

Beale, Ackley, Goldstein, Levi Test Fonksiyonları İle Karınca Kolonisi Optimizasyonu ve Benzetimli Tavlama Algoritması Optimizasyonu 2020

Eda Altuntaş

Bursa Teknik Üniversitesi, 18360859036@ogrenci.btu.edu.tr

Özet

Optimizasyon, bir problem için amaç doğrultusunda çeşitli şartlar sağlanarak en optimum çözümün elde edilmesidir. Sezgisel optimizasyon algoritmaları, doğrusal olmayan sistemlerin optimizasyon problemlerinde fazlasıyla kullanılmaktadır Bu araştırmada 4 adet test fonksiyonu (Ackley, Beale, Goldstein Price, Levi) ile metasezgisel algoritmalar olan Benzetimli Tavlama Algoritması (BTA) ve Karınca Kolonisi Optimizasyonu(KKO) incelenip uygulanmıştır. Algoritmalar Python programlama diliyle kodlanmıştır. Sonuçlar grafiklerle gösterilmiştir.

1. Giriş

Optimizasyon, bir problemin belirli şartlar altında çözüm uzayındaki optimum çözüm noktasının bulunma sürecidir. Bir problemi optimize ederken çeşitli algoritmalarından yararlanılır. Bu algoritmalar, belirli kurallara ve işleyişe göre tanımlanan, birtakım matematiksel ifadeler yardımıyla problemleri çözmeye yöntemleridir.

Test fonksiyonları, optimizasyon algoritmalarının • Yakınsama oranı,

- Hassaslık,
- Sağlamlık,
- Genel performans,

özelliklerini değerlendirmek için kullanılmaktadır.

Bu çalışmada Ackley, Beale, Goldstein, Levi test fonksiyonları kullanılarak Karınca Kolonisi Optimizasyonu(KKO) ile Benzetimli Tavlama Algoritmasının(BTA) optimizasyonu gerçekleştirilmiş ve iki algoritma içinde test fonksiyonu sonuçları görselleştirilmiştir.

2. Benzetimli Tavlama Algoritması(BTA)

Benzetimli Tavlama Algoritmasının temeli oluşturulan fikirler, ilk olarak Metropolis tarafından 1953 yılında önerilmiştir. Daha sonra Kirkpatrick 1983 yılında, Metropolis' in geliştirmiş olduğu algoritmayı temel alarak kombinatoriyel eniyileme problemlerinde kullanılabilecek bir algoritma önermiştir Stokastik arama yöntemi olan BTA, katıların fiziksel tavlama süreci ile olan benzerlikten ileri gelmektedir(Katıların ısıtılması ve sonra yavaş yavaş soğutulması esasına dayanır). BTA' da amaç fonksiyonunda artışa neden olabilecek komşu hareketler bazen kabul edilerek yerel en iyi noktalarından kurtulmak mümkündür ve amaç fonksiyonunda artışa neden olabilecek bu komşu hareketin kabul edilip edilmemesi, rassal olarak belirlenmektedir. Amaç fonksiyonunda Δ kadar bir yükselmeye yol açan hareketin kabul edilme olasılığını veren fonksiyon kabul fonksiyonu olarak adlandırılır.

$$P(\text{accept}) = e^{-\left(\frac{\Delta}{T}\right)}$$

$\Delta = (\text{komşu çözüm} - \text{mevcut çözüm})$

Boltzmann dağılımı

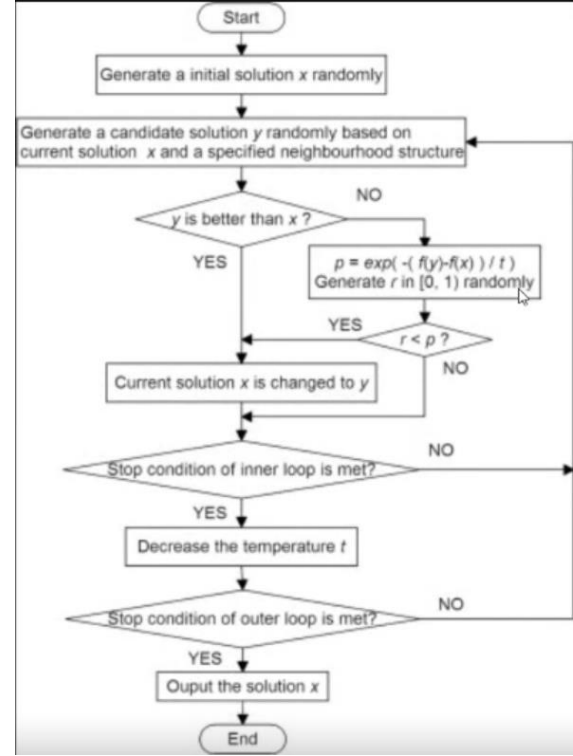
Sıcaklık yüksek olduğunda, amaç fonksiyonunda artışa neden olabilecek hareketlerin kabul edilme olasılığı çok yüksek olacak, sıcaklık düştükçe bu olasılık da azalacaktır. Bu sebeple, aramaya yeterli kadar yüksek bir sıcaklık değeri ile başlamak gereklidir. Algoritmada, sıcaklık yavaş yavaş azaltılırken, her sıcaklık değerinde belli sayıda hareket deneyerek arama işlemi sürdürülür.

Pseudo-code

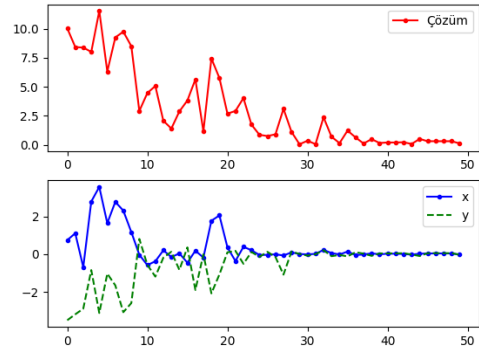
```

Bir başlangıç çözümünü seç:  $s_0 \in S$  ve amaç fonksiyonu  $f(s_0)$  hesapla;
Bir başlangıç sıcaklığını belirle:  $T > 0$ ;
Sıcaklık değişim sayacını sıfırla:  $t \leftarrow 0$ ;
 $s \leftarrow s_0$ ;  $f(s) \leftarrow f(s_0)$ ;
 $s_{\text{yeni}} \leftarrow s_0$ ;  $f(s_{\text{yeni}}) \leftarrow f(s_0)$ ;
Repeat
     $n \leftarrow 0$ ;
    Repeat
         $s'$ 'nin bir komşusu olan  $s'$  çözümünü ( $s' \in N(s)$ ) rassal olarak üret;
         $\Delta \leftarrow f(s') - f(s)$ ;
         $\Delta \leq 0$  ise  $s \leftarrow s'$ ;
        değilse  $(0,1)$  aralığında düzgün dağılımdan bir rassal sayı üret ( $u$ ) ve
             $u < \exp(-\Delta/T)$  ise  $s \leftarrow s'$ ;
         $f(s') < f(s_{\text{yeni}})$  ise  $s_{\text{yeni}} \leftarrow s'$ ;
         $n \leftarrow n + 1$ ;
    Until  $n > M$ 
     $t \leftarrow t + 1$ ;
     $T = T(t)$ ;
Until (durdurma koşulu sağlanana kadar)
 $s_{\text{yeni}}$  problem için bulunan sezgisel çözüm
    
```

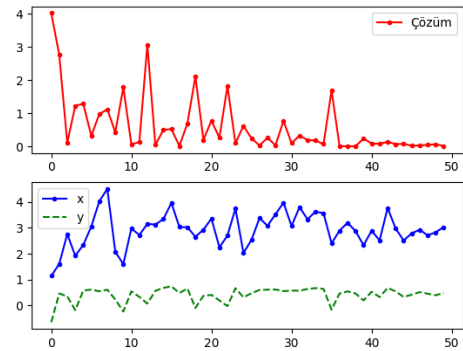
Flowchart



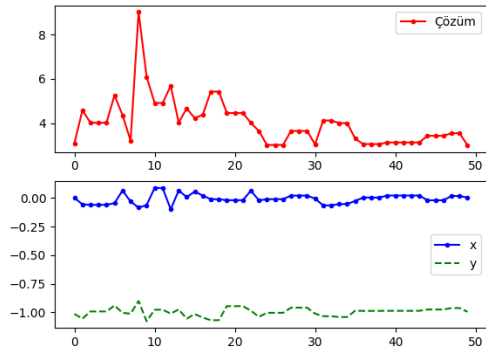
Ackley



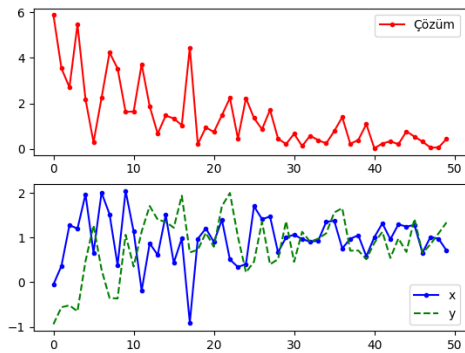
Beale



Goldstein-price



Levi



3. Karınca Kolonisi Optimizasyonu(KKO)

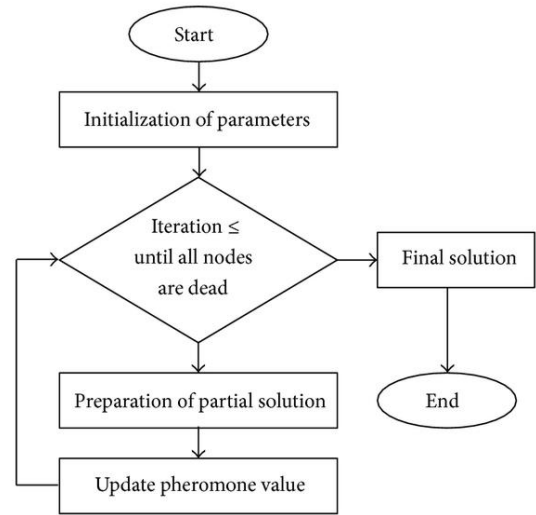
NP-hard optimizasyon problemlerine başarılı bir şekilde uygulanabilen popülasyon temelli bir yaklaşımdır. İlk olarak Gezgin Satıcı Problemi (Travelling Salesman Problem) (TSP) üzerinde uygulanmıştır. Bir noktadan başlayarak ve tüm noktalara bir kez uğrayarak tekrar aynı noktaya en kısa yoldan dönmek diye tanımlanan TSP'nin çözümünde KKO'nun oldukça etkili olduğu görülmüştür. KKO gerçek karınca kolonilerinin davranışlarından esinlenilerek 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 kimyasalıdır. Feromon izleri, karıncalar tarafından güncellenmekte ve bir bilgiyi temsil etmektedir. Bir yolda feromon izinin yoğun olması, yolun kalitesini gösterir ve tercih olasılığını artırır.

KKO'da, 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.

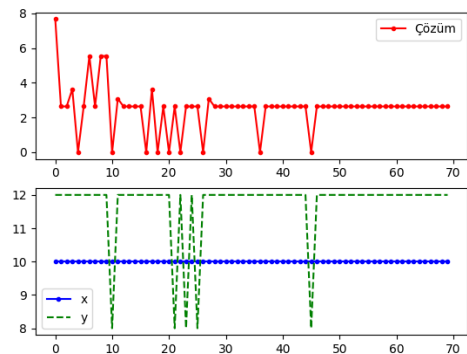
Pseudo-code

Adım 1 : Başlangıç feromon değerleri belirlenir.
Adım 2 : Karıncalar her düğüme rastsal olarak yerleştirilir.
Adım 3 : Her karınca, sonraki şehri denklemde verilen lokal arama olasılığına bağlı olarak seçmek suretiyle turunu tamamlar.
Adım 4 : Her karınca tarafından katedilen yolların uzunluğunu hesaplanır ve lokal feromon güncellemesi yapılır.
Adım 5 : En iyi çözüm hesaplanır ve global feromon yenilemesinde kullanılır.
Adım 6 : Maksimum iterasyon sayısı yada yeterlilik kriteri sağlanana kadar Adım 2'ye gidilir.

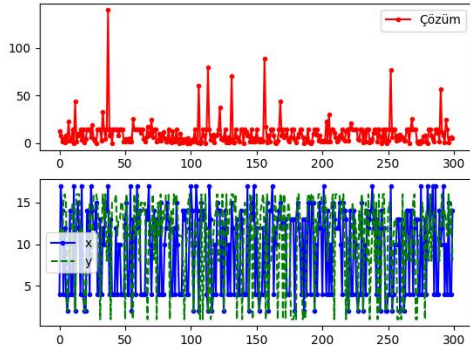
Flowchart



Ackley



Beale

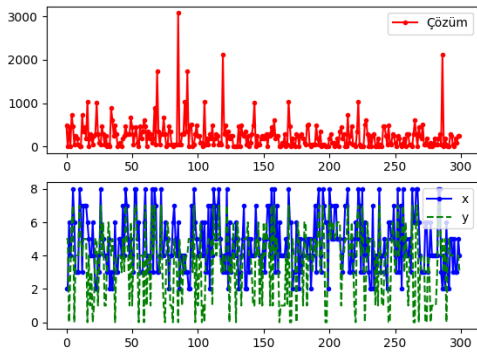


http://web.firat.edu.tr/iaydin/bmu579/bmu_579_bolum7.pdf

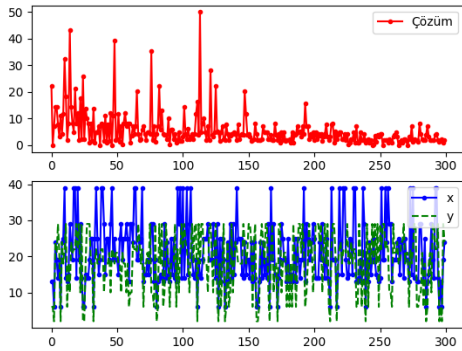
http://www.journalagent.com/iuyd/pdfs/IUYD-20592-RESEARCH_ARTICLE-AKCETIN.pdf

<https://eds.yildiz.edu.tr/AjaxTool/GetArticleByPublishedArticleId?PublishedArticleId=1735>

Goldstein-price



Levi



4. Kaynakça

[Benzetimli Tavlama \(Simulated Annealing\)](#)

[Algoritması – Global AI Hub](#)

<https://medium.com/@berkekeser/kar%C4%B1nca-kolonisi-optimizasyon-algoritmas%C4%B1-4da0b37cb393>

http://web.firat.edu.tr/iaydin/bmu579/bmu_579_bolum6.pdf