

T.C.
YOZGAT BOZOK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
MEKATRONİK MÜHENDİSLİĞİ ANABİLİM DALI

YÜKSEK LİSANS

FOİLSİM VERİLERİ İLE EĞİTİLMİŞ SİNİR AĞI
DESTEKLİ YAPAY ARI KOLONİSİ ALGORİTMASI İLE
KANAT PROFİL OPTİMİZASYONU

ŞEYMA DOĞAN

Tez Danışmanı

Dr. Öğr. Üyesi Cemil ALTIN

YOZGAT- 2022

T.C.
YOZGAT BOZOK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
MEKATRONİK MÜHENDİSLİĞİ ANA BİLİM DALI

YÜKSEK LİSANS

FOİLSİM VERİLERİ İLE EĞİTİLMİŞ SİNİR AĞI
DESTEKLİ YAPAY ARI KOLONİSİ
ALGORİTMASI İLE KANAT PROFİL OPTİMİZASYONU

ŞEYMA DOĞAN

Tez Danışmanı

Dr. Öğr. Üyesi Cemil ALTIN

YOZGAT- 2022

TEZ BEYANI

Tez yazım kurallarına uygun olarak hazırlanan bu tezin yazılmasında bilimsel ahlak kurallarına uyulduğunu, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezin içerdiği yenilik ve sonuçların başka bir yerden alınmadığını, kullanılan verilerde herhangi bir tahrifat yapılmadığını, tezin herhangi bir kısmının bu üniversite veya başka bir üniversitedeki başka bir tez çalışması olarak sunulmadığını beyan eder, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

ŞEYMA DOĞAN

10/08/2022

ÖZET

YÜKSEK LİSANS TEZİ

FOILSİM VERİLERİ İLE EĞİTİLMİŞ SİNİR AĞI DESTEKLİ YAPAY ARI KOLONİSİ ALGORİTMASI İLE KANAT PROFİL OPTİMİZASYONU

ŞEYMA DOĞAN

YOZGAT BOZOK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

MEKATRONİK MÜHENDİSLİĞİ ANA BİLİM DALI

TEZ DANIŞMANI: DR. ÖĞR. ÜYESİ CEMİL ALTIN

Bu çalışmada hesaplanması ve belirlenmesi zor olan kanat profilinin FoilSim verileri ve optimizasyon algoritmaları yardımıyla eniyilenmesi üzerine çalışılmıştır. NASA (National Aeronautics Space Administration) tarafından sunulan ve birçok araştırmacının özellikle model uçak geliştirmekte kullandığı FoilSim verilerinin uçak kanat profil şekli optimizasyonunda kullanılması sağlanmıştır. FoilSim, web ortamında çalışmakta olan tasarımcılara uçak kanadına ait parametrelere göre kaldırma (L) ve sürüklenme (D) miktarları hakkında bilgi veren bir web yazılımıdır. Optimizasyon süreçlerinde kullanılabilmesi için FoilSim web sayfasından açı, kamburluk, kalınlık ve kaldırma katsayısı (C_L) değerlerine sahip 8000 örnekten oluşan özgün veri seti oluşturulmuştur. Bu veri seti, üç girişli (açı, kamburluk, kalınlık) ve bir çıkışlı (C_L ; Kaldırma katsayısı) İleri Beslemeli Yapay Sinir Ağı (YSA) ile MATLAB programında eğitilmiştir. Yapay sinir ağının eğitilmesinde 5-katlamalı çapraz doğrulama yöntemi kullanılmıştır. Vekil ağa ait doğruluk oranı %99,9 olarak tespit edilmiştir. Vekil ağ için bulunan doğruluk oranı kaldırma denkleminde yer alan kaldırma değeri için de geçerlidir. Tez çalışmasının ikinci bölümünde YSA ile eğitilen bu ağ, Yapay Arı Kolonisi (YAK) algoritmasının kullanılması için optimizasyon sürecinde farklı açı, kamburluk ve kalınlık değerleri için kaldırma katsayısı üretmektedir. Böylece tasarımcının istediği kaldırma kuvveti için en uygun açı, kamburluk ve kalınlık değerleri bulunmaktadır. Bulunan bu ideal değerler *FoilSim III Elementary Version 1.1.c* programında test edilmiş ve yeterli doğrulukta çalıştığı saptanmıştır. Kısacası bu tez çalışmasında FoilSim verileri ile optimizasyon algoritmaları kullanılmıştır. Tasarımcının istediği kaldırma kuvveti değerini karşılayan kanada ait belirlenmesi ve hesaplanması zor olan en uygun açı, kamburluk ve kalınlık değerlerinin belirlenmesi konusunda destek olunmuştur. Böylece tasarım ve karar verme sürecine yardımcı olunarak verimli hava araçları üretimine katkı sağlanmıştır.

2022, xi+ 87 SAYFA

ANAHTAR KELİMELEER: Kanat Profili, Optimizasyon, YAK Algoritması, YSA Algoritması, MATLAB.

ABSTRACT

MASTER THESIS

AIRFOIL OPTIMIZATION WITH ARTIFICIAL BEE COLONY ALGORITHM SUPPORTED BY NEURAL NETWORK TRAINED WITH FOILSIM DATA

ŞEYMA DOĞAN

**YOZGAT BOZOK UNIVERSITY
SCHOOL OF GRADUATE STUDIES**

MECHATRONIC ENGINEERING DEPARTMENT

SUPERVISOR: ASST. PROF. DR. CEMİL ALTIN

In this study, it has been studied on the optimization of the wing profile, which is difficult to calculate and determine, with the help of FoilSim data and optimization algorithms. FoilSim data provided by NASA (National Aeronautics Space Administration) and used by many researchers, especially in developing model airplanes, has been provided to use in aircraft wing shape optimization. FoilSim is web software that works in the web environment and provides information to the designers about the amount of lift (L) and drag(D) according to the parameters of the aircraft wing. In order to be used in optimization processes, an original data set consisting of 8000 samples with angle, camber, thickness and lift coefficient (C_L) values was created from the FoilSim web page. This dataset is trained in MATLAB program with a Feed Forward Artificial Neural Network (ANN) with three inputs (angle, camber, thickness) and one output (C_L). The artificial neural network was trained using the 5-fold cross validation method. The accuracy rate of the surrogate network has been determined as 99,9%. The accuracy rate found for the surrogate network also applies to the lift value in the lift(L). In the second stage, this trained network generates a lift coefficient for different angles, camber, and thicknesses during the optimization process for use by the Artificial Bee Colony (ABC) algorithm ~~to use~~. Thus, the most suitable angle, camber and thickness values are found for the lifting force desired by the designer. These ideal values have been tested in *FoilSim III Elementary Version 1.1c* and found to work with sufficient accuracy. In short, in this study, FoilSim data and optimization algorithms were used. Support was provided for the determination of the wing that meets the lift force value requested by the designer and the determination of the most suitable angle, camber and thickness values, which are difficult to calculate. Thus, it contributed to the production of efficient aircraft by assisting in the design and decision-making process.

2022, xi+ 87 PAGE

KEYWORDS: Wing Profile, Optimization, ABC Algorithm, ANN Algorithm, MATLAB.

ÖNSÖZ

Tez çalışmamın her aşamasında beni çalışmaya teşvik eden, bilgi ve birikimleriyle bana her daim yol gösteren değerli danışmanım Sayın Dr. Cemil ALTIN' a, akademik tecrübelerinden faydalandığım, her aşamada bana rehberlik eden çok kıymetli arkadaşım Şahin BAYRAKTAR'a, maddi ve manevi destekleriyle yanımda olan aileme, tez çalışma sürecimin her aşamasında yanımda olup beni motive eden, akademik bilgi ve birikimleriyle beni destekleyen sevgili eşim Mustafa Kemal DOĞAN'a ve çalışmam sırasında küçük veya büyük yardımını esirgemeyen herkese sonsuz teşekkürlerimi sunarım.

ŞEYMA DOĞAN

06/07/2022

İÇİNDEKİLER

Sayfa

TEZ BEYANI.....	i
ÖZET	ii
ABSTRACT.....	iii
ÖNSÖZ	iv
İÇİNDEKİLER	v
TABLoların LİSTESİ	vii
ŞEKİLLERİN LİSTESİ.....	viii
SİMGELER VE KISALTMALAR	x
1. GİRİŞ	1
2. LİTERATÜR ARAŞTIRMASI	6
3. MATERYAL VE YÖNTEM.....	21
3.1. Uçak Aerodinamiği ve Kanat Profili	21
3.2. Kanat Profili Geometrisi.....	22
3.3. NASA FoilSim Web Sayfası	23
3.4. Yapay Sinir Ağları (YSA)	24
3.4.1. Biyolojik Sinir Hücresi ve Yapay Sinir Ağının Temel Öğeleri	26
3.4.2. Yapay Sinir Ağının Yapısı.....	30
3.4.3. Yapay Sinir Ağlarının Genel Özellikleri	31
3.4.4. Yapay Sinir Ağlarında Öğrenme Süreci	32
3.4.5. Yapay Sinir Ağı için Verilerin Toplanması	34
3.4.6. Yapay Sinir Ağında Yer Alan Hiperparametreler.....	34
3.4.7. Yapay Sinir Ağının Eğitilmesi.....	38
3.5. Optimizasyon.....	40
3.5.1. Optimizasyon Problemlerinin Temel Parçaları.....	40
3.5.2. Optimizasyon Problemlerine ait Sınıflandırmalar	41
3.5.3. Sezgisel Optimizasyon.....	42
3.5.4. Sürü Tabanlı Sezgisel Algoritma Çeşitleri.....	43
3.5.4.1. Karınca Koloni Algoritması	44
3.5.4.2. Parçacık Sürü Optimizasyonu	45

3.5.4.3. Yapay Balık Sürüsü Algoritması.....	46
3.5.4.4. Gri Kurt Optimizasyonu Algoritması.....	46
3.5.4.5. Balina Optimizasyon Algoritması	46
3.5.4.6. Guguk Kuşu Algoritması.....	48
3.5.4.7. Yapay Arı Kolonisi Algoritması.....	49
3.5.4.7.1. YAK Algoritmasının Parametrelerinin Başlatılması : Adım 1	53
3.5.4.7.2. Başlangıç Besin Kaynağı Pozisyonlarının Üretilmesi: Adım 2.....	53
3.5.4.7.3. İşçi Arı Fazı : Adım 3	53
3.5.4.7.4. Olasılık Değerlerinin Hesaplanması: Adım 4	54
3.5.4.7.5. Gözcü Arı Fazı: Adım 5.....	54
3.5.4.7.6. Kaşif Arı Fazı: Adım 6	55
3.5.4.7.7. Sonlandırma Koşulları: Adım 7	55
4. BULGULAR.....	61
5. SONUÇ VE ÖNERİLER	65
6. KAYNAKLAR	69
EKLER.....	78
EK 1. Farklı değerlere sahip hiperparametreler ile vekil YSA yapısı testleri.	78
ÖZGEÇMİŞ	

TABLoların LİSTESİ

Tablo	<u>Sayfa</u>
Tablo 2.1. Uçak kanat profil optimizasyonu üzerine literatür araştırması.....	14
Tablo 2.2. FoilSim'den yararlanılarak yapılan çalışmalara dair literatür araştırması. .	18
Tablo 3.1. Yapay sinir hücresi ve biyolojik sinir hücresi benzeştirilmesi.....	28
Tablo 3.2. Aktivasyon fonksiyonlarının matematiksel olarak gösterimi.....	30
Tablo 3.3. Yapay sinir ağının eğitimde değerlendirilen hiperparametreler.	37
Tablo 3.4. Yapay sinir ağında kullanılan hiperparametreler ve değerleri.	38
Tablo 3.5. Yapay Sinir ağı için kullanılan durdurma kriterleri.	39
Tablo 3.6. Optimizasyon Sınırlılıkları	58
Tablo 3.7. Optimizasyon Algoritması (YAK)'nın kontrol parametreleri	59
Tablo 4.1. Açık, kamburluk ve kalınlık değerlerinin gözlemlenmesi (Kaldırma Değeri = 8 Ibs olduğu durumda).	61
Tablo 4.2. Açık, kamburluk ve kalınlık değerlerinin gözlemlenmesi (Kaldırma Değeri =17 Ibs olduğu durumda).	62
Tablo 4.3. Açık, kamburluk ve kalınlık değerlerinin gözlemlenmesi (Kaldırma Değeri = 52 Ibs olduğu durumda).	63
Tablo 4.4. Açık, kamburluk ve kalınlık değerlerinin gözlemlenmesi. (Kaldırma Değeri = 157 Ibs olduğu durumda).	63

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 1.1. Uçağa etki eden aerodinamik kuvvetler	1
Şekil 1.2. Kanat üzerindeki aerodinamik kuvvet	2
Şekil 3.1. Kanadın yandan kesit görüntüsü: İki boyutlu kanat profili	21
Şekil 3.2. İki boyutlu akım modeli	22
Şekil 3.3. Kanat profil geometrisi	22
Şekil 3.4. Hücüm açısı	23
Şekil 3.5. FoilSim web sayfası.....	24
Şekil 3.6. Beyin üzerindeki temel duyu bölgeleri.....	26
Şekil 3.7. Biyolojik sinir hücresi	27
Şekil 3.8. Biyolojik sinir hücresi ve sinaps bölgesi	27
Şekil 3.9. Yapay sinir hücresi	28
Şekil 3.10. Yapay sinir ağı.....	31
Şekil 3.11. Holdout yöntemiyle verinin bölümlendirilmesi.	36
Şekil 3.12. K-katlamalı çapraz doğrulama.....	36
Şekil 3.13. Sezgisel yöntemler.....	43
Şekil 3.14. Karıncaların yiyecek arama davranışı	45
Şekil 3.15. Balinaların kabarcık stratejisi ile avlanma yöntemleri.	48
Şekil 3.16. YAK algoritması teorik akış diyagramı.....	52
Şekil 3.17. YAK algoritması uygulama diyagramı.....	56
Şekil 5.1. Ağa ait nntraintool arayüzü.	65
Şekil 5.2. Ağa ait regresyon analizi.	66
Şekil 5.3. Ağa ait performans görseli.	67

SİMGELER VE KISALTMALAR

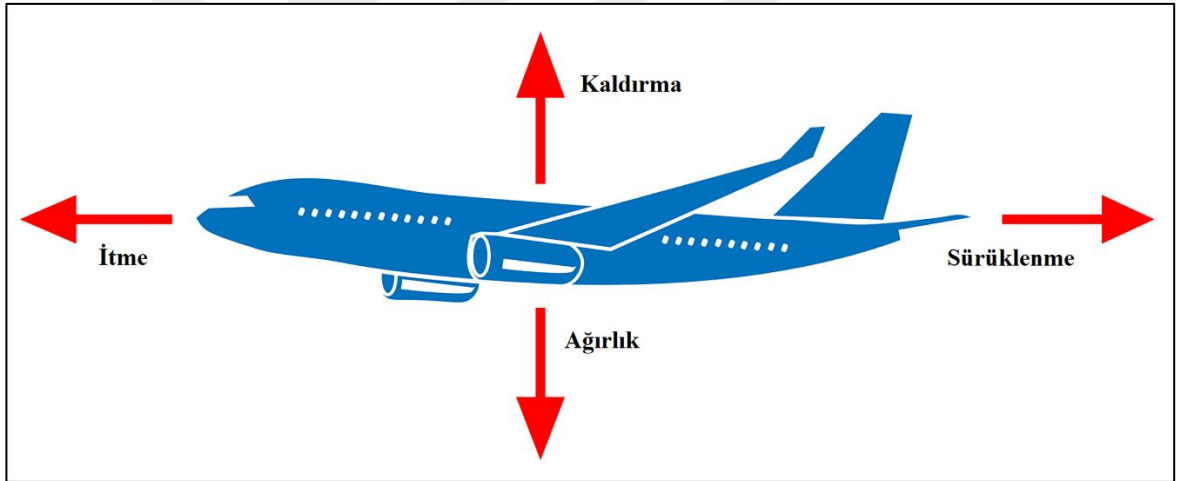
Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklamalar
A	: Kanat alanı
C_D	: Sürüklenme katsayısı
C_L	: Kaldırma katsayısı
D	: Sürüklenme değeri
$failure_i$	: Geliştirilememe sayacı
$fitness_i$	: Çözümün uygunluk değerinin hesaplanması
L	: Kaldırma kuvveti değeri
$limit$	: Denemelerle iyileştirilemeyen bir besin kaynağının terk edilmesinden sorumlu değer
M	: Optimize edilecek parametre sayısı
$MaxCycle$	: Yiyecek arama için maksimum döngü sayısı
ρ	: Hava yoğunluğu
p_i	: Göreceli seçim olasılığı
$\emptyset_{ij}$	: [-1,1] aralığında seçilen rastgele değer
S	: Sürüklenme değeri
SN	: Yiyecek kaynağı sayısı
V	: Kanat profili üzerindeki hava hızı
vb	: Ve benzeri
vd	: Ve diğerleri
x_j^{max}	: j.parametrenin üst sınırı
x_j^{min}	: j.parametrenin alt sınırı
x_{kj}	: Rastgele seçilen bir çözüm

Kısaltmalar	Açıklamalar
ADALINE	: Adaptive Lineer Neuron
BOA	: Balina Optimizasyon Algoritması
DİA	: Derin İnanç Ağı
DSO	: Doğrudan Sayısal Optimizasyon
GA	: Genetik Algoritma
GKA	: Gri Kurt Algoritması
HAD	: Hesaplamalı Akışkanlar Dinamiği
İHA	: İnsansız Hava Aracı
KKA	: Karınca Kolonisi Algoritması
MADALINE	: Many Adaptive Lineer Neuron
PSO	: Parçacık Sürü Optimizasyonu
ReLU	: Doğrultulmuş Lineer Birim
YAK	:Yapay Arı Kolonisi Algoritması
YSA	:Yapay Sinir Ağı

1. GİRİŞ

Son yıllarda havacılık teknolojisi uçuşla ilgili birçok gelişme sebebiyle hızlı bir büyüme gerçekleştirmiştir. İleriki zamanlarda da bu büyümenin devam etmesi beklenmektedir. Bu büyüme göz önünde bulundurularak havacılık sektöründeki talepleri karşılayabilmek için bu alanda yapılacak çalışmalara ihtiyaç duyulmaktadır (Kumar, Prakash, Mishra ve Kumar, 2020). Havacılık sektöründeki talepler artıka tüketim ve çevreye verilen zararın etkisi de bir o kadar artmaktadır (Baharözü, Soykan ve Özerdem, 2017). Enerji verimliliği, hava araçlarının enerji tüketimini azaltmak, maliyeti en aza indirmek ve aynı zamanda çevreye verdiği zararları azaltmayı hedeflediği için birçok araştırmacının ilgi odağı olmuştur. Bu konuda çevresel etkiyi en aza indirirken hız ve kapasiteyi artırmak üzerinde durulan bir konu haline gelmiştir (Afonso, Vale, Lau ve Suleman, 2017).



Şekil 1.1. Uçağa etki eden aerodinamik kuvvetler (Aerodynamic forces, <https://www.aerorann.com/basics-of-aerodynamics-aeroscience.>).

Aerodinamik, nesnelerin havadaki kuvvetleri ve hareketlerinin incelenmesidir. Şekil 1.1 de gösterildiği gibi bu kuvvetler kaldırma, ağırlık, sürüklenme, itme olarak adlandırılır. İtme, bir cismi ya da uçan gövdeye ileri hareketi veren yönünün motora nasıl bağlandığına göre değişen uçaktaki bir tahrik sistemi tarafından üretilen kuvvettir (Kumar vd., 2020). Diğer bir ifade ile itme, bir cismi ileri hareket ettiren kuvvettir, bu kuvvet sürüklenmenin tersi olan kuvvettir. Uçaklar bu itme gücünü motordan alarak sürüklenme kuvvetine karşı koymuş olur, kanatların çevresinde bir hava akımının oluşması ve havada tutunacak güce sahip olması sağlanır.

Kaldırma, uçağın havada uymasını sağlayan ve yer çekimine zıt yönde hareket etmesini ve toplam ağırlığın üstesinden gelmesi için oluşan kuvvettir. Kanadın sahip olduğu yapı sayesinde kanadın alt ve üst yüzeyinde basınç farkı oluşması ve kaldırma kuvvetinin kanada etki etmesiyle birlikte uçakların havalanması sağlanmış olur. Kaldırma denklemi Eşitlik 1.1 de verilmiştir (The Lift Equation, <https://www.grc.nasa.gov/www/k-12/rocket/lifteq.html>).

$$L = \frac{\rho \cdot A \cdot C_L \cdot V^2}{2} \quad (1.1)$$

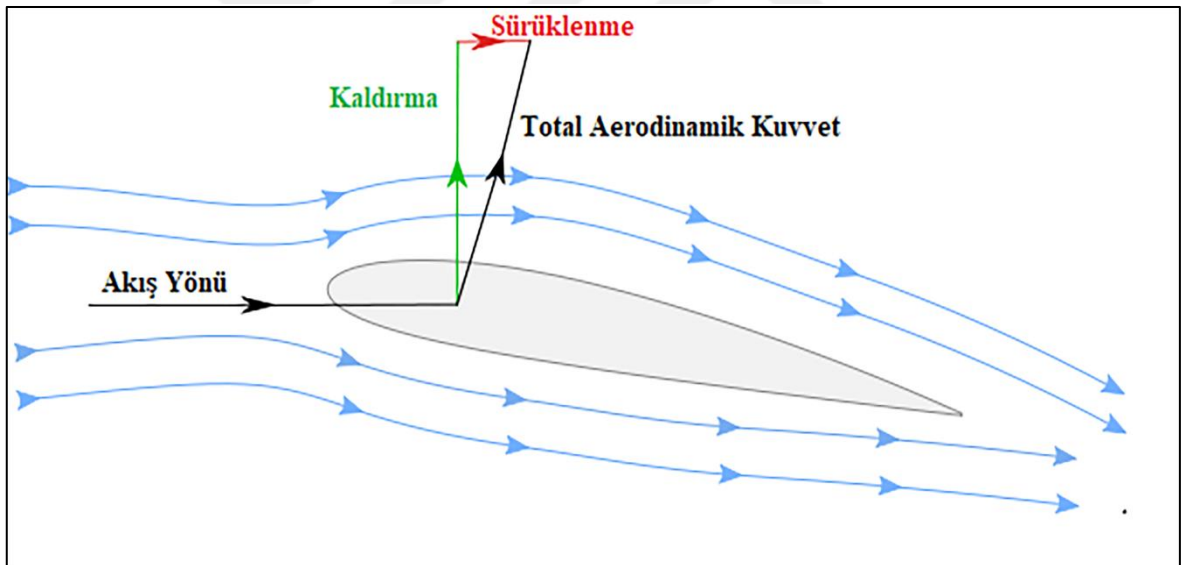
L : Kaldırma değeri

ρ : Hava yoğunluğu

V : Kanat profili üzerindeki hava hızı

A : Kanat alanı

C_L : Kaldırma katsayısı, istenen hücum açısındaki katsayıdır.



Şekil 1.2. Kanat üzerindeki aerodinamik kuvvet (Airfoil, <https://swyde.com/s/Airfoil>).

Sürüklenme, şekildeki gibi uçağın hareketine karşı çıkan ve onu yavaşlatan aerodinamik bir kuvvettir. Hava akımı molekülleri ile uçak yüzeyi arasındaki yüzey sürtünmesi olmak üzere benzer birçok neden sebebiyle oluşur. Kanat yüzeyine yakın olan havanın yavaşlamasına neden olur ve bu yavaşlayan hava tabakasına sınır tabakası denir. Sürüklenme kuvveti, motorun itiş gücü ile önlenmektedir. Bu nedenle sürüklenme kuvveti, motor gücü ve yakıt tüketim performansında etkilidir.

Yerçekimi, her zaman yerin merkezine doğru yönelen bir kuvvettir. Diğer tüm cisimlerde olduğu gibi uçağın da bir ağırlığı vardır ve uçağın uçabilmesi için onu yeryüzüne çeken kuvvete karşı (ağırlığı) daha fazla bir kuvvet olması gerekmektedir.

Aerodinamik şekil tasarımı, uçak tasarımının temel adımıdır, kanat profil optimizasyonu ise aerodinamik şekil tasarımının en önemli parçasıdır (Ma, Yu, Chen ve Xue, 2017). Kaldırma-sürüklenme oranı (L/D) uçak kanatları veya rüzgar tribünü gibi kanatların tasarımında en önemli hususlardan biridir (MacEachern ve Yıldız, 2018). Yüksek performanslı bilgisayar donanımı ve algoritmalarındaki gelişmeler uçak performansını değerlendirmek için kullanılan Hesaplamalı Akışkanlar Dinamiği (HAD) doğruluğunun artması için imkan sağlamıştır (Lyu, Kenway ve Martins, 2014).

Aerodinamik şekil optimizasyonu üç farklı şekilde ele alınmaktadır. Bunlar gradyan temelli yöntem, gradyan içermeyen yöntem ve vekil tabanlı yöntemlerdir.

Gradyan tabanlı yöntemde tasarım değişkenlerine göre maliyet fonksiyonlarının gradyanları kullanılır. Yöntemin dezavantajları her bir tasarım değişkenini değiştirmesi ve akış alanının yeniden hesaplaması gereken ve sonlu farklar yöntemiyle gradyanların doğrudan değerlendirilmesi ile ilgili büyük hesaplama maliyetidir. Bu hesaplama maliyeti ilk defa 1988’de Jameson tarafından birleşik yöntem yoluyla, gradyanın dolaylı olarak değerlendirilmesiyle önlenebilmiştir (Jameson, 1988). Bitişik tabanlı optimizasyon yöntemi oldukça etkilidir, optimuma 5-100 tasarım döngüsü içinde yaklaşılabılır (Jameson ve Martinelli, 2000). Gradyanlı Quasi Newton Metodu, İç Nokta Metodu, Eşlenik Gradyan Yöntemi ve Sıralı Kuadratik Programlama gibi gradyan tabanlı optimizasyon yöntemleri, birleşik bir yaklaşımla hesaplanan gradyan bilgisine dayalı optimal çözümü aramak için oldukça verimli yöntemlerdir (Jameson, 1988; Tang vd., 2014; Wang vd., 2018; Wang vd., 2020). Verimli olmasının yanında başlangıç değerlerine duyarlıdır ve yerel bir optimuma hapsolme eğilimindedirler (Han vd., 2020).

Gradyansız yöntemler yoğun ilgiye sahiptir, sebebi ise global optimizasyon kabiliyetine sahip olmasıdır. Bitişik tabanlı gradyan temelli optimizasyon yöntemi yerel bir optimizasyon yöntemine sahiptir. Global optimizasyon yöntemlerinden en çok bilinenleri genetik algoritmalar ve parçacık sürü algoritması gibi evrimsel algoritmalarlardır. Evrimsel algoritmalar en popüler global optimizasyon yöntemidir. Evrimsel algoritmaların global optimumun bulunmasında büyük öneme sahip olduğu kanıtlanmıştır ancak hesaplama maliyeti çok yüksek olması ve çok sayıda HAD simülasyonu gerektirmesi bu yöntemin kullanılmasını zorlaştırmaktadır (Han, Zhang, Liu ve Song, 2013). Benzetimli Tavlama

Yöntemi, Gradyansız Optimizasyon yöntemine verilebilecek örneklerden biridir. (Kirkpatrick, Gelatt ve Vecchi, 1983). Parçacık Sürü Optimizasyonu (Poli, Kennedy ve Blackwell, 2007), Karınca Kolonisi Algoritması (Dorigo, Maniezzo ve Colorni, 1996) vb. algoritmalar birçok optimizasyon ile tüm tasarım alanlarında optimal çözümü bulabilme kabiliyetine sahiptir. Bunun yanında Gradyansız optimizasyon yöntemi oldukça zaman alan bir yöntemdir. (Ronco vd., 2015; Wang vd., 2011). Optimizasyon probleminin çözümünde uygun zamanda optimal çözümü elde edebilmek için vekil tabanlı optimizasyon yöntemleri ortaya çıkmaktadır. Vekil tabanlı optimizasyon yöntemi; Kriging Yöntemi (Jeong vd., 2005), Destek Vektör Makinaları (Wang vd., 2011), Yapay Siniri Ağı (Mitchell vd., 1988) gibi algoritma yöntemleri ile bir çok pahalı yöntemin yerini alarak öne çıkan bir model haline gelmiştir.

Vekil model ile optimizasyon büyük ilgi görmektedir. Vekil model, değerlendirilmesi ucuz olan yaklaşık modeldir ve bu özelliği, değerlendirilmesi pahalı olan hedef ve sınırlama fonksiyonlarından ayırır. Bir vekil model, oluşturulduktan sonra optimizasyon süreci sırasında maliyet fonksiyonunu veya durum fonksiyonunu tahmin etmede fiziksel bir HAD modelinin yerini almak için kullanılabilir. Bu durumda vekil modelle yapılan bir tahmin daha verimli olduğu için HAD çözücü yerine vekil model kullanılmaktadır (Han, Abu-Zurayk, Görtz ve Ilic, 2015). Günümüzde birçok hesaplama maliyeti yüksek problemlerin çözümü için vekil destekli metasezgisel optimizasyon algoritmaları kullanılmıştır. Vekil model, birçok tekniğe dayalı olarak oluşturulmuştur. Polinom regresyon, radyal temel fonksiyon, destek vektör regresyonu ve yapay sinir ağı vb. modeller baz alınarak, aerodinamik optimizasyonla uçak kanadı tasarımı için çeşitli araştırmacılar tarafından metasezgisel optimizasyon çalışmaları uygulanmıştır (Mar Aye, Pholdee ve Bureerat, 2020).

Araştırmacılar kanat profil şekli optimizasyonunda çok seviyeli hiyerarşik bir Kriging modeli önermişlerdir. Modellerinde düşük doğruluk ve yüksek doğruluğun arasındaki korelasyonun doğrusal olduğu varsayılmaktadır (Han vd., 2020). Yapılan bir çalışmada Korelasyon parametrelerinin Gauss dağılımına uyduğuna dair bir yaklaşımla lineer otoregresif çoklu aslına uygunluk modelleme yöntemi lineer olmayan bir modellemeye genişletilmiştir (Perdikaris vd., 2017). Her ne kadar Gauss dağılımına dayalı çoklu aslına uygunluk yaklaşımları geliştirilmeye çalışılsa da yüksek boyutlu problemler için hala bazı sınırlamalara sahiptir. Bu problemi çözmek amacıyla çoklu aslına uygunluk

verilerle ters kısmi diferansiyel denklem problemlerinde araştırmacılar kompozit bir sinir ağı geliştirilmiştir (Meng ve Karniadakis, 2020).

Bu tez çalışmasında kanat profil aerodinamiğinde önemli bir rol alan kanat profiline ait şekilsel özellikleri tahmin etmek için FoilSim'den alınan verilerle vekil bir model ile optimizasyon yaklaşımı sunulmaktadır. Uçak aerodinamiği için en yaygın olarak kullanılan yöntem, uçağın menzilin ve dayanıklılığını belirlediği için kaldırma- sürüklenme oranıdır. FoilSim, web ortamında çalışmakta olan ve tasarımcılara uçak kanadına ait parametrelere (açı, kamburluk, kalınlık, hız ve kanat alanı) göre kaldırma ve sürüklenme miktarları hakkında bilgi veren bir web yazılımıdır. FoilSim tasarımcılar için çok kullanışlı bir simülasyon programı olmasına rağmen optimizasyon süreçlerinde daha etkin kullanılması adına aşağıda sıralanan adımlar uygulanmıştır.

- FoilSim web sayfasından 8000 adet veri seti JAVA awt robot ve MATLAB 2019a programı aracılığıyla toplanmıştır.
- Toplanan veriler Yapay sinir Ağları ile 5 katlamalı çapraz doğrulama yöntemi ile eğitilerek 486 test yapılmıştır ve vekil ağ oluşturulmuştur.
- Oluşturulan bu vekil ağ YAK algoritmasında model olarak kullanılmış ve tasarımcının girdiği kaldırma değeri için uygun açı kamburluk ve kalınlık değerlerinin üretilmesi sağlanmıştır.

Araştırmalarımız sonucunda elde ettiğimiz sonuca dayanarak literatürde FoilSim web sayfasından özgün veri seti oluşturup, YSA model oluşturarak optimizasyon algoritması olarak da YAK algoritması kullanan ve tasarımcılara istediği kaldırma değeri için uygun açı kamburluk ve kalınlık değerini sunan çalışmaya rastlanmamıştır.

Bu çalışmada üç girişli (açı, kamburluk, kalınlık) ve bir çıkışlı (C_L : kaldırma katsayısı), beş katlamalı çapraz doğrulama yöntemiyle YSA vekil modeli eğitilmiş ve bu modelle YAK yardımıyla optimizasyon problemi çözülmüştür. YAK algoritması yardımıyla kanat profili optimizasyonu gerçekleştirilip, istenen kaldırma değeri için en uygun açı, kamburluk ve kalınlık değerleri bulunmuştur. Tasarımcı tarafından belirlenen kaldırma kuvvetini sağlamak üzere FoilSim verileri ile optimizasyon algoritması kullanılarak, tasarım ve karar verme sürecine yardımcı olunup verimli hava araçları üretimine yardımcı olunmuştur.

2. LİTERATÜR ARAŞTIRMASI

Literatürde uçak kanat profil şekli optimizasyonu üzerine yapılan farklı çalışmalar bulunmaktadır. Bu çalışmalar ağırlıklı olarak kanat profili üzerine yoğunlaşmıştır.

Khurana Winarto ve Sinha (2009) sundukları Doğrudan Sayısal Optimizasyon (DSO) yaklaşımında şeklin fizibilitesini, tanımlanan hedefler ve kısıtlamalara dayanan yakınsamaya kadar yinelemeli analizle incelemişlerdir. Aerodinamik şekil optimizasyon süreci, öngörülen tasarım hedeflerini ele almak amacıyla oluşturulan akıllı modellerin geliştirilmesini gerektirmektedir. Çalışmalarında hesaplama yoğunluğunu azaltmak için DSO mimarisinin bileşenleri verimli bir arama optimizasyon modeli oluşturmak için tanımlanmış, doğrulanmış ve değiştirilmiştir. Aerodinamik olarak uygun olmayan şekillere sahip çözüm bölgeleri belirlenmiş ve hesaplama süresini azaltmak için arama sürecinden çıkarılmıştır. Tanımlanan arama uzayı üzerinde Gradyan Tabanlı yöntem yardımıyla tek noktalı kanat profili optimizasyonu incelenmiştir. Yerel optimal çözümleri azaltmak için bir çift mutasyon operatörü içeren PSO algoritması tanımlanmış ve doğrulanmıştır. Kanat profil şekil optimizasyonu için sürü algoritması, sürtünmede %16 azalma ile küresel bir çözüm oluşturarak gradyan yöntemlerinin sınırlı arama esnekliğini doğrulamıştır. Sürü algoritması şekil optimizasyonu için hesaplama açısından yoğundur, bir YSA modeli PARSEC çözüm uzayı ile kurulan aerodinamik kaldırma ve sürüklenme katsayıları arasındaki bir ilişki ile geliştirilmiş ve doğrulanmıştır. Yapılan ağ duyarlılık çalışması aerodinamik katsayıları kabul edilebilir doğrulukta oluşturmak için kaldırma için 30 nöron, sürüklenme için 20 nöron içeren çift katmanlı bir ağın gerekli olduğunu göstermiştir. Doğrudan PSO ve akış çözücü kombinasyonunun sonuçları doğrultusunda çıkış çözümünün aerodinamik performansı ile zaman tasarrufu sağlanmıştır.

Arias-Montano, Coello Coello ve Mezura-Montes (2012) kanat profil aerodinamik tasarımlarını optimize etmek için vekil tabanlı çok amaçlı evrimsel bir optimizasyon yaklaşımı üzerine yaptıkları çalışmalarında maliyetli ve çok amaçlı optimizasyon problemini çözerken, özelliklerini birleştirmek amacıyla paralel olarak çalışan çoklu vekil modelleri kullanmışlardır. Önerilen yaklaşımda beş adet çok amaçlı kanat profili, aerodinamik optimizasyon problemini çözmek için kullanılmıştır. Çok amaçlı bir evrimsel algoritmanın performansı, aynı yaklaşımı vekil modelleri kullanmadan ve vekil modelleri kullanarak karşılaştırılmıştır. Çalışmanın sonucunda vekil modelle yapılan çalışmanın pahalı amaç fonksiyonlarına karşı açık bir şekilde avantajlı olduğu görülmüştür.

Pehlivanoglu ve Yağız (2012) gerçekleştirdiği çalışmada kanat profil optimizasyonunda dört farklı durum için bir optimizasyon yöntemi önermiş ve test etmişlerdir. Bunlar, kıyaslama testi fonksiyonları, ses altı akışta çok elemanlı kanat profili optimizasyonu, transonik akışta aktif akış kontrol parametresi optimizasyonu ve transonik akışta kanat profilinin doğrudan şekil optimizasyonudur. Önerilen algoritma genetik algoritmaya dayanan global ve yerel bir vekil modellemeye dayalı bir aerodinamik tasarım tahmini üzerine odaklanmıştır. Düşük dereceli polinomlar kullanılarak küresel bir yanıt yüzeyi yaklaşımı modellenmiştir. Sinir ağları kullanılarak yerel bir yanıt yüzeyi yaklaşımı oluşturulmuştur. Hesaplama sürelerinde önemli ölçüde azalma olmuştur, önerilen yeni yöntemle HAD analizi süreçlerinde dikkate değer bir azalma fark edilmiştir. Bu azalmalar sırasıyla çok elemanlı kanat profili optimizasyonunda %50, aktif akış kontrol parametresi optimizasyonunda %70 den fazla, doğrudan şekil optimizasyon probleminde yaklaşık %50 olmuştur.

Kozziel ve Leifsson (2013) gerçekleştirmiş olduğu çalışmalarında transonik kanat profil tasarımı için vekil tabanlı bir optimizasyonu sunarken vekil modeli her tasarım yinelemesi sırasında düşük kaliteli model ve yüksek kaliteli model değerlendirmesinden elde edilen veriler kullanılarak oluşturmuşlardır. Düşük kaliteli model, yüksek kaliteli modelle aynı akış denklemlerine dayanır ancak daha kaba ağ çözünürlüğü ve rahat yakınsama kriterleri kullanır. Şekil koruyucu tepki tahmin tekniği, yüksek kaliteli model tepkisini (kanat profili basınç dağılımını) tahmin etmek için kullanılır. Bu tahmin süresince şekli koruyan tepki, tahmin tasarım değişkeni ayarlamalarından dolayı düşük kaliteli model tepkisinin gerçek değişikliklerini kullanır. Şekil koruyucu yanıt tahmin algoritması optimizasyon prosedürünün iyi yakınsama özelliklerini sağlayan güven bölgesi çerçevesine gömülüdür. Bu yöntem iki boyutlu viskoz olmayan transonik akışta kısıtlı kanat profili kaldırma maksimizasyonuna ve sürüklenme minimizasyonuna uygulanmıştır. Yapılan çalışmada önerilen algoritmaların sağlamlık ve ölçekleme özellikleri araştırılmıştır.

Lyu ve Martins (2014) çalışmalarında yüksek performanslı modellere dayalı aerodinamik kanat şekil optimizasyonunun hesaplama açısından yoğun çalışma gerektirmesinden dolayı genel hesaplama maliyetini azaltmak için farklı yaklaşımlar sunmuşlardır. Bu yaklaşımlar ADODG (Aerodynamic Design Optimization Discussion Group) tarafından tanımlanan kanat kalite testi kullanılarak test edilmiştir. Aerodinamik model, Spalart-Allmaras türbülans modeli ile Reynolds Ortalamalı Navier-Stokes denklemlerini çözmektedir. Gradyan tabanlı bir optimizasyon algoritması gerekli türevleri

hesaplayan birleşik bir yöntemle kullanılmıştır. Sürtünme katsayısı, kaldırma-yunuslama momenti ve geometrik kısıtlamalara bağlı olarak en aza indirilir. Çalışmanın ilk yaklaşımında akış çözümlerini ve gradyanları yaklaşık olarak tahmin etmek için Richardson ekstrapolasyonunu kullanılmıştır. İkinci yaklaşımda ise birçok ızgara boyutuyla çok düzeyli optimizasyon gerçekleştirilmiştir. Hesaplama maliyetinde %85'lik bir azalma sağlayan çok düzeyli yaklaşımın en etkili yöntem olduğu tespit edilmiştir.

Sun, Sun ve Wang (2015) gerçekleştirdikleri çalışmalarında tasarım verimliliğini arttırmayı hedefleyen, uygun şekilde eğitilmiş bir ağı doğrudan gerekli aerodinamik özelliklere uyan kanat profili-kanat oluşturulabilmesi için YSA ve kanat profili-kanat veritabanı yardımıyla kanat profili-kanat ters tasarım yöntemi önermişlerdir. Optimum kanat profil şeklinin sayısal olarak elde edilmeye çalışılması uçak ve turbo makine tasarımlarında büyük ilgi görmektedir. Başlangıç geometrisinin akış alanı hesaplamasına dayalı olarak kanat profili-kanat geometrisi verilerini değiştirme sürecini tekrarlamak olan geleneksel tasarım ve optimizasyon yöntemi, hesaplama açısından yoğun ve zaman açısından maliyetlidir. Bu sebeple sundukları alternatif çalışma bu sorunlar için çözüm niteliği taşımaktadır.

Segee, Schetz ve Kapania (2016) gerçekleştirdiği çalışmada Multidisipliner tasarım optimizasyonunun bir optimum aranırken birçok tasarımın değerlendirilmesini gerektirmesinden dolayı tasarımları değerlendirmek için yapılan hesaplamaların hızlı ve basit olması beklentisindedirler. Bu durum transonik rejimde aerodinamik hesaplamaları güçleştirmektedir. Multidisipliner tasarım optimizasyonu baz alındığında HAD analizlerini gerçekleştirmek hesaplama açısından çok pahalıdır. Bu sebeple yapılan çalışmada HAD'dan Multidisipliner tasarım optimizasyonu dışında seçilen bir kanat profil şeklinin bir dizi kalınlık-kordon oranları, serbest akış mach sayıları ve kaldırma katsayıları üzerindeki bir noktada iki boyutlu aerodinamik özelliklerini bulmak için faydalanılır. Bu noktalar Multidisipliner tasarım optimizasyonu hesaplamalı tasarım ortamının gerektirdiği iki boyutlu aerodinamik hesaplamalar için kullanılabilen vekil modeller oluşturmak için kullanılmaktadır. Şerit teorisi bu iki boyutlu sonuçları üç boyutlu kanatla ilişkilendirmek için kullanılmaktadır. Modeller kaldırma eğrisi, dalga sürüklemesi, basınç konum merkezi ve maksimum kaldırma katsayısı için geliştirilmiştir. Bu modeller kanat profili için orijinal HAD sonuçlarıyla uyumaktadır.

Yu, Lyu, Xu ve Martins (2018) tarafından sunulan çalışmada NASA Ortak Araştırma modeli kanat geometrisine dayalı hesaplamalı akışkanlar dinamiği tabanlı aerodinamik şekil

optimizasyon problemleri için gradyanlı ve gradyansız optimizasyon algoritmaları karşılaştırılmıştır. C_D , 720 şekil değişkeni ve 11 büküm değişkeni ile kaldırma, yunuslama momenti ve geometri kısıtlamalarına bağlı olarak en aza indirilmiştir. Hem optimumun doğruluğunu hem de hesaplama maliyetini karşılaştırarak altı gradyan tabanlı ve üç gradyansız algoritma kıyaslanmıştır. Optimize edicilerin çoğu benzer optimuma ulaşır ancak gradyan tabanlı yöntemler çok daha düşük hesaplama maliyetiyle daha doğru çözümlere yakınsamaktadır. Tasarım uzayının multimodalitesi ve düzgün olmaması gradyansız yöntemlerin kullanımı için ortak argümanlar olması sebebiyle bu sorunları rastgele oluşturulmuş bir dizi başlangıç geometrisinden başlayarak sıfır bükümlü, sabit kalınlık-koro oranına sahip NACA 0012 kanat profiline dayalı bir kanatla araştırılmıştır. Çalışmada gerçekleştirilen tüm optimizasyonlar, sürüklenmedeki farklılıkların %0,05 içinde olduğu şekiller ile basınç dağılımlarının çok benzer olduğu yerlerde aynı sonuçlara yakınsamaktadır.

Mukesh, Lingadurai ve Selvakumar (2018) tarafından kanat profil şekli optimizasyon tasarım hedeflerini değerlendirmek ve sağlamak için fazlaca gerçek zamanlı deneyler ve hesaplamalı simülasyonlara ihtiyaç duyulması problemine çözüm önerilmiştir. Bu süreçte vekil model olarak adlandırılan yaklaşım modelleri, hesaplama maliyetini ve zaman kaybını azaltması sebebiyle tercih edilmiştir. Çalışmada k-kat çapraz doğrulama yöntemi kullanılmıştır. Sıradan Kriging ve deney tasarım yaklaşımları, kanat profilleri ve uçak kanatları etrafındaki akışı çözümlmek için kullanılan panel ve viskoz algoritmalarını bir araya getirerek vekil modelleri oluşturmak için kullanılırlar. Çalışmada kanat profilleri için bir aerodinamik tasarım optimizasyon sürecini gerçekleştirmek için vekil modellerin uygun bir optimizasyon şemasıyla birleştirme yöntemi de çalışma için geliştirilebilecek bir konudur. Çalışmada kanat profil geometrisini (NACA2411) parametreleştirmek için parametrik kesit (PARSEC) yaklaşımı kullanılmıştır. PSO, bahsi geçen tasarım alanı içinde örnek oluşturmak için kullanılmıştır. HAD simülasyonları, vekil modeli oluşturmak için kullanılmıştır. HAD simülasyonları vekil modeli oluşturmak için oluşturulan örnek noktalarında gerçekleştirilmiştir. PARSEC; panel, PSO, Kriging ve k-kat çapraz doğrulama yaklaşımlarını uygulamak için oluşturulmuştur.

Tao ve Sun (2019) tarafından gerçekleştirilen başka bir optimizasyon çalışmasında Çoklu aslına uygun vekil tabanlı bir optimizasyon yöntemi kullanılmış ve mach sayısının belirsizliği altında uçak kanadı ve uçak kanat profiline uygulanmıştır. Düşük kaliteli model olarak Derin İnanç Ağı (DİA) ve ağı eğitmek için k-adımlı karşılaştırmalı sapma algoritması

kullanılmıştır. İyi eğitilmiş DİA modeli ve yüksek kalitedeki veriler sayesinde doğrusal regresyonlu çoklu aslına uygun vekil model kurulmuştur. Doğrulama sonuçları çoklu aslına uygun vekil modelin DİA modelinden daha doğru tahminler elde ettiğini ve bir tahmin modeli olarak oldukça güvenilir olduğunu göstermiştir. Çoklu aslına uygun vekil modelin geliştirilmiş bir PSO algoritmasıyla hem kanat hem de kanat profili için sağlam optimizasyon süreçlerinin her yinelenmesinde güncellenmiştir. Çoklu aslına uygun vekil model ve HAD sonuçları karşılaştırıldığında çoklu aslına uygun vekil modele ait tahminlerin yinelenme sayısı arttıkça HAD sonuçlarına yaklaşıma eğiliminde olduğu gözlemlenmiştir. Kanat profili ve kanadın güçlü optimizasyon sonuçları çoklu aslına uygun modelin bir tahmin modeli olarak performansının iyi olduğunu ve optimizasyon verimliliğinin açıkça geliştirdiğini göstermektedir.

Li, Cai ve Qu (2019) yaptıkları çalışmalarında kanat profil optimizasyonunda vekil tabanlı optimizasyon yöntemi girdi boyutuyla iyi ölçeklenemediği için yüksek boyutlu girdilerde sorun yaşaması problemine çözüm getirmek amacıyla girdi boyutunu azaltmak için aktif alt uzay yöntemini benimsemişlerdir. Bu işlem bir yandan hesaplama maliyetini büyük oranda azaltırken diğer yandan daha uygun alt uzaylara yol açmaktadır. Alt uzayın boyutunu tanımlamak için %90-95 enerji kapsama kriteri kullanılmıştır. Bu kriter ile aktif alt uzayda vekil tabanlı kanat profili optimizasyonu hem verimli hem de etkilidir. Bu optimizasyon yaklaşımı 220 şekil değişkenli ONERO M6 kanat optimizasyonu ile doğrulanmıştır. Orjinal vekil tabanlı optimizasyon ile karşılaştırıldığında yeni yaklaşım hesaplama süresini %70 oranında azaltmış ve daha az sürüklenme ile daha pratik tasarım elde edilmiştir.

Han vd. (2020) tarafından yapılan çalışmada kanat profil optimizasyonu için yeni bir değişken aslına uygunluk optimizasyon yöntemi önerilmiştir. Çalışmanın temel bileşenleri, verilerini keyfi doğruluk verileriyle birleştirebilen vekil bir model olarak adlandırdıkları çok düzeyli hiyerarşik kringing teorisi ve algoritmasıdır. Deney tasarımı tekniği ile her bir aslına uygunluk düzeyi için örnekleme seçilmiştir. Daha sonra HAD simülasyonları yürütülmüş, amaç ve kısıtlama fonksiyonları çok düzeyli hiyerarşik kringing vekil modellerini oluşturmada kullanılmıştır. Ardından yeni örnekler seçilmiş ve vekil modeller global optimum bulunana kadar tekrar tekrar güncellenmiştir. Önerilen yöntem, testlerle doğrulanmış ve NACA0012 kanat profili ve ONERA M6 kanadının aerodinamik şekil optimizasyonuna uygulanmıştır. Sonuçlar, önerilen yöntemin optimizasyon verimliliğini önemli ölçüde artırabildiğini doğrulamıştır.

Zhang, Xie, Ji, Zhu ve Zheng (2021) tarafında yapılan bir diğer çalışmada da farklı aslına uygunluk bilgilerini harmanlayarak, önceden herhangi bir varsayım olmaksızın lineer veya lineer olmayan korelasyonlarını uyarlayarak bir uçağı konfigürasyon parametreleri ile aerodinamik performansını ilişkilendiren yüksek doğruluklu, çoklu aslına uygun bir temsili model oluşturulmuştur. Her optimizasyon yinelemesinde vekil model doğruluğunu iyileştirmek amacıyla yüksek kaliteli veri tabanına mevcut en uygun vekil model çözümünü ekleyerek yüksek kaliteli stratejiler uygulanmıştır. Sonrasında önerilen çoklu aslına uygun optimizasyon model, 10 tasarım değişkeni ile parametrelenen bir RAE2822 kanat profilinin ve 30 tasarım değişkeni ile parametrelenen bir DLR-F4 kanat gövdesi konfigürasyonunun yüksek boyutlu aerodinamik şekil optimizasyonuna uygulanmıştır. Optimizasyon sonuçlarıyla, önerilen çoklu aslına uygun optimizasyon modelinin optimizasyon verimliliğini önemli ölçüde etkileyebileceğini ve tekli aslına uygun optimizasyon yönteminden daha iyi performans gösterebildiği gözlemlenmiştir.

Literatürde kanat profil optimizasyon süreci dahil farklı alanlarda FoilSim arayüzünden faydalanılarak yapılan çalışmalar aşağıda sunulmuştur.

Yang (1995) gerçekleştirdiği çalışmada aerodinamik alanında çalışan öğrencilerin yalnızca akışı simüle etmek için değil, görselleştirme konusunda da faydalanabileceği bir çalışma önermiştir. Aerodinamik alanında yer alan araçları kullanmak için karmaşık hesaplamalı programlama dilinde oldukça bilgili olmak gerekmektedir. Çoğu havacılık öğrencisi için aerodinamiğin karmaşık dünyasını ve bir dizi matematiksel denklemleri anlamak oldukça zordur. Bunun yanında HAD dünyasında tipik olarak simülasyonun doğruluğu, çözümünü elde etme hızı için feda edilmektedir. Aksi olarak hızın doğruluk için feda edilmesi de söz konusu olabilmektedir. Tam olarak bu noktada HAD'da yer alan karmaşık matematiği basitleştirme adına Düşük Hızlı Sanal Rüzgar Tüneli simülasyonu oluşturulmuştur. Bu program kullanıcıdan gerekli bilgileri alarak kaldırma, sürüklenme ve yunuslama momenti gibi aerodinamik özelliklerin tahminini yapabilmektedir. Bu simülasyon programının kanat profili ve akış analizinin görselleştirmesini sağlayan web tabanlı modüllerinde FoilSim ve JavaFoil'den faydalanılmıştır. Yapılan bu çalışmanın sürekli gelişen havacılık endüstrisinde öğrencilere katkı sağlayacağı öngörülmüştür.

Çetinsoy vd. (2008) tarafından gerçekleştirilen bir diğer çalışmada da dört-döner rotoru olan helikopter benzeri dikine kalkış ve iniş yapma yeteneğine sahip hem de uçak gibi uzun menzil yatay uçuş kabiliyetine sahip bir insansız otonom hava aracının mekanik ve aerodinamik tasarımı sunulmuştur. Hava aracının kanat profil tasarımında hafif ve mekanik

olarak basit olması açısından kanatlar tek parça olarak tasarlanmıştır. Kanatların hücum açıları, kontrol değişkeni olarak planlanmış ve kaldırma kuvvetinin aracın ağırlığı taşıyabilecek şekilde olması hedeflenmiştir. Farklı hızlarda kanatlar tarafından taşınmanın sağlanabilmesi için lazım olan veter uzunluğu, kanat hücum açısı ve kanat profilinin belirlenmesi için NASA FoilSim II programından benzetimler yapılmıştır. Neticede 25 cm veter, %12,5 kalınlık ve %4 kamburluk değerlerinde kanat profilinin havada tutunabileceği ve kaldırma kuvvetini oluşturabileceği öngörülmüştür. Donanım seçiminde yapılan deneyler ve sonuçlarının yanı sıra aerodinamik ve mekanik tasarım süreci içinde yapılan analizlerin de sonuçları sunulmuştur. Sonuçlar, tasarlanan insansız hava aracının başarılı bir şekilde dikey uçuş modunda yaklaşık 25 dakika, yatay uçuş modunda ise iki saat havada kaldığını göstermektedir.

Post, Seetharaman, ve Abimannan (2010) tarafından sunulan başka bir çalışmada akışkanlar mekaniği derslerindeki öğrenciler için ortaokul düzeyinde bir dönem boyu süren sınıf projesi açıklanmaktadır. Çalışmanın amaçları, öğrencileri mühendislik tasarımı, proje yönetimi ile tanıştırmak ve mühendislik grafikleri, sayısal yöntemler, ölçümler ile üretim süreçlerindeki diğer derslerden malzemeleri tek bir projede birleştirmektir. Çalışma, hesaplama araçları kullanılarak kanat profili tasarımına odaklanmaktadır. Ana vurgu, deneysel olarak ölçülen değerlerle hesaplama yöntemlerinden elde edilen sonuçların doğrulanması üzerinedir. Çalışmada öğrenciler, bir model köpük planörü elde edebilme adına kanat yapmak için oluşturdukları kanat profil şeklini kullanacaklardır. Çalışmanın ilk aşamasında öğrencilere, ücretsiz olarak temin edebildikleri FoilSim, Xfoil veya JavaFoil gibi kaldırma ve sürüklenme katsayılarının analizinde kullanılan hesaplama araçları kullanılarak belirli sınırlandırmalarla bir kanat profili tasarlamasını önermektedir. Öğrencilerin kullanımı için projeyi daha basit tutmak adına NACA 4 kanat profili kullanılması önerilmiştir. Hesaplama sonuçlarına dayalı en yüksek kaldırma-sürüklenme (L/D) oranını veren kanat profil şekli bulunduktan sonra hesaplama sonuçlarını doğrulamak için bir kanat profil şekli, rüzgar tüneline test etmesi beklenmektedir.

Jameel vd. (2015) tarafından sunulan bir diğer çalışmada da Corvus Splendens'in doğal modeline dayalı olarak biyo-ilham kanatlı İHA (İnsansız Hava Aracı)'nın kinematik bir modeli önerilmiştir. Bu model keşif görevlerinde kullanılmak üzere tasarlanmıştır. Model, dört çubuklu mekanizma teorisine dayanan tek serbestlik dereceli çırpma mekanizması sunmaktadır. Tüm kanat hareketinin analiz edildiği mekanizma tasarım yazılımı SAM 6.1'de kanat düzeneğinin sanal bir modeli tasarlanmıştır. Tüm kuş tasarımının

görselleştirildiği katı bir modelleme yazılımında tüm kuşun CAD modeli de geliştirilmiştir. Son olarak, kanat profili seçimini doğrulamak için çeşitli analizler yapılmıştır. Analiz paketi NASA FoilSim, önerilen modelin üreteceği kaldırma kuvvetlerini sağlamak için bazı ilk hesaplamaları yapmak için kullanılmıştır, böylece tasarımın aerodinamik verimliliği hakkında gerçekçi bir fikir elde edilebilmiştir. Seçilen kanat profilinin analizi Ansys 14.0 kullanılarak yapılmıştır.

Greyber Rand ve Stockburger (2016) tarafından gerçekleştirilen bir çalışmada da 3B yazdırılmış kanat profilinin deneysel verileriyle farklı simülasyonlar karşılaştırılmıştır. Çalışma, düz tabanlı bir kanat profili üzerindeki dört farklı veri kaynağını karşılaştırmaktadır. Rüzgar tüneli testlerinden elde edilen veriler, karşılaştırma için temel teşkil etmektedir. Autodesk Computation Fluid Dynamics, XFOIL ve FoilSim III'ü karşılaştırarak, hangi simülasyonun kaldırma ve sürüklenme katsayılarını tahmin etmede en doğru olduğunu bulmak hedeflenmektedir. Her simülasyonun güçlü ve zayıf yönleri olduğu bakılan simülasyonların hiçbirinin her konuda diğerlerinden daha iyi olmadığını anlaşılmıştır. Spesifik olarak Autodesk CFD, doğru artış verilerine sahiptir ancak sürüklenme hesaplanırken sorunlar yaşamaktadır. XFOIL sürüklemeye çok daha iyidir ve FoilSim kaldırma ve hücum açısı için sürüklenme eğrilerinin şeklini bulmada daha iyidir. Bu çalışmadan elde edilen sonuçlar gelecek çalışmalarda kullanılma potansiyeli sahiptir.

Sahwee, Mahmood, Rahman ve Sahari (2017) gerçekleştirdikleri çalışmada, arızalı asansör aktüatörünün barınma tekniği sunulmaktadır. Bu teknik, geçici kontrol yeniden tahsisi olarak bir bekleme kontrol yüzeyi kullanır. Deneysel doğrulama için rüzgar tüneli ölçüm tesisi kurulmuş ve FoilSim yazılımı ile karşılaştırılmıştır. Yatay stabilizatör olarak kullanılan düz plaka kanat profili, sayısal model kullanılarak simüle edilmiş ve rüzgar tüneli testi kullanılarak doğrulanmıştır. Ardından, hatalı asansör aktüatörünün kurtarılması için stabilatör olarak kullanılmak üzere düz bir kanat profili tasarlanmıştır. Sonuçlar, bir kontrol yüzeyinden diğerine aktarım sırasında farklı sapma açısına ihtiyaç olduğunu göstermektedir. Analizlerle önerilen yöntemin kontrol yüzeyi geçişi sırasında yunuslama kararlılığını etkilemeden uygulanabildiğini göstermektedir. Alternatif kontrol yüzeyi yerleştirme tekniği, arızalı bir yerleşik aktüatör durumunda daha yüksek güvenilirlik küçük insansız hava araçları için umut verici olduğunu göstermektedir.

Shen (2017) tarafından sunulan başka bir çalışmada, sağlam parametre tasarım kriterlerini verimli bir şekilde optimize etmek için stokastik yaklaşım (RPD-SA) yoluyla yeni bir çerçeve çalışması sağlam parametre tasarımı önerilmektedir. Çalışma genel sağlam

parametre tasarım problemlerine uygulanabilmekle birlikte bilgisayar deneyleri konusunda özellikle güçlüdür. RPD-SA'da geleneksel ürün dizisi tasarımları ve boşluk doldurma tasarımlarının performansını Gauss süreciyle karşılaştırmak için bir kanat profil örneği kullanılmıştır. Sonuçlar, RPD-SA'nın etkinlik, verimlilik ve güvenilirlik açısından tercih edilebilir bir performansa sahip olduğunu göstermektedir. Önerilen RPD-SA prosedürü için bilgisayar simülatörü olarak FoilSim kullanılmıştır.

KT, R, Sharma, Babu ve Khanapure (2020) tarafından sunulan başka bir çalışma büyük stabilite sağlayan ve orta hızda yeterli kaldırma sağlayarak Clark Y tipi kanat profili kullanan bir İHA'yı açıklamaktadır. Tasarlanan İHA, üç kontrol yüzeyli ve üç kanallı bir İHA'dır. İHA'nın üzerinde durulan özelliği, manuel olarak ve yerleşik bir otopilot ile uçabilmesinin ikili özellikte oluşudur. Daha açıklayıcı olması adına literatür araştırmasına aşağıda yer alan Tablo 2.1 de yer verilmiştir.

Tablo 2.1. Uçak kanat profil optimizasyonu üzerine literatür araştırması.

Çalışma	Amaç	Yöntem	Sonuç
Airfoil optimisation by swarm algorithm with mutation and Artificial Neural Networks (Khurana, Winarto ve Sinha, 2009).	Doğrudan Sayısal Optimizasyon (DNO) yaklaşımı, bir şeklin fizibilitesini, tanımlanmış hedefler ve kısıtlamalar temelinde yakınsamaya kadar yinelemede analiz edilerek incelenmektedir.	DSO (Doğrudan Sayısal Optimizasyon) Gradyan tabanlı yöntem PSO YSA.	Kanat profili şekli optimizasyonu için sürü algoritması, sürtünmede %16 azalma ile küresel bir çözüm oluşturarak gradyan yöntemlerinin sınırlı arama esnekliğini doğrulamıştır. Doğrudan PSO ve Akış çözücü kombinasyonu ile çıkış çözümünün aerodinamik performansı ile zaman tasarrufu sağlanmıştır.
Multi-objective airfoil shape optimization using a multiple-surrogate approach (Arias-Montaña, Coello Coello ve Mezura-Montes, 2012).	Vekil tabanlı çok amaçlı evrimsel bir optimizasyon yaklaşımı önerilmiştir. Paralel olarak çalışan çoklu vekil modelleri kullanılmışlardır.	Vekil tabanlı çok amaçlı evrimsel yaklaşım modeli.	Çalışmanın sonucunda vekil modellerle yapılan çalışmanın pahalı amaç fonksiyonlarına karşı açık bir şekilde avantajlı olduğu görülmüştür.

Aerodynamic Design Prediction Using Surrogate-Based Modeling in Genetic Algorithm Architecture (Pehlivanoglu ve Yağız ,2012).	Kanat profil optimizasyonunda dört farklı durum için optimizasyon yöntemi önermişlerdir. Kıyaslama testi fonksiyonları. Ses altı akışta çok elemanlı kanat profili optimizasyonu. Transonik akışta aktif akış kontrol parametresi optimizasyonu. Transonik akışta kanat profilinin doğrudan şekil optimizasyonu.	Genetik Algoritma Vekil modelleme.	Önerilen yeni yöntemle HAD analizi süreçlerinde dikkate değer bir azalma fark edilmiştir. Sırasıyla çok elemanlı kanat profili optimizasyonunda %50, aktif akış kontrol parametresi optimizasyonunda %70 den fazla, doğrudan şekil optimizasyon probleminde yaklaşık %50 kadar olmuştur.
Surrogate-Based Aerodynamic Shape Optimization by Variable-Resolution Models (Koziel ve Leifsson, 2013).	Transonik kanat profil tasarımı için vekil tabanlı bir optimizasyonu sunarken vekil modeli her tasarım yinelemesi sırasında düşük kaliteli model ve yüksek kaliteli model değerlendirmesinden elde edilen veriler kullanılarak oluşturmuşlardır.	Şekil koruyan yanıt tahmin tekniğine dayalı vekil model.	Önerilen yöntem iki boyutlu viskoz olmayan transonik akışta kısıtlı kanat profili kaldırma maksimizasyonuna ve sürüklenme minimizasyonuna uygulanmıştır. Önerilen algoritmaların sağlamlık ve ölçekleme özellikleri araştırılmıştır.
Strategies for Solving High-Fidelity Aerodynamic Shape Optimization Problems (Lyu ve Martins, 2014).	Yüksek performanslı modellere dayalı aerodinamik kanat şekil optimizasyonunun hesaplama açısından yoğun çalışma gerektirmesinden dolayı genel hesaplama maliyetini azaltmak için farklı yaklaşımlar sunmuşlardır. Yaklaşımlar ADODG tarafından test edilmiştir.	Gradyan tabanlı optimizasyon algoritması.	Hesaplama maliyetinde %85'lik bir azalma sağlayan çok düzeyli yaklaşımın en etkili yöntem olduğu tespit edilmiştir.
Artificial Neural Network Based Inverse Design: Airfoils and Wings (Sun, Sun ve Wang 2015).	Tasarım verimliliğini arttırmayı hedefleyen, uygun şekilde eğitilmiş bir ağın doğrudan gerekli aerodinamik özelliklere uyan kanat profili-kanat ters tasarım yöntemi önemişlerdir.	YSA modeli Kanat profili/Kanat ters tasarım yöntemi.	Sunulan bu çalışma ile yoğun hesaplama ve zaman maliyetine çözüm niteliği taşımaktadır.

Transonic Aerodynamics Analysis for Multidisciplinary Design Optimization Applications (Segee, Schetz ve Kapania 2016).	Multidisipliner tasarım optimizasyonunun çok boyutluluğuna çözüm önerilmiştir.	YSA tabanlı vekil model Şerit teorisi.	Kaldırma eğrisi, dalga sürüklemesi, basınç konum merkezi ve maksimum kaldırma katsayısı için modeller geliştirilmiştir. Bu modeller kanat profili için orijinal HAD sonuçlarıyla uyusmaktadır.
On the Influence of Optimization Algorithm and Initial Design on Wing Aerodynamic Shape Optimization (Yu, Lyu,Xu ve Martins, 2018).	NASA Ortak Araştırma modeli kanat geometrisine dayalı hesaplamalı akışkanlar dinamiği tabanlı aerodinamik şekil optimizasyon problemleri için gradyanlı ve gradyansız optimizasyon algoritmaları karşılaştırılmıştır.	Gradyanlı ve Gradyansız Optimizasyon algoritmaları.	Çalışmada gerçekleştirilen tüm optimizasyonlar, sürüklenmedeki farklılıkların %0,05 içinde olduğu şekiller ile basınç dağılımlarının çok benzer olduğu yerlerde aynı sonuçlara yakınsamaktadır.
Airfoil Shape Optimization Based on Surrogate Model (Mukesh, Lingadurai ve Selvakumar 2018).	Kanat profil şekli optimizasyon tasarım hedeflerini değerlendirmek ve sağlamak için fazlaca gerçek zamanlı deneyler ve hesaplamalı simülasyonlara ihtiyaç duyulması problemiye çözüm önerilmiştir.	Kriging ve deney tasarım yöntemleri vekil modeli oluşturmada kullanılmıştır. K-kat çapraz doğrulama PSO HAD Simülasyonları.	HAD simülasyonları vekil modeli oluşturmak için oluşturulan örnek noktalarında gerçekleştirilmiştir. PARSEC; panel, PSO, Kriging ve k-kat çapraz doğrulama yaklaşımlarını uygulamak için oluşturulmuştur.
Application of Deep Learning Based Multi-Fidelity Surrogate Model to Robust Aerodynamic Design Optimization (Tao ve Sun, 2019).	Çoklu aslına uygun vekil tabanlı bir optimizasyon yöntemi kullanılmış ve mach sayısının belirsizliği altında uçak kanadı ve uçak kanat profiline uygulanmıştır.	Derin İnanç Ağı (DİA) K adımlı karşılaştırmalı sapma algoritması PSO.	Çoklu aslına uygun vekil model ve HAD sonuçları karşılaştırıldığında çoklu aslına uygun vekil modele ait tahminlerin yineleme sayısı artıkça HAD sonuçlarına yaklaşma eğiliminde olduğu gözlemlenmiştir.

Surrogate-Based Aerodynamic Shape Optimization with the active subspace method (Li, Cai ve Qu, 2019).	Kanat profil optimizasyonunda vekil tabanlı optimizasyon yönteminin girdi boyutuyla iyi ölçeklenemediği için yüksek boyutlu girdilerde sorun yaşaması problemine çözüm getirmek amaçlanmıştır.	Kriging yöntemi ile vekil model.	Bu optimizasyon yaklaşımı 220 şekil değişkenli ONERO M6 kanat optimizasyonu ile doğrulanmıştır. Orjinal vekil tabanlı optimizasyon ile karşılaştırıldığında yeni yaklaşım hesaplama süresini %70 oranında azaltmış ve daha az sürüklenme ile daha pratik tasarım elde edilmiştir.
Efficient Aerodynamic Shape Optimization Using Variable-Fidelity Surrogate Models And Multilevel Computational Grids (Han vd.,2020).	Yapılan çalışmada kanat profil optimizasyonu için yeni bir değişken aslına uygunluk optimizasyon yöntemi önerilmiştir.	Çalışmanın temel bileşenleri, verilerini keyfi doğruluk verileriyle birleştirebilen vekil bir model olarak adlandırdıkları çok düzeyli Hiyerarşik Kriging teorisi ve algoritmasıdır. Çalışmada HAD simülasyonlarından faydalanılmıştır.	Önerilen yöntem, testlerle doğrulanmış ve NACA0012 kanat profili ve ONERA M6 kanadının aerodinamik şekil optimizasyonuna uygulanmıştır. Sonuçlar, önerilen yöntemin optimizasyon verimliliğini önemli ölçüde artırdığını doğrulamıştır.

Tablo 2.2. FoilSim’den yararlanılarak yapılan çalışmalara dair literatür araştırması.

Çalışma	Amaç	Yöntem	Sonuç
Low Speed Virtual Wind Tunnel Simulation For Educational Studies In Introducing Computational Fluid Dynamics And Flow Visualization (Yang,1995).	Aerodinamik alanında çalışan öğrencilerin yalnızca akışı simüle etmek için değil, görselleştirme konusunda da faydalanabileceği bir çalışma yapmak amaçlanmıştır.	Düşük Hızlı Sanal Rüzgar Tüneli simülasyonu Bu program kullanıcıdan gerekli bilgileri alarak kaldırma, sürüklenme ve yunuslama momenti gibi aerodinamik özelliklerin tahminini yapabilmektedir. FoilSim.	Bu simülasyon programının kanat profili ve akış analizinin görselleştirmesini sağlayan web tabanlı modüllerinde FoilSim ve Javafoil’den faydalanılmıştır. Yapılan bu çalışmanın sürekli gelişen havacılık endüstrisinde öğrencilere katkı sağlayacağı öngörülmüştür.
Yeni Bir İnsansız Hava Aracının (SUAVi) Mekanik ve Aerodinamik Tasarımı (Çetinsoy vd., 2008).	Dört-döner rotoru olan helikopter benzeri dikine kalkış ve iniş yapma yeteneğine sahip hem de uçak gibi uzun menzile yatay uçuş kabiliyetine sahip bir insansız otonom hava aracının mekanik ve aerodinamik tasarımı çalışması amaçlanmıştır.	Farklı hızlarda kanatlar tarafından taşınmanın sağlanabilmesi için lazım olan veter uzunluğu, kanat hücum açısı, ve kanat profilinin belirlenmesi için NASA FoilSim II programından benzetimler yapılmıştır. Neticede 25 cm veter, %12,5 kalınlık %4 kamburluk değerlerinde kanat profilinin havada tutunabileceği kaldırma kuvvetini oluşturabileceği öngörülmüştür.	Sonuçlar, tasarlanan insansız hava aracının başarılı bir şekilde dikey uçuş modunda yaklaşık 25 dakika, yatay uçuş modunda ise iki saat havada kaldığını göstermektedir.
A Design-Build-Test-Fly Project Involving Modeling, Manufacturing, and Testing (Post, Seetharaman ve Abimannan, 2010).	Çalışmanın amaçları, öğrencileri mühendislik tasarımı, proje yönetimi ile tanıştırmak ve mühendislik grafikleri, sayısal yöntemler, ölçümler ile üretim süreçlerindeki diğer derslerden malzemeleri tek bir projede birleştirmektir. Çalışma, hesaplama	FoilSim, Xfoil, JavaFoil.	Hesaplama sonuçlarına dayalı en yüksek kaldırma-sürüklenme (L/D) oranını veren kanat profil şekli bulunduktan sonra hesaplama sonuçlarını doğrulamak için bir kanat profil şeklinin rüzgar tüneline test etmesi beklenmektedir.

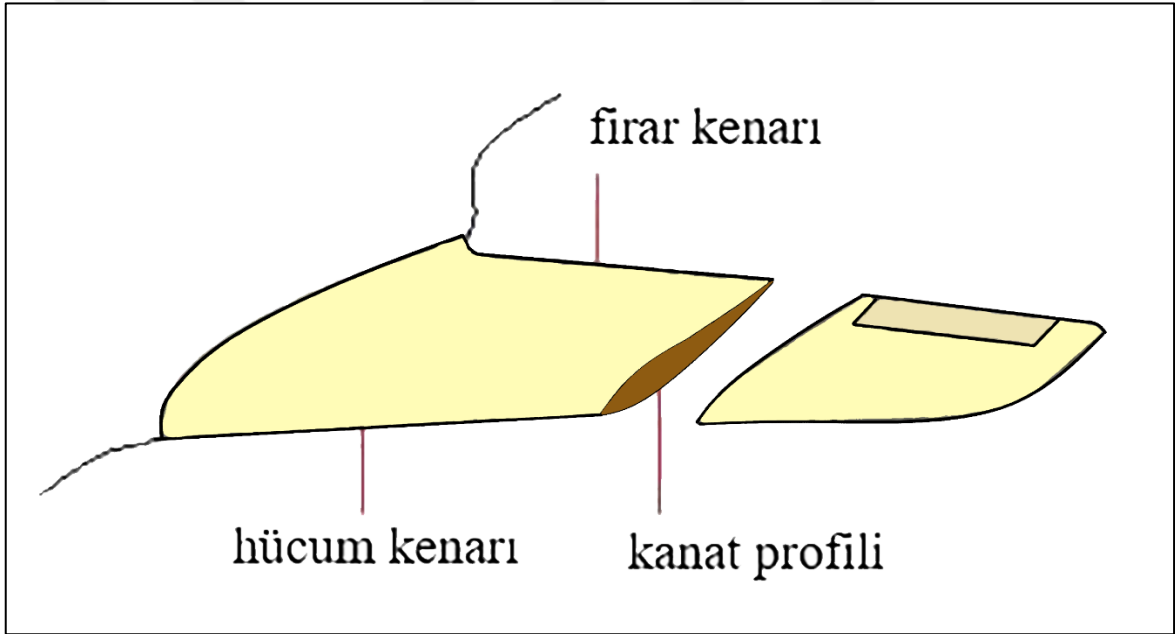
	araçları kullanılarak kanat profili tasarımına odaklanmaktadır ve ana vurgu, deneysel olarak ölçülen değerlerle hesaplama yöntemlerinden elde edilen sonuçların doğrulanması üzerinedir.		
A Bio Inspired UAV Model Based on Corvus Splendens (Jameel vd., 2015).	Corvus Splendens'in doğal modeline dayalı olarak biyo-ilham kanatlı İHA (İnsansız Hava Aracı)'nın kinematik bir modeli önerilmiştir.	SAM 6.1 FoilSim Ansys 14.0.	Bu model keşif görevlerinde kullanılmak üzere tasarlanmıştır.
Airfoil Simulation Comparison (Greyber, Rand ve Stockburger 2016).	3B yazdırılmış kanat profilinin deneysel verileriyle farklı simülasyonlar karşılaştırılmıştır.	Autodesk Computation Fluid Dynamics, XFOIL ve FoilSim II.	Autodesk CFD, doğru artış verilerine sahiptir ancak sürüklenme hesaplanırken sorunlar yaşamaktadır. XFOIL sürüklenmede çok daha iyidir ve FoilSim kaldırma ve hücum açısı için sürüklenme eğrilerinin şeklini bulmada daha iyidir.
Wind Tunnel Evaluation for Control Transition from Elevator to Stabilator of Small UAV (Sahwee, Mahmood,Rahman ve Sahari,2017)	Çalışmada arızalı asansör aktüatörünün barınma tekniği sunulmaktadır. Bu teknik, geçici kontrol yeniden tahsisi olarak bir bekleme kontrol yüzeyi kullanır.	Rüzgar tüneli ölçüm tesisi FoilSim	Sonuçlar, bir kontrol yüzeyinden diğerine aktarım sırasında farklı sapma açısına ihtiyaç olduğunu göstermektedir. Analizlerle önerilen yöntemin kontrol yüzeyi geçişi sırasında yunuslama kararlılığını etkilemeden uygulanabildiğini göstermektedir.
Robust Parameter Designs in Computer Experiments Using Stochastic Approximation (Shen,2017).	Sağlam parametre tasarım kriterlerini verimli bir şekilde optimize etmek için stokastik yaklaşım (RPD-SA) yoluyla yeni bir çerçeve çalışması sağlam parametre tasarımı önerilmektedir.	FoilSim.	Sonuçlar, RPD-SA'nın etkinlik, verimlilik ve güvenilirlik açısından tercih edilebilir bir performansa sahip olduğunu göstermektedir.

Design and Fabrication of a Fixed Wing UAV (KT, R, Sharma, Babu ve Khanapure, 2020).	Büyük stabilite sağlayan ve orta hızda yeterli kaldırma sağlayarak Clark Y tipi kanat profili kullanan bir İHA önerilmektedir.	CATIA FoilSim.	İHA için kanat profili FoilSim ortamında belirlenen hücum açısı, kamburluk, kalınlık, hız, hava yoğunluğu ve en-boy oranında simüle edilmiştir. Tasarım süreci CATIA ortamında gerçekleştirilmiştir. İlk olarak kanat sonrasında da gövde tasarlanmıştır.
--	--	----------------	---

3. MATERİYAL VE YÖNTEM

3.1. Uçak Aerodinamiği ve Kanat Profili

Bir uçak için temel olarak en basit uçuş hali sabit irtifada düzgün simetrik uçuş halidir. Bu haldeki uçuşun sürdürülebilmesi için uçağın ağırlığına eşit bir taşıma kuvveti gerekmektedir. Aerodinamik olarak bir uçağın havada kalabilmesi için ancak belirli bir yüksek hızda uçuşması gerekmektedir. Havada yüksek hızda bulunan uçağa etki eden bir sürüklenme kuvveti vardır. Uçağın hareketinin devam ettirilebilmesi için sürüklenme kuvvetine ters yönde çekme kuvveti ile dengelenmesi gerekir. Uçağın bütün elemanlarının uçağın taşınmasında etkisi söz konusudur. Uçağın uçuşu için gerek duyduğu taşıma, büyük oranda kanatlar aracılığıyla gerçekleşir (Yükselen, 2006b).

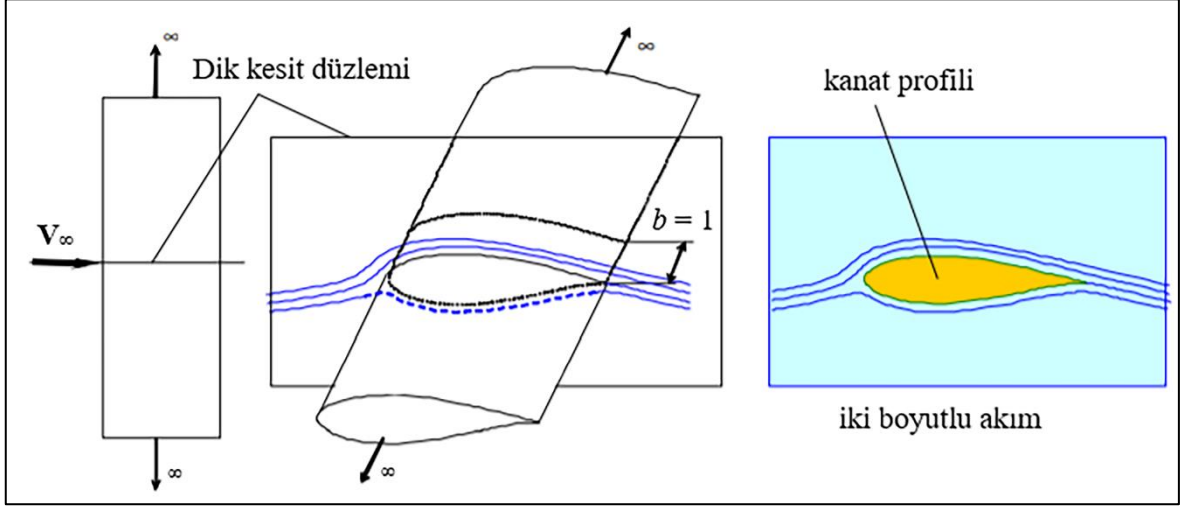


Şekil 3.1. Kanadın yandan kesit görüntüsü: İki boyutlu kanat profili (Pehlivanoğlu, 2013).

Kanat profilinin şekli doğadan ilham alınarak oluşturulmuştur. Uçan canlıların kanat geometrileri veya daha sade haliyle yağmur damlalarının biçimi de esinlenilen konular arasındadır. Araştırmacıların bu gibi örnekleri doğadan esinlenme sebepleri daha az sürüklenme ve az sürüklenmenin etkisiyle de daha fazla taşıma kuvveti oluşturma yeteneğidir (Pehlivanoğlu, 2013).

Kanat profili, kanadın gövde doğrultusunda kesilmesiyle elde edilen yanal kesit alanıdır. Kanat profillerindeki amaç akışkan bir ortamda hareket eden araca en uygun L/D (Kaldırma/Sürüklenme) oranını vermektedir. Şekilleri kıvrımlı veya düz olan genellikle

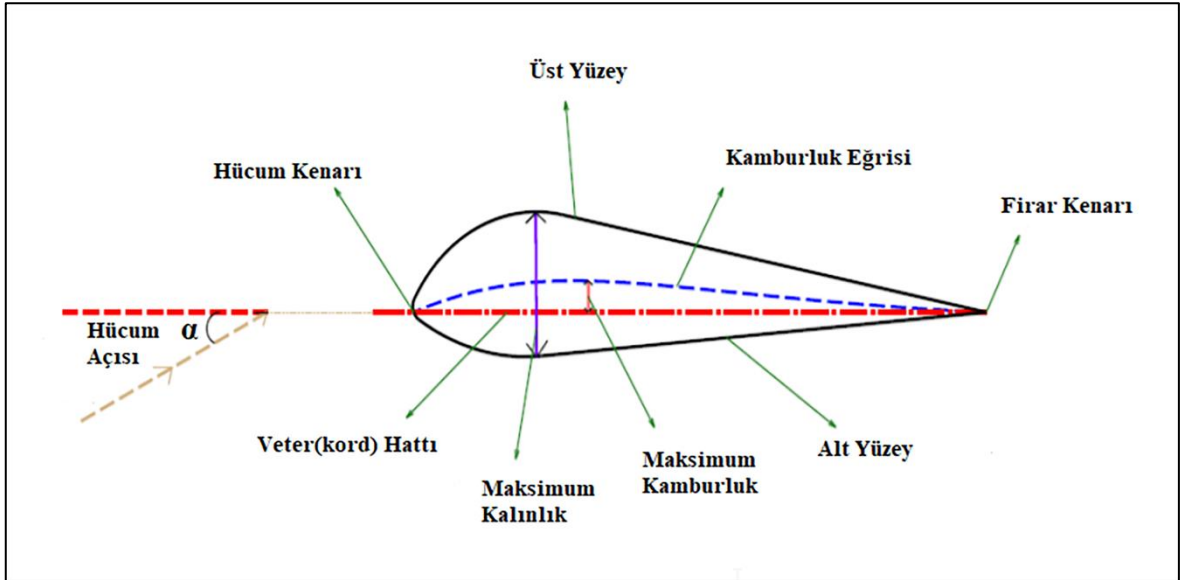
damla şeklinde olan yapılardır.



Şekil 3.2. İki boyutlu akım modeli (Yükselen, 2006a).

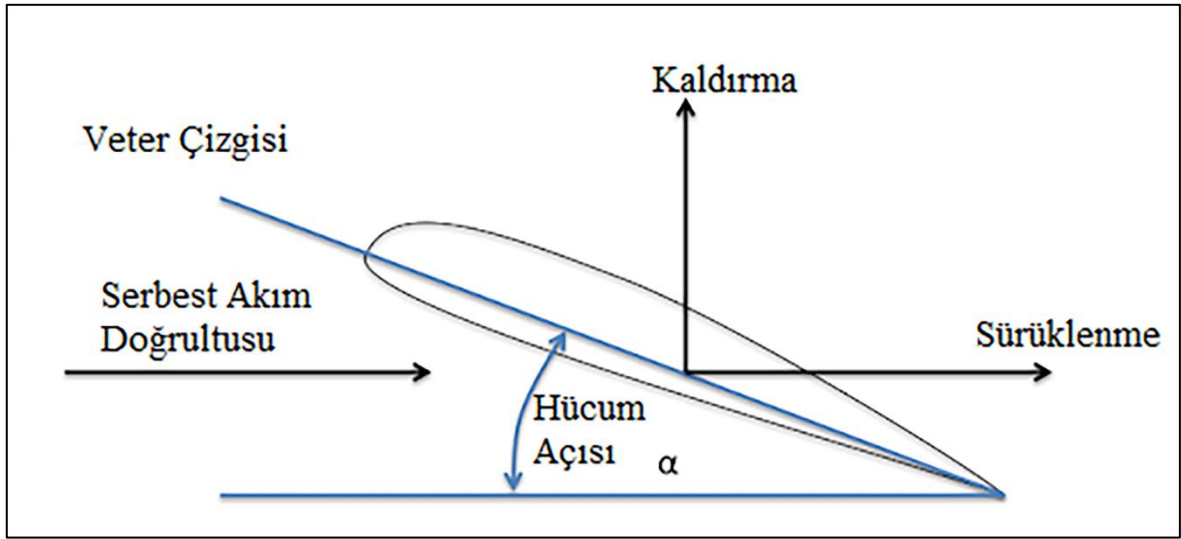
Uçak kanadı ve pervane gibi parçalar üzerindeki akımın çoğu zaman iki boyutluya oldukça yakın olduğu görülür. Bu nedenle bu materyaller başlangıçta iki boyutlu varsayılarak incelenir. Şekil 3.2 de görülen kanadın iki ucunun sonsuza gidiyor olduğu düşünülür. Tüm kesitleri birbiriyle aynıdır. Kanadın açıklığına dik kesitler boyunca akım gider ancak açıklık boyunca akım söz konusu değildir. İki boyutta inceleme söz konusu olmasından dolayı tüm kesitlerde birbiri ile eşit olması sebebiyle aerodinamik inceleme söz konusu olduğunda tek bir kesitteki akımın incelenmesi yeterli olmaktadır.

3.2. Kanat Profili Geometrisi



Şekil 3.3. Kanat profil geometrisi (Kevadiya ve Vaidya, 2013).

Şekil 3.3 de görüldüğü gibi kanat profilinde uçuş doğrultusu ve aynı zamanda yönünü göz önünde bulundurarak kanadın en önde bulunan noktalarına hücum kenarı, kanadın en geride kalan noktalarına ise firar kenarı denir (Yükselen, 2006b). Hücum kenarı ile firar kenarı arasındaki en kısa uzaklık, veter(kord) hattı olarak adlandırılır. Hücum kenarı ile firar kenarını birleştiren üst yüzey ile alt yüzey arasındaki veter hattına dik çizilerek elde edilen doğru parçalarına kalınlık denir, bu elde edilmiş doğru parçalarının da orta noktaları birleştirildiğinde elde edilen eğriye de kamburluk eğrisi adı verilir.



Şekil 3.4. Hücum açısı (Yılmaz, Koten, Çetinkaya ve Coşar, 2018).

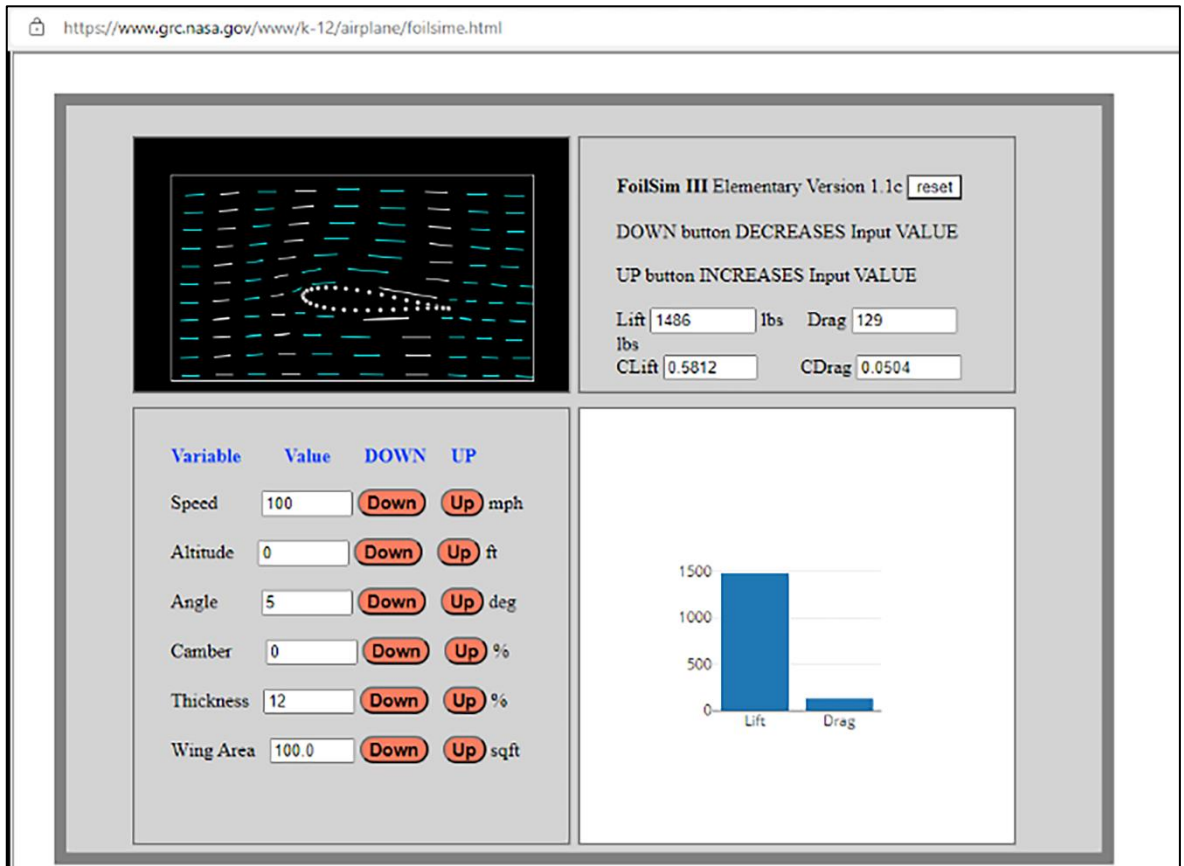
Hücum açısı (α), kanat veter hattı ile serbest akım doğrultusu arasında kalan açı değeridir. Hücum açısının aerodinamik kuvvetler üzerinde etkisi önemli boyuttadır. Dolayısıyla akış ve kanat profil geometrisi arasındaki bağlantı hakkında bize bilgi verir (Pehlivanoğlu, 2013).

3.3. NASA FoilSim Web Sayfası

FoilSim III Elementary Version 1.1c, bir uçak kanadındaki kaldırma ve sürüklenme hakkında bilgi edinmemizi sağlayan bir web sayfasıdır ve indirilebilir bir programdır. Web sayfasında kanada ait hız, hücum açısı, kamburluk, kalınlık, kanat alanı bilgileri ve kanada etki eden hava yoğunluğu bilgileri girilip optimum kaldırma ve sürüklenme bilgileri edinilebilmektedir.

FoilSim III, interaktif FoilSim ailesinin en son (Haziran 2010) sürümüdür. FoilSimin farklı sürümleri farklı aerodinamik ve bilgisayar programları bilgisi gerektirir. İlgili web sayfası FoilSim III programının çevrimiçi temel sürümünü içerir ve programdaki çeşitli

seenekleri aıklayan bir online kılavuz iermektedir.



Şekil 3.5. FoilSim web sayfası.

3.4. Yapay Sinir Ağları (YSA)

İnsan beyni çalışma prensibi hakkında uzunca bir süre çalışmalar yapılmıştır. Bu çalışmalardan ilki bilimsel olarak 1980lerde ortaya konulmuştur (James, 1984). Yapay sinir ağı modeline yakın ilk çalışmalar Warren Mjamesch ve Walter Pitts tarafından gerçekleştirilmiştir. Bu araştırmacılar yapay sinir ağlarının temellerini insan beynini hesaplama kabiliyetinden ilham alarak ilk yapay sinir ağı hücresini geliştirmişlerdir. Daha da ilerleyerek her türlü mantıksal ifadenin formülize edilebileceğini gösterip sonrasında da bu sayede birden fazla sinir hücresinin paralel olarak çalışma tekniğini geliştirmişlerdir. Böylelikle öğrenme kurallarını belirlemeye başlamışlardır (McCulloch ve Pitts, 1943).

1949’da Donald Hebb, “Hebbian Öğrenme” adında bir hipotez oluşturmuştur. Bu hipotez sayesinde sinir ağındaki bağlantı sayısı ile öğrenme ve uyum arasındaki ilişki tespit edilmiştir (Hebb, 1949). Farley ve Clark 1954 yılında yaptıkları çalışma ile Hebb in hipotezine karşılık olarak hesaplama makinelerini kullanarak rassal ağlarla adaptif tepki üretme kavramını ortaya çıkarmışlardır (Farley ve Clark, 1954). Frank Rosentblatt

tarafından tek katman ve çıkıştan oluşan algılamının keşfi ile yapay sinir ağı alanında yapılan çalışmalar hız kazanmıştır (Öztemel, 2006). Daha sonraları Rosentblatt bir sineğin gözü ile ilgili “kavrama düğümü” anlamına gelen ağ kavramının günümüze kadar gelen bir kavram olmasını sağlamıştır (Elmas, 2018).

Bernard Widrow ve Marcian Hoff yapay sinir ağlarının mühendislikte kullanılan ilk modelini oluşturmuşlardır. Bu modeller ADALINE (Adaptive Lineer Neuron) ve MADALINE (Many Adaptive Lineer Neuron) olarak isimlendirilmektedir. ADALINE modeli kendinden sonraki yapay sinir ağı modelini oluşturmuştur (Widrow ve Hoff, 1989). MADALINE telefon hatlarında oluşan yankı problemlerini çözerek gerçek problemlere uygulanan ilk sinir ağı olmuştur.

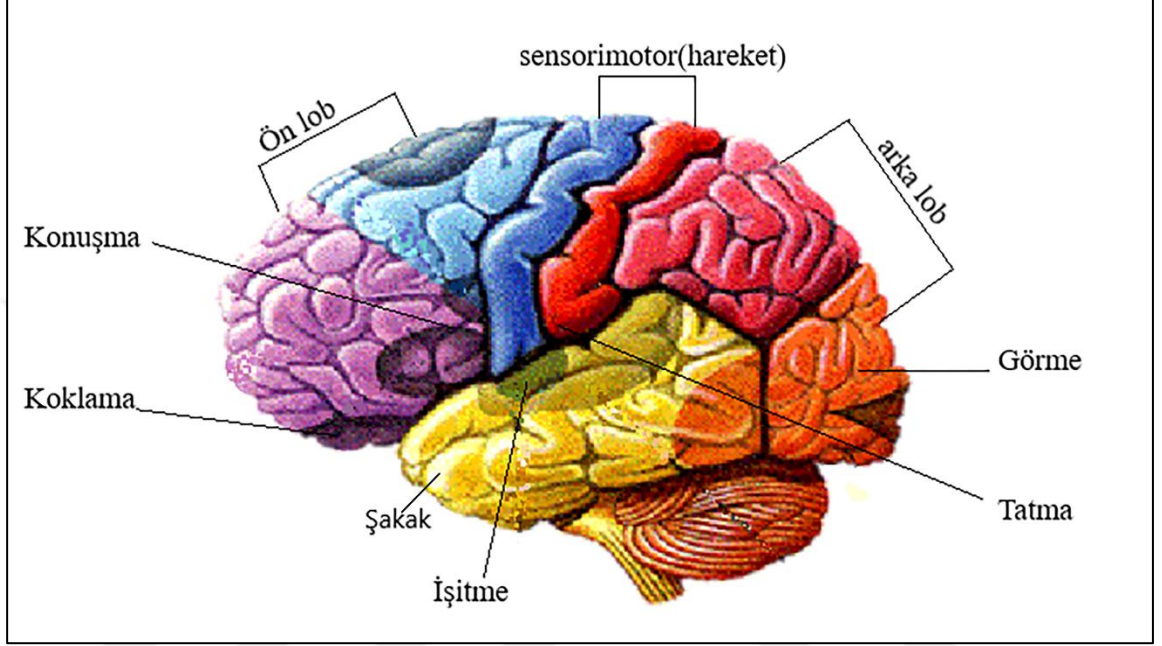
1969 yılında yapay sinir ağları büyük gerileme yaşanmıştır. Pappert ve Minsky ‘Algılayıcılar’ adlı kitabında yapay sinir ağlarının lineer olmayan sorunlara çözüm üretemediğini kanıtlamaya çalıştılar ve XOR problemini çözebilmesi için iki katmandan oluşan ileri beslemeli bir yapay sinir ağı ile tek katmanlı ağlardaki eksikliği çözmüş ancak ağırlık katsayılarının nasıl güncelleneceğini açıklayamamışlardır (Minsky ve Papert, 1969). Fukushima’nın NEOCOGNITRON modelini bilim camiasına kazandırmasıyla görüntü işleme ve örüntü tanıma alanında ilerleme elde edilmiştir.

1970 de Linnainma iç içe farklılaşabilir fonksiyonların ayırık bağlantılı ağlarının otomatik olarak farklılaştırılması için bir yöntem açıklamışlardır. Dreyfus kontrolcülerinin parametrelerinin hata gradyanıyla orantılı olarak uyarlamak için backproagation kullanmıştır. İlerleyen zamanlarda Webos’un backproagation algoritması çok katmanlı ağların eğitilmesini sağlamıştır (Öztemel, 2006).

Kohonen “Kendi kendine öğrenme nitelik haritaları” adlı çalışmasını 1982 yılında yayınladı (Kohonen, 1982). Bu süreçte yapay sinir ağlarının bilgisayar yazılımları ile çözülmesi zor olan sorunlara çözüm bulabileceği gösterilmiş oldu. Kohonen danışmansız öğrenme ağını geliştirmiş ve sinirlerin düzenli sıralanışına eşleme özelliğini de eklemiştir. 1987 yılında Yapay Sinir Ağları konulu ilk uluslararası konferans Elektrik Elektronik Mühendisliği Enstitüsü (IEEE) vasıtasıyla gerçekleşmiştir (Elmas, 2018). Yapay sinir ağlarındaki bu ilerleme bilgisayar aksamalarının gelişmesi ile de büyük ilerleme kat etmiştir. Yapay sinir ağları teorik uygulamaların dışında artık ihtiyaç doğrultusunda çözüm odaklı çalışmalarda kullanılmaktadır.

3.4.1. Biyolojik Sinir Hücresi ve Yapay Sinir Ağının Temel Öğeleri

Beynimizde bulunan milyarlarca sinir hücresinin birleşmesiyle biyolojik sinir ağı meydana gelmektedir. Sinir hücreleri arasında bir bağ vardır ve bu bağlar sayesinde görevlerini yerine getirebilirler. İnsan beyinde bulunan biyolojik sinir ağlarının kabiliyetlerinden esinlenilerek YSA bilgisayara kazandırılmak istenmiştir (Öztemel, 2006).

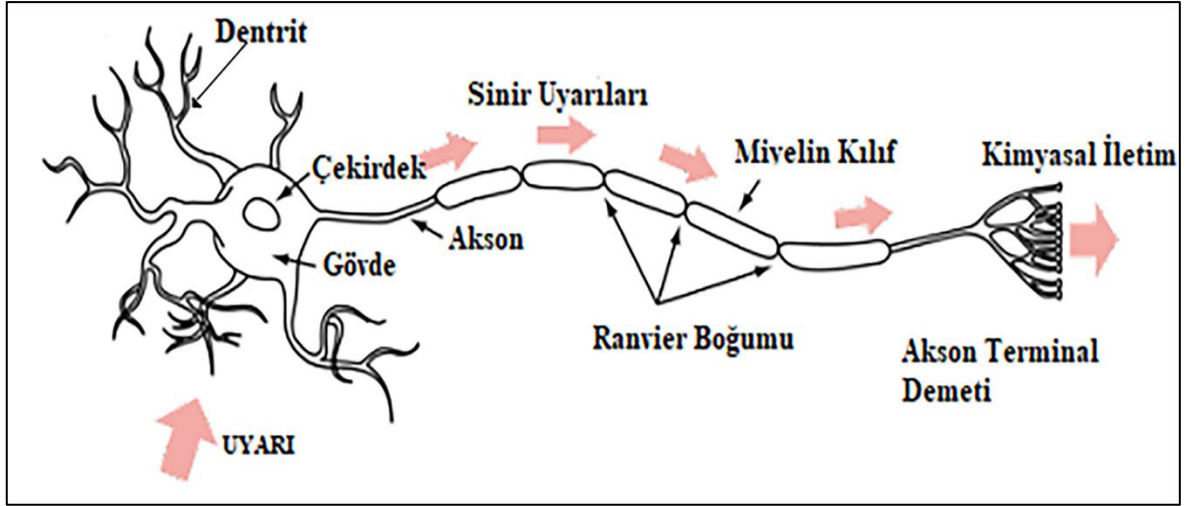


Şekil 3.6. Beyin üzerindeki temel duyu bölgeleri (Nwadiugwu, 2020).

İnsan beyninin çalışmasını sağlayan en önemli yapılar biyolojik sinir ağıdır. Biyolojik sinir ağı insan bedeninde bulunan beş duyu organından gelen bilgiler aracılığıyla oluşturduğu anlama mekanizmasıyla olaylar arasındaki bağlantıyı kurar. Şekil 3.6’de insan beyninin beş duyu bölgesinin beyin içerisindeki konumları gösterilmiştir.

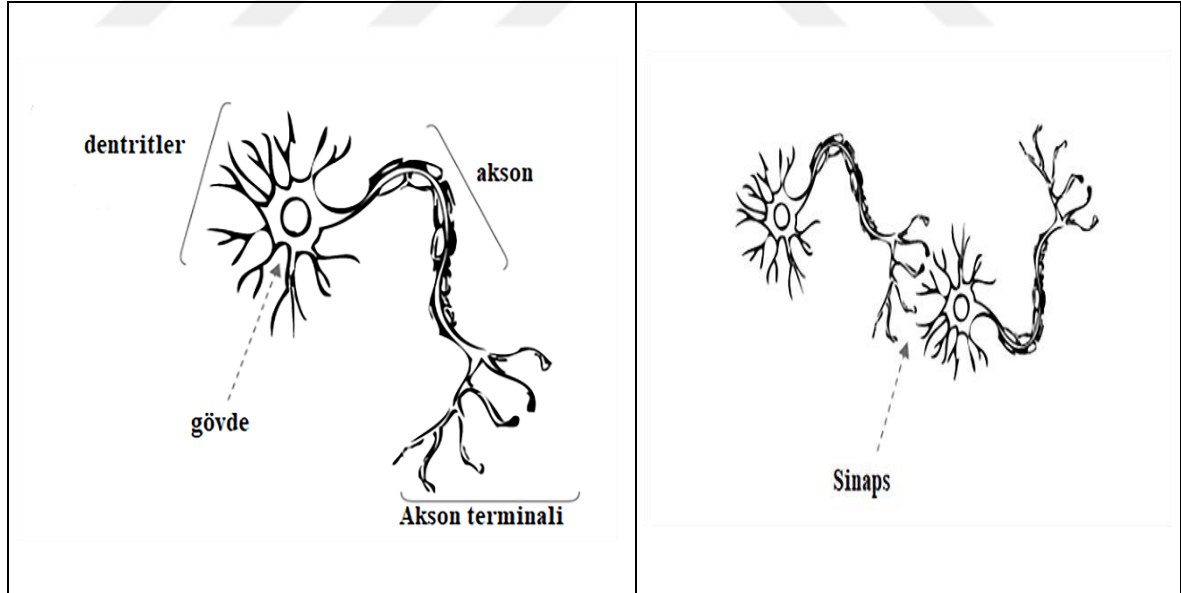
Öncelikli olarak duyu organlarından gelen sinyaller, sinir sistemi aracılığıyla beyine taşınmakta ve sonrasında beynin verdiği kararları yine sinir sistemi aracılığıyla vücudun ilgili organlarına aksiyon olarak gönderilmektedir.

Temel olarak bir biyolojik sinir hücresi dentritler, gövde, çekirdek, akson ve sinapslardan meydana gelmektedir. Sinapslar, sinir hücreleri arasındaki bağlantı olarak adlandırılmaktadır. Bir sinir hücresinden diğer bir sinir hücresine iletilen elektrik sinyalleri sinaps boşlukları aracılığıyla sağlanır.



Şekil 3.7. Biyolojik sinir hücresi (Maltarollo, Honório ve Silva, 2013).

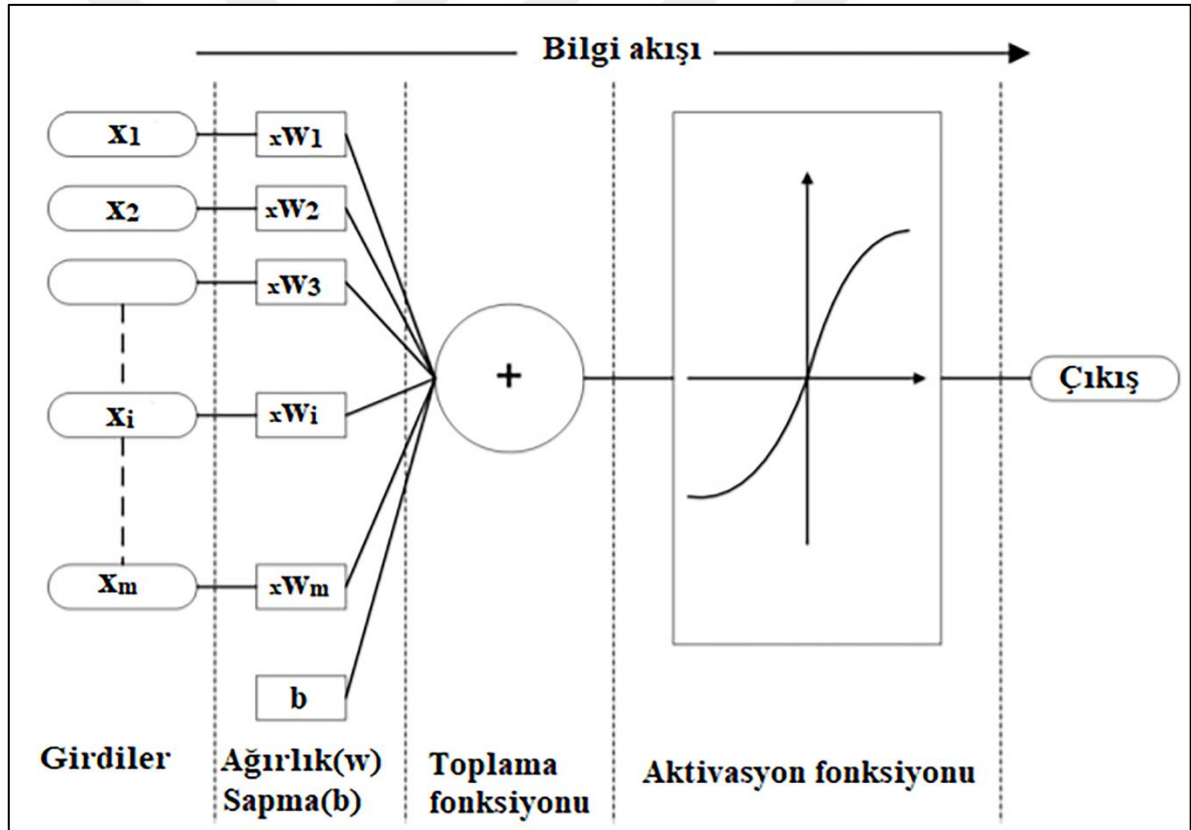
Sinir hücresinin uçlarında dentritler yer almaktadır. Duyu organları ya da bağlantılı diğer nöronlardan gelen sinyallerin çekirdeğe iletilmesinde dentritler görev almaktadır (Yılmaz, 2020). Bahsi geçen sinyaller gövdeye gider, gövdede gerekli işlemi görür, sinir hücresi kendine ait elektrik sinyalini oluşturur ve akson aracılığıyla diğer hücrenin dentritlerine gönderir (Öztemel, 2006). Miyelin kılıf yayılma hızına etki eden yalıtımlı katmandır (Kızrak, 2018).



Şekil 3.8. Biyolojik sinir hücresi ve sinaps bölgesi (Maltarollo, Honório ve Silva, 2013).

Tablo 3.1. Yapay sinir hücresi ve biyolojik sinir hücresi benzeştirilmesi (Öztürk ve Şahin, 2018).

Biyolojik Sinir Hücresi	Yapay Sinir Hücresi
Nöron	Düğüm
Dentrit	Toplama Fonksiyonu
Hücre Gövdesi	Aktivasyon Fonksiyonu
Aksonlar	Çıkış
Sinapslar	Ağırlıklar



Şekil 3.9. Yapay sinir hücresi (Krenker, Beşter ve Kos, 2011).

Girdi (x), bir yapay sinir ağı hücresine dış dünyadan, başka hücrelerden veya kendi kendisinden gelen bilgileri ifade etmektedir.

Ağırlık (w), bir yapay sinir ağı hücresine gelen bilginin önemini ve bu hücreye etkisini ifade etmektedir. Ağırlığın büyük ya da küçük olması veya eksi değer olması önemli

ya da önemsiz olduğu anlamına gelmez. Eksi veya artı olması etkisinin pozitif ve negatif olma durumunu anlatmaktadır.

Toplama işlevi, bir hücreye gelen net girdiyi hesaplamaktadır. Bu hesaplamayı yapmak için farklı fonksiyonlar kullanılmaktadır. En çok kullanılanı ağırlıklı toplamdır. Bu yöntemde her girdi kendi ağırlığı ile çarpılmaktadır. Bir yapay sinir ağında bulunan proses elemanlarının hepsinin aynı toplama fonksiyonuna sahip olma zorunluluğu yoktur; her biri farklı fonksiyon da kullanabildiği gibi hepsi aynı fonksiyon da kullanabilir. Herhangi bir problemin çözümü için en uygun toplama fonksiyonunu belirleyen net bir formül bulunmamaktadır. Girdiler $x_1, x_2, \dots, x_n$; ağırlıklar $w_1, w_2, \dots, w_n$; çıktılar $y_1, y_2, \dots, y_n$ olarak adlandırıldığında toplam fonksiyonun denklem, Eşitlik 3.1 de gösterilmektedir.

$$y = \sum_{i=1}^n x_i \cdot w_i \quad (3.1)$$

Aktivasyon fonksiyonu, hücreye gelen net girdiyi işleyerek hücrenin bu girdiye karşılık üreteceği çıktıyı belirler. Aktivasyon için kullanılan çeşitli fonksiyonlar vardır. En sık kullanılan aktivasyon fonksiyonlarından biri sigmoid fonksiyonudur (Shiruru, 2016). Başlıca aktivasyon fonksiyonları doğrusal fonksiyon, sigmoid aktivasyon fonksiyonu, hiperbolik tanjant fonksiyonu ve ReLU'dur.

Doğrusal fonksiyon, bu fonksiyona gelen giriş değerleri doğrudan hücrenin çıktısı olarak kabul edilmektedir. Toplama fonksiyonundan hesaplanan sonuç, belirlenen katsayı ile çarpılarak hücrenin çıkışı belirlenir (Öztemel, 2006; Yılmaz, 2020).

Sigmoid aktivasyon fonksiyonu, doğrusal olmayan problemlerin çözümü için doğru aktivasyon fonksiyonunu kullanmak gerekmektedir. Bu durumda önemli olan kısım türevlenebilir bir fonksiyona ihtiyaç duyulmasıdır. Çok katmanlı yapay sinir ağı mimarisinde Sigmoid fonksiyonu bu sebeple tercih edilmektedir. Fonksiyon girdi değerlerine karşılık 0 ile 1 arasında bir değer üretir (Yılmaz, 2020).

Hiperbolik tanjant fonksiyonunun dikey olarak taşınması ve yeniden ölçeklenmesi özellikleri dışında Sigmoid fonksiyonuna benzerdir. Bu fonksiyona ait çıkış değeri -1 ile 1 arasında değişmektedir (Aggarwal, 2018).

ReLU (Rectified Linear Unit: Doğrultumuş Lineer Birim), ağda bazı nöronların aktif olup aktivasyonu seyrek olan yani verimli bir hesaplama yükü oluşturulmak istendiğinde kullanılır. Sigmoid ve hiperbolik tanjant fonksiyonlarında neredeyse tüm nöronların aynı

şekilde aktive olması sebep olduğundan ilgili aktivasyon çok fazla işlem gerektirmektedir. Güncel çalışmalarda da ReLU daha fazla kullanılır hale gelmiştir. ReLU aktivasyon fonksiyonunun ürettiği çıktı 0 ile sonsuz arasındadır (Kızrak, 2019; Aggarwal, 2018).

Çıkış, aktivasyon fonksiyonu aracılığıyla üretilen değerdir.

Tablo 3.2. Aktivasyon fonksiyonlarının matematiksel olarak gösterimi.

Aktivasyon Fonksiyonu	Denklem	Aralık
Doğrusal Fonksiyon	$f(x) = x$	$(-\infty, \infty)$
Sigmoid Fonksiyon	$(x) = \frac{1}{1 + e^{-x}}$	$(0,1)$
Hiperbolik Tanjant Fonksiyonu	$(x) = \frac{e^x + e^{-x}}{e^x - e^{-x}}$	$(-1,1)$
ReLU	$(x) = \begin{cases} 0, & x < 0 \\ x, & x \leq 0 \end{cases}$	$[0, \infty)$

3.4.2. Yapay Sinir Ağı'nın Yapısı

Temel olarak çoğu yapay sinir ağı, Şekil 3.9 da gösterildiği gibi benzer bir yapıya sahiptir. Bu yapıda bazı nöronların girdilerini almak için dış dünya ile doğrudan bağlantısı varken bazılarının doğrudan bağlantısı bulunmamaktadır. Bir sinir ağı, bir grup nörondan daha fazladır. Konu ile ilgili ilk çalışmalarda nöronları rastgele bir şekilde basitçe bağlamaya çalışmışlardır ancak başarılı olamamışlardır. Sonrasında bir yapıyı tasarlarken en basit yolun katmanlara ayırmak olduğu keşfedilmiştir (Abdulla, Maind ve Wankar, 2014). Yapay sinir ağına ait katmanlar aşağıdaki gibi sıralanabilir;

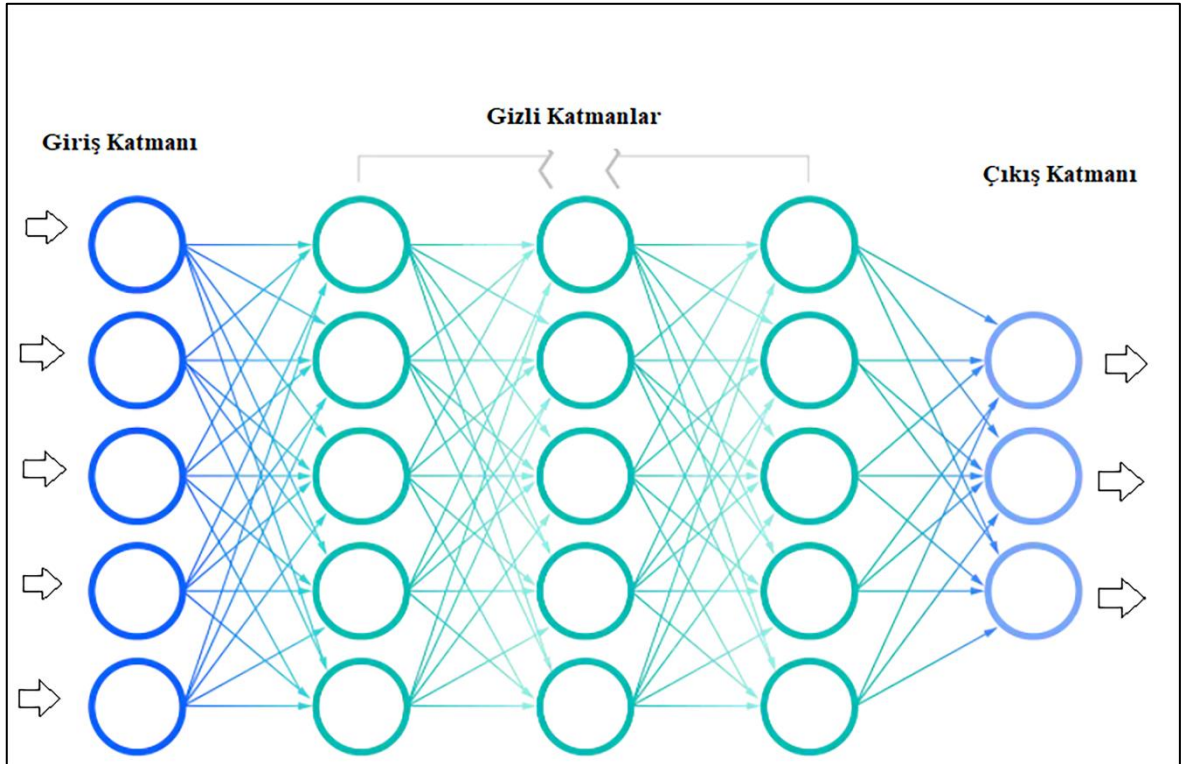
Giriş Katmanı: Giriş değerlerini alır.

Gizli Katman/Katmanlar: Giriş ve çıkış katmanları arasında bulunan bir dizi nöronlardır. Tek ya da çoklu katmanlar şeklinde olabilir (Shiruru, 2016). Gizli katmanlar, giriş katmanından aldıkları bilgiler ile gerekli hesaplamaları yaparak kendilerinden sonra gelen katmana bu bilgiyi iletirler (Öztemel, 2006).

Çıktı Katmanı: Bir veya birden fazla çıktı mevcut olabilmektedir(Shiruru, 2016). Önceki katmandan gelen bilgileri işler ve bir aktivasyon fonksiyonu aracılığıyla çıktıları üretir (Öztemel, 2006).

Yapay sinir ağları ağı'nın yönüne göre değerlendirildiğinde ileri beslemeli ve geri

beslemeli ağlardan söz edilmektedir. İleri beslemeli ağlarda işaretler girişten çıkışa doğru tek yönlüdür (Kaya ve Elfarra, 2019; Yılmaz, 2020). Geri beslemeli yapay sinir ağlarında ise ileri beslemeli yapay sinir ağlarından farklı olarak giriş katmanı ve gizli katmanlar arasında çift yönlü bir bağlantı söz konusudur. Bu çift yönlü bağlantı aynı zamanda çıkış katmanı ve gizli katmanlar arasında da mevcuttur (Öztemel, 2006). Katmandaki nöronlardan bir sonraki tüm nöronlara bağlantı varken, aynı katman içindeki nöronlar birbirine bağlı değildir. Gizli katman sayısı ve gizli katmanda bulunan hücre sayısı problemlerin yapısı ile farklılık gösterebilmektedir (Yılmaz, 2020).



Şekil 3.10. Yapay sinir ağı (Sinir Ağları nedir?, 2020).

3.4.3. Yapay Sinir Ağlarının Genel Özellikleri

Yapay Sinir Ağları temel bazı özelliklere sahip olmaktadır. Bunlardan bazıları;

- Doğrusal olmama
- Genelleme
- Eksik veriler ile çalışma
- Çok sayıda değişken ve parametre kullanma
- Genelleme
- Öğrenme
- Uygulanabilirlik

- Paralel çalışma

Yapay sinir ağıları en çok tahmin, sınıflandırma, veri ilişkilendirme, veri yorumlama ve veri filtreleme işlemlerinde kullanılmaktadır (Öztürk ve Şahin, 2018).

Tahmin, ilgili projeler için geliştirilen yapay sinir ağıları verilen giriş değerlerinde bir çıktı tahmin etmek üzere kurgulanmıştır. Örneğin otoyol trafik yoğunluğu tahmini.

Sınıflandırma, ilgili projelerde kullanılan yapay sinir ağıları, girdi değerlerini sınıflandırma görevini üstlenir. Göğüs kanseri verilerinin yapay sinir ağıları ile sınıflandırılması buna örnek verilebilir.

Veri filtrelemenin kullanıldığı çalışmalar için eğitilen ağlar, erişilen veriler arasından en faydalı olanı tespit etme görevini yerine getirir.

Veri ilişkilendirme, öğrendiği bilgilerle konular arasında ilişkilendirme yapar ve bu sayede eksik olan bilgileri tamamlamasına fırsat doğmuş olur.

Veri yorumlama, bir araştırma konusu ile ilgili önceden edinilen bilgiler ile eğitim sonucunda edinilen bilgileri derleyip başka yeni olayların yorumlanabilmesini sağlar (Ağyar, 2015).

3.4.4. Yapay Sinir Ağlarında Öğrenme Süreci

YSA'da giriş verileri dış dünyadan sisteme verilerek öğrenme işleminin başlangıcı yapılmış olur. Sonrasında alınan veriler işlenir, işlenen veriler aktivasyon fonksiyonuna ulaşır ve çıkış değeri üretilmiş olur. Sistemin çıkışı eğitim setinden girişe karşılık gelmesi gereken, beklenen değer ile karşılaştırılarak hata değeri bulunmuş olur. Amaç hata değerini azaltarak beklenen çıkışa ulaşmaktır. Bu işlem boyunca sürekli olarak bu ağa ait ağırlık değerleri güncellenir. Bahsi geçen ağırlıklar her iterasyonda güncellenerek beklenen çıkışa yaklaştırmaya ve hata minimize edilmeye çalışılır. Yapay sinir ağı öğrenme aşamasını tamamladıktan sonra daha önce ağa verilmeyen girişler verilerek eğitilen ağ test edilir.

Yapay sinir ağına ait öğrenme adımları:

1. Veri setinin toplanması: Ağın öğrenmesi için kullanılacak verilerin toplanması sürecidir.
2. Ağ yapısının belirlenmesi: Bir sonraki adımda ağın öğrenmesi istenilen konu için ağın yapısının belirlenmesidir. Bu aşamada kaç adet girdi olacağı kaç adet gizli katman olacağı, bununla birlikte gizli katmanlarda kaç adet nöronların olacağı ve sonuç olarak kaç adet çıktı olacağı belirlenmektedir.

3. Ağın öğrenme parametrelerinin tespiti: Ağ için kullanılacak yapay sinir ağı yapısının belirlenmesi aşamasının tamamlanmasıyla birlikte eğitim süreçlerinin devam etmesi için gerekli olan aktivasyon fonksiyonu, öğrenme oranı gibi parametreler bu aşamada tespit edilmektedir.
4. Ağırlık ve sapma değerleri: Bu değerler ağın kendisi tarafından başlangıçta rastgele verilmek üzere sonrasında da öğrenme sürecinde ağ tarafından güncellenerek ideal değerler belirlenmektedir (Elmas, 2018).
5. Ağın eğitimi için veri setinin seçilip ağı sunulması: Hazırlıkların tamamlanmasıyla birlikte ağ öğrenme süreci için hazırdır. Öğrenmeye başlaması için veri seti girdi ve çıktı değerleri verilir. Ağın problemi öğrenmesi için kullanılacak ağırlık ve eşik değerinin ilk değerleri rastgele olarak verilir ve öğrenme sırasında ortaya çıkan hata değerlerine göre güncellenmesi yapılır.
6. İleri hesaplamaların yapılması: Veri girdi değerlerine uygun çıktı değerlerinin üretilmesinin hesaplanması bu adımda gerçekleştirilmektedir (Yılmaz, 2020).
7. Ağdan oluşan çıktı değerleri beklenen çıktı ile karşılaştırılarak hata değerleri hesaplanır.
8. Hesaplanan hatayı azaltmak amacıyla geriye doğru hesaplama yardımıyla ağırlık güncellemesi yapılır.
9. Ağ öğrenmeyi tamamlayıncaya kadar işlemler devam eder.

Yapay sinir ağlarında elde edilen girdiler ile çıktı üretilmesini sağlayan yöntem öğrenmedir. Öğrenme de farklı algoritma yöntemleriyle gerçekleştirilebilir.

Danışmanlı öğrenme, çözmek istenilen probleme ait giriş değerlerine karşılık gelen çıkış değerlerinin de verildiği öğrenme yöntemidir. Veri setinin bir kısmı etiketlenmiş veri seti olarak adlandırılır ve bu grup kullanılan algoritma ile sistem çıktısı arasında bir ilişki kurarak karar verme yöntemini belirlemeyi amaçlar ve bu süreç içerisinde öğrenme sürecini de tamamlar. Ağı bulduğu çıktı değeri ile beklenen çıktı arasında bulunan fark sayesinde hata değeri hesaplanır ve bu değer ağırlıkların güncellenmesinde kullanılır (Sammut ve Webb, 2010).

Danışmansız öğrenme, çözülmesi istenilen probleme ait elde sadece girdi değerleri varsa ve ağı sadece bu girdi değerlerinden sonuç elde etmesi bekleniyorsa seçilen yöntem danışmansız öğrenme yöntemidir. Problemi çözecek olan ağ, kendisine verilen girdileri sınıflar ve ağırlıklarda yine bu sınıflandırma işlemlerini yapabilecek şekilde düzenlenmesi gerekmektedir. Veri setlerinde etiketleme bulunmaması öğrenme sürecini zorlaştıran bir

faktördür (Yılmaz, 2020).

Yarı Danışmanlı öğrenme modelinde yine öğretici verilse de danışmanlı öğrenme kadar detaylı değildir. İterasyon gerçekleştikten sonra ağın sonucu hakkında iyi veya kötü olduğuna dair ağa bilgi vererek, bu bilgiler sayesinde ağın kendini tekrar düzenlemesi sağlanır (Yılmaz, 2020).

3.4.5. Yapay Sinir Ağı için Verilerin Toplanması

Kaldırma miktarını hesaplariken kullanılacak en önemli parametre Kaldırma katsayısıdır. C_L katsayısı kompleks değişkenler içermekte olup genellikle deneysel olarak elde edilmektedir. C_L katsayısı kanadın hücum açısına, kamburluğuna ve kalınlığına bağlı olan bir katsayıdır. NASA tarafından geliştirilmiş FoilSim online arayüzünde girilen hücum açısı, kamburluk ve kalınlığa bağlı olarak kanada ait C_L değeri kullanıcılar için hesaplanmaktadır. FoilSim online arayüzünde hücum açısı($^\circ$), kamburluk(%) ve kalınlık(%) girdileri herbiri için 0-20 aralığındadır. Söz konusu değerler 0-20 arasında tamsayı veya ondalıklı değerler olabilmektedir. Bu çalışma kapsamında kullanılan C_L veri seti doğrudan FoilSim online arayüzüne hücum açısı, kamburluk ve kalınlık değerleri girilerek toplanmıştır. Söz konusu veri seti; hücum açısı, kamburluk ve kalınlık değerlerinin her biri için 0-20 arasında (1-1-1 den başlayarak 20-20-20 ye kadar) tamsayı değerleri girilerek 20x20x20=8000 adet C_L değerinden oluşmaktadır.

8000 adet verinin toplanması için MATLAB programı ve Java awt Robot kullanılmıştır. Robot, kullanıcının yapacağı işleri yapan bir uygulamadır. Yani kullanıcı yerine fareyi hareket ettirir, butonlarına basar ve ayrıca klavyeden tuşlara basar. Bu robot çeşitli amaçlar için kullanılır. Bu çalışmada Robot, FoilSim online arayüzünde hücum açısı, kamburluk ve kalınlık değerlerini 1-1-1 den 20-20-20 ye kadar girerek her bir değer için elde edilen C_L değerini bir diziye kaydetmektedir. Hücum açısı, kamburluk ve kalınlık değerleri 1-1-1, 1-1-2, 1-1-3 ... 1-1-20, 1-2-1, 1-2-2, 1-2-3 ... 1-20-20, 2-1-1, 2-1-2, 2-1-3 ... 20-20-20 şeklinde en küçük değerlerden (1-1-1) başlayarak en büyük değerlere kadar (20-20-20) ayrı ayrı girilerek her farklı durum için C_L değerleri otomatik olarak toplanmıştır.

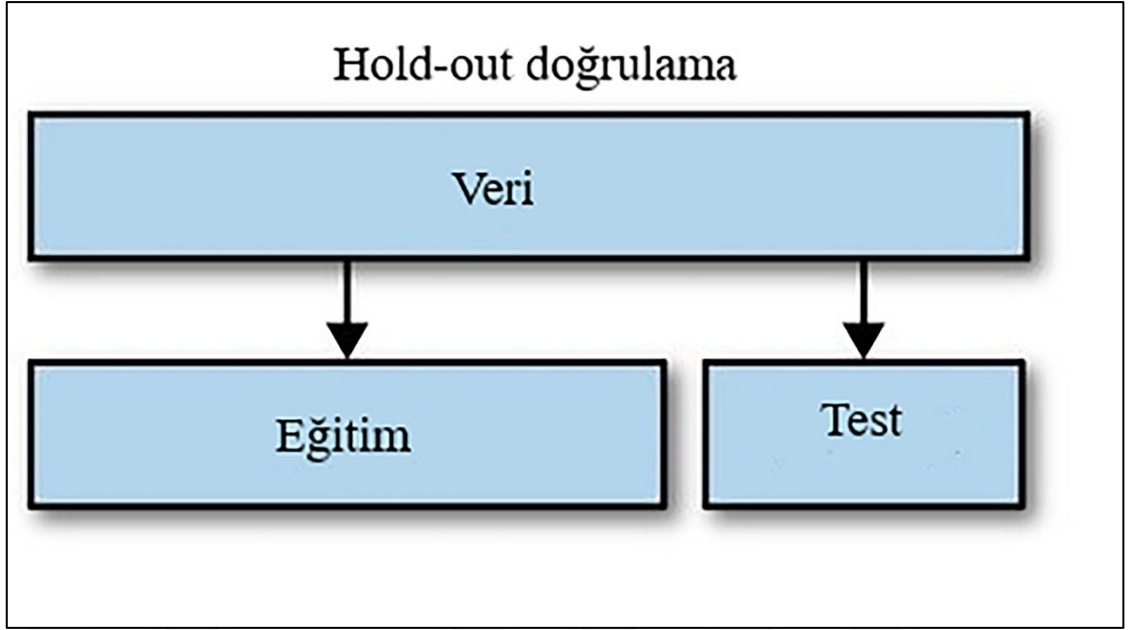
3.4.6. Yapay Sinir Ağında Yer Alan Hiperparametreler

Yapay sinir ağı algoritmalarında, algoritmalara ait davranışların kontrol edilmesi hiperparametreler aracılığıyla gerçekleşmektedir. Bu hiperparametreler öğrenme

algoritmaları tarafından belirlenemez, ağı oluşturan tasarımcı tarafından oluşturulması gerekir (Goodfellow, Bengio ve Courville, 2016).

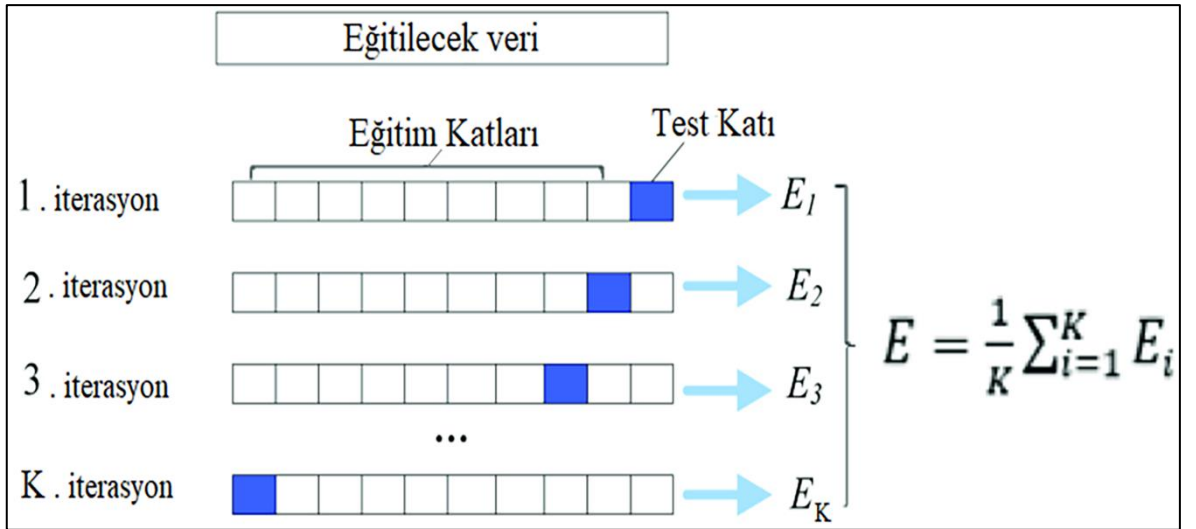
Ağa ait hiperparametrelerin belirlenmesinde faydalanan çapraz doğrulama yöntemi, verileri bölümlere ayırarak algoritmaları değerlendirmenin ve karşılaştırmanın istatistiksel bir yöntemidir. Bu bölümlerden biri modeli eğitmek diğeri ise modeli doğrulamak için kullanılır (Refaeilzadeh, Tang ve Liu, 2009). Çapraz doğrulama, tahmine dayalı modellerin genelleme yeteneğini değerlendirerek fazla uydurmayı önlemeyi amaçlar (Berrar, 2018). Genel kullanım alanı olarak bir öngörü/regresyon sisteminin pratikte hangi doğrulukta, başka bir deyişle ne kadar hata ile çalışacağını kestirmektir. Çapraz doğrulanın ilk adımı olarak farklı sayılarda örnekler seçilerek veri kümeleri oluşturulur. Eğitim kümesi üzerinde analiz yapılarak model oluşturulur, test kümesinde ise model test edilir. Bu doğrulama birkaç defa tekrar edilir. Çapraz doğrulama, adını aynı veri örneklerinin farklı şekillerde gruplandırılmasından alır. Çaprazlama sonrasında gruplanan veriler üzerinde yapılan testlerin sonuçları birleştirilerek modelin genel tahmin başarısı elde edilir (Cross-validation, [https://en.wikipedia.org/wiki/Cross-validation_\(statistics\)](https://en.wikipedia.org/wiki/Cross-validation_(statistics))). Çapraz doğrulama yöntemlerinden bazıları aşağıdaki gibi sıralanabilir.

Holdout Yöntemi ile eğitim ve test verileri belirlenen oranda ayrılır ve algorithmada uygulanarak test edilir. Test verisi eğitim verisine oranla daha az miktardadır. Yeterince veri yoksa sınıflandırma/regresyon algoritmasını test verilerini öğrenmede kullanmadığı için ağın performansı düşebilmektedir. Test setindeki örneklerin sınıflandırılması çok kolay veya çok zor olabilir ve bu da sonuçları çarpıtabilir. İyi bir bölümlendirme işlemi yapılmaması durumunda tek bir test sürecinden geçildiği için algoritmanın yanıltıcı hata oranı döndürmesine sebep olabilmektedir. Bu gibi zorluklarla başa çıkabilmek için k-katlamalı çapraz doğrulama kullanılır (Refaeilzadeh, Tang ve Liu, 2009).



Şekil 3.11. Holdout yöntemiyle verinin bölümlendirilmesi (Mishra, 2019).

K-katlı çapraz doğrulamada veriler başlangıçta eşit (ya da neredeyse eşit) boyutlu parçalara ayrılmaktadır. Burada ‘k’ sonuçta ortaya çıkan alt kümelerin sayısını ifade eder. Ardından her yinelemede k alt kümelerinden biri test kümesi olarak kullanılırken diğer k-1 alt kümeleri bir eğitim kümesi oluşturulmak üzere bir araya getirilmektedir. Tüm alt kümelerdeki hatalar hesaplanır, ardından ortalama hata ve performans hesaplanır (Berrar, 2018). E, her iterasyondaki hatayı temsil etmektedir.



Şekil 3.12. K-katlamalı çapraz doğrulama (Ashfaq ve Markowet, 2019).

Özel bir k-katlamalı çapraz doğrulama yöntemi olan tek çıkışlı çapraz doğrulama yönteminde veriler üzerinde bir nokta dışında tüm veriler üzerinde eğitimin gerçekleşmesi ve bu nokta için tahmin yapılması gerçekleşir. Eğitim, veri setindeki her örnek için bir

modelin oluşturulmasını ve değerlendirilmesini gerektirir. Önce k-katlamalı çapraz doğrulamada olduğu gibi hata hesaplanır ve modeli değerlendirmek için kullanılır. Eğitim için mümkün olan en büyük miktarda veri kullanıldığı için verilerden en iyi şekilde yararlanılır. Dezavantajı ise hesaplama açısından maliyeti fazla olmasıdır (Berrar, 2018).

İleri beslemeli yapay sinir ağı mühendislik faaliyetlerinde ve bilimsel çalışmalarda oldukça yer almaktadır. Bu sebeple bu çalışmada ileri beslemeli yapay sinir ağı tercih edilmiştir (Du, He ve Martins, 2021; Khurana, Winarto ve Sinha, 2009; Li, Cai ve Qu, 2019).

Tez çalışması kapsamında vekil ağa ait hiperparametrelerin belirlenmesinde 5 katlamalı çapraz doğrulama yöntemi kullanılmıştır. Ağa ait katman sayısı, nöron sayısı, öğrenme oranı, aktivasyon fonksiyonu ve geri yayılım algoritması gibi hiperparametreler arasından en uygun değerlerin bulunması üzerine araştırma yapılmıştır. Tablo 3.3'te değerlendirilen hiperparametreler gösterilmiştir. Bu değerlerin her biri 5 katlamalı çapraz doğrulama ile mevcut veri seti üzerinde denenmiştir. Toplamda 486 test gerçekleştirilmiştir. Böylece 5 katlamalı çapraz doğrulama ile ayrılmış her parçanın performansları toplanmış ve sonrasında her parça için performans ortalaması alınmıştır. Yapılan bu testler sonucunda en iyi performansa (en az hata oranına) sahip ağ seçilmiş ve Tablo 3.4'te gösterilmiştir.

YSA vekil modelini oluştururken MATLAB R2019a sayısal hesaplama ortamından faydalanılmıştır. Regresyon problemlerinde ileri beslemeli ağlar için varsayılan olan ve sıklıkla kullanılan Levenberg-Marquadt geri yayılım algoritmasının testler sonucunda en uygun olduğu belirlenmiştir ve kullanılmıştır (Du, He ve Martins, 2021).

Gizli katman sayısının fazlalığı, ağın karmaşıklığının artmasına neden olmaktadır. Model, karmaşık olursa eğitim ve test süresi daha uzun olabilmektedir. Bu durumlar göz önünde bulundurularak, çalışmada daha az katman sayısı olan ağ tercih edilmiştir.

Aktivasyon fonksiyonları YSA da bir giriş değerini işleyerek, sonraki katmana giriş olarak besleyen çıkış değerine dönüştürebilmek için kullanılmaktadır. Çeşitli aktivasyon fonksiyonları bulunmaktadır. Bu çalışma kapsamında gerçekleştirilen testler sonucunda Sigmoid aktivasyon fonksiyonu tercih edilmiştir.

Tablo 3.3. Yapay sinir ağının eğitimde değerlendirilen hiperparametreler.

Parametre	Değer
Katman Sayısı	1,2
Nöron Sayısı	2,3,...,9,10
Öğrenme Oranı	0,001, 0,005, 0,01
Aktivasyon Fonksiyonu	sigmoid, ReLU, hiperbolik tanjant
Geri yayılım Algoritması	trainlm, trainbfg, trainbr

Tablo 3.4. Yapay sinir ağında kullanılan hiperparametreler ve değerleri.

Parametre	Değer
Katman sayısı	1
Nöron	10
Öğrenme oranı	0,005
Aktivasyon Fonksiyonu	logsig
Geri yayılım Algoritması	trainlm

3.4.7. Yapay Sinir Ağının Eğitilmesi

Yapay sinir ağında kullanılacak verilerin toplanması ve gerekli hiperparametrelerin belirlenmesinden sonra eğitilecek YSA'nın geliştirilmesine başka bir ifade ile minimum hata oranına yaklaşması hedeflenmektedir. Veri setinin büyüklüğü, eğitim ve test verilerinin dağılımı gibi detaylar öğrenme algoritmasının başarısını önemli oranlarda etkilemektedir. Bu denli önemli olması sebebiyle de veri seti için uygun tekniklerin kullanılması gerekmektedir.

Çapraz doğrulama yöntemi, vekil model kullanılan yöntemler arasında yaygın olarak kullanılmaktadır (Mukesh, Lingadurai ve Selvakumar, 2018). Diğer çapraz doğrulama yöntemlerinden daha avantajlı olduğu düşünüldüğünden bu çalışmada k-katlamalı çapraz doğrulama yöntemi kullanılmıştır. Optimizasyon süreçlerinde kullanılabilmesi için FoilSim web sayfasından Toplanan 8000 adet C_L veriseti ile 3 girişli 1 çıkışlı bir yapay sinir ağı eğitilerek hücum açısı, kamburluk ve kalınlık değeri girildiğinde otomatik olarak C_L değerini

hesaplayan bir yapı haline getirilmiştir. Söz konusu ağ için 8000 adet verinin %80'i (6400) eğitim, %20'si (1600) test için kullanılmış ve 5 katlamalı çapraz doğrulama ile yüksek başarıda eğitimi gerçekleştirilmiştir. Yapay sinir ağının modelini oluşturmak için MATLAB R2019a sayısal hesaplama programı kullanılmıştır. Bu programda ağların oluşturmada nntaintool aracı kullanılmıştır. Yapay sinir ağının eğitilmesi esnasında eğitimin sonlanmasında kullanılan parametrelere ait bilgilere Tablo 3.5'te yer verilmiştir. Ağın eğitiminde belirli bir zaman kısıtlaması kullanmak istenirse zaman parametresi kullanılmaktadır. Bir diğer parametre de iterasyon sayısıdır. Ağın eğitiminde kullanacak sınır iterasyon sayısına ulaşıldığında eğitimi sonlanmaktadır. Ağın eğitim esnasında zaman kısıtı kullanılmamıştır. Sonraki parametre ise ağ, istenen en az gradyan değerine ulaştığında eğitim sonlandırılır. Ağın istenen minimum performansa ulaşması durumunda eğitim sonlanmaktadır.

Tablo 3.5. Yapay Sinir ağı için kullanılan durdurma kriterleri.

Parametre	Değer
İterasyon	1000
Performans	1e-06
Gradyan	1e-07
Zaman	Zaman kısıtı kullanılmamıştır.

Çalışmada yapay sinir ağı kullanmak için yer alan ana motivasyonlar aşağıda sıralanmıştır.

- YSA'nın sınırsız girdi ve çıktı kabul edebilme özelliği, küçük ve büyük veri kümesi analizleri için uygun hale getirir.
- Doğrusal olmayan karmaşık ilişkileri öğrenme ve modelleme becerileri sayesinde YSA farklı alanlardaki çeşitli gerçek yaşam uygulamalarını ele alabilmektedir.
- YSA benzer olaylara daha iyi ulaşmak için olayları öğrenebilir ve yorum yaparak akıllıca kararları sıralayabilmektedir.
- Diğer tahmin tekniği alternatiflerinden farklı olarak YSA'lar verilerin nasıl dağıtılması gerektiği gibi girdi değişkenleri üzerinde herhangi bir kısıtlamaya ihtiyaç duymamaktadır.
- Çoklu işlem yeteneği olan YSA aynı anda birden fazla görevi yerine getirme becerileri sayesinde verimlilik sağlamaktadır.

- YSA genelleme özelliği sayesinde ilk girdi ilişkilerinden öğrendikleri ile bilinmeyen veriler arasındaki ilişkiyi tahmin ederek hiç görmediği girdiler hakkında çıktı üretebilmektedir ve böylece modeli genelleştirmiş olur.

Araştırmalardan elde edilen sonuçlara dayanarak kanat profil optimizasyonu üzerine farklı yaklaşımlar sunulmaktadır. Ancak tez çalışması kapsamında yer alan NASA FoilSim verilerini kullanarak kanat profili için açı, kamburluk ve kalınlık değerlerini girdi olarak belirleyen ve çıktı olarak da kaldırma katsayısını sunan bir çalışmaya rastlanmamıştır.

3.5. Optimizasyon

Optimizasyon (eniyileme), soyut olarak tanımlanırsa; bir kararın verilmesi, somut olarak ifade etmek gerekirse sistem veya sistem parçalarının tasarımının gerçekleştirilmesi süreçlerinde elde edilecek ürünün benzer çözümler arasında mümkün mertebede en iyiye yakın olması için yapılan eylem, takip edilen süreç veya yöntem olarak ifade edilebilir (Pehlivanoğlu, 2013). Yapılan çalışmaları optimizasyon, modelleme (tasarım) ve analiz (çözümleme) temel iki bölüm olarak nitelendirilmektedir. Modelleme gerçek hayatta karşılaştığımız problemlerin matematiksel olarak ifade edilmesi olarak tanımlanır. Analiz ise elde edilen bu modelin sağlanması için gerçekleştirilen en iyi çözüm olarak tanımlanabilir.

3.5.1. Optimizasyon Problemlerinin Temel Parçaları

Optimizasyon problemlerinin sahip olması gereken bazı parçalar bulunmaktadır. Bunlar amaç fonksiyonu, tasarım değişkenleri ve kısıtlamalardır.

Amaç fonksiyonu; tasarım sürecinde ana gaye istenilen fiziksel ve işlevsel özellikleri karşılayan bir tasarım elde etmektir. Aynı amaca yönelik farklı çözümler üretmek mümkündür, ancak hangi çözümün daha ekonomik, etkin ve güvenilir olduğunu tespit etmek nesnel olarak ifade etmeyi gerektirir. Bu kavramı karşılamak için ayırt ediciliği içeren ve nesnel olarak ifade edilebilen yöntem amaç fonksiyonu aracılığıyla sağlanır. Optimizasyon sürecinde amaç fonksiyonu minimize veya maksimize edilmeye çalışılır (Pehlivanoğlu, 2013).

Tasarım değişkenleri; tasarımın yapılması esnasında dikkate alınan ve tasarımdan beklenen işlevleri ve özellikleri ifade eder (Pehlivanoğlu, 2013). Optimizasyon problemimizde ulaşmak istediğimiz en iyi sonuç için bazı sınırlandırmalar gerçekleştirmek

gerekir. Bu sınırlandırmalar eşitlik, eşitsizlik ve geometrik sınırlandırmalar olabilir.

Tasarım kısıtları; optimizasyon sürecinde tasarım değişkenleri tarafından sağlanması gereken koşullar olarak tanımlanabilir. Önerilen bir tasarım kendisinden beklenen tüm işlevsel ve fiziksel özellikleri taşıyor olsa bile tasarım kısıtlarını karşılamadığı müddetçe olası çözüm olarak görülmez (Pehlivanoğlu, 2013).

3.5.2. Optimizasyon Problemlerine ait Sınıflandırmalar

Optimizasyon problemleri farklı ölçütlere göre sınıflandırılabilir. Bu sınıflandırmada optimizasyon problemlerini oluşturan temel parçalardan faydalanılabilir. Standart bir optimizasyon problemleri farklı yönleriyle sınıflandırılabilir, bunlar doğrusallık, değişken sayısı, değişken türü ve sınırlardır (Pehlivanoğlu, 2013).

Doğrusallık, optimizasyon problemlerinde doğrusal problemler, lineer değişkenli amaç fonksiyonlarına sahiptir (Engelbrecht, 2007). Aynı zamanda doğrusal problemlerde tasarım kısıtları da lineer değişkenli olmalıdır. Amaç fonksiyonu ve kısıtlamalar doğrusal değilse bu problem doğrusal olmayan optimizasyon problemi olarak ifade edilir, doğrusal optimizasyon problemlerini çözmek doğrusal olmayan optimizasyon problemlerine göre çok daha kolaydır (Messac, 2015).

Değişken sayısı, optimizasyon problemi tek değişkenden oluşuyorsa, bu durumda problemi tek değişkenli optimizasyon problemi olarak tanımlamak gerekiyor, tasarım değişkenin sayısı birden fazla olması durumunda ise çok değişkenli optimizasyon problemi olarak adlandırılır (Pehlivanoğlu, 2013).

Değişken türü, bazı optimizasyon problemleri için değişkenlerini ayrık bir kümeden seçmek anlamıyla bazı modellerde ise herhangi bir gerçek değeri alması daha anlamlı olmaktadır. Ayrık değişkene sahip modeller ayrık optimizasyon problemleridir. Sürekli değişkene sahip modeller ise sürekli optimizasyon problemlerini ifade eder (Sadrehaghighi, 2021).

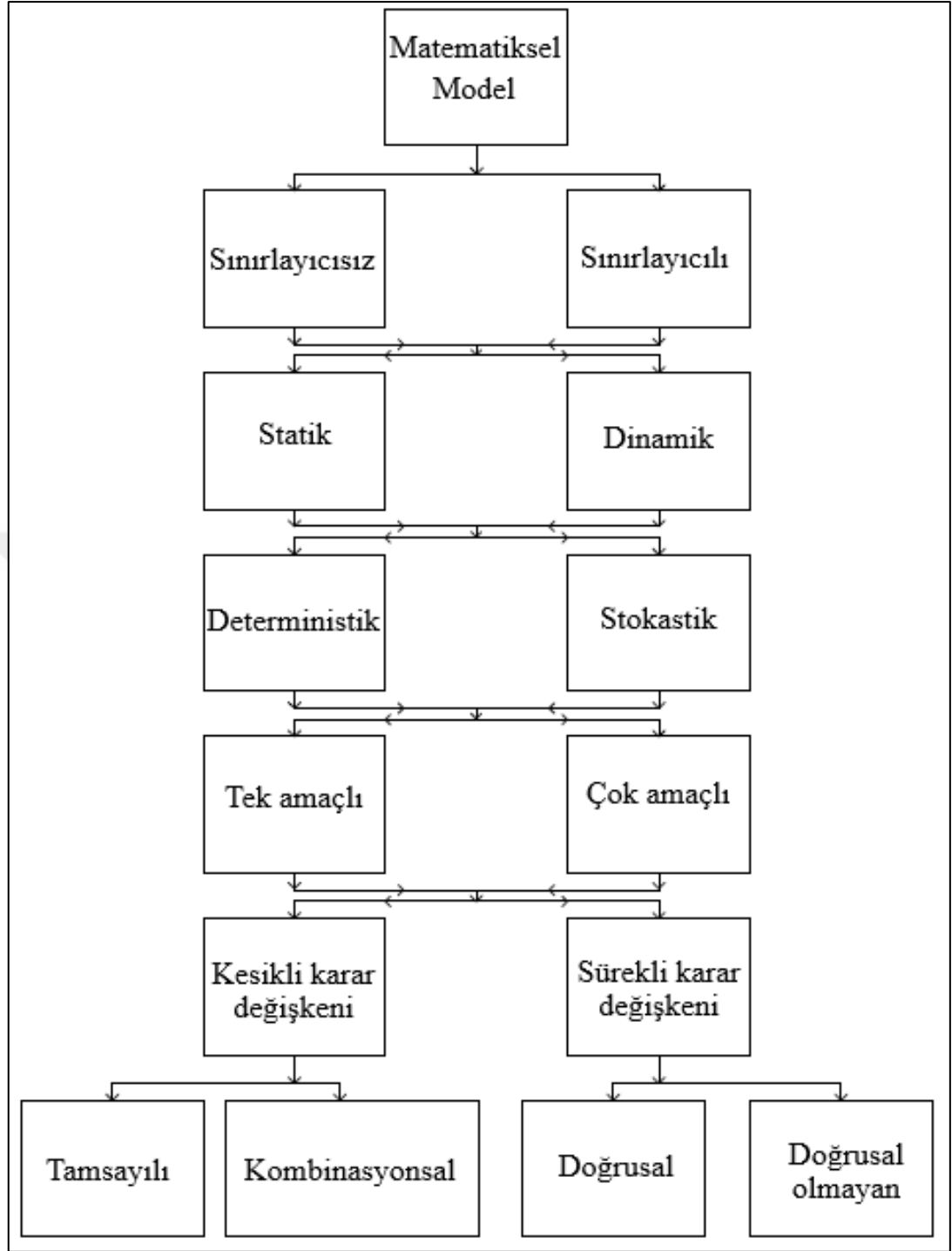
Sınırlar, problemleri sınıflandırma için kullanılabilecek başka bir konu ise değişkenlere yapılan sınırlandırmanın olup olmamasıdır (Engelbrecht, 2007). Eğer optimizasyon problemi içerisinde sadece kenar kısıtları içeriyorsa sınırsız optimizasyon problemi olarak adlandırılır. Sınırlandırılmış optimizasyon problemi değişkenler üzerinde açık sınırlandırmaların olduğu uygulamalarda bulunur (Sadrehaghighi, 2021).

3.5.3. Sezgisel Optimizasyon

İnsanlık tarihinin ilk dönemlerinde problem çözmeye yönelik temel yaklaşım yoluyla sezgisel olarak gerçekleşmiştir (Yang, 2010). Sezgisel algoritmalar, istenilen hedefi gerçekleştirmek için doğal olaylardan yola çıkılarak üretilmişlerdir. Sezgisel algoritmalar optimal çözümü yakınsama özelliğine sahiptirler ancak kesin çözümü garanti edememektedirler (Karaboğa, 2011). Sezgisel optimizasyon algoritmalarının başarısının birçok nedeni vardır, bu nedenlerin başında biyolojik sistemler fiziksel ve kimyasal süreç olmak üzere doğadaki en başarılı süreçlerden ilham alarak gerçekleştiriliyor olması gelmektedir (Yang, 2010). Genel amaçlı sezgisel yöntemler, biyolojik tabanlı, fiziki tabanlı, sosyal tabanlı olmak üzere üç farklı gruba ayrılmaktadır. Bu yöntemlere ek olarak karma yöntemler de mevcuttur.

Biyolojik tabanlı olanları, genetik algoritma (GA), diferansiyel gelişim algoritması, karınca koloni algoritması, yapay sinir ağları, arı kolonisi algoritmaları ve yapay bağışıklık sistemi algoritmaları ve modelleridir. Isıl işlem ve elektromanyetizma algoritması fizik tabanlı ve tabu arama sosyal tabanlı algoritmalarıdır. Parçacık sürü optimizasyonu (PSO), hem biyolojik hem de sosyal tabanlı bir algoritmadır. Kültürel algoritma ise hem biyolojik hem de sosyal algoritma grubundadır. Optimizasyon algoritmaları tek bir çözümle başlayıp sonrasında operatörlerle devam ediyorsa bu yöntemlere tek noktalı yöntemler denir, tabu arama ve ısıtma işlem gibi yerel arama algoritmaları bu gruba girmektedir. Bir popülasyon üzerinden başka bir ifadeyle çok noktadan çözüme başlanıp bu farklı noktalarla optimizasyon yapılıyorsa bu yöntemler çok noktalı ya da popülasyon tabanlı yöntemler olarak ifade edilir. PSO, GA, karınca kolonisi algoritması, arı kolonisi algoritması çok noktalı tabanlı algoritmaya örnek verilebilir (Alataş, 2007).

Sezgisel algoritmalar, genel olarak tek komşuluk yapısında sınıflandırılır. Bazıları ise arama işlemini düzenli bir şekilde değiştirir, birden çok çözüm arama seçenekleri ile farklı çözüm arama yöntemlerine ulaşmaya çalışır ve değişken komşuluk yapılı şeklinde sınıflandırılabilir. Algoritmalar çalışırken daha önceki durumlar ya da en iyi durumlar hatırlanıyorsa hafızalı, hatırlanmıyorsa hafızasız şeklinde sınıflandırılabilir. Örnek olarak PSO ve tabu arama algoritmaları hafızalı, GA hafızasızdır (Alataş, 2007).



Şekil 3.13. Sezgisel yöntemler (Alataş, 2007).

3.5.4. Sürü Tabanlı Sezgisel Algoritma Çeşitleri

Sürü, birbirleriyle iletişim halinde olan dağınık yapılı bireyler topluluğu olarak ifade edilir. Bahsi geçen bu bireyler insan, karınca, arı vb. olarak tanımlanabilir. Sürülerde N adet temsilci hedefledikleri amaca yönelik davranışı sergilemek ve hedefe ulaşmak için birlikte çalışmaktadırlar (Akyol ve Alataş, 2012). Sürü zekası modelleri doğal sürü sistemlerinden

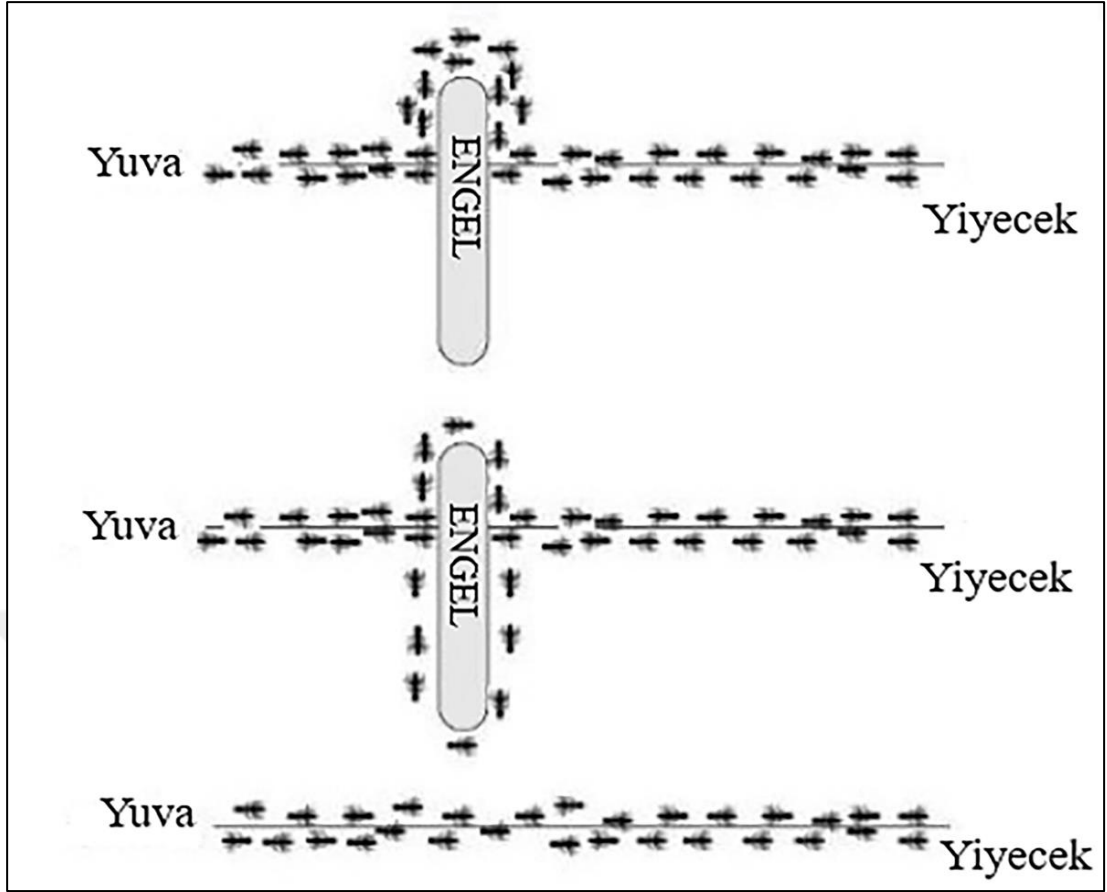
esinlenilerek oluşturulmuş hesaplama modelleri olarak tanımlanabilmektedir (Ahmed ve Glasgow, 2012).

Sürü zekası temelinde yuva inşa etme, yiyecek arama, böcek kolonilerindeki öğelerin sınıflandırılması gibi doğadaki canlıların toplu davranışlarının incelenmesi ve araştırılması ile ilgili bir disiplin olarak kabul edilir. Konu mühendislik sistemi olarak yorumlandığında çevreleri ile iletişime geçen birimlerin çeşitli eylemlerin sonuçları nedeniyle daha iyi davranışlar sergileyen dağıtık sistemlerin tasarımı olarak kabul edilebilir. Sürü zekası, farklı araştırmacılar tarafından problem çözme, hesaplama ve çözümleri optimize etmek için uygulanan önemli uygulama alanlarından olduğu kabul edilmektedir (Nayyar, Puri ve Suseendran, 2019).

Sürü zekası algoritmaları, çeşitli sorunları çözmek için kullanılır. Bunlardan en klasik olarak bilineni genetik algoritmadır. Sonrasında zaman geçtikçe çeşitli araştırmacılar tarafından karınca kolonisi algoritması, kedi sürüsü algoritması gibi birçok çeşitli sürü zekası algoritmaları önerilmiştir (Nayyar, Puri ve Suseendran, 2019).

3.5.4.1. Karınca Koloni Algoritması

Karınca kolonisi algoritması (KKA), karınca kolonilerinin yiyecek arama davranışlarından esinlenilerek oluşturulmuş bir algoritmadır (Blum ve Li, 2008; Dorigo, Maniezzo ve Coloni, 1996). Karıncalar en kısa yol problemini çözmek konusunda oldukça başarılı olan hayvanlardır. Yuvaları ve besin kaynakları arasındaki aynı zamanda tersine besin kaynağı ve koloninin yuvası arasındaki en kısa yolu bulan canlılardır. Karınca kolonileri birbirleri ile etkileşim halinde olan, görevlerini paylaşan ve iş birliği içinde çalışan sürülerdir.



Şekil 3.14. Karıncaların yiyecek arama davranışı (Sadriia, Soltani, Zulkifli, Ismail ve Ariffin, 2014).

Özetle KKA, çözülmesi kolay olmayan problemlere çözüm getirmek için kullanılan algoritmalarından biridir (Karaboğa, 2011).

3.5.4.2. Parçacık Sürü Optimizasyonu

Parçacık Sürü Optimizasyonu (PSO) kuş ve balık sürülerinin hareketlerinden esinlenilerek nümerik problemlere en optimum çözümü elde edebilmek amacıyla ortaya atılmıştır. PSO çok parametrelili ve çok değişkenli problemlere çözüm üretmek amacıyla geliştirilmiştir (Kennedy ve Eberhart, 1995).

PSO algoritmasında her bir kuş bir parçacık ve bu kuşlardan oluşan topluluk da sürü olarak adlandırılır. PSO’da her parçacığın, spesifik bir optimizasyon problemine çözüm olduğu varsayılır, eksik var sanki burada konum ve hız vektörlerinden oluşur. Konum vektörü, problemdeki değişkenlerden her birinin değerini içerir. Problemin iki parametresi olduğunu varsayarsak parçacıkların da iki boyutlu konum vektörü olacaktır (Mirjalili, Song Dong, Lewis ve Sadiq, 2020).

3.5.4.3. Yapay Balık Sürüsü Algoritması

Yapay Balık Sürüsü Algoritması (YBSA) ilk kez 2002’de gündeme gelmiştir. YBSA, temeli sürü kavramına ve balık sürüsünün doğasındaki sosyal davranışlarını simüle eden bir sisteme dayanmaktadır. Doğada balıklar, diğer balıkları tek tek araştırarak veya takip ederek daha besleyici olan bölgeyi bulabilirler. Sonuç olarak, en fazla balığın bulunduğu su alanı, genellikle en fazla beslenme alanıdır (Cheng, Jiang ve Yuan, 2009).

YBSA algoritmasında temel konu küresel optimuma ulaşmak amacıyla toplanma ve izleme gibi balık davranışlarının gözlemlenip taklit edilmesini içerir. Başlıca çözüm uzayı, yapay balığın içinde yaşadığı ortam olarak tanımlanabilir. Bu ortam diğer balıkların da içinde bulunduğu bir alandır.

YBSA genetik algorithmada kullanılan mutasyon adımları ve çaprazlama gibi özelliklere sahip değildir. Bu sebeple YBSA algoritmasının süreçleri daha kolay yönetilebilmektedir. Algoritma başlangıçta rastgele olarak üretilen olası çözümlerle başlar sonrasında tekrarlı olarak optimum bir çözüm elde etmeyi hedefleyen arama yapar (Zhang, Shao, Li, Gan ve Sun, 2006).

3.5.4.4. Gri Kurt Optimizasyonu Algoritması

Gri Kurt Algoritması (GKA), kurtlara ait avlanma ve sosyal davranışları simüle etmek için önerilmiş biyoloji tabanlı meta sezgisel sürü algoritmasıdır. GKA’nda optimizasyon araması rastgele olarak bir kurt popülasyonu oluşturarak başlar. Bu popülasyon avın bulunduğu yeri tahmin etmek için avlanma süresince sürekli olarak güncellenir. GKA için av, optimum çözüm olarak kabul edilmektedir (Abd Elaziz vd., 2022).

Toplumsal sınıf olarak kurtlar, alfa, beta, delta, omega olarak sınıflandırılır. Alfa grubu, kararları alan kurt grubuna hakim bir türdür. Beta sınıfı, kurt grubuna ait kararlarda alfa sınıfına yardımcı roldedir. Omega sınıfı ise en gerideki kurtları ifade eder. Eğer bir kurt belirtilen bu sınıflara ait değilse delta olarak isimlendirilir. GKA’nın temel adımları avı çevreleme, avlanma ve avına saldırma aşamalarından oluşmaktadır (Mirjalili, Mirjalili ve Lewis, 2014).

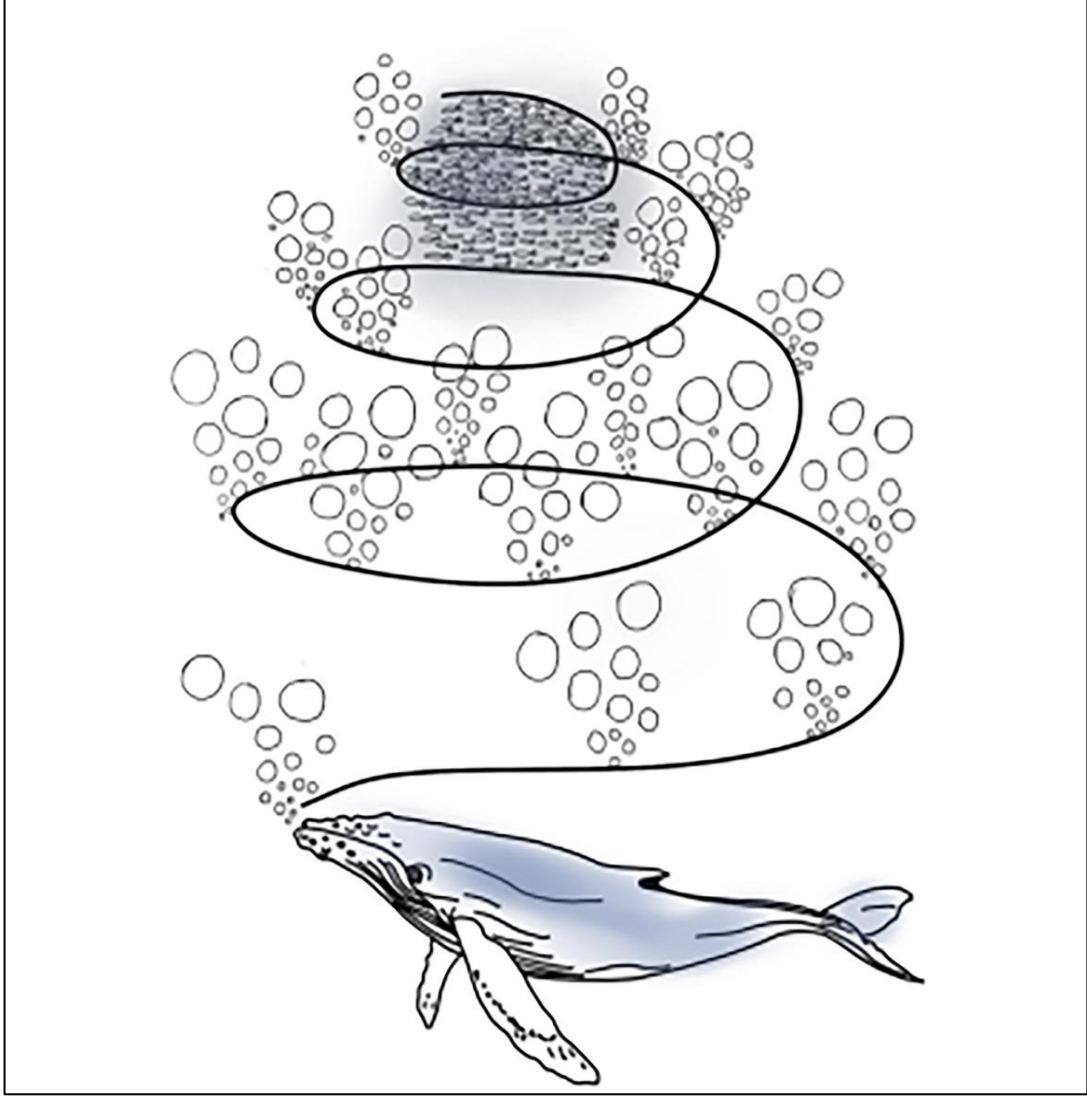
3.5.4.5. Balina Optimizasyon Algoritması

Balinalar, en büyük memeli hayvan olarak kabul edilmektedir. Yetişkin bir balina 30 metre uzunluğu ve 180 ton ağırlığa erişebilmektedir. Bu devasa memelinin katil, kuzey,

minke, kambur, mavi, buzul ve oluklu gibi yedi temel çeşidi bulunmaktadır. Balinalar genel olarak yırtıcı olarak kabul edilmektedirler (Abd Elaziz vd., 2022).

Balina Optimizasyon Algoritması (BOA), balina gruplarının yırtıcı davranışlarını simüle eden sezgisel bir algoritmadır. Balina optimizasyon algoritmasının basit bir çalışma prensibi vardır. Ek olarak uygulanması kolaydır ve birkaç parametresi vardır. Kambur balinalar genellikle okyanustaki balık ve karidesleri kabarcık-ağı yırtıcı yöntemi ile avlarlar. İlk olarak balinalar rastgele olarak av ararlar ve diğer balinalardan uygun bir mesafeyi koruyarak daha iyi av bulurlar (Huang vd., 2022).

Balinalar suyun altında soluk vererek hava kabarcığı bulutları ve sütunları meydana getirirler. Bu kabarcıklar büyükçe bir alana yayılır ve birbirleri ile bağlantılı kabarcıklar avları belli bir bölgede toplamış olur. Sonrasında balina, kabarcıklarının içinden yukarı doğru yükselir. O esnada nefes alıp vermeye devam edip kabarcık oluşturmaya da devam eder.



Şekil 3.15. Balinaların kabarcık stratejisi ile avlanma yöntemleri (Dökeroğlu, Sevinç, Küçükyılmaz ve Coşar, 2019).

Avına yaklaştıkça avlanacağı çember sürekli daralır. BOA’da balinalar avlarının konumu tahmin edebilir, bu sebeple de avlarının çevresini hava kabarcıklarıyla sararlar. BOA’da balinanın avı erişilebilen en iyi çözüm olduğundan, optimum sonuç olarak kabul edilir. Arama ajanlarından en iyisi belirlendikten sonra başka arama ajanlarının konumu en iyi arama ajanından faydalanılarak güncellenir. Balinaların ava doğru hareket etme davranışı avın etrafında bulunan çemberi daraltma ve spiral hareket olmak üzere iki kısımda tasarlanmıştır (Huang vd. 2022).

3.5.4.6. Guguk Kuşu Algoritması

Guguk kuşlarını diğer kuşlardan ayıran ve aynı zamanda optimizasyon problemlerinde çözüm olmasının sebebi sahip oldukları saldırgan çoğalma sistemidir. Xin -

She Yang Suash Deb tarafından geliştirilen Guguk Kuşu algoritmasının temeli bazı Guguk kuşlarının Levy uçuşu denilen uçuşa sahip olmaları aynı zamanda kuluçka parazitliği denilen özelliklerine dayanmaktadır (Liu, Wang ve Luo, 2020; Yang ve Deb, 2009). Guguk kuşları yumurtalarını başka kuşların yuvasına bırakmaktadırlar. Guguk kuşlarından bazıları kendine ait yumurtayı bırakacağı yuvada yumurta varsa o yumurtayı alıp kendisinininkini bırakır, böylece amacı yavrusunu bir an önce yumurtadan çıkmasını sağlamaktır. Eğer yuvanın sahibi Guguk kuşunun bıraktığı yumurtayı farketmezse kendine ait sanar ve üzerine oturur böylece kuluçka parazitliği gerçekleşir (Payne, 1977).

3.5.4.7. Yapay Arı Kolonisi Algoritması

Yapay Arı kolonisi algoritması bal arısı sürülerinin kendilerine özgü zeki davranışlarını örnek alarak arıların besin ararken kullandıkları yöntemlerden esinlenilerek oluşturulmuş bir optimizasyon algoritmasıdır. Sürü zekasına dayanan bu algoritma doğada sürü halinde hareket eden arıların besin bulmada sergilemiş oldukları davranışları temel olarak optimizasyon problemlerini çözmek için kullanılmaktadır (Ab Wahab, Nefti-Meziani ve Atyabi, 2015). Bu algoritma 2007 yılında Karaboğa ve Baştürk tarafından tasarlanmış ve sürekli olarak geliştirilmiştir. YAK kolonisi üç farklı türü barındırmaktadır. Bunlar işçi arı, gözücü arı ve kaşif arıdır ve bu arıların yararlandığı besin kaynağı problemin olası çözümünü temsil etmektedir (Gabor, Simon, Korenschi ve Botez, 2016).

Yiyecek arama davranışlarına odaklanıp bu faaliyetlerinin modellenmesiyle başarılı algoritmalar ortaya çıkmıştır. Bunlardan en önemlilerinden biri de Karaboğa tarafından geliştirilen Yapay Arı Kolonisi algoritmasıdır (Kaya ve Eke, 2020).

Gerçek arıların yem bulma davranışları

Arı kolonileri için yaşamlarının devamını sağlayan en önemli faktör besin kaynağıdır. Kovan içinde biriktirilen kaynaklar ve ortamda bulunabilecek yem kaynakları ve arıların etkileşimleri bu süreçteki önemli etkenlerdendir. Bu arı kovanında yapılacak görevler, o görev için belirlenmiş arılar tarafından yapılır. Bu da arılar arasında görev dağılımı olduğunu ifade eder. Arı kovandan ayrılır ve yiyecek araştırması rastgele olarak devam eder. Bir yiyecek kaynağı bulunur ve bulunan kaynakta yiyecek azalmasıyla birlikte arılar yeni besin aramaya başlar ya da diğer arılardan aldığı bilgiye göre başka kaynaklara yönelmeye başlarlar (Karaboğa, 2017).

Tereshko, reaktif difüzyon denklemlerinin ışığında önerdiği kolektif zekanın ortaya çıkmasına yol açan minimal besin arama modeliyle üç temel ögeye sahip bir yiyecek modeli

öngürmüştür. Bunlar yiyecek kaynakları, görevi belirli işçi arılar, görevi belirsiz işçi arılardır. Akay çalışmasında benzeri bir gruplandırma kullanmış ve çalışmasında da bunu açıklamıştır (Akay, 2009). Bileşenler aşağıdaki gibi açıklanabilir:

Yiyecek kaynakları, arıların besin bulmak için gittiği kaynaklardır. Bir yiyecek kaynağının değeri basite indirgemek adına zenginliği olarak tanımlanmıştır. Bu değer kaynağın çeşidi, yuvaya uzaklığı, nektar miktarı gibi faktörlere bağlanabilir.

Görevi belirli işçi arılar, daha önceden belirlenen kaynaktan toplanan yiyeceğin kovana getirilmesi ile ilgilenmektedir. İşçi arılar aynı zamanda gittikleri kaynakların kalitesi ve yeri ile ilgili bilgileri kovanda bekleyen diğer arılar ile paylaşırlar.

Görevi belirli olmayan işçi arılar, besin kaynaklarını aramada görevlidirler. Görevi belli olmayan arılar; kaşif arılar ve gözcü arılardır. Kaşif arı rastgele kaynak arar. Gözcü arı ise kovanda bekler, görevli arıları izler ve bu arılardan gelen bilgiler ile yeni kaynaklara yönelir (Kaya ve Eke, 2020).

Arılar, yiyecek kaynağının kalitesi ve yeri ile ilgili bilgiyi dans ile dans alanı denilen bölgede paylaşmaktadırlar. Arılardan biri dans ederken diğeri anteni ile ona dokunur ve ondan bulduğu kaynağın tadı ve kokusu ile ilgili bilgi alır. Kovan ile aradaki uzaklık, besinin durumu hava şartları gibi dansı etkileyen faktörler bulunmaktadır. Ziyaret ettikleri kaynaklara daha fazla arı yönlendirebilmek için kovandaki çeşitli alanlarda bu dansı gerçekleştirir ve sonrasında kaynağına geri döner (Kaya ve Eke, 2020).

Kaynağın kovana olan mesafesine göre dairesel kuyruk ve titreme dansı vardır. Titreme dansı nektar miktarı ve besini getirebilme kabiliyeti arasındaki dengeyi sağlamaktadır. Bu dans 50-100 metre uzaklıktaki kaynağa ait yön ve uzaklık verilmeden sağlanan dans biçimidir. 10 metreden 100 kilometreye kadar geniş bir alan içerisinde yineleme sayısının uzaklık bilgisini verdiği ve sekiz rakamına benzeyen bir dans yapılarak güneş ve besin kaynağı arasındaki açının 45° olduğu anlaşılmaktadır (Tereshko, 2005).

Arıların kendiliğinden örgütlenebilme yeteneklerinin dayandığı bazı özellikler aşağıdaki gibi tanımlanabilir.

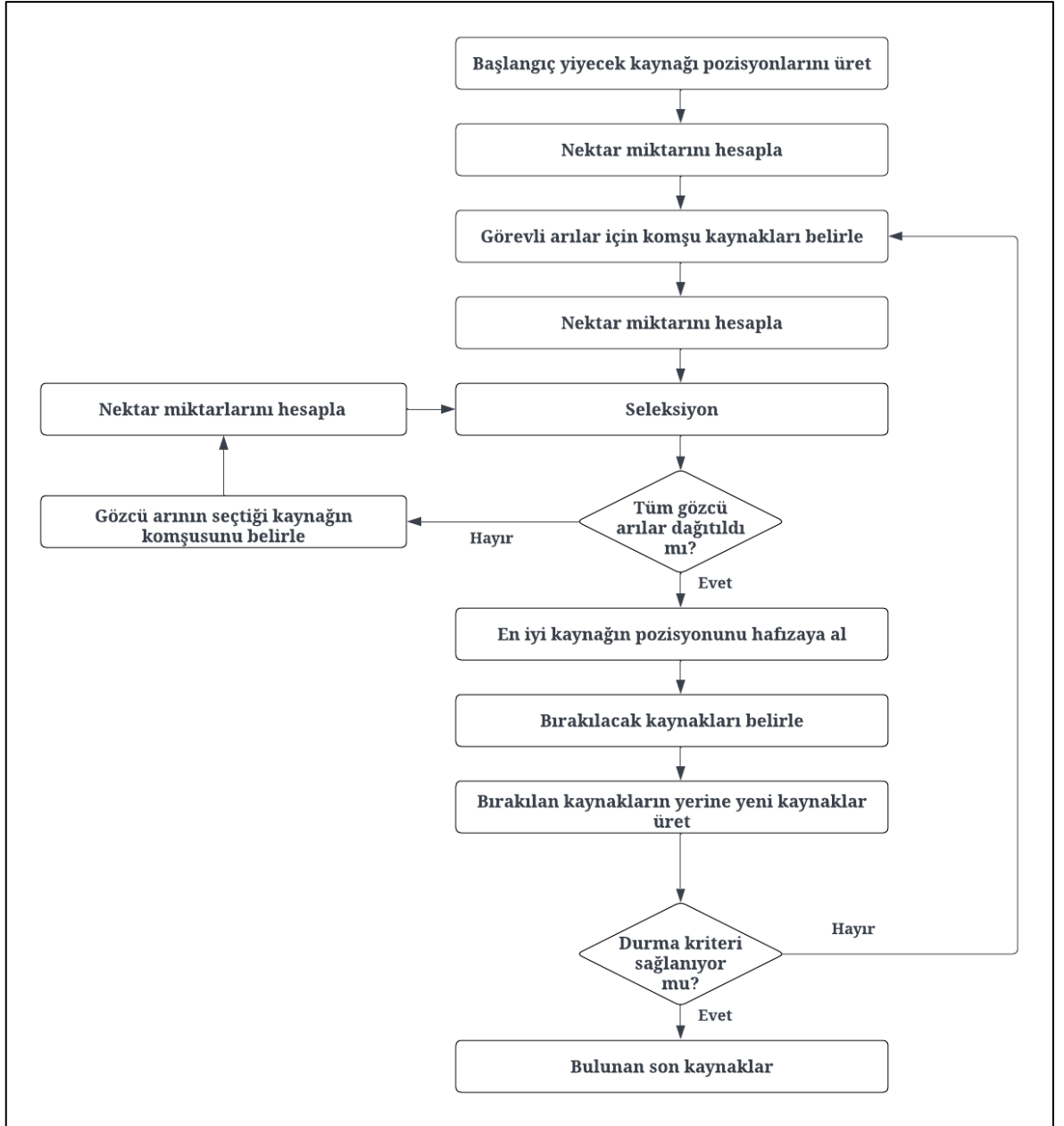
- Pozitif Geri Besleme: Besin kaynaklarının artması ile birlikte daha çok gözcü arı besin kaynaklarını ziyaret etmektedir.
- Negatif Geri Besleme: Besin kaynağı kullanımı arılar tarafından durdurulur.
- Salınımlar: Gözcü arılar çevredeki besin kaynaklarını bulabilmek için arama yaparlar.

- Çoklu etkileşimler: İşçi arılar besin kaynaklarına ait bilgileri, gözcü arılarla besinleri kullanmak ve çıkarmak için dans bölgesinde paylaşır (Nayyar, Puri ve Suseendran, 2019).

Arı kolonilerinin besin arama sürecini incelenip bunu model alan Yapay Arı Kolonisi Algoritmasında bazı kabuller yapılmıştır. Görevli arılar ile toplam yiyecek kaynağı sayısı eşit miktardadır. Bir de işçi arı ile gözcü arıların sayısı birbirine eşittir. Ek olarak bir kaynağın nektarı tükenmiş ise bu kaynaktan görevli arı, kaşif arı olarak görevine devam etmektedir. Kaynaklara ait nektar miktarı sonuçların kalitesini, yiyecek kaynaklarının yerleri optimizasyona ait olasılık sonuçlarını ifade etmektedir (Karaboğa ve Akay, 2009).

YAK Algoritmasında kaşif arı, işçi arı ve gözcü arı olmak üzere üç farklı türde arı vardır. Her bir yiyecek kaynağı için bir işçi arı vardır. Kaynakların tükenmesi durumunda işçi arılar kaşif arılara dönüşmekte ve kaşif arılar yeni kaynak arama yoluna gitmektedir. Böylece kaynakların tükenmesi durumunda işçi arıların o kaynakları terk etmesi negatif geri besleme ve kaşif arıların yeni kaynaklar araması rasgelelik- salınımı olarak tanımlanır. Ortaya çıkan kaşif arı sayısı 'limit parametresi' ile belirlenmektedir. YAK Algoritmasında bir yiyecek kaynağının pozisyonu bir çözüme denk gelmektedir. Yiyecek kaynağının nektar açısından zenginliği çözümün amaç fonksiyonu aracılığıyla tanımlanır. İşçi arılar yiyecek kaynaklarını araştırır ve bu kaynaklarla ilgili bilgileri gözcü arılarla paylaşırlar. Gözcü arılar da işçi arılardan aldıkları bilgilerle zengin yiyecek kaynaklarına doğru daha fazla yönelme eğilimi gösterirler.

Böylece gözcü arılar pozitif geri besleme özelliği gösterirler. YAK algoritması ilk olarak yiyecek kaynaklarına uygun pozisyonların üretilmesiyle işleme başlar (Öztürk, Hançer ve Karaboğa, 2014).



Şekil 3.16. YAK algoritması teorik akış diyagramı (Akay, 2009).

Yapay Arı Kolonisi Algoritması temel adımları aşağıdaki şekilde sıralanabilir:

- Yiyecek kaynağı alanlarının oluşturulması,
- Tekrar,
 - İşçi arıların yiyecek kaynaklarına gönderilmesi ve nektar miktarı hesaplanması,
 - Olasılık değerlerinin görevli arılardan gelen bilgi paylaşımına göre hesaplanması,
 - Gözcü arıların olasılık değerlerine göre yiyecek kaynağı alanları seçmeleri,

- Kaynağı bırakma kriteri: Limit ve kaşif arı üretme sayısı,
- Kontrol: Maksimum çevrim.

YAK Algoritmasının adımlarını detaylandırılması gerekirse aşağıdaki gibi sıralanabilir.

3.5.4.7.1. YAK Algoritmasının Parametrelerinin Başlatılması : Adım 1

Temel olarak bir optimizasyon probleminin çözümü için YAK Algoritmasında başlatılması gereken üç parametre vardır. SN, popülasyondaki çözümü temsil eden yiyecek kaynağı sayısıdır. MaxCycle, yiyecek arama için maksimum döngü sayısıdır ve *limit*, denemelerle iyileştirilemeyen bir besin kaynağının terk edilmesinden sorumlu değerdir ve temel olarak aramayı çeşitlendirmek için kullanılır.

3.5.4.7.2. Başlangıç Besin Kaynağı Pozisyonlarının Üretilmesi: Adım 2

Burada algoritma rastgele yiyecek kaynağı üreterek çalışmaya başlamaktadır. Bu süreçte her bir parametrenin alt ve üst sınırları arasında rastgele değerler üreterek gerçekleşir. Değer olarak $i=1 \dots SN, j=1 \dots M$, SN yiyecek kaynağı sayısı, M optimize edilecek parametre sayısıdır, x_j^{min} , j.parametrenin alt sınırı, x_j^{max} ise j. parametrenin üst sınırıdır. Ek olarak başlangıç aşamasında her kaynağa ait geliştirilememe sayısını ifade eden *failure_i* değeri bulunmaktadır ve sıfır ve bir arasında rastgele sayı üreten rand(0,1) değeri kullanılmaktadır (Akay, 2009).

$$x_{ij} = x_j^{min} + rand(0,1)(x_j^{max} - x_j^{min}) \quad (3.2)$$

3.5.4.7.3. İşçi Arı Fazı : Adım 3

Her bir kaynağın bir görevli arısı vardır. İşçi arı çalıştığı yiyecek kaynağı komşuluğunda yeni bir yiyecek kaynağı belirler ve bu kaynağın kalitesini değerlendirir. Yeni kaynak daha değerli ise yeni kaynağı hafızasına alır. Yeni kaynağın belirlenmesi Eşitlik 3.3 vasıtasıyla sağlanmış olur. $\emptyset_{ij}$, [-1, 1] aralığında rastgele oluşturulmuş bir sayıdır. x_{kj} ($k \neq i, k \notin \{1,2,\dots, SN\}$) rastgele seçilen bir çözümdür.

$$v_{ij} = x_{ij} + \emptyset_{ij}(x_{ij} - x_{kj}) \quad (3.3)$$

Bu işlem ile üretilen v_{ij} değerinin daha önceden belirtilmiş olan alt-üst sınırları aşması durumunda Eşitlik 3.4 kullanılarak j. parametreye ait olan alt ve üst sınır değerlerine ötelenir (Akay, 2009).

$$v_{ij} = \begin{cases} x_j^{min}, & v_{ij} < x_j^{min} \\ v_{ij}, & v_j^{min} \leq v_{ij} \leq x_j^{max} \\ x_j^{max}, & v_{ij} > x_j^{max} \end{cases} \quad (3.4)$$

Sınırlar aralığında üretilen v_i parametre vektörü yeni bir kaynağı temsil eder. Bu kaynağa ait bir maliyet değerini $f(v_i)$ yerine koyarak çözümün uygunluk değeri hesaplaması yapılır.

$$fitness_i = \begin{cases} 1/(1 + f_i), & f_i \geq 0 \\ 1 + abs(f_i), & f_i < 0 \end{cases} \quad (3.5)$$

x_i ile v_i arasında $fitness_i$ değerine göre bir seçim işlemi uygulanır. Yeni v_i çözümü daha iyi ise bununla görevli arı hafızasından eski kaynağın yerini siler ve v_i kaynağının yerini hafızaya atar. Eğer bu olmasaydı bununla görevli arı x_i kaynağına gitmeye devam edecekti ve çözüm geliştirilemediği için x_i kaynağı ile ilgili geliştirilememe sayacı ($failure_i$) bir artacaktır, geliştirildiğinde ise geliştirilememe sayacı sıfırlanacaktır (Akay, 2009).

3.5.4.7.4. Olasılık Değerlerinin Hesaplanması: Adım 4

Gözcü arıların seçim işlemi yaparken kullanacakları uygunluk değerlerine dayalı olasılık değerlerinin hesaplanması Eşitlik 3.6 ile yapılır. Yani bir kaynağın uygunluk değerinin tüm kaynaklara ait uygunluk değerlerinin toplamına oranı, o kaynağın diğer kaynaklara göre nisbi seçilme olasılığını verir.

$$p_i = \frac{fitness_i}{\sum_{i=1}^{SN} fitness_i} \quad (3.6)$$

Olasılıksal seçim işlemi algoritmada nektar miktarlarına göre uygunluk değerlerinin hesaplanması ile gerçekleşir. Uygunluk değerine göre seçim işlemi rulet tekerleği, rastlantısal, örnekleme, sırlamaya dayalı, turnuva ya da diğer seçim yöntemlerinden herhangi biri ile yapılabilir. YAK algoritmasında bu seçim işlemi rulet tekerleği yöntemi ile gerçekleştirilir. Bu yöntemde tekerlekteki her bir dilime ait açının uygunluk değerinin tüm dilimlerin açısının uygunluk değerleri toplamına oranı kullanılır (Akay, 2009).

3.5.4.7.5. Gözcü Arı Fazı: Adım 5

Görevli arılar bir iterasyon sonrası araştırmalarını bitirmeleriyle birlikte tekrardan kovanlarına geri dönüp önceki kaynaklarının kalitesi ile ilgili gözcü arılara bilgi aktarırlar. Gözcü arılarda kendilerine dans ile verilen bilgiler ışığında besin kaynağının kalitesi ile orantılı olarak bir kaynak seçer.

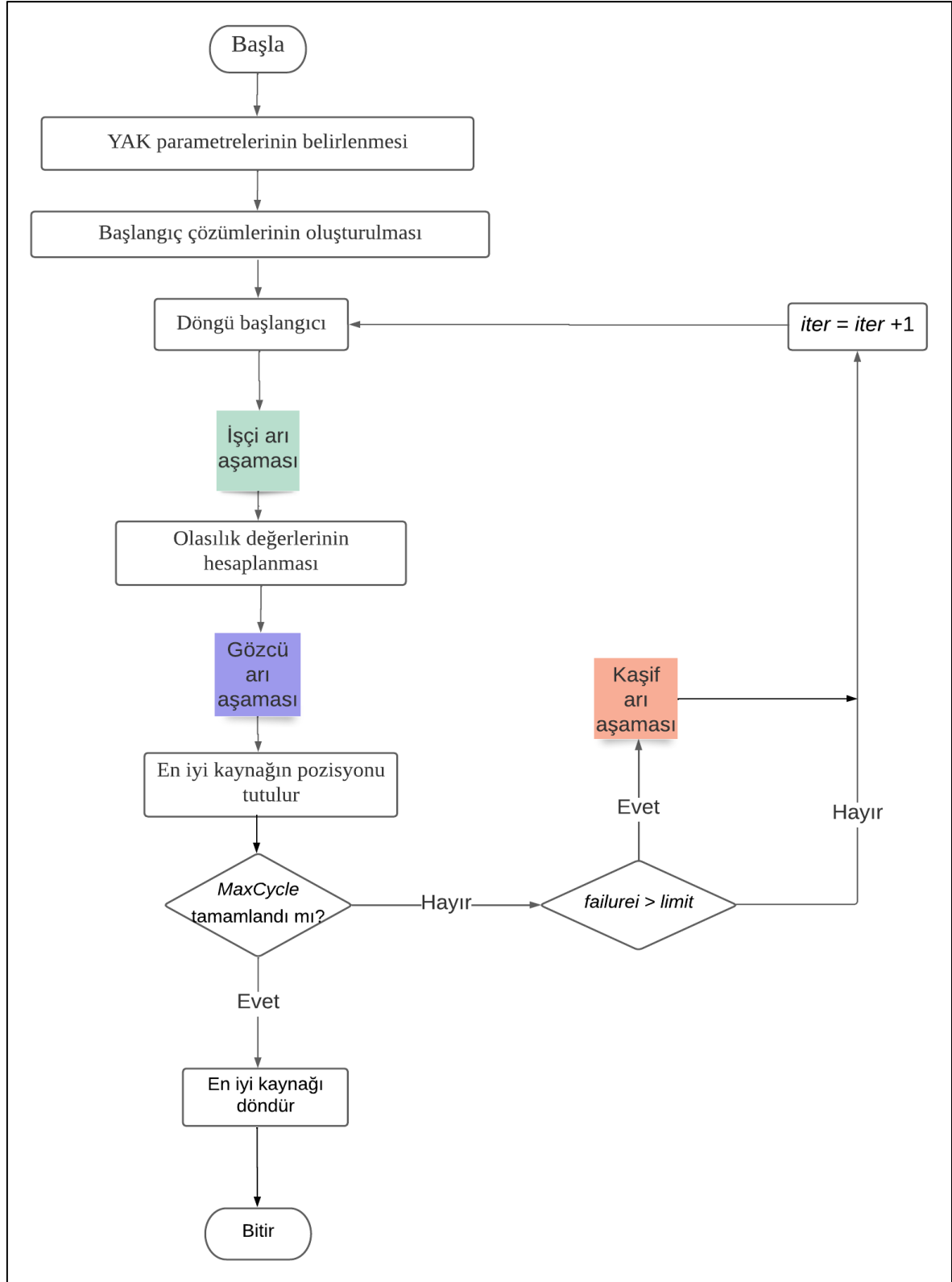
Olasılık değerleri hesaplandıktan sonra rulet tekerleğine göre seçim işlemi sürecinde her bir kaynak için $[0,1]$ aralığında rastgele bir sayı üretilir(*random*). Sonrasında p_i değeri, rastgele üretilen bu sayıdan büyükse görevli arılar gibi gözcü arılarda Eşitlik 3.3'ü kullanarak bu kaynak bölgesinde yeni bir çözüm üretir. Bulunan yeni çözüm değerlendirilir ve kalitesi hesaplanır. Daha sonra yeni çözüm ile eski çözüme ait uygunluklar karşılaştırılır ve iyi olanın seçildiği bir seleksiyon işlemine tabi tutulur. Bulunan yeni çözüm daha iyi ise eski çözüm yerine bu çözüm alınır ve çözüm geliştirilememe sayacı $hata_i$ sıfırlanır, bu işlem tüm gözcü arılar besin kaynakları bölgelerine gönderilene kadar devam eder (Akay, 2009).

3.5.4.7.6. Kaşif Arı Fazı: Adım 6

Bir iterasyonun tamamlanmasının ardından tüm görevli ve gözcü arılar arama işlemlerini bittirdikten sonra çözüm geliştirilememe sayacı (*failure_i*) kontrol edilmektedir. Bir arının gidip geldiği besin kaynağındaki nektarın bitip bitmediği çözüm geliştirilememe sayaçları tarafından belirlenebilir. Her bir kaynak için olan çözüm geliştirilememe sayacı belirli bir eşik değerin üzerinde ise artık bu kaynak için ihtiyaç duyulan görevli arının bitmiş olduğu anlamına gelmektedir ve başka bir çözüm bulması gerekmektedir. Bu da kaynağın tükenmesinden sorumlu görevli arının kaşif arılara dönüşmesi anlamına gelmektedir. Kaşif arıya dönüşmesiyle birlikte Eşitlik 3.2 den yararlanılarak bu arı için rastgele çözüm arama aşaması başlamış demektir. Sonrasında en iyi çözüm hafızada tutulur. Kaynağın terk edilme olayının belirlenmesi için kullanılan eşik değeri YAK Algoritması'nın önemli bir aşamasıdır ve *limit* olarak adlandırılır. YAK da her iterasyonda tek bir kaşif arının gezinmesine müsadde edilmektedir (Akay, 2009).

3.5.4.7.7. Sonlandırma Koşulları: Adım 7

MaxCycle değeri tarafından belirlenen durma koşulu elde edilene kadar 3. ve 6. adımlar arasını tekrarlayın.



Şekil 3.17. YAK algoritması uygulama diyagramı.

Yukarıda adımları anlatılan YAK Algoritmasına ait sözde kod aşağıdaki gibi özetlenebilir.

Adım 1: YAK Algoritmasının parametrelerinin başlatılması

- 1: YAK parametrelerinin hazırlanması (SN, Maxcycle, *limit*).
- 2: Amaç(maliyet) fonksiyonun tanımlanması

Adım 2: Başlangıç besin kaynağı pozisyonlarının üretilmesi

- 1: **for** $i = 1$ to SN **do**
- 2: **for** $j = 1$ to D **do**
- 3: $x_{ij} = x_j^{min} + rand(0,1) * (x_j^{max} - x_j^{min})$
- 4: **end for**
- 5: $f_i = f(x_i)$
- 6: **end for**

Adım 3 : İşçi Arı Fazı

- 1: **for** $i = 1$ to SN **do**
- 2: select k , where $k \in \{1,2, \dots, SN\}$ and $k \neq i$
- 3: select j , where $j \in \{1,2, \dots, D\}$
- 4: $v_{ij} = x_{ij} + rand(-1,1) * (x_{ij} - x_{kj})$
- 5: **if** ($f(v_i) > f_i$) **then**
- 6: $x_i = v_i$
- 7: $f_i = f(v_i)$
- 8: $failure_i = 0$
- 9: **else**
- 10: $failure_i = failure_i + 1$
- 11: **end if**
- 12: **end for**

Adım 4: Olasılık değerlerinin hesaplanması

- 1: $sum_probability = 0$
- 2: **for** $i = 1$ to SN **do**
- 3: $sum_probability = sum_probability + f_i$
- 4: **end for**
- 5: **for** $i = 1$ to SN **do**
- 6: $p_i = f_i / sum_probability$
- 7: **end for**

Adım 5 : Gözcü arı Fazı

- 1: $random = rand(0, 1)$
- 2: $i = 1, t = 1;$
- 3: **while** ($t \leq SN$) **do**
- 4: **if** ($p_i > random$) **then**
- 5: $t++;$
- 6: select k , where $k \in \{1,2, \dots, SN\}$ and $k \neq i$
- 7: select j , where $j \in \{1,2, \dots, D\}$
- 8: $v_{ij} = x_{ij} + rand(-1,1) * (x_{ij} - x_{kj})$
- 9: **if** ($f(v_i) > f_i$) **then**
- 10: $x_i = v_i$
- 11: $f_i = f(v_i)$
- 12: $failure_i = 0$
- 13: **else**
- 14: $failure_i = failure_i + 1$
- 15: **end if**
- 16: **else**

```

17:          $i++$ 
18:     end if
Adım 6: Kaşif Arı Fazı
1: if ( $failure_i \geq limit$ ) then
2:     for  $j = 1$  to  $D$  do
3:          $x_{ij} = x_j^{min} + rand(0,1) * (x_j^{max} - x_j^{min})$ 
4:     end for
5:      $f_i = f(x_i)$ 
6:      $failure_i = 0$ 
7: end if
Adım 7: Sonlandırma Koşulları
1: while ( $iter \leq MaxCycle$ ) do
2:     Repeat STEP III to STEP VI
3: end while
4: Ulaşılan en iyi besin kaynağını döndür.

```

Bu çalışma kapsamında YAK algoritmasında yer alan sınırlılıklara ait parametreler ve değerlerine Tablo 3.5 ve Tablo 3.6’da yer verilmiştir.

Tablo 3.6. Optimizasyon Sınırlılıkları

Parametre	Değer
Açı	1-20
Kamburluk	1-20
Kalınlık	1-20
Hava Yoğunluğu	0.00511844

Tablo 3.7. Optimizasyon Algoritması (YAK)'nın kontrol parametreleri

Parametre	Değer
Koloni Boyutu (işçi arı + gözcü arı)	40
Sınır (Denemelerle geliştirilemeyen besin kaynağı)	10000
Maksimum Döngü (Yiyecek arama döngülerinin sayısı, durdurma kriterleri)	180
Algoritmaya ait parametre sayısı: D	6

Uygunluk fonksiyonu, aday çözümler için gerçek çözümler üretmektedir ya da iki aday çözüm arasından hangisinin daha iyi çözüm olduğu ile ilgili bilgi vermektedir. Optimizasyonda asıl amaç $X = \{x_1, x_2, \dots, x_n\}$ girdi değerleri doğrultusunda oluşan $F = \{f_1, f_2, \dots, f_n\}$ maliyet değerleri için X kümesi içerisinde mümkün olabilecek en iyi değeri bulmaktır. Amaç fonksiyonu genellikle direkt uygunluk fonksiyonu olarak kullanılmasını yanında uygunluk fonksiyonu amaç fonksiyonu ile aynı olmak zorunda değildir (Akay, 2009). Amaç fonksiyonu tasarımda optimize edilecek olan fonksiyondur ve problemin türüne göre maksimizasyon (en büyüklenmesi) ya da minimizasyon (en küçükleme) yapılmaya çalışılır (Yang, 2010). Bu çalışmada tasarımcıların istedikleri kaldırma kuvveti değerine en uygun açı, kamburluk ve kalınlık değerlerini sumak için YAK algoritmasında kullanılan ve minimize edilmeye çalışılan amaç fonksiyonu aşağıda verilmiştir.

$$f = @(x_1, x_2, x_3, x_4, x_5, x_6) (abs((1/2) * MLFFNN_Model([x_1; x_2; x_3]) * x_4 * x_5.^2 * x_6 - L)); \quad (3.7)$$

$$f = \frac{\rho \cdot A \cdot C_L \cdot V^2}{2} \quad (3.8)$$

$$x_1 = \text{Açı}$$

$$x_2 = \text{Kamburluk}$$

$$x_3 = \text{Kalınlık}$$

$$x_4 = \text{Hava yoğunluğu}(\rho)$$

$$x_5 = \text{Kanat profili üzerindeki hava hızı}(V)$$

$$x_6 = \text{Kanat alanı}(A)$$

MLFNN_Model = Eğitilmiş Çok Katmanlı Yapay Sinir Ağı (çıkış değeri C_L)

L = Kaldırma değeri (Tasarımcının istediği kaldırma değeri).

Literatürde birçok optimizasyon algoritması yer almaktadır. Hem gösterdiği performans hem de algoritmaya hakimiyet sebebiyle tez çalışması kapsamında YAK algoritması tercih edilmiştir.



4. BULGULAR

Uçak kanat profil şekli optimizasyonu hesaplanması zor ve maliyetlidir. FoilSim verilerinin 5-katlamalı çapraz doğrulama yöntemleriyle kullanılmasıyla YSA ile eğitilmiş vekil ağ elde edilerek YAK algoritması ile optimize edildikten sonra 28 deney gerçekleştirilmiştir. Deneyler başlangıç aşaması, birinci aşama ve ikinci aşama olmak üzere gruplandırılmıştır. Bu deneylerin başlangıç aşamasında sabit hız ve sabit kanat alanında açılış kamburluk ve kalınlık değerleri bulunmuştur (Deney 1, Deney 8, Deney 15, Deney 22).

Sonrasında aynı sabit hız ve farklı bir sabit kanat alanında; kamburluk ve kalınlık değerleri sabit tutulmuş ve açılış değeri gözlemlenmiştir (Deney 2, Deney 9, Deney 16, Deney 23). Deney sonucunda açılış değerlerinde **düşüş** olduğu tespit edilmiştir. Deney 3, Deney 10, Deney 17, Deney 24'te de açılış ve kalınlık değerleri sabit tutulup kamburluk değeri gözlemlenmiştir. Gözlem sonucunda kamburluk değerinde **düşüş** olduğu gözlemlenmiştir. İlk aşamanın son deneylerinde de (Deney 4, Deney 11, Deney 18, Deney 25) açılış ve kamburluk değerleri sabit tutulmuş ve kalınlık değerleri gözlemlenmiştir. Gözlem sonucunda kalınlık değerlerinde **düşüş** olduğu tespit edilmiştir.

Tablo 4.1. Açılış, kamburluk ve kalınlık değerlerinin gözlemlenmesi (Kaldırma Değeri = 8 lbs olduğu durumda).

YAK Kaldırma Değeri (lbs)	Deney no	Sabit Hız (mph)	Sabit Kanat Alanı (sqft)	Optimize edilmiş Açılış (°)	Optimize edilmiş Kamburluk (%)	Optimize edilmiş Kalınlık (%)	FoilSim-Kaldırma Değeri (lbs)
	1	24	2.6	14.1459	8.35663	7.27498	8
	2	24	2.8	9.41287	8.35663	7.27498	8
	3	24	2.8	14.1459	5.66572	7.27498	8
	4	24	2.8	14.1459	8.35663	1.99969	8
	5	25.4	2.6	17.3841	8.35663	7.27498	8
	6	25.4	2.6	14.1459	6.1912	7.27498	8
	7	25.4	2.6	14.1459	8.35663	1.19178	8

Deneyin ikinci aşamasında; ilk aşamanın başlangıcındaki kanat alanı aynı bırakılıp farklı bir sabit hızda, kamburluk ve kalınlık değerleri sabit tutulmuş ve açı değeri gözlemlenmiştir (Deney 5, Deney 12, Deney 19, Deney 26). Gözlem sonucunda açı değerlerinde **artış** olduğu tespit edilmiştir.

Tablo 4.2. Açı, kamburluk ve kalınlık değerlerinin gözlemlenmesi (Kaldırma Değeri =17 lbs olduğu durumda).

YAK Kaldırma Değeri (lbs)	Deney no	Sabit Hız (mph)	Sabit Kanat Alanı (sqft)	Optimize edilmiş Açı (°)	Optimize edilmiş Kamburluk (%)	Optimize edilmiş Kalınlık (%)	FoilSim-Kaldırma Değeri (lbs)
17	8	30.4	3.3	12.6009	10.2219	8.22825	17
	9	30.4	3.45	9.80014	10.2219	8.22825	17
	10	30.4	3.45	12.6009	8.41263	8.22825	17
	11	30.4	3.45	12.6009	10.2219	4.05814	17
	12	31.28	3.3	16.7070	10.2219	8.22825	17
	13	31.28	3.3	12.6009	8.73887	8.22825	17
	14	31.28	3.3	12.6009	10.2219	1.01248	17

Deney 6, Deney 13, Deney 20 ve Deney 27’ de açı ve kalınlık değerleri sabit tutulmuş ve kamburluk değerleri gözlemlenmiştir. Kamburluk değerlerinde **düşüş** olduğu tespit edilmiştir. İkinci aşamanın son deneylerinde (Deney 7, Deney 14, Deney 21, Deney 28) de açı ve kamburluk değerleri sabit tutulmuş ve kalınlık değeri gözlemlenmiştir. Deney sonucunda kalınlık değerlerinde **düşüş** olduğu gözlemlenmiştir.

Tablo 4.3. Aç, kamburluk ve kalınlık deęerlerinin gözlemlenmesi (Kaldırma Deęeri = 52 lbs olduęu durumda).

YAK Kaldırma Deęeri (lbs)	Deney no	Sabit Hız (mph)	Sabit Kanat Alanı (sqft)	Optimize edilmiş Aç (°)	Optimize edilmiş Kamburluk (%)	Optimize edilmiş Kalınlık (%)	FoilSim-Kaldırma Deęeri (lbs)
52	15	41	4.2	14.6894	19.9987	13.6096	52
	16	41	4.3	9.56763	19.9987	13.6096	52
	17	41	4.3	14.6894	18.9501	13.6096	52
	18	41	4.3	14.6894	19.9987	2.7278	52
	19	41.8	4.2	16.0302	19.9987	13.6096	52
	20	41.8	4.2	14.6894	18.2971	13.6096	52
	21	41.8	4.2	14.6894	19.9987	1.1346	52

Tablo 4.4. Aç, kamburluk ve kalınlık deęerlerinin gözlemlenmesi. (Kaldırma Deęeri = 157 lbs olduęu durumda).

YAK Kaldırma Deęeri (lbs)	Deney no	Sabit Hız (mph)	Sabit Kanat Alanı (sqft)	Optimize edilmiş Aç (°)	Optimize edilmiş Kamburluk (%)	Optimize edilmiş Kalınlık (%)	FoilSim-Kaldırma Deęeri (lbs)
157	22	70.7	4.5	15.0302	17.9987	12.6096	157
	23	70.7	4.65	9.29409	17.9987	12.6096	157
	24	70.7	4.65	15.0302	16.6899	12.6096	157
	25	70.7	4.65	15.0302	17.9987	1.8696	157
	26	72.1	4.5	16.2354	17.9987	12.6096	157
	27	72.1	4.5	15.0302	16.5710	12.6096	157
	28	72.1	4.5	15.0302	17.9987	1.089009	158

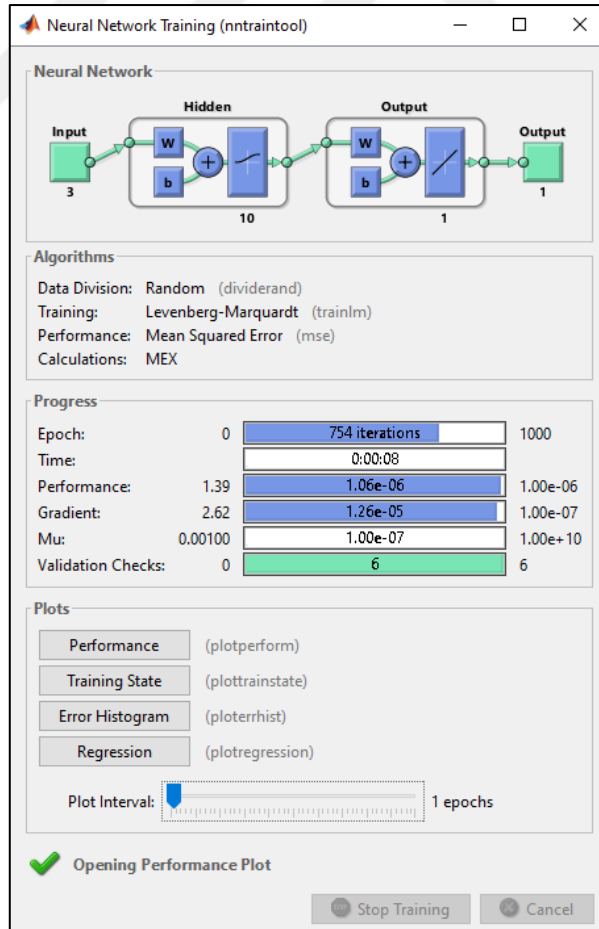
Bu çalışma, optimizasyonda FoilSim verilerini kullanarak optimum kanat profilini belirleyerek verimli uçak tasarımına katkı sağlamayı amaçlamaktadır. Deney sonucunda aç, kamburluk ve kalınlık deęerleri ABC algoritması ile elde edilmiřtir. Sonular FoilSim web ortamında kontrol edilmiřtir. FoilSim verileri ve ABC algoritması ile optimize edilmiř deęerlerle elde edilen kaldırma deęerlerinin birbirine ok yakın olduęu gzlemlenmiř olup, test sonuları Tablo 4.1, Tablo 4.2, Tablo 4.3, Tablo 4.4'da sunulmuřtur.



5. SONUÇ VE ÖNERİLER

Havacılık sektörüne olan talepte son yıllarda belirgin bir şekilde artış gözlemlenmiştir. Sektöre olan talep artıka hava yolu trafiğinin çevreye verdiği zararı da beraberinde getirmektedir. Bu sebeptendir ki hava yolu araçları için enerji verimliliği, hava araçlarının enerji tüketimini azaltmak, maliyet en aza indirmek ve aynı zamanda çevreye verdiği zararlar sebebiyle gündemde olan ve birçok araştırmacının üzerinde çalıştığı bir konudur. Çevresel etkiyi en aza indirirken hız ve kapasiteyi artırmak üzerinde durulan bir konu haline gelmiştir. Aerodinamik, nesnelerin havadaki kuvvetleri ve hareketlerinin, başka bir ifadeyle hava ile iletişiminin incelenmesidir.

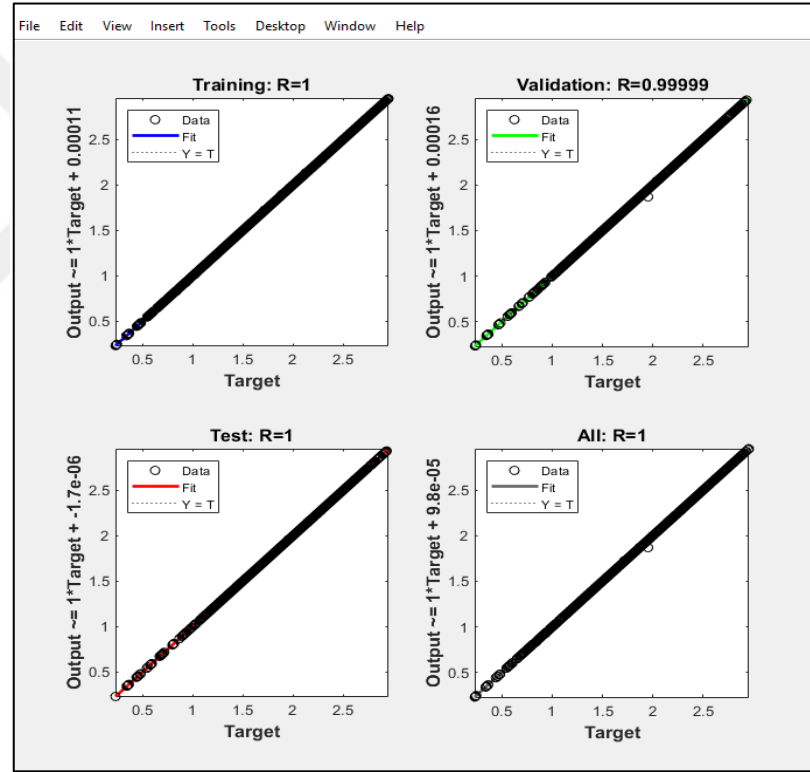
Çalışmada FoilSim web yazılımından alınan verilerle kanat profilinin aerodinamiğinde önemli bir rol alan kanat profiline ait şekilsel özellikleri tahmin etmek için vekil bir model ile optimizasyon yaklaşımı sunulmaktadır. Önerilen bu çalışma MATLAB R2019 a sayısal hesaplama programı ile tasarlanmıştır.



Şekil 5.1. Ağa ait nntraintool arayüzü.

FoilSim web sayfasından açılış, kamburluk, kalınlık ve bu değerleri neticesinde kanat profil şeklinde olması gereken kaldırma katsayısı (C_L) değerini barındıran 8000 adet özgün veri seti oluşturulmuştur. Açılış kamburluk ve kalınlık değerlerinden kaldırma katsayısını tahmin etmek için 5-katlamalı çapraz doğrulama yöntemi kullanılmıştır. Böylelikle üç girişli (açılış, kamburluk, kalınlık) ve bir çıkışlı (C_L : Kaldırma Katsayısı) İleri Beslemeli Yapay Sinir Ağı (YSA) ile regresyon (tahmin) algoritması ile MATLAB 2019a programında eğitilmesi sağlanmıştır.

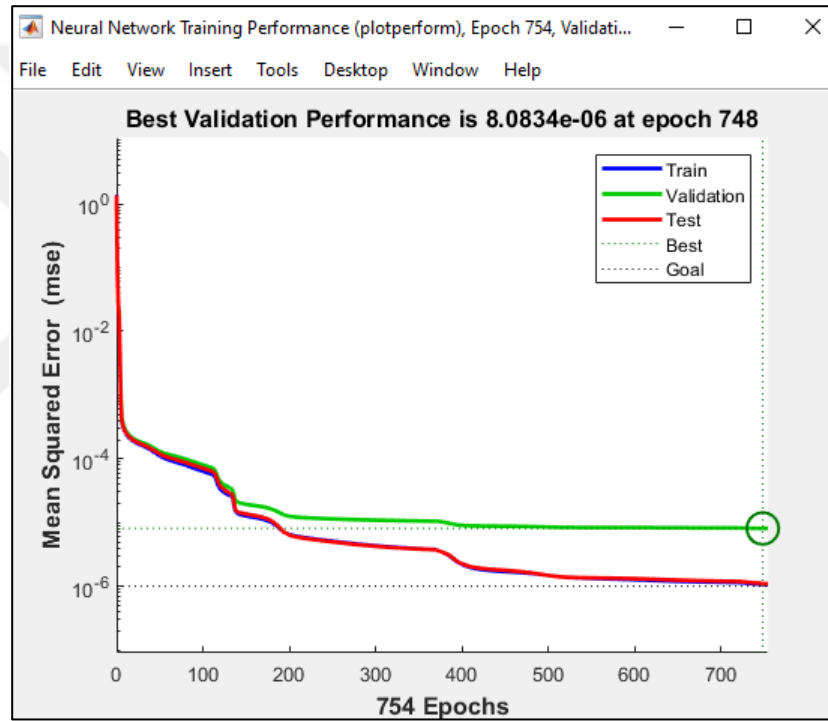
$$f = \frac{\rho \cdot A \cdot C_L \cdot V^2}{2} \quad (4.1)$$



Şekil 5.2. Ağa ait regresyon analizi.

Eşitlik 4.1’de yer alan ρ (hava yoğunluğu), V (kanat profili üzerindeki hava hızı), A (kanat alanı), L (Kaldırma değeri) değerlerinin her bir deneyler esnasında sabit tutulan değerlerdir. Optimizasyon için tahmin başarısını etkileyen en önemli faktör kaldırma katsayısıdır (C_L). Kaldırma katsayısının doğruluk oranı kaldırma denklemi içerisinde önemli bir yere sahiptir. Dolayısıyla değişken olarak yer alan kaldırma katsayısı değeri (C_L) testlerle belirlenmiştir.

Yapılan eğitim ile vekil ağ oluşturulmuş ve EK1’de de detaylı yer verildiği gibi toplamda 486 test gerçekleştirilmiştir. Her bir test için 8000 adet veri seti 5 katlamalı çapraz doğrulama ile ayrılmış her parçanın performansları toplanmış ve sonrasında her parça için performans ortalaması alınmıştır. Yapılan bu testler sonucunda en iyi performansa (en az hata oranına) sahip ağ seçilmiştir. Çıktı değeri kaldırma katsayısı olan vekil ağa ait doğruluk oranı %99,9 olarak belirlenmiştir. Vekil ağ için bulunan doğruluk oranı kaldırma denkleminde yer alan kaldırma değeri için de geçerlidir. Kaldırma katsayısının doğruluk oranı kaldırma değerinin doğruluk oranı ile paraleldir ve Tablo 4.1, Tablo 4.2, Tablo 4.3 ve Tablo 4.4’te optimizasyon algoritması içerisinde yer alan ve FoilSim web sayfasında da denenip test edilmiş değerlerle de desteklenmektedir.



Şekil 5.3. Ağa ait performans görseli.

Yapay Arı Kolonisi (YAK) algoritmasının kullanılması için optimizasyon sürecinde farklı aç, kamburluk ve kalınlık değerleri için kaldırma katsayısı, eğitilen YSA tarafından üretilmektedir. Böylece tasarımcının istediği kaldırma değerine en uygun aç, kamburluk ve kalınlık değerleri bulunmaktadır. Bulunan bu ideal değerler *FoilSim III Elementary Version 1.1.c* programında test edilmiş ve yeterli doğrulukta çalıştığı gözlemlenmiştir.

Mevcut çalışmaya sürüklenme katsayısı (C_D) ve sürüklenme değeri (D) de dahil edilip çalışma daha da genişletilebilir. Bunun yanında tez çalışması kapsamında hazırlanan optimizasyon gerçek kanat profillerinde kullanılıp, gerçek kanat profili sonuçları ile test edilip eksiklikleri giderilebilir. YAK algoritması ile elde edilen optimum değerlerin

performansları farklı optimizasyon algoritmalarından elde edilecek sonuçlar ile karşılaştırılabilir.



6. KAYNAKLAR

- Ab Wahab, M. N., Nefti-Meziani, S. ve Atyabi, A. (2015). A Comprehensive Review of Swarm Optimization Algorithms. *PLOS ONE*, 10(5), e0122827. doi.org/10.1371/journal.pone.0122827.
- Abd Elaziz, M., Elsheikh, A. H., Oliva, D., Abualigah, L., Lu, S. ve Ewees, A. A. (2022). Advanced Metaheuristic Techniques for Mechanical Design Problems: Review. *Archives of Computational Methods in Engineering*, 29(1), 695–716. doi:10.1007/S11831-021-09589-4/FIGURES/6.
- Aerodynamic Forces*. Eriřim: 28.12.2021. <https://www.aerorann.com/basics-of-aerodynamics-aeroscience>.
- Afonso, F., Vale, J., Lau, F. ve Suleman, A. (2017). Performance Based Multidisciplinary Design Optimization Of Morphing Aircraft. *Aerospace Science and Technology*, 67, 1–12. doi:10.1016/j.ast.2017.03.029.
- Aggarwal, C. C. (2018). *Neural Networks and Deep Learning*. Springer Cham <https://link.springer.com/book/10.1007/978-3-319-94463-0>. doi:10.1007/978-3-319-94463-0.
- Ağyar, Z. (2015). Yapay Sinir Ağlarının Kullanım Alanları ve Bir Uygulama. *Mühendis ve Makina*, 56(662), 22–23.
- Ahmed, H. R., Glasgow, J. I. (2012). Swarm Intelligence: Concepts, Models and Applications. *Queen's University, School of Computing Technical Reports, Teknik Rapor* 2012-585, Kingston, Canada.
- Airfoil*. Eriřim: 3.01.2022. <https://swyde.com/s/Airfoil>.
- Akay, B. (2009). *Nümerik Optimizasyon Problemlerinde Yapay Arı Kolonisi Algoritmasının Performans Analizi*, 40-41, (doktora tezi), Erciyes Üniversitesi, Kayseri.
- Akyol, S. ve Alatař, B. (2012). Güncel Sürü Zekâsı Optimizasyon Algoritmaları. *Nevşehir Üniversitesi Fen Bilimleri Enstitü Dergisi*, 1, 36–50.
- Alatař, B. (2007). *Kaotik Haritalı Parçacık Sürü Optimizasyon Algoritmaları*. 1-15. (doktora tezi), Fırat Üniversitesi. Elazığ.
- Arias-Montaño, A., Coello Coello, C. A. ve Mezura-Montes, E. (2012). Multi-objective airfoil shape optimization using a multiple-surrogate approach. *2012 IEEE Congress on Evolutionary Computation*. doi:10.1109/CEC.2012.6256491.
- Ashfaq, J. M. ve Iqbal, A. (2019). *Introduction to Support Vector Machines and Kernel Methods*. 1-9, <https://www.researchgate.net/publication/332370436>.

- Baharözü, E., Soykan, G. ve Özerdem, M. B. (2017). Future Aircraft Concept in Terms of Energy Efficiency and Environmental Factors. *Energy*, 140, 1368–1377. doi:10.1016/j.energy.2017.09.007.
- Berrar, D. (2018). Cross-Validation. S. Ranganathan, M. Gribskov, K. Nakai ve C. Schönbach (Ed.), *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, 1-3, 542–545. Academic Press. doi:10.1016/B978-0-12-809633-8.20349-X.
- Blum, C., Li X. (2008) Swarm Intelligence in Optimization. Blum C., Merkle D.(Ed). *Swarm Intelligence. Natural Computing Series*, 43-85, Berlin, Heidelberg: Springer. doi.org/10.1007/978-3-540-74089-6_2.
- Cheng, Y., Jiang, M. ve Yuan, D. (2009). Novel Clustering Algorithms Based on Improved Artificial Fish Swarm Algorithm. *6th International Conference on Fuzzy Systems and Knowledge Discovery*, (s.141–145). doi: 10.1109/fskd.2009.534.
- Comis Da Ronco, C., Ponza, R. ve Benini, E. (2015). Aerodynamic Shape Optimization of Aircraft Components Using an Advanced Multi-Objective Evolutionary Approach. *Computer Methods in Applied Mechanics and Engineering*, 285, 255–290. doi:10.1016/j.cma.2014.10.024.
- Cross-validation. Erişim: 19.03.2022, [https://en.wikipedia.org/wiki/Cross-validation_\(statistics\)](https://en.wikipedia.org/wiki/Cross-validation_(statistics)).
- Çetinsoy, E., Öner, K. T., Kandemir, İ., Akşit, M. F., Ünel, M. ve Gülez, K. (2008). Yeni Bir İnsansız Hava Aracının (SUAVi) Mekanik ve Aerodinamik Tasarımı. *Türkiye Otomatik Kontrol Konferansı*, (s. 890-896). İstanbul.
- Dina, A., Danaila, S., Pricop, M.-V. ve Bunescu, I. (2019). Using Genetic Algorithms to Optimize Airfoils in Incompressible Regime. *INCAS BULLETIN*, 11, 79–90. doi:10.13111/2066-8201.2019.11.1.6.
- Dorigo, M., Maniezzo, V. ve Coloni, A. (1996). Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 26(1), 29–41. doi:10.1109/3477.484436.
- Dökeroğlu, T., Sevinç, E., Küçükyılmaz, T. ve Coşar, A. (2019). A Survey on New Generation Metaheuristic Algorithms. *Computers and Industrial Engineering*, 137. doi:10.1016/J.CIE.2019.106040.
- Du, X., He, P. ve Martins, J. R. R. A. (2021). Rapid Airfoil Design Optimization via Neural Networks-Based Parameterization and Surrogate Modeling. *Aerospace Science and Technology*, 113, 106701. doi:10.1016/J.AST.2021.106701.
- Elmas, Ç. (2018). *Yapay Zeka Uygulamaları*, (Birinci Basım), 1-58, Ankara: Seçkin Yayınları.
- Engelbrecht A.P. (2007). *Fundamentals of Computational Swarm Intelligence*, Chicester John Wiley & Sons Inc.

- Farley, B. G. ve Clark, W. A. (1954). Simulation of Self-Organizing Systems by Digital Computer. *IRE Professional Group on Information Theory*, 4(4), 76–84. doi:10.1109/TIT.1954.1057468.
- Gabor, O. S., Simon, A., Koreanschi, A. ve Botez, R. M. (2016). Aerodynamic Performance Improvement of The UAS-S4 Éhecatl Morphing Airfoil Using Novel Optimization Techniques. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 230(7), 1164–1180. doi:10.1177/0954410015605548.
- Goodfellow, I., Bengio, Y., Courville, A.(2016). *Deep Learning* (1.Basım). 96-152, Cambridge: MIT press.
- Greyber, A., Rand, C. ve Stockburger, A. (2016). *Airfoil Simulation Comparison*. Erişim:11.04.2022.https://workstory.s3.amazonaws.com/assets/1436089/Technical_Report_V2.pdf
- Han, Z. H., Abu-Zurayk, M., Görtz, S. ve Ilic, C. (2015). Surrogate-Based Aerodynamic Shape Optimization of a Wing-Body Transport Aircraft Configuration. *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, 138, 257–282. doi:10.1007/978-3-319-72020-3_16.
- Han, Z. H., Zhang, K. S., Liu, J. ve Song, W. P. (2013). Surrogate-Based Aerodynamic Shape Optimization with Application to Wind Turbine Airfoils. *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition 2013*. doi:10.2514/6.2013-1108.
- Han, Z., Xu, C., Zhang, L., Zhang, Y., Zhang, K. ve Song, W. (2020). Efficient Aerodynamic Shape Optimization Using Variable-Fidelity Surrogate Models and Multilevel Computational Grids. *Chinese Journal of Aeronautics*, 33(1), 31–47. doi:10.1016/j.cja.2019.05.001.
- Hebb, D. O. (1949). *The Organization of Behavior; a Neuropsychological Theory*, (1.Basım). New York: Wiley.
- Huang, S., Lu, X., Han, G., Zhao, S., Zhou, C. ve Yang, C. (2022). Research on Aerodynamic Optimization Design Method and Flow Mechanism of a High-Subsonic Compressor Cascade. *Engineering Applications of Computational Fluid Mechanics*, 16(1), 316–334. doi:10.1080/19942060.2021.2020170.
- Jameel, A., Khalid, S., Khan, M., Kamal, K., Aqeel, A., Mathavan, S., Zafar, T. (2015). A Bio Inspired UAV Model Based on Corvus Splendens. *Proceedings of 2015 International Conference on Fluid Power and Mechatronics, Institute of Electrical and Electronics Engineers Inc.* (ss. 797–802). doi:10.1109/FPM.2015.7337223.
- James,W.(1984). *Psychology:Briefer Course*. USA: Harvard University Press.
- Jameson, A. (1988). Aerodynamic Design via Control Theory. *Journal of Scientific Computing*, 3(3), 233–260. doi:10.1007/BF01061285.

- Jameson, A. ve Martinelli, L. (2000). Aerodynamic Shape Optimization Techniques Based on Control Theory, *Computational Mathematics Driven by Industrial Problems, Lecture Notes in Mathematics*, Hiedberg: Spring. 151-221. doi:10.1007/bfb0103920.
- Jeong, S., Obayashi, S. ve Yamamoto, K. (2005). Aerodynamic Optimization Design with Kriging Model. *Transactions of the Japan Society for Aeronautical and Space Sciences*, 48(161), 161–168. doi:10.2322/tjsass.48.161.
- Karaboğa, D., (2011). *Yapay Zekâ Optimizasyon Algoritmaları*, (2.Basım) Ankara: Nobel Yayın Dağıtım.
- Karaboğa, D., (2017). *Yapay Zekâ Optimizasyon Algoritmaları*, (5.Basım), Ankara: Nobel Yayın Dağıtım.
- Karaboğa, D. ve Akay, B. (2009). A Comparative Study of Artificial Bee Colony Algorithm. *Applied Mathematics and Computation*, 214(1), 108–132. doi:10.1016/J.AMC.2009.03.090.
- Kaya, B. ve Eke, İ. (2020). Yapay Arı Kolonisi Algoritması ile Yapılan Geliştirmeler ve Sonuçları. *Verimlilik Dergisi T.C. Sanayi ve Teknoloji Bakanlığı Yayını*, 17(1), 99–115.
- Kaya, M. ve Elfarrar, M. (2019). Optimization of the Taper/Twist Stacking Axis Location of NREL VI Wind Turbine Totor Blade Using Neural Networks Based on Computational Fluid Dynamics Analyses. *Journal of Solar Energy Engineering, Transactions of the ASME*, 141(1). doi:10.1115/1.4041102.
- Kennedy, J. ve Eberhart, R. (1995). Particle Swarm Optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, 4, 1942–1948. doi:10.1109/ICNN.1995.488968.
- Kevadiya, M. ve Vaidya, H. A. (2013). 2D Analysis of Naca 4412 Airfoil. *International Journal of Innovative Research in Science, Engineering and Technology*, 2(5). 1686-1691.
- Khurana, M. S., Winarto, H. ve Sinha, A. K. (2009). Airfoil optimisation by swarm algorithm with mutation and Artificial Neural Networks. *47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 1278. doi:10.2514/6.2009-1278.
- Kızrak, A. (2018). Şu Kara Kutuyu Açalım: Yapay Sinir Ağları. Erişim: 2 Ocak 2022 <https://ayyucekizrak.medium.com/şu-kara-kutuyu-açalım-yapay-sinir-ağları-7b65c6a5264a>.
- Kızrak, A. (2019). Derin Öğrenme için Aktivasyon Fonksiyonlarının Karşılaştırılması. Erişim: 8 Ocak 2022 <https://ayyucekizrak.medium.com/derin-öğrenme-için-aktivasyon-fonksiyonlarının-karşılaştırılması-cee17fd1d9cd>.

- Kirkpatrick, S., Gelatt, C. D. ve Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598), 671–680. doi:10.1126/science.220.4598.671.
- Kohonen, T. (1982). Self-Organized Formation of Topologically Correct Feature Maps. *Biological Cybernetics*, 43(1), 59–69. doi:10.1007/BF00337288.
- Koziel, S. ve Leifsson, L. (2013). Surrogate-Based Aerodynamic Shape Optimization by Variable-Resolution Models. *AIAA Journal*, 51(1), 94–106. doi:10.2514/1.J051583.
- Krenker A., Bešter, J. ve Kos, A. (2011). Introduction to the Artificial Neural Networks. Kenji Suzuki (Ed.). *Artificial Neural Networks Methodological Advances and Biomedical Applications*. doi: 10.5772/15751.
- KT, M. V., R, A., Sharma, B., Babu, G. ve Khanapure, K. B. (2020). Design and Fabrication of a Fixed Wing UAV. *International Journal of Scientific Research and Engineering Development*, 3(3), 365–373.
- Kumar, K., Prakash Mishra, O. ve Kumar, S. (2020). Simulation of Airfoil Shape for Optimum Wing Characteristics. *Materials Today: Proceedings*, 24, 2231–2237. doi:10.1016/j.matpr.2020.03.749.
- Li, J., Cai, J. ve Qu, K. (2019). Surrogate-Based Aerodynamic Shape Optimization with the active subspace method. *Structural and Multidisciplinary Optimization*, 59(2), 403–419. doi:10.1007/S00158-018-2073-5.
- Liu, J., Wang, D. ve Luo, S. (2020). An Effective Constraint-Handling Improved Cuckoo Search Algorithm and Its Application in Aerodynamic Shape Optimization. *IEEE Access*, 8, 139121–139142. doi:10.1109/access.2020.3012606.
- Lyu, Z., Kenway, G. K. W. ve Martins, J. R. R. A. (2014). Aerodynamic Shape Optimization Investigations of the Common Research Model Wing Benchmark. *AIAA Journal*, 53(4), 968–985. doi:10.2514/1.J053318.
- Lyu, Z., ve Martins, J. R. R. A. (2014). Strategies for Solving High-Fidelity Aerodynamic Shape Optimization Problems, *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2594. doi:10.2514/6.2014-2594.
- Ma, P., Yu, J., Chen, F. ve Xue, Z. (2017). Airfoil Optimization Design Based On a Combined Optimization Strategy, *Proceedings of the 2017 5th International Conference on Frontiers of Manufacturing Science and Measuring Technology*, 529–537. doi:10.2991/fmsmt-17.2017.109.
- MacEachern, C. ve Yıldız, I. (2018). Wind Energy. *Comprehensive Energy Systems*. 1(5), 665–701. Elsevier Inc. doi:10.1016/b978-0-12-809597-3.00118-8.
- Maltarollo, V. G. , Honório, K. M. , Silva, A. B. F. d. (2013). Applications of Artificial Neural Networks in Chemical Problems. Kenji Suzui (Ed.) *Artificial Neural Networks - Architectures and Applications*. Rijeka: IntechOpen. doi:10.5772/51275.

- Mar Aye, C., Pholdee, N. ve Bureerat, S. (2020). Surrogate-Assisted Meta-Heuristic Method for Aerodynamic Design of an Aircraft Wing. *IOP Conference Series: Materials Science and Engineering*, 886(1), 012026. doi:10.1088/1757-899x/886/1/012026.
- McCulloch, W. S. ve Pitts, W. (1943). A Logical Calculus of the Ideas Immanent in Nervous Activity. *The bulletin of mathematical biophysics* 5(4), 115–133. doi:10.1007/bf02478259.
- Messac, A. (2015). *Optimization in Practice with MATLAB : For Engineering Students and Professionals* (1. Basım). USA: Cambridge University Press, 119-125.
- Minsky, M. ve Papert, S. (1969). *Perceptrons; an Introduction to Computational Geometry*. Massachusetts: MIT Press.
- Mirjalili, S., Mirjalili, S. M. ve Lewis, A. (2014). Grey Wolf Optimizer. *Advances in Engineering Software*, 69, 46–61. doi: 10.1016/J.advengsoft.2013.12.007.
- Mirjalili, S., Song Dong, J., Lewis, A. ve Sadiq, A. S. (2020). Particle Swarm Optimization: Theory, Literature Review, and Application in Airfoil Design. Mirjalili S., Song Dong J., Lewis A. (Ed.). *Nature-Inspired Optimizers. Studies in Computational Intelligence*. 811, 167–184. Cham: Springer. doi:10.1007/978-3-030-12127-3_10.
- Mishra, A. (2019). Evaluating Machine Learning Models. *Machine Learning in the AWS Cloud*. Wiley. 115-132. doi:10.1002/9781119556749.
- Mitchell, T. M., Carbonell, J. G. ve Michalski, R. S. (1986). *Machine Learning: A Guide to Current Research*, (3. Basım) Norwell: Kluwer Academic Publishers. doi.org/10.1007/978-1-4613-2279-5.
- Mukesh, R., Lingadurai, K. ve Selvakumar, U. (2018). Airfoil Shape Optimization Based on Surrogate Model. *Journal of The Institution of Engineers (India): Series C*, 99(1), 1–8. doi:10.1007/S40032-017-0382-X.
- Nayyar A., Puri V., Suseendran G. (2019) Artificial Bee Colony Optimization Populatiton Based Meta- Heuristic Swarm Intelligence Technique. Balas V., Sharma N., Chakrabarti A. (Ed.) *Data Management, Analytics and Innovation. Advances in Intelligence Systems and Computing*, 839, 513-525, Singapur: Springer, doi: 10.1007/978-981-13-1274-8_38.
- Nwadiugwu, M. C. (2020). Neural Networks, Artificial Intelligence and the Computational Brain. *arXiv*, (1-36). doi:10.48550/arxiv.2101.08635.
- Öztemel, E. (2006). *Yapay Sinir Ağları*, (2.Basım), 29-103, İstanbul:Papatya Yayıncılık.
- Öztürk, C., Hançer, E. ve Karaboğa, D. (2014). Küresel En İyi Yapay Arı Koloni Algoritması ile Otomatik Kümeleme. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 29(4), 677–687.
- Öztürk, K. ve Şahin, M. E. (2018). A General View of Artificial Neural Networks and

- Artificial Intelligence, *Takvim-i Vekayi*, 6(2), 25–36.
- Payne, R. B. (1977). The Ecology of Brood Parasitism in Birds. *Annual Review of Ecology and Systematics*, 8(1), 1–28. doi:10.1146/annurev.es.08.110177.000245.
- Pehlivanoglu, V. (2013) *Havacılık ve Uzay Bilimlerine Giriş*, (3.Basım), 224-244, İstanbul:Hava Harp Okulu Matbaası.
- Pehlivanoglu, Y. V. ve Yağız, B. (2012). Aerodynamic Design Prediction Using Surrogate-Based Modeling in Genetic Algorithm Architecture. *Aerospace Science and Technology*, 23(1), 479–491. doi:10.1016/J.AST.2011.10.006.
- Perdikaris, P., Raissi, M., Damianou, A., Lawrence, N. D. ve Karniadakis, G. E. (2017). Nonlinear Information Fusion Algorithms for Data-Efficient Multi-Fidelity Modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473, 0751. doi:10.1098/rspa.2016.0751.
- Poli, R., Kennedy, J. ve Blackwell, T. (2007). Particle Swarm Optimization: An Overview. *Swarm Intelligence*, 1(1), 33–57. doi:10.1007/S11721-007-0002-0.
- Post, S., Seetharaman, S. ve Abimannan, S. (2010). A Design-Build-Test-Fly Project Involving Modeling, Manufacturing, And Testing. *ASME Annual Conference and Exposition, Conference Proceedings. 20-23 Haziran 2010*, 81438, doi:10.18260/1-2--15791.
- Refaeilzadeh, P., Tang, L. ve Liu, H. (2009). Cross-Validation. Liu L., Özsu M.T.(Ed.) *Encyclopedia of Database Systems*, 532–538, Boston, MA : Springer, doi:10.1007/978-0-387-39940-9_565.
- Sadrehaghghi, I.(2021). Aircraft Wing Shape Optimization (Rapor No:2.23), Erişim: 21.12.2021, https://www.researchgate.net/publication/318562320_Aircraft_Wing_Shape_Optimization. doi: 10.13140/rg.2.2.11383.73127/9
- Sadrnia, A., Soltani, H. R., Zulkifli, N., Ismail, N. ve Ariffin, M. K. A. (2014). A Review of Nature-Based Algorithms Applications in Green Supply Chain Problems. *International Journal of Engineering and Technology*, 6(3), 204–211. doi:10.7763/IJET.2014.V6.697.
- Sahwee, Z., Mahmood, A. S., Rahman N.A. ve Sahari, K. S. M. (2017). Wind Tunnel Evaluation for Control Transition from Elevator to Stabilator of Small UAV. *Journal of Engineering Science and Technology*, 12(6). 1617 – 1626
- Sammut, C. ve Webb, G. I. (2010). *Encyclopedia of machine learning*. Boston MA: Springer. doi:10.1007/978-0-387-30164-8
- Segee, M. C., Schetz, J. A. ve Kapania, R. K. (2016). Transonic Aerodynamics Analysis for Multidisciplinary Design Optimization Applications. *34th AIAA Applied Aerodynamics Conference. AIAA-2016-4039*. doi:10.2514/6.2016-4039.
- Shen, W. (2017). Robust Parameter Designs in Computer Experiments Using Stochastic

- Approximation. *Technometrics*, 59(4), 471–483. doi:10.1080/00401706.2016.1272493.
- Shiruru, K. (2016). An Introduction to Artificial Neural Network Article. *International Journal of Advance Research and Innovative Ideas in Education*. 1(5), 27-30.
- Sinir Ağları nedir?* (2020). Sinir Ağları , Erişim: 2 Ocak 2022 <https://www.ibm.com/cloud/learn/neural-networks>.
- Sun, G., Sun, Y. ve Wang, S. (2015). Artificial Neural Network Based Inverse Design: Airfoils and Wings. *Aerospace Science and Technology*, 42, 415–428. doi:10.1016/J.AST.2015.01.030.
- Tang, Z., Périaux, J. ve Dong, J. (2014). Constraints Handling in Nash/Adjoint Optimization Methods for Multi-Objective Aerodynamic Design. *Computer Methods in Applied Mechanics and Engineering*, 271, 130–143. doi:10.1016/J.CMA.2013.12.006.
- Tao, J. ve Sun, G. (2019). Application of Deep Learning Based Multi-Fidelity Surrogate Model to Robust Aerodynamic Design Optimization. *Aerospace Science and Technology*, 92, 722–737. doi:10.1016/j.ast.2019.07.00.
- Tereshko, V. ve Loengarov, A. (2005). Collective Decision-Making in Honey Bee Foraging Dynamics. *Computing and Information Systems Journal*, 9(3), ISSN 1252-9404.
- The Lift Equation*. Erişim: 12.11.2021. <https://www.grc.nasa.gov/www/k-12/rocket/lifteq.html>.
- Wang, J., Xie, F., Zheng, Y., Zhang, J., Yang, B. ve Ji, T. (2018). Virtual Stackelberg Game Coupled with The Adjoint Method for Aerodynamic Shape Optimization. *Engineering Optimization*, 50(10), 1733–1754. doi:10.1080/0305215X.2017.1409349.
- Wang, J., Zheng, Y., Chen, J., Xie, F., Zhang, J., Périaux, J. ve Tang, Z. (2020). Single/Two-Objective Aerodynamic Shape Optimization by a Stackelberg/Adjoint Method. *Engineering Optimization*, 52(5), 753–776. doi:10.1080/0305215X.2019.1618287.
- Wang, X. D., Hirsch, C., Kang, S. ve Lacor, C. (2011). Multi-Objective Optimization of Turbomachinery Using Improved NSGA-II and Approximation Model. *Computer Methods in Applied Mechanics and Engineering*, 200(9–12), 883–895. doi:10.1016/J.CMA.2010.11.014.
- Widrow, B. ve Hoff, M. E. (1989). Adaptive Switching Circuits. *Wescon Conference Record*, 33 ,709–717. San Francisco: Electronic Conventions Management.
- Yang, C. C. (1995). *Low Speed Virtual Wind Tunnel Simulation For Educational Studies In Introducing Computational Fluid Dynamics And Flow Visualization*. [yüksek lisans tezi], 1-50, University of Kansas, Lawrence.
- Yang, X. S., (2010). *Engineering Optimization: An Introduction with Metaheuristic Applications*, 3-83, Hoboken: John Wiley and Sons. doi:10.1002/9780470640425.

- Yang, X. S. ve Deb, S. (2009). Cuckoo Search via Lévy Flights. *World Congress on Nature and Biologically Inspired Computing, NABIC 2009 - Proceedings*, 210–214. doi:10.1109/NABIC.2009.5393690.
- Yılmaz, A. (2020). *Yapay Zeka*, (4.Basım), 60-89, İstanbul: İnkilap Kitabevi.
- Yılmaz, M., Koten, H., Çetinkaya, E. ve Coşar, Z. (2018). A Comparative CFD Analysis of NACA0012 and NACA4412 Airfoils. *Journal of Energy Systems*, 2(4), 145–159. doi:10.30521/jes.454193.
- Yu, Y., Lyu, Z., Xu, Z. ve Martins, J. R. R. A. (2018). On the Influence of Optimization Algorithm and Initial Design on Wing Aerodynamic Shape Optimization. *Aerospace Science and Technology*, 75, 183–199. doi:10.1016/j.ast.2018.01.016.
- Yükselen, M. Adil. (2006a). Kanat Profillerinin Aerodinamiği [PDF belgesi]. Erişim: 20 Aralık 2021. <https://web.itu.edu.tr/yukselen/Uck351/05-%20Kanat%20profillerinin%20aerodinami%F0i.pdf>.
- Yükselen, M. Adil. (2006b). Uçuşun Esasları:Kanatlar,Kuyruk Yüzeyleri [PDF belgesi]. Erişim: 20 Aralık 2021 <https://web.itu.edu.tr/yukselen/UCK111/B%F6l%FCm%205-%20U%E7u%FEun%20Esaslar%FD.pdf>.
- Zhang, M., Shao, C., Li, F., Gan, Y. ve Sun, J. (2006). Evolving Neural Network Classifiers and Feature Subset Using Artificial Fish Swarm. *International Conference on Mechatronics and Automation*, 25-28 Haziran 2006, (s.1598–1602). Luoyang: Institute of Electrical and Electronics Engineers. doi:10.1109/ICMA.2006.257414.
- Zhang, X., Xie, F., Ji, T., Zhu, Z. ve Zheng, Y. (2021). Multi-fidelity deep neural network surrogate model for aerodynamic shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 373, 113485. doi:10.1016/J.CMA.2020.113485.

EKLER

EK 1. Farklı deęerlere sahip hiperparametreler ile vekil YSA yapısı testleri.

Katman sayısı	Nöron	Öğrenme oranı	Aktivasyon fonksiyonu	Geri yayılım alg.	İşlem süresi(sn)	Hata değeri
1	2	0,001	sigmoid	trainbfg	24.890792	0.0113
1	2	0,001	sigmoid	trainlm	42.896020	3.5354e-04
1	2	0,001	sigmoid	trainbr	66.766429	4.8271e-04
1	2	0,001	Hiperbolik tanj.	trainbfg	44.072717	0.0011
1	2	0,001	Hiperbolik tanj.	trainlm	94.873060	3.5286e-04
1	2	0,001	Hiperbolik tanj.	trainbr	68.034830	4.7540e-04
1	2	0,001	ReLU	trainbfg	19.511010	0.0154
1	2	0,001	ReLU	trainlm	22.392797	0.0153
1	2	0,001	ReLU	trainbr	37.071917	0.0157
1	2	0,005	sigmoid	trainbfg	21.244281	0.0113
1	2	0,005	sigmoid	trainlm	38.304585	3.5430e-04
1	2	0,005	sigmoid	trainbr	51.219428	3.5291e-04
1	2	0,005	Hiperbolik tanj.	trainbfg	56.130343	7.2600e-04
1	2	0,005	Hiperbolik tanj.	trainlm	49.116204	0.0103
1	2	0,005	Hiperbolik tanj.	trainbr	50.896393	6.0218e-04
1	2	0,005	ReLU	trainbfg	14.538098	0.0759
1	2	0,005	ReLU	trainlm	13.634714	0.0152
1	2	0,005	ReLU	trainbr	48.318899	0.0034
1	2	0,01	sigmoid	trainbfg	24.437114	0.0220
1	2	0,01	sigmoid	trainlm	49.361670	4.7932e-04
1	2	0,01	sigmoid	trainbr	69.460162	4.8271e-04
1	2	0,01	Hiperbolik tanj.	trainbfg	36.721145	0.0011
1	2	0,01	Hiperbolik tanj.	trainlm	52.215444	4.7939e-04
1	2	0,01	Hiperbolik tanj.	trainbr	57.986910	3.5286e-04
1	2	0,01	ReLU	trainbfg	19.404729	0.0135
1	2	0,01	ReLU	trainlm	14.255487	0.0155
1	2	0,01	ReLU	trainbr	17.752416	0.0034
1	3	0,001	sigmoid	trainbfg	25.126319	0.0110
1	3	0,001	sigmoid	trainlm	51.051916	1.8943e-04
1	3	0,001	sigmoid	trainbr	61.111096	1.6322e-04
1	3	0,001	Hiperbolik tanj.	trainbfg	41.880116	0.0110
1	3	0,001	Hiperbolik tanj.	trainlm	35.461822	1.8828e-04
1	3	0,001	Hiperbolik tanj.	trainbr	56.164156	1.4445e-04
1	3	0,001	ReLU	trainbfg	17.470664	0.0027
1	3	0,001	ReLU	trainlm	14.160652	0.0019
1	3	0,001	ReLU	trainbr	44.263751	0.0021
1	3	0,005	sigmoid	trainbfg	20.843697	3.7323e-04
1	3	0,005	sigmoid	trainlm	47.369446	1.9058e-04
1	3	0,005	sigmoid	trainbr	54.429870	1.9177e-04
1	3	0,005	Hiperbolik tanj.	trainbfg	21.768310	4.9226e-04
1	3	0,005	Hiperbolik tanj.	trainlm	36.025581	1.8374e-04
1	3	0,005	Hiperbolik tanj.	trainbr	58.951185	1.4380e-04
1	3	0,005	ReLU	trainbfg	16.294392	0.0641
1	3	0,005	ReLU	trainlm	14.550681	0.0024

1	3	0,005	ReLU	trainbr	39.921007	0.0021
1	3	0,01	sigmoid	trainbfg	21.396143	8.0428e-04
1	3	0,01	sigmoid	trainlm	38.426865	1.9945e-04
1	3	0,01	sigmoid	trainbr	61.859601	1.5016e-04
1	3	0,01	Hiperbolik tanj.	trainbfg	19.878925	3.6469e-04
1	3	0,01	Hiperbolik tanj.	trainlm	37.033406	1.7568e-04
1	3	0,01	Hiperbolik tanj.	trainbr	52.508652	2.2057e-04
1	3	0,01	ReLU	trainbfg	15.604900	0.0029
1	3	0,01	ReLU	trainlm	13.679675	0.0019
1	3	0,01	ReLU	trainbr	48.759689	0.0021
1	4	0,001	sigmoid	trainbfg	26.770132	7.7378e-04
1	4	0,001	sigmoid	trainlm	37.825758	6.8417e-05
1	4	0,001	sigmoid	trainbr	46.424667	7.5166e-05
1	4	0,001	Hiperbolik tanj.	trainbfg	26.810064	2.9599e-04
1	4	0,001	Hiperbolik tanj.	trainlm	42.393012	7.9054e-05
1	4	0,001	Hiperbolik tanj.	trainbr	53.847982	5.8897e-05
1	4	0,001	ReLU	trainbfg	15.298285	0.0029
1	4	0,001	ReLU	trainlm	13.094262	0.0014
1	4	0,001	ReLU	trainbr	49.831243	7.0703e-04
1	4	0,005	sigmoid	trainbfg	19.128189	3.8651e-04
1	4	0,005	sigmoid	trainlm	37.310402	8.0365e-05
1	4	0,005	sigmoid	trainbr	37.165744	1.1067e-04
1	4	0,005	Hiperbolik tanj.	trainbfg	20.795381	3.8146e-04
1	4	0,005	Hiperbolik tanj.	trainlm	30.296934	7.9420e-05
1	4	0,005	Hiperbolik tanj.	trainbr	47.599537	8.3415e-05
1	4	0,005	ReLU	trainbfg	18.887095	0.0727
1	4	0,005	ReLU	trainlm	13.110387	0.0019
1	4	0,005	ReLU	trainbr	14.859351	0.0016
1	4	0,01	sigmoid	trainbfg	19.989434	7.9019e-04
1	4	0,01	sigmoid	trainlm	29.931999	8.6108e-05
1	4	0,01	sigmoid	trainbr	49.230966	1.2013e-04
1	4	0,01	Hiperbolik tanj.	trainbfg	20.945753	2.7377e-04
1	4	0,01	Hiperbolik tanj.	trainlm	37.535242	1.0606e-04
1	4	0,01	Hiperbolik tanj.	trainbr	38.504015	8.1743e-05
1	4	0,01	ReLU	trainbfg	14.217510	0.0273
1	4	0,01	ReLU	trainlm	13.479886	0.0011
1	4	0,01	ReLU	trainbr	36.549942	0.0013
1	5	0,001	sigmoid	trainbfg	22.253794	2.4640e-04
1	5	0,001	sigmoid	trainlm	38.072140	3.9125e-05
1	5	0,001	sigmoid	trainbr	56.182172	2.4419e-05
1	5	0,001	Hiperbolik tanj.	trainbfg	22.628411	2.6024e-04
1	5	0,001	Hiperbolik tanj.	trainlm	47.787976	1.5670e-05
1	5	0,001	Hiperbolik tanj.	trainbr	49.569876	1.8150e-05
1	5	0,001	ReLU	trainbfg	15.270168	0.0024
1	5	0,001	ReLU	trainlm	13.353087	9.0967e-04
1	5	0,001	ReLU	trainbr	39.120433	9.0348e-04
1	5	0,005	sigmoid	trainbfg	23.910819	4.3612e-04
1	5	0,005	sigmoid	trainlm	37.776928	6.1449e-05
1	5	0,005	sigmoid	trainbr	54.287221	3.7121e-05
1	5	0,005	Hiperbolik tanj.	trainbfg	23.066726	1.7547e-04
1	5	0,005	Hiperbolik tanj.	trainlm	39.077583	4.6064e-05
1	5	0,005	Hiperbolik tanj.	trainbr	53.43298	1.4660e-05
1	5	0,005	ReLU	trainbfg	15.622255	0.0024

1	5	0,005	ReLU	trainlm	13.451353	5.6554e-04
1	5	0,005	ReLU	trainbr	39.022715	5.4499e-04
1	5	0,01	sigmoid	trainbfg	35.759703	2.0388e-04
1	5	0,01	sigmoid	trainlm	46.921081,	1.9415e-05
1	5	0,01	sigmoid	trainbr	46.224218	1.5786e-05
1	5	0,01	Hiperbolik tanj.	trainbfg	26.848093	2.1999e-04
1	5	0,01	Hiperbolik tanj.	trainlm	60.631440	1.5589e-05
1	5	0,01	Hiperbolik tanj.	trainbr	45.320210	1.4026e-05
1	5	0,01	ReLU	trainbfg	23.631032	0.0638
1	5	0,01	ReLU	trainlm	13.707917	4.5972e-04
1	5	0,01	ReLU	trainbr	34.681806	7.2109e-04
1	6	0,001	sigmoid	trainbfg	23.293975	7.1236e-04
1	6	0,001	sigmoid	trainlm	43.989677	1.5905e-05
1	6	0,001	sigmoid	trainbr	53.649361	6.6669e-06
1	6	0,001	Hiperbolik tanj.	trainbfg	31.331510	1.2599e-04
1	6	0,001	Hiperbolik tanj.	trainlm	57.565252	7.5755e-06
1	6	0,001	Hiperbolik tanj.	trainbr	67.619537	8.2264e-06
1	6	0,001	ReLU	trainbfg		
1	6	0,001	ReLU	trainlm	13.610629	5.1310e-04
1	6	0,001	ReLU	trainbr	52.179305	3.2600e-04
1	6	0,005	sigmoid	trainbfg	20.630253	6.7263e-04
1	6	0,005	sigmoid	trainlm	7.400500	6.9264e-06
1	6	0,005	sigmoid	trainbr	57.936638	5.7120e-06
1	6	0,005	Hiperbolik tanj.	trainbfg	40.978281	1.3066e-04
1	6	0,005	Hiperbolik tanj.	trainlm	55.402677	1.0717e-05
1	6	0,005	Hiperbolik tanj.	trainbr	77.975066	6.8688e-06
1	6	0,005	ReLU	trainbfg	18.707116	0.0638
1	6	0,005	ReLU	trainlm	17.120296	6.2790e-04
1	6	0,005	ReLU	trainbr	29.137064	4.4834e-04
1	6	0,01	sigmoid	trainbfg	23.743992	2.4516e-04
1	6	0,01	sigmoid	trainlm	55.648097	8.1453e-06
1	6	0,01	sigmoid	trainbr	56.005719	6.0739e-06
1	6	0,01	Hiperbolik tanj.	trainbfg	23.779560	1.4655e-04
1	6	0,01	Hiperbolik tanj.	trainlm	40.352701	1.7458e-05
1	6	0,01	Hiperbolik tanj.	trainbr	51.993761	5.9660e-06
1	6	0,01	ReLU	trainbfg	16.017375	0.0121
1	6	0,01	ReLU	trainlm	14.191954	4.1739e-04
1	6	0,01	ReLU	trainbr	33.103800	3.3560e-04
1	7	0,001	sigmoid	trainbfg	84.466757	1.8967e-04
1	7	0,001	sigmoid	trainlm	66.025221	4.0086e-06
1	7	0,001	sigmoid	trainbr	58.375585	4.9222e-06
1	7	0,001	Hiperbolik tanj.	trainbfg	41.888025	1.0439e-04
1	7	0,001	Hiperbolik tanj.	trainlm	71.039124	5.1680e-06
1	7	0,001	Hiperbolik tanj.	trainbr	60.800913	3.8387e-06
1	7	0,001	ReLU	trainbfg	20.273127	0.0014
1	7	0,001	ReLU	trainlm	13.599247	3.0728e-04
1	7	0,001	ReLU	trainbr	17.546129	2.8748e-04
1	7	0,005	sigmoid	trainbfg	32.170221	2.4066e-04
1	7	0,005	sigmoid	trainlm	71.394049	5.1629e-06
1	7	0,005	sigmoid	trainbr	101.442670	4.4942e-06
1	7	0,005	Hiperbolik tanj.	trainbfg	30.698280	1.3018e-04
1	7	0,005	Hiperbolik tanj.	trainlm	76.040225	4.3993e-06
1	7	0,005	Hiperbolik tanj.	trainbr	74.687681	2.9803e-06

1	7	0,005	ReLU	trainbfg	20.352999	0.0019
1	7	0,005	ReLU	trainlm	16.347489	3.5157e-04
1	7	0,005	ReLU	trainbr	40.648429	7.1458e-04
1	7	0,01	sigmoid	trainbfg	36.698056	2.5786e-04
1	7	0,01	sigmoid	trainlm	75.598654	3.5781e-06
1	7	0,01	sigmoid	trainbr	85.338724	4.5594e-06
1	7	0,01	Hiperbolik tanj.	trainbfg	33.351217	1.0767e-04
1	7	0,01	Hiperbolik tanj.	trainlm	76.915723	4.7556e-06
1	7	0,01	Hiperbolik tanj.	trainbr	88.310925	3.5520e-06
1	7	0,01	ReLU	trainbfg	25.099662	5.0632e-04
1	7	0,01	ReLU	trainlm	21.774107	3.0368e-04
1	7	0,01	ReLU	trainbr	36.453775	1.9477e-04
1	8	0,001	sigmoid	trainbfg	25.938154	4.0026e-04
1	8	0,001	sigmoid	trainlm	59.867673	2.2080e-06
1	8	0,001	sigmoid	trainbr	59.931503	2.3164e-06
1	8	0,001	Hiperbolik tanj.	trainbfg	24.456064	1.1586e-04
1	8	0,001	Hiperbolik tanj.	trainlm	93.472523	5.4505e-06
1	8	0,001	Hiperbolik tanj.	trainbr	112.618560	2.1344e-06
1	8	0,001	ReLU	trainbfg	21.483188	0.0257
1	8	0,001	ReLU	trainlm	17.594053	1.9202e-04
1	8	0,001	ReLU	trainbr	36.241591	2.2861e-04
1	8	0,005	sigmoid	trainbfg	31.980522	1.9766e-04
1	8	0,005	sigmoid	trainlm	88.079554	2.1039e-06
1	8	0,005	sigmoid	trainbr	92.682182	1.8182e-06
1	8	0,005	Hiperbolik tanj.	trainbfg	26.063464	1.8368e-04
1	8	0,005	Hiperbolik tanj.	trainlm	64.294774	2.2528e-06
1	8	0,005	Hiperbolik tanj.	trainbr	112.754028	2.1283e-06
1	8	0,005	ReLU	trainbfg	30.415035	8.2303e-04
1	8	0,005	ReLU	trainlm	22.703319	2.0378e-04
1	8	0,005	ReLU	trainbr	22.244718	2.0659e-04
1	8	0,01	sigmoid	trainbfg	50.693071	1.8353e-04
1	8	0,01	sigmoid	trainlm	67.940277	2.5948e-06
1	8	0,01	sigmoid	trainbr	59.281769	2.1083e-06
1	8	0,01	Hiperbolik tanj.	trainbfg	32.106525	1.5579e-04
1	8	0,01	Hiperbolik tanj.	trainlm	72.534302	3.4941e-06
1	8	0,01	Hiperbolik tanj.	trainbr	105.363301	2.3071e-06
1	8	0,01	ReLU	trainbfg	27.960882	7.3721e-04
1	8	0,01	ReLU	trainlm	28.009259	2.7833e-04
1	8	0,01	ReLU	trainbr	69.579096	2.9170e-04
1	9	0,001	sigmoid	trainbfg	29.675643	3.9388e-04
1	9	0,001	sigmoid	trainlm	56.466667	3.7058e-06
1	9	0,001	sigmoid	trainbr	74.725787	1.6519e-06
1	9	0,001	Hiperbolik tanj.	trainbfg	29.675643	3.9388e-04
1	9	0,001	Hiperbolik tanj.	trainlm	89.570797	3.7815e-06
1	9	0,001	Hiperbolik tanj.	trainbr	102.004710	1.4820e-06
1	9	0,001	ReLU	trainbfg	18.809947	0.0011
1	9	0,001	ReLU	trainlm	14.535851	2.3004e-04
1	9	0,001	ReLU	trainbr	17.320735	1.8552e-04
1	9	0,005	sigmoid	trainbfg	43.043235	1.9677e-04
1	9	0,005	sigmoid	trainlm	96.931945	2.0540e-06
1	9	0,005	sigmoid	trainbr	75.040505	1.8279e-06
1	9	0,005	Hiperbolik tanj.	trainbfg	34.148591	1.2607e-04
1	9	0,005	Hiperbolik tanj.	trainlm	130.779307	1.6778e-06

1	9	0,005	Hiperbolik tanj.	trainbr	130.779307	1.6778e-06
1	9	0,005	ReLU	trainbfg	17.055389	0.0119
1	9	0,005	ReLU	trainlm	17.544753	2.4519e-04
1	9	0,005	ReLU	trainbr	32.773915	1.8680e-04
1	9	0,01	sigmoid	trainbfg	41.383737	1.6790e-04
1	9	0,01	sigmoid	trainlm	88.269612	3.6577e-06
1	9	0,01	sigmoid	trainbr	96.811887	1.5800e-06
1	9	0,01	Hiperbolik tanj.	trainbfg	42.036085	9.5854e-05
1	9	0,01	Hiperbolik tanj.	trainlm	84.930559	2.5959e-06
1	9	0,01	Hiperbolik tanj.	trainbr	103.412030	1.5654e-06
1	9	0,01	ReLU	trainbfg	27.322713	8.6311e-04
1	9	0,01	ReLU	trainlm	22.152982	1.5421e-04
1	9	0,01	ReLU	trainbr	62.769489	2.0405e-04
1	10	0,001	sigmoid	trainbfg	38.148684	1.9507e-04
1	10	0,001	sigmoid	trainlm	80.981761	2.6194e-06
1	10	0,001	sigmoid	trainbr	119.891373	1.3462e-06
1	10	0,001	Hiperbolik tanj.	trainbfg	43.256428	1.2852e-04
1	10	0,001	Hiperbolik tanj.	trainlm	123.218196	1.5782e-06
1	10	0,001	Hiperbolik tanj.	trainbr	119.034545	1.7153e-06
1	10	0,001	ReLU	trainbfg	26.341590	3.6741e-04
1	10	0,001	ReLU	trainlm	25.811130	1.3435e-04
1	10	0,001	ReLU	trainbr	41.884764	1.2961e-04
1	10	0,005	sigmoid	trainbfg	44.359378	4.3146e-04
1	10	0,005	sigmoid	trainlm	89.749320	1.0823e-06
1	10	0,005	sigmoid	trainbr	107.105087	1.4871e-06
1	10	0,005	Hiperbolik tanj.	trainbfg	43.815010	6.9956e-05
1	10	0,005	Hiperbolik tanj.	trainlm	82.313766	1.8262e-06
1	10	0,005	Hiperbolik tanj.	trainbr	89.494654	1.5885e-06
1	10	0,005	ReLU	trainbfg	25.881612	9.1690e-04
1	10	0,005	ReLU	trainlm	27.038486	1.6546e-04
1	10	0,005	ReLU	trainbr	32.663775	1.4375e-04
1	10	0,01	sigmoid	trainbfg	36.305265	3.6740e-04
1	10	0,01	sigmoid	trainlm	75.982962	1.7678e-06
1	10	0,01	sigmoid	trainbr	105.079204	1.5937e-06
1	10	0,01	Hiperbolik tanj.	trainbfg	38.349503	1.1142e-04
1	10	0,01	Hiperbolik tanj.	trainlm	76.714803	1.9237e-06
1	10	0,01	Hiperbolik tanj.	trainbr	91.679286	1.3754e-06
1	10	0,01	ReLU	trainbfg	23.834544	8.0658e-04
1	10	0,01	ReLU	trainlm	24.458275	1.9971e-04
1	10	0,01	ReLU	trainbr	47.667380	2.3195e-04
2	2	0,001	sigmoid	trainbfg	19.107231	0.1520
2	2	0,001	sigmoid	trainlm	15.827821	0.0849
2	2	0,001	sigmoid	trainbr	18.975814	0.1427
2	2	0,001	Hiperbolik tanj.	trainbfg	19.589219	0.0241
2	2	0,001	Hiperbolik tanj.	trainlm	51.889510	0.3098
2	2	0,001	Hiperbolik tanj.	trainbr	53.164697	0.3087
2	2	0,001	ReLU	trainbfg	21.297796	0.3182
2	2	0,001	ReLU	trainlm	17.106627	0.1429
2	2	0,001	ReLU	trainbr	22.614618	0.0928
2	2	0,005	sigmoid	trainbfg	23.132221	0.1435
2	2	0,005	sigmoid	trainlm	24.149031	0.0925
2	2	0,005	sigmoid	trainbr	21.635489	0.1435
2	2	0,005	Hiperbolik tanj.	trainbfg	26.005843	0.0034

2	2	0,005	Hiperbolik tanj.	trainlm	43.783548	0.0136
2	2	0,005	Hiperbolik tanj.	trainbr	56.122521	0.0032
2	2	0,005	ReLU	trainbfg	14.580274	0.2605
2	2	0,005	ReLU	trainlm	12.031075	0.2010
2	2	0,005	ReLU	trainbr	20.201649	0.2590
2	2	0,01	sigmoid	trainbfg	16.640615	0.1061
2	2	0,01	sigmoid	trainlm	14.252713	0.0849
2	2	0,01	sigmoid	trainbr	22.213653	0.0850
2	2	0,01	Hiperbolik tanj.	trainbfg	27.180540	0.0034
2	2	0,01	Hiperbolik tanj.	trainlm	25.874945	0.3115
2	2	0,01	Hiperbolik tanj.	trainbr	68.298744	0.0034
2	2	0,01	ReLU	trainbfg	18.437845	0.2597
2	2	0,01	ReLU	trainlm	13.841708	0.2093
2	2	0,01	ReLU	trainbr	24.746969	0.1500
2	3	0,001	sigmoid	trainbfg	21.834951	0.0930
2	3	0,001	sigmoid	trainlm	24.486883	0.0837
2	3	0,001	sigmoid	trainbr	18.538702	0.0838
2	3	0,001	Hiperbolik tanj.	trainbfg	26.506291	0.0030
2	3	0,001	Hiperbolik tanj.	trainlm	25.999627	0.0018
2	3	0,001	Hiperbolik tanj.	trainbr	53.115997	0.0022
2	3	0,001	ReLU	trainbfg	14.040228	0.2616
2	3	0,001	ReLU	trainlm	16.156657	0.2095
2	3	0,001	ReLU	trainbr	25.396847	0.1421
2	3	0,005	sigmoid	trainbfg	23.234433	0.0931
2	3	0,005	sigmoid	trainlm	34.095276	0.0840
2	3	0,005	sigmoid	trainbr	36.607751	0.0843
2	3	0,005	Hiperbolik tanj.	trainbfg	23.234433	0.0931
2	3	0,005	Hiperbolik tanj.	trainlm	42.571586	0.0014
2	3	0,005	Hiperbolik tanj.	trainbr	38.908897	0.3067
2	3	0,005	ReLU	trainbfg	33.993213	0.1425
2	3	0,005	ReLU	trainlm	14.108809	0.1418
2	3	0,005	ReLU	trainbr	35.590950	0.0833
2	3	0,01	sigmoid	trainbfg	21.965013	0.0885
2	3	0,01	sigmoid	trainlm	24.951453	0.0836
2	3	0,01	sigmoid	trainbr	24.935896	0.0842
2	3	0,01	Hiperbolik tanj.	trainbfg	26.291362	0.0033
2	3	0,01	Hiperbolik tanj.	trainlm	30.405108	0.0024
2	3	0,01	Hiperbolik tanj.	trainbr	38.563131	0.0016
2	3	0,01	ReLU	trainbfg	15.171184	0.3181
2	3	0,01	ReLU	trainlm	14.112501	0.1511
2	3	0,01	ReLU	trainbr	26.706201	0.0833
2	4	0,001	sigmoid	trainbfg	30.803722	0.0925
2	4	0,001	sigmoid	trainlm	25.051852	0.0833
2	4	0,001	sigmoid	trainbr	31.551920	0.0835
2	4	0,001	Hiperbolik tanj.	trainbfg	29.973274	0.0026
2	4	0,001	Hiperbolik tanj.	trainlm	37.055268	0.0011
2	4	0,001	Hiperbolik tanj.	trainbr	50.797254	0.0011
2	4	0,001	ReLU	trainbfg	14.023772	0.3768
2	4	0,001	ReLU	trainlm	17.285262	0.0829
2	4	0,001	ReLU	trainbr	16.394003	0.1415
2	4	0,005	sigmoid	trainbfg	22.902148	0.0848
2	4	0,005	sigmoid	trainlm	43.317966	0.0830
2	4	0,005	sigmoid	trainbr	18.817151	0.1425

2	4	0,005	Hiperbolik tanj.	trainbfg	21.540420	0.0020
2	4	0,005	Hiperbolik tanj.	trainlm	32.393689	0.0012
2	4	0,005	Hiperbolik tanj.	trainbr	67.246232	0.0010
2	4	0,005	ReLU	trainbfg	16.313608	0.3768
2	4	0,005	ReLU	trainlm	15.731158	0.1419
2	4	0,005	ReLU	trainbr	16.689953	0.0831
2	4	0,01	sigmoid	trainbfg	21.314606	0.0978
2	4	0,01	sigmoid	trainlm	35.642545	0.0829
2	4	0,01	sigmoid	trainbr	22.165671	0.1423
2	4	0,01	Hiperbolik tanj.	trainbfg	23.590577	0.0028
2	4	0,01	Hiperbolik tanj.	trainlm	26.529136	0.0015
2	4	0,01	Hiperbolik tanj.	trainbr	61.439162	0.0014
2	4	0,01	ReLU	trainbfg	14.475577	0.2589
2	4	0,01	ReLU	trainlm	13.099250	0.0831
2	4	0,01	ReLU	trainbr	14.137494	0.0830
2	5	0,001	sigmoid	trainbfg	25.974448	0.0844
2	5	0,001	sigmoid	trainlm	24.045358	0.0827
2	5	0,001	sigmoid	trainbr	20.549584	0.0829
2	5	0,001	Hiperbolik tanj.	trainbfg	24.010746	0.0016
2	5	0,001	Hiperbolik tanj.	trainlm	52.419648	6.0158e-04
2	5	0,001	Hiperbolik tanj.	trainbr	57.213482	9.2174e-04
2	5	0,001	ReLU	trainbfg	12.745391	0.2607
2	5	0,001	ReLU	trainlm	12.861151	0.0832
2	5	0,001	ReLU	trainbr	30.254000	0.1425
2	5	0,005	sigmoid	trainbfg	26.262310	0.0838
2	5	0,005	sigmoid	trainlm	35.852752	0.0827
2	5	0,005	sigmoid	trainbr	25.476129	0.0832
2	5	0,005	Hiperbolik tanj.	trainbfg	27.015736	0.0019
2	5	0,005	Hiperbolik tanj.	trainlm	34.319127	4.1615e-04
2	5	0,005	Hiperbolik tanj.	trainbr	68.564586	7.1014e-04
2	5	0,005	ReLU	trainbfg	12.508006	0.3180
2	5	0,005	ReLU	trainlm	12.839442	0.1417
2	5	0,005	ReLU	trainbr	14.940280	0.0828
2	5	0,01	sigmoid	trainbfg	23.376316	0.0845
2	5	0,01	sigmoid	trainlm	20.590646	0.0827
2	5	0,01	sigmoid	trainbr	21.945681	0.0830
2	5	0,01	Hiperbolik tanj.	trainbfg	27.717009	0.0014
2	5	0,01	Hiperbolik tanj.	trainlm	35.220852	3.7774e-04
2	5	0,01	Hiperbolik tanj.	trainbr	65.432342	3.7801e-04
2	5	0,01	ReLU	trainbfg	14.190199	0.3768
2	5	0,01	ReLU	trainlm	14.658430	0.2583
2	5	0,01	ReLU	trainbr	28.469056	0.1414
2	6	0,001	sigmoid	trainbfg	21.608593	0.0836
2	6	0,001	sigmoid	trainlm	31.396553	0.0828
2	6	0,001	sigmoid	trainbr	21.879637	0.0829
2	6	0,001	Hiperbolik tanj.	trainbfg	25.324620	0.0013
2	6	0,001	Hiperbolik tanj.	trainlm	26.310548	5.1240e-04
2	6	0,001	Hiperbolik tanj.	trainbr	51.984923	2.6254e-04
2	6	0,001	ReLU	trainbfg	12.675532	0.2606
2	6	0,001	ReLU	trainlm	14.512527	0.0826
2	6	0,001	ReLU	trainbr	14.788030	0.1415
2	6	0,005	sigmoid	trainbfg	18.411843	0.1424
2	6	0,005	sigmoid	trainlm	21.492430	0.0827

2	6	0,005	sigmoid	trainbr	18.330418	0.0828
2	6	0,005	Hiperbolik tanj.	trainbfg	22.577696	8.6493e-04
2	6	0,005	Hiperbolik tanj.	trainlm	28.704196	3.1396e-04
2	6	0,005	Hiperbolik tanj.	trainbr	56.751924	2.5666e-04
2	6	0,005	ReLU	trainbfg	11.935209	0.3180
2	6	0,005	ReLU	trainlm	13.391086	0.0826
2	6	0,005	ReLU	trainbr	7.509665	0.1413
2	6	0,01	sigmoid	trainbfg	18.660733	0.2014
2	6	0,01	sigmoid	trainlm	12.576456	0.0827
2	6	0,01	sigmoid	trainbr	17.708269	0.0829
2	6	0,01	Hiperbolik tanj.	trainbfg	21.496734	7.9554e-04
2	6	0,01	Hiperbolik tanj.	trainlm	33.342656	3.1669e-04
2	6	0,01	Hiperbolik tanj.	trainbr	50.417646	3.4969e-04
2	6	0,01	ReLU	trainbfg	11.985772	0.3179
2	6	0,01	ReLU	trainlm	13.289353	0.2009
2	6	0,01	ReLU	trainbr	29.425394	0.0827
2	7	0,001	sigmoid	trainbfg	21.218516	0.0838
2	7	0,001	sigmoid	trainlm	40.151503	0.0826
2	7	0,001	sigmoid	trainbr	19.164363	0.0828
2	7	0,001	Hiperbolik tanj.	trainbfg	25.804799	7.8069e-04
2	7	0,001	Hiperbolik tanj.	trainlm	40.022366	1.7961e-04
2	7	0,001	Hiperbolik tanj.	trainbr	62.738323	2.2132e-04
2	7	0,001	ReLU	trainbfg	12.977828	0.3180
2	7	0,001	ReLU	trainlm	4.685835	0.0825
2	7	0,001	ReLU	trainbr	15.566203	0.1417
2	7	0,005	sigmoid	trainbfg	17.158490	0.2006
2	7	0,005	sigmoid	trainlm	51.374178	0.0826
2	7	0,005	sigmoid	trainbr	21.154980	0.0828
2	7	0,005	Hiperbolik tanj.	trainbfg	25.804142	7.0393e-04
2	7	0,005	Hiperbolik tanj.	trainlm	32.367908	2.0948e-04
2	7	0,005	Hiperbolik tanj.	trainbr	62.613320	1.9574e-04
2	7	0,005	ReLU	trainbfg	13.388520	0.3185
2	7	0,005	ReLU	trainlm	13.406665	0.2011
2	7	0,005	ReLU	trainbr	26.845309	0.0826
2	7	0,01	sigmoid	trainbfg	19.826795	0.0938
2	7	0,01	sigmoid	trainlm	52.183115	0.0826
2	7	0,01	sigmoid	trainbr	23.265322	0.1418
2	7	0,01	Hiperbolik tanj.	trainbfg	24.221685	0.0011
2	7	0,01	Hiperbolik tanj.	trainlm	36.583454	2.2660e-04
2	7	0,01	Hiperbolik tanj.	trainbr	60.449410	1.8138e-04
2	7	0,01	ReLU	trainbfg	17.122356	0.3184
2	7	0,01	ReLU	trainlm	17.416432	0.0826
2	7	0,01	ReLU	trainbr	16.349622	0.0826
2	8	0,001	sigmoid	trainbfg	22.195809	0.0836
2	8	0,001	sigmoid	trainlm	41.798217	0.0825
2	8	0,001	sigmoid	trainbr	22.025380	0.0827
2	8	0,001	Hiperbolik tanj.	trainbfg	29.609050	4.8076e-04
2	8	0,001	Hiperbolik tanj.	trainlm	58.826985	1.7565e-04
2	8	0,001	Hiperbolik tanj.	trainbr	22.025380	0.0827
2	8	0,001	ReLU	trainbfg	18.939036	0.3182
2	8	0,001	ReLU	trainlm	14.607096	0.0826
2	8	0,001	ReLU	trainbr	16.494038	0.1414
2	8	0,005	sigmoid	trainbfg	19.634484	0.1416

2	8	0,005	sigmoid	trainlm	35.520325	0.0825
2	8	0,005	sigmoid	trainbr	21.243051	0.0827
2	8	0,005	Hiperbolik tanj.	trainbfg	30.630173	5.1509e-04
2	8	0,005	Hiperbolik tanj.	trainlm	53.451888	2.0564e-04
2	8	0,005	Hiperbolik tanj.	trainbr	78.492913	1.6516e-04
2	8	0,005	ReLU	trainbfg	15.373971	0.2019
2	8	0,005	ReLU	trainlm	14.579196	0.0825
2	8	0,005	ReLU	trainbr	14.659143	0.2013
2	8	0,01	sigmoid	trainbfg	18.273450	0.2015
2	8	0,01	sigmoid	trainlm	54.954788	0.0825
2	8	0,01	sigmoid	trainbr	22.191125	0.0827
2	8	0,01	Hiperbolik tanj.	trainbfg	29.250595	8.3677e-04
2	8	0,01	Hiperbolik tanj.	trainlm	33.965825	1.8012e-04
2	8	0,01	Hiperbolik tanj.	trainbr	68.612134	1.5756e-04
2	8	0,01	ReLU	trainbfg	12.277506	0.3767
2	8	0,01	ReLU	trainlm	17.000727	0.0825
2	8	0,01	ReLU	trainbr	20.341781	0.0826
2	9	0,001	sigmoid	trainbfg	17.332295	0.2006
2	9	0,001	sigmoid	trainlm	47.882569	0.0825
2	9	0,001	sigmoid	trainbr	30.319979	0.1413
2	9	0,001	Hiperbolik tanj.	trainbfg	31.632993	5.2299e-04
2	9	0,001	Hiperbolik tanj.	trainlm	41.809906	1.2065e-04
2	9	0,001	Hiperbolik tanj.	trainbr	67.083300	1.3112e-04
2	9	0,001	ReLU	trainbfg	14.105140	0.2600
2	9	0,001	ReLU	trainlm	13.618118	0.0825
2	9	0,001	ReLU	trainbr	19.942486	0.1417
2	9	0,005	sigmoid	trainbfg	27.700352	0.0832
2	9	0,005	sigmoid	trainlm	58.001362	0.0825
2	9	0,005	sigmoid	trainbr	56.102381	0.0828
2	9	0,005	Hiperbolik tanj.	trainbfg	34.166375	0.0011
2	9	0,005	Hiperbolik tanj.	trainlm	38.008913	1.5196e-04
2	9	0,005	Hiperbolik tanj.	trainbr	83.635249	1.0523e-04
2	9	0,005	ReLU	trainbfg	16.152889	0.1997
2	9	0,005	ReLU	trainlm	15.823000	0.0825
2	9	0,005	ReLU	trainbr	17.857385	0.1423
2	9	0,01	sigmoid	trainbfg	23.697519	0.2016
2	9	0,01	sigmoid	trainlm	54.099092	0.0825
2	9	0,01	sigmoid	trainbr	40.784630	0.0826
2	9	0,01	Hiperbolik tanj.	trainbfg	36.028869	7.9215e-04
2	9	0,01	Hiperbolik tanj.	trainlm	72.031801	1.1412e-04
2	9	0,01	Hiperbolik tanj.	trainbr	83.373971	1.0333e-04
2	9	0,01	ReLU	trainbfg	17.663070	0.2012
2	9	0,01	ReLU	trainlm	14.798614	0.2007
2	9	0,01	ReLU	trainbr	19.752260	0.1411
2	10	0,001	sigmoid	trainbfg	26.830485	0.0831
2	10	0,001	sigmoid	trainlm	56.045977	0.0825
2	10	0,001	sigmoid	trainbr	31.119356	0.0827
2	10	0,001	Hiperbolik tanj.	trainbfg	30.689270	4.3037e-04
2	10	0,001	Hiperbolik tanj.	trainlm	56.932518	1.0699e-04
2	10	0,001	Hiperbolik tanj.	trainbr	78.856241	1.1527e-04
2	10	0,001	ReLU	trainbfg	12.720724	0.3769
2	10	0,001	ReLU	trainlm	13.664411	0.1412
2	10	0,001	ReLU	trainbr	27.805434	0.0826

2	10	0,005	sigmoid	trainbfg	24.589986	0.1432
2	10	0,005	sigmoid	trainlm	32.941203	0.0825
2	10	0,005	sigmoid	trainbr	30.259486	0.0827
2	10	0,005	Hiperbolik tanj.	trainbfg	24.641753	0.0011
2	10	0,005	Hiperbolik tanj.	trainlm	42.187523	1.0061e-04
2	10	0,005	Hiperbolik tanj.	trainbr	70.154579	8.1924e-05
2	10	0,005	ReLU	trainbfg	12.299041	0.3184
2	10	0,005	ReLU	trainlm	17.011851	0.0824
2	10	0,005	ReLU	trainbr	16.290939	0.1410
2	10	0,01	sigmoid	trainbfg	23.143256	0.1420
2	10	0,01	sigmoid	trainlm	43.078668	0.0825
2	10	0,01	sigmoid	trainbr	34.039400	0.1424
2	10	0,01	Hiperbolik tanj.	trainbfg	32.500143	7.5341e-04
2	10	0,01	Hiperbolik tanj.	trainlm	72.461529	6.9789e-05
2	10	0,01	Hiperbolik tanj.	trainbr	83.905118	8.9522e-05
2	10	0,01	ReLU	trainbfg	16.134477	0.3171
2	10	0,01	ReLU	trainlm	16.904335	0.0824
2	10	0,01	ReLU	trainbr	20.844990	0.0825