

AKILLI ANTENLERDE GİRİŞİM BASTIRMA İÇİN LMS ve SMI/LMS ALGORİTMALARININ BAŞARIM DEĞERLENDİRMESİ

Aybike KOCAKAYA¹

Hakan BURDURLU²

Hasan DİNÇER³

^{1,2,3}Elektronik ve Haberleşme Mühendisliği Bölümü
Kocaeli Üniversitesi, 41040, Kocaeli

¹e-posta : adirikol@kou.edu.tr

²e-posta : hburdurlu@kou.edu.tr

³e-posta : hdincer@kou.edu.tr

Anahtar sözcükler: Akıllı anten, uyarlamalı hüzme şekillendirme, LMS ve SMI

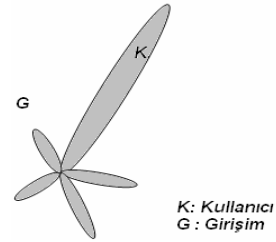
ÖZET

Akıllı antenler, kablosuz iletişim sistemlerindeki girişimleri bastırma, frekans bandını daha verimli kullanabilme ve kapasiteyi artırma gibi özelliklere sahip dizi anten sistemleridir. Bu çalışmada, istenen işaret girişim işaretinden ayırmak için, önce LMS (Least Mean Square) algoritması kullanılmıştır. Daha sonra LMS algoritmasının daha hızlı yakınsaması için SMI (Sample Matrix Inversion) algoritmasıyla birleştirilerek SMI/LMS algoritması oluşturulmuştur. Bu iki algoritma MATLAB ortamında benzetimleri gerçekleştirilerek yakınsama hızları karşılaştırılmıştır.

1. GİRİŞ

Kablosuz iletişim sistemlerinde çok yollu ve eş-kanal girişimi sistem başarımını ve kapasitesini olumsuz etkiler. Çok yollu terimi iletilen işaretin birden fazla yayılım yoluyla alıcıya ulaşması durumu ifade eder. Eş-kanal girişimi ise aynı frekansı kullanan işaretlerin arasındaki girişimdir. Akıllı antenler kablosuz sistemlerinde çok yollu ve eş-kanal girişimini azaltarak kapasite artışı sağlayacak teknolojilerden biridir. Akıllı olan antenin fiziksel yapısı değil, o yapı ile birlikte sayısal işaret işleyicilerini de kapsayan bir sistemdir.

Anten dizisine gelen işaretler karmaşık ağırlıklar (w) ile çarpılıp toplanır. Bu sayede antenin ışma örüntüsü, istenen işaret doğrultusunda en büyük ve girişim işaretleri doğrultusunda en küçük olacak şekilde oluşturulur. Antenin ışma örüntüsünü yukarıda anlatıldığı gibi oluşturmaya “hüzme şekillendirme” denmektedir. Şekil1’de antenin ışma örüntüsünün K kullanıcı işareti doğrultusunda en büyük ve G girişim işareti doğrultusunda en küçük olacak şekilde oluşturulduğu görülmektedir.



Şekil1. Akıllı antenlerde hüzme şekillendirme

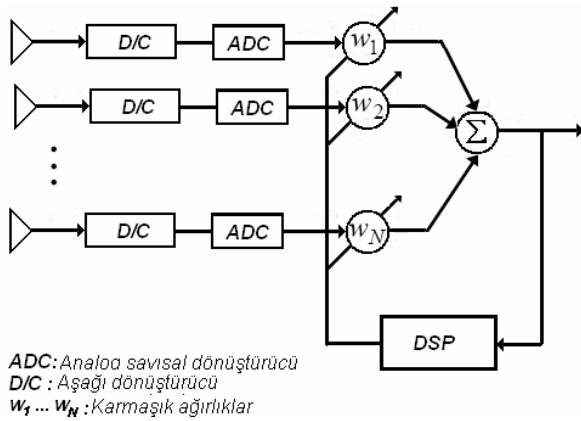
2. AKILLI ANTEN SİSTEMLERİ

Akıllı anten sistemleri anahtarlamalı ve uyarlamalı olmak üzere iki çeşittir. Anahtarlamalı akıllı antenlerde, ışma örüntüsü anten dizisi tarafından oluşturulmuş belli sayıdaki sabit hüzmeden meydana gelir. Sistemin çalışması istenen kullanıcı yönündeki maksimum kazancı sağlayan hüzme aktif hale getirmesi prensibine dayanır. Uyarlamalı anten sistemleri ise ortamdaki istenen işareti, girişim işaretlerini, çok yollu işaretleri ve gürültü değerlerini dikkate alarak ışma örüntüsünü oluşturur. Bu bakımdan uyarlamalı anten sistemleri anahtarlamalı sistemlere göre daha yüksek performansı gösterir. Ancak anahtarlamalı anten sistemleri daha basit ve ucuzdur.

Bu bildiri uyarlamalı akıllı anten tipi üzerinde çalışılmıştır. Akıllı antenler, girişim işaretleri de dahil olmak üzere diziye gelen tüm işaretlerin geliş yönlerini (direction of arrivals) geliş yönü tahmin algoritmalarını kullanarak kestirebilir. Daha sonra istenen işaret saptanır ve istenmeyen işaretlerden ayrılır. Son olarak antenin ışma örüntüsü istenen işaret doğrultusunda yönlendirilir. Genel bir uyarlamalı anten sistemi Şekil2’de görülmektedir.

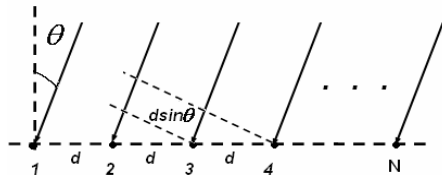
Uyarlamalı bir anten sistemi dört ana kısımdan meydana gelir. Bunlar antenler, radyo arabirimi, hüzme şekillendirici ve işaret işleyicilerdir. Anten dizisi N adet antenden oluşmaktadır. Her bir anten elemanına gelen işaretler bir araya getirilerek tek bir çıkış işareti elde edilir. İşaret önce temel banda çevrilir. Analog-sayısal dönüştürücüler yardımcıyla

İşaret sayısal formata dönüştürülür. İşaret işleyici birim ise karmaşık ağırlık vektörlerini belirler ve her bir anten elemanından gelen işaretle ağırlıkları çarpar. Anten ışınma örüntüsünü şekillendirir. Ağırlıklar iki ölçüte göre optimize edilir. Bunlar istenen kullanıcıdan gelen işaretin en büyük yapılması veya işaret girişim oranı (SIR) en büyük olacak şekilde diğer kullanıcılarından gelen işaretin bastırılmasıdır. Sistem en fazla $N-1$ adet işareti ayırt edebilir. Ancak çok yönlü işaretlerden ötürü bazı sınırlamalar söz konusudur. Huzme şekillendirme sayısal olarak yapıldığı için sayısal işaret işleyici birimde gerçekleştirilir.

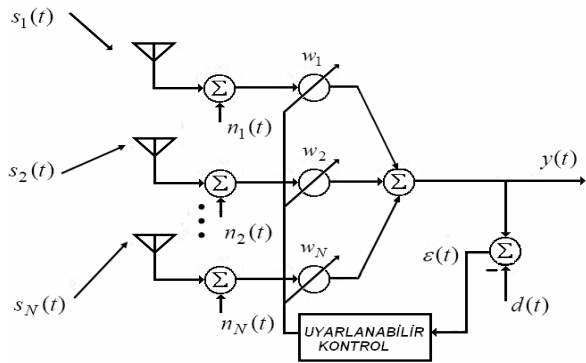


Şekil 2. Uyarlamalı akıllı anten sistemlerinin blok diyagramı

Şekil3'te düzgün doğrusal anten dizisi görülmektedir. Elemanlar arasındaki uzaklık d , işaretin geliş açısı θ 'dir. Her bir eleman arasındaki uzaklık eşittir.



Şekil 3. Düzgün doğrusal anten dizisi



Şekil 4. Uyarlanabilir anten sistemi

Şekil4'te uyarlanabilir bir anten sistemi görülmektedir. Anten dizindeki her bir elemana gelen

$s_i(t)$ işaretleri ile $n(t)$ gürültü işaretlerinin toplamlarının ağırlık vektörleri ile çarpımlarının toplamı dizi çıkışı $y(t)$ 'i verir. Önceki ağırlıklar dizi çıkışı $y(t)$ ve referans işareti $d(t)$ ile w_m ağırlıkları yinelemeli olarak hesaplanır. Referans işareti olarak hem alıcı hem de verici tarafından bilinen bir eğitim dizisidir. Referans işaretinin şekli uyarlanabilir huzme şekillendirme sistemine bağlı olarak değişir. Dizi çıkışı

$$y(t) = w^H x(t) \quad (2.1)$$

olarak ifade edilir.

Optimum ağırlıkları hesaplamak için, dizi yanıt vektörü bilinmesi gereklidir. Dizi yanıt vektörü açının bir fonksiyonudur. N. Antende alınan temel bant işareti, orijinal işaret $s_i(t)$ 'nin zayıflamış ve faz kaymış versiyonlarının toplamlarından oluşmaktadır.

$$x_N(t) = \sum_{i=1}^N a_N(\theta_i) s_i(t) e^{-j2\pi f_c \tau_N(\theta_i)} \quad (2.2)$$

$\tau_k(\theta_i)$ gecikmeyi ve f_c taşıyıcı frekansı ifade eder.

$$a(\theta) = (a_1(\theta) e^{-j2\pi f_c \tau_1(\theta)}, a_2(\theta) e^{-j2\pi f_c \tau_2(\theta)}, \dots, a_N(\theta) e^{-j2\pi f_c \tau_N(\theta)})^T \quad (2.3)$$

$$A(\theta) = [a(\theta_1) \ a(\theta_2) \ \dots \ a(\theta_d)] \quad (2.4)$$

$$S(t) = [s_1(t) \ s_2(t) \ \dots \ s_d(t)]^T \quad (2.5)$$

$$x(t) = A(\theta)S(t) \quad (2.6)$$

Gürültü eklersek,

$$x(t) = A(\theta)S(t) + n(t) \quad (2.7)$$

$a(\theta)$, θ açısı için yönlendirme vektörüdür.

İstenen işaret ile girişim işaretlerinden ayırarak (2.7) denklemini tekrar yazalım. Diziye θ_0 açısı ile $s(t)$ istenen işaret $u_i(t)$ ise θ_i açıları ile diziye gelen N_u adet istenmeyen girişim işaretlerini ifade etmektedir. Dizi çıkışı $x(t)$ işaretini tekrar yazarsak;

$$x(t) = s(t)a(\theta_0) + \sum_{i=1}^{N_u} u_i(t)a(\theta_i) + n(t) \quad (2.8)$$

Sonuç olarak, orijinal $s(t)$ işareti ile kestirilmiş $\hat{s}(t)$ işareti arasındaki hatayı en küçük yaparak, $x(t)$ işaretinden $s(t)$ işaretini kestirmek için uyarlanabilir algoritmalara ihtiyaç vardır. Uyarlanabilir algoritmalar kör algoritmalar ve kör olmayan algoritmalar olarak iki grupta toplamak mümkündür. Kör olmayan algoritmalarda alıcı ve verici tarafından

binen bir eğitim işaretini kullanılarak optimum ağırlık vektörleri hesaplanır. Kör algoritmalar, eğitim işaretine duymazlar. Bunun yerine istenen işaretin bazı özelliklerinden (geliş açısı, işaretin modülü, işaretin zarfı vb.) faydalanmak suretiyle optimum ağırlık vektörlerini hesaplarlar.

3. LMS ve SMI/LMS ALGORİTMALARI

Şekil4'ten de görüldü üzere, w uyarlanabilir kontrol birimi tarafından ayarlanmaktadır. Eğer referans işaretini $d(t)$ biliniyorsa, dizi çıkışı ile referans işaretini arasındaki ortalama karesel hatayı en küçük yapacak şekilde ağırlıklar belirlenir. Optimum ağırlık vektörleri aşağıdaki denklem ile bulunur.

$$\hat{w}_{opt} = \hat{R}^{-1} \hat{r} \quad (3.1)$$

Bu çözüme Wiener çözümü denir. Burada R öz ilişki matrisi r çapraz ilişki vektörü olup aşağıdaki bağıntıları ile hesaplanır.

$$\hat{R} = \sum_{n=1}^N x(n)x^H(n) \quad (3.2)$$

$$\hat{r} = \sum_{n=1}^N x(n)d^*(n) \quad (3.3)$$

Bu metot için istenen ve girişim işaretlerini geliş açılarının bilinmesine gerek yoktur.

En küçük karesel ortalama (LMS) algoritması Widrow&Hoff tarafından 1959 yılında sunulmuştur. En yaygın kullanıma sahip kör olmayan algoritmalar sınıfındadır. Optimum ağırlık vektörlerini hesaplamak için referans işaretine yani, istenen işaret bilgisine ihtiyaç vardır. LMS algoritmasında matris ters alma veya ilinti matrislerini hesabı gerekmediğinden basit bir algoritmadır.

LMS algoritması aşağıdaki denklemler ile ifade edilir.

$$\text{Çıkış, } y(n) = w^H x(n) \quad (3.4)$$

$$\text{Hata, } e(n) = d^* - y(n) \quad (3.5)$$

$$\text{Ağırlık, } w(n+1) = w(n) + \mu x(n)e^*(n) \quad (3.6)$$

Algoritmada kullanılan μ adım büyüklüğü parametresi $w(n)$ ağırlıklarının yakınsamasını kontrol eder. Bu algoritma sürekli olup her örneklenen giriş işaretinden sonra ağırlıklar güncellenir. μ adım büyüklüğü parametresinin değeri öz ilinti matrisinin öz değer yayılımına bağlıdır. Öz değer yayılımı arttıkça algoritmanın yakınsamasını etkileyen adım büyüklüğü parametresinin değeri azalır. Öz değer yayılımının artması N anten sayısının artması ile mümkündür. Ayrıca öz değer yayılımı anten dizine

gelen işaretlerin gücüne de bağlıdır. Adım büyüklüğü parametresi aşağıdaki aralıkta seçilebilir.

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (3.7)$$

İletim sürekli olmadığı zaman SMI gibi blok uyarlamalı yaklaşımlar LMS gibi sürekli yaklaşımlara göre daha iyi sonuç verirler. Ancak blok işlem süresince girişim işaretlerinin sayısı ve konumlarının durağan olması gereklidir. SMI metodu klasik Wiener çözümüne göre daha hızlı yakınsar. Veri bloklara ayrılır ve öz ilişki matrisi ve çapraz ilişki vektörü belli bir zaman aralığı için hesaplanır.

$$\hat{R} = \sum_{n=N_1}^{N_2} x(n)x^H(n) \quad (3.8)$$

$$\hat{r} = \sum_{n=N_1}^{N_2} x(n)d^*(n) \quad (3.9)$$

Ağırlık vektörü

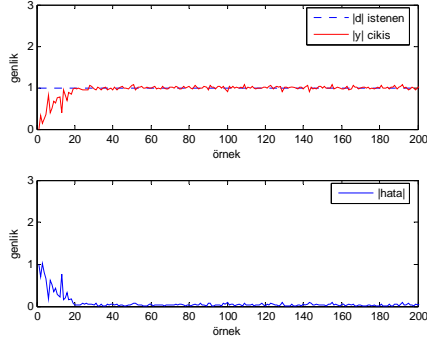
$$\hat{w} = \hat{R}^{-1} \hat{r} \quad (3.10)$$

ile hesaplanır. Her yeni blok için ağırlık vektörü güncellenir. Ağırlıkların güncellenmesi için birçok yöntem bulunmaktadır. Yaygın olarak kullanılan blok uyarlama yönteminde zaman aralığı ayrışık seçilir. İşaret çevresi hızlı değişen sistemler için uygundur. Örtüşmeli blok uyarlama yönteminde zaman aralıkları birbirleriyle örtüşür. Blok uyarlama yöntemini ile karşılaştırıldığında daha iyi sonuç verir. Ancak hesaplama yükü daha fazladır. Hafızalı blok uyarlama yönteminde ise bir önceki bloklar için hesaplanan kestirim matrisleri kullanılır. Bu yöntem yüksek ilişkili uzamsal kanallar için hızlı yakınsama sağlar. Ayrıca işaret çevresi durağan olduğu zaman daha iyi çalışır.

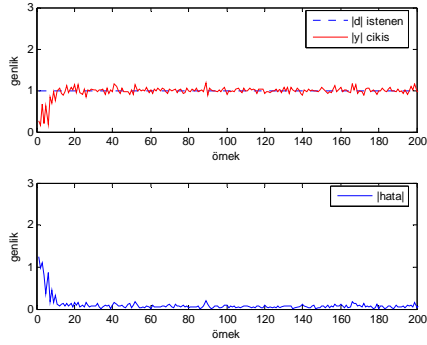
LMS algoritmasının başında ağırlık vektörleri rasgele yada sıfıra eşit alınır. Ancak ağırlık vektörleri daha uygun bir değer seçilerek algoritma başlatılırsa yakınsama daha hızlı olur. İşte bu ağırlık vektörlerinin ilk değerlerini seçmek için örneklenmiş ters matris (Sampled matrix inversion) yöntemi kullanılabilir.

Yapılan benzetimlerde istenen ve girişim işaretleri olarak MSK (Minimum shift keying) işaretleri kullanılmıştır. Sisteme girişintisi 0.01 olan gauss dağılımlı gürültü işareti eklenmiştir. Doğrusal anten dizisi 8 elemanlı olup eleman aralıkları yarım dalga boyudur. Anten dizisine istenen işaret ve 2 adet girişim işareti olmak üzere 3 adet işaret uygulanmıştır.

Öncelikle LMS algoritması için adım büyüklüğü parametresi sırasıyla 0.015 ve 0.03 seçilerek şekil.5a ve şekil.5b'de elde edilen çıkış ve hata işaretleri elde edilmiştir.

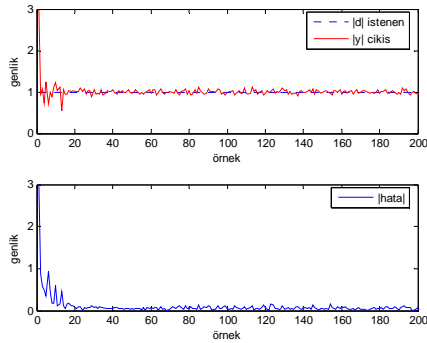


Şekil.5a $\mu = 0.015$ için LMS algoritması

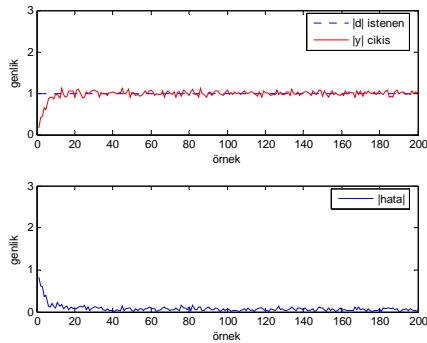


Şekil.5b $\mu = 0.03$ için LMS algoritması

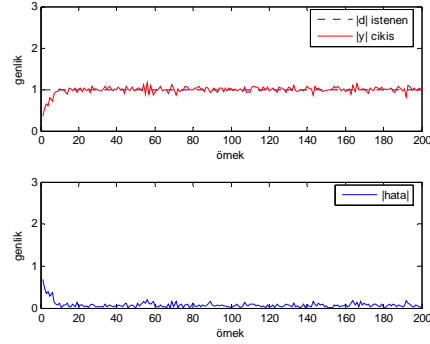
Daha sonra SMI/LMS algoritması adım büyüklüğü parametresi 0.03 alınarak, blok büyüklüğü 10, 20 ve 50 alınarak elde edilen çıkış ve hata işaretleri sırasıyla şekil.6a, şekil.6b ve şekil.6c’de verilmiştir. Ağırlık vektörlerini güncellemek için hafızalı blok uyarlama yöntemi kullanılmıştır.



Şekil.6a Blok sayısı=10 için SMI/LMS algoritması



Şekil.6b Blok sayısı=20 için SMI/LMS algoritması



Şekil.6c Blok sayısı=50 için SMI/LMS algoritması

4. SONUÇLAR ve TARTIŞMALAR

Şekil.5a ve şekil.5b’de adım büyüklüğü parametresinin LMS algoritmasının yakınsamasına etkisi gösterilmiştir. Giriş işaretin öz ilişki matrisinin öz değerlerine bağlı olarak seçilen adım büyüklüğü parametresi $0 < \mu < 0.2141$ aralığında seçilmiştir. Bu aralıkta μ değeri azaldıkça yakınsama yavaşlamaktadır.

Şekil.6a, şekil.6b ve şekil.6c de SMI/LMS algoritmasında blok büyüklüğüne bağlı olarak yakınsama hızı incelemesinin sonuçları görülmektedir. Blok büyüklüğü arttıkça ağırlık vektörlerinin başlangıç değerleri Wiener çözümü ile elde edilen optimum değerlere yaklaştığından yakınsama hızı da artmaktadır.

Benzetimlerden elde edilen sonuçlara göre LMS algoritmasının yakınsaması SMI/LMS algoritmasının yakınsamasına göre daha yavaştır. SMI/LMS algoritmasının daha hızlı yakınsamasının sebebi başlangıç ağırlık vektörü katsayılarının matris ters alma işlemleri yardımıyla belirlenmiş olmasıdır.

Gelecek çalışmalarda SMI/LMS algoritması farklı işaret yayılım çevreleri için test edilecektir. Ayrıca bu algoritmaların Texas Instruments TMS320C6701 EMV modülü kullanılarak gerçek zamanlı olarak gerçekleştirilmesi amaçlanmaktadır.

KAYNAKLAR

- [1]. Haykin S., Adaptive Filter Theory, fourth edition, Prentice Hall Information System Science Series, New Jersey, 2002.
- [2]. Libert, J.R.Jr. and Rappaport T.S., Smart Antennas for Wireless Communications: IS-95 and Third Generation Applications, Prentice Hall Upper Saddle River, New Jersey, .
- [3]. Rong Z. Simulation of Adaptive Array Algorithms for CDMA Systems, MS. Thesis, Virginia PI&S Univ., USA