



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ÇOK SERBESTLİK DERECELİ BİR
HİDROLİK ROBOT MANİPÜLATÖRÜN
DENEYSEL MODELLENMESİ VE OPTİMAL
KONTROLÜ

Hasan Basri ÖKSÜZ

YÜKSEK LİSANS

Elektrik Elektronik Mühendisliği
Anabilim Dalı

Ekim-2019
KONYA
Her Hakkı Saklıdır
TEZ KABUL VE ONAYI

Hasan Basri ÖKSÜZ tarafından hazırlanan “Çok Serbestlik Dereceli Bir Hidrolik Robot Manipülatörün Deneysel Modellenmesi ve Optimal Kontrolü” adlı tez çalışması 18/10/2019 tarihinde aşağıdaki jüri tarafından oy birliği / oy çokluğu ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Elektrik Elektronik Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS olarak kabul edilmiştir.

Jüri Üyeleri

Başkan

Doç. Dr. Ömer AYDOĞDU

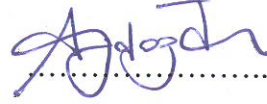
Üye

Dr. Öğr. Üyesi Burak YILMAZ

Danışman

Prof. Dr. Seral ÖZŞEN

İmza



Yukarıdaki sonucu onaylarım.

Prof. Dr. Hakan KARABÖRK
FBE Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.


Hasan Basri ÖKSÜZ

Tarih: 28.11.2019

ÖZET

YÜKSEK LİSANS

ÇOK SERBESTLİK DERECELİ BİR HİDROLİK ROBOT MANİPÜLATÖRÜN DENEYSEL MODELLENMESİ VE OPTİMAL KONTROLÜ

Hasan Basri ÖKSÜZ

**Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü
Elektrik Elektronik Mühendisliği Anabilim Dalı**

**Danışman: Prof. Dr. Seral ÖZŞEN
İkinci Danışman: Prof. Dr. Mete KALYONCU
2019, 112 Sayfa**

Jüri

**Dr. Öğr. Üyesi Burak YILMAZ
Doç. Dr. Ömer AYDOĞDU
Prof. Dr. Seral ÖZŞEN**

Bu çalışmada, çok serbestlik dereceli bir hidrolik robot manipülatörün deneysel modellenmesi ve optimal kontrolü gerçekleştirilmiştir. Hidrolik robot manipulator kaldırma-indirme ve uzatma-çekme olmak üzere iki tür hareket tipine sahiptir. Bu hareketler toplamda on dört farklı şekilde olmak üzere yedi adet tahrik elemanı ile tahrik edilmektedir. Bu on dört farklı hareketin deneysel modellenmesi yapılmış ve transfer fonksiyonları elde edilmiştir. Elde edilen transfer fonksiyonları kullanılarak sistemin optimal kontrolü için PID kontrolcü tasarımları yapılmış ve PID kontrolcü kazançları ayarlanmıştır. On dört farklı harekete ait transfer fonksiyonu deneysel olarak elde edilen veriler kullanılarak MATLAB System Identification Toolbox ile tanımlanmıştır. Elde edilen transfer fonksiyonları kullanılarak Genetik Algoritma, Parçacık Sürü Optimizasyonu, Arı Algoritması, Guguk Kuşu Algoritması ve Büyük Patlama Büyük Çöküş optimizasyon yöntemleri aracılığıyla sistemin kapalı çevrim basamak cevabına göre özgün olarak belirlenen amaç fonksiyonunu minimize edilmiştir. Bu çalışma kapsamında Büyük Patlama Büyük Çöküş ve Hooke Jeeves algoritmasının birlikte kullanıldığı hibrit bir optimizasyon algoritması önerilmiştir. Önerilen algoritma ile diğer optimizasyon algoritmaları karşılaştırılmış ve elde edilen sonuçlar grafiksel olarak sunulmuştur.

Anahtar Kelimeler: Deneysel modelleme, Sistem tanılama, Hidrolik robot manipülatör, PID kontrolcü tasarımı, Optimizasyon, Sezgisel Algoritmalar.

ABSTRACT

MS THESIS

EXPERIMENTAL MODELING AND OPTIMAL CONTROL OF A MULTI-DEGREE OF FREEDOM HYDRAULIC ROBOT MANIPULATOR

Hasan Basri ÖKSÜZ

**Konya Technical University
Institute of Graduate Studies
Department of Electrical Electronic Engineering**

**Advisor: Prof. Dr. Seral ÖZŞEN
Second Advisor: Prof. Dr. Mete KALYONCU**

2019, 112 Pages

**Jury
Assist. Prof. Üyesi Burak YILMAZ
Assoc. Prof. Ömer AYDOĞDU
Prof. Dr. Seral ÖZŞEN**

In this study, experimental modeling and optimal control of a multi degree of freedom hydraulic robot manipulator has been realized. The hydraulic robot manipulator has two types of movement: lifting-lowering and extension-pulling. These movements fourteen in total are driven by seven drive elements. Experimental modeling of these fourteen different movements were performed and transfer functions were obtained. PID controller designs were made for optimal control of the system using the transfer functions obtained and PID controller gains were tuned. The transfer function of fourteen different movements is defined by MATLAB System Identification Toolbox using the data obtained experimentally. Using the transfer functions obtained, Genetic Algorithm, Particle Swarm Optimization, The Bees Algorithm, Cuckoo Bird Algorithm and Big Bang Big Crunch optimization methods have been minimized the cost function of the system, which was originally determined according to the closed loop step response. In this study, a hybrid algorithm using Big Bang Big Crunch and Hooke Jeeves algorithm is proposed. The proposed algorithm and other optimization algorithms were compared and the results obtained were presented graphically.

Keywords: Experimental modeling, System identification, Hydraulic robot manipulator, PID controller design, Optimization, Heuristic Algorithms.

ÖNSÖZ

Endüstride çokça kullanılan PID kontrolcü etkili bir biçimde kullanılamamaktadır. PID kontrolcünün etkili bir biçimde kullanılamaması, gereğinden fazla güç tüketilmesine ve stabil olmayan hareketlere neden olur. Gereğinden fazla güç tüketilmesini engellemek ve daha stabil hareket sağlamak için PID katsayılarının optimize edilmesi büyük önem arz etmektedir. Bu optimizasyon işlemi için çeşitli sezgisel algoritmalar kullanılmış ve Hooke Jeeves algoritması ile Büyük Patlama Büyük Çöküş Algoritmasının hibrit edilmesi önerilmiş ve başarı ile kullanılmıştır.

Tez çalışmamda emekleri geçen ekip arkadaşlarım; MPG A.Ş. personeli Abdullah ERDEMİR, Mustafa KOÇAK, Salih KAYACAN, Mehmet DİNÇ, Bülent GENÇER, Nuri ALAŞAHİN, Fatih DAVŞANCI, Recep GÜLER, değerli hocalarım Prof. Dr. Seral ÖZŞEN, Prof. Dr. Mete KALYONCU ve Dr. Öğr. Üyesi Burak YILMAZ'a, MPG A.Ş. Genel Müdürü Veysel ALVER'e ve her an desteğini esirgemeyen sevgili eşim Fatma ÖKSÜZ'e ve aileme teşekkür ederim.

Hasan Basri ÖKSÜZ
KONYA-2019

İÇİNDEKİLER

ÖZET	iv
ABSTRACT	v
ÖNSÖZ	vi
İÇİNDEKİLER	vii
SİMGELER VE KISALTMALAR	ix
1. GİRİŞ	1
1.1. Tezin Amacı	2
1.2. Tezin Önemi.....	2
2. KAYNAK ARAŞTIRMASI	5
3. MATERYAL VE YÖNTEM	19
3.1. PID Kontrolcü ve Ziegler Nichols Yaklaşımı.....	19
3.2. Deneysel Modelleme.....	22
3.3. Sezgisel Arama Algoritmaları.....	31
3.3.1. Genetik algoritma	32
3.3.2. Parçacık sürü optimizasyonu	33
3.3.3. Arı algoritması.....	35
3.3.4. Büyük patlama büyük çöküş algoritması	37
3.3.5. Guguk kuşu algoritması.....	39
3.4. Yerel Arama Algoritmaları	43
3.4.1. Tepe tırmanma algoritması.....	43
3.4.2. Hooke Jeeves algoritması	44
3.5. Bu Çalışmada Önerilen Hibrit Algoritma	45
4. HİDROLİK ROBOT MANİPÜLATÖR YAPISI	48
5. ARAŞTIRMA BULGULARI VE TARTIŞMA	51
5.1. Problemin Tanımı	51
5.2. Deneysel Modelleme ile Elde Edilen Sonuçlar.....	52
5.3. Kullanılan Amaç Fonksiyonu	62
5.4. Algoritmalarda Seçilen Katsayılar	62
5.5. Sayısal Sonuçlar	64
5.6. Ana Bom Hareketi Elde Edilen Sonuçlar	64
5.6.1. Ana bom kaldırma hareketi	64
5.6.2. Ana bom indirme hareketi	67
5.7. 1. Kırma Hareketi Elde Edilen Sonuçlar.....	69
5.7.1. 1. kırma kaldırma hareketi.....	69
5.7.2. 1. kırma indirme hareketi.....	71
5.7.3. 1. kırma bom uzatma hareketi	73

5.7.4. 1. kırma bom çekme hareketi	75
5.8. 2. Kırma Hareketi Elde Edilen Sonuçlar.....	76
5.8.1. 2. kırma kaldırma hareketi.....	76
5.8.2. 2. kırma indirme hareketi.....	78
5.8.3. 2. kırma bom uzatma hareketi	80
5.8.4. 2. kırma bom çekme hareketi	82
5.9. Kule Dönüş Hareketi Elde Edilen Sonuçlar.....	84
5.9.1. Kule sola dönüş hareketi.....	84
5.9.2. Kule sağa dönüş hareketi.....	86
5.10. Ataçman Dönüş Hareketi Elde Edilen Sonuçlar.....	88
5.10.1. Ataçman sola dönüş hareketi.....	88
5.10.2. Ataçman sağa dönüş hareketi	90
5.11. Tüm Hareketler İçin Elde Edilen Sonuçlar.....	92
6. SONUÇ VE ÖNERİLER.....	94
KAYNAKLAR	96
7. ÖZGEÇMİŞ	102



SİMGELER VE KISALTMALAR

Simgeler

c_1 ve c_2	: Hız katsayıları
ess	: Sürekli durum hatası
f_i	: i. parçacığın amaç fonksiyonu değeri
gs	: Gözcü arı sayısı
k	: İterasyon sayısı
K_d	: Türevsel kazanç
K_i	: İntegral kazancı
K_p	: Oransal kazanç
K_{pu}	: Kritik oransal kazanç değeri
L	: Ölü zaman
M_p	: Maksimum aşma
m	: Seçilecek en iyi arı sayısı
n	: Parçacık sayısı ya da Birey sayısı
nep	: Her bir elit arı etrafına gönderilecek arı sayısı
ngh	: Komşuluk arama yarıçapı
nsp	: Elit arılardan sonraki en düşük amaç fonksiyonu değerine sahip arılar
P_C	: Çaprazlama oranı
P_{ki}	: i. iterasyona kadar küresel en iyi nokta
P_M	: Mutasyon Oranı
P_S	: Seçilim Oranı
P_u	: Kritik salınım periyodu
P_{yi}	: i. iterasyondaki yerel en iyi değer
P_{yi-1}	: i-1. iterasyondaki yerel en iyi değer
R	: Eğim
rand	: 0-1 arasında rastgele sayı
t	: Arama uzayı daraltma katsayısı
T_d	: Gecikme zamanı
T_p	: Maksimum tepe zamanı
T_r	: Yükselme zamanı
T_s	: Yerleşme zamanı
w	: Ağırlık katsayısı
w_{max}	: Ağırlık katsayısının maksimum değeri
w_{min}	: Ağırlık katsayısının minimum değeri
X_{maksKd}	: K_d için standart sapma üst sınırı
X_{maksKi}	: K_i için standart sapma üst sınırı
X_{maksKp}	: K_p için standart sapma üst sınırı
X_{minKd}	: K_d için standart sapma alt sınırı
X_{minKi}	: K_i için standart sapma alt sınırı
X_{minKp}	: K_p için standart sapma alt sınırı

Kısaltmalar

AA	: Arı Algoritması
AGA	: Adaptif Genetik Algoritma
AİH	: Ana Bom İndirme Hareketi
AKH	: Ana Bom Kaldırma Hareketi
APSO	: Adaptif Parçacık Sürü Optimizasyonu
ASADH	: Ataçman Sağa Dönüş Hareketi
ASODH	: Ataçman Sola Dönüş Hareketi
BKBÇH	: Birinci Kırma Bom Çekme Hareketi
BKKH	: Birinci Kırma Kaldırma Hareketi
BKİK	: Birinci Kırma İndirme Hareketi
BKBUH	: Birinci Kırma Bom Uzatma Hareketi
BPBÇHJ	: Büyük Patlama Büyük Çöküş ile Hooke Jeeves Algoritmasının Hibrit Edildiği Algoritma (Önerilen Algoritma)
AVR	: Otomatik Voltaj Regülatörü
BPBÇ	: Büyük Patlama Büyük Çöküş
BYAA	: Bakteriyel Yiyecek Arama Algoritması
CHR	: ChienHrones-Reswick
DE	: Diferansiyel Evrim
GA	: Genetik Algoritma
GKA	: Guguk Kuşu Algoritması
HBPBÇ	: Hibrit Büyük Patlama Büyük Çöküş
HYAKA	: Hibrit Yapay Arı Kolonisi Algoritması
HJ	: Hooke Jeeves
İKBÇ	: İkinci Kırma Bom Çekme
İKBU	: İkinci Kırma Bom Uzatma
İKİH	: İkinci Kırma İndirme Hareketi
İKKH	: İkinci Kırma Kaldırma Hareketi
KA	: Küresel Arama
KKA	: Karınca Koloni Algoritması
KSADH	: Kule Sağa Dönüş Hareketi
KSODH	: Kule Sola Dönüş Hareketi
MOL	: Basitleştirilmiş Parçacık Sürü Optimizasyonu
MSI	: MATLAB System Identification
LQR	: Doğrusal Kuadratik Regülatör
PSO	: Parçacık Sürü Optimizasyonu
TT	: Tepe Tırmanma
YAKA	: Yapay Arı Kolonisi
YA	: Yerel Arama
ZN	: Ziegler Nichols

1. GİRİŞ

İnsanlar yaşamlarını kolaylaştırmak için yeni arayışlarda bulunmaktadır. Yapıldığında bedene yorgunluk veren ya da insan sağlığını bozan işler (kimyasal enerji, nükleer enerji, çok yüksek ısı, yüksek gürültü) insanlar için büyük problemler haline gelmiştir. Bu problemleri çözmek için probleme özgün mekanik yapılar üretilmiştir. Bu mekanik yapılar ilk olarak insanların elle kontrol ettiği mekanik yapılardır. Teknolojinin gelişmesiyle bu mekanik yapılar çeşitli kontrolcülerle otomatik kontrol edilebilir hale getirilmiş ve robotlar ortaya çıkmaya başlamıştır (Çengelci ve Çimen, 2005).

Gelişen teknolojiyle birlikte robotların insan hayatındaki etkisi giderek artmaktadır. Sadece insanların yapamayacağı işler değil, insanlar tarafından yapıldığında insana yorgunluk veren işler, sıkıcı rutin işler, insan sağlığına zararlı işler robotlar aracılığıyla yapılabilir. İnsanların stres altında çalışması, zamanla yorulması hata yapma risklerini artırır. Robotlar insanların sahip olduğu duygulara sahip olmadığından stres altına girmezler ve yorulmazlar. Aynı zamanda insanlara göre çok daha hassas çalışabilirler. Bu nedenle kritik öneme sahip işler robotlara yaptırılabilir (Akkoç, 2008).

Robotların; manipülasyon, tekrar programlanabilme, algılayabilme özelliklerine sahip olması gerektiği kabul edilir. Manipülasyon, cisimler üzerinde işlem yapabilme, cisimleri bir yerden bir yere taşıma işlemidir. Tekrar programlanabilme, bir amaç için yapılan mekanik yapının amaç değiştiğinde yeni amacı gerçekleştirebilmesi için yeniden programlanmasıdır. Algılayabilme ise robot çevresindeki fiziksel olaylar veya robotun mevcut fiziksel durumu hakkındaki verilerin elde edilebilmesidir (Beykont, 2007). Bu algılayıcılara örnek olarak, açı algılayıcıları, uzunluk enkoderleri, basınç algılayıcıları, debi algılayıcıları, manyetik pusulalar verilebilir. Bu algılayıcıların verileri mikro denetleyici de değerlendirilir ve robot uzuvları bu veriler dikkate alarak hareket ettirilir (Akkoç, 2008).

Hidrolik robotlar ve makineler endüstride, inşaat sektöründe, taşıma işlerinde, madencilikte ve savunma sanayinde yüksek ağırlık taşıma kapasitesi gerektiren işlerde sıkça kullanılmaktadır (Sirouspour ve Salcudean, 2001).

Hidrolik manipülatörler yüksek güç ağırlık oranı nedeniyle genellikle ağır nesnelerin (örneğin kütüklerin) kullanıldığı işlemler veya fiziksel ortama (örneğin kazı) büyük kuvvetlerin uygulandığı işler için inşa edilirler (Mattila, Koivumaki, Caldwell, & Semini, 2017) .

1.1. Tezin Amacı

Bu çalışmada çok serbestlik dereceli robot manipülatörün deneysel modellenmesi ve optimal kontrolü amaçlanmaktadır. Her bir serbestlik derecesi kaldırma-indirme (açısal) hareketi ya da uzatma-çekme(doğrusal) hareketi yapmaktadır. Toplam 14 adet farklı hareketin optimum PID katsayılarının bulunması amaçlanmaktadır. Bu hareketlerin sistem transfer fonksiyonlarının MATLAB Sistem Identification (MSI)'da elde edilmesi amaçlanmıştır. Elde edilen transfer fonksiyonları kullanılarak kapalı çevrim basamak cevabına göre PID katsayılarının optimize edilmesi amaçlanmaktadır. Optimizasyon işlemi için Küresel Arama (KA) ve Yerel Arama (YA) algoritmaları hibrit edilerek, yeni bir arama algoritması kullanılması amaçlanmaktadır.

Bu çalışmada ulaşılması amaçlanan hedefler şunlardır:

- Sistemindeki her bir hareketin transfer fonksiyonlarının MSI'da elde edilmesi.
- Optimizasyon için özgün amaç fonksiyonunun belirlenmesi.
- KA algoritmalarından olan Parçacık Sürü Optimizasyonu (PSO), Genetik Algoritma (GA), Büyük Patlama Büyük Çöküş (BPBÇ), Guguk Kuşu Algoritması (GKA) ve Arı Algoritması (AA) ile optimum PID katsayılarının elde edilmesi.
- Önerilen hibrit algoritma aracılığıyla optimum PID katsayılarının elde edilmesi.
- KA algoritmalarında elde edilen en iyi sonuç ile önerilen hibrit algoritmanın karşılaştırılması.
- Her bir hareket için elde edilen PID katsayılarının karşılaştırılması.

1.2. Tezin Önemi

Hidrolik manipülatörlerde kontrol uzaktan kumanda aracılığıyla her bir uzvun ayrı ayrı hareket ettirilmesiyle sağlanır. Esnek yapıda olan hidrolik manipülatörlerin belirli bir hıza ulaşması için öncelikle kademeli olarak hızının artırılması gerekmektedir. Sisteme yol verildikten sonra istenilen hızda hareket sağlanabilir. Yol verilmeden yüksek hızlara çıkılması istenirse sistemde istenmeyen titreşimler meydana gelir. Esnek sistemlerde titreşimi ortadan kaldırmak için titreşimi sönmölemek ya da titreşimi oluşturan etkeni ortadan kaldırmak gerekir. Stabil olmayan hareketler yük taşıyan robotlarda büyük sorunlara neden olabilir. Operatörün sistemin ani hızlandırmasını engellemek için kontrol

(giriş) sinyali rampa sinyali haline getirilir. Böylelikle operatör basamak şeklinde giriş sinyali talep ederse sinyal rampa şekline çevrilir ve ani ivmelenmeden kaçınılır.

Robot otonom olarak hareket ettirilmek istenirse ya da robotun gitmesi gereken açı biliniyorsa PID kontrolcü veya Doğrusal Kuadratik Regülatör (LQR) gibi kontrol sistemleri robot kontrolü için kullanılabilir. Duruş rampa sinyali robota dur komutu verildiğinde ayarlanan saniye kadar (1.5-2 s) robotun hareketine devam etmesine neden olur. Böylelikle hassas kontrol operatörün tecrübesine bırakılmış olur. Rampa fonksiyonu kullanılmamış olsa ve robotun hemen durması istenirse operatörün ani hareketiyle robotta büyük titreşimler meydana gelecektir.

PID kontrolcü ve LQR'nin uygulanabilmesi için sistemin transfer fonksiyonun elde edilmiş olması gerekmektedir. Hidrolik robot üretildiğinden dolayı robottan alınan verilerle sistemin transfer fonksiyonu elde edilebilir. Üretilen sistemden deneysel metotlarla elde edilen transfer fonksiyonunun teorik yöntemle bazı varsayımlar yapılarak türetilen transfer fonksiyonundan daha gerçeğe yakın sonuç vereceği şüphesizdir. Burada sistemin durumu hakkında bilgi veren algılayıcının önemi çok büyüktür. Algılayıcı, değeri ne kadar doğruyu yansıtırsa sistem o kadar gerçeğe yakın sonuç verir. Aynı zamanda algılayıcıdan elde edilen sistem verilerinin gürültülerinin filtrelenmesi gerekmektedir. Sistem yeniden oluşturulmayacaksa üretilmiş gerçek bir sistem ise bazı varsayımlar yaparak sistemin transfer fonksiyonunu türetmek çok zaman alıcı ve gereksizdir. Üretilen sistemlerin davranışlarını incelemek için deneysel modelleme yöntemi kullanılır. Elde edilen model sadece üretilen sistem için geçerlidir. Sistem boyutları değişirse yeniden verilerin alınıp sistem modelinin çıkarılması gerekir.

PID kontrolcü en eski kontrol yöntemlerinden olmasına rağmen sanayide halen en çok kullanılan otomatik kontrol yöntemidir. PID kontrolcünü katsayılarını elde etmenin birçok yolu vardır. Bu çalışmada çeşitli optimizasyon metotlarının yanında daha önce hiç kullanılmamış yeni bir hibrit optimizasyon algoritması ile optimum PID kontrolcü katsayılarının elde edilmesi amaçlanmaktadır. PID kontrolcünün katsayılarının optimizasyonla elde edildiği çalışmalar literatürde mevcuttur. Fakat PID katsayılarının KA ve YA algoritmalarının hibrit edildiği yeni optimizasyon yöntemleriyle bulunduğu çalışmalar literatürde sayılıdır.

KA algoritmaları geniş çözüm uzayını aramaları nedeniyle yerel maksimum veya minimumları bulmakta yavaşlırlar. YA algoritmaları ise yerel maksimum ve minimum değerleri bulmakta çok hızlıdır. YA ve KA algoritmaları hibrit edilirse geniş arama alanında sonuca daha hızlı bir şekilde ulaşır. GA ve Hooke Jeeves (HJ)'in ve YAKA ve

HJ'nin hibrit edildiđi optimizasyon alıřmaları mevcuttur. Yapılan literatür taraması sonucunda BPB Algoritması ile HJ Algoritması'nın hibrit edildiđi herhangi bir alıřmanın literatürde mevcut olmadığı gözlenmiştir. Bu alıřma ile literatüre kazandırılmıştır. Evrenin varoluřu teorisinden esinlenen BPB Algoritması keřif özelliđi ile ilgili katsayıları yüksek tutulduğunda küresel arama konusunda oldukça başarılıdır. HJ Algoritması literatürde oka kullanılan yerel arama algoritmalarından olan Tepe Tırmanma (TT) Algoritması'nın geliřmiş halidir. BPB algoritmasının keřif kabiliyetiyle HJ Algoritması'nın sömürü kabiliyetinin hibrit edilmesi ok etkili bir optimizasyon algoritması oluřurmaktadır.



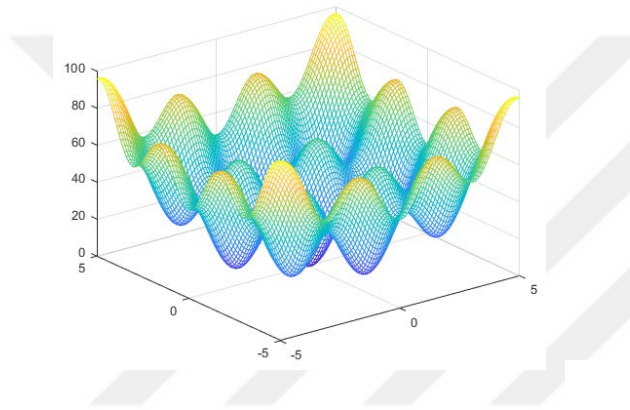
2. KAYNAK ARAŞTIRMASI

Sanayi ya da endüstri büyük oranda mekanikleşmiş ve otomatikleştirilmiş, kar amacı güden ürünler üreten kuruluşlardır. Sanayileşme kavramının çıkmasından bu yana, teknolojiye meydana gelen gelişmeler insan hayatını büyük ölçüde etkilemiştir. Sanayileşme kavramı ile birlikte insanlar, yaşam konforunu daha iyi hale getirmek için; insanlar tarafından yapıldığında yorgunluk veren işleri zamanla makinelere yaptırmaya başlamışlardır. Endüstride meydana gelen gelişmeler: mekanizasyon (1.Sanayi Devrimi), elektrik enerjisinin yoğun kullanımı (2.Sanayi devrimi), dijitalleşmenin yaygınlaşması (3.Sanayi devrimi) olarak sınıflandırılabilir (Lasi ve ark., 2014).

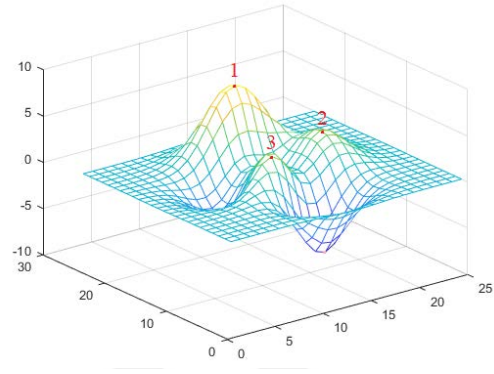
Makinelerin hayatımıza girmesiyle birlikte makine kontrolü konusunda hızla gelişmeler meydana gelmiştir. Operatör yardımıyla kontrol edilen makineler otomatik olarak kontrol edilmek istenmiştir. 1910'da bulunuşundan itibaren geniş alanlarda kullanılan PID kontrolcü bu kontrol yöntemlerinden bir tanesidir (Almabrok ve ark., 2018). PID kontrolcü, günümüzde çok daha yeni teknikler olmasına rağmen halen endüstride en çok kullanılan kontrol yöntemidir (Lin ve Liu, 2010). PID katsayılarının etkin bir şekilde elde edilip kullanılması için halen çalışmalar yapılmaktadır (Chang ve Chen, 2009) . Literatürde en çok kullanılan metotlar şu şekilde sınıflandırılabilir: i) Ziegler-Nichols (ZN) ve ChienHrones-Reswick (CHR) gibi deneysel metotlar, ii) Kök yer Eğrileri gibi analitik metotlar, iii) Sürü zekâsına dayanan iteratif geri beslemeyle ayarlama yapan optimizasyon tabanlı metotlar (Lin ve Liu, 2010).

Optimizasyon algoritmaları, tatmin edici bir çözüm buluncaya kadar elde edilen farklı çözümleri karşılaştırarak iteratif olarak gerçekleştirilen mevcut en iyiyi bulmayı amaçlayan bir yaklaşımdır. Optimizasyon metotlarının varlığına, diferansiyel hesabın gelişimi, varyasyonlar hesabı ve diğer temel yöntemlerin gelişmesiyle Newton, Langrange ve Cauchy'nin dönemlerinde rastlanır (18yy. sonu). Bu katkılara rağmen 20.yy. ortalarına kadar çok az bir gelişme meydana gelir. Optimizasyon alanında meydana gelen gelişmeler hızlı bilgisayarların icat edilmesiyle karşımıza çıkar. Mühendislik faaliyetlerindeki karmaşıklığın artmasından dolayı, bu problemleri en iyi şekilde çözmek için birçok optimizasyon metodu ortaya çıkmıştır. Bu metodları 2 farklı ana başlıkta ayırabiliriz. Bunlardan birincisi bir sonuçtan diğerine geçerken belirli kuralları olan deterministik (yerel) algoritmalar, ikincisi ise sezgisel algoritmalar (Kiryat ve Surde, 2014).

YA algoritmaları yerel maksimum veya yerel minimum değeri bulmakta çok etkilidir. Fakat bulunan her yerel maksimum veya minimum noktası küresel maksimum veya minimum noktası olmayabilir (Erdemir ve Kalyoncu, 2015). Sezgisel algoritmalar ise KA algoritmalarındandır. Fakat yerel maksimum ve minimum değeri bulmakta YA algoritmalarına göre çok daha yavaştır. Şekil 2.2’de numaralanmış 3 tepe gösterilmektedir. Bu numaralandırılmış her 3 tepe noktası da yerel maksimum noktasıdır. 1 numaralı tepe ise küresel maksimum noktasıdır. Şekil 2.1’de gösterilen Egg Crate fonksiyonu gibi çok tepeli fonksiyonlarda YA algoritmalarının en iyi sonucu bulma olasılığı çok düşüktür.



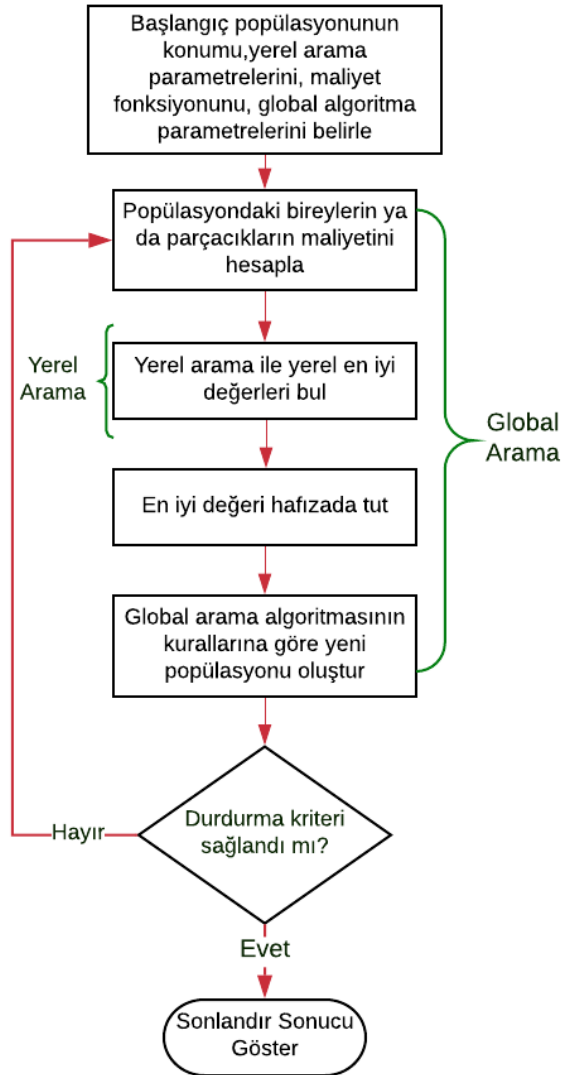
Şekil 2.1. Egg Crate Fonksiyonu



Şekil 2.2. Peaks(25) Fonksiyonu

KA algoritmalarında keşif-sömürü dengesi çok önemlidir. Keşif, küresel olarak arama sınırları doğrultusunda olabildiğince çok noktayı tarama işlemidir. Sömürü ise bulunan iyi noktanın çevresini arayarak en iyi değeri aramaktır. Eğer keşif için gerekli olan katsayılar artırılırsa arama algoritması rastgele arama algoritması haline gelir (Lin ve Liu, 2010). Yine aynı şekilde sömürü katsayıları çok fazla artırılırsa arama algoritması YA algoritması haline gelir. Keşif-sömürü dengesi problemden probleme değişiklik göstermektedir. Keşif-sömürü dengesini sağlamak adına Lin ve Liu (2010) yaptıkları çalışmada Adaptif Genetik Algoritma’yı (AGA) önermiştir. AGA’da bilinen GA’nın aksine sabit çaprazlama ve sabit mutasyon oranları yoktur. Uygunluk değeri, belirlenen maksimum mutasyon oranı ve minimum mutasyon oranı, yine belirlenen maksimum çaprazlama oranı ve minimum çaprazlama oranına bağlı olarak değişken mutasyon oranı ve değişken çaprazlama oranı elde edilir. Böylece nüfustaki bireylerin uygunluğu yerel olarak aynı ve optimum olma eğilimindeyse P_C ve P_M artacaktır; Popülasyondaki bireylerin uygunluğu dağılıma eğilimindeyse P_C ve P_M azalacaktır.

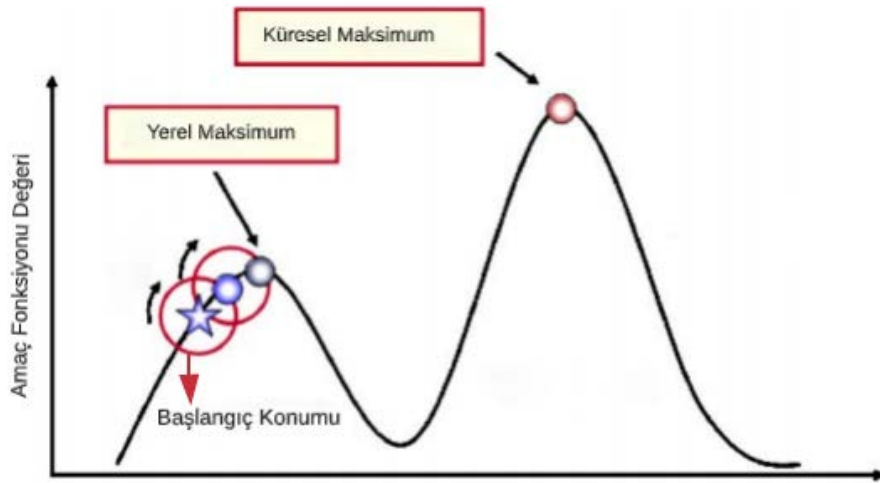
Keşif-sömürü dengesine göre katsayıları değiştirmek ve problemde tekrardan probleme tekrar ayarlama yapmak oldukça zahmetli bir işdir. Daha etkili bir arama algoritması için GA ile YA algoritmaların hibrit edilmesi Gen ve Yun (2006) tarafından önerilmiştir. Gen ve Yun (2006) yaptıkları çalışmada GA ile Tepe Tırmanma (TT) algoritmasını hibrit ederek güvenilirlik probleminde kullanmışlardır. Bu hibrit yaklaşım küresel arama algoritmalarına YA algoritmalarının eklenmesiyle oluşturulur. KA algoritmalarının geniş arama uzayını araması özelliğini ve YA algoritmalarının yerel optimum noktayı çok hızlı bulması özelliğini birleştirerek etkili bir yöntem oluşturmayı amaçlar. Hibrit yaklaşımla amaçlanan hedeflerden bir tanesi de KA algoritmalarındaki yoğun işlem yükünü hafifleterek daha hızlı bir şekilde en iyi noktayı bulmaktır. Gen ve Yun (2006) tarafından önerilen hibrit algoritma şeması Şekil 2.3’de gösterilmiştir.



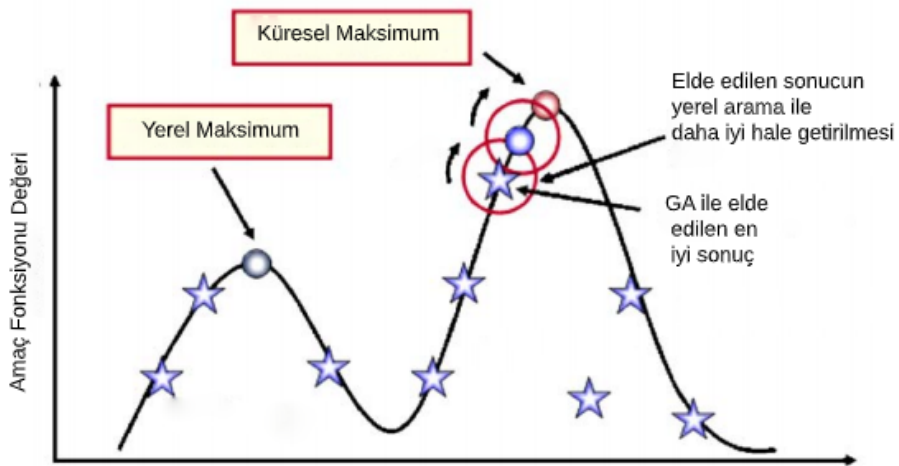
Şekil 2.3. KA ve YA algoritmaların hibrit edilmesi

Algoritma kromozomların arama uzayı sınırlarınca rastgele dağıtılmasıyla başlar. Bireylerin maliyet değerleri hesaplanır en düşük maliyet değerine sahip olan birey YA işleminde kullanılır. Daha sonra seçim, çaprazlama ve mutasyon işlevleri ile yeni popülasyon oluşturulur. Elde edilen sonuçlar tatmin edici düzeyde ise algoritma sonlandırılır. Tatmin edici düzeyde değilse yeni oluşturulan popülasyon ile algoritma yeniden başlatılır ve tatmin edici bir sonuç buluncaya kadar veya belirlenen iterasyon sayısı sağlanıncaya kadar algoritma devam eder.

Gen ve Yun (2006)'un yaptığı çalışmada YA algoritmasının konum hareketin Şekil 2.4'te, KA ile elde edilen maksimum amaç fonksiyonu değerine sahip bireyin YA algoritması kullanılarak küresel en iyi değere ulaşması Şekil 2.5'te gösterilmektedir.



Şekil 2.4. YA algoritması (Gen ve Yun, 2006)



Şekil 2.5. KA ile YA algoritmasının hibrit edilmesiyle amaçlanan sonuç (Gen ve Yun, 2006)

Sömürü-keşif dengesini sağlamak için önerilen hibrit yaklaşımlardan bir tanesi de bir KA algoritması ile başka bir KA algoritmasının hibrit edilmesidir. Kaveh ve Talatahari (2009) yaptıkları çalışmada sömürü özelliği yüksek olan BPBÇ algoritmasına PSO algoritmasını ekleyerek algoritmanın keşif kabiliyetini artırabileceğini öne sürmüşlerdir. İskelet tasarım parametrelerini PSO, GA, BPBÇ ve önerdiği Hibrit Büyük Patlama Büyük Çöküş (HBPBÇ) algoritması ile elde etmişlerdir ve en iyi sonucu HBPBÇ algoritmasının verdiğini gözlemlemişlerdir. Sedighizadeh ve ark. (2014), Kaveh ve Talatahari (2009) tarafından ortaya atılan HBPBÇ algoritmasını hibrit dağıtım sistemlerinde optimal yeniden yapılandırma ve dağıtılmış üretim gücü tahsisine uygulanmasında kullanmış; amaç fonksiyonu olarak toplam aktif güç kayıplarının en aza indirilmesini, voltaj kararlılık endeksinin en üst düzeye çıkarılmasını, toplam maliyetin en aza indirilmesini, şebekenin ürettiği toplam emisyonun en aza indirilmesini ele almışlardır. HBPBÇ algoritmasının, Bulanık Arı Algoritması, GA, Harmoni Arama Algoritmasına göre çok daha hızlı yakınsadığını ve yerel maksimum ya da minimuma takılmadığını gözlemlemişlerdir.

Kim (2011) yaptığı çalışmada küresel algoritmaların keskin geçişli fonksiyonlardaki yerel en iyi değeri bulamama sorununu gidermek için hibrit bir algoritma önermiştir. Bakterilerin yiyecek ararken sergiledikleri davranışlardan esinlenerek oluşturulan Bakteriyel Yiyecek Arama Algoritması (BYAA) ile GA'yı hibrit ederek belirlediği test fonksiyonları üzerinde test etmiş, elde ettiği sonuçları GA sonuçları ile karşılaştırmıştır. Önerdiği algoritmayla AVR sisteminin PID katsayılarını optimize etmiştir. Amaç fonksiyonu olarak yükselme zamanı, maksimum aşma, sürekli durum hatasının içeren bir kriter önermiştir. Bu kriter ve kriterdeki bir katsayının farklı değerlerini göz önüne alarak GA, PSO, hibrit PSO-GA ve hibrit BYA-GA'nın sonuçlarını karşılaştırmış bu problem için hibrit BYA-GA'nın daha iyi sonuç verdiğini gözlemlemiştir.

Kang ve ark. (2011) yaptıkları çalışmada YAKA'nın KA algoritmaları arasında genel anlamda küreseli arama konusunda diğerlerinden daha başarılı olduğunu fakat keskin kıvrımlı fonksiyonlarda halen başarısız olduğunu ifade etmişlerdir. Bu kusuru gidermek ve yakınsama hızını artırmak için YAKA ile HJ algoritmalarının hibrit edilmesi önermişlerdir. Elde edilen hibrit algoritma sıkça kullanılan performans kriterlerince PSO ve YAKA ile karşılaştırmışlar, Hibrit Yapay Arı Koloni Algoritması'nın (HYAKA) oldukça iyi sonuçlar verdiğini ifade etmişlerdir.

Amiri ve ark. (2019) yaptıkları çalışmada rehabilitasyon için üretilen 4 serbestlik dereceli giyilebilir dış iskeletin uygun PID kontrolcü kazançlarını elde etmek için GA ile başlayan PSO ile devam eden hibrit algoritma önermişlerdir. Dış iskeletin her bir uzvunun transfer fonksiyonunu fiziksel metotlarla türetmişler ve transfer fonksiyonunu elde ederken, her bir uzvu bir serbestlik derecesi olarak ele almışlardır. Önerdikleri hibrit algoritmayı ITAE amaç fonksiyonuna göre GA ve PSO ile kıyaslamışlardır.

Luan ve ark. (2019) yaptıkları çalışmada tedarikçi seçim problemi için GA ile Karınca Koloni Algoritması (KKA) hibrit ederek uygulamışlardır. Önerdikleri hibrit algoritmanın sonuçlarını GA ve KKA'nın sonuçlarıyla karşılaştırmışlardır.

Önder ve ark. (2013) yaptıkları çalışmada, YAKA, PSO ve GA'nın hibrit edilmesini önermiş, gezen satıcı probleminde uygulamışlardır. Elde ettikleri sonuçları Karınca Kolonisi algoritmasının sonuçları ile karşılaştırmışlardır.

Shunmugapriya ve Kanmani (2017) yaptıkları çalışmada veri kümesinden alakalı ve faydalı bilgileri çıkararak sınıflandırma sürecini hızlandırmaya yardımcı olan Özellik Seçimini önerdikleri ABC ve KKA hibrit edilmesinden oluşan algoritma ile optimize etmeyi amaçlamışlardır.

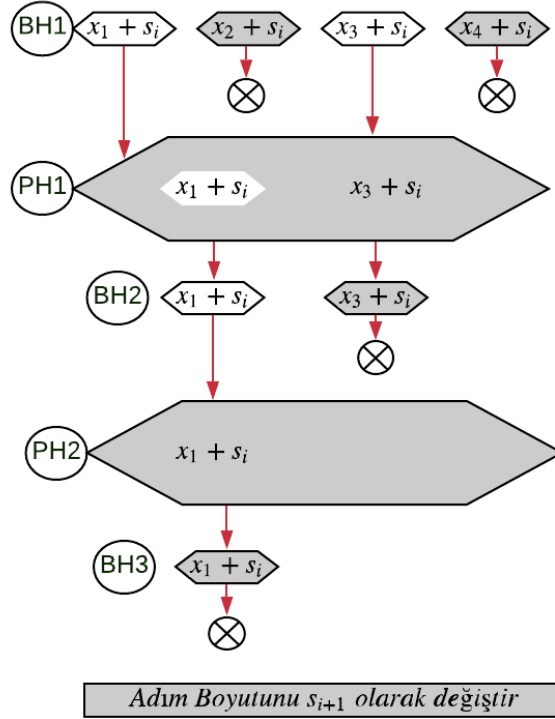
Erdemir ve Kalyoncu (2015) Bir Ağır Vasıtanın Çok Akslı Direksiyon Mekanizmasının Ackerman direksiyonlama hatasını minimize etmek için AA ile HJ Algoritmasını hibrit ederek kullanmışlardır.

YA algoritmalarının yüksek yakınsama hızını KA algoritmalarının keşif özellikleriyle birleştirmek adına ortaya atılan yaklaşımlardan bir tanesi de YA algoritmalarını birden fazla, farklı başlangıç noktalarından başlatarak en iyi değeri küresel en iyi olarak atamaktır. Benasla ve ark. (2008) yaptıkları çalışmada birden fazla başlangıç noktasından başlanarak elde edilen minimum değerlerin karşılaştırılmasıyla HJ algoritmasının küresel optimizasyonda da kullanılabileceğini ortaya koymuşlardır.

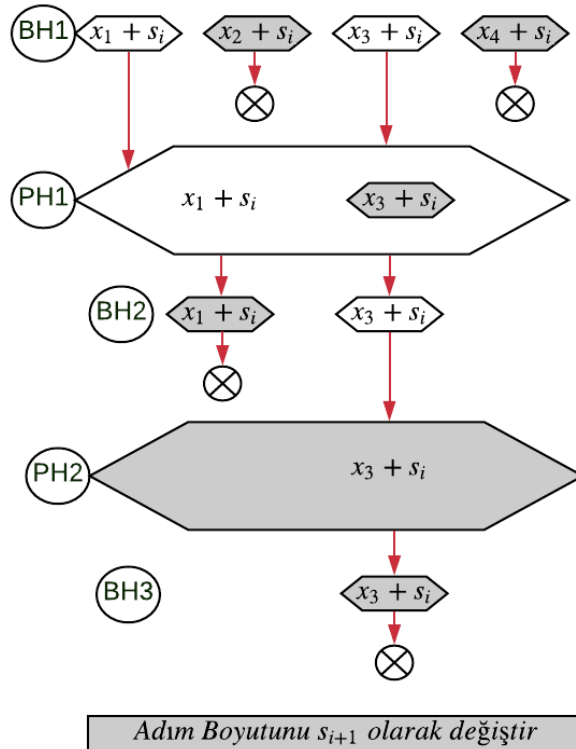
Alfi ve Modares (2011) yaptıkları çalışmada doğrusal olmayan bir sistemin matematiksel modelini ve optimum PID katsayılarını bulmak amacıyla PSO'nun kolay yerel en iyiye kolayca yakalanmasından dolayı Adaptif Parçacık Sürü Optimizasyonunu (APSO) önermişlerdir. Sistemin matematiksel modelini ve sistemin optimum PID katsayılarını doğrusal olarak ağırlık faktörü azalan PSO algoritması, GA ve APSO ile elde etmişlerdir ve APSO'nun diğerlerine göre daha etkili olduğunu gözlemlemişlerdir.

Moser (2009) yaptığı çalışmada HJ algoritmasında çok boyutlu arama ve örüntü hareketi konusunda 2 olası durumu açıklamıştır. Şekil 2.6'da gösterilen senaryoda ilk olarak her bir boyutta adım boyutu kümesindeki en büyük eleman iki yönde eklenir ve

bireysel olarak noktanın önceki noktadan daha iyi bir konuma gelip gelmediğine bakılır. Senaryo 1 ve 2’de (Şekil 2.6 ve Şekil 2.7).



Şekil 2.6. Çok değişkenli arama şeması senaryo 1



Şekil 2.7. Çok değişkenli arama şeması senaryo 2

1. Boyutta ve 3. Boyuttaki hareketler başarılı olmuştur (BH1)(Beyaz kutucuklar başarılı sayılmış gri olanlar ise başarısız olmuştur.). Başarılı olan boyutlarda örüntü (yapı) hareketi her iki boyutun konumu da adım boyutu kadar artırılarak yapılır(BH1). Örüntü hareketi Şekil 2.6'da başarısız olarak ele alınmış Şekil 2.7'de ise başarılı olarak ele alınmıştır. Örüntü hareketi başarılı olmuştur her iki konumda da iki yönde (ilk yön başarısız olursa) yeniden bireysel hareket yapılır. Örüntü hareketi başarısız olursa başarılı olan hareketler üzerine yeniden adım boyutu eklenerek değerlendirilir. Her seferinde sırayla 1 bireysel hareket 1 örüntü hareketi yapılır. Algoritma tek bir boyutta başarısız olduğunda, bu boyut, adım boyutu tüm boyutlar için daha küçük bir değere ayarlanana kadar bekler.

Literatürde amaç fonksiyonu olarak hata değerinin zamanla çarpımının integralinin mutlak değeri (ITAE), hata değerleri integralinin mutlak değeri (IAE), hata değerinin karesinin zamanla çarpımının integralinin mutlak değeri(ITSE), hata değerlerinin karesinin integralinin mutlak değeri (ISE) kullanılmıştır. Bahsedilen performans kriterlerinin matematiksel ifadeleri Denklem 2.1, Denklem 2.2, Denklem 2.3 ve Denklem 2.4'te gösterilmektedir.

$$f_{IAE} = \int_0^t |e(t)| dt \quad (2.1)$$

$$f_{ISE} = \int_0^t (e(t))^2 dt \quad (2.2)$$

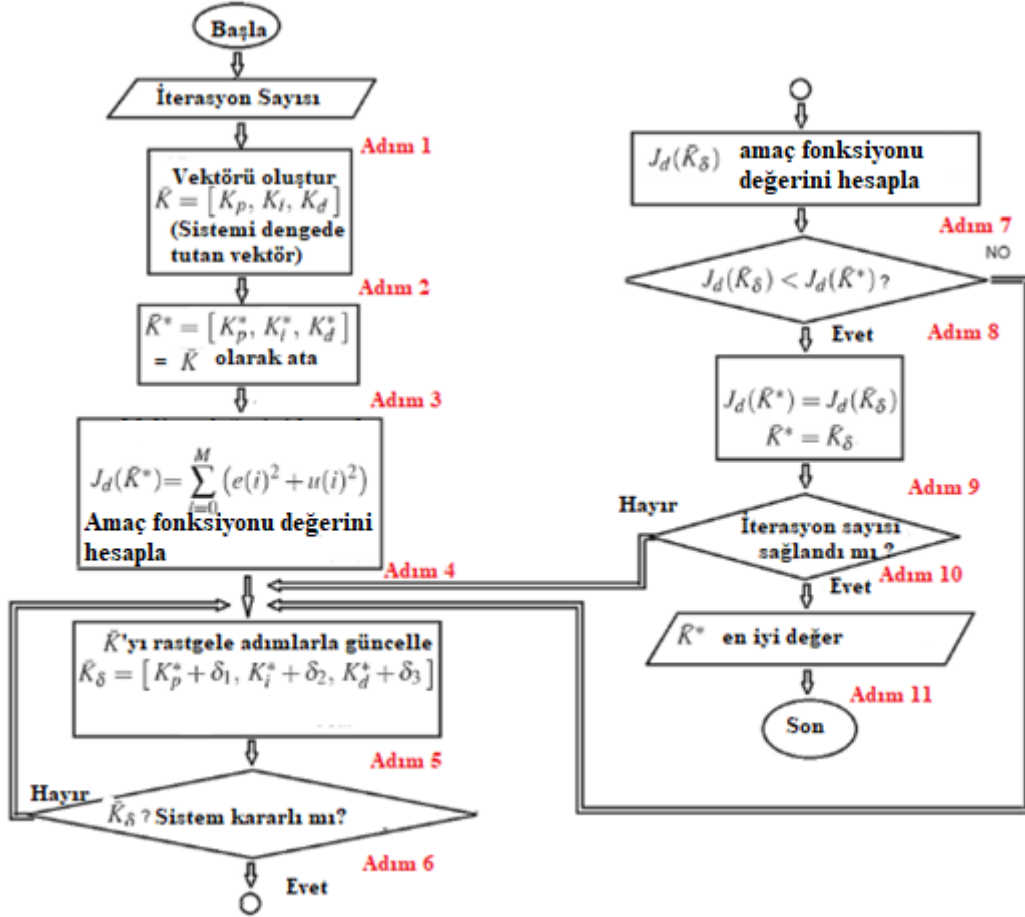
$$f_{ITAE} = \int_0^t t|e(t)| dt \quad (2.3)$$

$$f_{ITSE} = \int_0^t t(e(t))^2 dt \quad (2.4)$$

Cervantes Escorcía ve ark. (2019), dilimlenmiş domates kurutma işleminde ortam sıcaklığı kontrolü konusunda ilk olarak PI kontrolcü uygulamış daha sonra sistemin çıkışında titreşimler gözlemlendiğinden dolayı PID kontrolcü uygulamıştır. PID kontrolcü katsayılarını Tepe Tırmanma (TT) algoritması ile optimize etmiştir. Amaç fonksiyonu olarak karesel performans indeksini kullanılmış tatmin edici sonuçlar elde edildiği ifade edilmiştir. Karesel performans indeksi denklem 2.5'de gösterilmektedir. Denklemde $e^2(t)$ terimi giriş ve çıkış arasındaki hatanın karesini, $u^2(t)$ kontrolcü girişinin karesini

ifade eder. Cervantes Escorcía ve ark. (2019) yaptıkları çalışmada Şekil 2.8'deki algoritma şemasını kullanmıştır.

$$f_{Karesel} = \int_0^{t1} (e^2(t) + u^2(t)) dt \quad (2.5)$$



Şekil 2.8. TT Algoritması ile PID katsayıları optimizasyonu algoritma adımları
(Cervantes Escorcía ve ark., 2019)

Almabrok ve ark. (2018), ikinci dereceden bir ısı, havalandırma ve iklimlendirme (HVAC) sisteminin oransal-integral-türev (PID) kontrolcü parametrelerinin hızlı ayarlanması için yeni bir metot olarak BPBÇ algoritmasını önermiştir. Elde edilen sonuçlar diğer KA metodları (PSO-GA) ile farklı performans kriterlerine göre (ITAE-ITSE-IAE-ISE) karşılaştırılmışlardır. Yakınsama hızı ve hesaplama zamanı bakımından önerilen metodun diğerlerine nazaran daha tatmin edici sonuçlar verdiğini gözlemlemiştir.

Altintas ve Aydin (2017), manyetik kaldırma süspansiyon sisteminin kararsız sistem olduğu ifade etmiş ve sistemi PID kontrolcü ile kararlı hale getirmeye çalışmıştır. Kesir dereceli ve tam sayı PID katsayıları, GA ve BPBÇ kullanılarak optimize edilmiştir. BPBÇ algoritmasının daha iyi sonuçlar elde ettiğini gözlemlemişlerdir. Amaç fonksiyonu olarak ISE'yi kullanmışlardır.

Oi ve ark. (2008), MSI ile sistemin transfer fonksiyonunu elde etmiş, sistem kontrolü için PID kontrolcüyü kullanmıştır. Tatmin edici bir kontrol performansı için PID kontrolcü kazançlarını PSO ile kısıtlamalı T_d , T_r , T_p , T_s *ess* ve M_p 'ye göre optimize etmiş ve sonuçları CHR yöntemi ile kıyaslamıştır. Elde edilen sonuçlarda PSO'nun daha iyi sonuçlar verdiğini gözlemlemişlerdir.

Chang (2013), literatürde YAKA'nın birçok mühendislik alanında etkili olarak kullanıldığını gözlemlemiş, yaptığı çalışmada doğrusal olmayan sürekli karıştırılan kimyasal tank reaktörü sisteminin, optimum PID kontrolcü parametrelerini YAKA ile elde etmeyi amaçlamıştır. YAKA ile elde edilen sonuçlar GA ile karşılaştırmış YAKA'nın daha iyi sonuçlar verdiğini gözlemlemiştir. ITAE'nin çıkış cevabının yerleşme zamanını minimize etmek için kullanıldığını hata minimizasyon problemleri için ISE ve IAE'nin iyi sonuçlar verdiğini ifade etmiştir.

Gaing (2004), otomatik voltaj regülatörü (AVR) sisteminde PID kontrolcü katsayılarını PSO ve GA ile elde etmiştir. Amaç fonksiyonu olarak Denklem 2.6'da gösterilen yeni bir kriter önermiştir. Önerilen kriterde katsayıları değiştirerek PSO ve GA ile elde ettiği değerleri karşılaştırmış ve PSO'nun AVR sistemlerinde daha başarılı olduğunu grafiklerle göstermiştir. Bu yöntemin GA gibi erken yakınsama probleminin olmadığını ve daha kaliteli bir çözüm sunduğunu aynı zamanda problemi daha hızlı çözdüğünü belirtmiştir. Geleneksel PSO'dan farklı olarak w değerinin iterasyon sayısına göre değişmesi gerektiği ifade edilmiştir. Denklem 2.7' de (Gaing, 2004) tarafından önerilen w ifadesi gösterilmektedir. Yapılan çalışmada popülasyondaki parçacık sayısı 20 seçilmiş ve 20 kez çalıştırılmıştır. Farklı problemler için w_{max} , w_{min} , c_1 , c_2 katsayılarının ve V_{max} değerlerinin değiştirilmesi gerektiğini ifade etmiştir (Gaing, 2004).

$$f_{Gaing} = (1 - e^{-k})(M_p + ess) + e^{-k}(T_s - T_r) \quad (2.6)$$

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} iter \quad (2.7)$$

Panda ve ark. (2012), AVR sistemine PID kontrolcü uygulamış ve PID kontrolcü katsayılarını optimize etmek için Pedersen ve Chipperfield (2010) tarafından önerilen Basitleştirilmiş Parçacık Sürü Optimizasyonu (MOL) Algoritmasını kullanmışlardır. PSO'dan tek farkı yeni hız değerinin elde edildiği denklemdeki c_1 katsayısının 0 alınmasıdır. MOL algoritmasında kullanılan yeni hız ifadesi Denklem 2.8 de gösterilmektedir. Böylelikle parametre seçimini de kolaylaştırmıştır. Elde ettiği sonuçları YAKA, PSO, Diferansiyel Evrim (DE) algoritmaları ile karşılaştırmışlardır. Bu problem için MOL'ün daha iyi sonuçlar verdiğini ve performans kriterlerinden ITAE'nin daha iyi sonuç gösterdiğini grafiklerle ifade etmişlerdir.

$$V_{iy} = wV_i + c_2rand(1)(P_{ki} - X_i) \quad (2.8)$$

Chang ve Shih (2010) doğrusal olmayan ters sarkaç modelini PID kontrol ile kontrol etmeyi amaçlamışlar, PID katsayılarını geliştirilmiş PSO algoritması ile optimize etmişlerdir. PSO algoritmasında arama etkisini artırmak için yeni hız değerinde bir değişiklik önermişlerdir. PSO algoritmasında bir parçacığın sadece yerel en iyi ve küresel en iyi parçacıktan etkilenmediğini aynı zamanda bir önceki popülasyonun en iyi değerinden de etkilenebileceğini ifade etmişlerdir. MOL algoritmasının aksine yeni hız değerine fazladan bir terimin girmesi parametre sayısını artırmış, parametre seçimini zorlaştırmıştır. Önerilen yeni hız denklemi aşağıda gösterilmiştir. P_{yi-1} ifadesi bir önceki popülasyonun en iyi değerini ifade etmektedir. Diğer simgeler PSO kısmında anlatılmaktadır.

$$V_{iy} = wV_i + c_1rand(1)(P_{yi} - X_i) + c_2rand(1)(P_{ki} - X_i) + c_3rand(1)(P_{yi-1} - X_i) \quad (2.9)$$

Reddy (2015), otomatik elektrik üretimi işleminin frekans kontrolünü PID kontrolcü ile kontrol etmeyi amaçlamış, PID katsayılarını PSO ve MOL ile optimize etmiştir. Amaç fonksiyonu olarak ITSE'yi kullanmıştır. MOL aracılığıyla elde edilen PID katsayılarının daha etkili sonuç gösterdiği gözlemlenmiştir.

Erkol (2017), ZN, PSO ve GA ile sabit mıknatıslı DC motorun optimum PID parametrelerini belirlemeyi amaçlamış ve PSO'nun daha iyi sonuç verdiğini gözlemlemiştir. GA ve PSO'da amaç fonksiyonu olarak IAE ve ISE'yi kullanmış IAE amaç fonksiyonunun daha iyi sonuçlar verdiğini gözlemlemiştir. ZN yönteminin

iyileştirilmesinin zaman alıcı olduğunu ve sonuçlar karşılaştırıldığında ZN'nin yetersizliği gözlemlenmiştir.

Fister ve ark. (2016) yaptıkları çalışmada DEA, GA, PSO, GKA, Yarasa Algoritması, Hibrit Yarasa Algoritmasını iki serbestlik dereceli SCARA robotun optimum PID kontrolcü katsayılarının bulunması için kullanmışlardır. Her bir algoritmada 10 adet bireyle 10 iterasyon boyunca işlem yaparak uzun süren işlemi hem kısaltmak istemiş hem de algoritmaların etkinliğini gözlemlemek istemişlerdir. PSO'nun bu problem için daha iyi sonuçlar verdiğini gözlemlenmiştir. Maksimum aşma, sürekli durum hatası, yerleşme zamanını içeren yeni bir amaç fonksiyonu önermişlerdir.

Elbayomy ve ark. (2008) yaptıkları çalışmada uzay taşıtlarının hareketli yüzeyini kontrol eden döner eyleyicinin açılma pozisyonunu kontrol etmek için elektro-hidrolik SERVO eyleyici sisteminin uygun PID katsayılarının elde edilmesi için IAE amaç fonksiyonunu göz önüne alarak GA'yı kullanmışlardır.

Jaen-Cuellar ve ark. (2013) yaptıkları çalışmada CNC makinelerinde ve endüstriyel bir robotta kullanılan hidrolik SERVO sistem PID kontrolcü ile kontrol edilmiş, uygun PID kontrol katsayıları GA aracılığıyla elde edilmiştir. Bu çalışmada amaç fonksiyonu olarak Denklem 2.11'de gösterilen yeni bir kriter önerilmiştir. Denklem 2.10'da kullanılan σ hata dizisindeki standart sapma değerini, n_e hata dizisindeki eleman sayısını ifade eder. Denklem 2.10'daki w_1, w_2 ve w_3 ağırlık katsayılarını ifade etmektedir.

$$MSE = \frac{\sigma^2}{n_e} \quad (2.10)$$

$$J = w_1 ISE + w_2 M_p + w_3 MSE \quad (2.11)$$

Mishra ve Swarnkar (2014) yaptıkları çalışmada elektro hidrolik SERVO sistemine P kontrolcü, PD kontrolcü ve PID kontrolcüyü ayrı ayrı uygulamışlardır. Kontrolcü kazançlarını PSO ile her bir kontrolcü için ayrı ayrı optimize etmişlerdir. Bu sistem için PID kontrolcünün daha iyi sonuçlar verdiğini ifade etmişlerdir.

Tandan ve ark. (2015) yaptıkları çalışmada Mishra ve Swarnkar (2014)'ın kullandığı elektro hidrolik SERVO sistemini kullanmışlar aynı şekilde sisteme PID kontrolcü uygulamayı amaçlamışlardır. PID kontrolcü kazançlarını PSO ile optimize etmişlerdir. Elde edilen sonuçları otomatik ayarlama ve ZN metodları ile karşılaştırmışlardır.

Lu ve ark. (2010) tarafından yapılan çalışmada kömür taşıma ve vinç sistemlerinde parametrelerinin AGA ile optimize edildiği PID kontrolcü kullanılmıştır. Amaç fonksiyonunda hata değeri, hatanın değişimi ve maksimum aşma dikkate alınmıştır.

Li ve ark. (2012) yaptıkları çalışmada elektro hidrolik oransal valfi konveyör hız kontrolünün etkin olarak yapılabilmesi için kullanmışlardır. Sistem transfer fonksiyonunu fiziksel yöntemlerle türetmişlerdir. Türetilen sistemi kullanarak PID kontrolcü uygulamışlar, PID kontrolcü parametreleri MATLAB yardımıyla elde etmişlerdir.

Yu ve ark. (2012) yaptıkları çalışmada metal şerit haddeleme için kullanılan hidrolik presin matematiksel modelini yapay zekâ tekniklerinden PSO'yu kullanarak elde etmeyi amaçlamışlardır. Hidrolik sistemlerin doğrusal olmayan karakteristikler içerdiğinden dolayı sistemin matematiksel modelinin fiziksel metotlarla doğru olarak türetilmeyeceğinden bahsetmişlerdir. Sistemin daha doğru matematiksel modelini elde etmek için geliştirilmiş kaotik PSO algoritmasını önermişlerdir.

Ye ve ark. (2017) hidrolik ekskavatör de silindir sisteminin pozisyon kontrolü üzerine çalışmışlardır. Doğrusal olmayan davranışlar dikkate alınarak sistemin matematiksel modeli fiziksel metotlarla türetilmiştir. PID kontrolcü kazançları geliştirilmiş PSO aracılığıyla elde edilmiş, sonuçlar geleneksel PSO ile karşılaştırılmış, geliştirilmiş PSO'nun daha iyi sonuçlar verdiğini gözlemlemişlerdir.

Kalyoncu ve Haydim (2009), yaptıkları çalışmada Elektro-hidrolik bir sistemin iç sızıntısını da hesaba alarak matematiksel olarak modellemişlerdir. Sistemi bulanık mantık kontrolcü ile kontrol etmişlerdir.

Kalyoncu (2008), yaptığı çalışmada dönen prizmatik eklemlili esnek bir robot manipülatörün matematiksel modellenmesi ve dinamik tepkisi incelenmiştir. Esnek manipülatörün matematiksel modelini, Lagrange'nin hareket denklemini kullanarak elde etmiştir. Çalışma da yapay sinir ağları kullanarak eklemlili bir robot kolun ters kinematiğinin modellenmesini ele alınmıştır. Arılar Algoritması, üç uzuvlu bir manipülatörün eklemlerinin ters kinematiğini öğrenmek için üç çok katmanlı algılayıcıyı eğitmek için kullanılmıştır. Yapının kontrolü için PID kontrolcü tasarımı yapılmıştır.

Pham ve Kalyoncu (2009), yaptıkları çalışmada esnek tek bağlantılı robot kol için ölçeklendirme kazançlarını ve bulanık mantık denetleyicisinin diğer parametrelerini ayarlamak için Arı Algoritmasını kullanmışlardır. Bu çalışmada en az titreşimle robot kolu istenen noktaya taşımayı amaçlanmıştır.

Kalyoncu ve Botsalı (2004) yaptıkları çalışmada dönen bir prizmatik eklemden kayan elastik bir robot kolun titreşim analizini incelemiştir. Elastik kolun matematiksel modelini Lagrange'nin hareket denklemini kullanılarak elde etmişlerdir.

Tinkir ve ark. (2010) yaptıkları çalışmada çift ters sarkaç sisteminin devrilme pozisyon kontrolünü tasarlamışlardır. Tabana monte edilmiş bir motorla sağlanan tork, sistemin konum kontrolünde kullanılmıştır. Sistemin kontrolü için PID kontrolcü ve bulanık mantık kontrolcü kullanılmıştır.

Botsalı ve ark. (2010) tarafından yapılan çalışmada bir robot manipülatörünün yörünge kontrolünde kullanılmak üzere bulanık mantık kontrolcü tasarlamışlardır.

Pham ve ark. (2008) tarafından yapılan çalışmada esnek robot manipülatörün PID kontrolcü katsayılarını Arı Algoritması ile optimize etmişlerdir.

Şahin ve ark. (2014b) tarafından yapılan çalışmada giyilebilir dış iskeletin kinematik sentezi, mekanik ve elektro-hidrolik sistem tasarımı konusunda çalışılmıştır.

Sen ve Kalyoncu (2015) yaptıkları çalışmada, ters sarkaç sisteminde PID kontrolcü katsayılarını Arı Algoritması kullanarak optimize etmişlerdir.

Şahin ve ark. (2014a) yaptıkları çalışmada dış iskelet tasarımının yük taşıyan bir insan için; yürüme, oturma ve ayakta durma hareketleri sırasında taşıdığı harici yüke karşı kuvvet geri besleme kontrolü sağlamak üzere, kontrolcü tasarımı konusunda çalışmışlardır. Önerilen dış iskelet sistemi, iki servo hidrolik eyleyici tarafından çalıştırılan ve kontrol edilen iki ayağı vardır. PI kontrolcüyü, sistemin güç kontrolü için kullanmışlardır.

Şen ve ark. (2018) tarafından yapılan çalışmada Northridge Depremi'nin uyarı altında iki katlı bir bina benzeri yapının aktif titreşim kontrolü için geleneksel PID kontrol katsayılarını ayarlamak için Arı Algoritmasını kullanmışlardır.

Samakwong ve Assawinchaichote (2016), elektro hidrolik valflerin sanayide yaygın olarak kullanıldığını, diğer cihazlara kıyasla yüksek güç, hızlı tepki ve hafif olma gibi avantajlara sahip olduğundan çok kullanıldığını ifade etmişlerdir. Elektro hidrolik SERVO sistemleri salınım problemine sahip olduğundan dolayı bu sistemleri kontrol etmek için GA ile PID kontrol katsayılarının elde edilmesini önermişlerdir. PSO, GA ve ZN karşılaştırılmış bu problemde GA'nın hem ZN'ye göre hem de PSO'ya göre daha başarılı olduğunu ifade edilmiştir.

Vatansever ve Şen (2013), uygun PID katsayılarının bulunması için ZN, Cohen-Coon, CHR ve Wang-Juang-Chan gibi yöntemlerle yapay zekâ yöntemlerinden olan GA'nın karşılaştırılabileceği bir simülatör tasarlamışlardır.

Bu tez çalışmasında çok serbestlik dereceli hidrolik robot manipülatörün her bir serbestlik derecesinin transfer fonksiyonu MSI aracılığıyla elde edilmiş, uzuvların PID Kontrolcü katsayıları, minimum maksimum aşma değeri, minimum yükselme zamanına göre optimize edilmesi amaçlanmıştır, optimizasyon metotlarından PSO, GA, BPBÇ, AA, GKA algoritmaları aracılığıyla optimize edilmiş, elde edilen sonuçlar BPBÇ algoritması ile HJ algoritmasının hibrit edilerek oluşturulan bu çalışmada önerilen algoritma ile karşılaştırılmıştır.

3. MATERYAL VE YÖNTEM

Bu çalışmada, hidrolik robot manipülatör sisteminin her bir hareketinin transfer fonksiyonu MSI'da elde edilmiştir. Bu hareketlerin optimum kontrolü için PID katsayılarının, belirlenen optimizasyon yöntemlerince kapalı çevrim basamak cevabına göre önerilen amaç fonksiyonu minimize edilerek bulunması amaçlanmıştır.

Bu amaçları gerçekleştirmek için bölüm 3.1'de PID kontrolcü yapısı, bölüm 3.2'de deneysel modelleme, bölüm 3.3'de sezgisel arama algoritmaları, Bölüm 3.4'te YA algoritmaları, Bölüm 3.5'te ise bu çalışmada önerilen hibrit arama algoritması anlatılmıştır.

3.1. PID Kontrolcü ve Ziegler Nichols Yaklaşımı

PID kontrolcü, kontrol sistemlerinde çıkış ile giriş arasındaki hata değerini, Laplace uzayında belirli katsayılarla çarparak kontrol edilen sisteme giriş değeri olarak verir ve bu hata değerini sifira indirmeyi amaçlar. P kontrolcü hatanın Laplace uzayında K_p (oransal kazanç) ile çarpılmasıyla elde edilir. D kontrolcü hatanın türevinin Laplace uzayında K_d katsayısı ile çarpılmasıyla elde edilir. I kontrolcü ise hatanın integralinin Laplace uzayında K_i katsayısı ile çarpılmasıyla elde edilir. PID kontrollerin tasarımında en önemli konulardan biri, belirli bir amaç fonksiyonuna göre kazanç parametrelerinin belirlenmesidir (Coşkun ve Terzioğlu, 2007). Üç terimli kontrolcü olarak da bilinen PID kontrolcünün transfer fonksiyonu Denklem 3.1'de gösterilmiştir. Denklem 3.2'de ise zaman sabitesini içeren formda gösterilmiştir (Ang ve ark., 2005).

$$G(s) = K_p + \frac{K_i}{s} + K_d s \quad (3.1)$$

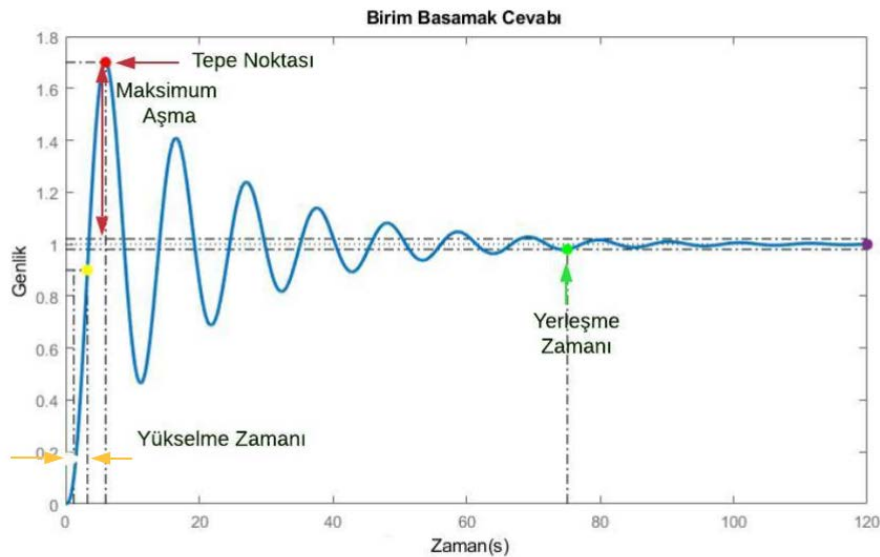
$$G(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (3.2)$$

Oransal kontrolcü, yükselme zamanının azalması sağlar buna karşı maksimum aşmayı artırır ve sürekli durum hatasını ortadan kaldırmada düşük etkiye sahiptir. İntegral kontrolcü, sistemin sürekli durum hatasını minimize etmede etkilidir fakat maksimum aşmayı artırır. Türev kontrolcüsü, maksimum aşmayı azaltır fakat sürekli durum hatasını pratikte kötüleştirebilir. Kapalı çevrim sisteminde PID katsayılarının (K_p , K_i , K_d) her birinin etkisi Çizelge 3.1'de gösterilmiştir (Ang ve ark., 2005).

Çizelge 3.1. PID Katsayılarının Sistem Parametrelerine Etkisi (Ang ve ark., 2005)

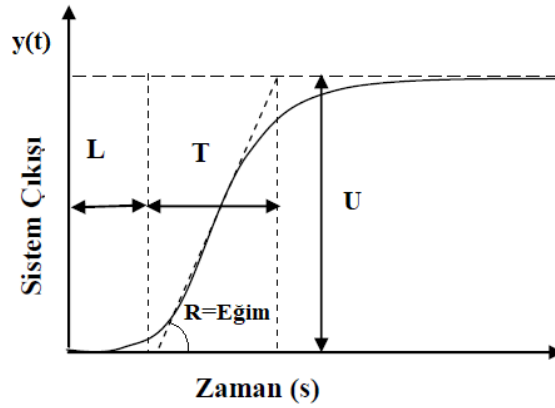
Kapalı Çevrim Cevabı	Yükselme Zamanı	Maksimum Aşma	Yerleşme Zamanı	Sürekli Durum Hatası	Kararlılık
Artan K_p	Azalır	Artar	Az Artar	Azalır	Geriler
Artan K_i	Az Azalır	Artar	Artar	Çok Azalır	Geriler
Artan K_d	Az Azalır	Azalır	Azalır	Az Değiştirir	Gelişir

Maksimum aşma tepe noktasının çıkış değerinin zaman sonsuz olduğundaki değerini ne kadar aştığının ifadesidir. Sistem çıkışının girişin %10'undan ile %90'ına yükselirken geçen zamanın ifadesidir. Son durum ile çıkış değeri arasındaki farkın %2'ye düştüğü zaman yerleşme zamanı olarak adlandırılır. Sürekli durum hatası ise zaman sonsuza gittiğindeki çıkış değeri ile referans giriş değeri arasındaki farkın ifadesidir. Şekil 3.1'de sistemin basamak cevabı parametreleri gösterilmektedir.



Şekil 3.1. Sistem basamak cevabı parametreleri

Açık çevrim ve kapalı çevrim kontrol sistemlerinde Ziegler ve Nichols (1942)'ün önerdiği ZN metodunda sistemin matematiksel modeline ihtiyaç duyulmadığından ve uygulama sağladığı kolaylıklardan dolayı yaygın olarak kullanılmaktadır. Açık çevrim ZN metodunda kontrol çevrimi kapalıysa öncelikle açık hale getirilir. Kontrol değişkeni manuel olarak ayarlanarak istenilen çalışma noktasına getirilir. Daha sonra sisteme basamak girişi uygulanır. Bu basamak girişinin genliğinin küçük olması istenir. Fakat uygulanan girişin kontrol çıkışında gözle görülebilir bir değişiklik sağlaması gerekmektedir. Sistem çıkışındaki değişiklik gözlemlenir ve karakteristik parametreler olan R (Eğim) ve L (ölü zaman veya gecikme) hesaplanır. Bahsedilen kontrol parametreleri Şekil 3.2'de grafik üzerinde ve kontrol parametrelerini gösteren denklemler Çizelge 3.2'de gösterilmiştir (Haugen, 2010b).



Şekil 3.2. Sistemin çıkış grafiği

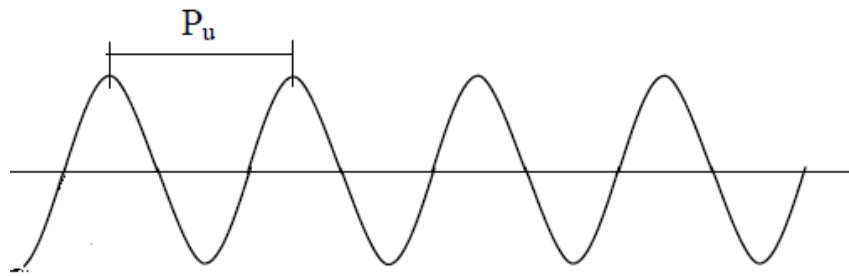
Çizelge 3.2. Açık Çevrim ZN Kontrol Parametrelerinin Formülü

Açık Çevrim Cevabı	K_p	$T_i = \frac{K_p}{K_i}$	$T_d = K_p K_d$
P Kontrolcü	$\frac{1}{LR/U}$	∞	0
PI Kontrolcü	$\frac{0.9}{LR/U}$	$3.3L$	0
PID Kontrolcü	$\frac{1.2}{LR/U}$	$2L$	$0.5L$

(Haugen, 2010b)

Kapalı çevrim ZN metodunda ise kontrol değişkeni çalışma noktasına getirilir. Böylece ayarlama sırasındaki değişkenlerin limitlere ulaşma şansını en aza indirilir. Kontrolcü kazanç değerleri (K_p , K_d , K_i) 0'a getirilir. Sistem kapalı çevrim haline

getirilir ve sisteme yaklaşık %5 genlikli step girişi uygulanır. Adım çok küçük olursa ölçüm gürültüsünden dolayı salınımlar gözlemlenemeyebilir. Sistem çıkışındaki sinyallerde sürekli salınımlar oluşana kadar K_p artırılır, sürekli eşit genlikli salınımlar sistemin kararlılık sınırında olması anlamına gelir. Elde edilen K_p değeri kritik kazanç değeridir. K_{pu} , olarak sembolize edilir. K_{pu} değeri sürekli salınım başladığı en küçük K_p değeri olmalıdır. Yüksek K_p , değerlerinde elde edilen değerler sistemde kararsızlığa neden olur. Daha sonra sürekli salınımların kritik periyodu (P_u) ölçülür. Şekil 3.3'de P_u periyodu gösterilmiş, Çizelge 3.3'te kontrol parametrelerini gösteren denklemler verilmiştir (Coşkun ve Terzioğlu, 2007; Haugen, 2010a).



Şekil 3.3 Kapalı çevrim ZN metoduna ait çıkış sinyali

Çizelge 3.3. Kapalı çevrim ZN kontrol parametrelerinin formülü

Açık Çevrim Cevabı	K_p	$T_i = \frac{K_p}{K_i}$	$T_d = K_p K_d$
P Kontrolcü	$0.5K_{pu}$	∞	0
PI Kontrolcü	$0.45K_{pu}$	$\frac{P_u}{1.2}$	0
PID Kontrolcü	$0.6K_{pu}$	$0.5 P_u$	$0.125 P U$

(Haugen, 2010a)

3.2. Deneysel Modelleme

Bu çalışmada hidrolik robot manipülatörün deneysel modellenmesi için sistem tanılama metodu kullanılmıştır. Sistem tanılama metodu, sistemin giriş ve çıkış sinyallerinin ölçümlerini kullanarak matematiksel dinamik sistem modelleri oluşturmak için bir metottur (Rozali ve ark., 2010; MATLAB, 2019). Sistem tanılama süreci şunları gerektirir:

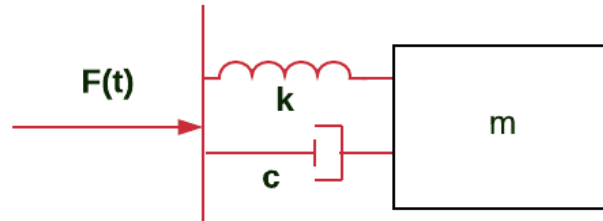
- i) Sistemin giriş ve çıkış sinyallerinin zaman veya frekans alanında ölçüm değerleri,
- ii) Model yapısı,

iii)Seçilecek model yapısındaki ayarlanabilir parametrelerin değerini tahmin etmek için bir tahmin yöntemi,

iv) Modelin yeterli olup olmadığını değerlendirme.

Dinamik bir sistemde, çıkış sinyallerinin değerleri hem giriş sinyallerinin anlık değerlerine hem de sistemin geçmiş davranışına bağlıdır. Örneğin, bir araba koltuğu dinamik bir sistemdir. Koltuk şekli (oturma pozisyonu), hem yolcunun mevcut ağırlığına (anlık değer) hem de bu yolcunun koltukta ne kadar oturduğuna (geçmiş davranış) bağlıdır. Model, sistemin girdi ve çıktı değişkenleri arasındaki matematiksel bir ilişkidir. Dinamik sistem modelleri genel olarak diferansiyel veya fark denklemleri, transfer fonksiyonları, durum-uzay denklemleri ve kutup-sıfır-kazanç modelleri ile tanımlanmaktadır (MATLAB, 2019).

Sık kullanılan bir dinamik model örneği, bir yay kütle damperi sisteminin hareket denklemdir. Bir sonraki şekilde gösterildiği gibi, kütle, bağlı olduğu tabana uygulanan $F(t)$ kuvvetine tepki olarak hareket eder. Bu sistemin girişi ve çıkışı sırasıyla; $F(t)$ kuvvet ve yer değiştirme $y(t)$ 'dir. Kütle-yay-damper sistemi Şekil 3.4'te gösterilmektedir.



Şekil 3.4. Kütle-yay-damper sistemi

Kütle yay damper sistemi, Denklem 3.2'de sürekli zamanlı ikinci dereceden diferansiyel denklem olarak ifade edilmektedir.

$$m \frac{d^2y}{dt^2} + c \frac{dy}{dt} + ky(t) = F(t) \quad (3.2)$$

Burada m kütle, k yay sertliği sabiti ve c sönümlenme katsayısıdır. Bu diferansiyel denklemin çözümü, m , c ve k sabit değerlerinin bilinen değerleri için, herhangi bir t zamanında $F(t)$ kuvvetinin bir fonksiyonu olarak, $y(t)$ kütlelerinin yer değiştirmesinin belirlenmesini sağlar.

Denklem 3.3'teki diferansiyel denklemin Laplace dönüşümü alınırsa kütle-yay-damper sisteminin transfer fonksiyonu modeli elde edilir. Denklem 3.3'de kütle-yay-damper sisteminin transfer fonksiyonu modeli gösterilmektedir (MATLAB, 2019).

$$G(s) = \frac{Y(s)}{F(s)} = \frac{1}{ms^2 + cs + k} \quad (3.3)$$

Sistem tanılama, belirli bir model yapısındaki ayarlanabilir parametrelerin değerlerini tahmin etmek için bir sistemden ölçülen giriş ve çıkış sinyallerini kullanır. Sistem tanılamada kullanılan veriler şu özelliklere sahip olmalıdır:

- i) Sistem çıkışını yeterince harekete geçiren giriş,
- ii) önemli zaman sabitlerini yakalamaya yetecek kadar veri ölçümü,
- iii) Gürültüden arındırılmış algılayıcı verisi,
- iv) Ölçülen veriler uygun örnekleme frekansı (Ljung, 2012; MATLAB, 2019).

Model yapısı, bilinmeyen parametreleri içeren girdi ve çıktı değişkenleri arasındaki matematiksel bir ilişkidir. Model yapı örnekleri, ayarlanabilir kutuplu ve sıfırlı transfer fonksiyonları, bilinmeyen sistem durum uzay denklemleri ve doğrusal olmayan parametrelili fonksiyonlardır. Denklem 3.4'te parametreleri bilinmeyen kutuplu sıfırlı bir transfer fonksiyonu gösterilmektedir.

$$G(s) = \frac{Y(s)}{F(s)} = \frac{a_0 + a_1s + a_2s^2}{1 + b_1s + b_2s^2} \quad (3.4)$$

a_0, a_1, a_2, b_1, b_2 ayarlanacak parametrelerdir.

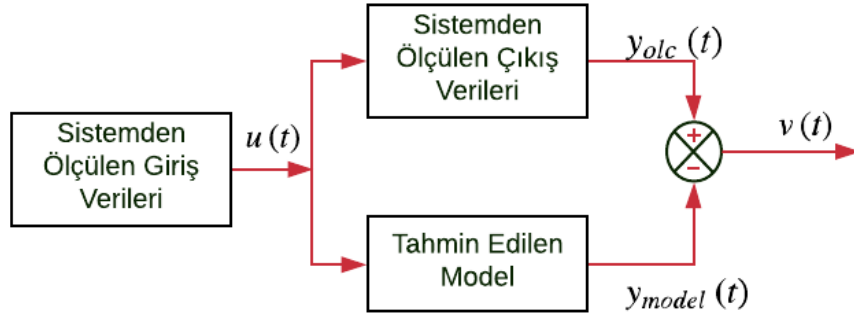
Sistem tanılama, model çıkışı ile ölçülen cevap arasındaki hatayı en aza indirerek model parametrelerini tahmin eder. Doğrusal model çıkışı Denklem 3.5'te gösterilmektedir. Burada $G(s)$ sistemin transfer fonksiyonunu, $U(s)$ Laplace uzayında sistem girişini ifade etmektedir. Model çıkışı $y_{model}(t)$, $G(s)U(s)$ çarpımının ters Laplace dönüşümü alınarak elde edilir.

$$y_{model}(t) = L^{-1}(G(s)U(s)) \quad (3.5)$$

Ölçülen cevap ile model çıkışı arasındaki fark Denklem 3.6'da gösterilmektedir.

$$v(t) = y_{olc}(t) - y_{model}(t) \quad (3.6)$$

Burada $v(t)$ ölçülen çıkış değeri ile model çıkışı arasındaki farkı, $y_{olc}(t)$ sistemden ölçülen çıkış verisini ve $y_{model}(t)$ ise model çıkışını ifade eder. Sistem tanılama sistem şeması Şekil 3.5'te gösterilmektedir (Ljung, 2012).

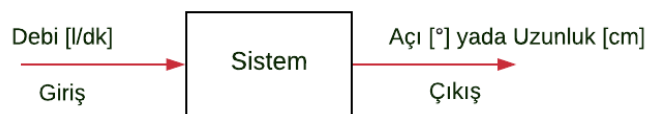


Şekil 3.5. Sistem tanılama şematığı

Bu çalışmada kullanılan sistemde kumandanadan alınan veriler mikro kontrolcü aracılığıyla valfin maksimum ve minimum debisine göre oranlanır. Anlamalı hale getirilen kumanda verisi bu uygulamada debi talebi olarak ifade edilmiştir. Debi talebi valf sistemine gönderilir. Valf sisteminin bu talebe verdiği cevap debi çıkışıdır. Valf sistemi hidrolik silindir ya da hidromotora debiyi gönderir. Gönderilen debi aracılığıyla hidrolik silindir uzama ya da kısılma hareketi, hidromotor ise dönüş hareketi yapar. Mekanik sistemde meydana gelen değişiklikler açı veya uzunluk algılayıcıları aracılığıyla zamana göre kaydedilir. Şekil 3.6'da giriş ve çıkış verileri alınırken kullanılan açık sistem şeması, Şekil 3.7'de kapalı sistem şeması gösterilmektedir.



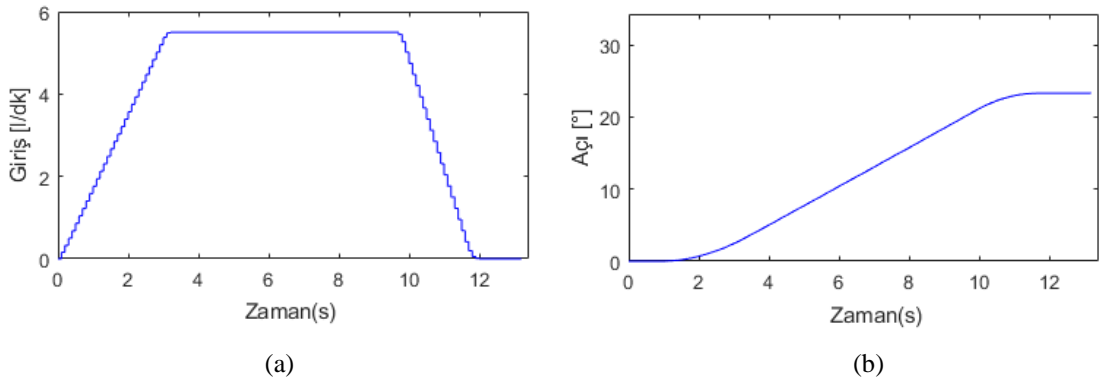
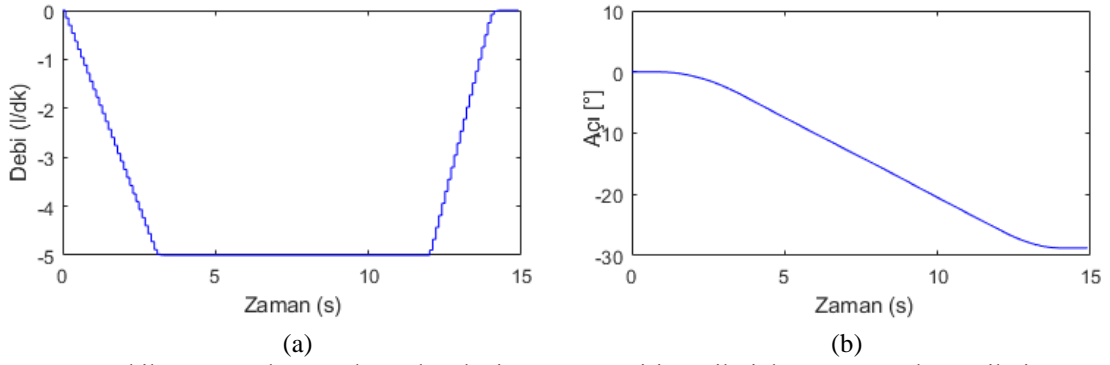
Şekil 3.6. Veri toplama için kullanılan sistemin açık şeması

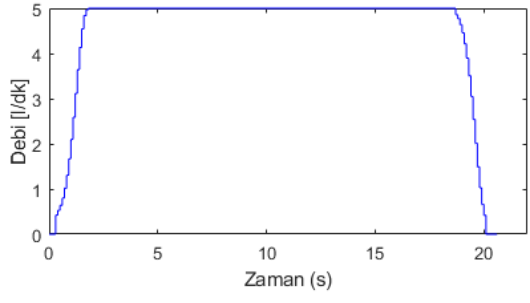


Şekil 3.7. Veri toplama için kullanılan sistemin kapalı şeması

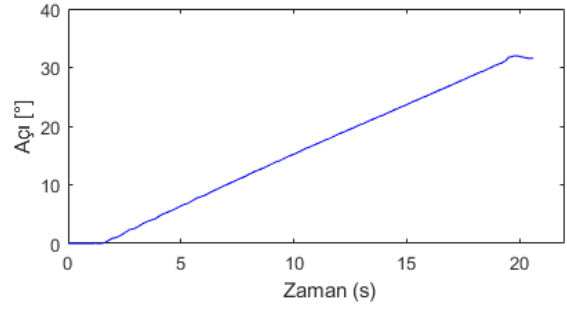
MSI'da kullanılacak olan veriler hafızaya alınırken başlangıç koşullarının sıfır olmasına dikkat edilir. Başlangıç koşulları sıfır değilse, başlangıç koşulları sisteme girilir ve bu değerler dikkate alınarak sistem transfer fonksiyonu elde edilir.

Ölçülen giriş ve çıkış değerleri model oluşturmada kullanılırlar. Fakat sistem algılayıcıları hata yapabilir ya da veriler yanlış kaydedilebilir. Ölçülen giriş ve çıkış değerleriyle model elde edildikten sonra, elde edilen modelin doğruluğunu test etmek amacıyla sistemden alınan ikinci bir sistem verisiyle model test edilir. Elde edilen sonuçlar arasındaki fark kabul edilebilir düzeyde ise elde edilen model kullanılabilir. Şekil 3.8-3.21'de sistemden ölçülen giriş ve çıkış verilerinin zamana göre değişimleri gösterilmektedir.



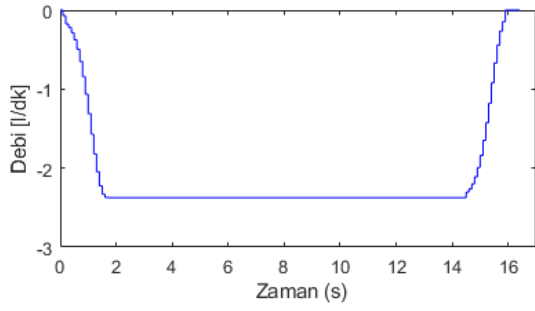


(a)

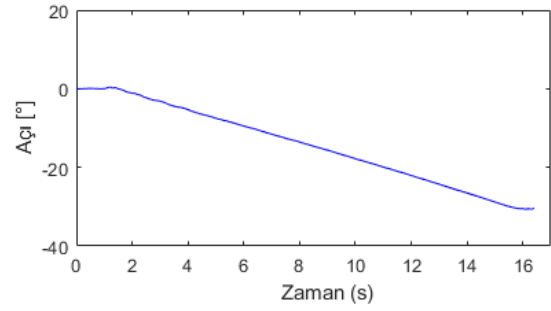


(b)

Şekil 3.10. a. Ana Bom kaldırma hareketi (ABKH) giriş verileri, b. ABKH çıkış verileri

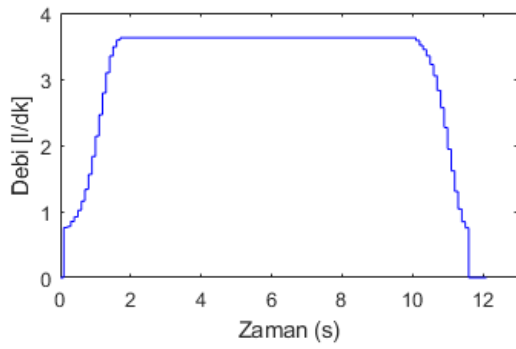


(a)

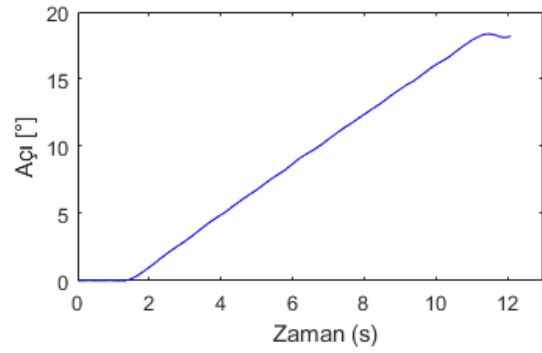


(b)

Şekil 3.11. a. Ana Bom indirme hareketi (ABİH) giriş verileri, b. ABİH çıkış verileri

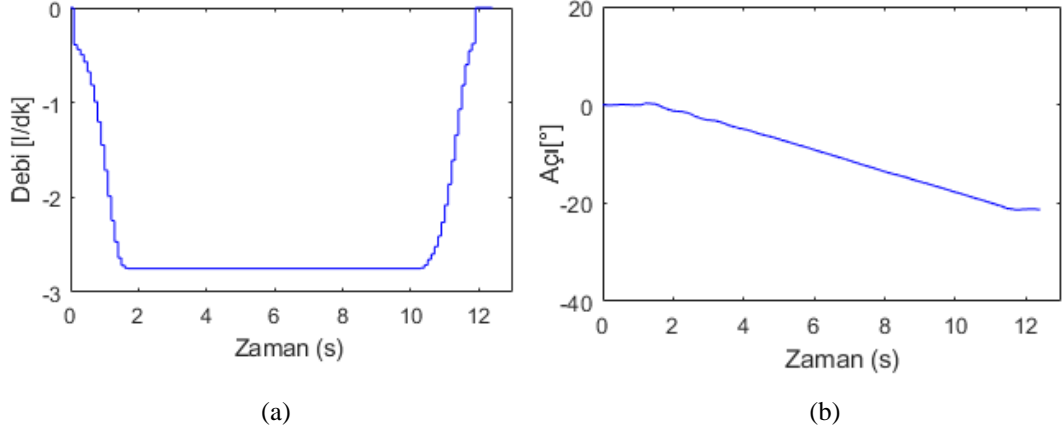


(a)

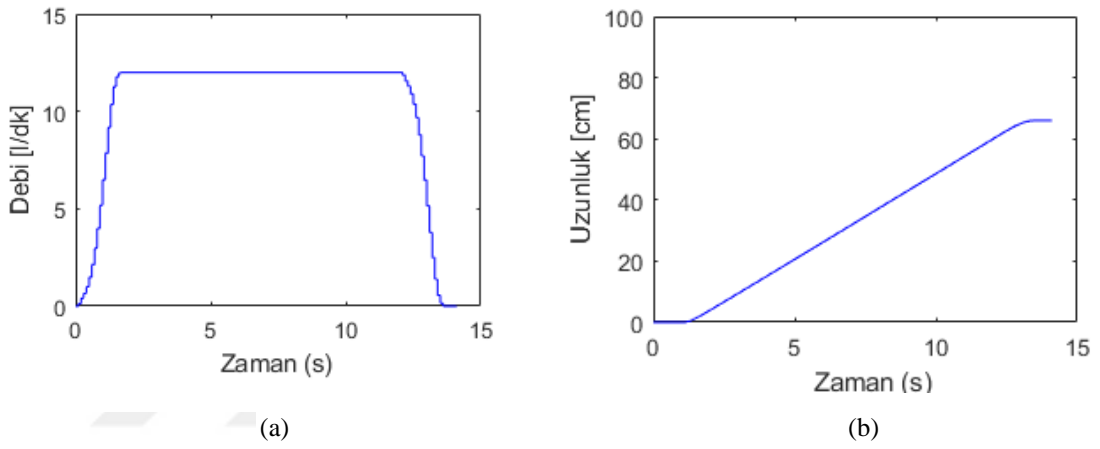


(b)

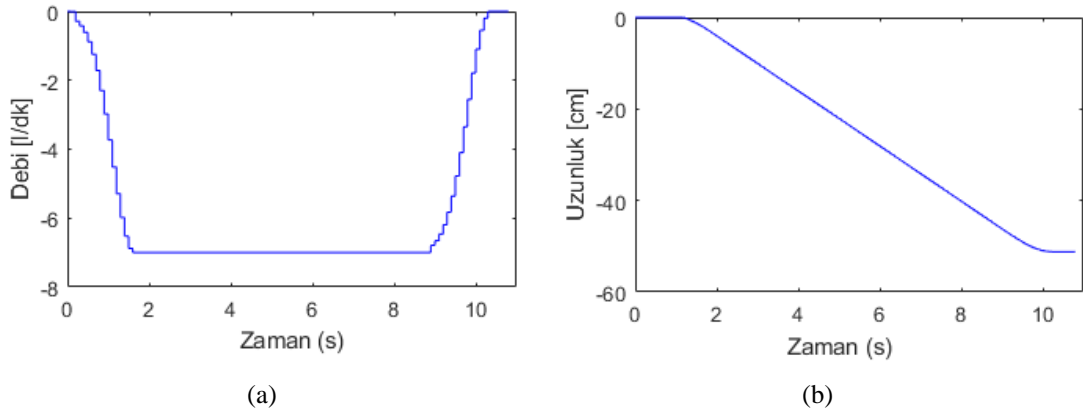
Şekil 3.12. a. 1. Kırma kaldırma hareketi (BKKH) giriş verileri, b. BKKH çıkış verileri



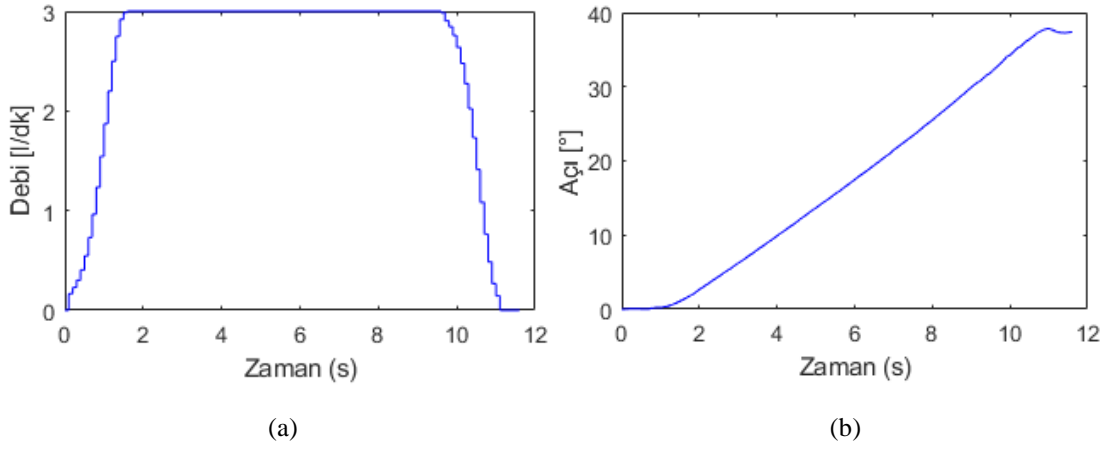
Şekil 3.13. a. 1. Kıırma indirme hareketi (BKİH) giriş verileri, b. BKİH çıkış verileri



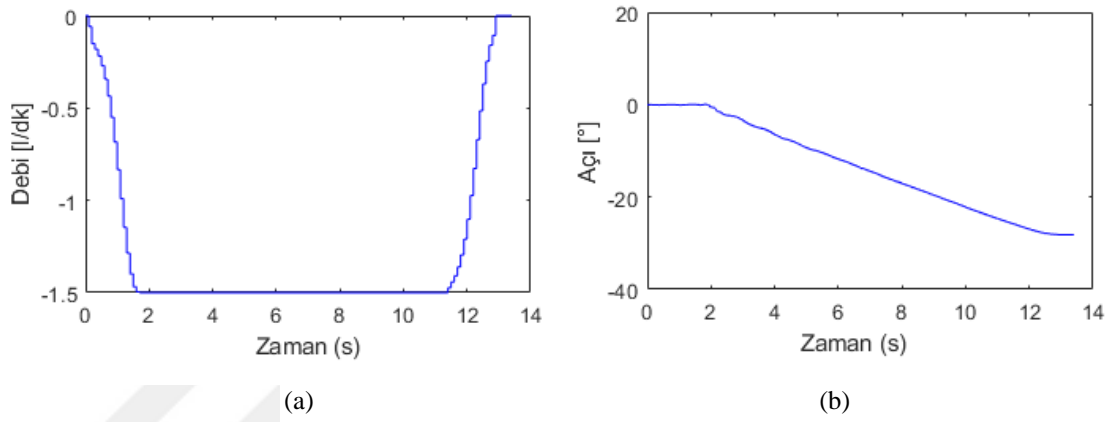
Şekil 3.14. a.1. Kıırma uzatma bom hareketi (BKBUH) giriş verileri, b. BKBUH çıkış verileri



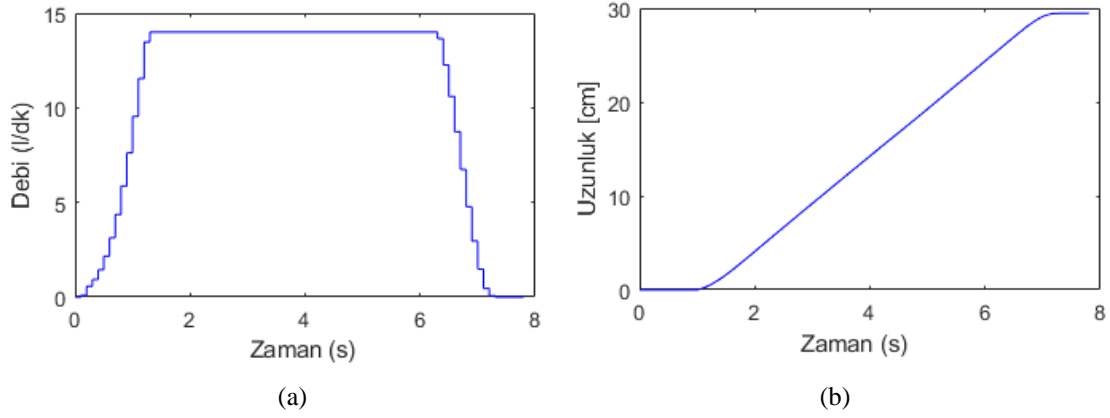
Şekil 3.15.a.1. Kıırma bom çekme hareketi (BKÇH) giriş verileri, b. BKÇH çıkış verileri



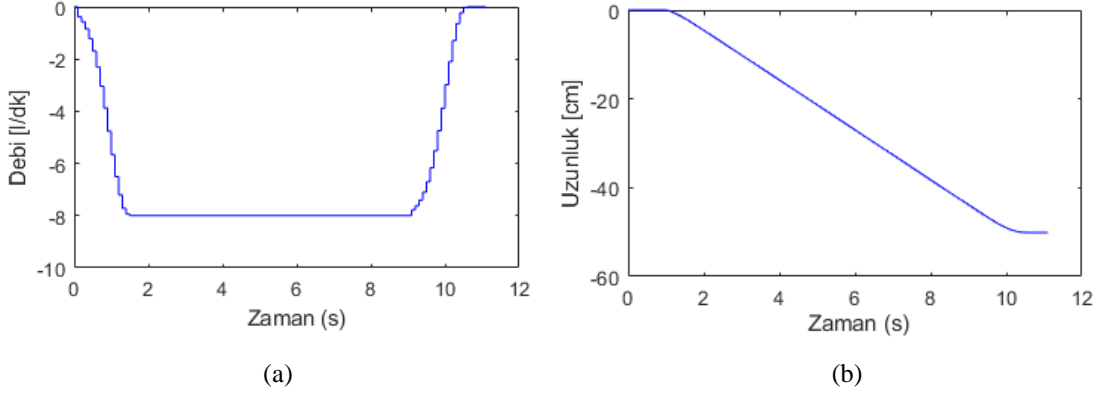
Şekil 3.16. a.2. Kırma kaldırma hareketi (İKKH) giriş verileri, b. İKKH çıkış verileri



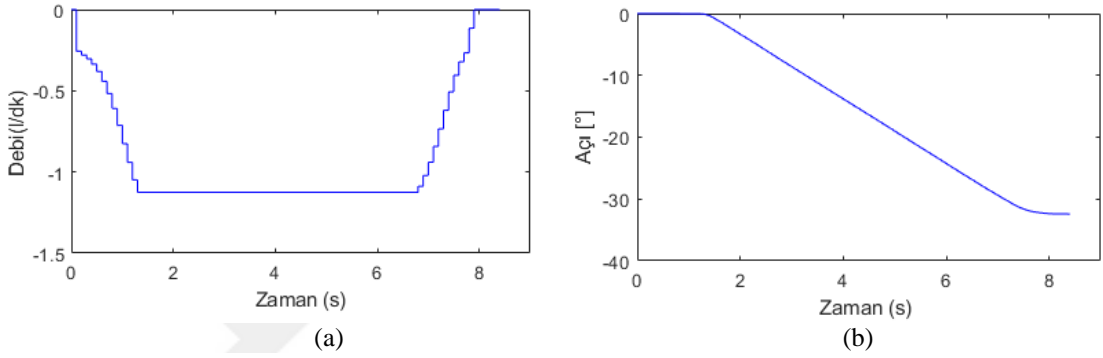
Şekil 3.17. a.2. Kırma indirme hareketi (İKİH) giriş verileri, b. İKİH çıkış verileri



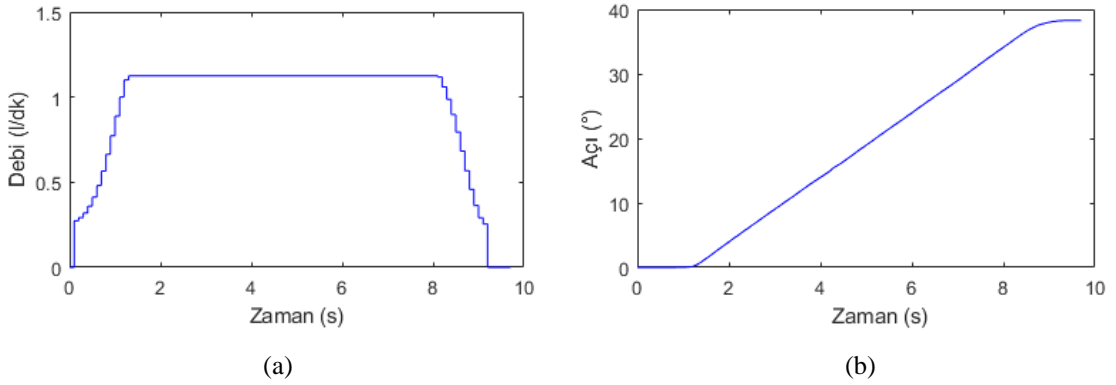
Şekil 3.18. a.2. Kırma bom uzatma hareketi (İKBUH) giriş verileri, b.2. İKBUH çıkış verileri



Şekil 3.19. a.2. Kırma bom çekme hareketi (İKBÇH) giriş verileri, b.2. İKBÇH çıkış verileri



Şekil 3.20. a. Ataçman sağa dönüş hareketi (ASADH) giriş verileri, b. ASADH çıkış verileri



Şekil 3.21. Ataçman sola dönüş hareketi (ASODH) giriş verileri, b. ASODH çıkış verileri

Bu çalışmada kullanılan hidrolik robot manipülatörde her bir serbestlik derecesi kaldırma indirme (açısal) hareketi ya da uzatma çekme (doğrusal) hareketi yapmaktadır. Debi farklılıklarından her bir hareket (ör. uzatma hareketi, çekme hareketi) bir sistem olarak ele alınmıştır.

Sistem tanımlanırken robot manipülatör yüksüz olarak ele alınmış, basınç değişiklikleri dikkate alınmamıştır.

3.3. Sezgisel Arama Algoritmaları

Sezgisel algoritmalar, optimizasyon problemlerini çözmek için doğadaki hayvanların, bireylerin, canlıların hareketlerinden esinlenerek oluşturulan algoritmalarıdır. Temel sezgisel arama algoritmaları kronolojiye göre Şekil 3.22’de gösterilmektedir (Gandomi ve ark., 2013).



Şekil 3.22. Kronolojiye göre sezgisel algoritmalar

3.3.1. Genetik algoritma

John Holland ve arkadaşları tarafından 1960-1970'li yıllarda geliştirilen GA Darwin'in doğal seçim prensibine dayanan küresel bir arama algoritmasıdır (Konak ve ark., 2006). GA'da kromozomlar bireyleri temsil eder. Kromozomlar gerçek sayılar ve birler ve sıfırlar ile gösterilebilir. Doğal seleksiyona göre doğadaki güçlü bireylerin başarılı olması daha olasıdır. Yaşam alanına uyum sağlamakta başarılı olan bireyler nesillerini devam ettirirler. Bu prensipten yola çıkılarak GA'da da nispeten daha iyi amaç fonksiyonu değerine sahip olan bireyler yeni nesil oluşturmak için görevlidirler. Bireyin sorun alanında gösterdiği iyi performans bireyin uygunluk ölçüsüdür ve uygunluk değeri yüksek olan bireyler neslini devam ettirir (Lin ve Liu, 2010; Köse, 2017).

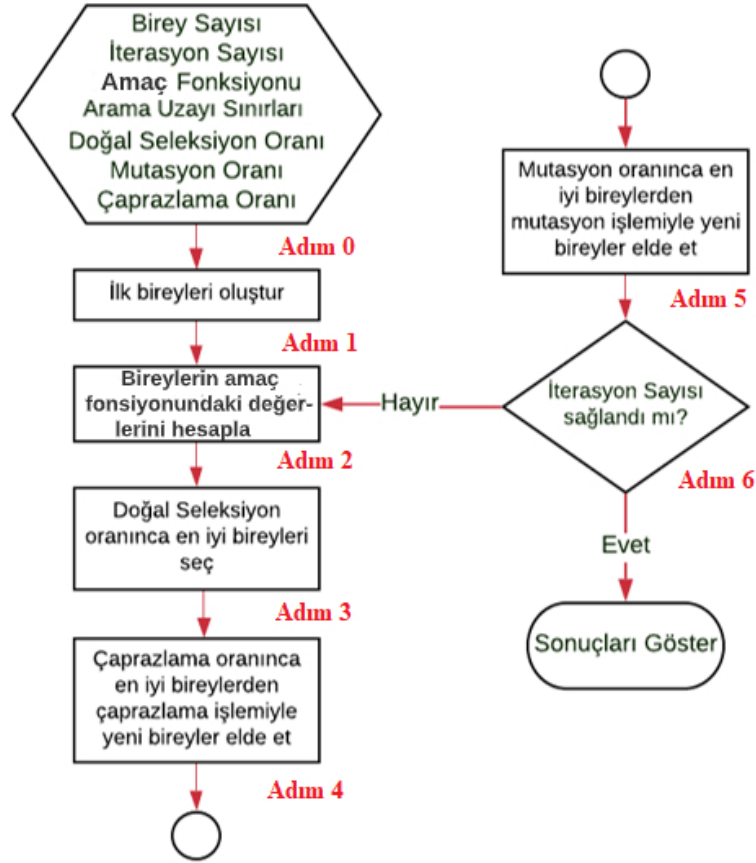
GA arama uzayı kısıtlarınca rastgele seçilen bireylerin amaç fonksiyonu değerlerinin hesaplanmasıyla başlar. Bireyler üç işleme tabi olur: Seçim, çaprazlama ve mutasyon. Seçim, rastgele dağıtılan bireylerin seçim oranına göre en iyi amaç fonksiyonu değerine sahip olanlarının sonraki nesle değişiklik yapılmadan aktarılmasıdır. Çaprazlama, uygunluk değeri yüksek olan ve seçimle belirlenen bireylerin çaprazlama oranına göre birbiriyle gen parçalarını karşılıklı olarak değiştirmeleridir. Mutasyon ise bireyin kromozomunun bir noktasında rastgele değişimler oluşturarak yeni bireylerin meydana getirilmesidir. Popülasyonlardaki çeşitlilik mutasyon aracılığıyla sağlanmaktadır P_M çok büyük seçilirse algoritma rastgele arama metodu haline gelir. Çok küçük seçilirse popülasyondaki çeşitlilik sağlanamaz (Elbayomy ve ark., 2008; Lin ve Liu, 2010; Köse, 2017).

GA'da zamanla nispeten uygunluk değeri düşük olan bireylerin yok olduğu, uygunluk değeri yüksek olan bireylerin arttığı gözlenir. Mutasyon sayesinde arama uzayında rastgele arama(keşif) yapılmaktadır ve algoritmanın yerel en iyi değerlere takılı kalmaktan kurtulması amaçlanmaktadır (Karaboğa, 2017). Şekil 3.23'te GA algoritma akış şeması ve Çizelge 3.4'te algoritma adımları gösterilmektedir.

Çizelge 3.4. GA Algoritma Adımları

Adım Numarası	Açıklama
Adım 0	Popülasyondaki birey sayısı, algoritma iterasyon sayısı, amaç fonksiyonu, arama uzayı, seçim oranı, çaprazlama oranı ve mutasyon oranı belirlenir.
Adım 1	n adet birey sayısına sahip K_p , K_d , K_i değerleri rastgele oluşturulur. Her bir $[K_p, K_d, K_i]$ dizisi bir birey olarak adlandırılır.
Adım 2	Her bir bireyin amaç fonksiyonundaki değeri hesaplanır.
Adım 3	Doğal seleksiyon oranınca amaç fonksiyonu değeri en iyi olan bireyler seçilir.

- Adım 4 Çaprazlama oranınca 3.adımda seçilmiş bireylerden yeni bireyler elde edilir.
 Adım 5 Mutasyon oranınca seçilen 3.adımda seçilen bireylerin kromozomlarında rastgele değişimler yapılarak yeni bireyler elde edilir.
 Adım 6 Durdurma kriterinin sağlanıp sağlanmadığı kontrol edilir. Sağlanmışsa algoritma bitirilir ve en uygun birey belirlenir. Sağlanmamışsa Adım 2'ye gidilir ve yeni bireylerle algoritma devam ettirilir.



Şekil 3.23. GA algoritma akış diyagramı

3.3.2. Parçacık sürü optimizasyonu

PSO, Eberhart ve Kennedy (1995) tarafından hayvan sürülerinin yiyecek arama, güvenlik ve göç hareketlerinden esinlenerek geliştirdiği sezgisel algoritma metodudur. Sürüdeki hayvanlar bahsedilen durumlarda konumları en iyi olan parçacığa yönelirler (Clerc ve Kennedy, 2002). Konumu en iyi olan parçacığa yaklaşma hızı parçacığın hız kapasitesine göre değişmektedir. Bu nedenle her bir parçacığın hızı rastgele belirlenmektedir (Özsağlam ve Çunkaş, 2008).

Diğer sezgisel arama algoritmalarında olduğu gibi PSO'da da parçacıklar arama uzayına, arama uzayı sınırları dikkate alınarak rastgele dağıtılır. En iyi amaç fonksiyonu değerine sahip olan parçacık diğer parçacıklara haber verir (Erkol, 2017). Diğer

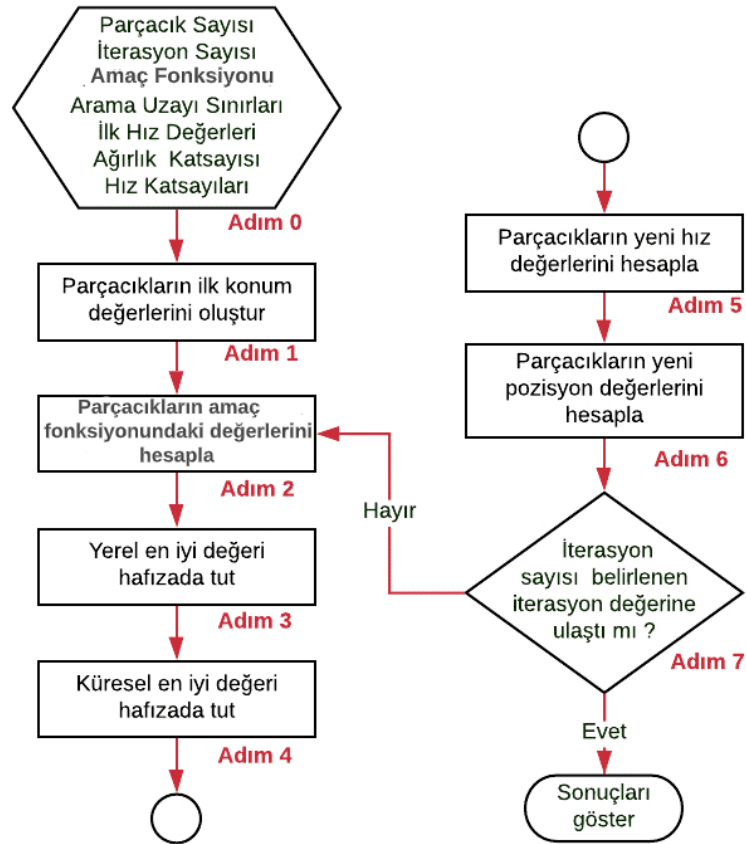
parçacıklar Denklem 3.7’de formülü gösterilen hızla konumunu en iyi olan parçacığa doğru yöneltir (Gaing, 2004).

V_{iy} i. parçacığın yeni hızı, w atalet ağırlık faktörü, V_i i. parçacığın hızı, c_1, c_2 hızlanma katsayıları, $rand$ 0-1 arasında rastgele sayı, P_y parçacıkların i. iterasyondaki en iyi(yerel en iyi) konumu, X_i i. parçacığın bulunduğu mevcut konumu, P_g parçacıkların o ana kadarki tüm iterasyonlardaki en iyi(küresel en iyi) konumudur. Denklem 3.8’de parçacıkların yeni konumu gösterilmektedir (Gaing, 2004; Çavuşlu ve ark., 2010).

$$V_{iy} = wV_i + c_1rand(P_y - X_i) + c_2rand(P_g - X_i) \quad (3.7)$$

$$X_{iy} = X_i + V_i \quad (3.8)$$

PSO Algoritma akış şeması Şekil 3.24’de ve algoritma adımları Çizelge 3.5’te gösterilmektedir.



Şekil 3.24. PSO algoritma akış diyagramı

Çizelge 3.5. PSO algoritma adımları

Adım Numarası	Açıklama
Adım 0	Popülasyonda kaç adet parçacık olacağı, iterasyon sayısı, amaç fonksiyonu, arama uzayı, parçacıkların ilk hız değeri, hızlanma katsayıları ve atalet ağırlık değeri belirlenir.
Adım 1	n adet eleman sayısına sahip K_p , K_d , K_i değerleri rastgele oluşturulur. Her bir $[K_p, K_d, K_i]$ dizisi bir parçacık olarak adlandırılır.
Adım 2	Her bir parçacığın amaç fonksiyonundaki değeri hesaplanır.
Adım 3	Amaç fonksiyonu değeri en düşük olan parçacık yerel en iyi değer olarak hafızada tutulur.
Adım 4	Her bir döngüdeki en iyi değerler karşılaştırılarak küresel en iyi değer belirlenir ve hafızada tutulur.
Adım 5	Denklem 1 kullanılarak parçacıkların yeni hızları hesaplanır.
Adım 6	Denklem 2 kullanılarak parçacıkların yeni konumu bulunur.
Adım 7	Durdurma kriterinin sağlanıp sağlanmadığı kontrol edilir. Sağlanmışsa algoritma bitirilir ve en uygun parçacık belirlenir. Sağlanmamışsa elde edilen yeni bireyler amaç fonksiyonda yerine konur ve döngüye devam edilir.

3.3.3. Arı algoritması

AA, arıların besin ararken ki davranışlarını ele alan bir KA algoritmasıdır. Bal arılarının davranışlarını modellerken üç ana unsur karşımıza çıkmaktadır: Yiyecek kaynakları, işçi arılar, işçi olmayan arılar. Yiyecek kaynağının verimi, yuvaya olan mesafesi, kaynağın zenginliği ve yiyeceği çıkarmak için gerekli olan enerjiyle ölçülebilir. İşçi arılar, kâşif arılar tarafından gelen bilgiye göre yiyecek kaynaklarına dağılırlar ve buldukları konumun zenginliği hakkında kovandaki arılara bilgi verirler (Karaboga, 2013). İşçi olmayan arılar iki türdür: Gözcü arılar ve kâşif arılar. Gözcü arılar işçi arıların getirdiği bilgiye göre yiyecek kaynağı etrafında yoğunlaşırlar. Kâşif arıların görevi ise yeni yiyecek kaynağı aramaktır. Kâşif arıların sürüdeki miktarı %5-%10 arasında değişmektedir (Karaboga ve Basturk, 2008).

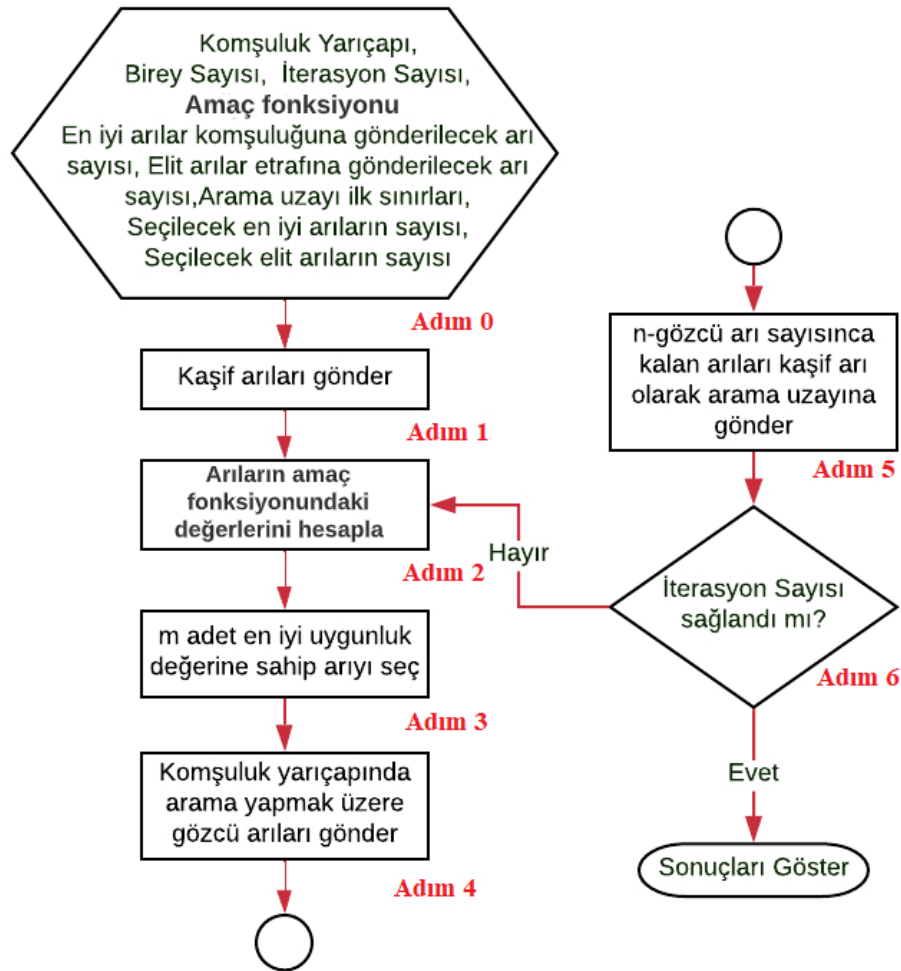
Arılar buldukları konumun uygunluğunu kovandaki dans pistinde dans ederek anlatır. Kovandaki arılar bu uygunluk değerine göre nereye gideceklerine karar verirler. Uygunluk değerine oranla arılar arama uzayına tekrar dağıtılır. Besin kaynağı bulduktan sonra arı besin kaynağının konumunu hafızasına alır ve besin kaynağını sömürmeye başlar. Besin sömürmeye başlayan arı işçi arı haline gelir. Daha sonra kapasitesi kadar besini alan arı kovana döner. Kovana dönen arının üç seçeneği mevcuttur (Karaboga, 2013):

- i. Gıda kaynağını terk ederek gözcü haline gelebilir.
- ii. Dans edip arkadaşlarını da yiyecek kaynağına götürebilir.

iii. Tek başına besin kaynağına geri dönebilir.

Algoritma başlangıcında probleme göre değişken n sayıda kâşif arı çözüm arama uzayına rastgele dağıtılmaktadır. Dağıtılan arıların her birinin konumu amaç fonksiyonunda yerine yazılır ve hata değerleri hesaplanır. Elde edilen hata dizisi küçükten büyüğe doğru sıralanmaktadır. m adet en düşük amaç fonksiyonu değerine sahip arı seçilir. m adet seçilen arıların etrafına amaç fonksiyonu değerine ters orantılı olarak komşuluk yarıçapı içinde yeni arılar gönderilir. En düşük amaç fonksiyonu değerine sahip olan arı yerel en iyi değer olarak atanır. İlk iterasyon da yerel en iyi değer küresel en iyi değer olarak atanır. Sonraki iterasyonlar da küresel en iyi ile yerel en iyi karşılaştırılır. Yerel en iyi daha düşük değere sahipse küresel en iyi olarak atanır. Kalan arılar kâşif arı olarak arama uzayına rastgele dağıtılır (Erdemir ve Kalyoncu, 2015).

Arı algoritması, algoritma akış diyagramı Şekil 3.25'te ve algoritma adımları Tablo Çizelge 3.6'da gösterilmektedir (Erdemir ve Kalyoncu, 2015) .



Şekil 3.25. AA algoritma akış şeması

Çizelge 3.6. AA algoritma adımları

Adım Numarası	Açıklama
Adım 0	Popülasyonda kaç adet arı olacağı belirlenir. İterasyon sayısı, amaç fonksiyonu, arama uzayı sınırları, ilk iterasyondan sonrası için gözcü arı sayısı, kâşif arı sayısı, en iyi uygunluk bölgesi sayısı, en iyi uygunluk bölgeleri arasında elit bölge sayısı, en iyi uygunluk bölgelerine gönderilecek gözcü arı sayısı, elit bölgelere gönderilecek gözcü arı sayısı ve arama bölgesi yarıçapı belirlenir.
Adım 1	n adet eleman sayısına sahip bir K_p , K_d , K_i değerlerinden oluşan kâşif arı popülasyonu arama uzayı kısıtlarına uyularak rastgele oluşturulur. Her bir $[K_p, K_d, K_i]$ dizisi bir arı olarak adlandırılır.
Adım 2	Her bir arının amaç fonksiyonundaki değeri hesaplanır ve küçükten büyüğe doğru sıralanır.
Adım 3	Amaç fonksiyonu değeri en düşük olan m adet arı belirlenir. Bu m adet arılar arasından e adet en küçük değere sahip olan elit bölge olarak adlandırılır. Elit bölgenin çevresine gönderilen gözcü arı sayısı elit olmayan arıların çevresine gönderilen gözcü arı sayısından daha fazla olmalıdır.
Adım 4	Gözcü arılar belirlenen m adet bölgeye komşuluk yarıçapına uygun olarak gönderilir.
Adım 5	Algoritmanın yerel maksimum ve minimumlara takılmaması için rastgele arama yapan kâşif arılar(n – gözcü arı sayısı kadar) arama uzayına kısıtlara uyularak gönderilir.
Adım 6	Durdurma kriterinin sağlanıp sağlanmadığı kontrol edilir. Sağlanmışsa algoritma bitirilir ve en uygun arı belirlenir. Sağlanmamışsa elde edilen yeni arılar amaç fonksiyonda yerine konur ve döngüye devam edilir.

3.3.4. Büyük patlama büyük çöküş algoritması

BPBÇ algoritması evrenin varoluş teorisi olan “Büyük Patlama Büyük Çöküş” teorisinden esinlenerek Erol ve Eksin (2006) tarafından oluşturulmuştur (Arşan, 2018). Teoriye göre evrenin oluşması iki safhadır: Büyük patlama ve büyük çöküş. Büyük patlama safhasında büyük bir enerji yayılımı meydana gelir. Bu nedenle düzensizlik oluşur. Bu düzensizliğin rastgele oluştuğu ifade edilmiştir. Büyük çöküş safhasında düzensiz halde rastgele dağılan enerji yüklü parçacıklar bir ağırlık merkezine doğru sürüklenir (Ersöz, 2011).

Erol ve Eksin (2006) tarafından önerilen algoritma da büyük patlama safhasında parçacıklar arama yapılacak uzaya rastgele dağıtılır. Büyük çöküş safhasında ise rastgele dağıtılan parçacıklar bir kütle merkezi etrafında yoğunlaşır. Parçacıkların buldukları konum ve amaç fonksiyonu değeri bu kütle merkezinin konumunu belirler.

Arama uzayına rastgele dağıtılan parçacıkların kütle merkezi X_m , popülasyondaki parçacık sayısı n , i 'inci parçacığın amaç fonksiyonundaki değeri f_i , i 'inci parçacığın ya da bireyin konumu X_i 'dir. Parçacıkların kütle merkezi X_m Denklem 3.9'daki formül kullanılarak hesaplanır.

$$X_m = \frac{\sum_{i=1}^n \frac{1}{f_i} X_i}{\sum_{i=1}^n \frac{1}{f_i}} \quad (3.9)$$

Büyük çöküş fazından sonra yeniden büyük patlama fazına geçilir. Büyük patlama fazında bu kez parçacıklar normal dağılıma göre kütle merkezi etrafına rastsal bir şekilde dağıtılır. X_i^{yeni} ; X_m etrafına gönderilen yeni parçacıklar, σ standart sapma olarak ifade edilir. Denklem 3.10'daki formül parçacıkların yeni konumu gösterir (Ersöz, 2011).

$$X_i^{yeni} = X_m \pm \sigma \quad (3.10)$$

n ; iterasyon sayısı, r ; 0 ile 1 arasında belirlenen rastgele bir sayı, t ; arama uzayı daraltma katsayısı, k iterasyon sayısı, X_{maks} arama uzayı üst değer sınırı, X_{min} arama uzayı alt değer sınırıdır. Denklem 3.11'deki formül standart sapma değerini ifade eder (Ersöz, 2011).

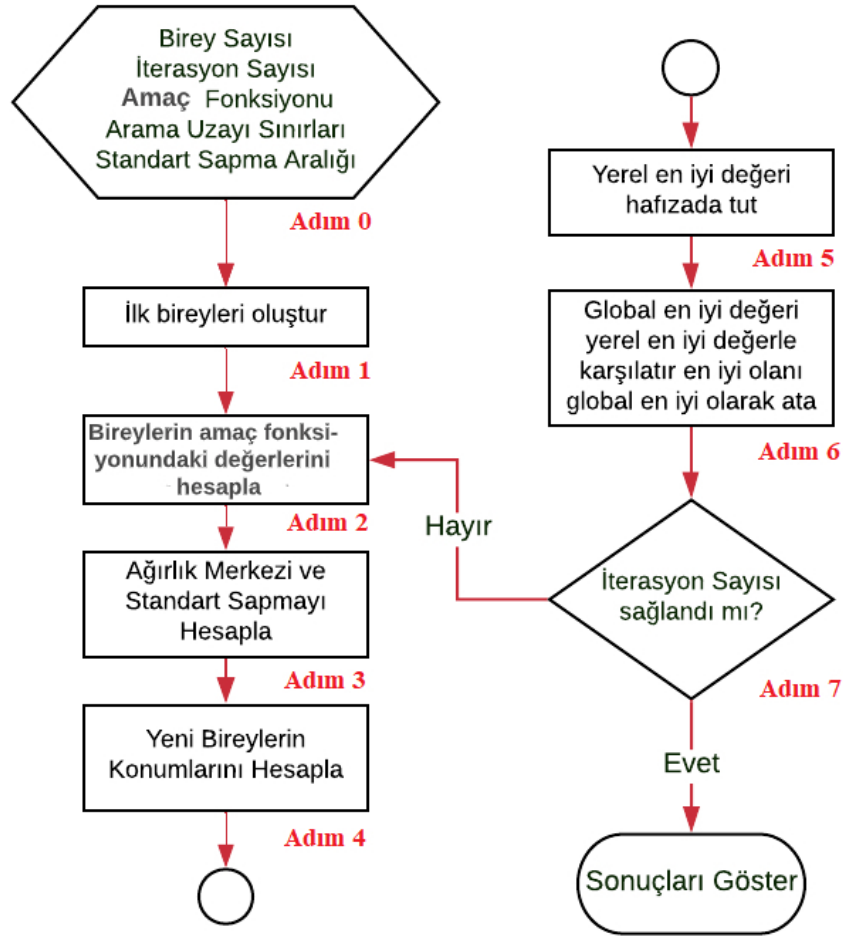
$$\sigma = \frac{0.5(X_{min} + r(X_{maks} - X_{min}))}{1 + \frac{k}{t}} \quad (3.11)$$

Şekil 3.26'da BPBÇ Algoritma şeması, Çizelge 3.7'de BPBÇ algoritma adımları gösterilmektedir.

Çizelge 3.7. BPBÇ algoritma adımları

Adım Numarası	Açıklama
Adım 0	Popülasyonda kaç adet birey olacağı, iterasyon sayısı, amaç fonksiyonu, arama uzayı sınırları, arama uzayı daraltma katsayısı, Normal dağılım için standart sapma aralığı belirlenir.
Adım 1	n adet eleman sayısına sahip K_p , K_d , K_i değerleri rastgele oluşturulur. Her bir $[K_p, K_d, K_i]$ dizisi bir birey olarak adlandırılır.
Adım 2	Her bir bireyin amaç fonksiyonundaki değeri hesaplanır.
Adım 3	Bireylerin konumuna ve amaç fonksiyonundaki değerlerine göre ağırlık merkezi belirlenir.
Adım 4	Amaç fonksiyonundaki değeri en iyi olan değer yerel minimum olarak atanır ve hafızada tutulur.
Adım 5	Her bir döngüdeki en iyi değerler karşılaştırılarak küresel en iyi değer belirlenir ve hafızada tutulur.
Adım 6	Ağırlık merkezinde normal dağılıma göre yeni bireyler oluşturulur.

Adım 7 Durdurma kriterinin sağlanıp sağlanmadığı kontrol edilir. Sağlanmışsa algoritma bitirilir ve en uygun birey belirlenir. Sağlanmamışsa normal dağılıma göre elde edilen yeni bireyler amaç fonksiyonda yerine konur ve döngüye devam edilir.



Şekil.3.26. BPBC algoritma akış şeması

3.3.5. Guguk kuşu algoritması

GK Algoritması 2009 yılında Yang ve Deb tarafından ortaya atılmış guguk kuşlarının üremek için sergiledikleri davranışlardan esinlenen KA algoritmasıdır.(Köse, 2017) Birçok kuş çeşidi yumurtlamak için yuva yapmaktadır. Fakat bazıları yumurtalarını başka kuşların yuvasına bırakmaktadır. Bu tip kuşlara kuluçka asalağı (brood parasite) denmektedir. Guguk kuşları bu türün en çok bilinenlerindedir. Guguk kuşu yumurtasını bırakacağı yuvada bulunan yumurtaları taklit etmek zorundadırlar. Aksi halde yumurta yuvanın gerçek sahibi tarafında tespit edilir ve yok edilir (Yang, 2010; Yang, 2014; Yang ve Deb, 2014).

Guguk kuşları bu algorithmada iki tipte bulunmaktadır: Yetişmiş guguk kuşları ve yumurtalar. Yumurtalar yetişkin guguk kuşları tarafından belirlenen yuvalara bırakılır. Zamanla yuva sahibi kuş tarafından beslenip büyütülen kuş daha fazla yiyecek elde etmek için diğer kuşları yuvadan atabilir. Kendi hayatını idame ettirecek düzeye geldiğinde ise daha iyi bir bölgeye göç etmek isteyecektir. Guguk kuşları, göç sayesinde beslenmek ve üremek için en iyi ortamı bulmayı amaçlamaktadırlar (Yang, 2010; Rajabioun, 2011; Yang, 2014; Yang ve Deb, 2014).

Algoritmanın üç ana kabulü vardır (Yang, 2010; Iglesias ve ark., 2018).

- Her bir guguk kuşu taşıma sorunu nedeniyle sadece bir tane yumurta taşıyabilir ve o yumurtayı rastgele seçilen bir yuvaya bırakabilir.
- Bırakılan yumurta yuva sahibi tarafından tespit edilemiyorsa ve amaç fonksiyonundaki değeri düşükse bir sonraki nesle aktarılır.
- Yuva sahibi kuş yumurtayı P_a olasılıkla tespit edebilmektedir. Eğer yumurtayı tespit ederse yuvayı terk edebilir ya da guguk kuşu yumurtasını yuvadan atabilir.

Bu üç kabule dayanarak GKA sözde kodu, Çizelge 3.8 aracılığıyla özetlenebilir (Iglesias ve ark., 2018; Zeybekoğlu, 2018).

Çizelge 3.8. GKA sözde kodu

```

Amaç Fonksiyonu  $f(x), x = [x_1, x_2, x_3, \dots \dots x_d]^T$  .
N adet konak yuvası başlangıç pozisyonu belirle.
Bu yuvaların  $F_i$  amaç fonksiyonu değerlerini hesapla
while( $t < Jenerasyon\ sayisi$  ||  $durdurma\ kriteri$ )
Levy uçuşları aracılığıyla guguk kuşlarının konum değerlerini güncelle
 $F_j$  amaç fonksiyonu değerlerini hesapla.
n adet konak yuvası arasından rastgele yuva seç.
    if( $F_j < F_i$ )
        En düşük amaç fonksiyonu değerini değiştir.
    end
En kötü yuvaları  $P_a$  oranınca terk et. Yerlerine yenilerini bul.
İyi çözümleri hafızada tut.
Elde edilen çözümleri amaç fonksiyonu değerlerine göre küçükten büyüğe sırala ve en iyi çözümü bul
End while

```

(Iglesias ve ark., 2018; Zeybekoğlu, 2018)

GK Algoritması, N adet yuva pozisyonunun alt ve üst sınırlar dikkate alınarak rastgele belirlenmesiyle başlar. İlk popülasyon oluşturulduktan sonra üyeler daha iyi yerlerin var olup olmadığını aramalıdır. GKA küresel rastgele yürüyüş ve yerel rastgele yürüyüş metodları kullanılmaktadır. GKA da ilk olarak Denklem 3.13-3.18'de ifade edilen küresel rastgele yürüyüş uygulanarak yeni konuk yuva konumları elde edilir.

p_a oranınca en kötü sonuçların bulunduğu yuva konumları yok edilir. Eksik olan yuva konumları yerel rastgele yürüyüş ile belirlenir. Yerel rastgele yürüyüş ifadesi Denklem 3.12’de ifade edilmektedir.

$$x_i^{t+1} = x_i^t + \alpha s \otimes H(p_a - \varepsilon) \otimes (x_j^t - x_k^t) \quad (3.12)$$

x_j^t ve x_k^t rastgele permütasyonla belirlenen iki çözüm, P_a konak yuvasına bırakılan yumurtanın tespit edilme olasılığı, $H(P_a - \varepsilon)$; p_a olasılığından büyük olan değerlerin 1 küçük olan değerlerin 0 olduğunu belirten Heaviside adım fonksiyonu (birim fonksiyon), ε rastgele belirlenen sayıyı, s adım boyutunu ve α adım boyutu ölçekleme katsayısını ifade eder.

$$\phi = \left(\frac{\Gamma(1 + \beta) \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1 + \beta}{2}\right) \beta 2^{\frac{\beta-1}{2}}}\right)^{\frac{1}{\beta}} \quad (3.13)$$

$$u = \phi rand() \quad (3.14)$$

$$v = rand() \quad (3.15)$$

$$s = \frac{u}{|v|^{\frac{1}{\beta}}} \quad (3.16)$$

$$\zeta = 0.01s(x_i^t - x_{best}) \quad (3.17)$$

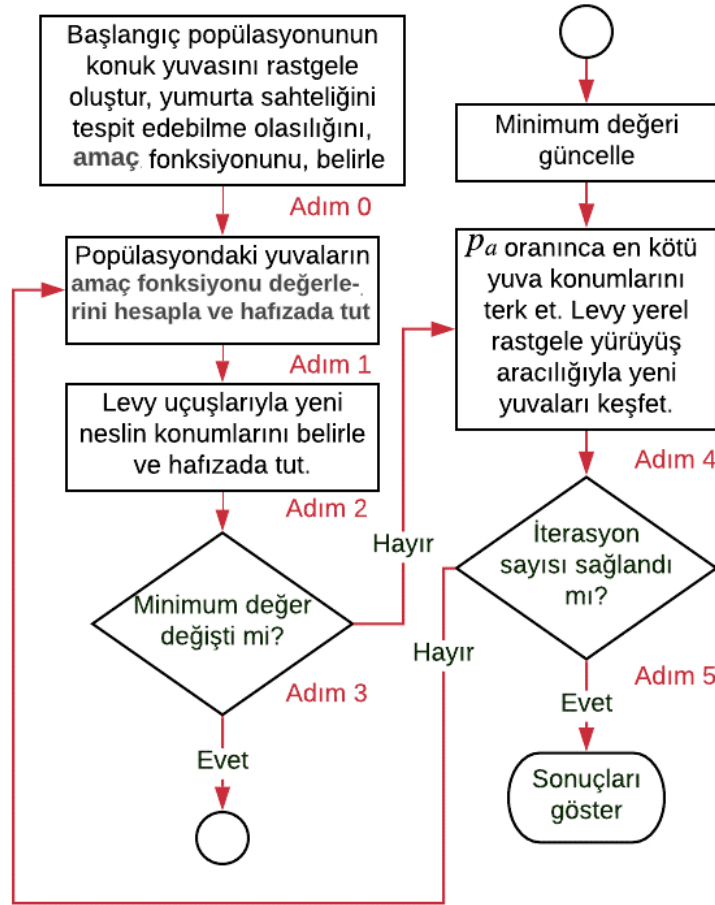
$$x_i^{t+1} = x_i^t + \zeta randn() \quad (3.18)$$

Levy uçuşlarıyla rasgele sayıların üretilmesi iki ana adımdan oluşur: ilk olarak, normal dağılıma göre rastgele bir yön seçilir; daha sonra, seçilen Levy dağılımını izleyen adımların sırası üretilir. Simetrik dağılımlar için Denklem 3.8’de ifade edilen Mantegna’nın algoritması kullanılabilir (Iglesias ve ark., 2018). Denklemde $\Gamma(1 + \beta)$ Gamma fonksiyonudur. Iglesias ve ark. (2018) ve Yang (2010) tarafından yapılan çalışmalarda $\beta = \frac{3}{2}$ olarak ele alınmıştır.

Mantegna’nın algoritmasında adım uzunluğu ifadesi s , Denklem 3.16’da gösterilmektedir. u ifadesi Denklem 3.14’te, v ifadesi ise Denklem 3.15’te gösterilmektedir. Adım uzunluğu, yuvanın mevcut konumu ve en iyi yuvanın konumu kullanılarak Denklem 3.17’de gösterilen adım boyutu hesaplanır. Denklem 3.18’de

yuvanın yeni konumunun ifadesi gösterilmektedir. α normal dağılımdan seçilen rastgele bir sayı, ζ adım boyutu ve x_i^{t+1} yeni yuvanın konumudur (Yang, 2010; Iglesias ve ark., 2018).

GKA algoritma akış diyagramı Şekil 3.27’de, algoritma adımları Çizelge 3.9’da gösterilmektedir (Barsainya ve ark., 2017).



Şekil 3.27. GKA algoritma akış şeması

Çizelge 3.9. GKA algoritma adım açıklamaları

Adım Numarası	Açıklama
Adım 0	Popülasyonda konak yuvası olacağı, iterasyon sayısı, amaç fonksiyonu, arama uzayı sınırları, yuva sahibi kuş tarafından sahte yumurtaların tespit edilebilme olasılığı belirlenir.
Adım 1	n adet eleman sayısına sahip K_p , K_d , K_i değerleri rastgele oluşturulur. Her bir $[K_p, K_d, K_i]$ dizisi bir yuva olarak seçilir. Mevcut yuva konumlarının amaç fonksiyonu değerleri belirlenir.
Adım 2	Küresel rastgele yürüyüş (Levy uçuşları) ile yeni yuva konumları belirlenir. Yeni yuva konumlarının amaç fonksiyonundaki değeri hesaplanır ve Adım 3’e gidilir.
Adım 3	Minimum değer değişip değişmediği kontrol edilir. Değişmişse minimum değer güncellenir. Değişmemişse Adım 4’e gidilir.

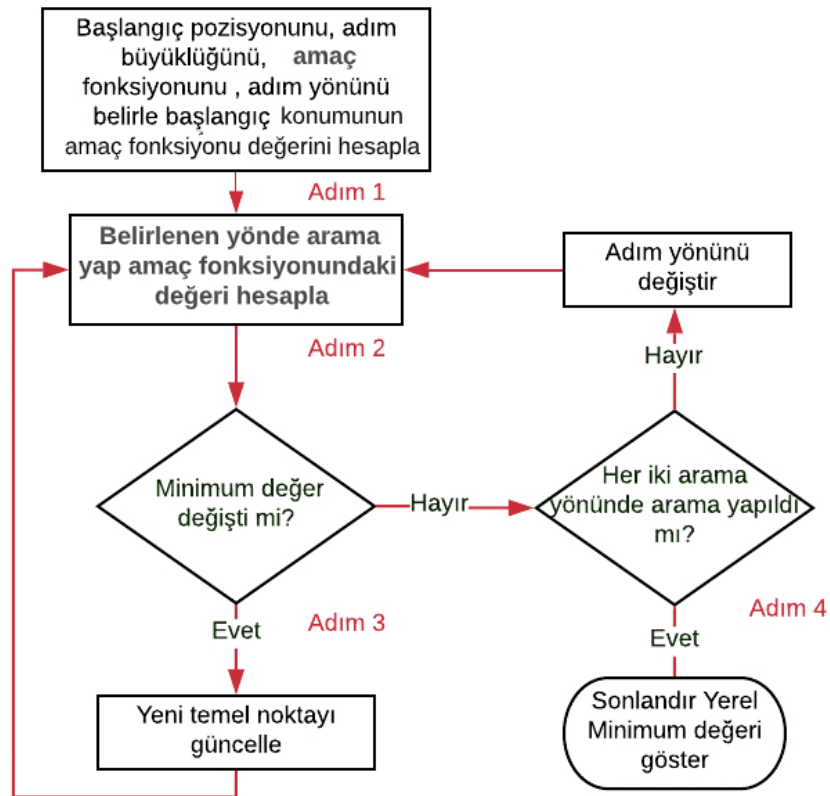
Adım 4	p_a oranınca en yüksek amaç fonksiyonu değerine sahip yuvaları terk et. Levy rastgele yerel yürüyüş ile eksik olan yeni yuvaları elde edilir ve Adım 5'e gidilir.
Adım 5	İterasyon sayısı veya durdurma kriteri sağlanmışsa algoritma sona erdirilir ve sonuçlar gösterilir. Sağlanmamışsa Adım 1'e gidilir.

3.4. Yerel Arama Algoritmaları

YA algoritmalarının sonuç bulma hızı sezgisel algoritmalara nispeten çok yüksektir. Bu çalışmada amaç fonksiyonunun türevine ihtiyaç duymadığı için TT ve HJ algoritmaları ele alınmıştır.

3.4.1. Tepe tırmanma algoritması

TT algoritması YA algoritmalarındandır. Kullanımının kolay olması ve hızlı sonuç vermesi nedeniyle tercih edilmektedir. TT algoritma akış şeması Şekil 3.28'de gösterilmiş algoritma adımları Çizelge 3.10'da açıklanmaktadır. (Gen ve Yun, 2006; Cervantes Escorcía ve ark., 2019).



Şekil 3.28. TT algoritma akış şeması

Çizelge 3.10. TT algoritma adımları

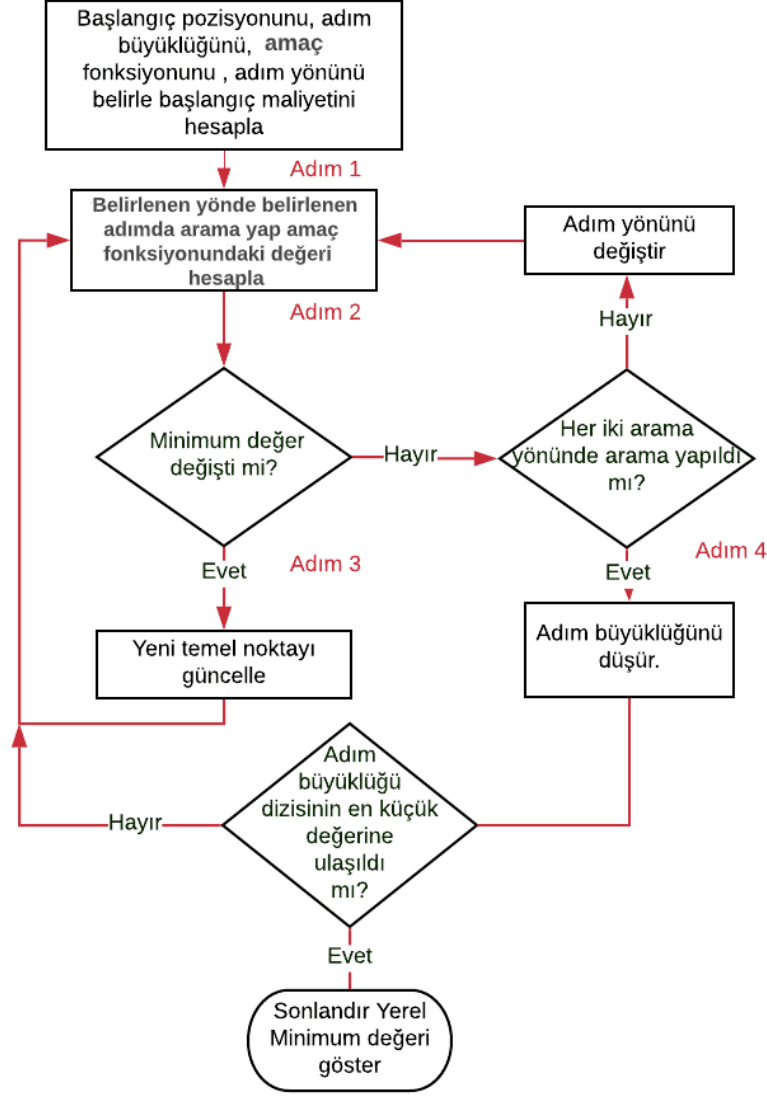
Adım Numarası	Açıklama
Adım 1	Arama için gerekli başlangıç pozisyonu, adım büyüklüğü, amaç fonksiyonu, adım yönü belirlenir ve amaç fonksiyonu değeri hesaplanır. Adım 2'ye gidilir.
Adım 2	Başlangıç pozisyonuna belirlenen yönde adım büyüklüğü eklenir. Elde edilen yeni pozisyonun amaç fonksiyonu değeri hesaplanır.
Adım 3	Yeni pozisyon ile temel noktanın amaç fonksiyonu değerleri karşılaştırılır, maliyet değeri azalmışsa yeni pozisyon temel nokta olarak atanır ve Adım 2'ye gidilir. Amaç fonksiyonu değeri azalmamışsa Adım 4'e gidilir.
Adım 4	Her iki yönde arama yapıp yapılmadığı kontrol edilir yapılmışsa algoritma sonlandırılır. Yapılmamışsa arama yönü değiştirilerek Adım 2'ye gidilir.

3.4.2. Hooke Jeeves algoritması

HJ Algoritması Hooke ve Jeeves (1961) tarafından önerilen literatürde de çokça kullanılan YA algoritmasıdır. TT algoritmasına göre çok daha gelişmiştir. TT algoritmasında sadece tek bir adım boyutunda arama yapılırken HJ algoritmasında adım boyutu dizisi belirlenir. Algoritma başarılı olduğunda temel nokta güncellenerek aynı adım boyutu kullanılarak yeniden temel nokta etrafında arama yapılır. Algoritma başarısız olduğunda ise adım boyutu dizideki bir sonraki değer olarak değiştirilir. Dizideki tüm değerler kullanıldığında ve minimum değer değişmediğinde algoritma sonlandırılır. Algoritma akış şeması Şekil 3.29'da gösterilmektedir. Algoritma adımları Çizelge 3.11'de açıklanmaktadır (Benasla ve ark., 2008; Moser, 2009; Kirgat ve Surde, 2014; Erdemir ve Kalyoncu, 2015) .

Çizelge 3.11. HJ algoritma adımları

Adım Numarası	Açıklama
Adım 1	Başlangıç temel noktası, adım büyüklüğü dizisi amaç fonksiyonu ve arama yönü belirlenir. Başlangıç amaç fonksiyonu değerleri hesaplanır.
Adım 2	Adım büyüklüğü dizisindeki en büyük değer başlangıç pozisyonuna belirlenen arama yönünde eklenir. Yeni amaç fonksiyonu değeri hesaplanır.
Adım 3	Yeni amaç fonksiyonu değeri ile eski amaç fonksiyonu değeri karşılaştırılır. Minimum değer değişmişse temel nokta ve minimum değer güncellenir ve Adım 2'ye gidilir. Minimum değer değişmemişse Adım 4'e gidilir.
Adım 4	Her iki yönde arama yapılmamışsa arama yönü değiştirilir, yapılmışsa adım büyüklüğü, adım büyüklüğü dizisindeki bir sonraki değerle değiştirilerek küçültülür ve Adım 5'e gidilir.
Adım 5	Adım büyüklüğü dizisinin en küçük değerine ulaşılmışsa algoritma sonlandırılır. Ulaşılmamışsa Adım 2'ye gidilir.



Şekil 3.29. HJ algoritma akış şeması

3.5. Bu Çalışmada Önerilen Hibrit Algoritma

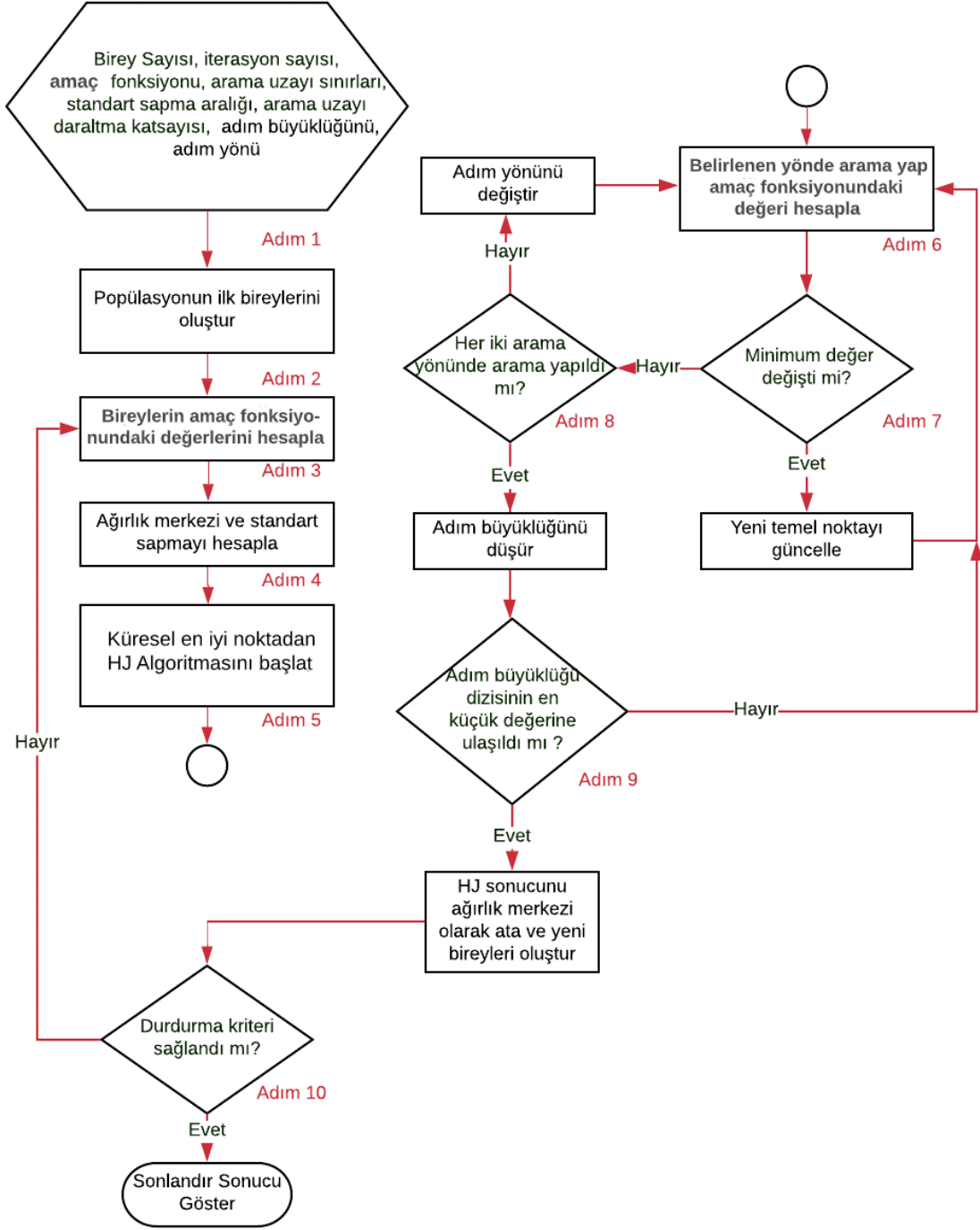
Bu çalışma kapsamında PID katsayıları optimizasyonu için küresel uzayı aramada kullanılan BPBÇ algoritması ile YA algoritması olan HJ algoritmasının hibrit edilmesi önerilmiştir. BPBÇ algoritmasının keşif özelliğini artırmak için standart sapma sınırları yüksek seçilir. Denklem 3.5 kullanılarak yeni elde edilen parçacıkların konumu arama uzayının dışına çıkarsa kütle merkezi etrafına rastgele yeniden gönderilir. Denklem 3.6'da iterasyon sayısının zamanla sömürü yeteneğini arttırmak adına arama genişliğini azalttığı gözlenmektedir. Önerilen algoritmada sömürü işini YA algoritması olan HJ yapacağından dolayı standart sapma eşitliği Denklem 3.19'da gösterilen şekilde güncellenmiştir.

$$\sigma = \frac{0.5(X_{min} + r(X_{maks} - X_{min}))}{1 + \frac{1}{t}} \quad (3.19)$$

Her bir iterasyon da bulunan kütle merkezinden Denklem 3.4 kullanılarak elde edilen kütle merkezinden HJ algoritması çalıştırılarak küresel en iyi değerin bulunması amaçlanmaktadır. HJ ile elde edilen en iyi değer hafızada tutularak KA için kullanılmaktadır. Bu çalışmada önerilen algoritma akış diyagramı Şekil 3.30'da, algoritma açıklamaları Çizelge 3.12'de gösterilmektedir.

Çizelge 3.12. BPBÇHJ hibrit algoritma adım açıklamaları

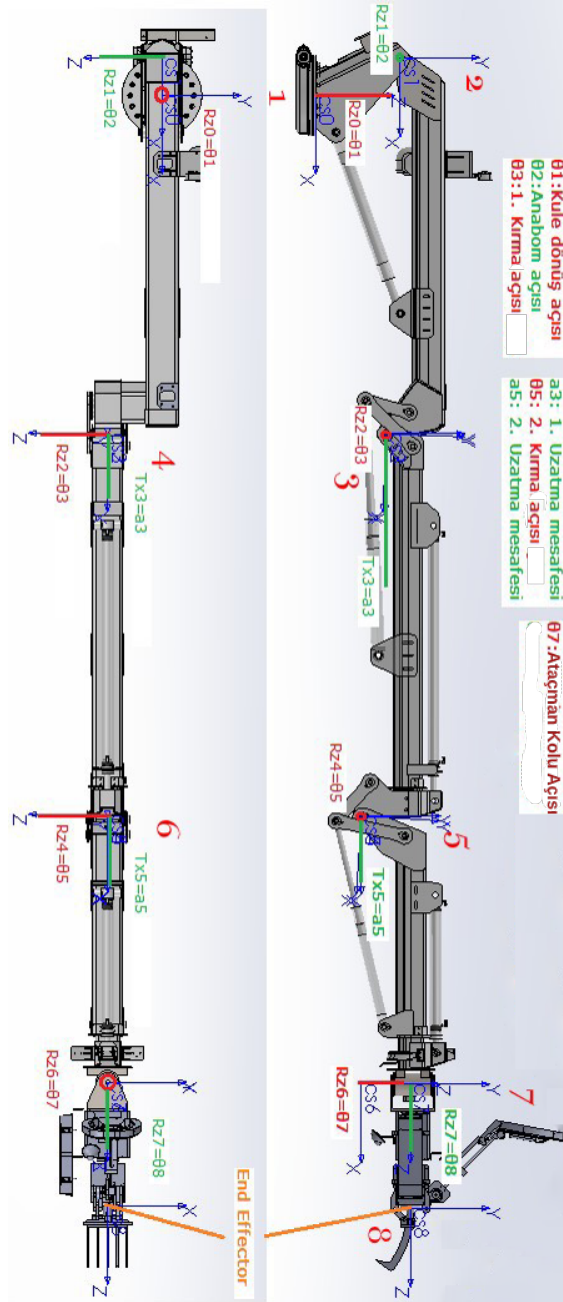
Adım Numarası	Açıklama
Adım 1	Popülasyonda kaç adet birey olacağı, iterasyon sayısı, amaç fonksiyonu, arama uzayı, arama uzayı daraltma katsayısı, Normal dağılım için standart sapma aralığı, HJ algoritması için adım büyüklüğü kümesi ve adım yönü belirlenir.
Adım 2	n adet eleman sayısına sahip K_p , K_d , K_i değerleri rastgele oluşturulur. Her bir [K_p , K_d , K_i] dizisi bir birey olarak adlandırılır.
Adım 3	Her bir bireyin amaç fonksiyonundaki değeri hesaplanır.
Adım 4	Bireylerin konumuna ve amaç fonksiyonundaki değerlerine göre ağırlık merkezi belirlenir.
Adım 5	Küresel en iyi noktadan öncelikle K_p sonra K_d , en son K_i kullanılarak YA başlatılır.
Adım 6	Her bir boyutta belirlenen yönde arama yapılır.
Adım 7	Minimum değer değişmişse temel nokta güncellenir değişmemişse Adım 8'e gidilir.
Adım 8	Her iki yönde de arama yapılmamışsa adım yönü değiştirilir. Yapılmışsa adım büyüklüğünü düşürerek Adım 9'a git
Adım 9	Adım büyüklüğü dizisinin en küçük değerine ulaşılmamışsa Adım 6'ya ulaşılmışsa elde edilen K_p , K_d , K_i değerlerini ağırlık merkezi olarak ata ve yeni bireylerin konumlarını elde et ve Adım 10'a git.
Adım 10	Durdurma kriteri sağlanmamışsa Adım 3'e git, sağlanmışsa algoritmayı sonlandır.



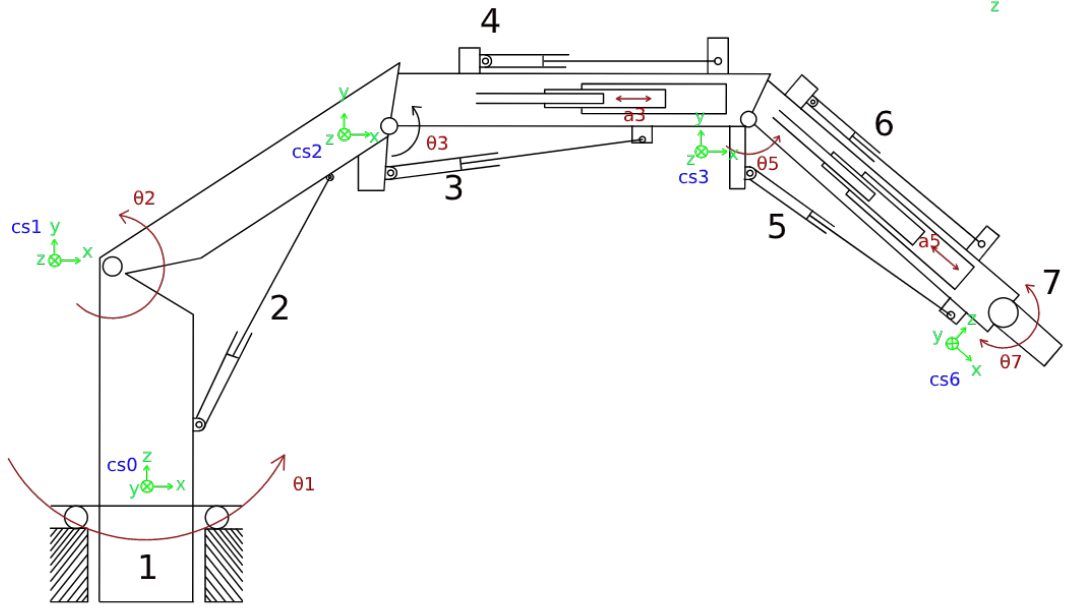
Şekil 3.30. BPBÇHJ hibrit algoritma akış diyagramı

4. HİDROLİK ROBOT MANİPÜLATÖR YAPISI

Bu çalışma MPG Makina Prodüksiyon Grubu Makine İmalat San. ve Tic. A.Ş.'de yapılmıştır. Hidrolik robot manipülatör, bir cismin bir yerden bir yere hassas bir şekilde taşınmasını amaçlamaktadır. Kullanılan sistemin mekanik yapısı, eksen yerleşimleri ve uzuv numaraları Şekil 4.1'de ve fiziksel modeli Şekil 4.2'de gösterilmektedir.



Şekil 4.1. Hidrolik robot manipülatör



Şekil 4.2. Hidrolik robot manipülâtörün fiziksel modeli

Uzuv numarası 1 ile gösterilen Kule, Cs0 koordinat sisteminin z ekseninde 360° açısal dönüş hareketi yapabilmektedir. Hidromotor ile tahrik edilir. Uzuv numarası 2 ile gösterilen Ana Bom, Cs1 koordinat sisteminin z ekseninde 0° ile 50° arasında açısal hareket yapabilmektedir. Tahrik elemanı hidrolik silindirdir. Uzuv numarası 3 ile gösterilen 1.Kırma, Cs2 koordinat sisteminin z ekseninde -180° ile 2° arasında açısal hareket yapabilmektedir. Tahrik elemanı hidrolik silindirdir. Uzuv numarası 4 ile gösterilen 1.Kırma'nın içerisinde çıkan uzatma hareketini sağlayan bom yapılarıdır. Cs2 koordinat sisteminin x ekseninde 0 cm ile 130 cm arasında doğrusal hareket yapabilmektedir. Tahrik elemanı hidrolik silindirdir. Uzuv numarası 5 ile gösterilen 2.Kırma, Cs3 koordinat sisteminin z ekseninde -180° ile 0° arasında açısal hareketi yapabilmektedir. Uzuv numarası 6 ile gösterilen 2.Kırma'nın içerisinde çıkan uzatma hareketini sağlayan bom yapılarıdır. Cs3 koordinat sisteminin x ekseninde 0 ile 70 cm, arasında doğrusal hareket yapabilmektedir. Tahrik elemanı hidrolik silindirdir. Uzuv numarası 7 ile gösterilen Ataçman Kolu, Cs6 koordinat sisteminin z ekseninde -90° ile 90° açısal hareket yapabilmektedir.

Çizelge 4.1'de, Şekil 4.1'de numaralarla gösterilen uzuvlara verilen isimleri ve hareket sınırları gösterilmektedir.

Çizelge 4.1. Uzuvarın isim ve hareket tanımlamaları

Uzuv Numarası	Uzuv İsmi	Hareket Çeşidi	Hareket Sınırı
1	Kule	Açısal	Cs0 z ekseninde -180° ile 180° arasında
2	Ana bom	Açısal	Cs1 z ekseninde 0° ile 50° arasında
3	1.Kırma	Açısal	Cs2 z ekseninde -180° ile 2° arasında
4	1.Kırma	Doğrusal	Cs2 x ekseninde 0 ile 130 cm arasında
5	2.Kırma	Açısal	Cs3 z ekseninde -180° ile 0° arasında
6	2.Kırma	Doğrusal	Cs3 x ekseninde 0 ile 70 cm arasında
7	Ataçman	Açısal	Cs6 z ekseninde -90° ile +90° arasında
8	Ataçman	Açısal	Cs8 z ekseninde -180° ile 180° arasında

Hidrolik robot manipülatörün uzuvarları yaptığı hareketlere göre isimlendirilmiştir: 1 no.lu uzuv Kule Sağa-Sola Dönüş, 2 no.lu uzuv Ana Bom Kaldırma-İndirme, 3 no.lu uzuv 1. Kırma Kaldırma-İndirme, 4 no.lu uzuv 1. Kırma Uzatma-Çekme, 5 no.lu uzuv 2. Kırma Kaldırma-İndirme, 6 no.lu uzuv 2. Kırma Uzatma-Çekme, 7 no.lu uzuv Ataçman Kolu Sağa-Sola Dönüş ve 8 no.lu uzuv Ataçman Dönüş olarak isimlendirilmiştir.

Hidrolik robot manipülatör hareketlerinde sınırlanan debi değerleri Çizelge 4.2’de gösterilmektedir.

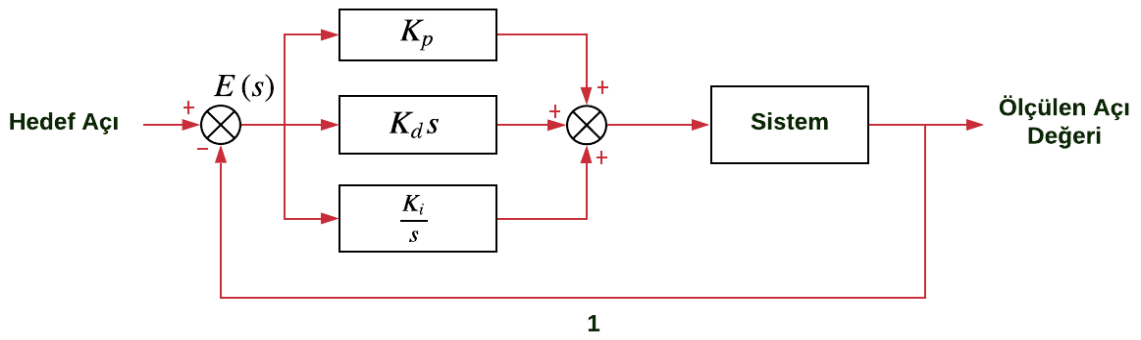
Çizelge 4.2. Hidrolik robot manipülatör hareket isimleri ve maksimum debi değerleri

Uzuv Numarası	Hareket İsmi	Maksimum Debi Litre(l/da)
1	KSODH	5,5
1	KSADH	5
2	ABKH	5
2	ABKH	2,375
3	BKKH	3,627
3	BKİH	2,75
4	BKBUH	12
4	BKBÇH	7
5	İKKH	3
5	İKİH	1,5
6	İKBUH	17,5
6	İKBÇH	8
7	ASADH	1,25
7	ASODH	1,25

5. ARAŞTIRMA BULGULARI VE TARTIŞMA

5.1. Problemin Tanımı

Otonom çalışan ya da hedef konumu belli olan robotlar kapalı çevrim kontrol sistemiyle kontrol edilmek zorundadır. Kontrol işlemi hedef konum ile mevcut konum arasındaki farkın istenen şekilde 0'a indirilmesiyle gerçekleştirilir. Klasik PID metotlarından Ziegler Nichols metodu deneysel olarak kullanılmakta olup hem zaman alıcı olması hem de operatörün kabiliyetine bağlı olmasından dolayı son yıllarda yerini yapay zekâ optimizasyon algoritmalarına bırakmaktadır. PID katsayılarının optimizasyonu için kontrol edilecek sistemin transfer fonksiyonu gerekmektedir. Sistemin kapalı çevrim blok diyagramı Şekil 5.1' de gösterilmektedir.



Şekil 5.1. Hareket sistemlerinin kapalı çevrim kontrol şeması

PID kontrolcü transfer fonksiyonu G_D Denklem 5.1'de, sistemin açık çevrim transfer fonksiyonu $G_{Aç}$ Denklem 5.2'de, sistemin kapalı çevrim transfer fonksiyonu ise Denklem 5.3'de gösterilmiştir.

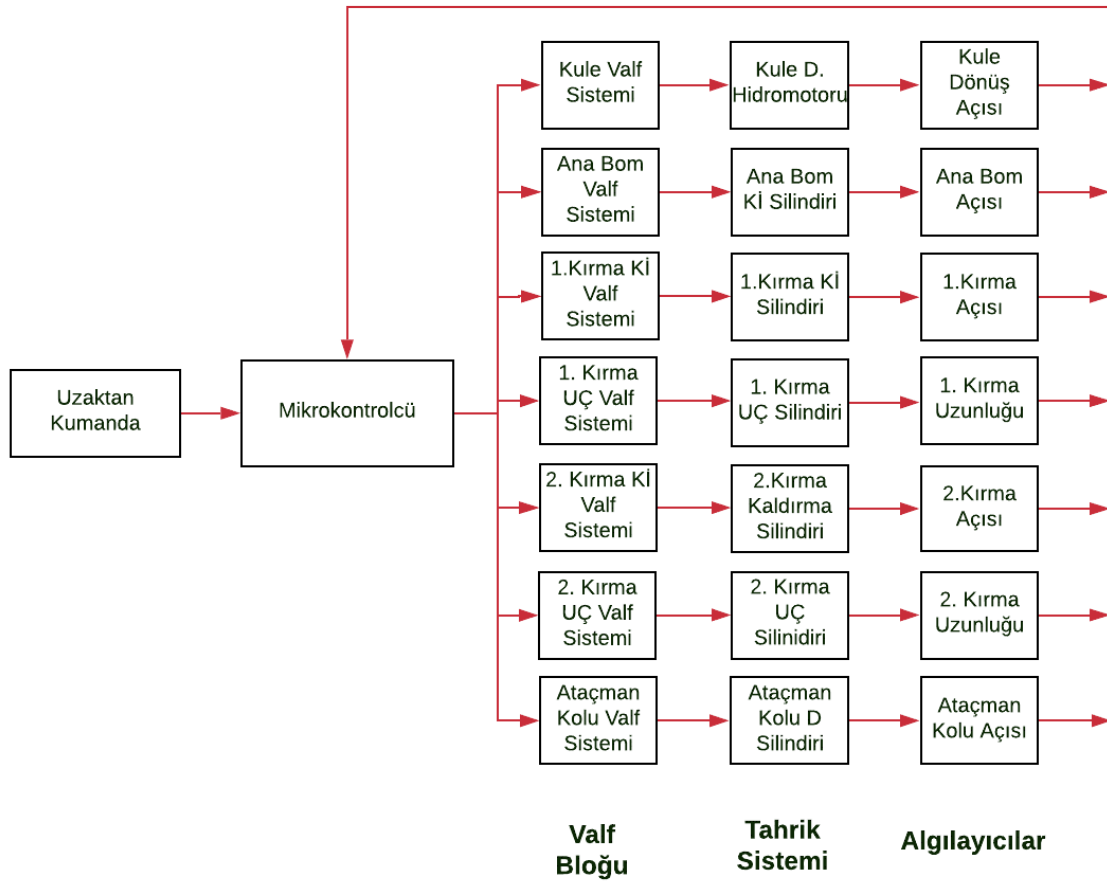
$$G_D = K_p + K_d s + \frac{K_i}{s} \quad (5.1)$$

$$G_{Aç} = G_D G_S \quad (5.2)$$

$$G_{Kç} = \frac{G_D G_S}{1 + G_D G_S} \quad (5.3)$$

5.2. Deneysel Modelleme ile Elde Edilen Sonuçlar

Hidrolik robot manipülatör uzaktan kumanda aracılığıyla manuel olarak kontrol edilebilmektedir. Kumandadan alınan sinyal mikro kontrolcüde valf sistemi için anlamlı hale getirilir ve valf sistemine gönderilir. Valf sistemi verilere uygun debiyi tahrik elemanlarına gönderir ve mekanik hareket elde edilir. Uzun konumları mikro kontrolcüye gönderilir. Şekil 5.2’de robot manipülatörün veri toplama şeması gösterilmektedir.

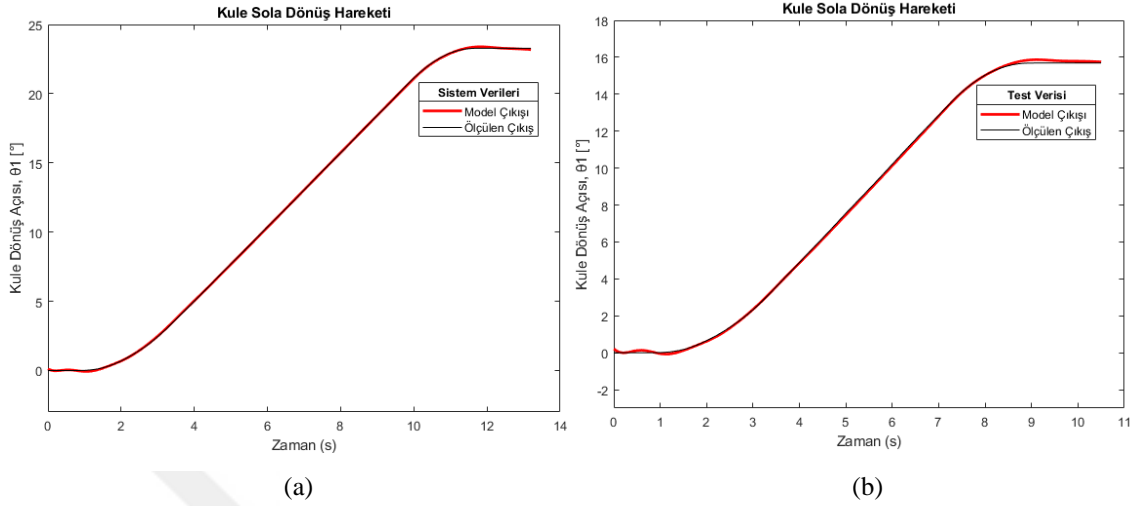


Şekil 5.2. Robot manipülatörün kontrolü ve veri toplama şeması

Robot kol sisteminin her bir uzvunun kaldırma-indirme ya da uzatma-çekme hareketlerinde talep edilen debi giriş verisi ve algılayıcıdan elde edilen açı ya da uzunluk değeri çıkış verisi alınarak transfer fonksiyonları, MSI’da elde edilmiş test verileri ile kontrol edilmiştir.

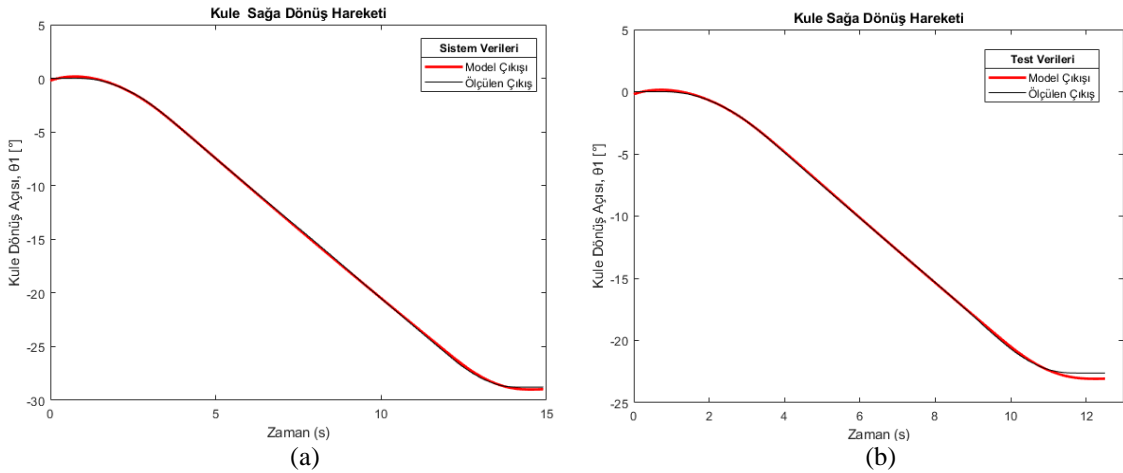
KSODH için 4 kutuplu ve 3 sıfırlı model daha iyi sonuçlar vermiştir. KSODH sistemi için elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.3. (a)’da gösterilmektedir. KSODH modeli ile ölçülen değerler arasındaki uyumluluk oranı

%99.56'dır. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.3. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %98.72'dir.



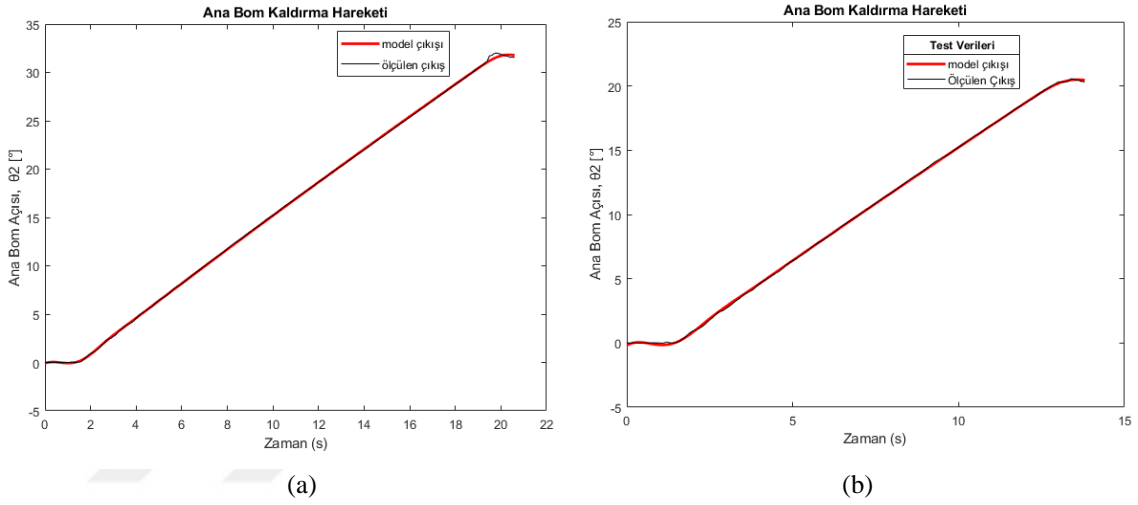
Şekil 5.3. a. KSODH sistem verileri, b. KSODH test verileri

KSADH için 2 kutuplu ve 1 sıfırlı sistemi modeli en iyi sonucu vermiştir. KSADH sistemi için elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.4. (a)'da gösterilmektedir. KSADH modeli ile ölçülen değerler arasındaki uyumluluk oranı %99'dur. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.4. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %98.21 olmuştur.



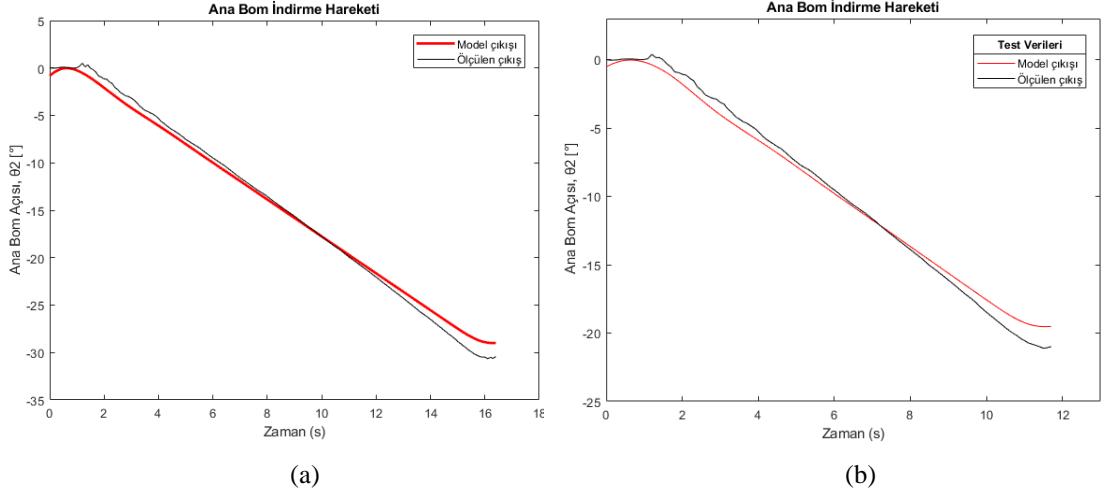
Şekil 5.4. a. KSADH sistem verileri, b. KSADH test verileri

ABKH için 3 kutuplu ve sıfırı olmayan model daha iyi sonuçlar vermiştir. ABKH sistemi için elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.5. (a)'da gösterilmektedir. ABKH modeli ile ölçülen değerler arasındaki uyumluluk oranı %99.06'dır. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri Şekil 5.5. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %98.96 olmuştur.



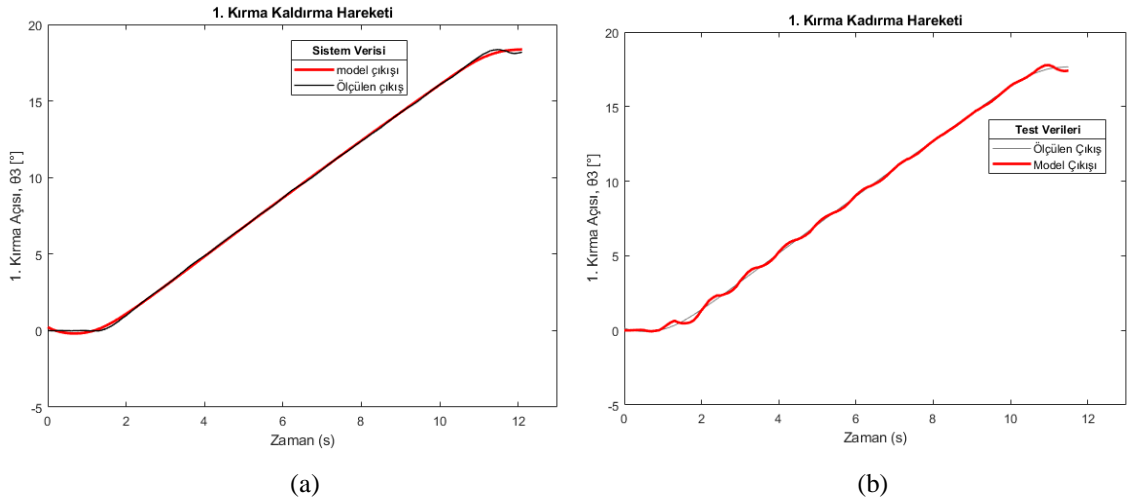
Şekil 5.5. a. ABKH sistem verileri, b. ABKH test verileri

ABİH için 3 kutuplu ve sıfırı olmayan model daha iyi sonuçlar vermiştir. ABİH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.6. (a)'da gösterilmektedir. ABİH modeli ile ölçülen değerler arasındaki uyumluluk oranı %92.25'dur. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.6. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %90.25 olmuştur.



Şekil 5.6. a. ABİH sistem verileri, b. ABİH test verileri

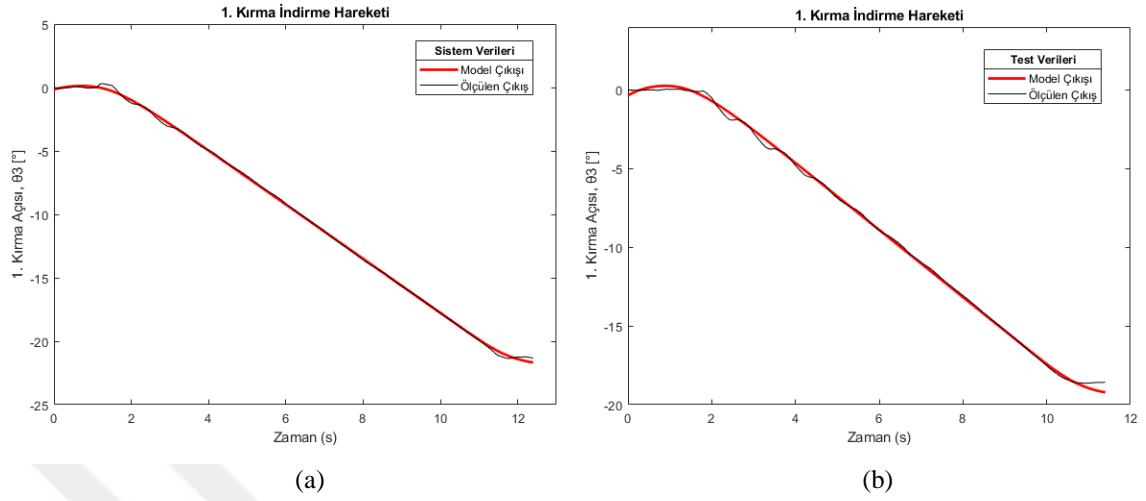
BKKH için 2 kutuplu ve 1 sıfırlı model daha iyi sonuçlar vermiştir. BKKH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.7. (a)'da gösterilmektedir. BKKH modeli ile ölçülen değerler arasındaki uyumluluk oranı %98.49'dur. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.7. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %97.69 olmuştur.



Şekil 5.7. a. BKKH sistem verileri, b. BKİH test verileri

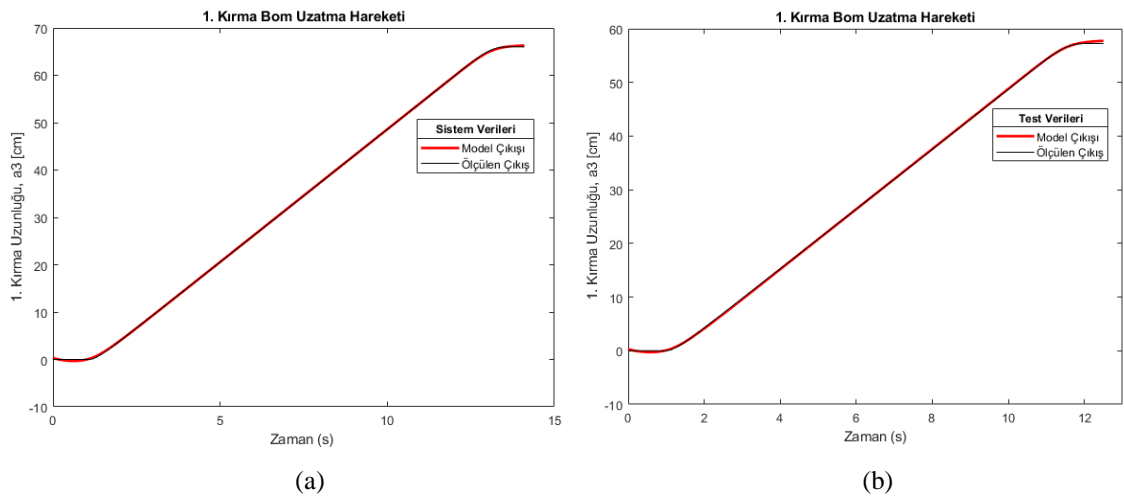
BKİH için 2 kutuplu ve sıfırı olmayan model daha iyi sonuçlar vermiştir. BKİH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.8. (a)'da gösterilmektedir. BKİH modeli ile ölçülen değerler arasındaki uyumluluk oranı %98.08'dir. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.8. (b)'de gösterilmektedir. Test verileri

değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %97.04 olmuştur.



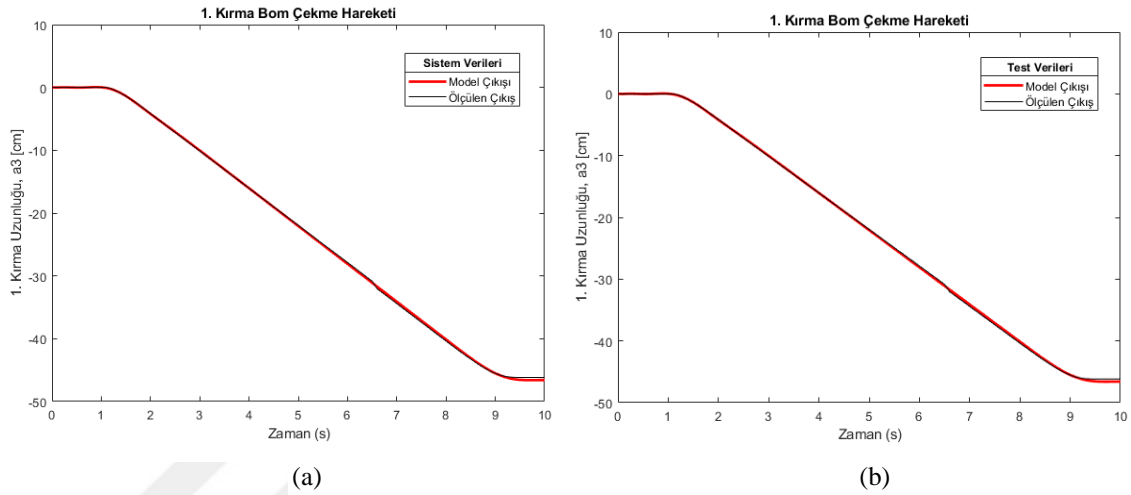
Şekil 5.8. a. BKİH sistem verileri, b. BKİH test verileri

BKBUH için 2 kutuplu ve 1 sıfırlı model daha iyi sonuçlar vermiştir. BKBUH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.9. (a)'da gösterilmektedir. BKBUH modeli ile ölçülen değerler arasındaki uyumluluk oranı %99.53'tür. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.9. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %99.31 olmuştur.



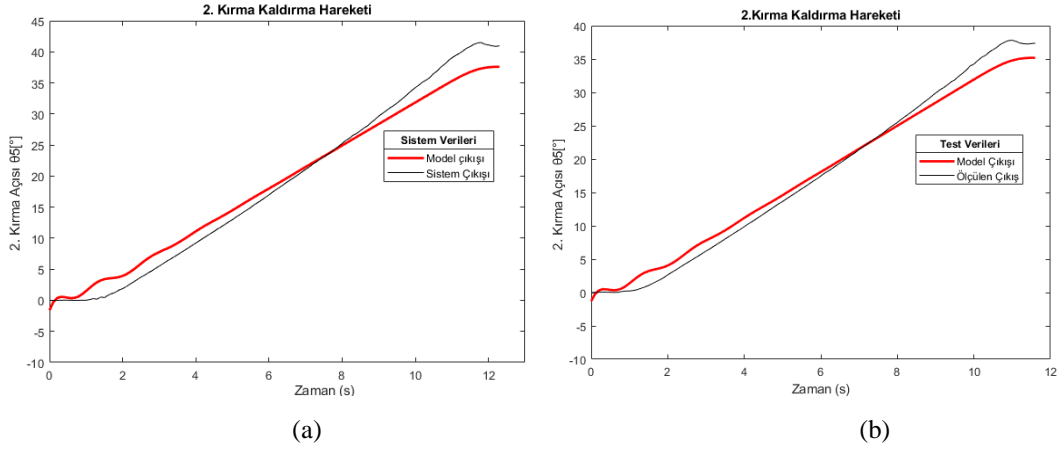
Şekil 5.9. a. BKBUH sistem verileri, b. BKBUH test verileri

BKBÇH için 5 kutuplu ve 2 sıfırlı model daha iyi sonuçlar vermiştir. BKBÇH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.10.(a)'da gösterilmektedir. BKBÇH modeli ile ölçülen değerler arasındaki uyumluluk oranı %99.6'dır. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.10. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %98.87 olmuştur.



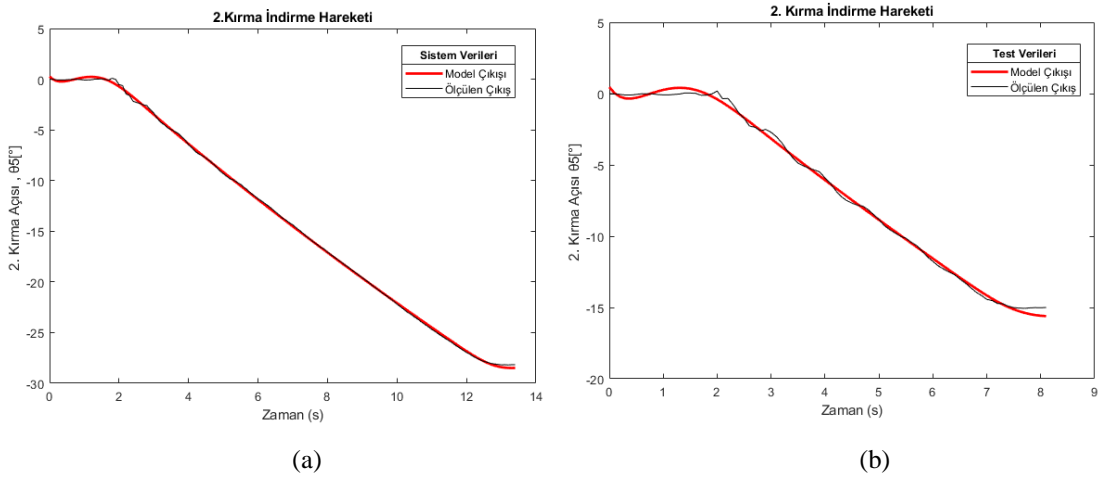
Şekil 5.10. a. BKBÇH sistem verileri, b. BKBÇH test verileri

İKKH için 4 kutuplu ve sıfır olmayan model daha iyi sonuçlar vermiştir. İKKH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.11. (a)'da gösterilmektedir. İKKH modeli ile ölçülen değerler arasındaki uyumluluk oranı %85.03'tür. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.11. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %88.13 olmuştur.



Şekil 5.11. a. İKKH sistem verileri, b. İKKH test verileri

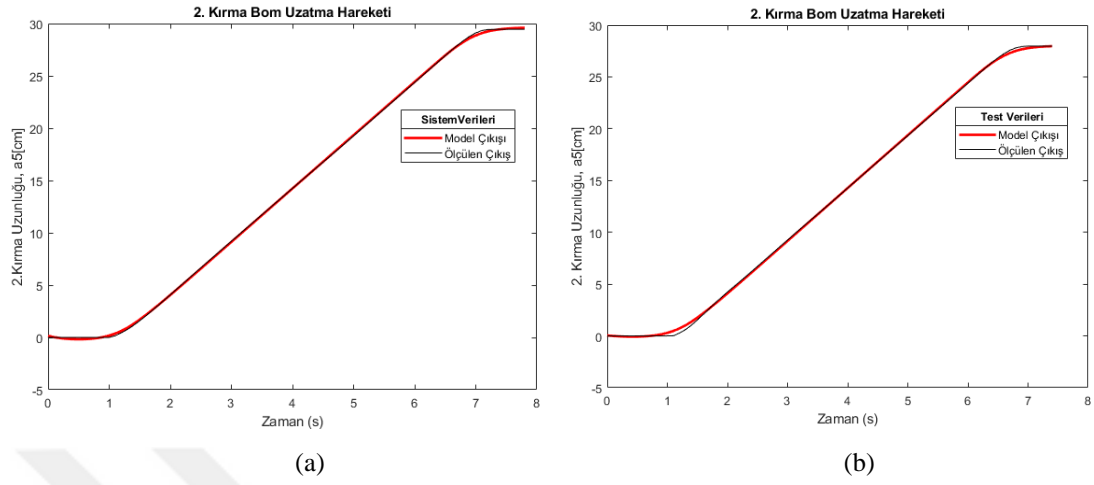
İKİH için 3 kutuplu ve sıfırı olmayan model daha iyi sonuçlar vermiştir. İKİH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.12. (a)'da gösterilmektedir. İKİH modeli ile ölçülen değerler arasındaki uyumluluk oranı %98.42'dur. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.12. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %94.68 olmuştur.



Şekil 5.12. a. İKİH sistem verileri, b. İKİH test verileri

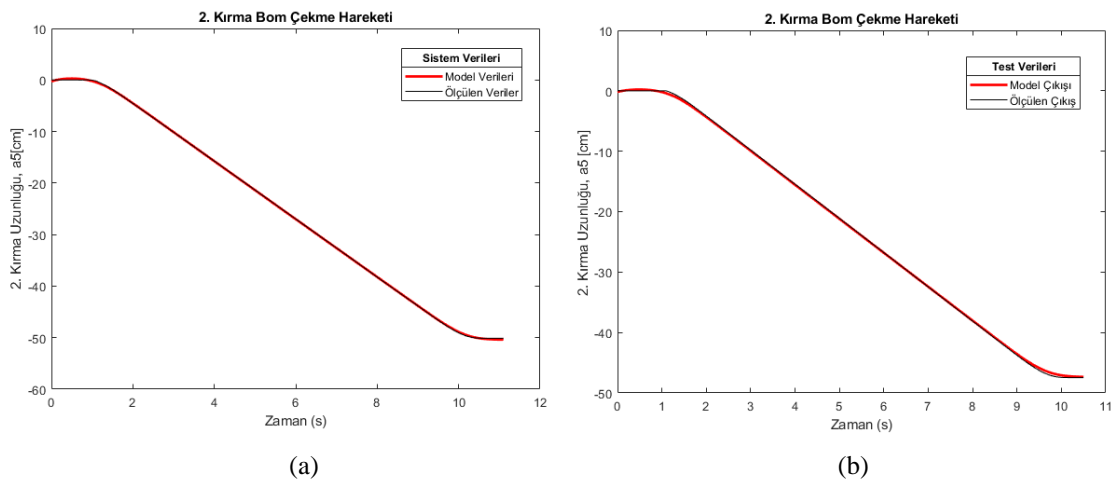
İKBUH için 2 kutuplu ve sıfırı olmayan model daha iyi sonuçlar vermiştir. İKBUH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.13. (a)'da gösterilmektedir. İKBUH modeli ile ölçülen değerler arasındaki uyumluluk oranı %99.06'dır. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.13. (b)'de gösterilmektedir. Test verileri

değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %98.6 olmuştur.



Şekil 5.13. a. İKBUH sistem verileri, b. İKBUH test verileri

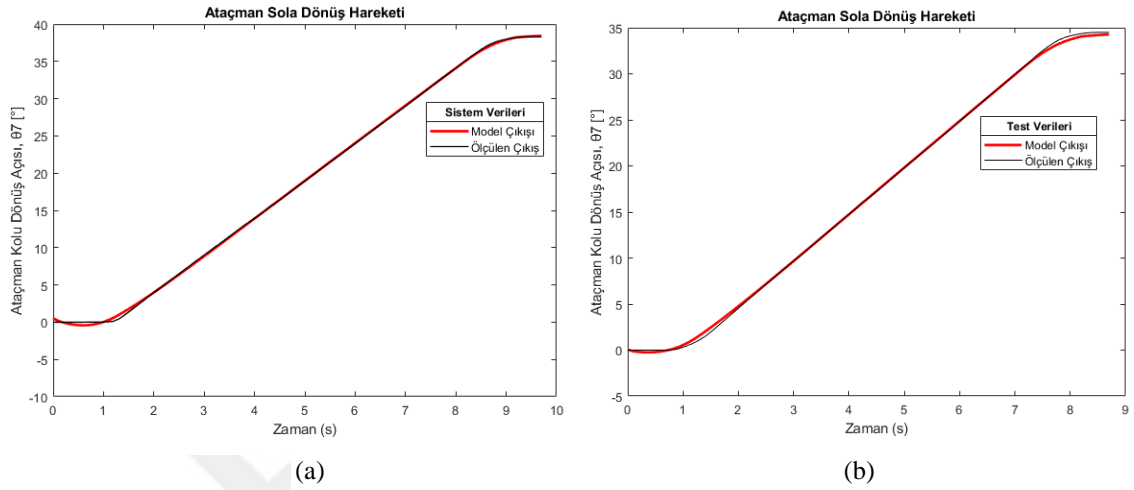
İKBÇH için 2 kutuplu ve sıfırı olmayan model daha iyi sonuçlar vermiştir. İKBÇH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.14. (a)'da gösterilmektedir. İKBÇH modeli ile ölçülen değerler arasındaki uyumluluk oranı %99.43'tür. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.14. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %98.89 olmuştur.



Şekil 5.14. a. İKBÇH sistem verileri, b. İKBÇH test verileri

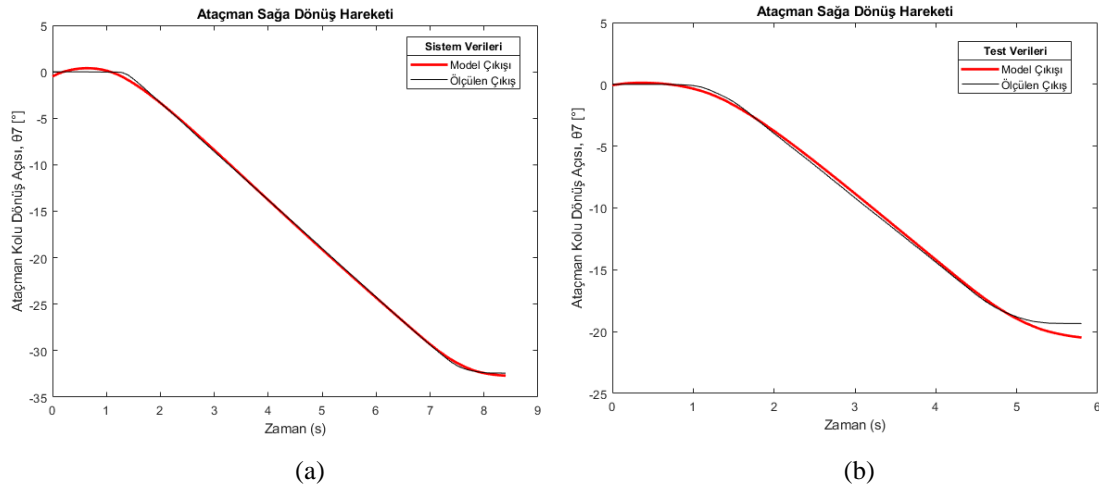
ASODH için 2 kutuplu ve 1 sıfırlı model daha iyi sonuçlar vermiştir. ASODH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.15. (a)'da

gösterilmektedir. ASODH modeli ile ölçülen değerler arasındaki uyumluluk oranı %98.38'dur. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.15. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %95.05 olmuştur.



Şekil 5.15. a. ASODH sistem verileri, b. ASODH test verileri

ASADH için 3 kutuplu ve sıfırı olmayan model daha iyi sonuçlar vermiştir. ASADH sisteminde elde edilen model çıkışı ve ölçülen çıkış değerleri Şekil 5.16. (a)'da gösterilmektedir. ASADH modeli ile ölçülen değerler arasındaki uyumluluk oranı %98.3'tür. Elde edilen transfer fonksiyonu ile sistem yeni verilerle test edildiğinde model çıkışı ve ölçülen çıkış değerleri 5.16. (b)'de gösterilmektedir. Test verileri değerlendirildiğinde model değerleri ile ölçülen değerler arasındaki uyumluluk oranı %93.59 olmuştur.



Şekil 5.16. a. ASADH sistem verileri, b. ASADH test verileri

Elde edilen transfer fonksiyonları ve uzuv isimleri ve uyumluluk oranları Çizelge 5.1’de gösterilmektedir.

Çizelge 5.1 Uzuvarın hareket şekilleri ve transfer fonksiyonları

Hareket Türü	Sistem TF	Sistem Verileri ile TF Uyumluluk Oranı	Test Verileri İle TF Uyumluluk Oranı
KSADH	$\frac{1.081}{s^2 + 1.992s + 0.01775}$	99.00	98.21
KSODH	$\frac{15.69s + 1.626}{s^4 + 3.313s^3 + 33.85s^2 + 13.81s + 0.1618}$	99.56	98.72
ABKH	$\frac{3.283}{s^3 + 2.644s^2 + 9.031s + 0.05001}$	99.06	98.96
ABİH	$\frac{4.047}{s^3 + 1.379s^2 + 4.941s + 2.6615e - 09}$	92.25	90.25
BKKH	$\frac{0.1953s + 0.9321}{s^2 + 1.708s + 0.01969}$	98.49	97.69
BKİH	$\frac{1.192}{s^2 + 1.535s + 2.224e - 6}$	98.08	97.04
BKBUH	$\frac{0.2248s + 1.062}{s^2 + 2.275s + 9.05e - 8}$	99.53	99.31
BKBÇH	$\frac{38.57s^2 + 166.4s + 269.5}{s^5 + 12.71s^4 + 80s^3 + 258.2s^2 + 313.1s + 0.06334}$	99.6	98.87
İKKH	$\frac{60.31}{s^4 + 4.054s^3 + 24.56s^2 + 52.24s + 1.660e - 10}$	85.03	88.13
İKİH	$\frac{12.02}{s^3 + 3.749s^2 + 6.179s + 0.1446}$	98.42	94.68
İKBUH	$\frac{1.246}{s^2 + 3.402s + 0.005659}$	99.09	98.6
İKBÇH	$\frac{2.405}{s^2 + 3.4s + 0.005326}$	99.43	98.89
ASADH	$\frac{8.416}{s^2 + 1.138s + 0.059}$	98.3	93.59
ASODH	$\frac{3.385s + 6.275}{s^2 + 1.37s + 0.006665}$	98.38	95.05

5.3. Kullanılan Amaç Fonksiyonu

PID katsayıları optimizasyonu konusunda literatürde ISE, ITAE, ITSE, IAE çokça kullanılmaktadır. Literatürde çokça kullanılan performans kriterleri kaynak araştırması bölümünde detaylı anlatılmaktadır. Gaing (2004) yaptığı çalışmada daha iyi bir sonuç elde etmek için Denklem 2.6'da gösterilen amaç fonksiyonunu önermiştir. PID katsayıları optimizasyonunda amaç fonksiyonu belirlenirken robotun esnekliği titreşimi, dayanıklılığı göze alınmalıdır. Bunun yanında sistem çıkış parametreleri de oldukça önemlidir. Önerilen özgün amaç fonksiyonu Denklem 5.4'de gösterilmektedir.

$$J = (0.1 * \int_0^t |e(t)| dt) + M_p * 100 + T_r \quad (5.4)$$

Kullanılan amaç fonksiyonunda $e(t)$ çıkış ile giriş (referans) arasındaki fark, M_p maksimum aşma miktarı, T_r ise yükselme zamanıdır. Minimum yükselme zamanı, minimum maksimum aşma ve minimum hatalar toplamı istenmektedir. Sistemde maksimum aşma hiç istenmediğinden yüksek bir katsayı ile çarpılmıştır. Yükselme zamanı ise hedefe hızlı yaklaşmak açısından önemlidir.

5.4. Algoritmalarda Seçilen Katsayılar

KA algoritmalarında da seçilen katsayıların önemi çok büyük önem arz etmektedir. GA'da P_M , P_S ve P_C , PSO'da c_1 , c_2 ve w , AA'da m , e , nep , nsp , ngh ve gs , BPBÇ algoritmasında X_{maks} , X_{min} ve t , GKA' da s ve P_a katsayıları ilgili KA algoritmasının sömürü-keşif özelliklerini belirtmektedir. Bazı katsayıların artırılması algoritmadaki keşif yeteneğini artırırken sömürü yeteneğini azaltmaktadır. Aksi durumu da söz konusudur. GA'da P_M değeri artırılırsa keşif yeteneği artar buna karşılık sömürü yeteneği azalır. PSO'da c_2 katsayısı küresel en iyi noktaya yaklaşma hızını belirlerken c_1 katsayısı yerel en iyi değere yaklaşma hızını belirler. Bu katsayıların büyük seçilmesi yakınsama hızını artırırken, keşif özelliğini azaltabilir. AA'da m , e , nep , nsp , ngh katsayılarının nispeten artması sömürü yeteneğini artırırken gs değerinin artması keşif yeteneğinin artmasını sağlar. BPBÇ algoritmasında standart sapma sınırlarının genişlemesi arama alanını artırarak keşif yeteneğini artırır. İterasyon sayısı arttıkça standart sapma sınırlarının daralması algoritmanın zamanla keşif yeteneğinin azalır

sömürü yeteneğinin artmasını sağlamaktadır. Shehata ve ark. (2014) yaptıkları çalışmada PSO'nun hız denklemindeki c_1 ve c_2 katsayılarının iterasyon sayısına göre azaltılarak algoritma başlarında keşif yeteneğini yüksek tutulması istenmiş, iterasyon sayısı arttıkça sömürü özelliğini öne çıkarılmıştır. GKA'da P_a değerinin artması keşif yeteneğini geliştirir. GA, PSO ve AA'da kullanılan parametreler Çizelge 5.2'de ve BPBÇ ve GKA'da kullanılan katsayılar Çizelge 5.3' de gösterilmiştir.

Çizelge 5.2. GA, PSO ve AA için Seçilen Katsayılar

GA			PSO			AA					
P_S	P_M	P_C	c_1	c_2	w	m	e	nep	nsp	ngh	gs
0.3	0.2	0.5	0.6	1.7	0.5	10	4	12	7	0.5	10

Çizelge 5.3. GKA ve BPBÇ için Seçilen Katsayılar

GKA	BPBÇ						
P_a	X_{minKp}	X_{minKd}	X_{minKi}	X_{maksKp}	X_{maksKd}	X_{maksKi}	t
0.25	-0.5	-0.3	-0.05	0.5	0.3	0.05	2

Algoritmelerde parametreler belirlenirken keşif sömürü dengesi dikkate alınmıştır. GA'da keşif işlemi için kullanılan parametre P_M , sömürü için kullanılan parametreler ise P_S ve P_C 'dir. P_M parametresinin artırılması keşif yeteneğini artırırken sömürü yeteneğini azaltmaktadır. GA'nın yerel maksimum ya da yerel minimum noktalara kolayca takılmasından dolayı bu eksikliği gidermek için keşif için ayrılan birey sayısı 20 seçilmiştir. P_S bir sonraki nesle mevcut nesilden aktarılan en güçlü bireylerin sayısını belirleyen parametredir. 30 adet güçlü birey bir sonraki nesle aktarılıp bu 30 bireyden 50 birey çaprazlama ile 20 birey ise mutasyonla elde edilmiştir. GA parametreleri deneme yanılma yoluyla seçilmiştir. PSO Algoritmasında parametreler Trelea (2003)'nın yaptığı çalışmadan yola çıkılarak; $c_1 = 0.6$ ve $c_2 = 1.7$ olarak seçilmiştir. AA'da keşif yeteneğini artıran gs parametresi Karaboga ve Basturk (2008) tarafından yapılan çalışmada %5-%10 arasında olacağı belirtilmektedir. Keşif yeteneği bu aralıkta verilen en büyük değer olan %10 seçilmiştir. 48 arı elit 4 bölgeye, kalan 42 arı ise daha az elit olan 6 bölgeye gönderilecek şekilde parametreler seçilmiştir. AA nispeten diğer sezgisel algoritmalara göre daha az yerel minimum değere takılmaktadır. Bundan dolayı GA'da %20 olan keşif için ayrılan birey AA'da %10 seçilmiştir. GKA seçilen parametrelerin tamamı Iglesias ve

ark. (2018) yaptığı çalışmadan alınmıştır. BPBÇ Algoritmasında kullanılan tüm parametreler deneme yanılma yoluyla elde edilmiştir. Merkez nokta belirlendikten sonra merkez noktaya göre yeni parçacıkların saçıldığı standart sapma aralığı K_p için (-0.5,0.5) K_d için (-0.3,0.3) ve K_i için (-0.05,0.05) seçilmiştir. Her bir algoritma için en iyi sonuç elde edilmeye çalışılmış, önerilen algoritmayı yüceltmek için diğer algoritmaların etkinliği azaltılmaya çalışılmamıştır.

5.5. Sayısal Sonuçlar

Sezgisel algoritmalar rastsal hareketler içerdiğinden dolayı algoritma yeniden çalıştırıldığında farklı sonuçlar vermektedir. Literatürde bu sorunu çözmek için (Gaing, 2004) 20 kez algoritmayı çalıştırılıp ortalamasının alınmasını önermiştir. Algoritma sayısının çok olması ve optimize edilecek 14 adet transfer fonksiyonu ve 6 farklı algoritma olmasından dolayı her bir algoritma her bir hareket için 3 kere çalıştırılıp en iyi sonuçlar karşılaştırılmıştır. İlk olarak KA algoritmaları (AA-BPBÇ-GA-GKA-PSO) karşılaştırılmış daha sonra ilgili problem için en iyi değeri veren KA algoritması BPBÇHJ algoritması ile karşılaştırılmıştır. Tüm algoritmalarda iterasyon sayısı 20, popülasyonun birey sayısı 100 seçilmiştir. Durdurma kriteri olarak iterasyon sayısının tamamlanması göz önüne alınmıştır.

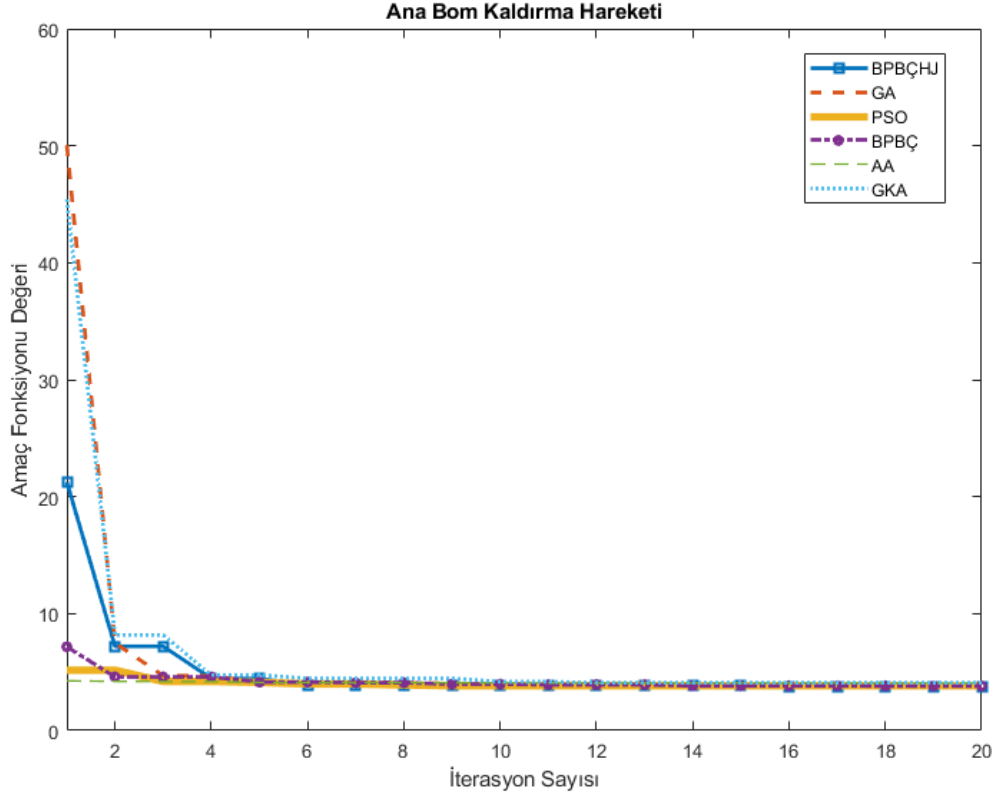
5.6. Ana Bom Hareketi Elde Edilen Sonuçlar

5.6.1. Ana bom kaldırma hareketi

Ana Bom diğer uzuvları taşıdığından dolayı kontrolü diğer uzuvlara göre daha çok önem arz etmektedir. ABKH için Şekil 5.17'deki en iyi sonuçlar değerlendirildiğinde BPBÇ Algoritması'nın diğer KA algoritmalarına nispeten daha düşük amaç fonksiyonu değerine sahip olduğu gözlenmektedir. BPBÇ Algoritması ile BPBÇHJ Algoritması karşılaştırıldığında BPBÇHJ Algoritması'nın 20. İterasyon sonunda daha düşük amaç fonksiyonu değerine sahip olduğu gözlenmektedir. 20 iterasyon sonunda ABKH için algoritmaların elde ettiği minimum maliyet değeri ve elde edilen PID katsayıları Çizelge 5.4'te gösterilmektedir.

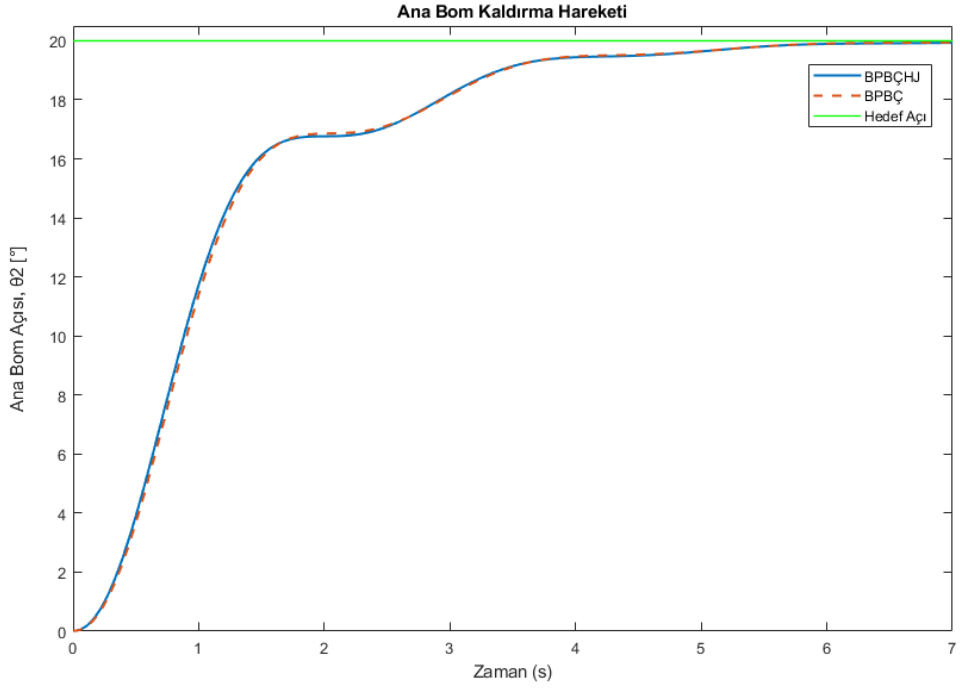
Çizelge 5.4 ABKH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç fonksiyonu Değeri	4.0684	3.7913	3.8045	4.0858	3.8235	3,8045
K_p	2.1790	2.0963	2.2164	2.2176	2.1749	2.2421
K_d	0.4051	0.2834	0.3770	0.5251	0.2622	0.4271
K_i	0.0037	0.0105	0.0131	0	0.0120	0.0124

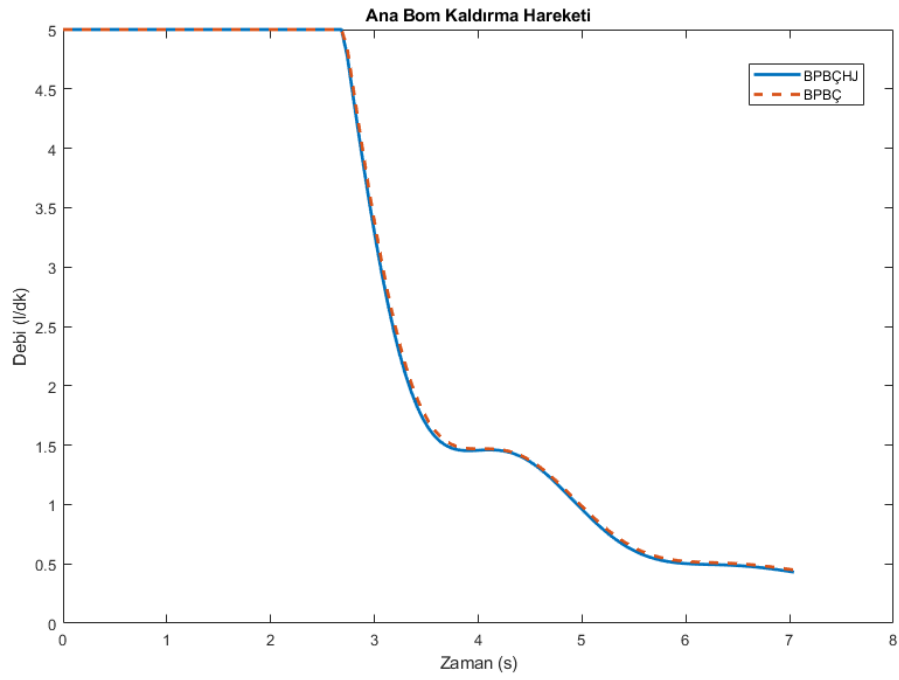


Şekil 5.17. ABKH, arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

BPBÇHJ ve BPBÇ ile elde edilen en uygun PID katsayıları uygulandığında ve Ana Bom'un 20° kaldırması istendiğinde Şekil 5.18'deki Ana Bom Açısı- Zaman grafiği elde edilmiştir. Her iki sonucun da uygulanabilir olduğu gözlenmektedir. ABKH, PID kontrolcü çıkışı Şekil 5.19'da gösterilmektedir.



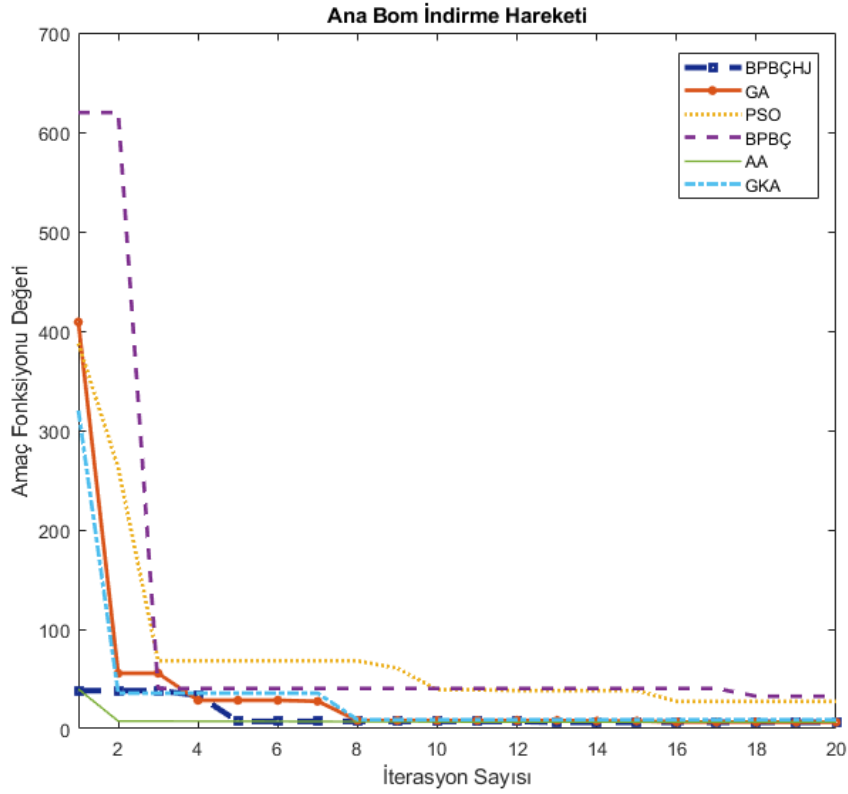
Şekil 5.18. ABKH Basamak Cevabı



Şekil 5.19. ABKH kontrolcü çıkışı

5.6.2. Ana bom indirme hareketi

ABİH için Şekil 5.20'deki en iyi sonuçlar değerlendirildiğinde GA'nın bu problem için diğer KA algoritmalarına nispeten daha düşük amaç fonksiyonu değerine sahip olduğu gözlenmektedir.



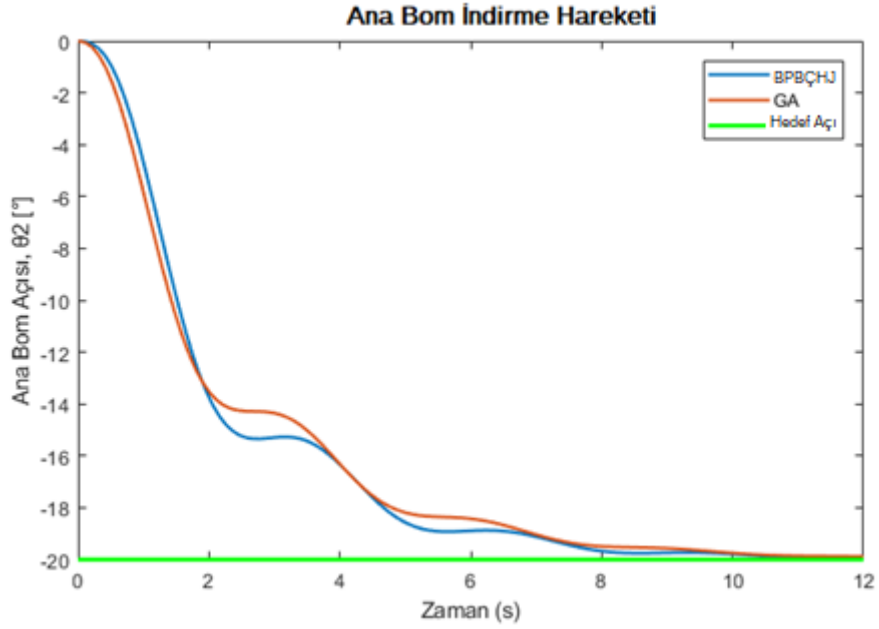
Şekil 5.20. ABİH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda ABİH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.5'te gösterilmektedir. Algoritma sonunda BPBÇHJ Algoritması'nın en düşük amaç fonksiyonu değerine sahip olduğu gözlenmektedir.

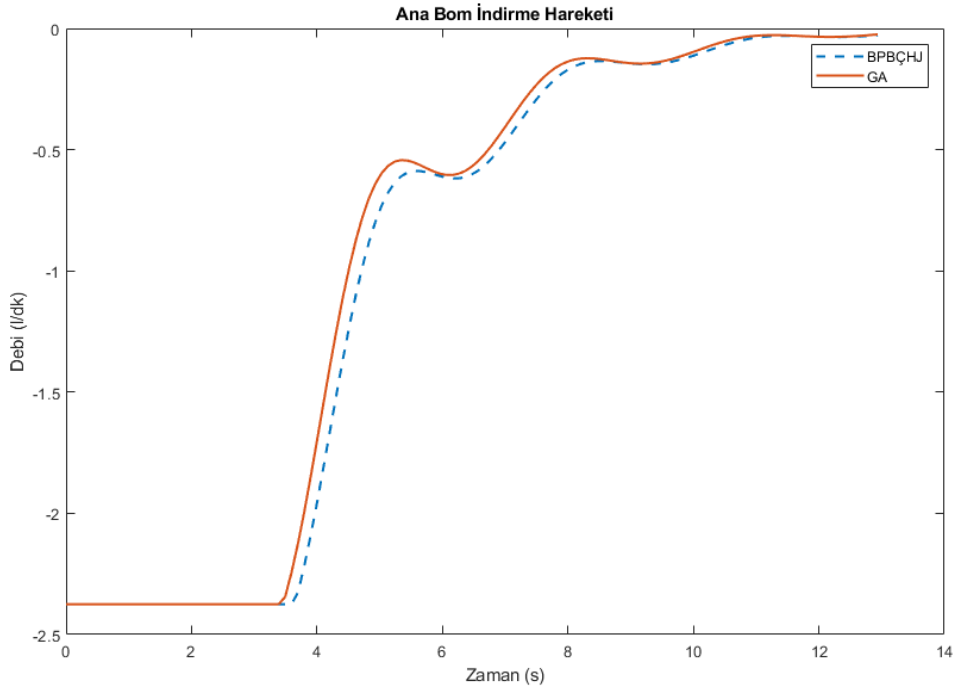
Çizelge 5.5 ABİH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	6.6154	32.5854	6.5155	7.5790	27.4283	6.4570
K_p	0.5142	0.8596	0.5308	0.4480	0.9212	0.5492
K_d	0.0549	0.9760	0.1047	0	1.2361	0.0242
K_i	0	0.0001	0	0	0.0001	0

BPBÇHJ Algoritması ve GA ile elde edilen en uygun PID katsayıları uygulandığında ve Ana Bom'un 20° indirmesi istendiğinde. Şekil 5.21'deki Ana Bom Açısı- Zaman grafiği elde edilmiştir. PID kontrolcü çıkışı Şekil 5.22'de gösterilmektedir.



Şekil 5.21. ABİH basamak cevabı



Şekil 5.22. ABİH kontrolcü çıkışı

5.7. 1. Kırma Hareketi Elde Edilen Sonuçlar

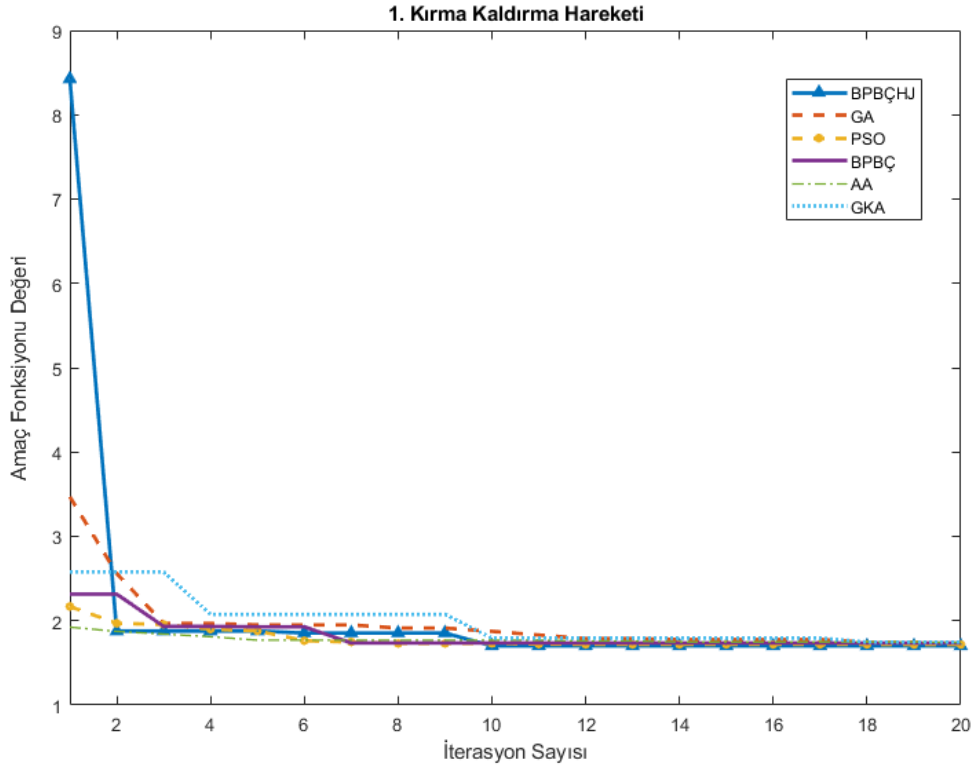
5.7.1. 1. kırma kaldırma hareketi

BKKH için Şekil 5.23'deki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında PSO'nun bu problem için diğer KA algoritmalarına nispeten daha düşük amaç fonksiyonu değerine sahip olduğu gözlenmektedir.

20 iterasyon sonunda BKKH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.6'da gösterilmektedir. PSO Algoritması, BPBÇHJ algoritması ile kıyaslandığında, BPBÇHJ algoritmasının daha düşük amaç fonksiyonu değerine sahip olduğu görülmektedir.

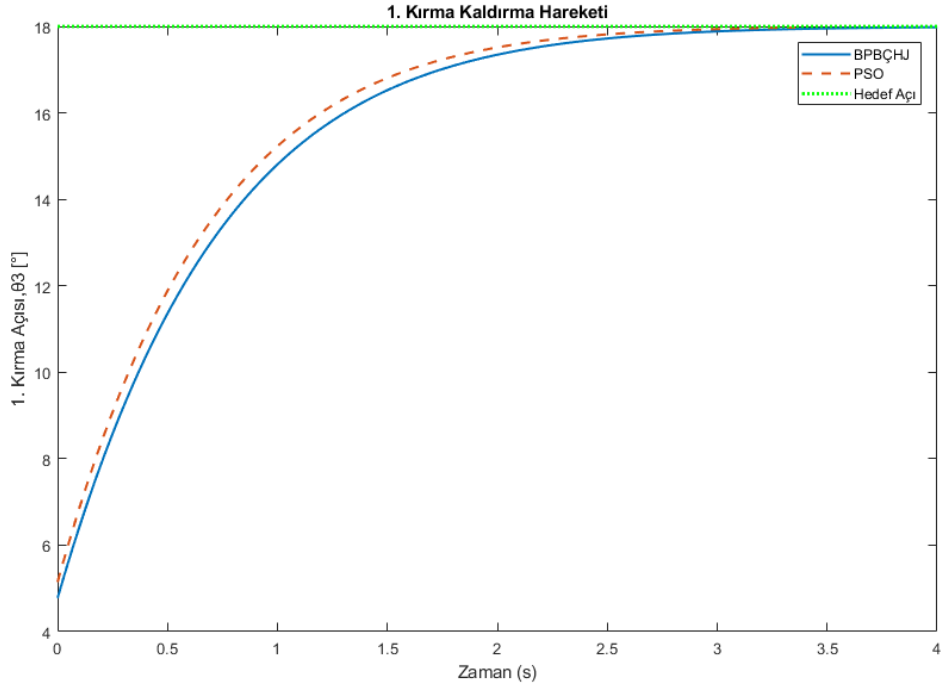
Çizelge 5.6 BKKH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	1.7407	1.7357	1.7222	1.7452	1.7172	1.7013
K_p	4	3.9583	3.9417	4	3.9983	3.6087
K_d	2.2926	2.0618	1.8144	2.3986	2.0502	1.8494
K_i	0.0429	0.0448	0.0382	0.0498	0.0457	0.0386

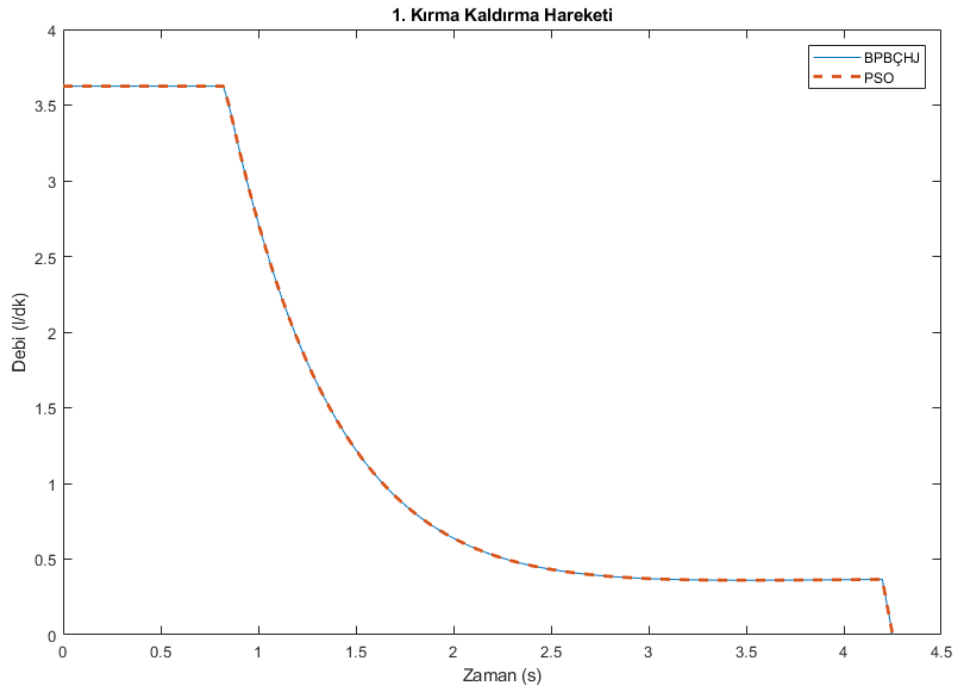


Şekil 5.23. BKKH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

BPBÇHJ ve PSO ile elde edilen en uygun PID katsayıları BKKH sistemine uygulandığında ve 1.Kırma'nın 18° kaldırması istendiğinde Şekil 5.24'deki 1.Kırma Açısı - Zaman grafiği elde edilmiştir. BKKH, PID kontrolcü çıkışı Şekil 5.25'de gösterilmiştir.



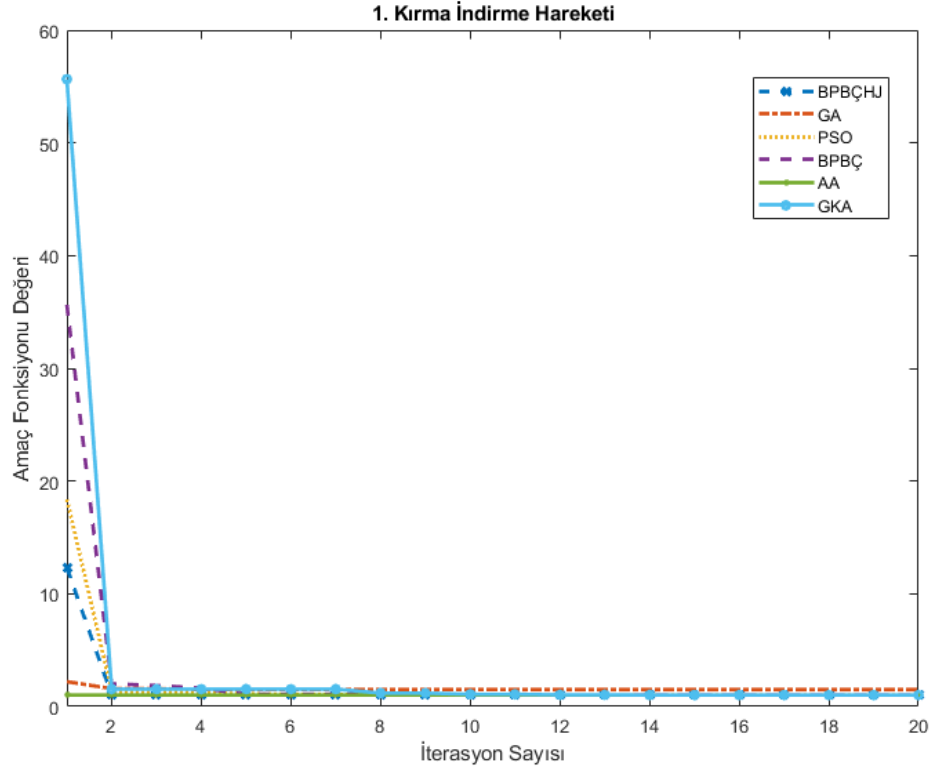
Şekil 5.24. BKKH basamak cevabı



Şekil 5.25. BKKH kontrolcü çıkışı

5.7.2. 1. kırma indirme hareketi

BKİH için Şekil 5.26'daki en iyi sonuçlar değerlendirildiğinde AA ve GKA'nın bu problem için diğer KA algoritmalarına nispeten daha düşük amaç fonksiyonu değerine sahip olduğu gözlenmektedir.



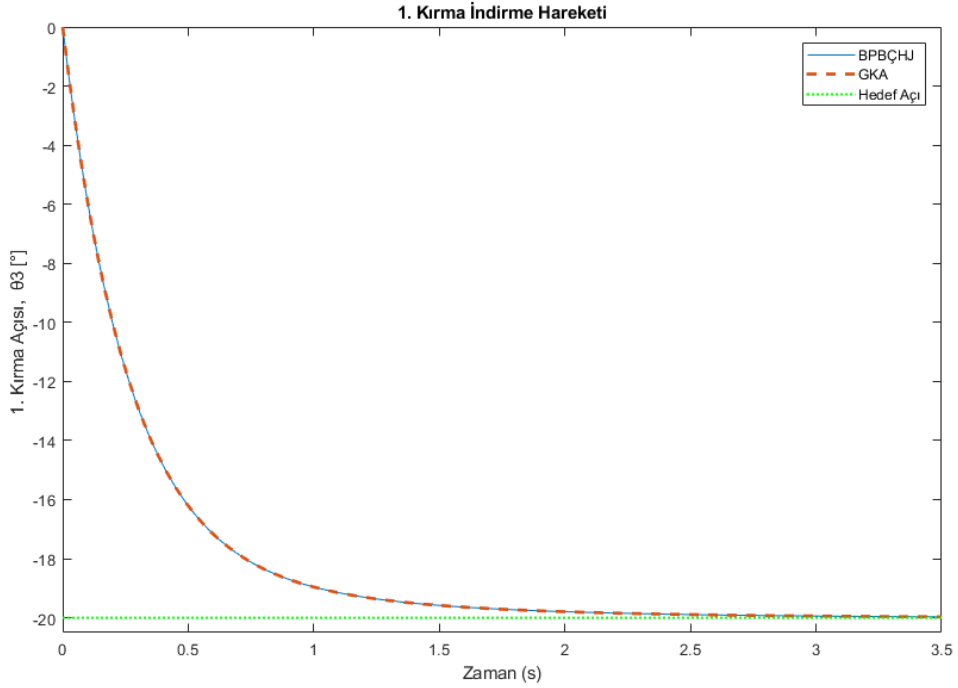
Şekil 5.26. BKİH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda BKİH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.7'de gösterilmektedir. AA ve GKA, BPBÇHJ algoritması ile kıyaslandığında iterasyonlar sonunda AA, GKA ve BPBÇHJ algoritmalarının amaç fonksiyonu değerlerinin eşit olduğu görülmektedir.

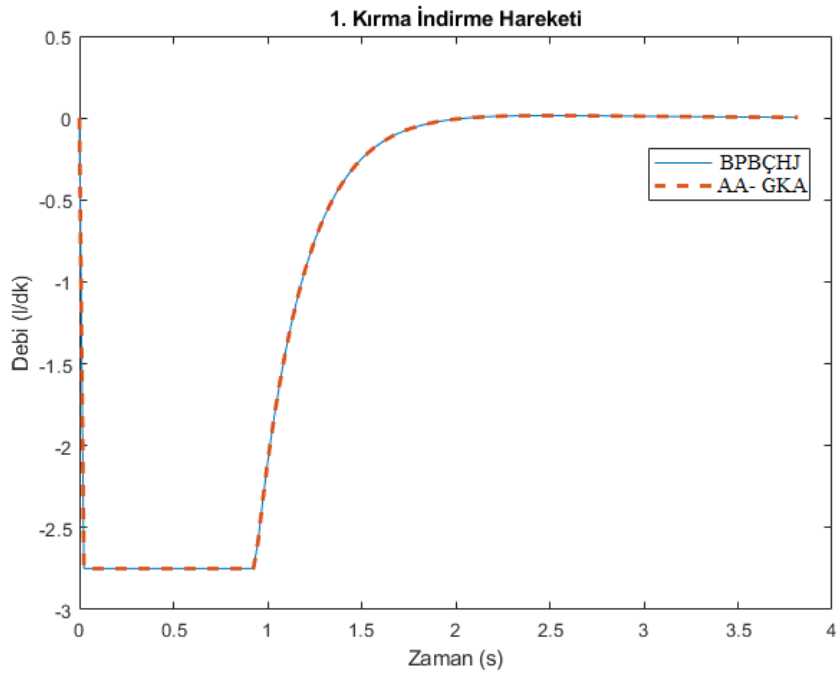
Çizelge 5.7 BKİH için 20 İterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	1.0162	1.0252	1.5	1.0162	1.0254	1.0162
K_p	4	3.9737	2.7522	4	3.9945	4
K_d	3	2.8372	2.0090	3	2.6481	3
K_i	0	0.0070	0.0049	0	0.0093	0

BPBÇHJ ve AA-GKA ile elde edilen PID katsayıları sisteme uygulandığında ve 1. Kırma'nın 20° indirilmesi istendiğinde Şekil 5.27'deki 1.Kırma Açısı - Zaman grafiği elde edilmiştir. BKİH, PID kontrolcü çıkışı Şekil 5.28'de gösterilmektedir.



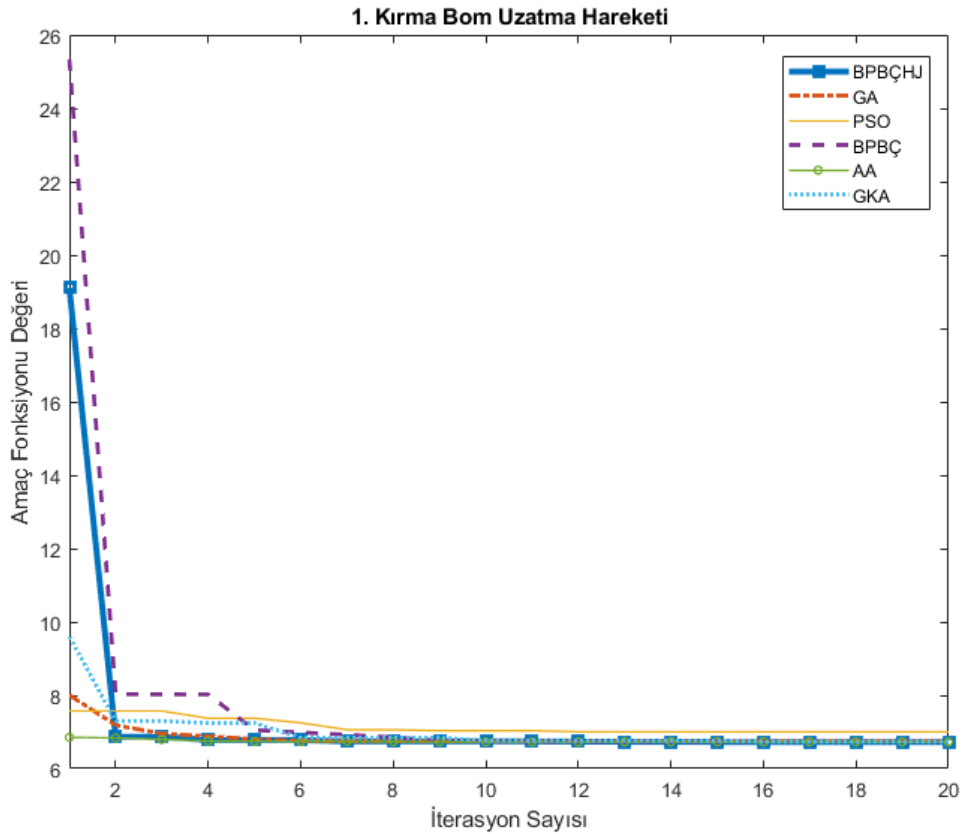
Şekil 5.27. BKİH basamak cevabı



Şekil 5.28. BKİH kontrolcü çıkışı

5.7.3. 1. kırma bom uzatma hareketi

BKBUH için en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında AA'nın diğer KA algoritmalarına göre nispeten daha düşük amaç fonksiyonu değerine sahip olduğu gözlenmektedir. Şekil 5.29'da algoritmaların BKBUH'ye ait amaç fonksiyonu değeri - iterasyon sayısı grafiği gösterilmektedir.



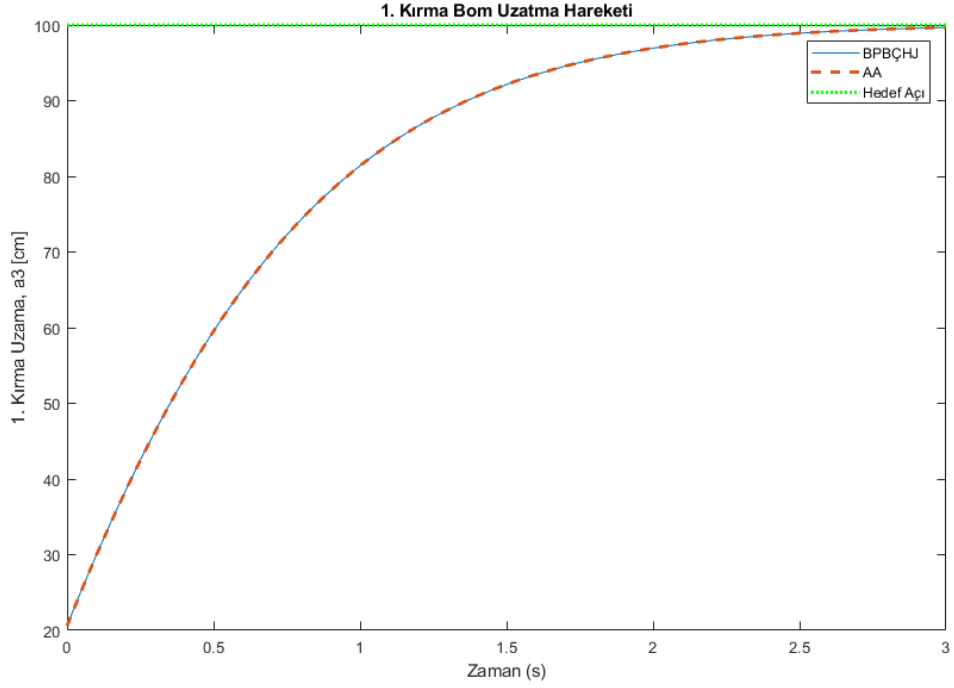
Şekil 5.29. BKBUH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda BKBUH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.8'de gösterilmektedir.

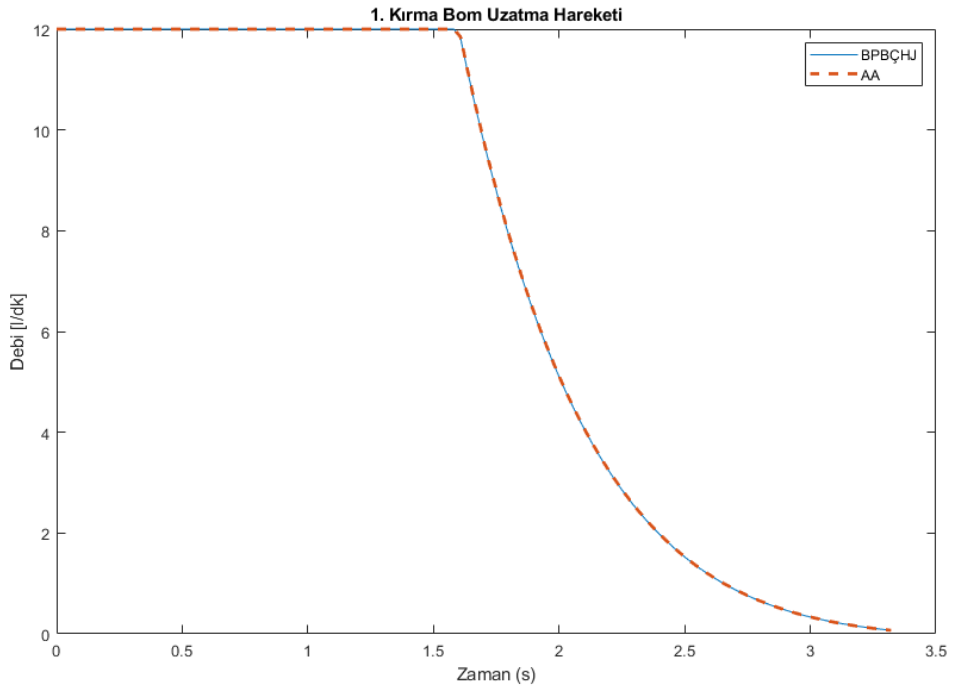
Çizelge 5.8 BKBUH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç fonksiyonu değeri	6.7259	6,733186	6,746646	6,73243	7,117453	6.7263
K_p	4	4	4	4	3.909598	4
K_d	1.154194	1.196434	1.272567	1.192072	2.613632	1.156319
K_i	0	0	0	0	0.000173	0.00003

BPBÇHJ ve AA ile elde edilen PID katsayıları BKBUH sistemine uygulandığında ve 1. Kırma'nın 100 cm bom uzatması istendiğinde Şekil 5.30'daki 1.Kırma Uzama - Zaman grafiği elde edilmiştir. BKBUH, PID kontrolcü çıkışı Şekil 5.31'de gösterilmektedir.



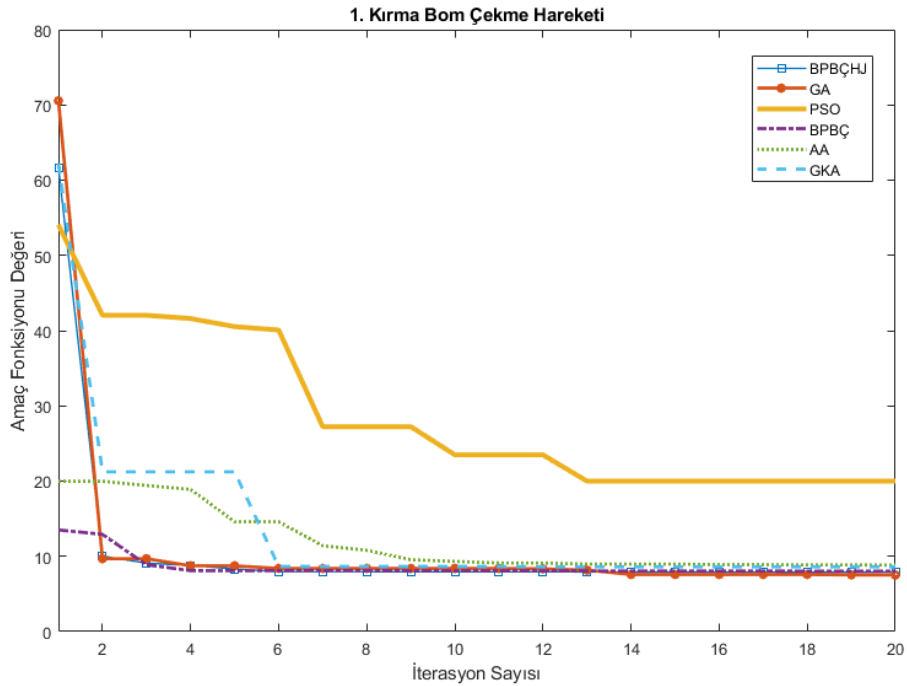
Şekil 5.30. BKBUH basamak cevabı



Şekil 5.31. BKBUH kontrolcü çıkışı

5.7.4. 1. kırma bom çekme hareketi

BKBÇH için en iyi sonuçlar değerlendirildiğinde algoritmalar arasında GA'nın diğer KA algoritmalarına göre nispeten daha düşük amaç fonksiyonu değerine sahip olduğu gözlenmektedir. Şekil 5.32'de algoritmaların BKBÇH'ye ait amaç fonksiyonu değeri-iterasyon sayısı grafiği gösterilmektedir.



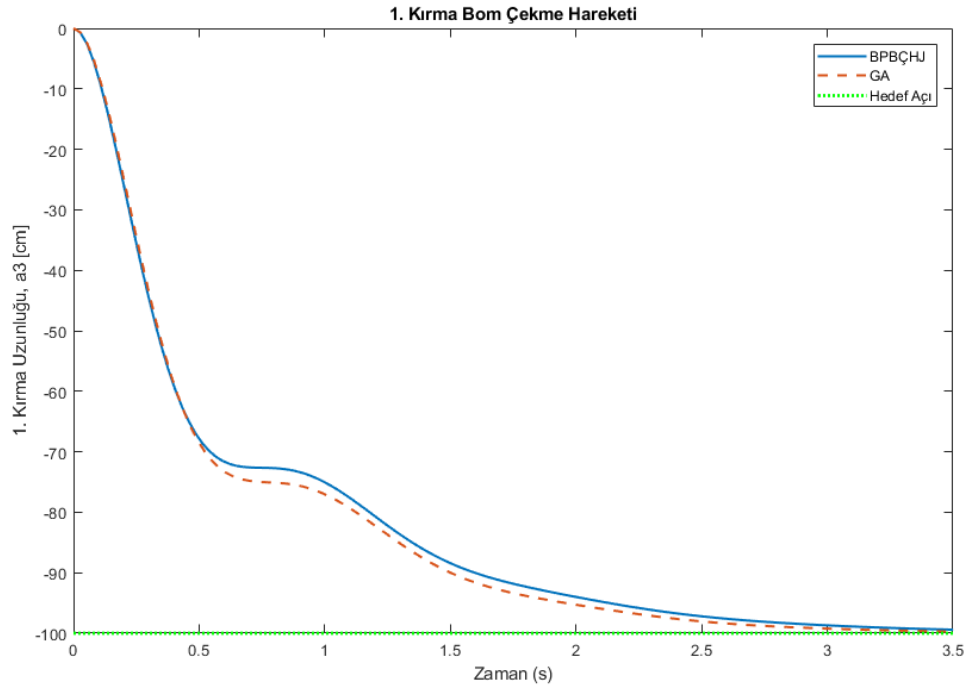
Şekil 5.32. BKBÇH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda BKBÇH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.9'da gösterilmektedir.

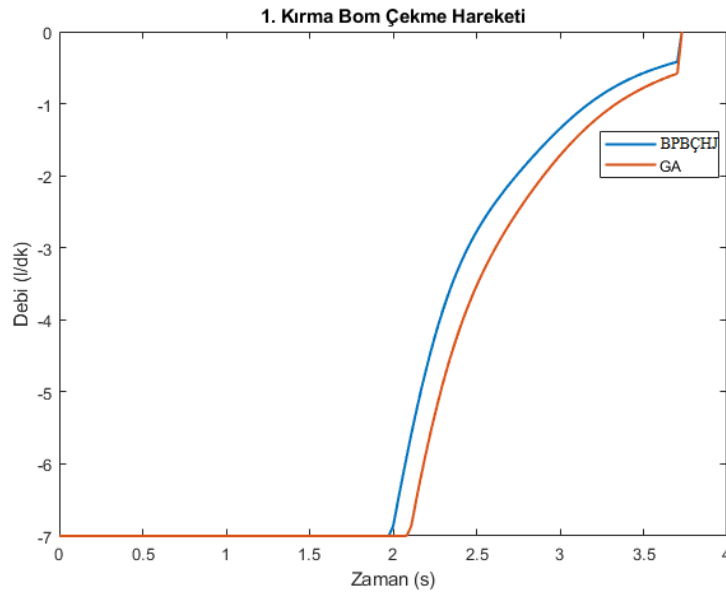
Çizelge 5.9 BKBÇH için 20 İterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç fonksiyonu değeri	8.833059	8.03163	7.52382644	8.612368	19.99019	7.907387
K_p	1.932986	1.790871	1.943164034	1.905917	3.328422	1.82354
K_d	0.470877	0.467065	0.428473787	0.435008	1.18753	0.508047
K_i	0	0	0.0009	0	0.0018	0

BPBÇHJ ve GA ile elde edilen PID katsayıları BKBÇH sistemine uygulandığında ve 1. Kırma'nın 100 cm bom çekmesi istendiğinde Şekil 5.33'Teki 1.Kırma Uzama - Zaman grafiği elde edilmiştir. BKBÇH, PID kontrolcü çıkışı Şekil 5.34'te gösterilmektedir.



Şekil 5.33. BKBÇH basamak cevabı

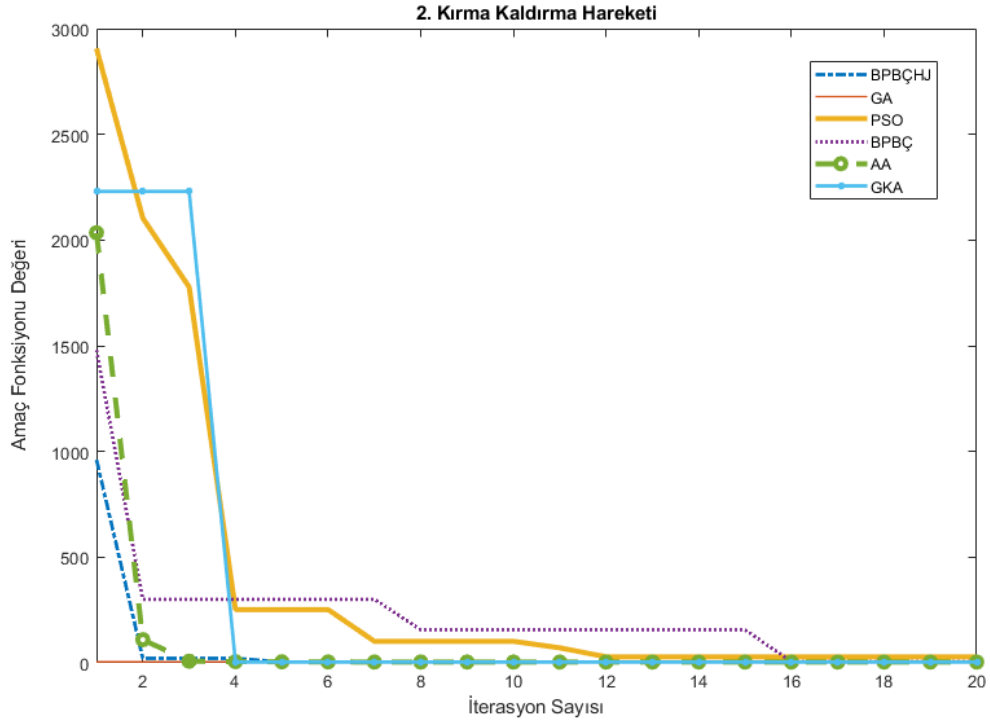


Şekil 5.34. BKBÇH Kontrolcü Çıkışı

5.8. 2. Kırma Hareketi Elde Edilen Sonuçlar

5.8.1. 2. kırma kaldırma hareketi

İKKH için Şekil 5.35'deki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında GKA'nın bu problem için diğer KA algoritmalarına nispeten daha iyi sonuç verdiği gözlenmektedir.



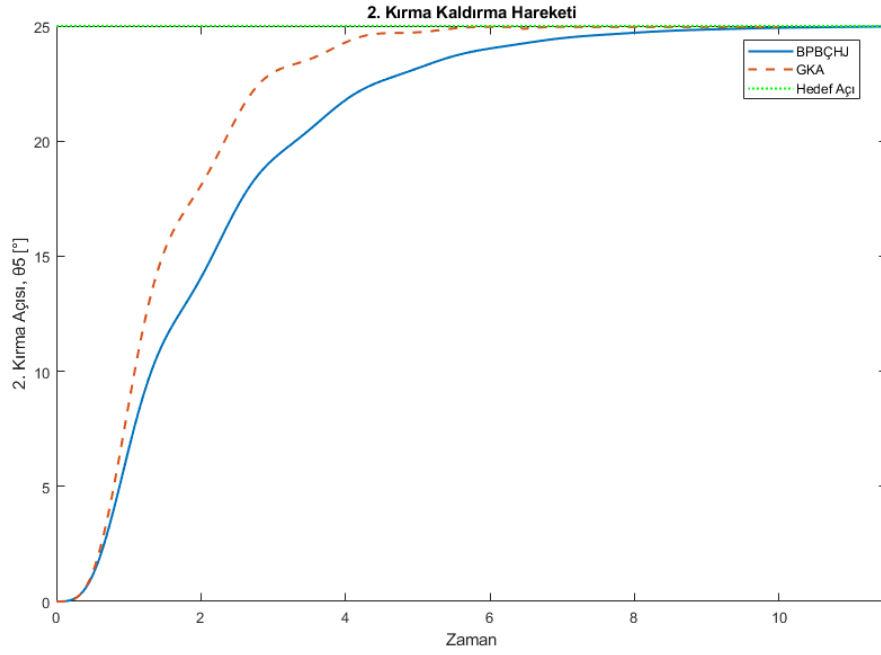
Şekil 5.35. İKKh arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda İKKh için algoritmaların elde ettiği minimum maliyet değeri ve elde edilen PID katsayıları Çizelge 5.10’da gösterilmektedir. İterasyonlar sonunda GKA nispeten daha iyi sonuçlar vermiştir. PSO ile elde edilen sonuç hariç diğer sonuçlar tatmin edici gözükmemektedir.

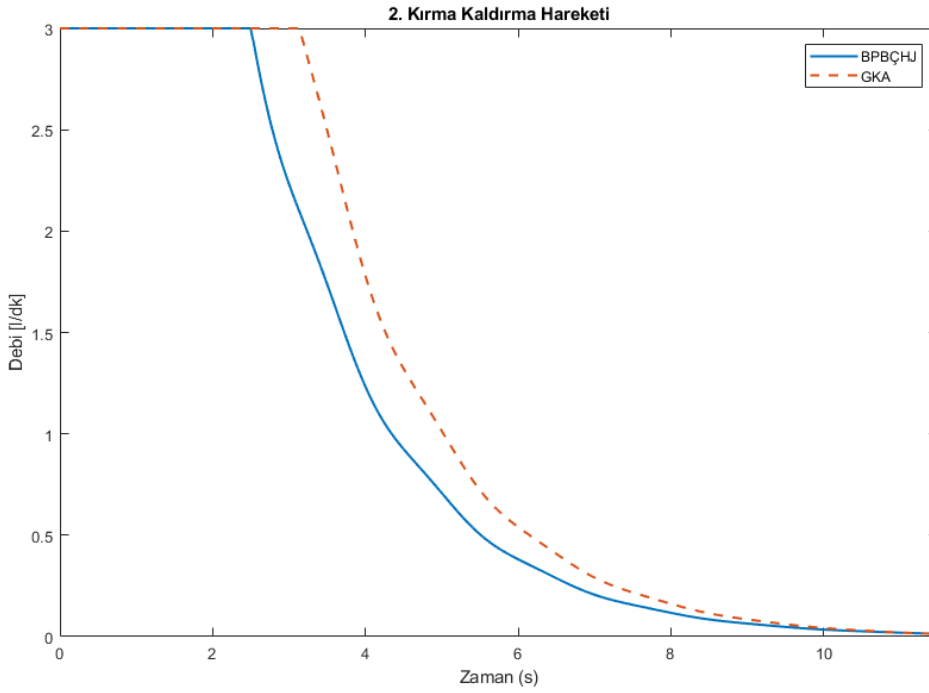
Çizelge 5.10 İKKh için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç fonksiyonu Değeri	3.880189	5.660552	3.902437	3.83234	28.57069	4.3496
K_p	0.549453	0.536552	0.55758	0.553142	0.636907	0.393094
K_d	0	0.072929	0.015625	0	0.104874	0.015856
K_i	0	7.15E-04	0	0	0.000317	9.97E-05

BPBÇHJ ve GKA ile elde edilen PID katsayıları sisteme uygulandığında ve 2. Kırma’nın 25° kaldırması istendiğinde Şekil 5.36’daki 2.Kırma Açısı - Zaman grafiği elde edilmiştir. İKKh, PID kontrolcü çıkışı Şekil 5.37’de gösterilmektedir.



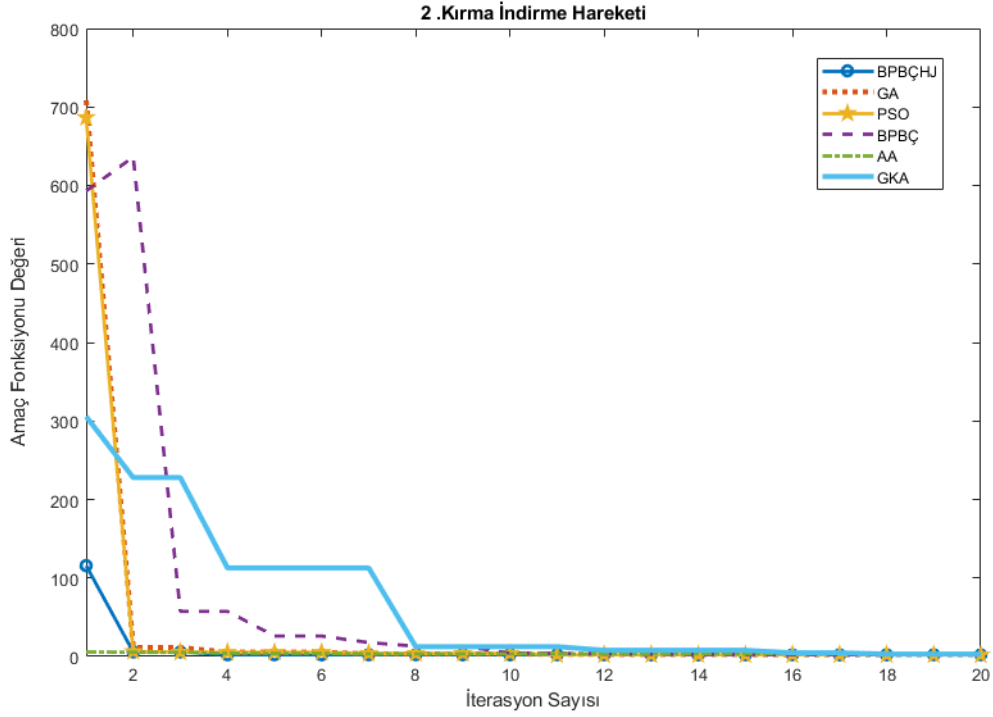
Şekil 5.36. İKKh basamak cevabı



Şekil 5.37. İKKh kontrolcü çıkışı

5.8.2. 2. kırma indirme hareketi

İKİH için Şekil 5.38'deki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında PSO'nun bu problem için diğer KA algoritmalarına nispeten daha iyi sonuç verdiği gözlenmektedir.



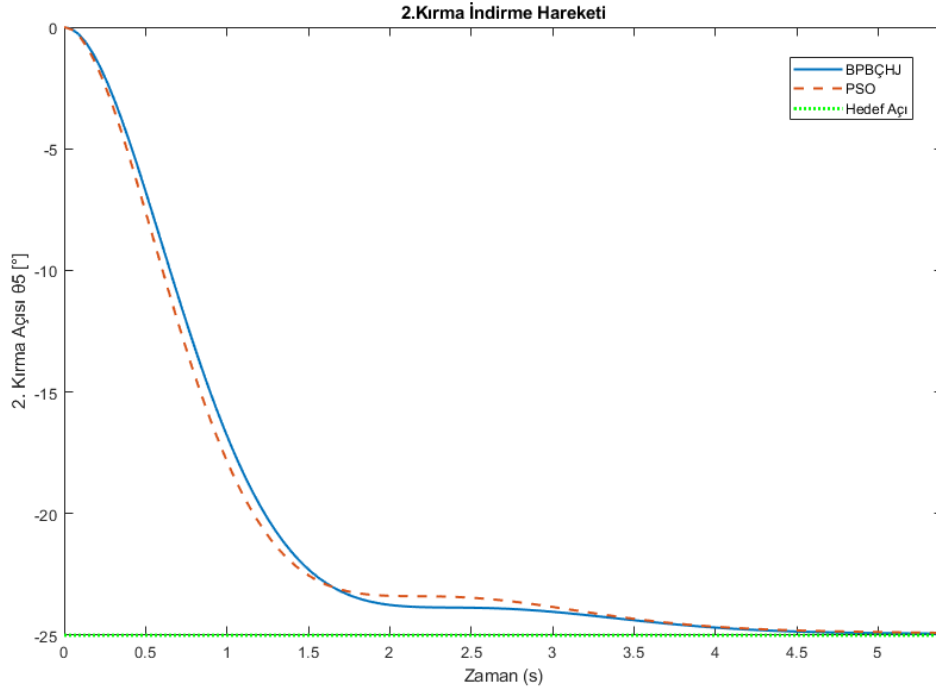
Şekil 5.38. İKİH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda İKİH için algoritmaların elde ettiği minimum maliyet değeri ve elde edilen PID katsayıları Çizelge 5.11’de gösterilmektedir. İterasyonlar sonunda PSO nispeten daha iyi sonuçlar vermiştir.

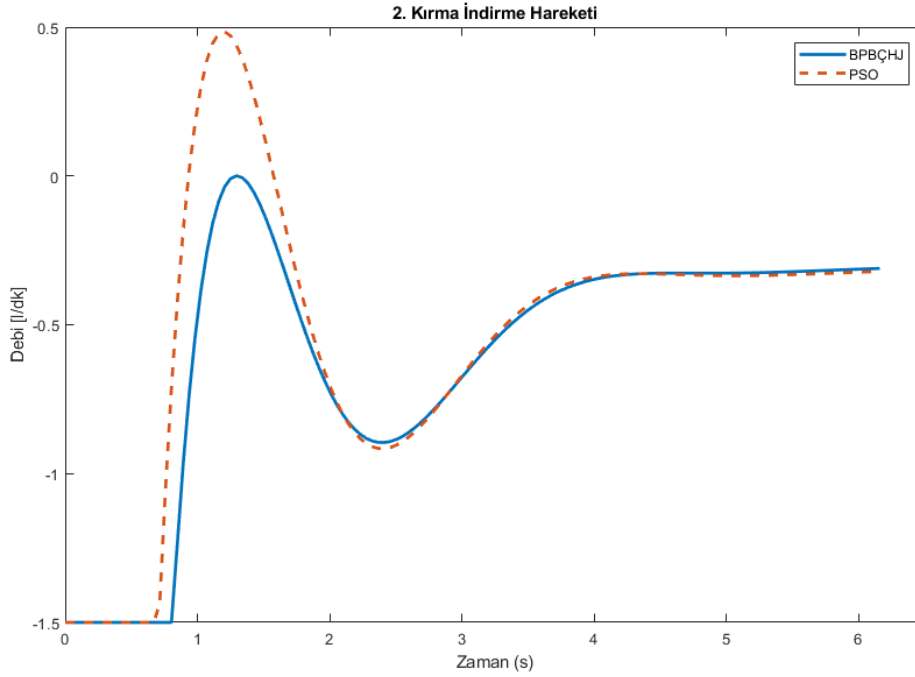
Çizelge 5.11 İKİH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	3.106166	2.17377	2.46187	3.04083	2.140654	2.182542
K_p	0.531964	0.558268	0.539321	0.538763	0.564329	0.553434
K_d	0.306654	0.272433	0.254396	0.283594	0.302894	0.255162
K_i	0	0.012402	0.008946	0	0.013391	0.0129

BPBÇHJ ve PSO ile elde edilen PID katsayıları sisteme uygulandığında ve 2. Kırma’nın 25° indirilmesi istendiğinde Şekil 5.39’daki 2.Kırma Açısı - Zaman grafiği elde edilmiştir. İKİH, PID kontrolcü çıkışı Şekil 5.40’da gösterilmektedir.



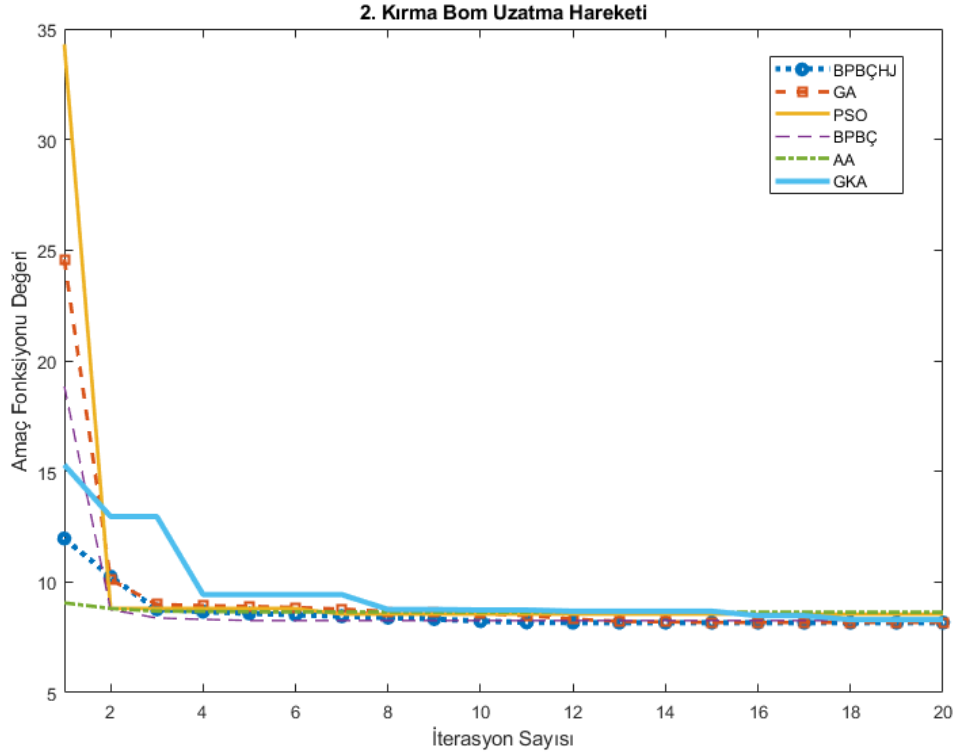
Şekil 5.39. İKİH basamak cevabı



Şekil 5.40. İKİH kontrolcü çıkışı

5.8.3. 2. kırma bom uzatma hareketi

İKBUH için Şekil 5.41'deki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında GA'nın bu problem için diğer KA algoritmalarına nispeten daha iyi sonuç verdiği gözlenmektedir.



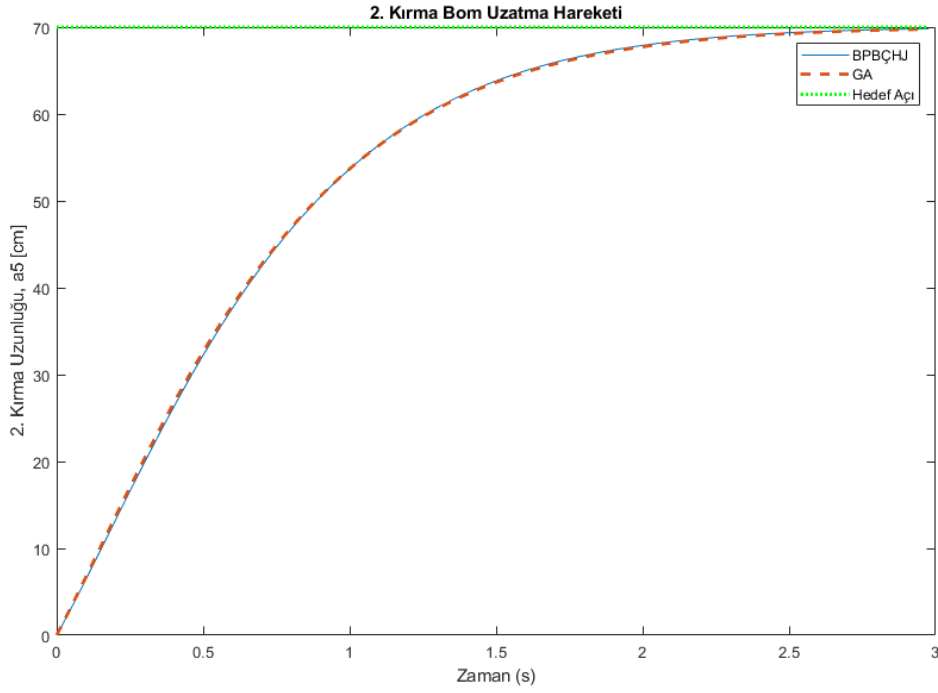
Şekil 5.41. İKBUH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda İKBUH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.12’de gösterilmektedir. İterasyonlar sonunda BPBÇHJ Algoritması nispeten daha iyi sonuçlar vermiştir.

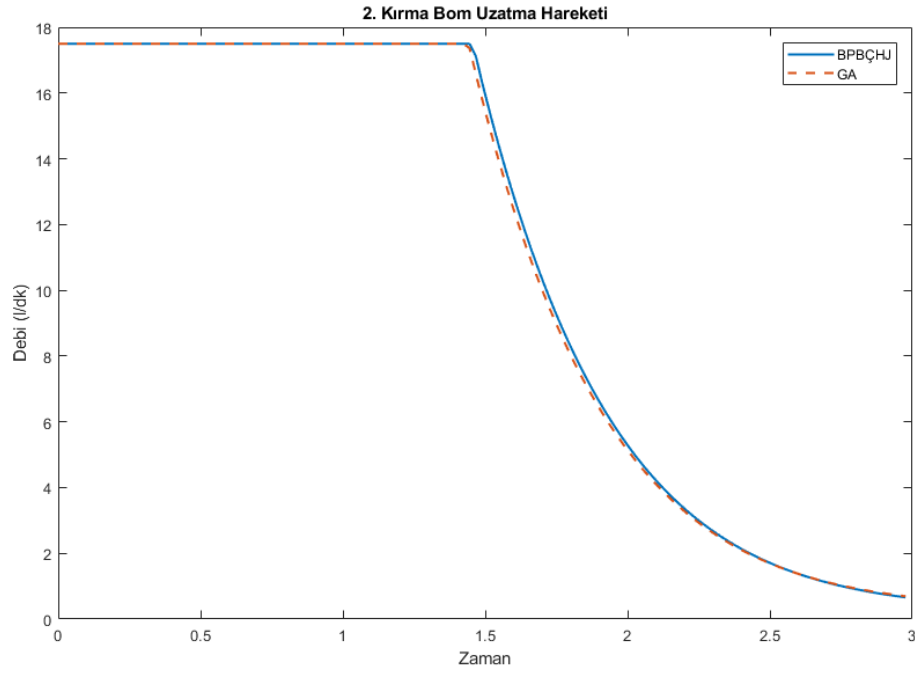
Çizelge 5.12 İKBUH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	8.634624	8.253741	8.172285	8.296014	8.548877	8.162314
K_p	4	4	4	4	3.968048	4
K_d	0.806137	0.843855	0.741303	1.065349	1.66006	0.698348
K_i	0	0.006	0.00719	0.0062	0.006686	0.006

BPBÇHJ ve GA ile elde edilen PID katsayıları sisteme uygulandığında ve 2. Kırma'nın 70 cm bom uzatması istendiğinde Şekil 5.42’deki 2.Kırma Uzama - Zaman grafiği elde edilmiştir. İKBUH, PID kontrolcü çıkışı Şekil 5.43’te gösterilmektedir.



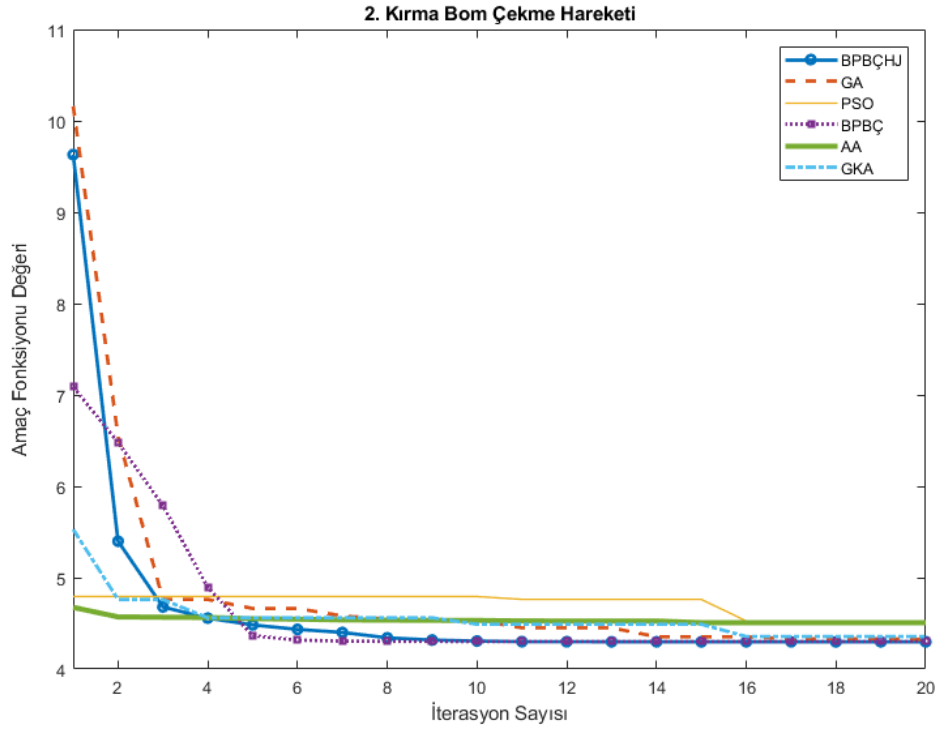
Şekil 5.42. İKBUH basamak cevabı



Şekil 5.43. İKBUH kontrolcü çıkışı

5.8.4. 2. kırma bom çekme hareketi

İKBUH için Şekil 5.44'teki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında BPBÇ Algoritması'nın bu problem için diğer KA algoritmalarına nispeten daha iyi sonuç verdiği gözlenmektedir.



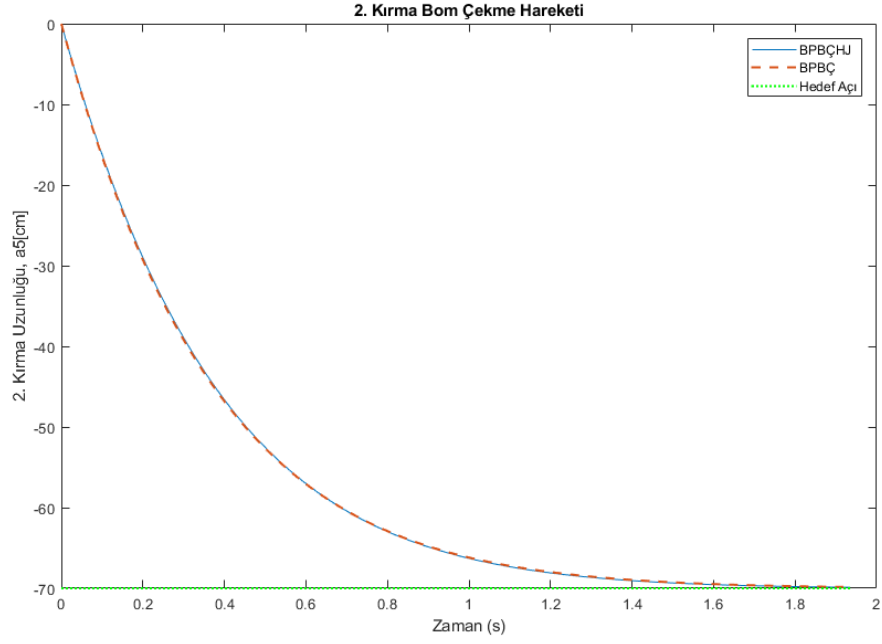
Şekil 5.44. İKBÇH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda İKBÇH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.13’de gösterilmektedir. İterasyonlar sonunda BPBÇHJ Algoritması nispeten daha iyi sonuçlar vermiştir.

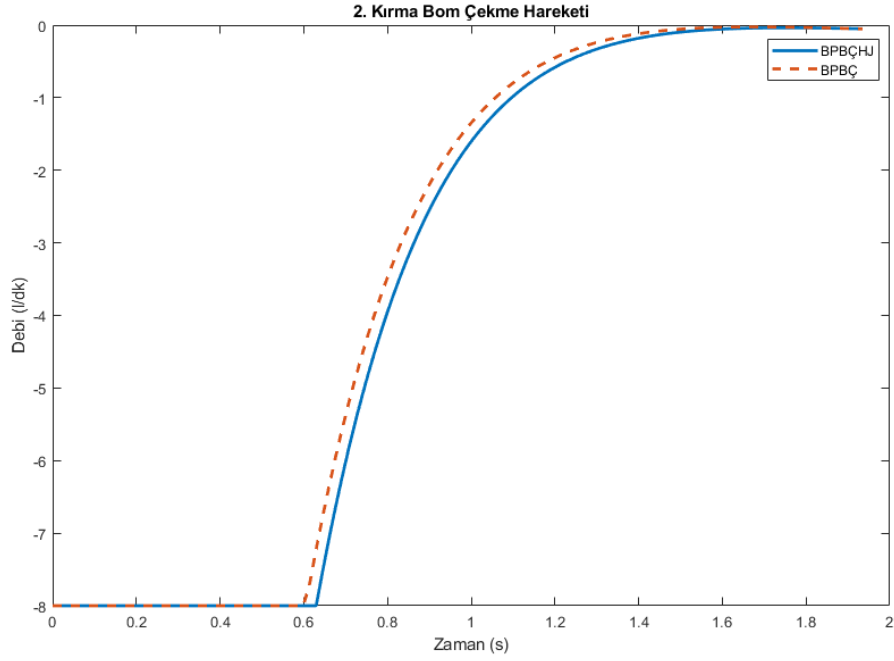
Çizelge 5.13 İKBÇH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	4.507262	4.301253	4.323652	4.355642	4.525107	4.299071
K_p	4	4	3.978493	4	3.819961	4
K_d	1.447944	1.102498	1.076922	1.427292	1.081676	1.080742
K_i	0.00194	0.006766	0.00551	0.00618	0.00493	0.00622

BPBÇHJ ve BPBÇ ile elde edilen PID katsayıları sisteme 2. Kıрма’nın 70 cm bom çekmesi istendiğinde Şekil 5.45’teki 2.Kırma Uzama - Zaman grafiği elde edilmiştir. İKBHU, PID kontrolcü çıkışı Şekil 5.46’da gösterilmektedir.



Şekil 5.45. İKBÇH basamak cevabı

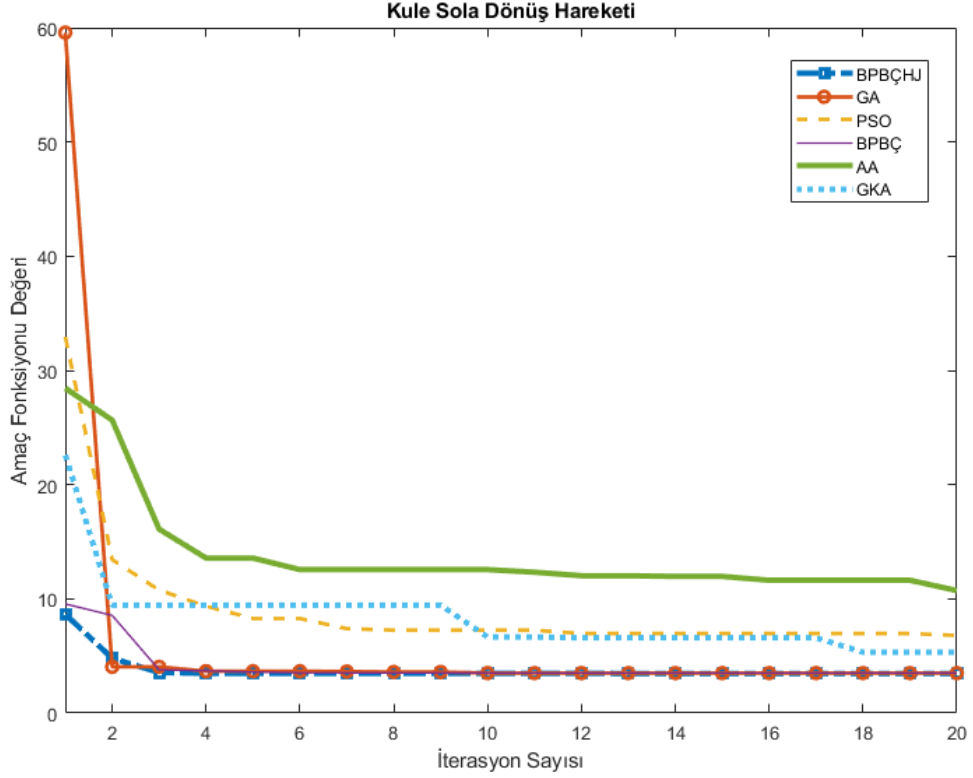


Şekil 5.46. İKBÇH kontrolcü çıkışı

5.9. Kule Dönüş Hareketi Elde Edilen Sonuçlar

5.9.1. Kule sola dönüş hareketi

KSODH için Şekil 5.47'deki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında GA'nın bu problem için diğer KA algoritmalarına nispeten daha iyi sonuç verdiği gözlenmektedir.



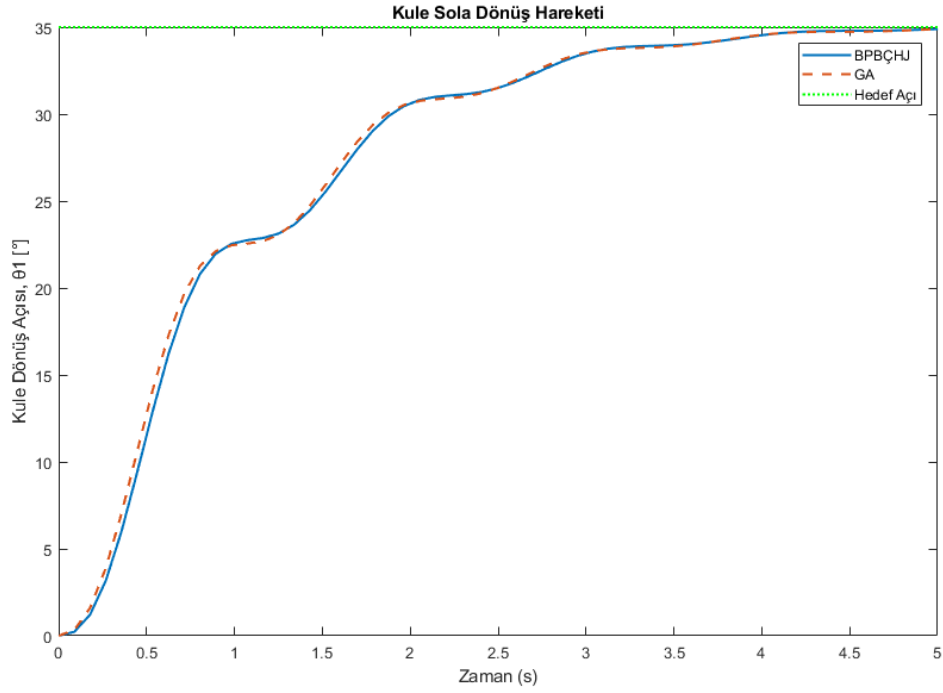
Şekil 5.47. KSODH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda KSODH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.14’de gösterilmektedir. İterasyonlar sonunda BPBÇHJ Algoritması nispeten daha iyi sonuçlar vermiştir.

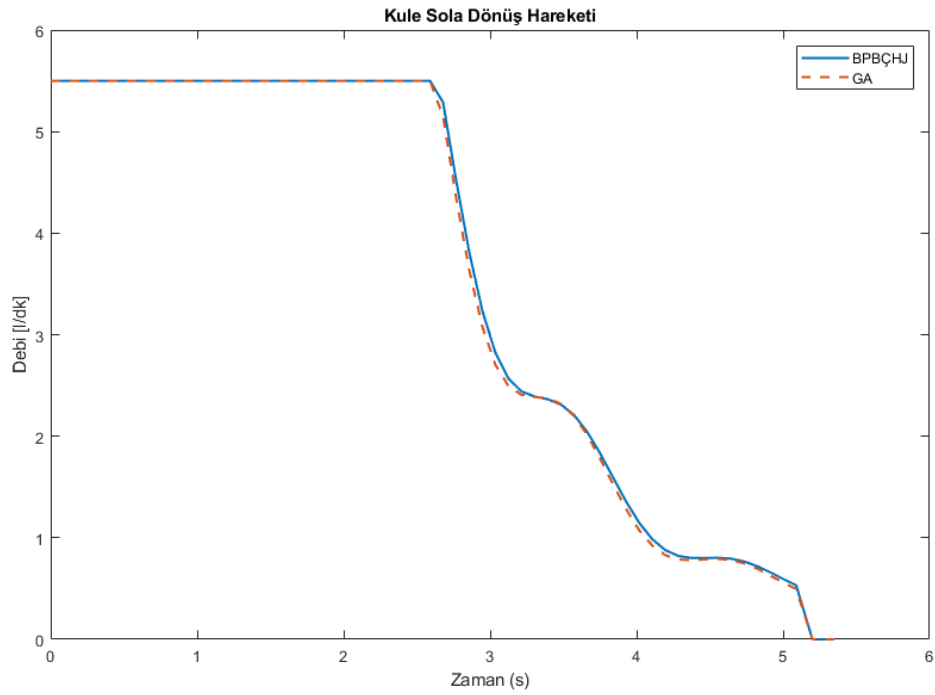
Çizelge 5.14 KSODH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	10.72315	3.505241	3.500113	5.325143	6.796168	3.460180707
K_p	2.747876	2.022995	1.874407	1.230656	3.288898	1.835119054
K_d	1.705271	1.781333	0.114843	0.079482	1.731759	0.060404334
K_i	0.0189	0.00662	0.010794	0.00388	0.0171	0.0113

BPBÇHJ ve GA ile elde edilen PID katsayıları sisteme uygulandığında ve Kule’nin 35° dönmesi istendiğinde Şekil 5.48’deki Kule Açısı - Zaman grafiği elde edilmiştir. KSODH, PID kontrolcü çıkışı Şekil 5.49’da gösterilmektedir.



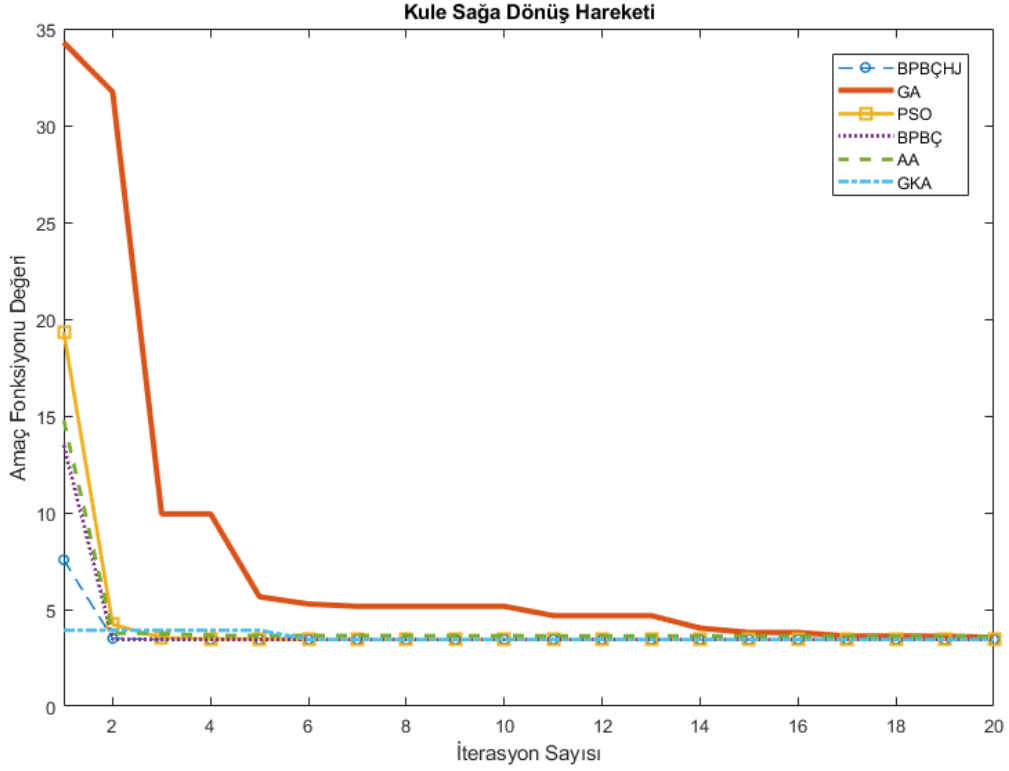
Şekil 5.48. KSODH basamak cevabı



Şekil 5.49. KSODH kontrolcü çıkışı

5.9.2. Kule sağa dönüş hareketi

KSADH için Şekil 5.50'deki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında PSO'nun bu problem için diğer KA algoritmalarına nispeten daha iyi sonuç verdiği gözlenmektedir.



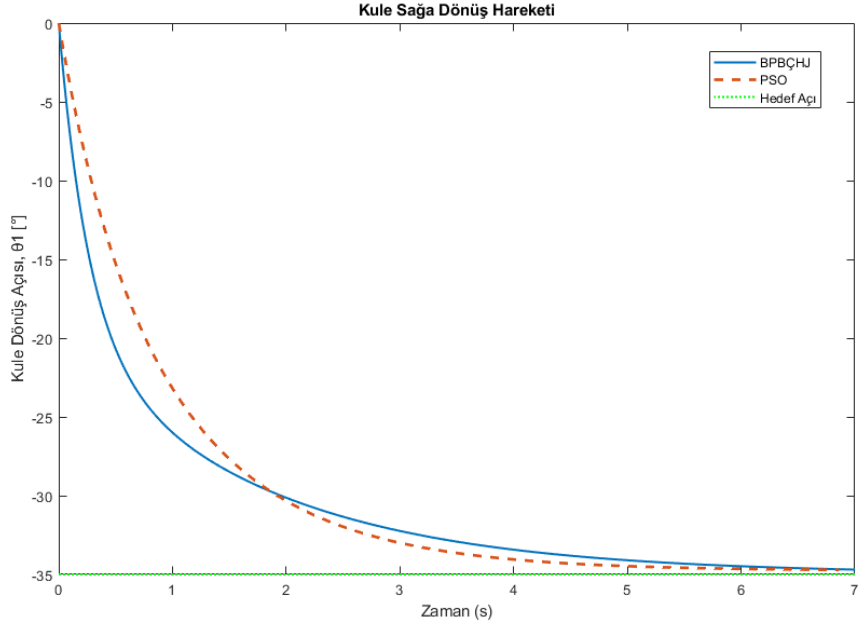
Şekil 5.50. KSADH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda KSADH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.15’de gösterilmektedir. İterasyonlar sonunda BPBÇHJ Algoritması nispeten daha iyi sonuçlar vermiştir.

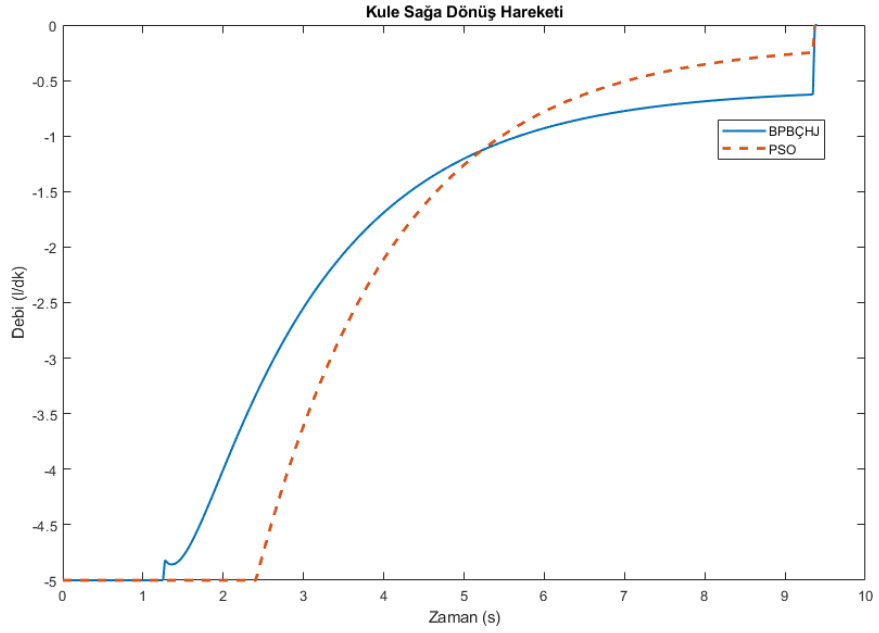
Çizelge 5.15 KSADH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	3.605696	3.451466	3.559696	3.451769	3.450346	3.450125
K_p	1.899092	2.151266	1.821251	3.454447	1.919938	2.03427
K_d	1.357433	2.474786	0.116826	1.779932	1.130593	2.232409
K_i	0	0.009168	0.0118	0.0359	0.00321	0.0162

BPBÇHJ ve PSO ile elde edilen PID katsayıları sisteme uygulandığında ve Kule'nin -35° dönmesi istendiğinde Şekil 5.51'deki Kule Açısı - Zaman grafiği elde edilmiştir. KSADH, PID kontrolcü çıkışı Şekil 5.52'de gösterilmektedir.



Şekil 5.51. KSADH basamak cevabı

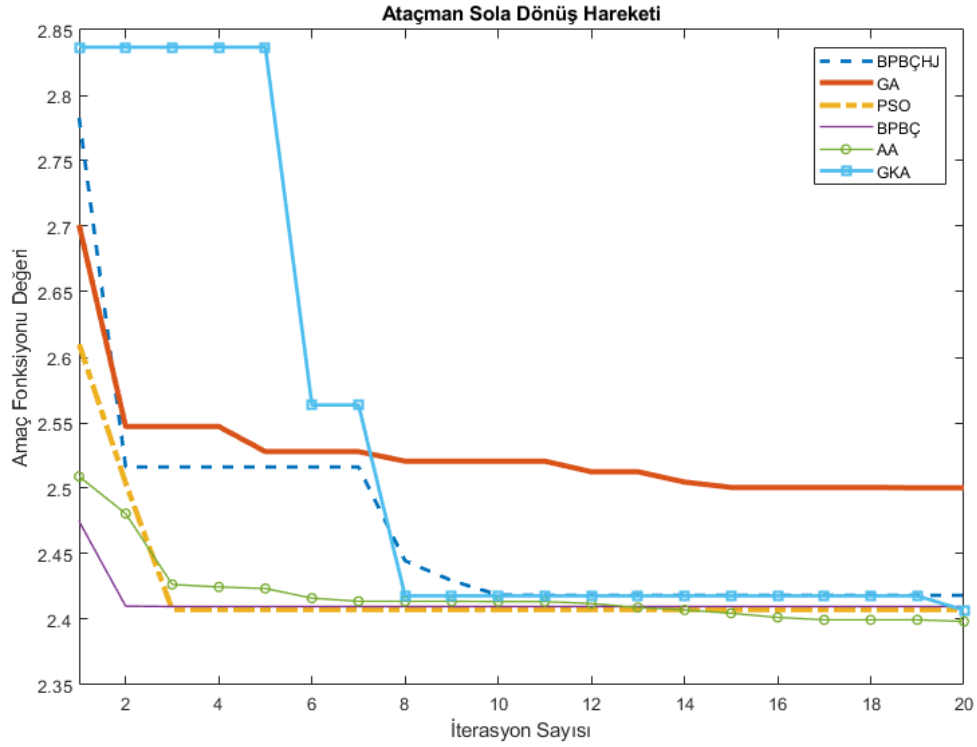


Şekil 5.52. KSADH kontrolcü çıkışı

5.10. Ataçman Dönüş Hareketi Elde Edilen Sonuçlar

5.10.1. Ataçman sola dönüş hareketi

ASODH için Şekil 5.53'teki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında GKA ve AA'nın bu problem için diğer KA algoritmalarına nispeten daha iyi sonuç verdiği gözlenmektedir.



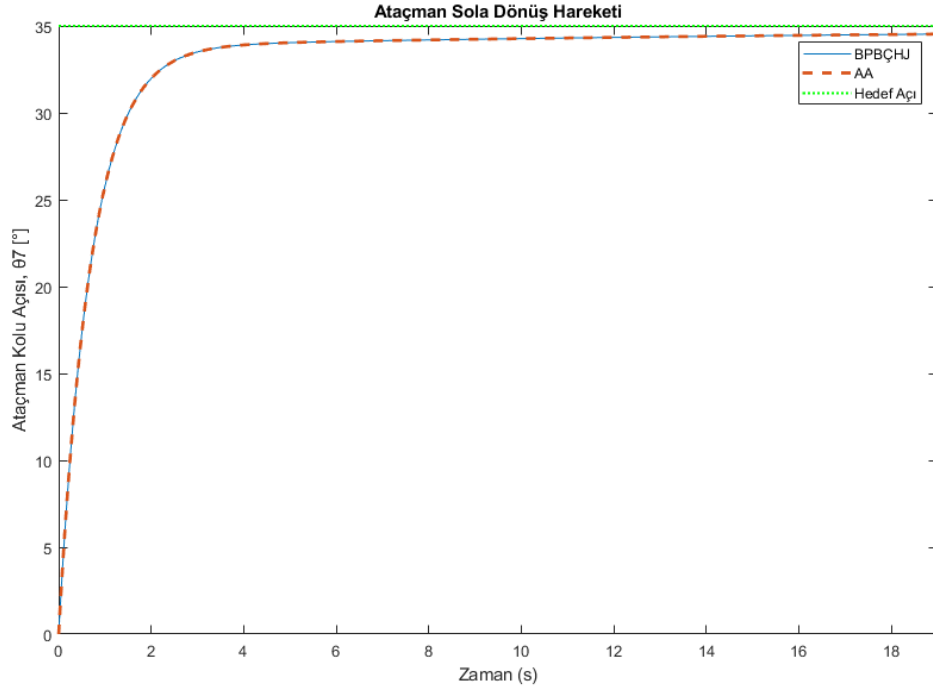
Şekil 5.53. ASODH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda ASODH için algoritmaların elde ettiği minimum amaç fonksiyonu değeri ve elde edilen PID katsayıları Çizelge 5.16’da gösterilmektedir. İterasyonlar sonunda AA nispeten daha iyi sonuçlar vermiştir.

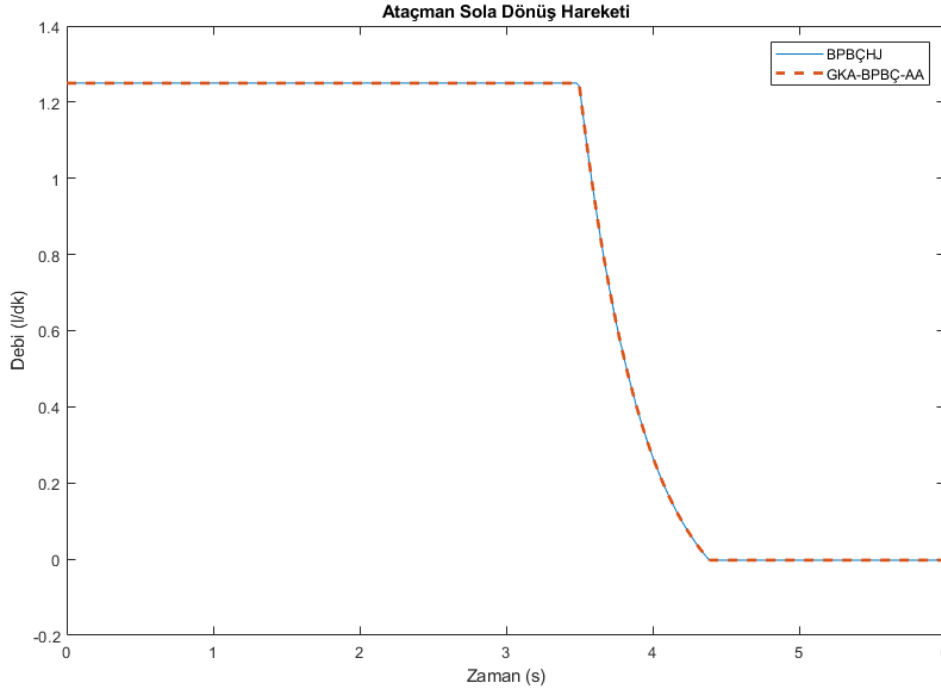
Çizelge 5.16 KSODH için 20 iterasyon sonunda elde edilen amaç fonksiyonu değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Amaç Fonksiyonu Değeri	2.398377	2.409749	2.500003	2.406968	2.407425	2.418299
K_p	4	4	3.838989	4	3.987112	4
K_d	3	3	1.999023	3	2.986688	3
K_i	0.135	0.108	0.243	0.114	0.13695	0.0897

BPBÇHJ ve AA ile elde edilen PID katsayıları sisteme uygulandığında ve Ataçman Kolu’nun 35° dönmesi istendiğinde Şekil 5.54’teki Ataçman Kolu Açısı - Zaman grafiği elde edilmiştir. ASODH, PID kontrolcü çıkışı Şekil 5.55’te gösterilmektedir.



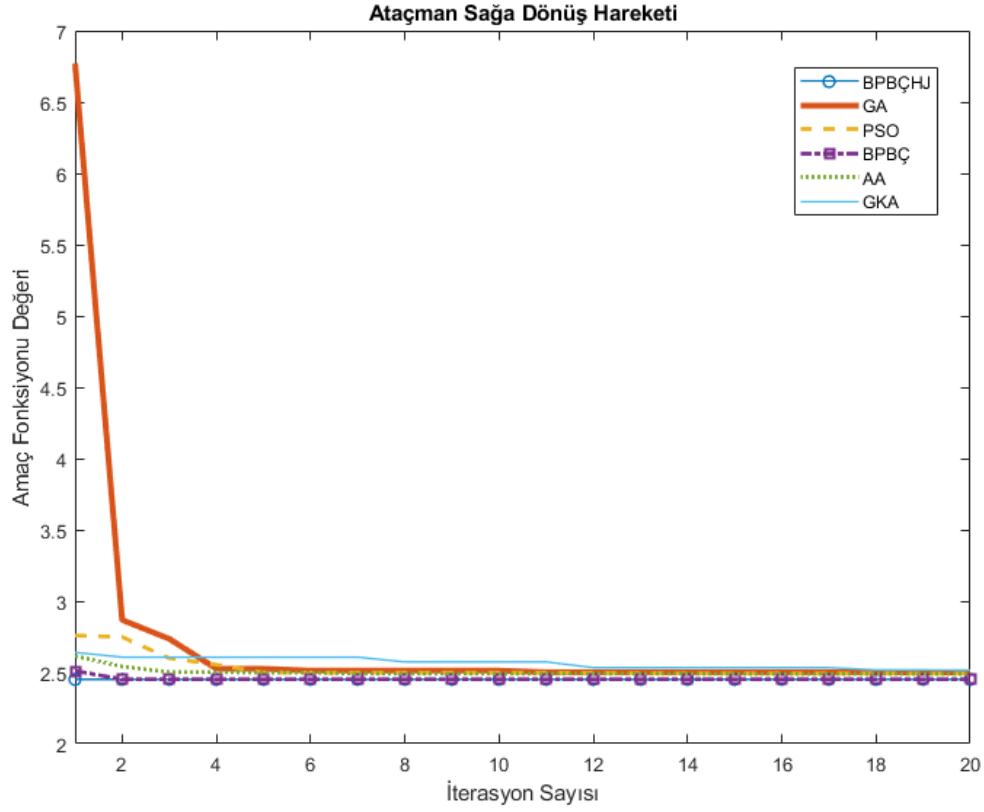
Şekil 5.54. ASODH basamak cevabı



Şekil 5.55. ASODH kontrolcü çıkışı

5.10.2. Ataçman sağa dönüş hareketi

ASADH için Şekil 5.56'daki en iyi sonuçlar değerlendirildiğinde bilinen algoritmalar arasında BPBÇ Algoritması'nın bu problem için diğer KA algoritmalarına nispeten daha iyi sonuç verdiği gözlenmektedir.



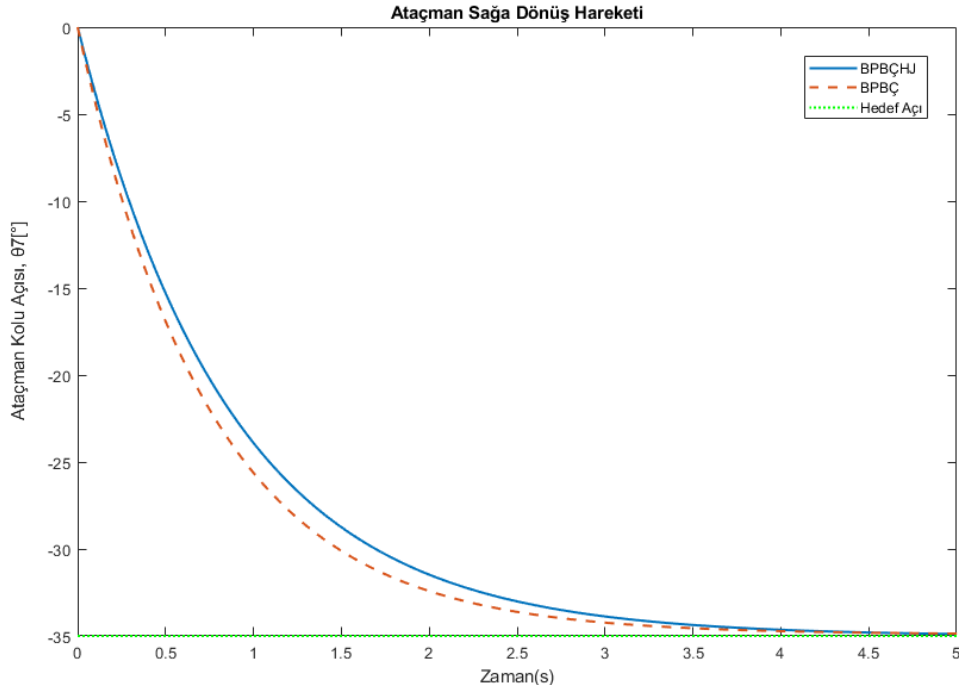
Şekil 5.56. ASADH arama algoritmaları amaç fonksiyonu değerlerinin iterasyon sayısına göre değişimi

20 iterasyon sonunda ASADH için algoritmaların elde ettiği minimum maliyet değeri ve elde edilen PID katsayıları Çizelge 5.17’de gösterilmektedir. İterasyonlar sonunda BPBÇHJ Algoritması nispeten daha iyi sonuç vermiştir.

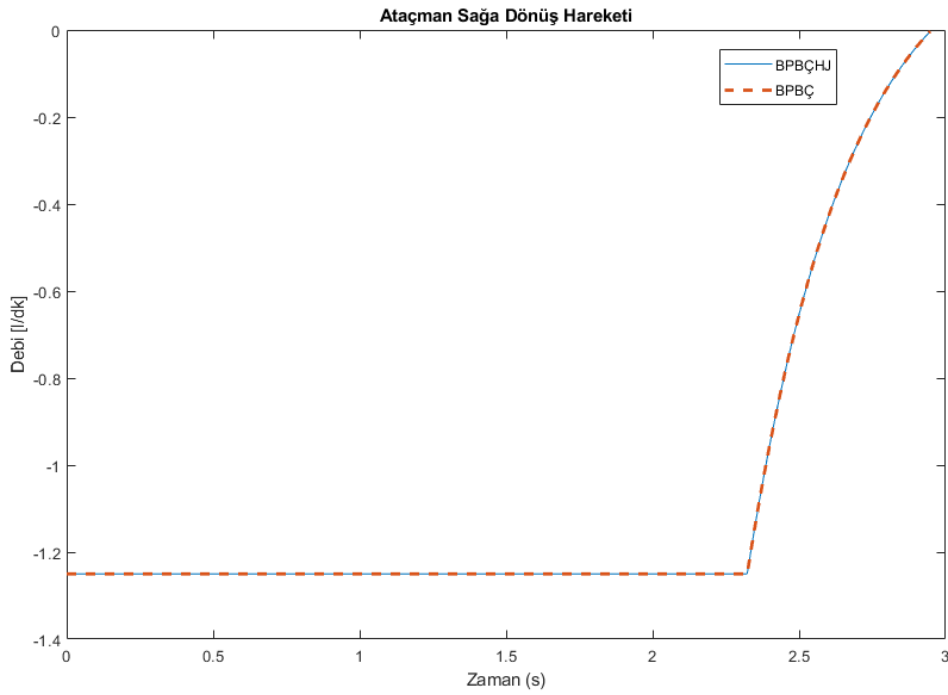
Çizelge 5.17 ASADH için 20 iterasyon sonunda elde edilen maliyet değeri ve PID katsayıları

Algoritmalar	AA	BPBÇ	GA	GKA	PSO	BPBÇHJ
Maliyet	2.489548	2.451895	2.500748	2.515977	2.491717	2.45095
K_p	3.455618	2.498844	2.0938	3.301027	3.44971	2.312471
K_d	3	2.356215	2.005355	3	2.997925	2.048165
K_i	0.215	0.237887	0.177	0.148	0.209697	0.205

BPBÇHJ ve BPBÇ ile elde edilen PID katsayıları sisteme uygulandığında ve Ataçman Kolu’nun -35° dönmesi istendiğinde Şekil 5.57’deki Ataçman Kolu Açısı - Zaman grafiği elde edilmiştir. ASADH, PID kontrolcü çıkışı Şekil 5.58’de gösterilmektedir.



Şekil 5.57. ASADH basamak cevabı



Şekil 5.58. ASADH kontrolcü çıkışı

5.11. Tüm Hareketler İçin Elde Edilen Sonuçlar

Tüm hareketler için elde edilen en düşük maliyet değerleri ve en kötü maliyet değerleri Çizelge 5.18'de gösterilmektedir. BPBÇHJ Algoritması aracılığıyla 9 hareket

için en iyi sonucu elde edilmiştir. GA ve PSO aracılığıyla 1'er hareket için en iyi sonuçlar elde edilmiştir. GKA ve AA aracılığıyla ise 2'şer hareket için en iyi sonuç elde edilmiştir.

Çizelge 5.18. Robot manipülatör hareketlerinin algoritmalara göre en iyi ve en kötü değerleri

Hareketler	BPBÇHJ	GA	PSO	BPBÇ	AA	GKA
	J_{ei}	J_{ei}	J_{ei}	J_{ei}	J_{ei}	J_{ei}
AKH	3,79	3,81	3,82	3,80	4,07	4,09
AİH	6,46	6,52	27,43	32,58	6,61	7,58
BKKH	1,70	1,72	1,72	1,74	1,74	1,75
BKİH	1,016	1,50	1,94	1,025	1,016	1,016
BKBUH	6.726	6.746	7.004	6.733	6.726	6.732
BKBÇH	7.91	7.52	19.99	8.03	8.83	8.61
İKKH	4,35	3,90	28,84	5,66	3,88	3,83
İKİH	2,18	2,46	2,14	2,17	3,11	3,04
İKBUH	8.162	8.172	8.549	8.253	8.634	8.296
İKBÇH	4.299	4.323	4.525	4.301	4.507	4.356
KSODH	3.46	3.5	6.98	3.50	10.72	5.32
KSADH	3.45	3.56	3.451	3.452	3,34	3.61
ASADH	2.451	2.5	2.49	2.452	2.49	2.51
ASODH	2.42	2.5	2.41	2.41	2.4	2.41

Kullanılan tüm algoritmalar optimum PID katsayıları eldesi için iyi sonuçlar vermiştir. Önerilen BPBÇHJ Algoritması aracılığıyla bu problem için kabul edilebilir sonuçlar elde edilmiştir.

6. SONUÇ VE ÖNERİLER

Bu çalışmada çok serbestlik dereceli hidrolik tahrikli robot manipülatörün en uygun PID katsayılarının elde edilmesi amaçlanmıştır. Her bir uzvun kaldırma-indirme veya uzatma-çekme hareketleri MSI'da ölçülen giriş ve çıkış verilerine göre deneysel olarak transfer fonksiyonu formatında modellenmiştir. Elde edilen transfer fonksiyonlarının tutarlılığı sistemden ölçülen giriş ve çıkış verileriyle test edilmiştir. Bu transfer fonksiyonları kullanılarak her bir sistemin kapalı çevrim basamak cevabına göre GA, PSO, BPBÇ, AA, GKA ve bu çalışmada önerilen BPBÇHJ Algoritması ile önerilen amaç fonksiyonu minimize edilerek en uygun PID katsayıları elde edilmiştir. İlk olarak literatürde çokça kullanılan algoritmalar birbiriyle kıyaslanmış, bu algoritmalar arasında en iyi olan algoritma ile BPBÇHJ Algoritması kıyaslanmıştır. Kullanılan algoritmaların hepsinde rastgelelik içeren terimler bulunduğundan dolayı algoritmalar yeniden çalıştırıldığında farklı sonuçlar vermektedir. Bu nedenle her bir algoritma her bir hareket için üç kez çalıştırılmış en iyi sonuçlar karşılaştırılmıştır. Bu çalışmada on dört farklı hareketin PID katsayıları elde edilmiştir. Bu on dört hareketin dokuz hareketinde BPBÇHJ Algoritması en iyi sonucu vererek etkinliğini kanıtlamıştır. HJ Algoritması'nın yerel en iyi değeri bulmadaki başarısı BPBÇ Algoritması'nın etkin arama yeteneğiyle birleştirilerek çok etkili bir arama algoritması oluşturulmuştur. Önerilen algoritma arama uzayı parametreleri değiştirilerek farklı optimizasyon problemlerinde rahatlıkla kullanılabilir.

PID katsayıları optimizasyonunda daha sonra yapılabilecek çalışmalar hakkında şunlar söylenebilir:

- Sistemde algılayıcılardan alınan açı ya da uzunluk değeri verileri filtrelenebilir. Böylece elde edilen transfer fonksiyonu sistemin daha doğru modellenmesini sağlayabilir.
- Yapılan tüm testler sistem yüksüzken yapılmıştır. Sisteme yük değerini ölçen bir algılayıcı eklenerek, sistemin üç farklı yükte PID katsayıları elde edilebilir. Arada kalan yük değerlerinde yük oranına göre PID katsayıları elde edilebilir.
- Sürekli zamanlı sistem yerine ayırık zamanlı sistem transfer fonksiyonu elde edilerek daha doğru sistem modeli oluşturulabilir.

- Sistem titreşimli çalışmıyorsa ve dayanıklıysa daha M_p 'ye bağlı olmayan amaç fonksiyonu seçilerek sistemin daha hızlı istenen amaç değerine ulaşması sağlanabilir.
- AA ile HJ Algoritması, YAKA ile HJ Algoritması veya GA ile HJ algoritması hibrit edilerek bu probleme uygulanabilir.
- BPBÇHJ algoritmasında HJ Algoritması eklenirken patern hareketi dikkate alınarak yeni bir algoritma oluşturulabilir.
- Tüm algoritmalarda yeni K_p K_d K_i değerleri oluşturulduktan sonra elde edilen sistemin kararlı olup olmadığı kontrol edilerek kararsızsa kararlı olması için rastsal K_p K_d K_i değeri atanarak algoritma daha etkili bir şekilde kullanılabilir.

KAYNAKLAR

- Akkoç, H., 2008, Robot Kontrol Sistemlerindeki Algılayıcılarda Meydana Gelen Arızaların Tespit Ve Teşhisi İçin Algoritmalar, Yüksek Lisans Tezi, *İstanbul Teknik Üniversitesi*, 85.
- Alfi, A. ve Modares, H., 2011, System identification and control using adaptive particle swarm optimization, *Applied Mathematical Modelling*, 35, 1210-1221.
- Almabrok, A., Psarakis, M. ve Dounis, A., 2018, Fast tuning of the PID controller in an HVAC system using the Big Bang-Big Crunch algorithm and FPGA technology, *Algorithms*, 11.
- Altintas, G. ve Aydin, Y., 2017, Optimization of Fractional and Integer Order PID Parameters using Big Bang Big Crunch and Genetic Algorithms for a MAGLEV System, *IFAC-PapersOnLine*, 50, 4881-4886.
- Amiri, M. S., Ramli, R. ve Ibrahim, M. F., 2019, Hybrid design of PID controller for four DoF lower limb exoskeleton, *Applied Mathematical Modelling*, 72, 17-27.
- Ang, K. H., Chong, G. ve Li, Y., 2005, PID control system analysis, design, and technology, *IEEE Transactions on Control Systems Technology*, 13, 559-576.
- Arsan, T., 2018, Büyük Patlama – Büyük Çöküş Optimizasyon Yöntemi Kullanılarak Bluetooth Tabanlı İç Mekan Konum Belirleme Sisteminin Doğruluğunun İyileştirilmesi, *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 22, 367.
- Barsainya, R., Rawat, T. K. ve Kumar, M. J. I. t., 2017, Design of minimum multiplier fractional order differentiator based on lattice wave digital filter, 66, 404-413.
- Benasla, L., Belmadani, A. ve Rahli, M., 2008, Hooke-Jeeves' method applied to a new economic dispatch problem formulation, *Journal of Information Science and Engineering*, 24, 907-917.
- Beykont, C., 2007, Bir Hidrolik Robotun Kinematik Kompanzasyonu ve Yapay Sinirsel Ağ ile İleri Beslemeli Bulanık Mantık Kontrolü, Yüksek Lisans Tezi, *İstanbul Teknik Üniversitesi*.
- Botsali, F. M., Tinkir, M., Kalyoncu, M. ve Önen, Ü., 2010, Fuzzy logic trajectory control of flexible robot manipulator with rotating prismatic joint, *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, 35-39.
- Cervantes Escorcía, N., Santos Sánchez, O. J., Rodríguez Guerrero, L., Romero Trejo, H. ve González Facundo, A., 2019, Optimal PI and PID Temperature Controls for a Dehydration Process, *Arabian Journal for Science and Engineering*, 44, 2519-2534.
- Chang, L.-Y. ve Chen, H.-C. J. W. T. S., 2009, Tuning of fractional PID controllers using adaptive genetic algorithm for active magnetic bearing system, 8 (1), 158-167.

- Chang, W.-D., 2013, Nonlinear CSTR control system design using an artificial bee colony algorithm, *Simulation Modelling Practice and Theory*, 31, 1-9.
- Chang, W. D. ve Shih, S. P., 2010, PID controller design of nonlinear systems using an improved particle swarm optimization approach, *Communications in Nonlinear Science and Numerical Simulation*, 15, 3632-3639.
- Clerc, M. ve Kennedy, J., 2002, The Particle Swarm-Explosion, Stability, and Convergence in a Multidimensional Complex Space, *IEEE Transactions on Evolutionary Computation*.
- Coşkun, İ. ve Terzioğlu, H., 2007, Hız performans eğrisi kullanılarak kazanç (PID) parametrelerinin belirlenmesi, *Journal of Technical-Online Teknik Bilimler Meslek Yüksekokulu Volume 6, Number:3-2007*, 215-226.
- Çavuşlu, M. A., Karakuzu, C. ve Şahin, S., 2010, Parçacık Sürü Optimizasyonu Algoritması ile Yapay Sinir Ağı Eğitiminin FPGA Üzerinde Donanımsal Gerçeklenmesi Hardware Implementation of Artificial Neural Network Training Using Particle Swarm Optimization on FPGA, *Politeknik Dergisi*, 83-92.
- Çengelci, B. ve Çimen, H., 2005, Teknik Not Endüstriyel Robotlar, *Makine Teknolojileri Dergisi*, 2005, 69-78.
- Eberhart, R. ve Kennedy, J., 1995, A new optimizer using particle swarm theory, *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 39-43.
- Elbayomy, K. M., Zongxia, J. ve Huaqing, Z. J. C. J. o. A., 2008, PID controller optimization by GA and its performances on the electro-hydraulic servo control system, 21 (4), 378-384.
- Erdemir, A. ve Kalyoncu, M., 2015, Bir Ağır Vasıtanın Çok Akslı Direksiyon Mekanizmasının Arı Algoritması Kullanılarak Optimizasyonu, *Uluslararası Katılımlı 17. Makina Teorisi Sempozyumu*, 17, 421-426.
- Erkol, H. O., 2017, GA ve PSO ile Kontrol Parametrelerinin Optimizasyonu, *Karaelmas Fen ve Mühendislik Dergisi*, 7, 179-185.
- Erol, O. K. ve Eksin, I., 2006, A new optimization method: Big Bang-Big Crunch, *Advances in Engineering Software*, 37, 106-111.
- Ersöz, A., 2011, Büyük Patlama Büyük Çöküş Optimizasyon Algoritması Tabanlı Bulanık Modelleme Yöntemi ve Yazılımı, Yüksek Lisans Tezi, *Istanbul Teknik Üniversitesi*, 125.
- Fister, D., Fister, I., Fister, I. ve Šafarič, R., 2016, Parameter tuning of PID controller with reactive nature-inspired algorithms, *Robotics and Autonomous Systems*, 84, 64-75.
- Gaing, Z. L., 2004, A particle swarm optimization approach for optimum design of PID controller in AVR system, *IEEE Transactions on Energy Conversion*, 19, 384-391.

- Gandomi, A. H., Yang, X.-S. ve Alavi, A. H., 2013, Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems, *Engineering with Computers*, 29, 245-245.
- Gen, M. ve Yun, Y., 2006, Soft computing approach for reliability optimization: State-of-the-art survey, *Reliability Engineering and System Safety*, 91, 1008-1026.
- Haugen, F., 2010a, Ziegler-Nichols' Closed Loop Method, *TechTeach*, 17.
- Haugen, F., 2010b, Ziegler-Nichols' Open Loop Method, *TechTeach*, 17.
- Hooke, R. ve Jeeves, T. A., 1961, "Direct Search" Solution of Numerical and Statistical Problems, *Journal of the ACM*.
- Iglesias, A., Gálvez, A., Suárez, P., Shinya, M., Yoshida, N., Otero, C., Manchado, C. ve Gomez-Jauregui, V. J. S., 2018, Cuckoo Search Algorithm with Lévy Flights for Global-Support Parametric Surface Approximation in Reverse Engineering, 10 (3), 58.
- Jaen-Cuellar, A. Y., de J. Romero-Troncoso, R., Morales-Velazquez, L. ve Osornio-Rios, R. A. J. I. J. o. A. R. S., 2013, PID-controller tuning optimization with genetic algorithms in servo systems, 10 (9), 324.
- Kalyoncu, M. ve Botsalı, F. M., 2004, Vibration analysis of an elastic robot manipulator with prismatic joint and a time-varying end mass, 29 (1), 27-38.
- Kalyoncu, M. J. A. M. M., 2008, Mathematical modelling and dynamic response of a multi-straight-line path tracing flexible robot manipulator with rotating-prismatic joint, 32 (6), 1087-1098.
- Kang, F., Li, J., Ma, Z. ve Li, H., 2011, Artificial bee colony algorithm with local search for numerical optimization, *Journal of Software*, 6, 490-497.
- Karaboga, D. ve Basturk, B., 2008, On the performance of artificial bee colony (ABC) algorithm, *Applied Soft Computing Journal*, 8, 687-697.
- Karaboga, D., 2013, an Idea Based on Honey Bee Swarm for Numerical Optimization, *Journal of Chemical Information and Modeling*.
- Karaboğa, D., 2017, Yapay Zeka Optimizasyon Algoritmaları, *nobel*, p. 232.
- Kaveh, A. ve Talatahari, S., 2009, Size optimization of space trusses using Big Bang-Big Crunch algorithm, *Computers and Structures*, 87, 1129-1140.
- Kim, D. H. J. A. s. c., 2011, Hybrid GA–BF based intelligent PID controller tuning for AVR system, 11 (1), 11-22.
- Kirgat, G. S. ve Surde, A. N., 2014, Review of Hooke and Jeeves Direct Search Solution Method Analysis Applicable To Mechanical Design Engineering, *International journal of innovations in engineering research and technology*.

- Konak, A., Coit, D. W. ve Smith, A. E., 2006, Multi-objective optimization using genetic algorithms: A tutorial, *Reliability Engineering and System Safety*, 91, 992-1007.
- Köse, U., 2017, Yapay Zekâ Tabanlı Optimizasyon Algoritmaları Geliştirilmesi, Doktora Tezi, *Selçuk Üniversitesi*, 186.
- Lasi, H., Fettke, P., Kemper, H.-G., Feld, T. ve Hoffmann, M., 2014, Industrie 4.0, *Wirtschaftsinformatik*, 56 (4), 261-264.
- Li, R., Luo, J., Sun, C.-g. ve Liu, S., 2012, Analysis of Electro-hydraulic Proportional Speed Control System on Conveyer, *Procedia Engineering*, 31, 1185-1193.
- Lin, G. ve Liu, G., 2010, Tuning PID controller using adaptive genetic algorithms, *2010 5th International Conference on Computer Science & Education*, 519-523.
- Ljung, L., 2012, Introduction to System Identification, <https://www.mathworks.com/videos/introduction-to-system-identification-81796.html>: [Erişim: 19.01.2019].
- Lu, D., Zhu, L. ve Li, X., 2010, PID parameter optimization of shunting and winch control system in coal transportation based online adaptive genetic algorithm, *2010 International Conference on E-Product E-Service and E-Entertainment*, 1-4.
- Luan, J., Yao, Z., Zhao, F. ve Song, X., 2019, A novel method to solve supplier selection problem: Hybrid algorithm of genetic algorithm and ant colony optimization, *Mathematics and Computers in Simulation*, 156, 294-309.
- MATLAB, 2019, System Identification Overview, <https://www.mathworks.com/help/ident/gs/about-system-identification.html>: [Erişim: 19.09.2019].
- Mishra, S. K. ve Swarnkar, K. K., 2014, Tuning of Controller for Electro-Hydraulic System Using Particle Swarm Optimization (PSO).
- Moser, I., 2009, Hooke-Jeeves revisited. 2009 IEEE Congress on Evolutionary Computation, IEEE: 2670-2676.
- Oi, A., Nakazawa, C., Matsui, T., Fujiwara, H., Matsumoto, K., Nishida, H., Ando, J. ve Kawaura, M., 2008, Development of PSO-based PID tuning method. 2008 International Conference on Control, Automation and Systems, ICCAS 2008.
- Önder, E., Ozdemir, M., Yıldırım, B. J. K. U. J. o. E. ve Faculty, A. S., 2013, Combinatorial Optimization Using Artificial Bee Colony Algorithm and Particle Swarm Optimization Supported Genetic Algorithm, 4 (6).
- Özsağlam, M. Y. ve Çunkaş, M., 2008, Optimizasyon Problemlerinin Çözümü için Parçacık Sürü Optimizasyonu Algoritması, *Politeknik Dergisi*, 11, 299-305.
- Panda, S., Sahu, B. K. ve Mohanty, P. K., 2012, Design and performance analysis of PID controller for an automatic voltage regulator system using simplified particle swarm optimization, *Journal of the Franklin Institute*, 349, 2609-2625.

- Pedersen, M. E. H. ve Chipperfield, A. J., 2010, Simplifying Particle Swarm Optimization, *Applied Soft Computing Journal*, 10, 618-628.
- Pham, D., Koç, E., Kalyoncu, M. ve Tinkir, M., 2008, Hierarchical PID Controller Design for a Flexible Link Robot Manipulator Using the Bees Algorithm, 25, 32.
- Rajabioun, R., 2011, Cuckoo Optimization Algorithm, *Applied Soft Computing Journal*, 11, 5508-5518.
- Reddy, P. V. P., 2015, Many Optimizing Liaisons based Assessment of PID Controller Performance for Automatic Generation Control, 4, 689-693.
- Rozali, S. M., Rahmat, M. F., Abdul Wahab, N., Ghazali, R. ve Zulfatman, 2010, PID controller design for an industrial hydraulic actuator with servo system, *Proceeding, 2010 IEEE Student Conference on Research and Development - Engineering: Innovation and Beyond, SCORED 2010*, 218-223.
- Samakwong, T. ve Assawinchaichote, W., 2016, PID Controller Design for Electro-hydraulic Servo Valve System with Genetic Algorithm. *Procedia Computer Science*.
- Sedighzadeh, M., Esmaili, M. ve Esmaeili, M., 2014, Application of the hybrid Big Bang-Big Crunch algorithm to optimal reconfiguration and distributed generation power allocation in distribution systems, *Energy*, 76, 920-930.
- Sen, M. A. ve Kalyoncu, M., 2015, Optimisation of a PID controller for an inverted pendulum using The Bees Algorithm, *Applied Mechanics and Materials*, 1039-1044.
- Shehata, R. H., Mekhamer, S. F., El-sherif, N. ve Badr, M. A. L., 2014, Particle Swarm Optimization : Developments and Application Fields, *International Journal on Power Engineering and Energy (IJPEE)*, Vol (5) - 437-449.
- Shunmugapriya, P. ve Kanmani, S., 2017, A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid), *Swarm and Evolutionary Computation*, 36, 27-36.
- Sirouspour, M. R. ve Salcudean, S. E., 2001, Nonlinear control of hydraulic robots, *IEEE Transactions on Robotics and Automation*, 17, 173-182.
- Şahin, Y., Botsalı, F. M., Kalyoncu, M., Tinkir, M., Önen, Ü., Yılmaz, N., Baykan, Ö. K. ve Çakan, A., 2014a, Force feedback control of lower extremity exoskeleton assisting of load carrying human, *Applied Mechanics and Materials*, 546-550.
- Şahin, Y., Botsalı, F. M., Kalyoncu, M., Tinkir, M., Önen, Ü., Yılmaz, N. ve Çakan, A., 2014b, Mechanical design of lower extremity exoskeleton assisting walking of load carrying human, *Applied Mechanics and Materials*, 141-145.
- Şen, M. A., Tinkir, M., Kalyoncu, M. J. J. o. L. F. N., Vibration ve Control, A., 2018, Optimisation of a PID controller for a two-floor structure under earthquake excitation based on the bees algorithm, 37 (1), 107-127.

- Tandan, N., Kuldeep Kumar Swarnkar, A. J. I. J. o. M. T. i. E. ve Research, 2015, Tuning of PID Controller using PSO and ITS Performances on Electro-Hydraulic Servo System, 2 (04), 233-235.
- Tinkir, M., Kalyoncu, M., Onen, U. ve Botsali, F. M., 2010, PID and interval type-2 fuzzy logic control of double inverted pendulum system, *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, 117-121.
- Trelea, I. C., 2003, The particle swarm optimization algorithm: Convergence analysis and parameter selection, *Information Processing Letters*.
- Vatansever, F. ve Şen, D., 2013, Genetik Algoritma Tabanlı PID Kontrolör Simülasyonu Tasarımı, *Uludağ Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, 18, 7-18.
- Yang, X.-s., 2010, Engineering Optimisation by Cuckoo Search, *Int. J. Mathematical Modelling and Numerical Optimisation*, Vol. 1, No. 4., 330-343.
- Yang, X. S., 2014, Nature-Inspired Optimization Algorithms, *Nature-Inspired Optimization Algorithms*, 1-263.
- Yang, X. S. ve Deb, S., 2014, Cuckoo search: Recent advances and applications, *Neural Computing and Applications*, 24, 169-174.
- Ye, Y., Yin, C.-B., Gong, Y. ve Zhou, J.-j., 2017, Position control of nonlinear hydraulic system using an improved PSO based PID controller, *Mechanical Systems and Signal Processing*, 83, 241-259.
- Yu, Y.-z., Ren, X.-y., Du, F.-s., Shi, J.-j. J. J. o. I. ve Steel Research, I., 2012, Application of improved PSO algorithm in hydraulic pressing system identification, 19 (9), 29-35.
- Zeybekoğlu, U., 2018, Metasezgisel Optimizasyon Yöntemlerinin Performanslarının Basit Bir Su Dağıtım Şebekesi Kullanılarak Araştırılması, *Karadeniz Fen Bilimleri Dergisi*, 7, 57-67.
- Ziegler, J. G. ve Nichols, N. B., 1942, Optimum settings for automatic controllers, *InTech*, 42, 94-100.

7. ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Hasan Basri ÖKSÜZ
Uyruğu : Türkiye Cumhuriyeti
Doğum Yeri ve Tarihi : Konya – 1993
Telefon : 0531 431 4686
Faks : -
e-mail : hasan.oksuz@gidatarim.edu.tr

EĞİTİM

Derece	Adı, İlçe, İl	Bitirme Yılı
Lise	: Konya Lisesi (Anadolu) Meram/KONYA	2012
Üniversite	: Selçuk Ü. Müh. Fak. EE. Müh. Böl. Selçuklu/KONYA	2017
Üniversite	: Selçuk Ü. Müh. Fak. Mak. Müh. Böl. Selçuklu/KONYA	2018
Yüksek Lisans	: Selçuk Ü. Fen Bilimleri Enst. EE. Müh. A.B.D. Selçuklu/KONYA	
Doktora	: -	

İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
2017-2018	MPG A.Ş.	Yazılım Tasarım Personeli
2018-	Konya Gıda ve Tarım Üniversitesi	Arş. Gör.

UZMANLIK ALANI

Optimizasyon Metodları, Mekatronik Sistemler, Yapay Zeka Optimizasyon Algoritmaları

YABANCI DİLLER

İngilizce

BELİRTMEK İSTEĞİNİZ DİĞER ÖZELLİKLER

YAYINLAR

Öksüz, H.B., Özşen, S. ve Kalyoncu, M. 2019, PID Kontrolör İle 8 Serbestlik Dereceli Hidrolik Robotun Ana Bom Hareketinin Kontrolü, *1st International Symposium on Implementations of Digital Industry and Management of Digital Transformation-ISIDIMDT'19*, 424-432.

(Yüksek Lisans tezinden yapılmıştır.)