

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**TANKER ŞAMANDIRA BAĞLAMA SİSTEMLERİNİN YAPAY SİNİR
AĞLARI TEKNİĞİYLE OPTİMİZASYONU**

YÜKSEK LİSANS TEZİ

Murat YETKİN

Gemi ve Deniz Teknolojisi Mühendisliği Anabilim Dalı

Gemi ve Deniz Teknolojisi Mühendisliği Programı

ARALIK 2014

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**TANKER ŞAMANDIRA BAĞLAMA SİSTEMLERİNİN YAPAY SİNİR
AĞLARI TEKNİĞİYLE OPTİMİZASYONU**

YÜKSEK LİSANS TEZİ

**Murat Yetkin
(508121103)**

Gemi ve Deniz Teknolojisi Mühendisliği Anabilim Dalı

Gemi ve Deniz Teknolojisi Mühendisliği Programı

Tez Danışmanı: Yrd. Doç. Dr. Ayhan MENTEŞ

ARALIK 2014

İTÜ, Fen Bilimleri Enstitüsü'nün 508121103 numaralı Yüksek Lisans **Murat YETKİN**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “**TANKER ŞAMANDIRA BAĞLAMA SİSTEMLERİNİN YAPAY SİNİR AĞLARI TEKNİĞİYLE OPTİMİZASYONU**” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Yrd. Doç. Dr. Ayhan MENTEŞ**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Yrd. Doç. Dr. Ayhan MENTEŞ**
İstanbul Teknik Üniversitesi

Prof. Dr. İsmail H. Helvacıoğlu
İstanbul Teknik Üniversitesi

Yrd. Doç. Dr. Uğur Buğra ÇELEBİ
Yıldız Teknik Üniversitesi

Teslim Tarihi : **15 Aralık 2014**
Savunma Tarihi : **26 Aralık 2014**

Kardeřlerim Yasin ve Yasemin'e,

ÖNSÖZ

Tez çalışmam sırasında bilgi ve deneyimini benimle paylaşarak destek olan danışmanım sayın Yrd. Doç. Dr. Ayhan MENTEŞ'e yardımları için teşekkür ederim.

Tez süresince bilgi ve desteklerini esirgemeyen araştırma ofisindeki arkadaşlarıma teşekkürlerimi sunarım. Yüksek lisans çalışmalarım sırasında burs desteği sağlayan Türk Loydu Vakfı'na minnettarlığımı belirtmek isterim.

Çalışmalarım sırasında bana her zaman destek olan kardeşlerim Yasemin YETKİN ve Yasin YETKİN'e çok teşekkür ederim.

Aralık 2014

Murat YETKİN
Gemi ve Deniz Teknolojisi Müh.

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	vii
İÇİNDEKİLER	ix
KISALTMALAR	xi
ÇİZELGE LİSTESİ.....	xiii
ŞEKİL LİSTESİ.....	xv
ÖZET.....	xix
SUMMARY	xxi
1. GİRİŞ	1
2. YAPAY SİNİR AĞLARI.....	3
2.1 Genel	3
2.2 Yapay Sinir Ağları Tarihçesi.....	4
2.3 Kullanım Alanları.....	5
2.3.1 Sınıflandırma	5
2.3.2 Tahmin	5
2.3.3 Desen tanıma	5
2.3.4 Fonksiyon yaklaşımı	6
2.3.5 Optimizasyon	6
2.4 YSA'nın Özellikleri	6
2.4.1 Doğrusal olmama	6
2.4.2 Öğrenme	7
2.4.3 Genelleme.....	7
2.4.4 Uyarlanabilirlik	7
2.4.5 Hata toleransı.....	7
2.4.6 Hız ve tasarım kolaylığı	8
2.5 YSA'nın Yapısı	8
2.5.1 Girdi katmanı.....	9
2.5.2 Ara katman	9
2.5.3 Çıktı katmanı	9
2.6 YSA'da İşlem Elemanları	9
2.6.1 Girdiler	9
2.6.2 Ağırlıklar	9
2.6.3 Birleştirme fonksiyonu	9
2.6.4 Aktivasyon fonksiyonu	10
2.6.4.1 Doğrusal aktivasyon fonksiyonu.....	10
2.6.4.2 Sigmoid aktivasyon fonksiyonu	10
2.6.4.3 Tanjant hiperbolik aktivasyon fonksiyonu.....	11
2.6.4.4 Diğer aktivasyon fonksiyonları	11
2.6.5 Çıktı	11
2.7 YSA Öğrenme Türleri ve Öğrenme Kuralları	12
2.7.1 Öğrenme türleri	12
2.7.1.1 Eğitimli öğrenme.....	12

2.7.1.2 Eğitimsiz öğrenme.....	13
2.7.1.3 Yarı eğitilmiş öğrenme	13
2.7.2 Öğrenme kuralları.....	13
2.7.2.1 Hebb kuralı	13
2.7.2.2 Hopfield kuralı	13
2.7.2.3 Kohonen kuralı	13
2.7.2.4 Delta kuralı	13
2.7.2.5 Levenberg-Marquart algoritması.....	14
2.7.2.6 Geri yayılım algoritması.....	15
2.8 Ağ Yapılarına göre YSA	15
2.8.1 İleri beslemeli yapay sinir ağları	15
2.8.2 Geri beslemeli yapay sinir ağları	15
2.9 Katmanlarına Göre YSA	16
2.9.1 Tek katmanlı algılayıcılar	17
2.9.1.1 Basit algılayıcı (Perceptron).....	17
2.9.1.2 Yinelemeli algılayıcı (ADALINE).....	17
2.9.1.3 Çoklu yinelemeli algılayıcı (MADALINE)	17
2.9.2 Çok katmanlı algılayıcılar	17
2.10 Yapay Sinir Ağlarının Avantajları ve Dezavantajları	18
2.10.1 YSA'ların avantajları.....	18
2.10.2 YSA'ların dezavantajları	19
2.11 YSA Yazılımları.....	19
2.11.1 MatLab: Neural network toolbox	20
2.11.2 NeuroSolutions yazılımı	22
3. TANKER-ŞAMANDIRA BAĞLAMA SİSTEMLERİ.....	25
3.1 Tanker-Şamandıra Bağlama Sistemi Dizaynı	26
3.1.1 Tanker-Şamandıra bağlama sistemi genel karakteristikleri	28
3.2 OrcaFlex Programı	29
3.2.1 OrcaFlex programı simülasyon testleri	31
4. TANKER-ŞAMANDIRA BAĞLAMA SİSTEMLERİNİN YAPAY SİNİR AĞLARI TEKNİĞİYLE OPTİMİZASYONU	35
4.1 Tanker-Şamandıra Bağlama Sistemi Modellemesi ve Simülasyonu.....	35
4.2 YSA Tekniğiyle Optimizasyon	39
5. SONUÇ VE ÖNERİLER.....	47
6. KAYNAKLAR.....	49
ÖZGEÇMİŞ.....	53

KISALTMALAR

ADALINE	: Adaptive Linear Element
ANN	: Artificial Neural Network
ART	: Adaptive Resonance Theory
FANN	: Fast Artificial Neural Network
HYSA	: Hücresel Yapay Sinir Ağı
LM	: Levenberg-Marquardt Algoritması
MADALINE	: Multi Adaptive Linear Element
NNT	: Neural Network Toolbox
WN	: Wavelet Networks
YSA	: Yapay Sinir Ağları

ÇİZELGE LİSTESİ

Sayfa

Çizelge 2.1 : Bazı YSA yazılımları.	21
Çizelge 3.1 : Tankerin genel özellikleri.	28
Çizelge 3.2 : Tanker için simülasyon senaryo aralıkları.....	29
Çizelge 3.3 : Deneme simülasyonu model özellikleri.	32
Çizelge 4.1 : OrcaFlex modeli ana boyutları.	36
Çizelge 4.2: OrcaFlex simülasyonları için çevre şartları ve bağlama konumları.....	37
Çizelge 4.3: Simülasyon sonuçları	38
Çizelge 4.4: Çevre şartları ve bağlama konumlarının normalizasyon sonuçları	40
Çizelge 4.5: Simülasyon sonuçları normalize edilmiş değerleri	41
Çizelge 4.6: Optimum gerilmenin bulunduğu şartlar.....	45

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : Nöronun yapısı.	3
Şekil 2.2 : YSA'nın temel yapısı.	8
Şekil 2.3 : Doğrusal aktivasyon fonksiyonu	10
Şekil 2.4 : Sigmoid aktivasyon fonksiyonu	11
Şekil 2.5 : Tanjant hiperbolik aktivasyon fonksiyonu	11
Şekil 2.6 : YSA mimarisi.....	12
Şekil 2.7 : İleri beslemeli yapay sinir ağı	16
Şekil 2.8 : Çok katmanlı YSA	18
Şekil 2.9 : Neural network toolbox arayüzü	20
Şekil 2.10 : Neural network toolbox arayüzü	21
Şekil 2.11 : Neural network toolbox eğitim bölümü	22
Şekil 2.12 : NeuroSolutions programı	23
Şekil 3.1 : Tek noktadan bağlı sistem.....	25
Şekil 3.2 : Çok noktadan bağlı sistem	25
Şekil 3.3 : Tasarlanan tanker-şamandıra bağlama sistemi.....	27
Şekil 3.4 : Global ve yerel koordinat sistemi.....	27
Şekil 3.5 : Baş iskele halatında rüzgar hızına bağlı gerilme grafiği.	32
Şekil 3.6 : Baş sancak halatında rüzgar hızına bağlı gerilme grafiği.....	33
Şekil 4.1 : Şamandıra görünümü.	35
Şekil 4.2 : İlk bağlama noktası ($l=0$).	36
Şekil 4.3 : YSA eğitim sonuçları.	42
Şekil 4.4 : Baş sancak halatı sonuçlarının karşılaştırılması.	43
Şekil 4.5 : Baş iskele halatı sonuçlarının karşılaştırılması.	43
Şekil 4.6 : Kıç iskele halatı sonuçlarının karşılaştırılması.....	43
Şekil 4.7 : Kıç sancak halatı sonuçlarının karşılaştırılması.	44
Şekil 4.8 : x-yönündeki yer değiştirme sonuçlarının karşılaştırılması.	44
Şekil 4.9 : y-yönündeki yer değiştirme sonuçlarının karşılaştırılması.	45

SEMBOL LİSTESİ

T1	: Baş sancak halatı (kN)
T2	: Baş iskele halatı (kN)
T3	: K1ç iskele halatı (kN)
T4	: K1ç sancak halatı (kN)
x	: x yönündeki yer değıştirme miktarı (m)
y	: y yönündeki yer değıştirme miktarı (m)
en_a	: Akıntı, dalga ve rüzgarın yönü (ačısı) (derece),
h	: Dalga yüksekliğı (m),
w	: Rüzgar hızı (m/s),
c	: Akıntı hızı (m/s),
t	: Dalga periyodu (s),
a	: Halatın gemiye bağlanma açısı (derece),
l	: Bağlama noktasının kabul edilen ilk noktaya uzaklığı (m),
M (p,a)	: Sistem atalet kuvveti,
C (p,v)	:Sistem sönüm kuvveti,
K (p)	:Sistem katılık kuvveti,
F (p,v,t)	: Dış kuvvet,
p,v,a	: Sırasıyla konum, hız ve ivme vektörleri,
t	: zaman

TANKER-ŞAMANDIRA BAĞLAMA SİSTEMLERİNİN YAPAY SİNİR AĞLARI TEKNİĞİYLE OPTİMİZASYONU

ÖZET

Çok karmaşık problemlerin çözümü günümüzde gelişen teknoloji ve bilgisayarlar ile daha basit hale gelmiştir. Birçok yöntem ve algoritma kullanılarak hızlı ve efektif çözümler üretilebilmektedir.

Bu yöntemlerden biri de yapay sinir ağı tekniğidir. Temelde biyolojik sinir hücresini esas alan yapay sinir ağı, yapay sinir hücreleri arasında bağlantılar kurarak öğrenilen durumlara uygun sonuçlar verebilmektedir.

Yapay sinir ağı hakkında ilk çalışma 1943 yılında nörofizikçi McCulloch ve matematikçi Pitts tarafından yapılmıştır. Yıllar geçtikçe yeni yöntemler ve algoritmalar geliştirilmiş ve yapay sinir ağı bugünkü halini almıştır.

Yapay sinir ağlarının birçok kullanım alanı bulunmaktadır. Bunların bazıları sınıflandırma, kümeleme, tahmin, desen tanıma, fonksiyon yaklaşımı ve optimizasyondur.

Yapay sinir ağı doğrusal olmaması, öğrenebilmesi, genelleme yapabilmesi, uyarlanabilmesi, hata toleransının az olması ve hızı nedeniyle güçlü bir tekniktir. Özellikle hata toleransının az olması ve problemlere hızlı bir şekilde sonuç bulması yapay sinir ağını diğer algoritmalara göre bir adım öne çıkarmaktadır.

Yapay sinir ağı yapısal olarak üç katmandan oluşmaktadır. Bu katmanlar; giriş katmanı, ara (gizli) katman ve çıkış katmanıdır. Veri akışı katmanlar arasında devamlı şekilde devam etmektedir.

Yapay sinir ağı yapısında temel olarak beş işlem elemanı bulunmaktadır. Bunlar girdi verileri, ağırlıklar, birleştirme fonksiyonu, aktivasyon fonksiyonu ve çıktı verileridir. Girdi verileri giriş katmanına gelerek ağırlıklar ile birleşir ve birleştirme fonksiyonu yardımı ile aktivasyon fonksiyonuna gönderilir. Aktivasyon fonksiyonundan geçtikten sonra çıktı verileri olarak alınırlar.

Üç çeşit öğrenme türü bulunan yapay sinir ağlarında en çok kullanılan öğrenme türü eğitmenli öğrenmedir. Diğer öğrenme türleri ise eğitmensiz öğrenme ve yarı eğitmenli öğrenme olarak bilinmektedir. Ayrıca yapay sinir ağlarında bilinen bir çok öğrenme kuralı bulunmaktadır. Bu kurallardan en çok kullanılanları Hebb kuralı, Hopfield kuralı, Kohonen kuralı, delta kuralı, Levenberg-Marquardt algoritması ve geri yayılım algoritmasıdır.

Ağ yapılarına göre ileri ve geri beslemeli olmak üzere ikiye ayrılmaktadır. İleri beslemeli ağda veriler bir sonraki katmana aktarılırken, geri beslemeli ağda veriler bir sonraki katmana aktarıldığı gibi kendinden önce bulunan katmanlara da aktarılabilir.

Tek katmanlı modeller ile başlayan yapay sinir ağı çalışmaları daha sonra çok karmaşık problemlere de çözüm bulunamadığı için çok katmanlı yapıları da ortaya çıkarmıştır. Tek katmanlı yapılarda perceptron, ADALINE ve MADALINE modelleri ilk bilinen modellerdir. Çok katmanlı yapılarda ise giriş ve çıkış katmanları arasında birden çok ara (gizli) katman bulunabilmektedir.

Yapay sinir ağlarının paralel olarak bilgiyi işleme, öğrenilmesi, eksik bilgiler ile çalışabilmesi, yeni örneklerle çabuk uyum sağlayabilmesi ve matematik modele ihtiyaç duymaması gibi özellikleri avantajları olarak söylenebilir. Dezavantajları ise sistem içinde ne olduğu bilinmediği için ağın verdiği sonuçları işleminin zorluğu, eğitilmek için zamana ihtiyaç duyması ve yalnızca nümerik bilgiler ile çalışması gösterilebilir.

Çok noktalı tanker-şamandıra bağlama sistemi dizaynında birçok parametre önem arz etmektedir. Bunlardan bazıları rüzgar, dalga karakteristiği, akıntı, bağlama halatlarının bağlanma noktası ve bağlama açısı olarak gösterilebilir.

Bu çalışmada OrcaFlex programı ile 4 noktalı tanker-şamandıra bağlama sistemleri modellendikten sonra değişik çevre şartları ve farklı bağlama şekilleri ile simülasyonlar yapılmıştır.

Bu simülasyonlardan elde edilen sonuçlar yapay sinir ağlarını eğitmede kullanılmış ve bir algoritma elde edilmiştir. Bu algoritma ile farklı bağlama şekilleri simülasyona gerek kalmadan, kullanılacak parametreler belirli aralıkta yapay sinir ağlarına veri olarak girilerek optimum bağlama şekli seçilebilir.

OPTIMIZATION OF SPREAD MOORING SYSTEMS WITH ARTIFICIAL NEURAL NETWORKS

SUMMARY

Artificial Neural Network (ANN) is an artificial representation of human brain that tries to simulate its learning process. The most basic element of human brain is a specific type of cell, which provides us with the abilities to remember, think and apply previous experiences to our every action. These cells are known as neurons. ANN use artificial neurons to create a network like human's nervous system.

The first study about ANN was produced in 1943 by neurophysicist McCulloch and mathematician Pitts. After that many studies about ANN were carried out. There are some important studies about ANN respectively, Hebbian Rule (1949), ADALINE (1960), Perceptron (1969) "Neural Networks and Physical Systems" (1982).

Artificial neural networks are used in many discipline and sciences. There are many problems that solved by ANN in mathematics, medicine, business and finance. The main areas of ANN's are classification, clustering, forecast, function approximation and optimization.

The basic properties of ANN are modelling of nonlinear problems, parallel structure, employment of missing information, error tolerance, learning and generalization. These properties show ability of ANN about solving complex problems.

Architecture of ANN is separated in 3 layers. The first part is input layer. This layer takes data from external environment. The second layer is hidden layers. There may be multiple hidden layers. The last layer is output layer. The output layer consists of neurons that communicate the output of the system to the user or external environment.

There are 5 process elements in ANN. One of them is inputs. Inputs are coming from external environment to neurons. Other element is weights which transfer inputs to process element. Weights show mathematics coefficient of inputs. Joining function sums the inputs after multiplication of weights. Activation function is applied a process to sum which comes from joining function. There are a few activation functions in use. Sigmoid activation function is the most commonly used function. Sigmoid function could differentiate and both sigmoid function and its differentiation are continuous. The other common activation function is tangent hyperbolic activation function. This function works like sigmoid function but gives results in $[-1,1]$ interval where sigmoid function gives $[0,1]$. The last element of ANN is output which is the value obtained after activation function process.

Learning types of ANN are divided in 3 groups. First group is supervised learning. This learning requires both inputs and outputs to make an algorithm. Another group is unsupervised learning. Unsupervised learning takes inputs without outputs and create an algorithm. Last group is semi-supervised group which like unsupervised learning but this make a comparison with outputs.

Also, there are many learning rules usable in ANN. First learning rule was produced by Canadian psychologist Donald Hebb in 1949 which calls as “Hebb Rule”. This rule’s basic is that if there are two neurons inform each other and both active, the network between them should be more strength. Hopfield rule is similar to Hebb rule. If inputs and outputs are both active or passive the network between them should strengthened or weakened. Kohonen rule say that the cell which has the biggest weight has power to impress other neighbour cells. Delta rule is a developed Hebb rule which is one of the most commonly used learning rule. This rule is similar to least squares method. Levenberg-marquardt algorithm is an approach to Newton method. Levenberg-Marquardt algorithm is a method that research minimum. This algorithm uses Hessian matrix. The other rule is back propagation algorithm. In this rule, the error propagate to back. The error between real data and calculated data changes by using weights.

There are two type of ANN according their network structures. First type is feed forward ANN. Feed forward ANN send data from input layer to hidden layer with no changes. Data take a process in this part and then send to output layer. Feedback ANN can send data to all layers without any consraint.

ANNs divided in 2 groups based on their layer number. ANN could be one layer or multi layers. One layer ANN gives result only 1 or 0. Perceptron, ADALINE and MADALINE are known as one layer ANN models. Multi layer ANN model has many hidden layers.

The advantages of ANN are learning ability, parallel structure, working with missing data and requiring no mathematic model. Also, ANN has some disadvantages. ANN needs time to train and has no rule in its networks. Only works with numerical data.

MatLab has a toolbox named “Neural Network Toolbox” where user can make artificial network’s training, learning, application and simulation. MatLab has two ways to use “Neural Network Toolbox”. First one is entering to command window the “nntool” command and second is starting toolbox from “Apps” part of MatLab.

This thesis use ANN to optimize tanker-buoy mooring systems. Tanker-buoy mooring systems are offshore systems, so that it affected from environmental conditions. To reduce these effects tanker-buoy systems are supported by mooring lines. Mooring systems could be single point or multi points. Single point system moves to adapt environmental conditions but multi points system stay in its first point. Multi points systems are exposed like wave, current and wind forces because of no adaptation to environment. The maneuvering area is important in tanker- buoy mooring systems. Because there may be other systems around it.

OrcaFlex program is used to design tanker-buoy mooring system to simulate and get results about mooring tensions and tanker displacements. 4 points mooring system is designed. This system is used to test program with different wind speeds and gets expected tension results when wind speed increases.

Before entering data to ANN, a normalization is made and data ranges are reduced to [0,1] interval. These data are used to train ANN structure and an algorithm is occured.

After creating algorithm, the results of training, test and validaton gets from ANN. At this algorithm Levenberg-Marquardt learning is used. To make error correction least squares method used. There are 14 neurons in hidden layers.

This ANN algorithm can be used for optimization of new designs without simulation studies. Designing and simulating process in OrcaFlex take more times according to ANN structure. Artificial neural networks is fast and simple system, to use in different mooring systems.

1. GİRİŞ

Mühendislikte tasarım bir ihtiyacı karşılamak ya da mevcut sistemden daha iyi bir çözüm ortaya koymak için yapılan çalışma olarak adlandırılabilir. Tasarım, bu şartları yerine getirirken belirli kısıtlamalar göz önüne alınarak yapılabilir. Bu şartlarda en iyi çözümü bulmak mühendisliğin temel amaçlarındandır.

Tanker-şamandıra bağlama sistemleri açık denizde demirlenmiş tankerlerin kargo yükleme/boşaltma işlemleri için kullanılan sistemlerdir. Bu sistemler uzun servis ömrü, değişik su derinliklerinde kullanılabilme faydası ve değişik tonajdaki tankerlere hizmet verebilme ve kısa zamanda yerleştirilebilme özellikleri ile yaygın kullanım alanına sahiptirler (Mentes, 2010).

Çok noktalı veya tek noktalı olabilen bu sistemlerde bağlama sayısı, bağlama noktaları ve bağlama açıları gibi parametreler sistemin çalışmasını etkilemektedir. Bu yüzden bu parametrelerin tayininde en iyi seçimi yapmak önem arz etmektedir. Bu sistemin optimize edilmesi maliyet/fayda oranını iyileştirecektir.

Optimizasyon belirli kısıtlamalar altında bir durumun en iyilenmesi olarak adlandırılabilir. Bu en iyileme işlemi birçok yöntem ile yapılabilmektedir. Geleneksel yöntemlere karşılık bilgisayar teknolojisi kullanılarak da optimizasyon yapılabilmektedir. Optimizasyon, geleneksel yöntemler olan fonksiyon yaklaşımı, doğrusal programlama vb ile yapıldığında çok amaçlı, büyük ölçekli ve yüksek karmaşıklıkta problemlere çözüm üretememektedir. Bu nedenle optimizasyonun gelişen teknoloji ile beraber kullanılması kaçınılmaz olmuştur.

Gelişen bilgisayar teknolojisi ile optimizasyon yeni bir boyut kazanarak geleneksel yöntemlerden uzaklaşmıştır. Genetik algoritmalar gelişen teknoloji ile geleneksel yöntemlere alternatif olarak kullanılan bir optimizasyon türüdür. Genetik algoritmalar ile büyük ölçekli, lineer olmayan ve süreksiz fonksiyon ile tanımlı bir problemin dahi çözümü gerçekleştirilebilmektedir. Fakat genetik algoritmalar ile çözüm yapmak ele alınan problemin karmaşıklığı nedeniyle uzun zaman alabilmektedir.

Yapay sinir ağı (YSA), bir algoritma oluşturarak hızlı bir şekilde çözüm üretme yeteneğine sahiptir. Basit olarak insan sinir sistemini temel alan bir algoritma olan yapay sinir ağı genetik algoritmalar gibi büyük ölçekli, lineer olmayan ve süreksiz/eksik problemlerde genetik algoritmalara göre daha hızlı çözümü ile daha çok kullanılmaktadır.

Literatürde yapay sinir ağı tekniğini kullanarak bir takım çalışmalar yapılmıştır. Pina ve diğ. (2013) yaptıkları çalışmada bağlama hatlarının ve rayzerlerin analiz ve dizaynını da içeren yapay sinir ağı tabanlı yeni bir model sunmuşlardır. Çalışmanın amacı sonlu elemanlar tabanlı lineer olmayan dinamik analiz sonuçlarına yakın bir sonuç üretirken zamandan kazanmaktır.

Guarize ve diğ. (2008) etkin bir yapay sinir ağı- sonlu elemanlar hibrit yöntemi kullanarak bağlama hatları ve rayzerlerin dinamik analizi için lineer olmayan akıntı verilerini işleyerek çıktı elde etmektedir. İlk olarak sonlu elemanlar yöntemi ile zamana bağlı kısa bir simülasyon yapılmış ve daha sonra yapay sinir ağı kullanılarak kalan simülasyon tahmin edilmeye çalışılmıştır. Bu hibrit yöntem çok uzun süreli çalışmalarda tahmin amaçlı kullanılabilir.

Jacob ve diğ. (2014) dalgacık ağı (Wavelet Networks-WN) adını verdikleri ileri beslemeli yapay sinir ağı mimarisi ve dalgacık dönüşümleri kombinasyonundan oluşan bir çalışma yapmışlardır. Bu çalışmada amaç, açık deniz yapıları bağlama hatları ve rayzerler için sonlu elemanlar yöntemi ile elde edilen sonuçları iyi tahmin etmek işlem ve zamanında önemli bir azalma sağlamaktır.

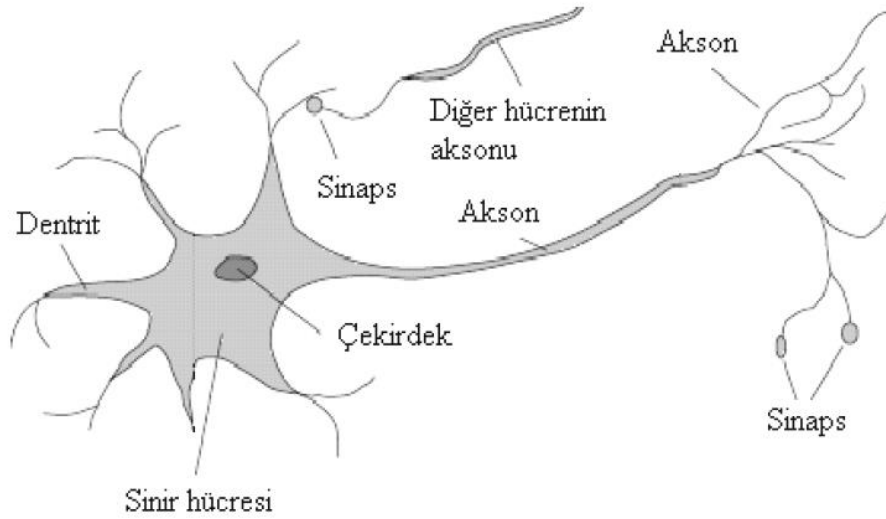
Mahfouz (2006), Capability Polar Plots Program, (CPPP) ve yapay sinir ağını kullanarak sondaj gemisi üzerinde bir çalışma yapmıştır. Bu çalışmada CPPP programı ile dinamik pozisyonlama verileri YSA'ya tanıtılmış ve optimum dinamik konumu elde etmek için gerekli verilere ulaşılmaya çalışılmıştır.

2. YAPAY SİNİR AĞLARI

2.1 Genel

Düşünme yeteneği ve zekâ beynin ve merkezi sinir sisteminin görevidir. Beyni hasara uğramış birçok kişide öğrenme ve çevreye uyumda bazı sorunlar olduğu gözlemlenmiştir (Topaloğlu, 2007).

Sinir sistemi nöron adı verilen sinir hücrelerinden oluşmaktadır. İnsan vücudunda yaklaşık olarak 10^{11} nöron bulunmaktadır (Kohonen, 1988). Beyin nöronların birleşiminden oluşmaktadır. Nöronlar sadece beyinde değil, sinir sistemi üzerinde bütün vücutta bulunmaktadır. Nöronlar grup halinde çalıştıklarından bir ağ olarak adlandırılabilir. Bu ağda bulunan her nöronun giriş kısmı (dendrit), çıkış kısmı (akson), bağlantıları (snaps) ve hücre çekirdeği bulunmaktadır (Şekil 2.1).



Şekil 2.1 : Nöronun yapısı.

Bağlantılar hücreler arasındaki iletişimi sağlar. Nöron, diğer nöronlardan gelen sinyalleri bağlantılar üzerinden giriş kısmına alır. Gelen sinyaller bağlantılar tarafından zayıflatılır ya da güçlendirilir. Giriş kısmına gelen sinyaller hücre çekirdeğine iletilir. Hücre çekirdeğine gelen sinyaller birbirleri ile etkileşimleri sonucu belirli bir eşik değerini aşabilirlerse çıkış kısmına sinyal gönderilir ve sinir aktif hale gelir. Eşik değeri aşılamazsa sinyal gönderilemez.

Beynin nöronlar sayesinde olan bu çalışması günümüzdeki en hızlı işleme sistemlerinin de üzerinde bir performans göstermektedir. Bu durum göz önünde bulundurularak yapay sinir ağları modelleri geliştirilmiştir.

2.2 Yapay Sinir Ağları Tarihçesi

Yapay sinir ağları ile ilgili ilk çalışma 1943 yılında nörofizikçi McCulloch ve matematikçi Pitts tarafından yapılmıştır. Bu çalışmada yapay sinir tanımı yapılarak bir hücre modeli geliştirilmiştir (McCulloch ve Pitts, 1943).

1949 yılında bir sinir ağının öğrenmeyi nasıl yapabildiği hakkında fikir veren bir çalışma yapılmıştır. Bu çalışma sayesinde günümüzde kullanılan öğrenme kurallarının çoğunun temeli olan ‘Hebbian Kuralı’ ortaya çıkmıştır (Hebb, 1949).

1954 yılında ilk defa uyarılara tepki veren ve adapte olabilen bir model oluşturulmuştur (Farley ve Clark, 1954).

1958’de Hebbian Kuralı’ndan sonraki en önemli adım olan algılayıcı (perceptron) Rosenblatt tarafından geliştirilmiştir. Bu gelişme günümüzdeki makinelerin öğrenme algoritmalarının da temelini oluşturmaktadır (Şen, 2004).

1960’da ADALINE (Adaptive Linear Combiner) öğrenme kuralı geliştirildi. Bu öğrenme kuralı Widrow-Hoff öğrenme kuralı olarak literatüre geçmiştir. En önemli özelliği modelin eğitimi boyunca oluşan hatayı en aza indirmeyi hedeflemesidir (Widrow ve Hoff, 1960).

1969 yılında ‘Perceptron’ adlı kitap yayınlanmış ve tek katmanlı algılayıcıların karmaşık problemleri çözemediğini XOR problemi örneği ile göstermiştir. XOR (eXclusive OR) probleminde işlem sonucu girişindeki işaretler birbirinden farklıysa çıkış olarak 1 diğer durumlarda çıkış olarak 0 verir. Bu kitap ile başlayan bir süreçte YSA ile ilgili çalışmalar neredeyse durma noktasına gelmiştir.

1980’li yıllarda bilgisayarların daha etkin kullanılmaya başlanmasından sonra YSA ile ilgili kayda değer çalışmalar ortaya çıkmıştır. 1982 yılında yapılan “Sinir Ağları ve Fiziksel Özellikler” çalışması ile YSA’da modern devrin başladığı söylenebilir. Bu çalışmada doğrusal olmayan ağlar geliştirilmiştir (Hopfield, 1982). Aynı yıllarda Kohonen(1982) ve Anderson (1983) tarafından yapılan çalışmalarda eğitici-siz öğrenen sistemlerinin geliştirilmesiyle YSA hakkındaki çalışmalar ivme kazanmıştır.

1986'da çok tabakalı algılayıcılar için geri yayılım algoritması geliştirilmiştir. Geliştirilen bu algoritma güçlü bir yapıya sahip olduğundan daha önce karşılaşılan sorunların çözülmesini sağlamıştır (Rumelhart ve diğ., 1986). Geri yayılım algoritması günümüzde de en çok kullanılan algoritmalarından biridir. Günümüzde YSA'nın kullanımı bilgisayar sistemlerindeki gelişmelere paralel olarak büyük bir yoğunluk kazanmıştır. Her geçen gün yeni öğrenme algoritmaları geliştirilmekte ve ağ mimarileri hakkında yeni çalışmalar yapılmaktadır.

2.3 Kullanım Alanları

Yapay sinir ağları hemen her disiplin ve bilim dalında kullanılabilmektedir. Fen bilimleri, matematik, tıp, işletme, finans gibi birçok alanda YSA ile çözülmüş birçok problem bulunmaktadır. Bu bölümde başlıca kullanım alanlarına değinilecektir.

2.3.1 Sınıflandırma

Sınıflandırma, bir nesnenin birden çok sınıf arasından hangisine ait olduğunun belirlenmesi işlemidir. YSA halihazırdaki bir sınıflandırmayı kullanarak gelecek verilerin hangi sınıfa ait olduğunun tesbit edilebilmesini sağlar. Kümeleme

Kümeleme, sınıflandırmadan farklı olarak belirli bir sınıra bağlı olmaksızın yapılır. İşlemden önce veriler hakkında herhangi bir bilgiye gerek yoktur. Benzer verilerin aynı gruba dahil edilmesi ile kümeleme yapılır.

2.3.2 Tahmin

Yapay sinir ağlarının en çok kullanıldığı alan tahmindir. Tahmin, geçmişteki verilere dayanarak gelecek veriler hakkında fikir üretmektir. Örnek olarak borsadaki salınımlar, satış miktarları, hava olayları tahminleri verilebilir. YSA özellikle doğrusal bir yapıda değişkenlerin tahmininde kullanılmasında önemli avantaj sağlar. 1990 yılında yapılan bir çalışmada YSA modelleri ile güneş lekelerinin klasik olarak 11 senede bir görülmesinin tahminleri yapılabilmektedir (Li ve diğ., 1990).

2.3.3 Desen tanıma

Bozuk ya da eksik desenlerin YSA'ya tanıtılıp desenlerin karşılaştırılmasıyla bozuk desenin doğrusunun elde edilmesi sağlanır. Bu yöntem benzer olan yüz tanıma ve retina tanıma gibi işlemlerde de kullanılabilir. Bu işlemlerle kimlik doğrulaması yapılması mümkün olmaktadır.

2.3.4 Fonksiyon yaklaşımı

Matematik ifadesi bilinmeyen birçok hesaplama modeli vardır. Yani girdileri ve çıktıları belli olmasına rağmen aradaki fonksiyonun bilinmediği durumlar vardır. Fonksiyon yaklaşımı, aynı giriş verilerine karşılık yaklaşık aynı çıkış verilerini üretecek işlevin tanımlanmasıdır. Fonksiyon ile tahmin yaklaşımları birbirine benzeyen modellerdir.

2.3.5 Optimizasyon

Optimizasyon, birçok ticari ve bilimsel olayın belirli kısıtlamalar ile hedefin en büyük veya en küçük olması yani en iyileme işlemidir. Mühendislikte optimizasyon özellikle tasarım aşamasında çok fazla kullanılmaktadır. Optimizasyon temel olarak ikiye ayrılabilir. İlki matematiksel verilere dayanarak bir fonksiyon oluşturulup yapılan geleneksel optimizasyondur. İkincisi ise yapay zeka kullanılarak yapılan optimizasyon tekniğidir. Bu yöntem fonksiyon olarak ifade edilemeyecek kadar karmaşık yapıların optimizasyonunda kullanılır. Yapay sinir ağları da bu yöntemi kullanır. Bu çalışmada da YSA kullanılarak optimizasyon yapılacaktır.

2.4 YSA'nın Özellikleri

Yapay sinir ağlarının doğrusal olmayan problemleri modelleyebilmesi, hesaplama ve bilgi işleme gücü, paralel dağılmış yapısı, eksik bilgi ile çalışabilmesi, hata toleransına sahip olması, öğrenebilme ve genelleme özelliklerine sahip olması en önemli özelliklerindendir. Bu özellikleri yapay sinir ağlarının karmaşık problemleri çözebilme yeteneğini göstermektedir. Literatürde birçok bilimsel çalışmada kullanılan YSA aşağıda ayrıntılı olarak verilen özellikleri sayesinde önemi bir yere sahiptir.

2.4.1 Doğrusal olmama

Yapay sinir ağlarının en temel elemanı olan nöronlar doğrusal değildirler. Bu nedenle nöronların oluşturduğu yapay sinir ağı mimarileri de doğrusal değildir. Bu özelliği YSA'nın günümüzde birçok kez karşımıza çıkan doğrusal olmayan problemlerin çözümünde önemli bir yere sahip olduğunu gösterir.

2.4.2 Öğrenme

İnsan sinir sistemindeki çalışma şeklini kullanan YSA, sorunu çözmek için eldeki verileri kullanarak işlemdeki saklı ilişkileri ortaya çıkarır. Bu işleme ağı fonksiyonu öğrenmesi işlemi denir. Öğrenme işlemi temel olarak YSA oluşturulurken hücrelerin arasındaki bağlantılarda kullanılan ağırlıkların belirlenmesi işlemidir.

Yapay sinir ağları belirli bir algoritma kullanmadığı için öğrenme özelliği önem teşkil etmektedir. İnsanlardaki öğrenmeye benzeyen YSA öğrenme sistemine örnek olarak, bir çocuğun sıcak bir cisme dokunmaması gerektiğini çeşitli tecrübelerle öğrenmesi gösterilebilir. Sıcak cisme bir kere dokunduktan sonra dokunmaması gerektiğini öğrenen çocuk, o andan sonra daha az sıcak cisimleri dokunması gerektiği bilgisini öğrenmiş olacaktır (Civalek ve Çatal, 2004).

2.4.3 Genelleme

Yapay sinir ağları öğrenme işlevini gerçekleştirirken giriş verilerinden farklı olarak yeni veriler için de anlamlı sonuçlar üretebilir. Bu özellik genelleştirme yeteneği olarak tanımlanır. Genelleme özelliği olmayan bir YSA'nın anlamlı olamayacağı açıktır. Desen tanıma, tahmin yapma ve sinyal gibi konularda genelleme yeteneğinin sonucu olarak başarı elde edilmektedir.

2.4.4 Uyarlanabilirlik

Yapay sinir ağları karşılaşılan probleme göre yeniden uyarlanabilir ve yeniden eğitilerek ağırlıklarını değiştirebilir. Bu özelliği ile yapay sinir ağları sistem tanıma, denetim, uyarlamalı örnek tanıma gibi alanlarda etkin olarak kullanılmaktadır.

2.4.5 Hata toleransı

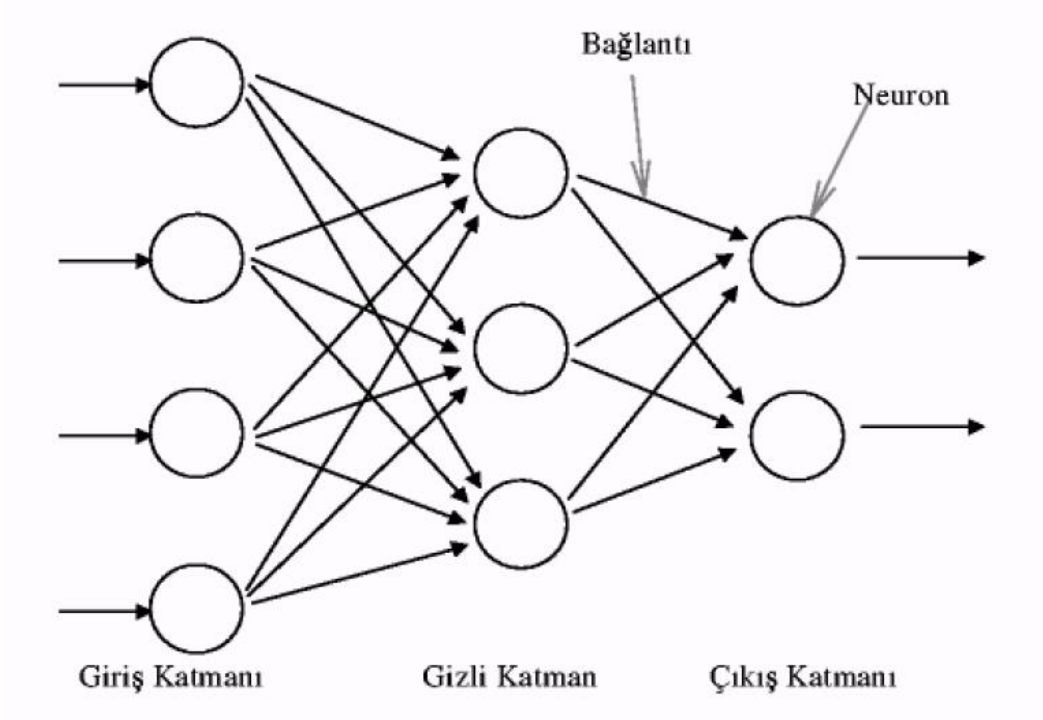
Yapay sinir ağları çok sayıda yapay sinir hücresinin çeşitli şekillerde birbirine paralel bağlanması ile oluşan bir yapı olduğundan ağa girilen veriler bütün bağlantılar üzerine dağılmış olarak bulunmaktadır. Zaman zaman ağ eğitimi için kullanılan veri kümesinde bazı hatalar olabilir. Bu hataların etkileri verilerin bütün bağlantılar üzerinde dağılmış olmasından dolayı azaltılmış olur ve bu nedenle tolere edilebilir. Bu özelliğinden dolayı YSA'ların hata toleransı kullanılan geleneksel sistemlere nazaran daha fazladır.

2.4.6 Hız ve tasarım kolaylığı

Yapay sinir ağlarının paralel bir sistem olmasından dolayı büyük ölçekli entegre devre teknolojisiyle gerçekleştirilebilir. Bu nedenle YSA hızlı bilgi işleme yeteneğine sahiptir ve gerçek zamanlı uygulamalarda kullanılabilir. YSA mimarisinde kullanılan temel birim olan hücre elemanının yapısı ve modeli, bütün YSA yapılarında yaklaşık aynıdır. Farklı uygulama alanlarındaki aynı YSA mimarileri de aynı hücreleri kullanılabilir. YSA'nın bu özelliği sayesinde tasarım ve analiz yaparken kolaylık sağlanmış olur.

2.5 YSA'nın Yapısı

Yapay sinir ağları birçok yapay sinir hücresinden meydana gelir. Biyolojik sinir sistemine benzetilmek istenen bu yapı ile işlem elemanlarının aynı doğrultu üzerinde bir araya gelmesi istenmektedir. Bu ağın yapısı temel olarak üç katmandan oluşur. Bunlar girdi katmanı, ara katman ve çıktı katmanıdır (Şekil 2.2).



Şekil 2.2 : YSA'nın temel yapısı.

2.5.1 Girdi katmanı

Girdi katmanı gelen verilerin alındığı katmandır. Buraya gelen veriler ara katmana aktarılır. Bazı ağ yapılarında veriler aktarılırken herhangi bir işlem yapılmaz (Svozil ve diğ., 1997).

2.5.2 Ara katman

Gizli katman olarak da bilinen bu kısım, girdi katmanından gelen verileri işleyerek çıktı katmanına gönderir. Bir YSA’da birden fazla ara katman bulunabilir.

2.5.3 Çıktı katmanı

Ara katmandan gelen bilgileri işleyerek, girdi katmanına sunulan veriler için üretmesi gereken çıktıyı üretirler.

2.6 YSA’da İşlem Elemanları

Bir önceki bölümde belirtilen katmanlarda yapılan işlevleri gerçekleştiren elemanlardır. Bunlar girdiler, ağırlıklar, birleştirme fonksiyonu, aktivasyon fonksiyonları ve çıktılarıdır.

2.6.1 Girdiler

Dış ortamdan hücreye gelen bilgilerdir. Hücre ile birleşerek girdi katmanını oluştururlar.

2.6.2 Ağırlıklar

Girdi katmanında oluşan veriler, bağlantılar üzerindeki ağırlıklar ile işlem elemanına gider. Ağırlıklar ara katmandaki bir işleme etkiyen girdinin matematiksel katsayını gösterir. Hesaplama başladığında ağırlıklar rastgele verilir. Yapılan işleme göre hata minimum olana kadar ağırlıklar değiştirilmeye devam edilebilir.

2.6.3 Birleştirme fonksiyonu

Ağırlıklar ile birleşen giriş verileri birleştirme fonksiyonu ile toplanır. Birleştirme fonksiyonu ağ yapısına göre maksimum ya da minimum alan veya çarpım fonksiyonu olabilir.

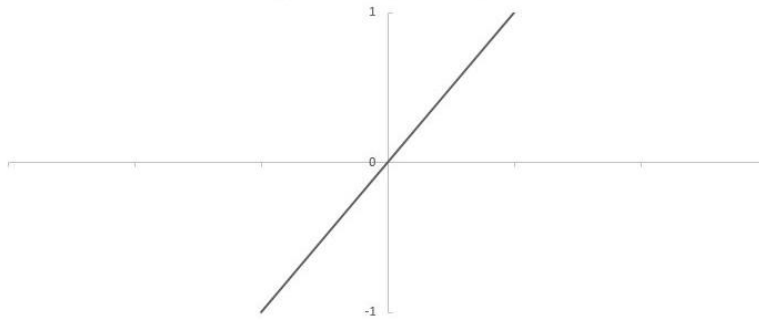
2.6.4 Aktivasyon fonksiyonu

Aktivasyon fonksiyonu transfer fonksiyonu olarak da bilinir. Birleştirme fonksiyonundan elde edilen net girdiyi bir işlemde geçirerek çıktıyı belirler. Birkaç tip aktivasyon fonksiyonu vardır. En uygun aktivasyon fonksiyonu tasarımcının denemeleri sonucunda belli olur. Girdi katmanında oluşan veriler bağlantılar üzerindeki ağırlıklar ile işlem elemanına gider. Ağırlıklar ara katmandaki bir işleme etkileyen girdinin matematiksel katsayını gösterir. Aktivasyon fonksiyonları içinde en çok kullanılan sigmoid ve tanjant hiperbolik fonksiyonlarıdır. YSA mimarilerinde kullanılan çeşitli aktivasyon fonksiyonları vardır:

- Doğrusal Aktivasyon Fonksiyonu
- Sigmoid Aktivasyon Fonksiyonu
- Tanjant Hiperbolik Aktivasyon Fonksiyonu

2.6.4.1 Doğrusal aktivasyon fonksiyonu

Doğrusal aktivasyon fonksiyonu, doğrusal YSA'larda ya da çok katmanlı YSA'ların çıkış katmanlarında kullanılmaktadırlar (Şekil 2.3). Doğrusal fonksiyon belirli bir aralıkta sınırlanırsa rampa fonksiyonu olur. Doğrusal aktivasyon fonksiyonu $y = ax$ şeklinde tanımlanabilir. Bu fonksiyonda a sabit bir sayıdır.

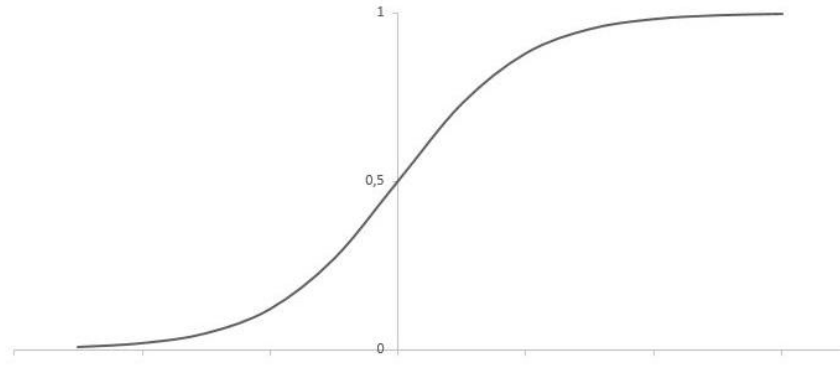


Şekil 2.3 : Doğrusal aktivasyon fonksiyonu

2.6.4.2 Sigmoid aktivasyon fonksiyonu

Sigmoid aktivasyon fonksiyonu türevi alınabilir ve hem kendisinin hem de türevinin sürekli olmasından dolayı en çok kullanılan fonksiyonlardandır (Şekil 2.4). Sigmoid aktivasyon fonksiyonu sadece pozitif değerler üretir ve şu formül ile ifade edilir:

$$f(x) = \frac{1}{1+e^{-x}} \quad (2.1)$$

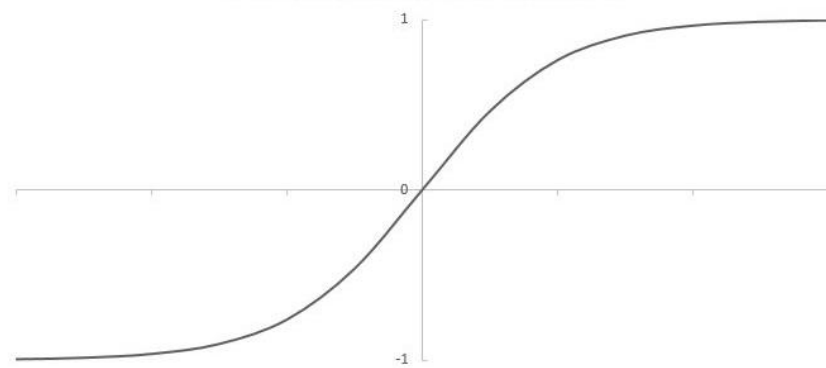


Şekil 2.4 : Sigmoid aktivasyon fonksiyonu

2.6.4.3 Tanjant hiperbolik aktivasyon fonksiyonu

Tanjant hiperbolik aktivasyon fonksiyonu sigmoid aktivasyon fonksiyonuna benzer fakat çıkış uzayı biraz daha geniş bir fonksiyondur (Şekil 2.5). Sigmoid aktivasyon fonksiyonu $[0,1]$ aralığında sonuç verirken tanjant hiperbolik aktivasyon fonksiyonu $[-1,1]$ aralığında sonuç verir (Ozkan ve Erbek, 2003). Tanjant hiperbolik aktivasyon fonksiyonunun formülü şu şekildedir:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.2)$$



Şekil 2.5 : Tanjant hiperbolik aktivasyon fonksiyonu

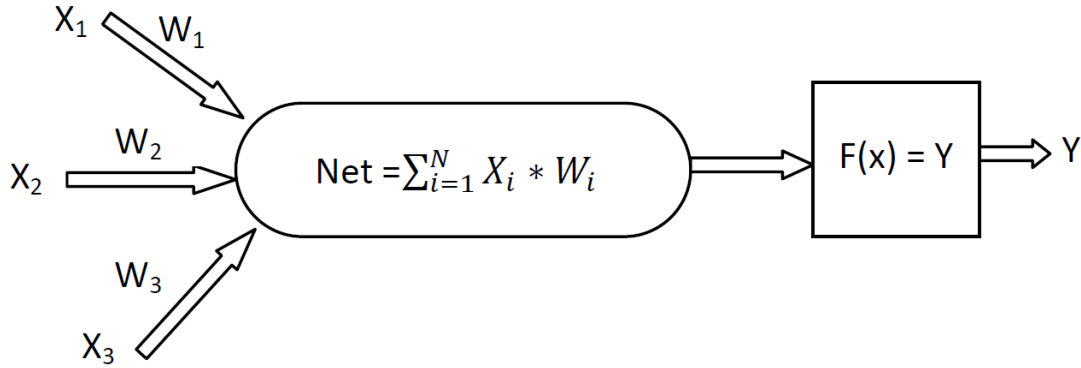
2.6.4.4 Diğer aktivasyon fonksiyonları

Basamak fonksiyonu, kutuplamalı basamak fonksiyonu ve parçalı doğrusal fonksiyon diğer aktivasyon fonksiyonlarıdır.

2.6.5 Çıktı

İşlemler gerçekleştikten sonra (aktivasyon fonksiyonundan sonra) ortaya çıkan değer çıktı değeridir.

YSA işlem elemanlarının basit gösterimi Şekil 2.6’da gösterilmiştir. Burada x girdi, w ağırlıklar net toplama fonksiyonu, $F(x)$ aktivasyon fonksiyonu ve Y çıktı değeridir.



Şekil 2.6 : YSA mimarisi

2.7 YSA Öğrenme Türleri ve Öğrenme Kuralları

Yapay sinir ağlarında sık olarak kullanılan öğrenme türleri ve belli başlı geliştirilmiş öğrenme kuralları vardır. Bunların bazıları bu bölümde incelenecektir.

2.7.1 Öğrenme türleri

Yapay sinir ağlarında temel olarak üç tip öğrenme şekli vardır. Bunlar eğitilmiş öğrenme, eğitimsiz öğrenme ve yarı eğitilmiş öğrenmedir.

2.7.1.1 Eğitilmiş öğrenme

En sık kullanılan öğrenme türlerindendir. Denetimli öğrenme olarak da adlandırılabilir. Eldeki çıktılarınla ağdan elde edilen çıktı değerleri karşılaştırılır. Hata toleransına göre minimum seviyeye ulaşmaya kadar iterasyon devam eder. İstenilen çıktı değerlerine yakınsayan değerler elde edilince öğrenme biter. Öğrenme bittikten sonra ağırlıklar değişmez.

Eğitilmiş öğrenmede, test sürecinde eğitim sürecinde kullanılmamış datalar ağı girdi olarak verilir. Çıkan sonuçlar eldeki verilerle karşılaştırılarak ağın test süreci de tamamlanmış olur. Test aşamasında çıkan sonuçların istenilen seviyede olmaması durumunda ağdaki iterasyon sayısı, ara katmanlar, ara katmanlardaki nöron sayıları ya da ağırlıklar değiştirilir. Yani öğrenme sürecine geri dönmüş olur.

2.7.1.2 Eđitimsiz ğrenme

Eđitimsiz ğrenme denetimsiz ya da eđitmensiz ğrenme olarak da bilinmektedir. Ađa eđitilmiş ğrenmeden farklı olarak sonuçlar verilmez. Sadece giriş deđerleri verilerek ađın eđitimi sađlanır. Sistem girdi deđerlerine gre ađırlıkları kendi belirler. Sınıflandırma problemlerinde yoğun olarak kullanılan bir ğrenme trdr.

2.7.1.3 Yarı eđitilmiş ğrenme

Yarı eđitilmiş ğrenmede sisteme sadece girdi deđerleri verilir. Fakat ıkan sonuçların eldeki ıktı deđerlerine gre bir karşılaştırması yapılır ve bir dođruluk derecesi verilerek ađın kendini eđitmesi sađlanır.

2.7.2 ğrenme kuralları

Yapay sinir ađlarında en ok kullanılan ğrenme kuralları Hebb kuralı, Hopfield kuralı, Kohonen kuralı, delta kuralı ve geri yayılım algoritmasıdır.

2.7.2.1 Hebb kuralı

retilen ilk ğrenme kuralıdır. 1949 yılında Kanada'lı psikolog Donald Hebb tarafından biyolojik temele dayalı olarak geliştirilmiştir. Hebb kuralında eđer iki nron birbirlerinden bilgi alıyorsa ve ikisi de aktifse o iki nron arasındaki bađın kuvvetlenmesi gerekir. Yani aktif olan nron bađlantılı olduđu nronu da aktif hale getirmeye alıacaktır. Hebb kuralı eđitimsiz bir ğrenme kuralıdır.

2.7.2.2 Hopfield kuralı

Hebb kuralı ile benzerlik gstermektedir. Eđer beklenen girdi ve ıktıların her ikisi de aktif ya da pasifse hcreler arası bađlantı, ğrenme katsayısı kadar kuvvetlendirilir ya da zayıflatılır. ğrenme katsayısının belirli bir deđer yoktur, kullanıcı tarafından atanır.

2.7.2.3 Kohonen kuralı

Teuvo Kohonen tarafından geliştirilen bir algoritmadır. Bu algoritmada en byk ađırlıđa sahip hcre evresindeki hcreleri sınırlama yetkisine sahiptir. Sadece en byk ađırlıđa sahip hcre ve komşu hcreleri ađırlıkları dzenler.

2.7.2.4 Delta kuralı

Hebb kuralının geliştirilmiş halidir. En ok kullanılan ğrenme kurallarından biridir. Widrow-Hoff kuralı olarak da adlandırılır. Kuralın temeli, beklenen deđerler ile ađın

ürettiği değerleri arasındaki farkı, ağırlıkları değiştirerek en aza indirgemeyi amaçlamaktadır. En küçük kareler yöntemine benzerlik göstermektedir. Delta kuralının amacı, çıkan değerler ile gerçek değerler arasındaki farkın karelerinin ortalamasını en aza indirmektir.

2.7.2.5 Levenberg-Marquart algoritması

Levenberg-Marquart (LM) algoritması Newton metoduna bir yaklaşımdır. LM algoritması minimumu araştırma metodudur. Her bir iterasyon adımında hata yüzeyine parabolik yaklaşımla yaklaşılr ve parabolün o adım için çözümü oluşturur. LM öğrenme kuralı 2.3 denklemindeki Hessian matrisini kullanır.

$$H = J^T J \quad (2.3)$$

Burada J, Jacobian matrisini sembolize eder. Jacobian matrisi, denklem 2.4'te belirtildiği gibi ağ hatalarının ağırlıklara göre birinci türevidir.

$$J(n) = \frac{\delta e(n)}{\delta w(n-1)} \quad (2.4)$$

Burada;

n: iterasyon sayısı

δ : türev sembolü

e: ağ hataları vektörü

w: bağlantı ağırlıkları

olarak alınmıştır.

Bu durumda gradyent, denklem (2.5) ile bulunur.

$$g = J^T e \quad (2.5)$$

ve nöronlar arasındaki bağlantı ağırlığı da denklem (2.6) ile elde edilir.

$$w_{k+1} = w_k [J^T J + \mu I]^{-1} J^T e \quad (2.6)$$

Burada;

w_k : k. İterasyondaki ağırlık

I : birim matris

μ : Marquardt parametresi

olarak alınmıştır.

Marquardt parametresi skaler bir değerdir. Marquardt parametresi sıfır olduğunda LM yöntemi Newton algoritması adını alır. Levenberg-Marquardt öğrenme algoritması, diğer algoritmalarla kıyasla oldukça hızlı sonuç üretir (Rojas, 1996).

2.7.2.6 Geri yayılım algoritması

Geri yayılım algoritması veya diğer adıyla genelleştirilmiş delta kuralı en çok kullanılan öğrenme kuralıdır. Rumelhart ve diğ. (1986) tarafından üretilen algoritma yapay sinir ağlarına önemli bir ivme kazandırmıştır. Geriye yayılım algoritmasında hata geriye doğru yayılır. Gerçek çıkışlar ile hesaplanan çıkışlar arasındaki hatanın geriye doğru yayılarak ağırlıklar değiştirilir. Öğrenme algoritmalarının tümünde olduğu gibi amaç giriş ve çıkış verileri arasındaki en iyi uyumu sağlayacak şekilde ağırlıkları belirlemektir.

Geri yayılım algoritması delta algoritmasının momentum terimi eklenmiş halidir. Momentum terimi hatanın en küçük olduğu noktayı bulmada ve doğrultunun ayarlanmasına yardımcı olur (Caudill, 1988).

2.8 Ağ Yapılarına göre YSA

Yapay sinir ağlarında bulunan hücreler ve bağlantılar çok değişik şekillerde bir araya gelebilmektedir. Yapılarına göre YSA'lar nöronlar arasındaki bağlantıların yönlerine ve ağ içindeki akış yönüne göre birbirlerinden ayrılmaktadır. Yapay sinir ağları yapılarına göre ileri beslemeli yapay sinir ağları ve geri beslemeli yapay sinir ağları diye ikiye ayrılır.

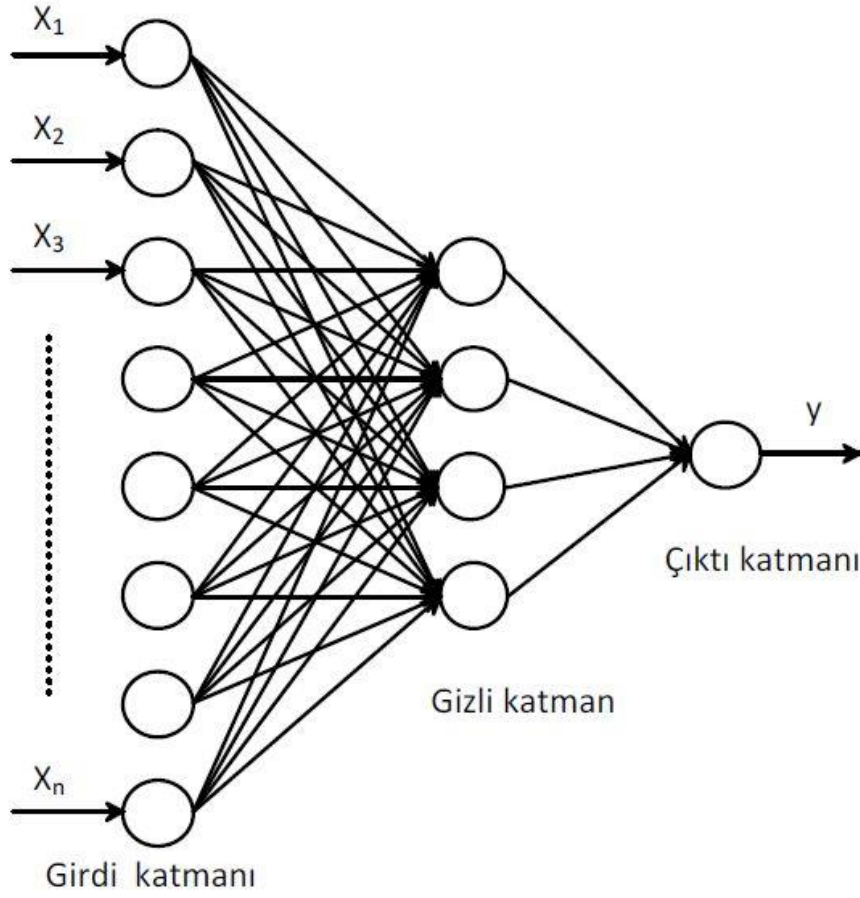
2.8.1 İleri beslemeli yapay sinir ağları

İleri beslemeli yapay sinir ağlarında işlemler giriş katmanından çıkış katmanına doğru iletilir(Şekil 2.7). Giriş katmanı dışarıdan aldığı bilgileri hiçbir değişikliğe uğratmadan ara(gizli) katmandaki hücrelere verir. Veriler bu katmanlarda işlenir ve çıktı katmanına gönderilir. Ara katmanlardaki her bir çıkış değeri bir sonraki katman için giriş değeri olarak kullanılır.

2.8.2 Geri beslemeli yapay sinir ağları

Geri beslemeli yapay sinir ağlarında en az bir hücrenin çıkışı kendisine ya da başka bir nörona giriş olarak verilir. Geri beslemeli ağda bilgi alışverişi bir katmandaki hücreler arasında olabileceği gibi katmanlar arasındaki hücreler arasında da olabilir.

Geri beslemeli ağ bu yapısı ile dinamik bir davranış özelliği göstererek doğrusal olmayan problemleri çözmede etkin olarak kullanılabilir.



Şekil 2.7 : İleri beslemeli yapay sinir ağı

Geri beslemeli YSA'lar sayısal filtre tasarımlarında sıkça kullanılırlar. Geri beslemeli YSA'lar ileri beslemeli YSA'lara göre daha yavaş çalışırlar. En çok bilinen geri beslemeli ağ tipleri; Hopfield ağı, hücresel yapay sinir ağı (HYSA), Grossberg ağı, Adaptive Resonance Theory-1 (ART1) ve ART2 ağlarıdır (Uçan ve diğ., 2003).

2.9 Katmanlarına Göre YSA

Yapay sinir ağları katmanlarına göre tek katmanlı ve çok katmanlı olmak üzere ikiye ayrılır.

2.9.1 Tek katmanlı algılayıcılar

1943 yılında ilk geliştirilen yapay sinir ağları, girdilerin ağırlıklı toplamının eşik değeri ile karşılaştırılması sonucu çıktı üreten bir yapıdaydı. Eşik değerine eşit ya da büyükse 1, aksi durumda 0 sonucuna ulaşıyordu. 1950’li yıllardan sonra algılayıcı adı verilen yapay sinir ağları geliştirildi.

2.9.1.1 Basit algılayıcı (Perceptron)

Basit algılayıcı ya da diğer adıyla tekli doğrusal algılayıcı, gelen verileri ağırlıklar ile çarparak çıkan sonuçları bir sabit değer ile toplayıp çıkışa gönderir. Bu da ağın eğitimini tamamlaması anlamına gelmektedir.

2.9.1.2 Yinelemeli algılayıcı (ADALINE)

Adaptive Linear Element ya da kısa adı ile ADALINE olarak bilinen bu algılayıcıda delta kuralına benzer bir şekilde karelerin ortalamasının en aza indirilmesi hedeflenir. Basit algılayıcı ile arasındaki en önemli fark ise öğrenme kuralıdır.

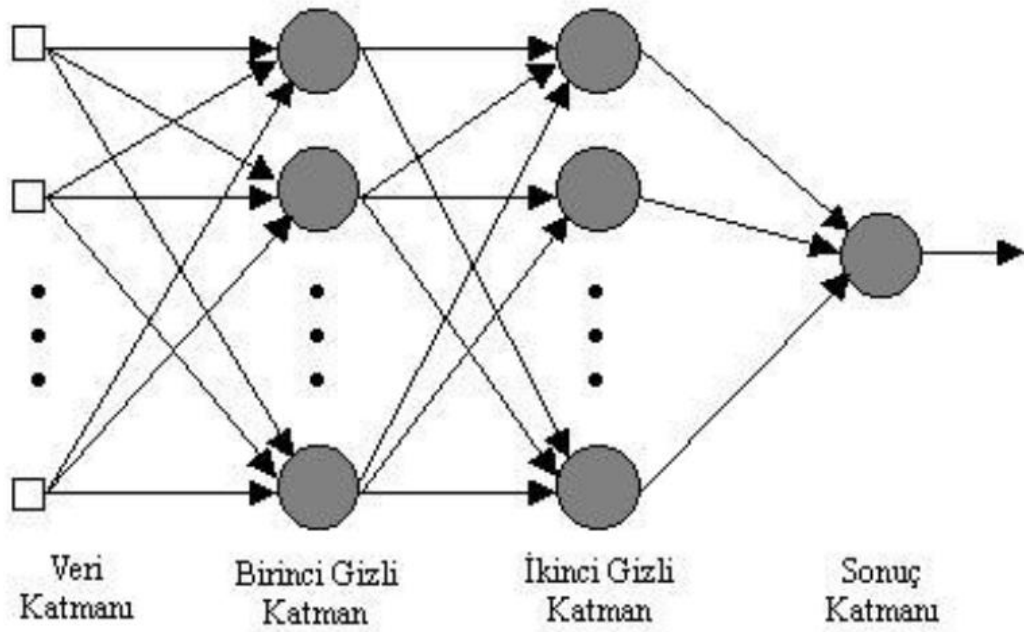
2.9.1.3 Çoklu yinelemeli algılayıcı (MADALINE)

Çoklu yinelemeli algılayıcı ya da kısa adı ile MADALINE olarak bilinen bu algılayıcının yinelemeli algılayıcıdan farkı değişken sayısının daha fazla olabilmesidir.

2.9.2 Çok katmanlı algılayıcılar

Çok katmanlı algılayıcı, birçok sinir hücresinin bir araya gelmesiyle ortaya çıkan çok tabakalı ve çok hücreli bir mimari yapıya sahiptir. Çok katmanlı algılayıcı, tek katmanlı algılayıcıların geliştirilmiş halidir (Şekil 2.8).

Çok katmanlı algılayıcılarda, ağdaki her bir nöron doğrusal olmama özelliği içerir. Bu doğrusal olmama biçimi genellikle sigmoid fonksiyonu ile sağlanır. Çok katmanlı algılayıcılar gizli nöronlardan oluşan gizli katmanlara sahiptir. Ağ, bağlantılar sayesinde çok yüksek derecede bilgi işleme yeteneğine sahiptirler. Ağın bilgi işleminde değişiklik olabilmesi için bağlantı sayısında ya da ağırlıklarda değişiklik olması gerekir.



Şekil 2.8 : Çok katmanlı YSA

Çok katmanlı algılayıcılarda bilgiler giriş katmanında kabul edilir. Daha sonra gizli katmanlardaki işlemlerde elde edilen sonuçlar çıkış katmanına verilir. İstenen hata toleransı içinde değerler elde edilinceye kadar ağırlıklar ya da gizli katman sayısı, gizli katmanlardaki nöron sayısı değiştirilerek işleme devam edilir.

2.10 Yapay Sinir Ağlarının Avantajları ve Dezavantajları

2.10.1 YSA'ların avantajları

Yapay sinir ağları öğrenme işlemini gerçekleştirebilmekte ve öğrenme işlemi gerçekleştirildikten sonra belirsizlikler altında öğrendikleri olaylar ile ilgili ilişkiler kurarak karar verebilmektedirler (Öztemel, 2003).

Yapay sinir ağlarındaki paralel yapı bilginin kolay anlaşılmasını ve işlenmesini sağlar.

Yapay sinir ağları eksik bilgiler ile de çalışabilmektedir. Ağ hücrelerin bir bölümü bozulsan bile çalışmaya devam edebilir. Bozulan hücrelerin durumuna göre ağın performansı düşebilir fakat hangi hücrenin önemli olduğuna ağ kendi karar verebilmektedir.

YSA yeni örneklerle çabuk uyum sağlayabilir. Ağ yeni verilerle tekrar tekrar eğitilip kullanılabilir. Bu sayede sisteme uygun yeni çözümler üretilebilir.

YSA matematiksel modele ihtiyaç duymaz. Probleme çözüm üretmeden önce belirli bir varsayımda bulunmaz.

2.10.2 YSA'ların dezavantajları

YSA sistemi içinde ne olduğu bilinemez. Bu nedenle bazı durumlarda ağların verdiği sonuçları değerlendirmek zor olabilir.

Eğitilmek için zamana ihtiyaç duyarlar. Bundan dolayı zaman ve para maliyeti vardır. Ağın çalışmasında iterasyon sayısı arttıkça çözüm süresi de artar. Bu da zamandan kayıptır.

YSA içerisindeki ağ parametrelerinin belirlenmesinde belirli bir kural bulunmamaktadır. Parametrelerin belirlenmesi kullanıcının tecrübesine bağlıdır. YSA her sistem için ayrı değerlendirme yapar.

YSA yalnızca nümerik bilgilerle çalışır. Bu nedenle problemlerin nümerik olarak tanımlamak zorunludur.

Ağın eğitiminini bitirmek için belirli bir zaman yoktur. Bu tamamen kullanıcının belirlediği hata toleransına bağlıdır. Kullanıcının belirlediği tolerans en iyi öğrenmeyi sağlamayabilir.

2.11 YSA Yazılımları

YSA'lar için geliştirilmiş birçok yazılım bulunmaktadır. Çizelge 2.1'de bu yazılımlardan bazıları belirtilmiştir.

Bu yazılımlardan MatLab: Neural Network Toolbox ve NeuroSolutions bu bölümde incelenecektir.

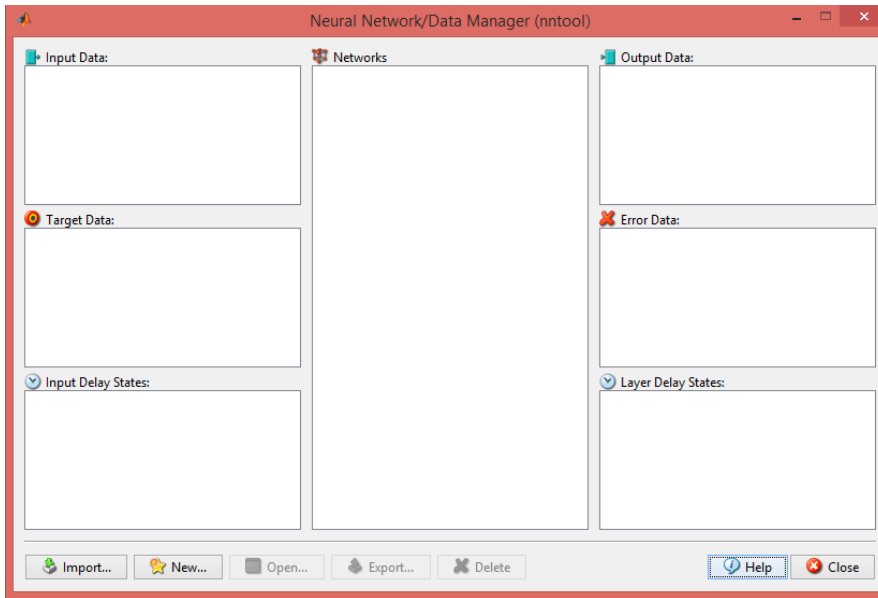
Çizelge 2.1: Bazı YSA yazılımları.

Ücretsiz Yazılımlar	Ücretli Yazılımlar
NeuraShell	STATISTICA: Neural Network
PDP++Software	NeuroForeCaster/GENETICA
Fann Tool	NeuroSolutions
MUME	BrainMaker
NNSYSID	Braincel
SNNS	NeuroLab
DartNet	MatLab: Neural Network Toolbox

2.11.1 MatLab: Neural network toolbox

Neural Network Toolbox (NNT), MatLab platformunda YSA kurulumu, eğitimi, uygulaması ve simülasyonunu sağlayan bir uygulamadır. Uygulama, kod yazmadan YSA kullanmak için tasarlanmıştır.

NNT kullanmak için iki farklı yol vardır. İlkinde MatLab komut penceresine “nntool” yazılıp enter’a basıldığında “Neural Network/Data Manager” arayüzü ekrana gelir (Şekil 2.9).

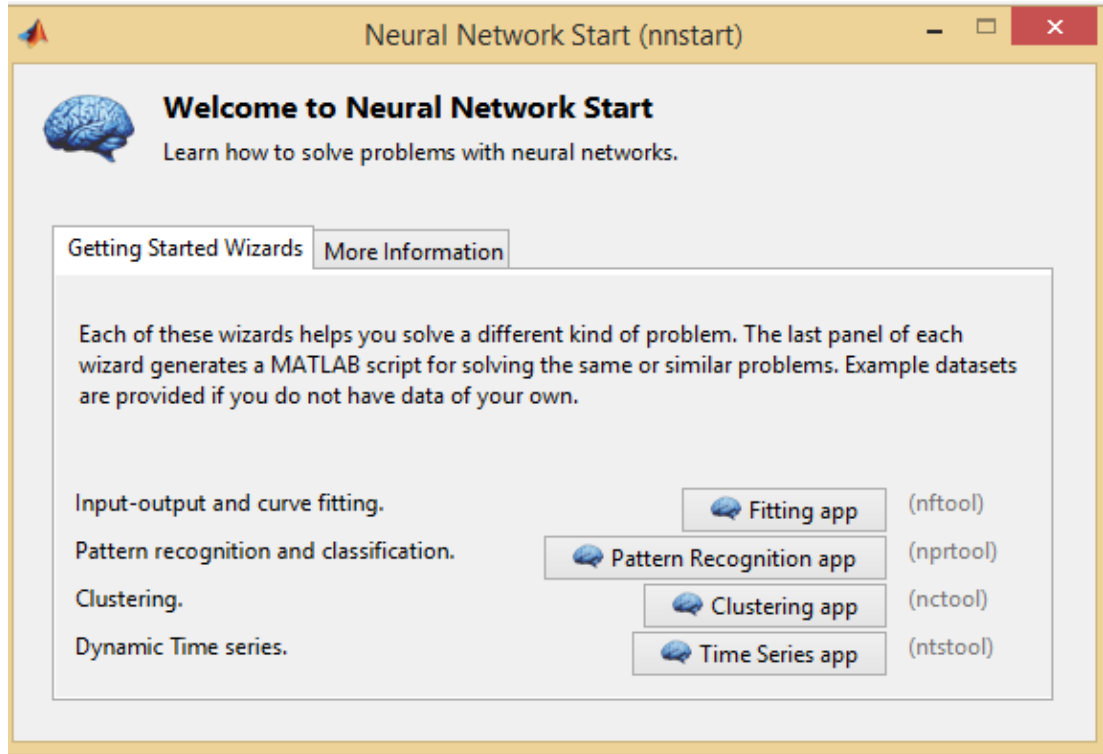


Şekil 2.9 : Neural network toolbox arayüzü

Açılan pencerede “new” butonuna tıklanarak “Create Network or Data” penceresi açılır. Bu pencereden hem ağ oluşturulur hem de giriş (inputs) ve çıkış (targets) değerleri matris olarak sisteme girilir. Daha sonra network sekmesinden ağ

oluşturulur. Ağ oluşturulurken ağ tipi, giriş ve çıkış verileri, eğitim algoritması, öğrenme fonksiyonu, performans fonksiyonu, katman sayısı, katmanlardaki nöron sayıları ve aktivasyon fonksiyonları seçilir. “Create” butonu ile ağ oluşturulur. Daha sonra Neural Network/Data Manager penceresinden oluşturulan ağ açılır. Açılan pencerede “Train” sekmesi tıklanır. Giriş ve çıkış verileri tanıtıldıktan sonra ağ eğitilir.

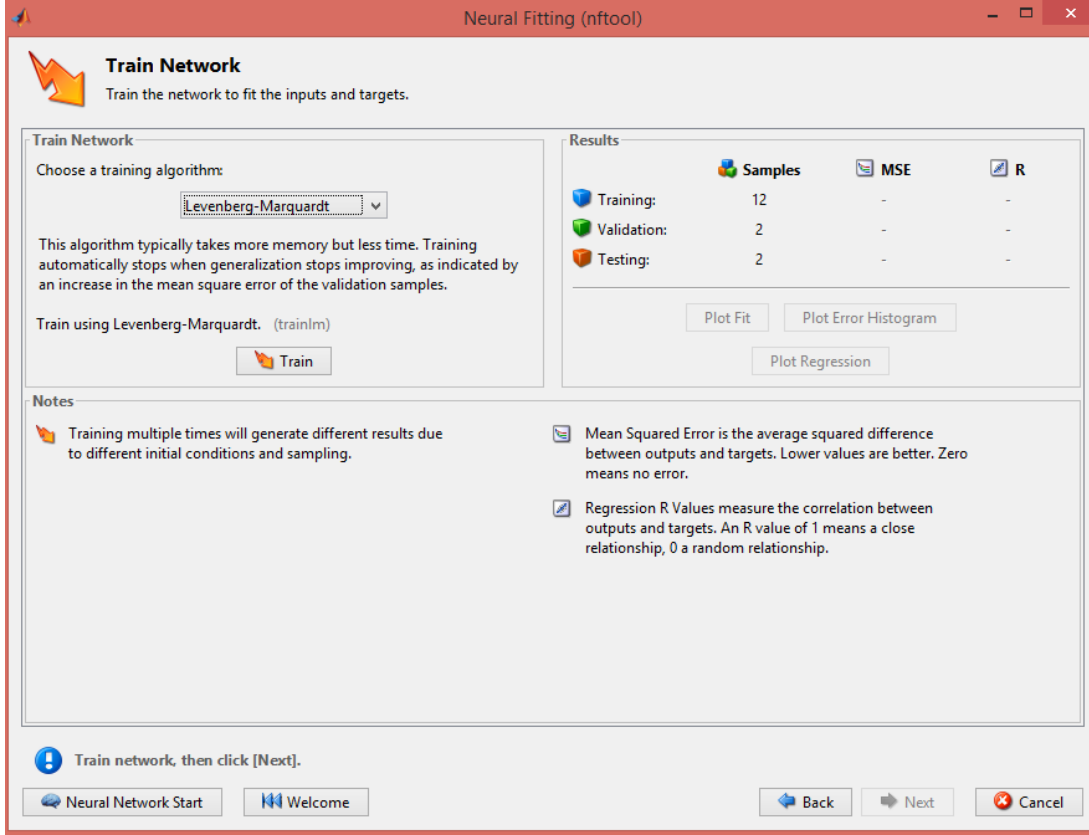
MatLab NNT kullanımı için ayrı bir arayüz daha tasarlamıştır (Şekil 2.10). Bu arayüzde kullanım daha kolay olup işlemler daha çabuk gerçekleştirilebilmektedir.



Şekil 2.10 : Neural network toolbox arayüzü

Bu arayüz açıldığında ekrana 4 seçenek çıkmaktadır. Bu seçeneklerden kullanılmak istenen seçenek işaretlenir ve açılan pencerede YSA hakkında bilgilendirme bulunmaktadır. “Next” butonu ile bir sonraki aşamaya geçilir. Bu aşamada giriş ve çıkış verileri sisteme tanıtılır. Bu arayüzde sisteme tanıtılan veriler, MatLab dosya uzantısı olan “.m” uzantısı olarak kaydedilmiş olur. “Next” butonu ile geçilen bir sonraki aşamada verilerin sınıflandırılması yapılır. Kullanılan verilerin yüzde kaçının eğitim verisi, yüzde kaçının test verisi ve yüzde kaçının doğrulama verisi olarak kullanılacağı belirlenir. “Next” butonu ile bir sonraki aşamaya geçilir. Bu aşamada ağ mimarisi oluşturulur. Gizli nöron sayıları belirlenir. Bir sonraki aşama ise eğitim

aşamasıdır (Şekil 2.11). Burada eğitim algoritması seçilir ve “Train” butonu ile eğitim gerçekleştirilir.



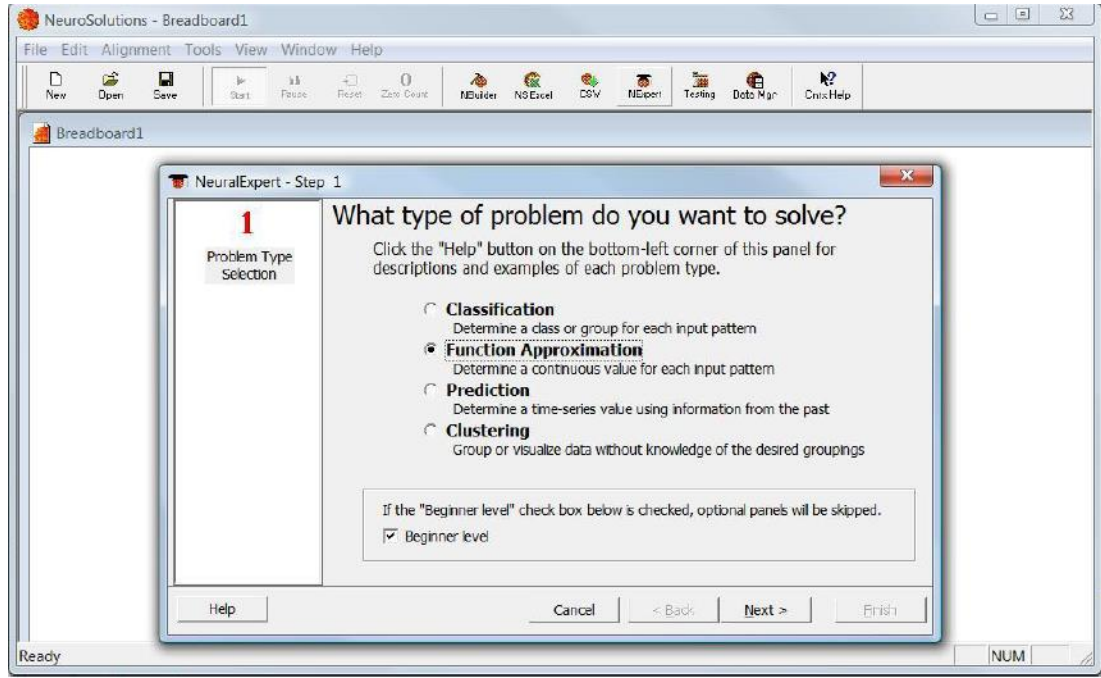
Şekil 2.11 : Neural network toolbox eğitim bölümü

Eğitim tamamlandıktan sonra yine aynı pencerede eğitim, test ve doğrulama hakkında çıkan istatistiki sonuçlar bulunur. Ayrıca eğitim bittikten sonra açılan yeni pencerede yapılan iterasyon sayısı, performans, eğitim zamanı gibi parametreler hakkında bilgiler bulunur. Bu pencereden çeşitli grafiksel sonuçlara da ulaşılabilir.

2.11.2 NeuroSolutions yazılımı

NeuroDimension firması tarafından geliştirilen NeuroSolutions yazılımı tahmin, sınıflandırma vb. amaçlar için kullanılabilecek farklı ağ yapılarını içeren güçlü bir yazılımdır. NeuroSolutions, NeuroSolutions for Excel ve NeuroSolutions for MatLab olmak üzere farklı kullanım seçenekleri sunar (Hamzaçebi, 2011).

YSA oluşturulurken, hangi tip problem ile çalışılacağı göz önüne alınarak nasıl bir yaklaşım yapılacağına karar verilir. İlk adımda hangi tip problemin çözüleceği sisteme girilir (Şekil 2.12).



Şekil 2.12 : NeuroSolutions programı

Program problem tipine en uygun YSA ağını seçmektedir. Eğer istenirse “Neuro Builder” bölümünden YSA yapısı değiştirilebilir. Excel üzerinden giriş ve çıkış değerleri programa tanıtıldıktan sonra problem karmaşıklık seviyesi belirtilir. Eldeki verilere göre düşük, orta ve yüksek seviyelerinden biri seçilerek YSA mimarisi oluşturulmuş olur.

YSA mimarisi oluşturulduktan sonra eğitimi yapılır. Eldeki veriler sisteme sırasıyla eğitim, doğrulama ve test etme biçiminde tanıtılmalıdır.

3. TANKER-ŞAMANDIRA BAĞLAMA SİSTEMLERİ

Tanker-şamandıra bağlama sistemleri (mooring systems) açık deniz ortamında çalışan sistemler olduğu için deniz ve diğer çevre hareketlerinden etkilenmektedirler. Bu etkileri azaltmak ya da en aza indirmek için bağlama hatları ile desteklenirler. Bağlama sistemleri tek noktadan bağlı ya da çok noktadan bağlı sistemler olmak üzere iki gruba ayrılır (Şekil 3.1 ve Şekil 3.2).



Şekil 3.1 : Tek noktadan bağlı sistem



Şekil 3.2 : Çok noktadan bağlı sistem

Tek noktadan baęlı sistem, evre kořullarına uyum saęlayana kadar hareket ederken ok noktadan baęlı sistemler baęlandığı ilk řekliyle sabit kalır. Tanker-řamandıra baęlama sistemleri dalga, akıntı ve rüzgarın oluřturduęu yklere maruz kalırlar.

Tanker-řamandıra baęlama sistemlerinde manevra alanı nemlidir. Manevra alanını belirlemek iin sistem kullanıma sunulmadan nce belirli testlerden geer. Manevra testlerinde dnme kabiliyeti, doęrusal rota stabilitesi, rota deęiřtirme kabiliyeti gibi durumlar test dilir. Etrafında bulunabilecek dięer sistemler nedeniyle sınırlı bir alanda alıřmak gerekebilir.

Tek noktadan baęlı sistemler bu aıdan daha az tercih edilir. Tek noktalı baęlamalar denizde deęiřen evre řartlarına gre hareket ettięi iin dięer sistemler ile teması konusunda problemler ile karřılařma ihtimali yksektir.

Tek noktalı baęlama sistemler aynı zamanda depo grevi de grrler. Baęlama konumu genellikle geminin bař tarafındadır. ok eřitli tek noktadan baęlı sistem bulunmaktadır. Bunların bazıları sabit kule, ayaklı baęlama řamandırası, mafsallı kule tipi ykleme platformu, tek nokta ve rezervuardır (Barltrop, 1998).

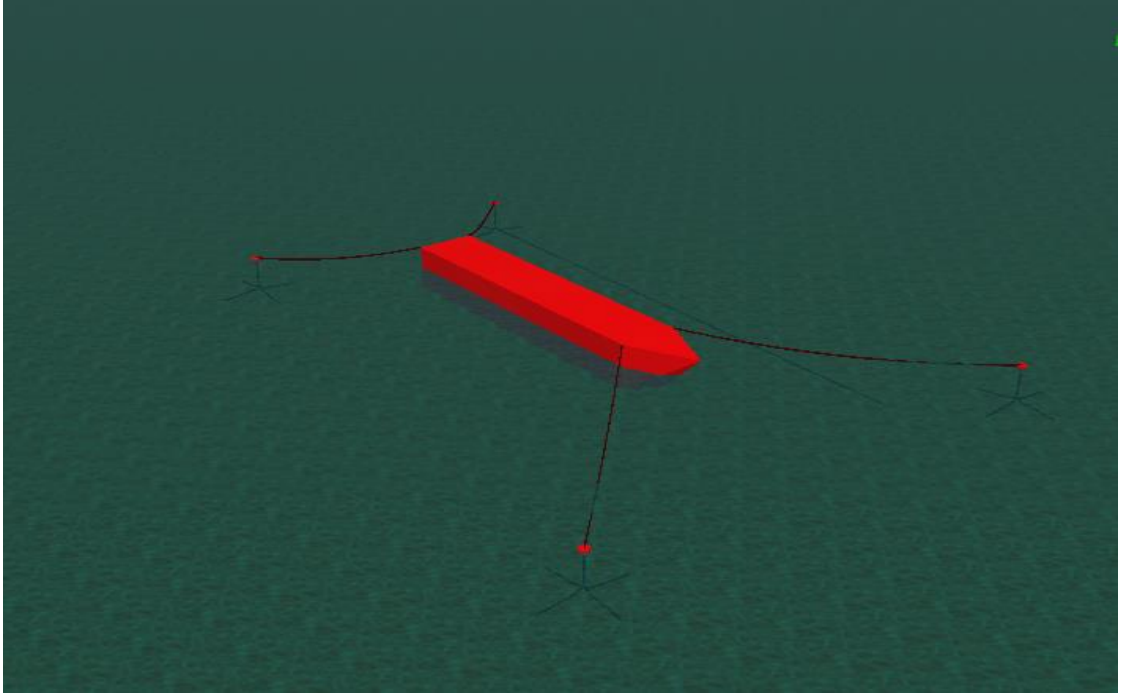
Tanker řamandıra baęlama sistemleri tasarımında dikkat edilmesi gereken pek ok faktr bulunmaktadır. Deniz derinlięi, deniz řartları, tasarım, imalat ve kurulumu, emniyet řartları bu faktrlerden bazılarıdır.

Tasarımda ayrıca baęlama tipi, baęlantı halatlarının dayanıklılıęı, deniz dibinin řekli ve sistemin ne kadar sre baęlı kalacaęı gz nne alınmalıdır. Baęlama halatları ok farklı fiziksel ve mekanik zellikleri olan maddelerden oluřturulabilmektedir. oęu mhendislik probleminde olduęu gibi maliyet-fayda boyutu da nemlidir. Seilen malzemeler, yapıya etkiyecek evre řartlarının iyi tanımlanması ve dięer seenekler direkt olarak maliyeti etkileyecektir (Carbono ve dię., 2005).

3.1 Tanker-řamandıra Baęlama Sistemi Dizaynı

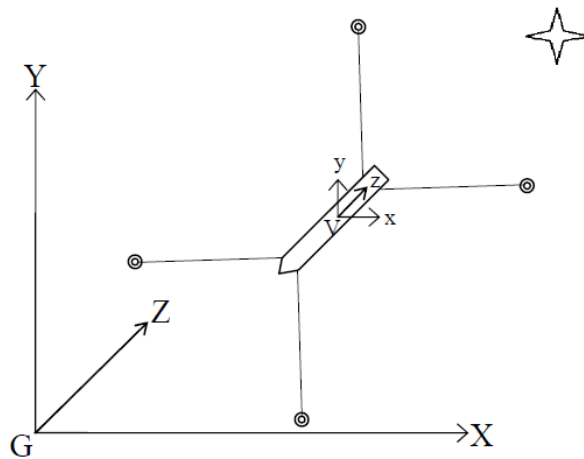
Dalga, akıntı ve rüzgar gibi evresel yklere maruz kalan tanker-řamandıra baęlama sistemleri iin baęlama halatlarına gelen ykler ile tankerin yer deęiřtirmeleri hesaplanacaktır. Bu hesaplar iin baęlama halatları geminin bař ve kı taraflarında 30, 45, 60 ve 75 derece aılarla baęlanmışır. Daha sonra bu halatlar geminin hem bař hem de kı tarafından sırası ile 5, 10, 15, 20, 25, 30, 35, 40, 50, 60 ve 70 metre yer deęiřtirilerek yeniden oluřturulmuřtur. Tanker-řamandıra baęlama sistemi bařtan

2 ve kıçtan 2 olmak üzere toplam 4 noktadan bağlama yapılarak tasarlanmıştır (Şekil 3.3).



Şekil 3.3 : Tasarlanan tanker-şamandıra bağlama sistemi

Çevre şartlarının tayininde Özhan ve Abdalla (2002) ‘nın dalga ve rüzgar atlasından faydalanılmıştır. Bu bölgede su derinliği 28 m olarak alınmıştır. Dalga, akıntı ve rüzgar karakteristiklerine uygun simülasyonlar yapılmıştır. Şekil 3.4’te gemi için kullanılan koordinat sisteminin global koordinat sistemine göre konumu gösterilmiştir.



Şekil 3.4 : Global ve yerel koordinat sistemi

Bağlama sistemi global eksen takımına göre 180 derecelik açı yapacak şekilde konumlandırılmıştır.

3.1.1 Tanker-Şamandıra bağlama sistemi genel karakteristikleri

Çizelge 3.1’de tanker-şamandıra sistemine ait bilgiler verilmiştir. Tankerlerde ABS (2009) kurallarına uygun demir, zincir ve bağlama halatı bulunduğu kabul edilmiştir.

Çizelge 3.1: Tankerin genel özellikleri.

			Tanker
Deplasman	Δ (ton)		125832
Tam boy	LOA (m)		250
Dikeyler arası boy	LBP (m)		228,2
Genişlik	B (m)		38,71
Tasarım su çekimi	Tdesign (m)		16,17
Boyuna rüzgâr kesit alanı	AL (m ²)		3601,4
Enine rüzgâr kesit alanı	AT (m ²)		857,2
Blok katsayısı	C _B		0,804

Sistem daha önce de belirtildiği gibi dalga, akıntı ve rüzgâr gibi dış kuvvetlerin tesiri altında hareket eder. Tanker ile şamandıra arasındaki bağlama halatı ön gergili olarak tasarlanmıştır. Tanker-şamandıra bağlama sistemi tekne, şamandıralar, şamandıralara ait yükselen zincirler, yatak zincirleri, yatak zincirlerine bağlı çapalardan oluşmaktadır.

Tanker-şamandıra sisteminde bağlama halatlarına gelen maksimum gerilme ve yer değiştirmeleri hesaplamak için farklı senaryolar ile simülasyonlar yapılmıştır. Çizelge 3.2 rüzgar, akıntı ve dalga karakteristikleri ve yönleri hakkında bilgi vermektedir.

Çizelge 3.2 : Tanker için simülasyon senaryo aralıkları

Yön	Rüzgâr hızı (m/s)	Akıntı hızı (m/s)	Dalga yüksekliği (m)	Dalga periyodu (s)
Güney	0-30	0-2	0-5	3,5-7,5
Güneybatı	0-30	0-2	0-5	3,5-7,5
Kuzey	0-30	0-2	0-5	3,5-7,5
Kuzeydoğu	0-30	0-2	0-5	3,5-7,5

Ayrıca bağlama açıları ve yerleri de değiştirilerek simülasyonlar yapılmıştır. Dinamik hesaplamalar için OrcaFlex programı kullanılmıştır.

3.2 OrcaFlex Programı

OrcaFlex, Orcina firması tarafından 1986 yılında üretilen bir benzetim ve hesap programıdır. OrcaFlex rayzer sistemleri, bağlama halatları, montaj ve kule sistemleri, yenilenebilir enerji ve savunma çalışmaları konularında kullanılabilmektedir (Orcina, 1987-2013). OrcaFlex programı deniz ortamında dalga, akıntı ve diğer dış tesirli hareketlerin etkisindeki açık deniz yapılarının hızlı ve doğru bir şekilde dinamik analizini yapmaya olanak sağlar. OrcaFlex, benzetim sonunda elde edilen sonuçların sunumu için geniş bir grafik arayüzüne sahiptir (Yetkin ve diğ., 2013).

OrcaFlex sonlu elemanlar yöntemini kullanarak 3 boyutlu tasarımların çözümünü yapabilmektedir. Program ilave çevre şartlarının eklenmesine izin vererek mühendislik gereksinimlerine uygun hızlı ve etkili bir tasarım ortaya çıkarılmasına olanak tanır. OrcaFlex, savunma ve yenilenebilir enerji sektörlerinde de kullanılabilmektedir.

OrcaFlex programı Şekil 3.4'te verilen koordinat sistemini kullanmaktadır. Bu koordinat sisteminde G global orijini, GX, GY ve GZ is global eksen yönlerini göstermektedir. Ayrıca OrcaFlex programı modeldeki her nesnenin yerel koordinatını gösteren sisteme de sahiptir. Dönme eksen yönü pozitif saat eksen yönüdür.

OrcaFlex statik ve dinamik analizler yapabilen bir programdır. Statik analizde sistemin statik haldeki ağırlık, yüzme vb. altında dengesi belirlenir ve dinamik simülasyonlar için başlangıç değerleri oluşturur.

Dinamik analizde ise statik analizden gelen başlangıç değerleri ile başlayan ve belirli bir zaman periyodunda sistemin hareketlerinin incelendiği bir simülasyon yapılır. Programda kullanılan zaman birimi saniyedir. Program dinamik hesaplar için iki tip integral kullanabilir. Bunlar açık ve kapalı integraldir.

OrcaFlex simülasyon sırasında aşağıdaki hareket denklemini çözer (Mentes, 2010):

$$M(p,a) + C(p,v) + K(p) = F(p,v,t) \quad (3.1)$$

Bu denklemde;

$M(p,a)$: Sistem atalet kuvveti,

$C(p,v)$:Sistem sönüm kuvveti,

$K(p)$:Sistem katılık kuvveti,
 $F(p,v,t)$: Dış kuvvet,
 p,v,a : Sırasıyla konum, hız ve ivme vektörleri,
 t : zaman

olarak alınmıştır. Her iki integral yönteminde de sistem geometrisi her zaman adımı için yeniden hesaplanır.

Programın kullandığı açık integral yönteminde sabit zaman adımlı ileri Euler yöntemi kullanılır. Simülasyon başlangıcında bütün nesnelerin başlangıç pozisyonları düğümler ve bağlantılar dahil statik analizden bilinmektedir. Kuvvetlerin ve momentlerin aşağıda belirtilen özellikleri içerdiği düşünülür (Orcina, 1987-2013) :

- Ağırlık
- Sephiye
- Hidrodinamik ve aerodinamik sürtünme
- Hidrodinamik ek kütle etkileri (Morison denklemi kullanılarak hesaplanmıştır)
- Gerilme ve kesme
- Eğilme momenti ve tork
- Deniz dibi sürtünme ve tepkisi
- Diğer nesneler ile olan temas kuvvetleri
- Bağlantılar ve vinçler tarafından uygulanan kuvvetler

Hareket denklemi (Newton yasası) her serbest cisim ve her bağlantı düğümü için şu hali alır:

$$M(p)a = F(p,v,t) - C(p,v) - K(p) \quad (3.2)$$

Bu sistem için genel bir hareket denklemi değildir. Fakat her cisim ve bağlantı için yerel hareket denklemdir. Bu hareket denklemlerinin çözümü yalnızca 3'e 3 veya 6'a 6 matrislerle mümkündür. Bu denklem başlangıç zaman adımında ivme vektörü için çözülmüş ve daha sonra Euler denklemi kullanılarak birleştirilmiştir. Konum, hız

ve ivme vektörlerini t anında sırasıyla p_t , v_t ve a_t olarak kabul edersek $t+1$ anında bu değerler şu şekilde bulunur:

$$v_{t+1} = v_t + dt.a_t \quad (3.3)$$

$$p_{t+1} = p_t + dt.v_{t+1} \quad (3.4)$$

burada dt zaman adımıdır.

Her zaman adımı sonunda, bütün düğümler ve serbest cisimler için konumlar belirlenir ve işlem tekrarlanır. Birçok nesne içeren OrcaFlex programında hesaplamaları en zahmetli elemanlar bağlantı hatlarıdır. Bağlantı hattı içeren birçok modelde dinamik analiz için gereken zaman yaklaşık olarak toplam düğüm sayısının toplam zaman adımı ile çarpılmasıdır. Zaman adımı önerilen değerde kalır ve düğümler bağlantı hatlarına düzgün olarak dağıtılsa analiz için geçen zaman yaklaşık olarak düğüm sayısının karesi olacaktır.

Kapalı integral yönteminde OrcaFlex Chung ve Hulbert (1993) tarafından tanımlanan genelleştirilmiş- α integrasyonunu kullanır. Kuvvetler, momentler, kütle vb. açık integral yöntemindeki yol ile hesaplanır. Daha sonra hareket denklemi sistemi zaman adımının sonunda çözülür. Çünkü, p , v ve a zaman adımının sonunda bilinmeyen durumlar olduğu için iteratif çözüm metodu kullanmak gerekir. Sonuç olarak, her kapalı zaman adımı açık zaman adımına göre daha fazla hesaplama gerektirir. Buna rağmen, kapalı yöntem açık yöntemle göre uzun zaman adımlarından daha istikrarlıdır ve bu da kapalı yöntemin daha hızlı olduğunu gösterir.

OrcaFlex hesapları yaparken bazı kabuller kullanır (Mentes, 2010). Bunlar;

- Halatlara ve şamandıralara gelen hidrodinamik yüklerin hesabı, genişletilmiş Morison denklemi (Morison ve diğ., 1950)
- Ek su kütlesi ve sönüm hesabı (Cummins, 1962, Wichers, 1979)
- Dalga sürüklenme kuvvetleri (Newman, 1974)
- Sabit rüzgar ve akıntı kuvvetlerinin hesabı (OCIMF, 1994)

3.2.1 OrcaFlex programı simülasyon testleri

OrcaFlex ile yapılan modellerden Çizelge 3.3'teki özelliklere sahip modelin değişik rüzgar hızlarında simülasyonları yapılmıştır. Rüzgar hızını artmasına bağlı olarak gerilme değerinin artıp artmadığı gözlemlenmiştir. Rüzgar hızına bağlı gerilme

grafığı Şekil 3.5’te baş iskele halatı, Şekil 3.6’da baş sancak halatına gelen gerilmeler değerlerindeki artış gösterilmiştir. Rüzgar hızı arttıkça gerilme değerinin de artması beklenen bir durumdur.

Çizelge 3.3 : Deneme simülasyonu model özellikleri

Değişkenler	Değerler
en_a	180
h	3,00
w	2-50
c	3,00
t	5,00
a	60,00
l	0,00

Burada;

en_a : Akıntı, dalga ve rüzgarın yönü (açısı) (derece),

h : Dalga yüksekliği (m),

w : Rüzgar hızı (m/s),

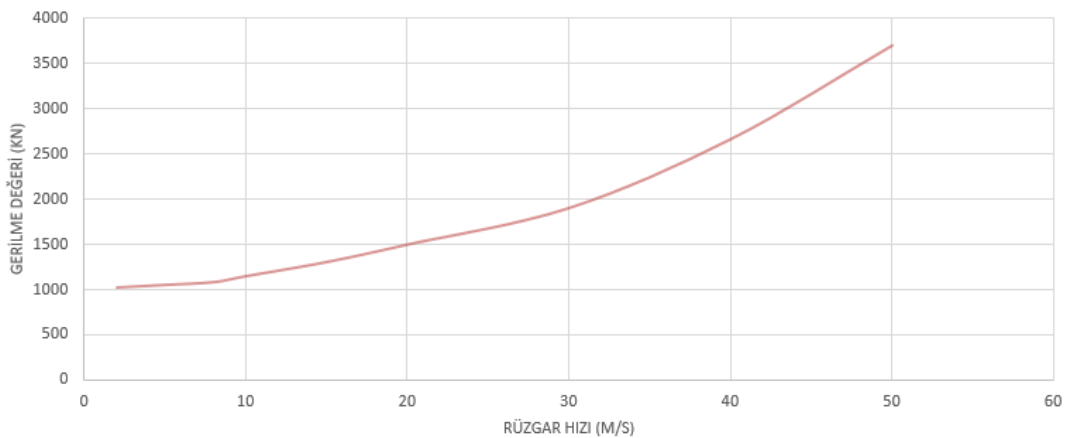
c : Akıntı hızı (m/s),

t : Dalga periyodu (s),

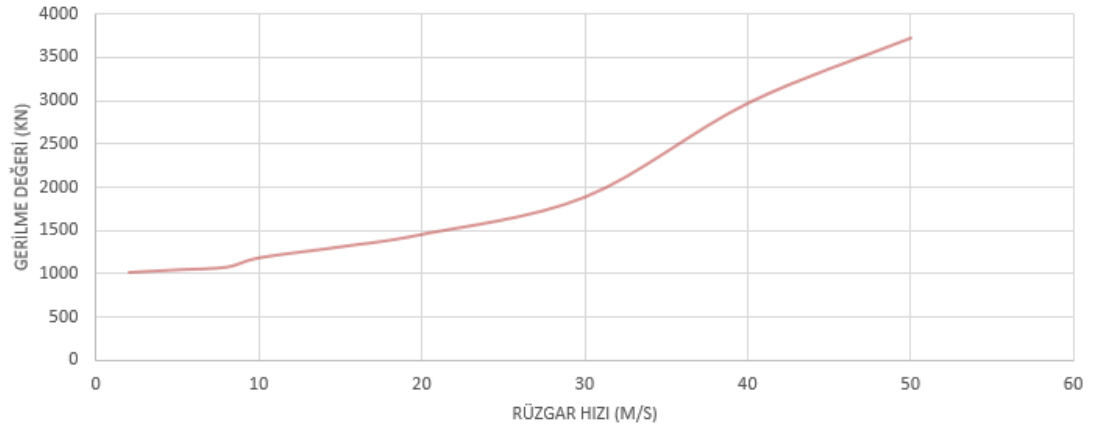
a : Halatın gemiye bağlanma açısı (derece),

l : Bağlama noktasının kabul edilen ilk noktaya uzaklığı (m),

olarak kabul edilmiştir.



Şekil 3.5 : Baş iskele halatında rüzgar hızına bağlı gerilme grafığı.

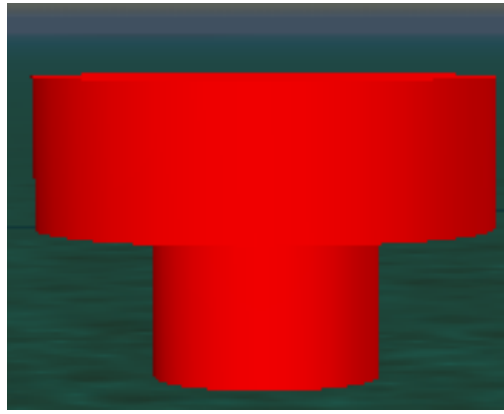


Şekil 3.6 : Baş sancak halatında rüzgar hızına bağlı gerilme grafiği.

4. TANKER-ŞAMANDIRA BAĞLAMA SİSTEMLERİNİN YAPAY SİNİR AĞLARI TEKİNİĞİYLE OPTİMİZASYONU

4.1 Tanker-Şamandıra Bağlama Sistemi Modellemesi ve Simülasyonu

OrcaFlex programı kullanılarak modelleme yapılmıştır. Çizelge 3.1’de özellikleri verilen tanker modellenmiştir. Modellenen tanker baş sancak, baş iskele, kış sancak ve kış iskele olmak üzere toplam 4 noktadan bağlanmıştır. Bağlantıda kullanılan şamandıranın (Şekil 4.1) yüzme şartını sağlayıp sağlamadığının kontrolü yapılmıştır. Şamandıraya bağlı yükselen zincir ve yatak zincirleri de hesaplamaya dahil edilmiştir.



Şekil 4.1 : Şamandıra görünümü.

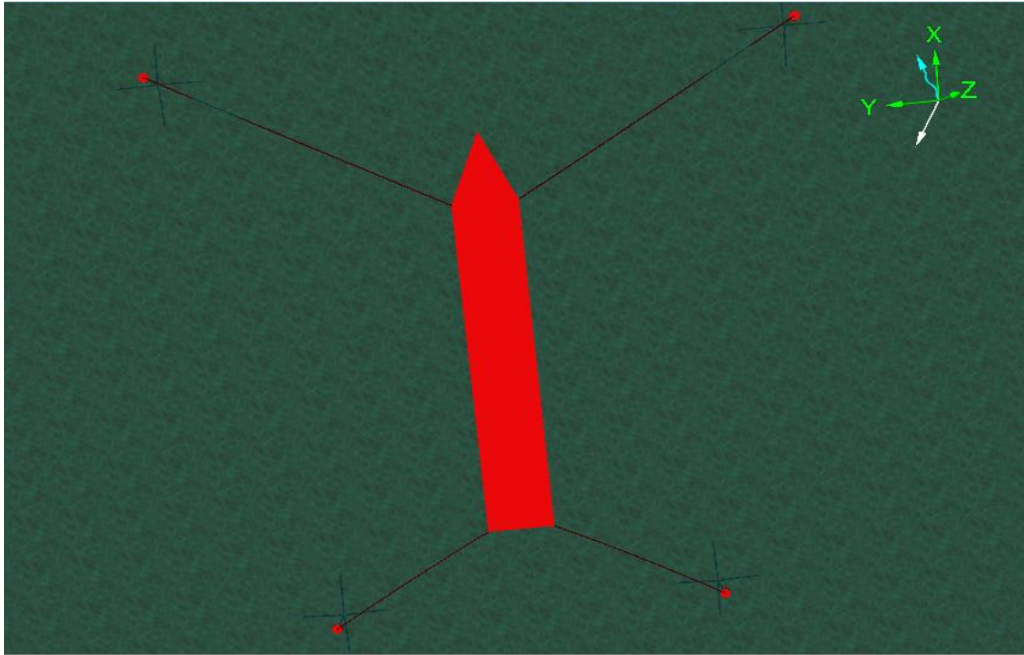
OrcaFlex programında gemi formu oluşturulurken programın kendi içinde bulunan gemi formu esas alınmış ve daha sonra boyut değişiklikleri uygulanmıştır. Boyut değişiklikleri uygulanırken Froud ölçeği kullanılarak RAO katsayıları, ek su kütlesi, sönüm, hidrodinamik direnç, rüzgar direnci, dalga sürüklenme kuvveti gibi katsayılar düzenlenmiştir. OrcaFlex programında kullanılan gemi formunun özellikleri Çizelge 4.1’de verilmiştir (Mentes, 2010).

Çizelge 4.1 : OrcaFlex modeli ana boyutları

OrcaFlex model ana boyutları	
Tam boy (m)	103
Genişlik (m)	15,95
Su çekimi (m)	6,66
Enine GM (m)	1,84
Boyuna GM (m)	114
Blok katsayısı	0,804

Programda su derinliğinin 28 m, deniz dibinin düz, kullanılan dalga tipinin JONSWAP, dalga periyodu 3,5-7,5 saniye, akıntı hesaplama yönteminin power law, dip akıntısı olmadığı, yüzey akıntısının olduğu, rüzgar tipinin sabit esen rüzgar olduğu ve rüzgar, dalga ve akıntının aynı yönden geldiği kabulü yapılmıştır.

Daha sonra oluşturulan 4 noktalı bağlama sistemleri ile simülasyon yapılmıştır. Bağlama açıları, bağlama noktaları, dalga özellikleri, akıntı ve rüzgar hızları ile rüzgar, akıntı ve dalganın yönü değiştirilerek simülasyonlar tekrarlanmıştır. Çizelge 4.2’de yapılan simülasyonlarda kullanılan çevre koşulları, Çizelge 4.3’te yapılan simülasyonlarda elde edilen sonuçlar verilmiştir. Bağlama konumu olarak verilen nokta Şekil 4.2’de gösterilen noktalardan gemi ortasına doğru değişimi göstermektedir.



Şekil 4.2 : İlk bağlama noktası (l=0).

Çizelge 4.2 : OrcaFlex simülasyonları için çevre şartları ve bağlama konumları

Sim. No	en_a	h	w	c	t	a	l
1	180	3	1	3	6	75	40
2	175	2	1	3,5	4	60	5
3	180	3	1	3	6	75	30
4	180	3	1,5	3	6	75	30
5	180	3	1	3	6	75	35
6	170	3	3	2	5,5	45	50
7	180	3	1	3	6	60	20
8	180	3	1	3	6	60	25
9	150	2	1	2	4	60	0
10	20	3	31	1,3	5,5	75	40
11	180	3	1	3	6	30	15
12	15	3	4	1,5	5,5	60	20
13	180	3	1	3	6	60	0
14	20	3	31	1,3	5,5	75	0
15	20	3	31	1,3	5,5	75	35
16	180	3	1	3	6	30	10
17	20	3	20	1,3	5,5	75	5
18	180	3	1	3	6	75	10
19	20	3	31	1,3	5,5	75	30
20	20	3	31	1,3	5,5	75	25
21	180	3	1	3	6	30	0
22	180	3	1	3	6	60	25
23	180	3	1	3	6	75	25
24	30	3	5	2	5,5	60	35
25	20	3	31	1,3	5,5	75	20
26	20	3	31	1,3	5,5	75	5
27	180	3	1	3	6	60	25
28	20	3,5	1,5	3,5	4,5	30	15
29	20	3	31	1,3	5,5	75	10
30	20	3	31	1,3	5,5	75	15
31	50	3	17	2	5,5	60	40
32	20	3	31	1,3	5,5	75	50
33	20	3	31	1,3	5,5	30	0
34	185	3	1	3	6	75	20
35	20	3	31	1,3	5,5	30	30
36	20	3	31	1,3	5,5	30	35
37	20	3	31	1,3	5,5	60	0
38	180	3	1	3	6	75	5
39	20	3	31	1,3	5,5	60	5

Çizelge 4.3 : Simülasyon sonuçları

Sim. No	T1	T2	T3	T4	x	y
1	222,0	6432,8	381,1	2758,5	5,3	0,0
2	1686,8	1686,7	280,5	676,2	7,5	2,1
3	4678,8	185,0	540,8	32,8	5,8	3,0
4	4680,0	185,3	515,7	32,8	5,8	3,0
5	214,3	5459,1	404,8	2087,7	3,6	3,5
6	5545,9	323,2	742,5	1057,0	4,7	3,7
7	143,5	4396,4	129,3	1694,3	6,2	3,9
8	134,7	5063,1	114,8	2076,3	5,8	4,5
9	4111,8	451,6	867,3	1057,9	1,6	4,8
10	747,3	374,6	216,6	3487,6	5,8	4,8
11	5643,1	288,1	297,9	1813,7	3,6	4,8
12	633,7	297,2	310,0	1048,8	5,8	4,8
13	1001,7	1995,9	147,1	603,2	9,0	4,9
14	1073,3	304,3	587,8	2540,3	3,4	4,9
15	782,8	362,0	166,2	3187,8	4,9	5,0
16	5014,2	283,0	388,1	1560,6	3,7	5,0
17	791,0	308,5	163,8	1299,4	2,7	5,1
18	1490,4	1944,3	627,3	764,2	11,3	5,1
19	809,7	359,0	126,4	2942,8	4,6	5,1
20	812,8	348,9	108,6	2683,9	4,6	5,1
21	3824,8	232,3	346,8	942,5	3,4	5,1
22	6431,4	372,4	1073,4	269,7	8,8	5,2
23	291,2	3758,4	510,0	1415,8	4,8	5,2
24	2025,1	228,8	564,4	4251,2	5,8	5,2
25	862,8	348,9	172,2	2495,2	4,5	5,2
26	1016,2	324,9	543,9	2419,3	3,1	5,2
27	130,7	5723,0	68,5	2123,7	5,4	5,2
28	5738,3	756,3	417,5	7995,8	4,7	5,3
29	946,3	326,7	322,8	2425,2	3,2	5,4
30	909,6	340,9	262,9	2371,4	3,9	5,5
31	5126,5	304,2	560,8	5926,4	5,5	5,5
32	784,7	412,1	205,9	4352,8	6,2	5,5
33	1149,8	191,9	333,6	2660,9	3,4	5,6
34	158,6	5013,7	523,7	1791,7	3,5	5,6
35	1684,7	256,3	115,4	3821,8	4,9	5,7
36	1829,5	247,5	113,8	4037,5	4,9	5,7
37	1040,6	343,5	298,9	2518,3	3,5	5,7
38	1651,8	1487,5	656,6	698,2	11,1	5,7
39	1022,7	393,0	145,0	2632,1	4,4	5,8

Burada;

T1 : Baş sancak halatı (kN)

T2 : Baş iskele halatı (kN)

T3 : K1ç iskele halatı (kN)

T4 : K1ç sancak halatı (kN)

x : x yönündeki yer değıştirme miktarı (m)

y : y yönündeki yer değıştirme miktarı (m)

olarak kabul edilmiştir.

Çizelge 4.2 ve 4.3'te veriler YSA eğitiminde kullanılmıştır.

4.2 YSA Tekniğıyle Optimizasyon

OrcaFlex programı ile elde edilen veriler YSA'da eğitim için kullanılmıştır. Fakat veri aralıkları birbirlerinden çok farklı olduğu için 0-1 aralığına normalize edilmiştir. Normalizasyon sonucu Çizelge 4.2'deki değerler Çizelge 4.4'e, Çizelge 4.3'teki değerler ise Çizelge 4.5'teki değerlere dönüşmüştür.

MatLab Neural Network Toolbox (NNT) YSA uygulaması için kullanılmıştır. Kullanılan YSA mimarisi ileri beslemeli mimaridir. Eğitimli öğrenme metodlarından Levenberg-Marquardt ağ eğitiminde kullanılmıştır. Hata düzeltmesi ise en küçük kareler yöntemi ile yapılmıştır.

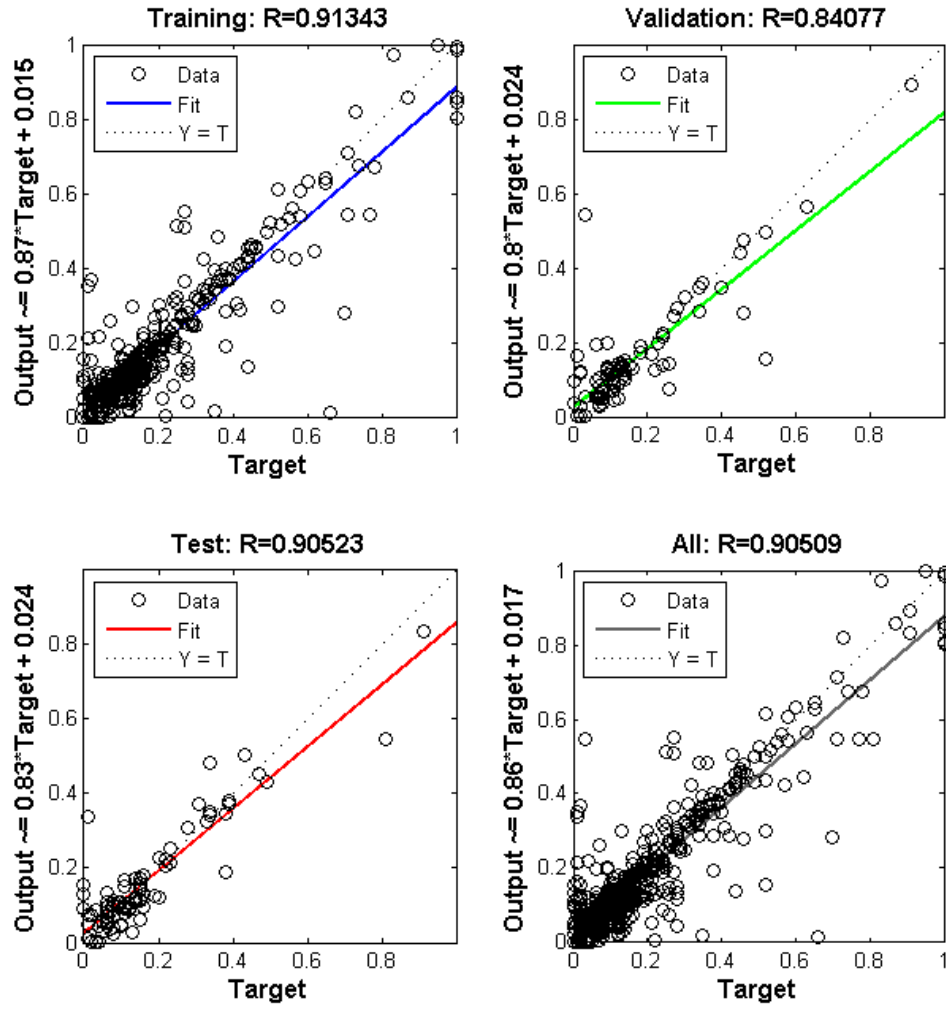
Gizli katmanda 14 nöronlu bir yapı kullanılmış olup aktivasyon fonksiyonu olarak tanjant hiperbolik fonksiyonu seçilmiştir. Şekil 4.3'te gösterildiğı gibi ağ eğitimi sonunda %90 gibi bir başarı yakalanmıştır. Daha sonra bu sonuçlar denormalize edilmiş ve simülasyondan elde edilen değerler ile karşılaştırma amaçlı Şekil 4.4'teki grafik oluşturulmuştur. Grafikte de görüldüğü gibi sonuçların birbirine çok yakın değerler aldığı görülmüştür. Bu da yapay sinir ağlarının başarısını göz önüne koymaktadır.

Çizelge 4.4 : Çevre şartları ve bağlama konumlarının normalizasyon sonuçları

Sim. No	en_a	h	w	c	t	a	l
1	0,895	0,333	0,000	0,800	0,500	1,000	0,571
2	0,868	0,000	0,000	1,000	0,000	0,667	0,071
3	0,895	0,333	0,000	0,800	0,500	1,000	0,429
4	0,895	0,333	0,017	0,800	0,500	1,000	0,429
5	0,895	0,333	0,000	0,800	0,500	1,000	0,500
6	0,842	0,333	0,067	0,400	0,375	0,333	0,714
7	0,895	0,333	0,000	0,800	0,500	0,667	0,286
8	0,895	0,333	0,000	0,800	0,500	0,667	0,357
9	0,737	0,000	0,000	0,400	0,000	0,667	0,000
10	0,053	0,333	1,000	0,114	0,375	1,000	0,571
11	0,895	0,333	0,000	0,800	0,500	0,000	0,214
12	0,026	0,333	0,100	0,200	0,375	0,667	0,286
13	0,895	0,333	0,000	0,800	0,500	0,667	0,000
14	0,053	0,333	1,000	0,114	0,375	1,000	0,000
15	0,053	0,333	1,000	0,114	0,375	1,000	0,500
16	0,895	0,333	0,000	0,800	0,500	0,000	0,143
17	0,053	0,333	0,636	0,114	0,375	1,000	0,071
18	0,895	0,333	0,000	0,800	0,500	1,000	0,143
19	0,053	0,333	1,000	0,114	0,375	1,000	0,429
20	0,053	0,333	1,000	0,114	0,375	1,000	0,357
21	0,895	0,333	0,000	0,800	0,500	0,000	0,000
22	0,895	0,333	0,000	0,800	0,500	0,667	0,357
23	0,895	0,333	0,000	0,800	0,500	1,000	0,357
24	0,105	0,333	0,134	0,400	0,375	0,667	0,500
25	0,053	0,333	1,000	0,114	0,375	1,000	0,286
26	0,053	0,333	1,000	0,114	0,375	1,000	0,071
27	0,895	0,333	0,000	0,800	0,500	0,667	0,357
28	0,053	0,500	0,017	1,000	0,125	0,000	0,214
29	0,053	0,333	1,000	0,114	0,375	1,000	0,143
30	0,053	0,333	1,000	0,114	0,375	1,000	0,214
31	0,211	0,333	0,536	0,400	0,375	0,667	0,571
32	0,053	0,333	1,000	0,114	0,375	1,000	0,714
33	0,053	0,333	1,000	0,114	0,375	0,000	0,000
34	0,921	0,333	0,000	0,800	0,500	1,000	0,286
35	0,053	0,333	1,000	0,114	0,375	0,000	0,429
36	0,053	0,333	1,000	0,114	0,375	0,000	0,500
37	0,053	0,333	1,000	0,114	0,375	0,667	0,000
38	0,895	0,333	0,000	0,800	0,500	1,000	0,071
39	0,053	0,333	1,000	0,114	0,375	0,667	0,071

Çizelge 4.5 : Simülasyon sonuçları normalize edilmiş değerleri

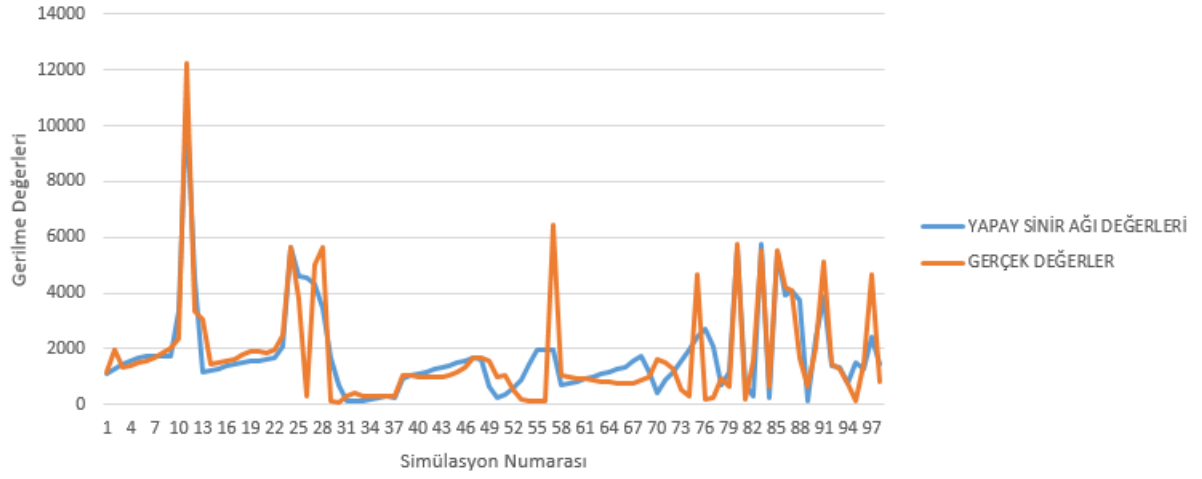
Sim. No	T1	T2	T3	T4	x	y
1	0,011	0,909	0,085	0,244	0,135	0,000
2	0,131	0,224	0,060	0,058	0,214	0,044
3	0,377	0,007	0,125	0,000	0,152	0,064
4	0,377	0,007	0,119	0,000	0,152	0,064
5	0,010	0,769	0,091	0,184	0,073	0,074
6	0,448	0,027	0,175	0,092	0,113	0,080
7	0,005	0,615	0,022	0,149	0,165	0,083
8	0,004	0,712	0,018	0,183	0,152	0,096
9	0,330	0,046	0,207	0,092	0,000	0,102
10	0,054	0,034	0,044	0,309	0,150	0,102
11	0,456	0,022	0,064	0,159	0,071	0,103
12	0,045	0,023	0,067	0,091	0,151	0,103
13	0,075	0,269	0,027	0,051	0,266	0,105
14	0,081	0,024	0,137	0,224	0,065	0,105
15	0,057	0,033	0,031	0,282	0,120	0,107
16	0,405	0,021	0,087	0,137	0,077	0,108
17	0,058	0,025	0,031	0,113	0,040	0,109
18	0,115	0,261	0,147	0,065	0,348	0,109
19	0,059	0,032	0,021	0,261	0,108	0,109
20	0,060	0,031	0,017	0,237	0,110	0,109
21	0,307	0,014	0,076	0,081	0,065	0,110
22	0,521	0,034	0,258	0,021	0,260	0,110
23	0,017	0,523	0,117	0,124	0,115	0,110
24	0,159	0,013	0,131	0,378	0,151	0,111
25	0,064	0,031	0,033	0,220	0,103	0,111
26	0,076	0,027	0,126	0,214	0,054	0,111
27	0,004	0,807	0,007	0,187	0,136	0,111
28	0,464	0,090	0,094	0,713	0,111	0,113
29	0,071	0,028	0,070	0,214	0,056	0,116
30	0,067	0,030	0,056	0,209	0,084	0,117
31	0,414	0,024	0,130	0,528	0,140	0,117
32	0,057	0,040	0,041	0,387	0,168	0,118
33	0,087	0,008	0,073	0,235	0,063	0,119
34	0,006	0,704	0,121	0,157	0,069	0,119
35	0,131	0,017	0,019	0,339	0,118	0,121
36	0,143	0,016	0,018	0,359	0,119	0,121
37	0,078	0,030	0,065	0,223	0,068	0,122
38	0,128	0,195	0,154	0,060	0,344	0,122
39	0,077	0,037	0,026	0,233	0,100	0,123



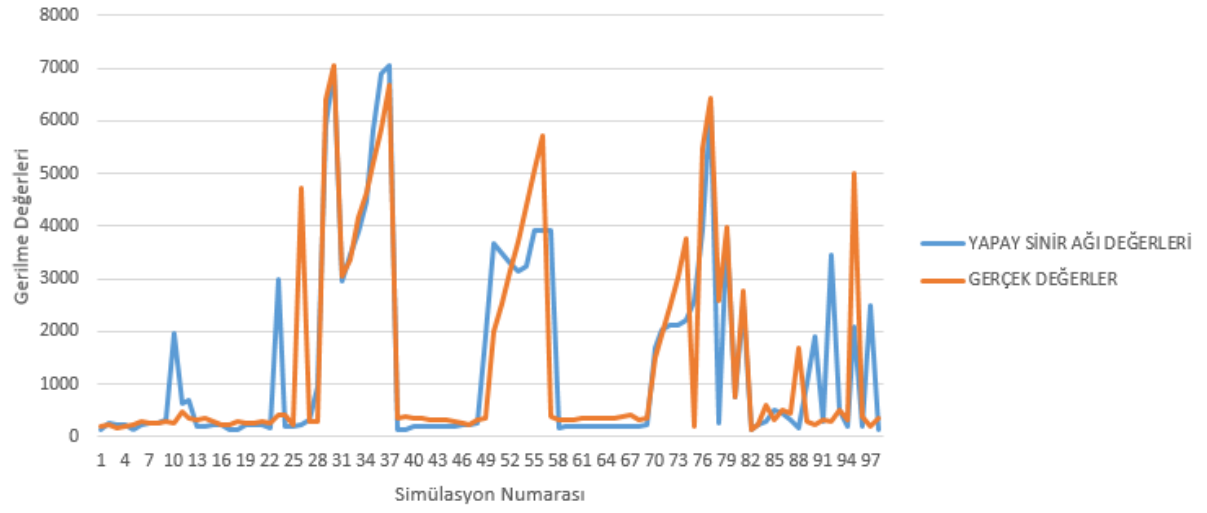
Şekil 4.3 : YSA eğitim sonuçları.

YSA eğitim sonuçlarında eğitim verileri % 91, doğrulama verileri %84 ve test verileri %91 başarı ile sonuçlanmıştır. Verilerin %70'i test verisi olduğu için toplam sonuç test verisine yakın sonuç olan %91 olarak alınmıştır.

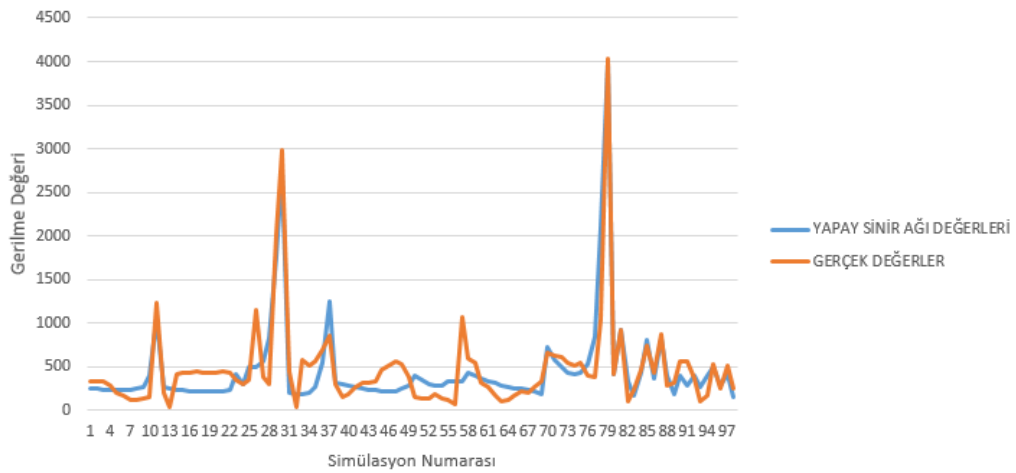
Şekil 4.4'te baş sancak halatı, Şekil 4.5'te baş iskele halatı, Şekil 4.6'da kık iskele halatı ve Şekil 4.7'de kık sancak halatı için simülasyonlardan elde edilen gerilme değerleri ile YSA'dan elde edilen değerlerin karşılaştırılması görülmektedir.



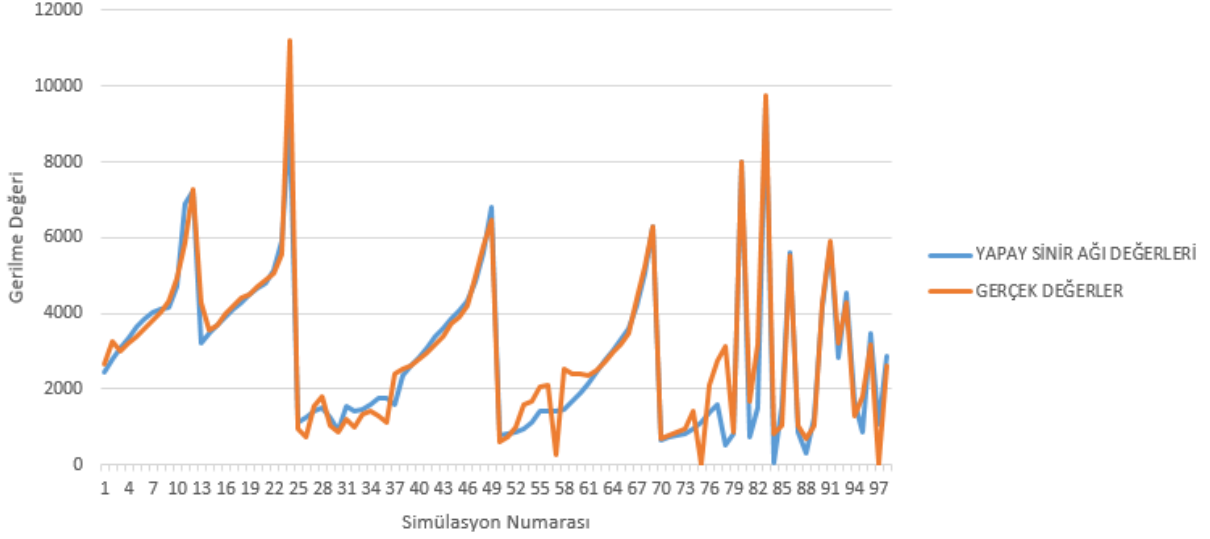
Şekil 4.4 : Baş sancak halatı sonuçlarının karşılaştırılması.



Şekil 4.5 : Baş iskele halatı sonuçlarının karşılaştırılması.



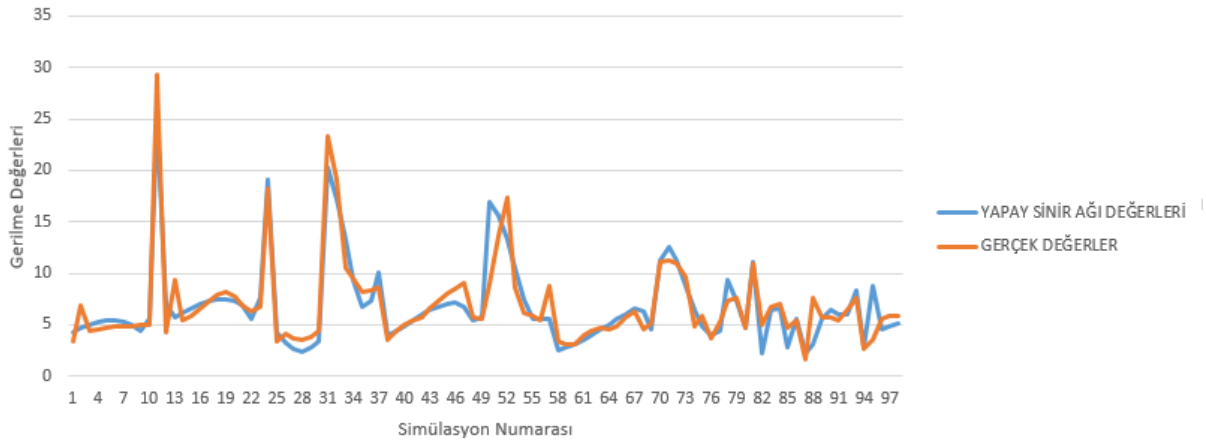
Şekil 4.6 : Kıç iskele halatı sonuçlarının karşılaştırılması.



Şekil 4.7 : Kıç sancak halatı sonuçlarının karşılaştırılması.

Tankerin x ve y yönlerindeki yer değiştirmeleri de YSA algoritmasında hesaplanmıştır. x-yönündeki yer değiştirmeler ile YSA'dan elde edilen yer değiştirme sonuçları Şekil 4.7'de, y-yönündeki sonuçların karşılaştırması ise Şekil 4.8'de gösterilmiştir.

Grafiklerde de %91 doğruluk oranında sonuçların örtüştüğü görülmektedir.

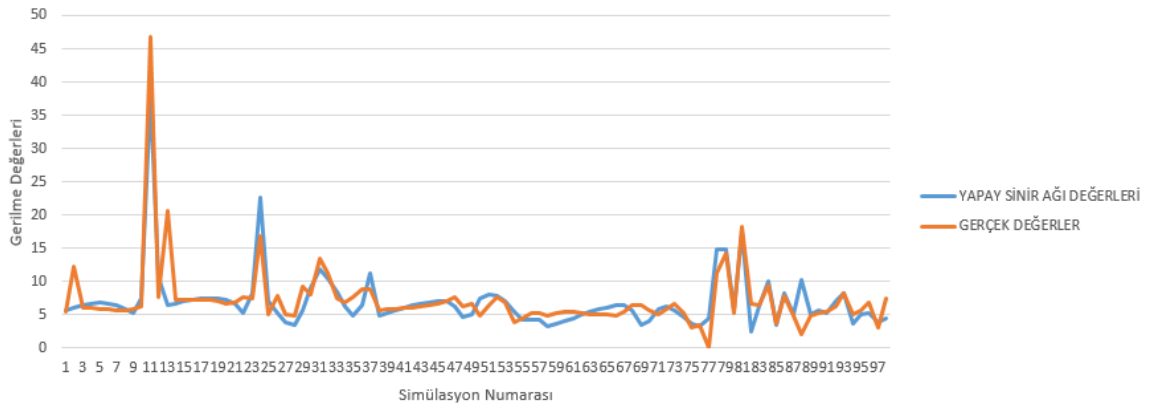


Şekil 4.8 : x-yönündeki yer değiştirme sonuçlarının karşılaştırılması.

Ayrıca MatLab yazılımı kullanılarak 1000 farklı çevre durumu için oluşturulan YSA simüle edilmiş ve Çizelge 4.6'da verilen değerler ile toplam gerilmenin minimum olduğu durum saptanmıştır.

Çizelge 4.6 : Optimum gerilmenin bulunduğu şartlar

Değişkenler	Değerler
en_a	180
h	3,8
w	10
c	1,2
t	8
a	47
l	0



Şekil 4.9 : y-yönündeki yer değiştirme sonuçlarının karşılaştırılması.

5. SONUÇ VE ÖNERİLER

Bu tez çalışmasında tanker-şamandıra bağlama sistemlerinin bağlama halatlarına gelen yükler ve tanker yer değiştirmeleri simülasyon yoluyla elde edildikten sonra yapay sinir ağları ile %91 doğruluk oranına sahip bir algoritma oluşturulmuştur.

Bu oluşturulan algoritma ile 4 noktalı bağlama sistemlerinde istenen girdi verileri kullanılarak OrcaFlex vb. programlarda simülasyon yapmaya gerek olmadan sonuçlar %91 doğruluk seviyesinde elde edilebilecektir.

1000 farklı durum için simülasyon yapılmış ve bu 1000 durum içinden optimum değerler belirlenmiştir.

Yapay sinir ağları ile algoritma oluşturma basit ve hızlı bir işlem olduğundan benzer şekilde farklı bağlama sayıları, farklı tonajlar ve farklı su derinlikleri ile de bu işlem yapılabilir. Ayrıca TLP, SPAR, yarıbatık gibi açık deniz yapılarında da uygulanabilir.

İleriki çalışmalarda yapay sinir ağları metodu bulanık mantık ile desteklenerek bulanık sinir ağları tabanlı açık deniz yapısı seçimi yapılabilir.

6. KAYNAKLAR

- ABS (2009)**, Rules for Building and Classing Steel Vessels, Part 3, *Hull Construction and Equipment*.
- Anderson, J.A.** (1983), Cognitive and psychological computation with neural models, *IEEE Transactions on Systems, Man and Cybernetics*, **SMC-13**, 799-814.
- Barltrop, N.D.P.** (1998), Floating structures: a guide for design and analysis, 1 & 2, OPL, Ledbury, England
- Chung, J. Ve Hulbert G.M.** (1993), A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized- α method. *ASME Journal of Applied Mechanics*, **60**, 371-375.
- Carbono, A.J.J., Menezes, I.F.M., Martha, L.F.** (2005), Mooring pattern optimization using genetic algorithms, 6th World Congresses of Structural and Multidisciplinary Optimization, Rio de Janeiro, Brazil.
- Caudill, M.** (1988), Neural Networks Primer Part III, *AI Expert*, 53-59.
- Civalek, Ö. ve Çatal, H.H.** (2004), Geriye yayılma yapay sinir ağları kullanılarak elastik kirişlerin statik ve dinamik analizi, DEÜ Mühendislik Fakültesi Fen ve Mühendislik Dergisi, **1**, 1-16.
- Cummins, W.E.** (1962), The Impulse Response Function and Ship Motions, *Schiffstechnik*, **9**, 101-109.
- Farley, B.G. and Clark, W.A.** (1954), Simulation of self-organizing systems by digital computer.
- Guarize, R., Matos, N.A.F., Sagrilo, L.V.S., Lima, E.C.P.** (2007). Neural networks in the dynamic response analysis of slender marine structures, *Applied Ocean Research*, **29**, 191-198.
- Hamzaçebi, C.** (2011), Yapay Sinir Ağları: Tahmin amaçlı kullanımı, MatLab ve Neurosolutions uygulamalı, 107-115.
- Hebb, D.O.** (1949). The organization of behaviour, The first stage of perception: growth of the assembly, MIT Press., **4**, 44-56.
- Hopfield, J.J.** (1982), Neural networks and physical systems with emergent collective computational abilities, *Proc. Natl. Acad. Sci.*, **79**, 2554-2558.
- Jacob, B.P., Pina, A.C., Albrecht, C.H., Lima, B.S.L.P.** (2014). Wavelet network meta-models for the analysis of slender offshore structures, *Engineering Structures*, **68**, 71-84.
- Kohonen, T.** (1982), Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, **43**, 59-69.
- Kohonen, T.** (1988), "An Introduction to Neural Computing" *Neural Networks*, **1** 4.

- Li, M., Mehrotra, K., Mohan, C. K., Ranka, S.** (1990), Forecasting sunspot numbers using neural networks, *Electrical Engineering and Computer Science Technical Reports*, **67**.
- Mahfouz, A.B.**, (2007). Predicting the capability-polar-plots for dynamic positioning systems for offshore platforms using artificial neural networks, *Ocean Engineering*, **34**, 1151-1163
- McCulloch, W.S. and Pitts, W.** (1943), A logical calculus of the ideas immanent in nervous activity, *Bulletin of Mathematical Biophysics*, **5**, 1-19.
- Mentes, A.** (2010). Açık Deniz Yapıları Bağlama Sistemlerinin Dizaynında Bulanık Çok Kriterli Karar Verme Yöntemlerinin Uygulanması, (doktora tezi), İTÜ Fen Bilimleri Enstitüsü, İSTANBUL
- Morrison, J.R., O'Brien, M.D., Johnson, J.W., Schaaf, S.A.** (1950), The force exerted by surface waves on piles, *Petrol Trans.*, AIME, **189**.
- Newman, J.N.** (1974), Second order, slowly varying forces on vessels in irregular waves, *International Symposium in the Dynamics of Marine Vehicles in Waves*, University College, London.
- OCIMF** (1994), Prediction of wind and current loads on VLCCs, *Second Edition*, London.
- OrcaFlex User Manual**, 1987-2013, *Version 9.8a*, Copyright Orcina Ltd.
- Ozkan, C. Ve Sunar Erbek, F.** (2003), The Comparison of Activation Functions for Multispectral Landsat TM Image Classification, *Photogrammetric Engineering & Remote Sensing*, **69**, 1225-1234.
- Özdemir, A.** (2013), Yapay sinir ağları metodu kullanılarak iris bitki türünün sınıflandırılması, (yüksek lisans tezi), Gaziantep Üniversitesi Matematik Bölümü, GAZİANTEP.
- Özhan, E. ve Abdalla, A.** (2002), Türkiye Kıyıları Rüzgar ve Derin Deniz Dalga Atlası, 151-152.
- Öztemel, E.** (2003), Yapay Sinir Ağları, Birinci Baskı, İstanbul, 31-36.
- Pina, A.C., Pina, A.A., Albrecht, C.H., Lima, B.S.L.P., Jacob, B.P.** (2013). ANN-based surrogate for the analysis of mooring lines and risers, *Applied Ocean Research*, **41**, 76-86.
- Rojas, R.** (1996), Neural Networks: A Systematic Introduction, *Springer-Verlag*, Berlin, 156-162.
- Rumelhart, D.E., Hinton, G.E., Williams, R.J.** (1986), Learning representations by back propagation error, *Nature*, **32**, 533-536.
- Svozil, D., Kvasnicka, V., Pospichal, J.** (1997), Introduction to multi-layer feed-forward neural networks, *Chemometrics and Intelligent Laboratory Systems*, **39**, 43-62.
- Şen, Z.** (2004), Yapay Sinir Ağları İlkeleri, 1, 12-13.
- Topaloğlu, V.**, (2007). *Yapay Sinir Ağları İle Dalga Yüksekliği Tahmini*, (doktora tezi), İTÜ Fen Bilimleri Enstitüsü, İSTANBUL.

- Uçan, O.N., Danacı, E. Bayrak, M.,** (2003), İşaret ve Görüntü İşlemede Yeni Yaklaşımlar:Yapay Sinir Ağları, *İ.Ü. Mühendislik Fakültesi Yayınları*, ISBN: 975-404- 686-7, 45-68.
- Wichers, J.E.W.** (1976), On the slow motions of tankers moored to single point mooring systems, *Offshore Technology Conference, OTC-2548*, 283-303.
- Widrow, B. and Hoff, M.E.** (1960). Adaptive switching circuits, 2-7.
- Yetkin, M., Menten, A., Yalçın, İ., Helvacıoğlu, İ.H., Türkoğlu, N., Akıldız, H.** (2013), Açık deniz petrol platformu modellenmesine yapay sinir ağları yaklaşımı, 5. Denizcilik Kongresi, İSTANBUL.

ÖZGEÇMİŞ

Ad Soyad : Murat YETKİN
Doğum Yeri ve Tarihi : Nazilli/AYDIN 09.08.1989
E-Posta : yetkinmu@itu.edu.tr

ÖĞRENİM DURUMU:

- **Lisans** : 2012, İTÜ, Gemi ve Deniz Teknolojisi Mühendisliği

PROFESYONEL DENEYİM: Araştırma Görevlisi, İTÜ Gemi İnşaatı ve Deniz Bilimleri Fakültesi (Aralık 2013-)