

**T.C**

**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

BSM 310 – YAPAY ZEKA

**Grup üyeleri:**

**B171210059 ERDEM ALTIPARMAK**

**B171210095** **RABİ GÜNÖZ**

**B171210101 BİLAL ŞENGÜL**

**B181210353 ÜMİT RECEP ÖĞÜT**

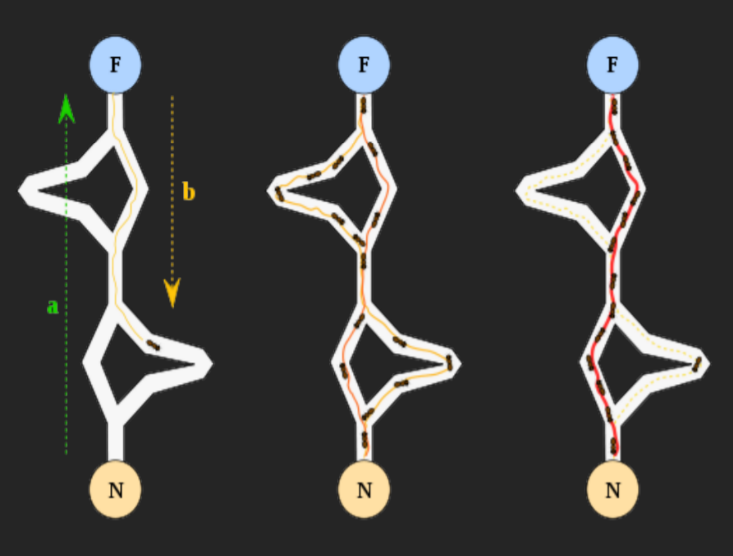
**B191210377 FURKAN GEDİK**

KARINCA KOLONİ ALGORİTMASI

GERÇEK KARINCALARDA BİYOLOJİK ALTYAPI

Karıncalar tek başlarına basit birer canlı olmalarına karşın koloni şeklinde yaşamaları onlara yetenekleri üstünde bir güç katmaktadır. Görme duyularının kötü olmasına karşın kimyasal algılayıcıları gelişmiştir.

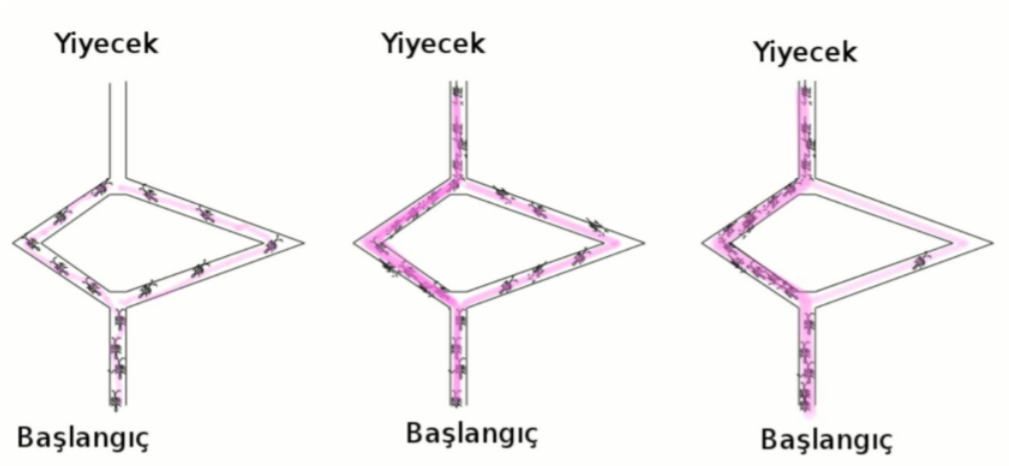
Karıncalar gündelik hayatta yiyecek ararken yollarını ve rotalarını kaybetmemek, ayrıca diğer karıncalar ile iletişim sağlamak için geçtikleri yollara “feromon” adında bir tür hormon salgısı bırakırlar. Bir karınca yiyecek bulduğunda diğer karıncalarında yiyeceğe ulaşabilmesi için dönüş yoluna da feromon salgılar. Böylece diğer karıncalar da izlerdeki feromon yoğunluğuna göre izleri takip ederek yiyeceğe ulaşabilirler. Bu sistem ile karıncalar diğer karıncaların deneyimlerini kullanarak optimal yolu ve yiyeceği bulurlar.

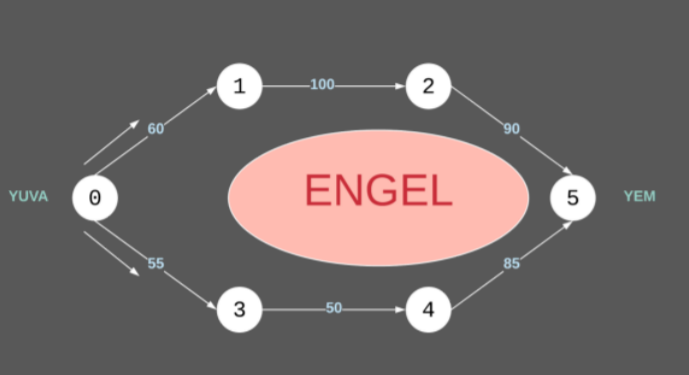


Önlerine bir engel konulduğunda feromonları takip edemediklerinden, karıncalar gidebilecekleri iki yoldan birini öncelikle rastsal olarak seçmektedirler. Kısa olan yoldan birim zamandaki geçiş daha fazla olacağından bırakılan feromon miktarı da daha fazla olur. Buna bağlı olarak, zaman içerisinde kısa olan yolu tercih eden karıncaların sayısında artış olur. Belli bir süre sonra tüm karıncalar kısa yolu tercih ederler.

Başta rastsal hareket eden karıncaların izleri kontrol ederek yüksek olasılıkla izlerin yoğun olduğu yönü takip etmesi otokatalitik bir davranış şeklidir ve karıncaların karşılıklı etkileşiminde sinerjik bir etki vardır.

Algoritma, karınca kolonilerinden esinlenerek geliştirildiğinden sisteme, karınca sistemi (KS), algoritma ise karınca kolonileri algoritması (KKA) olarak adlandırılır. Karınca kolonileri optimizasyon problemlerinde kullanılır. Karınca sistemindeki karıncalar doğal karıncalardan farklıdır. Hafızaya sahiptirler, tamamen kör değildirler ve zamanın kesikli olduğu bir çevrede yaşarlar.





ŞEKİL I

Şekil 1’deki gibi bir engel karşısında karınca kolonisinin nasıl davrandığı aşağıda adım adım verilmiştir.

1.Algoritmanın çalışmasında kesikli (discrete) zaman (t) kullanılır.

2. Her zaman aralığında her bir karınca bir birim yer değiştirir.

3. Her yer değiştirmede 1 birim feromon maddesi bırakır.

4. Başlangıçta (t=0) hiçbir yolda (kenarda) feromon maddesi yoktur.

5. Örnekte karıncalar 0 ile gösterilen yuvadan çıkarak 5 düğümündeki yeme engeli aşarak ulaşacaklardır.

6. Bir müddet sonra en kısa yol 0-3- 4-5 bulunacaktır.

7. Düğümler arasındaki yol birim cinsinden şekilde verilmiştir.

8. Başlangıçta feromon miktaları 0 dır.

9. Her yoldan geçtiklerinde 1 birim feromon bırakırlar.

10. Feromon eşik değeri 25’tir.

11. Dolayısıyla bir yolun seçilmesi için karıncaların bu yolu 25 kez geçmesi gerekir.

12. Feromonun uçuculuk değeri ihmal edilmekte ve karıncaların aynı hızda ilerlediği düşünülmektedir.

13. Karınca sayısı 10 ‘dur.

14. Örneğin üst ve alt yolu 5’er karınca seçsin.

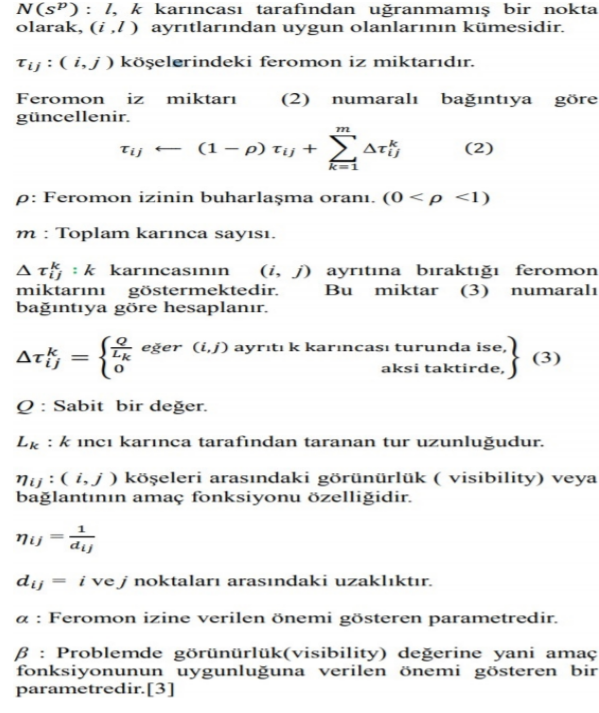
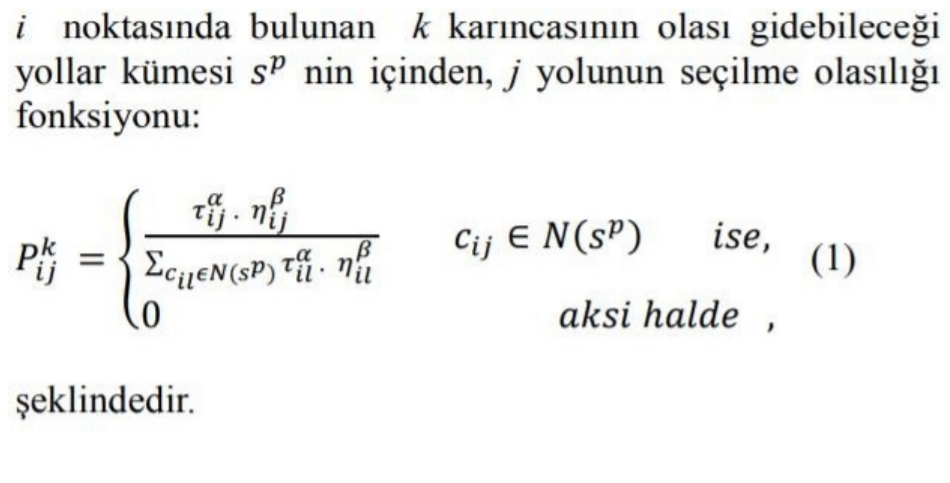
15. Karıncalar her geçtiklerinde yolda 1 birim feromon bırakır.

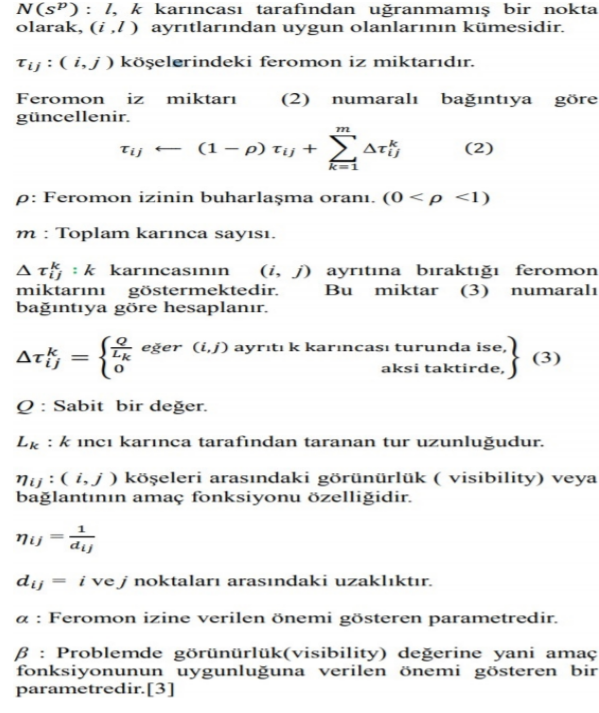
16. Alt yol daha kısa olduğu için bu yolu izleyen karıncalar daha önce yuvaya ulaşır.

17. Kenardaki feromon miktarları ise [0-3] kenarı için 10 birim (5 karıncadan dolayı)

18. [0-1] kenarı için 5 birim olacaktır.

19. Feromon değeri 25 değerini aşınca yuvadan çıkan karıncalar bu yolu izleyerek yeme ulaşmaya çalışırlar.







FEROMON BUHARLAŞMASI

Gerçek karınca kolonilerinde, feromon yoğunluğu buharlaşmadan ve bozulmadan dolayı azalmaktadır. KKA’da buharlaşmanın etkisi, uygun şekilde tanımlanmış bir buharlaşma kuralının uygulanmasıyla simüle edilmektedir.

Örneğin, feromonin yapay gecikmesi sabit bir oran olarak ayarlanabilir. Feromon buharlaşması, araştırmanın erken safhalarındaki yapay karıncaların oluşturduğu düşük kaliteli çözümlerde bırakılmış olan feromonlerin etkisini azaltır. Feromon buharlaşması gerçek karıncalarda fark edilir bir etki yapmasa da yapay karınca kolonilerinde çok yararlı olabilmektedir.

KULLANIM ALANLARI

Karınca Sürüsü Algoritmasının ve diğer sezgisel algoritmaların kullanım alanları en kısa zaman problemleriyle sınırlı değildir. Karıncalar yalnızca bu amaç için kullansa da matematiksel formüller değiştirilerek aynı yapı en az kaynak (para veya enerji) veya başka optimizasyonlar için Örneğin yukarıda bahsettiğimiz gibi bir kargo şirketinin dağıtım rotasını belirlemek için kullanabilir -ki kargo şirketleri için sezgisel algoritmalar kullanan bu tür yazılımlar vardır. Bunun yanında en kısa zaman yerine maksimum sözcük ilişkisi koyarsak, rahatlıkla web arama benzeri bir metin tarama ve arama sistemi elde ederiz.

Google ve benzeri arama motorları arka planda sözcük ilişkisi ve sayfa değerini maksimize etmekte optimizasyon algoritmalarını kullanırlar. Google optimizasyon için kullandığı algoritmayı (ve bu algoritmanın bu yazıda bahsedilen sezgisel algoritmalardan biri olup olmadığını) açıklamamaktadır. Bununla beraber örneğin Karınca Optimizasyonu Algoritması bir arama motoru tarafından bu amaçla kullanılabilir.

YAPAY KARINCALAR

Gerçek karıncalar, kör olmalarına rağmen yuvalarından yiyeceğe giden en kısa yolu bulabilmektedir. Karıncaların bu özelliği, birtakım özellikler aynen kullanılarak ve bazı eklemelerle gerçek problemlerin çözümünde kullanılabilir hale getirilmiştir.

Yapay karıncalar oluşturulurken gerçek karıncalardan alınan özellikler:

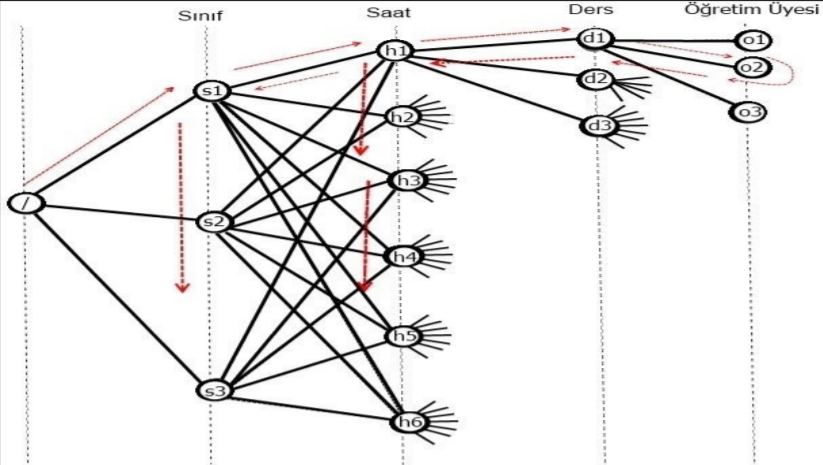
Karıncalar arasında feromen aracılığı ile kurulan iletişim, Feromeni fazla olan yolların öncelikle tercih edilmesi, Kısa yollarda feromen miktarının daha hızlı artması.

Gerçek karıncaların dışında eklenen özellikler:

Zamanın ayrık olarak hesaplandığı ortamda yaşarlar, Tamamen kör olmayıp problem ile ilgili detaylara erişebilirler, Belli bir miktar hafıza ile problemin çözümü için oluşturdukları bilgileri tutabilirler. Karınca tabanlı algoritmalarda temel fikir, basit iletişim mekanizmalarını kullanan yapay akıllı aracıların, birçok karmaşık problem için çözümler üretebilmesidir

KARINCA KOLONİ ALGORİTMASININ BİLGİSAYARDA UYGULANMASI

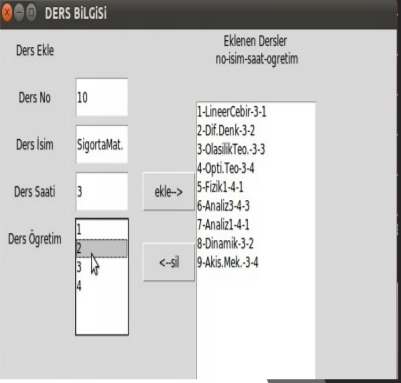
Çözülecek problemin bir amaç fonksiyonu vardır ve bir sonraki iterasyon ile elde edilecek nokta bir önceki noktanın amaç fonksiyonunda aldığı değere göre belirlenmektedir. Bu çalışmada öğretim planı, öğretim üyelerinin zaman talepleri derecelendirilerek, sezgisel algoritmalardan sürü zekasına dayanan “Karınca Kolonisi Algoritması” üzerinde modellenmiş ve Python diliyle yazılan bir uygulama ile göreceli parametre dahil edilerek öğretim planı oluşturulmuştur



Program iki ana bölümden oluşur. Birinci kısım, verilerin programda kullanılmak üzere, veri tabanına girilmesi için ara yüzlerden ve veritabanından, ikinci kısım ise girilen bilgiler doğrultusunda ders dağılımını yapan ve çıktı olarak bir text dosyası oluşturan ana programdan oluşmaktadır.

Sırası ile program bilgisi, ders bilgisi, öğretim üyeleri bilgisi ara yüzler ile veritabanına alınır. Daha sonra ana program çalıştırılır. Programda beş parametre kullanılmaktadır. Bunlar: KKA ’nın bir bileşeni olan α, , , iterasyon ve karınca sayılarıdır.

Program veri tabanına bağlanıp veri tablolarını kendi içine kopyalar. KKA’ nın çalışması için eşit başlangıç değerleri ile feromon listesi oluşturulur. Optimal noktaya çok uzak bir amaç fonksiyon değeri belirlenir. İterasyon döngüsü başlar. Her iterasyon sonunda genel feromon listesine eklenmek üzere geçici bir feromon listesi oluşturulur. Karınca döngüsü başlar. Tüm karıncalar dolaşımını tamamladıktan sonra genel liste, geçici feromon listesi ile güncellenir. İterasyon tamamlanana kadar program devam eder. En sonunda geçerli program ve değer bir text dosyası olarak sunulur.



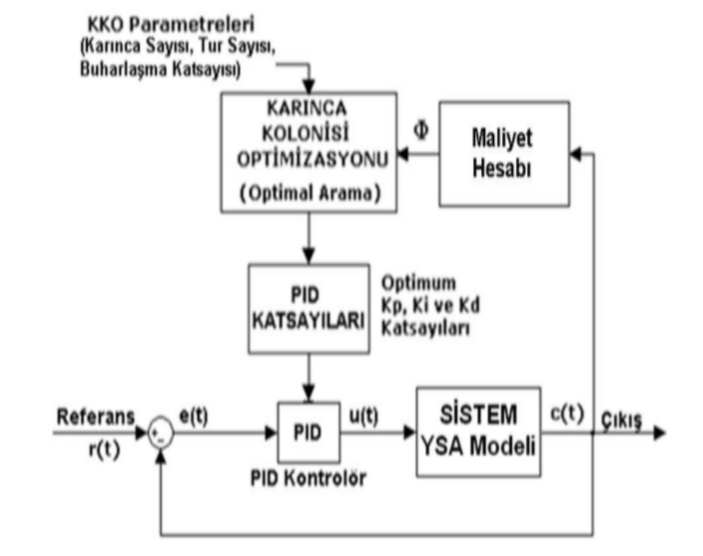


Bu çalışmada öğretim planının, bir parametreye bağlı derecelendirme ile Karınca Kolonisi Algoritması’na uygun ağırlıklı graf modeli geliştirilmiş ve bu modelin uygulanabilirliğini gösteren bilgisayar programı ile de desteklenmiştir.

Ayrıca model, bir çok ağırlıklandırma parametresi ve amaç fonksiyonu değişkeni eklenerek geliştirilmeye de açıktır. Bu doğrultuda bilgisayar programına bir çok modül eklenerek karmaşık atamalar yapılabilir. Diğer taraftan bu model ve uygulaması ders çizelgelemenin dışında bir çok probleme de uyarlanmaya açıktır.

KKA ALGORİTMASI İLE PID CONTROLLER TASARLANMASI

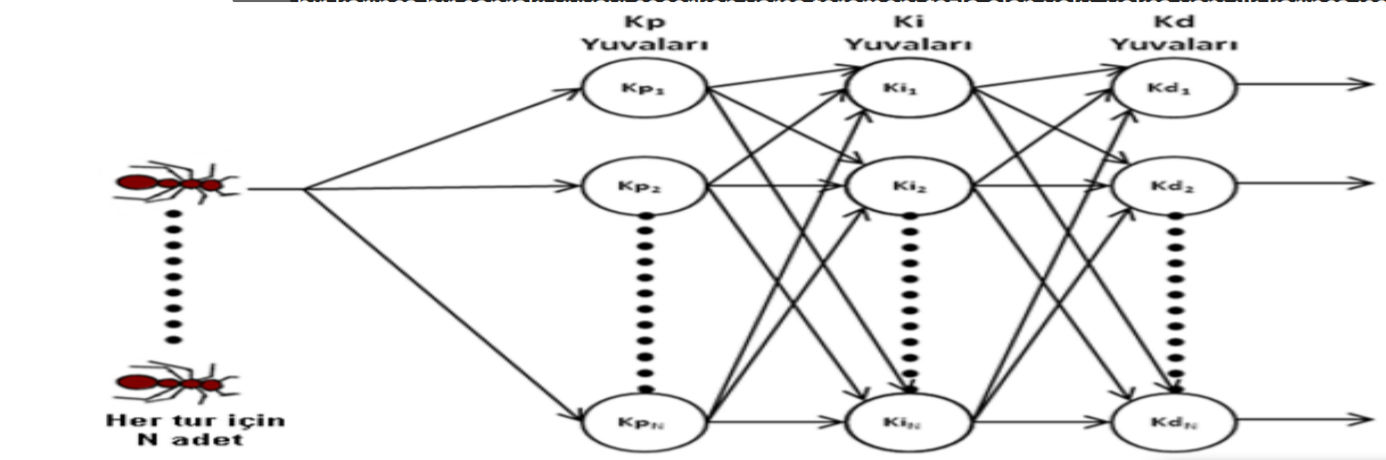
Karıncaların en kısa yolu bulma mantığından yola çıkılarak elde edilen KKA algoritması ile herhangi bir kontrolöre ait parametre optimizasyonu sağlanabilmektedir. PID kontrollü bir sistemin Kp, Ki, ve Kd parametrelerini KKA algoritması ile optimize etmek mümkün olmaktadır. Şekildeki verilen referansı izlemek üzere gerekli PID katsayılarının bulunması için KKA algoritması ile tasarlanmış optimizasyonun blok diyagramı görülmektedir.



KKA algoritmasında her yapay karıncanın seçmiş olduğu yollar farklı PID katsayılarına denk gelir. Optimizasyon iilemi başladığında KKA, kontrolör için PID katsayılarını atar ve bu katsayılar ile sistem çevrimdışı çalıştırılır. Sisteme verilen referans olarak yörünge verilmektedir. Böylece KKA’nın bulacağı PID katsayı değerleriyle çalıştırılan simülasyondan elde edilen çıkış ile referans yörünge karşlaştırılacaktır.

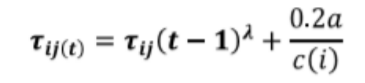
Birbirinden farklı bu çıkış eğrileri aslında her bir karıncanın izlediği yol olarak düşünülebilir. Dolayısıyla referans yörüngeye en yakın çıkışı verecek olan PID değeri optimal değerlerdir. Çıkış işareti optimizasyon sürecinde incelenirse ilk başlardaki rastgele ve dağınık olan eğriler kendini daha dar bir bantta ve referansın etrafında hareket eden eğrilere bırakır. Böylece sistem çıkışı ile referans yörünge arasında az hata oluşturan yani yapay karıncalar tarafından fazla tercih edilerek feromoni yüksek olan yollara ait değerler, optimizasyon sonucu elde edilen Kp, Ki ve Kd katsayılarıdır.

Optimize edilecek parametre sayısı arttıkça yuva sayısı arttıkça karıncaların işi daha da zorlaşmaktadır. Ancak bir süre sonra kötü yolların feromon miktarının azalması, kısa yolların feromon miktarının artması ile karıncalar sadece iyi yollar arasından en iyisini bulmaya çalışacaklardır. Bu sürecin sonunda karıncalar referansa çok yakın yol bulacaklardır. KKA algoritmasının tasarımında dikkat edilmesi gereken bazı özellikler vardır..

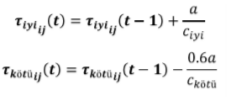
edilecek parametre sayısı arttıkça yani

Öncelikle optimize edilecek değişkenlerin arama uzayının çok büyük olmasını engellemek için maksimum ve minimum değerleri seçilir. Bu sayede KKA, sınırlandırılan çalışma uzayı içerisinde arama yapabilecektir. Şekilde gösterildiği gibi PID kontrolör için her bir katsayı değeri bir yuva gibi düşünülürse bu yuvalara giren bir karınca bir sonraki yuvayı seçerken varsa feromoni fazla olan yolu, yoksa yani ilk karınca ise rastgele bir yuva seçerek diğer yuvaya girmektedir.

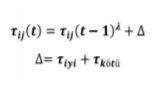
Her yuvadan çıkışında önceki yuvalardan aldığı değerleri de tutmaktadır. 3. yuvadan çıktığı zaman elde ettiği değerler PID parametrelerine ait değerlerdir. Bu değerler karıncanın izlediği yol ile elde edilecek yörüngeyi verecektir. Dolayısıyla elde edilecek yörünge ile referans yörünge karşılaştırılarak karıncanın izlediği yollara “ödül” veya “ceza” feromonleri verilecektir. Gerçekleştirilen çalışmada, aynı tur içerisinde “lokal feromon güncelleme”, her bir tur sonunda ise “global feromon güncelleme” olmak üzere feromon güncelleme iki şekilde yapılmaktadır. Lokal feromon güncelleme işleminde her bir karınca izlediği yolu bitirdikten sonra bırakılan feromoni her bir tur içerisinde aşağıdaki denklemde verilen eşitlik ile güncellemektedir



Burada **τij**, feromon matrisinin satır ve sütun indisindeki feromon miktarını, a genel feromon güncelleme katsayısı, λ buharlaşma katsayısı ve c(i) ise karınca tarafından o tur içerisinde elde edilen referans yörünge ile elde edilen çıkış eğrisi arasındaki RMS hatasıdır. Dolayısıyla karıncanın izlediği yola ait RMS hatası düşük oldukça yola verilen feromon değeri artacaktır. Global feromon güncelleme için ise pozitif ve negatif olmak üzere iki farklı feromon güncelleme yöntemi kullanılmaktadır. Her tur sonunda o tura ait en iyi ve en kötü tura ait feromon değerleri aşağıdaki deneklemlerde verilen eşitlikler ile tekrar değiştirilmektedir.



Bu denklemlerde verilen **τiyi**, referans ile sistem çıkışı arasında oluşan en küçük RMS hatası iken ise en büyük RMS hatasını göstermektedir. Aynı tur içerisinde ve ise elde edilen feromon değişimlerini göstermektedir. Dolayısıyla tur içerisindeki en iyi karıncaya ait yola kadar feromon daha eklenirken tur içerisindeki en kötü karıncadan kadar feromon azaltılmaktadır. Karıncaların yolları izlemeleri sırasında zaman içerisinde kullanılmayan yollardaki feromon miktarları azaltılmaktadır. Bu sayede karıncanın yeni yollar denemesine olanak tanınmakta ve algoritmanın doğru çalışması sağlanmaktadır. Her bir turdan sonra yapılacak feromon buharlaştırma işlemi için kullanılacak eşitlik aşağıdaki denklemlerde verilmiştir.



**: Buharlaşma katsayısı**

**: En iyi karıncaya ait indisleri tutan matris**

**: En kötü karıncaya ait indisleri tutan matris**

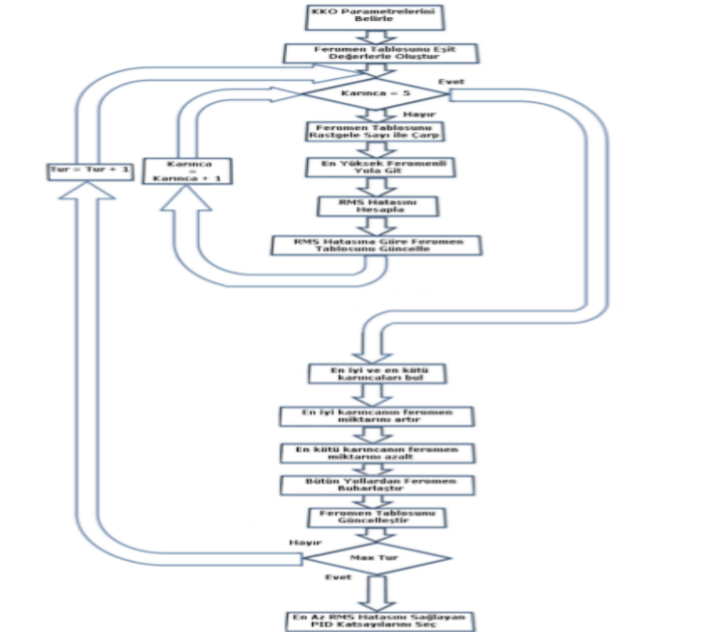
**: Her tur sonundaki feromon miktarını tutan matris**

Burada KKA değişkenlerinin (λ,a) değerleri deneme yanılma ile verilmelidir. En kısa zamanda referans yörünge üzerinde sonuç veren değerler o sistem için seçilen en uygun parametre değerleridir. Benzer şekilde çıkış eğrisi referans yörüngeye yüksek doğrulukta oturmuştur.

KKA ile PID kontrolörünün katsayıları elde edilirken seçilen KKA algoritmasının parametreleri deneme yanılma metodu ile bulunmuştur. Yapay karıncaların geçecekleri yuva, PID katsayıları olarak ayarlanmıştır.

PID katsayılarının alt değerleri 0, Üst değerler verilen referansa göre değişiklik göstermektedir. Ancak farklı üst değerler verilerek yapılan çalışmalar sonucunda bulunan PID parametrelerinin sınır değerleri Kp=20, Ki=5 ve Kd=10 olarak civarında bulunmuştur. Her bir PID parametresi alt limit ile üst limit arasında 100‟er parçaya bölünerek, optimizasyon çalıştırıldığında yapay karıncaların geçebilecekleri yuvalar yani her turda bir karıncanın seçebileceği Kp, Ki ve Kd değerleri olarak ayarlanmıştır.

KKA ile PID kontrolörün katsayıları bulunurken izlenen adımlar aşağıda gösterilmiştir. İlk olarak optimizasyon için KKA parametreleri belirlenmiştir. Yapılan çalışmada her turda 5 adet karınca olmak üzere toplam 100 tur gerçekleştirilmiştir. Karıncaların izledikleri yollardaki feromon değerini artırmak için feromon sabiti a= 0,06 olarak seçilmiştir. Her tur sonunda karıncaların kullandıkları yollara bıraktıkları feromon miktarı, buharlaşma sabiti λ= 0.95 ile buharlaştırılmıştır.



KKA parametreleri seçildikten sonra yapay karıncaların gidebilecekleri yolların hepsine eşit sayıda feromon değeri verilmiş ve bu değer feromon matrisinde tutulmuştur. Ġlk yapay karınca yuvadan çıktığında gidebileceği bütün yollarda eşit miktarda feromon olduğu için rastgele bir yol seçerek toplam üç yuvaya uğrayarak yolunu tamamlamıştır.

Yolun sonunda seçilen PID katsayıları ile simülasyon çalıştırılmış ve referans ile simülasyon çıkışı arasındaki RMS hatası hesaplanarak bu yapay karıncanın geçmiş olduğu yolların feromon matrisindeki değeri artırılmıştır. Arkasından gelen yapay karıncanın da tur içinde aynı yoldan gitmemesi için feromon tablosu rastgele sayı ile çarpılarak güncellenmiştir. Böylece her turda karıncalar farklı PID katsayıları değerlerini alarak verilen referans yörüngeyi takip etmeye çalışmışlardır. 5 yapay karınca da turunu tamamlandığında bu turdaki en az RMS hatasına sahip yapay karıncanın yolundaki feromon miktarı ekstra artırılmış ve en fazla RMS hatasına sahip yapay karıncanın yollundaki feromon miktarı iyi olan karıncanın artırılan miktarından daha fazla oranda azaltılmıştır.

Ayrıca her tur sonunda bütün yapay karıncaların geçtikleri yollardaki feromon miktarı, buharlaşma sabiti(λ) ile buharlaştırılmıştır. Eğer seçilen PID katsayısı ile küçük RMS hatası elde edilmiş ise bir sonraki karıncalarda bu katsayılar etrafında dolanarak toplam turlarını bitirmeye çalışacaktır. Maksimum tur tamamlandığında feromon tablosundaki en yüksek değere sahip yollar, yani seçilmiş olan PID katsayıları optimizasyon sonucu olarak kaydedilir.

Sonuç

Şu anda optimizasyon amacıyla yazılmış birçok programda yukarıda bahsedilen “Karınca Koloni Optimizasyonu” algoritması kullanılıyor. Bununla beraber, bu yöntem, optimizasyonla uğraşan mühendislerin doğadan esinlendikleri tek yöntem değildir. Bu bakımdan insan için hala doğa, bulunabilecek en iyi öğretmendir…

# Kaynakça

Marmara üniversitesi yüksek lisans tezi 2008: Muhammet ÜNAL

Karınca Kolonisi Algoritmasının Çizelgelemesi Üzerine Zaman Bir Modellemesi ve Uygulaması : Hülya ÖZDAĞ, Nilgün AYGÖR, Aykut PARLAK