



Karga Arama Algoritması İçin Parametre Analizleri

Emine BAŞ^{1*}

^{1*} Selçuk Üniversitesi, Kulu Meslek Yüksekokulu, Bilgisayar Teknolojileri Bölümü, Konya, Türkiye, (ORCID: 0000-0003-4322-6010), emineozcan@selcuk.edu.tr

(International Conference on Design, Research and Development (RDCONF) 2021 – 15-18 December 2021)

(DOI: 10.31590/ejosat.1039646)

ATIF/REFERENCE: Baş, E., (2021). Karga Arama Algoritması için Parametre Analizleri. *Avrupa Bilim ve Teknoloji Dergisi*, (32), 878-882.

Öz

Bu makale, kargaların akıllı davranışına dayanan, Karga Arama Algoritması (KAA) adlı yeni bir metasezgisel algoritmayı tanıtmaktadır. KAA, kargaların fazla yiyeceklerini saklanma yerlerinde sakladığı ve yiyecek gerektiğinde geri aldığı bu fikirden yola çıkarak çalışan popülasyona dayalı bir tekniktir. KAA metodu üzerinde fl sabit parametresi lokal ve global arama yeteneği arasında önemli farklılıklar yaratmaktadır. Bu çalışmada beş farklı fl değeri belirlenmiş ve KAA 'nın performansı üzerindeki etkisi araştırılmıştır. KAA ile on farklı son yıllarda geliştirilmiş CEC-C06-2019 seri fonksiyonları çözülmüştür. KAA ile çeşitli sonuçlar elde edilmiştir (ortalama, standart sapma, en iyi ve en kötü). KAA ile elde edilen sonuçlar birbirleri ile ve çeşitli sezgisel algoritmaların sonuçları ile karşılaştırılmıştır. Test sonuçları, KAA kullanımının diğer algoritmalara kıyasla umut verici sonuçlar bulmasına yol açabileceğini ortaya koymaktadır.

Anahtar Kelimeler: Karga Arama Algoritması, Sürekli optimizasyon, CEC-C06-2019.

Parameter Analysis for Crow Search Algorithm

Abstract

This paper introduces a new metaheuristic algorithm named Crow Search Algorithm (CSA) based on the intelligent behavior of crows. CSA is a population-based technique that works from this idea that crows store their excess food in their hiding places and retrieve it when needed. On the CSA method, the constant parameter fl creates significant differences between local and global search capabilities. In this study, five different fl values were determined and the effect of CSA on performance was investigated. CEC-C06-2019 serial functions developed in ten different recent years have been solved with CSA. Various results were obtained with CSA (mean, standard deviation, best and worst). The results obtained by CSA were compared with each other and with the results of various heuristic algorithms. The test results reveal that the use of CSA can lead to promising results compared to other algorithms.

Keywords: Crow Search Algorithm, Continuous optimization, CEC-C06-2019.

* Sorumlu Yazar: emineozcan@selcuk.edu.tr

1. Giriş

Metasezgisel algoritmalar, son derece doğrusal olmayan ve çok modlu olan çoğu gerçek dünya optimizasyon problemini çözmek için umut verici bir performans göstermektedir. Tüm meta-sezgisel algoritmalar, belirli bir rastgeleştirme ve yerel arama dengesi kullanır. Bu algoritmalar zor optimizasyon problemleri için iyi çözümler bulabilir, ancak optimal çözümlere ulaşabileceğinin garantisi yoktur. Bu algoritmaların çoğu zaman işe yaraması umulmaktadır, ancak her zaman bunu garantilemez. Metasezgisel algoritmalar küresel optimizasyon için uygun olabilir. Tüm modern doğadan ilham alan yöntemlere metasezgisel şekilde isim verilmektedir (Yang, 2011).

Mevcut eğilim, zor problemlerin üstesinden gelmek için doğadan ilham alan metasezgisel algoritmaları kullanmaktır ve metasezgisellerin şaşırtıcı bir şekilde çok verimli olduğu gösterilmiştir. Bu nedenle, metasezgisel literatür son yirmi yılda muazzam bir şekilde genişlemiştir. İyi bilinen metasezgisel algoritmalarından bazıları şunlardır: doğal seleksiyona dayalı genetik algoritma (GA), kuş sürüsünün sosyal davranışına ve balık sürüsüne dayalı parçacık sürü optimizasyonu (PSO), müzik doğaçlama sürecine dayalı armoni araması (HS), bazı guguk kuşu türlerinin kuluçka parazitliğine dayalı guguk kuşu arama algoritması, mikro yarasaların ekolojisi davranışına dayalı yarasalar algoritması (BA), hayvan arama davranışına dayalı grup arama optimize edicisi (GSO), tropiklerin yanıp sönen ışık modellerine dayalı ateş böceği algoritması (FA) vb. Bu makalede tanıtılan metasezgisel yöntem optimizasyon problemlerini çözerken umut verici sonuçlar elde edebileceğimiz kullanıcı dostu (basit konsept ve kolay uygulama) bir metasezgisel tekniktir.

Kargalar, şu anda dünyanın en zeki hayvanları arasında sayılan, yaygın olarak dağılmış bir kuş cinsidir. Bir grup olarak, kargalar dikkat çekici zeka örnekleri gösterirler ve zeka testlerinde genellikle çok yüksek puanlar alırlar. Yüzleri ezberleyebilir, araçları kullanabilir, karmaşık yollarla iletişim kurabilir ve mevsimler boyunca yiyecekleri saklayabilir ve alabilirler (Askarzadeh, 2016).

Bir karga sürüsü, optimizasyon süreciyle pek çok benzerliği olan bir davranış sergilerler. Bu davranışa göre kargalar fazla yiyeceklerini ortamın belirli konumlarında (saklanma yerlerinde) saklarlar ve depolanan yiyecekleri gerektiğinde geri bulurlar. Kargalar daha iyi besin kaynakları elde etmek için birbirlerini takip ettikleri için açgözlü kuşlardır. Bir karganın gizlediği besin kaynağını bulmak kolay bir iş değildir, çünkü bir karga onu takip eden başka bir kargayı görürse, karga ortamın başka bir pozisyonuna giderek o kargayı kandırmaya çalışır. Optimizasyon açısından, kargalar arama ortamında araştırmacıdır, mevcut ortam arama alanıdır, ortamın her konumu uygun bir çözüme karşılık gelir, besin kaynağının kalitesi objektif (uygunluk) fonksiyonudur ve ortamın en iyi besin kaynağı problemin en iyi çözümüdür (Global çözüm). Bu benzerliklere dayanarak, KAA, optimizasyon problemlerinin çözümünü bulmak için kargaların akıllı davranışlarını simüle etmeye çalışır.

2. Materyal ve Metot

2.1. Karga Arama Algoritması (KAA)

Kargaların diğer kuşları izlediği, diğer kuşların yiyeceklerini nereye sakladıklarını gözlemlediği ve yiyeceğin sahibi gittikten sonra çaldıkları bilinmektedir. Bir karga hırsızlık yaptıysa, gelecekteki bir kurban olmamak için saklanma yerlerini

değiştirmek gibi ekstra önlemler alırlar. Aslında, bir hırsızın davranışını tahmin etmek için kendi hırsız olma deneyimlerini kullanırlar ve önbelleklerini çalmaktan korumak için en güvenli yolu belirleyebilirler. Bu makalede, daha önce bahsedilen akıllı davranışlara dayalı olarak, popülasyon tabanlı bir metasezgisel algoritma olan KAA anlatılmış ve geliştirilmiştir. KAA ilkeleri şu şekilde sıralanmıştır:

- ✓ Kargalar sürü şeklinde yaşar.
- ✓ Kargalar saklandıkları yerin konumunu ezberler.
- ✓ Kargalar hırsızlık yapmak için birbirini takip eder.
- ✓ Kargalar, önbelleklerini bir olasılıkla çalmaktan korur

Bir dizi kargayı içeren d boyutlu bir ortamın olduğu varsayılmaktadır. Kargaların sayısı (sürü boyutu) N 'dir ve karga i 'nin arama uzayındaki zaman (yineleme) $iter$ 'deki konumu bir vektör $x^{i,iter}$ ($i=1, 2, \dots, N; iter = 1, 2, \dots, iter_{max}$) burada $x^{i,iter} = [x^{i,iter}_1, x^{i,iter}_2, \dots, x^{i,iter}_d]$ ve $iter_{max}$, maksimum yineleme sayısıdır. Her karganın saklandığı yerin hafızaya alındığı bir hafızası vardır. Yineleme $iter$ inde, karga i 'nin saklanma yeri $m^{i,iter}$ ile gösterilir. Bu, karganın şimdiye kadar elde ettiği en iyi pozisyonudur. Her karganın hafızası, en iyi deneyiminin konumunu ezberlemiştir. Kargalar çevrede hareket eder ve daha iyi besin kaynakları ararlar (saklanma yerleri).

Varsayalımki $iter$ yinelemede, karga j 'nin saklanma yerini yani $m^{j,iter}$ 'i ziyaret etmek istediğini varsayalım. Bu yinelemede, karga i , karga j 'nin saklanma yerine yaklaşmak için karga j 'yi takip etmeye karar verir. Bu durumda, iki durum olabilir:

Durum 1: Karga j , karga i 'nin kendisini takip ettiğini bilmiyordur. Bir sonuç olarak, karga i , karga j 'nin saklandığı yere yaklaşacaktır. Bu durumda, karga i 'nin yeni konumu aşağıdaki gibi elde edilir:

$$x^{i,iter+1} = x^{i,iter} + r_i \times fl^{i,iter} \times (m^{j,iter} - x^{i,iter}) \quad (1)$$

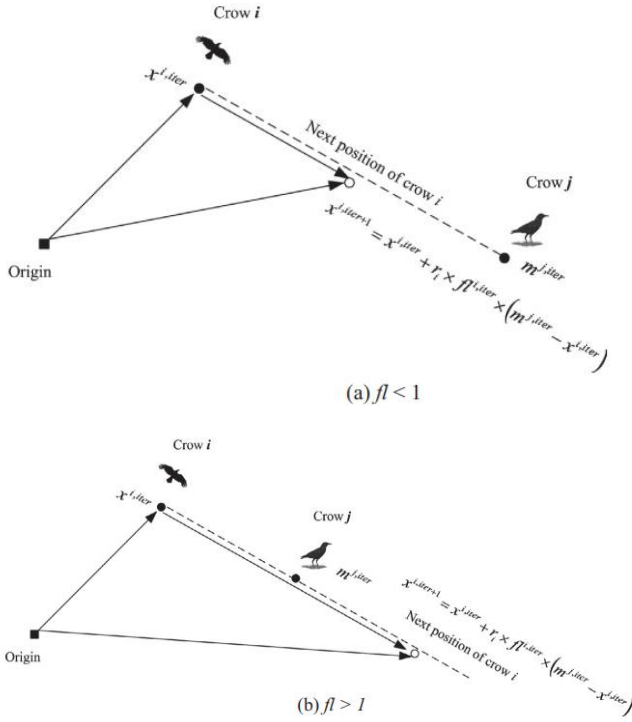
Burada r_i , 0 ile 1 arasında düzgün dağılıma sahip rastgele bir sayıdır ve $fl^{i,iter}$, yinelemede karga i 'nin uçuş uzunluğunu gösterir.

Durum 2: Karga j , karga i 'nin onu takip ettiğini biliyordur. Sonuç olarak, karga j , önbelleğini çalmaktan korumak için, arama uzayının başka bir konumuna giderek karga i 'yi kandıracaktır.

$$x^{i,iter+1} = \begin{cases} x^{i,iter} + r_i \times fl^{i,iter} \times (m^{j,iter} - x^{i,iter}) & r_j \geq AP^{i,iter} \\ Rastgele \text{ bir pozisyon} & \text{aksi halde} \end{cases} \quad (2)$$

Burada r_i , 0 ile 1 arasında düzgün dağılıma sahip rastgele bir sayıdır ve $AP^{i,iter}$, $iter$ yinelemede j kargasının farkındalık olasılığını belirtir.

Şekil 1 bu durumun şemasını ve fl 'in arama kabiliyeti üzerindeki etkisini göstermektedir. Küçük fl değerleri yerel aramaya yol açar ($x^{i,iter}$ yakınlarında) ve büyük değerler genel aramaya yol açar ($x^{i,iter}$ 'den uzak). Şekil 1 (a)'da gösterildiği gibi, fl değeri 1'den küçük seçilirse, karga i 'nin bir sonraki konumu, $x^{i,iter}$ ve $m^{j,iter}$ arasındaki kısa çizgi üzerindedir. Şekil 1(b)'nin gösterdiği gibi, fl değeri 1'den fazla seçilirse, karga i 'nin bir sonraki konumu, $m^{j,iter}$ 'i aşabilecek olan çizgi çizgisindedir.



Şekil 1. KAA'da durum 1'in akış şeması (a) $fl < 1$ ve (b) $fl > 1$. Karga i , çizgideki her konuma gidebilir (Askarzadeh, 2016).

Metasezgisel algoritmalar, çeşitlendirme ve yoğunlaştırma arasında iyi bir denge sağlamalıdır. KAA'da, yoğunlaştırma ve çeşitlendirme esas olarak farkındalık olasılığı (AP) parametresi tarafından kontrol edilir. Farkındalık olasılık değerinin düşürülmesiyle, KAA, aramayı bu bölgede mevcut iyi bir çözümün bulunduğu yerel bir bölgede yürütme eğilimindedir. Sonuç olarak, küçük AP değerleri kullanmak yoğunlaşmayı artırır. Öte yandan, farkındalık olasılık değerinin artmasıyla, mevcut iyi çözümlerin çevresini arama olasılığı azalır ve KAA, arama alanını küresel ölçekte keşfetmeye (randomizasyon) yönelir. Sonuç olarak, AP'nin büyük değerlerinin kullanılması çeşitliliği artırmaktadır.

KAA'nın sözde kodu Şekil 2'de gösterilmiştir

16: Kargaların hafızasını güncelleme
17: *end while*

Şekil 2. KAA için sözde kod

3. Araştırma Sonuçları ve Tartışma

Algoritma kodlamaları Matlab R2014a üzerinde kodlanmıştır. Uygulamalar işlemcisi intel Core i5 1.19Ghz, ram değeri 12Gb olan işletim sistemi windows10 olan bir makine üzerinde çalıştırılmıştır.

10 farklı sürekli optimizasyon problemini içeren CEC-C06 2019 kıyaslama fonksiyonları üzerinde test işlemleri gerçekleştirilmiştir. Bu test fonksiyonlarının seçilmesinin nedeni literatürde son yıllarda geliştirilmiş olmalarıdır. Bu test fonksiyonları Price ve ark. tarafından tek bir amaç optimizasyon problemi için geliştirilmiştir (Price et al., 2018). Test fonksiyonları, yıllık optimizasyon yarışmasında kullanılması amaçlanan "100-Number Challenge" olarak bilinir (Abdullah ve Rashid, 2019). CEC-C06 2019 kıyaslama test fonksiyonlarının tanımları Tablo 2'de gösterilmektedir. Tüm test fonksiyonları minimizasyon fonksiyonlarıdır ve ölçeklenebilir. İlk üç CEC fonksiyonunun boyutları sırasıyla 9, 16 ve 18'dir. Diğer CEC işlevlerinin boyutu 10'dur. CEC04 ile CEC10 arasındaki işlevler kaydırılır ve döndürülürken, ilk üç CEC fonksiyonu değildir.

KAA'nın performansı en iyi, en kötü, ortalama ve standart sapma hesaplamaları yapılarak gösterilmiştir. Her bir uygulama 20 bağımsız çalışma şeklinde değerlendirilmiştir. Uygulamalarda kullanılan parametre değerleri Tablo 1 de gösterilmiştir. fl parametre ayarı için 5 farklı değer seçilmiştir. Bu değerlerin sonuç üzerindeki etkisi bu makalede araştırılmıştır. Sonuçlar Tablolar 3–7'de gösterilmiştir. Tablo 8 de farklı fl parametre değerlerine göre ortalama sonuçlarının karşılaştırması gösterilmiştir. Tablo 9 ve Tablo 10'da ise KAA, DA (Mirjalili, 2015) ve WOA (Mirjalili and Lewis, 2016) sezgisel algoritmaları ortalama ve standart sapma kıyaslama ölçütlerine göre karşılaştırılmıştır.

Tablo 8 sonuçlarına göre, beş farklı fl parametre ortalama değeri karşılaştırılmış ve üstün performans gösteren sonuçlar kalın font ile işaretlenmiştir. Kıyaslama fonksiyonlarının yarısında üstün başarı gösteren fl değeri 4 olarak belirlenmiştir. Diğer değerler daha düşük başarı göstermiştir.

Tablo 9 sonuçlarına göre KAA ile DA ve WOA ortalama sonuçları karşılaştırılmıştır. Üstün performans gösteren sonuçlar kalın font ile işaretlenmiştir. On kıyaslama fonksiyonunun dört tanesinde DA, üç tanesinde KAA ve üç tanesinde de WOA üstün başarı sağlamıştır.

Algoritma: Karga Arama Algoritması

```

1: Arama alanındaki N karga sürüsünün konumunu rastgele başlat
2: Kargaların konumunu değerlendirme
3: Her karganın hafızasını başlatma
4: while  $iter < iter_{max}$ 
5:   for  $i = 1 : N$  (sürünün tüm N kargaları)
6:     Takip edilecek kargalardan birini rastgele seçme (örneğin j)
7:     Bir farkındalık olasılığı tanımlama
8:     if  $r_j \geq AP^{i,iter}$ 
9:        $x^{i,iter+1} = x^{i,iter} + r_i \times fl^{i,iter} \times (m^{j,iter} - x^{i,iter})$ 
10:    else
11:       $x^{i,iter+1} =$  arama uzayında random bir pozisyon
12:    end if
13:  end for
14:  Yeni pozisyonların fizibilitesini kontrol etme
15:  Kargaların yeni konumunu değerlendirme

```

Tablo 1. Parametre ayarları

Parametre	Değer
Populasyon sayısı	50
Boyut	{9, 10, 16, 18}
Maksimum iterasyon	500
AP	0.1
fl	{2, 4, 6, 8, 10}

Tablo 2. CEC-C06 2019 Test Fonksiyonlarının Tanımlaması

ID	Functions	Dimension	Range	f_{min}
1	Storn's Chebyshev Polynomial Fitting Problem	9	[-8192, 8192]	1
2	Inverse Hilbert Matrix Problem	16	[-16384, 16384]	1
3	Lennard-Jones Minimum Energy Cluster	18	[-4, 4]	1
4	Rastrigin's Function	10	[-100, 100]	1
5	Griewank's Function	10	[-100, 100]	1
6	Weierstrass Function	10	[-100, 100]	1
7	Modified Schwefel's Function	10	[-100, 100]	1
8	Expanded Schaffer's F6 Function	10	[-100, 100]	1
9	Happy Cat Function	10	[-100, 100]	1
10	Ackley Function	10	[-100, 100]	1

Tablo 3. KAA 'nın CEC-C06 2019 test fonksiyonları üzerindeki performans sonuçları ($fl=2$)

Fonksiyon No	En iyi	En kötü	Ortalama	Standart Sapma
CEC01	1,99E+06	4,97E+08	8,57E+07	1,15E+08
CEC02	7,75E+03	2,41E+04	1,66E+04	4,69E+03
CEC03	1,27E+01	1,27E+01	1,27E+01	9,98E-04
CEC04	1,23E+04	5,21E+04	2,47E+04	8,20E+03
CEC05	3,97E+00	1,04E+01	7,03E+00	1,62E+00
CEC06	1,33E+01	1,68E+01	1,51E+01	9,75E-01
CEC07	1,38E+03	2,72E+03	1,99E+03	3,06E+02
CEC08	6,68E+00	9,11E+00	8,01E+00	5,61E-01
CEC09	2,81E+03	8,87E+03	5,47E+03	1,45E+03
CEC10	2,06E+01	2,14E+01	2,11E+01	1,66E-01

Tablo 4. KAA 'nın CEC-C06 2019 test fonksiyonları üzerindeki performans sonuçları ($fl=4$)

Fonksiyon No	En iyi	En kötü	Ortalama	Standart Sapma
CEC01	4,61E+06	5,04E+08	1,37E+08	1,58E+08
CEC02	1,00E+04	2,19E+04	1,54E+04	3,31E+03
CEC03	1,27E+01	1,27E+01	1,27E+01	8,24E-04
CEC04	7,47E+03	4,20E+04	2,58E+04	8,13E+03
CEC05	4,09E+00	9,63E+00	7,01E+00	1,35E+00
CEC06	1,43E+01	1,73E+01	1,55E+01	7,99E-01
CEC07	1,05E+03	2,50E+03	1,92E+03	4,20E+02
CEC08	6,80E+00	8,88E+00	8,07E+00	5,13E-01
CEC09	3,43E+03	9,90E+03	5,71E+03	1,49E+03
CEC10	2,06E+01	2,13E+01	2,10E+01	1,58E-01

Tablo 5. KAA 'nın CEC-C06 2019 test fonksiyonları üzerindeki performans sonuçları ($fl=6$)

Fonksiyon No	En iyi	En kötü	Ortalama	Standart Sapma
CEC01	1,15E+08	2,98E+09	8,18E+08	7,09E+08
CEC02	6,86E+03	2,19E+04	1,34E+04	3,96E+03
CEC03	1,27E+01	1,27E+01	1,27E+01	6,65E-04

CEC04	8,94E+03	3,70E+04	2,40E+04	7,57E+03
CEC05	4,46E+00	1,20E+01	7,14E+00	1,77E+00
CEC06	1,38E+01	1,76E+01	1,58E+01	1,06E+00
CEC07	1,61E+03	2,54E+03	2,01E+03	2,76E+02
CEC08	7,54E+00	8,98E+00	8,38E+00	3,76E-01
CEC09	2,50E+03	8,66E+03	5,70E+03	1,54E+03
CEC10	2,08E+01	2,13E+01	2,11E+01	1,47E-01

Tablo 6. KAA 'nın CEC-C06 2019 test fonksiyonları üzerindeki performans sonuçları ($fl=8$)

Fonksiyon No	En iyi	En kötü	Ortalama	Standart Sapma
CEC01	4,68E+08	6,27E+09	1,90E+09	1,32E+09
CEC02	5,98E+03	2,11E+04	1,35E+04	4,41E+03
CEC03	1,27E+01	1,27E+01	1,27E+01	1,03E-03
CEC04	9,10E+03	3,90E+04	2,56E+04	8,00E+03
CEC05	4,77E+00	1,06E+01	7,58E+00	1,38E+00
CEC06	1,11E+01	1,73E+01	1,52E+01	1,43E+00
CEC07	1,43E+03	2,97E+03	2,09E+03	3,61E+02
CEC08	7,83E+00	9,39E+00	8,48E+00	4,80E-01
CEC09	3,06E+03	7,89E+03	5,61E+03	1,07E+03
CEC10	2,07E+01	2,14E+01	2,11E+01	1,55E-01

Tablo 7. KAA 'nın CEC-C06 2019 test fonksiyonları üzerindeki performans sonuçları ($fl=10$)

Fonksiyon No	En iyi	En kötü	Ortalama	Standart Sapma
CEC01	5,10E+08	1,09E+10	3,24E+09	2,71E+09
CEC02	8,03E+03	2,50E+04	1,40E+04	3,66E+03
CEC03	1,27E+01	1,27E+01	1,27E+01	1,09E-03
CEC04	1,05E+04	4,22E+04	2,56E+04	8,75E+03
CEC05	3,95E+00	9,96E+00	7,23E+00	1,59E+00
CEC06	1,17E+01	1,73E+01	1,56E+01	1,33E+00
CEC07	1,51E+03	2,52E+03	2,15E+03	2,77E+02
CEC08	7,20E+00	8,63E+00	8,01E+00	4,15E-01
CEC09	3,13E+03	8,10E+03	5,45E+03	1,39E+03
CEC10	2,07E+01	2,14E+01	2,11E+01	1,49E-01

Tablo 8. KAA 'nın farklı fl değerlerine göre ortalama sonuçlarının karşılaştırması

Fonksiyon No	$fl=2$	$fl=4$	$fl=6$	$fl=8$	$fl=10$
CEC01	8,57E+07	1,37E+08	8,18E+08	1,90E+09	3,24E+09
CEC02	1,66E+04	1,54E+04	1,34E+04	1,35E+04	1,40E+04
CEC03	1,27E+01	1,27E+01	1,27E+01	1,27E+01	1,27E+01
CEC04	2,47E+04	2,58E+04	2,40E+04	2,56E+04	2,56E+04
CEC05	7,03E+00	7,01E+00	7,14E+00	7,58E+00	7,23E+00
CEC06	1,51E+01	1,55E+01	1,58E+01	1,52E+01	1,56E+01
CEC07	1,99E+03	1,92E+03	2,01E+03	2,09E+03	2,15E+03
CEC08	8,01E+00	8,07E+00	8,38E+00	8,48E+00	8,01E+00
CEC09	5,47E+03	5,71E+03	5,70E+03	5,61E+03	5,45E+03
CEC10	2,11E+01	2,10E+01	2,11E+01	2,11E+01	2,11E+01

Tablo 9. IKAAs'nın ve diğer metasezgisel algoritmaların ortalama karşılaştırma sonuçları

Fonksiyon No	DA	WOA	KAA
CEC01	543×10 ⁸	411×10 ⁸	1,37E+08
CEC02	78.0368	17.3495	1,54E+04
CEC03	13.7026	13.7024	1,27E+01
CEC04	344.3561	394.6754	2,58E+04
CEC05	2.5572	2.7342	7,01E+00
CEC06	9.8955	10.7085	1,55E+01
CEC07	578.9531	490.6843	1,92E+03
CEC08	6.8734	6.909	8,07E+00
CEC09	6.0467	5.9371	5,71E+03
CEC10	21.2604	21.2761	2,10E+01

Tablo 10. IKAAs'nın ve diğer metasezgisel algoritmaların standart sapma karşılaştırma sonuçları

Fonksiyon No	DA	WOA	KAA
CEC01	669×10 ⁸	542×10 ⁸	1,58E+08
CEC02	87.7888	0.0045	3,31E+03
CEC03	0.0007	0.0	8,24E-04
CEC04	414.0982	248.5627	8,13E+03
CEC05	0.3245	0.2917	1,35E+00
CEC06	1.6404	1.0325	7,99E-01
CEC07	329.3983	194.8318	4,20E+02
CEC08	0.5015	0.4269	5,13E-01
CEC09	2.871	1.6566	1,49E+03
CEC10	0.1715	0.1111	1,58E-01

4. Sonuç

Mevcut eğilim, zor problemlerin üstesinden gelmek için doğadan ilham alan metasezgisel algoritmaları kullanmaktır. Karga Arama Algoritması (KAA) da bu tarz algoritmalarından bir tanesidir. Bir karga sürüsü, optimizasyon süreciyle pek çok benzerliği olan bir davranış sergilerler. Bu davranışa göre kargalar fazla yiyeceklerini ortamın belirli konumlarında (saklanma yerlerinde) saklarlar ve depolanan yiyecekleri gerektiğinde geri bulurlar. Bu

davranıştan esinlenilerek KAA oluşturulmuştur. KAA sürekli optimizasyon problemlerini çözsede detaylı bir parametre analizi yapılmamıştır. Bu çalışmada *fl* parametre değerinin arama kabiliyeti üzerindeki etkisi araştırılmıştır. Çünkü küçük *fl* değerleri yerel aramaya yol açarken büyük değerler genel aramaya yol açar. Beş farklı *fl* değerinin on farklı CEC 2019 kıyaslama fonksiyonu üzerinde testleri gerçekleştirilmiştir. En başarılı *fl* değeri dört olarak tespit edilmiştir. Bu da göstermiştir ki yerel arama kabiliyeti performans üzerindeki etkisi daha fazladır. En başarılı KAA sonuçları literatürden seçilen DA ve WOA algoritmaları ile karşılaştırılmıştır. Sonuç olarak KAA etkili algoritmalarından biri olduğu ispatlanmıştır.

Geleceğe dair çalışmalarda KAA'nın arama yeteneğinin farklı teknikler kullanılarak geliştirilmesi düşünülmektedir. CEC 2017 test fonksiyonlarında başarısı gösterilmesi düşünülmektedir.

Kaynakça

- Abdullah, J. M., Rashid, A.T. "Fitness Dependent Optimizer: Inspired by the Bee Swarming Reproductive Process," in *IEEE Access*, vol. 7, pp. 43473-43486, 2019, doi: 10.1109/ACCESS.2019.2907012.
- Askarzadeh, A., 2016. A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm, *Computers and Structures*, 169, 1-12.
- Mirjalili, S., "Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. Appl.*, vol. 27, no. 4, pp. 1053–1073, May 2015.
- Mirjalili, S., Lewis, A., "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016.
- Price, K. V., Awad, N. H., Ali, M. Z., Suganthan, P. N., "The 100-digit challenge: Problem definitions and evaluation criteria for the 100-digit challenge special session and competition on single objective numerical optimization," *School Elect. Electron. Eng., Nanyang Technol. Univ., Singapore, Tech. Rep.*, Nov. 2018.
- Yang, X.S, 2011. Metaheuristic optimization, *Scholarpedia* 2011; 6 11472.