Beale, Ackley, Goldstein, Levi Test Fonksiyonları İle BalinaAlgoritması ve Benzetimli Tavlama Algoritması Optimizasyonu

Bursa Teknik Üniversitesi, [18360859001@ogrenci.btu.edu.tr](mailto:18360859001@ogrenci.btu.edu.tr)

# Özet

*Matematiksel test fonksiyonları optimizasyon algoritmalarını test etmek için kullanılan matematiksel formüllerdir[1]. Bu fonksiyonlar, optimizasyon algoritmalarını hem hız hem de bulunan çözüm performansı açısından karşılaştırmamıza olanak tanır. Bu çalışmada Ackley, Beale, Goldstein, Levi test fonksiyonları kullanılarak BalinaOptimizasyon Algoritması(GOA) ile Benzetimli Tavlama Algoritması (BT) optimizasyon işlemi gerçekleştirilmştir. Çalışma Pycharm ve Microsoft Visual Studio Code geliştirme ortamları kullanılarak Python programlama diliyle gerçekleştirilmiş olup iki optimizasyon algoritması karşılaştırılmıştır. Sonuçlar incelenerek Benzetimli Tavlama Algoritmasının BalinaAlgoritmasına göre daha az iterasyonla optimize hale ulaştığı gözlemlenmiştir.*

adlandırılmaktadır. Bu sezgisel algoritmalardan biri de çalışma içerisinde kullanılan Balina Optimizasyon Algoritması(WOA)’dır. Benzetimli Tavlama Algortiması ise en iyi çözümün en kısa zamanda üretimini hedefleyen, eniyileme problemi için tasarlanmış olasılıksal yaklaşımlı bir optimizasyon algoritmasıdır[3].

Test fonksiyonları, optimizasyon algoritmalarının

* Yakınsama oranı,
* Hassaslık,
* Sağlamlık,
* Genel performans,

özelliklerini değerlendirmek için kullanılmaktadır[4].

# Giriş

Optimizasyon, verilen amaç veya amaçlar için belirli kısıtlamaların sağlanarak en uygun çözümün elde edilme sürecidir. Optimizasyon problemlerinin çözümünde klasik yöntemler olarak adlandırılan matematiksel yöntemler önceleri çok yaygın olarak kullanılmaktaydı. Bu tür yöntemlerin esnek olmaması ve matematiksel fonksiyonlarla tanımlama gereksinimi gibi dez avantajları, son zamanlarda, bilim adamlarında genel amaçlı ve performansı yüksek yöntemler geliştirme çabalarını artırmış ve doğadaki olaylardan esinlenmeye başlamışlardır. Tabiattaki olaylar temel alınarak geliştirilen optimizasyon algoritmaları sezgisel yöntemler olarak

Bu çalışmada Ackley, Beale, Goldstein, Levi test fonksiyonları kullanılarak BalinaAlgoritması ile Benzetimli Tavlama Algoritmasının optimizasyon işlemi gerçekleştirilmiş olup, iki algoritma içinde test fonksiyonu sonuçları görselleştirilmiştir.

# Balina Algoritması

Algoritma buble net avlanma stratejisinden esinlenmiştir. Balina optimizasyon algoritması sayısal problemleri optimize etmek için oluşturulmuştur. Algoritma kambur balinaların akıllıca avlanma davranışlarını simule eder. Bu davranış buble net beslenme metodu olarak adlandırılır ve sadece kambur balinalarda gözlemlenmiştir. Balina avlanma sırasında avın etrafında daralan çember şeklinde bir yol izlerken

baloncukları çıkartır. Basitçe anlatmak gerekirse şöyle diyebiliriz. Kambur balina yaklaşık 12 metre derine dalar. sonra spiral şeklinde baloncuk çıkartarak yüzmeye başlar. Ardından baloncuklarla birlikte yüzeye doğru yüzmeye başlar.

## Avı Çevrelemek

Kambur balinalar avlarını bulur ve onları kuşatırlar. Balina şuan ki en iyi arama ajanının konumunu hedef av veya çok yakın bir konum olarak kabul eder. Diğer arama ajanları ise en iyi konuma göre yeniden pozisyon alırlar. Bu davranış şu şekilde formülleşmiştir.

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

 Mevcut iterasyon *t*  ile ifade ediliyor,  ise *t* ’inci iterasyonda elde edilen sonuçlara göre oluşturulmuş en iyi pozisyon vektörü,  ise arama ajanının konum konum vektörü.  ve  aşağıdaki şekilde hesaplanmaktadır:

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

 iterasyonlar sırasında 2 den 0 a doğru doğrusal olarak azalır ve [0,1] aralığında rastgele bir sayıdır.

## Baloncuk ağı saldırı metodu

Bu metod 2 farklı metodun hibrit halde kullanılması ile oluşur ve şu şekilde matematiksel olarak ifade edilebilir.

### **2.2.1. daralan çevreleme mekanizması**

 ‘ya [-1,1] arasında yeni rastgele değer vererek arama ajanının, en iyi pozisyon ile kendi konumu arasında rastgele bir konuma atar. Böylelikle arama alanımız ava doğru daralır

### **2.2.2 Spiral pozisyon güncelleme**

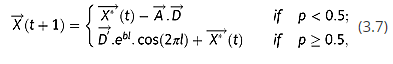
Bu yaklaşımda balina ve av arasında spiral bir hareket şekili oluşturmak için bir spiral denklem oluşturulur denklem şu şekildedir:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

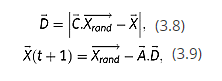
 balina ve av arasındaki mesafedir. *b* logaritmik şekli tanımlayan sabittir. *l* [-1,1] arasında rastgele bir sayıdır

Balinalar spiral şeklindeki yolu takip ederken aynı zamanda çemberi de daraltırlar bunu %50 ihtimalle rastgele olarak yaparlar matematiksel olarak şu şekilde gösterebiliriz: (p [0,1] arasında rastgele bir sayıdır)



## Avın aranması

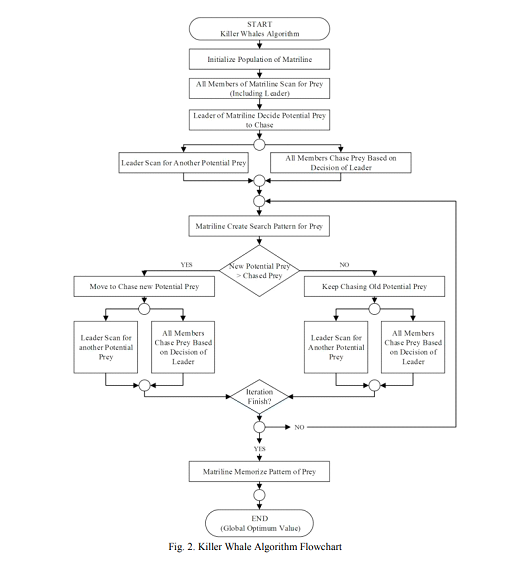
Neredeyse tüm sezgisel algoritmalar optimum değerleri rastgele olarak seçer. Baloncuk avı metodunda da pozisyonlar bilinmemektedir yani kambur balinalarda avlarını rastgele ararlar. Arama sırasında  vektörü [-1,1] harici değerler alır yani 1 den büyük -1 den küçük bu şekilde arama ajanı referans balinadan çok daha uzaklara gidebilir. Şu şekilde formulize edebiliriz:



* rastgele pozisyon vektörüdür

Balina rastgle çözümler ile başlar her iterasyonda arama ajanları konumlarını yuakrıda anlatılan yollarla günceller. Balina küresel bir optimize edicidir.

**2.4.Akış Şeması**

**–**

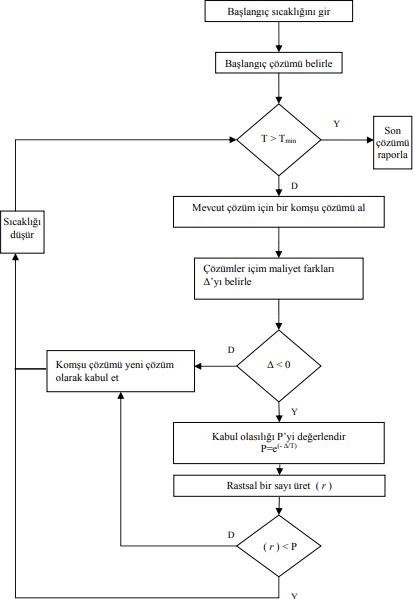
# 3. Benzetimli Tavlama

Benzetimli Tavlama Algoritması, yinelemeli rastgele bir arama tekniğidir ve çeşitli kombinasyonel optimizasyon problemlerini çözmek için yaygın olarak kullanılmaktadır. SA ilk olarak Metropolis, Rosenbluth ve Teller (1953) tarafından metallerin tavlama işlemini simüle etme amacıyla önerilmiştir ve Kirkpatrick, Gelatt ve Veechi (1983) tarafından yinelemeli bir optimizasyon yöntemi olarak tanıtılmıştır. Benzetimli Tavlama Algoritması önceden belirlenen bir başlangıç sıcaklığı ile başlar (T0). Bir başlangıç çözümü oluşturulur (S0) ve bu aynı zamanda geçerli çözüm (SC) ve ilk en iyi çözüm olur (SB). Başlangıç çözümünün amaç fonksiyonu değeri (E0), geçerli çözüm (EC) ve en iyi çözümün amaç fonksiyonu değeri olur (EB). Geçerli çözümden (SC) bir değişiklikle komşu çözüm (SN) oluşturulur ve amaç fonksiyon değerleri hesaplanır (ΔE). Eğer komşu çözümün amaç fonksiyonu değeri (EN), EC’den daha iyiyse, yeni geçerli çözüm olarak kabul edilir. Ancak EN EC’den daha kötüyse, exp(-ΔE/T) (T geçerli sıcaklık) olasılığı ile SN yeni SC olarak kabul edilir, diğer durumda SN reddedilir. T parametresi bir soğutma fonksiyonu (T= α x T) ile yavaşça azaltılır ve Tmin ulaşılacak son sıcaklıktır. TB algoritmasının kötü çözümleri kabul ederek yerel optimum noktalardan kurtulma kapasitesi vardır[10]

Benzetimli Tavlama Sözde Kodu[11]:

1. Giriş parametreleri: T0,Tmin, α, T = T0.
2. Bir S0 üret ve E0’ı hesapla.
3. SC = S0, SB = S0, EC = E0, EB = E0.
4. SC’den bir SN üret ve EN’i hesapla
5. ΔE = EN - EC.
6. Eğer ΔE ≤ 0 komşu çözümü geçerli çözüm olarak Kabul et, SC = SN, EC = EN ve Adım 9’a git
7. Eğer ΔE > 0 rnd isminde [0, 1] arasında bir değişken üret.
8. Eğer exp (-ΔE/T) > rnd ise komşu çözümü geçerli çözüm olarak Kabul et, SC = SN , EC = EN ; diğer durumda, SC ve E değişmeden kalır ve Adım 10’a git.
9. Eğer EC < EB ise SB = SC; EB = EC; diğer durumda, SB ve EB değişmeden kalır.
10. T = T × α
11. Eğer T ≥ Tmin ise Adım 4’e git ; diğer durumda, SB ve EB’yi raporla ve bitir.

Benzetimli Tavlama Algoritması’na ait akış diyagramı Şekil 3’dedir.



*Şekil 3. Benzetimli Tavlama Algoritması akış diyagramı[12]*

# 4. Yöntem

Çalışma Spider platformunda Python programlama dili kullanılarak yapılmıştır. Balina Algortiması için kullanıcıdan optimizasyonda kullanılacak test fonksiyonunu algoritma içinde çalışma zamanından önce belirtimelidir. Benzetimli Tavlama Algoritması içinde kullanıcıdan optimizasyonda kullanılacak test fonksiyonunu belirlemesi istenmiştir.. Sonuç olarak iki algoritmada, optimizasyon sonucunda kullanılan test fonksiyonlarına göre optimum çözüm değerleri(x,y) ve bu çözüm değerlerine göre fonksiyon sonucunu(f(x,y)) çıktı olarak ekrana vermektedir. Aynı zamanda her iterasyonda optimizasyon işlemi gerçekleşirken çözüm değerleri ile fonksiyon sonuçları kaydedilerek bu değerlerin değişimleri kodun sonunda grafiksel olarak gösterilmiştir..

# Sonuç

Beale, Ackley, Goldstein ve Levi test fonksiyonları kullanılarak Balina ve Benzetimli Tavlama optimizasyon algoritmaları uygun girdi değerleriyle çalıştırıldığında her iki algoritmanında fonksiyonlar için minimize değerlere(fmin) çok yakın değerler üretebildiği görülmektedir. Buradan her iki algoritma içinde başarılı bir şekilde optimizasyon işlemlerini gerçekleştiren algoritmalar olduğu söylenebilmektedir. İterasyon sayılarına bakıldığında Benzetimli Tavlama Algoritması’nın BalinaAlgoritmasına göre çok az adım sayısında değerleri optimize edebildiği görülmektedir. Buradan BT’nin GOA’ya göre daha hızlı çalışan bir optimizasyon algortiması olduğu anlaşılmaktadır.

# Kaynaklar

<https://www.tandfonline.com/doi/full/10.1080/25742558>.2018.1483565#:~:text=Clustering%20is%20a%20powerful%20technique,on%20the%20values%20of%20attributes.&text=Simulations%20show%20that%20the%20proposed,be%20used%20for%20data%20clustering.