

# Gerçek Kodlu Genetik Algoritma Kullanılarak Sistem Kimliklendirme

Aytekin BAĞIŞ<sup>1</sup>

Yıldız ÖZÇELİK<sup>2</sup>

<sup>1,2</sup>Erciyes Üniversitesi, Elektrik-Elektronik Mühendisliği, 38039, Kayseri, Türkiye

<sup>1</sup>e-posta: bagis@erciyes.edu.tr

<sup>2</sup>e-posta: yildiz@arkenerji.com.tr

## Özetçe

Dinamik bir sistem modelinin gerçek sistemden alınan giriş-çıkış ölçümleriyle belirlenmesi işlemine sistem kimliklendirme adı verilir. Bu çalışma, kontrol edilmek istenen sisteme ait olarak verilen çeşitli giriş-çıkış değerlerini sağlayabilecek uygun transfer fonksiyon parametrelerinin belirlenmesi esasına dayandırılmıştır. Sistem kimliklendirme yöntemi için yapay zekâ optimizasyon algoritmalarından gerçek kodlu genetik algoritma (GGA) (Real-Coded Genetic Algorithm) kullanılmıştır. Ele alınan algoritmanın sistem kimliklendirme konusundaki performans analizleri verilerek, benzetim sonuçları ikili kodlu genetik algoritma (GA) (Binary Coded Genetic Algorithm) ve tabu araştırma algoritması (TAA) (Tabu Search Algorithm) ile kıyaslanmıştır. Sistem kimliklendirme konusunda GGA'nın başarılı bir performans sergilediği gözlemlenmiştir.

## 1. Giriş

Kontrol sistemlerinin tasarımındaki temel güçlüklerden en önemlisi, kontrol edilen sistemin doğru şekilde kimliklendirilmesinde ortaya çıkar. Sistem kimliklendirmede esas olan, önceki bilgi ve gözlemlerden yararlanılarak sistemin bir çeşit matematiksel modelinin ortaya konulmasıdır. Sistemin etkin bir şekilde kontrol edilebilmesi, çıkış değeri hakkında her zaman bilgi sahibi olunabilmesi ve geliştirilebilmesi açısından sistem modelinin ve tahmin edilen parametre değerlerinin mümkün olan en az hata ile belirlenmesi gerekir. Eğer sistemin giriş ve çıkış değerleri arasında doğrusal olmayan bir ilişki mevcutsa ve istenmeyen bozucu girişler sistemi etkiliyorsa, parametre değerlerinin minimum hata ile bulunması zorlaşır.

Sistemlerin kimliklendirilmesine yönelik olarak literatürde çeşitli çalışmalar mevcuttur [1–10]. Sistem kimliklendirme için parametrik ve parametrik olmayan çeşitli yaklaşımlar Ljung tarafından incelendi ve uygulandı [2]. Solbrand ve arkadaşları çevrimdışı kimliklendirme için rekürsif yöntemler önermişlerdir [1]. Frekans domeninde transfer fonksiyonunun parametrik tahminine yönelik bir çalışma Pintelon ve arkadaşları tarafından ele alınmıştır [5]. Açık ve kapalı çevrim basamak cevaplarından yararlanarak model parametrelerinin tahminini sağlayan yöntemler Ahmed ve arkadaşları tarafından geliştirilmiştir [10].

Sistem kimliklendirme problemlerine hızlı ve etkili bir çözüm sağlayabilmek için yararlanılan yöntemlerden biri de, genetik algoritmalar, tabu araştırma algoritması, farksal gelişim algoritması gibi evrimsel gelişime dayalı yapay zekâ optimizasyon algoritmalarının kullanımıdır [9, 11]. Amaç fonksiyonu optimizasyonu, sinir ağlarının eğitimi, bulanık sistemlerde üyelik fonksiyonlarının belirlenmesi, sistem

kimliklendirme ya da sistem kontrolü gibi alanlarda bu algoritmalar başarılı şekilde kullanılabilir [11].

Bu çalışmanın amacı, sistem kimliklendirme işleminde gerçek kodlu genetik algoritma (GGA) performansının incelenmesi ve elde edilen sonuçların karşılaştırmalı olarak ortaya konulmasıdır. Bölüm 2'de GGA hakkında genel bilgiler verilmiştir. Bölüm 3'de sistem kimliklendirme problemi ele alınmaktadır. GGA, ikili kodlu GA ve TAA'nın uygulanmasıyla elde edilen performans sonuçları karşılaştırılmalı olarak Bölüm 4'de sunulmuştur. Sonuçlar Bölüm 5'de verilmiştir.

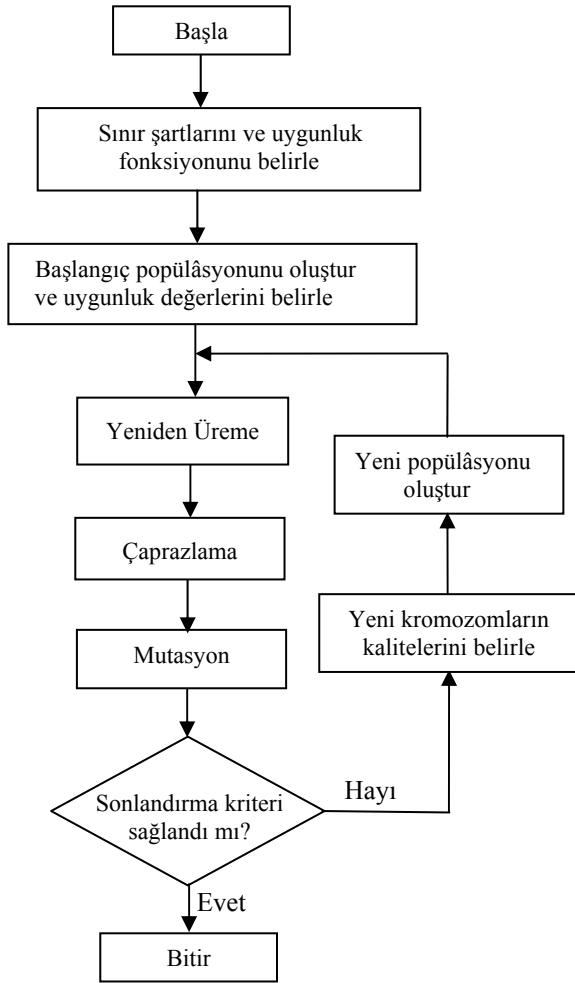
## 2. Gerçek Kodlu Genetik Algoritma (GGA)

GGA'lar, genetik operatörler yardımıyla evrim mekanizmasını uygulayan araştırma algoritmalarıdır. Olası çözümleri ifade eden kromozomlar çözüm uzayı sınırları içerisinde yer alan onluk düzendeki gerçek sayısal değerlerle ifade edilir. Bu özellik GGA'nın ikili kodlu GA'dan temel farkını da ortaya koymaktadır. Birçok gerçek optimizasyon problemi için genlerin gerçek değerleriyle kullanılması problemlerin çözümünde daha yararlıdır. Çözümlerin ikili düzende tanımlanması, kullanılan bit sayısı ile ilişkili olarak çözüm kesinliğini etkiler. Eğer kromozomların değerleri çok büyük ve ondalıklı yapıya sahipse, bu kromozomları ikili kod ile temsil edilmesi çok karmaşık ve zor olacaktır. Ayrıca, çözümlerin gerçek değerlerle tanımlanmış olması hesaplama etkinliğini de oldukça artırır. Bu nedenle birçok mühendislik probleminin çözümünde GGA'ların kullanılması kullanıcıya büyük kolaylık ve yararlar sağlayabilmektedir [12–16].

GGA'nın genel akış şeması Şekil 1'de verilmiştir. Şekilden de görüldüğü gibi, GGA'nın genel işleyişi ikili kodlanmış GA'lar ile aynıdır. İkili kodlu GA'da kullanılan çaprazlama, mutasyon ve seçim operatörleri GGA için de kullanılır. Algoritmadaki temel farklılık, çözümlerin tanımlanmasında ve genetik operatörlerin uygulanmasında ortaya çıkar. Özellikle yüksek seviyeli programlama dilleri ile ve değişken sayısının büyük olduğu problemler için gerçek kodlama tekniğinin kullanımı problemin çözümüne yönelik olumlu katkılar getirebilmektedir.

Gerçek kodlu algoritma, olası çözümlerden (kromozomlardan) oluşan rastgele başlangıç popülasyonunun oluşturulmasıyla başlar. Kromozomlar belli sınırlar içerisinde belirlenirler ve daha önceden belirlenen bir amaç fonksiyonundan elde edilen uygunluk değerlerine göre değerlendirilirler. Seçim yöntemi ile en iyi uygunluk değerine sahip kromozomların (ebeveynlerin) seçilmesi sağlanır. Seçim operatörü ile belirlenen iki ebeveyn çaprazlama operatörü uygulanır. İki ebeveynin genetik bilgisinin değiştirilmesiyle çaprazlama operatörü uygulanmış olur ve yeni bir çocuk birey elde edilir. Fakat GGA'da çaprazlama

operatörünün uygulanması bu kadar basit değildir. Gerçek değerlerle tanımlanmış kromozomlar arasındaki çaprazlama işlemi çeşitli şekillerde uygulanabilmektedir. Tek noktali ve çok noktali çaprazlama işlemlerinin yanında kromozom yapısında oluşturulan değişikliğin belirli bazı kural ve eşitliklere bağlanması da mümkündür. Algoritmanın bir diğer aşaması olan mutasyon işlemi de yine gerçek kodlu yapıya uygun formda gerçekleştirilir. Elde edilen yeni olası çözümlerin kaliteleri istenen değerlere ulaşana ya da istenilen sonlandırma kriteri sağlanana dek bu değerlendirme ve yeni popülasyon oluşturma işlemi devam eder.

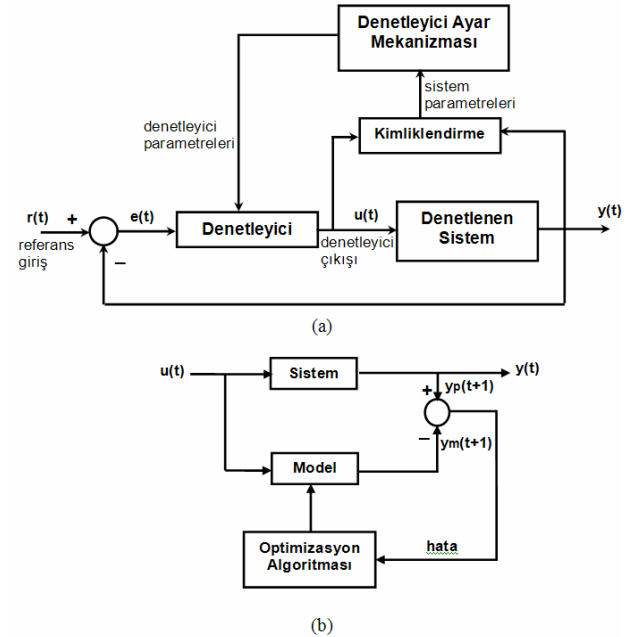


Şekil 1: GGA Genel Akış Şeması

### 3. Sistem Kimliklendirme Problemine GGA Uygulaması

Sistem kimliklendirme işleminde temelde iki yaklaşım söz konusudur: Yapının kimliklendirilmesi ve parametre kimliklendirme. Parametre kimliklendirmede yapının bilindiği varsayılır. Bu çalışmada, farklı dinamik cevaplara sahip olan farklı derecelerdeki süreçlerin birinci ve ikinci dereceli modellerle nasıl tanımlanabileceği üzerinde durulmuş, bu amaca yönelik olarak iki farklı olası model yapısı önerilerek bu modellerdeki parametreler kimliklendirilmeye çalışılmıştır.

Zaman domenine bağlı bir denetleyici tasarımı, sistem kimliklendirme ve denetleyici ayar mekanizması olmak üzere iki temel yapıya sahiptir. İyi bir denetim sisteminin elde edilmesinde kimliklendirme işleminin önemi büyüktür. Şekil 2 denetleyici ayarının ve sistem modelleme işlemlerinin temel blok gösterimini sunmaktadır.



Şekil 2: a) Denetleyici Parametrelerinin Ayarlanması  
b) Sistem Modelleme İşlemi

Modele ait parametreler optimizasyon algoritmaları tarafından belirlenir. Bu çalışmada ele alınan sistem kimliklendirmedeki amaç, kontrol edilmek istenen sisteme ait olarak verilen çeşitli giriş-çıkış değerlerini sağlayabilecek uygun transfer fonksiyonunun belirlenmesidir. Optimizasyon algoritmasının buradaki temel görevi, transfer fonksiyonunu oluşturan katsayı değerlerini belirlemektir. Yüksek dereceli süreçleri düşük dereceli modellere uygunlaştırmaya çalışmak çok yaygın ve bilinen bir kimliklendirme yöntemidir [9, 17]. Çalışmadaki kimliklendirme problemi Eşitlik 1-4'de gösterilen farklı derecelerdeki süreçlerin, Eşitlik 5 ve 6'da verilen modellerle tanımlanmasının hangi parametre değerleri ile mümkün olabileceğini araştırır.

$$G1(s) = \frac{e^{-0.5s}}{(s+1)^2} \quad (1)$$

$$G2(s) = \frac{4.228}{(s+0.5)(s^2+1.64s+8.456)} \quad (2)$$

$$G3(s) = \frac{27}{(s+1)(s+3)^3} \quad (3)$$

$$G4(s) = \frac{2.1299}{0.356s^4 + 0.8116s^3 + 8.5057s^2 + 3.4020s + 2.1299} \quad (4)$$

$$Gm1(s) = \frac{e^{-Ls}}{as+b} \quad (5)$$

$$Gm2(s) = \frac{e^{-Ls}}{as^2+bs+c} \quad (6)$$

Eşitlik 5 ve 6'da verilen model yapıları sırasıyla 1. ve 2. derece modellerdir. Bu modellerde yer alan  $L$ ,  $a$ ,  $b$  ve  $c$  parametreleri algoritma tarafından kimliklendirilecek değerlerdir. İfadelerin pay kısımlarında bulunan  $e^{-Ls}$  terimi zaman domeninde  $L$  sn'lik bir gecikmeye karşılık gelmektedir. Eğer  $L=0$  ise bu terim 1 değerine sahip olur ve sistemin dinamik cevabında herhangi bir gecikme meydana gelmez.

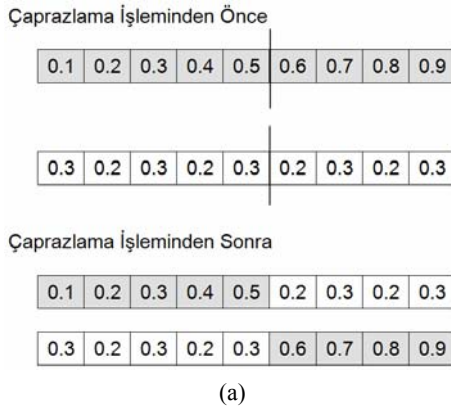
Parametrelerin tahmin edilebilmesi için süreçlerin açık çevrim basamak cevaplarından yararlanır. Optimizasyon süresince gerçek model ile kimliklendirilen model arasındaki fark hesaplanır. Algoritmanın performansı ve elde edilen çözümlerin kalitesi bu farka göre değerlendirilir. Bu çalışmada Eşitlik 7'de verilen performans fonksiyonu kullanılmıştır.

$$J(L, a, b, c) = \int_0^{\infty} (y_p(t+1) - y_m(t+1))^2 \cdot dt \quad (7)$$

Burada  $y_p(t+1)$  sürecin basamak cevabı ve  $y_m(t+1)$  Gm1 ya da Gm2 tarafından kimliklendirilen modelin basamak cevabıdır. Bu performans fonksiyonuna integral kare hatası (Integral Square Error-ISE) adı da verilir. Kaliteli model parametrelerinin elde edilebilmesi için bu ISE hata değerinin minimum yapılması gerekir.

Bu çalışmada ele alınan sistem kimliklendirme probleminin çözümünde gerçek kodlu genetik algoritma (GGA) kullanılmıştır. Bu amaçla, çözüm kümesini oluşturan parametre değerleri onluk sistemdeki gerçek nümerik değerlerle tanımlanmıştır. Bu değerlerin çaprazlanması ve mutasyona uğratılması için de bu amaca uygun kodlama ve tanımlamalar yapılmıştır. Aradaki farkın gözlenmesi amacıyla aynı çalışma ikili kodlanmış genetik algoritma (GA) kullanılarak tekrarlanmıştır. İkili kodlu ve gerçek kodlu algoritmalarda uygulanan çaprazlama ve mutasyon işlemleri için temel gösterimler Şekil 3'de sunulmuştur. Dikkate alınan problemin tipine göre, bu operatörlerin uygulanmasında bir takım değişikliklerin yapılması da kaçınılmazdır. Bu çalışmada kullanılan genetik operatörlerin uygulanmasında Matlab 6.5 programlama paketi içerisinde mevcut olan standart tanımlamalar kullanılmıştır.

Her iki algoritma için kullanılan tanımlamalar Tablo 1 ve Tablo 2'de verilmiştir. GGA ve GA kullanılarak elde edilen model parametrelerinin değerleri sırasıyla Tablo 3 ve Tablo 4'de verilmiştir.



#### GA için Mutasyon

önce: (1 0 1 | 1 0 1 1 0)  
sonra: (0 1 1 | 0 0 1 1 0)

#### GGA için Mutasyon

önce: (1.38 -69.4 | 326.44 0.1)  
sonra: (1.38 -67.5 | 326.44 0.1)  
(b)

Şekil 3: (a) GGA için tipik çaprazlama işlemi  
(b) GA ve GGA için mutasyon işlemi

Tablo 1: GGA kontrol parametre değerleri

Gerçek Kodlu Genetik Algoritma (GGA)
♦ popülasyon büyüklüğü = 30
♦ çaprazlama oranı = 0.9
♦ mutasyon oranı = 0.1
♦ jenerasyon sayısı = 200
♦ çaprazlama tipi = tek noktalı
♦ seçim operatörü = rulet tekerleği
♦ değişken sayısı = 4 (L, a, b, c)
♦ değişken sınırları= L ∈ [0,5], a ∈ [1,10] b ∈ [1,10], c ∈ [1,10]

Tablo 2: GA kontrol parametre değerleri

İkili Kodlu Genetik Algoritma (GA)
♦ popülasyon büyüklüğü = 30
♦ çaprazlama oranı = 0.9
♦ mutasyon oranı = 0.1
♦ jenerasyon sayısı = 200
♦ çaprazlama tipi = tek noktalı
♦ seçim operatörü = rulet tekerleği
♦ değişken sayısı = 4 (L, a, b, c)
♦ her değişken için bit sayısı= 8
♦ değişken sınırları= L ∈ [0,5], a ∈ [1,10] b ∈ [1,10], c ∈ [1,10]

Daha önce, bu çalışmada ele alınan sistem kimliklendirme probleminde tabu araştırma algoritması (TAA) kullanılarak da çözüm araştırılmıştır [21]. Karşılaştırma yapma imkânı sağlaması bakımından literatürde yer alan bu çalışmaya ait değerler de burada sunulmuştur. Tablo 5 TAA kullanılarak elde edilen parametre değerlerini göstermektedir.

## 4. Tartışma

Tablo 3-5'den de görüleceği gibi, algoritma performansının test edilmesi amacıyla üç farklı başlangıç noktası dikkate alınmıştır. Bunlar minimum ( $L=0$ ,  $a=b=c=1$ ), ortalama ( $L=2.5$ ,  $a=b=c=5$ ) ve maksimum ( $L=5$ ,  $a=b=c=10$ ) başlangıç noktalarıdır. Bu noktalar, algoritmaların araştırmaya ilk olarak başladıkları parametre bölgelerini ifade eder. Her bir

algoritmanın farklı süreçler için elde ettiği en iyi parametre değerleri bu tablolarda koyu renkli olarak gösterilmiştir.

Modellerin kimliklendirilmesi sırasında farklı algoritmalar tarafından elde edilen en düşük ISE hata değerleri Tablo 6'da karşılaştırmalı olarak yeniden özetlenmiştir.

Tablo 3: GGA ile elde edilen model parametre değerleri

Süreç	Model	Başlangıç noktası	Zaman (sn.)	L	a	b	c	J (ISE)
G1	Gm1	minimum	<b>75.05</b>	<b>0.9376</b>	<b>1.5954</b>	<b>1.0000</b>	-	<b>0.0038</b>
		ortalama	73.52	1.0219	1.6362	1.0000	-	0.0052
		maksimum	72.13	0.9395	2.0519	1.0044	-	0.0322
	Gm2	minimum	<b>78.27</b>	<b>0.3906</b>	<b>1.0000</b>	<b>2.1169</b>	<b>1.0000</b>	<b>0.0018</b>
		ortalama	80.44	0.0783	2.4345	1.8525	1.0176	0.0484
		maksimum	90.31	1.0046	1.0114	3.2164	1.0000	0.3300
G2	Gm1	minimum	<b>76.44</b>	<b>0.3125</b>	<b>1.6998</b>	<b>1.0000</b>	-	<b>0.0064</b>
		ortalama	78.64	0.5120	1.6419	1.0000	-	0.0076
		maksimum	78.72	0	2.3951	1.0000	-	0.0206
	Gm2	minimum	<b>87.27</b>	<b>0</b>	<b>1.0000</b>	<b>2.1606</b>	<b>1.0000</b>	<b>0.0112</b>
		ortalama	87.00	0	1.0000	1.6152	1.0000	0.0598
		maksimum	86.00	0.0783	7.1037	3.2676	1.0000	0.5893
G3	Gm1	minimum	<b>77.74</b>	<b>0.7818</b>	<b>1.2815</b>	<b>1.0000</b>	-	<b>0.0035</b>
		ortalama	77.16	0.9398	1.0000	1.0000	-	0.0109
		maksimum	78.59	0.9789	1.0000	1.0011	-	0.0120
	Gm2	minimum	<b>83.09</b>	<b>0.1563</b>	<b>1.0000</b>	<b>1.7207</b>	<b>1.0009</b>	<b>0.0026</b>
		ortalama	89.45	0.4735	1.0000	1.2813	1.0000	0.0356
		maksimum	86.78	1.0809	1.7184	1.0000	1.0000	0.4187
G4	Gm1	minimum	<b>77.81</b>	<b>1.5332</b>	<b>1.0000</b>	<b>1.0000</b>	-	<b>0.3007</b>
		ortalama	79.09	1.5400	1.0000	1.0000	-	0.3007
		maksimum	78.66	1.3658	1.2683	1.0000	-	0.3141
	Gm2	minimum	<b>85.38</b>	<b>0</b>	<b>4.0938</b>	<b>1.5804</b>	<b>1.0025</b>	<b>0.0032</b>
		ortalama	87.33	0.5175	3.6816	1.2043	1.0000	0.0564
		maksimum	86.47	1.2401	3.3834	1.0000	1.0000	0.3242

Tablo 4: GA ile elde edilen model parametre değerleri

Süreç	Model	Başlangıç noktası	Zaman (sn.)	L	a	b	c	J (ISE)
G1	Gm1	minimum	40.025	1.2549	1.2824	1.0000	-	0.0117
		ortalama	<b>39.53</b>	<b>1.0392</b>	<b>1.6353</b>	<b>1.0000</b>	-	<b>0.0057</b>
		maksimum	39.52	1.3333	1.3529	1.0000	-	0.0188
	Gm2	minimum	<b>38.84</b>	<b>0.4706</b>	<b>1.0706</b>	<b>2.1294</b>	<b>1.0000</b>	<b>0.0017</b>
		ortalama	39.02	0.4118	1.1765	2.4824	1.0000	0.0230
		maksimum	41.69	0.4314	2.7647	1.9882	1.0353	0.0822
G2	Gm1	minimum	<b>39.45</b>	<b>0.3137</b>	<b>1.8471</b>	<b>1.0000</b>	-	<b>0.0035</b>
		ortalama	41.31	0.3922	1.5647	1.0000	-	0.0095
		maksimum	39.70	0.3922	1.8118	1.0000	-	0.0043
	Gm2	minimum	<b>42.11</b>	<b>0</b>	<b>1.0000</b>	<b>2.1294</b>	<b>1.0000</b>	<b>0.0114</b>
		ortalama	41.75	0.6275	2.2706	1.3882	1.1059	0.4716
		maksimum	41.74	0.0784	3.6471	2.4471	1.0353	0.2220
G3	Gm1	minimum	<b>44.09</b>	<b>0.7059</b>	<b>1.3529</b>	<b>1.0000</b>	-	<b>0.0043</b>
		ortalama	42.34	0.6275	1.4941	1.0353	-	0.0446
		maksimum	42.84	0.6078	1.2824	1.0000	-	0.0137
	Gm2	minimum	<b>40.08</b>	<b>0.0784</b>	<b>1.0000</b>	<b>1.8471</b>	<b>1.0000</b>	<b>0.0043</b>
		ortalama	40.33	0.6078	2.0235	1.1412	1.0000	0.2698
		maksimum	40.13	0.6863	2.3412	2.5529	1.0353	0.3812
G4	Gm1	minimum	<b>42.64</b>	<b>0.4118</b>	<b>1.0000</b>	<b>1.0000</b>	-	<b>0.3076</b>
		ortalama	44.17	1.0980	1.0353	1.0000	-	0.3468
		maksimum	42.80	1.3922	1.2471	1.0000	-	0.3138
	Gm2	minimum	<b>42.08</b>	<b>0.0392</b>	<b>3.2588</b>	<b>1.6706</b>	<b>1.0000</b>	<b>0.0339</b>
		ortalama	41.61	0.3529	3.7529	1.7765	1.0706	0.2016
		maksimum	41.45	0.3922	3.1176	1.7412	1.0000	0.0589

Tablo 5: TAA ile elde edilen model parametre değerleri [9]

Süreç	Model	Başlangıç noktası	L	a	b	c	J (ISE)
G1	Gm1	minimum	<b>1.2549</b>	<b>1.3176</b>	<b>1.0000</b>	-	<b>0.0112</b>
		ortalama	0.6078	2.1294	1.0000	-	0.0224
		maksimum	0.6078	1.9529	1.0000	-	0.0181
	Gm2	minimum	<b>0.3529</b>	<b>1.2824</b>	<b>2.1294</b>	<b>1.0000</b>	<b>0.0006</b>
		ortalama	0.1569	2.1294	2.2706	1.0000	0.0071
		maksimum	0.2941	1.3882	2.0941	1.0000	0.0019
G2	Gm1	minimum	<b>0.1569</b>	<b>2.1294</b>	<b>1.0000</b>	-	<b>0.0086</b>
		ortalama	0.1569	2.1294	1.0000	-	0.0086
		maksimum	0.4510	1.5294	1.0000	-	0.0095
	Gm2	minimum	<b>0.0000</b>	<b>1.0000</b>	<b>2.1647</b>	<b>1.0000</b>	<b>0.0112</b>
		ortalama	0.0000	1.0000	2.1647	1.0000	0.0112
		maksimum	0.0000	1.0000	2.0941	1.0000	0.0119
G3	Gm1	minimum	1.2549	1.0000	1.0000	-	0.0379
		ortalama	0.3137	2.1294	1.0000	-	0.0515
		maksimum	<b>0.6078</b>	<b>1.4588</b>	<b>1.0000</b>	-	<b>0.0080</b>
	Gm2	minimum	0.0000	1.3176	2.1294	1.0000	0.0112
		ortalama	0.0000	1.5647	2.1294	1.0000	0.0132
		maksimum	<b>0.1373</b>	<b>1.1059</b>	<b>1.8118</b>	<b>1.0000</b>	<b>0.0017</b>
G4	Gm1	minimum	<b>1.5686</b>	<b>1.0000</b>	<b>1.0000</b>	-	<b>0.3012</b>
		ortalama	0.6275	2.1294	1.0000	-	0.4423
		maksimum	1.2353	2.1218	1.0000	-	0.3115
	Gm2	minimum	<b>0.0196</b>	<b>3.9647</b>	<b>1.5647</b>	<b>1.0000</b>	<b>0.0014</b>
		ortalama	0.0392	3.8588	1.5647	1.0000	0.0014
		maksimum	0.2941	3.2235	1.4235	1.0000	0.0107

Tablo 6: Farklı algoritmalar kullanılarak elde edilen en düşük ISE hata değerleri

Süreç	ISE		
	GGA	GA	TAA
G1	0.0018	0.0017	0.0006
G2	0.0064	0.0035	0.0086
G3	0.0026	0.0043	0.0017
G4	0.0032	0.0339	0.0014

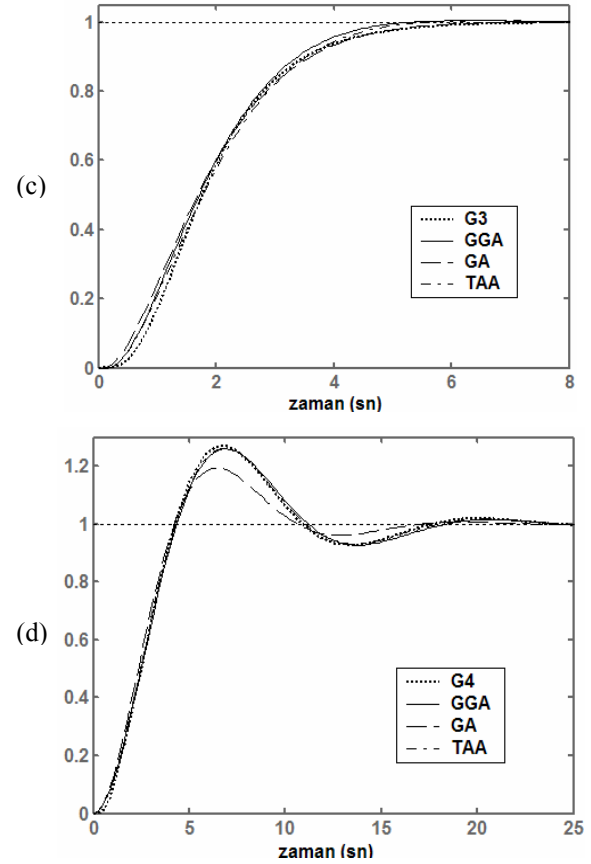
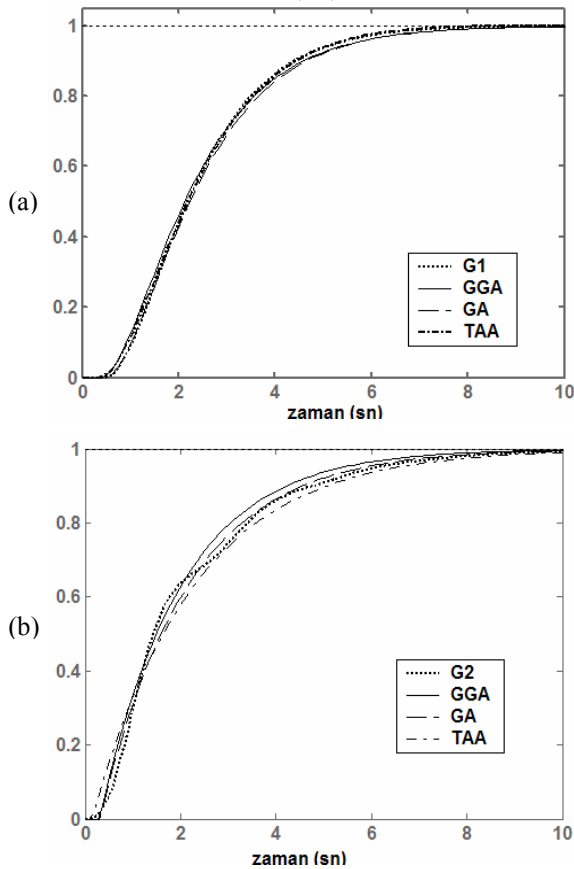
Tablo 3-5 incelendiğinde, yüksek dereceli sistemlerin modellenmesinde ikinci derece model yaklaşımının daha etkili sonuçlar verdiği görülebilir. G2 sürecinin modellenmesinde ise birinci dereceli model kullanılarak ikinci dereceli model kullanımına göre biraz daha iyi sonuçlar elde edilmiştir. Üstelik bu durum bu çalışmada dikkate alınan algoritmaların tamamında geçerliliğini korumuştur. G2 süreci için birinci dereceli model kullanıldığında GGA, GA ve TAA için sırasıyla 0.0064, 0.0035 ve 0.0086 minimum ISE değerlerine ulaşılmıştır (Tablo 6).

Tablolarda verilen sonuçlar incelendiğinde, GGA kullanılarak GA'ya göre genellikle düşük hataya sahip daha iyi değerler elde edilmiştir. Modellenen sistemin derecesi arttığında elde edilen hatalar arasındaki farkın daha da belirginleştiğini görmek mümkündür (Tablo 3-6). Buna karşılık ikili kodlanmış GA kullanılarak GGA'ya göre daha kısa sürelerde kimliklendirme yapıldığı da tablolardan açıkça görülmektedir. Diğer yandan tablolardaki sonuçlar, sistem kimliklendirme için TAA'nın etkili biçimde kullanılabilirliğini de göstermektedir. Özellikle G1 süreci için TAA kullanılarak elde edilen ISE değeri etkileyicidir (Tablo 6). G3 ve G4 süreçleri için de yine TAA'nın özellikle GA'ya göre minimum ISE hatası yönünden belirli bir üstünlüğü görülmektedir. Bu iki süreç için GGA kullanıldığında elde edilen ISE değerlerinin TAA ile bulunan değerlerden çok fazla farklı olmadığını da belirtmek gerekir. G2 sürecinde ise GGA ve GA ile TAA'na göre daha iyi değerlere ulaşılmıştır (Tablo 3-6).

Farklı algoritmalar tarafından optimize edilen modellerin birim basamak girişe karşılık zaman domeni yanıtları Şekil 4’de sunulmuştur. Bu şekilde yer alan grafik gösterimler, her bir süreç için Tablo 6’da verilen değerleri sağlayan model yapıları kullanılarak elde edilmiştir. Örneğin, G1 süreci için TAA kullanıldığında 0.0006 ISE değerini sağlayan  $L=0.3529$ ,  $a=1.2824$ ,  $b=2.1294$  ve  $c=1.00$  değerlerine sahip ikinci derece modelden yararlanılmıştır (Tablo 5).

Şekil 4’de verilen zaman domeni yanıtları bir kez daha kanıtlamaktadır ki, farklı derecelerdeki doğrusal sistemlerin tanımlanmasında düşük dereceli modeller başarılı şekilde kullanılabilir. Kullanılan modelin başarısında model parametrelerinin kimliklendirilmesi için yararlanılan yöntem temel rol oynamaktadır. Gerçek sistemler ile kimliklendirilen modellere ait zaman domeni yanıtları arasında Şekil 4’de gözlenen dikkat çekici uyum, GGA’ların bu amaçla kullanılabilir etkili yöntemlerden birisi olduğu gerçeğini açıkça ortaya koymaktadır.

Elbette ki, bir sistemin kimliklendirilmesinde dikkate alınması gereken en önemli faktörlerden birisi de, bu sisteme ait frekans cevabı özellikleridir. İyi yapılmış bir kimliklendirmede gerçek sistem ile model arasında kazanç payı ve faz payı gibi önemli frekans özellikleri bakımından da bir uyum bulunmalıdır. Yapılan bu çalışmada sistemlerin frekans özellikleri dikkate alınmamış ve model parametreleri zaman domeni yanıtlarına göre belirlenmeye çalışılmıştır. Buna karşılık, burada yer verilmemiş olmasına rağmen, elde edilen modellerin ve gerçek sistemlerin frekans yanıtları arasında da dikkate değer bir benzerlik ve uyum bulunmuştur. İleri çalışmalarda sistemlerin bu özelliklerinin de incelenmesi ve ortaya konulması planlanmaktadır.



Şekil 4: (a) G1, (b) G2, (c) G3 ve (d) G4 süreçleri için farklı algoritmalar tarafından elde edilen modellere ait birim basamak yanıtları

## 5. Sonuçlar

Bu çalışmada, sistemlerin kimliklendirilmesi işleminde gerçek kodlu genetik algoritma kullanımı incelenmiştir. Doğrusal sistemlerin dikkate alındığı çalışmada literatürde yer alan farklı derecelere sahip süreçlerin önerilen model yapıları ile tanımlanması amaçlanmıştır. Model yapılarındaki farklı sayıdaki parametrelerin gerçek kodlu genetik algoritma tarafından belirlenmesine çalışılmıştır. Benzetim çalışmalarının sonuçları, ikili kodlu genetik algoritma ve tabu araştırma algoritmaları ile karşılaştırmalı biçimde sunulmuştur. Elde edilen benzetim sonuçları, gerçek kodlu genetik algoritmaların sistem kimliklendirme ve model uydurma çalışmalarında oldukça tatmin edici bir performans sergileyebileceğini ortaya koymuştur.

## 6. Teşekkür

Bu çalışma; Erciyes Üniversitesi Bilimsel Araştırma Projeleri Birimi tarafından EÜBAP-FBT-06-58 proje kod numarası ile desteklenmiştir.

## 7. Kaynakça

- [1] Solbrand, G., Ahlen, A., Ljung, L., "Recursive Methods for Off-Line Identification", *Int. Journal of Control*, 41(1):177-191, 1985.
- [2] Ljung, L., *System Identification*. Prentice-Hall Englewood Cliffs, NJ., 1987.
- [3] Söderström, T., Stoica, P., *System Identification*, Prentice-Hall Englewood Cliffs NJ., 1989.
- [4] Pintelon, R., Guillaume, P., Rolain, Y., Verbeyst, F., "Identification of Linear Systems Captured in a Feedback Loop", *IEEE Trans. on Inst. and Measurement*, 41(6):747-754, 1992.
- [5] Pintelon, R., Guillaume, P., Rolain, Y., Schoukens, J., VanHamme, H., "Parametric Identification of Transfer Functions in the Frequency Domain-A Survey", *IEEE Trans. on Automatic Control*, 39(11):2245-2260, 1994.
- [6] Schoukens, J., Pintelon, R., VanHamme, H., "Identification of Linear Dynamic Systems using Piecewise Constant Excitations: Use, Misuse and Alternatives", *Automatica*, 30(7):1153-1169, 1994.
- [7] Schoukens, J., Pintelon, R., Vandersteen, G., Guillaume, P., "Frequency Domain System Identification using Nonparametric Noise Models Estimated from A Small Number of Data Sets", *Automatica*, 33(6):1073-1086, 1997.
- [8] Rolain, Y., Schoukens, J., Pintelon, R., "Order Estimation for Linear Time-Invariant Systems using Frequency Domain Identification Methods", *IEEE Trans. on Automatic Control*, 42(10), 1997.
- [9] Bagis, A., "Performance comparison of genetic and tabu search algorithms for system identification", *Lecture Notes in Computer Science* (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 4251 LNAI - I, pp. 94-101, 2006.
- [10] Ahmed, S., Huang, B., Shah, S. L., "Novel Identification Method From Step Response", *Control Engineering Practice*, 15(5), 545-556, May 2007.
- [11] Pham, D.T., Karaboga, D., "Intelligent Optimisation Techniques: Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks", Springer-Verlag, 2000.
- [12] Chang, W.-D., "Nonlinear System Identification and Control Using A Real-Coded Genetic Algorithm", *Applied Mathematical Modelling*, 31:541-550, 2007.
- [13] Liu, Z., Liu, A., Wang, C., Niu, Z., "Evolving Neural Network Using Real-coded Genetic Algorithm For Multispectral Image Classification", *Future Gene. Comput. System*, 20:1119-1129, 2004.
- [14] Jain, A., Srinivasalu, S., Bhattacharjya, R.K., "Determination of an Optimal Unit Pulse Response Function Using Real-Coded Genetic Algorithm", *Journal of Hydrology*, 303:199-214, 2005.
- [15] Ha, J.-L., Kung Y.-S., Fung, R.-F., Hsien, S.-C., "A Comparison of Fitness Functions For The Identification of A Piezoelectric Hysteretic Actuator Based On The Real-Coded Genetic Algorithm", *Sensors and Actuators*, 132:643-650, 2006.
- [16] Khalik, M.A., Sherif, M., Saraya S., Areed F., "Parameter Identification Problem: Real-coded GA Approach", *Applied Math. And Computation*, 2006.
- [17] Wang, Q.-G., "PID Tuning For Improved Performance", *IEEE Transactions On Control Systems Technology*, 7(4):457-465, 1999.