



Alınış tarihi (Received): 04.11.2016
Kabul tarihi (Accepted): 11.04.2017

Baş editor/Editors-in-Chief: **Ebubekir ALTUNTAŞ**
Alan editörü/Area Editor: **Zafer DOĞAN**

7-DOF Seri Robotun Ters Kinematik Çözümünde Eğitim Amaçlı Kullanılan Çok Katmanlı Yapay Sinir Ağının Tasarlanması ve Sonuçların Analizi

Serkan DERELİ^{a,*} Raşit KÖKER^b

^a*Sakarya Üniversitesi, Sakarya Meslek Yüksekokulu, Bilgisayar Programcılığı Bölümü, SAKARYA*

^b*Sakarya Üniversitesi, Teknoloji Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, SAKARYA*

**Sorumlu yazar e-posta: dereli@sakarya.edu.tr*

ÖZET: Yapay Sinir Ağı, insan beyninin çalışma ve düşünebilme yeteneğinden yola çıkılarak oluşturulmuş bir bilgi işlem teknolojisidir. Bu yönüyle eğitime ve öğrenebilme kabiliyetine sahiptir. Bu sebeple, robotik alanında ters kinematik çözümü için geniş bir araştırma konusu olmuş ve bu uygulamalarda sıklıkla kullanılmıştır. Çünkü geometrik, iteratif ve cebirsel gibi yöntemler redundant robotların ters kinematik çözümü için yetersiz kalmaktadır. Redundant robotlar yapısı itibarıyla karmaşık ve sonsuz sayıda çözüme sahip olan robotlardır. Bu çalışmada, 7-Dönel eklemlili redundant bir robot kolunun ters kinematik çözümü için oluşturulmuş çok katmanlı bir sinir ağını eğitim konusu ele alınmıştır. Bunu yaparken kübik yörünge planlama yöntemi kullanılarak robot koluna ait çok fazla sayıda yörünge belirlenmiş ve bu yörüngeler robot koluna öğretilmiştir. Bu sayede son konumu belli olan robot kolu, öğrenmiş olduğu yörüngelerden eklem açılarını tahmin etmektedir.

Anahtar Kelimeler – *Yapay sinir ağı, ters kinematik, robot kolu, 7-DOF, yörünge planlaması*

Design and Analysis of Multi-Layer Artificial Neural Network Used for Training in Inverse Kinematic Solution of 7-DOF Serial Robot

ABSTRACT: The Artificial Neural Network is an information processing technology created by the human brain's ability to work and think. In this sense, it has the ability to study and learn. For this reason, it has been an extensive research topic for the inverse kinematic solution in the field of robotics and has been frequently used in these applications. Because methods of geometric, iterative and algebraic are inadequate for inverse kinematic solution of redundant robots. Redundant robots are robots that have complex and infinite number of solutions in structure. In this work, the topic of training a multilayered neural network for inverse kinematic solution of a redundant robot arm that has 7 revolute angles is discussed. In doing so, a large number of trajectories of the robot arm were identified using cubic trajectory planning method and these trajectories were taught to the robot arm. Thus, the robotic arm whose final position is certain, predicts the joints from the orbits it has learned.

Keywords – *Artificial neural networks, inverse kinematics, robotic arm, 7-DOF, trajectory planning*

1. Giriş

Bir robot koluna ilişkin iş planlaması, yörünge planlaması, dinamik ve kontrol problemleri ele alındığı zaman ilk gereksinime duyulan hususlardan biri, bu robot kolun kinematik modelinin oluşturulması ve buna dayanarak gerekli kinematik ilişkilerin elde edilmesidir (Yuan ve Feng, 2014). Robotik alanda kinematik, ileri ve ters olmak üzere ikiye ayrılır. Daha basit, karmaşık olmayan ve kapalı form yöntemleriyle kolaylıkla hesaplanan (Huang ve ark., 2014) ileri kinematik, eklem açıları kullanılarak son elemanın çalışma uzayındaki konumunun ve oryantasyonunun belirlenmesi işlemidir. Çok daha karmaşık, geometrik, iteratif ve cebirsel gibi klasik yöntemlerle çözümün yetersiz kaldığı ters kinematik ise uç elemanın konum bilgilerinden eklem açılarının hesaplanmasıdır (Qiao ve ark., 2010).

Ters kinematik çözüm için kapalı form yöntemler gibi geleneksel yöntemler uzun zaman kullanılmıştır. Ancak her metodun bazı dezavantajları mevcuttur. Cebirsel metotlar, kapalı form çözümleri garanti etmedikleri bir gerçektir. Geometrik metotların kullanılması durumunda ise her defasında en fazla üç eklemi dikkate almak gerekmektedir. İteratif metotlar sadece tek bir noktaya yaklaşır ve bu nokta başlangıç noktasına bağlıdır (Dash ve ark., 2010). Eğer robot kolu; eklem yapısı çok karmaşık ise ters kinematik çözüm bu yöntemlerle çok uzun zaman almaktadır (Huang ve ark., 2012). Günümüzde, özellikle 6-DOF ve 7-DOF robot kolları hem akademide hem de endüstride ziyadesiyle çekicidir. Çünkü robot kolundaki eklem sayısı arttıkça karmaşıklık artar, fakat aynı zamanda kinematik performans, manevra kabiliyeti ve dinamik performansı da artmaktadır (Küçük ve Bingül, 2005).

Özellikle karmaşık yapıları robotların ters kinematik çözüm süresini kısaltmak için Evrimsel algoritma, yapay sinir ağları, genetik algoritma, parçacık sürü optimizasyonu, yapay arı kolonisi gibi zeki optimizasyon teknikleri günümüzde sık bir şekilde tercih edilmektedir (Wang ve ark., 2008). Bu teknikleri kullanabilmek için, ters kinematik problem denklemleri en iyi çözümü ölçebilecek uygunluk fonksiyonuna dönüştürülmektedir (Rokbani ve Alimi, 2012).

2. Yapay Sinir Ağı ile Robot Kinematiği Uygulamaları

Köker ve arkadaşları yaptıkları çalışmada (Köker ve ark., 2004) üç eklemli bir robotun ters kinematik hesabını, geri yayılım öğrenme algoritması temelli çok katmanlı bir sinir ağı tasarımı ile gerçekleştirmişlerdir. Sigmoid aktivasyon fonksiyonu kullandıkları gizli katman, 40 sinir hücresinden oluşmaktadır. Eğitim aşamasında 6000 adet veri kullanmışlar ve sonuçta kabul edilebilir bir hata ile sinir ağlarının çalıştığını ortaya koymuşlardır.

Bingül ve arkadaşları (Bingül ve ark., 2005) çalışmalarında geometrik özelliği nedeniyle analitik çözümü mümkün olmayan 6-dönel eklem sahip bir endüstriyel robotun kinematik hesabını geri yayılım algoritmasını kullanan sinir ağı ile hesaplamışlardır. Ayrıca çalışmalarında son elemanın oryantasyonunu belirlemek için üç farklı yöntem kullanmışlar ve bu yöntemleri karşılaştırmışlardır. Çalışmada 8000 adet verinin %70'ini eğitim için %30'unu ise test işlemi için kullanmışlardır.

Hasan ve arkadaşları (Hasan ve ark., 2010) çalışmalarında özellikle robot kinematiğinde ki belirsizlik ve tekillikleri ortadan kaldırmak adına yapay sinir ağı temelli bir model önermişlerdir. Bunun için 6 eklemli endüstriyel bir robotu çalışmalarında kullanmışlardır.

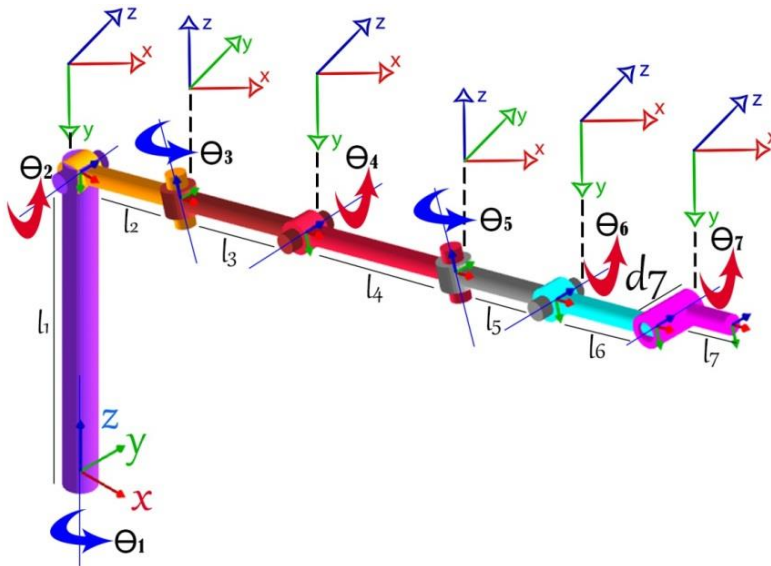
Tasarladıkları sinir ağına 55 nörona sahip tek gizli katman kullanmışlar, giriş verileri olarak uç noktanın X, Y, Z kartezyen konumları, oryantasyon verileri ve uç noktanın hızını esas almışlar, çıkış verileri olarak ise her bir eklem açısı ve hızını elde etmeyi amaçlamışlardır.

(Raj ve ark., 2015)'de ise Raj ve arkadaşları 5 eklemlilik bir robot simülasyonunu tercih etmişlerdir. Ters kinematik çözüm için hem ileri beslemeli sinir ağını hem de radyal tabanlı fonksiyonu kullanmışlar ve sonuçlarını karşılaştırmışlardır.

Yuan ve Feng ise çalışmalarında (Yuan ve Feng, 2014), PUMA560 endüstriyel robotu üzerinde BP ve RBF algoritmalarının birleşiminden oluşan bir sinir ağı tasarımı vasıtasıyla ters kinematik hesabı gerçekleştirmişlerdir. Giriş olarak 12 adet bilgiden oluşan hem konum hem de oryantasyon verilerini kullanarak çıkışta 6 adet eklem açısını elde etmişlerdir.

3. 7-DOF Robot Kolu Kinematik Analizi

Seri robotik manipulatörler, kayar ve dönel eklemlerden oluşmakla birlikte birbirine seri olarak bağlanarak tasarlanırlar ve görevlerini bu eklemlerin dönmesiyle ya da ileri geri hareketleriyle gerçekleştirirler. Kinematik, bu eklem hareketlerinin birbirleriyle olan ilişkilerini tanımlar (Tejomurtula ve Kak, 1999). Bu çalışma için tasarlanan ve Şekil-1'de görünen robot kolu, 7 adet dönel eklemden oluşmaktadır. Eklem sayısı itibariyle redundant olma özelliğini taşımaktadır. Redundant robot manipulatörü, gereğinden fazla serbestlik derecesine sahip olan robot kolunun bir türüdür. Serbestlik derecesinin fazla olması nedeniyle, karmaşık görevleri başarıyla yerine getirebilir, engellerden kaçabilecek manevra kabiliyeti yüksektir (Chen ve Lau, 2016). Tüm bu avantajlara karşın çözüm sayısı sınırsız olduğundan ve çok karmaşık olması nedeniyle klasik yöntemlerle çözümü olanaksızdır (Atawnih ve ark., 2016).



Şekil 1. Çalışmada kullanılan 7-DOF redundant robot kolu

Figure 1. 7-DOF redundant robotic arm used in the study

Günümüzde, kinematik hesaplamalar dört parametre yardımıyla oluşturulan ve Denavit-Hartenberg olarak isimlendirilen homojen dönüşüm matrisleri kullanılarak gerçekleştirilmektedir (Gan ve ark., 2005; Dash ve ark., 2011).

Çizelge 1. 7-DOF robot manipülatorüne ait DH parametreleri

Table 1. DH parameters of the 7-DOF robot arm

| i | a_i (m) | α_i | d_i (m) | Θ_i ($^\circ$)(Range) |
|---|------------|------------|------------|--------------------------------|
| 1 | 0 | -90 | $l_1=0,5$ | $-180 < \Theta_1 < 180$ |
| 2 | $l_2=0,2$ | 90 | 0 | $-90 < \Theta_2 < 30$ |
| 3 | $l_3=0,25$ | -90 | 0 | $-90 < \Theta_3 < 120$ |
| 4 | $l_4=0,3$ | 90 | 0 | $-90 < \Theta_4 < 90$ |
| 5 | $l_5=0,2$ | -90 | 0 | $-90 < \Theta_5 < 90$ |
| 6 | $l_6=0,2$ | 0 | 0 | $-90 < \Theta_6 < 90$ |
| 7 | $l_7=0,1$ | 0 | $d_7=0,05$ | $-30 < \Theta_7 < 90$ |

Tablo 1’ de “i” eklem sırasını ifade eder. “a” ve “d” birimi metre, “ α ” ve “ Θ ” birimi ise derecedir. Kinematik hesaplamalar için kullanılacak genel dönüşüm matrisi aşağıda görülmektedir.

$${}_{i-1}^i T = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \cdot \sin\theta_i & \sin\alpha_i \cdot \sin\theta_i & a_i \cdot \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cdot \cos\theta_i & -\cos\theta_i \cdot \sin\alpha_i & a_i \cdot \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Yukarıdaki genel matrizen robot manipülatorü uç elemanının konum ve oryantasyonunu bulmak için kullanılacak denklem ise şu şekildedir:

$$A_{\text{End-Effector}} = {}_0^7 T = {}_0^1 T \cdot {}_1^2 T \cdot {}_2^3 T \cdot {}_3^4 T \cdot {}_4^5 T \cdot {}_5^6 T \cdot {}_6^7 T = \begin{bmatrix} n_x & s_x & a_x & P_x \\ n_y & s_y & a_y & P_y \\ n_z & s_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

${}^i T_{i+1}$, i. uzvun transfer matrisini, ${}^0 T_7$ matrisi, ana çerçeve ile uç nokta arasındaki dönüşüm matrisini ifade eder. Her bir dönüşüm matrisi 3x3’lük dönme matrisi ($n_x, n_y, n_z, s_x, s_y, s_z, a_x, a_y, a_z$) ve 3x1’lik konum vektöründen (P_x, P_y ve P_z) oluşur. Bu çalışma için kullanılan robot kolunun konum denklemleri aşağıda verilmiştir. Bu denklemlerde s, sinüs ve c, kosinüs ifade edilmektedir.

$$P_x = (c\theta_1 c\theta_2 c\theta_3 c\theta_4 - s\theta_1 s\theta_3 s\theta_4 - c\theta_1 s\theta_2 s\theta_4)(c\theta_5 c\theta_6 l_7 c\theta_7 - c\theta_5 s\theta_6 l_7 s\theta_7 - s\theta_5 d_7 + c\theta_5 l_6 c\theta_6 + l_5 c\theta_5) \\ + (-c\theta_1 c\theta_2 s\theta_3 - s\theta_1 c\theta_3)(s\theta_5 c\theta_6 l_7 c\theta_7 - s\theta_5 s\theta_6 l_7 s\theta_7 + c\theta_5 d_7 + s\theta_5 c\theta_6 l_6 + l_5 s\theta_5) \\ + (c\theta_1 c\theta_2 c\theta_3 s\theta_4 - s\theta_1 s\theta_3 s\theta_4 + c\theta_1 c\theta_4 s\theta_2)(-s\theta_6 l_7 c\theta_7 - c\theta_6 l_7 s\theta_7 - l_6 s\theta_6) \\ + c\theta_1 c\theta_2 (c\theta_3 c\theta_4 l_4 + l_3 c\theta_3) - s\theta_1 (s\theta_3 c\theta_4 l_4 + l_3 s\theta_3) - c\theta_1 s\theta_2 l_4 s\theta_4 + c\theta_1 c\theta_2 l_2$$

$$\begin{aligned}
P_y = & s\theta_1 c\theta_2 c\theta_3 c\theta_4 + c\theta_1 s\theta_3 c\theta_4 - s\theta_1 s\theta_2 s\theta_4)(c\theta_5 c\theta_6 l_7 c\theta_7 - c\theta_5 s\theta_6 l_7 s\theta_7 - s\theta_5 d_7 + c\theta_5 c\theta_6 l_6 + l_5 c\theta_5) \\
& + (-s\theta_1 c\theta_2 s\theta_3 + c\theta_1 c\theta_3)(s\theta_5 c\theta_6 l_7 c\theta_7 - s\theta_5 s\theta_6 l_7 s\theta_7 + c\theta_5 d_7 + s\theta_5 c\theta_6 l_6 + l_5 s\theta_5) \\
& + (s\theta_1 c\theta_2 c\theta_3 s\theta_4 + c\theta_1 s\theta_3 s\theta_4 + s\theta_1 s\theta_2 c\theta_4)(-s\theta_6 l_7 c\theta_7 - c\theta_6 l_7 s\theta_7 - l_6 s\theta_6) \\
& + s\theta_1 c\theta_2 (c\theta_3 c\theta_4 l_4 + l_3 c\theta_3) + c\theta_1 (s\theta_3 c\theta_4 l_4 + l_3 s\theta_3) - s\theta_1 s\theta_2 s\theta_4 l_4 + s\theta_1 c\theta_2 l_2
\end{aligned}$$

$$\begin{aligned}
P_z = & (-s\theta_2 c\theta_3 c\theta_4 - c\theta_2 s\theta_4)(c\theta_5 c\theta_6 l_7 c\theta_7 - c\theta_5 s\theta_6 l_7 s\theta_7 - s\theta_5 d_7 + c\theta_5 c\theta_6 l_6 + l_5 c\theta_5) \\
& + s\theta_2 s\theta_3 (s\theta_5 c\theta_6 l_7 c\theta_7 - s\theta_5 s\theta_6 l_7 s\theta_7 + c\theta_5 d_7 + s\theta_5 c\theta_6 l_6 + s\theta_5 l_5) \\
& + (-s\theta_2 c\theta_3 s\theta_4 + c\theta_2 c\theta_4)(-s\theta_6 l_7 c\theta_7 - c\theta_6 l_7 s\theta_7 - s\theta_6 l_6) - s\theta_2 (c\theta_3 c\theta_4 l_4 + l_3 c\theta_3) \\
& - c\theta_2 s\theta_4 l_4 - s\theta_2 l_2 + l_1
\end{aligned}$$

3.1. Yörünge Planlama

Manipülâtörün uç noktasının, sabit bir sürede başlangıç noktasından son durumuna kadar hareketi esnasında, yer değıştirme ve dönme yollarını belirleyen noktalar kümesi “Yörünge” olarak isimlendirilir (Sciavicco ve Siciliano, 2000). Robot hareketlerinde temel problem robot uç elemanını, o anki başlangıç değerinden, istenilen bir son değere taşımaktır. Bu harekette, robot kolunun hem yönü hem de pozisyonu değışmektedir. Bu değışimi zamana bağılı olarak gerçekleştirmek yörünge planlaması olarak tanımlanmaktadır. Yörünge planlamasında her bir eklem açısı belli oranda değışmektedir. Bu çalışmada, bu değışimler aşağıda formülü verilen kübik yörünge planlama metodu ile hesaplanmıştır. Bu sayede eğitim ve test işlemi için kullanılan 5000 adet veri elde edilmiştir.

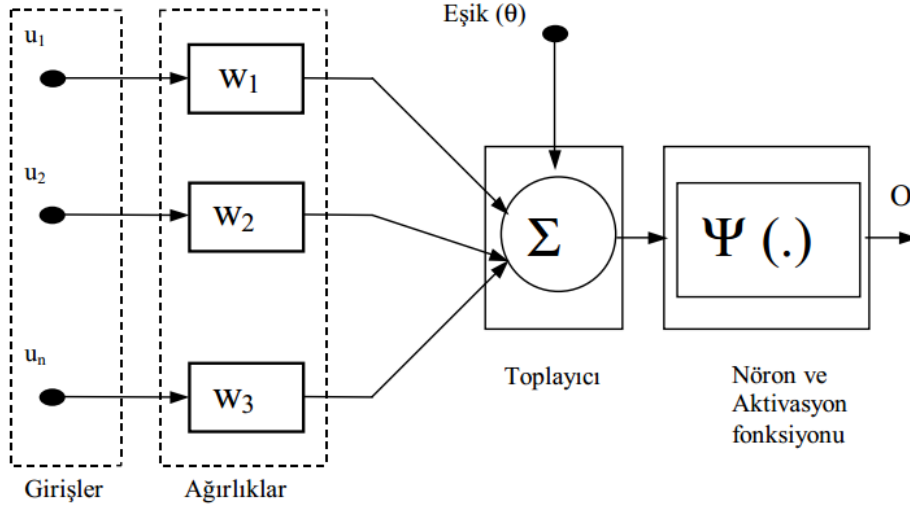
$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

Yukarıdaki denklemde yer alan bütün katsayılar, pozisyonun, hızın ve ivmenin başlangıç ve son değerleri kullanılarak bulunabilmektedir. Her bir eklem açısının belirlenen süre (t) dahilinde başlangıç ve son değerleri aşağıdaki denklemle elde edilmiştir. Bu denklemde “ m ” eklem açısını ifade etmektedir (Köker ve ark., 2004).

$$\theta_i(t) = \theta_{i0} + \frac{3}{t_f^2}(\theta_{if} - \theta_{i0})t^2 - \frac{2}{t_f^3}(\theta_{if} - \theta_{i0})t^3 \quad i = 1, \dots, m$$

4. Yapay Sinir Ağları

Yapay Sinir Ağları (Artificial Neural Networks) insan beyninden esinlenerek geliştirilmiş, ağırlıklı bağlantılar aracılığı ile birbirine bağlanan işlem elemanlarından oluşan paralel ve dağıtılmış bilgi işleme yapılarıdır. En önemli özelliği, deneyimlerden (tecrübe) yararlanarak öğrenebilmesidir. Yapay sinir ağları, insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri herhangi bir yardım almadan otomatik olarak gerçekleştirebilmenin yanı sıra bilgiler arasında ilişki kurabilme yeteneğine de sahiptir (Toshani ve Farrokhi, 2014).



Şekil 2. Yapay sinir ağı hücresi

Figure 2. Artificial neural network cell

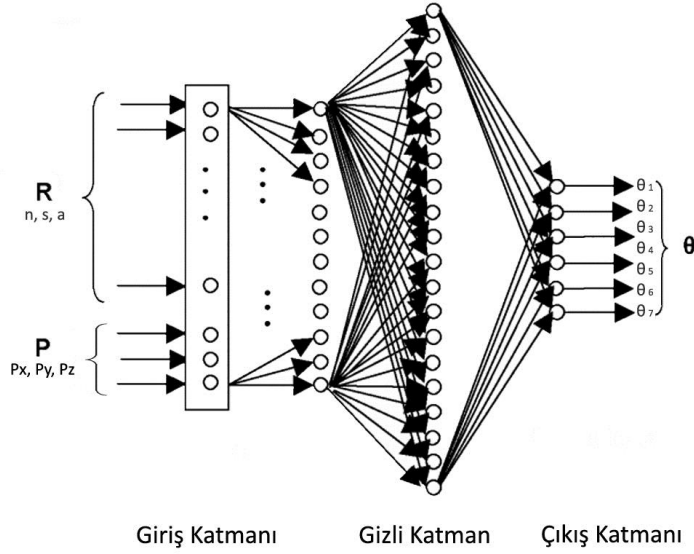
Şekil 2’de görüldüğü üzere yapay sinir hücresinin görevi; u_n girdi örüntüsüne karşılık “O” çıktısı sinyali oluşturmak ve bu sinyali diğer hücelere iletmektir. Her u_n ile “O” arasındaki ilişkiyi temsil eden W_n ağırlıkları, her yeni girdi örüntüsü ve çıktı sinyaline göre tekrar ayarlanır. Bu ayarlama süreci öğrenme olarak adlandırılır. Öğrenmenin tamamlandığının belirtilebilmesi için; girdi örüntüleri, W_n ağırlıklarındaki değişim sabit olana dek sistemi beslemektedir. Stabilizasyon (durağanlık) sağlandığı zaman hücre öğrenmesini tamamlamıştır (Duka, 2013).

Genellikle verileri belli olan girdi ve çıktı değerleri sisteme girilerek öğrenme kuralına göre ağırlık değerleri otomatik olarak değiştirilmektedir. Eğitim işleminin tamamlanmasından sonra eğitilmiş olan ağ, ağırlık değerlerinin son durumuna göre, verilen herhangi bir veri setinin sonucunu tahmin edebilmektedir (Bingül ve ark., 2005; Raj ve ark., 2015).

4.1. Geri Yayılım Sinir Ağı Yapısı

Bu çalışma için tasarlanan sinir ağı, geri yayılım algoritması ile öğrenme gerçekleştirmektedir ve çok katmanlı bir yapıya sahiptir. Geriye yayılım öğrenmesinde, ağ her giriş örüntüsüne göre çıkışta üretilen sonuçları gizli katmanlardan geçirir. Devamında çıkışta oluşan hataları bulmak için elde edilen değerlerle gerçek değerler karşılaştırılır ve bu hataların türevi geriye doğru gizli katmanlara iletilir. Gizli katmanlardaki nöronlar kendi hatalarını azaltmak için ağırlıklarını kullanırlar. Bu sinir ağı yapısında hayati derecede önemli faktör; gizli katmanda kullanılacak nöron sayısının seçimidir. Çünkü az sayıda seçilen nöron sayısı öğrenmeyi zorlaştırdığı gibi, çok sayıda seçilen nöron sayısı da genelleşme yeteneğini düşürmektedir. Bu nedenle bu çalışmada iki adet gizli katman kullanılmış ve bu katmanlardaki nöron sayıları da sırayla 16 ve 8’dir.

Bu çalışmada, geriye yayılım (BP) ağı, LM algoritması vasıtasıyla eğitilmiştir ve kullanılan algoritmanın yapısı Şekil 3’te gösterildiği gibi giriş verileri sayısı 12 ($n_x, n_y, n_z, s_x, s_y, s_z, a_x, a_y, a_z$), çıkış verileri sayısı 7 ($\Theta_1, \Theta_2, \Theta_3, \Theta_4, \Theta_5, \Theta_6, \Theta_7$) adettir.



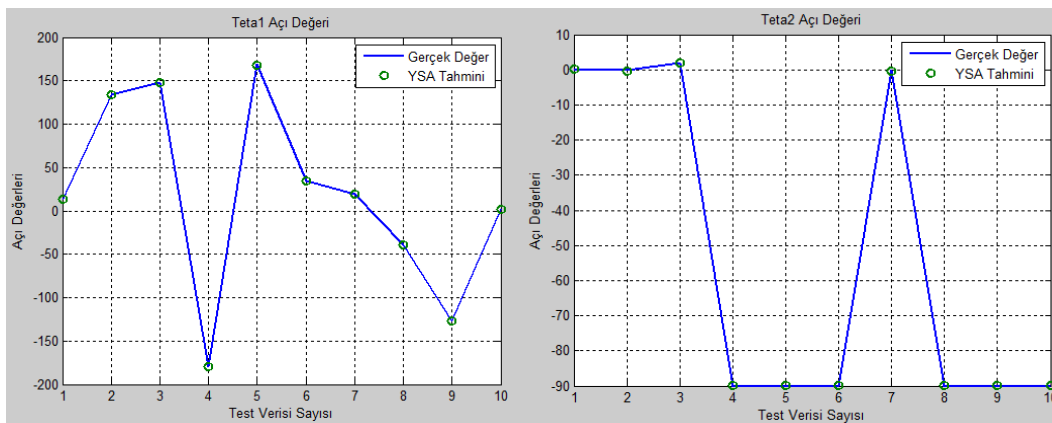
Şekil 3. Çalışmada kullanılan BP tabanlı sinir ağı yapısı

Figure 3. BP based neural network structure used in the study

Toplamda 6000 adet veriden 4800'ü ağı eğitilmesi için geri kalan 1200 ise test işlemi için kullanılmıştır. Yapay sinir ağı LM geri yayılım algoritmasıyla eğitilmiş, gizli katmanlarda sigmoid aktivasyon fonksiyonu ve MSE (ortalama karesel hata) performans fonksiyonu kullanılmıştır.

6. Simülasyon Sonuçları ve Tartışma

Bu çalışma için önerilen çok katmanlı sinir ağı ile test işlemi için ayrılan 1200 veri ile elde edilen sonuçlar, yapay sinir ağlarının kabul edilebilir bir hata ile çalışmalarına adeta kanıttır. Ancak, herkesçe bilinen bir gerçek var ki, eğitime işleminin uzun sürmesi ise yine bu çalışmada açıkça gözlemlenmiştir. Çalışmaya göre test verilerinden rasgele seçilen 10 adet veri ile ilgili karşılaştırmalı sonuçlar aşağıda verilmektedir.

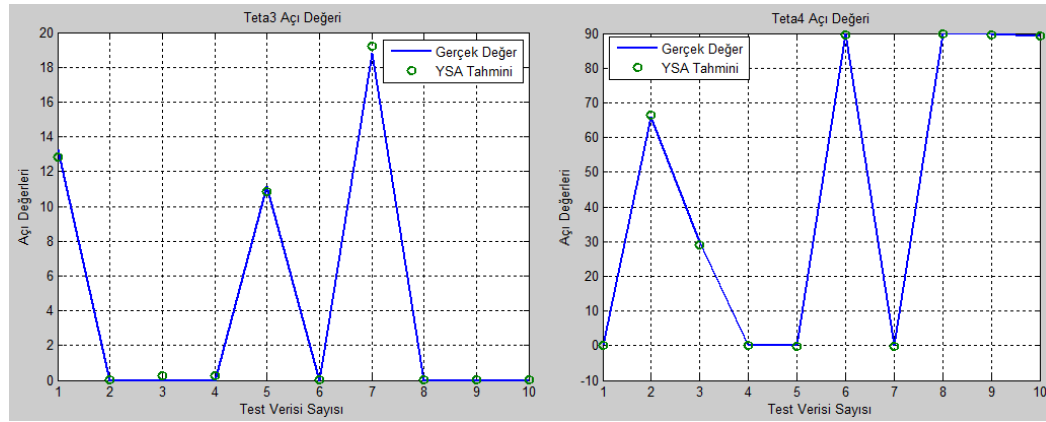


Şekil 4. Teta1 ve Teta2 açıları gerçek ve tahmini değerler

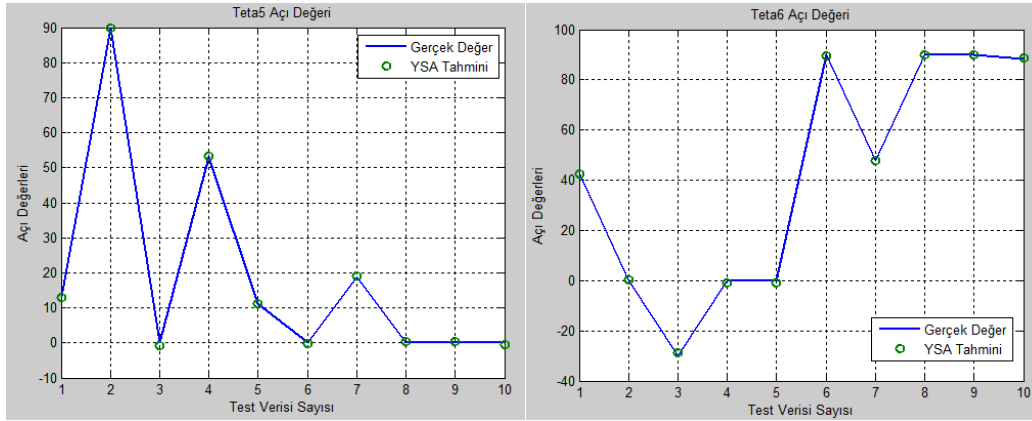
Figure 4. Teta1 and Teta2 angles are real and estimated values

Çizelge 2. Θ_1 ve Θ_2 açıları gerçek ve tahmin değerleri arasındaki farklar**Table 2.** Differences between Θ_1 and Θ_2 real and predicted values

| Θ_1 | | | Θ_2 | | |
|------------|----------|---------|------------|----------|---------|
| Gerçek | YSA | Fark | Gerçek | YSA | Fark |
| 13,2496 | 13,81264 | 0,56304 | 0 | 0,11348 | 0,11348 |
| 133,9085 | 134,3131 | 0,40462 | 0 | -0,51659 | 0,51659 |
| 147,9418 | 148,1997 | 0,25798 | 2,05824 | 2,09598 | 0,03774 |
| -180 | -179,614 | 0,38640 | -90 | -89,9609 | 0,03912 |
| 168,8486 | 168,0645 | 0,78418 | -90 | -89,9077 | 0,09234 |
| 34,12512 | 34,41163 | 0,28650 | -90 | -89,7482 | 0,25176 |
| 18,80064 | 18,8183 | 0,01765 | 0 | -0,26233 | 0,26233 |
| -38,8147 | -39,0626 | 0,24783 | -90 | -89,8769 | 0,12312 |
| -126,812 | -127,092 | 0,27955 | -90 | -89,9969 | 0,00301 |
| 2,45376 | 2,3147 | 0,13906 | -90 | -89,8419 | 0,15814 |

**Şekil 5.** Teta3 ve Teta4 açıları gerçek ve tahmini değerler**Figure 5.** Teta3 and Teta4 angles are real and estimated values**Çizelge 3.** Θ_3 ve Θ_4 açıları gerçek ve tahmin değerleri arasındaki farklar**Table 3.** Differences between Θ_3 and Θ_4 real and predicted values

| Θ_3 | | | Θ_4 | | |
|------------|----------|----------|------------|----------|---------|
| Gerçek | YSA | Fark | Gerçek | YSA | Fark |
| 13,2496 | 12,82818 | 0,42141 | 0 | -0,03132 | 0,03132 |
| 0 | 0,00066 | 0,00066 | 65,86272 | 66,38569 | 0,52297 |
| 0 | 0,25739 | 0,25739 | 30 | 29,12179 | 0,87820 |
| 0 | 0,24009 | 0,24009 | 0 | 0,18059 | 0,18059 |
| 11,15136 | 10,83566 | 0,31570 | 0 | -0,21391 | 0,21390 |
| 0 | 5,37E-07 | 5,37E-07 | 90 | 89,70659 | 0,29340 |
| 18,80064 | 19,18304 | 0,38240 | 0 | -0,10294 | 0,10293 |
| 0 | 1,50E-08 | 1,50E-08 | 90 | 89,94183 | 0,05817 |
| 0 | 0,00411 | 0,00411 | 90 | 89,69897 | 0,30102 |
| 0 | 4,10E-07 | 4,10E-07 | 89,18208 | 89,38381 | 0,20173 |



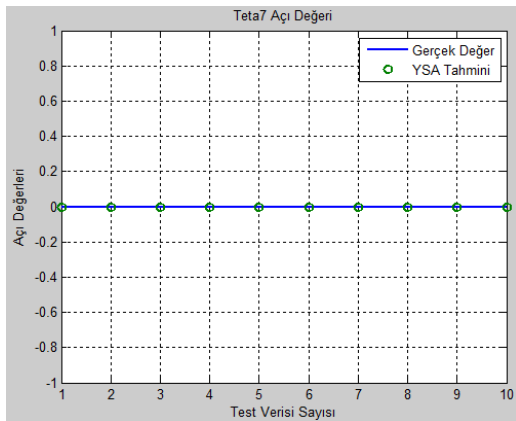
Şekil 6. Teta5 ve Teta6 açıları gerçek ve tahmini değerler

Figure 6. Teta5 and Teta6 angles are real and estimated values

Çizelge 4. Θ_5 ve Θ_6 açıları gerçek ve tahmin değerleri arasındaki farklar

Table 4. Differences between Θ_5 and Θ_6 real and predicted values

| Θ_5 | | | Θ_6 | | |
|------------|----------|---------|------------|----------|---------|
| Gerçek | YSA | Fark | Gerçek | YSA | Fark |
| 13,2496 | 12,91418 | 0,33541 | 42,30144 | 42,5284 | 0,22696 |
| 90 | 89,80837 | 0,19163 | 0 | 0,31503 | 0,31503 |
| 0 | -0,84721 | 0,84721 | -30 | -28,6171 | 1,38287 |
| 53,06112 | 53,33439 | 0,27326 | 0 | -0,90295 | 0,90295 |
| 11,15136 | 11,0678 | 0,08356 | 0 | -0,8287 | 0,82869 |
| 0 | -0,1386 | 0,13856 | 90 | 89,5113 | 0,48863 |
| 18,80064 | 19,07872 | 0,27807 | 47,69856 | 47,72494 | 0,02638 |
| 0 | 0,52123 | 0,52123 | 90 | 89,99576 | 0,00423 |
| 0 | 0,31568 | 0,31568 | 90 | 89,95342 | 0,04658 |
| 0 | -0,37627 | 0,37627 | 88,36416 | 88,6163 | 0,25213 |



Şekil 7. Teta7 açıları gerçek ve tahmini değerler

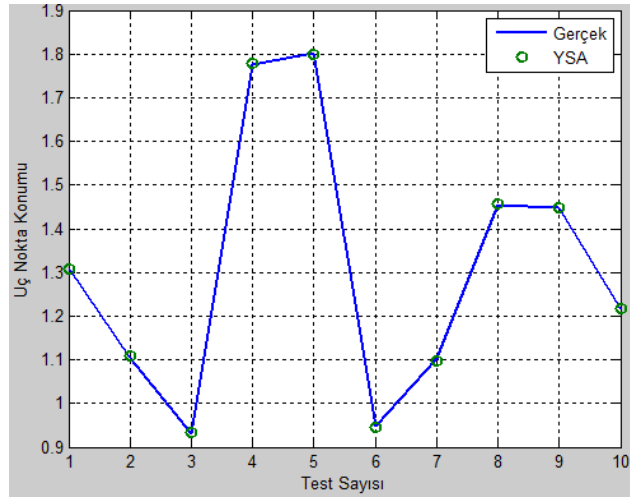
Figure 7. Teta7 angles are real and estimated values

Θ_7 açısının tüm değerleri sıfır olduğundan ilave bir çizelgede bu belirtilmemiştir. Her bir eklem için elde edilen değerler yukarıda detaylıca verilmiştir. Ancak bilindiği üzere seri robotlarda her bir eklemde ki pozisyon hatası uç noktaya katlanarak yansımaktadır. Bu nedenle uç nokta için elde edilen değerlerde son derece önemlidir.

Çizelge 5. Her bir test sonucu için uç noktanın konum hataları**Table 5.** The end effector position errors for each test result

| Seçilen Veriler | Uç nokta pozisyon hatası |
|-----------------|--------------------------|
| 1. Test Sonucu | 9.5906e-04 |
| 2. Test Sonucu | 4.7620e-03 |
| 3. Test Sonucu | 3.3173e-03 |
| 4. Test Sonucu | 2.2387e-03 |
| 5. Test Sonucu | 2.9672e-03 |
| 6. Test Sonucu | 1.2387e-03 |
| 7. Test Sonucu | 7.6193e-03 |
| 8. Test Sonucu | 4.4940e-04 |
| 9. Test Sonucu | 9.0779e-04 |
| 10. Test Sonucu | 7.8550e-04 |

Bu çalışmada gerçekleştirilen sonuçlara göre elde edilen en kötü değer $7.6193e-03$, en iyi değer ise $4.4940e-04$ olarak görülmektedir.

**Şekil 8.** Uç noktanın gerçek ve YSA ile elde edilen konumları**Figure 8.** The positions of the end point with real and YSA

7. Sonuç

Bu çalışmada, redundant özellikli ve 7-dönel ekleme sahip bir robot kolunun ters kinematik hesaplaması, geri yayılım temelli bir yapay sinir ağı ile gerçekleştirilmiş ve kabul edilebilir hatalarla eklem açılarının elde edildiği görülmüştür. Özellikle eğitim aşamasının son derece zor ve uzun süre alması, bu yöntemin bir dezavantajı gibi görünse de elde edilen değerler açısından başarılı ve bu tip hesaplamalarda kullanılabilir bir yöntem olduğu açıkça ortaya çıkmıştır. Çalışmada özellikle serbestlik derecesi fazla olan robot kolunun kullanılmasının temel amaçlarından biri de; yapay sinir ağları için literatürde çokça bahsedilen yüksek tahmin ve hızlı yakınsama özelliğinin, çözümü geleneksel yöntemlerle elde edilemeyen hesaplamalarda ortaya konmasıdır. Böylece geleneksel yöntemlerle ortaya çıkan pek çok dezavantajın, yapay sinir ağları ile üstesinden gelindiği görülmektedir. Özellikle seri robotlardaki eklem açılarında meydana gelen hataların uç elemanın pozisyonuna katlanarak yansımaları en büyük engel olsa da bu çalışmada, bu durumun gözlemlenmediği ve elde edilen değerlerin kabul edilebilir sınırlar içinde kaldığı görülmektedir.

Kaynaklar

- Atawnih, A., Papageorgiou, D., Zou, D., 2016. Kinematic Control of Redundant Robots with Guaranteed Joint Limit Avoidance. *Robotics and Autonomous Systems*, 79:122-131.
- Bingül, Z., Ertunç, H.M., Oysu, C., 2005. Comparison of Inverse Kinematics Solutions Using Neural Network for 6R Robot Manipulator with Offset, *Computational Intelligence Methods and Applications*.
- Chen, J., Lau, H.Y.K., 2016. Inverse Kinematics Learning for Redundant Robot Manipulators with Blending of Support Vector Regression Machines. *IEEE Advanced Robotics and its Social Impacts*.
- Dash, K.K., Choudhury, B.B., Khuntia, A.K., Biswal, B.B., 2011. A Neural Network Based Inverse Kinematic Problem. *Recent Advances in Intelligent Computational Systems*.
- Dash, K.K., Choudhury, B.B., Khuntia, A.K., Biswal, B.B., 2011. A Neural Network Based Inverse Kinematic Problem. *Recent Advances in Intelligent Computational Systems*, 471-476.
- Duka, A.V., 2013. Neural Network based Inverse Kinematics Solution for Trajectory Tracking of a Robotic Arm. *The 7th International Conference Interdisciplinarity in Engineering*.
- Gan, J.Q., Oyama, E., Resales, E.M., Hu, H., 2005. A Complete Analytical Solution to the Inverse Kinematics of the Pioneer 2 Robotic Arm. *Robotica*, 23: 123-129.
- Hasan, A.T., İsmail, N., Hamouda, A.M.S., Aris, I., Marhaban, M.H., Al-Assadi, H.M.A., 2010. Artificial Neural Network Based Kinematics Jacobian Solution for Serial Manipulator Passing Through Singular Configurations. *Advances in Engineering Software*, 41: 359-367.
- Huang, H.C., Xu, S.D., Hsu, H., S., 2014. Hybrid Taguchi DNA Swarm Intelligence for Optimal Inverse Kinematics Redundancy Resolution of Six-DOF Humanoid Robot Arms. *Mathematical Problems in Engineering*, 1: 1-9.
- Huang, H.C., Chen, C.P., Wang, P.R., 2012. Particle Swarm Optimization for Solving the Inverse Kinematics of 7-DOF Robotic Manipulators. *IEEE International Conference on Systems, Man, and Cybernetics*.
- Köker, R., Öz, C., Çakar, T., Ekiz, H., 2004. A study of neural network based inverse kinematics solution for a three-joint robot. *Robotics and Autonomous Systems*, 49: 227-234.
- Küçük, S., Bingül, Z., 2005. The inverse kinematics solutions of fundamental robot manipulators with offset wrist, *IEEE International Conference on Mechatronics*.
- Qiao, S.G., Liao, Q.Z., Wei, S.M., Su, H.J., 2010. Inverse kinematic analysis of the general 6R serial manipulators based on double quaternions. *Mechanism and Machine Theory*, 45 (2): 193-199.
- Raj, D.R., Raglend, I.J., Anand, M.D., 2015. Inverse Kinematics Solution of a Five Joint Robot Using Feed Forward and Radial Basis Function Neural Network. *Computation of Power, Energy Information and Communication (ICCPEIC)*.

- Rokbani, N., Alimi, A.M., 2012. IK-PSO, PSO Inverse Kinematics Solver with Application to Biped Gait Generation. *International Journal of Computer Applications*, 58:33-39.
- Sciavicco, L., Siciliano, B., 2000. *Modeling and Control of Robot Manipulators*. Springer-Verlag, London.
- Tejomurtula, S., Kak, S., 1999. Inverse Kinematics in Robotics Using Neural Networks. *Information Sciences*, 116: 147-164.
- Toshani, H, Farrokhi M. Real-Time Inverse Kinematics of Redundant Manipulators Using Neural Networks and Quadratic Programming: A Lyapunov-Based Approach. *Robotics and Autonomous Systems* 2014; 62: 766-781.
- Wang, X.S., Hao, M.L., Cheng, Y.H., 2008. On the use of differential evolution for forward kinematics of parallel manipulators, *Applied mathematics and computation*, 205:760-796.
- Yuan, T., Feng, Y., 2014. A New Algorithm for Solving Inverse Kinematics of Robot Based on BP and RBF Neural Network. *International Conference on Instrumentation and Measurement, Computer, Communication and Control*.