

PAPER DETAILS

TITLE: Ince Dikdörtgen Plakların Titresim Frekanslarının Yapay Sinir Ağları Yaklaşımı ile Tahmini

AUTHORS: Ömer CİVALEK, Yusuf CALAYIR

PAGES: 4161-4176

ORIGINAL PDF URL: <https://dergipark.org.tr/tr/download/article-file/136667>

İnce Dikdörtgen Plakların Titreşim Frekanslarının Yapay Sinir Ağları Yaklaşımı ile Tahmini

Ömer CİVALEK*
Yusuf CALAYIR**

ÖZ

Çalışmada, 21 farklı mesnet koşuluna sahip dikdörtgen plakların titreşim hesabı için bir yapay sinir ağı uygulaması sunulmuştur. Hatanın geriye yayılım algoritmasına dayalı çok katmanlı yapay sinir ağı kullanılarak plağın iki ardışık moduna ait ilk üç frekansı elde edilmiştir. Ağın eğitimi, literatürde iyi bilinen Leissa'nın[37] makalesindeki veri kullanılarak 11 farklı mesnet durumu için yapılmıştır. Bununla birlikte, ağın test edilmesi için eğitim setinde olmayan diğer 10 farklı mesnet koşulu kullanılmıştır. Yapay sinir ağları ile elde edilen sonuçlar sayısal yöntemlerle elde edilen sonuçlara çok yakın çıkmıştır.

ABSTRACT

Estimation of the Vibration Frequencies of Thin Rectangular Plates by Artificial Neural Networks Approach

An artificial neural network application is presented for vibration analysis of rectangular plates with 21 various support conditions. The first three natural frequencies of two sequences mode of plates are obtained using multi-layer neural network based back-propagation error learning algorithm. The training of the network has been made using data that obtained the well-known paper of Leissa [37], for 11 different support conditions. The trained neural network, however, had been tested for 10 other different support conditions which were not included in the training set. The results found by using artificial neural network are sufficiently close to the numerical results.

1. GİRİŞ

Yapay zeka bir bilgisayarın ya da bilgisayar denetimli bir makinanın, genellikle insana özgü nitelikler olduğu varsayılan akıl yürütme, karar verme, anlam çıkartma, genelleme yapma ve geçmiş deneyimlerden yararlanarak öğrenme gibi yüksek zihinsel süreçlere ilişkin görevleri yerine getirme yeteneği olarak tanımlansa bile yapay zeka kavramı, üzerinde çok tartışılan bir konudur [1,2,3,4,5].

Not: Bu yazı

- Yayın Kurulu'na 17.03.2005 günü ulaşmıştır.
- 30 Eylül 2007 gününe kadar tartışmaya açıktır.

* Akdeniz Üniversitesi, İnşaat Mühendisliği Bölümü, Antalya – civalek@yahoo.com

** Fırat Üniversitesi, İnşaat Mühendisliği Bölümü, Elazığ

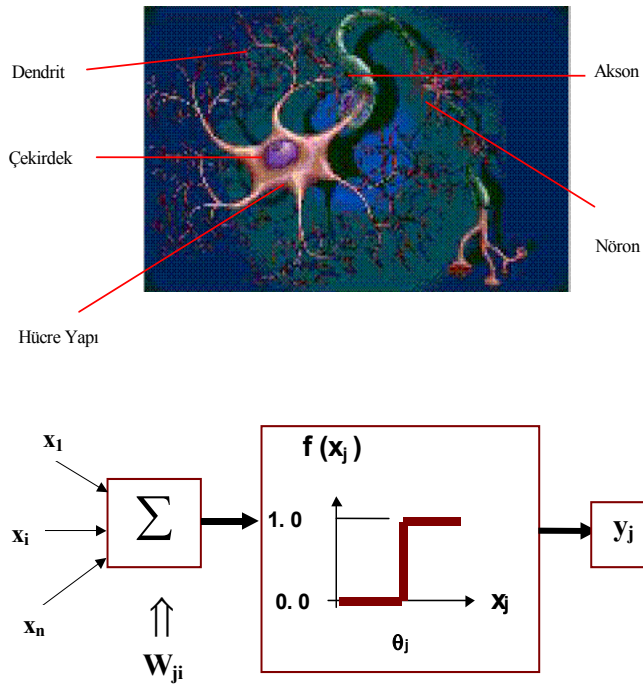
20. yüzyılın ikinci yarısında temelleri atılan ve günümüzde büyük bir kitlenin üzerinde çalıştığı yapay zeka tekniği, pek çok problemin analizinde başarı ile kullanılmış ve klasik programlamaya alternatif olmuştur. Başlangıçta çok farklı amaçlar (tıp alanında hastalık teşhisi; uydu fotoğraflarının okunması ve belirlenmesi, strateji belirleme gibi askeri amaçlı uygulamalarda, kontrol problemlerinde, endüstriyel uygulamalarda) için geliştirilen bu teknik, günümüzde hemen her disiplinde kullanılır olmuştur. Yapay zeka olarak bilinen ve yapay sinir ağları; bulanık küme teorisi, genetik algoritmalar, uzman sistemler gibi çeşitli dalları olan mantıksal programlama tekniği geçen birkaç yıl içinde hemen her disiplinde yaygın olarak kullanılmaya başlanmıştır. Geçen zaman içinde yapay sinir ağları ve diğer mantıksal programlama tekniklerinin ispatlanmış teorilerinin ortaya çıkartılması nedeniyle konu pek çok teorisinin ilgisini çekmiştir.

Bilgisayar tekniğindeki bu gelişmeler teorik ve pratik çalışmalar yapan yapı mühendislerinin de ilgisini çekmiş ve yapay zeka genel adıyla bilinen yöntemler; bir çok tasarım parametresi arasından etkin çözüme ulaşma, taşıyıcı sistem seçme, standartların kontrolü, deprem riski altındaki yapıların hasar görebilirliği [6-10], zemin özelliklerinin belirlenmesi ve sınıflandırılması, şevlerde stabilite analizi, şantiye yönetimi ve organizasyonu, malzeme kalite kontrolü, şehir içi trafik sinyalizasyonu ve optimum sinyal süreleri belirlenmesi, üstyapı malzeme karışım hesabı, hava trafik kontrolü, ulaşım sistemlerinin planlanması, trafik akışının kontrolü ve kaza analizi, yol alt yapısı sırasında tahmini maliyet gibi uzman sistem; yapıların optimum boyutlandırılmasında genetik algoritmalar, betonun karmaşık davranışının modellenmesi, yapıların statik ve dinamik analizi, risk analizi, plak ve kabukların çözümü [12,14,15], depreme dayanıklı yapı tasarımı, hasar tespiti, köprülerin dinamik analizi gibi uygulamalarda yapay sinir ağları (YSA) uygulamaları, zemin sıvılaşması ve zemin parametrelerinin analizi, optimizasyon ve kontrol uygulamaları bu alanda yapılmış çalışmalardan bazılarıdır [17-24]. Tekniğin inşaat mühendisliğinde kullanım potansiyeli için geniş bir literatür Vanluchene[11] ve Hajela ve Leh [16], uygulama prensipleri ise Adeli ve diğerleri tarafından verilmiştir [1,2,6]. Bu uygulamalar daha çok malzeme modellenmesi, boyutlandırma, optimizasyon, hasar analizi, deprem mühendisliği gibi konuları içermektedir. Daha önce yapılan bu çalışmalarda daha çok geri-yayılma ve Hopfield sinir ağları kullanılmıştır. Yapılardaki hasarların belirlenmesi konusunda yapılan çalışmada yapay sinir ağları uygulaması hayli ümit verici olmuştur. Yapay sinir ağları boyutlandırma problemlerine başarıyla uygulanmış, elde edilen sonuçların klasik optimizasyon teknikleri kullanılarak elde edilen sonuçlardan daha elverişli olduğu gösterilmiştir [13].

2. YAPAY SİNİR AĞLARI (YSA)

Genel anlamda YSA, beynin bir işlevini yerine getirme yöntemini modellemek için tasarlanan bir sistem olarak tanımlanabilir [25,26]. YSA, yapay sinir hücrelerinin birbirleri ile çeşitli şekilde bağlanmasından oluşur ve genellikle katmanlar şeklinde düzenlenir. Bir başka tanımlamada da yapay sinir ağları, insan beyninin temel birimi olan nöronlara benzer olarak teşkil edilen yapay nöronların farklı topoloji ve ağ modelleriyle birbirine bağlanmasıyla oluşan karmaşık sistemlerdir. Bir yapay sinir ağı, bir biriyle etkileşim içindeki pek çok yapay nöronun paralel bağlı bir hiyerarşik organizasyondur [30]. Nöral hesaplama algoritmik programlamaya bir seçenek oluşturan, temel olarak yeni ve farklı bir bilgi işleme tekniğidir. Programda adım adım yürütülen bir yöntemin verilmesi yerine nöral

ağ ilişkilendirmeyi yapan iç kurallarını kendi üretir ve bu kuralları, sonuçları örneklerle karşılaştırarak düzenler [31]. Hücrelerin bağlantı şekillerine, öğrenme kurallarına ve aktivasyon fonksiyonlarına göre çeşitli YSA yapıları geliştirilmiştir. En genel anlamda yapay sinir ağları ileri beslemeli ve geri beslemeli ağlar şeklinde iki ana grupta düşünülebilir. İleri beslemeli ağlarda nöronlar; girdi, saklı ve çıktı olarak adlandırılan katmanlar vasıtasıyla organize edilir. Her bir katmandaki nöronlar; bir sonraki katman nöronları ile bağlantı ağırlıkları vasıtasıyla ilişkilidir. Ancak katmanların kendi aralarında her hangi bir bağlantı yoktur. Bilgi, girdi katmanından çıktı katmanına doğru ilerler. Buna aktivasyon yönü de denilir. Bu tür yapay sinir ağına örnek olarak tek ve çok katmanlı perceptron verilebilir. Bu tür ağlar denetimli öğrenme teknikleriyle eğitilir [32]. Geri beslemeli ağların en belirgin özelliği; katmanlar arasındaki nöronlar birbiriyle bağlantılı olup ayrıca bir dinamik hafızaya sahiplerdir. Bu ağlara örnek olarak kendi kendini düzenleyen öz örgütlenme haritası özelliğine sahip Kohonen [33] ağı ve ağ iç enerjisinin minimizasyonuna dayanan Hopfield [29] ağı verilebilir. Bu tür ağların eğitilmesi takviyesiz öğrenmeye bir örnek oluşturur. Hopfield ağları daha çok bir içerikli adreslenebilir bellek olarak veya optimizasyon tipi problemlerde başarılıdır. Yapay sinir ağlarındaki işleme elemanları biyolojik olarak insan beynindeki nöronlara karşılık gelmektedir (Şekil 1). Dendrit olarak adlandırılan yapı, diğer hücrelerden bilgiyi alan nöron girişleri olarak görev yapar. Diğer hücrelere bilgiyi transfer eden eleman aksonlardır. Dolayısıyla aksonlar nöron çıkışları olarak görev yaparlar. Akson ile dendrit arasındaki bağlantı ise sinapslar tarafından gerçekleştirilir [35].



Şekil 1. Biyolojik ve Yapay Nöron Modeli

Yapay sinir ağlarının işleyişi de buna benzer olarak gelişmektedir. 1940 yılında McCulloch ve Pitts nöronun, mantık sistemlerinde basit eş değer yapısıyla modellenebileceğini ortaya atmışlardır [34]. Bu amaçla yaptıkları çalışmalar sonunda Şekil 1’de altta görüldüğü gibi bir yapay nöron modeli geliştirmişlerdir. Bu modele göre, bir nöron N tane ağırlıklandırılmış girişi toplamakta ve sonucu lineer olmayan bir fonksiyondan geçirmektedir. Herhangi bir katmandaki j. birime gelen toplam giriş, önceki katmandaki birimlerin y_i çıkışlarının (ilk katman için girişlerin) bağlantılar üzerindeki w_{ij} ağırlıkları ile hesaplanmış bir ağırlıklı toplamıdır.

$$net_j = \sum_i w_{ij} y_j \quad (1)$$

birimin çıkışı, bu değer bir eşik değerden çıkartılıp lineer olmayan bir fonksiyondan geçirilmesiyle

$$o_j = f(net_j) = \frac{1}{1 + \exp(\sum_i w_{ij} y_j)} \quad (2)$$

olarak hesaplanır.

2.1. Yapay Sinir Ağlarında Öğrenme ve Ağırlık Eğitilmesi

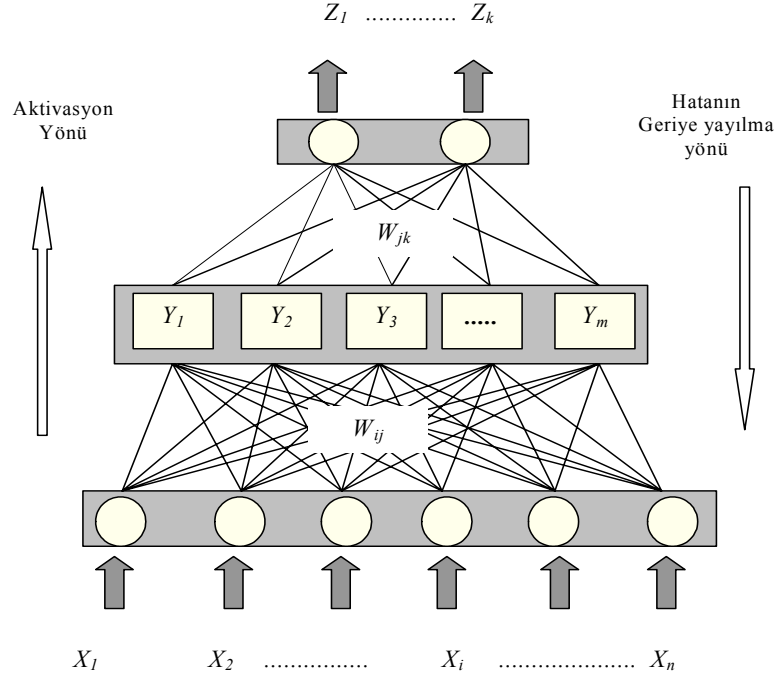
Yapay sinir ağları konvansiyonel programlama gibi belirli bir algoritma çerçevesinde programlanmazlar [32]. Sinir ağları insanlar gibi örnekler ile eğitilirler. Yapay sinir ağlarının öğrenmesi bir çocuğun öğrenmesi gibidir. Sıcak bir nesneye dokunmaması gerektiğini deneyerek öğrenen çocuklar zamanla daha az sıcak olan bir cisme dokunabilme cesaretini gösterirler ve sıcak süt dolu bardağı elleriyle tutarlar. Yani çocuk sıcaklık bilgisini öğrenmiş olmaktadır. Yapay nöronlar da benzer olarak; mevcut örnek kümesi üzerinde girdi ile çıktı arasındaki bağlantının ağırlıkların değiştirilmesiyle eğitilirler. Sunulan girdi kümesi için; transfer fonksiyonu tarafından sağlanan değerlere cevap olarak bağlantı ağırlıklarının tamamının veya bir kısmının istenen çıktı ile ağ çıktısı arasındaki farkın belirli bir değere düşüncüye kadar değiştirilmesidir. Günümüze kadar çeşitli öğrenme algoritmaları geliştirilmiştir. Bunlar temel olarak denetimli öğrenme, denetimsiz öğrenme ve takviyeli öğrenme olarak üç ana gruba ayrılır. Denetimli öğrenmede sinir ağına hem girdi hem de çıktı değerleri sunulur. Ağı ürettiği çıktı ile istenen çıktı arasındaki fark sıfır veya ona yakın bir değere gelinceye kadar ağırlıklar değiştirilir. Bu tür öğrenme modelini kullanan ağlar; perceptron ve ilişkili hafızalar, takviyeli öğrenme, stokastik öğrenme, vektör nicelik öğrenmesi, delta ve genelleştirilmiş delta kuralı, geri yayılma algoritması, bu grup öğrenmede kullanılan etkin metotlardır.

Denetimsiz öğrenmede ağa sadece girdi vektörü uygulanır. Girdi değerlerine uygun bir çıktı üretilinceye kadar bağlantı ağırlıkları değiştirilir. Bu metot görüntü işleme, işaret işleme ve kontrol problemlerinde etkin olarak kullanılır. Kohonen’in kendini düzenleyen uzaylar ve adaptif rezonans teorisi (ART) denetimsiz öğrenmeye örnek olarak verilebilir. Takviyeli öğrenmede ise giriş değerlerine karşılık gelecek uygun çıktıların elde edilmesi sırasında ağırlıkların en uygun değerlerinin bulunmasında genetik algoritmalar veya tabu en iyileme

yöntemleri kullanılır. Böylece ağırlıklar optimize edilmektedir. Bunların dışında hibrit öğrenme algoritmaları da geliştirilmiştir. Yapay sinir ağlarındaki ağırlıklar sabit değildir. Öğrenme kuralı giriş işaretlerine ve transfer fonksiyonu tarafından sağlanan değerlere cevap olarak, yerel bellekteki ağırlıkların hepsini veya bazılarını değiştiren bir denklem olarak bilinir. Öğrenme kuralı giriş işaretlerinin doğasına bağlı olarak işleme elemanının cevabının zamanla değişmesine imkân sağlar. Bu şekilde ağ, kendisini istenen cevaplara uyarlayabilir ve kendi içinde bilgiyi düzenleyebilir, yani öğrenir. Eğitim yeterli sayıda giriş-çıkış örnek kümesi için yapılmalıdır. Eğitim sonucunda ağ hem eğitime örnek kümesine ait veriyi tanımalı hem de interpolasyon ve ekstrapolasyon yapabilmelidir. Yani eğitim kümesinde olmayan örnek için uygun çıktılar vermelidir. Aslında eğitilmiş bir yapay sinir ağından beklenen de budur. Yapay sinir ağının eğitimi iki aşamalı bir işlemdir. İlk adımda, nöral ağa bir giriş vektörü uygulanarak, ağ içindeki her birim için toplam giriş ve çıkış hesaplanır ve her birimin çıkışı daha sonraki ağırlık düzenlenmesi için saklanır. Çıkış vektörü oluşturulduktan sonra, eğitime işlemi çıkış hatasını hesaplar ve bu hatanın geriye doğru yayılımını sağlar. Birimlerin ağırlıkları bu hata işaretine göre ayarlanır. Bu iki adımlı işlem ağ istenilen cevapları üretinceye kadar, eğitime vektör çiftleri ile tekrarlanır. Çıkış katmanını eğitmek kısmen daha basittir. Eğitimin nispeten zor olduğu aşama, ara katmanının eğitilmesidir. Bunu aşmak için, hata işaretinin çıkıştan daha iç katmanlara yayılımını sağlamakla bulunur. Yani, bir iç veya ara katman birimi kendisinden sonraki katmanın hata işaretlerinin ağırlıklı toplamı olan bir hata işareti alır. Bu hata işareti her birimin ağırlıklarını eğitmek için kullanıldıktan sonra, eğitime işlemi oluşturulan yeni hata işaretinin daha önceki katmanlara doğru yayılımını sağlar. Nöral ağların öğrenme mekanizması, girdi vektörü ve transfer fonksiyonu tarafından sağlanan değerlere karşılık olarak bağlantı ağırlıklarının hepsinin veya bir kısmının değiştirilmesidir.

2.2. Geriye Yayılma Öğrenme Algoritması

Yapay sinir ağlarında kullanılan öğrenme metotları en genel anlamda; denetimli öğrenme, denetimsiz öğrenme ve takviyeli öğrenme olarak üç ana gruptur. Sinir ağlarında öğrenme girdi ve çıktı kümesi için test edilen ağın, istenilen çıktıyı verecek ağırlık kümesinin belirlenmesidir. Bu ağırlık vektörü istenen çıktı ile ağ çıktısı arasındaki fark minimum oluncaya kadar ayarlanarak bulunur. Bu amaçla her bir öğrenme grubu için farklı öğrenme algoritmaları geliştirilmiştir. Geriye yayılma algoritması esasen çok katmanlı ağların eğitiminde kullanılan bir denetimli öğrenme algoritmasıdır. Çok katmanlı sinir ağları girdi ve çıktı katmanına ilaveten bir veya daha fazla saklı katmandan oluşur. Şekil 2’de bir adet saklı katmandan oluşan çok katmanlı bir ağ görülmektedir. Katmanlardaki nöronlar arasında ve bir katmandan önceki katmana geriye doğru bağlantı yoktur. Geriye yayılma algoritması bu tür ağlarda yaygın olarak kullanılan bir öğrenme algoritması olmasına rağmen daha çok geriye yayılma sinir ağları olarak tanınmıştır. Yayınma ve uyum gösterme olmak üzere iki aşamada işlemleri gerçekleştiren bu ağlar, katmanlar arasında tam bir bağlantının bulunduğu çok katmanlı, ileri beslemeli ve denetimli olarak eğitilen bir YSA modelidir. Geri yayılmalı öğrenme kuralı ağ çıkışındaki mevcut hata düzeyine göre her bir tabakadaki ağırlıkları yeniden hesaplamak için kullanılmaktadır. Çalışmada genelleştirilmiş delta kuralı kullanılmıştır. Genelleştirilmiş delta kuralı özellikle çok katmanlı ağların eğitiminde kullanılır [36].



Şekil 2. Tek Saklı Katmana Sahip Bir Geri-Yayılma Ağ Mimarisi

Herhangi bir k. Girdi-çıkı örnek çifti için j. girdi ile i. nöron arasındaki ağırlık değişimi;

$$\Delta w_{ij}^k = \alpha (T_i^k - O_i^k) x_j^k \quad (3)$$

ile ifade edilir. Burada T_i istenen hedef çıktı, O_i gerçek çıktı, α öğrenme oranı ve x_j ise j. ağ girdisidir. Herhangi bir k. tabakadaki örnek çifti için toplam hata ve ortalama hata fonksiyonu [1,31]

$$E = \frac{1}{2} \sum_k (T_k - O_k)^2, \quad E = \frac{1}{2P} \sum_p \sum_j (T_j - O_j)^2 \quad (4,5)$$

olarak tanımlanır. Burada, T_k istenen hedef çıktı, O_k hesaplanan çıktı, P ise eğitim setinde bulunan girdi-çıkı çiftlerinin toplam sayısıdır. Yine Rumelhart tarafından, ağın

yakınsamaması durumunda β momentum terimini de içerecek şekilde ağırlıkların ayarlanması formülü aşağıdaki şekilde genelleştirilmiştir [32,36].

$$\Delta w_{ij}(t+1) = \alpha \delta_i^k o_j^k + \beta \Delta w_{ij}(t) \quad (6)$$

Burada β momentum terimi, δ_i^k eğitim setindeki k . örnek çifti için i . nöronun hata değeridir ve kullanılan eşik fonksiyonunun türevini içeren bir terimdir. Bu nedenle eşik fonksiyonu ağırlıklandırılmış girdi toplamları ile hesaplanmış net girdiyi alarak, işlem elemanlarının çıktısını belirleyen ve türevlenebilir olması tercih sebebi olan bir fonksiyondur. Aktivasyon veya işaret fonksiyonu olarak da adlandırılan eşik fonksiyonları pek çok sayıda olmasına rağmen geriye yayılma algoritmasında daha çok sigmoid fonksiyon kullanılmaktadır.

3. DİKDÖRTGEN PLAKLARIN TİTREŞİMİ ve YSA'nın EĞİTİMİ

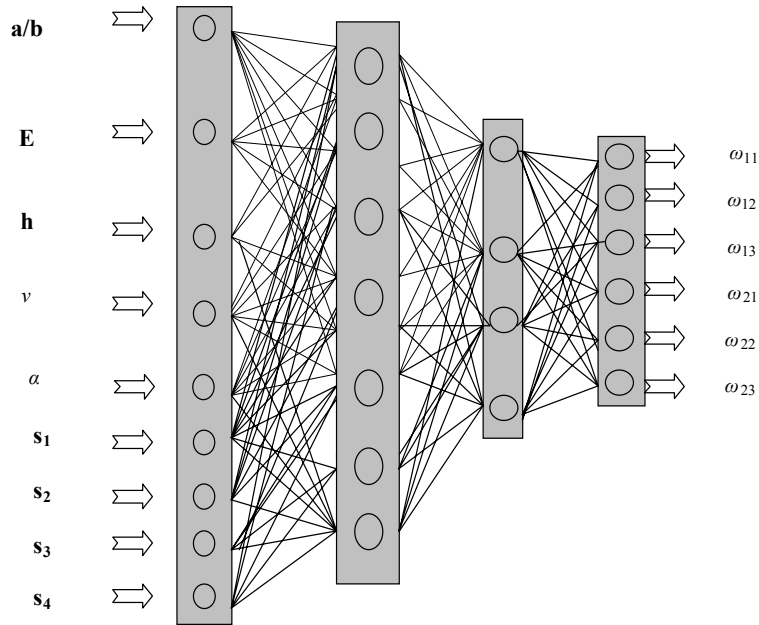
İnce ve izotrop dikdörtgen bir plağın eğilme titreşimine ait genel denklem [37];

$$D \nabla^4 w + \rho \frac{\partial^2 w}{\partial t^2} = 0 \quad (7)$$

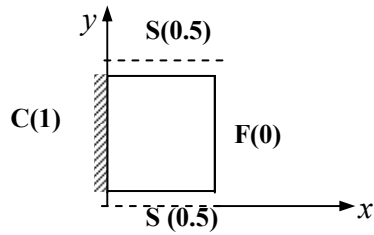
ile tanımlıdır. Bu denklemin analitik çözümü bazı basit durumlar dışında yoktur. Bu nedenle günümüze kadar sonlu farklar, sonlu elemanlar, sınır elemanlar, Ritz, Galerkin ve diferansiyel quadrature gibi pek çok sayısal yöntem geliştirilmiştir. Bu çalışmada, dikdörtgen plakların titreşim frekanslarını elde etmek için geliştirilen geri-yayılma yapay sinir ağı modeli Şekil 3'te verilmiştir. Çalışmada 21 farklı mesnet koşulu (Tablo 1) dikkate alınmıştır. Ağ; bir girdi katmanı, iki saklı katman ve bir çıktı katmanından oluşur. Girdi katmanında 9 adet yapay nöron kullanılmıştır. İki adet saklı katman olup, birinci katmanda 7, ikinci katmanda 4 adet yapay nöron kullanılmıştır. Çıktı katmanında ise elde edilmek istenen 6 adet mod dizisi için 6 adet yapay nöron kullanılmıştır. Ağ yapısı kısaca 9-(7-4)-6 olarak ifade edilebilir. Girdi ve çıktılar sabit olup ara katman nöron sayıları deneme-yayılma ile bulunmuştur. Girdi katmanında kullanılan değişkenler a/b plak kenar boyutlarının oranı, E elastisite modülü, h , plak kalınlığı, ν Poisson oranı, α ise $\alpha = m\pi/a$ şeklinde tanımlanan bir parametre olup, burada $m=1,2,\dots$ şeklinde ardışık tamsayı olup, bu çalışmada sadece 1 ve 2 değerlerini almaktadır. Mesnet koşullarını ağa tanıtmak için s_1, s_2, s_3 ve s_4 kullanılmış olup her bir mesnet durumu için bu değişkenlerin alacağı değerler Tablo 2'de verilmiştir. Çünkü, Şekil 3'te çıktı katmanından görüleceği üzere sadece ω_i . ($i=1,2,3$ olmak üzere) modlar hesaplanmaktadır. Tablo 1'de çalışmada dikkate alınan 21 farklı mesnet koşuluna sahip plak görülmektedir. Bunlardan 11 tanesi ağın eğitiminde kullanılan mesnet koşulu, diğer 10 tanesi test aşamasında kullanılmış olup, bu plaklar tabloda (*) ile işaretlenmiştir. Eğitim setinde en fazla Ankastre (C) ve Serbest(F) mesnet koşuluna sahip plaklar seçilmiştir. Böylece ağın diğer mesnetlerle birlikte özellikle basit mesnet koşulunu öğrenip öğrenmediği kontrol edilmiştir. Ağın eğitimi ile ilgili önemli bir diğer kavram, eğitim aşamasında plak kenar boyutları a/b 'nin 0.3, 0.7 ve 1 değerleri kullanılmıştır. Test aşamasında $a/b = 0.4$ ve 1.5 değerleri için ağ test edilmiştir. Tablo 2'de ağın eğitimi ve test

İnce Dikdörtgen Plakların Titreşim Frekanslarının Yapay Sinir Ağları Yaklaşımı...

aşamasında plak 4 kenarına ait mesnet koşuluna ait girdi katmanında bulunan ve mesnet koşullarını ağı tanıtmak için kullanılan s_1 , s_2 , s_3 ve s_4 değerleri verilmiştir. Bu tablo hazırlanırken Ankastre (C) mesnet için 1, Basit (S) mesnet durumu için 0.5 ve Serbest (F) mesnet koşulu için 0 katsayısı kullanılmıştır. Bu kodlama, geri-yayılma ağlarında girdi-çıkış setinin 0 ile 1 arasında normalize edilmesi zorunluluğu açısından da uygundur. Bu kodlama yapılırken Şekil 4'te görülen plak geometrisi ele alınmıştır ve kodlamaya sol dikey kenardan başlanarak saat yönünün tersine devam edilmiştir. Bu kodlamayı anlamak için örnek alınmak üzere Tablo 1'de Durum 19 olarak verilen plak mesnet koşulu Şekil 4'de dikkate alınmıştır. Bu plak mesnet durumu CSFS olarak kodlanır ve yukarıda anlatılan ve Tablo 2'de özetlenen ve bu koşula karşı gelen sayısal kod 1;0.5;0;0.5 olmaktadır.

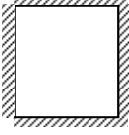
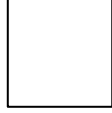
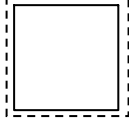
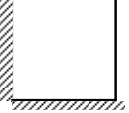
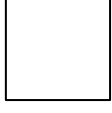
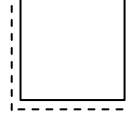
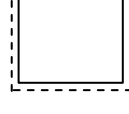
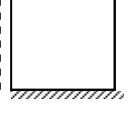
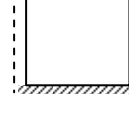
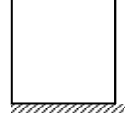


Şekil 3. Plak titreşim frekansı için geliştirilen yapay sinir ağı modeli

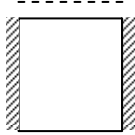
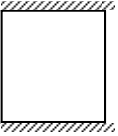
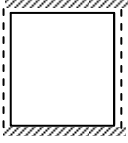
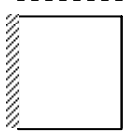
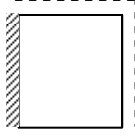


Şekil 4. Plak mesnet koşullarının gösterimi

Çizelge 1. Çalışmada dikkate alınan 21 farklı plak türü

Ankastre Mesnet (C)	Serbest Mesnet (F)	Basit Mesnet (S)
 Durum 1: CCCC	 Durum 8: FFFF	 Durum 15: SSSS (*)
 Durum 2: CCFF (*)	 Durum 9: SFSF (*)	 Durum 16: SSSF (*)
 Durum 3: CCCS	 Durum 10: CFFF (*)	 Durum 17: SSFF(*)
 Durum 4: CCSF	 Durum 11: SCSF(*)	 Durum 18: SCSS (*)
 Durum 5: CCSS	 Durum 12: FCFF	 Durum 19: CSFS

Çizelge 1. (Devam)

		
Durum 6: CFCS	Durum 13: FCFC	Durum 20: SCSC(*)
		
Durum 7: CFFS	Durum 14: CSFF(*)	Durum 21: CFSS

Çizelge 2. Mesnet koşulları için ağ girdi değerleri (s_1, s_2, s_3 ve s_4)

Plak mesnet koşulu	Karşılık gelen ağ girdisi (s_1, s_2, s_3, s_4)
Durum 1: CCCC	(1;1;1;1)
Durum 2: CCCF	(1;1;1;0)
Durum 3: CCCS	(1;1;1;0.5)
Durum 4: CCSF	(1;1;0.5;0)
Durum 5: CCSS	(1;1;0.5;0.5)
Durum 6: CFCS	(1;0;1;0.5)
Durum 7: CFFS	(1;0;0;0.5)
Durum 8: FFFF	(0;0;0;0)
Durum 9: SFSF	(0.5;0;0.5;0)
Durum 10: CFFF	(1;0;0;0)
Durum 11: SCSF	(0.5;1;0.5;0)
Durum 12: FCFF	(0;1;0;0)
Durum 13: FCFC	(0;1;0;1)
Durum 14: CSFF	(1;0.5;0;0)
Durum 15: SSSS	(0.5;0.5;0.5;0.5)
Durum 16: SSSF	(0.5;0.5;0.5;0)
Durum 17: SSFF	(0.5;0.5;0;0)
Durum 18: SCSS	(0.5;1;0.5;0.5)
Durum 19: CSFS	(1;0.5;0;0.5)
Durum 20: SCSC	(0.5;1;0.5;1)
Durum 21: CFSS	(1;0;0.5;0.5)

4. SAYISAL SONUÇLAR

Sayısal örnekler için öğrenme oranı başlangıçta 0.7 olarak alınmış ve bu çalışma neticesinde öğrenme oranı 0.82 olarak uygun bulunmuştur. Momentum terimi 0.5 olarak seçilmiş ve eğitim sırasında bu terimin 0.79 civarında olması halinde ağın yakınsadığı görülmüştür. Çalışmada; eğitim safhasında maksimum hata % 4.7, test esnasında % 3.1 olarak elde edilmiştir. Eğitim aşamasından sonra ağın yakınsayıp yakınsamağının kontrolü için dört kenarından basitçe oturmuş SSSS mesnet koşulu için ağ test edilmiştir. Elde edilen sonuçlar Leissa [37] tarafından verilen nümerik sonuçlar ile birlikte karşılaştırmalı olarak Tablo 3’de verilmiştir. Geliştirilen ağ, uygun sonuçlar verdiği için geliştirilen ağın eğitimi tamamlanmıştır. Bu aşamada Poisson oranı $\nu=0.3$, $D=1$, $a=1$ ve $p=1$ için Tablo 4 ve Tablo 5’deki sonuçlar geliştirilen YSA tarafından üretilmiştir. Tablo 4’te plak kenar boyutları oranı (a/b) 0.4, Tablo 5’te ise $a/b=1.5$ için sonuçlar verilmektedir. Bu sonuçlar için elde edilen maksimum hata %3.8 olmuştur. Öğrenme oranı ve momentum terimine bağlı olarak hata miktarının değişimi yaklaşık olarak Şekil 5 ve Şekil 6’da verilmiştir. Görüldüğü gibi öğrenme oranı 0.82 ve momentum teriminin 0.79 olduğu durum hatanın minimum olduğu durumdur.

Çizelge 3. $a/b = 1$ için YSA ile elde edilen sonuçların karşılaştırılması

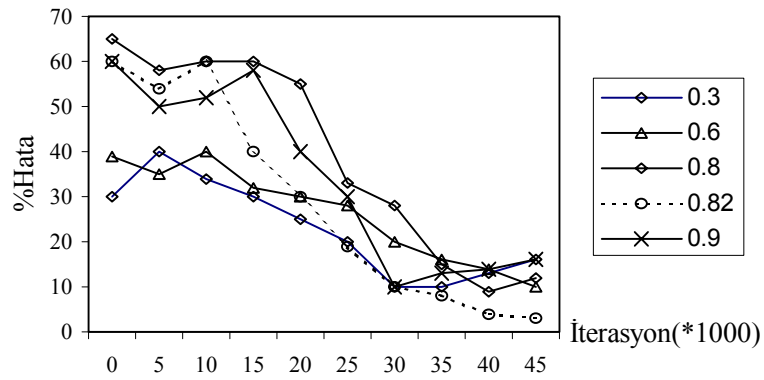
Mesnet Koşulu	SSSS					
	ω_{11}	ω_{12}	ω_{13}	ω_{21}	ω_{22}	ω_{23}
YSA	20.12	47.85	101.32	51.36	81.58	130.32
Ref. 37	19.73	49.35	98.29	49.35	78.96	128.31

Çizelge 4. $a/b = 0.4$ için YSA ile elde edilen sonuçlar

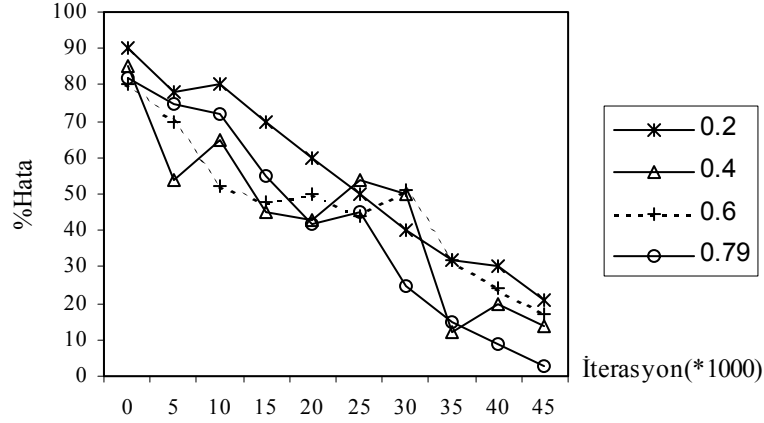
Mesnet Koşulu	$a/b = 0.4$					
	ω_{11}	ω_{12}	ω_{13}	ω_{21}	ω_{22}	ω_{23}
SSSS	11.65	16.26	25.31	42.96	46.85	52.63
SCSC	13.21	19.32	28.85	42.03	48.11	57.16
SCSS	12.32	18.95	26.34	39.41	44.36	56.20
SCSF	11.10	14.52	21.65	40.63	44.02	51.85
SSSF	10.96	14.02	20.69	41.36	44.30	49.67
SFSF	9.96	12.90	16.54	41.65	43.38	46.81
CCFF	4.11	7.25	12.54	23.12	27.54	31.52
CSFF	3.99	6.67	12.85	22.69	25.34	32.96
CFFF	3.61	4.86	8.45	22.04	24.12	28.69
SSFF	1.35	4.81	10.56	16.32	21.01	25.53

Çizelge 5. $a/b = 1.5$ için YSA ile elde edilen sonuçlar

Mesnet Koşulu	$a/b = 1.5$					
	ω_{11}	ω_{12}	ω_{13}	ω_{21}	ω_{22}	ω_{23}
SSSS	33.58	100.12	211.58	62.28	130.47	241.05
SCSC	57.89	148.80	311.82	80.24	172.25	334.96
SCSS	43.22	122.36	244.48	71.17	148.33	303.40
SCSF	17.38	62.04	151.74	46.68	93.35	182.65
SSSF	14.85	49.80	126.19	44.65	83.57	161.13
SFSF	9.75	23.54	68.75	40.01	56.60	105.41
CCFF	12.26	53.45	71.90	31.19	79.63	128.04
CSFF	7.12	40.29	75.69	26.04	65.58	96.78
CFFF	3.67	12.05	56.87	23.08	41.20	61.25
SSFF	5.11	38.65	59.60	22.66	61.85	76.89



Şekil 5. Öğrenme oranına bağlı hata değerleri (50 .000 iterasyon sonucu)



Şekil 6. Momentum terimine bağlı hata değerleri (50 .000 iterasyon sonucu)

5. SONUÇLAR VE TARTIŞMA

Çalışmada, yapay sinir ağları kullanılarak dikdörtgen plakların titreşim analizi sunulmuştur. Yapay sinir ağları eğitim aşaması bittikten sonra klasik programlama tekniklerinden hız ve gerektirdikleri kapasite açısından bazı avantajlara sahiptir. Ancak sayısal analiz teknikleri hala uygulama potansiyelinin geniş ve araştırmacılar tarafından tercih edilir olması nedeniyle popülaritesini korumaktadır. Ülkemizde on yıllık bir geçmişi olan bu konuda her geçen gün yeni bir çalışma yapılmakta ve metodun kullanılabilirliği ve üstünlüğü zaman zaman kendisini göstermektedir. Buna ilaveten deney sonuçları ile ağı eğitilmesi deneysel çalışmalar ile birlikte kullanılması halinde sonuçların bir hayli sağlıklı olacağını düşündürmektedir. Tekniğin klasik programlamadan üstün olacağını söylemek ise her zaman mümkün değildir. Ancak klasik programlama ile çok uzun sürede çözülebilecek problemler daha hızlı ve eğitim tamamlandıktan sonra daha seri bir şekilde çözülebilir.

Yapılan çalışma; ön boyutlandırma problemlerinde, frekans kısıtlayıcısı altında optimizasyon tipi problemlerde yada ön tasarım gerektiren mühendislik ve mimari tasarımlarda kullanılabilecektir. Ağ topolojisinde yapılacak küçük değişiklikler ile farklı geometri ve malzeme özelliklerine sahip yapı sistemlerinin titreşim hesabı yapılabileceğinden çalışma daha kapsamlı uygulamalar için geliştirilebilir. Esasında yapay sinir ağlarının en önemli ve ayırt edici özelliklerinden birisi de; yeni durumlara adaptasyon ve hiyerarşik yapılarının esnek olması ve sonradan veri girişini kabul etmeleridir. Kullanılan metot ve yapılan kabul ne olursa olsun sayılar mühendise tasarım aşamasında sadece bir fikir vermek için vardır ve yapının gerçekteki davranışının tam olarak anlaşılması ise daha da önemlidir.

Teşekkür

Birinci yazar; yüksek lisans tez çalışmasında geliştirdiği programın güncellendirilmesi süresince, yaptığı değerli katkılardan dolayı, İnş. Müh. Ali Kemal BALTACIOĞLU'na teşekkür eder.

Kaynaklar

- [1] Adeli, H., Hung, S.L., Machine learning- neural networks, genetic algorithms and fuzzy systems, John Wiley & Sons, Inc., 1995.
- [2] Adeli, H., Yeh, C. Perceptron learning in engineering design, Microcomputer in Civil Eng., 4: 247-256, 1989.
- [3] Aleksander, I., Morton, I., An introduction to neural computing, International Thomson Computer Press., 1995.
- [4] Park, H.S. and Adeli, H., Distributed Neural Dynamics Algorithms For Optimization Of Large Steel Structures, J. of Struc. Engin., ASCE, 123(7), 880-888, 1997.
- [5] Zurada, J M., Introduction to artificial neural networks, West Publishing Com., 1992.
- [6] Adeli, H., and Park, H.S., A neural dynamics model for structural optimization-theory, Computers and Structures, 57(3), 383-390, 1995.
- [7] Chen, H.M., Qi, G.Z., Yang, J.C.S., and Amini, F., Neural Networks for Structural Dynamic Model Identification, J. of Eng. Mech., ASCE, 121(12), 1377-1381, 1995.
- [8] Ghaboussi, J., An overview of the potential applications of neural networks in civil engineering", in Proc., ASCE Structures Cong.'93, Irvine, California, 1993.
- [9] Hani, K.B. and Ghaboussi, J. Neural Networks For Structural Control Of A Benchmark Problem, Active Tendon System, Earthquake Eng. and Structural Dynamics, 27, 1225-1245, 1998.
- [10] Szewczyk, Z.P., Hajela, P., Damage detection in structures based on feature-sensitive neural networks, J Computing Civil Eng., ASCE, 8(2), 163-178, 1994.
- [11] Vanluchene, RD., and Roufei, S., Neural networks in structural engineering, Microcomputers in Civil Eng., 1990; 5:207-215.
- [12] Wu, X., Ghaboussi, J., Garrett, J.H., Use of neural networks in detection of structural damage, Computers & Structures, 42(4), 649-659, 1992.
- [13] Kang, H.T., Yoon, C J., Neural networks approaches to aid simple truss design problems, Microcomputers in Civil Eng., 9, 211-218, 1994.
- [14] Ghaboussi, J., Garrett, Jr., Wu, X., Knowledge- based modeling of material behavior with neural networks, Journal of Structural Engineering, ASCE, 117, 1, 132-153, 1991.

- [15] Ghaboussi, J., Lin, C.C., New method of generating spectrum-compatible accelerograms using neural networks, *Earth. Eng. Structural Dynamics*, 27,377-396, 1998.
- [16] Hajela, P., Berke, L., Neurobiological computational models in structural analysis and design, *Computers and Structures*, 41(4), 657-67, 1991.
- [17] Civallek, Ö., The design of structures under earthquake effects by using neuro-fuzzy method., *Fourth National Earthquake Engineering Conferences*, 17-19 September, Ankara, 431-38, 1997.
- [18] Civallek, Ö., Plak ve Kabukların Nöro-Fuzzy Tekniği ile Lineer ve Lineer Olmayan Statik-Dinamik Analizi, *Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ*, 1998.
- [19] Civallek, Ö., The analysis of the rectangular plates without torsion via hybrid artificial intelligent technique, *Proceedings of the Second International Symposium on Mathematical & Computational Applications*, September 1-3, Azerbaijan,95-101, 1999.
- [20] Civallek, Ö., Dikdörtgen Plakların Nöro-Fuzzy Tekniği ile Analizi, III. Ulusal Hesaplamalı Mekanik Konferansı, 16-18 Kasım, İstanbul, 517-525; 1998.
- [21] Civallek, Ö., Nöro- Fuzzy Tekniği ile Dairesel Plakların Analizi, *Journal of Eng. Science of Dokuz Eylül University*, Vol. 1(2); 13-31,1999.
- [22] Civallek, Ö., Çatal, H.H., Geriye Yayılma Yapay Sinir Ağı Kullanılarak Elastik Kirişlerin Statik ve Dinamik Analizi, *Dokuz Eylül Üniversitesi Fen ve Mühendislik Dergisi Cilt 6(1)*, 1-17,2004.
- [23] Civallek, Ö., Çatal, H.H., ve Demirdağ, O., Geriye Yayılma Yapay Sinir Ağları ile Düzlem Çerçeve ve Kirişlerin Titreşim Frekanslarının Tahmini, *İMO Teknik Dergi*, Cilt 13(3), 2709-2726,2002.
- [24] Civallek, Ö., Çatal, H.H., Geriye Yayılma Sinir Ağı Kullanılarak Elastik Kirişlerin Eğilme Titreşimlerinin Yaklaşık Hesabı, *TMMK, Ulusal Mekanik Kongresi*, Eylül, Konya Selçuk Üniversitesi, 2001.
- [25] Civallek, Ö., The analysis of time dependent deformation in R.C. Members by artificial neural network, *Journal of Eng. Sciences of Pamukkale Univ.*, 1997,3(2),331-335
- [26] Civallek, Ö., Elastik Zemin Oturan Kirişlerin Nöro-Fuzzy Tekniği ile Analizi, 7th. National soil mechanics and foundation engineering conferences, 22-23 October, Yıldız Univ., İstanbul, 1998.
- [27] Civallek, Ö., Flexural And Axial Vibration Analysis of Beams With Different Support Conditions Using Artificial Neural Networks, *International Journal of Structural Engineering and Mechanics*, 18 (3),303-314, 2004.
- [28] Ülker, M., Civallek, Ö., “The Buckling Analysis Of Axially Loaded Columns With Artificial Neural Networks”, *Turkish J. Eng. Env. Sci.*, TUBITAK, 26, 117-125,2002.

- [29] Hopfield, JJ., Neural networks and physical systems with emergent collective computational abilities., In Proc. of National Academy of Sciences,79: 2554-58, 1982.
- [30] Eberhart, R. C., and Dobbins, R. W., Neural network PC tools , Academic Press, San Diego, California,1990.
- [31] Fausett, L., Fundamentals of neural networks, architectures, algorithms, and applications, Prentice-Hall, Inc., New-Jersey, 1994.
- [32] Fu, LM, Neural Networks in Computer Intelligence, McGraw-Hill, Inc. New York.,1994.
- [33] Kohonen, T., Associative memory: a system-theoretical approach, Spring-Verlag, New York., 1977.
- [34] McCullogh, WS., and Pitts, W., A logical calculus of ideas imminent in nervous activity., Bull. Math. Biophysics, 5, 115-33, 1943.
- [35] Rojas, R., Neural networks, A systematic introduction., Springer, Germany,1996.
- [36] Rumelhart, DE., Hinton, GE., and Williams, R J., Learning internal representation by error propagation. in parallel distributed processing : Explorations in the microstructures of cognition, MIT Press, Cambridge, MA., 1986.
- [37] Leissa, The free vibration of rectangular plates, Journal of Sound and vibration, 31(3), 257-293, 1973.