

## GENETİK ALGORİTMALAR ve UYGULAMA ALANLARI

**Gül Gökay EMEL\***  
**Çağatan TAŞKIN\*\***

### Özet

*Bu çalışmanın amacı bir arama ve optimizasyon yöntemi olan genetik algoritmayı ve uygulama alanlarını incelemektir. Çalışmada ilk olarak genetik algoritma kavramı ve temel teoremi hakkında bilgi verilmiştir. Daha sonra, basit genetik algoritmanın çalışma adımları ve parametre seçimi incelenmiş ve fonksiyon optimizasyonu için bir çözüm örneği verilmiştir. Genetik algoritmaların çeşitli alanlardaki uygulamaları için literatür araştırması yapılmıştır. Yaygın olarak uygulandığı görülen alanlar, genel ve işletme alanları olmak üzere iki temel gruba ayrılarak incelenmiştir. Ayrıca, araştırmacılara ışık tutmak amacıyla, gelecekte daha yaygın olarak uygulanabileceği işletme alanları tespit edilmeye çalışılmıştır.*

**Anahtar Kelimeler:** Genetik Algoritmalar, Uygulama Alanları.

### Abstract

*The purpose of this study is to describe genetic algorithm, which is a search and optimization method and to give information about its applications. In this study, firstly, information about the genetic algorithm concept and fundamental theorem is given. Then, the working scheme of simple genetic algorithm and parameter selection are examined. An example about function optimization has been solved. A literature survey has been made for the application areas of genetic algorithms in various areas. Application areas which are mostly implemented are*

---

\* Yard. Doç. Dr.; Uludağ Üniversitesi İ.İ.B.F., İşletme Bölümü, Sayısal Yöntemler ABD Öğretim Üyesi.

\*\* Araş. Gör.; Uludağ Üniversitesi İ.İ.B.F., İşletme Bölümü, Üretim Yönetimi ve Pazarlama ABD.

*divided into two basic groups. These are general application and enterprise application areas. Application areas which are likely to be implemented more in the future are mentioned to give a light to the researches.*

**Keywords:** *Genetic Algorithms, Application Areas.*

## 1. GİRİŞ

Günümüzün karmaşık ve zor koşulları problemlere hızlı ve kolay çözüm veren yeni çözüm yöntemleri arayışına neden olmuştur. Özellikle sert(hard) optimizasyon teknikleri yerine, yumuşak hesaplama (soft computing) ve evrimsel algoritma (evolutionary algorithm) kullanımı ön plana çıkmıştır. Evrimsel yaklaşımlardan olan genetik algoritmalar da, bu arayışlar içinde önemli bir yer tutmaya başlamıştır. Uygulama başarıları artan ve sürekli geliştirilmeye çalışılan genetik algoritmalar diğer yumuşak hesaplama yöntemleri ile birlikte kullanılarak hibrid (hybrid) çözümler geliştirilmesine çalışılmaktadır.

Genetik algoritmalar, doğal seçim ilkelerine dayanan bir arama ve optimizasyon yöntemidir. Temel ilkeleri John Holland tarafından ortaya atılmıştır. Temel ilkelerinin ortaya atılmasından sonra, genetik algoritmalar hakkında bir çok bilimsel çalışma yayınlanmıştır. Ayrıca, genetik algoritmaların teorik kısmı ve uygulamaları hakkında bir çok uluslararası konferans da düzenlenmektedir. Genetik algoritmaların, fonksiyon optimizasyonu, çizelgeleme, mekanik öğrenme, tasarım, hücresel üretim gibi alanlarda başarılı uygulamaları bulunmaktadır. Geleneksel optimizasyon yöntemlerine göre farklılıkları olan genetik algoritmalar, parametre kümesini değil kodlanmış biçimlerini kullanırlar. Olasılık kurallarına göre çalışan genetik algoritmalar, yalnızca amaç fonksiyonuna gereksinim duyar. Çözüm uzayının tamamını değil belirli bir kısmını tararlar. Böylece, etkin arama yaparak çok daha kısa bir sürede çözüme ulaşırlar (Goldberg, 1989: 1-7). Diğer bir önemli üstünlükleri ise çözümlerden oluşan popülasyonu eş zamanlı incelemeleri ve böylelikle yerel en iyi çözümlere takılmamalarıdır.

“Reeves; ulusal hükümetler ve organizasyonlar tarafından genetik algoritma tabanlı projelere, tavlama benzetimi (simulated annealing) ve yasaklı arama (tabu search) tabanlı projelere göre daha fazla kaynak ayrılmakta olduğunu belirtmektedir” (Yeniay, 2001: 37). Bu ifadeden anlaşılacağı üzere, genetik algoritmaların geleneksel optimizasyon yöntemlerine olduğu gibi sözü edilen yapay zeka yöntemlerine göre de çeşitli alanlarda üstünlükleri bulunmaktadır. Bu üstünlükler genetik algoritmaların arama yapısı ile ilgilidir. Genetik algoritmaların arama yapısı ise, alt diziler teoremi ve yapı blokları hipoteziyle açıklanmaktadır.

## 2. GENETİK ALGORİTMALARIN TEMEL TEOREMİ

Genetik algoritmaların nasıl arama yaptığı alt dizi kavramıyla açıklanmaktadır. Alt diziler, genetik algoritmaların davranışlarını açıklamak için kullanılan teorik yapılardır. Bir alt dizi, belirli dizi kümeleri arasındaki benzerliği tanımlayan bir dizidir. Alt diziler,  $\{0, 1, *\}$  alfabesi kullanılarak tanımlanır. Örneğin H alt dizisi, ilk konumunda 0, ikinci ve dördüncü konumunda 1 değeri olan kromozomlar kümesi içindir.

$$H = 0 1 * 1 *$$

\* sembolü dizinin o konumunun hangi değeri alıp almadığının önemli olmadığı anlamındadır. Dizi o konumda 0 veya 1 değeri alabilir. Eğer bir x dizisi, alt dizinin kalıbına uyarsa x dizisine “H’nin bir örneğidir” denir. Alt dizilerin iki özelliği mevcuttur. Bu özellikler aşağıda verilmiştir (Goldberg, 1989:28-29).

**1. Alt dizi derecesi:** Bir H alt dizisinin derecesi  $o(H)$  ile gösterilir ve mevcut alt dizi kalıbında bulunan sabit konumların sayısıdır. Bu sayı ikili alfabede 0 ve 1 değerlerinin sayısının toplamına eşittir.

**2. Alt dizi uzunluğu:** Bir H alt dizisinin uzunluğu  $\delta(H)$  ile gösterilir ve mevcut alt dizi kalıbında bulunan belirli ilk ve son konumlar arasındaki uzaklıktır.

Alt dizi derecesi ve alt dizi uzunluğu kavramlarının genetik algoritmaların temel teoreminde son derece önemli bir yeri vardır. Alt dizi derecesi düşük, alt dizi uzunluğu kısa olan diziler “yapı blokları” olarak adlandırılır. John Holland, genetik algoritmaların işleyişinde uygun yapı bloklarının tanımlanmasını ve bu yapı bloklarının daha uygun yapı blokları elde etmek amacıyla birleştirilmesini önermektedir. Bu fikir yapı blokları hipotezi olarak bilinmektedir. Genetik algoritmanın temel teoremi ise şöyle açıklanmaktadır (Yeniay, 2001: 41):

Populasyon ortalamasının üstünde uyum gücü gösteren, kısa uzunluğa ve düşük dereceye sahip alt diziler zamanın ilerlemesiyle üstsel olarak çoğalırlar.

Bu çoğalma, genetik işlemler aracılığı ile gerçekleşmektedir ve sonucunda ana-babadan daha üstün özellikler taşıyan bireyler ortaya çıkmaktadır. Bu çözüm kalitesinin kuşaktan kuşağa artması iki nedene bağlanmaktadır. Bu nedenler şöyle açıklanabilir (İşlier, 2001: 146):

- Başarısız olan bireylerin üreme şansları azaltıldığı için kötüye gidip zorlaşmaktadır.

- Genetik algoritmaların yapısı kötüye gidişi engellemekle kalmakta, genetik algoritmaların temel teoremi uyarınca, zaman içinde hızlı bir iyiye gidiş de sağlayabilmektedir.

Genetik algoritmaların işleme adımları incelendiğinde bu nedenler daha iyi anlaşılmaktadır. Genetik algoritmalar yapısı gereği, kötü bireyleri yani uygun olmayan çözümleri, operatörleri sayesinde elemektedir. Bu işlemler bir döngü içerisinde durdurma kriteri sağlanana kadar devam etmektedir.

### 3. BASİT GENETİK ALGORİTMA

Bir çok alanda uygulama imkanı ve uygulamaları olan genetik algoritmaların işleme adımları şöyle açıklanabilir(Engin,2001:21):

- Arama uzayındaki tüm mümkün çözümler dizi olarak kodlanır.
- Genellikle rastsal bir çözüm kümesi seçilir ve başlangıç popülasyonu olarak kabul edilir.
- Her bir dizi için bir uygunluk değeri hesaplanır, bulunan uygunluk değerleri dizilerin çözüm kalitesini gösterir.
- Bir grup dizi belirli bir olasılık değerine göre rastsal olarak seçilip çoğalma işlemi gerçekleştirilir.
- Yeni bireylerin uygunluk değerleri hesaplanarak, çaprazlama ve mutasyon işlemlerine tabi tutulur.
- Önceden belirlenen kuşak sayısı boyunca yukarıdaki işlemler devam ettirilir.
- İterasyon, belirlenen kuşak sayısına ulaşıncaya işlem sona erdirilir. Amaç fonksiyonuna göre en uygun olan dizi seçilir.

Genetik algoritmalar bir çözüm uzayındaki her noktayı, kromozom adı verilen ikili bit dizisi ile kodlar. Her noktanın bir uygunluk değeri vardır. Tek bir nokta yerine, genetik algoritmalar bir popülasyon olarak noktalar kümesini muhafaza eder. Her kuşakta, genetik algoritma, çaprazlama ve mutasyon gibi genetik operatörleri kullanarak yeni bir popülasyon oluşturur. Birkaç kuşak sonunda, popülasyon daha iyi uygunluk değerine sahip üyeleri içerir. Bu, Darwin'in rastsal mutasyona ve doğal seçime dayanan evrim modellerine benzemektedir. Genetik algoritmalar, çözümlerin kodlanmasını, uygunlukların hesaplanmasını, çoğalma, çaprazlama ve mutasyon operatörlerinin uygulanmasını içerir (Jang, 1997: 175-176).

### 3.1. Çözümlerin Kodlanması

Bir problemin çözümü için genetik algoritma geliştirmenin ilk adımı, tüm çözümlerin aynı boyutlara sahip bitler dizisi biçiminde gösterilmesidir. Dizilerden her biri, problemin olası çözümler uzayındaki rastsal bir noktayı simgeler (Yeniay, 2001: 38). Parametrelerin kodlanması, probleme özgü bilgilerin genetik algoritmanın kullanacağı şekle çevrilmesine olanak tanır (Jang, 1997: 176).

### 3.2. İlk Populasyonun Oluşturulması

Olası çözümlerin kodlandığı bir çözüm grubu oluşturulur. Çözüm grubu populasyon, çözümlerin kodları da kromozom olarak adlandırılır. İkili alfabenin kullanıldığı kromozomların gösteriminde, ilk populasyonun oluşturulması için rastsal sayı üreticileri kullanılabilir. Rastsal sayı üreticisi çağrılır ve değer 0,5'den küçükse konum 0'a değilse 1 değerine ayarlanır (Yeo ve Agyel, 1998: 269). Birey sayısının ve kromozom uzunluğunun az olduğu problemlerde yazı-tura ile de konum değerleri belirlenebilmektedir. Genetik algoritmalarda ikili kodlama yöntemi dışında, çözümü aranan probleme bağlı olarak farklı kodlama yöntemleri de kullanılmaktadır (Goldberg, 1989: 9).

### 3.3. Uygunluk Değerinin Hesaplanması

Bir kuşak oluşturulduktan sonraki ilk adım, popülasyondaki her üyenin uygunluk değerini hesaplama adımıdır. Örneğin, bir maksimizasyon problemi için  $i$ . üyenin uygunluk değeri  $f(i)$ , genellikle o noktadaki amaç fonksiyonunun değeridir (Jang, 1997: 176). Çözümü aranan her problem için bir uygunluk fonksiyonu mevcuttur. Verilen belirli bir kromozom için uygunluk fonksiyonu, o kromozomun temsil ettiği çözümün kullanımıyla veya yeteneğiyle orantılı olan sayısal bir uygunluk değeri verir. Bu bilgi, her kuşakta daha uygun çözümlerin seçiminde yol göstermektedir. Bir çözümün uygunluk değeri ne kadar yüksekse, yaşama ve çoğalma şansı o kadar fazladır ve bir sonraki kuşakta temsil edilme oranı da o kadar yüksektir (Yeniay, 2001: 38-39).

### 3.4. Çoğalma İşleminin Uygulanması

Çoğalma operatöründe diziler, amaç fonksiyonuna göre kopyalanır ve iyi kalıtsal özellikleri gelecek kuşağa daha iyi aktaracak bireyler seçilir. Üreme operatörü yapay bir seçimdir. Dizileri uygunluk değerlerine göre kopyalama, daha yüksek uygunluk değerine sahip dizilerin, bir sonraki kuşaktaki bir veya daha fazla yavruya daha yüksek bir olasılıkla katkıda bulunması anlamına gelmektedir. Çoğalma, bireyleri seçme işleminden,

seçilmiş bireyleri bir eşleme havuzuna kopyalama işleminden ve havuzda bireyleri çiftler halinde gruplara ayırma işleminden oluşur (Fırlalı, 2002: 3).

Uygunluk değerinin hesaplanması adımımdan sonra mevcut kuşaktan yeni bir popülasyon yaratılmalıdır. Seçim işlemi, bir sonraki kuşak için yavru üretmek amacıyla hangi ailelerin yer alması gerektiğine karar vermektedir. Bu doğal seçimdeki en uygunun yaşaması durumuna benzerdir. Bu yöntemin amacı, ortalama uygunluğun üzerindeki değerlere çoğalma fırsatı tanımadır. Bir dizinin kopyalanma şansı, uygunluk fonksiyonuyla hesaplanan dizinin uygunluk değerine bağlıdır (Jang, 1997: 176). Seçim yöntemlerine rulet tekerleği seçimi, turnuva seçimi ve sıralama seçimi gibi seçim yöntemleri örnek verilebilir.

### 3.5. Çaprazlama İşleminin Uygulanması

Mevcut gen havuzunun potansiyelini araştırmak üzere, bir önceki kuşaktan daha iyi nitelikler içeren yeni kromozomlar yaratmak amacıyla çaprazlama operatörü kullanılmaktadır. Çaprazlama genellikle, verilen bir çaprazlama oranına eşit bir olasılıkla seçilen aile çeşitlerine uygulanmaktadır (Jang, 1997: 176).

Genetik algoritmanın performansını etkileyen önemli parametrelerden biri olan çaprazlama operatörü doğal popülasyonlardaki çaprazlamaya karşılık gelmektedir. Çoğalma işlemi sonucunda elde edilen yeni popülasyondan rastsal olarak iki kromozom seçilmekte ve karşılıklı çaprazlama işlemine tabi tutulmaktadır. Çaprazlama işleminde dizi uzunluğu  $L$  olmak üzere,  $1 \leq k \leq L-1$  aralığında  $k$  tamsayısı seçilmektedir. Bu tamsayı değerine göre dizi çaprazlamaya uğratılır. En basit çaprazlama yöntemi tek noktalı çaprazlama yöntemidir. Tek noktalı çaprazlama yapılabilmesi için her iki kromozomun da aynı gen uzunluğunda olması gerekir. İki noktalı çaprazlamada ise kromozom iki noktadan kesilir ve karşılıklı olarak pozisyonlar yer değiştirilir (Fırlalı, 2002: 3).

### 3.6. Mutasyon İşleminin Uygulanması

Çaprazlama mevcut gen potansiyellerini araştırmak üzere kullanılır. Fakat popülasyon gerekli tüm kodlanmış bilgiyi içermez ise, çaprazlama tatmin edici bir çözüm üretmez. Bundan dolayı, mevcut kromozomlardan yeni kromozomlar üretme yeteneğine sahip bir operatör gerekmektedir. Bu görevi mutasyon gerçekleştirir. Yapay genetik sistemlerde mutasyon operatörü, bir daha elde edilemeyebilir iyi bir çözümün kaybına karşı koruma sağlamaktadır (Goldberg, 1989: 14). İkili kodlama sisteminin kullanıldığı problemlerde mutasyon, düşük bir olasılık değeri altında bir bit değerini (0 veya 1 olabilir) diğer bit değerine dönüştürür. İkili kodlama sisteminin kullanılmadığı problemlerde ise daha farklı mutasyon yöntemleri

kullanılmaktadır. Hangi yöntem kullanılırsa kullanılsın, mutasyonun genel amacı, genetik çeşitliliği sağlamak veya korumaktır (Braysy, 2001: 54).

### 3.7. Yeni Kuşağın Oluşması ve Döngünün Durdurulması

Yeni kuşak çoğalma, çaprazlama ve mutasyon işlemlerinden sonra tanımlanmakta ve bir sonraki kuşağın ebeveynleri olmaktadır. Süreç yeni kuşakla çoğalma için belirlenen uygunluk ile devam eder. Bu süreç, önceden belirlenen kuşak sayısı kadar veya bir hedefe ulaşıncaya kadar ya da başka bir durdurma kriteri sağlanana kadar devam eder (Yeo ve Agyel, 1998: 271). İstenen hassasiyet derecesine göre de maksimum iterasyon sayısı belirlenilmekte ve iterasyon bu sayıya ulaştığında döngü durdurulabilmektedir. Durdurma kriteri iterasyon sayısı olabileceği gibi hedeflenen uygunluk değeri de olabilmektedir (Fung, Tang ve Wang, 2001: 270).

## 4. GENETİK ALGORİTMALARDA PARAMETRE SEÇİMİ

Parametreler, genetik algoritma performansı üzerinde önemli etkiye sahiptir. Optimal kontrol parametreleri bulmak için bir çok çalışma yapılmıştır fakat tüm problemler için genel olarak kullanılacak parametreler bulunamamıştır (Altıparmak, Dengiz ve Smith, 2000: 61-62). Bu parametreler, kontrol parametreleri olarak adlandırılmaktadır. Kontrol parametreleri popülasyon büyüklüğü, çaprazlama olasılığı, mutasyon olasılığı, kuşak aralığı, seçim stratejisi ve fonksiyon ölçeklemesi olarak sayılabilir. Bu parametreler aşağıda açıklanmıştır (Yeniay, 2001: 40; Sinreich, 1999: 89; Deb: 10):

- **Popülasyon Büyüklüğü:** Genetik algoritma kullanıcısı tarafından verilen en önemli kararlardan birisidir. Bu değer çok küçük olduğunda, genetik algoritma yerel bir optimuma takılabilmektedir. Popülasyonun çok büyük olması ise çözüme ulaşma zamanını arttırmaktadır. Bu konuda Goldberg 1985'de, yalnızca kromozom uzunluğuna bağlı bir popülasyon büyüklüğü hesaplama yöntemi önermiştir. Ayrıca Schaffer ve arkadaşları 1989'da çok sayıda test fonksiyonları üzerinde yaptıkları araştırmalar sonucunda, 20-30 arası bir popülasyon büyüklüğünün iyi sonuçlar verdiğini belirtmişlerdir.

- **Çaprazlama Olasılığı:** Çaprazlamanın amacı, mevcut iyi kromozomların özelliklerini birleştirerek daha uygun kromozomlar yaratmaktır. Kromozom çiftleri  $P(c)$  olasılığı ile çaprazlamaya uğramak üzere seçilirler. Çaprazlamanın artması, yapı bloklarının artmasına neden olmakta fakat aynı zamanda bazı iyi kromozomların da bozulma olasılığını arttırmaktadır.

- **Mutasyon Olasılığı:** Mutasyonun amacı popülasyondaki genetik çeşitliliği korumaktır. Mutasyon  $P(m)$  olasılığı ile bir kromozomdaki her

bitte meydana gelebilir. Eğer mutasyon olasılığı artarsa, genetik arama rastsal bir aramaya dönüşür. Fakat bu aynı zamanda kayıp genetik malzeme-yi tekrar bulmada yardımcı olmaktadır.

- **Kuşak Aralığı:** Her kuşaktaki yeni kromozom oranına kuşak aralığı denmektedir. Genetik operatörler için kaç tane kromozomun seçildiğini gösterir. Yüksek bir değer bir çok kromozomun yer değiştirdiği anlamına gelmektedir.

- **Seçim Stratejisi:** Eski kuşağı yenilemenin çeşitli yöntemleri mevcuttur. Kuşaksal stratejide, mevcut popülasyondaki kromozomlar tamamen yavrular ile yer değiştirir. Popülasyonun en iyi kromozomu da yenilediğinden dolayı bir sonraki kuşağa aktarılamaz ve bu yüzden bu strateji en uygun (elitist) stratejisiyle beraber kullanılmaktadır. En uygun stratejisinde, popülasyondaki en iyi kromozomlar hiçbir zaman yenilenmemektedir, bundan dolayı çoğalma için en iyi çözüm her zaman elverişlidir. Denge durumu stratejisinde ise, her kuşakta yalnızca birkaç kromozom yenilenmektedir. Genellikle, yeni kromozomlar popülasyona katıldığıında en kötü kromozomlar yenilenir.

- **Fonksiyon Ölçkleme:** Doğrusal ölçkleme, üstsel ölçkleme gibi yöntemler mevcuttur. Probleme göre en uygun ölçkleme yönteminin seçilmesi genetik algoritmanın etkin işlemesi açısından önem taşımaktadır.

## 5. GENETİK ALGORİTMANIN BİR FONKSİYON OPTİMİZASYONUNDA UYGULANMASI

Burada, genetik algoritmaların nasıl çalıştığını göstermek için tek değişkenli bir fonksiyonun optimizasyonu ele alınacaktır. Örnek fonksiyon aşağıdaki gibi olup, grafiği Şekil 1’de gösterilmiştir:

$$y = 2\cos(50x) + 3\sin(30x) \quad 0 \leq x \leq 12$$

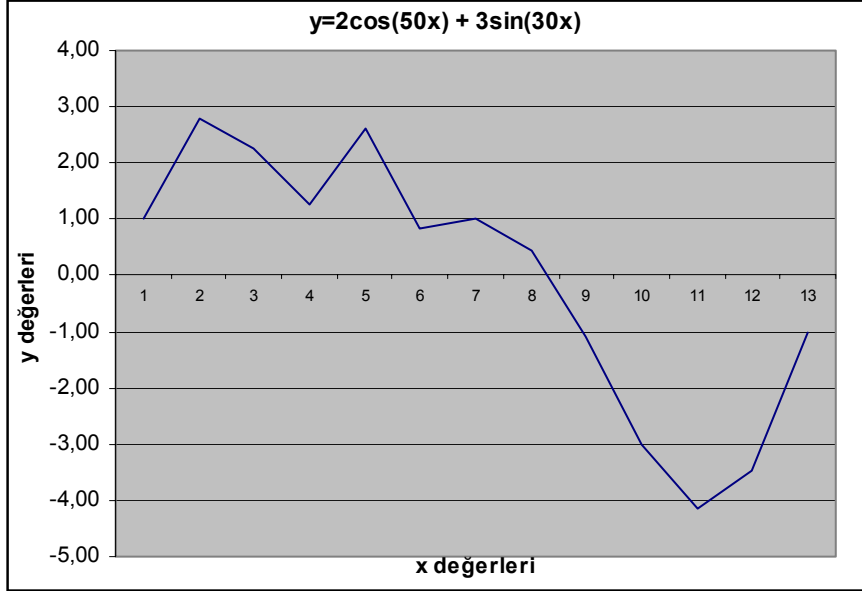
Amaç, 0 ile 12 aralığında fonksiyonu minimize eden  $x^*$  değerini bulmaktır. Çözümü zor olmayan bu problem çeşitli yöntemlerle ve elle çözülebilir. Bu yöntemlerden biri ile çözüldüğünde,  $x^*=10,2$ 'ye karşılık gelen  $f(x^*)$  değeri  $-4.15$  olarak bulunmaktadır. Elde edilen değer, burada sadece genetik algoritmayla yapılan çözümün sonucu ile karşılaştırma amacıyla hesaplanmıştır. Genetik algoritmayla fonksiyonun optimum değerini bulmak için ise Turbo Pascal’da kodlanmış Goldberg’in basit genetik algoritma kodu örneğe uyarlanarak kullanılmıştır.

Problemin çözümü için ikili kodlama kullanılmış ve kromozomlar 30 bitlik dizilerden oluşturulmuştur. Kromozom uzunluğu genellikle gerçek



uygulamalarda uzun alınmaktadır. İlk populasyon oluşturulmuş, minimum değer olarak  $-4,9039264020$  elde edilmiştir. Kullanılan genetik algoritmada, çaprazlama ve mutasyon olmak üzere iki temel genetik operatör kullanılmıştır. Parametreler ise şöyle seçilmiştir:

- Populasyon büyüklüğü : 100  
 Çaprazlama olasılığı : 0,70  
 Mutasyon olasılığı : 0,01  
 Durdurma kriteri : 15 kuşak  
 Seçim stratejisi : kuşaksal strateji



**Şekil 1.**  $F(x)=2\cos(50x) + 3\sin(30x)$  Grafiği

Deney sonuçları: İlk populasyon yaratıldıktan sonra program, pentium 120 mHz bir PC’de çalıştırılmıştır. Hesaplama zamanı 13 saniye olarak ölçülmüştür. Tablo 1’de iyileşmenin görüldüğü kuşaklar ve minimum değerler özetlenmiştir.

Genetik algoritmanın çalıştırılması sonucunda 12. kuşakta minimum değer bulunmuştur. Bu değer, optimal çözüme çok yakın olup kısa bir sürede elde edilmiştir. Kullanılan parametreler ise, yapılan çeşitli deneysel çalışmalara ve literatürde karşılaşılan benzer örneklere dayanılarak belirlenmiştir. Örnekte uygulanan genetik algoritmanın akış diyagramı Ek 1’ de verilmiştir.

**Tablo 1. İyileşmenin Görüldüğü Kuşaklar ve Uygunluk Değerleri**

Kuşak Sayısı	Minimum Uygunluk Değeri
0	-4,9039264020
1	-4,6593255932
2	-4,5596444645
9	-4,1082673943
12	-4,1131100874

## 6. GENETİK ALGORİTMALARIN UYGULAMA ALANLARI

Karmaşık problemleri hızlı ve optimale yakın olarak çözebilen genetik algoritmalar, çeşitli problem tiplerine uygulanabilmektedir. Büyük çözüm uzaylarının geleneksel yöntemlerle taranması hesaplama zamanını arttırmaktadır. Ancak bu tip problemlere, genetik algoritmalar ile kısa sürede, kabul edilebilir çözümler bulunabilmektedir (Gonzales, 2000: 683). Genetik algoritmalar özellikle çözüm uzayının geniş, süreksiz ve karmaşık olduğu problem tiplerinde başarılı sonuçlar vermektedir.

Genetik algoritmaların uygulama alanları bu çalışmada genel uygulama alanları ve işletmelerdeki yaygın uygulama alanları olmak üzere iki sınıfa ayrılarak incelenmiştir.

### 6.1. Genel Uygulama Alanları

Genetik algoritmaların genel uygulama alanları aşağıdaki gibi verilebilir:

#### • Optimizasyon

Bir arama yöntemi olan genetik algoritmalar, farklı bilim dallarındaki optimizasyon problemlerini çözmeye kullanılmaktadır. Genetik algoritmaların uygulandığı optimizasyon problemleri, fonksiyon optimizasyonu ve birleşim (combinatorial) optimizasyonu altında toplanabilir (<http://www.doc.ic.ac.uk>).

Genetik algoritma araştırmalarının önemli bir bölümü fonksiyon optimizasyonu ile ilgilidir. Genetik algoritmalar, geleneksel optimizasyon tekniklerine göre zor, süreksiz ve gürültü (noisy) içeren fonksiyonları çözmeye daha etkindirler (Beasley, Bull ve Martin, 1993: 13). Optimize edilecek amaç fonksiyonunun süreksiz olması halinde, süreksizlik noktalarında fonksiyonun türevi alınamayacağından, türev almaya dayalı optimizasyon yöntemleri kullanılamamaktadır. Oysa, genetik algoritmalar, problemlerin çözümü için türev veya diğer yardımcı bilgilere gereksinim

duymadığından özellikle bu tip problemlerin çözümünde geleneksel yöntemlere göre önemli bir üstünlük sağlamaktadır (Karr ve Freeman, 1999: 4).

Genetik algoritmaların uygulandığı diğer bir optimizasyon problem sınıfı olan birleşti optimizasyon problemleri ise, istenen amaçlara ulaşmak üzere, sınırlı kaynakların etkin tahsis edilmesiyle ilgilidir. Bu sınırlar genel olarak, işgücü, tedarik veya bütçe ile ilgilidir. Sözü geçen “birleşti” kelimesi, yalnızca sonlu sayıda alternatif uygun çözümün mevcut olması ile ilgilidir. Birleşti optimizasyon, iyi tanımlanmış bir problem uzayında bir veya daha fazla optimal çözüm bulma sürecidir. Bu tip problemler yönetim biliminin tüm dallarında da (finans, pazarlama, üretim, stok kontrolü, veri-tabanı yönetimi vb.) ortaya çıkmaktadır. Gezgin satıcı problemi, araç rotalama problemi, Çinli postacı problemi, iş atölyesi çizelgeleme problemi, atama problemi, yerleşim tasarımı problemi ve sırt çantası problemi birleşti optimizasyon problemlerine örnektir (Hoffman ve Padberg: 1-4). Birleşti optimizasyon problemlerinde, incelenen değişken sayısı arttıkça çözüme ulaşma zamanı üstsel olarak artmaktadır. Çözüm uzayının tamamının taranmasını gerektiren geleneksel çözüm yöntemlerinde problem çözümü değişken sayısının artmasıyla imkansız hale gelebilmektedir. Genetik algoritmalar ise çözüm uzayının yalnızca belirli bir kısmını taradığı ve eş zamanlı arama yaptığı için, bu tip problemlerde çözüme daha kısa sürede ulaşabilmektedir.

Çeşitli avantajlarına rağmen genetik algoritmaların uygulamalarında bir takım sorunlarla da karşılaşmaktadır. Bu sorunları aşmak için çeşitli yöntemler geliştirilmiştir. Buna kısıtların ele alınmasındaki soruna karşı ceza fonksiyonu yönteminin kullanılması örnek verilebilir. Ancak, bulunan çeşitli yöntemlere rağmen bu konuda yeni yaklaşımlara gereksinim duyulmaktadır (Deb: 16-17).

#### • Otomatik Programlama ve Bilgi Sistemleri

Genetik algoritmaların yaygın olarak kullanıldığı alanlardan biri, belirli ve özel görevler için gerekli olan bilgisayar programlarını geliştirmedir. Ayrıca, diğer hesaplama gerektiren yapıların tasarımı için de kullanılmaktadır. Bunlara örnek olarak, bilgisayar çipleri tasarımı, ders programı hazırlanması ve ağların çizelgelenmesi verilebilir (<http://www.doc.ic.ac.uk>).

Genetik algoritmalar kullanılarak dağıtılmış bilgisayar ağlarının tasarımı da gerçekleştirilmektedir. Bu problem tipinde ağ güvenilirlik parametrelerini (çap, ortalama uzaklık ve bilgisayar ağ güvenilirliği gibi) optimize etmek için birden fazla amaç fonksiyonu kullanılmaktadır. Genetik algoritmalar ile 100 düğüme kadar olan ağlar başarıyla tasarlanmıştır. Ağ tasarımında genetik algoritmaların kullanılması, tasarım sürelerinin ve maliyetlerinin azalmasında önemli bir katkı sağlamıştır. Özellikle, maksimum miktardaki verinin minimum iletişim hattıyla taşınmasında yüksek bir

performans göstermiştir (Davis ve Coombs,1987: 252-256). Ayrıca genetik algoritmaların kullanımıyla, çeşitli alanlara dağıtılmış bir sistem için en uygun dosya tahsisatı gerçekleştirilmektedir (<http://www.doc.ic.ac.uk>).

#### • Mekanik Öğrenme

Mekanik öğrenme; ilki, gözlenmiş bir veri takımını anlamak ve yorumlamak, ikincisi de görülmemiş objelerin özelliklerini tahmin etmek olan iki temel amaç için model kurmayı amaçlar. Parametrik istatistikten ziyade çok büyük veri takımlarının yönetimi üzerinde çalışır. Kullandığı metotların çoğu dağılımdan bağımsız metotlar olarak sınıflanabilir. Uygun model seçimi için işe problem hakkındaki varsayımlarla başlamaz. Onun yerine uygun model yapısını belirlemek için doğrudan mevcut veriden hareketle bir araç kutusu yaklaşımı kullanır (Wehenkel: 3).

Sınıflama sistemi, genetik algoritmaların mekanik öğrenme alanında bir uygulamasıdır (Karr ve Freeman, 1999: 299). Basit dizi kurallarını öğrenen bir mekanik öğrenme sistemi olan sınıflama sisteminin kural ve mesaj sistemi, özel bir üretim sistemi olarak adlandırılabilir. Bu üretim sistemi, “eğer-sonra” kural yapısını kullanır. Bir üretim kuralı, “eğer” yapısından sonra belirtilen durum için, “sonra” yapısından sonra gelen faaliyetin gerçekleştirilmesini içerir (Goldberg, 1989: 221). Genetik algoritmalar, sınıflama sistemlerinde kural-bulma mekanizması olarak kullanılmaktadırlar (Karr ve Freeman, 1999: 304). Genetik algoritmalar ayrıca, sinir ağlarında ve proteinin yapısal analizinde de kullanılmaktadır (Goldberg, 1989: 217-222).

#### • Ekonomik ve Sosyal Sistem Modelleri

Bir sistemi ölçen ampirik olarak gözlenmiş değişkenler arasındaki matematiksel ilişkiyi keşfetme problemi ekonomide en önemli problemlerden biridir. Pratikte gözlenmiş veri gürültü içerebilir ve kapsanan ilişkileri kesin ve açık bir şekilde açıklayacak bir yol bilinmeyebilir. Bu tip problemler, sembolik sistem tanımlama, kara kutu, veri madenciliği ve modelleme problemleri olarak bilinir. Eğer keşfedilen model, sistemin durum değişkenlerinin gelecek değerlerini tahmin etme için kullanılacaksa problem öngörüleme problemi adını alır. Geleneksel doğrusal, kuadratik ve üstsel regresyon modellerinde sapma hataları minimize edilerek fonksiyonlara uygun sayısal katsayılar bulunur. Buradaki yaklaşım, model seçildikten sonra uygun sayısal katsayıların aranmasıdır. Gerçek problem ise verinin değerlendirilmesi için hangi tip modelin uygun olduğunun kararıdır. Keyfi bir matematiksel ilişkiyi açıklamada bilgisayarlar, bu ilişkiyi formüller ve denklemler aracılığı ile açıklamaktan daha esnektir. Bu nedenle, bu tip ilişki açıklamaları için sembolik regresyon kullanılabilir. Sembolik regresyonlar, hem fonksiyon formunu hem de o fonksiyondaki uygun

katsayıyı araştırmaktadır. Bunu bulma ise, verilen girdiler için arzu edilen çıktıları üreten özel bir hesaplama programı ile program uzayında arama yapmaya benzemektedir. Genetik algoritmaların kullanıldığı genetik programlamayla bu tip problemlere tatmin edici çözümler çok daha kolay getirilebilmektedir (Koza, 1992: 57-75).

Genetik algoritmalar yenilik sürecinin modellenmesi amacıyla da kullanılmaktadır. Ayrıca genetik algoritmaların, fiyat verme stratejilerinin gelişim süreçlerini ve kazanç getiren pazarların ortaya çıkış süreçlerini modelleme alanlarında da kullanımları oldukça yaygındır. Genetik algoritmalar sosyal sistemlerin evrimsel yönlerini anlamak amacıyla kullanılmaktadır. Bunlara örnek olarak işbirliğinin evrimi, iletişimin evrimi ve karıncalardaki iz takibi davranışının evrimi verilmektedir (<http://www.doc.ic.ac.uk>).

## 6.2. İşletmelerdeki Uygulama Alanları

Genetik algoritmalar; başta üretim/işlemler olmak üzere finans ve pazarlama gibi işletmelerin fonksiyonel alanlardaki bir çok farklı iş probleminin çözümü için kullanılmaktadır. Genetik algoritmaların özellikle, kaynak tahsisi, iş atölyesi çizelgelemesi, makine parça gruplaması ve bilgisayar ağ tasarımı gibi çeşitli alanlarda uygulamaları mevcuttur. İşletmelerdeki en yaygın kullanım alanları aşağıdaki gibi verilebilir:

### • Finans

Genetik algoritmalar, finansal modelleme uygulamaları için son derece uygundur. Genetik algoritmalar amaç fonksiyonu odaklıdır. Finans problemlerinde genel olarak amaç fonksiyonları tahmin etme gücüne veya bir kıyaslama sonucuna bağlı getirilerdeki gelişmeleri içerir. Kullanılan araç ve problemler arasında mükemmel bir eşleşme mevcuttur. Özellikle hisse senedi fiyatlarındaki değişim kalıplarını tahmin etmede ve bulmada, kaynak tahsisi ve uluslararası sermaye tahsisi stratejilerini belirlemede genetik algoritmalar kullanılabilir (<http://www.doc.ic.ac.uk>). Bu yaklaşımla, kısıtlanmış portföy optimizasyonu, endeks izleme, işlem maliyetleri ve risk tercihleri kısıtlarının da katıldığı çok dönemli portföy yönetim sistemlerinin kurulması, yine minimum işlem lotlu portföy seçimi problemlerinin çözümü yapılabilmektedir. Daha yüksek getiriler elde etmek için FX piyasalarındaki ticari kuralları geliştirmede (al-tut stratejilerinden daha karlı olanları bularak) genetik algoritmalar kullanılabilir. Ayrıca, müşterilerinin kredi değerliliğini ölçmede, yatırım araçlarının performansını belirlemede, işletmedeki mali kayıpların araştırılmasında, finansal opsiyonların geliştirilmesinde kullanılan veri madenciliğine uygulanabilmektedir. Müşteri kredi değerliliğini ölçme, kredi kartı puanlama, piyasalar ile ilgili tahminleri ve

şirketlerdeki iflas tahminlerini yapma genetik algoritmaların en sık uygulandığı finans problemlerindedir.

Finans problemlerinin çözümünde genetik algoritmalar, bulanık ve yapay sinir ağları yaklaşımlarıyla birlikte kullanılmaktadır. Yumuşak hesaplama ve hibrid genetik algoritma yaklaşımı sık görülmektedir (Schlottmann, 2001: 1). Ayrıca, çözüm performansı açısından finans problemlerindeki genetik algoritma çözümleri yasaklı arama, tavlama benzetimi arama metotları ile karşılaştırılmakta ve o probleme uygun çözüm yöntemi önerilmektedir. Genetik algoritmaların optimal kaynak tahsisi problemlerine uygulanması ile ortalama-varyans optimumundan farklı çözüm yöntemi geliştirilmiş ve kuadratik optimizasyona genetik algoritmalar uygulanmış olmaktadır.

#### • Pazarlama

Tüketicilere ait verileri analiz etmek, çeşitli tüketici kalıpları çıkarmak ve bu kalıplara dayanarak pazarlama stratejileri uygulamak, pazarlamanın en önemli fonksiyonlarından biridir. Tüketicilerin profilleri çıkarılarak, belirli satın alma kalıpları yakalanabilmektedir. Ancak tüketici profilini çıkarabilmek için, çok büyük veri tabanlarını işletme amaçları doğrultusunda hızlı ve etkin biçimde kullanmak gerekmektedir. Burada kullanılan teknik veri madenciliğidir. Veri madenciliği, çok geniş veri tabanlarından veriyi süzme tekniğidir. Pazarı ve tüketiciyi tanımada son derece önemli rol oynayan veri madenciliği, veriyi bilgiye bilgiyi de güvenli kararlara dönüştürür. Veri madenciliğinin verimlilik, karlılık, müşteri tatmini ve rekabet edebilme yeteneği gibi yaşamsal konularda işletme üzerinde çok önemli etkileri bulunmaktadır. Rekabet edebilme yeteneği karar alma kalitesine bağlıdır ve bundan dolayı işletmeler sürekli karar kalitelerini geliştirmeye çalışırlar. Veri madenciliğinde kullanılan tekniklerden birisi de genetik algoritmadır. Genetik algoritma tabanlı yaklaşım kullanılarak veri yığınlarından modeller elde edilmektedir (Bhattacharyya, 1999: 248-249). Bu konuda Siddhartha Bhattacharyya'nın 1999 yılında ve Marshall'ın da aynı yılda yayınlanmış çalışması bulunmaktadır.

#### • Üretim/İşlemler

Genetik algoritmaların en çok uygulandığı alanların başında üretim/işlemler gelmektedir. Burada üretim/işlemler alanıyla ilgili çeşitli problemler ele alınacaktır.

### • Montaj Hattı Dengeleme Problemi

Montaj işlemi endüstrilerde çok önemli bir rol oynamaktadır. Nof ve arkadaşlarının 1997'de yayınlanan çalışmalara göre üretilen mamullerin montajı, toplam üretim zamanının % 50'sine, toplam birim üretim maliyetinin % 20'sine ve işçilik maliyetlerinin % 30-% 50'sine karşılık gelmektedir. Bundan dolayı montaj hattı dengeleme problemi, firmalar açısından yaşamsal öneme sahiptir (Lit, 2001: 3623). Bu konuda; Leu, Motheson ve Rees'in 1994'de, Rubinovitz ve Levitin'in 1995'de, Tsujimura, Gen ve Kubota'nın 1995'de, Sabuncuoğlu, Erel ve Tanyer'in 1999'da, Ponnambalam, Aravindan ve Naidu'nun 2000'de yayınlanmış çalışmaları bulunmaktadır. Bu çalışmalardan Tsujimura, Gen ve Kubota'nın 1995 yılında yayınlanan çalışmasında; her bir iş istasyonundaki toplam işlem zamanlarını minimize etmeyi hedefleyen amaç fonksiyonunun çözümü, genetik algoritma ile bulanık küme mantığı birlikte kullanılarak gerçekleştirilmiştir (Özçakar, 1998: 77).

### • Çizelgeleme Problemi

Genetik algoritmaların çizelgeleme problemine ilk uygulama çalışması, Davis tarafından 1985 yılında yapılmıştır. 1987'de Liepins ve arkadaşları, belirli teslim tarihleri ve işlem süreleri olan işlerin çizelgelenmesi problemini araştırmışlardır. Bu problem en basit çizelgeleme problemi adlandırılmaktadır. 1993'de Gupta ve arkadaşları, akış zamanını minimize etme amacını taşıyan tek makine modeli üzerindeki çalışmalarını yayınlamışlardır. Lee ve Kim 1995'de gecikme ve sarkma cezalarını da modele katan çalışmalarını sunmuşlardır. Cheng ve arkadaşları gene aynı yıl, özdeş paralel makinalardan oluşan model üzerindeki çalışmalarını yayınlamışlardır (Wadhwa ve Chopra, 2000: 2). Bunun dışında; iş atölyesi çizelgeleme problemi için Biegel ve Davern'nin 1990'da, akış atölyesi problemi için Badami ve Parks'ın 1991'de, süreç planlama problemi için Vancza ve Markus'un 1991'de yayınlanmış çalışmaları bulunmaktadır (Yeniay, 2001: 43). Genel olarak genetik algoritmalar, çizelgeleme problemlerine optimale yakın çözüm bulmuşlardır. Fakat çözüm bulma süreleri diğer çözüm yöntemlerine göre oldukça hızlı olmuştur (<http://www.doc.ic.ac.uk>).

### • Tesis Yerleşim Problemi

Tesis yerleşim problemleri araç/gereçleri veya diğer kaynakları belirli bir kritere göre optimum performans sağlayacak şekilde yerleştirme kararını içermektedir. Bu gibi kararlar, araç/gereçlerin genellikle farklı ürünleri üretme esnasında kullanılmasından dolayı karmaşık hale gelmektedir. Her ürünün kendine özgü gereksinimleri olabilir ve tüm ürünler için toplam üretim maliyetinin optimum olması sağlanacak şekilde yerleşim

tasarlanabilir. Yerleşim kararları hızlı ve doğru verilmelidir. Çünkü kararların zayıflığı üretim esnasında ortaya çıkmakta ve bu da artı maliyetlere yol açmaktadır. Örneğin, üretimde robot kullanan işletmelerin tesis yerleşimi tasarımında karmaşıklık söz konusudur. Tek bir robot bir makineden diğerine parçalar taşıırken hareketsiz bir noktada sabitlenir ve yalnızca bir eksen etrafında hareket eder. Robotun hareketine göre, makineler tek-sıra, doğrusal çift-sıra, dairesel tek-sıra ve çoklu-sıra gibi dört farklı yerleşim şekliyle yerleştirilebilir. Burada, dairesel tek-sıra, doğrusal tek-sıranın özel bir durumudur. Ayrıca doğrusal çift-sıra da çoklu sıra probleminin bir alt kümesidir (Dong, 2000: 8). Tesis yerleşim problemleri bunun gibi bir çok zorluğu içermektedir. Genetik algoritmalar, bu tip problemlerin çözümünde uygun bir çözüm yöntemi olabilmektedir. Bu alanda Tam'ın 1992 yılında, Chan ve Tansri'nin 1994 yılında, Tom ve Chan'nin 1998 yılında, İşlier'in 1998 yılında ve Al-Hakim'in 2000 yılında yayınlanmış çalışmaları bulunmaktadır (Yeniay, 2001: 43).

#### • Atama Problemi

Genel olarak atama problemi;  $n$  elemanın,  $n$  farklı göreve atanması problemi.  $i$ . kişinin,  $j$ . işi yapma maliyeti  $c_{ij}$  dir. Bu durumda problem amaç fonksiyonunu minimize edecek  $\{\Pi_1, \dots, \Pi_n\}$  atama kümesinin bulunması şeklinde tanımlanabilir. Burada problem çözümü,  $\{1, \dots, n\}$  sayılarının  $\{\Pi_1, \dots, \Pi_n\}$  permütasyonu olarak gösterilmektedir (<http://www.mmf.gazi.edu.tr>). Genetik algoritmaların atama problemlerine uygulanması konularında, Huntley ve Brown'nun 1991'de, Nissen'nin 1992'de, Tate ve Smith'in 1995'de, Ahuja, Orlin ve Tiwari'nin 2000'de yayınlanmış çalışmaları mevcuttur (Yeniay, 2001: 43). Ayrıca 1996 yılında Chu ve Beasley, minimum maliyetli atamanın hedeflendiği problem için genetik algoritmaların kullanıldığı bir çözüm önermiştir. 1995 yılında Zhao, Tsujimura ve Gen'nin yayınlanan makalelerinde, iş istasyonu atama problemi için genetik algoritma kullanılmıştır (Özçakar, 1998: 78).

#### • Hücresel Üretim Problemi

Hücresel üretim kavramı, üretim sistemlerinin verimliliğini arttırmada anahtar faktörlerden biridir. Hücresel üretim, parça ailelerini belirledikten sonra, her parça ailesini ayrı bir üretim hücresinde imal ederek hücreler arası taşımaları en aza indirmeyi amaçlamaktadır. Genetik algoritmalar, hücreler arası taşımanın minimum olduğu bir hücre kuruluşu amaçlanmasında kullanılabilir. Bu konuda Tate ve Smith'in, Kamrani ve Parsai'nin yayınlanmış çalışmaları bulunmaktadır. Ayrıca, Joines'in 1996'da yayınlanmış çalışması mevcuttur. Venugopal'ın 1999'daki çalışması, hücresel üretim konusu için uygulanmış çözüm tekniklerinin genel bir



değerlendirmesini içermektedir (Brown ve Sumichrast, 2001: 3654). İşler'in bu konudaki çalışmasında ise, üretim hücrelerinin yapısını temsil eden bilgilerin gösterimi, iki bölüm halindedir. Birinci bölümde, tezgah-hücre ilişkileri, bunun devamı olan ikinci bölümde de parça-hücre ilişkileri yer almaktadır. Genetik algoritmalar ile birden fazla çözüm aynı anda ele alınmakta ve bu sayede farklı bölgeler eş zamanlı olarak taranmaktadır. Bunun sonucunda da daha kısa zamanda daha uygun sonuçlar elde edilmektedir (İşler, 2001: 137-149).

#### • Sistem Güvenilirliği Problemi

Bir sistemin güvenilirliği, belirli koşullar altında belirli bir zaman aralığında sistemin başarılı olarak çalışma olasılığı olarak tanımlanmaktadır. Çoğu sistem, çeşitli işlemlerde kritik bir role sahiptir ve eğer sistemde arıza olursa sonuçları oldukça ciddi olmaktadır. Bu alanda optimizasyon, etkisiz parçaların sisteme en iyi şekilde tahsis edilebilme veya yararlanabilme yolunu bulmayı içermektedir. Parçalara, güvenilirliklerinin etkin olarak ölçülebilmesi için olasılıklar atanmaktadır (Dong, 2000: 7). Bu konuda; Painton ve Campbell'in 1995'de, Sasaki, Gen ve Yamashiro'nun 1995'de, Coit ve Smith'in 1996'da, Dengiz, Altıparmak ve Smith'in 1997 ve 2000'de yayınlanmış çalışmaları bulunmaktadır (Yeniay, 2001: 43).

#### • Taşıma Problemi

Taşıma problemi; tedarikçilerden tüketicilere, talebi karşılamak üzere, minimum maliyetle tek tipte mamul gönderilmesini içermektedir. M tane tedarikçi ve n tane de tüketici mevcuttur. Tek tedarikçiden her bir tüketiciye bir birim mamul ulaştırma maliyeti bilinmektedir. Problem, tüm talebin karşılanması ve maliyet minimizasyonu şartıyla mamulün arz yerinden talep yerine optimum tahsisini sağlamaktır. Son zamanlarda, çeşitli taşıma problemlerinin çözümü için evrimsel (evolutionary) yaklaşımlarla çözüm önerileri sunulmaktadır. Michalewicz ve arkadaşları, doğrusal ve doğrusal olmayan taşıma problemleri için genetik algoritma kullanımını ilk öneren araştırmacılarıdır. Ayrıca, Gen ve Li de genetik algoritmaları taşıma problemlerinin çözümü için kullanmışlardır (Gen, 2000: 297).

#### • Gezgin Satıcı Problemi

Genetik algoritmaların, birleşti optimizasyon problemlerine uygulamaları ile ilgili çeşitli çalışmalar mevcuttur. En yoğun yapılan çalışmalardan biri de gezgin satıcı problemleri için yapılmaktadır. Gezgin satıcı probleminde amaç, katedilen toplam mesafeyi minimize eden bir yolculuk planı oluşturmaktır. Bir çok problem tipi gezgin satıcı problemi gibi model-

lenebilmektedir. Bunlara örnek olarak; devre tasarımı, posta taşıyıcılarının, havayolu uçaklarının, okul otobüslerinin rotalarının bulunması verilebilir (<http://www.doc.ic.ac.uk>). Gezgin satıcı probleminin bir özelliği de değişken sayısı arttıkça üstsel artış gösteren zaman ihtiyacı içinde çözüme ulaştırılabilmemesidir. Bu durum bir örnekle şöyle açıklanabilir; bir satış görevlisinin ziyaret etmek durumunda olduğu  $n$  tane şehir olsun. Burada tüm şehirler arasındaki maksimum izlenecek rota sayısı  $(n-1)!$  dir. Tüm mümkün rotaları basitçe inceleyen ve en kısa olan rotayı bulan bir algoritma kullanılır. Fakat şehir sayısı arttıkça algoritmanın hesaplama için gereksinim duyduğu zaman daha da büyük bir oranda artmaktadır. Ziyaret edilmesi gereken 25 şehir varsa, algoritmanın inceleyeceği rota sayısı  $24!$ 'dir. Bu da yaklaşık  $6,2 \times 10^{23}$  sayısına karşılık gelmektedir. Saniyede bir milyon rota inceleme kapasitesine sahip bir bilgisayar, bu problemi,  $6,2 \times 10^{17}$  saniyede yani,  $1,96 \times 10^{10}$  yılda çözebilmektedir (Dong, 2000: 6). Herhangi bir problem için kullanılan algoritmanın en yaygın performans ölçütü, algoritmanın çözüme ulaşma süresidir. Gezgin satıcı gibi değişken sayısı arttıkça çözüm zamanı üstsel olarak artan problemlerde bu daha da önemlidir. Genetik algoritmalar birleşti optimizasyon problemlerini klasik yöntemlere göre çok daha kısa sürede çözmektedir. Sonuçta optimale yakın ve kabul edilebilir bir çözüm bulunmaktadır (Engin, 2001: 9).

#### •Araç Rotalama Problemi

Birleşti optimizasyon problemlerinin örneklerinden biri de araç rotalama problemidir. Temel araç rotalama problemi, talebi belirli olan müşterileri kapsar. Tek bir depodan araçlar ayrılmakta ve müşteri taleplerini karşılayarak tekrar depoya dönmektedir. Her aracın kapasite kısıtı vardır. Bu temel probleme ayrıca, her aracın alacağı yol da mesafe kısıtı olarak eklenebilir. Her bir müşterinin talebini yalnızca bir araç karşılamaktadır. Problem, bu kısıtlar altında minimum toplam maliyeti veren rotaları bulmaktır (Baker ve Ayechev, 2002: 1). Daha karmaşık bir araç rotalama problemi olan zaman pencereli rotalama probleminde ise amaç müşteri talebini belirli zaman aralıkları içerisinde minimum toplam maliyetle karşılamaktır.

Genetik algoritmalar özellikle zaman pencereli araç rotalama problemlerinin çözümü için kullanılmaktadır. Thangiah, Blanton ve Wainwright, Prinetto bu konuda çalışmalarda bulunan araştırmacılardan bazılarıdır (Tan, 2001: 283). Ayrıca, Kopfer, Pankratz ve Erkens ile Filipec, Sklec ve Krajcar'ın bu konuda yayınlanmış çalışmaları bulunmaktadır (Yeniay, 2001: 43).

#### • Minimum Yayılan Ağaç Problemi

Minimum yayılan ağaç problemi, grafik teorisinde klasik bir problemdir. Coğrafi bir alanda dağılmış çeşitli şehirler arasında fiber-optik kablo

döşeme problemi bu tip problemler için uygun örneklerden birisidir. Her şehir arasında kablo döşeme maliyeti bilinmektedir. Problemin amacı, tüm şehirleri fiber-optik kablo ağına bağlayan yerleşimi en düşük maliyetle bulabilmektir. Yerleşim herhangi iki şehir arasında veri paketinin gidip gelmesine, uzaklık ne olursa olsun olanak tanımalıdır. Grafik teorisinde, şehirler vertisler, kablolar da kenarlar olarak tanımlanır. Burada her kenarın bir maliyeti vardır. Bu problem şöyle formüle edilmektedir.  $G = (V, E)$  bağlantılı ve yönlendirilmemiş bir grafik olsun.  $V = \{v(1), v(2), \dots, v(n)\}$  vertisler kümesi ve  $E = \{e(1), e(2), \dots, e(m)\}$  kenarlar kümesi olsun. Her kenarın negatif olmayan  $w$  ağırlığı mevcuttur. Bu da  $w = \{w(1), w(2), \dots, w(m)\}$  olarak gösterilmektedir. Yayılan ağaç, grafikteki tüm vertisleri birbirine bağlayan minimum değerli kenarlar kümesidir. Buradaki “ağaç” tanımı, döngü kuran vertisleri veya yalnızca tek bir vertisi kapsamaz. Kısacası, minimum yayılan ağaç problemi her bir olay veya nokta çiftleri arasındaki kenarların (dalların) en kısa olanının bulunarak ağaç içindeki kenarların birbirleriyle toplam en kısa yolu bulacak şekilde ilişkilendirilmesidir (Taha, 2000: 213, Öztürk, 2001: 266).

Genetik algoritmaları minimum yayılan ağaç problemini çözmek için kullanırken önemli olan nokta, ağacın nasıl kodlanacağıdır. Kodlama; kenar kodlaması, vertis kodlaması veya her iki kodlamanın birleşimi kullanılarak yapılabilmektedir. Genetik algoritmaların etkin sonuç verebilmesi için kodlamanın doğru şekilde yapılması gerekmektedir. Kodlama planında, tüm ağaçlar temsil edilebilmeli ve her ağaç aynı sayıda kodlanmalıdır (Dong, 2000: 5). Bu konuda, Wainwright, Ali ve Schoenefeld’in yayınlanmış çalışması mevcuttur (Wainwright, AbuAli ve Schoenefeld, 1996: 299).

## 7. SONUÇ

Bu çalışmada basit genetik algoritma ve genetik algoritmanın genel ve işletme alanlarındaki problemlere uygulanması literatür araştırması yapılarak incelenmiştir. Genetik algoritmaların kimya, fizik gibi bir çok farklı alanlarda da uygulamaları bulunmasına rağmen bu alanlar çalışmanın kapsamında yer almamıştır.

Çalışmada incelenen problem tiplerinin çözümlerinde başka arama yöntemleri de kullanılmaktadır. Fakat, başlangıç çözümünden bağımsız olma, paralel çözüm arama ve hızlı çalışma özelliklerinden dolayı genetik algoritmalar daha uygun çözüm yöntemleri olarak kabul görmektedir. Sözü edilen problemlerde genetik algoritmaların kullanımında güçlükler de söz konusudur. Bunlardan birisi, problem parametrelerinin belirlenmesidir. Her problem çeşidinin parametreleri farklı olduğu için, bu parametreler genellikle deney tasarımı yöntemiyle belirlenmektedir. Ayrıca genetik algorit-

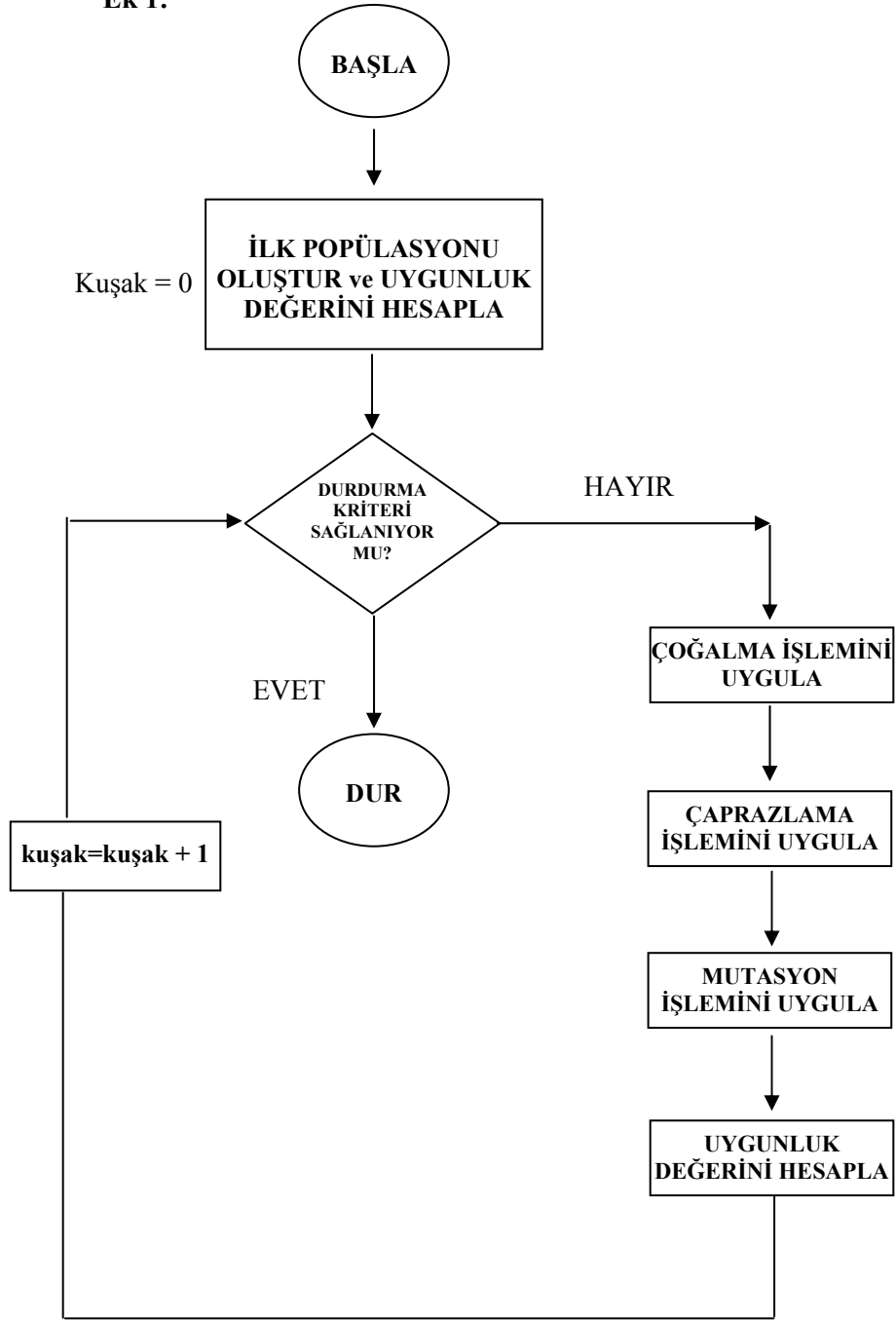
malar kısıt içeren optimizasyon problemlerinin çözümünde de sorunlarla karşılaşabilmektedirler.

Rekabetin gittikçe arttığı günümüzün müşteri odaklı pazarlarında üretim işletmelerinin yaşamlarını sürdürebilmesi için öncelikle iki temel işletme fonksiyonunu en etkin şekilde yerine getirmesi gerekmektedir. Bu fonksiyonlar, üretim ve pazarlama fonksiyonlarıdır. Bu fonksiyonlara ait problemlerinin çözümleri için yoğun çalışmalar yapılmaktadır.

Bu çalışmaların arasında yer alan genetik algoritma çalışmalarının, ilgili literatürün araştırılması sırasında üretim ve tasarım alanlarında oldukça yaygın uygulama alanı olduğu görülmüştür. Pazarlama alanında ise üretim ve tasarım alanlarındaki kadar yaygın bir uygulaması görülmemektedir. Genetik algoritmaların üretim problemlerinde, özellikle çizelgeleme, hücre- sel üretim ve tasarım alanlarında geleneksel yöntemlere göre daha iyi sonuçlar verdiği görülmektedir. Bu tip işletme problemlerinin kısa sürede optimale yakın olarak çözülmesi, üretim planlama ve tasarımda etkin çalışmayı sağlamaktadır. Zamanın maliyet olduğu günümüz pazar şartlarında üretim planlama, tasarım gibi kritik bölümlerin etkin çalışması, işletmenin müşterilerine daha çabuk yanıt vermesi demektir. Bu da önemli bir rekabet avantajı sağlamaktadır.

Pazarlama fonksiyonu açısından konuya bakıldığında ise, tüketici profillerinin ve satın alma kalıplarının bilinmesi işletme açısından son derece kritik bir bilgidir. Bu bilgiyi elde etmeye yönelik bir takım teknikler mevcuttur. Bunların başında, veri yığınları içerisinde belirli kalıpları çıkarmaya yarayan veri madenciliği tekniği gelmektedir. Genetik algoritmalar ise veri madenciliğinde arama amaçlı olarak kullanılmaktadır. Veri madenciliği uygulamalarında, genetik algoritmalar çözüm uzayının yalnızca belirli bir kısmını taraması nedeniyle diğer yöntemlere göre daha hızlı sonuçlar vermektedir. Bu nedenle de genetik algoritmaların veri madenciliğinde bir arama yöntemi olarak, özellikle diğer yumuşak hesaplama yöntemleriyle birlikte çalışarak önemli bir uygulama yeri bulacağı beklenebilir.

## Ek 1:



Şekil 2. Genetik Algoritma Akış Diyagramı

**KAYNAKLAR**

- Altıparmak F., Dengiz B. ve Smith A.E. (2000), “An Evolutionary Approach For Reliability Optimization in Fixed Topology Computer Networks”, *Transactions On Operational Research*, Volume: 12, Number: 1-2, s. 57-75.
- Baker B. M. ve Ayechev M. A. (2002), “A Genetic Algorithm For The Vehicle Routing Problem”, *Computers & Operations Research*, s. 1-14.
- Beasley D., Bull D.R. ve Martin R.R. (1993), “A Sequential Niche Technique For Multimodal Function Optimization.” [www.citeseer.nj.nec.com/beasley93sequential.html](http://www.citeseer.nj.nec.com/beasley93sequential.html). (06/05/2002).
- Bhattacharyya S. (1999), “Direct Marketing Performance Modeling Using Genetic Algorithms”, *Journal on Computing*, Volume: 11, Issue: 3, s. 248-267.
- Braysy O. (2001), Local Search and Variable Neighborhood Search Algorithms for The Vehicle Routing Problem With Time Windows, PH D Thesis.
- Brown C. E. ve Sumichrast R.T. (2001), “CF-GGA: A Grouping Genetic Algorithm For The Cell Formation Problem”, *International Journal of Production Research*, Volume: 39, Number: 16, s. 3651-3669.
- Davis L. ve Coombs S. (1987), “Genetic Algorithms And Communication Link Speed Design: Theoretical Considerations”, *Grefenstette*, s. 252-256.
- Deb K., “Genetic Algorithm in Search and Optimization: The Technique and Applications”, s. 1-29.
- Dengiz B., “Sezgisel Optimizasyon.” [www.mmf.gazi.edu.tr/~berna/turkce/courses/enm543.html](http://www.mmf.gazi.edu.tr/~berna/turkce/courses/enm543.html). (08/05/2002).
- Dong Z. (1999), “Genetic Algorithm Applications.” [www.me.uvic.ca/~zdong/courses/mech620/GA\\_App.PDF](http://www.me.uvic.ca/~zdong/courses/mech620/GA_App.PDF). (07/05/2002), s. 1-29.
- Engin O. (2001), Akış Tipi Çizelgeleme Problemlerinin Genetik Algoritma ile Çözüm Performansının Arttırılmasında Parametre Optimizasyonu, İTÜ, Fen Bilimleri Enstitüsü, Yayınlanmamış Doktora Tezi.
- Fırlı A. ve Engin O. (2002), “Genetik Algoritmalarla Akış Tipi Çizelgelemede Üreme Yöntemi Optimizasyonu”, *İTÜ Dergisi*, s. 1-6.
- Fung R.Y.K. , Tang J. ve Wang D. (2001), “Extension Of A Hybrid Genetic Algorithm For Nonlinear Programming Problems With Equality And Inequality Constraints”, *Computers & Operations Research*, Volume: 29, Issue: 3, s. 261-274.
- Gen M. ve Cheng R. (2000), *Genetic Algorithms and Engineering Optimization*, John Wiley&Sons, Inc., USA.
- “Genetic Algorithms.” [www.doc.ic.ac.uk/~nd/surprise\\_96/journal/vol1/tcw2/article1.html](http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/tcw2/article1.html). (07/05/2002).
- Goldberg D.E. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, USA.

- Gonzales E. L. ve Fernandez M.A.R. (2000), "Genetic Optimisation of A Fuzzy Distribution Model", *International Journal of Physical Distribution & Logistics Management*, Volume 30, Number 7/8, s. 681-696.
- Hoffman K. ve Padberg M., "Combinatorial and Integer Optimization." [www.iris.gmu.edu/~khoffma/papers/newcomb1.html](http://www.iris.gmu.edu/~khoffma/papers/newcomb1.html). (07/05/2002)., s. 1-10.
- İşlier A.A. (2001), "Üretim Hücrelerinin Bir Genetik Algoritma Kullanılarak Oluşturulması", *Anadolu Üniversitesi Bilim ve Teknoloji Dergisi*, Cilt: 2, Sayı: 1, s. 137-157.
- Jang J. S. R. (1997), *Neuro-Fuzzy and Soft Computing: A Computational Approach To Learning and Machine Intelligence*, Chapter 7: Derivative-Free Optimization, Prentice-Hall, USA, s. 173-196.
- Karr C. L. ve Freeman M. L. (1999), *Industrial Applications of Genetic Algorithms*, CRC Press, USA.
- Koza, J. R. (1992), "A Genetic Approach to Econometric Modelling", in Bourguine, P. and B. Walliser (eds.), *Economics and Cognitive Science*, Pergamon Press, s. 57-75
- Lit P., Latinne P. , Rekiek B. ve Delchambre A. (2001), "Assembly Planning With An Ordering Genetic Algorithm", *International Journal of Production Research*, Volume: 39, Number: 16, s. 3623-3640.
- Özçakar N. (1998), "Genetik Algoritmalar", *İstanbul Üniversitesi İşletme Fakültesi Dergisi*, Cilt: 27, Sayı: 1, s. 69-82.
- Öztürk A. (2001), *Yöneylem Araştırması*, Ekin Kitapevi, Bursa.
- Schlottmann F. (2001), "A Hybrid Genetic Quantitative Method For Risk Return Optimisation Of Credit Portfolios." [www.business.uts.edu.au/finance/resources/qmf2001/Schlottmann F.](http://www.business.uts.edu.au/finance/resources/qmf2001/Schlottmann%20F.pdf) (04/05/2002). s. 1-27.
- Sinriech D. ve Samakh E. (1999), "A Genetic Approach to the Pickup/Delivery Station Location Problem in Segmented Flow Based Material Handling Systems", *Journal of Manufacturing Systems*, Volume: 18, Number: 2, s. 81-99.
- Taha H. A. (2000), *Yöneylem Araştırması*, Literatür Yayıncılık, İstanbul.
- Tan K.C., Lee L.H., Zhu Q.L. ve Ou K. (2001), "Heuristic Methods For Vehicle Routing Problems With Time Windows." *Artificial Intelligence in Engineering*. [www.elsevier.com](http://www.elsevier.com). (05/05/2002). s. 281-295.
- Wadhwa S. ve Chopra A. (2000), "A Genetic Algorithm Application: Dynamic Re-configuration in Agile Manufacturing Systems." [www.ici.ro/ici/revista/sic2000\\_4/art01.htm](http://www.ici.ro/ici/revista/sic2000_4/art01.htm). (08/05/2002).
- Wainright R.L., Abuali F.N. ve Schoenefeld D.A. (1996), "Solving The Subset Interconnection Design Problem Using Genetic Algorithms." [www.citeseer.nj.nec.com/484928.html](http://www.citeseer.nj.nec.com/484928.html). (07/05/2002), s. 299-304.
- Wehenkel L., "Machine Learning & Datamining." <http://www.montefiore.ulg.ac.be/~lwh/> . (08/05/2002).

- Yeniay Ö. (2001), "An Overview of Genetic Algorithms", *Anadolu Üniversitesi Bilim ve Teknoloji Dergisi*, Cilt: 2, Sayı: 1, s. 37-49.
- Yeo M. F. ve Agyel E. O. (1996), "Optimising Engineering Problems Using Genetic Algorithms", *Engineering Computations*, Volume: 15, Number: 2, s. 268-280.