

T.C.
ERCIYES ÜNİVERSİTESİ
BİLİMSEL ARAŞTIRMA PROJELERİ
KOORDİNASYON BİRİMİ



**SEZGİSEL YÖNTEMLER İLE ADAPTİF GÜRÜLTÜ GİDERİMİ VE
KESTİRİM**

Proje No: FYL-2013-4821

Tez Projesi, Yüksek Lisans

SONUÇ RAPORU

Proje Yürütücüsü:
Mehmet Bahadır ÇETİNKAYA
Elektrik Elektronik Mühendisliği

Araştırmacı:
Cüneyt ÖZTÜRK
Türk Telekom Kayseri, Transmisyon Servisi

Ekim 2014

KAYSERİ

TEŐEKKÖR

Bu proje alıőmasına maddi destek veren Erciyes Üniöersitesi Bilimsel Araőtırma Projeleri Birimi'ne (Proje No: FYL-2013-4821) teőekkÖr ederiz.

İÇİNDEKİLER

	Sayfa No
TEŞEKKÜR	2
İÇİNDEKİLER	3
ÖZET	4
ABSTRACT	5
1. GİRİŞ	6
2. KONU VE KAPSAM	8
3. BULGULAR	14
4. SONUÇ VE ÖNERİLER	22
KAYNAKLAR	23

ÖZET

Sayısal yöntemlerin üstünlüklerinden dolayı işaretler büyük çoğunlukla sayısal olarak kaydedilmekte ve iletilmektedir. Bunun sonucu olarak, sayısal işaret işleme teori ve uygulamaları çok önemli bir çalışma alanı haline gelmiştir. Özellikle, sayısal işaret işlemenin en temel uygulamalarından birisi olan sayısal süzgeç tasarımına yönelik çalışmalar dikkat çekecek düzeyde artmıştır. Sonlu darbe cevaplı (Finite Impulse Response-FIR) sayısal süzgeçler teori ve uygulama açısından iyi bir şekilde geliştirilmiş olmalarına karşın, aynı durum tasarımlarında karşılaşılan bazı önemli problemlerden dolayı sonsuz darbe cevaplı (Infinite Impulse Response-IIR) sayısal süzgeçler için geçerli değildir. Hata yüzeylerinin tek-modlu olmasından dolayı FIR süzgeçlerin tasarımında türeve dayalı algoritmalar etkin şekilde kullanılabilirken, çok-modlu hata yüzeyine sahip IIR süzgeç tasarımında bu algoritmalar yerel minimumlara takılabilmektedir. Bu nedenle IIR süzgeç tasarımlarında yeni yöntemlerin geliştirilmesine ihtiyaç duyulmuştur. Sezgisel algoritmalar küresel araştırma yetenekleri sayesinde, FIR ve IIR sayısal süzgeçleri yüksek yakınsama hızıyla başarılı bir şekilde tasarlayabilmektedirler.

Bu çalışmada, en güncel sezgisel optimizasyon algoritmalarından biri olan yapay arı koloni (Artificial Bee Colony-ABC) algoritması ve bu algoritmanın geliştirilmiş bir versiyonu (Modified Artificial Bee Colony-M-ABC) adaptif IIR süzgeçlerin gürültü giderimi amacıyla tasarımlarına uygulanmıştır. M-ABC ve ABC algoritmalarının adaptif IIR süzgeçlerin tasarımlarındaki performansları parçacık sürüsü optimizasyon (Particle Swarm Optimization-PSO) ve diferansiyel gelişim (Differential Evolution-DE) algoritmaları ile mukayese edilmiştir.

Adaptif IIR süzgeç tasarımlarında en düşük hata değerleri ve en yüksek yakınsama hızları M-ABC algoritması tarafından gerçekleştirilen tasarımlarda elde edilmiştir. Ayrıca ABC, PSO ve DE algoritmalarının performanslarının benzer olduğu görülmüştür.

Anahtar Kelimeler : Yapay arı koloni algoritması; Adaptif süzgeç tasarımı; Gürültü giderimi.

ABSTRACT

Signals are mostly stored and transmitted digitally due to the superiority of digital methods. As a result of this, the theory and application of digital signal processing have become an important area of research. Especially, the studies intended for digital filter design which is one of the most common application of digital signal processing have attracted increasing attention. While FIR digital filters are well developed in terms of theory and application, the same is not true for digital IIR filters due to some important problems encountered in the design of these filters. Although the gradient based optimization algorithms are able to design FIR filters efficiently since their error surfaces are uni-modal, these algorithms can get stuck in local minimum of error surfaces which might be multi-modal in the design of IIR filters. Thus, it is required to improve new methods for the design of IIR filters. Due to their global search ability, the heuristic algorithms can successfully design the FIR and IIR filters with higher convergence speeds.

In this work, the artificial bee colony (ABC) algorithm which is one of the most novel heuristic algorithms and a modified version of this algorithm (M-ABC) are applied to the design of adaptive IIR filter structures for the aim of noise cancellation. The performance of the ABC and M-ABC algorithms in the design of adaptive IIR filters are compared to that of particle swarm optimization (PSO) and differential evolution (DE) algorithms.

In the design of adaptive IIR filters, the minimum error values and the highest convergence rates are obtained in the designs realized by M-ABC algorithm. Also, it is seen that the performance of the ABC, PSO and DE algorithms are similar to each other.

Keywords: Artificial bee colony algorithm; Adaptive filter design; Noise cancellation.

1. GİRİŞ

Fiziksel bir durum hakkında bilgi taşıyan, bir veya birden fazla değişkene bağlı olan fonksiyonlar işaret olarak adlandırılırlar. İletim kanalının karakteristik özelliklerinden dolayı oluşan bozulmalar ve iletim hattı boyunca harici etkenlerin neden olduğu gürültü bileşenleri, bilgi taşıyan işareti zayıflatmakta ve işaret üzerinde bozucu etkiler oluşturmaktadır. İşaret üzerindeki istenmeyen bileşenleri yok etmek için, süzgeç olarak adlandırılan yapılar kullanılmaktadır. Süzgeçleme, işaretin belirli frekanslarının güçlendirildiği, zayıflatıldığı veya işaretin bazı arzu edilen özel operasyonlara tabii tutulduğu bir işlemdir [1]. Süzgeçlerin hem analog hem de sayısal formda gerçekleştirilmesi mümkündür. Sayısal süzgeçler güvenilir ve düşük maliyetli olmalarının yanı sıra bozulmalardan daha az etkilendikleri ve daha esnek yapıda oldukları için analog süzgeçlere göre daha çok tercih edilmektedirler [2]. Sayısal süzgeçler iki ana grupta incelenebilir: Sonlu darbe cevaplı (Finite Impulse Response-FIR) sayısal süzgeçler ve sonsuz darbe cevaplı (Infinite Impulse Response-IIR) sayısal süzgeçler. FIR süzgeçlerin hata yüzeyleri konveks yapıdadır, bu nedenle hata yüzeylerinde en iyi çözümü temsil eden tek bir minimum noktaları (küresel minimum) bulunur, yani tek-modlu ve kuadratik bir hata yüzeyine sahiptirler. Ayrıca, geri beslemesiz bir yapıya sahip oldukları için daima kararlıdır [3]. IIR süzgeçlerin hata yüzeyleri ise süzgeç katsayılarına göre çok-modludur ve kuadratik değildir. Bu nedenle en iyi çözümleri temsil eden küresel minimumların yanı sıra daha kötü çözümleri temsil eden yerel minimum noktalarını da içerebilmektedir. Ayrıca, adaptasyon süreci boyunca süzgeç sürekli olarak kararlı kalmayıp kararsız bir davranışta sergileyebilmektedir [4,5]. Hata yüzeylerinin analitik tanımlamalarının basit olmasından ve daima kararlı yapıda olmalarından dolayı, sonlu darbe cevaplı süzgeçlerin teori ve uygulamaları iyi bir şekilde geliştirilmiştir [6]. FIR süzgeçlerin önemli avantajlarına ve IIR süzgeçlerin tasarımında karşılaşılan zorluklara rağmen, aynı dereceden FIR süzgeçlere göre performanslarının belirgin derecede iyi olmasından ve donanım maliyetini düşürmelerinden dolayı son yıllarda IIR süzgeç tasarımı önemli bir çalışma alanı haline gelmiştir [6,7].

Hata yüzeyleri tek-modlu ve kuadratik olduğundan dolayı FIR süzgeçlerin tasarımında türeve dayalı algoritmalar etkin şekilde kullanılabilirken, kuadratik olmayan çok-modlu hata yüzeylerine sahip IIR süzgeçlerin tasarımında kullanıldıklarında yerel minimumlara takılabilmekte ve küresel minimumlara genellikle ulaşamamaktadırlar [6,8]. IIR süzgeçlerin tasarımında karşılaşılan zorlukların üstesinden gelebilmek için, sezgisel optimizasyon algoritmaları kullanılır. Sezgisel optimizasyon algoritmaları sahip oldukları küresel araştırma yetenekleri sayesinde çok-modlu hata yüzeylerinde küresel minimumlara veya yakınındaki çözümlere yüksek yakınsama hızları ile ulaşabilmekte ve arzu edilen tasarımları etkin şekilde gerçekleştirebilmektedir.

Frekans özellikleri zamandan bağımsız durağan işaretlerin işlenmesinde doğrusal zamanla değişmeyen süzgeç yapıları yeterli olmaktadır. Fakat, çoğu durumda üzerinde çalışılan işaretin istatistiksel özellikleri bilinmemekte ve işaretin frekans özellikleri zamana bağlı olarak değişmektedir. İstatistiksel özellikleri bilinmeyen veya durağan olmayan işaretlerin işlenmesinde, geleneksel metotlarla tasarlanan süzgeçler büyük oranda yetersiz kalmakta fakat bu tip işaretlerin işlenmesinde adaptif süzgeçler çok başarılı şekilde kullanılmaktadır [9,10]. Adaptif süzgeçler, katsayılarının adaptif olarak ayarlanması suretiyle bir hata fonksiyonunu optimize etmek veya daha önceden tanımlanmış olan bir optimizasyon kriterini sağlayabilmek için tanımlanmış olan zamanla değişen süzgeç yapılarıdır [11]. Klasik işaret işleme tekniği ile adaptif işaret işleme tekniği arasındaki temel fark, klasik işaret işleme

sistemlerinde durađan yapılar analiz edilebilirken adaptif işaret işleme sistemlerinde karakteristikleri her bir iterasyonda sürekli olarak deđişen yapılar analiz edilebilmektedir.

2. KONU VE KAPSAM

Evrimsel Tabanlı Algoritmalar

i.) Yapay Arı Koloni Algoritması

Yapay arı kolonisi algoritmasında, bir koloni de üç grup arı bulunmaktadır: işçi arılar, gözcü arılar ve kaşif (scout) arılar. Modelimizde, koloninin yarısı işçi, yarısı gözcü arı olarak seçilmiştir. Her bir nektar kaynağı için sadece bir işçi arı bulunmaktadır. Yani işçi arıların sayısı nektar kaynağı sayısına eşittir. Algoritmanın temel adımları ise şu şekildedir:

Initialization

REPEAT

- İşçi arıları kaynaklara gönder ve nektar miktarlarını hesapla
- Gözcü arıları kaynaklara gönder ve nektar miktarlarını hesapla
- Rasgele yeni kaynaklar bulmaları için kaşif arıları gönder
- O ana kadarki en iyi kaynağı hafızada tut

UNTIL (durma kriteri sağlanana kadar)

Her bir çevrim üç adımdan oluşmaktadır: işçi ve gözcü arıların kaynaklara gönderilmesi, gidilen kaynakların nektar miktarlarının hesaplanması, kaşif arının belirlenerek yeni bir kaynağa rasgele konumlanması. Yiyecek kaynakları optimize edilmeye çalışılan problemin olası çözümlerine karşılık gelmektedir. Bir kaynağa ait nektar miktarı, o kaynakla ifade edilen çözümün kalite değerini ifade etmektedir. Gözcü arılar rulet tekerleği prensibine [12] göre gidecekleri kaynakları belirlemektedirler. Her kolonide rasgele araştırma yapan kaşif arılar bulunmaktadır. Bu arılar yiyecek ararken herhangi bir ön bilgi kullanmamakta, tamamen rasgele araştırma yapmaktadırlar. Dolayısıyla arama maliyetleri düşüktür ve de buldukları kaynağın ortalama kalite değeri düşüktür. Zengin nektar kaynağına sahip keşfedilmemiş kaynakları bulmaları da olasıdır. ABC algoritmasında işçi arılardan biri seçilerek kaşif arı haline gelmektedir. Bu seçme işlemi “limit” parametresine göre yapılmaktadır. Bir kaynağı ifade eden çözüm belli sayıdaki deneme ile geliştirilememişse bu kaynak terk edilir ve bu kaynağa gidip gelen işçi arı kaşif arı haline gelir. Kaynağın terk edilmesi için belirlenmiş deneme sayısı “limit” parametresi ile belirlenmektedir.

Sağlıklı bir arama sürecinde keşif (exploration) ve keşfedilenden faydalanma (exploitation) aynı anda gerçekleşmelidir. ABC algoritmasında gözcü ve işçi arılar keşfedilen kaynaklardan faydalanma işleminde, kaşif arılar ise keşif sürecinde görev alırlar. Gerçek arılarda taşıma hızı koloninin bir kaynağı bulması ve onu kovana getirmesi ile belirlenirken, yapay arılar durumunda bulunan çözümün kalite değeri yani uygunluğu ile belirlenir.

Diğer sosyal yiyecek arayıcıları gibi, arılar E/T değerini yani birim zamanda yuvaya getirilen yiyecek miktarını belirten enerji fonksiyonunu maksimize etmek için çalışırlar. Bir maksimizasyon probleminde de amaç fonksiyonunun $F(\theta_i)$, $\theta_i \in R^p$, maksimize edilmesi işlemi gerçekleşir. θ_i , i. kaynağın pozisyonu olmak üzere $F(\theta_i)$ bu nektar miktarına karşılık gelir ve $E(\theta_i)$ ile orantılıdır. c çevrim sayısı (cycle), S: kovan etrafındaki nektar kaynağı sayısı olmak üzere $P(c) = \{\theta_i(c) | i = 1, 2, \dots, S\}$ tüm kaynakların pozisyon bilgilerini içeren nektar kaynağı popülasyonudur. Daha önce belirtildiği gibi gözcü arıların bir kaynağı

seçmeleri $F(\theta)$ değerine bağlı idi. Kaynağın nektar miktarı ne kadar fazla olursa, bu kaynağı bir gözcü arı tarafından seçilme olasılığı o kadar fazla olmaktadır. Yani θ_i pozisyonundaki bir kaynağın seçilme olasılığı şu şekildedir:

$$P_i = \frac{F(\theta_i)}{\sum_{k=1}^S F(\theta_k)} \quad (1)$$

Gözcü arı, işçi arıların bilgi aktarımını sağlayan danslarını izledikten ve (1) eşitliğindeki olasılık değeri ile θ_i konumundaki kaynağı seçtikten sonra, bu kaynağın komşuluğunda bir kaynak belirler ve kaynağın nektarını almaya başlar. Yani θ_i civarındaki kaynaklar arasında bir kıyaslama yapar. Seçilen komşuya ait pozisyon bilgisi şu şekilde hesaplanmaktadır:

$$\theta_i(c+1) = \theta_i(c) \pm \phi_i(c) \quad (2)$$

$\phi_i(c)$, θ_i civarında daha fazla nektara sahip bir kaynak bulabilmek için kullanılan, rasgele üretilen adım büyüklüğüdür. $\phi_i(c)$, k i' den farklı rasgele üretilen popülasyondaki bir çözüme ait indis olmak üzere $\theta_i(c)$ ve $\theta_k(c)$ çözümlerinin bazı bölümlerinin farkının alınması ile hesaplanır. $\theta_i(c+1)$ e ait nektar miktarı $F(\theta_i(c+1))$, $\theta_i(c)$ konumundaki kaynağa ait nektar miktarından daha fazla ise arı kovana giderek bu bilgisini diğerleri ile paylaşır ve yeni pozisyon olarak $\theta_i(c+1)$ değerini hafızasına kaydeder, aksi durumda $\theta_i(c)$ 'yi hafızasında saklamaya devam eder. θ_i konumundaki nektar kaynağı “limit” parametresi sayısınca gelişmemiş ise θ_i deki kaynak terk edilir ve o kaynağın arısı kaşif arı haline gelerek rasgele araştırma yapar, yeni bulduğu kaynak θ_i ye atanır.

Geliştirilmiş ABC (Modified ABC, M-ABC): Temel ABC algoritmasının 3 kontrol parametresi bulunmaktadır, koloni büyüklüğü (işçi arı sayısı yada yiyecek kaynağı sayısı), maksimum çevrim sayısı ve limit değeri. Temel ABC algoritmasında, bir çözümün komşuluğunda yeni bir çözüm üretilirken mevcut çözümün yalnızca rasgele seçilen bir parametresi değiştirilmektedir. Bu ise, algoritmanın yakınsama hızının düşük olmasına yol açmaktadır. Temel ABC algoritmasının bu dezavantajının üstesinden gelebilmek için, mevcut çözümün komşuluğunda yeni çözümler üretilirken mevcut çözümün birden fazla parametresi değiştirilebilir. Komşu çözümlerin bu şekilde üretildiği ABC algoritması, bu tez çalışmasında geliştirilmiş ABC algoritması olarak adlandırılmıştır. Dolayısı ile, geliştirilmiş ABC algoritması modifikasyon oranı olarak adlandırılan fazladan bir kontrol parametresine daha sahiptir. Modifikasyon oranı, komşu çözümler üretilirken mevcut çözümdeki parametrelerin değişim frekansını belirlemektedir. Modifikasyon oranı için önerilen değer aralığı $[0,1]$ şeklinde tanımlanmıştır.

ii.) Diferansiyel Gelişim Algoritması

Storn ve Price tarafından önerilmiş olan [13] diferansiyel gelişim algoritması, genetik algoritmanın prensiplerini temel alan bir algoritmadır ancak genetik algoritmadan farklı olarak rasgele olarak seçilmiş amaç vektör çiftlerinin farklarına dayalı bir mutasyon işlemi kullanır. D parametreyi içeren bir optimizasyon işlemi D boyutlu bir vektör ile gösterilebilir. Bu nedenle, DE algoritmasında NP adet çözüm vektörü başlangıçta rasgele üretilir. Rasgele

üretmiş olan bu popülasyon mutasyon, çaprazlama ve seçme operatörleri ile hatayı minimize edecek şekilde geliştirilir.

Temel bir DE algoritmasının adımları aşağıdaki gibidir,

Başlangıç popülasyonunu oluştur

Değerlendirme

REPEAT

Mutasyon

Rekombinasyon

Değerlendirme

Seçme

UNTİL (*maksimum jenerasyon sayısı veya minimum hata kriteri sağlanıncaya kadar*)

Mutasyon operatöründe, her bir $x_{i,G}$ hedef (parent) vektörü için Eşitlik 1 ile tanımlanan deneme (child) vektörü üretilir

$$v_{i,G+1} = x_{i,G} + K \cdot (x_{r_1,G} - x_{i,G}) + F \cdot (x_{r_2,G} - x_{r_3,G}) \quad (3)$$

burada, $i, r_1, r_2, r_3 \in \{1, 2, \dots, NP\}$ indisleri her birisi birbirinden farklı olan rasgele seçilmiş popülasyon indisleridir. Ayrıca, $F \in [0, 2]$, ölçekleme faktörü olarak tanımlanır ve $(x_{r_2,G} - x_{r_3,G})$ fark vektörünün çarpanıdır, K ise kombinasyon vektörüdür. Çaprazlama operatöründe, $u_{ji,G+1}$ deneme vektörünü üretmek için, hedef vektör ile deneme vektörü karşılaştırılır ve Eşitlik 4'te tanımlandığı gibi yeni deneme vektörü üretilir,

$$u_{ji,G+1} = \begin{cases} v_{ji,G+1} & \text{if } (rnd_j \leq CR) \text{ veya } j = rn_i \\ x_{ji,G} & \text{if } (rnd_j > CR) \text{ ve } j \neq rn_i \end{cases} \quad (4)$$

burada, $j = 1, 2, \dots, D$; $r_j \in [0, 1]$ rasgele sayılardır; $CR \in [0, 1]$ çaprazlama sabitini tanımlamaktadır ve $rn_i \in (1, 2, \dots, D)$ ise rasgele seçilmiş olan indekstir. Hedef vektörü ve ondan üretilmiş olan deneme vektörünün performansları karşılaştırılır ve iyi olanı seçilerek bir sonraki jenerasyona aktarılır.

iii.) Parçacık Sürüsü Algoritması

Parçacık sürüsü algoritması Eberhart ve Kennedy tarafından geliştirilmiş olan stokastik bir optimizasyon tekniğidir [14]. PSO algoritmasında, her birisi olası bir çözümü temsil eden bireylerin oluşturduğu popülasyon D-boyutlu bir uzayda hareket eder. Bireyler problem uzayı boyunca bu hareketleri esnasında mevcut optimum bireylerden bilgiler toplarlar. Herhangi bir iterasyonda, bireyin o andaki pozisyonu olası bir çözümü temsil eder. Bireyin o andaki pozisyonunda amaç fonksiyonu değerlendirildiğinde o bireye ait kalite değeri elde edilir. Ayrıca her bir bireyin bir hızı vardır ki bu hız parametresi ile bireylerin araştırma esnasındaki yönleri tanımlanmaktadır.

PSO algoritması da diğer algoritmalara benzer olarak, rasgele üretilmiş olan bireylerden oluşan bir başlangıç popülasyonu ile başlatılır ve daha sonra her bir jenerasyonda bireylerin konumları sürekli güncellenerek optimal çözümler bulunmaya çalışılır. Bütün iterasyonlarda, her bir bireyin konumu o ana kadarki en iyi çözüm olan $[\bar{p}(t)]$ ve popülasyonun en iyi çözümü olan $[\bar{g}(t)]$ tarafından güncellenir. Böylece iyi çözümler hakkındaki bilgiler sürü boyunca yayılır ve diğer bireylerde çözüm uzayında iyi çözümlerin olduğu bölgelere doğru hareket etmeye başlarlar. Her bir t adımında, $\vec{v}(t)$ hızı güncellenir ve birey farklı bir pozisyona, $\vec{x}(t+1)$, hareket eder. Bireyin bu yeni pozisyonu, önceki pozisyonu ile yeni hızının toplamı şeklinde aşağıdaki gibi tanımlanır,

$$\vec{x}(t+1) = \vec{x}(t) + \vec{v}(t+1) \quad (5)$$

Hızın eski değerinden yeni değerine güncellenmesi aşağıdaki eşitlik ile tanımlanır,

$$\vec{v}(t+1) = \omega \vec{v}(t) + r_1 c_1 (\vec{p}(t) - \vec{x}(t)) + r_2 c_2 (\vec{g}(t) - \vec{x}(t)) \quad (6)$$

burada, r_1 ve r_2 uniform dağılımlı rasgele sayılardır. w parametresi eylemsizlik faktörü (inertia factor) olarak adlandırılır ve bireyin eski hızı olan $\vec{v}(t)$ hızının genliğini tanımlar. c_1 ve c_2 değerleri ise sırasıyla $\vec{p}(t)$ ve $\vec{g}(t)$ vektörlerinin ağırlıklarıdır. Ayrıca, algoritmanın herhangi bir anında v_i parametresi v_{max} parametresi tarafından sınırlanır. PSO algoritmasında başlangıç popülasyonunda her bir bireye araştırma uzayında uniform olarak ve rasgele konumlar atanır. Bireylerin hızları $[v_{min}, v_{max}]$ aralığında rasgele seçilen hızlarla başlatılır.

PSO algoritmasının adımları aşağıdaki gibi tanımlanabilir,

Başlangıç popülasyonunu oluştur

REPEAT

Bireylerin kalite değerlerini hesapla

Sürüdeki en iyi bireyleri modifiye et

Bütün bireyler içerisinde kalite değeri en yüksek olan bireyi seç

Her bir bireyin hızlarını hesapla

Bireylerin konumlarını güncelle

UNTIL ((maksimum jenerasyon sayısı veya minimum hata kriteri sağlanıncaya kadar)

Şayet hızların toplamı v_{max} hızının değerini aşarsa araştırma uzayının o noktasındaki hız v_{max} ile sınırlandırılacaktır.

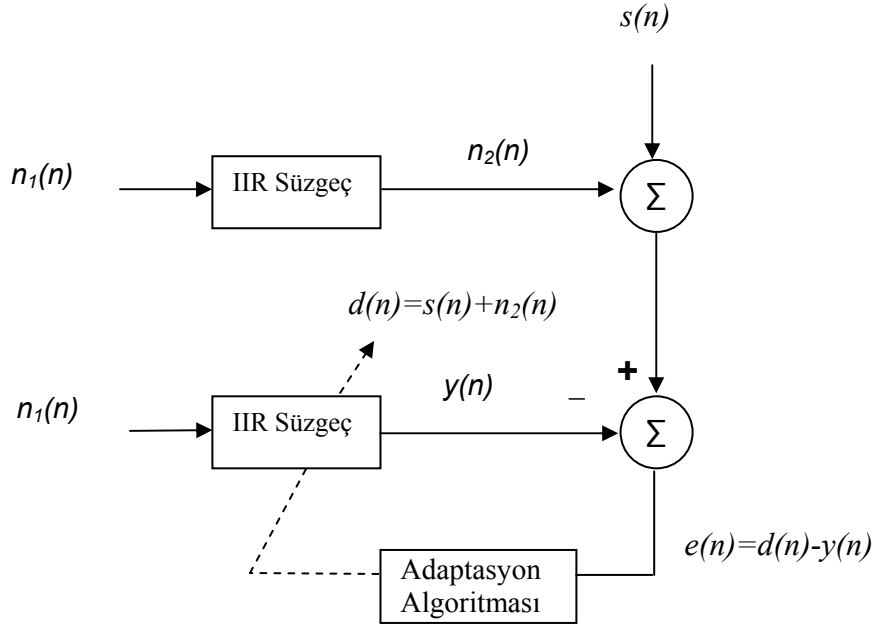
Problemin Tanımlanması

Gürültü giderimi uygulamasındaki amacımız gürültüyle bozulmuş olan işareti süzgeçleyerek yani işaret üzerindeki gürültü bileşenini yok ederek veya zayıflatarak arzu edilen işareti yeniden elde etmektir. Temel bir adaptif gürültü giderimi yapısı Şekil 1’de verilmiştir. Burada, dört farklı işaret tanımlanmaktadır,

$s(n)$: bilgi işareti $n_1(n)$: giriş işareti (Uniform Dağılımlı Gürültü)
 $e(n)$: hata işareti $n_2(n)$: gürültü işareti (Uniform Dağılımlı Gürültü)

Her bir iterasyonda giriş işareti $n_1(n)$, zamanla değişen süzgeç tarafından çıkış işareti $y(n)$ ’i üretecek şekilde işlenir. Elde edilen çıkış işareti ile arzu edilen işaret $[d(n)]$ karşılaştırılarak

bunların farkından oluşan hata işareti elde edilir. Elde edilen bu hata işareti, hata fonksiyonunu minimize edecek şekilde adaptif süzgeç katsayılarını ayarlayan adaptif algoritmaya geri beslenir. Süzgeç çıkışı ile hata işareti arasındaki fark minimize edilinceye kadar bu döngü devam eder.



Şekil 1. Adaptif IIR süzgeç ile gürültü giderimi uygulaması

Tasarlanacak olan adaptif IIR süzgece ait transfer fonksiyonu aşağıdaki gibi tanımlanır,

$$H(z) = \frac{\sum_{i=0}^M b_i z^{-i}}{1 + \sum_{i=1}^N a_i z^{-i}} \quad (7)$$

Optimize edilecek katsayı matrisimiz ise , $\mathbf{w} = [a_1 \dots a_N \ b_0 \ b_1 \dots b_M]^T$ şeklinde tanımlanmaktadır. Payda polinomunun kökleri olan sistem kutupları tasarım esnasında z- düzleminde birim dairenin içerisinde kalacak şekilde sınırlandırılırlar böylece tasarlanan sistem kararlı bir sistem olur [11,12]. Tasarımın amacı, \mathbf{w} parametrelerini ayarlayarak $J(\mathbf{w})$ hata fonksiyonunu minimize etmektir. Hata fonksiyonu (Mean Squared Error, Ortalama Karesel Hata - MSE) Eşitlik 2 ile tanımlandığı gibidir.

$$J(\mathbf{w}) = \mathbf{E}[|e(n)|^2] \quad (8)$$

burada, \mathbf{E} beklenen değerdir. $e(n)$ ise kestirim hatasıdır ve arzu edilen işaret ile adaptif süzgecin çıkışı arasındaki farkı temsil eder.

Simülasyonlarda kullanılmış olan MR ABC, ABC, DE ve PSO algoritmalarına ait kontrol parametre değerleri aşağıda Tablo 1. ile verilmiştir.

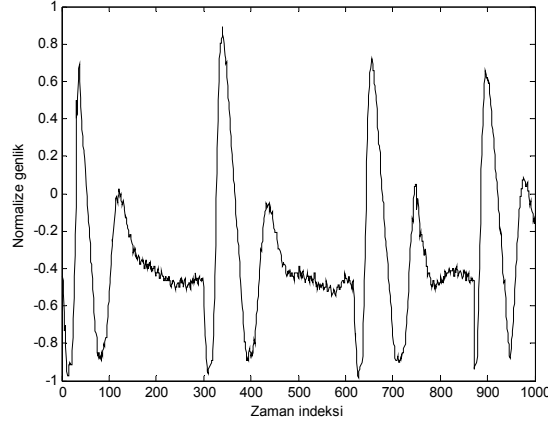
Tablo 1. Simülasyonlarda kullanılmış olan algoritmaların kontrol parametre değerleri

MR ABC	ABC	DE	PSO
Colony size = 40	Colony size = 40	Population size = 40	Swarm size = 40
limit value = 180 MR=0.7	limit value = 180	Crossover rate = 0.8 Scaling factor (F) = 0.6	Inertia factor, $\omega = 0.4$ Cognitive factor, $c_1 = 2$ Social factor, $c_2 = 2$
$X_{\max} = 1$ $X_{\min} = -1$	$X_{\max} = 1$ $X_{\min} = -1$	$X_{\max} = 1$ $X_{\min} = -1$	$X_{\max} = 1, X_{\min} = -1$ $V_{\max} = 0.5, V_{\min} = -0.5$

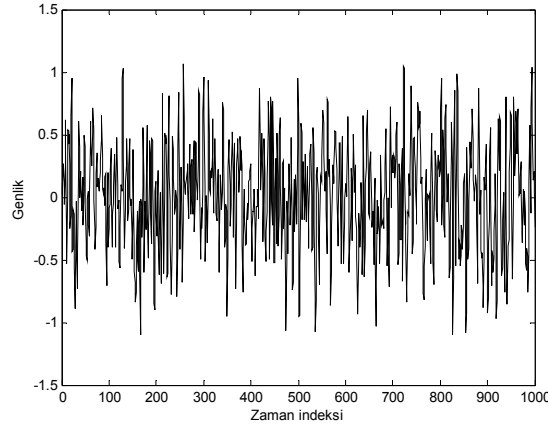
3. BULGULAR

Pratik sistemlerde karşılaşılan gürültüler genelde Uniform Gürültü ve Beyaz Gauss Gürültüsü ile modellenmektedir. Bu çalışmada her iki gürültü tipinin etkisi ve giderilmesine yönelik bulgular elde edilmiştir.

Bilgi işaretimiz ve bu bilgi işaretimize eklenmiş olan uniform dağılımlı gürültü işareti sırasıyla, Şekil 2 ve Şekil 3'te verilmiştir.

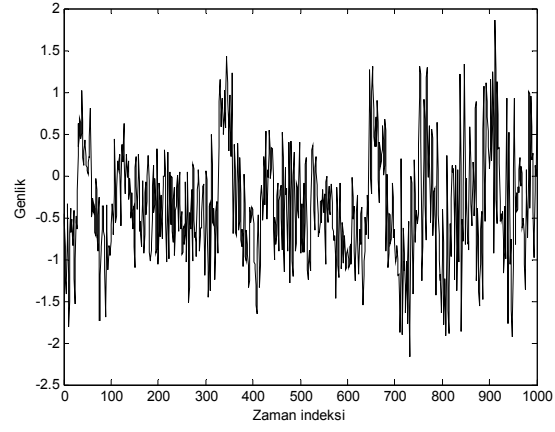


Şekil 2. EKG işareti

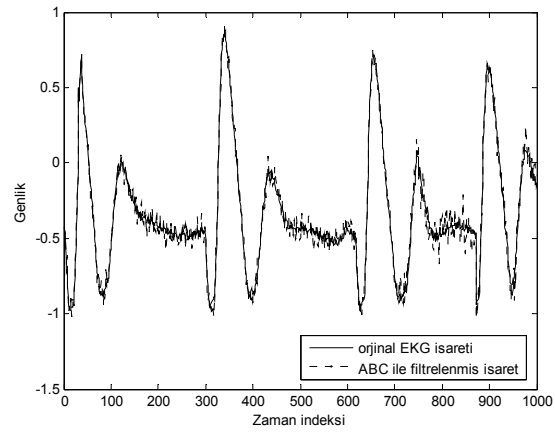
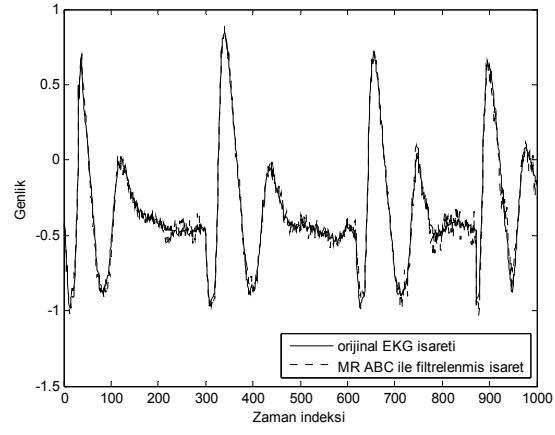


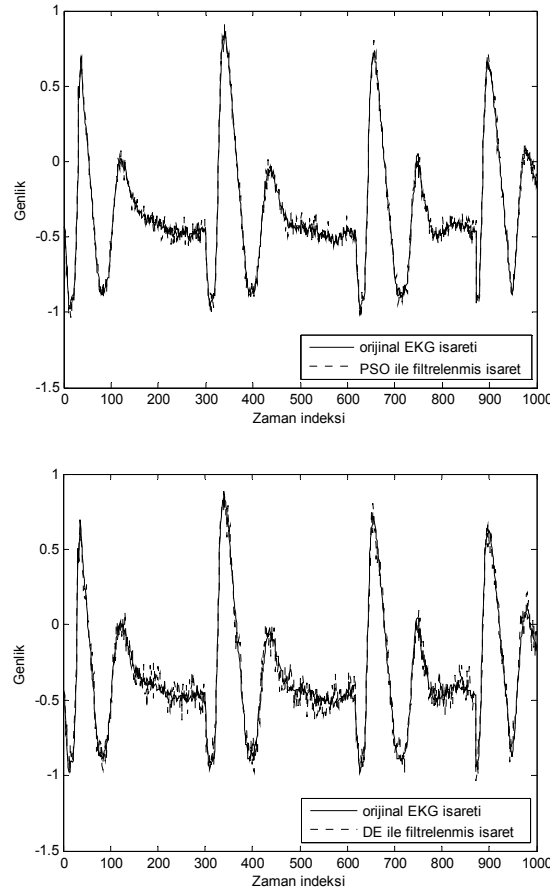
Şekil 3. EKG işaretine eklenen uniform dağılımlı gürültü

Şekil 3 ile verilmiş olan gürültü işareti, EKG işaretine eklendikten sonra elde edilmiş olan ve süzgeçlenmesi gereken gürültülü EKG işareti Şekil 4'te verilmiştir.



Şekil 4. Uniform dağılımlı gürültü eklenmiş EKG işareti

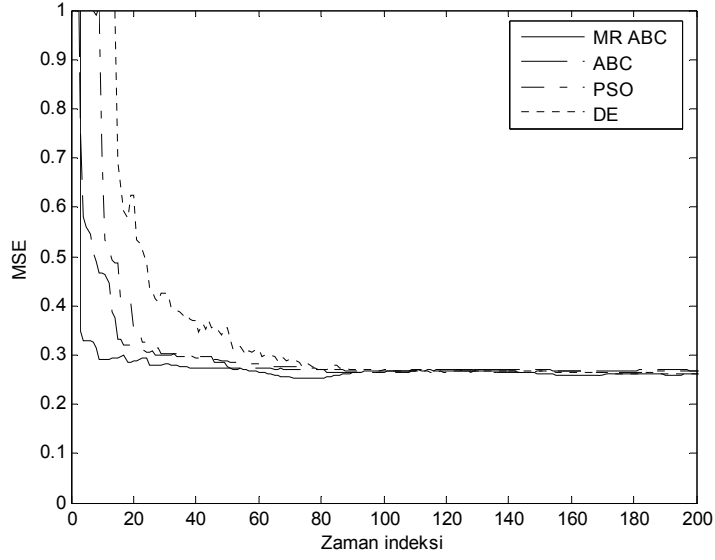




Şekil 5. MR ABC, ABC, PSO ve DE algoritmaları ile tasarlanmış adaptif IIR süzgeçler kullanılarak süzgeçlenmiş EKG işareti

Adaptif süzgeçleme işlemi MR ABC, ABC, DE ve PSO adaptasyon algoritmaları tarafından tasarlanan adaptif IIR süzgeçler ile yapılmış ve süzgeçleme sonucunda elde edilmiş olan işaretler Şekil 5’te verilmiştir. Adaptif süzgeç her bir adımda farklı frekansta bir gürültüyle karşılaşmakta ve kendisini o gürültüyü süzecek şekilde yeniden adapte etmektedir. Bu nedenle, sürekli bir gürültü salınımı oluşmakta, adaptif süzgeç katsayıları bu yeni frekanslı gürültüye adapte olacak şekilde güncellenmekte ve adaptasyon süreci tamamlanınca gürültü nispeten başarılı bir şekilde süzulebilmektedir.

Yakınsama hızı algoritmaların performans mukayesesi açısından önemli bir parametredir. Gürültü giderimi uygulaması için algoritmalara ait yakınsama hızları Şekil 6 ile verilmiştir. Görüldüğü gibi ABC ve PSO algoritmalarının yakınsama hızları benzer ve DE algoritmasından daha iyidir. MR ABC algoritması ise gerek MSE performansı gerekse yakınsama hızı açısından performansı en iyi algoritma olarak gözükmektedir.

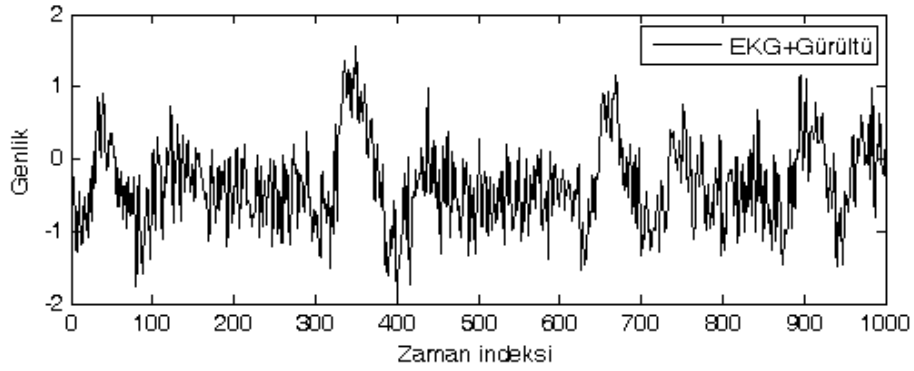
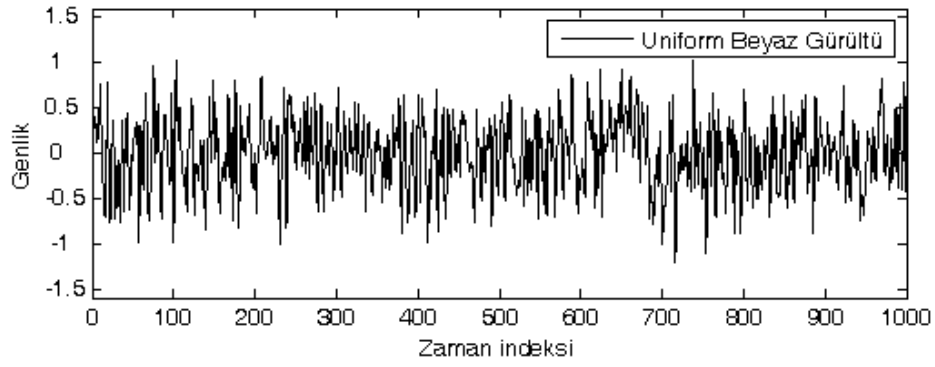
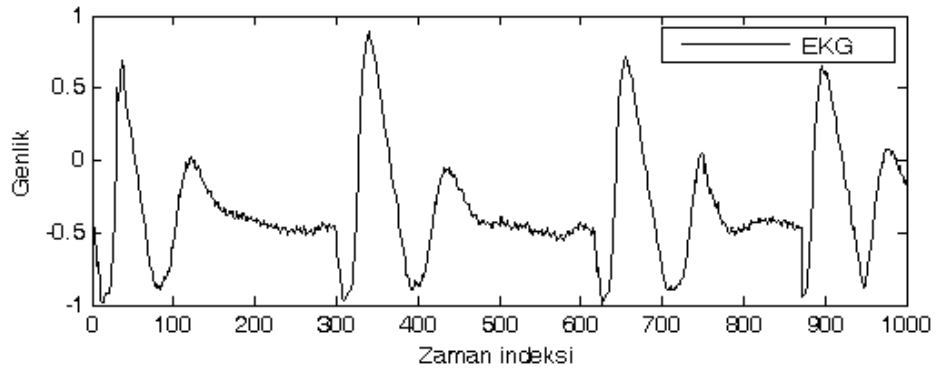


Şekil 6. Adaptif IIR süzgeçler ile gürültü giderimi uygulaması için MR ABC, ABC, PSO ve DE algoritmalarının yakınsama hızları

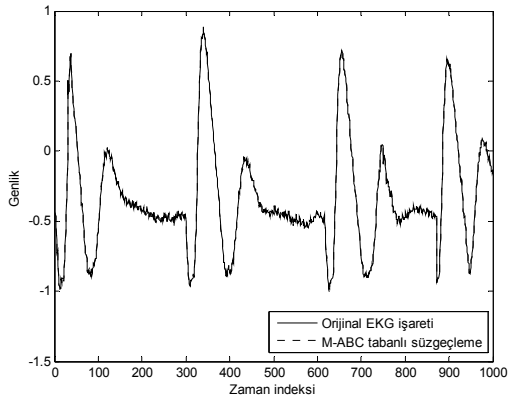
Yukarıda yapılmış olan çalışmada Uniform Gürültünün etkisi ve bu gürültünün giderilmesi incelenmiştir. Şimdi de pratikteki şebeke gürültüsünü en iyi modelleyen Beyaz Gauss Gürültüsü'nün etkisini ve bu gürültünün giderilmesini analiz edeceğiz. Adaptif gürültü giderimi uygulamasına yönelik tasarlanan adaptif IIR süzgeçlerin derecesi 5 olarak seçilmiş ve benzetimlerde kullanılan temel işaretler Şekil 7'de gösterilmiştir.

Elektrokardiyografi işareti üzerinde bozucu etki oluşturan uniform dağılımlı beyaz gürültü işareti M-ABC, ABC, PSO ve DE algoritmaları ile tasarlanan adaptif IIR süzgeçler kullanılarak süzgeçlendiğinde elde edilen EKG işaretleri Şekil 8 ile gösterilmiştir. Adaptif gürültü giderimi uygulamasında sezgisel algoritmalar ile tasarlanan adaptif IIR süzgeçler kullanıldığında, EKG işareti üzerindeki gürültünün büyük oranda bastırıldığı ve orijinal işaretin çok az bir bozulma ile yeniden elde edilebildiği görülmektedir. Algoritmaları kendi içerisinde mukayese etmek gerekirse, gürültü işaretinin M-ABC, ABC ve PSO algoritmaları ile gerçekleştirilen gürültü giderimi uygulamalarında daha iyi bastırıldığı, DE tabanlı uygulamada ise süzgeçlemiş işarete gürültüye ait nispeten yüksek salınımların bulunduğu görülmektedir.

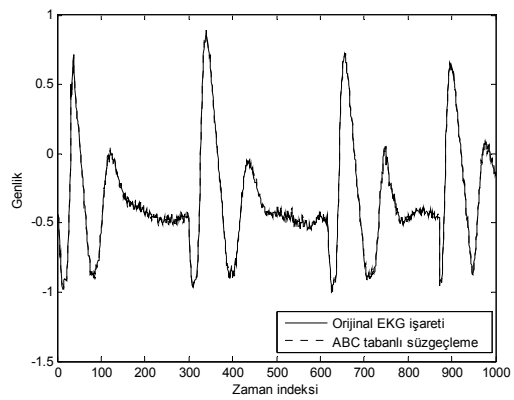
Her bir algoritma için 15 koşmanın ortalamasından elde edilen ve Şekil 9 ile verilen yakınsama grafiklerinden, algoritmaların çok yakın hata değerlerine yakınsadıkları dolayısı ile hata açısından benzer performanslar sergiledikleri görülmektedir. M-ABC algoritması optimal hata değerine en hızlı yakınsayan algoritma iken, optimal çözümlere en geç ulaşan algoritmanın DE algoritması olduğu görülmektedir. ABC ve PSO algoritmaları ise yakınsama hızı açısından benzer performanslar göstermektedirler.



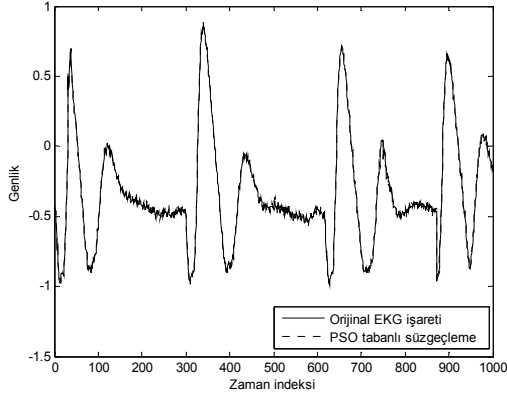
Şekil 7. Adaptif IIR süzgeçlerin gürültü giderimi uygulaması ile tasarımında kullanılan temel işaretler



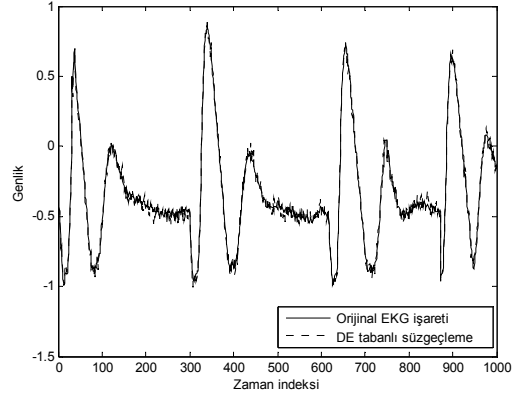
(a) M-ABC tabanlı süzgeçleme



(b) ABC tabanlı süzgeçleme



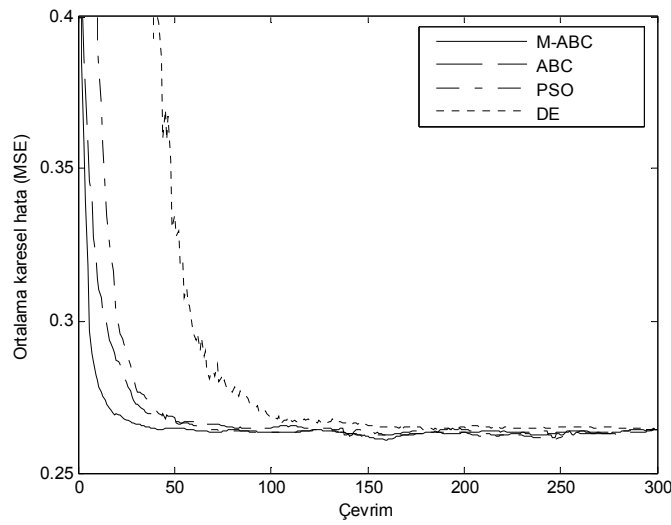
(c) PSO tabanlı süzgeçleme



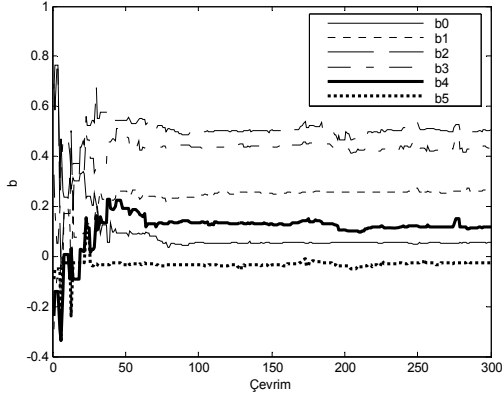
(d) DE tabanlı süzgeçleme

Şekil 8. M-ABC, ABC, PSO ve DE algoritmaları ile tasarlanmış olan adaptif IIR süzgeçler kullanılarak süzgeçlenmiş EKG işaretleri

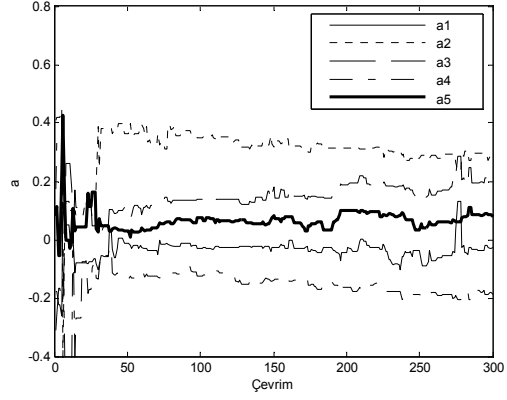
M-ABC ve ABC algoritmaları ile tasarlanan optimal adaptif IIR süzgeçlere ait katsayı değerlerinin optimizasyon süresince gelişimini gösteren grafikler, sırasıyla, Şekil 10 ve Şekil 11 ile verilmiştir. M-ABC algoritması ile gerçekleştirilen tasarımda ilk 50 çevrime kadar katsayı değerlerinde ani değer değişimlerinin olduğu ancak bu çevrimden sonra katsayıların daha kararlı bir şekilde gelişimini tamamladığı görülmektedir. ABC algoritması ile gerçekleştirilen tasarımda ise katsayı değerlerinde optimizasyon süresince ani değer değişimleri gözlenmekte ve son çevrimlere gelinceye kadar birçok katsayının optimum değerine ulaşamamakta ve katsayı değerlerindeki anlık değişimler devam etmektedir. Grafiklerden açıkça görülmektedir ki, M-ABC algoritması ile gerçekleştirilen tasarımlarda optimum ya da optimuma yakın değerlere daha hızlı ulaşılmakta ve gelişim daha kararlı bir şekilde devam etmektedir.



Şekil 9. Gürültü giderimi uygulamasına yönelik adaptif IIR süzgeçlerin tasarımında sezgisel algoritmalarının yakınsama hızları ve hata performansları

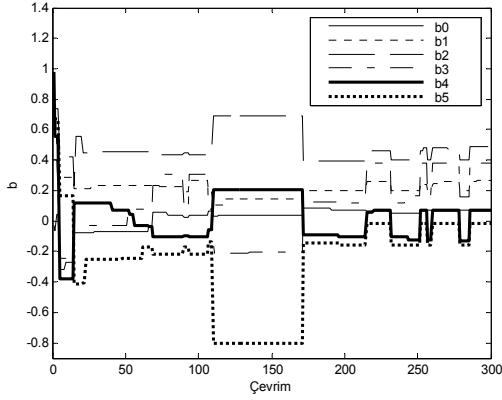


(a) Pay katsayıları

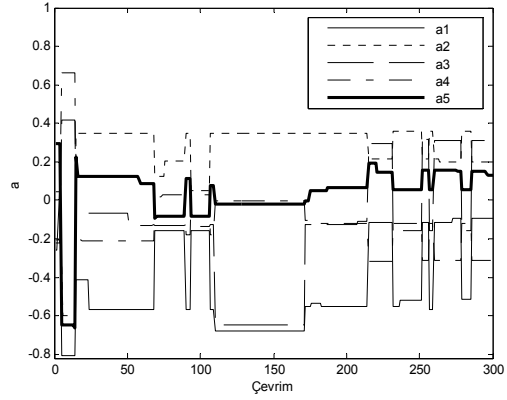


(b) Payda katsayıları

Şekil 10. M-ABC algoritması ile en düşük hata değerinin elde edildiği kořmada tasarlanmış olan adaptif IIR süzgecin katsayı gelişim grafikleri



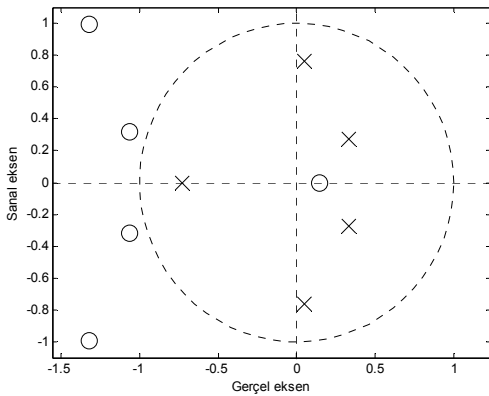
(a) Pay katsayıları



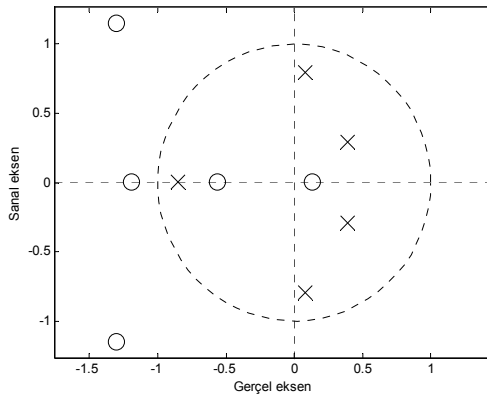
(b) Payda katsayıları

Şekil 11. ABC algoritması ile en düşük hata değerinin elde edildiği kořmada tasarlanmış olan adaptif IIR süzgecin katsayı gelişim grafikleri

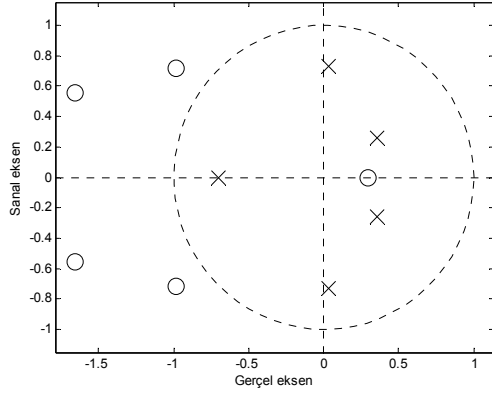
Sezgisel algoritmalar ile tasarlanan optimal süzgeçlerin Şekil 12’de verilen kutup-sıfır diyagramlarından bütün tasarımlarda kararlılık kriterinin sağlandığı görülmektedir.



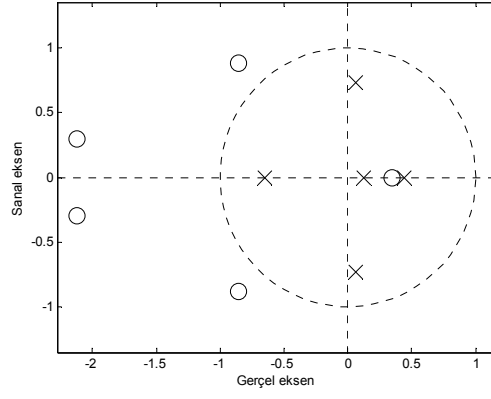
(a) M-ABC



(b) ABC



(c) PSO



(d) DE

Şekil 12. Gürültü giderimi uygulamasına yönelik sezgisel algoritmalar ile tasarlanan beşinci dereceden optimal adaptif IIR süzgeçlerin kutup-sıfır diyagramları

4. SONUÇ VE ÖNERİLER

Sezgisel algoritmalara dayalı adaptif IIR gürültü giderimi uygulamasında EKG işareti üzerindeki uniform dağılımlı gürültü ve beyaz Gauss gürültüsü başarılı bir şekilde bastırılarak, işaret orijinaline yakın bir şekilde elde edilmiştir. Adaptif gürültü giderimi uygulamasında en düşük hata değerine en hızlı ulaşan diğer bir ifade ile gürültü bileşenini en iyi ve hızlı şekilde bastıran algoritma M-ABC algoritması olmuştur. ABC ve PSO algoritmaları yakınsama hızı açısından benzer performansa sahipken, DE algoritmasının yakınsama hızının diğer algoritmalara göre daha düşük olduğu açıkça görülmüştür.

Yapılan çalışmalar sonucunda, ABC tabanlı yaklaşımların daha basit yapıda olması ve daha az kontrol parametresi içermesine rağmen performansının PSO ve DE algoritmaları ile benzer olduğu ve daha dinç bir davranış sergilediği görülmüştür.

Bu tez projesinde elde edilen bulgularda, pratik sistemlerde görülen bu iki gürültü tipinin giderilerek bilgi işaretinin tek ve bozulmadan yeniden elde edilmesinde sezgisel yöntemlerin başarılı bir şekilde kullanılabileceği görülmüştür. Bu iki gürültü tipi haricindeki diğer gürültü tiplerinde de bu yaklaşımların başarılı bir şekilde kullanılabileceğini söylemek mümkündür.

KAYNAKLAR

1. Çetinkaya, B., Genetik Algoritma Kullanarak Optimum Genlik Cevaplı ve Minimum Fazlı Sayısal Süzgeçlerin Tasarımı, Yüksek Lisans Tezi, Erciyes Üniversitesi, Kayseri, 2004.
2. Ertürk, S., Sayısal İşaret İşleme, Birsen Yayınevi, İstanbul, 2005.
3. Haykin, S., Adaptive Filter Theory, Prentice Hall, USA, 2002.
4. Karaboga, N., Digital IIR Filter Design Using Differential Evolution Algorithm, Eurasip Journal on Applied Signal Processing, 2005 (8), 1269-1276, 2005.
5. NG, S.C., et al., The Genetic Search Approach :A New Learning Algorithm for Adaptive IIR Filtering, IEEE Signal Processing Magazine, 13 (6), 38-46, 1996.
6. Krusienski, D.J., Jenkins, W. K., Design and Performance of Adaptive Systems based on Structured Stochastic Optimization Strategies, IEEE Circuits and Systems Magazine, 5 (1), 8-20, 2005.
7. Pasquato, L., Kale, İ., System Identification via Hybrid FIR-IIR Adaptive Filtering, IEEE Instrumentation and Measurement Technology Conference, Venice, 1064-1069, May 24-26, 1999.
8. Soleit, E. A., A Fast Adaptive Recursive Filter, International Journal of Electronics, 82 (4), 327-333, 1997.
9. Diniz, P. S. R., Adaptive Filtering Algorithms and Practical Implementations, Springer, USA, 2008.
10. Doğançay, K., Tanrikulu, O., Adaptive Filtering Algorithms with Selective Partial Updates, IEEE Transactions on Circuits and Systems-II, 48 (3), 762-769, 2001.
11. Jiang, D., A Study on Adaptive Filtering for Noise and Echo Cancellation, M.S. Thesis, Windsor University, Canada, 2005.
12. Kennedy, J. and Eberhart, R. C., Particle swarm optimization, 1995 IEEE International Conference on Neural Networks. 1995, 1942-1948.
13. Storn R, Price K. Differential evolution- A simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, 1995.
14. Eberhart RC, Kennedy J. A new optimizer using particle swarm theory. Sixth International Symposium on Micromachine and Human science 1995; 39-43.