

Geliştirilmiş Adaptif Sinüs Kosinüs Algoritması (ASKA)'nın Optimizasyon Problemlerinin Çözümünde Kullanılması

Erkan TANYILDIZI, Gökhan DEMİR

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü, Elazığ
etanyildizi@gmail.com

(Geliş/Received: 17.08.2017; Kabul/Accepted: 10.03.2018)

Özet

Bu çalışmada; optimizasyon problemlerinin çözümü için yeni bir popülasyon tabanlı optimizasyon algoritması olan Sinüs Kosinüs Algoritması (SKA)'nın geliştirilmiş yeni bir versiyonu verilmiştir. Matematiksel tabanlı olan SKA, sinüs ve kosinüs fonksiyonları kullanılarak oluşturulmuş matematiksel model ile arama uzayında içe veya dışa doğru hareket ederek arama uzayının keşfini ve sömürülmesini garanti ederek en iyi çözümü bulmaya çalışır. SKA'nın performansını arttırmak için tanımlı olan rast gele değişkenlerin algoritmadaki yeri ve katsayılarının değiştirilmesi ile evrimsel olarak daha uyumlu kılan bir özellik eklenmiştir. Geliştirilmiş yeni optimizasyon algoritması Adaptif Sinüs Kosinüs Algoritması (ASKA) olarak tanımlanır. ASKA'nın performansını değerlendirmek için literatürde yaygın olarak kullanılan kısıtsız kalite testi fonksiyonları üzerinde testler yapılmıştır. Ayrıca kısıtlı problemler üzerindeki etkinliği test etmek için mühendislik tasarım problemlerinden biri olan basınçlı kap probleminin çözümünde kullanılmıştır. Wilcoxon işaretli sıralar testi yapılarak ASKA'nın karşılaştırılan diğer metasezgisel algoritmalara göre performansı incelenmiştir.

Anahtar Kelimeler: Adaptif Sinüs Kosinüs Algoritması, Optimizasyon, Metasezgisel Algoritmalar.

The Use of Improved Adaptive Sine Cosine Algorithm (ASKA) in the Solution of Optimization Problems

Abstract

In this study; a new version of the Sinus Cosine Algorithm (SKA), a new population-based optimization algorithm for solving optimization problems, is presented. SKA, mathematically based, tries to find the best solution by guaranteeing exploration and exploitation of search space by moving inward or outward in search space with a mathematical model created using sine and cosine functions. A feature that is evolutionarily more compatible with the location and coefficients of the random variables defined in the algorithm to improve the performance of the SKA is added. The new optimized optimization algorithm is called the Adaptive Sine Cosine Algorithm (ASKA). Unrestricted quality testing functions commonly used in the literature have been tested to evaluate the performance of ASKA. It was also used to solve the problem of the pressure vessel, one of the engineering design problems, to test the efficiency on constrained problems. Wilcoxon signed rank tests were performed to evaluate the performance of ASKA relative to other comparative meta-static algorithms.

Keywords: Adaptive Sine Cosine Algorithm, Optimization, Metaheuristic Algorithms.

1. Giriş

Metasezgisel algoritmalar, gerçek yöntemlerin makul bir hesaplama süresi içinde optimum çözüme erişemeyeceği, özellikle de çok sayıda yerel minimum ile çevrelenen global minimum bulunduğu, gerçek hayat problemlerine yakın optimal çözümler bulma potansiyellerini kanıtlamıştır. Bu nedenden ötürü metasezgisel algoritmalar, gerçek mühendislik problemlerinin global optimumunu bulması için oldukça sık kullanılmaktadır [2]. Gerçek dünya

mühendislik optimizasyonu problemlerinin çoğu çok karmaşıktır ve çözülmesi oldukça zordur.

Mevcut sayısal yöntemlerin basitlik, etkinlik ve doğruluk gibi dezavantajları, araştırmacıları, mühendislik optimizasyon problemlerini çözmek için doğadan veya farklı bilim dallarından ilham alan yöntemlere dayalı metasezgisel algoritmalara güvenmeye teşvik etmektedir.

Mühendislik probleminde birden fazla lokal optimum varsa, başlangıç noktasının seçimine bağlı olarak elde edilen en uygun çözüm global optimum olmayabilir [2]. Genel amaçlı sezgisel

yöntemlerin yaygın olarak kullanılmasındaki başlıca nedenler, lokal optimumdan kaçınma, kod basitliği, gerçekleştirilebilirlik, esneklik, sağlamlık, basitlik, analiz edilebilirlik, ve türetilbilmedir [3].

Metasezgisel algoritmalar, genellikle doğal fenomeni taklit eden kuralları ve rasgeleliği birleştirerek en doğru çözümü bulmaya çalışırlar [4]. Metasezgisel algoritmaların bir diğer avantajı ise, probleme bağımlı olmamasıdır. Bu yüzden kısıtlı veya kısıtsız her türlü problemleri çözmek için kullanılabilen genel amaçlı yöntemlerdir. Buna karşılık sonuçların doğruluğundan minimum hata değeri kadar taviz verilebilir [5]. Genel amaçlı metasezgisel yöntemler genellikle doğadan ilham alınarak gerçekleştirilmiştir. Ayrıca farklı bilim dallarından veya bu bilim dallarının birleşiminden ilham alınarak gerçekleştirilmiş olan biyoloji, fizik, kimya, matematik ve sosyal tabanlı yöntemler mevcuttur. Doğadan ilham alan algoritmaların büyük bölümü biyoloji tabanlıdır. Biyolojik sistemin, doğal seçim süreci, biyolojik canlıların coğrafik dağılımı, yayılım, istila ve insanların yaratıcı bir şekilde problem çözme süreci, sürü zekâsına bağlı olarak besin arama, avlanma, göç ve üreme gibi bazı başarılı özelliklerinden ilham almaktadır. Fizik ve kimya biliminden ilham alınarak geliştirilen algoritmalar, elektrik yükleri, kara delik olgusu, yerçekimi, nehir sistemleri, mükemmel armoni vb. içeren belli fiziksel ve kimyasal kanunları taklit etmişlerdir. Matematik tabanlı metasezgisel yöntemler, matematiksel programlama tekniklerinin kullanılmasıyla geliştirilen sezgisel algoritmalarlardır. Sosyal tabanlı metasezgisel algoritmalar ise insanların sosyal ve rekabetçi davranışlarından ilham alınarak geliştirilmeleridir [1,6].

Literatüre kazandırılmış çok başarılı algoritmalar ve teknikler geliştirilmiş olsa da; bilimsel alanda sürekli iyileşme ve daima daha iyiyi arama felsefesi altında yeni tekniklerin tasarlanması, geliştirilmesi ve uygulanması önemli bir görevdir. Bu amaç doğrultusunda bu çalışmada kapsamında SKA [7] üzerinde iyileştirmeler yapılarak Adaptif Sinüs Kosinüs Algoritması (ASKA) önerilmektedir. Önerilen algoritmanın başarısını test etmek için literatürde yaygın olarak kullanılan kısıtsız yirmi üç kıyaslama fonksiyonuna ve kısıtlı olan

mühendislik problemlerinden basınçlı kap problemine uygulanmıştır. Ayrıca önerilen algoritmadan elde edilen sonuçlar literatürdeki bazı optimizasyon algoritmaları ile karşılaştırılmıştır.

2. Adaptif Sinüs Kosinüs Algoritması (SKA)

Sinüs Kosinüs Algoritması'nda her arama ajanının her boyutu için r_2 , r_3 ve r_4 parametrelerinin rastgele olarak yeniden belirlenmesi hedefe ulaşmada sapsmalara neden olmaktadır [7]. Ayrıca her boyut bu parametrelerin yeniden belirlenmesi problemlerin çözümünde geçen süreyi artırmaktadır. Bu nedenle önerilen yeni yöntemde r_2 , r_3 ve r_4 parametrelerinin her arama ajanı için belirlenip tüm boyutları için sabit tutularak hem hedeften sapsmaların azaltılması hem de performansta iyileştirmelerin sağlanması gerçekleştirilmiştir.

Şekil 1'de sözde kodu verilen SKA üzerinde iyileştirmeler yapılarak geliştirilen Adaptif Sinüs Kosinüs Algoritması (ASKA)'nın sözde kodu Şekil 3'de gösterilmektedir.

```

1. Başlangıç popülasyonunu her boyut için arama ajanı sayısı kadar düzgün dağılıma bağlı olarak rastgele oluştur
2. Arama ajanlarının uygunluğunu hesapla
3. En iyi arama ajanını bul ve hedef değer olarak ata
4. while maksimum iterasyon sayısı
5.    $t = 2, c = 2$ 
6.    $r_1 = c - t * (c / \text{Maksimum iterasyon})$ 
7.   for arama ajanı sayısı
8.     for boyut sayısı
9.        $r_2 \leftarrow 2\pi * \text{rand}, r_3 \leftarrow 2 * \text{rand}, r_4 \leftarrow \text{rand}$ 
10.      if  $r_4 < 0$ 
11.         $X(i, j) \leftarrow X(i, j) + (r_1 * \sin(r_2) * |r_3 * P(j) - X(i, j)|)$ 
12.      else
13.         $X(i, j) \leftarrow X(i, j) + (r_1 * \sin(r_2) * |r_3 * P(j) - X(i, j)|)$ 
14.      end if
15.    end for
16.  end for
17. En iyi çözümü (arama ajanı) bul ve  $P(j)$ 'ye hedef değer olarak ata
18. end while
19. return en iyi çözüm kümesi ve elde edilen global optimum sonuç

```

Şekil 1. Sinüs Kosinüs Algoritması sözde kodu

SKA' da keşif ve sömürü için önerilen pozisyon güncelleme denklemleri Denklem 1' de,

pozisyon üzerindeki etkileri ise Şekil 2 de verilmiştir.

$$X_i^{t+1} = \begin{cases} x_i^t + r_1 \times \sin(r_2) \times |r_3 P_i^t - X_i^t|, r_4 < 0.5 \\ x_i^t + r_1 \times \cos(r_2) \times |r_3 P_i^t - X_i^t|, r_4 \geq 0.5 \end{cases} \quad (1)$$

x_i^t : i . boyutun t . iterasyonundaki güncel çözümü.

r_1, r_2, r_3 : Rastgele sayılar.

r_4 : $[0, 1]$ aralığında rastgele bir sayıdır.

P_i : i . boyuttaki hedef noktanın pozisyonu.

$||$: Mutlak değer.

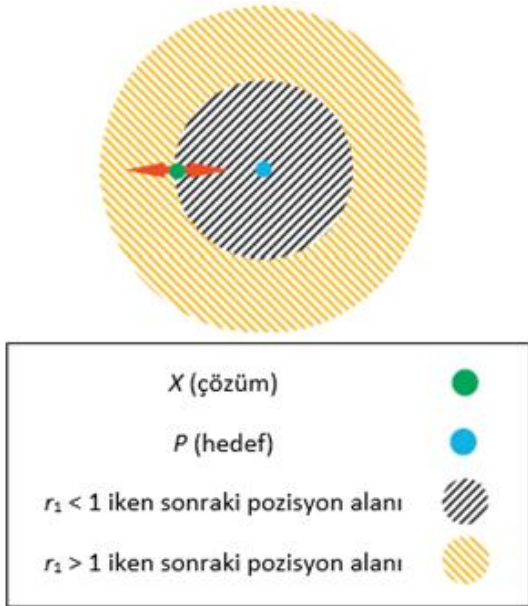
Yukarıdaki denklemlerde görüleceği üzere SKA'da; r_1, r_2, r_3, r_4 olmak üzere 4 ana parametre kullanılır.

r_1 : Bir sonraki pozisyon bölgesini (veya hareket yönü) belirler.

r_2 : Hedefe ulaşmak için, içe doğru ya da dışa doğru ne kadar hareket edileceğini belirler.

r_3 : Stokastik ağırlığı rastgele belirler. $r_3 > 1$ olması stokastikliğin önemli olduğunu, $r_3 < 1$ olması ise stokastikliğin daha az etkili olduğunu belirtir.

r_4 : Denklemdaki sinüs ve kosinüs bileşenleri arasındaki geçişi sağlar.



Şekil 2. Denklem 1 ve Denklem 2'deki sinüs ve kosinüsün bir sonraki pozisyon üzerindeki etkileri [7]

```

1. Başlangıç popülasyonunu her boyut için arama
   ajanı sayısı kadar düzgün dağılıma bağlı olarak
   rastgele oluştur
2. Arama ajanlarının uygunluğunu hesapla
3. En iyi arama ajanını bul ve hedef değer olarak ata
4. while maksimum iterasyon sayısı
5.   t = 2, c = 2
6.   r1 = c - t* ( c / Maksimum iterasyon)
7.   for arama ajanı sayısı
8.     r2 ← 2π * rand, r3 ← 2 * rand, r4 ← rand
9.     for boyut sayısı
10.      if r4 < 0
11.        X(i, j) ← X(i, j) + (r1 * sin(r2) * |r3 * P(j) - X(i, j)|)
12.      else
13.        X(i, j) ← X(i, j) + (r1 * cos(r2) * |r3 * P(j) - X(i, j)|)
14.      end if
15.    end for
16.  end for
17. En iyi çözümü (arama ajanı) bul ve P(j)'ye hedef
   değer olarak ata
18. end while
19. return en iyi çözüm kümesi ve elde edilen global
   optimum sonuç

```

Şekil 3. Adaptif Sinüs Kosinüs Algoritması sözde kodu

3. Deneysel ve Sonuçlar

ASKA iyi bilinen sürü tabanlı algoritmalarından Parçacık Sürü Optimizasyonu[8] (PSO), Karınca Koloni Optimizasyonu (KKO)[9] Algoritması ve güncel algoritmalar Balina Optimizasyon Algoritması (BOA)[10], Yerçekimi Arama Algoritması (YAA)[11], Sinüs Kosinüs Algoritması (SKA)[7] olmak üzere 5 metasezgisel algoritma ile karşılaştırılmıştır. Karşılaştırılan algoritmalarda kullanılan parametreler şu şekildedir:

1. PSO: Atalet ağırlığı = 1, Atalet ağırlığı sönüm oranı = 0.99, Kişisel öğrenme katsayısı = 1.5, Küresel öğrenme katsayısı = 2.0
2. KKO: Örnek boyutu = 40, Yoğunlaşma faktörü = 0.5, Sapma - uzaklık oranı=1
3. BOA: b = 1
4. YAA: R_norm = 2, R_gücü = 1, Elitist kontrolü = 1
5. SKA: t = 2, c = 2

Algoritmaların popülasyon büyüklüğü 30 ve iterasyon sayıları 1000 olarak kabul edilmiştir. Boyutları fonksiyon tanımlarında Vno olarak belirtilen her kalite testi fonksiyonu için algoritmalar 30 kez çalıştırılmıştır. Tablo 1'de gösterilen F1 - F7 arasındaki fonksiyonlar tek modlu fonksiyonlardır ve bir tek global optimuma sahiptirler. Bu fonksiyonlar arama

algoritmalarının yakınsama oranını test etmek için tasarlanmıştır. Birden fazla lokal minimuma sahip olan ve bundan dolayı optimize edilmesi oldukça zor olan F8 – F13 arasındaki çok modlu fonksiyonlar Tablo 2’de gösterilmiştir. Çok modlu fonksiyonlarda problem boyutu sayısı arttıkça yerel optimum sayısı da artmaktadır. Bu nedenle bu tür test problemleri optimizasyon algoritmalarının arama kapasitelerini değerlendirmede oldukça önemlidir. Tablo 3’te gösterilen F14 – F23 arasındaki sabit boyutlu çok modlu fonksiyonların çok modlu fonksiyonlardan tek farkı boyutlarının düşük sayıda olmasından

dolayı az sayıda yerel minimum içermeleridir. Tablo 1 – 3 te verilen yirmi üç kıyaslama fonksiyonlarına ait sonuçlar sırasıyla Tablo 4-6 da verilmiştir. Tablolarda verilen kısaltmalar; *F*: fonksiyon adı, *İ*: istatistik sonuçları *ort*: 30 çalıştırmadaki ortalama çözüm, *sd*: 30 çalıştırmadaki standart sapmayı, *en iyi*: 30 çalıştırmadaki en iyi çözümü, *en kötü*: 30 çalıştırmadaki en kötü çözümü, *süre*: saniye cinsinden ortalama çalışma süresini, ifade etmektedir.

Tablo 1. Tek modlu kalite testi fonksiyonlarının tanımı

Fonksiyon	Formülasyon	V_{no}	Aralık	F_{min}
<i>Hyper Sphere</i>	$F_1(x) = \sum_{i=1}^n x_i^2$	30	[-100, 100]	0
<i>MultiModal</i>	$F_2(x) = \sum_{i=1}^n x_i^2 x_i + \prod_{i=1}^n x_i $	30	[-10, 10]	0
<i>Schwefel (02)</i>	$F_3(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	30	[-100, 100]	0
<i>Schwefel (21)</i>	$F_4(x) = \max\{ x_i , 1 \leq i \leq n\}$	30	[-100, 100]	0
<i>Rosenbrock</i>	$F_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	[-30, 30]	0
<i>Step</i>	$F_6(x) = \sum_{i=1}^n ([x_i + 0.5])^2$	30	[-100, 100]	0
<i>Quartic</i>	$F_7(x) = \sum_{i=1}^n ix_i^4 + \text{random}[0,1]$	30	[-1.28, 1.28]	0

Tablo 2. Çok modlu kalite testi fonksiyonlarının tanımı

Fonksiyon	Formülasyon	V_{no}	Aralık	F_{min}
<i>Schwefel (26)</i>	$F_8(x) = \sum_{i=1}^n -x_i \sin(\sqrt{ x_i })$	30	[-500, 500]	-418.9829 x 5
<i>Rastrigin</i>	$F_9(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[-5.12, 5.12]	0
<i>Ackley</i>	$F_{10}(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$	30	[-32, 32]	0
<i>Griwank</i>	$F_{11}(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{i}\right) + 1$	30	[-600, 600]	0
<i>Penalty(01)</i>	$F_{12}(x) = \frac{\pi}{n} \{10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 + \sum_{i=1}^n u(x_i, 10, 100, 4)\}$ $y_i = 1 + \frac{x_i + 1}{4} u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	30	[-50, 50]	0
<i>Penalty(02)</i>	$F_{13}(x) = 0.1 \{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	30	[-50, 50]	0

Tablo 3. Sabit boyutlu çok modlu kalite testi fonksiyonlarının tanımları

Fonksiyon	Formülasyon	V_{no}	Aralık	F_{min}
De Jong (5)	$F_{14}(x) = (\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^j (x_i - a_{ij})^6})^{-1}$	2	[-65, 65]	1
Kowalik	$F_{15}(x) = \sum_{i=1}^{11} [a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}]^2$	4	[-5, 5]	0.00030
Six Hump Camel Back	$F_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	[-5, 5]	-1.0316
Branin (01)	$F_{17}(x) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 10$	2	[-5, 5]	0.398
Goldstein & Price I	$F_{18}(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	[-2, 2]	3
Hartmann (H3,4)	$F_{19}(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{j=1}^3 a_{ij}(x_j - p_{ij})^2)$	3	[1, 3]	-3.86
Hartmann (H6,4)	$F_{20}(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{j=1}^6 a_{ij}(x_j - p_{ij})^2)$	6	[0, 1]	-3.32
Shekel (m=5)	$F_{21}(x) = -\sum_{i=1}^5 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0, 10]	-10.1532
Shekel (m=7)	$F_{22}(x) = -\sum_{i=1}^7 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0, 10]	-10.4028
Shekel (m=10)	$F_{23}(x) = -\sum_{i=1}^5 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0, 10]	-10.5363

Tablo 4. Tek modlu kalite testi fonksiyonları sonuçları

F	\hat{I}	ASKA	PSO	SKA	BOA	KKO	YAA
F1	ort	4.1321e-137	6.5085e-16	0.0310	1.4109e-154	1.2766	1.0976e-16
	sd	2.2632e-136	1.1800e-15	0.0866	7.0872e-154	0.8429	3.5117e-17
	en iyi	2.5945e-179	1.0232e-17	2.4792e-06	3.3634e-166	0.4405	6.6897e-17
	en kötü	1.2396e-135	5.4858e-15	0.3945	3.8875e-153	4.4251	2.1405e-16
	süre	1.01338	8.3309	1.2570	4.2849	43.7911	12.0372
F2	ort	1.3305e-70	2.1071e-05	2.1026e-05	5.0254e-102	3.6260e+03	5.2733e-08
	sd	4.3038e-70	1.0554e-04	4.3726e-05	1.9523e-101	1.9618e+04	1.6126e-08
	en iyi	8.3720e-84	6.9921e-10	2.9768e-09	1.4491e-114	1.1190	2.7875e-08
	en kötü	2.1926e-69	5.7914e-04	2.1087e-04	9.2622e-101	1.0750e+05	1.0028e-07
	süre	1.12487	8.5741	1.3360	4.5383	46.1645	10.7295
F3	ort	2.3615e-71	7.6568	4.0282e+03	1.9940e+04	1.0596e+05	432.9601
	sd	1.2678e-70	5.5415	3.2698e+03	1.1221e+04	2.9680e+04	155.3030
	en iyi	9.8711e-120	1.0370	109.8140	3.6202e+03	4.7676e+04	217.7493
	en kötü	6.9476e-70	22.0031	1.4489e+04	4.3344e+04	1.6262e+05	780.2057
	süre	5.92186	17.1235	6.2230	9.6729	51.2220	14.0386
F4	ort	7.9899e-69	0.6941	19.9222	40.8308	78.1506	1.5515
	sd	4.3646e-68	0.3446	11.2356	32.4037	9.6495	1.5699
	en iyi	4.6344e-89	0.2272	1.5395	0.0519	42.8106	1.0044e-08
	en kötü	2.3908e-67	1.3777	42.7265	92.0102	92.1774	5.3571
	süre	1.42658	12.2517	1.6804	5.2079	39.2201	7.3923
F5	ort	0.0337	41.3337	1.0127e+03	27.2594	5.0124e+04	36.3915
	sd	0.0428	32.4499	2.8445e+03	0.6228	4.5712e+04	53.9666
	en iyi	1.0003e-04	2.3651	28.3801	26.5891	6.4021e+03	24.5371
	en kötü	0.1531	110.3879	1.2227e+04	28.7495	1.9159e+05	322.0843
	süre	1.55189	11.0633	1.7429	4.7569	38.6504	9.5483
F6	ort	0.0020	2.7841e-15	4.5788	0.1062	1.0898	0.2000
	sd	0.0029	8.7185e-15	0.5852	0.1165	0.5781	0.7611
	en iyi	8.5915e-06	1.7238e-17	3.8463	0.0094	0.4107	0
	en kötü	0.0120	4.7394e-14	6.7212	0.4399	3.1241	4.0000
	süre	1.52137	9.5745	1.8441	4.7322	36.7716	10.1557
F7	ort	7.1326e-05	0.0148	0.0421	0.0024	0.1854	0.0647
	sd	7.7696e-05	0.0059	0.0528	0.0023	0.0757	0.0254
	en iyi	2.2908e-07	0.0063	0.0047	5.6430e-05	0.0546	0.0094
	en kötü	3.9592e-04	0.0282	0.2775	0.0090	0.4039	0.1131
	süre	2.03204	8.5231	2.3571	6.2950	42.1256	10.7440

Tablo 4 de verilen tek modlu kalite testi fonksiyonlarına ait istatistiksel sonuçları incelendiğinde ASKA'nın optimum sonuca karşılaştırılan diğer tüm algoritmalarından (SKA, PSO, BOA, KKO, YAA) çok daha kısa sürede

yakınsadığı açıkça görülmektedir. ASKA, F1, F3, F5 ve F7 fonksiyonlarında en başarılı algoritma olmuştur. F2 fonksiyonunda BOA, F6 fonksiyonunda ise PSO en başarılı algoritmalar olmuştur.

Tablo 5. Çok modlu kalite testi fonksiyonları sonuçları

F	\bar{I}	ASKA	PSO	SKA	BOA	KKO	YAA
F8	ort	-1.2569e+04	-6.3686e+03	-3.9367e+03	-1.1224e+04	-4.4590e+149	-2.4485e+03
	sd	0.0103	867.5216	258.2417	1.6018e+03	2.4407e+150	425.4308
	en iyi	-1.2569e+04	-8.0276e+03	-4.5524e+03	-1.2569e+04	-1.3368e+151	-3.5570e+03
	en kötü	-1.2569e+04	-4.3764e+03	-3.5344e+03	-7.8490e+03	-3.0059e+98	-1.7879e+03
	süre	1.74441	8.6910	2.0996	5.0582	40.6682	9.4981
F9	ort	0	44.7399	23.4859	0	252.9477	27.0629
	sd	0	13.7505	32.5145	0	18.7779	6.2785
	en iyi	0	23.8790	6.9014e-06	0	193.7615	17.9093
	en kötü	0	81.5864	168.7336	0	276.6462	41.7882
	süre	1.66793	9.8125	1.9521	4.8869	39.8226	13.6292
F10	ort	8.8818e-16	1.0915	12.6852	4.0856e-15	0.6876	7.8281e-09
	sd	0	0.7594	9.5190	2.3511e-15	0.3804	1.6719e-09
	en iyi	8.8818e-16	2.7719e-09	2.5556e-04	8.8818e-16	0.1548	5.7326e-09
	en kötü	8.8818e-16	2.3162	20.3227	7.9936e-15	1.7909	1.3408e-08
	süre	1.74515	10.4712	2.1039	5.0623	42.3656	15.1583
F11	ort	0	0.0240	0.3685	0.0050	0.9188	8.2013
	sd	0	0.0246	0.3339	0.0193	0.0798	3.2014
	en iyi	0	0	7.5275e-04	0	0.6492	2.6444
	en kötü	0	0.0860	0.9431	0.0875	1.0283	14.4065
	süre	1.90766	11.9485	2.2472	5.6325	41.3976	14.9480
F12	ort	5.5727e-05	0.2319	3.1533	0.0110	3.2754e+04	0.1608
	sd	1.0953e-04	0.5471	6.2240	0.0177	7.9615e+04	0.2849
	en iyi	4.6736e-08	1.0560e-18	0.3258	0.0012	18.9615	3.5333e-19
	en kötü	5.2938e-04	2.7038	33.4485	0.0956	3.2040e+05	1.4847
	süre	3.67295	16.8464	3.9721	4.5318	42.5764	15.5377
F13	ort	6.4582e-04	0.1020	18.5548	0.1962	8.1417e+04	0.0033
	sd	7.6868e-04	0.4602	65.8292	0.1581	1.1285e+05	0.0104
	en iyi	6.5958e-06	5.5146e-18	2.2029	0.0398	961.1513	4.1030e-18
	en kötü	0.0032	2.5085	365.0753	0.7707	3.9818e+05	0.0548
	süre	3.65425	11.5014	3.9894	7.6812	43.5993	13.4597

Tablo 5 deki çok modlu kalite testi fonksiyonlarına ait istatistiksel sonuçlara göre, ASKA, F8-F13 fonksiyonların tamamında en iyi

sonucu veren algoritma olmuştur. F8, F9 ve F11 fonksiyonlarında ise optimum sonucu elde etmiştir.

Tablo 6. Sabit boyutlu çok modlu kalite testi fonksiyonları sonuçları

F	\bar{I}	ASKA	PSO	SKA	BOA	KKO	YAA
F14	ort	1.2626	4.4098	1.5275	2.7961	1.3235	3.4221
	sd	0.6860	2.9700	0.8922	3.2852	1.7829	2.6992
	en iyi	0.9980	0.9980	0.9980	0.9980	0.9980	0.9980
	en kötü	2.9821	11.7187	2.9821	10.7632	10.7632	13.8192
	süre	8.50384	20.2909	10.3350	10.0869	19.4695	17.3979
F15	ort	3.8053e-04	3.4190e-04	9.7180e-04	5.6780e-04	0.0011	0.0023
	sd	7.1784e-05	1.6780e-04	3.8543e-04	2.7009e-04	3.1570e-04	8.7184e-04
	en iyi	3.1178e-04	3.0749e-04	3.4077e-04	3.0836e-04	8.8731e-04	6.2116e-04
	en kötü	5.9624e-04	0.0012	0.0015	0.0015	0.0019	0.0052
	süre	1.63308	9.4237	1.9942	2.0983	8.7483	9.6892
F16	ort	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316
	sd	2.1956e-05	6.7752e-16	2.5344e-05	1.6070e-10	6.7752e-16	4.8787e-16
	en iyi	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316
	en kötü	-1.0315	-1.0316	-1.0315	-1.0316	-1.0316	-1.0316
	süre	1.19193	9.8322	1.3675	1.4481	5.5078	7.5780

Tablo 6. Sabit boyutlu çok modlu kalite testi fonksiyonları sonuçları (devam)

<i>F</i>	<i>İ</i>	ASKA	PSO	SKA	BOA	KKO	YAA
F17	<i>ort</i>	0.3981	0.3979	0.3985	0.3979	0.3979	0.3979
	<i>sd</i>	4.8979e-04	0	6.0681e-04	3.9088e-06	0	0
	<i>en iyi</i>	0.3979	0.3979	0.3979	0.3979	0.3979	0.3979
	<i>en kötü</i>	0.3998	0.3979	0.4003	0.3979	0.3979	0.3979
	<i>süre</i>	0.7232	7.8293	1.2613	1.3837	5.69707	7.2001
F18	<i>ort</i>	3.0002	3.0000	3.0000	3.0000	3.0000	3.0000
	<i>sd</i>	2.3677e-04	1.7954e-15	3.2611e-05	9.2119e-05	1.3194e-15	3.1250e-15
	<i>en iyi</i>	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000
	<i>en kötü</i>	3.0008	3.0000	3.0001	3.0005	3.0000	3.0000
	<i>süre</i>	0.76903	9.1013	1.3814	1.4617	5.8880	7.2429
F19	<i>ort</i>	-3.8146	-3.8370	-3.8551	-3.8609	-3.8628	-3.8628
	<i>sd</i>	0.0800	0.1411	0.0024	0.0022	2.7101e-15	2.2913e-15
	<i>en iyi</i>	-3.8627	-3.8628	-3.8618	-3.8628	-3.8628	-3.8628
	<i>en kötü</i>	-3.6144	-3.0898	-3.8519	-3.8549	-3.8628	-3.8628
	<i>Süre</i>	2.05760	10.1333	2.3266	2.4045	7.7821	9.2076
F20	<i>ort</i>	-2.9617	-3.2863	-2.8866	-3.2429	-3.2665	-3.3220
	<i>sd</i>	0.2679	0.0554	0.4145	0.1113	0.0603	1.4889e-15
	<i>en iyi</i>	-3.2900	-3.3220	-3.2333	-3.3219	-3.3220	-3.3220
	<i>en kötü</i>	-2.0038	-3.2031	-1.2291	-2.9868	-3.2031	-3.3220
	<i>süre</i>	2.09464	9.9590	1.8505	3.1532	11.7540	8.5954
F21	<i>ort</i>	-10.1516	-5.4786	-2.2850	-9.0471	-5.6870	-6.4134
	<i>sd</i>	0.0033	3.4558	1.8892	2.2807	3.7094	3.6846
	<i>en iyi</i>	-10.1532	-10.1532	-5.9900	-10.1530	-10.1532	-10.1532
	<i>en kötü</i>	-10.1375	-2.6305	-0.4965	-2.6303	-2.6829	-2.6305
	<i>süre</i>	2.30390	11.3522	2.6262	2.7609	11.0864	8.8242
F22	<i>ort</i>	-10.4017	-6.9147	-3.5487	-7.7342	-6.8594	-10.4029
	<i>sd</i>	0.0025	3.6103	1.6587	2.9214	2.5485	1.4378e-15
	<i>en iyi</i>	-10.4029	-10.4029	-5.2185	-10.4027	-10.4029	-10.4029
	<i>en kötü</i>	-10.3923	-2.7519	-0.9071	-3.7235	-5.0877	-10.4029
	<i>süre</i>	3.05327	11.5451	3.1160	3.1399	14.0219	9.9985
F23	<i>ort</i>	-10.5352	-5.9754	-4.0940	-8.5166	-7.2484	-10.2897
	<i>sd</i>	0.0029	3.6315	1.7448	2.9532	3.0040	1.3511
	<i>en iyi</i>	-10.5363	-10.5363	-7.9749	-10.5363	-10.5363	-10.5363
	<i>en kötü</i>	-10.5221	-2.4217	-0.9460	-2.4217	-2.4217	-3.1359
	<i>süre</i>	3.14048	12.0961	3.5207	3.8171	13.2174	11.2024

Tablo 6 daki sabit boyutlu çok modlu kıyaslama fonksiyonlarının istatistiksel sonuçlar incelendiğinde, Kalite testi fonksiyonları sonuçları incelendiğinde F14, F16- F18, F21-F23 fonksiyonunda ASKA ve diğer optimizasyonlardan bazıları en iyi çözüm değerine bakıldığında optimum sonuca erişmişlerdir. F19 fonksiyonunda sadece ASKA optimum sonuca erişirken, F15 ve F17 fonksiyonlarında optimum sonuca erişen optimizasyon olmamıştır.

Sayısal optimizasyon problemlerinin çözümü için önerilen algoritmaların başarımını kıyaslamak için parametrik olmayan testlerden Wilcoxon işaretli sıralar testi uygulanmıştır [12].

İstatistiksel anlamlılık değeri $\alpha = 0.05$ olmak üzere ASKA ile diğer metasezgisel algoritmalar arasında Wilcoxon işaretli sıralar testi yapılarak

istatistiksel sonuçlar Tablo 7’de gösterilmiştir. Burada $p < 0.05$ olması durumu karşılaştırılan algoritmalar arasında elde edilen sonuçlar arasında istatistiksel açıdan anlamlı bir farkın olduğunu göstermektedir. Ayrıca R^+ ASKA’nın karşılaştırılan algoritmaya göre daha üstün sonuçlar elde ettiği rankların toplamını gösterirken, R^- ise karşılaştırılan algoritmanın ASKA’ya göre daha üstün sonuçlar elde ettiği rankların toplamını gösterir. K kazanma durumunu ifade eder.

Tablo 7’deki Wilcoxon işaretli sıralar testi sonuçlarına göre ASKA 23 kalite testi fonksiyonu üzerinden; SKA’ya karşı 21/2, PSO’ya karşı 16/7, BOA’ya karşı 16/7, KKO’ya karşı 16/7, YAA’ya karşı 14/9 üstün gelmektedir.

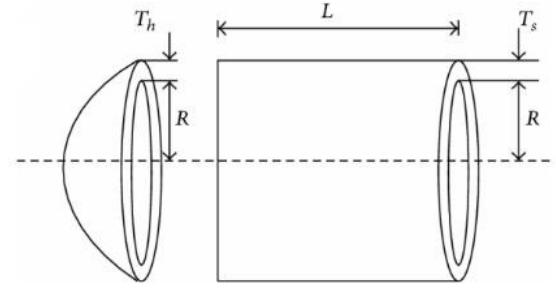
Tablo 7. Wilcoxon işaretli sıralar testi karşılaştırma sonuçları

F	SKA / ASKA				PSO / ASKA				BOA / ASKA				KKO / ASKA				YAA / ASKA			
	p - value	R ⁺	R ⁻	K	p-değeri	R ⁺	R ⁻	K	p - value	R ⁺	R ⁻	K	p - value	R ⁺	R ⁻	K	p - value	R ⁺	R ⁻	K
F1	1.73e-06	465	0	+	1.73e-06	465	0	+	2.23e-04	53	412	-	1.73e-06	465	0	+	1.73e-06	465	0	+
F2	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	0	465	-	1.73e-06	465	0	+	1.73e-06	465	0	+
F3	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+
F4	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+
F5	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+
F6	1.73e-06	465	0	+	1.73e-06	0	465	-	1.73e-06	465	0	+	1.73e-06	465	0	+	0.0028	87	378	-
F7	1.73e-06	465	0	+	1.73e-06	465	0	+	1.92e-06	464	1	+	1.73e-06	465	0	+	1.73e-06	465	0	+
F8	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+	1.73e-06	0	465	-	1.73e-06	465	0	+
F9	1.73e-06	465	0	+	1.73e-06	465	0	+	1	0	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+
F10	1.73e-06	465	0	+	1.73e-06	465	0	+	1.33e-05	253	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+
F11	1.73e-06	465	0	+	2.56e-06	435	0	+	0.5000	3	0	+	1.73e-06	465	0	+	1.73e-06	465	0	+
F12	1.73e-06	465	0	+	0.6435	255	210	+	1.73e-06	465	0	+	1.73e-06	465	0	+	0.0166	349	116	+
F13	1.73e-06	465	0	+	0.3820	275	190	+	1.73e-06	465	0	+	1.73e-06	465	0	+	0.0571	140	325	-
F14	0.0020	383	82	+	1.13e-05	446	19	+	0.4405	270	195	+	3.11e-05	30	435	-	1.64e-05	442	23	+
F15	1.73e-06	465	0	+	1.60e-04	49	416	-	4.90e-04	402	63	+	1.73e-06	465	0	+	1.73e-06	465	0	+
F16	0.7189	250	215	+	1.73e-06	0	465	-	1.73e-06	0	465	-	1.73e-06	0	465	-	1.73e-06	0	465	-
F17	5.71e-04	400	65	+	1.73e-06	0	465	-	6.33e-06	13	452	-	1.73e-06	0	465	-	1.73e-06	0	465	-
F18	0.0064	100	365	-	1.73e-06	0	465	-	0.0024	85	308	-	1.73e-06	0	465	-	1.73e-06	0	465	-
F19	0.4165	193	272	-	3.11e-05	30	435	-	6.16e-04	66	399	-	1.73e-06	0	465	-	1.73e-06	0	465	-
F20	0.6733	253	212	+	1.73e-06	0	465	-	1.02e-05	18	447	-	2.60e-06	4	461	-	1.73e-06	0	465	-
F21	1.73e-06	465	0	+	2.61e-04	410	55	+	0.0024	380	85	+	0.0015	387	78	+	0.0087	360	105	+
F22	1.73e-06	465	0	+	0.0207	345	120	+	2.37e-05	438	27	+	2.61e-04	410	55	+	1.73e-06	0	465	-
F23	1.73e-06	465	0	+	6.15e-04	399	66	+	1.63e-05	442	23	+	0.0036	374	91	+	3.12e-05	30	435	-

Önerilen yeni adaptif algoritma, mühendislik tasarım problemlerinden biri olan basınçlı kap probleminin çözümünde kullanılarak algoritmanın kısıtlı problemler üzerindeki etkinliği test edilmiştir. ASKA ve karşılaştırılan diğer algoritmalar (PSO, SKA, BOA, KKO, YAA) basınçlı kap problemi üzerinde test edilirken popülasyon sayısı 200, iterasyon sayısı 10000 olarak belirlenmiştir ve algoritmalar 30 kez çalıştırılmıştır. Elde edilen sonuçlar Tablo 8'de, Wilcoxon sıra toplam testi istatistiksel sonuçları Tablo 9'da gösterilmektedir.

Şekil 4'te gösterildiği gibi silindirik bir kabın her iki ucu yarı küresel başlıklarla kapalıdır. Burada amaç; malzeme, şekillendirme ve kaynak maliyeti olmak üzere toplam maliyeti en aza indirmektir. Bu problemde dört tasarım değişkeni vardır: T_s (kabuk kalınlığı, x_1), T_h (başım kalınlığı, x_2), R (iç yarıçap, x_3) ve L (kabın silindirik bölümünün başlık haricindeki uzunluğu, x_4) [13].

Bu problem 4 kısıttan oluşur. Bu kısıtlar ve problem Denklem 2'de gösterilmektedir.



Şekil 4. Basınçlı kap problemi

$$\begin{aligned}
 \min f(\vec{x}) &= 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \\
 \text{s.t. } g_1(\vec{x}) &= -x_1 + 0.0193x_3 \leq 0 \\
 g_2(\vec{x}) &= -x_2 + 0.00954x_3 \leq 0 \\
 g_3(\vec{x}) &= -\pi x_3^2x_4 - \frac{4}{3}\pi x_3^3 + 1.296000 \leq 0 \quad (2) \\
 g_4(\vec{x}) &= x_4 - 240 \leq 0 \\
 \text{Değişken aralığı} & \quad 0 \leq x_1 \leq 99, \\
 & \quad 0 \leq x_2 \leq 99, \\
 & \quad 10 \leq x_3 \leq 200, \\
 & \quad 10 \leq x_4 \leq 200
 \end{aligned}$$

Tablo 8. Basınçlı kap problemi karşılaştırma sonuçları

Algoritma	Optimum değişkenler				Optimum Maliyet	Süre
	T_s	T_h	R	L		
ASKA	0.778566726025	0.387863076807	40.322074399104	200	5897.91316	59.53190
PSO	0.828797029813	0.409674804493	42.942851305476	166.44667183143	5977.62589	517.86163
SKA	0.780408762923	0.386271554329	40.406252027563	200	5920.53556	59.77302
BOA	0.778189825887	0.390165645138	40.319639981412	199.99970408291	5901.42948	87.78191
KKO	0.921103032395	0.455301706112	47.725545735306	117.48048122090	6177.27956	498.07486
YAA	0.896775543832	0.443276615916	46.465054100846	129.12087575297	6120.54193	2907.64910

Tablo 9. Basınçlı kap problemi Wilcoxon sıra toplam testi karşılaştırma sonuçları

	PSO / ASKA	SKA / ASKA	BOA / ASKA	KKO / ASKA	YAA / ASKA
p - değeri	0.0056	8.5641e-04	0.3555	0.0020	0.0083

ASKA basınçlı kap problemini **59.53190** saniyede çözerek problemin minimum maliyetini **5897.91316** olarak bulması yapılan iyileştirmenin gücünü kanıtlamaktadır. Çalışma zamanı ve minimum maliyeti bulma açısından ASKA basınçlı kap probleminin çözümünde karşılaştırılan algoritmalar arasında en iyi sonucu vermektedir. Tablo 9’da verilen Wilcoxon sıra toplamı testi karşılaştırma sonuçlarına göre de ASKA’nın rakiplerine karşı üstün geldiği açıkça görülmektedir.

4. Sonuç

Bu çalışmada, SKA üzerinde iyileştirmeler yapılarak geliştirilen Adaptif Sinüs Kosinüs Algoritması (ASKA) 23 kalite testi fonksiyonu ve mühendislik tasarım problemlerinden biri olan basınçlı kap probleminin çözümünde test edilmiştir. Ayrıca stokastik yöntemlerin performansının daha güvenilir bir şekilde belirlenmesini sağlayan istatistiksel testlerden Wilcoxon sıra toplamı testi kullanılarak algoritmalarından elde edilen sonuçlar istatistiksel olarak karşılaştırılmıştır. Test sonuçlarından önerilen yeni algoritmanın SKA’ya ve karşılaştırılan diğer algoritmalara göre daha iyi sonuçlar verdiği görülmektedir. Wilcoxon sıra toplamı testi sonuçları da ASKA’nın diğer algoritmalara karşı üstünlüğünü kanıtlamaktadır.

5. Kaynaklar

1. Tanyildizi, E. and Demir, G. (2017). “Golden Sine Algorithm: A Novel Math-Inspired Algorithm”, *Advances in Electrical and Computer Engineering*, **17(2)**:71-78.

2. H. Eskandar, A. Sadollah, A. Bahreininejad and M. Hamdi, (2012). “M. Water cycle algorithm—a novel metaheuristic optimization method for solving constrained engineering optimization problems”, *Computers & Structures*, **110-111**:151-166.
3. Mirjalili, S., Mirjalili, S.M., Lewis, A. (2014). Grey wolf optimizer, *Adv Eng Softw*, **69**, 46-61.
4. K. S. Lee, Z. W. Geem, (2005). “A new metaheuristic algorithm for continuous engineering optimization: harmony search theory and practice”, *Comput. Methods Appl. Mech. Engrg.*, **194**:3902–3933,
5. A. Prakasam, N. Savarimuthu, (2015). “Metaheuristic Algorithms and Polynomial Turing Reductions: A Case Study Based on Ant Colony Optimization”, *Procedia Computer Science*, vol.46, pp. 388 – 395,
6. I. Fister Jr., X. S. Yang, D. Fister, I. Fister, (2013). “A brief review of nature-inspired algorithms for optimization”, *Elektrotehniski Vestnik*, **80(3)**: 1-7,
7. Mirjalili, S. (2016). SCA: A sine cosine algorithm for solving optimization problems. *Knowledge-Based Systems*, **96**, 120-133.
8. Kennedy, J., Eberhart, R. (1995). Particle swarm optimization in neural networks, *IEEE International Conference*, **4**, 1942–1948.
9. Dorigo, M. (1992). Optimization, learning and natural algorithms. Ph. D. Thesis, Politecnico di Milano, Italy.
10. Mirjalili, S., Mirjalili, S.M. (2016). The whale optimization algorithm, *Adv Eng Softw*, **95**, 51-67.
11. Rashedi, E., Pour H.N., Saryazdi, S. (2009). GSA: a gravitational search algorithm, *Information sciences*, **179(13)**, 2232-2248.

12. Derrac, J., García, S., Molina, D., Herrera, F., (2011). A practical tutorial on the use of non-parametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, *Swarm Evol. Comput.* **1**:3–18.
13. Nasseri, S.H., Alizadeh, Z., Taleshian, F., (2012). Optimized solution of pressure vessel design using geometric programming. *The Journal of Mathematics and Computer Science.* **4**(3):344 – 349.