

**T. C.
İSTANBUL GELİŞİM ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

Mekatronik Mühendisliği Anabilim Dalı

**6 SERBESTLİK DERECESİNE SAHİP SİSTEMİN
YAZILIMSAL VE DONANIMSAL OLARAK
OPTİMİZASYONU**

Yüksek Lisans Tezi

Ömer ŞAHİN

Danışman
Prof. Dr. Bedri YÜKSEL

İstanbul – 2023

TEZ TANITIM FORMU

Yazar Adı Soyadı : Ömer ŞAHİN

Tezin Dili : Türkçe

Tezin Adı : 6 Serbestlik Derecesine Sahip Paralel Manipülatör Sisteminin Simülasyon Ortamında Yazılımsal Olarak Optimizasyonu ve Analizi

Enstitü : İstanbul Gelişim Üniversitesi Lisansüstü Eğitim Enstitüsü

Anabilim Dalı : Mekatronik Mühendisliği

Tezin Türü : Yüksek Lisans

Tezin Tarihi : 24.01.2023

Sayfa Sayısı : 173

Tez : Prof. Dr. Bedri YÜKSEL

Danışmanları

Dizin Terimleri : Çalışma uzayı analizi, Altı eksen serbestliği, Stewart Platform Sistemi, Paralel robot sistemleri, Java 3B simülasyonu

Türkçe Özet : Bu çalışmada java programlama dilinde bir masaüstü arayüz uygulaması ile görsel simülasyon yapılmış, farklı Stewart platformların çalışma uzaylarının incelenmesi ve bu platformların geliştirilmesi amaçlanmıştır.

Dağıtım Listesi : 1. İstanbul Gelişim Üniversitesi Lisansüstü Eğitim Enstitüsüne
2. YÖK Ulusal Tez Merkezine

İmzası

Ömer ŞAHİN

T. C.
İSTANBUL GELİŞİM ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

Mekatronik Mühendisliği Anabilim Dalı

6 SERBESTLİK DERECESİNE SAHİP SİSTEMİN
YAZILIMSAL VE DONANIMSAL OLARAK
OPTİMİZASYONU

Yüksek Lisans Tezi

Ömer ŞAHİN

Danışman
Prof. Dr. Bedri YÜKSEL

İstanbul – 2023

BEYAN

Bu tezin hazırlanmasında bilimsel ahlak kurallarına uyulduđu, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduđu, kullanılan verilerde herhangi tahrifat yapılmadığını, tezin herhangi bir kısmının bu üniversite veya başka bir üniversitedeki başka bir tez olarak sunulmadığını beyan ederim.

Ömer ŞAHİN

.../.../2023



İSTANBUL GELİŞİM ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Ömer ŞAHİN'in 6 Serbestlik Derecesine Sahip Paralel Manipülator Sisteminin Simülasyon Ortamında Yazılımsal Olarak Optimizasyonu ve Analizi adlı tez çalışması, jürimiz tarafından Mekatronik Mühendisliği anabilim dalı, Mekatronik Mühendisliği bilim dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

İmza

Başkan

Prof. Dr. Bedri YÜKSEL
(Danışman)

İmza

Üye

Prof. Dr. Hamdi Alper ÖZYİĞİT

İmza

Üye

Doç. Dr. Armağan Fatih KARAMANLI

ONAY

Yukarıdaki imzaların, adı geçen öğretim üyelerine ait olduğunu onaylarım.

... / ... / 2023

İmzası

Prof. Dr. İzzet GÜMÜŞ

Enstitü Müdürü

ÖZET

Paralel mekanizmalar, günümüzde askeri, cerrahi ve endüstriyel alanlarda yaygın olarak kullanılmaktadır. Paralel mekanizmalı sistemler, genel itibari ile en az iki çalışma uzayı üzerinde bulunan rijit cisimlerin belirlenmiş koordinat merkezleri arasında çalışan kinematik zincirlerdir. Örnek olarak, Stewart Platform Mekanizmasının (SPM) 6 serbestlik derecesi bulunmaktadır. Stewart Platform Mekanizmaları genellikle sabit alt plaka ve hareketli üst plaka arasında değiştirilebilir uzunlukta kolları içeren yapılardır. Bu çalışmada, kol uzunluklarını ve kolların bağlantı noktaları arasındaki mesafeyi değiştirerek sistemi farklı çalışma uzaylarında hareket ettirebilecek bir modele dayalı yazılımsal prototip oluşturulmuş ve çalışma uzayı incelenmiştir. Tez çalışmasında prototipi geliştirilen 3B modelin platform hareketlerine göre, piston hareketlerinin pozisyonları ters kinematik yöntemler kullanılarak hesaplanmıştır. Ters kinematik yöntemler, bir sistemdeki son işleyicinin konumunu ve sistemdeki eyleyicilerin bu konuma göre pozisyonlarını hesaplayan, başlangıç koşullarını bulmaya yarayan bir yöntemdir. Bu hesaplamalar sonucunda, masaüstü GUI uygulaması (Java programlama dilinde yazılmış bir arayüz uygulaması) kullanılarak bir görsel simülasyon oluşturulmuştur. Oluşturulan yazılım ile istenilen parametrelerde ayarlanan SPM için nokta bulutu metodu kullanılarak çalışma uzayı analizi yapılmıştır. Ayrıca oluşturulan yazılım istenilen açısal ve doğrusal hareketleri arayüz üzerinden kontrol etmeye olanak tanır. Bu simülasyonun JAVA programlama dilinde oluşturulmasında JZY3D adlı bir kütüphane kullanılmıştır. Bu kütüphane, 3B grafikler oluşturmaya ve bu grafikleri görselleştirmeye yardımcı olan bir araçtır. Bu çalışmada stewart platform mekanizmasının tasarım parametrelerindeki değişikliklerin çalışma uzayı şekli ve hacmine olan etkisi nokta bulutu yöntemi kullanılarak incelenmiş. Örneklem olarak alınan platform tasarım parametleri simülasyon yazılımına girdi olarak verilmiş ve nokta bulutu oluşturulmuştur. Tez çalışma içerisinde görsel ve sayısal sonuçlar alınmıştır.

Anahtar Kelimeler: Çalışma uzayı analizi, Altı eksen serbestliği, Stewart Platform Sistemi, Paralel robot sistemleri, Java 3B simülasyonu

SUMMARY

Nowadays, Parallel mechanisms are widely used in the field of military, surgical and industry. Parallel mechanism systems are generally kinematic chains that operate between the predetermined coordinate centers of rigid bodies located on at least two workspaces. As an example, Stewart Platform Mechanism has 6 degrees of freedom. Stewart Platform Mechanisms are usually the structures that consist of arms with interchangeable length between the stationary lower-plate and mobilized upper-plate. In this study, a software prototype based on a model that can move the system in different workspaces by changing the arm lengths and changing the distance between the junction points of the arms was created and the workspace was examined. In this thesis-study, according to the movements of the developed prototype of the 3D model, the position of the piston movements are calculated by using inverse kinematics methods. Inverse kinematic methods, is a method that helps to find the conditions in a system by calculating the end effector's location and the actuators' location relative to the end effector's location. As a result of the calculations, a visual simulation is created by using GUI desktop application (an interface application that is programmed in JAVA programming language). Working space analysis was carried out using the point cloud method for the SPM, which was adjusted in the desired parameters with the created software. In addition, the software created allows controlling the desired angular and linear movements via the GUI. To create this simulation in JAVA programming language, a library named JZY3D was used. This library is a tool that helps to create and visualize 3D graphics. In this study, the effect of changes in the design parameters of the Stewart platform mechanism on the shape and volume of the workspace was investigated by using the point cloud method. The platform design parameters taken as a sample were given as input to the simulation software and a point cloud was created. Visual and numerical results were obtained in the thesis study.

Keywords: Workspace analysis, 6 Degrees of Freedom, Stewart Platform Mechanism, Parallel Robot Mechanisms, Java 3D simulation

İÇİNDEKİLER

ÖZET.....	i
SUMMARY	ii
İÇİNDEKİLER	iii
KISALTMALAR	vi
TABLolar LİSTESİ.....	vii
ŞEKİLLER LİSTESİ.....	x
EKLER LİSTESİ	xiii
ÖNSÖZ.....	xiv
GİRİŞ	1

BİRİNCİ BÖLÜM GENEL BİLGİLER

1.1. Literatür Taraması	4
1.2. Mekanizma Türleri	8
1.2.1. Paralel mekanizmalar	10
1.2.1.1. Paralel manipulatörlerin avantajları	12
1.2.1.2. Paralel manipulatörlerin dezavantajları	13
1.2.2. Seri mekanizmalar	14
1.2.2.1. Seri manipulatörlerin avantajları	14
1.2.2.2. Seri manipulatörlerin dezavantajları.....	15
1.2.3. Hibrit mekanizmalar	15
1.2.4. Haptik mekanizmalar.....	16
1.2.5. İki Farklı Manipulatörün Karşılaştırılması	17
1.3. Stewart Platform Sistemi	19
1.3.1. SPM çeşitleri	20
1.3.1.1. 3x3'lük SPM türü	22
1.3.1.2. 3x6'luk SPM türü	23
1.3.1.3. 6x3'lük SPM türü	23
1.3.1.4. 6x6'luk SPM türü	23

1.3.2. Stewart platformunun kullanım alanları	23
1.4. Robot Mekanizmasının Kontrol Edilmesi	25
1.4.1. Geometrik eniyileme	28
1.4.2. Robot sistemlerinde mobilite	30
1.5. Çalışma Uzayı.....	31
1.5.1. Stewart platform mekanizması ile sağlanan doğrusal hareketler	32
1.5.2. Sistemdeki açısal hareketler	32
1.5.3. Kısıtlı çalışma uzayına sahip mekanizmalar	33
1.5.4. Çalışma uzayında yörünge takip planlaması	33
1.5.5. Değişken uzay yönteminin çalışma uzayına etkisi	34
1.6. Ters Kinematik Analiz.....	35
1.7. Düz Kinematik Analiz	35
1.8. Dinamik Analiz ve Modelleme.....	36

İKİNCİ BÖLÜM

SİMULASYON VE KİNEMATİK ANALİZ

2.1. Kinematik Hesaplama Metodları.....	38
2.2. Yazılımsal Yöntemler ile Çeşitli Tasarım Parametrelerin Simüle Edilmesi	40
2.2.1. Gerçekleştirilen sistemin parametre girdi ve çıktıları.....	42
2.3. Ters Kinematik Hesaplama.....	43
2.3.1. Model tasarım parametreleri.....	44
2.3.1.1. Sabit alt zemin	45
2.3.1.2. Hareketli uç işlevci	49
2.3.1.3. Bağlantı uzuvları (piston).....	52
2.3.2. Uç işlevci hareketleri ve kinematik denklemler	54
2.3.2.1. Ters kinematik yöntemlerle manipülasyon (öteleme, dönme) hareketleri	54
2.3.2.1.1. Trigonometrik yöntemlerle uç işlevci hareketleri	56
2.3.2.1.2. Matris çarpım yöntemi ile uç işlevci hareketleri.....	57
2.4. Sistemde Tekillik - Tekillik Sınır Bayrakları	59
2.5. Çalışma Uzayı Analizi.....	60

ÜÇÜNCÜ BÖLÜM
MEKANİZMA TASARIM PARAMETRELERİNİN ÇALIŞMA
UZAYINA ETKİLERİ

3.1. Hareketli Platform Tasarım Parametrelerinin Çalışma Uzayına Etkisi.....	68
3.2. Sabit Alt Zemin Tasarım Parametrelerinin Çalışma Uzayına Etkisi	87
3.3. Piston Tasarım Parametrelerinin Çalışma Uzayına Etkisi.....	105
SONUÇLAR VE ÖNERİLER	124
KAYNAKÇA	131
EKLER.....	136
ÖZGEÇMİŞ.....	152

KISALTMALAR

GO	:	Zemin merkezi
GUI	:	Grafik kullanıcı arayüzü
P_i	:	Pistonlar
P_l	:	Piston uzunluğu
PB_l	:	Piston gövdesi
PET	:	Platform eklem takımı
PL_{max}	:	Piston uzunluğunun maksimum değeri
PL_{min}	:	Piston uzunluğunun minimum değeri
PO	:	Platform orta noktası
PT_L	:	Piston tüpü
SPM	:	Stewart platform mekanizması
ZET	:	Zemin eklem takımı
PJG_D	:	Pistonun platform bağlantı noktası
PJG_L	:	Pistonun zemine bağlantı noktası
GJ_i	:	Zemin bağlantıları
GJ_R	:	Zemin bağlantılarının platforma yarıçapı
PJG_L	:	Platform bağlantıları takımının uzunluğu
PJG_d	:	Platformun bağlantı gruplarının mesafesi
PJ_i	:	Platform bağlantı noktaları
DPM(1-18)	:	DeneySEL platform modeli(1-18)
ARPM	:	Ana Referans Platform Modeli

TABLolar LİSTESİ

Tablo 1. Mekanizma Türleri Arasındaki Farklar.....	10
Tablo 2. Paralel ve Seri Mekanizma sistemlerine ait karşılaştırma.....	18
Tablo 3. X, Y ve Z eksenlerindeki manipülasyon parametreleri.....	54
Tablo 4. Deneysel Platform Modeli tasarım parametreleri.....	65
Tablo 5. ARPM'nin tasarım parametreleri	66
Tablo 6. ARPM'nin nokta bulutu hacmi ve nokta sayısı.....	67
Tablo 7. ARPM'nin platform merkezine göre X, Y ve Z eksenlerindeki sistem sınırları.....	67
Tablo 8. DPM1'in tasarım parametreleri	69
Tablo 9. DPM1'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı	71
Tablo 10. DPM1'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre değişim oranı	72
Tablo 11. DPM2'nin tasarım parametreleri	72
Tablo 12. DPM2'nin nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı	74
Tablo 13. DPM2'nin platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre değişim oranı	75
Tablo 14. DPM3'ün tasarım parametreleri	75
Tablo 15. DPM3'ün nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı	77
Tablo 16. DPM3'ün platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre değişim oranı	78
Tablo 17. DPM4'ün tasarım parametreleri	78
Tablo 18. DPM4'ün nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı	80
Tablo 19. DPM4'ün platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre değişim oranı	81
Tablo 20. DPM5'in tasarım parametreleri	81
Tablo 21. DPM5'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı	83
Tablo 22. DPM5'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre değişim oranı	84
Tablo 23. DPM6'nın tasarım parametreleri	84
Tablo 24. DPM6'nın nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı	86

Tablo 25. DPM6'nın platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	87
Tablo 26. DPM7'nin tasarım parametreleri	87
Tablo 27. DPM7'nin nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı	89
Tablo 28. DPM7'nin platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	90
Tablo 29. DPM8'in tasarım parametreleri	90
Tablo 30. DPM8'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı	92
Tablo 31. DPM8'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	93
Tablo 32. DPM9'un tasarım parametreleri	93
Tablo 33. DPM9'un nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı	95
Tablo 34. DPM9'un platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	96
Tablo 35. DPM10'un tasarım parametreleri	96
Tablo 36. DPM10'un nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı ...	98
Tablo 37. DPM10'un platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	99
Tablo 38. DPM11'in tasarım parametreleri	99
Tablo 39. DPM11'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı ..	101
Tablo 40. DPM11'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	102
Tablo 41. DPM12'nin tasarım parametreleri	102
Tablo 42. DPM12'nin nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı	104
Tablo 43. DPM12'nin platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	105
Tablo 44. DPM13'ün tasarım parametreleri	106
Tablo 45. DPM13'ün nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı ..	107
Tablo 46. DPM13'ün platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	108
Tablo 47. DPM14'ün tasarım parametreleri	108
Tablo 48. DPM14'ün nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı ..	110

Tablo 49. DPM14'ün platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	111
Tablo 50. DPM15'in tasarım parametreleri	111
Tablo 51. DPM15'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı ..	113
Tablo 52. DPM15'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	114
Tablo 53. DPM16'nın tasarım parametreleri	114
Tablo 54. DPM16'nın nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı	116
Tablo 55. DPM16'nın platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	117
Tablo 56. DPM17'nin tasarım parametreleri	117
Tablo 57. DPM17'nin nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı	119
Tablo 58. DPM17'nin platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	120
Tablo 59. DPM18'in tasarım parametreleri	120
Tablo 60. DPM18'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre deęişim oranı ..	122
Tablo 61. DPM18'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı	123
Tablo 62. ARPM ve DPM için çalışma uzayının hacim ve eksen sınırları	128
Tablo 63. DPM'lerin ARPM'ye göre çalışma uzayı farklarının orantısal gösterimi	129
Tablo 64. DPM'lerin ARPM'ye göre çalışma uzayı farklarının orantısal gösterimi	130

ŞEKİLLER LİSTESİ

Şekil 1. General Motors Firmasının geliştirdiği robot Unimate.....	2
Şekil 2. Çalışmada geliştirilen Stewart platform simülasyonu ve çalışma analizi yazılımı ...	40
Şekil 3. Özer (Solda) ve Öztürk (Sağda) çalışmalarının son hali.....	6
Şekil 4. Cerrahi sisteminin yapısı	7
Şekil 5. EJ200 Jet motorunun Stewart Platformu ile yönlendirilmesi.....	7
Şekil 6. LX4 modelinde Logabex robotunun gösterimi	8
Şekil 7. D. Stewart tarafından tasarlanan platform modeli.....	9
Şekil 8. Stewart Platform Mekanizması türleri	22
Şekil 9. Cerrahi uygulamada	24
Şekil 10. Piston modellerine örnekler	26
Şekil 11. O merkezine göre platformun açısız hareketleri.....	33
Şekil 12. Stewart Platform mekanizmasının temel bileşenlerinin tasarım parametreleri	43
Şekil 13. Stewart platform tasarımı modelinin temel elemanlarının 3d uzaydaki kurulu gösterimi.....	45
Şekil 14. Sabit alt zeminin görünümü ve tasarım parametreleri	46
Şekil 15. Zemin eklem takımlarının açısız konumları	46
Şekil 16. ZET için i. eklem $G_j i$ konum vektörü	48
Şekil 17. Hareketli platformun görünümü ve tasarım parametreleri	50
Şekil 18. Platform eklem takımlarının açısız konumları	50
Şekil 19. PET için i. eklem $P_j i$ konum vektörü	51
Şekil 20. Bağlantı uzvunun görünümü ve tasarım parametreleri.....	53
Şekil 21. Yapılan sistemin çalışma uzayı çıktıları.....	62
Şekil 22. Özgün SPM'nin tasarımını ve çalışma uzayını etkileyen temel tasarım parametreleri	64
Şekil 23. Referans platform modelinin simülasyon ortamındaki görünümü	66
Şekil 24. ARPM'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	67
Şekil 25. DPM1'in simülasyon ortamındaki görünümü	70
Şekil 26. DPM1'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	71

Şekil 27. DPM2'nin simülasyon ortamındaki görünümü	73
Şekil 28. DPM2'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	74
Şekil 29. DPM3'ün simülasyon ortamındaki görünümü	76
Şekil 30. DPM3'ün çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	77
Şekil 31. DPM4'ün simülasyon ortamındaki görünümü	79
Şekil 32. DPM4'ün çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	80
Şekil 33. DPM5'in simülasyon ortamındaki görünümü	82
Şekil 34. DPM5'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	83
Şekil 35. DPM6'nın simülasyon ortamındaki görünümü	85
Şekil 36. DPM6'nın çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	86
Şekil 37. DPM7'nin simülasyon ortamındaki görünümü	88
Şekil 38. DPM7'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	89
Şekil 39. DPM8'in simülasyon ortamındaki görünümü	91
Şekil 40. DPM8'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	92
Şekil 41. DPM9'un simülasyon ortamındaki görünümü	94
Şekil 42. DPM9'un çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	95
Şekil 43. DPM10'un simülasyon ortamındaki görünümü	97
Şekil 44. DPM10'un çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	98
Şekil 45. DPM11'in simülasyon ortamındaki görünümü	100
Şekil 46. DPM11'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü	101
Şekil 47. DPM12'nin simülasyon ortamındaki görünümü	103

Şekil 48. DPM12'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	104
Şekil 49. DPM13'ün simülasyon ortamındaki görünümü	106
Şekil 50. DPM13'ün çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	107
Şekil 51. DPM14'ün simülasyon ortamındaki görünümü	109
Şekil 52. DPM14'ün çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	110
Şekil 53. DPM15'in simülasyon ortamındaki görünümü	112
Şekil 54. DPM15'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	113
Şekil 55. DPM16'nın simülasyon ortamındaki görünümü.....	115
Şekil 56. DPM16'nın çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	116
Şekil 57. DPM17'nin simülasyon ortamındaki görünümü.....	118
Şekil 58. DPM17'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	119
Şekil 59. DPM18'in simülasyon ortamındaki görünümü	121
Şekil 60. DPM18'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü.....	122
Şekil 61. Örnek çalışma uzayı çıktısı (d=400 için, nokta bulutu aralığı 15 mm).....	124
Şekil 62. Örnek çalışma uzayı çıktısı (d=200 için, nokta bulutu aralığı 15 mm).....	125

EKLER LİSTESİ

EK-A	136
EK- B	138
EK- C	141
EK- Ç	146
EK- D	147
EK- E	149



ÖNSÖZ

Bu tez çalışması süresince beni yönlendiren ve çalışmanın tamamlanmasında büyük katkısı olan tez tanışmanım olan Prof. Dr. Bedri YÜKSEL hocama danışmanlığı konusunda minnettarım. Çalışma sürecinde beni destekleyen ve tez ile ilgili birçok konuda tartışma fırsatı bulabildiğim çalışma arkadaşlarım Yük. Müh. Muhammet ÖZER, Yük. Müh. Volkan ÖZTÜRK ve Yusuf Altay DÖNMEZ Bey'e çok teşekkür ederim.

Yaşadığım süre boyunca desteklerini hiçbir zaman esirgemeyen, yanımda olduklarını her zaman hissettiren aileme sonsuz minnet sunarım.

Tez çalışmam süresince desteğini esirgemeyen ve bana bu tezi gerçekleştirmem adına motivasyon sağlayan tüm hocalarım ve arkadaşlarıma teşekkür ederim.

GİRİŞ

Günümüzde, teknolojinin ilerlemesiyle birlikte, robotik sistemlerin uygulandığı alanlar neredeyse yok denecek kadar azdır. Özellikle savunma sanayi, endüstri ve eğlence sektörleri gibi birçok alanda, bu sistemlerin kullanımı daha da önem kazanmaktadır. İnsanların yaptığı monoton veya riskli işlerde, robotların tasarımı ve kullanımı; iş yükünü azaltmayı, zamandan tasarruf etmeyi ve ortada bir tehlike olması durumunda, o tehlikenin risklerini azaltmayı amaçlar. Bu sayede, robotik sistemler birçok avantaj sağlamaktadır.

Manipülatörler, çalışma alanlarına ve genel yapılarına göre üç farklı tipte üretilmiştir. Bunlar seri manipülatörler, paralel manipülatörler ve nadiren kullanılan hibrit manipülatörlerdir. Hibrit manipülatörlerin nadiren kullanılması, bu tip manipülatörlerin hangi koşullarda kullanılabilceği konusunda araştırmaların yönlendirilmesine neden olmuştur. Ayrıca, hibrit manipülatörlerin çalışma alanının işlevselliği konusu üzerinde yoğunlaşan araştırmalar vardır. Genel hatları olarak hibrit manipülatörler, açık-kapalı kinematik zincir yapısından oluşmaktadır. Paralel manipülatörler işlev olarak çözüm matrisi türünde hesaplamalar gerçekleştirirler. Paralel manipülatörlerin kapalı kinematik zinciri hesaplamalarından gerçekleşmektedir. Farklı bir şekilde çalışan seri manipülatörlerde ise işlemler açık kinematik zinciri hesaplamaları olarak gerçekleşmektedir. Hibrit manipülatör modeli, paralel manipülatörlerin kısıtlı çalışma alanındaki dezavantajını seri manipülatör sistemi yapısının avantajları ile birleştirerek ortadan kaldırmaktadır. Sonuç olarak bu gibi durumlar hibrit bir sisteme yönelimi artırmıştır. (Güneri, 2007).

Paralel mekanizmalı sistemler, çalışma yapıları bakımından incelendiğinde iki veya daha fazla bağlantı kolu aracılığıyla sabit ve hareketli platformlar arasında bir bağlantı sağlayarak işlemlerini gerçekleştirir. Bu tür manipülatör sistemlerinin ilk örneklerinden biri, Eric Gogh tarafından geliştirilen Stewart Platformudur. Stewart Platformu, otomotiv endüstrisinde ve uçak simülasyonlarında sıklıkla kullanılmış bir yapıdır. Bu yapı, endüstriyel çok sayıda sektörde de yaygın olarak tercih edilmiştir. Dr. Eric Gogh tarafından geliştirilen bu mekanizma, Stewart-Gough platformu olarak da bilinir (Dasgupta ve Mruthyunjaya, 2000).

Günümüzde, mekatronik ve robotik bölümlerin medikal sistemler üzerindeki ilgisi giderek artmaktadır. Bu trend, cerrahi uygulamalarda robotik sistemlerin kullanımının da arttığını göstermektedir. Medikal robotların kullanımı, cerrahın yerini alarak aynı işlemleri gerçekleştirirken, aynı zamanda çeşitli cerrahi uygulamalarında yeni tedavi seçeneklerinin entegre edilebilmesini de mümkün kılar. Bu sayede, medikal sistemlerin robotikleştirilmesi, hastaların daha etkin bir şekilde tedavi edilmesini sağlar.

General Motors firması tarafından 1958 yılında geliştirilen robot sistemleri hem gelecekte insan emeğinin yerini alacak hem de yürütülen iş performansını daha yüksek noktalara çıkaracaktı. Yapılan çalışmalara istinaden geliştirilen robot, Unimate adını aldı ve General Motors tarafından insanlara sunuldu(Şekil 1.). Daha sonra, bu robotu 1961 yılında bir sanayi sistemine entegre ettiler. Robotik yapıların iş alanlarında daha sistemli ve hızlı çalışmaları robotların daha çok yayılmasını sağladı ve günümüzde insanların hayatına önemli katkılar sağlıyorlar (Gourin ve Terris, 2007).



Şekil 1. General Motors Firmasının geliştirdiği robot Unimate
Kaynak: Gourin, C.G. ve Terris D.J. (2007)

Robotik sistemler, insanların endüstriyel alanda yaptığı işlerdeki karmaşıklığı azaltmaya yardımcı olmak için giderek daha fazla popüler hale gelmiştir. Özellikle tekrarlayan işlemlerde, işlemi gerçekleştiren mekanizmalara manipülatörlerin fonksiyonelliği eklenerek, işlemlerin daha verimli çalışmasını sağlamıştır. Robotların süreklilik ve hassasiyet özellikleri, endüstri, medikal, uzay ve matematik bilimlerinde sıklıkla kullanılmaya başlanmıştır. Ancak robotların insanların yerini almaya yönelik düşünce popülerliği gözlemlenmektedir. Robotların, son yıllarda medikal alanda kullanımı giderek daha yaygın hale gelmektedir. Bu trendin arkasında, endüstriyel yapıda az maliyeti, stabil çalışmaları ve verimli performans özelliklerinin etkisi vardır. Ancak, robotik sistemlerin dezavantajları arasında sistem güvenliği, robotun kontrolü, sistemin çalışma sınırları, bakım ve onarım gibi konular bulunmaktadır. Bu dezavantajları aşmak için, bilimsel araştırmalar yapılarak yenilikçi çözümler üretilmektedir (Braun, Mayer, Knoll, Lange ve Bauernschmitt, 2008).

BİRİNCİ BÖLÜM

GENEL BİLGİLER

Paralel mekanizmalar, D. Stewart tarafından ortaya atılan bir teknoloji olup, ilk olarak uçuş simülatörlerinde kullanılmak ve uçuklara alışmak için tasarlanmıştır. Ancak zamanla, bu platformlar eğlence makinaları, cerrahi robotlar, endüstriyel makine ve askeri projeler gibi çeşitli alanlarda da kullanılmaya başlamıştır. Bu mekanizmaların kullanım alanlarının keşfi amacıyla yapılan bilimsel araştırmalarda, mevcut sistemlerin yeni uygulama alanlarını ortaya çıkarmıştır. Ayrıca, yapısal olarak farklı şekilde çalışan sistemlerin geliştirilmesine istinaden birlikte, paralel mekanizmaların verimliliği üzerine yapılan çalışmaları da arttırmıştır. Bu mekanizmaların çeşitli kullanım alanları, özellikle tez çalışmalarında sıklıkla ele alınmıştır.

1.1. Literatür Taraması

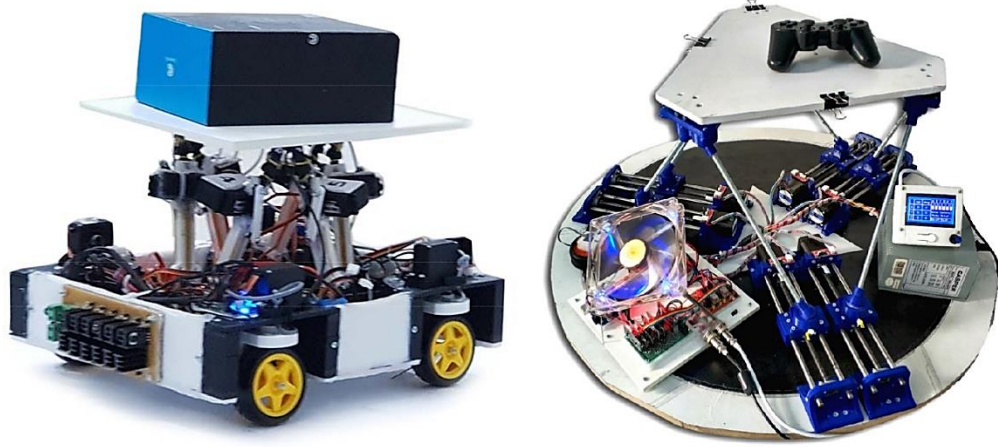
Çalışmalarını doğrusal yapıda paralel sistemler üzerinde yoğunlaştıran Bengü ve Küçük (Bingü ve Küçük, 2005), adını kendilerinin vermiş olduğu SIDED adında yazılım projesini MATLAB ortamında geliştirmiştir. Bu yazılım projesi ile benzetim amaçlanmıştır. Çalışmaları doğrultusunda geliştirmiş oldukları SIDED isimli proje yazılımı ile doğrusal yapıda paralel bir sistemi tasarlayıp farklı yapılarda benzetimi ve önizlemesi yapılabilmektedir. Ters ve doğrusal kinematik hesaplamaların benzetimi yapılabilen bir sistem geliştiren Angeles ve Gosselin (Gosselin ve Angeles, 1990) bu hesaplamaları üç serbestlik derecesindeki paralel yapıdaki manipülatörler üzerinde yapabilmektedir. Çalışmalarında oluşturmuş oldukları sistem sayesinde, analizleri istenilen yörünge uzayı bazında yaparak grafiksel animasyonlar ile benzetim önizlemesi yapılabilmektedir. Ülkemizde paralel manipülatör sistemleri üzerinde ilk master tez çalışmasını Gökçen (Gökçen, 1992) yazmıştır. 6x3 - 3x3 yapıda paralel bir sistem olan Stewart Platform Mekanizmasını dört ayaklı modelini matematiksel olarak tasarlamıştır. Bu stewart platformunun ayak yapıları, sistem, dinamik, düz kinematik ve ters kinematiği ile bütünleşik yapı oluşmuştur. Çalışmalarında ilgili matematiksel yapının yazılımı oluşturulmuş, benzetim analizleri sonrasında sonuçlarını paylaşmıştır.

Manipülâtör türlerinden günümüzde en çok kullanılan paralel manipülâtörler ile birçok çalışma yapılmış ve ayrıntılı bir şekilde Merlet (Merlet, 2006) tarafından kitap haline getirilmiştir. Paralel manipülâtör türlerinden olan Stewart platformunda da birçok sorunlar oluşmuş ve bu sorunlara yönelik yapılmış olan çalışmalar derlenerek Mruthyunjaya ve Dasgupta (Dasgupta ve Mruthyunjaya, 2000), tarafından köklü araştırmalar yapılmıştır. Çalışmalarına paralel mekanizma türlerinin cerrahi sistemler ve insan hayatını yük taşıma kolaylaştıran sistemlerde stabil çalıştığını da eklemiştirlerdir.

Paralel manipülâtör türü olan Stewart platformunda birçok bacak bağlantı kombinasyonu bulunmakta olup, bu kombinasyonlara Gao ve Zhang (Majid, Huang ve Yao, 2000) tarafından öneriler (6x3, 3x6, 3x3 gibi.) verilmiştir. Çalışmalarında bacak bağlantı kombinasyonlarını, bacakların bağlandığı noktalar arasında formülse dayalı olarak 3850 farklı türde Stewart platform mekanizmasının oluşabileceğini ispatlarla göstermişlerdir. Paralel mekanizmalar üzerinde çalışmalarını sürdüren Wang (Wang, 2007), yüksek lisans tezi üzerindeki çalışmasında 20 farklı Stewart platform mekanizması kurulabileceğini ayrıntılı açıklamıştır. Günümüzde endüstriyel sistemlerde kullanılmakta olan örnek paralel mekanizmaları örneklemiştir.

Bir aracın hareketlerinde esneklik sağlamayı hedefleyen Özer (Özer, 2019), bir sistem tasarımı üzerine odaklanmıştır. Bu sistem, araçtaki dört tekerleğin ayrı ayrı kontrolünü sağlayan bir tekerlek sisteminin yanı sıra bu tekerlek sisteminin üzerine yerleştirilen bir denge platformundan oluşmaktadır. Tekerlek sistemi, Ackermann yönlendirme geometrisini kullanarak açısal hesaplamalar yaparken, denge platformu ise bir Stewart platformu tasarımıdır ve paralel bir mekanizmaya sahip 3x3 bağlantı modeli kullanılmıştır. Öztürk (Öztürk, 2019) ise çalışmasında, paralel mekanizmaların tasarımı, çalışma prensipleri ve uygulamaları üzerine odaklanmıştır. Bu tez çalışmasında, bir SPM tasarımı incelenmiş ve bu mekanizmanın zemine monte bir alt ve uç işlevi yerine getiren hareketli üst tabakası arasında eşit uzunlukta bağlantı kollarına sahip bir manipülâtör hazırlanmıştır. Bu tasarımın, mekanik avantaj oranını arttıran ve çalışma kapasitesini artıran bir yapı olduğu belirtilmiştir. Ayrıca, bu tasarımın daha az sayıda parçaya sahip olması nedeniyle daha kolay ve ucuz bir şekilde üretilebileceği vurgulanmıştır. Bu tez çalışmasında, SPM'nin kinematik modeli

oluşturulmuş ve çalışma prensipleri açıklanmıştır. Ayrıca, SPM'nin birçok farklı uygulaması örnekler verilerek anlatılmıştır.

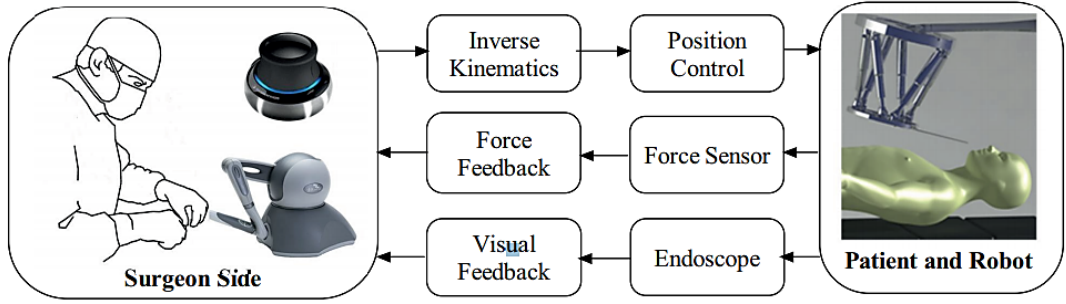


Şekil 2. Özer (Solda) ve Öztürk (Sağda) çalışmalarının son hali
Kaynak: Özer, M. (2019); Öztürk, V. (2019).

Günümüz endüstrisinde gelişmeyi ele alan Anlı (Anlı, 2005), her geçen gün paralel sistemlerin daha da gelişeceğini araştırmış çalışmasında yer vermiştir. Paralel platformlar üzerinde belli aksiyon planlamalarını paylaşan Ulaş (Ulaş, 2009), planlamalar üzerinde manipülatör bağlantı mafsallarının yük taşıma durumunu, mekanizmanın geometrik yapıda ölçülendirilmesini ve çalışma uzayını ele almıştır. Çalışmasında iki farklı yapıda paralel manipülatör sistemini karşılaştırmıştır.

Stewart platform mekanizma yapısında bir paralel manipülatör Fichter tarafından yapılmıştır. Çalışmasında kinematik, teklik ve kinetik çıktılarını açığa kavuşturmak için Oregon State Üniversitesi'nde daha önceden yapılan eski çalışmaları sunmuştur. Çalışması kendisinden sonra paralel manipülatörler üzerine literatür taraması yapanlar için büyük bir kapı aralamıştır (Fichter, 1986).

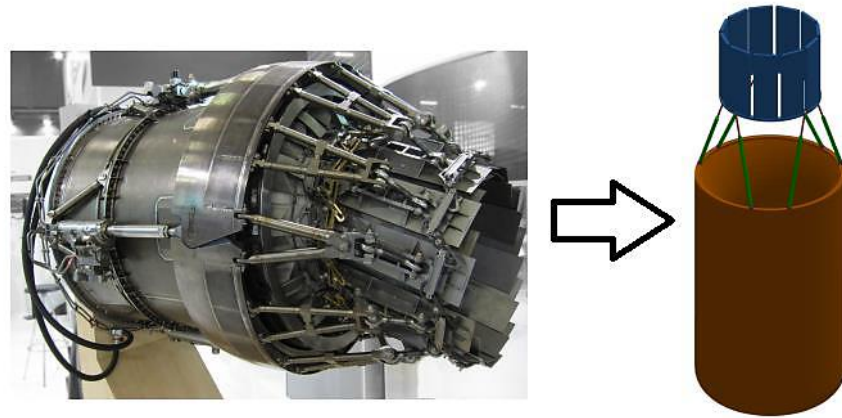
Stewart platform mekanizmasının tabla diye tanımlanan hareket uzayını belirleyen uç eyleyicisininin x, y ve z eksenleri üzerinde doğrusal ve dairesel olarak 6 serbestlik derecesinde hareket edebilmektedir. Çalışmalarında Kızır ve Bingül, karmaşık cerrahi operasyonların robotlarla tam otomatik hale getirilebilmesi için Stewart platformunun hassas cerrahi müdahalelerde kullanımına yönelik bir sistem simülasyonu geliştirdiler. Yaptıkları çalışma çıktısı Şekil 4. 'te verilmiştir (Kızır ve Bingül, 2019).



Şekil 3. Cerrahi sisteminin yapısı

Kaynak: Kizir, S. ve Bingül, Z. (2019).

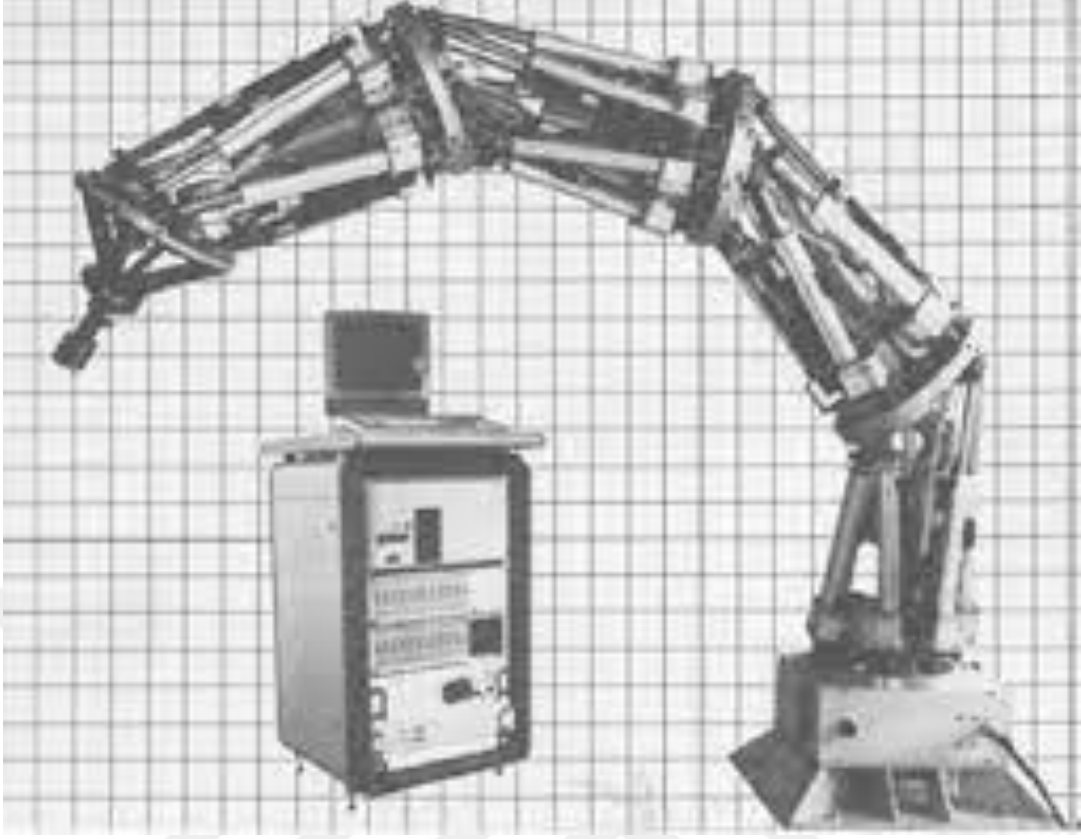
Stewart platformları güçlü ve çalışma uzayları büyük olmasından dolayı birçok alanda kullanılmaktadır. Neeraj, Jonathan ve Elvis çalışmalarında bir jet motorunun yön tayininde uç eyleyiciyi stewart platformu sayesinde geliştirmişlerdir. Sistemin geliştirilmiş hali Şekil 5. 'te verilmiştir (Neeraj, Jonathan ve Elvis, 2018).



Şekil 4. EJ200 Jet motorunun Stewart Platformu ile yönlendirilmesi

Kaynak: Neeraj vd. (2018).

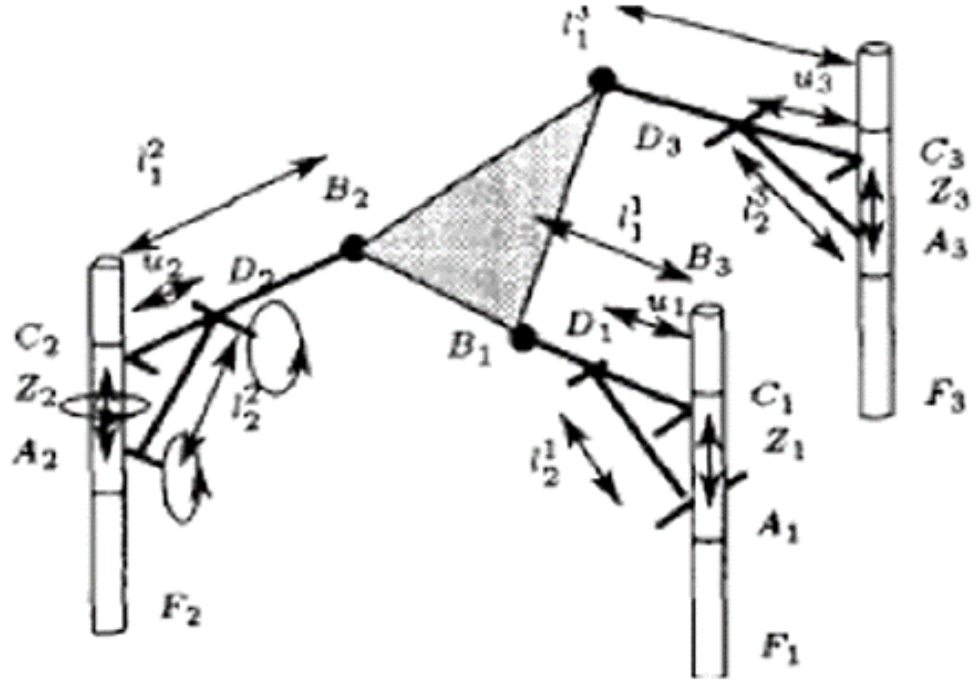
Logabex isimindeki robot seri halde birbiri ile bağlanmış stewart platform mekanizmalarından oluşmaktadır. Khail ve Dombre çalışmalarında logabex seri robotu hakkında ve paralel stewart robotlar hakkında bilgiler vermişlerdir. Logabex robotunun bir görseli Şekil 6. 'da verilmiştir (Khail ve Dombre, 2002).



Şekil 5. LX4 modelinde Logabex robotunun gösterimi
Kaynak: Khail, W. ve Dombre, E. (2002).

1.2. Mekanizma Türleri

Seri yapı ve paralel yapı şeklinde üretilerek geçmişten günümüze birçok alanda yoğun bir şekilde kullanılan mekanizma yapılarıdır. Kullanımı oldukça yaygın olan Stewart Paralel Platformu D. Stewart tarafından hava araçlarının simülatörlerinde eğitim amaçlı kullanılmak üzere tasarlanmıştır. Paralel stewart platform mekanizmaları sistem çalışırken sabit kalan bir plaka ve hareketli bulunan bir plaka arasına yerleştirilmiş altı adet bağlantı kolu şeklinde oluşmaktadır. Bu bağlantı kollarının uzunlukları değişmektedir. Bu paralel mekanizma bağlantı kolu sayısı ve kolların bağlanma durumuna göre yaptıkları çalışma durumu hareketleri farklılık göstermektedir.



Şekil 6. D. Stewart tarafından tasarlanan platform modeli

Kaynak: Anlı, E. (2005).

Stewart paralel mekanizmaları Kapalı kinematik zincir yapısını oluşturmaktadırlar. Bu kapalı kinematik zincir yapısında Paralel platforma bulunan altı bağlantı kolu, üst plakadaki yüzey ile alt plakada bulunan bağlantı noktalarına paralel bir yapıda bulunmaktadırlar. Burada yapılan bağlantılar çalışma alanında gerçekleştirilen işlemlerde hassasiyete ve hızlı çalışmaya izin vermektedir. Manipülatörler daha önce de bahsettiğimiz gibi çalışma alanlarına göre seri yapıda ve paralel yapıda üretilmektedirler. Seri kinematik manipülator sistemlerine göre paralel kinematik manipülator sistemleri çalışma yapısı olarak daha hassas ve daha hızlı bir hareket kabiliyetine sahiptirler. Paralel manipülatörler bu avantajları sayesinde endüstriyel alanlarda yapılan çalışmalarda kullanım olarak daha uygun yapıdadırlar. Örnek verecek olursak endüstriyel bir ortamda bir nesneyi kaldırması ve bir noktaya bırakması konusunda hareket olarak hassas ve hızlı bir şekilde işlem sağlayabilirler. (Ünsal, 2007)

Tablo 1. Mekanizma Türleri Arasındaki Farklar

Seri Yapıda Mekanizmalar	Hibrit Yapıda Mekanizmalar	Paralel Yapıda Mekanizmalar
Hızlı çalışır.	Sistemin hızı, tasarıma bağlıdır.	Yavaş çalışır.
Bu mekanizmada hatalar en uç noktaya kadar birikerek gider. Uç noktanın hata olasılığı artabilir.	Birleşme bir yapı olduğundan tasarıma göre iyi veya kötü bir hal alabilir.	Birçok bağ olduğundan, hataların ortalama parametresi en uç noktaya taşınır.
Çalışma uzayı genelde diğerlerine göre büyüktür.	Çalışma uzayı, seri ve paralel birleşimi olduğu için tasarımına göre değişebilir.	Çalışma uzayı, birçok bağ kullanımından dolayı küçük olabilmektedir.
Birbirine ardışık eklemlerden meydana gelir.	Seri ve paralel yapının birleşimidir. Kullanım uygunluğuna göre tasarımı değişebilir.	Birçok bağ, üst ve alt plaka arasında yer alır.
Güçsüzdür.	Güç dengesi, tasarım yapısına göre belirlenir.	Güçlü bir yapıya sahiptir.
Basit kinematik denklemler kullanılır.	Kinematik denklemlerin karmaşık bir yapısı kullanılır.	Kinematik denklemlerin karmaşık bir yapısı kullanılır.

1.2.1. Paralel mekanizmalar

Paralel yapıya sahip mekanizmalar teknoloji çağında yaşanan gelişmeleri ile hassas çalışma gereğinde bulunulan endüstriyel birçok alanda kullanılmıştır. Paralel manipülatörler karşılaştırma yapıldığında seri manipülatör sistemlerine göre birçok konuda avantajlı bir yapıdadırlar. Bu manipülatör yapısında bağlantı kollarına düşen toplam yük eşit olarak dağıtılmakta ve bu nedenle yük kapasiteleri seri manipülatör sistemlerine göre daha çöktür. Bunun dışında atalet momentinin düşük olması, rijit seviyesinin yüksek olması ve paralel çalışma yapısına uygun olarak bazı durumlarda

oluşabilecek hataların minimum yansımaları için daha hassas ve hızlı çalışmaktadır. Burada da belirtildiği gibi paralel sistemlerin avantajlarına rastlamak mümkündür. Paralel mimari yapısına sahip sistemler, seri sistemlerde kullanılan kinematik denklemlere göre daha kompleks bir kinematik yapıya sahiptir. (Dumlu ve Erentürk, 2011)

Paralel yapıya sahip manipülatör sisteminin özellikle delta robot sistemleri ve uçuş simülasyon sistemlerinde kullanılmasının en temel yapısı sabit bulunan platformun aynı aktüatör kuvvetleri ile geniş bir çalışma yapısına ulaşmasıdır.

Paralel mekanizmalar, birçok farklı alanda kullanılmaktadır. Özellikle, askeri, cerrahi ve endüstriyel alanlarda sıkça kullanılmaktadır.

Askeri alanlarda, paralel mekanizmalar genellikle hava kuvvetleri uçaklarında, helikopterlerde ve roketlerde kullanılmaktadır. Bu mekanizmalar, uçakların ve helikopterlerin güçlü motorlarının ürettiği torku iletme, açılı pozisyonları tutma ve havada istikrarı sağlama gibi görevlerde kullanılır. Ayrıca, roketlerde de paralel mekanizmalar kullanılmaktadır. Örneğin, roketlerin fırlatılma ve uçuş sırasında istenen pozisyonları tutmak için paralel mekanizmalar kullanılır.

Cerrahi alanlarda ise, paralel mekanizmalar robotik cerrahi sistemlerinde kullanılmaktadır. Bu sistemler, cerrahi işlemleri yaparken doktorların hareketlerini taklit ederek, hassas ve özel bir şekilde cerrahi işlemler gerçekleştirirler. Bu sayede, cerrahi işlemler daha hızlı ve daha az zaman alırken, aynı zamanda da daha hassas ve doğru bir şekilde gerçekleştirilir.

Endüstriyel alanlarda ise, paralel mekanizmalar genellikle endüstriyel robotların ve otomatik makine ünitelerinin hareketlerinde kullanılmaktadır. Örneğin, endüstriyel robotlar, parçaları yerleştirme, işleme ve taşıma gibi görevleri yaparken paralel mekanizmalar kullanılır. Aynı şekilde, otomatik makine ünitelerinde de paralel mekanizmalar kullanılmaktadır. Bu mekanizmalar, makine ünitelerinin hareketlerini esnek ve hassas bir şekilde gerçekleştirme görevini üstlenir.

Paralel mekanizmalar, daha farklı farklı alanlarda da kullanılmaktadır. Örneğin, hava taşımacılığı, uzay teknolojisi, deniz taşımacılığı, enerji üretimi, mühendislik simülasyonları, havacılık, otomobil ve motosiklet endüstrisi gibi alanlarda da paralel mekanizmalar kullanılmaktadır.

Bunun yanı sıra, paralel mekanizmalar, günümüzde birçok farklı sektörde de gösteri ve eğlence amaçlı kullanılmaktadır. Örneğin, sirklerde ve çeşitli gösterilerde paralel mekanizmalar kullanılmaktadır. Bu mekanizmalar, çeşitli hareketleri gerçekleştirerek, gösterilerin daha eğlenceli ve izlenebilir hale getirilmesine yardımcı olurlar.

Sonuç olarak, paralel mekanizmalar, farklı alanlarda ve birçok farklı amaçla kullanılmaktadır. Bu mekanizmaların en yaygın kullanım alanları arasında askeri, cerrahi, endüstriyel ve gösteri amaçlı kullanımlar yer almaktadır.

1.2.1.1. Paralel manipülatörlerin avantajları

Paralel manipülatör sistemlerindeki en büyük avantajlardan biri yapısı esasında bağlantı kollarının iki platforma farklı noktalardan bağlı olarak platform üzerindeki ağırlığı paylaşırlar. Bu sayede paralel manipülatör sistemleri bu yapı ile fazla ağırlık taşıma kapasitesi ile işlem sağlayabilirler. (Liu ve diğ. 2003). Bu durumun neticesinde paralel manipülatörler sistemlerinin ikinci bir avantajları ise atalet momentlerinin düşük olmasıdır. İki platform(hareketli, hareketsiz platformlar) arasındaki bağlantı kolları, manipülatörün çalışma uzayında yapacağı konumlamaya göre bağlantı kolları ağırlıkları paylaştıkları için aktüatör tercih edilirken daha az maliyetli ve küçük aktüatörler ve daha kısa bağlantı kolları ile tasarım gerçekleştirilebilir (Merlet, 2006). Bu şekilde çalışan manipülatörler seri manipülatör lere göre daha hassas ve hızlı bir şekilde çalışmaktadırlar. Hızlı ve hassas çalışmak istenilen bir sistemde en önemli nüans mekanizmadaki ağırlığı azaltmaktan geçmektedir. Hassasiyetinin daha iyi olma nedeni ise manipülatörde sabit halde ve hareketli halde bulunan platformlar arasındaki bağlantı kollarının eğimleri azalarak yaptıkları işlemlerde daha hassas çalışacaktır (Bulut, 2015).

Paralel mekanizmaların avantajları şunlardır:

- Ölçülü ve kontrollü hareketler: Paralel mekanizmalar, ölçülü ve kontrollü hareketler gerçekleştirebilen yapılardır. Bu sayede, paralel mekanizmalar, hassas ve özel hareketler gerçekleştirme görevlerinde kullanılabilirler.
- Yüksek mukavemet: Paralel mekanizmalar, genellikle yüksek mukavemete sahiptir. Bu sayede, paralel mekanizmalar, ağır yükleri taşıma ve kontrol etme görevlerinde de kullanılabilirler.

- Esnek hareketler: Paralel mekanizmalar, ayrıca, genellikle esnek hareketler gerçekleştirebilen yapılardır. Bu mekanizmalar, birçok farklı hareket gerçekleştirmeye kapasitesine sahiptir. Örneğin, paralel mekanizmalar, düzgün ve açılı hareketler gerçekleştirebilirler. Ayrıca, paralel mekanizmalar, birçok farklı açıda hareket edebilirler. Bu sayede, paralel mekanizmalar, birçok farklı görevde kullanılabilirler.
- Yüksek doğruluk: Paralel mekanizmalar, genellikle yüksek doğrulukta hareketler gerçekleştirebilen yapılardır. Bu sayede, paralel mekanizmalar, hassas ve özel hareketler gerçekleştirme görevlerinde de kullanılabilirler.
- Yüksek hız: Paralel mekanizmalar, genellikle yüksek hızda hareketler gerçekleştirebilen yapılardır. Bu sayede, paralel mekanizmalar, hızlı hareketler gerçekleştirme görevlerinde de kullanılabilirler.

Sonuç olarak, paralel mekanizmalar, ölçülü ve kontrollü hareketler gerçekleştirebilen, yerine göre düşük maliyetli, yerine göre küçük boyutlu, yüksek mukavemete sahip, esnek hareketler gerçekleştirebilen, yüksek doğrulukta hareketler gerçekleştirebilen ve yüksek hızda hareketler gerçekleştirebilen yapılardır. Bu avantajları sayesinde, paralel mekanizmalar birçok farklı alanda ve birçok farklı amaçla kullanılmaktadır.

1.2.1.2. Paralel manipülatörlerin dezavantajları

Paralel manipülatör sistemleri, manipülatör sisteminin çalışma şekline bağlı olarak hareket işlemleri dar bir açıda çalışmaktadır. Bunun dışında mafsalları ile bağlı olan hareket uzuvlarının çarpışmaları da paralel manipülatörlerde çalışma uzayını kısıtlamaktadır (Bulut, 2015).

Paralel mekanizmalar, ölçülü ve kontrollü hareketler gerçekleştirebilen, düşük maliyetli, küçük boyutlu, yüksek mukavemete sahip, esnek hareketler gerçekleştirebilen, yüksek doğrulukta hareketler gerçekleştirebilen ve yüksek hızda hareketler gerçekleştirebilen yapılardır. Ancak, paralel mekanizmalar aynı zamanda yüksek maliyet, büyük boyut, düşük mukavemet, düşük hız, düşük esneklik ve yüksek hassasiyet gibi dezavantajlara da sahip olabilirler. Bu nedenlerle, paralel mekanizmaların avantajları ve dezavantajları, kullanım amacına göre değerlendirilmelidir.

1.2.2. Seri mekanizmalar

Seri manipülatör sistemleri, birden fazla sıralı uzuv mafsalsal/mafsalların ara bağlantıları ile oluşmaktadır. Uzuvun son noktasında bir uç noktası bulunmaktadır. Uzuvlar bu yapısal şekil hareketlerini aktüatörler sayesinde gerçekleştirmektedirler (Tsai, 1999).

Seri mekanizmalar, çeşitli endüstriyel uygulamalarda ve makine bileşenlerinde sıklıkla kullanılır. Örneğin, seri mekanizmalar, otomatik üretim hatlarında, endüstriyel makine parçalarında ve robotik sistemlerde güç aktarımı sağlamak ve hareketleri kontrol etmek için kullanılabilir. Ayrıca, seri mekanizmalar, araç ve ekipmanlarda, kapı ve pencere açılış sistemlerinde ve ev aletlerinde de kullanılabilir.

Seri mekanizmalar, hareketleri düzenlemek ve güç aktarımı sağlamak için kullanılan mekanik araçlardır. Örneğin, bir dişli çarkın bir dişliyle çalışması, bir seri mekanizmadır. Seri mekanizmalar, genellikle hareketleri çok düzenli ve hassas bir şekilde yapabilirler ve bu nedenle endüstriyel uygulamalarda ve makine bileşenlerinde sıklıkla tercih edilirler.

Seri mekanizmalar, birçok farklı şekilde tasarım edilebilir ve çeşitli güç ve hareketleri kontrol etmek için kullanılabilirler. Örneğin, bir dişli çarkın bir dişliyle çalışması gibi, bir seri mekanizma hareketleri çok düzenli ve hassas bir şekilde yapabilir. Ayrıca, bir seri mekanizma, bir motorun gücünü bir mekanik cihazın hareketine dönüştürebilir veya bir mekanik cihazın hareketini bir motora dönüştürülebilir. Bu sayede, seri mekanizmalar, çeşitli makine ve sistemlerde hareketleri kontrol etmek ve güç aktarımı sağlamak için kullanılabilir.

1.2.2.1. Seri manipülatörlerin avantajları

Endüstriyel alanlarda insan yerine süreklilik arz eden işlerde sıklıkla kullanılmaktadır. Endüstriyel alanda kullanım talebi avantaj olarak değerlendirilmektedir. Seri manipülatörler, endüstriyel otomasyon sistemlerinde ve robotik uygulamalarda sıklıkla tercih edilir. Seri manipülatörler, çeşitli görevlerde kullanılabilir ve verimliliği artırırken, maliyet etkinliğini de sağlar.

1.2.2.2. *Seri manipülatörlerin dezavantajları*

Seri manipülatörlerin paralel manipülatör sistemlerine göre en büyük dezavantajlarından biri de bağlantı kollarının ard arda mafsal yapısı ile bağlı olmasıdır. Burada sıralı halde bağlanan bağlantı kollarında bir sonraki kol kendisinden önceki bağlantı kollarına hem kendi yükünü hem de uç noktasında bulunan ağırlığın yükünü yüklediği için sıralı bu kolların eğilme momenti oldukça fazladır. Seri manipülatörlerde aktüatör olarak motor kullanımı durumunda sabit platforma en uzak bağlantı kolundan en yakın bağlantı koluna her bağlantıda kullanılan motor bir öncekine göre daha büyük ebatlarında daha dayanıklı motorlar olmalıdır. Bu yapıda bu şekilde bir motor sistemi kullanılmaz ise bağlantı kolları birbirini taşıyamayacaktır. Bu durumda büyük motorların kullanılması maliyetin yanı sıra taşıyacağı ağırlıkta artıracaktır. Seri manipülatör sistemleri ağır yüklerin taşınmasına pek uygun sistemler değildir. Bu şekilde ağır motorların kullanılması sistemin ataletini arttıracığı için hassas bir çalışma çıkarılmaktadır (Bulut, 2015).

1.2.3. **Hibrit mekanizmalar**

Tanım ve yapı olarak hibrit manipülatör sistemleri, seri halde birbirine bağlı olarak çalışan paralel manipülatör sistemleridir. Hibrit manipülatörler bu şekilde bağlanmaları sonucu seri manipülatörlerin en büyük avantajlarından biri olan çalışma uzayının geniş olması ve ayrıca paralel manipülatör sistemlerinin rijitlik yapısı gibi avantajlarını ortak olarak kullanılabilmektedir (Bulut, 2015). Hibrit sistem, seri ve paralel manipülatörlerin birleşiminden oluştuğu için seri manipülatöre göre daha az karmaşık bir kinematik hesaplama olmasına karşın paralel manipülatörlere göre kinematik hesaplamaları karmaşık bir hal almakta ve oldukça zordur (Anlı, 2005; Anlı, Alp, Sait, Yurt ve Özkol, 2005). Hibrit mekanizmalar, mekanik ve elektronik bileşenlerin birleştirilmesiyle oluşturulan mekanik araçlardır. Bu cihazlar, mekanik parçaların hareketlerini düzenleyerek, bir nesnenin konumunu veya pozisyonunu değiştirebilir. Hibrit mekanizmalar, çeşitli endüstriyel uygulamalarda ve robotik sistemlerde sıklıkla kullanılır.

Hibrit mekanizmaların avantajları şunlardır:

- Hassas hareket: Hibrit mekanizmalar, hareketlerini çok düzenli ve hassas bir şekilde yapabilirler. Bu, nesnelerin konumunu ve pozisyonunu çok düzenli bir şekilde değiştirmek için kullanılır.
- Ölçeklenebilirlik: Hibrit mekanizmalar, çeşitli ölçeklerde tasarım edilebilir ve böylece farklı uygulamalar için uygun hale getirilebilir.
- Programlanabilirlik: Hibrit mekanizmalar, hareketlerini önceden programlanmış bir kod kullanarak yapabilirler. Bu, mekanizmaların çok düzenli ve tekrarlanan hareketler yapmasını sağlar.
- Verimlilik: Hibrit mekanizmalar, verimliliği artırmada yardımcı olur. Örneğin, mekanizmalar, çok düzenli ve hassas hareketler yaparak, üretim süreçlerini hızlandırır ve hataları azaltır.
- Esneklik: Hibrit mekanizmalar, farklı endüstriyel uygulamalar için esnek bir şekilde tasarım edilebilir. Bu, cihazın çeşitli görevlerde kullanılmasını sağlar.
- İleri teknolojik özellikler: Hibrit mekanizmalar, yüksek teknolojik özelliklere sahip olabilir. Örneğin, mekanizmalar, hareketlerini sensörler ve diğer ileri teknolojik cihazlar kullanarak daha hassas bir şekilde yapabilir.

1.2.4. Haptik mekanizmalar

Yunan dil terminolojisinde bulunan ‘haptesthai’ olarak ifade edilen haptik kelimesi tanım olarak mekanik sistemlere temas ederek bu sistemleri hissetme anlamındadır. Haptik yapıya sahip olan bir sistem sanal ortamda bulunan nesnelere temas etme, kontrol altına almak ve bu nesnelere ile ilgili düzenlemek gibi işlemler yapmamıza olanak tanır. Bu sistemler kullanıcılardan aldıkları veriye göre insan ile makine arasında temassal bir iletişim kurmayı sağlar. (Hayward, Choksi, Lanvin ve Ramstein, 1994)

Sanal ortamda veya uzaktan kontrollü yapıya sahip sistemlerinde oldukça önemli bir yer tutmaktadır. Sanallaştırılmış bir ortamda kullanıcı deneyimi ve memnuniyeti için kullanıcının sanal ortamı hissetmesini ele alan çalışmaları içermektedir. Haptik sistemler tıp, medikal, askeri uygulamalar, savunma sanayi,

eđitim, oyun ve eđence gibi birok uygulama alanında kullanılmaktadır (Seiichiro, Wataru ve Kouhei, 2005; Zhan ve Lan, 2008).

Haptik mekanizmalar, dokunsal geri bildirim sađlayan mekanizmalar ve sistemlerdir. Bu mekanizmalar, bir kullanıcının bir cihazı kullanırken, dokunsal olarak cihazın nasıl tepki verdiđini anlamasını sađlar. Örneđin, bir telefon ekranına dokunulduđunda, ekranın nasıl hareket ettiđini veya bir joystick kullanıldıđında, joystickin nasıl hareket ettiđini hissetmenizi sađlar. Bu geri bildirim, kullanıcıların cihazı daha dođru ve daha az zorlukla kullanmalarına yardımcı olur.

Haptik mekanizmalar, birok farklı sektörde kullanılır. Örneđin, mobil cihazlar, oyun konsolları, masaüstü bilgisayarlar ve otomobil kumandaları gibi cihazlarda haptik mekanizmalar bulunur. Ayrıca, haptik mekanizmalar, makine öğrenimi ve robotik gibi alanlarda da kullanılır.

Haptik mekanizmaların tasarımları, birok farklı sisteme farklı işlevlerle eklenerek tasarımları gerçekleştirilebilir. Örneđin, bir haptik motor, bir cihazın hareketini veya titreşimini sađlayabilir. Bir haptik sensör, cihazın bir yüzeyin nasıl hareket ettiđini ölçebilir. Bir haptik ekran ise, dokunuşa cevap vererek bir kullanıcının ekranı dokunduđunu hissettirebilir.

Sonuç olarak, haptik mekanizmalar, kullanıcıların cihazları kullanırken dokunsal geri bildirim almalarını sađlar ve bu geri bildirim, cihazı daha dođru ve daha az zorlukla kullanmalarını sađlar. Haptik mekanizmalar, birok farklı sektörde ve birok farklı şekilde tasarım edilebilir ve kullanım alanları geniştir.

1.2.5. İki Farklı Manipölatörün Karşılaştırılması

Mekanizmalar, bir görevin yerine getirilmesi için kullanılan yapılar ve araçlar olarak tanımlanabilir. Bu yapı ve araçlar, birbirleriyle ilişkili olarak alışır ve bir görevi yerine getirmek için tasarlandıklarından mekanizmalar sıklıkla birbirleriyle ilişkilidir.

Mekanizmalar, genellikle paralel ve seri olarak ikiye ayrılır. Bu kategorileme, mekanizmaların nasıl alıştıklarına ele alınacaktır ve ayrıca mekanizmaların birbirleriyle nasıl bir alışma ilişkisi olduđu anlatılmaktadır.

Paralel mekanizmalar, birbirleriyle paralel olarak çalışır ve aynı anda çalışırlar. Bu mekanizmalar, genellikle aynı görevi yerine getirirler ve birbirlerine göre ayrılmazlar. Örneğin, bir araba motorunda bulunan silindirler paralel yapıda bulunan mekanizmadır ve aynı anda çalışırlar.

Seri mekanizmalar ise, birbirleriyle sırasıyla çalışır ve birbirlerini takip ederler. Bu mekanizmalar, genellikle farklı görevleri yerine getirirler ve birbirlerine göre ayrılmışlardır. Örneğin, bir araba vites sistemi seri mekanizmadır ve birbirlerini takip ederek çalışırlar.

Mekanizmaların paralel ve seri olarak kategorize edilmesi, mekanizmaların nasıl çalıştıklarını anlamaya yardımcı olur. Bu kategorileme, mekanizmaların birbirleriyle nasıl ilişkili olduğunu ve hangi görevleri yerine getirdiğini anlamaya yardımcı olur. Mekanizmaların paralel ve seri olarak kategorize edilmesi, aynı zamanda mekanizmaların nasıl tasarlandıkları ve nasıl çalıştıklarını anlamaya yardımcı olur. Tablo 2. 'de karşılaştırma yapılmıştır.

Tablo 2. Paralel ve Seri Mekanizma sistemlerine ait karşılaştırma

	Paralel Mekanizmalar	Seri Mekanizmalar
Mekanizma tipi	Kapalı döngü	Açık döngü
Son etkileyici	Platform (hareketli tabla)	Tutucu (gripper)
Tanımı	Kartezyen uzay	Ekleme uzay
Aktüatör yeri	Sabit tablanın çevresi	Bağlantılar üzerinde
Atalet kuvveti ve rijitlik	Daha az ve yüksek	Yüksek ve daha az
Tasarım ilkeleri	Yapı, çalışma alanı, tekillik,	Dayanıklılık ve sertlik hususu,

	bağlantı kesişmesi	titreşim özellikleri
Tercih özelliği	Rijitlik	Beceri
İleri kinematik kullanımı	Zor ve karmaşık	Basit
Ters kinematik kullanımı	Basit	Karmaşık

1.3. Stewart Platform Sistemi

Stewart Platform, bir robotik sistem olarak bilinen bir mekanizmadır. Bu mekanizma, yüksek hassasiyetli hareketler yapabilen bir platform sağlar. Stewart Platform, genellikle havacılık, gemi inşaatı, mühendislik tasarımı gibi alanlarda kullanılır.

Stewart Platform, bir dizi düzgün dağılmış paralel eksenler üzerinde hareket eden bir platform ve bu eksenlerin her birine bağlı olan bir dizi kol ve piston tarafından desteklenir. Bu kollar ve pistonlar, hidrolik veya pnömotik sistemler kullanılarak hareket ettirilebilir. Stewart Platform, hareket ettirilen kolların uzunluklarını değiştirerek platformu hareket ettirir. Bu hareketler, yüksek hassasiyetli bir şekilde kontrol edilerek, platformun istenen yönde hareket etmesini sağlar.

Stewart Platformu, hareket ettirilen kolların sayısına göre farklı şekillerde tasarlanan sistemlerdir. Genellikle 6 adet bağlantı kolu ve piston kullanılır, ancak daha fazla kol ve piston kullanılarak daha da hassas hareketler yapılabilir. Stewart Platform, hareket ettirilen kolların uzunluklarını değiştirerek platformun düzgün bir şekilde hareket etmesini sağlar. Bu sayede, platformun belirli bir noktaya gidebilmesi veya belirli bir eğri üzerinde hareket edebilmesi gibi hassas hareketler yapılabilir.

Stewart Platform, birçok farklı amaç için kullanılabilir. Örneğin, bir hava aracının tasarımı sırasında, platform kullanılarak hava aracının yapısının hassas bir şekilde test edilebilir. Aynı şekilde, gemi inşaatında da platform kullanılarak gemi

yapısının hassas bir şekilde test edilebilir. Stewart Platform, ayrıca mühendislik tasarımı sırasında da kullanılabilir. Örneğin, bir otomobilin tasarımı sırasında, platform kullanılarak otomobilin hareketlerinin hassas bir şekilde test edilebilir.

Stewart Platform, yüksek hassasiyetli hareketler yapabilen bir robotik sistemdir. Ancak, bu hareketlerin hassasiyeti, platformun tasarımına ve kullanılan sistemlere göre değişebilir. Örneğin, hidrolik sistemler daha hassas hareketler yapmaya yönelik olarak tasarlanmış olsa da bu sistemlerin bakım ve onarımı daha zor olabilir. Pnömotik sistemler ise daha az hassas hareketler yapmaya yönelik olarak tasarlanmış olabilir ancak daha kolay bakım ve onarım gerektirir. Bu nedenle, Stewart Platform kullanım amacına ve mevcut sistemlerin avantaj ve dezavantajlarına göre en uygun sistemler seçilmelidir.

Stewart Platformun diğer bir önemli özelliği ise, hareket ettirilen kolların uzunluklarını değiştirerek platformun belirli bir noktaya gitmesini gerçekleştirmektedir. Bu özellik, platformun hareketlerinin daha da hassas hale getirilmesine yardımcı olur. Ayrıca, platformun kollarının uzunluklarını değiştirerek platformun belirli bir eğri üzerinde hareket etmesini de sağlayabilir. Bu sayede, platformun daha esnek bir şekilde hareket etmesi ve daha farklı amaçlar için kullanılması mümkün hale gelir.

Sonuç olarak, Stewart Platform, yüksek hassasiyetli hareketler yapabilen ve geniş bir kullanım alanı olan bir robotik sistemdir. Bu sistem, hava aracı tasarımından gemi inşaatına kadar birçok alanda kullanılabilir ve platformun hareketlerini hassas bir şekilde kontrol edebilme özelliği sayesinde, birçok farklı amaç için kullanılabilir.

1.3.1. SPM çeşitleri

Stewart platform mekanizmaları günümüzde seri, hibrit ve paralel yapıda olmak üzere 3 farklı şekilde bulunmaktadır. Seri ve paralel manipülatör sistemlerinin aksine hibrit model manipülatörler kullanım alanı çeşitliliğinin az olmasından dolayı pek tercih edilememektedirler. Bu durum hibrit model platform sistemleri üzerine yapılan çalışmaları gün geçtikçe artırmaktadır. Bu tez çalışmasına konu olarak alınan stewart platform yapısı paralel stewart platform yapısında bulunmaktadır. D. Stewart, 1965 senesinde pilot adaylarının uçuş eğitimlerini kazanabilecekleri uçuş simülasyon sistemlerinde kullanılmak üzere paralel stewart platform mekanizmasını tasarladı.

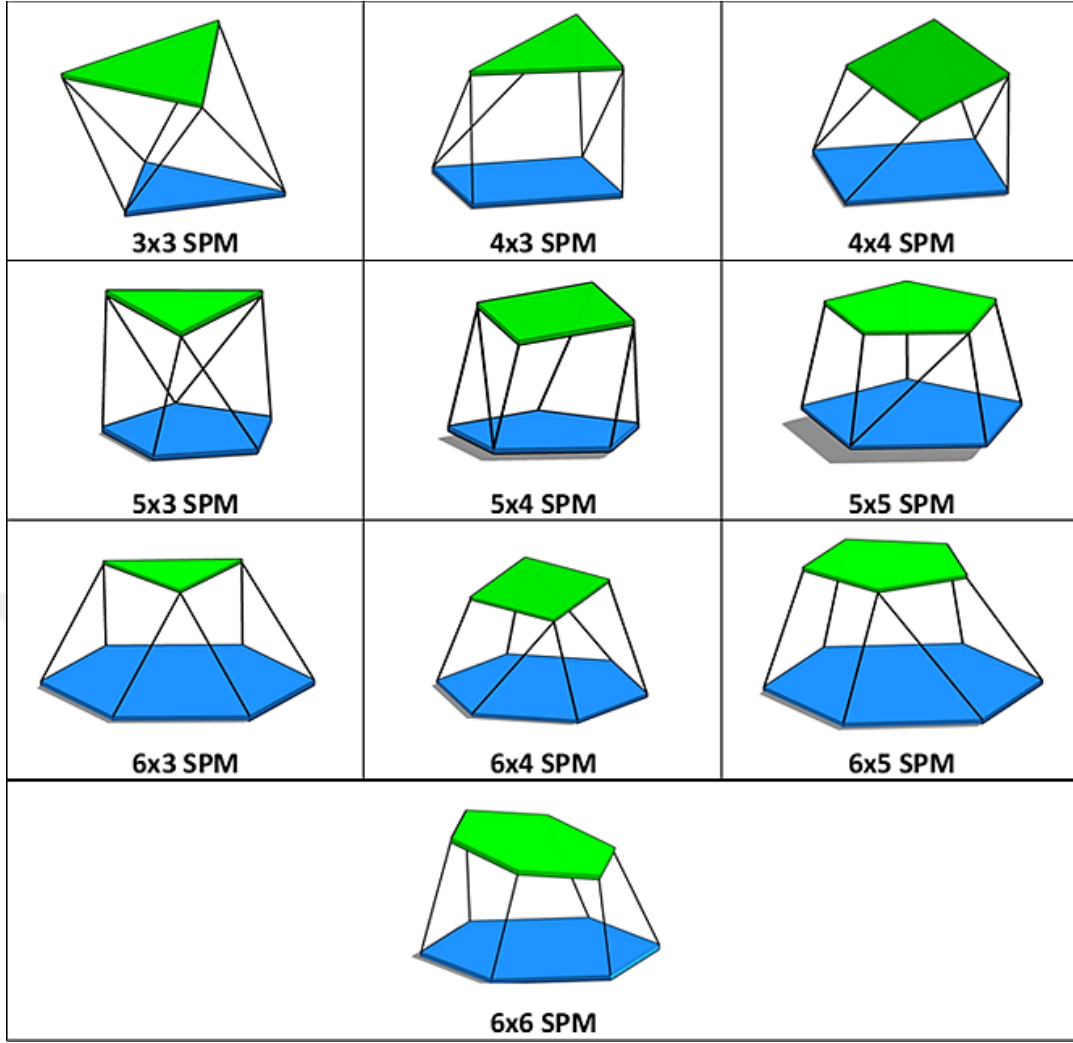
Buna benzer yapıda endüstri, eğlence, denizcilik, biomedikal gibi birçok alanda birçok hizmette kullanılmıştır (Öztürk, 2019).

Seri mekanizma, platformun hareketini, platforma monte edilen birkaç servo motor tarafından kontrol edilen birçok bileşen aracılığıyla gerçekleştirir. Bu tasarım, platformun hareketini kontrol etmek için birçok bileşen kullandığından, yapısı daha karmaşıktır ve bakımı daha zordur.

Hibrit mekanizma, seri ve paralel mekanizmaların özelliklerini birleştirerek, platformun hareketini kontrol etmek için birçok bileşen kullanır. Bu tasarım, platformun hareketini kontrol etmek için daha az bileşen kullandığından, yapısı daha az karmaşıktır ve bakımı daha kolaydır.

Paralel mekanizma, platformun hareketini, platforma monte edilen birkaç servo motor tarafından kontrol edilen birkaç paralel bileşen aracılığıyla gerçekleştirir. Bu tasarım, platformun hareketini kontrol etmek için az sayıda bileşen kullandığından, yapısı daha az karmaşıktır ve bakımı daha kolaydır.

Her üç tasarım da Stewart platform mekanizmasının temel özelliklerini taşır ve hareketli bir platformun sabit bir yüzey üzerinde hareket etmesini sağlar. Hangi tasarımın kullanılacağı, uygulamanın gereksinimlerine ve mevcut kaynaklara göre belirlenir.



Şekil 7. Stewart Platform Mekanizması türleri

Kaynak: Özer, M. (2019).

Stewart platform mekanizması bağlantı kollarının bağımsız bağlanma uç noktalarının sayısına göre 3x3, 3x6, 6x3 ve 6x6 şeklinde farklı yapılarda üretilmektedirler. Bu çalışmada incelenen Stewart platform mekanizması 6x3 yapısında bir platform yapısına sahiptir. Bu Stewart Platform Mekanizması türlerinin çizgi modelleri Şekil 8. 'de gösterilmiştir.

1.3.1.1. 3x3'lük SPM türü

3x3 bağlantılarına sahip olan Stewart Platform Mekanizması yaptığı işlemlerin uzay alanı ve denge yapısının kararlılığı ele alındığında diğer platform mekanizma türlerine göre daha çok tercih edilmektedirler. Burada en önemli nokta bağlantı kollarının tablaya bağlanan uç nokta kısımlarının yeteri kadar sağlam bir şekilde bağlantı noktalarına monte edilmelidir. 3x3 bağlantı yapısına sahip platformlarda

hareketli üst tablaya bağlantı kollarının üç noktada sabitlenmesi ve hareketsiz alt tablaya yine 3 noktadan bağlanarak sistemin iskelet yapısı oluşmaktadır (Öztürk, 2019).

1.3.1.2. 3x6'lık SPM türü

3x6 bağlantısına sahip olan Stewart Platform sistemlerinde hareketli bulunan üst tablaya bağlı toplam 6 noktadan bağlanmaktadır. Hareketsiz bulunan alt tablada ise üst tablaya bağlı bağlantı kolu çiftlerinin altta tek bağlantı noktasında birleştirilmesi esasına dayanmaktadır.

1.3.1.3. 6x3'lük SPM türü

6x3 sistemine sahip Stewart Platform Mekanizmaları 3x6 yapısına sahip platformunun bağlantı noktalarına göre tam tersi bir durum sergilemektedir. Bu yapıda hareketli olan üst tablaya bağlı toplam 3 bağlantı noktasından oluşmaktadır. Hareketsiz olan alt tablada ise üste ikili çift şekilde bağlantı noktasında birleştirilen bağlantı kollarının alt kısmında ayrı ayrı altı bağlantı noktasında bağlanmış halinde bulunmaktadır.

1.3.1.4. 6x6'lık SPM türü

6x6 stewart platform modelinde isminde verildiği gibi alt tabana 6 noktadan bağlı ve üst kısma altı noktadan bağlı olmaktadır. Bacak bağlantı noktaları birbirini dengelediği için oldukça dayanıklı bir yapıya sahiptir (Özer, 2019).

1.3.2. Stewart platformunun kullanım alanları

1965 senesinde D. Stewart tarafından fikrin ana yapısını açıkladıktan sonra birçok farklı alanda insanların çalışma ve geliştirme yapmasına olanak sağlamıştır. Stewart platformu çalışma amacı olarak uçuş simülatörü olarak insan yaşamı için tehlikeli ve riskli görülebilecek bir alanda kullanılmıştır. Stewart Platformu üzerine genel olarak yapılan çalışma alanlarına gözlemlersek;

- Talaş imalatı fabrikalarında yapılan ve oldukça hassas bir işlem olan talaş kaldırılması konusunda,
- Hassasiyet gerektiren cerrahi uygulamalarda,

- Tutma mekanizmalarında,
- Eğlence sektöründe,
- Uçuş simülâtör eğitimlerinde,
- İnsanların kaldıramayacağı ağır yükleri taşımada,
- Nesne ile ilgili hassas konumlandırma işlemlerinde,
- Helikopter vinç yapısında.



Şekil 8. Cerrahi uygulamada
Kaynak: Bonev, I. (2004).

Hegzapod yapıya sahip olan paralel stewart platformları kullanım avantajlarından dolayı birçok alanda kullanılmakta ve üzerinde geliştirme çalışmaları devam etmektedir (Yanık, 2016).

Stewart Platformun kullanım alanları çeşitlidir ve genellikle hareket eden yapıların konumunu kontrol etmek için kullanılır. Bunun yanı sıra, bir otomobil test pistinde bir aracın yol tutuşunu test etmek için de kullanılabilir.

Stewart Platform, ayrıca havacılık ve uzay endüstrisinde de kullanılmaktadır. Örneğin, bir uçağın hareketini simüle etmek için kullanılabilir veya bir roketin fırlatılması sırasında güçlü sarsıntıları azaltmak için kullanılabilir.

Bunun yanı sıra, Stewart Platform, sismik etkilere karşı yapıların dayanıklılığını test etmek için de kullanılmaktadır. Örneğin, bir yapının deprem sırasındaki performansını test etmek için Stewart Platform kullanılabilir.

Sonuç olarak, Stewart Platform, hareket eden yapıların konumunu kontrol etmek için geniş bir kullanım alanına sahiptir ve havacılık, uzay, robotik, otomobil ve yapı testleri gibi çeşitli endüstrilerde kullanılmaktadır.

1.4. Robot Mekanizmasının Kontrol Edilmesi

Robot mekanizmalar, bir tür robotik sistemleri oluşturur. Robotik yapıdaki sistemlerin bazı çevrelerle ve diğer donanım birimleri ile etkileşime girebildikleri bilinmektedir. Bu çevreler, kimi zaman bir insan veya yapay zekayla bütünleşmiş kinematik yapıda veya dinamik yapıda bir sistem olabilir (Fasse ve Broenink, 1999). Robotik kollar günden güne nesne taşıma, lazer kesim işlemleri, kaynak yapma işlemleri ve boyama gibi birçok alanda kullanılmaktadır (Broenink, Visser ve Tierneho, 1998). Yapılacak işlemler için hassas konum tanımlama işlemlerinde için yeterlidir. Bu gibi işlemlerin yanı sıra işlem gören malzemenin zımpara mantığına benzer şekilde çapak alma işlemi, hassas olan nesnede kırılgen yapısının fiziksel olarak bozulmaması adına sabit bir şekilde tutularak delme gibi işlemlerde temas etme durumlarında sistemin yapısına göre geri kuvvet beslemesi yapılmaktadır. (Jinno, Ozaki, Tatsuno ve Takahashi, 1995). Benzer çalışma yapılarında nesnelerin konumsal olarak yer değiştirme durumları işlemlerinde sabitleme gibi bir pozisyon kontrolü yapılması gerekiyorsa gerekiyorsa, hassas konumlandırma işlemi ile dinamik çalışma yapısına sahip robotlardaki kararlılığa ve konumlandırma işlemlerindeki hassasiyeti

etkileyebileceği bilinir ise bu çalışmalarda en uygun işlem olacağı önem kazanmaktadır. (Yoshikawa, 1990).



Şekil 9. Piston modellerine örnekler

Kaynak: Özer, M. (2019).

Manipülâtör sistem tasarımının çalışma esnekliđi ve esneklik ile birlikte yapılan hareketin hassasiyeti üzerinde 1970’li yılların başından itibaren çeşitli araştırma ve çalışma bulunmaktadır. Hareketlerin esnekliđi ve esneklikle birlikte hassasiyetin geliştirilmesi adına birçok çalışma ve bu çalışmalar ile birlikte birçok teori sunulmuştur. Bu çalışmalar birçok farklı amaca ufuk açacak yeni modeller ortaya çıkarmıştır. Bu yeni modellerin temellerini atan çalışmalardan bir tanesini de Salisbury ve Craig yapmıştır. Salisbury ve Craig yaptıkları manipülâtör sisteminde sistemin kuvvet geri beslemesi ile orantılı bir şekilde çalışacak ve bu çalışmasına göre geri beslemeye göre uygun bir pozisyon yörüngesini temel almış ve bu pozisyona uygun kontrol modeli çıkarmıştır (Salisbury, 1980). Bu çalışmadan farklı bir çalışma ise Paul, Whitney ve Shimano tarafından gerçekleştirilmiştir. Paul, Whitney ve Shimano sistem uzuv noktasının uç noktasında oluşan hız parametresinin kontrol edilmesine yönelik

sönüm parametresi seçeneğini önermişler. İki çalışmadan farklı olarak farklı bir çalışmada Hogan tarafından gerçekleştirilmiştir. Hogan yaptığı çalışmada ise ilk iki yöntemden farklı olarak iki yöntemin iç içe geçmiş hali şeklinde empedanslı kontrol sistemini önerisinde bulunmuşlardır (Benali, Richard ve Bidaud, 1999). Hogan, deneysel çalışmalarının amacı olan empedans kontrolünü kullanmasının amacı, yaptığı çalışmanın etkileşim anı ile birlikte çalışma uzayına olan geçişi belirlemede kullanmıştır. Yaptığı bu çalışmaya yaklaşımla kararlılık sağlama doğal çevre katılığına göre sonuç vermesinin yanında bu çalışmada ters kinematik hesaplama tekniklerini kullanma gereği duymamıştır. (Hogan, 1985). Yapılan çalışma ve deneylerin bir sonucu olan bilimsel araştırmaların çıkarmış olduğu sonuca bakacak olursak sertlik, güç, direnç, değişken, pozisyonu ve gücü şeklinde karşımıza çıkmaktadır. Bu çalışmaların çıkardığı sonuç ve bilgilerin sonucuna göre Raibert, Mason ve Craig yaptıkları araştırmaların sonucunda robotik sistemlerde, konumlandırılmalarını ve sonlandırma kuvvetlendirmelerini bir kontrol algoritması ile birlikte kontrol edilmesini sağlamışlardır (Peng, 1992).

Stewart Platform Mekanizması çalışmaları ve bu yapıda çalışan paralel robot sistemleri kapalı çevrime çalışma yapısına sahip sistemlerdir. Kapalı çevrim çalışma yapısındaki sistemlerde bağlantılarından dolayı çözüm algoritmalarının kontrol yapılarının daha karmaşıklaştırır (Yoshikawa, 1990). Seri robot bundan önceki belirtmelerimizde de olduğu gibi uzuv yapılarının birbiriyle bağlantılı olarak ard arda gelmesi ile oluşmaktadır. Seri sistem yapılarında bağlantıları seri halde uygulanması mafsal ve uzuv bağlantı kollarını oluşturan yapısında toplanmasına ve uç noktada istenmeyen sonuçların çıkmasına neden olmaktadır. Bu çalışma yapısı ile seri mekanizmaların çalışma uzaylarında öngörülme veya istenmeyen sonuçlar doğurarak performansı olumsuz olarak etkilemektedir. Bu çalışmaya ters olarak paralel yapıya sahip mekanizmaların çalışmalarında ise uzuvların uç işlemlerinde hata toplamamasının yanı sıra ortalamasını almaktadır. Paralel mekanizmalarda uç işlevci hassasiyeti değerlendirildiğinde değerlendirmede paralel ve seri mekanizma uç işlevleri ele alınırsa paralel mekanizmaların hassasiyeti bakımından daha iyi sonuçlar doğurmaktadır (Benali vd., 1999).

Robot manipülatörler, hareket edebilen robotlar olarak tanımlanabilirler. Bu robotlar, mekanik parçaları ve motorları sayesinde hareket edebilirler ve belirli

görevleri gerçekleştirebilirler. Robot manipülatörler, çeşitli endüstriyel uygulamalarda kullanılabilirler ve bu uygulamalar arasında üretim, imalat, paketlenme, nakliye ve depolama gibi alanlar bulunur. Robot manipülatörler, ayrıca test ve ölçüm görevleri de yapabilirler. Robot manipülatörlerin kontrolü, genellikle bir kontrol sistemi aracılığıyla gerçekleştirilir. Bu kontrol sistemi, robotun hareketlerini ve görevlerini yönetir. Kontrol sistemi, robotun sensörlerine ve hareket mekanizmalarına bağlıdır ve bu sensörler ve mekanizmalar aracılığıyla robotun dünya ile etkileşimini sağlar. Kontrol sistemi, ayrıca robotun programlanmış hareketlerini ve görevlerini gerçekleştirir. Robot manipülatörlerin kontrolü, genellikle bir yazılım aracılığıyla gerçekleştirilir. Bu yazılım, robotun hareketlerini ve görevlerini yönetir ve robotun sensörlerine ve hareket mekanizmalarına bağlıdır. Bu yazılım, ayrıca robotun programlanmış hareketlerini ve görevlerini gerçekleştirir. Robot manipülatörlerin kontrolü, ayrıca bir kumanda paneli veya bir bilgisayar aracılığıyla da gerçekleştirilebilir. Bu kontrol yöntemleri, genellikle daha yüksek seviyede bir kontrol ve özelleştirme imkânı sağlar. Kumanda paneli ve bilgisayar, robotun hareketlerini ve görevlerini yöneten yazılımları kullanarak robotun daha kompleks hareketlerini ve görevlerini gerçekleştirme imkânı verir. Bu yöntemler, ayrıca robotun performansını izleme ve değerlendirme imkânı da sağlar.

Sonuç olarak, robot manipülatörlerin kontrolü, bir kontrol sistemi, yazılım, kumanda paneli veya bilgisayar aracılığıyla gerçekleştirilebilir. Bu kontrol yöntemleri, robotun hareketlerini ve görevlerini yönetir ve robotun sensörlerine ve hareket mekanizmalarına bağlıdır. Bu yöntemler sayesinde, robot manipülatörler çeşitli endüstriyel uygulamalarda ve test ve ölçüm görevlerinde kullanılabilir.

1.4.1. Geometrik eniyileme

Paralel mekanizmalar için tip sentezleme araştırmaları yapılmaktadır. Bu araştırmaların amacı, sistemin teorik çalışmalarından pratik uygulamalara aktarılması için gerekli olan bilgileri belirlemektir. Bu bilgi, sistemin nerede ve nasıl çalışacağı ve hangi amaçlar için çalışacağı hakkında kinematik, dinamik ve statik hesaplamaları içerir ve bu hesaplamalar, paralel sistemin mekanik ölçülerinin nasıl çalıştırılacağını belirler. Ayrıca, tasarım sırasında çalışma alanı ve tasarım modeli belirlenirken, yapıyı oluşturan parametreler geometrik tasarıma uygun olarak belirlenmelidir. Bu amaçla, mekanizmadaki kinematik hesaplamaları en verimli şekilde işlemek için optimize

edici çalışmalar gerçekleştirilir. Optimizasyon, tasarım yapısının parametre gruplamaları tarafından tanımlanan olası durumlardan en iyi olanı bulma amaçlıdır. Bu konudaki çalışmalar aşağıdaki gibi sıralanabilir:

- Paralel robot sistemlerinde programlama, eniyileme ve kinematik hesap analizleri üzerine odaklanan çalışmalar,
- Kendine özgü bir çalışma uzayı için delta robotların eniyilemesini kullanan çalışmalar,
- Genetik algoritma kullanılarak Stewart platform mekanizmasının altı serbestlik derecesine sahip eniyileştirilmesi üzerine yapılan çalışmalar,
- Paralel robot sistemlerinin geometrik duruşunun eniyileştirme ve tekil nokta uzaklığına göre çalışma uzayını inceleyen araştırmalar,
- Paralel robot sistemlerinin özelliklerini optimize etme amacıyla yapılan çalışmalar yer almaktadır.

Cheng ve arkadaşları, Stewart platform mekanizmasının eniyilemesi için kullanılan kontrol algoritmalarını incelemişlerdir. Çalışmalarında, çeşitli kontrol algoritmalarının, mekanizmanın hareketlerinin doğruluğu ve verimliliği açısından karşılaştırılmasını yapmışlar ve sonuçlarını değerlendirmişlerdir. Ayrıca, Zha ve arkadaşları, paralel robot sistemlerinin eniyilemesi için genetik algoritma ve çok değişkenli optimizasyon yöntemlerini kullanan çalışmalar yapmışlardır. Bu çalışmalarında, iki yöntemin karşılaştırılmasını yapmışlar ve sonuçlarını değerlendirmişlerdir.

Diğer yandan Li ve arkadaşları, paralel robot sistemlerinin kinematik özelliklerini optimize etme amacıyla yapılmış olan çalışmaları incelemişlerdir. Bu çalışmalarında, paralel robot sistemlerinin kinematik özelliklerinin optimize edilmesi için kullanılan yöntemleri ve bu yöntemlerin avantajlarını ve dezavantajlarını araştırmışlar ve sonuçlarını değerlendirmişlerdir (Ou, 2005). Stewart platformlarının ve paralel mekanizmaların geometrik eniyilemesi konusunda yapılan çalışmaların birçoğu, platformların ve mekanizmaların hareketlerinin optimize edilmesi amacıyla yapılmıştır. Bu çalışmalar arasında, Stewart platformlarının hareketlerinin doğrusal olmayan dağılım göstermemesi gibi kinematik problemleri çözme amacıyla yapılan çalışmalar, paralel robot sistemlerinin programlanması, eniyilemesi ve kinematik

hesap analizleri üzerine odaklanan çalışmalar, kendine özgü bir çalışma uzayı için delta robotların eniyilemesini kullanan çalışmalar, genetik algoritma yardımıyla stewart platform sisteminin altı serbestlik derecesine sahip halinin eniyileştirilme çalışmaları, paralel robot sistemlerinin geometrik yapısını optimize eden ve bu mekanizmaların tekil noktanın uzağında olan çalışma alanını inceleyen araştırmalar, paralel robot sistemlerinin özelliklerini optimize etme amacıyla yapılan çalışmalar gibi çeşitli çalışmalar yer almaktadır. Bu çalışmaların birçoğu, Stewart platformlarının ve paralel mekanizmaların hareketlerinin doğrusal olmayan dağılım göstermesi gibi kinematik problemleri çözme amacıyla yapılmıştır. Bu problemlerin çözümü için, çeşitli optimizasyon yöntemleri ve algoritmaları kullanılmaktadır. Örneğin, Zha ve arkadaşları, paralel robot sistemlerinin eniyilemesi için genetik algoritma ve çok değişkenli optimizasyon yöntemlerini kullanan çalışmalar yapmışlardır. Bu çalışmalarında, iki yöntemin karşılaştırılmasını yapmışlar ve sonuçlarını değerlendirmişlerdir (Ou, 2005).

Diğer yandan, Li ve arkadaşları, paralel robot sistemlerinin kinematik özelliklerini optimize etme amacıyla yapılmış olan çalışmalarını incelemişlerdir. Bu çalışmalarında, paralel robot sistemlerinin kinematik özelliklerinin optimize edilmesi için kullanılan yöntemleri ve bu yöntemlerin avantajlarını ve dezavantajlarını araştırmışlar ve sonuçlarını değerlendirmişlerdir.

1.4.2. Robot sistemlerinde mobilite

Teknoloji çağımızda sürekli gelişim mobilite kavramını çıkarmıştır. Tanım olarak bir nesnenin hareketliliği ya da esnekliği şeklinde tanımlanma yapılsada günümüzde akıllı giyilebilir sistemlerde mobilite olarak tanımlanmaktadır. Robot sistemlerinde ise mekanizmaların hareketliliği mobilite olarak tanımlanmaktadır. Sistemlerde mobilitenin verimli çalışması için elde edilen formüllerin analizi ve sonuçlandırılmasının doğruluğu adına mekanizmadaki tüm uzuv boyutları ve eklemlerin açısal konumlarının noktasal şekilde belirtilmesi gerekir. Sistemin bu gibi parçalarının bağlantıların çalışmaya yapacağı etki çalışma uzayına yansıtacağı için büyük bir öneme sahiptir. Çalışma uzayına etki edecek hareketlerin nasıl yapılacağı, bu parçaların birbirlerine nasıl bağlandığıyla ilişkilidir ve bu nedenle önemlidir.

Mobilite, bir robot sisteminin hareket kabiliyeti ve esnekliğidir. Bu, robotun farklı alanlarda ve koşullarda hareket edebilme yeteneğini, farklı yükler taşıyabilme ve farklı tasarımlar kullanarak görevleri gerçekleştirme yeteneğini ifade eder. Mobilite, bir robot sisteminin görevlerini gerçekleştirirken önemli bir rol oynar ve robotun performansını etkileyebilir. Örneğin, bir robotun çevresinde daha esnek hareket edebilmesi, daha geniş bir alanda görevleri gerçekleştirilmesine yardımcı olabilir. Ayrıca, bir robotun hareket kabiliyeti, onun güvenliği için de önemlidir. Örneğin, bir robotun kolayca takılıp düşebileceği yerleri tespit etmesi ve bu alanlardan uzak durması gerekebilir.

1.5. Çalışma Uzayı

Mekanik sistemlerde hareketli yapının oluşturduğu tüm işlem hareketleri çalışma uzayını temsil etmektedir. Paralel manipülatör sistemlerinde çalışma uzayının sınırlarını, hareketli kısımda bulunan yapının uç sınır noktaları belirlemektedir. Sistemin hareketli kısmının oluşturduğu koordinat noktalarının tüm konumsal kombinasyonların oluşturduğu kümeye çalışma uzayı denir. (Tsai, 1999; Alp, 2007) Paralel sistemlerin geometrik çalışmasına bağlı olarak birkaç farklı model ile çalışma uzayı belirlenmektedir. Bu yöntemler; ayrıklaştırma yöntemi, geometrik ve sayısal yöntemlerdir. Tasarımın oluşturması gerektiği çalışma uzayını belirlemede bu yöntemlerden uygun olan kullanılmaktadır. (Merlet, 2006; Franck, Vidala, Nicholas, Derek, Gould ve Healey, 2005).

Paralel mekanizmalar, bir nesnenin hareketini birden fazla kol veya bileşen aracılığıyla gerçekleştirirler. Bu tür mekanizmalar, genellikle hareketli nesnelerin hareketini kontrol etmek veya bir nesnenin hareketini bir başka nesneye aktararak güç çevrimini sağlamak amacıyla kullanılırlar. Paralel mekanizmalarda çalışma uzayı, paralel mekanizmaların hareketlerini modelleme ve analiz etmek için kullanılan bir kavramdır. Bu uzay, mekanizmanın tüm kolları ve bileşenleri için bir koordinat sistemine sahiptir ve her bir kol veya bileşen için bir koordinat değişkeni tanımlar. Bu koordinat değişkenleri, mekanizmanın konumunu ve hareketini tanımlar ve çalışma uzayında çözüm bulunarak, mekanizmanın hareketleri modellenir ve analiz edilebilir.

Genellikle, paralel mekanizmaların hareketlerini anlamak ve tasarlamak için, çalışma uzayının geometrisi ve dinamiği üzerine ayrıntılı bir anlayışa sahip olunur.

Bu, mekanizmanın kinematik ve dinamik özelliklerini belirlemek ve önceden tasarlamak için gereklidir.

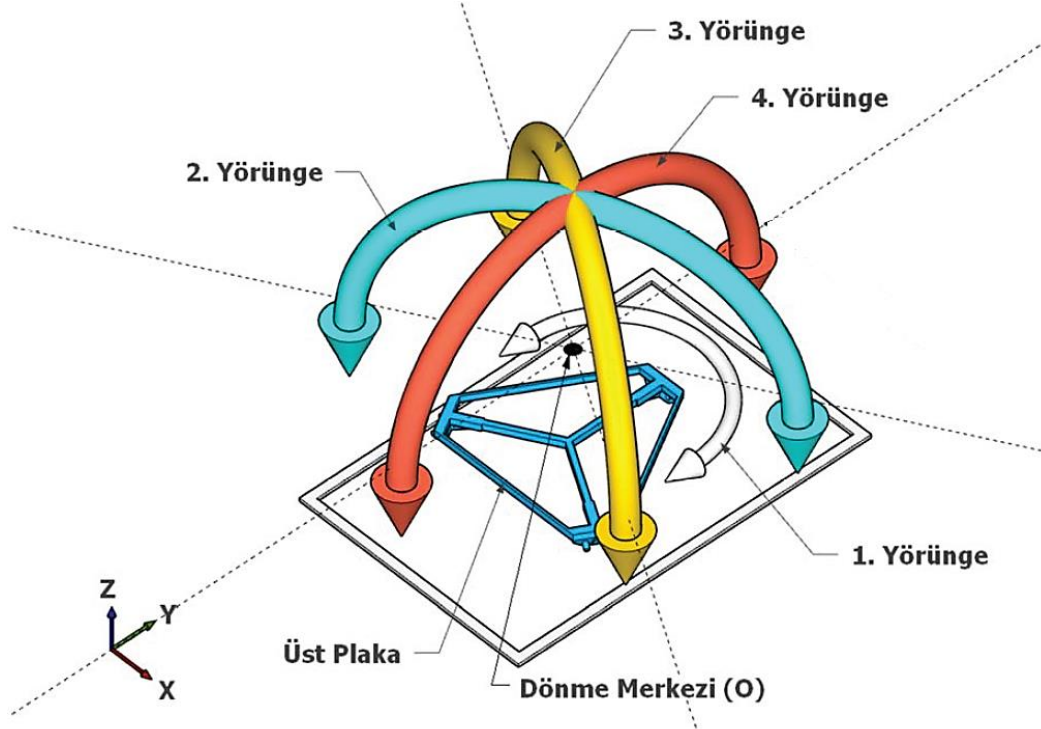
1.5.1. Stewart platform mekanizması ile sağlanan doğrusal hareketler

Stewart Platformu doğrusal hareketler yapabilen bir sistemdir. Bu, platformun herhangi bir yöne doğrusal olarak hareket etmesi anlamına gelir. Örneğin, platform yatay bir şeritte hareket edebilir veya dikey bir şeritte hareket edebilir. Bu hareketler, platformun hareketlerini kontrol eden kontrol sistemine göre yapılır ve platformun pozisyonunu ve hareketlerini ölçmek için sensörler kullanılır. Stewart Platform, çeşitli uygulamalar için kullanılabilir. Örneğin, havacılık ve savunma sektörlerinde simülasyon sistemlerinde kullanılır. Ayrıca, otomotiv sektöründe test merkezlerinde ve endüstriyel robotik sistemlerinde de kullanılabilir. Bu platform, hareketi ve pozisyonu kontrol etmek için kullanılan bir sistem olduğu için, doğrusal hareketler yapabilen bir sistemdir.

Stewart paralel platform mekanizmalarının çalışma yapısının temelinde doğrusal ve açısal hareketler bulunmaktadır. Paralel platformun doğrusal yaptığı hareketlerde ana eksen bloğunda bulunana x , y , z olarak isimlendirilen ve herhangi bir açı tanımlaması olmayan ve uzaklığı sabit olarak bir yönde yer değiştirmesi ile gerçekleştirir. Bu hareketler literatür taramalarında Sway, Surge, Heave olarak adlandırılmaktadır.

1.5.2. Sistemdeki açısal hareketler

Stewart paralel platform mekanizmalarının çalışma yapısının temelindeki diğer bir yapıda sistemin açısal olarak hareket yaptığı durumdur. Genel olarak Stewart Platformlarının yapısındaki üst hareketli plakanın merkeze bağlı olarak açısal hareketi aşağıdaki şekilde verilmiştir (Akkaya, Özer, Şentürk, Öztürk ve Alkan, 2022).



Şekil 10. O merkezine göre platformun açısall hareketleri
Kaynak: Özer, M. (2019).

1.5.3. Kısıtlı çalışma uzayına sahip mekanizmalar

İçinde bulunduğumuz zamanda serbestlik derecesi olarak altı serbestlik dereceli sistemlere göre serbestlik derecesi sayısı az olan sistemler kısıtlanmış bir çalışma uzayı yapısına sahip olmaktadır. Bu kısıtlı çalışma uzayına sahip sistemler endüstriyel alanlarda maliyet bakımından fazla veya çok sayıda aynı işlem ile üretilen ürünlerde önemli derecede yer tutmaktadır. Altı serbestlik derecesine sahip sistemlere göre kısıtlı çalışma alanına sahip sistemler hızlı olmaktadır. Diğer sistemlere göre daha ağır yüklerde daha verimli ve hızlı bir performans göstermektedir. Bu tür sistemlerin teknik olarak montaj kolaylığı ve yazılımsal olarak verilen komutların karmaşık olmaması avantajları arasında yer almaktadır (Öztürk, 2019).

1.5.4. Çalışma uzayında yörünge takip planlaması

Günümüzde, engelleri aşan veya yörüngeleri takip eden sistemler, araştırma konuları arasında önemli bir yer tutmaktadır. Özellikle en uç noktadaki uç işleveciye bağlı hareketli üst plakanın hareketi sırasında anlık oluşan titreşimler önemlidir. Bu titreşimler, istenilen sonuca ulaşılmasına rağmen, farklı açı ve konumları da beraberinde getirebilir. Yörünge takip planlaması, hızlı platform sistem yapılarında üst

plakanın uygulanan konum ve açılarını sağlamak amacıyla yapılır. Bu hareket uzayı doğru uygulandığında, üst plakanın istenilen hareketleri yaptığı sırada bağlantı bacakları arasındaki hız farkı optimize edilir ve sistemin son kullanıcının isteği doğrultusunda hareketler yapması sağlanır. Bu nedenle, yörünge takibi hesaplamalarını doğru yapmak önemlidir (Bingü ve Küçük, 2005; Kızıllan, Toz, Aliustaoğlu ve Bingül, 2007).

1.5.5. Değişken uzay yönteminin çalışma uzayına etkisi

Paralel yapıdaki manipülatör için sistemleri için oluşturulan çalışma uzayı, geometrik, ayırıklaştırma ve sayısal yöntemlerini kullanır. Matematiksel bu işlemler tasarımın işlev doğruluğunu verir. Burada belirtilen üç yöntem, tasarım açısından da yeterli görülür (Merlet, 2006; Merlet, 1998).

Bu yöntem, çalışma alanı içinde belirli bir sayıda koordinat noktasının seçilerek hareketli tabla sisteminde bulunabilecek konumların hesaplanmasını ve kullanılabilir geometrik tasarım aralıklarının verilmesini içerir. Bu parametre uzayı yöntemi, aralıklar kullanarak hesaplamaların süresini kısaltmayı amaçlar. Paralel sistemlerde ise, sınırsız sayıda parametre ele alınabilir ancak bu parametrelerin artışı hesaplamaların süresini uzatır. Algoritma, belirtilen noktalar tarafından oluşturulan köşeler üzerinden doğru bir prizmanın işlev göstermesini sağlamak için her bir köşe noktasına farklı hesaplamalar uygular ve sonuçlar sağlar. Eğer bu seçilen noktaların oluşturduğu sınır değerleri doğrulanırsa, prizma kabul edilir. Bu parametrede belirlenen değerlerin oluşturduğu prizma herhangi bir parametre kümesini çalışma alanında kullanmaya izin verir. Eğer çalışma alanında doğrulanmayan tek bir nokta bile bulunursa, bulunduğu bölge için maksimum ve minimum sınırlar belirlenir ve bu sınırların değerlerine göre bulunduğu kenar alanı ikiye bölünür. Bölünme işlemi sonucu büyük boyutlarda ve küçük boyutlarda prizmalar oluşur ve küçük prizma üzerinden devam eden algoritma büyük prizma yapısını ele almaz. Bu bölmeler devam ettirilir ve elde edilen prizmaların kenarlarının uzunluğu sınır olarak belirlenen bir eşik değerinin altına indiğinde işlemler sonlandırılır. Eğer, ele alınmayan prizmaların kenar uzunlukları eşik olarak belirtilen sınırın altına iner veya bu durumda ise gerçekleşen algoritma işlemi sona erer.

1.6. Ters Kinematik Analiz

Paralel manipülatörlerin konumunu belirlemek için iki yöntem mevcuttur. Bunlardan ilki ters kinematik hesaplamanın yanı sıra ikincisi ise düz kinematik hesaplama yöntemleridir (İnal, 2019). Ters kinematik analiz, manipülatörün talep edilen konumu ve doğrultuyu gerçekleştirmesi için bağlantı kollarının uzunluk ve hızlarını belirlemeyi amaçlar. Bu yöntem, seri robotlar için zor çözümler üretir ve nümerik metotların kullanımını zorunlu kılar. Ancak, paralel manipülatörler için çözümler analitik olarak hesaplanabilir ve kolaydır. Bu nedenle, ters kinematik analiz, kontrol uygulamaları için anahtar yöntemdir (El-Badawy ve Youssef, 2013). Düz kinematik analiz ise eyleyici ve bağlantı kollarında ele alınan uzunluk verilerinin platforma ne gibi bir konum ve bu konuma göre nasıl yönelim gerçekleştireceğini inceler. Düz kinematik hesaplamalar kullanılarak gösterilen yaklaşımda, seri yapıda oluşturulan robot sistemleri için daha anlaşılır analitik sonuç vermektedir. ancak paralel robotlarda veya manipülatör sistemlerinde bu durum tam tersi olarak sonuçlar verir ve karmaşık denklemlerin sonucu olan çözülmesi zor bir hal alır. Bu nedenle, nümerik yöntemlerin kullanımı gerekir (Lebret, Liu ve Lewis, 1993).

1.7. Düz Kinematik Analiz

Robotik sistemlerde, uç noktada bulunan sistem parçalarının konumları ve hareketleri, düz kinematik bağlantılar arasında bulunan eklemler sayesinde sistemin yapısı ele alınarak hesaplanır. Örneğin, Stewart mekanizmalarında düz kinematikleri hesaplamak için farklı iki metot ele alınır. Bunlardan birincisi sayısal yöntem ve ikincisi ise analitik yöntemdir. Bu yöntemler, sistem parçalarının konumlarını ve hareketlerini doğru bir şekilde hesaplamak için kullanılır. (Merlet, 1993).

Analitik metot, verileri kullanarak bir konuda doğru sonuçlara ulaşmak için kullanılan bir yöntemdir. Bu yöntem genellikle "polinom metodu" ismi ile bilinen karmaşık ve zor bir algoritma ile hesaplanır, fakat bu metot senkron yapıda işlem gören kontrollerde yavaş olduğundan genellikle talep edilmeyen bir metottur. Örneğin, hareketli bir paralel platform sistem için uzunlukları aynı olan bağlantı kollarının 8 tane muhtemel konumu oluşturabileceği düşünülürse, yörüngede doğru olan konumu bulmak adına, muhtemel konumu veya konumları ele almaması gerekmektedir.

Neticede sistemdeki hesaplamaları düz kinematik ile yapmak işlemleri zor hale getirmektedir (Öztürk, 2019).

1.8. Dinamik Analiz ve Modelleme

Mekanizmaların hareketlerini inceleyen dinamik analizler, kapalı zincir yapısı ve bağlantı kolları etkilerini değerlendirir. Dinamik modelleme yöntemleri arasında, kabullerin yapıldığı tahmini yöntemler ve tam hesaplanan dinamik yöntemler bulunur. Tahmini dinamik analizler, bacakların ataletlerini göz ardı eder ve kütleleri ağırlık merkezlerinde nokta kütle olarak kabul eder. Bu tip kolaylaştırmalar, çözümleri hızlandırır ancak sistemi tam olarak tanımlamaktan uzaklaşmayı da beraberinde getirir ve bu durum tasarlanacak kontrol sistemlerinin etkisini azaltabilir. Matris ve vektörel hesaplamalar, dinamik analizleri daha kolay çözülebilir hale getirir (İnal, 2019).

Stewart platformunun dinamik denklemleri Newton-Euler yöntemi kullanılarak tam dinamik yaklaşım kullanılarak elde edilmiştir (Dasgupta ve Mruthyunjaya, 2000). Bu yöntem, uygulanabilir olmasının yanı sıra, hareketleri, hızları, sınırlamaları ve momentleri de hesaplamaya gerek duyar. Bunun yanı sıra, başka çalışmalarda, dinamik denklemler Lagrange yöntemi kullanılarak elde edilmiştir. Bu yöntemler sayesinde, sistemin eylemsizliği, enerji kaybı ve yerçekimi etkisinin tesir ettiği yönlerin matris verileri elde edilebilir. Ele alınan bu matrisler, dinamik denklemleri kontrol uygulamalarına uygun hale getirmeyi mümkün hale getirir (Lebret vd., 1993). Lagrange yöntemi ile bir mekanizmanın kuvvetlerinin ve ivmelerinin bilinmesine gerek kalmadan dinamik denklemleri çıkarılabilir. (Kim, Kang ve Lee, 2000; Harib ve Srinivasan, 2003).

Stewart platformunun dinamik analiz ve modeli farklı yöntemlerle elde edilebilir. Öncelikle, vida teorisi kullanılarak yapılan çalışmalar mevcuttur. Bu çalışmalar sayesinde bu platformun dinamik analizi ve dinamik modeli elde edilebilir (Gallardo, Ricoi Frisoli, Checcacci ve Bergamasco, 2003).

Yapılan çalışmalarda, kütle eylemsizlik momenti matrisinin köşegen bir matris şeklinde kabul gördüğü ve kütlelerinin merkezinin platformun merkez noktasında bulunduğu ve (bu, diğer çarpım eylemsizlik momentlerinin önemsenmediği anlamına gelir) varsayımları yapılmıştır. Bacak eylemsizliği de, fiziksel yapısı bunu mümkün kılar ve makul bir koşul varsa, ihmal edilebilir. Bununla birlikte, bazı çalışmalar bacak

eylemsizliklerinin hesaplara dahil edilmemesi durumunda sistemin hassasiyetinde kayıpların olma olasılığını ortaya koymuştur (İnal, 2019; Ji, 1993).

Genel olarak yapılan çalışmaların sonuçlarına göre, Newton-Euler ve Lagrange yöntemleri Stewart platform mekanizmalarında dinamik denklemlerin türetilmesinde ve kontrol sistemi tasarım çalışmalarında önemli bir rol oynamıştır (Harib ve Srinivasan, 2003; Guo ve Li, 2006). Bu yöntemler, benzetim yoluyla kullanılarak, platformun hareketlerinin modellenmesi ve kontrol edilmesinde kullanılmıştır.



İKİNCİ BÖLÜM

SİMULASYON VE KİNEMATİK ANALİZ

2.1. Kinematik Hesaplama Metodları

Paralel mekanizmaların kinematik analizi, iki kinematik analiz yöntemiyle yapılır. Bunlardan ilki olan, düz kinematik hesaplamadır ve bağlantı kollarının uzunluk değerleri girildiğinde üst hareketli platformun altta bulunan sabit platforma nazaran öteleme ile yönelme konumunu bulmayı amaçlar. Bu, hareketli olan üst platformun bağlantı noktalarını çıkaran vektörlerin ve hareketli tablanın koordinatları ile hareketsiz tablanın koordinat sistemine geçişini gerçekleştiren dönme hareketinin matrisinin bulunmasını gerektirir. Bacak boyları belirlendiğinde platformun alabileceği tüm konfigürasyonların bilinmesi gerektiği için önem taşımaktadır. Diğeri ise ters kinematik hesaplamadır ve üst platformun konum ve yönelmesi bilindiğinde, o konum ve yönelmeyi sağlayacak bacak boylarının bulunmasını hedefler. Seri ve paralel mekanizmalar arasında, düz kinematik hesaplamasının kolay ve ters kinematik hesaplamasının zor olduğu seri mekanizmaların aksine, paralel mekanizmalar için düz kinematik hesaplamasının zor ve ters kinematik hesaplamasının kolay olduğu söylenebilir. Paralel mekanizmaların düz kinematik hesaplamalarının tek bir çözümü olmadığı, birçok araştırmacının çalışmalarına göre belirlenmiştir. Bu, bir girişin, yani bacak boylarının birden fazla konfigürasyon ile eşleşebileceği anlamına gelir. Ters kinematik hesaplamaların ise tek bir çözümü olduğu anlamına gelir, yani bir üst platformun bir konfigürasyonu yalnızca bir giriş ile oluşabilir (Ünsal, 2007).

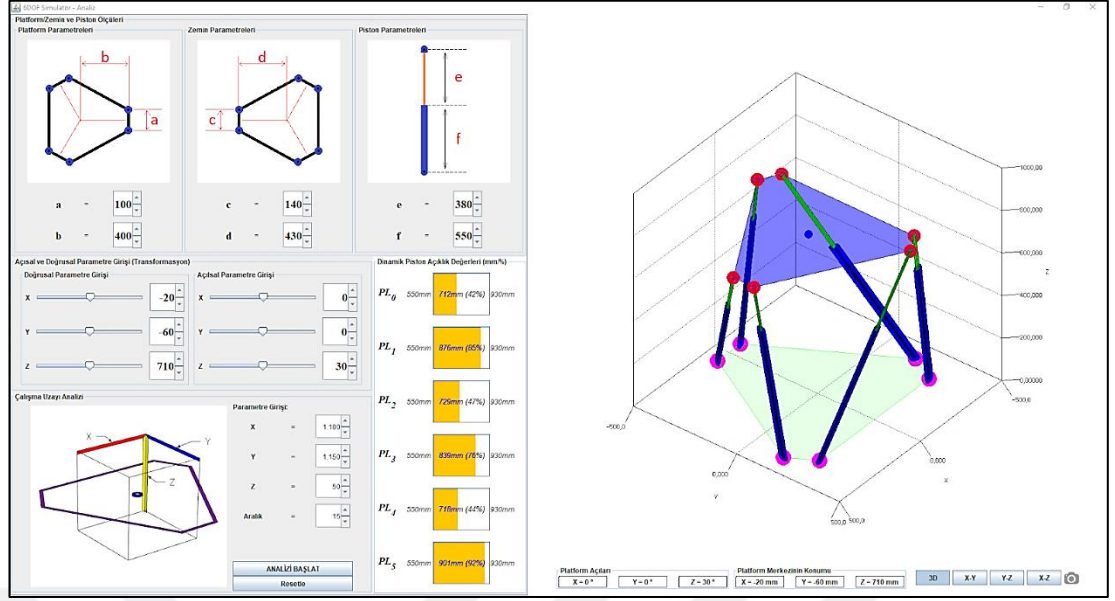
Çalışmadaki Stewart platform hesaplamaları, ters kinematik denklemlerle yapılmış ve simülasyona yazılımsal olarak uygulanmıştır. Stewart platform mekanizmaları, bir sabit zemin üzerine monte edilmiş üç boyutlu bir kinematik yapıya sahiptirler ve hareket ettirilmek istenen cisim taşımak için altı adet kol ve bu kolların uçlarına monte edilmiş olan altı adet işlevci kullanılır. Ters kinematik analizi, mekanizmanın hareketlerinin belirlenmesi için kullanılan bir yöntemdir ve hareket ettirilmek istenen cismin yerleştirilmesi için gereken işlevci hareketlerinin hesaplanmasını sağlar.

Trigonometrik yöntemlerle uç işlevci hareketleri hesaplanabilir. Bu yöntemde, işlevci hareketlerinin yönünü ve uzunluğunu belirlemek için trigonometrik

fonksiyonlar kullanılır. Matris çarpım yöntemi de benzer şekilde kullanılabilir. Bu yöntemde, işlevci hareketlerinin yönü ve uzunluğu için matrisler kullanılır ve bu matrisler arasında çarpım işlemi gerçekleştirilerek hareketler hesaplanır. Ters kinematik analizi, trigonometrik yöntemler ve matris çarpım yöntemi gibi yöntemlerle hesaplanan işlevci hareketleri sayesinde, bu mekanizmalar kontrollü bir şekilde hareket ettirilebilir ve cismin istenen yerleşimine ulaştırılabilir.

Bu çalışmada, benzetim yöntemlerini kullanarak tasarımı gerçekleştirilen bir model ile özgün bir platform oluşturulmuştur. Bu platform, Stewart mekanizmasından farklı olarak, eyleyiciler tarafından harekete geçirilerek, kol uzunlukları değiştirilerek hareket yapabilme özelliğine sahiptir. Modelin altı bağlantı kolu bulunmaktadır ve bu kolların bir ucu sabit zemine, diğer ucu ise hareketli üst platforma bağlıdır. Bu platform, altı serbestlik derecesine sahip bir yapıdır.

Çalışma kapsamında, özgün tasarıma sahip bir platform yazılımı geliştirilmiştir. Bu sistem, modellenmiş, benzetilmiş ve çalışma uzayı analizleri yapılarak incelenmiştir. Sistemdeki hareketler, eyleyicilerin kol uzunluklarının değişimine göre yapılarak, platformun çalışma uzayı içerisinde istenilen konuma ulaşmasını sağlar. Bu tez çalışmasında ele alınan simülasyon, zemine bağlı bulunan bağlantı kolları aracılığıyla yenilikçi bir platform modeli oluşturur. Tüm öteleme ve dönme hareketleri için, çalışma uzayı içerisinde yapılan geometrik çözümlerle yardımcıyla ters kinematik hesaplamaları gerçekleştirilmiştir. Bu çalışmada, literatür araştırmalarından farklı çalışma uzayına sahip bir simülasyon yazılımı geliştirilmiştir.



Şekil 11. Çalışmada geliştirilen Stewart platform simülasyonu ve çalışma analizi yazılımı

Geliştirilen yazılım ile istenilen ölçülerde istenilen tasarım parametrelerine göre canlı bir simülasyon oluşturabilmektedir. Oluşturulmuş tasarım prototipinin çalışma uzayı analizi de gerçekleştirilerek gerçek hayata uyarlanması halinde karşılaşılabilecek sistem sınırlarının yazılımsal tespiti sağlanmıştır. Bu yazılım eniyileme algoritmaları kullanarak çalışma uzayı analizini nokta bulutu metodu ile gerçekleştirir. Bu yazılım test laboratuvarlarında gerçek ve eniyilenmiş veri çıktısı almak amacıyla geliştirilmiştir.

2.2. Yazılımsal Yöntemler ile Çeşitli Tasarım Parametrelerin Simüle Edilmesi

Bu çalışmada, stewart platform mekanizmasının GUI (Grafik Kullanıcı Arayüzü) simülasyonu ile çalışma uzayı incelenmiş ve yazılımsal en iyileme çalışmaları yapılmıştır. GUI öğelerini oluşturmak için kullanılacak diğer programlama dilleri arasında C#, Visual Basic ve Ruby gibi diller de bulunmaktadır. Genellikle, bir programlama dilinin GUI öğeleri oluşturmak için kullanılacak araçları vardır, ancak bunların özellikleri ve kullanım şekilleri dilin özelliklerine göre değişiklik gösterebilir.

İlgili simülasyon java programlama diliyle yazılmıştır. Java programlama dilini kullanarak GUI (Grafik Kullanıcı Arayüzü) simülasyonu tasarlamak, bir programı

kullanıcılar için daha kolay kullanılabilir ve anlaşılır hale getirmek için bir seçenektir. GUI, kullanıcıların bir bilgisayar programını kullanırken görsel olarak etkileşimde bulunabilecekleri bir arayüz sağlar. Örneğin, bir GUI ile bir kullanıcı bir form doldurarak bir veri girişi yapabilir veya bir tıklama ile bir işlemi başlatabilir.

Java programlama dilinde GUI simülasyonları yapmak için, Java Swing kütüphanesi kullanılabilir. Swing, Java'da GUI uygulamaları yazmak için kullanılan bir kütüphanedir ve birçok farklı GUI öğesi (örneğin, butonlar, menüler, girdi alanları vb.) sağlar.

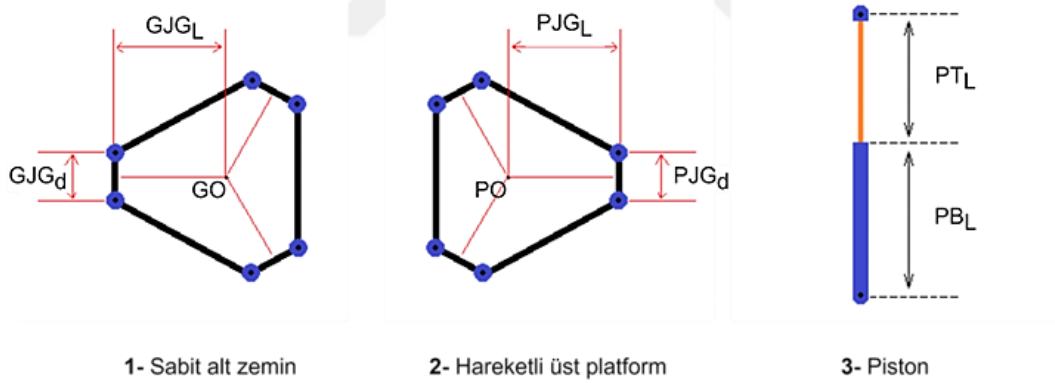
Simülasyonda 3 boyutlu işlemleri yapabilmek için Jzy3d kütüphanesi kullanılmıştır. Jzy3d, Java programlama dilinde 3 boyutlu görselleştirme için kullanılan bir açık kaynak kütüphane olup, Java Swing ve Java FX gibi GUI (Grafik Kullanıcı Arayüzü) kütüphaneleriyle birlikte çalışır. Jzy3d, 3 boyutlu grafikler oluşturmak için birçok araç ve özellik sunar, özellikle de Java programlama dilinde 3 boyutlu grafikler oluşturmak için diğer kütüphanelerden daha kolay kullanılır. Jzy3d, 3 boyutlu veri görselleştirme için kullanılabilir ve çeşitli veri türlerini (örneğin, noktalar, çizgiler, yüzeyler vb.) görselleştirmeye uygun hale getirir. Ayrıca, Jzy3d, 3 boyutlu grafikleri etkileşimli hale getirmeyi de sağlar, yani kullanıcılar grafikleri çeşitli yönlerden görüntüleyebilir veya görüntüleme seçeneklerini değiştirebilirler. Jzy3d'yi kullanmak için, öncelikle Java projesi oluşturulur ve Jzy3d kütüphanesi projeye dahil edilir. Daha sonra, Jzy3d sınıflarını kullanarak 3 boyutlu veri oluşturulur ve bu veri Jzy3d sınıflarını kullanarak görselleştirilir. Örneğin, bir 3 boyutlu nokta grafiği oluşturmak için, veri noktaları oluşturulur ve bu veri Jzy3d sınıflarını kullanarak bir nokta grafiğine dönüştürülür.

Java programlama diliyle sistem tasarlanırken thread kullanılmıştır. Java programlama dilinde Thread, bir uygulamanın birden çok parçasının aynı anda çalıştırılmasını sağlayan bir mekanizmadır. Bu sayede bir uygulamanın çeşitli işlemlerini aynı anda yapması ve daha hızlı çalışması mümkün hale gelir. Java'da Thread kavramı, java.lang.Thread sınıfını kullanarak gerçekleştirilir. Bu sınıf, Thread sınıfının bir örneğini oluşturarak kullanılır. Thread sınıfı, run() adlı bir abstract method içerir. Bu method, Thread'in ne yapacağını belirlemek için kullanılır. Örneğin, bir Thread'in bir dosya okumasını yapması için run() metodu içerisinde dosya okuma işlemleri yazılabilir.

2.2.1. Gerçekleştirilen sistemin parametre girdi ve çıktıları

Bu tez çalışmasında Java programı ile gerçekleştirilen sanal ortamda tasarlanmış sistemin değişebilen boyut ölçülerini ele alan değişken parametrelerinin yanı sıra sistemde üst platform ile sabit alt platform arasında bulunan bağlantı kollarının sistem hareketlerini ele aldığı açı parametreleri bulunmaktadır. Bu bağlantı kolları ve bağlantı kollarının hareket ettirdiği hareketli üst platformun çalıştığı alanın hesaplamasını yapan çalışma uzayı parametreleri ve piston yapısında bulunan bağlantı kollarının açıklık sınır değerlerini barındıran parametrelerden oluşmaktadır. Sistemin Platform, Zemin ve Piston Ölçülerini ele alan toplam sekiz adet parametre bulunmaktadır. Bu parametrelerden PJG_d ve PJG_L parametreleri sistemin hareketli olan üst platformunun boyutlarını şekillendirmektedir. PJG_d ve PJG_L parametrelerine girilen mm'lik değerler ile platformun üst tablasının yeni bir boyut olmasını sağlamaktadır. Zemin kısmında bulunan GJG_d ve GJG_L parametreleri ise hareketsiz olan zemin için boyutlandırmasını sağlamaktadır. Alt hareketsiz platform ile hareketli üst platform arasında bulunan pistonların (bağlantı kolları) açıları platform büyüklüklerine göre değişmektedir. Piston için kullanılan parametrelerden PT_L ve PB_L parametreleri pistonun sınırlarını şekillendirmektedir. Pistonun üst kısmını oluşturan tüp uzunluk değeri (PT_L) ve pistonun alt kısmını oluşturan gövdenin uzunluk değeri (PB_L) parametreleri için girilen değerler ile piston bağlantı kolunun hareketli tablalar arasındaki sınırını belirlemektedir. Açısal ve doğrusal parametre giriş alanında ele alınan doğrusal parametreler ile sistemin 3 boyutlu eksen takımında bulunduğu konumu vermektedir. Burada X, Y ve Z parametrelerine girilen değerler ile hareketli tablanın ve hareketli tablaya bağlı bulunan bağlantı kollarının konumlarını değiştirmektedir. Açısal parametre girişleri ise hareketli tablanın ve bağlantı kollarının yapacağı hareketleri ele almaktadır. Açısal parametre girişlerinde bağlantı kollarının koordinat ekseninde x parametresine verilen değer için X ekseninde, y parametresine verilen değer ile Y ekseninde ve z parametresine verilen değer ile Z eksenine göre platformun hareketlerini gerçekleştirmektedir. Hareketli platforma bağlı olarak sistemin çalışma analizlerinin gerçekleştirildiği parametreler çalışma uzayı analizi kısmındaki X, Y, Z parametrelerinde girilen değerler ile sistemin hareketli uç noktalarının eksen takımı üzerinde çalıştığı alan sınırlarını belirlemektedir. Bu sınırların analizi tasarımda bulunan analizi başlat butonu ile hareketli üst platformun eksen takımı üzerinde

gidebildiği koordinatlara nokta ekleyerek hareket alanını nokta bulutu ile haritalandırılmış olur. Bu nokta bulutu sayesinde sistem pistonlarının açıları, sistem piston uzunluğu, hareketli platform ve hareketsiz platform boyutları ve açıları ile çalışma uzayı analizi ele alınmıştır. Ayrıca sistemin 3 boyutlu alanda çalışması ve analiz hesaplamaları yapılırken bağlantı kollarının hareketli üst platform ve sabit alt platforma bağlantılarının uç noktaları arasında maksimum ve minimum değerlerini veren dinamik piston açıklık değerleri alanında ele alınmıştır. Bu kısımda tüm pistonların (P_0 , P_1 , P_2 , P_3 , P_4 , P_5) boyutunu oluşturan tüp (PT_L) ve gövde (PB_L) kısımlarına girilen parametreler ele alınmıştır. Açısal parametrelerin veri girişleri (X , Y , Z), doğrusal parametrelerin veri girişi (X , Y , Z), hareketli üst platform (PJG_d ve PJG_L) ve sabit alt zemin (GJG_d ve GJG_L) parametrelerinin yanı sıra piston parametrelerinde (PJ_0 , PJ_1 , PJ_2 , PJ_3 , PJ_4 , PJ_5) ele alınan parametre değerleri piston bağlantılarının çalışma uzayındaki minimum ve maksimum değerleri vermektedir. Pistonların alt hareketsiz platform ve üst hareketli platform arasında açıklıklarını yani uzayabileceği maksimum sınırı belirleyen dinamik piston açıklık değerleri parametreleri alanı kullanılmıştır.



Şekil 12. Stewart Platform mekanizmasının temel bileşenlerinin tasarım parametreleri

2.3. Ters Kinematik Hesaplama

Manipulatör sistemler genel itibari ile en az iki çalışma uzayı üzerinde bulunan rijit cisimlerin belirlenmiş koordinat merkezleri arasında gerçekleşir. Bu çalışma uzaylarının her biri için kendi öteleme ve rotasyon verilerini ayrı ayrı hesaplanır. Genellikle düz ve ters kinematik hesaplamalar yapılırken çalışma uzaylarından birisinin merkezi veya rijit bağlantı/eklem noktası global eksen takımının orijini (merkezi) olarak kabul edilir. Sistemdeki diğer rijit cisimlerin tanımlı pozisyon ve

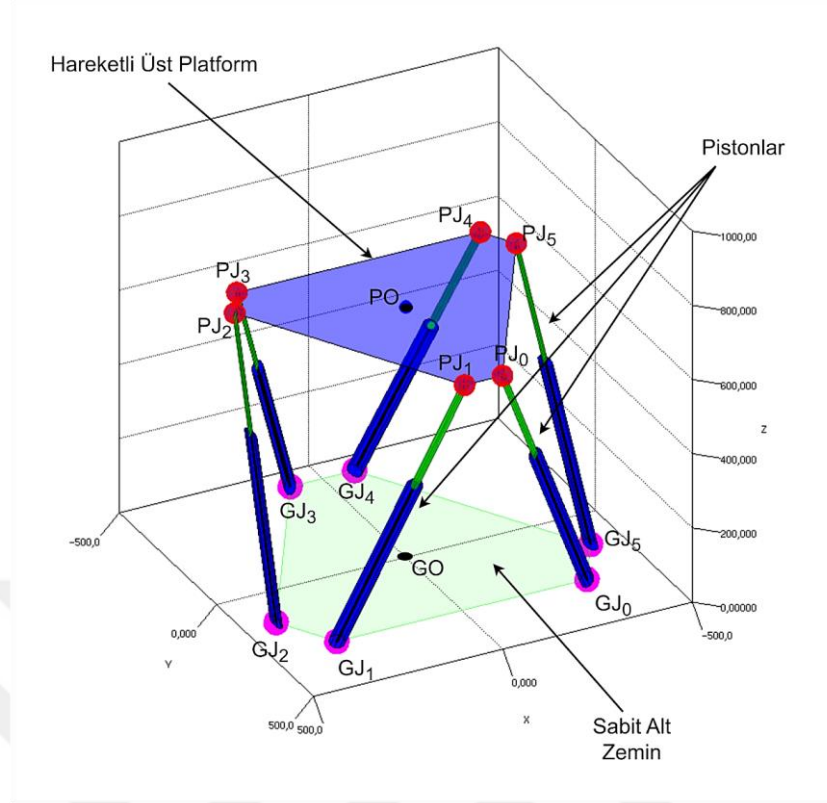
rotasyon verileri kendi eksen takımlarına göre hesaplanır. Sistemdeki rijit yapıların merkezi kabul edilen noktaların birbirlerine göre arasındaki uzuvların uzunluk, açıklık değerlerinin ileri (düz) ve ters kinematik yöntemler ile hesaplanarak sistemin sabit çalışma uzayına göre uç işlevcisinin konum ve yönelimi ayarlanır.

Ters kinematik hesaplama modelinin gerçekleştirilebilmesi uç işlevcinin eksen takımına göre pozisyon ve ötelemesi yani manipülasyon parametreleri hesaplanır. Hesaplanan manipülasyon parametreleri için uç işlevci eklem noktaları ile global eksen takımı üzerindeki bağlantı noktaları arasındaki tüm uzuvların açısal ve doğrusal pozisyonları hesaplanır. Açısal ve doğrusal parametre hesaplamaları, uzuvların tasarım modeli ile doğrudan ilişkilidir. Uzuvların tasarım modeline göre gerek eklem açısı gerekse piston uzunluğunun gibi parametrelerinin gerçekleştirilmesi sürecinde sistemin tekilliğinin korunması gerekir.

Sistem çalıştırılırken gerçekleştirilen tüm rotasyonel ve öteleme hareketleri belirli bir çözünürlüğe (açısal yada doğrusal en küçük adım, hassasiyet) göre hareketi sağlanır. Örneğin; Manipülasyon uygulanan 100 mm öteleme sınırı bulunan bir işlevci 1 mm'lik çözünürlüğe göre 100 adımda uç noktalar arasındaki hareketini tamamlayabiliyor iken çözünürlüğün 0.5mm olması durumunda 200 adımda hareketi tamamlar. Açısal hareketlerde çözünürlük radyan veya derece, doğrusal harekette ise mm veya inç gibi birimler ile ifade edilir. Sistemde tekillik analizi ile sistemin çalışma sürecinde her bir öteleme ve dönme hareket adımındaki hareketi yaparken, sistem çalışmasının çözünürlük hassasiyetine göre beklenen hareketin gerçekleştirilebilir olup olmadığının kontrolü sağlanır.

2.3.1. Model tasarım parametreleri

Stewart platform mekanizması temelde 3 parçadan oluşmaktadır. Bunlar; sabit zemin, hareketli platform ve bu ikisi arasındaki bağlantıyı sağlayan sistemin temel hareket kaynağı olan pistonlardır. Zemin merkezinden (GO) Platform merkezine (PO) doğru bir doğrusal uzanan öteleme vektörü bulunmaktadır. Ayrıca PO merkezi hareketli platformun x, y ve z eksenlerinde rotasyon merkezini ifade eder. (Şekil 13.)



Şekil 13. Stewart platform tasarımı modelinin temel elemanlarının 3d uzaydaki kurulu gösterimi

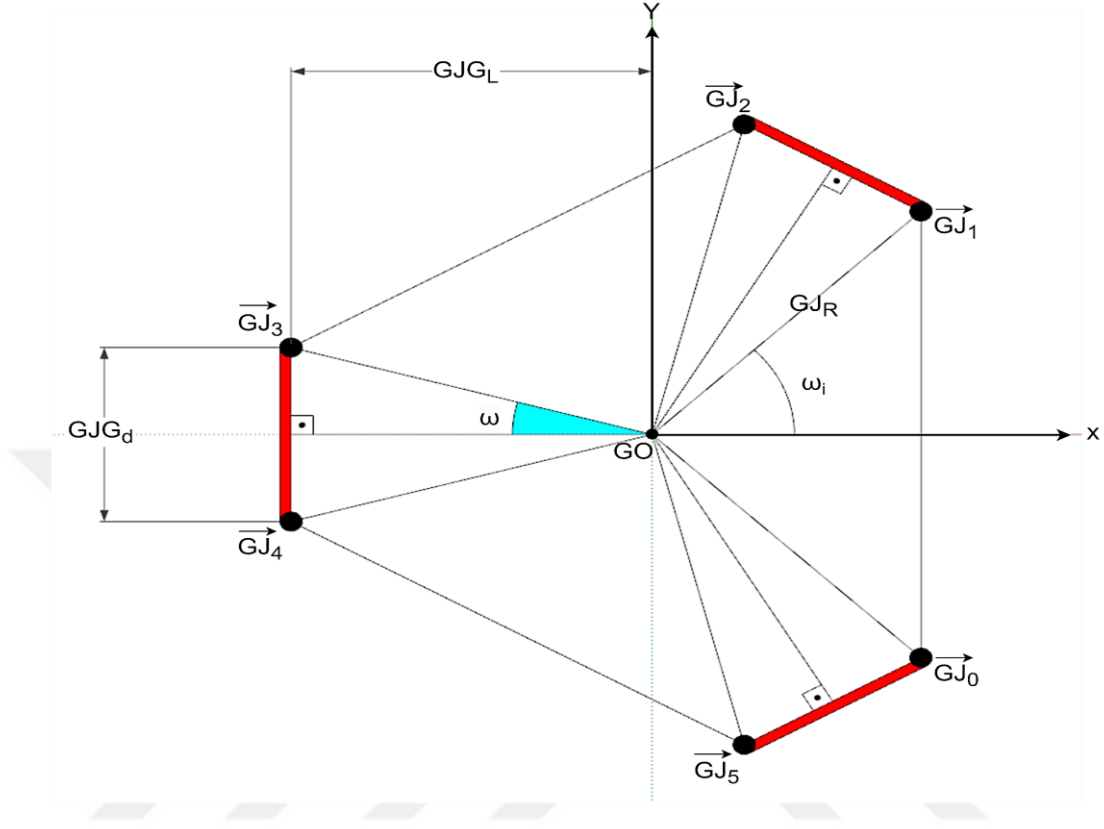
Platformun tasarım parametreleri alt başlıklarda detaylandırılmıştır.

2.3.1.1. Sabit alt zemin

Manipülator sistemler genellikle bir genel koordinat sistemi merkezine göre uç işleyicinin hareketlerinin hesaplanmasına yönelik çalışır. Bu tasarım modelinde sabit alt zemin parçası genel koordinat sistemi merkezi olarak kabul edilmiştir.

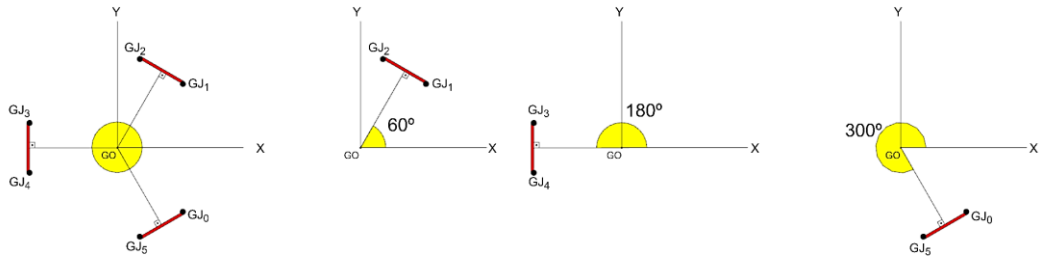
GJ_0 , GJ_1 , GJ_2 , GJ_3 , GJ_4 , GJ_5 , olmak üzere 6 eklem bağlantı noktası bulunmaktadır. Bu noktaların i . indisindeki elemanı GJ_i olarak ifade edilir. Eklem bağlantı noktalarının parçanın ortasında yer alan PO merkezine göre konumunu belirleyen temel parametreler GJG_d ve GJG_L 'dir. Bu parametreler ile z eksenindeki yüksekliği görmezden gelinerek noktaların x ve y eksenlerindeki konumları bulunur. Zemin üzerinde yer alan GJ_i bağlantı noktaları (1 - 2), (3 - 4) ve (5 - 0) eşlisi şeklinde üç adet zemin eklem takımı (ZET) oluşturulmuştur (Şekil 14.). ZET eklemlerinin

arasındaki mesafe GJG_d ile, gruptaki eklemlerin orta noktası ile zemin merkezinin (GO) arasındaki uzaklık GJG_L ile ifade edilmiştir.



Şekil 14. Sabit alt zeminin görünümü ve tasarım parametreleri

Sistemdeki zemin eklem takımlarının simetrik olarak zemine yerleştirilebilmesi için 120° 'lik açılarda konumlandırılmıştır (Şekil 15.).



Şekil 15. Zemin eklem takımlarının açısal konumları

Eklem takımlarının yerleştirme açıları ZET içerisindeki her bir elemanın bulunduğu takımın merkezinin açısal konumunu $\mu_{G(6x1)}$ matrisi içerisinde bulundurur (1).

$$\mu_{G(6x1)} = [300,60,60,180,180,30]^{-1} \quad (1)$$

ZET konumlandırmasında GJi için; Konumlarının hesaplanmasında GJi, GJ(i-1), GO üçgenininin GJi ve GJ(i-1) kenarlarına göre d(GO) köşesindeki ω açısı hesaplanırken denklem 3 'ten faydalanılır.

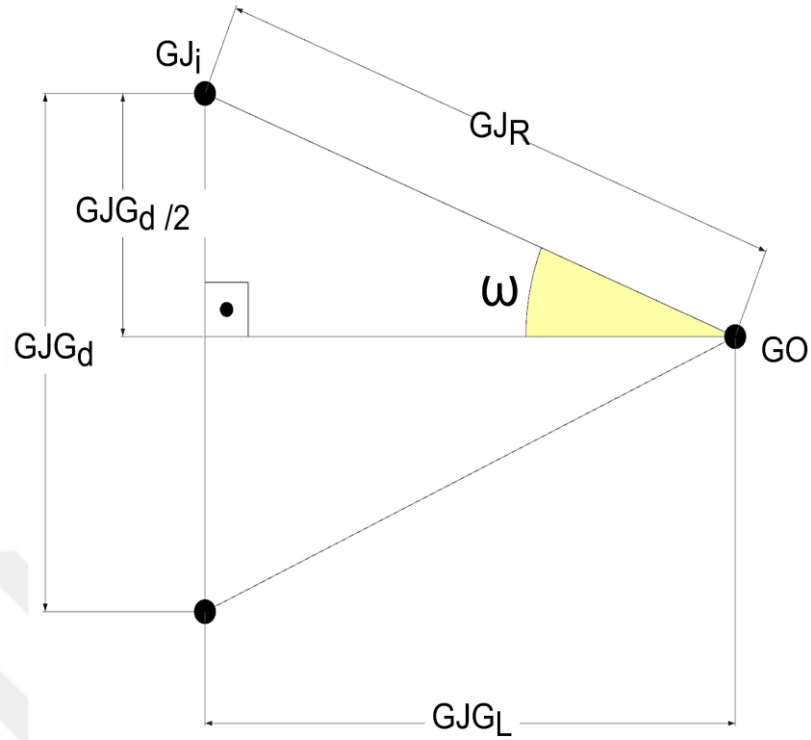
$$\omega = \arctan\left(\frac{GJG_d}{2.GJG_L}\right) \quad (2)$$

Sonraki aşamada hesaplanan bu açı ZET merkezine göre $\pm\omega$ derece kaydırılır. $\mu_{G(6x1)}$ matrisindeki parametreler kaydırma işleminin eklem bazlı değeri olan ω_i parametresini hesaplamak için kullanılır (denklem 3). Denklem içerisindeki $i\%2$ ile i parametresinin mod2'si ifade edilmiştir.

$$\omega_i = \mu_{G_i} + (\omega \cdot ((+1) - 2 \cdot (i\%2))) \quad (3)$$

Şekil 16. ile gösterilen her bir GJi eklemının GO merkezine göre dik uzaklığı (GJ_R) hipotenüs teoremi kullanılarak hesaplanır (4).

$$GJ_R = \sqrt{\left(\frac{GJG_d}{2}\right)^2 + GJG_L^2} \quad (4)$$



Şekil 16. ZET için i. eklemin GJ_i konum vektörü

GJ_i konum vektörünün x ve y parametrelerinin hesaplanması için $z=0$ kabul edilerek iki boyutta düşünülür. GJ_i konum vektörünün hesaplanması için Şekil 16. ile gösterildiği gibi hesaplanan GJ_R yarıçapının ω_i açısı kadar hesaplanma yöntemi denklem 5 ile gösterilmiştir.

$$GJ_i^{\rightarrow} = x\hat{i} + y\hat{j} + z\hat{k} \quad (5)$$

$$GJ_i^{\rightarrow} = (GJ_R \cdot \cos(\omega_i))\hat{i} + (GJ_R \cdot \sin(\omega_i))\hat{j} + (0)\hat{k} \quad (6)$$

$$GJ_i^{\rightarrow} = (GJ_R \cdot \cos(\omega_i))\hat{i} + (GJ_R \cdot \sin(\omega_i))\hat{j}$$

$$(7) GJ_i^{\rightarrow} = (GJ_R \cdot \cos(\omega_i))\hat{i} + (GJ_R \cdot \sin(\omega_i))\hat{j} \quad (7)$$

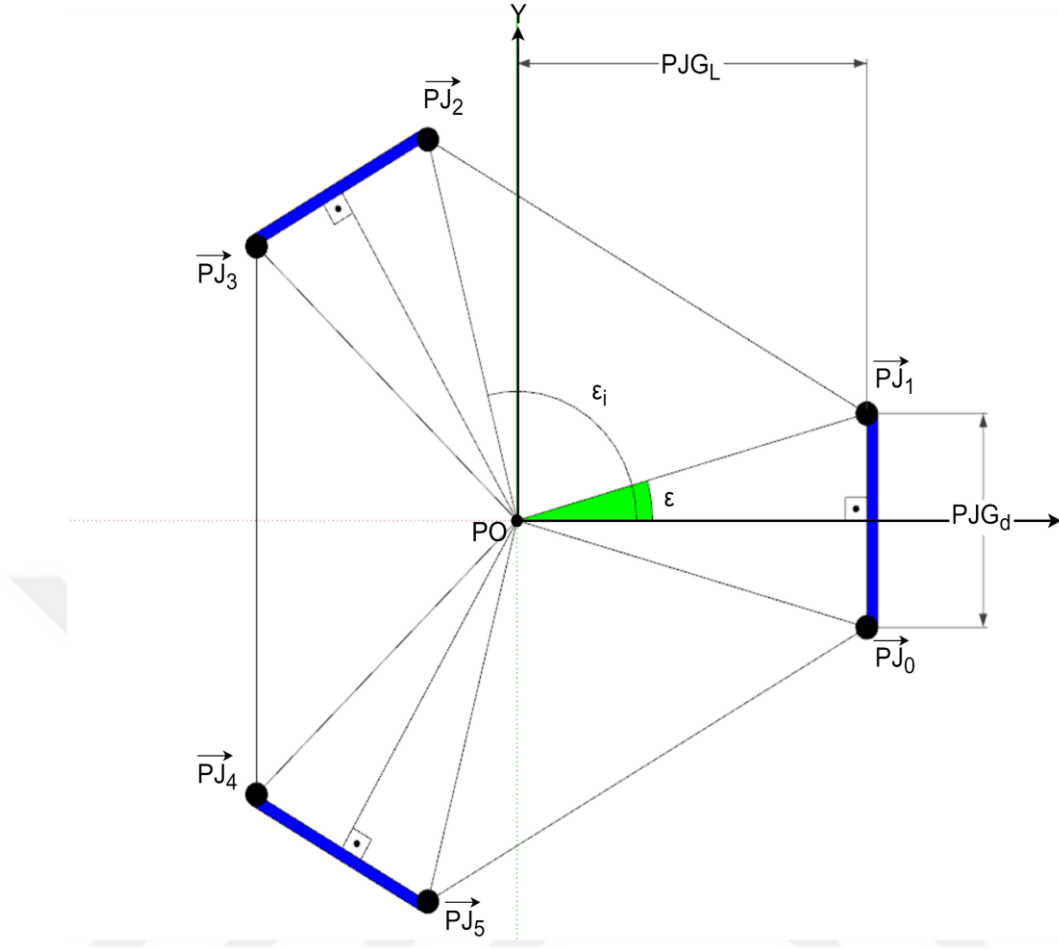
Zeminde bulunan eklemlerin konum vektörlerinin hesaplama işlemleri GJ_Gd ve GJ_GL değişken parametreleri kullanılarak hesaplanmıştır. Bu parametreler değiştirildiğinde sistemin çalışma uzayı doğrudan etkilenir.

2.3.1.2. Hareketli uç işlevci

Paralel mekanizmaların tasarım parametrelerine göre sınırlı bir çalışma uzayı bulunmaktadır. Bu çalışma uzayı en nihayetinde uç işleyicinin ulaşabileceği alanların sınırlarını ifade etmektedir.

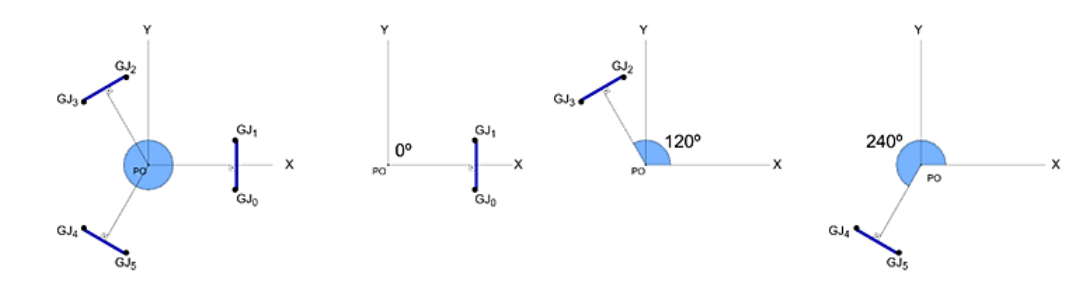
SPM'nin en temel ve olmazsa olmaz parçası olan hareketli uç işlevci (hareketli üst platform) birçok farklı şekilde tasarlanabilmektedir. Platform üzerinde bulunan eklem vektörleri ($PJ_i \vec{}$) ile zemin eklem vektörleri ($GJ_i \vec{}$) arasında hareketli kollar bulunur ve bu kolların uzunluklarına (P_L) göre çalışma sınırları belirlenir.

Hareketli üst platformun temel tasarımında PJG_L ve PJG_d parametreleri girdi olarak sunulur. Bu girdiler ile platform eklem vektörlerinin ($PJ_0 \vec{}$, $PJ_1 \vec{}$... $PJ_5 \vec{}$) uzaydaki konumları hesaplanabilir ($z=0$ için başlangıç konumları). Platformun simetrik düzenli yapısından dolayı eklemler birbirleri ile eklem takımı oluşturularak yerleştirilmiştir. Platform eklem takımı (PET) oluşturmak için hareketli platform üzerinde yer alan PJi bağlantı noktaları (0 - 1), (2 - 3) ve (4 - 5) eşlisi şeklinde üç adet platform eklem takımı (PET) oluşturulmuştur (Şekil 17.). PET eklemlerinin arasındaki mesafe PJG_d ile, gruptaki eklemlerin orta noktası ile platformun merkezinin (PO) arasındaki uzaklık PJG_L ile ifade edilmiştir.



Şekil 17. Hareketli platformun görünümü ve tasarım parametreleri

Sistemdeki platform eklem takımlarının simetrik olarak X-Y düzlemine yerleştirilebilmesi için 120° 'lik açılarda konumlandırılmıştır (Şekil 18.).



Şekil 18. Platform eklem takımlarının açısal konumları

Eklem takımlarının yerleştirme açıları PET içerisindeki her bir eklem bulduğu takımın merkezinin açısal konumunu $\mu_P(6 \times 1)$ matrisi içerisinde bulundurur.

(8).

$$\mu_{P(6 \times 1)} = [0,0,120,120,240,240]^{-1} \quad (8)$$

PET konumlandırmasında her bir PJ_i vektörünün hesaplanmasında PJ_i , $PJ(i-1)$, PO üçgeninin PJ_i ve $PJ(i-1)$ kenarlarına göre $d(PO)$ köşesindeki ε açısı hesaplanırken denklem 9'dan faydalanılır.

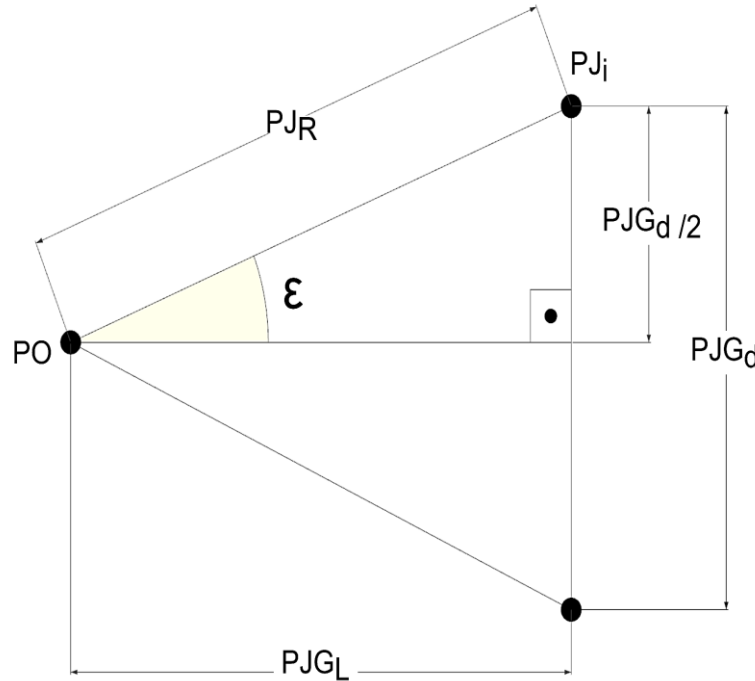
$$\varepsilon = \arctan\left(\frac{PJG_d}{2 \cdot PJG_L}\right) \quad (9)$$

Sonraki aşamada hesaplanan bu açı PET merkezine göre $\pm\varepsilon$ derece kaydırılır. $\mu_{P(6 \times 1)}$ Matrisindeki parametreler kaydırma işleminin eklem bazlı değeri olan ε_i parametresini hesaplamak için kullanılır (denklem 10). Denklem içerisindeki $i\%2$ ile i parametresinin mod 2'si ifade edilmiştir.

$$\varepsilon_i = \mu_{P_i} + (\varepsilon \cdot ((-1) + 2 \cdot (i\%2))) \quad (10)$$

Şekil 19. ile gösterilen her bir GJ_i eklemünün GO merkezine göre dik uzaklığı (GJ_R) hipotenüs teoremi kullanılarak hesaplanır (11).

$$PJ_R = \sqrt{\left(\frac{PJG_d}{2}\right)^2 + PJG_L^2} \quad (11)$$



Şekil 19. PET için i . eklem için PJ_i konum vektörü

PJ_i konum vektörünün x ve y parametrelerinin hesaplanması için z eksenindeki konumu görmezden gelinir ($z=0$). Platformun ilk yüksekliği PJ_i konum vektörü hesaplandıktan sonra verilir. PJ_i 'nin hesaplanması için Şekil 17. ile gösterildiği gibi hesaplanan PJ_R yarıçapının ϵ_i açısı kadar hesaplanma yöntemi denklem 12 ile gösterilmiştir.

$$PJ_i^{\rightarrow} = x\hat{i} + y\hat{j} + z \quad (12)$$

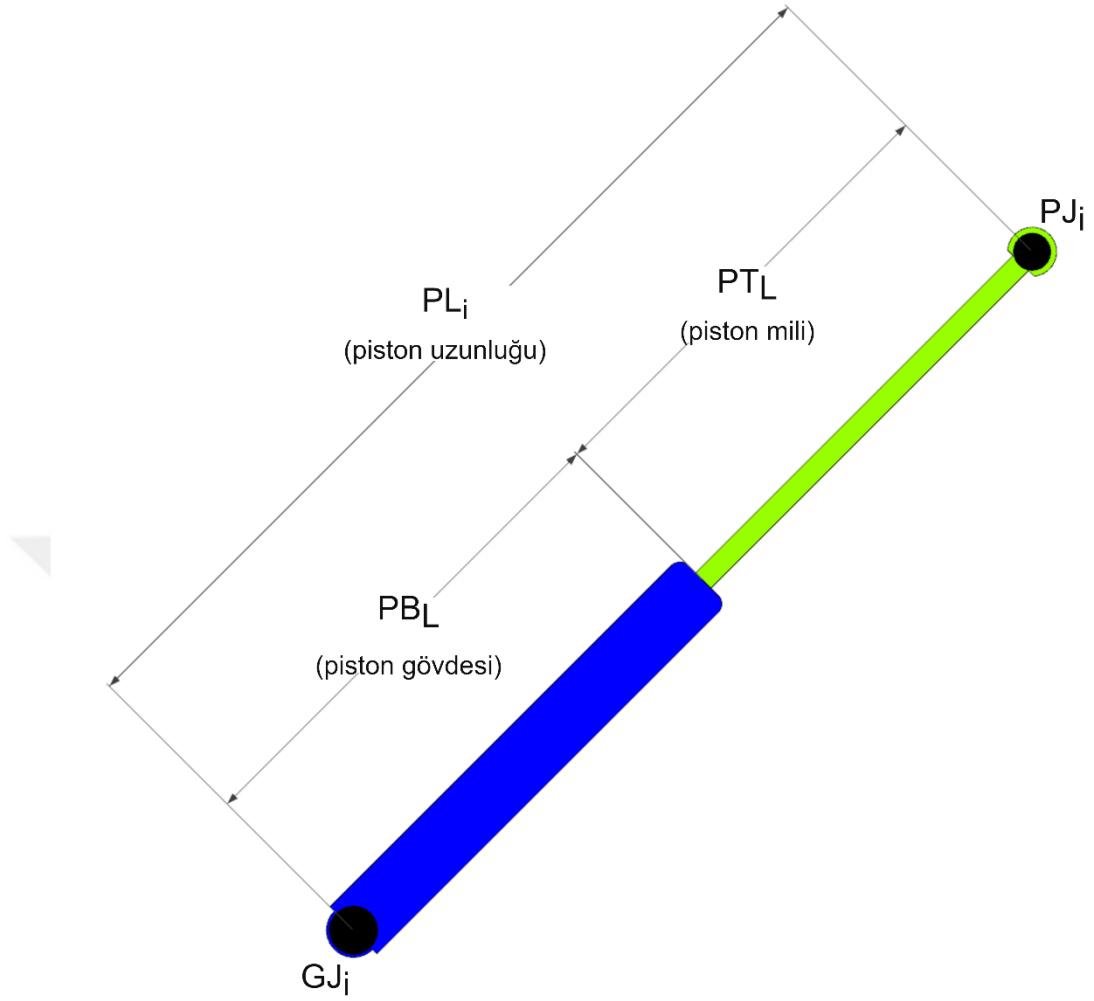
$$PJ_i^{\rightarrow} = (PJ_R \cdot \cos(\epsilon_i))\hat{i} + (PJ_R \cdot \sin(\epsilon_i))\hat{j} + (0)\hat{k} \quad (13)$$

$$PJ_i^{\rightarrow} = (PJ_R \cdot \cos(\epsilon_i))\hat{i} + (PJ_R \cdot \sin(\epsilon_i))\hat{j} \quad (14)$$

Platformda bulunan eklemlerin konum vektörlerinin hesaplama işlemleri PJG_d ve PJG_L değişken parametreleri kullanılarak hesaplanmıştır. Bu parametreler değiştirildiğinde sistemin çalışma uzayı doğrudan etkilenir.

2.3.1.3. Bağlantı uzuvları (piston)

Robotik mekanizmalarının hareket edebilmeleri için en temelde eyleyicilere ihtiyaç duyulur. Bu eyleyiciler genellikle elektrik motorlarından alınan dairesel hareketi, dairesel veya doğrusal olarak bağlı bulunduğu bağlantı elemanına aktarır. Böylelikle doğrusal veya açısal hareket elde edilir. SPM'de bağlantı uzuvları (Piston) sabit alt zemin ve hareketli üst platform arasında yer alır (Şekil 20.). Platformun PJ_i konum vektörleri ile zeminin GJ_i konum vektörleri arasındaki birleşimi P_i pistonu sağlar.



Şekil 20. Bağlantı uzvunun görünümü ve tasarım parametreleri

Pistonun çalışma sınırlarını piston gövdesi (PB_L) ve bu gövde içerisinde yer alan milin uzunlukları (PT_L) belirler. PL_i değerinin en kısa ve en uzun değerleri PL_{min} ve PL_{max} aralığındadır. PL_{min} hesaplanırken pistonun gövdesi veya pistonun mili uzunluklarından hangisi daha uzun ise pistonun en kısa hali o uzunlukta olacaktır (15). PL_{max} hesaplanırken ise pistonun mili ve pistonun gövdesinin toplam uzunlukları PL_{max} sınır değerini verecektir (16).Denklem 15 içerisinde tanımlı bulunan $\max(x,y)$ fonksiyonu içerisine gönderilen parametrelerden hangisi büyükse sonuç olarak onu verir.

$$PL_{min} = \max(PB_L, PT_L) \quad (15)$$

$$PL_{max} = PB_L + PT_L \quad (16)$$

Ters kinematik yöntemler ile yapılan hesaplamalara göre platforma açısız ve rotasyonel hareket uygulanmak istenildiğinde her bir P_i pistonunun PL_i uzunluğu PL_{min} ve PL_{max} arasında olmalıdır (17).

$$PL_{min} < PL_i < PL_{max} \quad (17)$$

2.3.2. Uç işlevci hareketleri ve kinematik denklemler

Platforma rotasyonel ve doğrusal hareket uygulanması durumunda sistemin tüm eyleyicileri (motor, piston vb.) yeniden pozisyonlandırılmaktadır. Bu pozisyonlar düz veya ters kinematik hesaplamalara göre yapılır. Düz kinematik analiz ters kinematik analize göre daha zor ve denklem çözümü daha karmaşıktır.

Platformun manipülasyon parametreleri altı adettir. Bunlar x, y, z eksenlerindeki iteratif olarak pitch (yatma hareketi), roll (yunuslama hareketi) ve yaw (sapma hareketi) şeklindeki dönme hareketleri ve surge, heave ve sway ile ifade edilen öteleme hareketleridir (Tablo 3.). Platforma uygulanan öteleme hareketleri $P\vec{M}$ vektörü, dönme hareketleri ise \vec{R} vektörü ile ifade edilir ($\vec{R}_{(3x1)} = [\alpha, \beta, \delta]^{-1}$).

Tablo 3. X, Y ve Z eksenlerindeki manipülasyon parametreleri

	X		Y		Z	
	Adı	Parametre	Adı	Parametre	Adı	Parametre
Açısal	Pitch	$R_x = \alpha$	Roll	$R_y = \beta$	Yaw	$R_z = \delta$
Doğrusal	Surge	PM_x	Heave	PM_y	Sway	PM_z

2.3.2.1. Ters kinematik yöntemlerle manipülasyon (öteleme, dönme) hareketleri

Ters kinematik yöntemlere göre manipülasyon doğrusal ve açısız olarak iki farklı şekilde ele alınabilir. Sisteme uygulanan manipülasyon sistemin hareket

çözünürlüğüne göre çalışma uzayı analizi yöntemleri ile kontrol edilir. Sistemin çözünürlüğe göre hareketinin gerçekleştirilebilir olup olmadığı kontrol edilir ve bağlı bulunan uzuvların tekillik bayrakları kapalı ise buna göre istenilen açısal ve doğrusal konuma doğru hareket tamamlanır. Aksi takdirde istenilen hareket için süreç tamamlanmaz. Bu durum çalışma uzayı sınırları içerisinde olmadığını doğrular.

Doğrusal manipülasyonda hareketli platform üzerinde yer alan hiçbir eklemdaki açı bozulmadan platform üzerindeki tüm $P\vec{J}_i$ pozisyon vektörlerine eşit bir şekilde $P\vec{M}$ pozisyon vektörü eklenerek artış veya azalış uygulanır. Bu işlemin sonunda güncel pozisyon $P\vec{J}'_i$ elde edilir. (Denklem 18).

$$P\vec{J}'_i = P\vec{J}_i + P\vec{M} \quad (18)$$

$$P\vec{J}_i = P\vec{J}_i x\hat{i} + P\vec{J}_i y\hat{j} + P\vec{J}_i z\hat{k} + P\vec{M}_x\hat{i} + P\vec{M}_y\hat{j} + P\vec{M}_z\hat{k} \quad (19)$$

$$P\vec{J}'_i = x\hat{i}' + y\hat{j}' + z\hat{k}' \quad (20)$$

Öteleme hareketi sonucunda platformun güncel pozisyon vektörü artık $P\vec{J}'_i$ olacaktır. Dönme hareketinin uygulanması ise geometrik, trigonometrik ve matris çarpımı gibi yöntemler ile yapılabilir.

Trigonometrik yöntem ile matris çarpım yöntemi arasındaki temel fark, çevirme işlemini gerçekleştirmede kullanılan matematiksel yapıdır. Trigonometrik yöntemde, çevirme işlemini gerçekleştirmek için trigonometrik işlemler kullanılırken, matris çarpım yönteminde ise matrisler kullanılır.

Trigonometrik yöntem, çevirme işlemini anlamaya çalışan bir kişi için daha anlaşılır olabilir. Bu yöntemde, çevirme işlemini gerçekleştirmek için trigonometrik işlemler kullanılır, bu yüzden çevirme işlemini anlamaya çalışan bir kişi için daha anlaşılır olabilir. Ancak, trigonometrik yöntem, geometrik yöntemle göre daha yavaş olabilir ve çok boyutlu uzaylarda çevirme işlemlerini gerçekleştirmede daha zor olabilir.

Matris çarpım yöntemi ise, çevirme işlemini hızlı ve kolay bir şekilde gerçekleştirmede avantaj sağlar. Örneğin, çevirme işlemini bir dönüş matrisi ile gerçekleştirmek daha kolay ve hızlı olabilir, çünkü matrisler arasında matris çarpımı

işlemi kullanılır. Bu yöntem ayrıca, çok boyutlu uzaylarda çevirme işlemlerini gerçekleştirmede de daha kolaydır.

Sonuç olarak, trigonometrik yöntem çevirme işlemini anlamaya çalışan bir kişi için daha anlaşılır olabilir, ancak matris çarpım yöntemi çevirme işlemini hızlı ve kolay bir şekilde gerçekleştirmede avantaj sağlar. Hangi yöntem kullanılacağı, özellikle çevirme işlemini gerçekleştirirken kullanılmak istenen özelliklerine ve amaçlara göre değişebilir.

2.3.2.1.1. Trigonometrik yöntemlerle uç işlevci hareketleri

PJ_i vektörünün platform RO konum vektörü merkez kabul edilerek trigonometrik yöntemlere göre çevrilmesi için platformun tüm eklemlerinin x, y ve z eksenlerindeki konumu (PJ_i) bilinmektedir. Dönme merkezi ile her bir PJ_i vektörü arasındaki öteleme farkı alınarak dönüş merkezi orjin kabul edilir. Daha sonra bu noktayı x ekseninde α derece, y ekseninde β derece ve z ekseninde δ derece çevirmek için denklem 21 ile trigonometrik yöntemler ile rotasyon uygulanmaya başlanır.

$$PJ_i'' = PJ_i' - R\vec{O} \quad (21)$$

$$x = PJ_i''\hat{i}, \quad y = PJ_i''\hat{j}, \quad z = PJ_i''\hat{k} \quad (22)$$

Hesaplanan dönme merkezine göre ilk olarak, x ekseninde α derece çevirilir:

$$x' = x \quad (23)$$

$$y' = y\cos(\alpha) - z\sin(\alpha) \quad (24)$$

$$z' = y\sin(\alpha) + z\cos(\alpha) \quad (25)$$

y ekseninde β derece çevirilir :

$$x'' = x'\cos(\beta) - z'\sin(\beta) \quad (26)$$

$$y'' = y' \quad (27)$$

$$z'' = x'\sin(\beta) + z'\cos(\beta) \quad (28)$$

z ekseninde δ derece çevirilir:

$$x''' = x''\cos(\delta) - y''\sin(\delta) \quad (29)$$

$$y''' = x''\sin(\delta) + y''\cos(\delta) \quad (30)$$

$$z''' = z'' \quad (31)$$

Son olarak, rotasyon merkezi çevirme işlemleri sonrası eski konuma getirilir:

$$P\vec{J}_i''' = x'''i + y'''j + z'''k + R\vec{O} \quad (32)$$

Böylece istenen açı dönme merkezine göre uygulanarak platformun yeni konumu hesaplanmış olur. Bu adımlar takip edilerek, x, y ve z eksenlerinde istenilen açılarda PJ_i vektörü çevirilmiş olur ($P\vec{J}_i'''$). Burada dikkat edilmesi gereken nokta, çevirme işlemi yaparken rotasyon işlemlerinin sırayla uygulaması gerektiğidir. Örneğin, x ekseninde α derece çevirme işlemi yaptıktan sonra y ekseninde β derece çevirme işlemi yapmalı, y ekseninde β derece çevirme işlemi yapıldıktan sonra z ekseninde δ derece çevirme işlemi yapmalıdır.

2.3.2.1.2. Matris çarpım yöntemi ile uç işlevci hareketleri

Matris çarpım yöntemi ile üç boyutlu bir uzayda platformun her PJ_i bir noktasının x, y ve z eksenlerinde α , β ve δ derece kadar çevirmek için, koordinatlarını matris çarpımı ile bir rotasyon matrisi ile çarpılır. Ancak dönüşüm uygulanmadan önce sırası ile denklem 21 ve denklem 22 denklemlerinde belirtilen işlemler gerçekleştirilir. Rotasyon sağlandıktan sonra denklem 32 ile belirtilen denklem ile $P\vec{J}_i'''$ güncel konum vektörleri bulunur.

Rotasyon matrisleri;

X ekseninde α derece çevirmek için kullanılacak matris:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (33)$$

Y ekseninde β derece çevirmek için kullanılacak matris:

$$\begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \quad (33)$$

Z ekseninde δ derece çevirmek için kullanılacak matris:

$$\begin{bmatrix} \cos(\delta) & -\sin(\delta) & 0 \\ \sin(\delta) & \cos(\delta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (34)$$

Bu matrisleri kullanarak, noktanın koordinatlarını PJ_i vektörü ile çarparak, noktanın çevrilmiş koordinatları bulunur. Örneğin, x ekseninde α derece çevirilmiş koordinatları (x' , y' , z') için denklem 35 denklemi kullanılır:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} PJ_i \hat{i} \\ PJ_i \hat{j} \\ PJ_i \hat{k} \end{bmatrix} \quad (35)$$

Benzer şekilde, y ekseninde β derece çevrilmiş koordinatları (x' , y' , z') için aşağıdaki denklem 36 kullanılabilir:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (36)$$

z ekseninde δ derece çevrilmiş koordinatları (x' , y' , z') için ;

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos(\delta) & -\sin(\delta) & 0 \\ \sin(\delta) & \cos(\delta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (37)$$

denklemleri kullanılabilir.

Bu matrisler kullanarak, x, y ve z eksenlerinde istenilen derecelerde çevrilebilen bir noktaya rotasyon uygulanabilir. Dikkat edilmesi gereken bir nokta, çevirme işlemini yaparken rotasyon matrislerini sırayla uygulamak gerektiğidir. Örneğin, x ekseninde α derece, y ekseninde β derece ve z ekseninde δ derece çevirmek için, ilk olarak x ekseninde α° çevirme matrisi, bir sonraki adımda y ekseninde β° çevirme matrisi ve z ekseninde δ° derece çevirme matrisi kullanılarak noktanın koordinatları çarpılır (Denklem 38).

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \begin{bmatrix} \cos(\delta) - \sin(\delta) & 0 \\ \sin(\delta) & \cos(\delta) \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 0 & 1 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (38)$$

Burada x'' , y'' ve z'' çevrilmiş koordinatlardır.

Önceki örnekte verilen rotasyon sırası ile önce x, sonra y ve sonra z eksenlerinde çevirmeyi sağlar. Bu işlemleri yaparken, rotasyon matrislerini sırayla uygulamak yerine, tüm matrisleri birleştirerek bir rotasyon matrisi oluşturarak da çevirme işlemi tek seferde uygulanabilir. Bu durumda, dönüş matrisi denklem 39 ile gösterildiği gibi olacaktır:

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \begin{bmatrix} \cos(\delta) \cos(\beta) \cos(\alpha) - \sin(\alpha) \cos(\delta) \cos(\beta) - \sin(\beta) \cos(\alpha) \\ \sin(\delta) \cos(\beta) \cos(\alpha) - \sin(\alpha) \sin(\delta) \cos(\beta) + \sin(\beta) \sin(\alpha) \\ \sin(\beta) \cos(\alpha) & - & \sin(\alpha) \sin(\beta) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (39)$$

Bu yöntem, rotasyon matrislerinin sırayla uygulanması yerine tek bir matris ile çevirme işlemini gerçekleştirir, ancak matrisin hesaplanması daha zordur.

2.4. Sistemde Tekillik - Tekillik Sınır Bayrakları

Paralel mekanizmaların tasarım, analiz, hareket planlama ve kontrolünde en önemli problemlerden biri tekilliktir. Paralel mekanizmaların kapalı yapıları nedeniyle uzay alanlarında yaptıkları hareketler kısıtlıdır ve çalışma alanından yetersizliğinden kaynaklı olarak sistem hareketlerinde yer değiştirme olmaz (Tekillilik). Bu durum manipülâtörün kontrol edilemediği konumlar olarak ifade edilebilir. Stewart platform mekanizmalarının çalışma alanındaki yapılan tekillik araştırmalarda, bağlantı

kollarının uzunlukları ile eklemlerin dönüş gerçekleştirdiği açıların sınırları kısıtlama şeklinde kullanılmıştır. Rıza gören Yunuslama (Porpoising), rotasyon ve sallanma açıları ve bağlantı kollarında herhangi bir temas olup olmadığı incelenmiştir. İlk olarak Hunt tarafından yapılan çalışmalar, sistemdeki 6 bağlantı kolunun eşzamanlı olarak doğru çevresinde dönerek hareket etmesi halinde tekillilik olduğu gösterilmiştir. Fitcher ise, sabit alt platformun ve üst hareketli platformun sisteme paralel olarak yerleştirilmesi durumunda, hareketli olan üst kısımdaki platformun +90 veya -90 derece z ekseninde döndürülmesi sistemde yine bir tekil durum oluşumuna neden olacağını belirtmişlerdir (Mruthyunjaya ve Dasgupta, 1998; Sait ve Yurt, 2002). Gosselin ve Angeles ise, 1990 yılında Stewart platform mekanizmalarında ekseriya 3 tekil durumun oluşma ihtimali olabileceğine değinmişlerdir (Gosselin ve Angeles, 1990). Son yıllarda, hareketlerin en iyi değerlendirileceği çalışma uzay alanı ve en elverişli yörüngeyi belirlemek için yapılan çalışmalar önem elde etmiştir (Collins ve Long, 1995; Kim, Chung ve Youm, 1999).

Bu tez çalışmasında Stewart platformunun tasarımında, tekillik için bağlantı kolları tasarlanmıştır. Bu bağlantı elemanları, platformun hareketine ilişkin olarak, platformun eksenlerine göre hareketini sınırlar ve platformun tekilliğini sağlar.

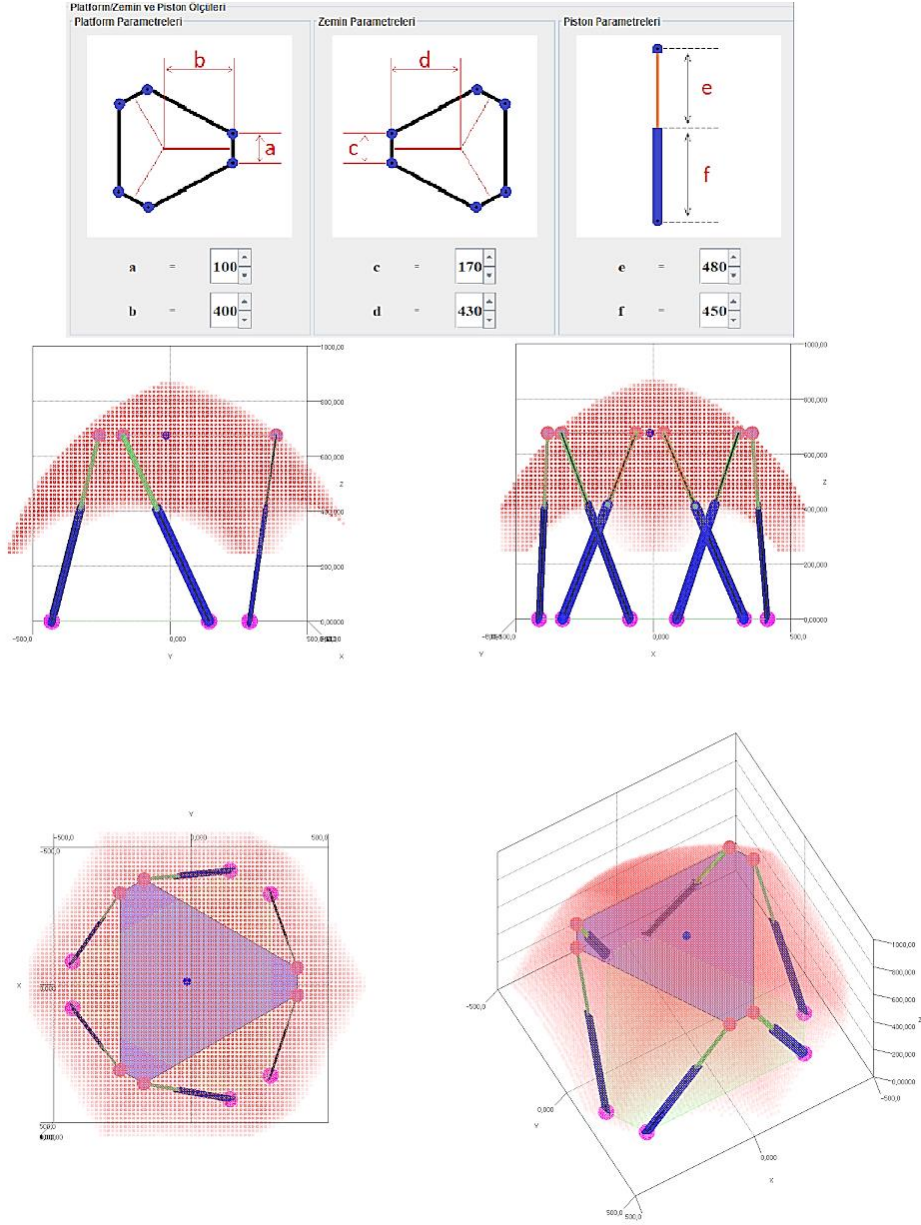
2.5. Çalışma Uzayı Analizi

Paralel mekanizmaların çalışma uzaylarının belirlenmesinde genel bir yaklaşım olmamasından dolayı daha çok endüstri ve mühendislik uygulamalarında kullanılır. Bu, paralel mekanizmanın konfigürasyonunun ve kinematik analizinin karmaşıklığından kaynaklanır. Paralel mekanizmalarda hareket (çalışma) uzayını belirten kriterler olarak piston bağlantı yerleri ve hareketli platform için kullanılan üst tabla ve hareketsiz alt tabla arasındaki bağlantıyı ve koordinat düzlemde hareketleri gerçekleştiren pistonların uzunlukları sistemdeki ağırlık merkezinin çalıştığı alan olarak belirlenmektedir. Stewart Platform Mekanizmasının hareket alanları olan çalışma uzay alanı, bacak uzunluk sınırları ve hareket sınırlarına bağlı olarak mekanik ve geometrik özelliklerine göre belirlenir. Son zamanlarda yapılan paralel mekanizmaların çalışma uzayı analizi, dönme ve öteleme çalışma uzayı analizleri olarak iki ana gruba ayrılmıştır. Öteleme hareketine bağlı bir analiz çalışmasında öteleme eylemi gerçekleştirilirken ve dönme eylemi gerçekleştirilmezken, hareket eden platformun, koordinat sisteminde bulunan x , y ve z 'deki konumlarında

ötenmesi sonucu belirlenir ve sistemin çalışma uzayı yapılırken konumsal olarak yapılmaktadır. Dönme hareketi gerçekleştirilirken yapılan çalışma uzayındaki analizi ise, platform ağırlık merkezinin değişmediği durumlar ele alınır ve sistemin koordinat sistemindeki x, y ve z parametrelerine sahip eksenlerde dönmesiyle tamamlanır. Transformasyon matrisi, hareketli platformun doğrusal hızı ile altı giriş ekleminin doğrusal ve açısal hızları arasındaki ilişkisini gösterir ve Stewart Platform Mekanizmalarının tasarımı, kullanılan özellikle platform ve bağlantı kollarının boyutu ve çalışma uzayını etkiler. Jacobian matrisi ise 6 serbestlik derecesine sahip sistemlerde 6x6'lık matristir. Çalışma uzayının belirlenirken farklı bir zorluk olarak, 6 serbestlik derecesi olan bir sistem, 6 boyutlu olarak ele alınması durumunda şu an için oluşturulamayacak bir analiz ile karşı karşıya getirecektir. Bu nedenle, çalışma uzayının analizi yapılırken 6 boyutlu bir çalışma uzayındaki analizinde 6 boyutlunun alt kümeleri olan 3 boyutlu gibi çalışma uzayının alt kümelerinden biri kullanılarak gösterilebilir. Bu nedenle, çalışma uzayının belirlenmesi için yapılan çalışmalar, genellikle daha düşük boyutlu çalışma uzaylarını inceler ve bu çalışmalar sonucu elde edilen veriler ile 6 boyutlu çalışma uzayının tahmini yapılır. Çalışma uzayının belirlenmesinde kullanılan yöntemler arasında, kinematik denklemleri kullanan yöntemler ve geometrik yöntemler yer alır. Kinematik denklemleri kullanan yöntemler, düz kinematik denklemleri kullanır ve bu denklemler, çalışma uzayının hesaplanmasında kullanılır. Ancak, bu denklemler gerçekten karışık denklemlerdir ve bu nedenle çalışma uzayının belirlenmesinde genel bir yaklaşım olmamasına neden olur. Geometrik yöntemler ise, çalışma uzayının belirlenmesinde kullanılan alternatif yöntemlerdir ve bu yöntemler, çalışma uzayının geometrik yapısını inceler ve bu yapıyı kullanarak çalışma uzayını belirler. Bu yöntemler, çalışma uzayının belirlenmesinde daha pratik ve anlaşılır olabilirler, ancak bu yöntemlerin doğruluğu ve hassasiyeti kinematik denklemleri kullanan yöntemlere göre daha düşüktür (Ünsal, 2007; Elmas, 2005).

Stewart platform tabanlı paralel manipülatörler, yapıldığında kinetik ve dinamik özellikleri sabit kalır. Bu nedenle, çeşitli uygulamaları karşılamak için çeşitli yapıları olan paralel manipülatörler gerekmektedir. Lee ve arkadaşları araştırmalarında, platform merkezine platform-bacak temas noktasına bağlantı sağlayan kolun uzunluğunu aktüatör tarafından değiştirilebilen değişken kol tipi olacak şekilde paralel manipülatör geliştirilmiştir. Bu manipülatörün çalışma alanı, geleneksel Stewart

platformundan daha büyüktür ve özellikle maksimum yönlendirme açısı önemli ölçüde genişletilmiştir. Ayrıca, bacaklar ve platform arasındaki kuvvet ve moment aktarımı özellikleri, çeşitli görevlerin gereksinimlerine göre ayarlanabilir. Bu değişken kol tipi paralel manipülatörün kullanılabilirliğini doğrulamak için kinematik ve dinamik analizler gerçekleştirilmiş ve gerçek donanım, bu manipülatörün uygulanabilirliğini göstermek için yapılmıştır (Lee, Song, Choi ve Hong, 2002).



Şekil 21. Yapılan sistemin çalışma uzayı çıktıları

ÜÇÜNCÜ BÖLÜM

MEKANİZMA TASARIM PARAMETRELERİNİN ÇALIŞMA UZAYINA ETKİLERİ

Mekanizma tasarımı, bir sistemin belirli bir görevi yerine getirebilmesi için gerekli olan hareketleri gerçekleştirme için gereken parçaların ve bu parçaların birbirleriyle nasıl etkileşime girdiğinin belirlenmesi olarak tanımlanabilir. Mekanizma tasarımı sırasında, tasarımcılar bir dizi parametre dikkate alırlar ve bu parametrelerin çalışma uzayına nasıl etki ettiğini değerlendirirler. Bu bölümde, mekanizma tasarımı parametrelerinin çalışma uzayına etkileri üzerinde durulmuştur.

Mekanizma tasarımı parametreleri, tasarım sürecinin her aşamasında dikkate alınması gereken önemli kavramlardır. Bunlar, mekanizmanın çalışma uzayına nasıl etki ettiğine dair bilgi verir ve tasarımın nasıl gerçekleştirileceğine dair önemli bilgiler sağlar. Aşağıda, mekanizma tasarımı parametrelerinin çalışma uzayına nasıl etki ettiğinin üzerinde durulması gereken bazı önemli parametreler verilmiştir:

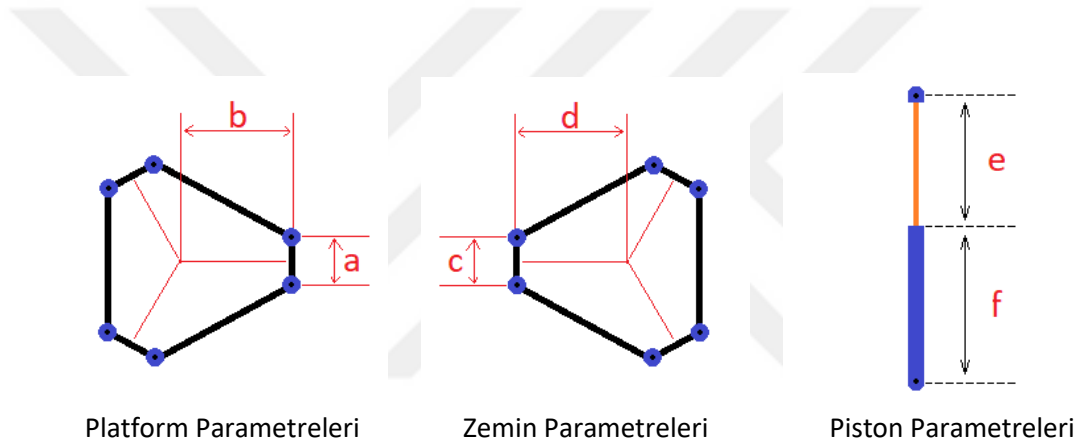
- Hareket eksenleri sayısı: Mekanizma tasarımı sırasında, hareket eksenleri sayısı belirlenir. Bu parametre, mekanizmanın çalışma uzayında nasıl hareket edeceğine dair önemli bilgiler sağlar. Örneğin, bir tekerlekli robotun hareket eksenleri sayısı iki olabilir, ancak bir manipülatörün hareket eksenleri sayısı daha fazla olabilir. Hareket eksenleri sayısı, mekanizmanın çalışma uzayına nasıl etki ettiğini belirler.

- Hareket aralığı: Mekanizma tasarımı sırasında, hareket aralığı belirlenir. Bu parametre, mekanizmanın çalışma uzayında nasıl hareket edeceğine dair önemli bilgiler sağlar. Örneğin, bir tekerlekli robotun hareket aralığı belirli bir mesafeyi geçemeyebilir, ancak bir manipülatörün hareket aralığı daha fazla olabilir. Hareket aralığı, mekanizmanın çalışma uzayında ne kadar alanı kapsayabileceğine dair bilgi verir ve bu da mekanizmanın görevlerine uygun olup olmadığını belirlemede önemlidir.

- Hareket hızı: Mekanizma tasarımı sırasında, hareket hızı da belirlenir. Bu parametre, mekanizmanın çalışma uzayında nasıl hareket edeceğine dair önemli bilgiler sağlar. Örneğin, bir tekerlekli robotun hareket hızı belirli bir seviyenin üzerine çıkamayabilir, ancak bir manipülatörün hareket hızı daha yüksek olabilir. Hareket hızı,

mekanizmanın çalışma uzayında ne kadar hızlı hareket edebileceğine dair bilgi verir ve bu da mekanizmanın görevlerine uygun olup olmadığını belirlemede önemlidir.

Bu parametreler, mekanizma tasarımı sırasında dikkate alınması gereken önemli parametrelerdir; mekanizmanın çalışma uzayında nasıl hareket edeceğine, ne kadar ağırlıkta yükleri taşıyabileceğine ve ne kadar enerji tasarrufu sağlayabileceğine dair bilgi verirler. Bu parametrelerin doğru bir şekilde belirlenmesi, mekanizmanın çalışma uzayında başarılı bir şekilde çalışmasını sağlar. Bu çalışmada Şekil 22. ile gösterilen a, b, c, d, e, f uzunlukları farklı kombinasyonlarda ayarlanarak çalışma uzayının erişilebilir alanlarının nokta bulutu metodu ile tespiti ve erişilebilir bu alandaki değişikliğin sonuca etki oranının tespit edilmesi üzerinde durulmuştur.



Şekil 22. Özgün SPM'nin tasarımını ve çalışma uzayını etkileyen temel tasarım parametreleri

Çalışmadaki parametre değişikliklerinin daha anlaşılır olabilmesi için Ana Referans Platform Modeli (ARPM) tanımlanmıştır. Daha sonrasında Şekil 22. ile gösterilen parametrelerin değişikliklerine göre Deneysel Platform Modelleri (DPM) için yeni analizler başlatılarak sonuçlar ARPM ile karşılaştırılmıştır. DPM tasarım parametreleri Tablo 4. ile verilen uzunluklara göre belirlenmiştir.

Tablo 4. Deneysel Platform Modeli tasarım parametreleri

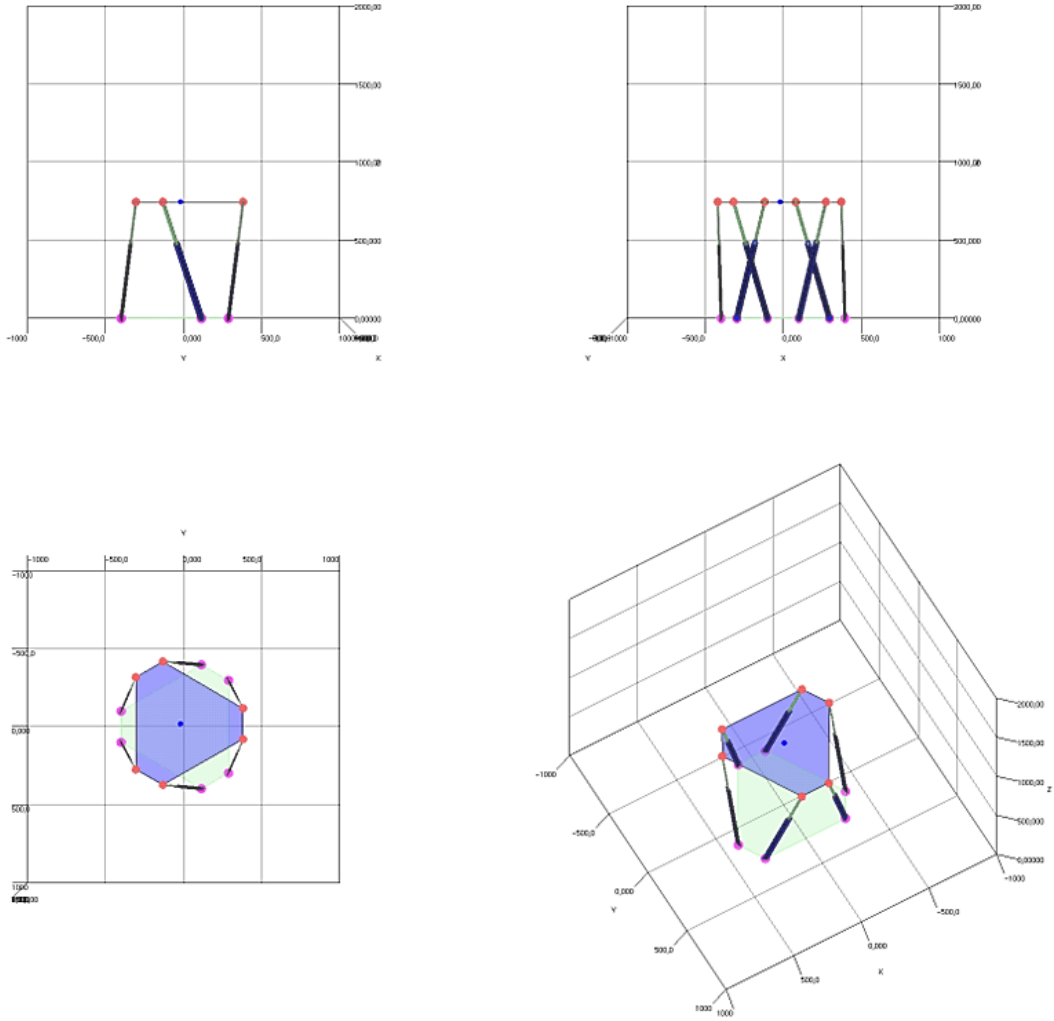
Güncellenen Tasarım Parametreleri	Konfigürasyon Adı	a	b	c	d	e	f
	Ana Referans Platform Modeli (ARPM)	200	400	200	400	500	500
PLATFORM	DeneySEL Platform Modeli 1 (DPM1)	100	400	200	400	500	500
	DeneySEL Platform Modeli 2 (DPM2)	400	400	200	400	500	500
	DeneySEL Platform Modeli 3 (DPM3)	200	200	200	400	500	500
	DeneySEL Platform Modeli 4 (DPM4)	200	800	200	400	500	500
	DeneySEL Platform Modeli 5 (DPM5)	100	200	200	400	500	500
	DeneySEL Platform Modeli 6 (DPM6)	400	800	200	400	500	500
ZEMİN	DeneySEL Platform Modeli 7 (DPM7)	200	400	100	400	500	500
	DeneySEL Platform Modeli 8 (DPM8)	200	400	400	400	500	500
	DeneySEL Platform Modeli 9 (DPM9)	200	400	200	200	500	500
	DeneySEL Platform Modeli 10 (DPM10)	200	400	200	800	500	500
	DeneySEL Platform Modeli 11 (DPM11)	200	400	100	200	500	500
	DeneySEL Platform Modeli 12 (DPM12)	200	400	400	800	500	500
PİSTON	DeneySEL Platform Modeli 13 (DPM13)	200	400	200	400	250	500
	DeneySEL Platform Modeli 14 (DPM14)	200	400	200	400	1000	500
	DeneySEL Platform Modeli 15 (DPM15)	200	400	200	400	500	250
	DeneySEL Platform Modeli 16 (DPM16)	200	400	200	400	500	1000
	DeneySEL Platform Modeli 17 (DPM17)	200	400	200	400	250	250
	DeneySEL Platform Modeli 18 (DPM18)	200	400	200	400	1000	1000

Referans platform tasarım parametreleri Tablo 5. ile gösterilmiştir.

Tablo 5. ARPM'nin tasarım parametreleri

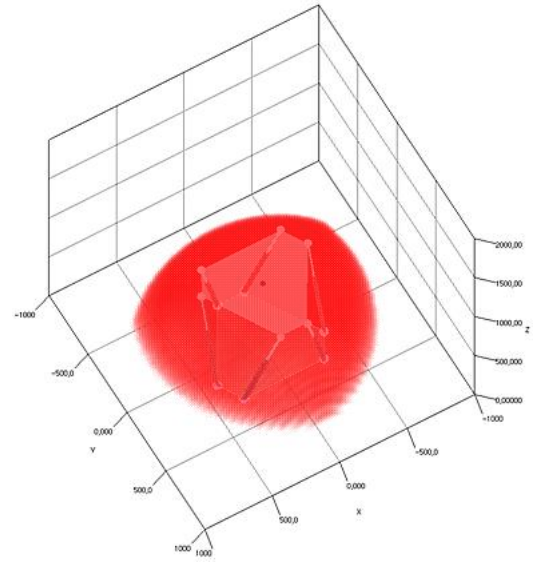
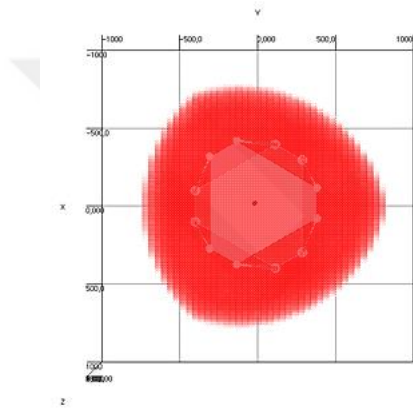
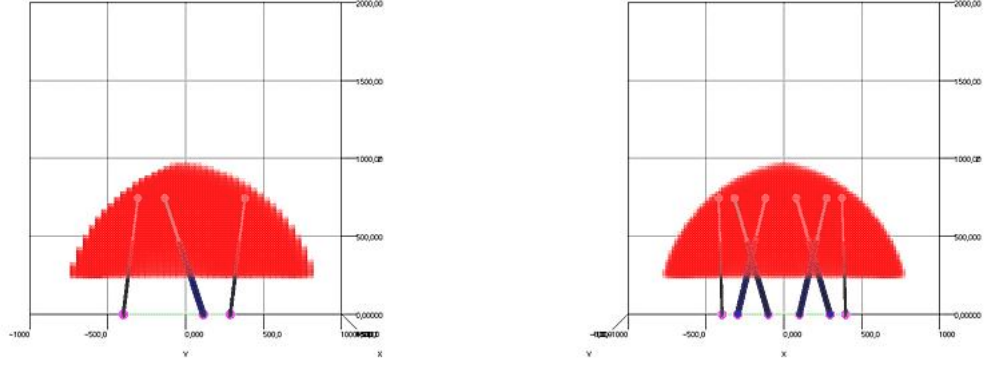
a	b	c	d	e	f
200mm	400mm	200mm	400mm	500mm	500mm

Tablo 5'de verilen tasarım parametreleriyle Şekil 23.'de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 23. Referans platform modelinin simülasyon ortamındaki görünümü

Tablo 5. parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 23. 'de verilmiştir.



Şekil 24. ARPM'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 24 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 6'da, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 7'de verilmiştir.

Tablo 6. ARPM'nin nokta bulutu hacmi ve nokta sayısı

Nokta Sayısı	Hacim
71225	569800 m ³

Tablo 7. ARPM'nin platform merkezine göre X, Y ve Z eksenlerindeki sistem sınırları

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar
En Az (min)	-760,2	-720	250
En Çok (max)	760,2	800	970
Aralık (Range)	1520,4	1520	970

ARPM'ye göre Tablo 4 ile belirtilen DPM'lerin çalışma uzayı analizleri ayrı başlıklar altında incelenmiştir. Çalışma uzayı analizi yapılırken rotasyonel hareket uygulanmadığı yani x, y ve z eksenlerinde yapılan açıların 0 olduğu kabul edilmiştir. Çalışma uzayı analizi platform merkezinin ulaşabildiği alanlar baz alınarak yapılmıştır. Ana Referans Platform Modeli (ARPM) değerleri daha önce anlatıldığı gibi örnek olarak alınmış ve a, b, c, d, e, f parametrelerinde bu değerlerin yine örnek olarak yarısı ve iki katı ele alınarak ortaya çıkan yeni model ile nokta bulutu analizi gerçekleştirilmiş ve uzay alanı sonuçları elde edilmiştir.

3.1. Hareketli Platform Tasarım Parametrelerinin Çalışma Uzayına Etkisi

Hareketli platform tasarımı, bir makine veya sistemin belirli bir görevi yerine getirebilmesi için gerekli olan hareketleri gerçekleştirilmesi için gereken parçaların ve bu parçaların birbirleriyle nasıl etkileşime girdiğinin belirlenmesi olarak tanımlanabilir. Hareketli platform tasarımı sırasında, tasarımcılar bir dizi parametre dikkate alırlar ve bu parametrelerin çalışma uzayına nasıl etki ettiğini değerlendirirler. Bu bölümde, hareketli platform tasarımı parametrelerinin çalışma uzayına etkileri üzerinde durulacaktır.

Hareketli platform tasarımı parametreleri, tasarım sürecinin her aşamasında dikkate alınması gereken önemli kavramlardır. Bunlar, hareketli platformun çalışma uzayına nasıl etki ettiğine dair bilgi verir ve tasarımın nasıl gerçekleştirileceğine dair önemli bilgiler sağlar. Aşağıda, hareketli platform tasarımı parametrelerinin çalışma uzayına nasıl etki ettiği üzerinde durulacak olan bazı önemli parametreler verilmiştir:

- Hareket eksenleri sayısı: Hareketli platform tasarımı sırasında, hareket eksenleri sayısı belirlenir. Bu parametre, hareketli platformun çalışma uzayında nasıl hareket edeceğine dair bilgi verir ve bu da hareketli platformun görevlerine uygun olup olmadığını belirlemede önemlidir. Örneğin, bir tekerlekli robotun hareket eksenleri sayısı belirli bir seviyenin üzerine çıkamayabilir, ancak bir manipülatörün hareket eksenleri sayısı daha fazla olabilir. Hareket eksenleri sayısı, hareketli platformun çalışma uzayında ne kadar alanı kapsayabileceğine dair bilgi verir.

- Hareket hızı: Hareketli platform tasarımı sırasında, hareket hızı da belirlenir. Bu parametre, hareketli platformun çalışma uzayında nasıl hareket edeceğine dair bilgi verir ve bu da hareketli platformun görevlerine uygun olup olmadığını belirlemede önemlidir. Örneğin, bir tekerlekli robotun hareket hızı belirli bir seviyenin üzerine çıkamayabilir, ancak bir manipülatörün hareket hızı daha yüksek olabilir. Hareket hızı, hareketli platformun çalışma uzayında ne kadar hızlı hareket edebileceğine dair bilgi verir.
- İşlem hassasiyeti: Hareketli platform tasarımı sırasında, işlem hassasiyeti de belirlenir. Bu parametre, hareketli platformun çalışma uzayında ne kadar hassas hareket edebileceğine dair bilgi verir ve bu da hareketli platformun görevlerine uygun olup olmadığını belirlemede önemlidir. Örneğin, bir tekerlekli robotun işlem hassasiyeti belirli bir seviyenin üzerine çıkamayabilir, ancak bir manipülatörün işlem hassasiyeti daha yüksek olabilir. İşlem hassasiyeti, hareketli platformun çalışma uzayında ne kadar hassas çalışabileceğine dair bilgi verir.

Bu parametrelerin doğru bir şekilde belirlenmesi, hareketli platformun çalışma uzayında başarılı bir şekilde çalışmasını sağlar.

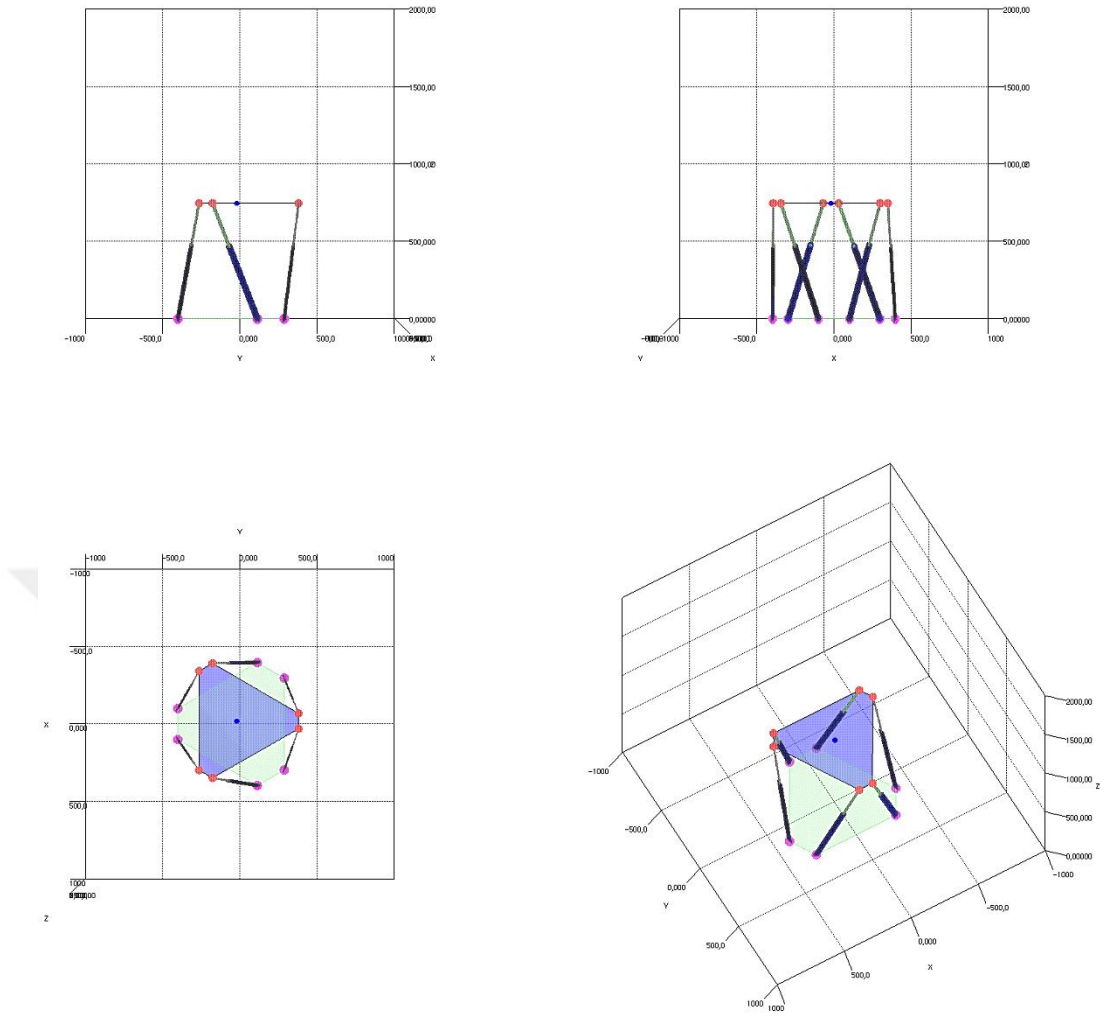
Şekil 22 ile verilen "a" parametresi 200mm'den 100mm'ye düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Tabloda 8.'de belirtildiği üzere a parametresinin yarısı olan *a öncelikle ele alınmıştır. *a değeri ile yeni bir nokta bulutu analizi gerçekleştirilmiştir. Deneysel platform modeli 1'in (DPM1) tasarım parametreleri tablo Tablo 8 ile verilmiştir.

Tablo 8. DPM1'in tasarım parametreleri

*a	b	c	d	e	f
100mm	400mm	200mm	400mm	500mm	500mm

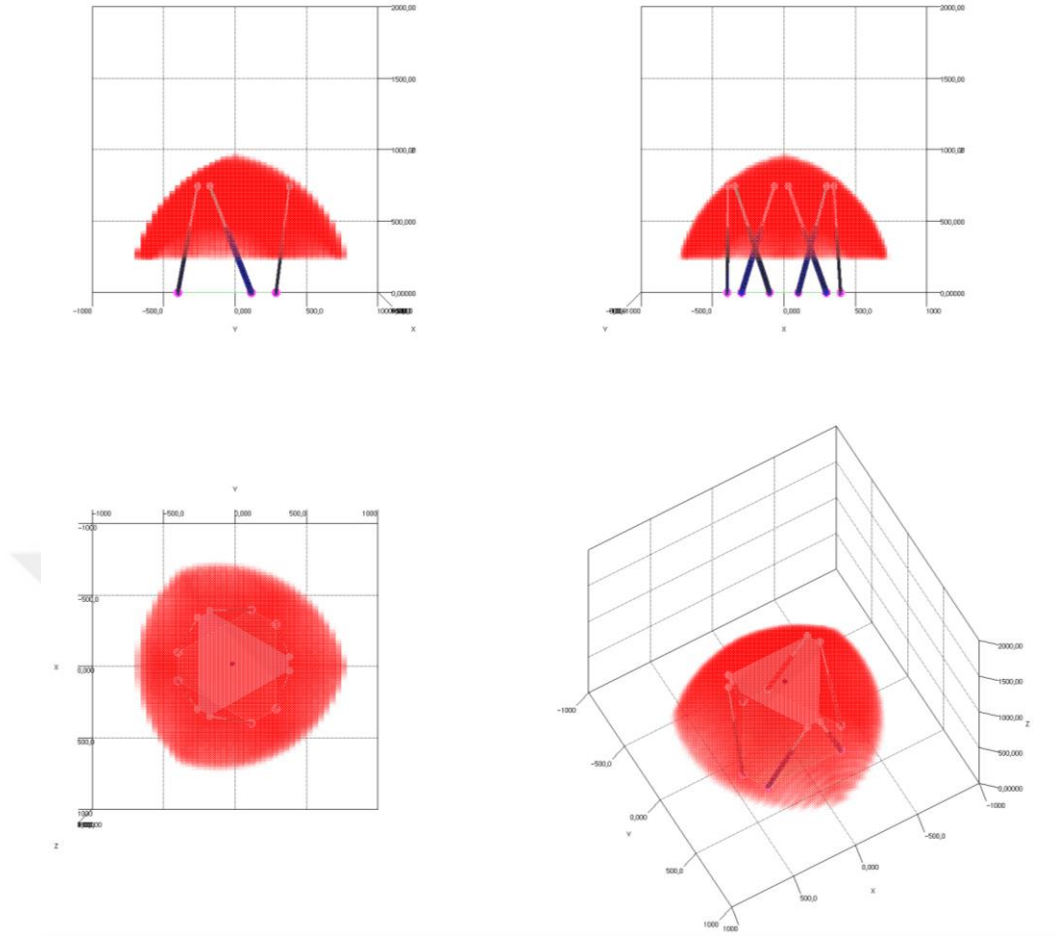
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 8'de verilen tasarım parametreleriyle Şekil 24'de gösterilen 3B platform tasarımı modeli oluşmaktadır.



Şekil 25. DPM1'in simülasyon ortamındaki görünümü

Tablo 8. parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 26'de verilmiştir.



Şekil 26. DPM1'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 26 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 9'da, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 10'da verilmiştir.

Tablo 9. DPM1'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
52996	423968 m ³	-25,59%

Tablo 10. DPM1'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-720mm	-680mm	250mm	-5,26%	-5,56%	0,00%
En çok (max)	720mm	760mm	950mm	-5,26%	-5,00%	-2,06%
Aralık (range)	1440mm	1440mm	700mm	-5,26%	-5,26%	-2,78%

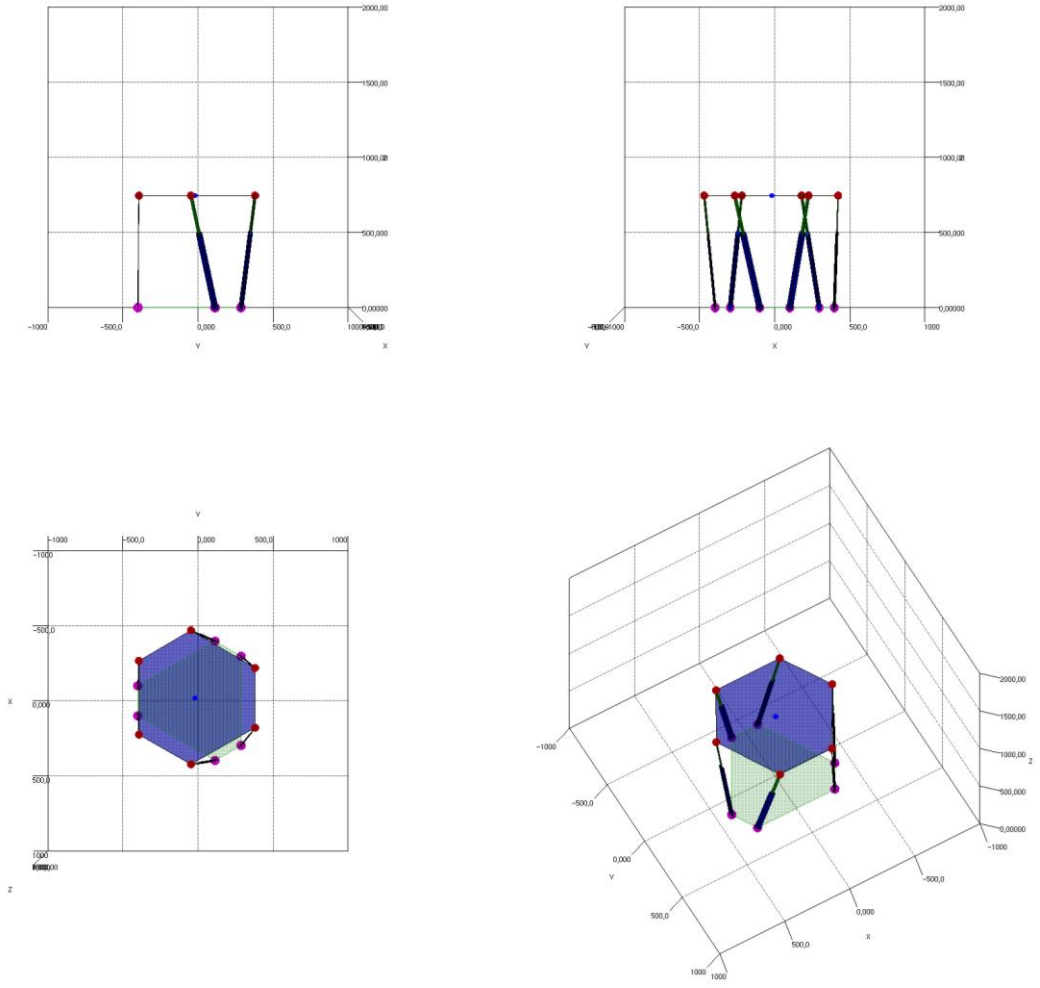
Şekil 22 ile verilen "a" parametresi 200mm'den 400mm'ye çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 2'nin (DPM2) tasarım parametreleri tablo Tablo 11 ile verilmiştir. Tablo 11.'de a parametresinin iki katı olarak **a değeri verilmiştir. Bu değeri ile platformun yeni modeli ile nokta bulutu analizi gerçekleştirilmiştir.

Tablo 11. DPM2'nin tasarım parametreleri

**a	b	c	d	e	f
400mm	400mm	200mm	400mm	500mm	500mm

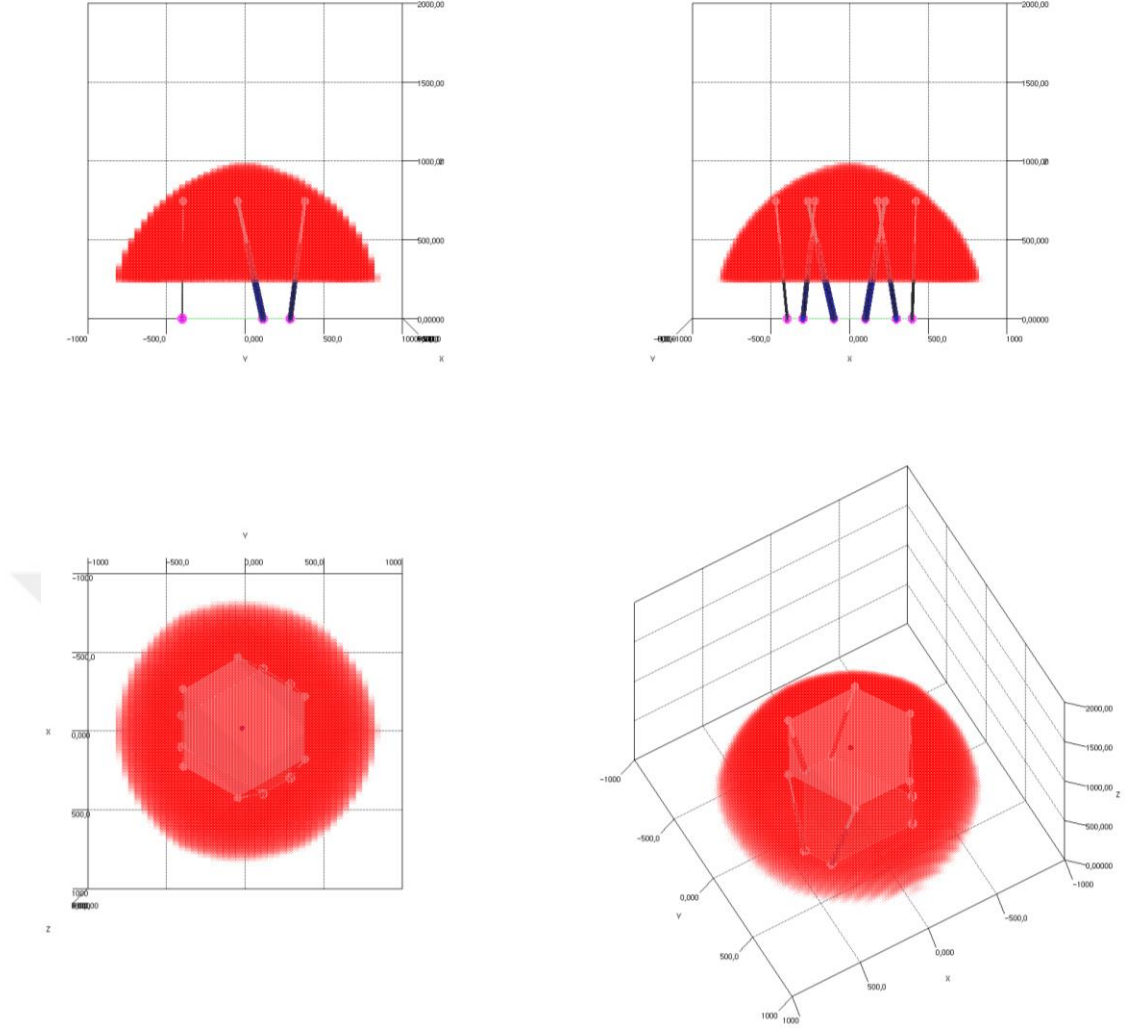
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 11'de verilen tasarım parametreleriyle Şekil 27 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 27. DPM2'nin simülasyon ortamındaki görünümü

Tablo 11 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde ± 1000 mm 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 28'de verilmiştir.



Şekil 28. DPM2'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 28 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 12'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 13'de verilmiştir.

Tablo 12. DPM2'nin nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
91673	733384 m ³	28,71%

Tablo 13. DPM2'nin platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	X Deęişim Oranı	Z Deęişim Oranı
En az (min)	-820mm	-800mm	250mm	7,89%	11,11%	0,00%
En çok (max)	820mm	840mm	970mm	7,89%	5,00%	0,00%
Aralık (range)	1640mm	1640mm	720mm	7,89%	7,89%	0,00%

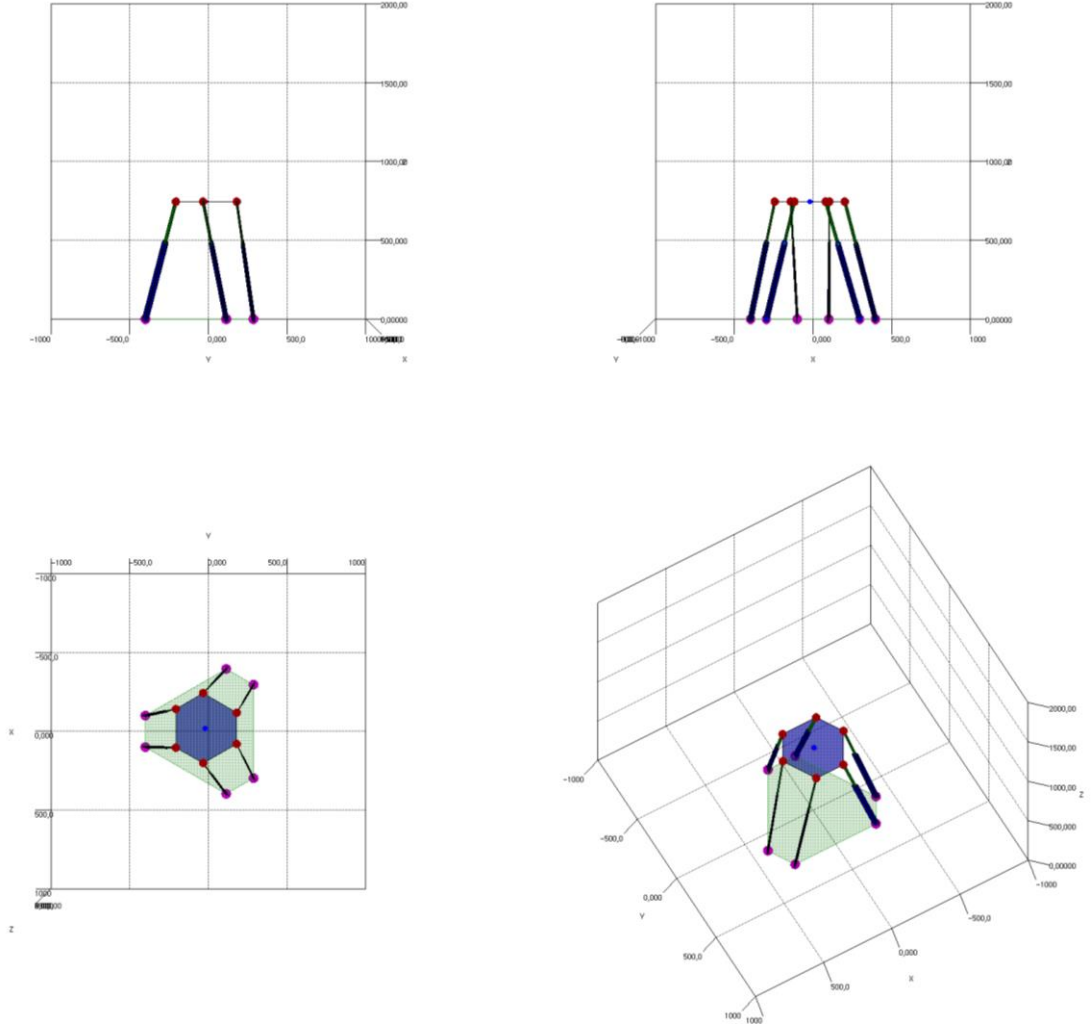
Şekil 22 ile verilen "b" parametresi 400mm'den 200mm'ye düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 3'ün (DPM3) tasarım parametreleri tablo Tablo 14 ile verilmiştir.

Tablo 14. DPM3'ün tasarım parametreleri

a	*b	c	d	e	f
200mm	200mm	200mm	400mm	500mm	500mm

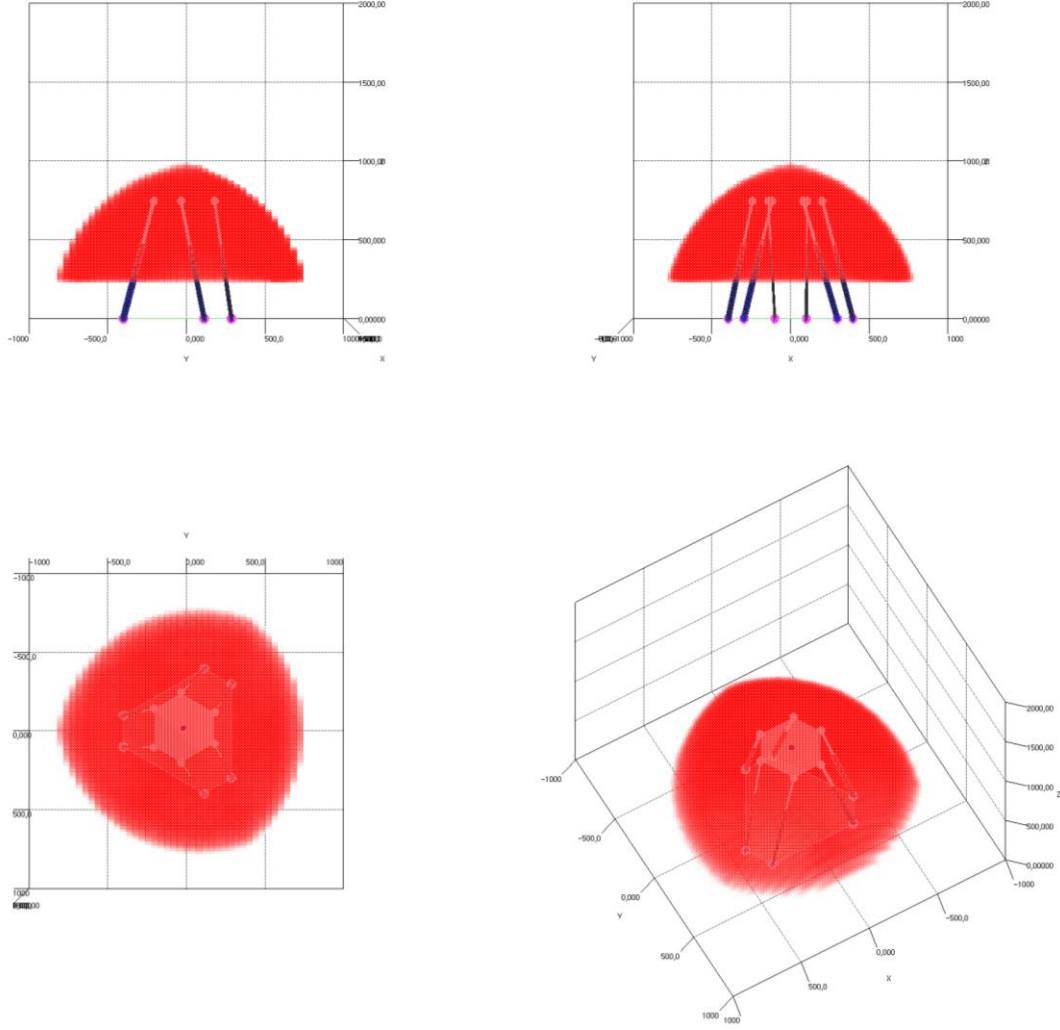
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 14 'de verilen tasarım parametreleriyle Şekil 29 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 29. DPM3'ün simülasyon ortamındaki görünümü

Tablo 14 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 30'de verilmiştir.



Şekil 30. DPM3'ün çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 30 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 15'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo_XX1109'de verilmiştir.

Tablo 15. DPM3'ün nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
71525	572200 m ³	0,41%

Tablo 16. DPM3'ün platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-760mm	-800mm	250mm	0,00%	11,11%	0,00%
En çok (max)	760mm	720mm	970mm	0,00%	-10,00%	0,00%
Aralık (range)	1520mm	1520mm	720mm	0,00%	0,00%	0,00%

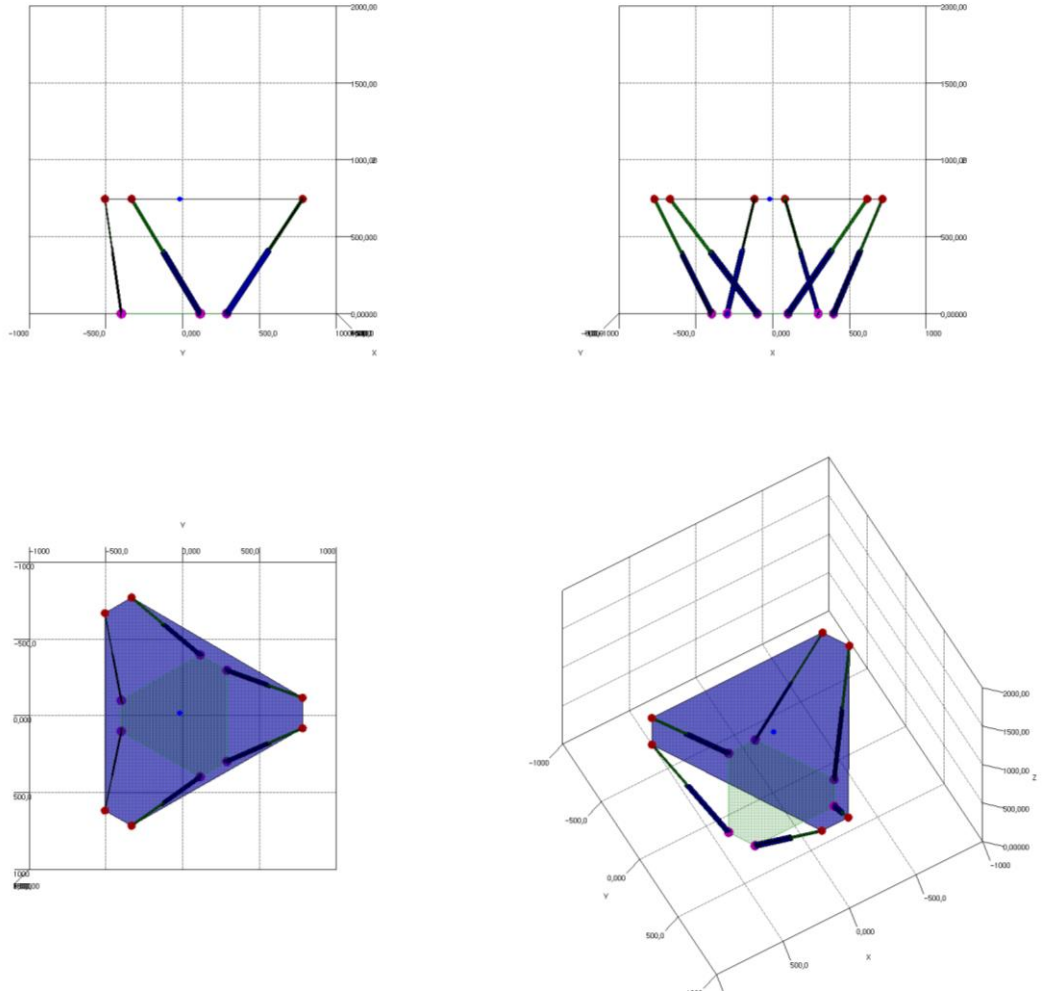
Şekil 22 ile verilen "b" parametresi 400mm'den 800mm'ye çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 4'ün (DPM4) tasarım parametreleri tablo Tablo 17 ile verilmiştir.

Tablo 17. DPM4'ün tasarım parametreleri

a	**b	c	d	e	f
200mm	800mm	200mm	400mm	500mm	500mm

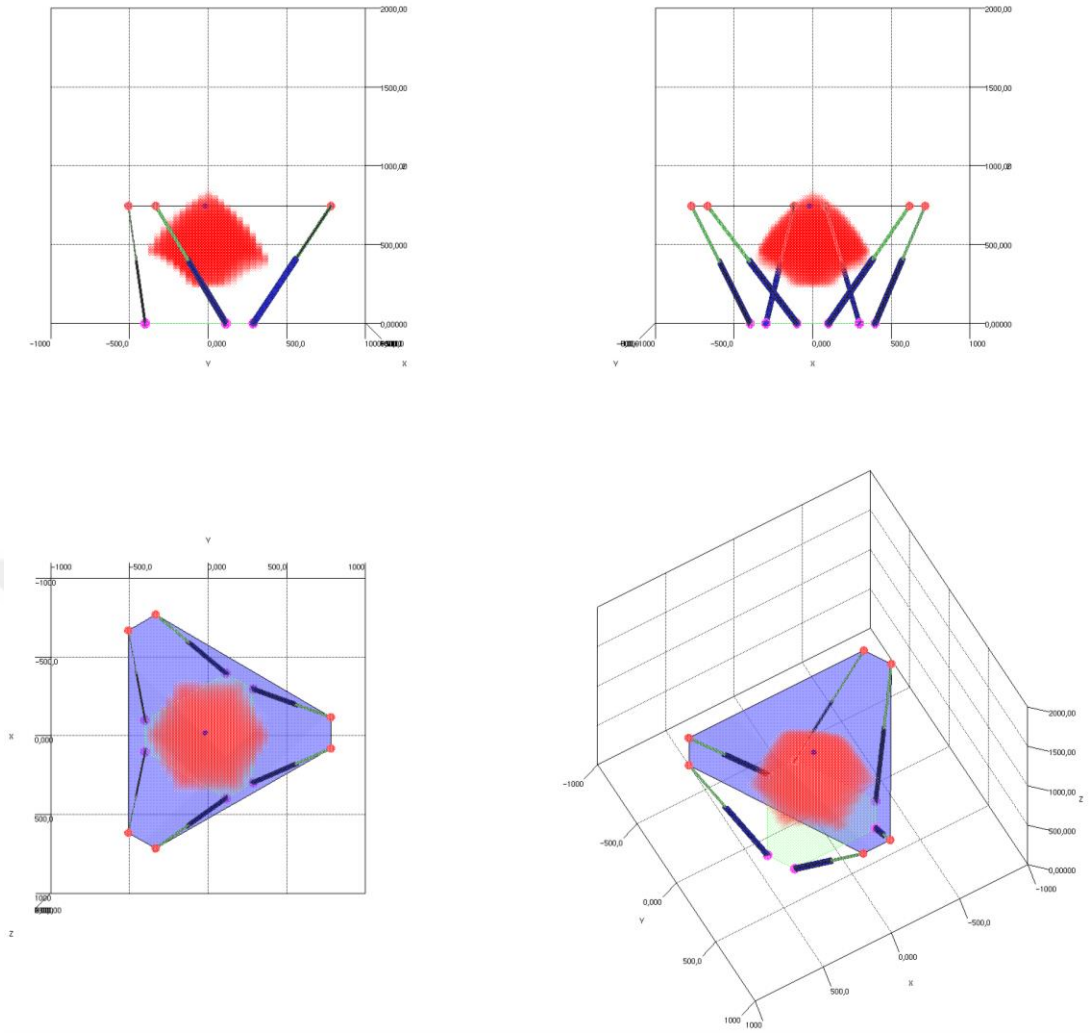
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 17'de verilen tasarım parametreleriyle Şekil 31 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 31. DPM4'ün simülasyon ortamındaki görünümü

Tablo 17 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 32'de verilmiştir.



Şekil 32. DPM4'ün çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 32 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 18'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 19'de verilmiştir.

Tablo 18. DPM4'ün nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
12727	101816 m ³	-82,12%

Tablo 19. DPM4'ün platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-340mm	-360mm	250mm	-55,25%	-50,00%	0,00%
En çok (max)	340mm	360mm	830mm	-55,25%	-55,00%	-14,43%
Aralık (range)	680mm	720mm	580mm	-55,25%	-52,63%	-19,44%

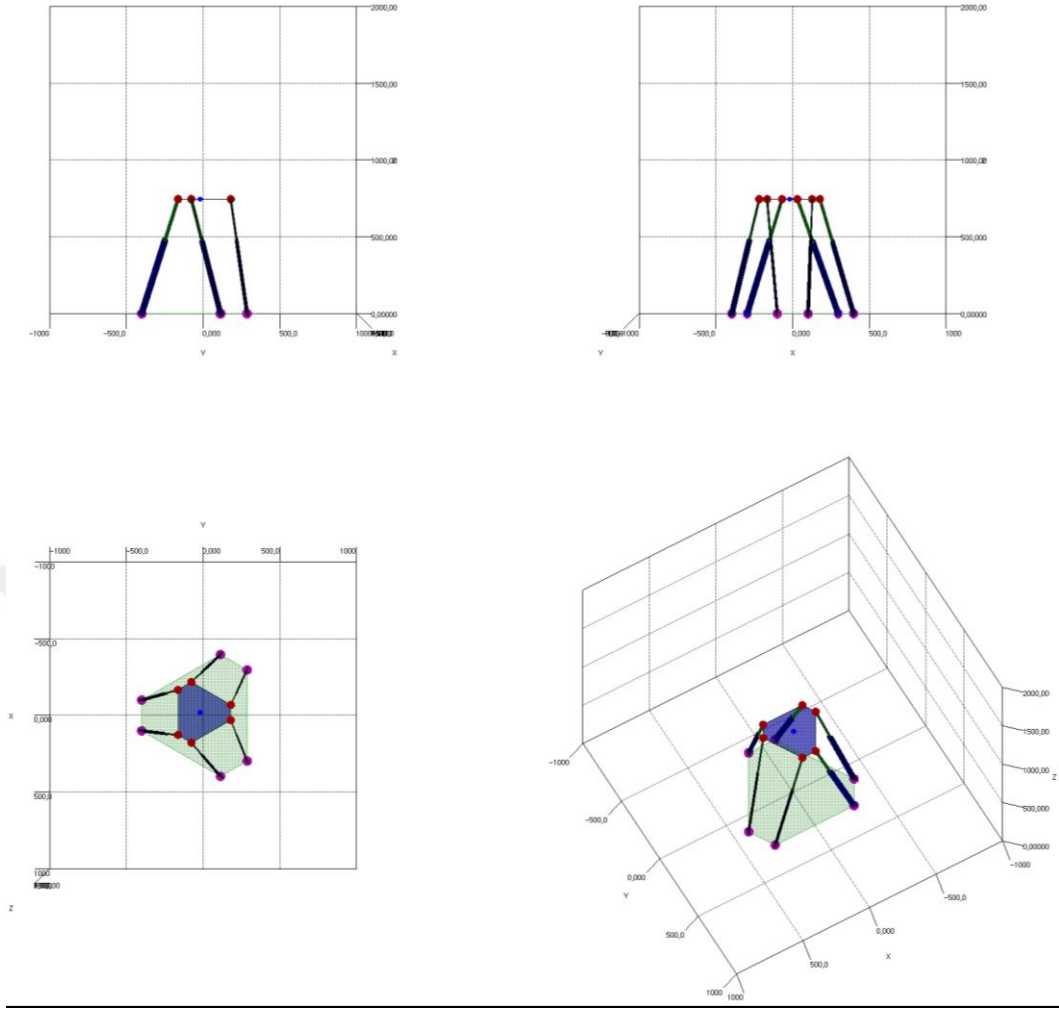
Şekil 22 ile verilen "a" ve "b" parametreleri yarı yarıya düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 5'in (DPM5) tasarım parametreleri tablo Tablo 20 ile verilmiştir.

Tablo 20. DPM5'in tasarım parametreleri

*a	*b	c	d	e	f
100mm	200mm	200mm	400mm	500mm	500mm

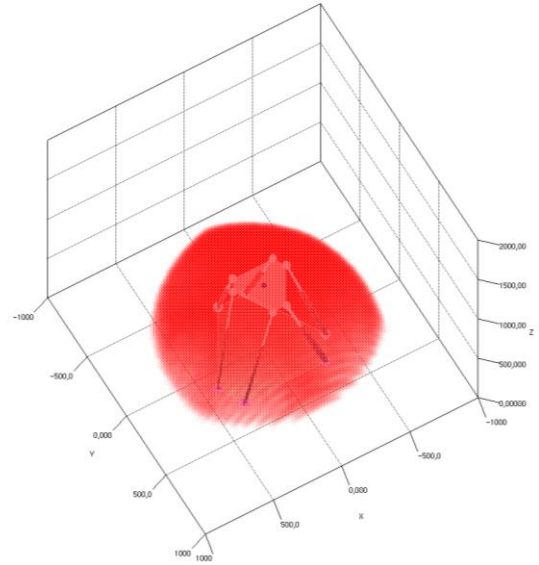
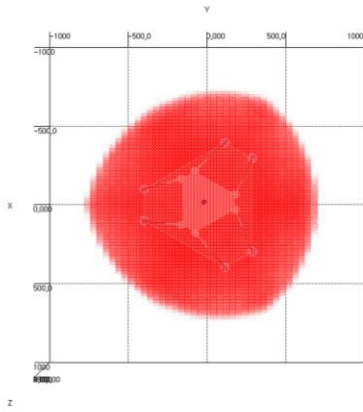
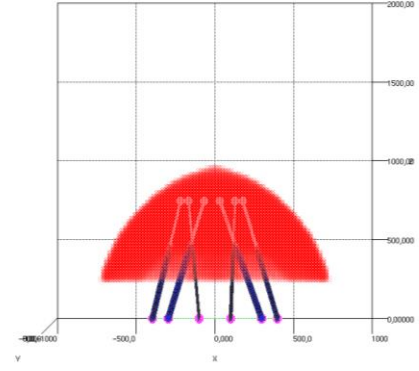
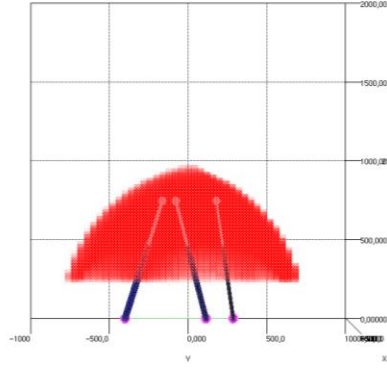
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 20'de verilen tasarım parametreleriyle Şekil 33 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 33. DPM5'in simülasyon ortamındaki görünümü

Tablo 20 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde ± 1000 mm 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 34'de verilmiştir.



Şekil 34. DPM5'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 34 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 21 'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 22 'de verilmiştir.

Tablo 21. DPM5'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
52881	423048 m ³	-25,76%

Tablo 22. DPM5'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-720mm	-760mm	250mm	-5,26%	5,56%	5,56%
En çok (max)	720mm	680mm	950mm	-5,26%	-15,00%	-15,00%
Aralık (range)	1440mm	1440mm	700mm	-5,26%	-5,26%	-5,26%

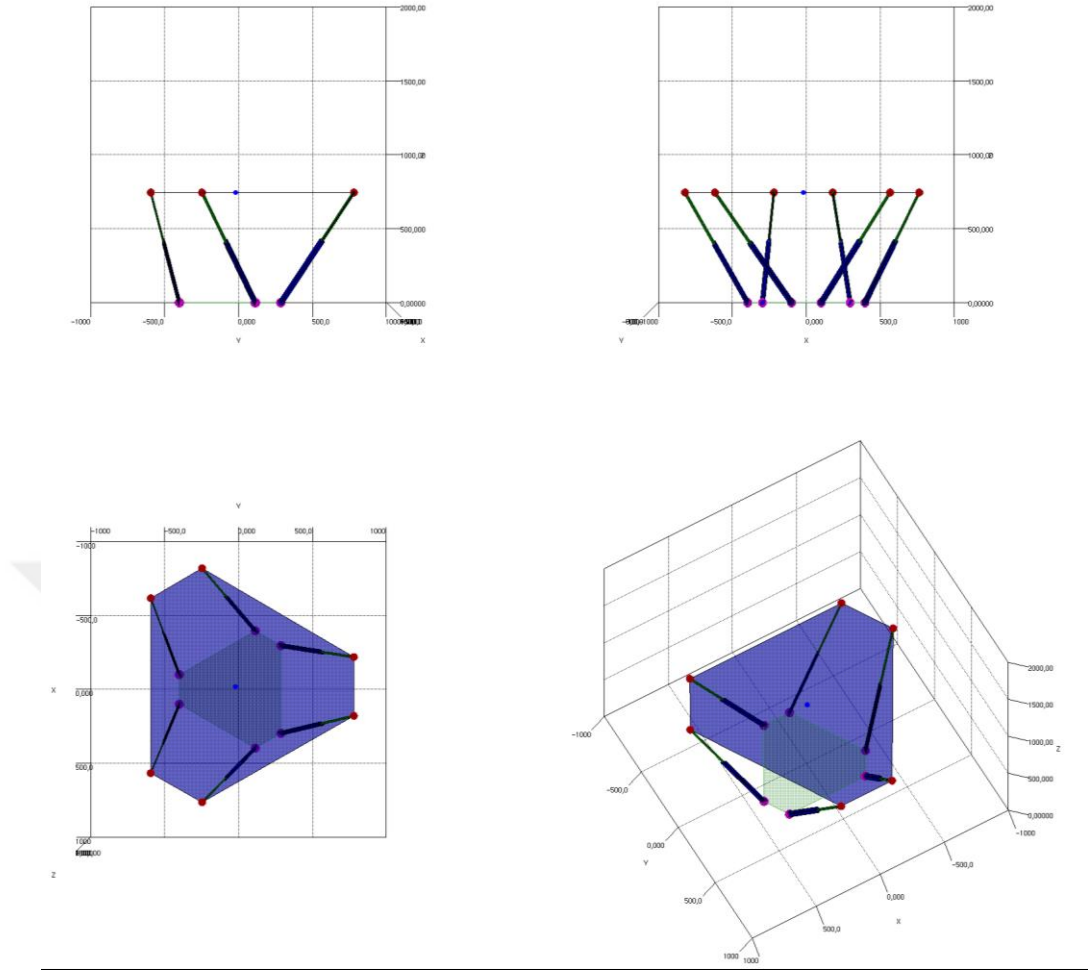
Şekil 22 ile verilen "a" ve "b" parametreleri iki katına çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 6'nın (DPM6) tasarım parametreleri tablo Tablo 23 ile verilmiştir.

Tablo 23. DPM6'nın tasarım parametreleri

**a	**b	c	d	e	f
400mm	800mm	200mm	400mm	500mm	500mm

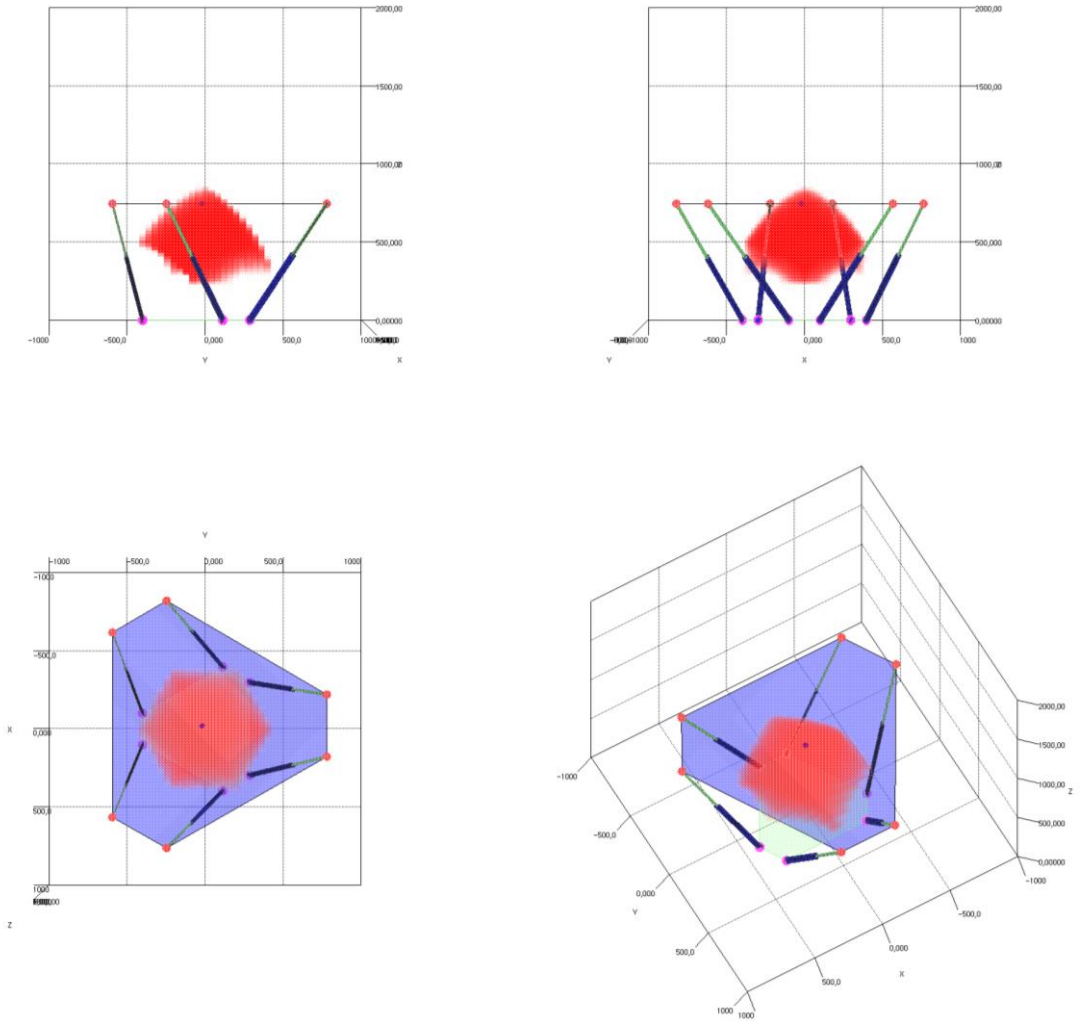
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 23'de verilen tasarım parametreleriyle Şekil 35 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 35. DPM6'nın simülasyon ortamındaki görünümü

Tablo 23 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 36'de verilmiştir.



Şekil 36. DPM6'nın çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 36 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 24'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 25'de verilmiştir.

Tablo 24. DPM6'nın nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
15093	120744 m ³	-78,81%

Tablo 25. DPM6'nın platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-360mm	-400mm	250mm	-52,62%	-44,44%	0,00%
En çok (max)	380mm	400mm	830mm	-49,99%	-50,00%	-14,43%
Aralık (range)	740mm	800mm	580mm	-51,30%	-47,37%	-19,44%

3.2. Sabit Alt Zemin Tasarım Parametrelerinin Çalışma Uzayına Etkisi

Sabit alt zemin tasarımı, bir makine veya sistemin belirli bir görevi yerine getirebilmesi için gerekli olan hareketleri gerçekleştirilmesi için gereken parçaların ve bu parçaların birbirleriyle nasıl etkileşime girdiğinin belirlenmesi olarak tanımlanabilir. Sabit alt zemin tasarımı sırasında, tasarımcılar bir dizi parametre dikkate alırlar ve bu parametrelerin çalışma uzayına nasıl etki ettiğini değerlendirirler. Bu bölümde, sabit alt zemin tasarımı parametrelerinin çalışma uzayına etkileri üzerinde durulacaktır.

Sabit alt zemin tasarımı parametreleri, tasarım sürecinin her aşamasında dikkate alınması gereken önemli kavramlardır. Bunlar, sabit alt zeminin çalışma uzayına nasıl etki ettiğine dair bilgi verir ve tasarımın nasıl gerçekleştirileceğine dair önemli bilgiler sağlar.

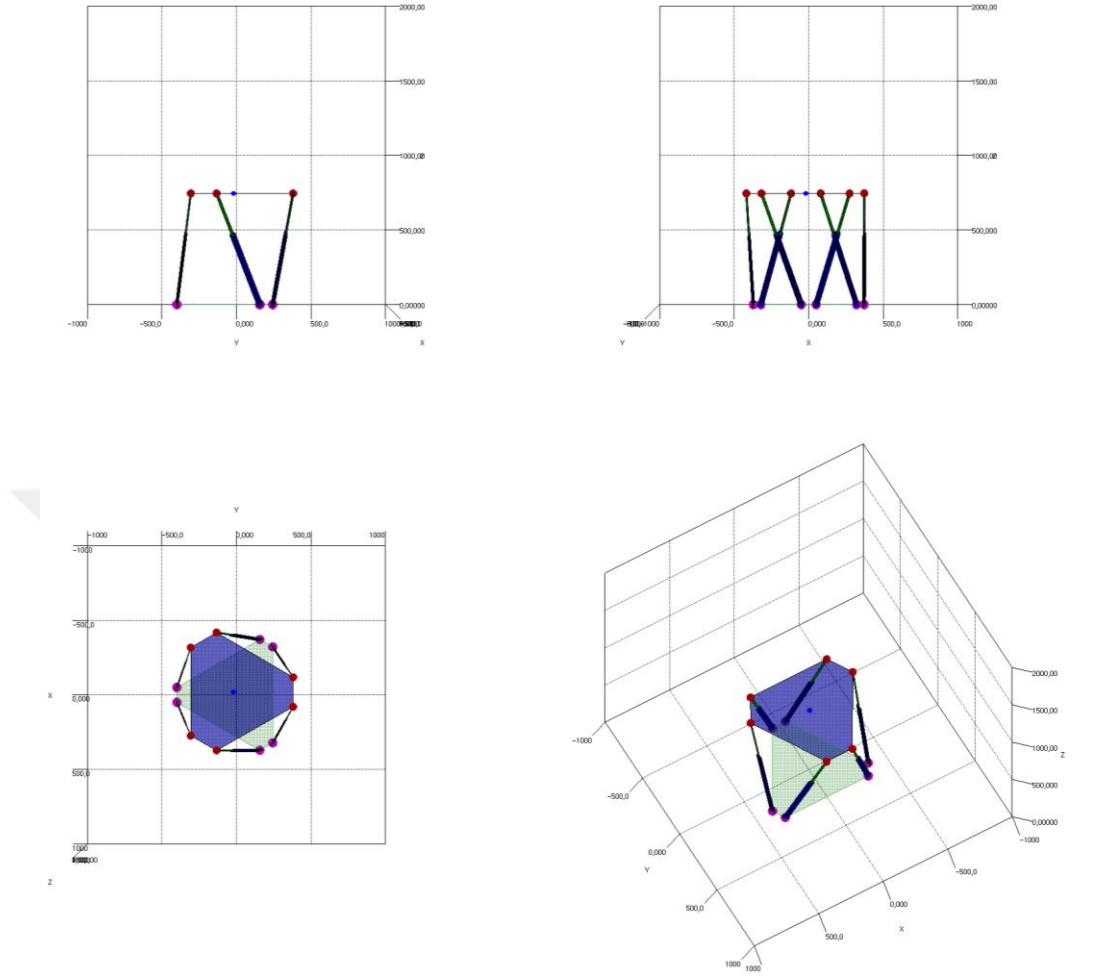
Şekil 22 ile verilen "c" parametresi 200mm'den 100mm'ye düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 7'nin (DPM7) tasarım parametreleri tablo Tablo 26 ile verilmiştir.

Tablo 26. DPM7'nin tasarım parametreleri

a	b	*c	d	e	f
200mm	400mm	100mm	400mm	500mm	500mm

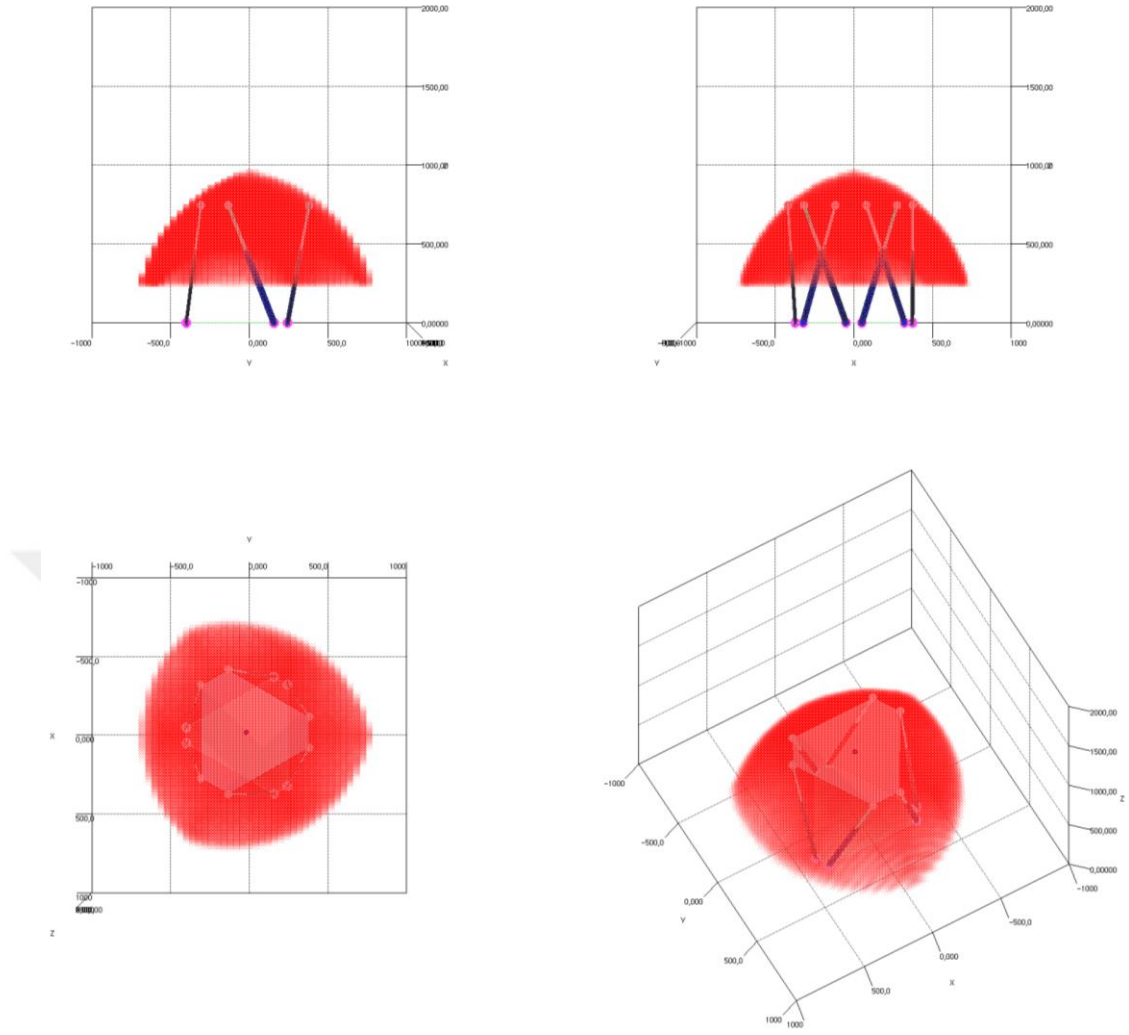
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 26'de verilen tasarım parametreleriyle Şekil 37 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 37. DPM7'nin simülasyon ortamındaki görünümü

Tablo 26 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 38'de verilmiştir.



Şekil 38. DPM7'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 38 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 27'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 28'de verilmiştir.

Tablo 27. DPM7'nin nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
53012	424096 m ³	-25,57%

Tablo 28. DPM7'nin platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-720mm	-680mm	250mm	-5,26%	-5,56%	0,00%
En çok (max)	720mm	760mm	950mm	-5,26%	-5,00%	-2,06%
Aralık (range)	1440mm	1440mm	700mm	-5,26%	-5,26%	-2,78%

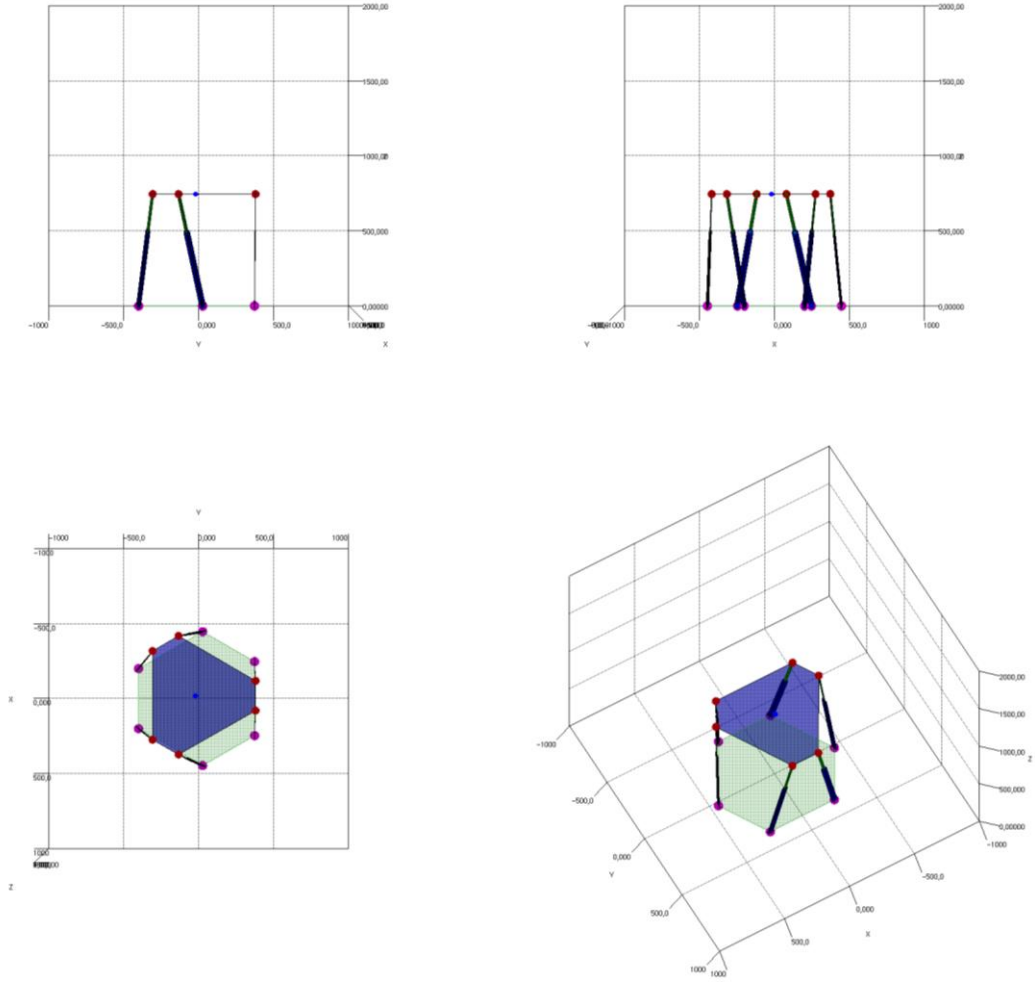
Şekil 22 ile verilen "c" parametresi 200mm'den 400mm'ye çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 8'in (DPM8) tasarım parametreleri tablo Tablo 29 ile verilmiştir.

Tablo 29. DPM8'in tasarım parametreleri

a	b	**c	d	e	f
200mm	400mm	400mm	400mm	500mm	500mm

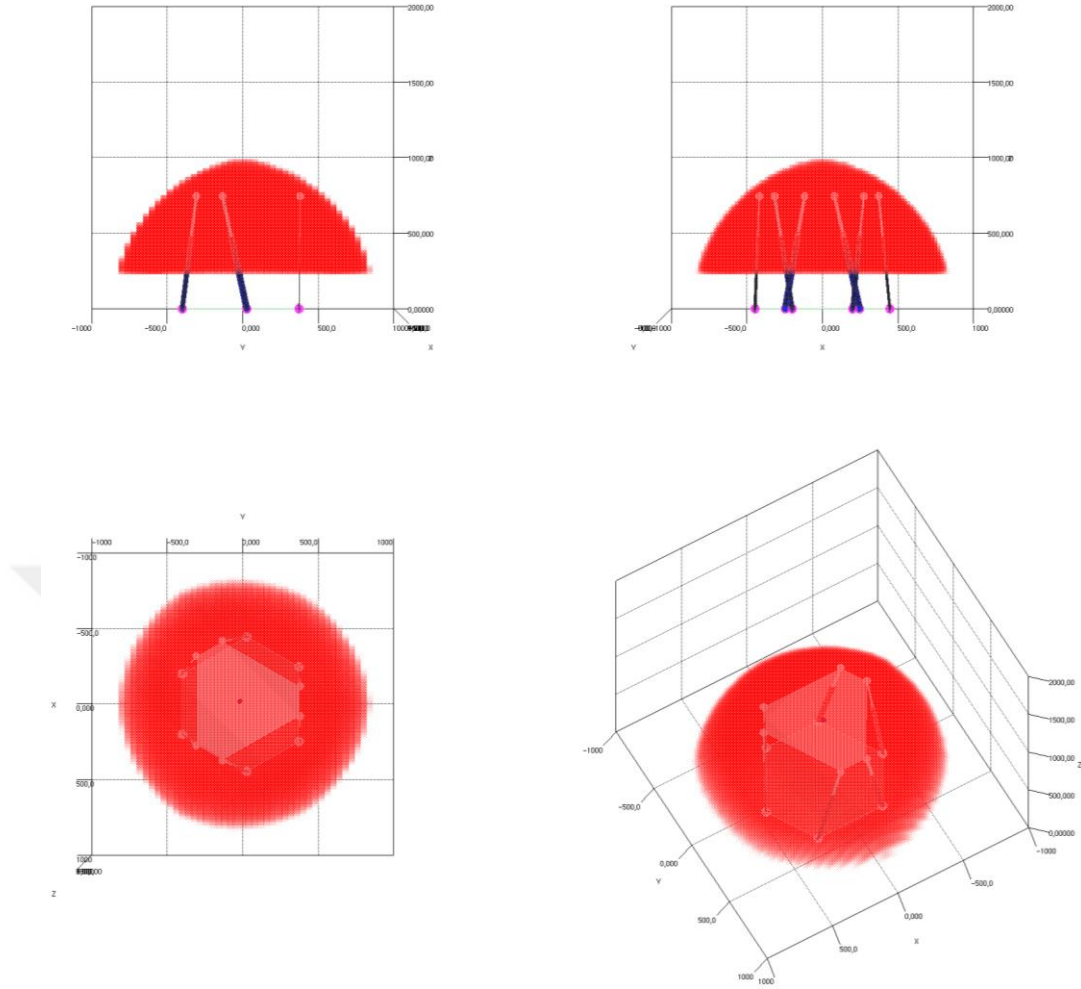
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 29'de verilen tasarım parametreleriyle Şekil 39 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 39. DPM8'in simülasyon ortamındaki görünümü

Tablo 29 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde ± 1000 mm 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 40'de verilmiştir.



Şekil 40. DPM8'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 40 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 30'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 31'de verilmiştir.

Tablo 30. DPM8'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
91603	732824 m ³	28,60%

Tablo 31. DPM8'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-820mm	-800mm	250mm	7,89%	11,11%	0,00%
En çok (max)	820mm	840mm	970mm	7,89%	5,00%	0,00%
Aralık (range)	1640mm	1640mm	720mm	7,89%	7,89%	0,00%

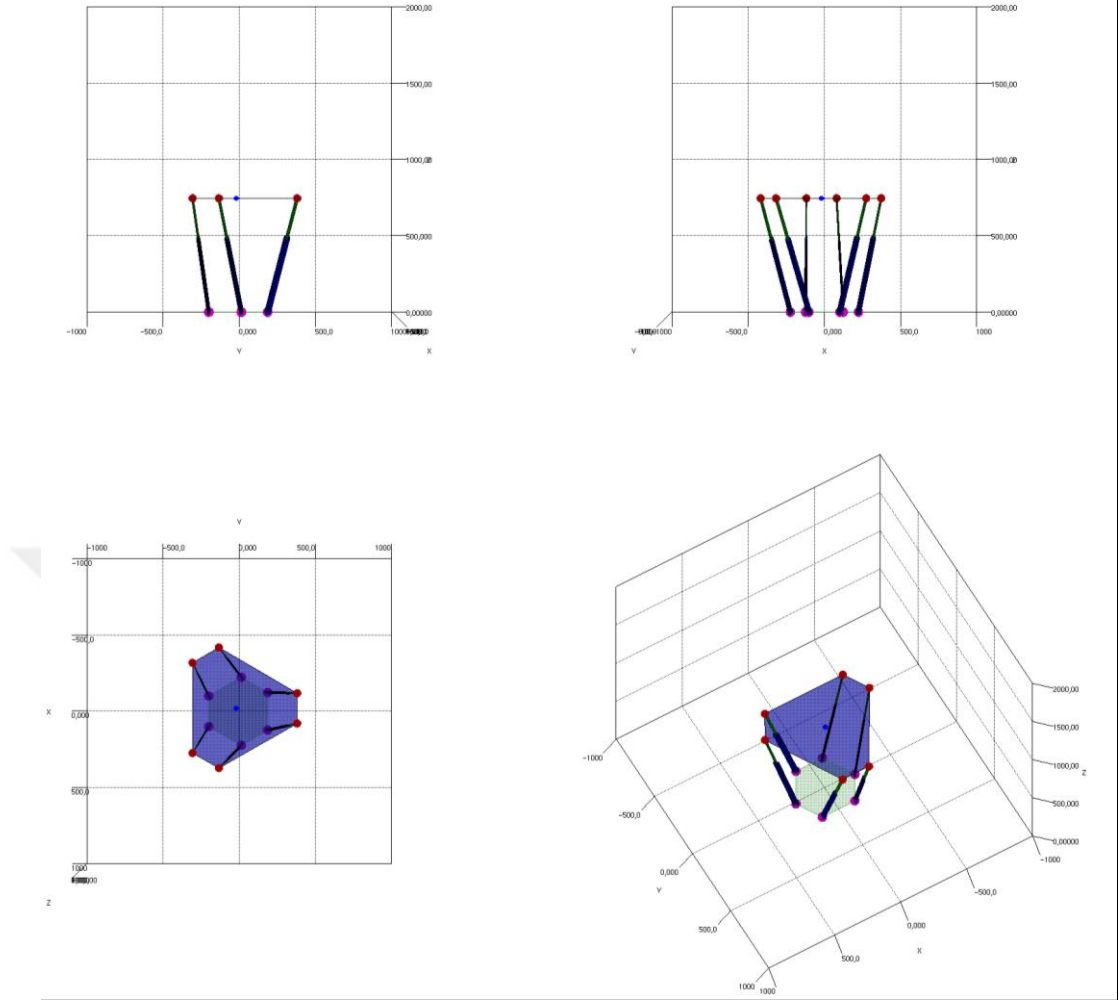
Şekil 22 ile verilen "d" parametresi 400mm'den 200mm'ye düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 9'un (DPM9) tasarım parametreleri tablo Tablo 32 ile verilmiştir.

Tablo 32. DPM9'un tasarım parametreleri

a	b	c	*d	e	f
200mm	400mm	200mm	200mm	500mm	500mm

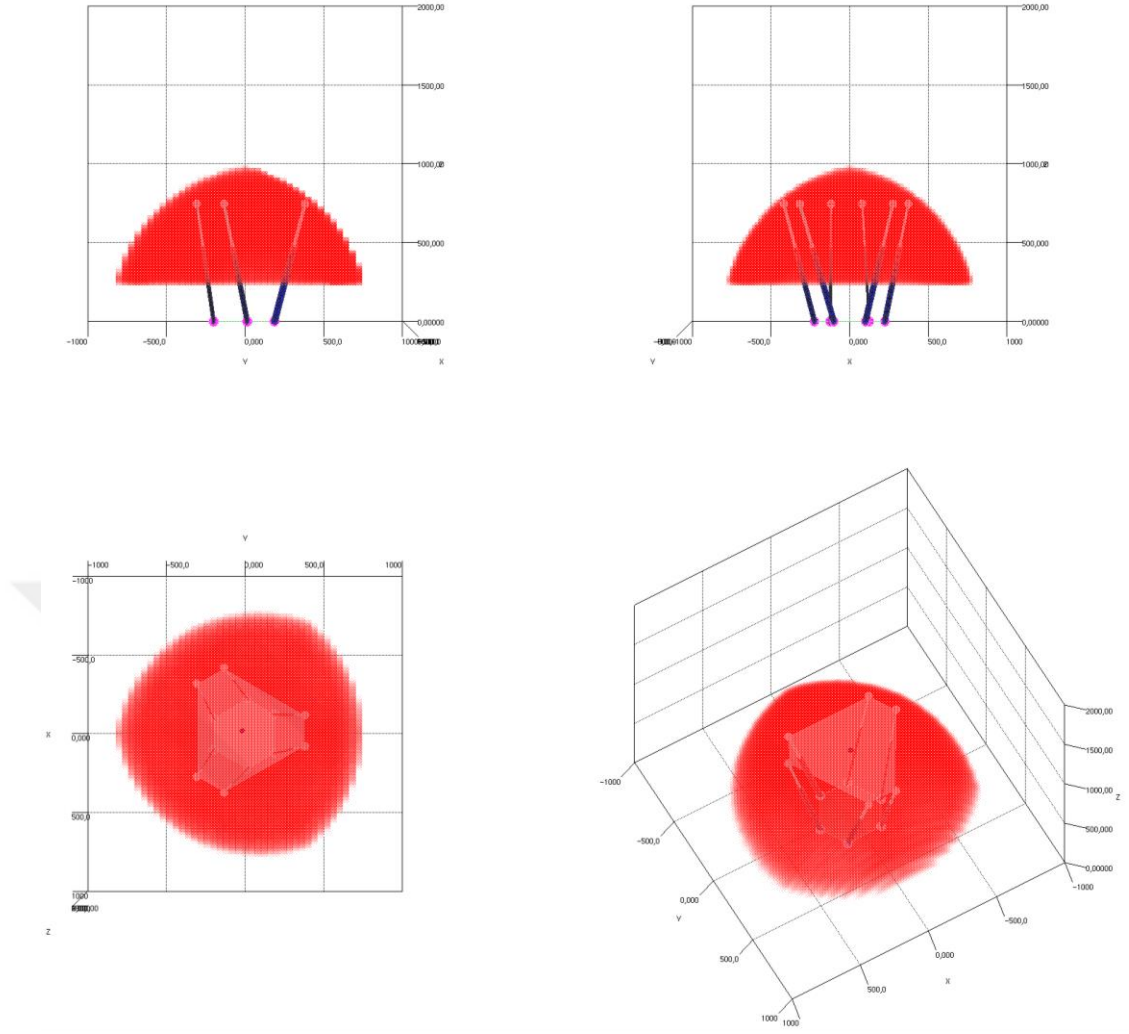
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 32'de verilen tasarım parametreleriyle Şekil 41 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 41. DPM9'un simülasyon ortamındaki görünümü

Tablo 32 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 42'de verilmiştir.



Şekil 42. DPM9'un çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 42 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 33'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 34'de verilmiştir.

Tablo 33. DPM9'un nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
71594	572752 m ³	0,52%

Tablo 34. DPM9'un platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-760mm	-800mm	250mm	0,00%	11,11%	0,00%
En çok (max)	760mm	720mm	970mm	0,00%	-10,00%	0,00%
Aralık (range)	1520mm	1520mm	720mm	0,00%	0,00%	0,00%

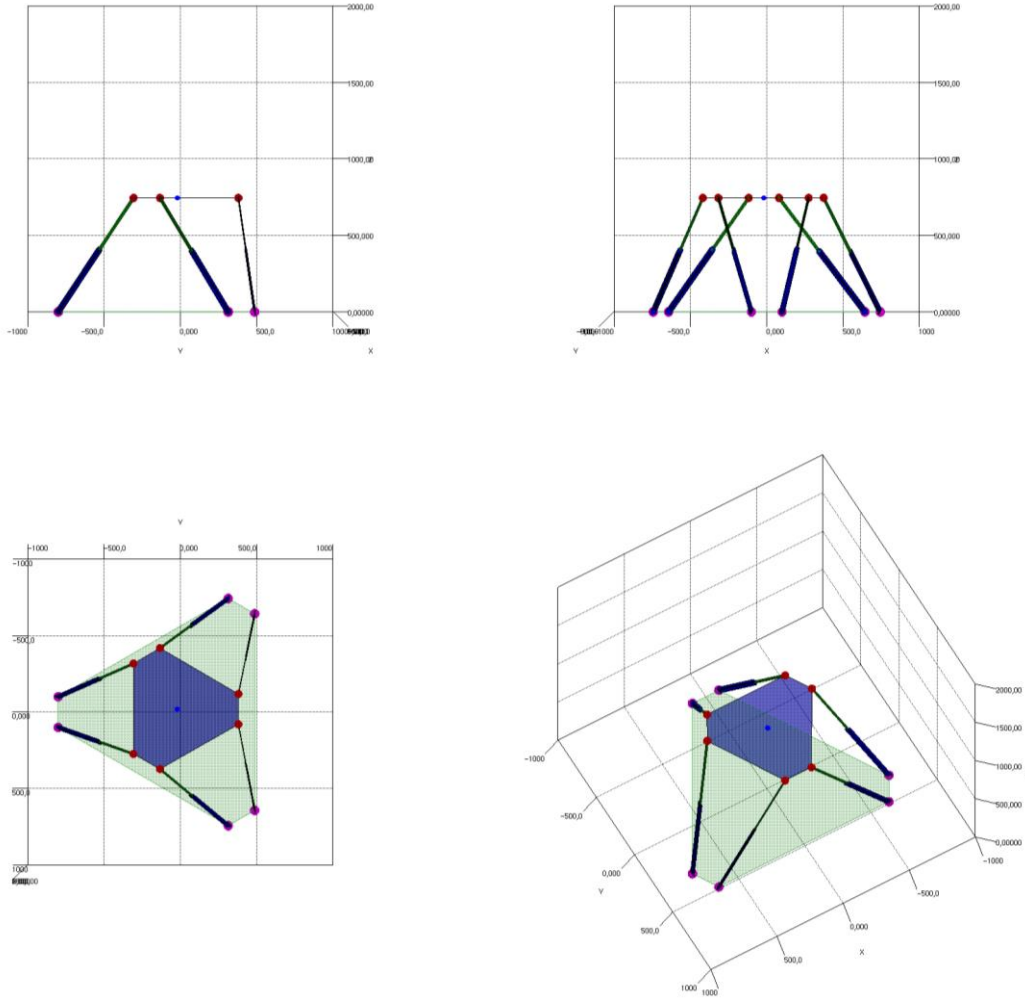
Şekil 22 ile verilen "d" parametresi 400mm'den 800mm'e çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 10'un (DPM10) tasarım parametreleri tablo Tablo 35 ile verilmiştir.

Tablo 35. DPM10'un tasarım parametreleri

a	b	c	**d	e	f
200mm	400mm	200mm	800mm	500mm	500mm

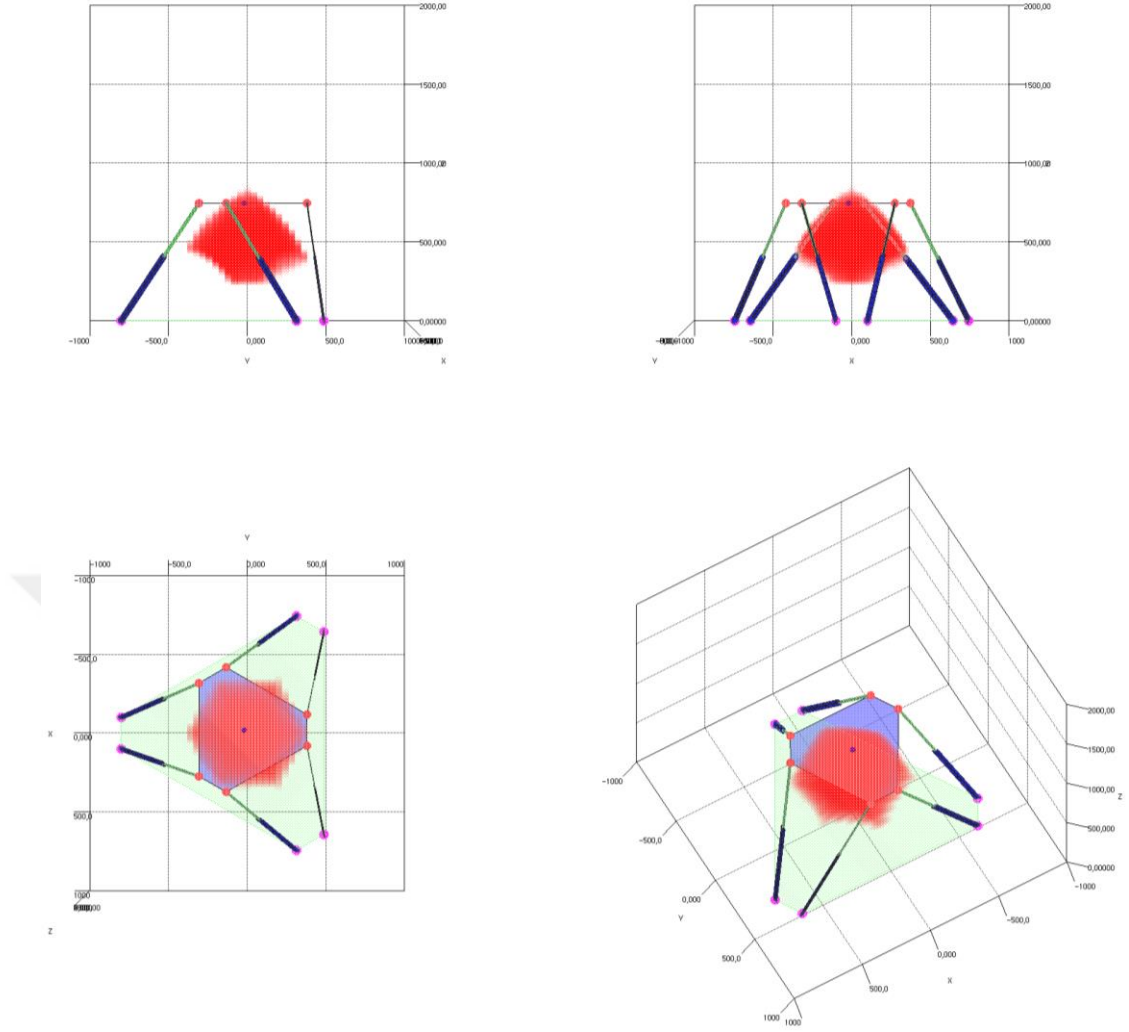
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 35'de verilen tasarım parametreleriyle Şekil 43 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 43. DPM10'un simülasyon ortamındaki görünümü

Tablo 35 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 44'de verilmiştir.



Şekil 44. DPM10'un çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 44 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 36'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 37'de verilmiştir.

Tablo 36. DPM10'un nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
12718	101744 m ³	-82,14%

Tablo 37. DPM10'un platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-340mm	-360mm	250mm	-55,25%	-50,00%	0,00%
En çok (max)	340mm	360mm	830mm	-55,25%	-55,00%	-14,43%
Aralık (range)	680mm	720mm	580mm	-55,25%	-52,63%	-19,44%

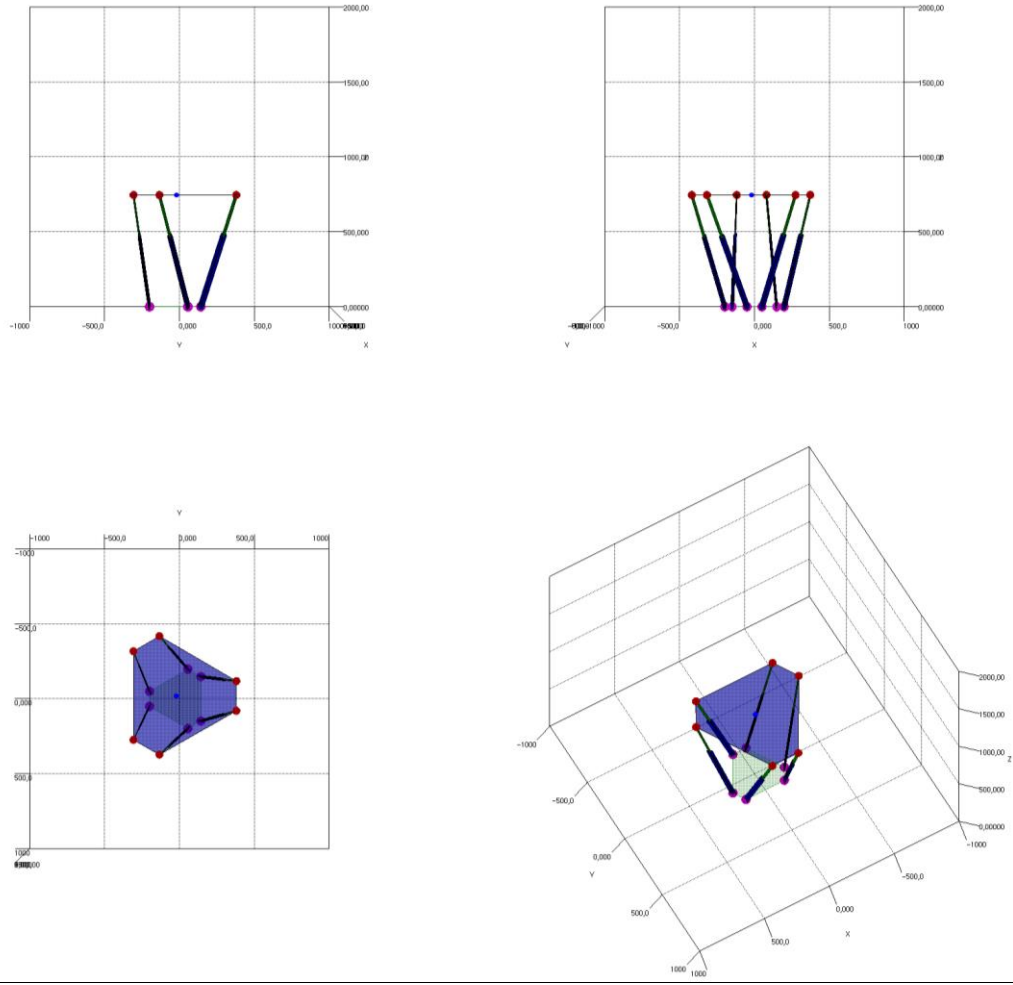
Şekil 22 ile verilen "c" ve "d" parametreleri yarı yarıya düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 11'in (DPM11) tasarım parametreleri tablo Tablo 38 ile verilmiştir.

Tablo 38. DPM11'in tasarım parametreleri

a	b	*c	*d	e	f
200mm	400mm	100mm	200mm	500mm	500mm

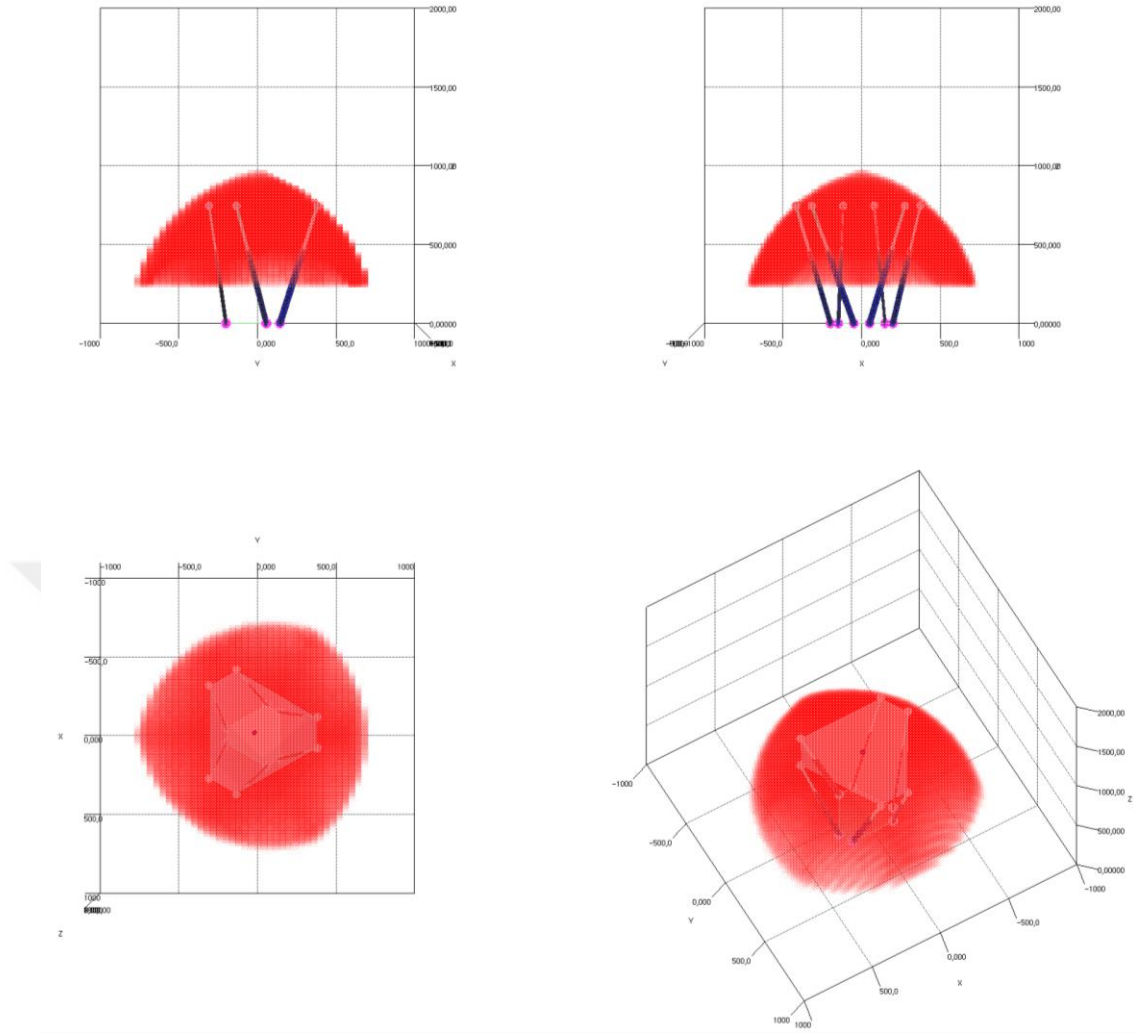
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 38'de verilen tasarım parametreleriyle Şekil 45 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 45. DPM11'in simülasyon ortamındaki görünümü

Tablo 38 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 46'de verilmiştir.



Şekil 46. DPM11'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 46 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 39'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 40'de verilmiştir.

Tablo 39. DPM11'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
52949	423592 m ³	-25,66%

Tablo 40. DPM11'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-720mm	-760mm	250mm	-5,26%	5,56%	0,00%
En çok (max)	720mm	680mm	950mm	-5,26%	-15,00%	-2,06%
Aralık (range)	1440mm	1440mm	700mm	-5,26%	-5,26%	-2,78%

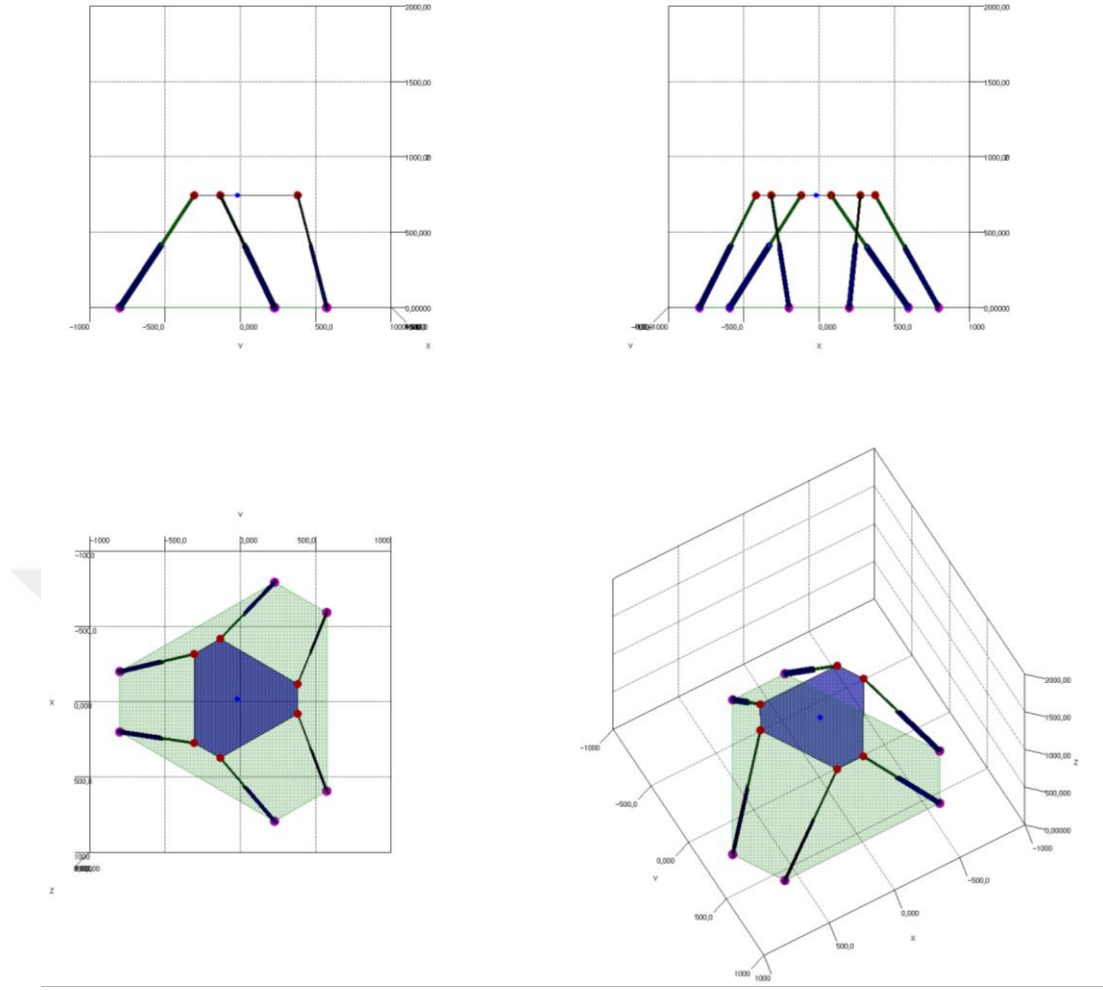
Şekil 22 ile verilen "c" ve "d" parametreleri iki katına çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 12'nin (DPM12) tasarım parametreleri tablo Tablo 41 ile verilmiştir.

Tablo 41. DPM12'nin tasarım parametreleri

a	b	**c	**d	e	f
200mm	400mm	400mm	800mm	500mm	500mm

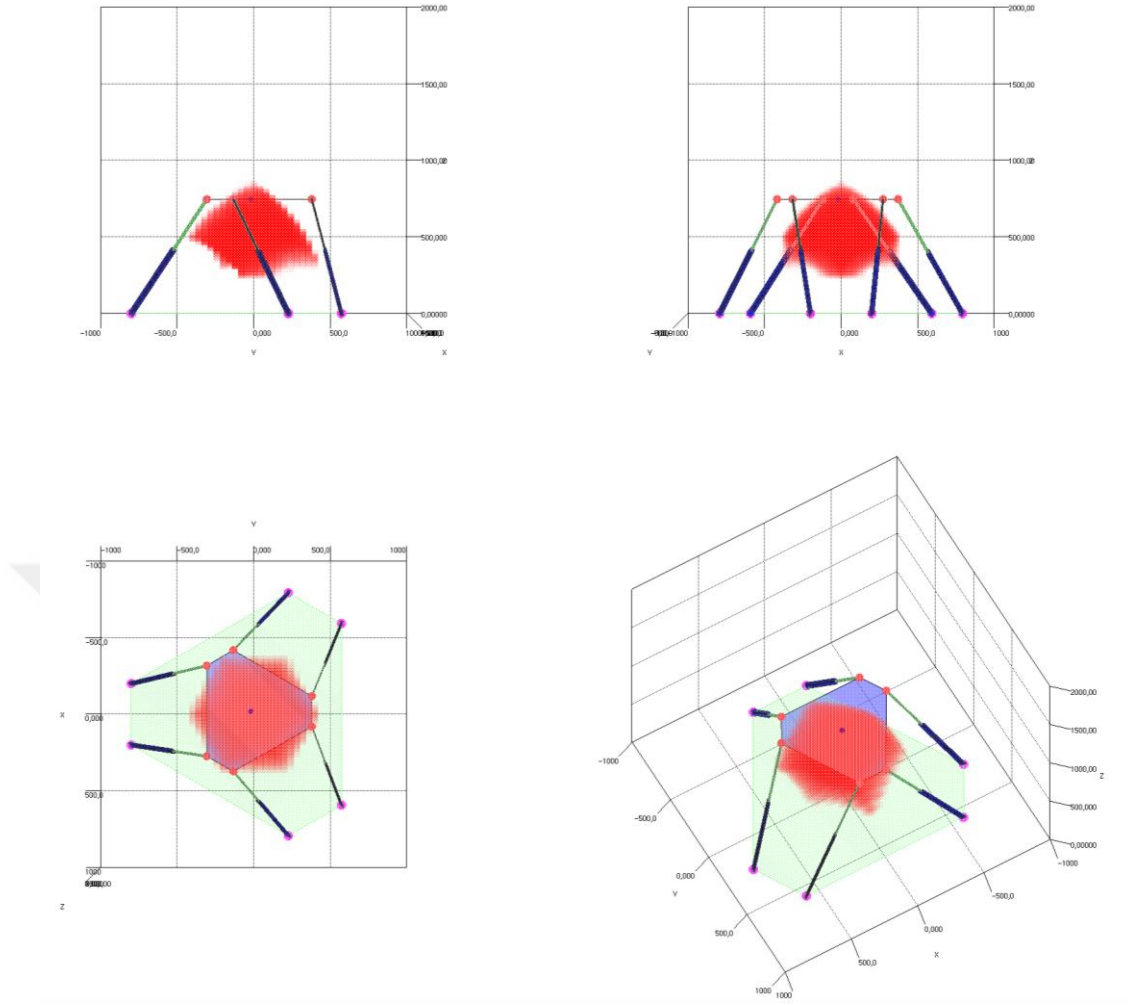
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 41'de verilen tasarım parametreleriyle Şekil 47 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 47. DPM12'nin simülasyon ortamındaki görünümü

Tablo 41 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lık alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 48'de verilmiştir.



Şekil 48.DPM12'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 48 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 42'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 43'de verilmiştir.

Tablo 42. DPM12'nin nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
15097	120776 m ³	-78,79%

Tablo 43. DPM12'nin platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-380mm	-400mm	250mm	-50,04%	-44,44%	0,00%
En çok (max)	360mm	400mm	850mm	-52,62%	-50,00%	-12,37%
Aralık (range)	740mm	800mm	600mm	-51,33%	-47,37%	-16,67%

3.3. Piston Tasarım Parametrelerinin Çalışma Uzayına Etkisi

Paralel manipülatörler, hareket edebilen bileşenleri olan mekanik sistemlerdir. Bu bileşenler, çalışma uzayına göre yapılandırılır ve tasarım parametrelerine göre optimize edilir. Piston tasarım parametreleri, paralel manipülatörlerin performansını ve çalışma uzayını etkileyen önemli faktörlerdir.

Piston tasarım parametreleri, paralel manipülatörlerin hareket kabiliyeti ve verimlilięi üzerinde önemli bir etkiye sahiptir. Örneęin, pistonların çapı, manipülatörlerin yük taşıma kapasitesini belirler. Büyük çaplı pistonlar, daha yüksek yükleri taşıyabilir ancak daha yavaş hareket edebilirler. Küçük çaplı pistonlar ise daha hızlı hareket edebilir ancak daha az yük taşıyabilirler. Bu nedenle, piston tasarım parametreleri, paralel manipülatörlerin görevlerine uygun olarak seçilmelidir. Ayrıca, piston tasarım parametreleri, paralel manipülatörlerin çalışma uzayını da etkiler. Örneęin, uzun ve ince pistonlar, daha uzun mesafeler kat edebilir ancak daha az yük taşıyabilirler. Kısa ve kalın pistonlar ise daha az mesafe kat edebilir ancak daha yüksek yükleri taşıyabilirler. Bu nedenle, paralel manipülatörlerin çalışma uzayı ve görevlerine uygun olarak piston tasarım parametreleri seçilmelidir.

Sonuç olarak, paralel manipülatörlerin performansı ve çalışma uzayı üzerinde önemli bir etkiye sahip olan piston tasarım parametreleri, paralel manipülatörlerin görevlerine ve çalışma uzayına uygun olarak seçilmelidir. Bu sayede, paralel manipülatörlerin verimlilięi ve hareket kabiliyeti optimize edilebilir.

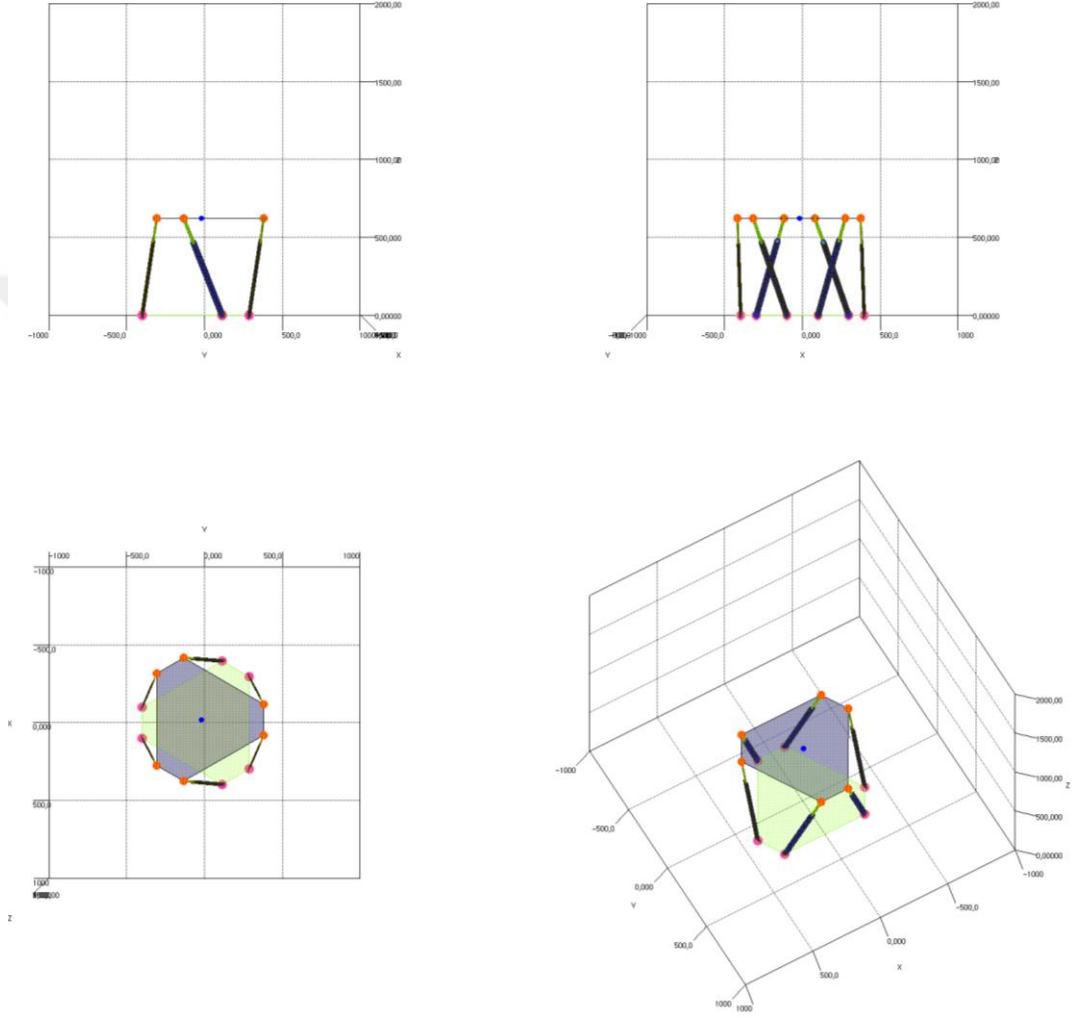
Şekil 22 ile verilen "e" parametresi 500mm'den 250mm'ye düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 13'ün (DPM13) tasarım parametreleri tablo Tablo 44 ile verilmiştir.

Tablo 44. DPM13'ün tasarım parametreleri

a	b	c	d	*e	f
200mm	400mm	200mm	400mm	250mm	500mm

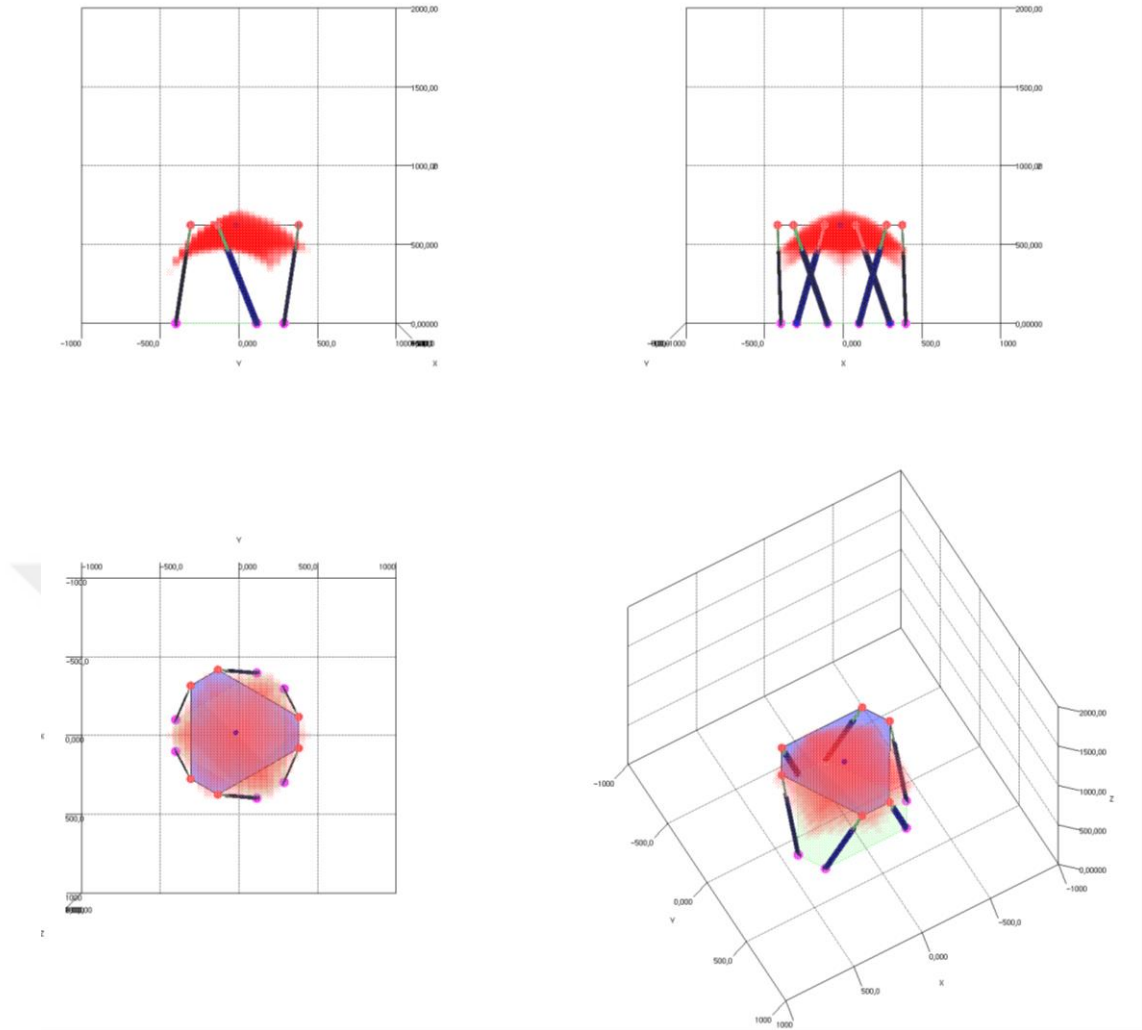
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 44'de verilen tasarım parametreleriyle Şekil 49 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 49. DPM13'ün simülasyon ortamındaki görünümü

Tablo 44 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 1000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 50'de verilmiştir.



Şekil 50. DPM13'ün çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 50 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 45'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 46'de verilmiştir.

Tablo 45. DPM13'ün nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
6542	52336 m ³	-90,81%

Tablo 46. DPM13'ün platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-380mm	-440mm	325mm	-49,99%	-38,89%	30,00%
En çok (max)	380mm	440mm	705mm	-49,99%	-45,00%	-27,32%
Aralık (range)	760mm	880mm	380mm	-49,99%	-42,11%	-47,22%

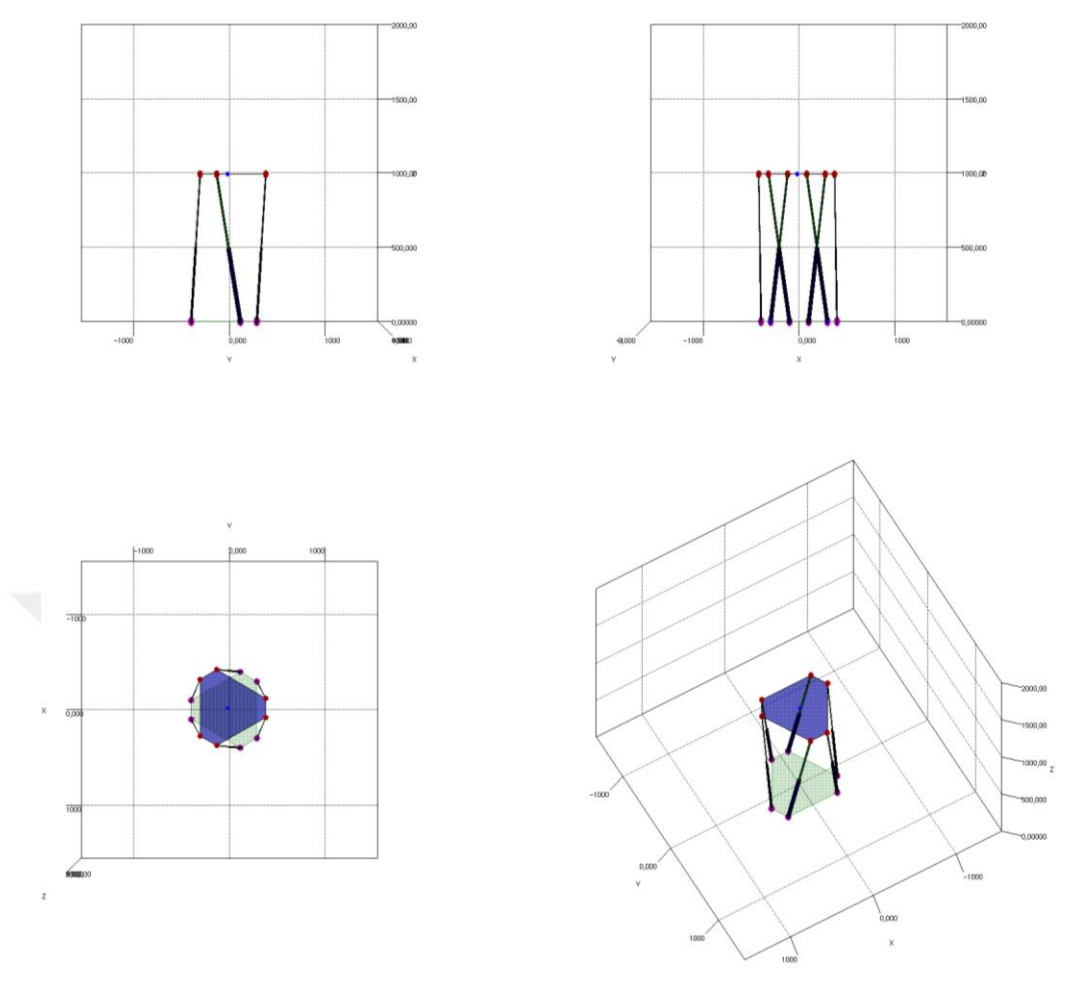
Şekil 22 ile verilen "e" parametresi 500mm'den 1000mm'ye çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 14'ün (DPM14) tasarım parametreleri tablo Tablo 47 ile verilmiştir.

Tablo 47. DPM14'ün tasarım parametreleri

a	b	c	d	**e	f
200mm	400mm	200mm	400mm	1000mm	500mm

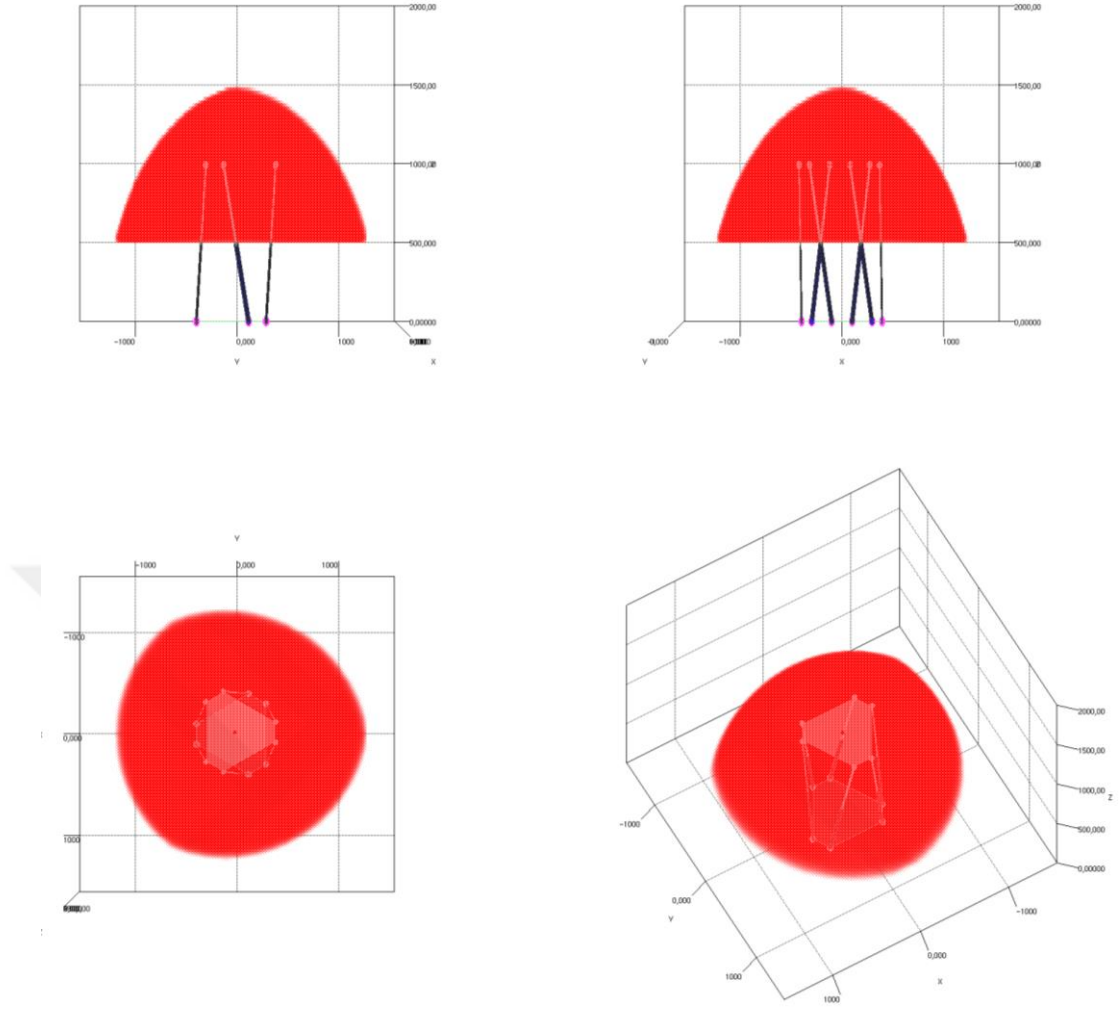
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 47'de verilen tasarım parametreleriyle Şekil 51 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 51. DPM14'ün simülasyon ortamındaki görünümü

Tablo 47 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 2500\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 52'de verilmiştir.



Şekil 52. DPM14'ün çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 52 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 48'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 49'de verilmiştir.

Tablo 48. DPM14'ün nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
183649	1469192 m ³	157,84%

Tablo 49. DPM14'ün platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-1210mm	-1180mm	520mm	59,16%	63,89%	108,00%
En çok (max)	1210mm	1240mm	1480mm	59,17%	55,00%	52,58%
Aralık (range)	2420mm	2420mm	960mm	59,17%	59,21%	33,33%

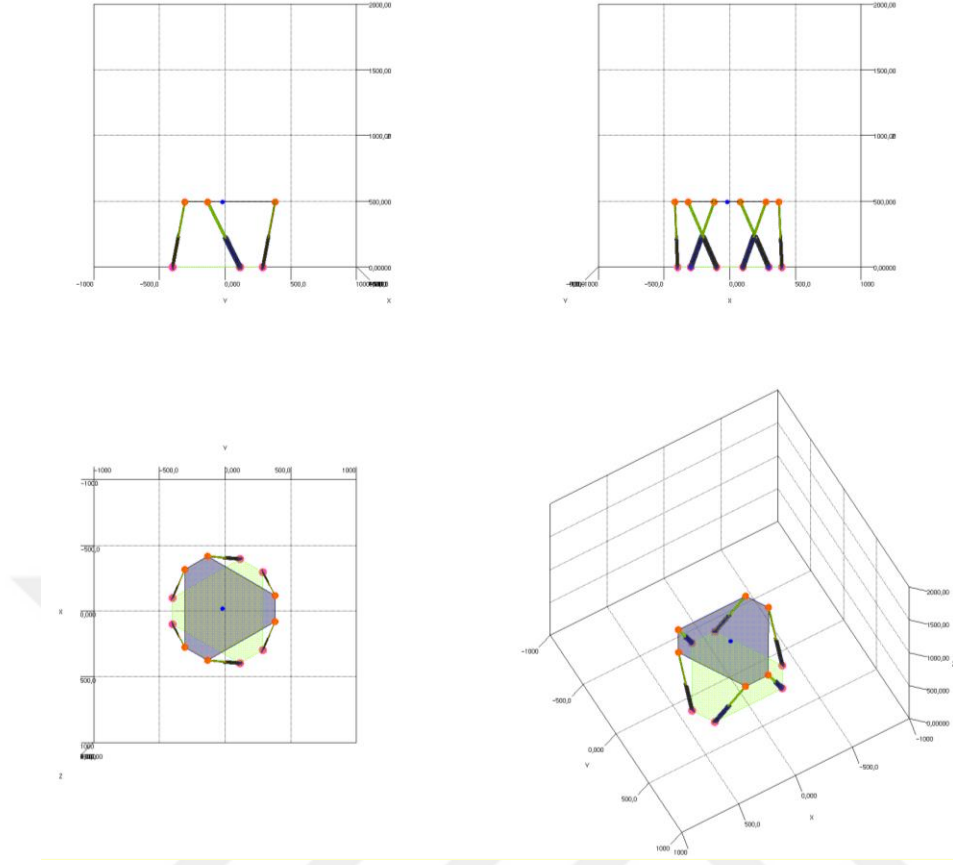
Şekil 22 ile verilen "f" parametresi 500mm'den 250mm'ye düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 15'in (DPM15) tasarım parametreleri tablo Tablo 50 ile verilmiştir.

Tablo 50. DPM15'in tasarım parametreleri

a	b	c	d	e	*f
200mm	400mm	200mm	400mm	500mm	250mm

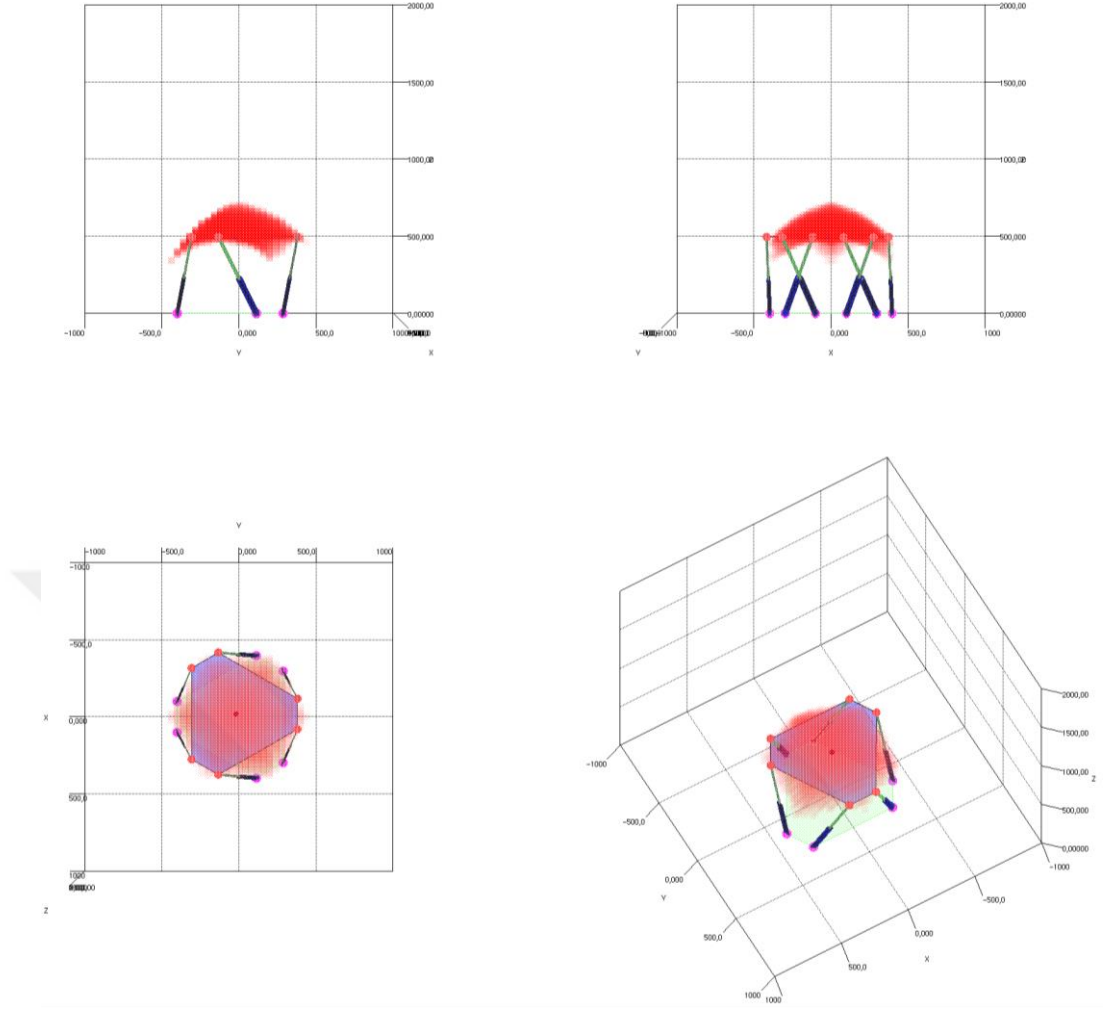
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 50'de verilen tasarım parametreleriyle Şekil 53 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 53. DPM15'in simülasyon ortamındaki görünümü

Tablo 50 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde ± 2500 mm 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 54'de verilmiştir.



Şekil 54. DPM15'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 54 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 51 'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 52 'de verilmiştir.

Tablo 51. DPM15'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
6636	53088 m ³	-90,68%

Tablo 52. DPM15'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-380mm	-440mm	340mm	-49,99%	-38,89%	36,00%
En çok (max)	380mm	440mm	700mm	-49,99%	-45,00%	-27,84%
Aralık (range)	760mm	880mm	360mm	-49,99%	-42,11%	-50,00%

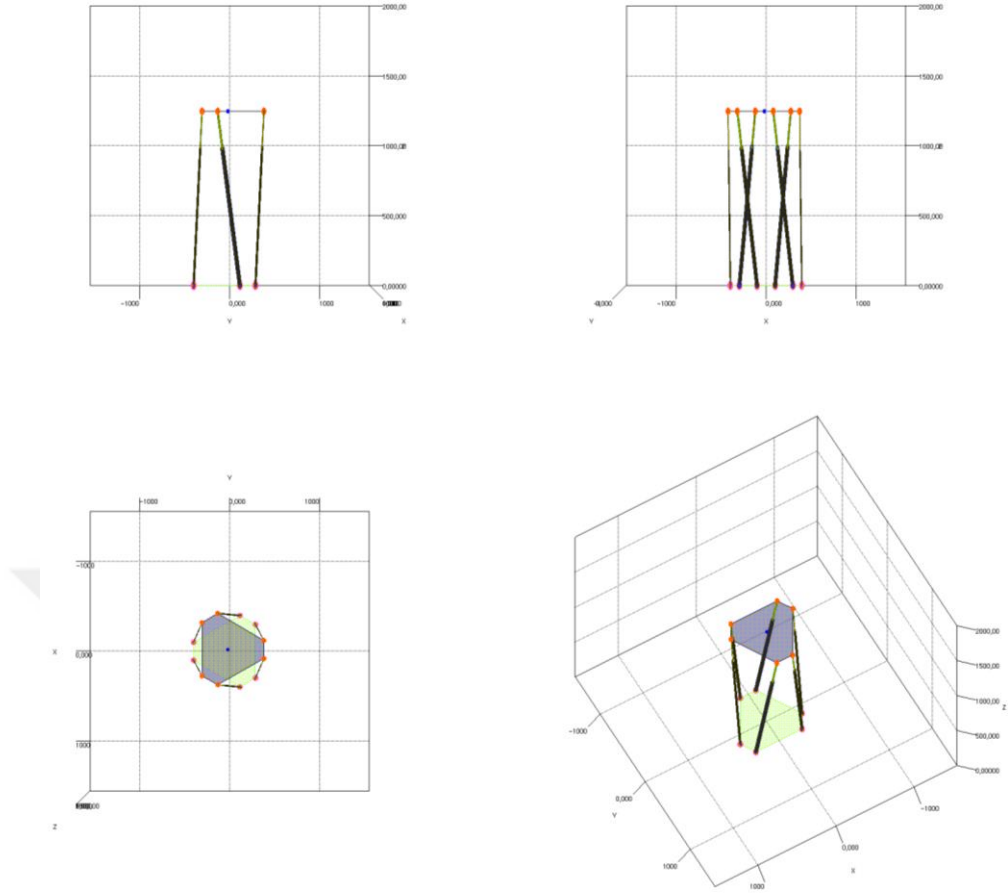
Şekil 22 ile verilen "f" parametresi 500mm'den 1000mm'ye çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 16'nın (DPM16) tasarım parametreleri tablo Tablo 53 ile verilmiştir.

Tablo 53. DPM16'nın tasarım parametreleri

a	b	c	d	e	**f
200mm	400mm	200mm	400mm	500mm	1000mm

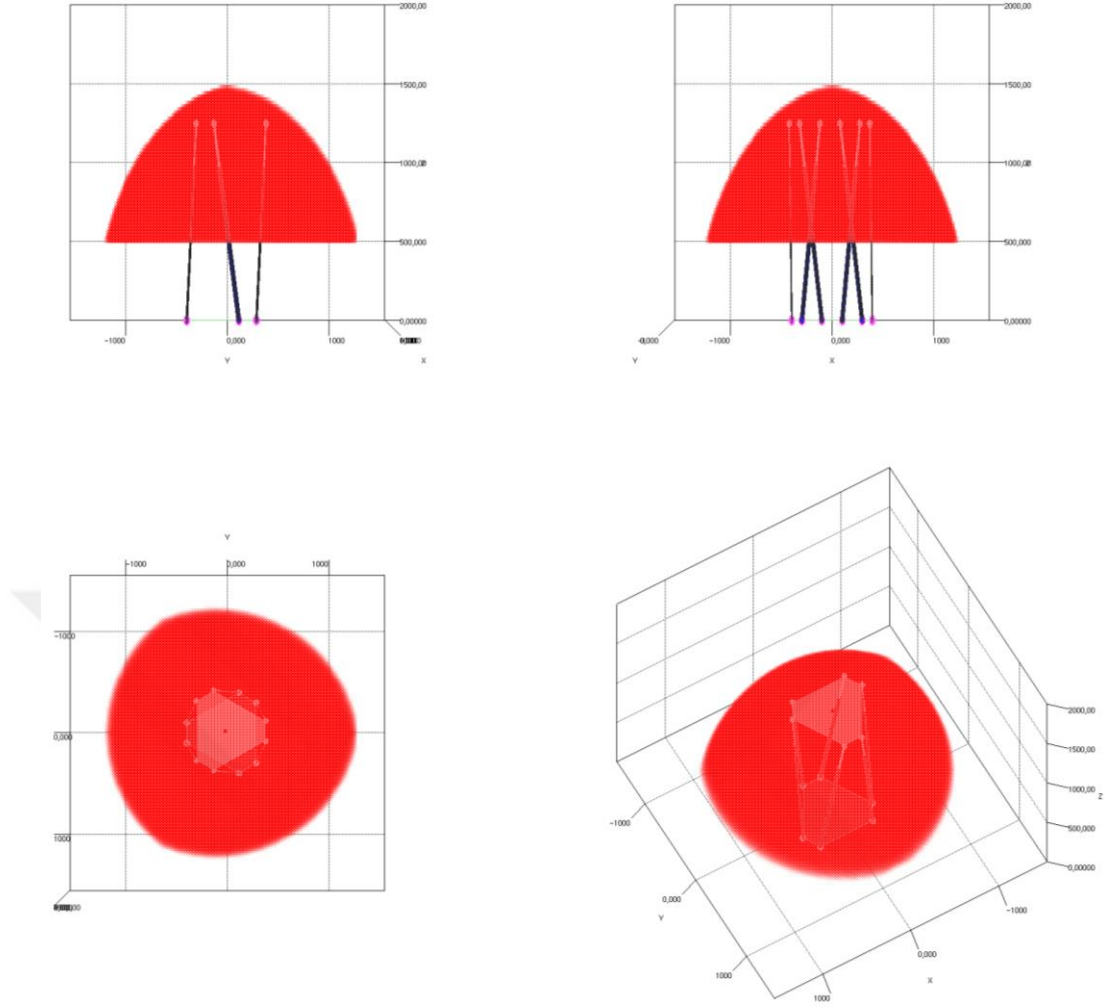
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 53'de verilen tasarım parametreleriyle Şekil 55 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 55. DPM16'nın simülasyon ortamındaki görünümü

Tablo 53 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 2500\text{mm}$ 'lık alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 56'de verilmiştir.



Şekil 56. DPM16'nın çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 56 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 54'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 55'de verilmiştir.

Tablo 54. DPM16'nın nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
185249	1481992 m ³	160,09%

Tablo 55. DPM16'nın platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-1210mm	-1180mm	510mm	59,17%	63,89%	104,00%
En çok (max)	1210mm	1240mm	1470mm	59,17%	55,00%	51,55%
Aralık (range)	2420mm	2420mm	960mm	59,17%	59,21%	33,33%

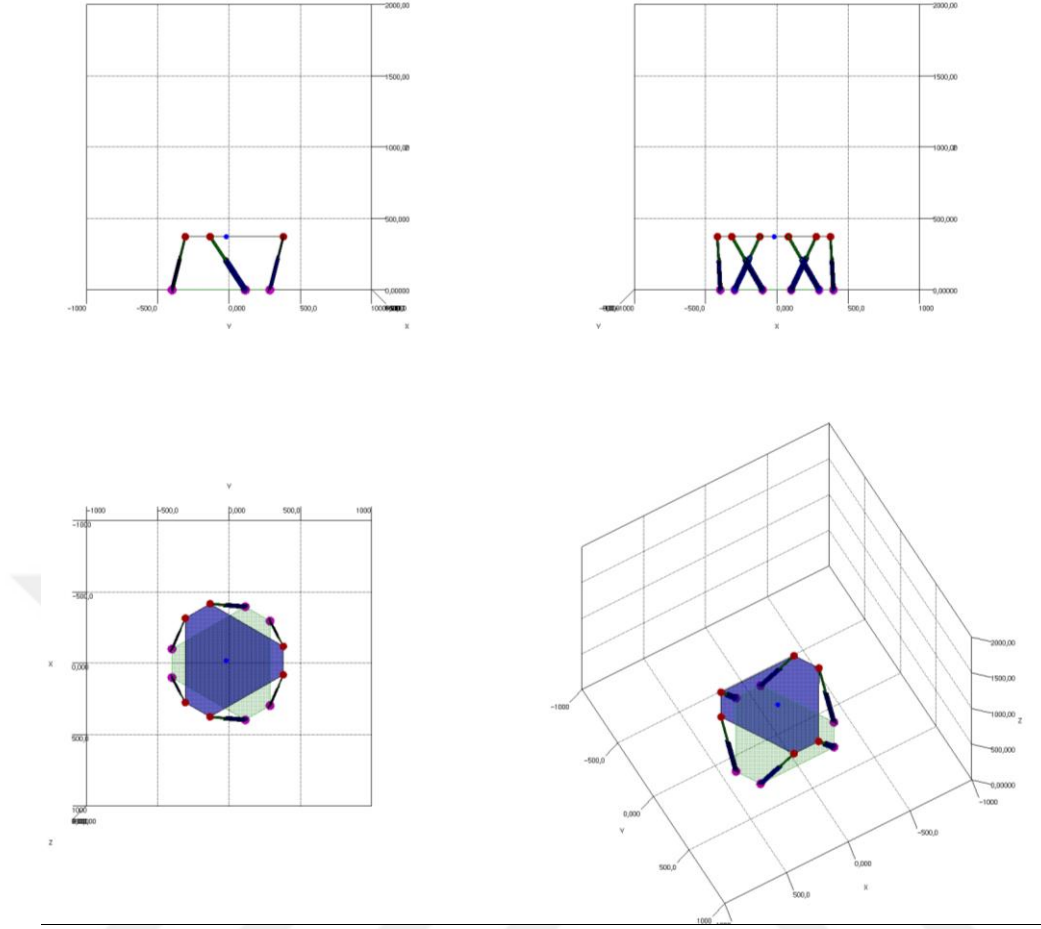
Şekil 22 ile verilen "e" ve "f" parametreleri yarı yarıya düşürüldüğünde (1/2) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 17'nin (DPM17) tasarım parametreleri tablo Tablo 56 ile verilmiştir.

Tablo 56. DPM17'nin tasarım parametreleri

a	b	c	d	*e	*f
200mm	400mm	200mm	400mm	250mm	250mm

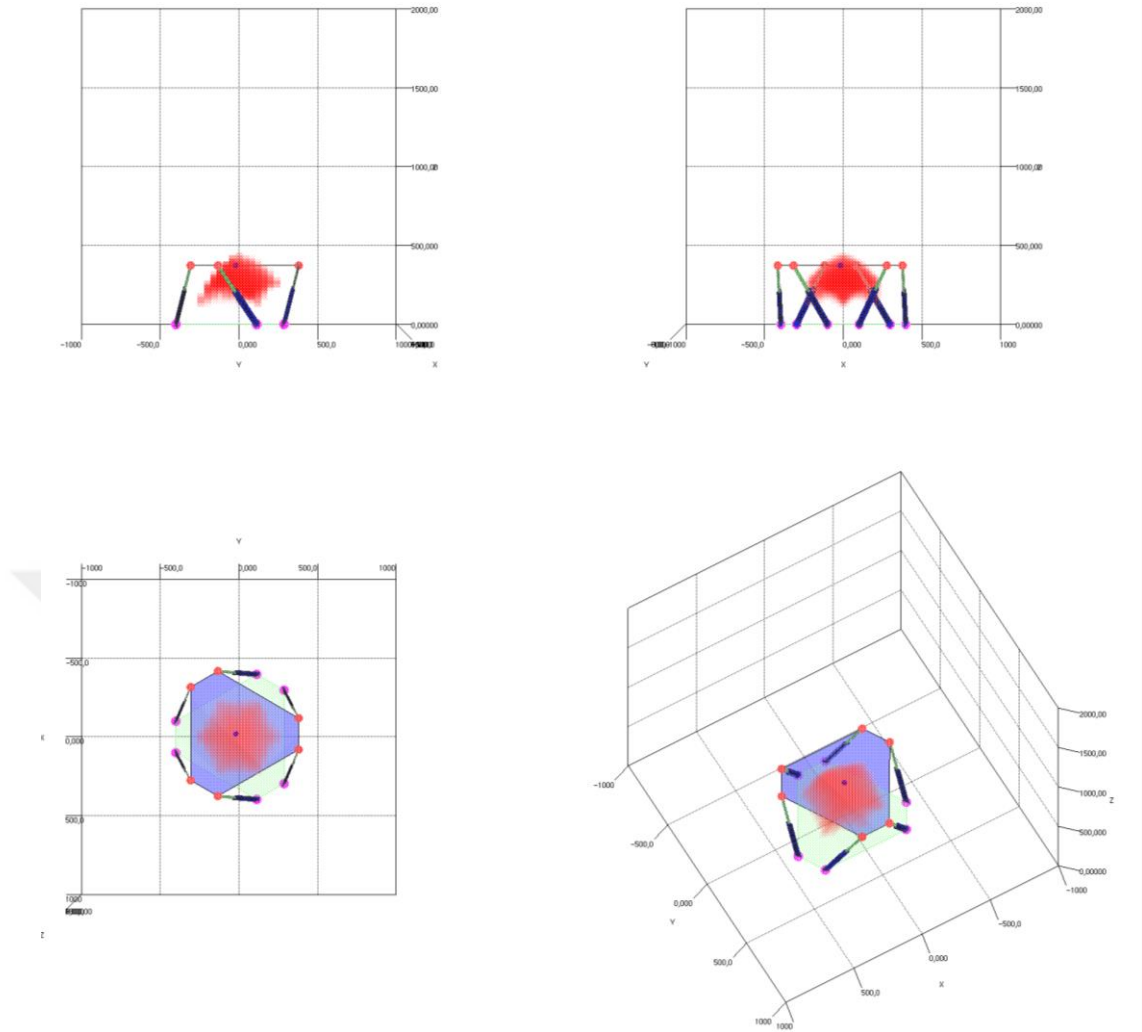
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 56'de verilen tasarım parametreleriyle Şekil 57 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 57. DPM17'nin simülasyon ortamındaki görünümü

Tablo 56 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 2000\text{mm}$ 'lık alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 58'de verilmiştir.



Şekil 58. DPM17'nin çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 58 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 57'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 58'de verilmiştir.

Tablo 57. DPM17'nin nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre değişim Oranı
2762	22096 m ³	-96,12%

Tablo 58. DPM17'nin platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-220mm	-240mm	135mm	-71,03%	-66,67%	-46,00%
En çok (max)	220mm	240mm	435mm	-71,03%	-70,00%	-55,15%
Aralık (range)	440mm	480mm	300mm	-71,03%	-68,42%	-58,33%

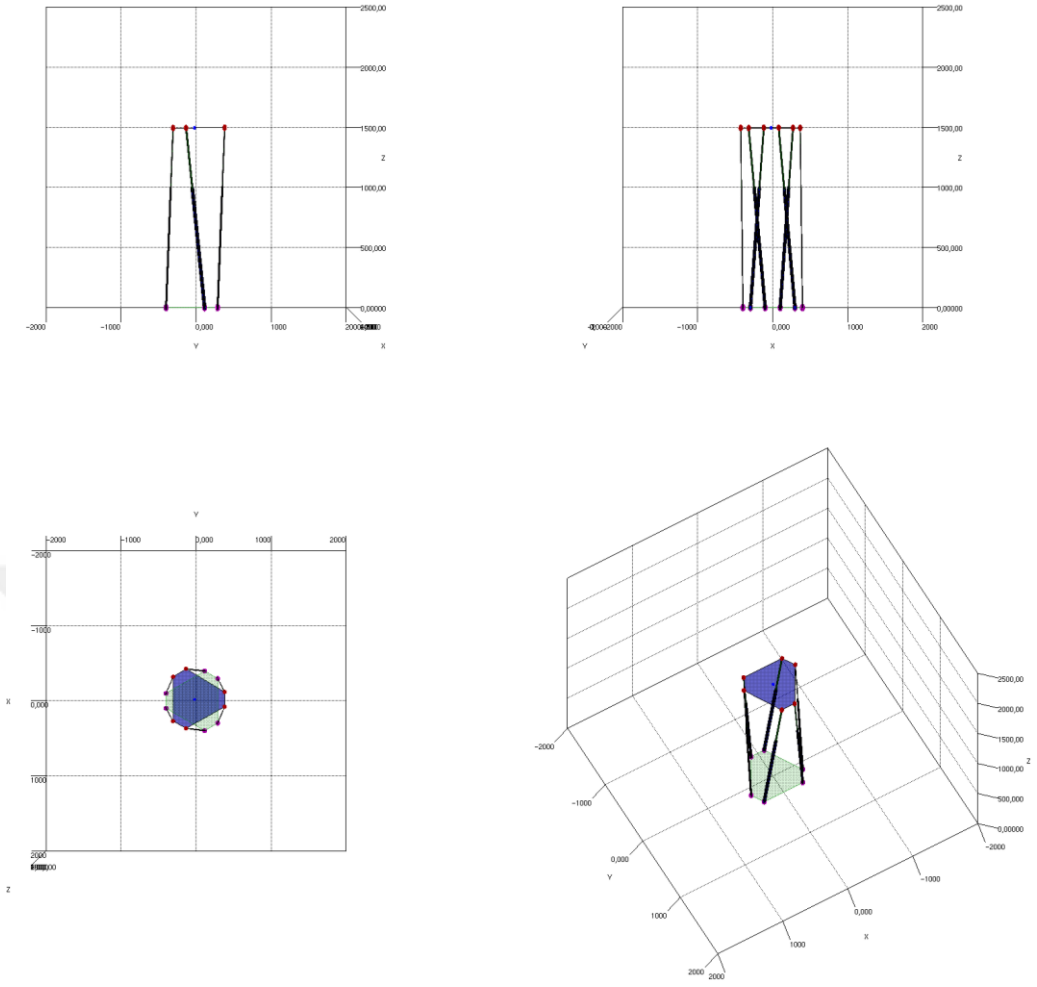
Şekil 22 ile verilen "e" ve "f" parametreleri iki katına çıkarıldığında (2x) çalışma uzayına etkisi incelenmiştir. Deneysel platform modeli 18'in (DPM18) tasarım parametreleri tablo Tablo 59 ile verilmiştir.

Tablo 59. DPM18'in tasarım parametreleri

a	b	c	d	**e	**f
200mm	400mm	200mm	400mm	1000mm	1000mm

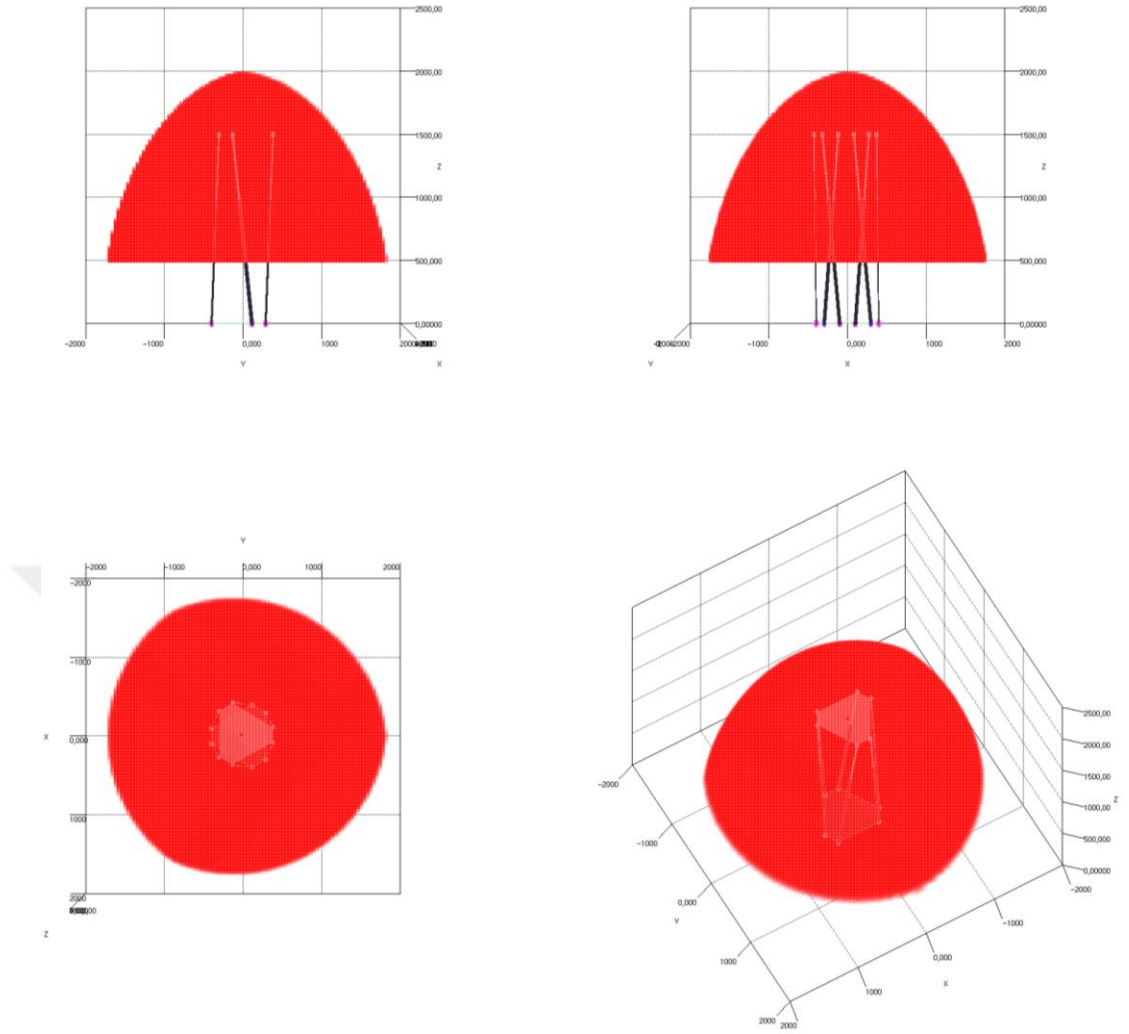
* ARPM'nin yarısı ** ARPM'nin iki katı

Tablo 59'de verilen tasarım parametreleriyle Şekil 59 de gösterilen 3D platform tasarımı modeli oluşmaktadır.



Şekil 59. DPM18'in simülasyon ortamındaki görünümü

Tablo 59 parametreleri ile oluşturulan platformun 20mm nokta aralıklarıyla platform merkezine göre X, Y ve Z eksenlerinde $\pm 4000\text{mm}$ 'lik alan taranarak nokta bulutu metodu ile çalışma uzayı analizi Şekil 60'de verilmiştir.



Şekil 60. DPM18'in çalışma uzayı analizi sonucu simülasyon ortamındaki nokta bulutu görünümü

Şekil 60 'de gösterilen çalışma uzayı analizi sonucuna göre nokta bulutu hacmi Tablo 60'de, platformun merkezine göre eksenlerdeki sistem sınırları ise Tablo 61'de verilmiştir.

Tablo 60. DPM18'in nokta bulutu hacmi, nokta sayısı ve ARPM'ye göre değişim oranı

Nokta Sayısı	Hacim	ARPM'e göre Değişim Oranı
890366	7122928 m ³	1150,07%

Tablo 61. DPM18'in platform merkezine göre X,Y ve Z eksenlerindeki sistem sınırları ve ARPM'ye göre deęişim oranı

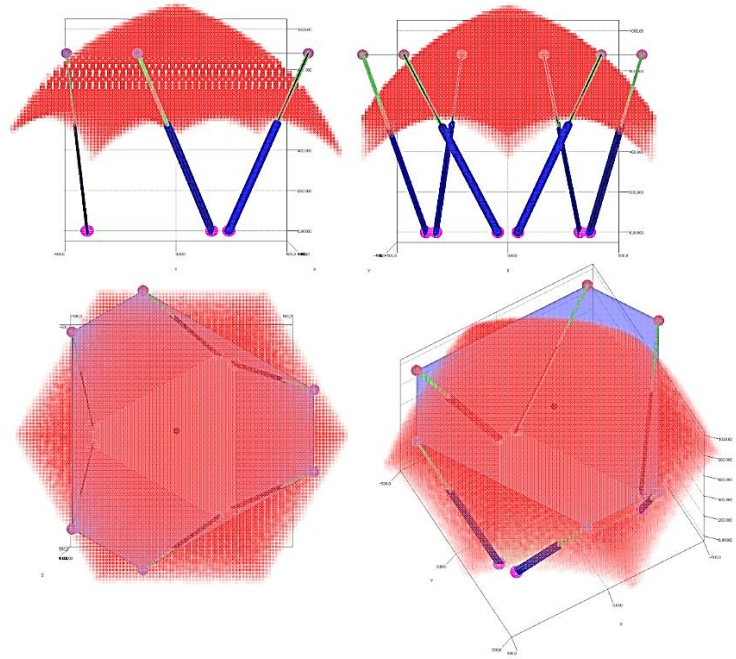
Veri Adı	X Eksenindeki Sınırlar	Y Eksenindeki Sınırlar	Z Eksenindeki Sınırlar	X Deęişim Oranı	Y Deęişim Oranı	Z Deęişim Oranı
En az (min)	-1720mm	-1680mm	510mm	126,22%	133,33%	103,86%
En çok (max)	1721mm	1800mm	1970mm	126,32%	125,00%	103,09%
Aralık (range)	3440mm	3480mm	1460mm	126,27%	128,95%	102,83%

Ana Referans Platform Modeli'nde (ARPM) belirtilen a, b, c, d, e, f parametrelerindeki deęerler bu çalışmada örnekleme oluşturmak için tercih edilmiştir. Bu deęerler tercih edilirken her bir a, b, c, d, e, f parametresinin ayrı ayrı yarısı ve iki katı alındığında platformun gerçekleştirilebilir bir örnek mekanizma oluşturabilmesinin mümkün oluşu göz önünde bulundurulmuştur.

SONUÇLAR VE ÖNERİLER

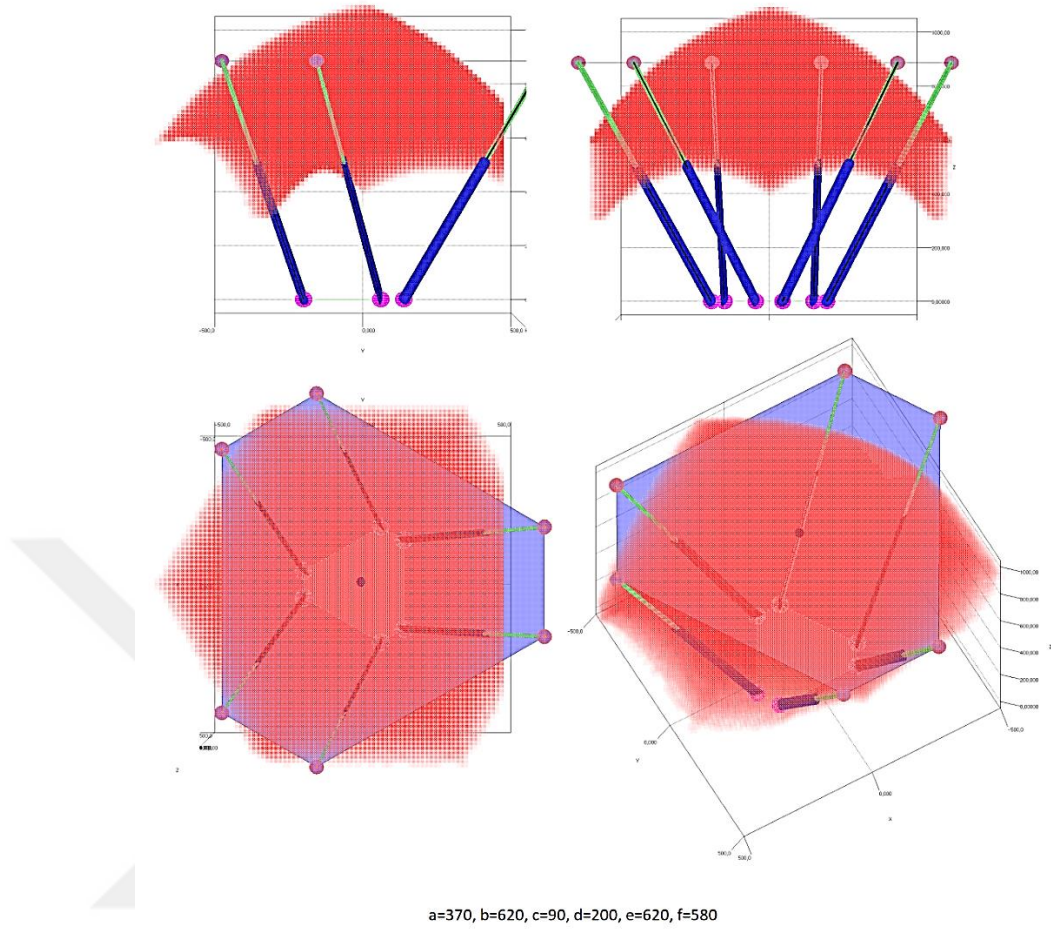
Bu çalışmada, Stewart Platform Mekanizmasının tip sentezleme yöntemiyle oluşturulan 6x3 bağlantı tipinden yararlanılarak altı serbestlik derecesine sahip hali geliştirilmiştir. Bu modelin kinematik hesaplamaları yapılırken, ters kinematik yöntemler kullanılarak platformun kontrol hesaplamaları yapılmıştır. Bu sayede, ters kinematik hesaplamaların iş uğraşı ve karmaşıklığının ileri kinematik yöntemle yapılan çözümlere göre daha az olması ve daha kısa sürede hesaplama yapma olanağı sunmaktadır. İleri kinematiğe tercihen ters kinematik hesaplama yaparak işlemleri gerçekleştirmek ve uygulamak fayda sağlamıştır.

Tip sentezleme süreci sonucunda ortaya çıkan platform mekanizması, Java ortamında ters kinematik hesaplamalar kullanılarak çözüm kümesine ulaşılmıştır. Bu sistemde, yazılımsal olarak kinematikler incelenmiş ve platform için en uygun hareket modeli belirlenmiştir. Örnek bir çalışma uzayı analizi Şekil 22 ve Şekil 23'te gösterilmektedir.



(a=370, b=620, c=90, d=400, e=620, f=580)

Şekil 61. Örnek çalışma uzayı çıktısı (d=400 için, nokta bulutu aralığı 15 mm)



Şekil 62. Örnek çalışma uzayı çıktısı (d=200 için, nokta bulutu aralığı 15 mm)

Platformların uzaysal hareket edebilme yetenekleri nedeniyle, çalışmalar hem gerçek ortamda hem de sanal ortamda yapılabilir. Bilgisayar yazılımları ile benzetimler yapılarak tasarımı yapılan bir Stewart platformu, gerçek hayatta uygulandığında daha hızlı sonuçlar alınabilir. Benzetimi yapılan bu sistemde bağlantı kollarının uzunluğu değiştirilebilir. Bu benzetimi yapılan sistemde, sistemin çalışma uzayı ile ilgili analizler yapıldı. Yapılan bu analizler sonucu parametreleri standart olan SPM'na göre daha iyi bir çalışma sağlayan sistemin oluşturulması sağlanmaktadır. Sistemin bütünlüğü bakımından incelendiğinde, yazılımsal prototipi yapılan model hızlı ve güçlü bir yapıya sahiptir. Sistemdeki doğrusal eyleyiciler raylı model üzerinde hareket sağladığı için güçlü bir yapıdadır. Doğrusal hareketin gerçek prototipte vidalı sistem yerine kayışlı veya kremayer/pinyon sistemi kullanılırsa, daha az sürtünme daha çok hız alınabilir. Hızlı çalıştırılmak istenilen bu sistemlerde hassasiyetini

engelleyecek yapıda zayıf bir yapıya sahip olması dikkat edilmesi gereken bir husustur.

Çalışmada gerçekleştirilen Ana Referans Platform Modeli ve Deneysel Platform Modelleri ile ilgili çalışma uzayı analizi yapılmıştır.

Şekil 22’de gösterilen parametrelerin değişik kombinasyonlarına göre platform merkezinin çalışma uzayındaki ulaşabildiği alanlar hesaplanmıştır. Çalışma uzayı içerisindeki hacimsel ve eksenel sınır bilgileri ile ilgili Tablo 62 ile gösterilen sonuçlar elde edilmiştir.

Tablo 62 tablosundaki her bir DPM’nin ARPM’ye göre hangi platform tasarım parametresine ne kadarlık bir güncelleme yapıldığı ve bunun sonucunda eksenel ve hacim parametrelerinin ARPM’ye göre birim olarak ne kadarlık değişime uğradığı Tablo 63 ile gösterilmiştir.

Tablo 62 tablosundaki her bir DPM’nin ARPM’ye göre hangi platform tasarım parametresine ne kadarlık bir güncelleme yapıldığı ve bunun sonucunda eksenel ve hacim parametrelerinin ARPM’ye göre orantısal olarak ne kadarlık değişime uğradığı Tablo 64 ile gösterilmiştir.

ARPM’de verilen $a=200$, $b=400$, $c=200$, $d=400$, $e=500$, $f=500$ tasarım parametre değerleri DPM1’de a parametresi yarıya indirilerek $a=100$, $b=400$, $c=200$, $d=400$, $e=500$, $f=500$ değerleri ele alınarak çalışma uzayının hacmi ve eksen sınırlarının yanı sıra hacmi belirten nokta bulutu analizindeki nokta sayısını ve ne kadar bir hacim oluştuğunu bize vermektedir. $a=100$, $b=400$, $c=200$, $d=400$, $e=500$, $f=500$ ele alınırken aldığımız nokta sayısı çıktısı 52996, çalışma uzayının hacmi(m^3), 423968 ve x , y , z eksenlerindeki sınırlar belirtilmiştir. Yine aynı şekilde DPM2’de a parametresi iki katına çıkarılarak $a=400$, $b=400$, $c=200$, $d=400$, $e=500$, $f=500$ değerleri ele alınarak çalışma uzayının hacmi ve eksen sınırlarının yanı sıra hacmi belirten nokta bulutu analizindeki nokta sayısını ve ne kadar bir hacim

oluşturduğunu bize vermektedir. $a=400$, $b=400$, $c=200$, $d=400$, $e=500$, $f=500$ ele alınırken aldığımız nokta sayısı çıktısı 916673 nokta, çalışma uzayının hacmi(m^3)=733384 ve x, y, z eksenlerindeki sınırlar belirtilmiştir. Bu durum diğer parametreler içinde yarı ve iki katı alınarak çalışma alanı hacmi, nokta sayısı, çalışma uzayı ve x, y, z eksenlerindeki çalışma sınırlarının sonuçları ele alınmaktadır.

ARPM'de verilen $a=200$, $b=400$, $c=200$, $d=400$, $e=500$, $f=500$ tasarım parametre değerleri DPM18'de e ve f parametrelerinin iki katı alındığında yani e parametresi 1000 ve f parametresi 1000 olduğunda oluşturulan nokta bulutu hacmi Tablo 64'te görüldüğü gibi 1150,08% şeklinde artış göstermiştir. ARPM'de verilen $a=200$, $b=400$, $c=200$, $d=400$, $e=500$, $f=500$ tasarım parametre değerleri DPM17'de e ve f parametreleri yarıya indirildiğinde yani e parametresi 250 ve f parametresi 250 olarak ele alındığında nokta bulutu analiz hacmi -96,12% olarak değişmektedir. Buradan anlaşıldığı üzere çalışma uzayımızın geniş bir alana sahip olmasını istediğimiz bir platform oluşturmamız gerekiyorsa bu duruma en çok etki eden parametrelerin değişimini Tablo 62, Tablo 63 ve Tablo 64'te ki gibi analiz sonuçlarına erişebilmekteyiz.

Tablo 62. ARPM ve DPM için çalışma uzayının hacim ve eksen sınırları

Güncellenen Tasarım Parametreleri	Konfigürasyon Adı	Tasarım Parametreleri (Girdi)						Çalışma Uzayı Hacim ve Eksen Sınırları (Çıktı)										
		a	b	c	d	e	f	Nokta Sayısı	Hacim (m ³)	X Eksenindeki Sınırlar			Y Eksenindeki Sınırlar			Z Eksenindeki Sınırlar		
										min (mm)	max (mm)	range (mm)	min (mm)	max (mm)	range (mm)	min (mm)	max (mm)	range (mm)
	ARPM	200	400	200	400	500	500	71225	569800	-760	760	1520	-720	800	1520	250	970	720
PLATFORM	DPM1	100	400	200	400	500	500	52996	423968	-720	720	1440	-680	760	1440	250	950	700
	DPM2	400	400	200	400	500	500	91673	733384	-820	820	1640	-800	840	1640	250	970	720
	DPM3	200	200	200	400	500	500	71525	572200	-760	760	1520	-800	720	1520	250	970	720
	DPM4	200	800	200	400	500	500	12727	101816	-340	340	680	-360	360	720	250	830	580
	DPM5	100	200	200	400	500	500	52881	423048	-720	720	1440	-760	680	1440	250	950	700
	DPM6	400	800	200	400	500	500	15093	120744	-360	380	740	-400	400	800	250	830	580
ZEMİN	DPM7	200	400	100	400	500	500	53012	424096	-720	720	1440	-680	760	1440	250	950	700
	DPM8	200	400	400	400	500	500	91603	732824	-820	820	1640	-800	840	1640	250	970	720
	DPM9	200	400	200	200	500	500	71594	572752	-760	760	1520	-800	720	1520	250	970	720
	DPM10	200	400	200	800	500	500	12718	101744	-340	340	680	-360	360	720	250	830	580
	DPM11	200	400	100	200	500	500	52949	423592	-720	720	1440	-760	680	1440	250	950	700
	DPM12	200	400	400	800	500	500	15097	120776	-380	360	740	-400	400	800	250	850	600
PISTON	DPM13	200	400	200	400	250	500	6542	52336	-380	380	760	-440	440	880	325	705	380
	DPM14	200	400	200	400	1000	500	183649	1469192	-1210	1210	2420	-1180	1240	2420	520	1480	960
	DPM15	200	400	200	400	500	250	6636	53088	-380	380	760	-440	440	880	340	700	360
	DPM16	200	400	200	400	500	1000	185249	1481992	-1210	1210	2420	-1180	1240	2420	510	1470	960
	DPM17	200	400	200	400	250	250	2762	22096	-220	220	440	-240	240	480	135	435	300
	DPM18	200	400	200	400	1000	1000	890366	7122928	-1720	1721	3440	-1680	1800	3480	510	1970	1460

Tablo 63. DPM'lerin ARPM'ye göre çalışma uzayı farklarının orantısal gösterimi

Güncellenen Tasarım Parametreleri	Konfigürasyon Adı	Tasarım Parametrelerinin Farkı (Girdi)						Çalışma Uzayı Hacmi Farkı (Çıktı)										
		a	b	c	d	e	f	Nokta Sayısı Farkı	Hacim Farkı (m³)	X Eksenindeki Fark			Y Eksenindeki Fark			Z Eksenindeki Fark		
		min (mm)	max (mm)	range (mm)	min (mm)	max (mm)	range (mm)			min (mm)	max (mm)	range (mm)						
	ARPM	200	400	200	400	500	500	71225	569800	-760	760	1520	-720	800	1520	250	970	720
PLATFORM	DPM1	-100	-	-	-	-	-	-18229	-145832	+40	-40	-80	+40	-40	-80	-	-20	-20
	DPM2	+200	-	-	-	-	-	+20448	+163584	-60	+60	+120	-80	+40	+120	-	-	-
	DPM3	-	-200	-	-	-	-	+300	+2400	-	-	-	-80	-80	-	-	-	-
	DPM4	-	+400	-	-	-	-	-58498	-467984	+420	-420	-840	+360	-440	-800	-	-140	-140
	DPM5	-100	-200	-	-	-	-	-18344	-146752	+40	-40	-80	-40	-120	-80	-	-20	-20
	DPM6	+200	+400	-	-	-	-	-56132	-449056	+400	-380	-780	+320	-400	-720	-	-140	-140
ZEMİN	DPM7	-	-	-100	-	-	-	-18213	-145704	+40	-40	-80	+40	-40	-80	-	-20	-20
	DPM8	-	-	+200	-	-	-	+20378	+163024	-60	+60	+120	-80	+40	+120	-	-	-
	DPM9	-	-	-	-200	-	-	+369	+2952	-	-	-	-80	-80	-	-	-	-
	DPM10	-	-	-	+400	-	-	-58507	-468056	+420	-420	-840	+360	-440	-800	-	-140	-140
	DPM11	-	-	-100	-200	-	-	-18276	-146208	+40	-40	-80	-40	-120	-80	-	-20	-20
	DPM12	-	-	+200	+400	-	-	-56128	-449024	+380	-400	-780	+320	-400	-720	-	-120	-120
PISTON	DPM13	-	-	-	-	-250	-	-64683	-517464	+380	-380	-760	+280	-360	-640	+75	-265	-340
	DPM14	-	-	-	-	+500	-	+112424	+899392	-450	+450	+900	-460	+440	+900	+270	+510	+240
	DPM15	-	-	-	-	-	-250	-64589	-516712	+380	-380	-760	+280	-360	-640	+90	-270	-360
	DPM16	-	-	-	-	-	+500	+114024	+912192	-450	+450	+900	-460	+440	+900	+260	+500	+240
	DPM17	-	-	-	-	-250	-250	-68463	-547704	+540	-540	-1080	+480	-560	-1040	-115	-535	-420
	DPM18	-	-	-	-	+500	+500	+819141	+6553128	-960	+961	+1920	-960	+1000	+1960	+260	+1000	+740

Tablo 64. DPM'lerin ARPM'ye göre çalışma uzayı farklarının orantısal gösterimi

Güncellenen	Konfigürasyon Adı	Tasarım Parametreleri (Girdi)						DPM Çalışma Uzayı Hacminin ARPM'ye Göre Değişim Oranı (Çıktı)										
		a	b	c	d	e	f	Hacim Değişim Oranı (%)	X Ekseninde Değişim Oranı			Y Ekseninde Değişim Oranı			Z Ekseninde Değişim Oranı			
									min (%)	max (%)	range (%)	min (%)	max (%)	range (%)	min (%)	max (%)	range (%)	
	ARPM	200	400	200	400	500	500	-	-	-	-	-	-	-	-	-	-	-
PLATFORM	DPM1	100	400	200	400	500	500	-25,59%	-5,26%	-5,26%	-5,26%	-5,56%	-5%	-5,26%	-	-2,06%	-2,78%	
	DPM2	400	400	200	400	500	500	28,71%	7,89%	7,89%	7,89%	11,11%	5%	7,89%	-	-	-	
	DPM3	200	200	200	400	500	500	0,42%	-	-	-	11,11%	-10%	-	-	-	-	
	DPM4	200	800	200	400	500	500	-82,13%	-55,25%	-55,25%	-55,25%	-50%	-55%	-52,63%	-	-14,43%	-19,44%	
	DPM5	100	200	200	400	500	500	-25,76%	-5,26%	-5,26%	-5,26%	5,56%	-15%	-5,26%	-	-2,06%	-2,78%	
	DPM6	400	800	200	400	500	500	-78,81%	-52,62%	-49,99%	-51,3%	-44,44%	-50%	-47,37%	-	-14,43%	-19,44%	
ZEMİN	DPM7	200	400	100	400	500	500	-25,57%	-5,26%	-5,26%	-5,26%	-5,56%	-5%	-5,26%	-	-2,06%	-2,78%	
	DPM8	200	400	400	400	500	500	28,61%	7,89%	7,89%	7,89%	11,11%	5%	7,89%	-	-	-	
	DPM9	200	400	200	200	500	500	0,52%	-	-	-	11,11%	-10%	-	-	-	-	
	DPM10	200	400	200	800	500	500	-82,14%	-55,25%	-55,25%	-55,25%	-50%	-55%	-52,63%	-	-14,43%	-19,44%	
	DPM11	200	400	100	200	500	500	-25,66%	-5,26%	-5,26%	-5,26%	5,56%	-15%	-5,26%	-	-2,06%	-2,78%	
	DPM12	200	400	400	800	500	500	-78,8%	-50,04%	-52,62%	-51,33%	-44,44%	-50%	-47,37%	-	-12,37%	-16,67%	
PISTON	DPM13	200	400	200	400	250	500	-90,82%	-49,99%	-49,99%	-49,99%	-38,89%	-45%	-42,11%	30%	-27,32%	-47,22%	
	DPM14	200	400	200	400	1000	500	157,84%	59,16%	59,17%	59,17%	63,89%	55%	59,21%	108%	52,58%	33,33%	
	DPM15	200	400	200	400	500	250	-90,68%	-49,99%	-49,99%	-49,99%	-38,89%	-45%	-42,11%	36%	-27,84%	-50%	
	DPM16	200	400	200	400	500	1000	160,09%	59,17%	59,17%	59,17%	63,89%	55%	59,21%	104%	51,55%	33,33%	
	DPM17	200	400	200	400	250	250	-96,12%	-71,03%	-71,03%	-71,03%	-66,67%	-70%	-68,42%	-46%	-55,15%	-58,33%	
	DPM18	200	400	200	400	1000	1000	1150,08%	126,22%	126,32%	126,27%	133,33%	125%	128,95%	103,86%	103,09%	102,83%	

KAYNAKÇA

- Akkaya, E. E., Özer, M., Şentürk, K., Öztürk, V. ve Alkan, Ü. (2022). Design and control of a mobile steward platform with four independent wheels. *Mugla Journal of Science and Technology*, 8 (1) , 19-25 . DOI: 10.22531/muglajsci.982013.
- Alp, H. (2007). Sezgisel yöntemlerle paralel mekanizmaların çalışma uzayı analizi. Doktora Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Anlı, E. (2005). Positional Kinematics Analysis of The 6-3 Stewart Platform Mechanism Using Heuristic Algorithms. M.SC.Thesis, ITU, 4-12.
- Anlı, E., Alp, H., Sait N. Yurt ve Özkol, İ. (2005). Paralel Mekanizmaların Kinematiği, Dinamiği ve Çalışma Uzayı. *Havacılık ve Uzay Teknolojileri Dergisi Cilt 2 Sayı 1*, 19- 36.
- Barbosa, M., Solteiro, P.E. ve Lopes, A. (2010). Optimization of Parallel Manipulators Using Evolutionary Algorithms. 10.1007/978-3-642-13161-5_11.
- Benali, A., Richard, P. ve Bidaud, P. (1999). Design, control and evaluation of a six DOF force feedback interface for virtual reality applications. *Robot and Human Communication - Proceedings of the IEEE International Workshop. Paris*. 338 - 343.
- Bingü, Z. ve Küçük, S. (2005). Robot tekniği 1. Birsen Yayınevi, İstanbul, 275 -282.
- Bonev I. (2004). Delta Parallel Robot the Story of Success. Erişim adresi: <http://www.parallemic.org/Reviews/Review007.html>
- Braun, E.U., Mayer, H., Knoll, A., Lange, R. ve Bauernschmitt, R. (2008). The MustHave in Robotic Heart Surgery: Haptic Feedback, Medical Robotics, I-Tech Education and Publishing, Vienna, 9-21.
- Broenink, J.F., Visser, M. ve Tierneho, M.L.J. (1998). Simulation of Sliding Mode Impedance Control for a 6 DOF Anthropomorphic Robot. *Proceeding of the 12th European Simulation Multiconference. Manchester UK*. 561-565.
- Bulut, Y. (2015). Lineer Aktüatörlü Paralel Modüllerden Oluşan Seri Robotun Kontrolü. Yüksek Lisans Tezi, Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü, Denizli.
- Cha, S.H., Lasky, T.A. ve Velinsky, S.A. (2006). Kinematic redundancy resolution for serial-parallel manipulators via local optimization including joint constraints. *Mechanics Based Design of Structures and Machines: An International Journal*. 34, 213-239.
- Collins, C. L. ve Long G. L., (1995). The Singularity analysis of an In-Parallel Hand Controller for Force-Reflected Teleoperation. *IEEE Transactions on Robotics and Automation*, 5, 661-669.

- Dasgupta, B., ve Mruthyunjaya, T., S. (2000). The Stewart Platform manipulator: a review. *Mechanism and Machine Theory*, 15-40.
- Dasgupta, D. ve Mruthyunjaya, T. S., (1998). Singularity-free Path Planning for the Stewart Platform Manipulator. *Mechanism and Machine Theory*, 33, 711-725.
- Dumlu, A. ve Erentürk, K. (2011). Kablo Tabanlı Paralel Manipülâtör Dizaynı ve İleri Kinematik Probleminin Sylvester Eliminasyon Yöntemi ile Çözümü. Fırat Üniversitesi, Elazığ.
- El-Badawy, A. ve Youssef, K. (2013). On modeling and simulation of 6 degrees of freedom Stewart platform mechanism using multibody dynamics approach. *ECCOMAS Multibody Dynamics*, 1, 751-759.
- Elmas, A. (2005). Positional Kinematics Analysis of The 6-3 Stewart Platform Mechanism Using Heuristic Algorithms. M.SC.Thesis, ITU.
- Fasse, E.D. ve Broenink, J.F. (1999). A Spatial Impedance Controller for Robotic Manipulation. *IEEE International Conference on Robotics and Automation*. 1-12.
- Fichter, E.F. (1986). A Stewart Platform-Based Manipulator: General Theory and Practical Construction. *The International Journal of Robotics Research*. 5, 157-182.
- Franck, P., Vidala, Nicholas, C., Derek, A., Gould A. E. ve Healey, N.W.(2005). Developing a needle guidance virtual environment with patient-specific data and force feedback, *International Congress Series Computer Assisted Radiology and Surgery*, 418-423.
- Gallardo, J., Rico, J. M., Frisoli, A., Checcacci, D. ve Bergamasco, M. (2003). Dynamics of parallel manipulators by means of screw theory. *Mechanism and Machine Theory*, 38, 1113-1131.
- Gao, Z., Zhang, D. ve Ge, Y. (2010). Design optimization of a spatial six degree-of freedom parallel manipulator based on artificial intelligence approaches. *Robotics and Computer-Integrated Manufacturing*. 26, 180-189.
- Gosselin, C. ve Angeles, J. (1990). Singularity Analysis of Closed-Loop Kinematic Chains. *IEEE Transactions on Robotics ve Automation*, Canada, 281-283.
- Gourin, C.G. ve Terris, D.J. (2007). *Robotics in surgery: history, current and future applications*, 1st ed., Nova Science Publishers Inc., New York, 3-12.
- Gökçen, H.M. (1992). Control of a Stewart mechanism using direct kinematic solution. Yüksek Lisans Tezi, Boğaziçi Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Guo, H. B. ve Li, H. R. (2006). Dynamic analysis and simulation of a six degree of freedom Stewart platform manipulator. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 220(1), 61-72.

- Güneri, B. (2007). Complete Dynamic Analysis of Stewart Platform Including Singularity Detection. Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü, İzmir, 1-9.
- Harib, K., ve Srinivasan, K. (2003). Kinematic and dynamic analysis of Stewart platform-based machine tool structures. *Robotica*, 21, 541-554.
- Hay, A.M. ve Snyman, J.A. (2005). A multi-level optimization methodology for determining the dextrous workspaces of planar parallel manipulators. *Structural and Multidisciplinary Optimization*. 30, 422-427.
- Hayward, V., Choksi, J., Lanvin, G. ve Ramstein, C. (1994). Design and multi-objective optimization of a linkage for a haptic interface. in Lenarcic, J. and Ravan, B. (Eds), *Advances in Robot Kinematics*, Kluwer Academic, Dordrecht, pp. 352-9.
- Hogan, N. (1985). Impedance Control: An Approach to Manipulation: Part I-II-III. *Journal of Dynamics Systems, Measurement and Control*. 1-24.
- İnal, R.U. (2019). Üç serbestlik dereceli paralel eksenli kararlılaştırma platformunun dinamik modellenmesi ve kontrolü, Gazi Üniversitesi, Ankara.
- Ji, Z. (1993). Study of the effect of leg inertia in Stewart platforms. *IEEE International Conference on Robotics and Automation*, 1, 121-126.
- Jiang, Q. (2008). Singularity-free workspace analysis and geometric optimization of parallel mechanisms, Phd Thesis, Université Laval, Faculté Des Sciences Et De Génie, Québec, Canada.
- Jinno, M., Ozaki, F., Tatsuno, K. ve Takahashi, M. (1995). Development of a Force Controlled Robot for Grinding, Chamfering and Polishing. *IEEE International Conference on Robotics and Automation*. 1455-1460.
- Khail, W. ve Dombre, E. (2002). *Modeling, Identification & Control of Robots*. 174-177.
- Kızılhan, A., Toz, M., Aliustaoğlu, C. ve Bingül, Z. (2007). Gezgin robot tasarımı ve hareket planlaması. *TOK'07 Bildiriler Kitabı*, İstanbul, 371 -376.
- Kim, D., Chung W. ve Youm Y. (1999). Analytic Singularity Equation and analysis of 6-Dof Parallel Manipulators using Local Structurization Method. *IEEE Transactions on Robotics and Automation*, 15, 612-622.
- Kim, D.H., Kang, J-Y. ve Lee II, K. (2000). Robust tracking control design for a 6 DOF parallel manipulator. *Journal of Robotic Systems*, 17(10), 527-547.
- Kızır, S., Bingül, Z. (2019). Design and development of a Stewart platform assisted and navigated transsphenoidal surgery. 1-2.

- Kızır, S., Bingül, Z., Oysu, C. ve Küçük, S. (2011). Development and Control of a High Precision Stewart Platform, *SDU International Journal of Technological Sciences* Vol. 3, No 1/Special Issue, 51-59
- Kosinska, A., Galicki, M. ve Kedzior, K. (2003). Designing and optimization of parameters of delta-4 parallel manipulator for a given workspace, *Journal of Robotic Systems*. 20, 539-548.
- Lebret, G., Liu K. ve Lewis F. L. (1993). Dynamics analysis and control of a Stewart platform manipulator. *Journal of Robotics Systems*, 10(5), 625-655.
- Lee, S. H., Song, J. B., Choi, W. C., ve Hong, D. (2002). Workspace and Force-Moment Transmission of a Variable Arm Type Parallel Manipulator. *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, Washington, May, 3666-3671.
- Li, D. ve Salcudean, S. E. (1997). Modeling, simulation, and control of a hydraulic Stewart platform. *IEEE International Conference on Robotics and Automation*, Albuquerque, New Mexico, 3360-3366.
- Lin, J., Chen, C.W. (2009). Computer-aided-symbolic dynamic modeling for Stewart platform manipulator, *Robotica* vol.27, 331–341.
- Liu, M. J., Li, C. X ve Li, C. N. (2000). Dynamics analysis of the Gough–Stewart platform manipulator. *IEEE Transactions on Robotics and Automation*, 16(1), 94-98.
- Majid, M.Z.A., Huang,Z. ve Yao, Y.L. (2000). Workspace analysis of a six-degrees of freedom, three-prismatic–prismatic–spheric-revolute parallel manipulator. *Int. J. Adv. Manuf. Technol*, 441–449.
- Merlet, J. P. (2006). *Parallel Robots (Second Edition)*. Springer, Dordrecht, Netherlands, 261-312.
- Merlet, J.P. (1993). Direct Kinematics of Parallel Manipulators. *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 6.
- Merlet, J.P. (1998). Determination of the Presence of Singularities in 6D Workspace of a Gough Parallel Manipulator, *Advances in Robot Kinematics: Analysis and Control*. Kluwer Academic Publishers, Netherlands.
- Neeraj, B., Jonathan, K. ve Elvis, E. (2018). Kinematic Analysis of 3 Designed Gough Stewart Platform using CAD and Mathematics Software. 1-6.
- Ou, F.M. (2005). A physical oriented methodology for the synthesis of functional alternatives of mechanism systems. Phd Thesis, Department of Mechanical Engineering, National Cheng Kung University, Tainan City, Taiwan.
- Özer, M. (2019). Stewart Platformu İle Üst Plaka Dengeleyici ve Açışal Teker Kontrollü Hibrit Araç Prototip Uygulaması. Yüksek Lisans Tezi, İstanbul Gelişim Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

- Öztürk, V. (2019). Doğrusal Motor Sistemi Üzerine 6 Serbestlik Derecesine Sahip Platform Tasarımının Yapılarak Uygulanması Ve Yazılımın En İyileştirilmesi. Yüksek Lisans Tezi, İstanbul Gelişim Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Peng, Z. (1992). Compliant Motion Control of Manipulators. Thesis of Doctor of Philosophy, Niigata University, Niigata Japan.
- Sait, N. Yurt, (2002). 6-3 Stewart Platform Mekanizmasının Kinematik, Dinamik Analizi ve Kontrolü. Doktora Tezi, ITU.
- Salisbury, J.K. (1980). Active Stiffness Control of a Manipulator in Cartesian Coordinates. Proceeding 19th IEEE Conference on Decision and Control, December. 95-100.
- Seiichiro, K., Wataru, I. ve Kouhei, O. (2005). Medical mechatronics An application to haptic forceps, Annual Reviews in Control. 29(2), 237-245.
- Su, Y.X., Duan, B.Y., Peng, B. ve Nan, R.D. (2001). A real-coded genetic optimal kinematic design of a Stewart fine tuning platform for a large radio telescope, Journal of Robotic Systems. 18, 507-516.
- Tsai, L.W. (1999). Robot Analysis: The Mechanics of Serial and Parallel Manipulators. John Wiley & Sons. New York.
- Ulaş, B. (2009). Stewart Platform Tasarımı. Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, İstanbul, 42-43.
- Ünsal, A. (2007). Farklı Yapıdaki Stewart Platform Mekanizmalarının Düz ve Ters Kinematik Analizi. Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Wang, A. (2007). Reconfigurable kinematics of General Stewart Platform and simulation interface. Master Thesis, University of Windsor, Canada.
- Wang, J. ve Gosselin, C. (1998). A new approach for the dynamic analysis of parallel manipulators. Multibody System Dynamics, 2, 317-334.
- Yanık, H. (2016).Kablo Tabanlı Bir Paralel Manipulatörün Konum Kontrolünün Sağlanması. Yüksek Lisans Tezi, Gaziosmanpaşa Üniversitesi, Fen Bilimleri Enstitüsü, Tokat.
- Yoshikawa, T. (1990). Foundation of Robotics Analysis and Control. MIT Press, Cambridge, Massachusetts.
- Zhan, G. ve Ian, G., (2008). Haptic sculpting of multi-resolution B-spline surfaces with shaped tools. Computer-Aided Design archive, 40(10-11), 1055 1066.

EKLER

EK-A

TRİGONOMETRİK YÖNTEMLE ROTASYON HESAPLAMA KAYNAK KODU

```
package math;

import org.jzy3d.maths.Coord3d;
import org.jzy3d.plot3d.primitives.Point;

public class PointRotation3D_TrigonometricMethod extends
AbstractRotation3D<PointRotation3D_TrigonometricMethod>{

    public PointRotation3D_TrigonometricMethod(double x, double y,
double z) {
        super(x,y,z);
    }
    public PointRotation3D_TrigonometricMethod(Point point){
        this(point.xyz);
    }
    public PointRotation3D_TrigonometricMethod(Coord3d coord3d){
        this(coord3d.x, coord3d.y, coord3d.z);
    }

    @Override
    public PointRotation3D_TrigonometricMethod rotateX(double angle)
    {
        angle = Math.toRadians(angle);
        double cos = Math.cos(angle);
        double sin = Math.sin(angle);
        double y = this.y * cos - this.z * sin;
        double z = this.y * sin + this.z * cos;
        return setData(new
PointRotation3D_TrigonometricMethod(this.x, y, z));
    }
    @Override
    public PointRotation3D_TrigonometricMethod rotateY(double angle)
    {
        angle = Math.toRadians(angle);
        double cos = Math.cos(angle);
        double sin = Math.sin(angle);
        double x = this.x * cos + this.z * sin;
        double z = -this.x * sin + this.z * cos;
        return setData(new PointRotation3D_TrigonometricMethod(x,
this.y, z));
    }
    @Override
    public PointRotation3D_TrigonometricMethod rotateZ(double angle)
    {
```



```
    angle = Math.toRadians(angle);  
    double cos = Math.cos(angle);  
    double sin = Math.sin(angle);  
    double x = this.x * cos - this.y * sin;  
    double y = this.x * sin + this.y * cos;  
    return setData(new PointRotation3D_TrigonometricMethod(x, y,  
this.z));  
    }  
}
```



MATRİS ÇARPIM YÖNTEMİYLE ROTASYON HESAPLAMA KAYNAK KODU

```

package math;

import org.apache.commons.math3.linear.Array2DRowRealMatrix;
import org.apache.commons.math3.linear.RealMatrix;
import org.jzy3d.maths.Coord3d;
import org.jzy3d.plot3d.primitives.Point;

public class PointRotation3D_RotationMatrixMethod extends
AbstractRotation3D<PointRotation3D_RotationMatrixMethod>{

    public PointRotation3D_RotationMatrixMethod(double x, double y,
double z) {
        super(x,y,z);
    }

    public PointRotation3D_RotationMatrixMethod(Point point){
        this(point.getCoord());
    }
    public PointRotation3D_RotationMatrixMethod(Coord3d coord3d){
        this(coord3d.x, coord3d.y, coord3d.z);
    }

    @Override
    public PointRotation3D_RotationMatrixMethod rotateX(double
angle) {
        angle = Math.toRadians(angle);
        double[][] rotationMatrix =
        {
            {1, 0, 0, 0},
            {0, Math.cos(angle), -Math.sin(angle), 0},
            {0, Math.sin(angle), Math.cos(angle), 0},
            {0, 0, 0, 1}
        };
        double[][] pointMatrix =
        {
            {x},
            {y},
            {z},
            {1}
        };

        RealMatrix matrix1 = new
Array2DRowRealMatrix(rotationMatrix);
        RealMatrix matrix2 = new Array2DRowRealMatrix(pointMatrix);
        RealMatrix resultMatrix = matrix1.multiply(matrix2);
    }
}

```

```

        return setData(
            new PointRotation3D_RotationMatrixMethod(
                resultMatrix.getEntry(0, 0),
                resultMatrix.getEntry(1, 0),
                resultMatrix.getEntry(2, 0)
            )
        );
    }

    @Override
    public PointRotation3D_RotationMatrixMethod rotateY(double
angle) {
        angle = Math.toRadians(angle);
        double[][] rotationMatrix =
        {
            {Math.cos(angle), 0, Math.sin(angle), 0},
            {0, 1, 0, 0},
            {-Math.sin(angle), 0, Math.cos(angle), 0},
            {0, 0, 0, 1}
        };
        double[][] pointMatrix =
        {
            {x},
            {y},
            {z},
            {1}
        };
        RealMatrix matrix1 = new
Array2DRowRealMatrix(rotationMatrix);
        RealMatrix matrix2 = new Array2DRowRealMatrix(pointMatrix);
        RealMatrix resultMatrix = matrix1.multiply(matrix2);
        return setData(
            new PointRotation3D_RotationMatrixMethod(
                resultMatrix.getEntry(0, 0),
                resultMatrix.getEntry(1, 0),
                resultMatrix.getEntry(2, 0)
            )
        );
    }

    @Override
    public PointRotation3D_RotationMatrixMethod rotateZ(double
angle) {
        angle = Math.toRadians(angle);
        double[][] rotationMatrix =
        {
            {Math.cos(angle), -Math.sin(angle), 0, 0},
            {Math.sin(angle), Math.cos(angle), 0, 0},
            {0, 0, 1, 0},
            {0, 0, 0, 1}
        };
        double[][] pointMatrix =

```

```

        {
            {x},
            {y},
            {z},
            {1}
        };
        RealMatrix matrix1 = new
Array2DRowRealMatrix(rotationMatrix);
        RealMatrix matrix2 = new Array2DRowRealMatrix(pointMatrix);
        RealMatrix resultMatrix = matrix1.multiply(matrix2);
        return setData(
            new PointRotation3D_RotationMatrixMethod(
                resultMatrix.getEntry(0, 0),
                resultMatrix.getEntry(1, 0),
                resultMatrix.getEntry(2, 0))
        );
    }
}

```

HAREKETLİ PLATFORMUN POLİGONAL KONUM HESAPLAMA KAYNAK KODU

```

package component;

import component.composites.Cylinder;
import controller.RootController;
import math.util.VectorProcessUtility;
import org.jzy3d.colors.Color;
import org.jzy3d.maths.Coord3d;
import org.jzy3d.plot3d.primitives.Point;
import org.jzy3d.plot3d.primitives.Polygon;
import org.jzy3d.plot3d.primitives.Sphere;
import org.jzy3d.plot3d.text.drawable.DrawableText;

public class Platform extends AbstractBase {
    private Coord3d center;
    private VectorProcessUtility.Rotation rotation;
    private Sphere[] spheres;
    private Sphere centerSpheres;
    private DotCloud dotCloud;
    public Platform(float range, float radius) {
        super(RootController.Component.PLATFORM, range, radius, new
double[] {0, 0, 120, 120, 240, 240}, true);

        center = new Coord3d(0, 0, 0);
        rotation = new VectorProcessUtility.Rotation(0, 0, 0);

        spheres = new Sphere[6];
        for (int i = 0; i < baseCoords.getPoints().size(); i++) {
            spheres[i] = new
Sphere(baseCoords.getPoints().get(i).getCoord(), 25f, 15,
Color.RED);
            Color ballColor = Color.RED;
            ballColor.a = 1f;
            spheres[i].setColor(ballColor);
            spheres[i].setWireframeColor(ballColor.alpha(0.5f));
            add(spheres[i]);
        }
        centerSpheres = new Sphere(new Coord3d().set(center), 15f,
30, null);
        centerSpheres.setColor(Color.YELLOW);
        centerSpheres.setWireframeColor(Color.BLUE);

        dotCloud = new DotCloud(this, 5f, 2, Color.RED.alpha(.05f));

        DrawableText textPlatformJoint = new
DrawableText("Merkez",center, Color.BLACK);
        add(dotCloud);
        add(baseCoords);
    }
}

```

```

        add(centerSpheres);
    }
    public long getDotCloudMarkCount () {
        return dotCloud.getDotCount ();
    }
    public void markForDotCloud () {
        dotCloud.mark ();
    }
    public void clearDotCloud () {
        dotCloud.clear ();
    }
    public void gotoZero () throws InterruptedException {
        RootController.RunMode tmpMode =
RootController.RunMode.getMode ();
        gotoZero (RootController.RunMode.WORKSPACE_ANALIYSIS);
        gotoCenter ();
        tmpMode.set ();
    }
    public void gotoCenter () throws InterruptedException {

RootController.ManipulateAxis.X.setValue (RootController.ManipulateAx
is.ManipulateType.LINEAR,0f);

RootController.ManipulateAxis.Y.setValue (RootController.ManipulateAx
is.ManipulateType.LINEAR,0f);

RootController.ManipulateAxis.Z.setValue (RootController.ManipulateAx
is.ManipulateType.LINEAR,RootController.calculateFirstHeight ());

RootController.ManipulateAxis.X.setValue (RootController.ManipulateAx
is.ManipulateType.ANGULAR,0f);

RootController.ManipulateAxis.Y.setValue (RootController.ManipulateAx
is.ManipulateType.ANGULAR,0f);

RootController.ManipulateAxis.Z.setValue (RootController.ManipulateAx
is.ManipulateType.ANGULAR,0f);

RootController.setPlatformManipulate (RootController.RunMode.WORKSPAC
E_ANALIYSIS, RootController.ManipulateAxis.ManipulateType.LINEAR);

RootController.setPlatformManipulate (RootController.RunMode.WORKSPAC
E_ANALIYSIS, RootController.ManipulateAxis.ManipulateType.ANGULAR);
    }
    public void gotoZero (RootController.RunMode mode) throws
InterruptedException {
        mode.set ();
        RootController.setPlatformManipulate (mode,
RootController.ManipulateAxis.ManipulateType.LINEAR,0,0,RootControll
er.calculateFirstHeight ());
        RootController.setPlatformManipulate (mode,
RootController.ManipulateAxis.ManipulateType.ANGULAR,0,0,0);

```

```

    }
    public Coord3d getCenter() {
        return this.center;
    }
    public float getCenter(RootController.ManipulateAxis axis) {
        switch (axis){
            case X:
                return getCenter().x;
            case Y:
                return getCenter().y;
            case Z:
                return getCenter().z;
            default:
                throw new RuntimeException("axis="+axis+" parametre
karşılığı bulunamadı.");
        }
    }
    public VectorProcessUtility.Rotation getRotation() {
        return rotation;
    }
    public float getRotation(RootController.ManipulateAxis
manipulateAxis) {
        switch (manipulateAxis){
            case X:
                return rotation.getX_Angle();
            case Y:
                return rotation.getY_Angle();
            case Z:
                return rotation.getZ_Angle();
            default:
                throw new RuntimeException("manipulateParam="+
manipulateAxis +" bulunamadı. Geçerli prametre gönderilmelidir.");
        }
    }
    public float
getPlatformManipulateParam(RootController.ManipulateAxis.ManipulateT
ype type, RootController.ManipulateAxis axis){
        switch (type){
            case ANGULAR:
                return getRotation(axis);
            case LINEAR:
                return getCenter(axis);
            default:
                throw new RuntimeException("type="+type+" parametre
bulunamadı.");
        }
    }
    public AbstractBase reCalculateBaseCoords(float range, float
radius) {
        super.reCalculateBaseCoords(range, radius);
        translationSet(center);
    }

```

```

        rotateSet(rotation);
        update();
        System.gc();
        return this;
    }
    public AbstractBase reCalculateBaseCoords() {
        Polygon tmpPolygon = super.calculateBaseCoords();
        translationSet(tmpPolygon, center);
        rotateSet(tmpPolygon, rotation);
        transferCoord(tmpPolygon, baseCoords);
//        update();
        return this;
    }
    public void translationApply(Coord3d value) {
        for (Point point : baseCoords.getPoints())
            point.setCoord(point.getCoord().add(value));
        center.set(center.add(value));
        update();
    }

    private void translationSet(Coord3d center) {
        translationSet(baseCoords, center);
    }
    private void translationSet(Polygon polygon, Coord3d center) {
        for (Point point : polygon.getPoints())
            point.setCoord(point.getCoord().add(center));
        this.center.set(center);
    }

    public void rotateApply(VectorProcessUtility.Rotation rotate) {
        rotateApply(baseCoords, rotate);
    }

    public void rotateApply(Polygon polygon,
VectorProcessUtility.Rotation rotate) {
        rotation.add(rotate);

VectorProcessUtility.rotateAndCalculateNewPosition_Trigonometrikyönt
em(polygon, center, rotate);
        update();
    }

    public void reset() {
        rotateSet(new VectorProcessUtility.Rotation(0, 0, 0));
        translationSet(new Coord3d(0, 0,
RootController.calculateFirstHeight()));
    }

    private void rotateSet(VectorProcessUtility.Rotation rotate) {
        rotateSet(baseCoords, rotate);
    }
}

```



```

        private void rotateSet(Polygon polygon,
VectorProcessUtility.Rotation rotate) {
            rotation.set(rotate);
        }

VectorProcessUtility.rotateAndCalculateNewPosition_TrigonometrikYönt
em(polygon, center, rotate);
        update();
    }

    @Override
    protected void update() {
        super.update();
        sphereUpdate();
        armsUpdate();
        sphereCenterUpdate();
    }

    private void sphereCenterUpdate() {
        centerSpheres.setPosition(center);
    }

    private void armsUpdate() {
        if (arms != null)
            for (Cylinder arm : arms)
                arm.reCalculate();
    }

    private void sphereUpdate() {
        for (int i = 0; i < baseCoords.getPoints().size(); i++) {
spheres[i].setPosition(baseCoords.getPoints().get(i).getCoord());

        }

    }

    @Override
    public String toString(){
        return String.format("(a=%.0f, b=%.0f, c=%.0f, d=%.0f,
e=%.0f, f=%.0f)",
            RootController.SkeletonParam.A.getValue(),
            RootController.SkeletonParam.B.getValue(),
            RootController.SkeletonParam.C.getValue(),
            RootController.SkeletonParam.D.getValue(),
            RootController.SkeletonParam.E.getValue(),
            RootController.SkeletonParam.F.getValue());
    }
}

```

SABİT ZEMİNİN POLİGONAL KONUM HESAPLAMA KAYNAK KODU

```

package component;

import controller.RootController;
import org.jzy3d.colors.Color;
import org.jzy3d.plot3d.primitives.Point;
import org.jzy3d.plot3d.primitives.Sphere;

public class Ground extends AbstractBase {

    public Ground(float range, float radius){
        super(RootController.Component.GROUND, range, radius, new
double[]{300, 60, 60, 180, 180, 300}, false);

        baseCoords.setColor(Color.BLUE);

        Color ghostColor = Color.GREEN;
        ghostColor.a = 0.1f;
        baseCoords.setColor(ghostColor);
        baseCoords.setWireframeColor(ghostColor.alpha(0.5f));

        add(baseCoords);

        for(Point p: baseCoords.getPoints()){
            Sphere sphere = new Sphere(p.getCoord(), 30, 10,
Color.MAGENTA);
            sphere.setWireframeColor(sphere.getColor().alpha(0.5f));
            add(sphere);
        }
    }
}

```

NOKTA BULUTU KAYNAK KODU

```

package component;

import analysis.WorkSpaceAnalisys;
import org.jzy3d.colors.Color;
import org.jzy3d.maths.Coord3d;
import org.jzy3d.plot3d.primitives.Point;
import org.jzy3d.plot3d.primitives.Shape;
import org.jzy3d.plot3d.primitives.Sphere;

import java.util.HashSet;

public class DotCloud extends Shape {
    Platform platform;
    float radius;
    int slices;
    Color color;

    HashSet<SmartSphere> cloudDotsSet;

    private long dotCount;
    public DotCloud(Platform platform, float radius, int slices,
Color color) {
        cloudDotsSet = new HashSet<>();
        dotCount=0;

        this.platform = platform;
        this.radius = radius;
        this.slices = slices;

                                                                 this.color = color;
    }

    private Coord3d newCoord() {
        return new Coord3d().set(platform.getCenter());
    }

    void mark() {
        SmartSphere dotSphere = new SmartSphere(newCoord(), radius,
slices, color);
        dotSphere.setColor(color);
        if(!cloudDotsSet.contains(dotSphere)){
            dotSphere.setWidth(WorkSpaceAnalisys.RESULITION);
            add(dotSphere);
            cloudDotsSet.add(dotSphere);
            dotCount++;
        }
    }

    public long getDotCount(){

```

```

        return dotCount;
    }

    public void clear() {
        dotCount=0;
        super.clear();
        cloudDotsSet.clear();
        System.gc();
    }
    private class SmartSphere extends Point { // Sphere
        boolean top, bottom, left, right;

        public SmartSphere(Coord3d position, float radius, int
slicing, Color color) {
            super(position, color);
            super.setWidth(radius);
            this.top = false;
            this.bottom = false;
            this.left = false;
            this.right = false;
        }

        public float getX() {
            return super.getCoord().x;
//            return super.getPosition().x;
        }

        public float getY() {

            return super.getCoord().y;
//            return super.getPosition().y;
        }

        public float getZ() {
            return super.getCoord().z;
//            return super.getPosition().z;
        }

        public boolean isSameX(SmartSphere other) {
            return this.getX() == other.getX();
        }

        public boolean isSameY(SmartSphere other) {
            return this.getY() == other.getY();
        }

        public boolean isSameZ(SmartSphere other) {
            return this.getZ() == other.getZ();
        }
    }
}

```

POLİGONAL FORMDAKİ PLATFORM VE ZEMİNİN SOYUT SINIF YAPISINDA KONUM HESAPLAMA KAYNAK KODU

```

package component;

import component.composites.Cylinder;
import component.polygons.Base;
import controller.RootController;
import org.jetbrains.annotations.NotNull;
import org.jzy3d.plot3d.primitives.Polygon;
import org.jzy3d.plot3d.primitives.Shape;
public class AbstractBase extends Shape implements Resizable {
    protected Cylinder[] arms;
    protected final Base.SixDofBaseSetupParameters
parametersForBase;
    protected final Polygon baseCoords;
    private RootController.Component component;

    public AbstractBase(RootController.Component component, float
range, float radius, @NotNull final double refAngles[], boolean
doubleNumIndisMinusProcess) {

        this.component = component;
        this.component.setComponent(this);

        setParameterForBase(range, radius);

        parametersForBase = new
Base.SixDofBaseSetupParameters(refAngles,
doubleNumIndisMinusProcess);

        baseCoords = calculateBaseCoords();
    }

    private void setParameterForBase(float range, float radius) {
        switch (this.component) {
            case PLATFORM:
                RootController.SkeletonParam.A.updateValue(range);
                RootController.SkeletonParam.B.updateValue(radius);
                break;
            case GROUND:
                RootController.SkeletonParam.C.updateValue(range);
                RootController.SkeletonParam.D.updateValue(radius);
                break;
        }
    }

    protected Polygon calculateBaseCoords() {

```

```

        return new
Base.Builder(parametersForBase).range(getRange()).radius(getRadius()
).build();
    }

    public float getRange() {
        switch (this.component) {
            case PLATFORM:
                return RootController.SkeletonParam.A.getValue();
            case GROUND:
                return RootController.SkeletonParam.C.getValue();
        }
        throw new RuntimeException(this.component.name()+"
componentType türü için tanımlanan range değeri verilemiyor.");
    }

    public float getRadius() {
        switch (this.component) {
            case PLATFORM:
                return RootController.SkeletonParam.B.getValue();
            case GROUND:
                return RootController.SkeletonParam.D.getValue();
        }
        throw new RuntimeException(this.component.name()+"
componentType türü için tanımlanan radius değeri verilemiyor.");
    }
    public Polygon getCoords() {
        return baseCoords;
    }

    protected void update() {
        reCalculateArmStatus();
    }

    private void reCalculateArmStatus() {
        if (arms != null)
            for (Cylinder arm : arms)
                arm.reCalculate();
    }

    public void setArms(Cylinder[] arms) {
        this.arms = arms;
    }

    public Cylinder[] getArms() {
        return this.arms;
    }

    public AbstractBase reCalculateBaseCoords(float range, float
radius) {
        setParameterForBase(range, radius);
        return reCalculateBaseCoords();
    }

```

```
}

public AbstractBase reCalculateBaseCoords() {
    transferCoord(calculateBaseCoords(), baseCoords);
    update();
    return this;
}

public void transferCoord(Polygon source, Polygon target){
    for (int i = 0; i < target.getPoints().size(); i++) {
        target.get(i).setCoord(source.get(i).getCoord());
    }
}

@Override
public void reSize() {
    reCalculateBaseCoords();
}
}
```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : ŞAHİN, ÖMER

Uyruğu : TÜRKİYE

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Lisans	İstanbul Gelişim Üniversitesi / Bilgisayar Müh.	2018
Lise	Kıraç Borsa İstanbul Mesleki ve Teknik Anadolu Lisesi	2012

