

Doğrusal parametrik ve doğrusal olmayan gerçek sistemlerin yapay arı kolonisi algoritması kullanılarak modellenmesi

Şaban ÖZER¹, İbrahim BARAN^{2,*}

¹ Erciyes Üniversitesi, Elektrik Elektronik Mühendisliği Bölümü, 38039, Melikgazi, Kayseri

² Dicle Üniversitesi Çüngüş Mehmet Adıgüzel Meslek Yüksekokulu, Elektrik Programı, 21870, Çüngüş, Diyarbakır

Makale Gönderme Tarihi: 25.06.2014

Makale Kabul Tarihi: 15.11.2014

Oz

Araştırmacılar, klasik optimizasyon tekniklerinin kullanılmasının yetersiz olduğu problemler için farklı çözüm arayışlarına girmiş, çalışmalarını bu yönde devam ettirmişlerdir. Yapay zeka optimizasyon teknikleri ise bu çalışmaların bir sonucu olarak ortaya çıkmıştır. Yapay zeka optimizasyon teknikleri, karmaşık optimizasyon problemlerinin çözümünde sık sık kullanılmaya başlanmış ve olumlu sonuçlar verdiği görülmüştür. Doğadan esinlenen algoritmaların yeni bir dalı olan sürü zekası yaklaşımı, böceklerin içgüdüsel problem çözüme becerilerini kullanan etkili metasezgisel yöntemler geliştirebilmek için böcek davranışlarının modellenmesine odaklanmıştır. Yapay Arı Kolonisi (YAK) Algoritması da sürü zekasına sahip olan arıların davranışlarını esas alıp geliştirilmiş ve karmaşık problemlerin çözümünde kullanılmaya başlanmıştır.

Bu çalışmada da, doğrusal olan parametrik ve doğrusal olmayan gerçek sistemlerin sürü zekası yaklaşımına örnek olan yapay zeka optimizasyon tekniklerinden Yapay Arı Kolonisi (YAK) Algoritması ile modellenmesi gerçekleştirilmiş olup, Genetik algoritma (GA) ve Klonal Seçme Algoritması (KSA) ile başarımları karşılaştırılmıştır. Benzetim çalışmalarında literatürde kıyaslama problemlerinde sıklıkla kullanılan bir adet doğrusal parametrik ve bir adet doğrusal olmayan gerçek sistem bu algoritmalar yardımıyla modellendirilmiştir. Benzetim sonuçlarına göre doğrusal parametrik sistemlerin modellenmesinde YAK algoritması, GA'ya çok yakın bir sürede parametre tahmini yapmış, farklı koloni sayılarında ise hem GA'dan hemde KSA'dan daha düşük modelleme hatası ile sonuç alındığı görülmüştür. Doğrusal olmayan gerçek sistem modellemelerde ise aynı durum söz konusu olup, koloni sayılarındaki değişime bağlı olarak YAK algoritması, GA ve KSA'na göre daha düşük hata ve daha erken sürede sistemi modelleyebildiği görülmüştür.

Anahtar Kelimeler: Yapay arı kolonisi algoritması, doğrusal sistem modelleme, doğrusal olmayan gerçek sistem modelleme

*Yazışmaların yapılacağı yazar: İbrahim BARAN. ibaran@dicle.edu.tr; Tel: + 90 (412) 241 10 00 - 7827

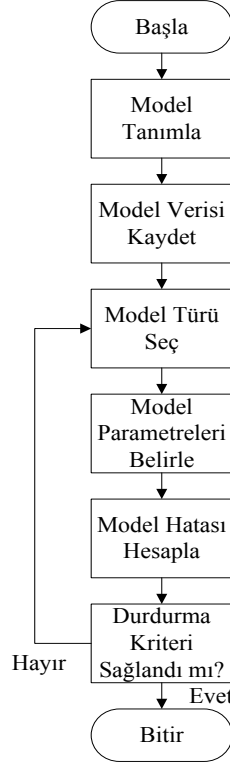
Giriş

Ortak bir amacı gerçekleştirmek veya başarmak amacıyla bir araya getirilen, ortak özellikleri olan, girdilere cevap olarak çıktı üreten, aralarında karşılıklı etkileşim olan elemanların oluşturduğu yapılar sistem denir. Sistem kavramının kullanılmasındaki temel neden, istenilen sonuca ulaşmak üzere sistemi incelemek, sistemle ilgili temel ilkeleri belirlemek ve mümkünse sistemi amaçlar doğrultusunda düzenlemektir. Bu amacı gerçekleştirmek için ise sistemi en iyi şekilde temsil edecek bir aracıya ihtiyaç duyulur. Bu aracı ise modeldir. Model daha basit tanımıyla sistemin temsilidir ve kısaca sistemin basitleştirilmiş, genellikle matematiksel veya hesaplanabilir yapıdaki gösterimidir. (Söderstrom, 1989; Zorlu ve Özer, 2009; Özer ve Baran, 2012).

Sistemlere ait modeller, yalnızca sistemin giriş işareti kullanılarak, yalnızca sistemin çıkış işareti kullanılarak veya hem giriş hem de çıkış işaretleri kullanılarak elde edilebilir. Bu modeller bulunurken parametreleri belirlenen modelin girişine, sisteme uygulanmış olan giriş işareti uygulanıp, modelden alınan çıkış işareti ile sistemin gerçek çıkışı arasındaki fark bulunur. Eğer fark büyükse, başka bir model yapısı veya yeni bir parametre tespit yöntemi belirlenir. Eğer fark yeterince küçükse bulunan bu model sistemi tanımlamak ve kontrol etmek için kullanılabilir. Sistem modelleme adımları Şekil 1'deki gibi özetlenebilir (Kaplan, 1995; Özer ve Baran, 2012).

Bu çalışmada da biri doğrusal olan parametrik diğeri ise doğrusal olmayan gerçek sistemin, literatüre yeni girmiş sayılabilecek ve sürü zeka yaklaşımının en önemli örneklerinden olan YAK algoritması ile modellenmesi amaçlanmıştır. Farklı birçok alanda yapılan çalışmalarda YAK algoritmasının çok iyi sonuçlar verdiği bilinmekle beraber, bu algoritmanın performansının doğrusal ve doğrusal olmayan sistemlerin modelleme çalışmalarında ne şekilde bir performans göstereceği hakkında bilgi edinilmeye çalışılmıştır. Ayrıca kullanılan örnek

sistemlerin modellemeleri yapılırken, YAK algoritmasının performansı, GA ve KSA ile karşılaştırılmıştır. Karşılaştırılma yapılırken, algoritmaların bulduğu parametreler, parametre bulma süreleri, MSE değerleri, MSE değerlerinin standart sapması ve hata yaklaşım eğrileri dikkate alınmıştır.



Şekil 1: Modelleme işlemi adımları

Yapay arı kolonisi algoritması

Doğadan esinlenen algoritmaların yeni bir dalı olan sürü zekâsı yaklaşımı, böceklerin içgüdüsel problem çözme becerilerini kullanan etkili metasezgisel yöntemler geliştirebilmek için böcek davranışlarının modellenmesine odaklanmıştır. Karaboğa tarafından geliştirilen YAK algoritması da sürü zeka yaklaşımına bir örnektir (Karaboğa, 2005; Karaboğa ve Baştürk, 2007; Karaboğa ve Baştürk, 2008).

Bal arılarında amaç kovan etrafında bulunan nektar kaynaklarından toplanan, en iyi ve en çok miktardaki nektarı kovana getirmektir. Bu işi yerine getirmek üç arı grubu çalışmaktadır ve bunlar işçi, gözcü ve kaşif arılardır. Öncelikle kâşif arılar probleme yönelik olarak rastgele belirlenen çözüm noktalarına yerleştiriliyor ve artık birer kaynakları olduğundan işçi arı durumuna geçiyorlar. Her bir nektar kaynağı için bir işçi arı bulunmaktadır. Dolayısıyla işçi arıların sayısı nektar kaynağı sayısına eşittir. Her bir işçi arı elindeki çözümden, diğer arılardan aldığı bilgiler ışığında alternatif çözümler üretiyor. Elindeki çözümden (nektar kaynağı) daha iyi bir çözüm bulduğunda ise diğerini terk edip bu kaynağa yöneliyor. Daha sonra devreye gözcü arılar giriyor ve işçi arıların kaynak hakkında verdikleri bilgileri kullanarak bu arıların gittikleri kaynakların etrafındaki kaynaklara (çözümlere) yöneliyorlar. Eğer kaynak (çözüm) işe yaramaz ise yapay kâşif arılar başka kaynaklar (çözümler) bulmak üzere yeniden işbaşı yapıyor (Karaboğa ve Akay, 2010; Karaboğa 2011).

Rastgele yer üretme süreci her bir parametrenin alt ve üst sınırları arasında rastgele üreterek gerçekleşir ve bu durum Denklem 1’de gösterilmektedir (Akay, 2009).

$$x_{i,j} = x_j^{\min} + rand(0,1)(x_j^{\max} - x_j^{\min}) \quad (1)$$

Burada $i=1\dots SN$, $j=1\dots D$ ve SN yiyecek kaynağı sayısı ve D optimize edilecek parametre sayısıdır. İşçi arı yeni bir arı kaynağı belirleyerek bunun kalitesini belirler ve yeni kaynağın yeri Denklem 2’deki gibi belirlenir (Akay, 2009).

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (2)$$

Burada ϕ , $[-1,1]$ aralığında üretilen rastgele değerlerdir. v_{ij} ’nin daha önceden belirlenen parametre sınırları dışına çıkması durumunda Denklem 3’e göre öteleme yapılır ve yeni kaynağın kalitesi hesaplanarak uygunluk değeri Denklem 4’e göre hesaplanır (Akay, 2009).

$$v_{ij} = \begin{cases} x_j^{\min}, v_{ij} < x_j^{\min} \\ v_{ij}, x_j^{\min} < v_{ij} < x_j^{\max} \\ x_j^{\max}, v_{ij} > x_j^{\max} \end{cases} \quad (3)$$

$$fitness_j = \begin{cases} 1/(1 + f_i), f_i \geq 0 \\ 1 + abs(f_i), f_i < 0 \end{cases} \quad (4)$$

Kaynağın miktarı kaynağın seçilme olasılığını etkiler ve x_{ij} pozisyonundaki kaynağın seçilme olasılığı Denklem 5’teki gibi ifade edilir.

$$p_i = \frac{fitness_i}{\sum_{i=1}^{SN} fitness_i} \quad (5)$$

Denklem 5’teki $fitness_i$, i. kaynağın kalitesini, SN görevli arı sayısını göstermektedir. nektar miktarına karşılık gelir. Bu olasılık hesaplama işlemine göre bir kaynağın nektar miktarı arttıkça, bu kaynak bölgesini seçecek gözcü arı sayısı da artacaktır (Akay, 2009).

YAK algoritmasının temel adımları şu şekilde özetlenebilir:

- İşçi arıları kaynaklara gönder ve nektar miktarlarını hesapla
- Gözcü arıları kaynaklara gönder ve nektar miktarlarını hesapla
- Rastgele yeni kaynaklar bulmaları için kaşif arıları gönder
- O ana kadarki en iyi kaynağı hafızada tut
- Durdurma kriteri sağlanana dek devam et.

Uygulamaların yapıldığı bilgisayar

Uygulamalar yapılırken kullanılan bilgisayara ait özellikler aşağıdaki gibidir. Farklı özellikteki bilgisayarların kullanılması süre olarak sonuçları etkileyebilir.

İşletim sistemi: Windows 7 Ultimate 32 bit
İşlemci hızı: i5-2410M CPU @ 2.30 Ghz
Bellek: 4096 MB Ram

Örnek1: doğrusal parametrik sistem

Çalışmanın ilk örneği olarak Denklem 6'da belirtilen lineer sistem ele alınmış ve algoritmaların bu sisteme ait parametreleri bulma sonuçları karşılaştırılmıştır (Baran, 2001; Özer vd, 2003).

$$y(n) = 0.0798 u(n) + 0.0791 u(n-1) + 0.0791 u(n-2) + 0.0798 u(n-3) + 1.5558y(n-1) - 1.270 y(n-2) + 0.3983 y(n-3) \quad (6)$$

Model parametrelerinin tahmini amacıyla parametre kestirim çalışmalarında giriş verisi $u(n)$, hem sistem hem de model girişi olarak kullanılmıştır. Giriş dizisi $n=100$ adet veriden oluşan Gaussian dağılımlı beyaz gürültüdür. Tablo 1'de algoritmaların parametre bulmaları için kullanılan kontrol değerleri verilmiştir.

Tablo 1. Denklem 6 için kullanılan kontrol değerleri

| Kontrol Parametreleri | YAK | GA | KSA |
|-----------------------------|-------|-------|-------|
| Parametre araştırma aralığı | -2 +2 | -2 +2 | -2 +2 |
| Popülasyon boyutu | - | 30 | 30 |
| Koloni sayısı | 30 | - | - |
| Klonlama oranı | - | - | 0.1 |
| Mutasyon oranı | - | 0.3 | - |
| Çaprazlama oranı | - | 0.8 | - |
| İterasyon | 100 | 100 | 100 |

Kullanılan kontrol parametreleri ile her bir algoritma 30 kez tekrarlatılmış ve algoritmaların bulduğu parametreler Tablo 2'de verilmiştir.

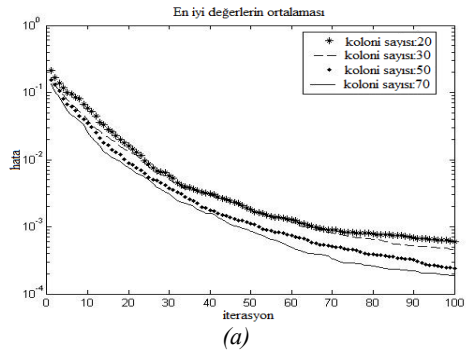
Tablo 2. Denklem 6 için algoritmaların bulduğu parametreler

| Gerçek Değer | YAK | GA | KSA |
|--------------|------------------------|-----------------------|-----------------------|
| 0.0798 | 0.0878 | 0.0390 | 0.0608 |
| 0.0791 | 0.1349 | 0.0941 | 0.0611 |
| 0.0791 | 0.1154 | 0.1159 | 0.0957 |
| 0.0798 | 0.0928 | 0.0779 | 0.1025 |
| 1.5558 | 1.1002 | 1.4406 | 1.6832 |
| -1.270 | -0.5940 | -1.0797 | -1.5893 |
| 0.3983 | 0.0343 | 0.3581 | 0.6267 |
| Ort.Süre(sn) | 3.7384 | 3.6515 | 7.1880 |
| MSE | 2.070×10^{-4} | 4.73×10^{-4} | 1.95×10^{-4} |
| Std. sap. | 3.040×10^{-4} | 1.6×10^{-3} | 3.8×10^{-3} |

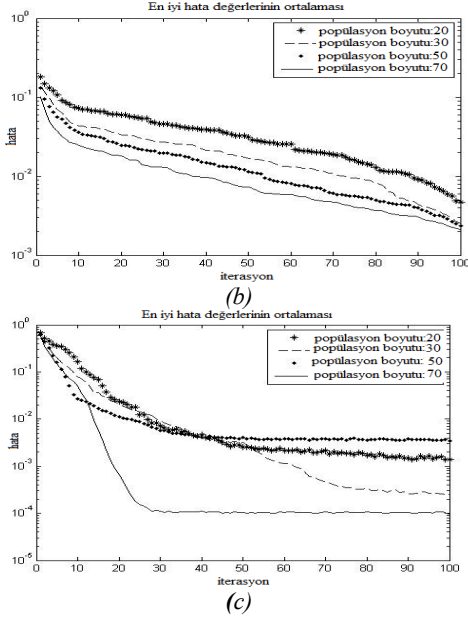
Ayrıca YAK algoritması için koloni sayısı, GA ve KSA'sı için popülasyon değerlerinin 20, 30, 50 ve 70 olması durumunda bu değerlerin algoritmaların parametre bulma sürelerine, en düşük MSE değerlerine ve MSE değerlerinin standart sapmasına etkisi incelenmiş olup, bu durum Tablo 3'te belirtilmiştir. Aynı zamanda koloni ve popülasyon sayılarının belirtilen değerlerde olması durumunda algoritmalara ait hata yaklaşım eğrileri aşağıdaki şekilde verilmiştir (Şekil 2).

Tablo 3. Koloni ve popülasyon sayılarının ortalama süre ve hatalara etkisi

| Araştırılan değer | YAK | GA | KSA |
|--|------------------------|-----------------------|-----------------------|
| Koloni sayısı =20, Popülasyon sayısı = 20 | | | |
| Süre(sn) | 3.2720 | 2.4825 | 3.86 |
| MSE | 8.897×10^{-4} | 3.8×10^{-3} | 1.2×10^{-3} |
| Std. sap. | 5.772×10^{-4} | 3.60×10^{-3} | 1×10^{-1} |
| Koloni sayısı = 30, Popülasyon sayısı = 30 | | | |
| Süre(sn) | 3.7384 | 3.6515 | 7.1880 |
| MSE | 2.070×10^{-4} | 4.34×10^{-4} | 1.95×10^{-4} |
| Std. sap. | 3.040×10^{-4} | 1.60×10^{-3} | 3.8×10^{-3} |
| Koloni sayısı = 50, Popülasyon sayısı = 50 | | | |
| Süre(sn) | 6.0673 | 5.7881 | 16.9484 |
| MSE | 7.905×10^{-5} | 1.1×10^{-3} | 3×10^{-3} |
| Std. sap. | 2.164×10^{-4} | 1.4×10^{-3} | 2.6×10^{-3} |
| Koloni sayısı = 70, Popülasyon sayısı = 70 | | | |
| Süre(sn) | 8.451 | 8.1558 | 31.0797 |
| MSE | 1.064×10^{-4} | 7.84×10^{-4} | 7.04×10^{-5} |
| Std. sap. | 1.835×10^{-4} | 1.60×10^{-3} | 1.9×10^{-3} |



Doğrusal parametrik ve doğrusal olmayan gerçek sistemlerin yapay arı kolonisi algoritması kullanılarak modellenmesi



Şekil 2. Hata yaklaşım eğrileri: (a) YAK algoritması, (b) GA, (c) KSA

Örnek2: doğrusal olmayan sistem

Çalışmada ikinci olarak kullanılan sistem verileri de gerçek sistem verileri olup, bu verilerde Katholieke Universiteit Leuven, Signals, Identification, System Theory and Automation's Identification Database'den alınmış, NASA'nın F-18 modeli bir hava araştırma aracına ait kanat sarsıntılarında alınan verilerdir (De Moor B.L.R., DaISy; Feron vd., 1998).

Bu sistem için Denklem 7' de önerilen model yapısı uygulanmış ve parametreler ile sistem MSE' leri hesaplanmıştır. Önerilen model parametrelerinin hesaplanması için 1024 adet datadan ilk 300 data, bulunan parametrelerin testi için ise sonraki 100 adetlik data kullanılmıştır. Tablo 4'te algoritmaların parametre bulmaları için kullanılan kontrol değerleri verilmiştir.

$$y(n) = a_{1,1} y(n-1)u(n-1) + a_2 y(n-2) + b_3 u(n-3) + b_4 u(n-4) + c_{3,3} y(n-3) u(n-3) + c_{4,2} y(n-4) u(n-2) + c_{1,2} y(n-1) u(n-2) \quad (7)$$

Tablo 4. Denklem 7 için kullanılan kontrol değerleri

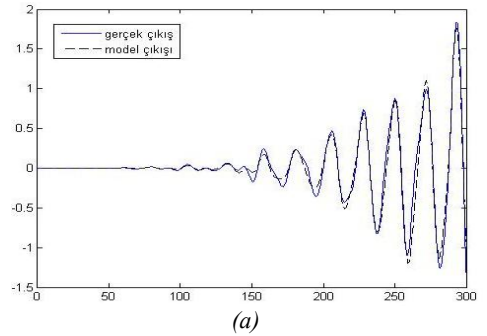
| Kontrol Parametreleri | YAK | GA | KSA |
|-----------------------------|------|-------|-------|
| Parametre araştırma aralığı | -1+1 | -1 +1 | -1 +1 |
| Popülasyon boyutu | - | 30 | 30 |
| Koloni sayısı | 30 | - | - |
| Klonlama oranı | - | - | 0.1 |
| Mutasyon oranı | - | 0.3 | - |
| Çaprazlama oranı | - | 0.8 | - |
| İterasyon | 500 | 500 | 500 |

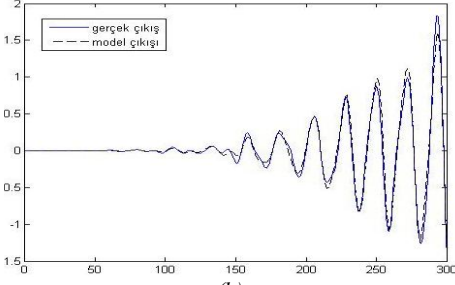
Kullanılan kontrol değerleri ile her bir algoritma 30 kez tekrarlatılmış ve algoritmaların bulunduğu parametreler Tablo 5'de verilmiştir.

Tablo 5. Denklem 7 için algoritmaların bulunduğu parametreler

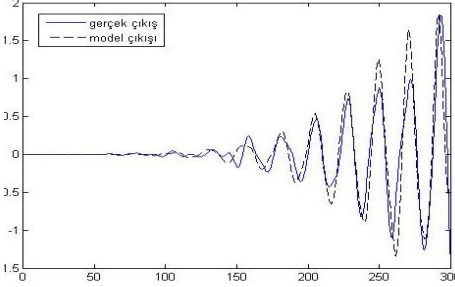
| Gerçek Değer | YAK | GA | KSA |
|----------------|------------------------|---------|------------------------|
| $a_{1,1}$ | 0.0107 | 0.1019 | -0.3750 |
| a_2 | 1.0000 | 0.9999 | -0.7500 |
| b_3 | -0.5823 | -0.5578 | 0.7500 |
| b_4 | -0.3258 | -0.3269 | -0.7500 |
| $c_{3,3}$ | -0.2574 | -0.4420 | 0.6563 |
| $c_{4,2}$ | 0.1128 | 0.2368 | -0.4028 |
| $c_{1,2}$ | 0.0986 | 0.0763 | -3.57×10^{-7} |
| Ort. süre (sn) | 27.49 | 59.92 | 128.87 |
| MSE | 0.0052 | 0.0057 | 0.0462 |
| Std. sap. | 9.878×10^{-4} | 0.0028 | 0.0132 |

Tablo 5'te hesaplanmış olan parametreler için sistemin gerçek çıkışları ile algoritmaların hesapladığı model ve test çıkışları aşağıdaki şekilde verilmiştir. (Şekil 3, Şekil 4)



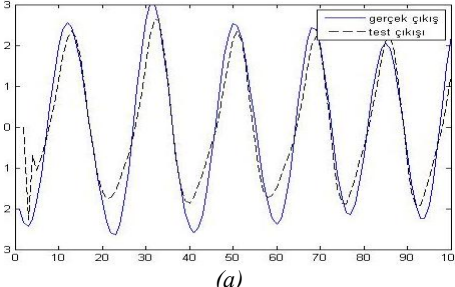


(b)

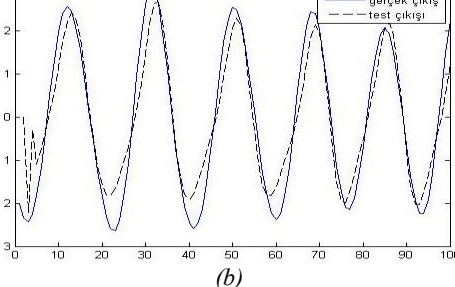


(c)

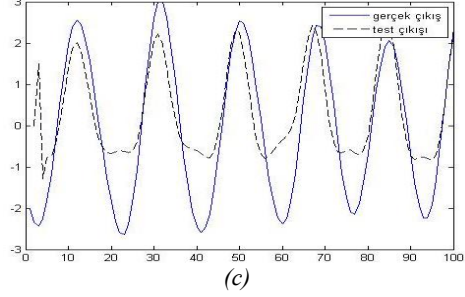
Şekil 3. Gerçek çıkış-Model Çıkışı: (a) YAK algoritması, (b) GA, (c) KSA



(a)



(b)



(c)

Şekil 4. Gerçek çıkış-Test Çıkışı: (a) YAK algoritması, (b) GA, (c) KSA

Ayrıca YAK algoritması için koloni sayısı, GA ve KSA'sı için popülasyon değerlerinin 20, 30, 50 ve 70 olması durumunda bu değerlerin algoritmaların parametre bulma sürelerine, en düşük MSE değerlerine ve MSE değerlerinin standart sapmasına etkisi incelenmiş olup, bu durum Tablo 6'da belirtilmiştir.

Tablo 6. Koloni ve popülasyon sayılarının ortalama süre ve hatalara etkisi

| Araştırılan değer | YAK | GA | KSA |
|--|------------------------|--------|--------|
| Koloni sayısı =20, Popülasyon sayısı = 20 | | | |
| Süre(sn) | 16.92 | 40.10 | 62.91 |
| MSE | 0.0061 | 0.0128 | 0.0125 |
| Std. sap. | 0.0011 | 0.0043 | 0.0194 |
| Koloni sayısı = 30, Popülasyon sayısı = 30 | | | |
| Süre(sn) | 27.49 | 59.92 | 128.87 |
| MSE | 0.0052 | 0.0057 | 0.0462 |
| Std. sap. | 9.878×10^{-4} | 0.0028 | 0.0132 |
| Koloni sayısı = 50, Popülasyon sayısı = 50 | | | |
| Süre(sn) | 111.13 | 98.39 | 277.11 |
| MSE | 0.0049 | 0.0040 | 0.0306 |
| Std. sap. | 3.818×10^{-4} | 0.0026 | 0.0084 |
| Koloni sayısı = 70, Popülasyon sayısı = 70 | | | |
| Süre(sn) | 156.311 | 136.46 | 381.56 |
| MSE | 0.0049 | 0.0017 | 0.0099 |
| Std. sap. | 1.778×10^{-4} | 0.0020 | 0.0073 |

Sonuçlar

Bu çalışmada kullanılan doğrusal parametrik sistemlerin parametre kestirim değerleri incelendiğinde, YAK algoritmasının GA'ya oldukça yakın bir sürede ve daha düşük bir hata ile çok başarılı bir şekilde parametre kestirimi yaptığı görülmüştür. Bu kestirimi yaparken de

GA'dan daha düşük bir hata ile yapmasında YAK algoritmasının doğrusal parametrik sistemlerde parametre kestiriminde başarılı bir algoritma olduğunun göstergesidir. Aynı zamanda hata yaklaşım eğrilerinden de anlaşılacağı üzere hatayı minimize edip optimum çözüme kararlı bir şekilde yaklaşan algoritma YAK algoritmasıdır. Farklı koloni ve popülasyon sayılarında ise hem GA'dan hemde KSA'dan daha düşük bir MSE ile parametre kestirimi yaptığı görülmüştür.

Doğrusal olmayan gerçek sistemlerde benzer durum söz konusu olup, popülasyon ve koloni sayılarının değişimine bağlı olarak hata oranları ve süre değişmekle birlikte YAK algoritması, GA'ya ve KSA'na göre her bir koşurmada bulunan hatalarda daha düşük bir standart sapma değeri ile parametre kestirimi yaptığından, doğrusal olmayan gerçek sistem parametre kestiriminde de YAK algoritmasının başarılı olduğu görülmüştür. Standart sapmanın düşük olması her koşurmada bulunan hatanın bir birine çok yakın olması anlamına gelir ki bu da algoritmanın parametre kestirimindeki kararlılığının göstergesidir.

Parametre bulma süresi açısından değerlendirildiğinde ise doğrusal olmayan gerçek sistemlerin modellenmesinde YAK algoritmasının bazı koloni sayılarında GA ve KSA'ya göre de daha erken sürede parametre kestirimi yaptığı görülmüştür. Kullanılan sistemler için farklı model yapıları uygulanıp parametre kestiriminde, YAK algoritmasının tüm koloni sayılarında daha erken sürede ve daha düşük MSE değerlerinde sonuç vermesi beklenebilir.

Kaynaklar

Akay, B. (2009) Nümerik Optimizasyon Problemlerinde Yapay Arı Kolonisi (Artificial Bee Colony) Algoritmasının Performans Analizi. Doktora Tezi, E.Ü., Fen Bilimleri Enstitüsü, Kayseri

Baran, A., (2001). Maksimum Olabilirlik Yaklaşımı ile Frekans Domeninde Lineer Sistemlerin Kimliklendirilmesi. Yüksek Lisans Tezi E.Ü., Fen Bilimleri Enstitüsü, Kayseri

De Moor B.L.R., DaISy: Database for the Identification of Systems, Department of Electrical Engineering, ESAT/SISTA, K.U.Leuven, Belgium, URL: <http://homes.esat.kuleuven.be/~smc/daisy/> (Son Erişim Tarihi: Nisan 2014)

Feron E., Brenner, M. Ve Paduano, Turevsky, J., (1998). Time-frequency analysis for transfer function estimation and application to fluetter clearance, in AIAA J. on Guidance, *Control&Dynamics*, Vol. 21, no.3, pp.375-382

Kaplan, A., (1995). Sistem Parametrelerinin Modern Spektral Analiz Yöntemleri Kullanılarak Belirlenmesi. Yüksek Lisans Tezi E.Ü., Fen Bilimleri Enstitüsü, Kayseri.

Karaboğa, D., (2005). An Idea on Honey Bee Swarm for Numerical Optimization, Technical Report-TR06.

Karaboğa, D. (2011). Yapay Arı Kolonisi Algoritması, Yapay Zeka Optimizasyon Algoritmaları, Nobel Yayın Dağıtım, İstanbul.

Karaboğa, D. ve Akay, B. (2010). A Modified Artificial Bee Colony Algorithm for Real-Parameter Optimization, Information Science, Available Online.

Karaboğa, D. ve Baştürk, B. (2008). On the Performance of Artificial Bee Colony (ABC) Algorithm", Applied Soft Computing, 687-697.

Karaboğa, D. ve Baştürk, B., (2007). A Powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony (ABC) Algorithm, *Journal of Global Optimization*, Vol.39, no.3, s.459-471.

Özer, Ş., Sağıroğlu, Ş. Ve Zorlu, H. (2003). Arma Sistem Modellemede Klasik Ve Yapay Sinir Ağları Algoritmalarının Karşılaştırılması, Elektrik – Elektronik -Bilgisayar Mühendisliği 10. Ulusal Kongresi.

Özer, Ş. ve Baran, İ., (2012). Doğrusal Olmayan Volterra Sistemlerin Yapay Arı Kolonisi Algoritması ile Modellenmesi, Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu.

Özer Ş. ve Baran, İ., (2012). Doğrusal Olmayan Bilineer ve Doğrusal Olmayan Gerçek Sistemlerin Yapay Arı Kolonisi Algoritması ile Modellenmesi, Otomatik Kontrol Türk Milli Komitesi Toplantısı.

Söderstrom, T., (1989). System Identification, 612, Prentice-Hall

Zorlu, H. ve Özer Ş., (2009). Doğrusal Olmayan Sistemlerin Klolanal Seçme Algoritması Kullanılarak Kimliklendirilmesi, IEEE 17. Sinyal İşleme ve İletişim Uygulamaları Kurultayı.

Modelling of linear parametric and non-linear real systems by using artificial bee colony algorithm

Extended abstract

A system can be define as in accordance with a specific purpose in response to inputs producing outputs that have a reciprocal interaction between one element to another and the relation between external world and within its elements. The main purpose of using the term of the system is to investigate the structure of the systems in order to attain desired outcomes, to determine the basic principles concerning with the system and to regulate the system based on purposes if possible.

However, it is not always possible to investigate the actual system and to determine its principles. Therefore, a tool is needed to represent all the features of the actual system. This tool will provide the best way to understand the system and its process . This representation tool, which will be used for this purpose, is called model. In other words, the model refers to simplified structures that can usually be either mathematical or computable.

The purpose of modelling is to determine the relation between input and output of an unknown system. In other words, modelling aims to find parameters of the transfer function that is characterized by the system. Modelling should be based on well defining for determination of complex parameters. By all means, if a model is established with chosen correct relationships, the solution would lead to more accurate and better results.

A trend towards the use of the natural simulations is increasing in order for modelling and solving complex optimization problem as day goes on because classical optimization algorithms are not sufficient to solve the problems that oversized, linear and non-linear mathematical or real system.

Modelling of real systems that suits a particular solution method is often not easy. Nature-inspired heuristic optimization algorithms, which are independent from the problem and model, are suggested to overcome deficiencies of conventional optimization techniques.

New ways of searching have brought along in cases where the use of classical optimization techniques is insufficient. Artificial intelligence optimization techniques have been proposed as a result of this search.

Swarm intelligence approach, which is a new branch of algorithms inspired by nature, is used instinctively problem solving skills of the insects. This approach has focused on the modeling of insects' behavior to develop effective meta-heuristics methods. An example of swarm intelligence approach is ABC algorithm developed by Karaboga. As a result of the interaction between insects, one of the most important parts of collective intelligence is to share information among insects individually. Oscillation dance of honey bees can be given as an example of types of interactive behaviors in which honey bees share information regarding the source and the quality of the discovered foods. By performing the dance, honey bees give messages to other bees in respect to quality of food supply, the direction of the food, the distance and the amount of the nectar. Through this successful mechanism, the colony can be directed to the region where good quality of food resources is available.

This study aims to determine the parameters estimation by using Artificial Bee Colony (ABC) Algorithm, Genetic Algorithm (GA) and Clonal Selection Algorithm (CSA). The obtained results have been compared with one algorithm to another. Furthermore, Artificial Bee Colony (ABC) Algorithm, which have been introduced into the literature newly, has been given as a good example of swarm intelligence approach.

In the literature, for comparison problems in the modelling studies, one linear parametric and one non-linear real system are often modelled by using these algorithms. The results of modelling have demonstrated that in the modelling of linear parametric system, ABC algorithm estimated parameter in a time very close to GA and indicated less modelling error in the number of different colonies than both GA and CSA. Same case also occurs in the modeling of non-linear parametric real system and depending on changes in the number of colonies, ABC algorithm modelled the system with less error in a shorter time compare to GA and CSA.

Keywords: Artificial Bee Colony Algorithm, linear system modelling, non-linear real system modelling