

BÖLÜM 26. GENETİK ALGORİTMALAR

26.1. Genetik Algoritmanın Tarihçesi

Michigan Üniversitesinde psikoloji ve bilgisayar bilimi uzmanı olan John Holland bu konuda ilk çalışmaları yapan kişidir. Makine öğrenmesi konusunda çalışan Holland, Darwin'in evrim kuramından etkilenecek canlılarda yaşanan genetik süreci bilgisayar ortamında gerçekleştirmeyi düşündü. Tek bir makinenin öğrenme yeteneğini geliştirmek yerine böyle yapılarda oluşan bir topluluğun çoğalma, çiftleşme, mutasyon, vb. genetik süreçlerden geçerek başarılı (öğrenebilen) yeni bireyler oluşturabildiğini gördü. Araştırmalarını, arama ve optimumu bulma için, doğal seçme ve genetik evrimden yola çıkarak yapmıştır. İşlem boyunca, biyolojik sistemde bireyin bulunduğu çevreye uyum sağlayıp daha uygun hale gelmesi örnek alınarak, optimumu bulma ve makine öğrenme problemlerinde, bilgisayar yazılımı modellenmiştir.

Çalışmalarının sonucunu açıkladığını kitabının 1975'te yayınlanmasından sonra geliştirdiği yöntemin adı Genetik Algoritmalar (GA) olarak yerleşti. Ancak 1985 yılında Holland'ın öğrencisi olarak doktorasını veren David E. Goldberg adlı inşaat mühendisi 1989 da konusunda bir klasik sayılan kitabını yayınlamaya dek genetik algoritmaların pek pratik yararı olmayan bir araştırma konusu olduğu düşünülüyordu. İlk olarak Holland' da makina öğrenme sistemlerine yardımcı olarak kullanılmış daha sonra De Jong Goldberg ve diğerleri tarafından analiz edilmiştir. Goldberg, GA' nın çok sayıda kollara ayrılmış gaz borularında, gaz akışını düzenlemek ve kontrol etmek için uygulamasını tanımlamıştır. Ayrıca kendisinin kullandığı makina öğrenmesi, nesne tanıma, görüntü işleme ve işlemsel arama gibi alanlarda kullanıldığını vurgulamıştır.

Goldberg'in gaz boru hatlarının denetimi üzerine yaptığı doktora tezi ona sadece 1985 National Science Foundation Genç Araştırmacı ödülünü kazandırmakla kalmadı, genetik algoritmaların pratik kullanımının da olabilirliğini kanıtladı. Ayrıca kitabında genetik

algoritmalara dayalı tam 83 uygulamaya yer vererek GA'nın dünyanın her yerinde çeşitli konularda kullanılmakta olduğunu gösterdi.

26.2. Genetik Algoritmanın Tanımı

Genetik algoritma, doğadaki evrim mekanizmasını örnek alan bir arama metodudur ve bir veri grubundan özel bir veriyi bulmak için kullanılır. Genetik Algoritmalar, Evrimsel Genetik ve Darwin'in doğal seçmeye (seleksiyona) benzerlik kurularak geliştirilmiş iteratif, ihtimali bir arama metodudur.

Genetik algoritmalar, gelişigüzel yürüyüş veya gelişigüzel üretilmiş çözümler arasında arama yapma mantığına dayalı bir evrimsel hesaplama yöntemidir. Bu yöntemde, problemin çözümü yapılırken bu problem ile ilgili gerekli bilgilerin hepsi kodlamada kullanılmaz. Bundan dolayı, bu yöntem zayıf arama yöntemi veya kör arama yöntemi adı da verilir.

Genetik algoritma kullanılarak problem çözümü yapmak için gelişigüzel birden fazla çözüm üretilir ve bu çözümlerin oluşturmuş olduğu kümeye populasyon denir ve populasyon dinamik bir özelliğe sahiptir ve zaman içerisinde değişiklik gösterir. Her çözüme birey veya kromozom adı verilir. Her kromozom genlerden oluşur ve bir gen, bir kromozom oluşturulurken kodlanabilen en küçük bilgiye verilen isimdir.

Genetik algoritmalar doğada geçerli olan en iyinin yaşaması kuralına dayanarak sürekli iyileşen çözümler üretir. Bunun için "iyi"nin ne olduğunu belirleyen bir uygunluk (fitness) fonksiyonudur. Diğer bir deyişle problemin özünü tasvir eden kriter, uygunluk fonksiyonu veya amaç fonksiyonudur. Uygunluk değeri bütün kromozomlar için hesaplanır ve populasyon içerisinde birden fazla kromozom olduğundan genetik algoritmaların bir diğer önemli özelliği olarak bir grup çözümle uğraşmasıdır. Bu sayede çok sayıda çözümün içinden iyileri seçilip kötülerini elenebilir.

Genetik algoritmaları diğer algoritmalarından ayıran en önemli özelliklerden biri de seçmedir. Genetik algoritmalarda çözümün

uygunluđu onun seřilme řansını artırır ancak bunu garanti etmez. Seřim de ilk grubun oluřturulması gibi rasgeledir ancak bu rasgele seřimde seřilme olasılıklarını cözümünün uygunluđu belirler.

Genetik Algoritmaları (GA) diđer yöntemlerden ayıran önemli noktalar vardır ve bu noktalar řu řekilde özetlenebilir.

- GA , sadece bir tane cözüm üzerinden iřleme bařlamaz, geliřigüzel üretilmiř olan birden fazla cözüm üzerinde arama yapar.
- GA , arama uzayında cözümünün kaliteli olup olmadığını uygunluk fonksiyonu ile kontrol eder. Bundan dolayı türev, integral veya kompleks matematiksel iřlemlere ihtiyaç duymaz.
- Bireyleri seřme ve birleřtirme ařamalarında belirli kurallar deđil olasılık kuralları kullanır.
- Diđer yöntemlerde olduđu gibi dođrudan parametreler üzerinde calışmaz. GA, en iyileme edilecek parametreleri kodlar ve bu kodlar üzerinde iřlem yapar.
- GA ne yaptıđı konusunda bilgi iřermez, nasıl yaptıđını bilir. Bu nedenle kör bir arama metodudur. Problemin tabiatı hakkında fazla bilgiye sahip deđildir.
- Olasılık kurallarına göre calıştıđından dolayı programın performansı hakkında önceden bir fikir yürütmek dođru olmaz, ancak olasılıksal bir sonuç verilebilir.
- GA, bir cözüm arama yöntemidir.

řekil 26.1’ de genetik algoritmanın řematik gösterimi görölmektedir. İlk olarak geliřigüzel olarak popülasyon oluřturulur. Popülasyon oluřturulduktan sonra deđerlendirme iřlemi bütün kromozomlara uygulanır. Diđer bir deyiřle bütün kromozomların uygunluk deđerleri hesaplanır. Uygunluklar hesaplandıktan sonra uygunluđu iyi olan bireylerin bir sonraki nesilde varlıklarını devam etmeleri için seřme iřlemi gerçekteřtirilir. Bundan sonra bazı ebeveynlere cıprazlama ve bazı kromozomlara mutasyon iřlemi uygulanır. Cıprazlama ve mutasyon iřlemlerinin hangi kromozomlara uygulanması gerektiđi uygunluk deđerinden yola çıkılarak bulunabilir. Genelde uygunluk deđerı kötü olan bireylere mutasyon iřlemi uygulanırken, uygunluk

değerleri iyi olan kromozom çiftlerine çaprazlama işlemi uygulanarak gelecek için umut verici olan evlatlar (yeni bireyler) elde edilir.

Genetik algoritmanın yapısal modeli aşağıdaki gibi olur Algoritma 26.1' de verilmiştir.

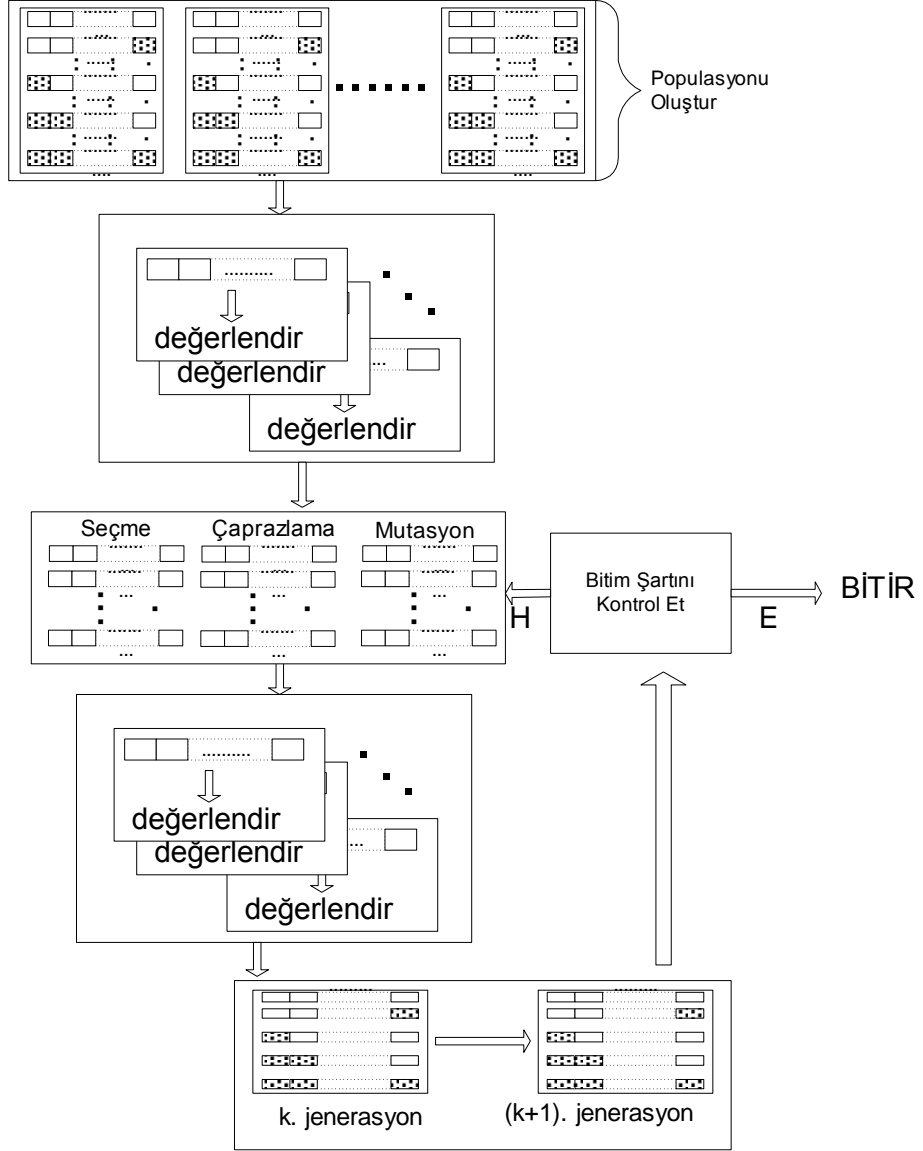
Algoritma 26.1. GA(P)

▷ P bir populasyon olup, t anındaki populasyon P(t) ile gösterilir.

1. $t \leftarrow 0$
2. $P(t) \leftarrow \text{Gelişigüzel_Populasyon_Oluştur.}$
3. $\text{Değerlendir}(P(t))$
4. Sonuç bulununcaya ve süre bitinceye kadar aşağıdaki bloğu tekrar et
 - a. $t \leftarrow t+1$
 - b. $B(t) \leftarrow \text{Seleksiyon}(P(t))$
 - c. $R(t) \leftarrow \text{Çaprazlama}(B(t))$
 - d. $M(t) \leftarrow \text{Mutasyon}(R(t))$
 - e. $\text{Değerlendir}(M(t))$
 - f. $P(t) \leftarrow \text{Yaşayanlar}(M(t), P(t-1))$

26.3. Kodlama yöntemleri

Kodlama planı, genetik algoritmanın önemli bir kısmını teşkil eder. Çünkü bu plan bilginin çerçevesini şiddetle sınırlayabilir. Öyle ki probleme özgü bilginin bir kromozomsal gösterimiyle temsili sağlanır. Kromozom genellikle, problemdeki değişkenlerin belli bir düzende sıralanmasıdır. Kromozomu oluşturmak için sıralanmış her bir değişkene “gen” adı verilir. Buna göre bir gen kendi başına anlamlı genetik bilgiyi taşıyan en küçük genetik yapıdır. Mesela; 101 bit dizisi bir noktanın x-koordinatının ikilik düzende kodlandığı gen olabilir.



Şekil 26.1. Genetik algoritmanın şematik gösterimi.

Aynı şekilde bir kromozom ise bir ya da daha fazla genin bir araya gelmesiyle oluşan ve problemin çözümü için gerekli tüm bilgiyi üzerinde taşıyan genetik yapı olarak tanımlanabilir. Örnek vermek

gerekirse; 100011101111 kromozomu x_1 , y_1 , x_2 , y_2 koordinatlarından oluşan iki noktanın konumu hakkında bilgi verecektir. Buna ikili kodlama ismi verilir, çünkü her gen bir bitten oluşmaktadır. Kodlama türleri:

- İkili Kodlama (Binary Encoding)
- Tamsayı Kodlama (Integer Encoding)
- Karakter Kodlama (Character Encoding)
- Gerçel Sayı Kodlama veya Değer Kodlama (Floating-Point Encoding).

şeklinde dört gruba ayrılırlar. Bir gen, ikili kodlama bir biti, tamsayı kodlamada bir tamsayıyı, karakter kodlamada bir karakteri ve gerçel sayı kodlamada bir gerçel sayıyı temsil eder.

Bu parametreleri kodlarken dikkat edilmesi gereken en önemli noktalardan biri ise kodlamanın nasıl yapıldığıdır. Örnek olarak, kimi zaman bir parametrenin doğrusal ya da logaritmik kodlanması GA performansında önemli farka yol açar. Kodlamanın diğer önemli bir hususu ise kodlama gösteriminin nasıl yapıldığıdır. Bu da yeterince açık olmamakla birlikte GA performansını etkileyen bir noktadır.

26.4. Uygunluk Değerlendirme Teknikleri

Başlangıç topluluğu bir kez oluşturulduktan sonra evrim başlar. Genetik algoritma bireylerin uygunluk ve iyiliklerine göre ayrılıp fark edilmesine gerek duyar. Uygunluk, topluluktaki bir kısım bireyin problemi nasıl çözeceği için iyi bir ölçüdür. O problem parametrelerini kodlamayla ölçülür ve uygunluk fonksiyonuna giriş olarak kullanılır. Yüksek ihtimalle uygun olan bu üyeler tekrar üreme, çaprazlama ve mutasyon operatörleriyle seçilirler.

Bazı problemler için bireyin uygunluğu, bireyden elde edilen sonuç ile tahmin edilen sonuç arasındaki hatadan bulunabilir. Daha iyi bireylerde, bu hata sifıra yakın olur. Bu hata genellikle, girişin tekrar sunulacak kombinasyonlarının ortalaması veya toplamıyla hesaplanır

(değerler değişkenlerden bağımsızdır). Beklenen ve üretilen değer arasındaki korelasyon etkeni, uygunluk değerini hesaplamak için kullanılabilir.

Amaç fonksiyonu (değerlendirme fonksiyonu, uygunluk fonksiyonu) her bir kromozomun durumunu değerlendirmek için mekanizmayı sağlayan ana bir kaynaktır. Bu GA ve sistem arasında önemli bir bağlantıdır. Fonksiyon giriş olarak kodu çözülmüş şekilde kromozom alır ve kromozomun performansına bir ölçü unsuru olarak bir amaç değer üretir. Bu işlem bütün bireyler için gerçekleştirilir ve hesaplama sonucunda bireylerin ayırt edilebilmesini sağlayan ayrıık değerler elde edilir ve bu değerler bireyin maliyetini belirler. Populasyondaki diğer bireyler de hesaba katıldığı zaman, her bireyin uygunluk değeri kendi maliyetinden elde edilir.

GA' nın maksimizasyon tabiatı ve minimizasyon problemleri arasında yüzleşmeden daha önemli olan nokta maliyet ile uygunluk değeri arasındaki ayırımın iyi yapılması gerekiyor. Aslında, maliyet probleme has olan bir durum olduğundan, belki de gelecekte bu değer hiç değişmeyecektir. Bundan başka maliyet genellikle sıralı bir kümedeki değerleri düşünür. Diğer yönden maliyet-uygunluk eşleşmesinin yapılması seçme süreci ile ilişkilidir. Bu, negatif olmayan gerçel sayılar kümesinin bir alt kümesine olan eşleştirmedir.

Normalleştirilmiş uygunluk değeri, bir sonraki nesilde uygunluk değerinin ait olduğu bireyin kaç tane çocuğunun olacağını gösterir. Normalleştirilmiş uygunluk değeri $[0,1]$ aralığında değer alır ve bundan dolayı bazı durumlarda olasılık değeri olarak ele alınabilmektedir. Bununla birlikte, bu durum bazı seçme yöntemleri için doğru olabilir.

Diğer bir durumda izafi uygunluk değeridir ve her birey için bu değerin elde edilebilmesine yönelik olarak bütün uygunluk değerleri toplandıktan sonra ortalaması alınır ve her bireyin uygunluğu bu ortalama değere bölünerek ait olduğu bireyin uygunluk değeri elde edilir. İzafi uygunluk değeri halihazırdaki populasyonun bir önceki nesile göre iyiye mi gidiyor, yoksa kötüye mi gidiyor bunun anlaşılmasında yardımcı olur.

Populasyon içindeki bireyler sıralanırken, maliyet-uygunluk eşleştirmesi ölçeği korunabilir olduğu gibi korunmayabilir de. Belli başlı iki tane uygunluk değeri atama yöntemi vardır.

Ölçekleme Uygunluk

Uygunluk, maliyetin bir fonksiyonu olarak hesaplanır ve populasyondaki en iyi kromozoma diğer kromozomların kontrolü verilirken, uygunluk değerlerinin negatif olmadığından emin olunması gerekiyor.

Rank Uygunluk

Populasyondaki bireylerin maliyetlerinin sıralanması ile eşleştirme elde edilir ve her bireye kendi pozisyonu ile ilişkili bir uygunluk değeri verilir.

Ölçekleme, daha geleneksel bir yöntemdir. Ham uygunluk değeri, maliyetin monotonik fonksiyonu olarak hesaplanır ve daha sonra lineer olarak ölçeklendirilirler. İlk zorluk bu aşamada ortaya çıkar; ölçekleme, farklı bireyler arasındaki performansı korumayı amaçlarken, başlangıç dönüşümü ve devamındaki dengeleme, sonuçta bireylere atanan ham uygunluk değerlerini önemli derecede etkilerler.

Ölçeklendirme ile diğer bireylerden daha güçlü olan bireye diğer uygunluk değerlerine göre çok büyük değer verilebilir ve seçmede hızlı bir şekilde populasyonda bu bireyin baskısı hissedilir. Tersine olarak, eğer bireylerin çoğu birbirlerine benzer olarak daha çok veya daha az iş yapıyorlarsa, diğer bireylere göre en iyi bireyin avantajı minimumdur ve bunun sonucunda arama işlemi amaçsız bir yola girebilir.

Yukarıdaki iki durumda arzu edilmediği zamanlarda en iyi kromozomun ham uygunluk değerinin başka bir yolla kontrol edilmesi gerekir. Örneğin, Sigma-ölçekleme ortalamadan yola çıkarak arzu edilen ofset miktarını hesaplar ve her nesildeki ham uygunluk değerleri arasındaki standart sapmaları da hesaplar, böylece en iyi bireyin uygunluk değeri daha anlamlı bir şekilde belirlenmiş olur.

Rank, problem formülasyonunda elimine ettiği bazı hassaslıklardan dolayı aynı zorluk ile karşılaşır. Çünkü en iyi bireye her zaman aynı

ham uygunluk değeri atanır. Benzer bir şekilde bütün bireyler birbirlerine çok yakın oldukları zaman (birbirlerine çok benzemeye başladıkları zaman) hala en iyi birey kalan için açık bir şekilde tercih edilir.

Rank-tabanlı uygunluk atanması, genellikle lineer veya üstel olarak seçilen rank-uygunluk eşleştirmesi ile karakterize edilir. N tane kromozomu olan bir populasyon için en iyi bireyin rank işlemi yapılacak olursa, bireyler p_0, p_1, \dots, p_{N-1} için rank τ ve izafi uygunluk $f=f(\tau)$ olmak üzere eşleştirme lineer için

$$f(\tau)=s-(s-1)\frac{2\tau}{N-1} \quad (26.1)$$

şeklinde olur ve $1 < s \leq 2$ ve bu en iyi birey için istenen izafi uygunluktur.

$$\sum_{i=0}^{N-1} f(i) = N \quad (26.2)$$

şartı sağlanırken, bütün bireylerin uygunluk değerlerinin negatif olmaması için s parametresine bir üst sınır gerekiyor.

Üstel seçildiği zaman

$$f(\tau)=p^\tau.s \quad (26.3)$$

ve $s > 1$ ve en iyi birey için arzu edilen uygunluk değeri s' tir ve p aşağıdaki şartı sağlar.

$$\sum_{i=0}^{N-1} p^i = \frac{N}{s} \quad (26.4)$$

s için bir üst sınır olmadığından üstel eşleştirme, lineer eşleştirmeden daha esneklerdir.

$1 < s \leq 2$ için rank-tabanlı uygunluk atanmasında lineer ve üstel arasındaki en önemli fark üstel eşleştirme kötü bireyleri fazla cezalandırmaz. Sonuç olarak üstel eşleştirme genellikle daha geniş bir alanda arama işlemini devam ettirir.

Uygunluk fonksiyonu inşa edilirken bazı problemler ortaya çıkabilir ve bu problemler kısaca aşağıda vurgulanmıştır:

- Uygunluk fonksiyonu, problemin minimizasyon veya maksimizasyon olmasına bağlıdır.
- Ortam uygunluk fonksiyonun hesaplanmasını etkileyebilir (örneğin kısmen hesaplama).
- GA' nın icra edilmesi sırasında uygunluk fonksiyonu dinamik olarak değişebilir.
- Uygunluk fonksiyonu çok karmaşık olduğu durumlarda sadece yaklaşım ile değeri hesaplanabilir.
- Uygunluk fonksiyonu seçme işleminin sağlıklı yapılabilmesi için bireylere oldukça farklı değerler vermelidir.
- Uygunluk fonksiyonu, problemin sınırlayıcılarını hesaba katmalıdır. Örneğin, eğer mümkün olmayan çözüm geldiğinde uygunluk fonksiyonu o bireye kötü anlamına gelecek bir değer vermelidir.

26.5. Genetik Operatörler

Genetik algoritmanın işleme sürecinde populasyonun çeşitliliği koruması veya daha iyi çözümleri yakalayabilmesi için büyük sıçrama gerektiği durumlarda kullanılan iki tane genetik operatör kullanılırken bir sonraki nesilde varlıklarını idame edecek bireylerin belirlenmesi için bir tane genetik operatör kullanılır. Bunlar:

- Seçme
- Çaprazlama
- Mutasyon

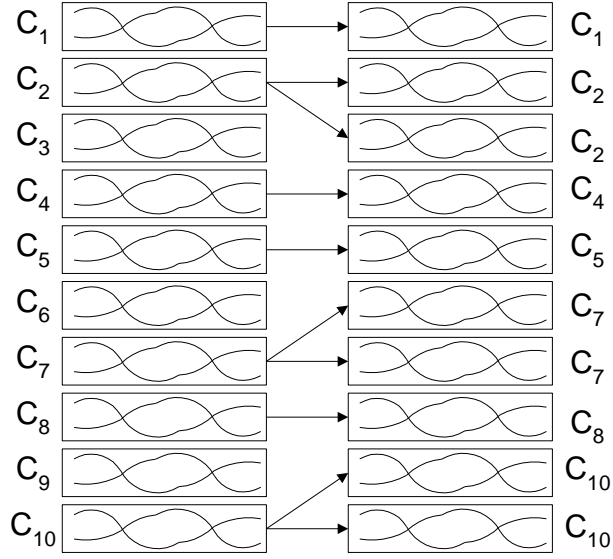
şeklinde. Bu operatörlerden seçme operatörü kromozom tabanlıyken çaprazlama ve mutasyon gen tabanlı operatörlerdir. Seçme işlemi, bir nesilden bir sonraki nesile geçilirken kromozomların uygunluk değerlerine bakılarak hangilerin eleneceğine ve hangilerin bir sonraki nesilde varlıklarını sürdüreceğine karar verme mekanizmasına seçme adı verilir. Hatta uygunluk değeri iyi olan bireylerden elenen bireylerin yerlerine birer kopyaları alınır veya iyi olan bireyler çaprazlandıktan sonra elenen bireyler yerine yeni bireyler konulur.

2.5.1. Seçme

t anındaki populasyon P_t olmak üzere seçme (seleksiyon) operatörü (S) aşağıdaki gibi bir bağıntıdır.

$$S: P_t \rightarrow P_t$$

Seçme operatörü, halihazır populyasyondan bazı kriterlere göre kromozomların bir kısmı seçip bir sonraki jenerasyona aktarırken bir kısmını eleyen bir operatördür. Şekil-2.2' de basit bir seçme işlemi görülmektedir. Bu şekilde C_1 , C_2 , C_4 , C_5 , C_7 , C_8 ve C_{10} kromozomları bir sonraki nesilde varlıklarını sürdürürken (seçilen bireyler olurken), C_3 , C_6 ve C_9 kromozomları bir sonraki jenerasyona geçebilecek kadar iyi olmadıklarından dolayı elenen kromozomlardır. Bu kromozomların yerlerine sırası ile C_2 , C_7 ve C_{10} kromozomlarının birer kopyası alınmıştır. Genetik algoritmanın süreci bu şekilde devam ederse, belli adım sonunda bütün kromozomlar aynı olacaktır. Bu durum çoğu zaman prematüre çözümler vereceğinden dolayı pek istenmeyen bir durumdur. Başka bir deyişle, seçme operatörü temelde iki farklı şekilde kullanılır. Bunlardan birincisi jenerasyonsal seçme ve bu durumda populyasyonun hepsi yeni bir populyasyon ile yer değiştirir. İkinci bir kullanım şekli de kararlı-durum seçmesidir ve bu durumda her jenerasyondaki çok az bireyin yerine yeni birey alınır.



Şekil 26.2. C_i , Kromozom $_i$, $1 \leq i \leq 10$, göstermek üzere basit bir seçim işlemi.

Seçme operatörünün prematüre çözümler vermesini engellemek veya iyi çözüm elde etmek için bu operatörün değişik uygulanma şekilleri vardır. Bunlar

- Oransal Seçim
- Rulet Tekerleği Seçimi
- Stokastik Evrensel Örnekleme
- Rank Seçimi
- Kararlı Durum Seçimi
- Turnuva Seçimi
- Genel Elitizm
- Yerel Elitizm
- Kalabalıklaştırma
- Deterministik Kalabalıklaştırma
- Paylaşım

Seçme işlemi, bireylerin uygunlukları ile orantılı olarak bir bireyin bir sonraki nesil için bir veya birden fazla birey üretmesi işlemidir. Bununla birlikte seçilecek birey sayısı tamsayı olduğundan genellikle

üretilebilecek birey sayısı ya istenenden çoktur ya da azdır. Bu durumda seçme hatası denilen bir hata sistematik olarak her nesilde vuku bulacaktır.

Seçme işleminin hatasından kurtulmak için farklı seçme yöntemleri kullanılmıştır ve bunlardan bir kısmı bundan sonraki bölümlerde kısaca açıklanacaklardır.

Oransal seçim

Oransal seçim yönteminde populasyonun ortalama uygunluk değeri bulunur ve bu değer \tilde{F} ile gösterilsin. $\forall p \in P$ için $F(p)$ değeri p kromozomunun uygunluk değerini göstermek üzere p kromozomundan seçilecek kopya sayısı

$$\left\| \frac{F(p)}{\tilde{F}} \right\| \quad (26.5)$$

kadar olur. Bu değer tamsayıya yuvarlandıktan sonra kaç çıkıyorsa, ilgili kromozomdan o kadar kopya seçilir. Bu yöntemin amacı, uygunluk değeri yüksek olan bireylerin bir sonraki nesil populasyonu için daha önemli olduğunu ve bundan dolayı bu bireylerin bir sonraki nesilde daha baskın (daha çok) olmaları gerekiyor ve bu sebeple bu bireyleri tespit edip seçmektir. Eğer bu kesirli değer küçük ise, bir sonraki nesilde bu bireyin var olması populasyonun toplam kalitesini fazlaca yükseltmeyeceğinden dolayı bu bireyin seçilme sayısı mümkün olan en küçük sayıya indirgenmeye çalışılmaktadır ve hatta mümkünse, bu bireyden hiç seçmemek eğilimindedir.

Rulet Tekerleği Seçimi

Rulet tekerleği seçimi, bir çok araştırmacı tarafından veya programcı tarafından tercih edilen bir seçim yöntemidir. N tane kromozom içeren bir populasyon verilsin; her bireyin izafi uygunluk değeri hesaplanır ve bu değer seçme olasılığında kullanılır. X , populasyon havuzu üzerinde tanımlanmış bir gelişigüzel değişken olsun ve $f_i \in \mathbb{R}^+$ değeri havuzdaki x_i kromozomun uygunluk değerini belirtsin. Maksimizasyon problemleri için

$$P(X = x_i) = \frac{f_i}{\sum_{i=1}^N f_i} \quad (26.6)$$

değeri x_i kromozomun seçilme olasılığıdır. Minimizasyon problemleri için halihazırdaki en büyük uygunluk değerine göre izafî standart sapma uygunluk değeri olarak kullanılır. O anki populasyon için f_{\max} maksimum uygunluk değeri olsun.

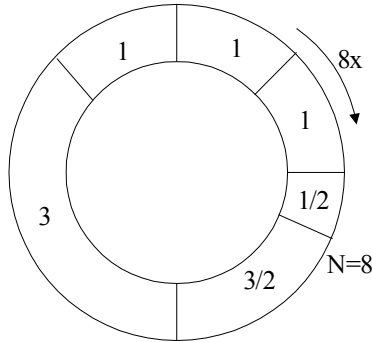
$$f_i' = f_{\max} - f_i \quad (26.7)$$

için x_i bireyini seçme olasılığı

$$P(X = x_i) = \frac{f_i'}{\sum_{i=1}^N f_i'} \quad (26.8)$$

şeklinde olur.

Şekil 2.3' te 8 kromozom için rulet tekerleği görülmektedir ve bu tekerlek üzerinde her bireyin seçilebilme olasılığı görülmektedir. Dikkat edilirse, herhangi bir anda sadece bir tane kromozom seçilir, çünkü bir tane başlık (yönlü doğru parçası) vardır.



Şekil 26.3. Rulet tekerleği seçimi.

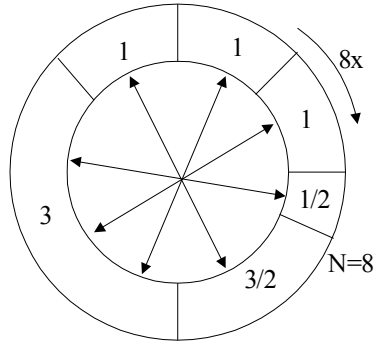
Stokastik Evrensel Örneklem

Seçme işleminin sistematik hatasından kurtulmak için stokastik olarak seçme işlemi uygulanır. Stokastik seçme hatası mümkün oldukça küçük olarak tutulurken, bir bireyden uygunluğu ile orantılı olarak beklenen bireyler üretilir. Bu iki amacı ilk olarak Baker (1987) birleştirip Stokastik Evrensel Örneklem yöntemi sunmuştur.

Bu yöntemde Rulet tekerleği yönteminde olduğu gibi tekerlek çevrilir ve durduğu zaman dikkat edilirse (Şekil 2.4) birden fazla yönlü doğru parçası vardır. Her doğru parçasının işaret etmiş olduğu alandaki sayının ya taban fonksiyonu ya da tavan fonksiyonu değeri alınır ve bu değer o bireyden kaç tane bireyin üretileceğini gösterir.

Rank Seçimi

Rulet tekerleği seçiminde eğer uygunluk değerlerinin kapsamış olduğu aralık çok büyürse, bu durumda bazı kromozomların uygunluk değeri çok küçük kalabilir ve bunun sonucunda bu kromozomların seçilme olasılığı sıfıra yakın bir değer olabilir. Bu problemi gidermek için eğer popülasyondaki kromozom sayısı P ise uygunluğu en iyi olan kromozoma P değeri, ondan sonraki kromozoma $P-1$ değeri ve bu şekilde devam ederek en kötü kromozoma 1 değeri verilir. Bunun anlamı en kötü kromozomun seçilme olasılığı sıfıra yakın bir değerden



Şekil 26.4. Stokastik evrensel örneklem yöntemi ile seçim yapmak.

$$\frac{2}{P(P+1)}$$

(26.9)

olur. Bu işlem yapıldıktan sonra rulet tekerleği işlemi bu yeni uygunluk değerlerine göre yapılır.

Kararlı Durum Seçimi

Bu seçim yönteminin temel amacı, iyi kromozomların bir sonraki nesilde varlıklarını sürdürmelerini sağlamaktır. Böylece için bu algoritmanın uygulamasını gerçekleştiren kişinin belirlediği bir oranda iyi kromozomlar seçilir ve kötü kromozomların yerine iyi kromozomlardan çaprazlama ile veya mutasyon ile yeni bireyler üretilerek bir sonraki nesil oluşturulur.

Turnuva Seçimi

Genetik algoritmalarda kullanılan seçim mekanizmalarından biri de Turnuva seçimidir ve bu yöntemde bireyler bir eşleştirme havuzuna atılırlar. Eşleştirme havuzundaki bireylerden evlatlar üretilir ve bu evlatlar bir sonraki neslin temelini teşkil ederler. Seçme mekanizmasının temel amacı iyi bireylerin seçilmesi yönünde baskı uygulamasıdır ve bunun sonucunda daha iyi bireylerin seçilmesi sağlanır. Seçme baskısı, iyi bireylerin seçilmesi yönünde hissedilen baskıdır ve bu baskı arttıkça iyi bireylerin seçilme şansı artar. Seçme baskısı çözüme yakınsama oranını arttırır. Yüksek seçme baskısı beraberinde yüksek yakınsama oranını getirirken, düşük seçme baskısı yanında düşük yakınsama oranını getirir ve bunun sonucunda daha fazla iterasyon ile belki çözüme gidilebilir. Çok yüksek seçme baskısı da istenmeyen bir durumdur ve prematüre çözümlere sebep olur.

Bu seçme yönteminde uygunluk değeri en yüksek olan birey eşleştirme havuzuna eklenir ve sonra geriye kalan bireyler için aynı işlem gerçekleştirilir. Bu işleme s tane birey seçilip eşleştirme havuzuna eklenene kadar devam edilir, çünkü turnuvanın boyutu s olduğu kabul edilmiştir. Bu şekilde havuzdaki popülasyonun ortalama uygunluk değeri daha yüksek olacaktır ve bunun sonucunda seçme baskısı ortaya çıkacaktır.

Global Elitizm ve Yerel Elitizm

Bu yöntem seçme algoritması olmaktan çok en iyi çözümü (kromozomu) korumaya yönelik bir yöntemdir. İkiye ayrılmasının

sebebi, global olarak bir bireyin korunmasını sağlamak veya yerel olarak bireylerin korunmasını sağlamak amacıyla.

Kalabalıklaştırma

Nesiller arasındaki dağılım farkını minimuma indirerek prematüre çözümlerin oluşmasını engelleme yöntemine Kalabalıklaştırma (Crowding) denir. Bu yöntemde popülasyonun belli bir orana göre bir kısmı seçilir ve bir sonraki nesil oluşturulur. Bu yöntemin temel prensibi aşağıdaki gibi özetlenebilir:

Var olan popülasyondan evlatlar üretilir ve sonuç evlatları popülasyondaki bireylerin yerlerini alırken, her evlat için popülasyonda gelişigüzel olarak belli sayıda birey seçilir ve en çok benzeyenin yerine evlat konulur. Benzerlik metriği tanım kümesinden bağımsız olabileceği gibi tanım kümesine bağımlı da olabilir. Yerine yeni birey seçilecek olan, en çok benzeyen birey seçilerek bir nesilden bir sonraki nesile geçilirken popülasyon dağılımları arasındaki farkın minimum olmasına çalışılır. Bu durum genetik algoritmanın tek çözüme meyilli olarak davranmasını engeller.

Deterministik Kalabalıklaştırma

Kalabalıklaştırma yöntemini geliştirmek için evlatlar ataları ile karşılaştırılırlar ve eğer evladiın uygunluk değeri atanın uygunluk değeriinden daha iyi ise, ata yerine evlat popülasyonda bir birey olarak yer alır ve atanın popülasyondaki ömrü son bulur ve buna deterministik kalabalıklaştırma yöntemi adı verilmiştir. Bu durum genetik algoritmada seçme operatörünün baskısının popülasyonda hissedilmesini sağlar.

Paylaşım

Genetik algoritmada popülasyonun dağınıklık özelliğini kaybetmesi istenmez. Bunun için seçim operatörünün değişik tanımlamaları ortaya çıkmıştır ve bunlardan biri de Paylaşım yöntemidir.

Bu yöntemin temel amacı, popülasyondaki benzer bireylerin uygunluk değerlerinin azalması ve çözüm için kaliteli aday olma özelliklerinin yavaş-yavaş kaybolmasıdır. Bu yöntemde benzeri olmayan bireyler bir sonraki nesile taşınmaları için cesaretlendirilirken, benzeri olan bireylerin bir sonraki nesilde var olmaları ile ilgili cesaretlerinin

kırılması durumudur. Bu populasyon dağınıklık baskısına sebep olur; bu şekilde bireylerin geçici olarak yerel çözümlerde beklemeleri sağlanmaya çalışılır.

i bir birey olmak üzere bu bireyin populasyonun geriye kalan kısmı ile örtüşen gen sayısı m_i olsun. i bireyinin ham uygunluk değeri f_i gösterilsin. i bireyinin paylaşılmış uygunluk değeri

$$f_{sh,i} = \frac{f_i}{m_i} \quad (26.10)$$

olur. Ortak olan gen sayısını bulmak için bütün bireyler üzerinde paylaşma fonksiyonunun toplamı yapılır ve

$$m_i = \sum_{j=1}^N sh(d_{i,j}) \quad (26.11)$$

toplamı bize bu değeri verir. $d_{i,j}$ değeri i ve j bireylerinin birbirine olan uzaklığıdır. Bir noktada, benzerliğin tersi olarak da ele alınabilir.

Eğer paylaşma fonksiyonu uzaklığın belli bir yarıçap σ_{sh} içerisinde kaldığını tespit ederse, paylaşma fonksiyonu 0 ile 1 arasında bir değer üretir ve bu değer daha büyük benzerliklerde artar, yani $d_{i,j}$ azalırken, fonksiyon değeri artar. Diğer durumda 0 değerini verir. Aslında populasyonun her bireyi bir bölgenin merkezi olarak kabul edilir ve bulunduğu bölgenin yarıçapı σ_{sh} için populasyonun diğer bireylerinden uzaklığı σ_{sh} küçük olan her birey için o bireyin paylaşılmış uygunluk değeri azalır ve aşağıdaki gibi verilebilir.

$$sh(d_{i,j}) = \begin{cases} 1 - \left(\frac{d_{i,j}}{\sigma_{i,j}} \right)^\alpha & d_{i,j} < \sigma_{sh} \text{ ise} \\ 0 & \text{aksi halde} \end{cases} \quad (26.12)$$

2.5.2. Çaprazlama

Çaprazlama operatörü \otimes simgesi ile gösterilmek üzere ve P popülasyonu için $\exists p_1, p_2, p_3, p_4 \in P$ için

$$\otimes: P \times P \rightarrow P \times P \text{ veya } \otimes: P \times P \rightarrow P$$

şeklinde olan bir operatördür ve

$$p_1 \otimes p_2 = \{p_3, p_4\} \text{ veya } p_1 \otimes p_2 = \{p_3\}$$

olur.

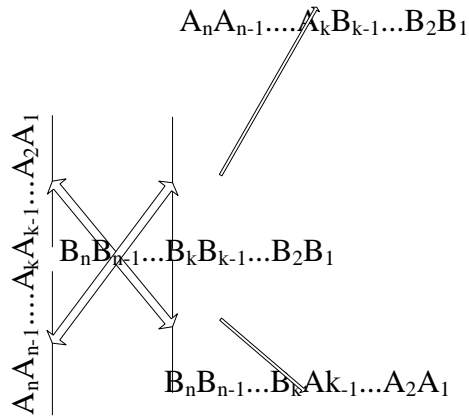
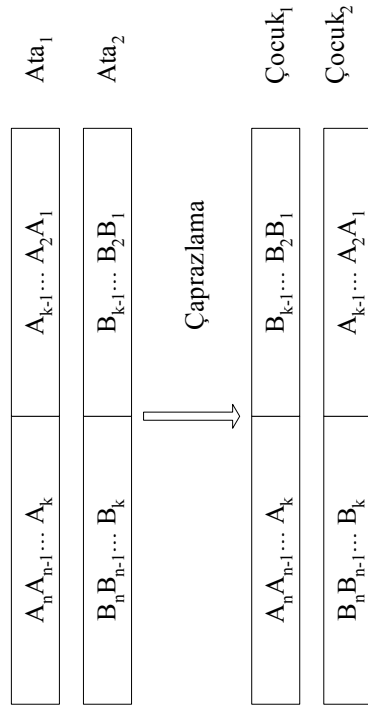
Bu tez çalışmasında kullanılan çaprazlama çeşitleri:

- Tek-noktalı çaprazlama
- Çift (iki)-noktalı çaprazlama
- Düzenli çaprazlama
- Yoğun çaprazlama

Tek-Noktalı Çaprazlama: Bu çaprazlama yönteminde kromozomun uzunluğu içinde gelişigüzel bir sayı üretilir ve bu noktadan seçilen iki ata kromozom için bölme noktası olur. Bir kromozomun en önemli parçası alınırken diğer kromozomun en önemsiz parçası alınarak bir tane çocuk oluşturulur.

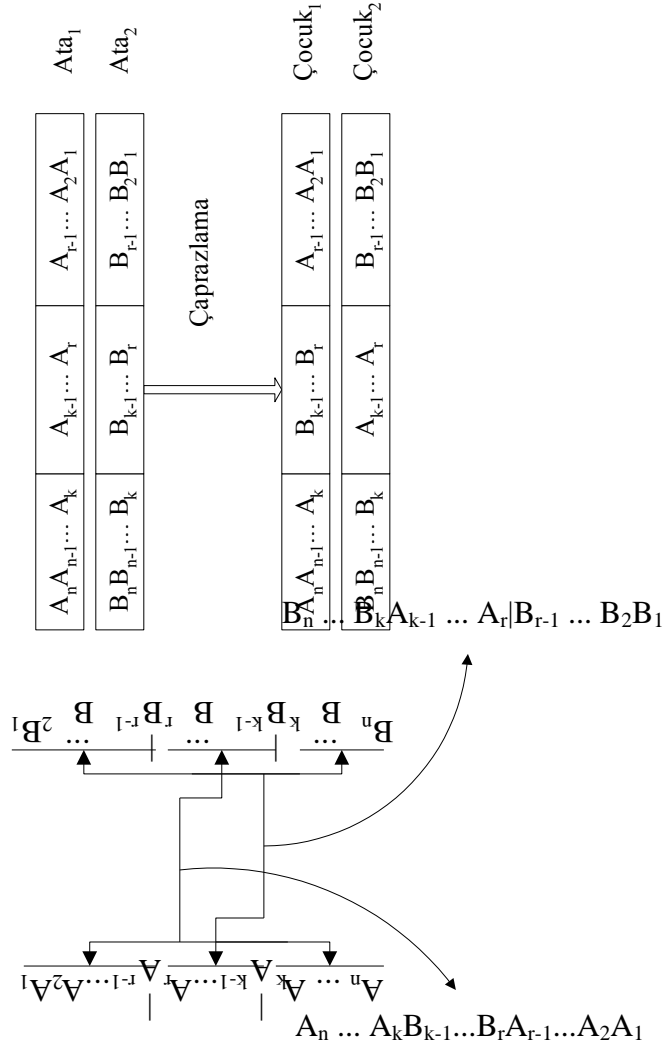
Şekil 26.5' de $A_n A_{n-1} \dots A_k A_{k-1} \dots A_2 A_1$ ve $B_n B_{n-1} \dots B_k B_{k-1} \dots B_2 B_1$ seçilen atalar görülmektedir. Gelişigüzel bir çaprazlama noktası seçildikten sonra atalar $A_n A_{n-1} \dots A_k | A_{k-1} \dots A_2 A_1$ ve $B_n B_{n-1} \dots B_k | B_{k-1} \dots B_2 B_1$ şeklinde olsunlar. Bu iki atadan elde edilen çocuklar $A_n A_{n-1} \dots A_k B_{k-1} \dots B_2 B_1$ ve $B_n B_{n-1} \dots B_k A_{k-1} \dots A_2 A_1$ şeklinde olurlar ve bu durum Şekil 26.5' te görülmektedir.

İki-noktalı çaprazlamada gelişigüzel olarak kromozomun uzunluğu içinde kalacak şekilde iki tane nokta belirlenir. Bir kromozomda iki nokta arasında kalan kısım alınırken diğer kromozomda iki nokta dışında kalan kısımlar alınır ve bu şekilde bir tane çocuk oluşturulur. Diğer çocuğun oluşturulması için ataların rolleri değiştirilir.



Şekil 26.5. Tek-noktalı çaprazlama işlemi.

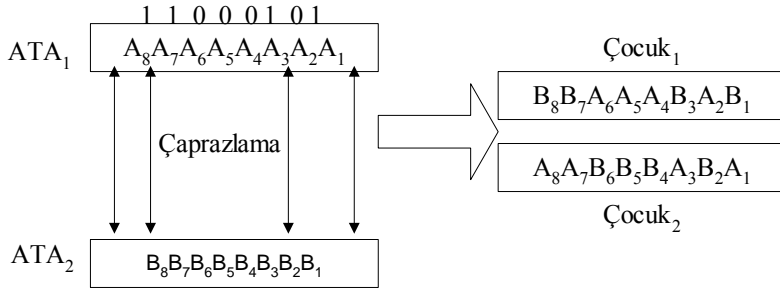
Şekil 26.6' da iki-noktalı çaprazlama işlemi şekil ile gösterilmiştir. Ata kromozomlar $A_n \dots A_k A_{k-1} \dots A_r A_{r-1} \dots A_2 A_1$ ve $B_n \dots B_k B_{k-1} \dots B_r B_{r-1} \dots B_2 B_1$ olmak üzere gelişigüzel oluşturulan iki nokta belirlendikten sonra atalar $A_n \dots A_k | A_{k-1} \dots A_r | A_{r-1} \dots A_2 A_1$ ve $B_n \dots B_k | B_{k-1} \dots B_r | B_{r-1} \dots B_2 B_1$ şeklinde olur. Bunun sonucunda çaprazlama işlemi yapıldığında elde edilen çocuklar $A_n \dots A_k B_{k-1} \dots B_r A_{r-1} \dots A_2 A_1$ ve $B_n \dots B_k A_{k-1} \dots A_r | B_{r-1} \dots B_2 B_1$ şeklinde olurlar.



Şekil 26.6. İki-noktalı çaprazlama işlemi.

Düzenli çaprazlama yönteminde her gen için gelişigüzel bir sayı üretilir ve bu sayı 0 veya 1 değeri alabilmektedir. Eğer bu sayı 1 değeri almışsa, iki ata arasında bu gen yer değiştirecektir. Eğer bu sayı 0 ise, her iki atanın bu geni sabit kalacak demektir.

Örneğin ata kromozomlar $A_8A_7A_6A_5A_4A_3A_2A_1$ ve $B_8B_7B_6B_5B_4B_3B_2B_1$ olmak üzere her gen için gelişigüzel üretilen sayılar dizisi 11000101 olsun. Bu durumda yapılan çaprazlama sonucunda elde edilen çocuklar $B_8B_7A_6A_5A_4B_3A_2B_1$ ve $A_8A_7B_6B_5B_4A_3B_2A_1$ olurlar. Şekil 26.7’ de düzenli çaprazlama işleminin grafiksel gösterimi görülmektedir.



Şekil 2.7. Düzenli çaprazlama işlemi.

26.5.3. Mutasyon

GA’ da işlemler devam ettikçe kromozomlar birbirine benzemeye başlarlar ve bunun sonucunda algoritma bitime gider. Eğer elde edilen çözüm optimal veya optimal çözüme yakın bir çözüm ise problem yoktur. Fakat, elde edilen çözüm optimal çözümünden çok uzak ise bu çözüme prematüre çözüm denir ve kromozomlar içerisinde bazılarının büyük sıçramalara tabi tutulmaları gerekmektedir. Bunu yapmanın yollarından biri çaprazlama idi ve diğeri de mutasyon işlemidir.

P popülasyonu için mutasyon operatörü M aşağıdaki gibi tanımlanır.

$M:P \rightarrow P$

ve bağıntı bir fonksiyondur. Bir fonksiyonun tersinin var olması için o fonksiyonun bire-bir ve örten olması gerekir. Mutasyon işlemi için verilen fonksiyon bire-bir veya örten olmak zorunda olmadığından dolayı mutasyon işlemi her zaman tersinir bir işlem değildir. $\exists p, q \in P$ için

$M[p]=q$

olur. Bu işlemin anlamı p kromozomuna mutasyon işlemi uygulandıktan sonra q kromozomu elde edilmektedir.

Bir gene mutasyon işlemi uygulamak ya o genin tümleyeni alınır veya aynı kromozom içerisinde iki genin yerleri birbiri ile değiştirilir. En çok kullanılan mutasyon yöntemi, genin tümleyenin alınmasıdır.

26.6. Genetik Algoritma Çeşitleri

Bu tezde önereceğim genetik algoritma çeşidi dışında en çok kullanılan yedi tane genetik algoritma çeşidi vardır. Bunlar:

- Mikro-Genetik Algoritmalar
- Basit Genetik Algoritmalar
- Kararlı Durum GA (Genitor)
- CHC
- Hibrid Genetik Algoritmalar
- Paralel Genetik Algoritmalar

26.6.1. Mikro-Genetik Algoritmalar

Mikro-Genetik algoritmalarda, eğer kullanılan popülasyonun boyutu 10'dan daha az ise, bu algoritmaya Mikro-Genetik Algoritma denir. Diğer genetik algoritmalarda kullanılan kodlama yöntemleri,

değerlendirme yöntemleri ve genetik operatörler aynıdır. Tek fark, kromozom sayısının az olmasıdır.

Kromozom sayısının az tutulmasının sebebi, GA iteratif bir yöntem olduğundan dolayı, algoritmanın icra edilmesi zamanı, populasyon boyutunun lineer veya kuadratik veya kübik bir fonksiyonudur. Bundan dolayı, populasyon boyutu küçültüldüğünde işlem zamanı doğal olarak kısalmaktadır.

26.6.2. Basit Genetik Algoritmalar

Bu genetik algoritmaların genel özellikleri şu şekilde sıralanabilir.

- Çaprazlama şekli tek-noktalı veya çift-noktalı.
- Seçim şekli rulet tekerleği veya turnuva.
- Mutasyon işlemi ile beraber elitizm kullanılır.

26.6.3. Durağan Durum Genetik Algoritmalar (Genitor)

Genitor veya Durağan Durum Genetik Algoritmanın genel özellikleri şu şekilde sıralanabilir.

- Rank tabanlı seçim.
- İki noktalı çaprazlama.
- Üretilen iki çocuktan gelişigüzel birinin seçilmesi.
- Mutasyon.
- Ekle ve en kötünün yerini değiştir.

26.6.4. CHC

Bu genetik algoritmanın özellikleri şu şekildedir.

- Populasyon boyutu küçük seçilir. En fazla 50 kromozomlu populasyon üretilir.
- Kısıltmalı seçim ile beraber gelişigüzel çiftleştirme.
- Yeterince farklı olan bireylerin çiftleşmelerine izin verilir.
- En iyi çözümle beraber gelişigüzel çözüm ile bir sonraki iterasyona başlanır.
- Yarı düzenli çaprazlama.

26.6.5. Hibrid Genetik Algoritmalar

Basit genetik algoritmaya yerel arama özelliğinin katılması ile elde edilen bir yöntemdir. Yerel arama adımı üç tane parametre ile tanımlanır. Yerel aramanın frekansı, yerel aramanın olasılığı ve yerel aramanın iterasyon sayısıdır. Algoritmanın tanımlanması için ilk eleman yerel arama frekansıdır ve bu global ve yerel arama arasındaki geçiş sayısını belirler.

Bu yöntemin amacı, global olarak populasyon biraz iyileştirildikten sonra yerel olarak yapılabilecek iyileştirmelerin yapılmasıdır. Ondan sonra eğer hala çözüme ulaşılmadıysa, genel olarak biraz daha iyileştirme sağlanır ve bunun peşinde yerel iyileştirme yapılır. Bu şekilde çözüm bulununcaya kadar devam edilir.

2.6.6. Paralel Genetik Algoritmalar

Bu genetik algoritma, paralel mimariler üzerinde kullanılan bir yöntemdir. Kendi içerisinde çeşitlilik gösterir ve bunlar şu şekilde sıralanabilir.

- Sahip-Köle (Master-Slave): Bu yöntemde bir tane populasyon vardır ve çaprazlama, mutasyon ve seçme işlemi bu populasyonun üzerinde bulunduğu işlemci tarafından gerçekleştirilirken, köle işlemciler değerlendirme işlemini gerçekleştirirler.
- Birden Fazla Populasyon Olması: Birden fazla populasyon vardır ve her populasyon kendi içerisinde genetik süreçten geçer.

- İyi Paracıklama: Bu yntemde her iřlemci kendi zerinde yerel olarak seme ve iftleřtirme yeteneėine sahiptir. Aynı zamanda bu yntem hcresel paralel genetik algoritma olarak da bilinmektedir.
- Hiyerarřık Kombinasyonlar: Birden fazla populasyon vardır ve her populasyonun bulunduėu iřlemci zerinde aprazlama, mutasyon ve seme iřlemleri gerekleřtirilirken bu iřlemcinin kle iřlemcisi durumundaki iřlemciler deėerlendirme iřlemini gerekleřtirirler. Bu řekilde Sahip-Kle yntemi ile Birden Fazla Populasyon yntemlerinin beraber kullanılmasını saėlar.