

2006 Gazi Üniversitesi Endüstriyel Sanatlar Eğitim Fakültesi Dergisi Sayı:18, s. 49-55

**BİR OPTİMİZASYON YÖNTEMİ: GENETİK ALGORİTMA
“MOLGA PROGRAMI İLE İTERASYON SAYILARININ
DEĞERLENDİRİLMESİ”**

Aysun COŞKUN¹

ÖZET

Bu çalışmada, öncelikle bir çok optimizasyon problemlerinin çözümünde kullanılan genetik algoritma tanıtılmış ve hazırlanan MOLGA(Molekül ve Genetik Algoritma) programı ile genetik algoritmanın iteratif olma özelliği değerlendirilmiştir. MOLGA programı, deriden geçen kimyasal maddelerin yapılarını bulmaya yönelik problemlerin çözümünde kullanılırken yapay zeka optimizasyon teknikleri de genetik algoritma kullanılarak oluşturulmuştur.

Program içerisinde sunulan farklı genetik operatörler ile rasgele belirlenen 10 farklı uygulama için iterasyon sayılarının değerlendirilmesi yapılmıştır.

Anahtar Kelimeler: Genetik Algoritma, İterasyon Sayısı, MOLGA

**AN OPTIMIZATION METHOD: GENETIC ALGORITHM
“EVALUATION OF ITERATION NUMBERS WITH MOLGA PROGRAM”**

ABSTRACT

In this study, genetic algorithm has been introduced, which is used to solve many optimization problems and the iterative features of genetic algorithm has been evaluated by using MOLGA (Molecul and Genetic Algorithm) program. The MOLGA program is used while solving the problem in order to fine the structure of the chemical matters to pass the skin, artificial intelligence optimization techniques were formed by using genetic algorithm.

The evaluation of iteration numbers has been done for different genetic operators which presented in program with randomly determined 10 different applications.

Keywords: Genetic Algorithm, Number of Iteration, MOLGA

¹G.Ü. Endüstriyel Sanatlar Eğitim Fakültesi, Bilgisayar Eğitimi Bölümü, Beşevler/Ankara, aysunc@gazi.edu.tr

1. GİRİŞ

Bir problemin amacına uygun çözümlerin bulunması her bilim dalı için oldukça zor ve karmaşıktır. Bilimsel çalışmalarda çok sayıda alternatif çözümlerin olması muhtemeldir. Bu çözümler içerisinde en iyiyi bulmak ve buna en kısa zamanda ulaşmak problem tasarımı için önemli olmaktadır. Bu yüzden ki optimizasyon problemlerinin çözümü ayrı bir disiplin olmaktadır.

Klasik optimizasyon tekniklerinde, problem tasarımında parametreler arasındaki ilişkiler genelde deneme yanılma yoluyla yapılmış ancak parametre sayısı artınca çözümsüzlük veya elde edilen çözümü değerlendirememesi problemi ortaya çıkmıştır. İstatistiksel yöntemler, araştırmacılara parametreler arasındaki ilişkileri bulmada faydalı olan ilk araçlardandır. İstatistiksel yöntemlerde: verinin normal toplandığı, verinin eşitlik ilişkisinin belirli formda olması(lineer, quadratik ve polinomsal) ve değişkenlerin bağımsız olması gerekir (Hayes,1998:2721). Bu şartlar sağlanabiliyorsa problem çözümü istatistik yöntemlerle yapılabilir. Ancak bütün problemler bu şartları sağlamayabilir.

Yapay zeka'nın ortaya çıkmasından günümüze kadar optimizasyon yöntemlerinde doğal hayat simule edilerek problem çözmek araştırma konusu olmuştur. Yapılan çalışmalarda, yapay zekanın optimizasyon tekniklerinden genetik algoritma öncelikli olarak kullanılmaktadır.

Doğal hayatı taklit ederek oluşturulan optimizasyon tekniklerinden, tabu araştırmaları, ısı işlem algoritmaları, karınca koloni algoritmaları, genetik algoritmalar literatürde en sık kullanılan tekniklerdir (Karaboğa, 2004:19).

Literatürde, bilgisayar bilimi, mühendislik, sosyal bilimler, tıp ve müzik gibi bir çok farklı disiplinde genetik algoritma uygulamalarına rastlamak mümkün olmaktadır. Bu uygulamalar göstermiştir ki, çok parametrelili ve doğrusal olmayan problemlerin çözümünde genetik algoritma başarılı sonuçlar vermektedir.

2. GENETİK ALGORİTMANIN TEMEL PRENSİBİ

Genetik algoritma stokastik ve iteratif bir süreçtir. Algoritmanın başlangıcında çözüm uzayı rasgele veya sezgisel belirlenmektedir. GA'nın en önemli özelliği olan rasgelelik başlangıçtan genetik operatörlerin kullanılmasına kadar algoritma içerisinde sürdürülür. GA tekrarlanan bir süreçtir. Bu özellik uygun çözümün bulunmasına kadar algoritmanın işletilmesinde görülmektedir.

GA diğer yapay zekanın optimizasyon yöntemlerinde olduğu gibi doğal olayları modelleyen bir tekniktir. Özellikle Darwin'in doğal seçme ve en iyi uyum sağlayanın hayatta kalması prensiplerinden ortaya çıkmıştır (Holland,1992:211). Bu yüzden ki özellikle temel prensipte yer alan çaprazlama ve mutasyon genetik algoritmanın önemli operatörlerini oluşturmaktadır.

Basit bir genetik algoritma şu şekilde işlemektedir (Lan Fang, 1994:30).

Adım1. Problem için uygun bir kodlama tasarlanır.

Adım2. Başlangıç popülasyonunu rasgele/sezgisel oluştur.

Adım3. Başlangıç popülasyonunu oluşturan her bir kromozomun uyum değerlerini belirle

Adım4. Başlangıç popülasyonunu değiştirmek için genetik operatörleri uygula

Adım5. Yeni oluşan popülasyonun uyum değerlerini belirle

Adım6. Oluşturulan yeni populasyonu test et

Adım7. Çözüm olumlu ise sonucu ver değil ise adım 4 e git

2.1. Başlangıç populasyonunu oluştur

GA' da ilk adım genellikle mevcut çözümlerin yer aldığı başlangıç populasyonunun oluşturulmasıyla başlar. Uygulamalarda bu adım probleme bağlı olarak rasgele çözümlerden veya sezgisel (heuristik) çözümlerden oluşturulur (Karaboğa, 2004: 81). Yani başlangıçta bazı çözümler tahmin edilebiliyorsa populasyonda bu çözümlere yer verilir. Zaman açısından sezgisel yöntem yararlı olabilmektedir.

2.2. Uygun değerlerin bulunması

Her problemin çözümü için bir uygunluk fonksiyonunun bulunması gerekmektedir. Uygunluk değeri algoritma içerisinde çözümlerin değerlendirilmesine olanak sağlar. Eğer başlangıç populasyonunda yer alan olası çözümün uygunluk değeri istenilen değerde ise veya çözüme yakın ise, bu olası çözüm bir sonraki nesile aktarılır. Ancak istenen değerde değil ise populasyondan çıkarılır veya değiştirilir. Bu işlemler genetik operatörler yardımıyla yapılır.

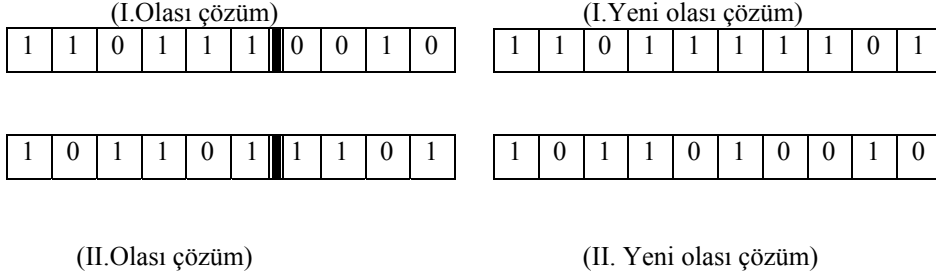
2.3. Genetik operatörler

Olası çözümlerden oluşan populasyon içerisindeki her bir bireye uygulanır. Ya istenilen olası çözümün bir sonraki çözüme aktarımını sağlar ya da istenmeyen olası çözümlerin populasyondan atılmasına veya değiştirilmesini sağlar. Böylece en kötü uyumu gösteren olası çözüm tekrar populasyonda yer almaz.

Genetik algoritmanın en önemli operatörleri çaprazlama ve mutasyondur. Tekrar populasyonda yer alma operatörü uygunluk fonksiyonuna göre değerlendirilir. Olası çözümlerin diğer nesile geçmesi için seçim operatörü kullanılır. En çok kullanılan seçim operatörleri, rulet çemberi ve en kötü bireylerin yerine en iyilerinin yazılması operatörleridir. Seçim işlemi yapıldıktan sonra değişim için çaprazlama ve mutasyon kullanılır.

Çaprazlama operatörü, doğal hayattan bire bir benzetim göstermektedir. Populasyon içerisindeki bireyler arasındaki en iyi özelliklerin bir araya getirilmesi prensibine dayanmaktadır. Böylece uygunluk fonksiyonunun amacına yönelik populasyonun oluşturulması sağlanmış olmaktadır. Bu operatör, tek nokta, iki nokta ve üniform çaprazlama şeklinde uygulanmaktadır (Yeniay, 1999: 17). Bu kullanıcının tercihi veya problemin çözümüne göre değişmektedir.

Şekil 1'de on parametreden ve 2 olası çözümden oluşan örnek üzerinde tek nokta çaprazlama işletilmiştir.



Şekil 1. Tek noktaçaprazlama

Mutasyon operatörü ise popülasyondaki her bir çözüm kendi içerisinde değerlendirilmektedir. Bu operatörün klasik gösterimi eğer olası çözümler ikili sistemle kodlanmışsa çözüm üzerindeki bitler 1 ise 0, 0 ise 1 yapılıdır. Böylece yeni bir olası çözüm oluşturulmuş olur.

Şekil 2’de 101011101 olası çözümünün üçüncü genin mutasyona uğradığı durumu göstermektedir.

Ata 10**1**011101
Yavru 10**0**011101

Şekil 2. Mutasyon

Genetik algoritmanın iteratif bir süreç olması, bu adımların uygun çözümlere ulaşıncaya kadar yürütülmesidir.

3. “MOLGA” PROGRAM YAPISI

Kimyasal maddenin molekül yapılarını belirleyen parametrelerin deriden geçiş katsayısı ile ilişkisi doğrusal bir özellik göstermektedir. Bu sebepten dolayı parametreler arasındaki değişimlerin matematiksel modelinin çıkarılması mümkün olabilmektedir. Klasik optimizasyon tekniklerinin yanı sıra yapay zeka optimizasyon tekniklerinden genetik algoritmanın uygulanması ile problem çözümü gerçekleştirilmektedir. Genetik algoritma kimyasal maddenin molekül yapılarının optimize edilmesinde ve özellikle deriden geçiş katsayılarının karşılaştırılmasında ve en uygununun bulunmasında yeni bir yaklaşım getirmektedir.

Genetik algoritma deriden en iyi geçişi sağlayan kimyasal maddelerin tespitinde rasgele belirlenen 100 kimyasal madde üzerinden değerlendirilmiş ve görsel programlama dili olan Delphi 7.0 ile MOLGA programı oluşturulmuştur. Tasarımda, başlangıç popülasyonunun genişliği kullanıcıya bırakılmış ve esnek tutulmuştur. Başlangıç popülasyonunu oluşturan kromozomların her bir geni, molekülü oluşturan parametreleri temsil etmektedir. Kromozomların kodlanma işleminde 10 tabanına göre kodlama biçimi seçilmiştir. Bir sonraki nesile seçim işleminde rulet çemberi ve en kötü bireylerin terine en iyilerini yaz

seçim metodu kullanılmıştır. İterasyonlarda seçim ve çaprazlama operatörleri program içerisinde esnek bırakılmıştır.

Molga program menüleri

Çalıştır: Programın çalıştırıldığı penceredir. Uygun çözümler, uygun kromozomlar ve seçilen sonuç bu menüde yer almaktadır. Uygun çözümlerde, kullanıcının belirlediği hassasiyet aralığına göre genetik algoritma sonuçları gösterilmektedir. İterasyon sayısı, alt ve üst sınır, kromozom sayısı, gen sayısını ve genlerin hangi parametreyi gösterdiğini buradan takip etmek mümkündür. Uygun kromozomlarda, uygun çözümlerde bulunan kromozomların gösterimi bulunmaktadır. Seçilen sonuçta, kullanıcı hangi deriden geçiş katsayısını seçti ise ona ait kimyasal maddenin molekül yapısını buradan görebilmektedir. Kimyasal maddenin molekülünü oluşturan her parametre belirlenebilmektedir.

Değerleri Belirle: Aranılan kimyasal maddenin kullanım amacına göre aralığının tayinini belirleyen menü. Burada kullanıcı tarafından hassasiyet aralığı alt sınır ve üst sınır olarak girilmekte ondalık sayı ile de belirlenebilmektedir. Deriden geçiş katsayısının maksimum veya minimum olması sonuçların değerlendirilmesinde bulunabilmektedir.

Veriler: Rasgele seçilen kimyasal maddelerin molekül yapısını belirleyen 9 parametre değerleri yer almaktadır. Veriler üzerinde değişiklik yapılabilmektedir. Kayıt silme, kayıt ekleme, değişikliği iptal etme ve kaydetme gibi esneklik mümkün olabilmektedir.

Ayarlar: Bu menü genetik algoritmanın özelliklerini ve operatörlerini gösteren menüdür. Başlangıç popülasyonunu oluşturan kromozomların sayısı, kromozom üzerindeki genlerin sayıları bu menü de yer almaktadır. Yine yeni bireylerin oluşturulmasında hangi atanın seçileceğine karar veren seçim operatörleri burada belirlenmektedir. Çaprazlama operatör seçimi, genetik algoritmanın işleyişinde önemli olan uyum fonksiyonu de ayarlardan belirlenmektedir.

Fonksiyon: Uyum değerlerinin belirlenmesinde kullanılacak regresyon denklemi yer almaktadır. Kullanıcıya esnek bırakılmıştır ve değiştirilebilmektedir.

Geçmiş: Seçilen her aralığın sonucunu kaydedilip ve buradan takip edilebilmektedir.

MOLGA programı ayrıca deriden geçiş katsayıları ile parametreler arasındaki ilişkin değerlendirilmesine imkan tanımaktadır. Bileşen sayıları grafiği, bileşen değerleri grafiği ve seçilen çözümün grafiği program içerisinde yer almaktadır.

4. MOLGA PROGRAMI İLE (DURDURMA KRİTERİNİN) İTERASYON SAYILARININ DEĞERLENDİRİLMESİ

Bu programı oluşturmadaki amaç, yapay zekanın optimizasyon tekniklerinden genetik algoritma kullanılarak, özellikle ilaç sektörü, kozmetik sektörü ve tarım sektörü olmak üzere benzer sektörlerde kullanılan kimyasal maddeleri tespit etmek ve bunların hangi deriden geçiş değerine sahip olduğunu bulmaktır. Kimyasal maddelerin deriden geçiş değerlerinin bilinmesi sektörlerde kullanıldıkları ürüne göre farklılık göstermektedir. MOLGA programının temel amacı bu olmakla birlikte genetik algoritmanın işleyiş açısından değerlendirildiğinde, program her çalıştığında iterasyon sayısı nasıl değişmektedir ve hangi koşullara bağlı olmaktadır.

İterasyon sayısının belirlenmesi için 10 farklı uygulama ve 4 tercihle uygulama yapılmıştır. Hem aynı aralık için farklı genetik algoritma operatörleri ile hem de farklı aralıklarda aynı operatörler kullanılarak iterasyon sayısı belirlenmiştir.

MOLGA programında sunulan genetik operatörlerden iki nokta çaprazlama kromozom üzerindeki 1.genden ve kromozom üzerindeki 5. genden yapılacaktır. Seçim için kullanılacak en kötü bireylerin yerine en iyilerini yaz operatörü için 5 kötü birey tercih edilmiştir. Böylece uyum değerleri en kötü olan 5 birey yeni oluşturulacak popülasyon içerisinde yer almayacaktır. Çizelge 1’de uygulama için kullanılacak operatörler yer almaktadır.

Çizelge 1. MOLGA programının sunduğu GA operatör alternatifleri

Aynı aralık farklı GA operatörleri	I.Tercih: Rasgele tek noktadan
	Rulet tekerleği
	II.Tercih: Rasgele tek noktadan
	En kötü bireyler yerine en iyilerini yaz
	III.Tercih: İki nokta çaprazlama
	Rulet tekerleği
	IV.Tercih: İki nokta çaprazlama
	En kötü bireyler yerine en iyilerini yaz

Çizelge 2’de uygulama için girilen 10 farklı aralık verilmektedir.Girilen bu aralıklar rasgele belirlenmiştir. Burada $\text{Log}k_p$ kimyasal maddelerin deriden geçiş katsayısını ifade etmektedir. İterasyon sayılarının değerlendirilmesine yönelik yapılan çalışmada aralıklar negatif, pozitif ve ondalık sayılar olarak girilmiştir.

Çizelge 2. Uygulama için girilen deriden geçiş aralıkları

Farklı aralık aynı GA operatörleri	1.Uygulama: $7 \leq \text{Log}k_p \leq 8$
	2.Uygulama: $1 \leq \text{Log}k_p \leq 2$
	3.Uygulama: $0,5 \leq \text{Log}k_p \leq 1$
	4.Uygulama: $-2 \leq \text{Log}k_p \leq -2,5$
	5.Uygulama: $-1 \leq \text{Log}k_p \leq 2$
	6.Uygulama: $-1 \leq \text{Log}k_p \leq -1,5$
	7.Uygulama: $0,5 \leq \text{Log}k_p \leq 0$
	8.Uygulama: $5 \leq \text{Log}k_p \leq 5,5$
	9.Uygulama: $5,5 \leq \text{Log}k_p \leq 7$
	10.Uygulama: $-4 \leq \text{Log}k_p \leq -5$

5. SONUÇ VE DEĞERLENDİRME

Genetik algoritma iteratif ve stokastik bir süreçtir. Algoritma bu iki temel üzerinde işletilmektedir. Uygulama problemlerinde amaç(uygunluk) fonksiyonunun doğrultusunda en iyi veya istenilen çözüme ulaşıncaya kadar genetik operatörler uygulanır.

Hazırlanan MOLGA programında amaç, rasgele seçilen kimyasal maddelerle başlangıç popülasyonu oluşturulmuş ve istenen deriden geçiş katsayısına göre belirlenen yeni kimyasal yapının bulunması hedeflenmiştir. Genetik algoritmanın tekrarlanan bir süreç olmasından dolayı bu uygulamada rasgele seçilen aralıklarla ve program içerisinde sunulan

genetik operatörler yardımıyla programın durdurma kriterini belirleyen iterasyon sayıları test edilmiştir. Bu çalışma ile ilgili iterasyon sayıları Çizelge 3’ de verilmiştir.

Çizelge 3. Uygulamada bulunan iterasyon sayıları

Farklı aralık aynı GA operatörleri için iterasyon sayısı	Aynı aralık farklı GA operatörleri için iterasyon sayısı			
	I.Tercih	II.Tercih	III.Tercih	IV.Tercih
1.Uygulama	18	-	100	-
2.Uygulama	1	1	1	1
3.Uygulama	1	1	1	1
4.Uygulama	100	-	100	-
5.Uygulama	1	1	1	1
6.Uygulama	10	-	4	-
7.Uygulama	1	1	1	1
8.Uygulama	1	1	1	1
9.Uygulama	1	1	1	1
10.Uygulama	23	-	100	-

Çizelge incelendiğinde, istenilen deriden geçiş değeri aralığına göre belirli bir değerin üzerinde ve negatif değerler için iterasyon sayıları artmaktadır. Negatif ve pozitif yüksek değerler için iterasyon II. ve IV. tercihlerde işlememektedir. Özellikle genetik operatörlerden II. tercihte yer alan, rasgele tek nokta çaprazlama ve en kötü bireylerin yerine en iyilerini yaz eleme operatörü ile IV. tercihte bulunan iki nokta çaprazlama ve en kötü bireylerin yerine en iyilerini yaz eleme operatörü birlikte kullanıldığında bazı aralıklar için sonuç vermemektedir. Yani iterasyon gerçekleşmemektedir.

Genel olarak literatürde yer alan deriden geçiş değerlerinin bulunduğu aralıklar, program içerisinde tek itersyonda bulunabilmektedir. Sadece genetik algoritma operatörleri değiştirildiğinde iterasyon sayısında farklılık görülmektedir.

KAYNAKLAR

- Hayes, M.H. (1998). “Hidden Markov Models for Face Recognition”, *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, USA, S.2721-2724.
- Holland, J.H. (1992). “*Adaptation in Naturel and Artificial System an Introductory Analysis with Applications to Biology,Control and Artificial Intelligence*”, Second Edition, Cambridge Mass:MIT Pres, S.211.
- Lan Fang, H. (1994). Genetic Algorithms in Timetabling and Scheduling, PhD., University of Edinburg, S.30.
- Karaboğa, D. (2004). “*Yapay Zeka Optimizasyon Algoritmaları*”, Atlas Yayınevi, Yayın No:38, İstanbul, S.81.
- Yeniay, Ö. (1999). Taguchi Deney Tasarımı Problemlerine Genetik Algoritma Yaklaşımı,(Yayınlanmamış Doktora Tezi), Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü