

DİFERANSİYEL GELİŞİM ALGORİTMASI KULLANILARAK SİNYAL KESTİRİMİNE YÖNELİK ADAPTİF SDY SÜZGEÇ TASARIMI

Nurhan KARABOĞA¹

Canan Aslıhan KOYUNCU²

Elektrik Elektronik Mühendisliği Bölümü
Mühendislik Fakültesi
Erciyes Üniversitesi, 38090, Talas, Kayseri

e-posta¹: nurhan_k@erciyes.edu.tr

e-posta²: cananaslihan@gmail.com

Anahtar sözcükler: Diferansiyel Gelişim Algoritması, Adaptif SDY süzgeçler

ÖZET

Optimizasyon problemiyle karşılaşıldığında genellikle ilk yapılması gereken, probleme özel sezgisel bir yaklaşım tekniği belirlemektir. Gelişime dayalı algoritmalar, genel metotlarla karşılaştırıldığında oldukça üstün performans göstermektedirler. Bu çalışmada diferansiyel gelişim algoritması kullanılarak Adaptif Sonlu Darbe Yanıtlı (SDY) süzgeç vasıtasıyla sinyal kestirimi yapılmaktadır. Algoritmada kullanılan basit mutasyon işlemi, algoritmanın performansını geliştirmekte ve onu daha güçlü (robust) yapmaktadır. DG algoritması aynı zamanda hızlı, basit, kolayca kullanılabilir ve değiştirilebilir, gürültülü ve zamana bağlı amaç fonksiyonları için kullanılabilir, özellikle doğrusal olmayan sınırlamalı optimizasyon problemlerinde etkili olma vb. özelliklere de sahiptir.

1. GİRİŞ

Sayısal sistemler güvenilir ve ucuz olmalarının yanı sıra diğer bazı avantajlara da sahiptir. Sayısal sistemlerdeki işaretlerin, sadece belirli değerleri aldıkları için bozulmalardan etkilenmeleri de analog işaretlere göre daha azdır. Sayısal süzgeç ayrık zamanlı işaretleri filtrelemek için kullanılan sayısal bir sistemdir. Sayısal süzgeçler, işaretin örneklenmiş değerleri üzerinde nümerik hesaplamaları yapabilmek için sayısal işlemcileri kullanmaktadırlar. Süzgeç tasarımcıları, farklı bir çok süzgeç yapısını, örneğin, Sonlu Darbe Yanıtlı (SDY) ve Sonsuz Darbeli Yanıtlı (SzDY) süzgeçler, bunların doğrudan ya da kaskat şekilleri veya kafes yapıları vs., incelemektedirler. Çoğu süzgeç tasarım prosedüründe tasarımcı, önce bir süzgeç yapısını ardından süzgeç katsayılarını ve sonra da seçilen süzgecin ihtiyaçları yerine getirip getirmediğini belirlemektedir[1]. Bu prosedür eğer dikkatlice gerçekleştirilirse, tatmin edici sonuçlar alınabilir. Çoğu tasarım algoritmaları daha az karmaşık yapılar üzerinde yoğunlaştığı için seçilen süzgeç yapısı yine de optimalliği sağlamaz[2]. Giriş verisinin istatistiksel karakteristiği zamana bağlı

olarak değişirse ya da giriş verisi hakkındaki gereken bilgi yeterli olmazsa adaptif süzgeçlere ihtiyaç duyulmaktadır [3]. Adaptif SDY süzgeçlerinin, adaptif lineer toplayıcıların, tasarlanması için bir çok metot vardır. Bilgisayarlar, performanslarının gelişimi ile evrimsel algoritmalara dayanan optimizasyon metotlarının yeni bir sınıfını kullanarak bazı teknik problemlerin çözümüne izin vermektedir. Bu çalışmada, Diferansiyel Gelişim Algoritması olarak adlandırılan algoritma, adaptif SDY süzgecin sinyal kestirimine uygulanmasına yönelik olarak kullanılmıştır.

DG algoritması ilk olarak 1995 yılında K. Price tarafından ortaya konmuştur [4]. DG algoritması basit ama güçlü popülasyon tabanlı bir algoritmadır. Özellikle bütünüyle düzenlenmiş uzayda tanımlı ve gerçek değerli tasarı parametrelerini içeren fonksiyonları, küresel olarak optimize etmek amacıyla kullanılan bir direkt araştırma algoritmasıdır[5]. Bu çalışmada Adaptif SDY süzgecin sinyal kestiriminde kullanılması için, DG algoritmasına dayalı bir metot sunulmaktadır.

Bölüm 2 ve 3'te Adaptif SDY süzgeçlerin, Sinyal Kestiriminin ve DG Algoritmasının kısa bir özeti verilmektedir. Bölüm 4'te Sinyal kestirimine yönelik Adaptif SDY süzgeç tasarımı için DE algoritmasının nasıl uygulandığı gösterilmekte ve simülasyon sonuçları sunulmaktadır.

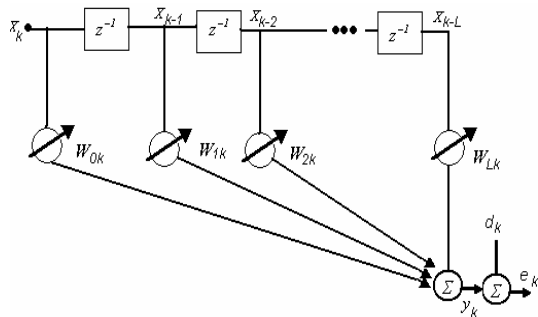
2. ADAPTİF SDY SÜZGEÇLER

Adaptif süzgeçler, otomatik ayar yapabilen sayısal süzgeçlerdir. Bu süzgeçler giriş işaretlerine göre değişebilirler. Adaptif SDY süzgeçler, değişken işaret durumlarına karşılık süzgeç karakteristiklerinin değişimini gerektiren uygulamalarda kullanılır[6].

Adaptif SDY süzgeçlerde iki giriş gerekmektedir: işaret ve referans giriş. Bir adaptif süzgeç kendi katsayılarını, ağırlıklarını, yenileme kabiliyetine sahiptir. Yeni katsayılar bir katsayı üreticinden

süzgece gönderilir. Katsayı üretici gelen bir sinyale cevaben katsayıları değiştiren adaptif bir algoritmadır. Çoğu uygulamalarda katsayı üreticinin amacı süzgeç katsayılarını referans girişe eşleştirmektir, bundan dolayı adaptif süzgeç referans girişi işaretten çıkarabilir. Referans giriş işareti değiştiğinden katsayılar da bunu yakalamak için değişmek zorundadır, bu çalışma prensibinden dolayı da adaptif süzgeç olarak adlandırılmaktadır.

Adaptif SDY süzgeçler, gürültü giderme, lineer kestirim, sistem modelleme, zaman gecikmesi kestirimi, parametre kestirimi ve adaptif kontrolü içeren uygulamaların çoğunda kullanılmaktadır. Aynı zamanda, adaptif süzgeçlemede de adaptif lineer toplayıcılar kullanılmaktadır. Şekil-1'de, SDY tek girişli bir adaptif lineer toplayıcının yapısı gösterilmektedir.

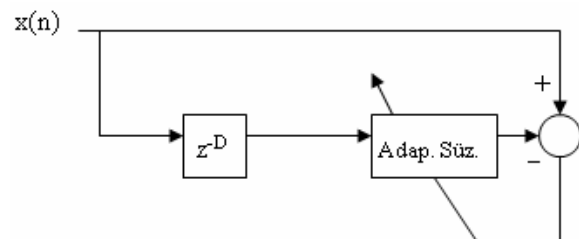


Şekil-1 Tek girişli bir adaptif lineer toplayıcı

SDY süzgeç, L+1 uzunluğundadır, katsayıları da w_k ile gösterilmektedir. w_k katsayılarının ayarlanması çevreye bağımlı olabilir ve katsayılar zamanla değişiyor olabilir. Şekil-1'de, x_k adaptif SDY süzgecinin giriş işareti, y_k çıkış işareti, d_k arzu edilen işaret ve e_k hata işaretidir.

2.1 Sinyal Kestirimi

Bir sinyalin geçmişteki değerlerini kullanarak gelecekteki değerini kestirme problemi sinyal kestirimi olarak bilinmektedir. Adaptif süzgecin başarılı olabilmesi için, $x(n)$ giriş işareti ile $d(n)$ istenilen çıkış işareti arasında korelasyon olmalıdır. Şekil 2'de verilen D, $d(n)$ 'nin girişteki gecikmiş değerlerini belirlemektedir. Yani ne kadar geciktiğinin göstergesi olmaktadır.



Şekil-2 Sinyal Kestirimi için genel bir gösterim

3. DİFERANSİYEL GELİŞİM ALGORİTMASI

Diferansiyel gelişim algoritması küresel optimizasyon için basit, ama güçlü popülasyon tabanlı bir algoritmadır[7]. Aerodinamik şekillerin optimizasyonunda, IIR süzgeç tasarımında, Sınır ağlarının öğrenmesi gibi çoğu uygulamada dinçliği ve gücü gösterilmiştir[8-10]. DG'nin basit yapısı, kullanım kolaylığı, hızı ve dinçliği en önemli avantajları arasında yer almaktadır. DG'nin önemli parametreleri: NP (Number of Population-popülasyon büyüklüğü), CR (Crossover Rate-çaprazlama sabiti), F (Scaling Factor-ölçekleme faktörü) olarak sayılabilir. D parametresini içeren bir optimizasyon, D boyutlu bir vektör ile gösterilebilir. DG'de, NP adet çözüm vektörünün bir popülasyonu başlangıçta rasgele meydana getirilir. Bu popülasyon mutasyon, çaprazlama ve seleksiyon operatörleri uygulanarak başarılı bir şekilde geliştirilir[11-12]. DG algoritmasının temel adımları aşağıdaki gibi verilebilir:

Başlangıç popülasyonunun oluşturulması
Değerlendirme

Repeat

Mutasyon

Rekombinasyon

Değerlendirme

Seleksiyon

Until (*durdurma kriteri sağlanıncaya kadar*)

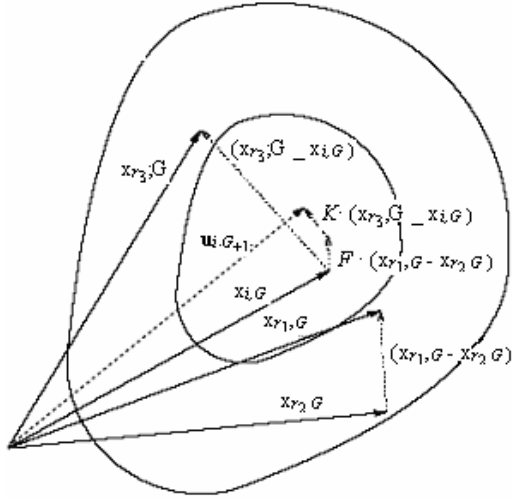
DG, diğer Evrimsel Algoritmalarından mutasyon ve rekombinasyon aşamalarında farklılık göstermektedir. DG popülasyonu karıştırmak için, çözüm vektörleri arasındaki ağırlıklandırılmış farkları kullanmaktadır.

$$\mathbf{u}_{i,G+1} = \mathbf{x}_{i,G} + K \cdot (\mathbf{x}_{r3,G} - \mathbf{x}_{i,G}) + F \cdot (\mathbf{x}_{r1,G} - \mathbf{x}_{r2,G})$$

Rasgele seçilen $r_1; r_2; r_3 \in \{1, 2, \dots, n\}$;

$$r_1 \neq r_2 \neq r_3 \neq i \quad (1)$$

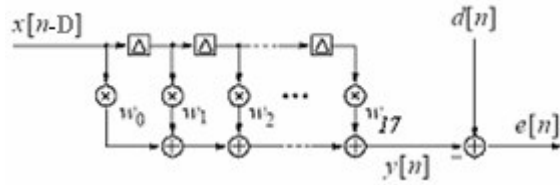
Bu çalışmada kullanılan DG, DG/current-to-rand/1 (Eşitlik 1) olarak bilinir ve rotasyonel olarak değişimsizdir(invariant)[13]. Bir Diferansiyel Gelişim Algoritması popülasyonu başlangıç parametre sınırları içerisinde rasgele başlatılır. Her bir G jenerasyonunda, popülasyon pertürbasyona uğrar. Birbirinden farklı üç birey ya da \mathbf{x} ile gösterilen çözüm vektörleri popülasyondan rasgele seçilir. K katsayısı $\mathbf{x}_{r3,G}$ ve mevcut birey $\mathbf{x}_{i,G}$ arasında meydana gelen birleşmenin seviyesini temsil eder. F katsayısı ise $\mathbf{x}_{r1,G} - \mathbf{x}_{r2,G}$ vektör farkından ileri gelen adım büyüklüğünün ölçeklenmesini gösterir.



Şekil-3 DG algoritmasında mevcut çözümden rastgele yeni bir çözümün üretilmesi

Şekil-3, yeni bir aday çözümün gelişimi ile ilgili vektörler arasındaki ilişkiyi göstermektedir. Tipik olarak tek amaçlı durumlarda, eğer yeni birey $u_{i,G+1}$ halen seçili olan birey $x_{i,G}$ 'den daha iyiyse bu durumda hali hazırdaki birey yeni olanla yer değiştirir. Algoritma i üzerinden 1 'den n 'e kadar tekrarlanır. Burada n , popülasyonun büyüklüğüdür.

4. SİMÜLASYON SONUÇLARI



Şekil-4 18 ağırlıklı adaptif SDY süzgeç

DG algoritmasında, adaptif süzgecin uygun değerini iyileştirmek için LMS hata fonksiyonu kullanılmıştır. En Küçük Karesel Hata fonksiyonu, arzu edilen işaretten tasarlanan süzgecin çıkışının çıkarılarak elde edilmiş olan hataların kareleri alınarak toplanması ve toplamın karekökünün minimize edilmesi amacıyla taşınmaktadır.

$$LMS = 1/N \left\{ \sum_{n=1}^N [d(n) - y(n)]^2 \right\}^{1/2} \quad (2)$$

Bu çalışmada kullanılan LMS sıfıra sürülecek hata fonksiyonu, y izlenecek gerçek işaret ve d arzu edilen işaret. Şekil 4'te, 18 ağırlıklı tek girişli adaptif SDY süzgeç gösterilmektedir. Bu çalışmada kullanılan işaretler, Eşitlik 3'te verilmektedir.

$$d(n) = \sin(t), \quad t = \pi/10 : 3 \cdot \pi/5 : 60 \cdot \pi$$

$$x(n) = d(n-D) \quad (3)$$

Tablo-1'de, bu çalışmada kullanılan DG'nin kontrol parametreleri verilmiştir.

Tablo-1. DG'nin kontrol parametreleri

Popülasyon büyüklüğü	100
Çaprazlama oranı	0.8
Birleşme faktörü	0.2
Ölçekleme faktörü	0.2
Jenerasyon sayısı	10000

Sinyal kestirimine yönelik adaptif SDY süzgeç için LMS hata fonksiyonu kullanıldığında elde edilen ağırlıklar Tablo-2'de verilmiştir.

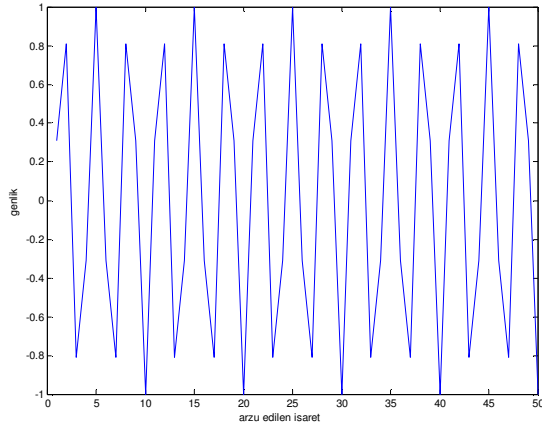
Tablo-2. LMS hata fonksiyonu kullanıldığında sinyal kestirimine yönelik Adaptif SDY Süzgeç için elde edilen ağırlıklar

Ağırlıklar	LMS hata fonksiyonu ile elde edilen değerler
w_0	0.05128181443994
w_1	0.84789573517644
w_2	0.86208286633288
w_3	0.27171351440327
w_4	0.68205686535859
w_5	0.66693234738127
w_6	0.64548701581248
w_7	0.20347011633153
w_8	-0.06951928626479
w_9	0.05921214436185
w_{10}	0.35152863016579
w_{11}	0.54536370691754
w_{12}	0.11582073257029
w_{13}	0.19573675589712
w_{14}	0.43535978491581
w_{15}	0.67030141345380
w_{16}	0.48493361488178
w_{17}	0.09850227686512

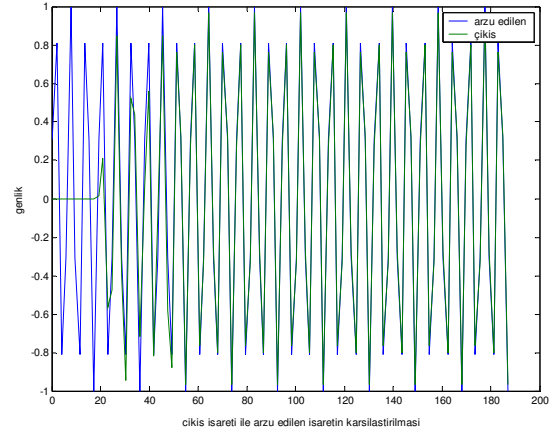
Sinyal kestirimine yönelik süzgeç tasarımı için, DG algoritması kullanılarak çıkışla arzu edilen işaret arasındaki hata minimize edilene kadar süzgeç parametreleri başarılı olarak ayarlanmıştır.

Şekil-4'e yönelik DG algoritmasına ait olan sonuçlar Tablo 1'deki parametre değerleri kullanılarak elde edilmiştir. Şekil- 5, 6'de gösterildiği gibi giriş işareti bir sinüs işareti ve arzu edilen işaret de giriş işaretinin önceki versiyonudur. Şekil-7, LMS hata fonksiyonu ile birlikte adaptif SDY süzgeç çıkışından elde edilen işareti göstermektedir. Şekil 8'de, arzu edilen işaret ile süzgeç çıkışında elde edilen işaretin karşılaştırılması

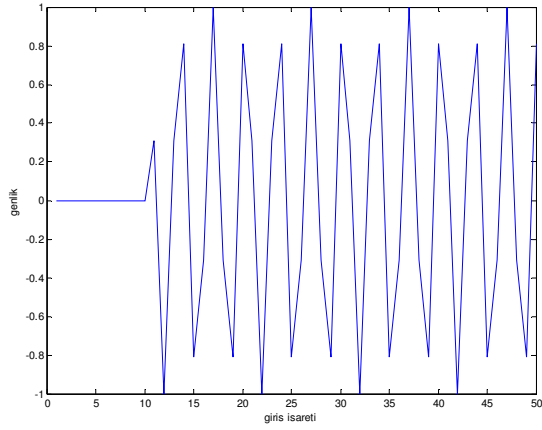
verilmektedir. Şekil-9'da, jenerasyona göre hata fonksiyonunun performansı görülmektedir.



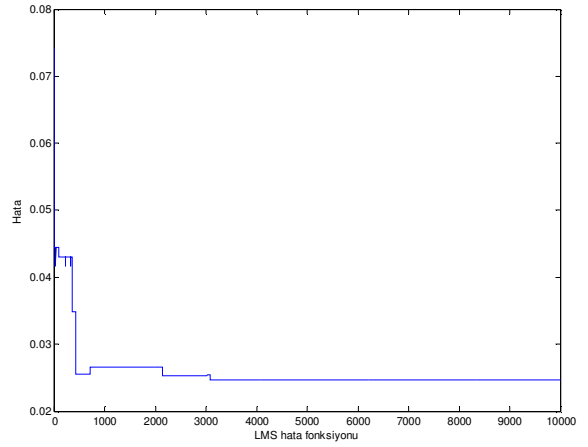
Şekil-5 Arzu edilen işaret, $d(n)$



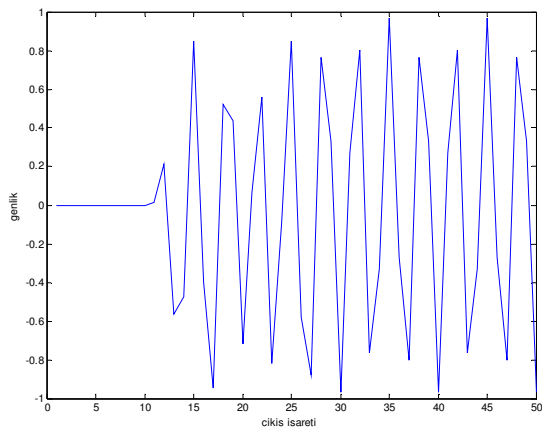
Şekil-8 Arzu edilen işaret ile süzgeç çıkışında elde edilen işaretin karşılaştırılması



Şekil-6 giriş işareti, $x(n)$



Şekil-9 LMS hatasının jenerasyona göre gelişimi



Şekil-7 Adaptif Süzgeç Çıkışında Elde Edilen İşaret

5. SONUÇ

Bu çalışmada, sinyal kestiriminde evrimsel bir hesaplama yöntemi olan Diferansiyel Gelişim (DG) algoritmasının kullanılmasına yönelik bir uygulama gerçekleştirilmiştir. DG algoritması aynı zamanda hızlı, basit, kolayca kullanılabilir ve değiştirilebilir, gürültülü ve zamana bağlı amaç fonksiyonları için de kullanılabilir gibi özellikle doğrusal olmayan sınırlamalı optimizasyon problemlerinde etkili olma vb. özelliklerine de sahiptir. Şekillerden de görüleceği gibi kestirim başarıyla gerçekleştirilmiştir.

KAYNAKLAR

[1] Nurhan Karaboğa, Canan Aslıhan Koyuncu: Optimal Determination of Adaptive Linear Combiner

Weights Using Differential Evolution Algorithm. INISTA, Yildiz Technical University, Istanbul (2005),71-74

[2] Lohmann R, Gorne T, Schneider M and Warstat M: Digital Systems and Structure Evolution, Prepr. 94th Conv. Audio Eng. Soc., D6-2, Berlin (1993)

[3] Adem Kalinli, Nurhan Karaboga: A New Method for Adaptive IIR Filter Design Based on Tabu Search Algorithm, AEÜ, 2004.

[4] Storn, R. - Price, K.: Differential Evolution - a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. Journal of Global Optimization, Vol. 11. Kluwer Academic Publishers (1997) 341 – 359

[5] Derviş Karaboğa, Yapay Zeka Optimizasyon Algoritmaları, Nobel Yayınevi, İstanbul, 2004.

[6] Allan V. Oppenheim: Adaptive Signal Processing. Prentice Hall Signal Processing Series, Prentice Hall (1989) 22-23

[7] Price K. V.: Differential Evolution: a Fast and Simple Numerical Optimizer. In: Smith, M., Lee, M., Keller, J., Yen., J. (eds.): Biennial Conference of the North American Fuzzy Information Processing Society, NAFIPS. IEEE Press, New York (1996) 524-527

[8] Ilonen, J., Kamarainen, J. -K., Lampinen, J.: Differential Evolution Training Algorithm for Feed-Forward Neural Networks. In: Neural Processing Letters Vol. 7, No. 1 (2003) 93-105

[9] Storn, R.: Differential Evolution Design of an IIR-Filter with Requirements for Magnitude and Group Delay. In: Proceedings IEEE International Conference on Evolutionary Computation ICEC'96. IEEE Press, New York (1996) 268 – 273

[10] Rogalsky, T., Derksen, R.W. and Kocabiyik, S.: Differential Evolution in Aerodynamic Optimization. In: Proceedings of the 46th Annual Conference of the Canadian Aeronautics and Space Institute. (1999) 29-36

[11] Dervis Karaboga, Selcuk Okdem,: A Simple and Global Optimization Algorithm for Engineering Problems: Differential Evolution Algorithm. Turk J Elec Engin, Vol.12, No.1,(2004)

[12] Nurhan Karaboga,: Digital IIR Filter Design Using Differential Evolution Algorithm. EURASIP Applied Signal Processing (2005) No.8, 1269-1276.

[13] Price, K V.: An Introduction to Differential Evolution. In: Corne, D., Dorigo, M., and Glover, F. (eds): New Ideas in Optimization McGraw-Hill, London UK (1999) 79-108