T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

SİSTEM TANIMA VE UYARLAMALI KONTROL UYGULAMALARINDA TEKRARLAMALI DOĞRUSAL DENKLEM TAKIMI ÇÖZÜM ALGORİTMALARI

Metin HATUN

DOKTORA TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

BURSA-2008

T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

SİSTEM TANIMA VE UYARLAMALI KONTROL UYGULAMALARINDA TEKRARLAMALI DOĞRUSAL DENKLEM TAKIMI ÇÖZÜM ALGORİTMALARI

Metin HATUN

Yrd. Doç. Dr. Osman Hilmi KOÇAL
(Danışman)

DOKTORA TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

BURSA-2008

T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

SİSTEM TANIMA VE UYARLAMALI KONTROL UYGULAMALARINDA TEKRARLAMALI DOĞRUSAL DENKLEM TAKIMI ÇÖZÜM ALGORİTMALARI

Metin HATUN

DOKTORA TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Bu Tez 27/11/2008 tarihinde aşağıdaki jüri tarafından oybirliği/~~oyçokluğu~~ ile kabul edilmiştir.

Yrd. Doç. Dr. Osman Hilmi KOÇAL
Danışman

Prof. Dr. Erdoğan DİLAVEROĞLU

Prof. Dr. İrfan KARAGÖZ

Prof. Dr. Abdurrahman KARAMANCIOĞLU

Prof. Dr. İbrahim YÜKSEL

ÖZET

Bu tez çalışmasında, sistem tanıma ve uyarlamalı kontrol uygulamalarında kullanmak için, tekrarlamalı Gauss-Seidel, tekrarlamalı Jacobi ve tekrarlamalı SOR yöntemleri önerilmiştir. Önerilen bu yöntemler ayrık-zaman sistem tanıma, sürekli-zaman sistem tanıma, bazı zaman-serileri modellerinin tahmin edilmesi, Volterra model parametrelerini tahmin edilmesi, uyarlamalı denetleyici parametrelerinin dolaylı olarak ayarlanması ve referans model tabanlı uyarlamalı denetleyici katsayılarının doğrudan ayarlanmasında kullanılmıştır. Ayrıca, önerilen Gauss-Seidel algoritmasının stokastik yakınsama analizi yapılmıştır. Matlab yazılımı kullanılarak yapılan benzetim çalışmalarıyla, elde edilen sonuçlar eşdeğer algoritmalarla elde edilen sonuçlarla karşılaştırılmıştır. Elde edilen benzetim sonuçlarına göre, önerilen tekrarlamalı algoritmaların eşdeğer RLS tabanlı algoritmalara çok yakın sonuçlar verdiği görülmüştür. Önerilen algoritmaların RLS ve diğer RLS tabanlı algoritmalara alternatif olarak kullanılabileceği yapılan benzetim çalışmalarında görülmüştür.

Anahtar Kelimeler: Gauss-Seidel algoritması, sistem tanıma, parametre tahmini, yakınsama analizi, uyarlamalı kontrol.

ABSTRACT

In this thesis, recursive Gauss-Seidel, recursive Jacobi and recursive SOR algorithms are proposed for use in system identification and adaptive control applications. The proposed methods are used for discrete-time system identification, continuous-time system identification, estimation of various time-series model parameters, estimation of Volterra model parameters, indirect tuning of adaptive controller parameters and direct tuning of reference model based adaptive controller parameters. Also, stochastic convergence analysis of the proposed Gauss-Seidel algorithm is performed. By computer simulations using Matlab software, the obtained results are compared with the results obtained by using equivalent algorithms. According to simulation results obtained, it is seen that the proposed recursive algorithms produce very close results obtained by equivalent RLS based algorithms. By using the simulations it is also seen that the proposed algorithms can be used alternatively to the RLS and the other RLS based algorithms.

Key Words: Gauss-Seidel algorithm, system identification, parameter estimation, convergence analysis, adaptive control.

İÇİNDEKİLER

KISALTMALAR DİZİNİ

ARMA	-	Auto-Regressive model with Moving-Average noise input
ARX	-	Auto-Regressive model with eXogeneous control input
ARMAX	-	Auto-Regressive model with Moving-Average noise and eXogeneous control input
ARARX	-	Auto-Regressive model with Auto-Regressive noise and eXogeneous control input
ARARMAX	-	Auto-Regressive model with Auto-Regressive-Moving-Average noise and eXogeneous control input
EDS	-	Euclidean Direction Search
ELS	-	Extended Least Squares
FIR	-	Finite Impulse Response
GMV	-	Generalized Minimum Variance – Genelleştirilmiş Minimum Varyans
GS	-	Gauss-Seidel
GSYD	-	Gauss-Seidel Yardımcı Değişkenler
IIR	-	Infinite Impulse Response
LMS	-	Least Mean Squares
MSE	-	Mean Square Error
NLMS	-	Normalized Least Mean Squares
PID	-	Proportional Integral Derivative
RLS	-	Recursive Least Squares
SOR	-	Successive Over-Relaxation
YD	-	Yardımcı Değişkenler

ÇİZELGELER DİZİNİ

ŞEKİLLER DİZİNİ

SİMGELER DİZİNİ

$A(z)$	-	Sistemin transfer fonksiyonu modelinin payda polinomu
$A(\lambda)$	-	Değiştirilmiş sürekli-zaman sistemin payda polinomu
$A_m(z)$	-	Referans modelin transfer fonksiyonu modelinin payda polinomu
$B(z)$	-	Sistemin transfer fonksiyonu modelinin pay polinomu
$B(\lambda)$	-	Değiştirilmiş sürekli-zaman sistemin pay polinomu
$B_m(z)$	-	Referans modelin transfer fonksiyonu modelinin pay polinomu
$C(z)$	-	Gürültü modelinin pay polinomu
$D(z)$	-	Gürültü modelinin payda polinomu
$E[\cdot]$	-	Beklenen değer operatörü
α , β	-	Sistemin baskın kompleks eşlenik köklerinin reel ve sanal kısımları
$\varepsilon(k)$	-	k anındaki tahmin hatasının
$e(k)$	-	Beyaz gürültü dizisinin k. elemanı
$v(k)$	-	Renkli gürültü dizisinin k. elemanı
$u(k)$	-	Sistemin giriş işaretinin k anındaki değeri, sistemin kontrol girişi
$x(k)$	-	Sistemin gürültüsüz çıkış işaretinin k anındaki değeri
$\hat{x}(k)$	-	Gürültüsüz çıkış işaretinin k anındaki tahmini değeri (yardımcı değişken)
$y(k)$	-	Sistemin gürültülü çıkış işaretinin k anındaki değeri
k	-	Ayrık-zaman indisi
K_u	-	Kapalı-çevrim sistemi kararsızlık sınırında çalıştıran oransal kazanç
K_p	-	Sürekli-zaman PID denetleyicinin oransal kazancı
K_i , T_i	-	Sürekli-zaman PID denetleyicinin integral kazancı
K_d , T_d	-	Sürekli-zaman PID denetleyicinin türev kazancı
K_P, K_I, K_D	-	Ayrık-zaman PID denetleyicinin oransal, integral, türev kazançları
q_0, q_1, q_2, γ	-	Ayrık-zaman PID denetleyicinin fark denkleminin katsayıları
$G(z^{-1})$	-	Ayrık-zaman sistemin transfer fonksiyonu gösterimi
$C(z^{-1})$	-	Denetleyicinin transfer fonksiyonu gösterimi
$D(s), D(z^{-1})$	-	Kapalı-çevrim sistemin karakteristik denklemi
M	-	Tahmin edilen parametre sayısı, uyarlamalı filtre uzunluğu
N	-	Kullanılan veri miktarı
m, n	-	Sistemin transfer fonksiyonunun pay ve payda polinomlarının dereceleri
μ	-	Kullanılan algoritmalardaki adım büyüklüğü
θ_{opt}	-	Optimum parametre vektörü

$\hat{\theta}(k)$	-	k anındaki parametre tahminleri vektörü
$\hat{\theta}_j(k)$	-	k anındaki parametre tahminleri vektörünün $j.$ elemanı
$\widetilde{\theta}(k)$	-	k anındaki parametre tahmin hatası vektörü
$\hat{\boldsymbol{h}}(k)$	-	Volterra modelin k anındaki parametre tahminleri vektörü
$\boldsymbol{h}_{opt}$	-	Volterra modelin optimum parametre vektörü
$h_j(k)$	-	Volterra modelin k anındaki parametre tahminleri vektörünün $j.$ elemanı
h_0	-	Volterra modelin sabit bileşeninin katsayısı
$h_1(n_1)$	-	Volterra modelin 1. derece (doğrusal) bileşenlerinin katsayıları
$h_2(n_1,n_2)$	-	Volterra modelin 2. derece (karesel) bileşenlerinin katsayıları
$h_3(n_1,n_2,n_3)$	-	Volterra modelin 3. derece (kübik) bileşenlerinin katsayıları
$h_l(n_1,\ldots,n_l)$	-	Volterra modelin $l.$ derece bileşenlerinin katsayıları
$\varphi(k)$	-	Kullanılan algoritmalardaki veri vektörü
$\phi(k)$	-	Kullanılan algoritmalardaki veri vektörü
$z(k)$	-	Yardımcı modelin veri vektörü
$\boldsymbol{x}(k)$	-	Volterra modelin veri vektörü
$P(k)$	-	Korelasyon matrisinin tersinin k anındaki değeri
R	-	Korelasyon matrisinin gerçek değeri
$R(k)$	-	Korelasyon matrisinin k adet veri kullanılarak tahmin edilmiş değeri
$R_L(k)$	-	Korelasyon matrisinin alt üçgen elemanlarını içeren, diğer elemanları sıfır olan kare matris
$R_D(k)$	-	Korelasyon matrisinin köşegen üzerindeki elemanlarını içeren, diğer elemanları sıfır olan kare matris
$R_U(k)$	-	Korelasyon matrisinin üst üçgen elemanlarını içeren, diğer elemanları sıfır olan kare matris
$R_{ij}(k)$	-	Korelasyon matrisinin $i.$ satırındaki ve $j.$ sütunundaki elemanı
p	-	Korelasyon vektörünün gerçek değeri
$p(k)$	-	Korelasyon vektörünün k adet veri kullanılarak tahmin edilmiş değeri
$p_i(k)$	-	Korelasyon vektörünün $i.$ elemanı
p,q	-	Alçak geçiren filtrenin fark denkleminin katsayıları
$r_v(k)$	-	Renkli gürültü dizisinin $k.$ oto-korelasyon katsayısı
$R(z), S(z), T(z)$	-	Model tabanlı kutup yerleştirmeli uyarlamalı denetleyici polinomları
$P(z), Q(z), Q'(z)$	-	Uyarlamalı PID denetleyici polinomları
T , T_s	-	Örnekleme periyodu
T , $T(k)$	-	Tekrarlamalı Gauss-Seidel algoritmasının iterasyon (sistem) matrisi
T_u	-	Kararsızlık sınırındaki kapalı-çevrim sistemin çıkış işaretindeki sabit genlikli salınımın periyodu
ω_u	-	Kararsızlık sınırındaki kapalı-çevrim sistemin çıkış işaretindeki

		sabit genlikli salınımın frekansı
λ_i	-	Korelasyon matrisinin $i.$ özdeğeri
$\lambda_{\max}$	-	Korelasyon matrisinin en büyük özdeğeri
λ , ƛ	-	Unutma faktörü
$\lambda(s)$	-	Alçak geçiren filtre operatörü (transfer fonksiyonu gösterimi)
$\lambda^i u(k)$	-	i defa filtrelenmiş giriş işaretinin k anındaki değeri
$\lambda^i x(k)$	-	i defa filtrelenmiş gürültüsüz çıkış işaretinin k anındaki değeri
$\lambda^i y(k)$	-	i defa filtrelenmiş gürültülü çıkış işaretinin k anındaki değeri
τ , a	-	Alçak geçiren filtrenin zaman sabiti ve kutbu
σ_e^2	-	Beyaz gürültü dizisinin varyansı
σ_v^2	-	Renkli gürültü dizisinin varyansı
$V_k(\hat{\theta})$	-	Ortalama karesel hata fonksiyonunun k anındaki değeri
$V_{\min}$	-	Ortalama karesel hata fonksiyonunun en küçük değeri
$V(k)$	-	Karesel Lyapunov fonksiyonu
$\Delta V(k)$	-	Karesel Lyapunov fonksiyonunun farkı (türevi)

1. GİRİŞ

Sistem tanıma işlemi, gerçek zamanda gerçeklenen birçok uyarlamalı işaret işleme ve uyarlamalı kontrol uygulamasının temelini oluşturmaktadır. Bu uygulamalarda uygulama alanına göre sistemin kendisi veya tersi uygun bir parametrik model kullanılarak modellenmekte ve bu modelin katsayıları bilinen bir iteratif algoritma ile gerçek zamanda alınan örnekler kullanılarak tahmin edilmektedir. Parametrik sistem modellerini deneysel veriler kullanılarak gerçek zamanda tahmin etmek için bir çok algoritma geliştirilmiştir. Bu konuda yayınlanan literatüre ve bilimsel çalışmalara göre, kullanılan algoritmaların işlem karmaşıklığı ve tahmin edilen parametrelerin yakınsama özellikleri, tahmin edilen parametrelerin kullanıldıkları uygulama alanlarına göre algoritma seçiminde birer tercih nedeni olmaktadır. Örneğin uyarlamalı işaret işleme uygulamalarında hız açısından çoğunlukla işlem karmaşıklığı az olan algoritmalar tercih edilirken, sistem belirleme ve uyarlamalı kontrol uygulamalarında yakınsama özellikleri iyi olan algoritmalar daha çok tercih edilmektedir. Bu uyarlamalı algoritmalar eğim tabanlı algoritmalar ve en küçük kareler tabanlı algoritmalar olmak üzere kabaca iki gruba ayrılabilir.

LMS (Least Mean Squares) algoritması ve türevleri en yaygın kullanılan eğim tabanlı algoritmalardır. Bu algoritmalar en dik düşüm (steepest descent) optimizasyon algoritması üzerine kuruludur. Düşük işlem yüküne ve kolay gerçekleme avantajına sahip olan bu gruptaki algoritmalar uyarlamalı işaret işleme uygulamalarında daha çok tercih edilmektedir. Stokastik eğim algoritması olarak da adlandırılan bu gruptaki algoritmalar yakınsama hızı düşük olduğu için sistem tanıma ve uyarlamalı kontrol uygulamalarında daha az tercih edilmektedir.

En küçük kareler tabanlı algoritmalar ise eğim tabanlı algoritmalara göre çok daha yoğun işlem yüküne sahip olmasına rağmen yakınsama özellikleri eğim tabanlı algoritmalara göre çok daha iyi olduğu için sistem tanıma ve uyarlamalı kontrol uygulamalarında daha çok tercih edilmektedir. Tekrarlamalı En Küçük Kareler (RLS – Recursive Least Squares) algoritması ve türevleri ise sistem tanıma ve uyarlamalı kontrol uygulamalarında en yaygın kullanılan algoritmadır. RLS algoritmasının bazı dezavantajlarını ortadan kaldırmak için önerilen RLS tabanlı diğer algoritmalarda ve Gauss-Newton optimizasyon algoritması üzerine kurulu olan diğer en küçük kareler tabanlı algoritmalarda ise işlem yükü RLS algoritmasına göre oldukça artmaktadır. RLS benzeri bu algoritmalar, işlem yükü RLS algoritmasına göre daha fazla olmasına rağmen, yakınsama özellikleri çok iyi olduğu için sistem tanıma ve uyarlamalı kontrol uygulamalarında yaygın olarak kullanılmaktadır.

Yukarıda bahsedilen LMS tabanlı algoritmalar ve RLS tabanlı algoritmalar arasında hem yakınsama hızı ve hem de işlem yükü açısından belirgin bir fark bulunmaktadır. Ayrıca, LMS ve RLS tabanlı algoritmalara alternatif olarak, doğrusal denklem sistemlerinin iteratif çözüm yöntemleri üzerine kurulu olan bazı yeni algoritmalar önerilmiştir ve bazı uyarlamalı işaret işleme uygulamalarında kullanılmıştır. Yapılan uyarlamalı işaret işleme çalışmalarında GS (Gauss-Seidel) algoritması uyarlamalı FIR (Finite Impulse Response) filtre katsayılarının ayarlanmasında kullanılmıştır. LMS ve RLS tabanlı algoritmalara alternatif olan ve bir ara yöntem olarak önerilen bu yeni algoritmalar LMS tabanlı algoritmalara göre yüksek yakınsama hızına ve RLS tabanlı algoritmalara oranla daha az işlem yüküne sahiptir. Bu yeni algoritmaların en önemli avantajı bahsedilen iki algoritma grubunun düşük işlem yükü ve yüksek yakınsama hızı özelliklerini birleştirmesidir. Bu algoritmalar temelde normal denklemin GS iterasyonlarıyla çözümü üzerine kurulu olup, yakınsama hızı açısından FIR filtrenin giriş işaretine ait korelasyon matrisinin özdeğer yayılımına bağımlı olmasına rağmen eğim tabanlı algoritmalara oranla çok daha yüksek bir yakınsama hızına sahiptir ve korelasyon matrisinin özdeğer yayılımının küçük olması durumunda RLS algoritmasına çok yakın sonuçlar vermektedir. GS algoritması temelde RLS algoritması gibi korelasyon matrisinin birikimli değerlerini kullanan tekrarlamalı bir algoritma olup, korelasyon matrisinin pozitif tanımlılığı sağlandığı sürece kararlılığı garantilenmiş olmaktadır.

Bu tezin amacı, daha önce uyarlamalı işaret işleme uygulamalarında FIR filtre katsayılarını güncellemek için kullanılan ve düşük işlem yükü, yüksek yakınsama hızı ve kararlılık garantisi avantajlarına sahip olan GS algoritmasını sistem tanıma ve uyarlamalı kontrol uygulamalarına aktarmaktır.

Bu tez çalışmasında, öncelikle GS algoritması transfer fonksiyonu parametrelerinin tekrarlamalı olarak tahmin edilmesinde kullanılmış ve elde edilen parametre tahminlerinin stokastik yakınsama analizi yapılmıştır. Aynı yöntem sürekli zaman sistem parametrelerinin tahmin edilmesi için de kullanılmıştır. Bu yöntemlerde sistemin çıkış işaretine karışan renkli ölçme gürültüsünün istenmeyen etkisinden kurtulmak ve yansız parametre tahminlerini elde etmek için GS algoritmasıyla birlikte yardımcı değişkenlerden yararlanılmıştır. GS algoritması ayrıca, ayrık zaman serileri modellerinin parametrelerinin çevrim-içi tahmin edilmesinde de kullanılmıştır. Elde edilen karşılaştırmalı benzetim çalışmalarında GS algoritmasının, düşük işlem yükü avantajının yanında yakınsama hızı açısından RLS tabanlı eşdeğer algoritmalara çok yakın sonuçlar verdiği görülmüştür.

Özellikle GS algoritmasının parametre sayısı arttıkça önemi artan düşük işlem yükü avantajı, tahmin edilen parametre sayısının fazla olduğu doğrusal olmayan Volterra model parametrelerinin tahmin edilmesinde artan parametre sayısına bağlı olarak artan işlem yükünü azaltmak amacıyla kullanılmıştır. Yapılan karşılaştırmalı benzetim çalışmalarında elde edilen sonuçlarda, GS algoritmasının yine düşük işlem yükü avantajının yanında yakınsama hızı açısından RLS tabanlı eşdeğer algoritmalara çok yakın sonuçlar verdiği ve LMS tabanlı algoritmalara göre yakınsama hızının çok iyi olduğu görülmüştür.

Yine aynı amaçla, GS algoritması tahmin edilen parametre sayısının fazla olduğu çok girişli çok çıkışlı sistemlerin transfer fonksiyonu matrisi parametrelerinin tahmin edilmesinde kullanılmıştır. GS algoritması, hem ayrık-zaman sistemlerin hem de sürekli-zaman sistemlerin transfer fonksiyonu matrisi parametrelerinin tahmin edilmesinde kullanılmıştır. Renkli ölçme gürültüsünün olumsuz etkisinden kurtulmak ve yansız parametre tahminlerini elde etmek için GS algoritmasıyla birlikte yardımcı değişkenlerden faydalanılmıştır. Yapılan karşılaştırmalı benzetim çalışmalarında elde edilen sonuçlarda, GS algoritmasının yine düşük işlem yükü avantajının yanında yakınsama hızı açısından RLS tabanlı eşdeğer algoritmalara çok yakın sonuçlar verdiği görülmüştür.

Daha sonra, GS algoritması, hem dolaylı uyarlamalı kontrol sistemlerinde sistem parametrelerinin çevrim-içi tahmin edilmesinde, hem de doğrudan uyarlamalı kontrol sistemlerinde denetleyici parametrelerinin doğrudan ayarlanmasında kullanılmıştır. Ayrıca, GS algoritmasının doğrudan uyarlamalı kontrol sistemlerinde denetleyici parametrelerinin doğrudan ayarlanmasında kullanılması durumunda elde edilen kapalı çevrim sistemin kararlılık analizi yapılmış ve Lyapunov kararlılık kriterine göre kapalı çevrim sistemin kararlı olduğu gösterilmiştir. Yapılan karşılaştırmalı benzetim çalışmalarıyla GS algoritmasının uyarlamalı kontrol sistemlerindeki performansı eşdeğer algoritmalarla birlikte karşılaştırmalı olarak incelenmiştir. Elde edilen sonuçlarda önerilen GS algoritmasının düşük işlem yükü avantajının yanında RLS tabanlı eşdeğer algoritmalara çok yakın sonuçlar verdiği görülmüştür.

2. KURAMSAL TEMELLER

Çevrim-içi sistem tanıma işlemi, gerçek zamanda gerçeklenen birçok uyarlamalı işaret işleme ve uyarlamalı kontrol uygulamasının temelini oluşturmaktadır (Johnson 1982, Widrow ve Stearns 1985, Haykin 2002, Widrow ve Haykin 2003, Landau 1979, Egardt 1979, Goodwin ve Sin 1984, Harris ve Billings 1985, Gupta ve Chen 1986, Chalam 1987, Narendra ve Annaswamy 1989, Wellstead ve Zarrop 1991, Isermann ve ark. 1992, Butler 1992, Aström ve Wittenmark 1995, Landau 1990, Landau ve ark. 1998, Ikonen ve Najim 2002, Tao 2003, Bobal ve ark. 2005, Landau ve Zito 2006, Ioannou ve Fidan 2006, Mikles ve Fikar 2007, Sanchez-Pena ve ark. 2007, Moudgalya 2007). Özellikle parametrik sistem modellerinin deneysel veriler kullanılarak gerçek zamanda tahmin edilmesi işlemi ve bu işlem için kullanılan uyarlamalı algoritmalar geçmişten günümüze artan bir ilgi çekmiştir (Aström ve Eykhoff 1971, Mendel 1973, Eykhoff 1974, Goodwin ve Payne 1977, Hsia 1977, Biermann 1977, Ljung ve Söderström 1983, Sinha ve Kuszta 1983, Young 1984, Norton 1986, Johnson 1988, Söderström ve Stoica 1983,1989, Johansson 1993, Kalouptsidis ve Theodoridis 1993, Walter ve Pronzato 1997, Pintelon ve Schoukens 2001, Raol ve ark. 2004, Ljung 1999, 2008). Bu konuda yayınlanan literatüre ve bilimsel çalışmalara göre, kullanılan algoritmaların işlem karmaşıklığı ve tahmin edilen parametrelerin yakınsama özellikleri, tahmin edilen parametrelerin kullanıldıkları uygulama alanlarına göre algoritma seçiminde birer tercih nedeni olmaktadır. Örneğin uyarlamalı işaret işleme uygulamalarında hız açısından çoğunlukla işlem karmaşıklığı az olan algoritmalar tercih edilirken, uyarlamalı sistem tanıma ve kontrol uygulamalarında yakınsama özellikleri iyi olan algoritmalar daha çok tercih edilmektedir.

Uyarlamalı sistem tanıma ve kontrol uygulamalarında yaygın olarak kullanılan RLS algoritmasının yakınsama özellikleri oldukça iyi olmasına karşın, sistemin çıkış işaretine karışan ölçme gürültüsünün renkli gürültü olması durumunda yanlı parametre tahminleri elde edilmektedir (Sinha ve Kuszta 1983, Söderström ve Stoica 1989, Ljung

1999). Bu dezavantajdan kurtulmanın ve yansız parametre tahminlerini elde etmenin bir yolu yardımcı değişkenlerden yararlanmaktır. Yardımcı değişkenlerin RLS algoritmasıyla birlikte kullanılması durumunda elde edilen tekrarlamalı algoritma literatürde tekrarlamalı yardımcı değişkenler (Recursive Instrumental Variables) algoritması olarak bilinmektedir (Young 1984, Söderström ve Stoica 1981,1983,1989,2002, Ljung, 1999). Yardımcı değişkenlerin kullanıldığı tekrarlamalı yardımcı değişkenler algoritması hem RLS algoritmasının hızlı yakınsama özelliğine sahiptir, hem de ölçme gürültüsünün renkli gürültü olması durumunda yansız parametre tahminleri elde edilebilmektedir. Fakat tekrarlamalı yardımcı değişkenler algoritmasının işlem yükü yardımcı değişkenlerin seçimine bağlı olarak RLS algoritmasından biraz daha fazladır. Sistem parametrelerinin yansız olarak tahmin edilmesinde kullanılan diğer algoritmalar ise RLS tabanlı tekrarlamalı çıkış hatası (Recursive Output Error) algoritması (Landau 1976,1979,1990, Dugard ve Landau 1980, Landau ve ark. 1998, Landau ve Zito 2006), yanlılığı düzeltilmiş en küçük kareler (Bias Eliminated Least Squares) algoritması (Sagara ve Wada 1977, Feng ve Zheng 1991, Stoica ve ark. 1995, Zhang ve ark. 1997, Zheng 1998,2000), yaklaşık maksimum benzerlik (Approximate Maximum Likelihood) algoritması olarak da bilinen genişletilmiş en küçük kareler (Extended Least Squares) algoritması (Solo 1979, Lai ve Wei 1986, Guo ve Chen 1991, Söderström ve Stoica 1989, Ljung 1999), genelleştirilmiş en küçük kareler (Generalized Least Squares) algoritması (Söderström ve Stoica 1989, Ljung 1999), ile Newton yöntemleri üzerine kurulu tekrarlamalı tahmin hatası (Recursive Prediction Error) ve tekrarlamalı maksimum benzerlik (Recursive Maximum Likelihood) algoritmaları kullanılmaktadır (Aström 1980, Söderström ve Stoica 1989, Ljung 1999). Tekrarlamalı yardımcı değişkenler ve tekrarlamalı çıkış hatası algoritmalarında ölçme gürültüsü ile ilişkisiz değişkenler kullanılmaktadır ve işlem yükü diğer algoritmalara oranla biraz daha azdır. yanlılığı düzeltilmiş en küçük kareler algoritmasında yanlı parametrelerin elde edilmesine neden olan kısım yanlı tahminlerden çıkarılmakta ve yansız parametre tahminleri elde edilmektedir. Genişletilmiş en küçük kareler, genelleştirilmiş en küçük kareler, tekrarlamalı maksimum benzerlik ve tekrarlamalı tahmin hatası algoritmalarında ise yanlı parametre tahminlerinin bulunmasına neden olan renkli bozucu giriş beyaz gürültüden elde edilmiş gibi modellenmekte ve bu modelin parametreleri de sistem parametreleriyle birlikte eşzamanlı olarak tahmin edilmektedir.

Bu yöntemler sistem tanıma işleminde en iyi sonuç veren yöntemler olarak bilinmektedir. Fakat bu durumda hesaplanan parametre sayısı ve algoritmaların işlem yükü diğer algoritmalara göre daha fazla olmaktadır (Isermann ve ark. 1974, Saridis 1974, Sinha 1975, Söderström ve ark. 1978, Kurz ve ark. 1980, Strejc 1980, Sinha ve Kuszta 1983, Ljung ve Söderström 1983, Söderström ve Stoica 1989, Ljung 1999). Yukarıda bahsedilen algoritmaların hepsinde işlem yükü RLS algoritmasına göre daha fazladır. Bu algoritmaların çoğu çok değişkenli sistem parametrelerinin tahmin edilmesinde de kullanılmaktadır (Sinha ve Kwong 1979, Söderström ve Stoica 1989, Ljung 1999). Fakat bu durumda tahmin edilen parametre sayısı arttığı için işlem yükü önemli ölçüde artmaktadır. Çok değişkenli sistem parametrelerinin tahmin edilmesi konusunda yapılan bilimsel çalışmalar hala devam etmektedir (Zheng 1999, Zhu 2001, Ding ve Chen 2005a,2005b, Ding ve diğ. 2006,2007). Çok değişkenli sistem tanıma işlemi çok girişli çok çıkışlı sistemlerin uyarlamalı kontrolünde kullanılmaktadır (Goodwin ve ark. 1980, Johansson 1987, Isermann 1991, Zhu ve Backs 1993, Nasirsi-Toussi 1997, Zhu 2001, Yu 2005, Kubalcık ve Bobal 2006).

Son yıllarda, zaman ortalamalı normal denklemin GS algoritmasıyla çözümü üzerine kurulu olan tekrarlamalı bir algoritma önerilmiştir (Koçal 1997, 1998a, 1998b, Koçal ve Çalışkan 1998). Aynı algoritma farklı bir bakış açısıyla bir nümerik optimizasyon algoritması biçiminde EDS (Euclidean Direction Search) adı ile de önerilmiş ve bazı uyarlamalı işaret işleme uygulamalarında kullanılmıştır (Xu 1999, Bose ve ark. 1997, Xu ve ark. 1998, 1999, Mathurasai ve ark. 1999, Bose ve Xu 2002, Li ve ark. 2002, Rocha ve ark. 2002, Mabey ve ark. 2004, Bose 2004, Zhao ve Abeysekara 2005, Ahmad 2007, Zhang ve ark. 2004a, 2004b, 2005, 2006, 2008). Bu çalışmalarda GS algoritması uyarlamalı FIR filtre katsayılarının ayarlanmasında kullanılmıştır. Önerilen bu algoritmaların işlem yükü RLS algoritmasından daha az olup, yakınsama hızı RLS algoritmasına yakındır ve korelasyon matrisinin özdeğer yayılımının küçük olması durumunda RLS algoritmasına çok yakın sonuçlar vermektedir (Koçal 1997, 1998a, 1998b, Bose 2004). Yapılan bu çalışmalarda kullanılan GS algoritmasında iterasyon indisi ayrık zaman indisi ile aynıdır. Yani her örnekleme aralığında her parametre bir defa güncellenmektedir.

GS algoritması matris tersi formülünü kullanmak yerine normal denklemin nümerik olarak çözmek için de kullanılmaktadır (Cilke ve Eter 1992, Chen ve ark. 1997,1999, Xu ve Bose 1998, Ng ve ark. 2003). Fakat burada GS iterasyonu tekrarlamalı bir algoritma biçiminde değil, bir nümerik çözüm yöntemi olarak kullanılmaktadır. Yani normal denklem her örnekleme aralığında çoklu iterasyon yaparak çözülmektedir. GS iterasyonu iterasyon hatası belirli bir seviyenin altına inene kadar tekrarlanmaktadır. Bu durum işlem yükü açısından avantajlı değildir. Bu şekilde kullanılan GS iterasyonunda iterasyon indisi ayrık zaman indisiyle aynı değildir. Fakat Koçal (1997) ve Xu (1999) tarafından yapılan doktora tezlerinde ve sonrasında yapılan çalışmalarda GS algoritmasının iterasyon indisi ayrık zaman indisine eşit alınmıştır ve GS iterasyonları gerçek zamanda kullanılabilen tekrarlamalı bir algoritma biçiminde kullanılmıştır. Bu şekilde gerçeklenen GS algoritmasıyla M tane parametrenin güncellenmesi durumunda en genel halde $3M^2 + 2M$ tane çarpma ve bölme işlemi yapılmaktadır. RLS algoritmasında ise $3M^2 + 11M + 8$ tane çarpma ve bölme işlemi yapılmaktadır (Haykin 1991). Algoritmalarda unutma faktörü kullanılmadığında GS algoritmasında $2M^2 + M$ tane çarpma ve bölme işlemi yapılmaktadır. RLS algoritmasında ise en az işlemle $2M^2 + 4M$ tane çarpma ve bölme işlemi yapılmaktadır (Treichler ve ark. 1987). İki algoritma arasındaki çarpma ve bölme işlemi sayısındaki fark düşünüldüğünde tekrarlamalı GS algoritmasının RLS algoritmasına göre işlem yükü açısından avantajlı olduğu görülmektedir (Koçal 1997,1998a,1998b, Koçal ve Çalışkan 1998, Bose 2004). Tekrarlamalı GS algoritmasının yakınsama hızının RLS algoritmasına göre daha düşük olduğu bilinmektedir. Fakat korelasyon matrisinin özdeğer yayılımının düşük olduğu durumlarda RLS algoritmasına çok yakın sonuçlar vermektedir. (Koçal 1997,1998a,1998b, Koçal ve Çalışkan 1998, Bose 2004).

Bu tezin amacı, daha önce uyarlamalı işaret işleme uygulamalarında FIR filtre katsayılarını güncellemek için kullanılan GS algoritmasını transfer fonksiyonu parametrelerini ve uyarlamalı denetleyici güncellemek için kullanmaktır. Bu amaçla tekrarlamalı GS algoritmasının RLS algoritmasına göre daha düşük olan işlem yükü avantajının yanında yüksek yakınsama hızı özelliği de göz önüne alınarak yapılan çalışmalarda tekrarlamalı GS algoritmasının oldukça başarılı sonuçlar verdiği görülmüştür. (Hatun ve Koçal 2007a,2007b,2008).

3. MATERYAL VE YÖNTEM

Bu tez çalışmasında çevrim-içi sistem tanıma işlemi için doğrusal denklem takımı çözüm yöntemleri üzerine kurulu olan tekrarlamalı GS algoritması önerilmiştir ve aşağıda verilen alanlarda kullanılmıştır.

- Ayrık-zaman sistemlerin transfer fonksiyonu parametrelerinin tahmin edilmesi
- Ayrık-zaman çok-girişli çok-çıkışlı sistemlerin transfer fonksiyonu matrisi parametrelerinin tahmin edilmesi
- Sürekli-zaman sistemlerin transfer fonksiyonu parametrelerinin tahmin edilmesi
- Sürekli-zaman çok-girişli çok-çıkışlı sistemlerin transfer fonksiyonu matrisi parametrelerinin tahmin edilmesi
- Sistem tanıma ve uyarlamalı kontrol sistemlerinde kullanılan bazı zaman serileri modellerinin parametrelerinin tahmin edilmesi
- Doğrusal olmayan Volterra model parametrelerinin tahmin edilmesi
- Dolaylı uyarlamalı kontrol
- Doğrudan uyarlamalı kontrol

Bu tez çalışmasında ağırlıklı olarak GS algoritması üzerinde durulacaktır. Bu algoritmanın benzeri olan Jacobi algoritması, yakınsama özellikleri ve kullanımı, GS algoritmasının adım parametresi kullanılarak hızlandırılması ve GS algoritmasının zamanla değişen sistemlerde kullanılması gibi özel durumlara bazı alt bölümlerde yer verilecektir. Yukarıda ana başlıklar halinde verilen konular bundan sonraki alt bölümlerde detaylı şekilde anlatılacak ve önerilen tekrarlamalı GS algoritmasının konuya özel olarak kullanımı ve yakınsama analizi gibi kavramlar bu alt bölümlerde verilecektir.

3.1. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Ayrık-Zaman Sistemlerin Transfer Fonksiyonu Parametrelerinin Yansız Tahmini

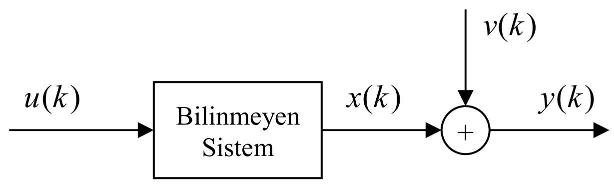
Bu kısımda, doğrusal zamanla değişmeyen ayrık-zamanlı deterministik sistemlerin transfer fonksiyonu parametrelerini yansız olarak tahmin etmek için, GS iterasyonları ile yardımcı değişkenlerin birlikte kullanıldığı tekrarlamalı bir algoritma önerilmiştir (Hatun ve Koçal 2007a). Yardımcı değişkenlerin GS algoritmasıyla birlikte kullanıldığı tekrarlamalı algoritma GSYD (Gauss-Seidel Yardımcı Değişkenler) algoritması olarak adlandırılmıştır. Ayrıca, önerilen algoritmanın stokastik yakınsama analizi yapılmış ve yardımcı değişkenlerin kullanılması durumunda sistemin çıkış işaretine karışan ölçme gürültüsünün renkli gürültü olması durumunda bile normal denklemlerin optimum çözümünü veren yansız bir kestireç olduğu analitik olarak gösterilmiştir.

3.1.1. Tekrarlamalı Gauss-Seidel yardımcı değişkenler algoritması ile ayrık-zaman sistem tanıma

Doğrusal zamanla değişmeyen tek girişli tek çıkışlı $n.$ dereceden ayrık zamanlı bir sistemin transfer fonksiyonu z -domeninde aşağıdaki gibi yazılabilir.

$$\frac{B(z)}{A(z)} = \frac{b_0 z^m + b_1 z^{m-1} + \cdots + b_m}{z^n + a_1 z^{n-1} + \cdots + a_n} \quad , \quad m \leq n \tag{3.1}$$

Açık çevrimde sistem belirleme işleminde, kullanılan sisteme ait giriş ve çıkış işaretleri Şekil 3.1'deki gibi gösterilebilir.



Şekil 3.1. Sistem belirleme işleminde kullanılan sistem modeli.

Şekil 3.1'de $u(k)$ sisteme uygulanan giriş işaretini, $x(k)$ ve $y(k)$ sırasıyla sistemin gürültüsüz ve gürültülü çıkış işaretini, $v(k)$ ise çıkış işaretine karışan ölçme gürültüsünü göstermektedir.

GS algoritması, sistem belirleme işleminde $R\theta_{opt} = p$ ile verilen normal denklemin çözümünde kullanılmaktadır, burada θ_{opt} ile $m+n$ adet sistem parametresini içeren $(m+n)\times 1$ boyutlu parametre vektörü, R ile $(m+n)\times(m+n)$ boyutlu korelasyon matrisi ve p ile $(m+n)\times 1$ boyutlu korelasyon vektörü gösterilmektedir. R matrisi ve p vektörü aşağıdaki gibi tanımlanmaktadır.

$$R = E[\varphi(k)\varphi^T(k)]\ ,\ \ p = E[\varphi(k)y(k)] \tag{3.2}$$

Burada E beklenen değer operatörüdür, $\varphi(k)$ ise sistemin giriş-çıkış işaretlerinden oluşan veri vektörüdür ve aşağıdaki gibi yazılabilir.

$$\begin{aligned} \varphi(k) &= \begin{bmatrix} -y(k-1) & \cdots & -y(k-n) & u(k) & \cdots & u(k-m) \end{bmatrix}^T \\ \theta_{opt} &= \begin{bmatrix} a_1 & \cdots & a_n & b_0 & \cdots & b_m \end{bmatrix}^T \end{aligned} \tag{3.3}$$

GS algoritmasıyla tekrarlamalı parametre tahmin işleminin başlangıç noktası, zaman ortalamalı normal denklemin GS algoritmasıyla çözümü üzerine kuruludur. Çünkü pratik uygulamalarda R matrisinin ve p vektörünün değerinin bilinmemesi durumunda tahmin edilmiş değerleri kullanılarak bir yaklaşıklık yapılır. Bu yaklaşık tahmini değerler k adet veri grubu kullanıldığında aşağıdaki gibi yazılabilir,

$$R(k) \cong \frac{1}{k}\sum_{n=1}^{k}\varphi(n)\varphi^T(n)\ ,\ \ p(k) \cong \frac{1}{k}\sum_{n=1}^{k}\varphi(n)y(n) \tag{3.4}$$

veya $1/k$ çarpanı göz önüne alınmadan giriş-çıkış veri örnekleri alındıkça iteratif olarak aşağıdaki gibi güncellenebilir.

$$R(k)=R(k-1)+\varphi(k)\varphi^{T}(k)\ ,\ \ p(k)=p(k-1)+\varphi(k)y(k) \tag{3.5}$$

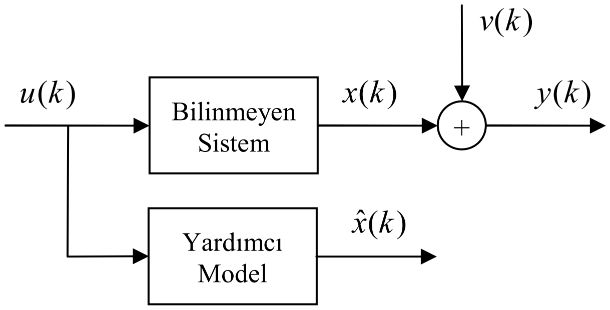
$R(k)$ matrisi ve $p(k)$ vektörü iteratif olarak güncellendikten sonra bir adımlık GS iterasyonu $R(k)\hat{\theta}(k)=p(k)$ olarak yazılan zaman ortalamalı normal denklemin çözümünde

$$\hat{\theta}_i(k+1)=\left[p_i(k)-\sum_{j=1}^{i-1}R_{ij}(k)\hat{\theta}_j(k+1)-\sum_{j=i+1}^{m+n}R_{ij}(k)\hat{\theta}_j(k)\right]\Big/R_{ii}(k) \tag{3.6}$$

$(\,i=1,2,\ldots,m+n\,)$

şeklinde kullanılır, yani iterasyon indisi ayrık zaman indisi olarak alınmıştır (Hatun ve Koçal 2007a). Burada $R_{ij}(k)$ korelasyon matrisinin i. satırına ve j. sütununa denk düşen elemanını gösterir, $p_i(k)$ korelasyon vektörünün i. elemanını gösterir ve $\hat{\theta}_i(k)$ parametre vektörünün i. elemanını gösterir. Yukarıda (3.5) ve (3.6) ile verilen tekrarlamalı algoritma tekrarlamalı GS algoritması olarak adlandırılmıştır. Tekrarlamalı GS algoritmasının RLS algoritmasına göre en önemli farkı korelasyon matrisinin tersi yerine doğrudan kendisinin güncellenmesi, yani matris tersini güncellemek için kullanılan ilave bir iterasyona gerek kalmaması ve parametre tahminlerini güncellemek için anlık hata bilgisini doğrudan kullanmamasıdır. Ayrıca parametrelerin skaler olarak güncellendiğini de göz önüne aldığımızda, RLS algoritmasına göre işlem yükünün önemli ölçüde azaldığı görülmektedir (Koçal 1998, Bose 2004).

Tekrarlamalı GS algoritmasında yardımcı değişkenlerin kullanılmasıyla elde edilen tekrarlamalı algoritma tekrarlamalı GSYD algoritması olarak adlandırılmıştır. Yardımcı değişkenlerin en yaygın kullanım biçimi aşağıdaki Şekil 3.2'de görüldüğü gibidir. Burada kullanılan yardımcı model sabit katsayılı kararlı bir filtre veya parametreleri sistem parametreleriyle eşzamanlı olarak güncellenen bir adaptif filtre olabilmektedir. Bunların haricinde geciktirilmiş giriş veya çıkış işaretleri de yardımcı değişken olarak kullanılabilmektedir (Söderström ve Stoica 1981, 1983, 1989, 2002, Ljung 1999).



Şekil 3.2. Sistem belirleme işleminde yardımcı değişkenlerin elde edilmesi.

Burada yardımcı değişkenleri elde etmek için sistemle aynı yapıya sahip bir uyarlamalı filtre kullanıldığında, kullanılan yardımcı modele ait veri vektörü

$$z(k) = \left[-\hat{x}(k-1) \quad \cdots \quad -\hat{x}(k-n) \quad u(k) \quad \cdots \quad u(k-m)\right]^T \tag{3.7}$$

olarak yazılabilir ve bu vektörün elemanları $\hat{x}(k) = z^T(k)\hat{\theta}(k)$ şeklinde önceki adımda bulunan parametre değerleri kullanılarak hesaplanabilir. Tekrarlamalı GSYD algoritmasında hesaplanan bu değişkenler kullanıldığında $R(k)$ korelasyon matrisi ve $p(k)$ korelasyon vektörü

$$R(k) \cong \frac{1}{k}\sum_{n=1}^{k} z(n)\varphi^T(n) \ , \quad p(k) \cong \frac{1}{k}\sum_{n=1}^{k} z(n)y(n) \tag{3.8}$$

şeklinde birikimli olarak veya $1/k$ çarpanı göz önüne alınmadan giriş-çıkış veri örnekleri alındıkça iteratif olarak aşağıdaki gibi güncellenebilir.

$$R(k) = R(k-1) + z(k)\varphi^T(k) \ , \quad p(k) = p(k-1) + z(k)y(k) \tag{3.9}$$

$R(k)$ matrisi ve $p(k)$ vektörü iteratif olarak güncellendikten sonra parametre tahminleri bir adımlık GS iterasyonu kullanılarak (3.6) eşitliğindeki gibi güncellenebilir.

3.1.2. Tekrarlamalı Gauss-Seidel yardımcı değişkenler algoritmasının stokastik yakınsama analizi

Parametre tahminlerini tekrarlamalı veya çevrim-içi (on-line) olarak güncellemek için kullanılan tekrarlamalı GS algoritmasını vektörel formda aşağıdaki gibi yazabiliriz.

$$\hat{\theta}(k+1) = -(R_L(k) + R_D(k))^{-1} R_U(k)\hat{\theta}(k) + (R_L(k) + R_D(k))^{-1} p(k) \qquad (3.10)$$

Burada $R_L(k)$, $R_D(k)$ ve $R_U(k)$ sırasıyla korelasyon matrisinin alt üçgen, köşegen ve üst üçgen kısımlarını içeren kare matrisleri göstermektedir. Yakınsama analizinde, $R(k)$ korelasyon matrisinin ve $p(k)$ korelasyon vektörünün (3.4) ile verilen birikimli tahmini değerlerinin kullanıldığını varsayalım. Sistemin çıkış işaretini doğrusal bağlaşımlı biçimde (linear regression form) $y(k) = \varphi^T(k)\theta_{opt} + v(k)$ olarak yazabiliriz. Çıkış işaretini $p(k)$ korelasyon vektörünün tahmini değerinde yerine yazdığımızda

$$\begin{aligned} p(k) &= \frac{1}{k}\sum_{n=1}^{k}\varphi(n)y(n) = \frac{1}{k}\sum_{n=1}^{k}\varphi(n)\left[\varphi^T(k)\theta_{opt} + v(k)\right] \\ &= R(k)\theta_{opt} + \frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n) \end{aligned} \qquad (3.11)$$

elde edilir. Bu değer (3.9) ile verilen vektörel algoritmada yerine yazıldığında

$$\begin{aligned} \hat{\theta}(k+1) &= -(R_L(k) + R_D(k))^{-1} R_U(k)\hat{\theta}(k) \\ &\quad + (R_L(k) + R_D(k))^{-1}\left[R(k)\theta_{opt} + \frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right] \end{aligned} \qquad (3.12)$$

elde edilir. Bu denklemde korelasyon matrisi $R(k) = R_L(k) + R_D(k) + R_U(k)$ şeklinde yerine yazılarak denklem düzenlendiğinde

$$\hat{\theta}(k+1) = -(R_L(k)+R_D(k))^{-1} R_U(k)\hat{\theta}(k) + \left[I + (R_L(k)+R_D(k))^{-1} R_U(k)\right]\theta_{opt}$$
$$+ (R_L(k)+R_D(k))^{-1}\left[\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right] \quad (3.13)$$

eşitliği elde edilir. Burada

$$\begin{aligned} & E[\hat{\theta}(k+1)] = E[\hat{\theta}(k)] \ , \ \ E[\varphi(k)v(k)] = \frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n) \\ & E[(R_L(k)+R_D(k))^{-1}] = (R_L + R_D)^{-1} \\ & E[I + (R_L(k)+R_D(k))^{-1} R_U(k)] = I + (R_L + R_D)^{-1} R_U \end{aligned} \quad (3.14)$$

kabulleri kullanılarak (3.13) ile verilen denklemin her iki tarafının beklenen değeri alındığında

$$[I + (R_L + R_D)^{-1} R_U] E[\hat{\theta}(k)] = [I + (R_L + R_D)^{-1} R_U]\ \theta_{opt}$$
$$+ (R_L + R_D)^{-1} E[\varphi(k)v(k)] \quad (3.15)$$

eşitliği elde edilir. Bu eşitlikten parametre tahminlerinin beklenen değeri

$$E[\hat{\theta}(k)] = \theta_{opt} + [I + (R_L + R_D)^{-1} R_U]^{-1} (R_L + R_D)^{-1} E[\varphi(k)v(k)] \quad (3.16)$$

olarak elde edilir. Burada $I + (R_L + R_D)^{-1} R_U = (R_L + R_D)^{-1} R$ matris eşitliği kullanıldığında

$$E[\hat{\theta}(k)] = \theta_{opt} + (R_L + R_D) R^{-1} (R_L + R_D)^{-1} E[\varphi(k)v(k)] \quad (3.17)$$

sonucu elde edilir (Hatun ve Koçal 2007a). Elde edilen bu sonuç tekrarlamalı GS algoritması ile elde edilen parametre tahminlerinin optimum değerine yakınsaması için gerekli şartları göstermektedir. Bunun için öncelikle korelasyon matrisinin tersi alınabilmelidir. Bunun için korelasyon matrisi pozitif tanımlı olmalıdır. Sistemin sürekli olarak uyarılması durumunda bu şart sağlanmaktadır. Korelasyon matrisinin pozitif tanımlı olması durumunda tekrarlamalı GS iterasyonlarının yakınsaması sağlanmış olmaktadır (Hatun ve Koçal 2005). Pratik uygulamalarda başlangıçta korelasyon matrisinin pozitif tanımlılığını garantilemek amacıyla $R(0)=\delta.I$ olarak alınabilir, burada I uygun boyutlu birim matris olup $\delta = 0.1,\ 0.01,\ 0.001$ gibi küçük pozitif değerler alabilir. Algoritmanın kararlılığı sağlandıktan sonra elde edilen parametre tahminlerinin optimum değerine yakınsaması için gerekli olan diğer şart ise giriş-çıkış örnekleri alındıkça $E[\varphi(k)v(k)]$ vektörünün sıfır vektöre yakınsamasıdır. Bu vektörü çıkış işaretini $y(k)=x(k)+v(k)$ şeklinde yazarak aşağıdaki gibi inceleyebiliriz.

$$E[\varphi(k)v(k)] = E\left[\begin{bmatrix} -y(k-1) \\ \vdots \\ -y(k-n) \\ u(k) \\ \vdots \\ u(k-m) \end{bmatrix} v(k)\right] = E\left[\begin{bmatrix} -x(k-1) \\ \vdots \\ -x(k-n) \\ u(k) \\ \vdots \\ u(k-m) \end{bmatrix} v(k)\right] + E\left[\begin{bmatrix} -v(k-1) \\ \vdots \\ -v(k-n) \\ u(k) \\ \vdots \\ u(k-m) \end{bmatrix} v(k)\right] \tag{3.18}$$

Burada $E[x(k)v(k)]=0$ ve $E[u(k)v(k)]=0$ olduğu göz önüne alındığında

$$E[\varphi(k)v(k)] = E\left[\begin{bmatrix} -v(k-1) \\ \vdots \\ -v(k-n) \\ u(k) \\ \vdots \\ u(k-m) \end{bmatrix} v(k)\right] = \begin{bmatrix} -r_v(1) \\ \vdots \\ -r_v(n) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.19}$$

sonucu elde edilir. Bu durumda $E[\varphi(k)v(k)]$ vektörünün sıfır vektöre yakınsayabilmesi için $r_v(i)=E[v(k-i)v(k)]$, $(i=1,2,\ldots,n)$ olarak tanımlanan, bozucu girişe ait ilk n adet oto-korelasyon katsayısının sıfır olması gerekir. Bu sonuca göre tekrarlamalı GS algoritmasıyla elde edilen parametre tahminlerinin optimum değerine yakınsayabilmesi için ölçme gürültüsünün beyaz gürültü olması gerekir. Tekrarlamalı GSYD algoritmasının yakınsama analizi de benzer şekilde yapıldığında (3.17) ile verilen sonuç

$$E[\hat{\theta}(k)] = \theta_{opt} + (R_L + R_D)R^{-1}(R_L + R_D)^{-1}E[z(k)v(k)] \tag{3.20}$$

olarak elde edilir. Bu sonuca göre yine algoritmanın kararlılığının sağlanabilmesi için (3.8) ve (3.9) ile verilen korelasyon matrisinin tersi alınabilir olmalıdır. Burada yardımcı değişkenleri elde etmek için sistemle aynı yapıya sahip bir uyarlamalı filtre kullanıldığında bu şart çok rahat bir şekilde sağlanabilmektedir (Söderström ve Stoica 1981, 1983, 1989, 2002, Ljung 1999). Ayrıca, tekrarlamalı GSYD algoritmasında gürültü ile ilişkisiz değişkenler kullanıldığı için giriş-çıkış örnekleri alındıkça zamanla $E[\hat{x}(k)v(k)] = 0$ olmakta ve $E[z(k)v(k)]$ vektörü

$$E[z(k)v(k)] = E\left[\begin{bmatrix} -\hat{x}(k-1) \\ \vdots \\ -\hat{x}(k-n) \\ u(k) \\ \vdots \\ u(k-m) \end{bmatrix} v(k)\right] = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.21}$$

şeklinde sıfır vektöre yakınsamaktadır (Hatun ve Koçal 2007a). Bu sonuca göre tekrarlamalı GSYD algoritması kullanıldığında (3.20) eşitliği gereği algoritmanın kararlılığı sağlandığı sürece, çıkış işaretine karışan ölçme gürültüsünün renkli gürültü olması durumunda bile doğru parametre tahminlerinin elde edildiği görülmektedir.

3.2. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Çok-Girişli Çok-Çıkışlı Ayrık-Zaman Sistemlerin Transfer Fonksiyonu Matrisi Parametrelerinin Yansız Tahmini

Bu kısımda, ayrık-zamanlı sistemlerin transfer fonksiyonu parametrelerini yansız tahmin etmek için önerilen tekrarlamalı GSYD algoritması, çok-girişli çok-çıkışlı ayrık-zamanlı sistemlerin parametrelerinin yansız tahmin edilmesinde kullanılmıştır. Önerilen algoritmanın stokastik yakınsama analizi yapılmış ve normal denklemlerin optimum çözümünü veren yansız bir kestireç olduğu gösterilmiştir.

Doğrusal zamanla değişmeyen q-girişli r-çıkışlı ayrık zamanlı bir sistemin transfer fonksiyonu matrisi z-domeninde aşağıdaki gibi yazılabilir (Sinha ve Kuszta 1983, Söderström ve Stoica 1989, Ljung 1999).

$$\begin{bmatrix} X_1(z) \\ X_2(z) \\ \vdots \\ X_r(z) \end{bmatrix} = \begin{bmatrix} \frac{B_{11}(z)}{A_1(z)} & \frac{B_{12}(z)}{A_1(z)} & \cdots & \frac{B_{1q}(z)}{A_1(z)} \\ \frac{B_{21}(z)}{A_2(z)} & \frac{B_{22}(z)}{A_2(z)} & \cdots & \frac{B_{2q}(z)}{A_2(z)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{B_{r1}(z)}{A_r(z)} & \frac{B_{r2}(z)}{A_r(z)} & \cdots & \frac{B_{rq}(z)}{A_r(z)} \end{bmatrix} \begin{bmatrix} U_1(z) \\ U_2(z) \\ \vdots \\ U_q(z) \end{bmatrix} \tag{3.22}$$

Bu denklemde ($i = 1,2,\ldots,q$) için $U_i(z) = Z[u_i(k)]$ şeklinde $i.$ giriş işaretinin z-dönüşümü, ($j = 1,2,\ldots,r$) için $X_j(z) = Z[x_j(k)]$ şeklinde $j.$ çıkış işaretinin z-dönüşümü gösterilmektedir. Sistemin her bir çıkış işareti birbirinden bağımsız olarak çok-girişli tek-çıkışlı sistem biçiminde aşağıdaki gibi yazılabilir (Zheng 1999, Ding ve ark. 2006).

$$X_j(z) = \frac{B_{j1}(z)}{A_j(z)}U_1(z) + \frac{B_{j2}(z)}{A_j(z)}U_2(z) + \cdots + \frac{B_{jq}(z)}{A_j(z)}U_q(z) \quad , \quad (j = 1,2,\ldots,r) \tag{3.23}$$

Bu denklemler z^{-1} birim gecikme operatörüne bağlı olarak fark denklemi formunda aşağıdaki gibi yazılabilir (Zheng 1999, Ding ve ark. 2006).

$$A_j(z^{-1})x_j(k) = B_{j1}(z^{-1})u_1(k) + B_{j2}(z^{-1})u_2(k) + \cdots + B_{jq}(z^{-1})u_q(k) \tag{3.24}$$
$$(j = 1,2,\ldots,r)$$

Bu denklemlerde ($j = 1,2,\ldots,r$) için $j.$ çıkışa ait payda polinomu ve $j.$ çıkışa ait pay polinomları sırasıyla

$$\begin{aligned}
A_j(z^{-1}) &= 1 + a_{j1}z^{-1} + \cdots + a_{jn_j}z^{-n_j} \\
B_{j1}(z^{-1}) &= b_{j10} + b_{j11}z^{-1} + \cdots + b_{j1m_{j1}}z^{-m_{j1}} \\
B_{j2}(z^{-1}) &= b_{j20} + b_{j21}z^{-1} + \cdots + b_{j2m_{j2}}z^{-m_{j2}} \\
&\vdots \\
B_{jq}(z^{-1}) &= b_{jq0} + b_{jq1}z^{-1} + \cdots + b_{jqm_{jq}}z^{-m_{jq}}
\end{aligned} \tag{3.25}$$

şeklinde yazılabilir, burada $n_j, m_{j1}, m_{j2}, \ \ldots \ , m_{jq}$ ile sırasıyla $A_j(z^{-1}), B_{j1}(z^{-1}), B_{j2}(z^{-1}), \ \ldots\ , \ B_{jq}(z^{-1})$ polinomlarının derecesi gösterilmektedir ve pay polinomlarının dereceleri $(m_{j1}, m_{j2}, \ \ldots, m_{jq}) \leq n_j$ şeklinde payda polinomunun derecesinden küçük veya eşit olduğu varsayılmıştır. Açık çevrimde sistem belirleme işleminde, sistemin çıkış işareti ölçümlerine ($j = 1,2,\ldots,r$) için $y_j(k) = x_j(k) + v_j(k)$ şeklinde renkli ölçme gürültüsü karıştığını varsaydığımızda (3.24) ile verilen fark denklemleri ($j = 1,2,\ldots,r$) için

$$\begin{aligned}
y_j(k) = &-a_{j1}y_j(k-1) - \cdots - a_{jn_j}y_j(k-n_j) \\
&+ b_{j10}u_1(k) + b_{j11}u_1(k-1) + \cdots + b_{j1m_{j1}}u_1(k-m_{j1}) \\
&+ b_{j20}u_2(k) + b_{j21}u_2(k-1) + \cdots + b_{j2m_{j2}}u_2(k-m_{j2}) \\
&+ \cdots + \\
&+ b_{jq0}u_q(k) + b_{jq1}u_q(k-1) + \cdots + b_{jqm_{jq}}u_q(k-m_{jq}) + v_j(k)
\end{aligned} \tag{3.26}$$

şeklinde giriş ve çıkış örneklerine bağlı olarak yazılabilir. Bu denklemleri sistem tanıma işleminde kullanabilmek için doğrusal bağlanımlı biçimde (linear regression form)

$$y_j(k) = \varphi_j^T(k)\theta_j + v_j(k) \quad , \quad (j = 1,2,\ldots,r) \tag{3.27}$$

şeklinde yazıldığında $j.$ çıkışa ait veri vektörü ve parametre vektörü sırasıyla

$$\begin{aligned}
\varphi_j^T(k) = [&-y_j(k-1) \ \cdots -y_j(k-n_j) \ u_1(k) \ u_1(k-1) \ \cdots \ u_1(k-m_{j1}) \ u_2(k) \ u_2(k-1) \ \cdots \\
&\cdots u_2(k-m_{j2}) \ \cdots \ u_q(k) \ u_q(k-1) \ \cdots \ u_q(k-m_{jq})]
\end{aligned} \tag{3.28}$$

$$\theta_j^T = [\,a_{j1} \;\; a_{j2} \;\cdots\; a_{jn_j} \;\; b_{j10} \;\; b_{j11} \;\cdots\; b_{j1m_{j1}} \;\; b_{j20} \;\; b_{j21} \;\cdots\; b_{j2m_{j2}} \;\cdots\; b_{jq0} \;\; b_{jq1} \;\cdots\; b_{jqm_{jq}}\,] \tag{3.29}$$

olarak yazılabilir. Bu iki vektör parametre sayısı $M_j = n_j + (m_{j1}+1) + \cdots + (m_{jq}+1)$ olmak üzere $(M_j \times 1)$ boyutlu sütun vektördür. Parametre tahmininde her bir çıkışa ait tahmin hatası

$$\varepsilon_j(k) = y_j(k) - \varphi_j^T(k)\hat{\theta}_j(k-1) \quad , \quad (\,j = 1,2,\ldots,r\,) \tag{3.30}$$

olarak tanımlandığında, minimum yapılmak istenen hata fonksiyonu

$$V\big(\theta_j(k)\big) = \frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\boldsymbol{\varepsilon}^T(i)\boldsymbol{\varepsilon}(i) \tag{3.31}$$

olarak yazılabilir, burada $0 < \lambda \le 1$ unutma faktörüdür ve hata vektörleri ($i = 1,2,\ldots,k$) için

$$\boldsymbol{\varepsilon}(i) = \begin{bmatrix} \varepsilon_1(i) \\ \varepsilon_2(i) \\ \vdots \\ \varepsilon_r(i) \end{bmatrix} = \begin{bmatrix} y_1(i) - \varphi_1^T(i)\hat{\theta}_1(i-1) \\ y_2(i) - \varphi_2^T(i)\hat{\theta}_2(i-1) \\ \vdots \\ y_r(i) - \varphi_r^T(i)\hat{\theta}_r(i-1) \end{bmatrix} \tag{3.32}$$

olarak yazılabilir. Bu hata fonksiyonunu minimum yapan *j*. çıkışa ait optimum parametre vektörü, θ_j ’ye göre türev alındığında

$$\hat{\theta}_{jopt}(k) = R_j^{-1}(k)p_j(k) \tag{3.33}$$

olarak hesaplanır. Burada *j*. çıkışa ait $(M_j \times M_j)$ boyutlu korelasyon matrisi $R_j(k)$ ve $(M_j \times 1)$ boyutlu korelasyon vektörü $p_j(k)$ aşağıdaki gibi yazılabilir,

$$R_j(k)=\frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\varphi_j(i)\varphi_j^T(i) \quad , \quad p_j(k)=\frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\varphi_j(i)y_j(i) \tag{3.34}$$

veya çevrim-içi (on-line) uygulamalarda $1/k$ çarpanı göz önüne alınmadan giriş-çıkış veri örnekleri alındıkça iteratif olarak aşağıdaki gibi güncellenebilir (Sinha ve Kuszta 1983, Söderström ve Stoica 1989, Ljung 1999).

$$R_j(k)=\lambda R_j(k)+\varphi_j(k)\varphi_j^T(k) \quad , \quad p_j(k)=\lambda p_j(k)+\varphi_j(k)y_j(k) \tag{3.35}$$

$R_j(k)$ matrisi ve $p_j(k)$ vektörü iteratif olarak güncellendikten sonra bir adımlık GS iterasyonu, çözümü (3.33) ile verilen zaman ortalamalı $R_j(k)\theta_{jopt}(k)=p_j(k)$ normal denkleminin çözümünde

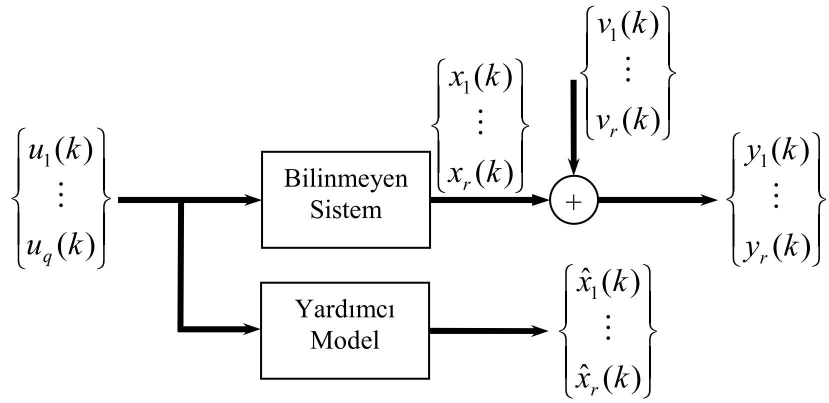
$$\hat{\theta}_{ji}(k)=\left[p_{ji}(k)-\sum_{l=1}^{i-1}R_{jil}(k)\hat{\theta}_{jl}(k)-\sum_{l=i+1}^{M_j}R_{jil}(k)\hat{\theta}_{jl}(k-1)\right]\Big/R_{jii}(k) \qquad (i=1,2,\ldots,M_j) \tag{3.36}$$

şeklinde kullanılır, yani iterasyon indisi ayrık zaman indisi olarak alınmıştır. Burada $R_{jil}(k)$ korelasyon matrisinin i. satırına ve l. sütununa denk düşen elemanını gösterir, $p_{ji}(k)$ korelasyon vektörünün i. elemanını gösterir ve $\hat{\theta}_{ji}(k)$ parametre vektörünün i. elemanını gösterir. Yukarıda (3.35) ve (3.36) ile verilen algoritma tekrarlamalı GS algoritması olarak adlandırılmıştır. Tekrarlamalı GS algoritmasının RLS algoritmasına göre en önemli farkı korelasyon matrisinin tersi yerine doğrudan kendisinin güncellenmesi, yani matris tersini güncellemek için kullanılan ilave bir iterasyona gerek kalmaması ve parametre tahminlerini güncellemek için anlık hata bilgisini doğrudan kullanmamasıdır. Ayrıca parametrelerin skaler olarak güncellendiğini de göz önüne aldığımızda, RLS algoritmasına göre işlem yükünün önemli ölçüde azaldığı görülmektedir (Koçal 1998, Bose 2004).

Tekrarlamalı GS algoritmasında yardımcı değişkenlerin kullanılmasıyla elde edilen tekrarlamalı algoritma Tekrarlamalı GSYD algoritması olarak adlandırılmıştır (Hatun ve Koçal 2007a). Yardımcı değişkenlerin en yaygın kullanım biçimi Şekil 3.3'te görüldüğü gibidir. Burada yardımcı değişken olarak sistemle aynı yapıya sahip olan ve parametreleri sistem parametreleriyle eşzamanlı olarak güncellenen çok-girişli çok-çıkışlı bir uyarlamalı modelin çıkışları kullanılabilir. Bunların haricinde geciktirilmiş giriş veya çıkış işaretleri de yardımcı değişken olarak kullanılabilmektedir (Söderström ve Stoica 1981, 1983, 1989, 2002, Ljung 1999). Yardımcı değişkenleri elde etmek için sistemle aynı yapıya sahip çok-girişli çok çıkışlı bir uyarlamalı model kullanıldığında, kullanılan yardımcı modele ait veri vektörü

$$\begin{aligned} z_j^T(k) = [&-\hat{x}_j(k-1) \ \cdots \ -\hat{x}_j(k-1) \ u_1(k) \ u_1(k-1) \ \cdots \ u_1(k-m_{j1}) \ u_2(k) \ u_2(k-1) \ \cdots \\ &\cdots u_2(k-m_{j2}) \ \cdots \ u_q(k) \ u_q(k-1) \ \cdots \ u_q(k-m_{jq})] \end{aligned} \tag{3.37}$$

olarak yazılabilir ve yardımcı değişkenler $\hat{x}_j(k) = z_j^T(k)\hat{\theta}_j(k-1)$ şeklinde önceki adımda bulunan parametre değerleri kullanılarak hesaplanabilir.



Şekil 3.3. Çok değişkenli sistem belirleme işleminde yardımcı değişkenlerin elde edilmesi.

Tekrarlamalı GSYD algoritmasında hesaplanan yardımcı değişkenler kullanıldığında j. çıkışa ait $R_j(k)$ korelasyon matrisi ve $p_j(k)$ korelasyon vektörü giriş-çıkış veri örnekleri alındıkça iteratif olarak aşağıdaki gibi güncellenebilir.

$$R_j(k) = \lambda R_j(k) + z_j(k)\varphi_j^T(k) \quad , \quad p_j(k) = \lambda p_j(k) + z_j(k)y_j(k) \tag{3.38}$$

$R_j(k)$ matrisi ve $p_j(k)$ vektörü iteratif olarak güncellendikten sonra parametre tahminleri bir adımlık GS iterasyonu kullanılarak (3.36) eşitliğindeki gibi güncellenebilir. Burada sırasıyla (3.37), (3.38) ve (3.36) eşitlikleriyle verilen tekrarlamalı GSYD algoritması çok-değişkenli sistem parametrelerini yansız olarak tahmin etmek için kullanılabilir.

Önerilen tekrarlamalı GSYD algoritmasının işlemsel yükünü hesaplamak için, öncelikle (3.24) ile verilen her çok-girişli tek-çıkışlı fark denkleminde eşit sayıda (M adet) parametre hesaplandığını varsayalım. Burada (3.38) eşitliğine bakıldığında, korelasyon matrisinin güncellenmesi için $2M^2$ adet, korelasyon vektörünün güncellenmesi için $2M$ adet çarpma işleminin olduğu görülebilir. Parametrelerin güncellendiği (3.36) iterasyonunun gerçeklenmesinde M^2 adet çarpma ve bölme işleminin yapıldığı görülebilir. Bu durumda tekrarlamalı GS algoritmasında yapılan çarpma ve bölme sayılarının toplamı $3M^2 + 2M$ olarak elde edilir. Tekrarlamalı GSYD algoritmasında yardımcı değişkenlerin hesaplanması için ilave olarak M adet çarpma işlemi daha yapıldığı için toplam çarpma ve bölme sayısı $3M^2 + 3M$ olarak hesaplanabilir. RLS algoritmasında ise çarpma ve bölme sayısının $3M^2 + 11M + 8$ olduğu bilinmektedir (Haykin 1991, Koçal 1998). Tekrarlamalı YD algoritmasında yardımcı değişkenlerin hesaplanması için ilave olarak M adet çarpma işlemi daha yapıldığı için toplam çarpma ve bölme sayısı $3M^2 + 12M + 8$ olacaktır. Bu durumda tekrarlamalı YD ve tekrarlamalı GSYD algoritmalarında parametre vektörünün güncellenmesi için gerekli çarpma ve bölme sayıları arasındaki fark ise $9M + 8$ olmaktadır. Bu fark bir parametre vektörü için hesaplanmıştır. Burada q-girişli r-çıkışlı sistemlerde çıkış sayısı kadar parametre vektörünün güncellendiği göz önüne alınırsa tekrarlamalı YD ve tekrarlamalı GSYD algoritmalarının arasındaki çarpma ve bölme sayısı farkı $r(9M + 8)$ olacaktır.

3.3. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Sürekli-Zaman Sistemlerin Transfer Fonksiyonu Parametrelerinin Yansız Tahmini

Bu kısımda, doğrusal sürekli-zamanlı sistemlerin diferansiyel denklem modeli parametrelerini tahmin etmek için, GS iterasyonlarının ve yardımcı değişkenlerin birlikte kullanıldığı tekrarlamalı bir algoritma önerilmiştir. Gürültülü işaretlerin türevinden kaçınmak için sistemin alçak geçiren durum değişkeni filtrelerini içeren bir modeli kullanılmıştır. Ayrıca, önerilen algoritmanın yakınsama analizi yapılmış ve yardımcı değişkenlerin kullanılması durumunda, ölçme gürültüsünün renkli gürültü olması durumunda bile normal denklemin optimum çözümünü veren bir yansız kestirici olduğu gösterilmiştir.

3.3.1. Tekrarlamalı Gauss-Seidel yardımcı değişkenler algoritması ile sürekli-zaman sistem tanıma

Sayısal bilgisayarların ve mikroişlemcilerin gelişmesiyle birlikte, ayrık sistem modellerini deneysel verileri kullanarak gerçek zamanda tahmin etmek için bilim adamları tarafından birçok algoritma önerilmiş ve kullanılmıştır (Ljung ve Söderström 1983, Sinha ve Kuszta 1983, Goodwin ve Sin 1984, Söderström ve Stoica 1989, Johansson 1993, Walter ve Pronzato 1998, Raol ve ark. 2004, Ljung 1999, 2008). Ancak, sürekli zaman modellerin fiziksel yapıyı daha doğru temsil etmesinden dolayı sürekli sistem parametrelerinin tahmin edilmesi işlemi önem kazanmış ve ayrık verileri kullanan sürekli sistem tahmin yöntemleri son yirmi yıldır artan bir ilgi çekmiştir (Young 1981, Gawthrop 1987, Unbehauen ve Rao 1987, 1990, 1998, Sinha ve Rao 1991, Garnier ve ark. 2003, Garnier ve Young 2004, Rao ve Unbehauen 2006, Garnier ve Wang 2008). Bu yöntemlerde sistemin diferansiyel denklem modelinin parametreleri tahmin edilmektedir. Fakat yüksek geçiren filtre özelliği göstererek gürültü bileşenlerini kuvvetlendiren türev işleminden kaçınmak amacıyla sistemin diferansiyel denklemi değiştirilerek kullanılır. Daha sonra elde edilen değiştirilmiş sistem modeli uygun şekilde ayrıklaştırılarak sayısal hale getirilmekte ve bilinen bir sistem tanıma algoritması kullanılarak sürekli zaman parametreleri tahmin edilebilmektedir. Bu çalışmada Johansson (1986b,1993,1994) tarafından önerilen alçak geçiren durum değişkeni filtrelerine bağlı parametre tahmin modeli kullanılmıştır. Burada öncelikle kullanılan doğrusal tahmin modeli tanıtılacaktır, sonra kullanılan sürekli modelin parametrelerini tahmin etmek için önerilen tekrarlamalı GS algoritması tanıtılacaktır. Daha sonra renkli ölçme gürültüsünün olumsuz etkisinden kurtulmak için, yardımcı değişkenlerin tekrarlamalı GS algoritmasıyla birlikte kullanımı anlatılacaktır.

3.3.2. Doğrusal tahmin modeli

Bir tek girişli tek çıkışlı doğrusal zamanla değişmeyen $n.$ derece sürekli-zamanlı sistemin diferansiyel denklem modeli

$$\frac{d^n x(t)}{dt^n} + a_1 \frac{d^{n-1} x(t)}{dt^{n-1}} + \cdots + a_n x(t) = b_1 \frac{d^{n-1} u(t)}{dt^{n-1}} + \cdots + b_n u(t) \tag{3.39}$$

olarak yazılabilir. Burada $u(t)$ ve $x(t)$ sistemin giriş ve çıkış işaretini göstermektedir. Parametre tahmin işlemini olumsuz etkileyen türev işleminden kaçınmak amacıyla, Johansson (1993, 1994) tarafından (3.39) ile verilen diferansiyel denklem modeli türev operatörü yerine nedensel, kararlı ve gerçeklenebilir bir doğrusal operatöre bağlı olacak şekilde tekrar düzenlenmiştir. Bunun için alçak geçiren filtre operatörleri içeren bir doğrusal model ele alınmış ve sürekli model parametreleri ile seçilen filtre çıkışları arasında ilişki kuran doğrusal bir dönüşüm yapılmıştır. Burada kullanılan alçak geçiren filtre operatörü

$$\lambda = \frac{a}{s+a} = \frac{1}{1+s\tau} \quad , \quad \tau = 1/a \tag{3.40}$$

olarak tanımlandığında, bu tanımlama bize

$$\lambda = \frac{1}{1+s\tau} \quad \leftrightarrow \quad s = \frac{1-\lambda}{\lambda\tau} \tag{3.41}$$

dönüşümünü yapma imkanı verir, burada $s = d/dt$ türev operatörüdür. Bu operatör dönüşümü (3.39) ile verilen diferansiyel denkleme uygulandığında

$$x(t)+\alpha_1\lambda x(t)+\cdots+\alpha_n\lambda^n x(t)=\beta_1\lambda u(t)+\cdots+\beta_n\lambda^n u(t) \tag{3.42}$$

denklemi elde edilir. Burada $\lambda^i u(t)$ ve $\lambda^i x(t)$ sırasıyla i defa filtrelenmiş giriş ve çıkış işaretlerini göstermektedir. Kullanılan giriş-çıkış verileri ayrık olduğundan filtreleme işlemi de ayrık olarak yapılmalıdır. Durum değişkeni filtreleme işlemini gerçeklemenin en hızlı ve kolay yolu filtrenin IIR formdaki ayrık-zaman modelini kullanmaktır (Sinha ve Rao 1991). Bu amaçla alçak geçiren filtrenin tustin eşdeğeri kullanıldığında tek katlı filtrenin fark denklemi

$$\lambda x(k) = p\lambda x(k-1) + q\big(x(k) + x(k-1)\big) \tag{3.43}$$

olarak elde edilir. Burada k ayrık zaman indisi olup,

$$p = \frac{2-aT}{2+aT} \quad , \quad q = \frac{aT}{2+aT} \tag{3.44}$$

filtrenin fark denkleminin katsayıları, T ise örnekleme periyodudur. Daha yüksek dereceli filtrelenmiş işaretler aynı fark denklemi kullanılarak $i = 0,1,\dots,n$ için

$$\lambda^i x(k) = p\lambda^i x(k-1) + q\big(\lambda^{i-1}x(k) + \lambda^{i-1}x(k-1)\big) \tag{3.45}$$

şeklinde ardışık olarak hesaplanabilir. Aynı filtreleme işlemi giriş işaretine de benzer şekilde uygulandığında (3.42) denklemi filtrelenmiş ayrık işaretlere bağlı olarak

$$x(k)+\alpha_1\lambda x(k)+\cdots+\alpha_n\lambda^n x(k)=\beta_1\lambda u(k)+\cdots+\beta_n\lambda^n u(k) \tag{3.46}$$

şeklinde yazılabilir. Çıkış işaretine ölçme gürültüsü karıştığını varsayarak aynı denklemi doğrusal bağlaşımlı biçimde

$$y(k) = \phi^T(k)\theta + v(k) \tag{3.47}$$

olarak yazdığımızda parametre vektörü

$$\theta = [\alpha_1 \quad \cdots \quad \alpha_n \quad \beta_1 \quad \cdots \quad \beta_n]^T \tag{3.48}$$

ve filtrelenmiş verilerden oluşan durum vektörü

$$\phi(k) = [-\lambda y(k) \quad \cdots \quad -\lambda^n y(k) \quad \lambda u(k) \quad \cdots \quad \lambda^n u(k)]^T \tag{3.49}$$

biçiminde yazılabilir. Burada $v(k)$ ölçme gürültüsünü, $y(k)$ gürültülü çıkış işaretini göstermektedir. Sonra filtreleme işlemi gerçekleştirilerek hem (3.46) ile verilen sistem denklemi sayısal hale getirilmiş olur, hem de çıkış işaretindeki istenmeyen gürültü bileşenleri filtrelenmiş olur. Filtrelenen giriş-çıkış verileri bilinen herhangi bir sistem tanıma algoritması kullanılarak sürekli zaman parametreleri tahmin edilmesinde kullanılır.

Sistemin diferansiyel denklem modelinin katsayı vektörü

$$\theta_s = [a_1 \quad \cdots \quad a_n \quad b_1 \quad \cdots \quad b_n]^T \tag{3.50}$$

ile aynı sisteme ait değiştirilmiş modelinin katsayıları arasında

$$\theta_s = F^{-1}(\theta - G) \tag{3.51}$$

matris ilişkisi vardır. Bu eşitlikte

$$F = \begin{bmatrix} M & 0_{n\times n} \\ 0_{n\times n} & M \end{bmatrix} \tag{3.52}$$

biçimindedir ve buradaki M matrisi

$$M = \begin{bmatrix} m_{11} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ m_{n1} & \cdots & \cdots & m_{nn} \end{bmatrix} \quad , \quad m_{ij} = (-1)^{i-j} \binom{n-j}{i-j} \tau^{j} \tag{3.53}$$

biçimindedir, $2n \times 1$ boyutlu G matrisi ise

$$G = [g_1 \quad \cdots \quad g_n \quad 0 \quad \cdots \quad 0]^T \quad , \quad g_i = \binom{n}{i} (-1)^i \tag{3.54}$$

olarak en genel haliyle yazılabilir (Johansson 1993, 1994). Bütün $\tau > 0$ değerleri için M matrisinin tersinin alınabildiği durumlarda F matrisinin de tersi alınabilir ve diferansiyel denklem katsayıları (3.51) ile verilen dönüşüm kullanılarak hesaplanabilir.

Sistemin çıkış işaretindeki sisteme ait frekans bileşenlerinin bastırılmaması için en yüksek dereceli filtrenin bant genişliğinin sistemin bant genişliğinden büyük olması gerekir. Pratik uygulamalarda en yüksek dereceli filtrenin bant genişliğinin sistemin bant genişliğine göre biraz daha geniş seçilmesi önerilmektedir (Sagara ve Zhao 1989,1990,1991, Sinha ve Rao 1991, Johansson 1994). Burada kullanılan n. derece filtrenin τ zaman sabitine bağlı bant genişliğini bulmak için filtrenin genlik denklemi

$$\left|\lambda^n(j\omega)\right| = \left|\lambda^n(j0)\right| / \sqrt{2} \tag{3.55}$$

eşitliğinde kullanıldığında

$$\tau = \sqrt{\sqrt[n]{2} - 1}\Big/\omega \tag{3.56}$$

sonucu elde edilir (Hatun 2002, Hatun ve Koçal 2007b). Bu denklem yardımıyla, sistemin bant genişliği yaklaşık olarak bilindiğinde, sistem tanıma işleminde kullanılan filtrelerin τ zaman sabiti hesaplanabilir veya herhangi bir τ değeri için kullanılan filtrenin bant genişliği hesaplanabilir.

3.3.3. Tekrarlamalı Gauss-Seidel algoritmasının kullanımı

Sistemin alçak geçiren filtre operatörüne bağlı denklemini matrisel olarak (3.47) ile verilen biçimde yazıldığında parametre tahmin işlemindeki tahmin hatası

$$e(k) = y(k) - \phi^T(k)\,\hat{\theta}(k-1) \tag{3.57}$$

olarak yazılabilir. Burada minimum yapılan fonksiyon

$$V_k(\theta) = \frac{1}{k}\sum_{i=1}^{k} \lambda^{k-i} e^2(i) \tag{3.58}$$

olup bu fonksiyonu minimum yapan optimum parametreler

$$\theta_{opt} = R(k)^{-1}\,p(k) \tag{3.59}$$

olarak verilebilir. Burada $R(k)$ ile $2n \times 2n$ boyutlu korelasyon matrisi, $p(k)$ ile $2n \times 1$ boyutlu korelasyon vektörü gösterilmektedir, $0 < \lambda \leq 1$ unutma faktörüdür. Pratik uygulamalarda $R(k)$ matrisinin ve $p(k)$ vektörünün değeri k adet veri kullanıldığında aşağıdaki gibi yazılabilir,

$$R(k)=\frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\phi(i)\phi^{T}(i) \quad , \quad p(k)=\frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\phi(i)y(i) \tag{3.60}$$

veya $1/k$ çarpanları göz önüne alınmadan giriş-çıkış verileri alındıkça ardışık olarak

$$R(k)=\lambda R(k-1)+\phi(k)\phi^{T}(k) \quad , \quad p(k)=\lambda r(k-1)+\phi(k)y(k) \tag{3.61}$$

şeklinde güncellenebilir. GS algoritmasıyla tekrarlamalı parametre tahmin işleminin başlangıç noktası, zaman ortalamalı normal denklemin GS iterasyonları ile çözümü üzerine kuruludur. Bu amaçla bir adımlık GS iterasyonu

$$R(k)\hat{\theta}(k)=p(k) \tag{3.62}$$

olarak yazılan zaman ortalamalı normal denklemin çözümünde

$$\hat{\theta}_i(k)=\left[p_i(k)-\sum_{j=1}^{i-1}R_{ij}(k)\hat{\theta}_j(k)-\sum_{j=i+1}^{2n}R_{ij}(k)\hat{\theta}_j(k-1)\right]\Big/R_{ii}(k) \tag{3.63}$$

şeklinde kullanılır, yani iterasyon indisi ayrık zaman indisi olarak alınmıştır (Hatun ve Koçal 2007b). Burada $R_{ij}(k)$ korelasyon matrisinin i. satırına ve j. sütununa denk düşen elemanını gösterir, $p_i(k)$ korelasyon vektörünün i. elemanını gösterir ve $\hat{\theta}_i(k)$ parametre vektörünün i. elemanını gösterir. Yukarıda (3.61) ve (3.63) ile verilen algoritma tekrarlamalı GS algoritması olarak adlandırılmıştır. Tekrarlamalı GS algoritmasının RLS algoritmasına göre en önemli farkı korelasyon matrisinin tersi yerine doğrudan kendisinin güncellenmesi, yani matris tersini güncellemek için kullanılan iterasyona gerek kalmaması ve parametre tahminlerini güncellemek için anlık hata bilgisini doğrudan kullanmamasıdır. Ayrıca parametrelerin skaler olarak güncellendiğini de göz önüne aldığımızda, RLS algoritmasına göre işlem yükünün önemli ölçüde azaldığı görülmektedir (Koçal 1998, Bose 2004).

3.3.4. Tekrarlamalı Gauss-Seidel algoritmasının yakınsama analizi

Parametre tahminlerini çevrimiçi olarak güncellemek için kullanılan tekrarlamalı GS algoritması

$$\hat{\theta}(k) = -(R_L(k) + R_D(k))^{-1} R_U(k)\hat{\theta}(k-1) + (R_L(k) + R_D(k))^{-1} p(k) \tag{3.64}$$

olarak yazılabilir. Burada $R_L(k)$, $R_D(k)$ ve $R_U(k)$ sırasıyla korelasyon matrisinin alt üçgen, köşegen ve üst üçgen kısımlarını içeren kare matrisleri göstermektedir. Sistemin doğrusal bağlaşımlı biçimde yazılmış çıkış işareti $p(k)$ korelasyon vektörünün tahmini değerinde yerine yazıldığında

$$p(k) = \frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\phi(i)y(i) = R(k)\,\theta_{opt} + \frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\phi(i)v(i) \tag{3.65}$$

elde edilir. Bu ifade yukarıdaki (3.64) eşitliğinde kullanılırsa

$$\hat{\theta}(k) = -(R_L(k)+R_D(k))^{-1} R_U(k)\hat{\theta}(k-1) + (R_L(k)+R_D(k))^{-1}\left[R(k)\,\theta_{opt} + \frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\phi(i)v(i)\right] \tag{3.66}$$

eşitliği elde edilir. Bu eşitlikte korelasyon matrisi $R(k) = R_L(k) + R_D(k) + R_U(k)$ şeklinde yerine yazılarak denklem düzenlendiğinde

$$\begin{aligned}\hat{\theta}(k) = &-(R_L(k) + R_D(k))^{-1} R_U(k)\hat{\theta}(k-1) + \left[I + (R_L(k) + R_D(k))^{-1} R_U(k)\right]\theta_{opt} \\ &+ (R_L(k) + R_D(k))^{-1}\left[\frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\phi(i)v(i)\right]\end{aligned} \tag{3.67}$$

elde edilir. Bu eşitliğin her iki tarafının beklenen değeri alınarak

$$E[\hat{\theta}(k)] = E[\hat{\theta}(k-1)]\,,\quad E[\phi(k)v(k)] = \frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\phi(i)v(i)$$
$$E[(R_L(k)+R_D(k))^{-1}] = (R_L + R_D)^{-1} \tag{3.68}$$
$$E[I+(R_L(k)+R_D(k))^{-1}R_U(k)] = I+(R_L+R_D)^{-1}R_U$$

olarak verilen kabuller kullanılırsa, ve parametre tahminlerinin beklenen değeri $E[\hat{\theta}(k)]$ çekilerek denklem uygun şekilde düzenlenirse

$$E[\hat{\theta}(k)] = \theta_{opt} + [I+(R_L+R_D)^{-1}R_U]^{-1}(R_L+R_D)^{-1}E[\phi(k)v(k)] \tag{3.69}$$

eşitliği elde edilir, ve $I+(R_L+R_D)^{-1}R_U = (R_L+R_D)^{-1}R$ matris eşitliği kullanıldığında

$$E[\hat{\theta}(k)] = \theta_{opt} + (R_L+R_D)R^{-1}(R_L+R_D)^{-1}E[\phi(k)v(k)] \tag{3.70}$$

sonucu elde edilir (Hatun ve Koçal 2007b). Elde edilen bu sonuç tekrarlamalı GS algoritması ile elde edilen parametre tahminlerinin optimum değerine yakınsaması için gerekli şartları göstermektedir. Bunun için öncelikle korelasyon matrisinin tersi alınabilmelidir. Bunun için korelasyon matrisi pozitif tanımlı olmalıdır. Sistemin sürekli olarak uyarılması durumunda bu şart sağlanmaktadır (Söderström ve Stoica 1989, Ljung 1999). Çünkü kullanılan kare matrisinin pozitif tanımlı olması durumunda GS iterasyonlarının yakınsaması sağlanmış olmaktadır (Golub ve Van Loan 1996). Pratik uygulamalarda başlangıçta korelasyon matrisinin pozitif tanımlılığını garantilemek amacıyla $R(0) = \delta.I$ olarak alınabilir, burada I uygun boyutlu birim matris olup $\delta = 0.1,\ 0.01,\ 0.001$ gibi küçük pozitif değerler alabilir. Algoritmanın kararlılığı sağlandıktan sonra elde edilen parametre tahminlerinin optimum değerine yakınsaması için gerekli olan diğer şart ise sistemden giriş-çıkış örnekleri alındıkça $E[\phi(k)v(k)]$ vektörünün sıfır vektöre yakınsamasıdır. Bu vektörün sıfır vektöre yakınsaması için $v(k)$ işaretinin $\phi(k)$ veri vektörünün elemanlarıyla ilişkisiz olması gerekir. Bu vektörün elemanları sistemin filtrelenmiş $y(k)$ çıkış verilerini içerdiğinden dolayı, $y(k)$ işaretinin önceki değerlerini, yani $v(k)$ gürültü işaretinin de önceki değerlerini içerir.

Buradan, $E[\phi(k)v(k)]$ vektörünün sıfır vektöre yakınsaması için ölçme gürültüsünün beyaz gürültü olması gerektiği, yani önceki örnekleriyle ilişkisiz olması gerektiği sonucu ortaya çıkar. Sistemin çıkış işaretine karışan ölçme gürültüsünün renkli gürültü olması durumunda ise yanlı parametre tahminleri elde edilmektedir. Sistem tanıma işleminde eşitlik hatasının kullanılması durumunda ortaya çıkan bu sorunu aşmanın bir yolu yardımcı değişkenler olarak bilinen, gürültü ile ilişkisiz olan işaretlerden yararlanmaktır (Söderström ve Stoica 1981, 1983, 1989, 2002). Ölçme gürültüsünün sistem ile birlikte modellenmesi durumunda ise sistem parametrelerine ilave olarak gürültü modeli parametrelerinin de hesaplanması gerekmektedir. Bu durumda tahmin edilen parametre sayısındaki artış algoritmadaki işlem yükünü arttırdığından dolayı burada sadece yardımcı değişkenlerin tekrarlamalı GS algoritmasıyla birlikte kullanımı önerilmiştir.

3.3.5. Tekrarlamalı Gauss-Seidel yardımcı değişkenler algoritması ile sürekli model tahmini

Yardımcı değişkenlerin en yaygın kullanım biçimi Şekil 3.2'de görüldüğü gibi sistemle aynı yapıya sahip ve parametreleri sistem parametreleriyle güncellenen bir uyarlamalı yardımcı model kullanmaktır. Ayrıca sistemle aynı girişi kullanan sabit katsayılı, kararlı bir yardımcı modelin çıkış işareti, geciktirilmiş giriş işareti ve geciktirilmiş çıkış işareti de yardımcı değişken olarak kullanılabilmektedir (Söderström ve Stoica 1981, 1983, 1989, 2002, Ljung 1999). Burada yardımcı değişkenleri elde etmek için sistemle aynı yapıda uyarlamalı bir yardımcı model kullanılmıştır. Sistemle aynı yapıya sahip olan bu yardımcı modelin denklemi

$$\hat{x}(k)+\hat{\alpha}_1\lambda\,\hat{x}(k)+\cdots+\hat{\alpha}_n\lambda^n\hat{x}(k)=\hat{\beta}_1\lambda\,u(k)+\cdots+\hat{\beta}_n\lambda^n u(k) \tag{3.71}$$

olarak yazılabilir. Yardımcı modelin girişi sistemin girişi ile aynı olup, parametreleri sistem parametreleri ile güncellenir ve çıkışında ise sistemin gürültüsüz çıkış işareti tahmin edilir. Yardımcı değişken olarak bilinen bu işaret (3.71) denkleminden hesaplanabilir. Bunun için alçak geçiren filtrelerin fark denklemleri

$$\overline{\lambda}^i \hat{x}(k) = \lambda^i \hat{x}(k) - q^i \hat{x}(k) \quad , \quad i = 0,1,\ldots,n \tag{3.72}$$

tanımlaması kullanılarak $\hat{x}(k)$ içeren terimleri eksik olarak

$$\overline{\lambda}^i x(k) = p\lambda^i x(k-1) + q\left(\overline{\lambda}^{i-1} x(k) + \lambda^{i-1} x(k-1)\right) \tag{3.73}$$

şeklinde yazılabilir. Giriş işareti alçak geçiren filtrenin fark denklemi kullanılarak filtrelenebilir. Burada son iki eşitlik kullanılarak (3.71) denkleminden tek bilinmeyen $\hat{x}(k)$ değeri

$$\hat{x}(k) = [-\hat{\alpha}_1 \overline{\lambda}\, \hat{x}(k) - \cdots - \hat{\alpha}_n \overline{\lambda}^n \hat{x}(k) + \hat{\beta}_1 \lambda u(k) + \cdots + \hat{\beta}_n \lambda^n u(k)] \big/ (1 + \hat{\alpha}_1 q + \hat{\alpha}_2 q^2 + \cdots + \alpha_n q^n) \tag{3.74}$$

şeklinde önceki giriş ve çıkış verileri kullanılarak hesaplanabilir. Daha sonra elde edilen $\hat{x}(k)$ bilgisi (3.72) denkleminde kullanılarak

$$\lambda^i \hat{x}(k) = \overline{\lambda}^i \hat{x}(k) + q^i \hat{x}(k) \quad , \quad i = 0,1,\ldots,n \tag{3.75}$$

değerleri hesaplanır ve yardımcı modelin durum vektörü

$$z(k) = [-\lambda\, \hat{x}(k) \quad \cdots \quad -\lambda^n \hat{x}(k) \quad \lambda u(k) \quad \cdots \quad \lambda^n u(k)]^T \tag{3.76}$$

şeklinde oluşturulur. $R(k)$ matrisi ile $p(k)$ vektörü ise

$$R(k) = \lambda\, R(k-1) + z(k)\phi^T(k) \ , \quad p(k) = \lambda\, r(k-1) + z(k) y(k) \tag{3.77}$$

şeklinde iteratif olarak güncellenir. Daha sonra önerilen tekrarlamalı GS algoritması zaman ortalamalı normal denklemin çözümünde kullanılır. Burada (3.73)-(3.77) ve (3.63) eşitlikleriyle verilen çevrimiçi yöntem, tekrarlamalı GSYD algoritması olarak adlandırılmıştır.

Tekrarlamalı GSYD algoritmasının yakınsama analizi tekrarlamalı GS algoritmasına benzer şekilde yapıldığında

$$E[\hat{\theta}(k)] = \theta_{opt} + (R_L + R_D)R^{-1}(R_L + R_D)^{-1} E[z(k)v(k)] \tag{3.78}$$

sonucuna ulaşılır. Bu sonuca göre $z(k)$ yardımcı durum vektörü gürültü ile ilişkisiz $\hat{x}(k)$ işaretini içerdiği için, sistemden giriş-çıkış örnekleri alındıkça $E[z(k)v(k)]$ vektörü sıfır vektöre yakınsayacaktır. Bu durumda (3.78) denklemindeki ikinci terim sıfır vektöre yakınsayacak ve elde edilen parametre tahminlerinin beklenen değeri optimum değerine yakınsayacaktır.

3.4. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Çok-Girişli Çok-Çıkışlı Sürekli-Zaman Sistemlerin Transfer Fonksiyonu Matrisi Parametrelerinin Yansız Tahmini

Doğrusal zamanla değişmeyen q-girişli r-çıkışlı sürekli-zamanlı bir sistemin transfer fonksiyonu matrisi s-domeninde aşağıdaki gibi yazılabilir (Mathew ve Fairman 1974, Unbehauen ve Rao 1987, Sagara ve Zhao 1989, Sinha ve Rao 1991, Datta ve Mohan 1995).

$$\begin{bmatrix} X_1(s) \\ X_2(s) \\ \vdots \\ X_r(s) \end{bmatrix} = \begin{bmatrix} \dfrac{B_{11}(s)}{A_1(s)} & \dfrac{B_{12}(s)}{A_1(s)} & \cdots & \dfrac{B_{1q}(s)}{A_1(s)} \\ \dfrac{B_{21}(s)}{A_2(s)} & \dfrac{B_{22}(s)}{A_2(s)} & \cdots & \dfrac{B_{2q}(s)}{A_2(s)} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{B_{r1}(s)}{A_r(s)} & \dfrac{B_{r2}(s)}{A_r(s)} & \cdots & \dfrac{B_{rq}(s)}{A_r(s)} \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \\ \vdots \\ U_q(s) \end{bmatrix} \tag{3.79}$$

Bu denklemde ($i = 1,2,\ldots,q$) için $U_i(s) = L[u_i(t)]$ şeklinde i. giriş işaretinin Laplace dönüşümü, ($j = 1,2,\ldots,r$) için $X_j(s) = L[x_j(t)]$ şeklinde j. çıkış işaretinin Laplace dönüşümü gösterilmektedir. Sistemin her bir çıkış işareti birbirinden bağımsız olarak çok-girişli tek-çıkışlı sistem biçiminde aşağıdaki gibi yazılabilir.

$$X_j(s) = \frac{B_{j1}(s)}{A_j(s)}U_1(s) + \frac{B_{j2}(s)}{A_j(s)}U_2(s) + \cdots + \frac{B_{jq}(s)}{A_j(s)}U_q(s) \quad , \quad (j = 1,2,\ldots,r) \tag{3.80}$$

Bu denklemlerin her iki tarafı $A_j(s)$ ile çarpılarak polinomal formda

$$A_j(s)X_j(s) = B_{j1}(s)U_1(s) + B_{j2}(s)U_2(s) + \cdots + B_{jq}(s)U_q(s) \quad , \quad (j = 1,2,\ldots,r) \tag{3.81}$$

olarak yazılabilir. Bu denklemlerde ($j = 1,2,\ldots,r$) için j. çıkışa ait payda polinomu $A_j(s)$ ve j. çıkışa ait pay polinomları $B_{j1}(s), B_{j2}(s)$, … , $B_{jq}(s)$ sırasıyla

$$\begin{aligned}
A_j(s) &= s^{n_j} + a_{j1}s^{n_j - 1} + \cdots + a_{jn_j} \\
B_{j1}(s) &= b_{j11}s^{m_{j1} - 1} + \cdots + b_{j1m_{j1}} \\
B_{j2}(s) &= b_{j21}s^{m_{j2} - 1} + \cdots + b_{j2m_{j2}} \\
&\vdots \\
B_{jq}(s) &= b_{jq1}s^{m_{jq} - 1} + \cdots + b_{jqm_{jq}}
\end{aligned} \tag{3.82}$$

şeklinde yazılabilir, burada n_j, m_{j1}, m_{j2}, … , m_{jq} ile sırasıyla $A_j(s), B_{j1}(s), B_{j2}(s)$, … , $B_{jq}(s)$ polinomlarının derecesi gösterilmektedir ve pay polinomlarının dereceleri $(m_{j1}, m_{j2}, \ldots, m_{jq}) < n_j$ şeklinde payda polinomunun derecesinden küçük olduğu varsayılmıştır. Bu denklemlerdeki polinomlar s türev operatörüne bağlıdır. Bilindiği gibi türev işlemi yüksek geçiren filtre karakteristiğine sahiptir ve sistemin çıkışından ölçülerek elde edilen gürültülü çıkış işaretlerinin türevinin alınması işlemi, yüksek frekanslı gürültü bileşenlerini kuvvetlendirmekte ve sisteme ait alçak frekanslı bileşenleri bastırmaktadır (Unbehauen ve Rao 1987, Sagara ve Zhao 1989, Sinha ve Rao 1991, Johansson 1993,1994). Türev işlemi sonucunda sisteme ait frekans bileşenleri bastırıldığı için sisteme ait bilgi kaybolmaktadır ve aynı zamanda kuvvetlendirilen gürültü bileşenleri parametre tahmin işlemini olumsuz etkilemektedir. Parametre tahmin işlemini olumsuz etkileyen türev işleminden kaçınmak amacıyla, Johansson (1993, 1994) tarafından önerilen alçak geçiren durum değişkeni filtreleri kullanılabilir. Bunun için (3.81) ile verilen s türev operatörüne bağlı polinomal denklemler, türev operatörü yerine nedensel, kararlı ve gerçeklenebilir bir doğrusal operatöre bağlı olacak şekilde tekrar düzenlenebilir. Bu amaçla alçak geçiren filtre operatörleri içeren bir doğrusal model ele alınmış ve sürekli-zaman sistem modelinin parametreleri ile seçilen alçak geçiren filtre çıkışları arasında ilişki kuran doğrusal bir dönüşüm yapılmıştır. Burada kullanılan alçak geçiren filtre operatörü

$$\lambda = \frac{a}{s+a} = \frac{1}{1+s\tau} \quad , \quad \tau = 1/a \tag{3.83}$$

olarak tanımlandığında, bu tanımlama bize

$$\lambda = \frac{1}{1+s\tau} \quad \leftrightarrow \quad s = \frac{1-\lambda}{\lambda\tau} \tag{3.84}$$

dönüşümünü yapma imkanı verir. Bu operatör dönüşümü (3.81) ile verilen s türev operatörüne bağlı olan polinomal denklemlere uygulandığında

$$A_j(\lambda)X_j(s) = B_{j1}(\lambda)U_1(s) + B_{j2}(\lambda)U_2(s) + \cdots + B_{jq}(\lambda)U_q(s) \quad , \quad (j = 1,2,\ldots,r) \tag{3.85}$$

denklemi elde edilir. Bu değiştirilmiş denklemlerde ($j = 1,2,\ldots,r$) için j. çıkışa ait payda polinomu $A_j(\lambda)$ ve j. çıkışa ait pay polinomları $B_{j1}(\lambda), B_{j2}(\lambda)$, … , $B_{jq}(\lambda)$ sırasıyla

$$\begin{aligned}
A_j(\lambda) &= 1 + \alpha_{j1}\lambda + \cdots + \alpha_{jn_j}\lambda^{n_j} \\
B_{j1}(\lambda) &= \beta_{j11}\lambda + \cdots + \beta_{j1m_{j1}}\lambda^{m_{j1}} \\
B_{j2}(\lambda) &= \beta_{j21}\lambda + \cdots + \beta_{j2m_{j2}}\lambda^{m_{j2}} \\
&\vdots \\
B_{jq}(\lambda) &= \beta_{jq1}\lambda + \cdots + \beta_{jqm_{jq}}\lambda^{m_{jq}}
\end{aligned} \tag{3.86}$$

şeklinde yazılabilir, burada yine $n_j, m_{j1}, m_{j2}, \ \ldots \ , m_{jq}$ ile sırasıyla $A_j(\lambda), B_{j1}(\lambda), B_{j2}(\lambda), \ \ldots \ , \ B_{jq}(\lambda)$ polinomlarının derecesi gösterilmektedir ve pay polinomlarının dereceleri $(m_{j1}, m_{j2}, \ \ldots, \ m_{jq}) < n_j$ şeklinde payda polinomunun derecesinden küçük olduğu varsayılmıştır. Burada (3.85) ile verilen λ alçak geçiren filtre operatörüne bağlı denklemi daha açık olarak

$$
\begin{aligned}
X_j(s) + \alpha_{j1}\lambda X_j(s) + \cdots + \alpha_{jn_j}\lambda^{n_j} X_j(s) = \beta_{j11}\lambda U_1(s) + \cdots + \beta_{j1m_{j1}}\lambda^{m_{j1}} U_1(s) \\
+ \beta_{j21}\lambda U_2(s) + \cdots + \beta_{j2m_{j2}}\lambda^{m_{j2}} U_2(s) \\
\vdots \qquad\qquad\qquad\qquad \\
+ \beta_{jq1}\lambda U_q(s) + \cdots + \beta_{jqm_{jq}}\lambda^{m_{jq}} U_q(s)
\end{aligned} \tag{3.87}
$$

şeklinde yazabiliriz. Burada $\lambda^i X_j(s)$ ile i katlı alçak geçiren filtre ile filtrelenmiş $j.$ çıkış işareti ve $\lambda^i U_j(s)$ ile i katlı alçak geçiren filtre ile filtrelenmiş $j.$ giriş işareti gösterilmektedir. Kullanılan giriş ve çıkış işaretleri ayrık olduğu için filtreleme işlemi de ayrık olarak yapılmalıdır. Alçak geçiren filtreleme işlemini ayrık olarak gerçeklemenin en hızlı ve kolay yolu filtrenin IIR formdaki ayrık zaman modelini kullanmaktır (Sinha ve Rao 1991). Bu amaçla alçak geçiren filtrenin bilineer dönüşüm ile elde edilen Tustin eşdeğeri kullanıldığında tek katlı filtrenin fark denklemi

$$\lambda\, x(k) = \mathrm{p}.\lambda\, x(k-1) + \mathrm{q}.\big(x(k) + x(k-1)\big) \tag{3.88}$$

olarak yazılabilir (Hatun 2002, Hatun ve Koçal 2007a). Burada k ayrık zaman indisi olup,

$$\mathrm{p} = \frac{2 - aT}{2 + aT} \quad , \quad \mathrm{q} = \frac{aT}{2 + aT} \tag{3.89}$$

filtrenin fark denkleminin katsayıları, T ise örnekleme periyodudur. Daha yüksek dereceli filtrelenmiş işaretler aynı fark denklemi kullanılarak $i = 1,2,\ldots$ için

$$\lambda^{i} x(k) = \mathrm{p}.\lambda^{i} x(k-1) + \mathrm{q}.\left(\lambda^{i-1} x(k) + \lambda^{i-1} x(k-1)\right) \tag{3.90}$$

şeklinde ardışık olarak hesaplanabilir. Aynı filtreleme işlemini giriş işaretlerine de benzer şekilde uygulandığında (3.87) denklemi ile verilen sürekli-zaman sistem modeli filtrelenmiş ayrık giriş ve çıkış işaretlerine bağlı olarak

$$\begin{aligned} x_j(k) + \alpha_{j1}\lambda\, x_j(k) + \cdots + \alpha_{jn_j}\lambda^{n_j} x_j(k) = {} & \beta_{j11}\lambda\, u_1(k) + \cdots + \beta_{j1m_{j1}}\lambda^{m_{j1}} u_1(k) \\ & + \beta_{j21}\lambda\, u_2(k) + \cdots + \beta_{j2m_{j2}}\lambda^{m_{j2}} u_2(k) \\ & \;\vdots \\ & + \beta_{jq1}\lambda\, u_q(k) + \cdots + \beta_{jqm_{jq}}\lambda^{m_{jq}} u_q(k) \end{aligned} \tag{3.91}$$

şeklinde yazılabilir. Açık çevrimde sistem belirleme işleminde, sistemin çıkış işareti ölçümlerine ($j = 1,2,\ldots,r$) için $y_j(k) = x_j(k) + v_j(k)$ şeklinde renkli ölçme gürültüsü karıştığını varsaydığımızda (3.91) ile verilen fark denklemleri ($j = 1,2,\ldots,r$) için

$$\begin{aligned} y_j(k) = {} & -\alpha_{j1}\lambda\, y_j(k) - \cdots - \alpha_{jn_j}\lambda^{n_j} y_j(k) \\ & + \beta_{j11}\lambda\, u_1(k) + \cdots + \beta_{j1m_{j1}}\lambda^{m_{j1}} u_1(k) \\ & + \beta_{j21}\lambda\, u_2(k) + \cdots + \beta_{j2m_{j2}}\lambda^{m_{j2}} u_2(k) \\ & \;\vdots \\ & + \beta_{jq1}\lambda\, u_q(k) + \cdots + \beta_{jqm_{jq}}\lambda^{m_{jq}} u_q(k) + v_j(k) \end{aligned} \tag{3.92}$$

şeklinde giriş ve çıkış örneklerine bağlı olarak yazılabilir. Bu denklemleri sistem tanıma işleminde kullanabilmek için doğrusal bağlanımlı biçimde

$$y_j(k) = \varphi_j^T(k)\theta_j + v_j(k) \quad , \quad (j = 1,2,\ldots,r) \tag{3.93}$$

şeklinde yazıldığında $j.$ çıkışa ait veri vektörü ve parametre vektörü sırasıyla

$$\varphi_j^T(k) = [-\lambda y_j(k) \cdots -\lambda^{n_j} y_j(k)\ \lambda u_1(k) \cdots \lambda^{m_{j1}} u_1(k)\ \lambda u_2(k) \cdots \lambda^{m_{j2}} u_2(k) \cdots \lambda u_q(k) \cdots \lambda^{m_{jq}} u_q(k)] \tag{3.94}$$

$$\theta_j^T = [\alpha_{j1}\ \ \alpha_{j2}\ \cdots\ \alpha_{jn_j}\ \ \beta_{j11}\ \cdots\ \beta_{j1m_{j1}}\ \ \beta_{j21}\ \cdots\ \beta_{j2m_{j2}}\ \cdots\ \beta_{jq1}\ \cdots\ \beta_{jqm_{jq}}] \tag{3.95}$$

olarak yazılabilir. Bu iki vektör parametre sayısı $M_j = n_j + m_{j1} + \cdots + m_{jq}$ olmak üzere $(M_j \times 1)$ boyutlu sütun vektördür. Burada alçak geçiren filtreleme işlemi sayısal olarak gerçeklendiğinde hem veri vektöründeki filtrelenmiş işaretler sistem tanıma işleminde kullanılmak üzere sayısal olarak elde edilir, hem de çıkış işaretine karışan istenmeyen gürültü bileşenleri filtrelenmiş olur. Filtrelenmiş giriş-çıkış verileri bilinen herhangi bir sistem tanıma algoritması kullanılarak θ_j ($j = 1,2,\ldots,r$) vektörlerindeki sürekli zaman sistem modeli parametrelerinin tahmin edilmesinde kullanılır.

Sistemin s -domeni modelinin katsayı vektörleri

$$\theta_{sj}^T = [a_{j1}\ \ a_{j2}\ \cdots\ a_{jn_j}\ \ b_{j11}\ \cdots\ b_{j1m_{j1}}\ \ b_{j21}\ \cdots\ b_{j2m_{j2}}\ \cdots\ b_{jq1}\ \cdots\ b_{jqm_{jq}}] \tag{3.96}$$

ile aynı sisteme ait λ alçak geçiren filtre operatörüne bağlı değiştirilmiş sistem modeline ait

$$\theta_j^T = [\alpha_{j1}\ \ \alpha_{j2}\ \cdots\ \alpha_{jn_j}\ \ \beta_{j11}\ \cdots\ \beta_{j1m_{j1}}\ \ \beta_{j21}\ \cdots\ \beta_{j2m_{j2}}\ \cdots\ \beta_{jq1}\ \cdots\ \beta_{jqm_{jq}}] \tag{3.97}$$

parametre vektörleri arasında

$$\theta_{sj} = F^{-1}(\theta_j - G) \tag{3.98}$$

matris ilişkisi vardır. Bu eşitlikteki F matrisi çok-girişli çok-çıkışlı sistemler için

$$F = \begin{bmatrix} M_{n_j \times n_j} & 0 & \cdots & 0 \\ 0 & M_{m_{j1} \times m_{j1}} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & M_{m_{jq} \times m_{jq}} \end{bmatrix} \tag{3.99}$$

biçiminde yazılabilir ve M matrisi

$$M = \begin{bmatrix} m_{11} & 0 & \cdots & 0 \\ \vdots & m_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ m_{n1} & \cdots & \cdots & m_{nn} \end{bmatrix} \quad , \quad m_{ij} = (-1)^{i-j} \binom{n-j}{i-j} \tau^{j} \tag{3.100}$$

biçimindedir, ve F matrisiyle uygun boyutlu G sütün matrisi ise

$$G = [g_1 \ \cdots \ g_{n_j} \ \ 0 \ \cdots \ 0]^T \quad , \quad g_i = (-1)^{i-j} \binom{n}{i} (-1)^{i} \tag{3.101}$$

olarak en genel haliyle yazılabilir. Burada bütün $\tau > 0$ değerleri için M matrisinin tersinin alınabildiği durumlarda F matrisinin de tersi alınabilir ve s-domeni modelinin katsayıları (3.98) ile verilen dönüşüm kullanılarak hesaplanabilir.

Parametre tahmininde her bir çıkışa ait tahmin hatası

$$\varepsilon_j(k) = y_j(k) - \varphi_j^T(k)\hat{\theta}_j(k-1) \quad , \quad (j = 1,2,\ldots,r) \tag{3.102}$$

olarak tanımlandığında, minimum yapılmak istenen hata fonksiyonu

$$V\big(\theta_j(k)\big) = \frac{1}{k} \sum_{i=1}^{k} \lambda^{k-i} \boldsymbol{\varepsilon}^T(i)\boldsymbol{\varepsilon}(i) \tag{3.103}$$

olarak yazılabilir, burada $0 < \lambda \leq 1$ unutma faktörüdür ve hata vektörleri ($i = 1,2,\ldots,k$) için

$$\boldsymbol{\varepsilon}(i)=\begin{bmatrix}\varepsilon_1(i)\\ \varepsilon_2(i)\\ \vdots\\ \varepsilon_r(i)\end{bmatrix}=\begin{bmatrix}y_1(i)-\varphi_1^T(i)\hat{\theta}_1(i-1)\\ y_2(i)-\varphi_2^T(i)\hat{\theta}_2(i-1)\\ \vdots\\ y_r(i)-\varphi_r^T(i)\hat{\theta}_r(i-1)\end{bmatrix} \tag{3.104}$$

olarak yazılabilir. Bu hata fonksiyonunu minimum yapan j. çıkışa ait optimum parametre vektörü, θ_j 'ye göre türev alındığında

$$\hat{\theta}_{jopt}(k)=R_j^{-1}(k)p_j(k) \tag{3.105}$$

olarak hesaplanır. Burada j. çıkışa ait $(M_j\times M_j)$ boyutlu korelasyon matrisi $R_j(k)$ ve $(M_j\times 1)$ boyutlu korelasyon vektörü $p_j(k)$ aşağıdaki gibi yazılabilir,

$$R_j(k)=\frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\varphi_j(i)\varphi_j^T(i) \quad , \quad p_j(k)=\frac{1}{k}\sum_{i=1}^{k}\lambda^{k-i}\varphi_j(i)y_j(i) \tag{3.106}$$

veya çevrim-içi (on-line) uygulamalarda $1/k$ çarpanı göz önüne alınmadan giriş-çıkış veri örnekleri alındıkça iteratif olarak aşağıdaki gibi güncellenebilir (Sinha ve Kuszta 1983, Söderström ve Stoica 1989, Ljung 1999).

$$R_j(k)=\lambda R_j(k)+\varphi_j(k)\varphi_j^T(k) \quad , \quad p_j(k)=\lambda p_j(k)+\varphi_j(k)y_j(k) \tag{3.107}$$

$R_j(k)$ matrisi ve $p_j(k)$ vektörü iteratif olarak güncellendikten sonra bir adımlık GS iterasyonu, çözümü (3.105) ile verilen zaman ortalamalı $R_j(k)\theta_{jopt}(k)=p_j(k)$ normal denkleminin çözümünde

$$\hat{\theta}_{ji}(k)=\left[p_{ji}(k)-\sum_{l=1}^{i-1}R_{jil}(k)\hat{\theta}_{jl}(k)-\sum_{l=i+1}^{M_j}R_{jil}(k)\hat{\theta}_{jl}(k-1)\right]\Bigg/R_{jii}(k)$$
$$(i=1,2,\ldots,M_j) \tag{3.108}$$

şeklinde kullanılır, yani iterasyon indisi ayrık zaman indisi olarak alınmıştır. Burada $R_{jil}(k)$ korelasyon matrisinin i. satırına ve l. sütununa denk düşen elemanını gösterir, $p_{ji}(k)$ korelasyon vektörünün i. elemanını gösterir ve $\hat{\theta}_{ji}(k)$ parametre vektörünün i. elemanını gösterir. Yukarıda (3.107) ve (3.108) ile verilen tekrarlamalı GS algoritması sürekli-zaman sistem modeli parametrelerinin tahmin edilmesinde kullanılabilir. Tekrarlamalı GS algoritmasında yardımcı değişkenlerin kullanılmasıyla elde edilen tekrarlamalı algoritma tekrarlamalı GSYD algoritması olarak adlandırılmıştır (Hatun ve Koçal 2007a, 2007b). Yardımcı değişkenlerin en yaygın kullanım biçimi Şekil 3.3'te görüldüğü gibidir. Burada sistemle aynı yapıya sahip olan ve parametreleri sistem parametreleriyle eşzamanlı olarak güncellenerek kullanılan çok-girişli çok-çıkışlı bir uyarlamalı yardımcı modelin denklemi

$$\begin{aligned}
\hat{x}_j(k) + \hat{\alpha}_{j1}\lambda\hat{x}_j(k) + \cdots + \hat{\alpha}_{jn_j}\lambda^{n_j}\hat{x}_j(k) = {} & \hat{\beta}_{j11}\lambda u_1(k) + \cdots + \hat{\beta}_{j1m_{j1}}\lambda^{m_{j1}}u_1(k) \\
& + \hat{\beta}_{j21}\lambda u_2(k) + \cdots + \hat{\beta}_{j2m_{j2}}\lambda^{m_{j2}}u_2(k) \\
& \quad\vdots \\
& + \hat{\beta}_{jq1}\lambda u_q(k) + \cdots + \hat{\beta}_{jqm_{jq}}\lambda^{m_{jq}}u_q(k)
\end{aligned} \tag{3.109}$$

olarak yazılabilir. Yardımcı değişkenleri elde etmek için sistemle aynı yapıya sahip bu uyarlamalı yardımcı model kullanıldığında, bu yardımcı modele ait veri vektörü

$$z_j^T(k) = [-\lambda\hat{x}_j(k) \ \cdots -\lambda^{n_j}\hat{x}_j(k) \ \ \lambda u_1(k) \ \cdots \ \lambda^{m_{j1}}u_1(k) \ \ \lambda u_2(k) \ \cdots \ \lambda^{m_{j2}}u_2(k) \ \cdots \ \lambda u_q(k) \ \cdots \ \lambda^{m_{jq}}u_q(k)] \tag{3.110}$$

olarak yazılabilir. Yardımcı modelin girişleri sistem ile aynı olup, parametreleri sistem parametreleri ile eşzamanlı olarak güncellenir ve modelin çıkışında sistemin gürültüsüz çıkış işaretleri tahmin edilir. Yardımcı modelin girişleri sistem ile aynı olup, parametreleri sistem parametreleri ile eşzamanlı olarak güncellenir ve modelin çıkışında sistemin gürültüsüz çıkış işaretleri tahmin edilir. Yardımcı değişkenler olarak bilinen bu gürültüsüz çıkış işaretleri (3.109) denkleminden hesaplanabilir. Bunun için filtrelerin fark denklemleri

$$\overline{\lambda}^i \hat{x}_j(k) = \lambda^i \hat{x}_j(k) - \mathrm{q}^i \hat{x}_j(k) \quad , \quad i = 0,1,...,n_j \quad , \quad j = 1,2,...,r \tag{3.111}$$

tanımlaması kullanılarak $\hat{x}_j(k)$ içeren terimleri eksik olarak $i = 0,1,...,n_j$ ve $j = 1,2,...,r$ için

$$\overline{\lambda}^i \hat{x}_j(k) = \mathrm{p}.\lambda^i \hat{x}_j(k-1) + \mathrm{q}.\left(\overline{\lambda}^{i-1} \hat{x}_j(k) + \lambda^{i-1} \hat{x}_j(k-1)\right) \tag{3.112}$$

şeklinde yazılabilir. Giriş işaretleri de aynı fark denklemleri kullanılarak $j = 1,2,...,q$ için

$$\lambda^i u_j(k) = \mathrm{p}.\lambda^i u_j(k-1) + \mathrm{q}.\left(\lambda^{i-1} u_j(k) + \lambda^{i-1} u_j(k-1)\right) \tag{3.113}$$

şeklinde filtrelenebilir. Burada son üç eşitlik kullanılarak (3.109) denklemindeki bilinmeyen j. gürültüsüz çıkış işaretinin k anındaki tahmini değeri

$$\begin{aligned} \hat{x}_j(k) = \Big[& -\hat{\alpha}_{j1}\overline{\lambda}\hat{x}_j(k) - \cdots - \hat{\alpha}_{jn_j}\overline{\lambda}^{n_j}\hat{x}_j(k) \\ & + \hat{\beta}_{j11}\lambda u_1(k) + \cdots + \hat{\beta}_{j1m_{j1}}\lambda^{m_{j1}} u_1(k) \\ & + \hat{\beta}_{j21}\lambda u_2(k) + \cdots + \hat{\beta}_{j2m_{j2}}\lambda^{m_{j2}} u_2(k) + \cdots \\ & + \hat{\beta}_{jq1}\lambda u_q(k) + \cdots + \hat{\beta}_{jqm_{jq}}\lambda^{m_{jq}} u_q(k)\Big] \Big/ \left(1 + \hat{\alpha}_{j1}\mathrm{q} + \cdots + \hat{\alpha}_{jn_j}\mathrm{q}^{n_j}\right) \end{aligned} \tag{3.114}$$

şeklinde önceki giriş ve çıkış verileri kullanılarak hesaplanabilir. Daha sonra hesaplanan $\hat{x}_j(k)$ değeri kullanılarak

$$\lambda^i \hat{x}_j(k) = \overline{\lambda}^i \hat{x}_j(k) + \mathrm{q}^i \hat{x}_j(k) \quad , \quad i = 0,1,...,n_j \quad , \quad j = 1,2,...,r \tag{3.115}$$

değerleri hesaplanır ve yardımcı modelin durum vektörü (3.110) eşitliğindeki gibi oluşturulabilir. Tekrarlamalı GSYD algoritmasında hesaplanan yardımcı değişkenler kullanıldığında j. çıkışa ait $R_j(k)$ korelasyon matrisi ve $p_j(k)$ korelasyon vektörü giriş-çıkış veri örnekleri alındıkça iteratif olarak aşağıdaki gibi güncellenebilir.

$$R_j(k) = \lambda R_j(k) + z_j(k)\varphi_j^T(k) \quad , \quad p_j(k) = \lambda p_j(k) + z_j(k) y_j(k) \tag{3.116}$$

$R_j(k)$ matrisi ve $p_j(k)$ vektörü iteratif olarak güncellendikten sonra parametre tahminleri bir adımlık GS iterasyonu kullanılarak (3.108) eşitliğindeki gibi güncellenebilir. Burada sırasıyla (3.116) ve (3.108) eşitlikleriyle verilen çok-değişkenli GSYD algoritması çok-girişli çok-çıkışlı sistem parametrelerini yansız olarak tahmin etmek için kullanılabilir.

Parametre tahminlerini tekrarlamalı veya çevrim-içi olarak güncellemek için kullanılan tekrarlamalı GS algoritmasını $j.$ çıkışa ait parametre vektörü için vektörel formda aşağıdaki gibi yazabiliriz.

$$\hat{\theta}_j(k+1) = -(R_{jL}(k) + R_{jD}(k))^{-1} R_{jU}(k)\hat{\theta}_j(k) + (R_{jL}(k) + R_{jD}(k))^{-1} p_j(k) \tag{3.117}$$

Burada $R_{jL}(k)$, $R_{jD}(k)$ ve $R_{jU}(k)$ sırasıyla $R_j(k)$ korelasyon matrisinin alt üçgen, köşegen ve üst üçgen kısımlarını içeren kare matrisleri göstermektedir. Yakınsama analizi tek-girişli tek-çıkışlı GSYD algoritmasına benzer şekilde yapıldığında aşağıdaki sonuç elde edilir.

$$E[\hat{\theta}_j(k)] = \theta_{jopt} + (R_{jL} + R_{jD})R_j^{-1}(R_{jL} + R_{jD})^{-1} E[\varphi_j(k) v_j(k)] \tag{3.118}$$

Elde edilen bu sonuç tekrarlamalı GS algoritması ile elde edilen $j.$ çıkışa ait parametre tahminlerinin optimum değerine yakınsaması için gerekli şartları göstermektedir. Bunun için öncelikle korelasyon matrisinin tersi alınabilmelidir. Bunun için korelasyon matrisi pozitif tanımlı olmalıdır. Sistemin sürekli olarak uyarılması durumunda bu şart sağlanmaktadır. Korelasyon matrisinin pozitif tanımlı olması durumunda tekrarlamalı GS iterasyonlarının yakınsaması sağlanmış olmaktadır (Hatun ve Koçal 2005). Pratik uygulamalarda başlangıçta korelasyon matrisinin pozitif tanımlılığını garantilemek amacıyla $R(0) = \delta.I$ olarak alınabilir, burada I uygun boyutlu birim matris olup

$\delta = 0.1, 0.01, 0.001$ gibi küçük pozitif değerler alabilir. Algoritmanın kararlılığı sağlandıktan sonra elde edilen parametre tahminlerinin optimum değerine yakınsaması için gerekli olan diğer şart ise giriş-çıkış örnekleri alındıkça $E[\varphi_j(k)v_j(k)]$ vektörünün sıfır vektöre yakınsamasıdır. Bu vektörü aşağıdaki gibi inceleyebiliriz.

$$E[\varphi_j(k)v_j(k)] = E\left\{\left[-\lambda y_j(k) \ \cdots -\lambda^{n_j} y_j(k) \ \lambda u_1(k) \ \cdots \ \lambda^{n_{j1}} u_1(k) \ \cdots \ \lambda u_q(k) \ \cdots \ \lambda^{n_{jq}} u_q(k)\right]^T v_j(k)\right\} \tag{3.119}$$

Burada ölçme gürültüsünün, sistemin giriş işaretleriyle $E[u_j(k)v_j(k)] = 0$ şeklinde ilişkisiz olduğu göz önüne alındığında $E[\lambda^i u_j(k)v_j(k)] = 0$ şeklinde filtrelenmiş giriş işaretleriyle de ilişkisiz olduğu söylenebilir. Fakat filtreleme işleminden dolayı çıkış işareti ölçümlerine karışan ölçme gürültüsü beyaz gürültü olsa bile (3.119) eşitliğindeki vektörün çıkış işaretine bağlı $E[\lambda^i y_j(k)v_j(k)]$ elemanları sıfır olmayacaktır. Bu durumda bu vektör

$$E[\varphi_j(k)v_j(k)] = \left[E[\lambda y_j(k)v_j(k)] \ \cdots \ E[\lambda^{n_j} y_j(k)v_j(k)] \ 0 \ \cdots \ 0 \ \cdots \ 0 \ \cdots \ 0\right]^T \tag{3.120}$$

olarak yazılabilir. Bu sonuca göre tekrarlamalı GS algoritmasıyla filtrelenmiş verilerin kullanılması durumunda ölçme gürültüsü beyaz gürültü olsa bile filtreleme işleminden dolayı yansız parametre tahminleri elde edilememektedir. Yansız parametre tahminlerini elde edebilmek için yardımcı değişkenlerden faydalanılmasıyla elde edilen çok-değişkenli GSYD algoritmasının yakınsama analizi de tek-girişli tek-çıkışlı GSYD algoritmasına benzer şekilde yapıldığında

$$E[\hat{\theta}_j(k)] = \theta_{jopt} + (R_{jL} + R_{jD})R_j^{-1}(R_{jL} + R_{jD})^{-1} E[z_j(k)v_j(k)] \tag{3.121}$$

sonucu elde edilir. Bu sonuca göre yine algoritmanın kararlılığının sağlanabilmesi için korelasyon matrisinin tersi alınabilir olmalıdır. Burada yardımcı değişkenleri elde etmek için sistemle aynı yapıya sahip bir uyarlamalı model kullanıldığında bu şart çok rahat bir şekilde sağlanabilmektedir (Söderström ve Stoica 1981, 1983, 1989, 2002, Ljung 1999). Ayrıca, tekrarlamalı GSYD algoritmasında gürültü ile ilişkisiz değişkenler kullanıldığı için giriş-çıkış örnekleri alındıkça zamanla $E[\lambda^i \hat{x}_j(k) v_j(k)]$ şeklinde ilişkisiz olmakta ve $E[z_j(k) v_j(k)]$ vektörü

$$\begin{aligned} E[z_j(k)v_j(k)] &= E\left\{ \left[-\lambda \hat{x}_j(k) \ \cdots -\lambda^{n_j} \hat{x}_j(k) \ \lambda u_1(k) \ \cdots \ \lambda^{m_{j1}} u_1(k) \ \cdots \ \lambda u_q(k) \ \cdots \ \lambda^{m_{jq}} u_q(k) \right]^T v_j(k) \right\} \\ &= \left[0 \cdots 0 \ 0 \cdots 0 \cdots 0 \cdots 0 \right]^T \end{aligned} \tag{3.122}$$

şeklinde sıfır vektöre yakınsamaktadır. Bu sonuca göre tekrarlamalı çok-değişkenli GSYD algoritması kullanıldığında (3.121) eşitliğine göre algoritmanın kararlılığı sağlandığı sürece, çıkış işaretine karışan ölçme gürültüsünün renkli gürültü olması durumunda bile doğru parametre tahminlerinin elde edilebildiği görülmektedir.

3.5. Sistem Modellemede Kullanılan Zaman Serileri Modellerinin Tekrarlamalı Gauss-Seidel Algoritması ile Tahmin Edilmesi

Sistem parametrelerinin tahmin edilmesinde ve bazı uygulamalarında kullanılan başlıca parametrik zaman serileri model yapılarını aşağıdaki gibi sıralayabiliriz (Ljung ve Söderström 1983, Young 1984, Söderström ve Stoica 1989, Ljung 1999, 2008):

- ARX (**A**uto-**R**egressive model with e**X**ogeneous control input) Model
- ARMAX (**A**uto-**R**egressive model with **M**oving-**A**verage noise and e**X**ogeneous control input) Model
- ARARX (**A**uto-**R**egressive model with **A**uto-**R**egressive noise and e**X**ogeneous control input) Model
- ARARMAX (**A**uto-**R**egressive model with **A**uto-**R**egressive-**M**oving-**A**verage noise and e**X**ogeneous control input) Model
- Çıkış Hatası (Output Error) Modeli
- Box-Jenkins Model

3.5.1. ARX model parametrelerinin tahmin edilmesi

Sistem modellemede kullanılan ARX model yapısının fark denklemi

$$A(z^{-1})y(k) = B(z^{-1})u(k) + e(k) \tag{3.123}$$

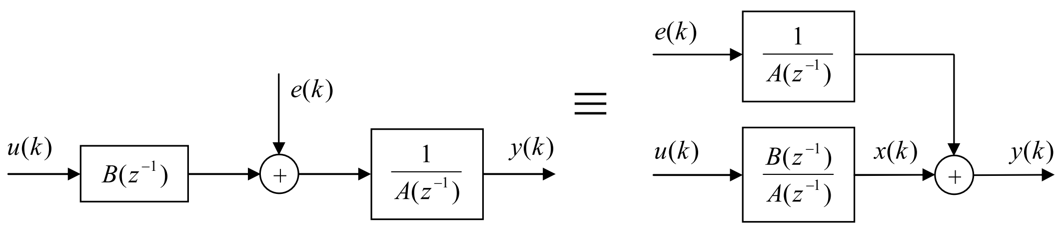
olarak verilebilir. Burada kullanılan polinomlar z^{-1} birim gecikme operatörünün polinomu cinsinden aşağıdaki gibi yazılmaktadır.

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n} \\ B(z^{-1}) &= b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n} \end{aligned} \tag{3.124}$$

Burada n sistemin derecesidir. Bu durumda ARX modelin transfer fonksiyonu gösterimi

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})}u(k) + \frac{1}{A(z^{-1})}e(k) \tag{3.125}$$

olarak yazılabilir. Burada $u(k)$ ve $y(k)$ sırasıyla sistemin kontrol girişi ve çıkış işaretidir, $e(k)$ ise sıfır ortalamalı, varyansı σ_e^2 olan normal dağılıma sahip ilişkisiz rastgele değişen bozucu giriş işaretidir. ARX modelin blok diyagramı Şekil 3.4'teki gibi verilebilir.



Şekil 3.4. ARX modelin eşdeğer blok diyagramı.

ARX modelin parametrelerinin tahmin edilmesi durumunda kullanılan veri vektörü ve parametre vektörü aşağıdaki gibidir.

$$\varphi^T(k) = \begin{bmatrix} -y(k-1) & \cdots & -y(k-n) & u(k) & \cdots & u(k-n) \end{bmatrix} \tag{3.126}$$

$$\theta^T = \begin{bmatrix} a_1 & \cdots & a_n & b_0 & \cdots & b_n \end{bmatrix} \tag{3.127}$$

şeklinde olup, tahmin hatası

$$\varepsilon(k) = y(k) - \varphi^T(k)\hat{\theta}(k-1) \tag{3.128}$$

şeklinde hesaplanmaktadır ve aşağıdaki karesel ortalama hata fonksiyonu iteratif olarak minimum yapılmaya çalışılmaktadır.

$$V_N(\theta) = \frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k) = \frac{1}{N}\sum_{k=1}^{N}\left(y(k) - \varphi^T(k)\theta\right)^2 \tag{3.129}$$

3.5.2. ARMAX model parametrelerinin tahmin edilmesi

Sistem modellemede kullanılan ARMAX model yapısının fark denklemi

$$A(z^{-1})y(k) = B(z^{-1})u(k) + C(z^{-1})e(k) \tag{3.130}$$

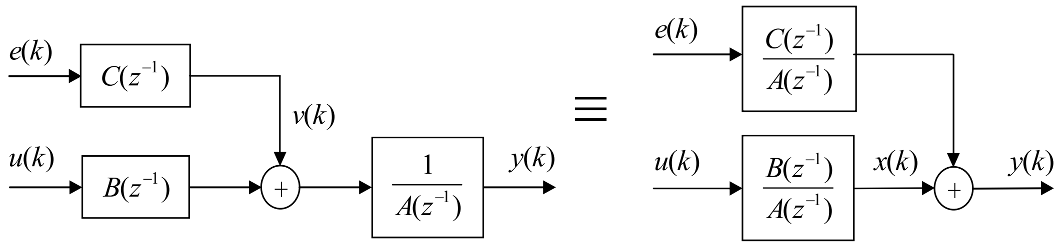
olarak verilebilir. Burada kullanılan $A(z^{-1})$ ve $B(z^{-1})$ polinomları (3.124) eşitliğindeki gibi tanımlanmış olup $C(z^{-1})$ polinomu aşağıdaki gibi tanımlanmaktadır.

$$C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \cdots + c_n z^{-n} \tag{3.131}$$

Bu durumda ARMAX modelin transfer fonksiyonu gösterimi

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})}u(k) + \frac{C(z^{-1})}{A(z^{-1})}e(k) \tag{3.132}$$

şeklinde olup eşdeğer blok diyagramı Şekil 3.5'teki gibi verilebilir.



Şekil 3.5. ARMAX modelin eşdeğer blok diyagramı.

ARMAX modelin parametrelerinin tahmin edilmesi durumunda kullanılan veri vektörü ve parametre vektörü

$$\varphi^T(k) = \left[-y(k-1) \quad \cdots \quad -y(k-n) \quad u(k) \quad \cdots \quad u(k-n) \quad \varepsilon(k-1) \quad \cdots \quad \varepsilon(k-n)\right] \tag{3.133}$$

$$\theta^T = \left[a_1 \quad \cdots \quad a_n \quad b_0 \quad \cdots \quad b_n \quad c_1 \quad \cdots \quad c_n\right] \tag{3.134}$$

şeklinde olup, tahmin hatası

$$\varepsilon(k) = y(k) - \varphi^T(k)\hat{\theta}(k-1) \tag{3.135}$$

şeklinde hesaplanmaktadır ve aşağıdaki karesel ortalama hata fonksiyonu iteratif olarak minimum yapılmaya çalışılmaktadır.

$$V_N(\theta) = \frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k) = \frac{1}{N}\sum_{k=1}^{N}\left(y(k) - \varphi^T(k)\theta\right)^2 \tag{3.136}$$

3.5.3. ARARX model parametrelerinin tahmin edilmesi

Sistem modellemede kullanılan ARARX model yapısı

$$A(z^{-1})y(k) = B(z^{-1})u(k) + \frac{1}{D(z^{-1})}e(k) \tag{3.137}$$

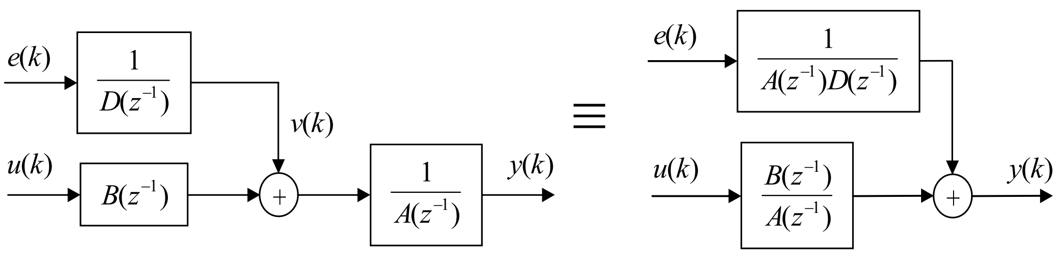
olarak verilebilir. Burada kullanılan $A(z^{-1})$ ve $B(z^{-1})$ polinomları (3.124) eşitliğindeki gibi tanımlanmış olup $D(z^{-1})$ polinomu aşağıdaki gibi tanımlanmaktadır.

$$D(z^{-1}) = 1 + d_1 z^{-1} + d_2 z^{-2} + \cdots + d_n z^{-n} \tag{3.138}$$

Bu durumda ARARX modelin transfer fonksiyonu gösterimi

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})}u(k) + \frac{1}{A(z^{-1})D(z^{-1})}e(k) \tag{3.139}$$

şeklinde olup eşdeğer blok diyagramı Şekil 3.6'daki gibi verilebilir.



Şekil 3.6. ARARX modelin eşdeğer blok diyagramı.

ARARX modelin parametrelerinin tahmin edilmesi durumunda kullanılan veri vektörü ve parametre vektörü

$$\varphi^T(k) = \left[-y(k-1) \quad \cdots \quad -y(k-n) \quad u(k) \quad \cdots \quad u(k-n) \quad -v(k-1) \quad \cdots \quad -v(k-n)\right] \tag{3.140}$$

$$\theta^T = \left[a_1 \quad \cdots \quad a_n \quad b_0 \quad \cdots \quad b_n \quad d_1 \quad \cdots \quad d_n\right] \tag{3.141}$$

şeklinde olup, bozucu giriş

$$v(k) = A(z^{-1})y(k) - B(z^{-1})u(k) \tag{3.142}$$

şeklinde hesaplanmakta, tahmin hatası ise

$$\varepsilon(k) = y(k) - \varphi^T(k)\hat{\theta}(k-1) \tag{3.143}$$

şeklinde hesaplanmaktadır ve aşağıdaki karesel ortalama hata fonksiyonu iteratif olarak minimum yapılmaya çalışılmaktadır.

$$V_N(\theta) = \frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k) = \frac{1}{N}\sum_{k=1}^{N}\left(y(k) - \varphi^T(k)\theta\right)^2 \tag{3.144}$$

3.5.4. ARARMAX model parametrelerinin tahmin edilmesi

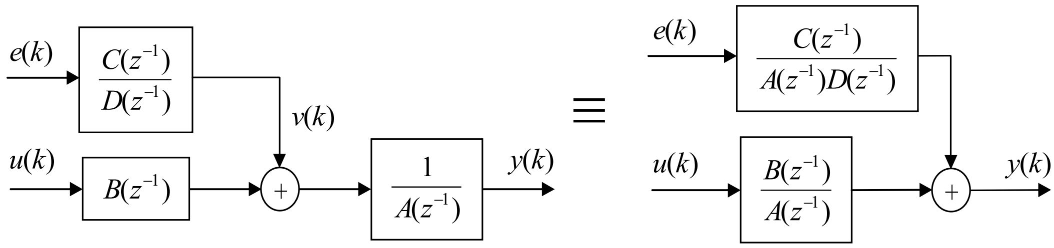
Sistem modellemede kullanılan ARARMAX model yapısı

$$A(z^{-1})y(k) = B(z^{-1})u(k) + \frac{C(z^{-1})}{D(z^{-1})}e(k) \tag{3.145}$$

olarak verilebilir. Burada kullanılan $A(z^{-1})$, $B(z^{-1}), C(z^{-1})$ ve $D(z^{-1})$ polinomları sırasıyla (3.124), (3.131) ve (3.138) eşitliklerindeki gibi tanımlanmıştır. Bu durumda ARARMAX modelin transfer fonksiyonu gösterimi

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})}u(k) + \frac{C(z^{-1})}{A(z^{-1})D(z^{-1})}e(k) \tag{3.146}$$

şeklinde olup, eşdeğer blok diyagramı Şekil 3.7'deki gibi verilebilir.



Şekil 3.7. ARARMAX modelin eşdeğer blok diyagramı.

ARARMAX modelin parametrelerinin tahmin edilmesi durumunda kullanılan veri vektörü ve parametre vektörü

$$\begin{aligned}
&\varphi^T(k)=\begin{bmatrix}-y(k-1) & \cdots & -y(k-n) & u(k) & \cdots & u(k-n) & \varepsilon(k-1) & \cdots & \varepsilon(k-n) & -v(k-1) & \cdots & -v(k-n)\end{bmatrix}\\
&\theta^T=\begin{bmatrix}a_1 & \cdots & a_n & b_0 & \cdots & b_n & c_1 & \cdots & c_n & d_1 & \cdots & d_n\end{bmatrix}
\end{aligned} \tag{3.147}$$

şeklinde olup, $v(k)$ ve $\varepsilon(k)$ değerleri (3.142) ve (3.143) eşitliklerindeki gibi hesaplanmaktadır ve aşağıdaki karesel ortalama hata fonksiyonu iteratif olarak minimum yapılmaya çalışılmaktadır.

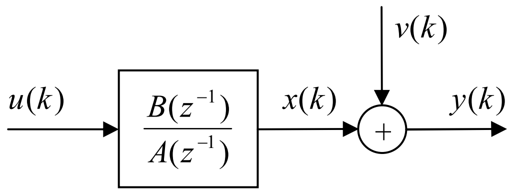
$$V_N(\theta)=\frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k)=\frac{1}{N}\sum_{k=1}^{N}\left(y(k)-\varphi^T(k)\theta\right)^2 \tag{3.148}$$

3.5.5. Çıkış hatası modelinin parametrelerinin tahmin edilmesi

Sistem modellemede kullanılan çıkış hatası modelinin transfer fonksiyonu gösterimi

$$y(k)=\frac{B(z^{-1})}{A(z^{-1})}u(k)+v(k) \tag{3.149}$$

olarak yazılabilir. Burada kullanılan $A(z^{-1})$ ve $B(z^{-1})$, polinomları (3.124) eşitliğindeki gibi tanımlanmıştır. Bu durumda çıkış hatası modelinin eşdeğer blok diyagramı Şekil 3.8'deki gibi verilebilir.



Şekil 3.8. Çıkış hatası modelinin eşdeğer blok diyagramı.

Çıkış hatası modelinin parametrelerinin tahmin edilmesi durumunda kullanılan veri vektörü ve parametre vektörü

$$z^T(k) = \begin{bmatrix} -x(k-1) & \cdots & -x(k-n) & u(k) & \cdots & u(k-n) \end{bmatrix} \tag{3.150}$$

$$\theta^T = \begin{bmatrix} a_1 & \cdots & a_n & b_0 & \cdots & b_n \end{bmatrix} \tag{3.151}$$

şeklinde olup, $x(k)$ değerleri aşağıdaki gibi hesaplanmaktadır.

$$x(k) = z^T(k)\,\hat{\theta}(k) \tag{3.152}$$

Bu durumda tahmin hatası

$$\varepsilon(k) = y(k) - z^T(k)\hat{\theta}(k-1) \tag{2.153}$$

şeklinde hesaplanmakta ve aşağıdaki ortalama karesel hata fonksiyonu iteratif olarak minimum yapılmaya çalışılmaktadır.

$$V_N(\theta) = \frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k) = \frac{1}{N}\sum_{k=1}^{N}\left(y(k) - z^T(k)\theta\right)^2 \tag{3.154}$$

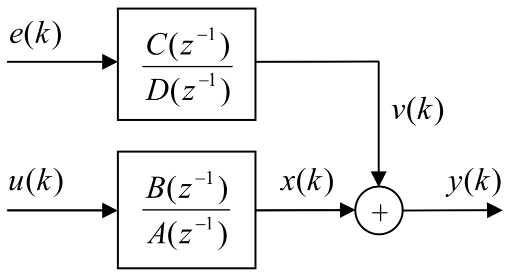
Çıkış hatası modeli uyarlamalı işaret işleme literatüründe uyarlamalı IIR (Infinite Impulse Response) filtre katsayılarının güncellenmesinde yaygın olarak kullanılmaktadır (Treichler ve ark. 1987, Netto ve ark. 1995, Regalia 1995, Diniz 1997, Farhang-Boroujeny 1998, Haykin 2002).

3.5.6. Box-Jenkins model parametrelerinin tahmin edilmesi

Sistem modellemede kullanılan Box-Jenkins modelin transfer fonksiyonu gösterimi

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})}u(k) + \frac{C(z^{-1})}{D(z^{-1})}e(k) \tag{3.155}$$

olarak yazılabilir. Burada kullanılan $A(z^{-1})$, $B(z^{-1})$, $C(z^{-1})$ ve $D(z^{-1})$ polinomları sırasıyla (3.124), (3.131) ve (3.138) eşitliklerindeki gibi tanımlanmıştır. Bu durumda Box-Jenkins modelin eşdeğer blok diyagramı Şekil 3.9'daki gibi verilebilir.



Şekil 3.9. Box-Jenkins modelin eşdeğer blok diyagramı.

Box-Jenkins model parametrelerinin tahmin edilmesi durumunda kullanılan veri vektörü ve parametre vektörü

$$\varphi^T(k) = [-x(k-1) \ \cdots \ -x(k-n) \ \ u(k) \ \cdots \ u(k-n) \ \ \varepsilon(k-1) \ \cdots \ \varepsilon(k-n) \ \ -v(k-1) \ \cdots \ -v(k-n)]$$

$$\theta^T = [a_1 \ \cdots \ a_n \ \ b_0 \ \cdots \ b_n \ \ c_1 \ \cdots \ c_n \ \ d_1 \ \cdots \ d_n] \tag{3.156}$$

şeklinde olup, $x(k)$ değerleri (3.152) eşitliğindeki gibi hesaplanmaktadır, $v(k)$ değerleri

$$v(k) = y(k) - x(k) \tag{3.157}$$

şeklinde çıkış hatası modelinin hata hesabındaki gibi, $\varepsilon(k)$ değerleri ise

$$\varepsilon(k) = y(k) - \varphi^T(k)\hat{\theta}(k-1) \tag{3.158}$$

şeklinde hesaplanmaktadır. Bu durumda aşağıdaki karesel ortalama hata fonksiyonu iteratif olarak minimum yapılmaya çalışılmaktadır.

$$V_N(\theta) = \frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k) = \frac{1}{N}\sum_{k=1}^{N}\left(y(k) - \varphi^T(k)\theta\right)^2 \tag{3.159}$$

3.5.7. Tekrarlamalı Gauss-Seidel algoritmasının deterministik yakınsama analizi

Korelasyon matrisinin ve korelasyon vektörünün N adet veri grubu kullanılarak tahmin edildiğini ve parametre tahminlerinin klasik GS iterasyonlarıyla hesaplandığını göz önüne alalım. Bilindiği gibi R korelasyon matrisi simetrik ve pozitif tanımlı olan bir matristir (Ljung ve Söderström 1983, Söderström ve Stoica 1989, Ljung 1999, Haykin 2002). Doğrusal denklem takımını oluşturan kare matrisin simetrik ve pozitif tanımlı olması durumunda, GS algoritmasının herhangi bir başlangıç değeri için denklem takımını sağlayan çözüm değerine iteratif olarak yakınsadığı matematiksel olarak Golub ve Van Loan (1996) tarafından gösterilmiştir. Buradan hareketle GS algoritmasının deterministik olarak normal denklemin optimum çözümüne yakınsadığı aşağıdaki teorem ile gösterilebilir.

Teorem: $R \in \mathrm{R}^{M\times M}$ korelasyon matrisi simetrik ve pozitif tanımlı olduğundan dolayı, GS iterasyonu herhangi bir $\hat{\theta}(0)$ başlangıç değeri için normal denklemin $\theta_{opt} = R^{-1}p$ ile verilen optimum çözümüne yakınsar.

İspat: Burada öncelikle yeterli miktarda veri kullanılması durumunda

$$E[R(N)] \cong R, \quad E[R_L(N)] \cong R_L, \quad E[R_D(N)] \cong R_D, \quad E[p(N)] \cong p \tag{3.160}$$

yaklaşıklıklarını yapabiliriz. R korelasyon matrisini $R = R_L + R_D + R_L^T$ şeklinde yazdığımızda, GS algoritmasındaki $T = -(R_D + R_L)^{-1} R_L^T$ iterasyon matrisinin bütün özdeğerleri birim daire içinde ise algoritma yakınsar. Bunun için spektral yarıçap olarak bilinen ve T matrisinin en büyük özdeğeri olarak tanımlanan $\max(|\lambda_T|) < 1$ olmalıdır. R_D matrisi pozitif tanımlı olduğundan dolayı

$$T_1 = R_D^{1/2}\, T\, R_D^{-1/2} \tag{3.161}$$

olarak tanımlanan T_1 matrisi T ile aynı özdeğerlere sahiptir. Bu durumda $\max(|\lambda_{T1}|) < 1$ olduğunu göstermek, GS algoritmasının pozitif tanımlı simetrik matrisler için yakınsadığını göstermek için yeterlidir. Bu amaçla

$$R_{L1} = R_D^{-1/2}\, T\, R_D^{-1/2} \tag{3.162}$$

tanımlamasını yaptığımızda T_1 matrisini

$$T_1 = R_D^{1/2}\, T\, R_D^{-1/2} = -(I + R_{L1})^{-1} R_{L1}^T \tag{3.163}$$

olarak yazabiliriz. T ile T_1 matrisleri aynı özdeğerlere sahip olduğundan dolayı T_1 matrisinin özdeğerlerini bulmak için

$$-(I + R_{L1})^{-1} R_{L1}^T\, x = \lambda\, x \tag{3.164}$$

eşitliğini yazabiliriz. Bu eşitliğin her iki tarafını soldan $(I + R_{L1})$ ile çarptığımızda

$$- R_{L1}^{T}\, x = \lambda (\, I + R_{L1})x \tag{3.165}$$

eşitliğini yazabiliriz. Bu eşitliğin her iki tarafını soldan x^H ile çarparak $x^H x = 1$ özelliğini kullandığımızda

$$- x^H R_{L1}^{T}\, x = \lambda \left(1 + x^H R_{L1} x\right) \tag{3.166}$$

eşitliğini elde ederiz. Buradan λ 'yı

$$\lambda = \frac{- x^H R_{L1}^{T}\, x}{\left(1 + x^H R_{L1} x\right)} \tag{3.167}$$

olarak elde ederiz. Burada $x^H R_{L1}\, x = a$ şeklinde skaler bir sayıdır. Burada

$$a^H = \left(x^H R_{L1}\, x\right)^H = x^H R_{L1}\, x = a \tag{3.168}$$

özelliğini kullanırsak

$$\lambda^2 = \frac{a^2}{1 + 2a + a^2} \tag{3.169}$$

sonucu elde edilir. Burada $|\, \lambda \,| < 1$ olması için

$$1 + 2a = 1 + x^H R_{L1} x + x^H R_{L1}^{T} x > 0 \tag{3.170}$$

olması gerekir. Burada R_{L1} pozitif tanımlı bir matris olduğundan dolayı bu şart her zaman sağlanır. Bu şart her zaman geçerli olduğundan dolayı $|\, \lambda_k \,| < 1$, $(\, k = 1,2,\ldots, M\,)$ şartı da her zaman geçerlidir (Hatun ve Koçal 2005).

3.5.8. Tekrarlamalı Gauss-Seidel algoritmasının stokastik yakınsama analizi

Bu kısımda sırasıyla parametre tahminleri vektörünün beklenen değerinin, parametre hata vektörünün beklenen değerinin, parametre hata vektörünün korelasyon matrisinin ve tahmin hatasının beklenen değeri ile varyansının kalıcı durumdaki yakınsama analizleri yapılmıştır.

3.5.8.1. Parametre tahminleri vektörünün beklenen değerinin yakınsama analizi

Parametre tahminlerini çevrim-içi yolla güncellemek için kullanılan tekrarlamalı GS algoritmasını aşağıdaki gibi yazabiliriz.

$$\hat{\theta}(k+1) = -(R_L(k) + R_D(k))^{-1} R_U(k)\hat{\theta}(k) + (R_L(k) + R_D(k))^{-1} p(k) \qquad (3.171)$$

Burada k anındaki $R(k)$ korelasyon matrisi ve $p(k)$ korelasyon vektörü tahminlerini, veri örnekleri alındıkça güncellendiğini göz önüne alarak aşağıdaki gibi yazabiliriz.

$$R(k) \cong \frac{1}{k}\sum_{n=1}^{k} \varphi(n)\varphi^T(n) \ , \quad p(k) \cong \frac{1}{k}\sum_{n=1}^{k} \varphi(n)y(n) \qquad (3.172)$$

Burada sistemin çıkış işaretini doğrusal bağlanımlı biçiminde aşağıdaki gibi yazabiliriz.

$$y(k) = \varphi^T(k)\theta_{opt} + v(k) \qquad (3.173)$$

Burada $v(k)$ zaman serileri modellerindeki bozucu girişi veya çıkış işareti ölçümlerine karışan ölçme gürültüsünü temsil etmektedir. Bu eşitliği $p(k)$ korelasyon vektörü tahminini gösteren denklemde yerine yazıp aşağıdaki işlemleri yaptığımızda

$$
\begin{aligned}
p(k) &= \frac{1}{k}\sum_{n=1}^{k}\varphi(n)y(n) \\
&= \frac{1}{k}\sum_{n=1}^{k}\varphi(n)\left[\varphi^{T}(n)\theta_{opt} + v(n)\right] \\
&= \frac{1}{k}\sum_{n=1}^{k}\varphi(n)\varphi^{T}(n)\theta_{opt} + \frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n) \\
&= R(k)\theta_{opt} + \frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)
\end{aligned} \tag{3.174}
$$

sonucunu elde ederiz. Bu sonucu tekrarlamalı GS algoritmasında yerine yazdığımızda

$$
\hat{\theta}(k+1) = -(R_L(k)+R_D(k))^{-1}R_U(k)\hat{\theta}(k) + (R_L(k)+R_D(k))^{-1}\left[R(k)\theta_{opt} + \frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right] \tag{3.175}
$$

eşitliğini elde ederiz. Korelasyon matrisini alt üçgen, köşegen ve üst üçgen kısımlarını içeren kare matrislerin toplamı şeklinde $R(k) = R_L(k) + R_D(k) + R_U(k)$ olarak yerine yazdığımızda

$$
\begin{aligned}
\hat{\theta}(k+1) &= -(R_L(k)+R_D(k))^{-1}R_U(k)\hat{\theta}(k) \\
&\quad + (R_L(k)+R_D(k))^{-1}\left[\left(R_L(k)+R_D(k)+R_U(k)\right)\theta_{opt} + \frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right] \\
\hat{\theta}(k+1) &= -(R_L(k)+R_D(k))^{-1}R_U(k)\hat{\theta}(k) \\
&\quad + \left[I + (R_L(k)+R_D(k))^{-1}R_U(k)\right]\theta_{opt} + (R_L(k)+R_D(k))^{-1}\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right)
\end{aligned} \tag{3.176}
$$

eşitliğini elde ederiz. Burada

$$
\begin{aligned}
&E[\hat{\theta}(k+1)] = E[\hat{\theta}(k)] \quad , \quad E[\varphi(k)v(k)] = E\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right) \\
&E[(R_L(k)+R_D(k))^{-1}] = (R_L + R_D)^{-1}
\end{aligned} \tag{3.177}
$$

$$E[I+(R_L(k)+R_D(k))^{-1}R_U(k)]=I+(R_L+R_D)^{-1}R_U$$

kabullerini de göz önüne alarak en son elde ettiğimiz iteratif denklemin her iki tarafının beklenen değerini aldığımızda

$$[I+(R_L+R_D)^{-1}R_U]E[\hat{\theta}(k)]=[I+(R_L+R_D)^{-1}R_U]\theta_{opt}+(R_L+R_D)^{-1}E[\varphi(k)v(k)] \quad (3.178)$$

eşitliğini elde ederiz. Bu eşitliğin her iki tarafını soldan $[I+(R_L+R_D)^{-1}R_U]^{-1}$ matrisi ile çarptığımızda

$$E[\hat{\theta}(k)]=\theta_{opt}+\left[\left(I+(R_L+R_D)^{-1}R_U\right)(R_L+R_D)\right]^{-1}E[\varphi(k)v(k)] \quad (3.179)$$

sonucunu elde ederiz. Burada $I-(R_L+R_D)^{-1}R=-(R_L+R_D)^{-1}R_U$ matris eşitliğini kullanarak elde edilen

$$I+(R_L+R_D)^{-1}R_U=(R_L+R_D)^{-1}R \quad (3.180)$$

eşitliğini yukarıdaki son denklemde yerine yazdığımızda aşağıdaki sonucu elde ederiz.

$$E[\hat{\theta}(k)]=\theta_{opt}+(R_L+R_D)R^{-1}(R_L+R_D)^{-1}E[\varphi(k)v(k)] \quad (3.181)$$

RLS algoritmasıyla yapılan parametre tahminlerinin beklenen değeri ise

$$E[\hat{\theta}(k)]=\theta_{opt}+R^{-1}E[\varphi(k)v(k)] \quad (3.182)$$

olarak elde edilmektedir (Sinha ve Kuszta 1983, Söderström ve Stoica 1989, Ljung 1999). Tekrarlamalı GS algoritması için bulunan (3.181) denklemi parametre tahminlerinin beklenen değerinin yakınsamasını göstermektedir. Elde edilen parametre tahminlerinin optimum değerine yakınsayabilmesi için öncelikle eşitliğin sağ tarafındaki ikinci terimin sıfır olması gerekir. Bu ikinci terimdeki

$(R_L + R_D)R^{-1}(R_L + R_D)^{-1}$ çarpanı sıfır değerini alamaz, çünkü parametre tahminlerinin optimum değerine yakınsayabilmesi için öncelikle $R^{-1}(R_L + R_D)^{-1}$ çarpanındaki R korelasyon matrisinin ve yine bu matrisin elemanlarından oluşan $(R_L + R_D)$ matrisinin tersinin alınabilmesi gerekir. Bu da algoritmanın kararlılığıyla ilgili bir durumdur. Daha önce deterministik yakınsama analizinde gösterildiği gibi R korelasyon matrisinin simetrik ve pozitif tanımlı olması durumunda bu şart sağlanmaktadır. Sistemin $u(k)$ kontrol sinyali ile sürekli olarak uyarılması durumunda bu şart kolaylıkla sağlanmaktadır (Sinha ve Kuszta 1983, Söderström ve Stoica 1989, Ljung 1999). Elde edilen parametre tahminlerinin optimum değerine yakınsayabilmesi için diğer şart, eşitliğin sağ tarafındaki ikinci terimde bulunan $E[\varphi(k)v(k)]$ karşı-ilişki (cross-correlation) katsayılarını içeren vektörün bütün elemanlarının sıfır olmasıdır. Bu şart kullanılan modeller için şu şekilde incelenebilir: Elde edilen sonuca göre, ARX model parametre tahminlerinin optimum değerine yakınsayabilmesi için bozucu girişin beyaz gürültü biçiminde olması gerekir. Yardımcı değişkenlerden yararlanılması durumunda ise doğru parametre tahminleri elde edilir. Çıkış hatası modeli için de aynı şekilde her durumda doğru parametre tahminleri elde edilir. ARMAX, ARARX, ARARMAX ve Box-Jenkins modellerinde renkli gürültü biçiminde değişen bozucu giriş beyaz gürültüden elde edilmiş gibi modellendiği için hesaplanan parametre tahminleri doğru değerine yakınsayacaktır. RLS algoritması için geçerli olan (3.182) denklemi incelendiğinde ise yine aynı uyarım ve ilişkisizlik şartının geçerli olduğu görülür.

3.5.8.2. Parametre tahmin hatası vektörünün beklenen değerinin yakınsama analizi

Parametre tahminlerini çevrim-içi yolla güncellemek için kullanılan tekrarlamalı GS algoritmasını tekrar aşağıdaki gibi yazabiliriz.

$$\hat{\theta}(k+1) = -(R_L(k) + R_D(k))^{-1} R_U(k)\hat{\theta}(k) + (R_L(k) + R_D(k))^{-1} p(k) \qquad (3.183)$$

Parametre tahmin hatası vektörü

$$\tilde{\theta}(k) = \hat{\theta}(k) - \theta_{opt} \quad \rightarrow \quad \hat{\theta}(k) = \tilde{\theta}(k) + \theta_{opt}$$
$$\tilde{\theta}(k+1) = \hat{\theta}(k+1) - \theta_{opt} \quad \rightarrow \quad \hat{\theta}(k+1) = \tilde{\theta}(k+1) + \theta_{opt} \tag{3.184}$$

şeklinde tanımlandığında ve

$$I - (R_L + R_D)^{-1} R = -(R_L + R_D)^{-1} R_U \tag{3.185}$$

matris eşitliğini kullanıldığında tekrarlamalı GS algoritması tahmin hatası vektörü cinsinden aşağıdaki gibi düzenlenebilir.

$$\begin{aligned}\tilde{\theta}(k+1) + \theta_{opt} &= \left[I - (R_L(k) + R_D(k))^{-1} R(k)\right]\left(\tilde{\theta}(k) + \theta_{opt}\right) + \left(R_L(k) + R_D(k)\right)^{-1} p(k) \\ &= \left[I - (R_L(k) + R_D(k))^{-1} R(k)\right]\tilde{\theta}(k) + \left[I - (R_L(k) + R_D(k))^{-1} R(k)\right]\theta_{opt} \\ &\quad + \left(R_L(k) + R_D(k)\right)^{-1} p(k)\end{aligned} \tag{3.186}$$

$$\tilde{\theta}(k+1) + \theta_{opt} = \left[I - (R_L(k) + R_D(k))^{-1} R(k)\right]\tilde{\theta}(k) + \theta_{opt} + \left(R_L(k) + R_D(k)\right)^{-1}\left(p(k) - R(k)\theta_{opt}\right)$$
$$\tilde{\theta}(k+1) = \left[I - (R_L(k) + R_D(k))^{-1} R(k)\right]\tilde{\theta}(k) + \left(R_L(k) + R_D(k)\right)^{-1}\left(p(k) - R(k)\theta_{opt}\right) \tag{3.187}$$

Bu son eşitlikte $p(k)$ korelasyon vektörünün daha önce elde edilen,

$$p(k) = R(k)\theta_{opt} + \frac{1}{k}\sum_{n=1}^{k} \varphi(n) v(n) \tag{3.188}$$

bozucu girişe veya renkli ölçme gürültüsüne bağlı değerini yerine yazdığımızda

$$\begin{aligned}\tilde{\theta}(k+1) &= \left[I - (R_L(k) + R_D(k))^{-1} R(k)\right]\tilde{\theta}(k) \\ &\quad + \left(R_L(k) + R_D(k)\right)^{-1}\left(R(k)\theta_{opt} + \frac{1}{k}\sum_{n=1}^{k} \varphi(n) v(n) - R(k)\theta_{opt}\right)\end{aligned} \tag{3.189}$$

$$\tilde{\theta}(k+1) = \left[I - (R_L(k) + R_D(k))^{-1} R(k)\right]\tilde{\theta}(k) + \left(R_L(k) + R_D(k)\right)^{-1}\left(\frac{1}{k}\sum_{n=1}^{k} \varphi(n) v(n)\right) \tag{3.190}$$

eşitliğini elde ederiz. Burada yine daha önce kullanılan kabullere benzer şekilde

$$E[\widetilde{\theta}(k+1)] = E[\widetilde{\theta}(k)] \quad , \quad E[\varphi(k)v(k)] = E\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right)$$
$$E[(R_L(k)+R_D(k))^{-1}] = (R_L + R_D)^{-1} \qquad (3.191)$$
$$E[I + (R_L(k)+R_D(k))^{-1} R_U(k)] = I + (R_L + R_D)^{-1} R_U$$

kabullerini yaptığımızda

$$E[\widetilde{\theta}(k)] = [I - (R_L + R_D)^{-1} R] E[\widetilde{\theta}(k)] + (R_L + R_D)^{-1} E[\varphi(k)v(k)]$$
$$[(R_L + R_D)^{-1} R] E[\widetilde{\theta}(k)] = (R_L + R_D)^{-1} E[\varphi(k)v(k)]$$
$$E[\widetilde{\theta}(k)] = [(R_L + R_D)^{-1} R]^{-1} (R_L + R_D)^{-1} E[\varphi(k)v(k)]$$
$$= [(R_L + R_D)^{-1} R (R_L + R_D)]^{-1} E[\varphi(k)v(k)]$$
$$E[\widetilde{\theta}(k)] = (R_L + R_D) R^{-1} (R_L + R_D)^{-1} E[\varphi(k)v(k)] \qquad (3.192)$$

eşitliğini elde ederiz. Bu sonuç önceki kısımda bulunan sonuçla tamamen aynıdır. Yani parametre hata vektörünün beklenen değerinin sıfıra yakınsayabilmesi için $E[\varphi(k)v(k)]$ vektörünün sıfır vektörüne yakınsaması gerekir. Örneğin ARX model parametrelerinin tekrarlamalı GS algoritmasıyla tahmin edilmesi durumunda $v(k) = e(k)$ şeklinde ilişkisiz rastgele değişken olmadığı sürece parametre hata vektörünün beklenen değeri limit durumunda aşağıda verilen $\Delta\theta$ "bias" terimine yakınsayacaktır.

$$E[\widetilde{\theta}(k)] \quad \rightarrow \quad \Delta\theta = -(R_L + R_D) R^{-1} (R_L + R_D)^{-1} \begin{bmatrix} r_v(1) & \cdots & r_v(n) & 0 & \cdots & 0 \end{bmatrix}^T \qquad (3.193)$$

Yardımcı değişkenlerden faydalanılması ve çıkış hatası modelinin kullanılması durumunda, ya da bozucu girişin modellendiği ARMAX, ARARX, ARARMAX ve Box-Jenkins model yapılarının kullanılması durumunda parametre tahmin hatası vektörünün beklenen değeri sıfır vektöre yakınsayacaktır. Yani yansız parametre tahminleri elde edilecektir. Bu durumda

$$\widetilde{\theta}(k+1) = \left[I - (R_L(k) + R_D(k))^{-1} R(k)\right]\widetilde{\theta}(k) + \left(R_L(k) + R_D(k)\right)^{-1}\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right) \qquad (3.194)$$

denklemini tekrar ele alalım. Bu denklem, parametre hata vektörüne bağlı olan ve sistem matrisi $[I - (R_L(k) + R_D(k))^{-1} R(k)]$ olan bir stokastik fark denklemidir. Bu denklemin her iki tarafının beklenen değerini aldığımızda ve (3.191) eşitliğiyle verilen kabulleri kullandığımızda

$$E[\widetilde{\theta}(k+1)] = [I - (R_L + R_D)^{-1} R]\, E[\widetilde{\theta}(k)] + (R_L + R_D)^{-1} E[\varphi(k)v(k)] \qquad (3.195)$$

eşitliğini elde ederiz. Yukarıda da bahsedildiği gibi yardımcı değişkenlerin ve OE, ARMAX, ARARX, ARARMAX, Box-Jenkins model yapılarının kullanılması durumunda $E[\varphi(k)v(k)]$ karşı-ilişki vektörü sıfır vektöre yakınsayacağından yukarıdaki eşitliğin sağ tarafındaki ikinci terimi ihmal edebiliriz ve aşağıdaki yaklaşıklığı yapabiliriz.

$$E[\widetilde{\theta}(k+1)] = [I - (R_L + R_D)^{-1} R]\, E[\widetilde{\theta}(k)] \qquad (3.196)$$

Burada $c(k) = E[\widetilde{\theta}(k)]$ tanımlamasını yaparak aynı denklemi

$$c(k+1) = [I - (R_L + R_D)^{-1} R]\, c(k) \qquad (3.197)$$

şeklinde yazabiliriz. Bu fark denkleminin çözümünü başlangıç değerine bağlı olarak

$$c(k) = [I - (R_L + R_D)^{-1} R]^k\, c(0) \qquad (3.198)$$

şeklinde yazabiliriz. Parametre tahminlerinin ortalamasının gerçek değerine yakınsayabilmesi için $E[\varphi(k)v(k)]$ şartından önce, parametre hata vektörünün yakınsamasına ait bu denklemin sıfır vektöre yakınsaması gerekir. Bu durum algoritmanın kararlılığı ile ilgili bir durumdur. Bunun için $[I - (R_L + R_D)^{-1} R]$

matrisinin k. kuvvetinin sıfır matrise yakınsaması gerekir. Bu da $[I-(R_L+R_D)^{-1}R]$ matrisinin bütün özdeğerlerinin birim daire içinde olmasını gerektirir. Daha önce deterministik yakınsama analizinde bahsedildiği gibi R korelasyon matrisinin ve bunun bileşenlerinden oluşan (R_L+R_D) matrisinin simetrik ve pozitif tanımlı olması, tekrarlamalı GS algoritmasındaki $[I-(R_L+R_D)^{-1}R]$ iterasyon matrisinin bütün özdeğerlerinin ve aynı zamanda spektral yarıçap olarak adlandırılan, iterasyon matrisinin en büyük özdeğerinin birim daire içinde olmasını sağlamaktadır. Bunun sonucunda ayrık zaman indisi $k\to\infty$ şeklinde sonsuza yaklaştıkça $[I-(R_L+R_D)^{-1}R]^k\to\mathbf{0}$ şeklinde sıfır elemanlı kare matrise yakınsamaktadır. Burada kararlılık problemine ilave olarak (3.195) eşitliği gereği, $E[\varphi(k)v(k)]$ karşı-ilişki vektörünün de sıfır vektöre yakınsaması durumunda, parametre tahmin hatasının beklenen değeri $E[\widetilde{\theta}(k)]\to\mathbf{0}$ şeklinde sıfır vektöre yakınsamakta ve yansız parametre tahminleri elde edilmektedir.

3.5.8.3. Parametre tahmin hatası vektörünün korelasyon matrisinin yakınsama analizi

Parametre tahmin hatası vektörünün yakınsama analizinde kullandığımız stokastik fark denklemini tekrar aşağıdaki gibi yazabiliriz.

$$\widetilde{\theta}(k+1)=\left[I-(R_L(k)+R_D(k))^{-1}R(k)\right]\widetilde{\theta}(k)+\left(R_L(k)+R_D(k)\right)^{-1}\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right)\tag{3.199}$$

Burada amacımız $\widetilde{\theta}(k)\widetilde{\theta}^T(k)$ matrisinin beklenen değerinin yakınsama analizini yapmaktır. Bu amaçla yukarıdaki stokastik fark denkleminden aşağıdaki denklemi yazabiliriz.

$$\widetilde{\theta}^T(k+1)=\widetilde{\theta}^T(k)\left[I-(R_L(k)+R_D(k))^{-1}R(k)\right]^T+\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right)^T\left(R_L(k)+R_D(k)\right)^{-T}\tag{3.200}$$

Burada üst indis $-T$, ele alınan matrisin tersinin transpozunu göstermektedir. Bu iki eşitliği çarptığımızda aşağıdaki sonucu elde ederiz.

$$\begin{aligned}\tilde{\theta}(k+1)\tilde{\theta}^T(k+1) &= \left[I-(R_L(k)+R_D(k))^{-1}R(k)\right]\tilde{\theta}(k)\tilde{\theta}^T(k)\left[I-(R_L(k)+R_D(k))^{-1}R(k)\right]^T \\ &+\left[I-(R_L(k)+R_D(k))^{-1}R(k)\right]\tilde{\theta}(k)\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right)^T\left(R_L(k)+R_D(k)\right)^{-T} \\ &+\left(R_L(k)+R_D(k)\right)^{-1}\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right)\tilde{\theta}^T(k)\left[I-(R_L(k)+R_D(k))^{-1}R(k)\right]^T \\ &+\left(R_L(k)+R_D(k)\right)^{-1}\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right)\left(\frac{1}{k}\sum_{n=1}^{k}\varphi(n)v(n)\right)^T\left(R_L(k)+R_D(k)\right)^{-T}\end{aligned} \tag{3.201}$$

Bu eşitliğin her iki tarafının beklenen değerini aldığımızda ve daha önce kullandığımız (3.191) eşitliğindeki kabulleri kullandığımızda

$$\begin{aligned}E\left[\tilde{\theta}(k+1)\tilde{\theta}^T(k+1)\right] &= \left[I-(R_L+R_D)^{-1}R\right]E\left[\tilde{\theta}(k)\tilde{\theta}^T(k)\right]\left[I-(R_L+R_D)^{-1}R\right]^T \\ &+\left[I-(R_L+R_D)^{-1}R\right]E\left[\tilde{\theta}(k)\right]E[\varphi(k)v(k)]^T\left(R_L+R_D\right)^{-T} \\ &+\left(R_L+R_D\right)^{-1}E[\varphi(k)v(k)]E\left[\tilde{\theta}^T(k)\right]\left[I-(R_L+R_D)^{-1}R\right]^T \\ &+\left(R_L+R_D\right)^{-1}E\left[\varphi(k)v(k)[\varphi(k)v(k)]^T\right]\left(R_L+R_D\right)^{-T}\end{aligned} \tag{3.202}$$

eşitliğini elde ederiz. Bu eşitlikte, eşitliğin sağ tarafındaki ikinci ve üçüncü terim $E[\varphi(k)v(k)]$ ilişki vektörünü içermektedir. Burada, parametre hata vektörünün yakınsama analizinde de bahsedildiği gibi yardımcı değişkenlerin ve çıkış hatası ARMAX, ARARX, ARARMAX, Box-Jenkins model yapılarının kullanılması durumunda $E[\varphi(k)v(k)]$ karşı-ilişki vektörü sıfır vektöre yakınsayacağından, yukarıdaki eşitliğin sağ tarafındaki ikinci ve üçüncü terimi ihmal edebiliriz ve aşağıdaki yaklaşıklığı yapabiliriz.

$$E[\widetilde{\theta}(k+1)\widetilde{\theta}^T(k+1)] = [I-(R_L+R_D)^{-1}R]E[\widetilde{\theta}(k)\widetilde{\theta}^T(k)][I-(R_L+R_D)^{-1}R]^T$$
$$+\left(R_L+R_D\right)^{-1}E\left[\varphi(k)v(k)v^T(k)\varphi^T(k)\right]\left(R_L+R_D\right)^{-T} \tag{3.203}$$

Bu eşitlikte

$$\begin{aligned} E[\varphi(k)v(k)v^T(k)\varphi^T(k)] &= E[\varphi(k)E[v(k)v^T(k)]\varphi^T(k)] \\ &= E[\varphi(k)\sigma_v^2\varphi^T(k)] \\ &= \sigma_v^2 E[\varphi(k)\varphi^T(k)] \\ &= \sigma_v^2 R \end{aligned} \tag{3.204}$$

değerini yerine yazdığımızda

$$E[\widetilde{\theta}(k+1)\widetilde{\theta}^T(k+1)] = [I-(R_L+R_D)^{-1}R]E[\widetilde{\theta}(k)\widetilde{\theta}^T(k)][I-(R_L+R_D)^{-1}R]^T$$
$$+\sigma_v^2\left(R_L+R_D\right)^{-1}R\left(R_L+R_D\right)^{-T} \tag{3.205}$$

sonucunu elde ederiz. Bu eşitlik parametre tahmin hatasının korelasyon matrisinin yakınsamasını göstermektedir. Bu eşitliğe göre, öncelikle ölçme gürültüsünün olmadığını göz önüne alalım bu durumda aynı eşitliği

$$E[\widetilde{\theta}(k)\widetilde{\theta}^T(k)] = [I-(R_L+R_D)^{-1}R]E[\widetilde{\theta}(k)\widetilde{\theta}^T(k)][I-(R_L+R_D)^{-1}R]^T \tag{3.206}$$

olarak yazabiliriz. Burada $C(k) = E[\widetilde{\theta}(k)\widetilde{\theta}^T(k)]$ tanımlamasını yaparak aynı denklemi

$$C(k+1) = [I-(R_L+R_D)^{-1}R]C(k)[I-(R_L+R_D)^{-1}R]^T \tag{3.207}$$

şeklinde yazabiliriz. Bu denklemi başlangıç değerine bağlı olarak

$$C(k) = [I-(R_L+R_D)^{-1}R]^k C(0)\left([I-(R_L+R_D)^{-1}R]^T\right)^k \tag{3.208}$$

şeklinde yazabiliriz. Daha önce deterministik yakınsama analizinde ve yukarıda parametre hata vektörünün beklenen değerinin yakınsama analizinde bahsedildiği gibi R korelasyon matrisinin ve bunun bileşenlerinden oluşan $(R_L + R_D)$ matrisinin simetrik ve pozitif tanımlı olması, tekrarlamalı GS algoritmasındaki $[I - (R_L + R_D)^{-1} R]$ iterasyon matrisinin bütün özdeğerlerinin birim daire içinde olmasını sağlamaktadır. Bunun sonucunda ayrık zaman indisi $k \to \infty$ şeklinde sonsuza yaklaştıkça $[I - (R_L + R_D)^{-1} R]^k \to \mathbf{0}$ şeklinde sıfır elemanlı kare matrise yakınsamaktadır. Bu durum tekrarlamalı GS algoritmasının kararlılığı ile ilgilidir. Bunun sonucunda yukarıdaki fark denklemi hem $[I - (R_L + R_D)^{-1} R]$ iterasyon matrisini, hem de bunun transpozunu içerdiğinden dolayı parametre hata vektörünün beklenen değerinin yakınsamasını gösteren fark denklemine göre sıfır matris değerine daha hızlı yakınsayacaktır. Burada kararlılık problemine ilave olarak ölçme gürültüsünü de göz önüne aldığımızda, $E[\varphi(k)v(k)]$ karşı-ilişki vektörünün de sıfır vektöre yakınsaması durumunda (3.205) eşitliğini elde ediyorduk. Bu eşitliğin sağ tarafındaki ikinci toplam limit durumunda

$$\frac{1}{k}\sigma_v^2\left(R_L + R_D\right)^{-1} R\left(R_L + R_D\right)^{-T} \;\to\; \mathbf{0} \tag{3.209}$$

şeklinde asimptotik olarak sıfır matrise yakınsayacaktır. Sonuç olarak parametre tahmin hatasının kovaryansı $k \to \infty$ şeklinde sonsuza yaklaştıkça, limit durumunda $E[\widetilde{\theta}(k)\widetilde{\theta}^T(k)] \to \mathbf{0}$ şeklinde sıfır matrise yakınsayacaktır.

3.5.8.4. Tahmin hatasının beklenen değerinin yakınsama analizi

Tekrarlamalı GS algoritmasında tahmin hatası aşağıdaki gibi yazılabilir.

$$\varepsilon(k) = y(k) - \varphi^T(k)\hat{\theta}(k) \tag{3.210}$$

Burada $y(k)$ değerini bozucu giriş değerine bağlı olarak, ve $\hat{\theta}(k)$ parametre vektörünü de parametre hata vektörüne bağlı olarak

$$y(k) = \varphi^T(k)\theta_{opt} + v(k) \quad , \qquad \hat{\theta}(k) = \widetilde{\theta}(k) + \theta_{opt} \tag{3.211}$$

şeklinde yukarıdaki tahmin hatası denkleminde yerine koyduğumuzda

$$\begin{aligned} \varepsilon(k) &= \varphi^T(k)\theta_{opt} + v(k) - \varphi^T(k)[\widetilde{\theta}(k) + \theta_{opt}] \\ &= \varphi^T(k)\theta_{opt} + v(k) - \varphi^T(k)\widetilde{\theta}(k) - \varphi^T(k)\theta_{opt} \\ \varepsilon(k) &= v(k) - \varphi^T(k)\widetilde{\theta}(k) \end{aligned} \tag{3.212}$$

eşitliğini elde ederiz. Bu eşitliğin her iki tarafının beklenen değerini aldığımızda

$$\begin{aligned} E[\varepsilon(k)] &= E[v(k)] - E[\varphi^T(k)\widetilde{\theta}(k)] \\ E[\varepsilon(k)] &= E[v(k)] - E[\varphi^T(k)]E[\widetilde{\theta}(k)] \end{aligned} \tag{3.213}$$

eşitliği elde edilir. Yukarıda parametre tahmin hatası vektörünün yakınsama analizinde bahsedildiği gibi $E[\varphi(k)v(k)]$ karşı-ilişki vektörünün sıfır vektöre yakınsaması durumunda, yani yansız parametre tahminlerinin elde edilmesi durumunda parametre hata vektörünün beklenen değeri de $E[\widetilde{\theta}(k)] \to \mathbf{0}$ şeklinde sıfır vektöre yakınsayacağı için

$$E[\varepsilon(k)] = E[v(k)] \tag{3.214}$$

şeklinde tahmin hatasının beklenen değeri, bozucu giriş veya renkli ölçme gürültüsünün beklenen değerine yakınsayacaktır. Bu durumda bozucu giriş veya renkli ölçme gürültüsünün ortalaması $E[v(k)] \to 0$ şeklinde sıfıra yakınsadığında, tahmin hatasının ortalaması da $E[\varepsilon(k)] \to 0$ şeklinde sıfıra yakınsayacaktır.

3.5.8.5. Tahmin hatasının varyansının yakınsama analizi

Tekrarlamalı GS algoritmasında tahmin hatasını bozucu giriş ve parametre tahmin hatasına bağlı olarak (3.212) eşitliğindeki gibi yazabiliyorduk. Tahmin hatasının varyansının yakınsama analizini yapabilmek için karesini aldığımızda

$$\varepsilon^2(k) = v^2(k) - 2\,v(k)\varphi^T(k)\widetilde{\theta}(k) + \widetilde{\theta}^T(k)\varphi(k)\varphi^T(k)\widetilde{\theta}(k) \tag{3.215}$$

şeklinde yazabiliriz. Bu eşitliğin her iki tarafının beklenen değerini aldığımızda

$$\begin{aligned} E[\varepsilon^2(k)] &= E[v^2(k)] - 2\,E[v(k)\varphi^T(k)\widetilde{\theta}(k)] + E[\widetilde{\theta}^T(k)\varphi(k)\varphi^T(k)\widetilde{\theta}(k)] \\ E[\varepsilon^2(k)] &= E[v^2(k)] - 2\,E[v(k)\varphi^T(k)]\,E[\widetilde{\theta}(k)] + E[\widetilde{\theta}^T(k)\,E[\varphi(k)\varphi^T(k)]\widetilde{\theta}(k)] \end{aligned} \tag{3.216}$$

olarak yazabiliriz. Bu eşitlikteki $E[v^2(k)] = \sigma_v^2$ ve $E[\varphi(k)\varphi^T(k)] = R$ değerlerini yerine yazdığımızda, ve yansız parametre tahminlerinin elde edilmesi durumunda $E[\varphi(k)v(k)]$ karşı-ilişki vektörünün ve $E[\widetilde{\theta}(k)]$ parametre hata vektörünün sıfır vektöre yakınsadığını göz önüne aldığımızda

$$\begin{aligned} E[\varepsilon^2(k)] &= \sigma_v^2 + E[\widetilde{\theta}^T(k)\,R\,\widetilde{\theta}(k)] \\ &= \sigma_v^2 + E[tr(R\,\widetilde{\theta}(k)\widetilde{\theta}^T(k))] \\ &= \sigma_v^2 + tr[R\,E[\widetilde{\theta}(k)\widetilde{\theta}^T(k)]] \\ E[\varepsilon^2(k)] &= \sigma_v^2 + tr[R\,C(k)] \end{aligned} \tag{3.217}$$

eşitliği elde edilir (Haykin 2002). Burada tekrarlamalı GS algoritmasına ait parametre hata vektörünün korelasyon matrisinin yakınsamasını gösteren eşitliğin, $E[\varphi(k)v(k)]$ karşı-ilişki vektörünün sıfır vektöre yakınsaması durumunda, yani yansız parametre tahminlerinin elde edilmesi durumunda, $C(k) = E[\widetilde{\theta}(k)\widetilde{\theta}^T(k)] \to \mathbf{0}$ şeklinde sıfır matrisine yakınsadığını göz önüne aldığımızda, tahmin hatasının varyansının yakınsamasını gösteren yukarıdaki (3.217) eşitliğindeki ikinci terim $tr[R\,C(k)] \rightarrow 0$ şeklinde sıfıra yakınsayacağı için aşağıdaki sonucu elde ederiz.

$$E[\varepsilon^2(k)] = \sigma_v^2 \tag{3.218}$$

Sonuç olarak tekrarlamalı GS algoritmasında yardımcı değişkenlerden yararlanılması veya çıkış hatası model parametrelerinin tahmin edilmesi durumunda $\varepsilon(k)$ tahmin hatasının beklenen değeri ve varyansı, $v(k)$ bozucu girişinin veya renkli ölçme gürültüsünün beklenen değerine ve varyansına yakınsamaktadır. Tekrarlamalı GS algoritması ile ARMAX, ARARX, ARARMAX, Box-Jenkins model parametrelerinin tahmin edilmesi durumunda ise, $v(k)$ bozucu girişi veya renkli ölçme gürültüsü beyaz gürültü biçiminde değişen bir işaretten elde edilmiş gibi modellenmektedir. Yani sisteme beyaz gürültü biçiminde bir giriş veya beyaz ölçme gürültüsü varmış gibi modelleme yapıldığı için, tahmin hatasının beklenen değeri ve varyansı, beyaz gürültünün beklenen değerine ve varyansına yakınsayacaktır. Sonuçta karesel ortalama hata fonksiyonu asimptotik olarak $k \rightarrow \infty$ şeklinde zaman sonsuza yaklaştıkça

$$V_k(\hat{\theta}(k)) = \frac{1}{k}\sum_{n=1}^{k} \varepsilon^2(k) \quad \rightarrow \quad \sigma_e^2 \tag{3.219}$$

şeklinde beyaz gürültünün varyansına yakınsayacaktır. Yani asimptotik olarak maksimum benzerlik tahminleri elde edilecektir.

Yukarıda (3.218) eşitliğinde, parametre tahminlerinin korelasyon matrisi sıfıra yakınsadığı için, tahmin hatasının varyansının kalıcı durumdaki değerinin minimum değerine yakınsadığı gösterilmiştir. Karesel ortalama hata fonksiyonunun kalıcı durumdaki değerinin asimptotik olarak $V_\infty(\hat{\theta}) \rightarrow \sigma_v^2$ değerine yakınsadığını, minimum değerini ise teorik olarak $V_{\min} = \sigma_v^2$ olduğunu göz önüne aldığımızda, bunların arasındaki fark, zamanla asimptotik olarak sıfıra yakınsayacaktır. Bu özellik uyarlamalı işaret işleme literatüründe "Excess of MSE (Mean Square Error)" olarak adlandırılmaktadır ve aşağıdaki gibi ifade edilmektedir (Haykin 2002). Burada

$$V_{\infty}(\hat{\theta}) = E\left[\varepsilon^2(k)\right] = \underset{k\to\infty}{Lim}\ \frac{1}{k}\sum_{n=1}^{k}\varepsilon^2(n) \tag{3.220}$$

olmak üzere "Excess-MSE"

$$V_{exc} = V_{\infty}(\hat{\theta}) - V_{\min} \tag{3.221}$$

olarak tanımlanmaktadır. Bu fark, tahmin edilen parametrik sistem modeliyle yapılan çıkış tahminlerinin (prediction), gerçek sistemin çıkışını izleyebilmesinin bir ölçüsüdür. Bu farkın zamanla asimptotik olarak sıfıra yakınsayabilmesi için, kullanılan adaptasyon algoritmasıyla elde edilen parametre tahminlerinin korelasyon matrisinin asimptotik olarak sıfıra yakınsaması gerekir. Buna benzer bir diğer özellik ise uyarlamalı işaret işleme literatüründe "Misadjustment" olarak bilinmektedir ve aşağıdaki gibi tanımlanmaktadır (Haykin 2002).

$$\mathrm{M} = \frac{V_{exc}}{V_{\min}} = \frac{V_{\infty}(\hat{\theta}) - V_{\min}}{V_{\min}} \tag{3.222}$$

Bu özellik de V_{exc} değerine bağlı olduğu için aynı anlama gelmektedir ve birlikte anılmaktadır. Sonuç olarak buraya kadar yapılan analiz sonuçlarını değerlendirdiğimizde, tekrarlamalı GS algoritmasıyla yapılan parametre tahmin işleminde parametre tahmin hatası vektörünün, parametre tahmin hatası vektörünün korelasyon matrisinin, ayrıca "Excess-MSE" ve "Misadjustment" değerlerinin zamanla asimptotik olarak sıfıra yakınsadığı görülmektedir. Bu değerler RLS algoritmasında da asimptotik olarak sıfıra yakınsamaktadır (Haykin 2002).

3.6. Tekrarlamalı Gauss-Seidel ve Tekrarlamalı Jacobi Algoritmalarıyla Doğrusal Olmayan Volterra Model Parametrelerinin Tahmin Edilmesi

Volterra serileri doğrusal olmayan birçok sistemi modelleyebilmektedir (Schetzen 1980). Bu serilerin katsayıları sayısal filtre katsayıları olarak ele alındığında, bu filtreler

Volterra filtre olarak adlandırılmaktadır (Mathews 1991, Sicuranza 1992, Mathews ve Sicuranza 2000). Volterra filtreler histerezis haricinde doğrusal olmayan birçok sinyal ve sistemlerin modellenmesinde kullanılmaktadır. Örneğin haberleşme alanında doğrusal olmayan kanal dengeleme, doğrusal olmayan eko silme, veri iletim sistemleri gibi, ayrıca elektrik-akustik dönüştürücüler, konuşma kodlama, görüntü işleme gibi alanlarda çeşitli uygulamalarda kullanılmaktadır (Kajikawa 2000, Mathews ve Sicuranza 2000). Ayrıca biyolojik ve fizyolojik sistemlerin modellenmesi (Westwick ve Kearney 2003), aktif gürültü kontrolü (Tan ve Jiang 2001, Carini ve Sicuranza 2004), uyarlamalı sistem tanıma ve kontrol (Doyle ve ark. 2001) gibi alanlarda da kullanılmaktadır.

Volterra filtreler, doğrusal filtrelerin performansının yetersiz kaldığı durumlarda doğrusal olmayan modelleme işleminde kullanılmaktadır. Uyarlamalı Volterra filtrelerin uygulamalarında bilinmeyen sistemin kendisi veya tersi modellenmektedir ve bilinmeyen sistem parametrelerinin tahmin edilmesi gerekmektedir. Bu amaçla doğrusal olmayan sistem tanıma işleminde etkili bir biçimde kullanılmaktadır (Mathews 1991, Sicuranza 1992, Haber ve Keviczky 1999, Mathews ve Sicuranza 2000, Doyle ve ark. 2001, Zaknich 2005, Ogunfunmi 2007).

Uyarlamalı Volterra filtrelerin çıkışları filtre parametrelerinin doğrusal bir fonksiyonu olduğundan (linear-in-the-parameters) dolayı parametre tahminlerinin yakınsaması doğrusal filtrelerde olduğu gibi garantilenebilmektedir. Uyarlamalı Volterra filtrelerde giriş işaretinin yavaş değişen bir işaret, örneğin renkli gürültü olması durumunda, giriş işaretinin önceki değerleriyle çoklu çarpımlarının kullanılmasından dolayı giriş işaretinin korelasyon matrisinin özdeğer yayılımı doğrusal filtrelere göre daha büyük olmaktadır. Giriş işaretinin beyaz gürültü olması durumunda ise korelasyon matrisinin özdeğer yayılımı doğrusal filtrelerde olduğu gibi küçük olmaktadır. Bu özelliklerden dolayı doğrusal uyarlamalı filtre katsayılarının ayarlanmasında kullanılan birçok algoritma Volterra filtre katsayılarının ayarlanmasında da kullanılmaktadır. Bu algoritmalar eğim tabanlı ve en küçük kareler tabanlı olmak üzere iki gruba ayrılmaktadır (Kajikawa 2000):

1. Eğim tabanlı algoritmalar:
   - LMS (Least Mean Squares) algoritması
   - NLMS (Normalized LMS) algoritması
   - AP (Affine Projection) algoritması
2. En küçük kareler tabanlı algoritmalar:
   - RLS (Recursive Least Squares) algoritması
   - Hızlı RLS algoritması
   - Hızlı Kalman algoritması
   - Kafes RLS algoritması
   - QR – RLS algoritması

Bu çalışmada, Volterra parametrelerinin güncellenmesi için, zaman ortalamalı normal denklemin doğrusal denklem takımı çözüm yöntemleri ile çözümü üzerine kurulu olan tekrarlamalı GS ve tekrarlamalı Jacobi algoritmaları önerilmiştir. Bu kısımda öncelikle doğrusal olmayan Volterra modeller tanıtılacaktır, daha sonra Volterra model parametrelerini güncellemek için önerilen tekrarlamalı GS ve Jacobi algoritmaları tanıtılacaktır.

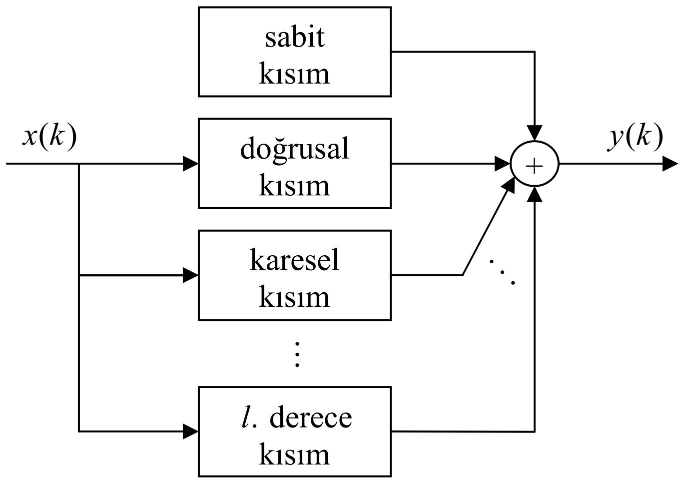
3.6.1. Doğrusal olmayan Volterra modellerin tanıtımı

Bir zamanla değişmeyen, nedensel, doğrusal olmayan ayrık-zamanlı sistem,

$$\begin{aligned} y(k) = h_0 &+ \sum_{n_1=0}^{\infty} h_1(n_1)x(k-n_1) + \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty} h_2(n_1,n_2)x(k-n_1)x(k-n_2) \\ &+ \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}\sum_{n_3=0}^{\infty} h_3(n_1,n_2,n_3)x(k-n_1)x(k-n_2)x(k-n_3) \\ &+ \cdots + \sum_{n_1=0}^{\infty}\cdots\sum_{n_l=0}^{\infty} h_l(n_1,\ldots,n_l)x(k-n_1)\ldots x(k-n_l) \end{aligned} \tag{3.223}$$

şeklinde seriye açılım biçiminde gösterilebilir (Schetzen 1980, Kajikawa 2000, Mathews ve Sicuranza 2000, Ogunfunmi 2007). Burada $x(k)$ ve $y(k)$ sırasıyla k anındaki giriş ve çıkış işaretlerini, l ise filtrenin derecesini göstermektedir. Ayrıca h_0 sabit bileşen (dc offset component), $h_1(n_1)$ filtrenin doğrusal bileşenlerinin katsayıları,

$h_2(n_1,n_2)$ filtrenin 2. derece (karesel) bileşenlerinin katsayıları, $h_3(n_1,n_2,n_3)$ filtrenin 3. derece (kübik) bileşenlerinin katsayıları ve en genel haliyle $h_l(n_1,\ldots,n_l)$ filtrenin l. derece bileşenlerinin katsayılarıdır. Yukarıdaki Volterra model denkleminde sabit kısımdan sonraki tek toplam filtrenin doğrusal kısmını, sonraki ikili çarpımların toplamı filtrenin 2. derece (karesel) kısmını, sonraki üçlü çarpımların toplamı filtrenin 3. derece (kübik) kısmını, ve en genel haliyle l adet çarpımların toplamını gösteren son terim ise filtrenin l. derece kısmını oluşturmaktadır. Volterra filtrenin çıkışı Şekil 3.10'da görüldüğü gibi filtrenin sabit, doğrusal ve doğrusal olmayan kısımlarının toplamından elde edilmektedir.



Şekil 3.10. l. derece Volterra filtre.

Pratik uygulamalarda çoğunlukla 2. derece olmak üzere en fazla 3. derece kısımların kullanıldığı sonlu veri uzunluğuna sahip Volterra filtreler kullanılmaktadır. Örneğin geriye dönük M adet veriyi kullanan 2. derece Volterra filtrenin denklemi aşağıdaki gibidir.

$$y(k)=h_0+\sum_{n_1=0}^{M-1}h_1(n_1)x(k-n_1)+\sum_{n_1=0}^{M-1}\sum_{n_2=0}^{M-1}h_2(n_1,n_2)x(k-n_1)x(k-n_2) \qquad (3.224)$$

Bu denklem doğrusal bağlanımlı biçimde

$$y(k) = \boldsymbol{x}^T(k)\boldsymbol{h} \tag{3.225}$$

olarak yazıldığında buradaki $\boldsymbol{h}$ parametre vektörü ve $\boldsymbol{x}(k)$ veri vektörü

$$\begin{aligned}\boldsymbol{h} = \big[&h_0\ h_1(0)\cdots h_1(M-1)\ h_2(0,0)\ h_2(0,1)\cdots h_2(0,M-1)\\ &h_2(1,0)\ h_2(1,1)\cdots h_2(1,M-1)\cdots h_2(M-1,0)\ h_2(M-1,1)\cdots\ h_2(M-1,M-1)\big]^T\end{aligned} \tag{3.226}$$

$$\begin{aligned}\boldsymbol{x}(k) = \big[&1\ \ x(k)\ \cdots\ x(k-M+1)\ \ x^2(k)\ \ x(k)x(k-1)\ \cdots\ x(k)x(k-M+1)\\ &x(k-1)x(k)\ \ x^2(k-1)\ \cdots\ x(k-1)x(k-M+1)\ \cdots\\ &\cdots\ x(k-M+1)x(k)\ \ x(k-M+1)x(k-1)\ \cdots\ x^2(k-M+1)\big]^T\end{aligned} \tag{3.227}$$

biçiminde yazılabilir (Mathews 1991, Mathews ve Sicuranza 2000, Zaknich 2005).

3.6.2. Tekrarlamalı Gauss-Seidel algoritması ile Volterra model tahmini

Sistem tanıma işleminde doğrusal olmayan sinyal ölçümlerine gürültü karıştığı varsayılarak sistemin doğrusal bağlaşımlı biçimde yazılan (3.225) denklemi

$$y(k) = \boldsymbol{x}^T(k)\boldsymbol{h} + v(k) \tag{3.228}$$

olarak yazıldığında parametre tahmin işlemindeki eşitlik hatasını

$$e(k) = y(k) - \boldsymbol{x}^T(k)\boldsymbol{h}(k-1) \tag{3.229}$$

olarak yazabiliriz. Burada minimum yapılmak istenen fonksiyon

$$V(\theta) = E[e^2(k)] \tag{3.230}$$

olup bu fonksiyonu minimum yapan optimum parametreler

$$\boldsymbol{h}_{opt} = R^{-1}p \tag{3.231}$$

olarak verilebilir. Burada R korelasyon matrisi ve p korelasyon vektörü

$$R = E[\boldsymbol{x}(k)\boldsymbol{x}^T(k)] \quad , \quad p = E[\boldsymbol{x}(k)y(k)] \tag{3.232}$$

olarak tanımlanmaktadır. Pratik uygulamalarda R matrisinin ve p vektörünün tahmin edilmiş zaman ortalamalı değerleri k adet veri kullanılarak aşağıdaki gibi yazılabilir,

$$R(k) = \frac{1}{k}\sum_{i=1}^{k}\boldsymbol{x}(i)\boldsymbol{x}^T(i) \quad , \quad p(k) = \frac{1}{k}\sum_{i=1}^{k}\boldsymbol{x}(i)y(i) \tag{3.233}$$

veya $1/k$ çarpanları göz önüne alınmadan giriş-çıkış verileri alındıkça ardışık olarak

$$R(k) = R(k-1) + \boldsymbol{x}(k)\boldsymbol{x}^T(k) \quad , \quad p(k) = r(k-1) + \boldsymbol{x}(k)y(k) \tag{3.234}$$

şeklinde güncellenebilir. Sonra bir adımlık GS iterasyonu (Tekrarlamalı GS algoritması)

$$R(k)\hat{\boldsymbol{h}}(k) = p(k) \tag{3.235}$$

olarak yazılan zaman ortalamalı normal denklemin çözümünde $i = 0,1,\ldots,N$ için

$$\hat{\boldsymbol{h}}_i(k) = \left[p_i(k) - \sum_{j=1}^{i-1}R_{ij}(k)\hat{\boldsymbol{h}}_j(k) - \sum_{j=i+1}^{N}R_{ij}(k)\hat{\boldsymbol{h}}_j(k-1)\right] \Big/ R_{ii}(k) \tag{3.236}$$

şeklinde kullanılır, burada N Volterra filtrenin parametre sayısını göstermektedir.

3.6.3. Tekrarlamalı Jacobi algoritması ile Volterra model tahmini

Tekrarlamalı Jacobi algoritmasında da minimum yapılmak istenen hata kriteri (3.230) eşitliğindeki gibidir. Tekrarlamalı GS algoritmasındaki gibi korelasyon matrisinin tersi yerine kendisi $R(k)$ ve korelasyon vektörü $p(k)$ (3.234) eşitliğindeki gibi güncellenir ve (3.235) ile verilen zaman ortalamalı normal denklemin Jacobi algoritması ile çözümünde kullanılır. Bu durumda elde edilen iteratif algoritma $i = 0,1,\ldots,N$ için

$$\hat{\boldsymbol{h}}_i(k) = \left[p_i(k) - \sum_{j=1}^{i-1} R_{ij}(k)\hat{\boldsymbol{h}}_j(k-1) - \sum_{j=i+1}^{N} R_{ij}(k)\hat{\boldsymbol{h}}_j(k-1) \right] \Big/ R_{ii}(k) \tag{3.237}$$

şeklinde verilebilir. Tekrarlamalı Jacobi algoritmasının yakınsaması, korelasyon matrisi tahmin edilen değerinin pozitif tanımlı olması durumunda garantilenmiş değildir. Jacobi algoritmasının yakınsama hızını yavaşlatarak kararlılığını garantilemek için algoritmaya bir adım parametresi ilave edilmektedir. Bunun için algoritma öncelikle vektörel biçimde

$$\hat{\boldsymbol{h}}(k) = \hat{\boldsymbol{h}}(k-1) + R_D^{-1}(k)[p(k) - R(k)\hat{\boldsymbol{h}}(k-1)] \tag{3.238}$$

olarak yazılabilir, burada $R_D(k)$ korelasyon matrisini köşegeninde bulunan elemanlarını içeren ve diğer elemanları sıfır olan bir kare matristir. Tekrarlamalı Jacobi algoritmasının avantajı, $R_D(k)$ matrisinin tersinin kolayca alınabilmesinden dolayı parametre tahminlerinin vektörel olarak da kolaylıkla güncellenebilmesidir. Tekrarlamalı GS algoritmasında ise parametre tahminleri ancak teker teker skaler olarak güncellenebilmektedir.

Tekrarlamalı Jacobi algoritması yukarıdaki gibi vektörel biçimde yazıldığında, algoritmanın kararlılığını kontrol edebilmek için gerekli μ adım parametresini LMS grubu algoritmalara benzer şekilde aşağıdaki gibi ilave edilebilir.

$$\hat{\boldsymbol{h}}(k) = \hat{\boldsymbol{h}}(k-1) + \mu R_D^{-1}(k)[p(k) - R(k)\hat{\boldsymbol{h}}(k-1)] \tag{3.239}$$

Çünkü bu eşitlikteki $[p(k) - R(k)\hat{\boldsymbol{h}}(k-1)]$ vektörü, uyarlamalı filtre parametrelerini güncellemek için kullanılan birikimli bir hata vektörüdür. Giriş sinyalinin varyansı büyük olduğunda algoritma kararsız olmaktadır. Bu durumda adım parametresi $\mu < 1$ olarak alınarak algoritmanın yakınsama hızını yavaşlaması pahasına kararlılığını garantilemektedir. Tekrarlamalı Jacobi algoritmasının kararlı olabilmesi için μ adım parametresinin

$$0 < \mu < \frac{2}{\lambda_{\max}[R_D^{-1} R]} \tag{3.240}$$

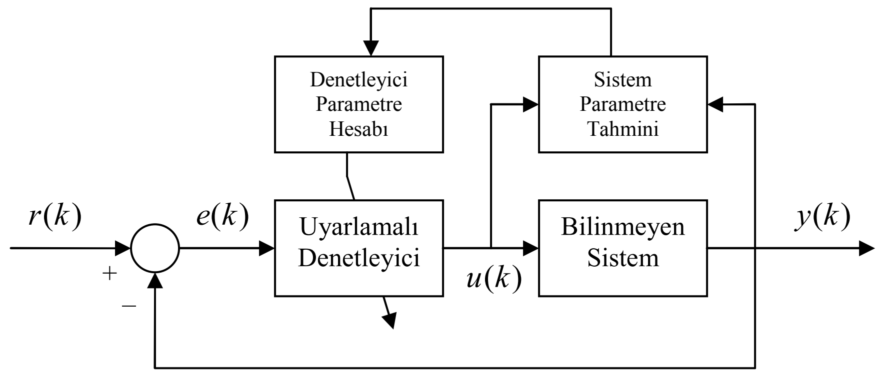
aralığında olması gerekmektedir, burada $\lambda_{\max}[R_D^{-1} R]$ değeri $[R_D^{-1} R]$ matrisinin en büyük özdeğeridir (Zhang ve ark. 2005). Bu durumda parametre tahminlerinin skaler olarak güncellemek için (3.239) ile verilen tekrarlamalı Jacobi algoritması $i = 0,1,\ldots,N$ için

$$\hat{\boldsymbol{h}}_i(k) = \frac{\mu}{R_{ii}(k)}\left[p_i(k) - \sum_{j=1}^{i-1} R_{ij}(k)\hat{\boldsymbol{h}}_j(k-1) - \sum_{j=i+1}^{N} R_{ij}(k)\hat{\boldsymbol{h}}_j(k-1)\right] + (1-\mu)\hat{\boldsymbol{h}}_i(k-1) \tag{3.241}$$

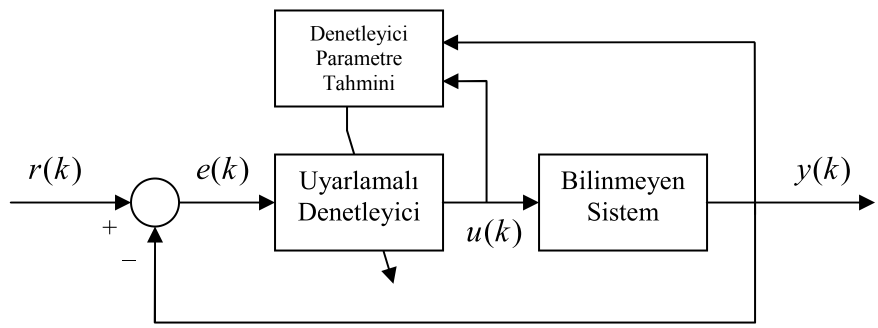
şeklinde yazılabilir. Bu yöntem literatürde “Jacobi Overrelaxation” yöntemi olarak da bilinmekte ve doğrusal denklem takımlarının çözümünde kullanılmaktadır (Young 1971).

3.7. Tekrarlamalı Gauss-Seidel Algoritması ile Uyarlamalı Kontrol

Uyarlamalı kontrol sistemlerinin tasarımı eğim yöntemleri, kararlılık teorisi, kutup yerleştirme ve tahmin (prediction) yöntemleri kullanılarak değişik şekillerde yapılmaktadır (Goodwin ve Sin 1984, Wellstead ve Zarrop 1991, Isermann ve ark. 1992, Butler 1992, Aström ve Wittenmark 1995, Landau ve ark. 1998, Bobal ve ark. 1999,2005). Sistem tanıma algoritmalarının önemli bir uygulama alanı da kutup yerleştirme yöntemini kullanan kendinden ayarlamalı (self-tuning) denetleyicilerdir. Kendinden ayarlamalı denetleyicilerde belirli bir performans kriterini sağlayan denetleyici parametreleri tekrarlamalı (recursive, on-line) bir biçimde güncellenir. Uyarlamalı kontrol yöntemlerinde denetleyici parametreleri dolaylı ve doğrudan olmak üzere iki farklı biçimde güncellenir. Dolaylı güncelleme işleminde önce sistemin giriş-çıkış verileri kullanılarak transfer fonksiyonu parametreleri tahmin edilir, sonra elde edilen parametre tahminleri yardımıyla denetleyici parametreleri hesaplanır. Doğrudan güncelleme işleminde ise kontrol kuralı uygun bir şekilde düzenlenerek, sistemin giriş-çıkış verileri kullanılarak doğrudan denetleyici parametreleri güncellenir (Goodwin ve Sin 1984, Gawthrop 1986,1987, Wellstead ve Zarrop 1991, Isermann ve ark. 1992, Aström ve Wittenmark 1995, Landau ve ark. 1998, Bobal ve ark. 1999,2005). Şekil 3.11'de dolaylı uyarlamalı kontrol yöntemlerinin blok diyagramı, Şekil 3.12'de ise doğrudan uyarlamalı kontrol yöntemlerinin blok diyagramı görülmektedir.



Şekil 3.11. Uyarlamalı denetleyici parametrelerinin dolaylı olarak ayarlanması.



Şekil 3.12. Uyarlamalı denetleyici parametrelerinin doğrudan ayarlanması.

Bu kısımda öncelikle dolaylı uyarlamalı denetleyiciler üzerinde durulacaktır. Daha sonra doğrudan uyarlamalı denetleyiciler ele alınacaktır.

3.7.1. Tekrarlamalı Gauss-Seidel algoritması ile dolaylı uyarlamalı kontrol

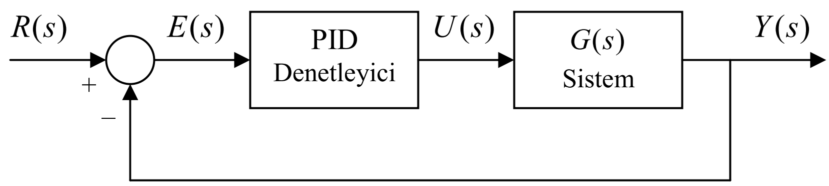
Burada sırasıyla kendinden ayarlamalı Ziegler-Nichols PID denetleyicilerin, kutup yerleştirmeli PID denetleyicilerin ve model-tabanlı dolaylı uyarlamalı denetleyicilerin tekrarlamalı GS algoritmasıyla ayarlanması üzerinde durulacaktır.

3.7.1.1. Kendinden ayarlamalı Ziegler-Nichols PID denetleyici katsayılarının tekrarlamalı Gauss-Seidel algoritması ile ayarlanması

PID denetleyiciler, gerçekleme kolaylığından ve iyi bilindiğinden dolayı günümüzde endüstride hala yaygın olarak kullanılmaktadır. Bu denetleyiciler, integral etkisi ile kalıcı durum hatasını ortadan kaldıran, türev etkisi ile kapalı çevrim sistemin bant genişliğini arttırarak aşımı azaltan veya sıfırlayan, oransal etki ile de sistem cevabının yükselme ve yerleşme zamanlarını küçülterek sistemin cevabını hızlandıran geribeslemeli denetleyicilerdir (Aström ve Hagglund 1988,1995,2006). Analog PID denetleyiciler pratikte opamplı devreler ile kolaylıkla gerçeklenebilmektedir ve birçok endüstriyel kontrol sisteminde yarım yüzyıldan beri yaygın olarak kullanılmaktadır. PID denetleyicinin girişi Şekil 3.13'te görüldüğü gibi hata sinyali olup çıkışı sistemi kontrol etmek için kullanılan kontrol sinyalidir. Analog PID denetleyicilerin transfer fonksiyonu

$$\frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \tag{3.242}$$

olarak yazılabilir, burada K_p oransal kazanç, $K_i = K_p / T_i$ integral kazancı, $K_d = K_p T_d$ türev kazancı, T_i integral sabiti, T_d türev sabiti olup $E(s)$ ve $U(s)$ ise sırasıyla hata ve kontrol sinyalinin Laplace dönüşümüdür.



Şekil 3.13. PID denetleyicili geribeslemeli kontrol sistemi.

PID denetleyicinin sayısal bir formunu elde etmek için (3.242) eşitliğindeki türev ve integral işlemlerinin ayrıklaştırılması gerekmektedir. İntegral işlemi bir eğrinin altında kalan alanın yaklaşık hesabına dayalı olarak ayrıklaştırılabilir. Bu amaçla ileriye doğru dikdörtgen yöntemi, geriye doğru dikdörtgen yöntemi ve yamuk kuralı kullanılmaktadır. Yamuk kuralının nümerik hatası daha azdır. Türev işlemi ise çoğunlukla iki nokta arasındaki farkın örnekleme aralığına oranı kullanılarak ayrıklaştırılmaktadır. Örneğin yamuk alanı hesabı kullanılarak gerçeklenen integrasyon işlemi ve fark alarak gerçeklenen türev işlemi kullanıldığında (3.242) ile verilen analog PID denetleyicinin sayısal eşdeğerini

$$\frac{U(z)}{E(z)} = K_P + \frac{K_I}{1-z^{-1}} + K_D(1-z^{-1}) \tag{3.243}$$

olarak elde edilir (Ogata 1995, Landau ve Zito 2006). Bu denklemdeki sayısal PID denetleyicinin katsayıları, analog PID denetleyicinin katsayılarından

$$\begin{aligned}
K_P &= K_p - K_p T/2T_i = K_p - K_I/2 && \text{(oransal kazanç)} \\
K_I &= K_p T/T_i && \text{(integral kazancı)} \\
K_D &= K_p T_d/T && \text{(türev kazancı)}
\end{aligned} \tag{3.244}$$

şeklinde T örnekleme periyoduna ve kullanılan integral ve türev yaklaşıklığına bağlı olarak hesaplanabilir. Bu formda elde edilen sayısal PID denetleyicinin fark denklemi

$$u(k) = u(k-1) + (K_P + K_I + K_D)e(k) - (K_P + 2K_D)e(k-1) + K_D e(k-2) \tag{3.245}$$

olarak yazılabilir (Isermann 1989, Ogata 1995, Landau ve Zito 2006). Aynı fark denklemini

$$u(k) = u(k-1) + q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \tag{3.246}$$

biçiminde yazdığımızda sayısal PID denetleyicinin transfer fonksiyonunu

$$\frac{U(z)}{E(z)} = \frac{Q(z^{-1})}{P(z^{-1})} = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{1 - z^{-1}} \tag{3.247}$$

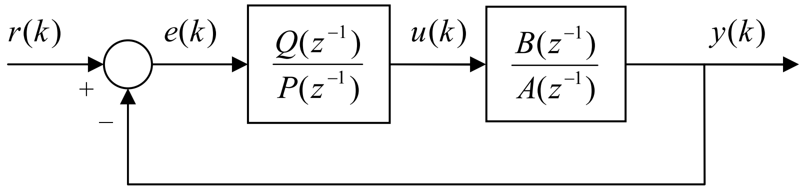
biçiminde yazabiliriz. Burada sayısal PID denetleyicinin fark denkleminin katsayılarını

$$\begin{aligned} q_0 &= K_P + K_I + K_D \\ q_1 &= -(K_P + 2K_D) \\ q_2 &= K_D \end{aligned} \tag{3.248}$$

olarak yazabiliriz. Sonuç olarak yapılan ayrıklaştırma işlemine bağlı olarak denetleyici katsayılarını

$$q_0, q_1, q_2 = f(K_P, K_I, K_D) = f(K_p, K_i, K_d, T) \tag{3.249}$$

şeklinde birbirine dönüştürebiliriz. Yani kullanılan yaklaşıklığa ve T örnekleme periyoduna bağlı olarak (K_p, K_i, K_d) analog PID katsayılarından (K_P, K_I, K_D) sayısal PID denetleyici katsayılarını, oradan da (q_0, q_1, q_2) sayısal PID denetleyicinin fark denkleminin katsayılarını hesaplayabiliriz. Burada girişi hata sinyali, çıkışı kontrol sinyali olan sayısal PID denetleyici literatürde konumsal biçim (positional form) PID denetleyici olarak bilinmektedir ve blok diyagramı aşağıda Şekil 3.14'te görülmektedir.



Şekil 3.14. Konumsal biçim sayısal PID denetleyicili geribeslemeli kontrol sistemi.

Konumsal biçim PID denetleyicilerin dezavantajı, referans girişin ani değişimlerinde kapalı çevrim sistemin fazla aşım yapmasıdır. Hızsal biçim (velocity form) PID denetleyici, konumsal biçim PID denetleyicide $e(k) = r(k) - y(k)$ yazıldıktan sonra referans girişin son üç örneğinin $r(k) = r(k-1) = r(k-2)$ şeklinde birbirine eşit alınmasıyla aşağıdaki gibi elde edilmektedir (Ogata 1995).

$$\begin{aligned}
u(k) &= u(k-1) + (K_P + K_I + K_D)e(k) - (K_P + 2K_D)e(k-1) + K_D e(k-2) \\
&= u(k-1) + (K_P + K_I + K_D)[r(k) - y(k)] - (K_P + 2K_D)[r(k-1) - y(k-1)] \\
&\qquad + K_D[r(k-2) - y(k-2)] \\
&= u(k-1) + K_I[r(k) - y(k)] - K_P[y(k) - y(k-1)] - K_D[y(k) - y(k-1) - y(k-2)] \\
u(k) &= u(k-1) + K_I r(k) - (K_P + K_I + K_D)y(k) + (K_P + 2K_D)y(k-1) - K_D y(k-2)
\end{aligned} \tag{3.250}$$

Burada kontrol sinyali referans girişin son üç örneğinin eşit olduğu varsayımıyla hesaplandığı için, değişim üç örnek sonra kontrol sinyalini değiştirecek ve kapalı çevrim sistemin çıkışında daha az aşım olacaktır.

Kendinden ayarlamalı Ziegler-Nichols PID kontrol yöntemi ayrık-zaman sistemler için Bobal (1995) tarafından geliştirilmiştir (Bobal ve ark. 1999,2005). Başlangıçta ayrık-zaman sistemin transfer fonksiyonu

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} z^{-d} \tag{3.251}$$

olarak yazılmıştır. Burada sistemin ölü zaman gecikmesi d olmak üzere transfer fonksiyonunun pay ve payda polinomları sırasıyla

$$B(z^{-1}) = b_1 z^{-1} + \cdots + b_n z^{-n}$$
$$A(z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_n z^{-n} \tag{3.252}$$

olarak tanımlanmıştır. Başlangıçta sistemde $C(z^{-1}) = K_P$ şeklinde oransal denetleyici olduğu varsayılmıştır. Bu durumda kapalı çevrim sistemin transfer fonksiyonu

$$\frac{Y(z)}{R(z)} = \frac{z^{-d} K_P B(z^{-1})}{A(z^{-1}) + z^{-d} K_P B(z^{-1})} \tag{3.253}$$

olup karakteristik denklem aşağıdaki gibi yazılabilir.

$$D(z^{-1}) = A(z^{-1}) + z^{-d} K_P B(z^{-1}) \tag{3.254}$$

Bu denklemin kökleri kapalı çevrim sistemin davranışını belirler. Oransal kazanç değeri $K_P = K_u$ değerini aldığında karakteristik denklemin kökleri z-düzleminde birim çember üzerinde olur ve kapalı çevrim sistem kararsızlık sınırında periyodu T_u olan sabit genlikli salınım yapar. Burada karakteristik denklemin baskın köklerinin z-düzleminde birim çember üzerinde $z_{1,2} = \alpha \mp j\beta$ şeklinde karmaşık-eşlenik olması durumunda kapalı çevrim sistemin çıkışı sabit genlikli salınım yapar ve $\alpha^2 + \beta^2 = 1$ için bu iki karmaşık kökün çarpımı

$$C_1(z) = z^2 - \alpha z + 1 = z^2 - 2\cos\omega T z + 1 \tag{3.255}$$

olur. Burada sistemin çıkışında sabit genlikli salınımın frekansı $\omega = \omega_u$ değerini aldığında karmaşık köklerin reel kısmı

$$\alpha = \cos\omega_u T = \cos 2\pi T / T_u \tag{3.256}$$

olur ve buradan

$$T_u = 2\pi T / \cos^{-1} \alpha \tag{3.257}$$

olarak hesaplanabilir. Örneğin ikinci dereceden gecikmesiz bir sistem için kararsızlık sınırındaki kazanç değeri K_u ve bu durumda kapalı çevrimin çıkışında oluşan sabit genlikli salınımın periyodu aşağıdaki gibi hesaplanır. Burada sistemin transfer fonksiyonunun

$$\frac{B(z^{-1})}{A(z^{-1})} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{3.258}$$

olması durumunda kapalı çevrim sistemin paydası, yani karakteristik denklem

$$D(z) = z^2 + (a_1 + b_1 K_P) z + (a_2 + b_2 K_P) = z^2 - 2\alpha z + 1 \tag{3.259}$$

olur. Burada $K_P = K_u$ için karakteristik denklemin katsayıları eşitlenerek

$$a_1 + b_1 K_u = -2\alpha \quad , \quad a_2 + b_2 K_u = 1 \tag{3.260}$$

yazılabilir. Bu eşitliklerden ikincisi kullanılarak kapalı çevrim sistemi kararsızlık sınırına getiren kazanç değeri transfer fonksiyonu katsayıları cinsinden

$$K_u = (1 - a_2) / b_2 \tag{3.261}$$

olarak hesaplanır, sonra bu değer ilk eşitlikte kullanılarak karmaşık köklerin reel kısmı

$$\alpha = -(a_1 + b_1 K_u) / 2 \tag{3.262}$$

olarak hesaplanır. Bu durumda kapalı çevrim sistemin çıkışındaki salınımların periyodu $T_u = 2\pi T / \cos^{-1} \alpha$ kullanılarak hesaplanabilir. İkinci ve üçüncü derece sistemler için

K_u ve T_u değerlerinin hesaplanması kolaydır ve Bobal ve ark. (1999,2005) tarafından verilmiştir. Daha yüksek dereceli sistemler için bu değerler nümerik optimizasyon yöntemleri kullanılarak hesaplanabilir (Bobal ve ark. 1999,2005).

Ayrık-zaman sistemin transfer fonksiyonu parametrelerinin tahmin edilerek kullanılması durumunda kendinden ayarlamalı Ziegler-Nichols PID kontrol yöntemi elde edilir. Burada ayrık-zaman sistemin transfer fonksiyonu parametrelerini çevrim-içi tahmin etmek için tekrarlamalı GS algoritması aşağıdaki gibi kullanılabilir. Burada sisteme ait veri vektörü ve parametre tahmin vektörü

$$\phi(k) = \begin{bmatrix} -y(k-1) & \cdots & -y(k-n) & u(k-1) & \cdots & u(k-n) \end{bmatrix}^T \tag{3.263}$$

$$\hat{\theta}(k) = \begin{bmatrix} \hat{a}_1 & \cdots & \hat{a}_n & \hat{b}_1 & \cdots & \hat{b}_n \end{bmatrix}^T \tag{3.264}$$

olmak üzere tekrarlamalı GS algoritması $(i = 1,2,\ldots,2n)$ için

$$R(k) = R(k-1) + \phi(k)\phi^T(k) \quad , \quad p(k) = p(k-1) + \phi(k)y(k) \tag{3.265}$$

$$\hat{\theta}_i(k) = \left[p_i(k) - \sum_{j=1}^{i-1} R_{ij}(k)\hat{\theta}_j(k) - \sum_{j=i+1}^{2n} R_{ij}(k)\hat{\theta}_j(k-1) \right] \Big/ R_{ii}(k) \tag{3.266}$$

şeklinde kullanılabilir. Burada tahmin edilen transfer fonksiyonu katsayıları Ziegler-Nichols PID katsayılarının ayarlanmasında kullanılabilir. Örneğin ikinci derece sistem için kararsızlık sınırındaki kazanç değeri ve sistemin çıkışındaki salınımın periyodu

$$K_u = (1-\hat{a}_2)/\hat{b}_2 \quad , \quad \alpha = -(\hat{a}_1 + \hat{b}_1 K_u)/2 \quad , \quad T_u = 2\pi T / \cos^{-1}\alpha \tag{3.267}$$

formülleri kullanılarak hesaplanabilir. Hesaplanan bu sayısal değerler ile sürekli-zaman Ziegler-Nichols PID katsayıları aşağıdaki gibi hesaplanabilir (Bobal ve ark. 1999,2005).

$$K_p = 0.6K_u \quad , \quad T_i = 0.5T_u \quad , \quad T_d = 0.125T_u \tag{3.268}$$

Sonra sayısal PID katsayıları hesaplanarak konumsal biçim veya hızsal biçimde verilen sayısal PID denetleyicinin fark denkleminde kullanılabilir.

3.7.1.2. Kendinden ayarlamalı kutup yerleştirmeli PID denetleyici katsayılarının tekrarlamalı Gauss-Seidel algoritması ile ayarlanması

Polinomsal kutup yerleştirme yöntemiyle sayısal PID denetleyici katsayılarını ayarlamak için, denetleyicinin transfer fonksiyonu aşağıdaki gibi iki farklı biçimde kullanılmaktadır (Bobal ve ark. 2005).

$$\frac{U(z)}{E(z)} = \frac{Q(z^{-1})}{P(z^{-1})} = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{1 - z^{-1}} \tag{3.269}$$

$$\frac{U(z)}{E(z)} = \frac{Q(z^{-1})}{P(z^{-1})} = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{(1 - z^{-1})(1 + \gamma z^{-1})} \tag{3.270}$$

Polinomsal kutup yerleştirme yöntemiyle PID parametrelerini hesaplayabilmek için denetleyicinin (3.269) ile verilen transfer fonksiyonunu kullandığımızda ve sistemin 2. derece transfer fonksiyonu

$$\frac{Y(z)}{U(z)} = \frac{B(z^{-1})}{A(z^{-1})} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{3.271}$$

şeklinde yazdığımızda kapalı çevrim sistemin transfer fonksiyonunu

$$\frac{Y(z)}{R(z)} = \frac{B(z^{-1})Q(z^{-1})}{A(z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1})} \tag{3.272}$$

olarak ve kapalı çevrim sistemin karakteristik denklemini

$$A(z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1}) = D(z^{-1}) \tag{3.273}$$

olarak yazabiliriz. Bu denklem kapalı çevrim sistemin dinamik davranışını belirleyen karakteristik polinom $D(z^{-1})$ çeşitli şekillerde belirlenebilir. Karakteristik polinomun basit bir seçimi

$$D(s) = s^2 + 2\xi\omega_n s + \omega_n^2 = 0 \tag{3.274}$$

biçimindedir. Bu polinomun sayısal eşdeğeri

$$D(z^{-1}) = 1 + d_1 z^{-1} + d_2 z^{-2} = 0 \tag{3.275}$$

biçimindedir. Bu polinomun katsayıları

$$\begin{aligned} d_1 &= -2\exp(-\xi\omega_n T)\cos(\omega_n T\sqrt{1-\xi^2}) \quad , \quad \xi \le 1 \text{ için} \\ d_1 &= -2\exp(-\xi\omega_n T)\cosh(\omega_n T\sqrt{1-\xi^2}) \quad , \quad \xi > 1 \text{ için} \\ d_2 &= \exp(-2\xi\omega_n T) \end{aligned} \tag{3.276}$$

olarak verilmiştir (Bobal ve ark. 1999,2005). Karakteristik denklem kullanılarak sayısal PID denetleyici katsayıları aşağıdaki gibi hesaplanabilir.

$$\begin{aligned} &\hat{A}(z^{-1})P(z^{-1}) + \hat{B}(z^{-1})Q(z^{-1}) = D(z^{-1}) \\ &(1+\hat{a}_1 z^{-1} + \hat{a}_2 z^{-2})(1-z^{-1})(1+\gamma z^{-1}) + (\hat{b}_1 z^{-1} + \hat{b}_2 z^{-2})(q_0 + q_1 z^{-1} + q_2 z^{-2}) \\ &= 1 + d_1 z^{-1} + d_2 z^{-2} \end{aligned} \tag{3.277}$$

$$\begin{bmatrix} \hat{b}_1 & 0 & 0 & 1 \\ \hat{b}_2 & \hat{b}_1 & 0 & \hat{a}_1 - 1 \\ 0 & \hat{b}_2 & \hat{b}_1 & \hat{a}_2 - \hat{a}_1 \\ 0 & 0 & \hat{b}_2 & -\hat{a}_2 \end{bmatrix} \cdot \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \gamma \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \gamma \end{bmatrix} = \begin{bmatrix} \hat{b}_1 & 0 & 0 & 1 \\ \hat{b}_2 & \hat{b}_1 & 0 & \hat{a}_1 - 1 \\ 0 & \hat{b}_2 & \hat{b}_1 & \hat{a}_2 - \hat{a}_1 \\ 0 & 0 & \hat{b}_2 & -\hat{a}_2 \end{bmatrix}^{-1} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{3.278}$$

Burada

$$x_1 = d_1 + 1 - \hat{a}_1 \ , \ x_2 = d_2 + \hat{a}_1 - \hat{a}_2 \ , \ x_3 = \hat{a}_2 \ , \ x_4 = 0 \tag{3.279}$$

olarak verilmektedir. Bu denklemlerdeki bilinmeyen sistem parametrelerini tahmin etmek için tekrarlamalı GS algoritmasını aşağıdaki gibi kullanabiliriz.

$$
\begin{aligned}
&\phi(k)=[-y(k-1) \quad -y(k-2) \quad u(k-1) \quad u(k-2)]^T \\
&\hat{\theta}(k-1)=\left[\hat{a}_1 \quad \hat{a}_2 \quad \hat{b}_1 \quad \hat{b}_2\right]^T \quad \text{olmak üzere} \\
&R(k)=R(k-1)+\phi(k)\phi^T(k) \quad , \quad p(k)=p(k-1)+\phi(k)y(k) \\
&\hat{\theta}_i(k)=\left[p_i(k)-\sum_{j=1}^{i-1}R_{ij}(k)\hat{\theta}_j(k)-\sum_{j=i+1}^{4}R_{ij}(k)\hat{\theta}_j(k-1)\right]\Big/ R_{ii}(k) \qquad , \quad (i=1,\ldots,4)
\end{aligned}
\tag{3.280}
$$

Çıkış işaretinin renkli ölçme gürültüsü içermesi durumunda doğru parametre tahminlerini elde edebilmek için yardımcı değişkenleri kullanan aşağıdaki tekrarlamalı GSYD algoritmasını kullanabiliriz (Hatun ve Koçal 2007).

$$
\begin{aligned}
&\phi(k)=[-y(k-1) \quad -y(k-2) \quad u(k-1) \quad u(k-2)]^T \\
&z(k)=[-\hat{x}(k-1) \quad -\hat{x}(k-2) \quad u(k-1) \quad u(k-2)]^T \\
&\hat{x}(k)=z^T(k)\hat{\theta}(k-1) \\
&\hat{\theta}(k-1)=\left[\hat{a}_1 \quad \hat{a}_2 \quad \hat{b}_1 \quad \hat{b}_2\right]^T \quad \text{olmak üzere} \\
&R(k)=R(k-1)+z(k)\phi^T(k) \quad , \quad p(k)=p(k-1)+z(k)y(k) \\
&\hat{\theta}_i(k)=\left[p_i(k)-\sum_{j=1}^{i-1}R_{ij}(k)\hat{\theta}_j(k)-\sum_{j=i+1}^{4}R_{ij}(k)\hat{\theta}_j(k-1)\right]\Big/ R_{ii}(k) \qquad , \quad (i=1,\ldots,4)
\end{aligned}
\tag{3.281}
$$

Polinomsal kutup yerleştirme yöntemiyle PID parametrelerini hesaplayabilmek için denetleyicinin (3.270) ile verilen transfer fonksiyonunu kullandığımızda sayısal PID denetleyicinin fark denklemi

$$
u(k)=q_0e(k)+q_1e(k-1)+q_2e(k-2)+(1-\gamma)u(k-1)+\gamma u(k-2) \tag{3.282}
$$

olmaktadır (Bobal ve ark. 1999,2005).

Polinomsal kutup yerleştirmeli denetleyicinin hızsal biçimdeki gösterimi

$$U(z) = \left[\beta E(z) - Q'(z^{-1})Y(z)\right]\frac{1}{P(z^{-1})} \tag{3.283}$$

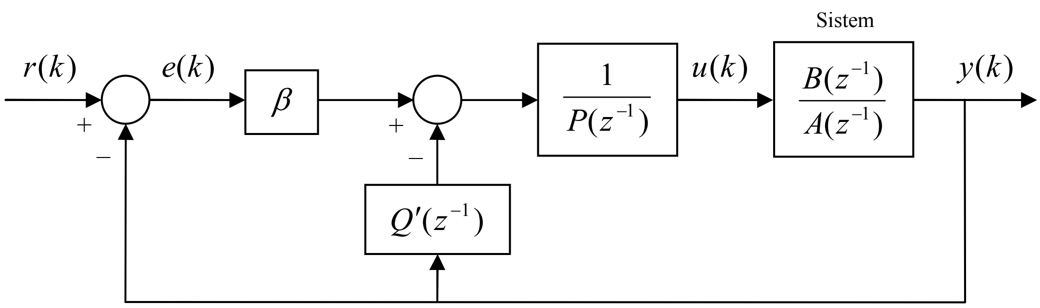
olarak verilmektedir (Ortega ve Kelly 1984, Bobal ve ark. 2005). Bu denetleyicinin pay ve payda polinomları sırasıyla aşağıdaki gibi verilmektedir.

$$Q'(z^{-1}) = (1 - z^{-1})(q'_0 - q'_2 z^{-1}) \quad , \quad P(z^{-1}) = (1 - z^{-1})(1 + \gamma\, z^{-1}) \tag{3.284}$$

Bu polinomlar denetleyici denkleminde yerine koyulduğunda sayısal PID denetleyicinin fark denklemi aşağıdaki gibi yazılabilir (Bobal ve ark. 2005).

$$\begin{aligned} u(k) = &-[(q'_0 + \beta)y(k) - (q'_0 + q'_2)y(k-1) + q'_2 y(k-2)] - (\gamma - 1)u(k-1) \\ &+ \gamma\, u(k-2) + \beta\, r(k) \end{aligned} \tag{3.285}$$

Hızsal biçimdeki sayısal PID denetleyicili kapalı çevrim sistemin blok diyagram gösterimi Şekil 3.15'te görülmektedir (Ortega ve Kelly 1984, Bobal ve ark. 2005).



Şekil 3.15. Hızsal biçim sayısal PID denetleyicili geribeslemeli kontrol sistemi.

Blok diyagramdan kapalı çevrim sistemin transfer fonksiyonu

$$\frac{Y(z)}{R(z)} = \frac{\beta\, B(z^{-1})}{A(z^{-1})P(z^{-1}) + B(z^{-1})[Q'(z^{-1}) + \beta]} \tag{3.286}$$

olarak ve kapalı çevrim sistemin karakteristik denklemini

$$A(z^{-1})P(z^{-1}) + B(z^{-1})[Q'(z^{-1}) + \beta] = D(z^{-1}) \tag{3.287}$$

olarak yazabiliriz. Bu denklem kapalı çevrim sistemin dinamik davranışını belirleyen karakteristik polinom $D(z^{-1})$ çeşitli şekillerde belirlenebilir. Karakteristik polinom (3.275) eşitliğindeki gibi seçildiğinde sayısal PID denetleyici katsayıları aşağıdaki gibi hesaplanabilir.

$$\begin{aligned} &\hat{A}(z^{-1})P(z^{-1}) + \hat{B}(z^{-1})[Q'(z^{-1}) + \beta] = D(z^{-1}) \\ &(1 + \hat{a}_1 z^{-1} + \hat{a}_2 z^{-2})(1 - z^{-1})(1 + \gamma z^{-1}) + (\hat{b}_1 z^{-1} + \hat{b}_2 z^{-2})[(1 - z^{-1})(q_0' + q_2' z^{-1}) + \beta] \\ &= 1 + d_1 z^{-1} + d_2 z^{-2} \end{aligned} \tag{3.288}$$

$$\begin{bmatrix} \hat{b}_1 & 0 & \hat{b}_1 & 1 \\ \hat{b}_2 - \hat{b}_1 & -\hat{b}_1 & \hat{b}_2 & \hat{a}_1 - 1 \\ \hat{b}_2 & \hat{b}_2 - \hat{b}_1 & 0 & \hat{a}_1 - \hat{a}_2 \\ 0 & \hat{b}_2 & 0 & -\hat{a}_2 \end{bmatrix} \cdot \begin{bmatrix} q_0' \\ q_2' \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \Rightarrow \begin{bmatrix} q_0' \\ q_2' \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} \hat{b}_1 & 0 & \hat{b}_1 & 1 \\ \hat{b}_2 - \hat{b}_1 & -\hat{b}_1 & \hat{b}_2 & \hat{a}_1 - 1 \\ \hat{b}_2 & \hat{b}_2 - \hat{b}_1 & 0 & \hat{a}_1 - \hat{a}_2 \\ 0 & \hat{b}_2 & 0 & -\hat{a}_2 \end{bmatrix}^{-1} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{3.289}$$

Burada

$$x_1 = d_1 + 1 - \hat{a}_1 \;,\; x_2 = d_2 + \hat{a}_1 - \hat{a}_2 \;,\; x_3 = -\hat{a}_2 \;,\; x_4 = 0 \tag{3.290}$$

olarak verilmektedir (Bobal ve ark. 2005). Buradaki bilinmeyen sistem parametrelerini tahmin etmek için tekrarlamalı GS algoritması kullanılabilir. Tahmin edilen sistem parametreleri konumsal veya hızsal biçimde gerçeklenebilen sayısal PID denetleyici katsayılarının güncellenmesinde kullanılır.

3.7.1.3. Model tabanlı kutup yerleştirmeli uyarlamalı denetleyici katsayılarının dolaylı olarak tekrarlamalı Gauss-Seidel algoritması ile ayarlanması

Bu kısımda polinomsal kutup yerleştirme temelli ayarlama yöntemleri üzerinde durulacaktır. Polinomsal kutup yerleştirme temelli tasarım formülasyonlardaki küçük farklılıklar haricinde temelde aynı yaklaşımlar kullanılarak birkaç değişik şekilde yapılabilmektedir (Aström ve Wittenmark 1995, Landau 1990, Landau ve Zito 2006).

Burada Johansson (1989) ile Akhtar ve Bernstein (2005) tarafından kullanılan polinomsal kutup yerleştirme işlemi kullanılacaktır.

Bir tek-girişli-tek-çıkışlı ayrık zamanlı doğrusal sistemin fark denklem modelini

$$y(k) = -\sum_{i=1}^{n} a_i y(k-i) + \sum_{j=1}^{m} b_j u(k-i) \tag{3.291}$$

olarak yazdığımızda z -domeninde

$$A(z)y(k) = B(z)u(k) \tag{3.292}$$

olarak yazılabilir. Burada transfer fonksiyonunun payda ve pay polinomları sırasıyla

$$\begin{aligned} A(z) &= z^n + a_1 z^{n-1} + \cdots + a_n \\ B(z) &= b_0 z^m + b_1 z^{m-1} + \cdots + b_m \end{aligned} \tag{3.293}$$

şeklinde tanımlanabilir. Burada n payda polinomunun derecesini, m pay polinomunun derecesini göstermektedir ve sistemin ölü zaman gecikmesi $d = n - m$ olarak tanımlanmaktadır. Kapalı çevrim sistemin davranışını kontrol etmek için kullanılan iki serbestlik dereceli genel denetleyicinin transfer fonksiyonu modeli

$$u(k) = \frac{T(z)}{R(z)} r(k) - \frac{S(z)}{R(z)} y(k) \tag{3.294}$$

olarak verilebilir, burada denetleyicinin çıkışı, yani sistemin kontrol sinyali $u(k)$, kapalı çevrim sistemin referans sinyali $r(k)$ ve çıkış sinyali $y(k)$ kullanılarak hesaplanmaktadır. Bu durumda denetleyicili kapalı çevrim sistemin transfer fonksiyonu

$$y(k) = \frac{B(z)T(z)}{A(z)R(z) + B(z)S(z)} r(k) \tag{3.295}$$

olarak yazılabilir. Kapalı çevrim sistemin çıkış işaretinin, transfer fonksiyonu modeli

$$y_m(k) = \frac{B_m(z)}{A_m(z)} r(k) \tag{3.296}$$

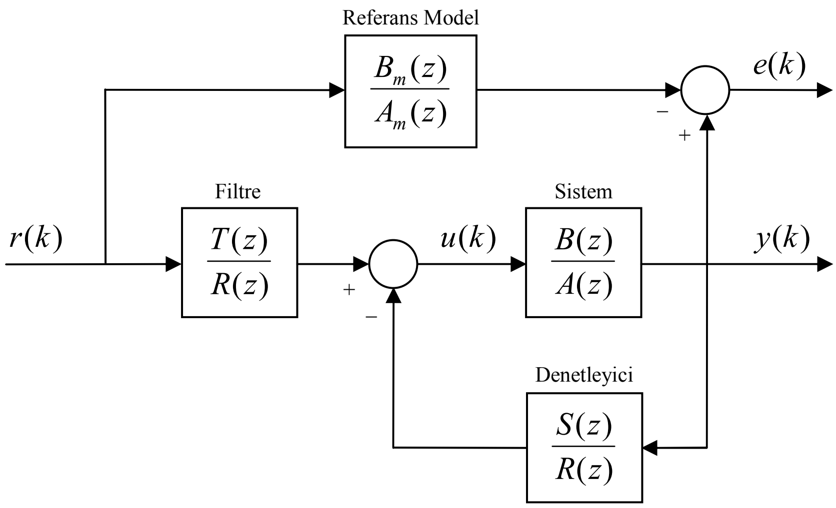
olarak verilen referans modelin çıkışını takip etmesi istenir. Aradaki fark denetleyici parametrelerini ayarlamak amacıyla hata sinyali olarak kullanılır. Şekil 3.16'da bu amaçla kullanılan referans modelli kapalı çevrim kontrol sistemi görülmektedir. Burada kapalı çevrim sistem (3.295) ile referans model (3.296)

$$\frac{B(z)T(z)}{A(z)R(z)+B(z)S(z)} = \frac{B_m(z)}{A_m(z)} \tag{3.297}$$

olması durumunda aynı referans sinyali için aynı cevabı verir. Aynı eşitlik

$$\frac{T(z)}{A(z)R(z)+B(z)S(z)} = \frac{B_m(z)}{A_m(z)B(z)} \tag{3.298}$$

olarak da yazılabilir.



Şekil 3.16. Referans modelli kapalı çevrim kontrol sistemi.

Kapalı çevrim sistemin karakteristik denklemi (3.298) eşitliğinden

$$A(z)R(z) + B(z)S(z) = A_m(z)B(z) = P(z) \tag{3.299}$$

olarak yazıldığında

$$T(z) = B_m(z) \tag{3.300}$$

olarak seçilmektedir. Burada denetleyici polinomları denetleyicinin nedensel ve minimum dereceli olabilmesi için ve (3.299) ile verilen eşitliğin çözümünün tek olabilmesi için

$$R(z) = r_0 z^{n-1} + r_1 z^{n-2} + \cdots + r_{n-1} \tag{3.301}$$

$$S(z) = s_0 z^{n-1} + s_1 z^{n-2} + \cdots + s_{n-1} \tag{3.302}$$

biçiminde olmalıdır (Akhtar ve Bernstein 2005). Burada bilinmeyen sistem parametrelerinin tahmin edilmesi durumunda, (3.299) eşitliği yardımıyla denetleyici katsayıları aşağıdaki gibi hesaplanabilir.

$$\mathrm{M}.\begin{bmatrix} \theta_R \\ \theta_S \end{bmatrix} = \theta_P \quad \Rightarrow \quad \begin{bmatrix} \theta_R \\ \theta_S \end{bmatrix} = \mathrm{M}^{-1}\theta_P \tag{3.303}$$

Bu şekilde yapılan uyarlamalı kontrol işlemi dolaylı uyarlamalı kontrol olarak bilinmektedir. Burada M matrisi sistem parametrelerini içermektedir ve yapısı en genel halde aşağıdaki gibidir.

$$\mathrm{M} = \begin{bmatrix} 1 & 0 & \cdots & 0_{(n-1)\times 1} & 0_{d\times 1} & 0_{(d+1)\times 1} & \cdots & 0_{(n+d-1)\times 1} \\ a_1 & 1 & \cdots & 1 & b_0 & b_0 & \cdots & b_0 \\ a_2 & a_1 & \cdots & a_1 & b_1 & b_1 & \cdots & b_1 \\ \vdots & \vdots & \ddots & a_2 & \vdots & \vdots & \ddots & b_2 \\ a_n & a_n & & \vdots & b_m & b_m & & \vdots \\ 0_{(n-1)\times 1} & 0_{(n-2)\times 1} & \cdots & a_n & 0_{(n-1)\times 1} & 0_{(n-2)\times 1} & \cdots & b_m \end{bmatrix} \tag{3.304}$$

Ayrıca θ_R, θ_S ile sırasıyla denetleyicinin $R(z^{-1})$, $S(z^{-1})$ polinomlarının katsayıları ve θ_P ile $P(z^{-1})$ karakteristik denklemin katsayıları gösterilmektedir. Bu katsayı vektörleri aşağıda verilmiştir.

$$\theta_R = [\, r_0 \quad r_1 \quad \cdots \quad r_{n_R} \,]^T \tag{3.305}$$

$$\theta_S = [\, s_0 \quad s_1 \quad \cdots \quad s_{n_S} \,]^T \tag{3.306}$$

$$\theta_P = [\, b_0 \quad p_1 \quad \cdots \quad p_{n_m+m} \,]^T \tag{3.307}$$

Burada n_R ve n_S sırasıyla $R(z^{-1})$, $S(z^{-1})$ polinomlarının derecesini gösterir ve n_m ise $A_m(z^{-1})$ polinomunun derecesini gösterir. Karakteristik denklem ise

$$P(z) = A_m(z)B(z) = b_0 z^{(n_m+m)} + p_1 z^{(n_m+m)-1} + \cdots + p_{(n_m+m)} \tag{3.308}$$

olarak tanımlanmıştır. Burada θ_R ve θ_S katsayılarının hesaplanmasında (3.303) denkleminin çözümünün tek olması için $n_m = 2n - m - 1$ ve $n_S = n - 1$ olmalıdır. Ayrıca $n_R \geq n_S = n_T$ olduğunda denetleyici nedensel olmaktadır (Akhtar ve Bernstein 2005).

Ayrık-zaman sistemin transfer fonksiyonu parametrelerini çevrim-içi yöntemle tahmin etmek için tekrarlamalı GS algoritması aşağıdaki gibi kullanılabilir. Burada sisteme ait veri vektörü ve parametre tahmin vektörü

$$\phi(k) = \left[-y(k-1) \quad \cdots \quad -y(k-n) \quad u(k-1) \quad \cdots \quad u(k-m) \right]^T \tag{3.309}$$

$$\hat{\theta}(k) = \left[\hat{a}_1 \quad \cdots \quad \hat{a}_n \quad \hat{b}_1 \quad \cdots \quad \hat{b}_m \right]^T \tag{3.310}$$

olmak üzere tekrarlamalı GS algoritması $(i = 1,2,\ldots,n+m)$ için

$$R(k) = R(k-1) + \phi(k)\phi^T(k) \quad , \quad p(k) = p(k-1) + \phi(k)y(k) \tag{3.311}$$

$$\hat{\theta}_i(k) = \left[p_i(k) - \sum_{j=1}^{i-1} R_{ij}(k)\hat{\theta}_j(k) - \sum_{j=i+1}^{n+m} R_{ij}(k)\hat{\theta}_j(k-1) \right] \Big/ R_{ii}(k) \tag{3.312}$$

şeklinde kullanılabilir. Burada tahmin edilen transfer fonksiyonu katsayıları model tabanlı uyarlamalı denetleyici katsayılarının dolaylı olarak ayarlanmasında kullanılabilir.

3.7.2. Tekrarlamalı Gauss-Seidel algoritması ile doğrudan uyarlamalı kontrol

Bu kısımda sırasıyla model-tabanlı dolaylı uyarlamalı denetleyici parametrelerinin ve kendinden ayarlamalı uyarlamalı denetleyici parametrelerinin tekrarlamalı GS algoritmasıyla doğrudan ayarlanması işlemi anlatılacaktır.

3.7.2.1. Tekrarlamalı Gauss-Seidel algoritması ile doğrudan model tabanlı uyarlamalı kontrol

Polinomsal kutup yerleştirme temelli denetleyici tasarımı formülasyonlardaki küçük farklılıklar haricinde temelde aynı yaklaşımlar kullanılarak birkaç değişik şekilde yapılabilmektedir (Aström ve Wittenmark 1995, Landau 1990, Landau ve ark. 1998, Landau ve Zito 2006). Burada Johansson (1989) ile Akhtar ve Bernstein (2005) tarafından kullanılan polinomsal kutup yerleştirmeli denetleyici kullanılacaktır.

Uyarlamalı denetleyici katsayılarını doğrudan ayarlayabilmek için filtrelenmiş çıkış ve filtrelenmiş hata sinyalleri yardımıyla denetleyici parametrelerine bağlı bir doğrusal tahmin modeli kullanılır. Bu amaçla filtrelenmiş çıkış işareti

$$y_f(k) = z^{-n-d+1} A_m(z) y(k) \tag{3.313}$$

olarak tanımlanmıştır. Burada sistemin transfer fonksiyonu kullanarak aynı denklem

$$y_f(k) = \frac{z^{-n-d+1} A_m(z) B(z)}{A(z)} u(k) \tag{3.314}$$

olarak yazılabilir. Bu eşitlikte karakteristik denklem kullanıldığında

$$\begin{aligned} y_f(k+d) &= \frac{z^{-n+1}[A(z)R(z) + B(z)S(z)]}{A(z)} u(k) \\ &= \left(R(z) + S(z)\frac{B(z)}{A(z)} \right) u(k-n+1) \\ &= R(z)u(k-n+1) + S(z)y(k-n+1) \end{aligned} \tag{3.315}$$

eşitliği elde edilir. Burada $r_0 = b_0$ olduğunu da göz önüne alarak elde edilen (3.315) denklemini doğrusal bağlaşımlı biçimde

$$y_f(k+d) = b_0 u(k) + \phi^T(k)\theta \tag{3.316}$$

olarak yazdığımızda buradaki denetleyici katsayı vektörünü ve veri vektörünü sırasıyla

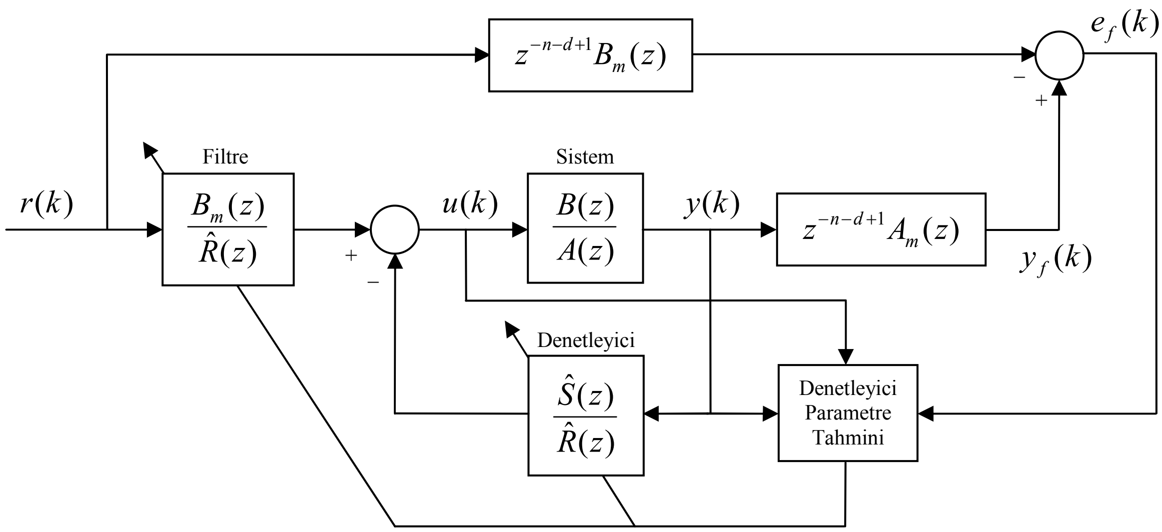
$$\theta = \begin{bmatrix} r_1 & \cdots & r_{n-1} & s_0 & \cdots & s_{n-1} \end{bmatrix}^T_{(2n-1)\times 1} \tag{3.317}$$

$$\phi(k) = \begin{bmatrix} u(k-1) & \cdots & u(k-n+1) & y(k) & \cdots & y(k-n+1) \end{bmatrix}^T_{(2n-1)\times 1} \tag{3.318}$$

olarak yazabiliriz. Buradan (3.315) ve (3.316) eşitlikleri kullanılarak (3.294) eşitliğindeki model tabanlı kontrol kuralını, yani sistemi kontrol etmek için kullanılan kontrol sinyalini

$$u(k) = -\frac{1}{b_0}\left[\phi^T(k)\hat{\theta}(k) - z^{-n+1} B_m(z) r(k) \right] \tag{3.319}$$

şeklinde en son güncellenen denetleyici parametrelerini kullanarak hesaplayabiliriz. Filtrelenmiş çıkış ve hata sinyallerini kullanarak ayarlanan model tabanlı uyarlamalı kontrol sistemi Şekil 3.17'de görülmektedir.



Şekil 3.17. Model tabanlı uyarlamalı denetleyicinin doğrudan ayarlanması.

Model tabanlı uyarlamalı denetleyici katsayılarını tekrarlamalı (recursive, on-line) olarak güncellemek için kullanılan tekrarlamalı GS algoritmasını aşağıdaki gibi yazabiliriz.

Tekrarlamalı GS Algoritması: (b_0'ın bilinmesi durumunda)

$$\phi(k-d)=[\,u(k-d-1)\ \cdots\ u(k-d-n+1)\ \ y(k-d)\ \cdots\ y(k-d-n+1)]^T_{\ (2n-1)\times 1}$$
$$\phi(k)=[\,u(k-1)\ \cdots\ u(k-n+1)\ \ y(k)\ \cdots\ y(k-n+1)]^T_{\ (2n-1)\times 1}$$
$$\hat{\theta}(k)=\left[\hat{r}_1\ \cdots\ \hat{r}_{n-1}\ \ \hat{s}_0\ \cdots\ \hat{s}_{n-1}\right]^T_{\ (2n-1)\times 1} \quad \text{olmak üzere}$$
$$R(k)=R(k-1)+\phi(k-d)\phi^T(k-d)\quad,\quad r(k)=r(k-1)+\phi(k-d)\big(y_f(k)+b_0u(k-d)\big) \tag{3.320}$$
$$\hat{\theta}_i(k)=\left[r_i(k)-\sum_{j=1}^{i-1}R_{ij}(k)\hat{\theta}_j(k)-\sum_{j=i+1}^{2n-1}R_{ij}(k)\hat{\theta}_j(k-1)\right]\Big/R_{ii}(k)\qquad,\quad (i=1,2,\ldots,2n-1)$$
$$u(k)=-\frac{1}{b_0}\left[\phi^T(k)\hat{\theta}(k)-z^{-n+1}B_m(z)r(k)\right]\ \rightarrow\ \text{kontrol kuralı}$$

Bu algoritma $b_0=r_0$ katsayısının bilinmesi durumunda kullanılabilir, bilinmemesi durumunda hesaplanan parametre sayısı bir fazla olacaktır. Bu durumda yukarıdaki tekrarlamalı GS algoritmasını şu şekilde yazabiliriz:

Tekrarlamalı GS Algoritması : (b_0 'ın tahmin edilmesi durumunda)

$$
\begin{aligned}
&\phi(k-d) = [\,u(k-d) \;\; \cdots \;\; u(k-d-n+1) \;\; y(k-d) \;\; \cdots \;\; y(k-d-n+1)]^T{}_{(2n)\times 1} \\
&\bar{\phi}(k) = [\,u(k-1) \;\; \cdots \;\; u(k-n+1) \;\; y(k) \;\; \cdots \;\; y(k-n+1)]^T{}_{(2n-1)\times 1} \\
&\hat{\theta}(k) = \left[\hat{r}_0 \;\; \cdots \;\; \hat{r}_{n-1} \;\; \hat{s}_0 \;\; \cdots \;\; \hat{s}_{n-1}\right]^T{}_{(2n)\times 1} \\
&\hat{\bar{\theta}}(k) = \left[\hat{r}_1 \;\; \cdots \;\; \hat{r}_{n-1} \;\; \hat{s}_0 \;\; \cdots \;\; \hat{s}_{n-1}\right]^T{}_{(2n-1)\times 1} \quad \text{olmak üzere} \\
&R(k) = R(k-1) + \phi(k-d)\phi^T(k-d) \quad , \quad r(k) = r(k-1) + \phi(k-d)\, y_f(k) \\
&\hat{\theta}_i(k) = \left[r_i(k) - \sum_{j=1}^{i-1} R_{ij}(k)\hat{\theta}_j(k) - \sum_{j=i+1}^{2n} R_{ij}(k)\hat{\theta}_j(k-1) \right] \Big/ R_{ii}(k) \quad , \quad (i = 1,2,\ldots,2n) \\
&u(k) = -\frac{1}{\hat{b}_0(k)}\left[\bar{\phi}^T(k)\hat{\bar{\theta}}(k) - z^{-n+1}B_m(z)r(k)\right] \;\rightarrow\; \text{kontrol kuralı}
\end{aligned}
\tag{3.321}
$$

Model tabanlı uyarlamalı denetleyici katsayılarını güncellemek için kullanılabilecek olan RLS ve NLMS algoritmaları aşağıdaki gibi verilebilir (Akhtar ve Bernstein 2005).

RLS Algoritması: (b_0 'ın bilinmesi durumunda)

$$
\begin{aligned}
&\phi(k-d) = [\,u(k-d-1) \;\; \cdots \;\; u(k-d-n+1) \;\; y(k-d) \;\; \cdots \;\; y(k-d-n+1)]^T{}_{(2n-1)\times 1} \\
&\phi(k) = [\,u(k-1) \;\; \cdots \;\; u(k-n+1) \;\; y(k) \;\; \cdots \;\; y(k-n+1)]^T{}_{(2n-1)\times 1} \\
&\hat{\theta}(k) = \left[\hat{r}_1 \;\; \cdots \;\; \hat{r}_{n-1} \;\; \hat{s}_0 \;\; \cdots \;\; \hat{s}_{n-1}\right]^T{}_{(2n-1)\times 1} \quad \text{olmak üzere} \\
&e_f(k) = y_f(k) - b_0 u(k-d) - \phi^T(k-d)\,\hat{\theta}(k-1) \\
&P(k) = P(k-1) + \frac{P(k-1)\phi(k-d)\phi^T(k-d)P(k-1)}{1+\phi^T(k-d)P(k-1)\phi(k-d)} \\
&\hat{\theta}(k) = \hat{\theta}(k-1) + P(k)\phi(k-d)e_f(k) \\
&u(k) = -\frac{1}{b_0}\left[\phi^T(k)\hat{\theta}(k) - z^{-n+1}B_m(z)r(k)\right] \;\rightarrow\; \text{kontrol kuralı}
\end{aligned}
\tag{3.322}
$$

RLS Algoritması: (b_0'ın tahmin edilmesi durumunda)

$$\begin{aligned}
&\phi(k-d)=[\,u(k-d) \;\; \cdots \;\; u(k-d-n+1) \;\; y(k-d) \;\; \cdots \;\; y(k-d-n+1)]^T{}_{(2n)\times 1} \\
&\overline{\phi}(k)=[\,u(k-1) \;\; \cdots \;\; u(k-n+1) \;\; y(k) \;\; \cdots \;\; y(k-n+1)]^T{}_{(2n-1)\times 1} \\
&\hat{\theta}(k)=\left[\hat{r}_0 \;\; \cdots \;\; \hat{r}_{n-1} \;\; \hat{s}_0 \;\; \cdots \;\; \hat{s}_{n-1}\right]^T{}_{(2n)\times 1} \\
&\hat{\overline{\theta}}(k)=\left[\hat{r}_1 \;\; \cdots \;\; \hat{r}_{n-1} \;\; \hat{s}_0 \;\; \cdots \;\; \hat{s}_{n-1}\right]^T{}_{(2n-1)\times 1} \quad \text{olmak üzere} \\
&e_f(k)=y_f(k)-\phi^T(k-d)\,\hat{\theta}(k-1) \\
&P(k)=P(k-1)+\frac{P(k-1)\phi(k-d)\phi^T(k-d)P(k-1)}{1+\phi^T(k-d)P(k-1)\phi(k-d)} \\
&\hat{\theta}(k)=\hat{\theta}(k-1)+P(k)\phi(k-d)e_f(k) \\
&u(k)=-\frac{1}{\hat{b}_0(k)}\left[\overline{\phi}^T(k)\hat{\overline{\theta}}(k)-z^{-n+1}B_m(z)r(k)\right] \;\rightarrow\; \text{kontrol kuralı}
\end{aligned} \tag{3.323}$$

NLMS Algoritması: (b_0'ın bilinmesi durumunda)

$$\begin{aligned}
&\phi(k-d)=[\,u(k-d-1) \;\; \cdots \;\; u(k-d-n+1) \;\; y(k-d) \;\; \cdots \;\; y(k-d-n+1)]^T{}_{(2n-1)\times 1} \\
&\phi(k)=[\,u(k-1) \;\; \cdots \;\; u(k-n+1) \;\; y(k) \;\; \cdots \;\; y(k-n+1)]^T{}_{(2n-1)\times 1} \\
&\hat{\theta}(k)=\left[\hat{r}_1 \;\; \cdots \;\; \hat{r}_{n-1} \;\; \hat{s}_0 \;\; \cdots \;\; \hat{s}_{n-1}\right]^T{}_{(2n-1)\times 1} \quad \text{olmak üzere} \\
&e_f(k)=y_f(k)-b_0u(k-d)-\phi^T(k-d)\,\hat{\theta}(k-1) \\
&\hat{\theta}(k)=\hat{\theta}(k-1)+\frac{\mu\phi(k-d)e_f(k)}{\alpha+\phi^T(k-d)\phi(k-d)} \\
&u(k)=-\frac{1}{b_0}\left[\phi^T(k)\hat{\theta}(k)-z^{-n+1}B_m(z)r(k)\right] \;\rightarrow\; \text{kontrol kuralı}
\end{aligned} \tag{3.324}$$

<u>NLMS Algoritması:</u> (b_0'ın tahmin edilmesi durumunda)

$$\begin{aligned}
&\phi(k-d)=[\,u(k-d) \;\; \cdots \;\; u(k-d-n+1) \;\; y(k-d) \;\; \cdots \;\; y(k-d-n+1)]^T{}_{(2n)\times 1} \\
&\overline{\phi}(k)=[\,u(k-1) \;\; \cdots \;\; u(k-n+1) \;\; y(k) \;\; \cdots \;\; y(k-n+1)]^T{}_{(2n-1)\times 1} \\
&\hat{\theta}(k)=\left[\hat{r}_0 \;\; \cdots \;\; \hat{r}_{n-1} \;\; \hat{s}_0 \;\; \cdots \;\; \hat{s}_{n-1}\right]^T{}_{(2n)\times 1} \\
&\hat{\overline{\theta}}(k)=\left[\hat{r}_1 \;\; \cdots \;\; \hat{r}_{n-1} \;\; \hat{s}_0 \;\; \cdots \;\; \hat{s}_{n-1}\right]^T{}_{(2n-1)\times 1} \quad \text{olmak üzere} \\
&e_f(k)=y_f(k)-\phi^T(k-d)\hat{\theta}(k-1) \\
&\hat{\theta}(k)=\hat{\theta}(k-1)+\frac{\mu\phi(k-d)e_f(k)}{\alpha+\phi^T(k-d)\phi(k-d)} \\
&u(k)=-\frac{1}{\hat{b}_0(k)}\left[\overline{\phi}^T(k)\hat{\overline{\theta}}(k)-z^{-n+1}B_m(z)r(k)\right] \;\rightarrow\; \text{kontrol kuralı}
\end{aligned} \tag{3.325}$$

3.7.2.2. Tekrarlamalı Gauss-Seidel algoritması ile doğrudan kendinden ayarlamalı uyarlamalı kontrol

Polinomsal kutup yerleştirme yöntemiyle ve tahmin (prediction) yöntemiyle tasarlanabilen kendinden ayarlamalı uyarlamalı denetleyici parametrelerinin dolaylı olarak veya doğrudan ayarlanmasında da yine RLS ve RLS tabanlı algoritmalar kullanılmaktadır (Aström ve Wittenmark 1973,1980,1995, Aström ve ark. 1977, Aström 1983,1987, Harris ve Billings 1985, Seborg ve ark. 1986, Warwick 1988, Wellstead ve Zarrop 1991, Isermann ve ark. 1992, Landau ve ark. 1998, Patete ve ark. 2006,2007,2008a,2008b).

Bu kısımda, ayrık-zaman deterministik sistemleri kontrol etmek için, GMV (Genelleştirilmiş Minimum Varyans) kontrol yöntemi üzerine kurulu olan bir uyarlamalı denetleyici kullanılmıştır (Patete ve ark. 2006,2007,2008a,2008b). Burada öncelikle GMV kontrol yöntemi anlatılacaktır. Ayrık-zaman tek-girişli tek-çıkışlı zamanla değişmeyen bir deterministik sistemin fark denklem modeli

$$A(z^{-1})y(k)=z^{-d}B(z^{-1})u(k) \tag{3.326}$$

olarak yazılabilir. Burada sistemin transfer fonksiyonunun payda polinomu $A(z^{-1})$ ile pay polinomu $B(z^{-1})$ arasında ortak çarpan olmadığı ve sistemin zaman gecikmesi d 'nin bilindiği varsayılmıştır. Başlangıçta, payda ve pay polinomlarının

$$A(z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_n z^{-n} \tag{3.327}$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + \cdots + b_m z^{-m} \ , b_0 \neq 0 \tag{3.328}$$

olarak tanımlandığı ve katsayılarının bilindiği varsayılmıştır. Bu durumda GMV kontrol yönteminin amacı, deterministik sistemler için

$$s(k+d) = C(z^{-1})[y(k+d) - r(k+d)] + Q(z^{-1})u(k) \tag{3.329}$$

olarak tanımlanan genelleştirilmiş çıkış ifadesinin varyansını minimum yapmaktır (Patete ve ark. 2006, 2007,2008a,2008b). Bu denklemindeki polinomlar

$$C(z^{-1}) = 1 + c_1 z^{-1} + \cdots + c_n z^{-n} \ , \ Q(z^{-1}) = q_0(1 - z^{-1}) \tag{3.330}$$

olarak tanımlanmıştır. Burada $r(k)$ referans işareti olup, (3.329) denklemindeki ilk terim $e(k) = y(k) - r(k)$ olarak tanımlanan hata terimiyle ilişkilidir ve ikinci terim ise tersi kararsız olan, yani minimum fazlı olmayan sistemlerin sıfırlarıyla denetleyicinin kutupları arasındaki olası kutup-sıfır sadeleşmelerinden kaçınmak için kullanılmaktadır (Clarke ve Gawthrop 1975,1979, Gawthrop 1980, Clarke 1984, Furuta 1993, Patete ve ark. 2006,2007,2008a,2008b). Aynı zamanda denetleyiciye integral etkisi kazandırdığı için sabit bozucu girişin sistem çıkışındaki olumsuz etkisini de ortadan kaldırır (Goodwin ve Sin 1984, Harris ve Billings 1985, Wellstead ve Zarrop 1991, Aström ve Wittenmark 1995, Landau ve ark. 1998). Burada (3.329) denklemindeki ifadenin sabit $u(k)$ giriş değerleri için $s(k+d) = 0$ olabilmesi $C(z^{-1})$ polinomunun bütün köklerinin z-düzleminde birim çember içinde olacak şekilde seçilmesi seçilir. Burada

$$G(z^{-1}) = E(z^{-1})B(z^{-1}) + Q(z^{-1}) \tag{3.331}$$

olarak tanımlandığında (3.329) denklemi

$$s(k+d) = G(z^{-1})u(k) + F(z^{-1})y(k) - C(z^{-1})r(k+d) \tag{3.332}$$

şeklinde yazılabilir. Burada $E(z^{-1})$ ve $F(z^{-1})$ polinomları

$$C(z^{-1}) = A(z^{-1})E(z^{-1}) + z^{-d}F(z^{-1}) \tag{3.333}$$

olarak verilen Diophantine eşitliğini sağlar. Bu durumda (3.333) denkleminde $s(k+d) = 0$ olabilmesi için kontrol kuralı

$$u(k) = -G(z^{-1})^{-1}[F(z^{-1})y(k) - C(z^{-1})r(k+d)] \tag{3.334}$$

olmalıdır. Bu kontrol kuralındaki denetleyici polinomları

$$F(z^{-1}) = f_0 + f_1 z^{-1} + \cdots + f_{n-1} z^{-n+1} \tag{3.335}$$
$$G(z^{-1}) = g_0 + g_1 z^{-1} + \cdots + g_{m+d-1} z^{-(m+d-1)} \tag{3.336}$$

olarak verilmektedir (Patete ve ark. 2006,2007,2008a,2008b). Bu durumda (3.334) ile hesaplanan kontrol işaretiyle $s(k+d) = 0$ olabilmesi için

$$G_d(z^{-1}) = A(z^{-1})Q(z^{-1}) + B(z^{-1})C(z^{-1}) \tag{3.337}$$

olarak yazılan karakteristik polinomunun bütün köklerinin z-düzleminde birim daire içerisinde olması gerekir (Patete ve ark. 2007,2008a,2008b). Burada sistemin polinomlarının derecesi n, m ve zaman gecikmesi d'nin bilinmesi, $C(z^{-1})$ polinomunun ve $G_d(z^{-1})$ polinomlarının köklerinin birim daire içinde olması ve $r(k)$ referans girişinin sınırlı olması varsayımlarıyla denetleyici parametreleri tekrarlamalı biçimde tahmin edildiğinde kapalı-çevrim sistemin kararlılığı sağlanmaktadır (Patete ve ark. 2007,2008a,2008b).

Sistem parametrelerinin değerlerinin tam olarak bilinmemesi durumunda denetleyici polinomlarının parametreleri yerine tahmin edilmiş değerleri

$$\hat{F}(z^{-1}) = \hat{f}_0 + \hat{f}_1 z^{-1} + \cdots + \hat{f}_{n-1} z^{-n+1} \tag{3.338}$$

$$\hat{G}(z^{-1}) = \hat{g}_0 + \hat{g}_1 z^{-1} + \cdots + \hat{g}_{m+d-1} z^{-(m+d-1)} \tag{3.339}$$

şeklinde kullanılabilir. Bu durumda GMV kontrol yöntemi üzerine kurulu olan uyarlamalı kontrol yöntemi tekrarlamalı RLS algoritması ile birlikte aşağıda gibi verilmiştir (Patete ve ark. 2007,2008a,2008b).

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{P(k-1)\phi(k-d)}{\lambda + \phi^T(k-d)P(k-1)\phi(k-d)} \cdot [s(k) + C(z^{-1}) - \phi^T(k-d)\hat{\theta}(k-1)] \tag{3.340}$$

$$P(k) = \frac{1}{\lambda}\left(P(k-1) + \frac{P(k-1)\phi(k-d)\phi^T(k-d)P(k-1)}{\lambda + \phi^T(k-d)P(k-1)\phi(k-d)} \right) \tag{3.341}$$

Burada kullanılan giriş-çıkış veri vektörü ve tahmin edilen denetleyici parametre vektörü

$$\phi^T(k) = [\, y(k)\, y(k-1) \cdots y(k-n+1)\; u(k) \cdots u(k-m+d-1)] \tag{3.342}$$

$$\hat{\theta}^T(k) = [\; \hat{f}_0 \;\; \hat{f}_1 \; \cdots \; \hat{f}_{n-1} \;\; \hat{g}_0 \; \cdots \; \hat{g}_{m+d-1}] \tag{3.343}$$

olarak verilmektedir. Bu durumda tahmin edilen parametreler (3.334) ile verilen GMV kontrol kuralında

$$u(k) = -\hat{G}(z^{-1})^{-1}[\hat{F}(z^{-1})y(k) - C(z^{-1})r(k+d)] \tag{3.344}$$

şeklinde kullanılmaktadır (Patete ve ark. 2007,2008a,2008b). Kontrol edilen sistemin minimum fazlı olması durumunda GMV kontrol yönteminde $Q(z^{-1}) = 0$, yani $q_0 = 0$ alınabilir. Bu durumda minimum varyans kontrol yöntemi elde edilir (Clarke ve Gawthrop 1975,1979, Gawthrop 1980, Clarke 1984, Wellstead ve Zarrop 1991, Landau ve ark. 1998, Patete ve ark. 2007,2008a,2008b).

Sistem tanıma algoritmaları ile aşağıdaki ortalama karesel hata fonksiyonunu minimum yapılmaktadır (Ljung ve Söderström 1983, Söderström ve Stoica 1989, Ljung 1999).

$$V_k(\theta) = \frac{1}{k}\sum_{i=1}^{k} \lambda^{k-i} \varepsilon^2(i) \tag{3.345}$$

Bu fonksiyonu minimum yapan optimum parametre tahminleri

$$\hat{\theta}_{opt}(k) = R(k)^{-1} p(k) \tag{3.346}$$

olarak yazılabilir. Burada $\hat{\theta}_{opt}(k)$ ile k adet veri kullanılarak elde edilen optimum parametre tahminleri, M ile tahmin edilen parametre sayısı, $R(k)$ ile $(M \times M)$ boyutlu korelasyon matrisi, $p(k)$ ile $(M \times 1)$ boyutlu korelasyon vektörü gösterilmektedir. Pratik uygulamalarda $R(k)$ matrisinin ve $p(k)$ vektörünün değerleri k adet veri kullanıldığında aşağıdaki gibi hesaplanabilir,

$$R(k) = \frac{1}{k}\sum_{i=1}^{k} \lambda^{k-i} \phi(i)\phi^T(i) \quad , \quad p(k) = \frac{1}{k}\sum_{i=1}^{k} \lambda^{k-i} \phi(i) y(i) \tag{3.347}$$

veya $1/k$ çarpanları göz önüne alınmadan giriş-çıkış verileri alındıkça ardışık olarak

$$R(k) = \lambda\, R(k-1) + \phi(k)\phi^T(k) \quad , \quad p(k) = \lambda\, p(k-1) + \phi(k) d(k) \tag{3.348}$$

şeklinde güncellenebilir (Ljung ve Söderström 1983, Söderström ve Stoica 1989, Ljung 1999). Burada $d(k)$ parametreleri güncellenen uyarlamalı modelin takip ettiği işarettir. GS algoritmasıyla tekrarlamalı parametre tahmin işleminin başlangıç noktası, zaman ortalamalı normal denklemin GS iterasyonları ile çözümü üzerine kuruludur. Bu amaçla bir adımlık GS iterasyonu

$$R(k)\hat{\theta}(k) = p(k) \tag{3.349}$$

olarak verilen zaman ortalamalı normal denklemin çözümünde kullanılmaktadır (Koçal 1998). Yukarıda verilen RLS algoritmasında denetleyici parametrelerini ayarlamak için kullanılan tahmin hatası (3.332) eşitliği kullanılarak

$$\varepsilon(k) = s(k) + C(z^{-1})r(k+d) - \phi^T(k-d)\hat{\theta}(k-1) \tag{3.350}$$

şeklinde yazılabilir. Bu ifade (3.329) eşitliği kullanılarak

$$\varepsilon(k) = C(z^{-1})y(k) + Q(z^{-1})u(k-d) - \phi^T(k-d)\hat{\theta}(k-1) \tag{3.351}$$

olarak da yazılabilir. Minimum-fazlı sistemler için istendiğinde $Q(z^{-1}) = 0$ alınabilir. Bu durumda tahmin hatası

$$\varepsilon(k) = C(z^{-1})y(k) - \phi^T(k-d)\hat{\theta}(k-1) \tag{3.352}$$

olur. Burada uyarlamalı denetleyici parametrelerini ayarlamak için kullanılan ve (3.351) ile verilen parametre tahmin hatasında, (3.329) denklemindeki $e(k) = y(k) - r(k)$ hata bilgisinin filtrelenmiş bir şekli kullanılır (Goodwin ve Sin 1984, Landau ve ark. 1998). Yani RLS algoritmasındaki anlık parametre tahmin hatası (3.351) veya (3.352) kullanılarak hesaplanabilir. Burada genelleştirilmiş çıkışı tahmin etme (prediction) işlemi, tahmin edilen denetleyici parametreleri yardımıyla $C(z^{-1})y(k) + Q(z^{-1})u(k-d)$ işaretini tahmin etme işlemine indirgenmiştir (Clarke ve Gawthrop 1975,1979, Gawthrop 1980, Clarke 1984, Goodwin ve Sin 1984, Landau ve ark. 1998). Minimum fazlı sistemlerde $Q(z^{-1}) = 0$ olarak alındığında tahmin edilen parametreler yardımıyla $C(z^{-1})y(k)$ işareti tahmin edilir. Tahmin edilen bu işaret tekrarlamalı GS algoritmasıyla denetleyici parametrelerini ayarlamak için (3.348) eşitliğinde kullanılabilir.

Model-tabanlı uyarlamalı denetleyici parametrelerini ayarlamak için kullanılan tekrarlamalı GS algoritması aşağıdaki gibi verilebilir.

$$R(k) = \lambda\, R(k-1) + \phi(k)\phi^T(k) \tag{3.353}$$

$$p(k) = \lambda\, p(k-1) + \phi(k)[C(z^{-1})y(k) + Q(z^{-1})u(k)] \tag{3.354}$$

$$\hat{\theta}_i(k) = \left[p_i(k) - \sum_{j=1}^{i-1} R_{ij}(k)\hat{\theta}_j(k) - \sum_{j=i+1}^{m+n+d-2} R_{ij}(k)\hat{\theta}_j(k-1) \right] \Big/ R_{ii}(k)$$
$$i = 1, 2, \ldots, (m+n+d-2) \quad , \quad k = 1, 2, \ldots \tag{3.355}$$

Bu algoritmada iterasyon indisi ayrık zaman indisi olarak alınmıştır (Koçal 1998). Burada $R_{ij}(k)$ korelasyon matrisinin i. satırına ve j. sütununa denk düşen elemanını gösterir, $p_i(k)$ korelasyon vektörünün i. elemanını gösterir ve $\hat{\theta}_i(k)$ parametre vektörünün i. elemanını gösterir. Bu algoritmada kullanılan veri vektörü (3.342) ile, güncellenen parametre vektörü ise (3.343) ile verilmiştir. Burada verilen tekrarlamalı GS algoritmasıyla ayarlanan denetleyici parametreleri, (3.351) veya (3.352) ile verilen parametre tahmin hatasının oluşturduğu ortalama karesel hata fonksiyonunu minimum yapmaktadır.

Tekrarlamalı GS algoritmasının RLS algoritmasına göre en önemli farkı korelasyon matrisinin tersi yerine doğrudan kendisinin güncellenmesi, yani matris tersini güncellemek için kullanılan iterasyona gerek kalmaması ve parametre tahminlerini güncellemek için anlık parametre tahmin hatasını doğrudan kullanmamasıdır. Parametrelerin skaler olarak güncellendiğini de göz önüne aldığımızda, RLS algoritmasına göre işlem yükünün önemli ölçüde azaldığı görülmektedir (Koçal 1998, Bose ve Xu 2002, Bose 2004). RLS algoritması ile denetleyici parametrelerini güncellemek için $3M^2 + 11M + 8$ tane çarpma ve bölme işlemi yapılmaktadır (Haykin 1991). Tekrarlamalı GS algoritmasında ise $3M^2 + 2M$ tane çarpma ve bölme işlemi yapılmaktadır. Tekrarlamalı GS algoritmasıyla parametrelerin güncellenmesi durumunda $9M + 8$ tane çarpma ve bölme işlemi daha az yapılmaktadır. Bu fark sadece parametre güncelleme işlemi için geçerlidir.

Yukarıda (3.344) ile verilen GMV tabanlı uyarlamalı kontrol kuralında $r(k)$ referans girişi yerine seçilen bir modelin çıkış işareti kullanılırsa, model-tabanlı uyarlamalı kontrol kuralı elde edilir (Goodwin ve Sin 1984, Landau ve ark. 1998). Bu durumda kapalı-çevrim sistemin çıkışı ile referans modelin çıkışı arasındaki hata sinyali $e(k) = y(k) - y_m(k)$ olacak ve kontrol kuralı bu hatanın varyansını minimum yapacaktır (Goodwin ve Sin 1984, Landau ve ark. 1998). Bu amaçla kapalı-çevrim sistem çıkışının seçilen referans modelin çıkışını takip etmesi için gerekli kontrol girişi

$$u(k) = -\hat{G}(z^{-1})^{-1}[\hat{F}(z^{-1})y(k) - C(z^{-1})y_m(k+d)] \tag{3.356}$$

eşitliği kullanılarak hesaplanabilir. Burada seçilen referans modelin çıkış işareti

$$A_m(z^{-1})y_m(k+d) = B_m(z^{-1})r(k) \tag{3.357}$$

model fark denklemi kullanılarak hesaplanabilir (Goodwin ve Sin 1984, Landau 1998).

3.7.2.3. Lyapunov kararlılık analizi

Lyapunov kararlılık kriteri kararlı bir kapalı çevrim sistem elde etmek amacıyla uyarlamalı denetleyici parametrelerinin ayarlanmasında kullanılmaktadır (Narendra ve Annaswamy 1989, Butler 1992, Aström ve Wittenmark 1995). Ayrıca denetleyici parametreleri sistem tanıma algoritmaları ile doğrudan ayarlanan uyarlamalı denetleyicili kapalı çevrim sistemin kararlılık analizinde de kullanılmaktadır (Johansson 1983,1986b,1989, Furuta 1993, Patete ve ark. 2006,2007,2008a,2008b, Hatun ve Koçal 2008). Bu amaçla önerilen tekrarlamalı GS algoritmasıyla birlikte kapalı-çevrim sistemin kararlılık analizini yapmak için aşağıdaki pozitif tanımlı Lyapunov fonksiyonu kullanılmıştır (Patete ve ark. 2007,2008a,2008b).

$$V(k) = \tfrac{1}{2}s^2(k) + \tfrac{1}{2}\widetilde{\theta}^T(k)R(k)\widetilde{\theta}(k) \tag{3.358}$$

Burada kapalı-çevrim sistemin kararlılığını ispatlamak için sistem sinyallerinin enerjilerinin sonlu olması şartıyla $\Delta V(k) = V(k) - V(k-1)$ ifadesinin negatif tanımlı olduğunu göstermek yeterlidir (Patete ve ark. 2007,2008a,2008b).

Burada öncelikle tekrarlamalı GS algoritmasının kararlılığı üzerinde durulacaktır. Bu amaçla öncelikle, (3.355) ile verilen skaler GS algoritması

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \left(R_L(k) + R_D(k)\right)^{-1}[p(k) - R(k)\hat{\theta}(k-1)] \tag{3.359}$$

şeklinde vektörel olarak yazılabilir. Bu denklem parametre hata vektörü $\tilde{\theta}(k) = \hat{\theta}(k) - \hat{\theta}_{opt}(k)$ olarak tanımlandığında ve $\hat{\theta}_{opt}(k)$ değeri denklemin her iki tarafından çıkarılarak gerekli düzenlemeler yapıldıktan sonra aşağıdaki gibi yazılabilir.

$$\tilde{\theta}(k) = [I - \left(R_L(k) + R_D(k)\right)^{-1} R(k)]\tilde{\theta}(k-1) + \left(R_L(k) + R_D(k)\right)^{-1}[p(k) - R(k)\hat{\theta}_{opt}(k)] \tag{3.360}$$

Bu denklemdeki ikinci terim normal denklem göz önüne alındığında sıfır vektör olur ve

$$\tilde{\theta}(k) = [I - \left(R_L(k) + R_D(k)\right)^{-1} R(k)]\tilde{\theta}(k-1) \tag{3.361}$$

olarak yazılabilir. Bu denklem sistem matrisi

$$T(k) = [I - \left(R_L(k) + R_D(k)\right)^{-1} R(k)] = -\left(R_L(k) + R_D(k)\right)^{-1} R_U(k) \tag{3.362}$$

olan bir homojen fark denklemidir ve bu denklemin çözümü parametre tahminlerinin başlangıç koşullarına bağlı olarak

$$\tilde{\theta}(k) = [T(k)]^k \tilde{\theta}(0) \tag{2.363}$$

şeklinde yazılabilir. Bu fark denkleminin sıfır vektöre yakınsayabilmesi için $T(k)$ sistem matrisinin bütün özdeğerlerinin 1'den küçük olması gerekir. Çünkü, doğrusal denklem takımlarındaki kare matrisin pozitif tanımlı olması durumunda yine pozitif tanımlı olan iterasyon matrisinin bütün özdeğerleri 1'den küçük olmakta ve GS iterasyonlarının optimum çözüme yakınsaması garantilenmektedir (Golub ve Loan 1996). Bilindiği gibi, sistem yeterli derecede uyarıldığı sürece $R(k)$ korelasyon matrisi pozitif tanımlı olmaktadır (Ljung ve Söderström 1983, Söderström ve Stoica 1989, Ljung 1999). Bu durumda bütün k değerleri için $R(k)$ matrisinin pozitif tanımlılığı sağlandığı sürece (3.362) ile verilen sistem matrisinin bütün özdeğerleri 1'den küçük olacak ve (3.363) iterasyonu sıfır vektöre yakınsayacaktır. Yani tekrarlamalı GS algoritmasının kararlılığı garantilenmiş olacaktır (Hatun ve Koçal 2005).

Yukarıda (2.344) ile verilen uyarlamalı kontrol kuralı (3.329) eşitliğinde kullanıldığında elde edilen

$$s(k+d) = \phi^T(k)\tilde{\theta}(k+d) \tag{3.364}$$

ifadesi Lyapunov fonksiyonda kullanılırsa

$$V(k) = \tfrac{1}{2}\tilde{\theta}^T(k)\phi(k-d)\phi(k-d)\tilde{\theta}(k) + \tfrac{1}{2}\tilde{\theta}^T(k)R(k)\tilde{\theta}(k) \tag{3.365}$$

elde edilir. Buradan $\Delta V(k)$ denklemi

$$\begin{aligned}\Delta V(k) = {} & \tfrac{1}{2}\tilde{\theta}^T(k)\phi(k-d)\phi(k-d)\tilde{\theta}(k) + \tfrac{1}{2}\tilde{\theta}^T(k)R(k)\tilde{\theta}(k) \\ & -\tfrac{1}{2}\tilde{\theta}^T(k-1)\phi(k-d-1)\phi(k-d-1)\tilde{\theta}(k-1) \\ & -\tfrac{1}{2}\tilde{\theta}^T(k-1)R(k-1)\tilde{\theta}(k-1)\end{aligned} \tag{3.366}$$

olur. Burada ilk iki terimde (3.362) eşitliği kullanıldığında

$$
\begin{aligned}
\Delta V(k) = & \tfrac{1}{2}\tilde{\theta}^T(k-1)T^T(k)\phi(k-d)\phi(k-d)T(k)\tilde{\theta}(k-1) \\
& + \tfrac{1}{2}\tilde{\theta}^T(k-1)T^T(k)R(k)T(k)\tilde{\theta}(k-1) \\
& - \tfrac{1}{2}\tilde{\theta}^T(k-1)\phi(k-d-1)\phi(k-d-1)\tilde{\theta}^T(k-1) \\
& - \tfrac{1}{2}\tilde{\theta}^T(k-1)R(k-1)\tilde{\theta}(k-1)
\end{aligned}
\tag{3.367}
$$

elde edilir. Burada 1. ve 3. terimlerde (3.353) eşitliği ile verilen korelasyon matrisi iterasyonu kullanıldığında ve gerekli düzenlemeler yapıldığında aşağıdaki sonuç elde edilir.

$$
\begin{aligned}
\Delta V(k) = & \tilde{\theta}^T(k-1)[T^T(k)R(k)T(k) - R(k-1)]\tilde{\theta}(k-1) \\
& - (\lambda/2)\tilde{\theta}^T(k-1)[T^T(k)R(k-1)T(k) - R(k-2)]\tilde{\theta}(k-1)
\end{aligned}
\tag{3.368}
$$

Bu durumda $T(k)$ iterasyon matrisinin bütün özdeğerleri 1'den küçük olduğu için köşeli parantezlerin içindeki ifadeler

$$T^T(k)R(k)T(k) - R(k-1) = -L_1^T(k)L_1(k) \tag{3.369}$$

$$T^T(k)R(k-1)T(k) - R(k-2) = -L_2^T(k)L_2(k) \tag{3.370}$$

şeklinde negatif tanımlıdır (Bitmead ve Anderson 1980, Anderson ve Johnson 1982). Bu ifadeler denklemde yerine yazıldığında aşağıdaki sonuç elde edilir.

$$\Delta V(k) = -\tilde{\theta}^T(k-1)L_1^T(k)L_1(k)\tilde{\theta}(k-1) + (\lambda/2)\tilde{\theta}^T(k-1)L_2^T(k)L_2(k)\tilde{\theta}(k-1) \tag{3.371}$$

Bu sonuçtaki pozitif işaretli ikinci terimin başındaki katsayı $0 < \lambda \le 1$ için 1/2 değerinden küçük olacağı için $\Delta V(k)$ değerinin her zaman negatif olacağı söylenebilir. Böylece kapalı-çevrim sistem asimptotik kararlı olur (Hatun ve Koçal 2008).

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Bu kısımda önerilen tekrarlamalı GS algoritmasının değişik uygulamaları ve karşılaştırmalı benzetim sonuçları verilmiştir. Yapılan bütün benzetim çalışmalarında Matlab programı kullanılmıştır.

4.1. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması İle Ayrık-Zaman Sistemlerin Transfer Fonksiyonu Parametrelerinin Yansız Tahmini

Bu kısımda, önerilen tekrarlamalı GSYD algoritmasının yakınsama özelliği, yapılan bir benzetim çalışmasıyla tekrarlamalı yardımcı değişkenler algoritmasıyla karşılaştırmalı olarak incelenmiştir. Benzetim çalışmasında kullanılan örnek ikinci derece sistemin transfer fonksiyonu

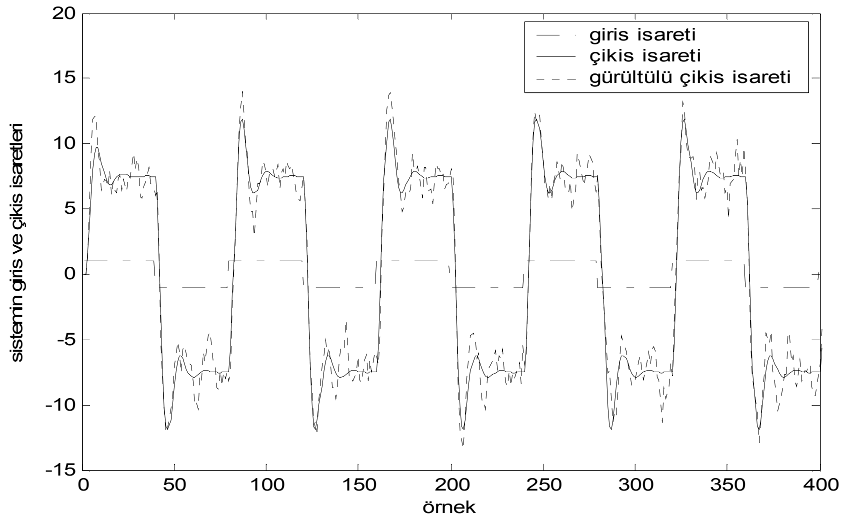
$$\frac{B(z)}{A(z)} = \frac{z+0.5}{z^2-1.5z+0.7} \tag{4.1}$$

olarak seçilmiştir (Ljung 2008). Burada kullanılan veri vektörü ve doğru parametre vektörü

$$\varphi(k) = \begin{bmatrix} -y(k-1) & -y(k-2) & u(k) & u(k-1) & u(k-2) \end{bmatrix}^T \tag{4.2}$$

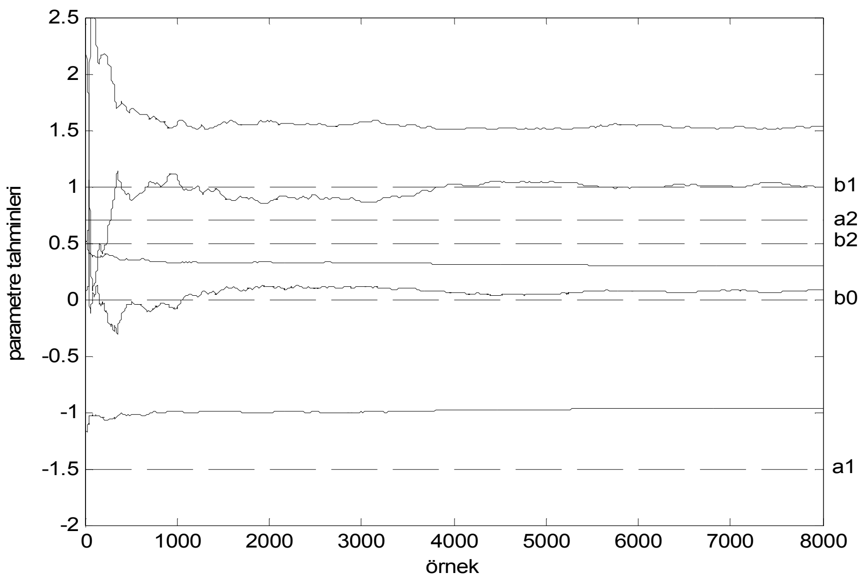
$$\theta_{opt} = \begin{bmatrix} a_1 & a_2 & b_0 & b_1 & b_2 \end{bmatrix}^T = \begin{bmatrix} -1.5 & 0.7 & 0.0 & 1.0 & 0.5 \end{bmatrix}^T \tag{4.3}$$

olup sistemin çıkışına $v(k) = e(k) - e(k-1) + 0.2e(k-2)$ şeklinde elde edilen renkli gürültü eklenmiştir, burada $e(k)$ gürültüsü sıfır ortalamalı varyansı 1 olan normal dağılıma sahip rastgele gürültü dizisidir. Sisteme giriş işareti olarak genliği ∓ 1 ve periyodu 80 örnekleme periyoduna eşit olan bir kare dalga uygulanmıştır. Bu durumda sistemin gürültüsüz çıkış işareti ve $y(k) = \varphi^T(k)\theta_{opt} + v(k)$ eşitliği kullanılarak elde edilen gürültülü çıkış işareti Şekil 4.1'de görülmektedir.



Şekil 4.1. Simülasyonda kullanılan örnek sistemin giriş ve çıkış işaretleri.

Bu durumda tekrarlamalı GS algoritmasıyla yapılan benzetim sonuçları Şekil 4.2'de görülmektedir.

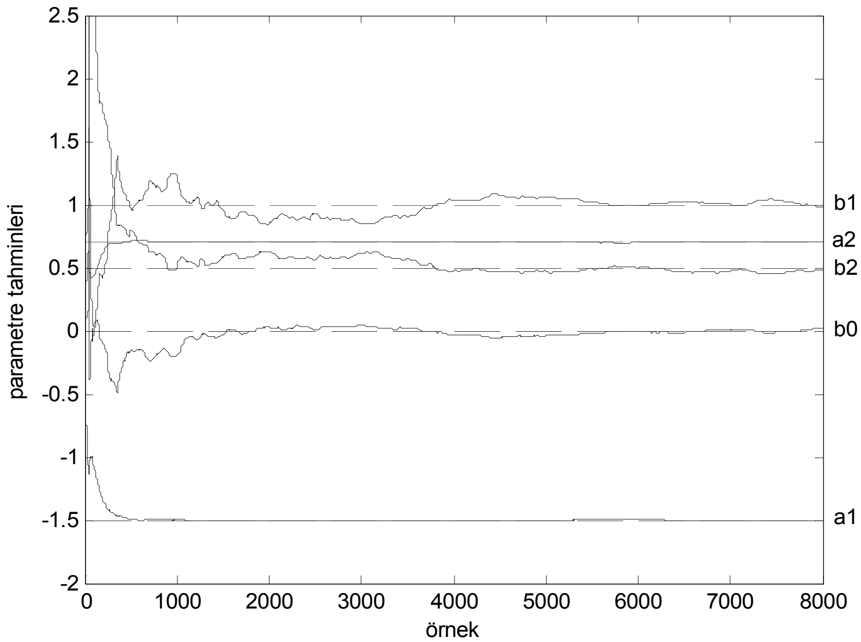


Şekil 4.2. Tekrarlamalı GS algoritmasıyla hesaplanan transfer fonksiyonu parametre tahminleri.

Burada kesikli düz çizgiler doğru parametre değerlerini göstermektedir. Tekrarlamalı GS algoritması kullanıldığında 8000 adım sonunda elde edilen parametre tahminlerinin

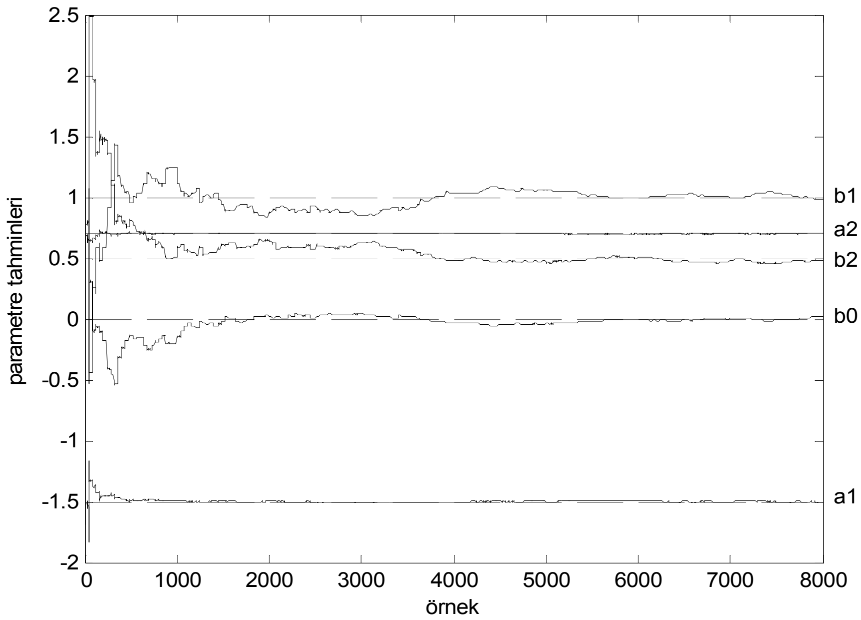
$$\hat{\theta}^T(8000) = [\hat{a}_1 \quad \hat{a}_2 \quad \hat{b}_0 \quad \hat{b}_1 \quad \hat{b}_2] = \left[-0.9656 \quad 0.2978 \quad 0.0849 \quad 0.9950 \quad 1.5275\right] \qquad (4.4)$$

değerlerine yakınsadığı görülmüştür. Bu değerler (4.3) ile verilen doğru değerlere uzaktır, yani yanlı parametre tahminleri elde edilmiştir. Daha sonra sırasıyla tekrarlamalı GSYD ve tekrarlamalı YD algoritmasıyla yapılan benzetim sonucunda elde edilen parametre tahminleri Şekil 4.3'te ve Şekil 4.4'te görülmektedir. Korelasyon matrisinin başlangıç değeri tekrarlamalı GSYD algoritmasında $R(0) = 0.1I$, tersinin başlangıç değeri ise tekrarlamalı YD algoritmasında $P(0) = 10I$ olarak alınmıştır. Elde edilen sonuçlara göre tekrarlamalı GSYD algoritması kullanıldığında 8000 adım sonunda elde edilen parametre tahminlerinin $\hat{\theta}^T_{GSYD} = [\hat{a}_1 \quad \hat{a}_2 \quad \hat{b}_0 \quad \hat{b}_1 \quad \hat{b}_2]$ $= \left[-1.5015 \quad 0.6985 \quad 0.0204 \quad 0.9820 \quad 0.4761\right]$ şeklinde (4.3) ile verilen doğru değerlerine yakınsadığı görülmüştür, yani yansız parametre tahminleri elde edilmiştir.



Şekil 4.3. Tekrarlamalı GSYD algoritmasıyla hesaplanan transfer fonksiyonu parametre tahminleri.

Tekrarlamalı YD algoritması kullanıldığında ise yine 8000 adım sonunda elde edilen parametre tahminlerinin $\hat{\theta}_{YD}^{T} = [\hat{a}_1 \quad \hat{a}_2 \quad \hat{b}_0 \quad \hat{b}_1 \quad \hat{b}_2]$ $= \left[-1.5005 \quad 0.6980 \quad 0.0171 \quad 0.9854 \quad 0.4790\right]$ şeklinde doğru değerlere yakınsadığı görülmüştür.



Şekil 4.4. Tekrarlamalı YD algoritmasıyla hesaplanan transfer fonksiyonu parametre tahminleri.

Elde edilen benzetim sonuçlarına bakıldığında önerilen tekrarlamalı GSYD algoritmasının en küçük kareler tabanlı algoritmalara göre işlem yükünün az olmasının yanında tekrarlamalı YD algoritmasına yakın sonuçlar verdiği görülmektedir. Algoritmanın yakınsama hızının korelasyon matrisinin özdeğer yayılımına bağımlılığından dolayı, başlangıçta tekrarlamalı YD algoritmasına göre daha yavaş yakınsamasına rağmen tekrarlamalı YD algoritmasına oldukça yakın sonuçlar vermiştir. Sonuç olarak işlem yükü ve yakınsama hızı açısından bakıldığında tekrarlamalı YD algoritmasına göre iyi bir alternatif olarak görülebilir.

Tek-girişli tek-çıkışlı ayrık zaman sistemin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.1`de verilmiştir.

Çizelge 4.1. Tek-girişli tek-çıkışlı ayrık-zaman sistemin parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	85	300 (yanlı tahmin)
GSYD	90	300 (yansız tahmin)
YD	143	100 (yansız tahmin)

4.2. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Çok-Girişli Çok-Çıkışlı Ayrık-Zaman Sistemlerin Transfer Fonksiyonu Matrisi Parametrelerinin Yansız Tahmini

Bu kısımda, yapılan bir benzetim çalışmasıyla, çok-değişkenli sistemler için önerilen tekrarlamalı GSYD algoritmasının yakınsama özelliği diğer en küçük kareler tabanlı algoritmalarla karşılaştırmalı olarak incelenmiştir. Benzetim çalışmasında kullanılan örnek iki-girişli iki-çıkışlı sistemin transfer fonksiyonu matrisi

$$\begin{bmatrix} X_1(z) \\ X_2(z) \end{bmatrix} = \begin{bmatrix} \dfrac{z+0.8}{z^2-1.5z+0.7} & \dfrac{2.6z-1.3}{z^2-1.5z+0.7} \\ \dfrac{0.12z+0.324}{z^2-0.24z+0.16} & \dfrac{0.25z+0.186}{z^2-0.24z+0.16} \end{bmatrix} \begin{bmatrix} U_1(z) \\ U_2(z) \end{bmatrix} \tag{4.5}$$

olarak seçilmiştir. Bu sistemin her bir çıkışı için fark denklem modeli ölçme gürültüsü de göz önüne alınarak aşağıdaki gibi yazılabilir.

$$y_1(k) = 1.5y_1(k-1) - 0.7y_1(k-2) + 1.0u_1(k-1) + 0.8u_1(k-2) + 2.6u_2(k-1) - 1.3u_2(k-2) + v_1(k) \tag{4.6}$$

$$\begin{aligned} y_2(k) = 0.24y_2(k-1) - 0.16y_2(k-2) + 0.12\,u_1(k-1) + 0.324\,u_1(k-2) + 0.25u_2(k-1) \\ - 0.186\,u_2(k-2) + v_2(k) \end{aligned} \tag{4.7}$$

Bu denklemler doğrusal bağlanımlı biçimde

$$y_1(k) = \varphi_1^T(k)\theta_{1opt} + v_1(k) \tag{4.8}$$

$$y_2(k) = \varphi_2^T(k)\theta_{2opt} + v_2(k) \tag{4.9}$$

olarak yazılabilir. Sistem tanıma işleminde kullanılan veri vektörleri ve optimum parametre vektörleri

$$\varphi_1^T(k) = \left[-y_1(k-1) \quad -y_1(k-2) \quad u_1(k-1) \quad u_1(k-2) \quad u_2(k-1) \quad u_2(k-2)\right] \tag{4.10}$$

$$\varphi_2^T(k) = \left[-y_2(k-1) \quad -y_2(k-2) \quad u_1(k-1) \quad u_1(k-2) \quad u_2(k-1) \quad u_2(k-2)\right] \tag{4.11}$$

$$\theta_{1opt}^T = \left[a_{11} \quad a_{12} \quad b_{111} \quad b_{112} \quad b_{121} \quad b_{122}\right] = \left[-1.5 \quad 0.7 \quad 1.0 \quad 0.8 \quad 2.6 \quad -1.3\right] \tag{4.12}$$

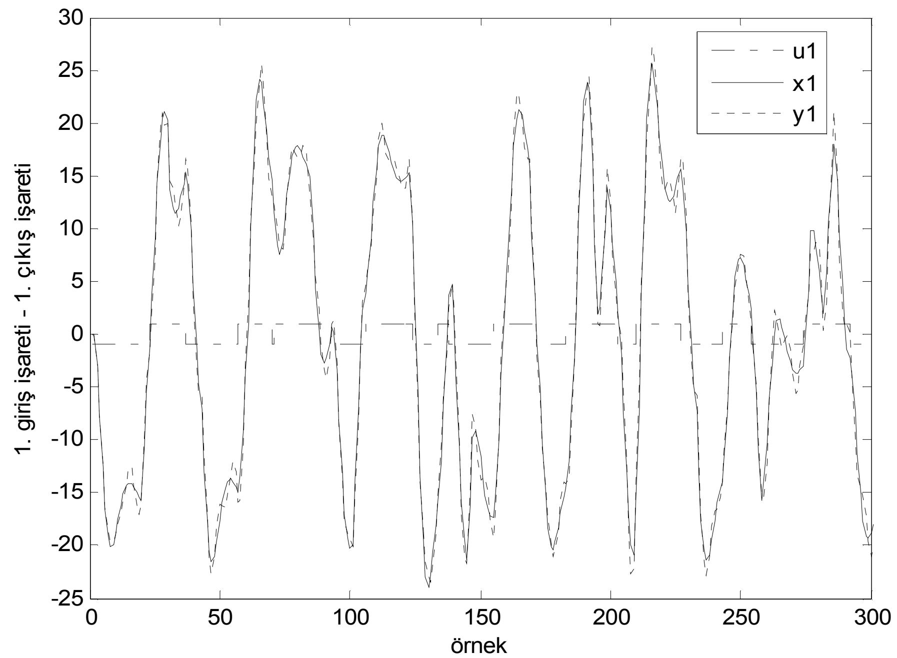
$$\theta_{2opt}^T = \left[a_{21} \quad a_{22} \quad b_{211} \quad b_{212} \quad b_{221} \quad b_{222}\right] = \left[-0.24 \quad 0.16 \quad 0.12 \quad 0.324 \quad 0.25 \quad -0.186\right] \tag{4.13}$$

olup sistemin çıkış işareti ölçümlerine sırasıyla

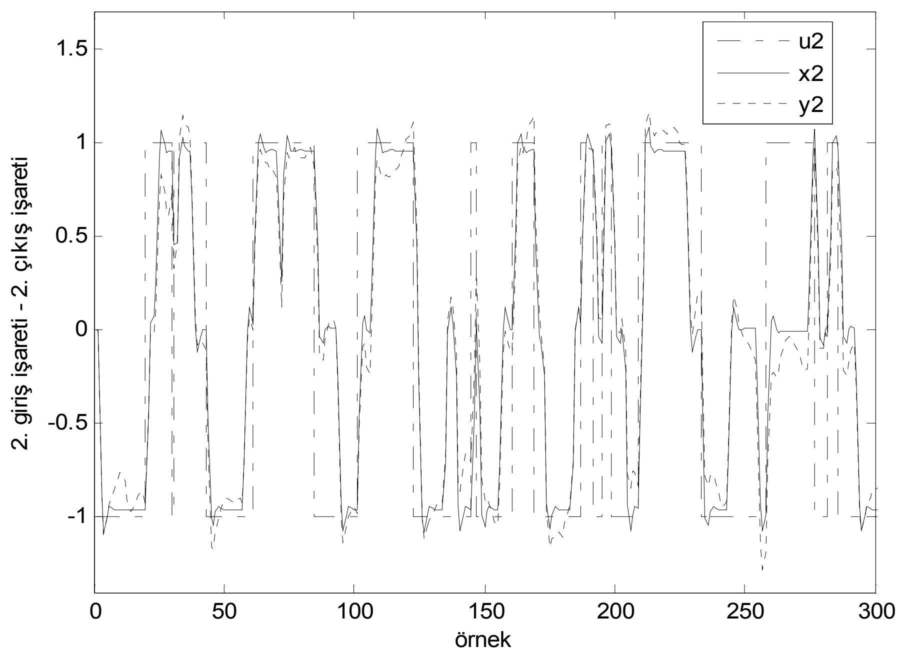
$$v_1(k) = 0.9v_1(k-1) - 0.95v_1(k-2) + e_1(k) \tag{4.14}$$

$$v_2(k) = 1.6v_2(k-1) - 0.68v_2(k-2) + e_2(k) \tag{4.15}$$

şeklinde elde edilen renkli gürültü dizileri eklenmiştir, burada $e_1(k)$ ve $e_2(k)$ sıfır ortalamalı ve varyansları 1 olan normal dağılıma sahip rastgele gürültü dizileridir. Sistemin $u_1(k)$ ve $u_2(k)$ girişlerine genliği ∓ 1 şeklinde değişen iki farklı yalancı rastgele ikili dizi (pseudo random binary sequence) uygulanmıştır. Bu durumda (4.8) ve (4.9) denklemleri kullanılarak elde edilen gürültülü çıkış işaretleri Şekil 4.5'te ve Şekil 4.6'da görülmektedir. Burada $v_1(k)$ ve $v_2(k)$ ölçme gürültülerinin varyansları birinci çıkış için gürültü işaret oranı %10 ve ikinci çıkış için gürültü işaret oranı %20 olacak şekilde ayarlanmıştır.

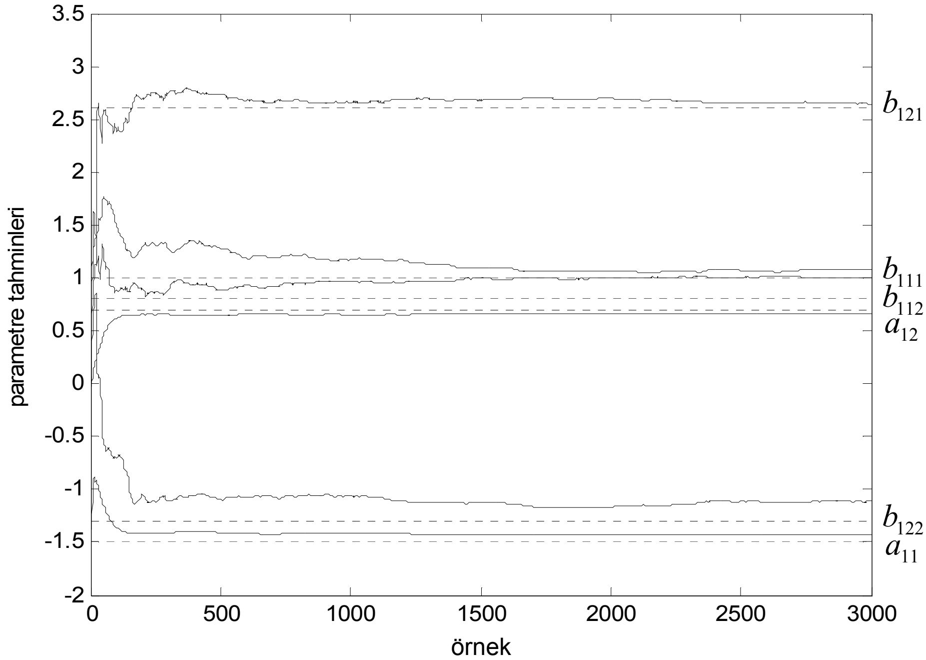


Şekil 4.5. Çok değişkenli ayrık-zaman sistemin 1. giriş ve 1. çıkış işaretleri.

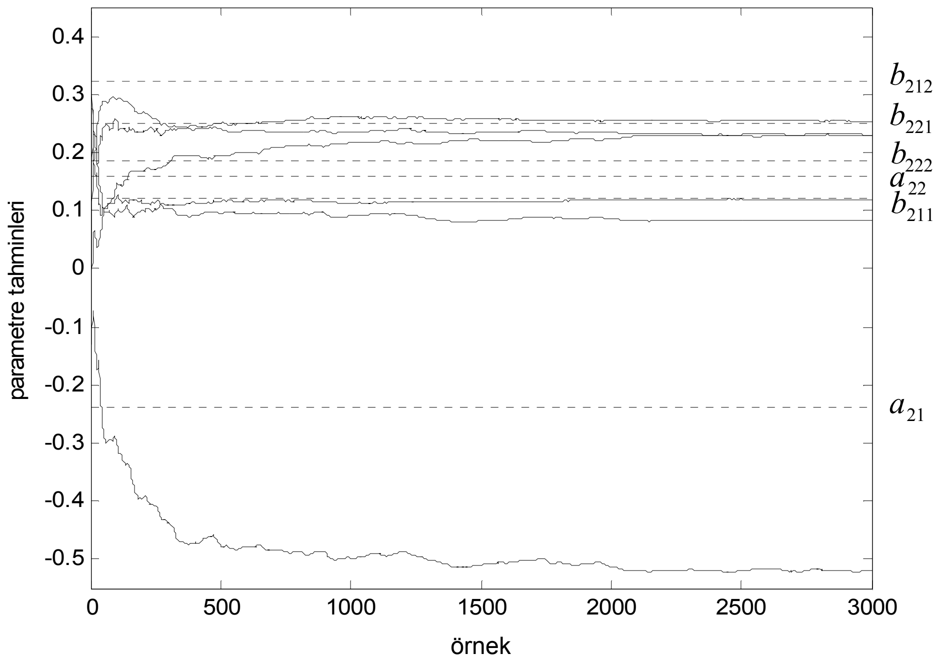


Şekil 4.6. Çok değişkenli ayrık-zaman sistemin 2. giriş ve 2. çıkış işaretleri.

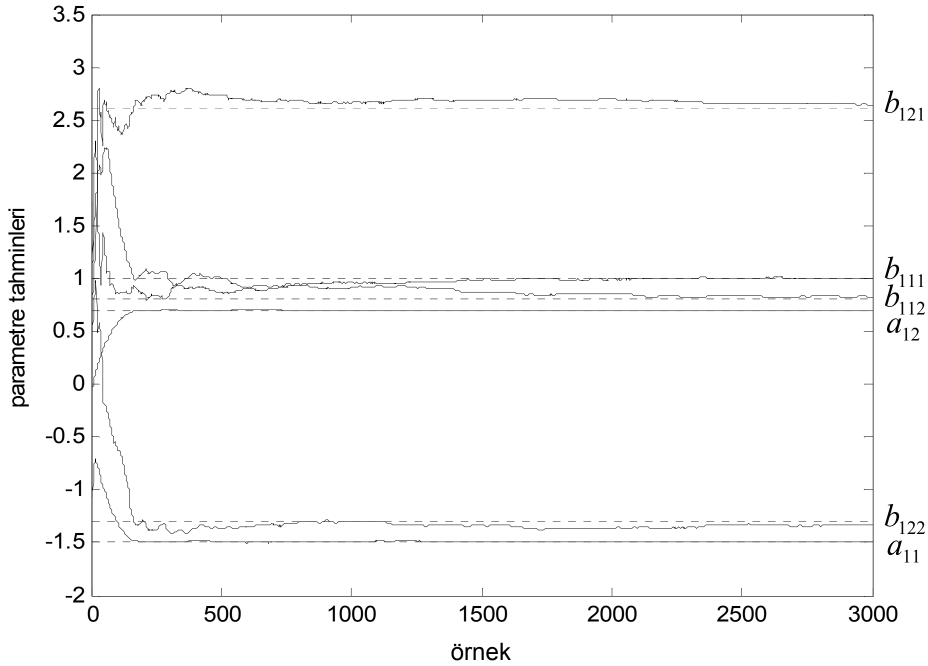
Bu durumda tekrarlamalı GS algoritmasıyla yapılan benzetim sonuçları Şekil 4.7'de ve Şekil 4.8'de görülmektedir. Burada noktalı düz çizgiler doğru parametre değerlerini göstermektedir. Bu sonuçlarda parametre tahminlerinin doğru değerlerine yakınsamadığı görülmektedir. Daha sonra yansız parametre tahminlerini elde edebilmek için, önerilen tekrarlamalı GSYD algoritmasıyla yapılan benzetim sonucunda elde edilen parametre tahminleri Şekil 4.9'da ve Şekil 4.10'da görülmektedir. Literatürde yaygın olarak kullanılan tekrarlamalı GSYD algoritmasıyla yapılan simülasyon sonucunda elde edilen parametre tahminleri de Şekil 4.11'de ve Şekil 4.12'de görülmektedir. Kullanılan bütün algoritmalarda korelasyon matrisinin başlangıç değerleri birim matris olarak, parametre tahminlerinin başlangıç değerleri ise sıfır vektör olarak alınmıştır ve parametreler sabit olduğu için unutma faktörü $\lambda = 1$ alınmıştır.



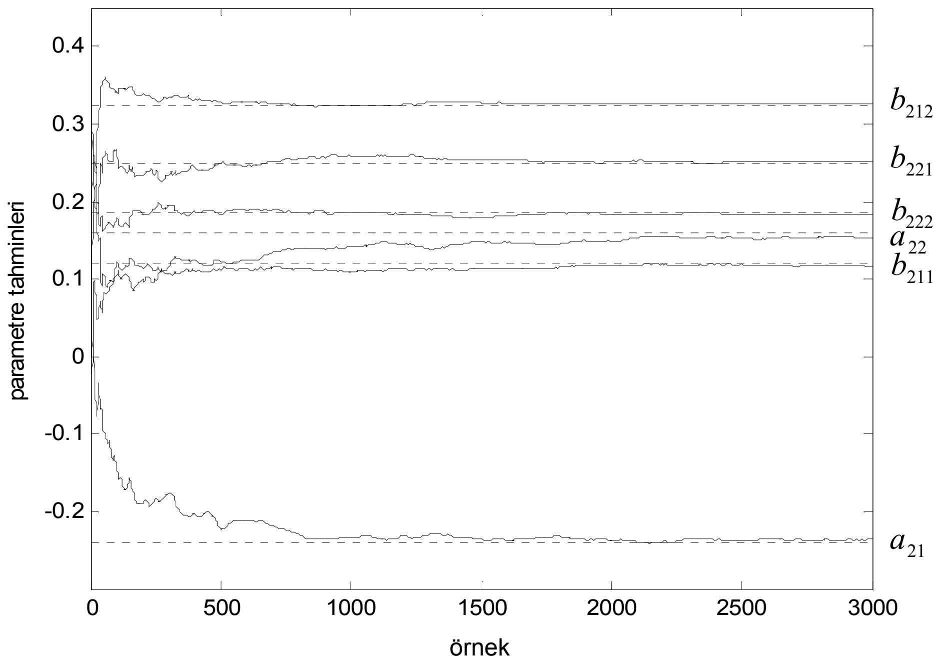
Şekil 4.7. Tekrarlamalı GS algoritmasıyla hesaplanan, çok-değişkenli ayrık-zaman sistemin $\hat{\theta}_1$ parametre tahminleri.



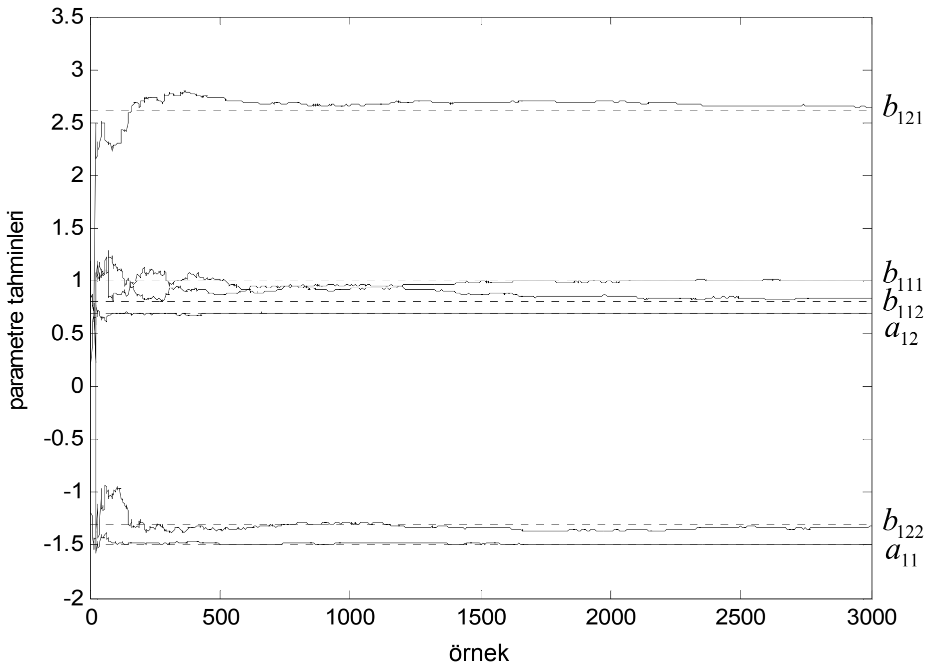
Şekil 4.8. Tekrarlamalı GS algoritmasıyla hesaplanan, çok-değişkenli ayrık-zaman sistemin $\hat{\theta}_2$ parametre tahminleri.



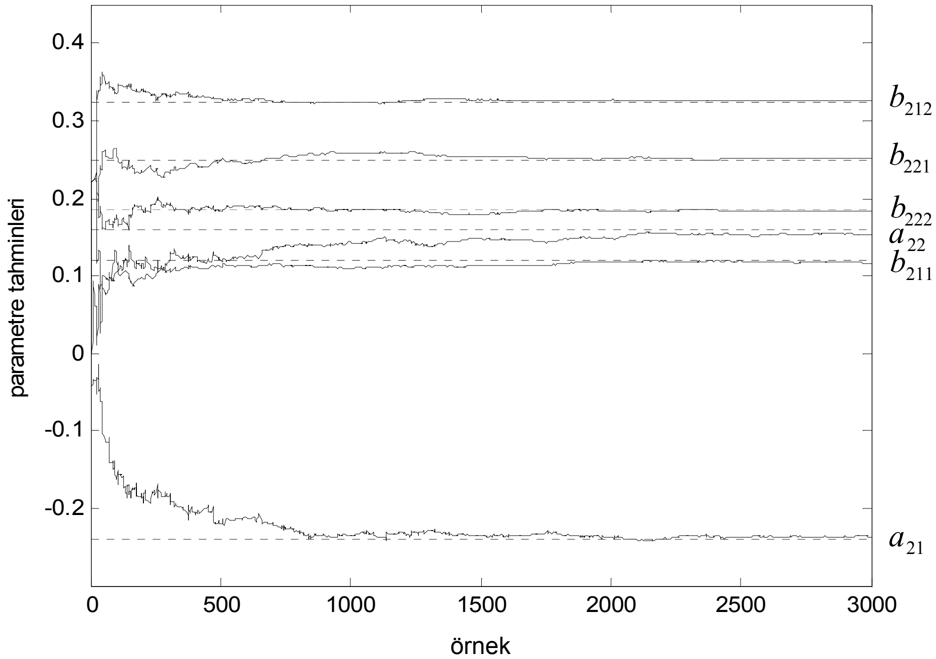
Şekil 4.9. Tekrarlamalı GSYD algoritmasıyla hesaplanan, çok-değişkenli ayrık-zaman sistemin $\hat{\theta}_1$ parametre tahminleri.



Şekil 4.10. Tekrarlamalı GSYD algoritmasıyla hesaplanan, çok-değişkenli ayrık-zaman sistemin $\hat{\theta}_2$ parametre tahminleri.



Şekil 4.11. Tekrarlamalı YD algoritmasıyla hesaplanan, çok-değişkenli ayrık-zaman sistemin $\hat{\theta}_1$ parametre tahminleri.



Şekil 4.12. Tekrarlamalı YD algoritmasıyla hesaplanan, çok-değişkenli ayrık-zaman sistemin $\hat{\theta}_2$ parametre tahminleri.

Yapılan benzetim sonuçlarında 3000 adım sonunda parametrelerin yakınsadığı sayısal değerler Çizelge 4.2`de verilmiştir.

Çizelge 4.2. Çok-girişli çok-çıkışlı ayrık-zaman sistemin parametre tahminlerinin yakınsadığı değerler.

Algoritma:	$\hat{a}_{11}$	$\hat{a}_{12}$	$\hat{b}_{111}$	$\hat{b}_{112}$	$\hat{b}_{121}$	$\hat{b}_{122}$
GS	-1.4271	0.6559	1.0043	1.0724	2.6411	-1.1072
GSYD	-1.4934	0.6953	1.0033	0.8285	2.6409	-1.3269
YD	-1.4921	0.6942	1.0037	0.8314	2.6407	-1.3238
Doğru değer	-1.5000	0.7000	1.0000	0.8000	2.6000	-1.3000
Algoritma:	$\hat{a}_{21}$	$\hat{a}_{22}$	$\hat{b}_{211}$	$\hat{b}_{212}$	$\hat{b}_{221}$	$\hat{b}_{222}$
GS	-0.5206	0.2305	0.1182	0.2302	0.2540	0.0837
GSYD	-0.2361	0.1530	0.1161	0.3265	0.2516	0.1837
YD	-0.2365	0.1531	0.1161	0.3264	0.2516	0.1835
Doğru değer	-0.2400	0.1600	0.1200	0.3240	0.2500	0.1860

Elde edilen sonuçlara bakıldığında tekrarlamalı GS algoritması kullanıldığında 3000 adım sonunda yanlı parametre tahminlerinin elde edildiği görülmüştür. Tekrarlamalı GSYD ve tekrarlamalı YD algoritmaları kullanıldığında ise 3000 adım sonunda elde edilen parametre tahminlerinin doğru değerlere yakınsadığı görülmüştür. Bu iki algoritma arasındaki en önemli fark $\hat{\theta}_1$ tahminlerinin hesaplandığı Şekil 4.9 ve Şekil 4.11'de görülmektedir. Yapılan benzetim çalışmalarında $\hat{\theta}_1$ parametrelerini güncellemek için kullanılan korelasyon matrisinin özdeğer yayılımı yaklaşık 2390 olarak hesaplanmıştır. Tekrarlamalı GSYD algoritması korelasyon matrisinin özdeğer yayılımından etkilendiği için parametre tahminleri tekrarlamalı YD algoritmasına göre daha yavaş yakınsamıştır. Şekil 4.10 ve Şekil 4.12'de parametre tahminlerinin yakınsama hızlarının çok yakın olduğu görülmektedir. Çünkü, yapılan benzetim çalışmalarında $\hat{\theta}_2$ parametrelerini güncellemek için kullanılan korelasyon matrisinin özdeğer yayılımı yaklaşık 158 olarak hesaplanmıştır.

Çok-girişli çok-çıkışlı ayrık zaman sistemin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.3`te verilmiştir.

Çizelge 4.3. Çok-girişli çok-çıkışlı ayrık-zaman sistemin parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü ($\hat{\theta}_1$ için) (çarpma-bölme işlemi sayısı)	Yakınsama Hızı ($\hat{\theta}_1$ için) (yaklaşık örnek sayısı)
GS	56	200 (yanlı tahmin)
GSYD	60	200 (yansız tahmin)
YD	104	100 (yansız tahmin)
Algoritma:	İşlem Yükü ($\hat{\theta}_2$ için) (çarpma-bölme işlemi sayısı)	Yakınsama Hızı ($\hat{\theta}_2$ için) (yaklaşık örnek sayısı)
GS	56	200 (yanlı tahmin)
GSYD	60	200 (yansız tahmin)
YD	104	100 (yansız tahmin)

4.3. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Sürekli-Zaman Sistemlerin Transfer Fonksiyonu Parametrelerinin Yansız Tahmini

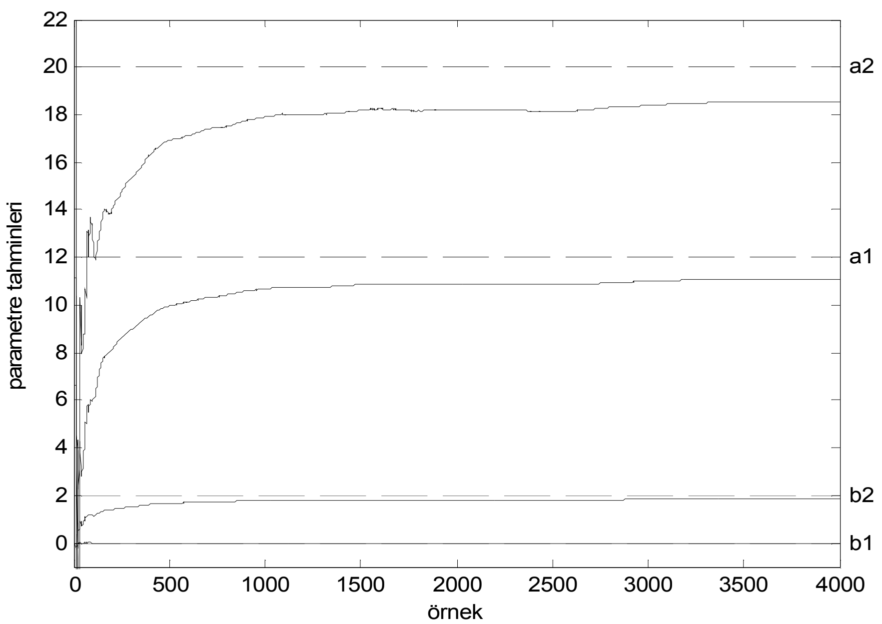
Bu kısımda, bir doğru akım motoruna sert bir mil ile bağlı bir yükten oluşan bir sistemin transfer fonksiyonu parametreleri tahmin edilmiştir. Benzetim çalışmasında kullanılan motorun armatür direnci $R_a = 1\Omega$, armatür endüktansı $L_a = 0.5$ H, motorun ve yükün eylemsizliği $J = 0.01$ kgm^2/s^2, motorun ve yükün sürtünme katsayısı $B = 0.1$ Nms, motorun zıt elektromotor sabiti $K_b = 0.01$ V/(rad/s) ve motor tork sabiti $K_t = 0.01$ Nm/A olarak alınmıştır. Bu durumda motor ve yükten oluşan sistemde armatür geriliminden yükün açısal hızına kadar olan sürekli zaman transfer fonksiyonu

$$\frac{X(s)}{U(s)} = \frac{b_1 s + b_2}{s^2 + a_1 s + a_2} = \frac{2}{s^2 + 12s + 20.02} \tag{4.16}$$

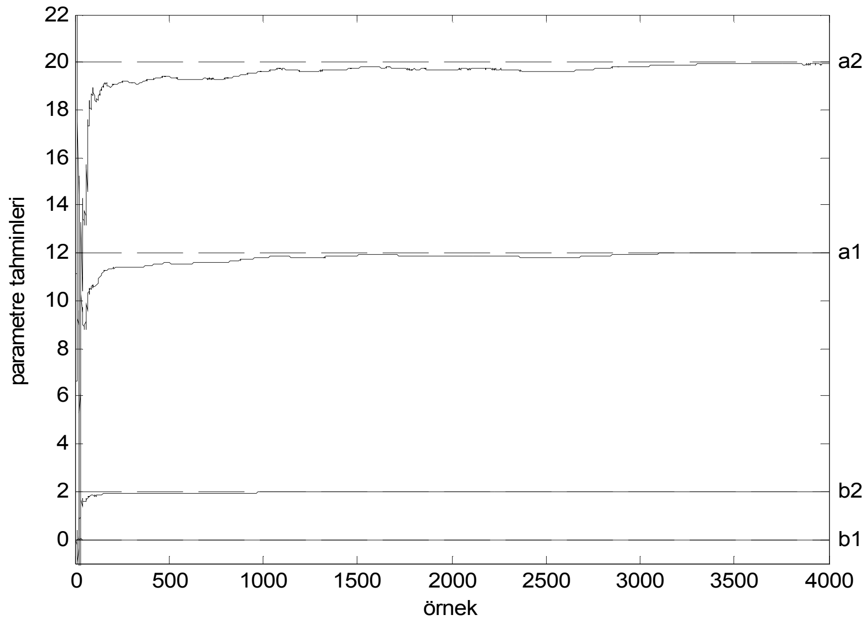
olarak yazılabilir. Burada optimum parametre vektörü

$$\theta_s = \begin{bmatrix} a_1 & a_2 & b_1 & b_2 \end{bmatrix}^T = \begin{bmatrix} 12 & 20.02 & 0 & 2 \end{bmatrix}^T \tag{4.17}$$

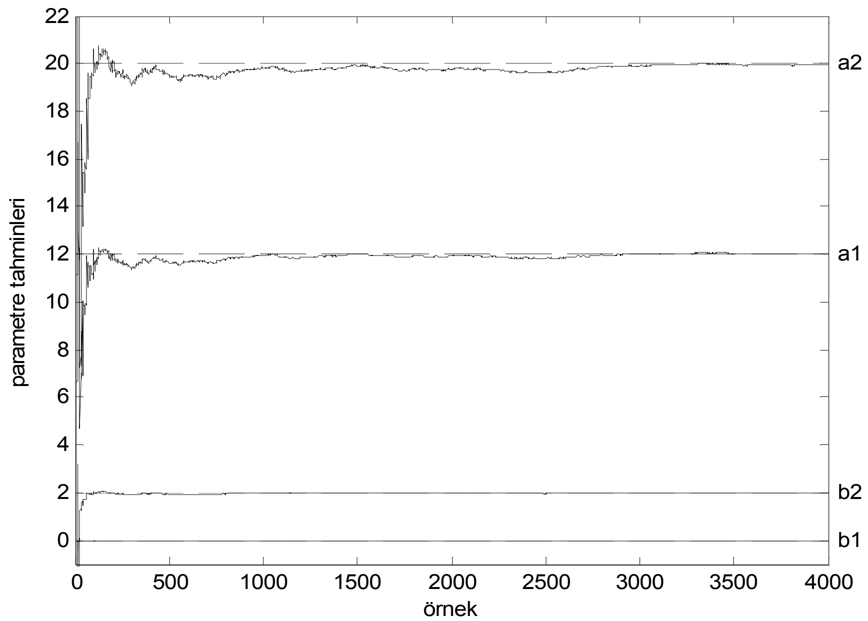
olarak yazılabilir. Sisteme giriş işareti (armatür gerilimi) olarak, uyarım özelliklerinin iyi olmasından dolayı ±10V seviyeli yalancı rastgele ikili dizi (pseudo random binary sequence) uygulanmıştır. Çıkış işaretine $v(k) = e(k) - e(k-1) + 0.2e(k-2)$ şeklinde elde edilen renkli gürültü eklenmiştir. Burada $e(k)$ normal dağılıma sahip sıfır ortalamalı rastgele gürültü dizisidir. Çıkış işaretine eklenen $v(k)$ ölçme gürültüsünün varyansı gürültü işaret oranı %30 olacak şekilde ayarlanmıştır. Örnekleme periyodu, Shannon maksimum örnekleme periyodu $T_S = \pi / \omega_n$ ve ω_n sistemin doğal frekans olmak üzere, $T_S / 150 < T < T_S / 10$ kriteri (Sagara ve Zhao 1990) göz önüne alınarak $T = 0.05$ s. olarak seçilmiştir. Sistemin bant genişliği yaklaşık 1.929 rad/s olup $\tau = 0.3$ ($a = 1/\tau \cong 3.33$) alınarak kullanılan ikinci derece alçak geçiren filtrenin bant genişliğinin 2.145 rad/s olması sağlanmıştır. Yani kullanılan ikinci derece alçak geçiren filtrenin bant genişliği sistemin bant genişliğinden bir miktar fazla olacak şekilde seçilmiştir. Parametre tahminlerinin başlangıç değerleri $\hat{\theta}(0) = 0_{(4\times1)}$ şeklinde sıfır vektör alınmıştır, $R(0) = 0.001 I_{(4\times4)}$ ve $r(0) = 0_{(4\times1)}$ olarak alınmıştır. İlk olarak tekrarlamalı GS algoritması kullanılmıştır ve gürültülü durumda Şekil 4.13'te görüldüğü gibi hesaplanan parametre tahminleri kesikli çizgilerle gösterilen doğru değerine yakınsamamıştır, yani yanlı parametre tahminleri elde edilmiştir. Sonra tekrarlamalı GSYD algoritması kullanılmış ve Şekil 4.14'te görüldüğü gibi yansız parametre tahminleri elde edilmiştir. Daha sonra ise tekrarlamalı YD algoritması (Ljung ve Söderström 1983, Söderström ve Stoica 1983,1989, Ljung 1999) kullanılmış ve Şekil 4.15'te görüldüğü gibi yine yansız parametre tahminleri elde edilmiştir. Başlangıç değerlerinin aynı olması için parametre tahminlerinin başlangıç değerleri sıfır alınmıştır ve $R^{-1}(0) = 1000 I_{(4\times4)}$ olarak alınmıştır.



Şekil 4.13. Tekrarlamalı GS algoritmasıyla hesaplanan, sürekli-zaman sistemin transfer fonksiyonu parametre tahminleri.

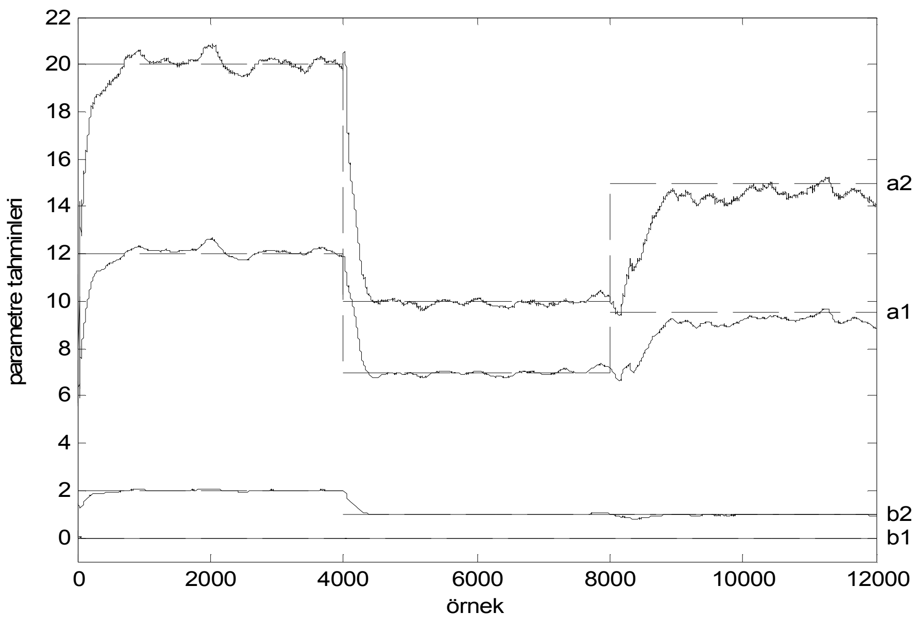


Şekil 4.14. Tekrarlamalı GSYD algoritmasıyla hesaplanan, sürekli-zaman sistemin transfer fonksiyonu parametre tahminleri.



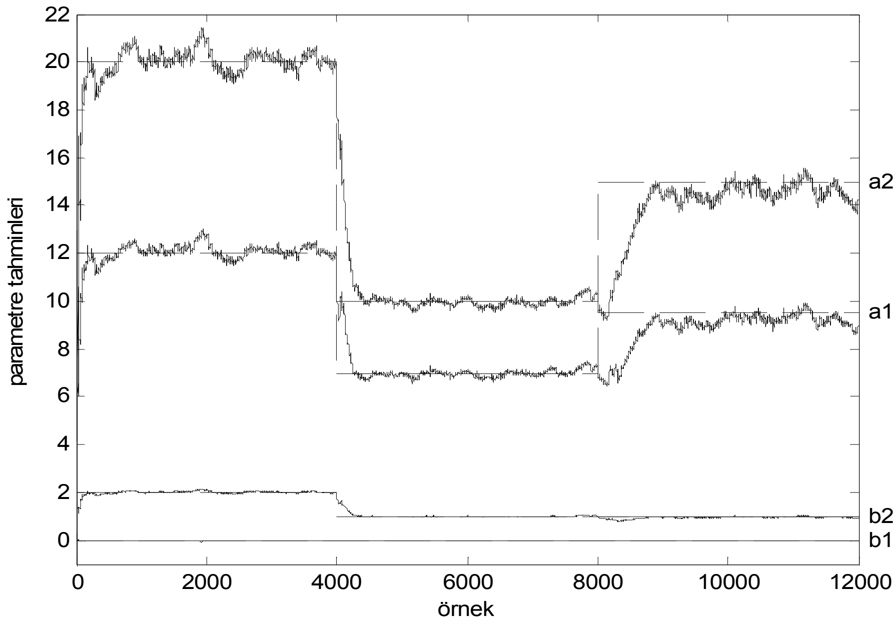
Şekil 4.15. Tekrarlamalı YD algoritmasıyla hesaplanan, sürekli-zaman sistemin transfer fonksiyonu parametre tahminleri.

Sonra sisteme $k = 4000$ anında eylemsizliği $J = 0.01$ kgm^2/s^2 olan bir yük ilave edilmiştir ve $k = 8000$ anında sürtünme katsayısı $B = 0.15$ Nms olarak değiştirilmiştir. Bu durumda $k = 4000 \sim 8000$ adım aralığında $\theta_s = \begin{bmatrix}7 & 10.01 & 0 & 1\end{bmatrix}^T$ olarak, $k = 8000 \sim 12000$ adım aralığında $\theta_s = \begin{bmatrix}9.5 & 15.01 & 0 & 1\end{bmatrix}^T$ olarak değişmiştir. Unutma faktörü $\lambda = 0.995$ alınarak tekrarlamalı GSYD algoritmasıyla Şekil 4.16'da görülen parametre tahminleri elde edilmiştir.



Şekil 4.16. Tekrarlamalı GSYD algoritmasıyla hesaplanan, zamanla değişen sürekli-zaman sistemin transfer fonksiyonu parametre tahminleri.

Daha sonra aynı şartlarda tekrarlamalı YD algoritması kullanılmış ve $\lambda = 0.995$ alınarak Şekil 4.17'de görülen parametre tahminleri elde edilmiştir.



Şekil 4.17. Tekrarlamalı YD algoritmasıyla hesaplanan, zamanla değişen sürekli-zaman sistemin transfer fonksiyonu parametre tahminleri.

Tek-girişli tek-çıkışlı sürekli zaman sistemin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.4`te verilmiştir.

Çizelge 4.4. Tek-girişli tek-çıkışlı sürekli-zaman sistemin parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	56	500 (yanlı tahmin)
GSYD	60	500 (yansız tahmin)
YD	104	100 (yansız tahmin)

Yapılan benzetim sonuçlarına göre, ölçme gürültüsünün renkli gürültü olması durumunda, önerilen tekrarlamalı GS algoritmasıyla yanlı parametre tahminlerinin elde edildiği görülmüştür. Bu problemden kurtulmak için önerilen tekrarlamalı GSYD algoritmasıyla ise yansız parametre tahminlerinin elde edilebildiği ve yakınsama hızının yaygın olarak kullanılan tekrarlamalı YD algoritmasına yakın olduğu görülmüştür. Önerilen algoritmanın yakınsama hızının tekrarlamalı YD algoritmasına göre yavaş olması korelasyon matrisinin özdeğer yayılımına bağımlı olmasından kaynaklanmaktadır. Buna rağmen, ayrık veriler ile sürekli-zamanlı uyarlamalı kontrol ve sürekli model tabanlı hata algılama sistemlerinde RLS ve YD algoritmalarına alternatif olarak kullanılabilir.

4.4. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Çok-Girişli Çok-Çıkışlı Sürekli-Zaman Sistemlerin Transfer Fonksiyonu Matrisi Parametrelerinin Yansız Tahmini

Benzetim çalışmasında kullanılan örnek iki-girişli iki-çıkışlı sürekli-zaman sistemin transfer fonksiyonu matrisi

$$\begin{bmatrix} X_1(s) \\ X_2(s) \end{bmatrix} = \begin{bmatrix} \dfrac{b_{111}s+b_{112}}{s^2+a_{11}s+a_{12}} & \dfrac{b_{121}s+b_{122}}{s^2+a_{11}s+a_{12}} \\ \dfrac{b_{211}s+b_{212}}{s^2+a_{21}s+a_{22}} & \dfrac{b_{221}s+b_{222}}{s^2+a_{21}s+a_{22}} \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} = \begin{bmatrix} \dfrac{5s+5}{s^2+2.8s+4} & \dfrac{2}{s^2+2.8s+4} \\ \dfrac{3}{s^2+3s+6} & \dfrac{5s+4}{s^2+3s+6} \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} \tag{4.18}$$

olarak seçilmiştir (Sagara ve Zhao 1989). Bu sistemin her bir çıkışı için polinomal denklemler s -domeninde aşağıdaki gibi yazılabilir.

$$(s^2 + a_{11}s + a_{12})X_1(s) = (b_{111}s + b_{112})U_1(s) + (b_{121}s + b_{122})U_2(s) \tag{4.19}$$

$$(s^2 + a_{21}s + a_{22})X_2(s) = (b_{211}s + b_{212})U_1(s) + (b_{221}s + b_{222})U_2(s) \tag{4.20}$$

Bu denklemlere Johansson (1993,1994) tarafından önerilen ve

$$\lambda = \frac{1}{1+s\tau} \quad \leftrightarrow \quad s = \frac{1-\lambda}{\lambda\tau} \tag{4.21}$$

olarak tanımlanan operatör değişimi aşağıdaki gibi uygulanabilir.

$$\left[\left(\frac{1-\lambda}{\lambda\tau}\right)^2+a_{11}\left(\frac{1-\lambda}{\lambda\tau}\right)+a_{12}\right]X_1(s)=\left[b_{111}\left(\frac{1-\lambda}{\lambda\tau}\right)+b_{112}\right]U_1(s)+\left[b_{121}\left(\frac{1-\lambda}{\lambda\tau}\right)+b_{122}\right]U_2(s) \qquad (4.22)$$

$$\left[\left(\frac{1-\lambda}{\lambda\tau}\right)^2+a_{21}\left(\frac{1-\lambda}{\lambda\tau}\right)+a_{22}\right]X_2(s)=\left[b_{211}\left(\frac{1-\lambda}{\lambda\tau}\right)+b_{212}\right]U_1(s)+\left[b_{221}\left(\frac{1-\lambda}{\lambda\tau}\right)+b_{222}\right]U_2(s) \qquad (4.23)$$

Bu denklemler payda eşitlenerek alçak geçiren filtre operatörü λ'ya göre düzenlendiğinde

$$\begin{aligned}[1+(-2+a_{11}\tau)\lambda+(1-a_{11}\tau+a_{12}\tau^2)\lambda^2]X_1(s)=&[(b_{111}\tau)\lambda+(-b_{111}\tau+b_{112}\tau^2)\lambda^2]U_1(s)\\&+[(b_{121}\tau)\lambda+(-b_{121}\tau+b_{122}\tau^2)\lambda^2]U_2(s)\end{aligned} \qquad (4.24)$$

$$\begin{aligned}[1+(-2+a_{21}\tau)\lambda+(1-a_{21}\tau+a_{22}\tau^2)\lambda^2]X_2(s)=&[(b_{211}\tau)\lambda+(-b_{211}\tau+b_{212}\tau^2)\lambda^2]U_1(s)\\&+[(b_{221}\tau)\lambda+(-b_{221}\tau+b_{222}\tau^2)\lambda^2]U_2(s)\end{aligned} \qquad (4.25)$$

eşitlikleri elde edilir. Bu değiştirilmiş sürekli-zaman modelin katsayıları s-domenindeki modelin katsayılarından farklı olarak aşağıdaki gibi isimlendirilebilir.

$$[1+\alpha_{11}\lambda+\alpha_{12}\lambda^2]X_1(s)=[\beta_{111}\lambda+\beta_{112}\lambda^2]U_1(s)+[\beta_{121}\lambda+\beta_{122}\lambda^2]U_2(s) \qquad (4.26)$$

$$[1+\alpha_{21}\lambda+\alpha_{22}\lambda^2]X_2(s)=[\beta_{211}\lambda+\beta_{212}\lambda^2]U_1(s)+[\beta_{221}\lambda+\beta_{222}\lambda^2]U_2(s) \qquad (4.27)$$

Sistemin giriş ve çıkış işaretleri ölçümleri ayrık olduğu için alçak geçiren filtreleme işleminin ayrık olarak yapıldığı ve sistemin çıkış işareti ölçümlerine ölçme gürültüsü karıştığı göz önüne alınarak aynı denklemleri aşağıdaki gibi yazabiliriz.

$$y_1(k)=-\alpha_{11}\lambda y_1(k)-\alpha_{12}\lambda^2y_1(k)+\beta_{111}\lambda u_1(k)+\beta_{112}\lambda^2u_1(k)+\beta_{121}\lambda u_2(k)+\beta_{122}\lambda^2u_2(k)+v_1(k) \qquad (4.28)$$

$$y_2(k)=-\alpha_{21}\lambda y_2(k)-\alpha_{22}\lambda^2y_2(k)+\beta_{211}\lambda u_1(k)+\beta_{212}\lambda^2u_1(k)+\beta_{221}\lambda u_2(k)+\beta_{222}\lambda^2u_2(k)+v_2(k) \qquad (4.29)$$

Bu denklemleri sistem tanıma işleminde kullanabilmek için doğrusal bağlanımlı biçimde

$$y_1(k)=\varphi_1^T(k)\theta_1+v_1(k) \qquad (4.30)$$

$$y_2(k) = \varphi_2^T(k)\theta_2 + v_2(k) \tag{4.31}$$

olarak yazdığımızda bu denklemlerdeki veri vektörleri ve parametre vektörleri

$$\varphi_1^T(k) = [-\lambda y_1(k) \quad -\lambda^2 y_1(k) \quad \lambda u_1(k) \quad \lambda^2 u_1(k) \quad \lambda u_2(k) \quad \lambda^2 u_2(k)] \tag{4.32}$$

$$\varphi_2^T(k) = [-\lambda y_2(k) \quad -\lambda^2 y_2(k) \quad \lambda u_1(k) \quad \lambda^2 u_1(k) \quad \lambda u_2(k) \quad \lambda^2 u_2(k)] \tag{4.33}$$

$$\theta_{1opt}^T = [\alpha_{11} \quad \alpha_{12} \quad \beta_{111} \quad \beta_{112} \quad \beta_{121} \quad \beta_{122}] \tag{4.34}$$

$$\theta_{2opt}^T = [\alpha_{21} \quad \alpha_{22} \quad \beta_{211} \quad \beta_{212} \quad \beta_{221} \quad \beta_{222}] \tag{4.35}$$

olarak yazılabilir. Sisteme ait transfer fonksiyonları ikinci derece olduğu için filtreleme işleminde de en fazla ikinci derece filtre kullanılmaktadır. Burada giriş ve çıkış işaretleri alçak geçiren filtrenin fark denklem modeli kullanılarak aşağıdaki gibi yapılabilir.

$$\lambda y_1(k) = p.\lambda y_1(k-1) + q.\big(y_1(k) + y_1(k-1)\big) \tag{4.36}$$

$$\lambda^2 y_1(k) = p.\lambda^2 y_1(k-1) + q.\big(\lambda y_1(k) + \lambda y_1(k-1)\big) \tag{4.37}$$

$$\lambda y_2(k) = p.\lambda y_2(k-1) + q.\big(y_2(k) + y_2(k-1)\big) \tag{4.38}$$

$$\lambda^2 y_2(k) = p.\lambda^2 y_2(k-1) + q.\big(\lambda y_2(k) + \lambda y_2(k-1)\big) \tag{4.39}$$

$$\lambda u_1(k) = p.\lambda u_1(k-1) + q.\big(u_1(k) + u_1(k-1)\big) \tag{4.40}$$

$$\lambda^2 u_1(k) = p.\lambda^2 u_1(k-1) + q.\big(\lambda u_1(k) + \lambda u_1(k-1)\big) \tag{4.41}$$

$$\lambda u_2(k) = p.\lambda u_2(k-1) + q.\big(u_2(k) + u_2(k-1)\big) \tag{4.42}$$

$$\lambda^2 u_2(k) = p.\lambda^2 u_2(k-1) + q.\big(\lambda u_2(k) + \lambda u_2(k-1)\big) \tag{4.43}$$

Filtreleme işleminde başlangıç koşulları sıfır alınabilir. Filtreler kararlı olduğu için başlangıç koşullarının etkisi zamanla kaybolacaktır. Filtreleme işlemiyle elde edilen filtrelenmiş giriş-çıkış verileri yukarıda verilen veri vektörlerinde kullanılarak değiştirilmiş model katsayıları tahmin edilir. Değiştirilmiş model katsayılarının tahmin edilmesinde tekrarlamalı GS algoritması aşağıdaki gibi kullanılabilir.

$$R_1(k) = \bar{\lambda}\, R_1(k-1) + \varphi_1(k)\varphi_1^T(k) \quad , \quad p_1(k) = \bar{\lambda}\, p_1(k-1) + \varphi_1(k) y_1(k) \tag{4.44}$$

$$\hat{\theta}_{1i}(k) = \left[p_{1i}(k) - \sum_{j=1}^{i-1} R_{1ij}(k)\hat{\theta}_{1j}(k) - \sum_{j=i+1}^{M_1} R_{1ij}(k)\hat{\theta}_{1j}(k-1) \right] \Big/ R_{1ii}(k) \tag{4.45}$$

$(i = 1,\ldots,M_1)$

$$R_2(k) = \lambda\, R_2(k-1) + \varphi_2(k)\varphi_2^T(k) \quad , \quad p_2(k) = \lambda\, p_2(k-1) + \varphi_2(k)y_2(k) \tag{4.46}$$

$$\hat{\theta}_{2i}(k) = \left[p_{2i}(k) - \sum_{j=1}^{i-1} R_{2ij}(k)\hat{\theta}_{2j}(k) - \sum_{j=i+1}^{M_2} R_{2ij}(k)\hat{\theta}_{2j}(k-1) \right] \Big/ R_{2ii}(k) \tag{4.47}$$

$(i = 1,\ldots,M_2)$

Değiştirilmiş model katsayılarının tahmin edilmesinde kullanılabilecek olan çok-değişkenli RLS algoritması da aşağıdaki gibi yazılabilir (Sinha ve Kuszta 1983, Söderström ve Stoica 1989, Ljung 1999).

$$\hat{\theta}_j(k) = \hat{\theta}_j(k-1) + L_j(k)[y_j(k) - \varphi_j^T(k)\hat{\theta}_j(k-1)] \quad , \quad (j = 1,2) \tag{4.48}$$

$$L_j(k) = \frac{P_j(k-1)\varphi_j(k)}{\lambda + \varphi_j(k)P_j(k-1)\varphi_j^T(k)} \tag{4.49}$$

$$P_j(k) = (1/\lambda)[I - L_j(k)\varphi_j^T(k)]P_j(k-1) \tag{4.50}$$

Yansız parametre tahminlerini elde edebilmek için yardımcı değişkenlerden yararlanılmıştır. Yardımcı değişken olarak kullanılabilecek olan, sistemle aynı yapıya sahip bir uyarlamalı modelin çıkış işaretleri, yani tahmin edilen gürültüsüz çıkış işaretleri aşağıdaki gibi hesaplanabilir. Bu değerler hesaplanırken eldeki en son parametre tahminleri kullanılır.

$$\begin{aligned} \hat{x}_1(k) = \Big(&- \hat{\alpha}_{11}\overline{\lambda}\hat{x}_1(k) - \hat{\alpha}_{12}\overline{\lambda}^2\hat{x}_1(k) + \hat{\beta}_{111}\lambda u_1(k) + \hat{\beta}_{112}\lambda^2 u_1(k) \\ &+ \hat{\beta}_{121}\lambda u_2(k) + \hat{\beta}_{122}\lambda u_1(k)\Big) \Big/ (1 + \hat{\alpha}_{11}q + \hat{\alpha}_{12}q^2) \end{aligned} \tag{4.51}$$

$$\begin{aligned} \hat{x}_2(k) = \Big(&- \hat{\alpha}_{21}\overline{\lambda}\hat{x}_2(k) - \hat{\alpha}_{22}\overline{\lambda}^2\hat{x}_2(k) + \hat{\beta}_{211}\lambda u_1(k) + \hat{\beta}_{212}\lambda^2 u_1(k) \\ &+ \hat{\beta}_{221}\lambda u_2(k) + \hat{\beta}_{222}\lambda u_1(k)\Big) \Big/ (1 + \hat{\alpha}_{21}q + \hat{\alpha}_{22}q^2) \end{aligned} \tag{4.52}$$

Burada kullanılan filtrelenmiş gürültüsüz çıkış işaretleri k anındaki değeri eksik olarak

$$\bar{\lambda}\hat{x}_1(k) = p.\lambda\hat{x}_1(k-1) + q.\big(\hat{x}_1(k-1)\big) \tag{4.53}$$

$$\bar{\lambda}^2\hat{x}_1(k) = p.\lambda^2\hat{x}_1(k-1) + q.\big(\bar{\lambda}\hat{x}_1(k) + \lambda\hat{x}_1(k-1)\big) \tag{4.54}$$

$$\bar{\lambda}\hat{x}_2(k) = p.\lambda\hat{x}_2(k-1) + q.\big(\hat{x}_2(k-1)\big) \tag{4.55}$$

$$\bar{\lambda}^2\hat{x}_2(k) = p.\lambda^2\hat{x}_2(k-1) + q.\big(\bar{\lambda}\hat{x}_2(k) + \lambda\hat{x}_2(k-1)\big) \tag{4.56}$$

şeklinde hesaplanabilir. Filtrelenmiş giriş işaretleri ise (4.40)-(4.43) eşitliklerindeki gibi hesaplanabilir. Gürültüsüz çıkış işaretlerinin k anındaki değerleri (4.51) ve (4.52) eşitlikleriyle tahmin edildikten sonra, aşağıdaki eşitlikler yardımıyla k anındaki filtrelenmiş değerleri

$$\lambda\hat{x}_1(k) = \bar{\lambda}\hat{x}_1(k) + q.\hat{x}_1(k) \quad , \quad \lambda^2\hat{x}_1(k) = \bar{\lambda}^2\hat{x}_1(k) + q^2.\hat{x}_1(k) \tag{4.57}$$

$$\lambda\hat{x}_2(k) = \bar{\lambda}\hat{x}_2(k) + q.\hat{x}_2(k) \quad , \quad \lambda^2\hat{x}_2(k) = \bar{\lambda}^2\hat{x}_2(k) + q^2.\hat{x}_2(k) \tag{4.58}$$

şeklinde hesaplanabilir ve hesaplanan bu değerler yardımcı modelin veri vektörlerinin oluşturulmasında kullanılır. Yardımcı modelin veri vektörleri

$$z_1^T(k) = [-\lambda\hat{x}_1(k) \quad -\lambda^2\hat{x}_1(k) \quad \lambda u_1(k) \quad \lambda^2 u_1(k) \quad \lambda u_2(k) \quad \lambda^2 u_2(k)] \tag{4.59}$$

$$z_2^T(k) = [-\lambda\hat{x}_2(k) \quad -\lambda^2\hat{x}_2(k) \quad \lambda u_1(k) \quad \lambda^2 u_1(k) \quad \lambda u_2(k) \quad \lambda^2 u_2(k)] \tag{4.60}$$

şeklinde oluşturulabilir. Burada oluşturulan (gürültü içermeyen) yardımcı veri vektörleri tekrarlamalı GSYD ve tekrarlamalı YD algoritmalarında aşağıdaki gibi kullanılabilir. Değiştirilmiş model katsayılarının tahmin edilmesinde tekrarlamalı GSYD algoritması aşağıdaki gibi kullanılabilir.

$$R_1(k) = \lambda R_1(k-1) + z_1(k)\varphi_1^T(k) \quad , \quad p_1(k) = \lambda p_1(k-1) + z_1(k)y_1(k) \tag{4.61}$$

$$\hat{\theta}_{1i}(k) = \left[p_{1i}(k) - \sum_{j=1}^{i-1} R_{1ij}(k)\hat{\theta}_{1j}(k) - \sum_{j=i+1}^{M_1} R_{1ij}(k)\hat{\theta}_{1j}(k-1) \right] \Big/ R_{1ii}(k) \ , \quad (i = 1,\ldots,M_1) \tag{4.62}$$

$$R_2(k) = \lambda R_2(k-1) + z_2(k)\varphi_2^T(k) \quad , \quad p_2(k) = \lambda p_2(k-1) + z_2(k)y_2(k) \tag{4.63}$$

$$\hat{\theta}_{2i}(k) = \left[p_{2i}(k) - \sum_{j=1}^{i-1} R_{2ij}(k)\hat{\theta}_{2j}(k) - \sum_{j=i+1}^{M_2} R_{2ij}(k)\hat{\theta}_{2j}(k-1) \right] \Big/ R_{2ii}(k), \quad (i = 1,\ldots,M_2) \tag{4.64}$$

Değiştirilmiş model katsayılarının tahmin edilmesinde kullanılabilecek olan çok-değişkenli tekrarlamalı YD algoritması da aşağıdaki gibi yazılabilir (Sinha ve Kuszta 1983, Söderström ve Stoica 1989, Ljung 1999).

$$\hat{\theta}_j(k) = \hat{\theta}_j(k-1) + L_j(k)[y_j(k) - \varphi_j^T(k)\hat{\theta}_j(k-1)] \quad , \quad (j = 1,2) \tag{4.65}$$

$$L_j(k) = \frac{P_j(k-1)z_j(k)}{\lambda + \varphi_j(k)P_j(k-1)z_j^T(k)} \tag{4.66}$$

$$P_j(k) = (1/\lambda)[I - L_j(k)\varphi_j^T(k)]P_j(k-1) \tag{4.67}$$

Bu simülasyon çalışmasında tahmin edilen parametre sayısı, parametre vektörlerinin boyutları $M_1 = M_2 = 6$ olmak üzere toplam 12'dir. Unutma faktörü $\lambda = 1$ olarak alınmıştır.

Tahmin edilen değiştirilmiş model katsayıları ile s-domenindeki modelin katsayıları arasında aşağıdaki matris ilişkileri yazılabilir.

$$\begin{bmatrix} \tau & 0 & 0 & 0 & 0 & 0 \\ -\tau & \tau^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \tau & 0 & 0 & 0 \\ 0 & 0 & -\tau & \tau^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \tau & 0 \\ 0 & 0 & 0 & 0 & -\tau & \tau^2 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ b_{111} \\ b_{112} \\ b_{121} \\ b_{122} \end{bmatrix} + \begin{bmatrix} -2 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \alpha_{11} \\ \alpha_{12} \\ \beta_{111} \\ \beta_{112} \\ \beta_{121} \\ \beta_{122} \end{bmatrix} \tag{4.68}$$

$$\begin{bmatrix} \tau & 0 & 0 & 0 & 0 & 0 \\ -\tau & \tau^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \tau & 0 & 0 & 0 \\ 0 & 0 & -\tau & \tau^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \tau & 0 \\ 0 & 0 & 0 & 0 & -\tau & \tau^2 \end{bmatrix} \begin{bmatrix} a_{21} \\ a_{22} \\ b_{211} \\ b_{212} \\ b_{221} \\ b_{222} \end{bmatrix} + \begin{bmatrix} -2 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \alpha_{21} \\ \alpha_{22} \\ \beta_{211} \\ \beta_{212} \\ \beta_{221} \\ \beta_{222} \end{bmatrix} \tag{4.69}$$

Parametre tahmin işlemi değiştirilmiş model kullanılarak yapılmaktadır ve yukarıdaki matrisel denklemler kullanılarak s-domenindeki modelin katsayıları elde edilebilmektedir. İki modelin katsayıları arasındaki bu matrisel ifadeleri ve bu katsayılar arasındaki dönüşüm işlemini kapalı formda aşağıdaki gibi yazabiliriz.

$$F\theta_{s1} + G = \theta_1 \quad \leftrightarrow \quad \theta_{s1} = F^{-1}(\theta_1 - G) \tag{4.70}$$

$$F\theta_{s2} + G = \theta_2 \quad \leftrightarrow \quad \theta_{s2} = F^{-1}(\theta_2 - G) \tag{4.71}$$

Tahmin işleminden sonra dönüşüm yapılarak elde edilmek istenen s-domenindeki modelin katsayı vektörlerinin sayısal değerleri aşağıdaki gibidir.

$$\theta_{s1opt}^T = \begin{bmatrix} a_{11} & a_{12} & b_{111} & b_{112} & b_{121} & b_{122} \end{bmatrix} = \begin{bmatrix} 2.8 & 4.0 & 5.0 & 5.0 & 0.0 & 2.0 \end{bmatrix} \tag{4.72}$$

$$\theta_{s2opt}^T = \begin{bmatrix} a_{21} & a_{22} & b_{211} & b_{212} & b_{221} & b_{222} \end{bmatrix} = \begin{bmatrix} 3.0 & 6.0 & 0.0 & 3.0 & 5.0 & 4.0 \end{bmatrix} \tag{4.73}$$

Yapılan simülasyon çalışmasında giriş işareti olarak sinüzoidal işaretlerin toplamından oluşan aşağıdaki giriş işaretleri kullanılmıştır.

$$u_1(t) = \sin(t) + 1.5\sin(1.5t) + 2.5\sin(2.5t) + 3.5\sin(3.5t) \tag{4.74}$$

$$u_2(t) = \sin(t) + 1.5\sin(2t) + 2\sin(3t) + 3\sin(4t) + 4\sin(5t) \tag{4.75}$$

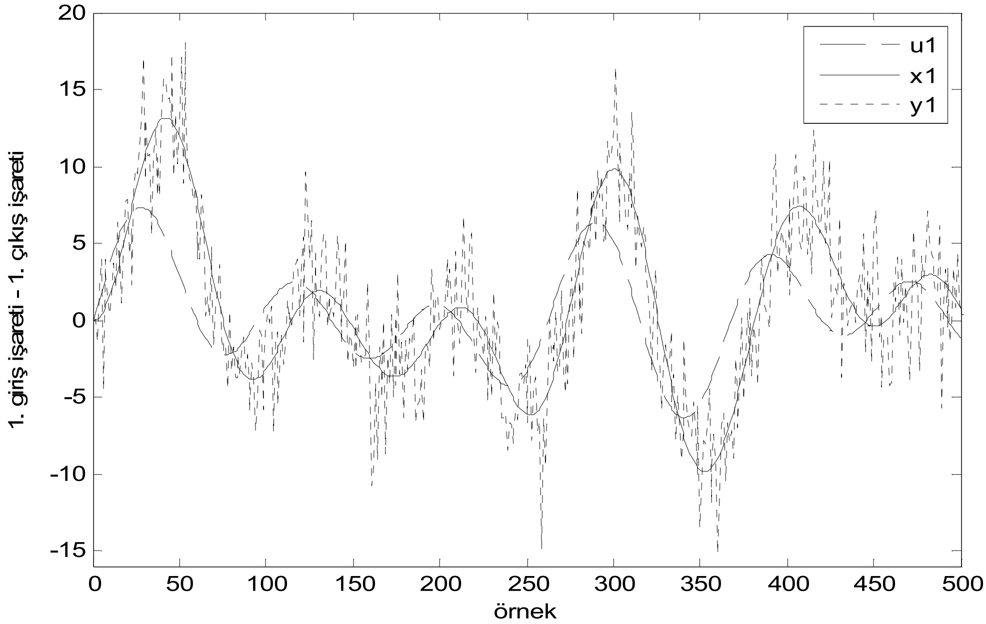
Bu giriş işaretleri kullanılarak yapılan giriş-çıkış simülasyonu sonrasında elde edilen çıkış işaretlerine ölçme gürültüsü olarak aşağıdaki ARMA (Auto-Regressive Moving Average) model kullanılarak elde edilen renkli gürültü dizileri ilave edilmiştir.

$$v_1(k) = \frac{1 - 1.5z^{-1} + 0.5725z^{-2}}{1 - 1.6z^{-1} + 0.68z^{-2}} e_1(k) \quad , \quad v_2(k) = \frac{1 - 1.5z^{-1} + 0.5725z^{-2}}{1 - 1.6z^{-1} + 0.68z^{-2}} e_2(k) \tag{4.76}$$

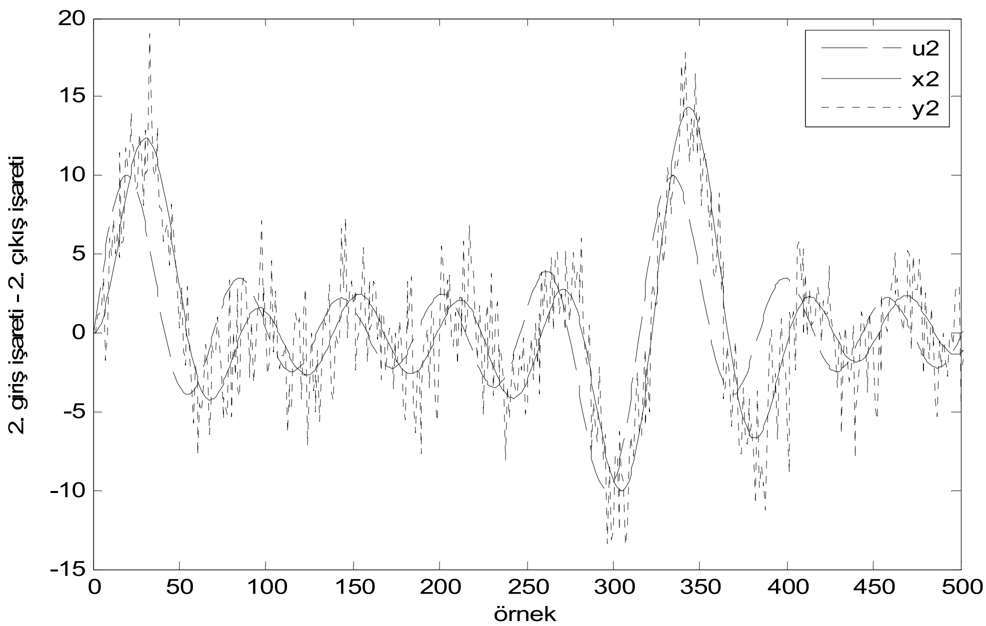
$$v_1(k) = 1.6v_1(k-1) - 0.68v_1(k-2) + e_1(k) - 1.5e_1(k-1) + 0.5725e_1(k-2) \tag{4.77}$$

$$v_2(k) = 1.6v_2(k-1) - 0.68v_2(k-2) + e_2(k) - 1.5e_2(k-1) + 0.5725e_2(k-2) \tag{4.78}$$

Burada $e_1(k)$ ve $e_2(k)$ sıfır ortalamalı ve varyansları 1 olan normal dağılıma sahip rastgele gürültü dizileridir. Burada $v_1(k)$ ve $v_2(k)$ renkli ölçme gürültülerinin varyansları 1. ve 2. çıkış işaretleri için gürültü işaret oranı %50 olacak şekilde ayarlanmıştır. Sistem tanıma işleminde kullanılan giriş-çıkış işaretleri aşağıdaki şekillerde görülmektedir. Şekil 4.18'de 1. giriş işareti $u_1(k)$, gürültüsüz 1. çıkış işareti $x_1(k)$ ve gürültülü 1. çıkış işareti $y_1(k)$ görülmektedir. Şekil 4.19'da ise 2. giriş işareti $u_2(k)$, gürültüsüz 2. çıkış işareti $x_2(k)$ ve gürültülü 2. çıkış işareti $y_2(k)$ görülmektedir.

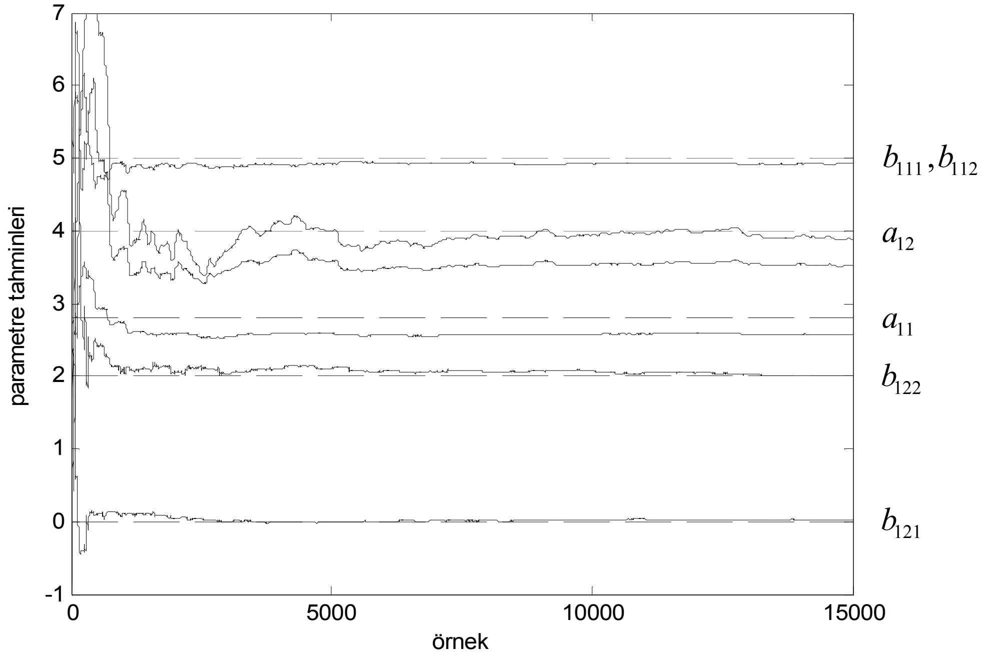


Şekil 4.18. Çok değişkenli sürekli-zaman sistemin 1. giriş ve 1. çıkış işaretleri.

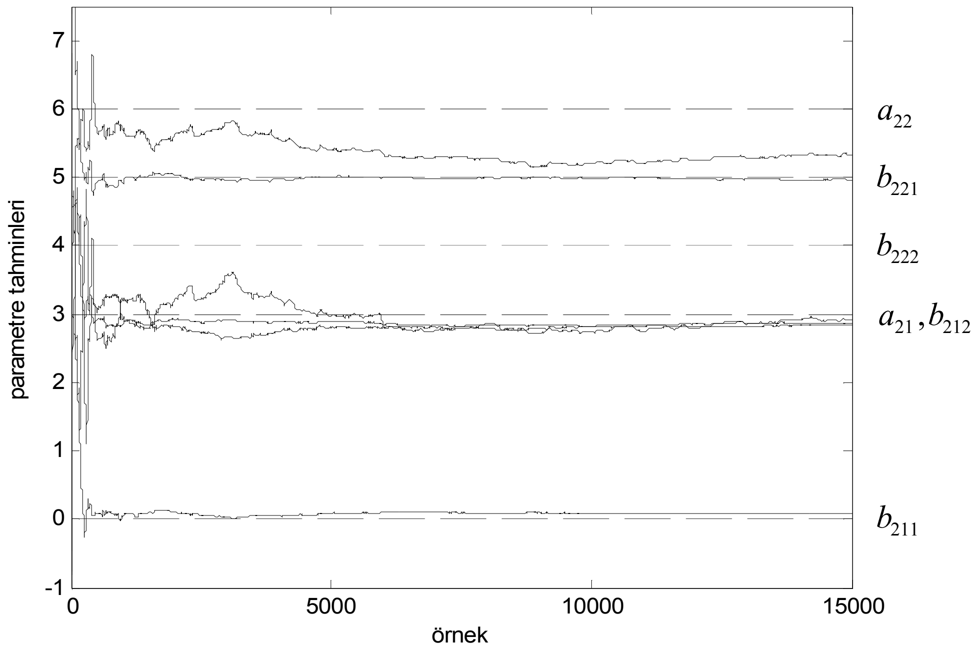


Şekil 4.19. Çok değişkenli sürekli -zaman sistemin 2. giriş ve 2. çıkış işaretleri.

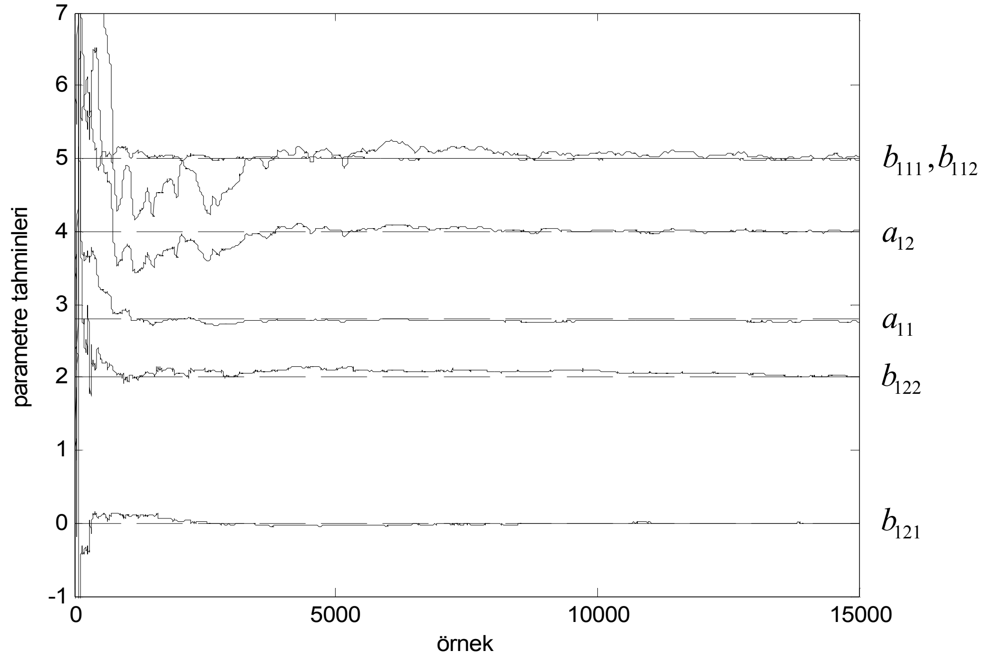
Bu durumda çok-değişkenli tekrarlamalı GS algoritmasıyla yapılan benzetim sonuçları Şekil 4.20'de ve Şekil 4.21'de görülmektedir. Yansız parametre tahminlerini elde edebilmek için, önerilen çok-değişkenli tekrarlamalı GSYD algoritmasıyla yapılan simülasyon sonucunda elde edilen parametre tahminleri Şekil 4.22'de ve Şekil 4.23'te görülmektedir. Literatürde yaygın olarak kullanılan çok-değişkenli tekrarlamalı YD algoritmasıyla yapılan simülasyon sonucunda elde edilen parametre tahminleri de Şekil 4.24'de ve Şekil 4.25'te görülmektedir. Kullanılan bütün algoritmalarda korelasyon matrisinin başlangıç değerleri birim matris olarak, parametre tahminlerinin başlangıç değerleri ise sıfır vektör olarak alınmıştır ve parametreler sabit olduğu için unutma faktörü $\lambda = 1$ alınmıştır. Burada kesikli düz çizgiler doğru parametre değerlerini göstermektedir.



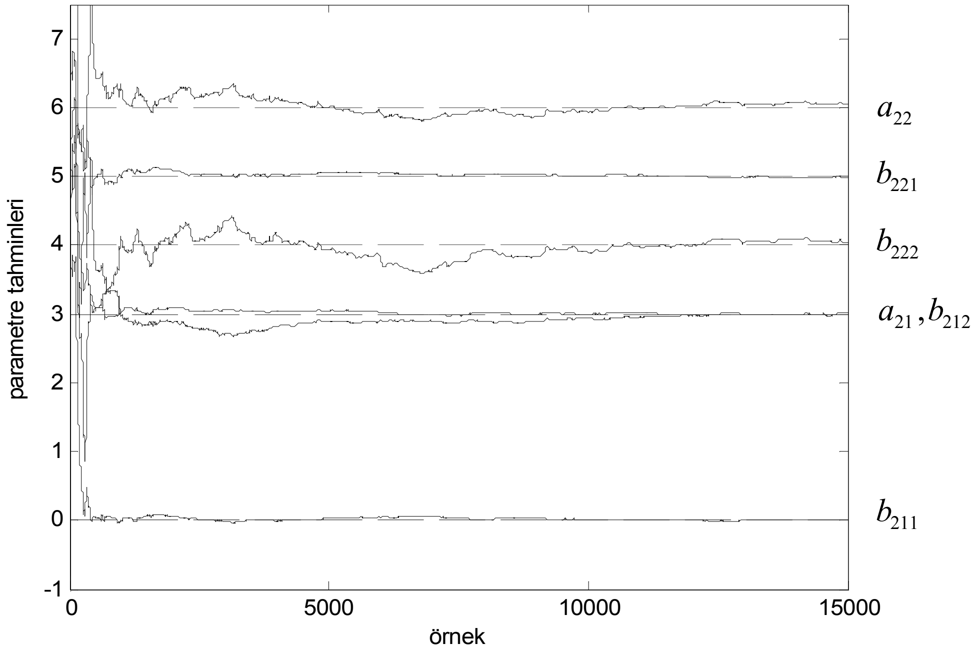
Şekil 4.20. Tekrarlamalı GS algoritmasıyla hesaplanan, çok-değişkenli sürekli-zaman sistemin $\hat{\theta}_1$ parametre tahminleri.



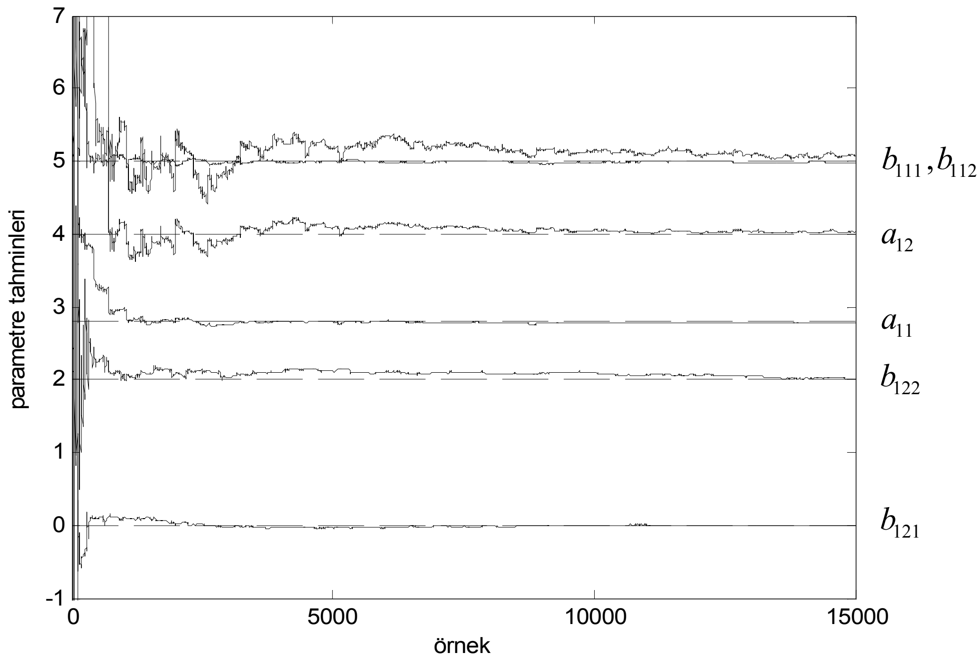
Şekil 4.21. Tekrarlamalı GS algoritmasıyla hesaplanan, çok-değişkenli sürekli-zaman sistemin $\hat{\theta}_2$ parametre tahminleri.



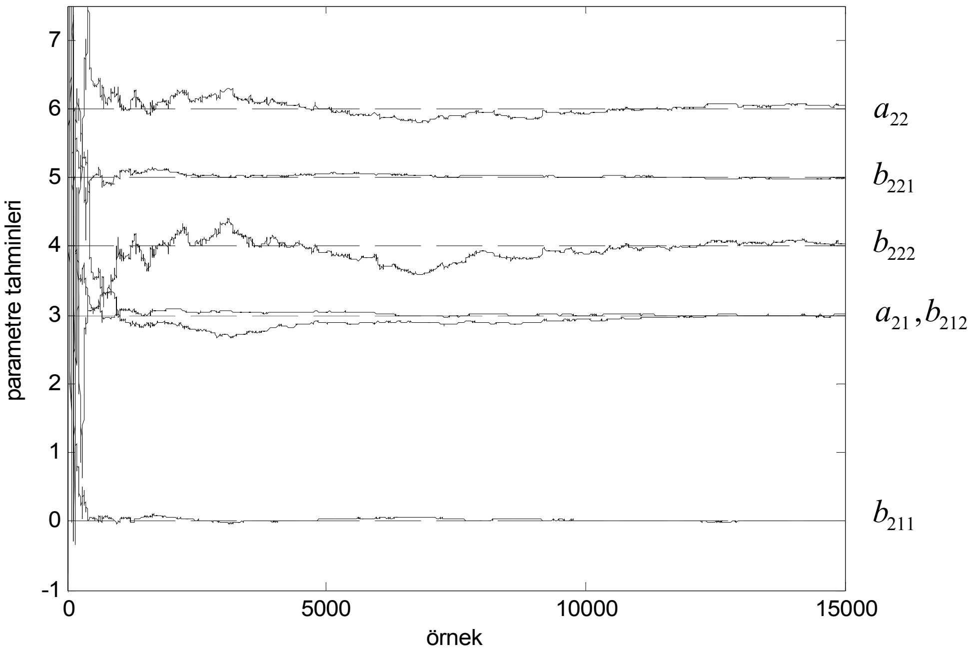
Şekil 4.22. Tekrarlamalı GSYD algoritmasıyla hesaplanan, çok-değişkenli sürekli-zaman sistemin $\hat{\theta}_1$ parametre tahminleri.



Şekil 4.23. Tekrarlamalı GSYD algoritmasıyla hesaplanan, çok-değişkenli sürekli-zaman sistemin $\hat{\theta}_2$ parametre tahminleri.



Şekil 4.24. Tekrarlamalı YD algoritmasıyla hesaplanan, çok-değişkenli sürekli-zaman sistemin $\hat{\theta}_1$ parametre tahminleri.



Şekil 4.25. Tekrarlamalı GSYD algoritmasıyla hesaplanan, çok-değişkenli sürekli-zaman sistemin $\hat{\theta}_2$ parametre tahminleri.

Yapılan simülasyon sonuçlarında 15000 adım sonunda parametre tahminlerinin yakınsadığı sayısal değerler Çizelge 4.5'te verilmiştir.

Çizelge 4.5. Çok-girişli çok-çıkışlı sürekli-zaman sistemin parametre tahminlerinin yakınsadığı değerler.

Algoritma:	$\hat{a}_{11}$	$\hat{a}_{12}$	$\hat{b}_{111}$	$\hat{b}_{112}$	$\hat{b}_{121}$	$\hat{b}_{122}$
GS	2.5654	3.5197	4.9203	3.8777	0.0189	1.9964
GSYD	2.7646	4.0098	4.9861	5.0281	-0.0061	2.0084
YD	2.7678	4.0296	4.9829	5.0657	-0.0070	2.0103
Doğru değer	2.8	4.0	5.0	5.0	0.0	2.0
Algoritma:	$\hat{a}_{21}$	$\hat{a}_{22}$	$\hat{b}_{211}$	$\hat{b}_{212}$	$\hat{b}_{221}$	$\hat{b}_{222}$
GS	2.8382	5.3349	0.0819	2.8764	4.9638	2.9118
GSYD	3.0047	6.0515	0.0086	2.9939	4.9890	4.0380
YD	3.0042	6.0458	0.0093	2.9931	4.9897	4.0302
Doğru değer	3.0	6.0	0.0	3.0	5.0	4.0

Çok-girişli çok-çıkışlı sürekli zaman sistemin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.6`da verilmiştir.

Çizelge 4.6. Çok-girişli çok-çıkışlı ayrık-zaman sistemin parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü ($\hat{\theta}_1$ için) (çarpma-bölme işlemi sayısı)	Yakınsama Hızı ($\hat{\theta}_1$ için) (yaklaşık örnek sayısı)
GS	56	500 (yanlı tahmin)
GSYD	60	500 (yansız tahmin)
YD	104	500 (yansız tahmin)
Algoritma:	İşlem Yükü ($\hat{\theta}_2$ için) (çarpma-bölme işlemi sayısı)	Yakınsama Hızı ($\hat{\theta}_2$ için) (yaklaşık örnek sayısı)
GS	56	500 (yanlı tahmin)
GSYD	60	500 (yansız tahmin)
YD	104	500 (yansız tahmin)

Elde edilen sonuçlara bakıldığında tekrarlamalı GS algoritması kullanıldığında, yardımcı değişkenlerden yararlanılmadığında 15000 adım sonunda yanlı parametre tahminlerinin elde edildiği görülmüştür. Tekrarlamalı GSYD ve tekrarlamalı YD algoritmaları kullanıldığında ise 15000 adım sonunda elde edilen parametre tahminlerinin doğru değerlere yakınsadığı görülmüştür. Elde edilen benzetim sonuçlarına genel olarak bakıldığında, önerilen tekrarlamalı GS ve GSYD algoritmalarının işlem yükünün RLS ve YD algoritmalarına göre daha az olmasının yanında RLS ve YD algoritmalarına çok yakın sonuçlar verdiği görülmüştür. Çok-girişli çok-çıkışlı sistemlerde tahmin edilen parametre sayısının arttığı göz önüne alındığında, önerilen çok-değişkenli tekrarlamalı GS ve GSYD algoritmalarının işlem yükü avantajı önem kazanmaktadır. İşlem yükü avantajının yanında aynı zamanda yakınsama hızı açısından çok-değişkenli RLS ve YD algoritmalarına çok yakın sonuçlar verdiği de göz önüne alındığında, işlem yükü fazla olan çok-değişkenli RLS ve YD algoritmalarına iyi bir alternatif olarak görülmektedir.

4.5. Sistem Modellemede Kullanılan Bazı Zaman Serileri Modellerinin Tekrarlamalı Gauss-Seidel ve Jacobi Algoritmaları ile Tahmin edilmesi

Bu kısımda, sistem parametrelerinin tahmin edilmesinde ve bazı uyarlamalı kontrol uygulamalarında kullanılan başlıca parametrik zaman serileri modellerinin parametreleri önerilen tekrarlamalı GS algoritmasıyla tahmin edilmiş ve diğer algoritmalarla karşılaştırmalı olarak incelenmiştir.

4.5.1. ARX model parametrelerinin tahmin edilmesi

Benzetim işleminde kullanılan örnek ikinci derece ARX modelin transfer fonksiyonu gösterimi aşağıdaki gibidir.

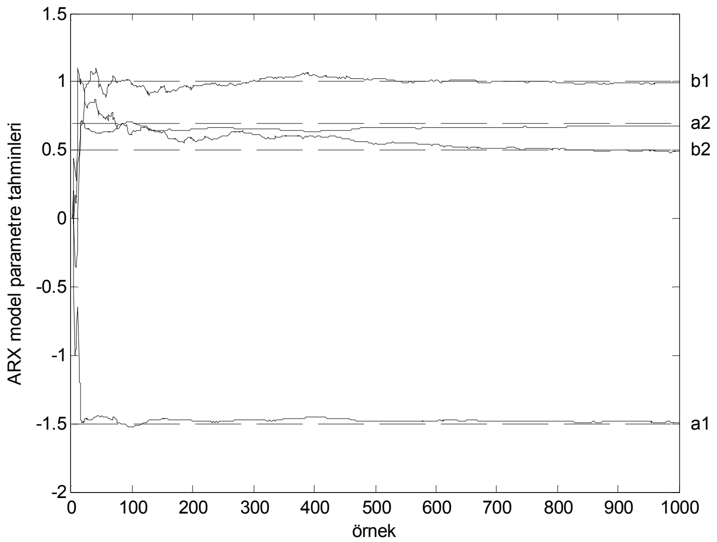
$$y(k) = \frac{z^{-1} + 0.5z^{-2}}{1 - 1.5z^{-1} + 0.7z^{-2}} u(k) + \frac{1}{1 - 1.5z^{-1} + 0.7z^{-2}} v(k) \tag{4.79}$$

Burada sistemin giriş işareti $u(k)$ ve bozucu giriş işareti $v(k)$ birbirinden bağımsız olan sıfır ortalamalı varyansı 1 olan normal dağılıma sahip rasgele gürültü dizileridir. Benzetim çalışmasında kullanılan optimum parametre vektörü ve veri vektörü aşağıdaki gibidir.

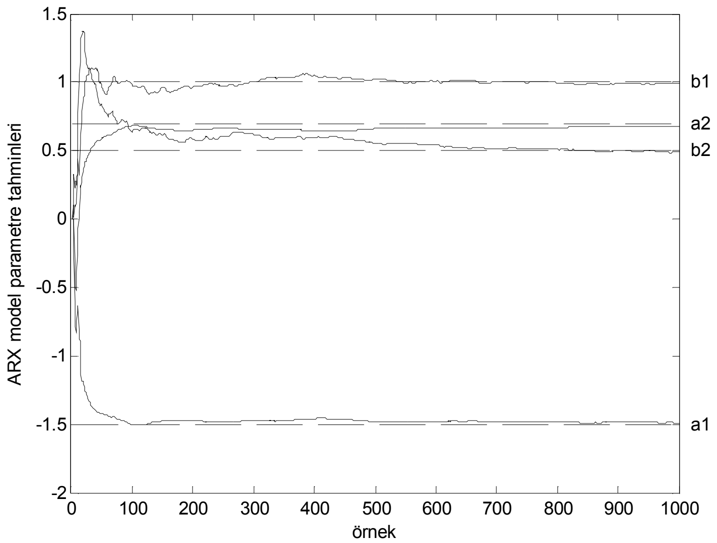
$$\theta_{opt}^T = \begin{bmatrix} a_1 & a_2 & b_1 & b_2 \end{bmatrix} = \begin{bmatrix} -1.5 & 0.7 & 1.0 & 0.5 \end{bmatrix} \tag{4.80}$$

$$\varphi^T(k) = \begin{bmatrix} -y(k-1) & -y(k-2) & u(k-1) & u(k-2) \end{bmatrix} \tag{4.81}$$

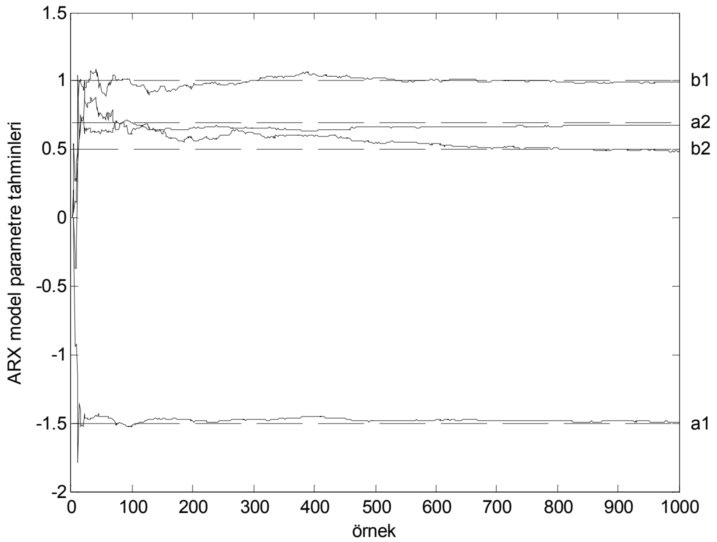
Tekrarlamalı GS ve Jacobi algoritmalarıyla ARX model katsayıları tahmin edildiğinde Şekil 4.26 ve Şekil 4.27’deki edilmiştir. Daha sonra RLS ve NLMS algoritmalarıyla elde edilen sonuçlar Şekil 4.28 ve Şekil 4.29’da verilmiştir. Burada korelasyon matrisinin kendisinin ve tersinin başlangıç değeri tekrarlamalı GS, Jacobi ve RLS algoritmalarında birim matris alınmıştır. Bu algoritmalarda unutma faktörü kullanılmamıştır. Adım parametresi tekrarlamalı Jacobi algoritmasında $\mu = 0.5$, NLMS algoritmasında $\mu = 0.15$ olarak alınmıştır. Bütün algoritmalarda parametre tahminlerinin başlangıç değerleri sıfır vektör alınmıştır.



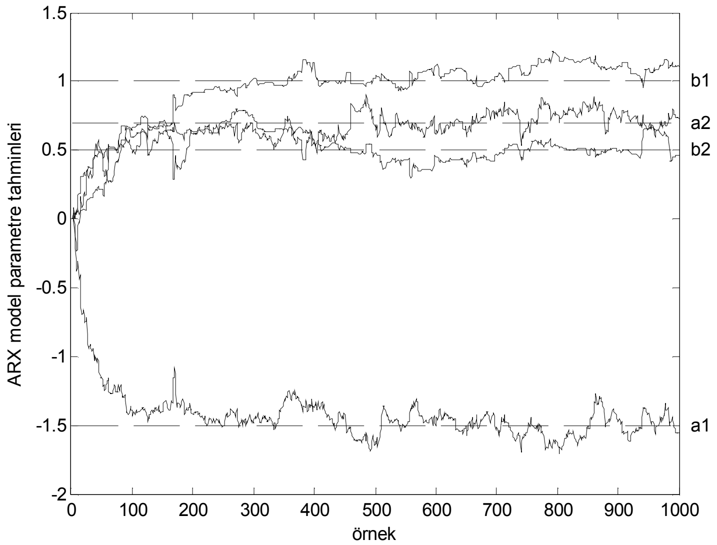
Şekil 4.26. Tekrarlamalı GS algoritması ile hesaplanan ARX model parametre tahminleri.



Şekil 4.27. Tekrarlamalı Jacobi algoritması ile hesaplanan ARX model parametre tahminleri.



Şekil 4.28. RLS algoritması ile hesaplanan ARX model parametre tahminleri.



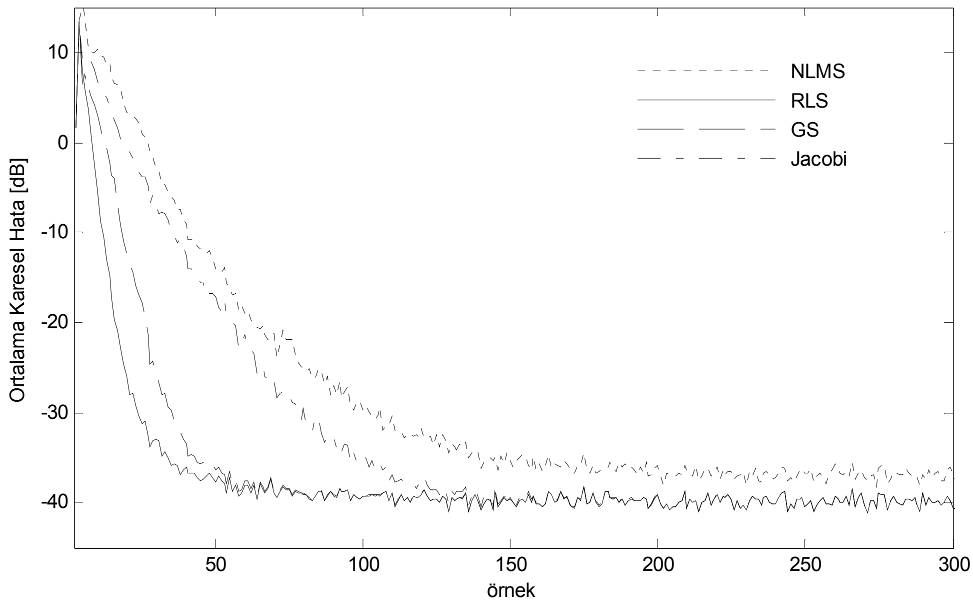
Şekil 4.29. NLMS algoritması ile hesaplanan ARX model parametre tahminleri.

ARX modelin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.7`de verilmiştir.

Çizelge 4.7. ARX model parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	56	10
Jacobi	64	100
RLS	100	10
NLMS	14	200

Daha sonra aynı algoritmalar 500 defa çalıştırılmış ve aşağıdaki Şekil 4.30'da görülen ortalama karesel hata eğrileri elde edilmiştir. Burada normal dağılıma sahip sıfır ortalamalı $e(k)$ bozucu girişinin varyansı 0.01 alınmıştır, yine normal dağılıma sahip sıfır ortalamalı $u(k)$ sistem girişinin varyansı 1 alınmıştır. NLMS algoritmasında $\mu = 0.3$ seçilmiştir. Tekrarlamalı Jacobi algoritmasında $\mu = 0.3$ seçilmiştir. RLS algoritmasında korelasyon matrisinin tersinin başlangıç değeri ve tekrarlamalı GS ve Jacobi algoritmalarında korelasyon matrisinin başlangıç değeri birim matris seçilmiştir. Bütün algoritmalarda parametre tahminlerinin başlangıç değerleri sıfır seçilmiştir. Şekil 4.30'a bakıldığında elde edilen tekrarlamalı GS algoritmasının performansının RLS algoritmasına çok yakın olduğu görülmektedir. Tekrarlamalı Jacobi algoritmasının ise adım parametresine bağlı olarak yakınsama hızının biraz daha düşük olduğu fakat NLMS algoritmasından daha hızlı yakınsadığı görülmektedir. Yapılan benzetim çalışmasında korelasyon matrisinin özdeğer yayılımı yaklaşık 59 olarak hesaplanmıştır. Burada rastgele gürültü şeklinde hızlı değişen işaretler kullanıldığı için korelasyon matrisinin özdeğer yayılımı küçük olmuştur. Bu yüzden algoritmaların yakınsama hızları arasındaki fark küçük olmaktadır.



Şekil 4.30. ARX model için elde edilen ortalama karesel hata eğrileri.

4.5.2. ARMAX model parametrelerinin tahmin edilmesi

Benzetim işleminde kullanılan örnek ikinci derece ARMAX modelin transfer fonksiyonu gösterimi aşağıdaki gibidir.

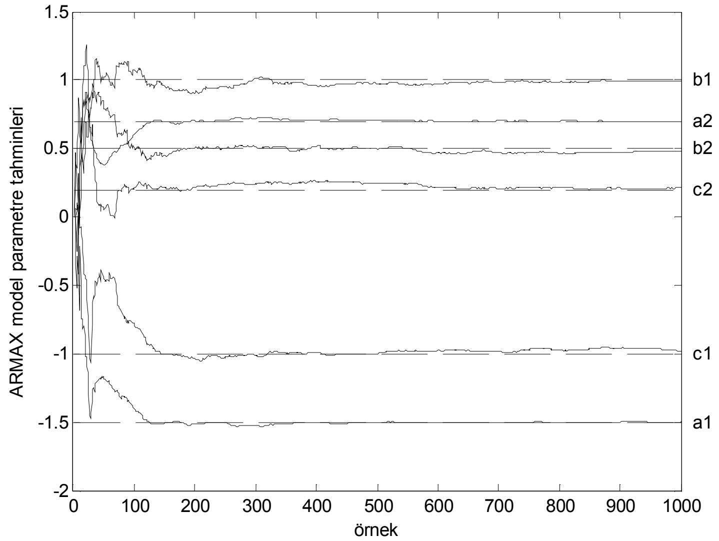
$$y(k) = \frac{z^{-1} + 0.5z^{-2}}{1 - 1.5z^{-1} + 0.7z^{-2}} u(k) + \frac{1 - z^{-1} + 0.2z^{-2}}{1 - 1.5z^{-1} + 0.7z^{-2}} e(k) \tag{4.82}$$

Burada sistemin giriş işareti $u(k)$ ve bozucu giriş işareti $e(k)$ birbirinden bağımsız olan sıfır ortalamalı varyansı 1 olan normal dağılıma sahip rastgele gürültü dizileridir. Benzetim çalışmasında kullanılan optimum parametre vektörü ve veri vektörü aşağıdaki gibidir.

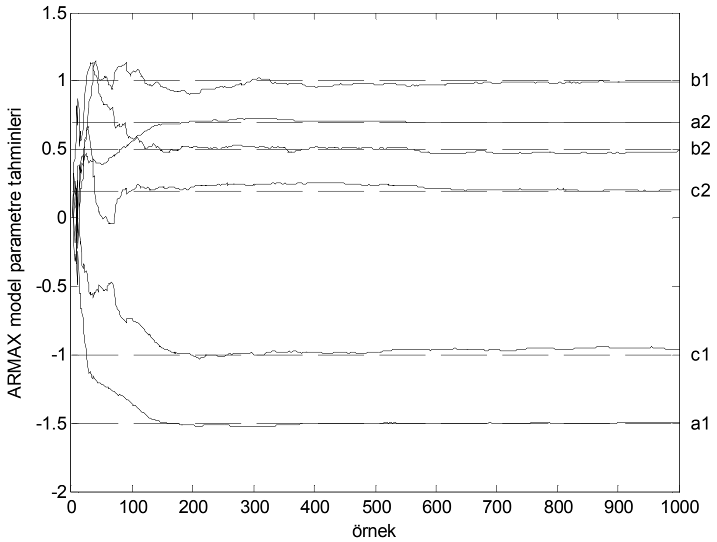
$$\theta_{opt}^T = \begin{bmatrix} a_1 & a_2 & b_1 & b_2 & c_1 & c_2 \end{bmatrix} = \begin{bmatrix} -1.5 & 0.7 & 1.0 & 0.5 & -1.0 & 0.2 \end{bmatrix} \tag{4.83}$$

$$\varphi^T(k) = \begin{bmatrix} -y(k-1) & -y(k-2) & u(k-1) & u(k-2) & \varepsilon(k-1) & \varepsilon(k-2) \end{bmatrix} \tag{4.84}$$

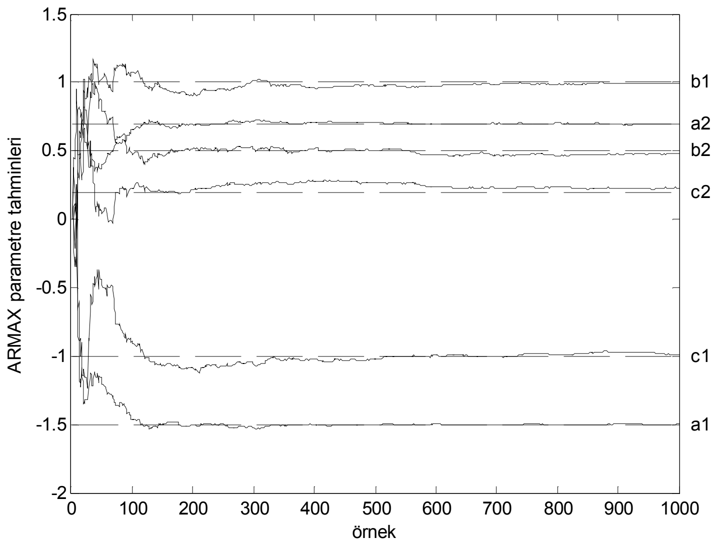
Benzetim çalışmasında öncelikle tekrarlamalı GS ve Jacobi algoritmalarıyla ARMAX model katsayıları tahmin edildiğinde Şekil 4.31 ve Şekil 4.32'deki sonuçlar elde edilmiştir. Daha sonra ELS (Extended Least Squares) ve NLMS algoritmalarıyla elde edilen sonuçlar Şekil 4.33 ve Şekil 4.34'te verilmiştir. Burada korelasyon matrisinin kendisinin ve tersinin başlangıç değeri tekrarlamalı GS, Jacobi ve ELS algoritmalarında birim matris alınmıştır. Bu algoritmalarda unutma faktörü kullanılmamıştır. Adım parametresi tekrarlamalı Jacobi algoritmasında $\mu = 0.5$, NLMS algoritmasında $\mu = 0.1$ olarak alınmıştır. Bütün algoritmalarda parametre tahminlerinin başlangıç değerleri sıfır vektör alınmıştır.



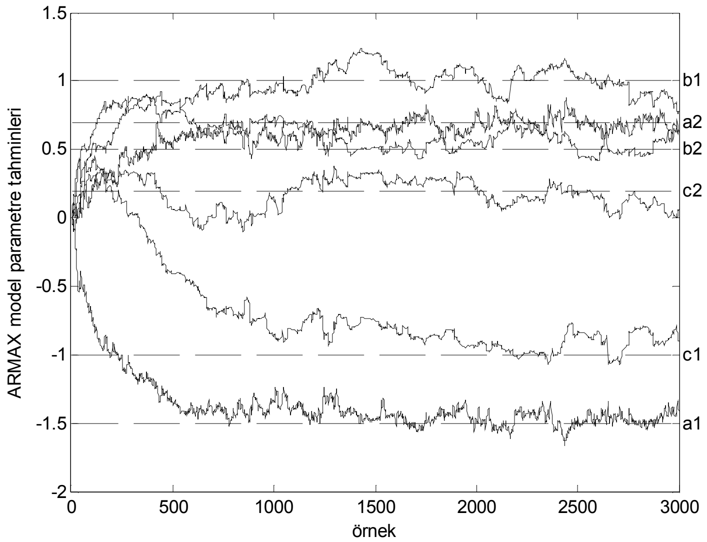
Şekil 4.31. Tekrarlamalı GS algoritması ile hesaplanan ARMAX model parametre tahminleri.



Şekil 4.32. Tekrarlamalı Jacobi algoritması ile hesaplanan ARMAX model parametre tahminleri.



Şekil 4.33. ELS algoritması ile hesaplanan ARMAX model parametre tahminleri.



Şekil 4.34. NLMS algoritması ile hesaplanan ARMAX model parametre tahminleri.

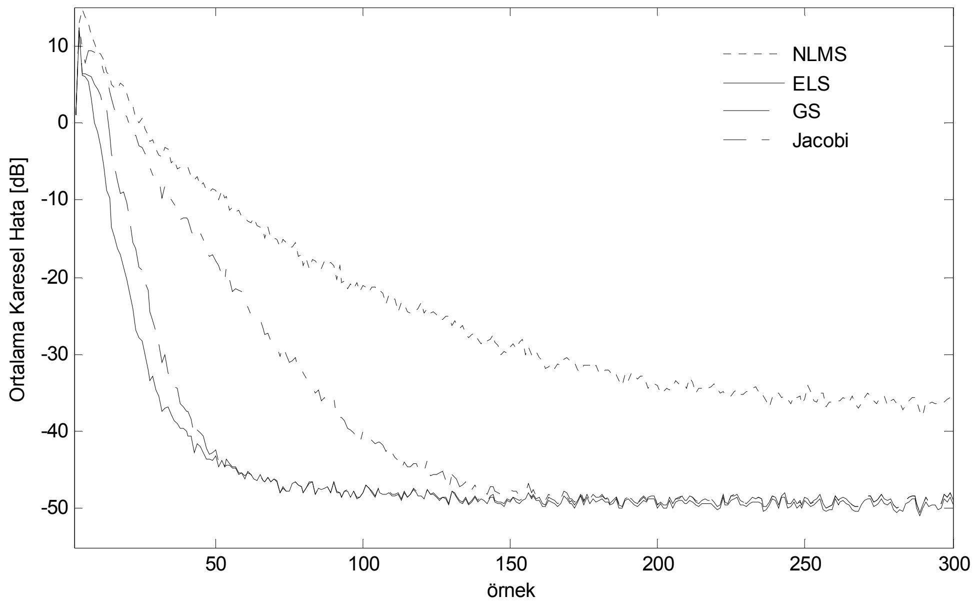
ARMAX modelin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.8`de verilmiştir.

Çizelge 4.8. ARMAX model parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	120	100
Jacobi	132	150
ELS	182	100
NLMS	20	600

Daha sonra aynı algoritmalar 500 defa çalıştırılmış ve aşağıdaki Şekil 4.35'te görülen ortalama karesel hata eğrileri elde edilmiştir. Burada normal dağılıma sahip sıfır ortalamalı $e(k)$ bozucu girişinin varyansı 0.001 alınmıştır, yine normal dağılıma sahip sıfır ortalamalı $u(k)$ sistem girişinin varyansı 1 alınmıştır. NLMS algoritmasında $\mu = 0.3$ seçilmiştir. Tekrarlamalı Jacobi algoritmasında $\mu = 0.3$ seçilmiştir. ELS algoritmasında korelasyon matrisinin tersinin başlangıç değeri ve tekrarlamalı GS ve Jacobi algoritmalarında korelasyon matrisinin başlangıç değeri birim matris seçilmiştir.

Bütün algoritmalarda parametre tahminlerinin başlangıç değerleri sıfır seçilmiştir. Şekil 4.35'e bakıldığında elde edilen tekrarlamalı GS algoritmasının performansının ELS algoritmasına çok yakın olduğu görülmektedir. Tekrarlamalı Jacobi algoritmasının ise adım parametresine bağlı olarak yakınsama hızının biraz daha düşük olduğu fakat NLMS algoritmasından daha hızlı yakınsadığı görülmektedir. Yapılan benzetim çalışmasında korelasyon matrisinin özdeğer yayılımı yaklaşık 410 olarak hesaplanmıştır. Burada korelasyon matrisinin özdeğer yayılımı daha büyük olduğu için algoritmaların yakınsama hızları arasındaki fark belirgin olmaktadır.



Şekil 4.35. ARMAX model için elde edilen ortalama karesel hata eğrileri.

4.5.3. ARARX model parametrelerinin tahmin edilmesi

Benzetim işleminde kullanılan örnek ikinci derece ARARX modelin transfer fonksiyonu gösterimi aşağıdaki gibidir.

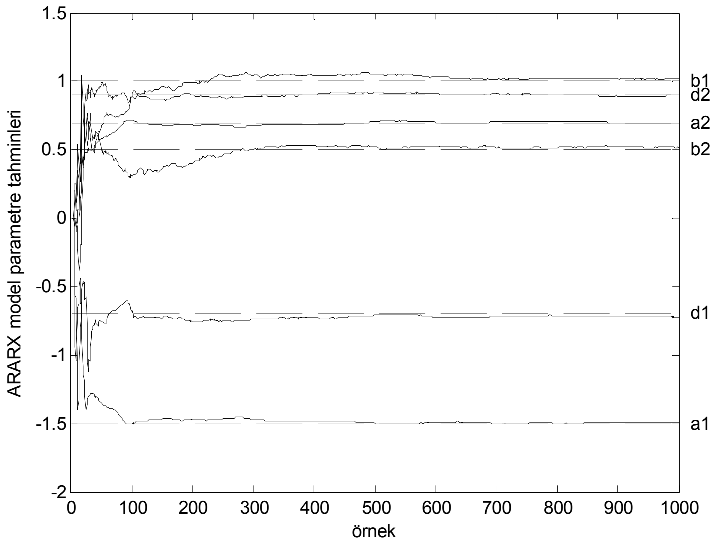
$$y(k) = \frac{z^{-1} + 0.5z^{-2}}{1 - 1.5z^{-1} + 0.7z^{-2}} u(k) + \frac{1}{(1 - 1.5z^{-1} + 0.7z^{-2})(1 - 0.7z^{-1} + 0.9z^{-2})} e(k) \qquad (4.85)$$

Burada sistemin giriş işareti $u(k)$ ve bozucu giriş işareti $e(k)$ birbirinden bağımsız olan sıfır ortalamalı varyansı 1 olan normal dağılıma sahip rasgele gürültü dizileridir. Benzetim çalışmasında kullanılan optimum parametre vektörü ve veri vektörü aşağıdaki gibidir.

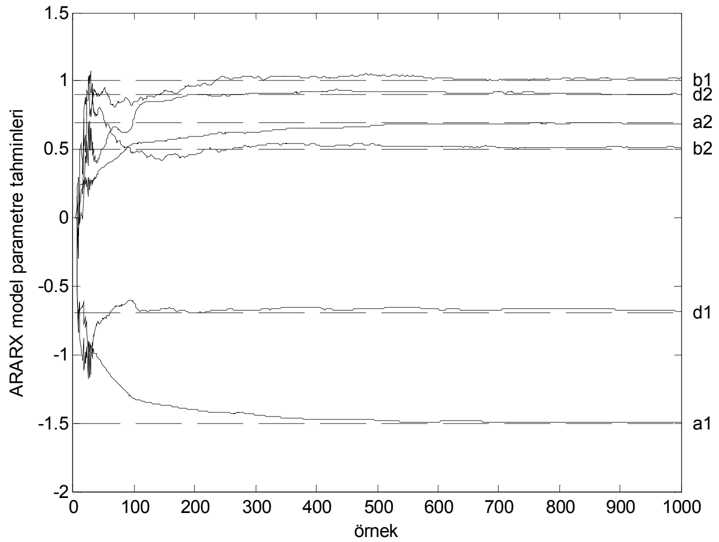
$$\theta_{opt}^{T} = \begin{bmatrix} a_1 & a_2 & b_1 & b_2 & d_1 & d_2 \end{bmatrix} = \begin{bmatrix} -1.5 & 0.7 & 1.0 & 0.5 & -0.7 & 0.9 \end{bmatrix} \tag{4.86}$$

$$\varphi^{T}(k) = \begin{bmatrix} -y(k-1) & -y(k-2) & u(k-1) & u(k-2) & -v(k-1) & -v(k-2) \end{bmatrix} \tag{4.87}$$

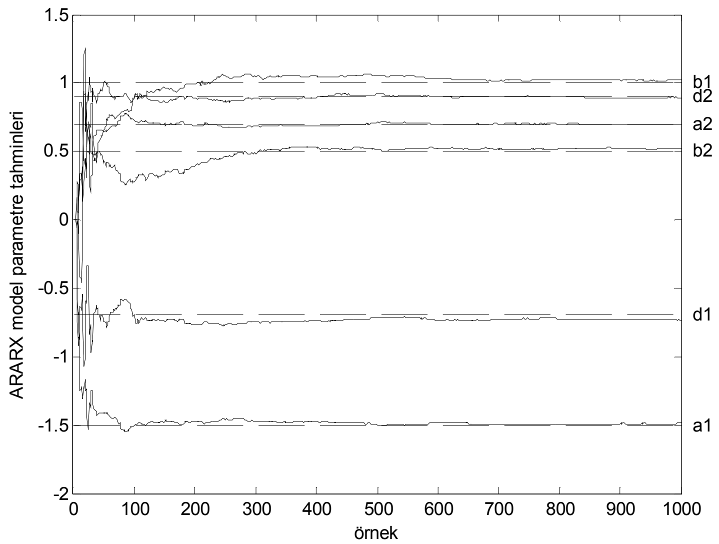
Benzetim çalışmasında tekrarlamalı GS, Jacobi ve RLS algoritmalarıyla ARARX model katsayıları tahmin edildiğinde elde edilen sonuçlar sırasıyla Şekil 4.36, Şekil 4.37 ve Şekil 4.38'de verilmiştir. Burada korelasyon matrisinin kendisinin ve tersinin başlangıç değeri birim matris alınmış ve unutma faktörü kullanılmamıştır. Adım parametresi tekrarlamalı Jacobi algoritmasında $\mu = 0.6$ olarak alınmıştır. Bütün algoritmalarda parametre tahminlerinin başlangıç değerleri sıfır vektör alınmıştır.



Şekil 4.36. Tekrarlamalı GS algoritması ile hesaplanan ARARX model parametre tahminleri.



Şekil 4.37. Tekrarlamalı Jacobi algoritması ile hesaplanan ARARX model parametre tahminleri.



Şekil 4.38. RLS algoritması ile hesaplanan ARARX model parametre tahminleri.

ARARX modelin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.9`da verilmiştir.

Çizelge 4.9. ARARX model parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	120	100
Jacobi	132	200
RLS	182	50

Önerilen tekrarlamalı GS ve tekrarlamalı Jacobi algoritmalarının yakınsama hızlarının yaygın olarak kullanılan RLS algoritmasına çok yakın sonuç verdiği görülmektedir. Yapılan benzetim çalışmasında korelasyon matrisinin özdeğer yayılımı yaklaşık 101 olarak hesaplanmıştır. Bu değer küçük değildir fakat çok büyük de olmadığı için algoritmaların yakınsama hızları arasındaki fark da çok fazla değildir.

4.5.4. ARARMAX model parametrelerinin tahmin edilmesi

Benzetim işleminde kullanılan örnek ikinci derece ARARMAX modelin transfer fonksiyonu gösterimi aşağıdaki gibidir.

$$y(k)=\frac{z^{-1}+0.5z^{-2}}{1-1.5z^{-1}+0.7z^{-2}}u(k)+\frac{1-z^{-1}+0.2z^{-2}}{(1-1.5z^{-1}+0.7z^{-2})(1-0.7z^{-1}+0.8z^{-2})}e(k) \tag{4.88}$$

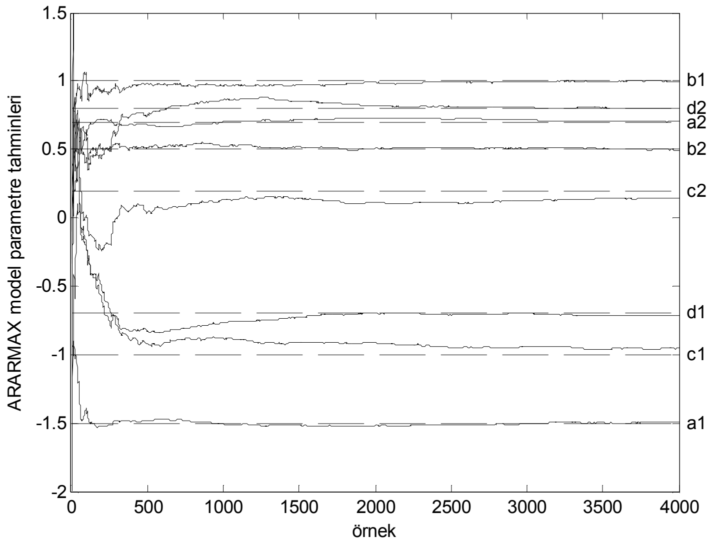
Burada sistemin giriş işareti $u(k)$ ve bozucu giriş işareti $e(k)$ birbirinden bağımsız olan sıfır ortalamalı varyansı 1 olan normal dağılıma sahip rasgele gürültü dizileridir. Benzetim çalışmasında kullanılan optimum parametre vektörü ve veri vektörü aşağıdaki gibidir.

$$\theta_{opt}^{T}=\begin{bmatrix}a_1 & a_2 & b_1 & b_2 & c_1 & c_2 & d_1 & d_2\end{bmatrix}=\begin{bmatrix}-1.5 & 0.7 & 1.0 & 0.5 & -1.0 & 0.2 & -0.7 & 0.8\end{bmatrix} \tag{4.89}$$

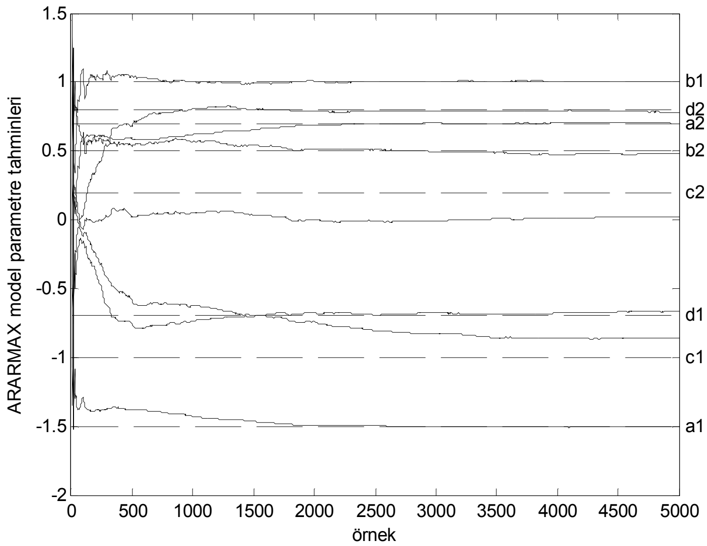
$$\varphi^{T}(k)=\begin{bmatrix}-y(k-1) & -y(k-2) & u(k-1) & u(k-2) & \varepsilon(k-1) & \varepsilon(k-2) & -v(k-1) & -v(k-2)\end{bmatrix} \tag{4.90}$$

Benzetim çalışmasında tekrarlamalı GS, Jacobi ve RLS algoritmalarıyla ARARMAX model katsayıları tahmin edildiğinde elde edilen sonuçlar sırasıyla Şekil 4.39, Şekil 4.40

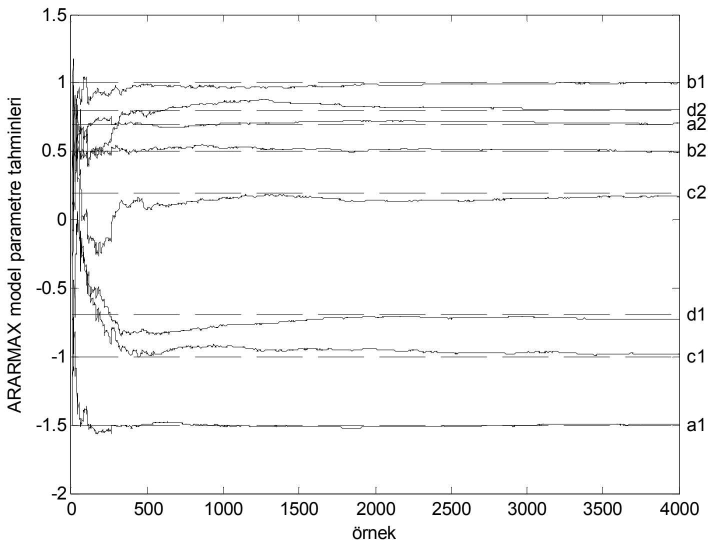
ve Şekil 4.41'de verilmiştir. Burada korelasyon matrisinin kendisinin ve tersinin başlangıç değeri birim matris alınmış ve unutma faktörü kullanılmamıştır. Adım parametresi tekrarlamalı Jacobi algoritmasında $\mu = 0.5$ olarak alınmıştır. Bütün algoritmalarda parametre tahminlerinin başlangıç değerleri sıfır vektör alınmıştır. Önerilen tekrarlamalı GS ve tekrarlamalı Jacobi algoritmalarının yakınsama hızlarının yaygın olarak kullanılan RLS algoritmasına çok yakın sonuç verdiği görülmektedir. Yapılan benzetim çalışmasında korelasyon matrisinin özdeğer yayılımı yaklaşık 138 olarak hesaplanmıştır. Bu değer küçük değildir fakat çok büyük de olmadığı için algoritmaların yakınsama hızları arasındaki fark da çok fazla değildir.



Şekil 4.39. Tekrarlamalı GS algoritması ile hesaplanan ARARMAX model parametre tahminleri.



Şekil 4.40. Tekrarlamalı Jacobi algoritması ile hesaplanan ARARMAX model parametre tahminleri.



Şekil 4.41. RLS algoritması ile hesaplanan ARARMAX model parametre tahminleri.

ARARMAX modelin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.10`da verilmiştir.

Çizelge 4.10. ARARMAX model parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	208	500
Jacobi	224	1000
RLS	288	300

4.5.5. Çıkış hatası model parametrelerinin tahmin edilmesi

Benzetim işleminde kullanılan örnek ikinci derece çıkış hatası modelin transfer fonksiyonu gösterimi aşağıdaki gibidir.

$$y(k) = \frac{z^{-1} + 0.5z^{-2}}{1 - 1.5z^{-1} + 0.7z^{-2}} u(k) + v(k) \tag{4.91}$$

Burada sistemin giriş işareti $u(k)$ ve bozucu giriş işareti $e(k)$ birbirinden bağımsız olan sıfır ortalamalı varyansı 1 olan normal dağılıma sahip rastgele gürültü dizileridir, $v(k)$ ise aşağıdaki gibi elde edilen renkli gürültü dizisidir.

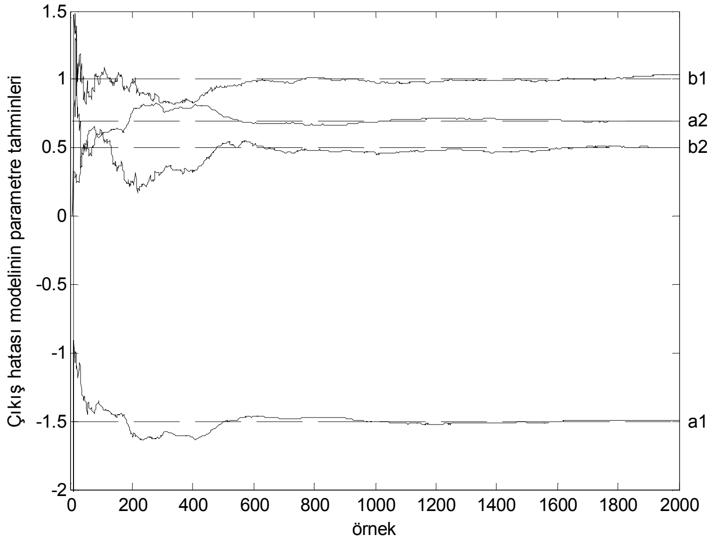
$$v(k) = \frac{1 + z^{-1} - 0.2z^{-2}}{1 - 0.7z^{-1} + 0.8z^{-2}} e(k) \tag{4.92}$$

Benzetim çalışmasında kullanılan optimum parametre vektörü ve veri vektörü aşağıdaki gibidir.

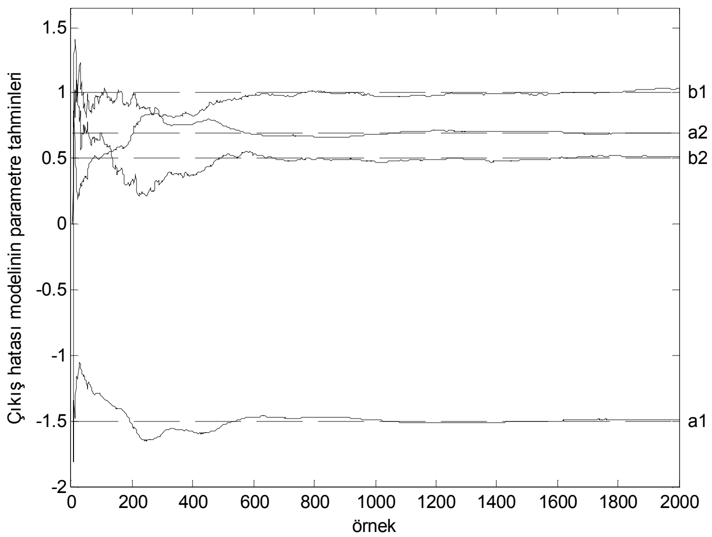
$$\theta_{opt}^T = \begin{bmatrix} a_1 & a_2 & b_1 & b_2 \end{bmatrix} = \begin{bmatrix} -1.5 & 0.7 & 1.0 & 0.5 \end{bmatrix} \tag{4.93}$$

$$z^T(k) = \begin{bmatrix} -\hat{x}(k-1) & -\hat{x}(k-2) & u(k-1) & u(k-2) \end{bmatrix} \tag{4.94}$$

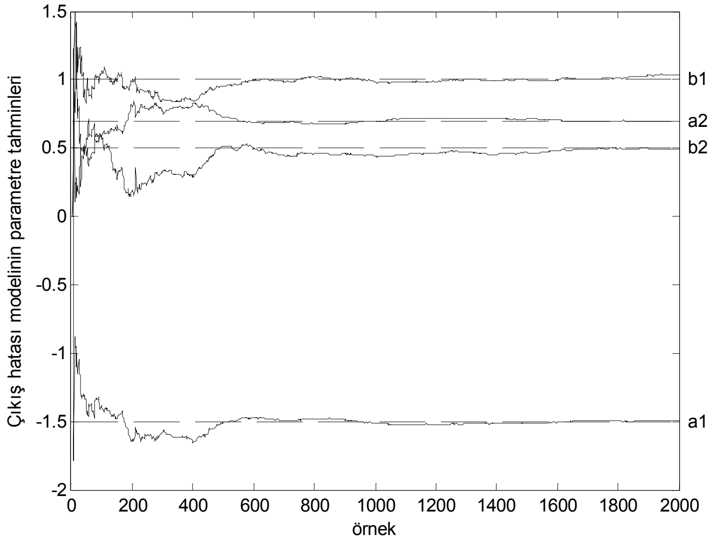
Benzetim çalışmasında öncelikle tekrarlamalı GS ve Jacobi algoritmalarıyla çıkış hatası model katsayıları tahmin edildiğinde elde edilen sonuçlar sırasıyla Şekil 4.42 ve Şekil 4.43'te verilmiştir. Daha sonra tekrarlamalı çıkış hatası ve NLMS algoritmalarıyla elde edilen sonuçlar elde edilen sonuçlar sırasıyla Şekil 4.44 ve Şekil 4.45'te verilmiştir. Burada korelasyon matrisinin kendisinin ve tersinin başlangıç değeri tekrarlamalı GS, Jacobi ve tekrarlamalı çıkış hatası algoritmalarında birim matris alınmıştır. Bu algoritmalarda unutma faktörü kullanılmamıştır. Adım parametresi tekrarlamalı Jacobi algoritmasında $\mu = 0.7$, NLMS algoritmasında $\mu = 0.03$ olarak alınmıştır. Bütün algoritmalarda parametre tahminlerinin başlangıç değerleri sıfır vektör alınmıştır. Yapılan benzetim çalışmasında korelasyon matrisinin özdeğer yayılımı yaklaşık 58 olarak hesaplanmıştır. Korelasyon matrisinin özdeğer yayılımı küçük olduğu için algoritmaların yakınsama hızları arasındaki farkın küçük olduğu görülmektedir.



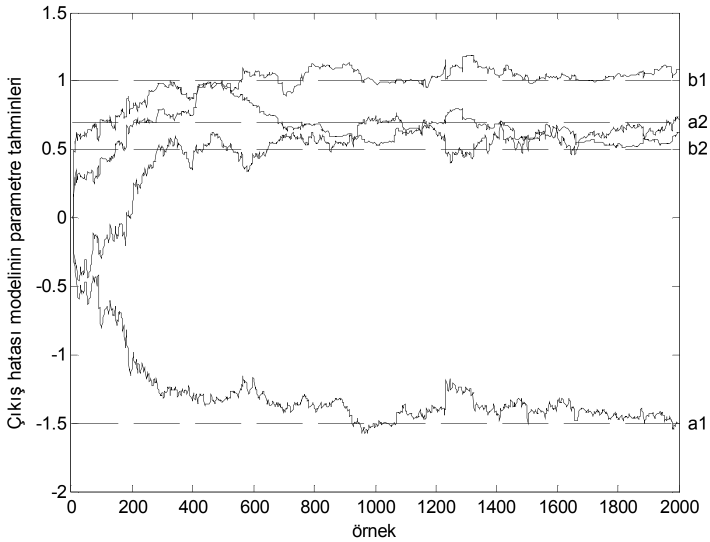
Şekil 4.42. Tekrarlamalı GS algoritması ile hesaplanan çıkış hatası modelinin parametre tahminleri.



Şekil 4.43. Tekrarlamalı Jacobi algoritması ile hesaplanan çıkış hatası modelinin parametre tahminleri.



Şekil 4.44. Tekrarlamalı çıkış hatası algoritması ile hesaplanan çıkış hatası modelinin parametre tahminleri.



Şekil 4.45. NLMS algoritması ile hesaplanan çıkış hatası modelinin parametre tahminleri.

Çıkış hatası modelinin parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.11`de verilmiştir.

Çizelge 4.11. Çıkış hatası modelinin parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	56	50
Jacobi	64	50
RLS	100	50
NLMS	14	300

4.5.6. Box-Jenkins model parametrelerinin tahmin edilmesi

Benzetim işleminde kullanılan örnek ikinci derece Box-Jenkins modelin transfer fonksiyonu gösterimi aşağıdaki gibidir.

$$y(k) = \frac{z^{-1} + 0.5z^{-2}}{1 - 1.5z^{-1} + 0.7z^{-2}} u(k) + v(k) \tag{4.95}$$

Burada sistemin giriş işareti $u(k)$ ve bozucu giriş işareti $e(k)$ birbirinden bağımsız olan sıfır ortalamalı varyansı 1 olan normal dağılıma sahip rastgele gürültü dizileridir, $v(k)$ ise aşağıdaki gibi elde edilen renkli gürültü dizisidir.

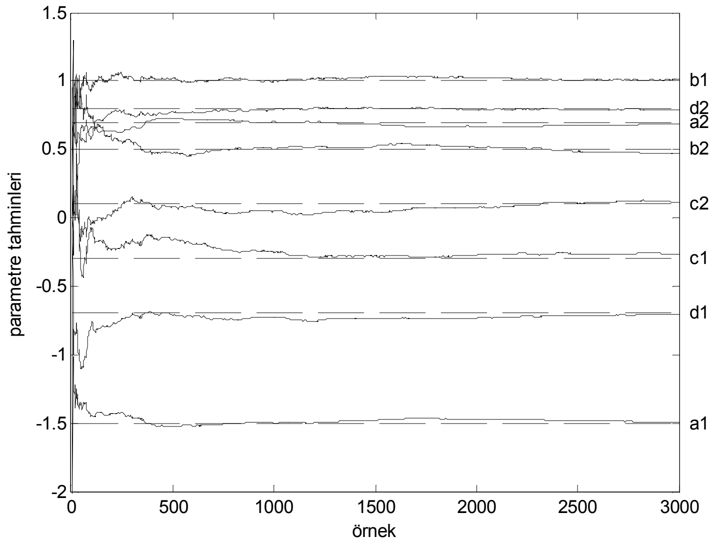
$$v(k) = \frac{1 - 0.3z^{-1} + 0.1z^{-2}}{1 - 0.7z^{-1} + 0.8z^{-2}} e(k) \tag{4.96}$$

Benzetim çalışmasında kullanılan optimum parametre vektörü ve veri vektörü aşağıdaki gibidir.

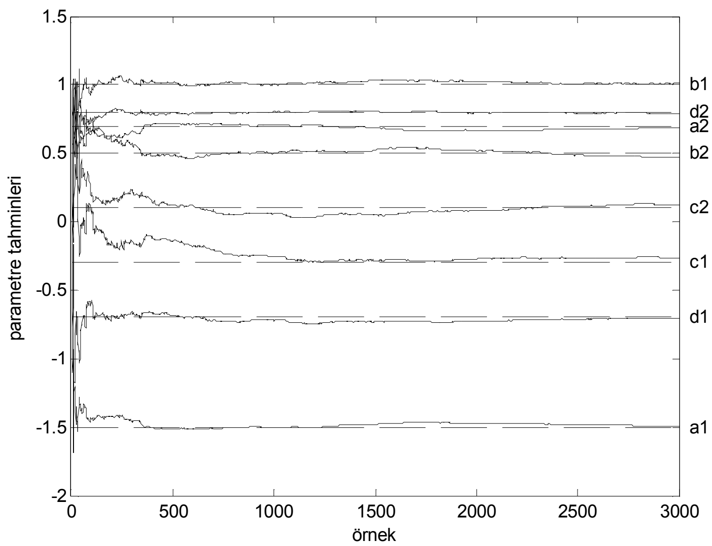
$$\begin{aligned} \theta_{opt}^T &= \begin{bmatrix} a_1 & a_2 & b_1 & b_2 & c_1 & c_2 & d_1 & d_2 \end{bmatrix} \\ &= \begin{bmatrix} -1.5 & 0.7 & 1.0 & 0.5 & -0.3 & 0.1 & -0.7 & 0.8 \end{bmatrix} \end{aligned} \tag{4.97}$$

$$\varphi^T(k) = \begin{bmatrix} -x(k-1) & -x(k-2) & u(k-1) & u(k-2) & \varepsilon(k-1) & \varepsilon(k-2) & -v(k-1) & -v(k-2) \end{bmatrix} \tag{4.98}$$

Benzetim çalışmasında RLS ve tekrarlamalı GS algoritmalarıyla Box-Jenkins model katsayıları tahmin edildiğinde elde edilen sonuçlar sırasıyla Şekil 4.46 ve Şekil 4.47'de verilmiştir. Burada korelasyon matrisinin kendisinin ve tersinin başlangıç değeri tekrarlamalı GS ve RLS algoritmalarında birim matris alınmıştır. Bütün algoritmalarda parametre tahminlerinin başlangıç değerleri sıfır vektör alınmıştır. Yapılan benzetim çalışmasında korelasyon matrisinin özdeğer yayılımı yaklaşık 86 olarak hesaplanmıştır. Korelasyon matrisinin özdeğer yayılımı küçük olduğu için algoritmaların yakınsama hızları arasındaki farkın küçük olduğu görülmektedir.



Şekil 4.46. Tekrarlamalı GS algoritması ile hesaplanan Box-Jenkins model parametre tahminleri.



Şekil 4.47. RLS algoritması ile hesaplanan Box-Jenkins model parametre tahminleri.

Box-Jenkins model parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.12`de verilmiştir.

Çizelge 4.12. Box-Jenkins model parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	208	200
RLS	288	100

4.6. Tekrarlamalı Gauss-Seidel Algoritması ile Volterra Model Tahmini

Bu kısımda yapılan benzetim çalışmalarında, Mathews (1991) tarafından kullanılan ve katsayıları aşağıda verilen ikinci derece Volterra model kullanılmıştır.

$$\underline{\text{filtrenin doğrusal bileşenlerinin katsayıları :}}$$
$$\left[h_1(0), h_1(1), h_1(2), h_1(3)\right] = \left[-0.78, -1.48, -1.39,\ 0.04\right]$$
$$\underline{\text{filtrenin doğrusal olmayan (ikinci derece) bileşenlerinin katsayıları :}}$$
$$\left[h_2(0,0), h_2(0,1), h_2(0,2), h_2(0,3), h_2(1,1), h_2(1,2), h_2(1,3), h_2(2,2), h_2(2,3), h_2(3,3)\right]$$
$$= \left[0.54,\ 3.72,\ 1.86,\ -0.76,\ -1.62,\ 0.76,\ -0.12,\ 1.41,\ -1.52,\ -0.13\right]$$

Sisteme giriş işareti olarak $x(k) = 0.25v(k) + 1.0v(k-1) + 0.25v(k)$ filtresinden geçirilerek elde edilen renkli gürültü kullanılmıştır, burada $v(k)$ sıfır ortalamalı, varyansı 1 olan normal dağılıma sahip rastgele gürültü dizisidir. Yapılan sistem tanıma işleminde çıkış işareti ölçümlerine $SNR \cong 20\,\text{dB}$, yani $NSR \cong \%10$ olacak şekilde, $x(k)$ giriş işaretiyle ilişkisiz normal dağılıma sahip sıfır ortalamalı rastgele gürültü eklenmiştir. Yapılan benzetim çalışmalarında korelasyon matrisinin özdeğer yayılımının yaklaşık 60 civarında olduğu görülüştür. Öncelikle LMS, NLMS, RLS, tekrarlamalı GS ve tekrarlamalı Jacobi algoritmaları birer defa çalıştırılmış ve elde edilen parametre tahminleri sırasıyla Şekil 4.48, Şekil 4.49, Şekil 4.50, Şekil 4.51 ve Şekil 4.52'de verilmiştir. Yapılan benzetim çalışmalarında kullanılan algoritmaların başlangıç koşulları ve adım parametreleri aşağıdaki gibi verilmiştir. Ayrıca bütün algoritmalarda parametre tahminlerinin başlangıç değeri sıfır vektör olarak alınmıştır.

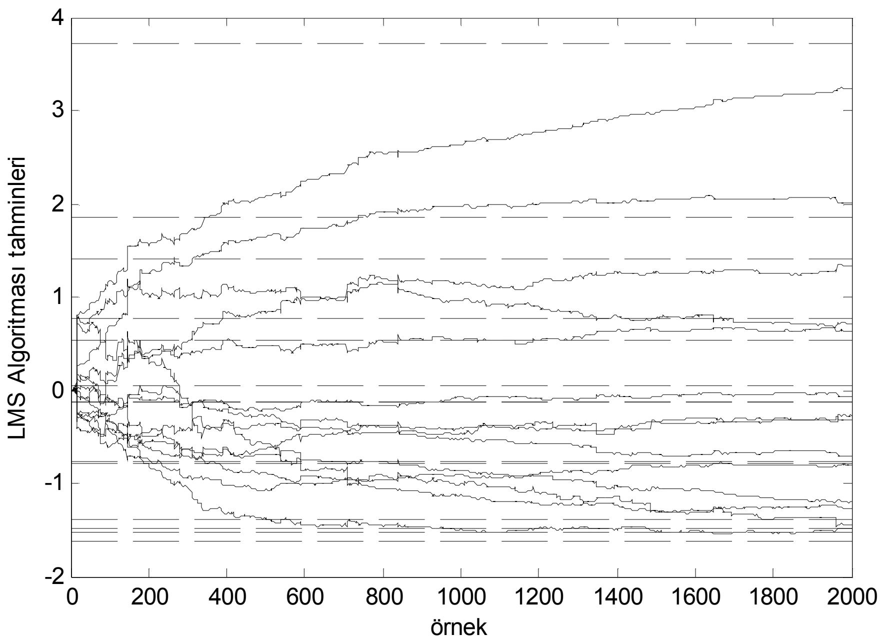
LMS : $\mu = 0.002$

NLMS: $\mu = 0.1$

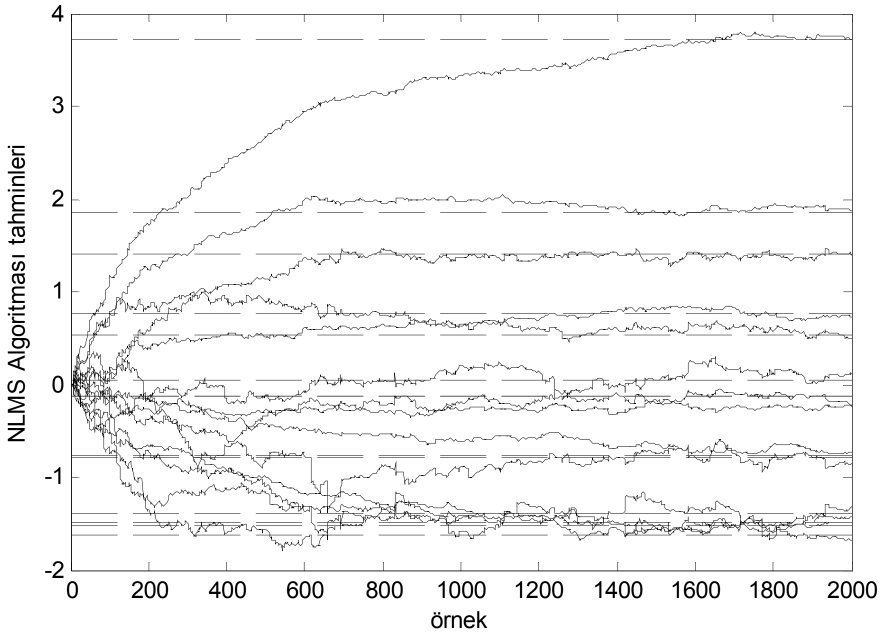
RLS: $R^{-1}(0) = 10\mathbf{I}$

GS: $R(0) = 0.1\mathbf{I}$

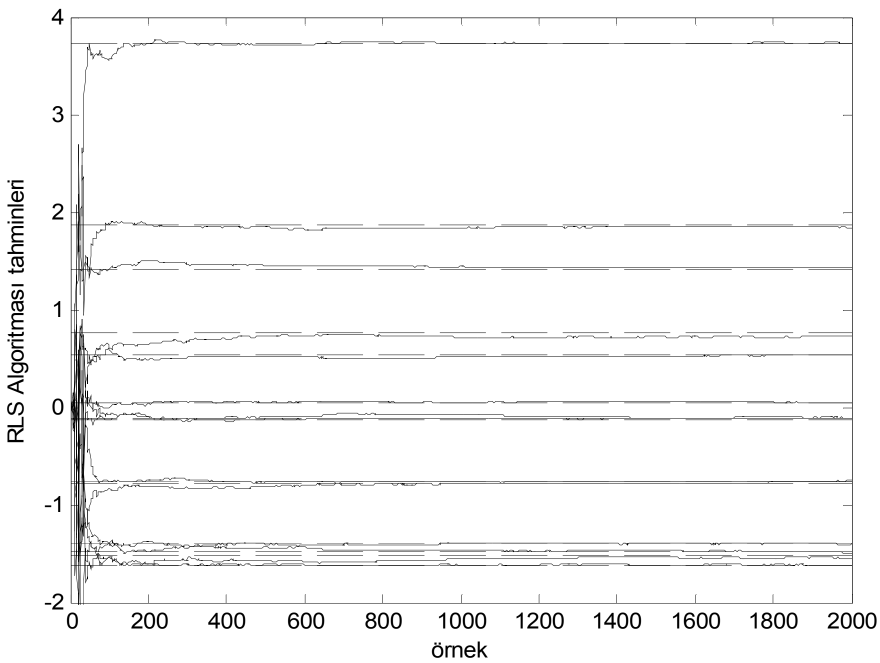
Jacobi: $R(0) = 0.1\mathbf{I}$ ve $\mu = 0.2$



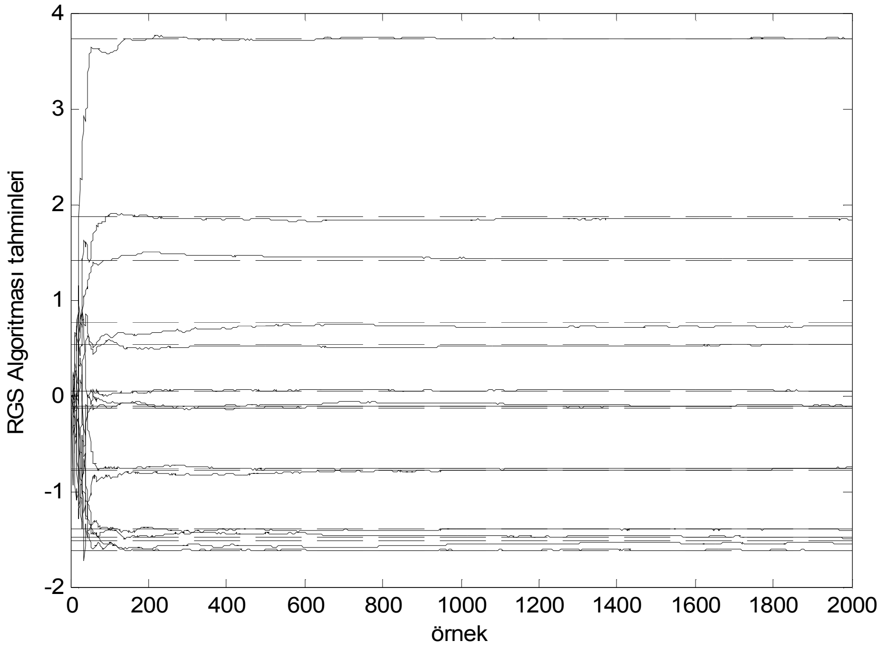
Şekil 4.48. LMS algoritması ile elde edilen Volterra model parametre tahminleri.



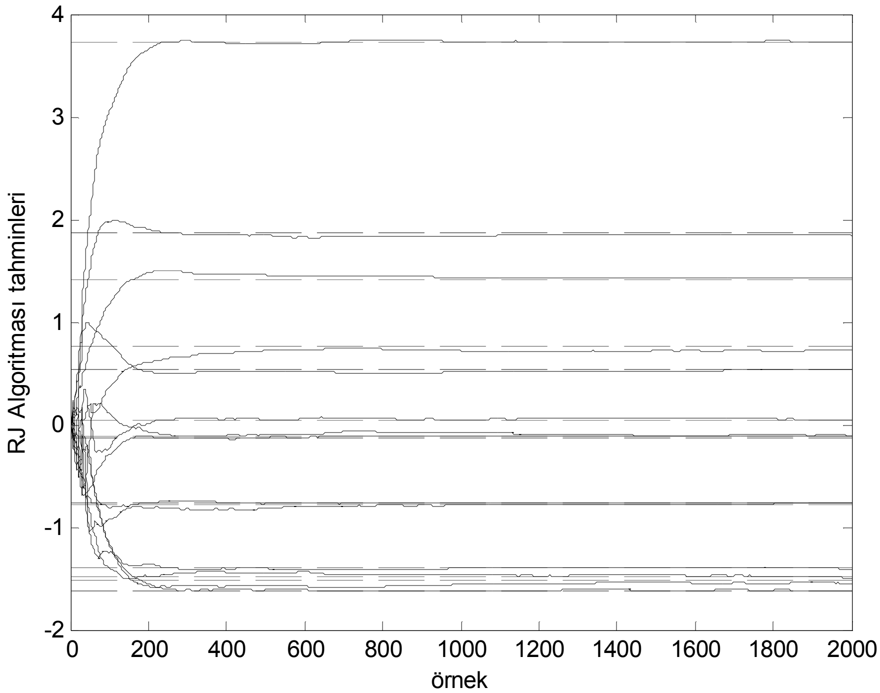
Şekil 4.49. NLMS algoritması ile elde edilen Volterra model parametre tahminleri.



Şekil 4.50. RLS algoritması ile elde edilen Volterra model parametre tahminleri.



Şekil 4.51. Tekrarlamalı GS algoritması ile elde edilen Volterra model parametre tahminleri.



Şekil 4.52. Tekrarlamalı Jacobi algoritması ile elde edilen Volterra model parametre tahminleri.

Volterra model parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.13`de verilmiştir.

Çizelge 4.13. Volterra model parametrelerinin tahmin edilmesi işleminde kullanılan algoritmaların karşılaştırılması.

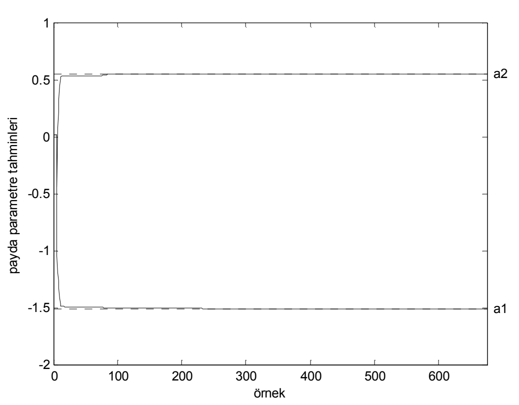
Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
LMS	43	1400
NLMS	44	600
RLS	750	50
GS	616	50
Jacobi	644	200

4.7. Tekrarlamalı Gauss-Seidel Algoritması ile Dolaylı Uyarlamalı Kontrol

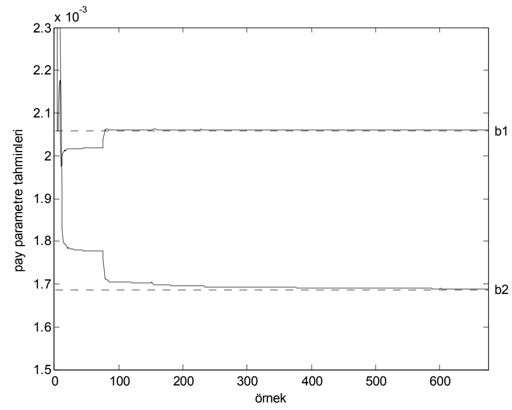
Bu kısımda yapılan benzetim çalışmalarında, bölüm 4.3'teki doğru akım motoru ve yükten oluşan sistem kullanılmıştır. Yapılan benzetim çalışmasında doğru akım motoru ve yükten oluşan sistemin $T = 0.05$ s. alınarak Z.O.H. dönüşümü ile elde edilen ayrık zaman eşdeğer modeli kullanılmıştır. Kapalı çevrim sisteme referans giriş olarak periyodu 150 örnek olan ±1 genlikli kare dalga uygulanmıştır. Bu kısımda öncelikle kendinden ayarlamalı PID denetleyicilerle elde edilen sonuçlar verilecektir, daha sonra ise model tabanlı uyarlamalı denetleyici ile elde edilen sonuçlar verilecektir.

4.7.1. Tekrarlamalı Gauss-Seidel algoritması ile kendinden ayarlamalı PID kontrol

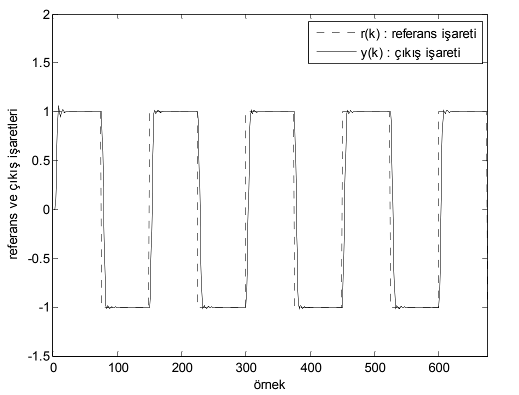
Yapılan ilk benzetim çalışmasında kendinden ayarlamalı Ziegler-Nichols PID denetleyicinin konumsal biçimde gerçeklenen şekli kullanılmıştır. Aşağıda sırasıyla Şekil 4.53'te RLS ve Şekil 4.54'te tekrarlamalı GS algoritmasıyla elde edilen benzetim sonuçları görülmektedir. Burada korelasyon matrisinin başlangıç koşulu tekrarlamalı GS algoritmasında $R(0) = 0.001I$, tersinin başlangıç koşulu RLS algoritmasında $P(0) = 1000I$ alınmıştır. Parametre tahminlerinin başlangıç değerleri 0.01 alınmıştır. Bu çalışmada korelasyon matrisinin özdeğer yayılımının çok büyük (yaklaşık 2.10^5 mertebesinde) olmasından dolayı NLMS algoritması ile kullanışlı parametre tahminleri elde edilememiştir.



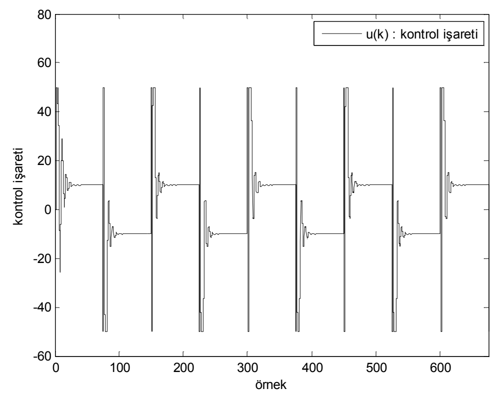
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.

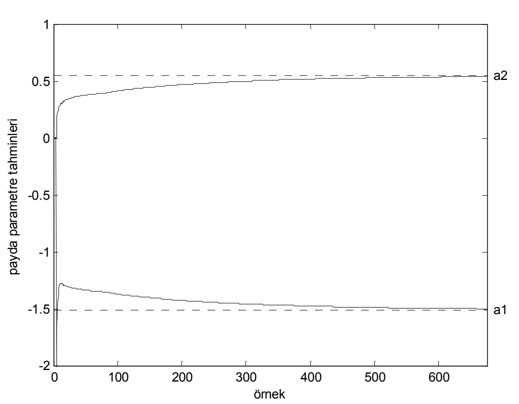


c) Sistemin referans ve çıkış sinyalleri.

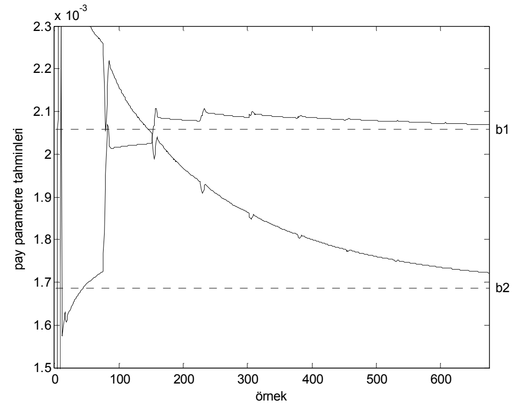


d) Sistemin hesaplanan kontrol girişi.

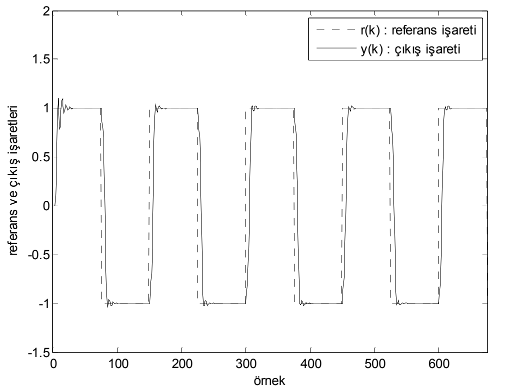
Şekil 4.53. RLS algoritmasıyla elde edilen uyarlamalı Ziegler-Nichols konumsal biçim PID kontrol benzetim sonuçları.



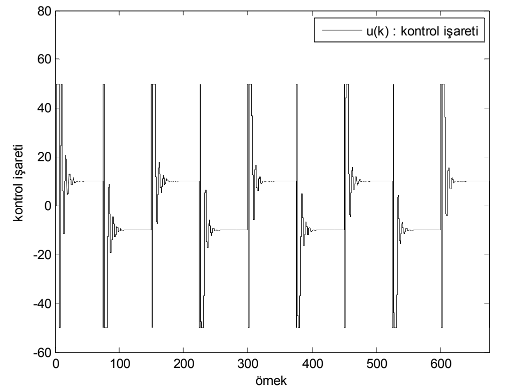
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.



c) Sistemin referans ve çıkış sinyalleri.



d) Sistemin hesaplanan kontrol girişi.

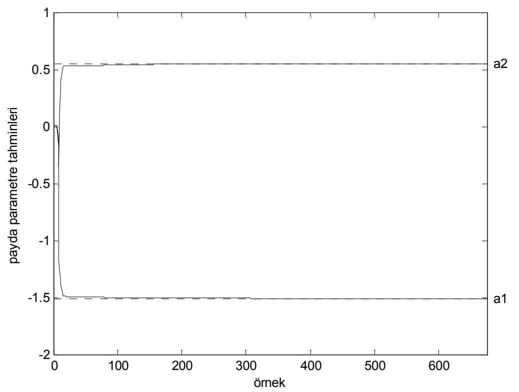
Şekil 4.54. Tekrarlamalı GS algoritmasıyla elde edilen uyarlamalı Ziegler-Nichols konumsal biçim PID kontrol benzetim sonuçları.

Konumsal biçimde gerçeklenen PID kontrol yönteminde model parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.14`de verilmiştir.

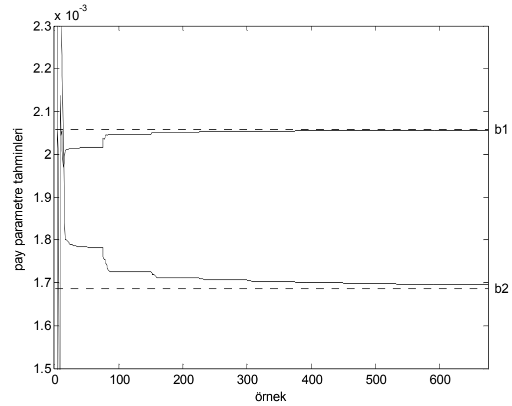
Çizelge 4.14. Konumsal biçimde gerçeklenen PID kontrol yönteminde model parametrelerini tahmin etmek için kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	56	400
RLS	100	10

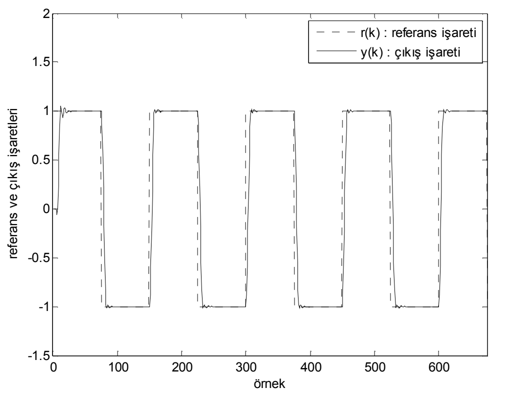
Sonra kendinden ayarlamalı Ziegler-Nichols PID denetleyicinin hızsal biçimde gerçeklenen şekli kullanılmıştır. Aşağıda sırasıyla Şekil 4.55’te RLS ve Şekil 4.56’da tekrarlamalı GS algoritmasıyla elde edilen benzetim sonuçları görülmektedir. Benzetim çalışmasında aynı başlangıç koşulları kullanılmıştır.



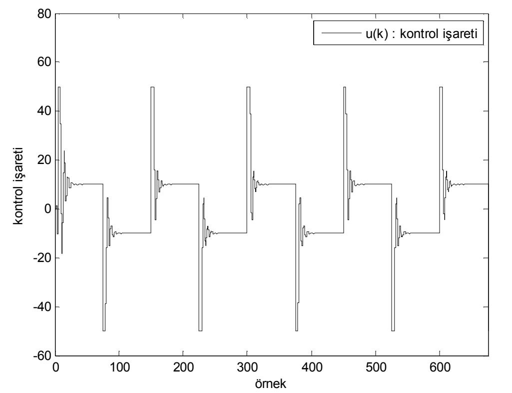
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.

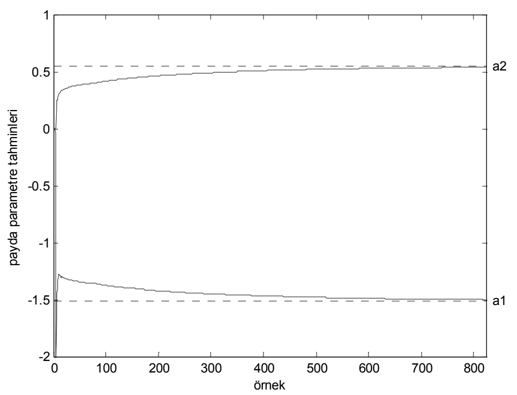


c) Sistemin referans ve çıkış sinyalleri.

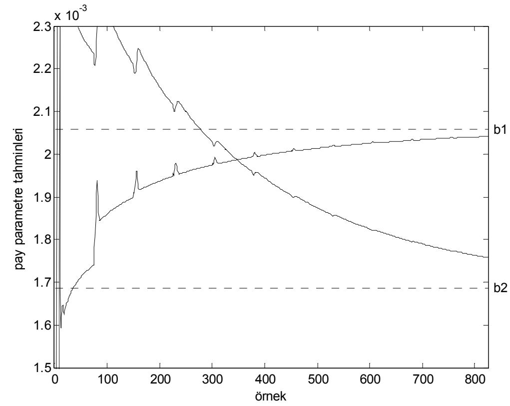


d) Sistemin hesaplanan kontrol girişi.

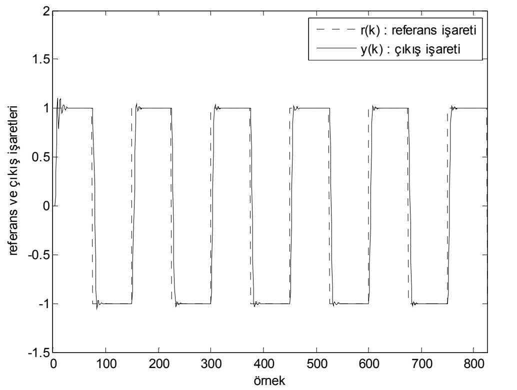
Şekil 4.55. RLS algoritmasıyla elde edilen uyarlamalı Ziegler-Nichols hızsal biçim PID kontrol benzetim sonuçları.



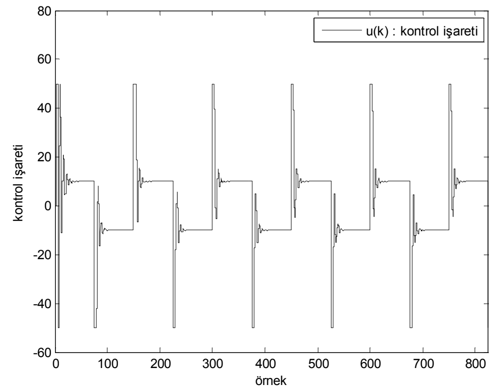
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.



c) Sistemin referans ve çıkış sinyalleri.



d) Sistemin hesaplanan kontrol girişi.

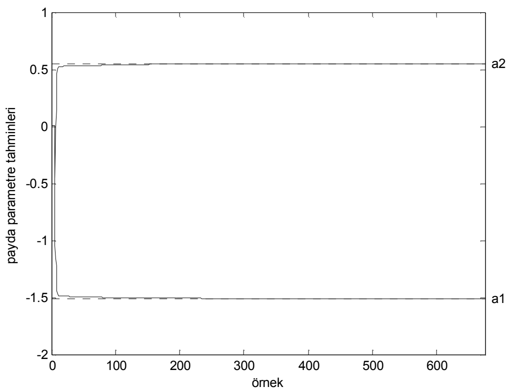
Şekil 4.56. Tekrarlamalı GS algoritmasıyla elde edilen uyarlamalı Ziegler-Nichols hızsal biçim PID kontrol benzetim sonuçları.

Hızsal biçimde gerçeklenen PID kontrol yönteminde model parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.15`te verilmiştir.

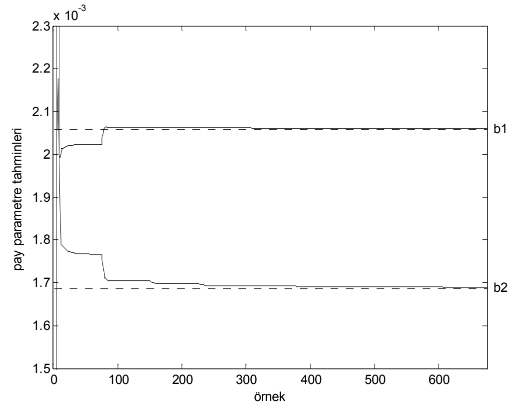
Çizelge 4.15. Hızsal biçimde gerçeklenen PID kontrol yönteminde model parametrelerini tahmin etmek için kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	56	500
RLS	100	20

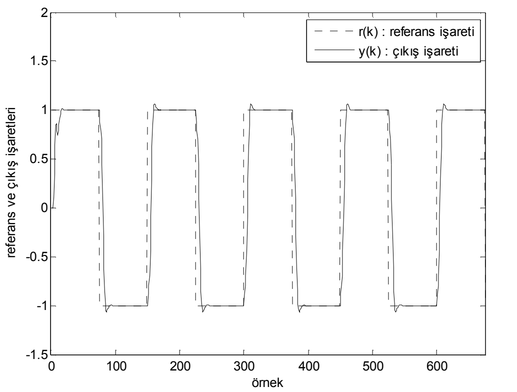
Daha sonraki benzetim çalışmasında polinomsal kutup yerleştirme yöntemiyle tasarlanan kendinden ayarlamalı PID denetleyicinin konumsal biçimde gerçeklenen şekli kullanılmıştır. Aşağıda sırasıyla Şekil 4.57'de RLS, Şekil 4.58'de tekrarlamalı GS algoritması ve Şekil 4.59'da tekrarlamalı SOR algoritmalarıyla elde edilen benzetim sonuçları görülmektedir.



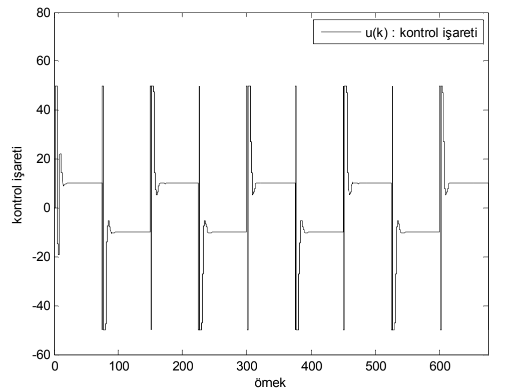
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.

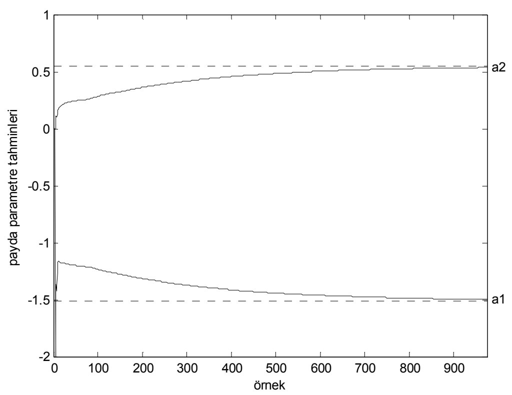


c) Sistemin referans ve çıkış sinyalleri.

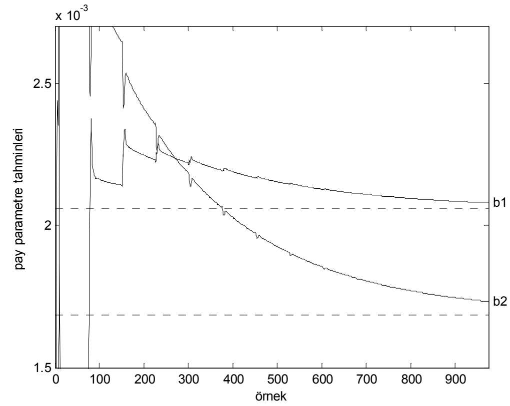


d) Sistemin hesaplanan kontrol girişi.

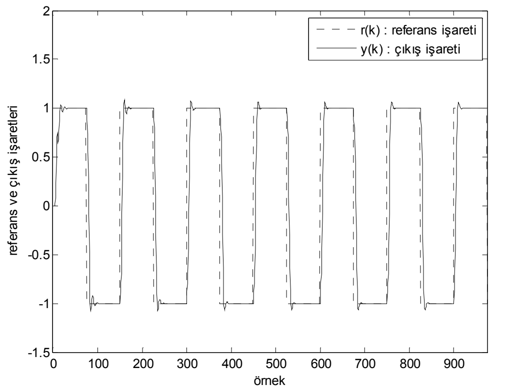
Şekil 4.57. RLS algoritmasıyla elde edilen kutup yerleştirmeli konumsal biçim uyarlamalı PID kontrol benzetim sonuçları.



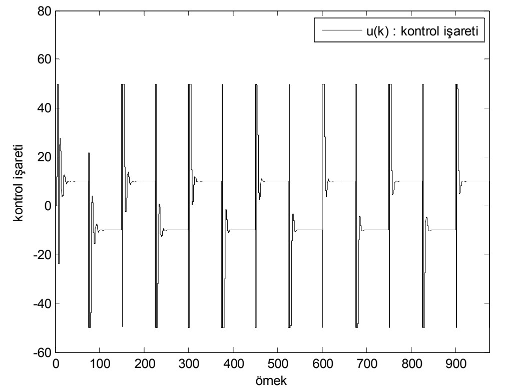
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.

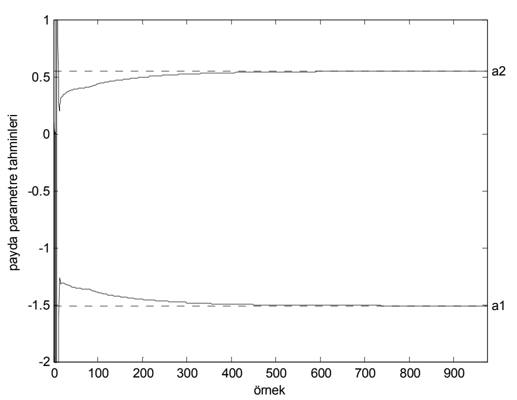


c) Sistemin referans ve çıkış sinyalleri.

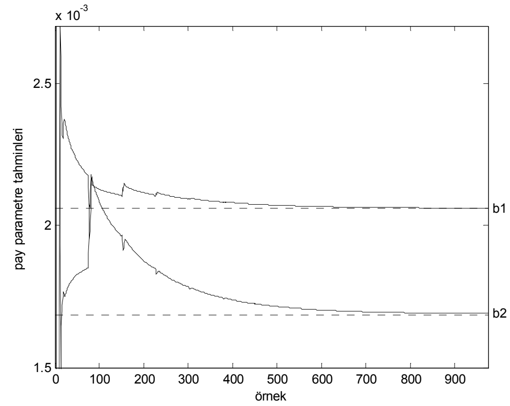


d) Sistemin hesaplanan kontrol girişi.

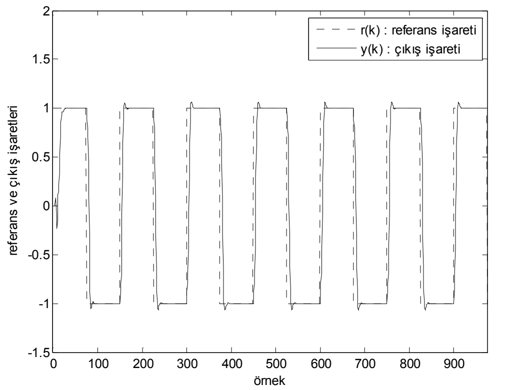
Şekil 4.58. Tekrarlamalı GS algoritmasıyla elde edilen kutup yerleştirmeli konumsal biçim uyarlamalı PID kontrol benzetim sonuçları.



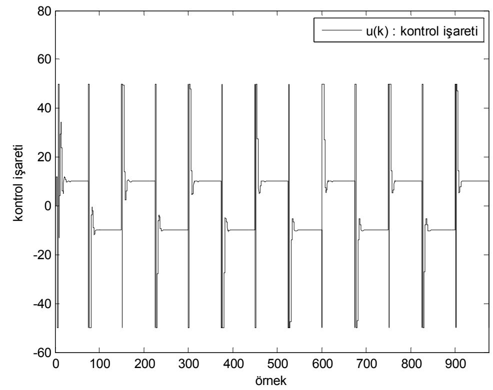
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.



c) Sistemin referans ve çıkış sinyalleri.



d) Sistemin hesaplanan kontrol girişi.

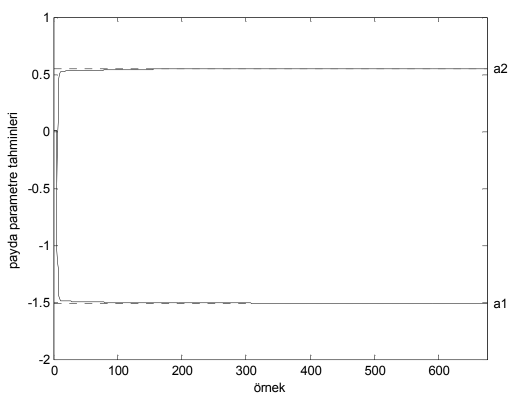
Şekil 4.59. Tekrarlamalı SOR algoritmasıyla elde edilen kutup yerleştirmeli konumsal biçim uyarlamalı PID kontrol benzetim sonuçları.

Konumsal biçimde gerçeklenen kutup yerleştirmeli PID kontrol yönteminde model parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.16`da verilmiştir.

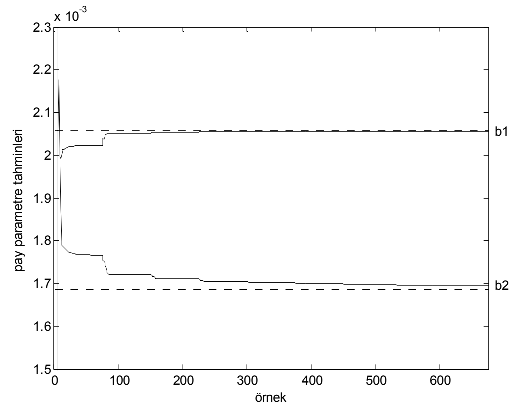
Çizelge 4.16. Konumsal biçimde gerçeklenen PID kontrol yönteminde model parametrelerini tahmin etmek için kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	56	600
SOR	64	300
RLS	100	10

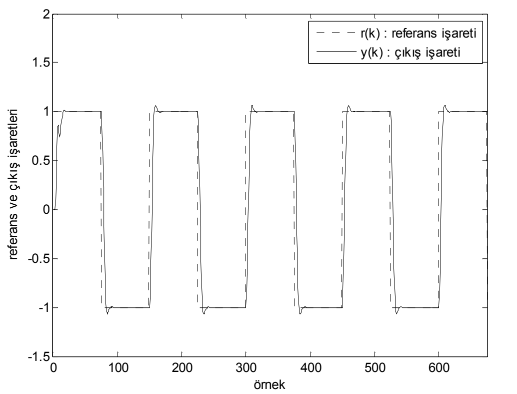
PID denetleyicilerle yapılan son benzetim çalışmasında polinomsal kutup yerleştirme yöntemiyle tasarlanan kendinden ayarlamalı PID denetleyicinin hızsal biçimde gerçeklenen şekli kullanılmıştır. Aşağıda sırasıyla Şekil 4.60'da RLS, Şekil 4.61'de tekrarlamalı GS algoritması ve Şekil 4.62'de tekrarlamalı SOR algoritmalarıyla elde edilen benzetim sonuçları görülmektedir.



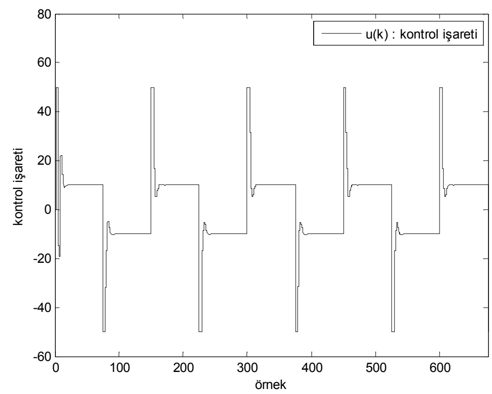
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.

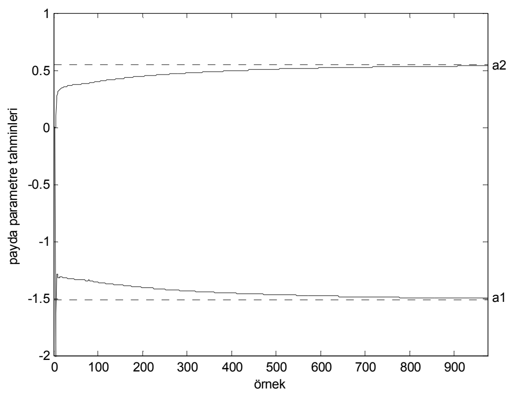


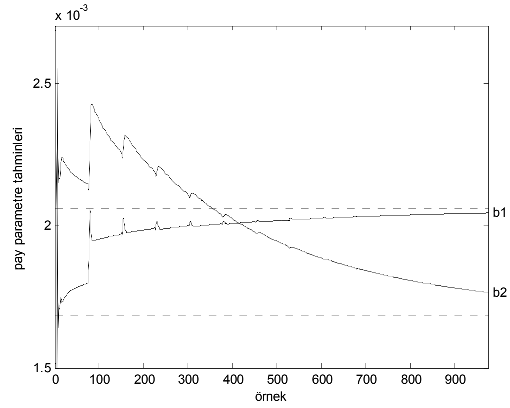
c) Sistemin referans ve çıkış sinyalleri.



d) Sistemin hesaplanan kontrol girişi.

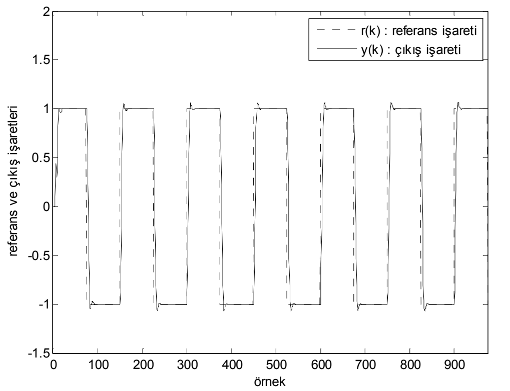
Şekil 4.60. RLS algoritmasıyla elde edilen kutup yerleştirmeli hızsal biçim uyarlamalı PID kontrol benzetim sonuçları.

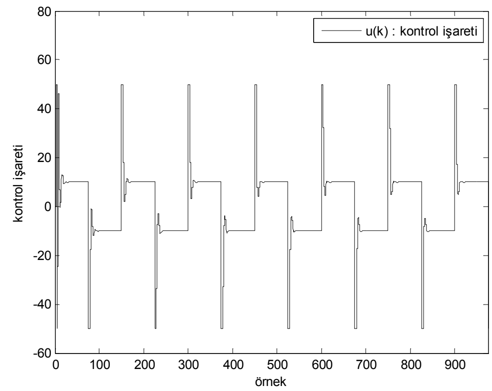




a) Sistemin payda katsayı tahminleri.

b) Sistemin pay katsayı tahminleri.

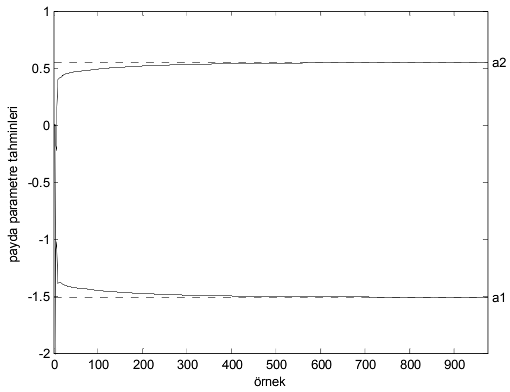




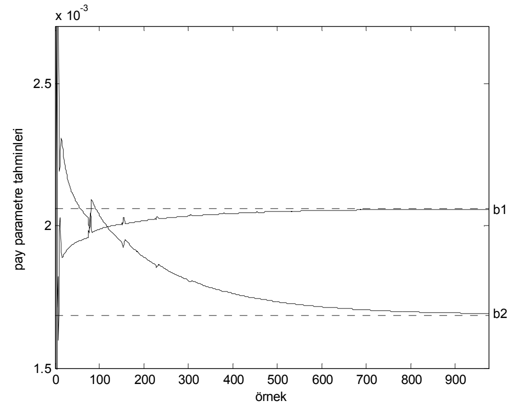
c) Sistemin referans ve çıkış sinyalleri.

d) Sistemin hesaplanan kontrol girişi.

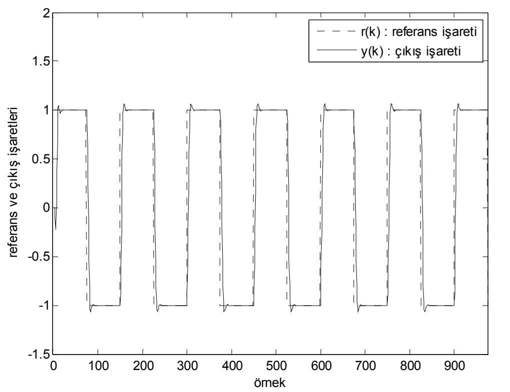
Şekil 4.61. Tekrarlamalı GS algoritmasıyla elde edilen kutup yerleştirmeli hızsal biçim uyarlamalı PID kontrol benzetim sonuçları.



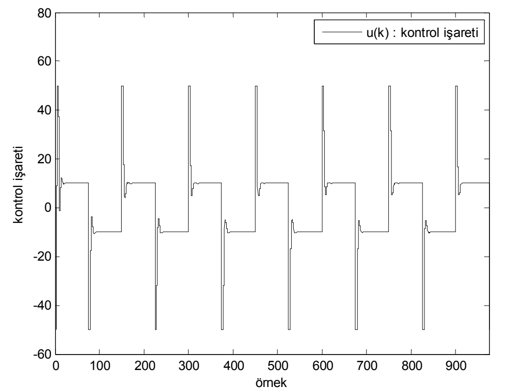
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.



c) Sistemin referans ve çıkış sinyalleri.



d) Sistemin hesaplanan kontrol girişi.

Şekil 4.62. Tekrarlamalı SOR algoritmasıyla elde edilen kutup yerleştirmeli hızsal biçim uyarlamalı PID kontrol benzetim sonuçları.

Hızsal biçimde gerçeklenen kutup yerleştirmeli PID kontrol yönteminde model parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.17`de verilmiştir.

Çizelge 4.17. Hızsal biçimde gerçeklenen PID kontrol yönteminde model parametrelerini tahmin etmek için kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	56	600
SOR	64	300
RLS	100	10

4.7.2. Tekrarlamalı Gauss-Seidel algoritması ile dolaylı model tabanlı uyarlamalı kontrol

Bu kısımda yapılan benzetim çalışmalarında yine bölüm 4.3'teki doğru akım motoru ve yükten oluşan sistem kullanılmıştır. Yapılan benzetim çalışmasında doğru akım motoru ve yükten oluşan sistemin $T = 0.05$ s. alınarak Z.O.H. dönüşümü ile elde edilen ayrık zaman eşdeğer modeli kullanılmıştır. Kapalı çevrim sisteme referans giriş olarak periyodu 150 örnek olan ±1 genlikli kare dalga uygulanmıştır. Burada tahmin edilen sistem modeli ve referans model sırasıyla aşağıdaki gibidir.

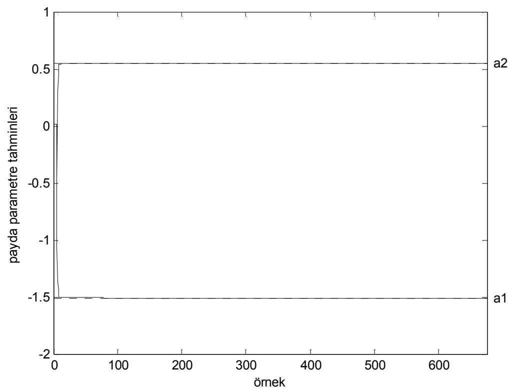
$$\frac{\hat{B}(z^{-1})}{\hat{A}(z^{-1})} = \frac{\hat{b}_0 + \hat{b}_1 z^{-1}}{1 + \hat{a}_1 z^{-1} + \hat{a}_2 z^{-2}} \cdot z^{-1} \quad , \quad \frac{B_m(z^{-1})}{A_m(z^{-1})} = \frac{b_{m0} + b_{m1} z^{-1}}{1 + a_{m1} z^{-1} + a_{m2} z^{-2}} \cdot z^{-1} \tag{4.99}$$

Denetleyici parametreleri aşağıdaki Diophantine denklemi çözülerek hesaplanmıştır.

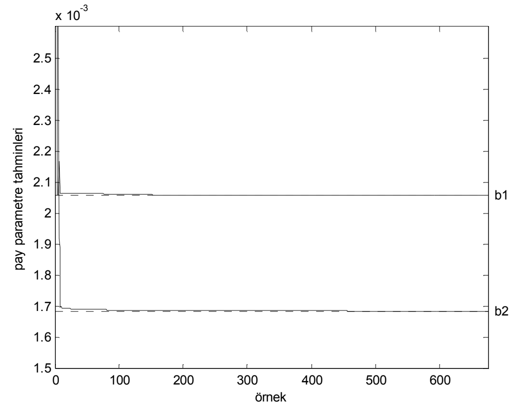
$$p_0 = \hat{b}_0 , \; p_1 = \hat{b}_0 a_{m1} + \hat{b}_1 , \; p_2 = \hat{b}_0 a_{m2} + \hat{b}_1 a_{m1} , \; p_3 = \hat{b}_1 a_{m2} \tag{4.100}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \hat{a}_1 & 1 & \hat{b}_0 & 0 \\ \hat{a}_2 & \hat{a}_1 & \hat{b}_1 & \hat{b}_0 \\ 0 & \hat{a}_2 & 0 & \hat{b}_1 \end{bmatrix} \cdot \begin{bmatrix} r_0 \\ r_1 \\ s_0 \\ s_1 \end{bmatrix} = \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} \Rightarrow \begin{bmatrix} r_0 \\ r_1 \\ s_0 \\ s_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \hat{a}_1 & 1 & \hat{b}_0 & 0 \\ \hat{a}_2 & \hat{a}_1 & \hat{b}_1 & \hat{b}_0 \\ 0 & \hat{a}_2 & 0 & \hat{b}_1 \end{bmatrix}^{-1} \cdot \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} \tag{4.101}$$

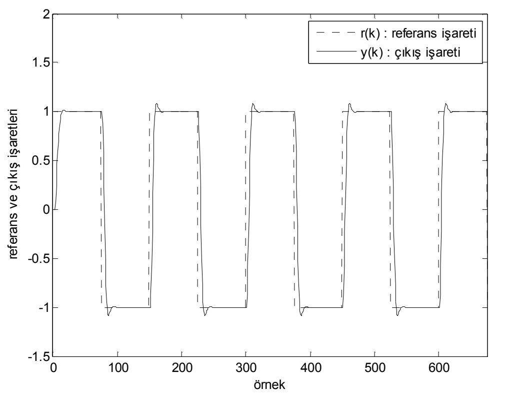
Aşağıda sırasıyla Şekil 4.63'te RLS ve Şekil 4.64'te tekrarlamalı GS algoritmasıyla elde edilen benzetim sonuçları görülmektedir. Burada korelasyon matrisinin başlangıç koşulu tekrarlamalı GS algoritmasında $R(0) = 0.0001 I$, tersinin başlangıç koşulu RLS algoritmasında $P(0) = 10000 I$ alınmıştır. Parametre tahminlerinin başlangıç değerleri 0.01 alınmıştır. Bu çalışmada korelasyon matrisinin özdeğer yayılımının çok büyük (yaklaşık 3.10^5) olmasından dolayı NLMS algoritması ile kullanışlı parametre tahminleri elde edilememiştir.



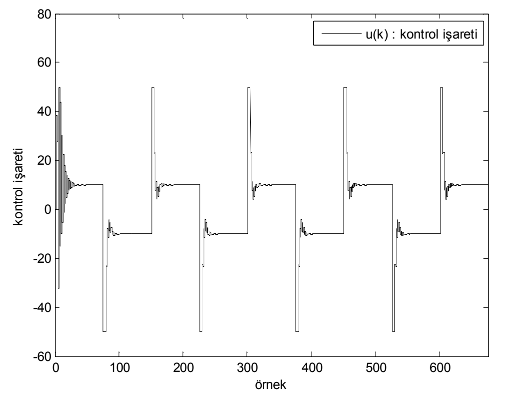
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.

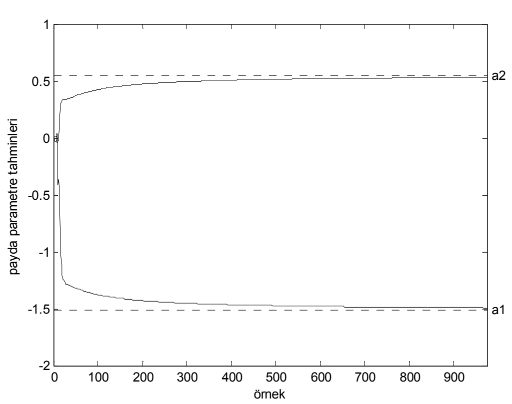


c) Sistemin referans ve çıkış sinyalleri.

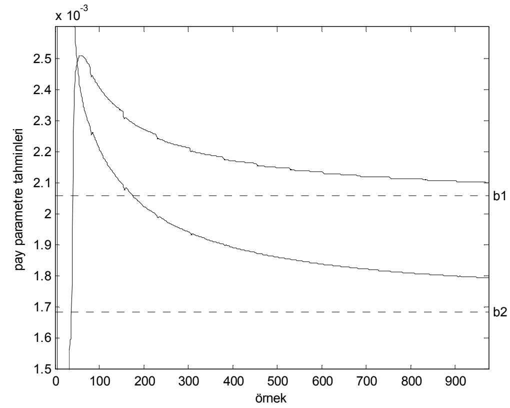


d) Sistemin hesaplanan kontrol girişi.

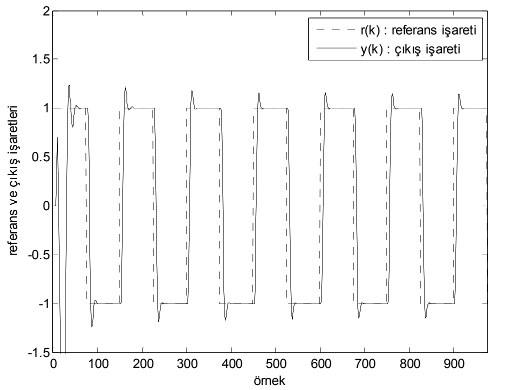
Şekil 4.63. RLS algoritmasıyla elde edilen kutup yerleştirmeli, dolaylı model tabanlı uyarlamalı kontrol benzetim sonuçları.



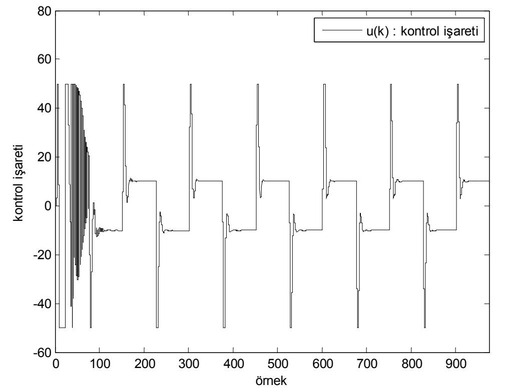
a) Sistemin payda katsayı tahminleri.



b) Sistemin pay katsayı tahminleri.



c) Sistemin referans ve çıkış sinyalleri.



d) Sistemin hesaplanan kontrol girişi.

Şekil 4.64. Tekrarlamalı GS algoritmasıyla elde edilen kutup yerleştirmeli, dolaylı model tabanlı uyarlamalı kontrol benzetim sonuçları.

Dolaylı model tabanlı uyarlamalı kontrol yönteminde model parametrelerini tahmin etmek için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.18`de verilmiştir.

Çizelge 4.18. Dolaylı model tabanlı uyarlamalı kontrol yönteminde model parametrelerini tahmin etmek için kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (yaklaşık örnek sayısı)
GS	56	400
RLS	100	10

4.8. Tekrarlamalı Gauss-Seidel Algoritması ile Doğrudan Uyarlamalı Kontrol

Bu kısımda, önerilen tekrarlamalı GS algoritması sırasıyla model tabanlı uyarlamalı denetleyici ve kendinden ayarlamalı denetleyici parametrelerinin doğrudan ayarlanmasında kullanılmıştır. Elde edilen sonuçlar, yaygın olarak kullanılan RLS algoritmasıyla elde edilen sonuçlarla karşılaştırılmıştır.

4.8.1. Tekrarlamalı Gauss-Seidel algoritması ile doğrudan model tabanlı uyarlamalı kontrol

Örnek-1: Bu kısımda yapılan ilk benzetim çalışmasında Akhtar ve Bernstein (2005) tarafından kullanılan aşağıdaki kararsız minimum fazlı doğrusal ayrık-zaman sistem modeli kullanılmıştır. Sistemin giriş ve çıkış işaretleri arasındaki zaman gecikmesi $d = 2$ örnekleme periyoduna eşittir.

$$\frac{B(z)}{A(z)} = \frac{z + 0.5}{z^3 + z^2 + z + 1.5} \tag{4.102}$$

Kapalı çevrim sistemin takip ettiği model

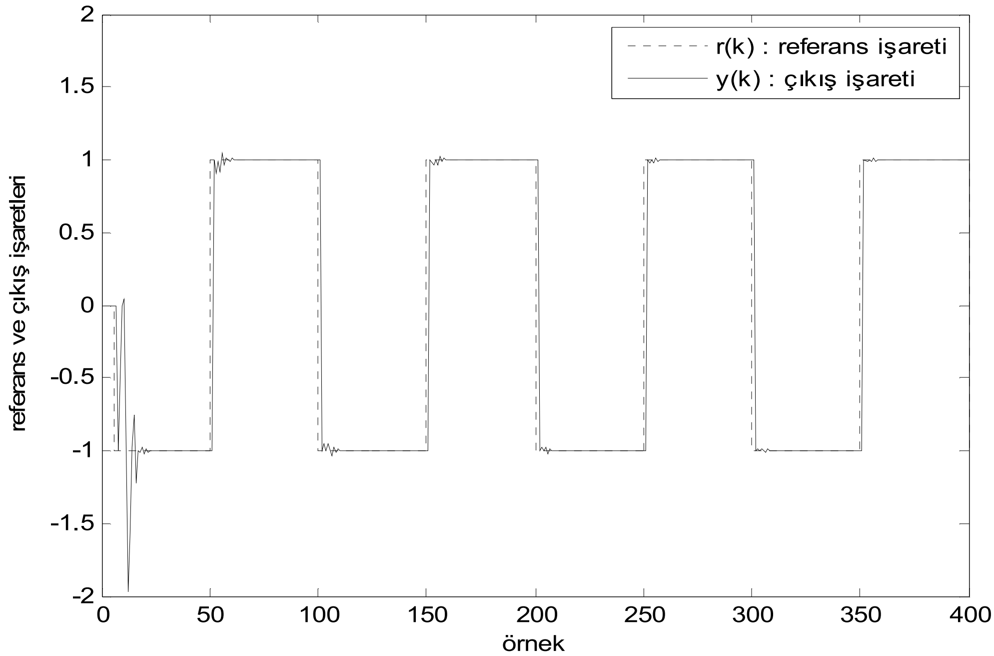
$$\frac{B_m(z)}{A_m(z)} = \frac{z^2}{z^4} \tag{4.103}$$

olarak seçilmiştir (Akhtar ve Bernstein 2005). Burada kullanılan veri vektörü ve denetleyicinin parametre vektörü aşağıdaki gibidir.

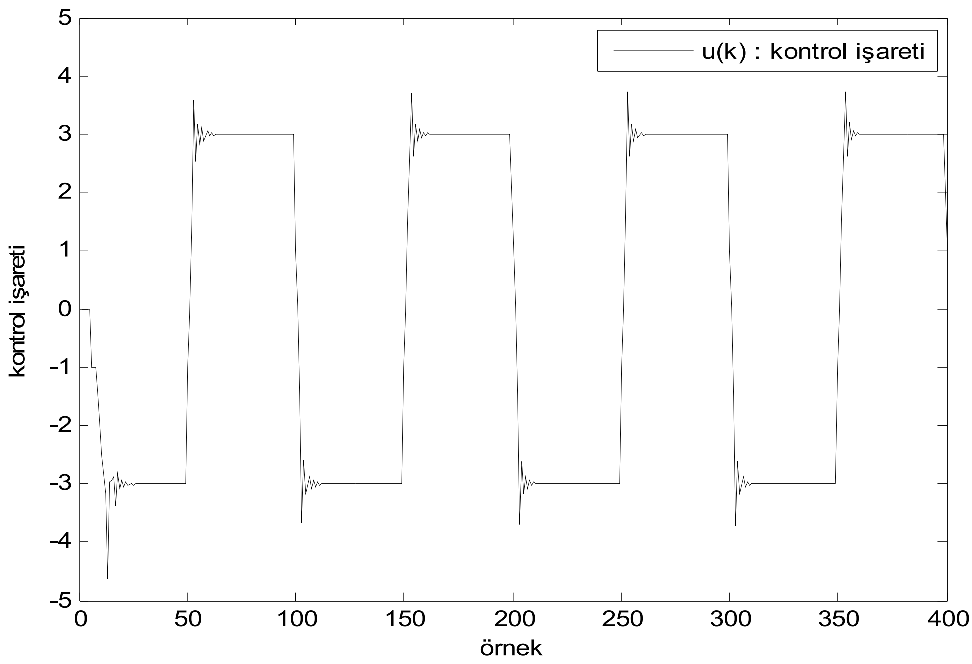
$$\phi^T(k-d) = [u(k-d-1) \quad u(k-d-2) \quad y(k-d) \quad y(k-d-1) \quad y(k-d-2)] \tag{4.104}$$

$$\theta_{opt}^T = \begin{bmatrix} r_1 & r_2 & s_0 & s_1 & s_2 \end{bmatrix} = \begin{bmatrix} -0.5 & -0.5 & 0.0 & -0.5 & 1.5 \end{bmatrix} \tag{4.105}$$

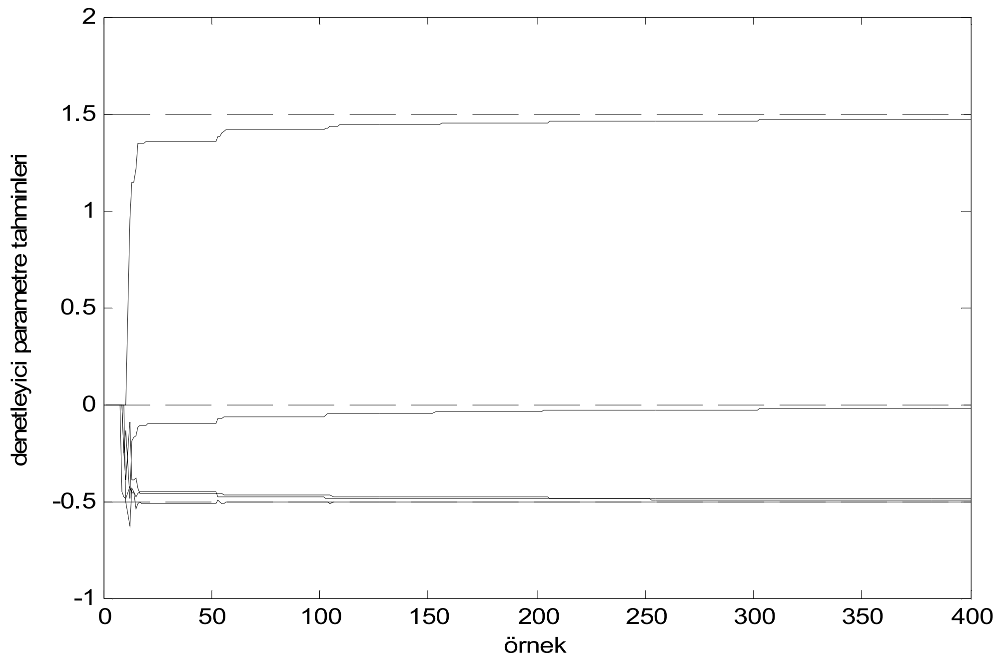
Bu benzetim çalışmasında $r_0 = b_0 = 1$ alınmıştır. Giriş işareti olarak periyodu 100 örnekleme periyoduna eşit olan ve genliği ∓ 1 birim olan kare dalga kullanılmıştır. Yapılan karşılaştırmalı benzetim çalışmasında tekrarlamalı GS algoritmasında korelasyon matrisinin başlangıç değeri $R(0)=0.1I_{(5\times5)}$ alınmıştır, RLS algoritmasında korelasyon matrisinin tersinin başlangıç değeri $R^{-1}(0)=10\,I_{(5\times5)}$ alınmıştır. NLMS algoritmasında adım parametresi $\mu=1.25$ alınmıştır. Bütün algoritmalarda parametre tahminlerinin başlangıç değeri 0.1 alınmıştır. Aşağıda sırasıyla Şekil 4.65'te RLS, Şekil 4.66'da tekrarlamalı GS ve Şekil 4.67'de NLMS algoritmalarıyla hesaplanan parametre tahminleri, sistemin referans ve çıkış ile kontrol işaretleri görülmektedir. Korelasyon matrisinin özdeğer yayılımı 2577 olarak hesaplanmıştır.



a) Sistemin referans ve çıkış işaretleri.

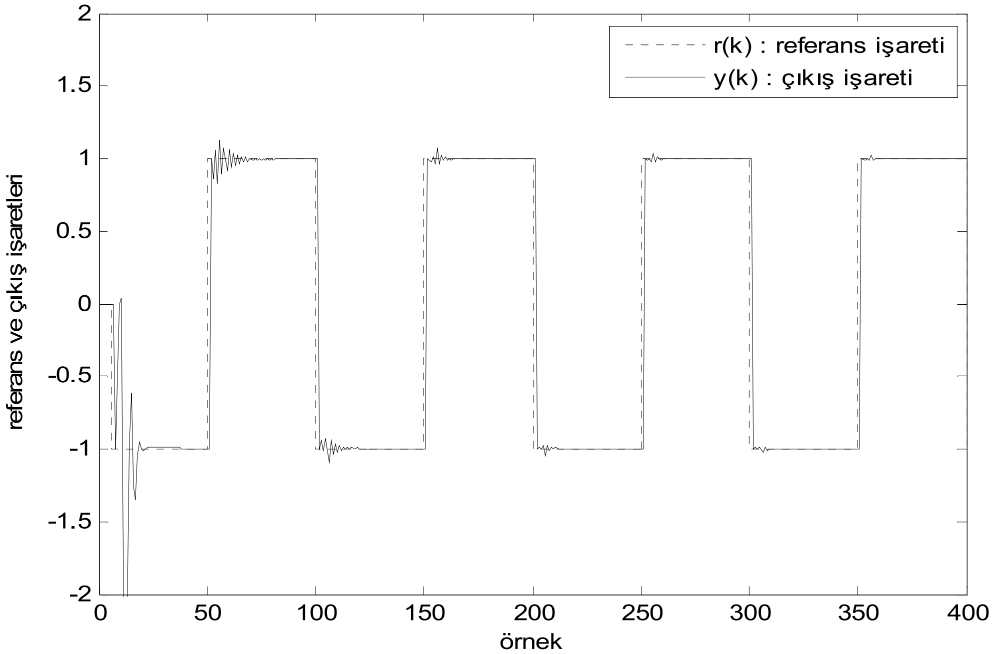


b) Sistemin hesaplanan kontrol girişi.

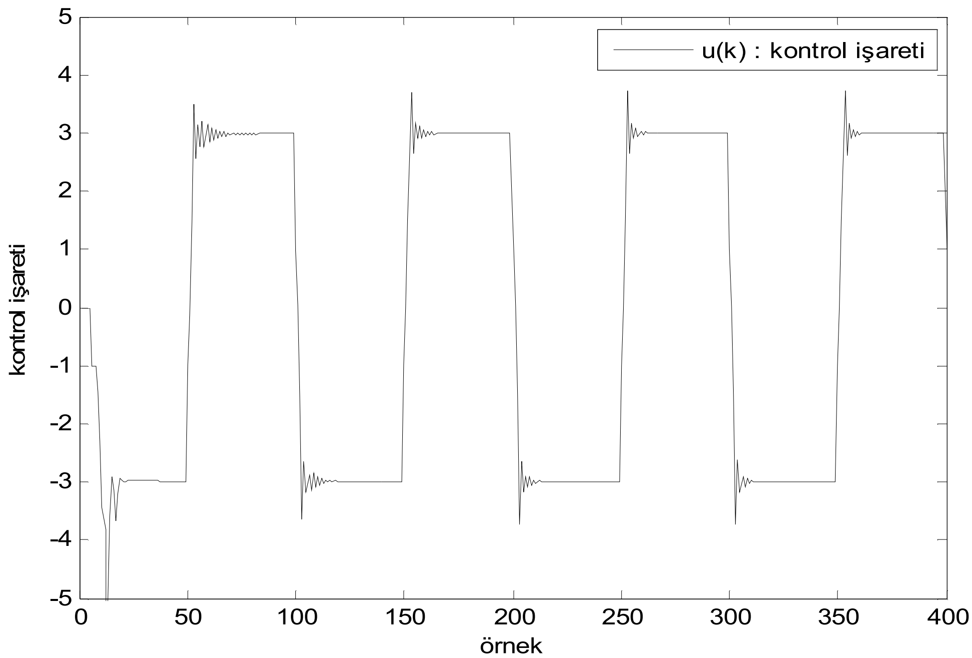


c) Denetleyici parametre tahminleri.

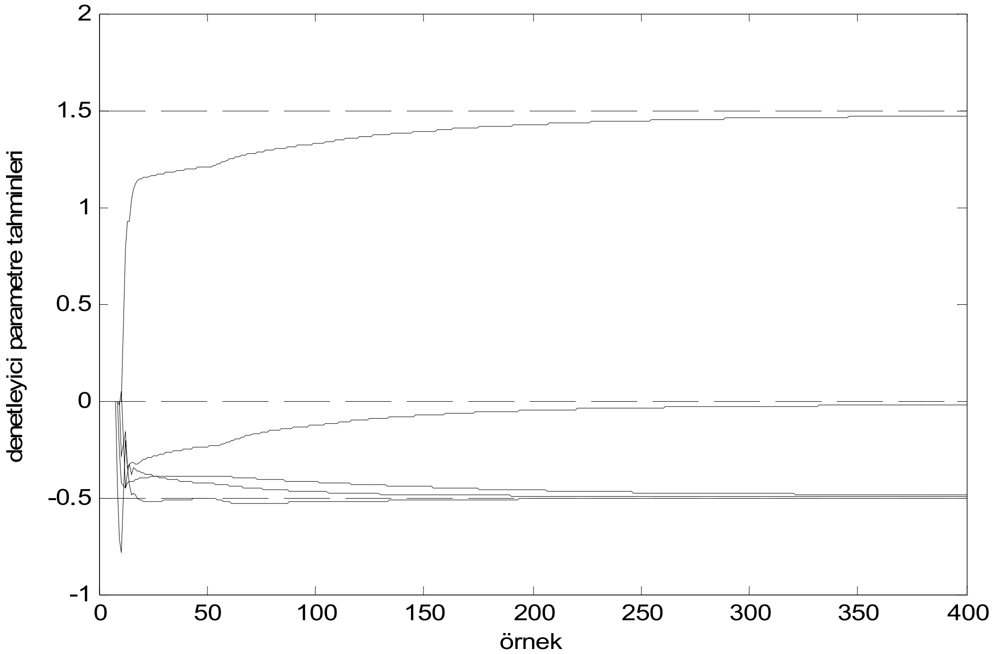
Şekil 4.65. RLS algoritmasıyla elde edilen doğrudan model tabanlı uyarlamalı kontrol benzetim sonuçları ($r_0 = b_0 = 1$ alınmıştır).



a) Sistemin referans ve çıkış işaretleri.

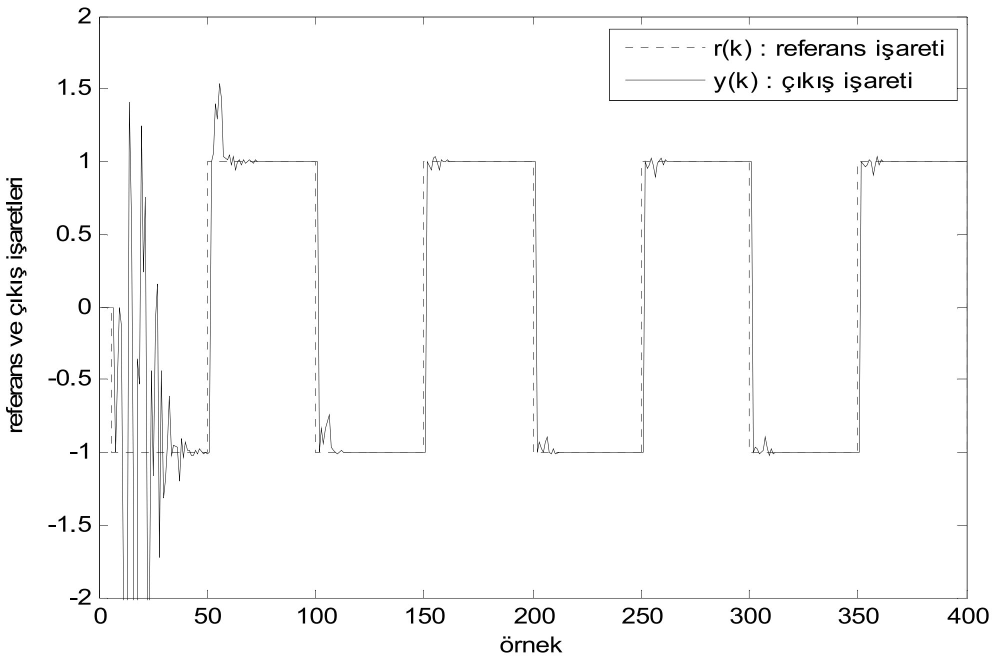


b) Sistemin hesaplanan kontrol girişi.

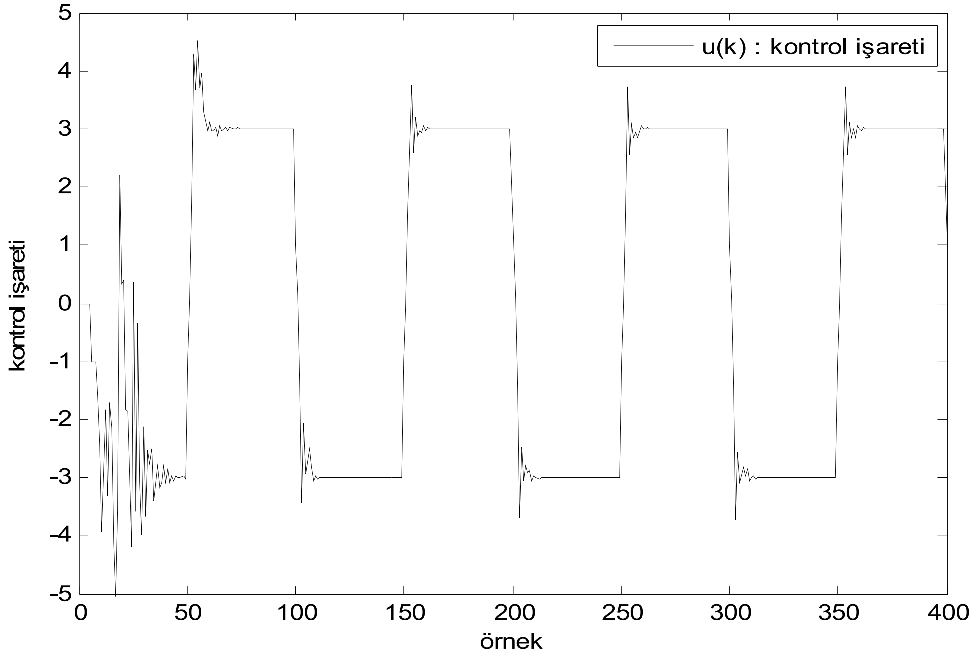


c) Denetleyici parametre tahminleri.

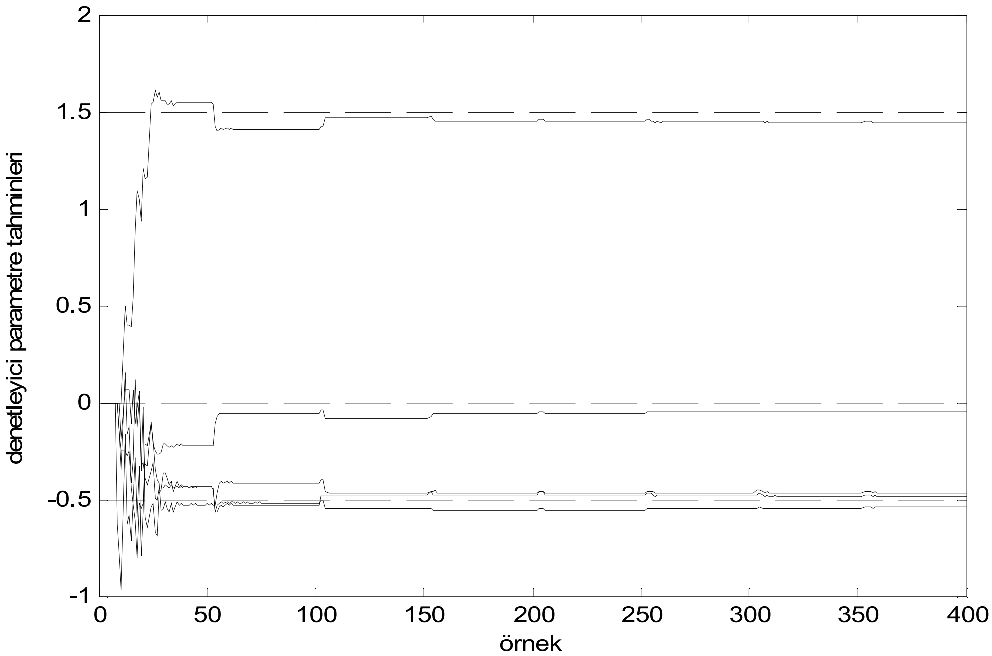
Şekil 4.66. Tekrarlamalı GS algoritmasıyla elde edilen doğrudan model tabanlı uyarlamalı kontrol benzetim sonuçları ($r_0 = b_0 = 1$ alınmıştır).



a) Sistemin referans ve çıkış işaretleri.



b) Sistemin hesaplanan kontrol girişi.



c) Denetleyici parametre tahminleri.

Şekil 4.67. NLMS algoritmasıyla elde edilen doğrudan model tabanlı uyarlamalı kontrol benzetim sonuçları ($r_0 = b_0 = 1$ alınmıştır).

Örnek-2: Yapılan ikinci benzetim çalışmasında bir doğru akım motoruna sert bir mil ile bağlı bir yükten oluşan bir sistemin hız kontrol modeli kullanılmıştır. Kullanılan motorun armatür direnci $R_a = 1\Omega$, armatür endüktansı $L_a = 0.5\,\mathrm{H}$, motorun ve yükün eylemsizliği $J = 0.01$ kgm^2/s^2, motorun ve yükün sürtünme katsayısı $B = 0.1\,$Nms, motorun zıt elektromotor sabiti $K_b = 0.01\,$V/(rad/s) ve motor tork sabiti $K_t = 0.01\,$Nm/A olarak alınmıştır. Bu durumda motor ve yükten oluşan sistemde armatür geriliminden ve yük torkundan yükün açısal hızına kadar olan sürekli zaman transfer fonksiyonu

$$\Omega_m(s) = \frac{\dfrac{K_t}{L_a J}}{s^2 + \left(\dfrac{R_a}{L_a} + \dfrac{B}{J}\right)s + \dfrac{R_a B + K_b K_t}{L_a J}} E_a(s) + \frac{\dfrac{1}{J}\left(s + \dfrac{R_a}{L_a}\right)}{s^2 + \left(\dfrac{R_a}{L_a} + \dfrac{B}{J}\right)s + \dfrac{R_a B + K_b K_t}{L_a J}} T_L(s) \qquad (4.106)$$

olarak yazılabilir (Kuo ve Golnaraghi 2003). Burada $\Omega_m(s)$ motorun açısal hızının, $E_a(s)$ motora uygulanan armatür geriliminin, $T_L(s)$ yük torkunun Laplace dönüşümünü göstermektedir. Kapalı-çevrim sistemin çıkışının, sönüm oranı $\xi = 1.2$ doğal frekansı $\omega_n = 10\ rad/s$ olan aşağıdaki referans modelin çıkışını takip etmesi istenmektedir.

$$\frac{Y_m(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} = \frac{100}{s^2 + 24s + 100} \qquad (4.107)$$

Örnekleme frekansı, referans modelin bant genişliğinin 2 katı Nyquist frekansı olmak üzere, bu frekansın yaklaşık 10 katı olacak şekilde $f_s \cong 10\,$Hz. olarak seçilmiştir (Franklin ve ark. 1998). Referans modelin sıfırıncı derece tutuculu eşdeğeri $T_s \cong 0.1\,$s. için

$$\frac{Y_m(z^{-1})}{R(z^{-1})} = \frac{0.1144z^{-1} + 0.0749z^{-2}}{1 - 1.0516z^{-1} + 0.2369z^{-2}} \qquad (4.108)$$

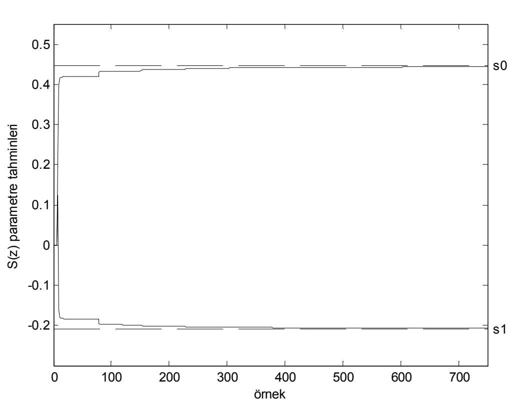
olarak elde edilir. Burada kullanılan veri vektörü ve denetleyicinin parametre vektörü aşağıdaki gibidir.

$$\phi^T(k-d) = [u(k-d) \quad u(k-d-1) \quad y(k-d) \quad y(k-d-1)] \tag{4.109}$$

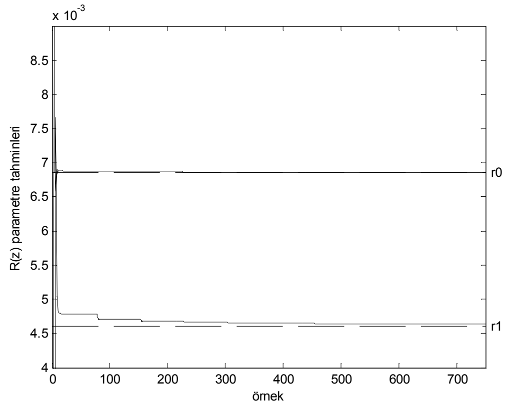
$$\theta_{opt}^T = \left[r_0 \quad r_1 \quad s_0 \quad s_1\right] = \left[0.0069 \quad 0.0046 \quad 0.4467 \quad -0.2105\right] \tag{4.110}$$

Bu benzetim çalışmasında $r_0 = b_0$ parametresi tahmin edilmiştir. Giriş işareti olarak periyodu 150 örnekleme periyoduna eşit olan ve genliği ∓ 1 birim olan kare dalga kullanılmıştır.

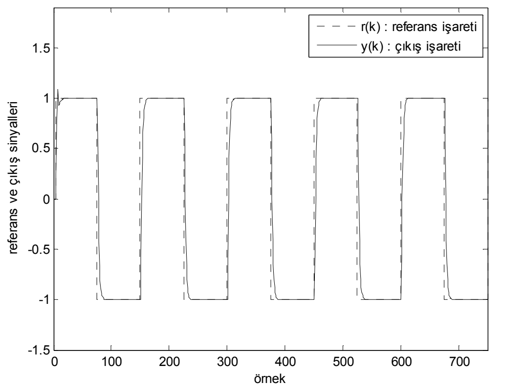
Tekrarlamalı GS algoritmasında korelasyon vektörünün başlangıç değeri sıfır vektör, korelasyon matrisinin başlangıç değeri $R(0) = 0.01.I$, tersinin başlangıç değeri RLS algoritmasında $P(0) = 100.I$ olarak alınmıştır. Armatür gerilimi $\pm 50\,\mathrm{V}$ ile sınırlandırılmıştır. Aşağıda sırasıyla Şekil 4.68'de RLS ve Şekil 4.69'da tekrarlamalı GS algoritmalarıyla hesaplanan parametre tahminleri, sistemin referans ve çıkış ile kontrol işaretleri görülmektedir. Korelasyon matrisinin özdeğer yayılımı yaklaşık $1{,}5.10^5$ olarak hesaplanmıştır. Bu değer çok büyük olduğu için NLMS algoritmasıyla kullanışlı parametre tahminleri elde edilememiştir.



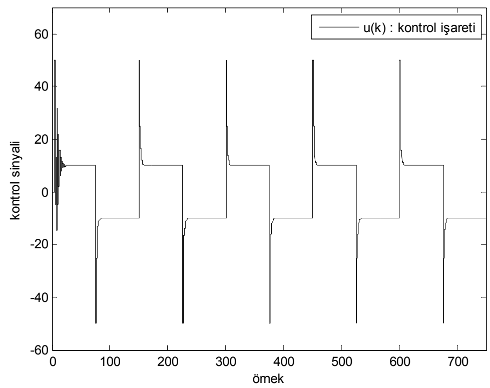
a) $S(z)$ parametre tahminleri.



b) $R(z)$ parametre tahminleri.

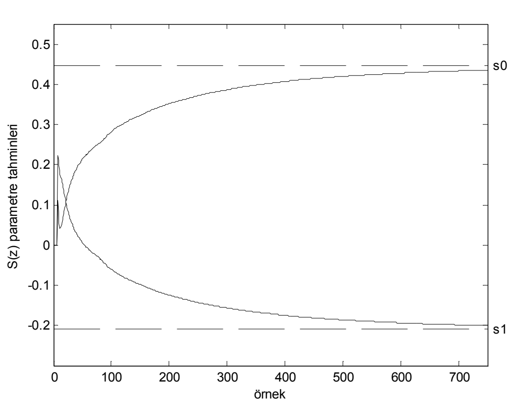


c) Sistemin referans ve çıkış işaretleri.

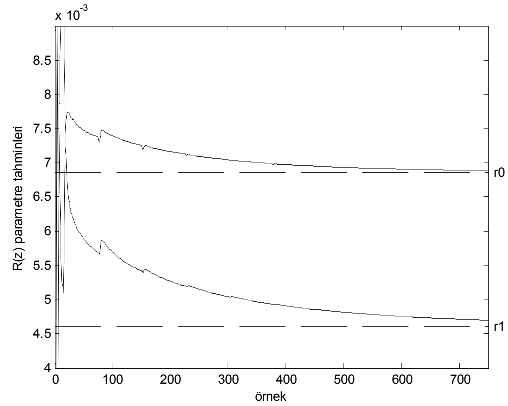


d) Sistemin hesaplanan kontrol girişi.

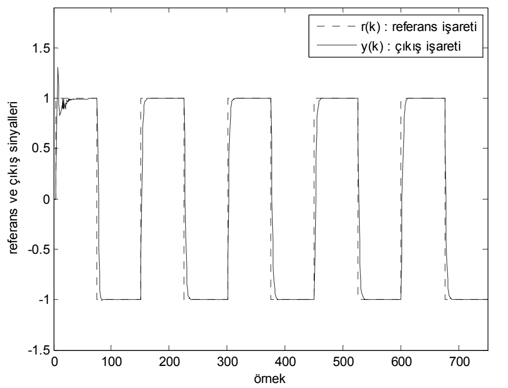
Şekil 4.68. RLS algoritmasıyla elde edilen doğrudan model tabanlı uyarlamalı kontrol benzetim sonuçları ($r_0 = b_0$ tahmin edilmiştir).



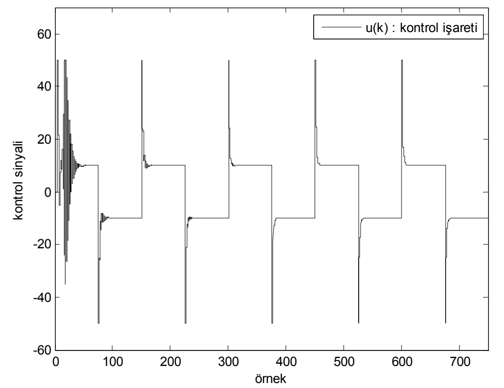
a) $S(z)$ parametre tahminleri.



b) $R(z)$ parametre tahminleri.



c) Sistemin referans ve çıkış işaretleri.



d) Sistemin hesaplanan kontrol girişi.

Şekil 4.69. Tekrarlamalı GS algoritmasıyla elde edilen doğrudan model tabanlı uyarlamalı kontrol benzetim sonuçları ($r_0 = b_0$ tahmin edilmiştir).

Doğrudan model tabanlı uyarlamalı kontrol yönteminde denetleyici parametrelerini doğrudan ayarlamak için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.19`da verilmiştir.

Çizelge 4.19. Doğrudan model tabanlı uyarlamalı kontrol yönteminde denetleyici parametrelerini doğrudan ayarlamak için kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (örnek-1 için) (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (örnek-1 için) (yaklaşık örnek sayısı)
GS	85	20
RLS	138	20
NLMS	17	50
Algoritma:	İşlem Yükü (örnek-2 için) (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (örnek-2 için) (yaklaşık örnek sayısı)
GS	56	500
RLS	100	100

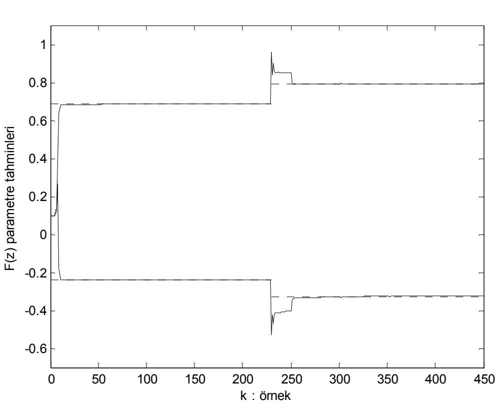
4.8.2. Tekrarlamalı Gauss-Seidel algoritması ile doğrudan kendinden ayarlamalı kontrol

Örnek-1: İlk benzetim çalışmasında yine önceki örnekte kullanılan, bir doğru akım motoruna sert bir mil ile bağlı bir yükten oluşan bir sistemin hız kontrol modeli kullanılmıştır. Kullanılan motorun parametreleri, örnekleme periyodu ve referans model önceki örnekteki gibi seçilmiştir. Önceki örnekten farklı olarak GMV kontrol tabanlı olan bir uyarlamalı kontrol yöntemi kullanılmıştır. Bu yönteme göre, sistemin transfer fonksiyonu minimum fazlı olduğu için $Q(z^{-1})=0$ ve $C(z^{-1})=1-0.5z^{-1}+0.06z^{-1}$ seçilmiştir. Bu durumda denetleyici polinomları 1. derece olmaktadır. Tekrarlamalı GS algoritmasında korelasyon vektörünün başlangıç değeri sıfır vektör, korelasyon matrisinin başlangıç değeri $R(0)=0.001.I$, tersinin başlangıç değeri RLS algoritmasında $R^{-1}(0)=1000.I$ olarak alınmıştır. Her iki algoritmada parametre tahminlerinin başlangıç değeri 0.1 ve unutma faktörü $\lambda=0.95$ olarak seçilmiştir. Sistemde $k=230.$ adımdan itibaren motorun ve yükün toplam eylemsizliği 2 katına, motorun ve yükün toplam sürtünme katsayısı 1.5 katına çıkarılmış ve sabit bozucu yük torku uygulanmıştır. Armatür gerilimi $\pm 50\,\mathrm{V}$ ile sınırlandırılmıştır. Giriş işareti olarak periyodu 100 örnekleme periyoduna eşit olan ve genliği ∓ 1 birim olan kare dalga kullanılmıştır. Burada kullanılan veri vektörü ve denetleyici parametre tahminleri vektörünün yapısı aşağıdaki gibidir.

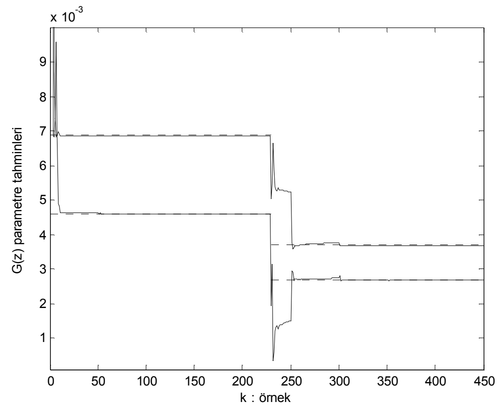
$$\phi^T(k-d) = [\, y(k-d) \quad y(k-d-1) \quad u(k-d) \quad u(k-d-1)] \qquad (4.111)$$

$$\hat{\theta}^T(k) = \left\lfloor \hat{f}_0(k) \quad \hat{f}_1(k) \quad \hat{g}_0(k) \quad \hat{g}_1(k) \right\rfloor \qquad (4.112)$$

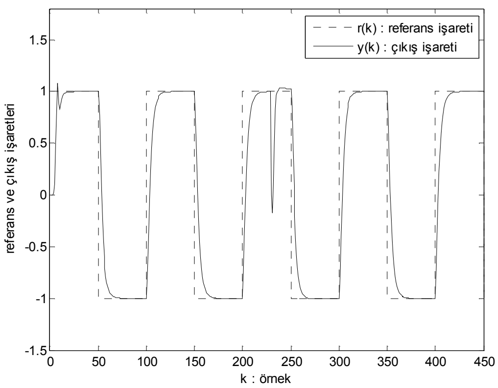
Aşağıda sırasıyla Şekil 4.70'te RLS ve Şekil 4.71'de tekrarlamalı GS algoritmalarıyla hesaplanan parametre tahminleri, sistemin referans ve çıkış ile kontrol işaretleri görülmektedir. Korelasyon matrisinin özdeğer yayılımı yaklaşık $1{,}4.10^6$ olarak hesaplanmıştır. Bu değer çok büyük olduğu için NLMS algoritmasıyla kullanışlı parametre tahminleri elde edilememiştir.



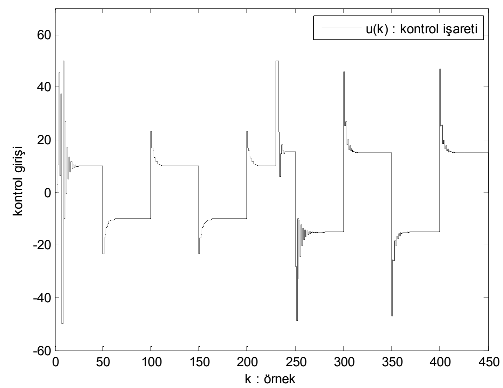
a) $F(z)$ parametre tahminleri.



b) $G(z)$ parametre tahminleri.

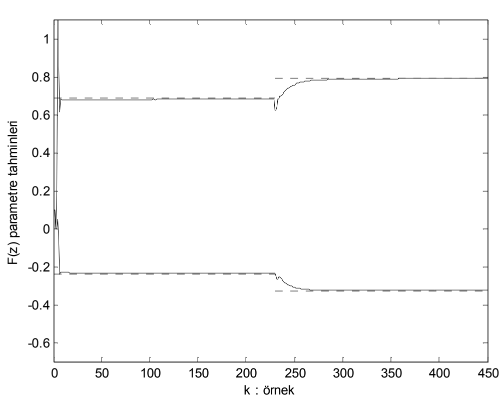


c) Sistemin referans ve çıkış işaretleri.

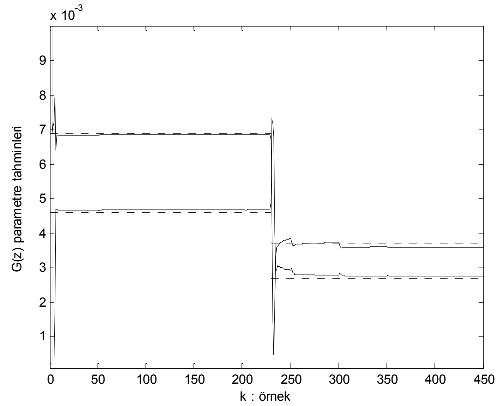


d) Sistemin hesaplanan kontrol girişi.

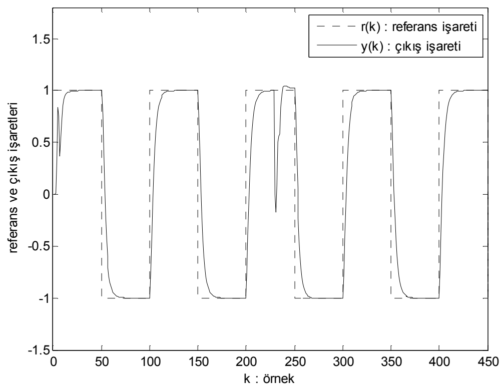
Şekil 4.70. RLS algoritmasıyla elde edilen doğrudan kendinden ayarlamalı kontrol benzetim sonuçları.



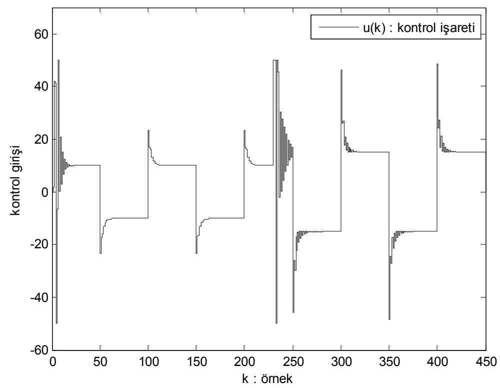
a) $F(z)$ parametre tahminleri.



b) $G(z)$ parametre tahminleri.



c) Sistemin referans ve çıkış işaretleri.



d) Sistemin hesaplanan kontrol girişi.

Şekil 4.71. Tekrarlamalı GS algoritmasıyla elde edilen doğrudan kendinden ayarlamalı kontrol benzetim sonuçları.

Örnek-2: Yapılan ikinci benzetim çalışmasında minimum fazlı olmayan aşağıdaki deterministik sistem kullanılmıştır (Patete ve ark. 2007).

$$(1-0.5z^{-1})y(k) = z^{-2}(1+2.3z^{-1})u(k) \tag{4.113}$$

Tekrarlamalı GS algoritmasında korelasyon vektörünün başlangıç değeri sıfır vektör, korelasyon matrisinin başlangıç değeri $R(0) = 0.01.I$, tersinin başlangıç değeri RLS algoritmasında $P(0) = 100.I$ olarak alınmıştır. Her iki algoritmada unutma faktörü $\lambda = 0.9$ olarak seçilmiştir. Sistem parametre değerleri başlangıçta

$$(1-0.3z^{-1})y(k)=z^{-2}(1+2z^{-1})u(k) \tag{4.114}$$

alınarak GMV kontrol yönteminde

$$Q(z^{-1})=40(1-z^{-1}) \ , \ \ C(z^{-1})=1+z^{-1}+0.25z^{-2} \tag{4.115}$$

olarak seçilmiş ve denetleyici parametre tahminlerinin başlangıç değerleri

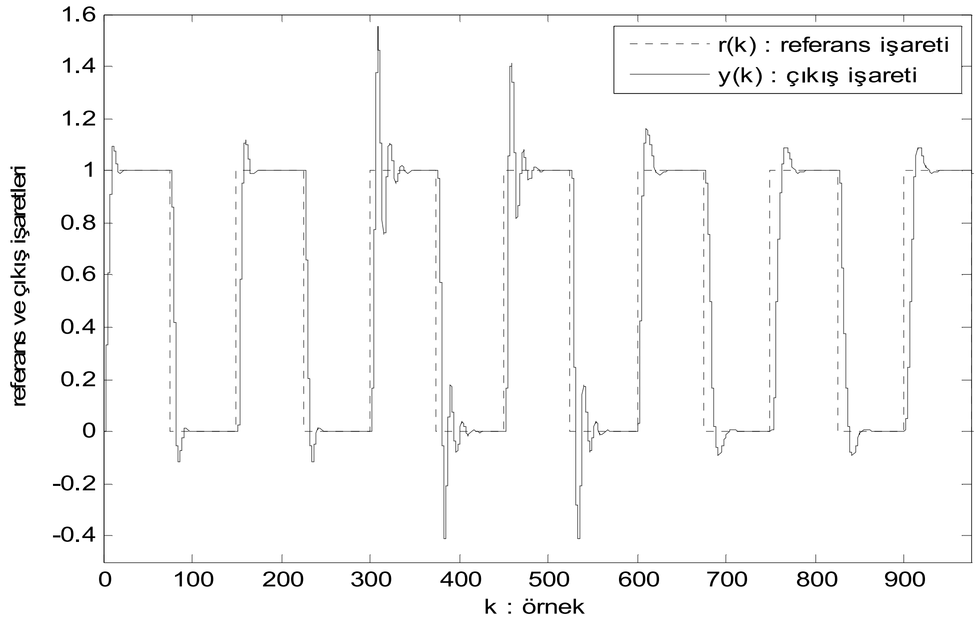
$$\hat{F}(z^{-1})=\hat{f}_0=0.64 \tag{4.116}$$

$$\hat{G}(z^{-1})=\hat{g}_0+\hat{g}_1 z^{-1}+\hat{g}_2 z^{-2}=41-36.7z^{-1}+2.6z^{-2} \tag{4.117}$$

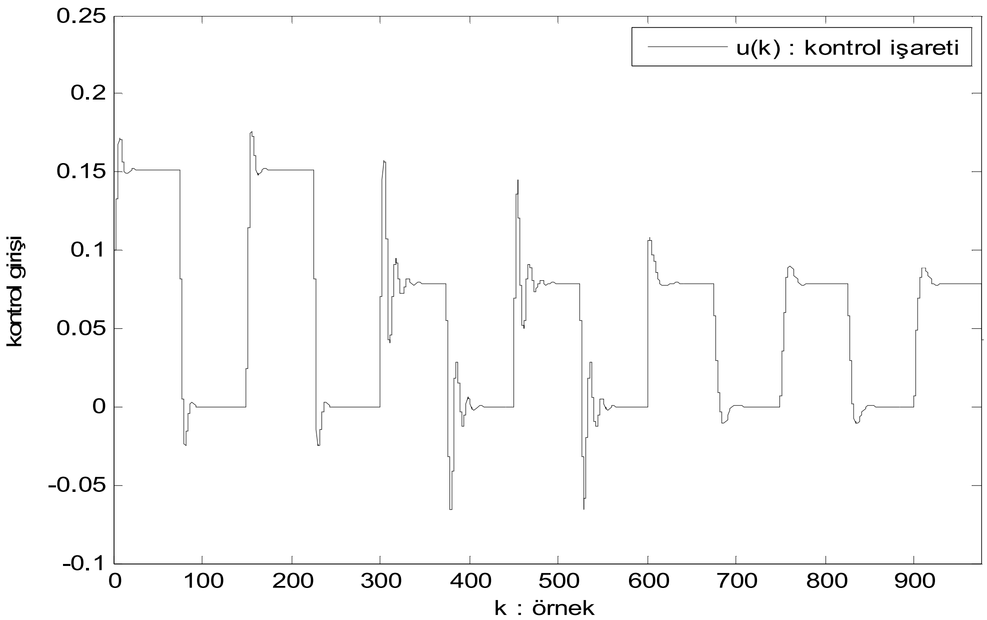
olarak hesaplanmıştır. Sistem parametreleri $k=300.$ adımda

$$(1-0.7z^{-1})y(k)=z^{-2}(1.2+2.6z^{-1})u(k) \tag{4.118}$$

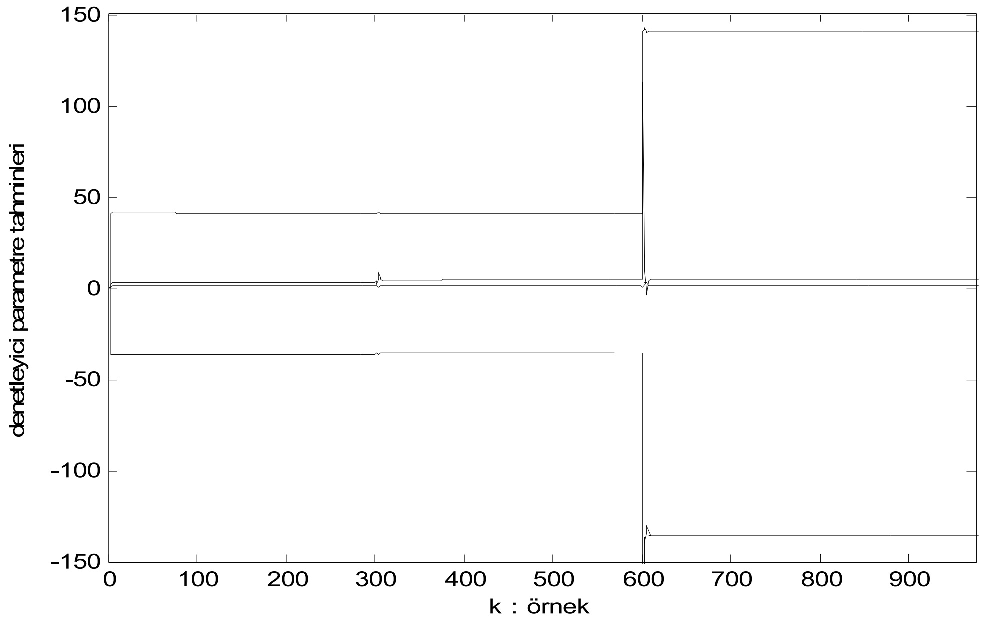
olarak değiştirilmiş ve $k=600.$ adımda $Q(z^{-1})=120(1-z^{-1})$ olarak değiştirilmiştir. Referans giriş işareti olarak genliği 0~1 birim arasında değişen kare dalga seçilmiştir. Aşağıda sırasıyla Şekil 4.72'de RLS ve Şekil 4.73'te tekrarlamalı GS algoritmalarıyla hesaplanan parametre tahminleri, sistemin referans ve çıkış ile kontrol işaretleri görülmektedir. Korelasyon matrisinin özdeğer yayılımı yaklaşık $1{,}2.10^{6}$ olarak hesaplanmıştır. Bu değer çok büyük olduğu için NLMS algoritmasıyla kullanışlı parametre tahminleri elde edilememiştir.



a) Sistemin referans ve çıkış işaretleri.

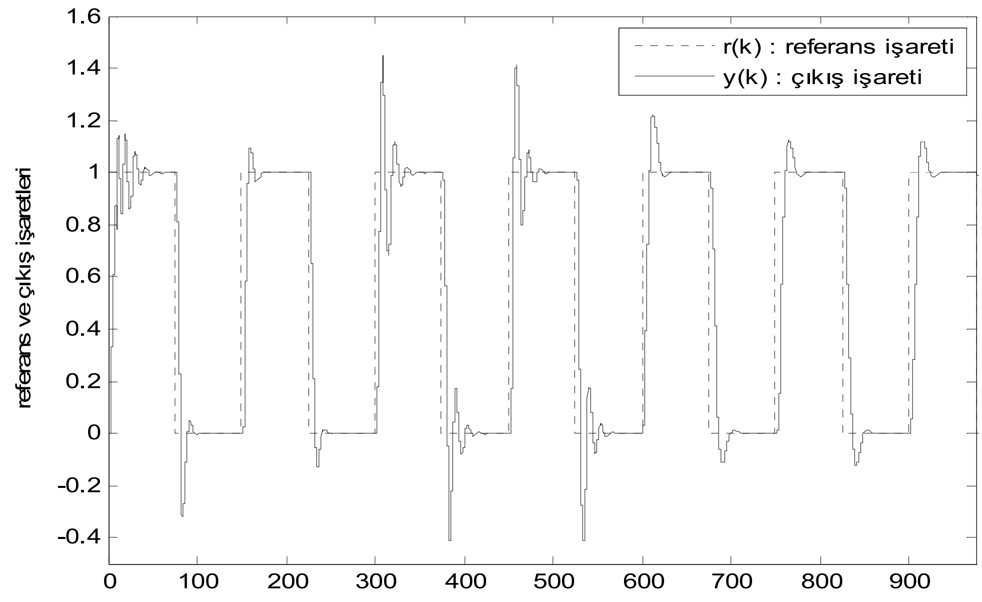


b) Sistemin hesaplanan kontrol girişi.

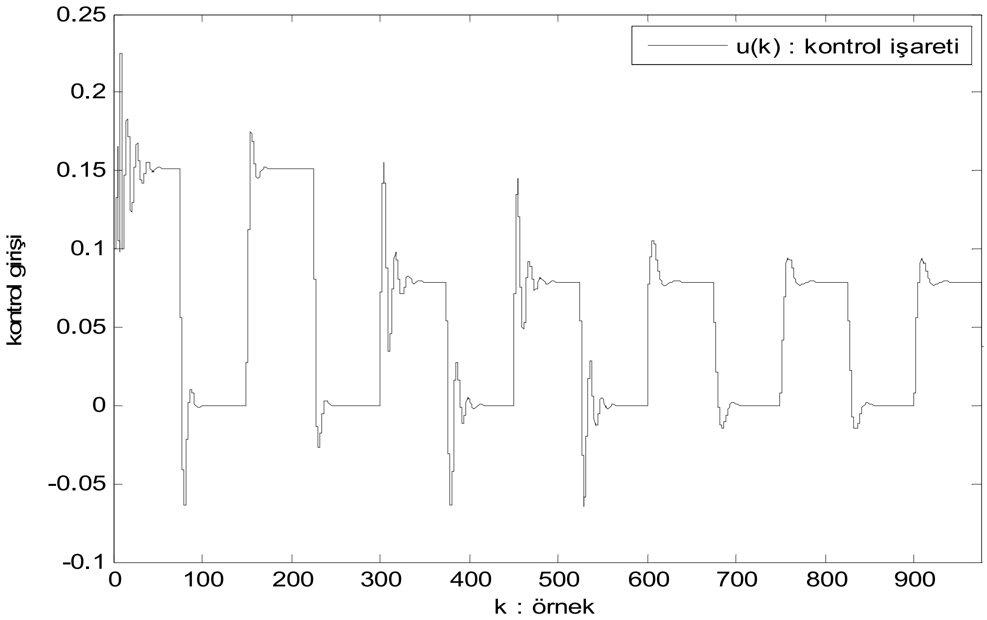


c) Denetleyici parametre tahminleri.

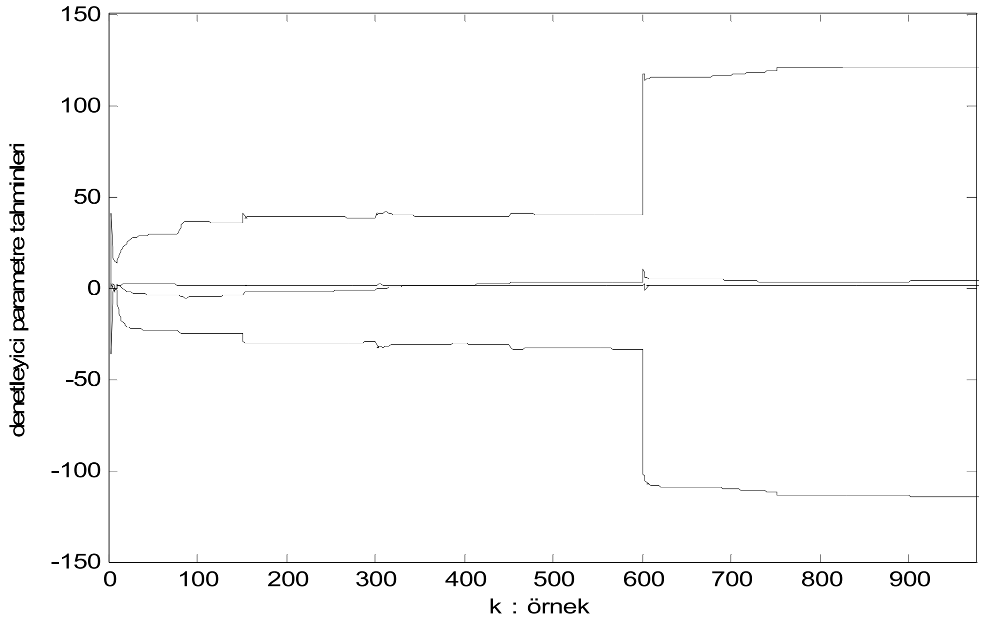
Şekil 4.72. RLS algoritmasıyla elde edilen doğrudan kendinden ayarlamalı kontrol benzetim sonuçları. (minimum fazlı olmayan sistem)



a) Sistemin referans ve çıkış işaretleri.



b) Sistemin hesaplanan kontrol girişi.



c) Denetleyici parametre tahminleri.

Şekil 4.73. Tekrarlamalı GS algoritmasıyla elde edilen doğrudan kendinden ayarlamalı kontrol benzetim sonuçları. (minimum fazlı olmayan sistem)

Doğrudan kendinden ayarlamalı kontrol yönteminde denetleyici parametrelerini doğrudan ayarlamak için yapılan benzetim çalışmasında kullanılan algoritmaların karşılaştırılması Çizelge 4.20`de verilmiştir.

Çizelge 4.20. Doğrudan kendinden ayarlamalı kontrol yönteminde denetleyici parametrelerini doğrudan ayarlamak için kullanılan algoritmaların karşılaştırılması.

Algoritma:	İşlem Yükü (örnek-1 için) (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (örnek-1 için) (yaklaşık örnek sayısı)
GS	56	30
RLS	100	20
Algoritma:	İşlem Yükü (örnek-2 için) (çarpma-bölme işlemi sayısı)	Yakınsama Hızı (örnek-2 için) (yaklaşık örnek sayısı)
GS	56	150
RLS	100	10

SONUÇ

Bu tez çalışmasında, uyarlamalı sistem tanıma ve uyarlamalı kontrol uygulamalarında yaygın olarak kullanılan RLS ve NLMS algoritmalarına alternatif olabilecek olan tekrarlamalı GS algoritması önerilmiştir. Ayrıca benzer algoritmalar olan tekrarlamalı Jacobi ve tekrarlamalı SOR algoritmalarının kullanımı önerilmiştir. Önerilen algoritmalar ayrık-zaman sistem parametrelerinin tahmin edilmesinde, sürekli-zaman sistem parametrelerinin tahmin edilmesinde, çok girişli çok çıkışlı sistem parametrelerinin tahmin edilmesinde, bazı zaman-serileri modellerinin parametrelerinin tahmin edilmesi, doğrusal olmayan Volterra model parametrelerinin tahmin edilmesinde ve uyarlamalı denetleyici parametrelerinin dolaylı olarak ve doğrudan ayarlanmasında kullanılmıştır.

Bu tez çalışmasının bilime olan katkıları aşağıdaki gibi özetlenebilir:

- Çevrim-içi sistem tanıma işleminde kullanmak için matematiksel olarak farklı bir temeli olan, yakınsama hızı ve işlem yükü açısından bir ara yöntem olarak yaygın olarak kullanılan RLS ve NLMS algoritmalarına alternatif olan tekrarlamalı GS algoritması önerilmiştir.
- Önerilen tekrarlamalı GS algoritması ayrık zaman sistemlerin ve sürekli zaman sistemlerin transfer fonksiyonu parametrelerinin yansız olarak tahmin edilmesinde kullanılmıştır. Ayrıca parametre tahminleri vektörünün stokastik yakınsama analizi yapılarak yansız parametre tahminlerinin elde edilebilmesi için gerekli şartlar analitik olarak verilmiştir.
- Tek girişli tek çıkışlı sistemler için önerilen GS algoritması çok girişli çok çıkışlı sitemlerin transfer fonksiyonu matrisi parametrelerinin tahmin edilmesi durumu için geliştirilmiş ve hem ayrık zaman hem de sürekli zaman sistemlerin transfer fonksiyonu matrisi parametrelerinin yansız olarak tahmin edilmesinde kullanılmıştır. Ayrıca stokastik yakınsama analizi yapılarak yansız parametre tahminlerinin elde edilebilmesi için gerekli şartlar analitik olarak verilmiştir.

- Önerilen tekrarlamalı GS algoritması ayrık zaman sistemlerin modellenmesinde kullanılan ve stokastik bozucu girişlerin veya ölçme gürültüsünün etkilediği değişik zaman serileri modellerinin parametrelerinin çevrim-içi olarak tahmin edilmesinde kullanılmıştır. Bu durumda parametre tahminleri vektörünün birinci derece ve ikinci derece yakınsama analizleri yapılarak elde edilen sonuçlar verilmiştir.
- Tekrarlamalı GS algoritması ve benzer yapıya sahip olan tekrarlamalı Jacobi algoritması doğrusal olmayan Volterra model parametrelerini çevrim-içi tahmin etmek için önerilmiştir. Önerilen algoritmaların kararlı olması durumunda doğrusal olmayan sistemlerin parametre tahminlerinin de doğru değerlerine yakınsadığı gösterilmiştir.
- Uyarlamalı denetleyici parametrelerinin dolaylı olarak ayarlanmak için tekrarlamalı GS algoritması önerilmiş ve benzer yapıya sahip olan tekrarlamalı Jacobi ve tekrarlamalı SOR algoritmaları da kullanılmıştır.
- Uyarlamalı denetleyici parametrelerinin doğrudan ayarlanmak için tekrarlamalı GS algoritması önerilmiştir. Önerilen tekrarlamalı GS algoritmasının kullanılması durumunda oluşan kapalı çevrim sistemin karesel bir Lyapunov fonksiyonunun var olduğu gösterilerek asimptotik kararlı olduğu ispatlanmıştır.

Bu tezde önerilen algoritmalarla yapılan benzetim çalışmalarıyla, elde edilen sonuçlar özellikle RLS ve RLS tabanlı eşdeğer algoritmalarla ve NLMS algoritmasıyla elde edilen sonuçlarla karşılaştırılmıştır. Elde edilen benzetim sonuçları parametre tahminlerinin yakınsama hızı açısından karşılaştırıldığında tekrarlamalı GS algoritmasının ve tekrarlamalı SOR algoritmasının RLS algoritmasına yakın sonuçlar verdiği görülmüştür. Önerilen algoritmalar RLS ve diğer RLS tabanlı algoritmalarda olduğu gibi korelasyon matrisinin ve korelasyon matrisinin birikimli tahmin edilmiş değerlerini kullanmaktadır. Tekrarlamalı GS algoritmasına göre daha düşük yakınsama hızına sahip olan tekrarlamalı Jacobi algoritmasıyla yapılan benzetim sonuçları bazı uygulamalarda kıyaslama yapmak için kullanılmıştır. Önerilen algoritmalar bu yüzden yakınsama hızı açısından RLS ve RLS tabanlı algoritmalara çok yakın sonuçlar vermektedir. Özellikle kontrol sistemlerinde NLMS algoritmasının yavaş yakınsamasından dolayı kullanılamadığı durumlarda önerilen algoritmalar iyi sonuçlar vermiştir. Yapılan benzetim çalışmalarıyla elde edilen sonuçlarda, önerilen algoritmaların RLS ve diğer RLS tabanlı algoritmalara alternatif olarak rahatlıkla kullanılabileceği görülmüştür. Özellikle çok girişli ve çok çıkışlı sistem tanıma uygulamalarında artan parametre sayısına bağlı olarak önerilen algoritmaların yakınsama hızı avantajının yanında işlem yükü avantajı önem kazandığı görülmüştür.

Bu tez çalışmasında elde edilen sonuçlar aşağıdaki gibi özetlenebilir:

- Önerilen tekrarlamalı GS algoritması sistemin çıkış işareti ölçümlerine karışan renkli ölçme gürültüsünün olumsuz etkisini ortadan kaldırmak için yardımcı değişkenler ile birlikte kullanılmıştır ve elde edilen parametre tahminlerinin eşdeğer tekrarlamalı yardımcı değişkenler algoritmasına çok yakın olduğu görülmüştür. Aynı algoritma parametre sayısının fazla olduğu çok girişli çok çıkışlı sistemlere de uygulanmış ve elde edilen parametre tahminlerinin eşdeğer RLS tabanlı algoritmalarla elde edilen sonuçlara çok yakın olduğu görülmüştür. Çok girişli çok çıkışlı sistemlerde tahmin edilen parametre sayısının fazla olduğu göz önüne alındığında tekrarlamalı GS algoritmasının yakınsama hızı avantajının yanında işlem yükü avantajı önem kazanmaktadır.
- Önerilen tekrarlamalı GS algoritması hem tek girişli tek çıkışlı sürekli zaman sistemlerin, hem de çok girişli çok çıkışlı sürekli zaman sistemlerin parametrelerinin tahmin edilmesinde de kullanılmış ve elde edilen sonuçların ayrık zamanlı sistemlerde olduğu gibi eşdeğer RLS tabanlı algoritmalarla elde edilen sonuçlara çok yakın olduğu görülmüştür. Sürekli zaman sistem tanıma uygulamalarında filtreleme işlemi işlem yükünü arttırsa da filtreleme işleminin bütün algoritmalarda ortak işlem olduğu düşünüldüğünde parametre tahminlerinin güncellenmesi aşamasındaki işlem yükü avantajının yine önemli olduğu görülmektedir.
- Önerilen tekrarlamalı GS algoritması, zaman ortalamalı normal denklemdeki korelasyon matrisi pozitif tanımlı olduğu için doğrusal olmayan Volterra model parametrelerinin tahmin edilmesinde de kullanılabilmiş ve parametre tahminlerinin yakınsama hızı açısından doğrusal sistemlerde elde edilen sonuçlara benzer sonuçlar elde edilmiştir. Tekrarlamalı GS algoritması parametre tahminlerinin yakınsama hızı açısından eşdeğer RLS algoritmasına çok yakın sonuçlar vermiştir. Volterra modellerdeki parametre sayısı fazla olduğu göz önüne alındığında tekrarlamalı GS

algoritmasının yakınsama hızı avantajının yanında işlem yükü avantajı önem kazanmaktadır.

- Tekrarlamalı GS algoritması kontrol sistemlerinde yaygın olarak kullanılan zaman serileri modellerinin parametrelerinin tahmin edilmesinde kullanılmış ve yine yakınsama hızı açısından eşdeğer algoritmalara çok yakın sonuçlar elde edilmiştir. Burada da önerilen algoritmanın yakınsama hızı ve işlem yükü avantajının önemli olduğu görülmüştür.
- Tekrarlamalı GS algoritmasıyla hesaplanan parametre tahminlerinin doğru değerine yakınsama hızı korelasyon matrisinin özdeğer yayılımına bağımlı olduğu için RLS algoritmasına göre düşüktür. Fakat yapılan dolaylı ve doğrudan uyarlamalı kontrol benzetim çalışmalarında elde edilen sonuçlara bakıldığında, sistemin giriş ve çıkış işaretlerinin yavaş değiştiği göz önüne alındığında aradaki yakınsama hızının fazla önemli olmadığı yapılan benzetim sonuçlarında görülmüştür. Ayrıca korelasyon matrisinin özdeğer yayılımının çok büyük olduğu durumlarda NLMS algoritmasıyla kullanışlı parametre tahminleri elde edilememiştir. Bu durumda bile önerilen tekrarlamalı GS algoritmasıyla kullanışlı parametre tahminleri elde edilebilmiştir.

KAYNAKLAR

AHMAD, N.A. 2007. Accelerated Euclidean Direction Search Algorithm and Related Relaxation Schemes for Solving Adaptive Filtering Problem. ICICS 2007, Proceedings of the 2007 6th International Conference on Information, Communications and Signal Processing, 10-13 Dec. 2007, page 1-5.

AKHTAR, S., D.S. BERNSTEIN. 2005. Lyapunov-Stable Discrete-Time Model Reference Adaptive Control. International Journal of Adaptive Control and Signal Processing. 19(10):745-767.

ANDERSON, B.D.O., C.R. JOHNSON. 1982. Exponential Convergence of Adaptive Identification and Control Algorithms. Automatica. 18(1):1-13.

ASTRÖM, K.J. 1980. Maximum Likelihood and Prediction Error Methods. Automatica. 16(5):551-574.

ASTRÖM, K.J. 1983. Theory and Applications of Adaptive Control – A Survey. Automatica. 19(5):471-486.

ASTRÖM, K.J. 1987. Adaptive Feedback Control. Proceedings of the IEEE. 75(2):185-217.

ASTRÖM, K.J., B. WITTENMARK. 1973. On Self Tuning Regulators. Automatica. 9(2):185-199.

ASTRÖM, K.J., B. WITTENMARK. 1980. Self-Tuning Controllers Based on Pole-Zero Placement. Proceedings of IEE, Part D – Control Theory and Applications. 127(3):120-130.

ASTRÖM, K.J., B. WITTENMARK. 1995. Adaptive Control. 2nd ed. Addison-Wesley, Reading, Massachusetts. 574 p.

ASTRÖM, K.J., U. BORISSON, L. LJUNG, B. WITTENMARK. 1977. Theory and Applications of Self-Tuning Regulators. Automatica. 13(5):457-476.

ASTRÖM, K.J., P. EYKHOFF. 1971. System Identification – A Survey. Automatica. 7(2):123-162.

ASTRÖM, K.J., T. HAGGLUND. 1988. Automatic Tuning of PID Controllers. Instrument Society of America, Research Triangle Park, North Carolina. 141 p.

ASTRÖM, K.J., T. HAGGLUND. 1995. PID Controllers: Theory, Design, and Tuning. 2nd ed. Instrument Society of America, Research Triangle Park, North Carolina. 343 p.

ASTRÖM, K.J., T. HAGGLUND. 2006. Advanced PID Control. The Instrumentation, Systems, and Automation Society, Research Triangle Park, North Carolina. 460 p.

BIERMANN, G.J. 1977. Factorization Methods for Discrete Sequential Estimation. Academic Press, New York. 241 p.

BITMEAD, R.R., B.D.O. ANDERSON. 1980. Lyapunov Techniques for the Exponential Stability of Linear Difference Equations with Random Coefficients. IEEE Trans. on Automatic Control. 25(4):782-787.

BOBAL, V. 1995. Self-Tuning Zeigler-Nichols PID Controller. International Journal of Adaptive Control and Signal Processing. 9(2):213-226.

BOBAL, V., J. BÖHM, R. BROKOP. 1999. Practical Aspects of Self-Tuning Controllers. International Journal of Adaptive Control and Signal Processing, 13(8):671-690.

BOBAL, V., J. BÖHM, J. FESSL, J. MACHACEK. 2005. Digital Self-Tuning Controllers: Algorithms, Implementation and Applications. Springer-Verlag, London. 317 p.

BOSE, T., M.Q. CHEN, G.F. XU. 1997. A Fast Adaptive Algorithm for Spectral Estimation. Proceedings of the 1997 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, 20-22 Aug. 1997, Vol. 1, page 146-149.

BOSE, T., G.F. XU. 2002. The Euclidean Direction Search Algorithm in Adaptive Filtering. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences. E85-A(3):532-539.

BOSE, T. 2004. Digital Signal and Image Processing. John Wiley, New Jersey. 656 p.

BUTLER, H. 1992. Model Reference Adaptive Control: From Theory to Practice. Prentice Hall, New York. 265 p.

CARINI, A, G.L. SICURANZA. 2004. Filtered-X Affine Projection Algorithms for Active Noise Control Using Volterra Filters. EURASIP Journal on Applied Signal Processing, 12:1841-1848.

CHALAM, V.V. 1987. Adaptive Control Systems: Techniques and Applications. Marcel Dekker, New York. 526 p.

CHEN, M.Q. 1998. A Direction Set Based Algorithm for Least Squares Problems in Adaptive Signal Processing. Linear Algebra and Its Applications, 284(1-3):73-94.

CHEN, M.Q., T. BOSE, G.F. XU. 1997. A Direction Set Based Algorithm for Adaptive Filtering. Proceedings of the 1997 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, 20-22 Aug. 1997, Vol. 1, page 457-460.

CHEN, M.Q., T. BOSE, G.F. XU. 1999. A Direction Set Based Algorithm for Adaptive Filtering. IEEE Trans. on Signal Processing. 47(2):535-539.

CILKE, J.T., D.M. ETTER. 1992. A New Adaptive Algorithm to Reduce Weight Fluctuations Caused by High Variance Data. IEEE Trans. on Signal Processing. 40(9):2324-2327.

CLARKE, D.W. 1984. Self-Tuning Control of Nonminimum-Phase Systems. Automatica. 20(5):501-517.

CLARKE, D.W., P.J. GAWTHROP. 1975. Self-Tuning Controller. IEE Proceedings. 122(9):929-934.

CLARKE, D.W., P.J. GAWTHROP. 1979. Self-Tuning Control. IEE Proceedings. 126(6):633-640.

DATTA, K.B., B.M. MOHAN. 1995. Orthogonal Functions in Systems and Control. World Scientific, Singapore. 275 p.

DING, F., T. CHEN. 2005a. Hierarchical Gradient-Based Identification of Multivariable Discrete-Time Systems. Automatica, 41:315-325.

DING, F., T. CHEN. 2005b. Hierarchical Least Squares Identification Methods for Multivariable Systems. IEEE Trans. on Automatic Control, 50(3):397-402.

DING, F., T. CHEN, L. QIU. 2006. Bias Compensation Based Recursive Least-Squares Identification Algorithm for MISO Systems. IEEE Trans. on Circuits and Systems – II, 53(5):349-353.

DING, F., T. CHEN, L. QIU. 2007. Multi-Innovation Least Squares Identification Methods Based on the Auxiliary Model for MISO Systems. Applied Mathematics and Computations, 187: 658-668.

DINIZ, P.S.R. 1997. Adaptive Filtering: Algorithms and Practical Implementation. Kluwer Academic Publishers, Boston. 443 p.

DOYLE, F.J., R.K. PEARSON, B.A. OGUNNAIKE. 2002. Identification and Control Using Volterra Models. Springer-Verlag, London. 314 p.

DUGARD, L., I.D. LANDAU. 1980. Recursive Output Error Identification Algorithms Theory and Evaluation. Automatica. 16(5):443-462.

EGARDT, B. 1979. Stability of Adaptive Controllers. Springer-Verlag, Berlin. 158 p.

EYKHOFF, P. 1974. System Identification: Parameter and State Estimation. Wiley-Interscience, London. 555 p.

FARHANG-BOROUJENY, B. 1998. Adaptive Filters: Theory and Applications. Wiley, Chichester. 529 p.

FENG, C.B., W.X. ZHENG. 1991. Robust Identification of Stochastic Linear Systems with Correlated Output Noise. IEE Proc.-D, 138(5):484-492.

FRANKLIN, G.F., J.D. POWELL, M.L. WORKMAN. 1998. Digital Control of Dynamic Systems. 3rd ed. Addison-Wesley, Menlo Park, California. 742 p.

FURUTA, K. 1993. VSS Type Self-Tuning Control. IEEE Trans. on Industrial Electronics. 40(1):37-44.

GAMBIER, A. 2004. Multivariable Adaptive State-Space Control: A Survey. Proceedings of the 5th Asian Control Conference. page 185-191.

GARNIER, H., M. MENSLER, A. RICHARD. 2003. Continuous-Time Model Identification from Sampled Data: Implementation Issues and Performance Evaluation. International Journal of Control. 76(10):1337-1357.

GARNIER, H., P. YOUNG. 2004. Time-Domain Approaches to Continuous-Time Model Identification of Dynamical Systems from Sampled Data. Proceedings of the American Control Conference. Boston, Massachusetts, 30 June – 2 July 2004, page 667-672.

GARNIER, H., L. WANG (editors). 2008. Identification of Continuous-Time Models from Sampled Data. Springer-Verlag, London. 413 p.

GAWTHROP, P.J. 1980. On the Stability and Convergence of a Self-Tuning Controller. International Journal of Control, 31(5):973-998.

GAWTHROP, P.J. 1986. Self-Tuning PID Controllers: Algorithms and Implementation. IEEE Trans. on Automatic Control, 31(3):201-209.

GAWTHROP, P.J. 1987. Continuous-Time Self-Tuning Control, Vol. I: Design. Research Study Press, Letchworth, England. 312 p.

GOLUB, G.H., C.F. VAN LOAN. 1996. Matrix Computations. 3rd ed. John Hopkins University Press, Baltimore, London. 694 p.

GOODWIN, G.C., R.L. PAYNE. 1977. Dynamic System Identification: Experiment Design and Data Analysis. Academic Press, New York. 291 p.

GOODWIN, G.C., P.J. RAMADGE, P.E. CAINES. 1980. Discrete-Time Multivariable Adaptive Control. IEEE Trans. on Automatic Control, 25(3):449-456.

GOODWIN, G.C., K.S. SIN. 1984. Adaptive Filtering, Prediction and Control. Prentice Hall, Englewood Cliffs, New Jersey. 540 p.

GUO, C., H.F. CHEN. 1991. The Aström-Wittenmark Self-Tuning Regulator Revisited and ELS-Based Adaptive Trackers. IEEE Trans. on Automatic Control. 36(7):802-812.

GUPTA, M.M., C.H. CHEN (editors). 1986. Adaptive Methods for Control System Design. IEEE Press, New York. 460 p.

HABER, R., L. KEVICZKY. 1999. Nonlinear System Identification – Input-Output Modeling Approach. Volume 1: Nonlinear System Parameter Estimation. Kluwer Academic Publishers, Dordrecht, The Netherlands. 398 p.

HARRIS, C.J., S.A. BILLINGS (editors). 1985. Self-Tuning and Adaptive Control: Theory and Applications, 2nd ed., Peter Peregrinus, London. 362 p.

HATUN, M. 2002. Sürekli Zaman Parametrelerinin Bulunması. Yüksek Lisans Tezi, Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Bursa. s.1-167.

HATUN, M., O.H. KOÇAL. 2005. Adaptif Filtrelerde Gauss-Seidel Algoritmasının Stokastik Yakınsama Analizi. Uludağ Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi. 10(2):87-92.

HATUN, M., O.H. KOÇAL. 2007a. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Transfer Fonksiyonu Parametrelerinin Yansız Tahmini. Uludağ Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi. 12(1):51-59.

HATUN, M., O.H. KOÇAL. 2007b. Tekrarlamalı Gauss-Seidel Yardımcı Değişkenler Algoritması ile Sürekli-Zaman Sistem Parametrelerinin Tahmin Edilmesi. TOK'07, Otomatik Kontrol Türk Milli Komitesi, Otomatik Kontrol Ulusal Toplantısı. Sabancı Üniversitesi, İstanbul, 5-7 Eylül 2007, sayfa 168-173.

HATUN, M., O.H. KOÇAL. 2008. Tekrarlamalı Gauss-Seidel Algoritması ile Doğrudan Model-Tabanlı Uyarlamalı Kontrol. TOK'08, Otomatik Kontrol Türk Milli Komitesi, Otomatik Kontrol Ulusal Toplantısı. İstanbul Teknik Üniversitesi, İstanbul, 13-15 Kasım 2008.

HAYKIN, S. 1991. Adaptive Filter Theory. 2nd ed. Prentice Hall, Upper Saddle River, New Jersey. 854 p.

HAYKIN, S. 2002. Adaptive Filter Theory. 4th ed. Prentice Hall, Upper Saddle River, New Jersey. 920 p.

HAYKIN, S., B. WIDROW (editors). 2003. Least Mean Square Adaptive Filters. Wiley-Interscience, Hoboken, New Jersey. 512 p.

HSIA, T.C. 1977. System Identification: Least Squares Methods. Lexington Books, Lexington, Massachusetts. 165 p.

IKONEN, E., K. NAJIM. 2002. Advanced Process Identification and Control. Marcel Dekker, New York. 310 p.

IOANNOU, P., B. FİDAN. 2006. Adaptive Control Tutorial. SIAM - Society for Industrial and Applied Mathematics, Philadelphia. 389 p.

ISERMANN, R. 1982. Parameter Adaptive Control Algorithms – A Tutorial. Automatica. 18(5):513-528.

ISERMANN, R. 1989. Digital Control Systems, Vol. 1: Fundamentals, Deterministic Control. 2nd ed. Springer-Verlag, Berlin. 334 p.

ISERMANN, R. 1991. Digital Control Systems, Vol. 2: Stochastic Control, Multivariable Control, Adaptive Control, Applications. 2nd ed. Springer-Verlag, Berlin. 325 p.

ISERMANN, R., U. BAUR, W. BAMBERGER, P.KNEPPO, H. SIEBERT. 1974. Comparison of Six On-Line Identification and Parameter Estimation Methods. Automatica. 10(1):81-103.

ISERMANN, R., K.H. LACHMAN, D. MATKO. 1992. Adaptive Control Systems. Prentice Hall, New York. 541 p.

JOHANSSON, R. 1983. Lyapunov Functions for Adaptive Systems. Proceedings of the 22nd IEEE Conference on Decision and Control. San Antonio, Texas, Dec. 1983, Vol. 22, page 449-454.

JOHANSSON, R. 1986a. Direct Adaptive Control - Global Lyapunov Stability and Exponential Convergence. Proceedings of the 25th IEEE Conference on Decision and Control. Athens, Greece, Dec. 1986, Vol. 25, page 813-818.

JOHANSSON, R. 1986b. Identification of Continuous-Time Dynamic Systems. Proceedings of the 25th IEEE Conference on Decision and Control. Athens, Greece, Dec. 1986, Vol. 25, page 1653-1658.

JOHANSSON, R. 1987. Parametric Models of Linear Multivariable Systems for Adaptive Control. IEEE Trans. on Automatic Control. 32(4):303-313.

JOHANSSON, R. 1989. Global Lyapunov Stability and Exponential Convergence of Direct Adaptive Control. International Journal of Control. 50(3):859-869.

JOHANSSON, R. 1993. System Modeling and Identification. Prentice Hall, Englewood Cliffs, New Jersey. 512 p.

JOHANSSON, R. 1994. Identification of Continuous-Time Models. IEEE Trans. on Signal Processing, 42(4):887-897.

JOHNSON, C.R. 1982. The Common Parameter Estimation Basis of Adaptive Filtering, Identification, and Control. IEEE Trans. on Acoustics, Speech, and Signal Processing. 30(4):587-594.

JOHNSON, C.R. 1988. Lectures on Adaptive Parameter Estimation. Prentice-Hall, Englewood Cliffs, New Jersey. 185 p.

KAJIKAWA, Y. 2000. The Adaptive Volterra Filter: Its Present and Future. Electronics and Communications in Japan, 83(12):51-60.

KALOUPTSIDIS, N., S. THEODORIDIS (editors). 1993. Adaptive System Identification and Signal Processing Algorithms. Prentice Hall, New York. 560 p.

KOÇAL, O.H. 1997. Uyarlamalı Süzgeçler İçin Yeni Bir Stokastik Kontrol Algoritması ve Sistem Tanılama Uygulaması. Doktora Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul. s.1-79.

KOÇAL, O.H. 1998a. A New Approach to Least Squares Adaptive Filtering. ISCAS'98, Proceedings of the 1998 IEEE International Symposium on Circuits and Systems. Monterey, California, 31 May–3 June 1998, Vol. 5, page 261-264.

KOÇAL, O.H. 1998b. A New Stochastic Algorithm for System Identification. Proceedings of the 17th IASTED International Conference on Modelling, Identification and Control. Grindelwald, Switzerland, 18-20 Feb. 1998, page 430-433.

KOÇAL, O.H., F. ÇALIŞKAN. 1998. A New Adaptive Filtering Algorithm for Fault Detection in Signals. Proceedings of the 17th IASTED International Conference on Modelling, Identification and Control. Grindelwald, Switzerland, 18-20 Feb. 1998, page 394-396.

KUBALCIK, M., V. BOBAL. 2006. Adaptive Control of Coupled-Drives Apparatus Based on Polynomial Theory. Proc. IMechE Part I: Systems and Control Engineering. 220(7):641-654.

KUO, B.C., F. GOLNARAGHI. 2003. Automatic Control Systems. 8th ed. John Wiley & Sons, New York. 850 p.

KURZ, H, R. ISERMANN, R. SCHUMANN. 1980. Experimental Comparison of Various Parameter-Adaptive Control Algorithms. Automatica. 16(2):117-133.

LAI, T.L., C.Z. WEI. 1986. Extended Least Squares and their Applications to Adaptive Control and Prediction in Linear Systems. IEEE Trans. on Automatic Control, 31(10):898-906.

LANDAU, I.D. 1976. Unbiased Recursive Identification Using Model Reference Adaptive Techniques. IEEE Trans. on Automatic Control. 21(2):194-202.

LANDAU, I.D. 1979. Adaptive Control: The Model Reference Approach. Marcel Dekker, New York, Basel. 406 p.

LANDAU, I.D. 1990. System Identification and Control Design. Prentice Hall, Englewood Cliffs, New Jersey. 272 p.

LANDAU, I.D., R. LOZANO, M. M'SAAD. 1998. Adaptive Control. Springer-Verlag, London. 562 p.

LANDAU, I.D., G. ZITO. 2006. Digital Control Systems: Design, Identification and Implementation. Springer-Verlag, London. 484 p.

LI, C., X. LIAO, J. YU. 2002. A Fast Complex-Valued Adaptive Filtering Algorithm. International Journal of General Systems. 31(4):395-403.

LJUNG, L., T. SÖDERSTRÖM. 1983. Theory and Practice of Recursive Identification. MIT Press, Cambridge, Massachusetts. 529 p.

LJUNG, L. 1999. System Identification: Theory for the User. 2nd ed. Prentice Hall, Upper Saddle River, New Jersey. 609 p.

LJUNG, L. 2008. System Identification Toolbox User's Guide. Version 7.2. The MathWorks, Inc. 528 p.

MABEY, G.W., J. GUNTHER, T. BOSE. 2004. A Euclidean Direction Based Algorithm for Blind Source Separation Using a Natural Gradient. ICASSP'04, Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 17-21 May 2004, Vol. 5, page 561-564.

MATHEW, A.V., F.W. FAIRMAN. 1974. Transfer Function Matrix Identification. IEEE Trans. on Circuits and Systems. 21(5):584-588.

MATHEWS, V.J. 1991. Adaptive Polynomial Filters. IEEE Signal Processing Magazine, 8(3):10-26.

MATHEWS, V.J., G.L. SICURANZA. 2000. Polynomial Signal Processing. Wiley-Interscience, New York. 452 p.

MATHURASAI, T., T. BOSE, D.M. ETTER. 1999. Decision Feedback Equalization using an Euclidean Direction Based Adaptive Algorithm. Conference Record of the Thirty-Third Asilomar Conference on Signals, Systems and Computers. 24-27 Oct. 1999, Vol. 1, page 519-523.

MENDEL, J.M. 1973. Discrete Techniques of Parameter Estimation. Marcel Dekker, New York. 408 p.

MIKLES, J., M. FIKAR. 2007. Process Modelling, Identification, and Control. Springer-Verlag, Berlin. 480 p.

MOUDGALYA, K.M. 2007. Digital Control. John Wiley & Sons, Chichester. 543 p.

NARENDRA, K.S., A.M. ANNASWAMY. 1989. Stable Adaptive Systems. Prentice Hall, Englewood Cliffs, New Jersey, 494 p.

NASSIRI-TOUSSI, K., W. REN. 1997. Indirect Adaptive Pole-Placement Control of MIMO Stochastic Systems: Self-Tuning Results. IEEE Trans. on Automatic Control. 42(1):38-52.

NETTO, S.L., P.S.R. DINIZ, P. AGATHOKLIS. 1995. Adaptive IIR Filtering Algorithms for System Identification: A General Framework. IEEE Trans. on Education. 38(1):54-66.

NG, T.M., B. FARHANG-BOROUJENY, H.K. GARG. 2003. An Accelerated Gauss-Seidel Method for Inverse Modeling. Signal Processing, 83:517-529.

NORTON, J.P. 1986. An Introduction to Identification. Academic Press, London. 310p.

OGATA, K. 1995. Discrete-Time Control Systems. 2nd ed. Prentice Hall, Englewood Cliffs, New Jersey, 745 p.

OGUNFUNMI, T. 2007. Adaptive Nonlinear System Identification: The Volterra and Wiener Model Approaches. Springer, New York. 229 p.

ORTEGA, K., R.KELLY. 1984. PID Self-Tuners: Some Theoretical and Practical Aspects. IEEE Trans. On Industrial Electronics, 31(4):332-338.

PATETE, A., K. FURUTA, M. TOMIZUKA. 2006. Self-Tuning of Repetitive Controllers Based on Generalized Minimum Variance Criterion. Proceedings of the American Control Conference, Minneapolis, Minnesota, USA, 14-16 June 2006, page 5468-5473.

PATETE, A., K. FURUTA, M. TOMIZUKA. 2007. Self-Tuning Control of Time-Varying Systems Based on Generalized Minimum Variance Criterion. Proceedings of SICE Annual Conference 2007, Kagawa University, Japan, 17-20 Sept. 2007, page 2563-2568.

PATETE, A., K. FURUTA, M. TOMIZUKA. 2008a. Self-Tuning Control Based on Generalized Minimum Variance Criterion for Auto-Regressive Models. Automatica. 44(8):1970-1975.

PATETE, A., K. FURUTA, M. TOMIZUKA. 2008b. Stability of Self-Tuning Control Based on Lyapunov Function. International Journal of Adaptive Control and Signal Processing. 22(8):795-810.

PINTELON, R., J. SCHOUKENS. 2001. System Identification: A Frequency Domain Approach. IEEE Press, New York. 605 p.

RAO, G.P., H. UNBEHAUEN. 2006. Identification of Continuous-Time Systems. IEE Proceedings - Control Theory and Applications, 153(2):185-220.

RAOL, R.J., G. GIRIJA, J. SINGH. 2004. Modelling and Parameter Estimation of Dynamic Systems. IEE - The Institution of Electrical Engineers, London. 388 p.

REGALIA, P. 1995. Adaptive IIR Filtering in Signal Processing and Control. Marcel Dekker, New York. 678 p.

ROCHA, K., T. BOSE, M. LARSEN. 2002. A Multiplier-Free Adaptive Algorithm for Channel Equalization. Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers. 3-6 Nov. 2002, Vol. 1, page 82-86.

SAGARA, S., K. WADA. 1977. On-line Modified Least-Squares Parameter Estimation on Linear Discrete Dynamic Systems. International Journal of Control, 25(3):329-343.

SAGARA, S., Z.Y. ZHAO. 1989. Recursive Identification of Transfer Function Matrix in Continuous Systems via Linear Integral Filter. International Journal of Control, 50(2):457-477.

SAGARA, S., Z.Y. ZHAO. 1990. Numerical Integration Approach to On-Line Identification of Continuous-Time Systems. Automatica, 26(1):63-74.

SAGARA, S., Z.Y. ZHAO. 1991. Application of Digital Filtering Techniques. In: N.K.Sinha, G.P.Rao (Editors), Identification of Continuous-Time Systems: Methodology and Computer Implementation, Kluwer Academic Publishers, Dordrecht, The Netherlands. p.291-325.

SANCHEZ-PENA, R.S., J.Q. CASIN, V.P. CAYUELA (editors). 2007. Identification and Control – The Gap between Theory and Practice. Springer-Verlag, London. 330 p.

SARIDIS, G.N. 1974. Comparison of Six On-Line Identification Algorithms. Automatica. 10(1):69-79.

SCHETZEN, M. 1980. The Volterra and Wiener Theories of Nonlinear Systems. Wiley, New York.

SEBORG, D.E., T.F. EDGAR, S.L. SHAH. 1986. Adaptive Control Strategies for Process Control: A Survey. AIChE Journal. 32(6):881-913.

SICURANZA, G.L. 1992. Quadratic Filters for Signal Processing. Proceedings of the IEEE, 80(8):1263-1285.

SINHA, N.K. 1975. Critical Evaluation of Online Identification Methods. IEE Proceedings D – Control Theory and Applications. 122(10):1153-1158.

SINHA, N.K., B. KUSZTA. 1983. Modeling and Identification of Dynamic Systems. Van Nostrand and Rainhold, New York. 334 p.

SINHA, N.K., Y.H. KWONG. 1979. Recursive Estimation of the Parameters of Linear Multivariable Systems. Automatica. 15(4):471-475.

SINHA, N.K., G.P. RAO (editors). 1991. Identification of Continuous-Time Systems: Methodology and Computer Implementation. Kluwer Academic Publishers, Dordrecht, The Netherlands. 637 p.

SOLO, V. 1979. The Convergence of AML. IEEE Trans. on Automatic Control, 24(6):958-962.

SÖDERSTRÖM, T., L. LJUNG, I. GUSTAVSSON. 1978. A Theoretical Analysis of Recursive Identification Methods. Automatica. 14(3):231-244.

SÖDERSTRÖM, T., P. STOICA. 1981. Comparison of Some Instrumental Variable Methods – Consistency and Accuracy Aspects. Automatica. 17(1):101-115.

SÖDERSTRÖM, T., P. STOICA. 1983. Instrumental Variable Methods for System Identification. Springer-Verlag, Berlin. 243 p.

SÖDERSTRÖM, T., P. STOICA. 1989. System Identification. Prentice Hall, New York. 612 p.

SÖDERSTRÖM, T., P. STOICA. 2002. Instrumental Variable Methods for System Identification. Circuits, Systems, and Signal Processing, 22(1):1-9.

STOICA, P., T. SÖDERSTRÖM, V. SIMONYTE. 1995. Study of a Bias-Free Least Squares Parameter Estimator. IEE Proceedings - Control Theory and Applications, 142(1):1-6.

STREJC, V. 1980. Least Squares Parameter Estimation. Automatica. 16(5):535-550.

TAN, L., J. JIANG. 2001. Adaptive Volterra Filters for Active Noise Control of Nonlinear Noise Processes. IEEE Trans. on Signal Processing, 49(8):1667-1676.

TAO, G. 2003. Adaptive Control Design and Analysis. Wiley-Interscience, Hoboken, New Jersey. 618 p.

TREICHLER, J.R, J.C. RICHARD, M.G. LARIMORE. 1987. Theory and Design of Adaptive Filters. Wiley-Interscience, New York. 342 p.

UNBEHAUEN, H., G.P. RAO. 1987. Identification of Continuous Systems. Elsevier Science, Amsterdam, The Netherlands. 378 p.

UNBEHAUEN, H., G.P. RAO. 1990. Continuous-Time Approaches to System Identification – A Survey. Automatica. 26(1):23-35.

UNBEHAUEN, H., G.P. RAO. 1998. A Review of Identification in Continuous-Time Systems. Annual Reviews in Control. 22:145-171.

WALTER, E., L. PRONZATO. 1997. Identification of Parametric Models from Experimental Data. Springer-Verlag, Berlin. 413 p.

WARWICK, K. (editor) 1988. Implementation of Self-Tuning Controllers. Peter Peregrinus, London. 311 p.

WELLSTEAD, P.E., M.B. ZARROP. 1991. Self-Tuning Systems: Control and Signal Processing. John Wiley, Chichester. 579 p.

WESTWICK, D.T., R.E. KEARNEY. 2003. Identification of Nonlinear Physiological Systems. Wiley-IEEE Press, New Jersey. 261 p.

WIDROW, B., S.D. STEARNS. 1985. Adaptive Signal Processing. Prentice Hall, Englewood Cliffs, New Jersey. 474 p.

XU, G.F., T. BOSE. 1998. Analysis of the Euclidean Direction Set Adaptive Algorithm. ICASSP'98, Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing. 12-15 May 1998, Vol. 3, page 1689-1692.

XU, G.F., T. BOSE, J. SCHROEDER. 1998. Channel Equalization Using an Euclidean Direction Search Based Adaptive Algorithm. GLOBECOM 98, Proceedings of the IEEE Global Telecommunication Conference. 8-12 Nov. 1998, Vol. 6, page 3479-3484.

XU, G.F. 1999. Fast Algorithms for Digital Filtering: Theory and Applications. Ph.D. Thesis, University of Colorado, Department of Electrical and Computer Engineering, Boulder. p.1-148.

XU, G.F., T. BOSE, J. SCHROEDER. 1999. The Euclidean Direction Search Algorithm for Adaptive Filtering. ISCAS'99, Proceedings of the 1999 IEEE International Symposium on Circuits and Systems. 30 May-2 June 1999, Vol. 3, page 146-149.

XU, G.F., T. BOSE, W. KOBER. 1999. A Fast Adaptive Algorithm for Image Restoration. IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications. 46(1):216-220.

YOUNG, D.M. 1971. Iterative Solution of Large Linear Systems. Academic Press, New York. 570 p.

YOUNG, P. 1981. Parameter Estimation for Continuous-Time Models – A Survey. Automatica. 17(1):23-39.

YOUNG, P. 1984. Recursive Estimation and Time-Series Analysis: An Introduction. Springer-Verlag, Berlin. 300 p.

YU, W.S. 2005. An Indirect Adaptive Pole-Placement Control for MIMO Discrete-Time Stochastic Systems. International Journal of Adaptive Control and Signal Processing, 19(7): 547-573.

ZAKNICH, A. 2005. Principles of Adaptive Filters and Self-Learning Systems. Springer-Verlag, London. 386 p.

ZHANG, Y., T.T. LIE, C.B. SOH. 1997. Consistent Parameter Estimation of Systems Disturbed by Correlated Noise. IEE Proceedings - Control Theory and Applications, 144(1):40-44.

ZHANG, Z., T. BOSE, J. GUNTHER. 2004a. Transient Analysis of the Euclidean Direction Search (EDS) Algorithm. Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers. 7-10 Nov. 2004, Vol. 2, page 1554-1558.

ZHANG, Z., T. BOSE, J. GUNTHER. 2004b. Optimal Order EDS and FEDS Algorithms. Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers. 7-10 Nov. 2004, Vol. 2, page 1559-1563.

ZHANG, Z., T. BOSE, J. GUNTHER. 2005. A Unified Framework for Least Square and Mean Square Based Adaptive Filtering. ISCAS 2005, Proceedings of the 2005 IEEE International Symposium on Circuits and Systems. 23-26 May 2005, Vol. 5, page 4325-4328.

ZHANG, Z., T. BOSE, L. XIAO, R. THAMVICHAI. 2006. Performance Analysis of the Deficient Length EDS Adaptive Algorithm. APCCAS 2006, IEEE Asia Pacific Conference on Circuits and Systems. 4-7 Dec. 2006, page 222-226.

ZHANG, Z., T. BOSE, M.S. RADENKOVIC. 2008. Fixed-Point Analysis of Adaptive Filters Based on the EDS Algorithm. ISCAS 2008, Proceedings of the 2008 IEEE International Symposium on Circuits and Systems. 18-21 May 2008, page 57-60.

ZHAO, C., S.S. ABEYSEKARA. 2005. Investigation of Euclidean Direction Search Based Laguerre Adaptive Filter for System Identification. Proceedings of the International Conference on Communications, Circuits and Systems. 27-30 May 2005, Vol. 2, page 691-695.

ZHENG, W.X. 1998. On a Least-Squares-Based Algorithm for Identification of Stochastic Linear Systems. IEEE Trans. on Signal Processing, 46(6):1631-1638.

ZHENG, W.X. 1999. Least-Squares Identification of a Class of Multivariable Systems with Correlated Disturbances. Journal of the Franklin Institute, 336(8):1309-1324.

ZHENG, W.X. 2000. Unbiased Identification of Stochastic Linear Systems Subject to Coloured Noise. IEE Proceedings - Control Theory and Applications, 147(5):485-491.

ZHU, Y., T. BACKS. 1993. Identification of Multivariable Industrial Processes for Simulation, Diagnosis and Control. Springer-Verlag, London. 187 p.

ZHU, Y. 2001. Multivariable System Identification for Process Control. Elsevier Science, Oxford. 349 p.

ÖZGEÇMİŞ

1977 yılında Bursa'nın Karacabey ilçesinde doğdu. İlk ve ortaokulu burada tamamladı. Lise eğitimini Bursa Erkek Lisesi'nde tamamladıktan sonra 1994 yılında Uludağ Üniversitesi Elektronik Mühendisliği Bölümü'nde lisans eğitimine başladı ve 1998 yılında mezun oldu. 1999 yılında Uludağ Üniversitesi Fen Bilimleri Enstitüsü Elektronik Mühendisliği Anabilim Dalı'nda yüksek lisans eğitimine başladı ve 2002 yılında yüksek lisans eğitimini tamamladı. Aynı yıl Uludağ Üniversitesi Fen Bilimleri Enstitüsü Elektronik Mühendisliği Anabilim Dalı'nda doktora eğitimine başladı. 2000 yılından beri Uludağ Üniversitesi Elektronik Mühendisliği Bölümü'nde araştırma görevlisi olarak çalışmaktadır ve halen bu görevine devam etmektedir.

TEŞEKKÜR

Doktora tez aşamasında her konuda yardımcı olan tez danışmanım değerli hocam Yrd. Doç. Dr. Osman Hilmi KOÇAL'a, tez izleme komitesinde bulunan ve tezime önerileriyle katkıda bulunan değerli hocalarım Prof. Dr. Erdoğan DİLAVEROĞLU'na ve Prof. Dr. İrfan KARAGÖZ'e, lisans, yüksek lisans eğitimim boyunca ve doktora eğitimimin bir kısmında bana yardımcı olarak otomatik kontrol konusunda yetişmemde çok büyük emekleri olan değerli hocam Dr. Murat TÜRE'ye ve doktora eğitimin boyunca bana destek olan aileme ve tüm araştırma görevlisi arkadaşlarıma çok teşekkür ederim.