

GAZİ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ



BM469 – Genetik Algoritmalar ve Programlama

Travelling Salesman Problemi İçin, Heuristik Operatör ve
Jumping Gene İçeren Bir Genetik Algoritma

Can Kayabaş

Ögr. No: 201180767

İÇİNDEKİLER

1. Özet.....	3
2. Giriş.....	3
3. Yöntemler.....	4
4. Sonuçlar.....	4
5. Kaynaklar.....	6

1. Özet

Bu raporda "A genetic algorithm with jumping gene and heuristic operators for traveling salesman problem" isimli makalede önerilen TSP çözümünün bir implementasyonu anlatılmaktadır. Makalede, klasik genetik algoritma yaklaşımına ek olarak; (1) rulet tekeri seçimi için iterasyon sayısına bağlı olarak farklı fitness fonksiyonları kullanma, (2) çift-yönlü heuristik çaprazlama operatörü (BHX) kullanımı, (3) zamanla değiştirilen mutasyon operatörleri kullanımı, (4) çeşitlilik sağlamak için zıplayan gen operatörü kullanımı ve (5) oluşturulan popülasyonda aynı bireylerin olmasını önleyen özel bir operatör kullanımı önerilmiştir. Bunlarla birlikte, üretilen sonucun 2-opt algoritması ile iyileştirilmesi sağlanmıştır. Deney sonuçları ve istatistiksel analizler sonucunda bu yaklaşımın farklı metodlara göre (Örn. Ant Colony Optimization, Artificial Bee Colony Algorithm, Genetik Algoritma) stabilite, doğruluk ve yakınsama hızı açısından daha iyi sonuç verdiği görülmüştür.

2. Giriş

Optimizasyon problemi çeşitli disiplinlerde yaygın olarak kullanılmaktadır. Problemler sürekli ve kombinatoriyel optimizasyon problemleri olarak ikiye ayrılır ve gerçek dünya uygulamalarında çoğunlukla kombinatoriyel problemlerle karşılaşılır. TSP, birden çok şehir arasında en iyi rotayı bulmayı amaçlayan bir kombinatoriyel optimizasyon problemidir. Her şehir sadece bir kez ziyaret edilmeli ve sonunda ilk şehre dönülmelidir. Bu problem, NP-zor bir problem olarak kanıtlanmıştır ve tanımlanması kolay ancak çözümü zor olan bir problem olarak bilinir. Araç rotalama problemleri (VRP), planlama problemleri, entegre devre tasarım problemleri, fiziksel eşleme problemleri ve filogenetik ağaç yapılandırma problemleri gibi birçok pratik problem TSP şeklinde genelleştirilebilir. Bu nedenle, TSP üzerinde çalışmak büyük bir pratik öneme sahiptir.

TSP için güncel olarak geliştirilen yaklaşımlar şu şekilde özetlenebilir: dinamik programlama, böl ve yönet yöntemi, 2-Opt ve 3-Opt algoritması gibi yerel arama algoritmaları, sinir ağı (NN), genetik algoritma (GA), simulated annealing (SA), parçacık sürü optimizasyonu (PSO), karınca koloni optimizasyon algoritması (ACO) ve yapay arı koloni algoritması (ABCTSP) gibi zeka temelli optimizasyon algoritmaları (IOA). TSP için uygun bir çözümün boyutu, şehir sayısının artmasıyla üssel olarak artar; problem boyutu çok büyük olduğunda optimal bir çözüm bulmak zordur. Bu nedenle deterministik algoritmalar bu probleme genel bir çözüm geliştirmek için uygun değildir. Yerel optimizasyon algoritmaları çözüm kalitesini etkili bir şekilde iyileştirebilir, ancak yerel minimuma takılma eğilimindedir. Bununla birlikte, IOA makul bir sürede kabul edilebilir bir çözüm sunar, bu yüzden sezgisel ve metasezgisel yöntemler tercih edilen ve genellikle önerilen yöntemlerdir. ACO genellikle TSP çözümünde kullanılır ancak algoritma yerel minimuma sıkışma eğilimindedir.

GA, sürekli ve kesikli optimizasyon problemlerini çözmek için etkili bir yöntemdir. TSP'nin karar değişkenleri tamsayı ve tekrarsız olmalıdır, bu yüzden reel sayı ve binary kodlama bu problemi çözmek için uygun değildir. Ancak, sıralı kodlama ile oluşturulan kromozomlar TSP'nin kodlama kurallarını karşılamaktadır.

3. Yöntemler

GA'nın performansını artırmak amacıyla birçok yaklaşım geliştirilmiş ve kayda değer sonuçlar elde edilmiştir. GA'nın performansı üzerinde büyük etkisi olan çaprazlama operatörleri göz önüne alındığında, kısmi eşleşme çiftleşme (PMX), sıra çiftleşme (OX), döngü çiftleşmesi (CX) ve açgözlü çiftleşme (GX) gibi çeşitli çaprazlama operatörleri önerilmiştir. Bu operatörler, çocukların geçerliliğini garanti edebilir, ancak rastgelelikleri nedeniyle algoritmanın yakınsama hızını artırmada pek etkili olmazlar. Ek olarak, popülasyondaki birbirleriyle aynı olan kromozomların sayısı iterasyon sayısı ile birlikte artar ve çaprazlamaların verimliliğini azaltır; bu durum algoritmanın yakınsama hızına olumsuz etki eder. GA'nın performansını iyileştirmek amacıyla bu çalışmada GA-JGHO yöntemi sunulmaktadır. Algoritmanın performansını artırmak için beş strateji kullanılmaktadır. İlk olarak, fitness fonksiyonunun zamana göre değiştirilmesine dayalı yeni bir rulet seçimi operatörü tasarlanmıştır. Bu yaklaşımın bir yandan popülasyonu çeşitlendirmek, diğer yandan daha iyi kromozomları seçmek için olduğu söylenebilir. İkinci olarak, yakınsama hızını hızlandırmak için açgözlü yaklaşımı içeren BHX operatörü önerilmiştir; bu yöntem ile ebeveyn kromozomlarının iyi özelliklerinin mümkün olduğunca çocuklara geçmesi hedeflenir. Üçüncüsü, birbirleriyle aynı olan kromozomlar çiftleşmeye olumsuz etki yapabilir, bu nedenle çiftleşmede aynı kromozomların katılımını azaltmak için tekilleştirme operatörü eklenmiştir. Dördüncü olarak, popülasyon çeşitliliğini korumak ve arama durgunluğunu önlemek için kombinasyon mutasyon operatörü kullanılmaktadır. Son olarak, arama uzayının genişletilmesi hedeflenerek biyolojideki zıplayan gen kavramından esinlenen bir zıplayan gen operatörü önerilmiştir. Bu yöntemlere ek olarak, elde edilen sonuçların kalitesini artırmak için 2-Opt algoritması entegre edilir.

4. Sonuçlar

Önerilen GA-JGHO algoritması, 30 örnek üzerinde yapılan hesaplamaların ortalaması ve yakınsama grafikleri ile incelenen yakınsama hızı göz önünde bulundurulduğunda, girişte bahsedilen yaklaşımlara göre çok daha iyi performans göstermiştir. GA-JGHO'nun etkinliği ve üstünlüğü hem teoride hem de pratikte doğrulanmıştır. Bununla birlikte, GA-JGHO'nun hala belirli sınırlamaları bulunmaktadır. Örneğin GA-JGHO sadece simetrik TSP çözümleri için uygundur; iş zamanlama problemlerini, knapsack problemlerini ve sürekli optimizasyon problemlerini çözmek için uygun değildir.

Şekil 1, 2 ve 3'de GA-JGHO algoritması için makalede uygulanan deney sonuçları ve uygulama ile elde edilen sonuçlar açıklanmıştır.

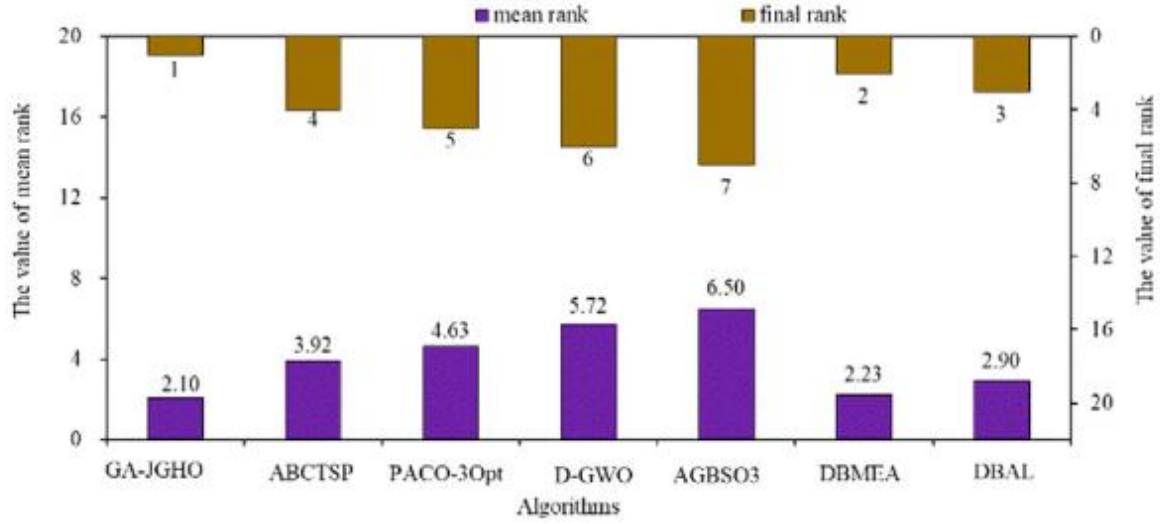
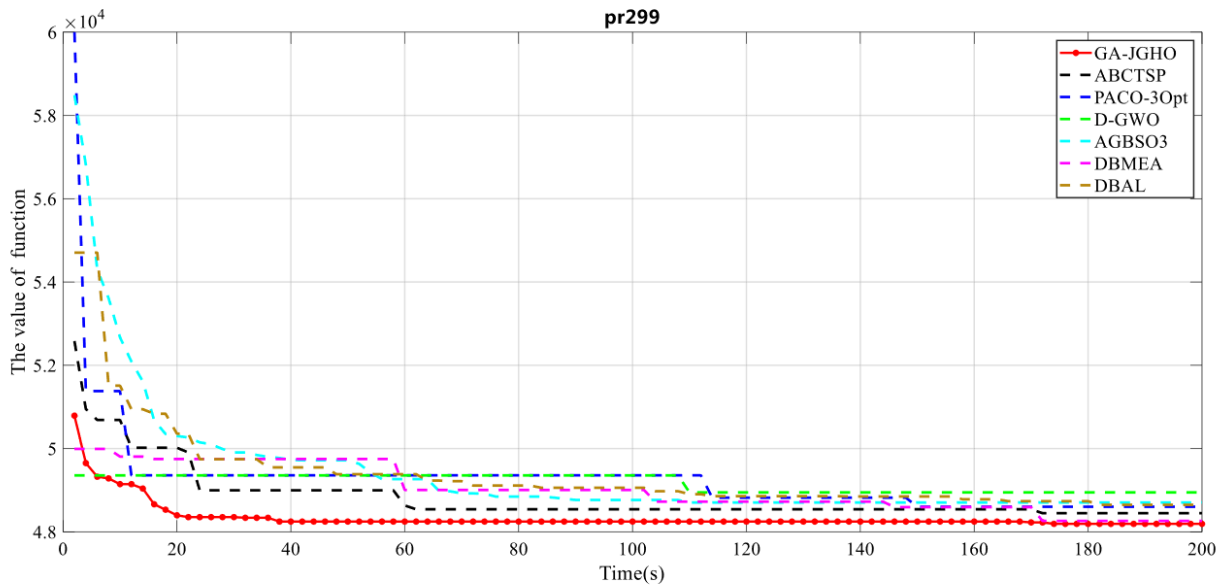
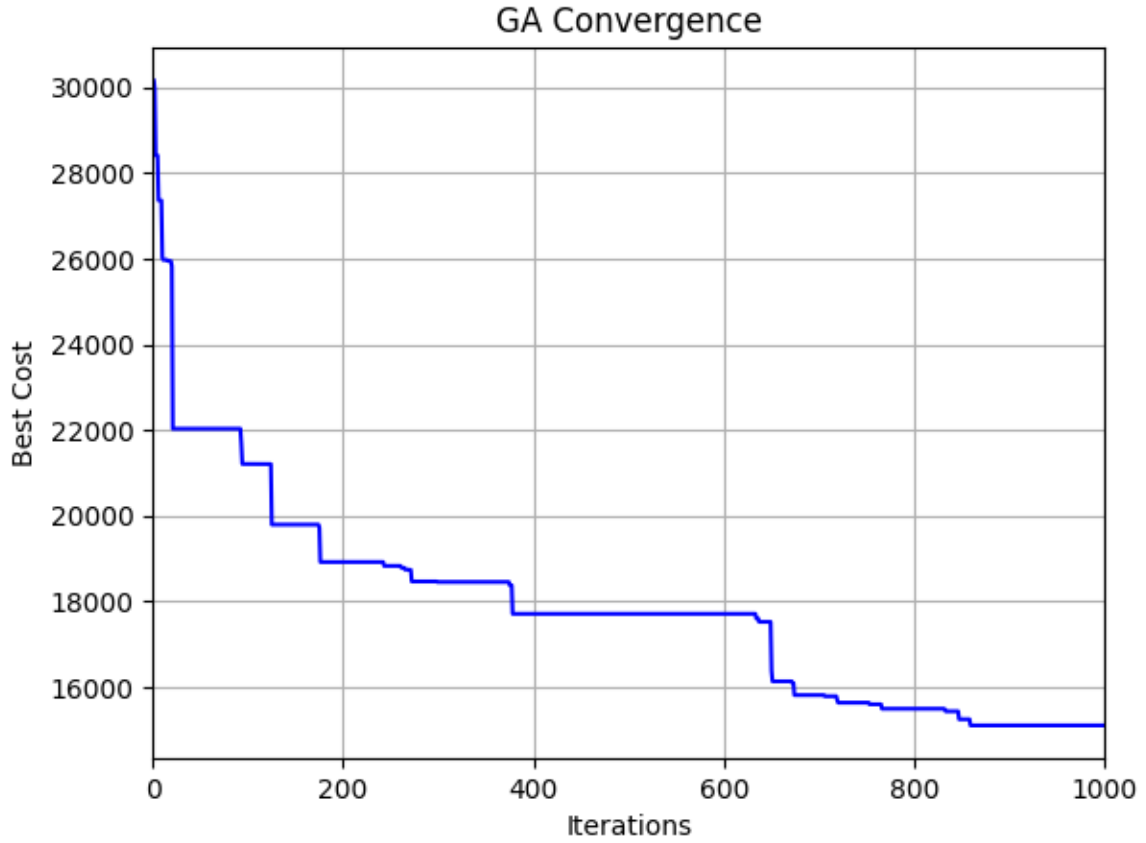


Fig. 12. The mean rank and final rank of algorithms.

Şekil 1. GA-JGHO ve diğer yöntemler ile elde edilen sonuçların sıralamaları



Şekil 2. GA-JGHO ve diğer yöntemlerin yakınsama hızları karşılaştırılması



Şekil 3. Paylaşılan uygulamadan elde edilen yakınsama sonuçları

5. Kaynaklar

- Panli Zhang, Jiquan Wang, Zhanwei Tian, Shengzhi Sun, Jianting Li, Jingnan Yang, "A genetic algorithm with jumping gene and heuristic operators for traveling salesman problem", Applied Soft Computing, Volume 127, 2022.