

ENERJİ KALİTESİ İÇİN YÜK-FREKANS KONTROLÜNDE YAPAY ARI KOLONİSİ ALGORİTMASI UYGULAMASI

Haluk GÖZDE¹, M.Cengiz TAPLAMACIOĞLU²

^{1,2} Gazi Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümü, Ankara
halukgozde@gmail.com, taplam@gazi.edu.tr

Anahtar Kelimeler: İki bölgeli güç sistemi, Yük-frekans kontrolü, Oran-integral kontrol, Yapay arı kolonisi algoritması, Parçacık sürüsü algoritması

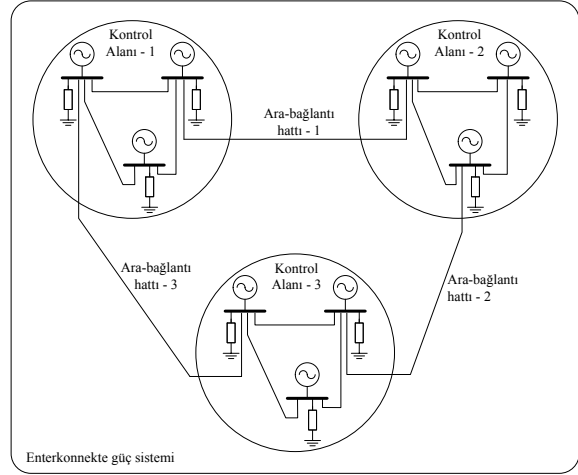
ÖZET: Bu çalışmada, güç sistemlerinin yük-frekans kontrolünün optimizasyonunda popülasyon tabanlı Yapay Arı Kolonisi algoritmasının kullanımı ele alınmıştır. İki bölgeli bir güç sisteminin ikincil yük-frekans kontrolünü sağlayan oran-integral (PI) kontrolörün optimal kazanç değerleri Yapay Arı Kolonisi algoritması ile elde edilmiş; bu parametrelerle elde edilen sonuçlar, benzer bir optimizasyon algoritması olan Parçacık Sürüsü algoritmasının literatürde mevcut sonuçları ile karşılaştırılmıştır. Çalışmanın sonunda Yapay Arı Kolonisi algoritmasının yük-frekans kontrolündeki performansı incelenmiştir.

GİRİŞ

Elektrik enerjisi; üretim noktasından tüketiciye kadar hızlı, kolay ve ekonomik taşınabilmesi ve kolay kullanılabilir olması nedenleriyle, en çok kullanılan enerji kaynaklarından birisi konumundadır. Günümüzde elektrik enerjisi, tüketiciye büyük güç sistemleri üzerinden ulaştırılır. Elektrik güç sistemlerinin görevi, anma uç gerilimine ve sistem frekansına sahip elektrik enerjisini ekonomik olarak üretmek, taşımak ve dağıtmaktır. Bu sistemler genel olarak, birbirlerine bağlı üretim üniteleri ile iletim ve dağıtım şebekelerinden oluşur. İletim şebekesi, elektrik enerjisinin üretim noktasından tüketim bölgesine kadar taşınması ile ilgili orta ve yüksek gerilim hatlarının tesisi ve işletilmesini sağlarken; dağıtım şebekesi, iletim şebekesinden son kullanıcıya kadar olan alçak gerilim hatlarının tesisi ve işletilmesini sağlar. Üretim ünitesinin görevi ise değerleri standartlarla belirlenmiş sabit ve kararlı anma frekansına ve uç gerilimine sahip elektrik enerjisini ekonomik olarak ve çevreye zarar vermeksizin üretmektir [1].

Frekansın anma değerinde sabit ve kararlı tutulması elektrik güç sistemlerinin ana problemlerinden birisidir. Frekansın değeri asıl olarak şebekede üretilen ve tüketilen aktif güçlerin dengesine bağlıdır. Tüketilen gücün üretilen güce göre artması ya da azalması, jeneratörlerin dönüş hızının, diğer deyişle sistem frekansının değişmesine neden olur. Güç sistemleri, genel olarak kendi tüketimlerini yani kendi yüklerini kendi üretim üniteleri ile karşılayabilecek şekilde tasarlanırlar. Tüketim arttığında üretim üniteleri de rezervlerini kullanarak üretimlerini

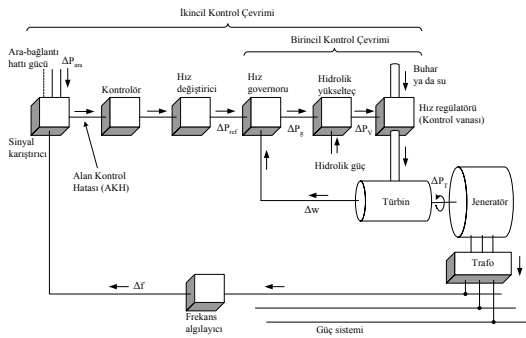
arttırmırlar. Bununla beraber, komşu güç sistemleri, birbirlerine bağlı çalışarak sistemde tüketilen aktif gücü paylaşabilir ve bu sayede, ünite başına fazladan üretilmesi gereken enerji miktarını düşürerek, frekansın yük değişiminden daha az etkilenmesini sağlayabilirler. Komşu güç sistemlerinin yükü paylaşarak ortak bir frekans referansı oluşturarak daha kararlı, güvenilir ve sabit bir sistem frekansını birbirlerine bağlı olarak çalıştıkları böyle şebekelere enterkonnekte güç sistemi adı verilir (Şekil 1). Enterkonnekte sistemde frekansın değeri sistemin her yerinde aynıdır.



Şekil 1. Enterkonnekte elektrik güç sistemi

Güç sisteminde frekansın kontrol altında tutulması, sistemde üretilen ve tüketilen aktif güçlerin dengede tutulmasına bağlıdır [2]. Ancak, elektrik enerjisinin yüksek miktarlarda depolanamaması ve yük talebinin sürekli değişmesi nedeniyle bu dengenin korunması bazı kontrol zorlukları içerir. Temel olarak, elektrik güç sistemlerinde frekans kontrolü, Şekil 2’de gösterilen ardışık iki otomatik kontrol çevrimini içeren yük-frekans kontrolü mekanizmasıyla gerçekleştirilir [3-5]. Sistemde, yük talebinin artması ile birlikte frekans düşmeye başlar. Üretim üniteleri, buna ilk tepki olarak kendileri üzerinde depo edilmiş kinetik enerjilerini kullanarak düşüşü yavaşlatmaya çalışırlar. Birkaç milisaniye sonra frekansın düşüşü algılanabilir seviyeye geldiğinde yük-frekans kontrolü çevrimleri devreye girmeye başlar. Öncelikle birincil kontrol adı verilen kontrol çevrimi ile üretim tüketim dengesi yeniden sağlanarak frekansın düşüşü

durdurulur ve belli bir değerde sabitlenir. Ardından, ikincil kontrol çevrimi ile frekans bir integral kontrolör ile yeniden ayarlanarak anma değerine getirilir [4]. Temel olarak birincil kontrol, üretim ünitelerinde governorlar ile yerel olarak gerçekleştirilen bir türbin kontrolüdür. İkincil kontrol ise yük-frekans kontrolü mekanizmasının asıl kontrol problemini oluşturur ve frekansta birincil kontrol sonunda oluşan hatayı algılayıp, bir integral kontrolör vasıtasıyla merkezi olarak gideren kontrolleri içerir [3-5].



Şekil 2. Yük-frekans kontrolü çevrimleri [5]

Bu çalışmada, yük-frekans kontrolünde ikincil kontrolü sağlayan PI-kontrolörün kazançları, sistemden optimum frekans sapması ve optimum ara-bağlantı hattı güç değişimi cevabı elde edilebilmesi amacıyla Yapay Arı Kolonisi algoritması kullanılarak yeniden ayarlanmış ve algoritmanın ayar performansı literatür ile karşılaştırılarak incelenmiştir. Karşılaştırma amacıyla benzer bir optimizasyon algoritması olan Parçacık Sürüsü algoritmasının literatürde mevcut sonuçları ele alınmıştır. Çalışmanın sonucunda frekans sapması ve ara-bağlantı hattı güç değişimi eğrileri ile bunların maksimum aşım ve oturma zamanı değerleri elde edilerek literatüre sunulmuştur.

YAPAY ARI KOLONİSİ ALGORİTMASI

Doğrusal olmayan fonksiyonların optimizasyonunda kullanılan klasik matematiksel yöntemler, gerçekleşmesinde yazılımsal ve donanımsal bazı zorlukları da beraberinde taşıyan türev ifadeleri içerir. Optimizasyon hesapları içinde türev ifadelerinden kaçınmak ve nispeten daha basit ve kısa yazılımla, daha kısa sürede sonuca yakınsamak için ekosistemdeki canlıların doğal seçim davranışlarından esinlenen iterasyona dayalı optimizasyon yöntemleri geliştirilmiştir. Önceleri daha çok mutasyon, çaprazlama ve yeniden üreme gibi evrimsel bir takım operatörler içeren bu yöntemler daha sonra yerini, canlıların besin bulma davranışlarından yola çıkan popülasyon tabanlı, sürü zekasına dayalı optimizasyon algoritmalarına bırakmaya başlamıştır. Yapay arı kolonisi algoritması da 2005 yılında Karaboğa

tarafından, bal arısı kolonilerinin besin bulma davranışlarından esinlenerek geliştirilmiş sürü zekasına dayalı bir optimizasyon algoritmasıdır [6]. Algoritmada arılar, arama uzayındaki konumları ile ifade edilir. Arama uzayının boyutunu, çözüm içinde optimize edilecek parametrelerin sayısı belirler. Her bir arı, arama uzayında bir besin kaynağına bağlı olarak çalışır. Arıların buldukları konumlar, yani besin kaynakları aynı zamanda problemin olası çözümleridir. En iyi besin kaynağı, o besin kaynağının nektar miktarının ölçülmesi ile yani bir maliyet fonksiyonu kullanılarak uygunluk değerinin hesaplanması ile belirlenir. Arılar yeni konumlarını, bir önceki konumlarını ve önceki diğer en iyi besin kaynaklarının konumlarını dikkate alarak değiştirirler. Arama uzayında, arılar tarafından bulunan en uygun besin kaynağının konumu problemin çözümü olarak atanır.

Yapay arı kolonisini üç grup arı oluşturur. Bunlar; besin kaynaklarını tutan işçi arılar, besin kaynaklarını seçmek üzere işçi arıların dansını izleyen gözcü arılar ve rastgele arama yapan kâşif arılardır. Popülasyonun yarısını işçi arılar, diğer yarısını da gözcü arılar oluşturur. Algoritmanın başlangıcında, işçi arılar yeni besin kaynakları bulurlar ve buldukları besin kaynaklarının uygunluklarını belirlemek üzere maliyet fonksiyonu ile nektar miktarlarını ölçerler. Ardından, buldukları besin kaynağının komşuluklarında daha uygun yeni bir besin kaynağı aramaya başlarlar. Bir işçi arı, daha uygun yeni bir besin kaynağı bulunduğu takdirde önceki kaynağı gözcü arılara devreder ve kendisi bu yeni besin kaynağını tutmaya başlar. Belli bir iterasyon sayısı kadar daha uygun yeni bir besin kaynağı bulamayan işçi arı bulunduğu besin kaynağını da bırakarak kâşif arıya dönüşür ve arama uzayında rastgele arama yapmaya başlar.

Gözcü arıların işçi arılardan besin kaynaklarını devralması, genel olarak olasılığa dayalı bir seçim işlemi ile gerçekleştirilir. Algoritmada, işçi arılar ve gözcü arılar yerel arama yaparken, kâşif arılar global arama yapar. Bu sayede, Yapay Arı Kolonisi algoritmasında yerel ve global arama yetenekleri birleştirilerek optimum sonuca yakınsama hızı artırılır. Yapay Arı Kolonisi algoritmasının algoritma adımları aşağıda belirtilmiştir.

Başlangıç parametrelerinin belirlenmesi

Tekrar (Repeat)

İşçi arıları besin kaynaklarına gönder ve nektar miktarlarını hesapla

Gözcü arıları besin kaynaklarına gönder ve nektar miktarlarını hesapla

Rastgele yeni besin kaynakları bulmaları için kâşif arıları gönder

O ana kadarki en iyi besin kaynağını hafızada tut

Kadar (Until), (Durdurma kriteri sağlanana kadar)

Yapay Arı Kolonisi algoritmasında öncelikle başlangıç parametrelerinin belirlenmesi gerekir.

Algoritmanın avantajlarından birisi de parametre sayısının az olmasıdır. Başlangıçta belirlenmesi gereken üç adet parametre vardır. Bunlar;

- *Besin kaynağı sayısı (İşçi arı sayısı)*: Her bir besin kaynağında yalnızca bir adet işçi arı bulunmaktadır. Dolayısı ile bu parametre toplam arı sayısının yarısını oluşturur.
- *Limit*: İşçi arıların kâşif arı olması için gerekli, gelişme olmayan iterasyon sayısıdır.
- *İterasyon sayısı*: Algoritmanın en fazla kaç adım koşturulacağını belirleyen parametredir.

Yapay Arı Kolonisi algoritması genel olarak dört işlem safhasından oluşur. Bunlar; başlangıç safhası, işçi arı safhası, gözcü arı safhası ve kâşif arı safhasıdır [6, 7].

Başlangıç Safhası

Başlangıç safhasında işçi arılar rastgele olarak besin kaynaklarına gönderilir. Bu işlem, S adet besin kaynağı içeren x_{mi} matrisinin elemanlarının Eş.1 ile rastgele atanmasıyla temsil edilir.

$$x_{mi} = l_i + \text{rand}(0,1).(u_i - l_i)$$

$$(m = 1, 2, \dots, S \quad \text{ve} \quad i = 1, 2, \dots, n) \quad (1)$$

Yukarıdaki eşitlikte m besin kaynağı indisini, i besin kaynağını yani çözümü oluşturan parametre indisini, u parametrelerin üst sınırını ve l ise parametrelerin alt sınırını göstermektedir [6, 7].

İşçi Arı Safhası

Arılar, işçi arı safhasının başında x_{mi} matrisi ile belirtilen besin kaynaklarında bulunurlar. Bu safhada işçi arılar buldukları besin kaynaklarının komşuluğunda yeni besin kaynakları aramak ve bu kaynakların nektar miktarlarını ölçmekle yani uygunluk değerlerini hesaplamakla görevlendirilmişlerdir. Algoritmada, yeni besin kaynakları Eş.2 ile tanımlanan v_{mi} matrisi ile belirtilmiştir. Eşitlikte x_k rastgele seçilen bir besin kaynağını, i rastgele seçilen parametre indisini ve ϕ_{mi} [-a, a] aralığında rastgele seçilen adım katsayısını belirtmektedir.

$$v_{mi} = x_{mi} + \phi_{mi} \cdot (x_{mi} - x_{ki}) \quad (2)$$

v_{mi} matrisinin oluşturulmasının ardından, probleme göre tanımlanmış bir maliyet fonksiyonu ile bu yeni besin kaynaklarına ait uyum değerleri hesaplanır. Uyum değerleri, maliyet fonksiyonu ile belirlenen $f_m(x_m)$ maliyet değeri kullanılarak Eş.3 ile hesaplanır. İşçi arı safhasının sonunda, uyum değerleri hesaplanan besin kaynakları “greedy selection” adı verilen olasılıksal bir seçim işlemi ile gözcü arılara devredilir [6, 7].

$$Uym_m(x_m) = \begin{cases} \frac{1}{1 + f_m(x_m)} & \text{eğer} \quad f_m(x_m) \geq 0 \\ 1 + |f_m(x_m)| & \text{eğer} \quad f_m(x_m) < 0 \end{cases} \quad (3)$$

Gözcü Arı Safhası

İşçi arıların dışında kalan iki grup arı vardır. Bu arılar gözcü arılar ve kâşif arılardır. İşçi arılar, besin kaynaklarının nektar miktarlarını yani uyum değerlerini belirledikten sonra bu kaynakların konum ve nektar miktarı bilgilerini kovanda bekleyen gözcü arılarla paylaşırlar. Bu işlemin ardından gözcü arılar, işçi arıların paylaştığı konum ve nektar bilgilerine dayanarak, olasılıksal bir hesaplama ile kendilerine bir besin kaynağı seçerler. Bu seçme işlemi, işçi arılar tarafından sağlanan uyum değerlerini kullanarak hesaplanan olasılık değerlerine bağlı olarak yapılır. Bu amaçla algoritmada rulet tekerleği seçim yöntemi kullanılmıştır (Eş.4), [6].

$$h_i = \frac{Uym_i}{\sum_n^S Uym_n} \quad (4)$$

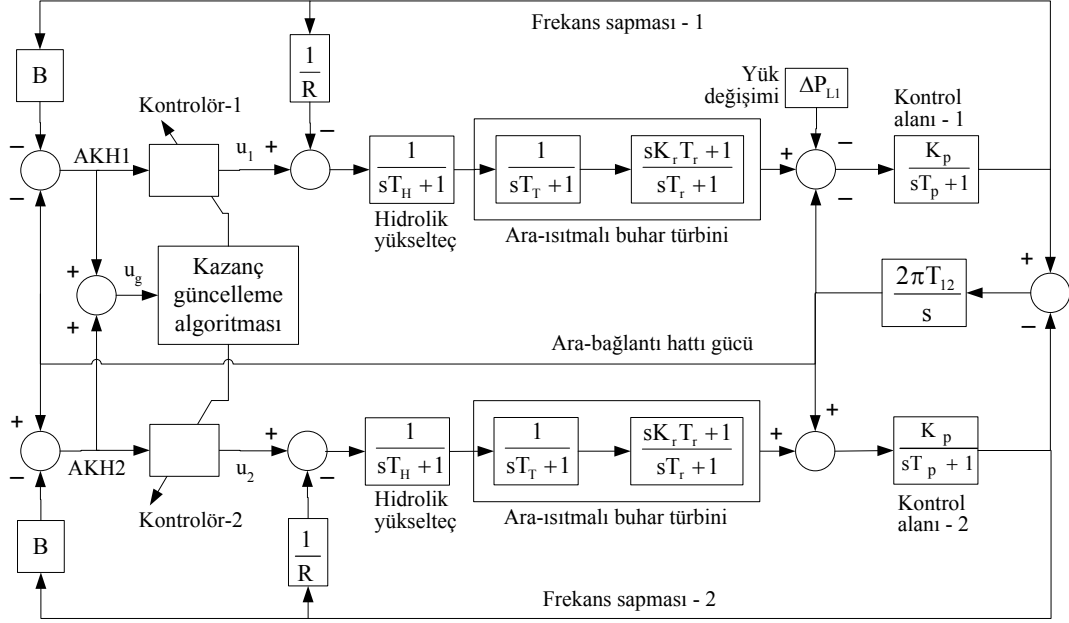
Eş.4’de h_i gözcü arı tarafından besin kaynağının seçilme olasılığıdır. Uygunluk değerinin toplam uygunluk değerine oranı ile i’inci besin kaynağının seçilme olasılığı belirlenmiş olur. Bir besin kaynağı gözcü arı tarafından seçilirken, 0 ile 1 arasında algoritma tarafından rastgele bir sayı üretilir. Besin kaynağının olasılığı bu değerden büyükse o besin kaynağı gözcü arı tarafından seçilir. Küçükse sıradaki besin kaynağına bakılır. Bu şekilde gözcü arıların tamamı bir besin kaynağı seçene kadar işlemlere devam edilir [6, 7].

Kâşif Arı Safhası

Algoritma boyunca bazı besin kaynaklarının konumları o besin kaynağını bağlı olan işçi arılar tarafından geliştirilemez. Eğer bir besin kaynağı limit parametresi kadar iterasyonda geliştirilememiş ise bu besin kaynağındaki işçi arı kâşif arıya dönüşür ve arama uzayında rastgele bir başlangıç konumuna geri döner. Arama işlemine bu konumdan işçi arı olarak yeniden başlarlar [6, 7].

SİMÜLASYON ÇALIŞMASI

Yapay arı kolonisi algoritmasının kontrol sistemi üzerindeki ayar performansı MATLAB simülasyon ortamı kullanılarak incelenmiştir. Simülasyonlarda kullanılan güç sistem modeli olarak ara-ısıtmalı buhar türbini içeren iki kontrol alanına sahip bir enterkonnekte sistemin doğrusallaştırılmış modeli kullanılmıştır. Sistemde her iki kontrol alanı için birer PI-kontrolör kullanıldığı ve literatür ile karşılaştırılabilmesi amacıyla her iki kontrol alanının da birbirine eşit ve Çizelge 1’de belirtilen sistem parametrelerine sahip olduğu kabul edilmiştir [8].



Şekil 3. İki bölgele enterkonnekte güç sistemi modeli

Çizelge 1. Güç sistem parametreleri

Parametre Simgesi	Parametrenin Adı	Parametrenin Değeri
T_H	Hidrolik zaman sabiti	0,08 s
T_T	Türbin zaman sabiti	0,3 s
T_r	Ara-ısıtma zaman sabiti	10 s
T_{12}	Senkronizasyon sabiti	0,086
K_r	Ara-ısıtma kazancı	0,5
T_p	Kontrol alanı zaman sabiti	20 s
K_p	Kontrol alanı kazancı	120
R	Regülasyon sabiti	2,4 Hz/MW
B	Frekans kutuplama sabiti	0,425 MW/Hz

Yapay Arı Kolonisi algoritmasının güç sistemi üzerindeki ayar performansı, Abdel-Magid ve Abido'nun literatürde mevcut çalışmalarında [8] elde ettikleri sonuçlar ile karşılaştırılmıştır. Simülasyonda, kontrolör girişine beslenecek hata sinyalinin ve kazanç güncelleme algoritmasına beslenecek güncelleme bilgisinin Eş.5'te gösterilen AKH (Area Control Error - ACE) olduğu, kontrolör çıkışında ise Eş.6'da belirtilen kontrol sinyalinin üretileceği öngörülmüştür.

$$AKH = B.\Delta f + \Delta P_{ara} \quad (5)$$

$$u = K_p.(B.\Delta f + \Delta P_{ara}) + \frac{1}{K_i} \int (B.\Delta f + \Delta P_{ara}) dt \quad (6)$$

Belirtilen kaynakta, araştırmacılar aynı model üzerinde kontrol amacıyla yine bir PI-kontrolör kullanmışlar, kontrolörün kazançlarını Parçacık Sürüsü algoritması ile ayarlamışlardır.

Optimizasyonda maliyet fonksiyonu olarak Hatanın Karelerinin İntegrali (HKİ) (Integral of Square of Error - ISE) ve Hatanın Zamanla çarpılmış Mutlak değerlerinin İntegrali (HZMİ) (Integral of Time multiplied Absolute of Error - ITAE) fonksiyonlarını kullanmışlardır [9]. Bu çalışmada, yukarıda sözü edilen kaynakta daha iyi sonuç verdiği belirtilen HZMİ fonksiyonu ile elde edilen sonuçlar kullanılmıştır (Eş.7).

$$HZMİ = \int_0^{\infty} t.|e|.dt \quad (7)$$

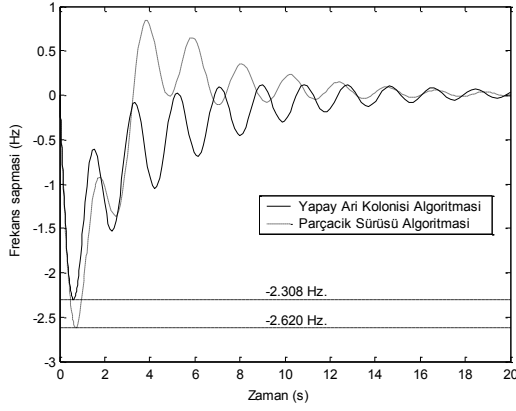
Çalışmada elde edilen sonuçlar ile literatür sonuçlarının karşılaştırılması Şekil 4'te ve Şekil 5'te gösterilmiştir. Ayrıca elde edilen optimum kontrolör parametreleri ve maksimum aşım değerleri Çizelge 2'de verilmektedir [10]. Metotların karşılaştırılmasında frekans sapsması ve ara-bağlantı hattı güç değişimlerinin maksimum aşım değerleri, performans kriteri olarak kabul edilmiştir.

Çizelge 2. Optimum kontrolör kazançları ve maksimum aşım değerleri

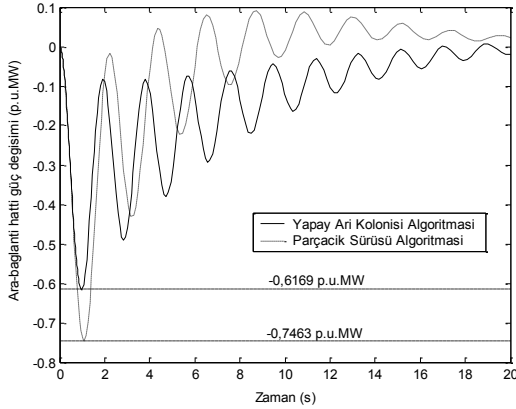
Kazanç Güncelleme Algoritması	PI-Kontrolör Kazançları		Maksimum Aşım	
	K_p	K_i	ΔP_{ara} (p.u.MW)	Δf_i (Hz)
Yapay Arı Kolonisi	0,601	0,201	-0,616	-2,308
Parçacık Sürüsü [Abdel-Magid ve Abido, 2003]	-0,036	0,490	-0,746	-2,620

Elde edilen sonuçlardan da görülmektedir ki, aynı yük altında (1 p.u.MW), Yapay Arı Kolonisi algoritması

ile optimize edilen PI-kontrolün geçiş cevabı analizinden elde edilen sonuçlar, literatür ile karşılaştırıldığında, frekans sapması aşımı için %11.9, ara-bağlantı hattı güç değişimi aşımı için %17.3 oranında azalma sağlandığı görülmüştür.



Şekil 4. Frekans sapması eğrileri



Şekil 5. Ara-bağlantı hattı güç değişimi eğrileri

Buna göre, Yapay Arı Kolonisi algoritması ile anma frekansının ve ara-bağlantı hattı gücünün daha az sapması sağlamıştır. Bu sonucun; Yapay Arı Kolonisi algoritmasının arama performansının, Parçacık Sürüsü algoritmasına göre üstünlüğünden kaynaklandığı değerlendirilmiştir. Zira Parçacık Sürüsü algoritmasında arama işlemi, bir iterasyon süresinde tek bir sürü ile bir kez gerçekleştirilirken; Yapay Arı Kolonisi algoritmasında arama işlemi, bir iterasyon süresi içinde işçi arı sürüsü, gözcü arı sürüsü ve kâşif arı sürüsü ile ardışık olarak üç kez yapılmaktadır. Ayrıca, Parçacık Sürüsü algoritmasında yerel ve global arama işlemleri tek bir güncelleme formülü içerisinde birleşik yapılırken; Yapay Arı Kolonisi algoritması ile bir iterasyon süresinde işçi ve gözcü arılar ile yerel, kâşif arılar ile global aramalar ayrı işlemler olarak gerçekleştirilmektedir.

Bununla beraber; eğrilerde oluşan sönümlü osilasyonların, model sistemde olduğu gibi çok mertebeli proseslerin kontrolünde PI-kontrolün sınırlı geçiş cevabı performansı nedeniyle meydana geldiği değerlendirilmiştir.

SONUÇ

Bu çalışmada, son yıllarda literatüre kazandırılan Yapay Arı Kolonisi algoritması, iki bölgesi bir enterkonnekte güç sisteminin yük-frekans kontrolüne uygulanmıştır. Algoritmanın, yük-frekans kontrolünde her iki kontrol bölgesinde ikincil kontrolü sağlayan PI-kontrolörlerin kazançlarını ayar performansı, benzer bir optimizasyon algoritması olan Parçacık Sürüsü algoritmasının literatürde mevcut sonuçları ile karşılaştırılmıştır. Sonuçta, enerji kalitesi açısından Yapay Arı Kolonisi optimizasyon algoritmasının güç sisteminin kontrol cevabında iyileştirici etki yaptığı gözlenmiştir.

KAYNAKLAR

- [1] Bergen, R., Vittal, V., "Power System Analysis 2th ed.", Prantice Hall, USA, 375-400 (2000).
- [2] Elgerd, O.I., Fosha, C.E., "Optimum megawatt-frequency control of multi-area electric energy systems", IEEE Transactions on Power Apparatus and Systems, 89(4):556-563 (1970).
- [3] Kundur, P., Power System Stability and Control, McGraw-Hill, Delhi, India, 1993.
- [4] Weedy, B.M., "Electric Power Systems 3th ed.", John Wiley&Sons, USA, 148-172 (1987).
- [5] Elgerd, O.I., "Electric Energy Systems Theory", Mc Graw-Hill Press., New York, 299-342 (1982).
- [6] Karaboğa, D., "An idea based on honey bee swarm for numerical optimization", Technical Report TR06, Erciyes University, Turkey, 1-6 (2005).
- [7] Karaboga, D., Basturk, B., "A powerful and efficient algorithm for numeric optimization: artificial bee colony (abc) algorithm", Journal Of Global Optimization, 39(3): 459-471 (2007)
- [8] Abdel-Magid, Y.L., Abido, M.A., "AGC tuning of interconnected reheat thermal systems with particle swarm optimization", 10th IEEE International Conference on Electronics, Circuits and Systems, United Arab Emirates, 376-379 (2003).
- [9] Aström, K.J., Hagglund, T., "PID Controllers", Instrument Society of America, New York, 64-70, 232-235 (1995).
- [10] Gözde, H., "Güç Sistemlerinde Yapay Arı Kolonisi Algoritması Yöntemi ile Yük-Frekans Kontrolü Optimizasyonu", Doktora Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü (2010).