

## GSA ALGORİTMASININ DEĞİŞKENLERİNİN İNCELENMESİ VE EN UYGUN DEĞERLERİNİN TESPİTİ

Uğur GÜVENÇ<sup>1</sup>, Ferzan KATIRCIOĞLU<sup>2</sup>

<sup>1</sup> Düzce Üniversitesi, Teknoloji Fakültesi, Elektrik Müh. Böl., 81620 Düzce, TÜRKİYE

<sup>2</sup> Düzce Üniversitesi, Düzce M YO, Elektronik ve Otomasyon Böl., 81100, Düzce, TÜRKİYE  
[ugurguven@duzce.edu.tr](mailto:ugurguven@duzce.edu.tr)

**Özet-**Yerçekimi arama algoritması (GSA) fizikte, kütleyle sahip nesnelere birbirine doğru hızlanma eğilimi özelliğini esas alır ve içinde birbirlerine bağlı birçok parametre ve değişken vardır. GSA'nın performansı, onun parametre ve değişkenlerine doğrudan bağlıdır. Bundan dolayı algoritmanın davranışı ve parametreleri arasındaki nitel ve nicel ilişkinin iyi bir analizi gereklidir. Bu çalışmada, GSA'nın modifikasyonunu yapabilmek için değişkenlerin algoritmaya etkisi incelenmiş ve en uygun olabilecek değişken sonuçları tespit edilmiştir.

**Anahtar Kelimeler-** Yerçekimi Arama Algoritması(GSA), Optimizasyon, Yerçekimi Sabiti

## INVESTIGATION VARIABLES OF GSA ALGORITHM AND THE DETERMINATION OF FAIR VALUE

**Abstract-**Gravitational search algorithm GSA is based property which objects with mass accelerating trend toward one another in physics and has many parameters and variables that connected to each other. GSA's performance is directly linked to its parameters and variables. Therefore, the behavior of the algorithm and among the parameters is required a qualitative and quantitative analysis of the relation. In order to make modifications GSA, in this study is examined the effect of variables to algorithm and then is found to be most suitable variable results.

**Key Words-** Gravitational Search Algorithm, Optimization, Gravitational Constant

### 1. GİRİŞ (INTRODUCTION)

Doğadaki sistemlerin ve canlıların davranışlarından esinlenerek ortaya konulmuş bir çok optimizasyon yöntemi çeşitli problemlerin çözümünde kullanılmaktadır [1-5]. Yerçekimi kanununu ve kütle etkileşimlerinden esinlenilerek gerçekleştirilmiş, fizik tabanlı sezgisel stokastik optimizasyon algoritması olan Yerçekimsel Arama Algoritması (GSA) ilk olarak Rashedi ve arkadaşları tarafından sunulmuştur [6]. 2009 yılında geliştirilen bu optimizasyon algoritması literatürdeki bir çok problemin çözümünde kullanılmıştır [7-12]. GSA optimizasyon algoritmasının geliştirilmesi amacıyla da içindeki değişken ve parametreler üzerinden değişiklik ve iyileştirmeler yapılarak, birçok modifikasyon çalışmaları yapılmıştır. Bunlardan ilkinde, kendinden uyarlamalı parametre ayar şeması ve rastgele üretim ile parametre ayarı gerçekleştirilmiştir [13]. Li ve arkadaşları geleneksel üstel sabit fonksiyonun yerine yerçekimi sabiti olarak bir parçalı fonksiyon tasarlamışlardır[14]. Başka bir çalışma içerisinde, GSA'nın içerisine kara delik operatörü yerleştirilmiş ve bir yerçekimi sistemi içerisinde ağır nesnelere

\* [ugurguven@duzce.edu.tr](mailto:ugurguven@duzce.edu.tr)

bazıları yıldızdır varsayımı yapılmıştır [15]. Rashedi ve Nezamabadi-pour fizik kuralları içerisindeki hızlı kaçış konseptinden esinlenerek seyahat, birleştirme ve kaçma şeklinde üç operatör tanıtmışlar ve görüntü işlemede bölütlüme çalışması içerisinde kullanılmışlardır [16]. Doğrusal olmayan filtre modellemede kanal gürültülerini azaltmak için parametre tahmininde kullanılan bu çalışmada, standart GSA'nın içerisine kaotik operatör yerleştirilmiştir [17]. Naji ve arkadaşları yüksek performans seviyesini koruyorken GSA'nın hızını geliştirmek için çoklu etken sisteminin sıralı uygulanmasını sağlamışlardır [18]. Ajanların çeşitliliğini yükseltmek için antikör çeşitliliği ve yakınsama hızını artırmak için aşı gibi hafıza antikoru kullanılmıştır. Yerel uygun çözüme düşmekten kaçınmak için GSA'nın yavaş yakınsama hızı geliştirilmiştir [19]. Araştırmanın yakınsama hızını artırmak için, her bir döngü süreci içerisindeki her bir etkene ağırlaştırılmış değer olarak atalet kütlesi atanmıştır [20]. Nobahari ve arkadaşları sürü içerisindeki çeşitliliği teşvik etmek için işaretleme ve tekrar sıralama önerilmiştir. Bu çalışma içerisinde parçacıkların yerçekimsel ivmesini güncellemek için bağımsız sıralama kavramına yer verilmiştir [21]. Bozulma olarak adlandırılan özünü astrofizikken alan bir operatör kullanılmıştır. Kütleli özel eşik değerinden daha küçük değerlere kesilmelidir. Böylece GSA'nın keşif ve sömürü yetenekleri geliştirilmiştir [22]. Sombra ve arkadaşları döngü içerisindeki alfa parametresinin değişimi üzerine çalışmışlardır [23]. Vijaya ve arkadaşları tarafından bulanık uyarlamalı GSA'da bulanık IF/THEN kuralları kullanarak yerçekimi sabiti(G) ayarlanmıştır. Zor olan G parametresinin uygun seçimi gerçekleştirilmiştir [24].

Bu çalışmada, yerçekimi arama algoritmasındaki değişkenlerin algoritmaya etkisi incelenmiş ve inceleme sonucunda en iyi aralıklar tespit edilmiştir. İkinci bölümde, algoritmaya sahip değişkenler tanımlanmış ve yapılan deneylere yer verilmiştir. Üçüncü bölümde elde edilen deney sonuçları her bir değişken için tablo olarak sunulmuş, elde edilen tespitlere yer verilmiştir. Son bölümde ilgili değişkenler için elde edilen sonuçlar karşılaştırılıp en uygun seçme aralıkları verilmiştir. Parametreler arama sonuçlarına (en uygun değere) ve arama sürelerine nasıl etki ediyor, parametrelerin en uygun değerleri için ne söylenebilir gibi sorulara yanıtlar aranmıştır.

## 2. YÖNTEM (METHOD)

### 2.1. Yerçekimi Arama Algoritması (Gravitational search algorithm)

Bu bölümde, yerçekimi kanununu esas alan optimizasyon algoritması tanımlanmıştır. Ajanlar bir nesne gibi düşünülmüş ve onların performansları kütleleri ile ölçeklendirilmiştir. Bütün bu nesnelere yerçekimi kuvveti ile birbirlerini çekerler ve bu kuvvet daha ağır kütleli nesnelere doğru bütün nesnelere hareketine sebep olur (3). Şimdi, N ajanlı(kütleli) bir sistem düşünelim.  $i$ . ajanın konumunu aşağıdaki ifade ile tanımlanır:

$$X_i = (x_i^1, \dots, x_i^d, \dots, x_i^n) \quad i = 1, 2, \dots, N \quad (1)$$

Formüldeki  $n$  problemin boyutunu ve  $x_i^d$  d. boyut içerisinde  $i$ . ajanın pozisyonu şeklinde tanımlanır. Ajanların çözümüne başlama noktası rastgele yapılmaktadır. Belirli bir  $t$  zamanında  $j$  kütlelerinden  $i$  kütleli üzerindeki kuvvet hareketi aşağıdaki gibi tanımlanır.

$$F_{ij}^d(t) = G(t) \frac{M_{pi}(t)xM_{aj}(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)), \quad (2)$$

$$R_{ij}(t) = \|X_i(t), X_j(t)\|_2 \quad (3)$$

Denklem (3) de  $i$  ve  $j$  arasındaki öklit mesafesidir. Algoritmaya stokastik özelliği vermek için  $d$  boyutu içerisindeki  $i$  ajanının üzerindeki toplam kuvvet  $d$ . komponentlerin rastgele ağırlaştırılmış toplamları şeklinde ifade edilmiştir.

$$F_i^d(t) = \sum_{j=1, j \neq i}^N rand_j F_{ij}^d(t). \quad (4)$$

$rand_j$  [0,1] olabilir ve rastgele belirlenebilir. Yukarıdaki eşitliği ve hareket kanunu kullanarak  $i$  ajanının ivmesi şeklinde yazılabilir.

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \quad (5)$$

Eşitlik içerisindeki  $M_{ii}$  ifadesi  $i$  ajanının atalet kütesini temsil etmektedir. Ayrıca ajanın gelecek hızı şu anki hızına ivmesi eklenmiş bir parçasıymış gibi düşünülebilir. Bu yüzden ajanın pozisyonu ve hızı aşağıdaki gibi hesaplanabilir.

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \quad (6)$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (7)$$

Yine yukarıdaki ilk eşitlikteki  $rand_i$  [0,1] aralığında rastgele bir değer olup, araştırmaya rastgele bir karakteristik vermek için kullanılmıştır.

Yerçekimi sabiti olan  $G$ , arama doğruluğunu kontrol etmek için azaltılacak ve başlangıçta sabit bir değer ile başlatılmıştır. Başka bir ifade ile  $G$ 'nin ilk değeri ( $G_0$ ) olup ve zamanın bir fonksiyonudur.

$$G(t) = G(G_0, t) \quad (8)$$

Yerçekimi ve atalet kütlelerinin eşit olduklarını varsayarsak, kütlelerin değerlerini uygunluk haritası kullanarak hesaplayabilir. Aşağıdaki eşitlik ile yerçekimi ve atalet kütleleri güncellenebilir.

$$M_{ai} = M_{pi} = M_{ii} = M_i \quad i = 1, 2, \dots, N \quad (9)$$

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (10)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (11)$$

$fit_i(t)$   $t$  zamanında  $i$  ajanının uygunluk değerini temsil eder ve  $worst(t)$  ve  $best(t)$  aşağıdaki şekilde tanımlanır.

$$best(t) = \min_{J \in \{1, \dots, N\}} fit_j(t) \quad (12)$$

$$worst(t) = \max_{J \in \{1, \dots, N\}} fit_j(t) \quad (13)$$

Maksimizasyon problemi için yukarıdaki eşitlikler sırasıyla yer değiştirerek aşağıdaki eşitliklerde kullanılabilir.

$$best(t) = \max_{j \in \{1, \dots, N\}} fit_j(t) \quad (14)$$

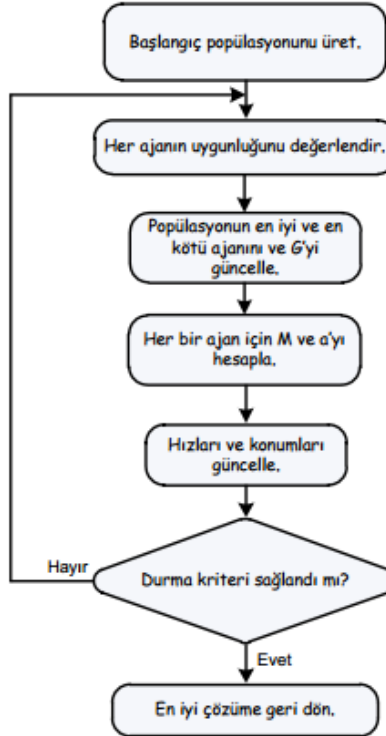
$$worst(t) = \min_{j \in \{1, \dots, N\}} fit_j(t) \quad (15)$$

Araştırma ve kötüye kullanım arasında iyi bir uzlaşma yolu için Eş. 4 yerine, büyük kuvvetler uygulayan ajanların kümesi önerilmiştir.  $Kbest$ ,  $Ko$  başlangıç değeri ile birlikte ve zamanla azalan zamanın bir fonksiyonudur. Böyle bir yol içerisinde başlangıçta, bütün ajanlar kuvvet uygular ve zaman geçtikçe  $Kbest$  doğrusal olarak azalır. Ve sonunda, orada diğerlerine kuvvet uygulayan tek bir ajan olacaktır. Bundan dolayı Eş. 4 tekrar aşağıdaki gibi düzenlenmiştir[1].

$$F_i^d(t) = \sum_{j \in Kbest, j \neq i}^N rand_j F_{ij}^d(t) \quad (16)$$

Eşitlikteki  $Kbest$  en iyi uygunluk değeri ve en büyük kütle ile birlikte ilk  $K$  etkenleri kümesidir. GSA algoritmasının adımları aşağıda verilmiştir ve ayrıca prensip şeması Şekil 1.'de gösterilmiştir.

- Arama alanını tanımlama
- Rastgele başlatma
- Etkenlerin uygunluk değerlendirilmesi
- $i = 1, 2, \dots, N$  için  $G(t)$ ,  $best(t)$ ,  $worst(t)$  ve  $M_i(t)$  değerlerini tekrar güncelleme
- Farklı yönler içerisindeki toplam kuvveti hesaplama
- İvme ve hızı hesaplama
- Etkenlerin pozisyonlarını güncelleme
- c'den g'ye kadar adımları verilen kritere ulaşana kadar tekrarlama
- Sonlandırma



Şekil 1. Yerçekimi arama algoritmasının akış şeması (Gravitational search algorithm flowchart)

## 2.2. Yerçekimi Sabitinin İncelenmesi (Investigation of Gravity Constant)

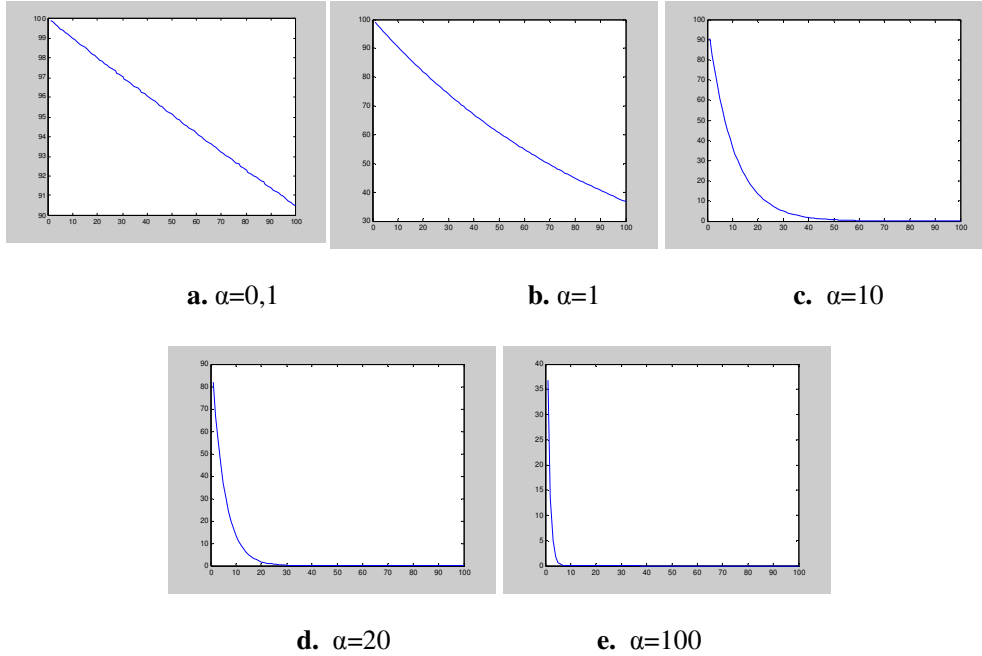
Yerçekimi sabiti olan  $G$ , arama doğruluğunu kontrol etmek için azaltılacak ve başlangıçta sabit bir değer ile başlatılmıştır. Başka bir ifade ile  $G$ 'nin ilk değeri ( $G_0$ ) olup ve zamanın bir fonksiyonudur.

$$G(t) = G_0 \exp\left(-\alpha \frac{t}{\max\_t}\right) \quad (17)$$

Yerçekimi arama algoritmasının bütün parametreleri sabit tutulmuş, sadece yerçekimi sabiti  $G_0$  1'den başlanılarak 10, 100, 1000 ve 10000' e kadar beş farklı değer üzerinden değiştirilerek sonuç değerleri ve optimizasyon bitiminde geçen süre veri olarak kaydedilmiştir.

## 2.3. $\alpha$ Parametresinin İncelenmesi (Investigation of the parameter $\alpha$ )

Eş. 17 de yer alan  $\alpha$  parametresinin yerçekimi sabiti üzerindeki etkisini incelemek için diğer optimizasyon parametreleri sabit tutulmuş,  $\alpha$  değeri 0,1, 1, 10, 20 ve 100 değerlerinin her biri için işlem yapılmıştır.



**Şekil 2.**  $\alpha$  parametresinin yerçekimi sabiti üzerine etkileri (Effects on the gravitational constant of the  $\alpha$  parameter)

## 2.4. Ajan Sayısının İncelenmesi (Investigation of the Number of Agents)

Eş. 1 de ifade edilen sistem içerisinde en uygun değeri arayan ve kütlesi, ivmesi ve hızı olan arama operatörüne ajan denir ve  $N$  ajanlı(kütleli) bir sistem içerisinde düşünülür. Ajanların çözümüne başlama noktası rastgele yapılmaktadır. Bu gözlem aşamasında, daha önce tespit edilen en iyi  $G_0$  performansı olan 100 ile en iyi  $\alpha$  sabitinin 10 değerleri altında Tablo 1. de verilen F3 ve F6 test fonksiyonları sırasıyla amaç fonksiyonu olarak kullanılmış, ajan sayısı 1, 10, 30, 100 ve 500 değerleri kullanılarak her birinden 10'ar ölçüm alınmıştır.

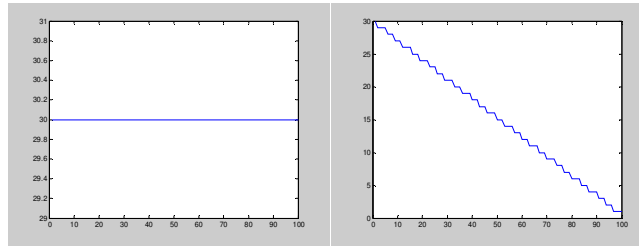
## 2.5. Döngü Sayısının İncelenmesi (Investigation of Cycle Count)

Bu bölümdeki deneysel çalışmada, döngü sayısının çok az verilmesi veya çok yüksek değerde verilmesi algoritmanın sonuçlarında ne gibi etkileri olacağı amaçlanmıştır. Algoritmanın farklı test fonksiyonlarında dışarıdan verilen döngü sayısı en iyi sonucu bulma noktasında önemli bir değişkendir. Daha önce tespit edilen en iyi  $G_0$  performansı 100, en iyi  $\alpha$  sabitinin 10 ve en iyi ajan sayısı 30 değerleri sabit tutularak, farklı test fonksiyonlarında döngü sayısı 1, 10, 50, 100 ve 500 değerleri verilerek GSA algoritmasına uygulanmıştır.

## 2.6. Kbest Ajanları İncelenmesi (Investigation of Kbest Agents)

GSA algoritmasında stokastik özelliği vermek için  $d$  boyutu içerisindeki  $i$  ajanının üzerindeki toplam kuvvet hesaplanırken iki şekilde gerçekleştirilir. Birincisinde Rashedi, Eş. 4 de görüldüğü gibi bütün ajanların  $i$  ajanına etkiyen kuvvetlerinin toplamları şeklinde ifade edilmiştir. İkincisinde Eş. 16 da, her döngüde ajan sayısını azaltarak arama ve sömürüyü kontrol ederek GSA'nın performansı artırılmak istenmiştir. Kbest zamanın bir fonksiyonudur, başlangıçta bütün ajanlar ilgili ajana kuvvet uygulayacak ve zaman geçtikçe her döngüde Kbest doğrusal olarak azalacaktır. En son döngüde programda belirtilecek final\_per kadar ajan, ilgili ajana etkiyecektir.

GSA Algoritmasında  $G_0=100$ ,  $\alpha=10$ , döngü=100,  $N=30$ , boyut=30 değişkenleri kendileri için en uygun değerler olarak seçilmiş olup, deney süresince değiştirilmemiştir. Deney farklı test fonksiyonlarına uygulanmıştır. Öncelikle her döngüde ilgili ajana etkiyen kuvvetler hesaplanırken bütün ajanlar dikkate alınmıştır. İkinci aşamada Kbest ajan sayısından başlatılmış, her döngüde azaltılarak son döngüde final\_per=2 olana kadar düşürülmüştür.



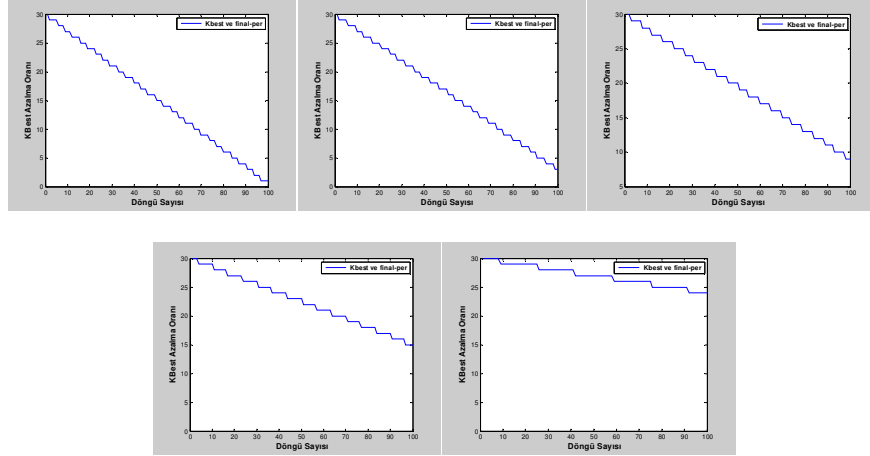
Şekil 3. a. Her döngüde  $N$  kadar ajan kuvvete etkimesi b. Kbest kadar ajan kuvvete etkimesi

a. The effect of each cycle in  $N$  as agent force b. The effect until the agent force Kbest

Şekil 3.(a) da ilgili ajana bütün ajanların her döngüde etkidiğini, Şekil 3.(b) de ise her döngü içerisinde ajan sayısının azaldığını gösteren grafikler verilmiştir. Son döngüde ajan sayısını küçük bir değerde kaldığı görülmektedir.

## 2.7. final\_per Değerinin İncelenmesi (Investigation of the final\_per Value)

İlgili ajana bütün ajanların etkidiği kuvveti katmak yerine, her döngüde ajan sayısını düşürerek kütlesi fazla olan ajanlar etkime işlemini gerçekleştiren Kbest yaklaşımında son döngüde final\_per kadar ajan kalmasına denir. Bu çalışmada final\_per sabiti %2, %10, %30, %50 ve %80 vererek en iyi sonuçlara etkisine bakılmıştır. GSA Algoritmasında  $G_0=100$ ,  $\alpha=10$ , döngü=100,  $N=30$ , boyut=30 değişkenleri kendileri için en uygun değerler olarak seçilmiş olup, deney süresince değiştirilmemiştir. Şekil 4 (a), (b), (c), (d) ve (e) de görüldüğü gibi final\_per sabitinin en son döngüde katılması gereken ajan sayısını göstermektedir.



Şekil 4. a. final\_per=%2 b. final\_per=%10 c. final\_per=%30 d. final\_per=%50  
e. final\_per=%80

### 3. BULGULAR (FINDINGS)

Deneysel çalışmalarda yerçekimi arama algoritmasının performansını değerlendirmek için, algoritma Tablo 1 de verilen 8 test fonksiyonu kullanılmıştır. Bunlardan F1-4 tek modelli test fonksiyonlar grubuna girerken son dördü çok modelli fonksiyonları grubuna girmektedir.

**Tablo 1.** Deneysel çalışmalarda kullanılan test fonksiyonları (Test functions used in experimental studies)

NO	TEST FONKSİYONLARI	BOYUT
1	$F_1(X) = \sum_{i=1}^n (\sum_{j=1}^i X_j)^2$	$[-100,100]^n$
2	$F_2(X) = \max\{ X_i , 1 \leq i \leq n\}$	$[-100,100]^n$
3	$F_3(X) = \sum_{i=1}^{n-1} [100(X_{i+1} - X_i^2)^2 + (X_i - 1)^2]$	$[-30,30]^n$
4	$F_4(X) = \sum_{i=1}^n ( X_i + 0.5 )^2$	$[-100,100]^n$
5	$F_5(X) = \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\}$ $+ \sum_{i=1}^n u(X_i, 10, 100, 4)$ $y_i = 1 + \frac{X_i + 1}{4}$ $u(X_i, a, k, m) = \begin{cases} k(X_i - a)^m & X_i > a \\ 0 & -a < X_i < a \\ k(-X_i - a)^m & X_i < -a \end{cases}$	$[-50,50]^n$
6	$F_6(X) = \sum_{i=1}^{11} \left[ a_i - \frac{X_1(b_i^2 + b_i(X_2))}{b_i^2 + b_i(X_3 + X_4)} \right]^2$	$[-5,5]^4$
7	$F_7(X) = - \sum_{i=1}^5 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	$[0,10]^4$
8	$F_8(X) = - \sum_{i=1}^7 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	$[0,10]^4$

Tablo 2. de GSA algoritmasının  $G_0$  yerçekimi sabiti üzerinden incelenmesi verilmiştir. F1 test fonksiyonu için  $G_0$  değerinin 1 ve 10 aralığında sonuç değerinin beş basamaklı büyük değerler oldukları, görülmektedir. 100 değerinde sonuç değerlerinin dört basamaklı sonuçlara düştüğü, 1000 ve 10000 sütunlarında sonuçların üç basamaklı olduğunu ve iki sütun arasında değişikliğin fazla olmadığı tespit edilmiştir. Yine aynı şekilde, F5 için  $G_0$  değerinin 1 ve 10 aralığında sonuç değerinin dokuz basamaklı büyük değerler oldukları, görülmektedir. 100 değerinde sonuç değerlerinin iki basamaklı sonuçlara düştüğü, 10000 sütunlarında sonuçların bir önceki sütuna göre değişikliğin fazla olmadığı tespit edilmiştir. Bu tespitlerden yola çıkılarak  $G_0$  değerinin 100 ile 1000 aralığında en iyi sonuçlar verdiği ve bütün değerler için optimizasyon süresine herhangi bir etkisinin olmadığı sonucu çıkarılmıştır.

GSA Algoritmasında  $\alpha$  parametresinin yerçekimi sabitinin üzerindeki etkisinin incelenmesi Tablo 3. de verilmiştir. Çalışma süreleri yaklaşık benzer olduğu için optimizasyon işlem süresine yer verilmemiştir.

$\alpha$  parametresinin 10 olduğu sütunda en iyi sonuçlar elde edildiği görülmektedir.  $\alpha$  parametresinin çok küçük verilmesi yerçekimi sabitinin her döngüde azalma oranının yavaş olması ile sonuçlandığını ve bununla optimizasyon algoritmasının en iyi sonuçları bulamadığı gözlemlenmiştir. 10 ve 20 tercihlerinde yerçekimi sabitinin 100 değerinden sıfıra doğru hızla azaldığında en iyi sonuçlar elde edilmiştir. Yine farklı olarak  $\alpha$  parametresini 100 ve daha büyük değerler verildiğinde yer çekimi sabitinin her döngüde azalma oranını artırdığı görülmüştür. Ancak bu durum optimizasyon sonuçlarına iyi bir şekilde yansımamıştır. Yani yerçekimi sabitini her döngüdeki azalma oranını artırmak aramanın kötü sonuçlandırılmasına neden olmaktadır.

**Tablo 2.**  $G_0$  yerçekimi sabitinin F1, F5 ve F8 test fonksiyonunda etkisi (The effect of gravity constant F1, F5 and F8 function tests.)

Kullanılan Fonksiyon		$G_0=1$	$G_0=10$	$G_0=100$	$G_0=1000$	$G_0=10000$
F1	En iyi	64899,6	59375,4	2748,9	467,8	423,784
	Geçen Zaman	0,95	0,948	0,951	0,95	0,949
F5	En iyi	5,031E+12	2,16E+12	178629,6	102662,8	114333,5
	Geçen Zaman	0,974	0,985	0,979	0,98	0,982
F8	En iyi	-1,492	-7,374	-8,914	-8,959	-9,222
	Geçen Zaman	0,896	0,889	0,905	0,91	0,896

Tablo 4. de ajan sayısının GSA algoritması üzerindeki etkisinin incelenmesi yapılmıştır. Görüldüğü gibi, küçük ajan sayılarında aramayı düzgün yapmadığı en iyi sonuçların çok büyük değerler olduğu tespit edilmiştir. Bu negatif özelliğe karşın optimizasyon süresinin çok kısa olduğu dikkat çekmiştir. Bu duruma aramanın yerel en küçük değerde takıldığı ifade edilir. Ajan sayısının 30 dan sonra en iyi sonuçları üzerinde bir değişiklik yapmadığı ve yüksek ajan sayısı vermenin bir anlamı olmadığı tespit edilmiştir.

30 ve 100 gibi orta değerli aşan sayılarında kayda değer en iyi sonuçlar elde edilmesinin yanında geçen sürenin normal olduğu söylenebilir. Ajan sayısının 500 gibi büyük değerler verilmesi durumunda ise geçen sürenin çok büyük olduğu görülmektedir. Ajan sayısının çok az verilmesi yerele takılma, çok büyük değerler verilmesi de optimizasyon süresinin çok uzun olması ile sonuçlanmaktadır. 30 ile 100 aralığında değerler vermek, en iyi sonuç ve zaman dengesini sağladığı düşünülmektedir.

Elde edilen Tablo 5. de döngü sayısının GSA algoritması üzerindeki etkisinin incelenmesi yapılmıştır. Buna göre en iyi sonuçların döngü 500 altında elde edildiği, geçen sürenin de bir önceki döngü sayısına göre dört kat arttığı tespit edilmiştir.



**Tablo 3.**  $\alpha$  parametresinin yerçekimi sabiti üzerindeki etkisinin F2 test fonksiyonunda gözlemlenmesi (The effect on the  $\alpha$  parameter is the gravitational constant for test function F2)

$\alpha$ sabitinin GSA algoritmasında etkisinin incelenmesi					
	$\alpha=0.1$	$\alpha=1$	$\alpha=10$	$\alpha=20$	$\alpha=100$
Kul.Fonk.	En İyi	En İyi	En İyi	En İyi	En İyi
F2	10,8687	6,6946	1,0361	7,8784	29,9182
	12,194	5,3318	0,0072	8,1611	21,5727
	11,2834	6,4826	0,4127	6,433	26,3746
	17,5827	6,4696	0,2189	6,0756	30,3449
	12,0754	7,2976	0,5373	4,129	29,4277
	14,574	7,6511	0,2787	7,4551	20,8522
	15,5809	6,3697	0,0873	7,1319	26,4088
	13,4361	7,2839	0,8654	6,9822	27,9537
	11,346	7,1696	0,8355	9,359	28,072
	13,6863	5,9329	0,1353	9,8424	29,6315
	13,263	6,668	0,441	7,345	27,056

**Tablo 4.** Ajan sayısının F3 ve F6 test fonksiyonunda etkisi (Impact on the number of agents for test functions F3 and F6)

Kullanılan Fonksiyon		Ajan=1	Ajan=10	Ajan=30	Ajan=100	Ajan=500
F3	En iyi	2813542	122775,8	3693,3	567,5	64,7
	Geçen Zaman	0,18	0,28	0,742	4,824	102,629
F6	En iyi	368	0,019	0,011	0,002	0,002
	Geçen Zaman	0,17	0,214	0,477	3,097	67,756

**Tablo 5.** F4 ve F7 test fonksiyonu kullanılarak Döngü sayısının etkisi (Effect of the number of cycles using the F4 and F7 function tests)

Kullanılan Fonksiyon		Döngü=1	Döngü=10	Döngü=50	Döngü=100	Döngü=500
F4	En iyi	87,3	75,5	21,3	14,7	1,719
	Geçen Zaman	0,165	0,201	0,368	0,562	2,142
F7	En iyi	-0,399	-1,795	-6,843	-7,901	-8,392
	Geçen Zaman	0,199	0,189	0,32	0,477	1,754

İki fonksiyonda dikkatli incelendiğinde döngü sayısının 100-500 aralığında verilmesinin uygun olacağı kanısına varılmıştır. Kullanıcının geçen zaman parametresini düşünerek uygulamalarda kısa zamanda sonuca ulaşmak isterse 100, geçen sürenin kullanıcı tarafından önemi olmayan uygulamalarda is 500 yaklaşmasının doğru olacağı sonucuna varılmıştır.

Bir diğer çalışmada, Kbest ajanları üzerinedir.Tablo 6 incelendiğinde Kbest kullanılması durumunda optimizasyon süresinin yaklaşık olarak yarıya düştüğünü ve özellikle en iyi sonuçları Kbest kullanılarak elde edildiği görülmektedir. İlgili ajana etkiyen kuvvetleri hesaplarken kütlesi büyük olanları tercih etmek en iyi sonuçlara ulaşma noktasında doğru bir tercihtir.

**Tablo 6.** F1 veF2 test fonksiyonlarında Kbest değerinin etkisi (Effect of the Kbest value using the F4 function tests)

Kbest'inGSA'ya olan ilişkisinin incelenmesi					Kbest'inGSA'ya olan ilişkisinin incelenmesi				
Kul.Fonk.	Her dön.AjanSayısı		Her dön. Kbest		Kul.Fonk.	Her dön.AjanSayısı		Her dön. Kbest	
	En İyi	G. Zaman	En İyi	G. Zaman		En İyi	G. Zaman	En İyi	G. Zaman
F1	1361	0,8811	489	0,5593	F2	0,914	0,9067	2,256	0,5761
	523	0,8936	1023	0,5629		2,223	0,8975	1,143	0,5687
	840	0,8883	1001	0,5589		2,417	0,9033	0,022	0,5760
	849	0,9047	1218	0,5683		2,340	0,8892	0,758	0,5726
	719	0,8927	268	0,5678		1,125	0,8979	0,005	0,5729
	1245	0,8855	1133	0,5809		3,185	0,8983	2,039	0,5665
	1966	0,8867	1202	0,5602		0,333	0,8966	0,440	0,5644
	1264	0,8837	751	0,5697		1,074	0,8903	0,064	0,5670
	863	0,8935	1093	0,5708		0,822	0,8856	0,714	0,5764
	969	0,8887	360	0,5629		2,951	0,8923	1,657	0,5737
	1059,900	0,890	853,800	0,566		1,738	0,896	0,910	0,571

Kbest ajanları uygulamasında final\_per değerinin, GSA algoritmasına etkisinin incelenmesi de son yapılan çalışmadır. Tablo 7. F3 test fonksiyonuna uygulanmış çeşitli final\_per katsayıları karşısında en iyi sonuçları etkilemediği görülmüştür. Optimizasyonun sonucunu büyük oranda etkileyecek bir etkisi görülmemiştir. Ancak algoritma küresel en küçük değere giderken son döngülerde sadece kütlesi çok büyük olan ajanların devreye sokulması, yaklaşım hızı açısından önemlidir.Yaklaşım hızını düşürür, son döngülerde hassas ince arama yapar. Bu yüzden final\_per yüzde değerlerini küçük vermek gerekir.

#### 4. SONUÇ VE TARTIŞMA (CONCLUSION AND DISCUSSION)

$G_0$  değerinin 100 ile 1000 aralığında en iyi sonuçlar verdiği ve bütün değerler için optimizasyon süresine herhangi bir etkisinin olmadığı tespit edilmiştir.  $\alpha$  parametresinin çok küçük veya çok büyük verilmesi optimizasyon sonuçlarına iyi bir şekilde yansımamıştır.  $\alpha$  parametresinin 10 olduğunda en iyi sonuçlar elde edildiği görülmüştür. Ajan sayısı değerlendirmesinde ise 30 ve 100 gibi orta değerli ajan sayılarında kayda değer en iyi sonuçlar elde edilmesinin yanında, geçen sürenin normal olduğu söylenebilir. Ajan sayısının çok az verilmesi yerele takılma, çok büyük değerler verilmesi de optimizasyon süresinin çok uzun olması ile sonuçlanmıştır. Kullanıcının geçen zaman parametresini düşünerek uygulamalarda kısa zamanda sonuca ulaşmak isterse 100, geçen sürenin kullanıcı tarafından önemi olmayan uygulamalarda is 500 yaklaşmasının doğru olacağı sonucuna varılmıştır. İlgili ajana etkiyen kuvvet hesabında, Kbest kullanılması durumunda optimizasyon süresinin yaklaşık olarak yarıya düştüğünü ve özellikle en iyi sonuçları Kbest kullanılarak elde edildiği görülmüştür.

**Tablo 7.** F3 test fonksiyonunun kullanılarak final\_perin etkisi (Effect of final\_per value using the F<sup>3</sup> function test)

Kul.Fonk.	f_p=2	f_p=10	f_p=30	f_p=50	f_p=80
	En İyi	En İyi	En İyi	En İyi	En İyi
F3	6531	3298	1571	5330	2644
	4210	2506	2932	2891	2526
	2003	2852	3806	2276	2940
	2287	2056	1777	2495	1771
	1860	1941	2181	2373	3180
	2469	5364	3313	1018	2732
	2035	2909	1894	2454	2082
	3731	3301	2199	3088	3643
	2091	2150	3613	1996	2939
	2525	2059	2233	4468	1870
		2974,200	2843,600	2551,900	2838,900

Ancak azalma oranının farklı yaklaşımlar uygulayarak incelenmesi ayrı bir çalışma konusu olabilir. Son olarak, final\_per katsayısında, küresel en küçük değere giderken son döngülerde sadece kütlesi çok büyük olan ajanların devreye sokulması, yaklaşım hızı açısından önemlidir. Yaklaşım hızını düşürür, son döngülerde hassas ince arama yapar. Bu yüzden final\_per yüzde değerlerini küçük vermek gerekir.

## 5. KAYNAKLAR (REFERENCES)

- [1]. Guvenc, U., (2010). Combined economic emission dispatch solution using genetic algorithm based on similarity crossover, *Scientific Research and Essays*, 5(17), 2451-2456.
- [2]. Çobanlı, S., Öztürk, A., Güvenç, U., ve Tosun, S., (2010). Active power loss minimization in electric power systems through artificial bee colony algorithm, *International Review of Electrical Engineering*, 5(5), 2217-2223.
- [3]. Öztürk, A., ve Selman A., (2013). Parçacık sürü optimizasyonu ile fotovoltaik sistemlerde kullanılan dc-dc dönüştürücünün kontrolü, *İleri Teknoloji Bilimleri Dergisi*, 110-120.
- [4]. Tosun, S., Öztürk, A., Demir, H. ve Kuru, L., (2012). Kuru tip transformatörün tabu arama algoritması yöntemi ile ağırlık optimizasyonu, *İleri Teknoloji Bilimleri Dergisi*, 2012. 17-26.
- [5]. Duman, S., Maden, D., ve Guvenc, U., (2011). Determination of the PID controller parameters for speed and position control of DC motor using gravitational search algorithm. *In Electrical and Electronics Engineering*, I-225.
- [6]. Rashedi, E., Nezamabadi-pour, H. ve Sarazdi, S., (2009). A gravitational search algorithm, *Information Science*, 179, 2232-2248.
- [7]. Duman, S., Sönmez, Y., Güvenç, U., ve Yörükeren, N., (2012). Optimal reactive power dispatch using a gravitational search algorithm, *IET generation, transmission & distribution*, 6(6), 563-576.
- [8]. Güvenç, U., Sönmez, Y., Duman, S., ve Yörükeren, N., (2012). Combined economic and emission dispatch solution using gravitational search algorithm, *Scientia Iranica*, 19(6), 1754-1762.

- [9]. Bhattacharya, A., ve Roy, P. K., (2012). Solution of multi-objective optimal power flow using gravitational search algorithm, *IET generation, transmission & distribution*, 6(8), 751-763.
- [10]. Duman, S., Güvenç, U., Sönmez, Y., ve Yörükeren, N., (2012). Optimal power flow using gravitational search algorithm, *Energy Conversion and Management*, 59, 86-95.
- [11]. Duman, S., Güvenç, U., ve Yörükeren, N., (2010). Gravitational search algorithm for economic dispatch with valve-point effects, *International Review of Electrical Engineering*, 5(6), 2890-2895.
- [12]. Duman, S., Sonmez, Y., Guvenc, U., ve Yorukeren, N., (2011). Application of gravitational search algorithm for optimal reactive power dispatch problem. *Innovations in Intelligent Systems and Applications (INISTA)*, 519-523.
- [13]. Niknam, T., Narimani, M. R., Azizipanah-abarghooee, R., ve Bahmanifrouzi, B., (2013). Multiobjective Optimal Reactive Power Dispatch and Voltage Control: A New Opposition-Based Self-Adaptive Modified Gravitational, *IEEE Systems Journal*, 7, 1-12.
- [14]. Li, C., Li, H. ve Kou, P. (2013). Piecewise function based gravitational search algorithm and its application on parameter identification of AVR system, *Neurocomputing*, 124, 139-148.
- [15]. Doraghinejad, M., Nezamabadi-pour, H., Hashempoursadeghian, A. ve Maghfoori, M., (2012). A hybrid algorithm based on gravitational search algorithm for unimodal optimization, *2nd International eConference on Computer and Knowledge Engineering (ICCKE)*, 129-132.
- [16]. Rashedi, E., Nezamabadi-pour, H. ve Sarazdi, S., (2009). A gravitational search algorithm, *Information Science*, 179, 2232-2248.
- [17]. Han, X., ve Chang. X., (2012). A chaotic digital secure communication based on a modified gravitational search algorithm filter, *Information Science*, 208, 14-27.
- [18]. Naji, H. R., Sohrabi, M. ve Rashedi, E. (2012). A High-Speed performance optimization algorithm based on a gravitational approach, *Computing in Science & Engineering*, 14, 56-62.
- [19]. Zhang, Z., Li, Y., Xia, Y. ve Luo, F., (2012). Immunity-based gravitational search algorithm, *In 3rd International Conference on Information Computing and Applications (ICICA 2012)*, 754-761.
- [20]. Li, P., ve Duan, H., (2012). Path planning of unmanned aerial vehicle based on improved gravitational search algorithm, *Science China Technological Sciences*, 55, 2712-2719.
- [21]. Nabahari, H., Nikusokhan, M., ve Siarry, P., (2011). Non-dominated sorting gravitatioanal search algorithm, *in International Conference on Swarm Intelligence (ICSI 2011)*, 1-10.
- [22]. Sarafrazi, S., Nezamabadi-pour, H. ve Saryazdi, S., (2011). Disruption: A new operator in gravitational search algorithm, *Scientia Iranica* 18, 539-548.
- [23]. Sombra, F., Valdez, F., Melin, P., ve Castillo. O., (2013). A new gravitational search algorithm using fuzzy logic to parameter adaptation, *in 2013 IEEE Congress on Evolutionary Computation*, 1068-1074.
- [24]. Vijaya K., Vinod Kumar, J. D. M. ve Edukondalu, K., (2012). Strategic bidding using fuzzy adaptive gravitational search algorithm in a pool based electricity market, *Applied Soft Computing*, 13, 2445-2455.