

**T.C.
MANİSA CELAL BAYAR ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**YÜKSEK LİSANS TEZİ
MAKİNE VE İMALAT MÜHENDİSLİĞİ ANABİLİM DALI
MAKİNE TEORİSİ VE DİNAMİĞİ BİLİM DALI**

**6 SERBESTLİK DERECELİ ROBOT KOLUN TERS KİNEMATİK
ANALİZİNİN YAPAY ZEKA YÖNTEMLERİ İLE
GERÇEKLEŞTİRİLMESİ**

Tuğçe Turgut AKSOY

Danışman

Dr. Öğr. Üyesi Serkan ÇAŞKA

İkinci Danışman

Öğr. Gör. Dr. Hacer ARIOL TAYMAZ



MANİSA-2024

**Tuğçe Turgut
AKSOY**

**6 SERBESTLİK DERECELİ ROBOT KOLUN TERS KİNEMATİK ANALİZİNİN
YAPAY ZEKA YÖNTEMLERİ İLE GERÇEKLEŞTİRİLMESİ**

2024

TAAHHÜTNAME

Bu tezin Manisa Celal Bayar Üniversitesi Hasan Ferdi Turgutlu Teknoloji Fakültesi Makine ve İmalat Mühendisliği Bölümü'nde, akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını beyan ederim.

Tuğçe TURGUT AKSOY



İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER	I
SİMGELER VE KISALTMALAR DİZİNİ	II
ŞEKİLLER DİZİNİ.....	III
TABLO DİZİNİ	IV
TEŞEKKÜR.....	V
ÖZET.....	VI
ABSTRACT.....	VII
1.GİRİŞ	1
2.GENEL BİLGİLER	6
3.MATERYAL VE YÖNTEMLER.....	8
3.1 Kara Dul Örümceği Optimizasyon (KDÖO) Algoritması	11
3.2 Balina Optimizasyon Algoritması (BOA).....	12
3.3 Yusufçuk Algoritması (YA).....	13
4.ARAŞTIRMA BULGULARI VE TARTIŞMA	15
5.SONUÇ VE ÖNERİLER	30
KAYNAKLAR	33
EKLER.....	35
ÖZGEÇMİŞ	38

SİMGELER VE KISALTMALAR DİZİNİ

BD	Bayesçi Düzenleme
BOA	Balina Optimizasyon Algoritması
İK	İleri Kinematik
İYYPSO	İkili Yeniden Yapılanma Parçacık Sürü Optimizasyonu
KAO	Karınca Aslanı Optimizasyonu
KDÖO	Kara Dul Örümceği Optimizasyonu
LM	Levenberg-Marquardt
MS	Meta Sezgisel
OKH	Ortalama Karesel Hata
ÖEG	Ölçeklenmiş Eşlenik Gradyan
PSO	Parçacık Sürü Optimizasyonu
R	Korelasyon Katsayısı
TK	Ters Kinematik
YA	Yusufçuk Algoritması
YAA	Yerçekimi Arama Algoritması
YAK	Yapay Arı Kolonisi
YSA	Yapay Sinir Ağı

ŞEKİLLER DİZİNİ

	Sayfa
Şekil 2.1. Stewart Platformu	7
Şekil 2.2. Seri Manipülatör	7
Şekil 3.1. RV-2SDB Robot Kol	8
Şekil 3.2. RV-2SDB Robot Kol Tutucu Parça Tasarımı.....	9
Şekil 3.3. RV-2SDB Robot Kol Ölçüleri ve Çalışma Uzayı.....	9
Şekil 3.4 Kinematik Şema, Bağlantı Boyutları ve Bağlantı Açılarının Sınırları	11
Şekil 3.5 Ağ İçindeki Tipik Örümcek Hareketi	12
Şekil 3.6. Kambur Balinaların Kabarcık Ağıyla Beslenme Davranışları.....	13
Şekil 3.7. Yusufçuk Algoritma Modeli	14
Şekil 4.1. 10 Hedef Noktası Konum Hatası Grafiği.....	16
Şekil 4.2. 10 Hedef Nokta İçin Çözüm Süresi Grafiği.....	17



TABLO DİZİNİ

	Sayfa
Tablo 1.1 Son Yıllarda Önerilen MS Algoritmaları.....	5
Tablo 4.1. Test Edilen Algoritmaların 20 İterasyon İçin Başarısı	15
Tablo 4.2. Test Edilen Algoritmaların 50 İterasyon İçin Başarısı	15
Tablo 4.3. Mafsal Açısı 1 İçin YSA Modelinin Optimizasyonu.....	18
Tablo 4.4. Mafsal Açısı 2 İçin YSA Modelinin Optimizasyonu.....	20
Tablo 4.5. Mafsal Açısı 3 İçin YSA Modelinin Optimizasyonu.....	22
Tablo 4.6. Mafsal Açısı 4 İçin YSA Modelinin Optimizasyonu.....	24
Tablo 4.7. Mafsal Açısı 5 İçin YSA Modelinin Optimizasyonu.....	26
Tablo 4.8. Mafsal Açısı 6 İçin YSA Modelinin Optimizasyonu.....	28



TEŞEKKÜR

Tez çalışmamda benden desteğini esirgemeyen Sayın. Dr. Öğr. Üyesi Serkan ÇAŞKA'ya, beni bugünlere getiren anneme ve babama, sevgili eşime, kız kardeşime ve bu süreçte çeşitli katkılarıyla yanımda olan herkese teşekkür ederim.

Tuğçe TURGUT AKSOY
Manisa, 2024



ÖZET

Yüksek Lisans Tezi

Tuğçe TURGUT AKSOY

**Manisa Celal Bayar Üniversitesi
Lisans Üstü Eğitim Enstitüsü
Makine ve İmalat Mühendisliği Anabilim Dalı**

**Danışman:
Dr. Öğr. Üyesi Serkan ÇAŞKA**

**İkinci Danışman:
Öğr. Gör. Dr. Hacer ARIOL TAYMAZ**

Ters kinematik (TK) manipülatörün yönelimi ve konumunu dikkate alarak eklem açılarının hesaplanmasıdır ve robotiğin en temel problemlerinden biridir. Meta-sezgisel algoritmalar mühendisliğin birçok alanında karmaşık problemleri çözmek için kullanılmaktadır. Son yıllarda robot manipülatörlerin ters kinematik problemini çözmek için meta-sezgisel algoritmalar uygulanmaktadır. Sürü zekası tabanlı meta-sezgisel algoritmaların kullanımı birçok mühendislik alanında önemli bir artış göstermiştir. Bu çalışmada robotik manipülatörün uç efektörünün konumu, yeni geliştirilen sürü zekâsı tabanlı algoritmalarından Kara Dul Örümceği Optimizasyon Algoritması (KDÖO), Balina Optimizasyon Algoritması (BOA) ve Yusufçuk Algoritması (DA) kullanılarak tahmin edilmiştir. Test edilen algoritmaların başarısını kanıtlamak için ileri kinematik (İK) modelinin girdisi olan 6 eklem açısı değiştirilerek robotik manipülatörün çalışma alanında rastgele seçilen birkaç nokta hesaplanmıştır. Test edilen algoritmalar kullanılarak hedef kartezyen koordinatları sağlayan en uygun açı seti elde edilmiştir. Ayrıca İK modeli kullanılarak deney tablosu oluşturulmuş ve bu tablonun verileri ile yapay sinir ağları modelleri oluşturularak TK modelleri elde edilmiştir.

Anahtar Kelimeler: Ters kinematik, 6-DOF robotik manipülatör, Kara dul optimizasyon algoritması, Balina optimizasyon algoritması, Yusufçuk algoritması, Yapay sinir ağları

2024, 48 sayfa

ABSTRACT

M.Sc. Thesis

Tuğçe TURGUT AKSOY

**Manisa Celal Bayar University
Postgraduate Education Institute
Department of Mechanical and Manufacturing Engineering**

**Supervisor:
Asst. Prof. Dr. Serkan CAŞKA**

**Co-Supervisor:
Lecturer Dr. Hacer ARIOL TAYMAZ**

Inverse kinematics (IK) is the calculation of joint angles taking into account the orientation and position of the manipulator and is one of the most fundamental problems in robotics. Meta-heuristic algorithms are used to solve complex problems in many areas of engineering. In recent years, meta-heuristic algorithms have begun to be applied to solve the inverse kinematics problem of robot manipulators. The use of swarm intelligence (SI)-based metaheuristic algorithms has increased significantly in many engineering fields. In this study, the position of the end effector of the robotic manipulator was estimated using the newly developed SI-based algorithms Black Widow Optimization Algorithm (BWOA), Whale Optimization Algorithm (WOA) and Dragonfly Algorithm (DA). To prove the success of the tested algorithms, a few randomly selected points in the working area of the robotic manipulator were calculated by changing the 6 joint angles that are the input of the forward kinematics (FK) model. Using the tested algorithms, the optimal angle set providing the target Cartesian coordinates was obtained. In addition, an experimental table was created using the IK model, and IK models were obtained by creating artificial neural network models with the data of this table.

Keywords: Inverse kinematics, 6-DOF robotic manipulator, Black widow optimization algorithm, Whale optimization algorithm, Dragonfly algorithm, Artificial neural networks

2024, 48 pages

1.GİRİŞ

Robot kolların ileri ve ters kinematiğini içeren çalışmalar oldukça yaygındır. Papcun ve ark. yaptığı çalışmada, Mitsubishi RV2SDB robot kolunun matematiksel modeli incelenmiş, robot kolunun matematiksel modelini oluşturmanın genel metodolojisi anlatılmıştır. İleri kinematik modelin hesaplanmasına yönelik temel matrisler ve matematiksel model adım adım oluşturulmuştur. Robotun ters kinematiğinde, robot kolun matematiksel modeli ile gerçek robottan deneysel olarak alınan sonuçlar karşılaştırılmıştır [1].

Dereli ve ark. yaptığı çalışmada, serbestlik derecesi yedi olan, bir robot manipülatörü kullanmıştır. Ters kinematik probleminin çözümünü elde etmek için İkili Yeniden Yapılanma Parçacık Sürü Optimizasyonu (İYYPSO) önerilmektedir. Önerilen yöntem, 7 serbestlik dereceli robot manipülatörünün mafsalları açılarının hesaplanmasına fiziksel sınırları dahilinde çözümler üretirken, ters kinematik problemin çözümü için de uygun olduğunu göstermektedir. Ayrıca önerilen yöntemle elde edilen uç efektörün konumsal hatalarının hesaplanmasında YAK, YAA gibi algoritmalar ve PSO 'nun farklı varyantları ile elde edilen sonuçlar karşılaştırılmaktadır. Sonuçlar, önerilen yöntemin literatürde mevcut olan bazı popüler yöntemlerinden daha iyi sonuçlar verdiğini kanıtlamaktadır [2].

Robot manipülatör kinematiğinin temel sorunu olan, ters kinematik hesabı, Danaci ve ark. tarafından incelenmiştir. Analitik ters kinematik çözümlerle sonuca ulaşma mümkün olmadığında Newton Raphson gibi sayısal yöntemler kullanılabilir, ancak hesaplamanın yinelemeli doğası nedeniyle çözüme istenen hızlarda ulaşılamamaktadır. Son yıllarda geliştirilen sürü zekâsı tabanlı optimizasyon yöntemleri aynı zamanda manipülatör ters kinematik çözümlerine de katkıda bulunmaktadır. Literatürde, genel seri robotik manipülatörler için ters kinematik probleminin çözümünde Parçacık Sürü Optimizasyonu (PSO) yaklaşımı da kullanılmıştır. Sürü zekâsı yöntemlerini içeren robotik manipülatör ters kinematik çözümlerinin çoğu, robotun yönelimiyle değil, yalnızca uç efektör konumuyla ilgilenmiştir [3].

Robotikte karmaşık ve zor problem gruplarından biri de ters kinematik çözümdür. Dereli ve ark., yaptıkları çalışmada literatürde önemli bir yeri bulunan ters kinematik çözümünün bulunması için sürü algoritmalarını kapsamlı bir şekilde araştırmışlardır. Eğer bir robot kolunun çok fazla sayıda mafsallı bulunuyorsa, bu durum çok daha zor bir hal almaktadır. Doğadaki hayvanların davranışlarından esinlenen sürü optimizasyon algoritmaları, bu nedenle araştırmacılar tarafından çokça tercih edilmektedir. Sürü zekasına dayalı optimizasyon algoritmaları, parçacık çözüm uzayındaki en iyi sonuçları elde etmektedirler. Literatürde sıklıkla kullanılan sürü zekâsı algoritmaları vardır. Bu algoritmalar, yapay arı kolonisi, ateş böceği algoritması ve parçacık sürü algoritması robotik alanındaki birçok çalışmada kullanılan yöntemlerdir [4].

Constantin ve ark. yaptığı çalışmada, 6 serbestlik dereceli bir robot kolunun tam İK modelini türetmiştir. Bir robot manipülatörünü çalışmasının gerektirdiği şekilde kontrol edebilmek için, kontrol algoritmasının tasarımında kinematik modeli dikkate alınmıştır [5].

Garip Z. yaptığı çalışmada kaotik tabanlı bir algoritma geliştirmiştir. Yapılan çalışmada 7 serbestlik dereceli bir seri robot manipülatörü kullanılmıştır. Robot manipülatörünün, ters kinematik problemini çözebilmek, çözüm karmaşıklığını minimuma indirebilmek için Henon kaotik harita ile desteklenmiş ve tekrar düzenlenmiş olan çiçek tozlaşma algoritması kullanılmıştır. Geliştirilen algoritma ile orijinal algoritmanın genel ve bölgesel arama özelliklerinin artırılması amaçlanmıştır. Simülasyon çalışmaları ile, düzenlenmiş algoritmaların performansını doğrulamak için üç farklı konum vektörü seçilmiştir. Ayrıca araştırma sonuçları, kullanılan algoritmaların uç işlevcisinin konum hatası ve çözüm süresi kriterlerine göre karşılaştırılmıştır. Simülasyon sonuçları, seri robot manipülatörünün ters kinematik çözümleri için kaotik tabanlı algoritmaların verimli bir şekilde kullanılabileceğini kanıtlamıştır [6].

Toz M. yaptığı çalışmada, altı serbestlik dereceli bir seri robot manipülatörü kullanmıştır. TK problemi Karınca Aslanı Optimizasyon algoritması (KAOA) ile çözülmüştür. Analitik olarak çözülmesi en zor problemlerden biri ters kinematik problemidir. Seri robotlar için ve bu problemin çözümü robotun mafsallı yapısı ve

türleriyle yakından ilişkilidir. Algoritmanın performansını test etmek amacıyla rastgele seçilmiş beş çalışma uzayı noktası için ters kinematik çözüm gerçekleştirilmiştir. Elde edilen sonuçlar konumlanma hatası ve çözüm süresi açısından değerlendirilmiştir. KAOA algoritmasının her iki değerlendirme kriterine göre de seri robotların ters kinematik probleminin çözümünde verimli bir şekilde kullanılabileceği gösterilmiştir [7].

Robot kollarının endüstriyel otomasyon sistemlerine entegrasyonunun artması ve robotik sistemlerin yeteneklerinin artması nedeniyle robotik alanında kullanılan algoritmaların daha hızlı ve verimli hesaplama yapmalarını gerektirmiştir. Seven ve ark. yaptığı çalışmada, IRB120 robot kolunun ters kinematik problemine kapalıformda bir çözüm sunmuştur. Bu amaçla robotun İK modeli Denavit-Hartenberg gösterimi kullanılarak formüle edilmiş ve TK modelin geometrik yaklaşımla analitik çözümü elde edilmiştir [8].

Endüstriyel robotlarda en zorlu problemlerden biri TK problemini çözmektir. TK problemi, istenilen kartezyen konuma ilişkin mafsalsal açı değerlerinin bulunmasıyla ilgilidir. Son yıllarda TK probleminin çözümünde sayısal, geometrik ve cebirsel gibi geleneksel yöntemleri kullanmak yerine yeterli doğrulukta ve yüksek hızda çözüm sağlayan yapay zekâ ve makine öğrenmesi tabanlı yöntemlerin tercih edilmektedir. El-Sherbiny ve ark. yaptığı çalışmada, TK problemine uygulanan farklı hesaplama tabanlı yöntemler (Yapay Sinir Ağı, Adaptif Nöro Bulanık Çıkarım Sistemi ve Genetik Algoritmalar) arasında karşılaştırmalı bir çalışma sunmuştur [9].

Literatürde, YSA ve Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistemi gibi yöntemler robotların ters kinematik modelinin elde edilmesinde kullanılmıştır. 5 serbestlik dereceli robot kol kullanılarak yapılan testler sonucunda bu yöntemlerin diğer yöntemlere göre daha iyi performans gösterebildiği kanıtlanmıştır [9]. Yapay zekanın önemli bir dalı olan sürü zekâsı diğer pek çok şeyin yanı sıra, doğadaki sosyal sürülerin akıllı kolektif davranışları üzerine inşa edilmiştir. Mittal ve ark. yaptığı çalışmada, en popüler sürü zekâsı yöntemlerinden biri olan Parçacık Sürü Optimizasyonu algoritması (PSO) incelenmiştir [10].

Literatürde, robot kinematiğinde PSO yönteminin birçok farklı versiyonu kullanılmıştır. Ele alınan her yöntemin tartışılan bazı avantajları ve kaçınılmaz dezavantajları bulunmaktadır. Buna göre PSO tabanlı yöntemlerin zayıf yönlerini ele almak ve algoritmaya ilişkin açık konuları ve gelecekteki araştırma perspektiflerini vurgulamak için ipuçları sunan çalışmalar mevcuttur [10]. PSO yönteminin yaygınlaşmasından bu yana PSO'nun birçok yeni versiyonları araştırmacılar tarafından türetilmiştir. PSO algoritmasının çeşitli parametrelerinin ve yönlerinin mühendislik problemlerinin çözümüne potansiyel etkisi üzerine çeşitli teorik ve uygulama içeren çalışmalar yayınlanmıştır. PSO ile ilgili açık konular ve geleceğe yönelik perspektifler dahil olmak üzere mevcut ve devam eden araştırmalara ilişkin çeşitli perspektifler incelenmiştir. Diğer algoritmalarla çeşitli yöntemler kullanılarak hibridizasyon gibi iyileştirme gibi seçenekler üzerinde durulmuştur.

Bu çalışmada öncelikle ucunda uç işlevcisi bulunan Mitsubishi RV-2AJ robot manipülatörünün, İK modeli türetilmiştir. Bir dizi kartezyen koordinat, hedef konumlar olarak tanımlanmıştır. İK modelin girdisi olan 6 eklem açısı değiştirilerek hedef kartezyen koordinatları sağlayan optimal açı seti elde edilmiştir. Robotik manipülatörün uç işlevcisinin konumu, son yıllarda geliştirilen sürü zekâsı tabanlı MS algoritmalar olan KDÖO, BOA ve YA kullanılarak tahmin edilmiştir. Algoritmaların başarısı, çözüm süreleri ve çözüm hataları dikkate alınarak tartışılmıştır.

Optimizasyon, olası çözümlerin bir koleksiyonundan en uygun olanı bulmak olarak tanımlanabilir. Meta sezgisel algoritmalar, büyük ölçekli optimizasyon problemlerini çözmek için optimuma yakın sonuçlar sağlamaktadır. MS algoritmaları mühendislik problemlerini optimize etmek için etkili araçlar haline gelmiştir. Literatürde MS algoritmaları birçok mühendislik probleminin çözümüne önemli katkılar sağlamıştır. Son zamanlarda geliştirilen birçok MS algoritması bulunmaktadır. MS algoritmaları geniş bir aileye aittir ancak MS algoritmalarının çoğu sürü tabanlıdır. Son 10 yılda geliştirilen MS algoritmaları Tablo 1'de verilmiştir.

Tablo 1. Son Yıllarda Önerilen MS Algoritmalar

Meta Sezgisel (MS) Algoritma	Tip
Hayvan Göç Optimizasyonu	Sürü Tabanlı
Gri Kurt Optimizasyon algoritması	Sürü Tabanlı
Yıldırım Arama Algoritması	Doğa Bazlı
Aslan Optimizasyon Algoritması	Doğa Bazlı
Çekirge Optimizasyon Algoritması	Sürü Tabanlı
Sosyal Mühendislik Optimizasyonu	İnsan Temelli
Harris Şahini Optimizasyonu	Doğa Bazlı
Siyasi Optimizasyon	İnsan Temelli
Afrika Akbabaları Optimizasyon Algoritması	Sürü Tabanlı
Ateş Şahini Optimizasyonu	Doğa Bazlı
Büyüme Optimizasyonu	İnsan Temelli

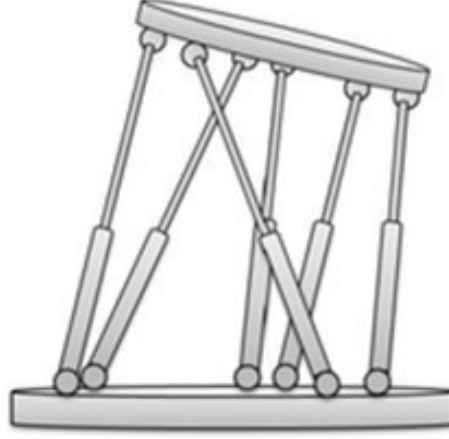
2.GENEL BİLGİLER

Endüstri 4.0 ve 5.0 gelişimiyle birlikte robot manipülatörlerin üretim açısından önemi daha da önem kazanmıştır [11], [12]. Sanayi, insan hayatına faydalı hızlı, güvenli üretimi hedeflemektedir. Teknolojideki gelişmeler sosyal yaşamın gereksinimleriyle birlikte artmaktadır.

Robot manipülatörler üretim, kaynak, savunma teknolojisi ürünleri, otomotiv sanayi gibi birçok alanda kullanıldıkları için insanlık açısından oldukça önemlidir. [13]. Bu manipülatörler, üretim tesisleri, fabrikalar sayesinde; Daha hızlı, güvenilir, hassas ve daha kaliteli ürünler ürettiği için sektör açısından oldukça önemlidir. Ayrıca bu manipülatörler çalışma ortamına göre hareketli veya sabit olarak yerleştirilerek sektörde oldukça önemli bir yere sahiptir [14] [15].

Ancak Endüstri 4.0 ve 5.0 için robot manipülatörlerin otonom olarak hareket ettirilebilmesi için kinematik denklem parametrelerinin çok doğru belirlenmesi gerekmektedir. [16], [17]. Ancak elektronik olarak külfetli olan yoğun matematiksel işlemlerle bu kinematik denklemlerin anında elde edilmesi büyük bir zorluktur.

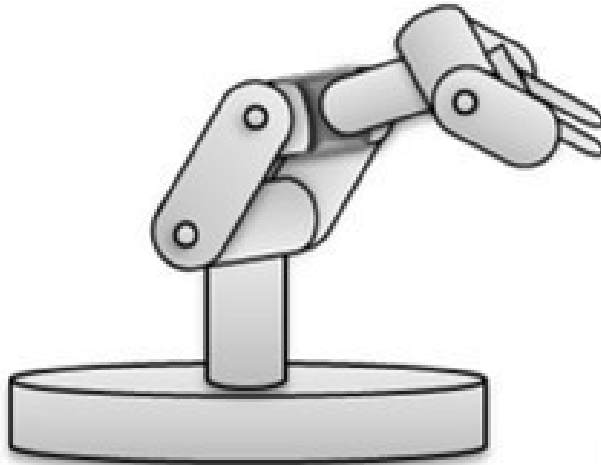
Robotik alanında, paralel manipülatörlere olan ilgi son zamanlarda artmaktadır. Seri manipülatörlere göre paralel manipülatörlerin üstünlükleri, yüksek doğruluk, yüksek sağlamlık, yüksek hızda çalışma ve yüksek yük kapasitesine sahip olmalarıdır. Bundan dolayı, paralel manipülatörler birçok alanda tercih edilmektedir. Örneğin hava taşıtı simülatörleri, mikro konumlandırma, hafif metal işleme, tıbbi operasyonlar ve petrol platformları gibi birçok uygulamada yer almaktadır. En yaygın paralel manipülatörlerden birisi de Şekil 2.1’de gösterilen Stewart Platformu’dur [18].



Şekil 2.1.Stewart Platformu [18]

Paralel robotlar, kapalı çevrimli yapılarıdır. Hareketli platformu sabit platforma en az iki noktadan birbirine bağlarlar. Birbirinden bağımsız, kinematik bağlantı elemanları kullanılır.

Seri robotların çalışma uzayları çok geniştir. Paralel robotlara göre az sayıda mekanik parçaya sahiptirler. Seri robotların kinematik denklemleri ve yörünge planlama denklemleri paralel robotlara göre daha az karmaşıktır. Bununla seri robotların yüklenme oranları paralel robotlara oranla düşüktür. Seri robotların mekanik aksamaları seri bir hat şeklindedir. Seri robotların hata oranları bu seri hat üzerinden toplanarak gider. Bu nedenle, paralel robotlara göre hassasiyetleri ve kararlılıkları daha düşüktür.



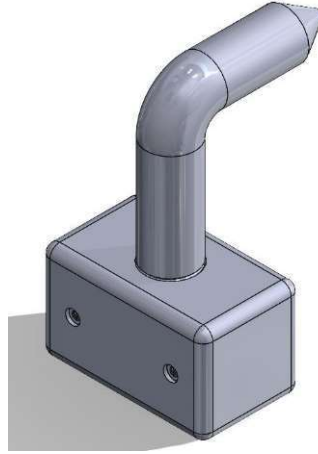
Şekil 2.2.Seri Manipülatör [18]

3.MATERYAL VE YÖNTEMLER

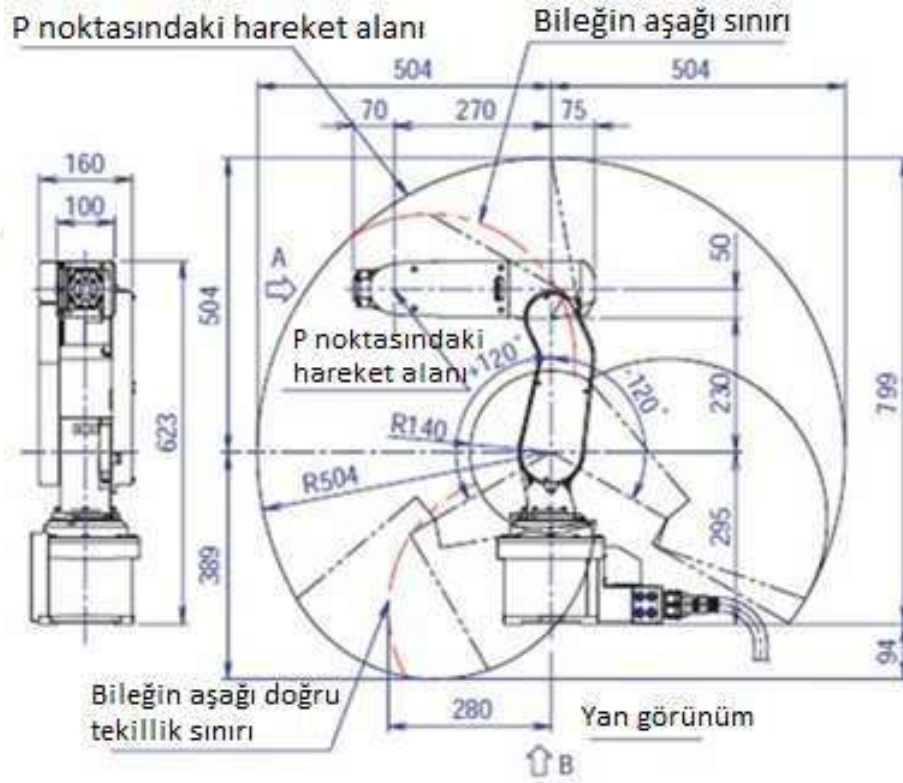
Bu tez çalışmasında ele alınan robot manipölatör 6 serbestlik derecelidir. 6. serbestlik derecesi robot kolun bilek kısmındadır ve burada meydana gelen açı değişimleri robotun uç noktasının kartezyen koordinatlarında bir değişime sebep olmamaktadır. Robot kolun bilek mafsalına bir parça eklenerek bilek mafsalında gerçekleşecek hareket sonucunda da kartezyen koordinatlarda bir değişim olması sağlanmıştır.



Şekil 3.1.RV-2SDB Robot Kol



Şekil 3.2. RV-2SDB Robot Kol Tutucu Parça Tasarımı



Şekil 3.3. RV-2SDB Robot Kol Ölçüleri ve Çalışma Uzayı

$$A = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.1)$$

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & b \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.2)$$

$$C = \begin{pmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.3)$$

$$D = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.4)$$

$$E = \begin{pmatrix} \cos\gamma & 0 & \sin\gamma & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\gamma & 0 & \cos\gamma & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.5)$$

$$F = \begin{pmatrix} 1 & 0 & 0 & -d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.6)$$

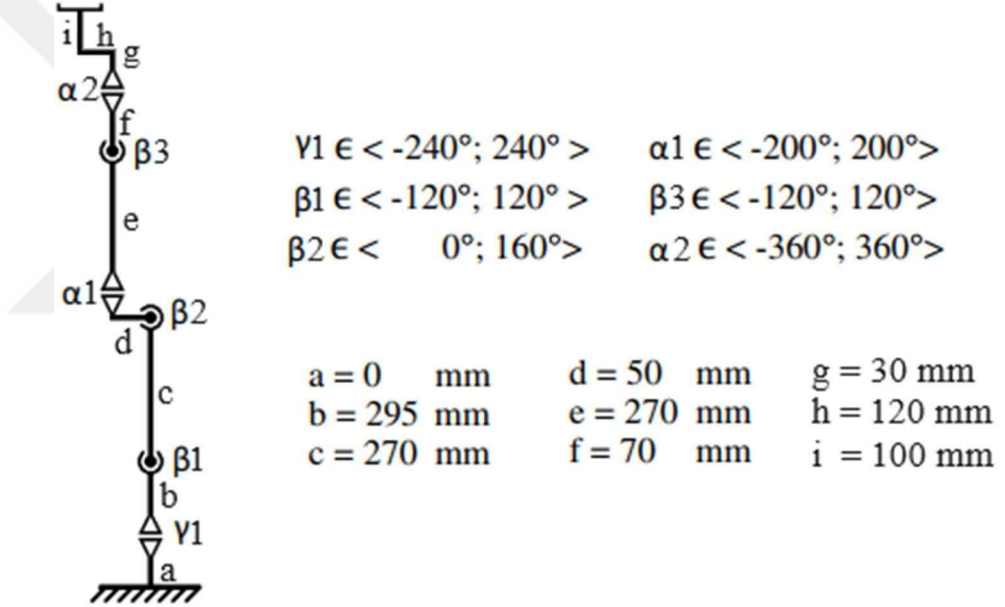
$$G = \begin{pmatrix} \cos\delta & -\sin\delta & 0 & 0 \\ \sin\delta & \cos\delta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.7)$$

$$H = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & e \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.8)$$

$$I = \begin{pmatrix} \cos \varepsilon & 0 & \sin \varepsilon & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \varepsilon & 0 & \cos \varepsilon & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.9)$$

$$J = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & f \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.10)$$

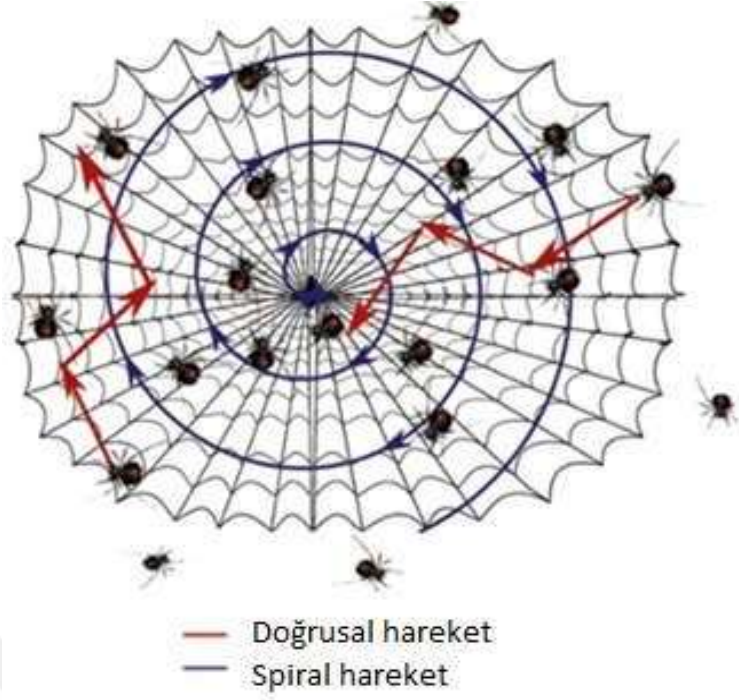
$$K = \begin{pmatrix} \cos \varepsilon & 0 & \sin \varepsilon & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \varepsilon & 0 & \cos \varepsilon & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.11)$$



Şekil 3.4. Kinematik Şema, Bağlantı Boyutları ve Bağlantı Açılarının Sınırları

3.1 Kara Dul Örümceği Optimizasyonu (KDÖO) Algoritması

KDÖO algoritması, kara dul örümceklerinin eşleşme davranışlarından ilham alınarak geliştirilmiştir. Bu algoritmayı literatürdeki diğer sürü zekâsı algoritmalarından ayıran özelliği, aynı türden başka bir bireyin tüketilme aşaması bulunmaktadır. Bu aşamadan dolayı uygunluk değeri uygun olmayan türlerçemberden çıkarılır, bu davranış erken yakınsamaya neden olur. Spiral ve doğrusal arama yaklaşımı Şekil 3.5'te gösterilmektedir.

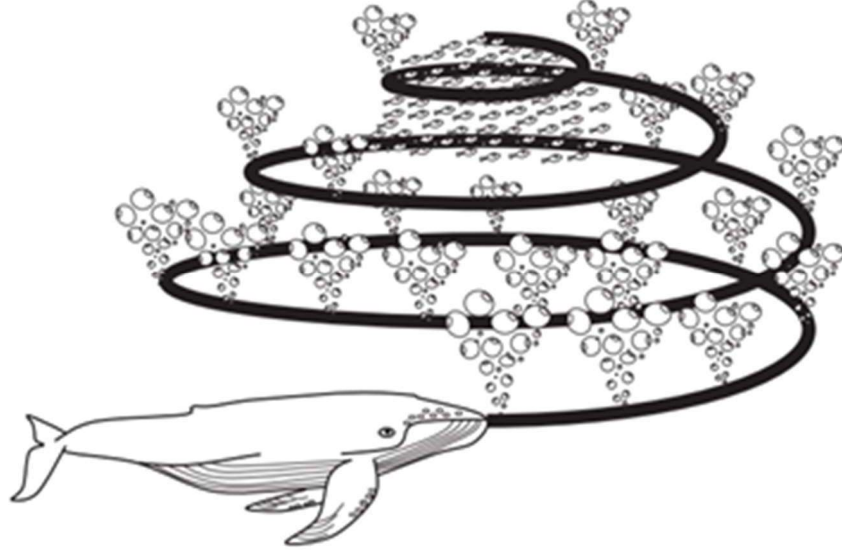


Şekil 3.5. Ağ İçindeki Tipik Örümcek Hareketi

3.2 Balina Optimizasyon Algoritması (BOA)

BOA, kambur balinaların avlanma davranışlarını taklit etmektedir. Bu algoritma, kabarcık ağı avlama stratejisinden ilham alınarak geliştirilmiştir. Kambur balinalar, okyanus yüzeyine yakın kril veya küçük balık sürülerini hedef alma eğilimindedir, Şekil 3.6’te gösterildiği gibi genellikle dairesel veya '9' şeklindeki desenleri takip eden bir yiyecek arama stratejisidir.

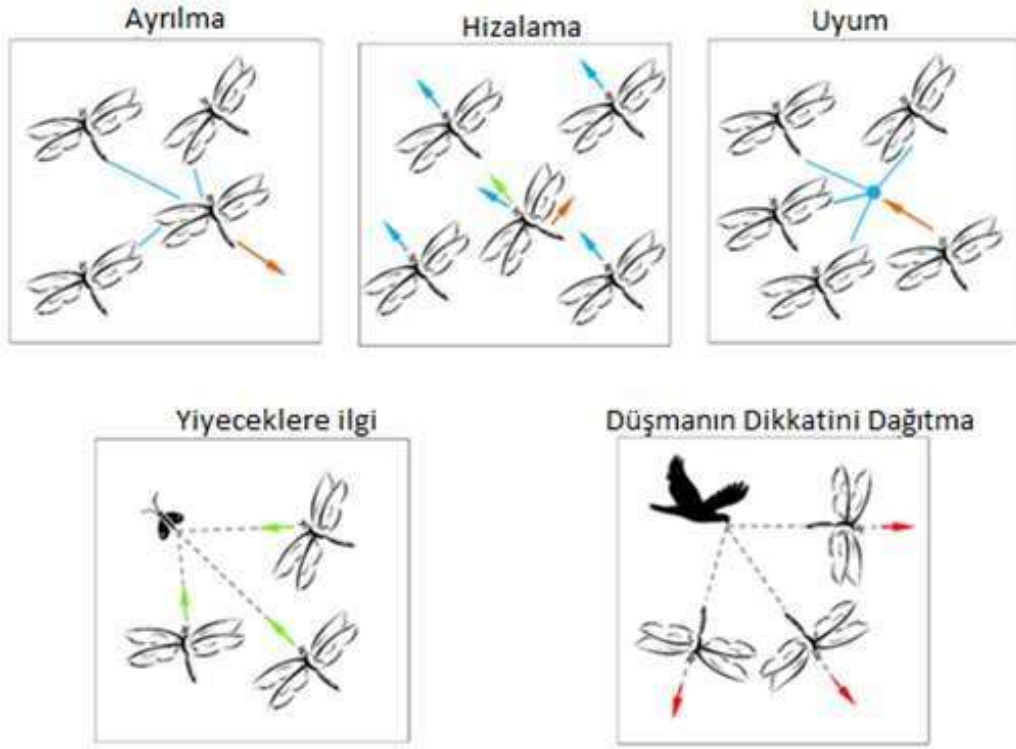
BOA bağlamında, kambur balinaların kabarcık ağıyla beslenme davranışlarının iki yaklaşım (Küçülen Çevreleme ve Spiral Konum Güncelleme) kullanılarak modellenmesi Şekil 3.6’te gösterilmektedir.



Şekil 3.6. Kambur Balinaların Kabarcık Ağıyla Beslenme Davranışları

3.3 Yusufçuk Algoritması (YA)

YA, yusufçukların statik ve dinamik sürü davranışına dayanan yeni bir sürü zekâsı optimizasyon yöntemidir. YA' da optimizasyonun keşif ve yararlanma olmak üzere, iki aşaması bulunmaktadır. Bu aşamalar, yusufçukların yön bulma, yiyecek arama ve düşmanlardan kaçınma konusundaki sosyal etkileşimi modellenerek tasarlanmıştır. Ayırma, bireylerle ortamdaki diğer bireyler arasındaki çarpışmaların statik olarak önlenmesiyle ilgilidir. Hizalanma, bireylerin hızının ortamdaki diğer bireylerin hızıyla eşleştirilmesi anlamına gelir. Uyum, bireylerin ortam içindeki kütle merkezine doğru hareket etme eğilimidir. Yiyeceğe çekim, yiyeceğe yaklaşma veya uzanma eylemidir. Düşmanın dikkatini dağıtmak, yusufçuğa gelebilecek herhangi bir potansiyel zararın oluşturduğu tehdidi en aza indirmeyi içerir. Yusufçuk algoritması modeli Şekil 3.7’de verilmiştir.



Şekil 3.7. Yusuřuk Algoritma Modeli

4.ARAŞTIRMA BULGULARI VE TARTIŞMA

Bu çalışmada hedef konumlar olarak bir dizi kartezyen koordinat tanımlanmıştır. İK modelinde 6 eklem açısı değiştirilerek hedef kartezyen koordinatlara yakınsayan optimal açı seti elde edilmiştir. Robotik manipülatörün uç efektörünün konumu KDÖO, BOA ve YA kullanılarak tahmin edilmiştir. Algoritmaların popülasyon büyüklüğü 50 olarak tanımlandı. Tablo 4.1 ve Tablo 4.2, test edilen algoritmaların sırasıyla 20 ve 50 yinelemedeki başarısını göstermektedir.

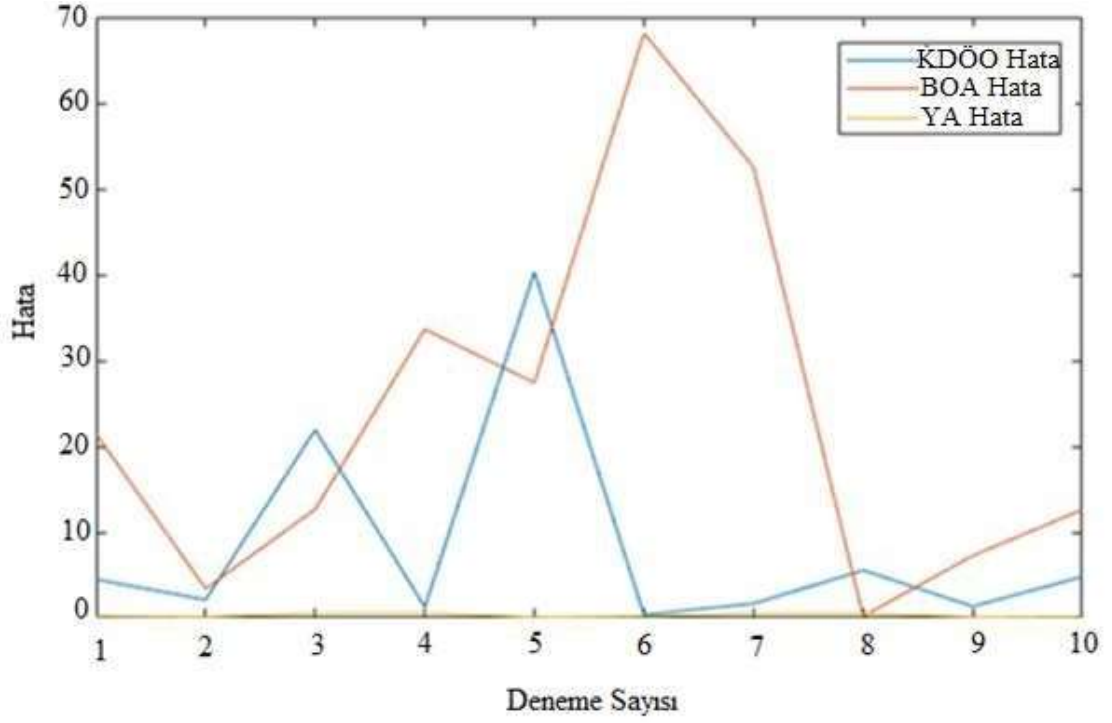
Tablo 4.1. Test edilen algoritmaların 20 iterasyon için başarısı

Algoritma	Hedef Pozisyon (x, y, z)	Hata(mm)	Açılar (Derece)	Çalışma süresi (Saniye)
KDÖO	[250, -450, 505]	23.333	[40.217,-45.078, 50.007, 64.086, 76.144, 148.219]	148.379
BOA	[250, -450, 505]	61.030	[-140.163,-158.942,-18.732, -108.376, 15.021, 29.3410]	137.533
YA	[250, -450, 505]	4.805	[-103.338,-166.438,-10.725, 29.911, 120.000, 157.275]	146.627

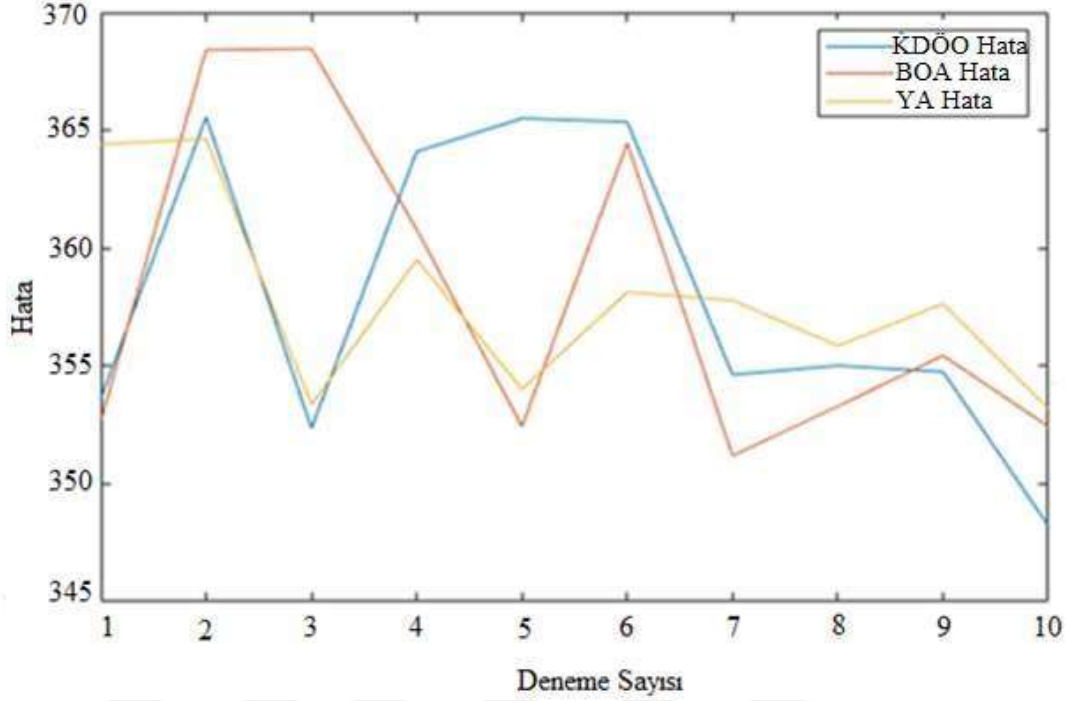
Tablo 4.2 Test edilen algoritmaların 50 iterasyon için başarısı

Algoritma	Hedef Pozisyon (x, y, z)	Hata(mm)	Açılar (Derece)	Çalışma süresi (Saniye)
KDÖO	[242, 266, -94]	84.518	[-73.4287, 50.0000,-3.9311, 130.5065,-7.2683, 350.0000]	356.935
BOA	[242, 266, -94]	21.441	[-27.069, 40.261, 17.186, 153.206,-38.824, -234.709]	352.672
YA	[242, 266, -94]	0.207	[-33.601, 50.000,-14.986, -175.000,-76.652, -260.595]	364.410

Tablo 4.1 ve Tablo 4.2'de verilen sonuçları kanıtlamak için, test edilen algoritmalar kullanılarak çalışma alanında rastgele seçilen 10 nokta hesaplanmıştır. Şekil 4.1 ve Şekil 4.2, sırasıyla, rastgele seçilen 10 hedef nokta için konum hatasını ve çözüm süresini göstermektedir. Tablo 4.2, maksimum yineleme sayısı 50'ye eşitken, 1 mm'nin altında hata elde etmenin mümkün olduğunu göstermektedir. Bu nedenle Şekil 4.1 ve Şekil 4.2'de gösterilen veriler elde edilirken test edilen algoritmaların maksimum iterasyon sayısı 50 olarak tanımlandı.



Şekil 4.1: 10 Hedef Nokta İçin Konum Hatası Grafiği



Şekil 4.2: 10 Hedef Noktası İçin Çözüm Süresi Grafiği

Taguchi yöntemi, bir sistemin veya sürecin performansını etkileyen kontrol edilebilir faktörleri (tasarım parametreleri) belirlemek ve optimize etmek için yapılandırılmış bir yaklaşımı izleyen bir tür deneysel tasarım yöntemidir. Yöntem, sistem performansını etkileyebilecek kontrol edilemeyen faktörlerdeki (gürültü faktörleri) değişikliklere karşı daha az duyarlı, sağlam tasarımlar oluşturmayı amaçlamaktadır. Taguchi yöntemi, bir sürecin hedef değerinden sapmasını ölçmek için bir kalite kaybı fonksiyonu kullanır. Bu işlev, istenen performanstan sapmalarla ilişkili kaybın ölçülmesine yardımcı olur. Taguchi tasarımları genellikle gerekli deneysel çalışma sayısını en aza indirmek için ortogonal diziler kullanır. Bu tez çalışmasında, yapay sinir ağlarının (YSA) optimizasyonunda Taguchi'nin L27 ortogonal dizisi kullanılmıştır. Tablo 4.3'te Mafsal açısı 1 için, Tablo 4.4'te Mafsal açısı 2 için, Tablo 4.5'te Mafsal açısı 3 için, Tablo 4.6'da Mafsal açısı 4 için, Tablo 4.7'de Mafsal açısı 5 için ve Tablo 4.8'de Mafsal açısı 6 için oluşturulan YSA modellerinin optimizasyonuna ait sonuçlar verilmiştir.

Tablo 4.3. Mafsal açısı 1 için YSA modelinin optimizasyonu

Model numarası	Sinir hücresi sayısı	Eğitim fonksiyonu	Eğitim verisi oranı	Eğitim verisi için OKH, R değerleri
1	10	LM	60	2.0943e-04, 0.9997
2	10	LM	70	1.7506e-04, 0.9997
3	10	LM	80	9.7525e-05, 0.9998
4	10	BD	60	0.0017, 0.9972
5	10	BD	70	6.6169e-05, 0.9999
6	10	BD	80	5.6037e-05, 0.9999
7	10	ÖEG	60	0.0027, 0.9956
8	10	ÖEG	70	0.0018, 0.9970
9	10	ÖEG	80	0.0059, 0.9903
10	20	LM	60	2.1610e-04, 0.9996
11	20	LM	70	5.0989e-05, 0.9999
12	20	LM	80	8.3091e-05, 0.9999
13	20	BD	60	6.8583e-04, 0.9989
14	20	BD	70	2.6394e-04, 0.9996

15	20	BD	80	5.5161e-04, 0.9991
16	20	ÖEG	60	0.0054, 0.9911
17	20	ÖEG	70	0.0030, 0.9951
18	20	ÖEG	80	0.0052, 0.9914
19	30	LM	60	8.8825e-05, 0.9999
20	30	LM	70	7.3406e-05, 0.9999
21	30	LM	80	7.7860e-05, 0.9999
22	30	BD	60	4.0199e-05, 0.9999
23	30	BD	70	2.0898e-04, 0.9997
24	30	BD	80	7.2897e-05, 0.9999
25	30	ÖEG	60	0.0143, 0.9764
26	30	ÖEG	70	0.0134, 0.9779
27	30	ÖEG	80	0.0139, 0.9770

Tablo 4.4. Mafsal açısı 2 için YSA modelinin optimizasyonu

Model numarası	Sinir hücresi sayısı	Eğitim fonksiyonu	Eğitim verisi oranı	Eğitim verisi için OKH, R değerleri
1	10	LM	60	0.0289, 0.9514
2	10	LM	70	0.0303, 0.9489
3	10	LM	80	0.0295, 0.9504
4	10	BD	60	0.0296, 0.9502
5	10	BD	70	0.0341, 0.9423
6	10	BD	80	0.0286, 0.9518
7	10	ÖEG	60	0.0420, 0.9286
8	10	ÖEG	70	0.0399, 0.9323
9	10	ÖEG	80	0.0418, 0.9289
10	20	LM	60	0.0251, 0.9580
11	20	LM	70	0.0255, 0.9573
12	20	LM	80	0.0262, 0.9561
13	20	BD	60	0.0281, 0.9528
14	20	BD	70	0.0266, 0.9554

15	20	BD	80	0.0280, 0.9529
16	20	ÖEG	60	0.0424, 0.9279
17	20	ÖEG	70	0.0421, 0.9284
18	20	ÖEG	80	0.0343, 0.9420
19	30	LM	60	0.0257, 0.9570
20	30	LM	70	0.0234, 0.9608
21	30	LM	80	0.0256, 0.9571
22	30	BD	60	0.0268, 0.9550
23	30	BD	70	0.0257, 0.9569
24	30	BD	80	0.0253, 0.9576
25	30	ÖEG	60	0.0474, 0.9191
26	30	ÖEG	70	0.0401, 0.9319
27	30	ÖEG	80	0.0483, 0.9175

Tablo 4.5. Mafsal açısı 3 için YSA modelinin optimizasyonu

Model numarası	Sinir hücresi sayısı	Eğitim fonksiyonu	Eğitim verisi oranı	Eğitim verisi için OKH, R değerleri
1	10	LM	60	0.1219, 0.7746
2	10	LM	70	0.1174, 0.7840
3	10	LM	80	0.1197, 0.7793
4	10	BD	60	0.1221, 0.7743
5	10	BD	70	0.1353, 0.7459
6	10	BD	80	0.1244, 0.7691
7	10	ÖEG	60	0.1872, 0.6210
8	10	ÖEG	70	0.1862, 0.6237
9	10	ÖEG	80	0.1857, 0.6248
10	20	LM	60	0.1048, 0.8100
11	20	LM	70	0.0978, 0.8241
12	20	LM	80	0.1070, 0.8056
13	20	BD	60	0.1089, 0.8015
14	20	BD	70	0.1025, 0.8146

15	20	BD	80	0.1155, 0.7884
16	20	ÖEG	60	0.1153, 0.7884
17	20	ÖEG	70	0.1863, 0.6256
18	20	ÖEG	80	0.1359, 0.7442
19	30	LM	60	0.1074, 0.8045
20	30	LM	70	0.0970, 0.8257
21	30	LM	80	0.1046, 0.8103
22	30	BD	60	0.1048, 0.8100
23	30	BD	70	0.1141, 0.7910
24	30	BD	80	0.1046, 0.8103
25	30	ÖEG	60	0.1443, 0.7255
26	30	ÖEG	70	0.1673, 0.6732
27	30	ÖEG	80	0.1776, 0.6463

Tablo 4.6: Mafsal açısı 4 için YSA modelinin optimizasyonu

Model numarası	Sinir hücresi sayısı	Eğitim fonksiyonu	Eğitim verisi oranı	Eğitim verisi için OKH, R değerleri
1	10	LM	60	0.0797, 0.8593
2	10	LM	70	0.0798, 0.8592
3	10	LM	80	0.0803, 0.8582
4	10	BD	60	0.0958, 0.8282
5	10	BD	70	0.0883, 0.8428
6	10	BD	80	0.0801, 0.8585
7	10	ÖEG	60	0.0794, 0.8600
8	10	ÖEG	70	0.1514, 0.7102
9	10	ÖEG	80	0.1350, 0.7468
10	20	LM	60	0.1350, 0.7468
11	20	LM	70	0.0698, 0.8779
12	20	LM	80	0.0702, 0.8773
13	20	BD	60	0.0763, 0.8658
14	20	BD	70	0.0709, 0.8759

15	20	BD	80	0.0708, 0.8762
16	20	ÖEG	60	0.1291, 0.7601
17	20	ÖEG	70	0.1360, 0.7442
18	20	ÖEG	80	0.1455, 0.7229
19	30	LM	60	0.0640, 0.8887
20	30	LM	70	0.0639, 0.8888
21	30	LM	80	0.0653, 0.8864
22	30	BD	60	0.0789, 0.8614
23	30	BD	70	0.0641, 0.8886
24	30	BD	80	0.0626, 0.8914
25	30	ÖEG	60	0.1207, 0.7773
26	30	ÖEG	70	0.1442, 0.7265
27	30	ÖEG	80	0.1439, 0.7265

Tablo 4.7: Mafsal açısı 5 için YSA modelinin optimizasyonu

Model numarası	Sinir hücresi sayısı	Eğitim fonksiyonu	Eğitim verisi oranı	Eğitim verisi için OKH, R değerleri
1	10	LM	60	0.0317, 0.9465
2	10	LM	70	0.0312, 0.9474
3	10	LM	80	0.0329, 0.9445
4	10	BD	60	0.0332, 0.9439
5	10	BD	70	0.0374, 0.9366
6	10	BD	80	0.0322, 0.9457
7	10	ÖEG	60	0.0621, 0.8926
8	10	ÖEG	70	0.0448, 0.9236
9	10	ÖEG	80	0.0458, 0.9218
10	20	LM	60	0.0278, 0.9532
11	20	LM	70	0.0301, 0.9493
12	20	LM	80	0.0286, 0.9519
13	20	BD	60	0.0280, 0.9529

14	20	BD	70	0.0276, 0.9537
15	20	BD	80	0.0284, 0.9523
16	20	ÖEG	60	0.0310, 0.9479
17	20	ÖEG	70	0.0339, 0.9427
18	20	ÖEG	80	0.0360, 0.9390
19	30	LM	60	0.0258, 0.9568
20	30	LM	70	0.0269, 0.9549
21	30	LM	80	0.0305, 0.9486
22	30	BD	60	0.0258, 0.9568
23	30	BD	70	0.0269, 0.9547
24	30	BD	80	0.0269, 0.9548
25	30	ÖEG	60	0.0554, 0.9046
26	30	ÖEG	70	0.0638, 0.8891
27	30	ÖEG	80	0.0415, 0.9293

Tablo 4.8: Mafsal açısı 6 için YSA modelinin optimizasyonu

Model numarası	Sinir hücresi sayısı	Eğitim fonksiyonu	Eğitim verisi oranı	Eğitim verisi için OKH, R değerleri
1	10	LM	60	0.0775, 0.8636
2	10	LM	70	0.0784, 0.8616
3	10	LM	80	0.0789, 0.8609
4	10	BD	60	0.0783, 0.8620
5	10	BD	70	0.0783, 0.8620
6	10	BD	80	0.0795, 0.8596
7	10	ÖEG	60	0.1049, 0.8097
8	10	ÖEG	70	0.1330, 0.7510
9	10	ÖEG	80	0.1330, 0.7510
10	20	LM	60	0.0683, 0.8808
11	20	LM	70	0.0671, 0.8831
12	20	LM	80	0.0653, 0.8863
13	20	BD	60	0.0663, 0.8845
14	20	BD	70	0.0669, 0.8834

15	20	BD	80	0.0689, 0.8797
16	20	ÖEG	60	0.1327, 0.7513
17	20	ÖEG	70	0.1274, 0.7634
18	20	ÖEG	80	0.1259, 0.7664
19	30	LM	60	0.0643, 0.8883
20	30	LM	70	0.0645, 0.8877
21	30	LM	80	0.0635, 0.8896
22	30	BD	60	0.0638, 0.8892
23	30	BD	70	0.0737, 0.8707
24	30	BD	80	0.0642, 0.8885
25	30	ÖEG	60	0.0971, 0.8258
26	30	ÖEG	70	0.0854, 0.8484
27	30	ÖEG	80	0.1278, 0.7622

5.SONUÇ VE ÖNERİLER

Bu çalışmada, öncelikle ucunda üç boyutlu parça bulunan Mitsubishi RV-2SDB robot manipülatörünün ileri kinematik modeli elde edilmiştir. Robotik manipülatörün uç efektörünün konumu ters kinematik çözümü kapsamında KDÖO, BOA ve YA yöntemleri kullanılarak tahmin edilmiştir. Bir dizi kartezyen koordinat, hedef konum olarak tanımlanmıştır. İleri kinematik modelinin girdisi olan, 6 mafsallık açısı değiştirilerek hedef kartezyen koordinatları sağlayan optimal açı seti elde edilmiştir.

Tablo 4.1'de optimizasyon algoritmalarının 20 iterasyon için çalıştırılması durumunda elde edilen sonuçlar verilmiştir. Buna göre, BOA en düşük çözüm süresine ve YA en düşük hataya sahiptir. Bu çalışmada kabul edilebilir maksimum hata 1 mm olarak tanımlanmıştır. Bu nedenle 20 iterasyon için test edilen algoritmaların sonuçlarının hiçbirisi kabul edilebilir değildir. Daha iyi sonuçlar elde etmek için algoritmaların maksimum iterasyon sayısı 50 olarak tanımlanmış ve Tablo 4.2'teki sonuçlar bu koşulda elde edilmiştir. Tablo 4.2'ye göre BOA en düşük çözüm süresine ve YA en düşük hataya sahiptir.

Algoritmanın maksimum iterasyon sayısı 50 iken sonuçları elde edilen Şekil 4.1'de YA sonuçlarının ortalaması 0,308 iken KDÖO ve BOA sonuçlarının ortalaması sırasıyla 8,391 ve 23,927'dir. Şekil 4.1, dikkate alındığında YA sonuçlarının konum hatası açısından KDÖO ve BOA sonuçlarına göre daha iyi olduğu görülmektedir. Ayrıca YA, KDÖO ve BOA sonuçlarına göre daha stabil sonuçlar sağlamıştır. Şekil 4.2 'de YA ile elde edilen sonuçların ortalaması 357.855 s iken KDÖO ve BOA ile elde edilen sonuçların ortalaması sırasıyla 357.918 s ve 357.959 s'dir. Şekil 4.2'ye bakıldığında algoritmaların sonuçlarının birbirine çok yakın olduğu ve test edilen algoritmaların çalışma süreleri açısından anlamlı bir fark elde edilmediği görülmüştür.

Tablo 4.3'e göre mafsallık açısı 1 için en iyi modelleme başarısı gizli katmandaki nöron sayısı 20, eğitim fonksiyonu ÖEG ve eğitim verisi oranı 80 olduğunda elde edilmiştir. Bu durumda eğitim verisinin OKH ve R değerleri, 0.0052, 0.9914 olarak elde edilmiştir.

Tablo 4.4'e göre mafsalsal açısı 2 için en iyi modelleme başarısı gizli katmandaki nöron sayısı 30, eğitim fonksiyonu LM ve eğitim verisi oranı 70 olduğunda elde edilmiştir. Bu durumda eğitim verisinin OKH ve R değerleri, 0.0234,0.9608 olarak elde edilmiştir.

Tablo 4.5'e göre mafsalsal açısı 3 için en iyi modelleme başarısı gizli katmandaki nöron sayısı 30, eğitim fonksiyonu LM ve eğitim verisi oranı 70 olduğunda elde edilmiştir. Bu durumda eğitim verisinin OKH ve R değerleri, 0.0970,0.8257 olarak elde edilmiştir.

Tablo 4.6'e göre mafsalsal açısı 4 için en iyi modelleme başarısı gizli katmandaki nöron sayısı 30, eğitim fonksiyonu BD ve eğitim verisi oranı 80 olduğunda elde edilmiştir. Bu durumda eğitim verisinin OKH ve R değerleri, 0.0626,0.8914 olarak elde edilmiştir.

Tablo 4.7'e göre mafsalsal açısı 5 için en iyi modelleme başarısı gizli katmandaki nöron sayısı 30, eğitim fonksiyonu BD ve eğitim verisi oranı 60 olduğunda elde edilmiştir. Bu durumda eğitim verisinin OKH ve R değerleri, 0.0258,0.9568 olarak elde edilmiştir.

Tablo 4.8'e göre mafsalsal açısı 6 için en iyi modelleme başarısı gizli katmandaki nöron sayısı 30, eğitim fonksiyonu LM ve eğitim verisi oranı 80 olduğunda elde edilmiştir. Bu durumda eğitim verisinin OKH ve R değerleri, 0.0635,0.8896 olarak elde edilmiştir.

Literatürde robotik manipülatörlerin TK çözümünde MS algoritmalar kullanılmıştır. Bu çalışma, yakın zamanda geliştirilen üç MS algoritmaya dayalı çözümleri içermektedir. Bu çalışmada elde edilen çözüm süresi literatürdeki ilgili çalışmalara göre oldukça yüksektir. Bu çalışmada programlama ortamı olarak MATLAB yazılımı kullanılmış ve algoritmaların kaynak kodları kullanılmıştır. Parçacık sürüşü algoritmasının atalet ağırlık stratejisi veya çözüm süresini kısaltan kuantum davranışlı yapı gibi ek adımlar bu çalışmada kullanılmamıştır ve bu da yüksek çözüm süresinin elde edilmesinin temel nedenidir. Robot manipülatörlerin ters kinematik çözümü için MS yaklaşımı içerecek ilerideki bir çalışmanınkapsamına yeni algoritmaların denenmesi, algoritmaların popülasyon büyüklüğünün

değiştirilmesi ve algoritmaların çözüm süresini kısıltacak ek adımların entegre edilmesi dahil edilebilir.

Tablo 4.3'ten Tablo 4.8'e kadar olan tablolarda yer alan OKH ve R değerleri incelendiğinde YSA modelleri kullanılarak 6 serbestlik dereceli bir robotun ters kinematik modelinin elde edilebildiği görülmektedir. Bu tez çalışmasına sığ YSA modelleri kullanılmıştır ve her mafsal açısı için bir YSA modeli oluşturulmuştur. Derin sinir ağları kullanılarak ve 6 mafsal açısına elde edebilecek tek bir YSA modeli de elde edilebilir ve daha düşük OKH değerleri elde edilebilir.



KAYNAKLAR

- [1] Papcun P., Jadlovský, J., Mathematical Model of Robot Melfa RV- 2SDB, Department of Cybernetics and Artificial Intelligence (DCAI), Faculty of Electrical Engineering and Informatics (FEEI), Technical University of Košice, Letná 9, 042 00 Košice , Slovakia, 2015
- [2] Dereli, S, Köker R., A meta-heuristic proposal for inverse kinematics solution of 7-DOF serial robotic manipulator: quantum behaved particle swarm algorithm. *Artif Intell* (2015) Rev. 53 949–964
- [3] Danaci H., Thomas L., Harman and Miguel P., Inverse Kinematics for Serial Robot Manipulators by Particle Swarm Optimization and POSIX Threads Implementation, *Journal of Applied Sciences* 2023, Volume 13, Issue 7, 4515.
- [4] Dereli S., Köker R., Öylek İ. ve Ay M., A comprehensive research on the use of swarm algorithms in the inverse kinematics solution, *Politeknik Dergisi*, 2019, 22(1): 75-79,
- [5] Constantin D., Lupoae M., Baciuc C., Buliga D., Forward Kinematic Analysis of an Industrial Robot. In: *New Developments in Mechanics and Mechanical Engineering*, 2015 pp. 90 – 95. Vienna, Austria
- [6] Garip Z, Seri Robot Manipulatoründe Ters Kinematik Problemi Çözmek İçin Kaotik Tabanlı Çiçek Tozlaşma Algoritmasının Uygulanması, *Iğdır Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 2022,12(1): 80-90.
- [7] Toz M., Inverse Kinematic Solution Of A 6 Dof Serial Robot Manipulator With Offset Wrist By Using ALO Algorithm, *Symposium Sigma J Eng & Nat Sci* 8 (2), 2017, 81-90
- [8] Seven O., Ankaralı A., 2 Inverse Kinematic Analysis of IRB120 Robot Arm, *SETSCI Conference Proceedings*, 4(1):383-390, 2019
- [9] El-Sherbiny A, Elhosseini MA, Haikal AY, A comparative study of soft computing methods to solve inverse kinematics problem, *Ain Shams Engineering Journal*, 2018 9(4):2535-2548.
- [10] Mittal, A., Pattnaik, A., and Tomar, A. Different variants of particle swarm optimization algorithms and its application, A review. *Metaheuristic and Evolutionary Computation: Algorithms and Applications*, 2021 , 131 163.
- [11] Oztemel, E., Gursev S., Literature review of Industry 4.0 and related Technologies, *J Intell Manuf*, 31(1): 127-182 (2020), doi: 10.1007/S10845-018-1433-8
- [12] Coronado, E., Kiyokawa, T., Ricardez, G. A. G., Ramirez-Alpizar, I. G., Venture, G., Yamanobe, N.: Evaluating quality in human-robot interaction: A

systematic search and classification of performance and human-centered factors, measures and metrics towards an industry 5.0. *J Manuf Syst* 63: 392–410 (2022). doi: 10.1016/J.JMSY.2022.04.007

[13] Singh, G., Banga, V. K., Robots and its types for industrial applications, *Mater Today Proc* 60: 1779-1786 (2022). doi: 10.1016/J.MATPR.2021.12.426

[14] Balanji, H. M., Turgut, A. E., Tunc, L. T., A novel vision-based calibration framework for industrial robotic manipulators, *Robot Comput Integr Manuf.* 73(102248): 1-15 (2022). doi: 10.1016/J.RCIM.2021.102248.

[15] Priyadarshana, V. P. S. C., Yaparathne, Y. M. P. G. R. R. Y., Jayasooriya, D. B. S., Thennakoon, T. M. T. S. D. B., Jayasekara, A. G. B. P., Chandima, D. P., Voice Controlled Robot Manipulator for Industrial Applications, 2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) 160-165 (2022). doi: 10.1109/IEMCON56893.2022.9946545

[16] Ayten, K. K., Dumlu, A., Real-Time Implementation of Sliding Mode Control Technique for Two-DOF Industrial Robotic Arm , *Journal of the Institute of Science and Technology* 8(4): 77-85 (2018). doi: 10.21597/JIST.396344

[17] Reinhold, J., Baumann, H., Meurer, T., Constrained-Differential-Kinematics-Decomposition-Based NMPC for Online Manipulator Control with Low Computational Costs, *Robotics* 12(7) (2023). doi: 10.3390/ROBOTICS12010007

[18] Limtrakul S., Arnonkijpanich B., Supervised learning based on the self organizing maps for forward kinematic modeling of Stewart platform, *Neural Comput Applic* (2019) 31:619-635 DOI 10.1007/s00521-017-3095-4

EKLER

Amaç fonksiyonu:

```
function [lb,ub,dim,fobj] = Get_Functions_details(F)
```

```
switch F
```

```
case 'F1'
```

```
fobj = @F1;
```

```
lb=[-170 -170 -110 -175 -120 -350];
```

```
ub=[ 170 50 155 175 120 350];
```

```
dim=6;
```

```
end
```

```
end
```

```
% F1
```

```
function o = F1(x)
```

```
% o=sum(x.^2);
```

```
x=x*pi/180;
```

```
syms T a alfa d H1 D3 D4 teta Q1 Q2 Q3 Q4 Q5 Q6 T1 T2 T3 T4 T5 T6 I1 I2 I3  
T51
```

```
Q1=x(1);
```

```
Q2=x(2);
```

```
Q3=x(3);
```

```
Q4=x(4);
```

```
Q5=x(5);
```

```
Q6=x(6);
```

```
%% 0. nokta
```

$m0r = [\cos(0) \ -\sin(0) \ 0 \ 0; \sin(0) \ \cos(0) \ 0 \ 0; 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1];$ %dönmüyor

$m0t = [1 \ 0 \ 0 \ 0; 0 \ 1 \ 0 \ 0; 0 \ 0 \ 1 \ 180; 0 \ 0 \ 0 \ 1];$

%% 1. nokta z'de dönüyor

$m1r = [\cos(Q1) \ -\sin(Q1) \ 0 \ 0; \sin(Q1) \ \cos(Q1) \ 0 \ 0; 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1];$ %z'de dönüyor

$m1t = [1 \ 0 \ 0 \ 0; 0 \ 1 \ 0 \ 0; 0 \ 0 \ 1 \ 165; 0 \ 0 \ 0 \ 1];$

%% 2. nokta

$m2r = [\cos(0) \ 0 \ \sin(0) \ 0; 0 \ 1 \ 0 \ 0; -\sin(0) \ 0 \ \cos(0) \ 0; 0 \ 0 \ 0 \ 1];$ %dönmüyor

$m2t = [1 \ 0 \ 0 \ 20; 0 \ 1 \ 0 \ 0; 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1];$

%% 3. nokta %y'de dönüyor

$m3r = [\cos(Q2) \ 0 \ \sin(Q2) \ 0; 0 \ 1 \ 0 \ 0; -\sin(Q2) \ 0 \ \cos(Q2) \ 0; 0 \ 0 \ 0 \ 1];$ %y'de dönüyor

$m3t = [1 \ 0 \ 0 \ 260; 0 \ 1 \ 0 \ 0; 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1];$

%% 4. nokta %y'de dönüyor

$m4r = [\cos(Q3) \ 0 \ \sin(Q3) \ 0; 0 \ 1 \ 0 \ 0; -\sin(Q3) \ 0 \ \cos(Q3) \ 0; 0 \ 0 \ 0 \ 1];$ %y'de dönüyor

$m4t = [1 \ 0 \ 0 \ 0; 0 \ 1 \ 0 \ 0; 0 \ 0 \ 1 \ 20; 0 \ 0 \ 0 \ 1];$

%% 5. nokta

$m5r = [\cos(0) \ 0 \ \sin(0) \ 0; 0 \ 1 \ 0 \ 0; -\sin(0) \ 0 \ \cos(0) \ 0; 0 \ 0 \ 0 \ 1];$ %dönmüyor

$m5t = [1 \ 0 \ 0 \ 130; 0 \ 1 \ 0 \ 0; 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1];$

%% 6. nokta x'te dönüyor

$m6r = [1 \ 0 \ 0 \ 0; 0 \ \cos(Q4) \ -\sin(Q4) \ 0; 0 \ \sin(Q4) \ \cos(Q4) \ 0; 0 \ 0 \ 0 \ 1];$ %x'te dönüyor

$m6t = [1 \ 0 \ 0 \ 130; 0 \ 1 \ 0 \ 0; 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1];$

%% 7. nokta y'de dönüyor

$m7r = [\cos(Q5) \ 0 \ \sin(Q5) \ 0; 0 \ 1 \ 0 \ 0; -\sin(Q5) \ 0 \ \cos(Q5) \ 0; 0 \ 0 \ 0 \ 1];$ %y'de dönüyor

$m7t = [1 \ 0 \ 0 \ 75; 0 \ 1 \ 0 \ 0; 0 \ 0 \ 1 \ 0; 0 \ 0 \ 0 \ 1];$

%% 8. nokta x'te dönüyor

m8r=[1 0 0 0;0 cos(Q6) -sin(Q6) 0;0 sin(Q6) cos(Q6) 0;0 0 0 1]; %x'te dönüyor

m8t=[1 0 0 -100;0 1 0 0;0 0 1 0;0 0 0 1];

%% 9. nokta

m9r=[cos(0) -sin(0) 0 0;sin(0) cos(0) 0 0;0 0 1 0;0 0 0 1]; %dönüyor

m9t=[1 0 0 0;0 1 0 0;0 0 1 100;0 0 0 1];

%% İLERİ KİNEMATİK DENKLEMİ

T=m0r*m0t*m1r*m1t*m2r*m2t*m3r*m3t*m4r*m4t*m5r*m5t*m6r*m6t*m7r*m7t
*m8r*m8t*m9r*m9t

hedef=[242 266 -94];%[255 -455 510];%[254 -454 509];%[253 -453 508];%[252 -
452 507];%[241 265 -93];%[251 -451 506];%[240 264 -92];%[239 263 -91];%[250 -
450 505];%

error=sqrt((hedef(1)-T(1,4))^2 + (hedef(2)-(-1)*T(2,4))^2 + (hedef(3)-T(3,4))^2);

o=error

x=x*180/pi

global iter

iter=iter

end