

Zeki Optimizasyon Teknikleri Intelligent Optimization Techniques

Hazırlayan: M. Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

Bu dersin sunumları, "Singiresu S. Rao, Engineering Optimization: Theory and Practice, Wiley, 2009." kitabı kullanılarak hazırlanmıştır.

İçerik

- ▶ Tavlama benzetimi algoritması
- ▶ Memetik algoritma

Tavlama benzetimi algoritması

- ▶ **Metallerin tavlama sürecinden esinlenerek geliştirilmiştir.**
- ▶ **Metallerin ısıtılarak şekil verilmesi ve kontrollü bir şekilde soğutularak şeklin kalıcı hale getirilmesi** sürecine benzetim yapılmıştır.
- ▶ Tavlama benzetimi **sürekli fonksiyonların optimizasyonunda** veya **kombinatoriyal optimizasyon problemlerinde** kullanılabilir.
- ▶ **Soğutma süresi uzun olursa** algoritma **global optimuma yakınsar.**
- ▶ **Komşu çözüm üretme maliyeti** diğer algoritmalara göre genellikle **daha düşüktür.**

3

Tavlama benzetimi algoritması

- ▶ **Isının yüksek olduğu durumlarda** çözüm uzayında **geçişler kolay yapılır** (yüksek ısıda metale kolay şekil verme).
- ▶ **Isı düştükçe** çözüm uzayında **lokal bölgelerde arama yapılır** (düşük ısıda metalin şekli çok değişmez).
- ▶ **Isı** belirlenen **eşik değerin altında düştüğünde** optimum değerin elde edilmesi amaçlanır.
- ▶ Isı değerine bağlı **rastgele bir çözüme geçme olasılığı her zaman vardır.**

4

Tavlama benzetimi algoritması

- ▶ Kirkpatrick ve arkadaşları tarafından **1983 yılında önerilmiştir.**
- ▶ Optimizasyon problemleri için iyi çözümler veren **olasılıklı bir arama tekniğidir.**
- ▶ **Sıcaklık değeri** elde edilen en iyi çözümden **daha kötü bir çözümün kabul edilme olasılığını belirlemek için kullanılır.**
- ▶ **Düşük ısı değerleri için** bir çözüme **rastgele geçiş olasılığı düşük olur.**

5

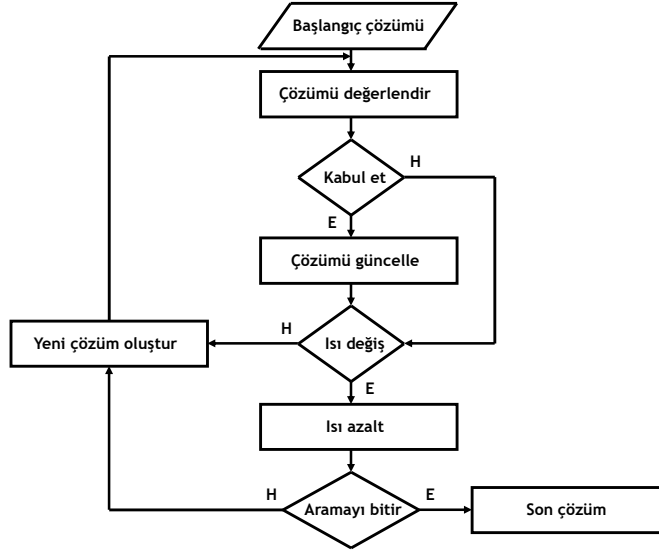
Tavlama benzetimi algoritması

- ▶ Tavlama benzetimi algoritması **yüksek bir sıcaklık değeriyle başlar.**
- ▶ Her bir hesaplama adımında **mevcut çözümün komşuları arasından çok sayıda çözüm üretilir.**
- ▶ **Yeni çözümler** belirlenen kriterlere göre **kabul edilir veya reddedilir.**
- ▶ Her bir hesaplama adımından sonra **sıcaklık belirlenen bir fonksiyona göre azaltılır.**
- ▶ Algoritmayı sonlandırma şartları:
 - ▶ **İstenen iterasyon sayısına ulaşılması**
 - ▶ **Sıcaklığın minimum değerine ulaşması**
 - ▶ **İstenen bir çözüme ulaşılması**

6

Tavlama benzetimi algoritması

Tavlama benzetimi akış diyagramı



7

Tavlama benzetimi algoritması

İşlem Adımları

Başla

- Başlangıç çözümü seç
- Başlangıç sıcaklığı seç (t=100)
- Sıcaklık azaltma fonksiyonu belirle

Tekrarla

Tekrarla

- Yeni bir komşu çözüm üret
- Eğer (yeni - eski) < 0 ise yeni çözümü seç
- Değilse

[0, 1] aralığında rassal sayı üret (r)

Eğer $r < \left(\frac{1}{\exp(\text{abs}(\text{yeni} - \text{eski})/t)} \right)$ ise yeni çözümü seç

$$r < \left(\frac{1}{e^{\frac{\text{yeni} - \text{eski}}{t}}} \right)$$

Son Eğer

Son Tekrarla iterasyon sayısına kadar

t = f(t);

Son Tekrarla (t < 0) veya uygun çözüm bulununcaya kadar

Bitir

8

Tavlama benzetimi algoritması

Pseudocode for Simulated Annealing

```
Input: ProblemSize,  $iterations_{max}$ ,  $temp_{max}$   
Output:  $S_{best}$   
1  $S_{current} \leftarrow \text{CreateInitialSolution}(\text{ProblemSize});$   
2  $S_{best} \leftarrow S_{current};$   
3 for  $i = 1$  to  $iterations_{max}$  do  
4    $S_i \leftarrow \text{CreateNeighborSolution}(S_{current});$   
5    $temp_{curr} \leftarrow \text{CalculateTemperature}(i, temp_{max});$   
6   if  $\text{Cost}(S_i) \leq \text{Cost}(S_{current})$  then  
7      $S_{current} \leftarrow S_i;$   
8     if  $\text{Cost}(S_i) \leq \text{Cost}(S_{best})$  then  
9        $S_{best} \leftarrow S_i;$   
10    end  
11  else if  $\text{Exp}(\frac{\text{Cost}(S_{current}) - \text{Cost}(S_i)}{temp_{curr}}) > \text{Rand}()$  then  
12     $S_{current} \leftarrow S_i;$   
13  end  
14 end  
15 return  $S_{best};$ 
```

9

Tavlama benzetimi algoritması

Parametreler

- ▶ Başlangıç sıcaklığı
- ▶ Her sıcaklıkta üretilecek çözüm sayısı fonksiyonu
- ▶ Sıcaklık azaltma fonksiyonu
- ▶ Algoritmayı durdurma şartı

10

Tavlama benzetimi algoritması

Her sıcaklıkta üretilecek çözüm sayısı

- $N_k = C$ (Sabit)
- $N_k = N_{k-1} + C$ (Aritmetik)
- $N_k = N_{k-1} / a$ ($a < 1$) (Geometrik)
- $N_k = C / \text{Log}(T_k)$ (Logaritmik)
- $N_k = (N_{k-1})(1/a)$ ($a < 1$) (Üstel)

11

Tavlama benzetimi algoritması

Sıcaklık azaltma fonksiyonu

- $T_k = T_{k-1} - C$ (Aritmetik)
- $T_k = T_{k-1} * a$ ($a < 1$ ve 1'yakın) (Geometrik)
- $T_k = C / (k+1)$ (Ters fonksiyon)
- $T_k = C / (\text{Log}(k+1))$ (Logaritmik)

12

Tavlama benzetimi algoritması

Algoritmanın durdurulması

- ▶ Belirlenen maksimum iterasyona ulaşılması
- ▶ Belirlenmiş minimum sıcaklığa ulaşılması
- ▶ İstenen kriterleri sağlayan çözüme ulaşılması
- ▶ Belirli bir süre en iyi çözümde iyileşme sağlanamaması

13

İçerik

- ▶ Tavlama benzetimi algoritması
- ▶ Memetik algoritma

14

Memetik algoritma

- ▶ **Memetik algoritma**, genetik algoritma gibi **popülasyon tabanlıdır**.
- ▶ **Genetik algoritma** ve **benzeri popülasyon tabanlı algoritmalar** elde edilen çözümü **optimum çözüme yaklaştırmak** için daha **hassas ayarlama yapmaz**.
- ▶ **Memetik algoritma**, popülasyon tabanlı **global arama yöntemleri** ile **lokal arama yöntemlerini birleştirerek kullanır**.
- ▶ **Genetik algoritma** ile **hill climbing gibi lokal arama algoritması** birleştirilebilir.

15

Memetik algoritma

- ▶ Memetik algortmada, **yeni popülasyon bir sezgisel ile oluşturulabilir**.
- ▶ Her **sonraki yeni popülasyon** mevcut **iyi bireyler ile oluşturulur**.
- ▶ Kasık genetik algortmadaki gibi, **bireyleri seçme, çaprazlama, mutasyon** ile yeni bireyler oluşturulur.
- ▶ Her **yeni birey lokal arama algoritmalarından birisi ile iyileştirilir**.

16

Memetik algoritma

Memetik algoritma işlem adımları

1. Encode solution space
2. (a) set pop_size, max_gen, gen=0;
(b) set cross_rate, mutate_rate;
3. initialize population
4. while(gen < gensize)
 Apply generic GA
 Apply local search
end while
Apply final local search to best chromosome

17

Memetik algoritma

Genetik algoritma işlem adımları

1. Encode solution space
2. (a) Set pop_size, max_gen, gen=0
(b) set cross_rate, mutate_rate;
3. initialize population
4. while max_gen \geq gen
 evaluate fitness
 for (i=1 to pop_size)
 select (mate1,mate2)
 if (rnd(0,1) \leq cross_rate)
 child = crossover(mate1,mate2)
 if (rnd(0,1) \leq mutate_rate)
 child = mutation();
 repair child if necessary
 end for
 Add offspring to new generation
 Gen=gen+1
End while
5. return best chromosomes

18

Memetik algoritma

Hill climbing lokal arama algoritması işlem adımları

```
While (termination condition ins not satisfied) do
    New solution ← neighbors(best solution);
    If new solution is better then actual solution then
        Best solution ← actual solution
    End if
End while
```

19

Memetik algoritma

- ▶ Memetik algoritmada kullanılan **global arama geniş çözüm aralığında aramayı sağlar.**
- ▶ **Lokal arama bireysel her yeni çözümün iyileştirilmesini sağlar.**
- ▶ Memetik algoritmada, **popülasyondaki bireyler, global arama algoritmasındaki (genetik algoritma) bireyler ile aynı olabilir.**
- ▶ Memetik algoritmada, global arama algoritmasındaki **iyi bireylerden bir grup seçilerek başka yeni bir popülasyon oluşturulabilir.**
- ▶ Memetik algoritma, **kısıtlı, kombinatoryal veya sürekli fonksiyonların optimizasyonunda** kullanılabilir.

20

Memetik algoritma

Pseudocode for the Memetic Algorithm

```
Input: ProblemSize,  $Pop_{size}$ ,  $MemePop_{size}$   
Output:  $S_{best}$   
1 Population  $\leftarrow$  InitializePopulation(ProblemSize,  $Pop_{size}$ );  
2 while  $\neg$ StopCondition() do  
3   foreach  $S_i \in$  Population do  
4      $S_{i_{cost}} \leftarrow$  Cost( $S_i$ );  
5   end  
6    $S_{best} \leftarrow$  GetBestSolution(Population);  
7   Population  $\leftarrow$  StochasticGlobalSearch(Population);  
8   MemeticPopulation  $\leftarrow$  SelectMemeticPopulation(Population,  
    $MemePop_{size}$ );  
9   foreach  $S_i \in$  MemeticPopulation do  
10     $S_i \leftarrow$  LocalSearch( $S_i$ );  
11  end  
12 end  
13 return  $S_{best}$ ;
```

21

Ödev

- Tavlama benzetimi algoritmasının bilgisayar ağları (en kısa yol bulma, yönlendirme vb.) veya bilgi güvenliği (identification systems vb.) alanlarında uygulamasını içeren bir makale araştırma ödevi hazırlayınız.

22