



*T.C*

**SAKARYA ÜNİVERSİTESİ**  
**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

# **BSM 310 – YAPAY ZEKA**

**Grup üyeleri:**  
**G171210073 Elifnur ARAT**  
**G171210091 Hilal YILDIZ**  
**G171210019 Şeyma TÜFEKÇİ**

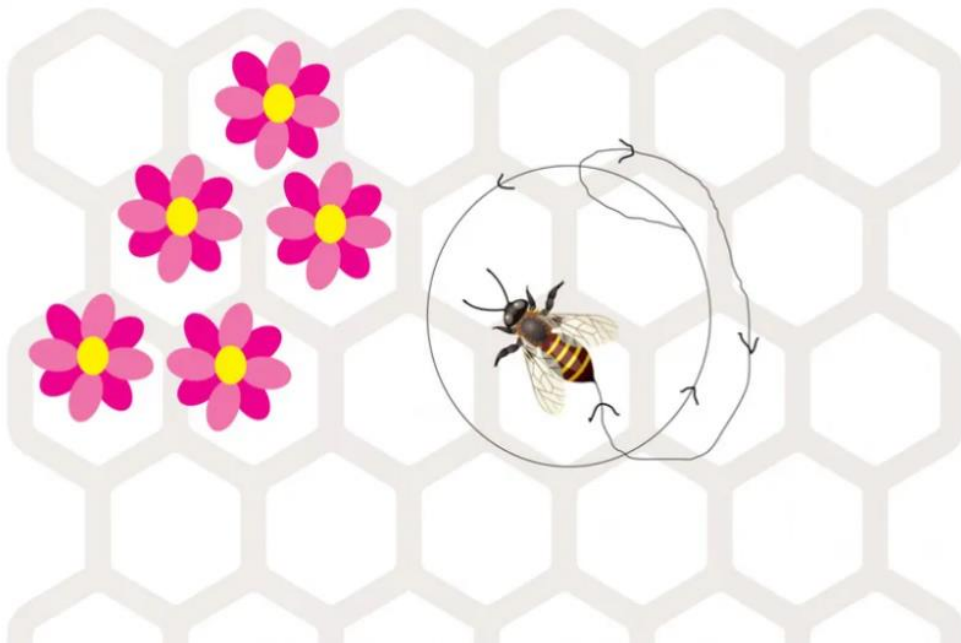
**Sakarya**

# Yapay Arı Kolonisi Algoritması (Artificial Bee Colony Algorithm)

Yapay Arı Kolonisi Algoritması(The Artificial Bee Colony Algorithm-ABC), sayısal problemleri optimize etmek için 2005 yılında Erciyes Üniversitesi Bilgisayar Mühendisliği Bölümü öğretim üyesi Prof. Dr. Derviş Karaboğa tarafından geliştirilen sürü tabanlı bir arama algoritmasıdır. Yani arıların topluluk olarak yiyecek arama davranışları baz alınarak geliştirilmiştir ve optimizasyon problemlerini çözmek için kullanılmaktadır. Karaboğa yaptığı bu çalışmayla, akademik çevrelerce “Bilim Oscarları” olarak adlandırılan Türkiye’den 10 akademisyenin girdiği ‘Dünyanın En Etkili Bilim İnsanları 2017’ listesinde yer almıştır. Ayrıca birçok makaleye konu olduğu gibi Scholarpedia’da bu çalışmaya yer verilmiş ve en çok okunanlar arasında yerini almıştır.

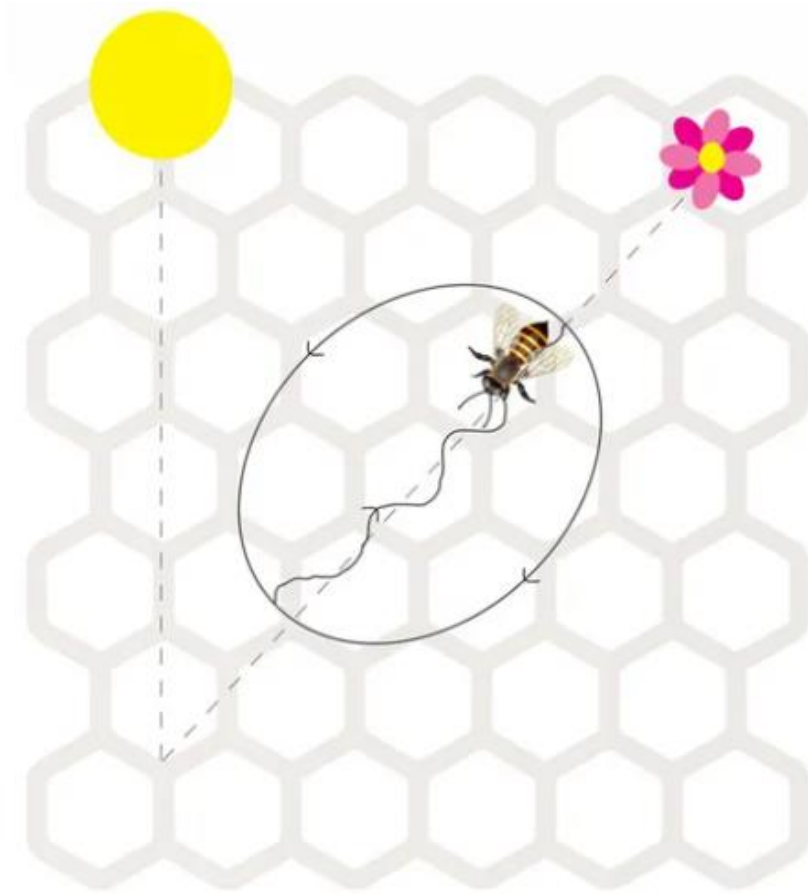
Sosyal bir düzen içinde yaşayan bal arıları, içgüdüsel olarak bu düzeni bilir ve hayatını bu kurallara uygun devam ettirir. Kovana ait olan her arının görevi bellidir. Arı kolonisinin yaşamının devamının sağlanması için en önemli işlerden birisi besin aramadır. Arının kovandan ayrılmasıyla başlayan arama süreci, başlangıçta rastgele yapılan yiyecek aramaları ile devam eder. Arılar arasındaki bilginin iletilmesi, kolektif yapının ve ortak bilgilerin oluşmasındaki en önemli konudur. Bu besin arama işlemi sırasında da yine her arının kendine ait görevleri vardır ve bu görevlere göre isimlendirilmişlerdir. (örn: kâşif arılar, işçi arılar) Arı kolonisinin yaşamını sürdürebilmesi için besin kaynağına ihtiyacı vardır. Kolonideki amaç ise en kısa sürede, en az enerjiyi harcayarak en kaliteli balı kovana getirmektir. Böylece zaman ve enerji minimuma, balın kalitesi ise maximuma yaklaşacaktır. Besin kaynağı arayışı genellikle kovanın 5-10 km çapında bir alanda gerçekleşir. Kâşif arılar öncelikle rastgele kaynak arayışı yaparlar. Buldukları kaynağı, kovana dönerek diğer arılara kendilerine has bir yöntemle tarif ederler. Böylece arılar arasında ‘bilgi aktarımı’ gerçekleşir. Bu yöntem ise danstır. Kovanın içinde ‘Dans Alanı’nda yapılan dans ile bulunan kaynağın yönü, uzaklığı, besinin kalitesi vb. hakkında diğer arılar bilgilendirilir. Besin kaynağının durumuna göre birkaç çeşit dans şekli geliştirilmiştir. Arıların 2 çeşit dansı bulunur.

## 1) Çember Dansı (Round Dance):



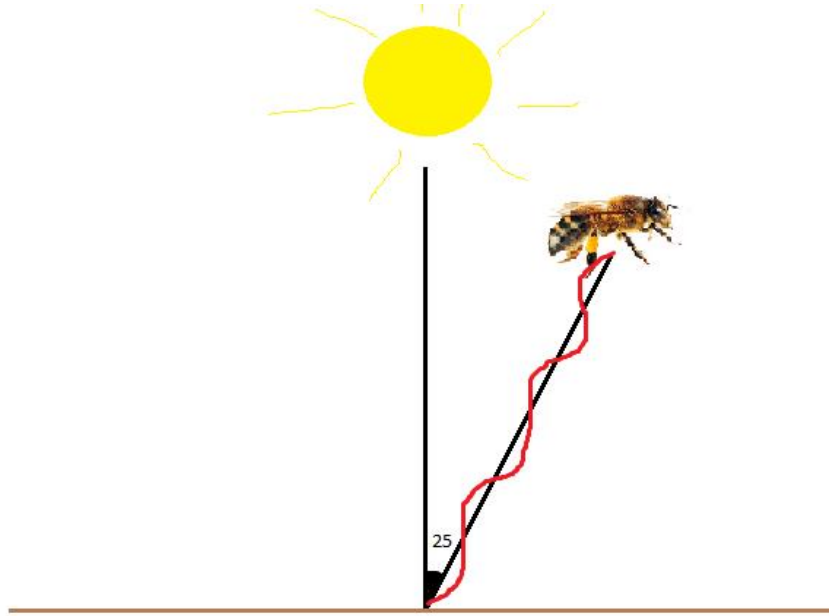
Birincisi, yukarıdaki fotoğrafta da görebileceğiniz, çok yakında olan ve pek de tarif gerektirmeyen kaynakları anlatmak için kullanılan çember dansıdır (round dance). Bu yakın mesafe 50-100 metre uzaklıklarıdır. Bu yakınlıktaki bir kaynağı keşfeden kâşif arı kovana dönerek dans alanında dansını sergilemeye başlar. Kâşif arı, saat yönünde ve aksi yönde hızlı hareketlerle çemberler çizerek diğer arılara çok yakında bir kaynak olduğunu işaret eder. Bu iletişim biçiminde uzaklık hakkında bilgi verilmesine karşın yön ve konum net olarak belli değildir. Ancak arılar gelişmiş koku yeteneklerini kullanarak zorlanmadan yakındaki bu kaynakları kolayca bulabilirler.

## 2) Sallantı Dansı (Waggle Dance):



Arı danslarının ikinci çeşidi ise çok daha karmaşıktır. Daha uzaktaki kaynaklara dair bilgi vermeyi amaçlayan bu dansa sallantı dansı (waggle dance) denir. Kovan içinde düşey olarak asılı petekler üzerinde yerini alan kâşif arı, önce vücudunun arka kısmını sağa sola sallayarak düz bir çizgi üzerinde hızla hareket eder. Ardından sağa ya da sola keskin bir dönüş yapıp, vücudunu sallamayı bırakarak yarım çember şeklinde bir eğriyi takip ederek dansına başladığı noktaya döner. Başlangıç noktasına ulaştığında aynı düz çizgiyi takip ederek sallantı dansını tekrarlar ve bu defa ilk seferkinin aksi yöne dönerek diğer yarım çemberi de çizer ve başlangıç noktasına döner. Kâşif arı bu dansı defalarca tekrarlayarak etrafındaki diğer arıların da onun bilgisini edinmesini sağlar. Dansın bilgi taşıyan kısmı ise sallantılı geçen düz çizgidir. Burada iki çeşit bilgi aktarılır: yön ve uzaklık. Başta dediğimiz üzere, kovandaki petekler yerçekimi esas

alınarak düşey doğrultuda konumlandırılmıştır. Arılar, bu yerçekimsel doğrultuyu güneşi temsil etmek için kullanırlar. Eğer bir kâşif arı, yerçekimine tam ters yönde (yani yukarı doğru) dans ederse; bunun anlamı kaynağa gitmek için güneşe doğru uçmak gerektiğidir. Dolayısıyla dansın yukarı yönden ne yönde ne kadar saptığına göre uçulacak doğrultu da belirlenmiş olur. Örneğin, sallantı dansının yapıldığı çizgi sağa doğru 20 derece eğikse, kovandan çıkılıp güneş karşıya alındıktan sonra, saat yönünde 20 derece dönüp dosdoğru uçmak gerekir. Uzaklık bilgisi ise dansın doğrusal kısmının (sallantının gerçekleştiği kısmın) ne kadar sürdüğü ile ilişkilidir. Kabaca 1 saniyelik sallantı, 1000 metre uzaklıktaki bir kaynağa işaret eder.



- Danslar diğer arılarında aynı titreşimleri yapmasıyla son bulur ve arılar kaynağa yönelir.

### Görevlerine Göre Arılar:

- Danslardan edinilen bilgiye göre arılardan bazıları bulunan besin kaynağına yönelir ve tarif edilen kaynağa gidip işlemeye başlar. Bu arılar **gözcü arılardır**. (Onlookers) Bu arılar genellikle sahip oldukları kaynaklardan memnun olmayan ve yeni kaynağın besin kalitesine göre kaynak değiştiren arılardır. Belli bir görevleri olmadığı ve bulunan yeni kaynaklara göre hareket ettikleri için 'unemployed' arılar olarak da bilinirler.

- Bazı arılar ise tarif edilen kaynağa gitmek yerine bambaşka bir kaynak aramak için kovandan ayrılırlar. Daha önceden keşfedilmemiş ve bilinmeyen kaynakları bulmaya çalışırlar. Bu arılar da **kâşif arılardır**. (Scouts) Bu arılar da genellikle sahip oldukları kaynaktan ve onun besin kalitesinden memnun olmayan arılardır. Bu sebeple el değmemiş kaynaklar arayarak günlerini geçirirler.

- **İşçi arılar**sa bulunan kaynakları işleyerek kovana getirmekle sorumludurlar. (Employee)  
İşlenen kaynak bittiğinde yeni kaynak aramak üzere eski kaynağın çevresinde aranarak kâşif arıya dönüşebilirler.



Besin kaynağının kalitesi kötüyse veya kaynak tükenmişse de yeni besin kaynağı aranmaya başlar. Besin kaynağı aramalarında en önemli etkenlerden biri de besinin kalitesidir. Kaliteli besin her zaman önce tercih edilir. Örneğin, yakın ama kalitesiz bir besindense uzaktaki kaliteli besin işlenir. Çünkü çiçeklerin kalitesi kârlılık oranını arttıran etkenlerdendir.

Tüm bu besin arama işlemleri sonuçta bir 'Repeat-Until' döngüsünden ibarettir. Öyle ki;

Scout Bees Phase (Initial Food Sources) // Kaşif Arılar

**REPEAT**

Employed Bees Phase // İşçi Arılar

Onlooker Bees Phase //Gözcü Arılar

Scout Bees Phase

**UNTIL** (requirements are met) // Güneş batana kadar

İlk başta kâşif arıların kovandan tamamen rastgele bir biçimde kaynak arayışına çıkmaya başlamasıyla olaylar başlar. Daha sonra döngü ise işçi ve gözcü arıların bulunan kaynağı işleyerek kovana getirmesiyle başlar. Bu sırada kâşif arılar da daha iyi ve başka besin kaynakları aramaya devam ederler. Böylece döngü güneş batana kadar tekrarlanır.

## Arıların Davranışları ve Algoritmadaki Karşılıkları

1. Kâşif arılar rastgele besin kaynakları bulur.	1. Rastgele çözüm kümeleri oluşturulur. Her çözüm kümesi, parametre sayısı kadar eleman içerir. Oluşturulan çözüm kümelerinin uygunluk değerleri bulunur.
2. İşçi arılar besin kaynaklarına gider ve besin kaynağını işleyerek kovana dönerler.	2. Tüm çözüm kümelerine sırayla iyileştirilme formülü uygulanır. Eğer çözüm kümesi iyileştirilmişse yeni çözüm kümesi mevcut ile değiştirilir.
3. Gözcü arılar işçi arıların danslarından işçi arıların gelmiş olduğu besin kaynakları hakkında bilgi edinirler ve işlemek üzere besin kaynaklarına giderler.	3. Çözüm kümelerinin uygunluk değeri yüksek olanların olasılığı yüksek olacak şekilde ayarlanır. Bu olasılığa bağlı rastgele seçilen çözüm kümesine bir önceki aşamadaki gibi iyileştirilme formülü uygulanır.
4. Eğer nektar kaynağı daha fazla besin üretemiyorsa işçi arı kâşif arıya dönüşür ve yeni bir nektar kaynağına yönelir.	4. Eğer belirlenen limit kadar iyileştirilmeyen çözüm kümesi varsa bu hafızadan silinir ve yerine rastgele bir çözüm kümesi oluşturulur.
5. Belirlenen tekrar sayısına ulaşıncaya kadar 2. Aşamaya geri dönlür.	5. Maksimum iterasyona ulaşıncaya kadar 2. Aşamaya geri dönlür.

## Algoritmadaki Başlıca Tanımlar

**Problem:** Algoritmada çözüm aradığımız problemdir ve genellikle bir matematiksel formülle belirtilir.

**Uygunluk fonksiyon:** Algoritmada çözmek istediğimiz problem için bulduğumuz çözüm kümelerinin iyilik derecesini bulmamıza yarayan formül. Probleme göre formül çözen tarafından belirlenir.

**Popülasyon Boyutu:** Popülasyondaki işçi ve gözcü arı sayısının toplamıdır.

**Parametre Boyutu:** Uygunluk fonksiyonumuzda bilinmeyen değişkenlerin sayısını ifade eder.

**Parametre Aralığı:** Parametrelerin alabileceği maksimum ve minimum değerlerdir. Yine probleme göre çözen tarafından belirlenir. Değer aralıkları ne kadar doğru belirlenirse algoritma o kadar iyi çalışır.

**Maksimum İterasyon:** Algoritmanın çalışacağı maksimum döngü sayısını belirtir.

**Besin Kaynağı Sayısı:** Popülasyon boyutunun yarısı kadardır. Algoritmadaki hafızada tutulan toplam çözüm kümesini ifade eder.

## Algoritma Adımları

**1-) Kâşif arıların rastgele besin kaynaklarına gitmesi:** İlk aşama olarak her çözüm kümesi için parametre boyutu kadar minimum ve maksimum parametre değerlerine göre rastgele çözüm kümesi oluşturulur. Uygunluk değerleri uygunluk fonksiyonu ile hesaplanır.

**Parametre Değeri Atama Formülü(f1):**  $X_{ik} = l_{bj} + rand(0,1) \times (u_{bj} - l_{bj})$

**i** = Seçilen çözüm kümesinin indeks numarası

**k** = i seçilen çözüm kümesinde seçilen parametrenin indeks numarası

**X<sub>ik</sub>** = Seçilen çözüm kümesinin geçerli parametre numarası

**rand0,1** = 0 ile 1 arasında rastgele sayı

**l<sub>bj</sub>** = Parametrenin alabileceği minimum sayı

**u<sub>bj</sub>** = Parametrenin alabileceği maksimum sayı

Çözüm kümeleri oluşturulduktan sonra her biri uygunluk formülüne koyulur bulunan değer Yapay Arı Kolonisi için ek uygunluk fonksiyonuna verilir ve dönen değer Yapay Arı Kolonisi için uygunluk değerini verir. Yapay arı algoritması kendi kullandığı uygunluk formülü aşağıda verilmiştir. Ancak probleme göre de uygunluk formülü oluşturulabilir.

**Yapay Arı Kolonisi Ek Uygunluk Formülü (f2):** Eğer  $D_i \geq 0 \rightarrow 11+D_i$   
Eğer  $D_i < 0 \rightarrow 1 + |D_i|$

## REPEAT

**2-) İşçi arıların besin kaynaklarına gitmesi:** Oluşturulan çözüm kümelerinin her biri sırayla geliştirilme formülü uygulanır. Eğer çözümün uygunluk değeri artmışsa eski çözüm kümesiyle yeni çözüm kümesi değiştirilir. Eğer uygunluk değerinde artış yoksa o indeks değerine ait çözüm kümesinin deneme değeri bir attırılır.

**Geliştirme Formülü (f3):**  $V_{ik} = X_{ik} + \phi_{ik} \times X_{ik} - X_{jk}$

i = Seçilen çözüm kümesinin indeks numarası

k = Rastgele seçilen parametre indeks numarası

j = Rastgele seçilen komşu çözüm kümesinin indeks numarası

$V_{ik}$  = Formül sonu oluşan yeni çözüm kümesinin parametre değeri

$X_{ik}$  = Seçilen çözüm kümesinin k. parametre değeri

$X_{jk}$  = Rastgele seçilen komşu çözüm kümesinin k. parametre değeri

$\phi_{ik}$  = Probleminize göre optimize edilir. Bir nevi seçilen iki çözüm kümesinin parametreleri arasındaki değişim mesafesini temsil eder. Orijinal Yapay Arı Algoritmasında (random(0,1)-0.5) \* 2 olarak alınmıştır.

Eğer çözüm kümesinde yeni bulunan parametre değerlerinin belirlediğiniz minimum ve maksimum değerleri arasında çıkmasını istiyorsanız aşağıdaki formülü uygulamalısınız.

**Parametre değeri limit kontrolü (f4):** Eğer  $X_{it} > u_{bi} \rightarrow X_{it} = u_{bi}$   
Eğer  $X_{it} < l_{bi} \rightarrow X_{it} = l_{bi}$

i = Seçilen çözüm kümesinin indeks numarası

k = i seçilen çözüm kümesinde seçilen parametrenin indeks numarası

$X_{ik}$  = Seçilen çözüm kümesinin geçerli parametre numarası

**l<sub>bj</sub>** = Parametrenin alabileceği minimum sayı

**u<sub>bj</sub>** = Parametrenin alabileceği maksimum sayı

Yukarıdaki işlemler tüm çözüm kümesi için uygulanır.

**3-) Gözcü arıların besin kaynaklarına gitmesi:** Uygunluk değerine göre çözüm kümelerinin seçim şansı oluşturulur. Örnek olarak Rulet Seçim Yöntemi, Turnuva Seçim Yöntemi gibi seçim yöntemleri kullanılabilir. Rulet seçim yöntemini inceleyelim. Rastgele gözcü arı sayısı kadar çözüm kümesi uygunluk değerlerinin olasılıklarına göre seçilir ve geliştirilme formülü (f3) işçi arı aşamasındaki gibi uygulanır. Tek farkı sırayla besin kaynakları değil olasılıklarına göre rastgele bir seçim yapılmasıdır. Yani aynı indeks değerine ait çözüm kümesi bu seçim yöntemiyle birden fazla kez seçilebilir. Eğer çözümün uygunluk değeri artmışsa eski çözüm kümesiyle yeni çözüm kümesi değiştirilir. Eğer uygunluk değerinde artış yoksa o indeks değerine ait çözüm kümesinin deneme değeri bir attırılır.

**Rulet seçim yöntemi formülü (f5):**  $P_i = U_i / U_j$

$P_i$  = Seçili çözüm kümesinin gözcü arı için seçilme olasılığı

$U_i$  = Seçili çözüm kümesinin uygunluk değeri

$\sum U_j$  = Tüm çözüm kümelerinin uygunluk değerlerinin toplamı

**4-) İşçi arıların kâşif arılara dönüşmesi:** Tüm çözüm kümeleri kontrol et.

**UNTIL** (Belirlenen tekrar sayısına ulaşana kadar)

## ABC'nin Temel Özellikleri

- Basit ve esnek bir algoritmadır.
- Arıların gerçek yiyecek arama davranışlarına çok yakın olarak tasarlanmıştır.
- Sürü zekasına dayalıdır.
- Başlangıçta nümerik problemler için tasarlanmış ancak ilerleyen süreçte gerçek tasarım problemleri için de kullanılmıştır.
- Kontrol parametresi azdır.

## ABC'nin Uygulama Alanları

- Sinir ağlarının ve derin ağların eğitiminde
- Elektrik ve endüstri mühendisliklerinde
- Makina ve inşaat mühendisliği alanlarında
- Veri madenciliği: kümeleme, öznitelik seçimi, sınıflandırma ve kural keşfinde
- Kablosuz sensor ağlarında yayılma, yerleştirme ve yönlendirmede
- Görüntü ve sinyal işlemede
- Protein yapısı optimizasyonunda, ilaç tasarımında
- Finansal tahminlerde
- Tıp alanında

Uygulama alanlarını daha iyi anlatabilmek için geliştirilen birkaç projeyi sunmakta fayda görüyoruz.



#### ABC Kullanılarak Geliştirilen Projelerden Bazıları:

- Yapay Arı Kolonisi Algoritması tabanlı kararlı güç sistemi dengeleyicisi tasarımı, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi
- İnsansız hava araçları için Yapay Arı Kolonisi Algoritması kullanarak rota planlama, Yalova Üniversitesi Bilgisayar Mühendisliği
- Modifiye ABC algoritması ile nümerik fonksiyon optimizasyonu, Erciyes Üniversitesi Bilgisayar Mühendisliği
- Yapay Arı Kolonisi Algoritması ile elektrik güç sistemi optimal yakıt maliyetinin belirlenmesi, Düzce Üniversitesi
- Uydu görüntülerinin Yapay Arı Kolonisi Algoritması ile bölütlenmesi, Erciyes Üniversitesi
- Yapay Arı Kolonisi Algoritması kullanarak otomatik ders çizelgeleme, Süleyman Demirel Üniversitesi
- Küresel en iyi Yapay Arı Kolonisi Algoritması ile otomatik kümeleme, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi

## KAYNAKÇA

- <http://mf.erciyes.edu.tr/abc/>
- <http://sujest.selcuk.edu.tr/sumbtd/article/view/417/538>
- [https://en.wikipedia.org/wiki/Artificial\\_bee\\_colony\\_algorithm](https://en.wikipedia.org/wiki/Artificial_bee_colony_algorithm)
- <https://abc.erciyes.edu.tr/projects.htm>
- [http://www.scholarpedia.org/article/Artificial\\_bee\\_colony\\_algorithm](http://www.scholarpedia.org/article/Artificial_bee_colony_algorithm)
- <https://www.bilalsaim.com/yapay-ari-kolonisi-algoritmasi-artificial-bee-colony-algorithm-h1630>
- <http://static.dergipark.org.tr:8080/article-download/797c/e1b9/1c95/5e1c59e1e02de.pdf?>