



# Düzce Üniversitesi Bilim ve Teknoloji Dergisi

*Araştırma Makalesi*

## Doğadan Esinlenen Optimizasyon Algoritmaları ve Optimizasyon Algoritmalarının Optimizasyonu

Pakize ERDOĞMUŞ

*Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi, Düzce Üniversitesi, Düzce, TÜRKİYE  
pakizeerdogmus@duzce.edu.tr*

### ÖZET

Matematiksel olarak optimizasyon, bir yada daha fazla bağımsız değişkene sahip olan bir fonksiyonun verilen kısıtlayıcı şartlar altındaki en iyi çözümünü arama işlemi olarak tanımlanabilir. Optimizasyon problemlerini çözmek için sezgisel optimizasyon algoritmaları ve klasik çözüm yöntemleri mevcuttur. Ancak son yıllarda özellikle doğal süreçlerden esinlenilmiş birçok optimizasyon algoritması geliştirilmiştir. Doğadaki birçok canlı sahip oldukları kusursuz tasarımlarıyla var olan kaynakları minimum kullanarak, maksimum başarımla elde etmeyi başararak en zorlu şartlarda bile hayatta kalmayı başarmıştır. Canlıların koordineli hareket ederek özellikle yiyecek bulmada gösterdikleri zeka, “sürü zekası”(swarm intelligence) olarak adlandırılmaktadır. Bu çalışmada özellikle son yıllarda geliştirilen ve doğadan esinlenen optimizasyon algoritmaları, bu algoritmaların uygulama alanları ve başarımları analizleri incelenmiştir. Bu algoritmaların optimum sonucu verebilmesinin algoritmaların optimum parametrelerle çalıştırılmasına bağlı olduğu ortaya konulmuştur.

**Anahtar Kelimeler:** *Sezgisel Araştırma, Doğadan Esinlenen Algoritmalar, Optimizasyon*

## Nature Inspired Optimization Algorithms and Optimization of the Optimization Algorithms

### ABSTRACT

Optimization is defined as a process of searching the best solution of a function which has one or more variable under some constraints. There are classical solution methods and heuristic optimization algorithms in order to solve optimization problems. Recently, it has been developed a lot of optimization algorithms inspired from nature. Most of the living being in the nature has been successful to survive perfect designs accomplishing to take maximum performance from minimum resource. The cooperative behavior of animals for finding foods is called swarm intelligence. In this study, recently developed optimization algorithms especially inspired from nature have been researched. It has been shown that for these algorithms in order to give optimum results is dependent to the parameters used in the algorithms.

**Keywords:** *Heuristic Search, Nature-Inspired Algorithms, Optimization*

## I. GİRİŞ

**G**ÜNÜMÜZDE bilim ve teknoloji her gün artan bir ivme ile gelişmesine devam etmektedir. Optimizasyon (en iyileme) bu gelişmelerin birçoğunda karşılaşılan problemlerin başında gelmektedir. Optimizasyon, en büyükleme ve en küçükleme işlemlerinin genel bir ismi olup, işletmelerden, ekonomiye, tasarımdan, üretime hemen her bilim dalını ilgilendirir. İşletmelerde karın en iyilenmesi veya maliyetin en aza indirgenmesi, tasarım mühendisliğinde un uygun boyutlarda cihaz üretimi, en minimum enerji kullanan cihazların tasarımı veya mühendislikte üretilen bir motorun çevreye yaydığı zararlı gazların minimizasyonu hedefleri birer optimizasyon işlemidir. Optimizasyon problemlerinin çözümünde kullanılan klasik yöntemler, gerçek hayat problemlerini çözmeye yetersiz kaldığı, çözüm sürelerinin ve çözüm karmaşıklığının problem boyutuyla orantılı olarak çok artması, sezgisel araştırma yöntemlerinin gelişmesine yol açmıştır. Sezgisel araştırma yöntemleri ise bilgisayarın gelişimine paralel olarak son kırk yıldır çalışılmaktadır. Tüm canlı türlerinin optimum bir tasarıma sahip olarak mükemmel bir şekilde ekosistemde yer alması, davranışlarında ve yiyecek arama davranışlarında optimum çözümü bulmaları bilim adamlarının araştırmalarına konu olmakta ve doğadan esinlenen algoritmaların artışı ile sonuçlanmaktadır.

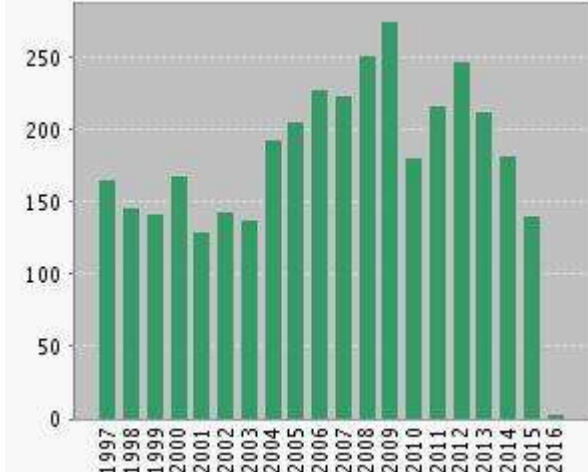
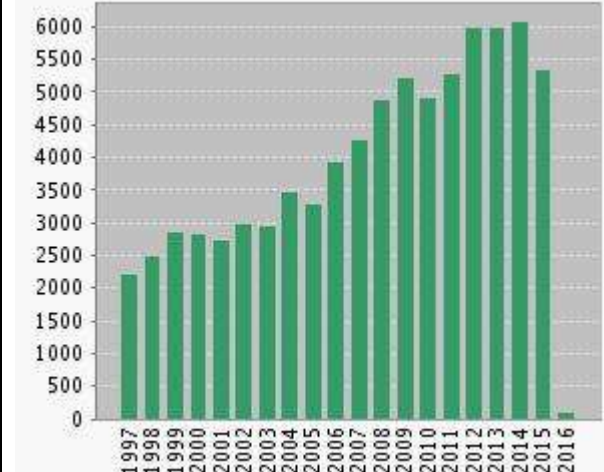
1970'li yıllardan itibaren gelişmeye başlayan sezgisel araştırma yöntemlerinin birçoğu doğadan esinlenilerek geliştirilmiştir. Bu yöntemlerin bazıları doğadaki fiziksel yasaları veya süreçleri temel alırken, büyük bir kısmı canlı varlıkların sosyal etkileşimlerini örnek almıştır.

Genetik Algoritma(Genetic Algorithm)(GA)[1][2] geliştirilen ilk ve en çok çalışılan sezgisel algoritmalarından biridir. 1976 yılında Holland tarafından geliştirilen, “En iyi olan hayatta kalır.” felsefesine dayalı çözüm yapan bir algoritmadır. Bir başlangıç popülasyonu ile çözüme başlanır. Bu popülasyon içerisinde en iyi bireyler seçilerek bu bireylerden çaprazlama, mutasyon ve elitizm işlemleri ile bir sonraki popülasyon üretilir. Nesilden nesile bireyler en iyi olana doğru hareket eder. Çözümlerin ikili kodlar ile oluşturulması ve çaprazlama işlemi ile çözüm uzayının çok iyi taranması genetik algoritmanın önemli bir üstünlüğüdür. Web of Science’de araştırıldığında Genetic Algoritma konusunda 1975-2016 yılları arasında 32 405[3] adet çalışma yapıldığı görülmektedir. Holland ve Goldberg’dan sonra Genetik Algoritma konusunda en çok atıf alan yayın çok amaçlı genetik algoritma 7927 kez atıf almıştır. Genetik algoritma konusunda yapılan çalışmaların bir kısmı algoritmanın kendine yönelik iyileştirme çalışmalarıdır. Bu çalışmalar da genellikle adaptiflik, diğer bir optimizasyon algoritması ile hibritleme, çaprazlama ve mutasyon operatörlerinde yenilik, kendi operatörlerinin optimum değerlerinin araştırılması ele alınmıştır. Diğer çalışmalarda ise hem kesikli hem de sürekli optimizasyon problemlerinin çözümünde kullanılmıştır. Çizelgeleme problemleri, gezgin satıcı problemi, araç rotalama problemi, astronomi ve astrofizik, elektromanyetik, yapay sinir ağları eğitimi, elektrik yük dağıtım problemleri Genetik Algoritma ile çözülmüştür.

Benzetim Tavlama(Simulated Annealing)(SA)[4] optimizasyon problemleri için Monte-Carlo tekniğini kullanarak en iyi çözümleri bulan olasılıklı bir arama yöntemi olup, 1983 yılında Kirkpatrick tarafından geliştirilmiştir. Tavlama, katıların fiziksel tavlama süreci ile olan benzerlikten ileri gelmektedir. Yüksek dayanımlı metallerin kristallerinin düzgün bir şekilde oluşması için, metaller çok yüksek sıcaklıklara kadar eritilirler ve daha sonra yavaş yavaş soğutulurlar. SA bu yaklaşımı optimizasyona uygulayan bir tekniktir. Yüksek sıcaklıklarda yani algoritmanın başlarında küresel en iyiyi bulmak ve çözüm uzayını iyi araştırmak için belirli bir olasılıkla kötü çözümlere de izin verilir. Sıcaklık azaldıkça algoritma yerel arama metodu gibi çalışmaya başlar. SA algoritmasında hem kesikli(combinatorial) hem de sürekli problemlere başarı ile uygulanmıştır. İş çizelgeleme, darboğaz

atama, gezgin satıcı, lojistik gibi endüstri problemlerine ve görüntü işlemeden, yapay sinir ağlarına kadar bir çok alandaki mühendislik problemine başarı ile uygulanmış ve iyi sonuçlar vermiş bir algoritmadır. Aşağıda Tablo 1’de SA algoritmasının atıf istatistikleri görülmektedir.

**Tablo 1. SA algoritmasının atıf sonuçları**

Her yıl yayınlanan makale sayısı(Son yirmi yıl)	Her yıl yapılan atıf sayısı
	

Tablo 1’de görüleceği üzere SA algoritması ile yapılan yayın sayısı azalsa da atıf sayısı 2015’e kadar artış göstermiştir. Sağlam bir algoritma olması sebebi ile günümüzde de birçok optimizasyon probleminin çözümünde başarı ile kullanılmaktadır.

Doğadan esinlenen algoritmalarından biri de Karınca Kolonisi Optimizasyonu(Ant Colony Optimizasyonu)’ dur [5]. 1993 yılında Dorigo tarafından geliştirilmiştir. Dorigo bu algoritmasını doktora tezinde geliştirmiştir [6]. Karıncaların yiyeceğe giden en kısa yolu buldukları ve bunu da birbirleri arasında iletişim aracı olarak kullandıkları kimyasal bir madde olan feromon salgısı ile gerçekleştirdikleri görülmüştür. İlk anda olasılığa dayalı olan karınca hareketleri yuvadan yiyeceğe doğru kısa olan yolda zaman içinde artmakta bu da diğer karıncaların da bu yolu takip ederek en kısa zamanda en optimum çözümü bulmalarına sebep olmaktadır. Dorigo, karıncaların sürü zekası ile yiyeceğe giden en kısa yolu bulmalarını optimizasyon problemlerinin çözümüne uyarlamıştır [7]. Başta Gezgin Satıcı olmak üzere bir çok ayrı optimizasyon problemlerinin çözümünde başarılı uygulamaları olan ACO algoritması adapte edilerek sürekli optimizasyon problemlerinin çözümünde de kullanılmış ve başarılı çözümler ürettiği görülmüştür [8].

Parçacık Sürü Optimizasyonu(Particle Swarm Optimization)(PSO) [9] kuş ve balık sürülerinin yiyecek arama davranışını simüle eden bir algoritmadır. Eberhart ve Kenedy tarafından 1995 yılında sürekli optimizasyon problemleri için geliştirilen bir algoritmadır. Problem bir başlangıç popülasyonu ile harekete başlar. Populasyondaki her bir birey bir parçacık(particle) olarak adlandırılır. Algoritmanın başında parçacıklar rastgele hızlarla ve rastgele başlangıç konumları ile çözüme başlar. Problemin çözümleri, parçacıkların uzaydaki konumlarına karşı gelmektedir. Parçacıklar hem kendi en iyilerini(personel best) hemde popülasyondaki en iyi(küresel best) çözüme göre hızlarını değiştirerek farklı konumlara ulaşırlar. Genetik algoritma gibi popülasyon temelli bir algoritma olmasına rağmen çaprazlama ve mutasyon gibi operatörlere sahip olmamasından dolayı daha hızlı bir algoritmadır. Ancak hızlı yakınsaması bazen küresel optimum değeri bulmaması ile sonuçlanabilir. PSO algoritması ile yapılan çalışmalarda adaptiflik, bulanıklık, kaos, evrimsellik gibi yeniliklerle algoritmanın başarımı arttırılmaya çalışılmıştır. PID(Propotional-Integral-Differential) kontrol,

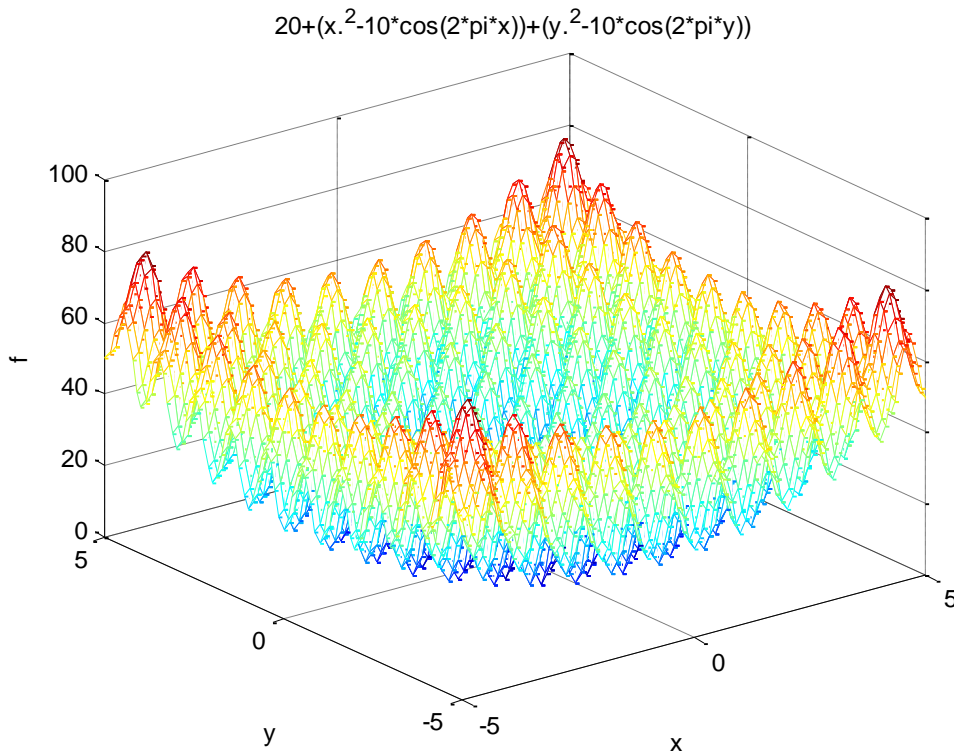
elektrik güç dağıtımı, Güç akış problemleri ve birçok sürekli optimizasyon probleminin çözümünde kullanılmıştır.

Karaboğa tarafından 2004 yılında geliştirilen Yapay Arı Kolonisi(Artificial Bee Colony)(ABC)[10], arı kolonilerinin yiyecek kaynağı arama davranışları modellenerek geliştirilmiştir. Kolonideki arılar kâşif, işçi ve gözcü olmak üzere üç gruba ayrılırlar. Bu üç grup arı negatif geri besleme, pozitif geri besleme ve rastgele hareketleri ifade eder. Yiyecek kaynağının konumu değişken değerlerini amaç fonksiyonu ise nektar açısından zenginliğini ifade eder.

ABC arıların yiyecek bulma davranışlarını simüle etmiş ve kablosuz ağ optimizasyonu, sayısal filtre optimizasyonu, elektrik güç sistemleri optimizasyonu ve daha birçok mühendislik problemine çözüm getirmiştir. Özellikle çok değişkenli kısıtsız optimizasyon test problemlerinde GA ve Diferansiyel Gelişim Algoritmasından(Differential Evolution Algorithm) daha iyi sonuçlar verdiği görülmüştür [11].

Yukarıda sayılan algoritmalar doğadan esinlenen algoritmaların atası sayılabilir ve bu algoritmalar genellikle bir optimizasyon probleminin çözümüne yönelik geliştirilmiş algoritmalar. Genetik Algoritmalar küresel optimizasyon amacı ile geliştirilirken, Diferansiyel Gelişim Algoritması Chebichev polinom katsayılarını bulma amaçlı geliştirilmiştir. Karınca Kolonisi Optimizasyon Algoritması ise önceleri Karınca Sistemi(Ant System) adı ile kesikli optimizasyon problemi olan bilgisayar ağ problemlerinin çözümü için geliştirilmiştir. Günümüzde ise sezgisel optimizasyon algoritmalarının kendileri başlı başına bir çalışma alanı oluşturmuştur. Bu algoritmaların test edilmesi için de bir dizi kısıtsız basit, kısıtsız karmaşık ve kısıtlı optimizasyon test problemleri geliştirilmiştir[12].

Şekil 1.'de bu test problemlerinden kısıtsız bir fonksiyon olan Rastrigin fonksiyonu görülmektedir. Görüldüğü gibi fonksiyonun (0,0) noktasında bir adet küresel minimum değeri var iken çok sayıda yerel minimum ve maksimum değerleri vardır.



Şekil 1. Rastrigin Test Fonksiyonu

Geliştirilen algoritmaların başarımları bu test problemlerinin çözüme başarılarına göre karşılaştırılmaktadır. Karşılaştırmalar genellikle probleme göre çözüm süreleri ve problemin optimum noktasına yakınlıklarına göre olmaktadır. Ancak bu sezgisel yöntemlerin her birinin kendine özel parametreleri vardır ve bu parametrelere göre algoritmaların başarımları tamamen değişmektedir.

Geliştirilen sezgisel algoritmaların başarımları test problemleri üzerinde sınılandıktan sonra algoritmanın başarımlarını arttırmaya dayalı çalışmalar yapılmaktadır. En çok karşılaşılan durumlar, temel algoritmada sabit olan bir parametrenin adaptifleştirilmesi [13,14,15,16,17,18,19,20], yeni bir parametre eklenmesi [21], bir başka sezgisel yöntemin güçlü olan özelliği kullanılarak yeni hibrit bir algoritma geliştirilmesi [22,23,24,25,26], kaos[27,28,29,30,31,32] veya bulanık mantık [33,34] ile yeni bir yöntemin geliştirilmesi sayılabilir. Başarımları arttırılmış olan algoritma bundan sonra bilimsel alanda karşılaşılan optimizasyon problemlerinin çözümünde kullanılmaktadırlar. En çok kullanılan alanların başında Yapay Sinir Ağlarının Eğitimi [35,36,37], Kümeleme [38,39,40,41,42,43,44], tanıma problemleri gelmektedir. Bir başka alan ise son yıllarda Paralel programlama olmuştur [45,46,47]. Özellikle popülasyon temelli algoritmalarda her bir çözüm bir iş parçacığı olarak ele alınarak, çözüm yapılmaktadır.

Çalışmanın ikinci bölümünde literatürde karşılaşılan doğadan esinlenen sezgisel optimizasyon algoritmaları üç alt başlık altında incelenmiştir. Üçüncü bölümünde ise bu algoritmalarda yapılan başarımların analizleri üzerine inceleme yapılmış ve sonuç bölümünde öneriler sunulmuştur.

## II. DOĞADAN ESİNLENİLEN ALGORİTMALAR

Bu çalışmada doğadan esinlenen sezgisel optimizasyon algoritmaları, fiziksel süreçler ve yasalardan esinlenen algoritmalar, doğal olaylardan esinlenen algoritmalar ve yiyecek arama davranışından esinlenen algoritmalar olarak üç grupta incelenmiştir.

### A. FİZİKSEL SÜREÇLER VE YASALARDAN ESİNLENEN ALGORİTMALAR

Benzetim Tavlama( Simulated Annealing)(SA)[4], Yerçekimi Arama Algoritması(Gravity Search Algorithm)(GSA) [48],Gaz Brownian Hareketi Optimizasyonu( Gases Brownian Motion Optimization: an Algorithm for Optimization (GBMO)) [49], Isı Transferi Arama ( Heat transfer search(HTS)) [50], Electromagnetic field optimization[51], Optic Inspired Optimization(OIO) [52], Ağırlıklı Süperpozisyon Çekimi (Weighted Superposition Attraction (WSA) [53,54] gibi bir kısım algoritmalar fiziksel süreçler ve yasalardan esinlenilerek geliştirilmiştir. Bunlardan en eskisi olan SA hala en iyi çözüm üreten algoritmalarından biridir. Aşağıda Tablo 2’de Web of Science’de, yapılan arama sonucu bulunan fiziksel süreç ve yasalardan yararlanan algoritmaların temel çalışmasının geliştirildiği yıl ve atıf sayıları görülmektedir.

**Tablo 2. Fiziksel Süreçleri ve Yasaları Temel Alan Algoritmalar**

Sezgisel Algoritma	Geliştirildiği Yıl ve Ay	Atıf Sayısı*
Benzetim Tavlama(SA)[4]	1983-Mayıs	15,147
Yerçekimi Arama[48]	2009-Haziran	546
Gaz Brownian Hareketi[49]	2013-May	3
Isı Transfer Arama[50]	2015-Aralık	0
Elektromanyetik Alan Optimizasyonu[51]	2015-August	0
Optikten Esinlenen Optimizasyon[52]	2015-Aralık	0
Ağırlıklı Süperpozisyon Çekimi[53,54]	2015-	1

\*Makale başlığına göre Web of Science’de arama yapılmıştır.

SA yerel optimumdan kaçınmak için yokuş yukarı(hill-climbing) hareketlere izin vererek küresel optimum çözümü bulmayı hedefler. Temel parametreleri sıcaklık ve soğutma planı parametreleridir. Sıcaklık yeterince yüksek seçilmez ise araştırma uzayı tam olarak incelenemez. Her sıcaklıkta belli iterasyon sayısınca uzayın araştırılması ve sıcaklığın yavaş yavaş azaltılması SA'nın çözüm başarısına etki eder.

Yerçekimi Araştırma Algoritması(Gravitiy Search Algorithm)(GSA) ise Newton'un yerçekimi yasasından esinlenilerek geliştirilmiştir. Arama uzayındaki her bir çözüm yapay bir kütle olarak düşünülür. Arama uzayındaki kütleler Newton'un yerçekimi yasasına göre birbirlerini çekerler ve birbirlerine kuvvet uygularlar. Yerçekimi kuvvetine maruz kalan kütleler arama uzayı içerisinde hareket ederek en optimum sonuca ulaşırlar [48]. Yerçekimi sabitinin başlangıç değeri ve değişim katsayısı ile maksimum iterasyon sayısı bu algorithmada kullanılan parametrelerdir. Yerçekimi sabiti gittikçe azalan bir değer olup, SA'daki sıcaklığa bağlı olasılık değerine çok benzer.

Gaz Brownian Algoritması ise  $PV=nRT$  şeklindeki İdeal Gaz Yasası, Gaz Brownian Hareketi ve rotasyonel gaz hareketlerini temel alan bir algoritmadır. Gazların küçük parçacıkları sürekli çözüm uzayında birer çözümü temsil eder. Rastgele konumlardaki gaz molekülleri rastgele rotasyonel yarıçaplarla belirli bir sıcaklıkta harekete başlarlar. Gaz moleküllerinin hızları sıcaklıkla değişir. GBMA algoritmasındaki sıcaklık parametresi SA algoritmasındaki ile aynı işleve sahiptir. Yüksek sıcaklıklarda araştırma uzayının iyice taranmasını, düşük sıcaklıklarda ise yerel optimum noktanın detaylı incelenmesini sağlar [49].

HTS algoritmasının temel dayanağı ise termodinamiğin temel kanunlarından birisi olan “Bir sistem çevresi ile denge durumunda olmaya çalışır” prensibidir [50]. Bu denge bir problemin optimum çözümü olarak ele alınmaktadır.

Elektromanyetik alan optimizasyonu ise elektromıknatıslığa dayalı bir popülasyon temelli algoritmadır. İçinden akım geçen bir tel, üzerinde bir elektromanyetik alan oluşur. Aynı yüklü elektromıknatıslar birbirini iter ve zıt yöndekiler çeker. Ancak çekim kuvveti itme kuvvetinden %5-10 oranında daha fazladır. Algorithmada ise bu çekme ve itme kuvveti altın oran kullanılarak dengelenmektedir [51]. Algorithmada uygunluk fonksiyonu en iyi olan çözümler pozitif yüklü ve en kötü olanlar ise negatif yüklü olarak kabul edilir.

OIO algoritması ise iç bükey ve dış bükey ayna özelliklerini fonksiyon optimizasyonuna uyarlamış bir algoritmadır. İç bükey ayna ışığı toplar, dış bükey ayna ise ışığı dağıtır. Algoritmanın başlangıç çözümleri yapay nesnelere kabul edilir. Yapay ışık gönderilerek fonksiyon yüzeyinin iç bükey veya dışbükey olma durumuna göre yeni çözümler(yapay görüntüler) ayna kanunları kullanılarak üretilir[52].

WSA algoritması ise ağırlıklı süperpozisyon kuralı ile ajanların çekim hareketlerini birleştirerek hem kısıtsız hem de kısıtsız optimizasyon problemlerini çözmede başarılı sonuçlar elde etmişlerdir. Algoritma kısıtsız ve kısıtlı problemler için başarılı sonuçlar vermektedir [53,54].

## B. DOĞAL OLAYLARDAN VE DOĞADAN ESİNLENEN ALGORİTMALAR

Aşağıda Tablo 3’de Web of Science’de, yapılan arama sonucu bulunan doğal olaylardan ve doğadan esinlenen algoritmaların temel çalışmasının geliştirildiği yıl ve atıf sayıları görülmektedir.

*Tablo 3. Doğal Olaylardan Ve Doğadan Esinlenen Algoritmalar*

<b>Sezgisel Algoritma</b>	<b>Geliştirildiği Yıl ve Ay</b>	<b>Atıf Sayısı*</b>
Orman Optimizasyonu Algoritması[55]	2014-Kasım	2
Kasırğa Temelli Optimizasyon Algoritması[56]	2014-	0
Su Damlası Algoritması[57]	2012-	15
Kara Delik Optimizasyon Algoritması[58]	2013-Şubat	42
Su Döngüsü Optimizasyon Algoritması[59]	2012-Kasım	30
Yıldırım Arama Algoritması[60]	2015-Kasım	0

Orman Optimizasyonu Algoritması(Forest Optimization Algorithm)(FOA)[55] algoritması ise en iyi olan hayatta kalır prensibinin ağaçlara uygulanması ile geliştirilmiştir. Sürekli ve doğrusal olmayan problemlerin çözümü için geliştirilmişlerdir. Ormanlık bir alanda birçok ağaç belli bir zaman geçince çürüyüp yok olmasına rağmen, birkaç ağaç, aradan uzun yıllar geçmesine rağmen hayatta kalmaktadır. Bu da tohumlarının uygun yerlerde olması ile açıklanmaktadır. Algoritmada yapay ağaçlar üretilerek, tohumları çözüm uzayına saçılır. Bu algoritma test problemleri dışında gerçek problemlere de uygulanarak sonuçları sunulmuştur.

Kasırğa Temelli Optimizasyon Algoritması(Hurricane Based Optimization Algorithm)[56] kasırğa oluşumunu modelleyerek (Rankine vortex model) ile problem çözümünde kullanmışlardır. Akan su damlacıklarının muazzam sürüler şeklinde hareket halinde oldukları gözlenir. Doğal ırmakların yatakları da bu su damlacıkları sürüsü tarafından oluşturulur. Her bir damlacık bir miktar toprak taşır[57].

Kara Delik Optimizasyon Algoritması[58] bir ilk çözüm ile başlar. Bu ilk çözümler yıldızlara karşı düşer. Kara delikler ise konumları sabit olup fonksiyonun minimum noktalarıdır. Algoritma ilerledikçe yıldızlar karadeliğe doğru hareket ederler. Bu arada kara delikten daha minimum değere sahip bir değer elde edildi ise kara delik güncellenir.

Su Döngüsü Optimizasyon Algoritması[59] nehir ve ırmakların denizlere doğru hareketinden ilham almıştır.

## C. CANLI VARLIKLARDAN ESİNLENEN ALGORİTMALAR

Aşağıda Tablo 4’de Web of Science’de, yapılan arama sonucu bulunan canlı varlıklardan esinlenen algoritmaların temel çalışmasının geliştirildiği yıl ve atıf sayıları görülmektedir.

**Tablo 4. Canlı Varlıklardan Esinlenen Algoritmalar**

<b>Sezgisel Algoritma</b>	<b>Geliştirildiği ve Ay</b>	<b>Yıl</b>	<b>Atf Sayısı*</b>
Karınca Sistemi[61]	1996-Şubat		4,097
Max-Min Ant System[62]	2000-Haziran		1109
Karınca Kolonisi Optimizasyonu[7]	2005-Kasım		587
Parçacık Sürü Optimizasyonu[9]	1995-Aralık		5721
Parçacık Sürü Optimizasyonu[63]	2002-Şubat		3331
Yapay Arı Kolonisi[64]	2007-Kasım		1023
Meyve Sineği Optimizasyon Algoritması[65]	2012-Şubat		126
Levy Uçuşuna dayalı Guguk kuşu optimizasyonu[66]	2009-		369
Krill Sürü Optimizasyon Algoritması[67]	Aralık-2012		121
Bakeri Yiyecek Arama Davranışı[68]	2002-Haziran		733
Yarasa Algoritması[69]	2010		217
Ateş Böceği Algoritması[70]	2009		389
Aslan Algoritması[71]	2012-		1
Gri Kurt Algoritması[72]	2014-Mart		52
Yunus Balığı[73]	2013-Mayıs		15
Çalı Kolonisi Algoritması[74]	2006-Aralık		188
Yapay Alg Algoritması[75]	2015-Mart		1
Virüs Koloni Arama Algoritması[76]	2016-Ocak		0
Köpekbalığı Koku Alma Optimizasyon Algoritması[77]	2014-Aralık		1
Sosyal Örümcek Algoritması[78]	2015-Mayıs		2

Yukarıda ismi anılan algoritmalar sosyal temelli algoritmalar olarak da anılmaktadır. Tüm bu canlılar bilgisayarlara problem çözmeye zeka vermesi için kullanılmaktadır. Herbir canlı türü de kendine özel bir yeteneğe sahiptir. Bu canlıların başarısı sahip oldukları yeteneği kullanarak bireysel olarak edindikleri tecrübeleri sürüye aktarmak ve böylece sosyal olarak en iyiyi bulmalarınıdır. Karıncaların feromon salgısı, guguk kuşlarının diğer kuş türlerinin yumurtasına benzer yumurtlaması, balık ve kuş türlerinin sürüden kopmadan topluca hareketleri bu yeteneklerden bir kaçıdır. Bu açıdan düşünülecek olursa herbir canlı türü veya her bir doğal olay veya kanun içinde bir optimizasyon yeteneği barındırmaktadır. Bu sebeple ilerleyen yıllarda da doğadan esinlenen optimizasyon yöntemleri geliştirilecektir. Bu algoritmalar sürü zekası olarak adlandırılan bir ortak akıl ile hareket ederek bir çok alandaki çözümü zor hatta klasik yöntemlerle çözümü imkansız olan problemleri başarı ile çözmektedir. Bu algoritmalar Bilgisayar Destekli Tasarım, Güç Sistemleri, Manyetik Alan, Makine Öğrenmesi, Yapay Sinir Ağları, Görüntü İşleme, VLSI, Doğal Hesaplama, Sinyal İşleme, Paralel Hesaplama, Kümeleme, Kenar Belirleme, Reaktif Güç Dağıtım, Filtre Tasarımı, Görüntü Sıkıştırma, Görüntü Tanıma, Reaktif Güç Optimizasyonu, kontrol sistemlerinde parametrelerin ayarlanması [79] gibi mühendisliğin hemen her alanında kullanılmaktadır. Son olarak İnsanın zeka ve öğrenme yeteneği de modellenerek optimizasyon problemlerinin çözümüne uygulanmıştır [80,81].

### III. OPTİMİZASYON ALGORİTMALARININ OPTİMİZASYONU

Yukarıda ismi geçen tüm optimizasyon algoritmaları kendine ait parametrelere sahiptir. Bir optimizasyon algoritmasının başarısı da bu parametrelerin optimum değerlerinin kullanımına bağlıdır. Her algoritma için optimum değerler bu algoritma üzerinde yapılan çalışmalar arttıkça daha da netleşmektedir. Her geliştirilen algoritma ise kendi algoritmasının diğer birkaç algoritmadan daha iyi



olduğunu öne sürmektedir. Çalışmalardaki tablolardan da bu sonuçlar incelenebilir. Ancak yapılan incelemelerde bu karşılaştırmaların bazılarında algoritmaların optimum olmayan parametrelerle çalıştırılarak sonuçlar alındığı görülmüştür. Tablo 5.'te kısıtsız optimizasyon test problemlerinden olan F1 fonksiyonunun yukarıda sadece iki algoritma için bulunan sonuçları verilmiştir. [48]'de test problemlerinden F1(Sphere) fonksiyonu için en iyi sonuç GSA ile bulunmuş iken [49]'da GBMO çalışmasında en iyi sonuç GBMO ile bulunmuştur. Ve daha ilginç olan bu çalışmada PSO GBMO'dan sonra en iyi sonucu veren ikinci algoritma iken [48]'de verilen Tablo 5'de GSA'nın gerisindedir. Bu sebeple bundan sonraki çalışmalarda algoritmaların başarımı test edilirken daha önceki çalışmalardaki optimum parametrelerle de karşılaştırma yapılması önerilmektedir.

Yine mevcut durumda algoritmaların başarımı ortaya konulurken bilinen test problemlerinin optimum değerine yakınlığı ve çözüm süresine göre bir karşılaştırma yapılmaktadır. Algoritmalar çalıştırılırken benzer parametreler eşitlenmektedir. Bu parametrelerden biri popülasyon büyüklüğü diye adlandırılan bir algoritmanın bir iterasyonda kaç çözüm ile çalışacağıdır. Bir başka parametre ise sonlandırma kriteri olarak kullanılan maksimum iterasyon sayısıdır. Oysa iki algoritma aynı popülasyon büyüklüğü veya iterasyon sayısı ile basit bir kısıtsız optimizasyon test problemi için karşılaştırıldığında operatör sayısı daha az olan daha kısa sürede çözüm bulacaktır. Bu sebeple bu karşılaştırmalarda çözüm süresi sabit tutularak algoritmaların başarımının karşılaştırılması da önerilmektedir.

Bir başka öneri ise algoritma karmaşıklığında olduğu gibi en iyi en kötü durum analizlerinin yapılmasıdır. Her geliştirilen algoritma geliştirildiği yazarı tarafından önerilen standart bir parametre değeri ile çalıştırılarak diğer algoritmalar ile karşılaştırılırsa daha gerçekçi sonuçlar ortaya çıkacağı düşünülmektedir. Yine geliştirilen algoritmaların parametrelerinin problemin değişken sayısına ve kısıt sayısına bağlı olarak değişiminin verilmesi de önerilmektedir.

**Tablo 5.** Unimodal F1 fonksiyonunun n=30 için optimum değerlerinin karşılaştırması

Referans Makale	Karşılaştırma yapılan algoritmalar			
GSA[48] Tablo 4				
	PSO	RGA/GA	Referans	GSA
Parametreler	c1 = c2 = 2 w 0.9'dan 0.2.'ye doğrusal olarak azaltılıyor.	Aritmetik çaprazlama(0.3), gaus mutasyon(0.1) rulet tekerleği seçimi	G <sub>0</sub> =100 a=20, T=max iter	
Optimum değer	1.8X10 <sup>-3</sup>	23.13	7.3X10 <sup>-11</sup>	7.3X10 <sup>-11</sup>
GBMO[49] Tablo 11	c1 = 1.5, c2 = 1.5, S =-0.5 PS = 80, Maks iterasyon = 1000	PS= 80, Maks iterasyon = 1000 Çaprazlama(0.4), Mutasyon(0.01) rulet tekerleği seçimi	Molekül boyutu= 80, sıcaklık = 1000, a = 0.5, b = 0.2	
Optimum değer	9.0371	18.83	1.2X10 <sup>-9</sup>	0.2X10 <sup>-8</sup>

## IV. SONUÇ

Bu çalışmada özellikle son yıllarda geliştirilen doğadan esinlenen optimizasyon algoritmaları ele alınarak bir sınıflandırma yapılmıştır. Bu sınıflandırmada özellikle 2015 yılında yayınlanan çok sayıda çalışmanın olduğu görülmüştür. Bu çalışmalarda her geliştirilen yeni sezgisel optimizasyon algoritmaları kendi çözümlerinin diğer bazı algoritmalarından daha iyi olduklarını öne sürüldüğü görülmüştür. Ancak her bir sezgisel algoritma kendine ait parametrelere sahiptir ve bu parametrelerinde optimize edilmeye ihtiyacı vardır. Bu sebeple bundan sonra yapılan çalışmalarda algoritmalar karşılaştırılırken, problemi çözme süreleri veya fonksiyon için buldukları optimum değerlerin karşılaştırılmasına ek olarak, algoritmaların verilen bir standart çözüm süresinde hangi optimum değeri bulduğu veya istenen bir optimum değeri ne kadar sürede bulduklarının da karşılaştırılması gerektiği önerilmektedir. Yine bundan sonra yapılan çalışmalarda bu parametrelerin optimum değerlerinin problem değişken sayısı ve kısıtlara bağlı olarak verilmesi gerektiği önerilmektedir. Sonuç olarak doğadaki her bir canlı varlık, süreç veya kanun en kusursuz şekilde yer almaktadır ve bir optimizasyon algoritması olmaya adaydır. Ancak bu canlı, süreç veya kanunların optimizasyon problemlerini başarılı bir şekilde çözebilmeleri için parametrelerin seçimi de bir optimizasyon işlemidir ve gelecekteki çalışmalarda üzerinde durulmalıdır.

## V. KAYNAKLAR

- [1] J.H Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Michigan, (1975).
- [2] D.E Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA, (1989).
- [3] Anonim, [http://apps.webofknowledge.com/UA\\_GeneralSearch\\_input.do?product=UA&SID=N2j2zzcxZelCvVdCRdO&search\\_mode=GeneralSearch](http://apps.webofknowledge.com/UA_GeneralSearch_input.do?product=UA&SID=N2j2zzcxZelCvVdCRdO&search_mode=GeneralSearch) (Erişim tarihi: 3<sup>th</sup> of January, 2016).
- [4] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi *Science* **220(4598)** (1983) 671-680.
- [5] C. Blum *Physics of Life Reviews* **2(4)** (2005) 353-373.
- [6] M. Dorigo, *Optimization learning and natural algorithms*, Doktora Tezi, Politecnico Di Milano, Italy, (1992).
- [7] M. Dorigo, C. Blum *Theoretical Computer Science* **344(2-3)** (2005) 243-278.
- [8] K. Socha, M. Dorigo *European Journal of Operational Research* **185(3)** (2008) 1155-1173.
- [9] J. Kennedy, R. Eberhart, *Particle swarm optimization*, **IEEE Conference: 1995 IEEE International Conference on Neural Networks(ICNN95), Proceedings**, (1-6) (1995) 1942-1948.
- [10] D. Karaboga, B. Basturk *Applied Soft Computing* **8(1)** (2008) 687-697.
- [11] C. Öztürk, E. Hançer, D. Karaboğa (2014) DOI: 10.17341/gummfd.00459.
- [12] Anonim, [http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar\\_files/TestGO\\_files/Page422.htm](http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO_files/Page422.htm). (Erişim tarihi: 4<sup>th</sup> of January, 2016).
- [13] A. Yurtkuran, E. Emel (2015) DOI: 10.1016/j.amc.2015.09.064.
- [14] Y. Wencho, Z. Yinzhi, G. Liang, L. Xinyu, M. Jianhui (2016) DOI: 10.1016/j.eswa.2015.09.031.
- [15] A.N.K. Nasir, M.O. Tokhi, N.M.A. Ghani (2015) DOI: 10.1016/j.eswa.2014.09.010.
- [16] M. Aziz, M. Tayarani (2014) DOI: 10.1016/j.engappai.2014.07.021.

- [17] L. Wang, H. Ni, R. Yang, P. M. Pardalos, X. Du, M. Fei (2015) DOI: **10.1016/j.ins.2015.05.022.**
- [18] X. Geng, Z. Chen, W. Yang, D. Shi, K. Zhao (2011) DOI: **10.1016/j.asoc.2011.01.039.**
- [19] A.H. Karami, M. Hasanzadeh (2015) DOI: **10.1016/j.compeleceng.2014.12.014.**
- [20] M.K. Naik, R. P.(2016) DOI: **10.1016/j.asoc.2015.10.039.**
- [21] T. Zhang, T. Hu, X. Guo, Z. Chen, Y. Zheng (2013) DOI: **10.1016/j.knosys.2013.07.015.**
- [22] S. Sarafrazi, H. Nezamabadi-pour, S. Saryazdi (2011) DOI: **10.1016/j.scient.2011.04.003.**
- [23] Z. Li, W. Wang, Y. Yan, Z. Li (2015) DOI: **10.1016/j.eswa.2015.07.043.**
- [24] H. Garg (2016) DOI: **10.1016/j.amc.2015.11.001.**
- [25] M. Toksari (2016) DOI: **10.1016/j.ijepes.2015.12.032.**
- [26] S. Sarafrazi, H. Nezamabadi-pour, S.R. Seydnejad (2015) DOI: **10.1016/j.jksuci.2014.10.003.**
- [27] M. Metlicka, D. Davendra (2015) DOI: **10.1016/j.swevo.2015.03.002.**
- [28] M. Ghasemi, S. Ghavidel, E. Akbari, A.A. Vahed (2014) DOI: **10.1016/j.energy.2014.06.026.**
- [29] X. Yuan, J. Zhao, Y. Yang, Y. Wang (2014) DOI: **10.1016/j.asoc.2013.12.016.**
- [30] M. Pluhacek, R. Senkerik, D. Davendra (2015) DOI: **10.1016/j.swevo.2015.10.008.**
- [31] M. Mitić, N. Vuković, M. Petrović, Z. Miljković (2015) DOI: **10.1016/j.knosys.2015.08.010.**
- [32] A. Askarzadeh (2013) DOI: **10.1016/j.solener.2013.08.014.**
- [33] H. Aguiar, O. Junior, M.A.S. Machado (2015) DOI: **10.1016/j.procs.2015.07.002.**
- [34] E. Amiri, S. Mahmoudi (2015) DOI: **10.1016/j.asoc.2015.12.008.**
- [35] F. Gaxiola, P. Melin, F. Valdez, J.R. Castro, O. Castillo (2016) DOI: **10.1016/j.asoc.2015.10.027.**
- [36] S. Shadmand, B. Mashoufi (2016) DOI: **10.1016/j.bspc.2015.10.008.**
- [37] R.J. Kuo, W.L. Tseng, F.C. Tien, T.W. Liao (2012) DOI: **10.1016/j.cie.2012.06.006.**
- [38] Z. Izakian, M.S. Mesgari, A. Abraham (2016) DOI: **10.1016/j.compenvurbsys.2015.10.009.**
- [39] H. Li, H. He, Y. Wen (2015) DOI: **10.1016/j.ijleo.2015.09.127.**
- [40] C. Ozturk, E. Hancer, D. Karaboga (2015) DOI: **10.1016/j.asoc.2014.11.040.**
- [41] B. Jiang, F. Qiu, L. Wang, Z. Zhang (2013) DOI: **10.1016/j.ipm.2015.11.003.**
- [42] J. Tvrdík, I. Křivý (2015) DOI: **10.1016/j.asoc.2015.06.032.**
- [43] Y. Ding, X. Fu (2015) DOI: **10.1016/j.neucom.2015.01.106.**
- [44] S. Alam, G. Dobbie, S.U. Rehman (2015) DOI: **10.1016/j.swevo.2015.10.003.**
- [45] T.O. Ting, J. Ma, K.S. Kim, K. Huang (2016) DOI: **10.1016/j.asoc.2015.10.054.**
- [46] C. Wang, D. Mu, F. Zhao, J.W. Sutherland (2015) DOI: **10.1016/j.cie.2015.02.005.**
- [47] P. Cai, Y. Cai, I. Chandrasekaran, J. Zheng (2016) DOI: **10.1016/j.autcon.2015.09.007.**
- [48] E. Rashedi, H. Nezamabadi-pour, S. Saryazdi (2009) DOI: **10.1016/j.ins.2009.03.004.**
- [49] M. Abdechiri, M.R. Meybodi, H. Bahrami (2013) DOI: **10.1016/j.asoc.2012.03.068.**
- [50] V.K. Patel, V.J. Savsani (2015) DOI: **10.1016/j.ins.2015.06.044.**
- [51] H. Abedinpourshotorban, S.M. Shamsuddin, Z. Beheshti, D.N.A. Jawawi (2015) DOI: **10.1016/j.swevo.2015.07.002.**
- [52] A.H. Kashan (2015) DOI: **10.1016/j.cor.2014.10.011.**
- [53] A. Baykasoğlu, Ş. Akpınar (2015) DOI: **10.1016/j.asoc.2015.10.036.**
- [54] A. Baykasoğlu, Ş. Akpınar (2015) DOI: **10.1016/j.asoc.2015.08.052.**
- [55] M. Ghaemi, M.R. Feizi-Derakhshi (2014) DOI: **10.1016/j.eswa.2014.05.009.**
- [56] I. Rbough, A. Ameer El Imrani (2014) DOI: **10.1016/j.aasri.2014.05.005.**
- [57] H. Shah-Hosseini (2012) DOI: **10.1016/j.sbspro.2012.01.033.**
- [58] A. Hatamlou (2013) DOI: **10.1016/j.ins.2012.08.023.**

- [59] H. Eskandar, A. Sadollah, A. Bahreininejad, M. Hamdi (2012) DOI: **10.1016/j.compstruc.2012.07.010**.
- [60] H. Shareef, A. Asrullbrahim, A. HusseinMutlag (2015) DOI: **10.1016/j.asoc.2015.07.028**.
- [61] M. Dorigo, V. Maniezzo, A. Colorni (1996) DOI: **10.1109/3477.484436**.
- [62] T. Stutzle, H.H. Hoos *Future Generation Computer Systems* **16(8)** (2000) 889–914.
- [63] M. Clerc, M. Kennedy, J. (2002) DOI: **10.1109/4235.985692**.
- [64] D. Karaboga, B. Basturk *Journal Of Global Optimazition* **39(3)** (2007).
- [65] W. Pan (2012) DOI: **10.1016/j.knosys.2011.07.001**.
- [66] X.S. Yang, S. Deb (2009) DOI: **10.1109/NABIC.2009.5393690**.
- [67] A.H. Gandomi, A.H. Alavi (2012) DOI: **10.1016/j.cnsns.2012.05.010**.
- [68] K.M. Passino (2002) DOI: **10.1109/MCS.2002.1004010**.
- [69] X.S. Yang, *A New Metaheuristic Bat-Inspired Algorithm*, **International Workshop on Nature Inspired Cooperative Strategies for Optimization (NICSO 2008)**, Tenerife-Spain, (2008).
- [70] X.S. Yang *Lecture Notes in Computer Science* **5792** (2009) 169-178.
- [71] B.R. Rajakumar *Procedia Technology* **6** (2012) 126-135.
- [72] S. Mirjalili, S.M. Mirjalili, A. Lewis (2014) DOI: **10.1016/j.advengsoft.2013.12.007**.
- [73] A. Kaveh, N. Farhoudi (2013) DOI: **10.1016/j.advengsoft.2013.03.004**.
- [74] A.R. Mehrabian, C. Lucas (2006) DOI: **10.1016/j.ecoinf.2006.07.003**.
- [75] S.A. Uymaz, G. Tezel, E. Yel (2015) DOI: **10.1016/j.asoc.2015.03.003**.
- [76] M.D. Li, H. Zhao, X.W. Weng, T. Han (2016) DOI: **10.1016/j.advengsoft.2015.11.004**.
- [77] O. Abedinia, N. Amjady, A. Ghasemi (2014) DOI: **10.1002/cplx.21634**.
- [78] J.J.Q. Yu, V.O.K. Li (2015) DOI: **10.1016/j.asoc.2015.02.014**.
- [79] M.A. Cavuslu, C. Karakuzu, F. Karakaya *Applied Soft Computing* **12** (2012) 2707–2718.
- [80] C. Karakuzu *International Journal of Innovative Computing, Information and Control* **6(10)** (2010) 4755-4770.
- [81] L. Wang, H. Ni, R. Yang, P.M. Pardalos, X. Du, M. Fei (2015) DOI: **10.1016/j.ins.2015.05.022**.