



T.C

SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BSM 310 – YAPAY ZEKA

Grup üyeleri:

B171210008 Affet Can

B171210026 Rabia Kaynar

G181210374 Ümmü Gülsüm AVCI

Sakarya
2020

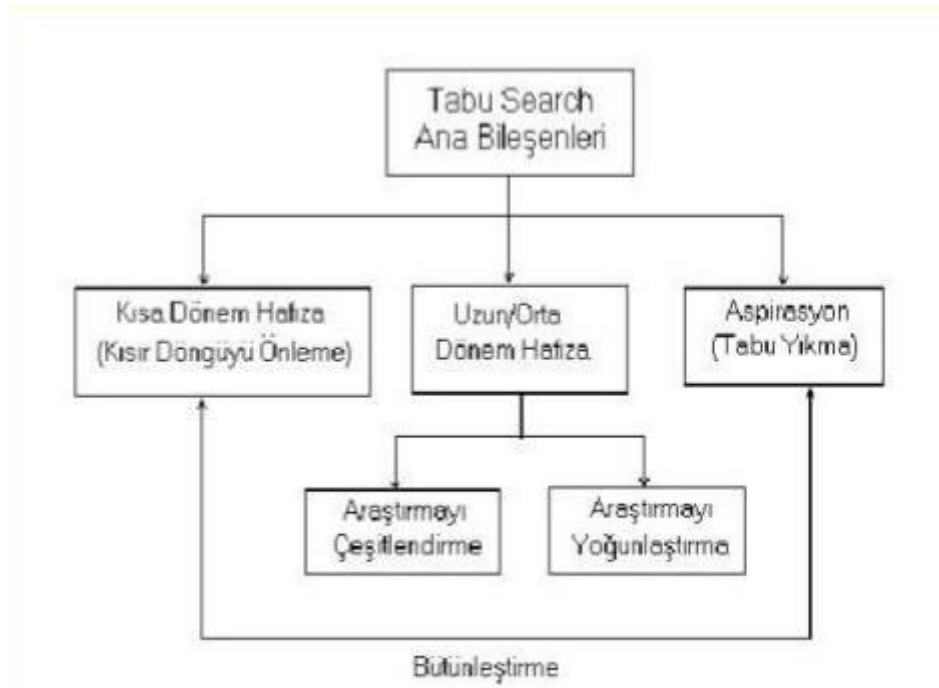
GİRİŞ

- Tabu Search Algoritması, F.Glover tarafından optimizasyon problemlerinin çözümü için geliştirilmiş iteratif bir araştırma algoritmasıdır.
- Tabu Search algoritması, bölgesel en iyi çözümün daha ilerisinde bulunan çözümlerin araştırılabilmesi için bölgesel-sezgisel araştırmaya kılavuzluk etmektedir.
- TS ‘nin bölgesel optimalliği aşmak amacıyla kullandığı temel prensip, değerlendirme fonksiyonu tarafından her iterasyonda en yüksek değerlendirme değerine sahip hareketin bir sonraki çözümü oluşturmak amacıyla seçilmesine dayanmaktadır.
- Yerel prosedür, verilen herhangi bir çözümün komşuluğunu tanımlamak için move olarak söylenen bir işlemi kullanan bir araştırmadır.
- Temel yaklaşım, 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 araştırılabilmesini sağlar.
- Tabu Arama algoritmasının temel prensibi, değerlendirme fonksiyonu tarafından her iterasyonda en yüksek değerlendirme değerine sahip hareketin bir sonraki çözümü oluşturmak amacıyla seçilmesine dayanmaktadır.
- Bunu sağlamak amacıyla bir tabu listesi oluşturulur, tabu listesinin amacı önceden yapılmış bir hareketin tekrarından çok tersine dönmesini önlemektir. Tabu listesi kronolojik bir yapıya sahiptir ve esnek bir hafıza yapısı kullanır.
- Tabu Arama’ nın temel bileşenlerinden birisi daha esnek bir araştırma durumu oluşturan kendi (adaptive) uyarlama hafızasını kullanmasıdır.
- Global optimizasyon için genel bir prosedür olan TS sezgisel algoritması bir çok zor optimizasyon problemleri için neredeyse optimale yakın çözümler üreten başarılı bir tekniktir .
- Optimizasyon problemlerin çözümü için çizelgelemede, tesis düzenleme probleminde, kuadratik atama problemi, grafik bölümlendirme, grafik renklendirmegibi problemlerde kullanılır.

Min $c(x)$
 $x \in X$

- Amaç fonksiyonu $c(x)$ maliyet veya kar fonksiyonunun en küçük veya en büyük değerini aramaktır fakat x vektörü ile belirtilen kısıtlamalara uyularak çözüme ulaşılabacaktır.
- Başka bir ifade ile her x elemanı bir hareketi temsil eder ve tüm hareketler X ile gösterilmektedir. X vektörleri tabu arama bellek yapısı olarak kullanılır. Böylece vektörde tutulan bellek değerine bağlı olarak çözüm aramada bazı hareketler tabu olarak kabul edilip engellenecek, bazılarına ise daha fazla odaklanacaktır.
 X vektöründeki her bir hareket ise mevcut çözümün bir komşusunun seçimini temsil eder.

Tabu Arama Algoritması İşleyiş Şekli



Başlangıç çözümünün oluşturulması

- En genel şekilde başlangıç çözümü rastsal olarak elde edilir. Ancak ilgilenilen, problem için geliştirilmiş olan bir sezgisel alitmadan yararlanarak da başlangıç çözümünden elde edilmesi mümkündür.

Hareket Mekanizması

- Mevcut bir çözümde yapılan bir değişiklikle yeni bir çözümün elde edilmesi hareket mekanizmasıyla gerçekleştirilir. Hareket mekanizmasındaki olası hareketler, mevcut çözümün komşularını oluşturur.

Komşuluk

- Tabu Aramada en önemli bileşenlerden birisi de komşuluk yapısıdır. Çözümü iyileştirmek için amaç fonksiyonun değeri açısından en iyi hareketlerin seçilmesidir. Komşulukların oluşturulmasında seçilen komşuluk üretme yapısına göre problem boyutu n olduğu durumda $(n-1)$ tane komşuluk üretilir.

Hafıza

- Tabu arama algoritmasının temel elemanlarından biride hafızadır. Arama boyunca ortaya çıkan durumlar, H hafızasına kayıt edilir. Yapılmasına izin verilmeyen hareketler "tabu" olarak adlandırılır ve esnek hafıza içinde "tabu listesi" adı altında kaydedilirler. Bu hareketler belli bir süre sonra tabu listesinden çıkarılır ve yapılmasına izin verilir.

- İki tür hafıza bulunmaktadır.

Hafıza Türleri

Yakınlık Geçmişe dayalı Bellek Yapısı (Kısa Dönem Hafıza)(YBY):

En temel TA bellek yapısıdır. Esas olarak görevi yakın geçmişte yapılan hareketlerin bir süre için tabu olarak işaretlemektir. Burada kast edilen süre tabu süresi yani çevrim sayısıdır. Belirlenen süre(t) kadar hareket tabu olarak belirlenir ve daha sonra belirlenen hareket tabu olmaktan çıkarılır.

Örneğin $t=5$ olacak şekilde statik olarak değer verilebildiği gibi t değeri min t ve max t sınırları arasında sistematik veya rastgele seçilebilir.

Sıklığa Dayalı Bellek Yapısı (Orta ve Uzun Dönem Hafıza)(SBY):

Sıklığa dayalı bellek yapısı tamamlayıcı bir özelliğe sahiptir genellikle YBY ile birlikte ikincil bir bellek olarak kullanılır. SBY hareketlerin sıklığını bellekte tutar. Ancak bir

hareketin kaç kez yapıldığı bilgisi yerine, çözümün kalitesine ve hareketin etkisine bağlı bilgilerin bellekte tutulması daha faydalıdır. Sıklık ölçütünün dört çeşidi vardır:

- a) Her bir hareketin toplam tekrar sayısı
- b) Toplam hareket sayısı
- c) En yüksek hareket tekrar sayısı
- d) Ortalama hareket tekrar sayısı

Tabu Listesi

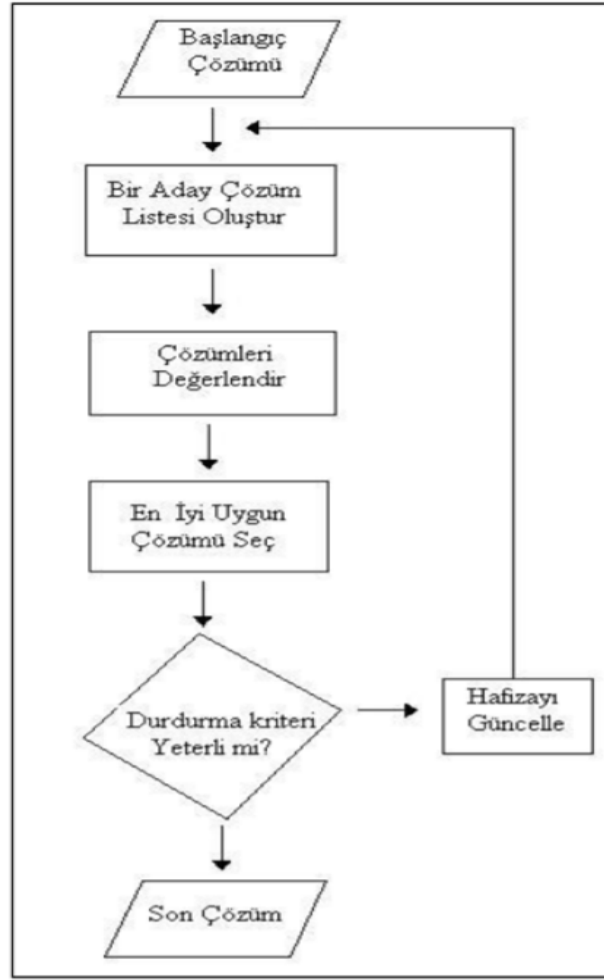
Tabu listesi, araştırmanın herhangi bir iterasyonu içinde hangi seçimlerin tabu grubu olmak zorunda olduğu ve kaç tanesi üzerinde karar verme ve tabu listesini güncelleme ile ilgilenir. Tabu listesinin boyutu, sonucu önemli derecede etkileyebilir. Problemin boyutu büyüdükçe tabu listesinin uzunluğuda büyür.

ASPIRASYON (tabu yıkma):

Algoritmanın temel özellikleri korunurken esnekliğini artırmak, yerel optimumdan kaçınmak ve kısır döngüden kurtulmak maksadıyla yasak konumundan (tabu konumundan) geçici olarak serbest bırakılması konumuna aspirasyon işlemi denir.

- Durdurma Kriteri: Durdurma kriterleri aşağıdaki şartlara dayandırılır: • 1. Araştırmada iterasyon sayısını verilen maksimum iterasyon sayısından daha büyük olduğunda işlem sonlandırılır.

- 2. Eğer o ana kadar bulunan en iyi çözümde arka arkaya yapılan iterasyonda hiç bir iyileşme olmadan verilen iterasyon sayısına ulaşırsa işlem sonlandırılır.



Şekil 1: Tabu arama algoritması

TA algoritması, bir başlangıç çözümü ile aramaya başlar. Algoritmanın her iterasyonunda tabu olmayan bir hareket ile mevcut çözümün komşuları içerisinde bir tanesi seçilerek değerlendirilir. Eğer amaç fonksiyonunun değerinde bir iyileştirme sağlanmışsa komşu çözüm, mevcut çözüm olarak dikkate alınır. Seçilen bir hareket tabu olmasına rağmen tabu yıkma kriterlerini sağlıyorsa, mevcut çözümü oluşturmak için uygulanabilir. Geriye dönüşleri önlemek için, bir takım hareketler tabu listesine kaydedilerek tekrar yapılması belirli bir süre için yasaklanır. Belirlenen bir durdurma koşuluna göre algoritmanın çalışması sonlanmaktadır

Örnek Tabu Arama Algoritması

Çözüm=Başlangıç çözümü,

En_İyi_Çözüm=Çözüm,

Tabu_Listesi (Boş),

Durdurma_Kriteri,

Kontrol=FALSE,

Repeat

– Eğer

Çözüm>En_İyi_Çözüm ise En_İyi_Çözüm=Çözüm – Eğer Durdurma_Kriteri’ne ulaşılmış ise Çözümü Tabu_Listesine ekle

• Eğer Tabu_Listesi dolu ise ilk gireni listeden çıkar,

çözümlerin içinden başka birini Yeni_Çözüm olarak seç

• Eğer Yeni_Çözüm bulunamadıysa veya

(Eğer geliştirilen Yeni_Çözüm, Uzun_Dönem_Hafıza da bulunuyor ise Yeni_Çözümü rasgele üret)

• Eğer Yeni_Çözüm, Tabu_Listesi’nde yok ise Çözüm=Yeni_Çözüm

Değil ise Kontrol=TRUE

Until Kontrol=TRUE

ÖRNEK:

Bir işletme, değişik tipte dört yeni makina satın almıştır. İşletme içinde makinaların yerleştirilebileceği dört seçenek yer bulunmaktadır. Birim zaman başına malzeme aktarma giderleri her makina için öngörölmüş ve aşağıda çizelge biçiminde sunulmuştur

		Aday Konumlar			
		1	2	3	4
Yeni Makinalar	A	180	10	96	146
	B	138	28	166	172
	C	114	186	4	158
	D	14	154	150	46

Amaç Tabu Arama Algoritmasını kullanarak en düşük maliyetli veya en düşük maliyetli çözüme yakın makine yerleştirmeyi elde etmek.

Başlangıç konumu ve değeri (başlangıç olarak en iyi yerleştirme)

DCBA : $14+186+166+146=512$ olsun

En iyi çözüm : 512

İterasyon sayısı : 5

Tabu Listesi uzunluğu : 3

Tabu Listesi : $((),(),())$

Komşuluklar: 1. iterasyon

CDBA : $114+154+166+146=580$

BCDA : $138+186+150+146=620$

ACBD : $180+186+166+46=578$

En iyi çözümden daha iyi bir çözüm yok bu nedenle bölgesel çeşitlendirme yapmak için en iyi yerleştirme konumundaki makinelerden 2 tanesinin yeri rasgele değiştirilir. Bu değişim 2. ve 4. konumdaki makineler olsun.

DABC : $14+10+166+158=348$

En iyi yerleştirme : DABC

En iyi Çözüm : 348

Tabu Listesi : $(a,c),(),()$

En iyi yerleştirme : DABC

En iyi Çözüm : 348

Tabu Listesi : $(a,c),(),()$

Komşuluklar: 2. İterasyon

ADBC : $180+154+166+158=658$

BADC : $138+10+150+158=456$

CABD : $114+10+166+46=336$

En iyi yerleştirme : CABD

En iyi Çözüm : 336

Tabu Listesi : $(a,c),(c,d),()$

En iyi yerleştirme : CABD

En iyi Çözüm : 336

Tabu Listesi : $(a,c),(c,d),()$

Komşuluklar: 3. İtersayon

ACBD : $180+186+166+46=578$

BACD : $138+10+4+46=198$

DABC : $14+10+166+158=348$

En iyi yerleştirme : BACD

En iyi Çözüm : 198

Tabu Listesi : (a,c),(c,d),(b,c)

Komşuluklar: 4. İterasyon

ABCD : $180+28+4+46 = 258$

CABD : $114+10+166+46 = 336$

DACB : $14+10+4+172 = 200$

En iyi çözümden daha iyi bir çözüm yok bu nedenle bölgesel çeşitlendirme yapmak için en iyi yerleştirme konumundaki makinelerden 2 tanesinin yeri rasgele değiştirilir. Bu değişim 2. ve 4. konumdaki makineler olsun.

BDCA : $138+154+4+146=442$

Komşuluklar: 5. İterasyon

DBCA : $14+28+4+146 = 192$

CDBA : $114+28+166+146 = 580$

ADCB : $180+154+186+172 = 692$

En iyi yerleştirme : DBCA

En iyi Çözüm : 192

Tabu Listesi : (c,d),(b,c),(d,b)