

# BBO Algoritmasının Optimizasyon Başarımının İncelenmesi

## Optimization Performance Investigation of BBO Algorithm

Tufan İNAÇ<sup>1</sup>, Cihan KARAKUZU<sup>2</sup>

<sup>1</sup>Bilgisayar Mühendisliği Anabilim Dalı

Bilecik Şeyh Edebali Üniversitesi, Fen Bilimleri Enstitüsü, Gülümbe Yerleşkesi, Bilecik  
tufan.inac@bilecik.edu.tr

<sup>2</sup>Bilgisayar Mühendisliği Bölümü

Bilecik Şeyh Edebali Üniversitesi, Mühendislik Fakültesi, Gülümbe Yerleşkesi, Bilecik  
cihan.karakuzu@bilecik.edu.tr

### Özet

*Bu çalışmada, doğrusal olmayan dinamik sistemlerin ANFIS (Adaptive Neuro-Fuzzy Inference System) bulanık çıkarım modeli kullanılarak modellenmesi problemi üzerinde Biyocoğrafya Tabanlı Optimizasyon (Biogeography Based Optimization) algoritmasının optimizasyon başarımı incelenmiştir. Kullanılan örnek sistemler için BBO algoritması ile elde edilen optimizasyon sonuçları ABC (Artificial Bee Colony), PSO (Particle Swarm Optimization) ve DE (Differential Evolution) algoritmalarıyla elde edilen sonuçlar ile karşılaştırılmıştır.*

### Abstract

*In this study, optimization performance of BBO (Biogeography Based Optimization) algorithm has been investigated on the problem of dynamic system modelling using ANFIS (Adaptive Neuro-Fuzzy Inference System) fuzzy inference system. The optimization results obtained for the used sample systems with BBO have been compared the results of ABC (Artificial Bee Colony), PSO (Particle Swarm Optimization) and DE (Differential Evolution) algorithms.*

### 1. Giriş

Günümüzde doğadaki biyolojik sistemlerden esinlenerek bulunmuş farklı Evrimsel Algoritmalar (EA) karmaşık problemlerin çözümünde sıklıkla kullanılmaktadır. Bu algoritmalarından en yaygın olarak kullanılanları Parçacık PSO (Particle Swarm Optimization), ABC (Artificial Bee Colony), ve DE (Differential Evolution) algoritmalarıdır. Doğadaki biyo çeşitlilikten ilham alınarak geliştirilmiş BBO (Biogeography Based Optimization) algoritması diğerlerine nazaran daha yeni bir sezgisel algoritmadır.

Bu çalışmada BBO algoritmasının bulanık sistem parametrelerini optimize etme başarımı, literatürden alınmış doğrusal olmayan örnek sistemler üzerinde kıyaslamalı olarak incelenmiştir. BBO algoritmasının örnek sistemler üzerindeki

sonuçları ABC, PSO ve DE algoritmalarından elde edilen sonuçlar ile kıyaslanmıştır.

### 2. Sezgisel Algoritmalar

Bu bölümde, çalışmada kullanılan BBO, ABC, PSO ve DE algoritmaları kısaca özetlenmiştir.

#### 2.1. Biyocoğrafya Tabanlı Optimizasyon (BBO)

BBO (Biogeography Based Optimization) algoritması, doğadaki türlerin dağılımından ilham alarak Simon tarafından 2008 yılında geliştirilen, biyolojik çeşitlilik ve popülasyon tabanlı yeni bir sezgisel optimizasyon yöntemidir [1].

Algoritmada her olası çözüm bir ada olarak kabul edilir. Her bir çözüm (ada) için, doğadaki türlerin dağılımı ve bu türlerin olası göç ve göçmenlik davranışları kullanılarak optimum sonucun bulunması amaçlanmaktadır. Algoritma temel olarak aşağıdaki işlem basamaklarından oluşur.

1. Arama uzayında rastgele başlangıç popülasyonu oluştur.
2. Başlangıç popülasyon bireylerinin uygunlukları belirlenir.
3. Popülasyondaki en iyi bireyin uygunluğu kaydedilir.
4. Göçmenlik ve göç oranlarına dayalı olarak popülasyondaki bireyleri belirle ve göç ettir.
5. Popülasyonu sırala ve en iyi bireyi ve uygunluğunu kaydet.

#### 2.2. Yapay Arı Kolonisi (ABC)

ABC (Artificial Bee Colony) algoritması, bal arılarının doğada yiyecek arama, nektar toplama ve bulunan kaynakları en verimli şekilde kovana getirme ilkelerinden esinlenerek Karaboğa tarafından geliştirilmiş sezgisel bir optimizasyon yöntemidir [2].

Algoritma çözülecek problemin parametrelerinin tanımlı olduğu uzayda en uygun değerlerini arıların nektar kaynağını

bulma davranışlarını işleterek belirlemeye çalışır. Aşağıda ABC algoritmasının temel adımları verilmiştir.

1. Başlangıç yiyecek kaynağı bölgeleri üretilir.
2. **Repeat**
3. İşçi arıların yiyecek kaynağı bölgelerine gönderilir.
4. Olasılıksal seleksiyonda kullanılacak olasılık değerlerinin görevli arılardan gelen bilgiye göre hesaplanır.
5. Gözcü arıların olasılık değerlerine göre yiyecek kaynağı bölgeleri seçilir.
6. Kaynağı bırakma kriteri: Limit ve Kaşif Arılar üretilir.
7. **Until** çevrim sayısı=maksimum çevrim sayısı.

### 2.3. Parçacık Sürü Optimizasyonu (PSO)

PSO (Particle Swarm Optimization) algoritması, kuş sürülerinin davranışından esinlenerek ortaya çıkarılmış sezgisel bir arama algoritmasıdır. PSO 1995 yılında Kennedy ve Eberhart tarafından geliştirilmiş popülasyon tabanlı sezgisel bir optimizasyon algoritmasıdır [3].

Algoritma, rastgele atanmış bireylerin (parçacık) ilgili arama uzayında kendilerinin ve sürüdeki tüm parçacıkların deneyimlerine dayalı olarak konumlarını her nesilde güncelleyerek optimum sonucu bulmayı amaçlamaktadır. Algoritma temel olarak aşağıdaki işlem basamaklarından oluşur.

1. Rastgele üretilen başlangıç parçacıkları ile başlangıç sürüsü oluşturulur.
2. Sürü içerisindeki tüm parçacıkların uygunluk değerleri hesaplanır.
3. Her bir parçacık için mevcut nesilde yerel en iyi parçacık (pbest) bulunur.
4. Mevcut nesile kadar bulunan yerel en iyiler içerisinde küresel en iyi parçacık (gbest) belirlenir.
5. Parçacık hızları ve parçacıklar güncellenir.
6. Durdurma kriteri sağlanıncaya kadar 2-3-4 ve 5. adımlar tekrar edilir.

### 2.4. Farksal Gelişim (DE)

DE (Differential Evolution) algoritması, Price ve Storn tarafından 1995 yılında geliştirilmiş, özellikle sürekli verilerin söz konusu olduğu problemlerde etkin sonuçlar veren popülasyon temelli sezgisel bir optimizasyon tekniğidir [4].

Temel olarak DE algoritması popülasyon içerisinde rastgele seçilen iki bireyin ağırlıklı farkının üçüncü bir bireye eklenmesi mantığına dayanmaktadır.

## 3. Örnek Dinamik Sistemler (ÖDS) ve Bulanık Mantık Yapısı

BBO algoritmasının bulanık sistem modellemedeki başarımını incelemek için literatürde sıkça kullanılan örnek dinamik sistemler kullanılmıştır. Dinamik sistemler ANFIS bulanık çıkarım sistemi kullanılarak modellenmiştir.

### 3.1. Örnek Dinamik Sistemler

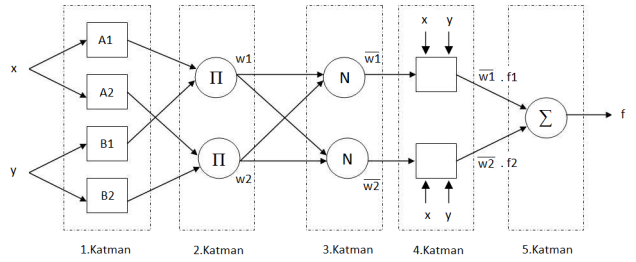
Literatürden seçilen örnek dinamik sistemlerin matematiksel ifadeleri (1) ve (2) numaralı denklemlerde sırasıyla verilmiştir [5], [6].

$$y(k) = \frac{y(k-1).y(k-2).(y(k-1)+2.5)}{1+y^2.(k-1)+y^2.(k-2)} + u(k) \quad (1)$$

$$y(k+1) = y(k) + u(k).e^{-3|y(k)|} \quad (2)$$

### 3.2. Bulanık Model

Yukarıda verilen örnek dinamik sistemler için kullanılan ANFIS bulanık çıkarım modeli Şekil 1’de verilmiştir [7].



Şekil 1: ANFIS bulanık çıkarım sistemi mimarisi

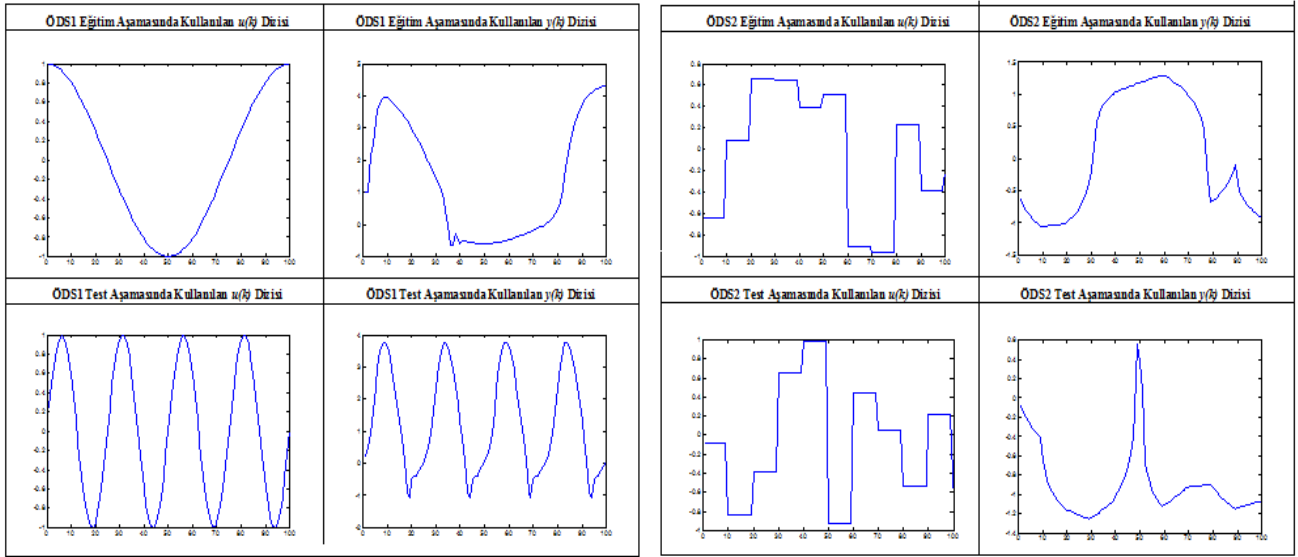
Şekil 1’de verilen ANFIS yapısına göre bulanık sistemin istenilen modellemeyi gerçekleştirebilmesi için birinci katmandaki üyelik fonksiyonlarının ve dördüncü katmandaki kuralların parametrelerinin optimizasyonu yapılmalıdır. Mevcut parametreler uygun değerlere getirildiğinde sistem istenilen (x,y)→z eşleşmesini gerçekleştirir. Buna uygun olarak ANFIS sisteminin girişleri belirlendikten sonra, her girişe ait ikişer adet “Gauss” üyelik fonksiyonu (ÜF) ve bu fonksiyonlar için belirlenen tüm kurallar tanımlanmıştır. ÖDS’lere ait girişler, giriş ÜF sayıları, kural sayıları ve optimize edilecek parametre sayıları Çizelge 1’de verilmiştir [7].

ÖDS	Girişler	Giriş ÜF Sayısı	Kural Sayısı	Parametre Sayısı
1	$u(k), y(k-2), y(k-1)$	2,2,2	8	36
2	$u(k), y(k)$	2,2	4	20

Çizelge 1: ÖDS’ler için giriş, ÜF, kural ve parametre sayıları

ÖDS’lerin modellenmesi amacıyla öncelikle eğitim veri setleri hazırlanmıştır. Tüm algoritmalar aynı şartlar altında bu veri setleri kullanılarak eşit sayıda koşturulmuş ve uygulanan ANFIS bulanık çıkarım modeliyle parametreleri belirlenmiştir. Kullanılan iki ÖDS için eğitim aşamasının ardından elde edilen modelin başarımının test edilmesi amacıyla eğitim setlerinden farklı test setleri kullanılmıştır. Hazırlanan test setleri tüm algoritmalar için aynı şartlarda ve aynı sayıda koşturulmuş ve sonuçlar kıyaslamalı olarak değerlendirilmiştir. İki ÖDS için eğitim ve test aşamalarında kullanılan grafikler Çizelge 2’de gösterilmiş, girişlere uygulanan  $u(k)$  dizileri Çizelge 3’de tanımlanmıştır.

Çizelge 3: Eğitim ve test setleri için kullanılan grafikler



Çizelge 2: Eğitim ve test setleri için uygulanan  $u(k)$  dizileri

ÖDS	Eğitim Seti İçin $u(k)$	Test Seti İçin $u(k)$
1	$\cos(2\pi k/100)$	$\sin(2\pi k/25)$
2	[-1 1] aralığında rastgele	[-1 1] aralığında rastgele

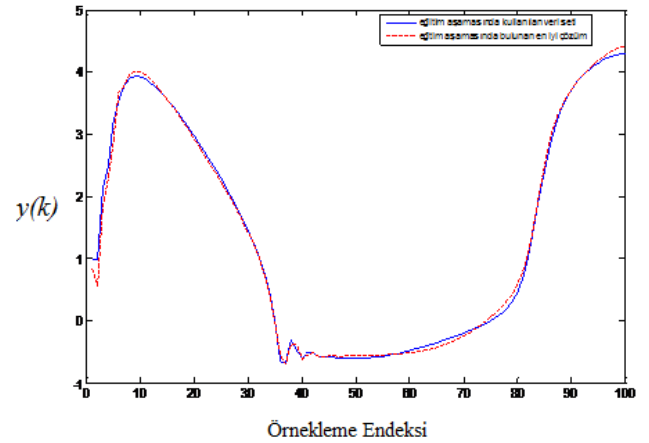
Yukarıda tanımlanan şekilde ÖDS'leri modelleme problemi için en uygun ANFIS parametrelerinin belirlenmesi amacıyla BBO, ABC, PSO ve DE algoritmaları, popülasyon büyüklüğü, maksimum nesil sayısı ve koşma sayıları aynı olmak üzere aynı şartlarda 50'şer kez koşturulmuştur. Yapılan çalışmada BBO algoritması temel alınmış ve bulanık sistem modellemedeki başarımı ABC, PSO ve DE algoritmaları ile kıyaslamalı olarak karşılaştırılmıştır.

### 3.3. BBO Algoritmasının Eğitim ve Test Seyri

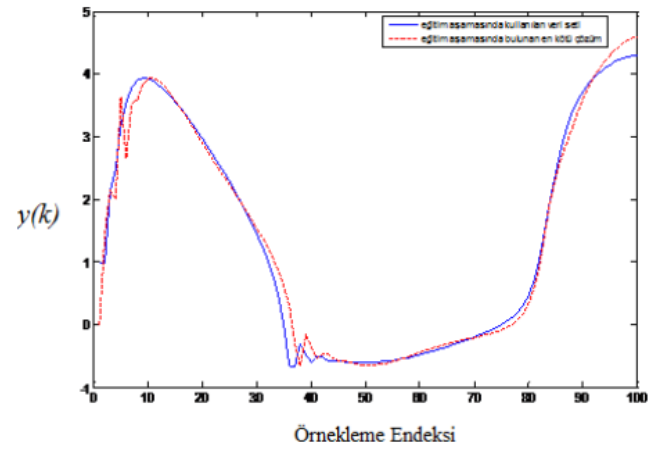
Tüm algoritmalar aynı şartlarda aynı değerler üzerinden eşit sayıda koşturulmuş ve elde edilen neticeler sonuçlar kısmında kıyaslamalı olarak karşılaştırılmıştır. Örnek olarak BBO algoritmasının eğitim ve test aşamalarındaki seyri aşağıda özetlenmiştir.

Eğitim ve Test aşamalarında ayrı veri setleri kullanılmış ve tüm algoritmalar her koşma 1000 tekrar olmak üzere 50'şer kez koşturulmuştur. ÖDS 1'in BBO algoritması için eğitim aşamasında elde edilen en iyi ve en kötü çözümlere ait grafikler Şekil 2 ve Şekil 3'de, test aşamasında elde edilen en iyi ve en kötü çözümlere ait grafikler Şekil 4 ve Şekil 5'de verilmiştir. ÖDS 1'in eğitim aşamasında en iyi koşma sonucunda elde ettiği eğitim seyri Şekil 6'da gösterilmiştir.

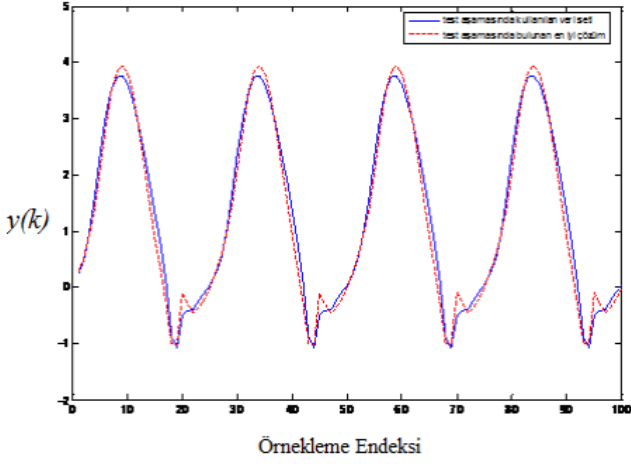
Grafiklerde düz çizgi ile eğitim veya test setleri için kullanılan  $y(k)$  dizileri, kesikli çizgi ile BBO algoritmasının eğitim ve test aşamalarında elde ettiği sonuçlar gösterilmiştir.



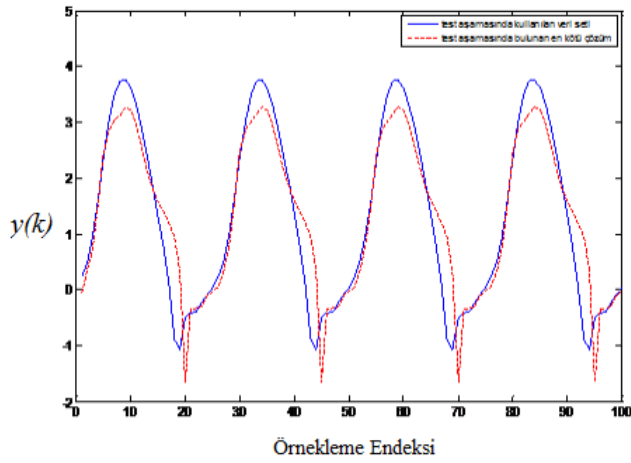
Şekil 2: Eğitim aşamasında elde edilen en iyi çözüm



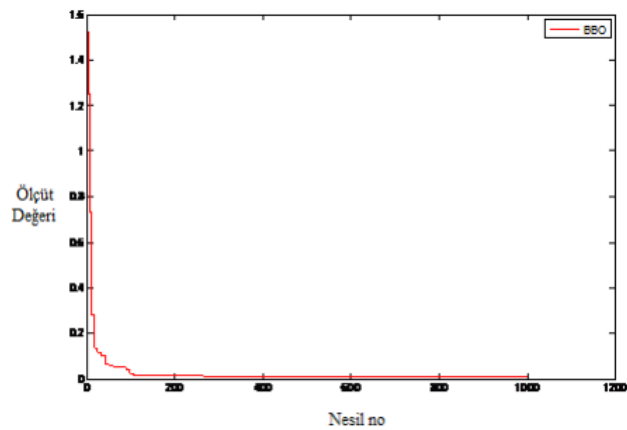
Şekil 3: Eğitim aşamasında elde edilen en kötü çözüm



Şekil 4: Test aşamasında elde edilen en iyi çözüm



Şekil 5: Test aşamasında elde edilen en kötü çözüm



Şekil 6: En iyi koşma için elde edilen eğitim seyri

#### 4. Sonuçlar

BBO algoritmasıyla ÖDS 1 ve ÖDS 2 için elde edilen sonuçlar, daha önce Yıldırım'ın yüksek lisans tez çalışmasında ABC, PSO ve DE algoritmaları için elde ettiği sonuçlarla Çizelge 4 ve Çizelge 5'de karşılaştırılmıştır [8].

Çizelge 4 ve Çizelge 5'de eğitim ve test aşamasında elde edilen sonuçlar ortalama başarımları baz alınarak değerlendirilmiştir. Tablolarda koyu olarak işaretlenen değerler o kategorideki en iyi sonucu göstermektedir. Bir algoritma için ortalama başarımları sayısı, o algoritmaya ait satırlardaki koyu olarak işaretlenen değerlerin toplamının, 2 ÖDS için Çizelge 4'de toplam 10 kategori sayısına, Çizelge 5'de toplam 4 kategori sayısına bölünmesiyle elde edilmiştir.

Kullanılan kategorilerin tanımlaması aşağıda verilmiştir.

- Ortalama; 50 ayrı koşma sonucunda bulunan eğitim ölçüt değerlerinin ortalaması
- Standart Sapma; 50 ayrı koşma sonucunda bulunan eğitim ölçüt değerlerinin standart sapması
- En İyi; 50 ayrı koşma sonucunda elde edilen en iyi ölçüt değeri
- En Kötü; 50 ayrı koşma sonucunda elde edilen en kötü ölçüt değeri
- 1 Nesil Süresi; 50 ayrı koşma sonucunda bulunan bir nesil için gerekli ortalama süre

Çizelge 4'de verilen eğitim aşamasındaki sonuçlara göre BBO algoritması ortalama başarımları sayısı baz alınarak kıyaslandığında DE algoritmasının ardından 0.3 ortalama başarımları sayısı ile en iyi sonucu vermiştir. BBO algoritmasının 1 nesil süresi dışındaki kategorilerde de en iyi sonuçları alan ABC ve DE algoritmalarına çok yakın sonuçlar verdiği görülmüştür. 1 nesil süresi kategorisinde en kötü sonucu BBO algoritması göstermiştir. Bu durum bize BBO algoritmasının diğer algoritmalara göre daha yavaş çalıştığı sonucunu vermiştir. Ancak BBO algoritmasının diğer rakiplerine göre daha basit bir yapıya sahip olmasının mevcut yapısından çok daha hızlı sonuçlar verebilecek bir yapıya dönüştürülebileceği değerlendirilmektedir.

Çizelge 4: ÖDS’lerin eğitim aşamasındaki başarımların kıyaslaması

Algoritma	Ölçüt	ÖDSP 1 Max.Nesil (1000)	ÖDSP 2 Max.Nesil (1000)
BBO	Ortalama	<b>0.025769</b>	0.014692
	Standart Sapma	0.011459	0.008455
	En İyi	<b>0.007467</b>	0.007876
	En Kötü	<b>0.054688</b>	0.038530
	1 Nesil Süresi (s)	0.092862	0.074790
	Başarım Sayısı	3	0
	Ortalama Başarım Sayısı	3/10=0.3	
ABC	Ortalama	0.036994	0.012165
	Standart Sapma	0.012073	0.013409
	En İyi	0.013664	0.004422
	En Kötü	0.058886	0.056209
	1 Nesil Süresi (s)	<b>0.026451</b>	<b>0.015753</b>
	Başarım Sayısı	1	1
	Ortalama Başarım Sayısı	2/10=0.2	
PSO	Ortalama	0.079258	0.212148
	Standart Sapma	0.063467	0.568219
	En İyi	0.027028	0.005427
	En Kötü	0.437995	3.759978
	1 Nesil Süresi (s)	0.047024	0.028630
	Başarım Sayısı	0	0
	Ortalama Başarım Sayısı	0	
DE	Ortalama	0.039115	<b>0.004050</b>
	Standart Sapma	<b>0.004217</b>	<b>0.001221</b>
	En İyi	0.337750	<b>0.002383</b>
	En Kötü	0.054579	<b>0.008280</b>
	1 Nesil Süresi (s)	0.040384	0.034371
	Başarım Sayısı	1	4
	Ortalama Başarım Sayısı	5/10=0.5	

Bu çalışmada BBO algoritmasının bulanık ağ yapısı üzerindeki başarımının incelenmesi amacıyla, farklı veri setleri (test seti) kullanılarak her bir ÖDS için elde edilen ANFIS modelleri test edilmiştir. Aynı veri setleri ile BBO, ABC, PSO ve DE algoritmaları için elde edilen test aşamasındaki sonuçlar Çizelge 5’de verilmiştir. Elde edilen sonuçlara göre BBO algoritmasının 0.75 ortalama başarımların sayısı ile en başarılı algoritma olduğu görülmüştür. DE algoritması 0.25 ortalama başarımların sayısı ile BBO algoritmasının ardından ikinci olmuştur. Yapılan karşılaştırma ve kıyaslama neticesinde ABC, PSO ve DE algoritmalarının ortalama başarımların sayısı baz alındığında BBO algoritmasını yakalayamadıkları sonucuna varılmıştır. Bu sonuca dayalı olarak BBO algoritmasının genelleme yapabilme yeteneğinin rakiplerine nazaran kuvvetli olduğu söylenebilir.

Çizelge 5: ÖDS’lerin test aşamasındaki başarımların kıyaslaması

Algoritma	Ölçüt	ÖDSP 1 Max.Nesil (1000)	ÖDSP 2 Max.Nesil (1000)
BBO	En İyi	<b>0.009313</b>	<b>0.009348</b>
	En Kötü	2.672800	<b>0.113219</b>
	Başarım Sayısı	1	2
	Ortalama Başarım Sayısı	3/4=0.75	
ABC	En İyi	0.027788	0.025316
	En Kötü	1.972510	547.1167
	Başarım Sayısı	0	0
	Ortalama Başarım Sayısı	0	
PSO	En İyi	0.054068	0.023815
	En Kötü	1.278393	51.89824
	Başarım Sayısı	0	0
	Ortalama Başarım Sayısı	0	
DE	En İyi	0.047424	0.031532
	En Kötü	<b>0.195956</b>	2.356835
	Başarım Sayısı	1	0
	Ortalama Başarım Sayısı	1/4=0.25	

## 5. Kaynaklar

- [1] Simon, D., “Biogeography-Based Optimization”, *IEEE Transactions on Evolutionary Computation*, 12(6):702-713, 2008.
- [2] Karaboğa, D., “An idea based on honey bee swarm for numerical optimization”, *Technical Report TR06*, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
- [3] Kennedy, J. ve Eberhart, R.C., “Particle Swarm Optimization”, *Proc. IEEE Int. Conf. on Neural Networks*, (4):1942-1948, 1995.
- [4] Price, K.V., “Differential Evolution: a Fast and Simple Numerical Optimizer”, *In: Smith, M., Lee, M., Keller, J., Yen., J.(EDS.): Biennial Conference of the North American Fuzzy Information Processing Society*, NAFIPS IEEE Pres, New York, :524-527, 1996.
- [5] Narendra, K.S., ve Parthasarathy, K., “Identification and Control of Dynamical Systems Using Neural Networks”, *IEEE Trans. On Neural Networks*, 1(1):4-27, 1990.
- [6] Babuska R., “Fuzzy System Modeling and Identification”, <http://www.dsc.tudelft.nl/~babuska/transp/fuzzmod.pdf> (son erişim 01.06.2013).
- [7] Jang, S., ve R., J., “ANFIS: adaptive –networks- based fuzzy inference system”, *IEEE Trans. on Systems Man and Cybernetics*, 23(3):665-684, 1993.
- [8] Yıldırım, Ö., “Sezgisel Arama Algoritma Tabanlı Bulanık Sistem Optimizasyonu”, *Bilecik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi*, 2012.