


## Sürekli Optimizasyon Problemleri için Çoklu Arama Stratejisi Kullanan Fil Sürü Optimizasyonu

Hüseyin Haklı

Bilgisayar Mühendisliği Bölümü, Mühendislik-Mimarlık Fakültesi, Necmettin Erbakan Üniversitesi, Konya  
hhakli@erbakan.edu.tr, 

Araştırma Makalesi

Geliş Tarihi: 29.08.2018

Kabul Tarihi: 15.03.2019

### Öz

Fillerin sosyal davranışlarını taklit eden fil sürü optimizasyonu (EHO), yakın zamanda önerilen sürü zekası ve popülasyon tabanlı bir optimizasyon algoritmasıdır. EHO, yerel arama konusunda iyi bir yeteneğe sahip olmasına rağmen popülasyon çeşitliliğini erken kaybetmesi nedeniyle global aramada etkili olamamaktadır. Temel EHO yönteminde, yeni bireylerin oluşturulması için tek bir çözüm arama denklemi kullanılmaktadır. Bu nedenle, arama uzayının etkili bir şekilde araştırılmasında ve farklı karakteristikteki problemlerin çözümünde yetersiz kalmaktadır. Bu çalışmada, bu sorunların üstesinden gelmek ve keşif ve faydalanma arasındaki dengeyi sağlayabilmek için en çok bilinen optimizasyon tekniklerinin arama stratejilerinden esinlenilerek çoklu arama stratejisi kullanan fil sürü optimizasyonu (Multi-EHO) önerilmiştir. Önerilen yöntem ile temel EHO'nun karşılaştırılması için farklı karakteristikteki 15 fonksiyona sahip CEC2015 test seti kullanılmıştır. Ayrıca Multi-EHO'nun performansını doğrulamak için, önerilen yöntem son yıllarda önerilen gri kurt algoritması (GWO) ve balina optimizasyonu algoritması (WOA) ile karşılaştırılmıştır. Deneysel sonuçlar, önerilen yöntemin diğer yöntemlere göre daha başarılı ve daha sağlam bir performansa sahip olduğunu göstermektedir.

**Anahtar Kelimeler:** Fil sürü optimizasyonu, sürekli optimizasyon, arama stratejileri, çoklu arama stratejisi

## Elephant Herding Optimization Using Multi-Search Strategy for Continuous Optimization Problems

Huseyin Hakli

Department of Computer Engineering, Faculty of Engineering-Architecture, Necmettin Erbakan University, Konya  
hhakli@erbakan.edu.tr

### Abstract

The elephant herding optimization (EHO), which imitates social behaviors of the elephants, is recently proposed a swarm intelligence and population-based optimization algorithm. Although EHO is a good at local search, it is not effective on the global search due to the rapid loss of population diversity. In the basic EHO method, a single solution search equation is used for the generating the new individuals. Therefore, it is insufficient on the solving the problems which have different characteristics and the exploring the search space effectively. In this study, in order to overcome these problems and to provide a balance between exploration and exploitation, elephant herding optimization using multi-search strategy (Multi-EHO) has been proposed which inspired by the search strategies of the most well-known optimization techniques. For the comparison of the proposed method and the basic EHO, the CEC2015 benchmark set with 15 different functions is used. In addition, to validate the performance of Multi-EHO, the proposed method is compared with the grey wolf optimizer (GWO) and the whale optimization algorithm (WOA) proposed in recent years. Experimental results show that the proposed method has more successful and more robust performance than other methods.

**Keywords:** Elephant herding optimization, continuous optimization, search strategies, multi-search strategy

### 1. GİRİŞ

Son yıllarda, karmaşık ve zor yapıdaki lineer olmayan

problemler, bir optimizasyon problemi şeklinde formüle edilerek doğa-esinli algoritmalar ile kabul edilebilir bir süre içinde çözülmektedir [1, 2]. Doğa esinli algoritmaların en

\*<sup>1</sup>Sorumlu yazar : Hüseyin Haklı, Bilgisayar Mühendisliği Bölümü, Mühendislik-Mimarlık Fakültesi, Necmettin Erbakan Üniversitesi, Konya - hhakli@erbakan.edu.tr

çok bilinenleri yapay arı kolonisi (ABC) [3], parçacık sürü optimizasyonu (PSO) [4] ve karınca kolonisi algoritmasının (ACO) [5] yanında, doğadaki canlıların davranış ve yaşayış biçimlerinden ilham alınarak birçok yeni algoritma önerilmeye devam edilmektedir. Son zamanlarda ortaya çıkan algoritmalar ise doğadaki gri kurtların liderlik hiyerarşisini ve avlanma mekanizmasını taklit eden gri kurt optimizasyonu (GWO) [6], kambur balinaların sosyal davranışlarından esinlenen balina optimizasyonu algoritması (WOA) [7] ve fil gruplarının sürü ve toplanma davranışlarını benzeten fil sürü optimizasyonu (EHO) [8] örnek olarak gösterilebilir.

Wang ve ark. tarafından 2015 yılında önerilen EHO hakkında literatür sınırlı olmasına rağmen, EHO birçok farklı problemin çözümünde kullanılmıştır. Strumberger ve ark. statik drone konum probleminin çözümünde EHO algoritmasından yararlanmışlardır [9]. Simülasyon sonuçları, dikkate alınan drone sayısı düşünüldüğünde EHO'nun hedefleri koruma konusunda başarılı sonuçlar elde ettiğini göstermiştir. EHO, çoklu imge eşikleme problemine ise Tuba ve ark. tarafından uygulanmıştır [10]. EHO diğer dört farklı sürü zekası yöntemi ile standart test resimleri üzerinde karşılaştırılmış ve elde edilen sonuçlara göre EHO diğer yaklaşımlara göre daha başarılı bulunmuştur. Çok güçlü bir sınıflandırma tekniği olan destek vektör makinelerinin (SVM) doğruluğu, öğrenme parametrelerine yüksek bağımlılık göstermesi nedeniyle, EHO algoritması ile SVM parametre ayarlaması gerçekleştirilmiştir [11]. Ayrıca EHO, insansız hava aracı yol planlaması [12], çizelgeleme [13], yük frekansı kontrolü [14] gibi problemlerin çözümünde kullanılmıştır. EHO'nun uygulanmasına yönelik çalışmaların yanı sıra performans analizi ve iyileştirilmesi üzerine de çalışmalar bulunmaktadır. Temel EHO algoritması, Tuba ve ark. tarafından CEC2013 test fonksiyonları üzerinde analiz edilmiş ve iyi bir karakteristiğe sahip olduğu görülmüştür [15].

Parashar ve ark., en iyi fil pozisyonunun güncellenmesi esnasında, temel EHO'dan farklı bir şekilde sadece merkez klan pozisyonunu kullanmak yerine, bir önceki en iyi fil pozisyonunu klanın merkez pozisyonuna ekleyerek değiştirilmiş fil sürü optimizasyonunu önermiştir [16]. Bir diğer çalışma da ise, Parashar ve ark. çalışmasına benzer olarak mevcut en iyi fil pozisyonunun merkez klan pozisyonuna eklenmesi ile en iyi fil pozisyonunun güncellenmesinin yanında, yeni oluşturulan bebek fil pozisyonlarının arama uzayında rasgele belirlenmesi yerine o klandaki yerel en iyi pozisyona göre oluşturulması olmak üzere EHO algoritmasına iki ayrı iyileştirme eklenmiştir [17]. Tuba ve ark. 2018 yılında, EHO algoritmasına iki farklı kaotik harita uygulayarak, rasgele çözümler ile başlangıç popülasyonu oluşturmak yerine kaotik haritalar ile tekrarlanmayan sayılar üretmek daha gelişmiş aramalar yapmasını sağlamışlardır [18].

Popülasyon tabanlı bir sürü-zekası tekniği olan EHO, fillerin sosyal yaşam davranışlarından etkilenerek popülasyonu ayrı klanlara bölmektedir. Her klan için aynı arama denklemi

kullanılmakta ve bu denkleme mevcut filin pozisyonu dışında o klana ait en iyi fil pozisyonlarını dikkate almaktadır. EHO algoritmasında, her klana ait en iyi çözüm ise o klana ait merkez pozisyon ile güncellenmektedir. Bu işlemler sayesinde EHO güçlü bir yerel aramaya sahip olmaktadır. Ancak yeni fillerin belirlenmesinde en iyi çözümden etkilenen tek bir arama denklemi global arama için yetersiz kalırken, en iyi fil pozisyonlarının merkez fil pozisyonuna yaklaştırılması popülasyon çeşitliğini azaltmaktadır. Bu çalışmada, bu sorunların üstesinden gelmek için çoklu arama stratejisi kullanan fil sürü optimizasyonu (Multi-EHO) önerilmiştir.

Önerilen yöntemde, ABC, PSO ve diferansiyel evrim algoritması (DE) [19] gibi bilinen optimizasyon algoritmalarının arama stratejilerinden faydalanılarak her klan için farklı bir arama denklemi belirlenmiştir. Ayrıca en iyi fil pozisyonunu klana ait merkez pozisyondan güncelleme işlemi ise göz ardı edilmiştir. Önerilen yöntemin performansı farklı karakteristikteki problemlere sahip CEC2015 test fonksiyonları üzerinde incelenmiş ve Multi-EHO, temel EHO'nun yanı sıra yeni ortaya çıkan GOA ve WOA yöntemleri ile de karşılaştırılmıştır.

Çalışmanın aşağıdaki bölümleri şöyledir: Bölüm 2, temel EHO algoritmasını kapsamaktadır. Önerilen yöntem Bölüm 3'te açıklanmıştır. Deneysel sonuçlar ve karşılaştırmalar Bölüm 4'te incelenmiştir ve sonuç Bölüm 5'te sunulmuştur.

## 2. TEMEL FİL SÜRÜ OPTİMİZASYON YÖNTEMİ

Fil gruplarının bazı temel sürü davranışları, optimizasyon problemlerinin çözümünde kullanılabilmesi için basitleştirilerek dikkate alınmıştır [20]:

- Bir fil popülasyonu klanların birleşimden oluşur ve her klan sabit sayıda file sahiptir.
- Sabit sayıda erkek filler, aile gruplarını terk edecek ve her neslin başlangıcında ana fil grubundan uzak bir şekilde yalnız yaşayacaklardır.
- Her klandaki filler bir maderşahinin liderliğinde yaşamaktadır.

Belirtilen davranışlar bir optimizasyon algoritması için benzetildiğinde, her bir fil bir aday çözümü temsil etmektedir. Popülasyon kendi içinde alt gruplara bölünerek klanları oluşturur. Ayrıca her klanda en iyi uygunluk değerine sahip fil maderşahi olarak adlandırılır. Her iterasyon sonunda o klanda bulunan en kötü fil ise sürüden ayrılarak, arama uzayında kendine yeni bir pozisyon belirlemektedir.

EHO algoritmasında klan güncelleme ve ayrılma olmak üzere iki temel işlem bulunmaktadır. Klan güncelleme işlemi, her fil için kendi pozisyonu ve o klana ait maderşahinin pozisyonunu kullanılarak gerçekleştirilmekte ve Denklem (1)'de gösterilmektedir [8, 20].

$$X_{\text{yeni},ci}^j = X_{ci}^j + \alpha \times (X_{\text{eni},ci}^j - X_{ci}^j) \times r \quad (1)$$

Denklem (1) 'de  $X_{ci}^j$  ci. klanına ait  $j$ . filin pozisyonunu temsil etmekteyken,  $X_{eniye.ci}^j$  ci. klandaki maderşahinin pozisyonunu göstermektedir.  $X_{yeni.ci}^j$  ise mevcut filin yeni pozisyonunu ifade etmektedir. Maderşahinin, güncellenen yeni pozisyonun üzerindeki etkisini belirleyen  $\alpha$  ölçek faktörü iken,  $r$  ise normal dağılıma göre  $[0,1]$  aralığında üretilen rasgele bir sayıyı temsil etmektedir. Denklem (1) ile o klandaki maderşahinin pozisyonunu güncellendiğinde, herhangi bir değişiklik olmayacaktır. Bu nedenle her klandaki maderşahinin pozisyon güncellenmesi Denklem (2) ile yapılmaktadır.

$$X_{yeni.ci}^j = \beta \times X_{ci}^{merkez} \quad (2)$$

Denklem (2)'de  $\beta$ , güncellenen yeni pozisyon üzerindeki  $X_{ci}^{merkez}$ 'nin etkisini belirleyen bir faktör iken,  $X_{ci}^{merkez}$  ise ci. klanına ait fillerin ortalama pozisyonlarını göstermektedir.  $X_{ci}^{merkez}$ 'nin hesaplama işlemi Denklem (3)'de verilmektedir.

$$X_{ci}^{merkez,d} = \frac{1}{n_{ci}} \times \sum_{j=1}^{n_{ci}} X_{ci}^{j,d} \quad (3)$$

Denklem (3)'de  $n_{ci}$ , ci. klandaki fil sayısını,  $X_{ci}^{j,d}$  ise ci. klanına ait  $j$ . filin  $d$ . boyutunu temsil etmektedir. Denklem (3) ile her boyut için o klandaki merkez pozisyon hesaplanarak  $X_{ci}^{merkez}$  bulunmaktadır. Klan güncelleme tamamlandıktan sonra, ayrılma işleminde klandaki en kötü uygunluk değerine sahip fil Denklem (4) ile arama uzayında yeni bir pozisyona atanmaktadır.

$$X_{enkötü.ci} = X_{min} + (X_{mak} - X_{min} + 1) \times rand \quad (4)$$

$X_{enkötü.ci}$  ci. klandaki en kötü uygunluk değerine sahip fili temsil ederken  $X_{min}$  ve  $X_{mak}$  arama uzayındaki alt ve üst sınır değerleri ifade etmektedir. Ayrılma işleminin tamamlanması ile klanlar birleştirilerek elde edilen fil popülasyonu uygunluk değerine göre sıralanmaktadır. Elitizm işlemi ile mevcut popülasyondan belirlenen sayıda elit filler, yeni oluşturulan popülasyondaki en kötü uygunluk değerine sahip filler yerine kopyalanır. Klan güncelleme ve elitizm işlemleri arasındaki adımlar durdurma kriteri sağlanana kadar tekrar edilir, durdurma kriteri sağlandığında ise algoritma sonlandırılır.

### 3. ÇOKLU ARAMA STRATEJİSİ KULLANAN FİL SÜRÜ OPTİMİZASYON YÖNTEMİ

Bir optimizasyon tekniğinin performansının belirlenmesinde, çözüm arama stratejisi önemli bir rol oynamaktadır [21]. Arama stratejisinin önceliğine göre bazı yöntemler global arama bazı yöntemler ise yerel arama konusunda ön plana çıkmaktadır. Bu nedenle keşif ve

aydalanma arasındaki dengeyi sağlayacak bir arama stratejisi belirlemek önemlidir [22]. Ayrıca, sürecin tüm aşamalarında tek bir arama stratejisi kullanmak yerine farklı arama stratejilerini bir arada kullanmanın daha başarılı olabileceği öngörülmüş ve farklı yöntemler için çoklu arama stratejileri uygulanmıştır [21-24]. Temel EHO yönteminde, her klan için tek bir arama denklemi kullanılmakta ve bu arama stratejisi de yerel arama konusunda başarılıyken, global arama konusunda zayıf kalmaktadır. Bu çalışmada, temel EHO yönteminin global arama başarısını arttırmak ve yerel aramadaki yeteneğini sürdürebilmek için temel EHO'nun kendi arama stratejisinin yanı sıra ABC, PSO ve DE yöntemlerinin arama stratejilerinden esinlenerek çoklu arama stratejisi kullanılmıştır. Denklem (1) dışında kullanılan diğer arama stratejileri Tablo 1'de verilmektedir.

**Tablo 1.** Kullanılan arama stratejileri ve esinlenen yöntemler

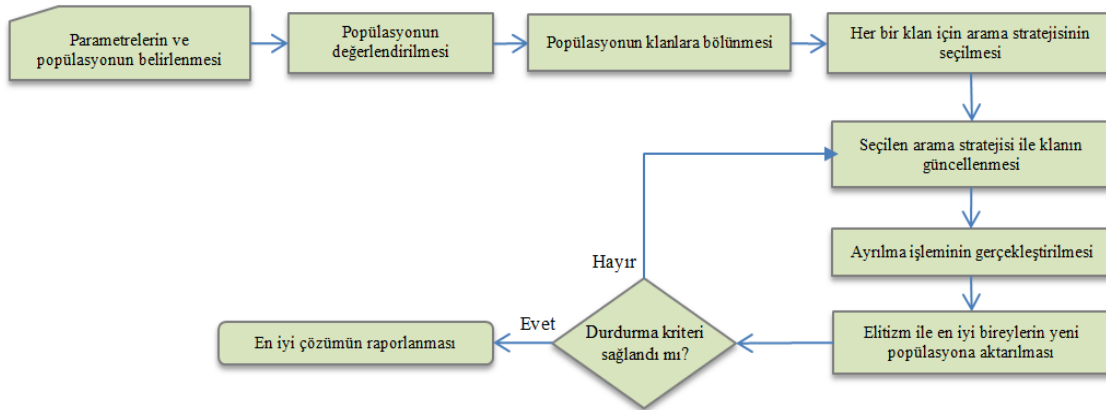
Esinlenen Yöntem	Denklem
ABC	$X_{yeni.ci}^{j,d} = X_{ci}^{j,d} + \Phi \times (X_{ci}^{j,d} - X_{ci}^{k,d})$ (5)
PSO	$X_{yeni.ci}^j = X_{ci}^j + c_1 \text{rand} \times (X_{eniye.ci}^j - X_{ci}^j) \dots + c_2 \text{rand} \times (X_{eniye}^j - X_{ci}^j)$ (6)
DE/Best/1	$X_{yeni.ci}^{j,d} = X_{eniye.ci}^{j,d} + \Phi \times (X_{ci}^{k,d} - X_{ci}^{r,d})$ (7)
DE/Rand/1	$X_{yeni.ci}^{j,d} = X_{ci}^{k,d} + \Phi \times (X_{ci}^{j,d} - X_{ci}^{r,d})$ (8)

Denklem (5), (7) ve (8)'de  $X_{yeni.ci}^{j,d}$ , ci. klandaki  $j$ . filin  $d$ . boyutunun yeni değerini temsil ederken,  $\Phi$  ise  $[-1,1]$  aralığında üretilen rasgele bir sayıyı simgelemektedir. Bu üç denklemde de tüm boyutları güncellenmek yerine sadece rasgele seçilen bir boyut değeri güncellenmektedir. Ayrıca bu denklemlerde verilen  $j$ ,  $k$  ve  $r$  indisleri farklı filleri göstermekte ve bu nedenle birbirine eşit olmamaktadır. Denklem (6)'da  $X_{eniye.ci}^j$  ci. klanına ait en iyi fil pozisyonunu (maderşahi),  $X_{eniye}^j$  ise tüm popülasyondaki en iyi bireyi göstermektedir.  $c_1$  ve  $c_2$  hızlandırma katsayıları olarak adlandırılmaktadır. Denklem (7)'deki  $X_{eniye.ci}^d$  ise ci. klanına ait maderşahinin  $d$ . boyutundaki değeri temsil etmektedir.

ABC yönteminin arama stratejisi keşif konusunda daha başarılı iken, PSO ise yerel aramayı çok daha etkin bir şekilde yapabilmektedir. DE/best/1, doğrudan klandaki en iyi pozisyonun etrafında arama yapması nedeniyle faydalanma yeteneğini güçlendirirken, DE/rand/1 ise mevcut bireyin yanında rasgele bireyler ile güncelleme yaparak global aramayı destekler. Temel EHO'nun arama stratejisi de faydalanma konusunda daha iyiyken, popülasyon çeşitliliğini erken kaybetmektedir. Kullanılan farklı arama stratejileri ile global ve yerel arama arasında başarılı bir

uzlaşma sağlanması hedeflenmiştir. Çoklu arama stratejisinde karar verilmesi gereken iki temel konu bulunmaktadır. Arama stratejilerinin eş zamanlı mı yoksa ayrı ayrı mı kullanılacağına ve arama stratejilerinin seçiminin nasıl sağlanacağına karar verilmelidir. Bu çalışmada arama stratejileri ayrı ayrı kullanılmıştır. Bunun en önemli nedeni eğer bir bireyin pozisyon güncellenmesi esnasında beş farklı denklem eş zamanlı olarak kullanılırsa, fonksiyon değerlendirme sayısı da beş kere kullanılmış olacaktır. Ayrıca hesaplama süresi de buna bağlı olarak artacaktır. Diğer durum için ise temel EHO'nun klanlara

ayrılan yapısının avantajı kullanılmış, her klanla bir denklem atanarak seçilme işlemi yöntem başında gerçekleştirilmiştir. Her bir klan için belirlenen arama stratejisi iterasyon sırasında değişmemekte ve algoritma sonlanana kadar aynı şekilde kalmaktadır. Önerilen yöntemde, temel EHO'nun maderşahinin güncellenmesi için kullandığı Denklem (2) ve Denklem (3), klandaki merkez pozisyonun tutarsızlığı ve dezavantajı üzerindeki tartışmalar [16, 17, 25] nedeniyle tercih edilmemiştir. Şekil 1'de önerilen yöntemin akış şeması verilmektedir.



Şekil 1. Önerilen yöntemin akış şeması

#### 4. DENEYSSEL SONUÇLAR

Önerilen yöntemin performansını ve doğruluğunu incelemek için, önerilen yöntem temel EHO ve diğer yöntemler GWO ve WOA ile karşılaştırıldı. Deneyler CEC2015 [26] test fonksiyonları üzerinde gerçekleştirildi ve bu fonksiyonların karakteristikleri ve özellikleri Bölüm 4.1'de verildi. Bölüm 4.2' de algoritmaların kontrol parametreleri verilirken, diğer bölümlerde ise deneysel sonuçlar ve yakınsama grafikleri sunuldu.

##### 4.1. CEC2015 Test Fonksiyonları

Bu çalışmada yöntemlerin performanslarını değerlendirmek için CEC2015 setinde bulunan 15 farklı karakteristikteki test fonksiyonları kullanılmıştır. Bu fonksiyonlar, Liang ve ark. [26] tarafından tek amaçlı optimizasyon algoritmalarını tutarlı bir şekilde karşılaştırmak için önerilmiştir. Tüm fonksiyonlar için arama uzayı  $[-100,100]$  aralığında sınırlandırılmıştır. Test setinde bulunan fonksiyonlardan, F1 ve F2 tek modlu (unimodal), F3-F5 basit çok modlu (multimodal), F6-F8 hibrit ve F9-F15 aralığı ise bileşik (composition) karakteristiktedir.

Tüm test fonksiyonları minimizasyon problemidir ve tüm fonksiyonların optimumu [26] referansında verildiği gibi kaydırılır. Deneylerde ve karşılaştırmalarda, boyut sayısı 30 belirlenirken, durdurma kriteri olarak ise maksimum fonksiyon değerlendirme sayısı kullanılmıştır.

##### 4.2. Parametre Ayarları

Deneylerde kullanılan yöntemlerin parametre ayarları, orijinal çalışmalarına uygun şekilde belirlenmiştir. Temel EHO yönteminde  $\alpha$  ve  $\beta$  sırasıyla 0.5 ve 0.1 olarak ayarlanmış, klan sayısı ise 5 olarak belirlenmiştir [20]. Elitizm ile bir sonraki popülasyona aktarılacak elit birey sayısı ise 2 olarak seçilmiştir. Önerilen yöntemde de belirtilen parametreler aynı şekilde kullanılırken, maderşahi için kullanılan merkez pozisyon denklemleri göz ardı edildiği için  $\beta$  parametresi önerilen yöntem için dikkate alınmamıştır. Multi-EHO' da temel EHO denkleminin yanı sıra kullanılan Denklem (6)'da ki  $c_1$  ve  $c_2$  parametreleri sabit ve 1.5 olarak belirlenmiştir. Multi-EHO' da klan sayısı, temel EHO'nun orijinal çalışmasında olduğu gibi 5 olarak seçilmiş bu nedenle her bir klanla bir arama stratejisi atanmıştır. Klanlar, tüm süreç boyunca kendilerine atanan arama stratejisi ile güncellenmiştir.

Karşılaştırmalarda kullanılan tüm yöntemlerde popülasyon sayısı 40, maksimum fonksiyon değerlendirme sayısı ise  $3E+5$  olarak ayarlanmıştır.

##### 4.3. Multi-EHO ve temel EHO yöntemlerinin karşılaştırılması

Önerilen yöntem üzerinde yapılan iyileştirmelerin etkisini gösterebilmek için ilk olarak temel EHO ile karşılaştırılması yapılmıştır. Yöntemler CEC2015' teki test fonksiyonları için

belirtilen şartlarda 25 kez çalıştırılmış, bu çalıştırmalar sonucu elde edilen ortalama ve standart sapma değerleri Tablo 2’de paylaşılmıştır. Fonksiyonlar için en iyi çözümü veren ortalama ve standart sapma değerleri kalın yazı tipi ile gösterilmiştir. Ayrıca sonuçlar arasında istatistiksel olarak anlamlı bir fark olup olmadığını test edebilmek için iki yöntemin 25 çalıştırma sonuçlarına Wilcoxon testi uygulanmıştır. Tablo 2’de verilen sonuçlar için Wilcoxon testi 0.05 anlamlılık düzeyinde ise +, olmaması durumunda ise – işareti kullanılmıştır.

**Tablo 2.** Multi-EHO ve temel EHO’nun karşılaştırması

Fonksiyon	EHO		Multi-EHO		
	Ort.	Std.Sap.	Ort.	Std.Sap.	
F1	2.53E+10	2.47E+09	<b>4.54E+03</b>	<b>6.82E+03</b>	+
F2	<b>3.53E+04</b>	<b>3.10E+03</b>	3.79E+04	1.02E+04	-
F3	3.61E+01	1.32E+00	<b>3.19E+01</b>	<b>4.12E+00</b>	+
F4	6.65E+03	3.08E+02	<b>4.16E-01</b>	<b>4.74E-01</b>	+
F5	2.59E+00	2.22E-01	<b>1.30E+00</b>	<b>5.38E-01</b>	+
F6	3.85E+00	1.96E-01	<b>5.16E-01</b>	<b>1.27E-01</b>	+
F7	6.56E+01	1.28E+01	<b>4.61E-01</b>	<b>2.84E-01</b>	+
F8	6.46E+05	2.49E+05	<b>4.13E+01</b>	<b>1.74E+01</b>	+
F9	1.30E+01	1.36E-01	<b>1.27E+01</b>	<b>3.70E-01</b>	+
F10	7.91E+06	2.77E+06	<b>8.83E+04</b>	<b>6.14E+04</b>	+
F11	1.09E+02	1.64E+01	<b>3.80E+01</b>	<b>2.85E+01</b>	+
F12	9.08E+02	1.78E+02	<b>7.46E+02</b>	<b>2.51E+02</b>	+
F13	5.63E+02	3.28E+01	<b>3.28E+02</b>	<b>1.62E-11</b>	+
F14	2.99E+02	9.18E+00	<b>2.48E+02</b>	<b>3.20E+01</b>	+
F15	1.21E+03	7.45E+01	<b>1.18E+03</b>	<b>8.13E+01</b>	-

Tablo 2 incelendiğinde önerilen yöntemin 15 test fonksiyonunun 14’ünde daha iyi sonuçlar elde ederek temel EHO’ya göre üstün olduğu görülmektedir. Çoklu arama stratejisi sayesinde farklı karakteristikteki problemler içinde başarısını sürdürmektedir. Özellikle F1, F4 ve F8 fonksiyonlarında iki yöntemin ortalama sonuçları arasındaki büyük fark göze çarpmaktadır. Tablo 2’de verilen Wilcoxon test sonuçları değerlendirildiğinde sadece F2 ve F15 hariç tüm fonksiyonlar için sonuçların istatistiksel olarak anlamlı olduğu gözlemlenmektedir.

Önerilen yöntemde çoklu arama stratejisinin eş zamanlı olarak değil her klan için bir denklem şeklinde kullanılması nedeniyle temel EHO ve Multi-EHO ‘nun hesaplama zamanları arasında büyük bir fark oluşması beklenmemektedir. Bu ifadeyi doğrulamak için iki yönteme ait 25 çalıştırma sonucu elde edilen ortalama hesaplama zamanları Tablo 3’de gösterilmektedir. Tablo 3’de verilen hesaplama zamanları değerlendirildiğinde, Multi-EHO’nun daha az hesaplama zamanı ile çalıştığı görülmektedir. Bunun

nedeni ise temel EHO yönteminde her klandaki maderşahinin pozisyon güncelleme işleminde kullanılacak olan merkez pozisyonunun hesaplanması olarak gösterilebilir.

**Tablo 3.** Multi-EHO ve temel EHO’nun hesaplama zamanları

Fonksiyon	EHO	Multi-EHO
F1	22.92	17.87
F2	23.26	18.08
F3	48.64	42.60
F4	25.81	17.88
F5	39.26	31.85
F6	24.99	18.31
F7	25.62	18.09
F8	25.53	18.38
F9	26.15	18.69
F10	26.78	18.69
F11	30.49	24.10
F12	26.34	20.52
F13	28.24	21.64
F14	27.89	20.28
F15	57.34	46.05
Toplam	459.27	353.02
Ortalama	30.62	23.53

#### 4.4. Multi-EHO ve diğer yöntemlerin karşılaştırılması

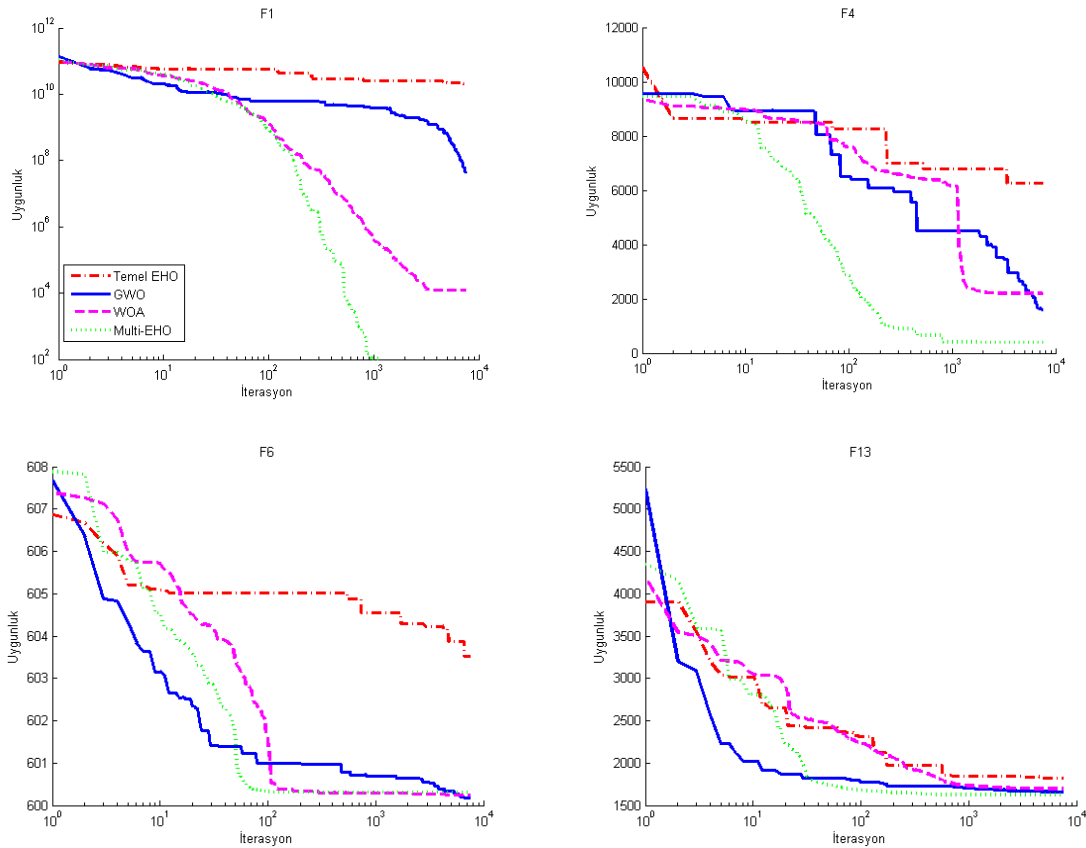
Multi-EHO’nun temel EHO’ya karşı elde ettiği başarılı performansı doğrulamak için, Multi-EHO son yıllarda önerilen iki yeni algoritma GWO [6] ve WOA [7] ile karşılaştırılmıştır. Yöntemlere ait 25 çalıştırma sonucu elde edilen ortalama ve standart sapma değerleri Tablo 4’te verilmektedir. Fonksiyonlar için elde edilen en iyi ortalama ve standart sapma değerleri kalın yazı tipi ile gösterilmiştir. Tablo 4 genel olarak incelendiğinde, WOA yönteminin diğer iki yönteme göre geride kaldığı ve sadece bir fonksiyon için en başarılı sonucu elde ettiği görülmektedir. Multi-EHO ise F1, F4, F5, F8, F10 ve F13 numaralı fonksiyonlarda diğer algoritmalara göre daha iyi performans sergilemektedir.

Yöntemlerin performanslarını açıkça karşılaştırabilmek için, Tablo 4’te verilen sonuçlara Friedman sıra testi uygulanarak teste ait sonuçlar aynı tabloda gösterilmiştir. Friedman testine göre 1.70 puan ile yöntemler arasında ilk sırayı Multi-EHO almaktadır. Önerilen yöntemden sonra az bir farkla GWO yöntemi gelirken, son sırada ise WOA algoritması yer almaktadır.

Yöntemlerin yakınsama performanslarını değerlendirmek için, farklı karakteristikteki F1, F4, F6, ve F13 fonksiyonlarına ait yakınsama grafikleri Şekil 2’de gösterilmektedir.

**Tablo 4.** Multi-EHO ve diğer yöntemlerin karşılaştırılması

Fonksiyon	GWO		WOA		Multi-EHO	
	Ort.	Std.Sap.	Ort.	Std.Sap.	Ort	Std.Sap.
F1	8.35E+08	8.08E+08	4.77E+05	9.73E+05	<b>4.54E+03</b>	<b>6.82E+03</b>
F2	<b>2.46E+04</b>	<b>5.93E+03</b>	1.49E+05	2.72E+04	3.79E+04	1.02E+04
F3	<b>1.53E+01</b>	<b>3.06E+00</b>	3.54E+01	3.75E+00	3.19E+01	4.12E+00
F4	2.22E+03	6.35E+02	3.96E+03	9.35E+02	<b>4.16E-01</b>	<b>4.74E-01</b>
F5	2.42E+00	7.03E-01	1.62E+00	3.99E-01	<b>1.30E+00</b>	<b>5.38E-01</b>
F6	<b>2.75E-01</b>	<b>7.05E-02</b>	5.12E-01	1.11E-01	5.16E-01	1.27E-01
F7	7.77E-01	1.53E+00	<b>3.64E-01</b>	<b>1.83E-01</b>	4.61E-01	2.84E-01
F8	1.68E+02	3.33E+02	6.82E+01	2.67E+01	<b>4.13E+01</b>	<b>1.74E+01</b>
F9	<b>1.13E+01</b>	<b>5.58E-01</b>	1.31E+01	4.17E-01	1.27E+01	3.70E-01
F10	3.83E+05	3.28E+05	2.29E+06	1.55E+06	<b>8.83E+04</b>	<b>6.14E+04</b>
F11	<b>1.95E+01</b>	<b>2.92E+00</b>	4.06E+01	3.03E+01	3.80E+01	2.85E+01
F12	<b>1.66E+02</b>	<b>8.31E+01</b>	8.27E+02	3.37E+02	7.46E+02	2.51E+02
F13	3.86E+02	2.40E+01	4.65E+02	3.44E+01	<b>3.28E+02</b>	<b>1.62E-11</b>
F14	<b>2.31E+02</b>	<b>6.56E+00</b>	2.61E+02	2.45E+01	2.48E+02	3.20E+01
F15	<b>6.71E+02</b>	<b>8.22E+01</b>	1.18E+03	1.74E+02	1.18E+03	8.13E+01
Friedman Puan	1.73		2.57		1.70	
Sıra	<b>2</b>		<b>3</b>		<b>1</b>	

**Şekil 2.** Yöntemlere ait yakınsama grafikleri

Şekil 2 incelendiğinde, özellikle tek modlu yapıdaki F1 fonksiyonu için Multi-EHO'nun yerel aramayı çok daha iyi yaparak daha hızlı yakınsadığı görülmektedir. F1 fonksiyonunda temel EHO'nun çözümü neredeyse hiç iyileştiremediği gözlemlenmektedir. Önerilen yöntem, çok modlu karakteristiğe sahip F4 fonksiyonunda iterasyonun sonlarına doğru bir durağanlık yaşasa da diğer yöntemlere göre daha başarılı bir performansa sahiptir. F6 fonksiyonunda, GWO diğer yöntemlere göre daha erken yakınsamasına rağmen WOA ve Multi-EHO yöntemlerine iterasyonun ortalarında yakalanmıştır. Buna rağmen çözüm iyileştirmeye devam ederek diğer yöntemleri geride bırakmıştır. F13 fonksiyonun da ise GWO ve önerilen yöntem performans olarak öne çıkarken, temel EHO'nun ara durağanlıklar yaşadığı gözlemlenmiştir.

## 5. SONUÇ

Bu çalışmada, yeni bir optimizasyon algoritması olan fil sürü optimizasyonunun tek bir arama stratejisi kullanması nedeniyle arama uzayını etkin araştıramaması ve global arama konusundaki yetersiz kalması sorunlarının üstesinden gelmek için çoklu arama stratejisi kullanan fil sürü optimizasyonu önerilmiştir. Arama stratejileri belirlenirken, yerel arama konusunda ön plana çıkan PSO ve global arama yeteneği iyi olan ABC algoritmalarının yanı sıra farklı arama stratejileri bulunan DE yönteminden yararlanılmıştır. Önerilen yöntemde, çoklu arama stratejisi uygulanırken her bir klanı bir strateji atanmış böylece popülasyon içinde her grubun arama uzayını farklı şekillerde araştırması sağlanmıştır. Multi-EHO'nun farklı karakteristikteki problem üzerindeki performansını değerlendirmek için CEC2015 test seti kullanılmıştır. Elde edilen sonuçlar hem temel EHO hem de yeni çıkan iki farklı optimizasyon tekniği ile karşılaştırılmıştır. Deneysel sonuçlar, önerilen yöntemin temel EHO'ya göre çok üstün bir performansa sahip olduğunu ve GWO ve WOA yöntemlerine göre daha başarılı olduğunu göstermiştir. Bu çalışma ve buna benzer çalışmalardan yola çıkılarak, optimizasyon teknikleri için tek bir arama stratejisi yerine çoklu arama stratejisi uygulanmasının algoritmaların performansını olumlu yönde iyileştirdiği ve geliştirdiği söylenebilir.

Gelecek çalışmalar için ise, farklı arama stratejileri kullanılarak Multi-EHO ile performans karşılaştırılması yapılabilir ve diğer sürü zekası yöntemler için çoklu arama stratejisi uygulanabilir.

## Teşekkür

Bu çalışma Necmettin Erbakan Üniversitesi Bilimsel Araştırma Projesi tarafından desteklenmiştir.

## KAYNAKÇA

[1] H. Hakli and H. Uguz, "A novel particle swarm optimization algorithm with Levy flight," *Applied Soft Computing*, vol. 23, pp. 333-345, Oct 2014.

- [2] I. Strumberger, N. Bacanin, and M. Tuba, "Hybridized Elephant Herding Optimization Algorithm for Constrained Optimization," *Cham*, 2018, pp. 158-166.
- [3] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," *Technical Report-TR06*, Erciyes University, Engineering Faculty, Comput. Eng.Dep.2005.
- [4] J. Kennedy and R. Eberhart, "Particle swarm optimization," presented at the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 1995.
- [5] M. Dorigo and G. D. Caro, "Ant colony optimization: a new meta-heuristic," presented at the Proceedings of the 1999 Congress on Evolutionary Computation, Washington, DC., 1999.
- [6] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46-61, Mar 2014.
- [7] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51-67, May 2016.
- [8] G. G. Wang, S. Deb, and L. D. Coelho, "Elephant Herding Optimization," *2015 3rd International Symposium on Computational and Business Intelligence (Iscbi 2015)*, pp. 1-5, 2015.
- [9] I. Strumberger, N. Bacanin, S. Tomic, M. Beko, and M. Tuba, "Static Drone Placement by Elephant Herding Optimization Algorithm," *2017 25th Telecommunication Forum (Telfor)*, pp. 808-811, 2017.
- [10] E. Tuba, A. Alihodzic, and M. Tuba, "Multilevel Image Thresholding Using Elephant Herding Optimization Algorithm," *2017 14th International Conference on Engineering of Modern Electric Systems (Emes)*, pp. 240-243, 2017.
- [11] E. Tuba and Z. Stanimirovic, "Elephant Herding Optimization Algorithm for Support Vector Machine Parameters Tuning," *Proceedings of the 9th International Conference on Electronics, Computers and Artificial Intelligence - Ecai 2017*, 2017.
- [12] A. Alihodzic, E. Tuba, R. Capor-Hrosik, E. Dolicanin, and M. Tuba, "Unmanned Aerial Vehicle Path Planning Problem by Adjusted Elephant Herding Optimization," *2017 25th Telecommunication Forum (Telfor)*, pp. 804-807, 2017.
- [13] M. A. Sarwar, B. Amin, N. Ayub, S. H. Faraz, S. U. R. Khan, and N. Javaid, "Scheduling of Appliances in Home Energy Management System Using Elephant Herding Optimization and Enhanced Differential Evolution," *Advances in Intelligent Networking and Collaborative Systems, Incos-2017*, vol. 8, pp. 132-142, 2018.
- [14] D. K. Sambariya and R. Fagna, "A Robust PID Controller for Load Frequency Control of Single Area Re-heat Thermal Power Plant using Elephant Herding Optimization Techniques," *2017 Ieee International Conference on Information, Communication, Instrumentation and Control (Icic)*, 2017.
- [15] V. Tuba, M. Beko, and M. Tuba, "Performance of Elephant Herding Optimization Algorithm on CEC 2013 real parameter single objective optimization," *WSEAS*

TRANSACTIONS on SYSTEMS, vol. 16, pp. 100-105, 2017.

[16] S. Parashar, A. Swarnkar, K. R. Niazi, and N. Gupta, "A modified elephant herding optimization for economic generation co-ordination of DERs and BESS in grid connected microgrid," *Journal of Engineering-Joe*, Nov 15 2017.

[17] N. K. Meena, S. Parashar, A. Swarnkar, N. Gupta, and K. R. Niazi, "Improved Elephant Herding Optimization for Multiobjective DER Accommodation in Distribution Systems," *IEEE Transactions on Industrial Informatics*, vol. PP, 2017.

[18] E. Tuba, R. Capor-Hrosik, A. Alihodzic, R. Jovanovic, and M. Tuba, "Chaotic Elephant Herding Optimization Algorithm," presented at the IEEE 16th World Symposium on Applied Machine Intelligence and Informatics, Kosice, Slovakia, 2018.

[19] R. Storn and K. Price, "Differential evolution-A simple and efficient adaptive scheme for global optimization over continuous spaces," Berkeley: ICSI, 1995.

[20] G. G. Wang, S. Deb, X. Z. Gao, and L. D. Coelho, "A new metaheuristic optimisation algorithm motivated by elephant herding behaviour," *International Journal of Bio-Inspired Computation*, vol. 8, pp. 394-409, 2016.

[21] H. Wang, Z. J. Wu, S. Rahnamayan, H. Sun, Y. Liu, and J. S. Pan, "Multi-strategy ensemble artificial bee colony

algorithm," *Information Sciences*, vol. 279, pp. 587-603, Sep 20 2014.

[22] M. S. Kiran, H. Hakli, M. Gunduz, and H. Uguz, "Artificial bee colony algorithm with variable search strategy for continuous optimization," *Information Sciences*, vol. 300, pp. 140-157, Apr 10 2015.

[23] H. Hakli, "A modified cuckoo search using different search strategies," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 4 (Special Issue), pp. 190-194, 2016.

[24] Y. Wang, B. Li, T. Weise, J. Y. Wang, B. Yuan, and Q. J. Tian, "Self-adaptive learning based particle swarm optimization," *Information Sciences*, vol. 181, pp. 4515-4538, Oct 15 2011.

[25] H. Hakli, "An improved elephant herding optimization by balancing local and global search for continuous optimization," presented at the 15th International Conference on Informatics and Information Technologies, CIIT 2018, Mavrovo, Macedonia, 2018.

[26] J. J. Liang, B. Y. Qu, P. N. Suganthan, and Q. Chen, "Problem Definitions and Evaluation Criteria for the CEC 2015 Competition on Learning-based Real-Parameter Single Objective Optimization," *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China And Technical Report, Nanyang Technological University, Singapore* 2014.