

# The Estimation of Bessel Type Low-Pass Active Filter Parameters with Charged System Search Algorithm

Bahadır Hiçdurmaz<sup>a</sup>, Fırat Ertaç Durak<sup>a</sup>, Serdar Özyön<sup>a,1</sup>

<sup>a</sup>Kütahya Dumlupınar Üniversitesi, Mühendislik Fakültesi, Elektrik Elektronik Mühendisliği, 43100, Kütahya, Türkiye

---

## Abstract

Filter circuits are one of the significant elements of communication systems. These circuits are ones that provide the characteristic of decay at a demanded level or a delay at a demanded time in a defined frequency area. They are also accepted as selective devices that pass or block the values under or over a defined frequency and can be designed for many different purposes. In this study the design of a 10th degree Sallen-Key structure Bessel type low pass filter (SK-B-AGF) whose component selections and gain calculations take a long time with traditional methods, has been done. And what the component values of the designed filter will be, has been optimized for continuous values by using charged system search algorithm (CSS). In the case where constant values have been used, the total error has been minimized by accepting the component values as ideal and unlimited. The obtained optimum filter component values together with have been presented for each stage the quality factor (Q), the results have been evaluated.

**Keywords:** “Active filters, Low-pass filters, Bessel type, Charged system search algorithm”

---

## 1. Giriş

Elektronik filtreler, belirli bir frekans bandını geçirip bu bandın dışındaki bütün işaretleri zayıflatmak ya da yok etmek için tasarlanan devrelerdir. Filtre devreleri aktif ya da pasif olarak tasarlanabilirler. Pasif filtre devreleri direnç, bobin ve kapasitör gibi devre elemanları içerirken, aktif filtreler bu devre elemanlarıyla birlikte işlevsel yükselteçler (op-amp'lar) ve transistörler gibi aktif devre elemanları da içerirler. Aktif filtrelerin maliyet değerleri pasif filtrelere göre daha avantajlıdır. Bu filtreler düşük maliyetlerinden dolayı özellikle düşük frekanslarda tercih edilirler. Geniş bir frekans bölgesine ayarlanabilen aktif filtreler boyut ve ağırlık bakımından çok pratiktir ve tasarımları da kolaydır. Bu tür filtrelerin bir diğer avantajı da birbirlerini etkilemeden kolayca kaskatlanabilmeleridir. Aktif filtre tasarlamının birçok yolu vardır. Bu tasarımlardan sıklıkla kullanılanlardan biri R.P. Sallen ve E.L. Key tarafından tanıtılan Sallen-Key topolojisidir [1]. Bu yapıdaki devreler yaygın olarak kullanılan genel amaçlı devrelerden biridir. Bu topolojinin kararlı çalışma ve az elemana ihtiyaç duyma gibi avantajları vardır. Bu avantajlarının yanında Sallen-Key topoloji dört ya da daha fazla kutba sahip kaskatlanmış filtreler için önemli bir özellik olan düşük empedans özelliğine de sahiptir. Bu yapının da Bessel, Butterworth, Chebyshev ve Elliptic gibi farklı aktif filtre tipleri bulunmaktadır [2]. Klasik yöntem kullanılarak aktif filtrelerin tasarlanması imalat kısıtlamaları nedeniyle çok zordur. Bu nedenle bu tür filtreler tasarlanırken genellikle, komponentlerin değerleri birbirine eşit olacak şekilde seçilir. Bu durum tasarımın esnekliğini sınırlamaktadır. Bunun yanında giderek yaygınlaşan bir diğer yöntem ise filtre parametrelerinin bir meta-sezgisel algoritma ile önceden belirlenmesi ve kısa sürede belirlenen bu komponent değerleri kullanılarak daha esnek yapıda filtreler oluşturmaktır [3].

Son yıllarda meta-sezgisel algoritmalarla farklı yapılar da filtre tasarımı için komponent değeri belirleme ve optimizasyonu uygulamaları giderek artmaktadır. Bu algoritmalarından bazıları geri-izleme arama algoritması [2], parçacık sürü optimizasyonu [3-7], yüklü sistem arama algoritması [3], ortalama diferansiyel gelişim algoritması [3], gri-kurt arama algoritması [7,8], armoni arama algoritması [9], diferansiyel gelişim algoritması [3,5,9,10], klon seçim algoritması [11], tabu arama algoritması [12], genetik algoritma [4,13,14], girdap arama algoritması [15], bakteri yiyecek optimizasyonu algoritması [16] ve yapay arı koloni algoritması [4,5,17] olarak verilebilir. Bütün bu çalışmalarda farklı yapıda ve tipteki birçok filtrenin komponent değerleri meta-sezgisel algoritmalarla başarıyla belirlenmiştir.

---

<sup>1</sup> Sorumlu Yazar. Tel.: +90-274-265-2031; fax: +90-274-265-2013.  
E-posta adresi: serdar.ozyon@dpu.edu.tr

Bu çalışmada literatürde birçok mühendislik probleminin çözümüne başarıyla uygulanmış yüklü sistem arama algoritması (CSS) ile 10. dereceden Sallen-Key yapıda Bessel tip alçak geçiren (SK-B-AGF) bir filtrenin devre elemanlarının değerleri belirlenmiştir. Algoritma ile ağırlıklı toplam metodu (ATM) ile birleştirilen filtrenin her bir katına ait frekans ölçeklendirme faktörleri (FSF) ve kalite faktörlerinden (Q) oluşan toplam hata oranı (amaç fonksiyonu) minimize edilmiştir.

## 2. Yüklü Sistem Arama Algoritması (CSS)

Yüklü sistem arama algoritması (CSS), tek düze hacimsel yük yoğunluğuna sahip bir  $a$  yarıçaplı yüklü küre olarak düşünülen her bir ajanın diğer ajanlara bir elektrik kuvveti uygulayabildiği popülasyon-tabanlı bir arama yaklaşımıdır. Küre içerisinde yerleştirilen ajanlar için kuvvetlerin büyüklüğü, ajanlar arasındaki ayrılma mesafesi ile doğru orantılı iken küre dışına yerleştirilen bir ajan için kuvvetin büyüklüğü ajanlar arasındaki mesafenin karesi ile ters orantılıdır. Ortaya çıkan kuvvetlerle ya da ivmelenme ile beraber kuvvet yasaları ajanların yeni yerini belirler. CSS algoritmasının çözüm adımları aşağıdaki gibi özetlenebilir [18,19]:

**1. Adım: Başlatma:** Ajanların ilk konumları arama uzayında rastgele belirlenir ve yüklü parçacıkların başlangıç hızlarının sıfır olduğu varsayılır. Ajanlar için uygunluk fonksiyonunun değerleri belirlenir ve ajanlar artan bir uygunluk değerleri sırasına göre sıralanır. Tüm ajanlar arasındaki en iyi ajan  $X_{eniye}$  ele alınacak ve ilgili uygunluğu eniyi uygunluk olacaktır. Benzer şekilde en kötü ajan ise en kötü uygunluğa sahip olacaktır. İlk ajanların bir kısmı uygunluk fonksiyonu ile ilgili değerleri yüklü hafıza (CM) olarak adlandırılan bir hafızada kaydedilir [18,19].

**2. Adım: Kuvvetleri Belirleme:** Aşağıdaki formülü kullanarak herbir ajanın kuvvet vektörünü hesaplayınız.

$$F_j = q_j \sum_{i,i \neq j} \left( \frac{q_i}{a^3} r_{ij} i_1 + \frac{q_i}{r_{ij}^2} i_2 \right) p_{ij} (X_i - X_j) \begin{cases} j = 1, 2, \dots, N \\ i_1 = 1, i_2 = 0 \Leftrightarrow r_{ij} < a \\ i_1 = 0, i_2 = 1 \Leftrightarrow r_{ij} > a \end{cases} \quad (1)$$

Burada  $F_j$ ,  $j$ . ajana etki eden sonuç kuvvettir.  $N$ , ajan sayısıdır. Herbir ajanın yükünün büyüklüğü ( $q_i$ ) çözüm kalitesi dikkate alınarak aşağıdaki gibi belirlenir.

$$q_i = \frac{fit(i) - fitworst}{fitbest - fitworst}, \quad i = 1, 2, \dots, N \quad (2)$$

Burada  $fitbest$  ve  $fitworst$  sırasıyla bütün ajanların en iyi ve en kötü uygunluğudur.  $fit(i)$ ,  $i$ . ajanın uygunluğunu ifade eder. İki yüklü parçacık arasındaki mesafe ( $r_{ij}$ ) aşağıdaki gibi verilir.

$$r_{ij} = \frac{\|X_i - X_j\|}{\|(X_i - X_j) / 2 - X_{eniye}\| + \varepsilon} \quad (3)$$

Burada  $X_i$  ve  $X_j$  sırasıyla  $i$ . ve  $j$ . ajanların konumlarıdır.  $X_{eniye}$ , mevcut en iyi ajanın konumudur.  $\varepsilon$ , tekilliklerden kaçınmak için küçük bir pozitif sayıdır. Burada  $p_{ij}$ , her bir ajani diğer ajanlara doğru taşıma olasılığıdır ve aşağıdaki ifade ile elde edilir.

$$p_{ij} = \begin{cases} 1, & \frac{fit(i) - fitbest}{fit(j) - fit(i)} > rand \vee fit(j) > fit(i) \\ 0, & \text{değilse} \end{cases} \quad (4)$$

Burada  $rand(0,1)$  arasında tek düze dağıtılan bir rasgele sayıdır.

Önceden söz edildiği gibi, herbir ajan tek düze hacimsel yük yoğunluğuna sahip  $a$  yarıçaplı bir yüklü küre olarak düşünülür.  $a$  için uygun bir değer, arama uzayının boyutu göz önüne alınarak aşağıdaki gibi belirlenir.

$$a = 0,01 \times \max \left( \{x_{i,max} - x_{i,min} \mid i = 1, 2, \dots, n_v\} \right) \quad (5)$$

Burada  $x_{i,max}$  ve  $x_{i,min}$  sırasıyla  $i$ 'nci karar parametresinin alt ve üst sınırlarıdır.  $n_v$ , tasarım değişkenlerinin sayısıdır.

**3. Adım: Çözüm Oluşturma:** Herbir ajan ( $CP$ ) yeni konumlarına hareket eder ve yeni hızları aşağıdaki gibi hesaplanır.

$$X_{j,yeni} = rand_{j1} \cdot k_a \cdot \frac{F_j}{m_j} \cdot \Delta t^2 + rand_{j2} \cdot k_v \cdot V_{j,eski} \cdot \Delta t + X_{j,eski} \quad (6)$$

$$V_{j,yeni} = \frac{X_{j,yeni} - X_{j,eski}}{\Delta t} \quad (7)$$

Burada  $k_a$ , ivmelenme katsayısı,  $k_v$ , bir önceki hızın etkisini kontrol edecek hız katsayısı,  $rand_{j1}$  ve  $rand_{j2}$  (0,1) arasında tekdüze dağılılan rasgele iki sayıdır.

Önceki hızın ve bir ajanı etkileyen sonuç kuvvetinin etkisi sırasıyla  $k_v$  ve  $k_a$  değerleri ile kontrol edilebilir. Erken iterasyonlarda aşırı arama keşif yeteneğini geliştirebilir. Bununla birlikte, başarımlı kabiliyetini arttırmak için kademeli olarak azaltılmalıdır.  $k_a$ , çeken kuvvetlerle ilgili bir parametre olduğundan başarımlı özelliğinin kontrol parametresi gibi çalışır. Bu nedenle doğrusal bir artımsal fonksiyon seçilmesi algoritmanın performansını arttırabilir. Ayrıca, bir ajanın önceki hızının yönü sonuçta ortaya çıkan kuvvetle aynı olması gerekmez. Bu,  $k_v$  hız katsayısının araştırma işlemini kontrol ettiğini ve dolayısıyla doğrusal bir azaltma işlevinin seçilebileceğini göstermektedir [18,19].

**4. Adım: Güncelleme İşlemi:** İzin verilen arama alanından yeni bir ajan çıkarsa, uyum-arama tabanlı işleme yaklaşımı ajanın konumunu düzeltmek için kullanılır. Bu mekanizmaya göre, değişken sınırlarını ihlal eden çözüm vektörünün herhangi bileşeni yüklü hafızadan ( $CM$ 'den) ya da olası değer aralığından rastgele bir değer seçilerek yeniden üretilebilir. Bu mekanizma sadece çözümlerinin pratik olması için izin verilen arama alanından gelen ajanlar ( $CP$ 'ler) için kullanılacaktır. Ayrıca, bazı yeni  $CP$  vektörleri,  $CM$ 'deki en kötülerinden daha iyiyse,  $CM$ 'deki en kötü olanlar yerine dahil edilirler.

**5. Adım: Sonlandırma Kriterini Kontrol Etme:** 2'den 4'e kadar olan adımlar sonlandırma kriteri yerine getirilinceye kadar devam ettirilir. Sonlandırma kriteri aşağıdaki koşullardan biridir.

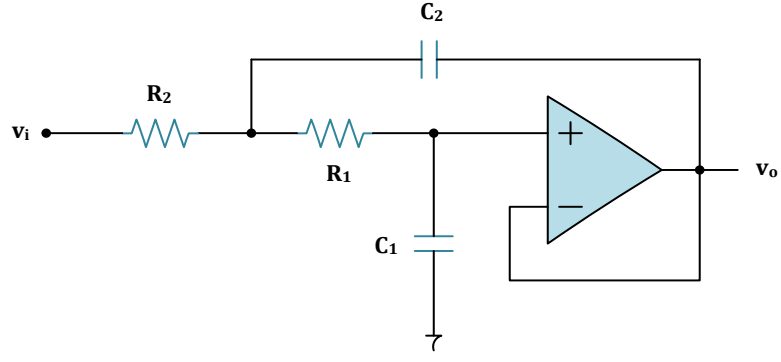
1).  **$CP$ 'lerin maksimum mesafesi:**  $CP$ 'ler arasındaki maksimum mesafe önceden belirlenmiş bir değerden daha azdır.

2). **Maksimum iterasyon sayısı:** Optimizasyon işlemi belirlenen sayıda iterasyonun tamamlanmasının ardından sona erer.

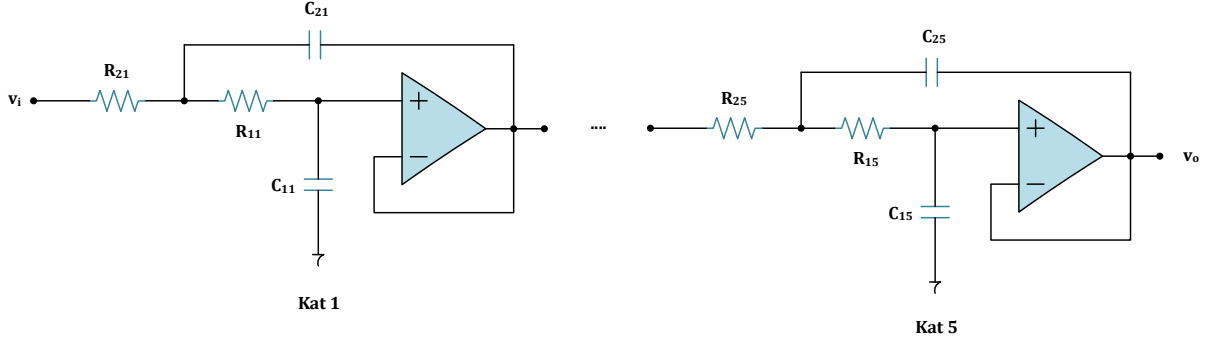
Pek çok sezgisel algoritma için, bütün ajanlar küçük bir alanda toplanırsa, yani, ajanlar arama alanının bir parçası olarak tutulursa, bundan kaçmanın çok zor olabileceği ortak bir özelliktir. CSS algoritması için geçerli kuvvetler, çeken kuvvetler olduğundan dolayı, yukarıdaki problem bu yöntem için çözülmemiş gibi görünüyor. Bununla birlikte, keşif ve başarımlı arasında iyi bir denge kurmak ve kendi kendine uyum, iş birliği ve rekabet içeren üç adımı göz önünde bulundurmak bu sorunu çözebilir. Bu üç temel kavram CSS algoritmasında dikkate alınmaktadır. İyi  $CP$ 'lere doğru ilerlemek kendi kendine adaptasyon adımını sağlar. Her  $CP$ 'ye etki eden sonuç kuvvetini belirlemek için iş birliği yapan  $CP$ 'ler, iş birliği basamağını sağlar ve kötü bir  $CP$ 'ye kıyasla iyi bir  $CP$  için daha büyük bir güce sahiptir ve  $CM$ 'de iyi  $CP$ 'lerin kaydedilmesi rekabet basamağını sağlar [18,19].

### 3. Sallen-Key Yapıda Bessel Tip Alçak Geçiren Filtre Tanımı (SK-B-AGF)

Belli bir frekans değeri altında kalan bileşenlerin geçişine izin veren, üzerindeki bileşenlerin geçişini engelleyen elektronik cihazlar alçak geçiren aktif filtre olarak tanımlanır. Bu filtreler literatürde farklı yapılarda tasarlanırlar. Bu çalışmada ele alınan Sallen-Key yapıda 2.dereceden alçak geçiren aktif filtrenin devre şeması Şekil 2'de verilmiştir. 10. dereceden bir alçak geçiren aktif filtre tasarımı ise beş adet 2.dereceden devrenin art arda bağlanması ile oluşturulur ve 10. dereceden alçak geçiren aktif filtrenin devre şeması ise Şekil 3'de verilmiştir [1-3,20].

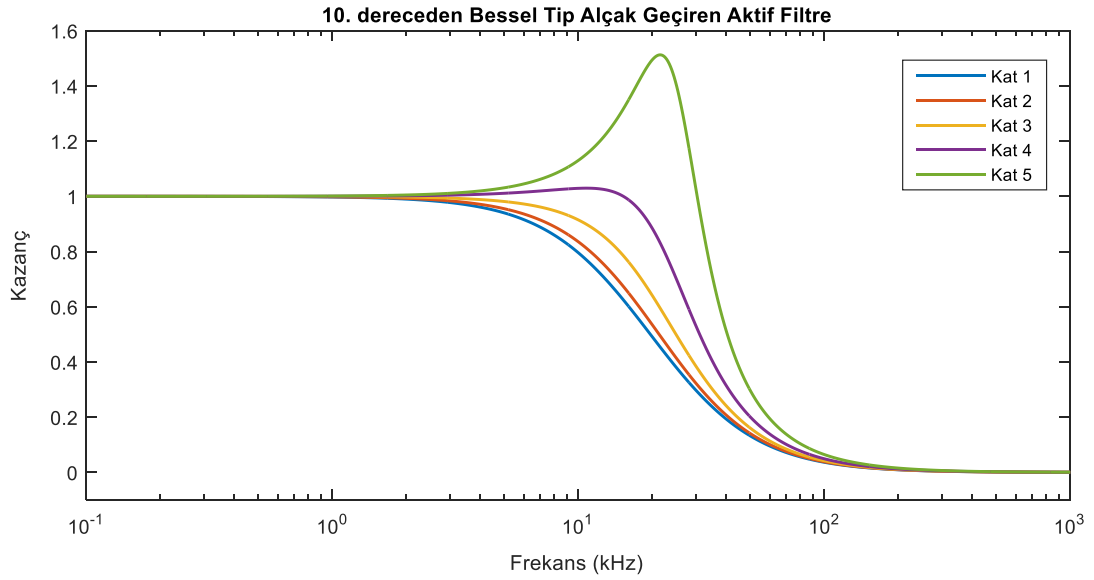


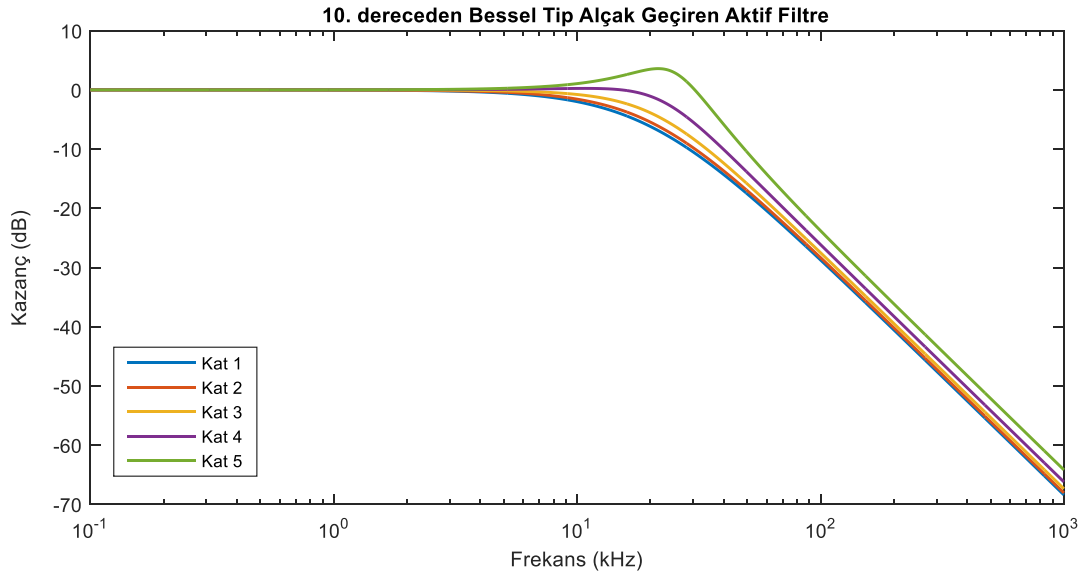
Şekil 2. 2. dereceden Sallen-Key yapıda AGF devre şeması



Şekil 3. 10. dereceden Sallen-Key yapıda AGF devre şeması

Alçak geçiren aktif filtreler farklı amaçlara uygun şekilde farklı tiplerde dizayn edilebilirler. Bu çalışmada Bessel tip bir alçak geçiren filtre tasarımı ele alınmıştır. Bu filtre tasarımında her kata ait 10 kHz kesim frekansı için kazanç eğrileri Şekil 4'de verilmiştir [1,20].





Şekil 4. 10. dereceden Bessel tip AGF kazanç eğrileri

Şekil 2’de verilen devrenin transfer fonksiyonu denklem (12)’de verilmiştir [1-3,20].

$$H_{AGF}(s) = \frac{V_o(s)}{V_i(s)} = \frac{1}{1 + s(R_1 + R_2)C_1 + s^2 R_1 R_2 C_1 C_2}, \quad s = j2\pi f \quad (8)$$

$$H_{AGF}(f) = \frac{1}{1 - (2\pi f)^2 R_1 R_2 C_1 C_2 + j2\pi f (R_1 + R_2) C_1}$$

Transfer fonksiyonunun standart formu denklem (13)’te tanımlanmıştır [1-3,20].

$$H_{AGF}(f) = \frac{1}{1 - \left(\frac{f}{FSF \cdot f_c}\right)^2 + j \frac{f}{Q \cdot FSF \cdot f_c}} \quad (9)$$

$$FSF = \frac{1}{2\pi f_c \sqrt{R_1 R_2 C_1 C_2}}; \quad Q = \frac{\sqrt{R_1 R_2 C_1 C_2}}{(R_1 + R_2) C_1}$$

Denklemden  $f_c$  kesim frekansını,  $FSF$  frekans ölçeklendirme faktörünü ve  $Q$  kalite faktörünü göstermektedir. Tasarlanan filtrenin genlik tepkisi ise denklem (14)’de tanımlanmıştır. 10. dereceden Bessel tip bir filtre tasarlamak için her kattaki  $FSF$  ve  $Q$  değerleri Tablo 1’de verilmiştir [20].

$$H_{AGF}(f) = \frac{1}{\sqrt{\left(1 - \left(\frac{f}{FSF \cdot f_c}\right)^2\right)^2 + \left(\frac{f}{Q \cdot FSF \cdot f_c}\right)^2}} \quad (10)$$

Tablo 1. 10. dereceden Bessel tip filtre için  $FSF$  ve  $Q$  değerleri

Filtre Derecesi	Kat 1		Kat 2		Kat 3		Kat 4		Kat 5	
	FSF	Q	FSF	Q	FSF	Q	FSF	Q	FSF	Q
10	1,9490	0,5040	1,9870	0,5380	2,0680	0,6200	2,2110	0,8100	2,4850	1,4150

CSS algoritması ile Şekil 3’de verilen devre şemasının her bir kata ait her bir pasif bileşeni, Tablo 2’de gösterildiği gibi dizi formunda kodlanmıştır. Bu durumda, filtrenin bileşen değeri, hata en aza indirilmeye kadar art arda CSS tarafından ayarlanır. Filtrenin tasarım hatası, FSF ve Q'nun maliyet fonksiyon hatalarının toplamıdır ve denklem (26)’da verilmiştir. Denklemde yer alan  $FSF_{t,i}$  hedeflenen FSF değeri,  $Q_{t,i}$  ise hedeflenen Q değeridir. Burada amaç toplam hatayı minimize etmektir. FSF ve Q iki farklı amaç değeri olduğu için bu değerler literatürde birçok uygulaması olan ağırlıklı toplam metodu ile tek bir amaç fonksiyonuna dönüştürülmüştür ( $Hata_{toplam}$ ) [2,3].

$$Hata_1 = \sum_{i=1}^5 \frac{|FSF_{t,i} - FSF_i|}{FSF_{t,i}}, \quad Hata_2 = \sum_{i=1}^5 \frac{|Q_{t,i} - Q_i|}{Q_{t,i}} \quad (11)$$

$$Hata_{toplam} = 0.5 \times Hata_1 + 0.5 \times Hata_2$$

**Tablo 2. Komponent değerlerinin algoritmada dizi biçiminde temsili**

Kat 1				...	Kat 5			
R <sub>11</sub>	C <sub>11</sub>	R <sub>21</sub>	C <sub>21</sub>	...	R <sub>15</sub>	C <sub>15</sub>	R <sub>25</sub>	C <sub>25</sub>

Problemin çözümünde kullanılan CSS algoritmasına ait parametre değerleri iterasyon sayısı 1000, popülasyondaki parçacık sayısı 40, boyut (bilinmeyen sayısı) 20, başlangıç  $k_v$  0,8, bitiş  $k_v$  0,2, başlangıç  $k_a$  0,8, bitiş  $k_a$  0,2 ve  $\varepsilon$  1e-6 olarak alınmıştır. Problem bu parametrelerle CSS algoritması ile 30 kez çözülmüştür. Tasarlanan filtre için CSS tarafından bulunan komponent değerleri Tablo 3’de, tasarlanan filtre için bulunan bu komponent değerleriyle bütün katmanlar için hesaplanan FSF ve Q değerleri Tablo 4’de ve 30 çözümden en iyi hata değerinin elde edildiği çözüme ait istatistiki değerler ise Tablo 5’de verilmiştir.

**Tablo 3. CSS tarafından eniyi çözümde belirlenen komponent değerleri**

	Komponent	Kat 1	Kat 2	Kat 3	Kat 4	Kat 5
CSS	R <sub>1</sub> (kΩ)	1,547604	6,703285	1,326916	5,869937	5,588178
	R <sub>2</sub> (kΩ)	1,105843	7,109156	8,039456	4,147107	1,845830
	C <sub>1</sub> (nF)	6,106146	1,077878	1,325276	0,887170	0,608856
	C <sub>2</sub> (nF)	6,381104	1,249020	4,189503	2,399260	6,531474

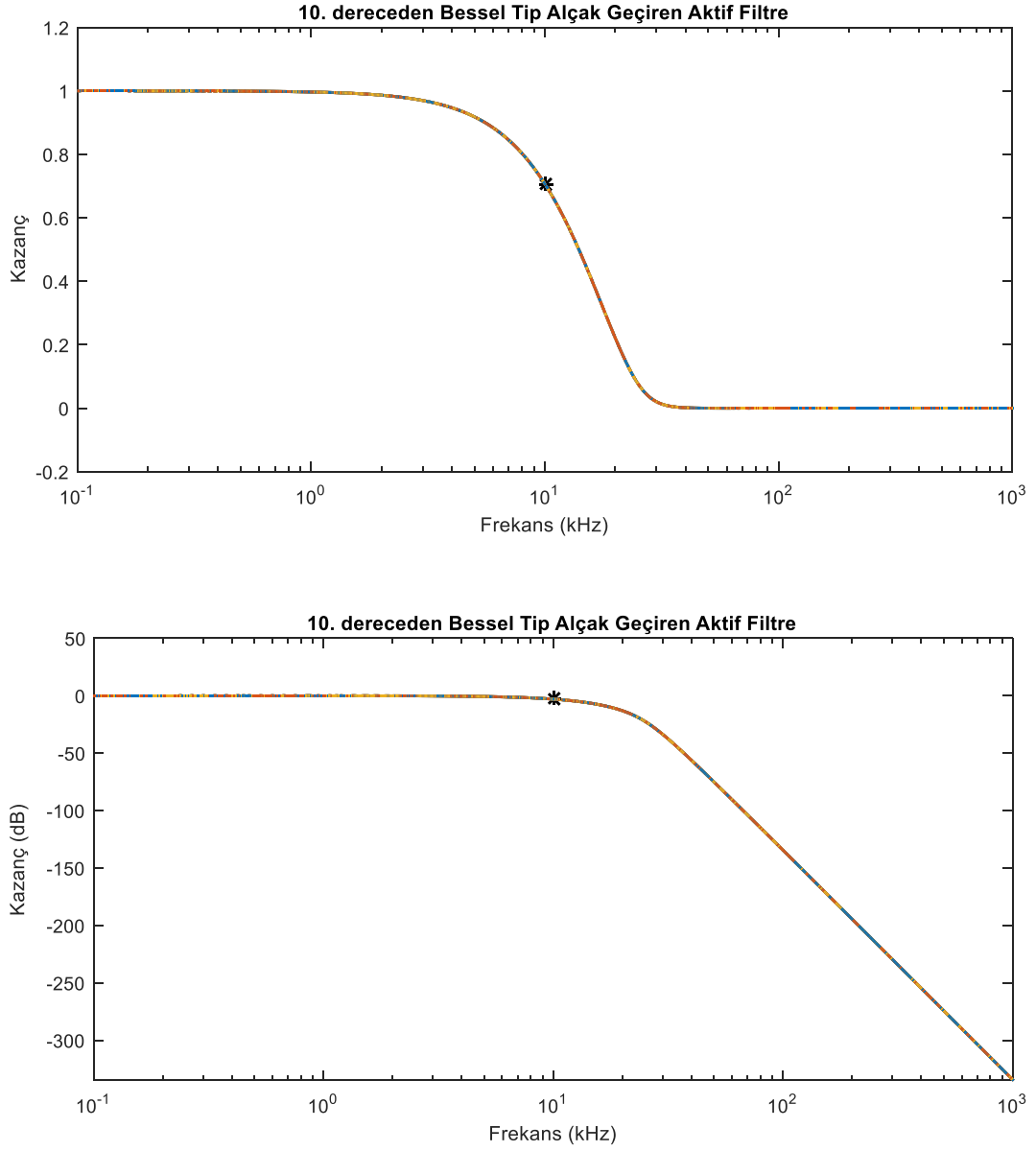
**Tablo 4. CSS tarafından eniyi çözümde belirlenen FSF ve Q değerleri**

	Kat 1		Kat 2		Kat 3		Kat 4		Kat 5	
	FSF	Q	FSF	Q	FSF	Q	FSF	Q	FSF	Q
<b>Hedef</b>	1,9490	0,5040	1,9870	0,5380	2,0680	0,6200	2,2110	0,8100	2,4850	1,4150
<b>CSS</b>	1,9490	0,5040	1,9870	0,5380	2,0680	0,6200	2,2110	0,8100	2,4850	1,4150

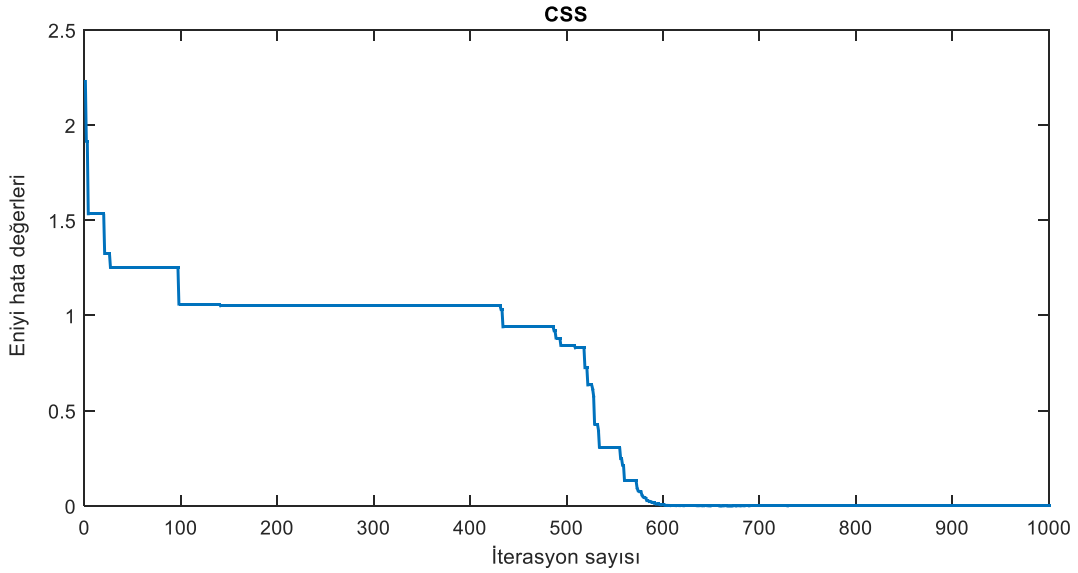
**Tablo 5. 30 çalışmada elde edilen değerler**

	Eniyi Hata	Ortalama Hata	Enkötü Hata	Std Sapma	Süre (sn)
<b>CSS</b>	5,329360e-09	3,800057e-02	7,508000e-01	1,349954e-01	6,90889

Tasarlanan filtrenin CSS algoritmasıyla çözümünde elde edilen kazanç eğrileri Şekil 5’de verilmiştir. Algoritma ile elde edilen en iyi çözüm değerine ait yakınsama eğrisi ise Şekil 6’da görülmektedir.



Şekil 5. CSS ile elde edilen 10. dereceden Bessel tip AGF kazanç eğrileri



Şekil 6. CSS ile elde edilen iterasyon sayısına göre yakınsama eğrileri

#### 4. Sonuçlar

Çalışmada elektronik mühendisliğinde büyük öneme sahip aktif filtrelerden olan 10. dereceden Sallen-Key yapıda Bessel tip alçak geçiren bir filtrenin tasarımı yapılmıştır. Tasarımda filtre devre elemanları yüklü sistem arama algoritması ile sürekli durum için belirlenmiş ve bu değerlerle FSF ve Q faktörleri hesaplanmıştır. CSS algoritması ile devre komponent değerleri başarıyla belirlenmiştir. Algoritmanın probleme uygulanmasında, art arda bağlanan beş adet 2. dereceden aktif filtrenin yirmi adet devre elemanı, teorik olarak hesaplanmış olan FSF ve Q değerlerini minimum hata ile yakalamak üzerine kurulmuştur. CSS algoritması ile yapılan 30 ayrı aramada elde edilen hata değerlerine bakıldığında, başarı oranı oldukça yüksektir. Eniyi çözüm değeri incelendiğinde algoritma yaklaşık 600. iterasyonda eniyi hata değerine yakınsamıştır. Bundan sonra yapılacak çalışmalarda farklı yapı ve tiplerde filtrelerin tasarlanması planlanmaktadır. Bunun yanında bu çalışmada ele alınmayan kesikli durum için yani komponent değerlerinin belirli bir üretim serisinden seçilmesi durumuna farklı optimizasyon algoritmaları ile çözüm aranacaktır.

#### Bilgilendirmeler

Bu çalışma Tokat Gaziosmanpaşa Üniversitesi ve İstanbul Gedik Üniversitesi destekleriyle Ankara’da düzenlenen 4. Uluslararası Bilimsel ve Mesleki Çalışmalar Kongresi-Mühendislik (Bilmes Eng 2019) sempozyumunda özet metin bildiri olarak sözlü sunulmuştur.

#### 5. Referanslar

- [1] R.P. Sallen, E.L. Key, “A practical method of designing RC active filters” IRE Transactions on Circuit Theory 2(1) (1955) 74-85.
- [2] B. Hiçdurmaz, B. Durmuş, H. Temurtaş, S. Özyön, “The prediction of butterworth type active filter parameters in low pass sallen key topology by backtracking search algorithm” Proceedings of 2nd International Conference on Engineering and Natural Sciences 9 (2016) 2422-2428.
- [3] B. Durmuş, H. Temurtaş, S. Özyön, “Optimizasyon algoritmalarının ile çoklu geri-beslemeli yüksek geçiren aktif filtre tasarımı” Mühendislik Alanında Araştırma ve Değerlendirmeler, Editör: Dr. Mahmut TURAN, p.123-147, Gece Akademi, 2019, Ankara. ISBN: 978-605-7852-96-0.
- [4] R.A. Vural, T. Yıldırım, T. Kadioğlu, A. Basargan, “Performance evaluation of evolutionary algorithms for optimal filter design” IEEE Transactions on Evolutionary Computation 16 (2012) 135-147.



- [5] D. Ustun, M. Akkus, M.B. Bicer, H. Temurtas, A. Akdagli, "Sezgisel algoritmalar ile Butterworth ve Chebyshev alçak geçiren filtre tasarımı" Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SIU-2015) (2015) 108-111.
- [6] B.P. De, R. Kar, D. Mandal, S.P. Ghoshal, "Optimal selection of components value for analog active filter design using simplex particle swarm optimization" International Journal of Machine Learning and Cybernetics 6(4) (2015) 621-636.
- [7] B. Nayak, T.R. Choudhury, B. Misra, "Component value selection for active filters based on minimization of GSP and E12 compatible using Grey Wolf and Particle Swarm Optimization" AEÜ - International Journal of Electronics and Communications 87 (2018) 48-53.
- [8] G.G. Bulut, H. Güler, M.T. Özdemir, "Optimal selection of components in a sixthorder Butterworth low-pass filter with using grey wolf algorithm" International Journal of Electrical, Electronics and Data Communication 5(10) (2017) 1-4.
- [9] R.A. Vural, U. Bozkurt, T. Yildirim, "Analog active filter component selection with nature inspired metaheuristics" AEÜ - International Journal of Electronics and Communications 67(3) (2013) 197-205.
- [10] A.F. Sheta, "Analogue filter design using differential evolution," International Journal of Bio-Inspired Computation 2(3) (2010) 233-241.
- [11] M. Jiang, Z. Yang, Z. Gan, "Optimal components selection for analog active filters using clonal selection algorithm" Proceedings of International Conference on Intelligent Computing (2007) 950-959.
- [12] A. Kalinli, "Component value selection for active filters using parallel tabu search algorithm" AEÜ - International Journal of Electronics and Communications 60 (2006) 85-92.
- [13] T. Kaya, H. Guler, "A hybrid genetic algorithm for analog active filter component selection" AEÜ - International Journal of Electronics and Communications 86 (2018) 1-7.
- [14] D.H. Horrocks, M.C. Spittle, "Component value selection for active filters using genetic algorithms" Proceedings IEEE Workshop on Natural Algorithms in Signal Processing 1(13) (1993) 1-6.
- [15] B. Doğan, T. Ölmez, "Vortex search algorithm for the analog active filter component selection problem" AEÜ - International Journal of Electronics and Communications 69(9) (2015) 1243-1253.
- [16] S. Gholami-Boroujeny, M. Eshghi, "Non-linear active noise cancellation using a bacterial foraging optimisation algorithm" IET Signal Processing 6 (2012) 364-373.
- [17] D. Bose, S. Biswas, A.V. Vasilakos, S. Laha, "Optimal filter design using an improved artificial bee colony algorithm" Information Sciences 281 (2014) 443-461.
- [18] A. Kaveh, S. Talahatari, "A novel heuristic optimization method: charged system search" Acta Mechanica 213(3-4) 2010 267-289.
- [19] R. Sheikholeslami, A. Kaveh, A. Tahershamsi, S. Talatahari, "Application of charged system search algorithm to water distribution networks optimization" International Journal of Optimization in Civil Engineering 4(1) 2014 41-58.
- [20] R. Mancini, "Op Amps for Everyone - Design References" Texas Instruments 2002.