



**PID VE KAYAN KIPLI KONTROL PARAMETRELERİNİN
GELİŞTİRİLMİŞ YAPAY ARI KOLONİSİ ALGORİTMASI İLE
OPTİMİZASYONU**

Recep KAYA

**YÜKSEK LİSANS TEZİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

**İSKENDERUN TEKNİK ÜNİVERSİTESİ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ**

EYLÜL 2020

Recep KAYA tarafından hazırlanan “PID VE KAYAN KİPLİ KONTROL PARAMETRELERİNİN GELİŞTİRİLMİŞ YAPAY ARI KOLONİSİ ALGORİTMASI İLE OPTİMİZASYONU” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile İskenderun Teknik Üniversitesi Elektrik-Elektronik Anabilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Dr. Öğr. Üyesi Murat FURAT

Elektrik-Elektronik Mühendisliği, İskenderun Teknik Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Başkan: Prof. Dr. Ahmet TEKE

Elektrik-Elektronik Mühendisliği, Çukurova Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Üye: Doç. Dr. Oğuzhan AKGÖL, İskenderun Teknik Üniversitesi

Elektrik-Elektronik Mühendisliği, İskenderun Teknik Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Tez Savunma Tarihi: 19 / 09 / 2020

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....

Prof. Dr. Tolga DEPCI

Mühendislik ve Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

İskenderun Teknik Üniversitesi Mühendislik ve Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- ☒ Tez üzerinde Yükseköğretim Kurulu tarafından hiçbir değişiklik yapılamayacağı için tezin bilgisayar ekranında görüntülendiğinde asıl nüsha ile aynı olması sorumluluğunun tarafıma ait olduğunu,
- ☒ Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- ☒ Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- ☒ Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- ☒ Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- ☒ Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Recep KAYA

19 / 09 / 2020

PID VE KAYAN KİPLİ KONTROL PARAMETRELERİNİN GELİŞTİRİLMİŞ YAPAY
ARI KOLONİSİ ALGORİTMASI İLE OPTİMİZASYONU
(Yüksek Lisans Tezi)

Recep KAYA

İSKENDERUN TEKNİK ÜNİVERSİTESİ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ

Eylül 2020

ÖZET

Kontrol, endüstrinin en önemli kavramlarından biridir. Endüstriyel platformda kalite odaklı ve uygun maliyetli hizmet üretebilmek, bu hizmetin sürdürülebilirliğini devam ettirebilmek için kontrol sistemlerine önem verilir. Üretim sistemlerini etkileyen dış bozucu ve belirsizliklerin matematiksel modellenmesinin zor olması nedeniyle, son yıllarda sezgisel algoritmalar optimum kontrol parametrelerinin ayarlanması için sıklıkla kullanılmaya başlanmıştır. Bunlar arasında Yapay Arı Kolonisi (ABC) Algoritması literatürde farklı alanlarda kendini ispatlayan bir optimizasyon algoritmasıdır.

Bu çalışmada, üzerinde endüstriyel üretim sistemlerinin vazgeçilmez kontrolcüsü PID ile bozucu ve belirsizliklere karşı dinamik sistem cevabı hızlı olmasından dolayı literatürde yoğun çalışılan kayan kipli kontrolcülerini ABC ile optimizasyonu yapılmıştır. Genel olarak, birçok çalışmada kontrolcü parametrelerinin optimizasyonu bir uygunluk fonksiyonu ile yapılırken bu tez araştırmasında ele alınan kontrolcülerin her birinin hem çıkışa olan etkisi hem de kontrol sinyali yaptığı katkı göz önünde bulundurularak farklı uygunluk fonksiyonları tanımlanmış, optimizasyonda daha iyi sonuçlara ulaşılmıştır. Literatürden seçilen bir elektromekanik sistem modeli üzerinde yapılan simülasyonlar hem grafiksel hem de istatistiksel olarak değerlendirilmiştir. Elde edilen sonuçlar, tek uygunluk fonksiyonu seçimine göre her bir parametreye özgü farklı uygunluk fonksiyonları seçiminin ABC algoritmasında en iyi değerleri bulmada daha etkin olduğunu göstermiştir.

Anahtar Kelimeler : Yapay Arı Kolonisi Algoritması, PID, Kayan Kipli Kontrol, Sürü Zekası, Uygunluk Fonksiyonu

Sayfa Adedi : 71

Danışman : Dr. Öğr. Üyesi Murat FURAT

OPTIMIZATION OF PID AND SLIDING MODE CONTROL PARAMETERS WITH
IMPROVED ARTIFICIAL BEE COLONY ALGORITHM

(M. Sc. Thesis)

Recep KAYA

ISKENDERUN TECHNICAL UNIVERSITY
ENGINEERING AND SCIENCE INSTITUTE

September 2020

ABSTRACT

Control is one of the most important concepts in the industry. Control systems are given importance in order to produce quality-oriented and cost-effective services on the industrial platform and to maintain the sustainability of this service. Due to the difficulty of mathematical modeling of external disturbances and uncertainties affecting production systems, heuristic algorithms have been used frequently to set optimum control parameters in recent years. Among these, the Artificial Bee Colony (ABC) Algorithm is an optimization algorithm that has proven itself in different areas in the literature.

In this study, PID, the indispensable controller of industrial production systems, and the sliding-mode controllers ABC, which have been studied extensively in the literature, have been optimized due to the fast dynamic system response to disturbances and uncertainties. In general, while optimization of controller parameters in many studies is done with a fitness function, different suitability functions are defined considering both the effect of each controller on output and the contribution it makes to the control signal in this thesis research, and better results are achieved in optimization. Simulations performed on an electromechanical system model selected from the literature were evaluated both graphically and statistically. The results show that choosing different fitness functions specific to each parameter is more effective in finding the best values in ABC algorithm compared to the single fitness function selection..

Key Words : Artificial Bee Colony Algorithm, PID, Sliding Mode Control, Swarm Intelligence, Fitness Function

Page Number : 71

Supervisor : Assist. Prof. Dr. Murat FURAT

TEŞEKKÜR

Bu çalışmamda, kıymetli bilgilerini benden esirgemeyen, her sorunumda bana değerli zamanını ayıran özveri ve ilgi ile bana yardımcı olabilmek için elinden geleni esirgemeyen, güler yüzü ve samimiyeti ile danışmanım olarak bana kattığı herşey için Dr. Öğr. Üyesi Murat FURAT'a teşekkürü bir borç biliyor ve şükranlarımı sunuyorum.

Teşekkürlerin yetersiz kalacağı ve beni bu günlere sevgi ve saygı ile yetiştirerek getiren ve benden asla desteklerini esirgemediği karşılıksız olarak beni seven aileme sonsuz teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET	xiv
ABSTRACT.....	xv
TEŞEKKÜR.....	xvi
İÇİNDEKİLER	xvii
ÇİZELGELERİN LİSTESİ.....	xx
ŞEKİLLERİN LİSTESİ.....	xxi
SİMGELER VE KISALTMALAR.....	xv
1. GİRİŞ.....	1
2. OPTİMİZASYON	4
2.1. Optimizasyon Algoritmaları.....	4
2.2. Optimizasyon Probleminin Temel Bileşenleri	4
2.3. Optimizasyon Problemlerinin Sınıflandırılması	4
2.4. Sezgisel Optimizasyon	5
2.5. Sürü Tabanlı Sezgisel Yöntemler.....	6
2.5.1. Karınca kolonisi optimizasyonu	7
2.5.2. Parçacık sürüsü optimizasyonu.....	8
2.5.3. Yapay balık sürüsü algoritması	9
2.5.4. Bakteriyel besin arama optimizasyon algoritması	10
2.5.5. Kurt kolonisi algoritması	10

2.5.6. Kedi sürüşü optimizasyonu	10
3. YAPAY ARI KOLONİSİ ALGORİTMASI.....	12
3.1. Gerçek Arıların Yiyecek Arama Davranışları.....	12
3.2. Başlangıç Popülasyonu	15
3.2.1. İşçi arı	15
3.2.2. Gözcü arılar.....	16
3.2.3. Kaşif Arılar	17
3.3. Seleksiyon Mekanizmaları	18
3.4. Arıların Yem Bulma Davranışları.....	18
3.5. YAKA Sözde Kodu.....	22
3.6. Yapay Arı Kolonisi Algoritmasının Temel Özellikleri.....	23
3.7. Yapay Arı Kolonisi İle İlgili Yapılan Bazı Çalışmalar	24
4. PID VE KAYAN KİPLİ KONTROLÇÜLER	29
4.1. PID Kontrolcü	29
4.1.1. Oransal kontrol (P Kontrol)	31
4.1.2. Oransal türevsel kontrol (PD kontrol)	31
4.1.3. Oransal integral kontrol (PI kontrol).....	32
4.1.4. Oransal integral türevsel kontrol (PID kontrol).....	32
4.2. Kayan Kipli Kontrol.....	34
4.3. Kayan Kipli Kontrolçülere Giriş	35
4.4. Kayan Kipli Kontrol Prensipleri	37

4.4.1. Birinci derece kayan kipli kontrol.....	40
4.4.2. Yüksek dereceden kayan kipli kontrol	42
4.4.3. İkinci dereceden kayan kipli kontrol	43
4.4.4. Burulma Algoritması	43
4.4.5. Süper burulma algoritması (Super twisting algorithm).....	44
4.5. Çeşitli Kayan Kipli Kontrol Uygulamaları	45
5. MATERYAL VE YÖNTEM	48
5.1. Materyal.....	48
5.2. Yöntem.....	48
5.2.1. PID optimizasyonu	48
5.2.2. KKK Optimizasyonu.....	55
6. SONUÇ	64
KAYNAKLAR	65
ÖZGEÇMİŞ	70
DİZİN.....	71

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 4.1. Parametrelerin sistem üzerindeki etkileri.....	33
Çizelge 5.1. PID parametrelerin limit değerleri	49
Çizelge 5.2. PID optimizasyonda kullanılan uygunluk fonksiyonları	49
Çizelge 5.3. PID simülasyon sonuçları	50
Çizelge 5.4. KKK parametre limit değerleri	57
Çizelge 5.5. KKK algoritmasında kullanılan uygunluk fonksiyonları.....	58
Çizelge 5.6. KKK simülasyon sonuçları	58

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Sezgisel yöntemler [10]	8
Şekil 3.1. Yapay Arı Kolonisi Algoritması akış diyagramı	14
Şekil 3.2. Kaynağın Kovana Olan Mesafesine Göre Arıların Dans Çeşitleri.....	19
Şekil 3.3. Arıların yem arama davranışları [30]	20
Şekil 4.1. PID Denetleyicinin blok diyagramı	29
Şekil 4.2. Zaman düzlemi cevabı parametreleri.....	30
Şekil 4.3. Kontrolcü türüne göre sistem cevapları.....	34
Şekil 4.4. Farklı başlangıç koşullarından başlayarak σ değişkeninin tipik evrimi	40
Şekil 4.5. Farklı Kontrol sinyalinin tipik evrimi (kesikli çizgi σ 'yi temsil eder.).....	41
Şekil 4.6. Kayma kipi kontrolünün yumuşak yaklaşımları [52]	42
Şekil 4.7. Burulma algoritması faz düzlemi yörüngeleri	44
Şekil 4.8. Süper burulma algoritması faz düzlemi yörüngeleri	45
Şekil 4.9. PI (solda) ve Süper Bükümlü (sağda) kontrol cihazlarının blok düzeni.....	45
Şekil 5.1. Önerilen 3-kanal uygunluk fonksiyonu tabanlı YAKA.....	50
Şekil 5.2. Arıların besin arama bölgesi	51
Şekil 5.3. Geleneksel YAKA'nın PID Uygunluk Grafiği.....	52
Şekil 5.4. Geleneksel YAKA'nın PID Sonuç Grafiği.....	52
Şekil 5.5. Önerilen YAKA'nın P Parametresi Uygunluk Grafiği.....	53
Şekil 5.6. Önerilen YAKA'nın I Parametresi Uygunluk Grafiği.....	53
Şekil 5.7. Önerilen YAKA'nın D Parametresi Uygunluk Grafiği	54
Şekil 5.8 Önerilen YAKA'nın PID sonuç grafiği.....	54
Şekil 5.9 Geleneksel YAKA ile KKK uygunluk grafiği.....	59

Şekil	Sayfa
Şekil 5.10 Geleneksel YAKA ile KKK sonuç grafiği	59
Şekil 5.11 Geleneksel YAKA ile KKK kontrol sinyali grafiği	60
Şekil 5.12 Geleneksel YAKA ile KKK referans izleme performans grafiği	60
Şekil 5.13 Önerilen YAKA ile lambda parametresi uygunluk grafiği.....	61
Şekil 5.14 Önerilen YAKA ile Ka parametresi uygunluk grafiği	61
Şekil 5.15 Önerilen YAKA ile KKK çıkış grafiği.....	62
Şekil 5.16 Önerilen YAKA ile KKK kontrol sinyali grafiği	62
Şekil 5.17 Önerilen YAKA ile KKK referans izleme performans grafiği.....	63

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler

Açıklamalar

D	Parametre Sayısı
SN	Görevli Arı Sayısı
p_i	Kaynağın Kalitesi
K_p	Oransal Kazanç
K_i	İntegral Kazanç
K_d	Türevsel Kazanç
σ	Kayma Yüzeyi
λ	Bağımsız Parametre
β	Adaptif Parametre

Kısaltmalar

Açıklamalar

GA_r	Gözcü Arı
$İA_r$	İşçi Arı
ISE	Hatanın Karesinin İntegrali
ITAE	Hatanın Mutlak Değerinin Zaman İntegrali
KA_r	Kaşif Arı
KKK	Kayan Kipli Kontrolcü
PID	Oransal İntegral Türev
YAKA	Yapay Arı Kolonisi Algoritması
SA	Sezgisel Algoritmalar

1. GİRİŞ

Günümüzde optimizasyon işlemleri, mühendislik tasarımlarında kullanılan önemli bir yöntemdir. Optimizasyon yöntemlerini kullanmanın önemli avantajları olduğu bilinmektedir. Reel sistem dizaynı için, geleneksel metodlar deneysel formülleri kullanarak yalnızca yapılabilirliğini esas alan çözümler içerirken, optimizasyon yöntemleri gerçekçi fiziksel modellere dayalı optimum sonuçlar arar. Mühendislik tasarımında kullanılan optimizasyon yöntemleri çok sayıda işlem yapmayı gerektirir. Çünkü, optimizasyon yöntemlerinin çözmeye çalıştıkları kompleks fonksiyonların farklı potansiyel çözümler için incelenmesi gerekmektedir [1].

Endüstride yoğun olarak tercih edilen PID ile buna alternatif olarak önerilen Kayan Kipli Kontrol (KKK) algoritmalarında, parametrelerin ayarlanması konusunda literatürde birçok çalışma yapılmaktadır. Literatürü incelediğimizde kabul edilmiş PID için çeşitli parametre ayarlama metodları mevcuttur [2]. Bu metodlar, uygulanan sistemde iyi bir çıkış vermesiyle beraber bazı durumlarda aşım, kalıcı durum hatası, düşük yerleşme zamanı gibi olumsuz özelliklere sahip çıkışa da sebep olabilmektedir. Bu sorunlar, PID parametrelerinin uygun seçimiyle aşılabılır niteliktedir. Literatürde hayli fazla yer bulan ilk KKK yöntemlerinde anahtarlama kontrol sinyalinin oluşturduğu genliği çok olan ve büyük frekanslı dalgalanma, bu methodun en kritik istenmeyen tarafı olarak bilinir. Çatırdama olarak isimlendirilen kontrolcü sinyalindeki bu osilasyonu oluşturan başlıca neden, genellikle gerçek sistemlerde bulunan ve matematiksel olarak modellenemeyen belirsizliklerdir [3]. Çatırdama, dinamik sistem cevabı hızlı olan sistemlere uygulandığında, özellikle hareket eden mekanik parçalara zarar verebilir [4]. Bu çalışmanın amaçlarından biri Yapay Arı Kolonisi Algoritması (YAKA) kullanılarak bu problemi en aza indirecek kontrol parametrelerini aramaktır. Aynı PID'e olduğu gibi gerek KKK algoritmasında yapılacak iyileştirme ile gerekse doğru genlikteki parametre seçimi ile bu problemin üstesinden gelinebilir [5]. Nitekim, istenilen karakteristikteki çıkışı elde etmek için parametre değer seçiminin önemi ortaya çıkmıştır.

Sürü zekâsına dayalı YAKA, her geçen gün birçok çalışmada yaygınlaşan algoritma türünden biridir [6]. 2005'de, Derviş Karaboğa tarafından öne sürülen YAKA, son yıllarda oldukça fazla kullanılmaya başlanmıştır. Kullanılan uygulama alanlarının birçoğunda da etkili sonuçlar vermiştir. Doğal ortamdaki arıların besin arama davranışlarından esinlenip geliştirilmiş olan bu metod, en iyi çözüme hızlı bir şekilde yaklaşması ile tüm dikkatleri

üzerine çekmiştir [6].

Bu tez çalışmasında, YAKA'da kullanılan uygunluk fonksiyonundaki parametrelerin ağırlıklarının sistem çıkışında ki performansına olan etkisi araştırılmış ve uygun ağırlık parametrelerinin belirlenmesi hedeflenmektedir. YAKA sanayi ve hidrolik mühendisliği alanlarından havacılık ve uzay bilimi gibi farklı bilim dallarına kadar başarılı bir şekilde uygulanmıştır [7].

En iyi kontrol parametrelerinin aranmasında YAKA literatürde de kullanılmasına rağmen birden fazla kontrolcü parametresinin en iyi değerleri için tek bir uygunluk fonksiyonu kullanılmış, bu parametrelerinin kontrol sinyali ve sistem çıkış üzerindeki farklı etkileri bazı çalışmalarda bir araya getirilmiştir. Bu tez çalışmasında ise, bunların aksine, her bir kontrol parametresinin gösterdiği karakteristik düşünülerek hem PID hem de KKK için farklı uygunluk fonksiyonları tanımlanmış ve aynı döngü içinde paralel olarak arama yapılmıştır. Böylece, daha iyi sonuçlara varılacağı hedeflenmiştir. Bu sonuçlar arasındaki hedefler kısaca, en kısa sürede optimizasyonun tamamlanması ve arama uzayı içinde küresel en iyi parametrelerin bulunması olarak sayılabilir.

YAKA gibi dögüsel arama yaparak en iyi değerleri bulan algoritmalar, gerçek sistemler için uygulanması zor olan sayılarda işlem ve zaman gerektirebilir. O yüzden genel yaklaşım, problemi ya da sistemi tanımlayan bir matematiksel model üzerinde bu algoritmaların çalıştırılması ve elde edilen sonuçların gerçek sistemlere uygulanması şeklindedir.

Bu konuda, günümüz teknolojisi ile dijitalleşen sistemlerin optimizasyonu için sistemlerin modelleri üzerinde çalışılmakta ve optimum değerlere daha hızlı bu modeller üzerinden karar verilmektedir. Geleneksel analog sistemlerin yerini dijital sistemlerin aldığı, Nesnelerin İnterneti (IoT) ile birbirleri haberleşmelerinin sağlandığı Endüstri 4.0 yapısı altında, matematiksel sistem modelleri üzerinde çalıştırılan optimizasyon algoritmalarının verimliliği daha da önem kazanmaktadır. Bu açıdan düşünüldüğünde, endüstride ve literatürde kendini ispatlamış olan PID ve KKK algoritmalarının matematiksel sistem modeli üzerinde optimizasyonu bir gereklilik halini almaya başlamıştır. Bununla beraber, YAKA, önemli derecede kabul gören bir algoritma olduğu için bu tez çalışmasında kontrolcülerin optimizasyonu için seçilmiştir. Bu çalışmanın özgünlüğü, YAKA ile kontrolcü parametrelerinin özelliklerine göre farkı uygunluk fonsiyonları seçilerek yapılan

optimizasyon olmasıdır.



2. OPTİMİZASYON

2.1. Optimizasyon Algoritmaları

Mühendislik alanında yapılan bilimsel araştırmaları genellikle iki ana gruba ayırırız:

- Model (Tasarım)
- Çözümleme (Analiz)

Modellenen bir sistemde beklentilerimiz fazla sayıda olası benzer çözümler içinde en optimum olanı en kısa zamanda bulması bir optimizasyon problemi olup önemli bir mühendislik çalışmasıdır [8]. Son yıllarda optimizasyon ile ilgili çalışmalarda bilim insanları genel amaçlarına uygun ve performansı yüksek yöntemler geliştirebilmek için doğa sistemlerine ve bu sistemlere doğrudan etki eden olan olaylara doğru yönelmişlerdir. Doğa ortamındaki sistemleri ve olayları temel alıp oluşturulmuş öteki en iyileme yöntemleri da sezgisel yöntemler olarak isimlendirilmişlerdir. Bu yöntemler başka mühendislik alanlarında birçok kompleks sorunların çözümünde kullanılmaya başlanmıştır.

2.2. Optimizasyon Probleminin Temel Bileşenleri

Her optimizasyon problemi aşağıdaki temel bileşenleri içermek zorundadır.

Uygunluk Fonksiyon: Minimize veya maksimize edilen niceliği gösterir. Uygunluk fonksiyon tüm kısıtları sağlayacak şekilde bir çözüm bulmalıdır.

Değişkenler veya bilinmeyenler kümesi: Bu kümedeki elemanlar uygunluk fonksiyonu etkileyen değerlerden oluşur. x bilinmeyenler kümesi olmak üzere $f(x)$, x aday çözümünün kalitesini belirtir.

Kısıtlar Kümesi: x bilinmeyenler kümesinin alabileceği değerleri kısıtlayan kümedir. Problemlerin çoğu her değişken için belirli bir sınır kısıtlayıcısına sahiptir.

Herhangi bir optimizasyon problemi yukarıdaki temel bileşenlere sahip olmalıdır [9].

2.3. Optimizasyon Problemlerinin Sınıflandırılması

Optimizasyon problemleri karakteristik özelliklerine göre aşağıdaki gibi sınıflandırılabilir [9]:

Değişkenlerin Sayısı: Uygunluk fonksiyonu etkileyen değişkenlerin sayısı bir ise tek parametrelili birden fazlaysa çok parametrelili optimizasyon problemi adını alır.

Değişkenleri Türü: Sürekli bir problem sürekli değere sahip değişkenlerden oluşur; $j=1 \dots n$ 'e kadar her $x_j \in \mathbb{R}$ ise bu probleme sürekli optimizasyon problemi denir. Eğer $x \in \mathbb{Z}$ ise o zaman bu problem tamsayı değerli veya ayrık optimizasyon problemi adını alır. Karışık tam sayı problemi hem sürekli hem de tam sayı değerli değişkenlere sahiptir. Tamsayı değerli değişkenlerin permütasyonuna dayalı çözümlere sahip problemler de kombinyasyonel (combinatorial) optimizasyon problemi adını alır.

Lineerlik (Doğrusallık): Lineer problemlerin amaç fonksiyonları lineer değişkenlere sahiptir. Kuadratik problemler kuadratik fonksiyonlar kullanır. Diğer non-lineer (doğrusal olmayan) amaç fonksiyonları kullanıldığında problem nonlinear optimizasyon problemi halini alır.

Sınırlar: Eğer bir optimizasyon problemi sadece kenar kısıtları içeriyorsa sınırsız optimizasyon problemi, kenar kısıtlarına ek olarak eşitlik veya eşitsizlik kısıtları da içeriyorsa sınırsız optimizasyon problemi olarak tanımlanır.

2.4. Sezgisel Optimizasyon

Sezgisel algoritmalar (SA), belirli bir hedefi yerine getirmek veya amaca ulaşmak için doğal olaylardan ilham alan algoritmalar. Bu SA'nın, çözüm kümesinde optimal çözüme yakınsaması kanıtlanamamaktadır. Yani SA yakınsama özelliğine bir anlamda sahiptirler, ama mutlak çözümü garanti edememektedirler ve bu durum mutlak çözümün civarında bir çözüm garanti edebilmektedir. Anlaşılabilirlik yönünden SA'nın karar verici açısından çok daha basit olabilmesinden, en iyileme problemlerinin mutlak çözümü bulma sürecinin tanımlanamadığı bir yapıda olmasından ve öğrenme hedefli ve mutlak çözümü bulma sürecinin bir parçası olarak kullanılabilirliğinden SA'a gereksinim duyulmaktadır. Genel amaçlı sezgisel yöntemler; biyoloji, fizik, sürü, sosyal, müzik ve kimya tabanlı olmak üzere altı çeşit grupta değerlendirilmektedirler. Aynı zamanda bu yöntemlerin karışımı olan melez yöntemler de bulunmaktadır. Bahsi geçen bu yöntemler Şekil 2.1'de gösterilmektedir.

Genetik algoritma (GA), Diferansiyel Gelişim Algoritması (DGA), Karınca Koloni Algoritması (KKA), Yapay Sinir Ağları (YSA), Arı Kolonisi Algoritması (AKA) ve Yapay Bağışıklık Sistemleri (YBS) biyoloji tabanlı; Emperyalist Yarışmacı Algoritması (EYA), Parlamenter Optimizasyon Algoritması (POA) ve Tabu Arama Algoritması (TAA) sosyal tabanlı; Yapay Kimyasal Reaksiyon Algoritması (YKRA) kimya tabanlı; Armoni Arama Algoritması (AAA) müzik tabanlı; Isıl İşlem Algoritması (İİA), Büyük Patlama Algoritması (BPA), Gravitasyonel Arama Algoritması (GAA), Merkez Kuvvet Optimizasyonu (MKO), Zeki Su Damlacıkları Algoritması (ZSDA) ve Elektromanyetizma Algoritması (EA) fizik tabanlı; ve Parçacık Sürü Optimizasyonu (PSO), Kedi Sürü Optimizasyonu (KSO) sürü tabanlı algoritma ve modellerdir. Kültürel Algoritma (KA) da hem biyolojik hem de sosyal tabanlı algoritma olarak kabul edilir [10].

2.5. Sürü Tabanlı Sezgisel Yöntemler

Zaman zaman tek başlarına hiçbir varlık gösteremeyen canlılar, beraber topluca eylem yaptıklarında olağanüstü davranışlar gösterebilmektedirler. Herhangi bir sürünün bireyleri, en başarılı bireyin tavrından veya öteki bireylerin tavırlarından ve bireysel tecrübelerinden yararlanır analiz eder ve bu bilgi ve birikimleri ilerleyen süreçte maruz kalabileceği sorunların çözümleri için bir araç olarak kullanırlar. Mesela, bir canlı topluluğunu meydana getiren bireylerden birisi bir tehlike ile karşılaşır bu tehlikeye karşı tepki verir ve bu tepki canlı toplulukları içinde zamanla genişleyip tüm sürü üyelerinin tehlikeye karşı ortak bir tavır göstermesini sağlar.

Bireylerin topluluklar şeklinde bir sorunu bireysel olarak çözemeyip beraber gösterdikleri gayrete ve beraber çözme yeteneğine sürü zekası denilmektedir.

Bu zamana kadar çeşitli bilim insanlarının öne sürdüğü Sürü Zekası Optimizasyon Algoritmaları (SZOA) başlıklar halinde verilmiştir:

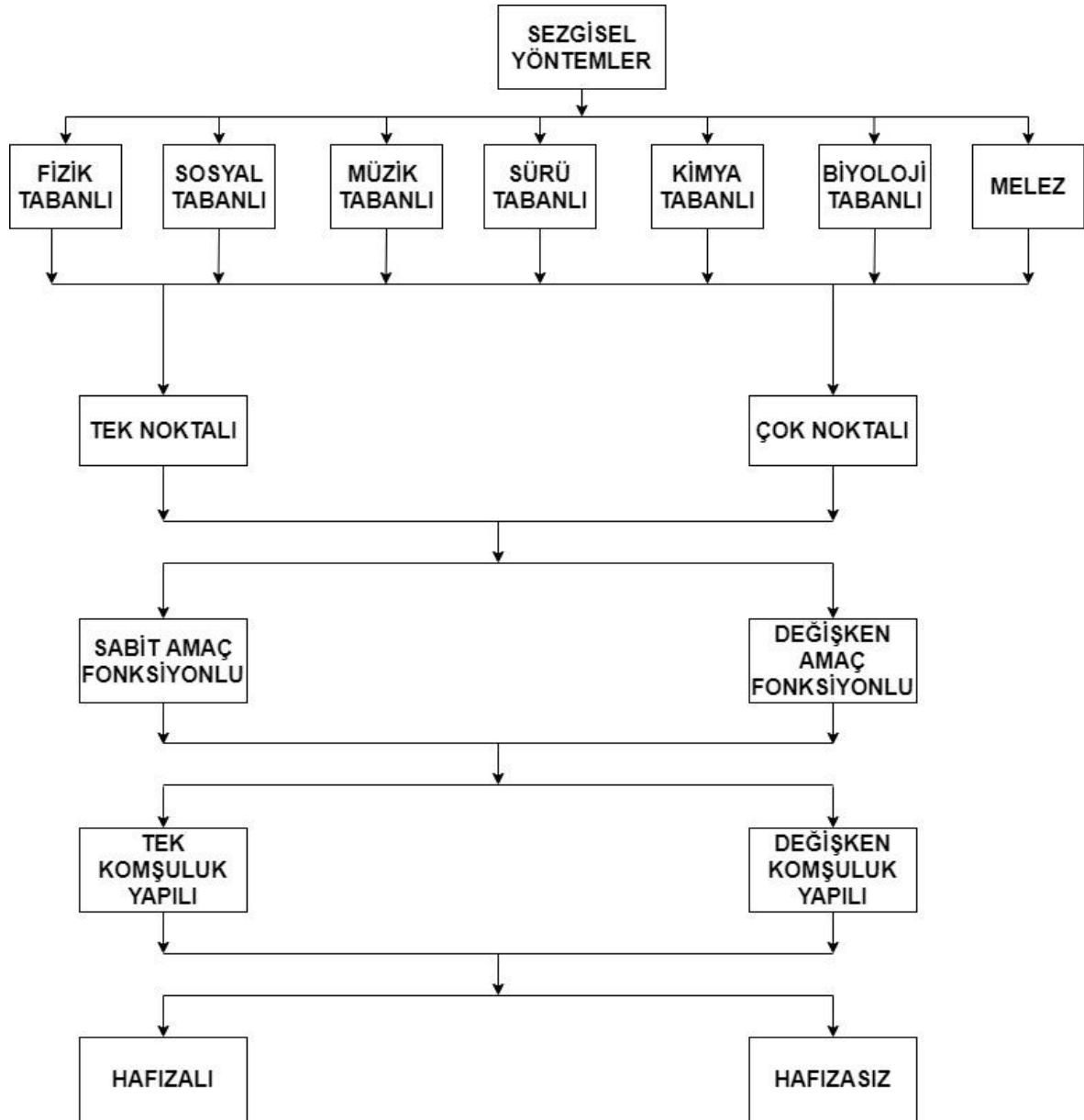
- Karınca Koloni Optimizasyonu (KKO)
- Parçacık Sürü Optimizasyonu (PSO)
- Yapay Balık Sürüsü Algoritması (YBSA)

- Bakteriyel Besin Arama Optimizasyon Algoritması (BBAOA)
- Kurt Kolonisi Algoritması (KKA)
- Kedi Sürüşü Optimizasyonu (KSO)
- Yapay Arı Koloni Algoritması (YAKA)

Sürü zekasına dayalı olan Yapay Arı Kolonisi Algoritması (YAKA) bu çalışmada detaylı olarak incelenecektir. Ardından, bu algoritmanın besin arama fonksiyonları yardımı ile Kayan Kipli Kontrolcünün parametreleri en iyileme yöntemi ile bulunması sağlanacaktır.

2.5.1. Karınca kolonisi optimizasyonu

Karıncalar koloni şeklinde yaşayan ve en kısa yol problemini çözmeden oldukça başarılı olan hayvanlardır. Görme yeteneklerinin kısıtlı olmasına rağmen yuva ile yiyecek kaynağı arasındaki ve tersine besin kaynağı ile koloninin yuvası arasındaki en kestirme yolu yüzyıllardır bulabilmektedirler. Ayrıca yuva ile kaynak arasındaki çevre değişkenlerine bağlı olası yol bozulmalarında (engeller, su vb.) aralarındaki etkileşimle yine en kısa yolu bulabilmektedirler. Kısacası karınca kolonileri aralarında belirli bir etkileşim ve görev paylaşımı olan ve kolektif davranan sürülerdir. Karıncaların bu davranışlarını örnek alarak oluşturulan karınca kolonisi algoritması Dorigo ve arkadaşları tarafından 1991 yılında geliştirilmiştir. Fazla sayıda etkileşim içerisinde olan basit bireylerin tavırlarının simülasyonundan esinlenen karınca algoritması, kolay olmayan sorunların dağıtılmış çözümüne bir yaklaşım olarak tanıtılmıştır [8].



Şekil 2.1. Sezgisel yöntemler [10]

2.5.2. Parçacık sürüsü optimizasyonu

Sezgisel yöntemlerden biri olan (Parçacık sürüsü optimizasyonu) PSO yöntemi ilk olarak kuş ve balık sürülerinin davranışlarından etkilenecek doğrusal olmayan nümerik problemlere uygun sonuçlar bulmak için 1995–1996 yıllarında sosyolog-psikolog James Kennedy ve elektrik mühendisi Russel Eberhart iş birliğinde ortaya çıkmıştır. PSO sürü tabanlı olasılıksal bir en iyileme metodu olup çok parametrelili ve çok değişkenli en iyileme sorunlarına çözümler üretebilmek için kullanılmaktadırlar.

Parçacık sürü kavramsal olarak basitleştirilmiş sosyal bir sistemin bir benzetimi olarak meydana gelmiştir. İlk başlarda hedef, kuş ya da balık sürü figürlerinin grafiksel olarak benzetimlerini yapmaktı. Fakat grafiksel benzetimden sonra, parçacık sürü prototipinin bir en iyileme metodu olarak kullanılabileceği bulunmuştur. Kuş sürüleri gerçek besin kaynağını bilmemelerinden ötürü, besin kaynağından ne kadar uzakta olduklarını öğrenmeye uğraşırlar. Öğrenmek için izlenen metod besin kaynağına en yakın olan kuşu gözlemlemektir.

PSO'da bütün kuşlar ayrı ayrı parçacık olarak ifade edilir, kuşların tamamı da sürü olarak ifade edilirler. Parçacık yer değiştirdiğinde, kendi konumlarının uygunluk değeri yani besine ne kadar uzaklıkta olduğunu hesaplar. Bir parçacık, konumunu, hızını yani çözüm kümesindeki her evrede ne kadar hızla ilerlediği bilgisini, bu zamana kadar elde ettiği en iyi uygunluk değerini ve bu değeri elde eder iken bulunduğu konumunu hatırlamalıdır. Çözüm kümesinde bütün boyuttaki hızının ve yönünün her defasında nasıl değişeceği, komşularının optimum konumlarını ve kendi bireysel optimum konumlarının birleşiminden elde edilecektir [10].

2.5.3. Yapay balık sürüsü algoritması

Yeni bir evrimsel aritmetik yöntemi olan Yapay Balık Sürüsü Algoritması (YBSA) ilk defa 2002'de ortaya atılmıştır. YBSA'nın öne sürdüğü sürü kavramına ve basitleştirilmiş balık topluluğunun doğal toplumsal tavırlarının benzetimine dayanmaktadır.

YBSA'da genetik algoritmada olduğu gibi evrim süreçleri ve çaprazlama teknikleri bulunmamaktadır. Bundan dolayı daha basit bir şekilde işlemektedir. Sistem ilk önce rastgele üretilen bir dizi potansiyel çözümler ile başlatılır ve daha sonra kendine tekrar eden bir şekilde en iyi bir çözüm elde edilene kadar arama yapar.

Hakiki bir balığın soyutlanmış bir şekli olan Yapay Balık (YB), problemin irdelenmesinde, analizinin gerçekleştirilmesinde ve hayvanlar için ekoloji anlamının görülebilmesinde kullanılmaktadır. Kuyruk ve yüzgeçlerinin kontrol edilmesi ile uyarıcı bir tepki görülmektedir. Daha sonraki tavır ise onun çevresinin durumuna ve günümüz durumuna bağlıdır. Bu ekolojik ortam, onun bireysel hareketlerini ve diğer eşlik eden bireylerin

hareketlerini de etkilemektedir [11].

2.5.4. Bakteriyel besin arama optimizasyon algoritması

Escherichia Coli (E. Coli) bakterisi, dünya üzerinde ki hemen her canlının bağırsaklarında yaşayan bir bakteri çeşididir. E. Coli bakterilerinin yiyecek bulma sürecindeki tavırlarına şekil veren bir kontrol mekanizması bulunmaktadır. Yiyecek bulma süreci, yiyecek kaynakları yönünde gerçekleştirilen bir seri aktiviteden meydana gelmektedir. Kontrol mekanizması, E. Coli bakterisinin bir sonraki konumunu değiştirme aktivitesine referans olacak şekilde bir yerden diğer yerlere giderken ortaya çıkan farklılıkların değerlendirilmesi kuralına uymaktadır. İki ayrı konum arasındaki öteleme bir “hareket” olarak isimlendirilir ve her bir hareket belirli bir yöne doğru ve ayrı bir adım mesafesi vardır. E. Coli bakterileri, kontrol mekanizmalarına göre yiyecek kaynaklarına adım adım hareket ederler [12].

2.5.5. Kurt kolonisi algoritması

Kurt kolonisi algoritması kurt sürülerinin kuvvetli bir teşkilat sistemine sahip olmasından yola çıkılarak geliştirilmiştir. Sürü içinde ki kurtlar görev dağılımı yaparlar ve avlanacakları zaman emin adımlar atmaktadırlar. Av sırasında bir takım kurtlar arama kurtları olarak tayin edilmekte ve avlarını gördükleri zaman avın konumunu diğer kurtlara ulayarak bildirmektedirler.

Diğer kurtlar ava yaklaşmakta ve avın etrafını kuşatmaktadır. Kurt kolonisinin tayin kuralı, besinin başlangıçta en güçlü kurda ve daha sonra adım adım diğer zayıf olan kurtlara verilmesidir [13].

2.5.6. Kedi sürüsü optimizasyonu

Kedi Sürüsü Optimizasyonu (KSO) algoritmasında arama kipi ve izleme kipi adı altında iki alt bileşeni bulunmaktadır. Bu iki kipi algoritma cinsinde bir araya getirebilmek için, arama kipi ile izleme kipini birbirine birleştirmeye yarayan bir karışım oranı (KO) ifade edilmektedir. Kediler dinlenme süresince aktivite etmeye karar aldıklarında usulca yerine getirirler. Bu aktivite, arama kipine de yansıtılır. İzleme kipinde ise kedi tarafından bir amacın takip edilmesi modellenmiştir. Kediler, enerji kaynaklarını aşırı kullanmalarına yol açan nesneleri takip etmeye minimum zaman harcamaktadırlar. Kediler zamanlarının

birçoğunda dinlenmeye ve gözlemlemeye (mesela zamanlarının çoğu arama kipinde harcamaktadırlar) ayırdıklarını garantilemek için KO'ya çok küçük bir değer tayin edilmektedir [14-16].



3. YAPAY ARI KOLONİSİ ALGORİTMASI

Bu bölümde YAKA için optimizasyon teknikleri incelenip algoritma geliştirme teknikleri açıklanacaktır. Daha sonra YAKA, kayan kipli kontrol optimizasyonunda kullanımı tanıtılacaktır.

YAKA doğal ortamda bulunan arıların hareketleri gözlemlenip geliştirilmiş bir en iyileme algoritmasıdır. Bu algoritma ilk kez Derviş Karaboğa tarafından 2005 yılında öne sürülmüştür [17]. Bir arı kovanında üç tip arı vardır. Bunlar işçi arı (İArı), gözcü arı (GArı) ve kâşif arıdır (KArı). Bilirli bir kaynaktan besin taşıyan arıya İArı denir. Arılar besin kaynağı bulurlarsa bunu kovanda ki diğer arılara dans ederek söylerler. Dans bölgesinde bekleyip, alınan bilgilere göre besin kaynağını seçen arılara GArı denir. Rastgele gezerek besin arayan arıya ise KArı adı verilir. Koloninin yarısı İArı, yarısı GArı dan meydana gelir. Bütün besin kaynaklarında sadece bir İArı vardır. Bu yüzden de kolonideki İArı veya GArı sayısı çözüm sayısına denktir. Besin kaynağı biten her arı KArı'ya dönüşür ve yeni bir besin kaynağı arar. Besin kaynağının konumu optimize edilen problem için çözümlerden sadece birini temsil etmektedir. Kaynaktaki besin miktarı ise çözümün uygunluğu için bir seçim kriteridir [18]. Bu yüzden YAKA en fazla besine sahip olan kaynağın yerini bulmak için çalışır ve çözüm kümesinde ki çözümlerden problemin minimumunu yada maksimumunu veren koordinatı bulmaya çalışır.

3.1. Gerçek Arıların Yiyecek Arama Davranışları

Doğada bir arı kovanında yapılacak görevler o görev için belirlenmiş arılar tarafından yapılır. Yani, yapılması gereken işler için arılar arasında bir görev dağılımı vardır ve herhangi bir merkezi otorite de olmadığı için bu görev dağılımını gerçekleştirebildikleri için kendi kendilerine organize olabilmektedirler. Görev dağılımı yapabilme ve kendi kendine organize olabilme sürü zekasının en önemli iki özelliğidir. Tereshko'nun reaktif difüzyon denklemlerine bağlı olarak öne sürdüğü kolektif zekanın ortaya çıkmasını gerçekleyen minimal besin arama modelinde temel üç alt grup bulunmaktadır: Besin kaynağı, Görevli-İşçi arılar ile Görevsiz-İşçi arılar. Bu minimal model iki kipte çalışır [19].

Bu minimal modele ait gerçekleşme adımları ise aşağıdaki gibi verilebilirler:

1. Besin Arama sürecinin başında, KArı'lar ekolojik ortamda rastgele arama yaparak besin

bulmaya başlarlar.

2. Besin kaynakları KArı'lar tarafından bulunduktan sonra, KArı artık görevli arı olur ve buldukları kaynaklardan kovana besin getirmeye başlarlar. Her bir görevli arı kovana dönüp getirdiği besini bırakır ve bundan sonra ya bulduğu besin kaynağına geri gider ya da kaynakla ilgili konum bilgisini dans bölgesinde gösterdiği dans ile kovanda bekleyen GArı'lara verir. Eğer fayda sağlanacak olan kaynak bitmiş ise görevli KArı'ya dönüşür ve yeni besin kaynakları keşfetmeye başlar.

3. Arı kovanında bekleyen GArı'lar zengin besin kaynaklarını gösteren dansları izler ve besinin kalitesi ile orantılı olan dansın frekansına göre bir besin kaynağını seçerler.

YAKA da bu süreçler ve temel adımlar ise aşağıdaki sıralanır;

1. Başlangıç besin kaynağı alanlarının üretilmesi

Repeat

2. İArı'ların besin kaynağı alanlarına gitmesi

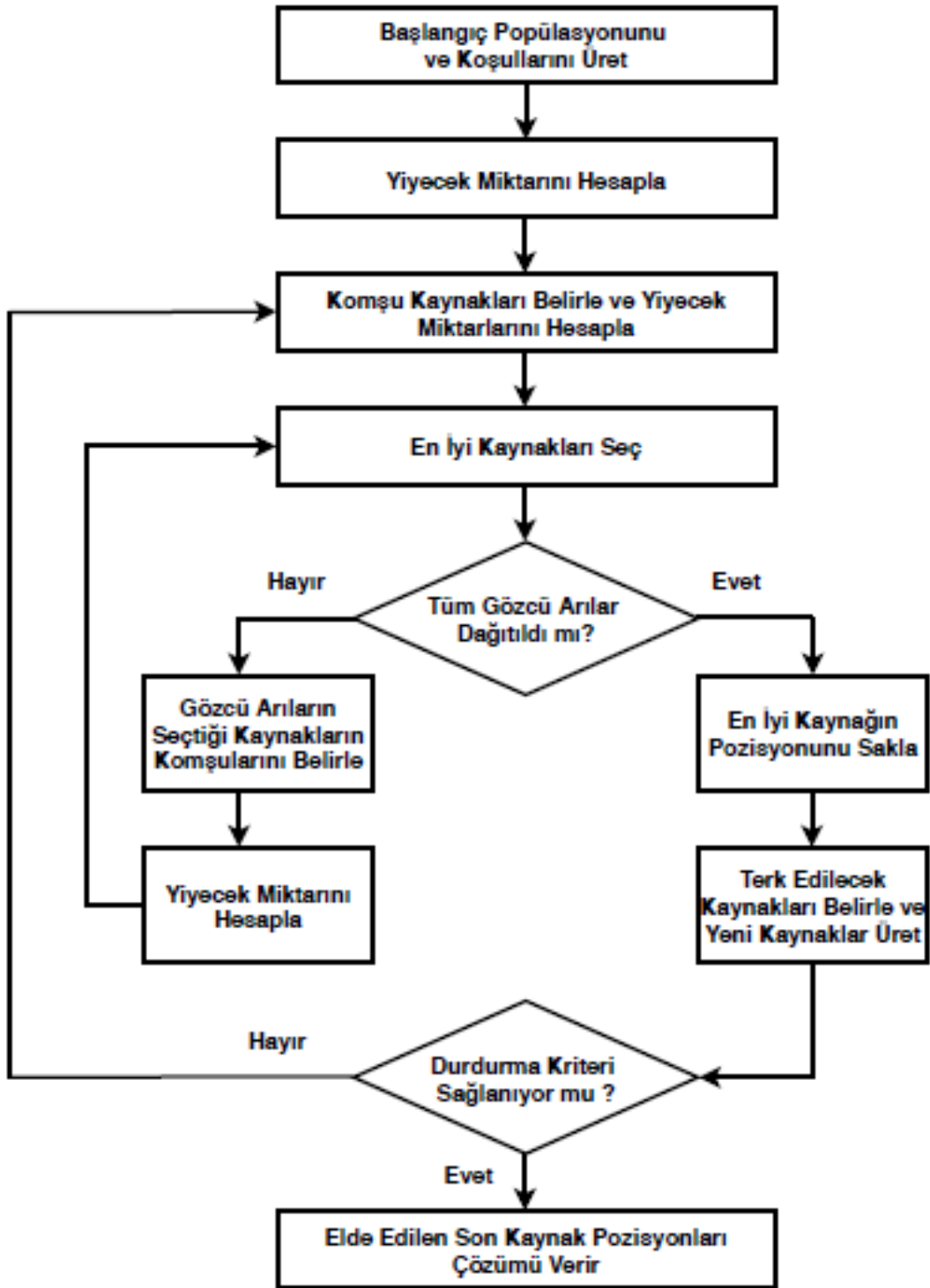
3. Muhtemel seçimde kullanılacak olan olasılık değerlerinin görevli arılardan gelen bilgi ile hesaplanması

4. GArı'ların muhtemel değerlerine göre besin kaynağı alanlarını seçmesi

5. Durdurma kriteri: limit ve KArı üretimi

Until (iterasyon sayısı=Maksimum iterasyon sayısı)

Algoritmanın genel yapısı Şekil 3.1'de verilmiştir.



Şekil 3.1. Yapay Arı Kolonisi Algoritması akış diyagramı

3.2. Başlangıç Popülasyonu

Arama kümesini besin kaynaklarını içeren kovan etrafı olarak düşünülürse algoritma arama kümesindeki çözümlere karşılık gelen rastgele besin kaynağı bölgeleri üretmek çalışmaya başlamaktadır. Rastgele bölge üretme süreci her bir değişkenin alt ve üst sınırları içerisinde rastgele değer üretmek gerçekleştirilir (Eş.3.1):

$$x_{ij} = x_{minj} + rand(0,1) * (x_{maxj} - x_{minj}) \quad (3.1)$$

Burada $i = 1 \dots SN$, $j = 1 \dots D$ ve SN yiyecek kaynağı sayısı ve D is optimize edilecek değişken sayısıdır. x_{minj} parametrelerin alt sınırıdır [19].

3.2.1. İşçi arı

Daha önceden de belirtildiği üzere her bir kaynağın bir görevli arısı bulunmaktadır. Bu nedenle besin kaynakların sayısı görevli arıların sayısına denktir. İArı çalıştığı besin kaynağının komşuluğu olan bölgede yeni bir besin kaynağı belirler ve bunun nektarını değerlendirir. Yeni kaynak eğer ki bir önceki kaynaktan daha iyi ise bu yeni kaynağı hafızasına kaydeder. Yeni kaynağın mevcut kaynak komşuluğunda belirlenebilmesinin benzetimi Eş. 3.2’de ifade edilmektedir [19].

$$v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}) \quad (3.2)$$

x_i ile ifade edilen her bir kaynak için bu kaynağın aslında çözümünün tek bir değişkeni (rastgele seçilen değişkeni, j) değiştirilerek x_i ’nin komşu bölgesinde v_i yeni besin kaynağı bulunur. Eş.3.2’de $j \in [1, D]$ arasında rastgele üretilen bir sayıdır. Rastgele seçilen j değişkeni değiştirilirken, yine rastgele seçilen x_k komşu çözümünün ($k \in \{1, 2 \dots SN\}$) j . değişkeni ile var olan kaynağın j . değişkeninin arasındaki farkları alınır ve $[-1, 1]$ aralığında rastgele değer alan bir sayı ile çarpıldıktan sonra var olan kaynağın j . değişkeninin üzerine eklenmektedir.

Eş.3.2’den de görüleceği üzere x_{ij} ve x_{kj} arasındaki fark minimuma yaklaştıkça yani çözümler birbirine yakın oldukça x_{ij} değişkenindeki değişim miktarı da azalacaktır. Bu durum bölgesel optimal çözüme yaklaştıkça değişim miktarı da beraberinde azalacaktır.

Bu işlemin sonunda meydana gelen v_{ij} nin daha önceden belli olan değişkeninin sınırları aşması durumunda ise j . değişkene ait olan alt veya üst limit değerlerine ötelenmektedir Eş.3.3 [19].

$$v_{ij} = \begin{cases} x_j^{min}, & v_{ij} < x_j^{min} \\ v_{ij}, & x_j^{min} \leq 0 \leq x_j^{max} \\ x_j^{max}, & v_{ij} > x_j^{max} \end{cases} \quad (3.3)$$

Limitler içerisinde üretilen v_i değişken vektörü yeni bir kaynağı ifade etmekte ve bunun nektar kalitesi hesaplanıp bir uygunluk değeri tayin edilmektedir (Eş.3.4):

$$fitness_i = \begin{cases} 1/(1 + f_i) & f_i \geq 0 \\ 1/abs(f_i) & f_i < 0 \end{cases} \quad (3.4)$$

Bu denklemde ki f_i ve v_i kaynağın yani çözümünün uygunluk değeridir. x_i ve v_i arasında nektar miktarlarına yani uygunluk değerlerine göre bir açgözlü (greddy) seçim işlemi tatbik edilir. Yeni bulunan v_i çözümü daha iyi ise görevli arı hafızasından eski kaynağın yerini silerek v_i kaynağının yerini hafızaya kaydeder. Aksi durumda görevli arı x_i kaynağına yönelmeye devam eder ve x_i çözümü daha fazla geliştirilemediği için x_i kaynağı ile ilgili geliştirememeye sayacı (failure) bir artar, geliştirebildiği durumlarda ise sayaç sıfırlanır [19].

3.2.2. Gözcü arılar

Bütün görevli arılar bir iterasyonda araştırmalarını bitirdikten sonra kovana geri döner ve buldukları kaynakların kalitesine göre ile ilgili GARı'lara bilgi paylaşımında bulunurlar. Bir GARı dans ile aktarılan bilgiden faydalanarak besin kaynağının kalitesi ile orantılı bir şekilde bir bölge (kaynak) seçer. Bu YAKA'nın aslında çoklu etkileşim gösterdiğinin bir işaretidir. Olasılıksal olarak seçim işlemi, algoritmada nektar miktarlarına göre uygunluk değerlerinin uygulanması ile gerçekleştirilir. Uygunluk değerine göre seçim işlemi rulet tekerliği, sıralamaya dayalı, stokastik (rastlantısal), örnekleme, turnuva yöntemi yada diğer seleksiyon işlemlerinden herhangi biri ile gerçekleştirilir. Temel YAKA da bu seleksiyon işlemi rulet tekerliği kullanılarak yapılmıştır. Tekerlekteki her bir dilimin açısının uygunluk değeri toplamına bölünmesi o kaynağın diğer kaynaklara göre göreceli seçilme olasılığı olduğunu verir (Eş.3.5) [51]:

$$p_i = \frac{fitness_i}{\sum_{i=1}^{SN} fitness_i} \quad (3.5)$$

Burada p_i kaynağın kalitesini SN görevli arı sayısını bildirmektedir. Bu olasılık hesaplama işlemine bağlı olarak bir kaynağın nektar miktarı arttıkça (uygunluk değeri arttıkça) bu kaynak bölgesini seçecek GARı sayısı da artacaktır. Bu özellik YAKA'nın pozitif geri besleme özelliğine denk gelmektedir [19].

Algoritma da muhtemel değerler hesaplandıktan sonra bu değerler kullanılmak sureti ile rulet tekerleğine göre seçme sürecinde her bir kaynak için [0,1] aralığında rastgele sayı üretilir ve p_i değeri bu üretilen sayıdan büyükse aynı görevli arılar gibi GARı'da Eş.3.2'yi kullanıp bu kaynak bölgesinde yeni bir çözüm üretirler. Yeni çözüm değerlendirmeye alınır ve kalitesi hesaplanır. Sonrasında yeni çözüm ile eski çözümün uygunluk değerleri karşılaştırılır ve en iyi olanın seçildiği açgözlü seleksiyon işlemine tabi olunur. Yeni çözüm daha iyi ise eski çözüm yerine bu çözüm hafızaya alınır ve çözüm geliştirememeye sayacı (failure) sıfırlanır. Eski çözümün uygunluğu daha iyi ise bu çözüm muhafaza edilir ve geliştirememeye sayacı (failure) bir artırılır. Bu işlem, tüm GARı'lar besin kaynakları bölgelerine gönderilene kadar devam eder [19].

3.2.3. Kaşif arılar

Bir iterasyon sonrası bütün görevli ve GARı'lar arama işlemleri bittikten sonra çözüm geliştirememeye sayaçları (failure) kontrol edilmektedir. Bir arının bir besin kaynağından yararlanıp yararlanmadığı, yani gidip geldiği besin kaynağının nektarının bitip bitmediği çözüm geliştirememeye sayaçları tarafından belirlenebilir. Bir kaynak için çözüm geliştirememeye sayacı belli bir eşik değerinin üstünde ise, artık bu kaynağın görevli arısının tükenmiş demektir ve o çözümü terkedip kendisi için başka bir çözüm bulması gerekmektedir. Bu da tükenen kaynakla ilişkili olan görevli arının KARı olması anlamını teşkil etmektedir. KARı haline dönüşür sonra, bu arı için rastgele çözüm arama işlemi başlar (Eş.3.1). Kaynağın terk edildiğinin belirlenmesi için kullanılan eşik değeri YAKA'nın önemli bir kontrol aşamasıdır ve "sınır" olarak isimlendirilmektedir. Temel YAKA'da her iterasyonda bir tek KARı'nın gezinmesine müsaade edilir.

Bütün bu birimler arasındaki bağlantı ve çevrim Şekil 2'deki gibi bir akış diyagramı ile gözlemlenebilmektedir [19].

3.3. Seleksiyon Mekanizmaları

YAKA’da dört çeşit seleksiyon süreci uygulanmaktadır. Bunlar:

1. Potansiyeli yüksek olan besin kaynaklarının belirlenmesi aşamasında Eş.3.5 olasılık değerlerinin hesaplandığı küresel olasılık temeline dayanan seleksiyon süreci.
2. Görevli ve GArı’ların besinin rengi, şekli, kokusu gibi nektar kaynağının cinsinin belirlenebilmesini sağlayan görsel bilginin kullanılması ile bir bölgedeki besin kaynağının bulunmasına aracı olan bölgesel olasılık tabanlı seleksiyon süreci (Eş.3.2).
3. İArı ve GArı’ların daha iyi olan besin kaynağını tayin etmek amacı ile kullandıkları açgözlü (greddy) seleksiyon.
4. KArı’lar tarafından (Eş.3.1) yardımıyla oluşturulan rastgele seleksiyon.

Aslında tüm bu seleksiyon yöntemlerinin birlikte kullanılmasıyla, YAKA’nın hem iyi bir küresel araştırma hem de bölgesel araştırma yapabilmesi sağlanır [19].

3.4. Arıların Yem Bulma Davranışları

Doğadaki bal arılarının görevi kovan civarında bulunan nektar kaynaklarından toplanan balı kovana stoklamaktır. Bu görevi yerine getirirler iken arılar iki farklı gruba ayrılır:

Çalışanlar: Herhangi bir besin kaynağından kovana besin getirebilen arılar,

Çalışmayanlar: Bir besin kaynağını bırakmış veya herhangi bir besin kaynağına uğramamış arıları temsil eder.

Besin kaynakları, kovan civarındaki nektar kaynaklarıdır. Besin kaynakları bal arılarında birden çok faktör ile belirlenebilir; kovana yakınlık, nektar miktarı, enerji vb. Bir nektar kaynağının değerini ise nektar kaynağından getirilen nektarın harcanan enerji ile oranı tarafından belirlenir.

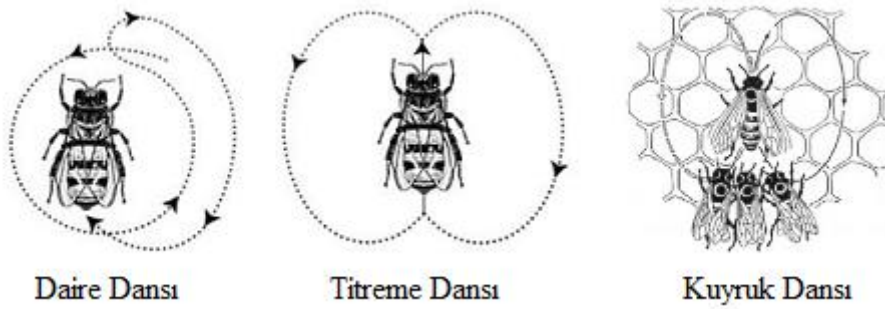
Çalışan toplayıcılar, belirli bir besin kaynağından nektarı kovana getiren arılar olduğunu belirtmişlerdir. Bunlar mevcut durumda nektar kaynağından yararlanmaya ve çalışmaya devam

ederler. Bu arılar yuvaya kaynağın değerini ve kovan-kaynak arasındaki mesafe bilgisini iletirler.

Çalışmayan toplayıcılar, faydalanılacak bir besin kaynağı ararlar. KArı ve GArı olmak üzere iki türdür. KArı'lar yuvanın civarında yeni besin kaynağı bulurken yuvada bekleyen GArı'lar çalışan toplayıcılar aracılığı ile nakledilen bilgiyi beklerler. Bir kovandaki ortalama KArı'ların sayısı uygun koşullarda toplam kovandaki arıların sayısının %5-10'u kadardır [20].

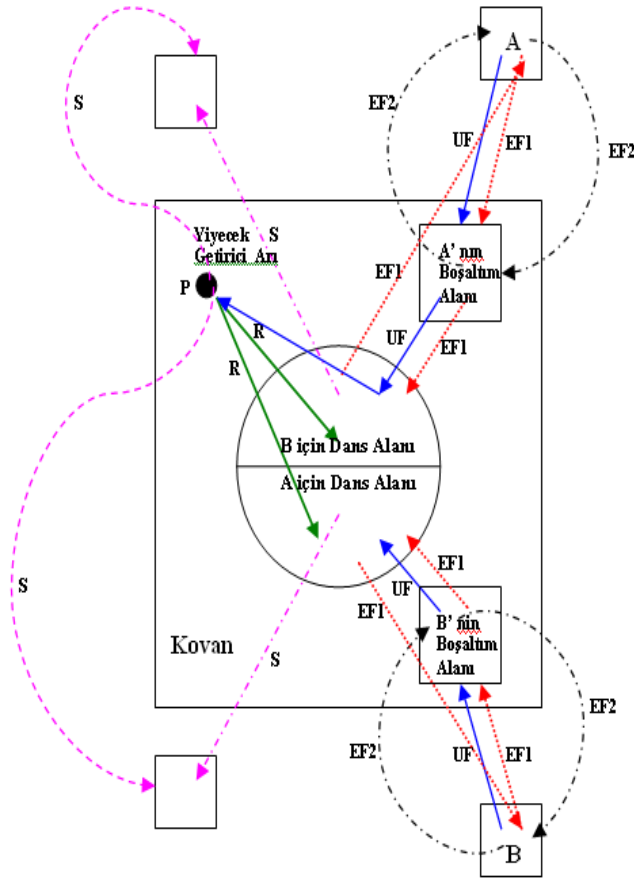
Arılar arasındaki bilgi paylaşımı ortak bilginin meydana gelmesinde en önemli durumdur. Genellikle kovanın bazı kısımlarını ayırt edebilmek muhtemeldir Bir kovan için en önemli kısım ise bilgi paylaşımının yapıldığı dans edilen (dancing area) bölgedir. Arılar arasında bilgi paylaşımı dansla gerçekleşir ve bu dans "dancing area" denilen kısımda gerçekleştirilir. Bal arıları bilgi paylaşımı neticesinde kaliteli yeni besin kaynaklarını bulurlar. Bal arılarında kaynağın konumu ve karlılığı "Waggle Dance" denilen bir dans ile diğer arılar bilgilendirilir [21].

Kaynağın yuvaya olan uzaklığına göre türlü danslar bulunmaktadır: Dairesel Dans, Kuyruk Dansı ve Titreme Dans'larıdır. Bu danslar Şekil 3.2'de gösterilmiştir.



Şekil 3.2. Kaynağın Kovana Olan Mesafesine Göre Arıların Dans Çeşitleri [22,23]

Daire Dansı Seçilen besin kaynağının yuvaya olan mesafesi maksimum 50-100 metre kadar olduğu için bu dans yön ve mesafe bilgisini belirtmez.



Şekil 3.3. Arıların yem arama davranışları [24]

Kuyruk Dansı 100 metre ile 10 kilometre arası mesafede olan geniş bir bölge içerisinde bulunan besin kaynakları ile ilgili bilgi paylaşımında kuyruk dansı kullanılmaktadır. Bu dans 8 rakamına benzetilen bir dans türüdür. Dansı gözlemleyen arıların bir titreşim oluşturması ile bu dansı yapan arı, dansı bitirir. Her 15 saniyede dansın tekrar sayısı, nektar kaynağının mesafesi ile ilgili bilgi içermektedir. Daha az tekrar sayısı ile daha uzak alanlar ifade edilmektedir. Yön bilgisi yukarıdaki Şekil 4'teki gibi 8 rakamı biçimindeki dansın açısı bilgisinden elde edilebilir. Şekil 4'te verilen örnekte dansı gözlemleyen arılar, danstan güneşle besin arasındaki açının 45 derece olduğunun bilgisine ulaşırlar. Arılar kendi aralarında besinin yuvaya olan uzaklığına göre buldukları kaynakla ilgili bilgi paylaşımında daha hızlı dans ederler [25].

Titreme Dansı Belirlenen nektar eğer çok kaliteli ise ve miktar bakımından da çok zengin ise bu titreme dansı yapılır. Titreme dansının arılar arasındaki anlamı “Bir besin kaynağı buldum, içinde taşımak için ve zengin nektar var, işleme görevine başlamak istiyorum”

anlamına gelir ve amacı kovandaki besin kapasitesi ve besin getirme işlemi arasındaki dengeyi sağlamaktır. Bütün bilgiler gerçekleştirilen dans çeşidine göre dans alanında ki GArı'lar ile paylaşıldıktan sonra GArı'lar besin kaynakları arasında mesafe, nektar oranı ve kalitesine göre karar alırlar. Karar alındıktan sonra İArı'lar ya yeniden kendi kaynaklarına döner ve nektar taşımaya başlarlar ya da yeni besin kaynaklarını bulmaya başlarlar. KArı'lar ise rastgele besin kaynaklarını bulmak için giderler. GArı belirlenen dansa göre nektar getirmek için besin kaynağına doğru yola çıkabilir veya gözlemlemeye devam eder. Arılar arasında bir sayısal denge vardır. Yani bütün arılar aynı zamanda çalışmaya çıkmazlar. Çalışan arılar ile bekleyen arılar arasında her zaman bir denge söz konusudur. Arılar bu dengeyi şu şekilde yerine getirirler.

Pozitif Geri Besleme: Besin kaynağın kalitesi artma meydana gelirse bu kaynağı seçen GArı sayısı da artar.

Negatif Geri Besleme: Besin kaynağı bitmeye başladığında terk edilir.

Salınımlar: KArı'lar besin kaynağı aramak için rastgele arama yaparlar.

Çoklu Etkileşimler: Bilgi aktarımı dans bölgesinde meydana gelir [18].

Besin aramak ile görevli olan arıların daha net anlaşılabilmesi açısından Şekil 3.3'te verilen modelin irdelenmesi yararlı olacaktır. A ve B ile ifade edilen iki tane bulunmuş olan besin kaynağı alanı olduğunu düşünelim. Araştırmanın en başında potansiyel besin arayıcı, görevi belirsiz bir İArı olarak bulmaya başlayacak ve bu arı yuva etrafındaki besin kaynaklarının yerleri hakkında bir bilgiye sahip değildir.

Bu durumda bu arı için iki muhtemel durum söz konusudur.

1. Bu arı KArı olabilir ve iç ve dış faktörlere bağlı olarak besin aramaya başlayabilir (Şekil 3.3'te S ile ifade edilen).
2. Bu arı kuyruk dansını izleyen bir GArı'da olabilir ve gözlemlediği dansla aktarılan besin kaynağına doğru yola çıkabilir(Şekil 3.3'te R ile ifade edilen). Bir besin kaynağı bulunduktan sonra arılar imkanları dahilinde bu besin kaynağının konumu hafızaya alır ve ardından nektar toplamaya başlarlar. Bu arı artık nektarın zenginliğine göre görevli arı halini

almış olur. İArı nektarı aldıktan sonra yuvaya döner ve bunu besin depolayıcılara aktarır.

Nektar taşıdıktan sonra arı için üç tür seçenek söz konusudur:

1. Gittiği besin kaynağını terkederek bağımsız İArı haline gelebilir (Şekil 3.3'te UF ile ifade edilen).
2. Gittiği besin kaynağına tekrar gitmeden önce dans edebilir ve yuvadaki diğer arıları da aynı besin kaynağına gönderebilir (Şekil 3.3'te EF1 ile ifade edilen).
3. Diğer arılar için herhangi bir yönlendirmede bulunmadan direkt besin kaynağına gidebilir (Şekil 3.3'e EF2 ile ifade edilen).

3.5. YAKA Sözde Kodu

Önceki bölümlerde genel prensipleri verilen YAKA'nın bilgisayar ortamında kodlarının yazılabilmesi için sözde kodu aşağıdaki gibi özetlenebilir:

- 1.Eş.3.1 tarafınan bütün x_{ij} $i=1 \dots SN$, $j =1 \dots D$, çözümlerine başlangıç değerlerinin tayin edilmesi ve çözüm geliştiremememe durumunda sayaçların sıfırlanması $failure_i=0$
2. $f_i=f(x_i)$ denklem değerlerinin ve bu değerlerin karşılığı olan uygunluk değerlerinin belirlenmesi
- 3.Repeat
- 4.for $i=1$ to SN do
- 5.Eş.3.2 ile x_i çözümünde ki görevli arı için yeni bir besin kaynağı üretilmesi yani v_i ve $f(v_i)$ 'yi Eş.3.4'te yerine koyarak bu çözümün uygunluk değerini belirleme
6. x_i ve v_i arasında greedy yani açgözlü bir seçim işlemi uygulanması ve daha iyi olanın seçilmesi
7. x_i çözümü eğer daha fazla gelişemiyorsa çözüm geliştiremememe sayacını bir artır $failure_i= failure_i+1$, eğer gelişme mevcut ise sıfırla, $failure_i=0$
- 8.End For
- 9.Eş.3.5 ile GArı'ların seçim yapar iken kullanacakları uygunluk değerine bağlı olasılık değerlerini p_i belirle
10. $t=0$ $i=1$
- 11.Repeat

- 12.If $\text{random} < p_i$ then
- 13.Eş.3.2'yi kullanarak GARı için yeni bir besin kaynağı, v_i belirle
14. x_i ve v_i arasında açgözlü seçim işlemi uygula ve daha iyi olan çözümü belirle.
15. x_i çözümü eğer gelişmiyorsa çözüm geliştirememeye sayacını bir artır $\text{failure}_i = \text{failure}_i + 1$, eğer çözüm gelişmemişse sıfırla, $\text{failure}_i = 0$
16. $t = t + 1$
- 17.End If
- 18.Until $t = \text{SN}$
- 19.if $\max(\text{failure}_i) > \text{limit}$ then
20. x_i 'yi Eş.3.1 ile üretilmiş rastgele bir çözümle değiştir.
- 21.End If
- 22.En iyi çözüm değerini hafızaya al.
- 23.Until Durdurma kriteri

3.6. Yapay Arı Kolonisi Algoritmasının Temel Özellikleri

YAKA'nın bilinen özellikleri şu şekilde yazılabilir;

- 1.Hayli esnek ve kolaydır.
- 2.Gerçek besin bulucu arıların hareketlerine çok benzer biçimde benzer.
- 3.Sürü zekâsına bağlı bir algoritma türüdür.
- 4.Nümerik problemler için geliştirilmiş fakat ayrık problemler içinde uygundur.
- 5.Hayli az kontrol değişkeni içerir.
- 6.KArı'lar aracılığıyla oluşturulan evrensel, görevli ve GARı'lar aracılığıyla gerçekleştirilen alansal araştırma yeteneğine sahiptir ve ikisi de paralel bir şekilde gerçekleşir [19].

3.7. Yapay Arı Kolonisi İle İlgili Yapılan Bazı Çalışmalar

Babayiğit ve diğerleri (2012) “Modifiye Yapay Arı Koloni Algoritması ile Nümerik Fonksiyon Optimizasyonu”, Araştırmada, standart YAKA’nın yerel çözümlerden faydalanma mekanizmasını geliştirilmiş performansının yükseltmesi ile modifiye bir YAKA öne sürülmüştür. Öne sürülen bu modifiye YAKA’da, standart YAKA’nda GArı’ların seleksiyonundaki olasılık değeri yerine yeni bir olasılık hesaplama metodu ile yeni besin arama mekanizmaları kullanılmış [27].

Pamuk (2013), “Enerji Sistemlerinde Yapay Arı Kolonisi (YAK) Algoritması Kullanarak Yük Akışı Optimizasyonu” Araştırmada, yük akışı eşitlikleri altında iletim hattı kayıplarını asgari yapan kontrol parametreleri YAKA tarafından PowerWorld simülasyon programı kullanılarak oluşturulmuştur [28].

Özkiş (2013), “Sayısal Optimizasyon Problemlerinin Çözümü İçin Yapay Arı Kolonisi Algoritmasının İyileştirilmesi” Bu tez çalışmasında, ABC algoritması üzerinde çeşitli düzenlemeler yapılarak Adaptif Boyut Limitli ABC (ABL-ABC, Adaptive Dimension Limit ABC, ADL-ABC) ve Hızlandırılmış ABC (H-ABC, Accelerated ABC, A-ABC) olarak isimlendirilen iki farklı algoritma önerilmiştir. Önerilen algoritmaların performans analizi, literatürde sıklıkla kullanılan sayısal optimizasyon problemleri üzerinde orijinal ABC algoritmasının ve yaygın olarak kullanılan diğer optimizasyon algoritmalarının elde ettiği sonuçlarla kıyaslanarak yapılmıştır [29].

Çavuş ve diğerleri (2017), “İnsansız Hava Araçları İçin Yapay Arı Kolonisi Algoritması Kullanarak Rota Planlama” Araştırmada, YAKA’nın simülasyon uygulamalarını gerçekleştirilmesi hedefi ile C# programlama dili tarafından kullanıcı etkileşimli bir arayüz tasarlanmıştır. Rota planlaması için yapılan deneysel çalışmalar, YAKA’nın uygun rotalar bulmada etkili sonuçlar verdiğini göstermiştir [30].

Kıran ve diğerleri (2010) “Arı Kolonisi İle Şoför-Hat-Zaman Optimizasyonu” Çalışmada Konya Cumhuriyet Otobüs Durağına ait yolcular için hazırlanmış olan araç kalkış saatleri bir problem olarak ele alınmış ve hatlara ait seferlerin şoförlere dağıtılması amaçlanmıştır. Arı sisteminin bir metodu olan arı kolonisi optimizasyonu tez problemindeki şoförlerin belirli saat çalışması kaydıyla optimum görev dağılımını oldukça başarılı şekilde gerçekleştirmiştir [31].

Kartal (2015), “Yapay Arı Kolonisi Algoritması ile Finansal Portföy Optimizasyonu” Önceden çözülmesi karmaşık ve zaman alıcı olan Portföy Optimizasyonu (PO) problemleri, son zamanlarda kullanımı oldukça artan sezgisel tekniklerle çözülmeye başlanmıştır. YAKA ile PO problemine çözüm aranmıştır. Literatürde yer alan 5 tane örnek problem ve BIST30 verileri ile sınamalar yapılmıştır. Daha önce, Tavlama Benzetimi (TB), Tabu Arama (TA), Genetik Algoritma (GA) ve Parçacık Sürü Optimizasyonu (PSO) yöntemleriyle elde edilen sonuçlarla kıyaslanmıştır. YAKA’dan elde edilen sonuçların, TB, TA, GA ve PSO’dan elde edilen sonuçlar kadar iyi olduğu gösterilmiştir [32].

ASLAN ve diğerleri (2018), “Paralel Yapay Arı Koloni Algoritması için Rastgele Kaynaklar ile Yeni Bir Göçmen Üretme Yaklaşımı” Bu çalışmada, alt nüfusda göç ettirmek üzere seçilen en iyi çözümün değişkenlerinin aynı alt nüfustaki rastgele seçilmiş çözüm ya da çözümlerin daha uygun değişkenleri ile güncellendiği yeni bir yaklaşım öne sürülmüş ve bu yaklaşımın paralel YAKA’nın performansı üzerindeki etkileri irdelenmiştir [33].

Karaboğa ve diğerleri (2007), “Yapay Sinir Ağlarının Eğitiminde Yapay Arı Kolonisi (ABC) Algoritması” Bu çalışmada, son zamanlarda önerilen bir algoritma olan yapay arı kolonisi algoritmasının performansı, sinyal işleme uygulamalarında yaygın olarak kullanılan yapay sinir ağları eğitimi konusunda test edilmiş ve algoritmanın performansı, diferansiyel gelişim ve parçacık sürüsü optimizasyonu ile karşılaştırılmıştır. Sonuçlar, Yapay Arı Koloni Algoritmasının diğer algoritmalarından daha iyi performans gösterdiğini göstermektedir [34].

Erkol (2017), “Ters Sarkaç Sisteminin Yapay Arı Kolonisi Algoritması ile Optimizasyonu”, Araştırmada, kullanımı gün geçtikçe fazlalaşan YAKA kullanılarak ters sarkacı denetleyen Oransal-İntegral-Türevsel kontrolcünün en iyilemesi yapılmıştır. Ters sarkacın simüle edilmesi ve kontrolü simülasyon ortamında yapılmış olup, benzetim sonucunda elde edilen veriler YAKA’nın geleneksel metodlardan daha iyi ve daha hızlı sonuçlar verdiği gözlemlenmiştir [35].

Yiğitbaşı (2014), “Yapay Arı Kolonisi Optimizasyonu ile Kenar Bulma” Araştırmada öncelikle YAKA en iyilenmesi aktarılmıştır. Sonrasında optimizasyon yöntemlerine değinilmiştir. Ardından YAK en iyilenmesi ve geliştirilen yöntem açıklanmıştır. Son olarak, geliştirilen yöntemin RADIUS/DARPA-IU Fort Hood veritabanından alınan çeşitli gri seviye nesne görüntüleri, siyah-beyaz görüntüler ve gri seviye hava resimleri üzerinde

uygulama sonuçları verilmiş ve sonuçların literatürde daha önceden yapılan uygulamalarla karşılaştırılması yapılmıştır [6].

Bao ve diğerleri (2009), “Yapay Arı kolonisi algoritmasında seçim mekanizmasının karşılaştırılması ve analizi” Yapay arı kolonisi (ABC) algoritması, geliştirilmesi ve çözülmesi gereken birçok konunun var olduğu ve bal arısı sürülerinin belirli akıllı davranışlarına dayanan, yeni bir küresel stokastik optimizasyon algoritmasıdır. Gözcü arılar, ABC algoritmasında orantılı seçim stratejisine bağlı olarak besin kaynağı seçiminde zamanı kötü kullanarak erken evrimsel sürece neden olabilirler. Bu yazıda, gelişen popülasyon çeşitliliğini erken önlemek için yıkıcı seçim stratejisi, turnuva seçim stratejisi ve rütbe seçim stratejisi gibi çeşitli erken seçim stratejilerini önlemek için, seçim mekanizmaları simülasyon yoluyla karşılaştırılıp analiz edilmiş ve sonuçlar modifiye algoritmanın temel YAKA’na göre daha iyi olduğu görülmüş [36].

Öztürk ve diğerleri (2014), “Küresel En İyi Yapay Arı Koloni Algoritması İle Otomatik Kümeleme” Kümeleme işlemi, verilerin analiz edilmesinde çok önemli bir teknik olup görüntü işleme ve istatistiksel veri analizi başta olmak üzere birden fazla alanda kullanılır. Özellikle son zamanlarda kümeleme probleminin çözümü için yapılan çalışmaların fazlaştığı görülmektedir. Bu çalışmanın amacı, otomatik kümeleme problemini çözmek için YAKA’nın global araştırma yeteneği geliştirilmiş ve algoritmanın vektörel araştırma yapabilmesi gerçekleştirilmiştir. Öne sürülen metod iyi bilinen data ve görüntü setleri üzerinde denenmiştir. Elde edilen sonuçlar neticesinde öne sürülen yöntemin diğer yöntemlere oranla daha iyi bir performans verdiği ve otomatik kümeleme problemlerinin çözümünde rahatlıkla kullanılabileceği gözlemlenmiştir [37].

Batbat ve diğerleri (2016), “Ayrık Yapay Arı Kolonisi Algoritması ile Protein Yapısı Tahmini” Proteinlerin görevleri doğal durumları olarak isimlendirilen üçüncül yapıları ile ilgilidir. Bugün birçok proteinin birincil yapı olarak isimlendirilen dizilimleri bilinmekle beraber işlevlerini açıklayan üçüncül yapıları bilinmemektedir. Ele alınan dizide bulunan amino asitlerin üçüncül yapı oluşumuna yol açan özelliklerinin kullanılarak üçüncül yapılarının tahmin edilmesi proteinlerin gerçek doğasının bütünüyle anlaşılması ve bilimsel açıdan kullanılmasına imkan sağlayacaktır. Proteinlerin doğal yapılarına katlanması ile ilgili yapılan çalışmalarda en düşük serbest enerji düzeyinin elde edilmesi eğilimi görülmüştür. Serbest enerji düzeylerinin hesaplanmasında, sudan kaçma özelliği yoğun olarak

kullanılmıştır. Bu özelliğe göre sınıflandırılan amino asitleri temel alan HP modeli basitliği ve gerçekçiliği ile dikkat çekmiştir. Çalışmada, HP modelinin doğadaki protein davranışı incelenmiş geliştirilmiş bir türü olan Üç Boyutlu Hidrofobik Polar Yan Zincir modeli kullanılmıştır. Aday protein modellerinin oluşturulmasında arıların besin bulma davranışlarını modelleyen YAKA temelli oluşturulan ayırık model öne sürülmüştür. Bu gibi çalışmalarda kullanılan başarımların test edildiği örnek dizilerde daha az maliyetle iyi sonuçlar alınmıştır [38].

Özer ve diğerleri (2014), “Doğrusal parametrik ve doğrusal olmayan gerçek sistemlerin yapay arı kolonisi algoritması kullanılarak modellenmesi” Araştırmada, doğrusal olan parametrik ve doğrusal olmayan gerçek sistemlerin sürü zekası yaklaşımına örnek olan yapay zeka en iyileme yöntemlerinden YAKA ile modellenmesi gerçekleştirilmiş olup, Genetik algoritma (GA) ve Klonal Seçme Algoritması (KSA) ile başarımlarını karşılaştırılmıştır. Simülasyon çalışmalarında literatürde kıyaslama problemlerinde sıklıkla kullanılan bir adet doğrusal parametrik ve bir adet doğrusal olmayan gerçek sistem bu algoritmalar tarafından modellenmiştir. Simülasyon sonuçlarına göre doğrusal parametrik sistemlerin modellenmesinde YAKA, GA’ya çok yakın bir sürede parametre tahmini yapmış, farklı koloni sayılarında ise hem GA’dan hemde KSA’dan daha düşük modelleme hatası ile sonuçlar verdiği gözlemlenmiştir. Doğrusal olmayan gerçek sistem modellemelerde ise aynı durum söz konusu olup, koloni sayılarındaki değişime bağlı olarak YAKA, GA ve KSA’na göre daha düşük hata ve daha kısa sürede sistemi modelleyebildiği görülmüştür [39].

Zorarpacı ve diğerleri (2015), “Nitelik Seçimi İçin Yapay Arı Koloni Tabanlı Algoritma” Nitelik seçiminin asıl amacı çalışma süresini ve sınıflandırıcının verimliliğini iyileştirmek için sınıflandırma süreci esnasında kullanılacak olan özellik sayısını en aza indirmektir. Bu çalışma, sınıflandırma süreçleri için yakın zamanda geliştirilmiş başarılı bir sürü zekası algoritması olan YAKA en iyileme tekniğine dayalı bir nitelik seçim metodu geliştirilmesidir. Geliştirilen metod UCI veritabanından elde edilen ve sınıflandırma problemlerinde sıkça kullanılan 15 veri kümesi üzerinde denenmiştir. Deney bulguları öne sürülen YAKA tabanlı algoritmanın sınıflandırma çalışmaları için iyi nitelikleri seçebileceğini gösterebilmiştir [40].

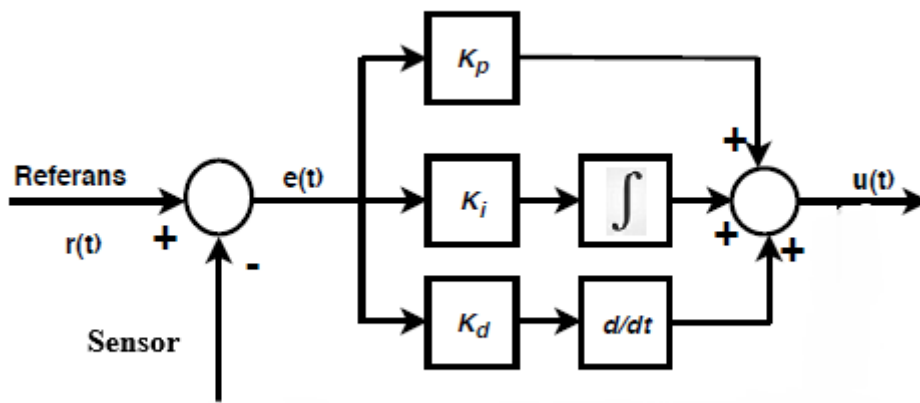
Baydoğan ve diğerleri (2014), “Tip imar yönetmeliğine uygun vaziyet planlarının yapay arı

kolonisi yöntemiyle eniyilenmesi” Araştırmada, Tip İmar Yönetmeliğinin hangi durumları ortaya çıkardığını tasarımcılara gösterebilmek üzerine bir yazılım geliştirilme çalışmasıdır. Evler için yerleşim planları için tip imar yönetmeliği koşullarında, doğru yerleşimi bulma, yoğunluk hesaplamalarını yapabilme ve sonuçta çıkan kütle büyüklüğü ve kat adedini gösterebilecek bir program ortaya koymak ve çalıştığını örnek bir alan üzerinde denemek araştırmanın asıl içeriğini oluşturabilmektir. Çalışmanın sonuçlarını denemek ve konvansiyonel sistemlerle karşılaştırma yapabilmek için; ada ölçeğinde konut alanları için, TOKİ’ nin toplu konutlar için yaptığı yerleşim planlarını; vaziyet planı ölçeğinde mekansal örgütlenme, YAKA ile yeniden yapılmıştır. E (emsal), Hmax, TAKS ve KAKS, çekme aralıklarına ve diğer hesaplarına uygunluk, uygun çözüm üretim süresinde karşılaştırma kriterleri olarak benimsenmiştir. Nihayetinde; algoritmanın, tip imar yönetmelikleri şartlarını sağlayabildiği, uygun yerleşim alternatiflerini bulabildiği, yoğunluk hesaplarını yapabildiği ve sonuçta çıkan kütle büyüklüğü ve kat adedini gösterebildiği gözlemlenmiştir. Konvansiyonel sistemlerle yapılan yerleşim planlarından hem alternatif olarak çok daha fazla çözüm ürettiği hem de zaman açısından avantaj sağladığı açıkça görülmüştür. Fakat modellemenin doğası gereği; bu çalışmanın ürettiği durum planlarının daha iyiyi daha uygunu bulmak için bir basamak olarak uygulanabileceği daima akılda tutulmalıdır [41].

4. PID VE KAYAN KIPLİ KONTROLÇÜLER

4.1. PID Kontrolcü

PID, üç ana kontrol etkisi (P,I,D) birleşiminden meydana gelir. PID ağırlık parametreleri oluşturan bölümlerin her biri bir katsayı ile temsil edilirler. Bunlar (K_p, K_i, K_d) bütün sistemler için farklı değerler alırlar. Şekil 4.1 'de PID kontrolörünün iç yapısına ait blok şema gösterilmiştir. Bu blok şemadan da görüleceği üzere PID kontrolörünün yapısı oransal-kazanç, integral-alıcı ve türev-alıcı devrelerinin sentezinden oluşmaktadır.



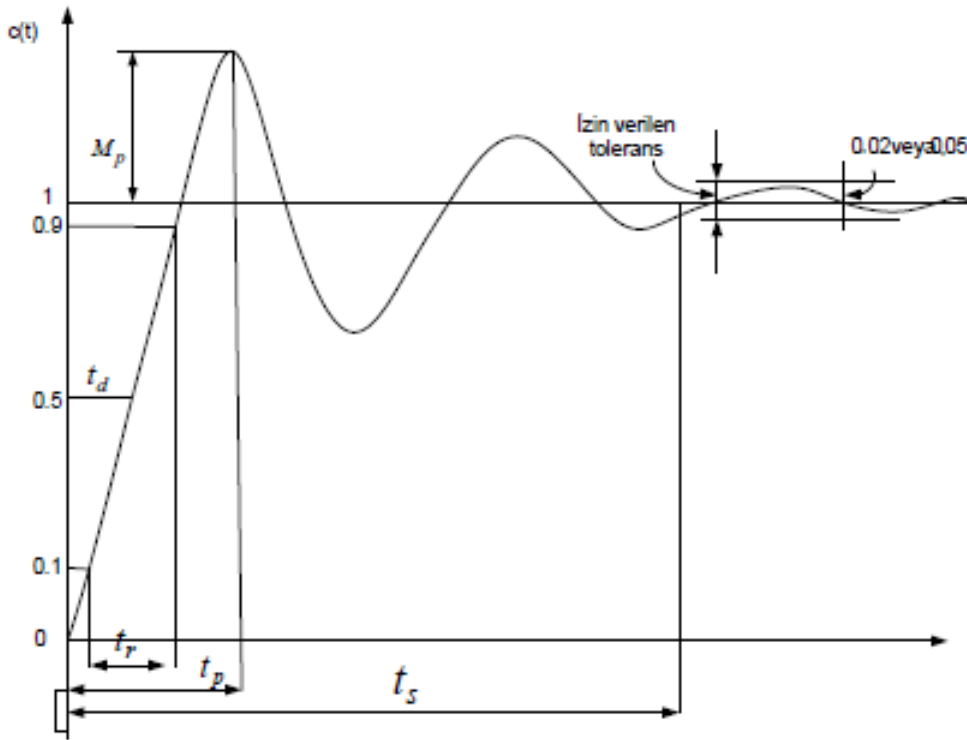
Şekil 4.1. PID Denetleyicinin blok diyagramı

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{d}{dt} e(t) \quad (4.1)$$

şeklinde ifade edilir. Bu denklemde $e(t)$ hata değerini ifade etmektedir:

$$e(t) = r(t) - y(t) \quad (4.2)$$

İntegral kontrol, sistemde oluşabilecek kalıcı-durum hatasını sıfır yaparken türev kontrol, sadece PI kontrolörün etkisi kullanılması durumuna göre sistemin aynı bağıl kararlılığı için cevap hızını artıracaktır. Buna göre PID kontrolör, sistemlerde sıfır kalıcı-durum hatası ile hızlı bir cevap verebilen bir kontrolcüdür. Herhangi bir sistemin zaman cevabının irdelenmesinde kullanılan parametreler Şekil. 4.2'de gösterilmiştir.



Şekil 4.2. Zaman düzlemi cevabı parametreleri

Bu parametrelerin tanımları şöyledir.

Gecikme Zamanı, t_d : Gecikme zamanı, sistem cevabının son değerinin yarısına ilk ulaştığı an için geçen süredir. 1. derecede bir sistemde gecikme zamanı, t_d zaman sabiti T' 'ye eşit alınabilmektedir.

Yükselme Zamanı, t_r : Yükselme zamanı ise, sistem cevabının nihai değerinin %10'dan %90'nına, %5' den %95'ine veya %0'dan %100'üne ulaşma süresidir. Aşırı sönümlü 1. dereceden sistemlerde %0 – 100 ulaşması için geçen zaman yükselme zamanı olarak kabul edilir. Salınlımlı, sönümlü sistemler için genellikle % 10 – 90 ulaşması için geçen zaman yükselme zamanı olarak kabul edilir.

Aşım Zamanı, t_p : Sistemdeki tepe zamanı cevabın referansı aşarak ilk kez aşarak bir tepe yaptığı noktaya ulaşması için gerekli zaman olarak adlandırılır.

Maksimum Aşım, M_p : Sistem cevap eğrisinde $t = t_p$ iken ulaştığı genlik değeridir. Maksimum aşımın miktarı, doğrudan direkt olarak sistemin bağıl kararlılığını belirler. 1.

derece sistemlerde cevap eğrisi asla olması gereken son değeri aşmadığından maksimum aşma tanımsızdır, yani sıfırdır.

Yerleşme Zamanı, t_s : Yerleşme zamanı, sistemin cevap eğrisinde oluşan salınım genliklerinin tolere edilebilir bir değerin sınırlarına erişmesi için geçen süredir. Kabul edilebilir bu tolerans değeri ise genellikle nihai değerin %5 veya %2 lik aşma değerleri olarak ifade edilir. t_s denetim sistemlerinde tanımlanan en büyük zaman sabitidir. I.dereceden gecikmeli sistemlerde t_s 'nin yükselme zamanına eşit olduğu kabul edilir.

Sönüm Oranı (ζ): Sönüm oranı (ζ), kararlı sistemler için $0 < \zeta < 1$ arasındadır. ζ küçüldükçe sistem cevabının aşımı artar [42].

Kullanılacak olan kontrolör sistemin karakteristiğine uygun olarak seçilir. PID kontrolörün türleri:

- Oransal Kontrol (P kontrol)
- Oransal Türevsel Kontrol (PD kontrol)
- Oransal İntegral Kontrol (PI Kontrol)
- Oransal İntegral Türevsel Kontrol (PID) [43].

4.1.1. Oransal kontrol (P Kontrol)

Oransal-kontrol en basit kontrol etkisini ifade eder. Kontrolcünün çıkışı o andaki hatanın değeri ve belirli kazanç değerinin çarpımıyla bulunur. Bu kazanç kontrolcü çıkışında sistem hatasının duyarlılığını gösterir. Kazanç değeri arttırıldıkça kontrolcünün reaksiyonu artar.

$$u(t) = K_p e(t) \quad (4.3)$$

Oransal Kontrol;

- Sistem tepkisini hızlandırır,
- Sürekli rejim hatasını azaltır,
- Aşma değerini arttırır.

4.1.2. Oransal türevsel kontrol (PD kontrol)

Türevsel kontrolün bu geçici rejim esnasında oluşabilecek salınımları azaltma etkisi oransal kontrolün neden olduğu aşım da ki artmayı azaltacaktır. Bu neden ile oransal kontrolün türevsel kontrol etkisi ile birlikte kullanılması izafi olarak daha iyi bir kontrolör yapı oluşturur.

$$u(t) = K_p e(t) + K_d \frac{d(e(t))}{dt} \quad (4.4)$$

Türevsel oransal kontrolcünün aşımı azalttığı ve sistemin cevap hızını arttırdığı öngörülebilir. Fakat türev etkisi bilhassa ölçme sistemindeki parazitlere karşı sistem duyarlılığını arttıracaktır.

4.1.3. Oransal integral kontrol (PI kontrol)

İntegral kontrol etkisi, oransal kontrol etkisi ile birlikte kullanıldığı zaman daha etkili bir şekilde belirgin olur. İntegral etkisi, kontrolcünün devreye müdahil olmasından belirli bir zaman sonra etkin hale gelmektedir. Yine oransal kontrol etkisi hata azaldıkça zayıfladığı bilindiği üzere bu iki etkinin birbirini tamamlayıcı olarak kullanılabileceği kolaylıkla öne sürülebilir.

$$u(t) = K_p e(t) + K_i \int e(t) dt \quad (4.5)$$

Oransal kazanç çabuk bir şekilde sistem cevabını giriş değerine yaklaştırdıkça etkisini kaybedecektir ve bu noktadan sonra integral etkisi kalıcı hatayı yok edecek ve sistemi çıkışını uzun sürse de referans değerine ulaştıracaktır.

4.1.4. Oransal integral türevsel kontrol (PID kontrol)

PID kontrolcüler, sağlam ve etkili performans göstermeleri sebebi ile endüstriyel ortamlarda sıkça tercih edilen denetleyicilerdir. PID kontrolcüyü meydana getiren oransal (P), integral (I), türev (D) kazanç ağırlık parametrelerinden her biri sistemin çalışması esnasında farklı türlerde etki etmektedirler.

Oransal denetleyicinin (K_p), yükselme zamanını azaltmada etkisinin var olduğu bilinir ve azaltır da, fakat hiçbir zaman tam anlamıyla yok edemez.

İntegral denetleyicinin (K_i) kararlı hal hatasını sıfırlama da etkisi olduğu bilinir.

Türevsel denetleyicinin ise (K_d) sistem kararlılığını arttırmada etkisi olduğu bilinir, aşımı azaltır ve geçici rejim davranışı gibi istenmeyen durumları düzeltir.

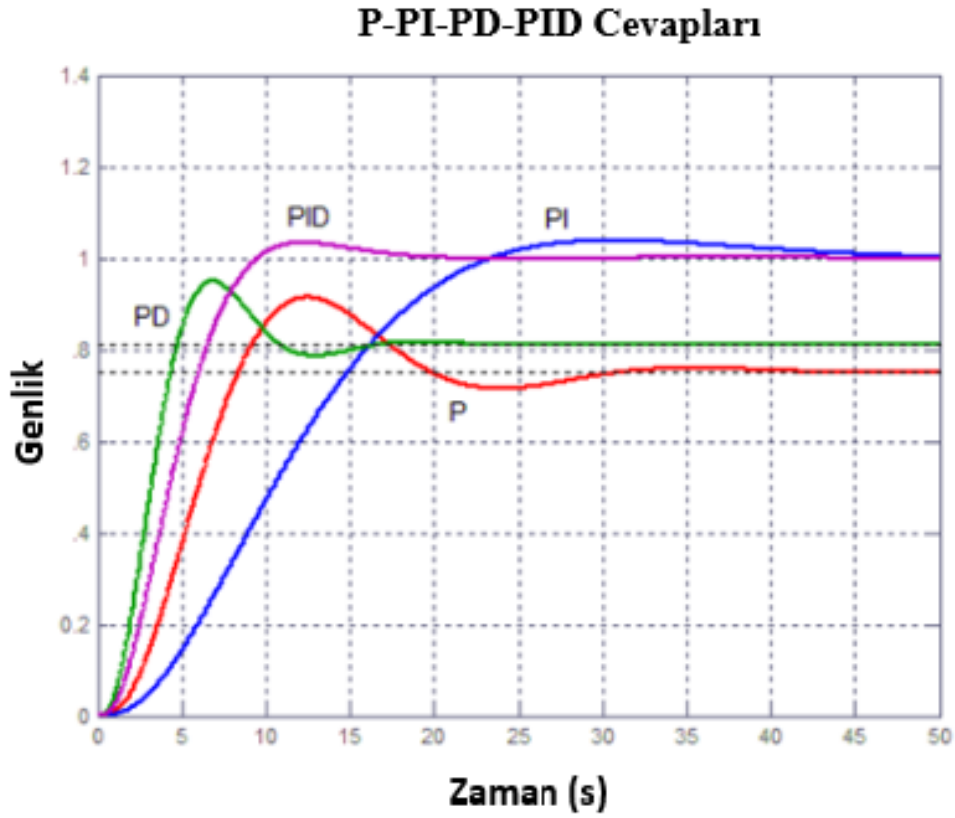
Oransal İntegral Türevsel denetleyici, etkisi altında olan sistemin cevabı, PI ve PD denetleyicilerin olumlu etkilerini birlikte gösterebildiği denetleyicidir. Sistemi hem hızlandırır, hem de daha kararlı hale getirir.

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (4.6)$$

Aynı sistem için yukarıdaki Çizelge.1 irdelendiği zaman (K_p , K_i , K_d) nin sistem üzerindeki etkileri aşağıdaki Şekil.7’de görsel olarak gözlemlenebilir.

Çizelge.4.1 Parametrelerin sistem üzerindeki etkileri

Kapalı Çevrim Cevabı	Yükselme Zamanı	Aşma Değeri	Sürekli Rejim Zamanı	Sürekli Rejim Hatası
K_p	Azalır	Artar	Küçük Değişim	Azalır
K_i	Azalır	Artar	Artar	Kalmaz
K_d	Küçük Değişim	Azalır	Azalır	Küçük Değişim



Şekil 4.3. Kontrolcü türüne göre sistem cevapları

4.2. Kayan Kipli Kontrol

Gelişen teknoloji ile birlikte birçok mühendislik ve fen bilimleri alanlarında yapılan çalışmalar farklı nedenlerden otomatik kontrolü ihtiyaç olarak görmüştür. Bu ihtiyaçlar;

Düzenleme, sabit çalışma noktasını bozucu ve belirsizliklerin olduğu durumda da koruyabilme

Dengeleme, sistemin dinamik yanıtını değiştirme (örneğin kararsız bir sistemi kararlı yapma)

Eniyileme, enerji zaman gibi değerler açısından en ekonomik gelişmeyi sağlama amaçları ile ortaya çıkabilir [44].

Bu bölümde, kayan kip kontrol sistemlerinin tanımı, özellikleri, tasarım kriterleri incelenmiş ve kayan kip kontrolün temel yapısı anlatılmıştır.

4.3. Kayan Kipli Kontrolcülere Giriş

Kayan kipli kontrolcü, uygun koşullar yerine getirildiğinde, sistemde üzerindeki belirsizliklere, parametre değişmelerine ve bozuculara rağmen istenilen dinamik davranışı gösterebilen son derece gürbüz bir kontrolcü tekniğidir. 1950’li yıllarda ilk kez Emelyanov aracılığı ile öne sürülen bu teknik, sürekli gelişim halinde olan hızlı anahtarlama teknolojisi ve 1977 yılında da ilk defa Vadim Utkin’in İngilizce olarak yayınlanan çalışması ile literatürde tanıtılmış, bugüne kadar da gün geçtikçe artan bir ivme ile kullanım alanı genişlemeye devam etmiştir. Kayan kipli kontrolcünün Vadim Utkin tarafından ifadesi: “Kayan kipli kontrolcü, sürekli kontrol fonksiyonlarıyla meydana gelmiş uygun anahtarlama tekniği ile de desteklenmiş bir alt sistemler kümesinden meydana gelir.” Kontrolcü tekniğinde sistemin durum uzayında belli bir yüzey üzerinde süreksizliğe maruz kaldığı ifade edilir [45].

Kolay uygulanabilmesi ve lineer sistemlerde başarılı sonuçlar vermesinden dolayı PID (Proportional-Integral-Derivative) kontrol günümüzde birçok endüstri uygulamalarında hala tercih edilen yöntemlerinin başında gelmektedir. Kayan kipli kontrol ise PID kontrole bir alternatif olarak kullanım alanı genişleyen bir kontrolör yapısıdır. PID denetleyici, doğrusal olmayan sistemlerin kontrolünde istenilen başarılı bir etkiyi gösterememektedir. PID denetleyicinin bu gibi olumsuzluklarını yok edebilmek için gürbüz kontrol yöntemleri kullanılmaya başlanmıştır. Bu kontrol yöntem ise gürbüz kontrolör yapısına sahip olan KKK (Kayan Kipli Kontrol)’dür.

Kayan Kip Kontrolcü metodunda hedef lineer olmayan sistemin durum yörüngesini, durum uzayında kullanıcı tarafından tarif edilmiş bir yüzeye, diğer bir ifade ile kayma yüzeyine yaklaştırmak ve sistemin durum yörüngesinin bu kayma yüzeyinin üstünde bulunmasını amaçlamaktır. Bu bağlamda en kritik nokta aslında sistemin durumu kullanıcı tarafından ifade edilen yüzeye yönlendirip ve daha sonrasında ise yüzeyin üstünde bulunmasını sağlayacak bir giriş sinyalinin oluşturulmasıdır [46].

Kayan Kipli Kontrol, deęişken yapılı kontrol sistemlerinin yaklaşımı için spesifik bir durumdur. Bu yaklaşımdaki temel hedef, hata deęerini “anahtarlama yüzeyine (switching surface)” veya “kayma yüzeyine (sliding surface)” getirmek ve bu yüzeyde bırakmaktır. Sonrasında ise sistem “kayma rejiminde”dir ve modelleme hataları ve/veya dış bozuculardan etkilenmemektedir. Geleneksel Kayan Kipli Kontrol’un bilinen iki ana problemi bulunmaktadır. Bunlardan biri, çatırdama(chattering) adı verilen kontrolcü çıkışında gözlemlenen yüksek frekanslı salınımlardır. Dięeri ise eşdeęer kontrolün (equivalent control) belirlenmesindeki zorluklardır. Çünkü, eşdeęer kontrol ifadesi kontrol edilen sistemin tüm dinamiklerinin bilinmesini ve hesaplamaya dahil olmasını gerektirir. Bir çok çalışmada bu problemleri çözebilmek için birtakım metodlar öne sürülmüştür. En çok bilineni ise çatırdama yok etme metodu olan geleneksel Kayan Kipli Kontrol’de kullanılan işaret fonksiyonu (sign) yerine yumuşak geçişli doyma fonksiyonu (saturation) kullanılmasıdır. Eşdeęer kontrolü hesaplamada ki zorluk en küçük kareler metodu ile kestirim veya ardışıl en küçük kareler metodu ile kestirim teknikleri ile giderilmeye çalışılmıştır [47].

Kayma kipinin önemli özellikleri şöyle sıralanabilir:

1) Kayma kipi, deęişken yapılı sisteme ait verilen yapılar dışında yeni bir yörüngede oluşabilir. Bir başka ifadeyle, kayma kipi sistem yörüngelerinden bağımsızdır.

2) Kayma kipi yörüngesi sisteminin derecesinden daha düşük boyutlu kayma yüzeyi ile sağlanmaktadır. Bu ise basitleştirme ve bağlaşmış sistemleri birbirinden ayırma (decoupling) tasarım yordamlarına izin verir.

3) Herhangi bir sistem yörüngesi, bir kümenin içerisinde seçilen başlangıç koşulları için tüm geçmiş ve gelecek zamanda yine o küme içerisinde kalıyor ise o kümeye deęişimsiz (invariant) küme denir. Eğer deęişken yapılı kontrol sistemine ait kayma kipi sarsım ve dış bozuculardan etkilenmiyorsa, o halde kayma kipi sarsım ve dış bozuculara deęişimsizdir denir. Açıkçası, deęişimsizlik özellięi gürbüzlük, uyarlanırlık özelliklerinden daha güçlü bir özelliktir. Çünkü dięerlerinde beklenmedik deęişiklikler karşısında sistemin deęişimsiz olması deęil, kontrol edilebilir ve beklenir bir davranış göstermesi amaçlanır. Geleneksel kayma kipi sırasında sistem dinamiklerini etkileyen tek parametre hata vektörü çarpanıdır. Bu yüzden sistem dinamikleri deęişimsizdir. Bu özellik kayma kipli kontrolörlerin en önemli

özelliğidir.

4) Orijin noktası sistemin denge noktasına karşı düşmektedir. Bu yüzden denge noktasına ulaşılan kadar, kayma kipi sırasındaki sistem davranışı, sisteme ait geçici hal davranışının bir parçasıdır.

5) Süreksiz kontrole ait giriş, sıfıra çok yakın olduğunda bile çıkışı sonlu değerler alır. Yani süreksiz kontrol işareti yüksek-teorik olarak sonsuz-kazançlı bir kontrolör çıkışı gibi düşünülebilir, bu ise geleneksel anlamda sistem davranışındaki belirsizlik ve bozucuları bastırma özelliğini açıklamaktadır. Sürekli kontrol sistemlerinden farklı olarak, değişimsizlik sonlu kontrol işareti ile elde edilmektedir [48].

4.4. Kayan Kipli Kontrol Prensipleri

Kayan Kipli Kontrolü (KKK), kayda değer doğruluk, sağlamlık, kolay ayar ve uygulama özelliklerine sahip doğrusal olmayan bir kontrol tekniğidir.

KKK sistemler, sistem durumlarını, kayan yüzey olarak adlandırılan durum alanındaki belirli bir yüzeye sürmek için tasarlanmıştır. Kayma yüzeyine ulaşıldığında, kayma kipi kontrolü, durumları kayma yüzeyinin yakın komşuluklarında tutar. Bu nedenle, kayan kip kontrolü, iki parçalı bir kontrol tasarımıdır. İlk bölüm, kayan bir yüzeyin tasarımını içerir, böylece kayma hareketi tasarım özelliklerine uygundur. İkincisi, bir kontrol yasasının seçilmesiyle ilgilidir bu anahtarlama yüzeyini sistem durumuna cazip hale getirecektir [49].

Kayma kipi kontrolünün iki ana avantajı vardır. Birincisi, sistemin dinamik davranışı kayan fonksiyonun özel seçimi ile uyarlanabilir. İkincisi, kapalı döngü cevabı bazı belirsizliklere karşı tamamen duyarsız hale gelir. Bu prensip, sınırlandırılmış olan parametre belirsizliklerini, bozulmayı ve doğrusallığı modellemeye yöneliktir.

Pratik bir bakış açısıyla KKK dış bozuculara ve modelin ağırlık parametrelerinin belirsizliklerine maruz kalan lineer olmayan işlemlerin kontrol edilmesini sağlar [49-51].

Kayma kipli kontrolör; kayma yüzeyinin her iki tarafında farklı yapılara sahip olan, gürbüz kontrol davranışına ulaşmayı amaçlayan, doğrusal olmayan bir kontrolördür. Bu özelliklere sahip bir kayma kipli kontrolör genel olarak ele alındığında üç temel bileşenden oluşur.

Bunlar:

- Kayma yüzeyinin belirlenmesi,
- Ulaşma koşulu ölçütünün seçilmesi,
- Kontrol kuralının elde edilmesi.

Elde edilen kayma yüzeyi ile; kayma kipi sırasında kararlılık, belirli başarımlı ölçütlerinin iyileştirilmesi, derece düşürme gibi tasarım amaçlarının yerine getirilmesi beklenir [48].

Doğrusal olmayan SISO sistemini göz önünde bulunduralım;

$$\dot{x} = f(x, t) + g(x, t)u \quad (4.7)$$

$$y = h(x, t) \quad (4.8)$$

y ve u skalar çıktısını ve girdi değişkenini belirtir ve $x \in R^n$ durum vektörünü belirtir.

Kontrol amacı, çıktı değişkeninin (y) istenen bir profili izlemesini sağlamaktır yani çıktı hatasının değişken olması gerekir. $e = y - r$ kabul edilebilir bir sürenin geçmesinden sonra sıfırın küçük bir çevresine değişim gösterir.

Belirtildiği gibi, KKK sentezi iki fazı gerektirir:

Faz 1 : “Kayan Yüzey Tasarımı”

Faz 2 : ”Kontrol Girişi Tasarımı”

İlk aşama, sistem durumunun belirli bir skalar fonksiyonunun tanımıdır, $\sigma(x) = R^n \rightarrow R$.

Çoğu zaman, kayma yüzeyi belirli bir sayıda türev ile birlikte $e_y(t)$ izleme hatasına bağlıdır.

$$\sigma = \sigma(e, \dot{e} \dots e^{(k)}) \quad (4.9)$$

σ fonksiyonu kaybolacak şekilde seçilmelidir, $\sigma = 0$, $e_y(t)$ sonunda herhangi bir çözümü sıfıra yönelecek olan “kararlı” bir diferansiyel denklem ortaya çıkarır.

Kayan yüzey için en tipik seçenek aşağıdaki tipte lineer bir kombinasyondur:

$$\sigma = \dot{e} + c_0 e \quad (4.10)$$

$$\sigma = \ddot{e} + c_1 \dot{e} + c_0 e \quad (4.11)$$

$$\sigma = e^{(k)} + \sum_{i=0}^{k-1} c_i e^{(i)} \quad (4.12)$$

Doğru seçilmiş c_i katsayıları ile, σ değişkeni sıfıra yönlendirilirse, hatanın üstel kaybolması ve türevleri sağlanır. Eğer böyle bir özellik varsa, o zaman kontrol görevi, σ 'un sonlu zamana sıfırlanmasını, diğer hususları “unutmasını” sağlamaktır.

Geometrik açıdan $\sigma = 0$ denklemi hata alanındaki “kayma yüzeyi” olarak adlandırılan bir yüzeyi tanımlar. Kontrollü sistemin yörüngeleri, sistem davranışının tasarım özelliklerini karşıladığı kayan yüzeye zorlanır.

Kayma yüzeyi için tipik bir form şöyledir: sadece tek bir skalar p parametresine bağlıdır,

$$\sigma = \left(\frac{d}{dt} + p\right)^n e \quad (4.13)$$

$$n = 1 \text{ için} \quad \sigma = \dot{e} + p e$$

$$n = 2 \text{ için} \quad \sigma = \ddot{e} + 2p\dot{e} + p^2 e$$

Pozitif parametrenin p seçimi neredeyse isteğe bağlıdır ve kayma halindeyken sistemin “azalmış dinamikleri” nin benzersiz kutuplarını tanımlar.

Tamsayı parametresi k , aksine oldukça kritiktir, $r - 1$ 'e eşit olmalıdır, r, y ve u arasındaki bağıl derecedir. Bu, σ değişkeninin göreceli derecesinin bir olduğu anlamına gelir.

Ardışık faz, sistem yörüngelerini kayan yüzey üzerine yönlendiren bir kontrol eylemi bulmaktır, yani kontrol, sonlu bir zamanda σ değişkenini sıfıra yönlendirebilmektedir.

Kayma kipi kontrol yaklaşımına dayanan birkaç yaklaşım vardır:

- Standart (veya birinci dereceden) kayan kip kontrolü
- Yüksek dereceli kayma kipi kontrolü

Tüm kayma kipe dayalı kontrol tekniklerin ortak özelliği, orijinal sistem dinamikleri hakkında kesin bir bilgi talep edilmemesi, kontrol edilen sistem tamamen belirsiz bir "kara kutu" nesnesi olarak ele alınmasıdır [52].

4.4.1. Birinci derece kayan kipli kontrol

Kontrol yüzey boyunca süresizdir $\sigma = 0$;

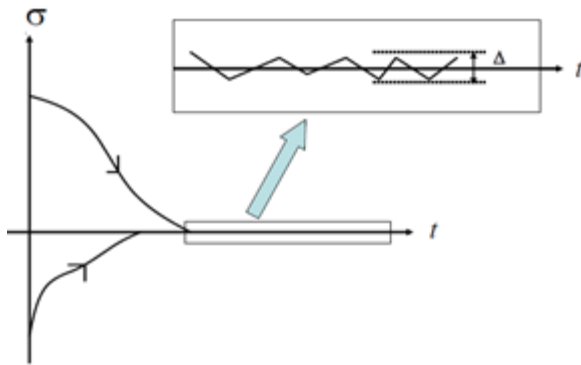
$$u = -U \operatorname{sgn}(\sigma) \quad (4.14)$$

ise,

$$u = \begin{cases} -U & \sigma > 0 \\ U & \sigma < 0 \end{cases} \quad (4.15)$$

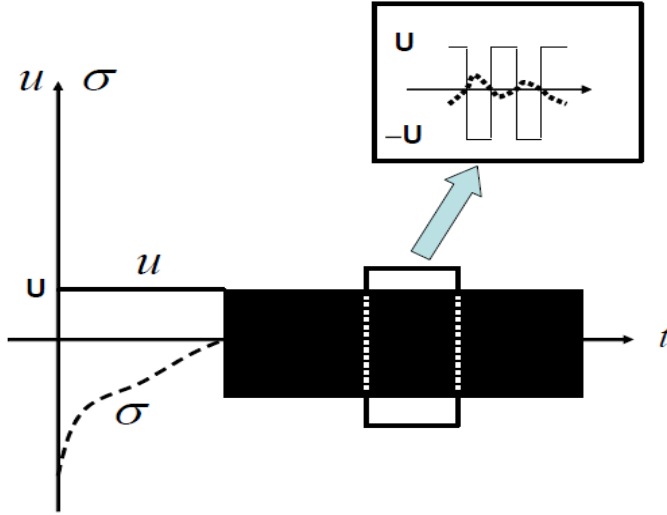
U , yeterince büyük bir pozitif sabittir.

Kararlı durumda, u kontrol değişkeni değerler arasında çok yüksek frekansta hareket edecektir. (teorik olarak sonsuz) $u = -U$ ve $u = U$ Şekil.4.4'te gösterilmiştir.



Şekil 4.4. Farklı başlangıç koşullarından başlayarak σ değişkeninin tipik evrimi

Kesintisiz yüksek frekanslı anahtarlama kontrolü (Şekil. 4.5) “elektrik” uygulamalarında (PWM kontrol sinyallerinin normalde kullanıldığı yerlerde) uygundur ancak mekanik sistemlerin kontrolü gibi farklı alanlarda birçok soruna yol açar.



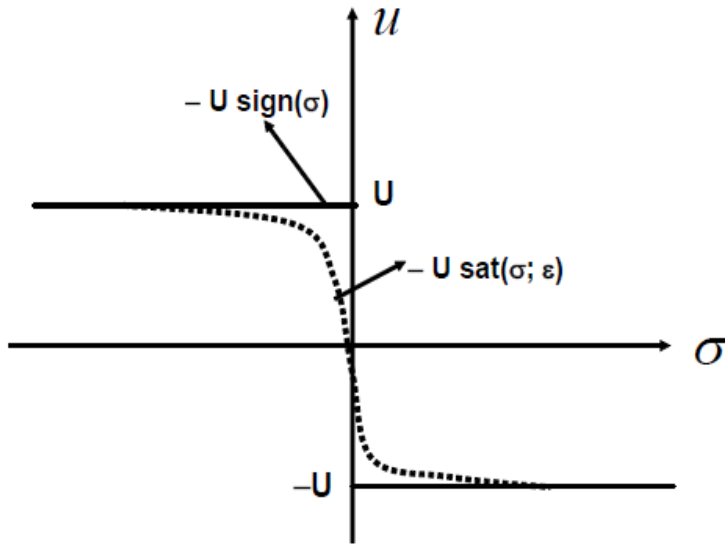
Şekil 4.5. Farklı Kontrol sinyalinin tipik evrimi (kesikli çizgi σ 'yi temsil eder) [47]

Öne sürülen anahtarlama kontrol kuramı, süreksiz işaret fonksiyonu ($\text{sgn}()$) içermektedir. $\text{sgn}()$ fonksiyonu, gerçek sistemler için tehlikeli doğurabilecek boyutta bir kontrol sinyalinde çatırdama etkisi meydana getirebilir. Bu sorunun çözümü için, daha önceki çalışmalarda $\text{sgn}()$ yerine tercih edilebilen fonksiyonlar $\tanh$, aşım ve sigmoid fonksiyonlarıdır [3].

Çatırdama problemini çözmek için, süreksiz işaret fonksiyonu yerine sürekli ve düzgün bir yaklaşımla (yumuşatılmış) Şekil 10'daki gibi karakteristiğe sahip aşağıdaki fonksiyonlar önerilmiştir:

$$SAT \quad u = -U \text{sat}(\sigma, \varepsilon) \equiv -U \frac{\sigma}{|\sigma| + \varepsilon} \quad \varepsilon > 0 \quad \varepsilon \approx 0 \quad (4.16)$$

$$TANH \quad u = -U \tanh(\sigma/\varepsilon) \quad \varepsilon > 0 \quad \varepsilon \approx 0 \quad (4.17)$$



Şekil 4.6. Kayma kipi kontrolünün yumuşak yaklaşımları [47]

4.4.2. Yüksek dereceden kayan kipli kontrol

Yüksek dereceden kayan kipli kontrol, klasik kayan kipli kontrol yaklaşımının kayma yüzeyinin zamana göre yüksek dereceden türevi üzerinde genelleştirilmiş halidir. Bu kontrol yaklaşımında orijinal kayan kipli kontrol yaklaşımın sağladığı avantajlar korunurken aynı zamanda oluşan çatırdama etkisini ortadan kaldırılabilir. Ayrıca çatırdamadan kaçınırken meydana gelebilecek hassasiyet kaybının da önüne geçilebilir.

Kayma derecesi, kayma yüzeyi $\sigma(x) = 0$ üzerindeki hareket kısıtları ile ilişkili dinamiklerin düzgünlük derecesini belirler.

Tanım 1: Sınırlı kayma yüzeyinin $\sigma = 0$ 'ın tam olarak üzerinde olan her harekete ideal kayma denir.

Tanım 2: Sınırlı kayma yüzeyi $\sigma = 0$ 'ın yakın komşuluğunda meydana gelen her harekete gerçek kayma denir.

Tanım 3: Kayma derecesi r , toplam sürekli türev sayısı olarak tanımlanabilir. r 'inci dereceden kayma seti, kayma yüzeyi ile aşağıdaki eşitlik yardımı ile ilişkilendirilebilir.

$$\sigma = \dot{\sigma} = \ddot{\sigma} = \dots = \sigma^{(r-1)} = 0 \quad (4.18)$$

Yukarıdaki şartı sağlayan hareket, $\sigma(t, x) = 0$ 'a göre r 'inci dereceden kayan kip olarak adlandırılır.

Yüksek dereceden kayan kipli kontrol uygulamalarında temel sorun bilgi talebinin artmasıdır. Yani r 'inci dereceden kayan kipli kontrolcü için kayma yüzeyinin $(r - 1)$ 'inci dereceye kadar zamana göre türevlerine ihtiyaç vardır. Ancak bazı ikinci dereceden kayan kipli kontrol algoritmaları bu ihtiyacı azaltan nitelikte oldukları için uygulamalarda bu tip ikinci dereceden kayan kipli kontrolcüler tercih edilmektedir [53].

4.4.3. İkinci dereceden kayan kipli kontrol

İkinci dereceden kayan kipli kontrol algoritmaları, sağlamlık özelliklerinden de ödün vermeden çatırdama konularını tamamen çözen güçlü bir alternatiftir [54,55].

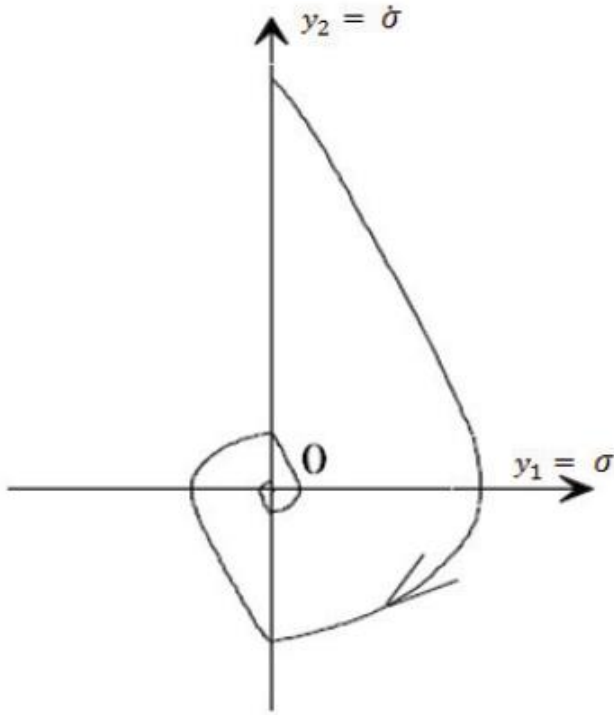
İkinci dereceden kayan kipli kontrol, yüksek dereceden kayan kipli kontrolcüler arasında daha basit ve anlaşılabilir olması sebebiyle en çok kullanılanıdır. Özellikle burulma kontrolcü ve süper burulma kontrolcü gibi algoritmalar tercih edilmektedir.

İkinci dereceden kayan kipli kontrolde, klasik kayan kipli kontrolden farklı olarak kontrol çabası, kayma yüzeyinin ikinci türevi σ'' 'in işareti ve büyüklüğü ile doğrudan ilişkilidir.

İkinci dereceden kayan kipli kontrol yaklaşımında kullanılan “Burulma Algoritması” ve “Süper Burulma Algoritması” aşağıdaki bölümlerde özetlenmektedir.

4.4.4. Burulma Algoritması

Burulma algoritmasına göre tasarlanan kontrolcü, ikinci dereceden kayan kipli kontrolcü algoritmalarında en çok kullanılanıdır. Bu kontrolcünde sistem $\sigma = 0$ durum uzayında orijin etrafında helezonlar çizer (Şekil.4.7). Yörüngeler sonlu zamanda orijine yakınsarken sonsuz sayıda döngü gerçekleştirirler.

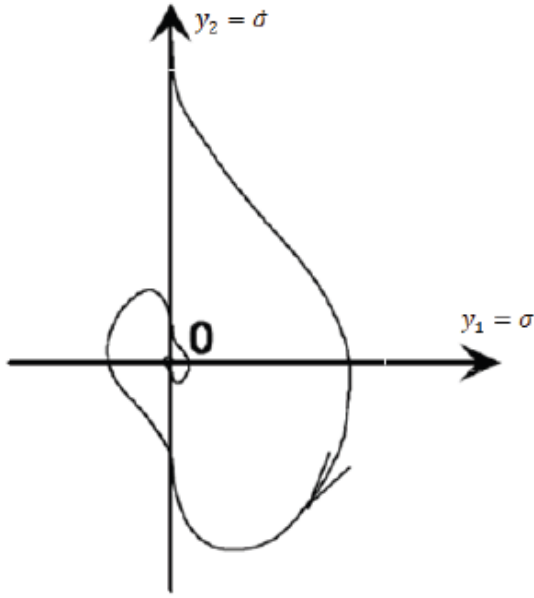


Şekil 4.7. Burulma algoritması faz düzlemi yörüngeleri

4.4.5. Süper burulma algoritması (Super twisting algorithm)

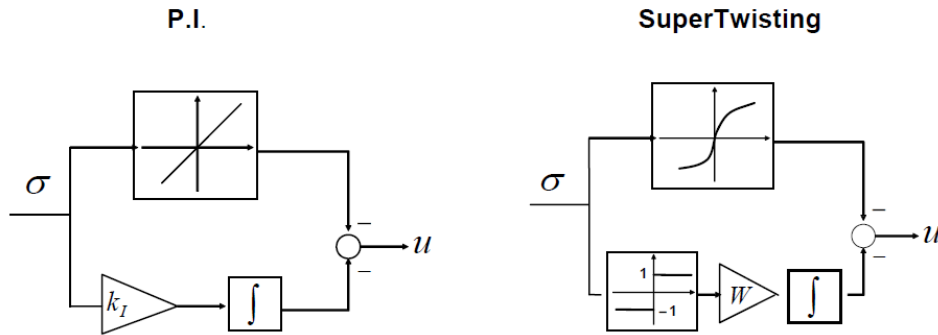
Bu algoritma bağıl derecesi bir olan sistemler için çatırdamayı önlemek amacı ile geliştirilmiştir. Bu kontrolcüde de burulma algoritmasında olduğu gibi yörüngeler orijin etrafında helezonlar çizerek yakınsar. Ancak sürekli olan kontrol terimi iki parçaya bölünmüştür. Bunlardan biri zamana göre türevli süreksiz bir fonksiyon ile tanımlanırken, diğeri kayma yüzeyinin sürekli bir fonksiyonudur.

Süper burulma algoritmasına göre tasarlanan kontrolcüde kayma yüzeyi ve kayma yüzeyinin türevinin oluşturduğu yörüngelerin $\sigma \approx 0$ durum uzayında davranışı burulma kontrolcüdeki yörünge davranışına benzerlik gösterir (Şekil 4.8).



Şekil 4.8. Süper burulma algoritması faz düzlemi yörüngeleri

Süper büküm algoritması, klasik PI kontrol cihazının doğrusal olmayan bir versiyonu olarak görülebilir. Bu, Şekil 4.9'a bakılarak daha net görülebilir.



Şekil 4.9. PI (solda) ve Süper Bükümlü (sağda) kontrol cihazlarının blok düzeni [52].

4.5. Çeşitli Kayan Kipli Kontrol Uygulamaları

Literatüre bakıldığında kayan kipli kontrolcü, robotik uygulamalara, motor sürücülerine, AC/DC çeviricilerine, nükleer enerji reaktörlerine, havacılık ve motor ile çalışabilen araçlara, DC/DC çeviricilerine, rüzgar-enerji sistemlerine, kimyasal süreçlere, hard disk gibi elektrik veya elektromekanik sistemlere uygulanmıştır [5].

Furat ve diğerleri, Araştırma konusu, geleneksel metodla kontrol sinyalinin elde edilmesine dayalı ikinci dereceden kayan kipli kontrolcü metodu geliştirilmiş, sistemdeki bilinmeyenlere karşı daha efektif kontrol sinyali elde edebilmek üzere kontrol sinyaline yeni bir parametre geliştirilmiştir. Kararlı olduğu teorik olarak ispat edilebilen metod, gerçek bir elektromekanik sisteme tatbik edilmiştir. Daha önce ki çalışmalarda önerilen başka bir ikinci derece metod da aynı sisteme uygulanmış ve başarımları kıyaslanmıştır. Elde edilen sonuçların geçici durumu ve kararlı durum cevapları grafiksel olarak verilmiş ve farklı başarımlar indeksleri ile ölçülmüştür [3].

Susperregui ve diğerleri, Doğru Akım (DA) servo motorun açısal referans takibi için birbirinden farklı iki tane birinci dereceden kayan kipli kontrolcü yaklaşımı çalışması yapmışlardır. Birincisi Slotine'in genel yaklaşımı ve ikincisi ise Utkin'in yaklaşımlarıdır. Kontrol için tam dereceli Luenberger gözlemleyici modellemişlerdir. Benzetim ve deneysel sonuçlarla her iki yaklaşımın da uygunluğu ve parametre değişimlerinde ki ilgisizliği belirlemeye çalışmışlardır [56].

Eker, Elektromekanik bir sistem için hız kontrollü PID kayma yüzeyi kullanma sureti ile çalışmalar yapmıştır. Kayan kipli kontrolcü'nün çatırdama efektini giderebilmek için hiperbolik fonksiyon tercih etmiştir. Deneysel çalışmanın sonuçları PID kontrol ile karşılaştırılmış ve deneysel sonuçlar, önerilen kayan kipli kontrolcünün bozuculara ve belirsizliklere rağmen gürbüz olduğu ispat edilmiştir [57].

Brandtstädter ve diğerleri DA motor tahrikli ters sarkaç sistemine, elektromekanik sistemler için öne sürdükleri kayan kipli kontrolcüyü tatbik etmişlerdir. Enerji tüketimini düşüren, gürbüzlük sağlayan, çalışmada basitlik sağlayan ve daha iyi performans gösteren yeni kontrol yaklaşımı öne sürülmüştür. Öne sürülen bu yaklaşımın performansı matematiksel benzetim ve deneyler ile doğrulanmıştır [58].

Taedong ve diğerleri Lineer pozisyon kontrolünde, kök yerleştirme ile belirlenen kayma yüzeyi ile kontrolcünün performansı deneysel olarak analiz edilmiştir. Yüksek kontrol hareketini elimine etmek için işaret fonksiyonu yerine sigmoid-benzeri fonksiyon kullanılmıştır [59].

Nafa ve diğerleri Dolaylı tahrik sistemler için düzenlenmiş kayan kipli kontrolcü öne

sürmüşlerdir. 3 serbestlik derecesi olan dolaylı tahrikli mekanik sistemi kontrol etmeye çalışmışlardır. Bu sistem 3 çeşit alt sistem barındırmaktadır ve her bir sistem için bir yüzey öne sürülmüştür. Yaptıkları arabalı çift ters sarkaç benzetimi, öne sürülen yöntemin uygunluğunu göstermiştir [60].

Rao ve diğerleri, Dönel bir ters sarkaç sistemi için, DA motor dinamiğini de modelleyip, kontrol sinyalini motorun akımı olarak seçip kayan kipli kontrolcü uygulamasını yapmışlardır. Lineer olmayan bu sistemin davranışı, doyma fonksiyonu ile kontrol edilmiş ve bu şekilde çalışmışlardır. Benzetim ile kontrol sinyalinin akım olarak seçildiği durumda çıkışların performans etkilerini gözlemlemişlerdir [61].

Shin ve diğerleri Araştırmalarında, belirsizlikler içeren bir dizi dolaylı tahrikli sistem için bulanık mantık ile uyarlamalı ayarlama yapılabilen ayrıştırılmış kayan kipli kontrolcü tasarlamışlar. Bu metod ile sistem alt sistemlere ayrıştırılmış ve her bir alt sistemin çeşitli hedefleri olan kayma yüzeylerini ifade etmişleridir. Benzetim sonuçları ile öne sürdükleri bu yaklaşımın doğruluğunu gözlemlemişlerdir [62].

5. MATERYAL VE YÖNTEM

Bu tez çalışması için simülasyon ortamında PID ve Kayan Kipli Kontrolcü metodları kullanılarak sabit mıknatıslı bir DA motorunun hız kontrolü için benzetim çalışmaları yapılmış ve Kayan Kipli Kontrolcünün parametreleri YAKA metodu ile belirlenmiştir. YAKA için literatürde birçok çalışma bulunmaktadır. Genel olarak, bu çalışmalarda kontrolcü parametrelerinin optimizasyonu bir uygunluk fonksiyonu ile yapılırken bu tez çalışmasında ele alınan kontrolcülerin her birinin hem çıkışa olan etkisi hem de kontrol sinyaline yaptığı katkı göz önünde bulundurularak farklı amaç fonksiyonları tanımlanmış, optimizasyonda daha iyi sonuçlara ulaşılmıştır. Literatürden seçilen bir elektromekanik sistem modeli üzerinde yapılan simülasyonlar hem grafiksel hem de istatistiksel olarak değerlendirilmiştir. Elde edilen sonuçlar, tek uygunluk fonksiyonu seçimine göre her bir parametreye özgü farklı amaç fonksiyonları seçiminin YAKA için en iyi değerleri bulmada daha etkin olduğunu göstermiştir.

5.1. Materyal

Bir simülasyon yazılımında algoritma ve seçilen elektromekanik sistem [5] kurgulanarak belirli farklı ölçüm parametreleri ile sonuçlar elde edilmiştir.

2. derece modellenen elektromekanik sistem modeli [5]:

$$G(s) = \frac{1694.58}{s^2 + 292.61s + 1970.4} \quad (5.1)$$

Bu modelin kapalı çevrimde PID ve birinci derece KKK ile kontrol edilebilmesi için optimum kontrol parametreleri önerilen yapıdaki YAKA ile aranmıştır.

5.2. Yöntem

5.2.1. PID optimizasyonu

Bu YAKA tabanlı PID optimizasyonunda başlangıç parametrelerinin rastgele oluşturulması için önceden belirlenmiş PID parametrelerinin alt ve üst limitleri aşağıdaki Çizelge 5.1’de verilmiştir.

Çizelge 5.1. PID parametrelerinin limit değerleri

Parametre	Alt Limit	Üst Limit
K_p	1	40
K_i	1	25
K_d	0.02	2.5

Önerilen YAKA algoritmasının PID için kullanılmasının blok diyagramı Şekil.5.1’de verilmiştir. Şekil.5.1 dikkatli şekilde incelendiğinde görülecektir ki hem geleneksel YAKA tabanlı PID’nin hem de önerilen YAKA tabanlı PID’nin başlangıçta oluşturulan rastgele pozisyonları aynıdır. Bunun aynı olmasının bize avantajı ise karşılaştırmanın daha doğru yapılabilmesini sağlamıştır.

Önerilen 3-kanal uygunluk fonksiyon tabanlı YAKA’da PID parametreleri belirlenirken kullanılan uygunluk fonksiyonları üç parametre içinde farklıdır. Bu fonksiyonlar Çizelge 5.2’de verilmiştir.

Çizelge 5.2. PID optimizasyonunda kullanılan uygunluk fonksiyonları

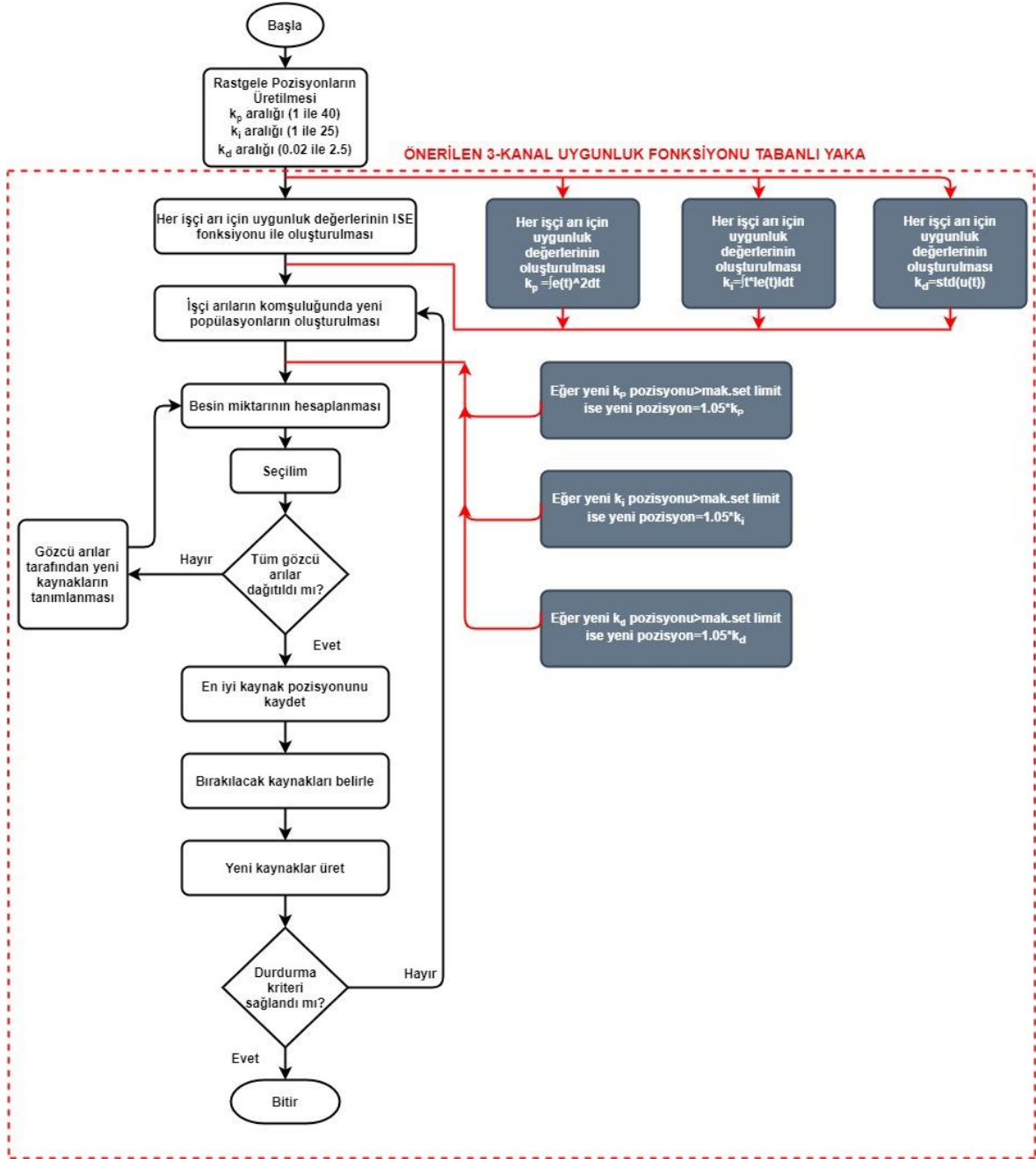
Parametre	Geleneksel	Önerilen
K_p	$10 * ISE + 3 * TS + OS$	$fp = 3 * ISE + ITAE + 5 * std(u(t))$
K_i		$fi = 3 * ITAE + ISE + 5 * std(u(t))$
K_d		$fd = 10 * fi + std(u(t))$

$$ISE = \int e((t))^2 dt \quad (5.2)$$

$$ITAE = \int t|e(t)|dt \quad (5.3)$$

$$std(u(t)) : \text{Kontrol sinyalinin standart sapması} \quad (5.4)$$

PID kontrol sinyali için Eş.4.1’de verilen kontrol kullanılmıştır. 50 iterasyon, 50 populasyon sayısı ile yapılan simülasyon ile elde edilen en iyi PID parametre değerleri Çizelge. 5.3’de verilmiştir.

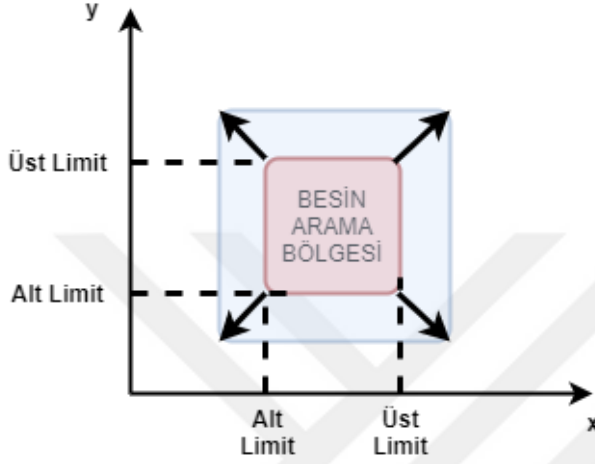


Şekil 5.1. Önerilen 3-kanal uygunluk fonksiyonu tabanlı YAKA

Çizelge. 5.3. PID simülasyon sonuçları

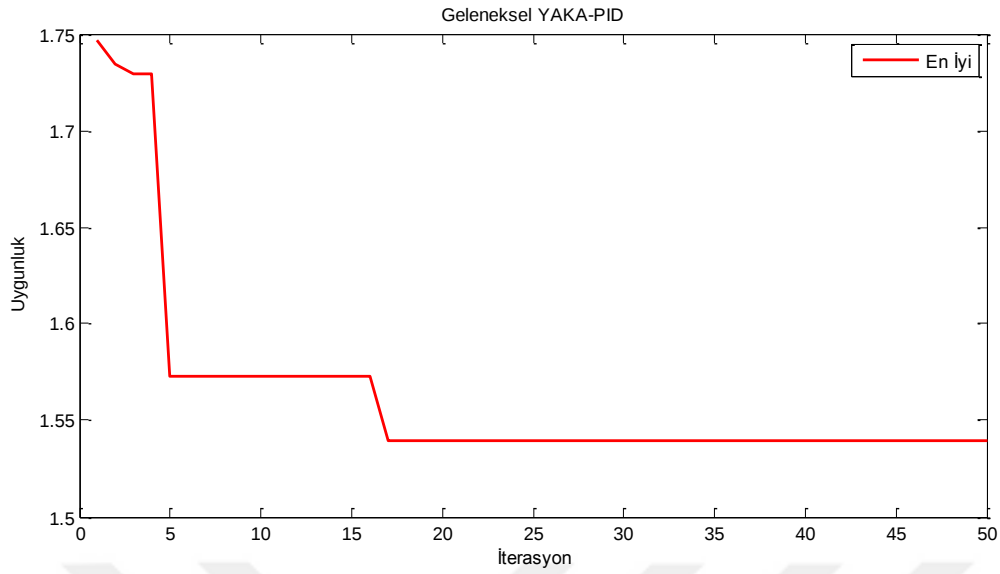
Özellik	K_p	K_i	K_d	ISE	ITAE	std(u)
Geleneksel	41.2058	6.8154	2.6144	4.1987	0.8119	158.9075
Önerilen	27.4532	15.4821	2.3753	1.6136	0.7360	138.4115

Burada K_p , K_i ve K_d ’nin daha önceden belirlenen üst limit değerlerinin dışına çıktığı görülmektedir. Bunun nedeni olan YAKA detaylı bir şekilde 3. bölümde anlatılmış ve orada ki Eş.3.2 ile yeni pozisyonlar üretilirken besin arama bölgesinin dışına çıkılabileceği rahatlıkla görülebilmektedir. Bu durum YAKA’nın bir karakteristiğidir. Şekil.5.2’de bu durum grafiksel olarak verilmeye çalışılmıştır.

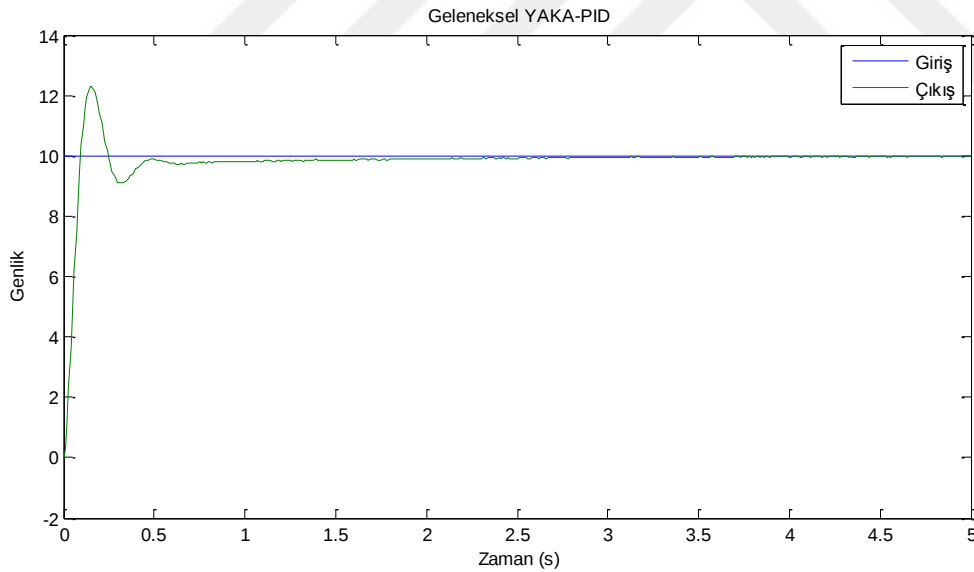


Şekil 5.2. Arıların besin arama bölgesi

Literatürde genellikle çok boyutlu optimizasyonlarda tek uygunluk fonksiyonu kullanılırken bu çalışmada ise Çizelge 5.2’de verildiği gibi her parametreye özel uygunluk fonksiyonları tanımlanarak daha etkin bir optimizasyon yapılmıştır. Şekil 5.3’de tek uygunluk fonksiyonunun 50 iterasyon sonucu değişimi görülmektedir. Bunun sonucunda elde edilen Çizelge 5.3’de verilen parametreler ile sistem çalıştırıldığında Şekil 5.4’deki çıkış elde edilmiştir. Sistem çıkışı incelendiğinde %20’den fazla aşım gerçekleştiği görülmektedir.

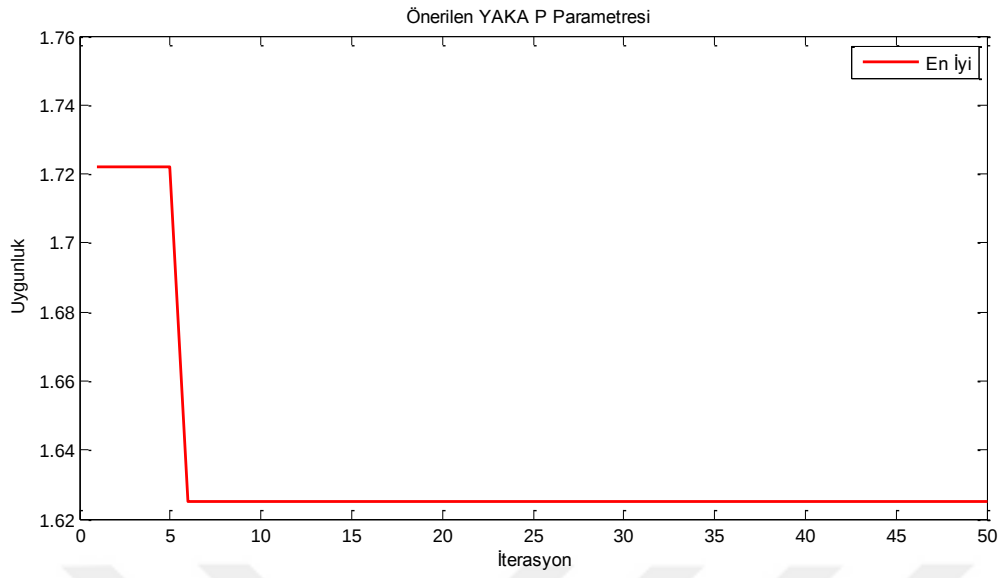


Şekil 5.3. Geleneksel YAKA'nın PID uygunluk grafiği

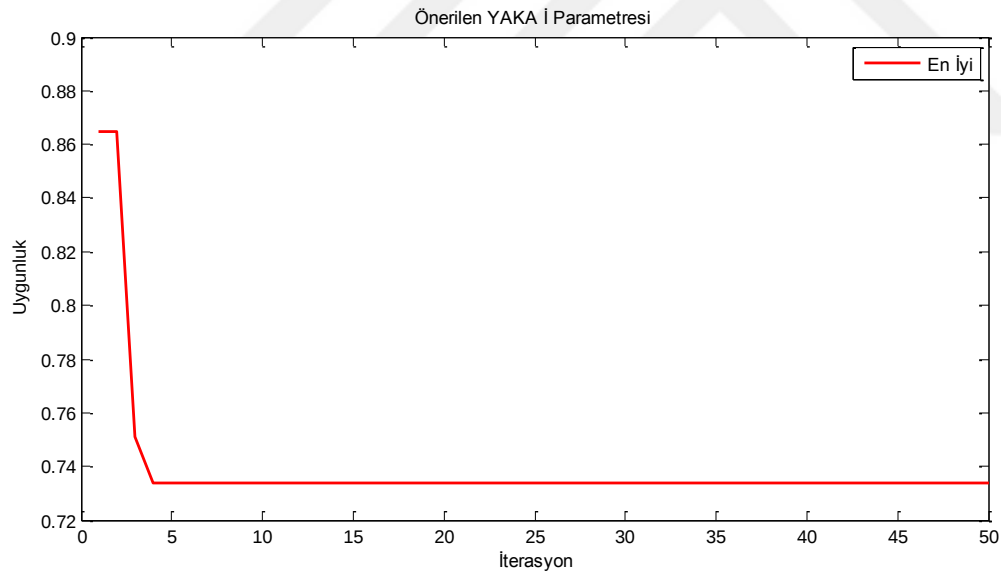


Şekil 5.4. Geleneksel YAKA'nın PID sonuç grafiği

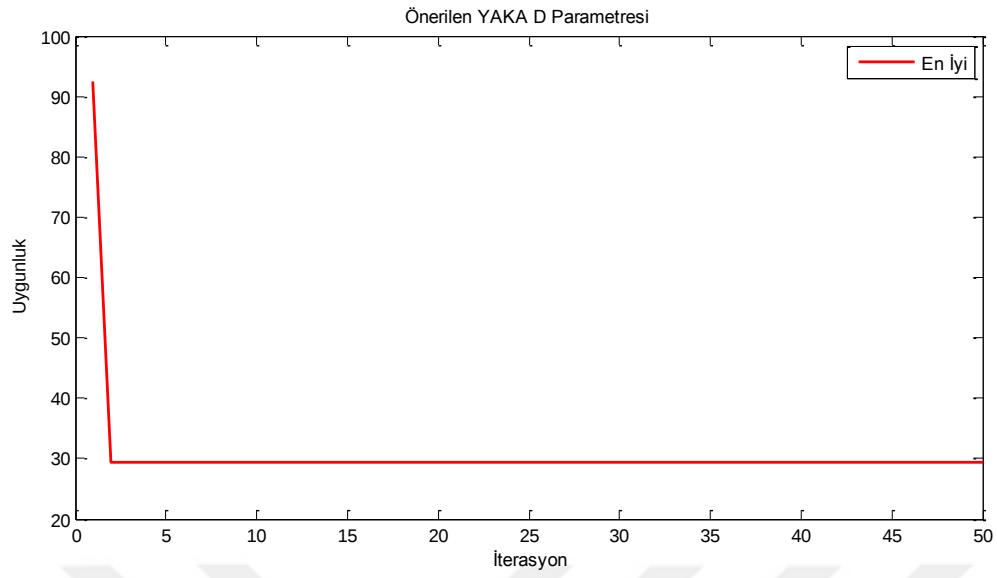
Bu çalışmada önerilen farklı uygunluk fonksiyonları ile yapılan optimizasyonda ise her bir parametrenin uygunluk fonksiyonundaki değişim Şekil 5.5, 5.6 ve 5.7'de verilmiştir. En fazla 6. iterasyonda son iyileşme gerçekleşmiştir. Elde edilen parametreler (Çizelge 5.3) PID kontrolcüye uygulandığında ise Şekil 5.8'deki aşırı çıkış gözlenmiştir. Tek uygunluk fonksiyonu ile yapılan optimizasyonda 17. iterasyona kadar iyileşme ile aşım görülürken, önerilen yöntem ile daha az iterasyon sayısı ile daha iyi sonuç elde edilmiştir.



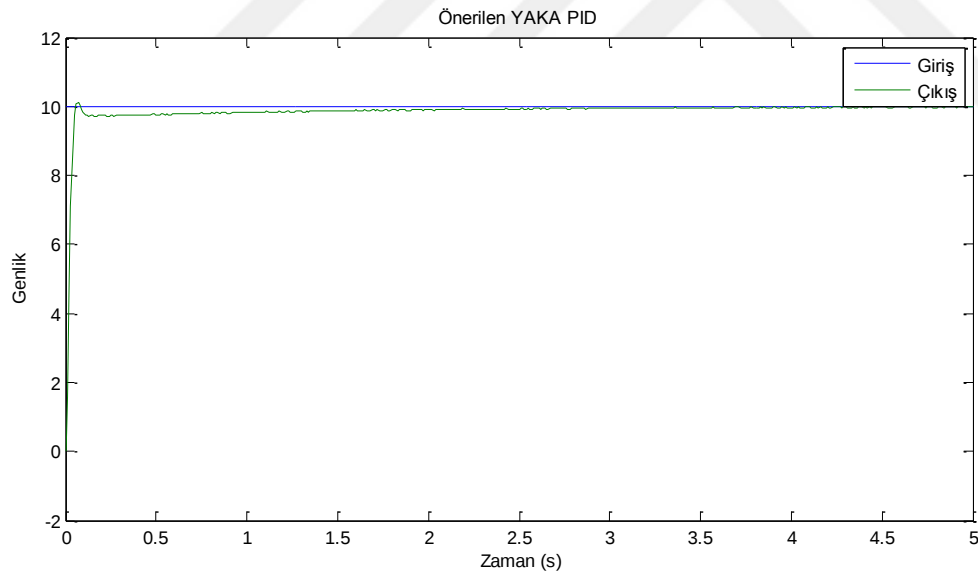
Şekil 5.5. Önerilen YAKA'nın P parametresi uygunluk grafiği



Şekil 5.6. Önerilen YAKA'nın I parametresi uygunluk grafiği



Şekil 5.7. Önerilen YAKA'nın D parametresi uygunluk grafiği



Şekil 5.8 Önerilen YAKA'nın PID sonuç grafiği

5.2.2. KKK optimizasyonu

Seçilen elektromekanik sistem için kayma fonksiyonu [5]:

$$\sigma(t) = \lambda e(t) + \beta \dot{e}(t) \quad (5.5)$$

Burada $e(t)$ hata, λ bağımsız parametre ve β adaptif parametredir.

İkinci derece bir sistemin açık çevrim transfer fonksiyonu:

$$G(s) = \frac{C}{s^2 + As + B} = \frac{Y(s)}{U(s)} \quad (5.6)$$

Buradan;

$$Y(s) = \frac{C}{s^2 + As + B} U(s)$$

$$(s^2 + As + B)Y(s) = CU(s)$$

$$s^2Y(s) + AsY(s) + BY(s) = CU(s)$$

$$s^2Y(s) = -AsY(s) - BY(s) + CU(s)$$

$$\ddot{y}(t) = -A\dot{y}(t) - By(t) + Cu(t) \quad (5.7)$$

elde edilir.

Hata; referans $r(t)$ ile çıkış $y(t)$ arasındaki fark olduğundan

$$e(t) = r(t) - y(t) \quad (5.8)$$

$$\ddot{e}(t) = \ddot{r}(t) - \ddot{y}(t) \quad (5.9)$$

$$\ddot{e}(t) = \ddot{r}(t) + A\dot{y}(t) + By(t) - Cu(t) \quad (5.10)$$

olarak bulunur. Hatanın ikinci türevinde kontrol sinyali görünmektedir. Buna göre, kayma fonksiyonunun türevi alındığında;

$$\dot{\sigma}(t) = \lambda \dot{e}(t) + \beta \ddot{e}(t) \quad (5.11)$$

Eş.5.10, Eş. 5.11'de yerine konulursa;

$$\dot{\sigma}(t) = \lambda \dot{e}(t) + \beta (\ddot{r}(t) + A\dot{y}(t) + By(t) - Cu(t)) \quad (5.12)$$

bulunur. Kayan Kipli Kontrol geleneksel çözümde, kontrol sinyali:

$$u(t) = u_e(t) + u_a(t) \quad (5.13)$$

Burada $u_e(t)$ eşdeğer kontrol sinyali, $u_a(t)$ ise anahtarlama kontrol sinyalidir.

Eşdeğer kontrol sinyali $u_e(t)$, Eş. 5.12 sıfıra eşitlenerek bulunur:

$$\begin{aligned} \dot{\sigma}(t) &= \lambda \dot{e}(t) + \beta(\ddot{r}(t) + A\dot{y}(t) + By(t) - Cu(t)) = 0 \\ u_e(t) &= \frac{1}{\beta C} [\lambda \dot{e}(t) + \beta(\ddot{r}(t) + A\dot{y}(t) + By(t))] \end{aligned} \quad (5.14)$$

İdeal kayan kipi için $u(t)$, $\dot{\sigma}(t)$ içinde yerine konulursa;

$$\dot{\sigma}(t) = \lambda \dot{e}(t) + \beta[(\ddot{r}(t) + A\dot{y}(t) + By(t) - C(u_e(t) + u_a(t)))] \quad (5.15)$$

Yukarıdaki eşitlikte $u_e(t)$ yerine konulursa;

$$\dot{\sigma}(t) = \lambda \dot{e}(t) + \beta[\ddot{r}(t) + A\dot{y}(t) + By(t)] - \beta Cu_a(t) - \beta C[\frac{1}{\beta C}(\lambda \dot{e}(t) + \beta(\ddot{r}(t) + A\dot{y}(t) + By(t)))]$$

Gerekli sadeleştirmeler yapıldığında ideal kayan kip:

$$\dot{\sigma}(t) = -\beta Cu_a(t) \quad (5.16)$$

Anahtarlama kontrol sinyali, sistemin kararlılığını sağlayacak şekilde seçilir. Bunun için Lyapunov kararlılık teoremi literatürde kullanılmıştır. Kararlılığı sağlamak için seçilen pozitif Lyapunov fonksiyonu:

$$V(t) = \frac{1}{2} \sigma^2(t) \quad (5.17)$$

Eş.5.17'nin türevi alındığında:

$$\dot{V}(t) = \sigma(t)\dot{\sigma}(t) \quad (5.18)$$

elde edilir. Kararlılığın sağlanması Eş. 5.18'i negatif yapacak $u_a(t)$ seçilmelidir.

Bunun için $\dot{\sigma}(t)$, Eş. 5.18'de yerine konulursa:

$$\dot{V}(t) = \sigma(t)(-\beta Cu_a(t)) \quad (5.19)$$

bulunur. Kararlılığı garanti etmek için uygun anahtarlama kuralı seçilirse:

$$u_a(t) = k_a \operatorname{sgn}(\sigma(t)) \quad (5.20)$$

$$\dot{V}(t) = -\beta C \sigma(t) k_a \operatorname{sgn}(\sigma(t))$$

$$\dot{V}(t) = -\beta C k_a \frac{|\sigma(t)|}{\sigma(t)}$$

$$\dot{V}(t) = -\beta C k_a (\dot{\sigma}(t)) < 0 \quad (5.21)$$

elde edilir. Çatırdama problemi oluşturan işaret fonksiyonu $\operatorname{sgn}(\sigma(t))$ yerine $\tanh(\sigma(t))$ kullanılırsa daha yumuşak bir anahtarlama kontrol sinyali elde edilir:

$$u_a(t) = k_a \tanh(\sigma(t)) \quad (5.22)$$

Böylece toplam KKK sinyali:

$$u(t) = \frac{1}{\beta C} [\lambda \dot{e}(t) + \beta (\ddot{r}(t) + A\dot{y}(t) + B y(t))] + k_a \tanh(\sigma(t)) \quad (5.23)$$

olarak bulunur.

Eş. 5.23'deki KKK sinyalindeki λ ve k_a bağımsız parametreleri YAKA ile aranacaktır. k_a doğrudan çatırdama ile ilgili olduğundan k_a için seçilen uygunluk fonksiyonu:

$$ISE + 2 * std(u(t)) \quad (5.24)$$

$$\text{Burada } ISE = \int (e(t))^2 dt$$

λ , daha çok sistemin kalıcı rejim çıkışında etkili olduğu için:

$$ITAE = \int t |e(t)| dt \quad (5.25)$$

seçilmiştir. Buna göre YAKA ile yapılan arama simülasyon parametreleri Çizelge.5.3'te verilmiştir.

Çizelge. 5.4. KKK parametre limit değerleri

Özellik	Alt Limit	Üst Limit
k_a	1	15
λ	0.1	10

Önerilen YAKA algoritmasının KKK için kullanılmasının blok diyagramı Şekil.23’de verilmiştir. Şekil 23 dikkatli şekilde incelendiğinde görülecektir ki hem geleneksel YAKA tabanlı KKK’nın hem de önerilen YAKA tabanlı KKK’nın başlangıçta oluşturulan rastgele pozisyonları aynıdır. Bunun aynı olmasının bize avantajı ise karşılaştırmanın daha doğru yapılabilmesini sağlamıştır.

Önerilen 2-kanal uygunluk fonksiyon tabanlı YAKA’da KKK parametreleri belirlenirken kullanılan uygunluk fonksiyonları iki parametre içinde farklıdır. Bu fonksiyonlar Çizelge 5.4’te verilmiştir.

Çizelge.5.5. KKK algoritmasında kullanılan uygunluk fonksiyonları

Özellik	Geleneksel	Önerilen
k_a	<i>ISE</i>	$ISE + 2 * std(u(t))$
λ	<i>ISE</i>	<i>ITAE</i>

50 iterasyon ve 50 populasyon ile yapılan simülasyon ile elde edilen en iyi KKK parametre değerleri Çizelge. 5.5’te verilmiştir.

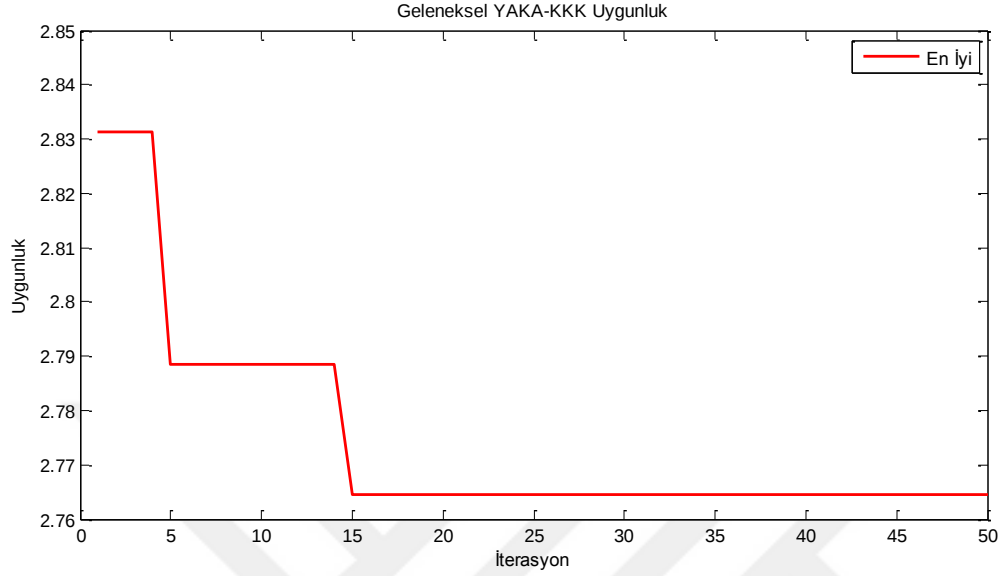
Çizelge.5.6. KKK simülasyon sonuçları

Özellik	k_a	λ	ISE	ITAE	std(u(t))
Geleneksel	15.1243	1.3813	2.7646	0.6751	5.1995
Önerilen	15.6802	0.3637	2.7258	0.3229	2.3857

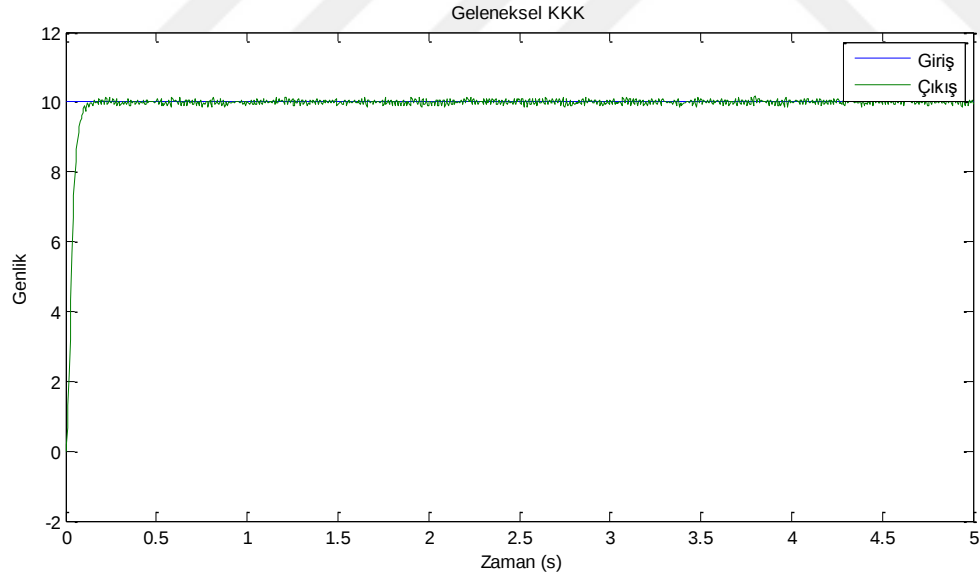
Burada k ’nın daha önceden belirlenen üst limit değerlerinin dışına çıktığı görülmektedir. Bunun nedeni olan YAKA detaylı bir şekilde 3. bölümde anlatılmış ve orada ki Eş.2 ile yeni pozisyonlar üretilirken besin arama bölgesinin dışına çıkabileceği rahatlıkla görülebilmektedir. Bu durum YAKA’nın bir karakteristiğidir. Şekil.15’de bu durum grafiksel olarak verilmeye çalışılmıştır.

Tek uygunluk fonksiyonu ile yapılan simülasyon çalışmasındaki uygunluk fonksiyonundaki

değişim Şekil 5.9’da, elde edilen parametrelerin KKK’de kullanılmasıyla elde edilen çıkış grafiği Şekil 5.10’da gösterilmiştir.

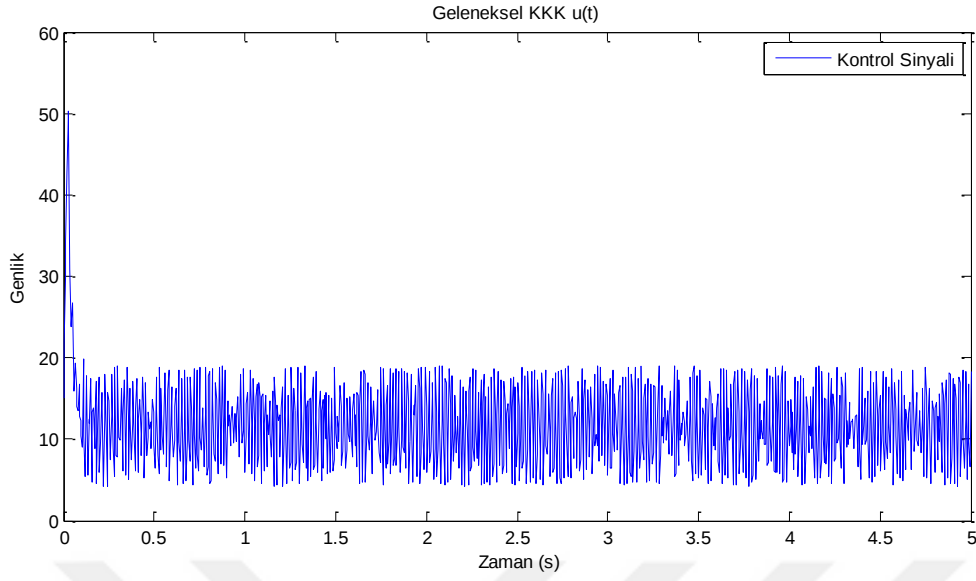


Şekil 5.9. Geleneksel YAKA ile KKK uygunluk grafiği

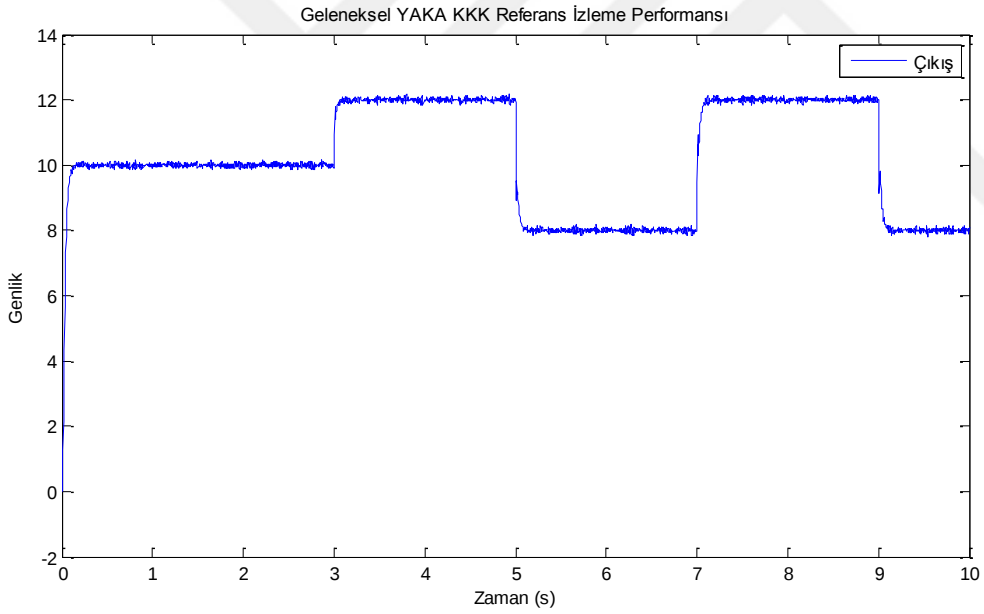


Şekil 5.10. Geleneksel YAKA ile KKK sonuç grafiği

Şekil 5.11’de görüldüğü gibi, sisteme uygulanan kontrol sinyali oldukça yüksek osilasyon genliğine sahiptir. Kontrol sinyalinde çatırdama mevcuttur. Ancak sistemin basamak cevabı (Şekil 5.10) ve referans izleme performansı (Şekil 5.12) tatmin edici düzeydedir.

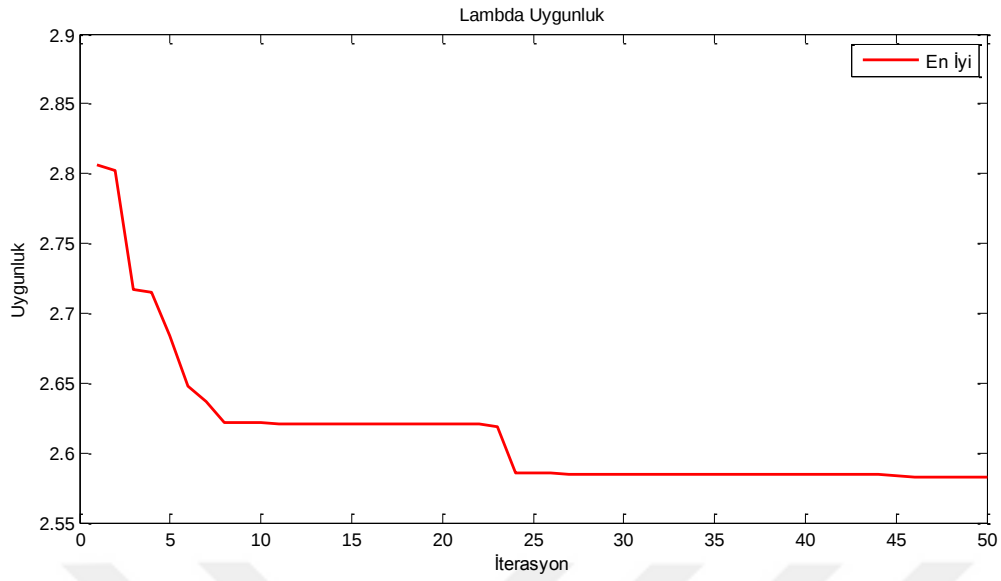


Şekil 5.11. Geleneksel YAKA ile KKK kontrol sinyali grafiği

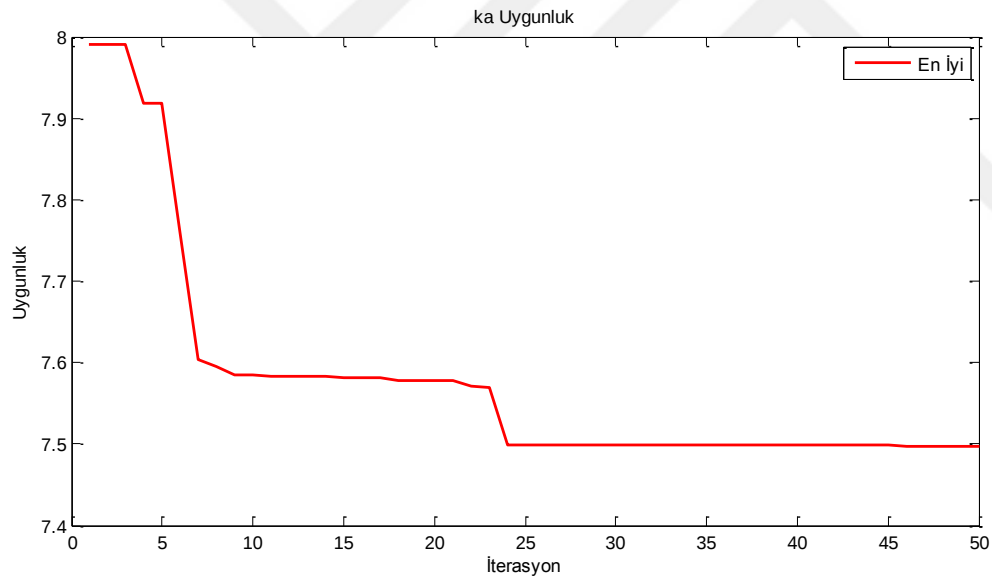


Şekil 5.12. Geleneksel YAKA ile KKK referans izleme performansı grafiği

Her parametreye uygun birer uygunluk fonksiyonu tanımlanarak yapılan bu çalışmada elde edilen uygunluk fonksiyonları değişimleri Şekil 5.13 ve 5.14'te verilmiştir. Tek uygunluk fonksiyonu kullanıldığında 15. iterasyonda duran iyileşme, önerilen YAKA ile yapıldığında daha ileriki iterasyonlarda da sürdüğü görülmektedir.

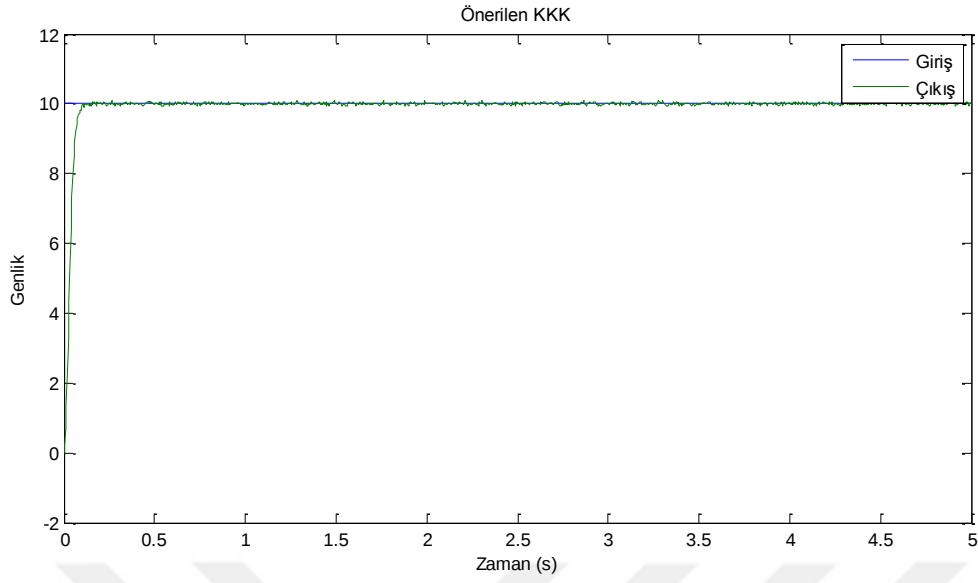


Şekil 5.13. Önerilen YAKA ile lambda parametresi uygunluk grafiği

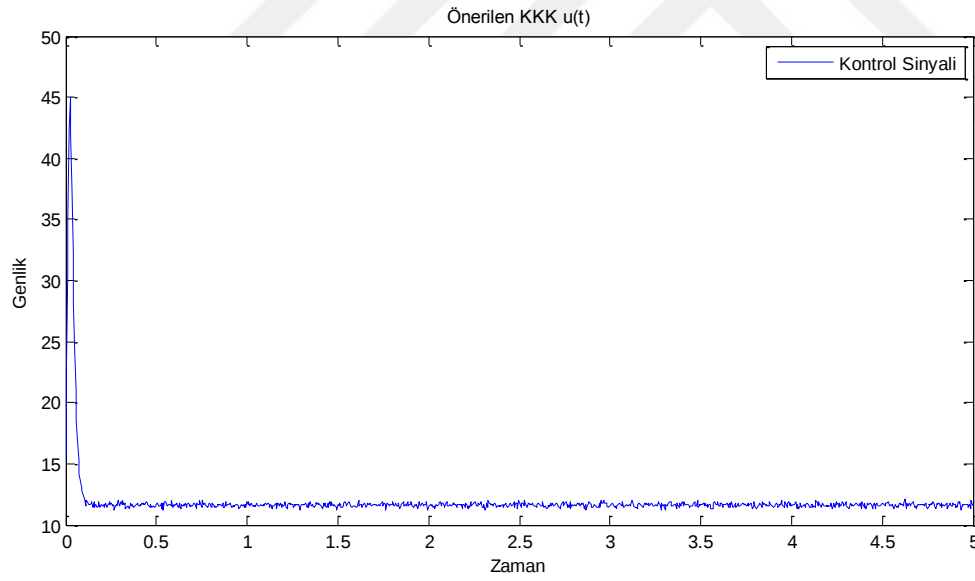


Şekil 5.14. Önerilen YAKA ile K_a parametresi uygunluk grafiği

Önerilen uygunluk fonksiyonları ile yapılan simülasyon sonucu elde edilen parametreler KKK ile sisteme uygulandığında Şekil 5.15 deki çıkış grafiği ile Şekil 5.16'daki kontrol sinyali karakteristiği elde edilmiştir. Her iki şekilden de anlaşılacağı gibi, önerilen yöntem ile YAKA performansı daha da arttırılmıştır. Kontrol sinyalindeki çatırdama azaltılmıştır.

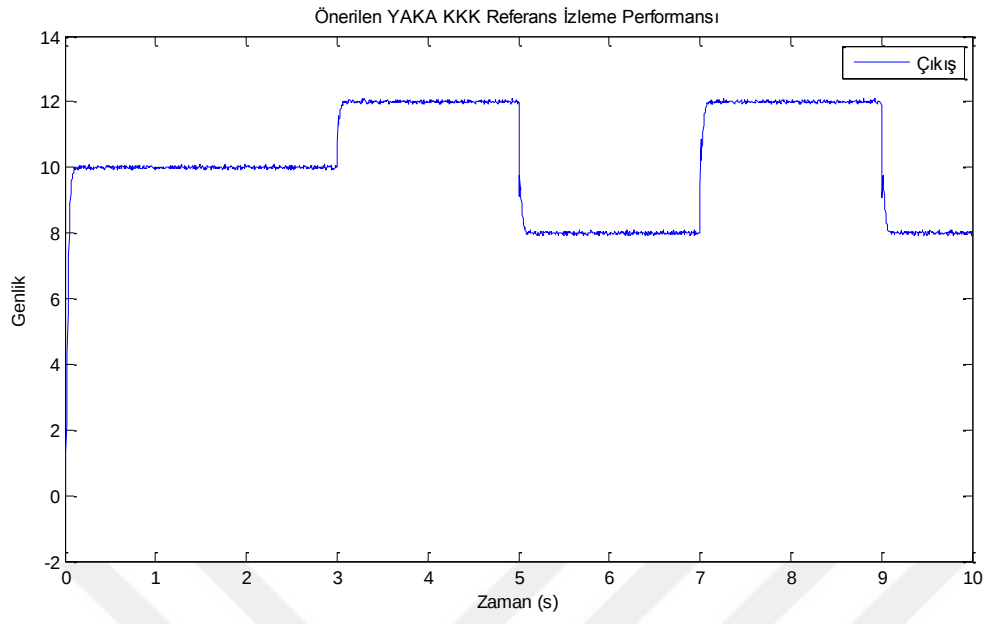


Şekil 5.15. Önerilen YAKA ile KKK çıkış grafiği



Şekil 5.16. Önerilen YAKA ile KKK kontrol sinyali grafiği

Aynı parametreler ile Şekil 5.17'deki referans izleme performansı incelendiğinde, daha yumuşak bir sistem çıkışı elde edildiği gözlenmiştir.



Şekil 5.17. Önerilen YAKA ile KKK referans izleme performans grafiği

6. SONUÇ

Bu tez çalışmasında, farklı sayılarda yapılan deneysel çalışmalar neticesinde YAKA'nın seçilen uygunluk fonksiyonları ile gösterdiği KKK ve PID denetleyicileri parametrelerinin ayarlanmasında oldukça başarılı verdiğini göstermiştir. YAKA'da, başlangıç değerleri kullanıcı tarafından belirlenen sınırlar içinde rastgele seçilerek oluşturulduğundan, optimuma ulaşma hızı her zaman aynı olmamakla beraber kullanılan uygunluk fonksiyonlarının etkisi rahatlıkla görülebilmektedir. Bu da sadece bir performans kriterinin kullanılmasının KKK ve PID denetleyicilerinde tüm parametrelerin optimum düzeyde ayarlanmasında yeterli olmayacağı sonucunu göstermiştir. En uygun değerlere ulaşma hızı bakımından sonuçlar değerlendirildiğinde, ITAE, ISE ve $\text{std}(u(t))$ ayrı ayrı kullanıldığında algoritmanın en iyi uygunluk değerini bulmakta daha güvenilir olduğu tespit edilmiştir. Bununla beraber, KKK denetleyici kontrol sinyalindeki kalitenin artırılması için kullanılan fonksiyonlara başka fonksiyonlar da eklenip daha düzgün kontrol sinyalleri, daha az iterasyon ile bulunabilir.

KAYNAKLAR

1. Korkmaz Tan R., Bora Ş., (2017). Modelleme ve Benzetim Ortamında Parametre Optimizasyonu ve Kullanılan Teknikler, *Journal of Engineering Sciences and Design*, 5(3), 685-697.
2. Gidemen, G. Furat, M. (2015). PID Parametrelerinin Ayarlama Yöntemleri: 2. Derece Sistem Modeline Uygulanması ve Karşılaştırmalı Olarak Değerlendirilmesi, *Çukurova Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, Cilt 30, Sayı 2, s. 355-362, 2015.
3. Furat, M., Eker, İ. (2016). Geliştirilmiş İkinci Derece Kayan Kipli Kontrol: Teori ve Deneysel Uygulama, *Engineer & the Machinery Magazine*
4. Köse, E., Abaci, K., Aksoy, S. Modeling and analysis of mechanical systems with PID and Sliding Mode Control, National Conference on Electrical, Electronics and Computer Engineering, Bursa, 2010, pp. 179-183.
5. Furat, M., & Eker, I. (2016). Chattering-eliminated adaptive sliding-mode control: an experimental comparison study. *Turkish Journal of Electrical Engineering & Computer Sciences*, 24(2), 605-620.
6. Yiğitbaşı, E. D. (2014). Yapay arı kolonisi optimizasyonu ile kenar bulma (Doctoral dissertation), Selçuk Üniversitesi Fen Bilimleri Enstitüsü.
7. Pamuk, N., Enerji Sistemlerinde Yapay Arı Kolonisi (YAK) Algoritması Kullanarak Yük Akışı Optimizasyonu, Akademik Bilişim 2013 – XV. Akademik Bilişim Konferansı Bildirileri 23-25 Ocak 2013 – Akdeniz Üniversitesi, Antalya
8. Karaboğa, D., (2004). Yapay Zeka Optimizasyon Algoritmaları, Atlas Yayın Dağıtım, İstanbul.
9. Engelbrecht A.P. (2007). Fundamentals of Computational Swarm Intelligence, Chicester.
10. Akyol, S., Alataş, B. (2012). Güncel Sürü Zekâsı Optimizasyon Algoritmaları”, *Nevşehir Üniversitesi Fen Bilimleri Enstitü Dergisi* 1, 36-50
11. Varol, E., Alataş, B., (2017). Sürü zekâsında yeni bir yaklaşım: Kuş sürüsü algoritması, *Dicle Üniversitesi Mühendislik Fakültesi, Mühendislik Dergisi*, Cilt:8, Sayı:1, 133-146.
12. Atasagun, Y., Bakteriyel Besin Arama Optimizasyon Algoritması ile Montaj Hattı Dengeleme, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi, 2010.
13. Telef Ç., A wolf Colony Algorithm, <https://caglartelefon.com/a-wolf-colony-algorithm/>, (Son Erişim Tarihi:25.04.2019).
14. Santosa, B., Ningrum, M. K., (2009). Cat Swarm Optimization for Clustering, International Conference of Soft Computing and Pattern Recognition, 54-59.

15. Wang, Z. H., Chang, C. C., Li, M. C., (2010). Optimizing Least-Significant-Bit Substitution Using Cat Swarm Optimization Strategy, *Information Sciences*, 10.1016/j.ins.2010.07.011.
16. Chu, S. C., Tsai, P. W., (2007). Computational Intelligence Based on The Behavior of Cats, *International Journal of Innovative Computing, Information and Control*, 3, 163-173.
17. Yan, G., Li C., (2011). An Effective Refinement Artificial Bee Colony Optimization Algorithm Based on Chaotic Search and Application for PID Control Tuning, *Journal of Computational Information Systems*, 7(9): 3309–3316.
18. Erkol, H. O. (2017). Ters Sarkaç Sisteminin Yapay Arı Kolonisi Algoritması ile Optimizasyonu, *Politeknik Dergisi*, 20(4) : 863-868.
19. Karaboğa D., Yapay Zeka Optimizasyon Algoritmaları, Nobel Akademik Yayıncılık, 5.Basım, Eylül 2017, s. 201.
20. Kıran, M. S., Gündüz, M., Şahman, M. A. Arı Kolonisi Algoritması Kullanarak En Kısa Yol Bulma. IV. İletişim Teknolojileri Ulusal Sempozyumu, 2009, Adana.
21. Grüter C., Farina M., (2009). The Honeybee Waggle Dance: Can We Follow The Steps?, *Trends in Ecology and Evolution*, 2009, Vol:24, No:5
22. Korkmaz, A., Bal Arılarında İletişim, <https://www.slideshare.net/alikorkmaznet/08-arlarda-iletim-49422922>, (Son Erişim Tarihi:15.04.2019).
23. Feldwick, O., Bees Colony Collapse and The Waggle Dance, <https://medium.com/@oliverfeldwick/bees-colony-collapse-and-the-waggle-dance-64658b7e8478>, (Son Erişim Tarihi:15.04.2019).
24. Küçüksillei E. U., Tokmaki M., (2011). Yapay Arı Kolonisi Algoritması Kullanarak Otomatik Ders Çizelgeleme, *Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü Dergisi*, 15(3), 203-210.
25. Eripek, O., Yapay Arı Kolonisi Algoritması (Artificial Bee Colony Algorithm), Senior Oracle DBA, 10g/11g OCP Ph. D. Program in Computer Engineering
26. Akay, B., Nümerik Optimizasyon Problemlerinde Yapay Arı Kolonisi (Artificial Bee Colony) Algoritmasının Performans Analizi, Doktora Tezi, 2009, Kayseri
27. Babayiğit, B., Özdemir, R. Modifiye Yapay Arı Koloni Algoritması ile Nümerik Fonksiyon Optimizasyonu, http://www.emo.org.tr/ekler/a27593adf5b4728_ek.pdf (Son Erişim Tarihi:15.04.2019).
28. Pamuk, N. Enerji sistemlerinde yapay arı kolonisi (yak) algoritması kullanarak yük akışı optimizasyonu. *2013 Akdeniz Üniversitesi Akademik Bilişim Konferansı*, 23-25.
29. Özkış, A. Sayısal optimizasyon problemlerinin çözümü için yapay arı kolonisi algoritmasının iyileştirilmesi, 2013, (Doctoral dissertation, Selçuk Üniversitesi Fen Bilimleri Enstitüsü).

30. Çavuş, V., Tuncer, A. (2017). İnsansız Hava Araçları İçin Yapay Arı Kolonisi Algoritması Kullanarak Rota Planlama. *Karaelmas Science and Engineering Journal*, 7(1), 259-265.
31. Kıran, M. S. Arı kolonisi ile şoför-hat-zaman optimizasyonu, 2010, (Doctoral dissertation, Selçuk Üniversitesi Fen Bilimleri Enstitüsü).
32. Kartal, B. Yapay Arı Kolonisi Algoritması ile Finansal Portföy Optimizasyonu. Doktora Tezi, 2015, İstanbul: İstanbul Üniversitesi Sosyal Bilimler Enstitüsü.
33. Karaboğa, D., Aslan, S. A new emigrant creation strategy based on local best sources for parallel artificial bee colony algorithm. 2016 24th Signal Processing and Communication Application Conference (SIU) (pp. 901-904). IEEE.
34. Karaboga, D., Akay, B. Artificial bee colony (ABC) algorithm on training artificial neural networks. 2007 IEEE 15th Signal Processing and Communications Applications (pp. 1-4). IEEE.
36. Bao, L., Zeng, J. C. Comparison and analysis of the selection mechanism in the artificial bee colony algorithm. 2009 ninth international conference on hybrid intelligent systems (Vol. 1, pp. 411-416). IEEE.
37. Öztürk, C., Hancer, E., & Karaboga, D. (2014). Küresel En İyi Yapay Arı Koloni Algoritması İle Otomatik Kümeleme. *Journal of the Faculty of Engineering & Architecture of Gazi University*, 29(4).
38. Batbat, T., Öztürk, C. (2016). Ayrık Yapay Arı Kolonisi Algoritması ile Protein Yapısı Tahmini. *Bilişim Teknolojileri Dergisi*, 9(3), 263.
39. Özer, Ş., & Baran, İ. (2014). Doğrusal parametrik ve doğrusal olmayan gerçek sistemlerin yapay arı kolonisi algoritması kullanılarak modellenmesi.
40. Zorarpacı, E., Özel, S. A., Güngör, S. (2015) An Artificial Bee Colony Based Algorithm for Feature Selection. *Çukurova Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, 30(1), 25-32.
41. Baydoğan, M. Ç., Şener, S. M. Tip imar yönetmeliğine uygun vaziyet planlarının yapay arı kolonisi yöntemiyle eniyilenmesi. *Erciyes Üniversitesi Fen Bilimleri Enstitüsü Fen Bilimleri Dergisi*, 30(2), 133-140.
42. Erkol H. O., (2017). Ters sarkaç sisteminin yapay arı kolonisi algoritması ile optimizasyonu, *Politeknik Dergisi*, 20(4): 863-868.
43. Gören A., Kontrol Sistemleri, <http://kisi.deu.edu.tr/aytac.goren/MAK3026/h5.pdf>, (Son Erişim Tarihi:26.05.2020).
44. Tokat S., Koordinat dönüşümüne Dayalı Zamanla Değişen Kayma Yüzeyi Tasarım Yöntemleri, Doktora Tezi, 2003, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü
45. Kalaycı M. B., Yiğit İ. (2015). Pratikte Kullanılan Bazı Kayan Kipli Kontrol Tekniklerinin Teorik ve Deneysel İncelenmesi, *Journal of the Faculty of Engineering and Architecture of Gazi University* Vol 30, No 1, 131-142

46. Akat S. B., Efe M. Ö., Bir Manyetik Askı Sisteminin Kontrolünde Dört Yöntemin Karşılaştırılması, TOK'07 Bildiriler Kitabı İstanbul, 5-7 Eylül 2007
47. Dumanay A. B., PID, Bulanık Mantık ve Kayan Kip Kontrol Yöntemleri İle İnternet Üzerinden DC Motor Hız Kontrolü, 2009, Balıkesir Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi.
48. Aydın S., Kayma Kipli Kontrolörlerde Kayma Yüzeyi Tasarımı Yöntemlerinin İncelenmesi ve Sınıflandırılması, 2008, Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi.
49. Utkin, V. I., (1977). Variable Structure systems with Sliding Modes, *IEEE Transaction on Automatic Control*, 22, 2, 212-222.
50. DeCarlo, R. A. DeCarlo, S., Zak, H., Mathews, G., Variable Structure Control of Nonlinear Multivariable Systems: A Tutorial, *Proceedings of the IEEE*, vol. 76, No. 3, Mar. 1988
51. Hung, J. Y., Gao, W., Hung, J. C. (1993). Variable Structure Control: A Survey, *IEEE Transaction on Industrial Electronics*, vol. 40, No. 1.
52. Università'Degli Studi Di Cagliari, Dipartimento Di Ingegneria Elettrica Ed Elettronica, "A Quick Introduction to Sliding Mode Control And Its Applications"
53. Birinci, B. E., Değişken Yapılı Kontrol Girişindeki Çatırtı Probleminin Giderilmesi için Yeni Yöntemlerin Geliştirilmesi, 2013, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi.
54. Levant, A. (1993). Sliding order and sliding accuracy in sliding mode control, *International Journal of Control*, 58(6), 1247-1263,
55. Bartolini, G., Ferrara, A., Levant, A., Usai, E. On second order sliding mode controller in Variable structure systems, sliding mode and nonlinear control, Springer Lecture Notes in Control and Information Sciences, Volume 247/1999.
56. Susperregui, A., Tapia, G., Y, Tapia, A., (2007) Application of Two Alternative Sliding-Mode Control Approaches to DC Servomotor Position Tracking", *IET Electric Power Applications*, Cilt 1, No 4, 611-621, 2007.
57. Eker İ., (2006). Sliding Mode Control with PID Sliding Surface and Experimental Application to an Electromechanical Plant, *ISA Transactions*, Cilt 45, No 1, 109-118.
58. Brandtstädter, H., Buss, M., Control of Electromechanical Systems Using Sliding Mode Techniques, *IEEE Conf. on Decision and Control*, 44th IEEE Conference, Seville, İspanya, 1947-1952, 12-15 Aralık 2005.
59. Park, T., Park, K., Sliding Mode Control Based on Pole-Placement Method for Position Control of Linear Stage, *ICCAS-SICE*, Fukuoka, Japonya, 5127-5130, 18-21 Ağustos 2009.
60. Nafa, F., Labiod, S., Chekireb, H., A Structured Sliding Mode Controller for a Class of Underactuated Mechanical Systems, *Systems, Signal Processing and Their Applications*

(WOSSPA), 7th International Workshop on Systems, Tipaza, Cezayir, 243-246, 9-11 Mayıs 2011.

61. Rao, S., Brandtstädter, H., Buss, M., Utkin, V.I., Sliding Mode Control in Mechanical Systems with Electric Actuators, Proceedings of the 8th Int. Workshop on VSS, Vilanova, İspanya, 1-4, 6-8 Eylül 2004.
62. Shin, S. Y., Lee, J. Y., Sugisaka, M., Lee, J. J., Decoupled Fuzzy Adaptive Sliding Mode Control For Under-Actuated Systems With Mismatched Uncertainties", Proc. IEEE Int. Conf. on Information and Automation, Harbin, Çin, 599-604, 20-23 Haziran 2010.



ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı :KAYA, Recep
 Uyuğu :T.C.
 Doğum tarihi/yeri :02.01.1988, Ceyhan
 Medeni hali :Bekar
 Telefon :0(545)723 7375
 Faks :0(326)755 9704
 e-mail :recep kaya.mf be17@iste.edu.tr



Özet:

Ekonomiye, güvenliğe odaklanan, istenen şartnamelere sahip bileşenleri, güvenilirlik, kalite çerçevesinde elektrik sistemlerinin geliştirilmesi ve sürdürülmesine yönelik kapsamlı bir tasarım bilgisine sahip yetkili bir elektrik mühendisi.

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Yüksek Lisans	İskenderun Teknik Üniversitesi Elektrik Elektronik Mühendisliği	2020
Lisans	İskenderun Teknik Üniversitesi Elektrik Elektronik Mühendisliği	2016
Lise	Dört yol Anadolu Lisesi	2005

İş Deneyimi

Yıl	Yer	Görev
2016-Halen	Er San Metalurji San. ve Tic. A.Ş	Elektrik Bakım Mühendisi

Yabancı Dil

İngilizce

Hobiler

Gitar, tiyatro, seyahat.

DİZİN

A

Artificial Bee Colony Algorithm · xv, 66

I

ISE · xv, 50, 58, 64

ITAE · xv, 50, 58, 64

K

Kayan Kipli Kontrol · xiv, xviii, xix, 1, 34, 35, 36, 37, 45,
56, 65, 67

KAYAN KIPLİ KONTROLÇÜLER · xviii, 29

O

OPTİMİZASYON · xvii, 4

P

PID · 3, xiv, xv, xviii, xix, xx, xxi, xv, 1, 2, 29, 31, 32, 35,
46, 48, 49, 50, 52, 54, 64, 65, 66, 67, 68

S

Sezgisel Optimizasyon · xvii, 5

Sliding Mode Control · xv, 65, 68, 69

Sürü Tabanlı Sezgisel Yöntemler · xvii, 6

Sürü Zekası, Uygunluk Fonksiyonu · xiv

Swarm Intelligence, Fitness Function · xv

Y

YAKA · xviii, xxi, xxii, xv, 1, 2, 7, 12, 13, 16, 17, 18, 22,
23, 24, 25, 26, 27, 28, 48, 49, 50, 51, 52, 53, 54, 57,
58, 59, 60, 61, 62, 63, 64

Yapay Arı Kolonisi · xiv, xviii, xxi, xv, 1, 7, 14, 23, 24,
25, 26, 27, 65, 66, 67

YAPAY ARI KOLONİSİ ALGORİTMASI · 2, 3, xiv, xviii, 12



TEKNOVERSİTE

