

GENETİK ALGORİTMA VE UYGULAMA ALANLARI

Mustafa KURT, Cumali SEMETAY**

**M.Ü. Teknik Eğitim Fakültesi Makina Bölümü*

Bu çalışmada, geleneksel yöntemlerle çözümü zor veya imkansız olan problemlerin çözümünde kullanılan yöntemlerden biri olan Genetik Algoritma (GA) tekniği incelenmiştir. GA tekniğinin ana prensipleri, uygulama alanları örneklerle anlatılmıştır

Anahtar sözcükler: *Genetik algoritma, optimizasyon, robot kontrol*

In this study, Genetic Algorithm (GA) which is an approach to solve hard problems that can not be solved in a traditional way is researched. The main principles of GA and its application areas have been outlined by examples.

Keywords: *Genetic algorithm, optimization, robotic control*

GİRİŞ

Genetik algoritmalar yapay zekanın gittikçe genişleyen bir kolu olan evrimsel hesaplama tekniğinin bir parçasını oluşturmaktadır. Adından da anlaşıldığı üzere, evrimsel hesaplama tekniğinin bir parçası olan genetik algoritma Darwin'in evrim teorisinden esinlenerek oluşturulmuştur. Herhangi bir problemin genetik algoritma ile çözümü, problemi sanal olarak evrimden geçirmek suretiyle yapılmaktadır.

Evrimsel hesaplama ilk olarak 1960'larda I.Rechenberg tarafından "Evrimsel Stratejileri (Evolutionstrategie)" isimli eserinde tanıtılmıştır. Onun fikri daha sonra başka araştırmacıların da ilgisini çekmiş ve geliştirilmiştir. John Holland evrim sürecinin bir bilgisayar yardımıyla kullanılarak, bilgisayara anlayamadığı çözüm yöntemlerinin öğretilebileceğini düşündü. Genetik Algoritma (GA) böylece John Holland tarafından bu düşüncenin bir sonucu olarak bulundu. Onun öğrencileri ve arkadaşları tarafından geliştirildi ve bu sayede Holland'ın kitabı "Doğal ve Yapay Sistemlerde Adaptasyon (Adaptation in Natural and Artificial Systems)" 1975 yılında yayınlandı.

1992 yılında John Koza genetik algoritmayı kullanarak çeşitli görevleri yerine getiren programlar geliştirdi. Bu metoda Genetik Programlama adını verdi.

Genetik algoritma geleneksel yöntemlerle çözümü zor veya imkansız olan problemlerin çözümünde kullanılmaktadır. Çok genel anlamda genetik algoritmanın üç uygulama alanı bulunmaktadır. Bunlar deneysel çalışmalarda optimizasyon, pratik endüstriyel uygulamalar ve sınıflandırma sistemleridir.

Mühendislik problemlerinde optimizasyon amaçlı olarak kullanılmaya başlanmıştır. Özellikle mekanizma tasarımında çok iyi sonuçlar verdiği bilinmektedir[3]. Bunlardan başka otomatik programlama, öğrenme kabiliyetli makineler, ekonomi, ekoloji, planlama, üretim hattı yerleşimi gibi alanlarda da uygulanmaktadır. Ayrıca dijital resim işleme tekniğinde de çokça uygulama alanı bulmuştur[1,5].

Bu problemlerin hemen hemen hepsi çok geniş bir çözüm havzasının taranmasını gerektirmektedir. Bu çözüm havzasının geleneksel yöntemlerle taranması çok uzun sürmekte, genetik algoritmayla ise kısa bir sürede kabul edilebilir bir sonuç alınabilmektedir.

GENETİK ALGORİTMA TEKNİĞİ

Algoritma ilk olarak popülasyon diye tabir edilen bir çözüm(kromozomlarla ifade edilir) seti ile başlatılır. Bir popülasyondan alınan sonuçlar bir öncekinden daha iyi olacağı beklenen yeni bir popülasyon oluşturmak için kullanılır. Yeni popülasyon oluşturulması için seçilen çözümler uyumluluklarına göre seçilir. Çünkü uyumlu olanların daha iyi sonuçlar üretmesi olasıdır. Bu istenen çözüm sağlanıncaya kadar devam ettirilir.

Genetik Algoritmanın Aşamaları:

1. Başlangıç: n adet kromozom içeren popülasyonun oluşturulması (problemin uygun bir çözümü)
2. Uyumluluk: her x kromozomu için uyumluluğun $f(x)$ değerlendirilmesi,
3. Yeni popülasyon: Yeni popülasyon oluşuncaya kadar aşağıdaki adımların tekrar edilmesi,
 1. Seçim: İki ebeveyn kromozomun uyumluluğuna göre seçimi (daha iyi uyum seçilme şansını artırır.),

2. Çaprazlama: Yeni bir fert oluşturmak için ebeveynlerin bir çaprazlama olasılığına göre çaprazlanması. Eğer çaprazlama yapılmazsa yeni fert anne veya babanın kopyası olacaktır.

3. Mutasyon: Yeni ferdin mutasyon olasılığına göre kromozom içindeki konumu (lokus) değiştirilir.

4. Ekleme: Yeni bireyin yeni popülasyona eklenmesi.

4. Değiştirme: Algoritmanın yeniden çalıştırılmasında oluşan yeni popülasyonun kullanılması,

5. Test: Eğer sonuç tatmin ediyorsa algoritmanın sona erdirilmesi ve son popülasyonun çözüm olarak sunulması.

6. Döngü: 2. adıma geri dönülmesi.

Görüldüğü üzere genetik algoritmanın yapısı oldukça geneldir ve herhangi bir probleme uygulanabilir. Kromozomların tanımlanması genellikle ikili düzendeki sayılarla yapılır. Çaprazlama işlemi için kullanılan bireyler iyi bireylerden seçilir.

GA kullanılarak bir problem çözülecekse algoritmanın ne zaman sonlanacağına kullanıcı karar vermektedir. GA'nın belli bir sonlanma kriteri yoktur. Sonucun yeterince iyi olması veya yakınsamanın sağlanması algoritmanın durması için kriter olarak kullanılabilir[8].

GA'da Kullanılan Operatörler

Genetik algoritmanın en önemli kısımları çaprazlama ve mutasyon işlemleridir. Bu işlemler bir olasılık değeri ile ve genelde rasgele olarak uygulanır. Bu şekilde iyi sonuç alınabilmektedir.

Bir kromozomun ikili sayılarla temsil edilmesi:

Kromozom 1 1101100100110110

Kromozom2 1101111000011110

Kromozom temsil ettiği çözümlerle ilgili bilgi içermelidir. Her kromozom ikili (binary) bir diziden oluşur. Bu dizi içindeki bit adı verilen her bir sayı çözümün

bir karakteristiğini temsil edebilir veya bir dizi bütünüyle bir sayıya işaret edebilir.

Kromozomu ikili düzendeki sayılar dizisiyle ifade etmek çok tercih edilen bir temsil şeklidir ancak bunun yerine tamsayı veya reel sayılar da kullanılabilir. İkili düzenin tercih edilmesinin sebebi basit olması ve bilgisayar tarafından daha kolay ve hızlı bir biçimde işlenebilmesidir.

Üreme:

Üreme işlemi belli bir seçme kriterine göre bireylerin seçilip yeni kuşağın oluşturulması işlemidir. Seçme kriterleri uyumluluğu esas alarak birbiriyile uyumlu olan bireyleri seçer. Daha sonra çaprazlama ve mutasyon uygulanacak olan bireylerden daha uyumlu yeni bireylerin ortaya çıkması olasıdır. Bireylerin tamamı uyumluluğa göre seçilebilir veya bir kısmı rasgele seçilerek yeni kuşağa aktarılabilir.

Çaprazlama:

Kromozomların nasıl temsil edileceğine karar verildikten sonra çaprazlama yapılabilir. Çaprazlama ebeveynlerden bazı genleri alarak yeni bireyler oluşturma işlemidir.

Kromozom 1 11011 / 00100110110

Kromozom 2 11011 / 11000011110

Birey 1 11011 / 11000011110

Birey 2 11011 / 00100110110

Çaprazlama yapılacak konum rasgele seçilir (|). Oluşan yeni birey ebeveynlerin bazı özelliklerini almış ve bir bakıma ikisinin kopyası olmuştur. Çaprazlama işlemi başka şekillerde de yapılabilir. Mesela birden fazla çaprazlama noktası seçilebilir. Daha iyi performans almak amacıyla değişik çaprazlamalar kullanılabilir.

Mutasyon:

Çaprazlama gerçekteştikten sonra mutasyon gerçekteştirilir. Mutasyon oluřan yeni çözümlerin önceki çözümlü kopyalamasını önlemek ve sonuca daha hızlı ulaşmak amacıyla yapılır. Mutasyon oluřan yeni bireyin bir bitini (eđer ikili düzende ifade edilmiş ise) rasgele deęiřtirir.

Orjinal Birey 1 1101111000011110

Orjinal Birey 2 1101100100110110

Deęiřmiş Birey 1 1100111000011110

Deęiřmiş Birey 2 1101101100110110

Elitizm:

Üreme, çaprazlama ve mutasyon işlemleri sonrasında kuřakta bulunan en iyi uyumluluęa sahip birey sonraki kuřaęa aktarılamayabilir. Bunu önlemek için bu işlemlerden sonra oluřan yeni kuřaęa bir önceki kuřaęın en iyi (elit) bireyi, yeni kuřaktaki herhangi bir birey ile deęiřtirilir. Buna elitizm adı verilir[1,9,10].

GA Parametreleri

Çaprazlama ve Mutasyon olasılıęı:

GA teknięinin çaprazlama olasılıęı ve mutasyon olasılıęı olmak üzere iki basit parametresi vardır.

Çaprazlama olasılıęı çaprazlamanın hangi sıklıkta yapılacaęını belirtir. Eđer hiç çaprazlama yapılmaz ise (çaprazlama olasılıęı %0) yeni bireyler eski bireylerin aynısı olur ama bu yeni kuřaęın eskisiyle aynı olacaęı anlamına gelmez. Eđer bu oran %100 olursa yeni bireyler tamamıyla çaprazlama ile elde edilir. Çaprazlama eski bireylerden iyi taraflar alınarak elde edilen yeni bireylerin daha iyi olması umuduyla yapılır.

Mutasyon olasılıęı ise mutasyonun hangi sıklıkta yapılacaęını belirtir. Mutasyon olmaz ise yeni birey çaprazlama veya kopyalama sonrasında olduęu gibi kalır. Eđer mutasyon olur ise yeni bireyin bir kısmı deęiřtirilmiş olur. Eđer bu oran %100 olursa kuřak içindeki bireyler tamamen deęiřir, %0 olursa hiç deęiřmeden kalır.

Diğer parametreler:

GA tekniđi başka parametreler de içerir. Bunların en önemlilerinden birisi de populasyon büyüklüğüdür.

Bu parametre populasyon içinde (yalnızca bir kuşakta) kaç adet kromozom yani birey olduğunu söyler. Eğer kromozom sayısı az olursa GA çözüm aranan uzayın ancak bir kısmını gezebilir ve çaprazlama için fazla bir seçeneđi yoktur. Kromozom sayısı çok fazla olursa GA çok yavaş çalışır. Araştırmalar belli bir noktadan sonra populasyon sayısını artırmanın bir yararı olmadığını göstermiştir.

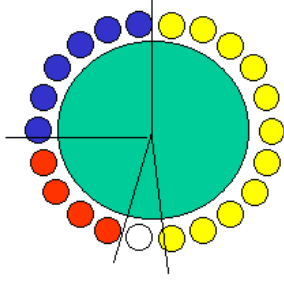
Yeni bireyler uyumluluđa göre veya rasgele olarak seçilebilir. Yeni bireylerin tamamen rasgele seçilme durumunda yakınsama zorlaşabilir. Tüm bireyler uyumluluđa göre seçildiğinde ise yeni kuşak içinde bölgesel yakınsamalar olabilir. Bu sorunların üstesinden gelmek için belli bir oranda uyumluluk seçimi belli bir oranda da rasgele seçim yapılabilir. Bu oran Kuşak Farkı (Generation Gap) ile ifade edilir. Kuşak farkı %100 olduğunda yeni bireylerin tamamı uyumluluđa göre seçilir.

Seçim:

Ebeveynleri oluşturmak üzere bazı bireylerin seçilmesi gerekir. Teoriye göre iyi olan bireyler yaşamını sürdürmeli ve bu bireylerden yeni bireyler oluşmalıdır. Bu seçim çeşitli kriterlere göre yapılabilir. Rulet seçimi, Boltzman seçimi, turnuva seçimi, sıralı seçim bunlardan bazılarıdır.

Rulet seçiminde kromozomlar uyumluluk fonksiyonuna göre bir rulet etrafına gruplanır. Uyumluluk fonksiyonu herhangi bir kritere uyan bireylerin seçilmesi için kullanılır. Bu rulet üzerinden rasgele bir birey seçilir. Daha büyük alana sahip bireyin seçilme şansı daha fazla olacaktır [8, 9, 10].

Rulet seçimi eđer uyumluluk çok fazla deđişiyorsa sorun çıkartabilir. Örneđin en iyi kromozomun uyumluluđu %90 ise diđer kromozomların seçilme şansı azalacaktır. Bunu önlemek için sıralı seçim kullanılabilir. Sıralı seçimde en kötü uyumlulukta olan kromozoma 1 deđer sonraki 2 deđer verilir ve böylelikle seçilmede bunlara öncelik tanınmış olur. Bu şekilde onların da seçilme şansı artar fakat bu çözümün daha geç yakınsamasına neden olabilir.



Şekil 1. Rulet Seçimi: Seçim yapmak için top atıldığında sarı alanlarda durma olasılığı daha yüksektir.

Örnek Çözüm:

Aşağıda rasgele seçilmiş 8 bitlik 4 adet birey görülmektedir. Uyumluluk kriteri her dizinin barındırdığı 1 adedi olsun (a bireyinde 2 adet 1 bulunmaktadır dolayısıyla uyumluluğu 2'dir);

a 00000110 2

b 11101110 6

c 00100000 1

d 00110100 3

Bu durumda aradığımız değer 8 uyumluluğa sahip bir değer olsun. Bu 4 adet dizi içinden ikişer çift seçerek sonraki kuşak oluşturulacak ve bunlara çaprazlama ve mutasyon uygulanacaktır. b,d ile b,c çiftleri seçilsin. Çaprazlama bu çiftler içinde çaprazlama olasılığı oranında gerçekleşecektir. b,d çiftinde çaprazlama olduğunu varsayalım. Bu durumda $e=10110100$ ve $f=01101110$ şeklinde yeni bireyler oluşacaktır. Burada çaprazlama ilk bit üzerinden gerçekleşmiştir. Diğer çift üzerinde çaprazlama yapılmasın. Mutasyon ise mutasyon olasılığı oranında etki edecektir. Burada e ile b bireyelerine etkimiştir (e'ye 6. bit, b'ye ilk bit üzerinden). Mutasyona uğrayan bitler değişir ve son durum aşağıdaki gibi olur;

e' 10110000 3

f 01101110 5

c 00100000 1

b' 01101110 5

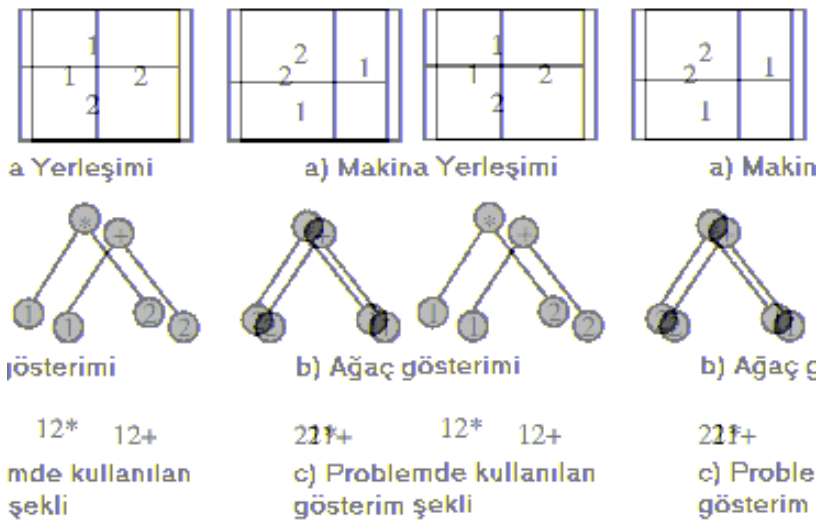
Görüldüğü gibi en yüksek uyumluluğa sahip olan b bireyinin kaybolmasına karşılık ortalama uyumluluk artmıştır. Bu işlem birkaç kez tekrar edildiğinde GA mükemmel diziyi yani 1111111 dizisini bulacaktır.

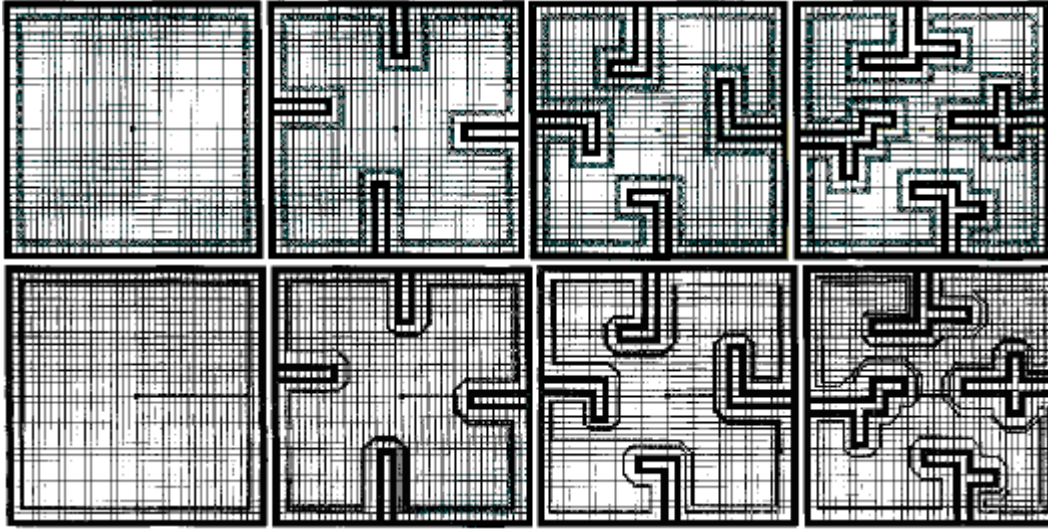
GA Kullanılarak Yapılmış Bazı Çalışmalar:

Fabrika Yerleşiminin Optimizasyonu

Makina yerleşiminin optimizasyonu imalat sisteminin malzeme transportunu minimize etmek ve dolayısıyla maliyeti düşürmek için yapılır. Transport edilen malzemenin miktarını düşürmenin önemine rağmen geleneksel metotlar ile malzeme transportu esnasında geçen zaman belirlenememektedir. Günümüzdeki imalat sistemleri kısa sürelerde çevrimlerini tamamlayarak ürün çıkartabilmektedir. Dolayısıyla aralardaki malzeme hareketinin üretilebilirliğe etkisi çok fazla olmaktadır. Bu çalışmada [12] yerleşim planı, sistemin dinamik karakteristikleri ve çalışma koşulları gözönünde bulundurularak, çevrim zamanı ve üretilebilirlik gibi sistem performansını karakterize eden değişkenlere bağlı olarak optimize edilmiştir.

Populasyonlar GA ile analiz edilmesi kolay diziler halinde elde edilmiştir. Ayrıca simülasyon yardımıyla mevcut çözüm değerlendirilmiş ve GA ile tekrar optimizasyon sağlanmıştır. Deneysel çalışmalar sonucunda GA'nın iyi sonuçlar verdiği gözlenmiştir.





Şekil 2. Fabrika Yerleşimi Probleminde Kullanılan Gösterim Şekilleri.

Duvarı Takip Eden Robot

Bu robotun amacı bir duvara rastlayınca kadar boş alanda hareket etmek ve duvara rastlayınca da belli bir mesafeden duvarı takip ederek ilerlemektir. Robotun başarılı sayılabilmesi için bulunduğu ortamın çevresini en azından bir kez çok yakın veya çok uzak kalmaksızın takip etmesi gerekir [18].

Genetik programlama ile elde edilen gezme algoritmasını test etmek için iki boyutlu konfigürasyonlar kullanılmıştır. Bu test ortamı 4 adet tamamen kapalı odadan oluşmaktadır ve her odanın iç çevresi farklı karmaşıklıktadır. Genetik programlama ile elde edilen popülasyonun her bireyi her bir oda için denenip duvar-takip işlemini nasıl gerçekleştirdiği test edilmiştir.

Geliştirilen robot, merkeziyle 45° açılı 8 adet uzaklık ölçme sensörü içermektedir. Genetik programlama ile öngörülen takip çizgisinin ne kadarının takip edilemediğine bakılmakta ve her bir bireyin (çözümün) uyumluluğu böylece saptanmaktadır. Takip edilemeyen miktarı veren uyumluluk fonksiyonu minimize edildiğinde GA en uygun çözümü vermiş olacaktır.

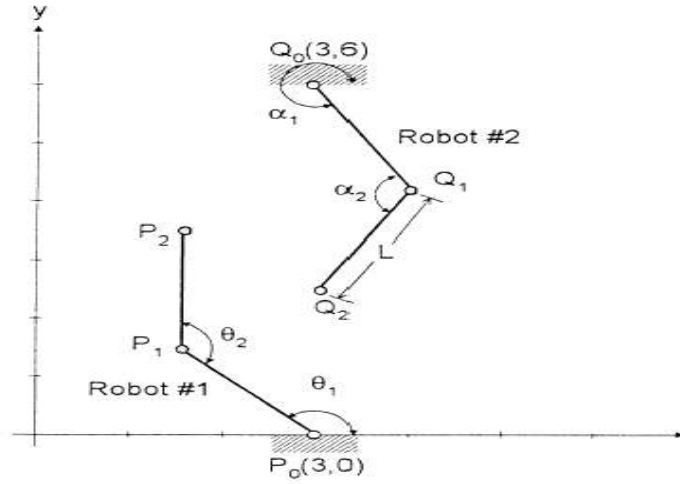
GP kullanılarak %100 çözüm elde edilmese bile istenen hareketi elde etmeye olanak sağlayan çözümler elde edilmiştir. Bu çalışmayla benzer komplekslikteki çevrelerde de iyi sonuçlar alınabileceği anlaşılmıştır.

Şekil 3. Duvar Takibi Yapılacak Odalar ve İdeal Takip Yolları.

Şekil 4. GA Kullanılarak Elde Edilen Takip Yolları.

Çok Kollu Robotların Çarpışmasız Hareketi

Bu çalışmada [17] birden fazla koldan oluşan robot sisteminin sabit engellere çarpmadan hareketinin yanı sıra hareketli çevre ve engellerle de çarpışmadan hareket etmesi sağlanmaya çalışılmıştır. Yörünge denklemleri dizilere çevrilmiş ve böylece GA çarpışmasız minimum hareket yolunu hesaplamak için kullanılmıştır.



Şekil 5. Problemede Çarpışmasız Hareketi İncelenen İki Linkli Düzlemsel İki Robot

GA tekniğinde genelde ikili sayı sisteminden oluşan diziler kullanılsa da artık bu standardın dışında bir gösterim daha sık kullanılmaya başlanmıştır. Bu çalışmada kayan noktalı vektör gösterimi kullanılmıştır. Yörüngeler direk olarak kayan noktalı vektörler şeklinde kodlanmıştır. Bu şekilde kodlamanın avantajı ikili sisteme göre çok daha hassas olmasıdır.

Uyumluluk fonksiyonu yörünge uzunlukları, dengeli dağılım ve çarpışma durumu gözönünde bulundurularak elde edilmiştir. Başlangıç popülasyonu ise Bezier eğrileri ile oluşturulmuştur. Bezier eğrisini tanımlamak için 4 noktaya ihtiyaç vardır. Elde edilecek eğri 1. ve 4. noktalardan geçer, 2. ve 3. noktalar ise kontrol noktalarıdır ve eğrinin şeklini belirler. İlk popülasyon bu şekilde elde edildikten sonra GA ile daha sonraki yörüngeler uyumluluk fonksiyonu gözönünde bulundurularak hesaplanır.

İki linkli düzlemsel iki robotun haricinde 3 serbestlik dereceli Puma benzeri iki robot için de benzer şekilde formülasyon yapılmıştır. Bu ortam öncekinden farklı olarak statik engeller de içermektedir.

Klasik GA (ikili diziler) kullanıldığında yüzlerce kuşak beklemek gerekirken bu çalışma sonucunda 30-40 kuşak sonra iyi sonuçlar alınabildiği görülmüştür. Bu çalışma sadece sonuçlarıyla değil, GA tekniğine yeni operatörler ve yaklaşımlar getirmesi açısından da önemlidir.

Robot Eli

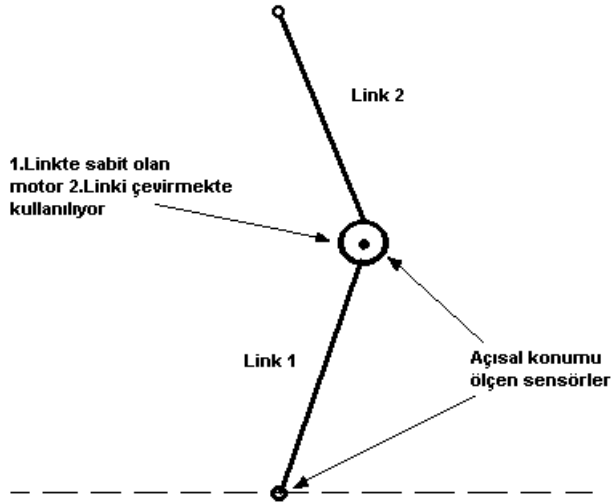
Bu çalışmada [13] insan eli benzeri beş parmaklı bir robot elinin bir nesneyi kavraması için gereken hareketler incelenmiş ve bu karmaşık problemin çözümünde GA kullanılmıştır. Bu problem çarpışmasız hareket yörüngesi saptamaya benzemektedir. Her bir parmak diğeri için çarpıması gereken bir engeldir. Uyumluluk her bir parmağın kontak noktasına (nesneye dokunduğu nokta) olan uzaklığı, stabilite, manipulasyon ve çarpışmasız hareket gözönünde bulundurularak hesaplanmaktadır. Klasik GA kullanılmasına rağmen çalışma uzayı oldukça geniş olduğundan (2^{230}) GA'nın biraz daha seçici davranması sağlanmıştır. Ayrıca GA operatörleri de bu sebepten dolayı biraz modifiye edilerek kullanılmıştır.

Problem ön kavrama (Preshaping) ve tutma (Regrasping) olmak üzere iki kısımda incelenmiştir. Preshaping' de elde edilen optimum populasyon, Regraspingte başlangıç populasyonu olarak kullanılmıştır.

Sonuçta GA'nın probleme en uygun araç olduğu görülmüştür. Sonuçlar verimli olmasına rağmen hesaplama süresi bir eksiklik olarak görülebilir çünkü hesaplama uzun zaman almıştır. Uzun hesaplama süresinin bir nedeni de eklem açılarının vektörel olarak gösterimidir.

Akrobot

Akrobot [11] adından da anlaşıldığı gibi bir akrobatın hareketini simule etmektedir. Aslında iki linkli düzlemsel bir robottur. Elleriyle barfikse tutunup başaşağı dengede duran bir akrobatın yaptığı hareket acrobota yaptırılmaya çalışılmıştır. Kullanılan tek motor birinci link ile sabittir ve 2. linki hareket ettirmektedir.



Şekil 6. Akrobot.

Dengeleme için LQR ve Fuzzy kullanılmıştır. Acrobotun başaşağı konuma gelmesi motorun sağa-sola salınma hareketi yaparak sallanmasıyla mümkündür. Bu sallanma hareketinin kontrolü ise PD ve Fuzzy ile yapılmıştır. GA'nın bu çalışmadaki rolü sallanma hareketi ve dengeleme kontrolü için kullanılan PD, Fuzzy ve LQR kontrolcülerinin parametre optimizasyonudur.

Bu çalışma sonucunda GA'nın dengeleme kontrolcülerinin performansını belli başlangıç koşullarında arttırdığı gözlenmiştir. LQR kontrolcüsünde bu fark fazla belirgin değildir. Fuzzy kontrolcüsünün ise bazı başlangıç koşullarında stabil kalmadığı görülmüştür. Bu Fuzzy kontrolcüsünün başlangıç koşullarına LQR'dan daha duyarlı olmasından kaynaklanmaktadır. Sallanma hareketinin kontrolcü parametrelerinin GA ile optimizasyonu ise daha iyi sonuçlar vermiştir. Çünkü sallanma hareketinin kontrolünde kullanılan parametre kombinasyonları çok daha fazladır.

Bu çalışmaların dışında robotik alanında birçok uygulamada GA yardımcı araç olarak kullanılmıştır. Bu çalışmalardan birinde robotun temasta bulunduğu yerlerdeki sürtünmeyi telafi etmek için sürtünmeyi öğrenme algoritması olarak kullanılan GA, ölçüm hataları olması durumunda bile iyi sonuçlar vermiştir.

Başka bir çalışmada robotun hareket yörüngesi kartezyen koordinatlarda elde edilmeye çalışılmıştır. Simulasyon denemeleri GA'nın bu çalışmada da iyi performans sunduğunu göstermiştir.

GA kullanılarak yapılan diğer bir çalışma ise kontrol alanındadır. Nonlineer sürtünmeli esnek sürücü sisteminin bulanık mantık ile kontrolünde GA,

parametrelerin optimizasyonunda kullanılmış ve düşük hızlardaki sürtünme etkilerini azaltarak performansı arttırmada önemli rol oynamıştır.

KAYNAKÇA

1. Mansfield, R.A.: *Genetic Algorithms*, University of Wales College of Cardiff, 1990.
2. Kurt, M., Konuralp, M.S., Tektaş, M.: *Burulmaya Zorlanan Düzgün Kesitli Millerin Tasarımında Genetik Algoritmanın Kullanılması*, M.Ü. Fen Bilimleri Dergisi, Sayı 15,1999
3. Konuralp, M.S., Işık A.H., Taçgın, E.: *Salınan Kol-Kızak Mekanizmaların Kinematik Sentezini Genetik Algoritma Tekniğini Kullanarak Gerçekleştiren Bir Prototip Yazılım*, 8. Uluslararası Makina Tasarım ve İmalat Kongresi, 9-11 Eylül 1998.
4. Riccardo, P., Langdon, William B.: *Schema Theory for Genetik Programming With One-Point Crossover and Point Mutation*, Evolutionary Computation, Fall 98, Vol.6, Issue 3.
5. Gizolme, O., Thollon, F.: *Shape Optimization of Synchronous Machine Rotor*, International Journal of Applied Electromagnetics and Mechanics, Jul98, Vol 9, Issue 3.
6. Reeves, Colin R., Yamada, Takeshi: *Genetic Algorithms, Path Relinking, and the Flowshop Sequencing Problem*, Evolutionary Computation, Spring 98, Vol 6, Issue 1.
7. Deb, Kalyanmony: *Multi-objective Genetic Algorithms, Evolutionary computation*, Fall99, Vol 7, Issue 3.
8. <http://lancet.mit.edu/galib-2.4/overview.html>
9. <http://www.man.ac.uk/hpctec/courses/biocomputing>
10. <http://osiris.sunderland.ac.uk/cbowww/AI>
11. Scott, C. Brown and Kevin M. Passino: *Intelligent Control for an Acrobot*, Journal of Intelligent and Robotic Systems, Vol 18, 209-248, 1997.
12. Farhad, Azadivar and John (Jian), Wang: *Facility layout optimization using simulation and genetic algorithms*, International Journal of Production Research, 2000, Vol.38, No.17, 4369-4383.

13. **I. Erkmen, A.M. Erkmen, H. Günver:** *Robot Hand Preshaping and Regrasping Using Genetic Algorithms*, The International Journal of Robotics Research, Vol.19, No.9, September 2000, 857-874.
14. **Liaw and Huang:** *Robot Contact Friction Compensation by GAs*, Journal of Intelligent and Robotic Systems, Vol 23.
15. **Nearchou and Aspragathos:** *Cartesian Trajectory Generation of Redundant Robots*, Journal of Intelligent and Robotic Systems, Vol 23.
16. **L.-C.Lin and Y-J.Lin:** *GA Based Fuzzy-Enhanced Control Schema for Flexible Drive Systems With Nonlinear Friction*, Journal of Intelligent and Robotic Systems, Vol 23.
17. **A.S.Rana and A.M.S. Zalzala:** *Collision-Free Motion Planning of Multiarm Robots Using Evolutionary Algorithms*, Proc. Instn. Mech. Engrs 1997, Vol.211, Part 1, 373-384.
18. **A.Dain,Robert:** *Developing Mobile Robot Wall-Following Algorithms Using Genetic Programing*, Applied Intelligence 8, 33-41, 1998