O., Pehlivan, İ., Coşkun, S., Delta Robot Tabanlı Örnek bir Kaotik Karıştırıcıda Farklı Kaotik Sistemlere ait Yörüngelerin İncelenmesi. ISITES, Bakü, 849-857, 2017.
- [2] Acartürk, F., Ağabeyoğlu, İ., Çelebi, N., Değim, T., Değim, Z., Doğanay, T., Tırnaksız, F., Modern Farmasötik Teknoloji. TEB Eczacılık Akademisi, 10-11, 2009.
- [3] Elüstün, H. G., Karıştırıcı Kolu Tasarımı Ve Gerilme Analizi. Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü, Makina Mühendisliği Bölümü, Yüksek Lisans Tezi, 2008.
- [4] Kalaycı, O., Kurt, E., Pehlivan, İ., Varan, M., LabVIEW ve Arduino Uno ile Bir Kaotik Karıştırıcının Kontrolü. ISITES, Bakü, 868-876, 2017.
- [5] Ören, M., Karıştırma ve Karıştırıcılar, Kullanım Alanları ve Hesaplama Yöntemleri. Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Makina Mühendisliği Bölümü, Yüksek Lisans Tezi, 2006.
- [6] <http://hbogm.meb.gov.tr/modulerprogramlar/kursprogramlari/kimya/modullerKaristiriciKullanma.pdf>, Erişim Tarihi: 27.10.2016.
- [7] Pehlivan, İ., Yeni Kaotik Sistemler: Elektronik Devre Gerçeklemeleri, Senkronizasyon ve Güvenli Haberleşme Uygulamaları. Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Bölümü, Doktora Tezi, 2007.
- [8] Lorenz, E. N., Deterministic Nonperiodic Flow. J. Atmos. Sci., 20: 130-141, 1963.
- [9] Rössler, O. E., An equation for continuous chaos. Phys. Lett. A, 57(5):397-398, 1976.
- [10] Lakshmanan, M., Murali, K., Chaos in Nonlinear Oscillators, Controlling and Synchronization. World Scientific, 1996.

- [11] Chua, L. O., Wu, C. W., Huang, A., Zhong, G., Universal Circuit for Studying and Generating Chaos-Part I: Routes to Chaos. *IEEE Trans. Circuits & Systems-I*, 40:732-761, 1993.
- [12] Pehlivan, İ., Akgül, A., Kurt, E., Başaran, A., 2-Sarmallı Skala Edilmiş Lu-Chen 2003 Kaotik Sistemi ve Elektronik Osilatörü. *ISITES*, Antalya, 1122-1134, 2016.
- [13] Neugebauer, R., Wittstock, V., Drossel, W., *Werkzeugmaschinen Mechatronik*. Chemnitz, 2010.
- [14] Özkılıç, E., 3 Serbestlik Dereceli Delta Robot Mekanizmasının Matlab-Simmechanics İle Parametrik Olarak Simulasyonu. Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Makina Mühendisliği Bölümü, Bitirme Tezi, 2012.
- [15] Banhero, J. T., Bodger, W. L., *Kimya Mühendisliğine Giriş*, 2. Cilt. İTÜ yayınları, 731 - 737, 1979.
- [16] Biesecker, B. O., *Begasen von Flüssigkeiten mit Rührern*. VDI-Forschungsheft 554, 1-32, 1972.
- [17] Vauck, W., Müller, H., *Kimyasal Mühendisliğin Temel İşlemleri Giriş*. 2. Cilt. Theodor Steinkopff, 307-328, 1966.
- [18] Henzler H. J., *Eignung von kontinuierlich durchströmten Mischern zum Homogenisieren*. WILEY-VCH yayınları, 1-8, 1972.
- [19] İlten, N., *Karıştırıcılar ve karıştırma kabındaki akım olaylarının incelenmesi*. Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Makina Mühendisliği Bölümü, Yüksek Lisans Tezi, 1986.
- [20] Can, B. A., *Karışma kabı içerisinde farklı akışkanların karıştırılması ve sıcaklık dağılımının analizi*. Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Makina Mühendisliği Bölümü, Yüksek Lisans Tezi, 2010.
- [21] Lorenz, E. N., *Deterministic nonperiodic flow*. *J. Atmos. Sci.*, 20:130–141, 1963.
- [22] Li, T., Yorke, J. A., *Period three implies chaos*. *Amer. Math. Monthly*, 82:985-992, 1975.
- [23] Kennedy, M. P., *Experimental Chaos from Autonomous Electronic Circuits*. *Phil. Trans. R. Soc.*, A(353):13-32, 1995.
- [24] Kennedy, M. P., Chua, L. O., *Van Der Pol and Chaos*. *IEEE Trans. Circuit Syst.*, CAS-33:974-980, 1986.

- [25] Lakshmanan, M., Murali K., Chaos in Nonlinear Oscillators, Controlling and Synchronization. World Scientific, 1996.
- [26] Angeles, J., Fundamentals of Robotic Mechanical Systems: Theory, Methods and Algorithms. Springer, 1997.
- [27] Zsombo-Murray, P. J., McGill University, Department of Mechanical Engineering, Centre for Intelligent Machines, 2004.
- [28] Şanlıtürk, İ. H., Görerek İşlem Yapabilen Delta Robotun Tasarımı ve Performans Karakteristiklerinin Araştırılması. Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Makine Mühendisliği Bölümü, Yüksek Lisans Tezi, 2012.
- [29] Qiaoling, Y., Shiming, J., Zhongfei, W., Guan, W., Yuehua, W., Li, Z., Optimal Design of the Linear Delta Robot for Prescribed Cuboid Dexterous Workspace based on performance chart. 8th WSEAS Int. Conf. on Robotics, Control And Manufacturing Technology (Rocom '08), P.P 35-41, Hangzhou, 2008.
- [30] Xin-Jun, L., Jinsong, W., Kun-Ku, O., Jongwon, K., A New Approach to the Design of a DELTA Robot with a Desired Workspace. Journal of Intelligent and Robotic Systems, Kluwer Academic Publishers, 39: 209–225, 2004.
- [31] Chau, K. T., Shuang, Y., Yuan, G., Chen, J. H., Application of Chaotic-Motion Motors to Industrial Mixing Processes. IAS2004, 1874-1880, 2004.
- [32] Ye, S., Chau, K. T., Destabilization Control of a Chaotic Motor for Industrial Mixers. IAS2005, 1724-1730, 2005.
- [33] Ye, S., Chau, K. T., Chaoization of DC Motors for Industrial Mixing. IEEE Transactions on Industrial Electronics, Vol. 54, 4, 2007.
- [34] Murtadha, M. A., Abdurrahman, M., Korman, A. I., Chaotic Control of Liquid Mixer. University of Sharjah, Department Of Electrical & Computer Engineering. Senior Design Project II, 2008.
- [35] Zhang, Z., Chen, G., Liquid mixing enhancement by chaotic perturbations in stirred tanks. ScienceDirect, Chaos, Solitons and Fractals 36, 144–149, 2008.
- [36] Kavur, A. E., Demiroğlu, S., Seydibeyoğlu, M. Ö., Baser, Ö., Güzeliş, C., Şahin, S., Design and Implementation of Chaotic System Based Robust Delta Robot for Blending Graphene Nanoplatelets. IEEE, 2016.
- [37] http://www.robotiksistem.com/dc_motor_ozellikleri.html, Erişim Tarihi: 27.10.2016.

- [38] <https://gelecegiyazanlar.turkcell.com.tr/konu/arduino/egitim/arduino-301/servo-motor,>. Erişim Tarihi: 15.10.2016.
- [39] <http://maker.robotistan.com/arduino-dersleri-13-servo-motor-kontrolu,>. Erişim Tarihi: 15.10.2016.
- [40] <http://diyot.net/arduino-uno-r3,>. Erişim Tarihi: 20.10.2016.
- [41] <https://urun.n11.com/diger-aksesuar-yedek-parca/pwm-ile-dc-motor-kontrol-demonte-P166817324,>. Erişim Tarihi: 12.11.2016.
- [42] <http://www.milwaukeeinst.com/site/products/products/digital-refractometers/165-products-g-digital-refractometers-g-ma871,>. Erişim Tarihi: 07.07.2017.

ÖZGEÇMİŞ

Ersin Kurt, 02.04.1987'de Bursa'da doğdu. İlk, orta, lise ve ön lisans eğitimini Bursa'da tamamladı. 2005 yılında Hürriyet Teknik Lisesi, Elektronik Bölümü'nden, 2007 yılında da Uludağ Üniversitesi, Endüstriyel Elektronik Bölümü'nden mezun oldu. 2008 yılında başladığı Sakarya Üniversitesi Elektronik Öğretmenliği Bölümü'nü 2012 yılında bitirdi. 2015 yılında Sakarya Üniversitesi Mekatronik Mühendisliği Bölümü'nde yüksek lisans eğitimine başladı.