

57447

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ÖZYİNELİ EN KÜÇÜK KARELER
KAFES ALGORİTMASI KULLANARAK
KANAL ÖZDEŞLEME

Elektronik ve Haberleşme Müh. S. Berrin ÖNER

F.B.E. Elektronik ve Haberleşme Mühendisliği
Anabilim Dalında hazırlanan

YÜKSEK LİSANS TEZİ

Tez Danışmanı : Yrd. Doç. Dr. Soner ÖZGÜNEL

İSTANBUL, 1996

İÇİNDEKİLER

TEŞEKKÜR.....	v
ÖZET.....	vi
ABSTRACT.....	vii
BÖLÜM 1. GİRİŞ.....	1
BÖLÜM 2. WIENER SÜZGEÇLER.....	4
2.1. Optimum Doğrusal Süzgeçleme Problemi.....	4
2.2. Diklik Kuralı.....	5
2.3. Minimum Karesel Ortalama Hata.....	9
2.4. Wiener-Hopf Eşitlikleri.....	11
2.4.1. Doğrusal Enine Süzgeçler İçin Wiener-Hopf Eşitliklerinin Çözümü.....	12
2.4.2. Wiener-Hopf Eşitliklerinin Matris Formu.....	13
2.5. Hata Başarım Yüzeyi.....	15
2.5.1. Minimum Karesel Ortalama Hata.....	17
2.5.2. Hata Başarım Yüzeyinin Kanonik Biçimi.....	19
BÖLÜM 3. DOĞRUSAL ÖNGÖRÜ	21
3.1. İleri Yönde Doğrusal Öngörü.....	21
3.1.1. İleri Yönde Öngörü Hata Süzgeci.....	25
3.1.2. İleri Yönde Öngörü İçin Genişletilmiş Wiener-Hopf Eşitlikleri.....	26
3.2. Geri Yönde Doğrusal Öngörü.....	27
3.2.1. Geri Yönde ve İleri Yönde Öngörücüler Arasındaki İlişkiler.....	30
3.2.2. Geri Yönde Öngörü Hata Süzgeci.....	31
3.2.3. Geri Yönde Öngörü İçin Genişletilmiş Wiener-Hopf Eşitlikleri.....	34

3.3. Levinson-Durbin Algoritması.....	35
3.3.1. Γ_m ve Δ_{m-1} Parametrelerinin Açıklanması.....	40
3.3.2. Levinson-Durbin Özyinelemesinin Uygulanması.....	42
3.3.3. Levinson-Durbin Özyinelemesinin Ters Biçimi.....	43
3.4. Kafes Öngörücüler.....	44
3.4.1. Öngörü Hatalarının Derece Güncelleme Özyinelemeleri.....	44
3.4.2. Kafes Süzgeçlerin İlişki Özellikleri.....	49
3.5. Gram-Schmidt Dikleştirme Yöntemiyle İleri Ve Geri Yönde Öngörü Hata Dizilerinin Oluşturulması.....	51
3.6. Birleşik Süreç Kestirimi.....	52
 BÖLÜM 4. UYARLAMALI SÜZGEÇLERİN YAPISI.....	56
 BÖLÜM 5. EN DİK İNİŞ YÖNTEMİ.....	59
5.1. En Dik İniş Algoritması.....	59
5.2. En Dik İniş Algoritmasının Kararlılığı.....	61
 BÖLÜM 6. EN KÜÇÜK KARESEL ORTALAMA (LMS: LEAST MEAN SQUARE) ALGORİTMASI KULLANILARAK GERÇEKLEŞTİRİLEN UYARLAMALI ENİNE SÜZGEÇLER.....	64
6.1. En Küçük Karesel Ortalama (LMS) Uyarlama Algoritması.....	64
6.2. LMS Algoritmasının Kararlılık Analizi.....	66
6.3. Ortalama Tap Ağırlık Vektörünün Davranışı.....	67
6.4. Ağırlık Hatasının İlişki Matrisi.....	69
6.5. Karesel Ortalama Hata Davranışı.....	70
6.6. Karesel Ortalama Hatanın Geçici Hal Davranışının Özellikleri.....	77
6.7. LMS Algoritmasının Özeti.....	78

BÖLÜM 7. ÖZYİNELİ EN KÜÇÜK KARELER

(RLS: RECURSIVE LEAST SQUARES) ALGORİTMASI

KULLANILARAK GERÇEKLEŞTİRİLEN

ENİNE SÜZGEÇLER.....80

7.1. Matris Tersi Alma Yasası..... 84

7.2. Üssel Ağırlıklı Yinelemeli En Küçük Kareler Algoritması..... 84

7.2.1. Tap Ağırlık Vektörünün Zaman Güncellemesi 86

7.2.2. Başlangıç Koşulları..... 87

7.3. Ağırlıklı Hata Kareleri Toplamının Güncelleme Yinelemeleri..... 89

7.4. RLS Algoritmasının Karesel Ortalama Olarak Yakınsama

Analizi..... 90

BÖLÜM 8. ÖZYİNELİ EN KÜÇÜK KARELER KAFES

SÜZGEÇLER (RLSL: RECURSIVE LEAST SQUARES

LATTICE FİLTERS)..... 95

8.1. Derece Güncelleme Yinelemeleri..... 98

8.1.1. $\Delta_{m-1}(n)$ ve $\Delta'_{m-1}(n)$ Parametreleri Arasındaki İlişki..... 102

8.1.2. Ardıl Öngörü Hatalarının Derece Güncelleme

Özyinelemeleri..... 103

8.1.3. Ardıl Öngörü Hata Karelerinin Ağırlıklı Toplamının

Derece Güncelleme Özyinelemeleri..... 106

8.1.4. Dönüşüm Faktörü $\gamma_m(n-1)'$ in Derece Güncelleme

Özyinelemeleri..... 107

8.2. Zaman Güncelleme Özyinelemeleri..... 108

8.3. RLST Algoritmasının Özeti..... 112

BÖLÜM 9. UYARLAMALI KAFES ALGORİTMASI

KULLANILARAK KANAL ÖZDEŞLEME..... 115

9.1. Özdeşleme Kavramı..... 115

9.2. Kanal Özdeşleme Probleminde Uyarlamalı Kafes Süzgeç

Kullanılması..... 116

9.3. Bilgisayar Simülasyonu..... 122

9.3.1. Kafes Derecesinin Karesel Ortalama Hataya Etkisinin	
İncelenmesi.....	123
9.3.2. Adım Uzunluğunun Karesel Ortalama Hataya Etkisinin	
İncelenmesi.....	152
9.3.3. Kanal Parametresi W' nin Karesel Ortalama Hataya Etkisinin	
İncelenmesi	156
9.3.4. Gürültünün Değişintisinin Karesel Ortalama Hataya Etkisinin	
İncelenmesi.....	160
 BÖLÜM 10. SONUÇLAR.....	164
 KAYNAKLAR.....	166
EK.....	168

TEŞEKKÜR

Yüksek Lisans Tezimin hazırlanışında, değerleri bilgilerini ve yardımlarını esirgemeyen değerli hocam Yrd. Doç. Dr. Soner ÖZGÜNEL' e ve çalışmalarım sırasında bana destek olan aileme teşekkürü bir borç bilirim.

S. Berrin ÖNER

ÖZET

Doğrusal kanalların özdeşlenmesi, son yıllarda bir çok araştırmanın ilgi odağı haline gelmiştir. Bu konuda oldukça fazla sayıda yayım bulunmaktadır. Bu çalışmada, Özyineli En Küçük Kare Kafes Algoritması (RLSL) kullanılarak doğrusal kanalların özdeşlenmesi problemi incelenmektedir.

Uyarlamalı bir alıcının tasarlanmasıyla ilgili yaklaşımlardan biri olan “ kanal özdeşleme” de amaç, önce ilgili kanal parametrelerinin kestirilmesi ve daha sonra bunların kullanılmasıyla alıcının kendisinin iletim ortamına uyarlanması sağlanmasıdır. Burada, bilinmeyen kanalı özdeşlemek üzere, bir model kullanılmaktadır. Bu modelin girişi, kanal giriş işareti olmak üzere elde edilen çıkış işareti ile kanal çıkışı arasındaki fark bir uyarlamalı algoritmada kullanılarak, kanal ile model arasındaki sapma giderilmeye çalışılmaktadır. Böylece modelin, bilinmeyen kanalı özdeşlemesi sağlanmış olur.

Uyarlamalı kanal özdeşlemesiyle ilgili olarak bir çok alıcı modeli kullanılmaktadır. Uyarlamalı algoritmaların çıkarılması için, üç temel yöntem tanımlanabilir. Bu yöntemler, Wiener Süzgeç Kuramı, Kalman Süzgeç Kuramı ve En Küçük Kareler yöntemidir. Wiener Süzgeçleme probleminin çözümü optimumdur ve Wiener çözümü olarak belirtilmektedir. Bu kuram üzerine kurulan uyarlamalı algoritmalarından bazıları; En Küçük Karesel Ortalama Algoritması (LMS), Özyineli En Küçük Kareler Algoritması (RLS) ve Uyarlamalı Kafes Algoritmalarıdır. Bu çalışmada kullanılan Özyineli En Küçük Kareler Kafes Algoritması, süzgeç katsayılarının güncellenmesinde kullanılmak üzere, dik işaret bileşenleri kümesi oluşturarak yakınsama hızının artmasını sağlayan oldukça verimli bir yapıya sahiptir. Bu bileşenlerin elde edilmesi Gram-Schmidt dikleştirme yöntemiyle sağlanmaktadır.

Bu çalışmada, özdeşlemek üzere bir kanal modeli seçilmekte ve kafes algoritması kullanılarak bu kanalı özdeşlenmesi sağlanmaktadır. Yapılan bilgisayar programında, kanal, gürültü ve algoritmaya ilişkin parametreler değiştirilerek algoritmanın başarımı, en küçük karesel ortalama hata anlamında incelenmektedir. Bilgisayar simülasyonu ile elde edilen sonuçlar yorumlanmakta ve sistemin başarımı değerlendirilmektedir.

ABSTRACT

In recent years, channel identification has received a great deal of attention from investigators. There are many studies on this subject. In this study, the identification of linear channels using Recursive Least Squares Lattice Algorithm (RLSL) has been investigated.

One of the approaches to design an adaptive receiver is the “channel identification”, in which, first the relevant channel parameters are estimated and then fed to an optimum receiver to adjust itself to the transmission medium. Here, in order to identify the unknown channel, a model is used, which has the channel input as its input, and the difference between its output and the channel’s output is used in the adaptive algorithm to compensate the deviation between the channel and the model, so obtaining a model which identifies the unknown channel.

For adaptive channel identification, several types of adaptive receivers are used. Basically, there may be identified three distinct methods for deriving recursive algorithms for the operation of adaptive receivers. These methods are based on Wiener Filter Theory, Kalman Filter Theory and Method of Least Squares. The solution of the Wiener filtering problem is optimal. Some of the adaptation algorithms based on these theories are the Least Mean Square Algorithm (LMS), the Recursive Least Squares Algorithm (RLS) and the Adaptive Lattice Algorithms. The Adaptive Lattice Algorithms represent a highly efficient structure for generating a set of orthogonal signal components, which is used for updating the tap weights, so making the convergence rate faster.

In this thesis, Recursive Least Squares Lattice Algorithm is used for the identification of a channel. In the computer simulations, by changing some parameters of the channel, noise and the algorithm, the performance of the algorithm, based on the mean square error, is investigated. The results obtained in the simulations are interpreted, and the performance of the system is discussed.

BÖLÜM 1. GİRİŞ

Bu çalışmada, Özyineli En Küçük Kareler Kafes (RLSL) algoritması kullanarak doğrusal kanalların özdeşlenmesi problemi incelenmektedir. Doğrusal bir kanal modeli seçilerek, bu kanalın Özyineli En Küçük Kareler Kafes (RLSL) algoritması yardımıyla özdeşlenmesine ilişkin bilgisayar simülasyonu yapılmaktadır. Yapılan bilgisayar simülasyonunda, kullanılan parametreler değiştirilerek sonuçların değişimi gözlenmekte ve yorumlanmaktadır.

Uyarlamalı süzgeçler, ilgili işaretin karakteristiklerinin tümünün bilinmediği bir ortamda süzgecin yeterli olarak çalışmasını sağlayan bir özyineli algoritmaya dayanır. Algoritmanın çalışması için, ortamdan tamamen bağımsız olan bazı başlangıç koşullarının önceden belirlenmesi gerekmektedir. Uyarlamalı süzgeçlere ilişkin bu özyineli algoritmaların çıkarılmasında kullanılan yaklaşımlar üç temel grupta toplanır:

1. Wiener Süzgeç kuramına dayalı yaklaşım
2. Kalman Süzgeç kuramına dayalı yaklaşım
3. En Küçük Kareler yöntemine dayalı yaklaşım

Bu yaklaşımlardan biri olan Wiener Süzgeç kuramı, girişlerin durağan olması halinde iyi sonuçlar vermektedir. Wiener Süzgeç kuramının yetersiz olduğu, girişlerin durağan olmadığı durumlarda ise Kalman Süzgeç kuramı kullanılır. İlk iki yaklaşım, istatistiksel kavramlara dayalıdır, yani istatistiksel ön bilgi kullanırlar. En Küçük Kareler yöntemine dayalı yaklaşım ise deterministik yapısı itibarıyla diğer iki yaklaşımdan farklıdır.

Uyarlamalı süzgeçler, bilinmeyen bir ortamda etkin çalışma yeteneği ve aynı zamanda giriş istatistiklerinin değişimlerini izleyebilme özelliği sayesinde farklı bir çok alanda başarıyla uygulanırlar. Uyarlamalı süzgeçlerin haberleşme, kontrol, radar, sonar, deprem bilimi, görüntü işleme ve örnek tanıma gibi farklı bir çok alanda uygulamaları

mevcuttur. Bu alanlarda uyarlamalı süzgeçlerin kullanılması, bir kaç temel başlık halinde şu şekilde verilebilir:

1. Sistem özdeşleme (System identification)
2. Uyarlamalı dengeleme (Adaptive equalization)
3. Yankı giderme (Echo cancellation)
4. Uyarlamalı biçimlendirme (Adaptive beamforming)
5. İşaret dedeksiyonu (Signal dedection)
6. Özbağımlımlı spektrum analizi (Autoregressive -AR- spectrum analysis)

Kanal dengeleme probleminin çözümünde kullanılan uyarlamalı süzgeçler; sistem özdeşleme, ses işareti işleme, yankı giderme gibi problemlerin çözümünde de oldukça iyi sonuçlar vermektedir. Özellikle Özyineli En Küçük Kareler Kafes süzgeç yapısı, hem yakınsama, hem de doğru sonuca çabuk ulaşması bakımından daha çok tercih edilir bir süzgeç türüdür. Bu durum Satorius ve diğerlerinin (1979) çalışmalarından açıkça görülmektedir. Falconer ve diğerleri de (1978) diğer bir yöntem olan Kalman Süzgeç kuramını kullanarak kanal dengeleme problemini incelemişlerdir.

Tıp alanında hastalıkların teşhisinde kullanılan EP (Evoked Potentials) kestiriminde uyarlamalı süzgeç yapısı Thakor ve diğerleri (1987) tarafından kullanılmıştır.

Bu çalışmada, Özyineli En Küçük Kareler Kafes (RLSL) algoritması kullanarak, doğrusal bir kanalın özdeşlenmesi problemi incelenmektedir. Bu amaçla öncelikle, kanal özdeşleme probleminin ne olduğu ve bu problemin çözümünün Özyineli En Küçük Kareler Kafes (RLSL) algoritması kullanılarak nasıl gerçekleştirileceği incelenmektedir. Tüm bunlar sağlandıktan sonra, özdeşlemek üzere bir kanal modeli seçilmekte ve kafes algoritması kulanılarak bu kanalın özdeşlenmesi sağlanmaktadır. Yapılan bilgisayar programında, kanal, gürültü ve algoritmaya ilişkin parametreler değiştirilerek algoritmanın başarımı, en küçük karesel ortalama anlamında incelenmektedir. Bilgisayar simülasyonuyla elde edilen sonuçlar yorumlanmakta ve sistemin başarımı değerlendirilmektedir.

Bu tezin düzenleniş biçimi ve bölümlerin içerikleri şöyledir:

Bölüm 2' de, uyarlamalı süzgeçlerdeki özyineli algoritmaların çıkarılmasına temel teşkil etmesi bakımından Wiener süzgeç kuramı tanıtılmakta ve Wiener-Hopf eşitlikleri verilmektedir.

Bölüm 3' de, Wiener süzgeç kuramı kullanılarak İleri ve Geri Yönde Doğrusal Öngörü (Forward and Backward Linear Prediction) incelenmektedir. Ayrıca Levinson-Durbin algoritmaları yardımıyla kafes süzgeç yapısı oluşturulmaktadır. Son olarak, bu bölümde Birleşik Süreç kestirimi (Joint-Process estimation) verilmektedir.

Bölüm 4, genel olarak bir uyarlamalı süzgecin yapısı ve işleyişi ile ilgilidir.

Bölüm 5' de, bir optimizasyon tekniği olan ve hata başarımlı yüzeyinin kendisini bilmeksizin onun minimum noktasını bulan En Dik İniş (Steepest Descent) yöntemi incelenmektedir. Bu yöntemin burada verilmesinin nedeni; diğer algoritmaların daha kolay anlaşılmasının sağlanmasıdır.

Bölüm 6' da, En Dik İniş algoritmasından faydalanılarak, Wiener Süzgeç kuramına dayalı bir yaklaşım olan En Küçük Karesel Ortalama (Least Mean Square, LMS) algoritması çıkarılmakta ve karesel ortalama hata kavramı incelenmektedir.

Bölüm 7' de, En Küçük Kareler (Least Squares, LS) yöntemine dayalı bir yaklaşım olan Özyineli En Küçük Kareler (Recursive Least Squares, RLS) algoritmasına ilişkin özyinelemeler çıkarılmakta ve RLS algoritmasının karesel ortalama olarak yakınsama analizi yapılmaktadır. Son olarak bu bölümde, RLS ve LMS algoritmaları karşılaştırılmaktadır.

Bölüm 8' de, En Küçük Kareler yöntemine dayalı başka bir yaklaşım olan Özyineli En Küçük Kareler Kafes (Recursive Least Squares Lattice, RLSTL) algoritmasına ilişkin derece ve zaman güncelleme özyinelemeleri çıkarılmaktadır.

Bölüm 9, kafes algoritması kullanarak kanal özdeşleme probleminin incelenmesine ilişkindir. Bu bölümde, değişik kanal modelleri seçilerek bilgisayar simülasyonlarıyla kanal özdeşlemeleri yapılmakta ve yorumlanmaktadır.

10. Bölüm' de ise bu çalışmada uygulanan yöntem ve elde edilen sonuçlar kısaca verilmektedir.

BÖLÜM 2. WIENER SÜZGEÇLER

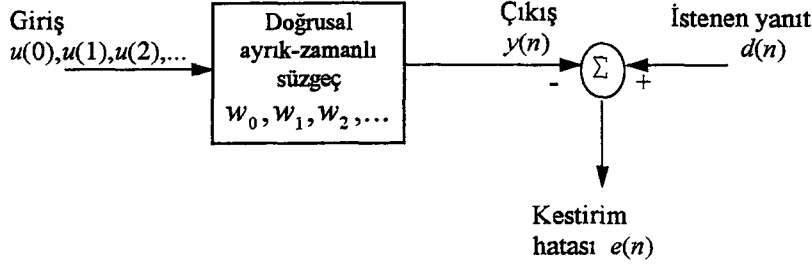
Bu bölümde, optimum doğrusal ayrık zamanlı süzgeçlerin bir sınıfı olarak bilinen Wiener Süzgeçleri tanıtılmaktadır. Burada, doğrusal süzgeçleme probleminin çözümü için, bazı istatistiksel parametrelerin bilindiği varsayılmaktadır. Doğrusal bir süzgeç, çıkışında gürültünün etkisini, bazı hata kriterlerine göre, minimum yapacak şekilde oluşturulur. Örneğin, hata işaretinin karesel ortalama değerinin minimum yapılması gibi. Hata işareti, istenen yanıtla süzgeç çıkışı arasındaki fark olarak tanımlanır. Durağan girişler için elde edilen sonuç Wiener süzgeçtir. Bu süzgeç, karesel ortalama anlamında optimumdur. Wiener süzgeci durağan olmayan girişler söz konusu olduğunda yetersizdir. Bu problemin çözümü Kalman süzgecinde söz konusudur. Gerçekte, Wiener süzgeci Kalman süzgecinin özel bir durumudur.

Wiener süzgeçleri üzerindeki çalışmamıza optimum doğrusal süzgeçleme problemini ele alarak başlayacağız.

2.1. Optimum Doğrusal Süzgeçleme Problemi

Doğrusal ayrık zamanlı bir süzgecin blok diyagramının Şekil 2.1' deki gibi olduğunu düşünelim. Süzgeç girişi, $u(0)$, $u(1)$, $u(2)$, ... biçiminde bir zaman serisinden oluşmakta ve süzgeç, w_0 , w_1 , w_2 , ... impuls yanıtlarıyla karakterize edilmektedir.

Süzgeç, $y(n)$ ile gösterilen çıkışı üretmektedir. Öyle ki, bu çıkış $d(n)$ ile gösterilen istenen yanıtın kestirimidir. $e(n)$ ile gösterilen kestirim hatası, istenen yanıt $d(n)$ ile süzgeç çıkışı $y(n)$ arasındaki farktır. Kestirim hatası $e(n)$ ' nin mümkün olduğunca küçük olması gerekir. Bu problem optimum süzgeçleme problemi olarak adlandırılır. Optimum süzgeçleme probleminin matematiksel çözümü için kullanılan temel kurallardan birisi diklik kuralı olarak bilinen önemli bir teoremdir.



Şekil 2.1 İstatistiksel Süzgeçleme Probleminin Blok Gösterilimi

2.2. Diklik Kuralı

İstatistiksel süzgeçleme probleminin Şekil 2.1 ile tanımlandığını düşünelim. $u(0)$, $u(1)$, $u(2)$, ... zaman serisi ile gösterilen süzgeç girişinin ve w_0 , w_1 , w_2 , ... ile gösterilen süzgeç impuls yanıtının karmaşık değerli ve sonsuz süreli olduğunu kabul edelim. n anındaki süzgeç çıkışı $y(n)$ doğrusal konvolüsyon toplamı,

$$y(n) = \sum_{k=0}^{\infty} w_k * u(n-k), \quad n = 1, 2, \dots \quad (2.1)$$

ile tanımlanmaktadır.

Süzgeç girişi ile istenen yanıt $d(n)$ ' nin sıfır ortalamalı, birleşik geniş anlamda durağan rasgele süreçler oldukları kabul edilmektedir. İstenen yanıt $d(n)$ ile gerçek süzgeç çıkışı $y(n)$ arasındaki fark,

$$e(n) = d(n) - y(n) \quad (2.2)$$

kestirim hatası olarak adlandırılır.

Kestirim problemi, istenen yanıt $d(n)$ ile gerçek süzgeç çıkışı $y(n)$ arasındaki fark istatistiksel anlamda mümkün olduğunca küçük olacak biçimde süzgecin tasarlanmasıyla çözülmektedir.

Wiener süzgeç teorisinde süzgecin optimizasyonu için ‘ en küçük karesel ortalama hata kriteri’ kullanılmaktadır. Yani süzgeç tasarımı optimize etmek için, kestirim hatası $e(n)$ ’nin karesel ortalama değeri,

$$\begin{aligned} J &= E[e(n)e^*(n)] \\ &= E[|e(n)|^2] \end{aligned} \quad (2.3)$$

minimum olacak şekilde seçilir. Burada E beklendik değer işlemcisini gösterir.

Karmaşık süzgeç girişi için, süzgeç katsayıları da karmaşıktır. k . süzgeç katsayısı w_k , gerçel ve sanal olmak üzere

$$w_k = a_k + jb_k, \quad k = 0, 1, \dots \quad (2.4)$$

biçiminde iki kısımdan oluşur.

Gradyen işlemcisi ∇ ’nın k . süzgeç katsayısı için değeri

$$\nabla_k = \frac{\partial}{\partial a_k} + j \frac{\partial}{\partial b_k}, \quad k = 0, 1, \dots \quad (2.5)$$

biçimindedir. J ye ∇ gradyen işlemcisinin uygulanması ile çok boyutlu karmaşık gradyen vektör $\nabla(J)$ ’nin k . elemanı,

$$\nabla_k(J) = \frac{\partial J}{\partial a_k} + j \frac{\partial J}{\partial b_k}, \quad k = 0, 1, \dots \quad (2.6)$$

biçiminde yazılabilir.

J nin minimum değerini elde etmek için, gradyen vektör $\nabla(J)$ ' nin bütün elemanları sıfıra eşitlenerek,

$$\nabla_k(J) = 0, \quad k = 0, 1, \dots \quad (2.7)$$

elde edilir. Bu koşulun sağlanması durumunda, süzgeç karesel ortalama hata anlamında optimumdur denir.

Buradan (2.3) eşitliğinin (2.6) eşitliğinde yerine konulmasıyla,

$$\nabla_k(J) = E \left[e^*(n) \frac{\partial e(n)}{\partial a_k} + e(n) \frac{\partial e^*(n)}{\partial a_k} + je^*(n) \frac{\partial e(n)}{\partial b_k} + je(n) \frac{\partial e^*(n)}{\partial b_k} \right] \quad (2.8)$$

eşitliği elde edilir.

(2.2) ve (2.1) eşitlikleri kullanılarak,

$$\begin{aligned} \frac{\partial e(n)}{\partial a_k} &= -u(n-k) \\ \frac{\partial e(n)}{\partial b_k} &= ju(n-k) \\ \frac{\partial e^*(n)}{\partial a_k} &= -u^*(n-k) \\ \frac{\partial e^*(n)}{\partial b_k} &= -ju^*(n-k) \end{aligned} \quad (2.9)$$

elde edilir. (2.9) eşitlikleri (2.8)' de yerlerine yazılırsa,

$$\nabla_k(J) = -2E[u(n-k)e^*(n)] \quad (2.10)$$

sonucu elde edilir.

Şimdi J nin minimize edilmesi için uygulanması gereken durumları belirtebiliriz. Süzgeç, optimum çalışma durumundayken kestirim hatasını e_o ile gösterelim. (2.7) eşitliğinde belirtilen durum bulunduğundan sonra ,

$$E[u(n-k)e_o^*(n)] = 0, \quad k = 0, 1, \dots \quad (2.11)$$

eşitliği yazılabilir. (2.11) eşitliğinden şu sonucu çıkarabiliriz: J nin minimum değerinin elde edilmesi için gerek ve yeter koşul, bu değere karşı düşen kestirim hatası $e_o(n)$ ' nin, giriş işaretinin her örnek değerine dik olmasıdır. Bu durum, optimum süzgeçleme probleminin temel kurallarından olan diklik kuralını oluşturur.

Kestirim hatası $e(n)$ ve süzgeç çıkışı $y(n)$ arasındaki ilişkinin incelenmesi ile diklik kuralının bir sonucu türetilir. (2.1) eşitliği kullanılarak,

$$\begin{aligned} E[y(n)e^*(n)] &= E\left[\sum_{k=0}^{\infty} w_k * u(n-k)e^*(n)\right] \\ &= \sum_{k=0}^{\infty} w_k * E[u(n-k)e^*(n)] \end{aligned} \quad (2.12)$$

yazılabilir.

Karesel ortalama hata anlamında optimize edilmiş süzgecin ürettiği çıkış y_o ile bu durumdaki kestirim hatası ise e_o ile gösterilsin. (2.11) eşitliği ile tanımlanan diklik kuralı kullanılarak,

$$E[y_o(n)e_o^*(n)] = 0 \quad (2.13)$$

sonucu elde edilir. Böylece, süzgeç optimum durumunda çalışırken, süzgeç çıkışında üretilen istenen yanıtın kestirimi $y_o(n)$ ve bu durumdaki kestirim hatası $e_o(n)$ ' nin birbirine dik olduğu sonucuna varılır.

n anını kadar (n dahil) olan giriş verileri U_n için, karesel ortalama hata anlamında optimize edilmiş istenen yanıt kestirimini,

$$\hat{d}(n/U_n) = y_o(n) \quad (2.14)$$

ile gösterelim. Burada, $\hat{d}(n/U_n)$ kestirimi sıfır ortalamalıdır. Çünkü tap girişleri sıfır ortalamalı olarak kabul edilir. Bu durum istenen yanıt $d(n)$ ' nin sıfır ortalamalı kabul edilmesi ile uyumludur.

2.3. Minimum Karesel Ortalama Hata

Şekil 2.1' deki doğrusal ayrık zamanlı süzgeç, optimum durumda çalıştığında eşitlik (2.2),

$$\begin{aligned} e_o(n) &= d(n) - y_o(n) \\ &= d(n) - \hat{d}(n/U_n) \end{aligned} \quad (2.15)$$

özel formuna dönüşür. Bu eşitlik yeniden düzenlenirse,

$$d(n) = \hat{d}(n/U_n) + e_o(n) \quad (2.16)$$

eşitliği elde edilir.

J_{min} minimum karesel ortalama hatayı gösterir ve

$$J_{min} = E[|e_o(n)|^2] \quad (2.17)$$

ile tanımlanır. Burada (2.16) eşitliğinin her iki yanının karesel ortalama değerleri alınır ve (2.13) eşitliği ile tanımlanan diklik kuralının sonucu uygulanırsa,

$$\sigma_d^2 = \sigma_{\hat{d}}^2 + J_{min} \quad (2.18)$$

elde edilir. Burada σ_d^2 istenen yanıtın değışintisi, $\sigma_{\hat{d}}^2$ ise $\hat{d}(n/U_n)$ kestiriminin değışintisidir. Bunların her ikisi de sıfır ortalamalı olarak kabul edilmiştir. (2.18) eşitliğinden minimum karesel ortalama hatanın çözümü,

$$J_{min} = \sigma_d^2 - \sigma_{\hat{d}}^2 \quad (2.19)$$

şeklinde bulunur. Bu ilişkiden optimum süzgeçleme için minimum karesel ortalama hata değerin, istenen hatanın değışintisi ile süzgeç çıkışında üretilen kestirimin değışintisi arasındaki farka eşit olduğu görülür.

(2.19) eşitliğinin her iki yanı σ_d^2 ’ye bölünürse,

$$\frac{J_{min}}{\sigma_d^2} = 1 - \frac{\sigma_{\hat{d}}^2}{\sigma_d^2} \quad (2.20)$$

eşitliği elde edilir. Görülüyor ki, bu eşitlik istenen yanıt $d(n)$ ’nin her n değeri için sıfır olduğu özel durum dışında, σ_d^2 değeri asla sıfır olamayacağından

$$\varepsilon = \frac{J_{min}}{\sigma_d^2} \quad (2.21)$$

olduğu kabul edilir ve ε niceliği normalize karesel ortalama hata olarak adlandırılır. (2.20) eşitliğinden,

$$\varepsilon = 1 - \frac{\sigma_{\hat{d}}^2}{\sigma_d^2} \quad (2.22)$$

elde edilir. ε asla negatif olamaz ve $\frac{\sigma_{\hat{d}}^2}{\sigma_d^2}$ daima pozitifdir. Böylece,

$$0 \leq \varepsilon \leq 1 \quad (2.23)$$

yazılabilir. Eğer ε sıfıra eşitse, optimum süzgeç çok iyi çalışıyor demektir. Öyle ki $\hat{d}(n/U_n)$ ile $d(n)$ arasında tam bir uyuşma vardır. Eğer ε bire eşitse, bu en kötü durumdur yani $\hat{d}(n/U_n)$ ile $d(n)$ arasında hiç bir uyum söz konusu değildir.

2.4. Wiener-Hopf Eşitlikleri

(2.11) eşitliği ile tanımlanan diklik kuralı süzgecin optimum çalışması için gerek ve yeter koşulu belirtir. (2.1) ve (2.2) eşitlikleri (2.11) eşitliğinde yerlerine konulursa optimallik için gerek ve yeter koşul

$$E \left[u(n-k) \left(d^*(n) - \sum_{i=0}^{\infty} w_{oi} u^*(n-i) \right) \right] = 0, \quad k = 0, 1, \dots$$

biçiminde yeniden formüle edilebilir. Burada, w_{oi} optimum süzgecin impuls yanıtının i . katsayısını gösterir. Bu eşitliğin genişletilmesi ve terimlerin yeniden düzenlenmesi ile,

$$\sum_{i=0}^{\infty} w_{oi} E[u(n-k)u^*(n-i)] = E[u(n-k)d^*(n)], \quad k = 0, 1, \dots \quad (2.24)$$

elde edilir. (2.24) eşitliğindeki iki beklendik değer aşağıdaki şekilde yorumlanabilir:

1. $E[u(n-k)u^*(n-i)]$ beklendik değeri, $(i - k)$ gecikmesi için süzgeç girişinin özilişki fonksiyonuna eşittir:

$$r(i-k) = E[u(n-k)u^*(n-i)] \quad (2.25)$$

2. $E[u(n-k)d^*(n)]$ beklendik değeri, $(-k)$ gecikmesi için istenen yanıt $d(n)$ ile süzgeç girişi $u(n-k)$ arasındaki çapraz ilişki fonksiyonuna eşittir:

$$p(-k) = E[u(n-k)d^*(n)] \quad (2.26)$$

(2.25) ve (2.26) eşitlikleri (2.24) eşitliğinde yerlerine konulursa,

$$\sum_{i=0}^{\infty} w_{oi} r(i-k) = p(-k), \quad k = 0, 1, \dots \quad (2.27)$$

biçiminde, süzgecin optimallığı için gerek ve yeter koşul olan eşitlik elde edilir.

Bu eşitlik, optimum süzgeç katsayılarının iki ilişki fonksiyonundan yararlanarak elde edilmesini gösterir. Bu ilişki fonksiyonları süzgeç girişinin özilişki fonksiyonu ve istenen yanıt ile süzgeç girişi arasındaki çapraz ilişki fonksiyonudur. Bu eşitlik Wiener-Hopf eşitlikleri olarak adlandırılır.

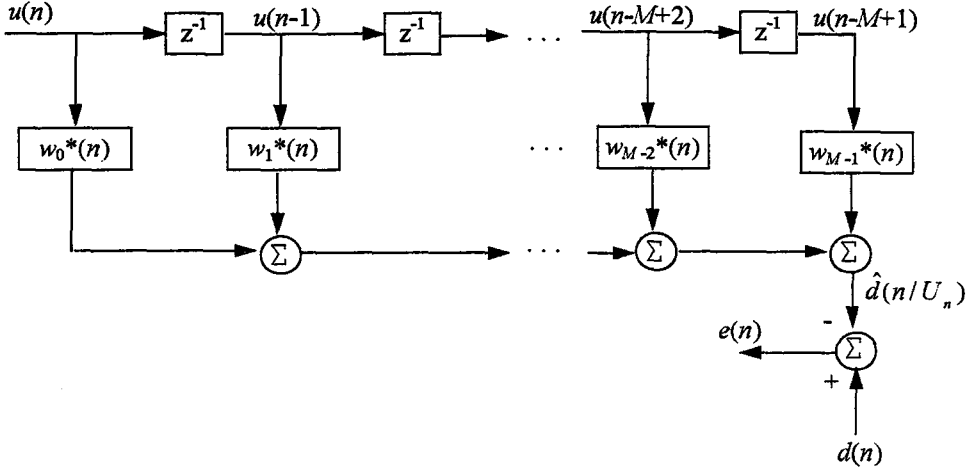
2.4.1. Doğrusal Enine Süzgeçler İçin Wiener-Hopf Eşitliklerinin Çözümü

Wiener-Hopf eşitliklerinin çözümü, süzgeç doğrusal enine (transversal) süzgeç (sonlu süreli impuls yanıtı -FIR (Finite Impuls Response)-) biçiminde olduğu özel durumlar için oldukça basitleşir. Enine bir süzgeç yapısı Şekil 2.2' de verilmektedir.

Şekildeki enine süzgecin impuls yanıtı, sınırlı sayıdaki tap ağırlıkları $w_0, w_1, \dots, w_{M-1}$ ile tanımlıdır. Buna göre, (2.27) Wiener-Hopf eşitlikleri,

$$\sum_{i=0}^{M-1} w_{oi} r(i-k) = p(-k), \quad k = 0, 1, \dots, M-1 \quad (2.28)$$

biçimine dönüşür. Burada, $w_{o0}, w_{o1}, \dots, w_{o,M-1}$, süzgeç tap ağırlıklarının optimum değerleridir.



Şekil 2.2 Enine Süzgeç

2.4.2. Wiener-Hopf Eşitliklerinin Matris Formu

$\mathbf{R}$, Şekil 2.2' deki enine süzgeçteki $u(n)$, $u(n-1)$, ..., $u(n-M+1)$ tap girişlerinin $M \times M$ boyutlu ilişki matrisini gösterir:

$$\mathbf{R} = E[\mathbf{u}(n)\mathbf{u}^H(n)] \quad (2.29)$$

Burada tap girişleri

$$\mathbf{u}(n) = [u(n), u(n-1), \dots, u(n-M+1)]^T \quad (2.30)$$

biçiminde $M \times 1$ boyutlu tap giriş vektörü şeklinde gösterilebilir.

Genişletilmiş formda ilişki matrisi,

$$\mathbf{R} = \begin{bmatrix} r(0) & r(1) & \dots & r(M-1) \\ r^*(1) & r(0) & \dots & r(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r^*(M-1) & r^*(M-2) & \dots & r(0) \end{bmatrix} \quad (2.31)$$

şeklinde yazılabilir.

Buna uygun olarak $\mathbf{p}$, istenen yanıt $d(n)$ ile süzgeç tap girişleri arasındaki $M \times 1$ boyutlu çapraz ilişki vektörünü gösterir:

$$\mathbf{p} = E[u(n)d^*(n)] \quad (2.32)$$

Genişletilmiş formda ise

$$\mathbf{p} = [p(0), p(-1), \dots, p(1-M)]^T \quad (2.33)$$

biçiminde yazılabilir.

(2.28) ile verilen Wiener-Hopf eşitliği kapalı matris formunda,

$$\mathbf{R}\mathbf{w}_o = \mathbf{p} \quad (2.34)$$

biçiminde yazılabilir. Burada,

$$\mathbf{w}_o = [\mathbf{w}_{o0}, \mathbf{w}_{o1}, \dots, \mathbf{w}_{o,M-1}]^T \quad (2.35)$$

olmak üzere enine süzgecin $M \times 1$ boyutlu optimum tap ağırlık vektörünü göstermektedir.

Wiener-Hopf eşitliğinin çözümü için, ilişki matrisi $\mathbf{R}$ 'nin tekil olmadığını kabul ederiz. Böylece (2.34) eşitliğinin her iki tarafını ilişki matrisinin tersi $\mathbf{R}^{-1}$ ile öncül çarparsak,

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p} \quad (2.36)$$

elde edilir.

Optimum tap ağırlık vektörü w_o ' nin hesaplanması iki nicelik gerektirir: (1) tap giriş vektörü $u(n)$ ' nin ilişki matrisi R ve (2) tap giriş vektörü $u(n)$ ile istenen yanıt $d(n)$ arasındaki çapraz ilişki vektörü p .

2.5. Hata Başarım Yüzeyi

Wiener-Hopf eşitlikleri Kısım 2.2' de verilmiş olan diklik kuralı ile izlenebilir. Wiener-Hopf eşitliklerini, J nin Şekil 2.2' deki enine süzgecin tap ağırlıklarına göre bağımlılığını inceleyerek de türetebiliriz. İlk olarak kestirim hatasını,

$$e(n) = d(n) - \sum_{k=0}^{M-1} w_k * u(n-k) \quad (2.37)$$

biçiminde yazalım. Burada $d(n)$ istenen yanıt, $w_0, w_1, \dots, w_{M-1}$ süzgecin tap ağırlıkları ve $u(n), u(n-1), \dots, u(n-M+1)$ tap girişleridir. Şekil 2.2' deki enine süzgeç yapısı için,

$$\begin{aligned} J &= E[e(n)e^*(n)] \\ &= E[|d(n)|^2] - \sum_{k=0}^{M-1} w_k * E[u(n-k)d^*(n)] - \sum_{k=0}^{M-1} w_k E[u^*(n-k)d(n)] \\ &\quad + \sum_{k=0}^{M-1} \sum_{i=0}^{M-1} w_k * w_i E[u(n-k)u^*(n-i)] \end{aligned} \quad (2.38)$$

tanımlanabilir.

Şimdi (2.38) eşitliğindeki dört beklendik değeri tanıyalım:

1. İlk beklendik değer için,

$$\sigma_d^2 = E[|d(n)|^2] \quad (2.39)$$

yazılabilir. Burada σ_d^2 , sıfır ortalamalı olarak kabul edilen istenen yanıt $d(n)$ ' nin değişintisidir.

2. İkinci ve üçüncü beklendik değerler için sırasıyla,

$$p(-k) = E[u(n-k)d^*(n)] \quad (2.40)$$

ve

$$p^*(-k) = E[u^*(n-k)d(n)] \quad (2.41)$$

yazılabilir. Burada $p(-k)$ istenen yanıt $d(n)$ ile tap girişi $u(n-k)$ arasındaki çapraz ilişkiyi göstermektedir.

3. Son olarak, dördüncü beklendik değer için,

$$r(i-k) = E[u(n-k)u^*(n-i)] \quad (2.42)$$

yazılabilir ve burada $r(i-k)$, $i-k$ gecikmesi için tap girişlerinin özilişki fonksiyonudur.

Böylece (2.38) eşitliği yeniden yazılırsa,

$$J = \sigma_d^2 - \sum_{k=0}^{M-1} w_k * p(-k) - \sum_{k=0}^{M-1} w_k p^*(-k) + \sum_{k=0}^{M-1} \sum_{i=0}^{M-1} w_k * w_i r(i-k) \quad (2.43)$$

elde edilir.

(2.43) eşitliği, enine süzgecin tap girişleri ile istenen yanıtın birleşik durağan olduğu durumu belirtir. Karesel ortalama hata J , süzgeç tap ağırlıklarının ikinci dereceden bir fonksiyonudur. Bu nedenle, J ' nin $w_0, w_1, \dots, w_{M-1}$ tap ağırlıklarına bağımlılığını $M+1$ boyutlu kase şeklinde bir yüzey olarak düşünebiliriz. Bu yüzey tek bir minimum ile karakterize edilir. Şekil 2.2' deki enine süzgecin hata başarıım yüzeyi bu şekilde düşünülür. Hata başarıım yüzeyinin dip ya da minimum noktasında karesel ortalama hata

J_{min} ile gösterilen minimum değerine ulaşır. Bu noktada gradyen vektör $\nabla(J)$ sifra eşittir. Başka bir deyişle,

$$\nabla_k(J) = 0, \quad k = 0, 1, \dots, M-1 \quad (2.44)$$

yazılabilir. Burada $\nabla_k(J)$, gradyen vektörün k . elemanını gösterir. k . tap ağırlığı,

$$w_k = a_k + jb_k$$

biçimindedir. Böylece, (2.43) eşitliği kullanılarak $\nabla_k(J)$ ifadesi,

$$\begin{aligned} \nabla_k(J) &= \frac{\partial J}{\partial a_k} + j \frac{\partial J}{\partial b_k} \\ &= -2p(-k) + 2 \sum_{i=0}^{M-1} w_i r(i-k) \end{aligned} \quad (2.45)$$

biçiminde yazılabilir. (2.44) eşitliği, (2.45) eşitliğine uygulanırsa, Şekil 2.2' deki enine süzgeç için,

$$\sum_{i=0}^{M-1} w_{oi} r(i-k) = p(-k), \quad k = 0, 1, \dots, M-1$$

elde edilir. Bu eşitlik sistemi ile optimum tap ağırlıkları $w_{o0}, w_{o1}, \dots, w_{o,M-1}$ bulunabilir. Ayrıca, bu eşitlik sistemi, Kısım 2.4' de türetilen Wiener-Hopf eşitlikleri (2.28) ile özdeşir.

2.5.1. Minimum Karesel Ortalama Hata

İstenen yanıt $d(n)$ ' nin kestirimi olarak tanımlanan $\hat{d}(n/U_n)$, karesel ortalama hata anlamında optimize edilmiş biçimde Şekil 2.2' deki enine süzgecin çıkışında üretilir. Şekil 2.2' den anlaşıldığı gibi,

$$\begin{aligned}\hat{d}(n/U_n) &= \sum_{k=0}^{M-1} w_{ok} * u(n-k) \\ &= \mathbf{w}_o^H \mathbf{u}(n)\end{aligned}\quad (2.46)$$

olur. Burada $\mathbf{w}_o$, elemanları $w_{o0}, w_{o1}, \dots, w_{o,M-1}$ olan optimum tap ağırlık vektörüdür ve $\mathbf{u}(n)$, (2.30) eşitliği ile tanımlanan tap giriş vektörüdür. $\mathbf{w}_o^H \mathbf{u}(n)$; optimum tap ağırlık vektörü $\mathbf{w}_o$ ve tap giriş vektörü $\mathbf{u}(n)$ ' nin iç çarpımını gösterir. $\mathbf{u}(n)$ ve $\hat{d}(n/U_n)$ ' nin sıfır ortalamalı olduğu kabul edilmektedir. Bundan sonra, $\hat{d}(n/U_n)$ ' nin değışintisini hesaplamak için (2.46) eşitliği kullanılabilir ve

$$\begin{aligned}\sigma_d^2 &= E[\mathbf{w}_o^H \mathbf{u}(n) \mathbf{u}^H(n) \mathbf{w}_o] \\ &= \mathbf{w}_o^H E[\mathbf{u}(n) \mathbf{u}^H(n)] \mathbf{w}_o \\ &= \mathbf{w}_o^H \mathbf{R} \mathbf{w}_o\end{aligned}\quad (2.47)$$

elde edilir. Burada $\mathbf{R}$, tap ağırlık vektörü $\mathbf{u}(n)$ ' nin ilişki matrisidir. (2.34) eşitliği kullanılarak σ_d^2 değışintisinin optimum tap ağırlık vektörü $\mathbf{w}_o$ ' a bağımlılığı ortadan kaldırılabilir. Böylece (2.47) eşitliği,

$$\begin{aligned}\sigma_d^2 &= \mathbf{w}_o^H \mathbf{p} \\ &= \mathbf{p}^H \mathbf{w}_o\end{aligned}\quad (2.48)$$

biçiminde yeniden yazılabilir.

Enine süzgeç ile üretilen minimum karesel ortalama hatanın hesabı için (2.48) ve (2.19) eşitlikleri kullanılarak,

$$\begin{aligned}J_{\min} &= \sigma_d^2 - \mathbf{p}^H \mathbf{w}_o \\ &= \sigma_d^2 - \mathbf{p}^H \mathbf{R}^{-1} \mathbf{p}\end{aligned}\quad (2.49)$$

biçiminde istenen sonuç elde edilir.

2.5.2. Hata Başarım Yüzeyinin Kanonik Biçimi

(2.43) eşitliği enine süzgeç ile üretilen karesel ortalama hata J nin açık formunu belirtir. Bu eşitlik (2.31) ve (2.33) eşitliklerinde verilen ilişki matrisi $\mathbf{R}$ ve çapraz ilişki vektörü $\mathbf{p}$ kullanılarak,

$$J = \sigma_d^2 - \mathbf{w}^H \mathbf{p} - \mathbf{p}^H \mathbf{w} + \mathbf{w}^H \mathbf{R} \mathbf{w} \quad (2.50)$$

biçiminde yeniden yazılabilir. (2.49) eşitliğinin ilk satırını (2.50)'de yerine yazar ve (2.34) eşitliğinin optimum çözümünü kullanırsak, bu işlemlerin sonucunda eşitlikte $\mathbf{p}$ çapraz ilişki vektörünü ortadan kaldırmış oluruz. Böylece karesel ortalama hatanın ikinci dereceden bir formu olan,

$$J = J_{\min} + (\mathbf{w} - \mathbf{w}_o)^H \mathbf{R} (\mathbf{w} - \mathbf{w}_o) \quad (2.51)$$

eşitliği elde edilir. Bu eşitlik minimize edilmiş tap ağırlık vektörü $\mathbf{w}_o$ ' nin optimalliğinin tek olduğunu açıkça gösterir. (2.51) eşitliğinin sağ tarafındaki ikinci dereceden form oldukça bilgi vericidir. Bununla beraber, hata başarım yüzeyinin gösteriliminin oldukça basitleştirilmesi için tabanların değiştirilmesi gerekir. Bunu yapmak için, $\mathbf{R}$ özilişki matrisinin özelliklerinden biri olan özdeğerleri ve özvektörleri cinsinden ifade edilebilmesi özelliği kullanılacaktır:

$$\mathbf{R} = \Phi \Lambda \Phi^H \quad (2.52)$$

Burada Λ ilişki matrisinin $\lambda_1, \lambda_2, \dots, \lambda_M$ özdeğerleri ile oluşturulmuş köşegen bir matris ve $\mathbf{Q}$ bu özdeğerlere karşı düşen $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_M$ özvektörlerinden oluşmuş bir satır matrisidir:

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 & \dots & 0 \\ 0 & \lambda_2 & 0 & \dots & 0 \\ 0 & 0 & \lambda_3 & & \cdot \\ \cdot & \cdot & & \cdot & 0 \\ \cdot & \cdot & & & \cdot \\ 0 & 0 & \dots & 0 & \lambda_M \end{bmatrix}, \Phi = [\varphi_1 \quad \varphi_2 \quad \dots \quad \varphi_M]$$

(2.52) eşitliği (2.51) eşitliğine yerleştirilirse,

$$J = J_{min} + (\mathbf{w} - \mathbf{w}_o)^H \Phi \Lambda \Phi^H (\mathbf{w} - \mathbf{w}_o) \quad (2.53)$$

elde edilir. Şu şekilde bir $\mathbf{v}$ vektörü tanımlansın:

$$\mathbf{v} = \Phi^H (\mathbf{w} - \mathbf{w}_o) \quad (2.54)$$

Bu ifade (2.53) eşitliğinde yerine yazılırsa,

$$J = J_{min} + \mathbf{v}^H \Lambda \mathbf{v} \quad (2.55)$$

$$J = J_{min} + \sum_{k=1}^M \lambda_k v_k v_k^*$$

$$J = J_{min} + \sum_{k=1}^M \lambda_k |v_k|^2 \quad (2.56)$$

kareseel ortalama hatanın kanonik formu elde edilir. Burada v_k , $\mathbf{v}$ vektörünün k . elemanıdır.

Hata başarımlı yüzeyinin gösterilimini çok daha kullanışlı yapan, bu kanonik form özelliği ile $\mathbf{v}$ vektörünün elemanları hata başarımlı yüzeyinin temel eksenlerini oluşturmaktadır.

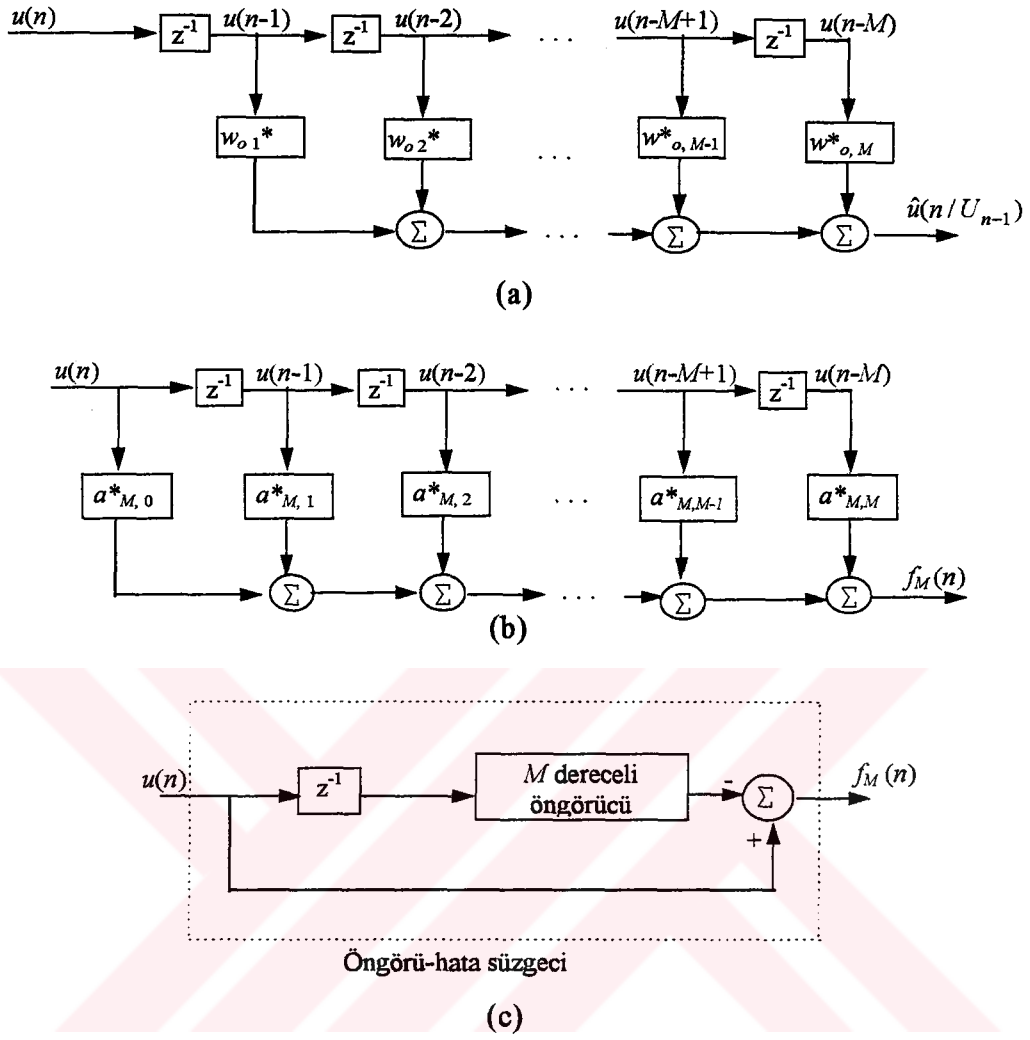
BÖLÜM 3. DOĞRUSAL ÖNGÖRÜ

Zaman serilerinin analizinde karşılaşılan önemli problemlerden biri, doğrusal ayrık zamanlı rasgele bir sürecin verilen geçmiş değerlerini göz önüne alarak gelecek değerinin öngörülmesidir. $u(n)$, $u(n-1)$, ..., $u(n-M)$ şeklinde düşünülen bir zaman serisi, o ana kadar ki $(M+1)$ tane değeri, n zamanı için temsil eder. Örneğin öngörü işlemi, $u(n)$ ' nin kestirimini yapmak için $u(n-1)$, $u(n-2)$, ..., $u(n-M)$ örneklerinin kullanımı içerebilir. $u(n-1)$, $u(n-2)$, ..., $u(n-M)$ ile oluşan M boyutlu U_{n-1} uzayı, $u(n)$ ' nin öngörülen değeri $\hat{u}(n/U_{n-1})$ ' nin hesaplanmasında kullanılır. Doğrusal öngöründe, bu öngörülen değeri $u(n-1)$, $u(n-2)$, ..., $u(n-M)$ örneklerinin doğrusal bir kombinasyonu olarak düşünebiliriz. Bu işlem, $n-1$ anına göre ölçülen “ geleceğe tek adımlı öngörü” olarak adlandırılır. Başka bir öngörü biçimindeyse $u(n)$, $u(n-1)$, ..., $u(n-M+1)$ örnekleri geçmişteki $u(n-M)$ örneğinin öngörüsünü yapmak için kullanılır. Buna da “geri yönde tek adımlı öngörü” denir.

Burada, Wiener Süzgeç kuramı kullanılarak, ileri ve geri yönde öngörücünün tasarlanmasında minimum karesel ortalama hata göz önüne alınacaktır. Wiener Süzgeç Kuramını uygulayabilmek için rasgele sürecin geniş anlamda durağan ve ortalamasının sıfır olduğu varsayılacaktır. Bu şartlar altında böyle bir sürecin ilişki matrisi Toeplitz yapıdadır.

3.1. İleri Yönde Doğrusal Öngörü

İleri yönde doğrusal öngörü işlemi, yukarıda da açıklandığı üzere, $u(n-1)$, $u(n-2)$, ..., $u(n-M)$ örnek değerleri kullanılarak $u(n)$ örneğinin öngörüsünü elde etme işlemidir. Şekil 3.1 (a), tap girişleri $u(n-1)$, $u(n-2)$, ..., $u(n-M)$ ve tap ağırlıkları w_{o1} , w_{o2} , ..., w_{oM} olan bir doğrusal enine süzgeçten oluşan bir ileri yönde öngörücüyü göstermektedir.



Şekil 3.1 (a) Tek Adımlı İleri Öngörücü
 (b) İleri Öngörü Hata Süzgeci
 (c) İleri Yönde Öngörücü ile İleri Yönde Öngörü Hata Süzgeci Arasındaki İlişki

Süzgecin tap girişleri sıfır ortalamalı, geniş anlamda durağan bir rasgele süreçtir. Bilinen örnek değerleri,

$$U_{n-1} = [u(n-1) \quad u(n-2) \quad \dots \quad u(n-M)]^T \quad (3.1)$$

olmak üzere, bu değerler yardımıyla öngörülen $u(n)$ değeri,

$$\hat{u}(n/U_n) = \sum_{k=1}^M w_{ok} * u(n-k) \quad (3.2)$$

ile tanımlanır. Burada tanımlanan durum için, istenen yanıt $d(n)$, $u(n)$ ' e eşittir:

$$d(n)=u(n) \quad (3.3)$$

İleri öngörü hatası, $f_M(n)$ ile gösterilir ve

$$f_M(n) = u(n) - \hat{u}(n/U_{n-1}) \quad (3.4)$$

şeklinde ifade edilir. Burada, M öngörücünün derecesidir ve öngörü işleminde kullanılan gecikme elemanlarının sayısını ifade eder.

P_M , minimum karesel ortalama öngörü hatasını gösterir:

$$P_M = E[|f_M(n)|^2] \quad (3.5)$$

Tap girişleri sıfır ortalamalı kabul edildiğinden, ileri öngörü hatası $f_M(n)$ de sıfır ortalamalıdır. Bu durumda, P_M değeri, ileri öngörü hatasının değışintisine eşit olacaktır.

w_o , ileri öngörücünün optimum tap ağırlıklarından oluşan,

$$\mathbf{w}_o = [w_{o1}, w_{o2}, \dots, w_{oM}]^T \quad (3.6)$$

biçiminde $M \times 1$ boyutlu bir vektördür.

Wiener-Hopf eşitliklerini w_o vektörü için çözmek için iki nicelik hakkında bilgi gerekir: (1) $u(n-1)$, $u(n-2)$, ..., $u(n-M)$ tap girişlerinin $M \times M$ boyutlu ilişki matrisi ve (2) bu tap girişleri ile istenen yanıt $u(n)$ arasındaki $M \times 1$ boyutlu çapraz ilişki vektörüdür. Ayrıca P_M ' nin belirlenmesi için de, üçüncü bir nicelik olarak $u(n)$ ' nin değışintisine gerek duyulur.

Bu üç büyüklük aşağıdaki biçimde tanımlanabilir:

1. $u(n-1), u(n-2), \dots, u(n-M)$ tap girişleri ile tanımlanan $M \times 1$ boyutlu tap giriş vektörü,

$$\mathbf{u}(n-1) = [u(n-1), u(n-2), \dots, u(n-M)]^T \quad (3.7)$$

olduğundan tap girişlerinin ilişki matrisi ,

$$\mathbf{R} = E[\mathbf{u}(n-1)\mathbf{u}^H(n-1)]$$

$$\mathbf{R} = \begin{bmatrix} r(0) & r(1) & \dots & r(M-1) \\ r(-1) & r(0) & \dots & r(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(-M+1) & r(-M+2) & \dots & r(0) \end{bmatrix} \quad (3.8)$$

biçiminde ifade edilir. Burada $r(k)$, k . gecikme için işaretin özilişki fonksiyonudur.

2. Tap girişleri $\mathbf{u}(n-1)$ ve istenen yanıt $u(n)$ arasındaki $M \times 1$ boyutlu çapraz ilişki vektörü

$$\mathbf{r} = E[\mathbf{u}(n-1)u^*(n)]$$

$$\mathbf{r} = \begin{bmatrix} r^*(1) \\ r^*(2) \\ \vdots \\ r^*(M) \end{bmatrix} = \begin{bmatrix} r(-1) \\ r(-2) \\ \vdots \\ r(-M) \end{bmatrix} \quad (3.9)$$

biçiminde ifade edilir.

3. $u(n)$ sıfır ortalamalı olduğundan $u(n)$ ' nin değışintisi $r(0)$ ' a eşittir.

Tablo 3.1 Wiener-Süzgeç Değişkenlerinin Özeti

Nicelik	Wiener-Süzgeci (Şekil 2.2)	İleri Yönde Öngörücü (Şekil 3.1(a))	Geri Yönde Öngörücü (Şekil 3.2(a))
Tap Giriş Vektörü	$u(n)$	$u(n-1)$	$u(n)$
İstenen Yanıt	$d(n)$	$u(n)$	$u(n-M)$
Tap Ağırlık Vektörü	w_o	w_o	g
Kestirim Hatası	$e(n)$	$f_M(n)$	$b_M(n)$
Tap Girişlerinin İlişki Matrisi	R	R	R
Tap Girişleri İle İstenen Yanıt Arasındaki Çapraz İlişki Vektörü	p	r	r^{B*}
Minimum Karesel Ortalama Hata	J_{min}	P_M	P_M

Bu tabloyu kullanarak daha önce elde edilen Wiener-Hopf eşitliklerini durağan girişler için ileri yönde doğrusal öngörü probleminin çözümünde kullanılmak üzere,

$$Rw_o = r \quad (3.10)$$

şeklinde değiştirebiliriz. Benzer biçimde (2.49) eşitliğini kullanarak ileri yönde öngörü hatası gücü

$$P_M = r(0) - r^H w_o \quad (3.11)$$

şeklinde hesaplanabilir.

3.1.1. İleri Yönde Öngörü Hata Süzgeci

(3.2) eşitliğinin (3.4) eşitliğinde yerine konulmasıyla ileri yönde öngörü hatası,

$$f_M(n) = u(n) - \sum_{k=1}^M w_{ok} * u(n-k) \quad (3.12)$$

biçiminde ifade edilebilir.

$a_{M,k}$, $k=0, 1, 2, \dots, M$, yeni bir enine süzgecin tap ağırlıkları olarak

$$a_{Mk} = \begin{cases} 1 & , k = 0 \\ -w_{ok} & , k = 1, 2, \dots, M \end{cases} \quad (3.13)$$

biçiminde tanımlanırsa, ileri öngörü hatası

$$f_M(n) = \sum_{k=0}^M a_{M,k} * u(n-k) \quad (3.14)$$

şeklinde ifade edilebilir. Bu giriş çıkış ilişkisi Şekil 3.1 (b)' deki gibi bir enine süzgeç ile gösterilebilir. Bu süzgeç, çıkışında ileri öngörü hatası $f_M(n)$ ' i üretmek için $u(n)$, $u(n-1)$, ..., $u(n-M)$ örnekler kümesini işler ve ileri yönde öngörü hatası süzgeci olarak adlandırılır. Ayrıca, Şekil 3.1 (c), ileri yönde öngörücü ile ileri yönde öngörücü hata süzgeci arasındaki ilişkiyi blok diyagram olarak gösterir.

3.1.2. İleri Yönde Öngörü İçin Genişletilmiş Wiener-Hopf Eşitlikleri

(3.10) ve (3.11) eşitliklerini tek bir matris ilişkisi ile,

$$\begin{bmatrix} r(0) & \mathbf{r}^H \\ \mathbf{r} & \mathbf{R} \end{bmatrix} \begin{bmatrix} 1 \\ -\mathbf{w}_o \end{bmatrix} = \begin{bmatrix} P_M \\ \mathbf{0} \end{bmatrix} \quad (3.15)$$

biçiminde gösterelim. (3.8) , (3.9) ve (3.13) eşitliklerini kullanarak bu ifade daha açık bir şekilde,

$$\begin{bmatrix} r(0) & r(1) & \dots & r(M) \\ r(-1) & r(0) & \dots & r(M-1) \\ \vdots & \vdots & \ddots & \vdots \\ r(-M) & r(-M+1) & \dots & r(0) \end{bmatrix} \begin{bmatrix} a_{M,0} \\ a_{M,1} \\ \vdots \\ a_{M,M} \end{bmatrix} = \begin{bmatrix} P_M \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (3.16)$$

şeklinde yazılabilir, veya $(M+1)$ tane eşitlik kullanarak,

$$\sum_{l=0}^M a_{M,l} r(l-i) = \begin{cases} P_M, & i = 0 \\ 0, & i = 1, 2, \dots, M \end{cases} \quad (3.17)$$

biçiminde de ifade edilebilir. Tüm bu eşitlikler M dereceli ileri yönde öngörü hatası süzgecinin genişletilmiş Wiener-Hopf eşitlikleridir.

3.2. Geri Yönde Doğrusal Öngörü

Geri yönde doğrusal öngörü işlemi, $u(n)$, $u(n-1)$, ..., $u(n-M+1)$ örnek değerleri kullanılarak $u(n-M)$ örneğinin öngörüsünü elde etme işlemidir. Şekil 3.2 (a), tap girişleri $u(n)$, $u(n-1)$, ..., $u(n-M+1)$ ve tap ağırlıkları g_1 , g_2 , ..., g_M olan bir doğrusal enine süzgeçten oluşan bir geri yönde öngörücüyü gösterir. Bilinen örnek değerleri

$$U_n = [u(n) \quad u(n-1) \quad \dots \quad u(n-M+1)]^T \quad (3.18)$$

olmak üzere, bu değerler yardımıyla öngörülen $u(n-M)$ değeri,

$$\hat{u}(n-M/U_n) = \sum_{k=1}^M g_k * u(n-k+1) \quad (3.19)$$

ile tanımlanır. Burada tanımlanan durum için istenen yanıt $d(n)$, $u(n-M)$ değerine eşittir:

$$d(n) = u(n-M) \quad (3.20)$$

Geri yönde öngörü hatası $b_M(n)$ ile gösterilir ve

$$b_M(n) = u(n-M) - \hat{u}(n-M/U_n) \quad (3.21)$$

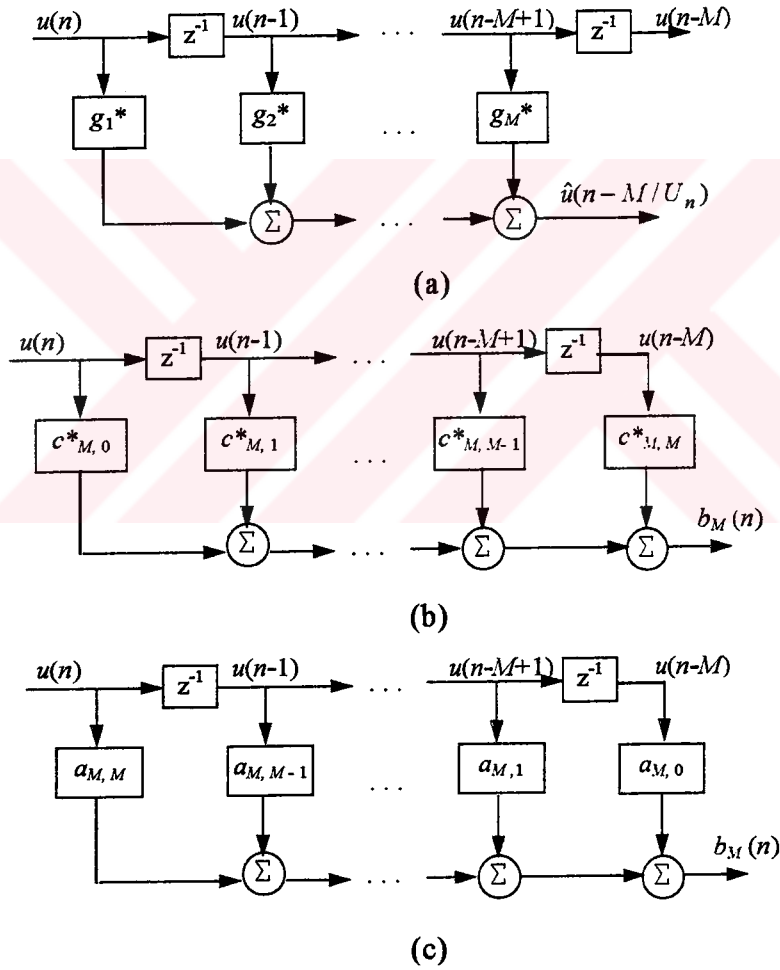
şeklinde ifade edilir. Minimum karesel ortalama öngörü hatası,

$$P_M = E[|b_M|^2] \quad (3.22)$$

ile ifade edilebilir. Geri yönde öngörücünün optimum tap ağırlıklarından oluşan $M \times 1$ boyutlu vektör ise,

$$\mathbf{g} = [g_1, g_2, \dots, g_M]^T \quad (3.23)$$

biçiminde gösterilebilir.



Şekil 3.2 (a) Tek Adımlı Geri Öngörücü
 (b) Geri Öngörü Hata Süzgeci
 (c) İleri Öngörü Hata Süzgecinin Tap Ağırlıkları
 İle Tanımlanmış Geri Öngörü Hata Süzgeci

Wiener-Hopf eşitliklerinin $\mathbf{g}$ vektörü için çözülmesi işleminde iki nicelik hakkında bilgi gerekir: (1) $u(n)$, $u(n-1)$, ..., $u(n-M+1)$ tap girişlerinin $M \times M$ boyutlu ilişki matrisi ve (2) bu tap girişleri ile istenen yanıt $u(n-M)$ arasındaki $M \times 1$ boyutlu çapraz ilişki vektörü. Ayrıca P_M ' nin belirlenmesi için de, üçüncü bir nicelik olarak $u(n-M)$ ' nin değişintisine ihtiyaç vardır.

Bu üç büyüklük aşağıdaki biçimde tanımlanabilir:

1. $u(n)$, $u(n-1)$, ..., $u(n-M+1)$ tap girişleri ile tanımlanan $M \times 1$ boyutlu tap giriş vektörü ,

$$\mathbf{u}(n) = [u(n) \quad u(n-1) \quad \dots \quad u(n-M+1)]^T \quad (3.24)$$

biçiminde olduğundan, tap girişlerinin ilişki matrisi

$$\mathbf{R} = E[\mathbf{u}(n)\mathbf{u}^H(n)] \quad (3.25)$$

$$\mathbf{R} = \begin{bmatrix} r(0) & r(1) & \dots & r(M-1) \\ r(-1) & r(0) & \dots & r(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(-M+1) & r(-M+2) & \dots & r(0) \end{bmatrix}$$

şeklinde gösterilebilir.

2. Tap girişleri $\mathbf{u}(n)$ ile istenen yanıt $u(n-M)$ arasındaki $M \times 1$ boyutlu çapraz ilişki vektörü,

$$\mathbf{r}^{B*} = E[\mathbf{u}(n)\mathbf{u}^*(n-M)] \quad (3.26)$$

$$\mathbf{r}^{B*} = \begin{bmatrix} r(M) \\ r(M-1) \\ \vdots \\ r(1) \end{bmatrix}$$

biçiminde ifade edilir.

3. $u(n-M)$ ' nin değışintisi $r(0)$ ' a eşittir.

Tablo 3.1 kullanılarak, geri yönde doğrusal öngörü probleminin çözümü için Wiener-Hopf eşitliği,

$$\mathbf{R}\mathbf{g} = \mathbf{r}^{B*} \quad (3.27)$$

biçiminde değıştirilebilir. Benzer şekilde geri yönde öngörü hatası gücü

$$P_M = r(0) - \mathbf{r}^{BT}\mathbf{g} \quad (3.28)$$

şeklinde hesaplanabilir.

3.2.1. Geri Yönde ve İleri Yönde Öngörücüler Arasındaki İlişkiler

Sırasıyla ileri yönde ve geri yönde öngörücüye ait matris biçiminde yazılan (3.10) ve (3.27) ifadeleri karşılaştırıldığında, eşitliklerin sağında kalan vektörlerin iki açıdan birbirlerinden farklı oldukları görülür: (1) Geri yönde öngöründe çıkarılan vektör elemanları geriye doğru dizilmiştir. (2) Elemanları birbirine göre kompleks eşleniktir. Birinci farklılığı gidermek için (3.27) eşitliğinin sağdaki vektörün elemanlarının sırasıyla ters çevrilmesi gereklidir. Bu işlem, eşitliğin sol tarafının $\mathbf{R}^T\mathbf{g}^B$ şeklinde değıştirilmesiyle

sağlanır. $\mathbf{R}^T$, $\mathbf{R}$ ilişki matrisinin transpozu ve $\mathbf{g}^B$ de tap vektörü $\mathbf{g}'$ nin geri yönde düzenlenmiş halidir. Yani,

$$\mathbf{R}^T \mathbf{g}^B = \mathbf{r}^* \quad (3.29)$$

biçiminde yazılabilir. Eşitliğin her iki tarafının kompleks eşleniği alınırsa,

$$\mathbf{R}^H \mathbf{g}^{B*} = \mathbf{r}$$

ifadesi elde edilir. Burada $\mathbf{R}$, Hermitian matris olduğundan $\mathbf{R}^H = \mathbf{R}$ dir. Bu ifade kullanılarak geri yönde öngörünün Wiener-Hopf eşitliği,

$$\mathbf{R} \mathbf{g}^{B*} = \mathbf{r} \quad (3.30)$$

haline gelir. (3.30) eşitliği ile (3.10) eşitliği karşılaştırılarak, geri ve ileri yönde tap vektörleri arasında

$$\mathbf{g}^{B*} = \mathbf{w}_o \quad (3.31)$$

şeklinde bir ilişki bulunabilir. Bu denklem aracılığıyla, ileri ve geri yönde öngörücüler birbirlerine dönüştürülebilir.

Görüldüğü gibi, geri yönde öngörücüyü ileri yönde öngörücüye çevirmek için tap ağırlıkları ters çevrilir ve kompleks eşlenikleri alınır. Ayrıca,

$$\mathbf{r}^{BT} \mathbf{g} = \mathbf{r}^T \mathbf{g}^B$$

eşitliği göz önüne alınarak, (3.28) eşitliğindeki geri yönde öngörü hatası gücü

$$P_M = r(0) - \mathbf{r}^T \mathbf{g}^B \quad (3.32)$$

şeklinde yazılabilir. Bu eşitliğin her iki yanının kompleks eşleniği alındığında,

$$P_M = r(0) - \mathbf{r}^H \mathbf{g}^B * \quad (3.33)$$

elde edilir. Bulunan bu eşitliğin (3.11)' deki ileri yönde öngörü hatası gücüne eşit olduğu görülecektir. Bu nedenle, birbirine eşit olan ileri ve geri yönde öngörü hatalarının her ikisinin gücü de P_M ile gösterilir.

3.2.2. Geri Yönde Öngörü Hata Süzgeci

(3.19) eşitliğinin, (3.21) eşitliğinde yerine konulmasıyla geri yönde öngörü hatası

$$b_M(n) = u(n-M) - \sum_{k=1}^M g_k * u(n-k+1) \quad (3.34)$$

biçiminde ifade edilebilir.

$c_{M,k}$, $k=0, 1, \dots, M$, yeni bir enine süzgecin tap ağırlıkları olarak,

$$c_{M,k} = \begin{cases} -g_{k+1} & , k = 0, 1, \dots, M-1 \\ 1 & , k = M \end{cases} \quad (3.35)$$

tanımlanır. Geri yönde öngörü hatası

$$b_M(n) = \sum_{k=0}^M c_{M,k} * u(n-k) \quad (3.36)$$

şeklinde ifade edilebilir. Bu giriş çıkış ilişkisi Şekil 3.2 (b)' deki gibi bir enine süzgeç ile gösterilebilir. Bu süzgeç, geri yönde öngörü hata süzgeci olarak adlandırılır.

(3.31) eşitliğinde ileri ve geri yönde öngörücü tap vektörleri arasındaki ilişki verilmişti. Bu eşitliğin skaler biçimi göz önüne alınarak

$$g_{M-k+1}^* = w_{ok}, \quad k = 1, 2, \dots, M$$

ya da benzer ifadeyle

$$g_k = w_{o,M-k+1}^*, \quad k = 1, 2, \dots, M \quad (3.37)$$

oluşturulabilir. (3.37) eşitliği (3.35) eşitliğinde yerine yazılırsa,

$$c_{M,k} = \begin{cases} -w_{o,M-k}^*, & k = 0, 1, \dots, M-1 \\ 1 & , k = M \end{cases} \quad (3.38)$$

ifadesi bulunur. Böylece, geri yönde öngörü hata süzgeci tap ağırlıkları, ileri yönde öngörü hata süzgeci tap ağırlıkları cinsinden,

$$c_{M,k} = a_{M,M-k}^* \quad (3.39)$$

biçiminde yazılabilir. Benzer şekilde ileri yönde öngörü hata süzgeci tap ağırlıkları da geri yönde öngörü hata süzgeci tap ağırlıkları cinsinden yazılabilir. Böylece ileri ve geri öngörü işlemleri arasında geçiş sağlandığı gibi, hata süzgeçleri arasında da geçiş sağlanır.

(3.36) eşitliği, (3.39) eşitliği kullanılarak yeniden yazılırsa, geri yönde öngörü hatası

$$b_M(n) = \sum_{k=0}^M a_{M,M-k} u(n-k) \quad (3.40)$$

şeklinde ifade edilebilir.

Şekil 3.2 (c), ileri yönde öngörü hata süzgeci tap ağırlıkları kullanılarak oluşturulmuş geri yönde öngörü hata süzgecini göstermektedir.

3.2.3. Geri Yönde Öngörü İçin Genişletilmiş Wiener-Hopf Eşitlikleri

(3.27) ve (3.28) eşitliklerini tek bir matris ilişkisi ile

$$\begin{bmatrix} \mathbf{R} & \mathbf{r}^B * \\ \mathbf{r}^{BT} & r(0) \end{bmatrix} \begin{bmatrix} -\mathbf{g} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ P_M \end{bmatrix} \quad (3.41)$$

biçiminde gösterelim.

(3.25) , (3.26) ve (3.39) eşitlikleri kullanılarak bu ifade daha açık bir şekilde

$$\begin{bmatrix} r(0) & r(1) & \dots & r(M) \\ r(-1) & r(0) & \dots & r(M-1) \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ r(-M) & r(-M+1) & \dots & r(0) \end{bmatrix} \begin{bmatrix} a_{M,M}^* \\ a_{M,M-1}^* \\ \cdot \\ \cdot \\ a_{M,0}^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ P_M \end{bmatrix} \quad (3.42)$$

yazılabilir veya $(M+1)$ tane eşitlik kullanılarak

$$\sum_{l=0}^M a_{M,M-l}^* r(l-i) = \begin{cases} 0, & i = 0, 1, \dots, M-1 \\ P_M, & i = M \end{cases} \quad (3.43)$$

biçiminde de ifade edilebilir.

Tüm bu eşitlikler, M dereceli geri yönde öngörü hatası süzgecinin genişletilmiş Wiener-Hopf eşitlikleridir.

3.3. Levinson-Durbin Algoritması

Öngörü hata süzgeci tap ağırlıklarının ve öngörü hata gücünün hesaplanması için, genişletilmiş Wiener-Hopf eşitliğinin çözümüyle doğrudan bir yöntem tanımlanabilir. Bu yöntem, özyineli yapıdadır ve süzgeç tap girişlerinin ilişki matrisinin Toeplitz yapıda olması özelliğini kullanır. Levinson-Durbin algoritması olarak bilinen bu yöntem, öncelikle Levinson (1947) tarafından tanıtılmış ve Durbin (1960) tarafından tekrar formüle edilmiştir.

m . dereceden ileri yönde öngörü hata süzgeci tap ağırlıklarını, $(m+1) \times 1$ boyutlu bir $\mathbf{a}_m$ vektörü ile tanımlayalım. m . dereceden geri yönde hata süzgeci tap ağırlıklarını da, $\mathbf{a}_m$ vektörünün değerlerinin geri düzenlenip kompleks eşlenikleri alınmış olarak, $\mathbf{a}_m^{B*}$ vektörü şeklinde gösterebiliriz. $(m-1)$. dereceden ileri ve geri yönde öngörü hata süzgeçlerinin $m \times 1$ boyutlu tap ağırlık vektörleri de sırasıyla, $\mathbf{a}_{m-1}$ ve $\mathbf{a}_{m-1}^{B*}$ şeklinde gösterilebilir. Levinson-Durbin özyinelemesi sonuç olarak iki esas üzerine oturtulmuştur:

1. İleri yönde öngörü hata süzgecinin tap ağırlık vektörleri,

$$\mathbf{a}_m = \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} + \Gamma_m \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix} \quad (3.44)$$

biçiminde derece güncelleştirilebilir. Bu ifadedeki Γ_m bir sabittir. Bu derece güncellemesinin skaler formu,

$$a_{m,k} = a_{m-1,k} + \Gamma_m a_{m-1,m-k}^* \quad k=0,1,\dots,m \quad (3.45)$$

şeklinde dir. Burada $a_{m,k}$, m . dereceden ileri yönde öngörü hatası süzgecinin k . tap ağırlığıdır. $a_{m-1,m-k}^*$ ise, $(m-1)$. dereceden geri yönde öngörü hatası süzgecinin k . tap ağırlığıdır. Ayrıca (3.45) eşitliğinde, $a_{m-1,0}=1$ ve $a_{m-1,m}=0$ 'dır.

2. Geri yönde öngörü hata süzgecinin tap ağırlık vektörleri,

$$\mathbf{a}_m^{B*} = \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix} + \Gamma_m^* \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} \quad (3.46)$$

biçiminde derece güncelleştirilebilir. Bu derece güncellemesinin skaler formu,

$$a_{m-m-k}^* = a_{m-1,m-k}^* + \Gamma_m^* a_{m-1,k} \quad k=0,1,\dots,m \quad (3.47)$$

şeklindedir. Burada $a_{m,m-k}^*$, m . dereceden geri yönde öngörü hata süzgecinin k . tap ağırlığıdır. Diğer elemanlar da benzer biçimde tanımlanabilir.

Levinson-Durbin özyinelemesinin geçerliliğinin ispatında, Γ_m sabitinin sağlaması gereken şartları belirlemek amacıyla aşağıdaki dört aşamayı gerçekleştirelim:

1. $\mathbf{R}_{m+1}$, m . dereceden ileri yönde öngörü hata süzgecinin tap girişleri $u(n)$, $u(n-1)$, ... $u(n-m)$ ' nin $(m+1) \times (m+1)$ boyutlu ilişki matrisidir. (3.44) eşitliğinin her iki tarafını $\mathbf{R}_{m+1}$ ile çarpalım. Eşitliğin sol tarafı için, (3.15) eşitliğinden yararlanılarak,

$$\mathbf{R}_{m+1} \mathbf{a}_m = \begin{bmatrix} P_m \\ \mathbf{0}_m \end{bmatrix} \quad (3.48)$$

yazılabilir.

2. $\mathbf{R}_{m+1}$ ilişki matrisinin parçalanmış biçimi (Haykin, 1991),

$$\mathbf{R}_{m+1} = \begin{bmatrix} \mathbf{R}_m & \mathbf{r}_m^{B*} \\ \mathbf{r}_m^{BT} & r(0) \end{bmatrix}$$

şeklinde gösterilebilir. Burada $\mathbf{R}_m$, $u(n)$, $u(n-1)$, ..., $u(n-m+1)$ tap girişleri için ilişki matrisi ve $\mathbf{r}_m^{B*}$, tap girişleri ile istenen yanıt $u(n-m)$ arasındaki çapraz ilişki vektörüdür. (3.44) eşitliğinin sağ tarafındaki ilk terim için,

$$\begin{aligned} \mathbf{R}_{m+1} \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} &= \begin{bmatrix} \mathbf{R}_m & \mathbf{r}_m^{B*} \\ \mathbf{r}_m^{BT} & r(0) \end{bmatrix} \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{R}_m \mathbf{a}_{m-1} \\ \mathbf{r}_m^{BT} \mathbf{a}_{m-1} \end{bmatrix} \end{aligned} \quad (3.49)$$

ifadesi elde edilir.

($m-1$). dereceden ileri yönde öngörü hata süzgeci için genişletilmiş Wiener-Hopf eşitliği,

$$\mathbf{R}_m \mathbf{a}_{m-1} = \begin{bmatrix} P_{m-1} \\ 0_{m-1} \end{bmatrix} \quad (3.50)$$

ile verilebilir. Burada,

$$\Delta_{m-1} = \mathbf{r}_m^{BT} \mathbf{a}_{m-1} = \sum_{k=0}^{m-1} a_{m-1,k} r(k-m) \quad (3.51)$$

şeklinde bir skaler tanımlanarak,

$$\mathbf{R}_{m+1} \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} = \begin{bmatrix} P_{m-1} \\ 0_{m-1} \\ \Delta_{m-1} \end{bmatrix} \quad (3.52)$$

ifadesi bulunur.

3. $\mathbf{R}_{m+1}$ ilişki matrisinin parçalanmış biçimi (Haykin, 1991),

$$\mathbf{R}_{m+1} = \begin{bmatrix} r(0) & \mathbf{r}_m^H \\ \mathbf{r}_m & \mathbf{R}_m \end{bmatrix}$$

şeklinde de gösterilebilir. Burada $\mathbf{R}_m$, $u(n-1)$, $u(n-2)$, ..., $u(n-m)$ tap girişleri için ilişki matrisi ve $\mathbf{r}_m$, bu tap girişleri ile istenen yanıt $u(n)$ arasındaki çapraz ilişki vektörüdür.

(3.44) eşitliğinin sağ tarafındaki ikinci terim için,

$$\begin{aligned} \mathbf{R}_{m+1} \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^B * \end{bmatrix} &= \begin{bmatrix} r(0) & \mathbf{r}_m^H \\ \mathbf{r}_m & \mathbf{R}_m \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^B * \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{r}_m^H \mathbf{a}_{m-1}^B * \\ \mathbf{R}_m \mathbf{a}_{m-1}^B * \end{bmatrix} \end{aligned} \quad (3.53)$$

ifadesi elde edilir. $\mathbf{r}_m^H \mathbf{a}_{m-1}^B *$ skaler değeri için,

$$\mathbf{r}_m^H \mathbf{a}_{m-1}^B * = \sum_{l=1}^m r(l) a_{m-1, m-l}^* = \Delta_{m-1}^* \quad (3.54)$$

eşitliği yazılabilir.

($m-1$). dereceden geri yönde öngörü hata süzgeci için genişletilmiş Wiener-Hopf eşitliği,

$$\mathbf{R}_m \mathbf{a}_{m-1}^B * = \begin{bmatrix} 0_{m-1} \\ P_{m-1} \end{bmatrix} \quad (3.55)$$

şeklinde verilebilir. (3.54) ve (3.55) eşitliklerinin (3.53) eşitliğinde yerine yazılmasıyla,

$$\mathbf{R}_{m+1} \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^B * \end{bmatrix} = \begin{bmatrix} \Delta_{m-1}^* \\ 0_{m-1} \\ P_{m-1} \end{bmatrix} \quad (3.56)$$

ifadesi bulunur.

4. Bu üç adımda elde ettiğimiz sonuçları, (3.44) eşitliğinin her iki yanının $\mathbf{R}_{m+1}$ ile çarpılması durumu için yerlerine yazarsak,

$$\begin{bmatrix} P_m \\ 0_m \end{bmatrix} = \begin{bmatrix} P_{m-1} \\ 0_{m-1} \\ \Delta_{m-1} \end{bmatrix} + \Gamma_m \begin{bmatrix} \Delta_{m-1}^* \\ 0_{m-1} \\ P_{m-1} \end{bmatrix} \quad (3.57)$$

ifadesi elde edilir ve bu son eşitlikten iki önemli sonuç çıkarılabilir:

1. (3.57) eşitliğinin sağ ve sol tarafındaki vektörlerin ilk elemanlarından,

$$P_m = P_{m-1} + \Gamma_m \Delta_{m-1}^* \quad (3.58)$$

eşitliği elde edilebilir.

2. Aynı şekilde, aynı eşitliğin sağ ve sol tarafındaki vektörlerin son elemanlarından,

$$0 = \Delta_{m-1} + \Gamma_m P_{m-1} \quad (3.59)$$

eşitliği çıkarılabilir. Bu eşitlik yeniden düzenlenirse, Γ_m sabitinin değerini

$$\Gamma_m = -\frac{\Delta_{m-1}}{P_{m-1}} \quad (3.60)$$

şeklinde buluruz. (3.60) eşitliğinden Δ_{m-1} değeri çekilir ve bu ifade (3.58) eşitliğinde yerine yazılırsa öngörü hatası gücü,

$$P_m = P_{m-1} (1 - |\Gamma_m|^2) \quad (3.61)$$

şeklinde ifade edilebilir. Öngörü hata süzgecinin derecesi m artarken, P_m değeri azalacaktır. P_m değeri hiç bir zaman negatif değer alamayacağından her zaman için,

$$0 \leq P_m \leq P_{m-1} \quad , m \geq 1 \quad (3.62)$$

bağıntısı sağlanacaktır. Aynı zamanda sıfır anındaki hata gücünün, giriş sürecinin özilişki fonksiyonunun sıfır gecikmesi değerine eşit olduğu bilinmektedir:

$$P_0 = r(0)$$

(3.61) eşitliğinde, $m=0$ ' dan başlayıp, birer birer artırılıp öngörü hata süzgecinin son derecesi M e gelindiğinde öngörü hatası gücü,

$$P_M = P_0 \prod_{m=1}^M (1 - |\Gamma_m|^2) \quad (3.63)$$

ifadesiyle verilebilir.

3.3.1. Γ_m ve Δ_{m-1} Parametrelerinin Açıklanması

$1 \leq m \leq M$ aralığında Γ_m parametreleri, son derecesi M olan öngörü hata süzgecine Levinson-Durbin özyinelemesinin uygulanması sonucunda elde edilen yansım katsayılarıdır. (3.62) eşitliğinde verilen güç değerinin sınır aralıklarından faydalanılarak,

$$|\Gamma_m| \leq 1 \quad , \text{ bütün } m \quad (3.64)$$

bağıntısı yazılabilir.

m dereceli bir öngörü hata süzgecinin yansım katsayısı bu süzgecin son tap ağırlığına eşittir:

$$\Gamma_m = a_{m,m}$$

Δ_{m-1} değeri ise, ileri yönde öngörü hatası $f_{m-1}(n)$ ile geciktirilmiş geri yönde öngörü hatası $b_{m-1}(n-1)$ arasındaki çapraz ilişkiyi gösterir:

$$\Delta_{m-1} = E[b_{m-1}(n-1)f_{m-1}^*(n)] \quad (3.65)$$

Burada $f_{m-1}(n)$, $u(n)$, $u(n-1)$, ..., $u(n-m+1)$ tap girişleri için $(m-1)$. dereceden ileri yönde öngörü hatası süzgecinin çıkışıdır. $b_{m-1}(n-1)$ ise, $u(n-1)$, $u(n-2)$, ..., $u(n-m)$ tap girişleri için $(m-1)$. dereceden geri yönde öngörü hatası süzgecinin çıkışıdır. Ayrıca,

$$f_0(n)=b_0(n)=u(n)$$

olduğundan, (3.65) eşitliğinden yararlanarak sıfır anındaki Δ_{m-1} değeri

$$\begin{aligned} \Delta_0 &= E[b_0(n-1)f_0^*(n)] \\ &= E[u(n-1)u^*(n)] \\ &= r^*(1) \\ &= r(-1) \end{aligned}$$

biçiminde elde edilir. (3.60) ve (3.65) eşitliklerinden faydalanılarak yansımaya katsayılarını,

$$\Gamma_m = -\frac{E[b_{m-1}(n-1)f_{m-1}^*(n)]}{E[|f_{m-1}(n)|^2]} \quad (3.66)$$

biçiminde yeniden formüle edebiliriz. Eşitliğin sağ tarafı, negatif işaret hariç, kısmi ilişki katsayısı (Partial Correlation, PARCOR) olarak bilinir. Sonuç olarak, yansımaya katsayısının, PARCOR katsayısının negatifi olduğu görülür.

3.3.2. Levinson-Durbin Özyinelemesinin Uygulanması

Son öngörü derecesi M için, öngörü hatası gücü P_M ve öngörü hatası süzgeci tap ağırlıkları $a_{M,k}$, $k=0, 1, \dots, M$, değerlerinin hesaplanması için Levinson-Durbin özyinelemesinin iki şekilde uygulanması söz konusudur:

1. Giriş işaretinin özilişki fonksiyonunun $r(0), r(1), \dots, r(M)$ değerleri zamana göre ortalaması alınan bir formül ile hesaplanabilir:

$$\hat{r}(k) = \frac{1}{N-k} \sum_{n=1}^{N-k} u(n)u^*(n-k), \quad k = 0, 1, \dots, M \quad (3.67)$$

N değeri, $N \gg M$ koşulu altında giriş serisinin toplam uzunluğudur. Yukarıdaki ifadeden elde edilen özilişki değerleri ile, (3.51) eşitliği kullanılarak Δ_{m-1} değeri ve (3.61) eşitliği kullanılarak da hata gücü P_m hesaplanabilir. Özyinelemenin başlanıç değeri $m = 0$ için $P_0 = r(0)$ ve $\Delta_0 = r^*(1)$ kullanılır. Ayrıca $a_{m,0} = 0$ (her $k > m$ için) eşitlikleri kullanılır. Öngörü hata süzgeci katsayıları ve öngörü hata gücünün kestirim değerlerinin bulunduğu bu yöntem Yule-Walker kestirimi olarak bilinir.

2. Yansıma katsayıları $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ değerleri ve özilişki fonksiyonunun sıfır gecikmesi için değeri $r(0)$ ' nin kesin olarak bilindiği durumlarda uygulanır. Bu değerler,

$$a_{m,k} = a_{m-1,k} + \Gamma_m a_{m-1,m-k}^*, \quad k = 0, 1, \dots, M$$

$$P_m = P_{m-1} (1 - |\Gamma_m|^2)$$

ifadelerinde kullanılarak özyineleme gerçekleştirilir. Özyineleme $m = 0$ ' dan başlar ve M son değerinde biter.

3.3.3. Levinson-Durbin Özyinelemesinin Ters Biçimi

Levinson-Durbin özyinelemesinin normal uygulamasında $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ değerleri verildiğinde $a_{M,1}, a_{M,2}, \dots, a_{M,M}$ tap ağırlıklarının hesaplanması istenir. Bazen bunun tersi gerekebilir. Yani; $a_{M,1}, a_{M,2}, \dots, a_{M,M}$ tap ağırlıkları verildiğinde $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ yansıma katsayılarının hesaplanması istenebilir. Bu problem ters özyineleme olarak adlandırabileceğimiz Levinson-Durbin özyinelemesinin tersi uygulanarak çözülebilir.

Ters özyinelemenin elde edilmesi için (3.45) ve (3.47) eşitliklerindeki skaler ifadeler ortak bir matriste birleştirilirse,

$$\begin{bmatrix} a_{m,k} \\ a_{m-m-k}^* \end{bmatrix} = \begin{bmatrix} 1 & \Gamma_m \\ \Gamma_m^* & 1 \end{bmatrix} \begin{bmatrix} a_{m-1,k} \\ a_{m-1,m-k}^* \end{bmatrix}, \quad k = 0, 1, \dots, m \quad (3.68)$$

eşitliği elde edilir ve bu ifadeden tap ağırlıklarının çözümü için,

$$a_{m-1,k} = \frac{a_{m,k} - a_{m,m} a_{m-m-k}^*}{1 - |a_{m,m}|^2}, \quad k = 0, 1, \dots, m \quad (3.69)$$

ifadesi elde edilir.

Ters özyineleme ile çözüme başlarken $\Gamma_m = a_{m,m}$ koşulu esas alınıp, tap ağırlıklarının hesaplanması için $m = M, M-1, M-2, \dots, 1$ sırası takip edilerek çözüm gerçekleştirilir.

3.4. Kafes Öngörücüler

Kafes öngörücüler, bir çok ileri ve geri yönde öngörü hata süzgeçleme işlemlerinin tek bir yapıda birleştirilmesiyle gerçekleştirilirler. Bir kafes öngörücüsü, adından da anlaşılacağı gibi, ardarda bağlanmış kafes şeklindeki yapılardan oluşur. Kafes öngörücülerdeki katların sayısı öngörü derecesine eşittir. Yani, m dereceli bir öngörü hata süzgecini gerçekleştirmek için m katlı bir kafes yapısı oluşturmak gerekir.

3.4.1. Öngörü Hatalarının Derece Güncelleme Özyinelemeleri

Kafes öngörücülerin giriş çıkış bağıntıları, Levinson-Durbin algoritmasından faydalanılarak, çeşitli yollarla çıkartılabilir. Bu işleme, öngörü hata süzgeçlerinin ileri ve geri işlemlerine ait matris formundaki (3.44) ve (3.46) eşitliklerinden faydalanarak başlayalım. Bu eşitlikler aşağıda verildiği biçimde ifade edilmiştir:

$$\mathbf{a}_m = \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} + \Gamma_m \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix} \quad (3.70)$$

$$\mathbf{a}_m^{B*} = \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix} + \Gamma_m^* \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} \quad (3.71)$$

$(m+1) \times 1$ boyutlu $\mathbf{a}_m$ ve $m \times 1$ boyutlu $\mathbf{a}_{m-1}$ vektörleri sırasıyla m . ve $(m-1)$. dereceden ileri yönde öngörü hata süzgeçlerine karşılık düşen tap ağırlıklarıdır. $(m+1) \times 1$ boyutlu $\mathbf{a}_m^{B*}$ ve $m \times 1$ boyutlu $\mathbf{a}_{m-1}^{B*}$ vektörleri de sırasıyla m . ve $(m-1)$. dereceden geri yönde öngörü hata süzgeçlerinin tap ağırlıklarına karşılık düşer. Γ_m değeri ise yansıma katsayısı olarak adlandırılır.

Öncelikle m . dereceden ileri yönde öngörü hata süzgecinin $u(n)$, $u(n-1)$, ..., $u(n-m)$ ile gösterilen tap girişlerini ele alalım. Bu süzgecin $(m+1) \times 1$ boyutlu tap giriş vektörü $\mathbf{u}_{m+1}(n)$ ' i aşağıdaki formda görüldüğü gibi parçalayalım:

$$\mathbf{u}_{m+1}(n) = \begin{bmatrix} \mathbf{u}_m(n) \\ \dots\dots\dots \\ u(n-m) \end{bmatrix} \quad (3.72)$$

Ya da eşdeğer formda,

$$\mathbf{u}_{m+1}(n) = \begin{bmatrix} u(n) \\ \dots\dots\dots \\ \mathbf{u}_m(n-1) \end{bmatrix} \quad (3.73)$$

yazılabilir.

(3.70) eşitliğindeki $\mathbf{a}_m$ vektörünün Hermitian dönüşümü alınıp, bu eşitliğin her iki tarafı $\mathbf{u}_{m+1}$ vektörü ile çarpılırsa;

1. Eşitliğin sol tarafındaki ilk terim için,

$$f_m(n) = \mathbf{a}_m^H \mathbf{u}_{m+1}(n) \quad (3.74)$$

ifadesi elde edilir. $f_m(n)$, m . dereceden ileri yönde öngörü hata süzgecinin çıkışında üretilen ileri yönde öngörü hatasıdır.

2. Eşitliğin sağ tarafındaki ilk terim için (3.72) eşitliğinde verilen $\mathbf{u}_{m+1}$ vektörünün parçalanmış biçimi kullanılarak,

$$\begin{aligned} \begin{bmatrix} \mathbf{a}_{m-1}^H & 0 \end{bmatrix} \mathbf{u}_{m+1}(n) &= \begin{bmatrix} \mathbf{a}_{m-1}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_m(n) \\ \dots\dots\dots \\ u(n-m) \end{bmatrix} \\ &= \mathbf{a}_{m-1}^H \mathbf{u}_m(n) \\ &= f_{m-1}(n) \end{aligned} \quad (3.75)$$

ifadesi elde edilir. Burada $f_{m-1}(n)$, $(m-1)$. dereceden ileri yönde öngörü hata süzgecinin çıkışında üretilen ileri yönde öngörü hatasıdır.

3. Eşitliğin sağ tarafındaki ikinci terim için (3.73) eşitliğinde verilen $\mathbf{u}_{m+1}(n)$ vektörünün parçalanmış biçimi kullanılarak,

$$\begin{aligned} \begin{bmatrix} 0 & \mathbf{a}_{m-1}^{BT} \end{bmatrix} \mathbf{u}_{m+1}(n) &= \begin{bmatrix} 0 & \mathbf{a}_{m-1}^{BT} \end{bmatrix} \begin{bmatrix} u(n) \\ \dots\dots\dots \\ \mathbf{u}_m(n-1) \end{bmatrix} \\ &= \mathbf{a}_{m-1}^{BT} \mathbf{u}_m(n-1) \\ &= b_{m-1}(n-1) \end{aligned} \quad (3.76)$$

ifadesi elde edilir. Burada $b_{m-1}(n-1)$, $(m-1)$. dereceden geri yönde öngörü hata süzgecinin çıkışında üretilen geri öngörü hatasının geciktirilmişidir. (3.74), (3.75) ve (3.76) eşitlikleri birleştirilerek,

$$f_m(n) = f_{m-1}(n) + \Gamma_m * b_{m-1}(n-1) \quad (3.77)$$

elde edilir. Benzer şekilde, m . dereceden geri yönde öngörü hata süzgecinin $u(n)$, $u(n-1)$, ..., $u(n-m)$ ile gösterilen tap girişlerini ele alalım. (3.71) eşitliğindeki $\mathbf{a}_m^{B*}$ vektörünün Hermitian dönüşümü alınıp, bu eşitliğin her iki tarafı $\mathbf{u}_{m+1}(n)$ vektörü ile çarpılırsa,

1. Eşitliğin sol tarafındaki ilk terim için,

$$b_m(n) = \mathbf{a}_m^{BT} \mathbf{u}_{m+1}(n) \quad (3.78)$$

ifadesi elde edilir. Burada $b_m(n)$, m . dereceden geri yönde öngörü hata süzgecinin çıkışında üretilen geri yönde öngörü hatasıdır.

2. Eşitliğin sağ tarafındaki ilk terim için (3.73) eşitliğinde verilen $\mathbf{u}_{m+1}(n)$ vektörünün parçalanmış biçimi kullanılarak,

$$\begin{aligned} \begin{bmatrix} 0 & \mathbf{a}_{m-1}^{BT} \end{bmatrix} \mathbf{u}_{m+1}(n) &= \begin{bmatrix} 0 & \mathbf{a}_{m-1}^{BT} \end{bmatrix} \begin{bmatrix} u(n) \\ \dots\dots\dots \\ \mathbf{u}_m(n-1) \end{bmatrix} \\ &= \mathbf{a}_{m-1}^{BT} \mathbf{u}_m(n-1) \\ &= b_{m-1}(n-1) \end{aligned} \quad (3.79)$$

elde edilir.

3. Eşitliğin sağ tarafındaki ikinci terim için (3.72) eşitliğinde verilen $\mathbf{u}_{m+1}(n)$ vektörünün parçalanmış biçimi kullanılarak,

$$\begin{aligned} \begin{bmatrix} \mathbf{a}_{m-1}^H & 0 \end{bmatrix} \mathbf{u}_{m+1}(n) &= \begin{bmatrix} \mathbf{a}_{m-1}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_m(n) \\ \dots\dots\dots \\ u(n-m) \end{bmatrix} \\ &= \mathbf{a}_{m-1}^H \mathbf{u}_m(n) \\ &= f_{m-1}(n) \end{aligned} \quad (3.80)$$

elde edilir. (3.78) , (3.79) ve (3.80) eşitlikleri birleştirilerek,

$$b_m(n) = b_{m-1}(n-1) + \Gamma_m f_{m-1}(n) \quad (3.81)$$

elde edilir. (3.77) ve (3.81) eşitlikleri, m . dereceden kafes öngörücüsü için düzenlenen derece güncelleme özyinelemeleridir. Bu eşitlikler birleştirilerek matris formda yazılabilirler:

$$\begin{bmatrix} f_m(n) \\ b_m(n) \end{bmatrix} = \begin{bmatrix} 1 & \Gamma_m^* \\ \Gamma_m & 1 \end{bmatrix} \begin{bmatrix} f_{m-1}(n) \\ b_{m-1}(n-1) \end{bmatrix}, \quad m = 1, 2, \dots, M \quad (3.82)$$

Bu matrislerde geçen $b_{m-1}(n-1)$ terimi $b_{m-1}(n)$ cinsinden,

$$b_{m-1}(n) = Z^{-1}[b_{m-1}(n)] \quad (3.83)$$

biçiminde yazılabilir. Böylece, (3.82) ve (3.83) eşitliklerinden yararlanılarak oluşturulan kafes öngörücünün m . katının yapısı Şekil 3.3 (a)' da gösterilmiştir.

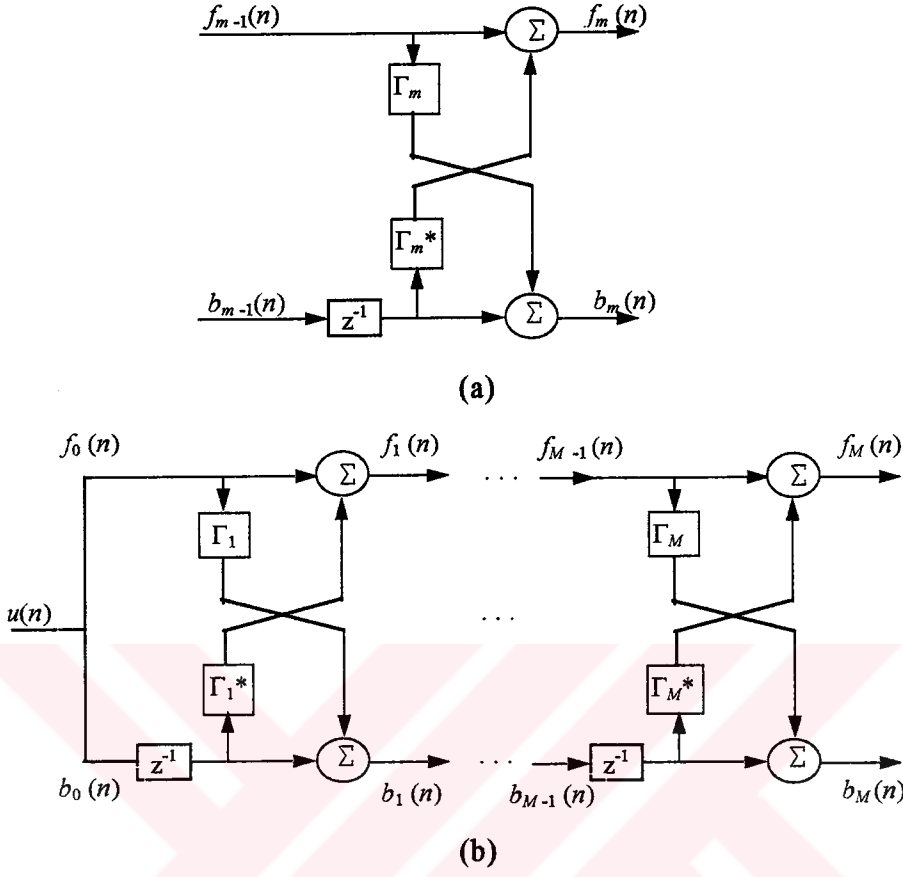
Başlangıç anı, $m=0$ için,

$$f_0(n) = b_0(n) = u(n) \quad (3.84)$$

yazılabilir. $u(n)$, n anındaki giriş işaretidir. $m = 0$ ' dan başlayarak ve süzgecin derecesini birer arttırarak, Şekil 3.3 (b)' de de gösterildiği gibi, son katının derecesi M olan bir öngörü hata süzgeci için eşdeğer kafes yapısı elde edilebilir. Bu süzgecin işlevinin anlaşılması ve yerine getirilebilmesi için $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ yansıma katsayılarının bilinmesi gerekir.

Bir kafes süzgeç aşağıdaki özelliklere sahiptir:

1. Bir kafes süzgeç yapısı itibariyle ileri ve geri yönde öngörü hataları dizilerini aynı anda üretir.
2. Bir kafes süzgeç modüler yapıdadır. Eğer öngörücünün derecesi arttırılmak isteniyorsa, daha önceki hesaplamalara dokunmadan, istenen dereceye ulaşmak için gereken katlar eklenebilir.



Şekil 3.3 (a) Bir Kafes Öngörücünün m . Katı
(b) M Dereceli Bir Kafes Öngörücü

3.4.2. Kafes Süzgeçlerin İlişki Özellikleri

Özellik 1. İleri yönde öngörü hatası $f_m(n)$ ile giriş vektörü $u(n)$ birbirine diktir:

$$E[f_m(n)u^*(n-k)] = 0, \quad 1 \leq k \leq m \quad (3.85)$$

Benzer şekilde, geri yönde öngörü hatası $b_m(n)$ de giriş vektörü $u(n)$ ' e diktir:

$$E[b_m(n)u^*(n-k)] = 0, \quad 1 \leq k \leq m-1 \quad (3.86)$$

Özellik 2. İleri yönde öngörü hatası $f_m(n)$ ile giriş işareti $u(n)$ ' nin çapraz ilişkisi, geri yönde öngörü hatası $b_m(n)$ ile zamanda kaydırılmış giriş işareti $u(n-m)$ arasındaki çapraz ilişkiye eşittir ve bu değer öngörü hata gücü P_m ' e eşittir:

$$E[f_m(n)u^*(n)] = E[b_m(n)u^*(n-m)] = P_m \quad (3.87)$$

Özellik 3. Geri yönde öngörü hataları birbirine diktir:

$$E[b_m(n)b_i^*(n)] = \begin{cases} P_m, & m = i \\ 0, & m \neq i \end{cases} \quad (3.88)$$

İleri yönde öngörü hataları ise bu özelliğe uymazlar. Aralarında aşağıdaki şekilde gösterildiği gibi bir ilişki söz konusudur:

$$E[f_m(n)f_i^*(n)] = P_m, \quad m \geq i \quad (3.89)$$

Özellik 4. İleri ve geri yönde öngörü hatalarının zamanda geciktirilmişleri birbirlerine diktir:

$$E[f_m(n)f_i^*(n-l)] = E[f_m(n+l)f_i^*(n)] = 0, \quad 1 \leq l \leq m-i \text{ ve } m > i \quad (3.90)$$

$$E[b_m(n)b_i^*(n-l)] = E[b_m(n+l)b_i^*(n)] = 0, \quad 0 \leq l \leq m-i-l \text{ ve } m > i \quad (3.91)$$

Özellik 5. Zamanda geciktirilmiş ileri yönde öngörü hataları $f_m(n+m)$ ve $f_i(n+i)$ birbirlerine diktir:

$$E[f_m(n+m)f_i^*(n+i)] = \begin{cases} P_m, & m = i \\ 0, & m \neq i \end{cases} \quad (3.92)$$

Zamanda geciktirilmiş geri yönde öngörü hataları $b_m(n+m)$ ve $b_i(n+i)$ ise aşağıdaki gibi ilişkilidir:

$$E[b_m(n+m)b_i^*(n+i)] = P_m, m \geq i \quad (3.93)$$

Özellik 6. İleri ve geri yönde öngörü hataları aşağıdaki çapraz ilişki özelliklerine sahiptir:

$$E[f_m(n)b_i^*(n)] = \begin{cases} \Gamma_i^* P_m & , m \geq i \\ 0 & , m < i \end{cases} \quad (3.94)$$

3.5. Gram-Schmidt Dikleştirme Yöntemiyle İleri Ve Geri Yönde Öngörü Hata Dizilerinin Oluşturulması

Kafes süzgeçlerdeki m katlı yapıya karşılık gelen geri yönde öngörü hata değerleri $b_0(n)$, $b_1(n)$, ..., $b_m(n)$ yerine, Gram-Schmidt dikleştirme özelliğinin, giriş değerleri $u(n)$, $u(n-1)$, ..., $u(n-m)$ ' e uygulanmasıyla değişik bir bakış açısı elde edilebilir.

Gram-Schmidt dikleştirme özelliği genel olarak şu şekilde ifade edilebilir: Birbirleriyle doğrusal olarak bağımsız (ilişkili) $(m+1)$ tane rasgele değişken u_0 , u_1 , ..., u_m kullanılarak, v_0 , v_1 , ..., v_m gibi birbirlerine dik (ilişkisiz) $(m+1)$ tane rasgele değişken oluşturulabilir. Bu değişkenler, başlangıçta verilen u_0 , u_1 , ..., u_m değerlerinin doğrusal bir kombinasyonudur. Bu özellik kullanılarak, şu şekilde bazı genellemeler yapmak mümkündür:

1. M katlı bir kafes öngörücü için $b_0(n)$, $b_1(n)$, ..., $b_M(n)$ değerlerinin oluşturulması ile Gram-Schmidt dikleştirme özelliğinin, giriş değerleri $u(n)$, $u(n-1)$, ..., $u(n-M)$ ' e uygulanması birbirine eşdeğerdir. Bu genelleme geri öngörü hata değerlerinin birbirine dik olma özelliğinin bir sonucu olarak düşünülebilir.

2. Tap giriş değerleri vektörü $u(n)$ ' nin, geri yönde öngörü hata değerleri vektörü $b(n)$ ' e çevrilebilmesi için, bir L üçgen matrisi ile çarpılması gereklidir. Bu matrisin sıfır değerlerinin karşısında, bulunduğu satırdaki dereceye ait süzgeç tap ağırlıkları yer alır.

$$b(n) = Lu(n) \quad (3.95)$$

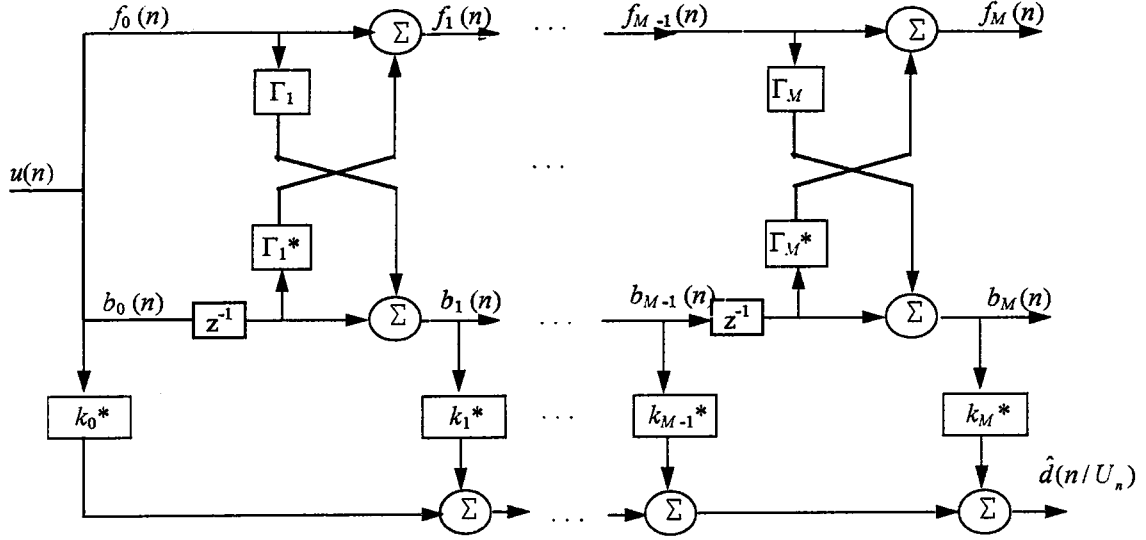
$$L = \begin{bmatrix} 1 & 0 & \dots & 0 \\ a_{1,1} & 1 & \dots & 0 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ a_{M,M} & a_{M,M-1} & \dots & 1 \end{bmatrix} \quad (3.96)$$

3. $u(n)$ vektörü ile $b(n)$ vektörünün elemanlarının birebir karşılaştırılması yöntemiyle, birinden diğerine geçişte bilgi kaybına uğramamaya çalışılır.

Aslında $u(n)$ vektörünün $b(n)$ vektörüne dönüşümünde pek çok karışıklık ortaya çıkacaktır. Bu karışıklıkları giderebilmek için özellikle birleşik süreç kestirimi kullanılır.

3.6. Birleşik Süreç Kestirimi

Karesel ortalama anlamda optimum olan birleşik süreç kestirim probleminin çözümü için, öncelikle altyapı olarak kafes öngörücü kullanılır. Bunun yanında, ilişkili bir süreç olan $\{u(n)\}$ den türetilen gözlenebilir küme kullanılarak oluşturulan ve istenen yanıt olarak adlandırılan $\{d(n)\}$ sürecinin minimum karesel ortalama kestirimini düşünelim. $\{u(n)\}$ ve $\{d(n)\}$ süreçlerinin birleşik durağan olduğu varsayılır. Bu kestirim problemi bir temel fark dışında Bölüm 2' deki kestirim problemine benzer. Bölüm 2' de, $\{u(n)\}$ süreci örnek değerleri doğrudan gözlenebilir değerler olarak kullanılmıştır. Burada ise, gözlenebilir değerler olarak, girişinde $\{u(n)\}$ süreci örnek değerleri olan çok katlı kafes öngörücü ile elde edilen geri yönde öngörü hataları dizisi kullanılmaktadır. Geri yönde öngörü hatalarının her birinin birbirine dik olması özelliği, problemin çözümünü basitleştirmektedir.



Şekil 3.4 Birleşik Süreç Kestirimi İçin Kafes Yapı

Şekil 3.4’ de birleşik süreç kestirimi için düzenlenen kafes yapı gösterilmiştir. Bu yapı iki optimum kestirim gerçekleştirir:

1. Kafes öngörücü, ardarda bağlanmış M kattan oluşmuştur ve $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ yansıma katsayıları ile karakterize edilir. $u(n), u(n-1), \dots, u(n-M)$ giriş dizisini (ilişkili), $b_0(n), b_1(n), \dots, b_M(n)$ geri öngörü hataları dizisine (ilişkisiz) dönüştürmede kullanılır.

2. Çoklu regresyon süzgeci, $k_0, k_1, \dots, k_M$ değerleriyle karakterize edilir. $b_0(n), b_1(n), \dots, b_M(n)$ geri öngörü hataları değerleri, istenen yanıt $d(n)$ ’ nin kestiriminde kullanılır.

Sonuç olarak, istenen yanıtın kestirim değeri,

$$\hat{d}(n | U_n) = \sum_{i=0}^M k_i^* b_i(n) \quad (3.97)$$

$$\hat{d}(n | U_n) = \mathbf{k}^H \mathbf{b}(n) \quad (3.98)$$

biçiminde toplamsal veya matris formda ifade edilebilir. U_n değeri, giriş değerleri $u(n)$, $u(n-1)$, ..., $u(n-M)$ ile boyutlandırılmış uzaydır. $\mathbf{k}$ ise, k_0 , k_1 , ..., k_M regresyon katsayılarından oluşmuş,

$$\mathbf{k}^T = [k_0, k_1, \dots, k_M] \quad (3.99)$$

biçiminde $(M+1) \times 1$ boyutlu bir regresyon vektörüdür.

$\mathbf{b}(n)$ ' nin $(M+1) \times (M+1)$ boyutlu ilişki matrisi $\mathbf{D}$ ile, istenen yanıtla geri öngörü hataları arasındaki çapraz ilişki vektörü de $\mathbf{s}$ ile gösterilsin:

$$\mathbf{D} = E[\mathbf{b}(n)\mathbf{b}^H(n)] \quad (3.100)$$

$$\mathbf{s} = E[\mathbf{b}(n)d^*(n)] \quad (3.101)$$

Wiener-Hopf eşitliği bu değerlere uygulanırsa,

$$\mathbf{D}\mathbf{k}_o = \mathbf{s} \quad (3.102)$$

yazılabilir. Optimum tap vektörü $\mathbf{k}_o$ ' nm çözümü için,

$$\mathbf{k}_o = \mathbf{D}^{-1}\mathbf{s} \quad (3.103)$$

yazılabilir. (3.100) eşitliğinde $\mathbf{b}(n)$ yerine (3.95) eşitliğinin sağ tarafındaki terim yazılırsa,

$$\mathbf{D} = \mathbf{LRL}^H \quad (3.104)$$

elde edilir.

Ters matris $\mathbf{D}^{-1}$ diagonaldır ve

$$\mathbf{D}^{-1} = \mathbf{LR}^{-1}\mathbf{L}^H \quad (3.105)$$

yazılabilir. (3.105) eşitliği (3.103) eşitliğine yerleştirilirse,

$$\mathbf{k}_o = \mathbf{L}\mathbf{R}^{-1}\mathbf{L}^H \mathbf{s} \quad (3.106)$$

elde edilir. (3.101) eşitliğinde $\mathbf{b}(n)$ yerine (3.95) eşitliğinin sağ tarafındaki terim yazılırsa,

$$\begin{aligned} \mathbf{s} &= \mathbf{L}\mathbf{E}[\mathbf{u}(n)\mathbf{d}^*(n)] \\ &= \mathbf{L}\mathbf{p} \end{aligned} \quad (3.107)$$

elde edilir. Böylece, (3.107) eşitliği (3.106) eşitliğinde yerine yazılırsa,

$$\begin{aligned} \mathbf{k}_o &= \mathbf{L}\mathbf{R}^{-1}\mathbf{L}^H \mathbf{L}\mathbf{p} \\ &= \mathbf{L}\mathbf{R}^{-1}\mathbf{p} \\ &= \mathbf{L}\mathbf{w}_o \end{aligned} \quad (3.108)$$

elde edilir. Burada $\mathbf{L}$, (3.96)' da tanımlandığı gibi geri öngörü hata süzgeci tap ağırlıklarından oluşmuş bir matristir.

BÖLÜM 4. UYARLAMALI SÜZGEÇLERİN YAPISI

Uyarlamalı süzgeçler, Şekil 4.1' de gösterildiği gibi iki temel bölümden oluşur: (1) n anındaki $w_1(n), w_2(n), \dots, w_M(n)$ değerleriyle gösterilen ayarlanabilir tap ağırlıklarını içeren bir enine süzgeç, ve (2) bu tap ağırlıklarını uyarlamalı bir yöntemle ayarlayan bir mekanizma. Süzme süreci esnasında, Bölüm 2' de açıklanan ve istenen yanıt olarak adlandırılan ilave bir işaret $d(n)$, süzgecin tap ağırlıklarının ayarlanabilmesi için gereklidir. n anındaki giriş vektörü $u(n)$ ile süzgeç çıkışında istenen yanıtın uygun kestirimi de $\hat{d}(n | U_n)$ ile gösterilmiştir. İstenen yanıt $d(n)$ ' nin gerçek değeri ile kestirim değerinin karşılaştırılması sonucu elde edilen kestirim hatası, daha önce Bölüm 2' de (2.15) eşitliğiyle verilen ifadede $w(n)$ katsayıları kullanılarak yeni biçimiyle,

$$\begin{aligned} e(n) &= d(n) - \hat{d}(n | U_n) \\ &= d(n) - \mathbf{w}^H(n)u(n) \end{aligned} \quad (4.1)$$

Şeklinde yazılabilir.

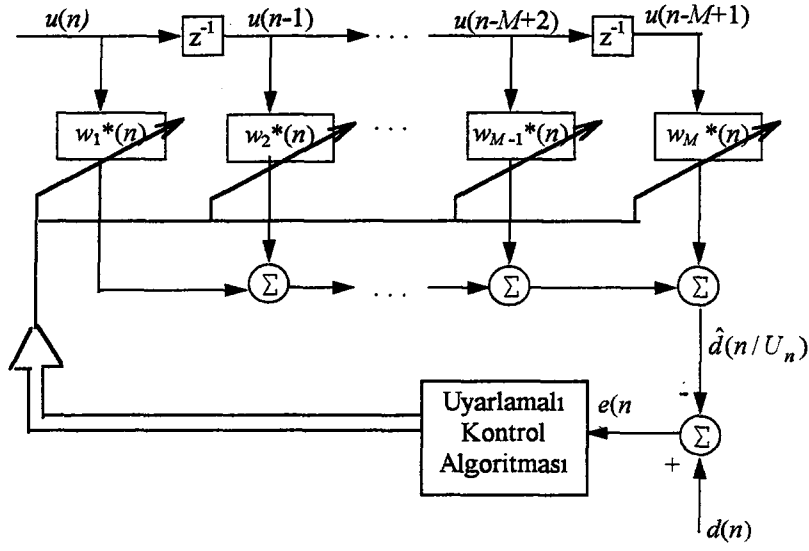
Tap ağırlık vektörünün genişletilmiş biçimi,

$$\mathbf{w}^T(n) = [w_1(n), w_2(n), \dots, w_M(n)] \quad (4.2)$$

şeklinde tanımlanır. Giriş vektörünün genişletilmiş biçimi de,

$$\mathbf{u}^T(n) = [u(n), u(n-1), \dots, u(n-M+1)] \quad (4.3)$$

şeklinde dir.



Şekil 4.1 Uyarlamalı Enine Süzgeç Yapısı

Eğer giriş vektörü $u(n)$ ile istenen yanıt $d(n)$ birleşik durağan ise, n anındaki karesel ortalama hata, daha önce elde edilen (2.50)' deki ifadeye benzer biçimde

$$J(n) = \sigma_d^2 - \mathbf{w}^H(n)\mathbf{p} - \mathbf{p}^H\mathbf{w}(n) + \mathbf{w}^H(n)\mathbf{R}\mathbf{w}(n) \quad (4.4)$$

yeniden yazılabilir.

Böylece karesel ortalama hata $J(n)$ ' nin, tap ağırlık vektörü $\mathbf{w}(n)$ ' nin elemanlarına bağımlılığı, tek minimumlu kase biçimli bir yüzey gibi düşünülebilir. Bu yüzey, uyarlamalı süzgecin hata başarımlı yüzeyi olarak kabul edilir. Uyarlama süreci, bu yüzeyin en altını veya minimum noktasını devamlı olarak arama işlemi yapar. Hata başarımlı yüzeyinin minimum noktasında tap ağırlık vektörü,

$$\mathbf{R}\mathbf{w}_o = \mathbf{p} \quad (4.5)$$

normal eřitlięi ile tanımlı w_o optimum deęerini alır.

Minimum karesel ortalama hata ise (2.49) eřitlięinde ifade edildięi gibi,

$$J_{min} = \sigma_d^2 - p^H w_o \quad (4.6)$$

bięimindedir.

Duraęan bir ortamda uyarlamalı bir enine süzgecin ęalıřması iin hata bařarım yzeyi, ynlendirme yapılabilecek kadar gzel sabit bir řekilde olmalıdır. O zaman problem, uyarlamalı szgecin bu yzeyin minimum noktasında veya ona yakın bir noktada ęalıřabilecek bięimde tasarlanması olarak basitleřir. Bununla birlikte uyarlamalı szge duraęan olmayan bir ortamda ęalıřtırılırsa, hata bařarım yzeyinin eęrilięi ve ynlendirmesi deęiřirken yzeyin enaltı da devamlı olarak hareket eder. Bu yzden giriřler duraęan olmadıęı zaman, uyarlamalı szge sadece yzeyin enaltını arama iřlemini deęil, aynı zamanda devamlı olarak onu izleme iřlemini de stlenir.

BÖLÜM 5. EN DİK İNİŞ YÖNTEMİ

Uyarlamalı süzgeçlerde iki temel süreç yer alır:

1. Uyarlamalı süreç: Uyarlamalı süzgecin tap ağırlıklarının bir algoritma yardımıyla otomatik olarak ayarlanmasını içerir.
2. Süzme süreci: Bu süreç, i) istenen yanıtın kestirimini üretmek için uyarlamalı süreçten elde edilen uygun tap ağırlıkları ile girişlerin çarpımını ve ii) istenen yanıtın gerçek değeri ile kestirimin karşılaştırılması sonucu elde edilen kestirim hatasını üretmeyi içerir. Bu kestirim hatası uyarlamalı sürecin işleminde kullanılır.

Uyarlamalı sürecin gerçekleştirilmesi için çeşitli algoritmalar geliştirilmiştir. Bu bölümde, eski bir optimizasyon tekniği olan ve hata başarımlı yüzeyinin kendisini bilmeksizin onun minimum noktasını bulan yinlemeli bir yöntem olan En Dik İniş Yönteminden bahsedilecektir. Çünkü bu yöntem, diğer algoritmaların anlaşılmasında kolaylık sağlayacaktır.

5.1. En Dik İniş Algoritması

Bir enine süzgecin gerçekleştirilmesi için, (4.5) normal eşitliğini gerçekleyen tap ağırlıkları vektörlerinin bir çözümünün bulunması gereklidir. Bunu yapmanın bir yolu, bu eşitliği analitik yollarla çözmektir. Bu işlem oldukça açık görünmesine rağmen, süzgeç çok sayıda tap ağırlığı içerdiğinde ve giriş veri hızı çok yüksek olduğu zaman ciddi hesaplama zorlukları ile karşılaşmaktadır.

Alternatif bir uygulama, en eski optimizasyon yöntemlerinden olan en dik iniş yönteminin kullanılmasıdır. Bu yöntemle karesel ortalama hatanın minimum değeri $J_{\min}$ 'i bulmak için şu adımlar uygulanır:

1. Keyfi olarak seçilen, tap ağırlıkları vektörü başlangıç değeri $\mathbf{w}(0)$ ile başlanacaktır. Bu $\mathbf{w}(0)$ değeri, hata başarımlı yüzeyinin minimum noktasının nerede olabileceği gibi bir başlangıç kestirimi yapar. Genellikle $\mathbf{w}(0)$, sıfır vektörüne eşit seçilir.
2. Bu başlangıç değeri veya şimdiki kestirim kullanılarak gradyen vektör hesaplanır. Bu vektör, karesel ortalama hata $J(n)$ ' nin n anındaki tap ağırlık vektörü $\mathbf{w}(n)$ ' e göre türevi olarak tanımlanır.
3. Tap ağırlık vektörünün bir sonraki kestirimi hesaplanır.
4. 2. adıma geri dönülerek işlem tekrarlanır.

Tap ağırlıkları vektörü, gradyen vektörünün negatif yönünde ardarda düzeltilmesi sonucu optimum değer olan $\mathbf{w}_o$ değerine ulaştığında, karesel ortalama hatanın da minimum değeri J_{min} ' e yakınsayacağı düşünülür.

$\nabla(n)$, gradyen vektörünün n anındaki değerini, $\mathbf{w}(n)$ ise tap ağırlıkları vektörünün n anındaki değerini gösterebilir. Tap ağırlıkları vektörünün $n+1$ anındaki güncellenmiş değeri,

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{1}{2} \mu [-\nabla(n)] \quad (5.1)$$

biçiminde özyineli bir ilişki kullanılarak hesaplanabilir. Burada μ , pozitif gerçel değerli bir sabittir.

Gradyen vektör için, (4.4) eşitliğinde verilen karesel ortalama hata $J(n)$ ' nin tap ağırlık vektörü $\mathbf{w}(n)$ ' e göre kısmi türevi alınırsa

$$\begin{aligned} \nabla(n) &= \frac{\partial J(n)}{\partial \mathbf{w}(n)} \\ &= -2\mathbf{p} + 2\mathbf{R}\mathbf{w}(n) \end{aligned} \quad (5.2)$$

elde edilir.

En dik iniş yöntemi uygulamalarında, verilen bir tap ağırlık vektörü $\mathbf{w}(n)$ için gradyen vektör $\nabla(n)$ hesaplanabilmesi için, ilişki matrisi $\mathbf{R}$ ve çapraz ilişki vektörü $\mathbf{p}$ ' nin

bilindiği varsayılır. (5.2) eşitliği (5.1)'de yerine konulursa, tap ağırlıkları vektörünün güncellenmiş değeri,

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu[p - \mathbf{R}\mathbf{w}(n)], \quad n = 0, 1, 2, \dots \quad (5.3)$$

biçiminde elde edilir. Bu eşitlik en dik iniş yönteminin matematiksel ifadesini göstermektedir. Burada, μ parametresi, bir iterasyon çevriminden diğerine geçerken tap ağırlıklarına uygulanacak artımsal düzeltmenin miktarını kontrol eder. Bu yüzden μ , adım uzunluğu parametresi olarak bilinir.

5.2. En Dik İniş Algoritmasının Kararlılığı

n anındaki ağırlık hata vektörü $\mathbf{c}(n)$,

$$\mathbf{c}(n) = \mathbf{w}(n) - \mathbf{w}_o \quad (5.4)$$

biçiminde tanımlansın. Burada $\mathbf{w}_o$, Wiener-Hopf eşitliği (4.5) ile tanımlanan optimum tap ağırlık vektörüdür. Benzer şekilde,

$$\mathbf{c}(n+1) = \mathbf{w}(n+1) - \mathbf{w}_o \quad (5.5)$$

eşitliği yazılabilir. (4.5) ve (5.3) eşitlikleri kullanılarak,

$$\mathbf{c}(n+1) = (\mathbf{I} - \mu\mathbf{R})\mathbf{c}(n) \quad (5.6)$$

ifadesi yazılabilir. $\mathbf{R}$ ilişki matrisinin aşağıdaki ifadesi (bakınız eşitlik (2.52)) göz önüne alınsın:

$$\mathbf{R} = \mathbf{Q} \mathbf{A} \mathbf{Q}^H \quad (5.7)$$

Bu eşitlik (5.6) eşitliğinde kullanılarak,

$$\mathbf{c}(n+1) = (\mathbf{I} - \mu \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^H) \mathbf{c}(n) \quad (5.8)$$

yazılabilir. Eşitliğin her iki tarafı $\mathbf{Q}^H$ ile öncül çarpılır ve $\mathbf{Q}^H$ değerinin $\mathbf{Q}^{-1}$ e eşit olması özelliği kullanılırsa,

$$\mathbf{Q}^H \mathbf{c}(n+1) = (\mathbf{I} - \mu \mathbf{\Lambda}) \mathbf{Q}^H \mathbf{c}(n) \quad (5.9)$$

eşitliği elde edilir.

Aşağıdaki gibi yeni bir koordinat grubu tanımlansın:

$$\begin{aligned} \mathbf{v}(n) &= \mathbf{Q}^H \mathbf{c}(n) \\ &= \mathbf{Q}^H [\mathbf{w}(n) - \mathbf{w}_o] \end{aligned} \quad (5.10)$$

Bu durumda (5.9) eşitliği kullanılarak,

$$\mathbf{v}(n+1) = (\mathbf{I} - \mu \mathbf{\Lambda}) \mathbf{v}(n) \quad (5.11)$$

yazılabilir. $\mathbf{v}(n)$ ' nin başlangıç değeri,

$$\mathbf{v}(0) = \mathbf{Q}^H [\mathbf{w}(0) - \mathbf{w}_o] \quad (5.12)$$

şeklinde dir. Tap ağırlık vektörünün başlangıç değeri sıfır olarak kabul edilirse (5.12) eşitliği,

$$\mathbf{v}(0) = - \mathbf{Q}^H \mathbf{w}_o \quad (5.13)$$

biçimine dönüşür. En dik iniş algoritması k . doğal mod için (Widrow, 1970),

$$v_k(n+1) = (1 - \mu \lambda_k) v_k(n), \quad k = 1, 2, \dots, M \quad (5.14)$$

şeklindedir ve burada λ_k , $\mathbf{R}$ ilişki matrisinin k . özdeğeridir. (5.14) eşitliği 1. dereceden homojen bir fark denklemdir. $v_k(n)$ ' nin ilk değerinin $v_k(0)$ olduğu göz önüne alınarak, bu eşitliğin çözümü olarak,

$$v_k(n) = (1 - \mu \lambda_k)^n v_k(0), \quad k = 1, 2, \dots, M \quad (5.15)$$

eşitliği yazılabilir. n sonsuza giderken $v_k(n)$ değerinin, her k değeri için, sıfır olması için,

$$-1 < 1 - \mu \lambda_k < 1$$

eşitliği sağlanmalıdır. İlişki matrisi $\mathbf{R}$ pozitif ve gerçel olduğundan, en dik iniş algoritmasının kararlılığı ya da yakınsaması için gerek ve yeter koşul, adım uzunluğu parametresi μ ' nün

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (5.16)$$

ifadesini sağlamasıdır. Burada $\lambda_{\max}$, $\mathbf{R}$ ilişki matrisinin en büyük özdeğeridir.

BÖLÜM 6. EN KÜÇÜK KARESEL ORTALAMA (LMS: LEAST MEAN SQUARE) ALGORİTMASI KULLANILARAK GERÇEKLEŞTİRİLEN UYARLAMALI ENİNE SÜZGEÇLER

Bu bölümde, çok yaygın olarak kullanılan ve en küçük karesel ortalama (LMS) algoritması olarak bilinen algoritma incelenecektir. LMS algoritmasının en büyük özelliği basit olması, yani ilgili ilişki fonksiyonlarının ölçümlerini ve matris tersi almayı gerektirmemesidir.

Burada, LMS algoritmasının durağan ortamlardaki çalışması incelenecektir. Bu inceleme, sadece LMS algoritmasının işleyişinin anlaşılmasına değil, aynı zamanda, daha karmaşık uyarlamalı süzgeç algoritmalarının anlaşılmasına da yardımcı olacaktır.

6.1. En Küçük Karesel Ortalama (LMS) Uyarlama Algoritması

Eğer her iterasyonda gradyen vektör $\nabla(J(n))$ tam olarak ölçülebilseydi ve adım uzunluğu parametresi μ uygun seçilseydi, en dik iniş yöntemi kullanılarak hesaplanan tap ağırlık vektörü gerçekten optimum Wiener çözümüne yakınsayacaktı. Ancak gerçekte gradyen vektörünü tam olarak ölçmek mümkün değildir ve gradyen vektör elde edilen veriden kestirmek zorundadır. Diğer bir deyişle tap ağırlık vektörü, gelen veriye uyarlanan uygun bir algoritma ile güncelleştirilir. En Küçük Karesel Ortalama (LMS) algoritması böyle bir algoritmadır.

Gradyen vektör $\nabla(J(n))$ ' nin kestirimini geliştirmek için en açık yöntem, ilişki matrisi $\mathbf{R}$ ' nin ve çapraz ilişki vektörü $\mathbf{p}$ ' nin kestirimlerinin,

$$\nabla(J(n)) = -2\mathbf{p} + 2\mathbf{R}\mathbf{w}(n) \quad (6.1)$$

ifadesinde kullanılmasıdır. $\mathbf{R}$ ve $\mathbf{p}$ için kestirim değerlerinin en basit seçimi, tap giriş vektörü ve istenen yanıtın örnek değerlerine dayalı olarak

$$\hat{\mathbf{R}}(n) = \mathbf{u}(n)\mathbf{u}^H(n) \quad (6.2)$$

$$\hat{\mathbf{p}}(n) = \mathbf{u}(n)d^*(n) \quad (6.3)$$

biçiminde ani kestirimler kullanılmaktadır. Bu durumda gradyen vektörünün ani kestirimi,

$$\hat{\nabla}(J(n)) = -2\mathbf{u}(n)d^*(n) + 2\mathbf{u}(n)\mathbf{u}^H(n)\hat{\mathbf{w}}(n) \quad (6.4)$$

biçiminde yazılabilir.

(5.1) eşitliğinde tanımlandığı gibi, en dik iniş algoritmasında gradyen vektörün kestirimi olan (6.4) eşitliği kullanılırsa, tap ağırlık vektörünün güncellenmesi için,

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \mu\mathbf{u}(n)[d^*(n) - \mathbf{u}^H(n)\hat{\mathbf{w}}(n)] \quad (6.5)$$

biçiminde yeni bir özyineli ilişki elde edilir. Eşdeğer olarak,

$$e(n) = d(n) - \hat{\mathbf{w}}^H(n)\mathbf{u}(n) \quad (6.6)$$

olduğundan

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \mu\mathbf{u}(n)e^*(n) \quad (6.7)$$

yazılabilir. (6.6) eşitliği, tap ağırlık vektörü $\mathbf{w}(n)$ ' nin şimdiki kestirimine dayalı olarak kestirim hatası $e(n)$ ' i tanımlar. (6.7) eşitliğinin sağ tarafındaki ikinci terim $\mu\mathbf{u}(n)e^*(n)$, $\mathbf{w}(n)$ ' nin şimdiki kestirimine uygulanan düzeltmeyi gösterir. İteratif işlem, başlangıç kestirimi $\hat{\mathbf{w}}(0)$ ile başlatılır. Bu başlangıç kestirimi için alışlagelmiş seçim sıfır vektörüdür, yani $\hat{\mathbf{w}}(0) = \mathbf{0}$ alınır.

(6.5) eşitliğiyle veya eşdeğer olarak (6.6) ve (6.7) eşitlikleri ile tanımlanan algoritma, uyarlamalı En Küçük Karesel Ortalama (LMS) algoritmasının karmaşık biçimidir.

$\mathbf{R}$ ve $\mathbf{p}$ ani kestirimleri, göreceli olarak büyük değişimlere sahiptir. Bu yüzden ilk bakışta LMS algoritması ani kestirimler kullanıldığından iyi bir başarıya sahip değilmiş gibi görünebilir. Ancak, LMS algoritmasının yapısı itibarıyla özyineli olduğu unutulmamalıdır.

6.2. LMS Algoritmasının Kararlılık Analizi

LMS algoritmasının kararlılığı için adım uzunluğu parametresi μ ' nün aşağıdaki iki biçimde yakınsamayı sağlayacak şekilde seçilmesi gerekmektedir.

1. İterasyon sayısı n sonsuza yaklaşırken tap ağırlık vektörünün kestirimi $\hat{\mathbf{w}}(n)$ ' nin optimum Wiener çözümü $\mathbf{w}_o$ ' a yakınsaması olarak düşünülen ortalamanın yakınsaması.
2. Karesel ortalama hatanın son değeri (kararlı durum) $J(\infty)$ ' nin sonlu bir değere ulaşması olarak düşünülen karesel ortalama anlamda yakınsama.

LMS algoritmasının matematiksel olarak kolayca işlenebilir istatistiksel analizini yapmak için, aşağıdaki dört bölümü içeren temel varsayımlar kullanılacaktır:

1. $\mathbf{u}(1), \mathbf{u}(2), \dots, \mathbf{u}(n)$ tap giriş vektörleri istatistiksel olarak birbirinden bağımsız vektörler dizisinden oluşur:

$$E[\mathbf{u}(n)\mathbf{u}^H(k)] = 0, \quad k = 0, 1, \dots, n-1 \quad (6.8)$$

2. n anındaki tap giriş vektörü $\mathbf{u}(n)$, daha önceki tüm istenen yanıt örneklerinden, yani $d(1), d(2), \dots, d(n-1)$ değerlerinden bağımsızdır:

$$E[\mathbf{u}(n)d^*(k)] = 0, \quad k = 0, 1, \dots, n-1 \quad (6.9)$$

3. n anındaki istenen yanıt $d(n)$, tap giriş vektörü $u(n)$ ' e bağımlıdır ancak, istenen yanıtın önceki tüm örneklerinden istatistiksel bağımsızdır.

4. Tap giriş vektörü $u(n)$ ve istenen yanıt $d(n)$, her n için karşılıklı Gauss dağılımlı değişkenler içerir.

Bu temel varsayımlara dayalı LMS algoritmasının istatistiksel analizi, Bağımsızlık Teorisi diye adlandırılır.

(6.5) eşitliğinde, $n+1$ anında sadece şu üç girişe bağlı tap ağırlık vektörü $\hat{w}(n+1)$ gözlemlenir:

1. Giriş işlevinin önceki örnek değerleri, $u(n)$, $u(n-1)$, ..., $u(0)$.
2. İstenen yanıtın önceki örnek değerleri, $d(n)$, $d(n-1)$, ..., $d(0)$.
3. Tap ağırlık vektörünün başlangıç değeri, $\hat{w}(0)$.

Varsayım 1 ve 2' ye göre $\hat{w}(n+1)$ tap ağırlık vektörü, $u(n+1)$ ve $d(n+1)$ ' den bağımsızdır. Gelecek iki bölümde, tap ağırlık vektörünün kestiriminin ilk iki anını çıkarmak için temel varsayımlar kullanılacaktır: (i) ortalama değer ve (ii) ilişki matrisi. Tap ağırlık vektörünün kendisinden çok,

$$\varepsilon(n) = \hat{w}(n) - w_o \quad (6.10)$$

biçimindeki ağırlık hatası vektörü ile çalışmak daha uygun olacaktır

6.3. Ortalama Tap Ağırlık Vektörünün Davranışı

(6.5) eşitliğinin her iki yanından w_o çıkarılır ve bu eşitliğin sağındaki düzeltme teriminden $\hat{w}(n)$ ' i yok etmek için (6.10) eşitliği kullanılırsa, LMS algoritması ağırlık hatası vektörü cinsinden,

$$\varepsilon(n+1) = [\mathbf{I} - \mu \mathbf{u}(n) \mathbf{u}^H(n)] \varepsilon(n) + \mu [\mathbf{u}(n) d^*(n) - \mathbf{u}(n) \mathbf{u}^H(n) \mathbf{w}_o] \quad (6.11)$$

biçiminde yazılır. Burada, $\mathbf{I}$ birim matristir. Temel varsayımların sonucu olarak, tap ağırlık vektörü $\hat{\mathbf{w}}(n)$, giriş vektörü $\mathbf{u}(n)$ ' den bağımsızdır. Aynı zamanda, ağırlık hatası vektörü $\varepsilon(n)$ de $\mathbf{u}(n)$ ' den bağımsızdır. (6.11) eşitliğinin her iki tarafının matematiksel beklendik değerleri alınıp, $\varepsilon(n)$ ' nin $\mathbf{u}(n)$ ile bağımsızlığı kullanılır ve ilişki matrisi $\mathbf{R} = E[\mathbf{u}(n) \mathbf{u}^H(n)]$ ve çapraz ilişki vektörü $\mathbf{p} = E[\mathbf{u}(n) d^*(n)]$ yerine konursa,

$$E[\varepsilon(n+1)] = (\mathbf{I} - \mu \mathbf{R}) E[\varepsilon(n)] + \mu (\mathbf{p} - \mathbf{R} \mathbf{w}_o) \quad (6.12)$$

elde edilir. Normal eşitlik $\mathbf{R} \mathbf{w}_o = \mathbf{p}$ idi. Bu durumda, (6.12) eşitliği

$$E[\varepsilon(n+1)] = (\mathbf{I} - \mu \mathbf{R}) E[\varepsilon(n)] \quad (6.13)$$

biçimine dönüşür. (6.13) eşitliği en dik iniş algoritması için geçerli olan,

$$\mathbf{c}(n+1) = (\mathbf{I} - \mu \mathbf{R}) \mathbf{c}(n)$$

eşitliği ile karşılaştırılırsa, matematiksel olarak tamamen aynı olduğu görülür. LMS algoritmasına ait olan $E[\varepsilon(n)]$ ile en dik iniş algoritmasında tanımlı olan $\mathbf{c}(n)$ aynı role sahiptirler. En dik iniş algoritmasında,

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (6.14)$$

ifadesine uygun olarak $\mathbf{c}(n)$, n sonsuza giderken sıfıra yakınsamaktaydı. Yukardaki benzeşimden dolayı LMS algoritması için de (6.14) ifadesi uygulanırsa $\varepsilon(n)$, n sonsuza giderken sıfıra yakınsayacaktır.

Adım uzunluğu parametresi μ , (6.14) ifadesiyle tanımlanan sınırlar içinde tutulursa, LMS algoritması ile hesaplanan tap ağırlık vektörü $\hat{\mathbf{w}}(n)$, iterasyon sayısı n sonsuza yaklaştığı zaman optimum Wiener çözüm $\mathbf{w}_o$ ' a yakınsar. Bu koşullar altında LMS algoritması ortalama olarak yakınsamıştır denir.

6.4. Ağırlık Hatasının İlişki Matrisi

Ağırlık hatası vektörü $\boldsymbol{\varepsilon}(n)$ ' nin ilişki matrisinin zamanla değişimini belirten bir özyineli eşitlik geliştirmek için, n anında bu ilişki matrisini belirten,

$$\mathbf{K}(n) = E[\boldsymbol{\varepsilon}(n)\boldsymbol{\varepsilon}^H(n)] \quad (6.15)$$

ifadesi kullanılmaktadır. Özyineli eşitliği elde etmek için (6.15) eşitliği kullanılacak ve bu eşitlikte $\boldsymbol{\varepsilon}(n+1)$ için (6.11) eşitliği yerine konulacak ve ayrıca temel varsayımlar kullanılarak toplamsal ortalama alınacaktır. Tüm bu işlemler yapılırsa (Haykin, 1991), sonuçta ağırlık hatası ilişki matrisinin zamanla değişimi,

$$\mathbf{K}(n+1) = \mathbf{K}(n) - \mu[\mathbf{R}\mathbf{K}(n) + \mathbf{K}(n)\mathbf{R}] + \mu^2 \mathbf{R} \text{tr}[\mathbf{R}\mathbf{K}(n)] + \mu^2 \mathbf{R}\mathbf{K}(n)\mathbf{R} + \mu^2 J_{\min} \mathbf{R} \quad (6.16)$$

biçiminde elde edilir. Sağ taraftaki son terim $\mu^2 J_{\min} \mathbf{R}$, eşitliğin tek bir çözümünün olması yani $\mathbf{K}(n) = 0$ olmasını engeller. Özyineli bir ilişki olan (6.16) eşitliği ile $\mathbf{K}(n)$ ilişki matrisinin pozitif belirli olması karakteri korunur. Bundan dolayı $\mathbf{K}(0)$ da pozitif belirlidir.

$$\begin{aligned} \mathbf{K}(0) &= \boldsymbol{\varepsilon}(0)\boldsymbol{\varepsilon}^H(0) \\ &= [\hat{\mathbf{w}}(0) - \mathbf{w}_o] [\hat{\mathbf{w}}^H(0) - \mathbf{w}_o^H] \end{aligned} \quad (6.17)$$

Sonuç olarak (6.16) eşitliği, ağırlık hatası ilişki matrisi $\mathbf{K}(n)$ ' i, $n = 0$ için $\mathbf{K}(0)$ ' dan başlayarak özyineli olarak, güncellemek için bir ilişki sağlar. Üstelik bu ilişki her

iterasyondan sonra ağırlık hatası ilişki matrisinin güncellenmiş değeri için pozitif belirli bir yanıt verir.

6.5. Karesel Ortalama Hata Davranışı

(6.16) eşitliği, Kısım 6.2' deki temel varsayımlara dayanan LMS algoritması karesel ortalama hatasının geçici hal davranışının hesaplanması için yararlı bir araçtır. LMS algoritması ile kestirim hatası $e(n)$ ifadesini elde etmek için (6.6) ve (6.10) eşitliklerinden yararlanılarak,

$$\begin{aligned} e(n) &= d(n) - \hat{\mathbf{w}}^H(n)\mathbf{u}(n) \\ &= d(n) - \mathbf{w}_o^H \mathbf{u}(n) - \boldsymbol{\varepsilon}^H(n)\mathbf{u}(n) \\ &= e_o(n) - \boldsymbol{\varepsilon}^H(n)\mathbf{u}(n) \end{aligned} \quad (6.18)$$

elde edilir. Burada $e_o(n)$, optimum Wiener çözümü için kestirim hatasıdır.

Karesel ortalama hata $J(n)$, Kısım 6.2' deki temel varsayımlar da kullanılarak,

$$\begin{aligned} J(n) &= E[|e(n)|^2] \\ &= E[(e_o(n) - \boldsymbol{\varepsilon}^H(n)\mathbf{u}(n))(e_o(n) - \mathbf{u}^H(n)\boldsymbol{\varepsilon}(n))] \\ &= J_{min} + E[\boldsymbol{\varepsilon}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n)] \end{aligned} \quad (6.19)$$

şeklinde ifade edilir. Burada J_{min} , optimum Wiener süzgeç ile üretilen minimum karesel ortalama hatadır. Eşitliğin sağ tarafındaki ikinci terim için aşağıdaki eşitlikler yazılabilir:

$$\begin{aligned} E[\boldsymbol{\varepsilon}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n)] &= E[\text{tr}\{\boldsymbol{\varepsilon}^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n)\}] \\ &= E[\text{tr}\{\mathbf{u}(n)\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n)\boldsymbol{\varepsilon}^H(n)\}] \\ &= \text{tr}\{E[\mathbf{u}(n)\mathbf{u}^H(n)]E[\boldsymbol{\varepsilon}(n)\boldsymbol{\varepsilon}^H(n)]\} \\ &= \text{tr}[\mathbf{R}\mathbf{K}(n)] \end{aligned} \quad (6.20)$$

Buna göre (6.19) eşitliği yeniden yazılırsa, karesel ortalama hata,

$$J(n) = J_{min} + tr[\mathbf{R}\mathbf{K}(n)] \quad (6.21)$$

biçiminde elde edilir.

LMS algoritmasında, kestirim hatasının karesel ortalama değeri iki bileşenden oluşur: Minimum karesel ortalama hata J_{min} ve ağırlık hatası ilişki matrisi $\mathbf{K}(n)$ ' nin geçici hal davranışına bağlı bir bileşen. Mademki her n değeri için sonraki eleman pozitif belirlidir, LMS algoritması daima minimum karesel ortalama hata J_{min} ' den daha büyük bir karesel ortalama hata üretir.

Uyarlamalı algoritmanın n anında ürettiği karesel ortalama hata $J(n)$ ile optimum Wiener çözüm ile elde edilen minimum değer J_{min} arasındaki fark artık karesel ortalama hata olarak tanımlanır ve

$$\begin{aligned} J_{ex}(n) &= J(n) - J_{min} \\ &= tr[\mathbf{R}\mathbf{K}(n)] \end{aligned} \quad (6.22)$$

ile gösterilir. $\mathbf{K}(n)$ ' i hesaplamak için, (6.16) eşitliğinin özyineli ilişkisi kullanılacaktır. Bununla birlikte Birim Benzerlik Dönüşümünü kullanmak faydalı olacaktır []:

$$\mathbf{Q}^H \mathbf{R} \mathbf{Q} = \Lambda \quad (6.23)$$

Λ ilişki matrisi, $\mathbf{R}$ ' nin özdeğerlerini içeren bir köşegen matris, $\mathbf{Q}$ ise bu özdeğerlere ilişkin özvektörleri içeren matristir. Ayrıca,

$$\mathbf{Q}^H \mathbf{K}(n) \mathbf{Q} = \mathbf{X}(n) \quad (6.24)$$

olsun. Genelde $\mathbf{X}(n)$ köşegen matris değildir. (6.23) ve (6.24) eşitlikleri (6.22)' de yerine konulursa,

$$\begin{aligned} \text{tr}[\mathbf{R}\mathbf{K}(n)] &= \text{tr}[\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^H\mathbf{Q}\mathbf{X}(n)\mathbf{Q}^H] \\ &= \text{tr}[\mathbf{Q}\mathbf{\Lambda}\mathbf{X}(n)\mathbf{Q}^H] \end{aligned} \quad (6.25)$$

elde edilir. Burada $\mathbf{Q}^H\mathbf{Q} = \mathbf{I}$ özelliği kullanılmıştır. Buna göre,

$$\begin{aligned} \text{tr}[\mathbf{R}\mathbf{K}(n)] &= \text{tr}[\mathbf{Q}\mathbf{Q}^H\mathbf{\Lambda}\mathbf{X}(n)] \\ &= \text{tr}[\mathbf{\Lambda}\mathbf{X}(n)] \end{aligned} \quad (6.26)$$

yazılabilir. Daha uygun bir biçimde,

$$J_{\text{ex}}(n) = \text{tr}[\mathbf{\Lambda}\mathbf{X}(n)] \quad (6.27)$$

$$J_{\text{ex}}(n) = \sum_{i=1}^M \lambda_i x_i(n) \quad (6.28)$$

yazılabilir. Burada $x_i(n)$, $i = 1, 2, \dots, M$ olmak üzere, $\mathbf{X}(n)$ matrisinin köşegen elemanları ve λ_i ilişki matrisi $\mathbf{R}'$ nin özdeğerleridir. Daha sonra (6.23) ve (6.24) eşitliklerinde tanımlanan dönüşümler kullanılırsa, (6.16) özyineli eşitliği $\mathbf{X}(n)$ ve $\mathbf{\Lambda}$ cinsinden

$$\mathbf{X}(n+1) = \mathbf{X}(n) - \mu[\mathbf{\Lambda}\mathbf{X}(n) + \mathbf{X}(n)\mathbf{\Lambda}] + \mu^2 \mathbf{\Lambda} \text{tr}[\mathbf{\Lambda}\mathbf{X}(n)] + \mu^2 \mathbf{\Lambda}\mathbf{X}(n)\mathbf{\Lambda} + \mu^2 J_{\min} \mathbf{\Lambda} \quad (6.29)$$

biçiminde yazılabilir.

$J_{\text{ex}}(n)$ ' nin sadece $x_i(n)$ ' e bağlı olduğu (6.28) eşitliğinden görülür. Bu, (6.29) özyineli eşitliğin köşegen elemanlarına bakmayı önerir:

$$x_i(n+1) = x_i(n) - 2\mu\lambda_i x_i(n) + \mu^2 \lambda_i \sum_{j=1}^M \lambda_j x_j(n) + \mu^2 \lambda_i^2 x_i(n) + \mu^2 J_{\min} \lambda_i \quad i = 1, 2, \dots, M \quad (6.30)$$

$M \times 1$ boyutlu $\mathbf{x}(n)$ ve $\boldsymbol{\lambda}$ vektörleri,

$$\mathbf{x}(n) = [x_1(n), x_2(n), \dots, x_M(n)]^T \quad (6.31)$$

$$\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \dots, \lambda_M]^T \quad (6.32)$$

biçimindedir.

(6.30) eşitliği matris biçiminde

$$\mathbf{x}(n+1) = \mathbf{B}\mathbf{x}(n) + \mu^2 J_{min} \boldsymbol{\lambda} \quad (6.33)$$

olarak yazılabilir. $\mathbf{B}$ matrisi elemanları aşağıdaki gibi tanımlı olan $M \times M$ boyutlu bir matristir:

$$b_{ij} = \begin{cases} (1 - \mu\lambda_i)^2 + \mu^2 \lambda_i^2, & i = j \\ \mu^2 \lambda_i \lambda_j, & i \neq j \end{cases} \quad (6.34)$$

(6.34) eşitliğinden, $\mathbf{B}$ matrisinin simetrik, elemanları b_{ij} ' lerin tümünün pozitif ve gerçel değerli olduğu görülür.

(6.33) eşitliği, matris biçiminde 1. dereceden fark denklemdir. $\mathbf{x}(0)$ başlangıç değeri olarak kabul edilirse bu eşitliğin çözümü,

$$\mathbf{x}(n) = \mathbf{B}^n \mathbf{x}(0) + \mu^2 J_{min} \sum_{i=0}^{n-1} \mathbf{B}^i \boldsymbol{\lambda} \quad (6.35)$$

şekline dönüşür. Geometrik serilerin toplamı için kullanılan formüle benzetilerek, sonlu toplam $\sum_{i=0}^{n-1} \mathbf{B}^i$ şöyle ifade edilebilir:

$$\sum_{i=0}^{n-1} \mathbf{B}^i = (\mathbf{I} - \mathbf{B}^n)(\mathbf{I} - \mathbf{B})^{-1} \quad (6.36)$$

(6.36) eşitliği (6.35)'de yerine konulursa,

$$\mathbf{x}(n) = \mathbf{B}^n [\mathbf{x}(0) - \mu^2 J_{min} (\mathbf{I} - \mathbf{B})^{-1} \lambda] + \mu^2 J_{min} (\mathbf{I} - \mathbf{B})^{-1} \lambda \quad (6.37)$$

elde edilir. (6.37) eşitliğinde sağ taraftaki ilk terim, $\mathbf{x}(n)$ vektörünün geçici hal elemanı, ikinci terim kalıcı hal elemanıdır. $\mathbf{B}$ matrisi simetrik olduğundan $\mathbf{B}$ matrisine, benzer olarak (6.23) eşitliği ile tanımlanan Birim Benzerlik Dönüşümü uygulanırsa,

$$\mathbf{G}^T \mathbf{B} \mathbf{G} = \mathbf{C} \quad (6.38)$$

elde edilir. $\mathbf{C}$ matrisi köşegen matristir ve elemanları $\mathbf{B}'$ nin özdeğerleri olan c_i ' lerdir. ($i = 1, 2, \dots, M$) $\mathbf{G}$ matrisi birim matristir ve dolayısıyla

$$\mathbf{G}^T \mathbf{G} = \mathbf{I} \quad (6.39)$$

yazılabilir. $\mathbf{G}$ matrisinin i . kolonu c_i özdeğerlerine ilişkin, $\mathbf{B}'$ nin $\mathbf{g}_i$ özvektörüdür. (6.39) eşitliğindeki özellik nedeniyle,

$$\mathbf{B}^n = \mathbf{G} \mathbf{C}^n \mathbf{G}^T \quad (6.40)$$

yazılabilir. Böylece (6.37) eşitliği şu biçimde yazılabilir:

$$\mathbf{x}(n) = \mathbf{G} \mathbf{C}^n \mathbf{G}^T [\mathbf{x}(0) - \mu^2 J_{min} (\mathbf{I} - \mathbf{B})^{-1} \lambda] + \mu^2 J_{min} (\mathbf{I} - \mathbf{B})^{-1} \lambda \quad (6.41)$$

$\mathbf{C}$ köşegen matris olduğu için,

$$\mathbf{C}^n = \text{diag} (c_1^n, c_2^n, \dots, c_M^n) \quad (6.42)$$

yazılabilir. (6.41) eşitliğinin ve buna bağlı olarak (6.37) eşitliğinin çözümleri, aşağıda gösterildiği gibi sadece $\mathbf{B}$ matrisinin tüm özdeğerleri 1' den küçük bir büyüklüğe sahip ise kararlıdır:

$$-1 < c_i < 1, \quad \text{tüm } i\text{'ler için} \quad (6.43)$$

Bu koşul yerine getirildiği zaman, (6.41) veya (6.37) eşitliğindeki geçici hal elemanı, iterasyon sayısı n sonsuza yaklaşırken sıfıra gider ve sadece eleman olarak kalıcı hal elemanı kalır.

$$\mathbf{x}(\infty) = \mu^2 J_{min} (\mathbf{I} - \mathbf{B})^{-1} \lambda \quad (6.44)$$

(6.44) eşitliği (6.41) eşitliğinde yerine konursa çözüm,

$$\mathbf{x}(n) = \mathbf{G}\mathbf{C}^n\mathbf{G}^T [\mathbf{x}(0) - \mathbf{x}(\infty)] + \mathbf{x}(\infty) \quad (6.45)$$

şeklinde bulunur. $\mathbf{C}^n$ matrisi köşegen matris olduğundan ve $\mathbf{G}$ birim matrisi sütunları $\mathbf{B}$ 'nin özvektörlerini içerdiğinden, $\mathbf{G}\mathbf{C}^n\mathbf{G}^T$ matris çarpımı,

$$\mathbf{G}\mathbf{C}^n\mathbf{G}^T = \sum_{i=1}^M c_i^n \mathbf{g}_i \mathbf{g}_i^T \quad (6.46)$$

biçiminde yazılabilir. (6.46) eşitliği (6.45) de yerine konulursa,

$$\mathbf{x}(n) = \sum_{i=1}^M c_i^n \mathbf{g}_i \mathbf{g}_i^T [\mathbf{x}(0) - \mathbf{x}(\infty)] + \mathbf{x}(\infty) \quad (6.47)$$

elde edilir. Artık karesel ortalama hata değeri,

$$\begin{aligned} J_{ex}(n) &= \lambda^T \mathbf{x}(n) \\ &= \sum_{i=1}^M c_i^n \lambda^T \mathbf{g}_i \mathbf{g}_i^T [\mathbf{x}(0) - \mathbf{x}(\infty)] + \lambda^T \mathbf{x}(\infty) \\ &= \sum_{i=1}^M c_i^n \lambda^T \mathbf{g}_i \mathbf{g}_i^T [\mathbf{x}(0) - \mathbf{x}(\infty)] + J_{ex}(\infty) \end{aligned} \quad (6.48)$$

biçiminde yazılabilir. Burada,

$$J_{ex}(\infty) = \lambda^T \mathbf{x}(\infty) \quad (6.49)$$

$$= \sum_{j=1}^M \lambda_j x_j(\infty)$$

ifadeleri geçerlidir. (6.48) eşitliğinde sağ taraftaki ilk terim, artık karesel ortalama hatanın geçici hal davranışını tanımlar, oysa ikinci terim, onun uyarlamadan sonraki değerini, diğer bir deyişle kalıcı hal değerini gösterir.

Kalıcı hal elemanı $J_{ex}(\infty)$ için, ilişki matrisi $\mathbf{R}'$ nin özdeğerleri cinsinden bir ifade çıkarılmak istensin. Bu, (6.44) eşitliğindeki $\mathbf{x}(\infty)$, λ^T ile öncül çarpılarak ve bazı düzenlemelerle yapılır. Bununla birlikte daha basit bir yaklaşım, (6.30) eşitliğine geri dönüp, $n = \infty$ koyarak, her iki tarafın i' ye göre toplamını almaktır. Bu durumda,

$$J_{ex}(\infty) = J_{min} \frac{\sum_{i=1}^M \mu \lambda_i / (2 - \mu \lambda_i)}{1 - \sum_{i=1}^M \mu \lambda_i / (2 - \mu \lambda_i)} \quad (6.50)$$

yazılabilir. Sonuç olarak, (6.48) eşitliğinde (6.50) yerine konur ve (6.22) eşitliğinin ilk satırı kullanılırsa, karesel ortalama hata şu şekilde ifade edilebilir:

$$J(n) = \sum_{i=1}^M \gamma_i c_i^n + \frac{J_{min}}{1 - \sum_{i=1}^M \mu \lambda_i / (2 - \mu \lambda_i)} \quad (6.51)$$

Burada,

$$\gamma_i = \lambda^T \mathbf{g}_i \mathbf{g}_i^T [\mathbf{x}(0) - \mathbf{x}(\infty)], \quad i = 1, 2, \dots, M \quad (6.52)$$

yazılabilir. Başlangıç değeri $\mathbf{x}(0)$ ' in i . elemanının, ağırlık hatası vektörü başlangıç değeri $\boldsymbol{\varepsilon}(0) = \hat{\mathbf{w}}(0) - \mathbf{w}_o$ ile aşağıdaki gibi bir ilişkisi vardır:

$$x_i(0) = E[\mathbf{q}_i^H \boldsymbol{\varepsilon}(0) \boldsymbol{\varepsilon}^H(0) \mathbf{q}_i], \quad i = 1, 2, \dots, M \quad (6.53)$$

Burada $\mathbf{q}_i$, λ_i özdeğerlerine bağlı $\mathbf{R}$ ilişki vektörünün özvektörüdür.

6.6. Karesel Ortalama Hatanın Geçici Hal Davranışının Özellikleri

1. Özellik: Karesel ortalama hata $J(n)$ ' nin geçici hal elemanı $\sum_{i=1}^M \gamma_i c_i^n$ dalgalanmalar içermez. Burada γ_i sabit katsayılar, c_i ise $\mathbf{B}$ matrisinin özdeğerleridir. $\mathbf{B}$, simetrik matris olduğu için bu özdeğerlerin tümü gerçel sayılardır. Bu yüzden, iterasyon sayısı n ' e göre karesel ortalama hatanın beklendik değeri $E[J(n)]$ ' nin çizilmesi ile elde edilen toplamsal ortalamalı öğrenme eğrisi sadece örnekleri içerir.

Bununla birlikte, ortalaması alınmaksızın, karesel ortalama hata $J(n)$ ' nin n ' e göre çizilmesi ile elde edilen öğrenme eğrisi, gürültülü örnekler içerir. Adım uzunluğu parametresi μ azaldıkça, gürültünün genliği daha küçük olacaktır.

2. Özellik: Karesel ortalama hata $J(n)$, sadece adım uzunluğu parametresi μ aşağıdaki iki koşulu sağladığında, $J(\infty)$ ' nin kalıcı hal değerine yakınsamaktadır:

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (6.54)$$

$$\sum_{i=1}^M \frac{\mu \lambda_i}{2 - \mu \lambda_i} < 1$$

Adım uzunluğu parametresi μ , $2/\lambda_{\max}$ ' dan küçükse, $(2 - \mu \lambda_i)$ ifadesi tüm i ' ler için 2' ye yaklaşır ve ortalamanın iyi bir şekilde yakınsaması elde edilir. Böylece, (6.54) eşitliği,

$$0 < \mu < \frac{2}{\sum_{i=1}^M \lambda_i} \quad (6.55)$$

biçiminde basit hale gelir. (6.55) ifadesindeki koşul, LMS algoritmasının tüm kararlılığı için gerek ve yeter koşuldur.

3. Özellik: LMS algoritmasında karesel ortalama hatanın son değeri,

$$J(\infty) = \frac{J_{min}}{1 - \sum_{i=1}^M \mu \lambda_i / (2 - \mu \lambda_i)} \quad (6.56)$$

biçimindedir. Bu eşitlik, kararlılık için $c_i < 1$ olması şartı ile, (6.51) eşitliğinde $n = \infty$ konulması ile elde edilir.

6.7. LMS Algoritmasının Özeti

Parametreler:

M ; Süzgeç derecesi veya tap girişleri sayısı

μ ; Adım uzunluğu parametresi

Başlangıç Koşulları:

$$\hat{\mathbf{w}}(0) = \mathbf{0}$$

Veri:

(a) Verilen : $\mathbf{u}(n)$; n anındaki $M \times 1$ boyutlu tap giriş vektörü
 $d(n)$; n anındaki istenen yanıt

(b) Hesaplanacak olan: $\hat{\mathbf{w}}(n+1)$; $n+1$ anındaki tap ağırlık vektörünün kestirimi

Hesaplamalar:

$n = 0, 1, 2, \dots$ için

$$e(n) = d(n) - \hat{\mathbf{w}}^H(n) \mathbf{u}(n)$$

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \mu \mathbf{u}(n) e^*(n)$$

Uyarı: Tap ağırlık vektörünün başlangıç değeri $\hat{\mathbf{w}}(0)$, sıfıra eşit alındığından ve ayrıca enine süzgecin başlangıçtaki bütün girişleri sıfır alındığından, ikinci ifadede $\hat{\mathbf{w}}(1)$ 'in de sıfır olacağı görülmektedir. Buna göre LMS algoritmasının ilk iterasyonundan sonra daima $e(1) = d(1)$ bulunur.



BÖLÜM 7. ÖZYİNELİ EN KÜÇÜK KARELER (RLS: RECURSIVE LEAST SQUARES) ALGORİTMASI KULLANILARAK GERÇEKLEŞTİRİLEN ENİNE SÜZGEÇLER

Bu bölümde, uyarlamalı enine süzgeçlerin tasarımında kullanılan Özyineli En Küçük Kareler (RLS) algoritması sunulmaktadır. Bu algoritmada, süzgecin tap ağırlık vektörünün $n - 1$ anındaki kestirimi kullanılarak, bu vektörün gelen yeni veri üzerinde n anındaki güncellenmiş kestirimi hesaplanmaktadır.

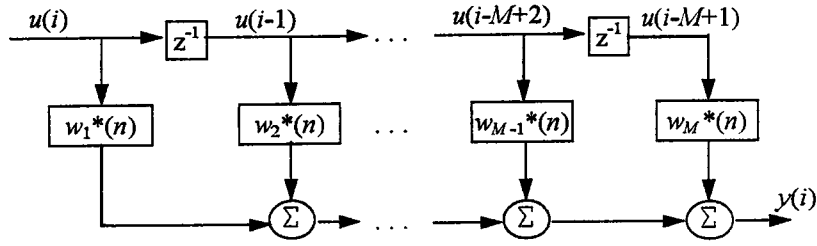
RLS algoritmasının çıkarılmasına geçilmeden önce, En Küçük Kareler metoduna ait bazı temel ifadeler gösterilmektedir. Sonra, Matris Tersi Alma Yasası olarak bilinen matris cebirindeki bir bağıntıdan yararlanarak RLS algoritması geliştirilmektedir.

RLS algoritmasının en önemli özelliği, algoritmanın başlangıç anına kadar uzanan giriş verisindeki tüm bilgiyi kullanmasıdır. RLS algoritmasının yakınsaması, LMS algoritmasının yakınsamasından daha hızlıdır. Bu başarısı, hesaplama karmaşıklığındaki büyük artış sonucu elde edilmiştir.

En Küçük Kareler metodunun özyineli gerçeklemelerine, başlangıç koşulları olarak bilinen hesaplamalarla ve yeni veri örneklerinde eski kestirimleri güncellemeyi kapsayan bilginin kullanımı ile başlanmaktadır. Bu yüzden gözlenecek verinin uzunluğu değişkendir. Uygun olarak, Başarı Göstergesi $\xi(n)$,

$$\xi(n) = \sum_{i=1}^n \beta(n,i) |e(i)|^2 \quad (7.1)$$

minimize edilmektedir. Burada n , gözlenecek verinin değişen uzunluğudur. Aynı zamanda başarı göstergesi $\xi(n)$ ' nin tanımındaki Ağırlık Faktörü veya İhmal Etme Faktörü de tanımlanmaktadır.



Şekil 7.1 Enine Süzgeç

Şekil 7.1' de enine süzgeç yapısı verilmektedir. i anındaki girişleri $u(i)$, $u(i-1)$, ..., $u(i-M+1)$ ' dir. $e(i)$ ise şöyle tanımlanır:

$$\begin{aligned} e(i) &= d(i) - y(i) \\ &= d(i) - \mathbf{w}^H(n) \mathbf{u}(i) \end{aligned} \quad (7.2)$$

Burada $\mathbf{u}(i)$, i anındaki giriş vektörüdür ve

$$\mathbf{u}^T(i) = [u(i), u(i-1), \dots, u(i-M+1)] \quad (7.3)$$

biçiminde tanımlanır. $\mathbf{w}(n)$ ise, n anındaki tap ağırlık vektörüdür ve

$$\mathbf{w}^T(n) = [w_1(n), w_2(n), \dots, w_M(n)] \quad (7.4)$$

biçiminde tanımlanır. Enine süzgecin tap ağırlıkları, $1 \leq i \leq n$ aralığındaki gözlem esnasında sabit kalmaktadır.

(7.1) eşitliğindeki ağırlık faktörü $\beta(n, i)$,

$$0 < \beta(n, i) \leq 1, \quad i = 1, 2, \dots, n \quad (7.5)$$

şartını sağlamaktadır. Ağırlık faktörü $\beta(n,i)$, süzgeç durağan olmayan ortamlarda çalışırken, gözlenecek verinin istatistiksel değişimlerini takip edebilmek için, daha önceki veriyi temin etmek için tasarlanmıştır. Ağırlık faktörünün yaygın olarak kullanılan biçimi, Üssel Ağırlık Faktörüdür ve

$$\beta(n,i) = \lambda^{n-i}, \quad i = 1, 2, \dots, n \quad (7.6)$$

biçiminde tanımlanır. λ , 1' e yakın ama 1' den küçük pozitif bir sayıdır. λ , 1' e eşit olduğu zaman, alışlagelmiş En Küçük Kareler metodu elde edilir. $1 - \lambda$ ' nın tersi, yaklaşık olarak algoritmanın belleğinin bir ölçüsüdür. $\lambda = 1$ özel durumu, sonsuz belleğe karşılık gelir. Bu yüzden Üssel Ağırlıklı En Küçük Kareler metodunda,

$$\xi(n) = \sum_{i=1}^n \lambda^{n-i} |e(i)|^2$$

biçimindeki başarı göstergesi minimize edilecektir. (7.7) eşitliğindeki başarı göstergesini minimum değere ulaştıran, tap ağırlık vektörü $w(n)$ ' nin optimum değeri,

$$\Phi(n)w(n) = \theta(n) \quad (7.8)$$

şeklindeki normal eşitlikle tanımlanmaktadır. $\Phi(n)$, $M \times M$ boyutlu ilişki matrisidir ve

$$\Phi(n) = \sum_{i=1}^n \lambda^{n-i} \mathbf{u}(i) \mathbf{u}^H(i) \quad (7.9)$$

biçiminde tanımlanır. $\theta(n)$, enine süzgecin girişleri ile istenen yanıt arasındaki $M \times 1$ boyutlu çapraz ilişki vektörüdür ve

$$\theta(n) = \sum_{i=1}^n \lambda^{n-i} \mathbf{u}(i) d^*(i) \quad (7.10)$$

biçiminde tanımlanır. $i = n$ terimi, (7.9) eşitliğinin sağ tarafındaki toplamdan ayrılırsa,

$$\Phi(n) = \lambda \left[\sum_{i=1}^{n-1} \lambda^{n-1-i} \mathbf{u}(i) \mathbf{u}^H(i) \right] + \mathbf{u}(n) \mathbf{u}^H(n) \quad (7.11)$$

elde edilir. (7.11) eşitliğinde köşeli parantezin içindeki ifade, ilişki matrisi $\Phi(n-1)$ ' e eşittir. Bu yüzden, girişlerin ilişki matrisinin güncellenmesi için

$$\Phi(n) = \lambda \Phi(n-1) + \mathbf{u}(n) \mathbf{u}^H(n) \quad (7.12)$$

biçiminde özyineli eşitlik elde edilir. $\mathbf{u}(n) \mathbf{u}^H(n)$ matris çarpımı düzeltme terimi görevini görür.

Benzer olarak (7.10) eşitliği kullanılırsa, çapraz ilişki vektörünün güncellenmesi için,

$$\theta(n) = \lambda \theta(n-1) + \mathbf{u}(n) d^*(n) \quad (7.13)$$

biçiminde özyineli eşitlik elde edilir.

(7.8) eşitliğine göre, tap ağırlık vektörünün en küçük kareler kestirimi $\hat{\mathbf{w}}(n)$ ' nin hesaplanması için, ilişki matrisi $\Phi(n)$ ' nin tersinin bulunması gereklidir. Pratikte, çok zaman harcayacağı için böyle bir işlemin uygulanmasından kaçınılır. Özellikle tap ağırlıkları sayısı M büyük olduğu zaman böyle bir işlemin uygulanması daha da zorlaşır. Burada, Matris Tersi Alma Yasası olarak bilinen matris cebirindeki bir temel sonuç kullanılacaktır. Başlangıç koşullarının, ilişki matrisi $\Phi(n)$ ' nin tekil çözümünün olmaması için uygun seçildiği varsayılacaktır.

7.1. Matris Tersi Alma Yasası

A ve **B**, pozitif belirli $M \times M$ boyutlu iki matris ise,

$$\mathbf{A} = \mathbf{B}^{-1} + \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^H \quad (7.14)$$

yazılabilir. Burada, **D** pozitif belirli $N \times N$ boyutlu bir başka matris, **C** ise $M \times N$ boyutlu bir matristir. Matris tersi alma yasasına göre, **A**'nın tersi,

$$\mathbf{A}^{-1} = \mathbf{B} - \mathbf{B}\mathbf{C}(\mathbf{D} + \mathbf{C}^H\mathbf{B}\mathbf{C})^{-1}\mathbf{C}^H\mathbf{B} \quad (7.15)$$

biçimindedir. Bu yasa, (7.14) ve (7.15) eşitliklerinin çarpımları ile kanıtlanabilir. Çünkü, bir kare matris ile tersinin çarpımı birim matrise eşittir.

7.2. Üssel Ağırlıklı Yinelemeli En Küçük Kareler Algoritması

İlişki matrisi $\Phi(n)$ 'nin pozitif belirli ve dolayısıyla tekil çözümünün olmamasının varsayılması ile, (7.12) özyineli eşitliğine matris tersi alma yasası uygulanırsa,

$$\mathbf{A} = \Phi(n)$$

$$\mathbf{B}^{-1} = \lambda\Phi(n-1)$$

$$\mathbf{C} = \mathbf{u}(n)$$

$$\mathbf{D} = 1$$

tanımlamaları yapılabilir. Bu tanımlamalar, (7.15) eşitliğindeki matris tersi alma yasasında yerine konulursa, ilişki matrisinin tersi için,

$$\Phi^{-1}(n) = \lambda^{-1}\Phi^{-1}(n-1) - \frac{\lambda^{-2}\Phi^{-1}(n-1)\mathbf{u}(n)\mathbf{u}^H(n)\Phi^{-1}(n-1)}{1 + \lambda^{-1}\mathbf{u}^H(n)\Phi^{-1}(n-1)\mathbf{u}(n)} \quad (7.16)$$

biçiminde özyineli eşitlik elde edilir. Hesaplama kolaylığı için,

$$\mathbf{P}(n) = \Phi^{-1}(n) \quad (7.17)$$

$$\mathbf{k}(n) = \frac{\lambda^{-1} \mathbf{P}(n-1) \mathbf{u}(n)}{1 + \lambda^{-1} \mathbf{u}^H(n) \mathbf{P}(n-1) \mathbf{u}(n)} \quad (7.18)$$

tanımlamaları kullanılır ve bu tanımlar (7.16) eşitliğinde yerine konursa,

$$\mathbf{P}(n) = \lambda^{-1} \mathbf{P}(n-1) - \lambda^{-1} \mathbf{k}(n) \mathbf{u}^H(n) \mathbf{P}(n-1) \quad (7.19)$$

eşitliği elde edilir. Burada, $\mathbf{P}(n)$, $M \times M$ boyutlu bir matris, $\mathbf{k}(n)$ ise $M \times 1$ boyutlu bir Kazanç Vektörüdür. (7.18) eşitliğinde düzenleme yapılırsa,

$$\begin{aligned} \mathbf{k}(n) &= \lambda^{-1} \mathbf{P}(n-1) \mathbf{u}(n) - \lambda^{-1} \mathbf{k}(n) \mathbf{u}^H(n) \mathbf{P}(n-1) \mathbf{u}(n) \\ &= [\lambda^{-1} \mathbf{P}(n-1) - \lambda^{-1} \mathbf{k}(n) \mathbf{u}^H(n) \mathbf{P}(n-1)] \mathbf{u}(n) \end{aligned} \quad (7.20)$$

elde edilir. (7.19) eşitliğinden, (7.20) deki köşeli parantez içindeki ifadenin $\mathbf{P}(n)$ ' e eşit olduğu görülür. Bu durumda (7.20) eşitliği,

$$\mathbf{k}(n) = \mathbf{P}(n) \mathbf{u}(n) \quad (7.21)$$

biçimine dönüşür. Böylece (7.17) eşitliğindeki tanım kullanılarak,

$$\mathbf{k}(n) = \Phi^{-1}(n) \mathbf{u}(n) \quad (7.22)$$

eşitliği yazılabilir.

7.2.1. Tap Ağırlık Vektörünün Zaman Güncellemesi

Tap ağırlık vektörünün en küçük kareler kestirimi $\hat{\mathbf{w}}(n)$ ' nin güncellenmesi için bir özyineli eşitlik geliştirilmektedir. Bunu gerçeklemek için (7.8), (7.13) ve (7.17) eşitlikleri kullanılmaktadır. Tap ağırlık vektörünün n anındaki en küçük kareler kestirimi,

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \Phi^{-1}(n)\boldsymbol{\theta}(n) \\ &= \mathbf{P}(n)\boldsymbol{\theta}(n) \\ &= \lambda\mathbf{P}(n)\boldsymbol{\theta}(n-1) + \mathbf{P}(n)\mathbf{u}(n)d^*(n)\end{aligned}\quad (7.23)$$

biçimindedir. (7.23) eşitliğinin sağ tarafındaki sadece ilk terimde, (7.19) eşitliğindeki $\mathbf{P}(n)$ yerine konursa,

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \mathbf{P}(n-1)\boldsymbol{\theta}(n-1) - \mathbf{k}(n)\mathbf{u}^H(n)\mathbf{P}(n-1)\boldsymbol{\theta}(n-1) + \mathbf{P}(n)\mathbf{u}(n)d^*(n) \\ &= \Phi^{-1}(n-1)\boldsymbol{\theta}(n-1) - \mathbf{k}(n)\mathbf{u}^H(n)\Phi^{-1}(n-1)\boldsymbol{\theta}(n-1) + \mathbf{P}(n)\mathbf{u}(n)d^*(n) \\ &= \hat{\mathbf{w}}(n-1) - \mathbf{k}(n)\mathbf{u}^H(n)\hat{\mathbf{w}}(n-1) + \mathbf{P}(n)\mathbf{u}(n)d^*(n)\end{aligned}\quad (7.24)$$

eşitliği elde edilir. (7.21) eşitliğine göre $\mathbf{P}(n)\mathbf{u}(n)$ çarpımı, kazanç vektörü $\mathbf{k}(n)$ ' e eşittir. Sonuç olarak tap ağırlık vektörü güncellemesi için istenen özyineli eşitlik,

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \hat{\mathbf{w}}(n-1) + \mathbf{k}(n)[d^*(n) - \mathbf{u}^H(n)\hat{\mathbf{w}}(n-1)] \\ &= \hat{\mathbf{w}}(n-1) + \mathbf{k}(n)\alpha^*(n)\end{aligned}\quad (7.25)$$

biçiminde elde edilmiş olur. Burada $\alpha(n)$, öncül kestirim hatasıdır ve

$$\begin{aligned}\alpha(n) &= d(n) - \mathbf{u}^T(n)\hat{\mathbf{w}}^*(n-1) \\ &= d(n) - \hat{\mathbf{w}}^H(n-1)\mathbf{u}(n)\end{aligned}\quad (7.26)$$

biçiminde tanımlanır. Burada, $\hat{\mathbf{w}}^H(n-1)\mathbf{u}(n)$ çarpımı, tap ağırlık vektörünün $n - 1$ anında elde edilmiş eski en küçük kareler kestirimine dayalı olarak, istenen yanıt $d(n)$ ' nin kestirimini gösterir.

Öncül kestirim hatası genelde,

$$e(n) = d(n) - \hat{\mathbf{w}}^H(n)\mathbf{u}(n) \quad (7.27)$$

şeklindeki Ardıl Kestirim Hatasından farklıdır. Çünkü Ardıl Kestirim Hatası, tap ağırlık vektörünün n anındaki, şimdiki en küçük kareler kestirimini içerir. Gerçekte $\alpha(n)$, $e(n)$ ' nin tap ağırlık vektörü güncellenmeden önceki 'deneme' değeridir. Bununla birlikte tap ağırlık vektörü için, (7.25) eşitliğinin özyineli algoritmasına götüren en küçük kareler optimizasyonunda, $\alpha(n)$ ' e değil de $e(n)$ ' e dayalı olarak başarı göstergesi minimize edilmiştir.

7.2.2. Başlangıç Koşulları

RLS algoritmasını uygulayabilmek için, ilişki matrisi $\Phi(n)$ ' nin tekil çözümünün olmamasını sağlayan $\mathbf{P}(0)$ başlangıç değerinin seçilmesiyle (7.19) eşitliğinin özyinelemesinin koşullandırılması gereklidir. Bu,

$$\left[\sum_{i=-n_0}^0 \lambda^{-i} \mathbf{u}(i) \mathbf{u}^H(i) \right]^{-1}$$

biçimindeki ifadenin tersinin alınması ile sağlanabilir. Buradaki $\mathbf{u}(i)$, $-n_0 \leq i \leq 0$ başlangıç veri bloğundan elde edilir. Daha basit bir yol, ilişki matrisi $\Phi(n)$ ' nin ifadesinde,

$$\Phi(n) = \sum_{i=1}^n \lambda^{n-i} \mathbf{u}(i) \mathbf{u}^H(i) + \delta \lambda^n \mathbf{I} \quad (7.28)$$

biçimde önemsiz sayılacak küçük bir değişiklik yapmaktır. Burada $\mathbf{I}$, $M \times M$ boyutlu birim matris ve δ , küçük pozitif sabit bir sayıdır. Bu değişiklik başlangıç değerini etkiler, RLS algoritmasının özyinelemelerine dokunmaz. Genellikle, büyük veri uzunluğu n için, δ 'nin seçimi önemsizdir. $n = 0$ için,

$$\Phi(0) = \delta \mathbf{I} \quad (7.29)$$

ifadesi söz konusudur. $\mathbf{P}(n)$, ilişki matrisi $\Phi(n)$ 'nin tersine eşit olduğuna göre, başlangıç değeri

$$\mathbf{P}(0) = \delta^{-1} \mathbf{I} \quad (7.30)$$

biçimindedir. Sonuçta, tap ağırlık vektörü için bir başlangıç değeri seçmek kalır. Ahşıl gelmiş biçimiyle,

$$\hat{\mathbf{w}}(0) = 0 \quad (7.31)$$

şeklinde bir seçim yapılır. Burada $\mathbf{0}$, $M \times 1$ boyutlu sıfır vektörüdür.

Tablo 7.1 RLS Algoritmasının Özeti

Algoritma şunlarla koşullandırılır:

$$\mathbf{P}(0) = \delta^{-1} \mathbf{I}$$

δ , Küçük pozitif sabit bir sayı

$$\hat{\mathbf{w}}(0) = 0$$

Her, $n = 1, 2, \dots$ için aşağıdakiler hesaplanır:

$$\mathbf{k}(n) = \frac{\lambda^{-1} \mathbf{P}(n-1) \mathbf{u}(n)}{1 + \lambda^{-1} \mathbf{u}^H(n) \mathbf{P}(n-1) \mathbf{u}(n)}$$

$$\alpha(n) = d(n) - \hat{\mathbf{w}}^H(n-1) \mathbf{u}(n)$$

$$\hat{\mathbf{w}}(n) = \hat{\mathbf{w}}(n-1) + \mathbf{k}(n) \alpha^*(n)$$

$$\mathbf{P}(n) = \lambda^{-1} \mathbf{P}(n-1) - \lambda^{-1} \mathbf{k}(n) \mathbf{u}^H(n) \mathbf{P}(n-1)$$

RLS algoritmasının hesaplamalarında kullanılan başlangıç koşullarının ve özyinelemelerin bir özeti Tablo 7.1 de verilmiştir.

7.3. Ağırlıklı Hata Kareleri Toplamının Güncelleme Yinelemeleri

Tap ağırlık vektörü, en küçük kareler kestirimi $\hat{\mathbf{w}}(n)$ ' e eşit ayarlandığı zaman, ağırlıklı hata karelerinin toplamı, minimum değeri $\xi_{min}(n)$ ' de sonuçlanır. $\xi_{min}(n)$ ' yi hesaplamak için

$$\xi(n) = \xi_d(n) - \theta^H(n) \hat{\mathbf{w}}(n) \quad (7.32)$$

ilişkisi kullanılır. Burada $\xi_d(n)$,

$$\begin{aligned} \xi_d(n) &= \sum_{i=1}^n \lambda^{n-i} |d(i)|^2 \\ &= \lambda \xi_d(n-1) + |d(n)|^2 \end{aligned} \quad (7.33)$$

biçiminde ifade edilir. (7.13), (7.25) ve (7.33) eşitlikleri (7.32) de yerine konursa,

$$\begin{aligned} \xi_{min}(n) &= \lambda [\xi_d(n-1) - \theta^H(n-1) \hat{\mathbf{w}}(n-1)] \\ &\quad + d(n)[d^*(n) - \mathbf{u}^H(n) \hat{\mathbf{w}}(n-1)] - \theta^H(n) \mathbf{k}(n) \alpha^*(n) \end{aligned} \quad (7.34)$$

eşitliği yazılabilir. Son terimde $\theta(n)$, orjinal biçimine getirilmiştir. (7.34) eşitliğinde sağ taraftaki ilk köşeli parantezin içindeki ifade, tanımı gereği $\xi_{min}(n-1)$ ' e eşittir. Aynı zamanda ikinci köşeli parantez içindeki ifade, tanımı gereği $\alpha^*(n)$ ' e eşittir. Son terimde, $\theta^H(n) \mathbf{k}(n)$ çarpımını aşağıdaki gibi ifade edebilmek için, kazanç vektörü $\mathbf{k}(n)$ ' nin tanımı kullanılırsa,

$$\begin{aligned} \theta^H(n) \mathbf{k}(n) &= \theta^H(n) \Phi^{-1}(n) \mathbf{u}(n) \\ &= [\Phi^{-1}(n) \theta(n)]^H \mathbf{u}(n) \end{aligned}$$

$$= \hat{\mathbf{w}}^H(n) \mathbf{u}(n)$$

eşitliği elde edilir. İkinci satırda ilişki matrisi $\Phi(n)$ ' nin Hermitian özelliği (yani, $\Phi^H = \Phi$) ve üçüncü satırda $\Phi^{-1}(n)\theta(n)$ ' nin en küçük kareler kestirimi $\hat{\mathbf{w}}(n)$ ' e eşit olduğu gerçeği kullanılmıştır. Buna göre, (7.34) eşitliği,

$$\begin{aligned} \xi_{min}(n) &= \lambda \xi_{min}(n-1) + d(n)\alpha^*(n) - \hat{\mathbf{w}}^H(n) \mathbf{u}(n) \alpha^*(n) \\ &= \lambda \xi_{min}(n-1) + \alpha^*(n) [d(n) - \hat{\mathbf{w}}^H(n) \mathbf{u}(n)] \\ &= \lambda \xi_{min}(n-1) + \alpha^*(n) e(n) \end{aligned} \quad (7.35)$$

biçiminde basitleştirilebilir. Burada $e(n)$, ardıl kestirim hatasıdır. (7.35) eşitliği, ağırlıklı hata karelerinin toplamını güncellemekte kullanılan özyinelemelerdir. Böylece $\alpha^*(n)e(n)$ çarpımının , bu güncelleme özyinelemesinde düzeltme terimini temsil ettiği görülmektedir. Bu çarpımın, gerçel değerli olduğuna ve

$$\alpha(n)e^*(n) = \alpha^*(n)e(n) \quad (7.36)$$

ifadesini gerçeklediğine de dikkat edilmelidir.

7.4. RLS Algoritmasının Karesel Ortalama Olarak Yakınsama Analizi

RLS algoritmasının karesel ortalama olarak yakınsama analizi için, istenen yanıt $d(n)$ ' nin ve giriş vektörü $\mathbf{u}(n)$ ' nin, Şekil 7.2' de verilen Çoklu Doğrusal Regresyon Modeli ile ilişkili olduğu varsayılacaktır. Özellikle daha önce kullanılan eşitlik

$$d(n) = e_o(n) + \mathbf{w}_o^H \mathbf{u}(n) \quad (7.37)$$

burada yine sözkonusudur. $M \times 1$ boyutlu w_o vektörü, modelin regresyon parametre vektörüdür. $e_o(n)$ ise, ölçü hatasıdır. Ölçü hatası süreci $\{e_o(n)\}$, sıfır ortalamalı, σ^2 değişintili beyaz gürültüdür. Durağan bir ortam için, tap ağırlıkları, $\lambda = 1$ seçilmesine uygun bir koşul ile yavaş yavaş uyarıldığı zaman, en iyi kalıcı hal sonuçları elde edilir.

RLS algoritmasında, öncül kestirim hatası $\alpha(n)$ ve ardıl kestirim hatası $e(n)$ olarak tanımlanmış iki hata işareti vardır. Kısım 7.2 de verilen başlangıç koşulları dikkate alınır, bu iki hatanın karesel ortalama değerlerinin n anında farklı şekilde değiştiği bulunur. $n = 1$ anında, $\alpha(n)$ ' nin karesel ortalama değeri, istenen yanıt $d(n)$ ' nin karesel ortalama değerine eşit olan, büyük bir değere erişir ve sonra n ' nin arttırılmasıyla esas değerine iner. Diğer yanda $e(n)$ ' nin karesel ortalama değeri, $n = 1$ anında küçük bir değere varır ve sonra n ' nin arttırılması ile esas değerine yükselir. RLS algoritmasında öğrenme eğrisini elde etmek için hata işareti olarak, LMS algoritmasındakilerle aynı genel şekle sahip olan $\alpha(n)$ seçilir. Böylece, RLS ve LMS öğrenme eğrileri arasında doğrudan bir karşılaştırma yapabilme imkanı doğmuş olur. Bu yüzden RLS algoritmasının karesel ortalama olarak yakınsama analizi, öncül kestirim hatası $\alpha(n)$ ' e dayandırılacaktır. (7.26) eşitliğinde (7.37) eşitliği yerine konularak istenen yanıt $d(n)$ yok edilirse, öncül kestirim hatası,

$$\begin{aligned}\alpha(n) &= e_o(n) - [\hat{w}(n-1) - w_o]^H u(n) \\ &= e_o(n) - \varepsilon^H(n-1)u(n)\end{aligned}\quad (7.38)$$

olarak bulunur. Burada $\varepsilon(n-1)$, $n - 1$ anındaki ağırlık hatası vektörüdür. Bu durumda karesel ortalama hata,

$$J'(n) = E[|\alpha(n)|^2] \quad (7.39)$$

biçiminde ifade edilir. $J'(n)$ terimindeki üs işareti, $\alpha(n)$ ' nin karesel ortalama değerini, $e(n)$ ' nin karesel ortalama değerinden ayırt etmek için kullanılmıştır. (7.38) eşitliği (7.39) eşitliğinde yerine yazılırsa,

$$J'(n) = E\left[|e_o(n)|^2\right] + E\left[\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n-1)\boldsymbol{\varepsilon}^H(n-1)\mathbf{u}(n)\right] - E\left[\boldsymbol{\varepsilon}^H(n-1)\mathbf{u}(n)e_o^*(n)\right] - E\left[e_o(n)\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n-1)\right] \quad (7.40)$$

elde edilir. Ölçü hatası $e_o(n)$ ile ağırlık hatası vektörü $\boldsymbol{\varepsilon}(n-1)$ ' nin istatistiksel bağımsız oldukları ve $e_o(n)$ ' nin sıfır ortalamalı olduğu varsayılacaktır. σ^2 , ölçü hatası $e_o(n)$ ' nin değışintisi olmak üzere karesel ortalama hata

$$J'(n) = \sigma^2 + E\left[\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n-1)\boldsymbol{\varepsilon}^H(n-1)\mathbf{u}(n)\right] \quad (7.41)$$

şeklini alır. (7.41) eşitliğinin sağ tarafındaki ikinci terim için,

$$\begin{aligned} E\left[\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n-1)\boldsymbol{\varepsilon}^H(n-1)\mathbf{u}(n)\right] &= E\left[\text{tr}\left\{\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n-1)\boldsymbol{\varepsilon}^H(n-1)\mathbf{u}(n)\right\}\right] \\ &= E\left[\text{tr}\left\{\mathbf{u}(n)\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n-1)\boldsymbol{\varepsilon}^H(n-1)\right\}\right] \\ &= \text{tr}\left\{E\left[\mathbf{u}(n)\mathbf{u}^H(n)\boldsymbol{\varepsilon}(n-1)\boldsymbol{\varepsilon}^H(n-1)\right]\right\} \\ &= \text{tr}\left\{E\left[\mathbf{u}(n)\mathbf{u}^H(n)\right]E\left[\boldsymbol{\varepsilon}(n-1)\boldsymbol{\varepsilon}^H(n-1)\right]\right\} \\ &= \text{tr}\left\{\mathbf{R}\mathbf{K}(n-1)\right\} \end{aligned} \quad (7.42)$$

yazılabilir. Bu eşitlik eşitliği (7.41) eşitliğinde yerine konulursa

$$J'(n) = \sigma^2 + \text{tr}\left\{\mathbf{R}\mathbf{K}(n-1)\right\} \quad (7.43)$$

elde edilir.

Kısım 6.4' den ağırlık hatası ilişki matrisinin

$$\begin{aligned}\mathbf{K}(n) &= E[\boldsymbol{\varepsilon}(n)\boldsymbol{\varepsilon}^H(n)] \\ &= \sigma^2 \Phi^{-1}(n)\end{aligned}\quad (7.44)$$

olduğu hatırlansın. Burada $\Phi^{-1}(n)$, zaman değişimli ilişki matrisinin tersidir. Büyük n değeri için $\Phi^{-1}(n)$, $n^{-1}\mathbf{R}^{-1}$ değerine yaklaşır ve

$$\mathbf{K}(n) \cong \frac{\sigma^2}{n} \mathbf{R}^{-1}, \quad \text{büyük } n \text{ değeri için} \quad (7.45)$$

elde edilir. Burada $\mathbf{R}^{-1}$, giriş vektörü $\mathbf{u}(n)$ 'nin toplamsal ortalamalı ilişki matrisidir. (7.45) eşitliğinde n yerine $n - 1$ yazılır ve bu yeni eşitlik (7.43) eşitliğinde yerine yazılırsa karesel ortalama hata

$$\begin{aligned}J'(n) &\cong \sigma^2 + \frac{\sigma^2}{n-1} \text{tr}[\mathbf{R}^{-1}\mathbf{R}] \\ &= \sigma^2 + \frac{\sigma^2}{n-1} \text{tr}[\mathbf{I}]\end{aligned}$$

olarak bulunur. Bir kare matrisin izi köşegen elemanlarının toplamına eşit olduğu için $\text{tr}[\mathbf{I}]$, M e eşittir. Aynı zamanda büyük n değeri için, $n - 1 \cong n$ dir. Böylece RLS algoritması ile üretilen Karesel Ortalama Yenilik için şu basit formül bulunur:

$$J'(n) \cong \sigma^2 + \frac{M\sigma^2}{n} \quad (7.46)$$

Bu elde edilenlere dayalı olarak, aşağıdaki sonuçlar çıkarılabilir:

1. RLS algoritması karesel ortalama olarak yaklaşık $2M$ iterasyonda yakınsar, burada M enine süzgeçteki katsayı adedidir. Bu demektir ki, RLS algoritmasının yakınsama hızı, LMS algoritmasının yakınsama hızından daha fazladır.

2. İterasyon sayısı n sonsuza yaklaşırken karesel ortalama yenilik, ölçü hatası $e_o(n)$ ' nin değışintisi σ^2 ' ye eşit olan sonuç değerine yaklaşır. Minimum karesel ortalama hata aynı zamanda σ^2 ' ye eşit olduğundan, RLS algoritması teorik olarak sıfır artık karesel ortalama hata veya sıfır hata oranı üretir.

Burada şunun belirtilmesi gerekir: RLS algoritmasının LMS algoritmasına göre yakınsama hızındaki gelişmişliği, sadece ölçü hatası $e_o(n)$ istenen yanıt $d(n)$ ile karşılaştırıldığında küçük olduğu zaman yani işaret-gürültü oranı yüksek olduğu zaman geçerlidir.

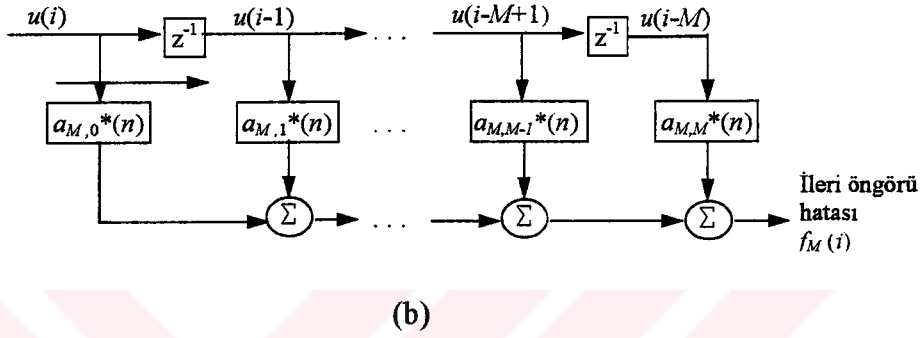
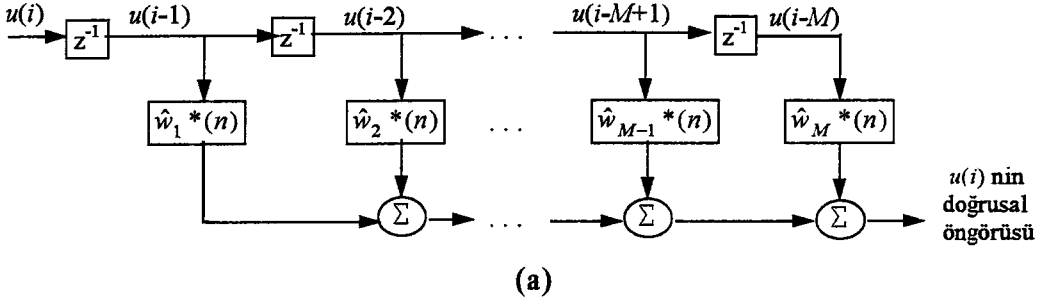


BÖLÜM 8. ÖZYİNELİ EN KÜÇÜK KARELER KAFES SÜZGEÇLER (RLSL: RECURSIVE LEAST SQUARES LATTICE FİLTERS)

Bu bölümde modüler biçimde olan çok aşamalı kafes öngörücü yapıma dayalı En Küçük Kareler algoritmalarının bir başka çeşidi tanıtılacaktır. Hem derece güncelleme özyinelemeleri, hem de zaman güncelleme özyinelemelerini içeren ve bir bütün olarak Özyineli En Küçük Kareler Kafes (RLSL) Algoritması olarak bilinen bu algoritma, RLS algoritması kadar güçlüdür. Çünkü, RLS algoritmasında olduğu gibi bu algoritma da, giriş işaretinin ilişki matrisinin özdeğer dağılımındaki değişimlere tamamen duyarlıdır.

RLSL algoritmalarındaki ilk çalışma, Morf (1974) tarafından gerçekleştirilmiştir. Daha sonra, resmi olarak algoritmanın elde edilmesi, Morf ve diğerlerinin (1977 ve 1978) çalışmaları ile olmuştur. Bu önemli çalışma, daha detaylı çalışmalara, genelleştirmelere ve bir çok uygulamalara Lee ve diğerleri (1981), Satorius ve diğerleri (1981), Kailath (1982), Friedlander (1982), Porat ve diğerleri (1982 ve 1983), Lev-Ari ve diğerleri (1984), Ling ve diğerleri (1985) önderlik etmiştir.

Uyarlamalı kafes öngörücülerin tasarımında sadece RLSL algoritmaları kullanılmaz. Farklı işaret işleme uygulamaları için bir çok uyarlamalı kafes algoritmaları çıkarılmıştır. Bunlardan biri, çok modüllü kafes öngörücülerin yansıma katsayılarının özyineli olarak hesaplanması için Griffiths (1977 ve 1978) tarafından önerilen Gradyent Uyarlamalı Kafes (GAL) algoritmalarıdır. GAL algoritmaları geliştirilirken LMS algoritmalarından esinlenilmiştir. Bu algoritmaların temel özelliği, RLSL algoritmaları ile karşılaştırıldığında hesaplama karmaşıklığının azaltılmış olmasıdır. Bununla birlikte genelde, GAL algoritmalarının başarımı RLSL algoritmalarının başarımından daha alt seviyededir. Çünkü RLSL algoritmaları, her zaman adımda doğrusal en küçük kareler probleminin tam çözümünü verirken, GAL algoritmaları yaklaşımlar kullanır.



Şekil 8.1 (a) M . Dereceden İleri Öngörücü
(b) İleri Öngörü Hata Süzgeci

En Küçük Kare tap ağırlık vektörü $\mathbf{a}_m(n)$ ile gösterilen m . dereceden bir ileri öngörü hata süzgeci Şekil 8.1 (a)'daki biçimde düşünölsün. Tanımı gereğı $\mathbf{a}_m(n)$ 'nin ilk elemanı birim değere eşittir. $\mathbf{u}_{m+1}(i)$, süzgecin i anındaki $(m+1) \times 1$ boyutlu tap giriş vektörünü gösterir, burada $1 \leq i \leq n$ 'dir. Şekil 8.1 (b)'de görölen ileri öngörü hata süzgecinin çıkışında, $\mathbf{u}_{m+1}(i)$ tap giriş vektörüne yanıt olarak üretilen ileri ardıl kestirim hatası,

$$f_m(i) = \mathbf{a}_m^H(n) \mathbf{u}_{m+1}(i), \quad 1 \leq i \leq n \quad (8.1)$$

biçiminde ifade edilebilir. Önpencerelemenin kullanıldığı varsayılacak ve bu yüzden, $i \leq 0$ için $u(i) = 0$ alınabilecektir. İleri ardıl öngörü hata karelerinin ağırlıklı toplamı,

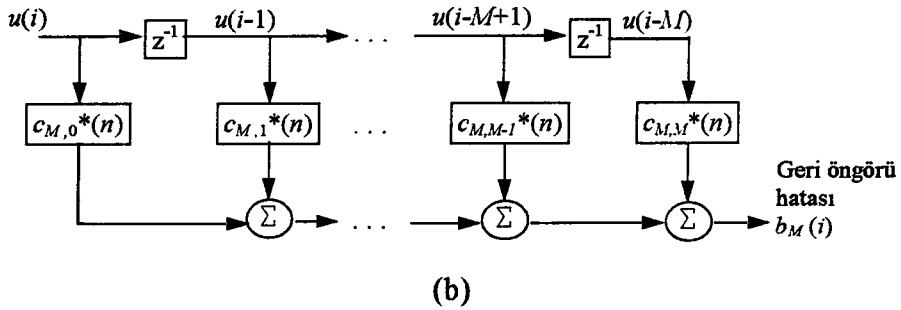
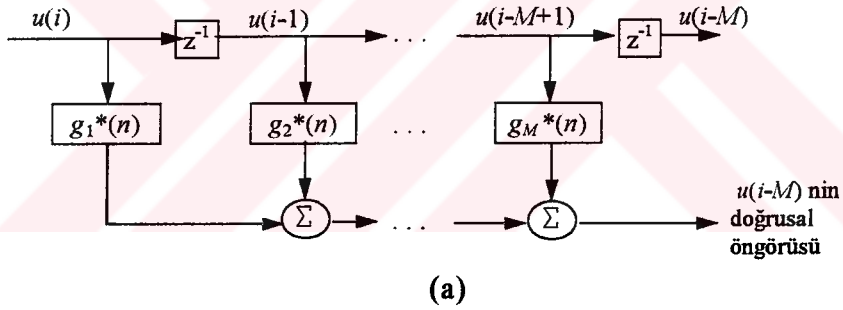
$$F_m(n) = \sum_{i=1}^n \lambda^{n-i} |f_m(i)|^2 \quad (8.2)$$

biçimindedir. Burada λ , üssel ağırlık faktörüdür. Tap ağırlık vektörü $\mathbf{a}_m(n)$, $F_m(n)$ başarı göstergesinin minimize edilmesinin sonucudur. $F_m(n)$, $\mathbf{a}_m(n)$ ' nin ilk elemanını birim değer eşitlemeye zorlar. $\mathbf{a}_m(n)$, bu minimizasyon esnasında sabit bir vektör olarak işlem görmektedir.

Şimdi, En Küçük Kareler tap ağırlık vektörü $\mathbf{c}_m(n)$ ile gösterilen m . dereceden bir geri öngörü hata süzgeci Şekil 8.2 (a)' daki biçimde düşünölsün. Tanımı gereğı $\mathbf{c}_m(n)$ ' nin son elemanı birim değere eşittir. Şekil 8.2 (b)' de görölen geri öngörü hata süzgecinin çıkışında, $\mathbf{u}_{m+1}(i)$ tap giriş vektörüne yanıt olarak üretilen geri ardıl öngörü hatası,

$$b_m(i) = \mathbf{c}_m^H(n) \mathbf{u}_{m+1}(i), \quad 1 \leq i \leq n \quad (8.3)$$

biçiminde ifade edilebilir.



Şekil 8.2 (a) M . Dereceden Geri Öngörücü
(b) Geri Öngörü Hata Süzgeci

Geri ardıl öngörü hata karelerinin ağırlıklı toplamı,

$$B_m(n) = \sum_{i=1}^n \lambda^{n-i} |b_m(i)|^2 \quad (8.4)$$

biçimindedir. Tap ağırlık vektörü $c_m(n)$, ikinci başarı göstergesi $B_m(n)$ ' nin minimize edilmesinin sonucudur. $B_m(n)$, $c_m(n)$ ' nin son elemanını birim değere eşitlemeye zorlar. Burada da yine $c_m(n)$, bu minimizasyon esnasında sabit bir vektör olarak işlem görmektedir.

8.1. Derece Güncelleme Yinelemeleri

$\Phi_{m+1}(n)$, m . dereceden ileri öngörü hata süzgecine uygulanan $u_{m+1}(i)$ tap giriş vektörünün $(m+1) \times (m+1)$ boyutlu ilişki matrisidir, burada $1 \leq i \leq n$ ' dir. Bu süzgeç,

$$\Phi_{m+1}(n) a_m(n) = \begin{bmatrix} F_m(n) \\ 0_m \end{bmatrix} \quad (8.5)$$

biçimindeki genişletilmiş normal eşitlik ile karakterize edilebilir (Bkz. (3.48) eşitliği). Burada, 0_m , $m \times 1$ boyutlu sıfır vektörüdür.

$\Phi_{m+1}(n)$ ilişki matrisi, tap giriş vektörü $u_{m+1}(i)$ ' nin ilk ve son elemanına bağlı olarak iki farklı biçimde parçalara ayrılabilir. Kullanılacak olan parçalama biçimi, $a_m(n)$ tap ağırlık vektörünün m . dereceden öngörüye ait olarak ilişkilendiren ve $a_{m-1}(n)$ tap ağırlık vektörünü $(m-1)$. dereceden öngörüye ait olarak ilişkilendiren parçalama biçimidir:

$$\Phi_{m+1}(n) = \begin{bmatrix} \Phi_m(n) & | & \theta_2(n) \\ \hline \theta_2^H(n) & | & U_2(n) \end{bmatrix} \quad (8.6)$$

Burada $\Phi_m(n)$, tap giriş vektörü $u_m(i)$ ' nin $m \times m$ boyutlu ilişki matrisi, $\theta_2(n)$, $u_m(i)$ ile $u(i-m)$ arasındaki $m \times 1$ boyutlu çapraz ilişki vektörü, $U_2(n)$ ise $1 \leq i \leq n$ için $u(i-m)$ tap

girişinin karesel değerlerinin ağırlıklı toplamıdır. $U_2(n)$, $n-m \leq 0$ için sıfırdır. (8.6) eşitliğinin her iki tarafı, ilk m elemanı $\mathbf{a}_{m-1}(n)$ vektörü ile belirlenen ve son elemanı sıfır olan $(m+1) \times 1$ boyutlu bir vektör ile ardıl çarpılırsa,

$$\begin{aligned} \Phi_{m+1}(n) \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} &= \begin{bmatrix} \Phi_m(n) & \theta_2(n) \\ \theta_2^H(n) & U_2(n) \end{bmatrix} \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \Phi_m(n) \mathbf{a}_{m-1}(n) \\ \theta_2^H(n) \mathbf{a}_{m-1}(n) \end{bmatrix} \end{aligned} \quad (8.7)$$

elde edilir. $\Phi_m(n)$ ve $\mathbf{a}_{m-1}(n)$ her ikisinde aynı zaman değişkeni n' e sahiptir. $(m-1)$. dereceden ileri öngörü hata süzgeci için, n anındaki genişletilmiş normal eşitlik, (8.5) eşitliğinde m yerine $m-1$ konularak,

$$\Phi_m(n) \mathbf{a}_{m-1}(n) = \begin{bmatrix} F_{m-1}(n) \\ \mathbf{0}_{m-1} \end{bmatrix} \quad (8.8)$$

biçiminde elde edilir. Burada, $F_{m-1}(n)$, süzgeç çıkışındaki ileri ardıl öngörü hata karelerinin ağırlıklı toplamı ve $\mathbf{0}_{m-1}$ ise, $(m-1) \times 1$ boyutlu sıfır vektördür. Şöyle bir skaler tanımlansın:

$$\Delta_{m-1}(n) = \theta_2^H(n) \mathbf{a}_{m-1}(n) \quad (8.9)$$

Bu durumda (8.7) eşitliği,

$$\Phi_{m+1}(n) \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} F_{m-1}(n) \\ \mathbf{0}_{m-1} \\ \Delta_{m-1}(n) \end{bmatrix} \quad (8.10)$$

biçiminde yazılabilir.

m . dereceden geri öngörü hata süzgeci,

$$\Phi_{m+1}(n)\mathbf{c}_m(n) = \begin{bmatrix} \mathbf{0}_m \\ B_m(n) \end{bmatrix} \quad (8.11)$$

biçimindeki normal eşitlikle karakterize edilebilir. Bu sefer, ilişki matrisi $\Phi_m(n)$ ' nin aşağıdaki gibi, bir başka parçalara ayrılmış biçimi kullanılacaktır:

$$\Phi_{m+1}(n) = \begin{bmatrix} U_1(n) & \vdots & \theta_1^H(n) \\ \theta_1(n) & \vdots & \Phi_m(n-1) \end{bmatrix} \quad (8.12)$$

Burada $U_1(n)$, $1 \leq i \leq n$ için $u(i)$ girişinin karesel değerlerinin ağırlıklı toplamı, $\theta_1(n)$, tap giriş vektörü $\mathbf{u}_m(i-1)$ ile $u(i)$ arasındaki $m \times 1$ boyutlu çapraz ilişki vektörü, $\Phi_m(n-1)$ ise $\mathbf{u}_m(i-1)$ ' nin $m \times m$ boyutlu ilişki matrisidir. (8.12) eşitliğinin her iki tarafı, ilk elemanı sıfır olan ve geri kalan m elemanı $(m-1)$. dereceden geri öngörü hata süzgecine ait $\mathbf{c}_{m-1}(n-1)$ tap ağırlık vektörü ile belirlenen $(m+1) \times 1$ boyutlu bir vektör ile ardıl çarpılırsa,

$$\begin{aligned} \Phi_{m+1}(n) \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} &= \begin{bmatrix} U_1(n) & \vdots & \theta_1^H(n) \\ \theta_1(n) & \vdots & \Phi_m(n-1) \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} \\ &= \begin{bmatrix} \theta_1^H(n)\mathbf{c}_{m-1}(n-1) \\ \Phi_m(n-1)\mathbf{c}_{m-1}(n-1) \end{bmatrix} \end{aligned} \quad (8.13)$$

elde edilir. $\Phi_m(n-1)$ ve $\mathbf{c}_{m-1}(n-1)$ her ikisinde aynı zaman değişkeni n ' e sahiptir. $(m-1)$. dereceden geri öngörü hata süzgeci için, $n-1$ anındaki genişletilmiş normal eşitlik (8.11) eşitliğinde m yerine $m-1$, n yerine $n-1$ konularak,

$$\Phi_m(n-1)\mathbf{c}_{m-1}(n) = \begin{bmatrix} \mathbf{0}_{m-1} \\ B_{m-1}(n-1) \end{bmatrix} \quad (8.14)$$

biçiminde elde edilir. Burada, $B_{m-1}(n-1)$, süzgeç çıkışında $n-1$ anındaki geri ardıl öngörü hata karelerinin ağırlıklı toplamıdır. Şöyle bir ikinci skaler tanımlansın:

$$\Delta'_{m-1}(n) = \theta_1^H(n) \mathbf{c}_{m-1}(n-1) \quad (8.15)$$

Üs işareti bu yeni skaleri, ilk skaler olan $\Delta_{m-1}(n)$ ' den ayırt etmek için kullanılmıştır. Bu tanımı ve (8.14) eşitliğini (8.13) eşitliğinde kullanırsak,

$$\Phi_{m+1}(n) \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} = \begin{bmatrix} \Delta'_{m-1}(n) \\ \mathbf{0}_{m-1} \\ B_{m-1}(n-1) \end{bmatrix} \quad (8.16)$$

elde edilir. (8.16) eşitliği, $\Delta_{m-1}(n)/B_{m-1}(n-1)$ oranı ile çarpılarak (8.10) eşitliğinden çıkarılırsa,

$$\Phi_{m+1}(n) \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} - \frac{\Delta_{m-1}(n)}{B_{m-1}(n-1)} \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} = \begin{bmatrix} F_{m-1}(n) - \frac{\Delta_{m-1}(n)\Delta'_{m-1}(n)}{B_{m-1}(n-1)} \\ \mathbf{0}_m \end{bmatrix} \quad (8.17)$$

elde edilir. (8.5) ve (8.17) eşitlikleri karşılaştırılırsa,

$$\mathbf{a}_m(n) = \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} - \frac{\Delta_{m-1}(n)}{B_{m-1}(n-1)} \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} \quad (8.18)$$

$$F_m(n) = F_{m-1}(n) - \frac{\Delta_{m-1}(n)\Delta'_{m-1}(n)}{B_{m-1}(n-1)} \quad (8.19)$$

biçimindeki derece güncelleme özyinelemeleri elde edilir.

Benzer biçimde, (8.10) eşitliği, $\Delta'_{m-1}(n)/F_{m-1}(n)$ oranı ile çarpılarak (8.16) eşitliğinden çıkartılırsa,

$$\Phi_{m+1}(n) \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} - \frac{\Delta'_{m-1}(n)}{F_{m-1}(n)} \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_m \\ B_{m-1}(n-1) - \frac{\Delta_{m-1}(n)\Delta'_{m-1}(n)}{F_{m-1}(n)} \end{bmatrix} \quad (8.20)$$

elde edilir. (8.11) ve (8.20) eşitlikleri karşılaştırılırsa,

$$\mathbf{c}_m(n) = \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} - \frac{\Delta'_{m-1}(n)}{F_{m-1}(n)} \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} \quad (8.21)$$

$$B_m(n) = B_{m-1}(n-1) - \frac{\Delta_{m-1}(n)\Delta'_{m-1}(n)}{F_{m-1}(n)} \quad (8.22)$$

biçimindeki derece güncelleme özyinelemeleri elde edilir.

8.1.1. $\Delta_{m-1}(n)$ ve $\Delta'_{m-1}(n)$ Parametreleri Arasındaki İlişki

(8.9) ve (8.15) eşitlikleri ile $\Delta_{m-1}(n)$ ve $\Delta'_{m-1}(n)$ parametreleri birbirlerinin kompleks eşleniğidir:

$$\Delta'_{m-1}(n) = \Delta_{m-1}^*(n) \quad (8.23)$$

Bu ilişki, üç aşamada ispatlanacaktır:

1. (8.10) eşitliğinin her iki tarafı, satır vektörü

$$[0, \mathbf{c}_{m-1}^H(n-1)]$$

ile öncül çarpılırsa, elde edilen sonuç skalerdir.

$$\begin{aligned} [0, \mathbf{c}_{m-1}^H(n-1)] \Phi_{m+1}(n) \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} &= [0, \mathbf{c}_{m-1}^H(n-1)] \begin{bmatrix} F_{m-1}(n) \\ \mathbf{0}_{m-1} \\ \Delta_{m-1}(n) \end{bmatrix} \\ &= \Delta_{m-1}(n) \end{aligned} \quad (8.24)$$

elde edilen sonuç skalerdir. Burada, $\mathbf{c}_{m-1}(n-1)$ 'nin son elemanının birim değere eşit olduğu gerçeği kullanıldı.

2. (8.16) eşitliğinin her iki tarafına Hermitian işlemi (transpoz ve kompleks eşlenik alma) uygulanırsa ve ilişki matrisi $\Phi_{m+1}(n)$ 'nin Hermitian özelliği (yani, $\Phi^H = \Phi$) kullanılırsa

$$\begin{bmatrix} 0, & \mathbf{c}_{m-1}^H(n-1) \end{bmatrix} \Phi_{m+1}(n) = \begin{bmatrix} \Delta'_{m-1} * (n), & \mathbf{0}_{m-1}^T, & B_{m-1}(n-1) \end{bmatrix}$$

elde edilir. Bu eşitliğin her iki tarafı $\begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix}$ ile ardıl çarpılırsa,

$$\begin{aligned} \begin{bmatrix} 0, & \mathbf{c}_{m-1}^H(n-1) \end{bmatrix} \Phi_{m+1}(n) \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} &= \begin{bmatrix} \Delta'_{m-1} * (n), & \mathbf{0}_{m-1}^T, & B_{m-1}(n-1) \end{bmatrix} \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} \\ &= \Delta'_{m-1} * (n) \end{aligned} \quad (8.25)$$

elde edilir. Burada, $\mathbf{a}_{m-1}(n)$ 'nin ilk elemanının birim değere eşit olduğu gerçeği kullanıldı.

3. (8.24) ve (8.25) eşitlikleri karşılaştırılırsa, $\Delta_{m-1}(n)$ ve $\Delta'_{m-1}(n)$ parametreleri arasındaki (8.23) eşitliğinin ispatı yapılmış olur.

8.1.2. Ardıl Öngörü Hatalarının Derece Güncelleme Özyinelemeleri

İleri yansımaya katsayısı,

$$\Gamma_{f,m}(n) = -\frac{\Delta_{m-1}(n)}{B_{m-1}(n-1)}, \quad m = 1, 2, \dots, M \quad (8.26)$$

biçiminde tanımlanır. Burada M , kafes öngörücünün son derecesidir.

Geri yansımaya katsayısı da,

$$\begin{aligned}\Gamma_{b,m}(n) &= -\frac{\Delta'_{m-1}(n)}{F_{m-1}(n)} \\ &= -\frac{\Delta_{m-1}^*(n)}{F_{m-1}(n)}, \quad m = 1, 2, \dots, M\end{aligned}\quad (8.27)$$

biçiminde tanımlanır. Bu tanımlar (8.18) ve (8.21) eşitliklerinde kullanılırsa,

$$\mathbf{a}_m(n) = \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} + \Gamma_{f,m}(n) \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} \quad (8.28)$$

ve

$$\mathbf{c}_m(n) = \begin{bmatrix} 0 \\ \mathbf{c}_{m-1}(n-1) \end{bmatrix} + \Gamma_{b,m}(n) \begin{bmatrix} \mathbf{a}_{m-1}(n) \\ 0 \end{bmatrix} \quad (8.29)$$

yazılabilir. Burada $m = 1, 2, \dots, M$ dir.

İleri ardıl öngörü hatası $f_m(n)$,

$$f_m(n) = \mathbf{a}_m^H(n) \mathbf{u}_{m+1}(n) \quad (8.30)$$

m . dereceden ileri öngörü hata süzgecinin tap giriş vektörü $\mathbf{u}_{m+1}(n)$ ' e yanıtı olarak elde edilen çıkışa eşittir. Tap giriş vektörü $\mathbf{u}_{m+1}(n)$, şu iki farklı biçimde parçalara ayrılabilir:

$$\mathbf{u}_{m+1}(n) = \begin{bmatrix} \mathbf{u}_m(n) \\ u(n-m) \end{bmatrix} \quad (8.31)$$

$$\mathbf{u}_{m+1}(n) = \begin{bmatrix} u(n) \\ \mathbf{u}_m(n-1) \end{bmatrix} \quad (8.32)$$

(8.28) eşitliği (8.30)'da yerine konulursa ve (8.28) eşitliğinin sağ tarafındaki ilk terim için (8.31)'deki parçalama biçimi, ikinci terim için de (8.32)'deki parçalama biçimi kullanılırsa, ileri ardıl öngörü hatası için

$$f_m(n) = f_{m-1}(n) + \Gamma_{f,m}^*(n)b_{m-1}(n-1), \quad m = 1, 2, \dots, M \quad (8.33)$$

biçimindeki derece güncelleme özyinelemeleri elde edilir. $f_{m-1}(n)$, ileri ardıl öngörü hatası ve $b_{m-1}(n-1)$, geri ardıl öngörü hatasının geciktirilmiş elemanıdır. Her ikisi de kestirim derecesi $m = 1$ içindir.

Geri öngörü hatası derece güncelleme özyinelemeleri için de, benzer bir işlem uygulanacaktır. Geri ardıl öngörü hatası $b_m(n)$,

$$b_m(n) = \mathbf{c}_m^H(n)\mathbf{u}_{m+1}(n) \quad (8.34)$$

m . dereceden geri öngörü hata süzgecinin tap giriş vektörü $\mathbf{u}_{m+1}(n)$ ' e yanıtı olarak elde edilen çıkışa eşittir. (8.29) eşitliği (8.34)'da yerine konulursa ve (8.29) eşitliğinin sağ tarafındaki ilk terim için (8.32) eşitliğindeki parçalama biçimi, ikinci terim için de (8.31) eşitliğindeki parçalama biçimi kullanılırsa, geri ardıl öngörü hatası için,

$$b_m(n) = b_{m-1}(n-1) + \Gamma_{b,m}^*(n)f_{m-1}(n), \quad m = 1, 2, \dots, M \quad (8.35)$$

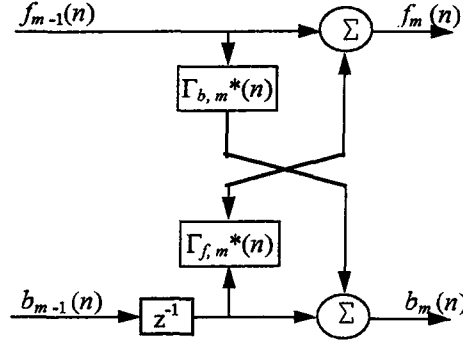
biçimindeki derece güncelleme özyinelemeleri elde edilir.

(8.33) ve (8.35) özyinelemeleri, kafes görünümüne sahip, Şekil 8.3 (a)'daki işaret-akış diyagramında gösterilmiştir.

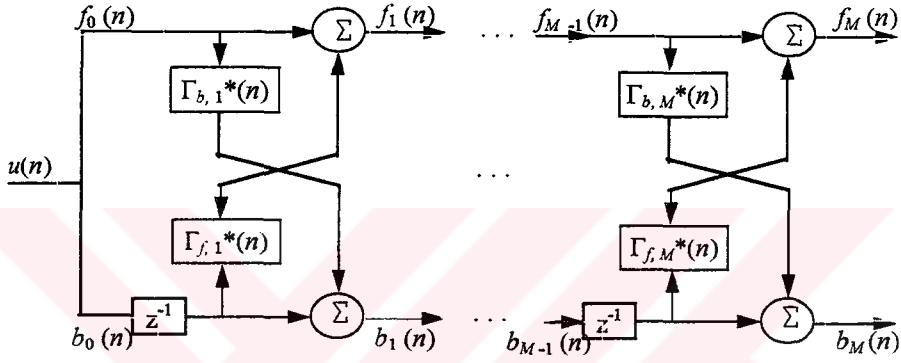
Öngörü derecesi m , değişken bir parametredir ve 0, 1, ..., M değerlerini alır, M burada öngörü derecesinin son değeridir. $m = 0$ olduğu zaman, giriş verisi üzerine işlenecek hiç bir öngörü yoktur. Bu ilk koşul,

$$f_0(n) = b_0(n) = u(n) \quad (8.36)$$

başlangıç değerine karşılık gelir. Öngörü derecesi m , sıfırdan son değeri M e kadar değişeceği için, Şekil 8.3 (b)'de gösterilen Çok Modüllü En Küçük Kareler Kafes Öngörücüsü elde edilir. Bu yapıdaki modül sayısı, öngörü derecesinin son değeri olan M e eşittir.



(a)



(b)

Şekil 8.3 (a) Tipik Bir En Küçük Kareler Kafes Modülü
(b) Çok Modüllü En Küçük Kareler Kafes Öngörücüsü

8.1.3. Ardıl Öngörü Hata Karelerinin Ağırlıklı Toplamının Derece Güncelleme Özyinelemeleri

(8.19) ve (8.22) eşitlikleri sırası ile, ileri ardıl öngörü hata kareleri $F_m(n)$ ' nin ve geri ardıl öngörü hata kareleri $B_m(n)$ ' nin derece güncelleme özyinelemelerini gösterir. (8.23) eşitliğinde $\Delta_{m-1}(n)$ ile $\Delta'_{m-1}(n)$ arasındaki ilişki kullanılırsa,

$$F_m(n) = F_{m-1}(n) - \frac{|\Delta_{m-1}(n)|^2}{B_{m-1}(n-1)} \quad (8.37)$$

ve

$$B_m(n) = B_{m-1}(n-1) - \frac{|\Delta_{m-1}(n)|^2}{F_{m-1}(n)} \quad (8.38)$$

eşitlikleri elde edilir.

(8.33), (8.35), (8.37) ve (8.38) eşitlikleri, en küçük kareler kafes öngörücüsünün temel derece güncelleme özyinelemeleridir.

8.1.4. Dönüşüm Faktörü $\gamma_m(n-1)$ ' in Derece Güncelleme Özyinelemeleri

RLSL algoritmasının derece güncelleme özyinelemelerini tamamlamak için gerekli bir başka özyineleme de, gecikmiş Kestirim Hatası veya Dönüşüm Faktörü $\gamma_m(n-1)$ ' dir.

Kestirim hatası $\gamma_m(n-1)$, tap giriş vektörü $\mathbf{u}_m(n-1)$ ' nin, tap ağırlık vektörü kazanç vektörü $\mathbf{k}_m(n-1)$ ' e eşit olan bir enine süzgece uygulanması ile elde edilir. Kazanç vektörü $\mathbf{k}_m(n-1)$ ' nin derece güncelleme özyinelemesi,

$$\mathbf{k}_m(n-1) = \begin{bmatrix} \mathbf{k}_{m-1}(n-1) \\ 0 \end{bmatrix} + \frac{b_{m-1}(n-1)}{B_{m-1}(n-1)} \mathbf{c}_{m-1}(n-1), \quad m = 1, 2, \dots, M \quad (8.39)$$

biçimindedir (Haykin, 1991). Bu eşitliğe Hermitian işlemi uygulanıp, her iki tarafı $\mathbf{u}_m(n-1)$ ile ardıl çarpılırsa,

$$\begin{aligned} \mathbf{k}_m^H(n-1) \mathbf{u}_m(n-1) &= [\mathbf{k}_{m-1}^H(n-1), 0] \mathbf{u}_m(n-1) \\ &+ \frac{b_{m-1}^*(n-1)}{B_{m-1}(n-1)} \mathbf{c}_{m-1}^H(n-1) \mathbf{u}_m(n-1) \end{aligned} \quad (8.40)$$

eld edilir. Şu tanım yapılırsa,

$$\mathbf{k}_m^H(n-1) \mathbf{u}_m(n-1) = 1 - \gamma_m(n-1) \quad (8.41)$$

ve yerine konulursa,

$$\begin{aligned}
 [\mathbf{k}_{m-1}^H(n-1), 0] \mathbf{u}_m(n-1) &= [\mathbf{k}_{m-1}^H(n-1), 0] \begin{bmatrix} \mathbf{u}_{m-1}(n-1) \\ u(n-m) \end{bmatrix} \\
 &= \mathbf{k}_{m-1}^H(n-1) \mathbf{u}_{m-1}(n-1) \\
 &= 1 - \gamma_{m-1}(n-1)
 \end{aligned} \tag{8.42}$$

elde edilir.

(8.3) eşitliğindeki geri ardıl öngörü hatası ifadesinden yararlanılarak,

$$\mathbf{c}_{m-1}^H(n-1) \mathbf{u}_m(n-1) = b_{m-1}(n-1) \tag{8.43}$$

yazılabilir. (8.41) ve (8.43) eşitlikleri (8.40) eşitliğinde yerine konulursa,

$$\gamma_m(n-1) = \gamma_{m-1}(n-1) - \frac{|b_{m-1}(n-1)|^2}{B_{m-1}(n-1)}, \quad m = 1, 2, \dots, M \tag{8.44}$$

derece güncelleme özyinelemelerinin sonucusu olan, $\gamma_m(n-1)$ ' nin derece güncelleme özyinelemesi elde edilmiş olur.

8.2. Zaman Güncelleme Özyinelemeleri

Kısım 8.1' de, ileri ve geri ardıl öngörü hataları ve onların ağırlıklı karelerinin toplamlarının, başlangıç durumu $m = 0$ ' dan başlayan özyinelemeleri çıkarıldı. Bununla birlikte bu özyinelemeleri zamanda uyarlamak için, derece güncelleme özyinelemelerinin hesaplanmasında kullanılan $\Delta_{m-1}(n)$ parametresinin zaman güncelleme özyinelemesinin çıkarılması gereklidir.

$\mathbf{a}_{m-1}(n-1)$, $(m-1)$. dereceden ileri öngörü hata süzgecinin $n-1$ anındaki $m \times 1$ boyutlu tap ağırlık vektörüdür. $\mathbf{a}_{m-1}(n-1)$ vektörünün ilk elemanı birim değere eşit olduğu için, $\Delta_{m-1}(n)$ şu biçimde ifade edilir [Bkz. (8.23) ve (8.25) eşitlikleri]:

$$\Delta_{m-1}(n) = \begin{bmatrix} \Delta_{m-1}(n), & \mathbf{0}_{m-1}^T, & B_{m-1}(n-1) \end{bmatrix} \begin{bmatrix} \mathbf{a}_{m-1}(n-1) \\ 0 \end{bmatrix} \quad (8.45)$$

(8.16) eşitliğinin her iki tarafına Hermitian işlemi uygulanırsa ve ilişki matrisi $\Phi_{m+1}(n)$ ' nin Hermitian özelliği (yani, $\Phi^H = \Phi$) ve (8.23) eşitliğindeki ilişki kullanılırsa,

$$\begin{bmatrix} 0, & \mathbf{c}_{m-1}^H(n-1) \end{bmatrix} \Phi_{m+1}(n) = \begin{bmatrix} \Delta_{m-1}(n), & \mathbf{0}_{m-1}^T, & B_{m-1}(n-1) \end{bmatrix} \quad (8.46)$$

elde edilir. (8.46) eşitliği (8.45) eşitliğinde yerine konularak,

$$\Delta_{m-1}(n) = \begin{bmatrix} 0, & \mathbf{c}_{m-1}^H(n-1) \end{bmatrix} \Phi_{m+1}(n) \begin{bmatrix} \mathbf{a}_{m-1}(n-1) \\ 0 \end{bmatrix} \quad (8.47)$$

yazılabilir.

(7.12) eşitliğinden yararlanılarak ilişki matrisi $\Phi_{m+1}(n)$ ' nin zaman güncellemesi,

$$\Phi_{m+1}(n) = \lambda \Phi_{m+1}(n-1) + \mathbf{u}_{m+1}(n) \mathbf{u}_{m+1}^H(n) \quad (8.48)$$

biçiminde yazılabilir. Bu ilişki (8.47) eşitliğinde kullanılarak

$$\begin{aligned} \Delta_{m-1}(n) &= \lambda \begin{bmatrix} 0, & \mathbf{c}_{m-1}^H(n-1) \end{bmatrix} \Phi_{m+1}(n-1) \begin{bmatrix} \mathbf{a}_{m-1}(n-1) \\ 0 \end{bmatrix} \\ &\quad + \begin{bmatrix} 0, & \mathbf{c}_{m-1}^H(n-1) \end{bmatrix} \mathbf{u}_{m+1}(n) \mathbf{u}_{m+1}^H(n) \begin{bmatrix} \mathbf{a}_{m-1}(n-1) \\ 0 \end{bmatrix} \end{aligned} \quad (8.49)$$

yazılabilir. (8.31) eşitliğinden ve ileri öncül öngörü hatasının tanımından faydalanılarak

$$\begin{aligned}
\mathbf{u}_{m-1}^H(n) \begin{bmatrix} \mathbf{a}_{m-1}(n-1) \\ 0 \end{bmatrix} &= [\mathbf{u}_m^H(n), u^*(n-m)] \begin{bmatrix} \mathbf{a}_{m-1}(n-1) \\ 0 \end{bmatrix} \\
&= \mathbf{u}_m^H(n) \mathbf{a}_{m-1}(n-1) \\
&= \eta_{m-1}^*(n)
\end{aligned} \tag{8.50}$$

eşitliği yazılabilir. Aynı şekilde (8.32) eşitliğinden ve geri öngörü hatasının tanımından yararlanılarak,

$$\begin{aligned}
[0, \mathbf{c}_{m-1}^H(n-1)] \mathbf{u}_{m+1}(n) &= [0, \mathbf{c}_{m-1}^H(n-1)] \begin{bmatrix} u(n) \\ \mathbf{u}_m(n-1) \end{bmatrix} \\
&= \mathbf{c}_{m-1}^H(n-1) \mathbf{u}_m(n-1) \\
&= b_{m-1}(n-1)
\end{aligned} \tag{8.51}$$

eşitliği yazılabilir.

(8.10) eşitliğinde n yerine $n-1$ konularak,

$$\Phi_{m+1}(n-1) \begin{bmatrix} \mathbf{a}_{m-1}(n-1) \\ 0 \end{bmatrix} = \begin{bmatrix} F_{m-1}(n-1) \\ 0_{m-1} \\ \Delta_{m-1}(n-1) \end{bmatrix} \tag{8.52}$$

yazılabilir. (8.52) eşitliği ve geri öngörü hata süzgeci aittap ağırlık vektörü $\mathbf{c}_{m-1}(n-1)$ 'nin son elemanının birim değere eşit olduğu göz önüne alınırsa, (8.49) eşitliğinin sağ tarafındaki ilk terim, λ hariç, şu şekilde yazılabilir:

$$\begin{aligned}
[0, \mathbf{c}_{m-1}^H(n-1)] \Phi_{m+1}(n-1) \begin{bmatrix} \mathbf{a}_{m-1}(n-1) \\ 0 \end{bmatrix} &= [0, \mathbf{c}_{m-1}^H(n-1)] \begin{bmatrix} F_{m-1}(n-1) \\ 0_{m-1} \\ \Delta_{m-1}(n-1) \end{bmatrix} \\
&= \Delta_{m-1}(n-1)
\end{aligned} \tag{8.53}$$

(8.50), (8.51) ve (8.53) eşitlikleri (8.49) eşitliğinde yerlerine konulursa $\Delta_{m-1}(n)$ ' nin zaman güncelleme özyinelemesi

$$\Delta_{m-1}(n) = \lambda \Delta_{m-1}(n-1) + b_{m-1}(n-1) \eta_{m-1}^*(n) \quad (8.54)$$

elde edilir.

İleri öncül öngörü hatası $\eta_{m-1}(n)$ ile ileri ardıl öngörü hatası $f_{m-1}(n)$ arasında şu ilişki vardır (Haykin, 1991):

$$\eta_{m-1}(n) = \frac{f_{m-1}(n)}{\gamma_{m-1}(n-1)} \quad (8.55)$$

(8.55) eşitliği (8.54) eşitliğinde kullanılırsa, zaman güncelleme özyinelemesi

$$\Delta_{m-1}(n) = \lambda \Delta_{m-1}(n-1) + \frac{b_{m-1}(n-1) f_{m-1}^*(n)}{\gamma_{m-1}(n-1)} \quad (8.56)$$

biçimine dönüşür. (8.56) eşitliğindeki düzeltme terimi, dönüşüm faktörü $\gamma_{m-1}(n-1)$ ' nin tersi kadar kuvvetlendirilir. Bu parametre, RLSL algoritmasının, giriş verisindeki ani değişikliklere hızlıca uyarlanmasını sağlar. Kuvvetlendirme faktörü $1/\gamma_{m-1}(n-1)$ hariç, $\Delta_{m-1}(n)$ parametresinin zaman güncellemesi gerçekte, gecikmiş geri öngörü hatası $b_{m-1}(n-1)$ ile ileri öngörü hatası $f_{m-1}(n)$ arasındaki çapraz ilişkinin zaman ortalamasıdır.

Dönüşüm faktörü $\gamma_{m-1}(n-1)$, gerçel değerlidir ve sıfır ile bir arasında sınırlandırılmıştır. $\gamma_{m-1}(n-1)$ sıfır olduğu zaman, (8.56) eşitliğinin özyinelemesi duracaktır, yani bu durumda ilişki matrisi $\Phi_{m+1}(n)$ ' nin tersinin alınamayacağı ortaya çıkar.

8.3. RLSL Algoritmasının Özeti

RLSL algoritmasını içeren derece ve zaman güncelleme özyinelemelerinin tamamı, Tablo 8.1' de verilmiştir. Bu tablo, (8.56), (8.26), (8.27), (8.33), (8.35), (8.37), (8.38) ve (8.44) eşitliklerinin özyinelemelerini ve ilişkilerini yazılan sırada olmak üzere kapsar.

Tablo 8.1 Ardıl Kestirim Hatalarını Kullanan RLSL Algoritmasının Özeti

$n = 1, 2, \dots$ ve $m = 1, 2, \dots, M$ olmak üzere aşağıdakiler sırasıyla hesaplanır:

$$\Delta_{m-1}(n) = \lambda \Delta_{m-1}(n-1) + \frac{b_{m-1}(n-1)f_{m-1}^*(n)}{\gamma_{m-1}(n-1)}$$

$$\Gamma_{f,m}(n) = -\frac{\Delta_{m-1}(n)}{B_{m-1}(n-1)}$$

$$\Gamma_{b,m}(n) = -\frac{\Delta_{m-1}^*(n)}{F_{m-1}(n)}$$

$$f_m(n) = f_{m-1}(n) + \Gamma_{f,m}^*(n)b_{m-1}(n-1)$$

$$b_m(n) = b_{m-1}(n-1) + \Gamma_{b,m}^*(n)f_{m-1}(n)$$

$$F_m(n) = F_{m-1}(n) - \frac{|\Delta_{m-1}(n)|^2}{B_{m-1}(n-1)}$$

$$B_m(n) = B_{m-1}(n-1) - \frac{|\Delta_{m-1}(n)|^2}{F_{m-1}(n)}$$

$$\gamma_m(n-1) = \gamma_{m-1}(n-1) - \frac{|b_{m-1}(n-1)|^2}{B_{m-1}(n-1)}$$

RLSL algoritmasını koşturmak için, $m = 0$ ' başlangıç durumu ile başlanacaktır. (8.30) ve (8.34) eşitliklerinden yararlanılarak

$$f_0(n) = b_0(n) = u(n)$$

yazılabilir. İleri ardıl öngörü hata kareleri ağırlıklı toplamı $F_m(n)$ ' nin zaman güncelleme özyinelemesi,

$$F_m(n) = \lambda F_m(n-1) + f_m(n) \eta_m^*(n) \quad (8.57)$$

biçimindedir. $f_m(n)$ öngörü derecesi m için ileri ardıl öngörü hatası, $\eta_m(n)$ ise ileri öncül öngörü hatasıdır. $m = 0$ olduğu zaman,

$$f_0(n) = \eta_0(n) = u(n)$$

olduğundan bu durumda (8.57) eşitliğinin $m = 0$ için zaman güncelleme özyinelemesi,

$$F_0(n) = \lambda F_0(n-1) + |u(n)|^2 \quad (8.58)$$

biçiminde olacaktır.

Benzer biçimde geri ardıl öngörü hata kareleri ağırlıklı toplamı $B_m(n)$ ' nin, $m = 0$ için zaman güncelleme özyinelemesi,

$$B_0(n) = \lambda B_0(n-1) + |u(n)|^2 \quad (8.59)$$

biçiminde gösterilebilir.

Dönüşüm faktörü $\gamma_{m-1}(n-1)$, sıfır ile bir arasında sınırlandırılmış olduğundan, bu parametrenin 0. derece değerinin aşağıdaki gibi seçilmesi mantıklıdır:

$$\gamma_0(n-1) = 1 \quad (8.60)$$

$n = 0$ için aşağıdaki koşullandırmalar yapılırsa, RLSL algoritmasının koşullandırılması tamamlanmış olur:

$$\Delta_{m-1}(0) = 0 \quad (8.61)$$

$$F_0(0) = B_0(0) = \delta \quad (8.62)$$

Burada δ , küçük pozitif sabit bir sayıdır ve özyinelemelerde sıfıra bölme olmaması için yani ilişki matrisi $\Phi_{m+1}(n)$ 'nin tekil çözümünün olmaması için kullanılır.

Tablo 8.2 Ardıl Kestirim Hatalarını Kullanan
RLSL Algoritmasının Koşullandırılmasının Özeti

1. Algoritma $n = 0$ anında aşağıdaki değerlerle koşullandırılır:

$$\Delta_{m-1}(0) = 0$$

$$F_{m-1}(0) = \delta$$

$$B_{m-1}(0) = \delta$$

$$\gamma_0(0) = 1$$

$\delta =$ Küçük pozitif sabit bir sayı

2. Her $n \geq 1$ anında, 0. derece değişkenleri şu şekilde elde edilir:

$$f_0(n) = b_0(n) = u(n)$$

$$F_0(n) = B_0(n) = \lambda F_0(n-1) + |u(n)|^2$$

$$\gamma_0(n-1) = 1$$

Uyarı: Önpencerelenmiş veri için, giriş $u(n)$ ve istenen yanıt $d(n)$, $n \leq 0$ için sıfırdır.

RLSL algoritmasının koşullandırılmasında kullanılan hesaplamaların özeti Tablo 8.2' de verilmiştir.

BÖLÜM 9. UYARLAMALI KAFES ALGORİTMASI

KULLANARAK KANAL ÖZDEŞLEME

9.1. Özdeşleme Kavramı

Uyarlamalı bir alıcının tasarlanmasıyla ilgili yaklaşımlardan biri olan “ kanal özdeşleme” de amaç, önce ilgili kanal parametrelerinin kestirilmesi ve daha sonra bunların kullanılmasıyla alıcının kendisinin iletim ortamına uyarlanması sağlanmasıdır. Burada, bilinmeyen kanalı özdeşlemek üzere, bir model kullanılmaktadır. Bu modelin girişi, kanal giriş işareti olmak üzere elde edilen çıkış işareti ile kanal çıkışı arasındaki fark bir uyarlamalı algoritmada kullanılarak, kanal ile model arasındaki sapma giderilmeye çalışılmaktadır. Böylece modelin, bilinmeyen kanalı özdeşlemesi sağlanmış olur. Sistem özdeşleme yapısı Şekil 9.1’ de görülmektedir. Burada,

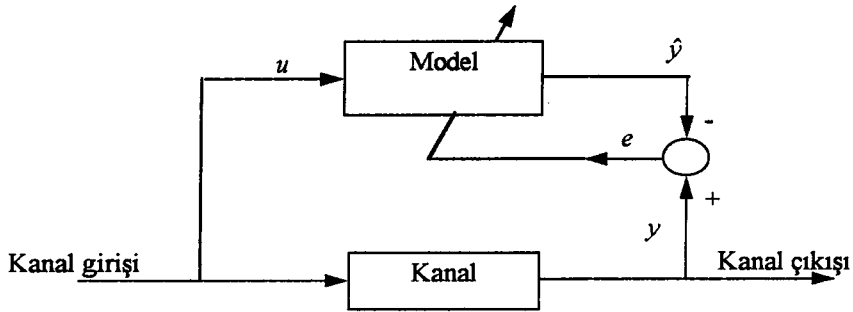
u ; kanal girişi

$\hat{y}$; model çıkışı

y ; kanal çıkışı

$e = y - \hat{y}$; kestirim hatası

biçiminde verilmektedir.



Şekil 9.1 Kanal Özdeşleme Yapısı

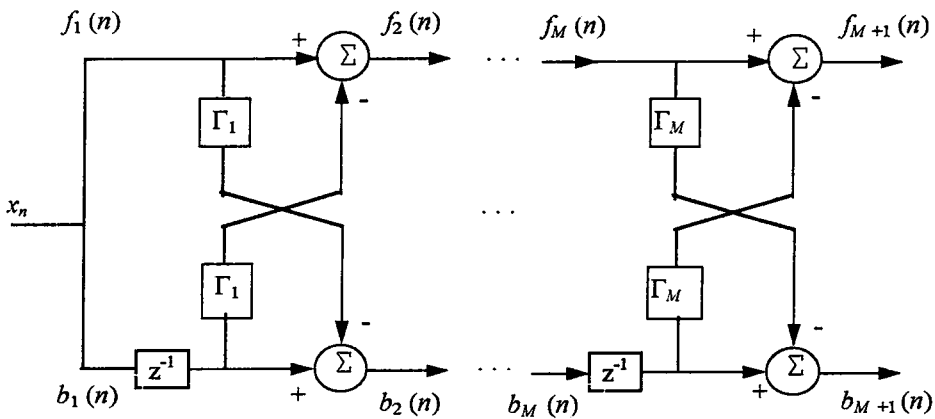
Bilinmeyen bir kanalın parametrelerinin kestirimi için bir çok farklı uyarlamalı süzgeçleme algoritmaları kullanılabilir.

Bu problemin çözümünde kullanılan algoritmalarından biri de uyarlamalı en küçük kareler kafes süzgeç yapısıdır. Bu bölümde, önce bu algoritma incelenmekte ve örnek bir kanalın özdeşlenmesi işlemi bilgisayar simülasyonu ile gerçekleştirilmektedir.

9.2. Kanal Özdeşleme Probleminde Uyarlamalı Kafes Süzgeç Kullanılması

Kanal özdeşleme probleminde kullanacağımız uyarlamalı kafes (AL) algoritması (Satorius ve diğerleri, 1979), giriş örnek değerlerini birbirine dik bir işaret bileşenleri kümesi olarak üretir. Bu bileşenlerin üretimi, önceki bölümlerde verilen Gram-Schmidt dikleştirme yöntemi ile gerçekleştirilir. Uyarlamalı kafes algoritmaları işaret işleme, gürültü giderme ve parametre kestirimi gibi pek çok alanda kullanılmak üzere oluşturulmuştur.

M katlı kafes yapısı Şekil 9.2' de verilmiştir. $u(n)$ dizisi süzgeç yapının giriş dizisi olmak üzere, bu algoritma ile bu diziden birbirlerine dik (ilişkisiz) $b_m(n)$ ve $f_m(n)$ işaret dizileri oluşturulur.



Şekil 9.2 M Katlı Temel Kafes Yapısı

$$b_1(n) = f_1(n) = u(n) \quad (9.1)$$

$$f_{m+1}(n) = f_m(n) - \Gamma_m b_m(n-1) \quad (9.2)$$

$$b_{m+1}(n) = b_m(n-1) - \Gamma_m f_m(n) \quad (9.3)$$

Buradaki $b_m(n)$ ve $f_m(n)$, daha önceki bölümlerde açıklandığı gibi geri ve ileri yönde öngörü hatalarıdır. Γ_m ' de yansıma katsayısıdır ve şöyle tanımlanır (Makhoul,1977):

$$\Gamma_m = \frac{2E[f_m(n)b_m(n-1)]}{E[f_m^2(n)] + E[b_m^2(n-1)]}, \quad 1 \leq m \leq M \quad (9.4)$$

Ayrıca, ileri ve geri yönde öngörü hataları şöyle de tanımlanabilir:

$$b_{m+1}(n) = u_{n-m} - \sum_{j=1}^m w_j^{(m)} u_{n-m+j}, \quad 1 \leq m \leq M \quad (9.5)$$

$$f_{m+1}(n) = u_n - \sum_{j=1}^m w_j^{(m)} u_{n-j}, \quad 1 \leq m \leq M \quad (9.6)$$

Buradaki $w_j^{(m)}$, aşağıdaki normal denklemden çıkarılan 1-adımlı öngörü katsayılarıdır:

$$\sum_{j=1}^m w_j^{(m)} \phi(p-j) = \phi(p), \quad 1 \leq p \leq m \quad (9.7)$$

Burada $\phi(p)$, giriş işaretinin özilişki fonksiyonunu gösterir.

En küçük karesel ortalama hata kriterine göre diklik koşulu,

$$E[u_{n-j}b_{m+1}(n)] = 0, \quad 0 \leq j \leq m-1 \quad (9.8)$$

biçiminde yazılabilir. (9.5) ve (9.8) eşitliklerinden,

$$E[b_m(n)b_j(n)] = \begin{cases} 0, & j \neq m \\ \beta_m, & j = m = 2, 3, \dots, M+1 \\ E[u_n^2], & j = m = 1 \end{cases} \quad (9.9)$$

sonuçları çıkarılabilir. Buradaki β_{m+1} , m - noktalı 1-adımlı doğrusal öngörü süzgecinin en küçük karesel ortalama hatasını temsil eder. u_{n-m} ile $b_m(n)$ arasındaki dönüşüm matris formunda

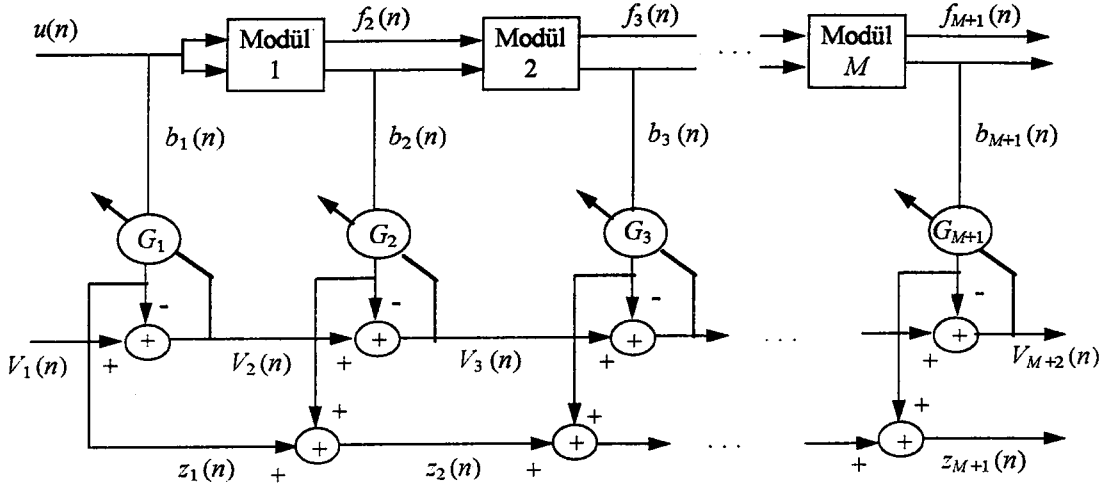
$$\mathbf{B}_n = \mathbf{L}\mathbf{U}_n \quad (9.10)$$

biçiminde yazılabilir. $\mathbf{L}$ matrisi,

$$\mathbf{L} = \begin{bmatrix} 1 & & & & 0 \\ -w_1(1) & 1 & & & \\ -w_2(2) & -w_1(2) & 1 & & \\ \vdots & \vdots & \vdots & \ddots & \\ -w_M(M) & -w_{M-1}(M) & -w_{M-2}(M) & \dots & 1 \end{bmatrix} \quad (9.11)$$

şeklinde bir üçgen matris olarak tanımlanır.

Şekil 9.3' de anlatılan bu temel özelliklere dayandırılarak oluşturulan uyarlamalı kafes özdeşleyici görülmektedir. Burada $u(n)$, kanalın giriş dizisini ifade etmektedir. Sistem özdeşleme yapısından hatırlanacağı gibi (bkz. Şekil 9.1) hem kanalın hem de uyarlama süzgecinin girişi aynı dizidir. $V_1(n)$ değeri ise, istenen yanıt yani kanal çıkışı $y(n)$ değeri olmalıdır.



Şekil 9.3 Uyarlamalı Kafes Özdeşleyici

Şekil 9.3' e göre, $G_1, G_2, \dots, G_{M+1}$ değerleri ayarlanabilir tap-kazançlarını göstermek üzere,

$$z_1(n) = G_1(n)b_1(n) \quad (9.12)$$

$$z_m(n) = z_{m-1}(n) + G_m(n)b_m(n), \quad 2 \leq m \leq M+1$$

yazılabilir.

Uyarlamalı süzgeç çıkışında istenen yanıtın kestirimi,

$$\hat{y}(n) = z_{M+1}(n) \quad (9.13)$$

olarak bulunur.

Özdeşleyici çıkışı ise,

$$V_{m+1}(n) = y(n) - z_m(n), \quad 1 \leq m \leq M+1 \quad (9.14)$$

olarak belirlenir.

Uyarlamalı kafes kanal özdeşleyici yapısı açıklığa kavuşturulduktan sonra, yapılan bilgisayar simülasyonunda kullanılan algoritma yapısını verelim:

$$f^{(1)}(n) = b^{(1)}(n) = u(n)$$

$$f^{(m+1)}(n) = f^{(m)}(n) - \Gamma_f^{(m+1)}(n)b^{(m)}(n-1)$$

$$b^{(m+1)}(n) = b^{(m)}(n-1) - \Gamma_b^{(m+1)}(n)f^{(m)}(n)$$

$$\Gamma_f^{(m+1)}(n+1) = \Gamma_f^{(m+1)}(n) + 2\delta_f^{(m+1)}(n)b^{(m)}(n-1)f^{(m+1)}(n)$$

$$\Gamma_b^{(m+1)}(n+1) = \Gamma_b^{(m+1)}(n) + 2\delta_b^{(m+1)}(n)f^{(m)}(n)b^{(m+1)}(n-1)$$

$$\delta_f^{(m+1)}(n) = \frac{\alpha}{(\sigma_f^{(m+1)}(n))^2}$$

$$\delta_b^{(m+1)}(n) = \frac{\alpha}{(\sigma_b^{(m+1)}(n))^2}$$

$$(\sigma_f^{(m+1)}(n+1))^2 = (1-\alpha)(\sigma_f^{(m+1)}(n))^2 + \alpha b^{(m)}(n-1)b^{(m)}(n-1)$$

$$(\sigma_b^{(m+1)}(n+1))^2 = (1-\alpha)(\sigma_b^{(m+1)}(n))^2 + \alpha f^{(m)}(n)f^{(m)}(n)$$

Buradaki α , adım uzunluğunu temsil eder ve $0 < \alpha < 1$ aralığında değer alır.

Ayarlanabilir tap-kazançları için, Şekil 9.3' den de çıkarılabilecek olan aşağıdaki formüller kullanılmıştır:

$$G^{(m+1)}(n+1) = G^{(m+1)}(n) + 2\delta_g^{(m+1)}(n)b^{(m+1)}(n)V^{(m+1)}(n)$$

$$\delta_g^{(m+1)}(n) = \frac{\alpha}{(\sigma_g^{(m+1)}(n))^2}$$

$$(\sigma_g^{(m+1)}(n+1))^2 = (1-\alpha)(\sigma_g^{(m+1)}(n))^2 + \alpha(V^{(m+1)}(n))^2$$

$$V^{(m+1)}(n) = V^{(m)}(n) - G^{(m+1)}(n)b^{(m+1)}(n)$$

Bu programda, öncelikle $u(n)$ girişlerine dik olan $b_m(n)$ geri öngörü hataları hesaplanır. Daha sonra, AL (Uyarlamalı kafes) algoritması bu $b_m(n)$ değerlerini, $G_m(n)$ tap kazançlarıyla değiştirerek, referans işaretiyle faklarını alıp, birbirleriyle dik $V_m(n)$ bileşenleri haline getirir. Değişen bu hata değerleri ile istenen değerler dizisinin kestirimi elde edilmiş olur.

Bu programda ilk koşullar şöyle seçilmektedir:

$$\Gamma_f^{(m+1)}(1) = 0$$

$$\Gamma_b^{(m+1)}(1) = 0$$

$$G^{(m)}(1) = 0$$

$$m = 0, 1, \dots, M$$

$$\sigma_f^{(m+1)}(1) = 1$$

$$\sigma_b^{(m+1)}(1) = 1$$

$$\sigma_g^{(m+1)}(1) = 1$$

9.3. Bilgisayar Simülasyonu

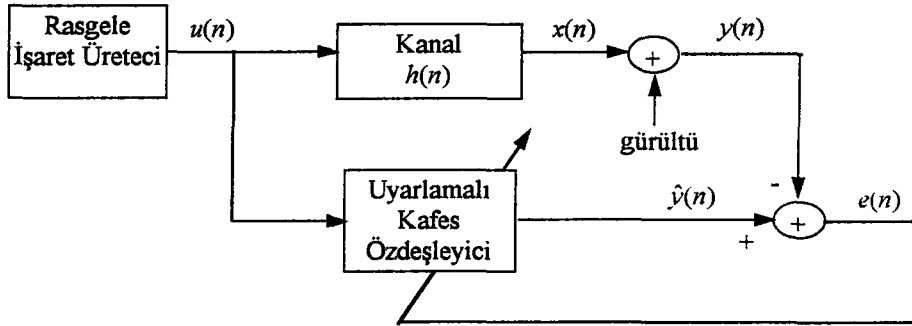
Yapılan bilgisayar programında, öncelikle işlenecek yani kanala girecek işaretlerin rasgele üretilmesi sağlanmıştır. Bu işaret $n = 1, 2, \dots, S$ anlarında $\{+1, -1\}$ değerlerinden oluşan bir dizi şeklindedir. Çok sayıda değer atandığından giriş dizisi yaklaşık olarak sıfır ortalamalı olmaktadır.

$$u(n) = \{u_n\}, \quad u_n = \{+1, -1\}$$

Daha sonra işaretin bir iletim kanalından geçtiği düşünülüp, örnek bir kanal modeli seçilmektedir. Kanalın birim örnek yanıtı,

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

biçiminde seçilmiştir (Haykin,1991)(Satorius ve diğerleri, 1981). Bu kanal basit bir doğrusal kanaldır. Burada, kanalın birim örnek yanıtındaki W değeri, tap girişlerinin ilişki matrisinin özdeğerlerini kontrol edebilen bir parametredir.



Şekil 9.4 Uyarlamalı Kafes Algoritması ile Gerçekleştirilen
Kanal Özdeşleyici Blok Diyagramı

Gerçek anlamda iletim kanallarında işarete rasgele bir gürültü bineceğinden, oluşturulan model gereğince, kanal çıkışında sıfır ortalamalı bir gürültü oluşturularak işarete eklenmektedir.

Bilgisayar programında gerçekleştirilen yapı Şekil 9.4' de verilmiştir. Hazırlanan bilgisayar programı Ek' de, sonuç grafik çıktıları ise Şekil 9.5, Şekil 9.6, Şekil 9.7, Şekil 9.8, Şekil 9.9, Şekil 9.10 ve Şekil 9.11' de verilmiştir. Bilgisayar programı çalıştırılırken kafes derecesi M , adım uzunluğu α , kanal parametresi W ve gürültünün değışintisi σ_{gur}^2 değıştirilerek algoritma sonuçlarının değışimi, karesel ortalama hatanın iterasyon sayısı ile değışim grafiğine (öğrenme eğrisi) göre gözlenmiştir.

9.3.1. Kafes Derecesinin Karesel Ortalama Hataya Etkisinin İncelenmesi

Kafes derecesi M in değışimlerine göre algoritma sonuçlarının değışimini incelemek üzere, diğere parametreler sabit tutulmuştur:

Adım uzunluğu $\alpha = 0,0015$

Kanal parametresi $W = 2,9$

Gürültünün değışintisi $\sigma_{gur}^2 = 0,001$

$M = 1$ için algoritma çalıştırıldığında, Şekil 9.5 (a)' dan görüldüğü gibi karesel ortalama hatada hiç bir azalma olmadığı yani modelimizin kanalı hiç bir şekilde özdeşleyemediği görülür.

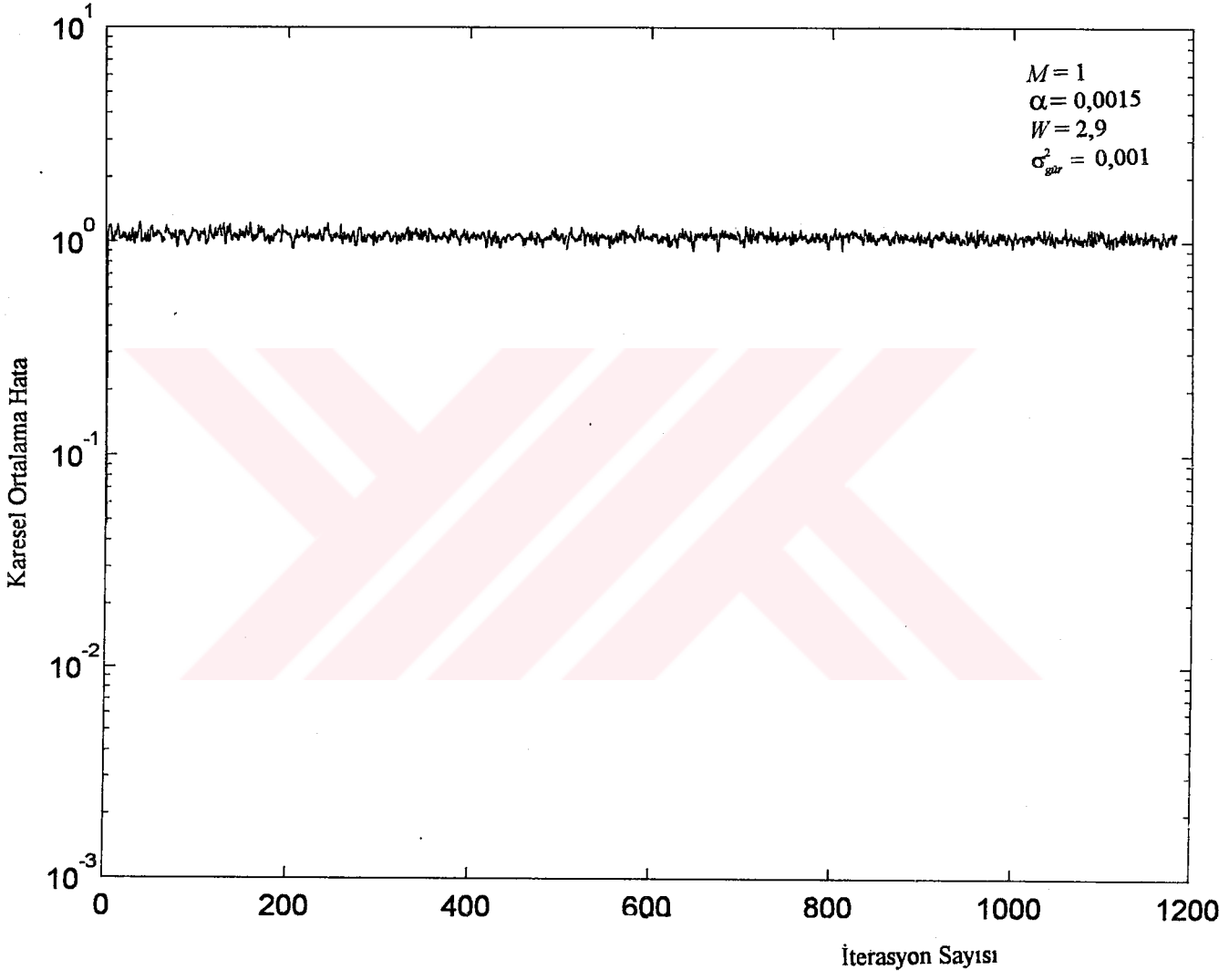
$M = 2$ için, Şekil 9.5 (b)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 800 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,740$ ' tır.

$M = 3$ için, Şekil 9.5 (c)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 780 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,00980$ ' dir. (Minimum karesel ortalama hata değeri yaklaşık olarak alınmıştır.)

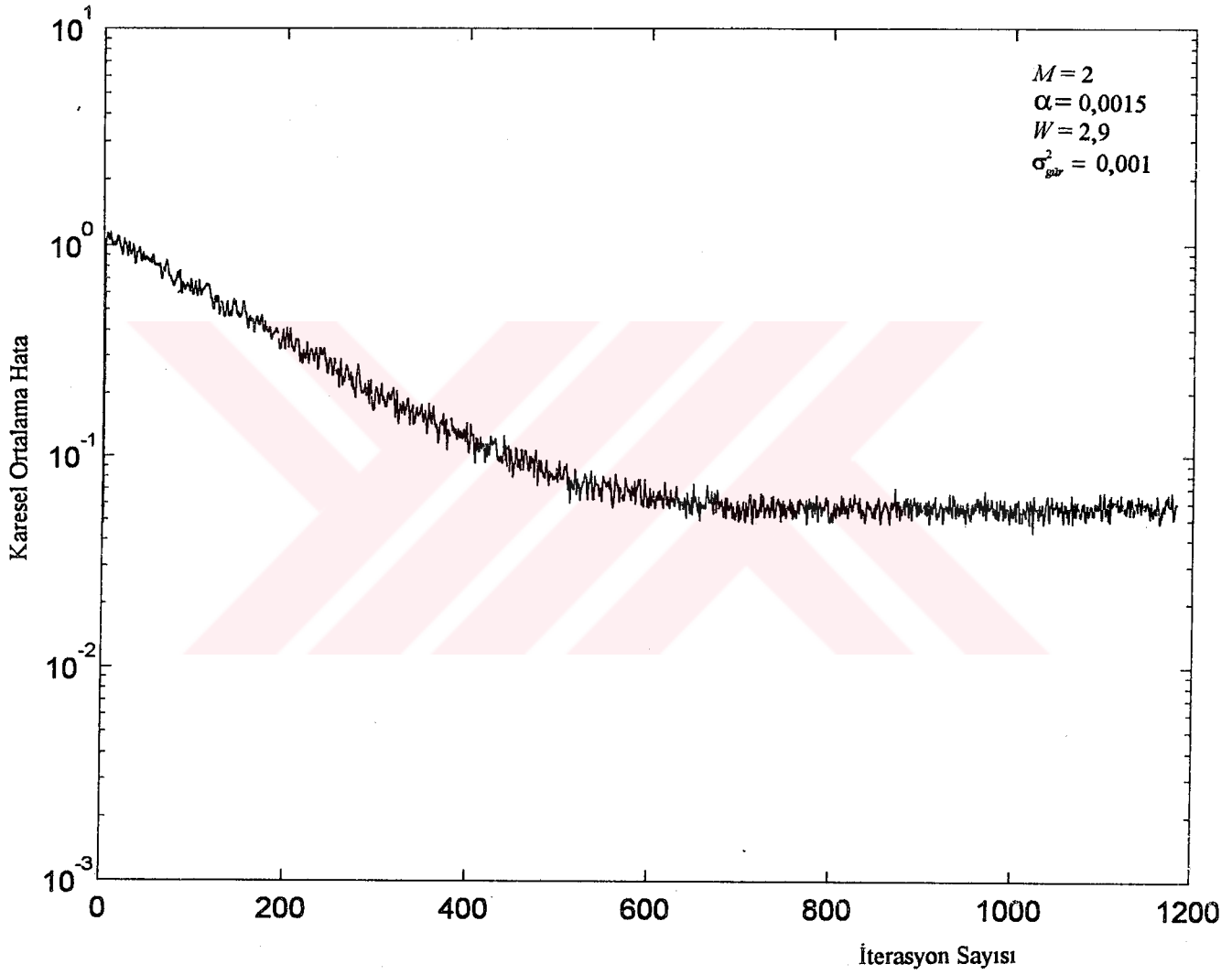
Son olarak, $M = 4$ için Şekil 9.5 (d) ve $M = 5$ için Şekil 9.5 (e) incelendiğinde, bu eğrilerin yakınsama hızlarının ve kararlı durumdaki minimum karesel ortalama hata değeri $M = 3$ değeri için çizdirilen Şekil 9.5 (c) ile aynı olduğu görülür.

$M = 2$ için çizdirilen Şekil 9.5 (b) ve $M = 3$ için çizdirilen Şekil 9.5 (c) karşılaştırıldığında, M değeri arttıkça yakınsama hızının arttığı ve kararlı durumda ulaşılan karesel ortalama hatanın azaldığı görülmektedir. Acaba belli bir kafes derecesi değerinden sonra, grafiklerde bir iyileşme olmaması normal midir? Bu sorunun yanıtı, daha farklı kanalların özdeşlenmeleri gözlenerek verilebilir.

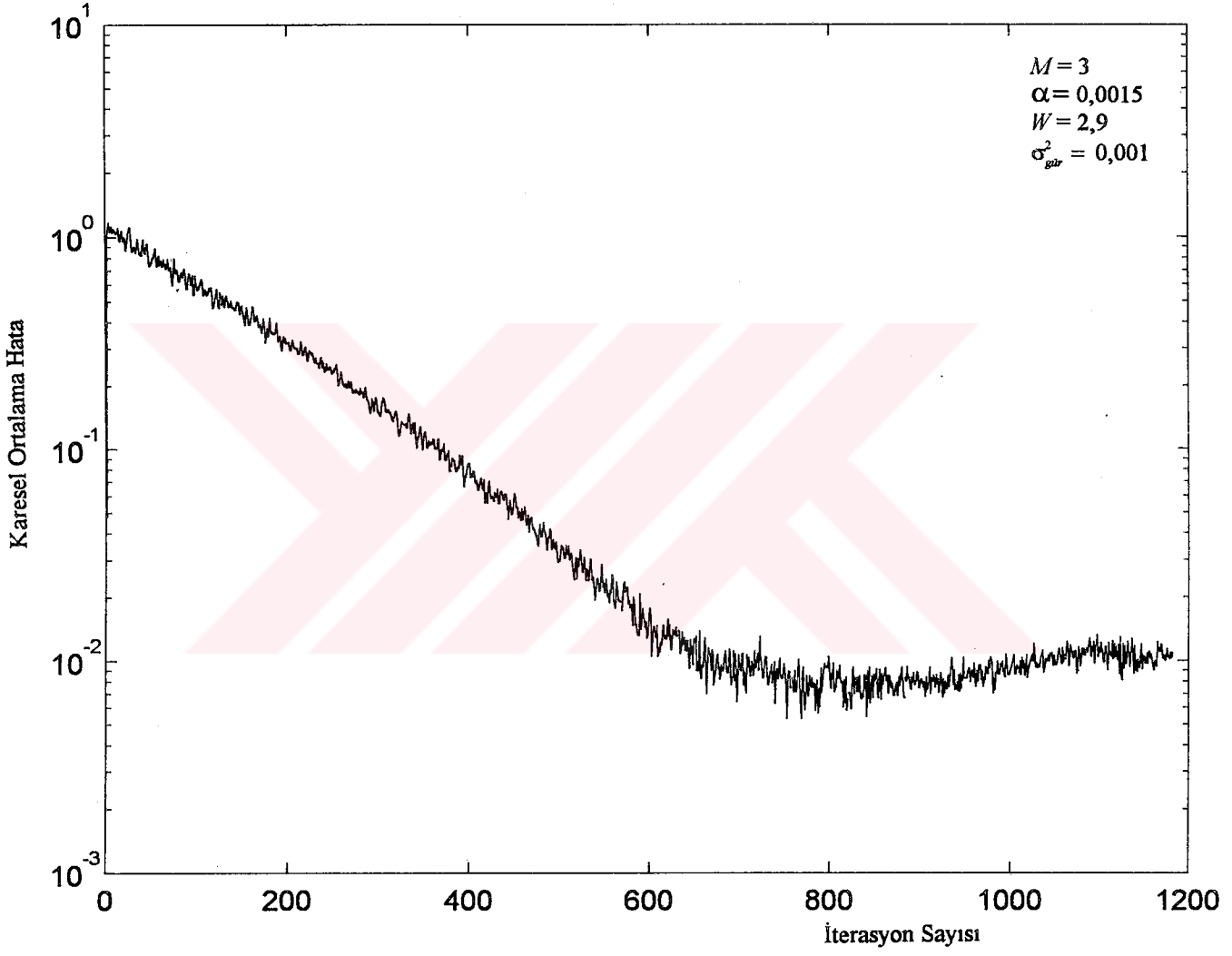




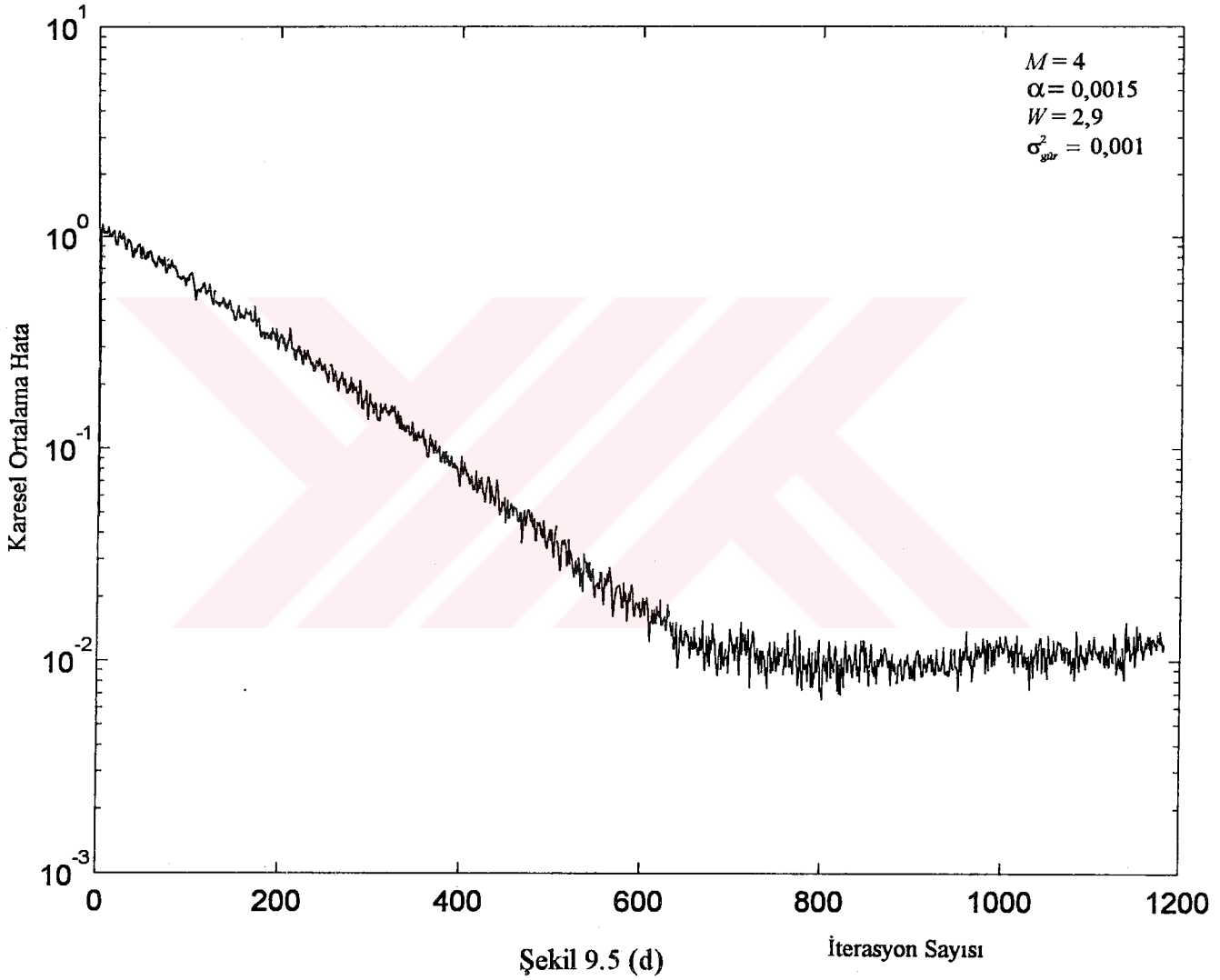
Şekil 9.5 (a)

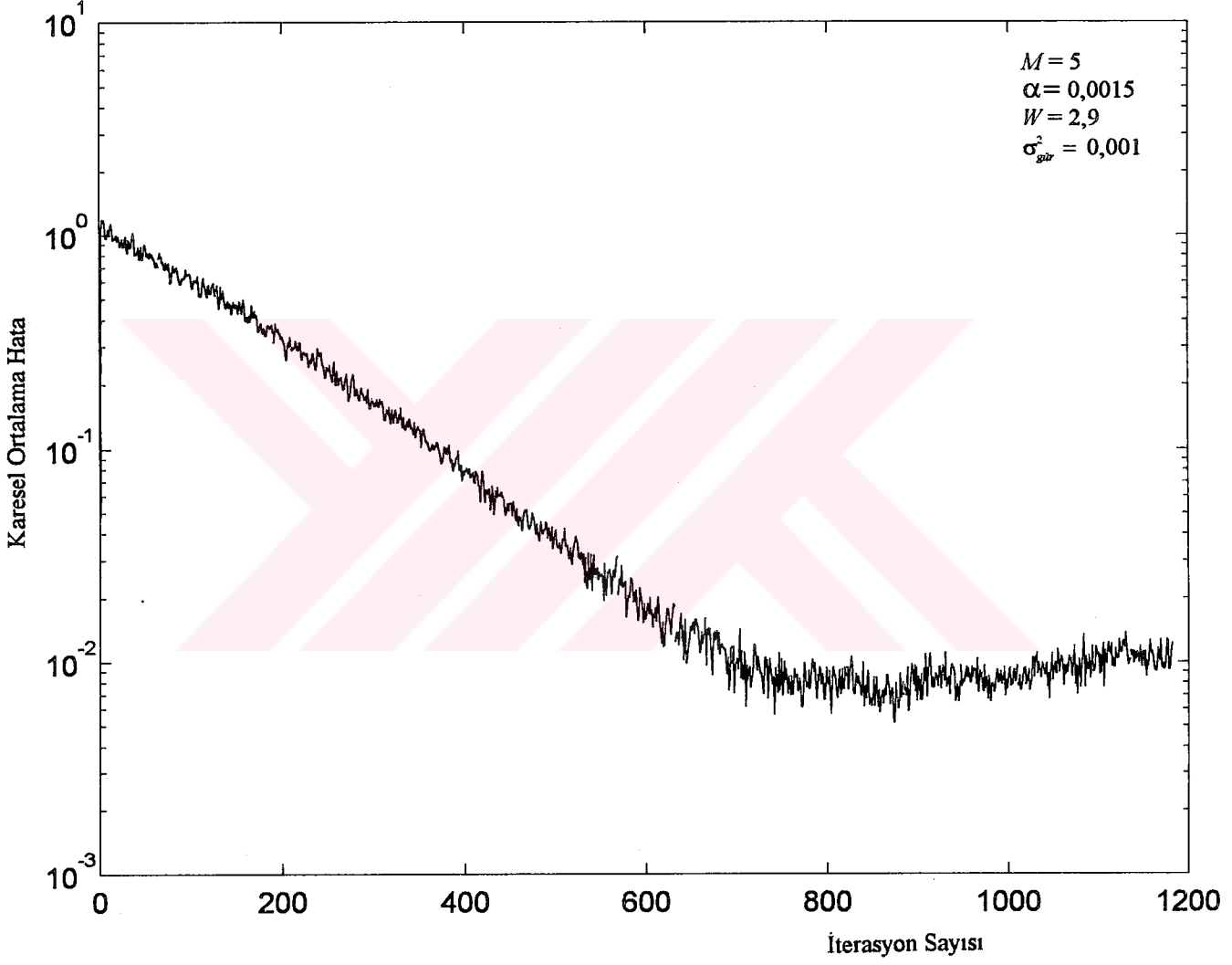


Şekil 9.5 (b)



Şekil 9.5 (c)





Şekil 9.5 (e)

Bu amaçla ilk olarak aşağıdaki gibi bir kanal modeli seçerek,

$$h(n) = \begin{cases} 0,3 & n = 1 \\ 0,2 & n = 2 \\ 1 & n = 3 \\ 0,2 & n = 4 \\ 0,3 & n = 5 \end{cases}$$

incelememize başladık.

$M = 2$ için algoritma çalıştırıldığında, Şekil 9.6 (a)' dan görüldüğü gibi karesel ortalama hatada hiç bir azalma olmadığı yani modelimizin kanalı hiç bir şekilde özdeşleyemediği görülür.

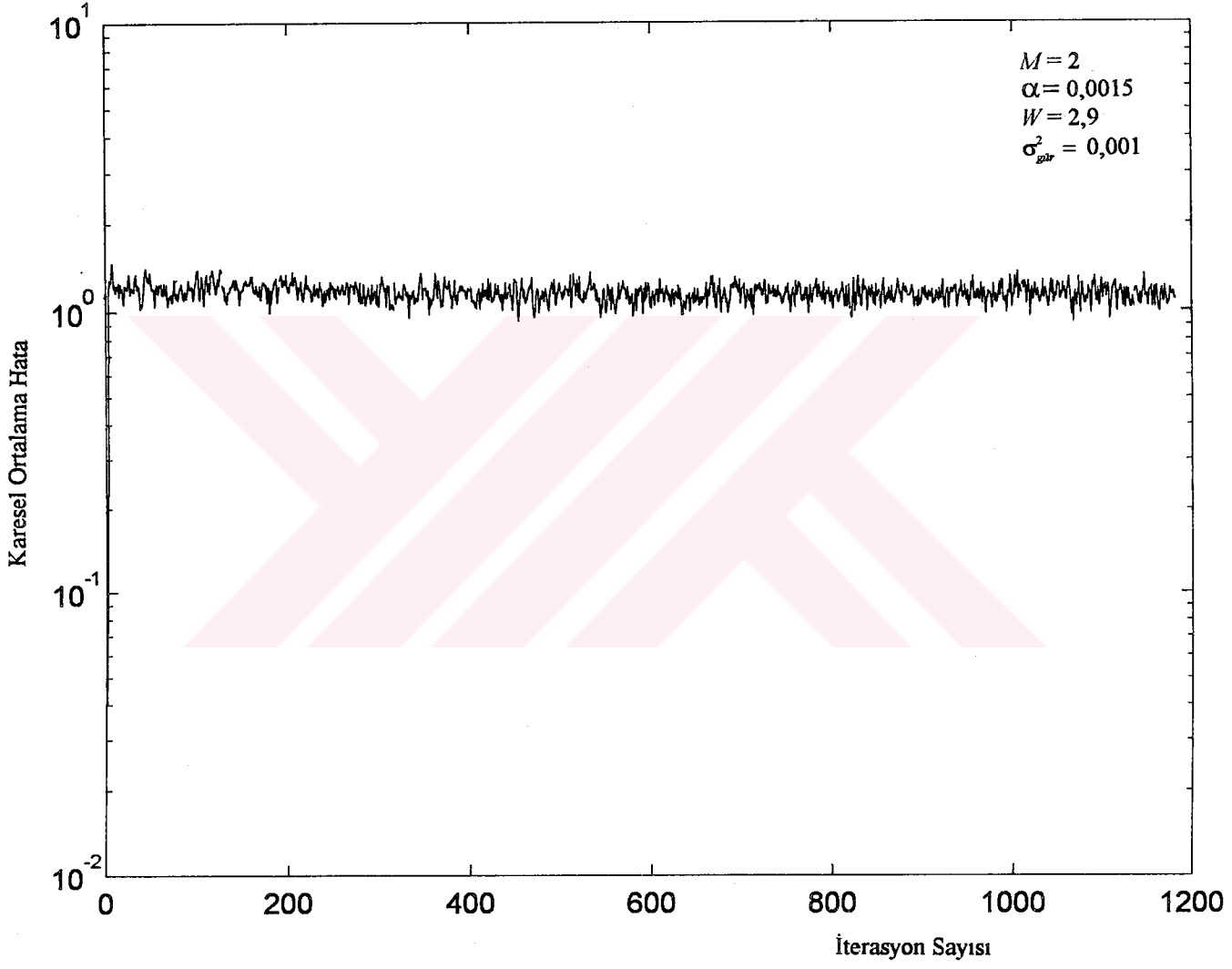
$M = 3$ için, Şekil 9.6 (b)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 810 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,830$ ' dur.

$M = 4$ için, Şekil 9.6 (c)' den görüldüğü gibi öğrenim eğrisi $M = 3$ için çizdirilen Şekil 9.6 (b) ile aynıdır.

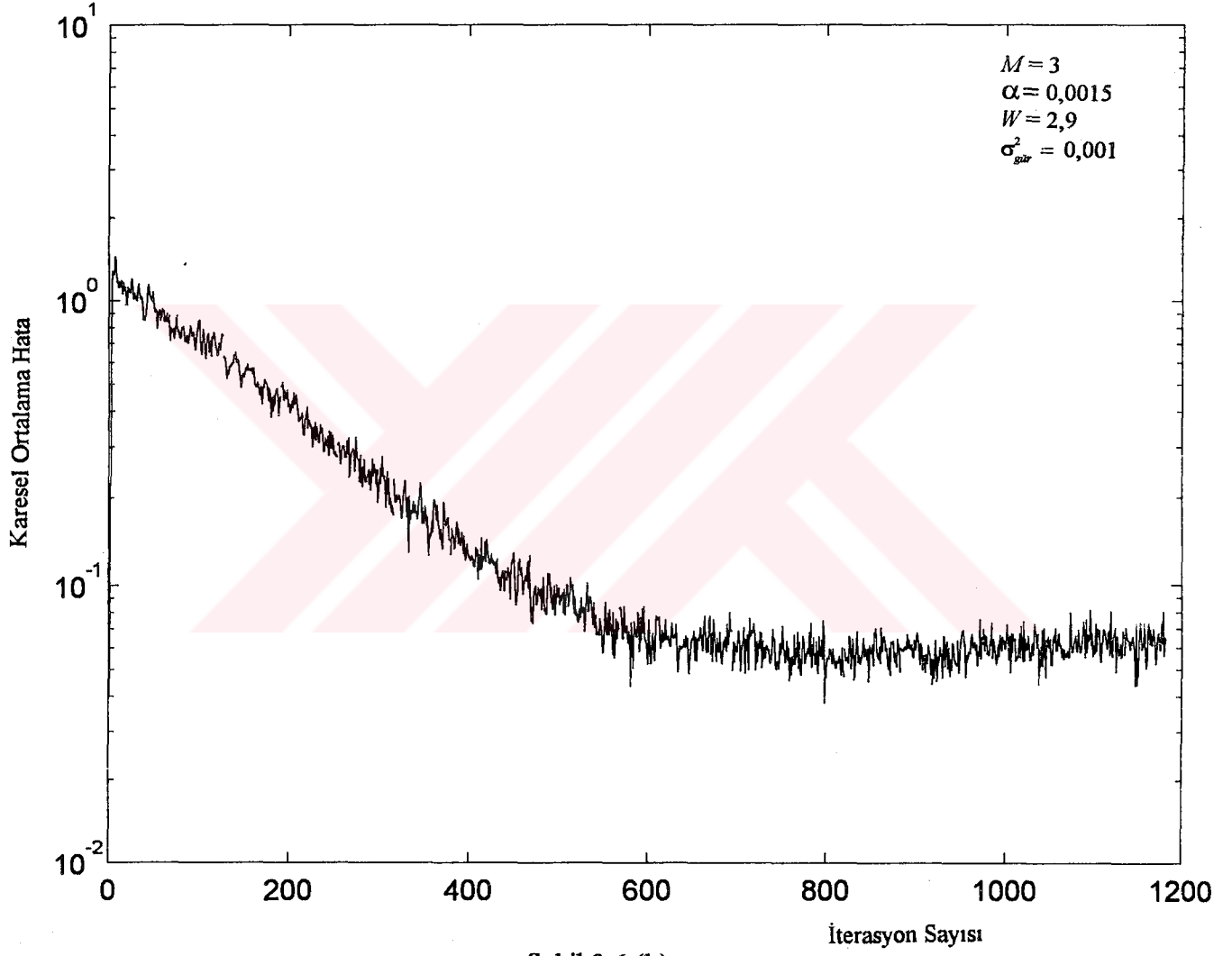
$M = 5$ için, Şekil 9.6 (d)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 800 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,0301$ ' dir.

$M = 6$ için, Şekil 9.6 (e)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 780 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,0210$ ' dur.

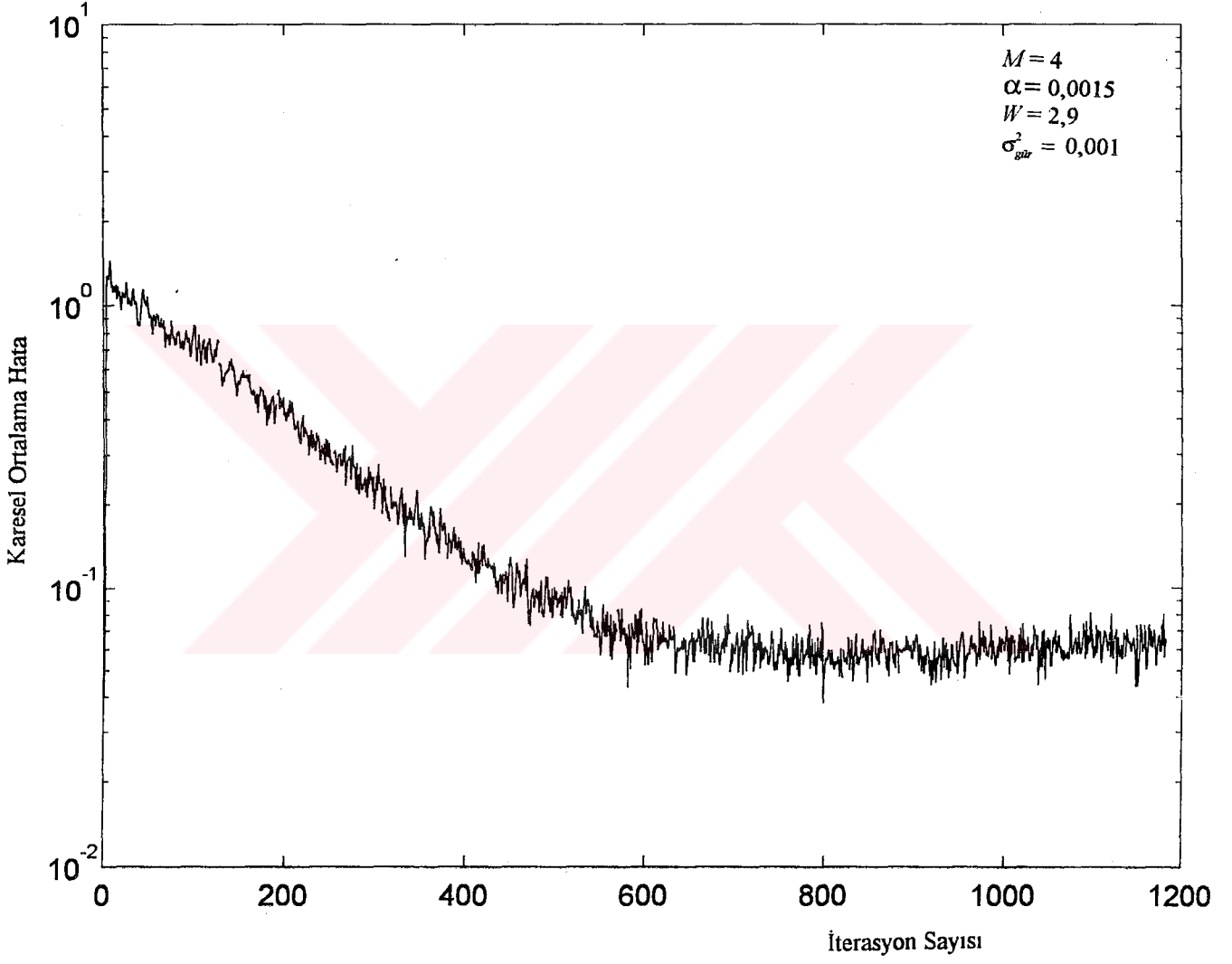
Son olarak, $M = 7$ için Şekil 9.6 (f) ve $M = 8$ için Şekil 9.6 (g) incelendiğinde, bu eğrilerin yakınsama hızlarının ve kararlı durumdaki minimum karesel ortalama hata değerlerinin $M = 6$ değeri için çizdirilen Şekil 9.6 (e) ile aynı olduğu görülür.



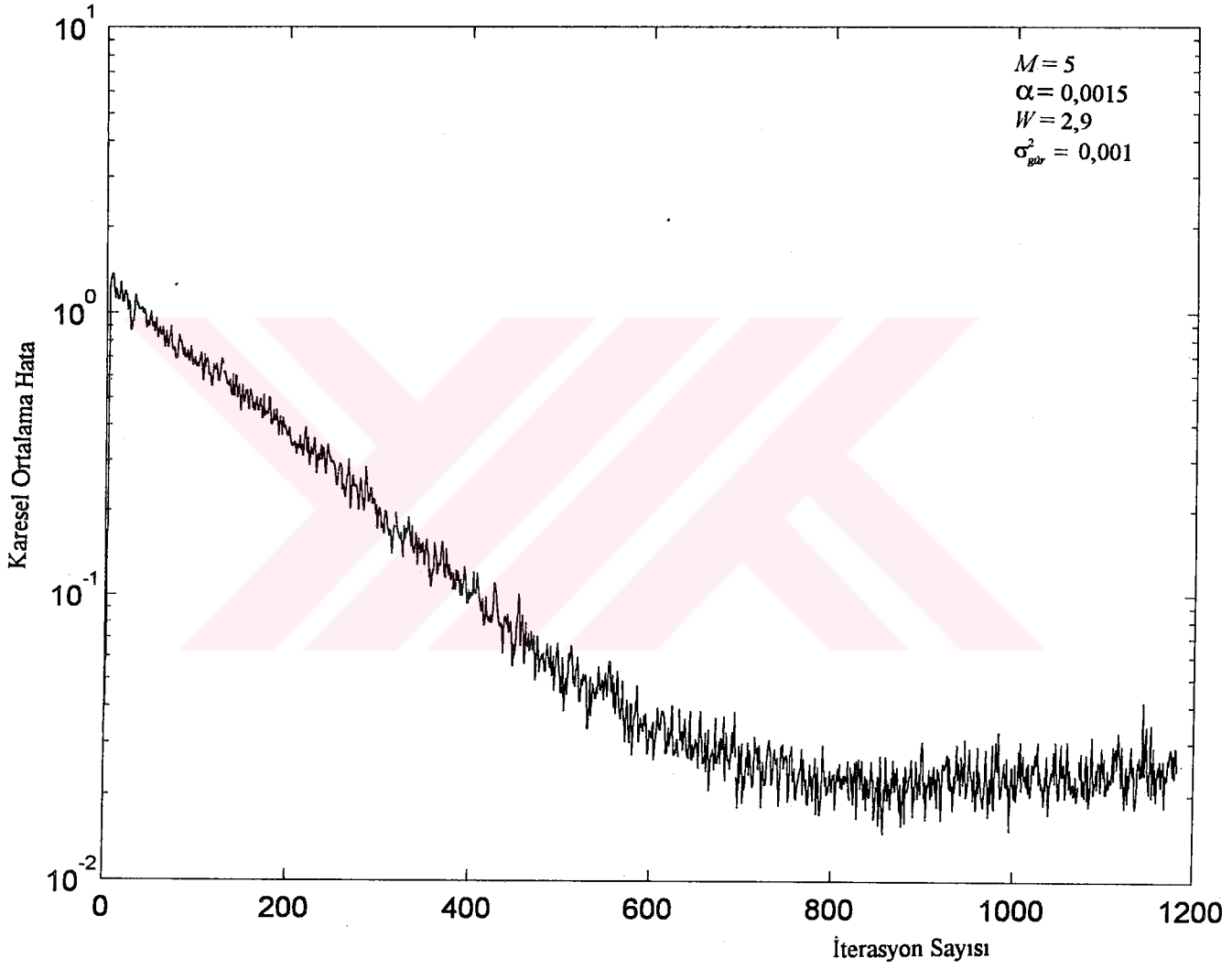
Şekil 9.6 (a)



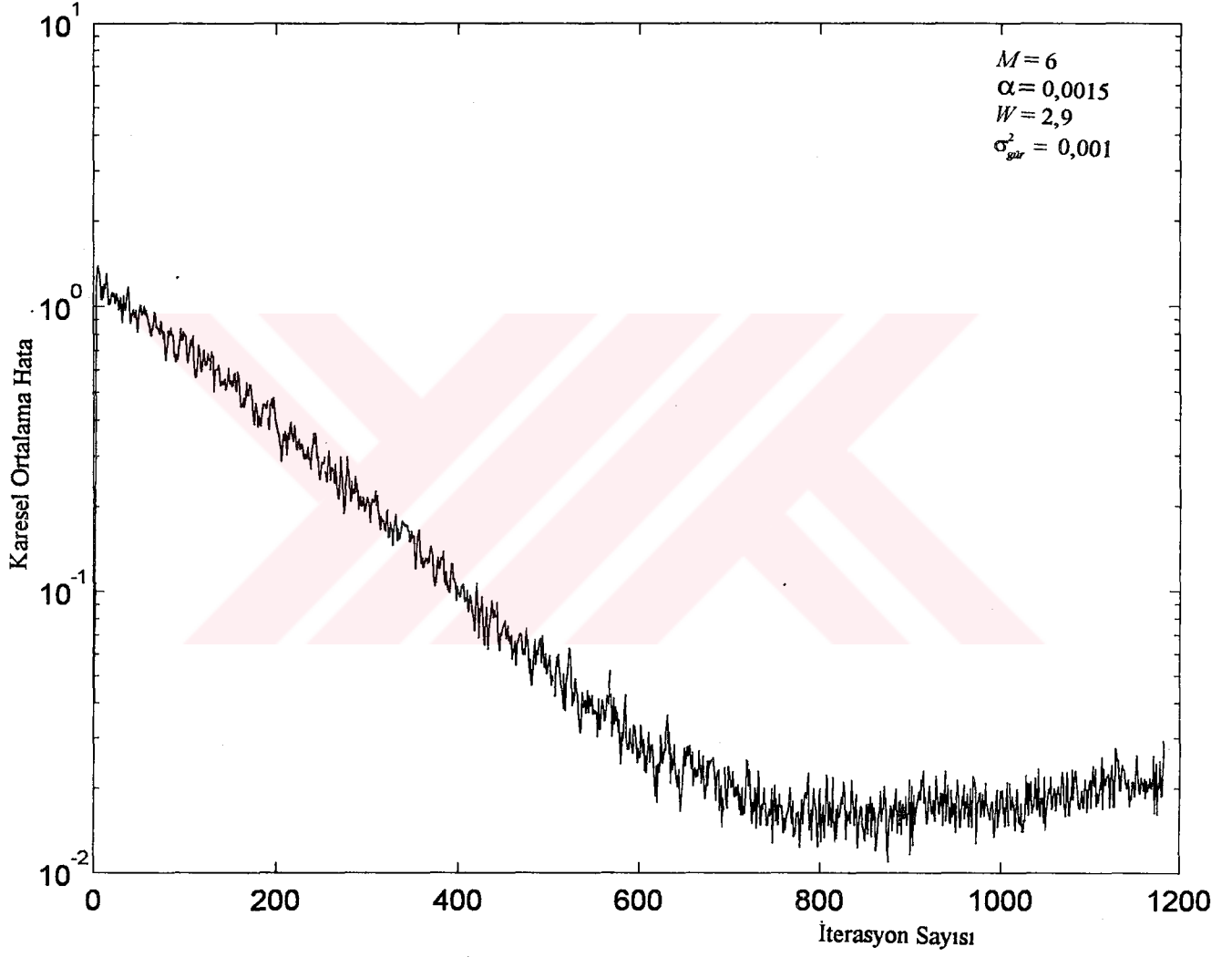
Şekil 9.6 (b)



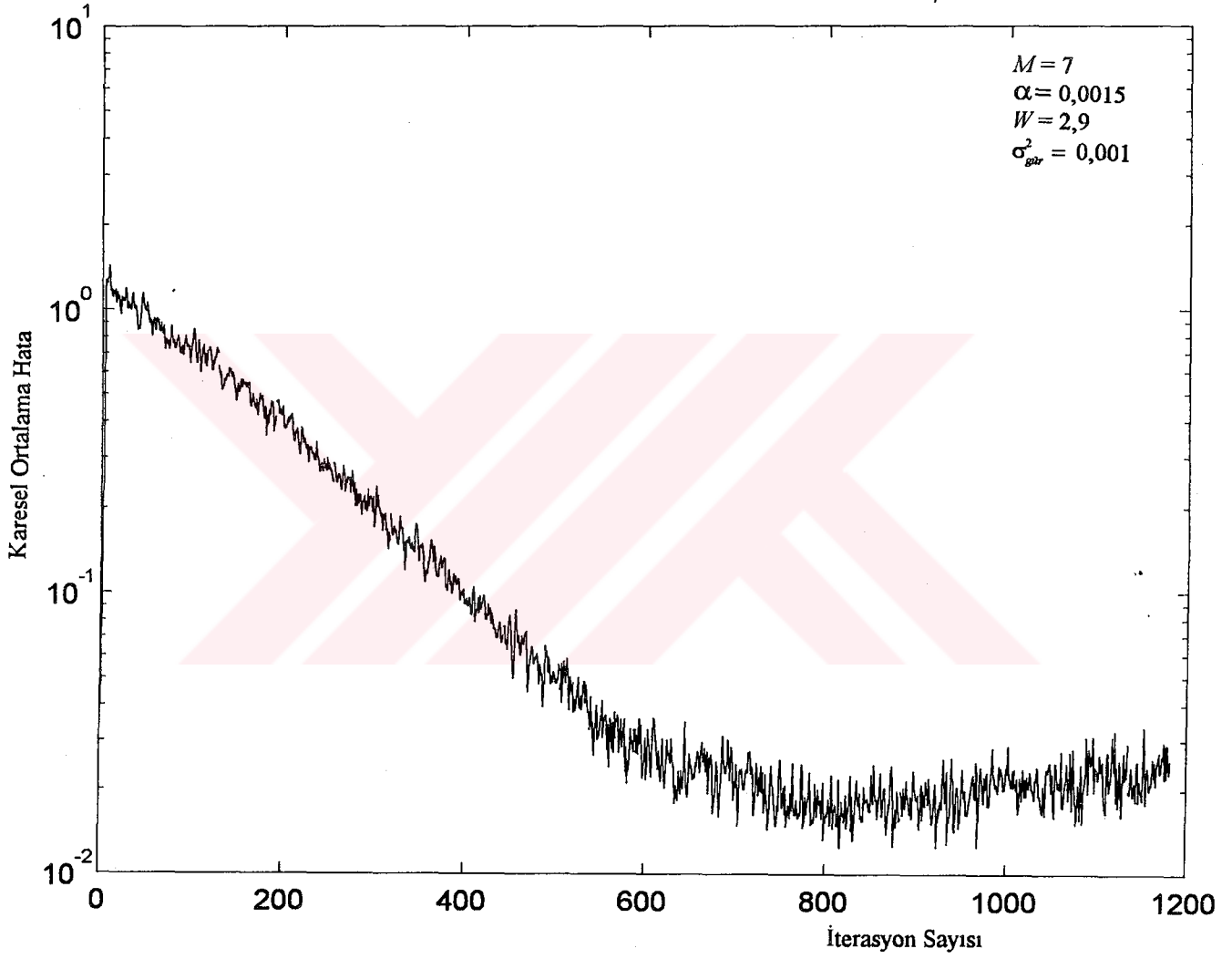
Şekil 9.6 (c)



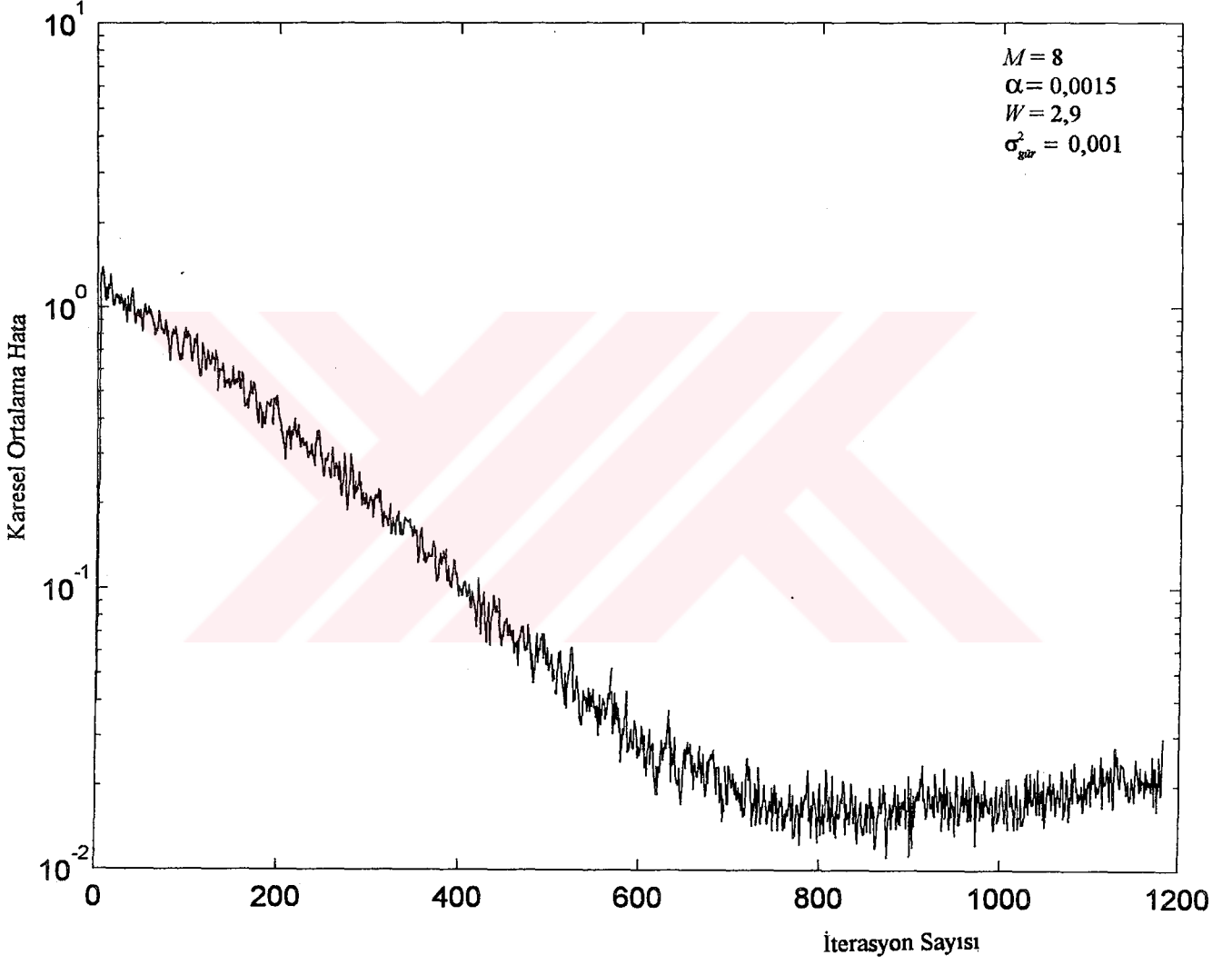
Şekil 9.6 (d)



Şekil 9.6 (e)



Şekil 9.6 (f)



Şekil 9.6 (g)

İkinci bir kanal modelini aşağıdaki gibi seçerek,

$$h(n) = \begin{cases} 0,4 & n = 1 \\ 0,3 & n = 2 \\ 0,2 & n = 3 \\ 1 & n = 4 \\ 0,2 & n = 5 \\ 0,3 & n = 6 \\ 0,4 & n = 7 \end{cases}$$

incelememize devam edelim.

$M = 3$ için algoritma çalıştırıldığında, Şekil 9.7 (a)' dan görüldüğü gibi karesel ortalama hatada hiç bir azalma olmadığı yani modelimizin kanalı hiç bir şekilde özdeşleyemediği görülür.

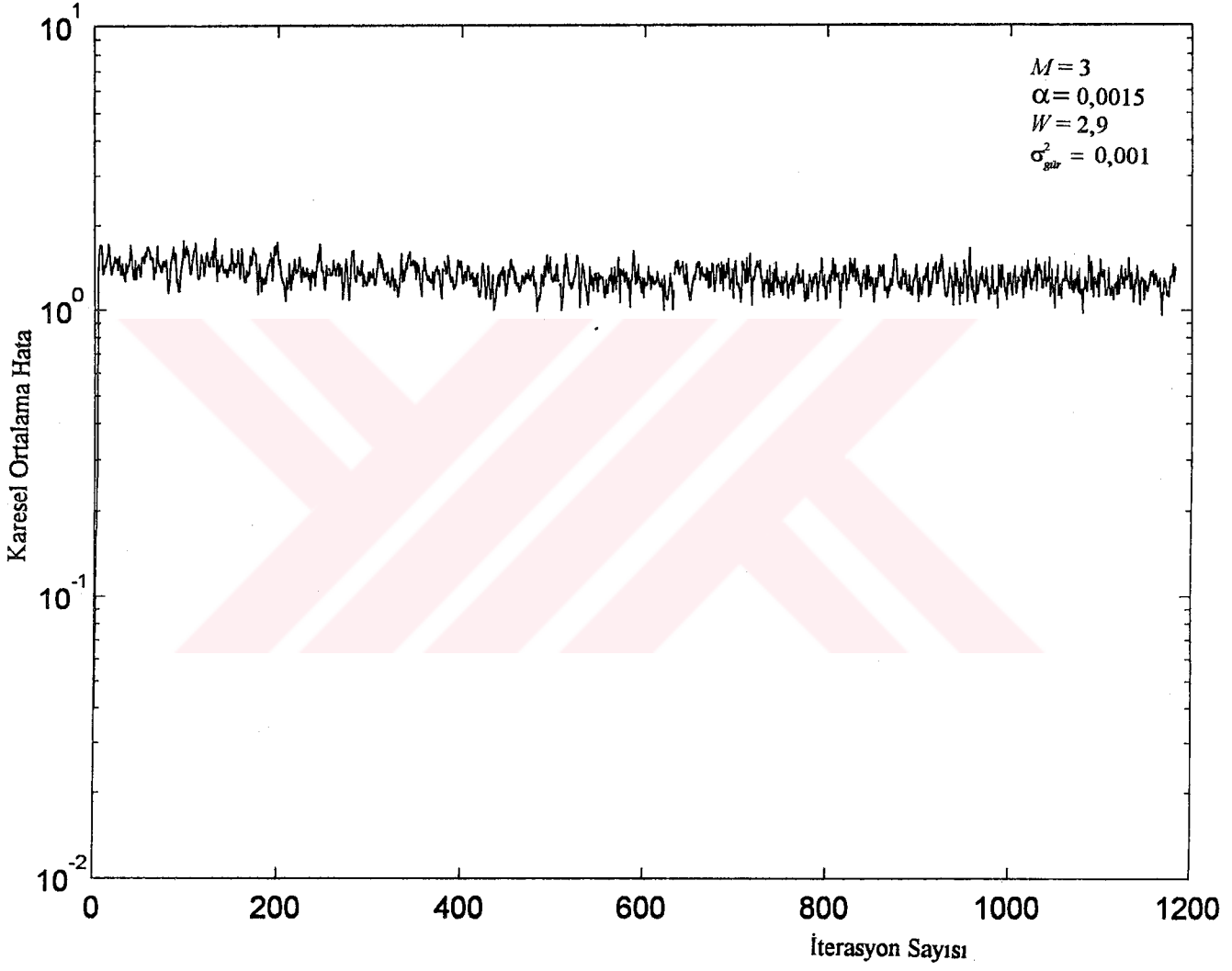
$M = 4$ için, Şekil 9.7 (b)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 820 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,447$ ' dir.

$M = 5$ için, Şekil 9.7 (c)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 810 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,255$ ' dir.

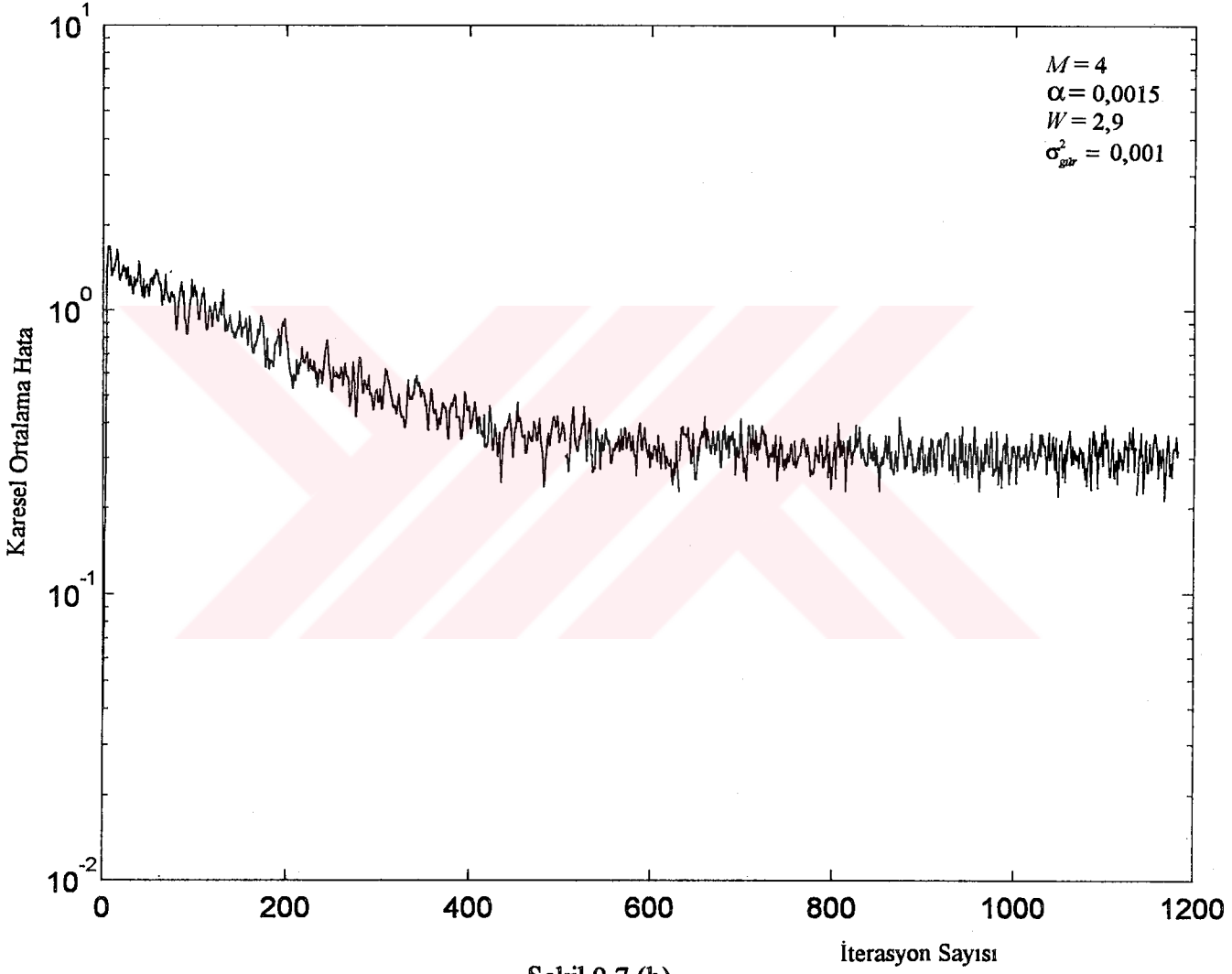
$M = 6$ için, Şekil 9.7 (d)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 800 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,0820$ ' dir.

$M = 7$ için, Şekil 9.7 (e)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 780 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,0477$ ' dir.

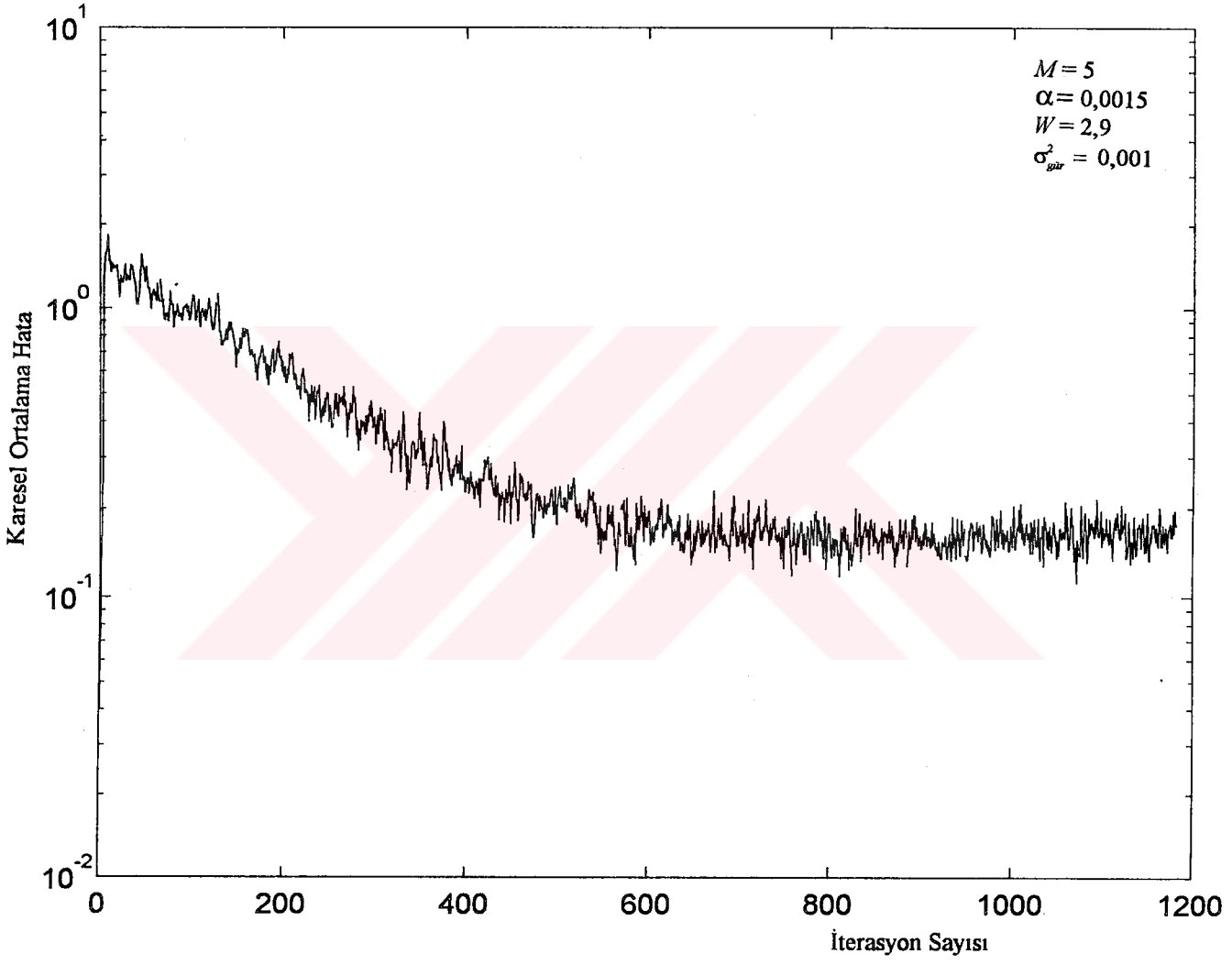
Son olarak, $M = 8$ için Şekil 9.7 (f) ve $M = 9$ için Şekil 9.7 (g) incelendiğinde, bu eğrilerin yakınsama hızlarının ve kararlı durumdaki minimum karesel ortalama hata değerlerinin $M = 7$ değeri için çizdirilen Şekil 9.7 (e) ile aynı olduğu görülür.



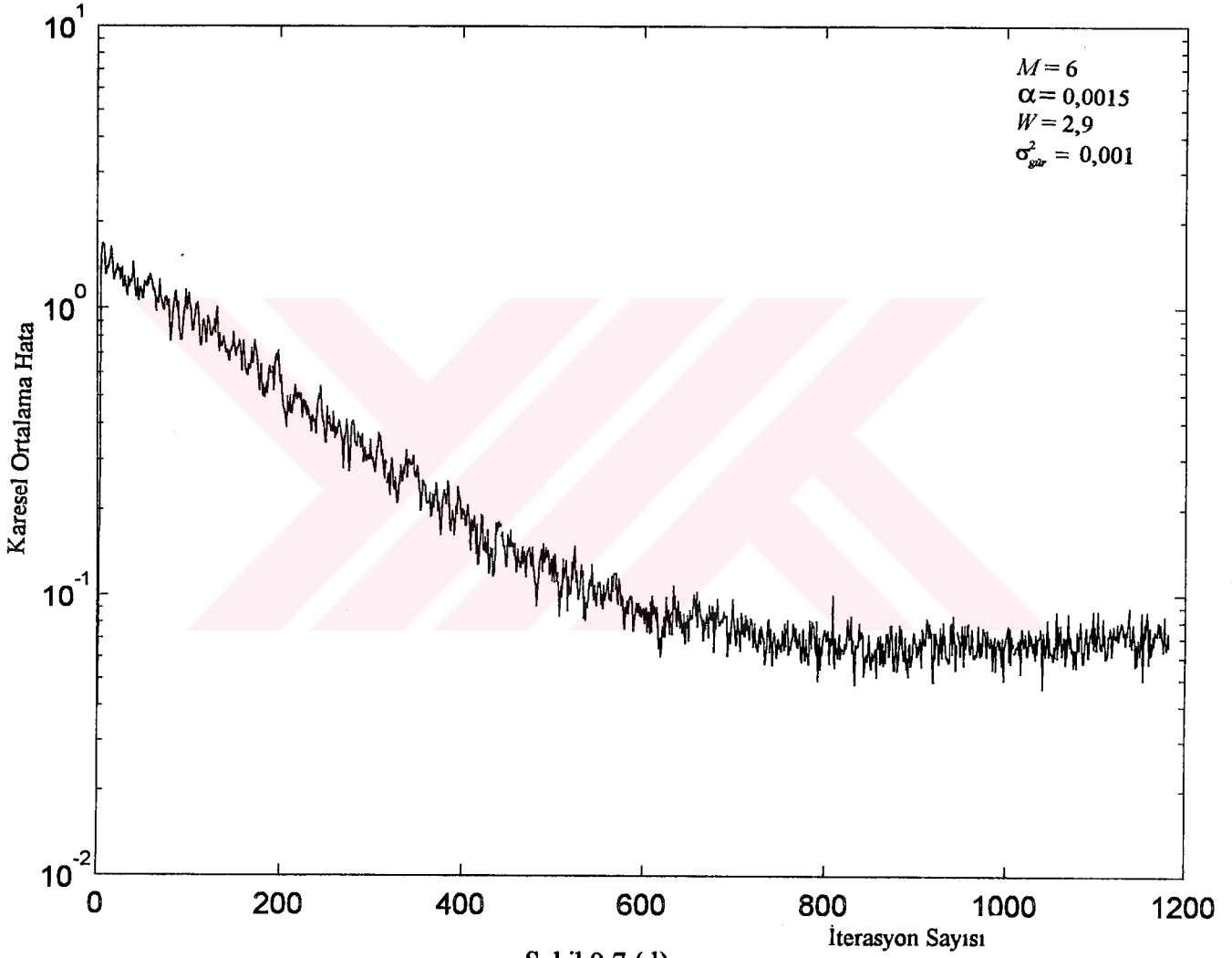
Şekil 9.7 (a)

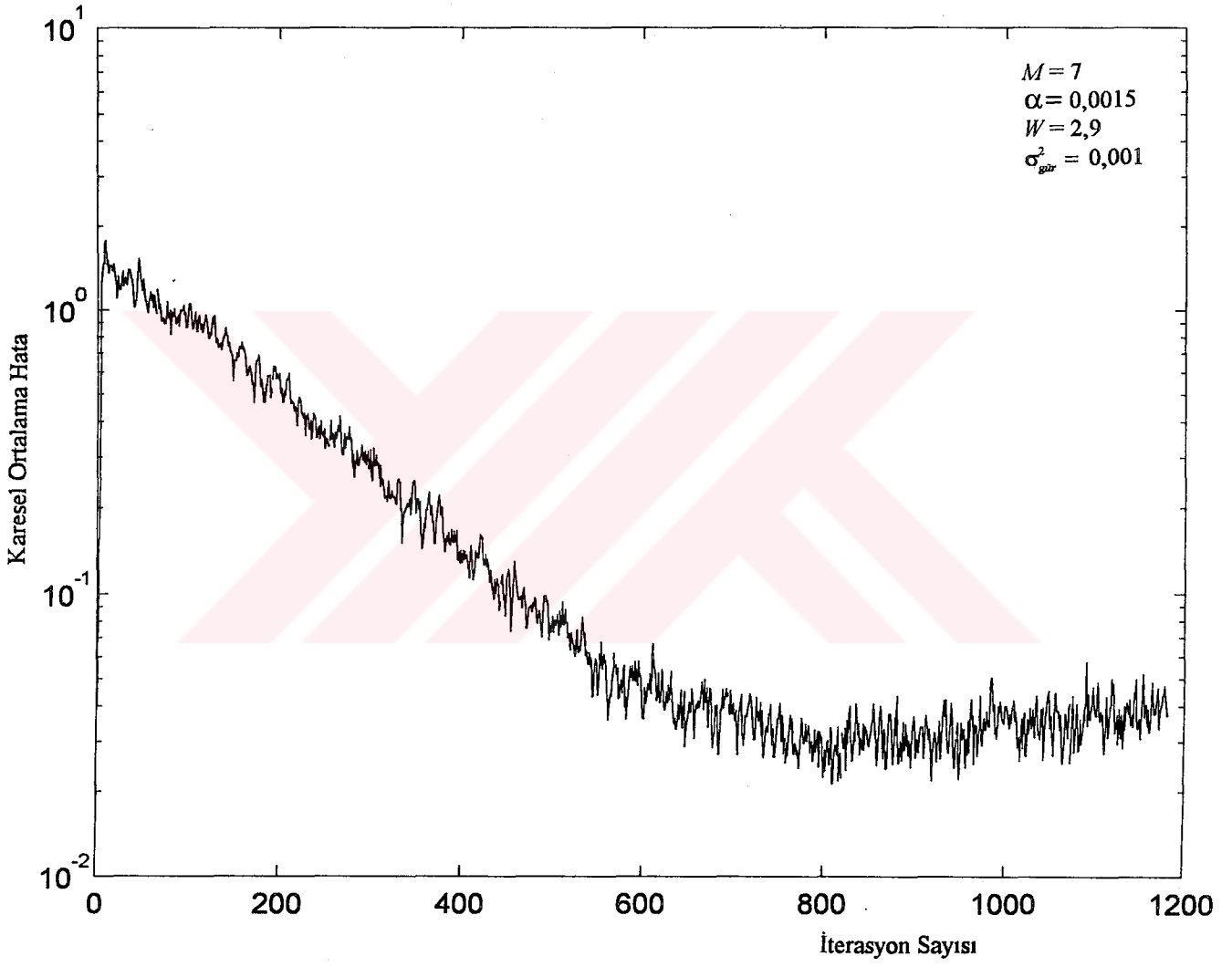


Şekil 9.7 (b)

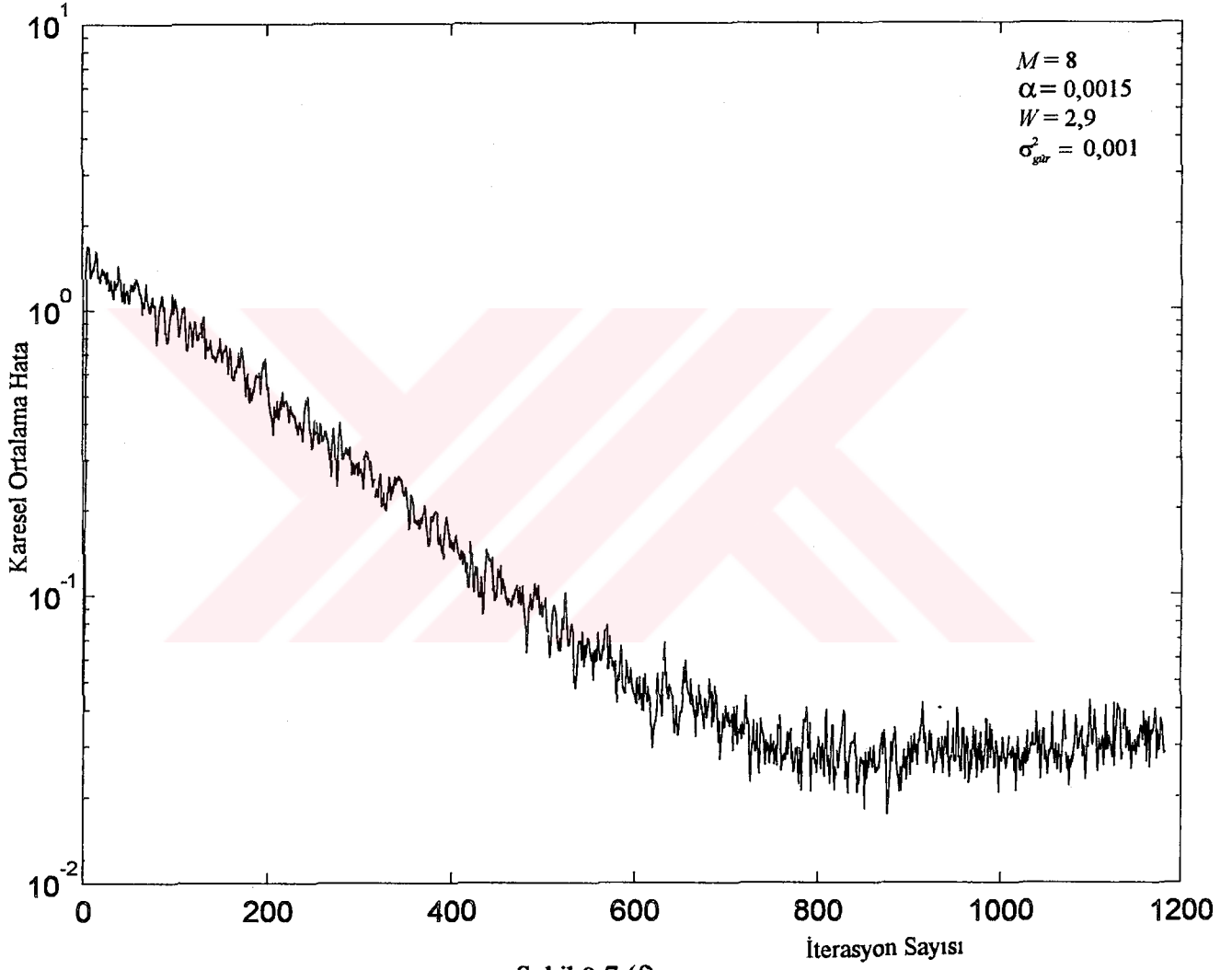


Şekil 9.7 (c)

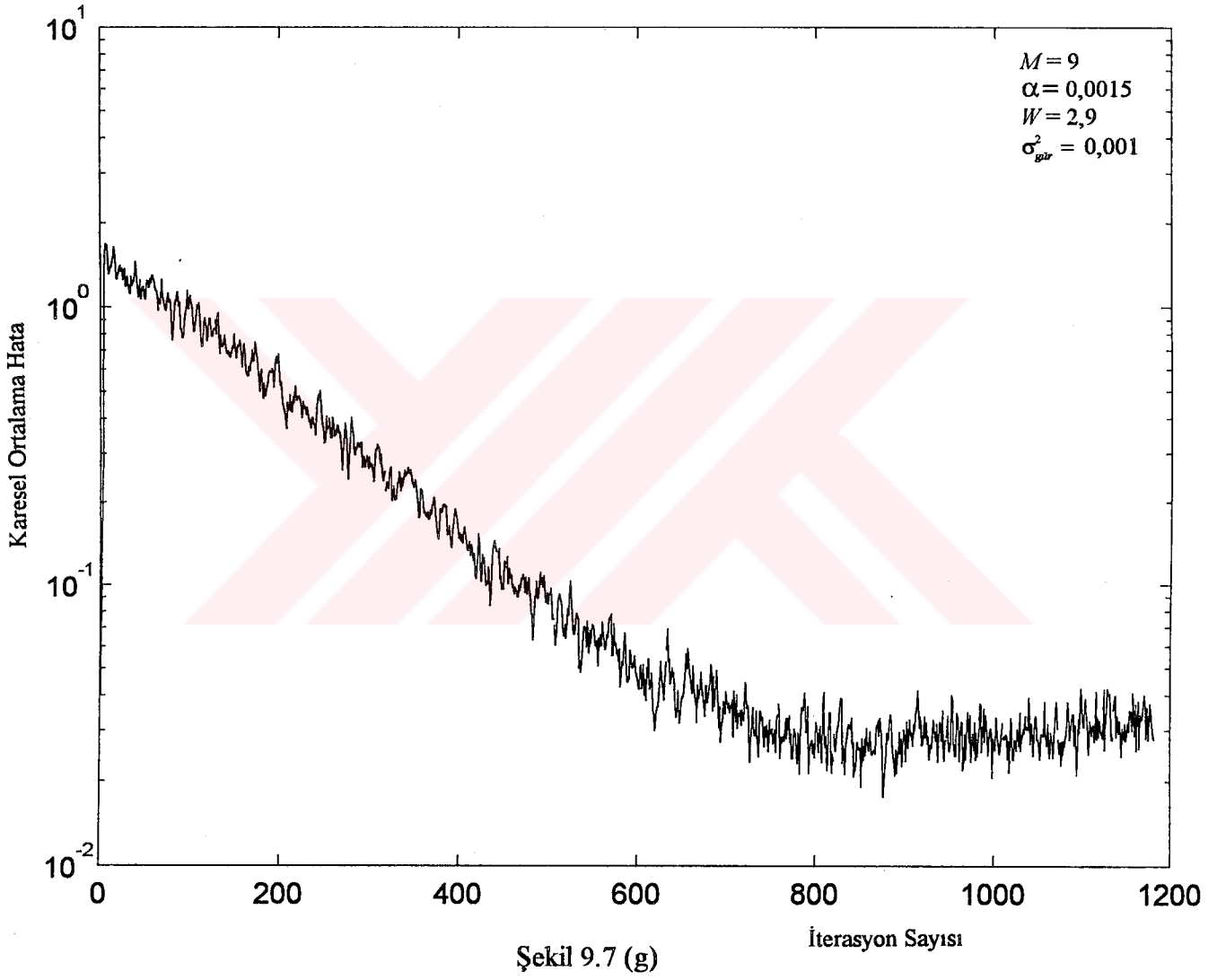




Şekil 9.7 (e)



Şekil 9.7 (f)



Üçüncü bir kanal modelini aşağıdaki gibi seçerek,

$$h(n) = \begin{cases} 0,2 & n = 1 \\ 0,4 & n = 2 \\ 0,6 & n = 3 \\ 0,8 & n = 4 \\ 1 & n = 5 \end{cases}$$

incelememize devam edelim.

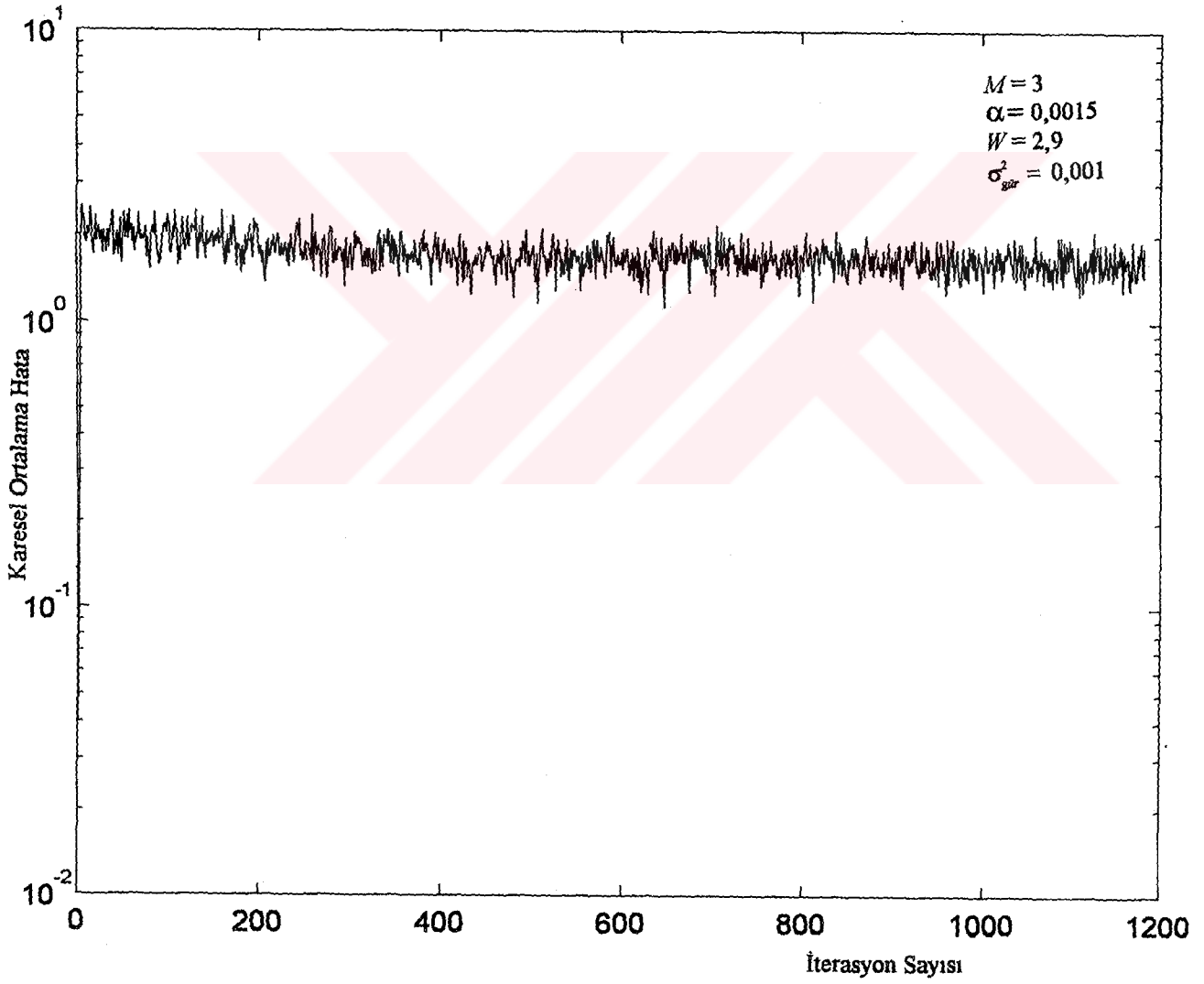
$M = 3$ için algoritma çalıştırıldığında, Şekil 9.8 (a)' dan görüldüğü gibi karesel ortalama hatada hiç bir azalma olmadığı yani modelimizin kanalı hiç bir şekilde özdeşleyemediği görülür.

$M = 4$ için de, Şekil 9.8 (b)' den görüldüğü gibi modelimizin kanalı özdeşleyemediği görülür.

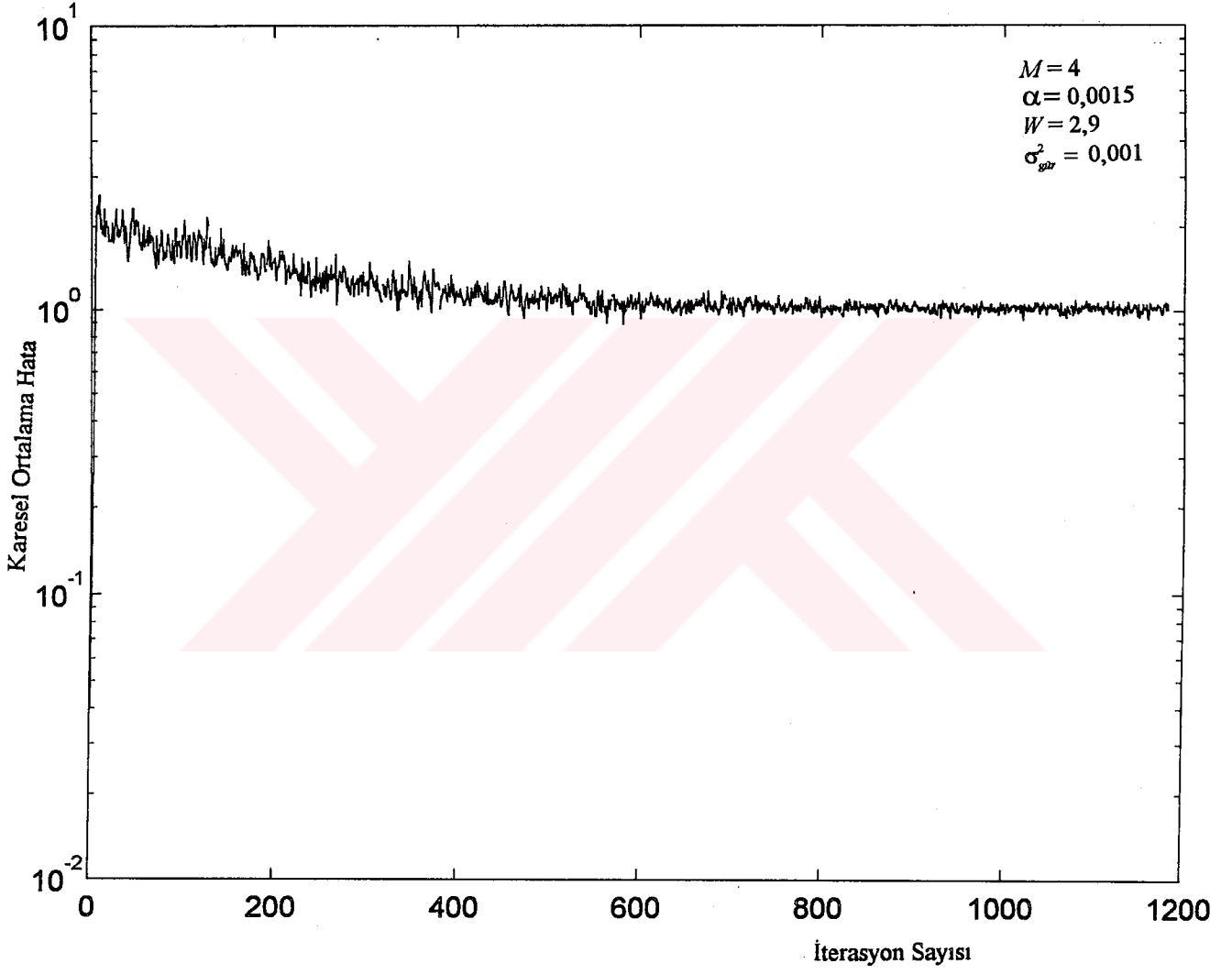
$M = 5$ için, Şekil 9.8 (c)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 920 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,0778$ ' dir.

Son olarak, $M = 6$ için Şekil 9.8 (d) ve $M = 7$ için Şekil 9.8 (e) incelendiğinde, bu eğrilerin yakınsama hızlarının ve kararlı durumdaki minimum karesel ortalama hata değerlerinin $M = 5$ değeri için çizdirilen Şekil 9.8 (c) ile aynı olduğu görülür.

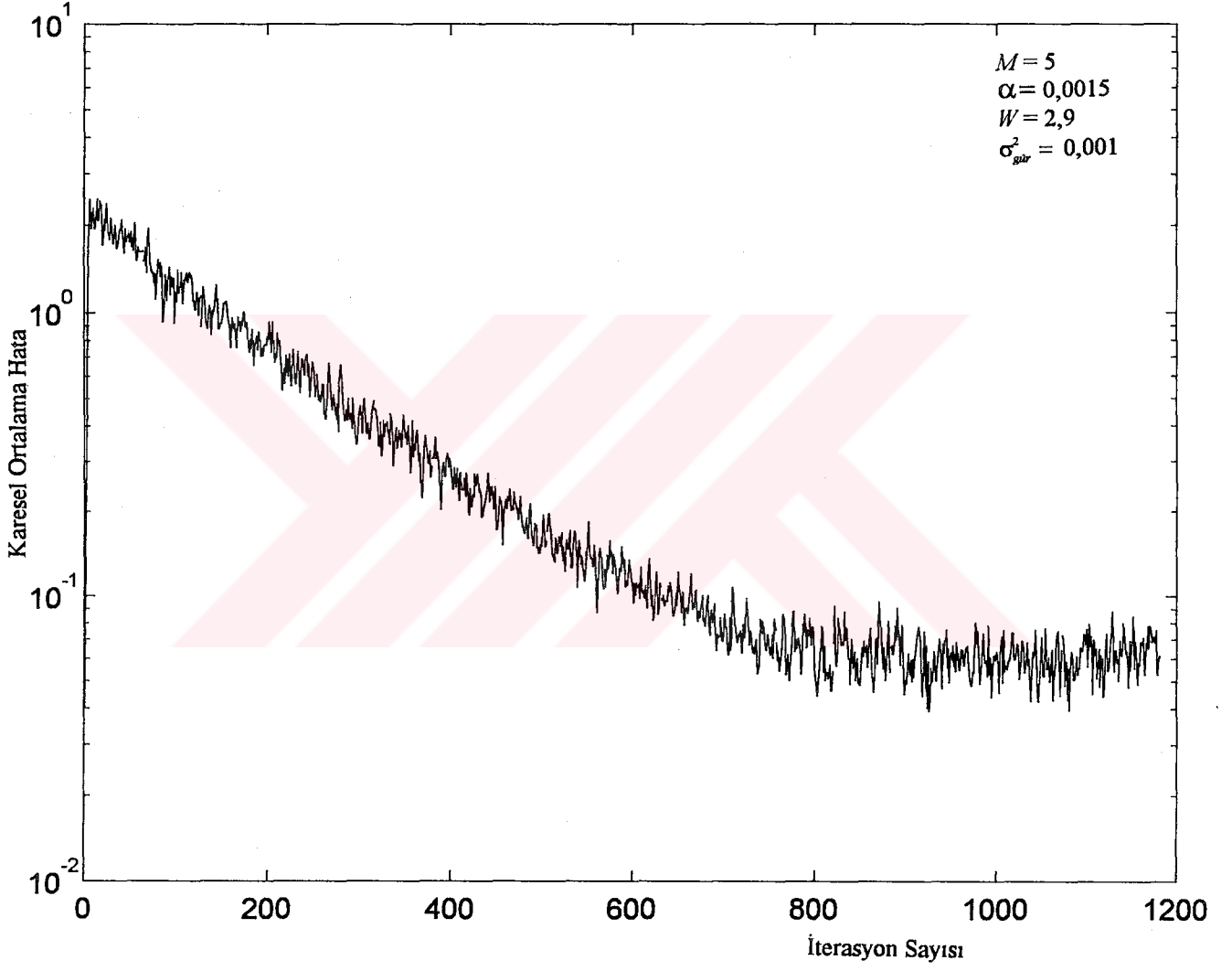
Sonuç olarak; adım uzunluğu, W kanal parametresi ve gürültünün varyansı sabit tutularak kafes derecesi M' e göre karesel ortalama hatayı incelediğimizde, kafes derecesi M' in artışı ile yakınsama hızının arttığı ve kararlı durumda ulaşılan karesel ortalama hatanın azaldığı görülmektedir. Ayrıca, seçilen kanalın yapısına göre, belli bir kafes derecesi değerinden sonra özdeşlemenin olabilecek en iyi şekilde sağlandığı ve bundan sonra kafes derecesinin arttırılmasının hiç bir olumlu değişim sağlamadığı görülmektedir. Bu aşamadan sonra meydana gelen tek değişim karesel ortalama hatanın değişim sınırlarının (dalgalanmaların) artmasıdır.



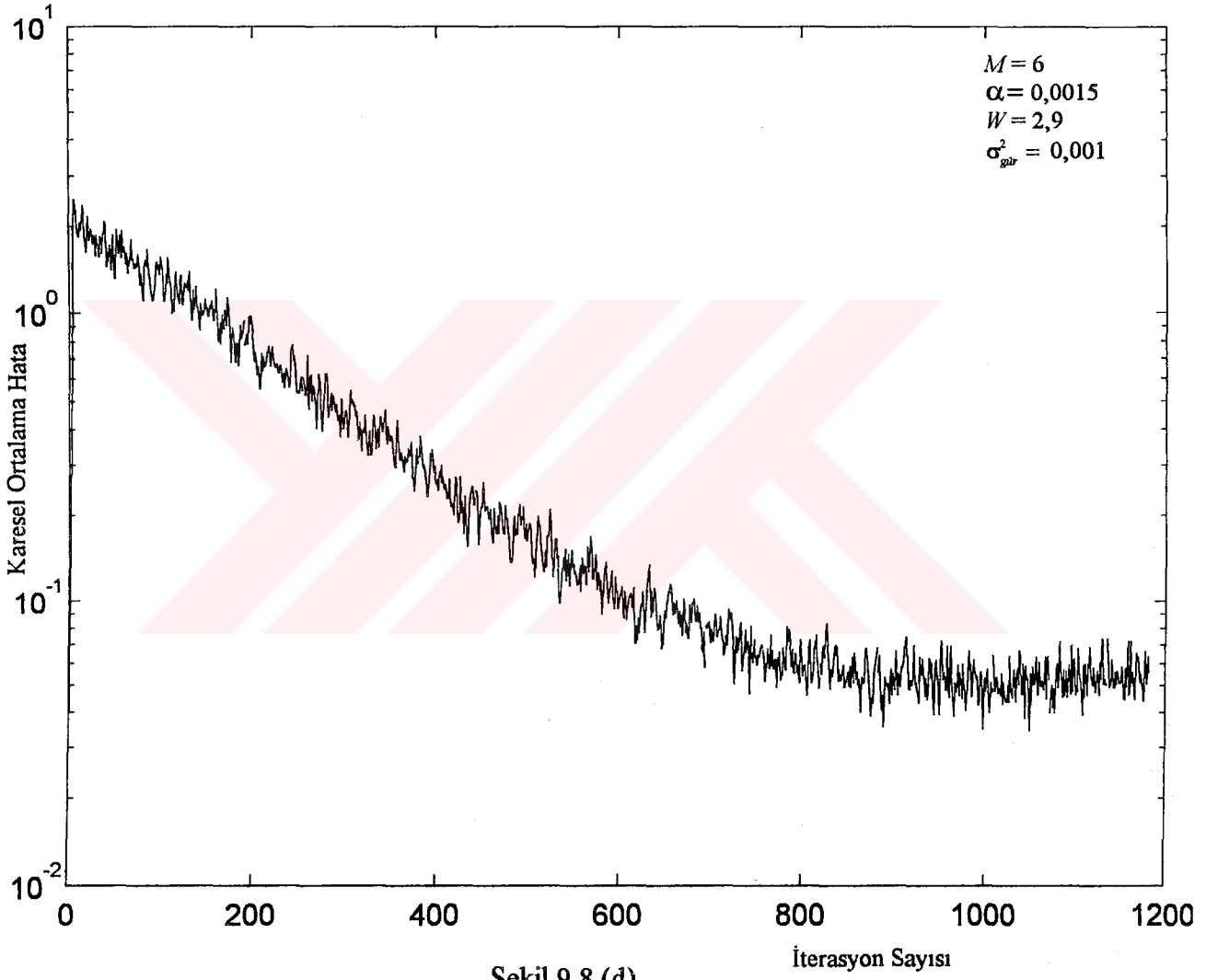
Şekil 9.8 (a)

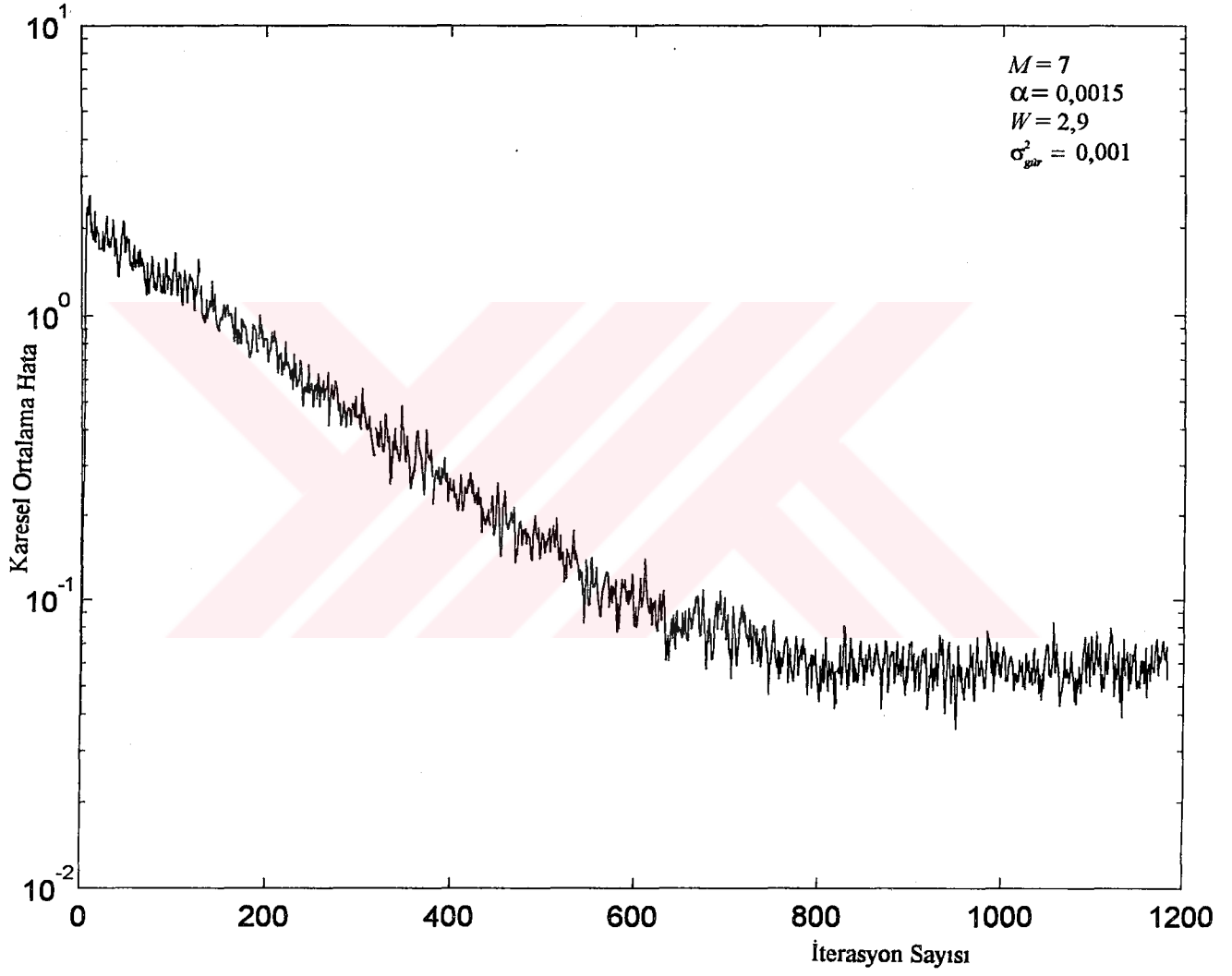


Şekil 9.8 (b)



Şekil 9.8 (c)





Şekil 9.8 (e)

9.3.2. Adım Uzunluğunun Karesel Ortalama Hataya Etkisinin İncelenmesi

Adım uzunluğu α 'nın değişimlerine karşı algoritma sonuçlarının değişimlerini incelemek üzere diğer parametreler sabit tutulmuştur:

Kafes derecesi $M = 3$

Kanal parametresi $W = 2,9$

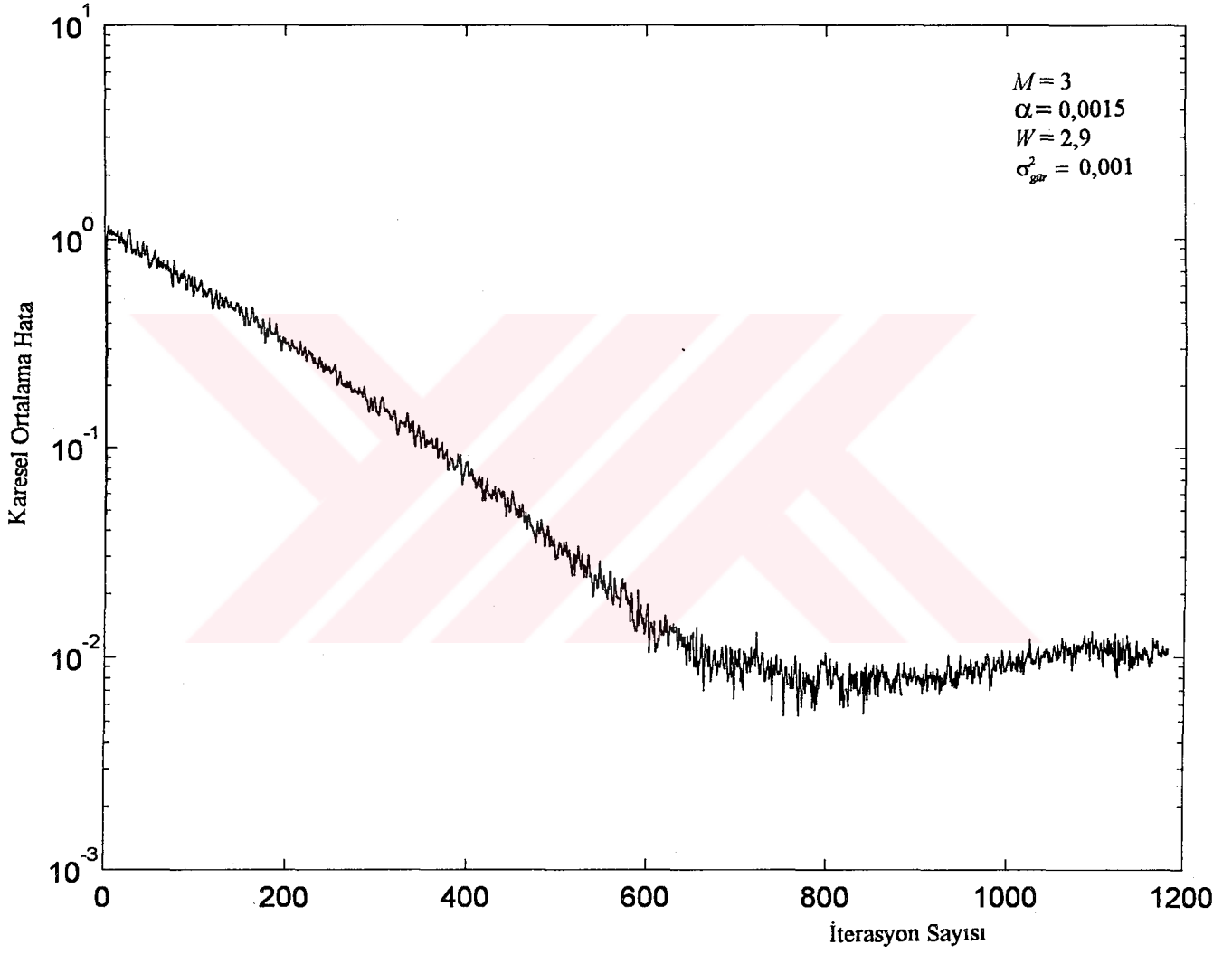
Gürültünün değışintisi $\sigma_{gür}^2 = 0,001$

$\alpha = 0,0015$ için, Şekil 9.9 (a)'den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 820 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{min} = 0,0095$ ' dir.

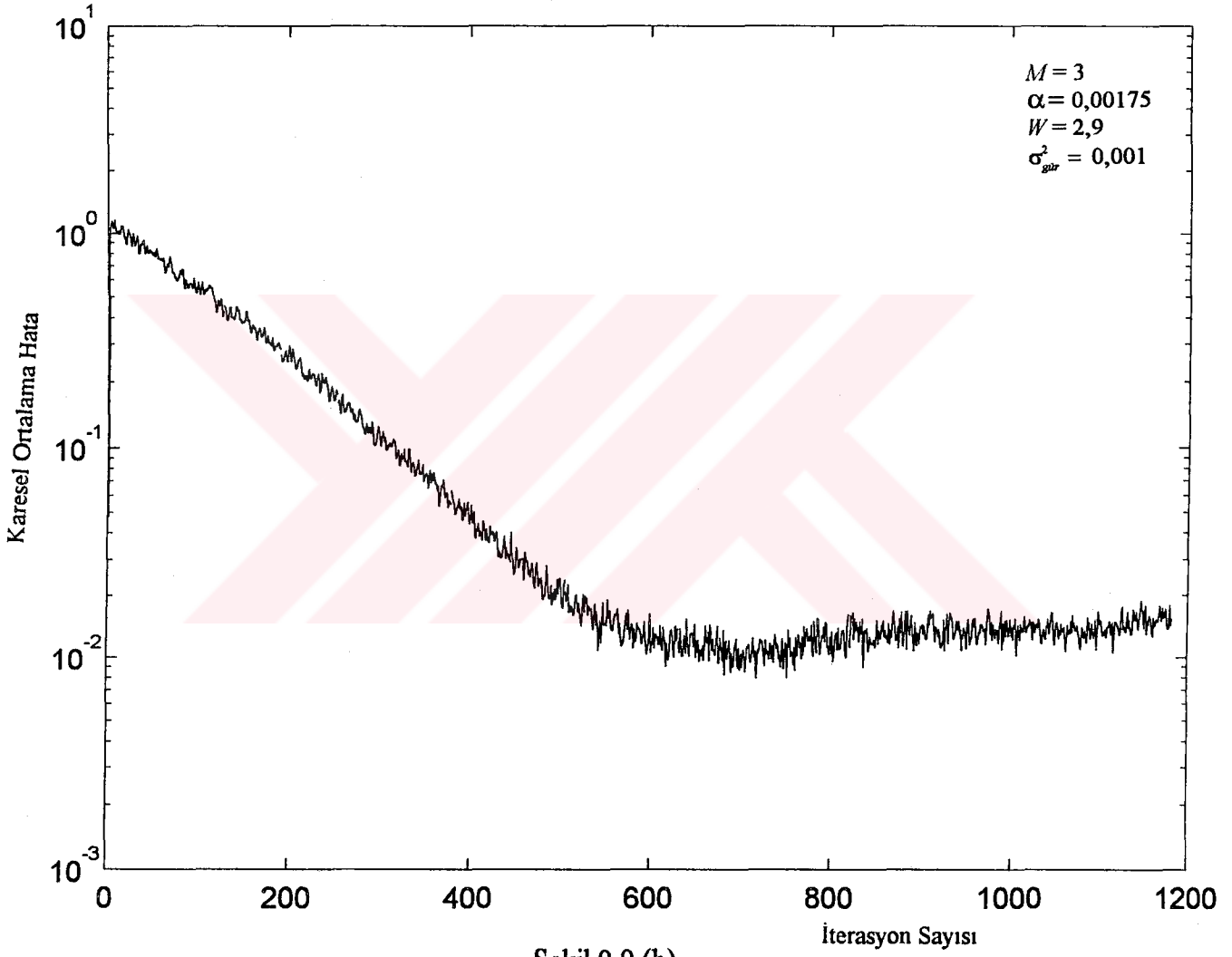
$\alpha = 0,00175$ için, Şekil 9.9 (b)'den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 790 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{min} = 0,0150$ ' dir.

$\alpha = 0,002$ için, Şekil 9.9 (c)'den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 650 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{min} = 0,0230$ ' dir.

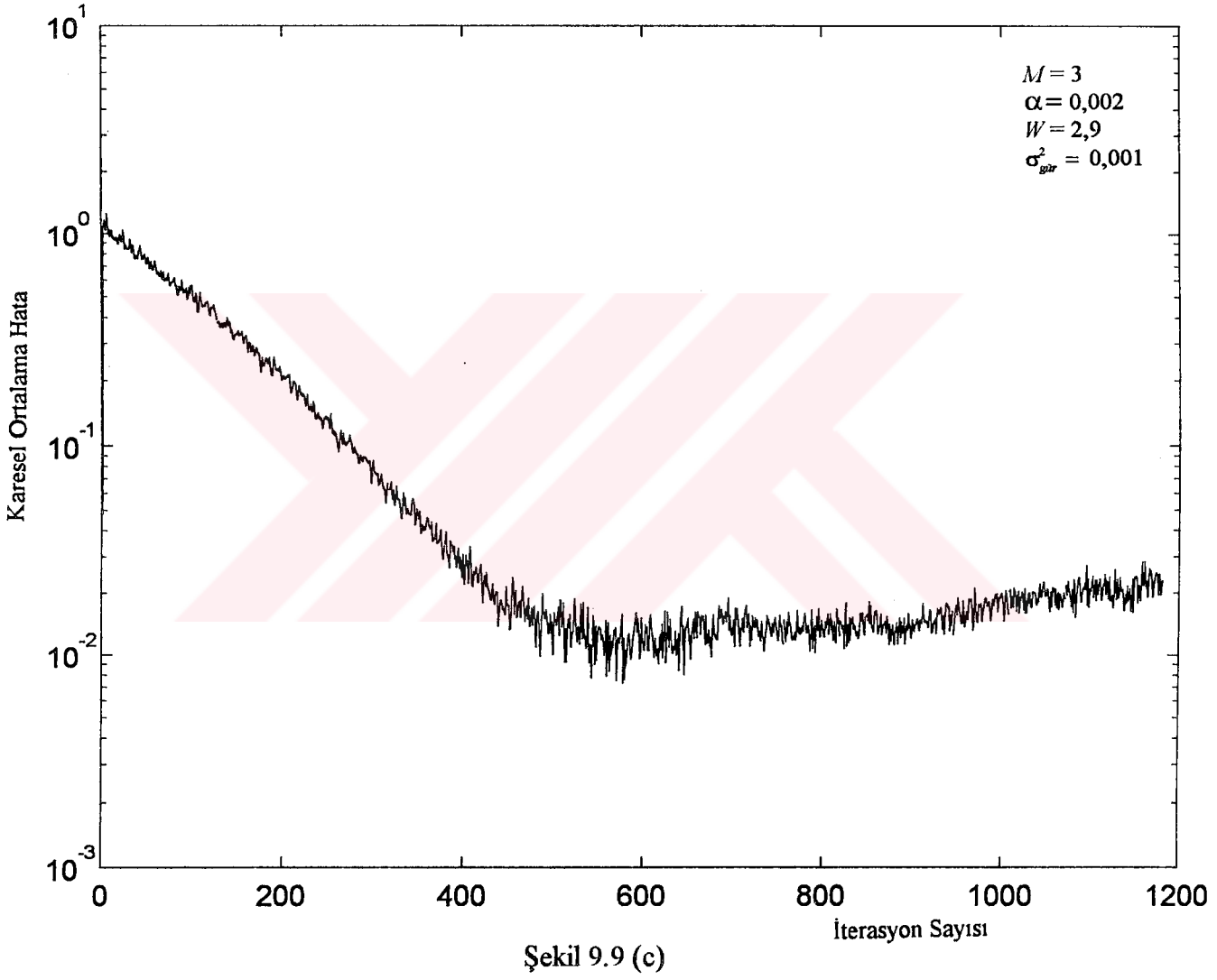
Sonuç olarak; kafes derecesi, W kanal parametresi ve gürültünün değışintisi sabit tutularak adım uzunluğu α 'ya göre karesel ortalama hatayı incelediğimizde, adım uzunluğu α 'nın artışı ile yakınsama hızının arttığı, fakat kararlı durumda ulaşılan karesel ortalama hatanın büyüdüğü görülmektedir.



Şekil 9.9 (a)



Şekil 9.9 (b)



9.3.3. Kanal Parametresi W ' nin Karesel Ortalama Hataya Etkisinin İncelenmesi

Kanalın birim örnek yanıtındaki W parametresinin tap girişlerinin ilişki matrisinin özdeğerlerini kontrol edebilen bir parametre olduğunu daha önce söylemiştik. W parametresinin artışı ile tap girişinin ilişki matrisinin özdeğer dağılımı ($X(\mathbf{R}) = \lambda_{\max}/\lambda_{\min}$) artar.

Kanal parametresi W ' nin değişimlerine göre algoritma sonuçlarının değişimlerini incelemek üzere diğer parametreler sabit tutulmuştur:

Kafes derecesi $M = 3$

Adım uzunluğu $\alpha = 0,0015$

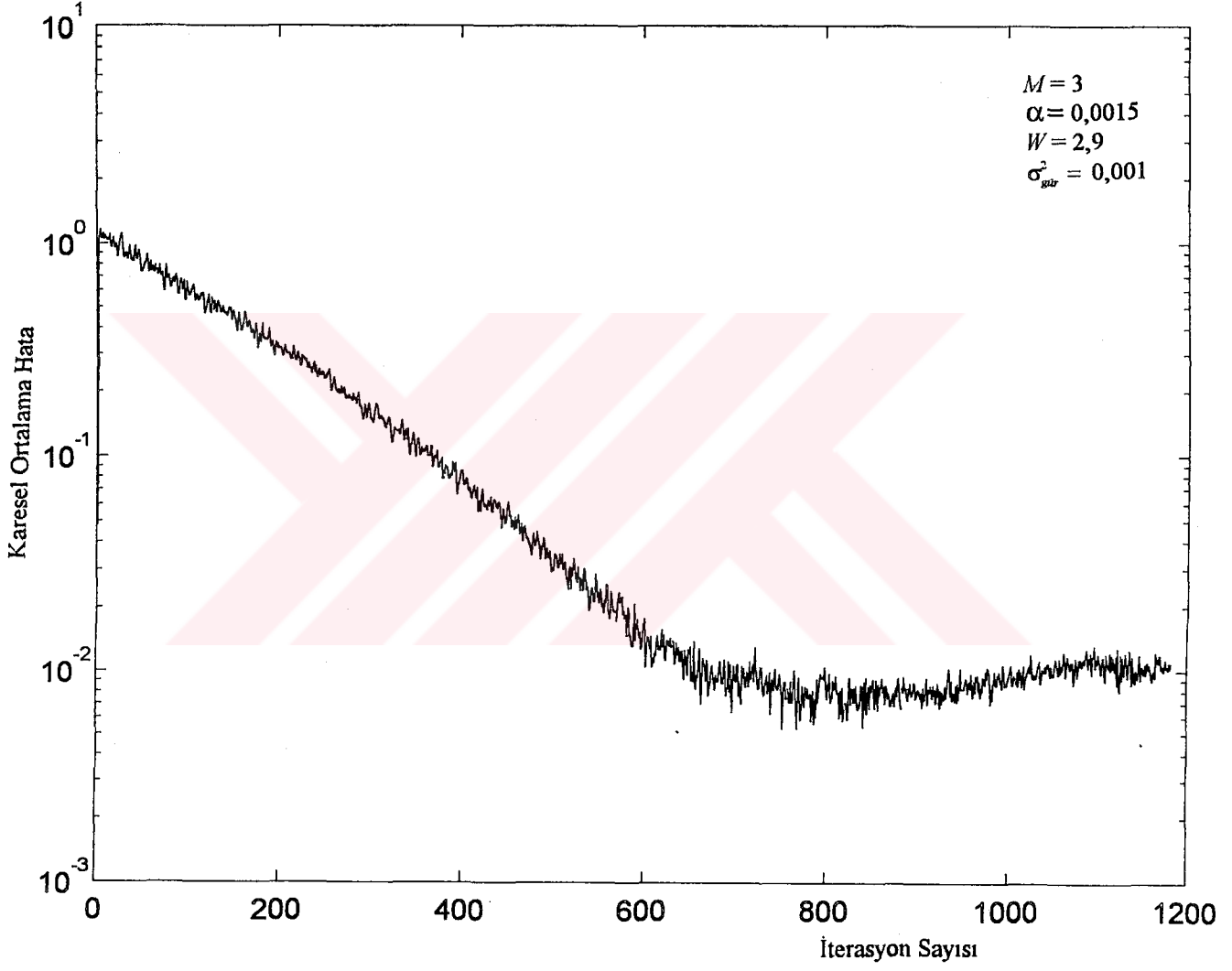
Gürültünün değışintisi $\sigma_{gur}^2 = 0,001$

$W = 2,9$ için, Şekil 9.10 (a)' dan görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 820 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,0095$ ' dir.

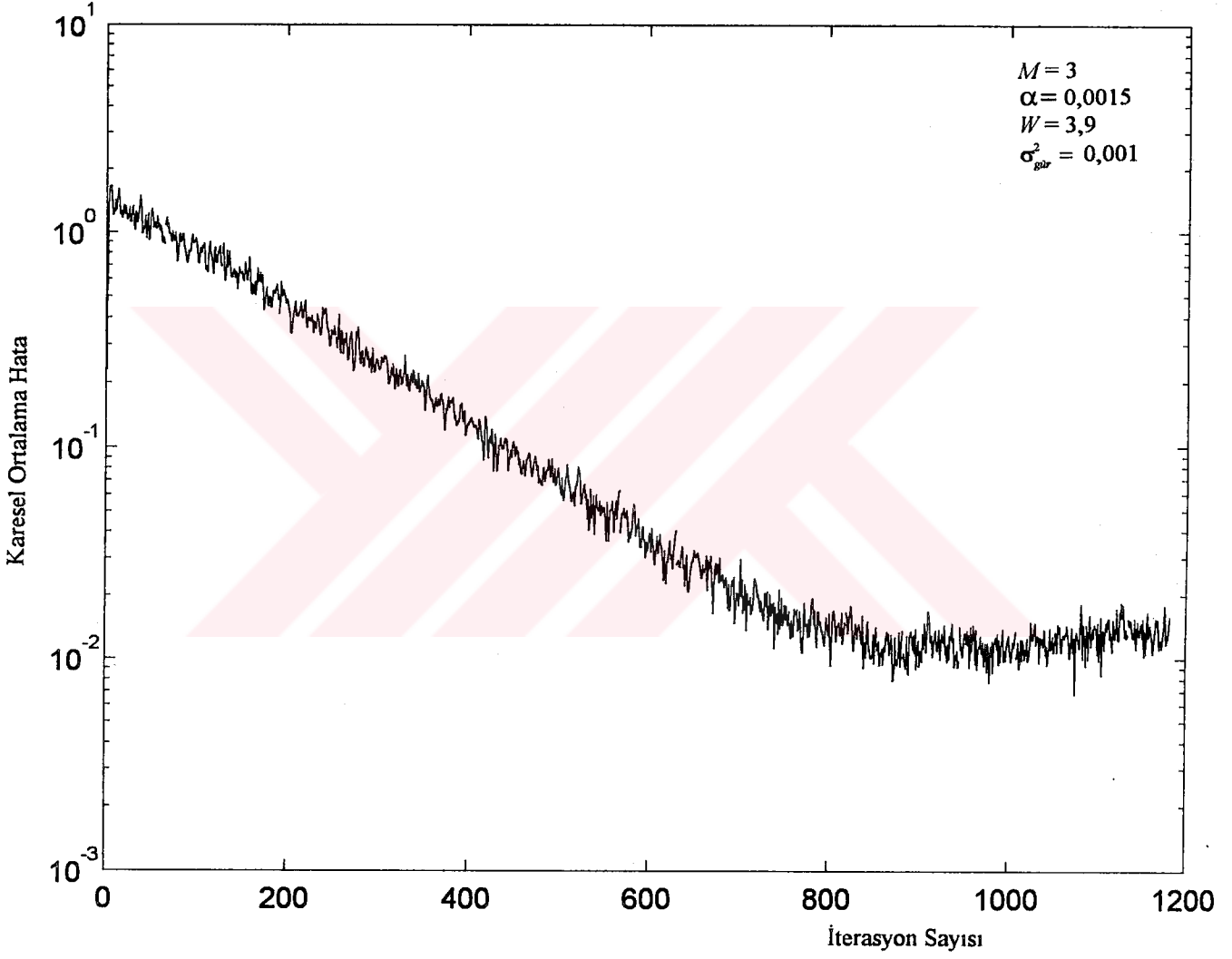
$W = 3,9$ için, Şekil 9.10 (b)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 830 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,020$ ' dir.

$W = 4,9$ için, Şekil 9.10 (c)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 830 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{\min} = 0,030$ ' dir.

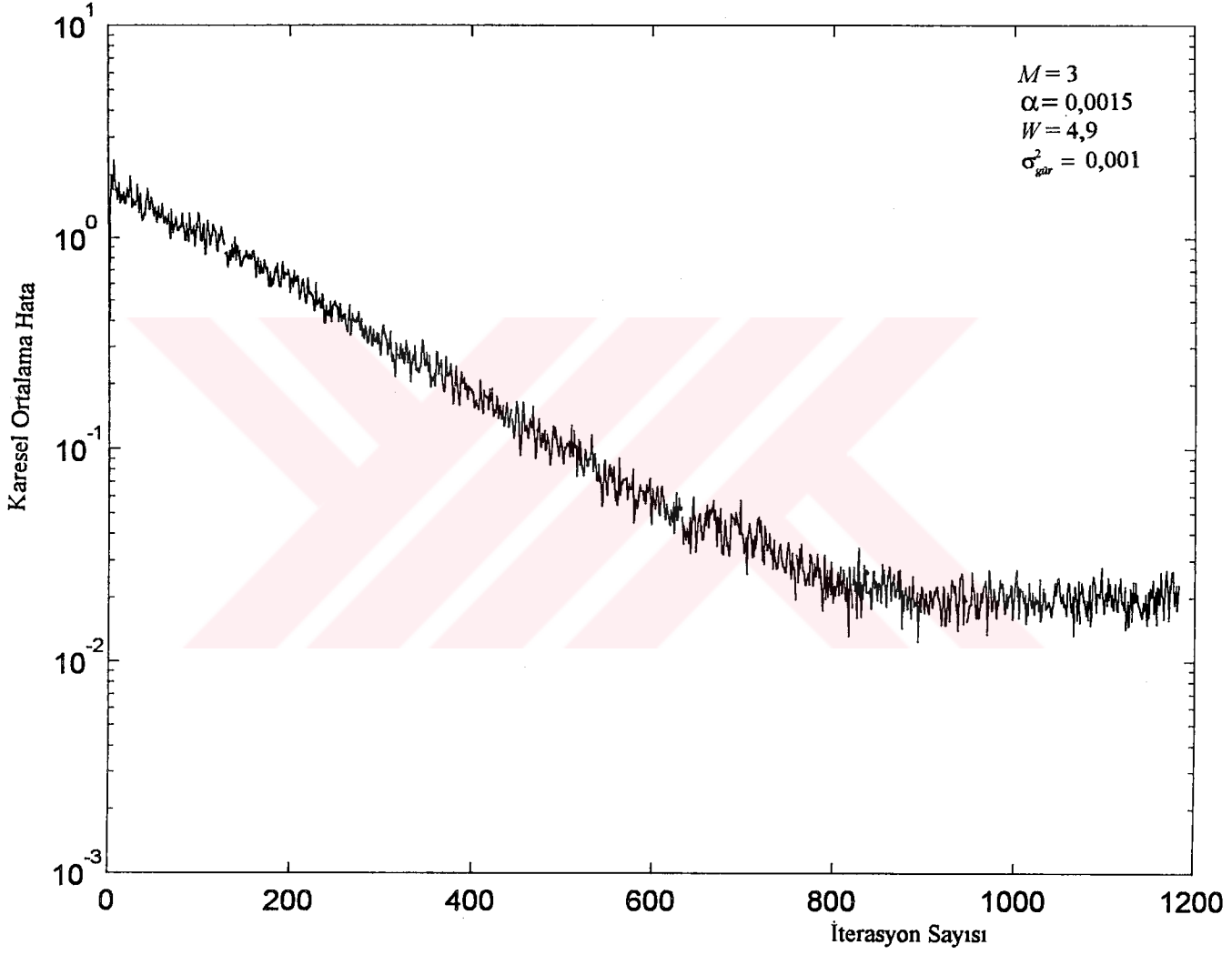
Sonuç olarak; kafes derecesi, adım uzunluğu ve gürültünün değışintisi sabit tutularak kanal parametresi W ' ye göre karesel ortalama hatayı incelediğimizde, kanal parametresi W 'nin artışı ile yakınsama hızında herhangi bir değışim olmazken, kararlı durumda ulaşılan karesel ortalama hatanın çok az bir miktarda büyüdüğü görülmektedir. Ancak, elde edilen sonuçlardan açıkça görülmektedir ki, kanal parametresi W ' nin değışimlerine karşı algoritmanın duyarlılığı oldukça azdır. Kanal parametresi W ' nin artışı ile dikkati çeken bir başka değışim de, karesel ortalama hatanın dalgalanmasının artmasıdır.



Şekil 9.10 (a)



Şekil 9.10 (b)



Şekil 9.10 (c)

9.3.4 Gürültünün Değişintisinin Karesel Ortalama Hataya Etkisinin İncelenmesi

Gürültünün değişintisinin değişimlerine göre algoritma sonuçlarının değişimlerini incelemek üzere diğer parametreler sabit tutulmuştur:

Kafes derecesi $M = 3$

Adım uzunluğu $\alpha = 0,0015$

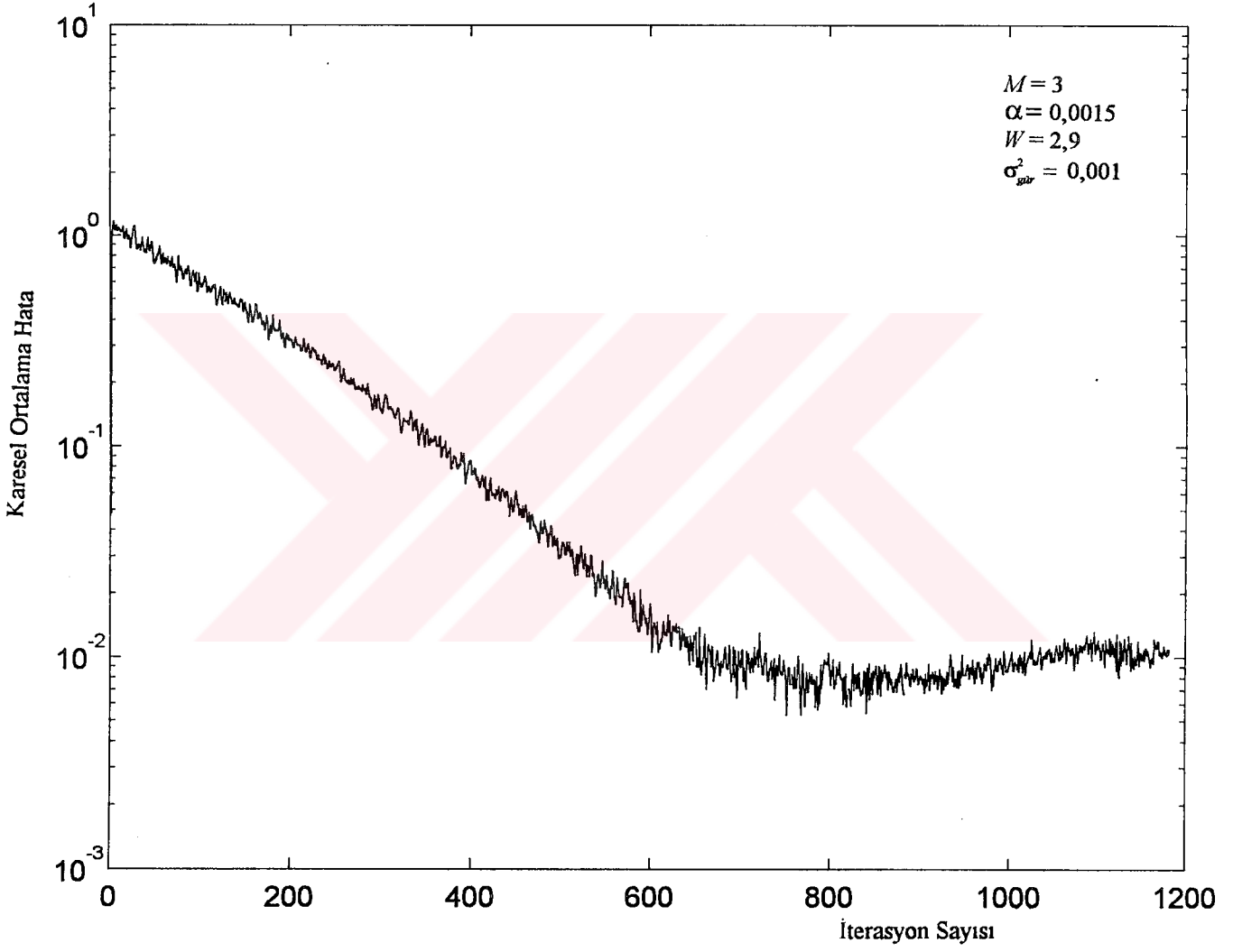
Kanal parametresi $W = 2,9$

$\sigma_{gur}^2 = 0,001$ için, Şekil 9.11 (a)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 820 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{min} = 0,0095$ ' dir.

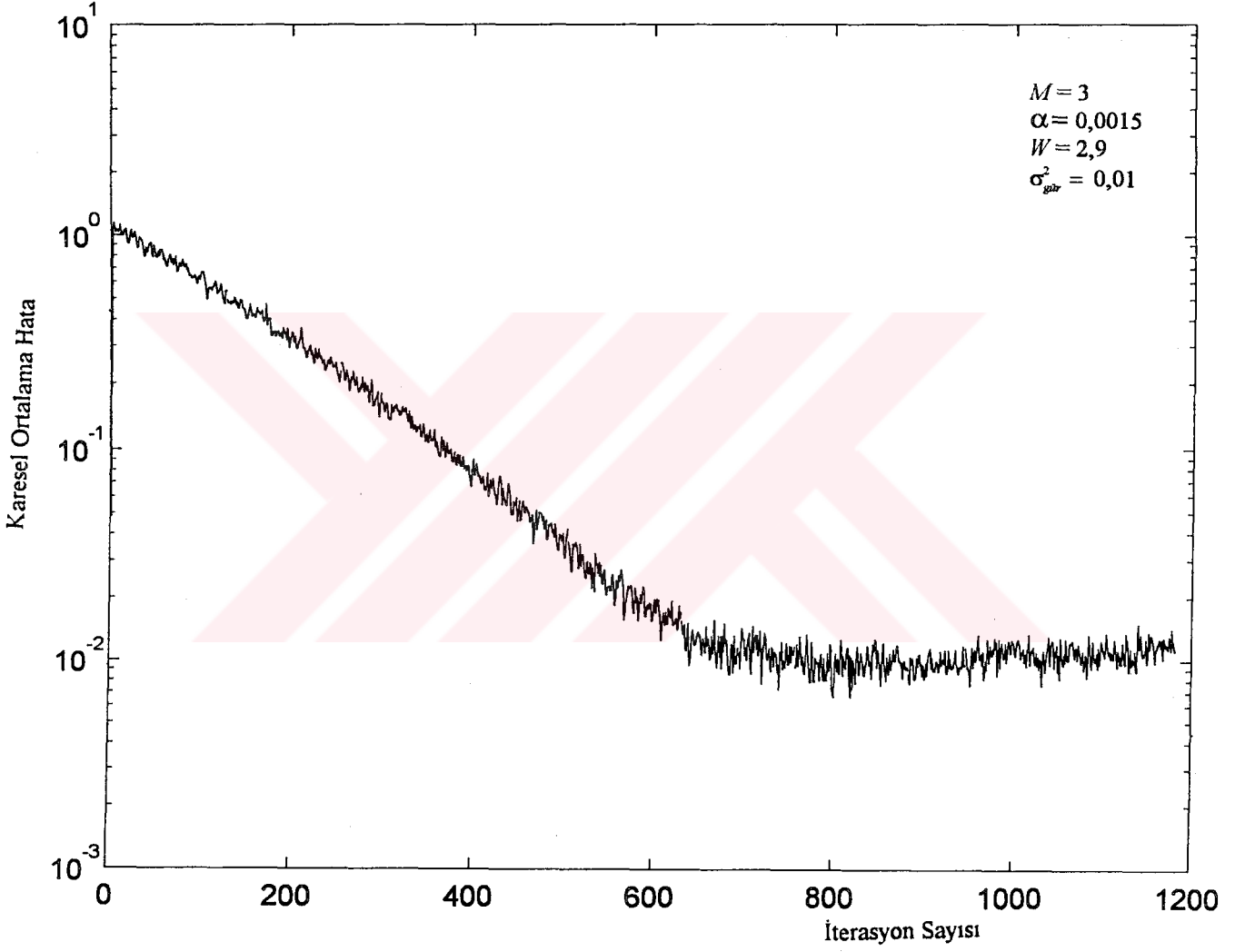
$\sigma_{gur}^2 = 0,01$ için, Şekil 9.11 (b)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 820 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{min} = 0,010$ ' dir.

$\sigma_{gur}^2 = 0,1$ için, Şekil 9.11 (c)' den görüldüğü gibi, karesel ortalama hata kararlı duruma yaklaşık 820 iterasyondan sonra ulaşmıştır ve bu durumdaki minimum karesel ortalama hata değeri $J_{min} = 0,0301$ ' dir.

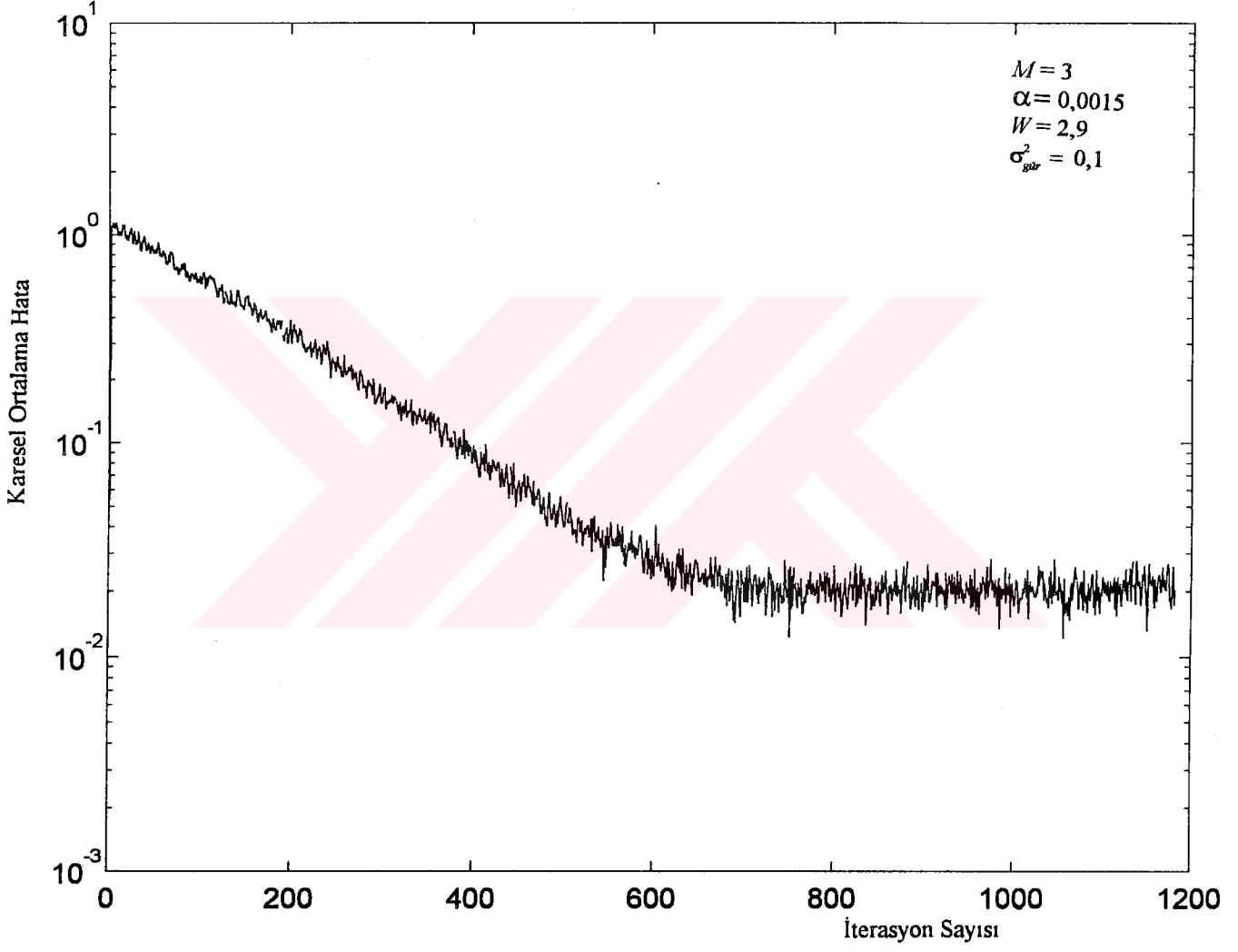
Sonuç olarak; kafes derecesi, adım uzunluğu ve W kanal parametresi sabit tutularak gürültünün değişintisi σ_{gur}^2 ' ye göre karesel ortalama hatayı incelediğimizde, gürültünün değişintisinin artışı ile yakınsama hızında önemli herhangi bir değişimin olmadığı, ancak kararlı durumda ulaşılan karesel ortalama hatanın büyüdüğü görülmektedir.



Şekil 9.11 (a)



Şekil 9.11 (b)



Şekil 9.11 (c)

BÖLÜM 10. SONUÇLAR

Bu tezde, Özyineli En Küçük Kareler Kafes Algoritması kullanarak, doğrusal kanalların özdeşlenmesi incelenmiştir.

Kanal özdeşlemede amaç, önce ilgili kanal parametrelerinin kestirilmesi ve daha sonra bunları kullanılmasıyla alıcının kendisinin iletim ortamına uyarlamasının sağlanmasıdır. Burada, seçilen kanal modelleri için algoritmanın başarımı, karesel ortalama hata bazında incelenmiştir.

Bu çalışmada, bir kanal modeli seçilmiş ve kafes algoritması kullanarak bu kanalın özdeşlenmesi sağlanmıştır. Yapılan bilgisayar simülasyonlarında, kanal, gürültü ve algoritmaya ilişkin bazı parametreler değiştirilerek, algoritmanın başarımı incelenmiştir.

Önce, kafes derecesine göre algoritmanın başarımı ele alınmıştır. Elde edilen simülasyon sonuçlarından, kafes derecesinin artışı ile yakınsama hızının arttığı ve kararlı durumda ulaşılan minimum karesel ortalama hatanın azaldığı gözlenmiştir.

Daha sonra, adım uzunluğuna göre algoritmanın başarımı ele alınmıştır. Elde edilen simülasyon sonuçlarından, adım uzunluğu arttıkça yakınsamanın daha hızlı olduğu, buna karşılık kararlı durumda ulaşılan minimum karesel ortalama hatanın büyüdüğü gözlenmiştir.

W kanal parametresine göre algoritmanın başarımını incelendiğinde, elde edilen sonuçlardan açıkça görülmektedir ki, kanal parametresi W nin değişimlerine karşı algoritmanın duyarlılığı oldukça azdır. Yine de, W nin düşük değerli olması halinde, kararlı durumda ulaşılan minimum karesel ortalama hatanın daha küçük olduğu gözlenmiştir. Kanal parametresi W nin artışı ile dikkati çeken bir başka değişim de, karesel ortalama hatanın dalgalanmasının artması olmuştur.

Son olarak, gürültünün değışintisine göre algoritmanın başarımı ele alınmıştır. Elde edilen simülasyon sonuçlarından, gürültünün değışintisinin artışı ile yakınsama hızında önemli herhangi bir değışimin olmadığı, ancak kararlı durumda ulaşılan minimum karesel ortalama hatanın büyüdüğü gözlenmiştir.

Algoritmanın başarımı ile ilgili olarak kanal, gürültü ve algoritmaya ilişkin bazı parametrelerin değıştirilmesiyle, elde edilen bu simülasyon sonuçları, Özyineli En Küçük Kare Kafes Algoritması kuramsal sonuçları ile uyum göstermektedir.



KAYNAKLAR

- Durbin, J., 1960, "The fitting of time series models", Rev. Int. Stat. Inst., Vol.28, pp.233-244
- Falconer, D.D., Ljung, L., 1978, "Application of Fast Kalman Estimation to Adaptive Equalization", IEEE Trans. Commun., vol.Com-26, no.1, pp. 1439-1446
- Friedlander, B., 1982, "Lattice filters for adaptive processing", Proc. IEEE, Vol.70, pp.829-867
- Griffiths, L.J., 1977, "A continuously-adaptive filter implemented as a lattice structure", Proc. IEEE Intern. Conf. Acoust., Speech and Signal Processing, Hartford, Conn., pp.683-686
- Griffiths, L.J., 1978, "An Adaptive Lattice structure for noise-cancelling applications", Proc. IEEE Intern. Conf. Acoust., Speech, and Signal Processing, Tulsa, Okla., pp.87-90
- Haykin, S., 1991, "Adaptive Filter Theory", Prentice-Hall Inc.
- Kailath, T., 1982, "Time-variant and time-invariant lattice filters for nonstationary processes" in the book entitled Outils et Modeles Mathematique pour L'Automatique, L'Analyse de Systems et le Traitement du Signal, Vol.2, edited by I. Landau, France, CNRS, pp.417-464
- Lee, D.T.L., Morf, M., Friedlander, B., 1981, "Recursive Least-Squares Ladder Estimation Algorithms", IEEE Trans.Circuit and Systems, Vol.CAS-28, pp.467-481
- Lev-Ari, H.T., Kailath, T., Cioffi, J., 1984, "Least-Squares Adaptive Lattice and Transversal Filters: A unified geometric theory", IEEE Trans. Information Theory, Vol.IT.30, pp.222-236
- Levinson, L., 1947, "The Wiener RMS (Root-mean-square) error criterion in filter design and prediction", J. Math Phys., Vol.25, pp.261-278
- Ling, F., Manolakis, D., Proakis, J.G., 1985, "New forms of LS lattice algorithms and an analysis of their round-off error characteristics", Proceedings ICASSP 85, Tampa, Fla., pp.1739-1742
- Makhoul, J., 1977, "Stable and Efficient Lattice Methods for Linear Prediction", IEEE Trans. Acoustics, Speech, and Signal Processing, Vol.ASSP-25, no.5, pp.423-428
- Morf, M., 1974, "Fast Algorithms for Multivariable Systems", Ph. D. dissertation, Stanford University, Stanford, Calif.

Moff, M., Viera, A., Lee, D.T., 1977, "Ladder forms for identification and speech processing", Proc. IEEE Conf. Decision and Control, New Orleans, pp.1074-1078

Morf, M., Lee, D.T., 1978, "Recursive least-squares ladder forms for fast parameter tracking", Proc. Conf. Decision and Control, San Diego, Calif., pp.1362-1367

Porat, B., Friedlander, B., Morf, M., 1982, "Square root covariance ladder algorithms", IEEE Trans. Automatic Control, Vol.AC-27, pp.813-829

Porat, B., Kailath, T., 1983, "Normalized lattice algorithms for least-squares FIR system identification", IEEE Trans. Acoust., Speech, and Signal Processing, Vol.ASSP-31, pp.122-128

Satorius, E.H., Alexander, S.T., 1979, "Channel Equalization using Adaptive Lattice Algorithms", IEEE Trans. on. Commun., Vol.COM-27, pp.899-905

Satorius, E.H., Pack, J.D., 1981, "Application of Least Squares Lattice Algorithms to Adaptive Equalization", IEEE Trans. Commun., Vol.COM-29, pp.136-142

Widrow, B., 1970, "Adaptive Filters in R. E. Kalman ve N. DeClaris, eds, Aspect of Network and System Theory", Holt, Rinehart, Winston, New York

EK**MATLAB ortamında hazırlanan Kanal Özdeşleme Bilgisayar Simülasyonu**

```

clear
S = input (" iterasyon sayısını giriniz:")
M = input (" modül sayısını giriniz:")
A = input (" adım uzunluğunu giriniz:")
W = input ("kanal parametresini giriniz:")
var = input ("gürültünün varyasını giriniz:")
den = input (" bağımsız denemeler sayısını giriniz:")

for n=1:S
J(n)=0;
end

for b=1:den

for n=1:S
    H(n)=0;
    if 1<=n
        if n<=3
             $H(n) = (1/2) * (1 + \cos(2 * \pi * (n-2) / W))$ ;
        end
    end
end
end

% H(1)=0.2;
% H(2)=0.4;
% H(3)=0.6;
% H(4)=0.8;
% H(5)=1;

% H(1)=0.2;
% H(2)=0.3;
% H(3)=1;
% H(4)=0.3;
% H(5)=0.2;

% H(1)=0.2;
% H(2)=0.3;
% H(3)=0.4;

```



```
% H(4)=1;  
% H(5)=0.4;  
% H(6)=0.3;  
% H(7)=0.2;
```

```
for n=1:S  
    L(n)=rand;  
    if L(n)<0.5  
        U(n)=1;  
    else  
        U(n)=-1;  
    end  
end
```

```
X=conv(U,H);
```

```
for n=1:(S/2)  
    R1(n)=rand;  
    R2(n)=rand;  
    S1(n)=sqrt(-2*log(R1(n)))*sin(2*pi*R2(n));  
    S2(n)=sqrt(-2*log(R2(n)))*cos(2*pi*R1(n));  
    N(2*n-1)=var*S1(n);  
    N(2*n)=var*S2(n);  
end
```

```
for n=1:S  
    Y(n)=X(n)+N(n);  
end
```

```
for m=1:M+1  
    KF(m,1)=0;  
    KB(m,1)=0;  
    TF(m,1)=1;  
    TB(m,1)=1;  
    KF(m,2)=0;  
    KB(m,2)=0;  
    TF(m,2)=1;  
    TB(m,2)=1;  
end
```

```
for n=1:S  
    F(1,n)=U(n);  
    B(1,n)=U(n);
```

end

for n=2:S

for m=1:M

B(m,1)=0;

F(m+1,n)=F(m,n)-KF(m+1,n)*B(m,n-1);

B(m+1,n)=B(m,n-1)-KB(m+1,n)*F(m,n);

HF(m+1,n)=A/TF(m+1,n);

HB(m+1,n)=A/TB(m+1,n);

TF(m+1,n+1)=(1-A)*TF(m+1,n)+A*B(m,n-1)*B(m,n-1);

TB(m+1,n+1)=(1-A)*TB(m+1,n)+A*F(m,n)*F(m,n);

KF(m+1,n+1)=KF(m+1,n)+2*HF(m+1,n)*B(m,n-1)*F(m+1,n);

KB(m+1,n+1)=KB(m+1,n)+2*HB(m+1,n)*F(m,n)*B(m+1,n-1);

end

end

for m=1:M+1

G(m,1)=0;

TG(m,1)=1;

end

for n=1:S

V(1,n)=Y(n);

for m=1:M

V(m+1,n)=V(m,n)-G(m,n)*B(m,n);

HG(m,n)=A/TG(m,n);

TG(m,n+1)=(1-A)*TG(m,n)+A*V(m+1,n)*V(m+1,n);

G(m,n+1)=G(m,n)+2*HG(m,n)*B(m,n)*V(m+1,n);

end

end

for n=1:S

Z(1,n)=G(1,n)*B(1,n);

end

for n=1:S

for m=1:M

Z(m+1,n)=Z(m,n)+G(m+1,n)*B(m+1,n);

end

YK(n)=Z(M+1,n);

```
end

for n=1:S
    J(n)=J(n)+(Y(n)-YK(n))^2;
end
end
for n=1:S
    J(n)=(1/den)*J(n);
end

semilogy(J)
```



ÖZGEÇMİŞ

Adı Soyadı : S. Berrin ÖNER

Doğum Tarihi : 16 / 07 / 1973

Doğum Yeri : Siirt

Öğrenim Gördüğü Okullar :

1979 - 1984 Trabzon Hasan Tahsin Kırall İlkokulu

1984 - 1990 Mersin Dumlupınar Lisesi

1990 - 1994 Yıldız Teknik Üniversitesi, Elektrik-Elektronik Fak.,
Elektronik ve Haberleşme Müh. Bölümü

1995 yılı Ocak ayından itibaren Yıldız Teknik Üniversitesi, Elektronik ve Haberleşme Mühendisliği Bölümü, Haberleşme Anabilim Dalında Araştırma Görevlisi olarak çalışmaktadır.