

## YENİ BİR ADAPTİF FİLTRELEME YÖNTEMİ: HİBRİD GS-NLMS ALGORİTMASI

*Sedat TİRYAKI\**

*Metin HATUN\*\**

*Osman Hilmi KOÇAL\*\**

**Özet:** Bu makalede, adaptif filtre katsayılarını ayarlamak için GS (Gauss-Seidel) ve NLMS (Normalized Least Mean Squares) algoritmalarının birlikte kullanıldığı hibrid bir algoritma önerilmiş ve ayrıca önerilen yeni algoritmanın yakınsama hızı, kararlılığı ve işlem karmaşıklığı incelenmiştir. Önerilen algoritma yapılan bir benzetim çalışmasıyla yakınsama hızı ve işlem yükü açısından benzer algoritmalarla karşılaştırmalı olarak incelenmiştir. Elde edilen sonuçlara göre, önerilen hibrid algoritmanın bir ara yöntem olarak işlem karmaşıklığı veya yakınsama hızı açısından diğer algoritmalara iyi bir alternatif olduğu görülmüştür.

**Anahtar Kelimeler:** Adaptif filtreler, Gauss-Seidel, NLMS, hibrid algoritma.

### A New Adaptive Filtering Method: Hybrid GS-NLMS Algorithm

**Abstract:** In this paper, a hybrid algorithm based on the use of Gauss-Seidel and Normalized Least Mean Squares algorithms together is introduced for adjusting of adaptive filter coefficients and also convergence rate, stability and computational complexity of the proposed algorithm is studied. The proposed algorithm is compared with similar algorithms by viewpoints of computational complexity and convergence rate by a simulation study. According to the results obtained, it is shown that the proposed hybrid algorithm is a good alternative to the others as an intermediate method by viewpoints of computational complexity or convergence rate.

**Key Words:** Adaptive filters, Gauss-Seidel, NLMS, hybrid algorithm.

## 1. GİRİŞ

Adaptif filtre katsayılarının ayarlamasında kullanılan algoritmalar eğim tabanlı algoritmalar ve en küçük kareler tabanlı algoritmalar olmak üzere iki temel grupta ele alınabilirler (Diniz, 1987, Farhang-Boroujeny, 1998, Haykin, 2002). Eğim tabanlı algoritmalar düşük işlem yükü avantajına sahip olup yüksek örnekleme hızlarında ve çoğunlukla adaptif sinyal işleme ve haberleşme uygulamalarında tercih edilmektedir. Bunlar LMS (Least Mean Squares), NLMS (Normalized LMS), ve AP (Affine Projection) algoritmalarıdır (Haykin, 2002, Haykin ve Widrow, 2003). Fakat bu algoritmalar giriş sinyaline ait korelasyon matrisinin özdeğer yayılımına da bağlı olarak yavaş yakınsama hızı dezavantajına sahiptir. En küçük kareler tabanlı algoritmalar ise yüksek işlem yüküne rağmen yakınsama özelliklerinin eğim tabanlı algoritmalara göre çok iyi olmasından dolayı tercih edilmektedir. RLS (Recursive Least Squares) algoritması bu gruba girmektedir. Fakat yüksek örnekleme hızı gerektiren durumlar yoğun işlem yükünden dolayı bu algoritmaların kullanım alanlarını sınırlamaktadır.

Diğer taraftan, bu iki algoritma grubuna alternatif olarak yüksek yakınsama hızı ve en küçük kareler tabanlı algoritmalara oranla daha az işlem yükü olan bazı algoritmalar da önerilmiş ve kullanılmıştır (Koçal, 1998, Bose, 2004). Bu algoritmalar normal denklemin GS (Gauss-Seidel) iterasyonlarıyla çözümü üzerine kurulu olup, yakınsama hızı açısından giriş sinyaline ait korelasyon matrisinin özdeğer yayılımına bağımlı olmasına rağmen eğim tabanlı algoritmalara oranla daha yüksek bir yakınsama hızına sahiptir ve korelasyon matrisinin özdeğer yayılımının küçük olması durumunda RLS algoritmasına çok yakın sonuçlar vermektedir (Koçal, 1998). Ayrıca işlem yükü RLS algoritmasına göre daha azdır, fakat LMS grubu algoritmalarından fazladır. GS algoritması temelde RLS algorit-

\* Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik Mühendisliği Bölümü, 16059, Görükle, Bursa.

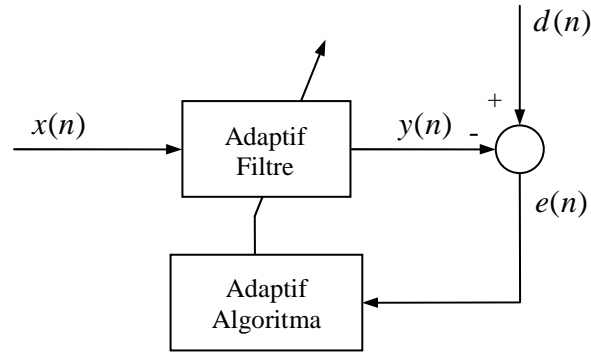
\*\* Uludağ Üniversitesi, Mühendislik-Mimarlık Fakültesi, Elektronik Mühendisliği Bölümü, 16059, Görükle, Bursa.

ması gibi korelasyon matrisinin birikimli değerlerini kullanan tekrarlamalı bir algoritma olup, korelasyon matrisinin pozitif tanımlılığı sağlandığı sürece kararlılığı garantelenmiş olmaktadır (Hatun ve Koçal, 2005). Bu makalenin amacı, düşük işlem yüküne sahip eğim tabanlı algoritmalar ile bu algoritmalarla oranla daha hızlı yakınsama ve kararlılık özelliklerine sahip olan GS algoritmasını birlikte kullanarak adaptif filtrelerde istenen özellikleri ön plana çıkaran yeni bir algoritma elde etmektir.

Bu makalede, düşük işlem yüküne sahip olan eğim tabanlı NLMS algoritması ile bu algoritmaya göre yakınsama hızı kararlılık özellikleri daha iyi olan GS algoritması birlikte kullanılarak yukarıda bahsedilen iki algoritma grubunun istenen özelliklerine sahip olan hibrid bir algoritma elde edilmiştir. Yapılan bir adaptif kanal dengeleyici uygulamasıyla önerilen algoritmanın yakınsama özellikleri ve işlem yükü benzer algoritmalarla karşılaştırmalı olarak incelenmiştir.

## 2. ADAPTİF FİLTRELEME ALGORİTMALARI

Adaptif filtreleme algoritmaları aşağıdaki şekilde görüldüğü gibi filtre parametrelerini iteratif olarak ayarlamak için kullanılır.



Şekil 1:  
Adaptif filtreleme işlemi

Burada  $x(n)$  adaptif filtrenin ayrık-zamanlı giriş sinyali,  $y(n)$  adaptif filtrenin tahmin edilen parametreler kullanılarak hesaplanan çıkış sinyali,  $d(n)$  ise adaptif filtrenin takip etmesi istenen sinyaldir. Parametre ayarlama algoritmalarıyla, tahmin edilen parametreleri kullanarak hesaplanan  $y(n)$  sinyali ile adaptif filtrenin izlemesi istenen  $d(n)$  sinyali arasında  $e(n) = d(n) - y(n)$  olarak tanımlanan hata sinyalinin karesinin beklenen değerini minimum yapan parametre tahminleri iteratif olarak hesaplanır (Widrow ve Stearns, 1985, Treichler ve diğ., 1987, Diniz, 1997, Farhang-Boroujeny, 1998, Haykin, 2002).

### 2.1. Eğim Tabanlı Algoritmalar

Bilindiği gibi LMS algoritması, azalan adım optimizasyon yönteminin tahmini bir versiyonudur. Azalan adım yönteminde optimum parametreleri bulmak için, öncelikle başlangıç değerinden başlanarak bir sonraki parametre değerinin hesaplanmasında bir sonraki adımın yönü ve büyüklüğü belirlenir. Bir sonraki adımın yönü ortalama karesel hata fonksiyonunun türeviyle belirlenir. Ortalama karesel hata fonksiyonunu minimum yapan azalan adım optimizasyon yöntemi

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu[\mathbf{p} - \mathbf{R}\mathbf{w}(n)] \quad (1)$$

olarak verilmektedir (Diniz, 1997, Haykin, 2002, Bose, 2004). Burada  $\mu$  seçilen yönde gidilecek adım boyutunu,  $\mathbf{R}$  adaptif filtrenin giriş sinyalinin otokorelasyon matrisini ve  $\mathbf{p}$  ise adaptif filtrenin takip etmesi gereken sinyal ile giriş sinyali arasındaki karşı-korelasyon vektörünü göstermektedir ve sırasıyla

$$\mathbf{R} = E[\mathbf{x}(n)\mathbf{x}^T(n)] \quad , \quad \mathbf{p} = E[\mathbf{x}(n)d(n)] \quad (2)$$

olarak tanımlanmaktadır, Burada  $E$  beklenen değer operatörüdür. Adaptif filtrenin uzunluğu  $M$  olmak üzere adaptif FIR (Finite Impulse Response) filtrenin giriş işaret vektörü  $\mathbf{x}(n)$  ve katsayı vektörü  $\mathbf{w}(n)$

$$\mathbf{x}^T(n) = [x(n) \quad x(n-1) \quad \cdots \quad x(n-M+1)] \quad (3)$$

$$\mathbf{w}^T(n) = [w_1(n) \quad w_2(n) \quad \cdots \quad w_M(n)] \quad (4)$$

olarak tanımlanmaktadır. Pratik uygulamalarda,  $\mathbf{R}$  korelasyon matrisi ve  $\mathbf{p}$  korelasyon vektörünün tam değeri bilinemediği için tahmin edilmiş değeri kullanılır. Azalan adım optimizasyon algoritmasında  $\mathbf{R}$  ve  $\mathbf{p}$  matrislerinin anlık tahminleri

$$\mathbf{R}(n) \cong \mathbf{x}(n)\mathbf{x}^T(n) \quad , \quad \mathbf{p}(n) \cong \mathbf{x}(n)d(n) \quad (5)$$

şeklinde kullanılarak elde edilen aşağıdaki algoritma LMS algoritması olarak bilinmektedir (Widrow ve Stearns, 1985, Diniz, 1997, Farhang-Boroujeny, 1998, Haykin, 2002, Bose, 2004).

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \mathbf{x}(n)e(n) \quad (6)$$

LMS algoritmasının en önemli avantajı algoritmanın gerçekleşmesinde  $(2M+1)$  çarpma işlemi yapılmasıdır. LMS algoritması tahminlerinin kararlı bir şekilde optimum parametre değerlerine yakınsayabilmesi için  $\mu$  adım parametresi

$$0 < \mu < 2/\lambda_{\max} \quad (7)$$

aralığında seçilmelidir. Burada  $\lambda_{\max}$  korelasyon matrisinin en büyük özdeğeridir.

LMS algoritmasındaki en önemli eksikliklerinden biri algoritmanın kararlı kalmasını sağlayacak kadar küçük ve aynı zamanda parametre tahminlerinin kısa zamanda optimum değerine yakınsayabilmesini sağlayacak kadar da büyük bir  $\mu$  adım parametresi seçiminin zorluğudur. LMS algoritmasının, giriş işaretinin filtre uzunluğu kadar kısmının gücü ile normalize edilmesiyle NLMS algoritması elde edilir. NLMS algoritması

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mu \mathbf{x}(n)e(n)}{\alpha + \mathbf{x}^T(n)\mathbf{x}(n)} \quad (8)$$

olarak yazılabilir, burada paydada olası bir sifıra bölme işlemine engel olması için küçük bir  $\alpha > 0$  katsayısı kullanılır. NLMS algoritması tahminlerinin kararlı bir şekilde optimum parametre değerlerine yakınsayabilmesi için  $\mu$  adım parametresi

$$0 < \mu < 2 \quad (9)$$

aralığında seçilmelidir (Treichler ve diğ., 1987, Diniz, 1997, Farhang-Boroujeny, 1998, Haykin, 2002, Haykin ve Widrow, 2003). Her adımda yapılan normalizasyon işlemi  $\mu$  adım parametresinin  $\lambda_{\max}$  özdeğerine olan bağımlılığını ortadan kaldırır. Böylece  $\mu$  adım parametresinin sayısal değeri algoritmanın daha hızlı yakınsayabilmesi için rahatlıkla büyük seçilebilir. NLMS algoritmasının gerçekleştirilmesinde yapılan normalizasyon işleminden dolayı çarpma sayısı ilk bakışta  $(3M + 2)$  gibi görünse de,  $\mathbf{x}^T(n)\mathbf{x}(n)$  ifadesinin değerinin hesaplanması için gereken işlem sayısı  $M$ 'den 1'e düşürülebilir. Burada NLMS algoritmasındaki  $\mathbf{x}^T(n)\mathbf{x}(n)$  çarpımı

$$\mathbf{x}^T(n)\mathbf{x}(n) = x^2(n) + x^2(n-1) + \dots + x^2(n-M+1) \quad (10)$$

şeklinde yazılabilir. Buradan  $\mathbf{x}^T(n+1)\mathbf{x}(n+1)$  çarpımı

$$\mathbf{x}^T(n+1)\mathbf{x}(n+1) = \mathbf{x}^T(n)\mathbf{x}(n) + x^2(n+1) - x^2(n-M+1) \quad (11)$$

şeklinde, daha önce hafızada saklanan toplama bir sonraki adımda sadece  $x^2(n+1)$  çarpımı ilave edilip önceden hesaplanmış olan son terim  $x^2(n-M+1)$  çıkarılarak elde edilebilir. Yani sadece  $x^2(n+1)$  çarpımından dolayı çarpma sayısı 1 olur (Treichler ve diğ., 1987). Bu durumda NLMS algoritmasının gerçekleştirilmesi için yapılan çarpma sayısı LMS algoritmasına göre 2 artarak  $(2M + 3)$  olmaktadır. İzdüşüm (projection) algoritması olarak da adlandırılan NLMS algoritması aynı zamanda  $\mathbf{w}(n+1) - \mathbf{w}(n)$  parametre değişiminin Euclidean normunun karesini  $d(n) = \mathbf{w}^T(n+1)\mathbf{x}(n)$  sınırlamasına göre minimum yapan  $\mathbf{w}(n+1)$  parametrelerinin hesaplanmasıyla da elde edilebilmektedir (Goodwin ve Sin, 1984, Haykin, 2002). AP algoritmasında sınırlama sayısı  $k = 1, 2, \dots, P$  için  $d(n-k) = \mathbf{w}^T(n+1)\mathbf{x}(n-k)$  olmak üzere birden fazla olup,  $P < M$  olmak şartıyla algoritmanın izdüşüm sayısını belirlemektedir ve kullanılan izdüşüm sayısına bağlı olarak işlem yükü artmaktadır (Haykin, 2002, Haykin ve Widrow, 2003).

## 2.2. GS (Gauss-Seidel) Algoritması

GS algoritmasındaki orijinal fikir, en küçük kareler tahminlerinin elde edilmesinde, lineer denklem takımlarının çözümünde kullanılan iteratif yöntemlerden yararlanmaktır. GS algoritması, adaptif filtre katsayılarının ayarlamak için  $\mathbf{R}\mathbf{w}_{opt} = \mathbf{p}$  ile verilen normal denklemin çözümünde kullanılmaktadır. Burada  $\mathbf{w}_{opt}$   $M$  adet filtre katsayısını içeren  $M \times 1$  boyutlu optimum katsayı vektörüdür.  $\mathbf{R}$  matrisi ve  $\mathbf{p}$  vektörü (2) ile verilmiştir. Bilindiği gibi  $\mathbf{R}$  korelasyon matrisi simetrik ve pozitif tanımlı olan bir matristir (Haykin, 2002). Nümerik olarak, doğrusal denklem takımını oluşturan kare matrisin simetrik ve pozitif tanımlı olması durumunda, GS algoritmasının herhangi bir başlangıç değeri için denklem takımını sağlayan çözüm değerine iteratif olarak yakınsadığı matematiksel olarak Golub ve Van Loan (1996) tarafından gösterilmiştir.

GS algoritmasıyla tekrarlamalı parametre tahmin işleminin başlangıç noktası, zaman ortalamalı normal denklemin GS algoritmasıyla çözümü üzerine kuruludur. Çünkü pratik uygulamalarda  $\mathbf{R}$  matrisinin ve  $\mathbf{p}$  vektörünün değerinin bilinmemesi durumunda tahmin edilmiş değerleri kullanılarak bir yaklaşıklık yapılır. GS algoritması  $\mathbf{R}$  korelasyon matrisinin ve  $\mathbf{p}$  korelasyon vektörünün birikimli tahmini değerini kullanır. Bu yaklaşık tahmini değerler  $n$  adet veri grubu kullanıldığında aşağıdaki gibi yazılabilir,

$$\mathbf{R}(n) \cong \frac{1}{n} \sum_{k=1}^n \mathbf{x}(k)\mathbf{x}^T(k) \quad , \quad \mathbf{p}(n) \cong \frac{1}{n} \sum_{k=1}^n \mathbf{x}(k)d(k) \quad (12)$$

veya  $1/n$  çarpanı göz önüne alınmadan veri örnekleri alındıkça iteratif olarak

$$\mathbf{R}(n) = \mathbf{R}(n-1) + \mathbf{x}(n)\mathbf{x}^T(n) \quad , \quad \mathbf{p}(n) = \mathbf{p}(n-1) + \mathbf{x}(n)d(n) \quad (13)$$

şeklinde güncellenebilir. Daha sonra bir adımlık GS iterasyonu  $\mathbf{R}(n)\mathbf{w}(n) = \mathbf{p}(n)$  olarak yazılan zaman ortalamalı normal denklemin çözümünde

$$\mathbf{w}(n+1) = -(\mathbf{R}_L(n) + \mathbf{R}_D(n))^{-1} \mathbf{R}_U(n)\mathbf{w}(n) + (\mathbf{R}_L(n) + \mathbf{R}_D(n))^{-1} \mathbf{p}(n) \quad (14)$$

şeklinde kullanılır, yani iterasyon indisi ayrık zaman indisi olarak alınmıştır. Burada  $\mathbf{R}_L(n)$ ,  $\mathbf{R}_D(n)$  ve  $\mathbf{R}_U(n)$  sırasıyla korelasyon matrisinin (13) ile verilen tahmini değerinin alt üçgen, köşegen ve üst üçgen kısımlarını içeren kare matrisleri gösterir. Burada (14) ile verilen eşitlikte iki işlem birleştirilmiştir. Bunlar, korelasyon matrisi ile korelasyon vektörünün tahmin edilmesi ve GS algoritması ile zaman ortalamalı normal denklemin çözümüdür (Koçal,1998). Pratik uygulamada filtre katsayıları

$$w_i(n+1) = \left[ p_i(n) - \sum_{j=1}^{i-1} R_{ij}(n)w_j(n+1) - \sum_{j=i+1}^M R_{ij}(n)w_j(n) \right] / R_{ii}(n), \quad (15)$$

$(i = 1, 2, \dots, M)$

şeklinde sırayla güncellenir. Burada  $R_{ij}(n)$  korelasyon matrisinin  $i$ . satırına ve  $j$ . sütununa denk düşen elemanını gösterir,  $p_i(n)$  korelasyon vektörünün  $i$ . elemanını gösterir ve  $w_i(n)$  katsayı vektörünün  $i$ . elemanını gösterir. Ayrıca, adaptif FIR filtre kullanıldığında ve korelasyon matrisinin simetrik olduğu göz önüne alındığında aynı iterasyonu giriş işaretinin korelasyon katsayılarına bağlı olarak

$$w_i(n+1) = \left[ p_i(n) - \sum_{j=1}^{i-1} r_{i-j}(n)w_j(n+1) - \sum_{j=i+1}^M r_{j-i}(n)w_j(n) \right] / r_0(n), \quad (16)$$

$(i = 1, 2, \dots, M)$

şeklinde de yazılabilir. Burada  $\mathbf{R}(k)$  korelasyon matrisinin elemanlarını oluşturan oto-korelasyon katsayılarını ve  $\mathbf{p}(k)$  korelasyon vektörünün elemanlarını oluşturan çapraz-korelasyon katsayıları sırasıyla aşağıdaki gibi tanımlanabilir.

$$r_i = E[x(n)x(n-i)] \quad , \quad (i = 0, 1, \dots, M-1) \quad (17)$$

$$p_i = E[d(n)x(n-i)] \quad , \quad (i = 1, 2, \dots, M) \quad (18)$$

Pratik uygulamalarda bu katsayıların tahmini değerleri veri örnekleri alındıkça iteratif olarak  $1/n$  çarpanı göz önüne alınmadan aşağıdaki gibi güncellenebilir.

$$r_i(n) = r_i(n) + x(n)x(n-i) \quad , \quad (i = 0, 1, \dots, M-1) \quad (19)$$

$$p_i(n) = p_i(n) + d(n)x(n-i) \quad , \quad (i = 1, 2, \dots, M) \quad (20)$$

Burada sırasıyla (19) ve (20) ile verilen korelasyon katsayıları tahminlerinin (16) ile birlikte kullanıldığında elde edilen algoritma stokastik GS algoritması olarak adlandırılmıştır (Koçal, 2008).

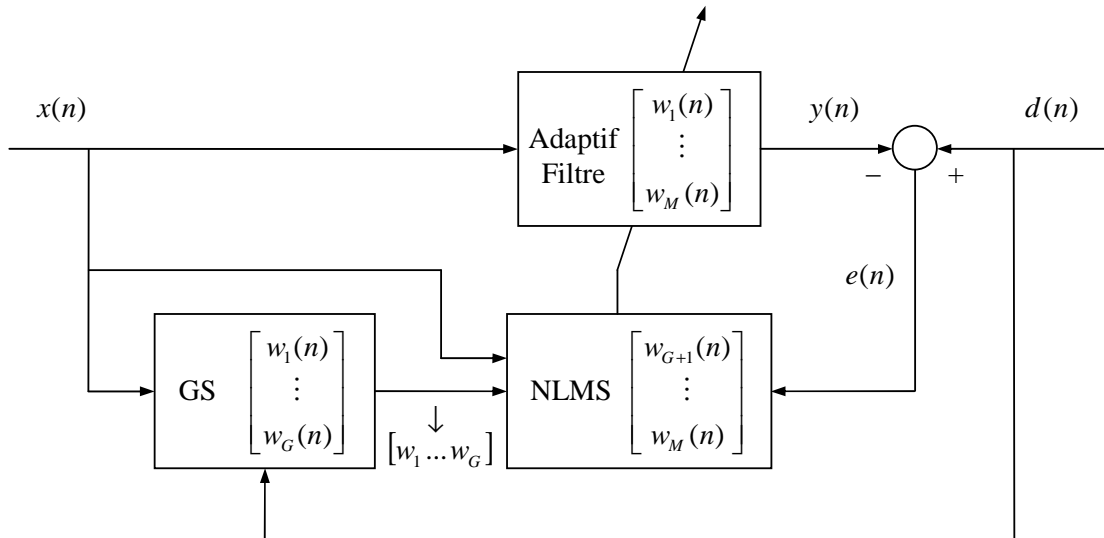
Bu algoritmanın gerçekleştirilmesinde  $(M^2 + 2M)$  adet çarpma işlemi yapılmaktadır. Korelasyon matrisinin (13) ile verilen tahmin edilmiş değerinin pozitif tanımlı olması durumunda stokastik GS algoritması tahminleri optimum değerine yakınsamaktadır (Hatun ve Koçal, 2005).

GS algoritması farklı bir bakış açısıyla bir optimizasyon algoritması şeklinde EDS (Euclidean Direction Search) adı ile de önerilmiş ve bazı adaptif sinyal işleme uygulamalarında kullanılmıştır (Xu ve diğ. 1998, 1999a, 1999b, Bose ve Xu, 2002, Mabey ve diğ. 2004, Bose, 2004). Bu çalışmalarda Gauss-Seidel algoritması adaptif FIR filtre katsayılarının ayarlanmasında kullanılmıştır. Önerilen bu GS tabanlı algoritmaların işlem yükü RLS algoritmasından daha az olup, yakınsama hızı RLS algoritmasına yakındır ve korelasyon matrisinin özdeğer yayılımının küçük olması durumunda RLS algoritmasına çok yakın sonuçlar vermektedir (Koçal, 1998, Bose, 2004).

### 3. HİBRİD GS-NLMS ALGORİTMASI

GS algoritmasının yakınsama hızı RLS algoritmasına yakın olmakla birlikte, gerçekleştirme esnasında  $(M^2 + 2M)$  adet çarpma işlemi yapılması, algoritmanın yüksek örnekleme hızlarında kullanıldığı bazı uygulamalarda dezavantaj oluşturabilmektedir. Bu çalışmada, yakınsama hızı iyi olan GS algoritmasıyla işlem yükü az olan NLMS algoritmaları birlikte kullanılarak işlem yükü GS algoritmasından daha az olan ve NLMS algoritmasından daha hızlı yakınsayan hibrid bir algoritma elde edilmiştir. Önerilen hibrid GS-NLMS algoritmasının adaptif filtreleme işleminde kullanımı Şekil 2'deki blok diyagramda görülmektedir.

Önerilen hibrid GS-NLMS algoritmasında, Şekil 2'de de görüldüğü gibi,  $M$  uzunluklu bir adaptif filtrenin katsayılarının  $1 \leq G < M$  olmak üzere ilk  $G$  tanesi GS algoritması ile güncellenmekte ve geriye kalan  $(M - G)$  tanesi de NLMS algoritması ile güncellenmektedir. Hibrid GS-NLMS algoritmasında bir önceki adımda güncellenmiş katsayı vektörünün ilk  $G$  elemanı GS algoritması kullanılarak güncellenir. NLMS algoritmasında ise öncelikle ilk  $G$  elemanı güncellenmiş katsayı vektörü kullanılarak anlık hata bilgisini hesaplanır, sonra hesaplanan anlık hata bilgisi kullanılarak katsayı vektörünün geriye kalan  $(M - G)$  tane elemanı güncellenir. Elde edilen yeni hibrid algoritmanın gerçekleştirilmesinde  $G(M - 1) + 3M + 2$  adet çarpma işlemi yapılmaktadır, yani önerilen hibrid algoritma işlem yükü açısından GS ile NLMS arasında kalmaktadır. Ayrıca, önerilen hibrid GS-NLMS algoritmasının işlem yükü, GS ve NLMS algoritmaları arasında paylaşılan parametre sayısına bağlı olarak değişmektedir. Önerilen hibrid GS-NLMS algoritmasının gerçekleştirilmesi ve işlem yükü aşağıda detaylı olarak açıklanmaktadır.



Şekil 2:  
Hibrid GS-NLMS algoritması

Önerilen hibrid algoritma iki kısımdan oluşmaktadır. Bu durumda ilk kısımda kullanılan GS algoritmasının kararlı olması için (13) eşitliğindeki gibi hesaplanan korelasyon matrisi tahminlerinin pozitif tanımlı olması durumunda ve aynı zamanda ikinci kısımda kullanılan NLMS algoritmasının kararlı olması için kullanılan adım parametresinin  $0 < \mu < 2$  aralığında algoritma kararlı olacak şekilde seçilmesi durumunda hibrid GS-NLMS algoritması da kararlı olacaktır.

### 3.1. Hibrid GS-NLMS Algoritmasının Gerçeklenmesi ve İşlem Yüğü

Bu kısımda, GS ve NLMS algoritmalarında hesaplanan benzer ifadeler arasındaki bağlantılar belirlenerek hibrid GS-NLMS algoritmasının işlem yükünün azaltılması için gerekli çalışmalar yapılmıştır.

Önce GS algoritmasında kullanılan korelasyon katsayılarıyla NLMS algoritmasında hesaplanan  $\mathbf{x}^T(n)\mathbf{x}(n)$  vektör çarpımı arasında bir ilişki bulunarak hibrid algoritmanın işlem yükünde bir azalma sağlanabilir. NLMS algoritmasındaki  $\mathbf{x}^T(n)\mathbf{x}(n)$  çarpımı

$$\mathbf{x}^T(n)\mathbf{x}(n) = \sum_{i=0}^{M-1} x^2(n-i) \quad (21)$$

şeklinde yazılabilir. Toplam sembolündeki indisler değiştirilerek aynı çarpım

$$\mathbf{x}^T(n)\mathbf{x}(n) = \sum_{i=n-(M-1)}^n x^2(i) \quad (22)$$

olarak da yazılabilir. Diğer taraftan, korelasyon katsayılarının birikimli tahmini değerleri ise  $1/n$  çarpanı göz önüne alınmadan aşağıdaki gibi yazılabilir.

$$r_k(n) = \sum_{i=1}^n x(i)x(i-k) \quad , \quad k = 0,1,\dots,(M-1) \quad (23)$$

Burada  $k = 0$  için sıfırıncı korelasyon katsayısı

$$r_0(n) = \sum_{i=1}^n x^2(i) \quad (24)$$

olarak yazıldığında iki parçanın toplamı halinde

$$r_0(n) = \sum_{i=1}^{n-M} x^2(i) + \sum_{i=n-(M-1)}^n x^2(i) \quad (25)$$

olarak yazılabilir. Bu eşitlikte (22) ve (24) eşitlikleri kullanıldığında

$$r_0(n) = r_0(n-M) + \mathbf{x}^T(n)\mathbf{x}(n) \quad (26)$$

sonucuna ulaşılır. Bu sonuca göre  $\mathbf{x}^T(n)\mathbf{x}(n)$  çarpımı sıfırıncı korelasyon katsayısına bağlı olarak

$$\mathbf{x}^T(n)\mathbf{x}(n) = r_0(n) - r_0(n-M) \quad (27)$$

şeklinde hesaplanabilir. Görüldüğü gibi NLMS algoritmasında bulunan ve (10) ile verilen  $\mathbf{x}^T(n)\mathbf{x}(n)$  çarpımı hibrid algoritmanın gerçekleşmesinde çarpma işlemi yapmadan tek bir çıkarma işlemi kullanılarak hesaplanabilir. GS algoritmasında  $r_0(n)$  korelasyon katsayısı zaten hesaplanmaktadır. Bu durumda hibrid GS-NLMS algoritmasının gerçekleşmesinde ikinci aşamada,  $(M - G)$  tane filtre katsayısını güncellemek için kullanılan NLMS algoritmasında (27) eşitliğinden yararlanıldığında, işlem yükü yapılan 1 adet bölme işlemi ile birlikte  $2(M - G) + 2$  olmaktadır.

Hibrid GS-NLMS algoritmasının gerçekleşmesinde ilk aşamada,  $G$  adet filtre katsayısının güncellenmesinde GS algoritmasının kullanılması durumunda, her bir parametrenin güncellenmesi için  $M$  adet olmak üzere  $G$  adet katsayının güncellenmesi için  $GM$  adet çarpma işlemi yapılmaktadır. Ayrıca,  $M$  adet oto-korelasyon katsayısının güncellenmesinde  $M$  adet çarpma işlemi yapılmaktadır ve  $G$  adet çapraz-korelasyon katsayısının güncellenmesinde  $G$  adet çarpma işlemi yapılmaktadır. Sonuç olarak hibrid algortmada ilk aşamada filtre katsayılarının ilk  $G$  tanesinin GS algoritmasıyla güncellenmesi durumunda  $(GM + G + M)$  adet çarpma işlemi yapılmaktadır. Hibrid algoritmanın ikinci aşamasında geriye kalan  $(M - G)$  tane filtre katsayısının NLMS algoritmasıyla güncellenmesi durumunda ise  $2(M - G) + 2$  adet çarpma işlemi yapılmaktadır. Sonuç olarak hibrid algoritmanın gerçekleşmesinde toplam  $G(M - 1) + 3M + 2$  adet çarpma işlemi yapılmaktadır.

Hibrid algoritmanın gerçekleşmesinde, ilk aşamada  $G$  tane filtre katsayısını güncellemek için kullanılan GS algoritmasındaki parametrelerin yeri de işlem yükünü bir miktar etkilemektedir. Burada  $M$  uzunluklu bir adaptif filtrenin katsayılarını güncellemek için kullanılacak olan GS algoritması daha detaylı olarak aşağıdaki gibi verilebilir.

$$\begin{aligned} w_1(n+1) &= \left\{ p_1(n) - [r_1(n) \quad r_2(n) \quad \cdots \quad r_{M-1}(n)] \cdot \begin{bmatrix} w_2(n) \\ w_3(n) \\ \vdots \\ w_M(n) \end{bmatrix} \right\} / r_0(n) \\ &\vdots \\ w_M(n+1) &= \left\{ p_M(n) - [r_{M-1}(n) \quad r_{M-2}(n) \quad \cdots \quad r_1(n)] \cdot \begin{bmatrix} w_1(n+1) \\ w_2(n+1) \\ \vdots \\ w_{M-1}(n+1) \end{bmatrix} \right\} / r_0(n) \end{aligned} \quad (28)$$

Burada  $w_1(n)$  ile  $w_M(n)$  arasında kalan filtre katsayıları,  $M$  tek olduğunda

$$\begin{aligned} j &= -\left(\frac{M-3}{2}\right), \dots, -1, 0, 1, \dots, \left(\frac{M-3}{2}\right) \quad \text{için} \\ w_{\frac{M+1+2j}{2}}(n+1) &= \left\{ p_{\frac{M+1+2j}{2}}(n) - \left[ r_{\frac{M-1+2j}{2}}(n) \quad \cdots \quad r_1(n) \quad r_1(n) \quad \cdots \quad r_{\frac{M-1-2j}{2}}(n) \right] \cdot \begin{bmatrix} w_1(n+1) \\ w_2(n+1) \\ \vdots \\ w_M(n) \end{bmatrix} \right\} / r_0(n) \end{aligned} \quad (29)$$

şeklinde hesaplanabilir,  $M$  çift olduğunda ise



$$j = -\left(\frac{M}{2}-2\right), \dots, -1, 0, 1, \dots, \left(\frac{M}{2}-1\right) \text{ için}$$

$$w_{\frac{M}{2}+j}(n+1) = \left\{ p_{\frac{M}{2}+j}(n) - \left[ r_{\frac{M}{2}+j-1}(n) \cdots r_1(n) \quad r_1(n) \cdots r_{\frac{M}{2}-j}(n) \right] \cdot \begin{bmatrix} w_1(n+1) \\ w_2(n+1) \\ \vdots \\ w_M(n) \end{bmatrix} \right\} / r_0(n) \quad (30)$$

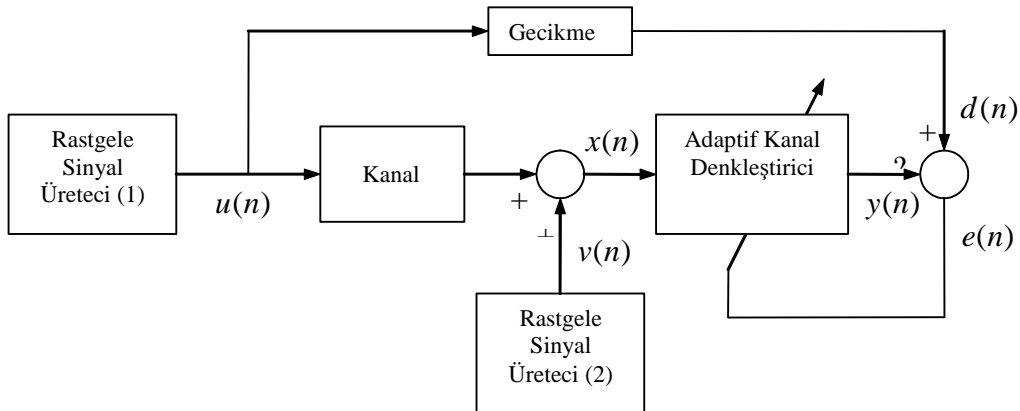
şeklinde hesaplanabilir. Hibrid GS-NLMS algoritmasının ilk aşamasındaki GS algoritmasıyla güncellenen sınırlı sayıdaki  $G$  adet filtre katsayısı,  $\mathbf{w}(n)$  katsayı vektörünün ortasından seçilirse, daha az oto-korelasyon katsayısının güncellenmesiyle birlikte hibrid algoritmadaki toplam çarpma işlemi yükünün bir miktar daha azalacağı görülebilir. Örneğin filtre uzunluğu  $M$  'nin tek olması durumunda ortadaki tek sayıda  $G$  tane filtre katsayısının, ve çift olması durumunda ortadaki çift sayıda  $G$  tane filtre katsayısının GS algoritmasıyla güncellenmesi durumunda  $N = (M + G)/2$  adet oto-korelasyon katsayısı kullanılmaktadır. Bu durumda hibrid algoritmanın ilk aşamasında kullanılan GS algoritmasındaki çarpma işlemi sayısı azalarak  $(GM + G + N)$  olmaktadır. Bu durumda hibrid algoritmanın gerçekleştirilmesinde toplam çarpma işlemi sayısı da bir miktar azalarak  $G(M - 1) + N + 2M + 2$  olmaktadır.

#### 4. SİMÜLASYON SONUÇLARI

Önerilen hibrid GS-NLMS algoritmasının yakınsama özellikleri yapılan bir simülasyon çalışmasıyla örnek bir adaptif kanal denkleştirme problemi üzerinde test edilmiştir ve benzer algoritmalarla birlikte karşılaştırmalı olarak incelenmiştir. Örnek olarak Haykin (2002) ve Bose (2004) tarafından da kullanılan ve darbe cevabı

$$h(n) = \begin{cases} \frac{1}{2} \left[ 1 + \cos\left(\frac{2\pi}{W}(n-2)\right) \right] & , \quad n = 1, 2, 3 \\ 0 & , \quad \text{diğer} \end{cases} \quad (31)$$

olan doğrusal kanal modeli kullanılmıştır. Simülasyon çalışmasında kullanılan sistemin blok diyagramını Şekil 3'te görülmektedir.

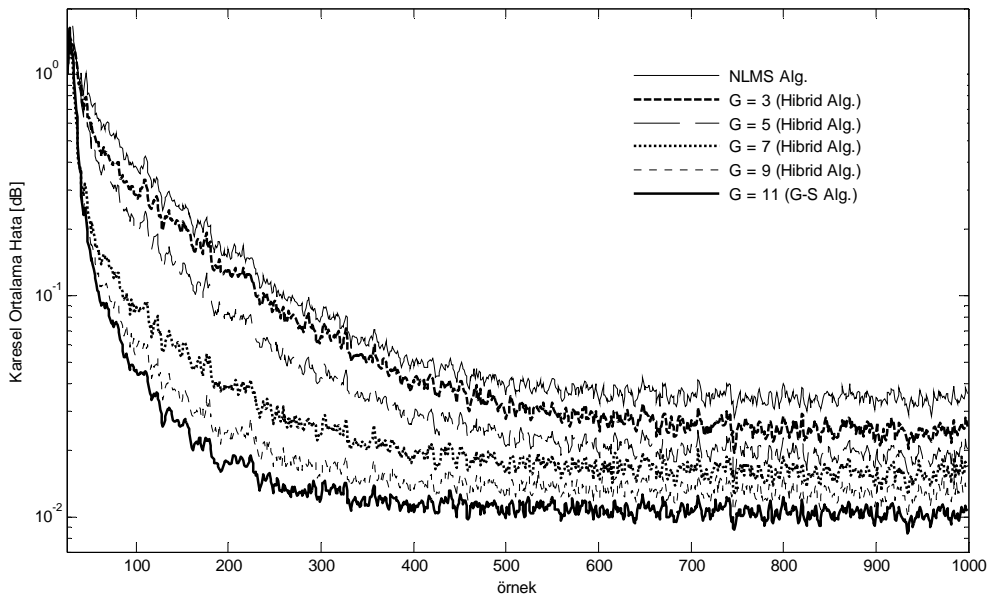


Şekil 3:  
Adaptif kanal denkleştiricinin blok diyagramı

Blok diyagramda görülen rastgele sayı üretici (1) kanalı uyarmak için kullanılan  $u(n)$  test sinyali üretmektedir. Burada  $u(n)$  sinyali  $\mp 1$  seviyeli ve sıfır ortalamalı rastgele sinyaldir. Rastgele sinyal üretici (2) ise kanal çıkışını bozan rastgele gürültü kaynağıdır. Simülasyon çalışmasında sıfır ortalamalı ve varyansı  $\sigma_v^2 = 0.001$  olan normal dağılıma sahip rastgele gürültü dizisi kullanılmıştır. Bu durumda kanal denkleştiricinin giriş işareti

$$x(n) = \sum_{k=1}^3 h(k)u(n-k) + v(n) \quad (32)$$

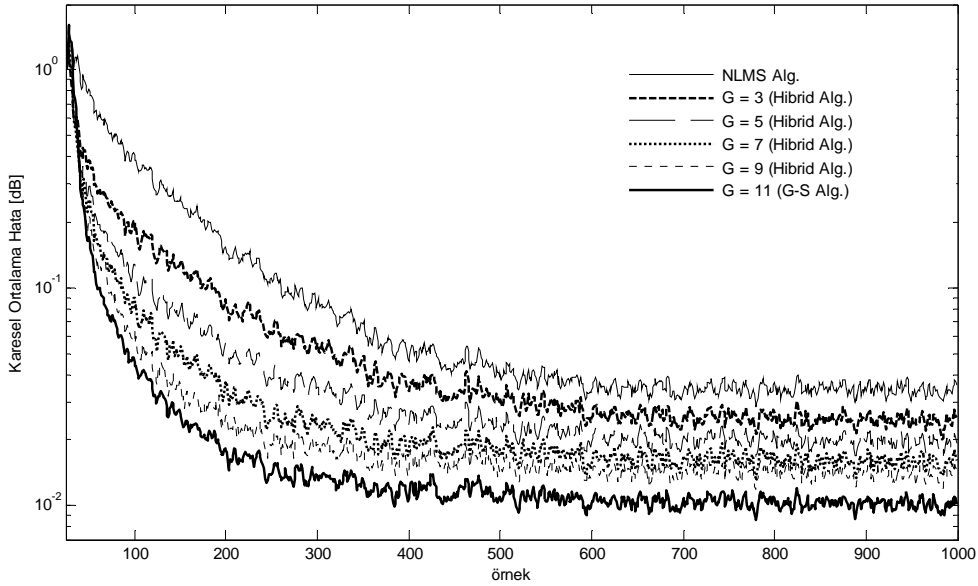
eşitliği kullanılarak hesaplanmıştır. Kanal denkleştirici olarak uzunluğu  $M = 11$  olan adaptif FIR filtre kullanılmıştır. Ayrıca, kanal denkleştiricinin takip etmesi gereken  $d(n)$  sinyali olarak 7 adım geciktirilmiş giriş sinyali kullanılmıştır (Haykin, 2002). Kanalin darbe cevabındaki  $W$  parametresi giriş işaretine ait korelasyon matrisinin özdeğer yayılımını kontrol etmek için kullanılmaktadır. Burada  $W$  değeri arttığında korelasyon matrisinin özdeğer yayılımı da artmaktadır (Haykin, 2002). Bu parametre, yapılan simülasyon çalışmasında özdeğer yayılımının büyük olması için ve algoritmaların yakınsama özellikleri arasındaki farkın belirgin olması için  $W = 3.7$  olarak seçilmiştir. Burada öncelikle GS-NLMS algoritması GS ve NLMS algoritmalarıyla birlikte karşılaştırmalı olarak incelenmiştir. Yapılan simülasyon çalışmasında, NLMS ve önerilen hibrid GS-NLMS algoritmalarında kullanılan adım parametresi  $\mu = 1$  olarak seçilmiştir. Birbirinden farklı giriş sinyallerinin ve ölçme gürültüsünün kullanıldığı 500 adet simülasyonun ortalaması alınarak elde edilen ortalama karesel hata grafikleri Şekil 4'te görülmektedir. Burada hibrid GS-NLMS algoritmasındaki GS algoritmasıyla adaptif FIR filtrenin katsayı vektörünün ilk  $G$  parametresi güncellenmiştir ve GS algoritmasıyla güncellenen  $G$  adet parametre sayısı sırasıyla  $G = 3, 5, 7, 9$  olarak alınmıştır. Elde edilen sonuçlara göre, hibrid GS-NLMS algoritmasının yakınsama hızı GS ve NLMS algoritmalarının arasında kalmaktadır ve GS algoritmasıyla güncellenen  $G$  adet parametre sayısı arttıkça, önerilen hibrid GS-NLMS algoritmasının yakınsama hızının ve hesaplanan ortalama karesel hatanın minimum değerinin GS algoritmasına yaklaştığı görülmektedir.



Şekil 4:

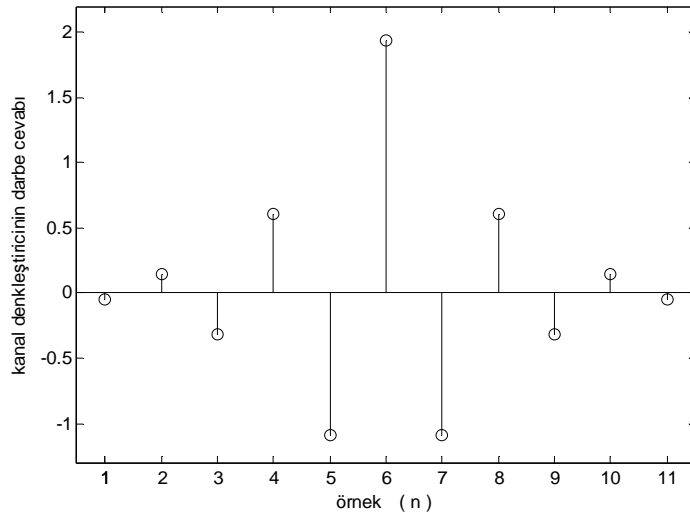
*Hibrid GS-NLMS algoritmasının performansının farklı  $G$  değerlerine bağlı değişimi (Katsayı vektöründeki ilk  $G$  tane katsayı GS algoritmasıyla güncellenmiştir)*

Daha sonra, adaptif FIR filtrenin katsayı vektörünün ortasındaki  $G$  tane parametrenin GS algoritmasıyla güncellendiği ikinci simülasyon çalışması yapılmıştır ve GS algoritmasıyla güncellenen  $G$  adet parametre sayısı sırasıyla  $G = 3, 5, 7, 9$  olarak alınmıştır. Birbirinden farklı giriş sinyallerinin ve ölçme gürültüsünün kullanıldığı 500 adet simülasyonun ortalaması alınarak elde edilen ortalama karesel hata grafikleri Şekil 5'te görülmektedir. Elde edilen sonuçlara göre, yine hibrid GS-NLMS algoritmasının yakınsama hızı açısından GS ve NLMS algoritmalarının arasında kaldığı görülmüştür ve GS algoritmasıyla güncellenen  $G$  adet parametre sayısı arttıkça, önerilen hibrid GS-NLMS algoritmasının yakınsama hızının ve hesaplanan ortalama karesel hatanın minimum değerinin GS algoritmasına yaklaştığı görülmüştür. Yapılan 500 adet simülasyon çalışmasının ortalaması alınarak elde edilen parametre tahminleri, yani adaptif kanal denkleştiricinin darbe cevabı ise Şekil 6'da görülmektedir. Yapılan ikinci simülasyon çalışması sonucunda elde edilen Şekil 5'e bakıldığında, katsayı vektörünün ortasındaki büyük değerli katsayıların GS algoritmasıyla güncellenmesi durumunda, hibrid algoritmanın yakınsama hızının daha belirgin bir şekilde arttığı gözlenmiştir.



Şekil 5:

*Hibrid GS-NLMS algoritmasının performansının farklı  $G$  değerlerine bağlı değişimi (Katsayı vektörünün ortasındaki  $G$  tane katsayı GS algoritmasıyla güncellenmiştir)*



Şekil 6:

*Yapılan simülasyon çalışmasında elde edilen parametre tahminleri (Adaptif kanal denkleştiricinin darbe cevabı)*

Hibrid GS-NLMS algoritmasının işlem yükü açısından diğer algoritmalarla karşılaştırılması durumunda,  $M = 11$  için aşağıdaki Tablo 1'de görülen sonuçlar elde edilir. Bu sonuçlara göre, GS-NLMS algoritmasının işlem yükü, kullanılan  $G$  değerine de bağlı olarak GS ve NLMS algoritmaları arasında kalmaktadır.  $G$  değeri küçük olduğunda işlem yükü NLMS algoritmasına yakın olmakta,  $G < M$  olmak üzere büyük değerler aldığında ise işlem yükü GS algoritmasına yakın olmaktadır.

**Tablo 1.**  
 **$M = 11$  için hibrid GS-NLMS algoritmasının işlem yükü**

Algoritma:	$M = 11$ için işlem yükü:
NLMS Algoritması	25
GS Algoritması	143
Hibrid GS-NLMS ( $G = 1$ )	45
Hibrid GS-NLMS ( $G = 2$ )	55
Hibrid GS-NLMS ( $G = 3$ )	65
Hibrid GS-NLMS ( $G = 5$ )	85
Hibrid GS-NLMS ( $G = 7$ )	105
Hibrid GS-NLMS ( $G = 8$ )	115
Hibrid GS-NLMS ( $G = 10$ )	135

Hibrid algoritmanın gerçekleşmesinde, ilk aşamada GS algoritmasıyla güncellenen sınırlı sayıdaki  $G$  adet filtre katsayısının  $\mathbf{w}(n)$  filtre katsayı vektörünün ortasından seçilmesi durumunda, daha az oto-korelasyon katsayısının kullanılmasından dolayı işlem yükünün bir miktar daha azaldığı Tablo 2'de görülmektedir.

**Tablo 2.**  
 **$M = 11$  için hibrid GS-NLMS algoritmasının işlem yükünün azaltılması**

Algoritma:	İlk $G$ adet katsayı için:	Ortadaki $G$ adet katsayı için:
Hibrid GS-NLMS ( $G = 1$ )	$[w_1] \rightarrow 45$	$[w_6] \rightarrow 40$
Hibrid GS-NLMS ( $G = 2$ )	$[w_1 \ w_2] \rightarrow 55$	$[w_5 \ w_6] \rightarrow 51$
Hibrid GS-NLMS ( $G = 3$ )	$[w_1 \ w_2 \ w_3] \rightarrow 65$	$[w_5 \ w_6 \ w_7] \rightarrow 61$
Hibrid GS-NLMS ( $G = 5$ )	$[w_1 \ \dots \ w_5] \rightarrow 85$	$[w_4 \ \dots \ w_8] \rightarrow 82$
Hibrid GS-NLMS ( $G = 7$ )	$[w_1 \ \dots \ w_7] \rightarrow 105$	$[w_3 \ \dots \ w_9] \rightarrow 103$
Hibrid GS-NLMS ( $G = 9$ )	$[w_1 \ \dots \ w_9] \rightarrow 125$	$[w_2 \ \dots \ w_{10}] \rightarrow 122$

Buradan  $M = 11$  filtre uzunluğu ve farklı  $G$  değerleri için GS algoritmasıyla  $\mathbf{w}(n)$  katsayı vektörünün ortasındaki  $G$  adet filtre katsayısının güncellenmesi durumunda, hibrid algoritmanın toplam işlem yükünde bir miktar daha azalma olduğu görülmektedir. Bu azalma miktarı işlem yükü açısından küçük  $G$  değerleri için daha fazla olmaktadır. Özellikle hibrid GS-NLMS algoritmasında  $\mathbf{w}(n)$  katsayı vektörünün tam ortasındaki 2 katsayısının GS algoritması tarafından güncellenmesi durumunda  $G = 2$  için 51 adet, tam ortadaki 1 katsayısının GS algoritması tarafından güncellenmesi durumunda  $G = 1$  için 40 adet çarpma işlemi yapılmaktadır.

## 5. SONUÇLAR

Bu çalışmada, adaptif filtre katsayılarını ayarlamak için GS ve NLMS algoritmalarının birlikte kullanılmasıyla hibrid bir algoritma oluşturulmuş ve elde edilen yeni algoritmanın yakınsama hızı ve işlem karmaşıklığı benzer algoritmalarla birlikte karşılaştırmalı olarak incelenmiştir. Elde edilen sonuçlara göre, önerilen hibrid algoritmanın işlem karmaşıklığı veya yakınsama hızı açısından bilinen diğer algoritmaların arasında kaldığı ve bir ara yöntem olarak diğer algoritmalara iyi bir alternatif olduğu görülmüştür.

## 6. KAYNAKLAR

1. Bose, T. and Xu, G. F. (2002) The Euclidean direction search algorithm for adaptive filtering, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E85-A(3), 532-539.
2. Bose, T. (2004) *Digital Signal and Image Processing*, John Wiley, New Jersey.
3. Diniz, P. S. R. (1997) *Adaptive Filtering: Algorithms and Practical Implementation*, Kluwer Academic Publishers, Boston.
4. Farhang-Boroujeny, B. (1998) *Adaptive Filters: Theory and Applications*, John Wiley & Sons, Chicester.
5. Golub, G. H. and Van Loan, C. F. (1996) *Matrix Computations*, 3rd Ed., John Hopkins University Press, Baltimore and London.
6. Goodwin, G. C. and Sin, K. S. (1984) *Adaptive Filtering, Prediction and Control*, Prentice-Hall, Englewood Cliffs, New Jersey.
7. Hatun, M. ve Koçal, O. H. (2005) Adaptif filtrelerde Gauss-Seidel algoritmasının stokastik yakınsama analizi, *Uludağ Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, 10(2), 87-92.
8. Haykin, S. (2002) *Adaptive Filter Theory*, 4<sup>th</sup> Ed., Prentice-Hall, Upper Saddle River, New Jersey.
9. Haykin, S., Widrow, B. (editors). (2003) *Least-Mean-Square Adaptive Filters*, Wiley-Interscience, New Jersey.
10. Koçal, O. H. (1998) A new approach to least squares adaptive filtering, *IEEE International Symposium on Circuits and Systems*, Monterey, California, 261-264.
11. Mabey, G.W., Gunther, J., Bose, T. (2004) An Euclidean direction based algorithm for blind source separation using a natural gradient, *IEEE International Conference on Acoustics, Speech and Signal Processing*, 5, 561-564.
12. Treichler, J. R., Johnson, C. R., Larimore, M. G. (1987) *Theory and Design of Adaptive Filters*, Wiley-Interscience, New York.
13. Widrow, B., Stearns, S. D. (1985) *Adaptive Signal Processing*, Prentice-Hall, Upper Saddle River, New Jersey.
14. Xu, G. F., Bose, T., Schroeder, J. (1998) Channel equalization using an Euclidean direction search based adaptive algorithm, *IEEE Global Telecommunication Conference*, 6, 3063-3068.
15. Xu, G. F., Bose, T., Schroeder, J. (1999a) The Euclidean direction search algorithm for adaptive filtering, *IEEE International Symposium on Circuits and Systems*, 3, 146-149.
16. Xu, G. F., Bose, T., Kober, W., Thomas, J. (1999b) A fast adaptive algorithm for image restoration, *IEEE Transaction on Circuits and Systems-I*, 46(1), 216-220.

Makale 12.05.2008 tarihinde alınmış, 13.08.2008 tarihinde kabul edilmiştir. İletişim Yazarı: O. H. Koçal (kocal@uludag.edu.tr).