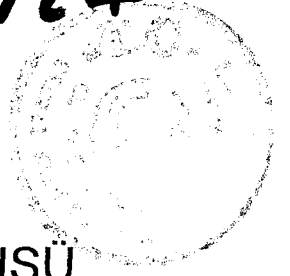


13/127



T.C

GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ

GENETİK ALGORİTMALAR KULLANARAK
PID KATSAYILARININ OPTİMİZASYONU

Nedim MUZOĞLU
YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ
ANABİLİM DALI

Tez Danışmanı
Yrd.Doç.Dr. Ş. Naci ENGİN

GEBZE
2005

Nedim MUZOĞLU'nun tez çalışması, G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun tarih ve ^{20.06.2005}2005/22 sayılı kararıyla oluşturulan jüri tarafından Elektronik MÜHENDİSLİĞİ Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE : Yrd. Doç. Dr. Ş. Naci ENGİN
(Tez Danışmanı)

ÜYE : Yrd. Doç. Dr. Abdulkadir BALIKÇI

ÜYE : Yrd. Doç. Dr. V. Emre ÖMÜRLÜ

ONAY

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 25.07.2005 tarih ve 2005/26 sayılı kararı.

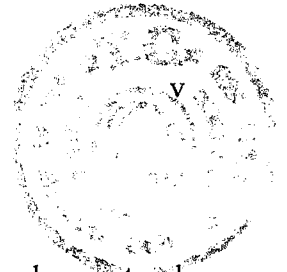
Yrd. Doç. Dr. Ahmet ÇAPOĞLU
GYTE Mühendislik ve Fen Bil.
Enstitüsü Müdür Yardımcısı

ÖZET

Genetik Algoritmalar (GA) en iyinin hayatta kalmasına dayanan ve genetik mekanizmasına göre adaptif araştırmalar yapan algoritmalar. Genetik Algoritmalar önceki bilgilerinden yeni çözümler üreterek daha iyi çözümlere ulaşmaya çalışır. Bunun için iyinin ne olduğunu belirleyen uygunluk(fitness) fonksiyonu, yeni çözümler üretmek için kopyalama(recombination), mutasyon gibi operatörleri kullanır.

Genetik Algoritmalar (GA) karmaşık matematik hesaplamaları gerektiren problemlerin optimizasyonunda tercih edilen bir yöntemdir. Bu çalışmada parametrik belirsizliği olan PID kontrolörlerin katsayıları ISE (integral of squared error – karesi alınmış hatanın integrali) performans indeksine göre genetik algoritmalar kullanılarak optimize edilmiştir. Böylece karesi alınmış hatanın integralini minimum yapan K_p , K_i ve K_d katsayıları elde edilmiştir. Kontrol sisteminde ISE performans kriterinin minimum olmasını sağlayan PID katsayıları aynı zamanda maksimum aşım ve yerleşme süresini de minimum yapması beklenir.

Yapılan çalışmanın üçüncü bölümünde Genetik Algoritmalarla elde edilen sonuçlarla klasik metotlarla elde edilen sonuçlar karşılatılmıştır. Sonuç olarak GA'lar kullanılarak optimize edilmiş PID kontrolörünün parametrik hatalara duyarsız olduğu tespit edilmiştir. Elde edilen sonuçlar klasik metotlarla bulunan sonuçlarla performans açısından karşılaştırılmıştır.



SUMMARY

Genetic Algorithms(GAs) are an adaptive search process and based on natural selection. Genetic Algorithm tries to reach to the solutions by producing new solution from the proposed solutions. GAs determine the best solutions by fitness function , uses recombination and mutation operators to produce new solutions.

Genetic Algorithms can be preferred to optimise the complex mathematical problems. In this study, ISE performance index are optimised by using GAs for PID systems which may have modelling errors and parametric uncertainty. Then Gas obtain the best K_p , K_i and K_d that make the ISE (integral of the square of the error) performance index minimum. It is thought that PID parameters which minimize the ISE performance index also supply the minimum peak and minimum settle time.

At the third part of the study GAs results are compared with results obtained by classical methods. Finally, it has been found that PID control parameters using GA have been insensitive to parametric error.

TEŞEKKÜR

Çalışmam esnasında göstermiş olduğu ilgi ve katkıları nedeniyle danışmanım Sayın Yrd. Doç. Dr. Ş.Naci ENGİN' e içtenlikle teşekkür ederim.

Yine arkadaşım Müh. Baykant ALAGÖZ'e, arkadaşım Arş. Gör. Hüseyin SERİM'e, algoritmaların yazımında bilgisayarını ödünç veren arkadaşım Müh. Ruda TUĞAN'a ve müdürüm Sayın Müh. Fikret ŞİR'e sonsuz teşekkürlerimi sunmayı borç bilirim.

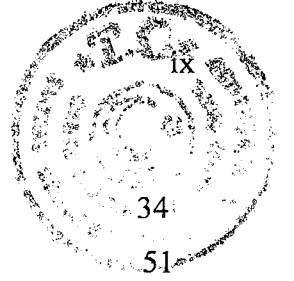


İÇİNDEKİLER DİZİNİ

Sayfa

ÖZET	IV
SUMMARY	V
TEŞEKKÜR	VI
İÇİNDEKİLER DİZİNİ	VII
SİMGELER DİZİNİ	X
ŞEKİLLER DİZİNİ	XI
TABLolar DİZİNİ	XIII
 1.GİRİŞ	 1
1.1 Genel Bilgiler	1
2.KONTROL SİSTEMLERİ	3
2.1. Kontrol Edilen Sistemler Hakkında Genel Bilgiler	3
2.1.1.Kontrolör Yapıları	4
2.2.Kontrol Sistemlerinde Kararlılığın İncelenmesi	5
2.2.1.Karmaşık S-domeni yaklaşımı	5
2.2.2.Zaman Domeni Yaklaşımı	6
2.2.2.1.Sürekli Hal Hatasının İncelenmesi	7
2.2.2.2.Hata Fonksiyonunun Optimize Edilmesi	9
2.2.3.Frekans Domeni Yaklaşımı	10
2.3.Pid Kontrolörler ve Tasarım Yöntemleri	10
2.3.1.Pid Kontrolörler	10
2.3.2.Pid Parametrelerinin İncelenmesi	11
2.3.3.Pid Tasarım Yöntemleri	12
2.3.3.1. Zaman Domeni Tasarımı	13
2.3.3.2. Frekans Domeni Tasarımı	14
2.3.3.3. Ziegler-Nichols Kuralları	15
2.3.3.4. Autotuning Sistemleri	17
2.4.Genetik Algoritmalar	18
2.4.1 Genetik Algoritmalar Giriş	18
2.4.2 Genetik Algoritmaların Sağladığı Avantajlar	19

2.4.3. Genetik Algoritmaların Kullanım Alanları	20
2.4.4. Genetik Algoritmaların Diğer Klasik Yöntemlerden Farklılıkları	20
2.4.5. Genetik Algoritmaların Temel Yapısı ve İşleyişi	20
2.4.6. Kromozomların Kodlanması	22
2.4.6.1 İkili Kodlama	22
2.4.6.2. Permütasyonlu Kodlama	23
2.4.7. Başlangıç Populasyonunun Oluşturulması	23
2.4.8. Uygunluk Değerlerinin Hesaplanması	23
2.4.9. Yeniden Üretim İşlemi	24
2.4.9.1. Uygun Bireylerin Seçilmesi	24
2.4.9.1.1. Orantılı Seçim Mekanizmaları	25
2.4.9.1.2. Sıralı Seçim Mekanizması	26
2.4.9.1.3. Turnuva Seçim Mekanizması	26
2.4.9.1.4. Denge Durum Seçim Mekanizması	26
2.4.9.1.5. Elitist Seçim Mekanizması	26
2.4.10. Genetik Operatörler	27
2.4.10.1. Çaprazlama Operatörü	27
2.4.10.1.1. Tek nokta çaprazlama	27
2.4.10.1.2. İki Noktalı Çaprazlama Operatörü	28
2.4.10.1.3. Uniform Çaprazlama Operatörü	28
2.4.10.1.4. Mutasyon Operatörü	29
2.4.11. Genetik Algoritmanın Kontrol Parametreleri	29
2.4.11.1. Populasyon Büyüklüğü (N)	29
2.4.11.2. Çaprazlama Oranı(p_c)	30
2.4.11.3. Mutasyon Oran(p_m)	30
2.4.11.4. Seçim Mekanizmaları	30
2.4.11.5. Ölçeklendirme Fonksiyonu Faktörü	30
2.4.11.6. Nesil Ayrılığı(G)	30
2.4.11.7. Bitirme Koşulu	31
2.4.11.8. Ölçeklendirme Fonksiyonu	31
3. GENETİK ALGORİTMALAR KULLANILARAK PID KATSAYILARININ OPTİMİZE EDİLMESİ	33
3.1 Pid Optimizasyon Genel Yapısı	33



3.2 Optimal Pid Parametrelerinin Hesaplanmasına İlişkin Örnek
Uygulamalar

4. SONUÇLAR VE ÖNERİLER

KAYNAKLAR

ÖZGEÇMİŞ

34

51

53

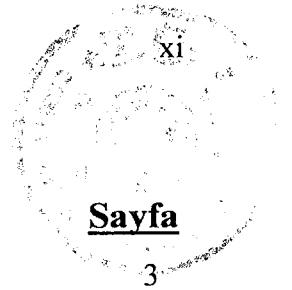
55



SİMGELER DİZİNİ

- N Populasyondaki kromozom sayısı.
- L Bir kromozomun ifade edildiği bit sayısı.
- f_i Populasyondaki i . kromozomun uygunluk değeri.
- f Populasyonun toplam uygunluk değeri.
- f_{avg} f / n Populasyonun ortalama uygunluk değeri.
- p_i f_i / f Populasyondaki i . kromozomun seçilme olasılığı.
- m_i f_i / f_{avg} Populasyondaki i . kromozomun kopyalarının beklenen sayısı.
- f_i^* i . kromozomun ölçeklendirilmiş uygunluk değeri.
- U_{max} Seçilen aralığın üst sınır değeri.
- U_{min} Seçilen aralığın alt sınır değeri.

ŞEKİLLER DİZİNİ



<u>Sekil</u>	<u>Sayfa</u>
2.1 Genel bir kontrol sistemi diyagramı .	3
2.2. Kontrol Edilen Sistem	4
2.3. Zaman tanım bölgesi birim basamak cevabı	7
2.4. Genel PID yapısı .	11
2.5. Ziegler-Nichols Birinci Metodu s-şeklindeki eğim cevabı .	15
2.6. Ziegler-Nichols İkinci Metodu, P_{cr} osilasyon periyodu .	16
2.7. Otomatik ayar sistemi blok diyagramı.	17
2.8.GA nın akış diyagramı.	21
3.1 GA larla PID Optimizasyon Genel Yapısı	33
3.2 GA Kullanarak PID Optimizasyonu Genel Program Yapısı.	34
3.3 Ziegler Nichols Yaklaşımı Birim Basamak Cevabı, $K_p=18$, $K_i=12.8$, $K_d=6.3$	35
3.4 Sistemin $K_p=18$, $K_i=5.8498$, $K_d=13.8456$ İçin Birim Basamak Cevabı.	35
3.5 GA larla Hesaplanan, Sistemin Birim Basamak Cevabı.	37
3.6 Her İterasyon Sonunda Ortalama Karesel Hatanın Değişim Grafiği.	38
3.7 K_p nin Genetik İterasyonlar Boyunca Değişimi .	39
3.8 K_d nin Genetik İterasyonlar Boyunca Değişimi .	39
3.9 K_i nin Genetik İterasyonlar Boyunca Değişimi .	40
3.10. PID Kontrolörlü Sistemin Birim Basamak Cevabı .	40
3.11 GA larla Hesaplanan, Sistemin Birim Basamak Cevabı .	42
3.12 Genetik İterasyon Boyunca Hatanın Minimize Olduğunu Gösterir.	43
3.13. PID Katsayılarının Elde Edildiği Kromozomun Uygunluk Değeri Değişim Grafiği.	44
3.14 Genetik İterasyon Boyunca Kromozomların Uygunluk Değerleri Grafiği	45
3.15 Proses Sıcaklık Kontrol Sistemi	45
3.16 Proses sıcaklık kontrol sistemine ait kök yer eğrisi	46
3.17 Proses sıcaklık kontrol sistemine ait kök yer eğrisi birim basamak cevabı	46
3.18 Proses sıcaklık kontrol sistemine ait Norman tarafından önerilen birim basamak cevabı	47

3.19 Proses Kontrol Sistemi GA Birim Basamak Cevabı	48
3.20 Sıcaklık Kontrol Sistemine ait Ortalama Karesel Hatanın Değişim Grafiği	49
3.21 Optimal Çözümü Sağlayan Kromozoma Ait Uygunluk Değeri Değişimi	50



TABLolar DİZİNİ

<u>Tablo</u>	<u>Sayfa</u>
2.1 Ziegler-Nichols Basamak Cevabı Metodu Sonuçları	16
2.2. Ziegler-Nichols Frekans Cevabı Metodu Sonuçları	17
2.3. Tek Noktalı Çaprazlama Operatörü	27
2.4. İki noktalı Çaprazlama Operatörü	28
2.5. Uniform Çaprazlama Operatörü	28
2.6. Mutasyon Operatörü	29
3.1. Hesaplanan Farklı PID parametreleri İin Optimizasyon Sonuçlarının Karşılaştırılması	37
3.2 Genetik Algoritmaların Bulduėu Sonuçlarla Ogata Tarafından Hesaplanan Sonuçların Karşılaştırılması	42
3.3 Proses Kontrol Sistemine ait Sonuçların Karşılaştırılması	48

1.GİRİŞ

1.1 Genel Bilgiler

Sistem kontrolünde sistemin geçici rejim çalışması sırasında istenmeyen sınır değerlerine ulaşması ve fiziksel değerlerini zorlaması yada aşması karşılaşılan sorunların başında gelmektedir. Bir otomatik kontrol sisteminde giriş büyüklükleri değiştikçe çıkış büyüklüklerinin de en kısa sürede beklenen değerlere ulaşması istenir. Eğer kontrol sistemi değişen giriş değerine karşı beklenen çıkış değerlerini üretilmiyorsa, sisteme kontrol cihazı (kontrolör) eklenerek sistemin uygun çıkışlar üretmesi sağlanır.

PID kontrolörler bu sorunu ortadan kaldırmak için endüstride en çok kullanılan kontrolör yapılarının başında gelmektedir. PID kontrolörler oransal-türevsel-integral denetleyicilerin etkisini tek bir birim içerisinde birleştirirler. PID kontrolörler sistemde sıfır kalıcı durum hatası olan hızlı bir cevap sağlar. Ancak PID kontrolörün K_p , K_i ve K_d parametreleri uygun olarak ayarlanmadığında kontrolörün bu üstün özelliklerinden yararlanılamaz.

PID kontrolörün katsayılarının ayarlanmasında kullanılan matematiksel modellerde birçok kabuller ve ihmaller yapılır. Bu da kontrolörün performansını düşürür. Sistemi tam olarak temsil eden karmaşık modellerin de lineerleştirilmesi zor olduğundan kullanışlı değildir. Bununla birlikte Genetik Algoritmalar (GA) bir çok karmaşık optimizasyon problemlerinde başarılı sonuçlar vermektedir. GA'nın temel avantajı optimize etmeye çalıştıkları problemle ilgili herhangi bir bilgiye ihtiyaç duymamalarıdır. Sistemin matematik model bilgilerini gerektirmez. Karmaşık matematik hesaplamaları yerine yalnızca giriş-çıkış bilgilerine ihtiyaç duyar. Bu avantajından dolayı karmaşık problemlerin optimizasyonu için tercih edilebilir bir yöntemdir. Uygun performans indeksi seçimi ile sistemin arzu edilen dinamik davranışları ve kararlılığı elde edilebilir. Bu şekilde, performans indekslerinin seçimi, sistemin istenilen dinamik davranışların elde edilmesi için çok önemlidir[Zuo, 1995].

Bu tez çalışmasında; PID katsayılarının hesaplanmasında kullanılan klasik yöntemlere hem alternatif hem de optimal katsayıları GA mantığı içerisinde araştıran Matlab dilinde bir program yazılmıştır. Program algoritma boyunca genetik süreç çerçevesinde PID sisteminin parametrelerini optimize etmeye ve en optimal değeri bulmayı amaçlamıştır. GA'nın çıkışın toplam karesel hatasını minimum değere

ekebilmesi iin ISE(integral of squared error – karesi alınmıř hatanın integrali) performans indeksi özümün bulunmasında esas alınmıřtır. Sonuç olarak genetik döngüdeki kriterlere ve performans indeksine göre ortalama karesel hatanın minimum olmasını saėlayan optimal K_p , K_i ve K_d deėerleri program tarafından bulunur.

PID parametrelerinin bulunmasında bir ok tasarım metodu bulunmaktadır. Tasarımı yapmak oėu zaman kök yer eėrileri, Bode ve Nyquist izimleri gibi karmařık, yorumlanması zor ve ayrıca belirsizlikleri olabilen kontrol yöntemlerinin uygulanmasını gerektirir. Bu tez alıřması kapsamında yazılan optimal PID katsayılarını GA'lar yardımıyla bulan Matlab programı kullanımda kolaylık ve performans saėlayacaktır.

Tezin birinci bölümünde yapılan tez alıřmasına ilişkin genel bilgiler verilmiřtir. İkinci bölümde PID katsayılarının tasarlanması iin kullanılan klasik yöntemler anlatıldıktan sonra optimizasyon iin kullanılacak olan genetik algoritmalar detaylı bir şekilde incelenmiřtir. Üüncü bölümde ise klasik metotlarla hesaplanmış optimal PID katsayılarıyla, genetik algoritmanın hesapladıėı PID katsayıları sistem performansı aısından karşılaştırılmıřtır.

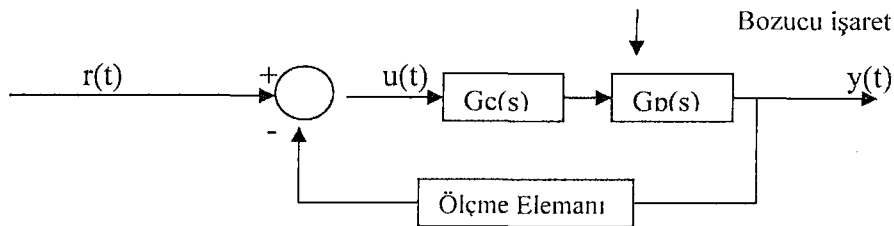
2. KONTROL SİSTEMLERİ

Yaşantımızın her alanındaki bir çok cihaz kontrol sistemleri yardımıyla bizlere yardımcı olmaktadır. Bu bölümde kontrol sistemleri hakkında genel bilgiler verildikten sonra kontrol sistemlerinin kararlılık kriterleri incelenecektir. Bundan sonra ise PID kontrolörünün tanıtılması ve katsayılarının klasik modellerle tasarlanması hakkın da bilgiler verilecektir.

2.1. Kontrol Edilen Sistemler Hakkında Genel Bilgiler

Kontrol edilen sistemlerde, sistem cevabının giriş işaretinin referans değerini izlemesi yada çıkış cevabının belirli değerde tutulması istenir. Sistemin çıkışından alınan geri besleme ile giriş işareti karşılaştırılarak kalıcı hata işareti belirlenir. Sistemin girişine uygulanan giriş işareti sonucunda, sistemin çıkışında oluşan kalıcı hal hatasının aldığı değer kullanıcı tarafından uygunluk kriterlerine göre değerlendirilir. Sistemin girişindeki işareti değiştikçe çıkış büyüklüğünün mümkün olduğu kadar kısa bir süre de beklenen sürekli hal çıkışına ulaşması istenir. Kontrol sistemi kısa bir surede beklenen kararlı yapıya ulaşamıyorsa sisteme bir kontrol cihazı eklenerek beklenen sonuçlar elde edilmeye çalışılır. Kontrol sistemini kullanan kullanıcı; kararlı hal hatasının minimum olmasına, geçici halin hızlılığına, sistemin kararlı olmasına dikkat etmelidir.

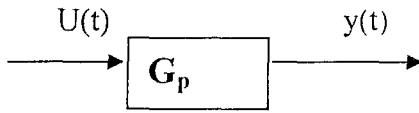
Kontrol edilen sistem çıkışındaki işaret, sistemden ya da dışardan gelen bazı etkenler tarafından devamlı değiştirilmek istenir. Bu nedenle çıkış işareti sürekli kontrol altında tutulmalıdır. Çıkış işareti bir ölçme cihazı kullanılarak girişteki işaretle karşılaştırılır. Girişle çıkış arasında istenmeyen ölçüde bir fark varsa kontrol sistemi çıkışı uygun değerlere çekecek bir işaret üretmelidir. Kontrol sistemi bu ayar değerine elle ayarlanabileceği gibi sürücü kontrol elemanı yardımıyla da çıkış işaretini uygun değere çekecek ayarlama yapılabilir. Genel bir kontrol sisteminin blok diyagramı aşağıdaki gibidir.



Şekil 2.1 Genel bir kontrol sistemi diyagramı

$r(t)$: Referans işareti
 $u(t)$: Kontrol değişkeni
 G_c : Kontrol eden sistem
 G_p : Kontrol edilen sistem
 $y(t)$: Çıkış işareti

Şekil 2.1 de gösterilen kontrol sisteminin gerçekleştirilmesinde aşağıdaki üç adım da takip edilebilir[Kuo ,1992].



Şekil 2.2. Kontrol Edilen Sistem

- 1)Sistemin ne yapması ve nasıl yapması gerektiğinin belirlenmesi.
- 2)Kontrolör ya da kompensatör yapısının kontrol edilen sisteme nasıl bağlanacağını belirlenmesi.
- 3)Kontrolör parametrelerinin tasarımın hedefine göre belirlenmesi.

2.1.1 Kontrolör Yapıları

Kontrol edilen doğrusal sistemin dinamiği şekil2.2 ile verilen blok ile ifade edilir. Tasarımın amacı $y(t)$ çıkış değişkeninin istenilen şekilde davranmasıdır. Geleneksel yöntemlerde, tasarımcı kontrolörünü kontrol edilen prosese göre nereye yerleştirmesi gerektiğini genellikle önceden belirler ve sisteme sabit bir yapıda tasarım uygulanır. Amaç sistem davranış özelliklerini karşılamaya ve değiştirmeye yönelik olduğunda genellikle sabit yapıya kompanzasyon adı verilir [Kuo ,1992].

Yaygın kontrol yapıları aşağıdaki gibidir;

Seri Kompanzasyon: Kontrolörün kontrol edilen sisteme seri olarak yerleştirildiği en yaygın kontrolör seri ya da kaskat bağlanabilir.

Geribeslemeli Kompanzasyon: Kontrolörün bir alt çevrime yerleştirildiği yapıya geri beslemeli kompanzasyon adı verilir.

Durum geribeslemeli Kompanzasyon: Sistem durum değişkenlerinin sabit kazanç katsayıları ile geri beslendiği yapıya durum geribeslemesi denir.

Seri-geribeslemeli Kompanzasyon: Bir seri birde geribeslemeli kontrolörün kullanıldığı yapılardır.

Yukarıdaki kompanzasyon yapıları içerisinde PID kontrolörü yaygın uygulama alanı bulan ve etkin işaretin oransal, integral ve türevsel bir bileşimini sisteme uygulayan bir kontrolördür. Bu işaret bileşenleri zaman tanım bölgesinde kolaylıkla gerçekleştirilebilir. PID kontrolörü genellikle zaman tanım bölgesi yöntemleriyle tasarlanır[Kuo ,1992].

Belirli bir kontrolörde karar kılındıktan sonra kontrolör türü elemanlarıyla birlikte belirlenmelidir. Kontrolör parametrelerini belirlemek ikinci adımdır. Bu parametre değerleri kontrolörü oluşturan bir ya da daha fazla transfer fonksiyona ilişkin katsayılardır. Kontrolör parametreleri tüm tasarım kriterlerini gerçekleştirecek şekilde seçilir. Parametre değerleri değiştirilerek aşım sağlatılabilir. Ancak yükselme zamanı koşulunu ayarlamak için başka bir parametre değiştirildiğinde aşım kriteri artık sağlanamayabilir. Tasarım kriterleri ve kontrolör parametrelerinin sayısı arttıkça tasarım işlemi de karmaşıklaşır[Kuo ,1992].

2.2 Kontrol Sistemlerinde Kararlılığın İncelenmesi

Bir sistemin kararlı olup olmadığını inceleyen çeşitli yöntemler vardır. Bu yöntemler;

i)s-domeni, yaklaşımı ii)zaman domenii yaklaşımı ve iii)frekans domenii yaklaşımıdır.

Çeşitli giriş fonksiyonlarına bağlı olarak bulunan cevap fonksiyonlarında zaman sonsuza giderken cevap sonlu bir değerde kalıyorsa, sistem kararlı olarak belirlenir. Aksi takdirde, sistem kararsızdır [Astrom, 1998; Yüksel, 1997].

2.2.1 Karmaşık S-domenii yaklaşımı

Kapalı döngü bir sistemin kararlılığının transfer fonksiyonu, kutupların karmaşık düzlemdeki yerlerine göre saptanabilir. Sistemin kutupları transfer fonksiyonunun paydasındaki denklemin kökleri olarak bulunur. Geribeslemeli bir sistemin kararlı olabilmesi için sistemin transfer fonksiyonu kutuplarının negatif gerçek kısımlara sahip olması gerekmektedir. Transfer fonksiyonunun kutuplarının karmaşık düzlemdeki yeri sistemin kararlılığı ve dinamik davranışını tanımlar. S düzleminin sol yarısında yer alan gerçek kısımları negatif işaretli kökler için cevap eğrileri ya üstel azalan ya da sönümlü

titreşimli şeklinde olmaktadır. Buda sistemin kararlı olduğu anlamına gelir. Buna karşılık s -düzleminin sağ yarısında yer alan gerçek kısımları pozitif işaretli kökler için cevap eğrisi ya üstel artan ya da genlikleri sürekli şekilde artan sinüsoidal şeklinde olmaktadır. Bu durumda sistemin kararsız olduğu anlamına gelir. Sanal eksen üzerinde yer alan ve gerçek kısmı sıfır kök durumuna karşılık sistem cevabı ya sabit ya da sönümsüz titreşimlidir. Kararlı bir cevap için sistemin mutlak olarak tüm kutuplarının sol yarı düzlemde o yer alması gerekir.

2.2.2. Zaman Domeni Yaklaşımı

Sistemlerin test edilebilmesi için bazı önemli giriş fonksiyonları tanımlanmıştır. Bu giriş fonksiyonları yardımıyla matematiksel çözümler gerçekleştirilebilir. Zaman tanım bölgesi davranışı birim basamak cevaba karşın incelenebilir. İkinci mertebeden birim geri beslemeli bir istemin zaman tanım bölgesi cevabının tipik bir şekli aşağıda verilmiştir.

$$c(t) = 1 - \frac{1}{1 - \zeta^2} e^{-\zeta \omega_n t} \sin(\omega_n \sqrt{1 - \zeta^2} t + \cos^{-1} \zeta) \quad (2.1)$$

Bir otomatik kontrol sistemine basamak şeklinde bir giriş uygulandığında;

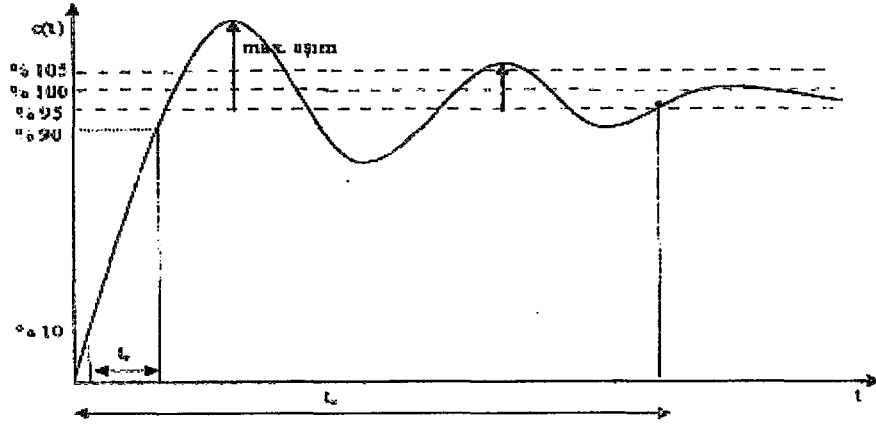
Gecikme Zamanı (t_D): ortalama çıkışın nihai değerin yarısına erişinceye kadar geçen zamana gecikme zamanı denir.

Yükselme Zamanı (t_R): Birim basamak cevabının %10 ile %90 ı arasındaki zamana yükselme zamanı denir.

Yerleşme Zamanı (t_s): Birim basamak giriş cevap eğrisinin son değerinin %x bölgesine, bundan sonra değişimi bu bölge içerisinde kalmak koşulu ile ilk girdiği ana kadar geçen zamana yerleşme zamanı denir. Burada x değeri %5 yada %2 seçilir.

Aşım: Birim basamak giriş cevap eğrisinin maksimum değeri ile sürekli hal değeri farkının sürekli hal değerine oranının yüzdesine aşım denir.

Sönüm oranı (ζ): Sisteme herhangi bir giriş uygulandıktan sonra meydana gelen geçici hal söner ve geriye kararlı hal hataları kalır, sisteme genellikle basamak, rampa ve parabolik test fonksiyonları kullanılır. Sönüm oranı (ζ), kararlı sistemler için $0 < \zeta < 1$ arasındadır, ζ küçüldükçe aşım artar.



Şekil 2.3. Zaman tanım bölgesi birim basamak cevabı

Bir kontrol sisteminden, zaman domenin de, birim basamak fonksiyonu girişi için elde olunacak $c(t)$ çıkışının; küçük aşım, küçük yükselme süresine, küçük gecikme zamanına, küçük yerleşme zamanına sahip olması ve hatasının küçük kalması beklenir[Sarıoğlu, 1999 ; Hurma, 1998].

2.2.2.1 Sürekli Hal Hatasının İncelenmesi

Bir kontrol sisteminden beklenen, başvuru değerinden uzaklaşan kontrol büyüklüğünün yeniden başvuru değerine erişmesini sağlamaktır. Başvuru değerinde herhangi bir değişiklik olduğunda kontrol büyüklüğü bu yeni başvuru değerine erişmeli ve düzenli bir şekilde bu değeri korumalıdır. Kontrol sisteminin bu durumuna sürekli durum çalışması denir. Kontrol sistemlerinin sürekli duruma erişinceye kadar süren çalışma şekline geçici durum çalışması denir. Kontrol sistemlerinin geçici durum ve kararlı durum çalışmalarında kararlılık, hızlı cevap ve duyarlılık gibi özelliklerin sağlanması gerekir. Eğer $y(t)$ bir sürekli sistem yanıtını ifade ediyorsa, $y_t(t)$ geçici yanıtı ve $y_{ss}(t)$ sürekli hal yanıtı ifade etmek üzere

$$y(t) = y_t(t) + y_{ss}(t) \quad (2.2)$$

yazılabilir .

Kontrol sisteminde geçici hal yanıtı, zaman yanıtının, zaman ilerledikçe, sıfıra doğru giden kısmı olarak tanımlanır. Sürekli hal yanıtı ise geçici hal yanıtının söndükten sonra zaman yanıtının geriye kalan kısmıdır. Tüm gerçek kararlı sistemler sürekli hale erişmeden önce, belirli ölçülerde geçici hal davranışlarıyla karşılaşılır. Geçici hal yanıtının kontrolü sistemin dinamik davranışı yönünden önemli ve anlamlıdır. Sürekli

hale erişmeden önce istenen yanıt ile sistem çıkış cevabı arasındaki sapma hassasiyetle kontrol edilmelidir.

Bir sistemin sürekli hal yanıtı da çok önemlidir, çünkü zaman ilerledikçe sistem yanıtının nerede sona erdiğini belirtir. Genelde çıkışın sürekli hal yanıtı istenen referansla uyuşmadığında sistemde kararlı hal hataları bulunduğu anlamına gelir.

Kontrol sistemlerinin zaman tanım bölgesinde incelenmesi temelde sistemlerin geçici ve sürekli hal yanıtlarını düzeltme anlamına gelir. Tasarım probleminde değerlendirme genellikle geçici hal ve sürekli hal davranışlarına göre yapılır. Sistem kontrolörleri tüm koşullar sağlanacak şekilde tasarlanır.

Kontrol sistemlerinin zaman tanım bölgesi analizini kolaylaştırmak için basamak fonksiyonu en çok kullanılan test işaretlerinden biridir.

Birim Basamak Fonksiyonu: Girişi referans girişinde ani değişimleri ifade eder. R genliğindeki basamak fonksiyonunun matematiksel ifadesi, R sabit olmak üzere;

$$r(t) = \begin{cases} R & t \geq 0 \\ 0 & t < 0 \end{cases} \quad (2.3)$$

şeklinde dir. Matematiksel olarak kolayca ifade edilebilen basamak fonksiyonu sistemlerin cevap analizinde kullanılmaktadır. Başlangıçtaki sıçrama nedeniyle basamak fonksiyonu sistemdeki ani değişimlere karşı davranışı belirlemede çok yararlı olur [Kuo ,1992].

Sürekli Hal Hatası: Kontrol sistemlerinin amaçlarından biri çıkış yanıtının sürekli halde belirli bir referans işaretini hatasız bir şekilde izlemektir. Sürekli halde referans girişiyle çıkış işareti arasındaki fark, sürekli hal hatası olarak tanımlanır. Burada hata mümkün olduğu en kısa zamanda sıfıra indirgenebilir bir işaret olarak değerlendirilebilir. $r(t)$ giriş işareti, $y(t)$ çıkış işareti, $e(t)$ hata olmak üzere;

$$e(t) = r(t) - y(t) \quad (2.4)$$

şeklinde ifade edilebilir. Kararlı hal hatası ise

$$e_{ss} = \lim_{s \rightarrow \infty} e(s) \text{ olarak tanımlanır [Kuo ,1992].} \quad (2.5)$$

2.2.2.2 Hata Fonksiyonunun Optimize Edilmesi

Modern kontrol teorisi istenilen sistem performansının sistem mühendisleri tarafından nicel olarak tanımlanabileceğini öngörür. Yapılan çalışmaya uygun tanımlanan bir performans indeksi sistem performansını artırabilir. Bir kontrol sisteminin optimum kontrol sistemi olarak tanımlanabilmesi için sistem parametreleri ve de performans indeksi en uygun değeri almalıdır. İndeks için en uygun değer genelde minimum olduğu değerlerdir. İndeks her zaman pozitif veya sıfır olan bir sayıdır. [Dorf and Bishop, 2001].

Kesin bir çalışma kriteri olmamakla birlikte, hata fonksiyonu $e(t)$ 'nin minimum yapılmasının, sürekli hal hatasını ve oturma zamanını da küçülteceği ya da minimum yapacağı söylenebilir. Ancak hata fonksiyonunu minimum yapmak için literatürde birçok önerilerde bulunulmuştur.

Bunlardan ilk akla gelen hatanın integralinin minimum yapılmasıdır. Ancak hata fonksiyonu bazen pozitif bazen negatif değerler alabileceği için hatanın mutlak değerinin integralinin minimum yapılması kabul görmüştür.

Burada minimum sistem parametrelerine göre yapılır. Parametrelere göre I' nin minimum yapılmasına parametre optimizasyonu denir[Sarioğlu,1999].

Hatanın karesel toplamını hesaplayan ISE performans indeksi pratik ölçümlere kolayca adapte edilebilir. Karesel toplam analitik ve ölçümsel amaçlar için uygundur.

$$ISE = \int_0^t e^2(t)dt \quad (2.6)$$

Üst sınır hatanın halıcı hal durumuna eriştiği zamanı ifade eder. T genelde oturma zamanıdır.

Performans kriteri IAE ise hatanın mutlak değerlerinin toplamıdır. Bilgisayar simülasyonları için uygun bir kriterdir.

$$IAE = \int_0^t |e(t)| dt \quad (2.7)$$

Girişteki büyük başlangıç hatalarının etkisini azaltmak ve sistem cevabına etkisini geciktirmek için ITAE performans indeksi kullanılmalıdır.

$$ITAE = \int_0^t |e(t)| dt$$

(2.8)

Diğer bir performans kriteri olan ITSE ise kullanım amaçları ITAE ile aynıdır.

$$ITSE = \int_0^t te^2(t) dt$$

(2.9)

Performans indeksleri kontrol sistemlerinin analiz ve dizaynları için önemlidir[Dorf and Bishop, 2001].

2.2.3. Frekans Domeni Yaklaşımı

Bir sistemin frekans cevabı, sistemin sinüsoidal giriş sinyaline gösterdiği kalıcı durum cevabı olarak tanımlanır. Sistemin çıkış cevabı, geçici-durum hali ortadan kalktıktan sonra, sadece genlik ve faz bakımından yine bir sinüsoidal sinyal olmak üzere girişten farklıdır. Frekans cevabı yönteminde transfer fonksiyonunda 's' yerine 'jw' koyarak, kalıcı-durum transfer fonksiyonu elde edilebilir. Elde edilen transfer fonksiyonunun hem büyüklük hem de faz açısı olmak üzere iki unsuru vardır. Frekans cevabının hem zaman hem de s-domeni yaklaşımına üstünlükleri vardır. Sistem parametre değerlerinin değişim etkilerini belirlemek zaman domenin de çok zorken frekans domenin de kolay olması, Nyquistik kararlılık yoluyla bağlı kararlılığında incelenebilmesi bunlardan bazılarıdır. Geçici durum cevabı ile frekans cevabının bazı parametreleri arasındaki dolaylı ilişki biliniyorsa frekans cevabı tercih edilebilir.

2.3 Pid Kontrolörler ve Tasarım Yöntemleri

2.3.1 Pid Kontrolörler

Bir otomatik kontrol probleminde karşılaşılan en büyük sorun, sistemin sürekli hal değerine ulaşana kadar istenmeyen çıkış değerlerine ulaşarak fiziksel sınırlarını zorlamasıdır. Çoğu endüstriyel kontrol sistemi hala PID kontrolüne dayanmaktadır. PID kontrolörünün endüstride yüksek oranlarda kullanıldığı bilinmektedir. Yaygın olarak kullanımın sebepleri arasında;

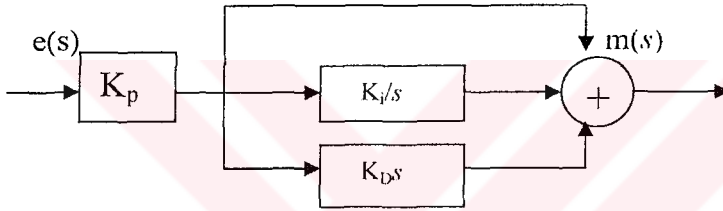
- a)Dayanıklıdır ve tasarımları basittir.
- b)Parametrelerinin ayarlanmalarına yönelik birçok metot geliştirilmiştir.

c)PID ile sistem cevabına ait parametreler arasındaki ilişki kullanıcılar tarafından en iyi şekilde kullanılmaktadır.

Sistemin matematiksel modeline ihtiyaç duymayan PID kontrolörler için çeşitli ayarlama yöntemleri endüstride sıkça kullanılmaktadır.1942’de Ziegler-Nichols adlı iki araştırmacı için öncü olmuş ve daha sonra Astrom-Hagglund kazanç-faz sınırı ayar metodu geliştirmiştir. Bu ayar metodu ise Nyquistik eğrisinin kritik kazanç (K_C) ve kritik frekans (w_C) kavramlarına dayanır.

2.3.2 Pid Parametrelerinin İncelenmesi

PID oransal-tümlevsel-türevsel kontrolör yapısı aşağıdaki şekil üzerinden açıklanacaktır.



Şekil 2.4. Genel PID yapısı

PID kontrolör çıkışı;

$$m(t) = K_p e(t) + \frac{K_i}{K_p} \int_0^{\infty} e(t)dt + \frac{K_d}{K_p} \frac{de(t)}{dt} \quad (2.10)$$

$$\frac{m(s)}{e(s)} = K_p \left[1 + \frac{1}{T_i s} + T_d s \right] \quad (2.11)$$

olarak elde edilir.

PID kontrolörü üç farklı kontrolün etkisi altında çalışmanın üstünlüğünü kullanır. İntegral etkisi, kalıcı durum hatasını sıfırlarken türev etkisi aynı kararlılık değeri için cevap hızını ve aşımı ayarlar. Böylece PID sıfır kalıcı durum hatası olan hızlı bir cevap sağlar. Burada K_p , T_i ve T_d parametreleri dengeli ayarlanırsa istenen cevabı sağlayan PID elde edilebilir. Katsayılar uygun ayarlanmazsa PID nin özelliklerinden yararlanılamaz.

PID parametrelerini incelerken genel transfer fonksiyonundan yararlanılacaktır[Yüksel,1997].

$$\frac{m(s)}{e(s)} = K_p \left[1 + \frac{1}{T_i s} + T_d s \right] \quad (2.12)$$

a) P-Kontrol : $T_i \rightarrow \infty$ ve $T_d \rightarrow 0$ halinde kontrolör yalnızca orantı etki ile çalışır. K_p nin ayarı ile kontrolör duyarlılığı değiştirilebilir.

b) PI Kontrolör : Orantı etkiye integral etki ilavesi ile elde edilen PI kontrolörün yapısı basit olup yaygın olarak kullanıldığı alanlar; basınç, seviye ve akış kontrol sistemleridir. İntegral etki denetlenen çıkış büyüklüğünde meydana gelebilecek kalıcı-durum hatalarını ortadan kaldırır. Denetlenen çıkış büyüklüğünde ortaya çıkan sapma sıfırlanmış olur.

c) PD Kontrolör : Türev etkisi kalıcı durum etkisini sıfırlayamaz ancak geçici durum davranışını iyileştirir. Türev etkisi, sisteme sönüm ilave ederek kararsız veya kararsızlığa yatkın sistemi daha kararlı hale getirebilir. Türev etkisini en büyük dezavantajı ise gürültü sinyallerini de kuvvetlendirebilmesidir.

d) PID Kontrolör : PI kontrolörünün tamamlayıcısı olarak türev etkisinin birlikte kullanıldığı yapı en çok sıcaklık, pH, yorgunluk, karışım ölçümlerinde ortaya çıkan ölü zaman gecikmelerinin telafisi için kullanılır. Çok küçük sönüm katsayısına sahip servomekanizmalarda PI kontrolör yeterli olamamaktadır. Bu durumda da türev ilavesi, sistemde fazla bir kararsızlık problemi yaratmadan K_p orantı kazancının yüksek tutulmasını sağlayarak sistemin kararsızlığını önlemiş olur. Böylece PID kontrolör etkisi ile bir taraftan kalıcı-durum hatası sıfırlanırken diğer taraftan da sistemin geçici-durum davranışı iyileştirilmiş olur.

2.3.3 Pid Tasarım Yöntemleri

PID kontrolörünü tasarlarken PD kısmının PI ile seri bağlandığı farz edilir. Tasarımda ilk olarak PD kontrolörün etki ettiği düşünülür. Zaman tanım bölgesinde kararlılık maksimum aşım ile ifade edilebilir. Başka bir yöntemde ise önce PI kontrolör tasarlanır, PD kısmı sonradan tanımlanır.

2.3.3.1. Zaman Domeni Tasarımı

a. PI Kontrolörler

PI kontrolörlerin kullanılmasıyla sistemlerde; maksimum aşım azalımı, sönümü iyileşmesi, yükselme zamanını artımı, BG bant genişliğini azalımı, kazanç payı, faz payı ve M_r ' yi iyileşmesi, yüksek frekanslı gürültülerin filtrelenmesi gözlenir.

PI kontrolörün açık çevrim transfer fonksiyonuna $s = K_I / K_p$ de bir sıfır ve $s = 0$ da bir kutup ekler. Sıfırın yeri uygun seçildiğinde kararlı hal hatasının yanı sıra sönümü de iyileştirebilir. $S = K_I / K_p$ sıfırını diğer baskın kutuplardan uzak ve orijine yakın seçilmelidir. $s = K_I / K_p$ sıfırı $G(s)$ in en yakın kutbuna göre çok küçük seçilmelidir. Böylece kök-yer eğrisinde orijin yakınlarında küçük bir daire oluşur.

$$G(s) = G_c(s) * G_p(s) = \frac{w_n^2 (K_p s + K_i)}{s^2 (s + 2\zeta w_n)} \quad (2.13)$$

Kararlı hal hatası kriterini sağlayacak şekilde K_I parametresi saptanır ve istenen sönüm oranını (ζ) veren K_p seçilir. K_I / K_p sıfıra çok yakın olduğundan transfer fonksiyonunda ihmal edilebilir.

b. PD Kontrolörler

Açık çevrim transfer fonksiyonu aşağıdaki gibi olan bir sistemi için ;

$$G(s) = G_c(s) * G_p(s) = \frac{w_n^2 (K_p + K_d s)}{s(s + 2\zeta w_n)} \quad (2.14)$$

$de(t)/d(t)$ hata işaretinin eğimini ifade ettiğinden, aslında sezgisel bir metottur. Eğimi bilmesi nedeniyle kontrolör hata yönünü sezer ve sistemi daha iyi kontrol eder. Lineer sistemlerde basamak cevaba ilişkin $e(t)$ ya da $y(t)$ ' nin eğimi büyük olursa yüksek aşım oluşur. Türevsel kontrol $e(t)$ 'nin ani eğimini ölçer, yüksek bir aşım olacağını tahmin eder ve aşırı aşım meydana gelmeden önce uygun bir düzeltme gayretinde bulunur. Kararlı hal hatası zamanla değişiyorsa sistemin hatasına etkide bulunur, eğer kararlı hal hatası sabit ise $de(t) / d(t) = 0$ olur. PD kontrolör sisteme herhangi bir katkıda bulunmaz[Hurma,1998].

2.3.3.2. Frekans Domeni Tasarımı

a. PI Kontrolörler

PI Kontrolörün transfer fonksiyonu frekans tanım bölgesinde incelemek için aşağıdaki gibi yazılır.

$$G_c(s) = K_p + \frac{K_i}{s} = \frac{K_i[1 + (K_p/K_i)s]}{s} \quad (2.15)$$

İstenen faz payını gerçekleştirmek için PI kontrolörü şu şekilde tasarlanır;

1. Kompanze edilmemiş sistemin ileri yol transfer fonksiyonu $G_p(s)$ 'in Bode diyagramı kararlı hal hatasını istenen değere getiren çevrim kazancı hesaba katılarak çizilir.
2. Kompanze edilmemiş sistemin faz ve kazanç payı Bode diyagramından bulunur. Belli bir faz payına ulaşmak için bu faz payına ilişkin yeni faz sınırı pay frekansı w_g Bode diyagramından bulunur. İstenen faz payını gerçeklenebilmesi için seçilen yeni w_g frekansında kompanze edilmiş transfer fonksiyonun genlik eğrisi x-ekseninde 0 dB' den geçmesi gerekir.
3. Kompanze edilmemiş transfer fonksiyonu genlik eğrisinin w_g ' yeni kazanç payı frekansında 0 dB' e indirilmesi için, PI kontrolörün w_g frekansındaki kazancına eşit miktarda zayıflatma yapması sağlanmalıdır. Bu amaçla K_p değeri aşağıdaki gibi bulunur.

$$|G_p(jw_g')|_{dB} = -20 \cdot \log_{10} K_p \text{ dB} \quad K_p < 1 \quad (2.16)$$

$$K_p = 10^{-|G_p(jw_g')|_{dB}/20} \quad K_p < 1 \quad (2.17)$$

K_p değeri saptandıktan sonra tasarımın tamamlanması için K_i değerinin seçilmesi gerekir. PI kontrolörün zayıflatma özelliği faz payına zarar verecek faz gerilemesi yapar. Eğer $w = K_i/K_p$ köşe frekansı w_g' nün çok altında yerleştirilirse, w_g' civarında P_i kontrolörün faz gecikmesi kompanze edilmiş sistemin fazında ihmal edilebilir düzeyde az etki eder. Diğer taraftan K_i / K_p değeri çok küçük seçilmemelidir. $K_i/K_p = w_g'/10$ rad / sn seçilmelidir.

4. Kompanze edilmiş sistemin Bode diyagramı istenen özelliklerin sağlanıp sağlanmadığını görmek için sınanır.

5. K_p ve K_i değerleri bulunur [Hurma, 1998].

b. PD Kontrolörler

Frekans domenin de PD kontrolör Bode diyagramının kolay çizilmesi için aşağıdaki şekilde yazılır.

$$G_c(s) = K_p + K_d s = K_p \left[1 + \frac{K_d}{K_p} s \right] \quad (2.18)$$

PD kontrolör faz ilerletmeli özelliği ile kontrol sisteminin faz payını iyileştirir. Fakat PD kontrolörün genlik karakteristiği, wg faz payı frekansını yüksek değere çeker. K_p/K_D köşe frekansını değiştirerek optimum çözüm bulunur.

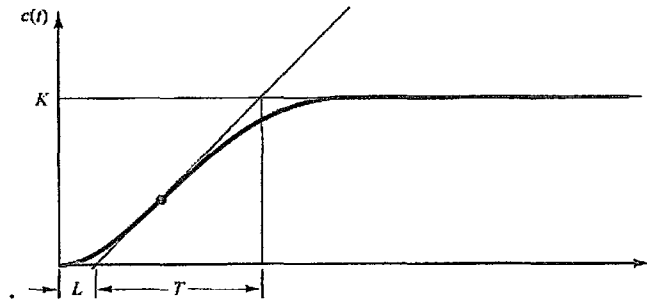
PD kontrolörler sönümü iyileştirir, maksimum aşımı azaltır, yükselme zamanını düşürür. Bant genişliğini artırır, kazan payı, faz payı ve M_r yi iyileştirir. Yüksek frekanslarda gürültüyü ön plana çıkarabilir. Az sönümlü ve kararsız sistemlerde etkili değildir. İntegral kısmı hata kalmadığında bile kontrol işareti üretmeye devam eder, bunun sonucunda sistemde salınımlar meydana gelir. Hatanın türevi alınarak kontrolöre hatanın sıfıra yakınlığı bildirilmelidir[Kuo,1999].

2.3.3.3. Ziegler-Nichols Kuralları

Ziegler-Nichols, verilen bir sistemin geçici cevap karakteristiğine dayanan oransal K_p , integral zamanını T_I ve türevsel zamanı T_D belirlemek için kurallar sunmuştur. Ziegler-Nichols tarafından sunulan iki metot bulunmaktadır. Bu metotlar halen kullanılmaktadır. Sistem dinamikleri bilinmediğinde kolaylıkla yerine getirilebilir[Zhuang,1992].

Birim Basamak Cevabı Metodu

İlk metotta bir sisteme ait birim basamak cevabı incelenmiştir.



Şekil 2.5. Ziegler-Nichols Birinci Metodu s-şeklindeki eğim cevabı

Eğer incelenen sistem integral yada baskın karmaşık kök içermiyorsa birim basamak cevabı şekildeki gibi olacaktır. Sistemin cevabı gecikme zamanı L ve zaman sabiti T ile ifade edilmiştir. Sistemin cevabını s noktasında kesen eğim çizgisinin tanjant değeri zaman eksenindeki T ve $c(t)$ değeri K ile ifade edilir. Yapılan çalışmalar sonucunda aşağıdaki sonuçlara ulaşılmıştır;

Tablo 2.1 Ziegler-Nichols Basamak Cevabı Metodu Sonuçları

Kontrolör Tipi	K_P	T_I	T_D
P	T/L	∞	0
PI	$0.9T/L$	$L/0.3$	0
PID	$1.2T/L$	$2L$	$0.5L$

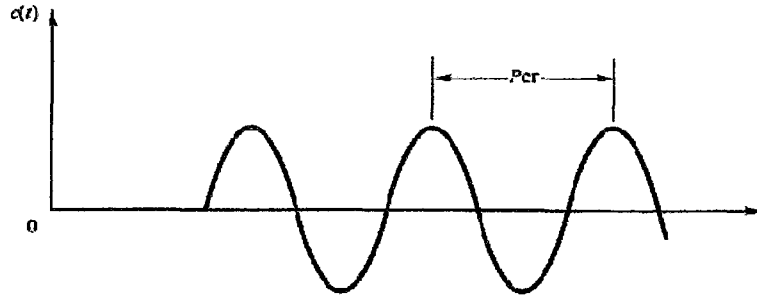
Ziegler-Nichols un önerdiği K_P , T_I , T_D değerleri tabloda belirtilmiştir. PID kontrolörün parametrelerini bulunan sonuçlara uygun yazarsak;

$$\begin{aligned}
 G_C(s) &= K_P [1 + 1/(T_I s) + T_D s] \\
 &= 1.2(T/L) [1 + 1/(2Ls) + 0.5Ls] \\
 &= 0.6T(s + 1/L)^2 / s
 \end{aligned}
 \quad (2.19)$$

Görüldüğü üzere PID kontrolör orijinde bir sifıra ve $s = -1/L$ de ise çift sifıra sahiptir.

b. Frekans Cevabı Metodu:

İkinci metotta ise $T_I = 0$ ve $T_D = 0$ a ayarlanır.



Şekil 2.6. Ziegler-Nichols İkinci Metodu, P_{cr} osilasyon periyodu

Oransal kontrol sıfır değerinden kritik değer olan osilasyon değeri K_{CR} ye kadar artırılır. Çıkışta osilasyon elde edilemezse bu metot kullanılamaz. K_{CR} nin oluştuğu periyot olan P_{CR} ye dayanarak K_P , T_I ve T_D elde edilir.

Tablo 2.2. Ziegler-Nichols Frekans Cevabı Metodu Sonuçları

Kontrolör Tipi\Parametreler	K_P	T_I	T_D
P	$0.5K_{Cr}$	∞	0
PI	$0.45 K_{Cr}$	$1/(1.2)P_{Cr}$	0
PID	$0.6K K_{Cr}$	$0.5 P_{Cr}$	$0.125 P_{Cr}$

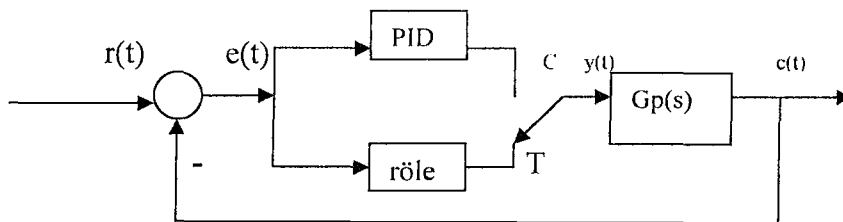
Ziegler-Nichols un önerdiği K_P , T_I , T_D değerleri tabloda belirtilmiştir. PID kontrolörün parametrelerini bulunan sonuçlara uygun yazarsak;

$$\begin{aligned}
 G_C(s) &= K_P [1 + 1/(T_I s) + T_D s] \\
 &= 0.6 K_{Cr} (1 + 1/(0.5 P_{Cr} s) + 0.125 P_{Cr} s) \\
 &= 0.075 K_{Cr} P_{Cr} (s + 4/(P_{Cr})^2 / s) \quad (2.20)
 \end{aligned}$$

Sonuç olarak PID kontrolör orijinde bir kutuba ve $s = -4/P_{Cr}$ de çift katlı sifıra sahiptir [Ogata, 1970].

2.3.3.4. Özuyarlamalı (Autotuning) Sistemler

Açık döngü sistemin Nyquist eğrisinde olan, negatif reel ekseni kesen kritik nokta bilgisi çoğunlukla PID kontrolör parametrelerinin belirlenmesinde kullanılır. Orjinal Ziegler-Nichols tasarımında kritik kazanç ve kritik frekans manuel olarak belirlenir. Bu metotta oransal bir regülatör bağlanarak güçlendirilmiş bir osilasyon elde edilinceye kadar kazanç yavaş yavaş yükseltilir. Bu bir dezavantajdır. Gürültü ve diğer faktörlerin karıştığı uzun zaman sabitleri yerine sistemin osilasyon genliğinin kontrolü zor olabilir. Astrom bu zorlukları yenmek için daha iyi bir metot önermiştir [Zhuang, 1992; Astrom, 1984; Astrom, 1988].



Şekil 2.7. Otomatik ayar sistemi blok diyagramı

Kritik kazanç ve kritik frekansı belirlemek için bir röle kontrolü, sistemle birlikte geri besleme döngüsüne bağlanır. Kontrolör ayar modunda(T) bir röle kontrolörü gibi kontrol modunda(C) ise bir PID gibi davranmaktadır. Ayar modunda hata sinyali $e(T)$ periyodik bir sinyaldir. Parametre K_C ve W_C tanımlama fonksiyonu metodu kullanılarak yaklaşık olarak osilasyonun birinci harmonik bileşenleri belirlenir. Osilasyonun genliği, röle çıkış seviyesi değiştirilerek kontrol edilebilir. Bu yüzden PID kontrolör ayarı mikrobilgisayarlar kullanılarak otomatik yürütülebilir. Pratikte osilasyon geçişleri arasındaki zamanı ölçerek kolaylıkla tespit edilir. Sistem ayar modundayken osilasyonun genliği, osilasyonun tepeden tepeye değerleri ölçülerek tahmin edilir. Eğer sistemde ilk harmonik baskın değilse ölçme sırasında bir filtreye ihtiyaç vardır. Sürecin transfer fonksiyonu bilindiğinde röle kontrolündeki limit saykıl çözümü, analitiksel, nümerik hesaplama ve bilgisayar simülasyonu olarak çıkarılabilir[Yüksel, 1997; Dumond,1985; Zhuang, 1992; Astrom, 1984; Astrom ,1988].

2.4. Genetik Algoritmalar

PID kontrolörün katsayılarının ayarlanmasında kullanılan matematiksel modeller de birçok kabuller ve ihmaller yapılır. Bu da kontrolörün performansını düşürür. Sistemi tam olarak temsil eden karmaşık modellerin de lineerleştirilmesi zor olduğundan kullanışlı değildir. Bununla birlikte Genetik Algoritmalar (GA) bir çok karmaşık optimizasyon problemlerinde başarılı sonuçlar vermektedir. Genetik algoritmaların başarılarının ana sebebi aday çözümlerin popülasyonunda çok yönlü bilgi alış verişi yapabilmeleridir. Kullanımdaki basitliği ve çözümlerde ki yeterliliği genetik algoritmalara olan ilgiyi artırmaktadır[Man vd.,1996;Chaiyaratana ve Zalzala,1997].

2.4.1 Genetik Algoritmalara Giriş

İlk defa Holland(1975)tarafından önerilmiştir[Goldberg,1989]. Darwin' in “en iyi olan yaşar(survival of the fittest)” prensibine dayalı olarak biyolojik sistemlerin gelişim sürecini simüle eder. Gecen otuz yıl boyunca popülasyonların potansiyel çözümlerinden ve evrim teorisinin prensiplerinden yararlanılarak problem çözümlerine büyük ilgi gösterilmiştir. Popülasyon içerisindeki bireylerin uygunluk değerleri, bazı seçim mekanizmaları ve genetik operatörler problemin çözümünde önemlidir.

Genetik Algoritmalar(GA) kelime anlamını doğal genetik terimlerinden almıştır. GA bireyleri açıklanırken bir populasyon içerisindeki kromozomlar kullanılmıştır. Genetik Algoritmalar(GA) stokastik bir arama yöntemidir.İyi uyum yapan bireyler yaşar ve daha fazla yavru sahibi olur. Düşük uyumlu bireyler az sayıda yavru sahibi olacak ve belki hiç olmayacaktır. Uyumlu bireylerin yavruları daha uyumlu bireyler oluştur ve zamanla populasyon içerisinde süper uyumlu bireylerin oluşacaktır.

GA optimizasyon problemlerinin çözümünde büyük çoğunlukla başvurulanan bir çözüm yöntemidir. Diğer bir çok yöntem tek bir noktanın takibiyle problem çözümünü araştırırken, GA bir populasyonun potansiyel çözümünden yola çıkarak araştırmasını sürdürür.

Sezgisel bir metot olduğundan dolayı verilen bir problem için optimum sonucu bulamayabilir, ancak bilinen metotlarla çözülemeyen yada çözüm zamanı problemin büyüklüğü ile üstel artan problemlerde optimala çok yakın çözümler vermektedir.[Liepins et all, 1990]

2.4.2 Genetik Algoritmaların Sağladığı Avantajlar

Genetik algoritmaların temel avantajı optimize etmeye çalıştıkları problemle ilgili herhangi bir bilgiye ihtiyaç duymamalarıdır. Sistemin matematik model bilgilerini gerektirmez. Karmaşık matematik hesaplamaları yerine yalnızca giriş-çıkış bilgilerine ihtiyaç duyar.Bu avantajından dolayı karmaşık problemlerin optimizasyonu için tercih edilebilir bir yöntemdir.Büyük parametre ve yapısal belirsizlikleri tolere edebilir.Yine uygun performans indeksi seçimi ile sistemin arzu edilen dinamik davranışları ve kararlılığı elde edebilir[Zuo,1995].

Oluşturulan başlangıç populasyonu girdiği GA döngüsü sonucunda en uygun çözümü kolayca bulabilir. GA evrim teorisinin steplerini kullanarak çalışır. Bir veri grubundaki özel bir veriyi bulmak için kullanılır. GA klasik yöntemlerin çok uzun zamanda bulabileceği çözümü yeterli doğrulukla çok kısa zamanda bulabilir. Optimal değeri bulmayı garanti etmez, fakat uzun nesiller sonucunda en uygun değere çok yakın çözümler bulmasını sağlar. GA'lar özellikle araştırmacının kesin, konu uzmanı olmadığı zamanlarda araştırmacıya o alanda bilgi edindirmede başarılıdır [Valenzuela,1995].

2.4.3. Genetik Algoritmaların Kullanım Alanları

Başlangıç da sürekli doğrusal olmayan optimizasyon problemine uygulanan GA, sonraları gezgin satıcı, karesel atama, yerleşim, ders/sınav programı hazırlanması gibi kombinyonsal optimizasyon problemlerine de başarıyla uygulanmıştır[Liepins et all,1990]. Ayrıca GA lar tasarım, ekonomi ve finans, haberleşme ağları, tahmin(hava, deprem, at yarışı vb.),suç şüphelilerinin teşhisi, müzik besteleme, ulaştırma vb gibi bir çok alanda kullanıldığı yüzlerce çalışma mevcuttur. Genel olarak GA lar matematiksel analizi zor olan, geleneksel araştırma yöntemleriyle çözilemeyen karmaşık yapılarda kullanılırsa başarılı sonuçlara ulaşla bilinir.

GA lar global çözümü garanti etmezler ancak kabul edilebilir hızla kabul edilebilir ölçüde en iyi çözümler bulunmasında genel olarak başarılıdır. Belirli problemlerin çözümünde bazı özel teknikler varsa bunların hem sonucun doğruluğu, hem de hız açısından GA lara göre daha iyi işlemesi olasıdır. GA nın esas alanı bu tekniklerin olmadığı alanlardır. Mevcut tekniklerin iyi işlediği yerlerde bile, bu teknikler GA larla birleşerek ilerlemeler sağlanmıştır[Cantoni vd.,1999].

2.4.4. Genetik Algoritmaların Diğer Klasik Yöntemlerden Farklılıkları

GA temel özellikleri aşağıdaki dört maddede toplanabilir.

1. GA parametrenin kodlanmış halleriyle çalışır, parametrelerin kendileri ile değil.
2. GA tek bir nokta yerine noktalar topluluğunda hareket eder.
3. GA türev ve daha farklı bilgileri değil sadece amaç fonksiyonu değerini kullanır.
4. GA deterministlik değil olasılık kurallarına dayanır.

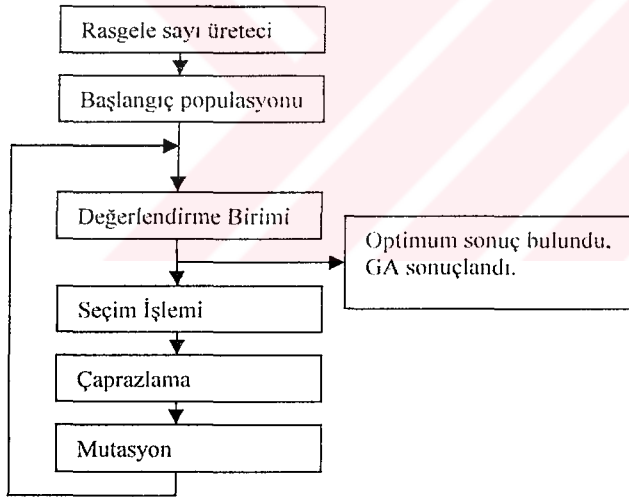
2.4.5. Genetik Algoritmaların Temel Yapısı ve İşleyişi

GA temelleri biyolojiye dayandığı için kullanılan terimler de bundan etkilenmektedir. Canlı organizmalarda kromozom organizmanın yapısını oluşturan temel bilgileri taşır. Her kromozom gen adı verilen özel kodlanmış yapılardır. Karmaşık yapıları bit dizileri halinde kodlamıştır. Oluşan her dizi çözüm uzayındaki popülasyonun bir üyesidir. Bu diziler GA süreçleri sonrasında uygun kriterlerden geçirilirse ortalama kalitesi çok yüksek yeni dizilerden oluşan bir popülasyon ortaya çıkar. Yeni popülasyon

içerisinden en uygun çözümü belirlenen kriterlere göre bulur. Bu kriter sağlanana kadar genetik süreç tekrar eder. Bu şart tekrar sayısının belirlenen bir sayıya ulaşması veya belirlenen bir başarımla değerine ulaşması olabilir[Goldberg, 1989 ; Biegel ve Davern ,1990 ; Lawrence ,1990].

GA populasyon olarak adlandırılan mümkün çözümlerin kodlandığı dizilerin, biyolojik evrim mekanizmasını taklit eden bir süreç sonucunda uygun değerlere ulaşmasını aşağıdaki algoritma yapısı içinde açıklanabilir. Basit bir GA gösterimi şekildeki gibidir[Davis,1991].

Algoritmanın yapısından da görüldüğü gibi GA ilk aşamada belirlenen kromozom gösterimine (representation) uygun olarak ilk populasyonu oluşturur. Populasyondaki her kromozomun bir uygunluk değeri vardır. Uygunluk değerini(fitness value) belirlenen uygunluk fonksiyonu bulur. Uygunluk değeri bir sonraki populasyona hangi bireylerin taşınacağını tespit eder. Bir kromozomun gücü uygunluk değeriyle tespit edilir ve iyi bir kromozom problemin yapısına göre maksimizasyon problemi ise yüksek, minimizasyon problemi ise düşük uygunluk değerine sahiptir[Michalewicz,1992].



Şekil 2.8.GA'nın akış diyagramı

GA'nın her iterasyonunda uygunluk değerlerine göre seçim yapılması, genetik operatörlerin kullanılması, uygunluk değerinin hesaplanması işlemleri tekrarlanır. Değerlendirme birimi iterasyona devam edilip edilmeyeceğine karar verir. İterasyon optimum değeri bulunca yada kullanıcı tarafından belirlenen belirli iterasyon sayısına ulaşıncaya tamamlanır.

Bu açıklamalar doğrultusunda, herhangi bir problemin çözümünde kullanılan bir GA'nın aşağıdaki bileşenlerden oluştuğu görülmektedir.

- Populasyonu oluşturacak bireylerin kromozom olarak gösterimi
- Başlangıç yığınının oluşturulması
- Dizilerin uygunluk değerinin belirlenmesi
- Yeniden üretim işlemi için seçim mekanizmasının çalıştırılması
- Yeni bireylerin elde edilmesi için genetik operatörlerin kullanılması
- Kontrol parametreleri olarak çaprazlama ve mutasyon olasılığının kullanılması

2.4.6. Kromozomların Kodlanması

GA ların başarılı sonuçlar verebilmesi için kromozomların algoritma sürecine girecek uygun parametreler haline getirilmesi gerekmektedir. Kromozomlar belirli bir çözüm hakkında tüm bilgileri içerir. Kodlama mekanizması problemin yapısına bağlıdır ve algoritmanın performansında önemli yer tutar. Değişik kodlama çeşitleri vardır. En çok kullanılan kodlama yöntemlerin başında ikili kodlama ve permütasyonlu kodlama gelmektedir. Bununla birlikte Gray kodlama gibi diğer metotların bazı durumlarda daha etkili olduğu gözlemlenmiştir[Caruana,1988].

2.4.6.1 İkili Kodlama

En yaygın kodlama yöntemidir. Kromozomlar sıfır ve birlerden oluşan dizilerdir. Her parametre n bitten oluşur. Dizinin uzunluğu parametre yada parametrelerin alt ve üst sınırları arasındaki tüm sonuçları temsil edecek şekilde olmalıdır. Alt ve üst sınır U_{min} ve U_{max} olarak verilen bir parametre için dizi uzunluğu için

$$l = (U_{max} - U_{min}) / (2^L - 1) \quad (2.21)$$

eşitliğinden hesaplanır. Birden fazla parametreye sahip bir fonksiyon için dizi uzunluğu ise, her parametre için belirlenen dizi uzunluklarının toplamına eşittir[Golberg,1989].İkili düzende kodlama çok sık kullanılmasına rağmen problemin yapısına göre kullanılmasında bazı sakıncalar doğurabilir. Gezgin satıcı, çizelgeleme, karesel atama gibi kombinatoryal eniyileme problemlerinde ikili düzende kodlama, araştırma uzayını tam olarak temsil edememektedir. Bu nedenle literatürde, gerçel sayı

kodlama ve permütasyon kodlama türleri de çok sık kullanılmaktadır [Michalewicz,1992]

2.4.6.2. Permütasyonlu Kodlama

Sıralama problemlerinde permütasyonlu kodlama kullanılır. Bu kodlamada, her kromozom dizideki bir sırayı temsil eden sayılardan oluşur. Örneğin, gezici satıcı probleminde “ verilen şehirler arasındaki uzaklıklar bilinirken, bütün şehirleri birleştiren minimum yol uzunluğu ne olmalıdır?” sorusuna cevap aranmaktadır. Burada kromozomlar, şehirlerin hangi sırada ziyaret edilebileceğini gösteren permütasyonlar şeklinde kurulur[Goldstein ,1991;Valenzuela ,1995].

2.4.7. Başlangıç Populasyonunun Oluşturulması

Başlangıç popülasyonunu oluştururken rasgele sayı üretici kullanılır. Problem için önerilen belirli çözüm aralığı varsa başlangıç populasyonu bu aralıktan oluşturulabilir. Bu yöntem optimal sonuca giderken sistemin daha hızlı davranmasını sağlayacaktır. Başlangıç popülasyonundaki kromozomların kaç bitle temsil edileceği ve başlangıç popülasyonundaki birey sayısı da sistemin hızını etkileyeceğinden dikkat edilmesi gereken parametrelerdendir.

2.4.8. Uygunluk Değerlerinin Hesaplanması

Uygunluk değeri, yeni popülasyona taşınacak kromozomların belirlenmesinde kullanılan bir araçtır. Bu nedenle, algoritmanın her iterasyonunda popülasyondaki kromozomların uygunluk değeri hesaplanır[Dengiz,1998]. Yüksek uygunluk değerine sahip bireyler daha büyük bir olasılıkla seçilmelidir.

Her bir çözümün kalitesinin belirlenmesinde kullanılan uygunluk değeri hesaplanması GA’ nın çalışma performansında önemlidir. Ancak GA’nın ilerleyen iterasyonlarında bireylerin uygunluk değerleri birbirlerine yakınsamaya başlar. Bu da bir sonraki aşamada uygunluk değerlerine göre seçim işlemi olanaksız kılar.Bu durumda GA’ nın optimal bir sonuca ulaşması zordur. Bu sorunu çözmek için uygunluk değeri işlemi normalizasyon işlemi takip etmelidir. Normalizasyon işleminde en yaygın olarak kullanılan teknik ölçeklendirme işlemidir[De Jong,1975].

2.4.9. Yeniden Üretim İşlemi

Rasgele oluşturulan ilk popülasyondan yada bir önceki iterasyonda oluşturulan popülasyondan yeni popülasyona bireylerin seçilmesidir. Bir sonraki popülasyona taşınan bireyler en güçlü genetik yapıya sahip dizilerdir. Bireyler uygunluk değerlerinin yüksekliğiyle orantılı olarak bir sonraki popülasyonda yer alırlar. Kalitesi yüksek olan kromozomların bir sonraki iterasyonda yer alma şansı yükselirken daha zayıf bireylerin popülasyondan kaybolması mantığına dayanır. Bu yapı döngü içerisinde değerlendirme fonksiyonu ve seçim mekanizması ile temsil edilmektedir.

2.4.9.1. Uygun Bireylerin Seçilmesi

Rastasal olarak bir ilk popülasyon oluşturulduktan sonra GA'nın parametrelerine ve sistemin yapısına bağlı olarak yeni bir popülasyon oluşturmak için seçim yapılmalıdır. Yüksek uyumluluğa sahip bireyler daha büyük olasılıkla seçilir. Seçim operatöründen bunu yapması beklenir. Böylece seçim arama uzayının daha başarılı bölgelerinde arama yapmaya yönlendirilir. Sonraki kuşak için, yavru meydana getirecek bireylerin kaç tane birey oluşturacağı da uygunluk değerine göre belli olur. Oluşturulan popülasyondan seçilen rasgele bireyler seçilir. Seçim yapılırken popülasyonun çeşitliliği ve seçicilik baskısı dikkat edilmesi gereken etkenlerdir. Seçicilik baskısı yüksek olduğunda popülasyondaki çeşitlilik azalır ve GA'nın zamansız yakınsamasına neden olur. Bunun terside doğrudur. Bu nedenle tercih edilen seçim mekanizmasının popülasyon çeşitliliğini, zamansız yada etkinliği azaltan yakınsamalara neden olabileceği göz önüne bulundurulmalıdır. Bu faktörler arasında denge kurulmalıdır.

GA literatüründe çeşitli seçim planları önerilmiştir. Holland'ın orijinal GA' sında popülasyon seçimi, en iyi uyum yapanın yaşaması ilkesine dayanır. GA da uygun bir dizi çok sayıda yavru elde eder ve bunun sonucunda, sonraki kuşakta daha yüksek bir yaşama şansına sahip olur. Mevcut popülasyondan bireyleri uygunluk değerlerine göre seçen yöntemler, uyuma orantılı seçim yöntemleri grubu olarak adlandırılır[Yeniay,1999].

Golberg ve Deb(1991) in dört ana sınıfta toplamış olduğu seçim mekanizmaları ve elitist seçim mekanizmaları literatürde çok sık kullanılan seçim mekanizmalarındandır.

2.4.9.1.1. Orantılı Seçim Mekanizmaları

Populasyondaki her bireyin seçilme olasılıkları belirlenir ve bu olasılıklar kullanılarak bir sonraki populasyon oluşturulur. Rulet çemberi, Yerine Koymadan Stokastik Örneklem, Kalanı Stokastik Örneklem bu mekanizmaya örnektir.

a. Yerine Koyarak Stokastik Örneklem(Rulet Çemberi)

Bu metot ilk defa Holland tarafından ortaya çıkarıldı. GA da kullanılan en eski seçim metodudur. Seçim metotları her bir kromozomun bir sonraki populasyonda olma ihtimaliyle orantılı kopya sayısını belirler. Bir kromozomun bir sonraki nesle devamı uygunluk değeri ile orantılıdır denebilir. Populasyon içerisindeki her bir bireyin seçilme olasılığı, bireyin uygunluk değerinin populasyondaki uygunluk değerlerinin toplamına bölünmesiyle ifade edilir. Bu ifade

$$p_i = \frac{f_i}{\sum_{i=1}^n f_i} \quad (2.22)$$

eşitliği ile hesaplanır. Rulet seçim mekanizması ismini bireylerin uygunluk değerleriyle orantılı olarak bir çemberin dilimlerini oluşturulmasına benzetilmesinden almıştır. Bu metotta çember N eşit aralığa bölünür. Çemberdeki i.aralık i. kromozomu temsil eder ve bu aralığın genişliği bu dizinin seçilme olasılığına eşittir. Bu durumda çemberdeki aralık genişliklerinin toplamı 1'e eşittir. Seçim aşamasında çember N defa çevrilir. Her çevirimde 0-1 arasında sayılar üretilir. Üretilen sayının düştüğü aralıktaki dizi yeni populasyona kopyalanır [Goldberg,1989].

Rulet çemberinde stokastik hatalar nedeniyle bir bireye pay edilen yavru sayısı, beklenen yavru sayısından önemli ölçüde farklı olabilir. İterasyon sayısı arttıkça örneklem hatası da artmaktadır. Rulet çemberi seçiminde bu örneklem hatası nedeniyle, arama uzayın farklı noktalarına kayabilir. Bu da zamansız yakınsamalara neden olmaktadır. Bu sorunu kaldırmak için yeni yöntemler önerilmiştir.

b. Yerine Koymadan Stokastik Örneklem

Bu yöntem rulet çemberi seçiminin stokastik hatalarını azaltır. İfade f_i i. bireyin uyum değerini, $\sum_{i=1}^n f_i$ populasyonun toplam uyumunu ve $f = \sum_{i=1}^n \frac{f_i}{N}$ ise populasyonun ortalama uyum değerini göstermektedir. Her birey için f_k / f beklenen yavru değeri

hesaplanır. Bu sayı, birey çaprazlamaya her seçildiğinde 0.5, çaprazlama olmadan üremeye seçildiğinde ise bir azaltılır.

c. Kalanı Stokastik Örneklemeye

Bir sonraki popülasyonda yer alacak birey sayılarını bilinen yöntemlerle hesaplar. Tam sayılı kısmını bireye atar. Popülasyon genişliğine ulaşmadıysa, beklenen değerlerin kesirli kısımlarından yararlanır. Yerine koyarak kalanı stokastik örneklemede, beklenen değerlerin kesirli kısımları, rulet çemberi seçim yönteminde bireylere pay edilen ağırlıkları hesaplamak için kullanılır. Yerine koymadan kalanı stokastik örneklemede, beklenen kopya sayısına tam sayılı kısım kesin olarak alınır. Kesirli kısım olasılığıyla da bu kısım popülasyona birey olarak katılır.

2.4.9.1.2. Sıralı Seçim Mekanizması

Baker (1985) tarafından önerilmiştir [Goldberg, 1989; Goldberg, Deb, 1991] . Uyumla orantılı seçimin dezavantajlarını yok etmek için önerilmiştir. Popülasyondaki bireyler uygunluk değerlerine göre en iyiden en kötüye doğru sıralanır. En iyiden başlamak üzere bir azalan fonksiyon yardımıyla dizilere kopya sayısı atanır ve orantılı seçim mekanizmalarından birisi kullanılarak yeni popülasyon elde edilir.

2.4.9.1.3. Turnuva Seçim Mekanizması

Popülasyon içerisinde rastgele olarak bir grup kromozom seçilir. Grup içerisindeki en iyi uygunluk değerine sahip kromozom yeni popülasyona kopyalanır. Bu işleme popülasyon genişliğine ulaşmaya kadar devam edilir. Genellikle grup genişliği ikidir [Goldberg, 1989; Goldberg, Deb, 1991].

2.4.9.1.4. Denge Durum Seçim Mekanizması

Sıralı seçim mekanizması kullanılarak seçilen bir yada iki bireye genetik operatörler uygulanır. Elde edilen yeni diziler mevcut popülasyondaki uygunluk değeri en düşük kromozomlar yer değiştirilerek yeni popülasyon oluşturulur.

2.4.9.1.5. Elitist Seçim Mekanizması

Kenneth De Jong tarafından önerilen bu yöntemle; popülasyon içerisindeki en iyi bir yada iki bireyi sabit tutarak geri kalan bireylerin mevcut seçim mekanizmalarından

birini kullanarak yeni popülasyonun oluşturulması amaçlanır. Yeni popülasyona önceki popülasyondan en az bir en iyi birey değişmeden aktarılır. Böylece en iyi birey örnekleme hatasına yada genetik operatörlerin işlemlerine girmeden bir sonraki popülasyona aktarılır.

2.4.10. Genetik Operatörler

2.4.10.1. Çaprazlama Operatörü

Çaprazlama operatörü kullanarak popülasyon içerisinde seçilen iki ayrı dizinin belirli bölümleri karşılıklı değiştirilerek farklı çözümler arasında bilgi değişimi sağlanmış olur. Böylece GA arama uzayının farklı yerlerine yönlendirilmiş olur. Çaprazlama sonucunda iki dizinin iyi özellikleri birleştirilerek daha iyi bir dizi oluşturulması amaçlanır. En çok kullanılan çaprazlama türleri tek nokta çaprazlama(single point crossover), iki nokta çaprazlama(two-point crossover), çok nokta çaprazlama (multi-point crossover) ve uniform(uniform crossover) çaprazlamadır.

2.4.10.1.1. Tek nokta çaprazlama

Çaprazlama noktası olarak 1 ile L-1 arasında bir nokta rastsal olarak seçilir. İki dizinin belirlenen çaprazlama noktasından sonraki kısımları karşılıklı olarak yer değiştirilir [Goldberg,1989]. Şekilde ki X ve Y çaprazlamaya giren kromozomlar olmak üzere, X' ve Y' ise çaprazlama sonrasındaki yeni oluşan kromozomları temsil etmektedir.

Tablo 2.3. Tek Noktalı Çaprazlama Operatörü

Kromozom\Bit	1	2	3	4	5	6	7	8	9	10	11
X	0	1	0	1	1	1	1	1	0	1	1
Y	1	1	1	0	0	0	1	0	1	0	1
X'	0	1	0	1	1	0	1	0	1	0	1
Y'	1	1	1	0	0	1	1	1	0	1	1

Örnekte çaprazlama noktası olarak 5 seçilmiştir. 5.bitten sonraki bitler her iki kromozomda karşılıklı olarak yer değiştirilmiştir.

2.4.10.1.2. İki Noktalı Çaprazlama Operatörü:

İki noktalı çaprazlama operatörü kullanılırken iki farklı çaprazlama noktası seçilir. Bu aralık da kalan bitler yer değiştirilerek yeni kromozomlar elde edilir[Lawrence Davis,1991].

Tablo 2.4. İki noktalı Çaprazlama Operatörü

Kromozom\Bit	1	2	3	4	5	6	7	8	9	10	11
X	0	1	0	1	1	1	1	1	0	1	1
Y	1	1	1	0	0	0	1	0	1	0	1
X'	0	1	0	1	1	0	1	0	0	1	1
Y'	1	1	1	0	0	1	1	1	1	0	1

Şekilde 4 ve 8 numaralı bitler çaprazlama bitleri olarak seçilmiştir. Bu aralıktaki bitler karşılıklı olarak yer değiştirilmiştir.

2.4.10.1.3. Uniform Çaprazlama Operatörü

İlk defa Syswerda(1989) tarafından önerilmiştir. Öncelikle geçici olarak ikili bir dizi oluşturulur. Yeni diziler elde edilirken geçici diziden yararlanılır. Geçici dizinin 1 leri ve sıfırlarına göre yeni diziler elde edilir.

Tablo 2.5. Uniform Çaprazlama Operatörü

Kromozom\Bit	1	2	3	4	5	6	7	8	9	10	11
X	0	1	1	1	1	0	0	0	1	0	1
Y	1	0	0	0	0	0	1	1	1	0	0
Geçici Dizi	1	1	1	0	0	0	1	1	1	0	0
X	0	1	1	0	0	0	0	0	1	0	0
Y	1	0	0	1	1	1	1	1	1	0	1

Birinci kromozomdan çaprazlama sonrasında oluşacak kromozomu elde etmek için; geçici dizinin 1 olduğu değerler için birinci kromozom değerleri, geçici kromozomun değerlerinin sıfır olduğu değerler içinse ikinci kromozomun bitleri kullanılarak çaprazlama sonrasında oluşacak yeni kromozom elde edilir. Benzer çaprazlama mantığıyla ikinci kromozomda çaprazlama sonrasında oluşur.

2.4.10.1.4. Mutasyon Operatörü

Mutasyon, GA'nın ikinci dereceden rol oynar. GA'da mutasyon operatörü küçük bir olasılıkla uygulanır. Kromozom içerisindeki bitlerden bazıları rastsal olarak değiştirilir ve popülasyonda yeni kromozomların elde edilmesi sağlanır[Goldberg,1989]. Popülasyona mutasyon p_m olasılıkla uygulanır ve kromozomdaki bit bir ise sıfır, sıfır ise bir yapılır. X kromozomun 3. ve 7. bitleri mutasyona uğratılıp popülasyona yeni bir birey olarak Y kromozomu girmiştir.

Tablo 2.6. Mutasyon Operatörü

Kromozom\Bit	1	2	3	4	5	6	7	8	9	10	11
X	1	1	1	0	0	1	0	0	0	0	0
Y	1	1	0	0	0	1	1	0	0	0	0

GA'nın son iterasyonlarına doğru popülasyondaki kromozomlar birbirlerine benzemeye başlarlar. Bu durum çaprazlanan operatörünün bulunduğu kromozomlarında birbirine benzemesine neden olur. Mutasyon operatörü GA'da bu noktadan itibaren devreye girer ve popülasyona dışardan bitler sokarak arama uzayında farklı noktalarda arama yapmasına olanak sağlar. Her popülasyona mutasyon operatörü p_m olasılıkla uygulanır.

2.4.10.11. Genetik Algoritmanın Kontrol Parametreleri

GA'ların işleyişi, arama uzayından daha önce örnekleşmiş olan bölgelerin işletilmesi ve bu uzaydan yeni bölgelerin araştırılmasının dengeli birleşimi olarak düşünülebilir. GA'nın performansını kontrol eden bu denge, kontrol parametrelerinin doğru seçilmesi ile belirlenmektedir[Yeniay,1999]. GA'nın işleyişini kontrol etmek için kullanılan parametreler aşağıda açıklanmıştır.

2.4.10.11.1. Popülasyon Büyüklüğü (N)

Popülasyon genişliğinin küçük olması çözüm uzayını etkilediği için GA'nın performansını azaltır. Çözüm uzayının küçük olması zamansız yakınsamaya neden olur. Popülasyon genişliğinin büyük olması ise çözüm uzayında yeterli aramaya olanak

sağladığı halde, her itersayonda daha geniş bir populasyon üzerinde işlem yapıldığı için çalışma zamanını artırır.

2.4.10.11.2. Çaprazlama Oranı(p_c)

Seçim işlemi ile oluşan populasyondaki kromozomlara her iterasyonda kaç tanesinin çaprazlamaya gireceğinin belirler. Her yeni populasyonda $p_c \times N$ tane kromozoma çaprazlama uygulanır. Düşük çaprazlama aramanın çok yavaş, büyük çaprazlama oranı ise hızlı populasyon değişikliğine neden olur.

2.4.10.11.3. Mutasyon Oran(p_m)

Seçim işlemi sonrasında elde edilen yeni populasyondaki her kromozomun her elemanı p_m mutasyon oranında mutasyona uğrar. Her iterasyon sonucunda $p_m \times N \times L$ (L , her bir kromozomda mutasyona uğrayan bit adeti) adet bit mutasyona uğrar.

2.4.10.11.4. Seçim Mekanizmaları

Rastsal olarak oluşturulan ilk populasyondan sonra seçim mekanizmaları devreye girerek yeni populasyonu oluşturur. Bu konu daha önce açıklandığı için tekrar üzerinde durulmamıştır.

2.4.10.11.5. Ölçeklendirme Fonksiyonu Faktörü

Son iterasyona kadar populasyondaki değişkenliğin devamı ölçeklendirme fonksiyonları ile sağlanır. Kontrol parametrelerinin en iyi kombinasyonları için yapılan çalışmalar vardır. İlk çalışma DeJong(1975) tarafından yapılmıştır(Goldberg,1989) . DeJong(1975), populasyon genişliğinin, çaprazlama ve mutasyon oranlarının etkilerini 5 farklı özellikteki problem üzerinde incelenmiştir. Populasyon 50 ile 100 arasında olduğunda, çaprazlama oranı 0.60 ve mutasyon oranı 0.001 alındığında GA performansının iyi olduğu görülmüştür[Altıparmak,1996].

2.4.10.11.6. Nesil Ayrılığı(G)

Her iterasyonda populasyonda meydana gelecek değişimin yüzdesini kontrol eder. Böylece populasyonun iterasyonlar boyunca değişimi nesil ayrılığı katsayısı ile belirlenmiş olur. Bu durumda, t . iterasyondaki populasyonun $N \times (1-G)$ adet kromozomu, $(t+1)$. iterasyonda yaşayacaktır.

2.4.10.11.7. Bitirme Koşulu

GA da üretim-değerlendirme-seçim süreci, kullanıcı tarafından belirlenen iterasyon sayısına ulaşınca yada populasyonun ortalama uygunluk değeri popülasyondaki en iyi dizinin uygunluk değerine önceden belirlenen oranda yaklaşıncaya kadar devam eder[Goldberg,1989].

2.4.10.11.8. Ölçeklendirme Fonksiyonu

GA nın ilk birkaç iterasyonundan sonra elde edilen popülasyonda uygunluk değeri yüksek kromozom bulunmaktadır. Bu diziler yüksek uygunluk değerlerinden dolayı bu uygunluk değerleri etrafında zamansız bir yakınsama olabilmektedir. Son iterasyonlarda da kromozomların uygunluk değerleri birbirine çok yakınlaştığı için iyi kromozomların korunması zorlaşmaktadır. Orantılı seçim mekanizmalarında bu sorunu ortadan kaldırmak için ölçeklendirme fonksiyonu kullanılmalıdır. Literatürde 3 farklı ölçeklendirme fonksiyonu kullanılmaktadır. Kullanılan ölçeklendirme fonksiyonlarından biri doğrusal ölçeklendirme fonksiyonudur. Bu metotta kromozomların uygunluk değerleri doğrusal bir fonksiyon kullanılarak yeniden düzenlenir.

$$f'_i = a \times f_i + b \quad i=1,2,3,\dots,N \quad (2.23)$$

Burada f ham uyum f' ise ölçeklendirilmiş uyumdur. Fonksiyonda a ve b katsayıları f_{ort} ortalama ölçeklendirilmiş uyumu, f_{ort} ortalama ham uyuma eşit olacak şekilde seçilmelidir. Böylece mevcut yığında ortalama uygunluk değerine sahip dizilerin yeni yığında birer kopyalarının olmasını sağlamış olur.

Ayrıca mevcut yığında maksimum uygunluk değerine sahip dizinin kopya sayısı kontrol edilmelidir. Bunun için ölçeklendirilmiş maksimum uygunluk değeri

$$f'_{max} = c \times f_{ort} \quad (2.24)$$

eşitliğinden hesaplanabilir. Goldberg(1989) c değerini 1.2 ile 2 arasında seçebileceğini belirtmiştir. Doğrusal fonksiyon kullanıldığında negatif değerlerden kurtulabilmek için Forrest(1985), yığının varyant bilgisinin kullanılmasını önermiştir. Sigma kesim planı

olarak adlandırılan ölçeklendirme modelinde hesaplamalar için aşağıdaki fonksiyon kullanılır.

$$f_i' = f_i - (f_{ort} - c \times \sigma) \quad i=1,2,3,\dots,N \quad (2.25)$$

Bu eşitlik de c sabiti 1 ile 3 arasında seçilmesi önerilmektedir. Forrest(1985) bu yaklaşımda elde edilen negatif değerlerin 0' a eşitlenmesini önermektedir. Negatif değerlerin üretilmediği ölçeklendirme fonksiyonu ise üs yaklaşımı modelidir. Aşağıdaki eşitlikten yararlanılır;

$$f_i' = f_i^k \quad i=1,2,\dots,N \quad (2.26)$$

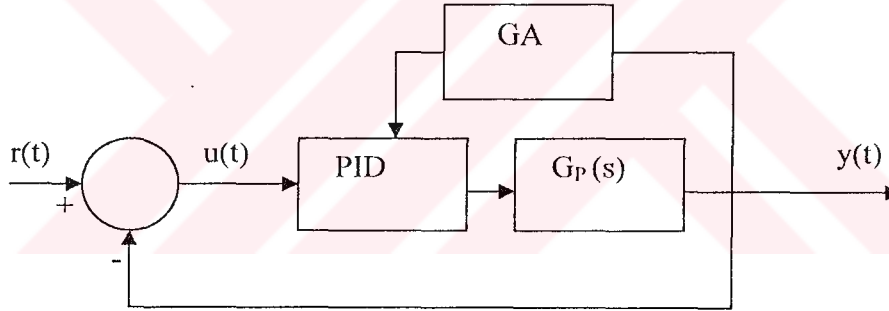
k sabiti problemin yapısına bağlı olup Gillies çalışmalarında k sabitini 1.005 olarak almıştır [Goldberg 1989; Michalewicz 1992].

3.GENETİK ALGORİTMALAR KULLANILARAK PID KATSAYILARININ OPTİMİZE EDİLMESİ

PID tasarımı yapılırken uygulama mühendislerinin elle yaptıkları düzenleme işlemi hem zaman almakta hem de doğru sonuçlar vermemektedir. Bu tez çalışmasında PID katsayılarının optimizasyonu Genetik Algoritmalar kullanılarak sağlanmıştır. Matlab dilinde yazılan Genetik Algoritma sürecinde performans analizi yapılırken, ISE, çıkıştaki hatanın karesel toplamının minimum olması esas alınmıştır. Hata fonksiyonu $e(t)$ nin minimum yapılmasının aşım, yükselme zamanı, gecikme zamanı ve yerleşme zamanını da minimum yapacağı söylenebilir[Sarıoğlu, 1999].

3.1 Pid Optimizasyon Genel Yapısı

PID kontrolörün sisteme bağlantısı ve Genetik Algoritmanın optimizasyon işlemi için sistemle etkileşimi aşağıdaki şekil üzerinden açıklanacaktır.



Şekil 3.1. GA larla PID Optimizasyon Genel Yapısı

Yukarıdaki yapının çalışması şu şekilde olmaktadır;

Genetik Algoritmanın her iterasyonundaki her birey için ürettiği k_p , k_i ve k_d parametrelerinden sistemin birim basamak cevabına bakılarak ISE performans kriteri hesaplanır. Genetik sürecin bitirme koşulunun sağlanması sonucunda elde edilen minimum toplam karesel hatayı veren bireye ait k_p , k_i ve k_d program tarafından seçilir. Programın çalışmasını kod yapısına göre açıklamak gerekirse ;

```

begin
t = 0;
Pt başlangıç popülasyonunu oluştur;
Popülasyondaki kromozomların birim basamak cevaplarını tespit etmek için PID
sistemine gönder;
ISE kriterine göre Pt nin uygunluk değerlerini değerlendir;
While {bitiş koşulu}
end
While not {bitiş koşulu} do
    Begin
t = t + 1;                                {(t+1). İterasyon}
Pt-1 den Pt yi seç;                       {yeniden seçmişlemi}
Pt ye genetik operatörleri uygula;        {çaprazlama ve mutasyon işlemleri}
    end
end
end

```

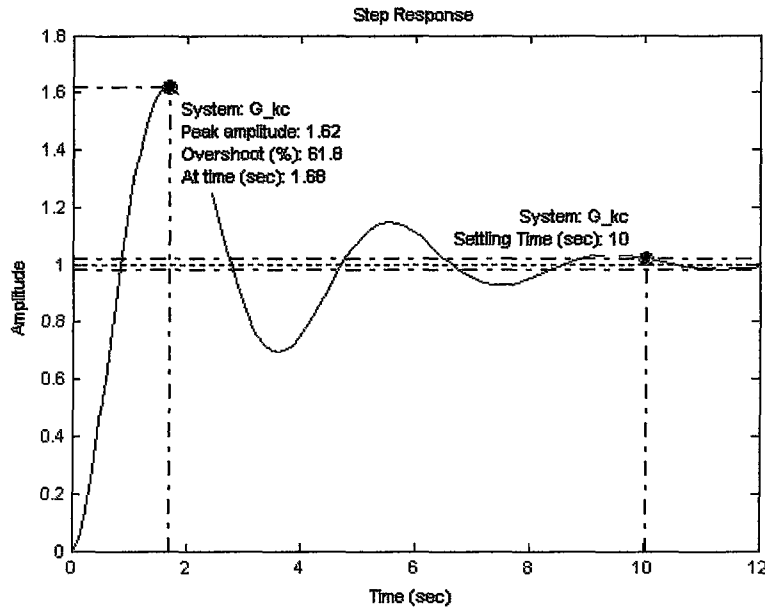
Şekil 3.2. GA Kullanarak PID Optimizasyonu Genel Program Yapısı.

3.2 Optimal Pid Parametrelerinin Hesaplanmasına İlişkin Örnek Uygulamalar

Genetik Algoritmalar kullanarak PID katsayılarının optimize edilmesinde elde edilen sonuçların performanslarının karşılaştırılması için Katsuhiko Ogata' nın Modern Control Engineering ve Norman S. Nise'ın Control Systems Engineering 4th ed. 2004 kitaplarının PID Kontrolörler bölümündeki örneklerden yararlanılmıştır. Kitaplarda yer alan örneklerde optimal PID katsayıları hesaplanırken Ziegler-Nichols , kök yer eğrileri ve yazarların kendi optimal yaklaşımları kullanılmıştır. Bu çözümlere karşın Genetik Algoritmanın ISE kriterlerini baz alarak yaptığı araştırma sonuçları karşılaştırılmıştır.

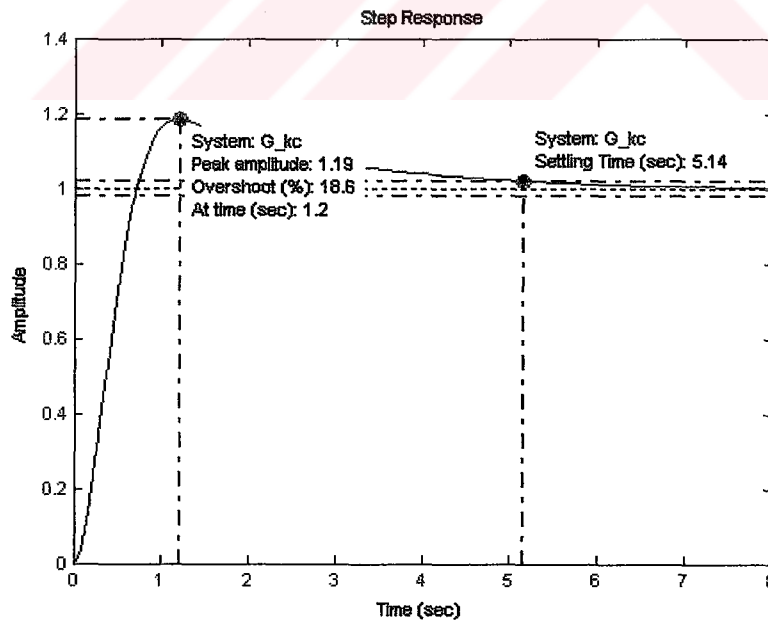
Ogata, transfer fonksiyonu $T(s) = \frac{1}{s(s+1)(s+5)}$ olan sistem için optimal PID

katsayılarını hesaplariken Ziegler-Nichols ikinci metodu yaklaşımıyla; $K_p=18$, $K_i=12.8$, $K_d=6.3$ elde etmiştir. Elde edilen bu sonuçlara göre PID ile kontrol edilen sisteme ait birim basamak cevabının grafiği şekil3.3 teki gibidir.



Şekil 3.3. Ziegler Nichols Yaklaşımı Birim Basamak Cevabı , $K_p=18$, $K_i=12.8$, $K_d=6.3$

PID nin türevsel özelliklerinden yararlanan Ogata, aşımı çok daha uygun değerlere çekmeyi başarmıştır. Sıfır eklenerek bulunan yeni PID parametreleri $K_p=18$, $K_i=5.8498$, $K_d=13.8456$ ye bağlı olarak sistemin çizilen birim basamak cevabı şekil 3.4 te gösterilmiştir.



Şekil 3.4. Sistemin $K_p=18$, $K_i=5.8498$, $K_d=13.8456$ İçin Birim Basamak Cevabı

Ziegler-Nichols ayarlama metodunu başlangıç noktası olarak kullanan Ogata maksimum aşım değerini %18.6' ya indirmiştir. Aynı sistem için Genetik Algoritmalar kullanılarak hesaplanan PID parametreleri ve ilgili sonuçlar aşağıdaki gibidir.

Matlab komut satırından Genetik Algoritma optimizasyonu için yazılan programın temel bölümü olan GenetikPid.m dosyası çalıştırılırsa çalışan Genetik Programın komut satırı çıktısı;

>> GenetikPid

OrnekPeriyot = 0.0040

 Tp = 10

Genetik Algoritma Optimizasyon işlemi sürüyor.....

(İşlemi herhangi bir anda durdurmak için

Ctrl+c basabilirsiniz)

!!İterasyon sayısı maksimum iterasyon sayısı 25 ye ulaştı!!

PID katsayı optimizasyonu tamamlandı

***** GENETİK OPTİMİZASYON RAPORU *****

*****Genetik Algoritma Parametreleri:

Populasyondaki kromozom sayısı=120

Her kromozom 39 bit ile temsil edilmiştir

Caprazlama Olasılığı=0.800000

Mutasyon Olasılığı=0.100000

PID katsayıları değişim aralığı=(0.000000 20.000000)

Maksimum İterasyon Sınırı=25

İterasyonu durduran hata eşiği=0.002000

Cmax=6.000000e-002

*****Elde Edilen PID Katsayıları:

25. iterasyon sonunda elde edildi:

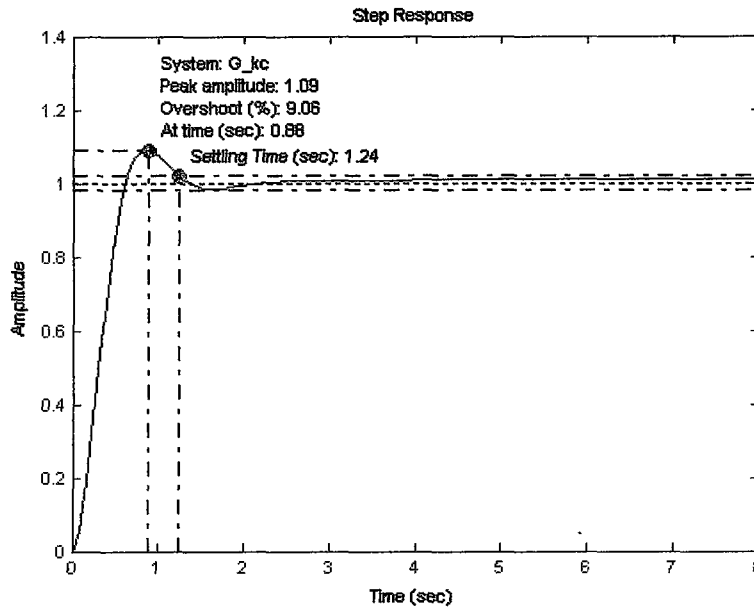
Optimum kp=17.113553

Optimum ki=0.688645

Optimum kd=19.980464

Toplam Karesel Ortalama 0.022765

* Birim basamak giriş için pid cevabı figure No. 1 de çizilmiştir



Şekil 3.5. GA larla Hesaplanan, Sistemin Birim Basamak Cevabı

Pentium 4, 1.8GHz işlemcili bilgisayarla 1 dakika 15 saniye süren genetik süreç, optimizasyon işlemlerinde Genetik Algoritmaların kısa sürede sonuca ulaşabildiğini göstermektedir. Genetik Algoritma kullanılarak yapılan PID optimizasyonu sonucunda maksimum aşım % 9 olarak ölçülmüştür. Oturma süresi ise 1.24 sn olarak ölçülmüştür.

Tablo 3.1. Hesaplanan Farklı PID parametreleri İçin Optimizasyon Sonuçlarının Karşılaştırılması

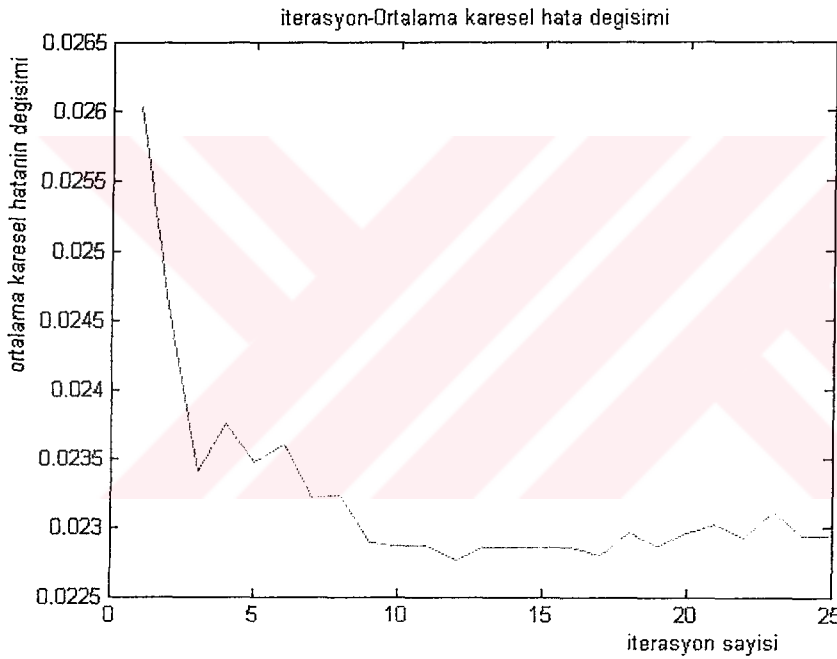
Yöntem\sonuç	K _p	K _i	K _d	Maksimum Aşım (%)	Yerleşme Süresi(s)
Ziegler-Nichols	18	12.8	6.3	61.8	10
Optimize Edilmiş Ziegler-Nichols	18	5.84	13.84	18.6	5.14
GA. Sonuçları	17.1	0.68	19.98	9	1.24

ISE performans indeksinden yararlanarak PID ile kontrol edilen sistemin GA tarafından bulunan PID parametrelerine göre optimizasyonu yapılmıştır. Çıktıdaki toplam karesel hatayı minimum yapan PID katsayıları hesaplanırken, bu işlem maksimum aşım ve yerleşme zamanında daha iyi sonuçlar elde edilmesini sağlamıştır.

Matlab komut satırı çıktısından da görüldüğü üzere 36'şar bitlik kromozomlardan oluşan her yeni popülasyondaki birey sayısı 120'dir. Her yeni genetik süreçte,

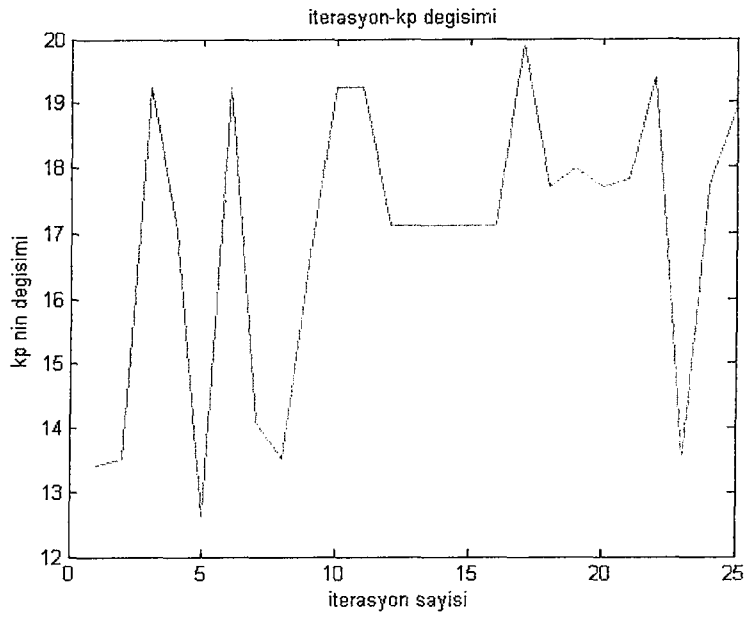
 aprazlama oranı 0.8 ve mutasyon oranı 0.1 olarak se ilmiřtir. Toplam karesel hatanın 0.002 bulunduđu iterasyonda d ng den  ıkılması istenmektedir. Bu hata d zeyine ulařamayan algoritma 25 iterasyon boyunca genetik s rece devam eder. Sonu  olarak en iyi ortalama karesel toplam 0.0022 olarak elde edilmiřtir.

ISE performansını saėlayan bireye ait K_p , K_i ve K_d deėerleri bulunmuřtur. ISE performans indeksini kullanarak genetik s re  boyunca algoritmanın her iterasyonda bulunduđu ortalama toplam karesel hatanın deėiřimi řekil 3.6 da g sterilmiřtir. řekilde PID ile kontrol edilen sistemin birim basamak cevabı i in hesaplanan ISE kriterinin her genetik iterasyon sonucunda daha d ř k  ıkıř hatası verecek řekilde genetik algoritma tarafından kontrol edildiėi g r lmektedir.



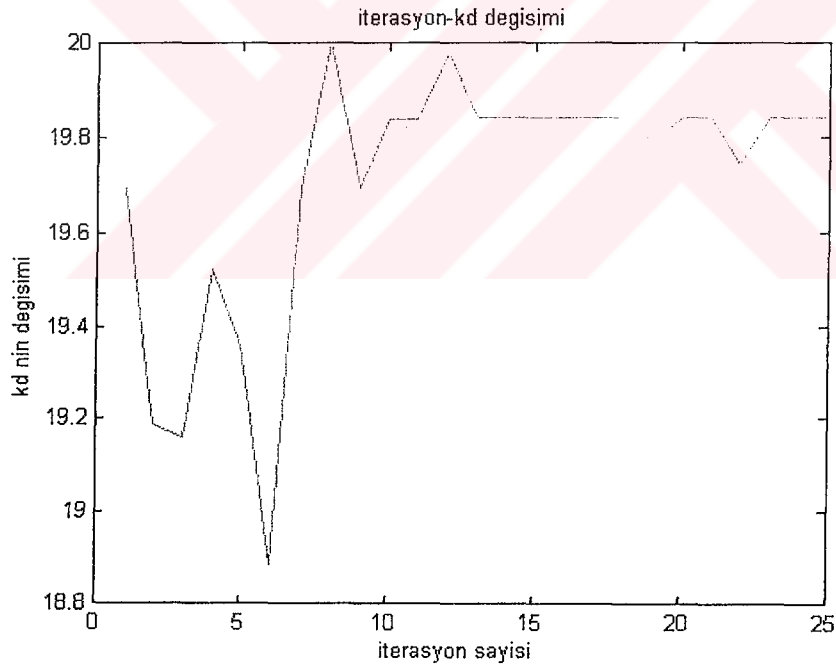
řekil 3.6 Her İterasyon Sonunda Ortalama Karesel Hatanın Deėiřim Grafii

Genetik Algoritma, kullanıcı tarafından beklenilen 0.002 deėerindeki karesel hata deėerinin altına d řmediėi i in 25 iterasyon boyunca genetik s rece devam etmiřtir.

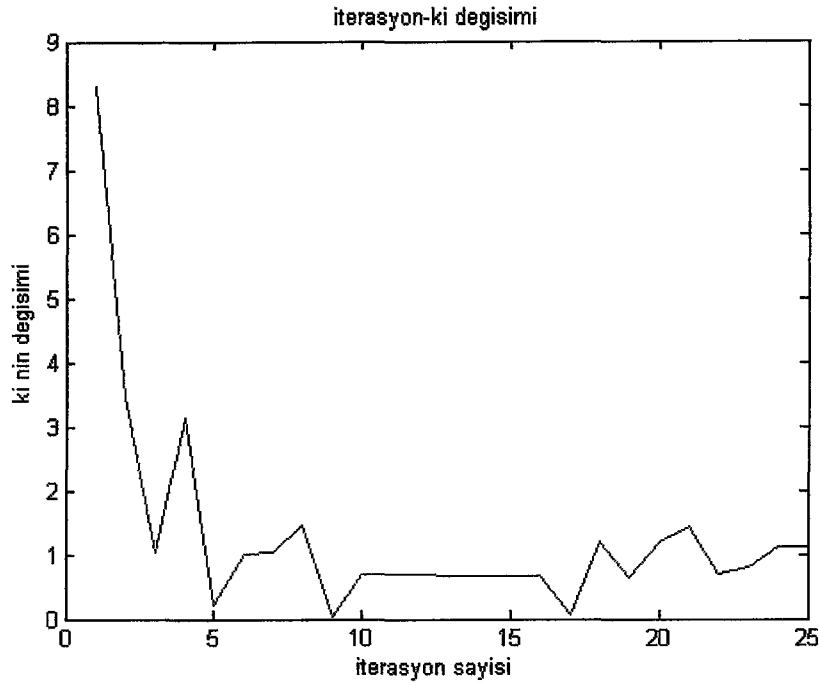


Şekil 3.7. Kp nin Genetik İterasyonlar Boyunca Değişimi

Genetik Algoritma süreci boyunca her iterasyonda K_p , K_i ve K_d farklı değerler alırlar. K_p , K_i ve K_d ' nin bu değişimi şekil 3.6, şekil 3.7 ve şekil 3.8 de görülmektedir.



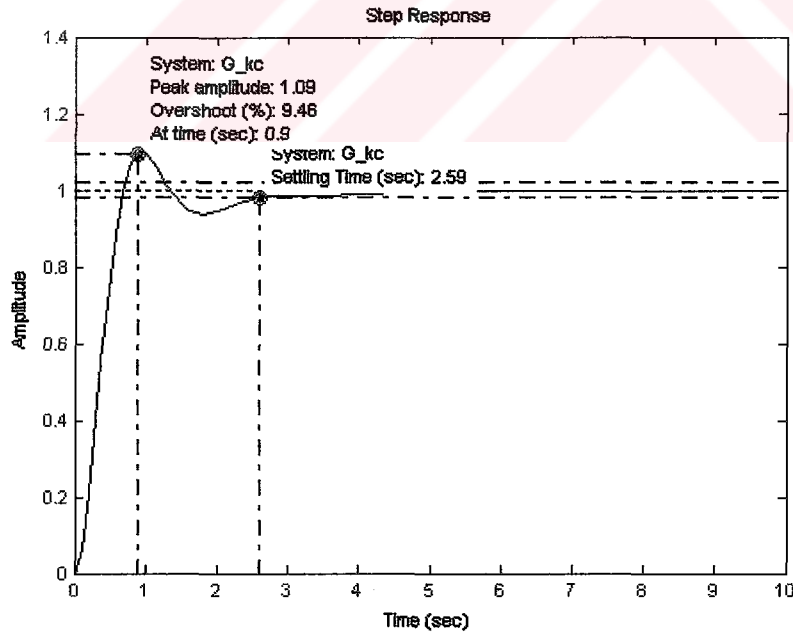
Şekil 3.8. K_d nin Genetik İterasyonlar Boyunca Değişimi



Şekil 3.9. K_i nin Genetik İterasyonlar Boyunca Değişimi

Ogata, transfer fonksiyonu $T(s) = \frac{1.2}{0.36s^3 + 1.86s^2 + 2.5s + 1}$ olan sistem için

optimal PID katsayılarını tasarlamış ve $K_p = 5.88$, $K_i = 2.058$, $K_d = 4.2$ bulmuştur. Bu sisteme ait birim basamak cevabı şekildeki gibidir.



Şekil 3.10. PID Kontrolörlü Sistemin Birim Basamak Cevabı

Ogata tarafından tasarlanan sisteme ait maksimum aşım % 9.46 ve oturma süresi 2.59 saniye olarak ölçülmüştür.

Bu sistemin Genetik Algoritmalar kullanılarak PID katsayılarının tespit edilmesi ve buna bağlı olarak ölçülen maksimum aşım, oturma süresi, toplam karesel hata ve performans karşılaştırmaları aşağıda belirtilmiştir.

Matlab komut satırından çağrılan ana fonksiyon sonuç çıktısı ;

>> GenetikPid

OrnekPeriyot = 0.0040

$T_p = 10$

Genetik Algoritma Optimizasyon işlemi sürüyor.....

(İşlemi herhangi bir anda durdurmak için

Ctrl+c basabilirsiniz)

!!İterasyon sayısı maksimum iterasyon sayısı 25 ye ulaştı!!

PID katsayı optimizasyonu tamamlandı

***** GENETİK OPTİMİZASYON RAPORU *****

*****Genetik Algoritma Parametreleri:

Populasyondaki kromozom sayısı=120

Her kromozom 39 bit ile temsil edilmiştir

Çaprazlama Olasılığı=0.900000

Mutasyon Olasılığı=0.100000

PID katsayıları değişim aralığı=(0.000000 6.000000)

Maksimum İterasyon Sınırı=25

İterasyonu durduran hata eşiği=0.002000

$C_{max}=6.000000e-002$

*****Elde Edilen PID Katsayıları:

25. iterasyon sonunda elde edildi:

Optimum $k_p=5.690842$

Optimum $k_i=3.377289$

Optimum $k_d=5.978022$

Toplam Karesel Ortalama 0.022147

Birim basamak giriş için pid cevabı figure No. 1 de çizilmiştir.

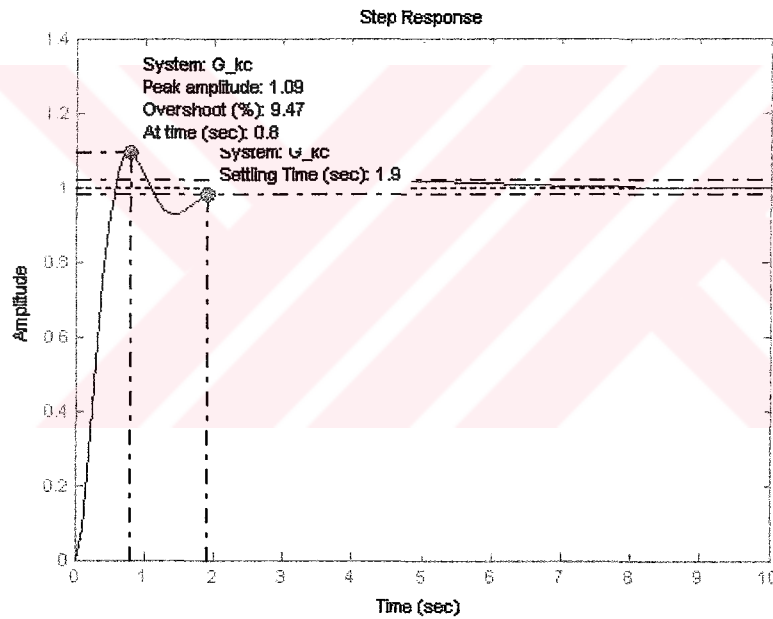
Genetik Algoritma sonuç raporundan da görüldüğü gibi 120 kromozom içeren populasyon, karesel hatanın toplamının ortalamasının minimum olduğu 0.022147

değeri için 25 iterasyon sonucunda $K_p = 5.69$, $K_i = 3.37$ ve $K_d = 5.97$ değerlerini bulmuştur. Maksimum aşımı % 9.47 ve oturma süresi 1.9 saniyedir.

Tablo 3.2. Genetik Algoritmaların Bulduğu Sonuçlarla Ogata Tarafından Hesaplanan Sonuçların Karşılaştırılması

Yöntem\Sonuç	K_p	K_i	K_d	Aşım(%)	$T_{ss}(s)$
Klasik Metotlar	5.88	2.05	4.2	9.46	2.59
GA Sonuçları	5.69	3.37	5.97	9.47	1.9

PID kontrollü sistemin Genetik Algoritmalarla hesaplanan hatanın karesel toplamının ortalamasının optimal olduğu birim basamak cevabı şekil3.11 de görülmektedir.



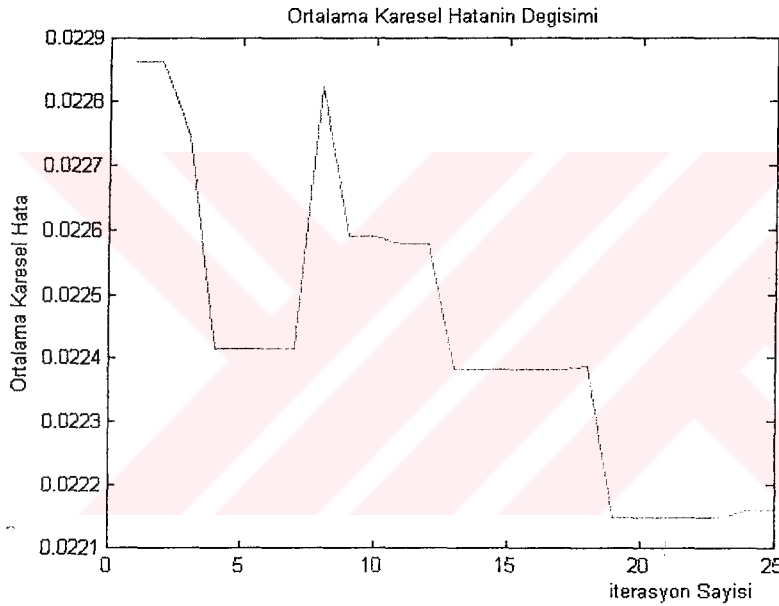
Şekil 3.11. GA larla Hesaplanan, Sistemin Birim Basamak Cevabı

ISE performans kriterine bağlı kalınarak genetik süreç boyunca hatanın karesel toplamının ortalaması minimize edilmiştir. Şekil 3.12 de değişen iterasyonlar boyunca hatanın minimize olduğu görülmektedir. Hatanın ilk iterasyonlarda 2.29×10^{-2} seviyesinden başlayan karesel hatanın toplamının ortalaması genetik süreç sonlarındaki iterasyonlarda $2,21 \times 10^{-2}$ seviyelerine indirilmiştir.

Genetik Algoritmalarla yapılan çalışmalarda algoritmanın ilk birkaç iterasyon sonucunda elde edilen popülasyonda uygunluk değeri yüksek birkaç kromozom

bulunabilir. Bu kromozomların seçilme olasılıkları yüksek olduğundan arama bu dizilerin etrafında olmaktadır. Bu olay algoritmanın zamansız yakınsamasına sebep olmaktadır[Altıparmak ,1996].

Genetik Algoritmalarındaki yakınsama probleminin görmek için kromozomların iterasyonlar boyunca değişimlerini incelemek yeterlidir. Her iterasyondaki ortalama karesel hatanın toplamını minimum yapan elemanların değişimi şekil 3.12 de gösterilmiştir. 25 iterasyon devam eden genetik süreç 19 iterasyonda en optimal sonuca ulaşmıştır. Elde edilen en optimal karesel hatanın ortalama değeri komut satırı çıktısından da görüldüğü gibi 0.022147 değerini göstermektedir. Elde edilen bu sonuç genetik süreç içerisinde bulunan uygunluk değeri en iyi bireye aittir.

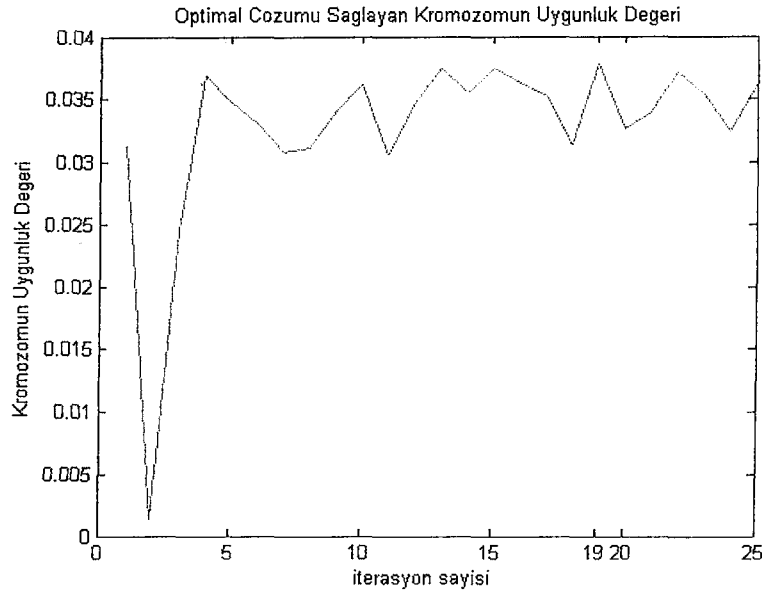


Şekil 3.12. Genetik İterasyon Boyunca Hatanın Minimize Olduğunu Gösterir.

Problemin çözümünde her bir iterasyonda 120 elemanlı bir populasyon oluşturulmuş ve genetik süreç uygun çözümü bulamazsa 25 iterasyon devam etmesi istenmiştir. Optimal çözümü sağlayan K_p , K_i ve K_d değerlerinin elde edildiği kromozom 19. iterasyondaki 93. sütunda bulunan, 120 kromozomlu populasyonun 93. kromozomudur.

Şekil 3.13.' te de görüldüğü gibi optimal çözümü sağlayan kromozomun uygunluk değeri genetik süreç boyunca artış sağlamış ve 19. iterasyonda maksimum uygunluk

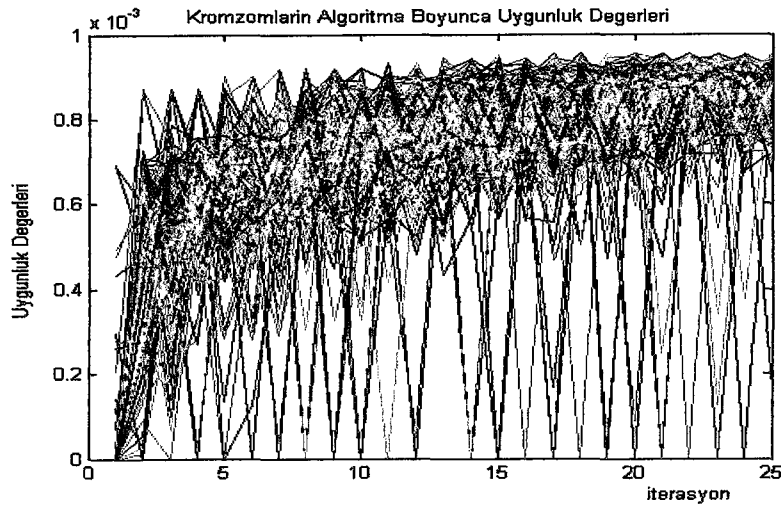
değerine ulaşmıştır. Bu kromozom aynı zamanda genetik süreç boyunca elde edilen en iyi optimal ISE değerini sağlamaktadır.



Şekil 3.13. PID Katsayılarının Elde Edildiği Kromozomun Uygunluk Değeri Değişimi

120 kromozomdan oluşan ve 25 iterasyon devam eden populasyon genetik süreç boyunca, PID sistemine her bir kromozom için uygunluk değeri gönderdiğinden, toplamda bu iki değerin çarpımı kadar, 3000 farklı uygunluk değeri almıştır. Algoritmanın araştırma uzayını gösteren grafik şekil 3.14’ deki gibidir.

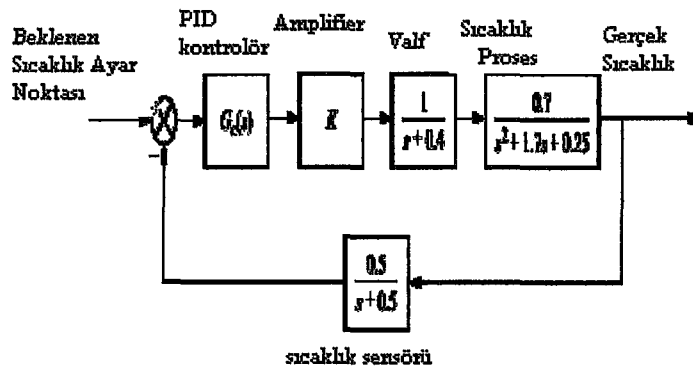
Genetik Algoritmaların çalışması için gerekli ilk kromozom populasyonu rastsal olarak oluşturulur ve uygunluk değerleri genelde sıfıra yakındır. Genetik süreç boyunca genetik operatörlere tabi tutulan kromozomların uygunluk değerleri artar. Şekil 3.14’ ten iterasyon sayısı arttıkça populasyonların uygunluk değerlerinin de arttığı görülmektedir.



Şekil 3.14. Genetik İterasyon Boyunca Kromozomların Uygunluk Değerleri Grafiği

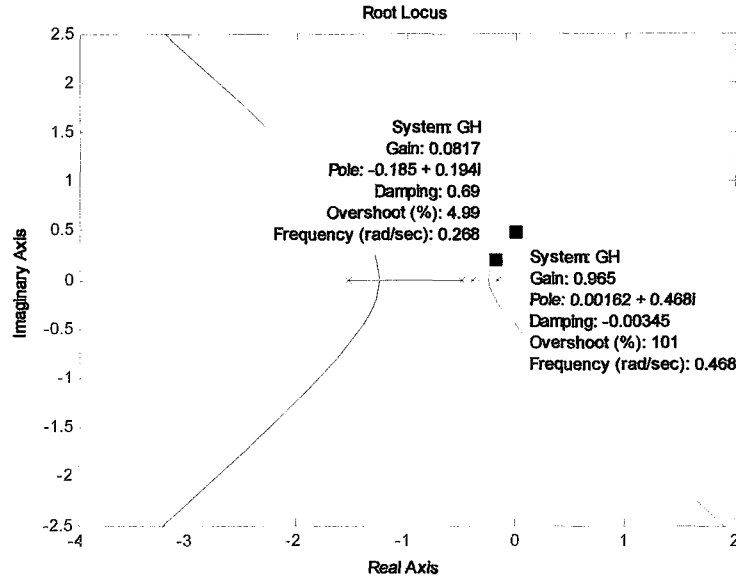
Performans açısından karşılaştırma yapacağımız üçüncü örnek ise Norman S.Nise'in Control Systems Engineering kitabından;

Bir kimyasal proses sıcaklık kontrol sistemine ait blok şeması verilmiştir. Sistem PID kontrol öncesi (yani $G_c(s)$ yokken ya da 1 iken) amplifikatörle $0.7 \cdot K = 0.082$ seçilerek kontrol edildiğinde (yani P kontrol yapıldığında) birim basamak cevabı %10.8 maksimum üst aşım (%ÜA), T_s (yerleşme zm.)= 20.9 sn. yapmaktadır (aşağıda çizdirilmiştir). Ayrıca sistem büyük bir sürekli hal hatasına (e_{ss}) sahiptir (%55 civarı).

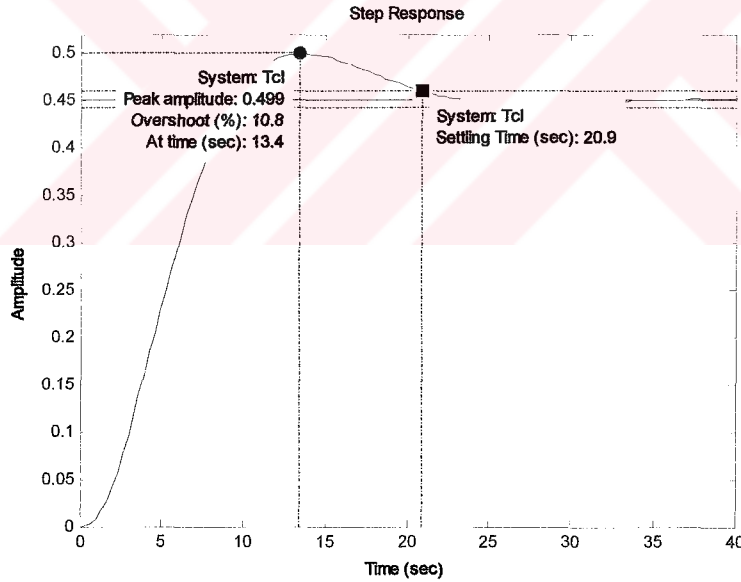


Şekil 3.15. Proses Sıcaklık Kontrol Sistemi

Sistemin kök yer eğrisi;



Şekil.3.16. Proses sıcaklık kontrol sistemine ait kök yer eğrisi

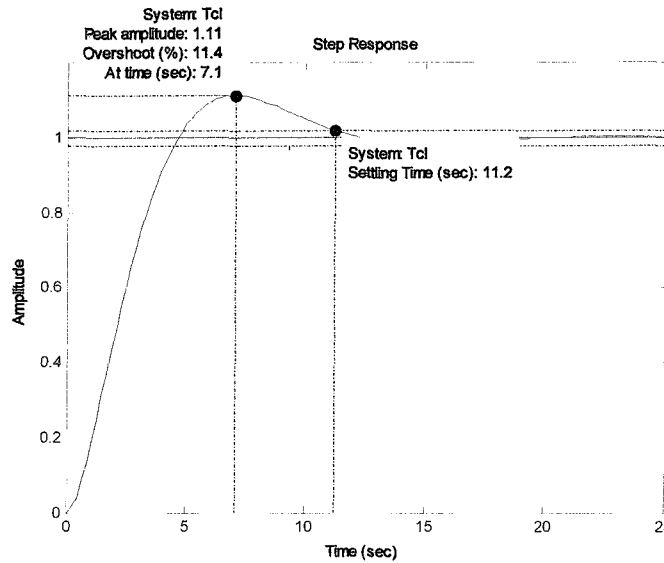


Şekil.3.17. Proses sıcaklık kontrol sistemine ait kök yer eğrisi birim basamak cevabı

Şekil.3.16'da kök yer eğrisinden bulunan değerlerle sistemin birim basamak cevabından da görüldüğü üzere maksimum aşım %10.8 ve oturma süresi 20.9 dur.

Norman, Control Systems Engineering kitabında birim darbe cevabının daha ideal hale getirilmesini önermiştir;

Sisteme öyle bir PID kontrolör tasarlayınız ki sistem birim basamak cevabı %ÜA değeri pek değişmeden, yerleşme cevabı yaklaşık 2 kat kısalsın. Ayrıca sürekli hal hatası sıfırlansın. Bu değerleri sağlayan birim basamak cevabı idealde aşağıdaki gibidir.



Şekil 3.18 Proses sıcaklık kontrol sistemine ait Norman tarafından önerilen birim basamak cevabı

Proses sıcaklık kontrol sistemine ait optimal PID katsayılarını hesaplamak için Genetik Algoritmalarından yararlanmak sistemin kapalı çevrim fonksiyonunu Matlab kodlarında uygun alana yazdıktan sonra Matlab komut satırından GenetikPid.m dosyasını çalıştırsak;

>> GenetikPid

OrnekPeriyot = 0.04

$T_p = 50$

Genetik Algoritma Optimizasyon işlemi sürüyor.....

(İşlemi herhangi bir anda durdurmak için

Ctrl+c basabilirsiniz)

!!İterasyon sayısı maksimum iterasyon sayısı 25 ye ulaştı!!

PID katsayı optimizasyonu tamamlandı

***** GENETİK OPTİMİZASYON RAPORU *****

*****Genetik Algoritma Parametreleri:

Populasyondaki kromozom sayısı=120

Her kromozom 39 bit ile temsil edilmiştir

Caprazlama Olasılığı=0.900000

Mutasyon Olasılığı=0.100000

PID katsayıları değişim aralığı=(0.000000 1.000000)

Maksimum Iterasyon Sınırı=25

Iterasyonu durduran hata eşiği=0.002000

Cmax=4.000000e-002

*****Elde Edilen PID Katsayıları:

25. iterasyon sonunda elde edildi:

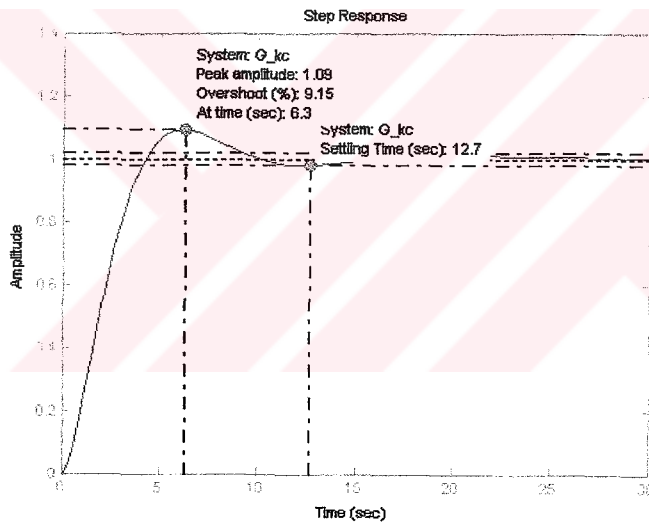
Optimum $k_p=0.364103$

Optimum $k_i=0.039072$

Optimum $k_d=0.991697$

Toplam Karesel Ortalama 0.028806

Birim basamak giriş için pid cevabı figure No. 1 de çizilmiştir.



Şekil 3.19. Proses Kontrol Sistemi GA Birim Basamak Cevabı

Kök yer eğrileri, Norman tarafından önerilen aşım-yerleşme süresi ve genetik algoritmalar tarafından bulunan sonuçların karşılaştırılması tablo.3.3’de verilmiştir.

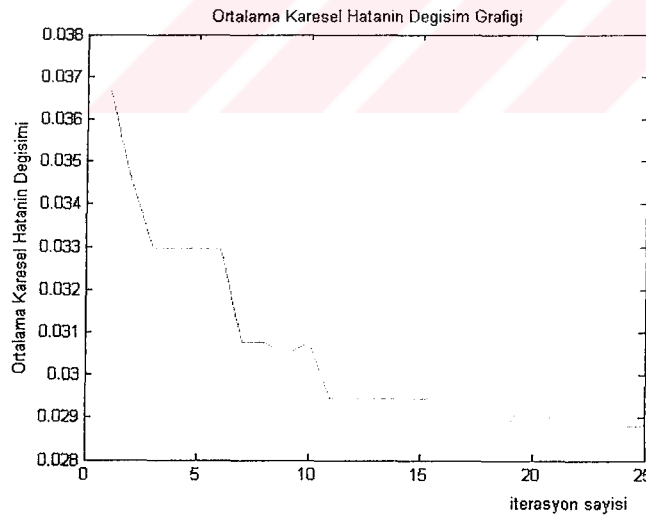
Tablo 3.3. Proses Kontrol Sistemine ait Sonuçların Karşılaştırılması

Yöntem/Sonuç	Aşım(%)	Yerleşme Süresi(sn)
Kök Yer Eğrisi	10.8	20.9
Optimize Kök Yer Eğrisi	11.4	11.2
GA Sonuçları	9.15	12.7

Kök yer eğrileriyle hesaplanan yerleşme süresi 20.9 sn. iken genetik algoritmalarla hesaplanan yerleşme süresi 12.7 sn. değerine inmiştir. Ayrıca genetik algoritmalarla elde edilen sonuçlarda maksimum aşım ait değer de klasik yöntemlerle karşılaştırıldığında daha uygun sonuçlara ulaşılmıştır.

Proses kontrol sistemine ait PID katsayılarını hesaplarken Genetik Algoritmalar için her bir populasyon 120 kromozomla temsil edilmiştir. Çaprazlama oranı 0.9 seçilerek her bir nesilden seçilen $0.9 \times 120 = 108$ kromozom çaprazlamaya girecektir. Mutasyon oranı 0.1 seçilerek her nesilde seçilen $0.1 \times 120 = 12$ kromozom mutasyona uğrayacaktır. İterasyonun 0.02 ortalama karesel hataya erişmesi durumunda genetik sürecin sonlanması, bu ortalama erişmemesi durumunda 25 iterasyon boyunca genetik iterasyona devam edilmesi istenmiştir. Genetik iterasyonlar boyunca 0.02 ortalama ISE değerinin altına düşülmediği için genetik süreç 25 defa tekrarlamıştır. Genetik Algoritma tarafından elde edilen ortalama ISE değeri 0.028806 bulunmuştur.

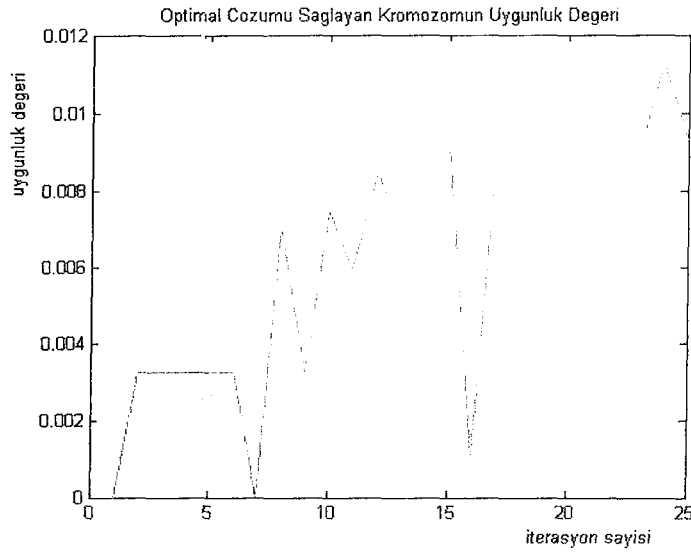
Proses sıcaklık kontrol sistemi için karesel hatanın ortalamasının değişim grafiği şekil.3.19.'da görülmektedir. İlk iterasyonlarda 0.037 değerinde ortalama karesel hataya sahip olan optimal çözümü sağlayan kromozom genetik iterasyonlar boyunca azalarak 0.0288 değerine inmiştir.



Şekil 3.20. Sıcaklık Kontrol Sistemine ait Ortalama Karesel Hatanın Değişim Grafiği

PID katsayıları genetik algoritmalarla hesaplanırken en uygun çözümün 24.iterasyonda 28. kromozom (120 kromozomlu populasyonun 28. kromozomu) için

elde edilmiştir. Optimal çözümü sağlayan kromozoma ait uygunluk değeri değişimi grafiği şekil 3.20 de görülmektedir.



Şekil 3.21. Optimal Çözümü Sağlayan Kromozoma Ait Uygunluk Değeri Değişimi

İlk iterasyonlarda sıfır uygunluk değerinden başlayan kromozoma ait uygunluk değeri 24. iterasyonda en iyi uygunluk değerin olan 0.015'e ulaşmıştır. Elde edilen en iyi uygunluk değeri için genetik algoritma minimizasyon probleminde minimum optimal sonuca ulaşarak en iyi ISE değerini elde etmiştir.

4. SONUÇLAR VE ÖNERİLER

PID kontrolörler endüstride yaygın olarak kullanılmaktadır. Kontrol edilmek istenen sistem için kazanç, integral ve türev zamanı parametreleri bulunarak, kontrolün iyi şekilde sağlanır. PID kontrolörler sistemde sıfır kalıcı durum hatası olan hızlı bir cevap sağlar. Ancak PID kontrolörün K_p , K_i ve K_d parametreleri uygun olarak ayarlanmadığında kontrolörün bu üstün özelliklerinden yararlanılamaz.

Bu çalışmada PID kontrolörlerinin K_p , K_i ve K_d katsayılarının optimizasyonu için Genetik Algoritmalarından yararlanılmıştır. Optimizasyon kriteri olarak ISE performans kriteri kullanılmıştır. Genetik Algoritma, karesi alınmış hatanın integralinin minimum olmasına genetik süreç boyunca yönlendirilmiştir. ISE performans kriterinin, minimum olduğu toplam integral hata değeri için maksimum aşım ve yerleşme süresini de minimum yapması beklenir.

Genetik Algoritmanın çalışma başarısının karşılaştırılması için Katsuhiko Ogata'nın Modern Control Engineering kitabının PID Kontrolörler bölümündeki uygulamalar kullanılmıştır. Ogata, kök yer eğrileri ve Ziegler-Nichols ikinci metoduyla elde edilen PID katsayılarını daha iyi çözüm noktalarına yönelmede ilk adım olarak kabul etmiştir. Üçüncü bölümdeki performans karşılaştırma tablolarından da görüldüğü üzere Genetik Algoritmalar ISE performans kriterini minimize ederken maksimum aşım ve yerleşme süresinde klasik metotlara üstünlükler sağlamıştır. Ayrıca hem klasik metotlarla hem de Ogata'nın optimizasyon yaklaşımıyla yapılan çözümlerin zaman alacağı görülmektedir. Genetik Algoritmaların optimizasyon problemlerini çözmede sağladığı zaman çabukluğuna da uygulamalar bölümünde değinilmiştir.

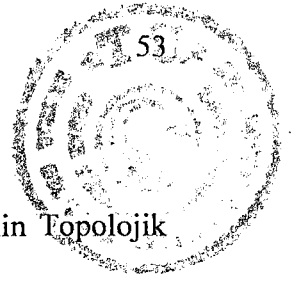
Elde edilen sonuçlardan ISE performans indeksinin hatayı minimize etmeye çalışırken maksimum aşımı ve oturma süresini de minimize ettiği gözlenmiştir. Buda beklenen bir sonuçtur. Ayrıca GA ların parametrik belirsizliği olan sistemlerde iyi sonuçlar vermekte olduğu görülmüştür.

Genetik Algoritmalarla işlem yaparken elde edilen sonuçların genetik süreç içerisindeki değişimi incelenmelidir. Genetik Algoritmalarından sağlıklı veriler elde edebilmek için genetik süreç boyunca popülasyonun uygunluk değerlerinde yakınsama olmamalıdır. Genetik Algoritmanın yakınsama problemiyle karşılaşp karşılaşmadığı

genetik süreçten elde edilen verilerden tespit edilebilir. Genetik Algoritmanın kontrol parametreleri etkili kullanılarak daha optimal sonuçlara ulaşılabilir.



KAYNAKLAR



1. Altıparmak, F., 1996, "Genetik Algoritmalar ile Haberleşme Şebekelerinin Topolojik Optimizasyonu", Gazi Üniversitesi Fen Bilimleri Enstitüsü.
2. Astrom, K.J. ve Hagglund T., Automatic Tuning of Simple Regulators with Specification on Phase and Amplitude Margins, Automatica, Vol: 20, No: 5, 1984.
3. Astrom, K.J., Automatic Tuning of PID Regulators, Research Triangle Park, N.C., Instrument Society of America, 1998.
4. Biegel, J.E ve Davern, J.J., (1990), "Genetic Algorithms and Job Shop Scheduling", Computers and Industrial Engineering, 19(1-4): 81-91.
5. Cantoni, M., Zio, E., ve Marseguera, M., (1999), "Genetik Algoritmalar ve Optimal İşletme Tasarımı İçin Monte Carlo Modeli", Reliability Engineering and System Safety, 68: 29-38.
6. Caruna, R.A. and Schaffer, J.D. Representation and hidden bias: Gray binary coding for genetic algorithms, Proc. Of the 5th Int. Conf. On Machine Learning, The American Association for Artificial Intelligence, pp. 153-161, (1988).
7. Goldberg, D.E., 1989, Genetic Algorithms in Search, Optimization and Machine Learning, Addison Wesley.
8. Davids, L. (ed), Handbook of Genetic Algorithms, Van Nostrand Reinhold, NY, New York, (1991).
9. De Jong, K.A. An analysis of the behaviour genetic adaptive systems, PhD Dissertation, University of Michigan, (1998).
10. Dengiz, B. ve Altıparmak F., (1998), "Genetik Algoritmalar Genel Bakış", Gazi Üniversitesi Mühendislik Mimarlık Fakültesi, Endüstri Mühendisliği, 9(3).
11. Dorf, Richard C. Robert H. Bishop Modern Kontrol System, 2001
12. Dumond, E.A et al., Automatic Tuning of Industrial PID Controller, American Control Conference, Boston, USA, 1985.
13. Goldberg, D.E., Deb, K., 1991, "A Comparative Analysis of Selection Schemes Used in Genetic Algorithms", Foundations of Genetic Algorithms, (Ed.: G.J.E. Rawlins), Morgan Kaufmann Publishers, Las Altos, C.A, 69-93



14. Hurma, Nevin, PID kontrolör ve PLC Uygulaması, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, İstanbul, 1998.
15. Kuo, Benjamin C., Otomatik Kontrol Sistemleri, (Çev: Atilla Biri), Literatür Yayınları: 35, Yedinci Baskı, İstanbul, 1999.
16. Liepins, G.E., Hillard, M.R., Richardson, J., Palmer, M., 1990, "Genetic Algorithm Applications to Set Covering and Travelling Salesman Problems", Operation Research and Artificial Intelligence, (Editors: Brown, D., White, C.), Kluwer Academic Publishers, 29-57.
17. Michalewicz, Z., 1992, Genetic Algorithm + Data Structures = Evolution Programs, Springer-Verlag.
18. Michalewicz, Z., Janikow, C.Z., 1992, "Genocop: A Genetic Algorithm for Numerical Optimization Problems with Linear Constraints", Communications of the ACM.
19. Ogata, Katsuhiko, Modern Control Engineering, Prentice Hall Inc., 1970.
20. Sarioğlu, Kemal, Otomatik Kontrol 1, 2 Birsen Yayınevi, İstanbul, 1999.
21. Valenzuela, C.L., (1995), Evolution Divide and Conquer; a novel genetic approach to the TSP, Imperial College.
22. W. Zuo, IEE Proc.-Control Theory Appl. Vol. 142, No. 2, March 1995.
23. Yeniay, M.Ö., (1999), Taguchi Deney Tasarımı Problemlerine Genetik Algoritma Yaklaşımı", Doktora Tezi, Hacettepe Üniversitesi Fen Bilimleri Enstitüsü.
24. Yüksel, İbrahim, Otomatik Kontrol Sistem Dinamiği ve Denetim Sistemleri, Uludağ Üniversitesi Güçlendirme Vakfı, Bursa, 1997.
25. Zhuang, Minxia, Computer Aided PID Controller Design, Doktora Tezi, Brighton, 1992.

ÖZGEÇMİŞ

1976 Elazığ doğumludur. Üniversite öncesi eğitimini Elazığ Cumhuriyet İlkokulu ve Elazığ Anadolu Lisesi' nde tamamladı. Mühendislik eğitimini Uludağ Üniversitesi Elektronik Mühendisliği Bölümünden 2000 yılında mezun olarak tamamladı. Halen Türk Telekom A.Ş. de network yönetimi ve güvenliği konularında çalışmaktadır.

