

DİNAMİK SİSTEM MODELLEME İÇİN YENİ BİR GERİ BESLEMELİ SİNİR AĞI YAKLAŞIMI

Ramazan ÇOBAN¹

Damla AVAN²

Özden ERÇİN³

Çukurova Üniversitesi, Bilgisayar Mühendisliği Bölümü, Sarıçam, Adana

¹e-mail:rcoban@cu.edu.tr

²e-mail:damlaavan@yahoo.com

³e-mail:ozdenercin@yahoo.com

Özetçe

Yapay sinir ağları dinamik sistemlerin modellenmesi için kullanıldığından beri, mühendislikte aktif bir araştırma konusu haline gelmiştir. Bu çalışmada dinamik sistemlerin modellenmesi için yeni bir geri beslemeli ağı önerilmektedir. Farklı yapay sinir ağı modelleri mukayese edilmekte ve eksik yönleri belirtilmektedir. Önerilen geri beslemeli ağı mimarisi, öğrenme metodolojisi ve etkili yönleri açıklanmakta ve simülasyon sonuçları verilmektedir.

Anahtar sözcükler: Yapay sinir ağları, dinamik sistemler, modelleme.

1. Giriş

Genellikle dinamik sistem modelleme için ileri beslemeli ve geri beslemeli ağlar kullanılmaktadır. İleri beslemeli ağlarda dinamik bellek olmadığından, verimli bir dinamik sistem modelleme için genellikle tapped-delay-line (TDL) metodu kullanılmaktadır [1]. Bu metodda giriş olarak ağa şimdiki ve önceki giriş-çıkış bilgileri verilmektedir. Sistemin bir sonraki çıkışı da öğrenme safhasında giriş işareti olarak kullanılmaktadır. Fakat bu metodun bazı dezavantajları vardır. Bunlardan biri eğitim ve test yapılarının farklı olmasıdır. Şöyle ki, ağı eğitim safhasında iyi eğitilmiş gibi görünmekte fakat test aşamasında kötü performans sergilemektedir [2].

Geri beslemeli ağlar yapılarından dolayı yukarıdaki güçlüklerle karşılaşmamaktadır. Fakat Parlos tipi [3], Elman tipi [4] ve Jordan tipi [5] geri beslemeli sinir ağları dinamik sistem modellemede eğitim ve test aşamasında başarılı olsalar bile, yapıları dolayısıyla istenen herhangi bir başlangıç koşulundan başlamaları mümkün değildir, çünkü gizli katmanlarında sabit başlangıç koşulları içermektedirler [2]. Fakat bazı dinamik sistem modelleme problemlerinde farklı başlangıç koşullarına ihtiyaç duyulabilmektedir.

Mevcut geri beslemeli ağlar arasında, Elman ağı standart geri-yayımlı (back-propagation) algoritmasıyla eğitilebilecek en basit tiplerden biridir. Pham ve Liu,

dinamik sistemleri modellemek için Elman ağını modifiye etmişlerdir [6]. Modifiye Elman sinir ağının da bir takım dezavantajları vardır. Bunlardan biri bazen yakınsama sağlamamasıdır. Bir diğeri ise her istenen başlangıç koşulunda çalışmasının mümkün olmamasıdır.

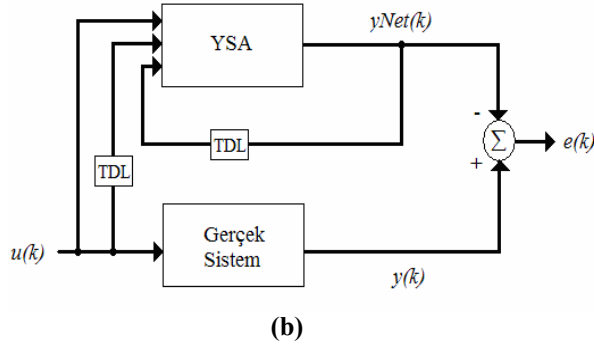
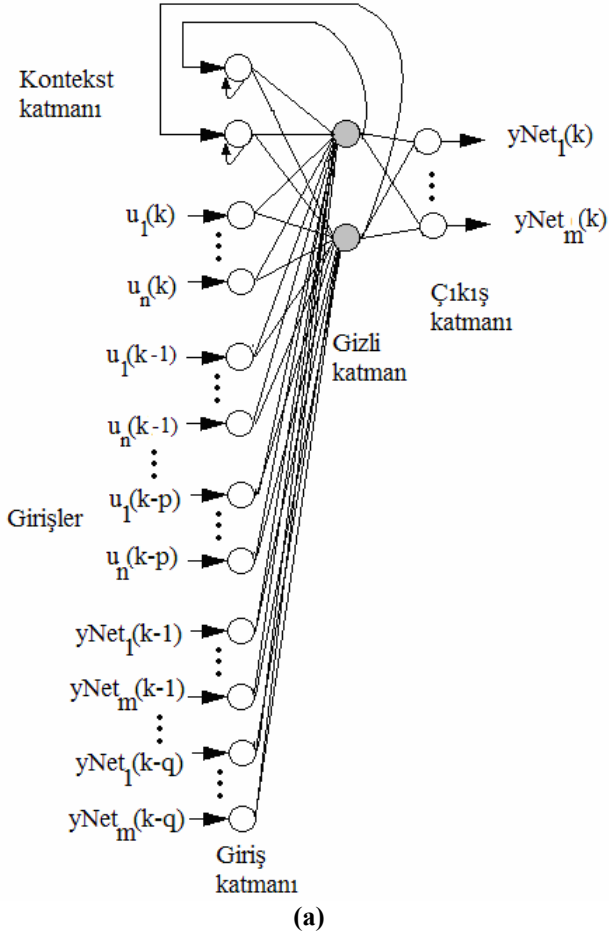
Bu çalışmada, yukarıda bahsedilen sınırlamalardan kurtulmak ve öğrenme kapasitesini arttırmak için modifiye Elman ağının [6] iyileştirilmiş bir versiyonu olan yeni bir geri beslemeli sinir ağı önerilmiştir. Önerilen Elman sinir ağının modeli ve öğrenme metodolojisi Bölüm 2’de verilmektedir. Makalenin geri kalan kısmı şu şekilde organize edilmiştir: Bölüm 3’te modifiye Elman ve önerilen Elman ağı iki farklı dinamik sistem modeli kullanılarak karşılaştırılmakta ve simülasyon sonuçları sunulmaktadır. Son bölümde, sonuç ve tartışmalara yer verilmiştir.

2. Önerilen Geri Beslemeli Sinir Ağı

Modifiye Elman sinir ağı, bir giriş, bir çıkış, bir gizli katman ve gizli katman ile aynı sayıda sinir hücresi içeren kontekst (context) katmanından oluşmaktadır. Bir anlamda, kontekst katmanının giriş katmanına dahil olduğu söylenebilir. Gizli katmandan kontekst katmanına geriye doğru bir bağlantı mevcuttur. Fakat giriş katmanında eski girişler değil sadece güncel girişler vardır.

Bu çalışmada önerilen geri beslemeli yapay sinir ağı, modifiye Elman ağının geliştirilmiş bir versiyonudur [6]. Önerilen Elman sinir ağının giriş katmanına, giriş ve çıkışların geçmiş değerlerini temsil eden bazı yapay sinir hücreleri eklenmiştir. Böylece giriş katmanında çıkışların geçmiş değerlerini temsil eden hücrelerin bulundurulmasıyla, istenilen başlangıç koşullarından başlamak mümkün hale gelmiştir. Önerilen Elman ağının mimarisi Şekil 1’de gösterilmektedir.

Dinamik çok giriş-çok çıkışlı bir sistemin ayrık zamandaki giriş-çıkış modeli aşağıdaki gibi tanımlanabilir [1]:



Şekil 1. Önerilen Elman ağı (a) test safhası, (b) öğrenme safhası.

$$y(k) = f(y(k-1), y(k-2), \dots, y(k-q), u(k-1), u(k-2), \dots, u(k-p)) \quad (1)$$

Burada $[u(k), y(k)]$, k zaman adımında sistemin giriş-çıkış çiftini; pozitif tamsayı olan p ve q sırasıyla geçmiş girişlerin ve geçmiş çıkışların sayısını göstermektedir. Uygulamada genellikle p , q 'dan küçük veya eşit olmaktadır. Geçmiş girişleri ve çıkışları yeni çıkışa haritalayan f fonksiyonu, doğrusal ya da doğrusal olmayan bir fonksiyon olabilir.

Önerilen Elman ağı ve bu ağı eğitmekte kullanılan standart geri yayılım algoritması, n girişli ve m çıkışlı bir sistem için aşağıda verilmiştir:

Ağın ileri yöndeki işareti:

$$X^h(k) = W^{hc} X^c(k) + W^{hu} U(k) \quad (2)$$

$$X^c(k) = X^h(k) + \alpha X^c(k-1) \quad (3)$$

$$yNet(k) = W^{yh} X^h(k) \quad (4)$$

$X^h(k)$, gizli katman çıkışı; $X^c(k)$, kontekst katman çıkışı; $yNet(k)$, ağı çıkışı; W^{hc} , gizli katman ile kontekst katmanı arasındaki bağlantı ağırlıklarını; W^{hu} , gizli katman ile giriş katmanı arasındaki bağlantı ağırlıklarını; W^{yh} , gizli katman ile çıkış katmanı arasındaki bağlantı ağırlıklarını; α , kontekst birimindeki öz bağlantıların geri besleme kazancını temsil etmektedir.

Ağın öğrenme algoritması ise:

$$\Delta W^{yh} = \eta (y(k) - yNet(k)) X^h(k) \quad (5)$$

$$\Delta W^{hu} = \eta (y(k) - yNet(k)) U(k) \quad (6)$$

$$\Delta W^{hc} = \eta (y(k) - yNet(k)) X^h(k-1) \quad (7)$$

Burada η , öğrenme katsayısını temsil etmektedir. Gizli katman çıkışından kontekst katman girişine olan geri beslemenin ağırlık katsayıları sabittir ve 1 olarak seçilmiştir.

Bu çalışmada hata kriteri olarak hataların karelerinin toplamını içeren aşağıdaki gibi bir fonksiyon kullanılmaktadır:

$$J = \frac{1}{2} \sum_k e(k)^2 \quad (8)$$

$$e(k) = y(k) - yNet(k) \quad (9)$$

Burada J , hata miktarını; $y(k)$, ideal çıkışı ve $yNet(k)$, ağı çıkışını temsil etmektedir. Bir öğrenme çevrimi için hata (root-mean-square, rms), şu şekilde hesaplanmaktadır:

$$RMS \text{ error} = \sqrt{\frac{\sum_{k=1}^r e(k)^2}{r}} \quad (10)$$

Burada r , veri sayısıdır.

Giriş/çıkış verilerini yapay sinir ağına uygulamadan önce, girişler ve çıkışlar doğrusal olarak aşağıdaki formülle normalize edilmiştir:

$$X_{norm} = \frac{V_{max} - V_{min}}{X_{max} - X_{min}} X + \frac{V_{min} X_{max} - V_{max} X_{min}}{X_{max} - X_{min}} \quad (11)$$

X_{norm} , normalize edilmiş giriş veya çıkış değerini; V_{min} ve V_{max} , sırasıyla normalizasyon aralığının minimum ve maksimum değerlerini; X_{min} and X_{max} , sırasıyla veri kümesindeki giriş ve çıkışların minimum ve maximum değerlerini göstermektedir. Bu çalışmada, V_{min} ve V_{max} sırasıyla -1,0 ve +1,0 olarak seçilmiştir.

Öğrenme aşamasında her çevrim için öğrenme katsayısı aşağıdaki kurala göre ayarlanmaktadır [7]:

$$\eta = \begin{cases} \eta + \gamma & \Delta J < 0 \\ \eta - \beta & \Delta J > 0 \\ 0 & \Delta J = 0 \end{cases} \quad (12)$$

3. Simülasyon Sonuçları

Ağların performanslarını göstermek için, modifiye Elman ve önerilen Elman ağları 2. ve 3. dereceden doğrusal sistem modelleriyle karşılaştırılmıştır. Her iki model için, öğrenme süreci 5 defa tekrarlanmış ve ağların ağırlıkları güncellenmiştir. Her koşum için rms değeri Tablo 1’de verilmiştir. Tablodan da görüldüğü üzere önerilen Elman ağı her koşum için yakınsama sağlamış fakat modifiye Elman ağı bazı koşumlar için yakınsama gösterememiştir. Test çalışması esnasında iki ağ için de en düşük rms hatasını veren ağırlık, eşik ve α değerleri kullanılmıştır.

Tablo 1. Model 1 ve Model 2 için modifiye Elman ile önerilen Elman ağlarına ait rms hataları.

Koşum	Model 1 için RMS		Model 2 için RMS	
	Modifiye Elman Ağı	Önerilen Elman Ağı	Modifiye Elman Ağı	Önerilen Elman Ağı
1	0,005170	0,002585	0,021770	0,002355
2	0,068979	0,002308	0,027521	0,002240
3	Yakınsamadı	0,002411	0,033123	0,002469
4	0,019232	0,001107	0,019440	0,001923
5	Yakınsamadı	0,001119	0,020140	0,002259

Model 1: Modellenen 2. dereceden sistemin ayrık zaman modeli aşağıdaki gibidir [2]:

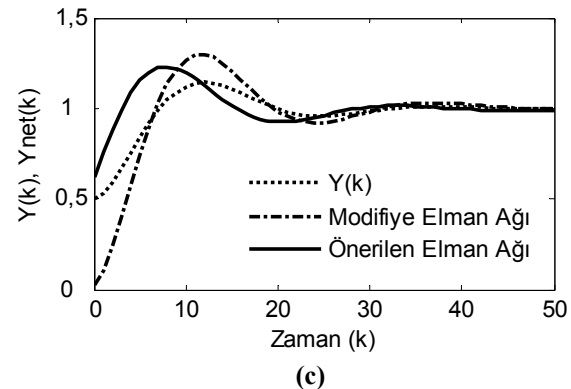
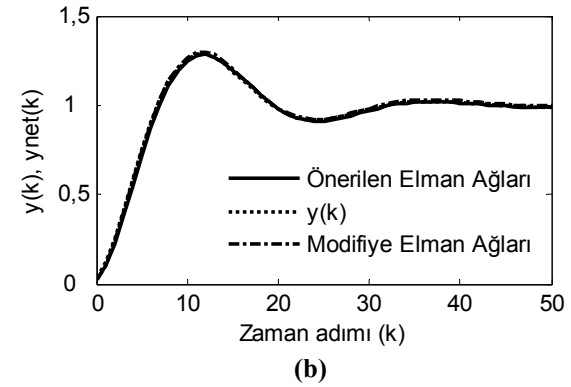
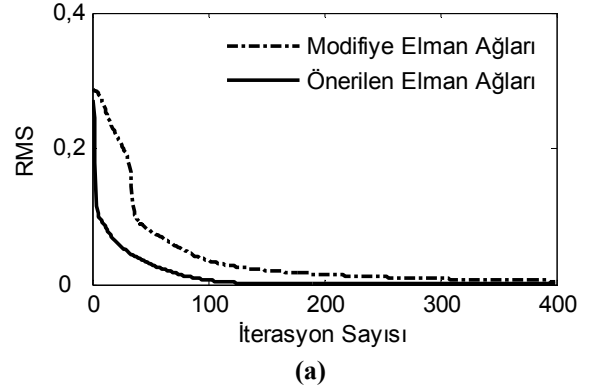
$$y(k) = A_1y(k-1) + A_2y(k-2) + B_1u(k-1) + B_2u(k-2) \quad (13)$$

Denklem (13)’deki katsayılar şu şekildedir:

$$A_1=1,752821; A_2=-0,818731; B_1=0,011689; B_2=0,010942.$$

Model 1’de verilen dinamik sistem için, ağlar (modifiye Elman ve önerilen Elman) 1 giriş sinir hücresi, 6 gizli/kontekst sinir hücresi ve 1 çıkış sinir hücresi olacak şekilde oluşturulmuştur. Tüm ağırlıklar başlangıçta [-0,1,+0,1] aralığında olacak şekilde rastgele atanmıştır. 400 veriden oluşan eğitim seti sıfır başlangıç koşulunda,

[-2,911160; 2,911160] aralığında rastgele giriş değerleri kullanılarak elde edilmiştir. Öğrenme ve momentum katsayıları 0,02’ye eşit, kontekst katmanındaki α katsayıları 0 ile 1 arasında değişen değerlere sahiptir ve eğitim çevrim (iterasyon) sayısı da 400 olarak alınmıştır. Ağı test etmek için, genliği 2,911160 olan birim basamak fonksiyonu şeklinde giriş işareti kullanılmıştır. Bu model için, her bir ağa ait rms hataları hesaplanmış ve elde edilen simülasyon sonuçları Şekil 2(a)’da gösterilmiştir. Ağların çıktıları da Şekil 2(b)’de verilmiştir.



Şekil 2. Model 1 için (a) modifiye Elman ve önerilen Elman ağı için rms hataları, (b) sıfır başlangıç koşulunda ağ çıktıları (c) 0,5 başlangıç koşulunda ağ çıktıları.

Model 2: 3. dereceden doğrusal sistem modelinin ayrık formu [2]:

$$y(k) = A_1y(k-1) + A_2y(k-2) + A_3y(k-3) + B_1u(k-1) + B_2u(k-2) + B_3u(k-3) \quad (14)$$

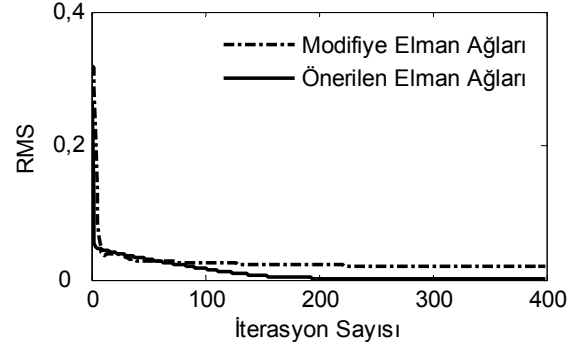
Denklem (14)'te bulunan katsayı değerleri: $A_1=2,627771$; $A_2=-2,333261$; $A_3=0,697676$; $B_1=0,017203$; $B_2=-0,030862$; $B_3=0,014086$.

Model 2'de verilen dinamik sistem için, her iki ağ da 1 giriş sinir hücresi, 6 gizli/kontekst sinir hücresi ve 1 çıkış sinir hücresi olacak şekilde oluşturulmuştur. Tüm ağırlıklar başlangıçta $[-0,1,+0,1]$ aralığında olacak şekilde rasgele atanmıştır. 400 veriden oluşan eğitim seti sıfır başlangıç koşullarında, $[-18,29367; 18,29367]$ aralığında rasgele giriş değerleri kullanılarak elde edilmiştir. Öğrenme ve momentum katsayıları 0,02, kontekst katmanındaki α katsayıları 0 ile 1 arasında değişen değerler ve eğitim çevrim sayısı da 400 olarak seçilmiştir. Ağı test etmek için, genliği 18,29367 olan birim basamak fonksiyonu şeklinde giriş işaretini kullanılmıştır. Bu model için, her bir ağa ait rms hataları hesaplanmış ve elde edilen simülasyon sonuçları Şekil 3(a)'da gösterilmiştir. Ağların çıkışları da Şekil 3(b)'de verilmiştir.

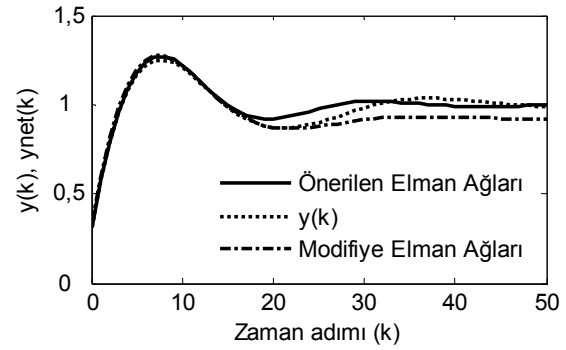
Şekillerden de gözlemleneceği üzere, önerilen Elman ağı diğer ağa göre daha iyi bir genelleme yeteneği göstermiştir. Model 1 için başlangıç koşulunun 0,5 olarak alınması durumunda istenen çıkış 0,501074 gibi bir değer almaktadır. Önerilen Elman ve modifiye Elman ağı için rms hataları sırasıyla 0,151469 ve 0,155357 olarak hesaplanmıştır. Bu sonuca göre rms hataları birbirine yakın çıkmıştır. Fakat modifiye Elman ağı 0,022017 değerinden başlarken önerilen Elman ağı 0,622129 gibi bir değerden başlamıştır. Bu da göstermektedir ki önerilen Elman ağı biraz daha eğitildiğinde rms hatası düşecek ve daha iyi bir performans sergileyecektir. Nitekim öğrenme çevrimi 400 yerine 10000 seçilmiş ve daha iyi bir netice elde edilmiştir. Fakat bu sonuç burada sunulmamıştır. Bu sonuçlar göstermektedir ki diğer ağ ile kıyaslandığında önerilen Elman ağı keyfi başlangıç koşullarında daha iyi bir performansa sahiptir. Şekil 2(a) ve 3(a)'dan da görüldüğü gibi, önerilen Elman ağının öğrenme hatası daha düşük olmaktadır.

4.Sonuçlar

Bu çalışmada, dinamik sistem modelleme için yeni bir geri beslemeli yapay sinir ağı önerilmiş ve başarımı simülasyon sonuçlarıyla ortaya konmuştur. Önerilen geri beslemeli sinir ağının modifiye Elman ağına göre öğrenme kapasitesinin daha fazla olduğu ve her istenen başlangıç koşulunda da daha iyi sonuç verdiği görülmüştür.



(a)



(b)

Şekil 3. Model 2 için (a) modifiye Elman ve önerilen Elman ağı için rms hataları, (b) sıfır başlangıç koşulunda ağ çıkışları.

5. Referanslar

- [1] K.S. Narendra, and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks", IEEE Transaction on Neural Networks. Vol.1, No.1,1990.
- [2] D.T. Pham and X. Liu, "Neural Networks for Identification, Prediction and Control", Springer-Verlag London Limited, 1997.
- [3] A.G. Parlos, K.T. Chong, A.F. Atiya, "Application of the recurrent multi-layer perceptron in modeling complex process dynamics", IEEE Transactions on Neural Networks, 5 (2), pp. 255-266, 1994.
- [4] J.L. Elman, "Finding structure in time", Cognitive Science, 14, pp. 179-211, 1990.
- [5] M.I. Jordan, "Attractor dynamics and parallelism in a connectionist sequential machines", Proceedings of the 8th Annual Conference of the Cognitive Science Society, pp. 531-546, 1986.
- [6] D.T. Pham and X. Liu, "Dynamic system modelling using partially recurrent neural networks", Journal of Systems Engineering, 2(2), pp. 90-97, 1992.
- [7] J. Hertz, A. Krogh, and R.G. Palmer, "Introduction to the theory of neural computation", Addison-Wesley, New York, 1991.