

**T.C.
KTO KARATAY ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

CE 454 INTRODUCTION TO EVOLUTIONARY COMPUTING

2019 GÜZ DÖNEMİ FINAL PROJESİ

**YAPAY ARI KOLONİSİ ALGORİTMASI İLE GEZGİN SATICI PROBLEMİ'NİN
KONYA İLÇE MERKEZLERİNE UYGULANMASI**

Öğrenci: 21501293, ERKAN ÇÖMEZ

**DERSİN ÖĞRETİM ELEMANI
Prof. Dr. NOVRUZ ALLAHVERDİ**

KONYA 2019
İÇİNDEKİLER

İÇİNDEKİLER.....	II
1. GİRİŞ	1
2. Yapay Arı Kolonisi Algoritması	2
2.1. YAKA'nın Adımları.....	3
2.2. Başlangıç Besin Kaynaklarının Belirlenmesi.....	4
2.3. Başlangıç Besin Kaynaklarının Belirlenmesi.....	4
2.4. Gözcü Arıların Besin Kaynaklarını Seçiminde Kullanacakları Olasılık Değerlerinin Hesaplanması.....	5
2.5. Gözcü Arıların Besin Kaynaklarını Seçmeleri.....	6
2.6. Besin Kaynağının Terk Edilmesi ve Kaşif Arı Üretimi.....	6
3. Yapay Arı Kolonisi Algoritması ile İlgili Yapılan Çalışmalar.....	7
3.1. Yapay Arı Kolonisi Algoritması ile İlgili Örnekler.....	7
3.1.1 Yapay Arı Kolonisi (YAK) Algoritması ile Rota Planlama.....	7
4. Sonuç.....	8
KAYNAKÇA.....	9

1. GİRİŞ

1.1. Yapılan Çalışmanın Amacı ve Önemi

Gezgin satıcı problem, herhangi bir listedeki bir şehirden başlayıp listede bulunan tüm şehirleri sadece bir kez ziyaret edip, tekrar başladığı şehre dönen bir satıcı için en kısa turun belirlenmesi problemdir. Problem ilk olarak 1800'lü yıllarda İrlandalı matematikçi Sir William Rowan Hamilton ve İngiliz matematikçi Thomas Penyngton Kirkman tarafından ele alınmış ve günümüze kadar üzerinde binlerce çalışma yapılmıştır. Dağıtım, rotalama, kuruluş yeri belirleme, planlama, lojistik gibi problemlerde geniş bir uygulama alanına sahip olan gezgin satıcı problemi, aynı zamanda optimizasyon alanında, araştırmacılar tarafından üzerinde uzun yıllardır çalışmalar yapılan NP-hard (çözümü zor) sınıfında yer alan bir problemdir.[1] Burada şehir sayısına n dersek, bir satıcının tüm şehirleri gezmesi için $n!$ tane farklı tur arasından seçim hakkı olacaktır. Eğer şehirler düğümlerle, yollar ise hatlar ile gösterilirse problem çizge üzerinde minimum maliyetli kapalı yolun bulunmasına karşılık gelmektedir [2]. Bugüne kadar en büyük seyyar satıcı problem 24.978 noktalıdır ve İsveç'te yerleşimi olan her nokta için çözülmüştür. Öte yandan bu problem, seyyar satıcılarından öte internet üzerinde paketlerin yönlendirilmesi gibi konuların çözümünde de büyük rol oynamaktadır. Problemde düğüm sayısı arttıkça problemin çözümünde harcanan zaman artmaktadır. Problemde başlangıç şehri verilmişse, mümkün olan Hamilton yolları sayısı geriye kalan $(n-1)$ adet şehrin yer değişimine, yani $(n-1)!$ 'e eşit olmaktadır.

Bu çalışmada yapay arı kolonisi kullanılarak gezgin satıcı probleminin Konya ilçe merkezlerine uygulanması gerçekleştirilmiştir.

Konya ilimizdeki 31 ilçe merkezinin mesafeleri alınmıştır. Gezgin satıcı problemine yapay arı kolonisi kullanılarak alınan sonuçların değerlendirilmesi de bu çalışmayı önemli kılmaktadır.

1.2. Organizasyon

Bu çalışma giriş, sezgisel algoritmalar, yapay arı kolonisi algoritması kullanarak gezgin satıcı probleminin Konya'daki ilçe merkezlerine uygulanması ve sonuçlar ve tartışma olmak üzere 4 bölümden oluşmaktadır.

Birinci bölüm, giriş bölümünü olmakla beraber gezgin satıcı problemi genel olarak açıklanmış ve yapılacak olan çalışma ve öneminden bahsedilmiştir.

İkinci bölümde sezgisel algoritmaların ve çeşitlerinden bahsedilmiştir.

Üçüncü bölümde yapay arı kolonisi algoritması kullanılarak gezgin satıcı probleminin Konya'daki ilçe merkezlerine uygulanması, işlem basamakları ve elde edilen sonuçlar hakkında bilgi verilmiştir.

Dördüncü bölümde Sonuçlar ve Tartışma kısmına yer verilmiştir.

Çalışmanın sonunda kaynak bilgisi verilmiştir.

2. SEZGİSEL ALGORİTMALAR

Bir problem içinden belirli koşullarda en iyi çözümü bulma işlemine optimizasyon denmektedir. Sezgisel algoritmalar, bilgisayar bilimlerinde sezgisel bir yaklaşımın problem çözümüne uygulandığı algoritmalarıdır.

Bir problemin çözümünde iki önemli unsur vardır;

- 1- Çözümün hızlı olması
- 2- Her zaman aynı sonucun elde edilmesi

Sezgisel algoritmalar günlük hayatımızda yaptığımız seçimler gibi klasik yöntemlerle yapılamayan seçim yapma yeteneği ile geliştirilmiştir. Seçim yapma işlemi sezgisel algoritmaya en iyi çözümü seçme veya rastgelelik olarak yansımıştır.

1991 yılı itibarıyle sezgisel algoritmaların kullanımında bir artış yaşanmıştır. Jones ve arkadaşlarına göre bu artışın birinci sebebi sezgisel algoritmaların hesaplama gücünün iyi olması, ikinci sebebi ise bu algoritmaların dönüştürülebilir olmasıdır[3].

Sezgisel algoritma, gelişime dayalı algoritma ve sürü zekâsı temelli algoritmalar içerir. Genetik algoritma, gelişime dayalı bir yapıya sahiptir ve sezgisel algoritmanın temelini oluşturmaktadır. Bununla birlikte bozkurt, kuş ve arı gibi sürü halinde yaşayan canlıların yaşantısından esinlenerek geliştirilen, sürü zekâsı temelli, parçacık sürü optimizasyonu, karınca kolonisi ve arı kolonisi algoritması sezgisel algoritmalar içerisinde büyük öneme sahiptir.

1- Gelişime Dayalı Optimizasyon Algoritmaları

- a. Tabu Arama Algoritması
- b. Genetik Algoritma

2- Sürü Zekası Temelli Optimizasyon Algoritmaları

- a. Karınca Kolonisi Algoritması
- b. Arı Kolonisi Algoritması

2.1. Gelişime Dayalı Optimizasyon Algoritmaları

2.1.1. Genetik Algoritma

Genetik algoritmalar klasik yöntemlerle çözümü zor olan problemlere çözüm bulmak amacıyla biyolojik evrim sürecinin bilgisayar ortamında kodlandığı ve bu evrim süreci

kullanılarak optimum ya da optimuma yakın sonuçların araştırıldığı bir arama yöntemi olarak tanımlanabilir. Bu yaklaşımın temelini Holland ortaya atmıştır[4].

Genetik algoritmaların ilk aşamasında incelenen problemin parametreleri uygun şekilde kodlanarak kromozomlar olusturulur. Daha sonra bu kromozomlardan oluşan bir baslangıç popülasyonu meydana getirilir. Popülasyonda yer alan her bir kromozom problemin çözümü için bir adaydır. Ardından popülasyondan bazı kromozomlar seçilir ve dogadaki genetik süreçlere dayanan “çaprazlama” ve “mutasyon” genetik operatörleri uygulanarak daha iyi kromozomlar elde edilmeye çalışılır. Ardı ardına gerçekleştirilen iterasyonlar ile yeni nesiller olusturulur ve bu yeni nesillerin verdiği uygunluk değerleri popülasyondakilerle karşılaştırılır. Yeni nesillerde yer alan kromozomlar içinde popülasyondakilere göre daha iyi uygunluk değerine sahip olan kromozomlar popülasyona dahil edilirler. Bu sayede popülasyonda yer alan kromozomlar bir evrim süreci içinde kendilerini daha iyi kromozomlarla yenilerler. Bu süreç yeni nesil sayısı belirli bir sayıya ulaşınca kadar veya popülasyondaki uygunluk değerlerinde belirli bir miktarda iyilesme olmayıncaya kadar devam eder.[5]

2.1.2 Tabu Arama Algoritması

Tabu Arama algoritması, optimizasyon problemlerinin çözümü için F. Glover tarafından geliştirilmiş iteratif bir araştırma algoritmasıdır[6]. Temel amacı, son çözüme götüren adımın dairesel hareketler yapmasını önlemek için bir sonraki döngüde tekrarın yasaklanması veya cezalandırılmasıdır. Böylece yeni çözümlerin incelenmesiyle Tabu Arama algoritması, bölgesel en iyi çözümün daha ilerisinde bulunan çözümlerin araştırılabilmesi için bölgesel-sezgisel araştırmaya yardım etmektedir. Tabu Arama algoritmasının bölgesel optimalliği aşmak amacıyla kullandığı temel prensip, değerlendirme fonksiyonu tarafından her iterasyon- da en yüksek değerlendirme değerine sahip hareketin bir sonraki çözümü oluşturmak amacıyla seçilmesine dayanmaktadır. Bunu sağlamak amacıyla bir tabu listesi oluşturulur, tabu listesinin orijinal amacı önceden yapılmış bir hareketin tekrarından çok tersine dönmesini önlemektir. Tabu listesi kronolojik bir yapıya sahiptir ve esnek bir hafıza yapısı kullanır. Tabu arama algoritması her ne kadar istenmeyen noktaların işaretlenmesi olarak açıklanmış olsa da daha cazip noktaların işaretlenmesi olarak da kullanılır.

2.2. Sürü Zekâsı Temelli Optimizasyon Algoritmaları

2.2.1 Yapay Arı Kolonisi Algoritması

Prof. Dr. Derviş KARABOĞA tarafından geliştirilen YAKA literature 2005 yılında girmiştir[7]. Arıların besin arama davranışlarının YAKA'nın geliştirilmesini sağlamıştır. YAKA'da görevli arı, kaşif arı ve gözcü arı olmak üzere üç tip arı bulunmaktadır. YAKA'da arıların tüm davranışlarından etkilenilmemiği gibi bazı varsayımlarda da bulunulmuştur. Bu varsayımlardan en göze çarpanı her bir besin kaynağının sadece bir görevli arı tarafından kullanılmasıdır. Besin kaynağı ve çalışan görevli arı sayısı birbirine eşittir. Bir diğer varsayımda ise gözcü arı ve çalışan arı sayısının birbirine eşit olmasıdır. Böyle bir varsayımda bulunulmasına rağmen aslında bir nektara gidip gelen arının görevli olduğu besin kaynağı tüketindiğinde bu arının kâşif arı olması da söz konusudur. Kaşif arılar görevli arıların işleri bitince ortaya çıkarlar. Yeni besin kaynaklarını ön bilgi olmadan rasgele olarak bulduğundan algoritmanın daha önceden keşfedemediği zengin besin kaynaklarını yani yüksek uygunluktaki olası çözümler kümesini bulabilir.

2.2.2 Karınca Kolonisi Algoritması

Karinca Kolonisi Optimizasyonu NP-hard optimizasyon problemlerine başarılı bir şekilde uygulanabilen popülasyon temelli bir yaklaşımdır[8]. İlk olarak Gezgin Satica Problemi üzerinde uygulanmıştır. Bir noktadan başlayarak ve tüm noktalara bir kez uğrayarak tekrar aynı noktaya en kısa yoldan dönmek diye tanımlanan Gezgin Satica Problemi'nin çözümünde Karınca Kolonisi Algoritmasının oldukça etkili olduğu görülmüştür[9]. Karınca Kolonisi Algoritması gerçek karınca kolonilerinin davranışlarından esinlenerek geliştirilmiştir. Tekniğin en temel unsurlarından biri haberleşme aracı olarak kullanılan ve problemlerde çözümün kalitesini gösteren feromon kimyasalıdır. Feromon, gerçek karıncaların da bir haberleşme ve yön bulma aracı olarak kullandıkları, vücutlarından salgıladıkları kimyasaldır. Feromon izleri, karıncalar tarafından güncellenmekte ve bir bilgiyi temsil etmektedir. Bir yolda feromon izinin yoğun olması, yolun kalitesini gösterir ve tercih olasılığını arttırır. Karınca Kolonisi Algoritması'nda, yapay karıncalar, gerçek mesafeler dikkate alınarak yapılmış olan model üzerinde en kısa yolu araştırmaktadırlar. Yollardaki feromon izleri yine yapay olarak, karıncaların geçiş sıklığıyla orantılı bir şekilde güncellenmektedir.

3. YAPAY ARI KOLONİSİ ALGORİTMASI KULLANILARAK GEZGIN SATICI PROBLEMİ'NİN KONYA'NIN İLÇE MERKEZLERİNE UYGULANMASI

Bu bölümde Konya'nın 31 ilçe merkezinin 2019 Kara Yolları Genel Müdürlüğü'nden alınmış olan ilçe mesafe cetveli kullanılarak Yapay Ari Kolonisi Algoritması uygulanmış ve elde edilen sonuçlar AKÇA M. R. (2011)'nın Hibrit Ari Kolonisi Algoritması ve ÖZSAĞLAM M. Y. (2009)'ın Genetik Algoritmalar ve Parçacık Sürü Optimizasyon sonuçları ile karşılaştırılmıştır. Kullanılan mesafe cetveli Ek-1 de sunulmuştur.

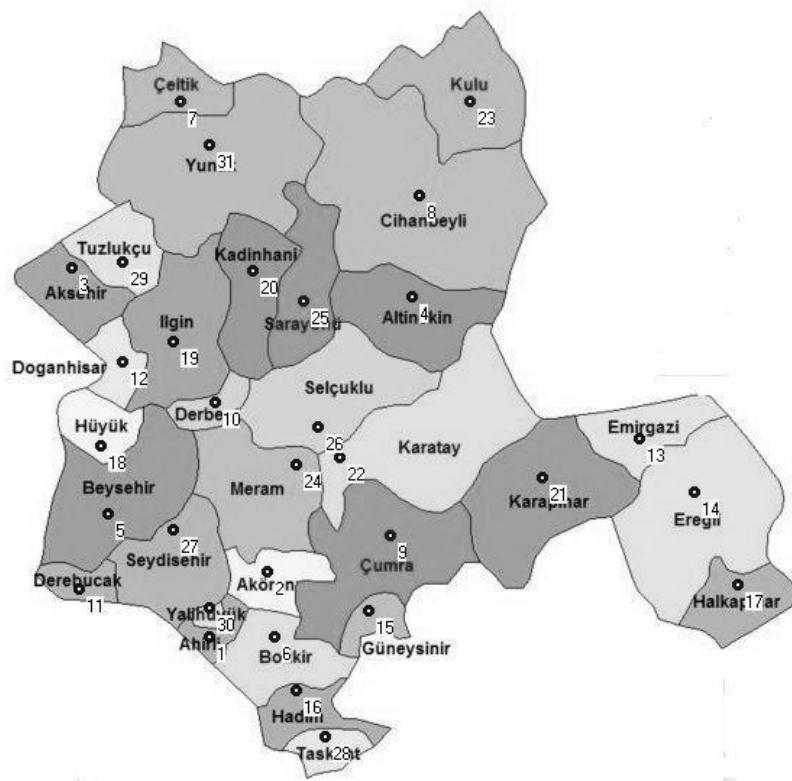
3.1. Konya İli İlçe Merkezleri İçin Gezgin Satıcı Problemi Çözümü

Konya ilinde yer alana 31 ilçe merkezi üzerinde Yapay Ari Kolonisi Algoritması, Gezgin Satıcı Problemi için uygulanmıştır. Algoritma 6 kez çalıştırılmış ve iterasyon 5000 seçilerek en iyi yol haritası uygulanmıştır. Şekil 3.1'de Konya ili haritası ve ilçe merkezlerinin numaralandırılmış hali gösterilmiştir. Elde edilen bulgular ise Çizelge 3.1'de verilmiştir.

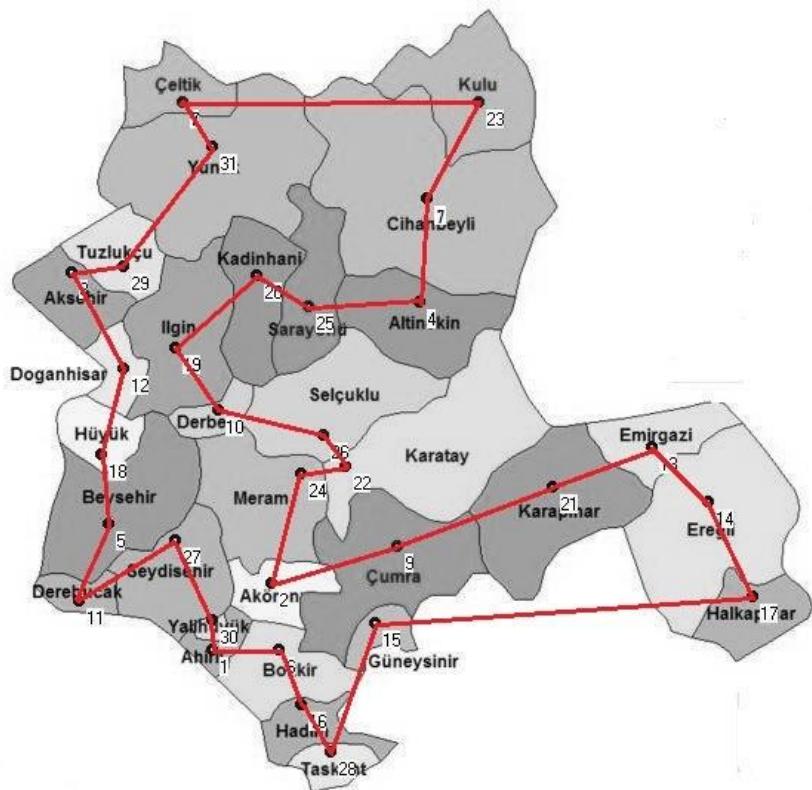
Çizelge 3.1. Konya ilçeleri için bulunan en iyi yol

	Bulunan En Kısa Yol	Ortalama Yol
YAKA	1459	1480
PSO	2250	2560,05
GA	2250	2478
HABC	2184,2	2198,24

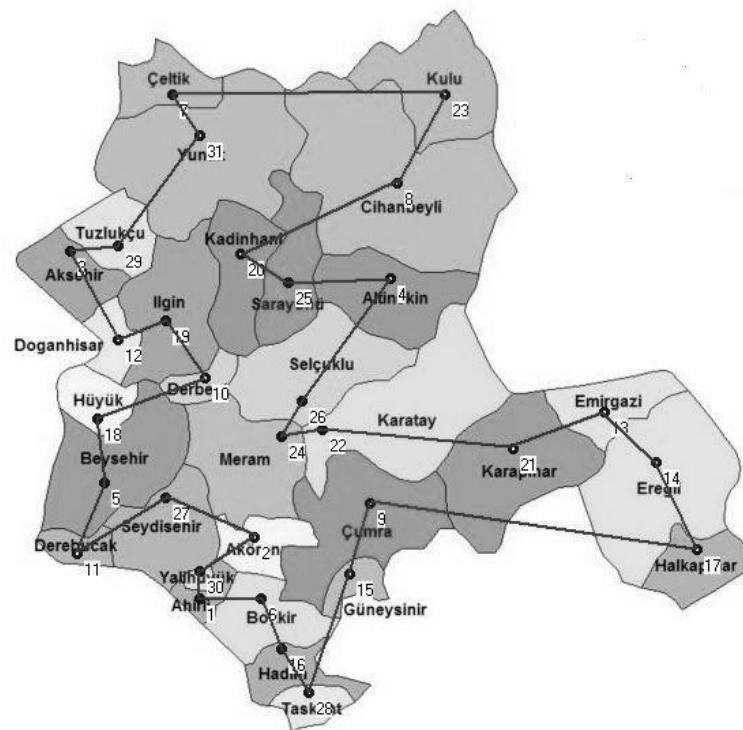
Çizelge 3.1'de görüldüğü üzere de Yapay Ari Kolonisi Algoritması ile elde edilen sonuçlar Hibrit Ari Kolonisi Algoritması, Parçacık Sürü Optimizasyonu ve Genetik Algoritmalar ile elde edilen sonuçlardan çok daha başarılıdır. Yapay Ari Kolonisi ile elde edilen en kısa yol Şekil 3.2'de gösterilmiştir. Genetik Algoritmalar ve Parçacık Sürü Optimizasyonu aynı olduğundan Şekil 3.3'de Hibrit Ari Kolonisi Algoritması ise Şekil 3.4'te gösterilmiştir.



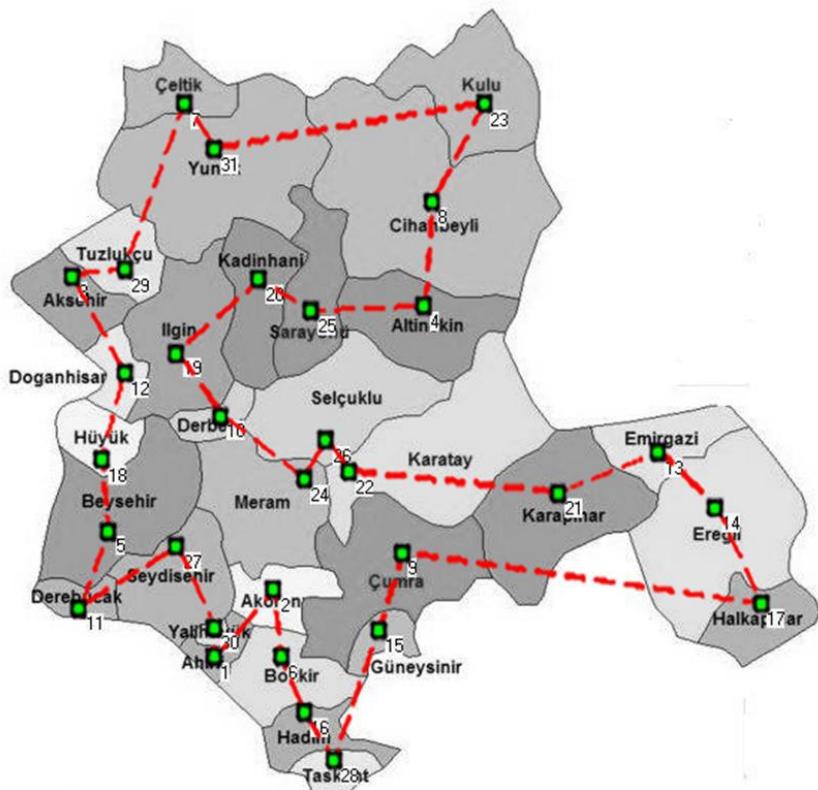
Şekil 3.1. Konya ilçeler haritası[10]



Şekil 3.2. Konya ilçeleri için Yapay Arı Kolonisi ile en iyi yol



Şekil 3.3. Konya ilçeleri için PSO ve GA ile en iyi yol[11].



Şekil 3.4. Konya ilçeleri için HABC ile en iyi yol[12]

4. SONUÇLAR ve TARTIŞMA

4.1. Sonuçlar

Bu çalışmada Yapay Arı Kolonisi Algoritmasının Konya ilçe merkezlerinin Gezgin Satıcı Problemine uygulanması gerçekleştirilmiştir. Bu çalışmada arı kolonisi optimizasyonu temel alınarak, gezgin satıcı problemi için bulunan yolların uygunluğu sadakat değerine göre belirlenmiş ve Yapay Arı Kolonisi Algoritması geliştirilmiştir. Yapay Arı Kolonisi ile elde edilen sonuçlar Özsağlam'a (2009) ve Akça'ya (2011) ait iki yüksek lisans tezindeki Genetik Algoritma, Hibrit Arı Kolonisi Algoritması ve Parçacıklı Süre Optimizasyonu algoritmalarının uygulanarak elde edilen sonuçlar ile karşılaştırılmış ve performans değerlendirilmesi yapılmıştır.

Konya ilçe merkezlerinin ilçeler arası mesafe cetveli Kara Yolları Genel Müdürlüğü'nün 2019 yılı verileridir. İlçeler arası mesafe cetveli Ek-1 de sunulmuştur.

Yapay Arı Kolonisi Algoritması tüm veri kümesi için 6 kez çalıştırılmıştır. Yapay Arı Kolonisi Algoritması ile bulunan en kısa yol bilgileri kaydedilmiş, Parçacıklı Süre Optimizasyonu, Genetik Algoritmalar ve Hibrit Yapay Arı Kolonisi Algoritması ile kıyaslandığında en iyi sonucu Yapay Arı Kolonisi Algoritmasının verdiği görülmüştür.

Gezgin satıcı problemlerine uygulanan Yapay Arı Kolonisi Algoritması ile elde edilen sonuçlar Çizelge 3.1'de verilmiştir. Yaptığımız uygulamada bilinen en iyi sonuca ulaşıldığı görülmüştür.

Konya ilinin 31 ilçe merkezine ait ilçeler arası mesafe cetveli verileri kullanılarak elde edilen sonuçlar Çizelge 3.1'de verilmiştir. Buna göre Yapay Arı Kolonisi Algoritması algoritması hem bulunan en kısa yol hem de ortalama yol bakımından Parçacıklı Süre Optimizasyonu, Genetik Algoritmalar ve Hibrit Arı Kolonisi Algoritması'na göre çok daha başarılı performans sergilediği görülmektedir.

Çizelge 3.1. Konya ilçeleri için bulunan en iyi yol

	Bulunan En Kısa Yol	Ortalama Yol
YAKA	1459	1480
PSO	2250	2560,05
GA	2250	2478
HABC	2184,2	2198,24

4.2. Tartışma

Yapılan uygulamalar sonucu elde edilen tüm sonuçlar incelendiğinde Yapay Arı Kolonisi Algoritması'nın çoğu test problemi için Parçacıklı Sürü Optimizasyonu, Genetik Algoritmalar ve Hibrit Arı Kolonisi Algoritması'na göre daha iyi bir çözüm ürettiği, bazı problemlerde ise literatürde bilinen en iyiden daha başarılı performans gösterdiği görülmüştür.

Gezgin satıcı test problemlerinde bilinen en iyiye her defasında ulaşarak %100 başarı elde edilmiştir.

Yapay Arı Kolonisi Algoritması'nın algoritması nokta sayısından bağımsız olarak noktaların yakınlık ve uzaklığından, yani diziliminden etkilenmektedir. Bunun sebebi olarak arıların besin bulma davranışından esinlenerek oluşturduğumuz en yakın rastgele noktaya gitme işlemi basamağı gösterilebilir. Nokta sayısı az olsa dahi eğer noktaların koordinatları birbirine çok yakınsa yapılan seçimde hatalar oluşabilmektedir.

KAYNAKÇA

- [1] Kuzu S., Önay O., Şen U., Tuncer M., Yıldırım B.F., Kesiktürk T., Gezgin Satıcı problemlerinin metasezgiseller ile çözümü, İstanbul Üniversitesi İşletme Fakültesi Dergisi, 2014, s. 2
- [2] Ateş E., Karınca Kolonisi Optimizasyonu Algoritmaları İle Gezgin Satıcı Probleminin Çözümü Ve 3 Boyutlu Benzetimi, Basılmamış Lisans Tezi, Ege Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, İzmir, 2012.
- [3] Özkan, S., 2010, Gezgin Satıcı Probleminin Çözümüne Yönelik Algoritmik Yaklaşımlar, Yüksek Lisans Tezi, Ankara
- [4] Holland J.H., 1975. Adaptation in Natural and Artificial Systems, The University of Michigan Press, Ann Arbor, MI.
- [5] Gonzales E.L. ve Fernandez M.A.R., 2000. “Genetic optimization of a fuzzy distribution model”, International Journal of Physical Distribution & Logistics Management, 30(7/8): 681-696.
- [6] Glover, F. (1986) “Future Paths for Integer Programming and Links to Artificial Intelligence,” Computers and Operations Research, Vol. 13, pp. 533-549.
- [7] Karaboga D., An Idea Based On Honey Bee Swarm For Numerical Optimization, TR-06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
- [8] Dorigo, M., Gambardella, L.M., “Ant Colonies for the Travelling Salesman Problem”, Biosystems, 43 (2), 73–81, 1997.
- [9] Dorigo, M., Gambardella L-M., “Ant Colonies for the Traveling Salesman Problem”, TR/IRIDIA/3, Université Libre de Bruxelles Belgium, 1996.
- [10] Özsağlam, M., Y., 2009, Parçacık Sürü Optimizasyonu Algoritmasının Gezgin Satıcı Problemine Uygulanması ve Performansının İncelenmesi, Yüksek Lisans Tezi, Konya
- [11] Özsağlam, M., Y., 2009, Parçacık Sürü Optimizasyonu Algoritmasının Gezgin Satıcı Problemine Uygulanması ve Performansının İncelenmesi, Yüksek Lisans Tezi, Konya
- [12] Akça, M. R., 2011, Hibrit Arı Kolonisi Algoritması Kullanarak Gezgin Satıcı Probleminin Türkiye’deki İl ve İlçe Merkezlerine Uygulanması, Yüksek Lisans Tezi, Konya