

Geliştirilmiş Yerçekimsel Arama Algoritması: MSS-GSA

*¹Nihan Kazak ve ²Alpaslan Duysak

*¹Mühendislik Fakültesi, Bilgisayar Mühendisliği, Bilecik Şeyh Edebali Üniversitesi, Türkiye

²Mühendislik Fakültesi, Bilgisayar Mühendisliği, Dumlupınar Üniversitesi, Türkiye

Özet

Sezgisel optimizasyon algoritmaları bir çok mühendislik probleminin çözümünde başarılı bir şekilde kullanılmaktadır. Bu algoritmalarından biri de Newton'un evrensel çekim kanunundan esinlenerek geliştirilmiş olan yerçekimsel arama algoritması (YAA) dır. Bu çalışmada MSS-GSA (Mass Spring System – Gravitational Search Algorithm: Kütle Yay Sistemi – Yerçekimsel Arama Algoritması) adı verilen yeni bir algoritma önerilmiştir. Önerdiğimiz metotta nesnelerin deformasyonunda sıkça kullanılan kütle-yay sistemleri incelenerek YAA içine entegre edilmiştir. Bu yeni algoritma arama uzayında hem global hem yerel arama yaparak daha fazla yüzey taraması yapmaktadır. Algoritmanın verimliliğini ölçmek için test fonksiyonları kullanılmıştır. Yapılan testler sonucunda geliştirilen algoritmanın, YAA ile karşılaştırıldığında, daha etkin ve verimli sonuçlar verdiği ancak hesaplama süresi bakımından daha yavaş olduğu gözlenmiştir.

Anahtar Kelimeler: Kütle yay sistemi, MSS-GSA, Sezgisel optimizasyon, Yerçekimsel arama algoritması

Abstract

Heuristic optimization algorithms are used successfully in the solution of many engineering problems. Gravitational search algorithm (GSA) which was inspired by Newton's law of universal gravitation is one of these algorithms. In this study, a new algorithm, MSS-GSA, is proposed. In our method, mass spring system, is used frequently in deformation of the objects, is integrated into the GSA. This new algorithm scans a wider area by both global and local search in search space. Test functions are used to measure effectiveness of the algorithm. The results of the tests demonstrate that the developed algorithm gives more effective and productive results compared to GSA, however it is slower in terms of its computation time.

Key words: Mass spring system, MSS-GSA, Heuristic optimization, Gravitational Search Algorithm

1. Giriş

Metasezgisel yöntemler, çözüm uzayında etkili bir şekilde arama yapmak için farklı yapılarıdaki alt kademe sezgisel algoritmaların zekice birleştirilmesi ile oluşturulmuş iteratif problem çözme yöntemlerdir [1]. Bu yöntemler, her iterasyonda bir çözümden veya çözüm koleksiyonundan yola çıkarak yeni çözümler üretirler. Çoğu metasezgisel yaklaşım, çözüm uzayında stokastik fakat bilinçli bir şekilde arama yapar [2].

*Corresponding author: Address: Faculty of Engineering, Department of Computer Engineering Bilecik Şeyh Edebali University, 11210, Bilecik TURKEY. E-mail address: nihan.kazak@bilecik.edu.tr, Phone: +902282141696

Bu çalışmada, optimizasyon yöntemlerinden biri olan YAA' dan [3] esinlenerek gerçek değerlere daha yakın çözümler elde edilebilmek amacıyla MSS-GSA (Kütle Yay Sistemi – Yerçekimsel Arama Algoritması) algoritması geliştirilmiştir. Geliştirilen algoritmanın etkinliğini ölçmek için bazı test fonksiyonları kullanılmıştır. Elde edilen sonuçlar YAA kullanılarak elde edilen sonuçlarla karşılaştırılmıştır. Bu çalışmanın ikinci bölümünde, kütle yay sistemi hakkında bilgi verilmiş ve yerçekimine dayalı YAA incelenerek akış şeması verilmiştir. Üçüncü bölümde geliştirilen MSS-GSA' nın adımları açıklanmıştır. Dördüncü bölümde algoritmaların test sonuçları sunulmuş ve son bölümde de yorumlanmıştır.

2. Materyal ve Yöntem

Optimizasyon yöntemlerinde arama uzayına atanan objeler birbirinden bağımsız bir şekilde optimum noktayı ararlar. Birbirine bağlı olan objeler kullanıldığında optimum noktaya daha yakın bir çözüm elde etmek mümkündür. Bu çalışmada, objeleri birbirine bağlamak amacıyla kütle-yay sistemi prensibi, YAA' nın çalışma prensibiyle birleştirilerek yeni bir algoritma geliştirilmiştir.

2.1. Kütle-Yay Sistemi

Bir nesnenin 3 boyutlu modellemesi yapılırken n adet düğüm ve bu düğümleri anlamlı olarak birbirine bağlayan m adet bağlantı kullanılır. Kütle yay sisteminde nesneyi oluşturan her düğüme noktasal kütle, iki noktasal kütle arasındaki bağlantıya ise yay adı verilir. Kütle-yay sistemi, ağda mevcut olan kütle ve yayların birbirleriyle ve kendi içindeki değişimlerini inceleyen bir yöntemdir.

Yaya bağlı noktasal kütlelerden herhangi birinde yer değiştirme söz konusu olduğunda yay dinamiği süreci başlar [4]. Kütleli noktada olan herhangi bir değişim ya da dışarıdan bir kuvvet, yayın içsel bir kuvvet oluşturmasına neden olur. Bu kuvvet, Hooke kanununa göre aşağıdaki eşitlikte gösterilmiştir:

$$f = -k_s \times (|d| - s) \times \frac{d}{|d|} \quad (1)$$

(1)'de s , yayın ilk uzunluğudur, $|d|$ ise yayın işlem yapılacağı andaki uzunluğunu belirtir. k_s ile gösterilen değer ise yayın sertliğini gösteren yay sabitidir. Eşitliğin son kısmında bulunan $\frac{d}{|d|}$ birim vektörü ifade eder. Dış kuvvetler veya yayların dinamiği sonucunda kütle dinamiği süreci başlamış olur. Kütleye etki eden toplam kuvvet belirlendikten sonra Newton'un ikinci kanununa göre ivme hesaplanır.

$$f = m \times a \quad (2)$$

İvmesi bilinen kütlelerin hızı ve yeni konumu Explicit Euler integrasyonu ile belirlenir [5]. Kütlelerin bir sonraki hızı (3) ile hesaplanır. Yeni konumu (4) ise önceki konumu ile yer

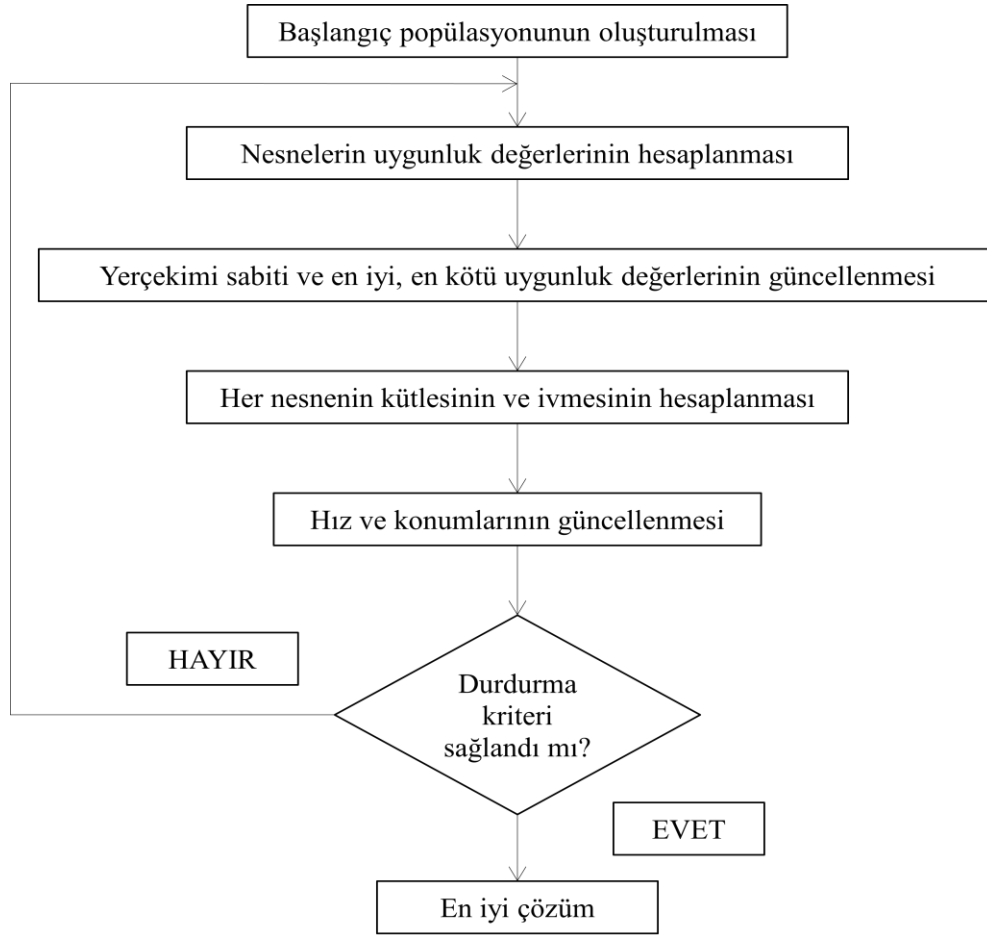
değiştirilmesi toplanarak elde edilir.

$$V(t + \Delta t) = V(t) + \Delta t \times \frac{F(t)}{m} \quad (3)$$

$$X(t + \Delta t) = X(t) + \Delta t \times V(t + \Delta t) \quad (4)$$

2.2. Yerçekimsel Arama Algoritması

YAA, Newton' un hareket kanunlarından ikincisi olan ivme kanunu ve yine Newton' un evrensel çekim kanunu temel alınarak tasarlanmış metasezgisel optimizasyon yöntemlerinden biridir. YAA' da her bir nesnenin kütle miktarı o nesnenin performansını gösterir. Her bir kütle, arama uzayında olan diğer kütleleri yerçekimi kuvveti ile çeker. Böylelikle kütleler arası etkileşim sağlanmış olur [6]. Bu kuvvet bütün kütlelerin en ağır olan kütleyle doğru hareket etmesini sağlar. Bundan dolayı da kütleler yerçekimi kuvveti doğrultusunda birlikte hareket ederler. Söz konusu algoritmanın akış şeması Şekil 1' de verilmiştir.



Şekil 1. YAA' nın akış şeması

Algoritmada iterasyon boyunca en ağır olan kütle diğer kütlelere nazaran daha yavaş hareket edecek ve diğerlerini kendine çekecektir [8]. İterasyon sayısı bitiminde ya da herhangi bir sonlandırma eylemi olduğunda kütlesi en fazla olan nesne, problemin optimum çözümünü oluşturmuş olacaktır.

3. MSS-GSA

Önerilen algoritmada kütle yay sisteminin çalışma prensibi ele alınarak, optimum noktayı arama amaçlanmıştır. Bu nedenle, YAA' da kullanılan kütlelerin arasına yay bağlanmıştır. Kütlelerin konum farkından dolayı potansiyel farktan doğan dış kuvvet ile bağlı olduğu yay tarafından oluşan iç kuvvet toplanarak bir üyeye uygulanan toplam kuvvet hesaplanır. Böylelikle arama uzayında hem global hem de yerel arama hedeflenir. Kütlelerin arama uzayındaki hareketi ise Newton'un ivme yasasına göre gerçekleşir.

Algoritmada ilk olarak arama uzayının sınırları belirlenir. Oluşturulan uzaya, üyeler rastgele konumlandırılır.

$$X_i = (x_i^1, \dots, x_i^d, \dots, x_i^n) \quad i = 1, 2, \dots, N \quad (5)$$

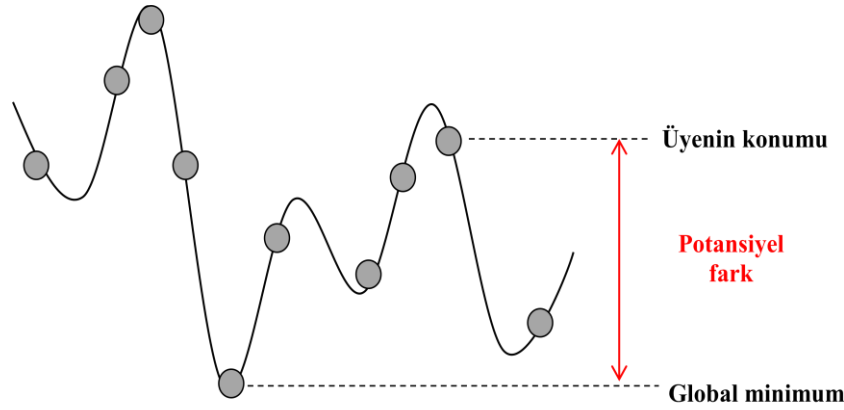
Burada x_i^d , arama uzayındaki i . üyenin d . boyuttaki konumunu belirtir. Üyeler tanımlandıktan sonra uygunluk fonksiyonuna gönderilerek her birinin uygunluk değerleri hesaplanır. Tüm üyelerin uygunluk değerlerine bakıldığında, problem minimizasyon problemiye en küçük uygunluk değerine sahip üye, maksimizasyon problemiye en büyük uygunluk değerine sahip üye, tüm üyeler içinde en iyi olanıdır. Optimum çözüm aranırken, arama hızının kontrol edilmesi gerekmektedir. Yerçekimi sabiti (6) ve yay sabiti (7) her iterasyonda azaltılarak yay sertliği ve arama hızı kontrolü sağlanır.

$$G(t) = G_0 \times e^{(-\frac{\alpha t}{T})} \quad (6)$$

$$K(t) = k_0 \times e^{(-\frac{\alpha t}{T})} \quad (7)$$

(6) eşitliğinde, t anındaki yerçekimi sabiti $G(t)$, $e^{(-\frac{\alpha t}{T})}$ ile yerçekimi sabitine verilen ilk değer G_0 çarpımıdır, (7)' de t anındaki yay sabiti $K(t)$, yay sabitinin başlangıç değeri k_0 ' dir. Eşitliklerde kullanılan T , toplam iterasyon sayısını; t , o andaki iterasyon sayısını; α ise sabit bir değeri ifade eder.

Global ve yerel minimumlara sahip bir minimizasyon probleminde Şekil 2'deki gibi, üyeler arama uzayına rastgele dağıtıldıklarında uygunluk değerlerine göre global minimuma daha yakın ya da uzak mesafede bulunurlar.



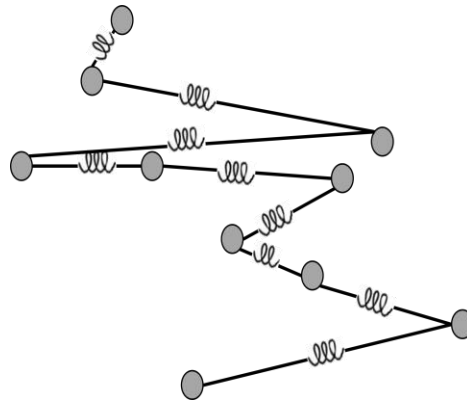
Şekil 2. Üyeler arasındaki potansiyel fark gösterimi

Üyelerin uygunluk değerlerindeki farklılık nedeniyle, arama uzayındaki konumlarının oluşturduğu potansiyel farktan dolayı bir dış kuvvet oluşur.

$$F_{ext_{ij}^d}(t) = G(t) \times \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \times \left(\frac{X_j^d(t) - X_i^d(t)}{|X_j^d(t) - X_i^d(t)| + \epsilon} \right) \quad (8)$$

$$F_{ext_i^d}(t) = \sum_{j \neq i}^N rand_j F_{ext_{ij}^d}(t) \quad (9)$$

(8)' de, t anında j üyesinin i üyesi üzerinde oluşturduğu kuvvet hesaplanmıştır. $G(t)$ yerçekimi sabitini, $fit_i(t)$ t anında i . üyenin uygunluk değerini, $best(t)$ ve $worst(t)$ ise yine o andaki en iyi ve en kötü uygunluk değerlerine sahip çözümleri, $|X_j^d(t) - X_i^d(t)|$, i ve j üyelerinin d . boyuttaki birbirlerine olan uzaklıklarını; ϵ , küçük sabit bir değeri ve $(X_j^d(t) - X_i^d(t))$, ifadesi j . üye ile i . üyenin d . boyuttaki konum farklarını belirtir. Uygunluk değerleri belirlenen üyeler en iyiden en kötüye doğru sıralanır. Sıralanan üyeler arasına yay bağlanarak Şekil 3' deki gibi bir kütle yay sistemi oluşturulur.



Şekil 3. En iyiden en kötüye doğru sıralanan üyeler arasına yay bağlanması

Algoritmada üyeler arasındaki potansiyel farkı sıfırlamak için kütle yay sisteminde kullanılan denklem (10) uygulanır.

$$F_int_{ij}^d(t) = K \times \frac{fit_j(t) - fit_i(t)}{best(t) - worst(t)} \times \left(\frac{X_j^d(t) - X_i^d(t)}{|X_j^d(t) - X_i^d(t)| + \varepsilon} \right) \quad (10)$$

Yay, kendini geren ya da sıkıştıran kuvvete eşit büyüklükte ve ters yönde (11)' deki gibi bir kuvvet uygular.

$$F_int_{ji}^d(t) = -F_int_{ij}^d(t) \quad (11)$$

Eşitlik (10) ve (11)' de j . üye, i . üyenin yay ile bağlı olduğu bir sonraki üyedir. (10) eşitliğinde, iterasyon t ' de d . boyutta j . üyenin i . üye üzerine uyguladığı kuvvet $F_int_{ij}^d(t)$, t anında i . üyenin uygunluk değeri; $fit_i(t)$, j . üyenin uygunluk değeri; $fit_j(t)$, t anındaki en iyi ve en kötü değerler $best(t)$ ve $worst(t)$, j . ve i . üyelerin d . boyuttaki konum farkı $X_j^d(t) - X_i^d(t)$ ve aralarındaki mesafe $X_j^d(t) - X_i^d(t)$ olarak gösterilmiştir. Eşitlikte kullanılan ε küçük bir sabit değerdir. Üyeler arasındaki yayın katsayısı ise K ile gösterilmiştir. Bu adımlardan sonra, üyeler en iyi üyeye doğru bir çekim içerisine girer. Bu sayede, arama uzayı içerisinde yerel arama işlemi de gerçekleşmiş olur. Son olarak bir üyenin diğer üyelere uyguladığı iç ve dış kuvvetler toplanarak toplam kuvvet hesaplanır.

$$F_i^d(t) = F_{int_i}^d(t) + F_{ext_i}^d(t) \quad (12)$$

Eşitlik (12)' de, $F_{int_i}^d(t)$ t anında d . boyuttaki i . üye üzerinde yay tarafından oluşan iç kuvveti, $F_{ext_i}^d(t)$ t anında d . boyuttaki i . üye üzerinde oluşan dış kuvveti ve $F_i^d(t)$ t anında d . boyutta i . üye üzerinde oluşan toplam kuvveti gösterir.

Algoritmanın geri kalan kısmı yani ivmenin, hızın ve yer değiştirmenin hesaplanması YAA' da olduğu gibi ilerler. Durdurma kriterine rastlanmadığı takdirde üyelerin uygunluk değerlerinin hesaplandığı adıma geri dönülerek algoritma devam ettirilir.

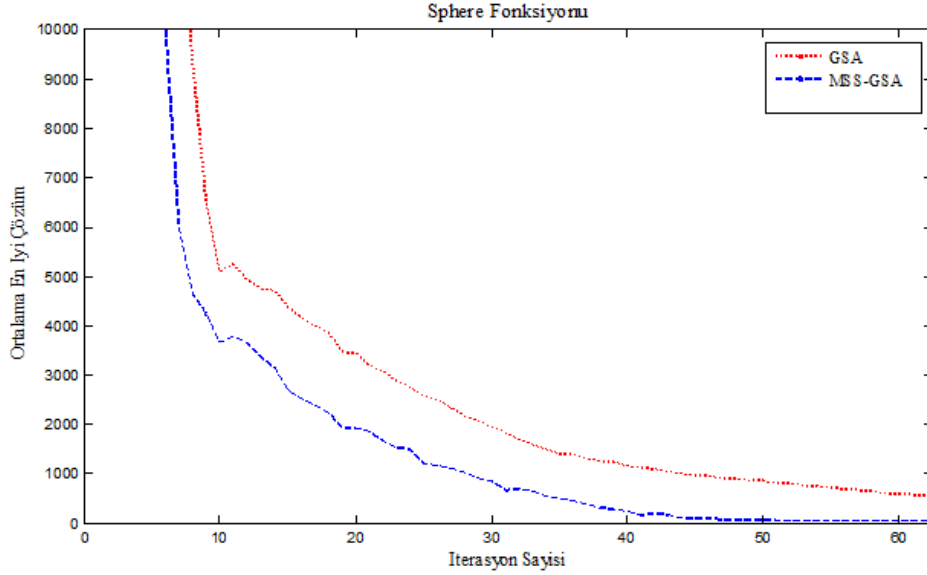
4. Algoritmanın Testi

MSS-GSA algoritmasının performansını test edebilmek amacıyla üç adet test fonksiyonu kullanılarak YAA ile karşılaştırması yapılmıştır. Algoritmaların kodları ve fonksiyonlar MATLAB programlama dilinde yazılmış ve Intel(R) Core(TM) i5 CPU M 430 @ 2.27 GHz, 4GB RAM donanımlı, 32 bit işletim sistemine ait bir dizüstü bilgisayarda çalıştırılmıştır.

Test fonksiyonlarından ilki olan Sphere fonksiyonu, tek bir global minimuma sahiptir. Sphere fonksiyonunun ayrıca düzgün ve ileri derecede konveks bir fonksiyon olma özelliği vardır.

$$F_{Sphere}(X) = \sum_{i=1}^n x_i^2 \quad (13)$$

(13)' de formülizasyonu verilen Sphere fonksiyonunun global minimum noktası 0'a eşittir. Eşitlikte fonksiyonun n boyutlu olduğu görülmektedir. MSS-GSA ve YAA' da n 30 alınmıştır. Toplam üye sayısı 25, kütlesi 10, iterasyon sayısı 500, değişken sınırları $[-100,100]$, $\epsilon = 10^{-12}$, $G_0 = 50$, $\alpha = 10$, yay sabitinin başlangıç değeri 50 olarak alınmış ve eksponansiyel olarak azaltılmıştır. Elde edilen sonuçlar Şekil 4' de gösterilmiştir.

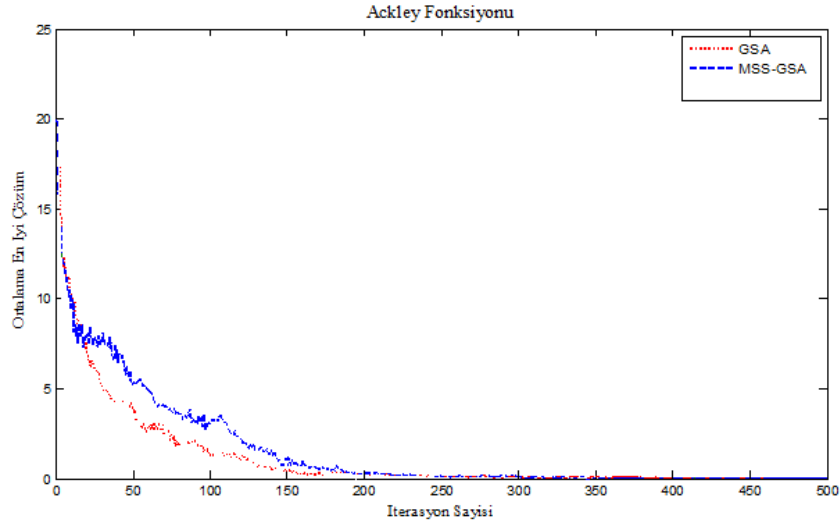


Şekil 4. Sphere fonksiyonu için MSS-GSA ve YAA' nın ürettiği çözümler

İkinci olarak kullanılan test fonksiyonu, Ackley fonksiyonudur. Fonksiyon, birçok yerel optimuma ancak tek bir global optimuma sahiptir.

$$F_{Ackley}(X) = -20 \times \exp\left(-0.2 \times \sqrt{\frac{\sum_{i=1}^n x_i^2}{n}}\right) - \exp\left(\frac{\sum_{i=1}^n \cos(2 \times \pi \times x_i)}{n}\right) + 20 + e \quad (14)$$

(14)' de verilen Ackley fonksiyonunun global minimum noktası 0' a eşittir. Uygulanan algoritmalarda fonksiyon boyutu 30, diğer değerler Sphere fonksiyonunda alınan değerlere eşit alınmıştır. Elde edilen sonuçlar Şekil 5' de gösterilmiştir.

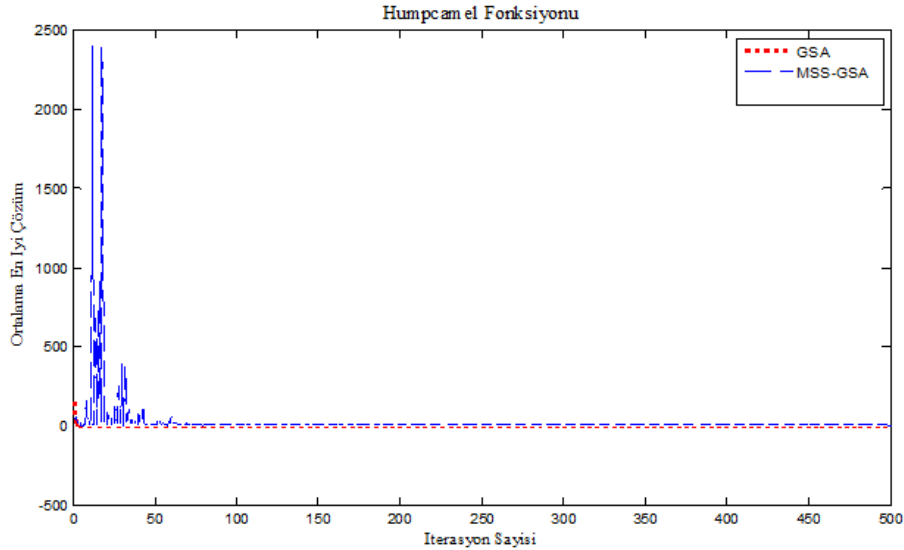


Şekil 5. Ackley fonksiyonu için MSS-GSA ve YAA' nın ürettiği çözümler

Son olarak; önerdiğimiz algoritma, iki global optimumla birlikte toplamda 6 adet yerel optimuma sahip çok modlu bir fonksiyon olan Humpcamel fonksiyonu (15) ile test edilmiştir.

$$F_{Humpcamel}(X) = 4 \times x_1^2 - 2.1 \times x_1^4 + \frac{x_1^6}{3} + x_1 \times x_2 - 4 \times x_2^2 + 4 \times x_2^4 \quad (15)$$

Fonksiyonun global minimum noktası -1.0316' dır. Kullanılan algoritmalarda toplam üye sayısı 25, maksimum iterasyon sayısı 500, değişken sınırları $[-5,5]$, $\varepsilon = 10^{-12}$, yerçekimi sabiti hesabında $\alpha = 10$, $G_0 = 50$, yay sabitinin başlangıç değeri 50 olarak alınmıştır. Elde edilen sonuçlar Şekil 6' da gösterilmiştir.



Şekil 6. Humpcamel fonksiyonu için MSS-GSA ve YAA' nın ürettiği çözümler

Sonuçlar

Bu çalışmada, optimizasyon problemlerinin çözümü için Rashedi ve ark. tarafından 2009 yılında önerilen YAA' dan esinlenerek MSS-GSA adı verilen bir optimizasyon yöntemi geliştirilmiştir. Algoritmada arama uzayındaki kütleler iki farklı kuvvete maruz kalmaktadır. Bunlardan ilki arama uzayındaki konumlarından dolayı birbirlerine oluşturdukları potansiyel farktan oluşan kuvvettir. Diğer kuvvet ise kütleler arasına bağlanan yaydan dolayı oluşan kuvvettir. Oluşan kütle yay sistemi ile kütleler birbirlerine kuvvet uygulayarak en iyi çözüme doğru çekilirler. Böylelikle MSS-GSA algoritmasında hem global hem yerel arama gerçekleştirilmiş olunur. Yapılan testler sonucunda, önerilen algoritmanın YAA' ya göre daha iyi çözümler elde ettiği, fakat hesaplama süresi açısından YAA' nın daha iyi olduğu görülmektedir. Daha sonraki çalışmalarda algoritmanın daha kısa sürede çözümlere ulaşabilmesi amaçlanacaktır.

Kaynaklar

- [1] Karaboğa D. Yapay Zeka Optimizasyon Algoritmaları. Atlas Yayın Dağıtım, İstanbul, 2004.
- [2] Blum C ve Roli A. Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. ACM Computing Surveys, 2003, 35(3):268-308.
- [3] Rashedi E, Nezamabadi H ve Saryazdi S. GSA: A Gravitational Search Algorithm. Information Sciences, 2009, 179(13):2232–2248.
- [4] Duysak A Zhang J J. Fast Simulation of Facial Tissue Deformations Using Mass- Spring Chain Algorithm, Theory and Practice of Computer Graphics. The 23rd Conference organized by the UK chapter of the Euro Graphics Association, 2005, 139-145.
- [5] Provot X. Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior. Proc. Graphics Interface '95, 1995, 147-154.
- [6] Kazak N ve Duysak A. Modified Gravitational Search Algorithm. International Symposium on Innovations in Intelligent Systems and Applications (INISTA), 2012, pp.1-4.
- [7] Chatterjee A ve Mahanti G K. Comparative Performance Of Gravitational Search Algorithm And Modified Particle Swarm Optimization Algorithm For Synthesis Of Thinned Scanned Concentric Ring Array Antenna. Progress In Electromagnetics Research B, 2010, 25: 331-348.
- [8] Taghipour M, Moradi A R ve Yazdani-Asrami M. GSA Trained ANN for Educational Purposes. 2010 IEEE Conference on Open Systems (ICOS 2010), 2010, Kuala Lumpur, Malaysia.