

SCARA ROBOT DİNAMİĞİNİN YAPAY SİNİR AĞLARI KULLANILARAK MODELLENMESİ

Aysun EĞRİSÖĞÜT TİRYAKİ,

Recep KAZAN *

Üretim kapasitesi ve verimliliğin artırılması, belirli standartlarda ve hassasiyette ürün elde edilmesi, iş gücü tasarrufu vb. gereksinimler nedeniyle birçok alanda değişik robotlar kullanılmaktadır. Bu çalışmada üç eklemlerli bir SCARA robotu ele alınmış ve dinamiği yapay sinir ağları (YSA) ile modellenmiştir. Robotun Lagrange-Euler dinamik denklemleri çıkarılmıştır. Modellenen SCARA robot kolu, kübik yörüngelerle ifade edilen iki dönme ve doğrusal yörüngeye sahip bir öteleme hareketi yapmaktadır. Belirlenen zaman aralığındaki sınır şartları kullanılarak kübik ve doğrusal yörüngelerin katsayıları elde edilmiş ve bu sayede konum, hız ve ivme denklemlerine ulaşılmıştır. Verilen zaman aralığındaki konum, hız ve ivme değerleri YSA'na giriş, tork değerleri ise çıkış olarak verilmiştir. Böylece YSA'nın dokuz giriş, üç çıkış nöronu bulunmaktadır. Bu çalışmada çok katmanlı ileri beslemeli geri yayılım ağı ve transfer fonksiyonu olarak da logaritmik sigmoid fonksiyonu kullanılmıştır. Geri yayılım ağı, eğitim fonksiyonu olarak Levenberg-Marquardt optimizasyon yöntemine göre ağırlık ve bias değerlerini güncellemiştir. Hedeflenen MSE (ortalama hataların karesi) değerine ulaşılarak YSA'nın eğitimi tamamlanmış ve çevrim sayısına göre MSE değerlerinin görülebildiği performans grafiği verilmiştir.

Anahtar sözcükler: SCARA robot, Robot dinamiği, Yapay sinir ağları.

Robot manipulators are used in various fields to enhance the production capacity and efficiency to produce products in particular standards and to law disposal the manpower for different requirements. In this study three jointed SCARA robot is considered and modeled by using artificial neural network (ANN). Lagrange-Euler dynamic equations of robot are obtained. The modeled SCARA robot manipulator has two rotary and one linear motion. The rotary motion is defined by cubic trajectory and the linear motion has linear trajectory. The cubic and linear trajectory coefficients are acquired by using the boundary conditions in a particular time period and thus position, velocity and acceleration equations are also carried out. Position, velocity and acceleration values are introduced to ANN as input and torque values are obtained as output in a particular time period. In this manner ANN has nine input and three output neurons. In this study, a multi-layer feed forward backpropagation network and logarithmic sigmoid function as a transfer function are used. Backpropagation network updates weight and bias values by considering train function as Levenberg-Marquardt optimization method. By attaining the proposed MSE values, ANN's train is completed and performance graphic is obtained. This graphics show MSE values connected with epochs.

Keywords: Scara, robot, Robot dynamics, Artificial neural network.

* Sakarya Üniversitesi, Mühendislik Fakültesi, Makina Mühendisliği Bölümü

GİRİŞ

İnsan beynindeki çalışma sisteminin yapay olarak benzetimi çalışmalarının sonucu ortaya çıkan yapay sinir ağları (YSA), klasik tekniklerle çözümü zor, karmaşık ve doğrusal olmayan problemlerin çözümünde etkin bir yoldur. YSA'nın pratik kullanımı genellikle, çok farklı yapıda ve formlarda bulunabilen verileri hızlı bir şekilde tanımlama ve algılama üzerinedir [1].

YSA'ların doğrusal olmayan haritalama veya fonksiyon yakınsama özelliği, robot dinamik çözümleri için oldukça uygundur. SCARA robotunun da içinde bulunduğu farklı robot tipleri üzerinde YSA'lar çok kez uygulanmıştır.

SCARA tipi robotlar insan kolunu model alan, özellikle kesme, yükleme, montaj ve üretim hatları gibi sürekliliği gerektiren ortamlarda kullanılan küçük endüstriyel robotlardır. SCARA sözcüğü dilimize 'Seçici Serbest Esnemeli Robot Kolu' (Selectively Compliant Articulated Robot Arm) olarak çevrilmiştir [2]. 1970'den sonra Japon Endüstriyel Konsorsiyumu ve bir grup araştırmacı tarafından Japonya'da Yamanashi Üniversitesi'nde geliştirilmiş olup, montaj hatlarında alıp-yerleştirme, kaynak, boya gibi işlemleri yapması amaçlanmıştır. Oldukça yüksek hıza ve en iyi tekrarlamaya kabiliyetine sahip olan SCARA robot hem dönel hem de silindirik koordinatlara karşılık gelen yatay düzlemde çalışır, buna karşın düşey düzlemde katıdır, esnemez.

YSA'nın robotlarda kullanımı ile ilgili farklı içerikte çalışmalar gözlenmiştir. Guez ve Ahmad [3] robotlarda kinematik ve ters kinematik hesaplanması için bir sinir ağı kullanmıştır. Miyamoto ve ark. [4] bir endüstriyel robotun yörünge kontrolü için ters dinamik modelini oluşturma ve geri besleme döngüsünde, Kawato ve ark. [5] da dinamik kontrol için bir kontrolör olarak yapay sinir ağlarını kullanmıştır. Chan yapay sinir ağlarını belirsizlik ve nonlineerlik için bir dengeleyici ve tanımlama için bir araç olarak kullanmıştır. Omedei ve ark. [6] endüstriyel robotların tanımlanmasını içeren farklı algoritmalar üzerinde çalışıp, SCARA IBM 7535 robotta denemiştir. Köker ve ark.

[7] üç eklemlı bir robot için ters kinematik çözüme dayanan yapay sinir ağı tasarlamıştır. Daş ve ark. [8] SCARA robotu matematiksel olarak modellemiş, simülasyon yapmış ve deneysel olarak doğrulamıştır. Bu konudaki diğer uygulamalar ise; Wilhelmsen ve Cotter [9], üç farklı sinir ağı yapısı kullanılarak lineer olmayan robot kontrolü, Kosko [10] hiyerarşik YSA'ları ile ters Jacobian kontrolü, Rabelo ve Avula [11], Fukuda ve ark. [12], Liu ve ark. [13], Karakasoğlu ve Sundareshan [14] çok dereceli robot elin kontrolü ve tasarımı, Smagt ve Krose [15] sensör bilgilerinin ters dinamiğe direk geçişinin sağlanmasında, Tsutsumi ve Matsumoto [16] robot pozisyon kontrolü üzerine çalışmışlardır.

Bu çalışmada ise üç eklemlı bir SCARA robotu ele alınmış ve robot dinamiği YSA'nın kullanımı ile modellenmiştir. Çalışmanın 2. kısmında üç eklemlı SCARA tipi robot dinamiği ve 3. kısmında YSA genel hatlarıyla tanıtılmıştır. 4. kısımda tasarlanan YSA mimarisi, ağırlık eğitimi performansı ve test sonuçları verilmiştir. 5. kısımda ise simülasyon sonuçları yorumlanmıştır.

ÜÇ EKLEMLİ SCARA TİPİ ROBOT DİNAMİĞİ

Dinamik Model

Bir robot kolun dinamik modeli, robot kolun dinamik davranışlarını belirleyen hareket denklemlerinden oluşur. Bu denklemlerin bilgisayar yardımı ile çözümü robot ve robot kontrol tasarımı açısından büyük kolaylık sağlar. Robot kolunun dinamik analizi, eklemlere tahrik elemanları tarafından uygulanan moment veya kuvvet büyüklükleri ile robot kolunun zamana göre konumu, hızı ve ivmesi arasındaki ilişkilerin incelenmesi demektir. Böyle bir analiz için, robot kolun dinamik davranışlarını tanımlayan lineer olmayan diferansiyel denklem takımlarının elde edilmesi ve çözülmesi gerekmektedir.

Dinamik problem düz dinamik (forward dynamics) ve ters dinamik (inverse dynamics) olmak üzere ikiye ayrılmaktadır. Düz dinamik problem, herhangi bir t anında istenen kuvvet ve momentler verildiğinde robot kolunun

alacağı konum ve yönlenme için gerekli olan genelleştirilmiş koordinatları, eklem hız ve ivmelerini bulma problemidir. Ters dinamik problem ise, robot kolunun istenen bir konum ve yönlenmeye gelmesi için (genelleştirilmiş koordinatlar, bunların birinci (hız) ve ikinci (ivme) türevleri verildiğinde) gerekli olan kuvvet ve momentlerin hesaplanmasıdır [17].

Robot kolunun dinamik denklemlerini elde etmek için bilinen birçok metot vardır. Bunlar Lagrange-Euler (L-E), Recursive Lagrange (R-L), Newton-Euler (N-E), Genelleştirilmiş D'Alambert (G-D) prensibi gibi yaklaşımlardır. Bu yöntemlerden en çok kullanılanları L-E ve N-E yaklaşımlarıdır.

Bu çalışmada robot kolun dinamik modelinin çıkarılmasında, iyi planlanmış bir yapıya sahip ve çıkarılması N-E yaklaşımından daha basit ve sistematik olan L-E yaklaşımı ile elde edilmiş denklemler kullanılacaktır. Bu yaklaşım sistemin içerdiği toplam iş ve enerji ile ifade edilir. Lagrange-Euler eşitliğinin genel ifadesi:

$$\frac{d}{dt} \left[\frac{\partial L}{\partial \dot{q}} \right] - \left[\frac{\partial L}{\partial q} \right] = \tau_i \quad i = 1, 2, \dots, n \quad (1)$$

$$L = K - P \quad (2)$$

denklemleri ile ifade edilir. Burada;

L = Lagrange fonksiyonu

K = Robot kolunun toplam kinetik enerjisi

$$K = \frac{1}{2} \sum_{i=1}^3 m_i v_i^2 + \frac{1}{2} \sum_{i=1}^3 I_i \dot{\theta}_i^2 \quad (3)$$

P = Robot kolunun toplam potansiyel enerjisi

$$P = m_i g d_i \quad (4)$$

qi = Robot kolunun genelleştirilmiş koordinatları (Döner eklemlerde θ_i ve kayar eklemlerde d_i olarak alınır.)

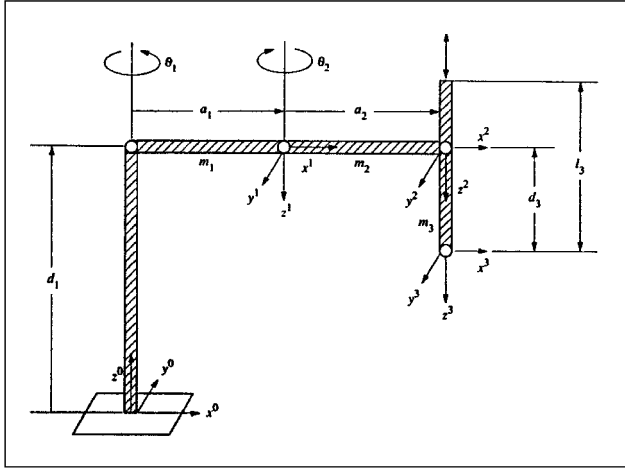
τ_i = Sisteme uygulanan genelleştirilmiş moment veya kuvvet

Dinamik denklemler matris formunda;

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) + F_v(\dot{\theta}) + F_c = \tau \quad (5)$$

şeklinde yazılabilir. Burada; $M(\theta)$ manipulatörün $n \times n$ boyutlu eylemsizlik matrisi, $C(\theta, \dot{\theta})$ $n \times 1$ boyutlu merkezkaç ve Coriolis vektörü, $G(\theta)$ $n \times 1$ boyutlu yerçekimi kuvveti vektörü, $F_v(\dot{\theta})$ $n \times 1$ boyutlu viskoz sürtünme vektörü, F_c $n \times 1$ Coulomb terimleri vektörü ve τ $n \times 1$ boyutlu ($i = 1, 2, \dots, n$) eklemlerine uygulanan genelleştirilmiş kuvvet veya moment vektörü [18-20].

Bu çalışmada kullanılan SCARA robot Şekil 1'de gösterilmiş ve robotun Lagrange-Euler dinamik denklemleri 6, 7 ve 8 nolu denklemlerdeki gibi elde edilmiştir;



Şekil 1. Üç Eklemlili SCARA Robot

$$\begin{aligned} \tau_1 = & \left[\left(\frac{m_1}{3} + m_2 + m_3 \right) a_1^2 + (m_2 + 2m_3) \right. \\ & a_1 a_2 C_2 + \left. \left(\frac{m_2}{3} + m_3 \right) a_2^2 \right] \ddot{\theta}_1 \\ & - \left[\left(\frac{m_2}{3} + m_3 \right) a_1 a_2 C_2 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \ddot{\theta}_2 \right] \\ & - a_1 a_2 S_2 \left[(m_2 + 2m_3) \dot{\theta}_1 \dot{\theta}_2 - \left(\frac{m_2}{3} + m_3 \right) \dot{\theta}_2^2 \right] \end{aligned} \quad (6)$$

$$\begin{aligned} \tau_2 = & - \left[\left(\frac{m_2}{3} + m_3 \right) a_1 a_2 C_2 + \left(\frac{m_2}{3} + m_3 \right) a_2^2 \right] \ddot{\theta}_1 \\ & + \left[\left(\frac{m_2}{3} + m_3 \right) a_2^2 \ddot{\theta}_2 \right] + \left(\frac{m_2}{3} + m_3 \right) a_1 a_2 S_2 \dot{\theta}_1^2 \end{aligned} \quad (7)$$

$$\tau_3 = m_3 \ddot{\theta}_3 - m_3 g \quad (8)$$

Denklemlerde kullanılan uzuv kütle değerleri ve boyutları:

$$\begin{aligned} m_1 &= 2 \text{ kg} & m_2 &= 2 \text{ kg} \\ m_3 &= 1 \text{ kg} & C_2 &= \cos(\theta_2) \\ a_1 &= 2 \text{ m} & a_2 &= 1 \text{ m} \\ l_3 &= 1 \text{ m} & S_2 &= \sin(\theta_2) \\ g &= 9.8 \text{ m/s}^2 \end{aligned}$$

ile eklemlere uygulanacak tork değerleri saptanabilir.

Scara Robotun Eklemlerinin Belirlenmesi

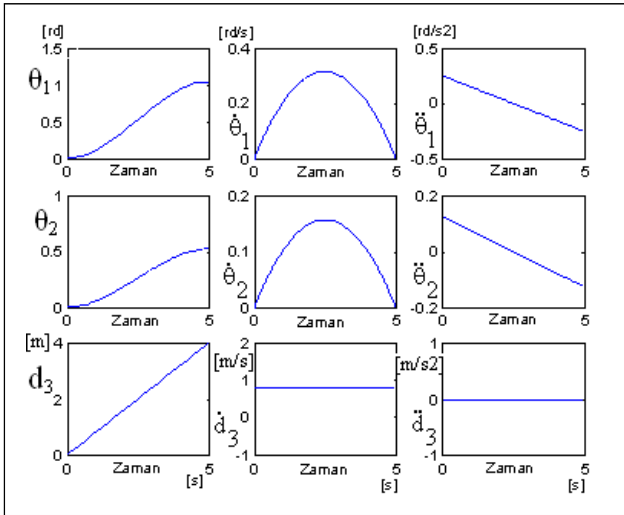
Modellenen SCARA robot kolu iki dönme ve bir öteleme hareketi yapmaktadır. Bu yüzden dönme hareketleri denk. 9 ve 10'da verilen kübik yörüngeler ile öteleme hareketi ise denk. 11'de verilen doğrusal yörünge ile ifade edilmiştir.

$$\theta_1 = (-2\pi/375)t^3 + (\pi/25)t^2 \quad (9)$$

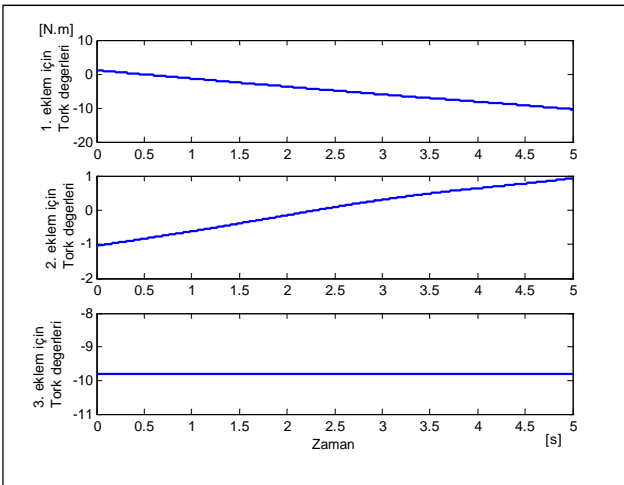
$$\theta_2 = (-\pi/375)t^3 + (\pi/50)t^2 \quad (10)$$

$$d_3 = 0,8t \quad (11)$$

Zaman aralığı 0'dan 5 saniyeye kadar alınmış ve 0.01 artımlarla konum, hız ve ivme değerleri belirlenmiştir. Elde edilen bu veriler Lagrange-Euler dinamik denklemlerinde yerine yazıldığında ise tork değerleri elde edilmiştir. Şekil 2'de konum, hız ve ivme grafikleri, Şekil 3'de tork grafikleri verilmiştir.



Şekil 2. Eklemlerin Konum, Hız ve İvme Grafikleri

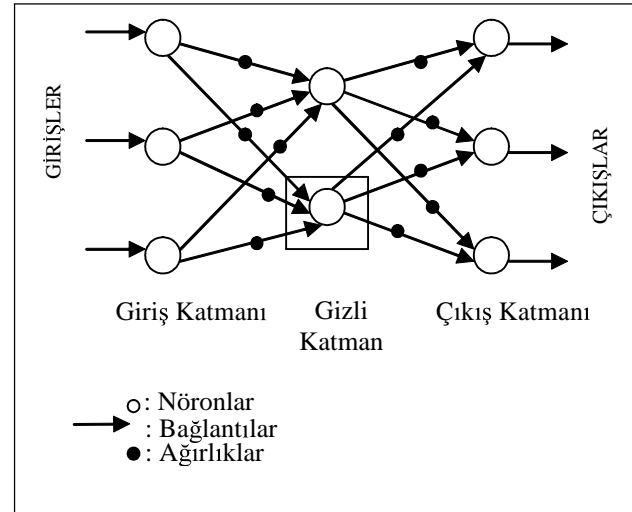


Şekil 3. Eklemlerin Tork Grafikleri

YAPAY SİNİR AĞLARI

YSA paralel dağıtılmış bir bilgi işleme sistemi olmak üzere tek yönlü işaret kanalları (bağlantılar) ile birbirine bağlanan nöronlardan (işlem elemanı) oluşur. Şekil 4'de yapay sinir ağının yapısı görülmektedir. Son derece karmaşık bir içyapıya sahip olan YSA çevre şartlarına göre davranışlarını şekillendirebilir, eğitimi esnasında girişler ve istenen çıkışların sisteme verilmesi ile kendisini farklı cevaplar verebilecek şekilde ayarlayabilir. Son yıllarda elektronik, üretim, robotik, malzeme bilimi ve

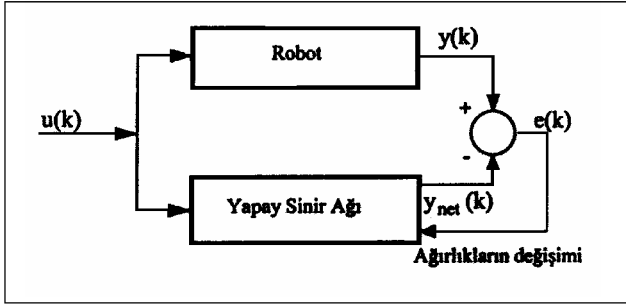
fiziksel metalurji, otomotiv, savunma ve telekomünikasyon gibi birçok mühendislik alanında başarılı sonuçlar veren YSA, paralel çalışabilmesi ve öğrenebilmesi ile insan beyninin çalışmasına benzeyen özellikleri taşır [21].



Şekil 4. YSA Modeli

Her nöronun birçok giriş işareti olmasına karşın tek bir çıkış işareti vardır ve bu diğer nöronlara giriş oluşturmak üzere birçok yola dağılır. Her bir nöron yerel bir belleğe sahiptir ve burada uyarılan katsayıların önceki hesaplamalar ile belirlenen değerleri saklanır. Her bağlantının da bir ağırlığı vardır ve bu ağırlık bir nöronun diğeri üzerindeki etkisini gösterir. Toplama fonksiyonu, bir nörona gelen net girdiyi hesaplayan bir fonksiyondur. Net girdi gelen bilgilerin ilgili bağlantıların ağırlıkları ile çarpılıp toplanması sonucunda belirlenir. Eşik fonksiyonu ise toplama fonksiyonu tarafından belirlenen net girdiyi alarak nöronun çıkışını belirleyen fonksiyondur. Nöronun çıkış ünitesi çıkış fonksiyonunun ürettiği işareti diğer nöronlara veya dış dünyaya aktarma işlevini yapar [1].

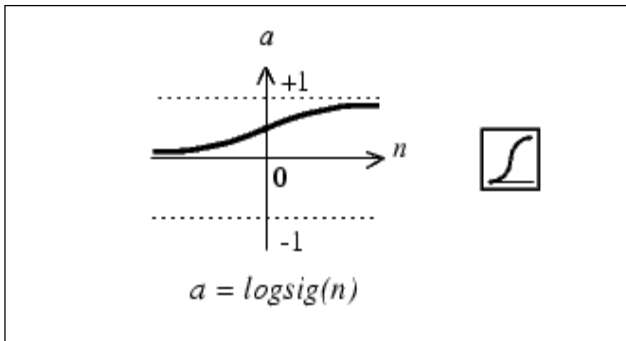
YSA ile robot modellemeye ait diyagram Şekil 5'de görülmektedir.



Şekil 5. Robot Modelleme İçin bir YSA Çözümü
($u(k)$: giriş, $y(k)$: çıkış ve $e(k)$: hata)

YSA İLE MODELLEME

Bu çalışmada ileri beslemeli geri yayılım ağı ve öğrenme stratejisi olarak da eğiticili öğrenme kullanılmıştır. Gizli katmanda ve çıkış katmanında transfer fonksiyonu olarak Log-sigmoid alınmıştır. Log-sigmoid fonksiyonu Şekil 6'da gösterilmiştir. SCARA robotun YSA ile modellenmesi için MATLAB 6.5 Neural Network Toolbox'ı kullanılmıştır.



Şekil 6. Log-Sigmoid Transfer Fonksiyonu

YSA'nın giriş katmanındaki giriş nöronlarının sayısını sistemin (problemin) girişlerinin sayısı, çıkış katmanındaki çıkış nöronlarının sayısını da sistemin çıkışlarının sayısı belirlemektedir. Sistemin girişleri her bir eklem hareketi için elde edilen konum, hız ve ivme değerleri, çıkışları ise Lagrange-Euler dinamik denklemleri sonucu bulunan tork değerleri olarak alınmıştır. Buna göre oluşturulan YSA'nın giriş nöron sayısı 9 iken çıkış nöron sayısı 3 olmaktadır. Gizli katmanda ise 16 adet nöron kullanılmıştır.

Sistemin yapılan matematiksel işlemler sonucu elde edilen toplam 500 adet giriş-çıkış vektörü bulunmaktadır. Sistemin giriş ve çıkışlarının sayısal değerlerinden oluşan veri seti 0,1-0,9 arasında olmak üzere normalize edilmiştir. Çünkü giriş-çıkış değerleri, kullanılan Log-sigmoid transfer fonksiyonunun sınırları arasında olmalıdır. Normalize ederken aşağıdaki formül dikkate alınmıştır:

$$\text{Nor. veri} = \frac{\text{Ger. veri} - \text{Min. veri}}{\text{Maks veri} - \text{Min. veri}} \times (0,9 - 0,1) + 0,1 \quad (10)$$

Oluşturulan geri yayılım ağı (backpropagation network) eğitim fonksiyonunun ağırlık ve bias değerlerini güncelleştirmek için Levenberg-Marquardt optimizasyonu kullanılmıştır. Levenberg-Marquardt metodu son yıllarda kullanılan en popüler algoritmalarından birisidir. Temelde bu algoritma maksimum komşuluk fikri üzerine kurulan en küçük kareler tahmin metodudur. Levenberg-Marquardt metodu Gauss-Newton tekniğinin ve Steepest-descent metodunun en iyi özelliklerini birleştirir ve bu algoritmaların sınırlamalarının çoğundan kaçınır [22]. Genellikle hızlı bir yakınsama yapar.

Yapay sinir ağının eğitimi için performans ölçütü olarak hataların kareleri ortalamasını kullanan Levenberg-Marquardt algoritması çok uygundur [23]. Verilen girişlere karşılık YSA'nın bulduğu çıkışlar ile hedef (istenen) çıkış karşılaştırılır. Network çıkışı ve hedef çıkış arasındaki fark hata olarak hesaplanır. Bu hatanın toplamının ortalaması minimize edilmek istenir. Bu minimize edilmek istenen değer MSE (Mean Squared Error) olarak tanımlanır ve ağ performansını belirler. MSE aşağıdaki gibi formülize edilir:

$$\text{MSE} = \frac{1}{Q} \sum_{k=1}^Q e(k)^2 = \frac{1}{Q} \sum_{k=1}^Q (t(k) - y(k))^2 \quad (13)$$

$e(k)$: Hedef ve YSA çıkışları arasındaki hata

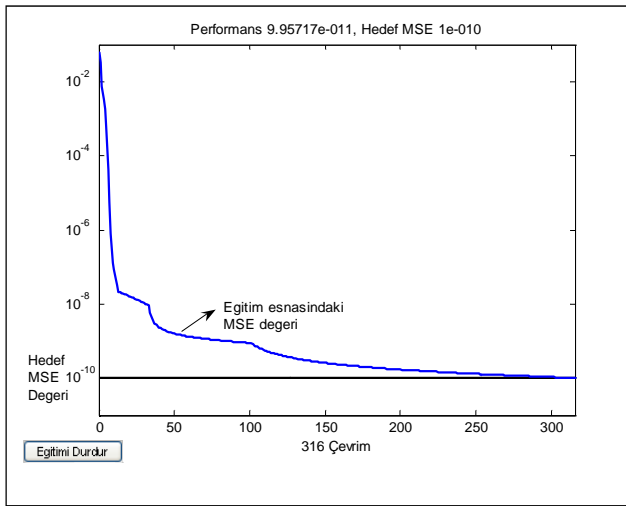
$t(k)$: Hedef çıkış

$y(k)$: YSA çıkışı

YSA modeli oluşturmak için, normalize edilen giriş-çıkış verilerinin 400 adedi ağı eğitiminde, 100 adedi de

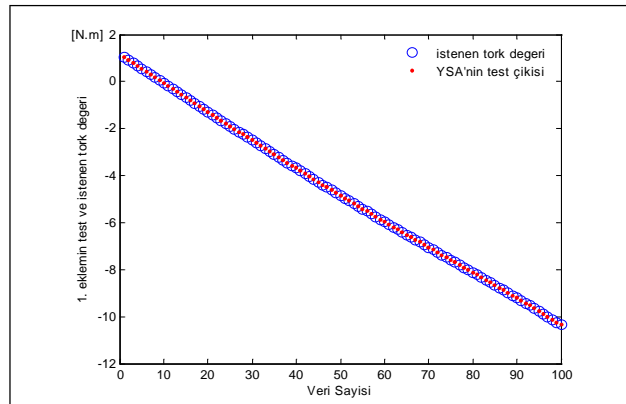
ağın testinde kullanılmıştır. Eğitim aşamasında ağa giriş ve çıkış verileri kullanılmış, test aşamasında ise sadece giriş verileri verilmek suretiyle ağ çıkışı elde edilmiştir. Test edilirken giriş vektörüne karşılık gelen çıkış vektörü, YSA'nın genelleştirme yeteneği sayesinde hedef (istenen) çıkış değerlerine oldukça yakın olarak bulunmuştur.

YSA'nın eğitimi esnasında hesaplanan MSE değerlerinin bulunduğu Neural Network Toolbox'ı tarafından çizilen performans grafiği Şekil 7'de görülmektedir.

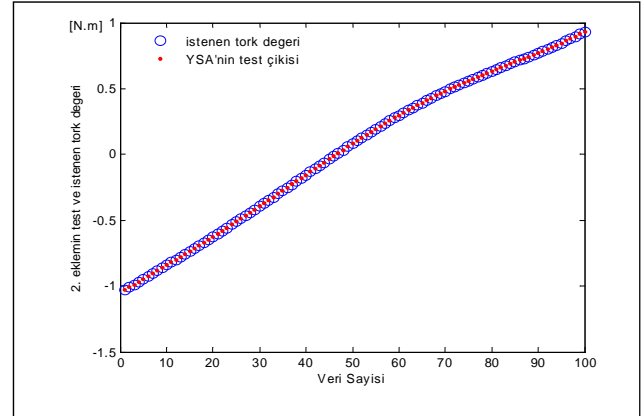


Şekil 7. YSA'nın Performans Grafiği

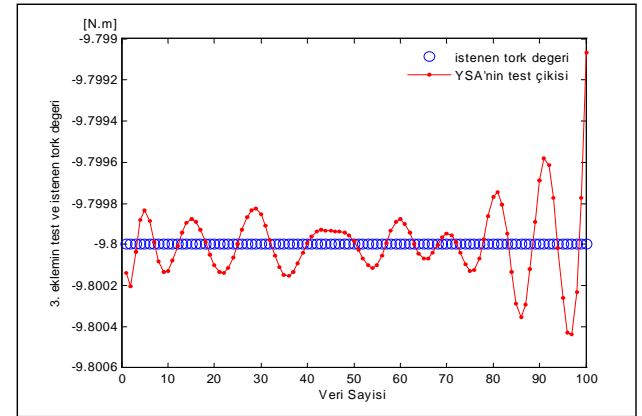
Eğitimi tamamlanan YSA'nın testi sonucunda elde edilen çıkış değerleri ve hedeflenen çıkış değerlerinin birlikte verildiği grafikler Şekil 8, Şekil 9 ve Şekil 10'da görülmektedir.



Şekil 8. 1. Ekleme Ait Test Sonucu Elde Edilen ve Hedeflenen Tork Değerleri



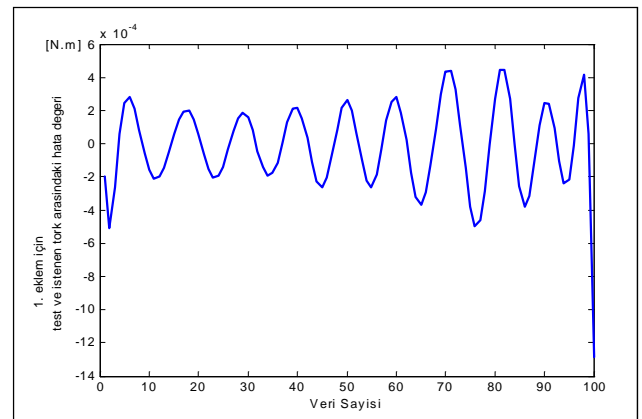
Şekil 9. 2. Ekleme Ait Test Sonucu Elde Edilen ve Hedeflenen Tork Değerleri



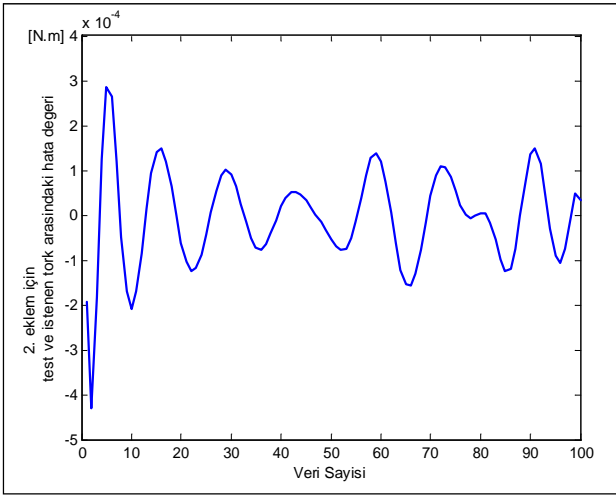
Şekil 10. 3. Ekleme Ait Test Sonucu Elde Edilen ve Hedeflenen Tork Değerleri

Ayrıca hedeflenen çıkış değerleri ve YSA'nın testi sonucunda elde edilen çıkış değerleri arasındaki hata grafikleri sırasıyla Şekil 11, Şekil 12 ve Şekil 13'de verilmiştir. Hata değerleri aşağıdaki şekilde elde edilmiştir:

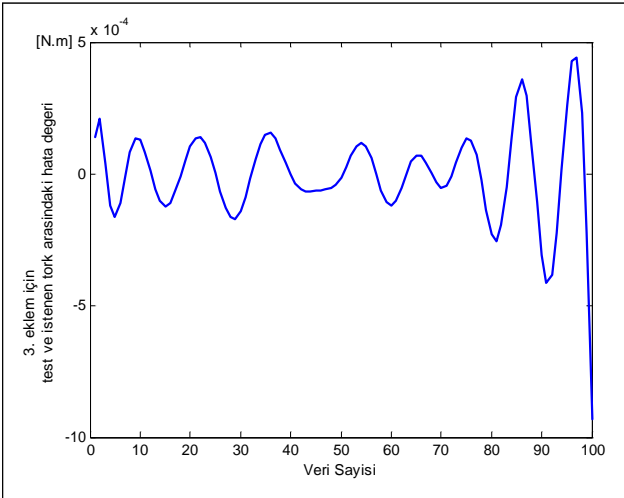
$$\text{Hata} = \text{Hedeflenen çıkış} - \text{YSA çıkışı}$$



Şekil 11. 1. Ekleme Ait Hedeflenen ve Test Sonucu Elde Edilen Tork Değerleri Arasındaki Hata Grafiği



Şekil 12. 2. Ekleme Ait Hedeflenen ve Test Sonucu Elde Edilen Tork Değerleri Arasındaki Hata Grafiği



Şekil 13. 3. Ekleme Ait Hedeflenen ve Test Sonucu Elde Edilen Tork Değerleri Arasındaki Hata Grafiği

SONUÇ

Eğitim esnasında elde edilen performans grafiğinde de görüldüğü gibi hedeflenen MSE değeri 10^{-10} olarak verilmiş ve eğitim sonucunda bulunan MSE değeri hedefe oldukça yaklaşarak 9.95717×10^{-11} olarak bulunmuştur. Böylece yapay sinir ağı 316 çevrim sonucunda hedef olarak belirlenen MSE değerini yakalayarak eğitimini tamamlamıştır.

Eğitimi tamamlanan YSA'nın testi sonucunda elde edilen çıkış değerleri ve hedeflenen çıkış değerlerinin

birlikte verildiği Şekil 8, 9 ve 10'da YSA'nın oldukça başarılı sonuçlar vererek istenen çıkışları oldukça küçük sapmalarla yakaladığı görülmektedir. Ayrıca hedeflenen çıkışlar ile YSA çıkışları arasındaki hatalar Şekil 11, 12 ve 13'de görüldüğü üzere 1. tork çıkışı maksimum hata değeri -11.5×10^{-4} , minimum hata değeri 4.5×10^{-4} , 2. tork çıkışı maksimum hata değeri -4.2×10^{-4} , minimum hata değeri 2.9×10^{-4} ve 3. tork çıkışı maksimum hata değeri -8.5×10^{-4} , minimum hata değeri ise 4.4×10^{-4} olarak hesaplanmıştır.

Sonuç olarak YSA hedeflenen çıkışları müsaade edilebilecek çok küçük sapmalarla başarılı bir şekilde yakalamış ve iyi bir performans sergileyerek SCARA robotun modellenmesi problemine oldukça iyi cevap vererek çözüm üretebilmiştir. Robotlarda serbestlik derecesi arttıkça dinamik denklemlerin elle yazılması çok karmaşık bir hale gelmektedir. Robot kontrolü uygulamalarında dinamik modelin kullanılması gerektiği durumlarda, bu çalışmadan elde edilen model, hataların az olmasından dolayı rahatlıkla kullanılabilir. İstendiği takdirde bu modele uygun bir kontrol algoritmasının da seçilmesi ile SCARA robotun kontrolü gerçekleştirilebilir.

KAYNAKÇA

1. Eğrisöğüt, A., Kazan, R., "Çimento Hammadde Karışım Prosesinin Yapay Sinir Ağları ile Matematiksel Modellenmesi", Makina Tasarım ve İmalat Teknolojileri Kongresi, 21-28, 2-3 Kasım 2001, Konya, Türkiye.
2. Daş, M.T., Dülger, L.C., "SCARA Tipi Robotun Programlanabilir Mantık Denetleyicisiyle (PLC) Kısmi Hareket Denetimi", 11. Ulusal Makina Teorisi Sempozyumu, 129-136, 4-6 Eylül 2003, Ankara, Türkiye.
3. Guez, A. and Ahmad, Z., "Solution to the Inverse Kinematics Problem in Robotics by Neural Networks", Proceeding IEEE Conference on Neural Networks, 617-624, 1988, San Diego.
4. Miyamoto, H., Kawato, M., Setoyama, T., and Suzuki, R., "Feedback-Error-Learning Neural Network for Trajectory Control of a Robotic Manipulator", IEEE Trans. Neural Networks 1, 251-265, 1988.

5. **Kawato, M., Uno, Y., Isobe, M., and Suzuki, R.**, "Hierarchical Neural Network Model for Vol-Untary Movement with Application to Robotics", *IEEE Control Systems Magazine* 8, 8-17, 1988.
6. **Omedei, A., Legnani, G., Adamini, R.**, "Three Methodologies for the Calibration of Industrial Manipulators: Experimental Results on a SCARA Robot", *Journal of Robotic Systems*, 17(6), 291-307, 2000.
7. **Köker, R., Öz, C., Çakar, T., Ekiz, H.**, "A Study of Neural Network Based Inverse Kinematics Solution for a Three-Joint Robot", *Robotics and Autonomous Systems*, 49, 227-234, 2004.
8. **Daş, M. T., Dülger, L. C.**, "Mathematical Modelling, Simulation and Experimental Verification of a Scara Robot", *Simulation Modelling Practice and Theory*, 13, 257-271, 2005.
9. **Wilhelmsen, K., Cotter N.**, "Neural Network Based Controllers for a Single-Degree-of Freedom Robotic Arm", *International Joint Conference on Neural Networks*, 407-413, 1990.
10. **Kosko, B.**, *Neural Networks and Fuzzy Systems, a Dynamical Systems Approach to Machine Intelligence*, Prentice-Hall International Editions, Englewood Cliffs, N.J., ISBN 0-13-612334-1, 1992.
11. **Rabelo, L. C., Avula, X. J. R.**, "Hierarchical Neurocontroller Architecture for Intelligent Robotic Manipulation", *Proceeding IEEE International Conference on Robotics and Automation*, Sacramento, California, 2656-2661, 1991.
12. **Fukuda, T., Shibata, T., Tokita, M., Mitsuoka, T.**, "Adaptation and Learning for Robotic Manipulator by Neural Network", *Proceeding of 29th IEEE Conference on Decision and Control*, 6, 3283-3288, 1990.
13. **Liu, H., Iberall, T., Bekey, G. A.**, "Neural Network Architecture for Robot Hand Control", *IEEE Control Systems Magazine*, 9(3), 38-43, 1989.
14. **Karakasoglu, A., Sundareshan, M. K.**, "Decentralized Variable Structure Control of Robotic Manipulators, Neural Computational Algorithm", *Proceeding of 29th Conference on Decision and Control*, 3258-3259, December, 1990.
15. **Smagt, P. P., Krose, B. J. A.**, "A Real-time Learning Neural Robot Controller", *Proceeding of International Conference on Neural Networks*, Espoo, Finland, 351-356, 1991.
16. **Tsutsumi, K., Matsumoto, H.**, "Neural Computation and Learning Strategy for Manipulator Position Control", *Proceeding of the IEEE First Annual International Conference on Neural Networks*, 4, 525-534, 1987.
17. **Çalışkan, M.**, *İki Uzuvlu Bir Robot Kolunun Hesaplanmış Moment Metoduyla Kontrolü*, Yüksek Lisans Tezi, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, 1996.
18. **Ed. Kreith, F.**, *Mechanical Engineering Handbook*, Lewis, F.L., Section 14 Robotics, CRC Press LLC, 1999.
19. **Er, M.J., Low, C.B., Nah, K.H., Lim, M.H., Ng, S.Y.**, "Real-Time Implementation of a Dynamic Fuzzy Neural Networks Controller for a SCARA", *Microprocessors and Microsystems*, 26, 449-461, 2002.
20. **Wai, R-J.**, "Tracking Control Based on Neural Network Strategy for Robot Manipulator", *Neurocomputing*, 51, 425-45, 2003.
21. **Eğrisöğüt, A.**, *Çimento Hammadde Karışım Prosesinin Yapay Sinir Ağları ile Matematiksel Modellenmesi*, Yüksek Lisans Tezi, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, 2002.
22. **Fletcher, R.**, *Unconstrained Optimization, Practical Methods of Optimization*, 1, John Wiley & Sons, ISBN 0-471-27711-8, 1980.
23. **Hagan, M.T., Demuth, H.B., Beale, M.**, *Neural Network Design*, PWS Publishing Company, Boston, 1995.