CENG 415 Evrimsel Hesaplama

Bölüm 5: Uygunluk, Seçim ve Popülasyon Yönetimi

Şevket Umut Çakır

Pamukkale Üniversitesi

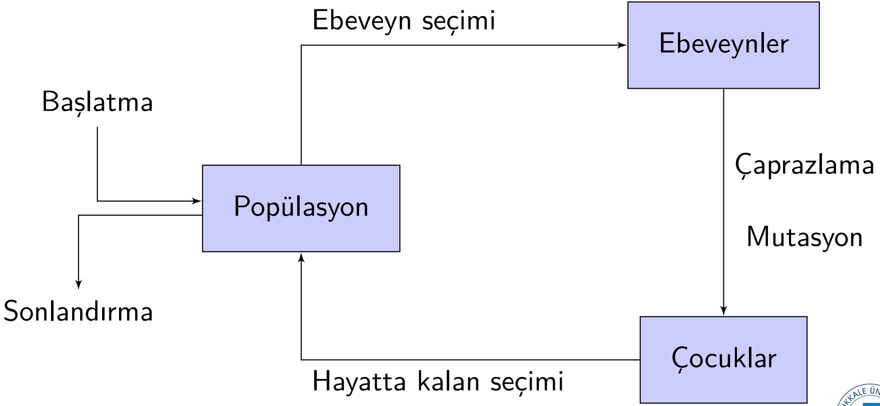
25 Kasım 2020

Uygunluk, Seçim ve Popülasyon Yönetimi

- Seçim, evrimsel sistemler için ikinci temel güçtür
- Bileşenler şunlardan oluşur:
  - Popülasyon yönetim modelleri
  - Seçim operatörleri
  - Çeşitliliği korumak

Evrimsel Hesaplama Şeması

Genel Şema



Popülasyon Yönetim Modelleri

Giriş

- İki farklı popülasyon yönetim modeli mevcuttur:
  - Kuşak(generational) modeli:
    - Her birey tam olarak bir nesil boyunca hayatta kalır
    - Tüm ebeveyn kümesinin yerini yavrular alır
    - μ bireyli popülasyonda λ yavru üretilir(tipik olarak $\frac{\lambda}{\mu} \in [5, 7]$)
  - Kararlı durum(steady-state) modeli:
    - Nesil başına bir yavru üretilir
    - Popülasyonun bir üyesi değiştirilir
    - Genellikle $\lambda < \mu$, tipik olarak $\lambda = 1$
- Nesil Boşluğu(Generation Gap):
  - Değiştirilen popülasyon oranı
  - Kuşak modeli için $= 1.0$, kararlı durum modeli için $= \frac{1}{popülasyon\ boyutu}$

Poülasyon Yönetim Modelleri

Uygunluk(Fitness) Temelli Rekabet

- Seçim iki yerde gerçekleşebilir:
  - Çiftleşmede yer almak için mevcut nesilden seçim (ebeveyn seçimi)
  - Gelecek nesile gitmek için {ebeveynler + yavrulardan} seçim (hayatta kalan seçimi)
- Seçim operatörleri tüm birey üzerinde çalışırlar
  - Yani temsilden bağımsızdırlar
- Seçim işlemleri arasındaki ayrım:
  - **Operatörler**: Seçim olasılıklarını belirler
  - **Algoritmalar**: Olasılıkların nasıl uygulanacağını belirler

Ebeveyn Seçimi

Uygunluk-Orantılı Seçim(Fitness-Proportionate Selection, FPS)

- μ boyutlu popülasyondaki i bireyinin FPS ile seçilme olasılığı:

$$P_{FPS}(i) = \frac{f_i}{\sum_{j=1}^{\mu} f_j}$$

- Sorunlar:
  - Nüfusun geri kalanı çok daha az uygunsa, çok formda bir üye hızla baskınlaşabilir: **Erken Yakınsama**(Premature Convergence)
  - Uygunluk değerlerinin benzer olduğu koşuların/nesillerin sonunda seçim baskısı kaybolur
  - Amaç fonksiyonu değişimine oldukça duyarlı(sonraki slayt)
- Ölçeklendirme son iki problemi çözebilir:
  - Pencereleme(windowing): $f'(i) = f(i) - \beta^t$
    β, son n nesildeki en kötü uygunluk(fitness) değeri
  - Sigma ölçekleme(sigma scaling): $f'(i) = max(f(i) - (\bar{f} - c \cdot \sigma_f), 0)$
    c genellikle 2 olarak seçilen bir sabit, $\bar{f}$: ortalama f, σ_f: standart sapma

Ebeveyn Seçimi

Uygunluk-Orantılı Seçim(Fitness-Proportionate Selection, FPS)

Individual	Fitness for f	Sel. prob. for f	Fitness for $f+10$	Sel. prob. for $f+10$	Fitness for $f+100$	Sel. prob. for $f+100$
A	1	0.1	11	0.275	101	0.326
B	4	0.4	14	0.35	104	0.335
C	5	0.5	15	0.375	105	0.339
Sum	10	1.0	40	1.0	310	1.0

Şekil: Uygunluk fonksiyonunun değişimi ile seçim olasılıklarının değişmesi

Ebeveyn Seçimi

Sıralamaya Dayalı(Rank-based) Seçim

- Seçim olasılıkları **mutlak uygunluk yerine göreli uygunluğa** dayandırılarak FPS sorunları ortadan kaldırılmaya çalışılır
- Popülasyonu uygunluğa göre sıralayın ve ardından sıralamaya göre seçim olasılıklarını temel alın (en iyi birey $\mu - 1$, en kötü birey 0 sırasına sahiptir)
- Bu, algoritmaya bir sıralama ek yükü getirir, ancak bu genellikle uygunluk değerlendirme süresine kıyasla ihmal edilebilir düzeydedir

Sıralamaya Dayalı Seçim

Doğrusal Sıralama(Linear Ranking)

$$P_{lin\_rank}(i) = \frac{2-s}{\mu} + \frac{2i(s-1)}{\mu(\mu-1)}$$

- s faktörü ile parametrelendirilmiş: $1 < s \leq 2$
  - En iyi bireyin avantajını ölçer
- Basit 3 elemanlı örnek

Individual	Fitness	Rank	P_{selFP}	P_{selLR} $(s = 2)$	P_{selLR} $(s = 1.5)$
A	1	0	0.1	0	0.167
B	4	1	0.4	0.33	0.33
C	5	2	0.5	0.67	0.5
Sum	10		1.0	1.0	1.0

Sıralamaya Dayalı Seçim

Üstel Sıralama(Exponential Ranking)

$$P_{exp\_rank}(i) = \frac{1 - e^{-i}}{c}$$

- Doğrusal Sıralama, seçim baskısında sınırlıdır
- Üstel Sıralama, en uygun kişiye 2'den fazla kopya ayırabilir
- Normalleştirme faktörü c, olasılıkların toplamı bir olacak şekilde seçilir, yani popülasyon büyüklüğünün bir fonksiyonu

Çiftleşme havuzu, seçim olasılık dağılımları ile örneklenir(rulet çarkı, stokastik evrensel örnekleme)

Ebeveyn Seçimi

Rulet Çarkı Algoritması

Input: Kümülatif olasılık dağılımı **a**

Result: Çiftleşme havuzundan λ eleman seçilir

```
mevcut ← 1
while mevcut ≤ λ do
    [0,1] aralığından tekörnek olarak rastgele r sayısını seç
    i ← 1
    while a_i < r do
        i ← i + 1
    end
    havuz[mevcut] ← ebeveyn[i]
    mevcut ← mevcut + 1
end
```

Algorithm 1: Rulet çarkı algoritması

Ebeveyn Seçimi

Rulet Çarkı Algoritması Örnek

Eleman	A	B	C	D	E
Olasılık	0.1	0.25	0.15	0.2	0.3
Kümülatif(a)					

$\lambda = 3,\ r_1 = \quad ,\ r_2 = \quad ,\ r_3 =$

Ebeveyn Seçimi

Stokastik Evrensel Örnekleme(Stochastic Universal Sampling)

Input: Kümülatif olasılık dağılımı **a**
Result: Çiftleşme havuzundan λ eleman seçilir
$mevcut \leftarrow 1$
$i \leftarrow 1$
$[0, \frac{1}{\lambda}]$ aralığından tekörnek olarak rastgele **r** sayısını seç
while $mevcut \leq \lambda$ **do**
  while $r \leq a_i$ **do**
    $havuz[mevcut] \leftarrow ebeveyn[i]$
    $r \leftarrow r + \frac{1}{\lambda}$
    $mevcut \leftarrow mevcut + 1$
  end
  $i \leftarrow i + 1$
end

Algorithm 2: Stokastik evresel örnekleme algoritması

Ebeveyn Seçimi

Stokastik Evrensel Örnekleme Algoritması Örnek

Eleman	A	B	C	D	E
Olasılık	0.3	0.2	0.15	0.2	0.15
Kümülatif(a)					

$\lambda = 3$, $r =$, $r \in [0, \frac{1}{3}]$

Ebeveyn Seçimi

Turnuva Seçimi

- Önceki tüm yöntemler küresel popülasyon istatistiklerine dayanır
  - Özellikle paralel makinelerde, popülasyonun büyük olduğu durumlarda darboğaz oluşabilir
  - Var olmayan harici uygunluk işlevinin varlığına dayanır: ör. gelişen oyun oyuncuları
- Yalnızca yerel uygunluk bilgisini kullanan bir prosedür için fikir:
  - Rastgele k üye seçin ve bunlardan en iyisini seçin
  - Daha fazla birey seçmek için tekrarlayın

Ebeveyn Seçimi

Turnuva Seçimi

Result: μ bireyin bulunduğu havuzdan λ üye seçilir
mevcut $\leftarrow 1$
while *mevcut* $\leq \lambda$ **do**
  Rastgele **k** bireyi seçin, yer değiştirme olabilir/olmayabilir
  Bu **k** bireyi karşılaştırın ve en iyisini seçin
  Bu bireyi ***i*** olarak işaretleyin
  havuz[*mevcut*] $\leftarrow$ *i*
  mevcut $\leftarrow$ *mevcut* $+ 1$
end

Algorithm 3: Turnuva seçimi algoritması

Ebeveyn Seçimi

Turnuva Seçimi

- i'yi seçme olasılığı şunlara bağlı olacaktır:
  - i'nin sırası
  - Örnekleme boyutu k
    - yüksek k değeri seçim basıncını artırır
  - Yarışmacıların değiştirilerek seçilip seçilmeyeceği
    - Değiştirmeden seçme, seçim basıncını artırır
  - En uygun(iyi) yarışmacı her zaman kazanır mı(deterministik) yoksa bu bir p olasılığı ile mi gerçekleşir

Ebeveyn Seçimi

Tekörnek(Uniform)

$$P_{uniform}(i) = \frac{1}{\mu} \tag{1}$$

- Operatörler bir ebeveyne ihtiyaç duyduğunda, ebeveynler tekörnek rastgele dağılımla seçilirler
- Tekörnek ebeveyn seçimi tarafsızdır, her birey aynı seçilme olasılığına sahiptir
- Çok büyük popülasyonlarla çalışırken aşırı seçim kullanılabilir
  - Popülasyon uygunluk değerine göre sıralanır
  - %x ve %(100-x) olarak iki parçaya ayrılır
  - Seçilenlerin %80'i ilk parçadan, kalan %20'si de ikinci parçadan seçilir

Hayatta Kalan Seçimi

- Evrimsel algoritmanın çalışma belleğini μ ebeveyn ve λ çocuktan oluşan kümeden, μ bireye indirme sürecidir
- Ebeveyn seçim mekanizmaları, hayatta kalanları seçmek için de kullanılabilir
- Hayatta kalan seçimi iki yaklaşıma ayrılabilir:
  - Yaşa dayalı seçim
    - Uygunluk hesaba katılmaz
    - Kararlı durum genetik algoritmalarda(SSGA) “rastgele silme(delete random)”(önerilmez) veya "ilk giren ilk çıkar(fifo)"(en eskiyi sil) olarak uygulanabilir
  - Uygunluk tabanlı değişim

Hayatta Kalan Seçimi

Uygunluk Tabanlı Değiştirme

- Elitizm(Elitism)
  - O ana kadarki en iyi(en uygun) çözümün bir kopyası saklanır
  - Her iki popülasyon modelinde de(kuşak, kararlı durum) yaygın olarak kullanılır
- GENITOR(en kötüyü sil)
  - Whitley'in orjinal kararlı durum algoritmasından[3](aynı zamanda ebeveyn seçimi için doğrusal sıralama kullandı)
  - Hızlı devralma: büyük popülasyonlarla ya da "kopya yok" politikası ile kullanılmalı

Hayatta Kalan Seçimi

Uygunluk Tabanlı Değiştirme

- Round-Robin Turnuva
  - $P(t)$: μ ebeveyn, $P'(t)$: μ yavru
  - Round-robin biçiminde ikili yarışmalar:
    - $P(t) \bigcup P' t()$ kümesindeki her **x** çözümü, rastgele seçilen **q** diğer çözüme karşı değerlendirilir
    - Her karşılaştırma için, x rakibinden daha iyiyse bir "galibiyet" atanır
    - En fazla galibiyete sahip μ çözümleri, gelecek neslin ebeveynleri olmaya devam eder
  - **q** parametresi seçim basıncının ayarlanmasına izin verir
  - Tipik olarak $q = 10$

Hayatta Kalan Seçimi

Uygunluk Tabanlı Değiştirme

- (μ, λ) seçim
  - Yalnızca çocuk kümesine göre($\lambda > \mu$)
  - En iyi μ tanesini seç
- $(\mu + \lambda)$ seçim
  - Ebeveyn ve çocukların kümesine göre
  - En iyi μ tanesini seç
- Genellikle (μ, λ) seçimi aşağıdakiler için tercih edilir
  - Yerel optimumdan ayrılmada daha iyi
  - Hareketli optimumu takip etmede daha iyi
  - $+$ stratejisini kullanıldığında kötü σ değerleri, eğer x değerleri çok uygunsa, $< x, \sigma >$ içinde çok uzun süre hayatta kalabilir
- $\lambda \approx 7 \cdot \mu$ geleneksel olarak iyi bir ayardır(son yıllarda oran biraz azalmıştır, $\lambda \approx 3 \cdot \mu$ son zamanlarda daha popüler)

Seçim Basıncı

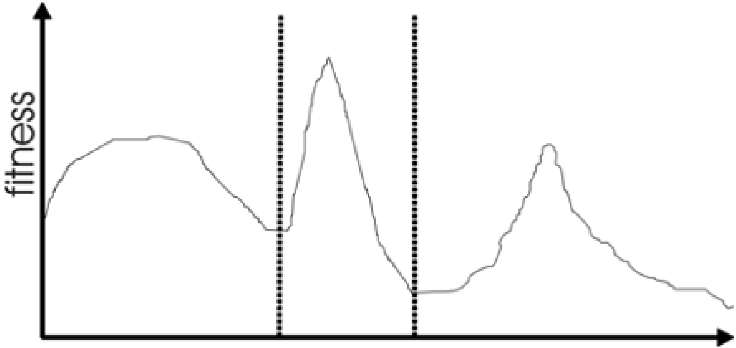
- **Seçim basıncı**: Daha iyi(uygun) bireylerin ebeveyn olma veya hayatta kalma olasılığının fazla olması
- Devralma(takeover) süresi $\tau*$, seçim basıncını ölçmek için bir ölçüdür
- Popülasyonu en iyi bireyin kopyalarıyla dolduran kadar geçen süre
- Goldberg ve Deb[2] formülleştirmiştir:

$$\tau* = \frac{ln\lambda}{ln(\mu/\lambda)} \tag{2}$$

- Bir genetik algoritmada uygunluk orantılı seçim(FPS) için devralma süresi $\tau* = \lambda ln\lambda$

Multimodalite/Çok Modluluk(Multimodality)

- En ilginç problemler birden fazla yerel optimuma sahiptir



Multimodalite/Çok Modluluk(Multimodality)

Genetik Sürüklenme

- Popülasyon boyutunun sonlu olması, herhangi iki ebeveyn arasında çaprazlama yapılabilmesi **genetik sürüklenme** olarak bilinen fenomene yol açar ve bir optimum etrafında yakınsar
- Neden?
- Genellikle birkaç olası zirveyi belirlemek isteyebilir
- Alt-optimum daha çekici olabilir

Çeşitliliği Koruma Yaklaşımları

Giriş

- Açık ve Örtülü
- Örtülü yaklaşımlar
  - Bir coğrafi ayrım eşdeğerini dayatır
  - Türleşme eşdeğerini dayatır
- Açık yaklaşımlar
  - Kaynaklar için rekabet eden bireyler oluşturur(uygunluk)
  - Hayatta kalmak için birbirleriyle mücadele eden bireyler oluşturur

Çeşitliliği Koruma Yaklaşımları

Giriş

- Farklı uzaylar
  - Genotip uzayı
    - Temsil edilebilen çözümlerin kümesi
  - Fenotip uzayı
    - Sonuç
    - Komşuluk yapısı genotip uzayındaki ile çok az benzerlik gösterir
  - Algoritmik uzay
    - Dünyadaki yaşamın evrildiği coğrafi alanla eşdeğerdir
    - Aday çözümlerden oluşan popülasyon yapılandırılabilir(popülasyonu işlemcilere veya çekirdeklere dağıtmak)

Çeşitliliği Korumak için Açık Yaklaşımlar

Uygunluk Paylaşımı(Fitness Sharing)

- Verilen bir nişteki[1] bireylerin sayısı, seçim sonrası uygunluk değerlerinin paylaşımı ile kontrol edilir.
- Bireylerin nişlere yerleştirilmesi niş uygunluğuna oranlı olarak yapılır
- Genotip veya fenotip uzayda niş boyutu σ_{share}'in ayarlanması gerekir
- Evrimsel algoritmayı normal çalıştır, ancak her nesil kümesinden sonra

$$F'(i) = \frac{F(i)}{\sum_j sh(d(i,j))'}$$

- $d(i,j)$, i ve j bireyleri arasındaki uzaklık(örn. ikili temsil için Hamming uzaklığı)

$$sh(d) = \begin{cases} 1 - (d/\sigma_{share})^{\alpha} & eğer\ d \leq \sigma_{share} \\ 0 & aksi\ halde \end{cases}$$

- α, paylaşım fonksiyonunun şeklini belirler, $\alpha = 1$ için doğrusal

[1] **niş**: yaşamda veya istihdamda rahat veya uygun bir pozisyon, oyuk

Çeşitliliği Korumak için Açık Yaklaşımlar

Uygunluk Paylaşımı(Fitness Sharing)

- Eğer $d < \sigma_{share}$ için $sh(d) = 1$ kullansaydık, uygunluğu azaltan toplam, basitçe komşuların sayısını, yani σ_{share}'den daha yakın bireyleri sayacaktır
- Bu komşulukta yalnız kalma avantajı oluşturur
- 1 yerine $1 - d/\sigma_{share}$ kullanmak, uzaktaki komşuların daha az sayıldığı anlamına gelir

Çeşitliliği Korumak için Açık Yaklaşımlar

Kalabalık(Crowding)

- Bireyleri nişler arasında eşit olarak dağıtmaya çalışır
- Yavruların ebeveynlere yakın olma eğiliminde olacağı varsayımına dayanır
- Fenotip/genotip uzayında bir uzaklık metriği kullanır
- Ebeveynleri rastgele karıştır ve eşleştir, 2 yavru üret
- Ebeveyn ve çocuk turnuvalarını, turnuva arası mesafeleri minimum olacak şekilde ayarlayın

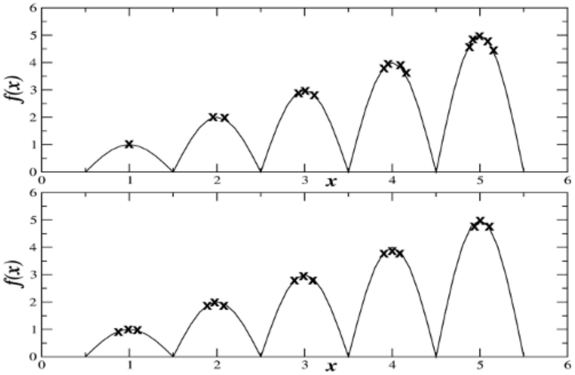
Çeşitliliği Korumak için Açık Yaklaşımlar

Kalabalık(Crowding)

- Yani, iki p'yi (ebeveynler) ve iki o'yu (yavrular), öyle ki
- $d(p_1, o_1) + d(p_2, o_2) < d(p_1, o_2) + d(p_2, o_1)$
- o_1'in p_1 ile o_2'nin p_2 ile mücadele etmesine izin verin

Çeşitliliği Korumak için Açık Yaklaşımlar

Kalabalık veya Uygunluk Paylaşımı



Şekil: Uygunluk paylaşımı(üstte) ve kalabalığa(altta) göre popülasyon dağılımı

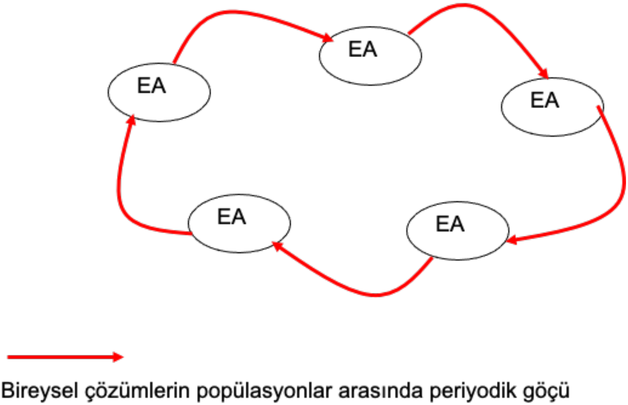
Çeşitliliği Korumaya Yönelik Örtülü Yaklaşımlar

Otomatik Türleşme(Speciation)

- Ya sadece genotipik / fenotipik olarak benzer üyelerle çiftleşme veya
- Problem temsiline bitler eklenir
  - başlangıçta rastgele ayarlanmış
  - rekombinasyon ve mutasyona tabi
  - rekombinasyon için partner seçerken, sadece iyi eşleşen üyeleri seçin

Çeşitliliği Korumaya Yönelik Örtülü Yaklaşımlar

Ada(Island) Modeli Paralel Evrimsel Algoritmalar



Çeşitliliği Korumaya Yönelik Örtülü Yaklaşımlar

Ada(Island) Modeli Paralel Evrimsel Algoritmalar

- Birden çok popülasyon paralel olarak çalıştırılır
- Bir (genellikle sabit) nesilden (bir Epoch) sonra, komşularla bireyleri değiştirilir
- Bitiş kriterleri karşılanana kadar tekrarlanır
- Paralel / kümelenmiş sistemlerden kısmen esinlenmiştir

Ada(Island) Modeli

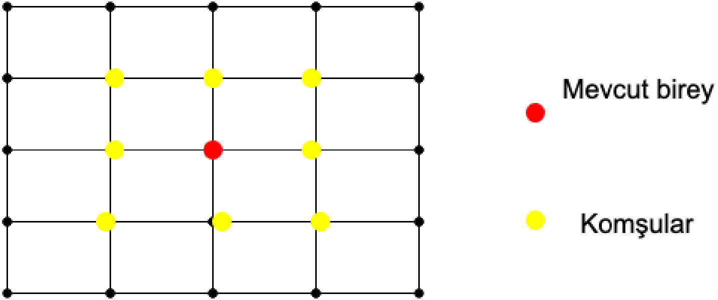
Parametreler

- Bireyler ne sıklıkla değiş tokuş edilir?
  - Çok hızlı: bütün alt popülasyonlar aynı çözüme yakınsar
  - Çok yavaş: zaman kaybı
  - Çoğu yazar $\sim$ 25-20 nesil aralığı kullanır
  - Adaptif olarak yapılabilir(25 nesil boyunca hiçbir gelişme olmadığında tüm popülasyonlarda eğitimi durdur)
- Kaç tane ve hangi bireyler değiş tokuş edilecek?
  - Genellikle $\sim$2-5, ancak popülasyo büyüklüğüne bağlıdır
  - Kopyalama veya taşıma
  - Martin vd. en iyi bireyi değil de rastgele seçilen bireyleri takas etmenin daha iyi olduğunu bulmuşlardır[1]
- Operatörler alt popülasyonlar arasında farklılık gösterebilir

Çeşitliliği Korumaya Yönelik Örtülü Yaklaşımlar

Hücresel Evrimsel Algoritmalar

- 1 popülasyonda uzamsal yapı (genellikle ızgara) uygulanır



Çeşitliliği Korumaya Yönelik Örtülü Yaklaşımlar

Hücresel Evrimsel Algoritmalar

- Her bireyin bir (genellikle dikdörtgen toroid) ızgara üzerinde bir noktada var olduğunu düşünün
- Seçim (dolayısıyla rekombinasyon) ve değiştirme, bir komşuluk(deme) kavramı kullanılarak gerçekleşir
- Izgaranın farklı bölümlerinin uzayın farklı bölümlerini aramasına yol açar, iyi çözümler birkaç nesil boyunca ızgaraya yayılır

Çeşitliliği Korumaya Yönelik Örtülü Yaklaşımlar

Hücresel Evrimsel Algoritmalar

- Her bir bireyin 8 yakın komşusu olacak şekilde dikdörtgen ızgara olduğunu varsayın
- 1 neslin eşdeğeri:
  - popülasyondaki bireyi rastgele seç
  - rulet tekerleğini kullanarak komşularından birini seç
  - 1 çocuk üretmek için çaprazla, mutasyona uğrat
  - daha uygunsa bireyi değiştirin
  - bitene kadar popülasyon içinde uygula

Kaynaklar I

James P Cohoon, Shailesh U Hegde, Worthy N Martin, and D Richards.
Punctuated equilibria: a parallel genetic algorithm.
In *Genetic algorithms and their applications: proceedings of the second International Conference on Genetic Algorithms: July 28-31, 1987 at the Massachusetts Institute of Technology, Cambridge, MA*. Hillsdale, NJ: L. Erlhaum Associates, 1987., 1987.

David E. Goldberg and Kalyanmoy Deb.
A comparative analysis of selection schemes used in genetic algorithms.
In *Foundations of Genetic Algorithms*, pages 69–93. Morgan Kaufmann, 1991.

Kaynaklar II

Darrell Whitley.
The genitor algorithm and selection pressure: Why rank-based allocation of reproductive trials is best.
In *Proceedings of the Third International Conference on Genetic Algorithms*, page 116–121, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc.