

JULY 2015

M.Sc. in Mechanical Engineering

AHMET ŞUMNU

**UNIVERSITY OF GAZİANTEP
GRADUATE SCHOOL OF
NATURAL & APPLIED SCIENCES**

**DESIGN AND ANALYSIS OF THE STEWART
PLATFORM**

**M.Sc. THESIS
IN
MECHANICAL ENGINEERING**

**BY
AHMET ŞUMNU**

JULY 2015

Design and Analysis of the Stewart Platform

M.Sc. Thesis

in

Mechanical Engineering

University of Gaziantep

Supervisor

Prof. Dr. İbrahim Halil GÜZELBEY

Co-Supervisor

Assist. Prof. Dr. M. Veysel ÇAKIR

By

Ahmet ŞUMNU

July 2015

© 2015 [Ahmet ŞUMNU]

REPUBLIC OF TURKEY
UNIVERSITY OF GAZİANTEP
GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES
MECHANICAL ENGINEERING

Name of the thesis: Design and Analysis of Stewart Platform

Name of the student: Ahmet ŞUMNU

Exam date: 21.07.2015

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Metin BEDİR

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Sait SÖYLEMEZ

Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. M. Veysel ÇAKIR

Co-Supervisor

Prof. Dr. İbrahim Halil GÜZELBEY

Supervisor

Examining Committee Members :

Prof. Dr. Naki TÜTÜNCÜ

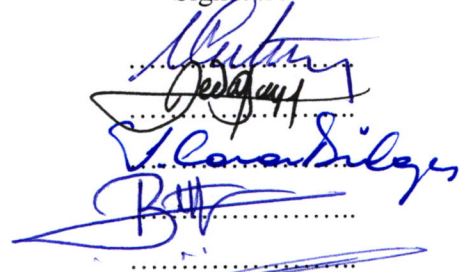
Prof. Dr. Sedat BAYSEÇ

Prof. Dr. L. Canan DÜLGER

Prof. Dr. Bahattin KANBER

Prof. Dr. İbrahim Halil GÜZELBEY

Signature



I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Ahmet ŞUMNU

ABSTRACT

DESIGN AND ANALYSIS OF THE STEWART PLATFORM

ŞUMNU, Ahmet

M.Sc. in Mechanical Eng.

Supervisor: Prof. Dr. İbrahim Halil GÜZELBEY

July 2015

83 pages

In this study, kinematic analysis, dynamics analysis and control of a Stewart platform using linear motors have been carried out. The main aim of this study is to obtain higher acceleration, velocity and to increase the efficiency using linear motors. In the kinematic analysis, lengths of the legs are found by using inverse kinematic method. Inverse Jacobian matrix is obtained to find velocity vectors. In the dynamic analysis, Lagrange equation and Newton Euler method are used to obtain the general equations of motion of Stewart platform. The leg dynamics of the system is carried out by using Lagrange equation. Moving platform dynamics is then combined with leg dynamics by means of Newton Euler method. In order to perform dynamic simulation, the required actuator forces are computed by developed MATLAB code. In addition, to determine the effect of the lower part of the leg inertia, dynamic simulation is performed both with and without inertia of the lower part of the leg. PID control of Stewart platform is performed by using the mathematical model of the system in MATLAB/Simulink. Using the linear motors dynamic equation, its transfer function is derived and block diagram is obtained in Simulink. Finally, some case studies are performed to verify the developed control system. The simulation results are presented.

Key Words: Stewart Platform, Linear Motor, PID controller

ÖZET

STEWART PLATFORMUN TASARIMI VE ANALİZİ

ŞUMNU, Ahmet

Yüksek Lisans Tezi, Makine Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. İbrahim Halil GÜZELBEY

Temmuz 2015

83 sayfa

Bu çalışmada, doğrusal motor kullanılan bir Stewart platformun kinematik ve dinamik analizi ve kontrolü gerçekleştirilmektedir. Bu çalışmanın amacı doğrusal motor kullanarak yüksek ivme, hız elde etmek ve verimliliği arttırmaktır. Kinematik analizde, tersine kinematik yöntem kullanılarak bacak uzunlukları bulunmaktadır. Hız vektörlerini bulmak için ters Jacobian matris elde edilmektedir. Dinamik analizde, Stewart platformun genel hareket denklemini elde etmek için Lagrange ve Newton Euler metodu kullanılmaktadır. Sistemin bacak dinamiği Lagrange denklemi kullanılarak gerçekleştirilmiştir. Newton Euler yöntemiyle daha sonra hareketli platformun dinamiği ile bacak dinamiği birleştirilmektedir. Dinamik benzetim gerçekleştirmek için gerekli olan eyleyici kuvvetler, MATLAB programında geliştirilen kodlarla hesaplanmaktadır. Ayrıca, bacağın alt kısmının atalet etkisini belirlemek için, bacağın alt kısmının hem ataletli hem de ataletsiz olarak dinamik benzetimi gerçekleştirilmektedir. Stewart platformun PID kontrolü MATLAB/Simulink programında elde edilen matematik model kullanılarak gerçekleştirilmektedir. Doğrusal motor dinamik denklemi kullanılarak, doğrusal motorun transfer fonksiyonu türetilmekte ve Simulink programında blok diyagramı elde edilmektedir. Son olarak geliştirilen kontrol sistemini doğrulamak için bazı örnek çalışmalar yapılmıştır. Benzetim sonuçları sunulmuştur.

Anahtar Kelimeler: Stewart Platformu, Doğrusal Motor, PID kontrol

ACKNOWLEDGEMENTS

Firstly, I am very grateful to my supervisor Prof. Dr. İbrahim Halil GÜZELBEY for his guidance and support from the beginning to the end of this study. It has been an honour to be his student as well. I also would like to convey my gratitude to my co-supervisor Asst. Prof. Dr. Mehmet Veysel ÇAKIR.

I would like to thank to Prof. Dr. L. Canan DÜLGER, Asst. Prof. Dr İbrahim GÖV, Instructor İlyas KARASU, Instructor Mehmet Hanifi DOĞRU, Instructor Ünal HAYTA and Research Asst. Suna GÜÇYILMAZ for their valuable comments and sharing their knowledge.

The deepest thankfulness is to my parents, for their endless support and interest.

TABLE OF CONTENTS

ABSTRACT	v
ÖZET	vi
ACKNOWLEDGEMENTS	vii
LIST OF TABLES	xi
LIST OF FIGURES	xii
LIST OF SYMBOLS	xiv
CHAPTER 1	1
INTRODUCTION	1
1.1 Scope of the Thesis	1
CHAPTER 2	4
LITERATURE SURVEY	4
2.1 Introduction	4
2.2 Kinematic Analysis of Stewart Platform	6
2.3 Dynamic Analysis and Design of Stewart Platform	8
2.4 Control and Simulation of Stewart Platform	12
2.5 Workspace and Singularity Analysis	15
2.6 Motivation of Study	17
CHAPTER 3	18
KINEMATIC ANALYSIS OF THE STEWART PLATFORM	18
3.1 Rotational Matrix	18
3.2 Angular Velocity of the Moving Platform	20

3.3	Inverse Kinematics	20
3.3.1	Inverse Position Analysis of the Mechanism	20
3.3.2	Inverse Velocity Analysis of the Mechanism	22
CHAPTER 4.....		26
DYNAMIC ANALYSIS OF THE STEWART PLATFORM.....		26
4.1	Dynamic Analysis of a Link.....	26
4.2	Dynamic Analysis of the Moving Platform	31
4.3	Linear Motor Dynamics	37
CHAPTER 5.....		39
PID CONTROL OF THE STEWART PLATFORM		39
5.1	Introduction	39
5.2	PID Control of the System	40
5.2.1	Case Study I	42
5.2.1.1	Results and Discussion of Case Study I	42
5.2.2	Case Study II	45
5.2.2.1	Results and Discussion of Case Study II.....	45
5.2.3	Case Study III.....	48
5.2.3.1	Results and Discussion of Case Study III.....	48
5.2.4	Case Study IV.....	51
5.2.4.1	Results and Discussion of Case Study IV	51
5.3	PID Control of the System with Linear Motor.....	54
5.3.1	Case Study I	55
5.3.1.1	Results and Discussion of Case Study I	55
5.3.2	Case Study II	58
5.3.2.1	Results and Discussion of Case Study II.....	58
5.3.3	Case Study III.....	61
5.3.3.1	Results and Discussion of Case Study III.....	61
CHAPTER 6.....		64

CONCLUSIONS	64
RECOMMENDATIONS FOR THE FUTURE WORK.....	65
REFERENCES	66
APPENDICES	71
Appendix A	71
Appendix B.....	74
Appendix C.....	75

LIST OF TABLES

	Page
Table 4.1 Mechanism Parameters.....	34
Table 4.2 Linear Motor Parameters	37

LIST OF FIGURES

Figure 2.1 One of the First Parallel Mechanism (Pollard, 1942)	4
Figure 2.2 Tyre Testing Machine (Gough and Whitehall, 1962)	5
Figure 2.3 Flight Simulator (Stewart, 1965)	5
Figure 2.4 Parallel Mechanism with Decoupled Design (Jin et al., 2009)	7
Figure 2.5 Design of the Parallel Manipulator with Elastic Joints (Wang et al., 2003)	8
Figure 2.6 Inertial Forces and Moments Acting on the Links (Wang and Gosselin, 1998)	9
Figure 2.7 A Six DOF Parallel Mechanism with Three Planar Motors (Ben-Horin et al., 1998)	10
Figure 2.8 Bond Graph Model of System for a Leg (Yıldız et al., 2009)	11
Figure 2.9 Schematic Representation of an Actuator (Meng et al., 2010)	12
Figure 2.10 Stewart Platform Control Design (Omran and Kassem, 2011)	14
Figure 3.1 Closed-Loop of one Leg	21
Figure 3.2 Schematic Representation of the Stewart Platform	21
Figure 3.3 Model of Stewart Platform (Williams II, 2015)	25
Figure 4.1 Force Analysis of the Actuator	26
Figure 4.2 Forces and Moments Acting on Moving Platform	32
Figure 4.3 (a) Driving Forces with respect to Desired Trajectories (Guo and Li, 2006) (b) Driving Forces with respect to Desired Trajectories	35
Figure 4.4 (a) Driving Forces Excluded Lower Part Inertia of the Leg (b) Included Lower Part Inertia of the Leg (c) Differences Between Included and Excluded Lower Part Inertia of the Leg for the First Trajectory	35
Figure 4.5 (a) Driving Forces Excluded Lower Part Inertia of the Leg (b) Included Lower Part Inertia of the Leg (c) Differences Between Included and Excluded Lower Part Inertia of the Leg for the First Trajectory	36

Figure 4.6 (a) Driving Forces Excluded Lower Part Inertia of the Leg (b) Included Lower Part Inertia of the Leg (c) Differences Between Included and Excluded Lower Part Inertia of the Leg for the Second Trajectory	36
Figure 5.1 PID Controller Block Diagram	39
Figure 5.2 Dynamic Model of the System	40
Figure 5.3 Inverse Kinematic Block Diagram.....	41
Figure 5.4 System Block Diagram.....	41
Figure 5.5 Desired and Actual Trajectories of the System for Case Study I (a-f).....	44
Figure 5.6 Desired and Actual Trajectories of the System for Case Study II (a-f)	47
Figure 5.7 Desired and Actual Trajectories of the System for Case Study III (a-f) ...	50
Figure 5.8 Desired and Actual Trajectories of the System for Case Study IV (a-f)...	53
Figure 5.9 System Block Diagram with Linear Motors.....	54
Figure 5.10 Linear Motor Transfer Function Block	55
Figure 5.11 Desired and Actual Trajectories of the System with Linear Motors for Case Study I (a-f)	57
Figure 5.12 Desired and Actual Trajectories of the System with Linear Motors for Case Study II (a-f)	60
Figure 5.13 Desired and Actual Trajectories of the System with Linear Motors for Case Study III (a-f).....	63

LIST OF SYMBOLS

A	Amplitude of the ripple force of the linear motor
b_i	Coordinates of the i th lower junction point (m)
B_l	Coefficient of damping and viscous friction of the linear motor (Ns/m)
B_p	Coriolis-Centrifugal matrix of the moving platform
$B(P, \dot{P})$	Coriolis-Centrifugal matrix of the Stewart platform
D_c	Distance matrix according to base platform which describes between upper junction point and origin of the moving platform (m)
d_p	Distance between upper junction point of the leg and c.o.g of the moving part of the i th actuator (m)
d_b	Distance between lower junction point of the leg and c.o.g of the rotating part of the i th actuator (m)
$(F_c)_i$	Constraint force i th of the leg (N)
$(F_d)_i$	Driving force of the i th actuator (N)
$(F_{m.g})_i$	Gravitational force of the moving part of the leg (N)
$(F_{r.g})_i$	Gravitational force of the rotating part of the leg (N)
F_m	Total force which is generated by the linear motor (N)
f_c	Coulomb friction
f_{dis}	External disturbances force of the linear motor (N)
f_f	Friction force of the linear motor (N)
f_{ripple}	Ripple force of the linear motor (N)
f_s	Static friction
f_v	Viscous friction parameter

g	Gravitational acceleration (m/s^2)
I	3x3 identity matrix
I_p	Inertia of the platform with respect to center point (kgm^2)
I_m	Mass moment of inertia of the leg (kgm^2)
I_r	Mass moment of inertia of the leg (kgm^2)
$i(t)$	Linear motor armature current (A)
J^{-1}	Inverse Jacobian matrix
$K(P)$	Gravity matrix of the Stewart platform
K_e	Electromotive force coefficient of the linear motor
K_f	Force constant
K_p	Proportional constant of the PID controller
K_i	Integral constant of the PID controller
K_d	Derivative constant of the PID controller
L	Linear motor armature inductance (H)
L_i	Link vectors of the leg ($i = 1, 2, \dots, 6$) (m)
l_i	Length of the leg ($i = 1, 2, \dots, 6$) (m)
\dot{l}_i	Velocity of the leg ($i = 1, 2, \dots, 6$) (m/s)
$M(P)$	Inertia matrix of the Stewart platform
M	Moments acting on moving platform (Nm)
M_p	Inertia matrix of the moving platform
m	Mass of the linear motor (kg)
m_p	Mass of the platform (includes payload) (kg)
m_m	Mass of the moving part of the leg (kg)
m_r	Mass of the rotating part of the leg (kg)
M_f	Mass of Forcer of the linear motor (kg)

M_s	Unit mass of stator of the linear motor (kg)
P	Position and orientation of the moving platform
\dot{P}	Velocity of the moving platform
\ddot{P}	Acceleration of the moving platform
q_i^p	Coordinates of the i th upper junction point with respect to the moving platform frame (m)
q_i^B	Position vector of i th upper junction point with respect to the base frame (m)
\dot{q}_i^B	Velocity vector of i th upper junction point with respect to the base frame (m/s)
\ddot{q}_i^B	Acceleration vector of i th upper junction point with respect to the base frame (m/s ²)
q_c^p	Position vector of the center point with respect to platform frame (m)
\dot{q}_c^p	Velocity vector of the center point with respect to platform frame (m/s)
\ddot{q}_c^p	Acceleration vector of the center point with respect to platform frame (m/s ²)
R	Linear motor armature resistance (Ω)
R_p^B	Rotation matrix which describes the orientation of the moving platform reference to base platform
R_x	Rotation matrix that describes rotation about x axis
R_y	Rotation matrix that describes rotation about y axis
R_z	Rotation matrix that describes rotation about z axis
r_b	Distance between the origin of the base frame and lower junction point (m)
r_p	Distance between the origin of the platform frame and upper junction point (m)
T	Kinetic energy (J)

T_d	Derivative time (s)
T_i	Integral time (s)
t	Translation vector of the origin of the moving platform with respect to base platform (m)
\dot{t}	Translation velocity vector of the origin of the moving platform with respect to base platform (m/s)
u_i	Unit vector of the i th leg in the direction of L_i
$V(t)$	Linear motor terminal voltage (V)
v_m	Velocity of moving part of the leg (m/s)
v_r	Velocity of rotating part of the leg (m/s)
w	Angular velocity of the moving platform (rad/s)
w_{act}	Angular velocity of the actuator (rad/s)
\dot{w}	Angular acceleration of the moving platform (rad/s ²)
x	Position of the motor (m)
\dot{x}	Linear velocity of the motor (m/s)
\dot{x}_s	Lubricant parameter of the linear motor
α	angle about x axis (rad)
β	angle about y axis (rad)
γ	angle about z axis (rad)
$\delta(W)$	Virtual work of the system
δ	Additional empirical parameter
τ	Total external forces acting on the leg (N)
c_i	$\cos\theta_i$
s_i	$\sin\theta_i$

CHAPTER 1

INTRODUCTION

Stewart platform which has six degrees of freedom (3 translational motion and 3 rotational motion) is a kind of parallel manipulator and its structure is obtained from a generalization of the mechanism originally proposed by Stewart as a flight simulator (Korayem and Shokri, 2006). The Parallel manipulator is a closed-chain mechanism which has a base and a moving platform connected to each other by several extensible legs. Parallel manipulators provide high stiffness, high accuracy, high speed and also high loading capacity compared to serial manipulators.

The parallel manipulators have been used for a variety of applications in flight and vehicle simulators, high-precision machining centers, mining machines, surgical devices, entertainment, micro manipulators, haptic devices and etc. The main application area of Stewart platform is the simulation technology to simulate motion effects in airplane simulators. Thanks to this equipment, it is possible to simulate the forces acting upon the pilot during the flight, thus bringing the simulator even closer to reality. Furthermore, the simulator decreases cost of the training and also does not cause any accidents leading to dangerous circumstances.

In Stewart platforms many types of actuators are used. Hydraulic, rotary and pneumatic actuators provide high force, torque. However, linear motor provides higher speed, acceleration and efficiency than the others because transmission system of the mechanism is not needed. So, in this study use of linear motors is planned to provide high acceleration, velocity and performance for the system.

1.1 Scope of the Thesis

In this thesis, kinematic analysis, dynamic analysis and PID (Proportional, Integral, Derivative) control of the Stewart platform are carried out using linear motors. Kinematic analysis of the Stewart platform comprises position, velocity and acceleration analysis. In position analysis, the rotation matrix is obtained from Euler

angle to find coordinates of the moving platform with respect to base platform. Then, inverse kinematic method is implemented to attain equation of the leg lengths using closed loop of the leg. After that, the velocity of the legs is calculated by inverse Jacobian matrix and acceleration of the system is obtained by taking the derivative of the velocity vectors. Moreover, angular velocity of the actuator and linear velocity of the lower and upper parts of the leg are obtained separately.

Dynamic analysis of the mechanism is performed by using the Lagrange and Newton Euler equations. Initially, Lagrange equation is used to calculate the leg dynamics. Then, by using the Newton Euler method, the moving platform and leg dynamics are combined in order to obtain general equation of motion of the Stewart platform. MATLAB codes are developed to compute the actuator forces for each leg and to perform dynamic simulation according to the planned trajectories. In addition, to determine the effect of the lower part of the leg inertia, dynamic simulation is performed both with and without inertia of the lower part of the leg. Furthermore, linear motor dynamic equation and its parameters are presented in this section.

Finally, PID control of the mechanism is carried out by using MATLAB/ Simulink toolbox. There are two stages in the control section. The first stage is performed without the linear motor and the other stage is performed using the linear motor. Mathematical model of the system and inverse kinematic blocks are developed to control the system in first and second stages. Mathematical model block of the system includes general equation of motion (inertia, Coriolis and gravity matrix of the mechanism) of the Stewart platform. Inverse kinematic blocks are obtained to compute leg lengths using position and orientation of the moving platform. Two inverse kinematic blocks are placed in the control system because the control system includes both actual and desired moving platform values. In the first stage, actual and desired moving platform values are used to find position error of the mechanism. Then, the error is compensated by using proper PID controller constants. In the second stage, addition to first stage linear motor transfer function is derived and its block diagram is developed in Simulink. The developed linear motor block diagram is combined to the system mathematical model block. At the end, some case studies are realized according to different planned trajectories in order to perform position control and verify our system. The obtained results are presented graphically.

This thesis has been organized in six chapters. Chapter 1 presents the area of research work to be undertaken, the objective and the work plan are discussed. In chapter 2, previous studies are presented on the Stewart Platform manipulator. Kinematic analysis of the mechanism and Jacobian matrix are carried out in the third chapter. Dynamic analysis of the Stewart platform and linear motor are given in chapter 4. In chapter 5, control of the system is implemented using dynamic equations. Some case studies are carried out about PID control. The conclusions drawn from the work and proposed the future plan are given in the sixth chapter.

CHAPTER 2

LITERATURE SURVEY

2.1 Introduction

Parallel mechanism is based upon the first theoretical article which was published by Maxwell at the end of the 1800's (Clerc et al., 2002). Then, the first parallel mechanism which was patented by Pollard in 1942 is shown in Figure 2.1 (Pollard, 1942). The hexapod parallel mechanism was firstly proposed in England by Dr. Eric Gough and it was called as universal tyre testing machine which is shown in Figure 2.2 (Gough and Whitehall, 1962). Subsequently, Stewart (1965) designed parallel mechanism in 1965 which has six degrees of freedom and it has been used as a flight simulator which is shown in Figure 2.3. Since that time, many researchers have focused on Stewart platform to solve kinematics, dynamics, control, singularity and workspace problems of the mechanism. Besides that, some researchers have tried to increase efficiency, load capacity and stiffness of the Stewart manipulators. In this chapter a brief survey related to parallel mechanism is presented.

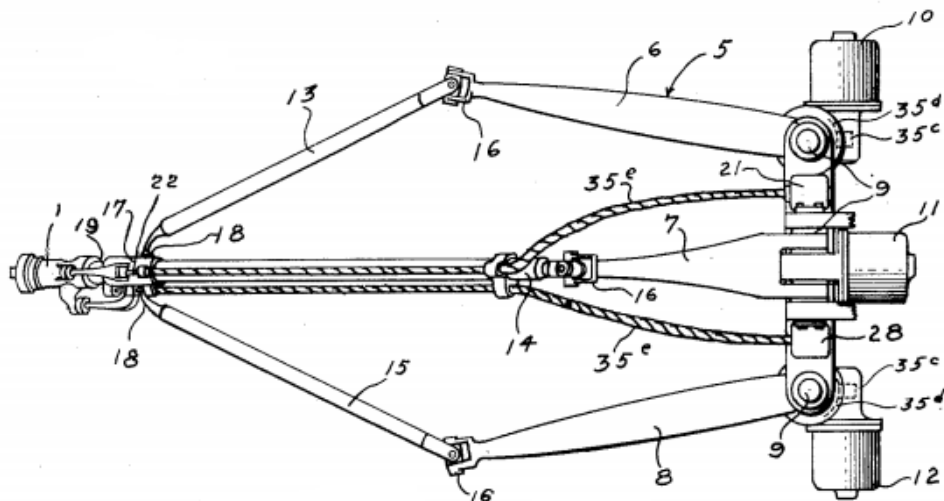


Figure 2.1 One of the First Parallel Mechanism (Pollard, 1942)

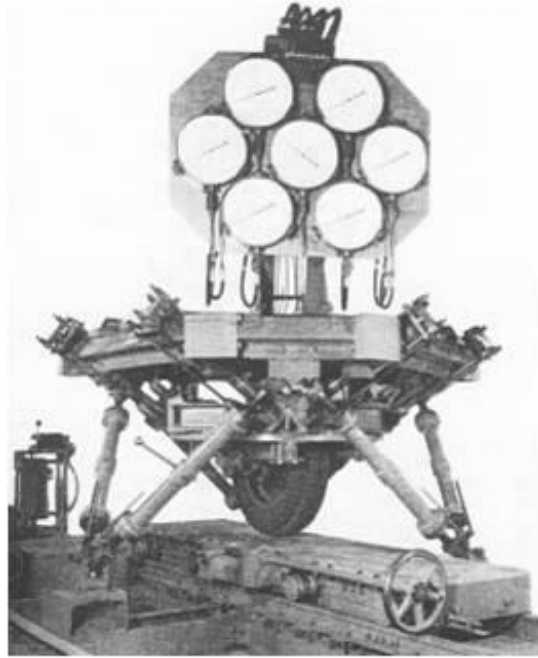


Figure 2.2 Tyre Testing Machine (Gough and Whitehall, 1962)

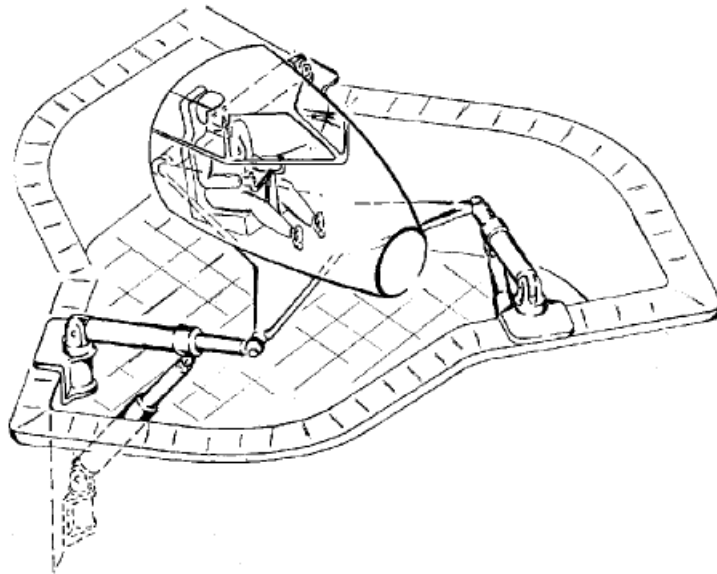


Figure 2.3 Flight Simulator (Stewart, 1965)

2.2 Kinematic Analysis of Stewart Platform

Kinematic analysis is performed to find position, velocity and acceleration of the system. There are two methods; inverse kinematic and forward kinematic analyses, to achieve kinematic analysis of parallel manipulator. Orientation and position of the platform is computed using legs position of the system in forward kinematic. Forward kinematic analysis is quite complex and solution is very difficult due to high order nonlinear equations. Researchers have developed some mathematical methods to solve the forward kinematic problems in literature. Contrarily, forward kinematic, in order to perform inverse kinematic analysis, position and orientation of the moving platform are used to compute legs position of the system. In this section, previous study related with kinematic analysis of the Stewart platform is presented.

Liu et al. (1993) developed a simple algorithm which has included three nonlinear algebraic equations. Because forward kinematic solution includes high non-linearity, these equations are simplified the solution of the problem. Another similar study was proposed related to forward kinematics problem by Jakobovic and Budin (2002). Various optimization algorithms were developed for the forward kinematics problem by combining different mathematical representations and found a suitable algorithm which can be used in real time environment. Bonev and Ryu (2000) proposed new method, which includes three extra linear sensors, to simplify the solving forward kinematics problem of a general Stewart Platform. The solution of method decreased 6 quadratic equation numbers to 3 unknowns thanks to three extra sensors. Korobeynikov and Turlapov (2005) developed an algorithm to solve the forward kinematics problems for the 6-3 and 6-6 Stewart platform. They also created the mechanism using computer program and observed that proposed algorithm is effective. Moreover, they computed curve of the tool, which is positioned on the moving platform, and possible deviation of tool according to leg lengths variations. Yıldız et al. (2010) carried out forward kinematics analysis of 3-3 six DOF parallel manipulator. Three methods were used to perform forward kinematics analysis of the system which is Newton Raphson method, Bezout method, and Artificial Neural Networks. These methods were applied actual Stewart platform to compare them. They concluded that Artificial Neural Networks gives fast response and Newton Rapson method has lower error. Harib and Srinivasan (2003) presented both inverse

kinematic and forward kinematic methods for the Stewart Platform. In this study, angular velocity and acceleration were determined according to different type of unpowered joints. They derived the inverse Jacobian matrix of the Stewart platform and its time derivative. Forward kinematic problem was also solved by using Newton Rapson method. Eventually, it is concluded that the angular velocity and acceleration of the legs indicated differences for different types of unpowered joints. Karimi and Nategh (2011) also proposed both inverse and forward kinematic analysis of the Gough-Stewart Platform. To solve nonlinearity of the forward kinematic problem of the mechanism, Bates and Watts measures of nonlinearity was used. Alrashidi et al. (2009) improved elbow joint measurement using Stewart platform. The joints motions were measured and the result of measurements was demonstrated by experimental studies. Additionally, rotating and motion angles of the forearm were obtained using Simmechanics program. They observed that the measurement device which based Stewart platform accurately measures the elbow motions in six axes. Jin et al. (2009) presented kinematic design of the 6-DOF partially decoupled parallel manipulators with leg symmetrical structure. The design is shown in Figure 2.4. This technique has provided convenience to perform the system kinematics as it reduces the order of the Jacobian matrices from six to three. Besides that, the system control and motion planning can be implemented easily thanks to this technique.

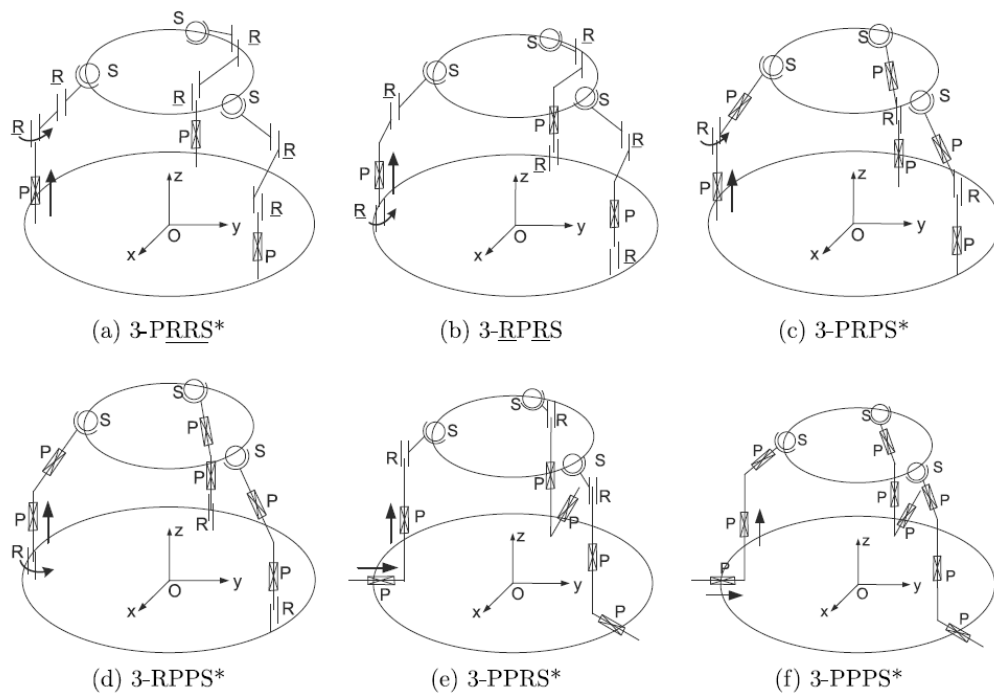


Figure 2.4 Parallel Mechanism with Decoupled Design (Jin et al., 2009)

Wang et al. (2003) carried out kinematic of parallel manipulator with elastic joints. Four bar linkages are used to give rotary input for the each leg. The mechanism is shown in Figure 2.5. They also calculated elastic deformation, elastic moment of the mechanism and presented displacement equations and velocity analysis. The effect of the torsional deformation in dynamics of the mechanism was also presented.

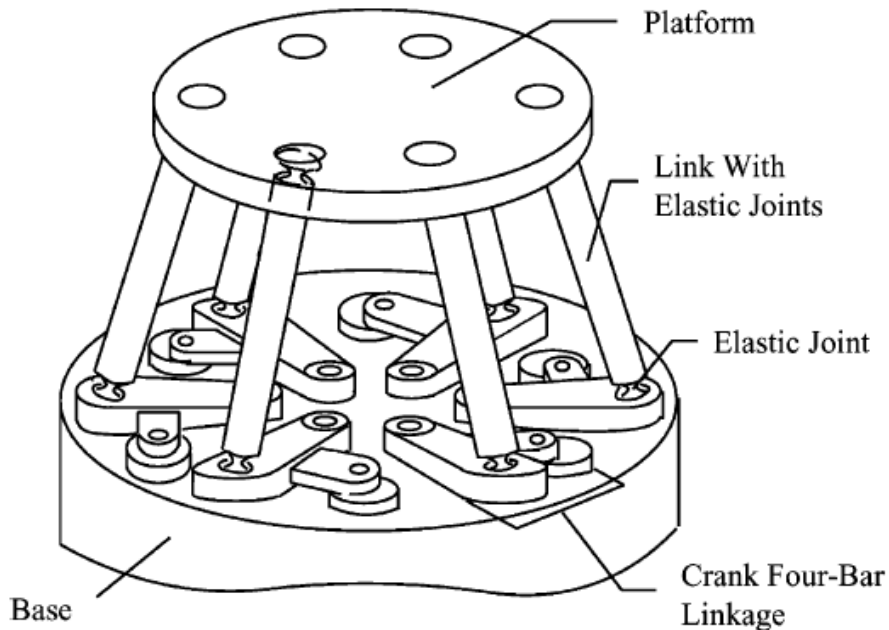


Figure 2.5 Design of the Parallel Manipulator with Elastic Joints (Wang et al., 2003)

2.3 Dynamic Analysis and Design of Stewart Platform

Dynamic analysis of the Stewart platform has been achieved by using Newton-Euler method, the principle of virtual work, screw theory and Lagrange method. Besides that, in literature, some researcher solved the Stewart platform dynamics by using Kane's equations. These methods are used to obtain equation of motion of the system. In dynamic analysis, moment and force values are found and necessary forces are specified for the actuators to rotate and translate of the moving platform with respect to desired trajectories in all directions. In this section, previous study related with dynamic analysis and design of Stewart platform is given.

Wang and Gosselin (1998) carried out dynamic analysis of the Gough-Stewart manipulator using principle of virtual work. They computed inertial forces and moments of the links according to translation and orientation of the moving platform. They concluded that the proposed approach is efficient and faster than Newton-Euler

approach. The forces and moments of the links are presented in Figure 2.6. Tsai (2000) also solved dynamics of the Stewart-Gough mechanism using principle of virtual work and derived equation of motion for the system. Additionally, dynamic simulation of the manipulator was performed for prescribed trajectories. Another similar study was proposed by Staicu (2011). Dynamics of the 6-6 Stewart Platform was carried out by using the principle of virtual work. Further, using Lagrange equation, the results of the presented method was compared. He concluded that the principle of the virtual work is more efficient due to eliminating the all internal forces.

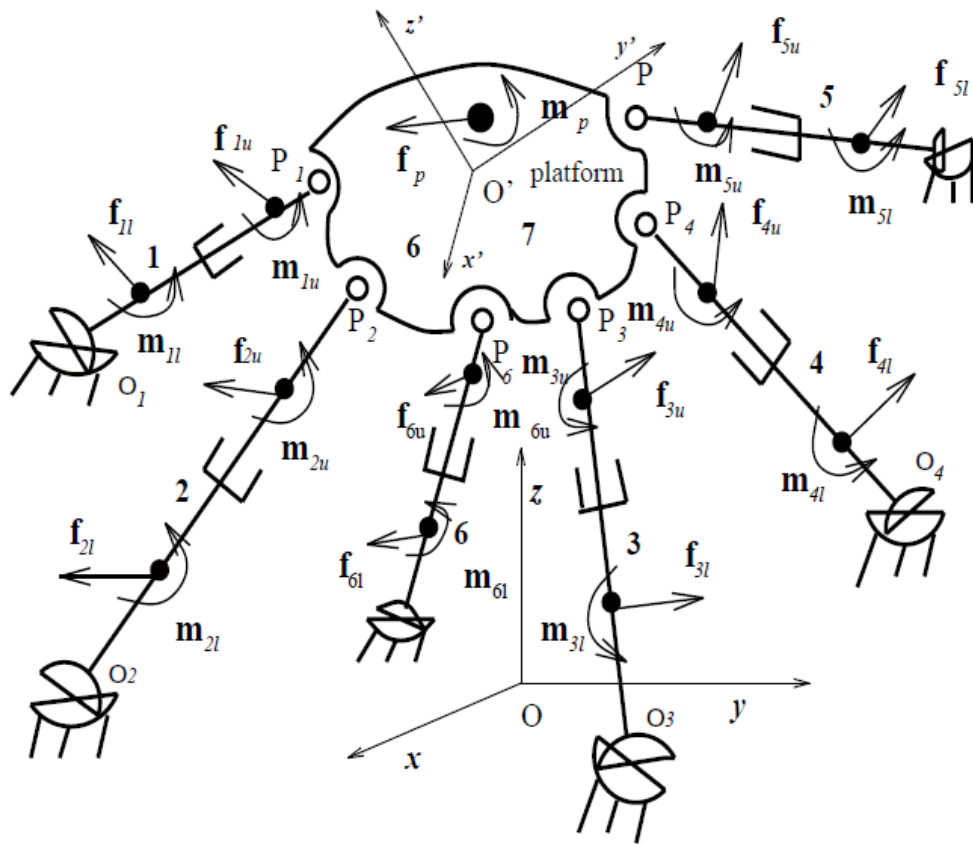


Figure 2.6 Inertial Forces and Moments Acting on the Links
(Wang and Gosselin,1998)

Guo and Li (2006) presented dynamic analysis of Stewart platform and implemented simulation of the system. They were firstly performed kinematic analysis and found rotational matrix of the mobile platform. Then, they used both Lagrange equation and Newton-Euler equation to obtain the equation of motion and taken into account the effect of the upper moving part actuators inertia of the system. Finally, driving forces have been obtained each leg and dynamic simulation was presented for two prescribed trajectories. Dasgupta and Mruthyunjaya (1998 a) were performed inverse dynamic

analysis by using Newton-Euler approach for a Stewart platform manipulator. They also improved fast algorithm to find actuator forces by applying appropriate elimination procedure. Besides that, the formulation was implemented in a MATLAB program to perform planned trajectories and to test of the system. Ben-Horin et al. (1998) performed kinematic, dynamic and construction of the planar actuated parallel robot which has six degree of freedom. Three planar motors were placed to perform the motion of the system. Each planar motor has moved in two coordinates. Further, they carried out experimental study to demonstrate the large work volume and high accuracy of this mechanism. The design of mechanism is shown in Figure 2.7. Brouwer et al. (2010) designed a six DOFs precision manipulator and modeled to use in a transmission electron microscope with small position sample. In this design has been used planar actuator which can move two coordinates. They analyzed and optimized their design by using flexible beam theory. Finally, they constructed the manipulator to verify and to test the design.

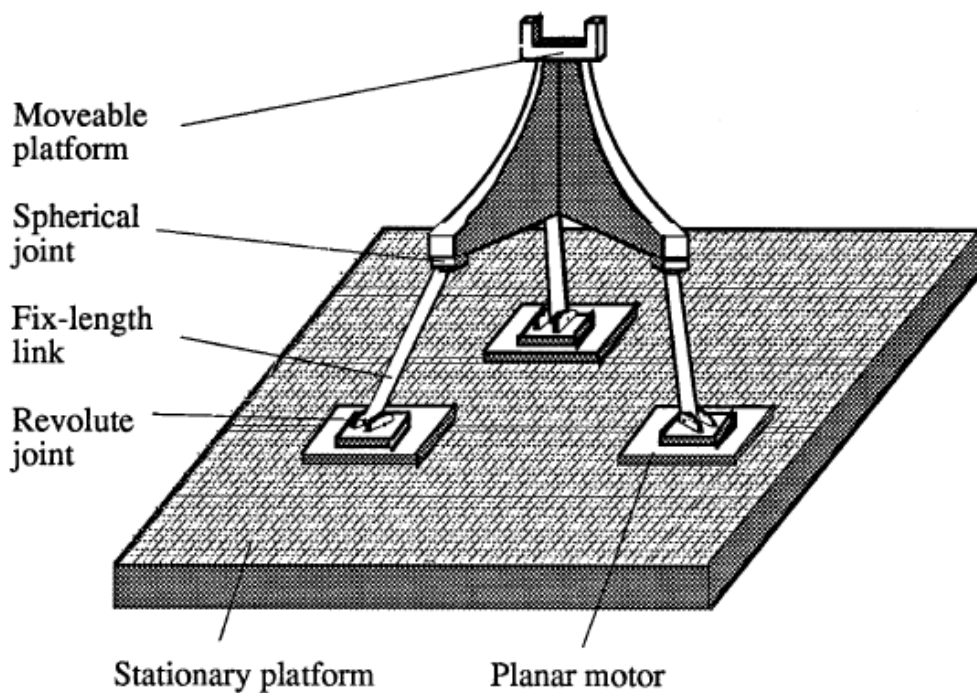


Figure 2.7 A Six DOF Parallel Mechanism with Three Planar Motors
(Ben-Horin et al., 1998)

Bai et al. (2006) carried out dynamic analysis of the Stewart Platform using Lagrange method to obtain equation of motion. They also improved a novel differentiation method which enables to solve the nonlinear equation of motion easily. Korayem and Shokri (2008) presented 6-UPS Stewart platform manipulator and developed a

computational method to attain the maximum Dynamic Load Carrying Capacity (DLCC). The formulation of the joint actuator and accuracy constraint were implemented to consider the maximum limit of load of the mechanism. They concluded that the simulation results of the 6-UPS Stewart Platform have demonstrated that the algorithm and model is acceptable. Yıldız et al. (2009) carried out dynamic modeling of the Stewart Platform by using a Bond Graph method and obtain dynamic equation of the system which has included gravity effect, linear motor dynamics and viscous friction. They also improved nonlinear state-space representation. Bond Graph method is shown in Figure 2.8.

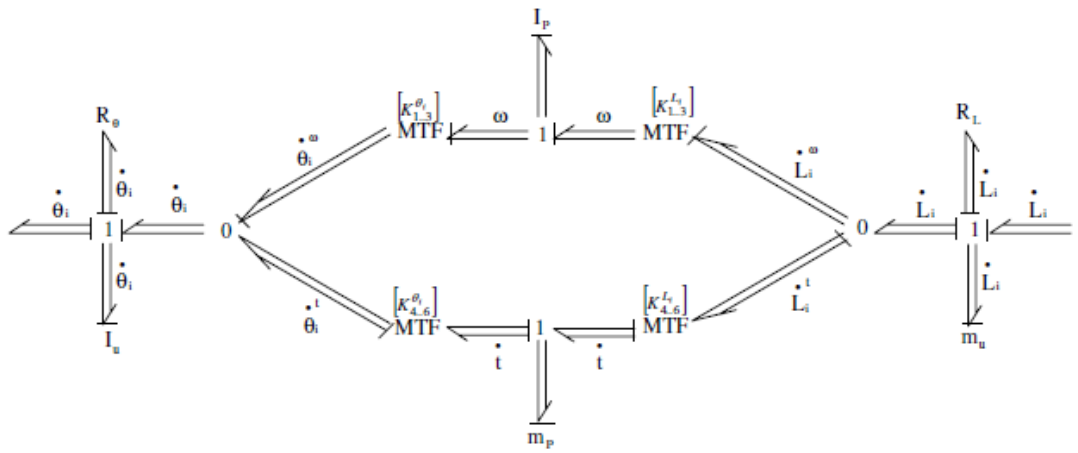


Figure 2.8 Bond Graph Model of System for a Leg (Yıldız et al., 2009)

In order to improve the dynamic performance of the Stewart platform, Evolutionary Multi-objective Optimization algorithms (EMO) was presented by Bangjun et al. (2012) Two EMO algorithms were improved which are NSGA-II and MOPSO-CD. The results of the comparison of algorithms have been observed that MOPSO-CD has given better result for the dynamic performance. Wu et al. (2012) carried out kinematic and inverse dynamic analysis of a 6-SPS parallel mechanism by using principle of Kane. Gravity and inertial forces were calculated for all links. Using inverse dynamic method, driving forces time-relation was evaluated. Bingul and Karahan (2012) focused on dynamic model and simulation of the Stewart Platform. Dynamic analysis was carried out using Lagrange equation. They also proposed Jacobian matrix by using two different ways. Eventually, using MATLAB-Simulink program, the dynamic equation of the Stewart Platform and actuator dynamics have been simulated and verified. Meng et al. (2010) carried out dynamic analysis of the Stewart platform using permanent magnet synchronous motor. In order to analyze the motion of the motor and

platform, they used Kane's equations and achieved equality of the driven torque and electromagnetic torque. The time varying inertia of the system is not needed to calculate thanks to this approach. The schematic view of the proposed actuator for the mechanism is shown in Figure 2.9.

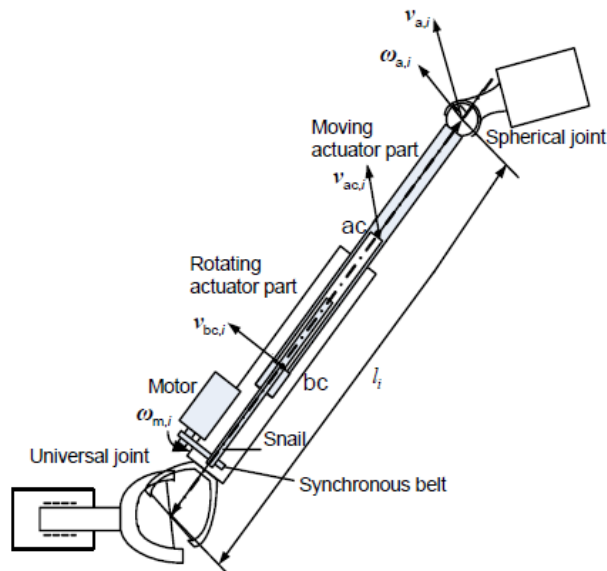


Figure 2.9 Schematic Representation of an Actuator (Meng et al., 2010)

2.4 Control and Simulation of Stewart Platform

Control is a very important concept to provide accuracy, robustness and precise position for a Stewart platform. Besides that, control helps to find optimum design and to obtain high dynamic performance. In order to perform control of the Stewart platform, PID controller, adaptive control, computed torque method are widely used. Simulation and control of the mechanism is generally carried out by using MATLAB/Simulink program. In this section, previous study related with control of the Stewart platform is presented.

Hsu and Fong (2001) carried out computed force feedback to control the Stewart platform. The orientation and position of the moving platform were computed and they also found static forces which act on the legs. Finally, using these forces in dynamic model, the feedback gains have been considered to control the leg lengths. Kallio (2002) carried out parallel piezohydraulic micromanipulator which has consisted of three prismatic actuators. He developed two inverse kinematic models to implement the position control of the manipulator. The first one is Hall sensor based control and

the second one is vision based control scheme. The results of this study, he concluded that two methods have same resolution and workspace. However, Hall sensor based control has high speed according to vision system. Lee et al. (2003) carried out position control of a Stewart platform by combining inverse dynamic control with approximate dynamics and H_∞ controller. H_∞ controller has enabled to compensate modelling error. It recovers the modeling error as disturbances. They concluded that H_∞ controller provides good tracking performance for the system. Huang et al. (2004) carried out dynamics of the Stewart Platform and developed sliding mode control to decrease the position error of the system. Furthermore, Lyapunov theory was used to provide stable controller design. Eventually, they observed that their design shown good performance.

Gewald (2006) carried out vibration control and investigated accurate motion of the tool center for a hexapod system with six degree of freedom. Feedforward-feedback controller was used to diminish vibration of the mechanism. Eventually, simulation of the system was implemented and observed that it has provided high performance and desired trajectories. Yingjie et al. (2006) employed feedback positioning control of the cable driven Gough-Stewart Platform for a large radio telescope. PID controller and optical sensor was utilized to achieve the system control. Finally, a simple analysis error was performed to verify the control system. Serrano et al. (2007) implemented control of the Stewart mechanism applied to a biomechanical system. They used fuzzy logic and PID controller to control the mechanism. In results of the simulation, which was performed in Simulink, was observed that fuzzy logic controller has provided some advantages according to PID controller as PID controller has high settling time and gain constant. Kizir et al. (2011) performed trajectory and position control of Stewart platform by using simple PID control for each motor. They designed the PID control in Simulink program and embedded real time controller. At the end of the study, they observed that the system position error is about 500nm. Davliakos and Papadopoulos (2008) were studied on feedback controller by using the equation of motion of the electrohydraulic Stewart Platform. The force and pressure feedback have not been required for this study due to independent load. Eventually, simulation was implemented with desired position input and observed good performance for controlling the system. Guo et al. (2008) carried out the cascade control of hydraulically driven 6 DOF parallel manipulators. The control was composed of two

parts which were inner and outer loop. Hydraulic dynamics and mechanical dynamics were separated each other and both of them controlled. Further, they implemented experimental study to show position tracking behavior of the mechanism. Wang (2008) carried out control of a Stewart manipulator by using feedback linearization. Control block was implemented using MATLAB/Simulink toolbox. To improve the controller, DynaFlexPro block was used. Finally, stabilizing inverse dynamic control was performed to improve the trajectory tracking of the mechanism.

Ömürlü and Yıldız (2011) carried out FBW (Fly-By-Wire) flight control unit with force feedback by using a 3-3 Stewart platform which was enabled to control single point of spatially moving vehicles (SMV) about three translational and three rotational axis. Omran and Kassem (2011) carried out optimal task space control design of Stewart platform to treat aircraft stall. To implement the system control, Genetic algorithms which included two optimization stages were developed. The first stage found optimal approximate model for the direct kinematics of the system. Other stage enabled to find optimal controller gains. They concluded that the proposed control design reduced the error for stall recovery maneuver. Control design of mechanism is shown in Figure 2.9.

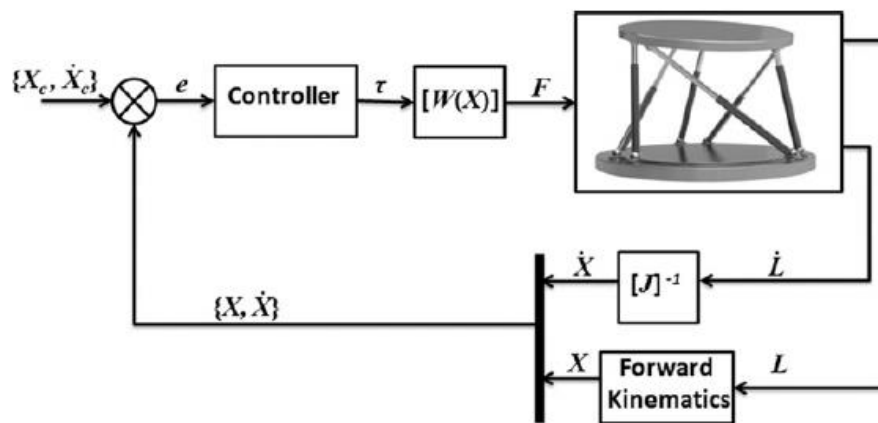


Figure 2.10 Stewart Platform Control Design (Omran and Kassem, 2011)

Yang et al. (2012) were presented decoupling controller for the six degree of freedom electro-hydraulic parallel manipulator. They implemented the feedback linearization theory to decrease coupling effects. The results of the experimental study showed that the presented controller developed the trajectory tracking performance. Akdağ et al. (2012) carried out simulation of the hexapod robot by using computer programming and obtained mathematical model to solve them. Solid modeling and assembly was

performed by using SolidWorks. CosmosMotion and CosmosWork were used to achieve rigid body dynamics and strength analysis, respectively. In addition, to control the actuator motion, Adlink module was developed. The manipulator was fabricated to test and verify the simulation results. Inner and Kucuk (2013) improved simulation tool (STEWSIM) to design the Stewart Platform. Using this program, different types of Stewart platform can be design, such as 3x3, 3x4...6x6. The STEWSIM has enabled to perform kinematic, dynamic, dexterity and workspace analysis for different position inputs of the moving platform. Furthermore, it has provided graphical representation for these analyses.

2.5 Workspace and Singularity Analysis

Workspace and singularity is the most important problem for the Stewart platform due to kinematics and geometrical constraints. Workspace is defined as the area within the reach of the center of gravity of mobile platform of the Stewart mechanism. The size of workspace of the parallel manipulators is determined according to application areas. For example, flight simulator requires large workspace volume according to medical robots.

Singularity is a point of the system which cannot be controlled due to the fact that the mechanism gains one or more degrees of freedom in this point. In order to avoid kinematic singularity, redundancy actuation can be used. Moreover, redundancy increase workspace and improve dexterity. Many researchers have studied to solve the workspace and singularity problem. In this section, previous study of the workspace and singularity analysis of the Stewart platform is presented.

Ma and Angeles (1991) classified the singularities in three groups for the parallel robots that are architecture, configuration, and formulation singularities. They focused on architecture singularity because it has contained all the workspace contrarily other groups. This study provided for the optimum design of the parallel robot. Takeda and Funabashi (1996) suggested a new method to solve the singularity problem for the parallel manipulator. They specified the location of the singular point of the mechanism which has depended on pressure angles. Then, the singular point curve was described which composed of singular points. This suggested method was applied to Stewart platform to determine the workspace of the mechanism. Dasgupta and Mruthyunjaya (1998 b) developed an algorithm which indicated the impossibility of a

valid path within the workspace of the system in order to prevent singularities and ill conditioning of the Stewart platform. They concluded that the algorithm was found reliable since it performed the planned paths. Kapur et al. (2007) carried out to overcome singularity problem for the Stewart Platform manipulators. They presented flexural joint which enables to prevent the dramatically gain in degree of freedom of the system. They concluded that flexural joint was effective method for parallel mechanism. Hua et al. (2007) proposed optimal design to perform safety mechanism for the Stewart Platform. In this study, they used homotopy method to solve the singular points of the extreme poses. Moreover, they developed the genetic algorithm to provide the design safety and optimization of the manipulator. Besides that, similar study was presented by Gao et al. (2010). Artificial intelligence approach was utilized to perform the design optimization of parallel manipulator. They performed analytical solution of the parallel manipulator stiffness and dexterity by using Levenberg-Marquardt algorithm. Eventually, they carried out the simulation of the system to show the effectiveness of this method. Lin et al. (2008) improved the 6-DOFs parallel robots which has included three legs and actuated three linear DC motor and AC servo motor. They developed genetic algorithms and proposed optimal path planning, which has depended on a DNA evolutionary computing algorithms, to prevent singularity within workspace of the system. Mishra and Omkar (2011) carried out genetic algorithm, PSO (Particle Swarm Optimization), QPSO (Quantum Particle Swarm Optimization), WQPSO (Weighted Particle Swarm Optimization) to eliminate the singularity problem for the Stewart Platform. Then, these algorithms were compared each other and observed that WQPSO indicated good performance than other algorithms.

Merlet (1995) presented geometrical approach to calculate the reachable workspace for the parallel manipulators. He also developed algorithm to execute this assignment efficiently. All constraints limiting of workspace has been taken into account by means of this algorithm. Ay et al. (2012) also carried out a new geometrical method to determine the reachable workspace for the 6-3 Stewart Platform manipulator. This method has provided to determine location of the system according to possible legs configurations. Güneri (2007) developed computer code to analysis workspace of Stewart platform by using MATLAB program. She also specified geometric and kinematic effects on the workspace for different position of mobile platform. Further,

MATLAB code was developed to perform the singularity analysis of the system by computing the Jacobian matrix. Abedinnasab et al. (2012) carried out redundant actuation for Gough-Stewart Platform. They compared parallel mechanism for redundant and non-redundant situations. The results of the study showed that redundancy has eliminated singularity and developed workspace, dexterity and increased sensitivity of the system. Toz and Kucuk (2013) designed an asymmetric six DOF Stewart- Gough Platform with ten different linear actuators (AMEDLAL). They carried out to increase volume of the dexterous workspace of the system. In this study, Particle Swarm Optimization was used to perform kinematic constraints. To achieve the dexterous of the system, Minimum Singular Value of homogenized Jacobian matrix was utilized. Eventually, proposed system showed better performance according to conventional Stewart Platform.

2.6 Motivation of Study

According to previous studies, it is observed that kinematic analysis, dynamic analysis and control of the Stewart platform was employed using hydraulic actuator, rotary DC motor, planar motor. However, linear motor has not been widely used. Further, the rotary part of the lower leg inertia has not been commonly considered while dynamic analysis is implemented.

In this thesis, Stewart platform has been analyzed using linear motors and rotary part of the leg inertia has computed. Finally, system control has been carried out in MATLAB/Simulink. In future works, the focus may be on workspace and singularity problem by improving new algorithms.

CHAPTER 3

KINEMATIC ANALYSIS OF THE STEWART PLATFORM

Kinematic analysis is the first step for designing a robot. Kinematic analysis comprises position, velocity and acceleration analysis of the system but force and torque are not taken into account. Two types of kinematic analysis methods (inverse and forward kinematic) are commonly used to perform kinematic analysis of any parallel manipulators in the literature. Forward kinematic analysis is finding of the moving platform position and orientation by using the leg vector of the mechanism. However, it is difficult to apply the parallel manipulators because the analysis results in complex equations and high non-linearity. Unlike the forward kinematics, the leg vector of the mechanism is determined by using the position and orientation of the moving platform in inverse kinematics. Thus, it can be performed easily because it doesn't include any complex equations.

In this chapter, kinematic analysis is composed of three sections. Rotation matrix is obtained in the first section and then, angular velocity of the moving platform is found in the second section. Finally, inverse position and velocity analysis of the system are carried out. Inverse Jacobian matrix is obtained to find leg velocity in the third section.

3.1 Rotational Matrix

To determine the motion of the moving platform, three translational and three rotational coordinates should be specified. Euler angles are used to obtain the rotation matrix. The rotations about z, y and x axes are carried out respectively.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R_P^B * \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \quad (3.1)$$

Rotate an angle γ about z axis,

$$R_z = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

Rotate an angle β about y axis

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

Rotate an angle α about x axis,

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \quad (3.4)$$

The rotation matrix of the moving platform relative to the base platform (R_p^B) is obtained as;

$$\begin{aligned} R_p^B &= R_z(\gamma) * R_y(\beta) * R_x(\alpha) \\ R_p^B &= \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \\ R_p^B &= \begin{bmatrix} c\beta c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ c\beta s\gamma & s\alpha s\beta s\gamma + c\alpha c\gamma & c\alpha s\beta s\gamma - s\alpha c\gamma \\ -s\beta & s\alpha c\beta & c\alpha c\beta \end{bmatrix} = \begin{bmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ a_z & b_z & c_z \end{bmatrix} \quad (3.5) \end{aligned}$$

The transformation matrix T_p^B which is a vector it includes rotational and translational motions with respect to the base frame.

$$T_p^B = \begin{bmatrix} a_x & b_x & c_x & t_x \\ a_y & b_y & c_y & t_y \\ a_z & b_z & c_z & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Generalized coordinate position and velocity vector of the moving platform of the Stewart mechanism can be expressed as follows.

$$q = [t_x \quad t_y \quad t_z \quad \alpha \quad \beta \quad \gamma]^T$$

$$\dot{q} = [\dot{t}_x \quad \dot{t}_y \quad \dot{t}_z \quad \dot{\alpha} \quad \dot{\beta} \quad \dot{\gamma}]^T$$

Generalized coordinate vector can be separated in two parts which are translational vector $t = [t_x \quad t_y \quad t_z]^T$ and the rotational angles $[\alpha \quad \beta \quad \gamma]^T$ which are defined in rotational matrix R_p^B .

3.2 Angular Velocity of the Moving Platform

The moving platform initially rotates about z axis, then rotates about y axis and finally rotates about x axis. So, the first axis includes two additional rotations, second axis includes one additional rotation and third axis has no additional rotation.

The angular velocity (w) of the moving platform can be transformed to the base frame using Euler angles as follows.

$$\begin{aligned}
 w &= R_x(\alpha) * R_y(\beta) * \begin{bmatrix} 0 \\ 0 \\ \dot{\gamma} \end{bmatrix} + R_x(\alpha) * \begin{bmatrix} 0 \\ \dot{\beta} \\ 0 \end{bmatrix} + \begin{bmatrix} \dot{\alpha} \\ 0 \\ 0 \end{bmatrix} \\
 w &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\gamma} \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\beta} \\ 0 \end{bmatrix} + \begin{bmatrix} \dot{\alpha} \\ 0 \\ 0 \end{bmatrix} \\
 w &= \begin{bmatrix} 1 & 0 & s\beta \\ 0 & c\alpha & -s\alpha c\beta \\ 0 & s\alpha & c\alpha c\beta \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \tag{3.6}
 \end{aligned}$$

The acceleration of the moving platform can be obtained by differentiating of angular velocity with respect to time.

$$\dot{w} = \begin{bmatrix} 1 & 0 & s\beta \\ 0 & c\alpha & -s\alpha c\beta \\ 0 & s\alpha & c\alpha c\beta \end{bmatrix} \begin{bmatrix} \ddot{\alpha} \\ \ddot{\beta} \\ \ddot{\gamma} \end{bmatrix} + \begin{bmatrix} 0 & 0 & \dot{\beta}c\beta \\ 0 & -\dot{\alpha}s\alpha & -\dot{\alpha}c\alpha c\beta + s\alpha\dot{\beta}s\beta \\ 0 & \dot{\alpha}c\alpha & -\dot{\alpha}s\alpha c\beta - c\alpha\dot{\beta}s\beta \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \tag{3.7}$$

3.3 Inverse Kinematics

Inverse kinematic method is commonly used since it provides facilities to solve the kinematics of the parallel manipulators. In inverse kinematic analysis, length of the legs are computed according to platform position and rotation with respect to base platform. The motion of the platform is specified for six coordinates (3 rotation and 3 translation). There are six unknowns ($l_1, l_2, l_3, l_4, l_5, l_6$).

3.3.1 Inverse Position Analysis of the Mechanism

A closed-loop of one leg which is shown in Figure 3.1 is used for determining the leg lengths according to position of the moving platform. The mechanism has two coordinate systems; placed mass center of the moving and the base platform see Figure 3.2. The first one is moving platform P (X_p, Y_p, Z_p), and the second one is base platform B (X, Y, Z).

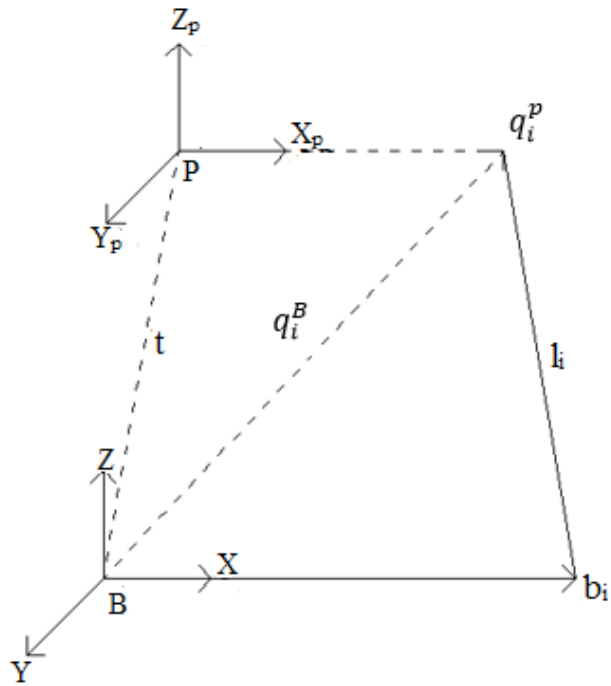


Figure 3.1 Closed-Loop of one Leg

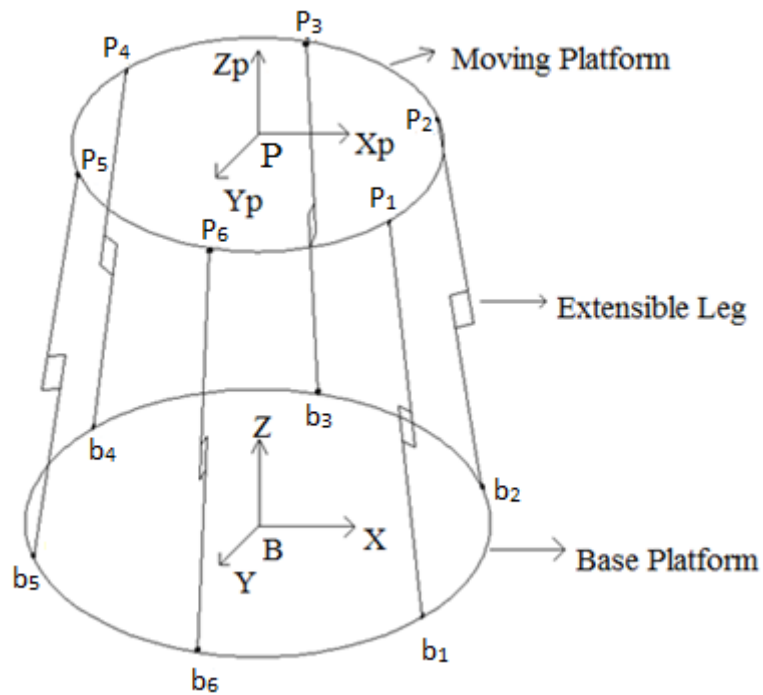


Figure 3.2 Schematic Representation of the Stewart Platform

Position vector of the i th upper junction coordinates with respect to moving platform q_i^p and position vector of the i th lower junction coordinates b_i can be described as

$$b_i = \begin{bmatrix} r_b * c_i \\ r_b * s_i \\ 0 \end{bmatrix} \quad \text{and} \quad q_i^p = \begin{bmatrix} r_p * c_i \\ r_p * s_i \\ 0 \end{bmatrix}$$

The leg vector with respect to base platform can be obtained from closed-form representation as follows;

$$L_i = q_i^B - b_i \quad (3.8)$$

The position vector of i th upper junction point with respect to the base frame q_i^B is described as

$$\begin{aligned} q_i^B &= T_P^B * q_i^p \\ q_i^B &= t + R_P^B * q_i^p \end{aligned} \quad (3.9)$$

Then, from equations (3.8) and (3.9), the following equation can be written.

$$L_i = T_P^B * q_i^p - b_i = t + R_P^B * q_i^p - b_i \quad (3.10)$$

Its matrix form is obtained as follows.

$$\begin{aligned} L_i &= \begin{bmatrix} a_x & b_x & c_x & t_x \\ a_y & b_y & c_y & t_y \\ a_z & b_z & c_z & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} q_{ix}^p \\ q_{iy}^p \\ q_{iz}^p \\ 1 \end{bmatrix} - \begin{bmatrix} b_{ix} \\ b_{iy} \\ b_{iz} \\ 1 \end{bmatrix} \\ L_i &= \begin{bmatrix} a_x * q_{ix}^p + b_x * q_{iy}^p + c_x * q_{iz}^p + t_x - b_{ix} \\ a_y * q_{ix}^p + b_y * q_{iy}^p + c_y * q_{iz}^p + t_y - b_{iy} \\ a_z * q_{ix}^p + b_z * q_{iy}^p + c_z * q_{iz}^p + t_z - b_{iz} \end{bmatrix} \\ L_i &= \begin{bmatrix} a_x * r_p * c_i + b_x * r_p * s_i + t_x - r_b * c_i \\ a_y * r_p * c_i + b_y * r_p * s_i + t_y - r_b * s_i \\ a_z * r_p * c_i + b_z * r_p * s_i + t_z \end{bmatrix} \end{aligned}$$

Then, the length of the i th leg is acquired as follows.

$$\begin{aligned} l_i^2 &= (a_x * r_p * c_i + b_x * r_p * s_i + t_x - r_b * c_i)^2 \\ &\quad + (a_y * r_p * c_i + b_y * r_p * s_i + t_y - r_b * s_i)^2 \\ &\quad + (a_z * r_p * c_i + b_z * r_p * s_i + t_z)^2 \end{aligned} \quad (3.11)$$

3.3.2 Inverse Velocity Analysis of the Mechanism

In this section, inverse velocity analysis of the Stewart platform is performed by inverse Jacobian matrix which enables to find velocity of the leg using moving platform velocity. Inverse Jacobian matrix describes relation between velocity of the platform and leg velocity. Equation 3.12 shows the leg velocity, where J^{-1} is the Inverse Jacobian matrix and \dot{P} is the moving platform velocity.

$$\dot{l} = J^{-1} * \dot{P} \quad (3.12)$$

$$\dot{P} = [\dot{t} \quad w]^T$$

$$u_i = \frac{L_i}{l_i}$$

$$L_i = u_i * l_i \quad (3.13)$$

$$u_i * l_i = R_P^B * q_i^p + t - b_i \quad (3.14)$$

Then, Equation (3.14) is differentiated with respect to time, the following equation is obtained.

$$\dot{l}_i * u_i + l_i * \dot{u}_i = R_P^B * q_i^p \times w + \dot{t} - \dot{b}_i \quad (3.15)$$

Since lower junction point is constant; $\dot{b}_i = [0 \quad 0 \quad 0]^T$

$$u_i * \dot{u}_i = u_i * (w \times u_i) = w * (u_i \times u_i) = 0$$

Since unit vector, $u_i * \dot{u}_i = 0$, $u_i * u_i = 1$

Then, the velocity of the i th leg is obtained as follows

$$\dot{l}_i = (R_P^B q_i^p \times u_i)^T * w + u_i^T * \dot{t} \quad (3.16)$$

$$\dot{l}_i = [u_i^T \quad (R_P^B q_i^p \times u_i)^T] \dot{P} \quad (3.17)$$

Jacobian matrix can be acquired as follows

$$J^{-1} = \begin{bmatrix} u_1^T & (R_P^B q_1^p \times u_1)^T \\ u_2^T & (R_P^B q_2^p \times u_2)^T \\ u_3^T & (R_P^B q_3^p \times u_3)^T \\ u_4^T & (R_P^B q_4^p \times u_4)^T \\ u_5^T & (R_P^B q_5^p \times u_5)^T \\ u_6^T & (R_P^B q_6^p \times u_6)^T \end{bmatrix}_{6 \times 6} \quad (3.18)$$

If the generalized coordinate are $\dot{q} = [\dot{t}_x \quad \dot{t}_y \quad \dot{t}_z \quad \dot{\alpha} \quad \dot{\beta} \quad \dot{\gamma}]^T$ then, Jacobian matrix can be obtained as;

$$\dot{l} = J_1^{-1} * \dot{q} \quad (3.19)$$

$$J_1^{-1} = \begin{bmatrix} u_1^T & (R_P^B q_1^p \times u_1)^T \\ u_2^T & (R_P^B q_2^p \times u_2)^T \\ u_3^T & (R_P^B q_3^p \times u_3)^T \\ u_4^T & (R_P^B q_4^p \times u_4)^T \\ u_5^T & (R_P^B q_5^p \times u_5)^T \\ u_6^T & (R_P^B q_6^p \times u_6)^T \end{bmatrix}_{6 \times 6} * \begin{bmatrix} I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 1 & 0 & s\beta \\ & 0 & c\alpha & -s\alpha c\beta \\ & 0 & s\alpha & c\alpha c\beta \end{bmatrix}_{6 \times 6} \quad (3.20)$$

The equation (3.16) can be expressed as follows

$$\dot{l}_i = u_i^T * \dot{q}_i^B$$

The velocity and acceleration $(\dot{q}_i^B, \ddot{q}_i^B)$ of the upper junction point with reference to base frame is obtained as;

$$\dot{q}_i^B = [I \quad R_P^B (\tilde{q}_i^p)^T (R_P^B)^T] \dot{P} \quad (3.21)$$

$$\ddot{q}_i^B = [I \quad R_P^B (\tilde{q}_i^p)^T (R_P^B)^T] \ddot{P} + \tilde{w}^2 R_P^B q_i^p \quad (3.22)$$

Where \tilde{q}_i^p and \tilde{w} symbolizes the skew symmetric matrix associated with the vector $u = [a_x \quad a_y \quad a_z]^T$

$$\tilde{a} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}$$

Note that \tilde{a}_b and \tilde{a}_p are arbitrary vector matrix with respect to base and platform frame, respectively.

$$\tilde{a}_b = R_P^B \tilde{a}_p (R_P^B)^T$$

$$a = b \times c = \tilde{b}c \text{ (Skew symmetric matrix properties)}$$

The actuator consists of two parts of which are rotating and moving parts. Rotating part is the lower part, and moving part is the upper part of the leg. Rotating part of the leg is attached to the base platform, and moving part of the leg is attached to the moving platform. These two parts are also fastened with each other with prismatic joint. The angular velocity of the actuator and linear velocity of rotating, and moving parts can be obtained as follows. Model of Stewart platform is presented in Figure 3.3.

Angular velocity of the actuator,

$$w_{act} = \frac{u_i \times \dot{q}_i^B}{l_i} = \frac{\tilde{u}_i \dot{q}_i^B}{l_i} \quad (3.23)$$

Linear velocity of moving part of the leg,

$$v_m = \dot{q}_i^B + w_{act} \times (-d_p u_i) = \dot{q}_i^B + \frac{\tilde{u}_i \dot{q}_i^B}{l_i} \times (-d_p u_i)$$

$$v_m = \dot{q}_i^B + \frac{d_p \tilde{u}_i^2 \dot{q}_i^B}{l_i} = \left(I + \frac{d_p \tilde{u}_i^2}{l_i} \right) \dot{q}_i^B \quad (3.24)$$

Linear velocity of rotating part of the leg,

$$v_r = w_{act} \times d_b u_i = \frac{\tilde{u}_i \dot{q}_i^B}{l_i} \times d_b u_i = -d_b u_i \times \frac{\tilde{u}_i \dot{q}_i^B}{l_i}$$

$$v_r = \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right) \dot{q}_i^B \quad (3.25)$$

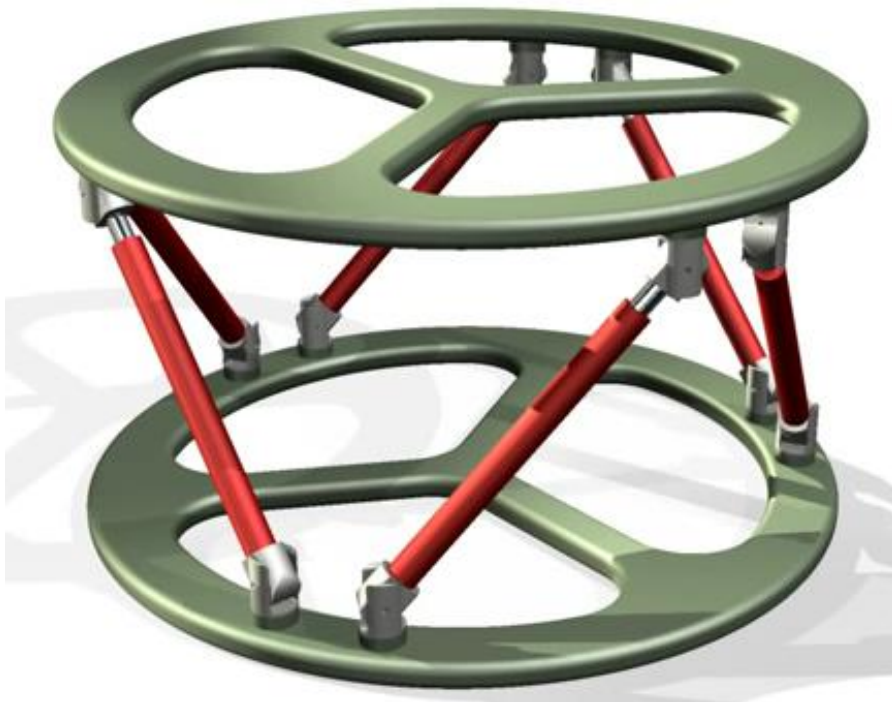


Figure 3.3 Model of Stewart Platform (Williams II, 2015)

CHAPTER 4

DYNAMIC ANALYSIS OF THE STEWART PLATFORM

In this section, dynamic analysis of the Stewart platform is carried out by taking the studies of Guo and Li (2006) as a reference. Initially, the constraint force F_c of the leg is obtained by using Lagrange formulation. Then, the platform and leg dynamics are combined by using Newton Euler method. Eventually, equation of motion of the Stewart platform is obtained, and dynamic simulation of the system is performed according to desired trajectories of the moving platform. Linear motor dynamic is also presented at the end of this chapter.

4.1 Dynamic Analysis of a Link

The legs of the system are exposed to some external forces which are gravitational forces, constraint forces and driving forces. These forces are shown on one leg of the system in Figure 4.1.

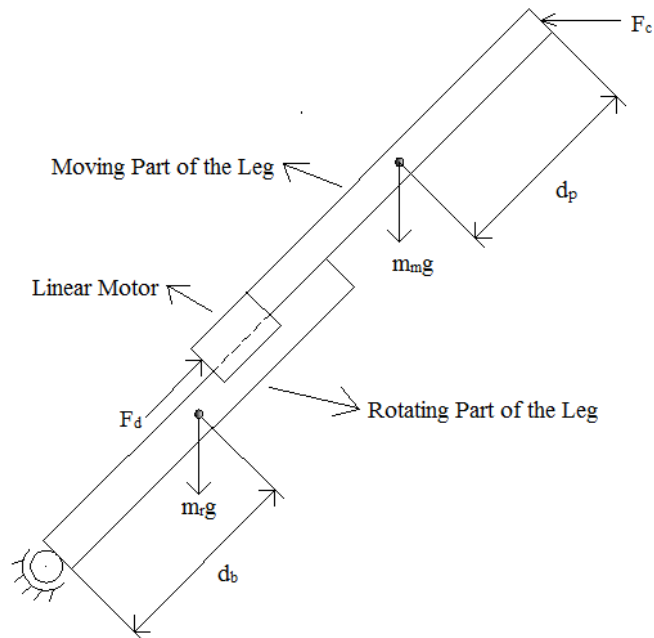


Figure 4.1 Force Analysis of the Actuator

Generally, dynamic effect of the rotating part of the leg is ignored because the moving part has higher dynamic effect than the rotating part. Since, linear motor is mounted on the rotating part so it increases mass and velocity of the rotating part therefore the inertial effects of the rotating part of the leg are taken into account also.

Lagrange equation of the link is described as follows.

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_i^B} \right) - \frac{\partial T}{\partial q_i^B} = \tau \quad (4.1)$$

Kinetic energy equation which includes linear velocity and angular velocity of the leg and it can be written as follows.

$$T = \frac{1}{2} v_m^T m_m v_m + \frac{1}{2} v_r^T m_r v_r + \frac{1}{2} w_{act}^T (I_r + I_m) w_{act} \quad (4.2)$$

Substituting equations (3.23), (3.24), (3.25) into the equation (4.2), kinetic energy equation can be rewritten as follows.

$$T = \frac{1}{2} (\dot{q}_i^B)^T \left[\left(I + \frac{d_p \tilde{u}_i^2}{l_i} \right)^T m_m \left(I + \frac{d_p \tilde{u}_i^2}{l_i} \right) + \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right)^T m_r \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right) + \frac{\tilde{u}_i^T \tilde{u}_i}{l_i^2} (I_r + I_m) \right] \dot{q}_i^B$$

To simplify the solution of the kinetic energy equation, it can be separated into three parts which are M_1 , M_2 , M_3 , as follows.

$$T = \frac{1}{2} (\dot{q}_i^B)^T (M_1 + M_2 + M_3) \dot{q}_i^B$$

$$M_1 = \left(I + \frac{d_p \tilde{u}_i^2}{l_i} \right)^T m_m \left(I + \frac{d_p \tilde{u}_i^2}{l_i} \right) \quad (4.3)$$

$$M_2 = \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right)^T m_r \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right) \quad (4.4)$$

$$M_3 = \frac{\tilde{u}_i^T \tilde{u}_i}{l_i^2} (I_r + I_m) \quad (4.5)$$

Lagrange equation is applied to find the constraint forces by using the following equations.

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i^B}\right) &= \frac{d}{dt}\left((M_1 + M_2 + M_3)\dot{q}_i^B\right) \\ &= \frac{d(M_1 + M_2 + M_3)}{dt}\dot{q}_i^B + (M_1 + M_2 + M_3)\ddot{q}_i^B\end{aligned}\quad (4.6)$$

The equations are presented in Appendix A which are used to derive M_1 , M_2 , M_3 , with respect to time.

$$\begin{aligned}\frac{dM_1}{dt} &= \frac{2m_m d_p}{l_i^3} (u_i (\dot{q}_i^B)^T \tilde{u}_i^T \tilde{u}_i + u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i + \tilde{u}_i^T \tilde{u}_i \dot{q}_i^B u_i^T) \\ &\quad - \frac{m_m d_p^2}{l_i^3} (2u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i + u_i (\dot{q}_i^B)^T \tilde{u}_i^T \tilde{u}_i + \tilde{u}_i^T \tilde{u}_i \dot{q}_i^B u_i^T)\end{aligned}\quad (4.7)$$

$$\frac{dM_2}{dt} = -\frac{m_r d_b^2}{l_i^3} (2u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i + u_i (\dot{q}_i^B)^T \tilde{u}_i^T \tilde{u}_i + \tilde{u}_i^T \tilde{u}_i \dot{q}_i^B u_i^T) \quad (4.8)$$

$$\frac{dM_3}{dt} = -\frac{I_r + I_m}{l_i^3} (\tilde{u}_i^T \tilde{u}_i \dot{q}_i^B u_i^T + u_i (\dot{q}_i^B)^T \tilde{u}_i^T \tilde{u}_i + 2u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i) \quad (4.9)$$

The partial derivative of the kinetic energy with respect to generalized coordinate is derived as follows.

$$\begin{aligned}\frac{\partial T}{\partial q_i^B} &= \frac{\partial}{\partial q_i^B} \left[\frac{1}{2} (\dot{q}_i^B)^T (M_1 + M_2 + M_3) \dot{q}_i^B \right] \\ &= \frac{\partial}{\partial q_i^B} \left[\frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B \left(m_m \left[\left(1 - \frac{d_p}{l_i}\right)^2 \mathbf{I} + 2\frac{d_p}{l_i} u_i u_i^T - \left(\frac{d_p}{l_i}\right)^2 u_i u_i^T \right] \right. \right. \\ &\quad \left. \left. + m_r \left(\frac{d_b}{l_i}\right)^2 (\mathbf{I} - u_i u_i^T) + \frac{(1 - u_i u_i^T)(I_r + I_m)}{l_i^2} \right) \right]\end{aligned}\quad (4.10)$$

The equation M_1 is derived with respect to the generalized coordinate q_i^B by separating in three parts as follows.

$$\begin{aligned}\frac{\partial}{\partial q_i^B} \left[\frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B m_m \left(1 - \frac{d_p}{l_i}\right)^2 \mathbf{I} \right] &= \frac{\partial}{\partial q_i^B} \left[\frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B m_m \left(1 - 2\frac{d_p}{l_i} \mathbf{I} + \frac{d_p^2}{l_i^2} \mathbf{I}\right) \right] \\ &= -(\dot{q}_i^B)^T \dot{q}_i^B d_p m_m \frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i}\right) \mathbf{I} + \frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B d_p^2 m_m \frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i^2}\right) \mathbf{I}\end{aligned}\quad (4.10a)$$

$$\begin{aligned}\frac{\partial}{\partial q_i^B} \left[\frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B m_m \left(2\frac{d_p}{l_i} u u^T\right) \right] &= (\dot{q}_i^B)^T \dot{q}_i^B d_p m_m \frac{\partial}{\partial q_i^B} \left(\frac{L_i L_i^T}{l_i^3}\right) \\ &= d_p m_m \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T L_i (\dot{q}_i^B)^T L_i}{l_i^3}\right)\end{aligned}\quad (4.10b)$$

$$\begin{aligned}
& \frac{\partial}{\partial q_i^B} \left[-\frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B m_m \left(\frac{d_p}{l_i} \right)^2 u_i u_i^T \right] \\
&= -\frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B d_p^2 m_m \frac{\partial}{\partial q_i^B} \left[\left(\frac{1}{l_i^2} \right) \left(\frac{L_i L_i^T}{l_i^2} \right) \right]
\end{aligned} \tag{4.10c}$$

Derivative of M_2 is found with respect to generalized coordinate q_i^B as follows.

$$\begin{aligned}
& \frac{\partial}{\partial q_i^B} \left[\frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B m_r \left(\frac{d_b}{l_i} \right)^2 (I - u_i u_i^T) \right] \\
&= \frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B d_b^2 m_r \frac{\partial}{\partial q_i^B} \left[\left(\frac{1}{l_i^2} \right) \left(I - \frac{L_i L_i^T}{l_i^2} \right) \right]
\end{aligned} \tag{4.10d}$$

Derivative of M_3 is carried out with respect to generalized coordinate q_i^B as follows.

$$\begin{aligned}
& \frac{\partial}{\partial q_i^B} \left[\frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B (I_r + I_m) \frac{(I - u_i u_i^T)}{l_i^2} \right] \\
&= \frac{1}{2} (\dot{q}_i^B)^T \dot{q}_i^B (I_r + I_m) \frac{\partial}{\partial q_i^B} \left[\left(\frac{1}{l_i^2} \right) \left(I - \frac{L_i L_i^T}{l_i^2} \right) \right]
\end{aligned} \tag{4.10e}$$

Equations (4.10a-4.10e) are used and by taking derivative of the kinetic energy according to the generalized coordinate.

$$\begin{aligned}
\frac{\partial T}{\partial q_i^B} &= d_p m_m \left(-(\dot{q}_i^B)^T \dot{q}_i^B \frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i} \right) + \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T L_i (\dot{q}_i^B)^T L_i}{l_i^3} \right) \right. \\
&\quad \left. + \frac{1}{2} (d_p^2 m_m + d_b^2 m_r + I_r + I_m) \left[(\dot{q}_i^B)^T \dot{q}_i^B \frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i^2} \right) \right. \right. \\
&\quad \left. \left. - \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T L_i}{l_i^2} \right)^2 \right] \right) \\
\frac{\partial T}{\partial q_i^B} &= d_p m_m \left(-(\dot{q}_i^B)^T \dot{q}_i^B \frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i} \right) + \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T L_i (\dot{q}_i^B)^T L_i}{l_i^3} \right) \right. \\
&\quad \left. + \frac{1}{2} (d_p^2 m_m + d_b^2 m_r + I_r + I_m) \left[(\dot{q}_i^B)^T \dot{q}_i^B \frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i^2} \right) \right. \right. \\
&\quad \left. \left. - 2 \frac{(\dot{q}_i^B)^T L_i}{l_i^2} \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T L_i}{l_i^2} \right) \right] \right)
\end{aligned} \tag{4.11}$$

To simplify the derivative of the equation (4.11), some expressions are presented in Appendix B. Then, using Appendix B, the equation (4.12) can be obtained.

$$\begin{aligned}
\frac{\partial T}{\partial q_i^B} &= \frac{d_p m_m}{l_i^2} (u_i (\dot{q}_i^B)^T \dot{q}_i^B + 2u_i^T \dot{q}_i^B \dot{q}_i^B - 3u_i (\dot{q}_i^B)^T u_i u_i^T \dot{q}_i^B) \\
&\quad - \frac{(d_p^2 m_m + d_b^2 m_r)}{l_i^3} (u_i (\dot{q}_i^B)^T \dot{q}_i^B + \dot{q}_i^B (\dot{q}_i^B)^T u_i - 2u_i u_i^T \dot{q}_i^B (\dot{q}_i^B)^T u_i) \\
&\quad - \frac{I_r + I_m}{l_i^3} (u_i^T \dot{q}_i^B \dot{q}_i^B + u_i (\dot{q}_i^B)^T \dot{q}_i^B - 2u_i (\dot{q}_i^B)^T u_i u_i^T \dot{q}_i^B) \quad (4.12)
\end{aligned}$$

The driving force that is produced by linear motor and its derivative are carried out by using the principle of virtual work (δW) which is based on equality of the work for two systems. The other external force is the gravitational force which occurs by virtue of mass of the lower and upper part of the link.

$$\delta W = f_d * \delta l_i = f_d u_i^T \delta q_i^B = (u_i f_d)^T \delta q_i^B \quad (4.13)$$

Then, the external force is F_d driving force which is obtained as follows

$$F_d = u_i f_d \quad (4.14)$$

The second forces are obtained due to the gravitational forces as follows

$$F_{m_m g} = \left(1 + \frac{d_p \tilde{u}_i^2}{l_i}\right) m_m g \quad (4.15)$$

$$F_{m_r g} = \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i}\right) m_r g \quad (4.16)$$

Constraint forces (F_c) that occurs on upper junction point of the legs are shown in Figure 4.1. Total external force can be written as,

$$\tau = F_d + F_{m_m g} + F_{m_r g} + F_c \quad (4.17)$$

Using equations (4.6) to (4.17), constraint force is obtained as follows,

$$\begin{aligned}
\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_i^B} \right) - \frac{\partial T}{\partial q_i^B} &= F_d + F_{m_m g} + F_{m_r g} + F_c \\
F_c &= (M_1 + M_2 + M_3) \ddot{q}_i^B + B \dot{q}_i^B - (F_d + F_{m_m g} + F_{m_r g}) \quad (4.18)
\end{aligned}$$

$$B \dot{q}_i^B = \frac{d(M_1 + M_2 + M_3)}{dt} \dot{q}_i^B - \frac{\partial T}{\partial q_i^B}$$

$$\begin{aligned}
B = & \frac{d_p m_m}{l_i^2} (u_i (\dot{q}_i^B)^T \tilde{u}_i^T \tilde{u}_i + u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i + \tilde{u}_i^T \tilde{u}_i \dot{q}_i^B u_i^T) \\
& - \frac{(d_p^2 m_m + d_b^2 m_r)}{l_i^3} (u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i + \tilde{u}_i^T \tilde{u}_i \dot{q}_i^B u_i^T) \\
& - \frac{2(I_r + I_m)}{l_i^3} (\tilde{u}_i^T \tilde{u}_i \dot{q}_i^B u_i^T)
\end{aligned} \tag{4.19}$$

Finally, the equation (4.18) can be rewritten as follows,

$$\begin{aligned}
F_c = & (M_1 + M_2 + M_3) \left[\mathbf{I} \quad R_p^B (\tilde{q}_i^p)^T (R_p^B)^T \right] \ddot{P} + B \left[\mathbf{I} \quad R_p^B (\tilde{q}_i^p)^T (R_p^B)^T \right] \dot{P} \\
& + (M_1 + M_2 + M_3) \tilde{w}^2 R_p^B q_i^p - (F_d + F_{m_{mg}} + F_{m_{rg}})
\end{aligned} \tag{4.20}$$

4.2 Dynamic Analysis of the Moving Platform

Dynamic analysis of the moving platform is used by Newton Euler method. Leg dynamics of the mechanism are then combined with moving platform dynamics. Having done this, the inertia, Coriolis and the gravity matrices of the mechanism are determined. Equation of motion of the mechanism is obtained, and the actuator forces are calculated according to the desired trajectory of the moving platform. The graphical representations of the actuator forces are presented in Figure 4.3 and 4.4. To solve the dynamic equations of the system, MATLAB code is developed which is included in Appendix C.

The origin of the platform can change depending on the load changes. Therefore, the position of the center point should be specified (Guo and Li, 2006). The position, velocity and acceleration of the center point with respect to base frame $q_c^B, \dot{q}_c^B, \ddot{q}_c^B$ can be expressed in following equations.

$$q_c^B = t + R_p^B * q_c^p \tag{4.21}$$

$$\dot{q}_c^B = \left[\mathbf{I} \quad R_p^B (\tilde{q}_c^p)^T (R_p^B)^T \right] \dot{P} \tag{4.22}$$

$$\ddot{q}_c^B = \left[\mathbf{I} \quad R_p^B (\tilde{q}_c^p)^T (R_p^B)^T \right] \ddot{P} + \tilde{w}^2 R_p^B q_c^p \tag{4.23}$$

Forces acting on the platform and moments on the platform are shown in Figure 4.2. Equilibrium equation of the forces acting on the platform is determined in equation 4.24.

$$\sum_{i=1}^6 (F_c)_i = m_p(g - \ddot{q}_c^B)$$

$$\sum_{i=1}^6 (F_c)_i = m_p g - [m_p I \quad m_p R_P^B \tilde{q}_c^p (R_P^B)^T] \ddot{P} - m_p \tilde{\omega}^2 R_P^B q_c^p \quad (4.24)$$

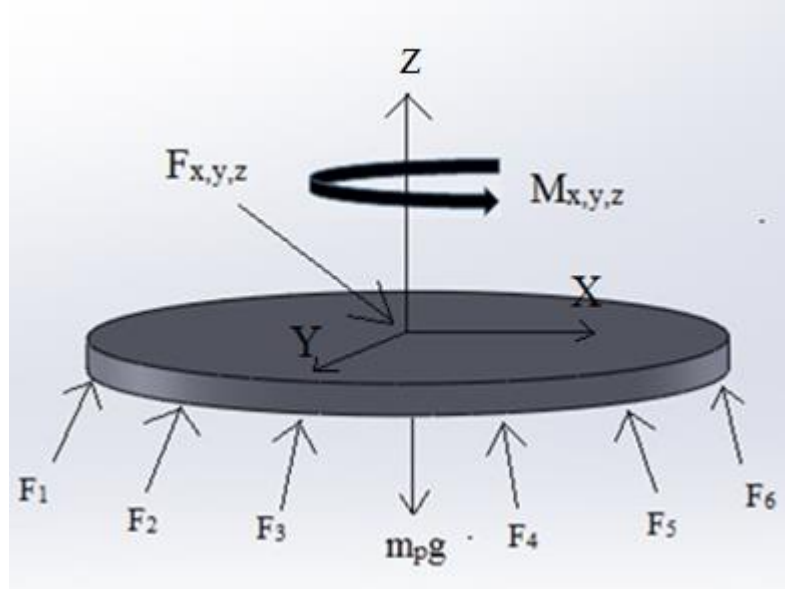


Figure 4.2 Forces and Moments Acting on Moving Platform

The moments come from the legs acting on the moving platform according to center point of the platform as shown in Figure 4.2. Moments equations are given as follows

$$M = I_p \dot{\omega} \quad \text{and} \quad M = F q_c^p$$

$$\sum_{i=1}^6 R_P^B q_i^p \times (F_c)_i$$

$$= m_p R_P^B q_c^p \times g - (m_p R_P^B q_c^p \times \ddot{q}_c^B + R_P^B I_p (R_P^B)^T \dot{\omega} + \tilde{\omega} R_P^B I_p (R_P^B)^T \omega)$$

$$\sum_{i=1}^6 R_P^B \tilde{q}_i^p (R_P^B)^T (F_c)_i$$

$$= m_p R_P^B \tilde{q}_c^p (R_P^B)^T g$$

$$- ([m_p R_P^B \tilde{q}_c^p (R_P^B)^T \quad m_p R_P^B \tilde{q}_c^p (\tilde{q}_c^p)^T (R_P^B)^T + R_P^B I_p (R_P^B)^T] \ddot{P}$$

$$+ \tilde{\omega} R_P^B I_p (R_P^B)^T \omega + m_p R_P^B \tilde{q}_c^p (R_P^B)^T \tilde{\omega}^2 R_P^B q_c^p) \quad (4.25)$$

$$F_c D_c = \begin{bmatrix} m_p g \\ m_p R_P^B \tilde{q}_c^p (R_P^B)^T g \end{bmatrix} - M_p \ddot{P} - B_p \dot{P} - \begin{bmatrix} m_p I \\ m_p R_P^B (\tilde{q}_c^p)^T (R_P^B)^T \end{bmatrix} \tilde{\omega}^2 R_P^B q_c^p \quad (4.26)$$

$$M_p = \begin{bmatrix} m_p I & m_p R_p^B \tilde{q}_c^p (R_p^B)^T \\ m_p R_p^B \tilde{q}_c^p (R_p^B)^T & m_p R_p^B \tilde{q}_c^p (\tilde{q}_c^p)^T (R_p^B)^T + R_p^B I_p (R_p^B)^T \end{bmatrix}$$

$$B_p = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{w} R_p^B I_p (R_p^B)^T \end{bmatrix}$$

$$D_c = \begin{bmatrix} I & & & & & \\ R_p^B \tilde{q}_1^p (R_p^B)^T & I & & & & \\ R_p^B \tilde{q}_2^p (R_p^B)^T & & I & & & \\ R_p^B \tilde{q}_3^p (R_p^B)^T & & & \cdots & & \\ \vdots & & & & \ddots & \\ R_p^B \tilde{q}_6^p (R_p^B)^T & & & & & I \end{bmatrix}$$

$$F_c = [F_{ci} \dots \dots]^T \quad (i = 1, 2 \dots 6)$$

The general equation (4.27) (Inertia, Coriolis-Centrifugal and Gravity matrix) of the Stewart mechanism can be described by equations (4.20) and (4.26).

$$M(P)\ddot{P} + B(P, \dot{P})\dot{P} + K(P) = (J^{-1})^T F \quad (4.27)$$

$$M(P) = M_p + \sum_{i=1}^6 \begin{bmatrix} I \\ R_p^B \tilde{q}_i^p (R_p^B)^T \end{bmatrix} (M_1 + M_2 + M_3)_i \begin{bmatrix} I & R_p^B (\tilde{q}_i^p)^T (R_p^B)^T \end{bmatrix}_i$$

$$\begin{aligned} B(P, \dot{P})\dot{P} &= B_p \dot{P} + \sum_{i=1}^6 \begin{bmatrix} I \\ R_p^B \tilde{q}_i^p (R_p^B)^T \end{bmatrix} (B) \begin{bmatrix} I & R_p^B (\tilde{q}_i^p)^T (R_p^B)^T \end{bmatrix}_i \dot{P} \\ &+ \begin{bmatrix} m_p I \\ m_p R_p^B (\tilde{q}_c^p)^T (R_p^B)^T \end{bmatrix} \tilde{w}^2 R_p^B q_c^p \\ &+ \sum_{i=1}^6 \begin{bmatrix} I \\ R_p^B \tilde{q}_i^p (R_p^B)^T \end{bmatrix} (M_1 + M_2 + M_3)_i \tilde{w}^2 R_p^B q_i^p \end{aligned}$$

$$K(P) = - \begin{bmatrix} m_p g \\ m_p R_p^B \tilde{q}_c^p (R_p^B)^T g \end{bmatrix} - \sum_{i=1}^6 \begin{bmatrix} I \\ R_p^B \tilde{q}_i^p (R_p^B)^T \end{bmatrix} (F_{m_m g} + F_{m_r g})_i$$

$$(J^{-1})^T = \begin{bmatrix} u_i & \cdots \\ R_p^B \tilde{q}_i^p (R_p^B)^T u_i & \cdots \end{bmatrix} \quad (i = 1, 2 \dots 6)$$

$$F = [F_i \dots]^T \quad (i = 1, 2 \dots 6)$$

In order to achieve dynamic simulation of the system, the mechanism parameters are specified as shown in Table 4.1. The rotary part of the leg has high mass value, because linear motor and its stator are mounted on the rotary part.

Table 4.1 Mechanism Parameters

m_p	8 kg
m_r	5 kg
m_m	0.5 kg
d_b	0.5 m
d_p	0.5 m

Mass moment of inertia of the moving platform and legs are presented as follows.

$$I_p = \begin{bmatrix} 0.425 & 0 & 0 \\ 0 & 0.425 & 0 \\ 0 & 0 & 0.85 \end{bmatrix} kg\ m^2$$

$$I_r = \begin{bmatrix} 0.25 & 0 & 0 \\ 0 & 0.25 & 0 \\ 0 & 0 & 0 \end{bmatrix} kg\ m^2$$

$$I_m = \begin{bmatrix} 0.03125 & 0 & 0 \\ 0 & 0.03125 & 0 \\ 0 & 0 & 0 \end{bmatrix} kg\ m^2$$

In this study, the legs junction coordinates of the moving and base platform and desired trajectories are taken from the studies of Wang and Gosselin (1998) and Tsai (2000). Two trajectories are specified. The first trajectory is prescribed as constant rotation and sinusoidal translation motion of the platform. The second trajectory is prescribed as a step input for translation motion, and rotation about Z axis of the moving platform. Then, the actuator forces are calculated using MATLAB codes, and graphical representation is presented in Figures 4.3, 4.4, 4.5 and 4.6 as follows.

Coordinates of the base platform are given as;

$$b_1 = \begin{bmatrix} -2.12 \\ 1.374 \\ 0 \end{bmatrix} m \quad b_2 = \begin{bmatrix} -2.38 \\ 1.224 \\ 0 \end{bmatrix} m \quad b_3 = \begin{bmatrix} -2.38 \\ -1.224 \\ 0 \end{bmatrix} m$$

$$b_4 = \begin{bmatrix} -2.12 \\ -1.374 \\ 0 \end{bmatrix} m \quad b_5 = \begin{bmatrix} 0 \\ -0.15 \\ 0 \end{bmatrix} m \quad b_6 = \begin{bmatrix} 0 \\ 0.15 \\ 0 \end{bmatrix} m$$

Coordinates of the moving platform are given as;

$$q_1^p = \begin{bmatrix} 0.17 \\ 0.595 \\ -0.4 \end{bmatrix} m \quad q_2^p = \begin{bmatrix} -0.6 \\ 0.15 \\ -0.4 \end{bmatrix} m \quad q_3^p = \begin{bmatrix} -0.6 \\ -0.15 \\ -0.4 \end{bmatrix} m$$

$$q_4^p = \begin{bmatrix} 0.17 \\ -0.595 \\ -0.4 \end{bmatrix} m \quad q_5^p = \begin{bmatrix} 0.43 \\ -0.445 \\ -0.4 \end{bmatrix} m \quad q_6^p = \begin{bmatrix} 0.43 \\ 0.445 \\ -0.4 \end{bmatrix} m$$

$$g = \begin{bmatrix} 0 \\ 0 \\ -9.81 \end{bmatrix} m/s^2$$

First trajectory; $[\alpha \ \beta \ \gamma]^T = [0 \ 0 \ 0]^T$

$$t_x = -1.5 + 0.2 \sin(\omega t) \text{ m}, \quad t_y = 0.2 \sin(\omega t) \text{ m}, \quad t_z = 1 + 0.2 \sin(\omega t) \text{ m}$$

Where $\omega = 3 \text{ rad/s}$ and $0 \leq \omega t \leq 2\pi$

$$m_p = 1.5 \text{ kg}, \quad m_r = 0.1 \text{ kg}, \quad m_m = 0.1 \text{ kg}$$

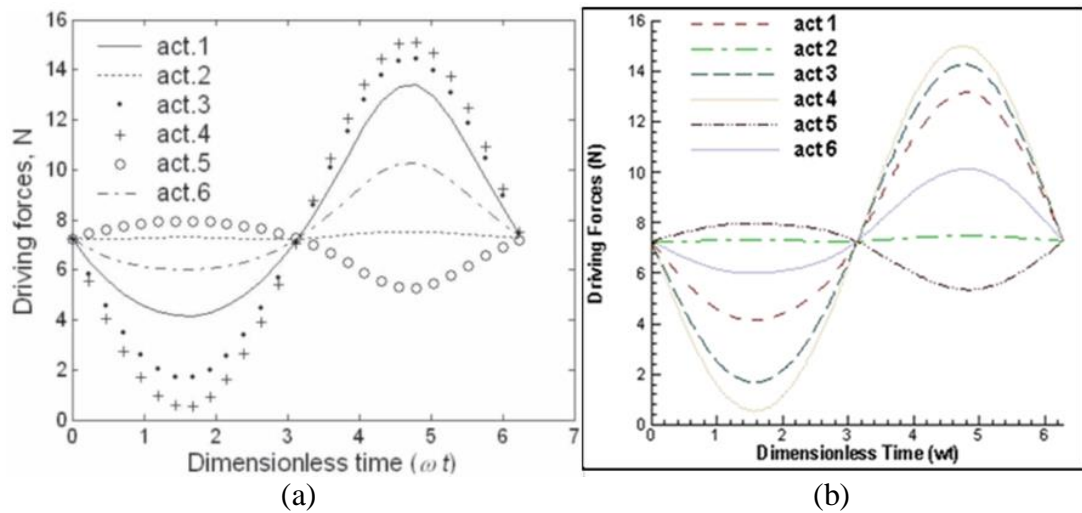


Figure 4.3 (a) Driving Forces with respect to Desired Trajectories (Guo and Li, 2006)
(b) Driving Forces with respect to Desired Trajectories

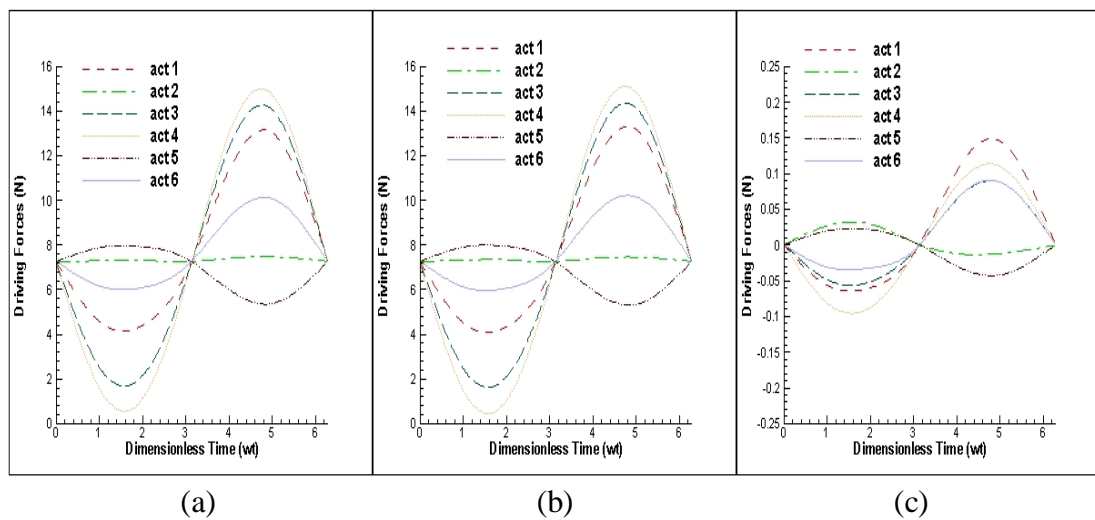


Figure 4.4 (a) Driving Forces Excluded Lower Part Inertia of the Leg (b) Included Lower Part Inertia of the Leg (c) Differences Between Included and Excluded Lower Part Inertia of the Leg for the First Trajectory

In order to observe the effect of the lower part of the leg inertia, the mass of the mechanism is increased as follows.

$$m_p=8 \text{ kg}, m_r= 5 \text{ kg}, m_m= 0.5 \text{ kg}$$

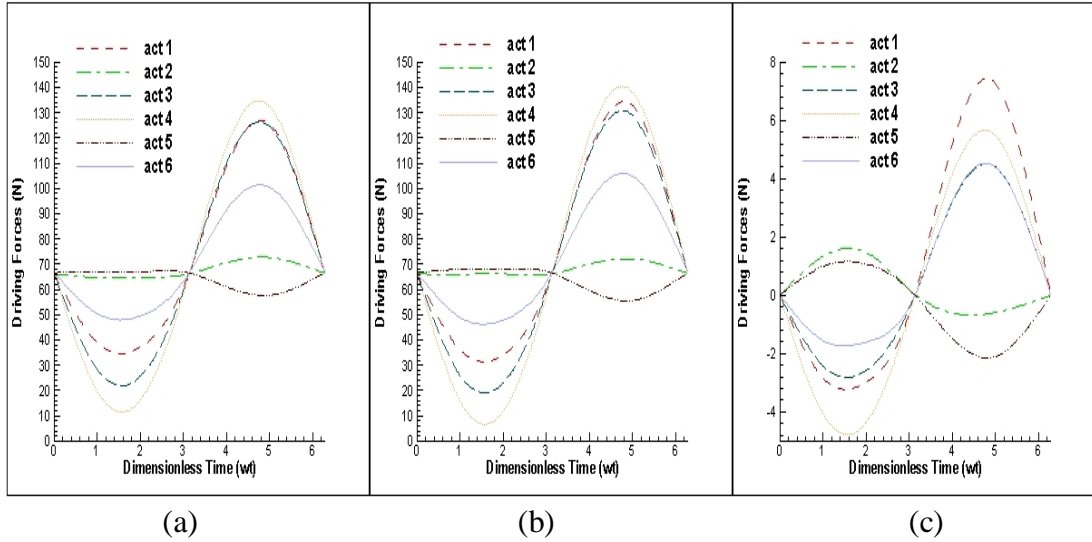


Figure 4.5 (a) Driving Forces Excluded Lower Part Inertia of the Leg (b) Included Lower Part Inertia of the Leg (c) Differences Between Included and Excluded Lower Part Inertia of the Leg for the First Trajectory

Second trajectory; $[\alpha \ \beta \ \gamma]^T = [0 \ 0 \ 0.35 \sin(\omega t)]^T$

$$t_x = -1.5 \text{ m}, t_y = 0 \text{ m}, t_z = 1 \text{ m}$$

$$m_p=8 \text{ kg}, m_r= 5 \text{ kg}, m_m= 0.5 \text{ kg}$$

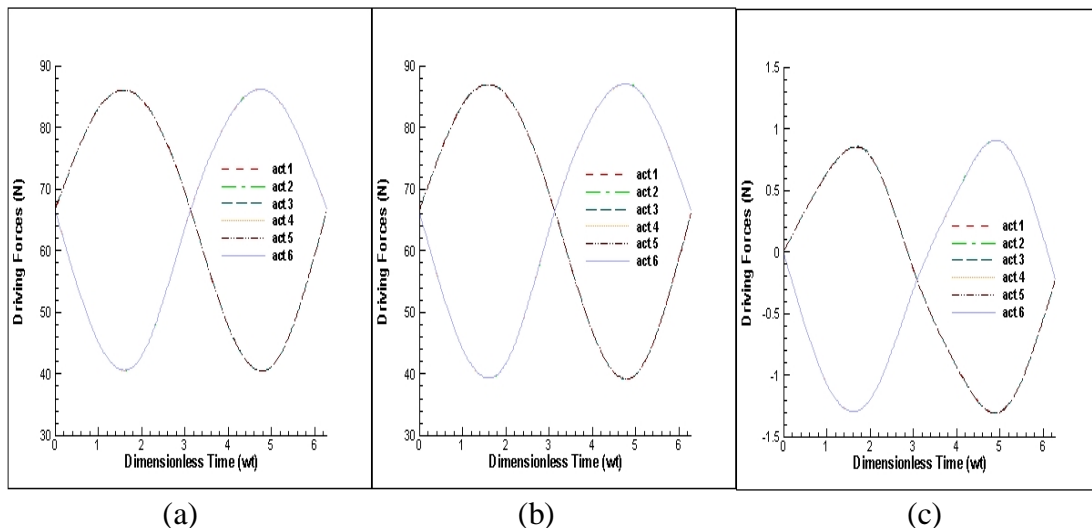


Figure 4.6 (a) Driving Forces Excluded Lower Part Inertia of the Leg (b) Included Lower Part Inertia of the Leg (c) Differences Between Included and Excluded Lower Part Inertia of the Leg for the Second Trajectory

4.3 Linear Motor Dynamics

The linear motor is preferred for some applications in the industry as it provides high acceleration, velocity and positioning accuracy. Further, mechanical transmission system can be eliminated and there is no need to compute inertial effect of the transmission system since it can be driven directly (Yao and Xu, 2002).

The dynamic equations of the linear motor are presented as follows.

$$F_m = K_f i(t) \quad (4.28)$$

$$F_m = m\ddot{x} + B_l \dot{x} + f_f(\dot{x}) + f_{ripple}(x) + f_{dis} \quad (4.29)$$

$$V(t) = K_e \dot{x} + R i(t) + L \frac{di(t)}{dt} \quad (4.30)$$

$f_f(\dot{x})$ and $f_{ripple}(x)$ are the friction force and ripple force of the linear motor, respectively. $f_{dis}(t)$ denotes disturbance in the linear motor motion. The equation of the friction and ripple forces can be written as follows (Lee et al., 2000).

$$f_f(\dot{x}) = \left(f_c + (f_s - f_c) e^{-|\dot{x}/\dot{x}_s|^\delta} + f_v \dot{x} \right) \text{sgn}(\dot{x}) \quad (4.30)$$

$$f_{ripple} = A \sin(wx + \varphi) \quad (4.31)$$

The values are given in Table 4.2 which is taken from reference (HIWIN, 2012).

Table 4.2 Linear Motor Parameters

<i>Force Constant</i>	K_f	44 N/A
<i>Electromotive Force Constant</i>	K_e	26 V/m/s
<i>Resistance</i>	R	3,1 Ω
<i>Damping Ratio</i>	B_l	0,01 Ns/m
<i>Mass of Forcer</i>	M_f	1.8 kg
<i>Unit Mass of Stator</i>	M_s	4,2 kg/m
<i>Continuous Force</i>	F_{net}	203 N

F_{net} is the force which is generated by i th actuator so, net force and driving force equal to each other.

$$F_{net} = F_d = F$$

When obtaining linear motor transfer function, ripple and friction forces of the motor are ignored. Then, the transfer function block is implemented using MATLAB/Simulink and combined with mechanism control block which are presented in chapter 5.

CHAPTER 5

PID CONTROL OF THE STEWART PLATFORM

5.1 Introduction

Proportional Integrated Derivative (PID) controller is commonly used in industry because it can easily be applied to perform system control. It is used to compensate the position error between desired input and actual output. In PID controller, three constants; proportional constant K_p , integral constant K_i and derivative constant K_d are used. Proportional constant K_p is used to decrease the response time and steady state error however it increases the oscillation of the system. The integral action is carried out by taking integral of the system error and it is also multiplied with integral constant K_i . Integral action eliminates the steady state error and decreases the response time of the system. Derivative action is performed by taking derivative of the error and the error is multiplied with derivative constant K_d . It creates damping effect on the system because it increases the stability and decreases the oscillation of the system. PID control block is shown in Figure 5.1. Equation (5.1) represents the relation between E and U which are the error and the control signal, respectively.

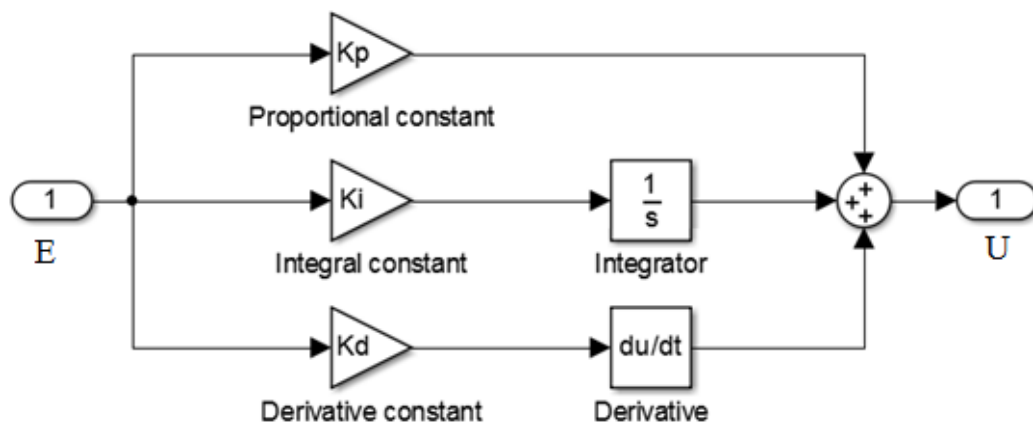


Figure 5.1 PID Controller Block Diagram

The equation (5.1) is taken from reference Johnson and Moradi (2005).

$$U(s) = K_p E(s) + \frac{K_i}{s} E(s) + K_p T_d s E(s) = \left(K_p + \frac{K_i}{s} + K_d s \right) E(s) \quad (5.1)$$

Where $K_i = \frac{K_p}{T_i}$ $K_d = K_p T_d$

5.2 PID Control of the System

The control of the system is implemented using PID control in MATLAB/Simulink. Dynamic equation of the system is obtained using equation (5.2) in Simulink. The dynamic model of the system is shown in Figure 5.2. Actual position and orientation of the mechanism (P) are obtained by taking integral (\dot{P}) acceleration of the moving platform.

$$\ddot{P} = M(P)^{-1} [(J^{-1})^T F - B(P, \dot{P}) \dot{P} - K(P)] \quad (5.2)$$

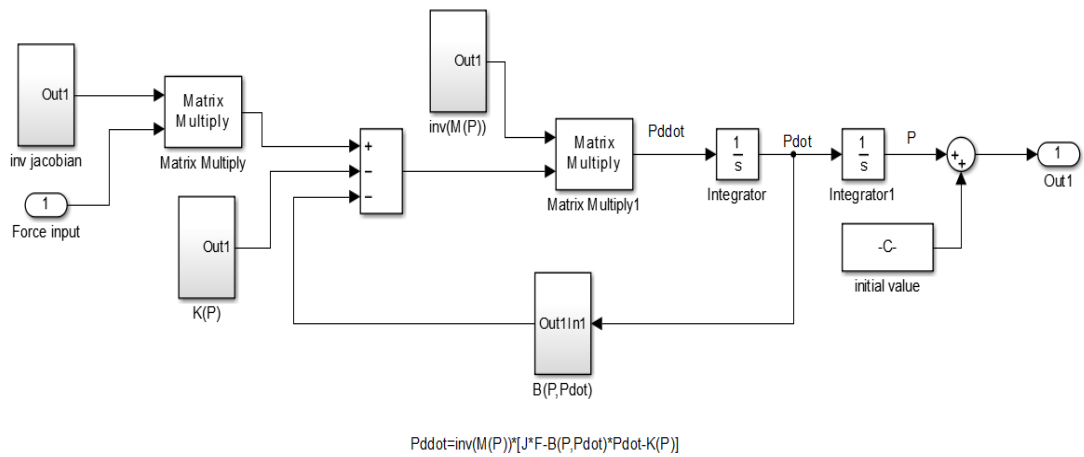


Figure 5.2 Dynamic Model of the System

Using inverse kinematic block, leg lengths are computed according to desired trajectory of the moving platform. Similar process is also applied for actual position and orientation of the moving platform which are taken from dynamic model of the system. The actual and desired leg lengths are then obtained, and the legs position errors are computed. Inverse kinematic block is seen in Figure 5.3. In order to provide the system control, the legs position error are multiplied with PID constants. Needed forces are calculated for each legs to perform desired trajectories of the moving platform and transferred to the dynamic model of the mechanism. System block

diagram includes dynamic model, inverse kinematic and PID blocks, it is given Figure 5.4.

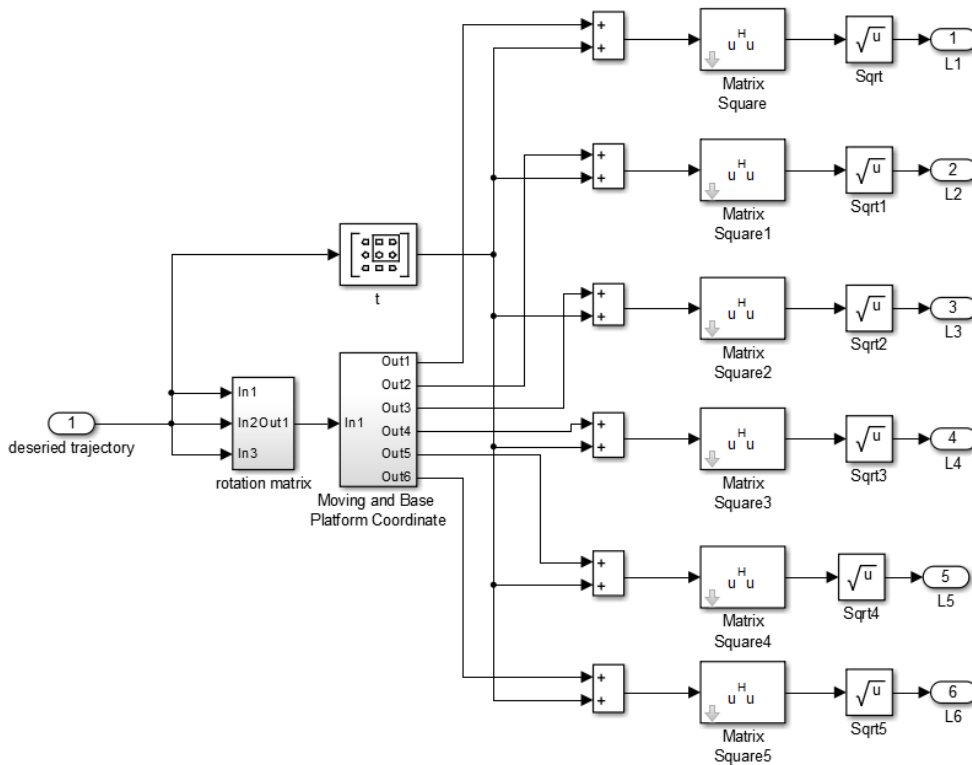


Figure 5.3 Inverse Kinematic Block Diagram

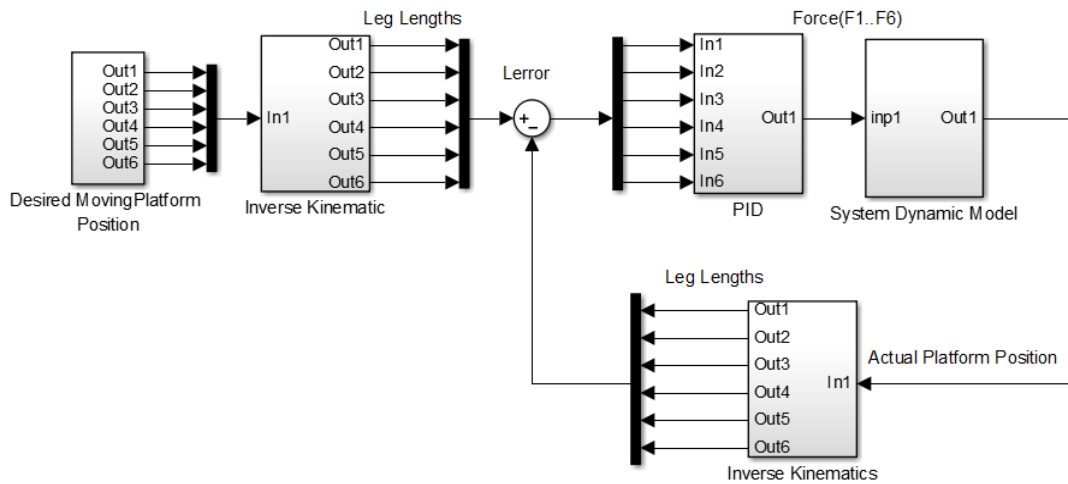


Figure 5.4 System Block Diagram

Case studies are implemented to verify the proposed system control with respect to different desired trajectories as follows.

5.2.1 Case Study I

Parameters in Case study I are taken from Wang and Gosselin's (1998) study to validate our PID controller system. These parameters are presented as follows.

$$I_p = \begin{bmatrix} 0.08 & 0 & 0 \\ 0 & 0.08 & 0 \\ 0 & 0 & 0.08 \end{bmatrix} kg\ m^2$$

$$I_r = \begin{bmatrix} 0.00625 & 0 & 0 \\ 0 & 0.00625 & 0 \\ 0 & 0 & 0 \end{bmatrix} kg\ m^2$$

$$I_m = \begin{bmatrix} 0.00625 & 0 & 0 \\ 0 & 0.00625 & 0 \\ 0 & 0 & 0 \end{bmatrix} kg\ m^2$$

$$m_p = 1.5\ kg, m_r = 0.1\ kg, m_m = 0.1\ kg$$

$$d_b = 0.5\ m, d_p = 0.5\ m$$

Initial positions of the moving platform are given as following;

$$P(0) = [-1.5\ 0\ 1\ 0\ 0\ 0]^T$$

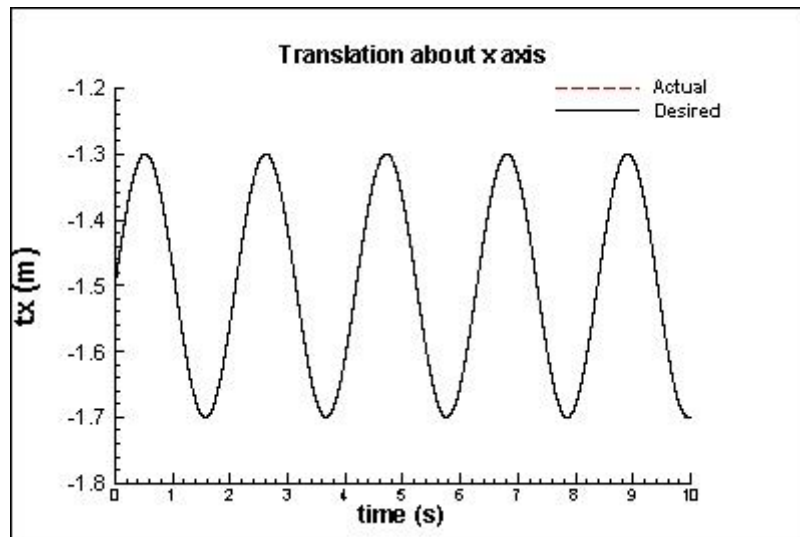
Desired trajectories of the moving platform are given as following;

$$P(t) = \begin{bmatrix} -1.5 + 0.2\sin(wt) \\ 0.2\sin(wt) \\ 1 + 0.2\sin(wt) \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

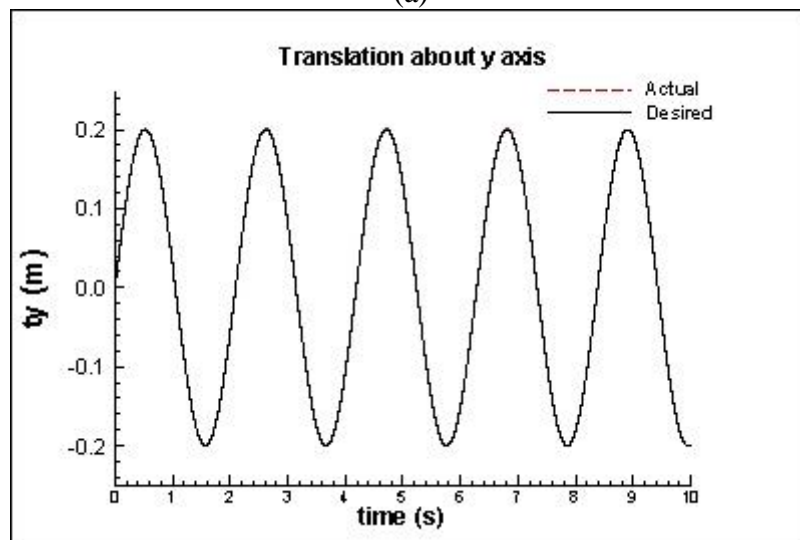
Where $w = 3\ rad/s$

5.2.1.1 Results and Discussion of Case Study I

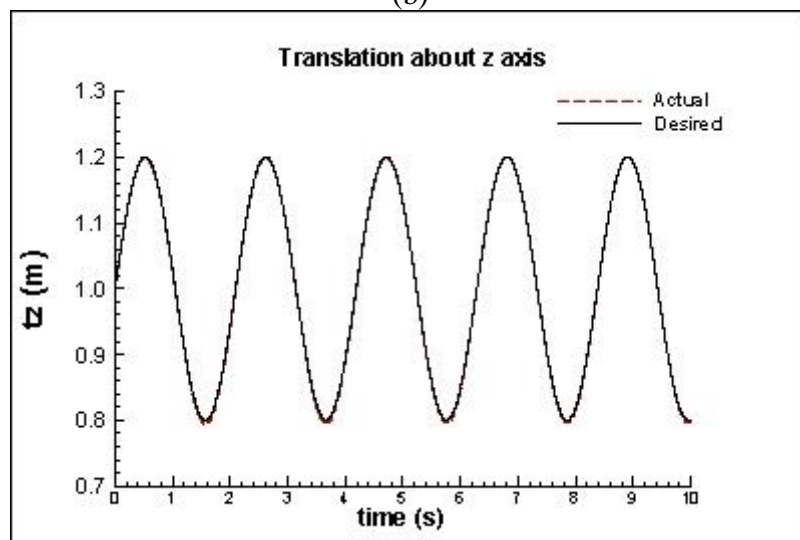
It is observed that the rotation error about x and y axes are less than 2×10^{-3} rad and in z axis is less than 10^{-4} rad. Translation error about x and y axes are less than 10^{-4} m and about z axis is less than 10^{-3} m. Position and orientation errors of the moving platform may be reduced by means of increasing PID controller constants. Oscillation occurred in rotation about x and y axes, may be decreased by increasing the derivative constant. But, it definitely increases the response time of the system. Orientation and position of the moving platform for the desired and actual trajectories are shown in Figure 5.5 (a-f).



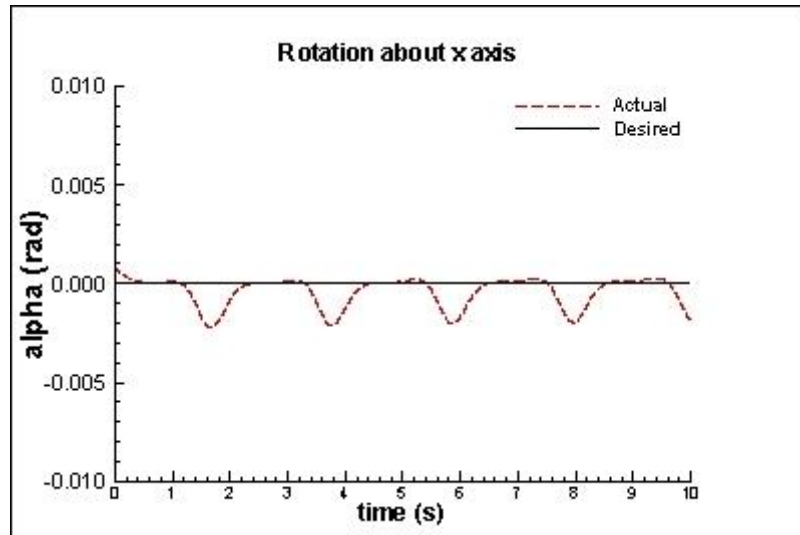
(a)



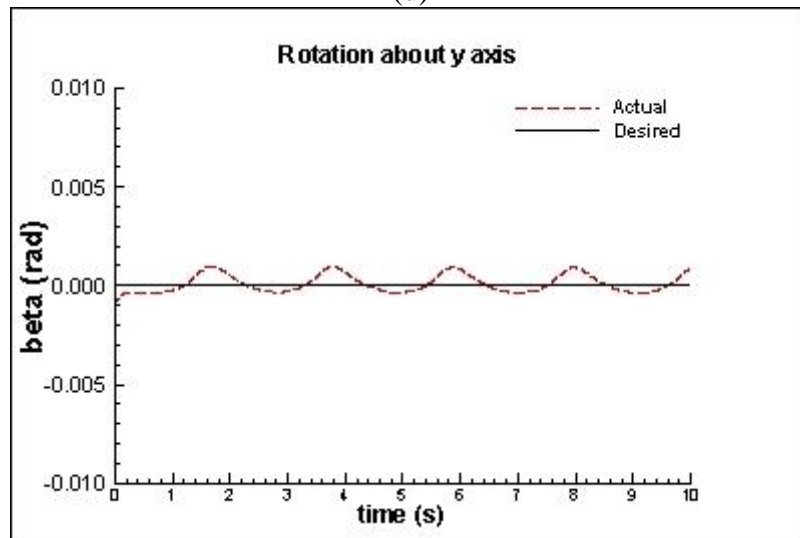
(b)



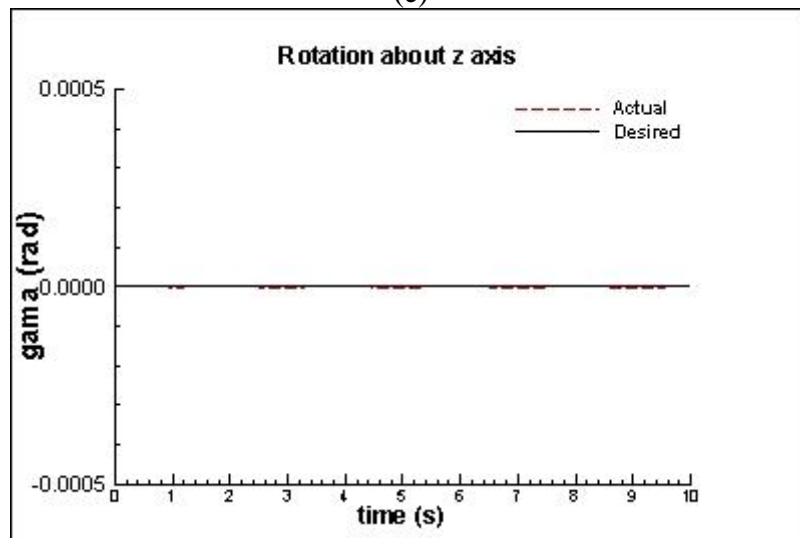
(c)



(d)



(e)



(f)

Figure 5.5 Desired and Actual Trajectories of the System for Case Study I (a-f)

5.2.2 Case Study II

In Case study II, mass moment of inertia of the legs and the moving platform and their mass are taken from Section 4.2.

Initial positions of the moving platform are given as following;

$$P(0) = [-1.5 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0]^T$$

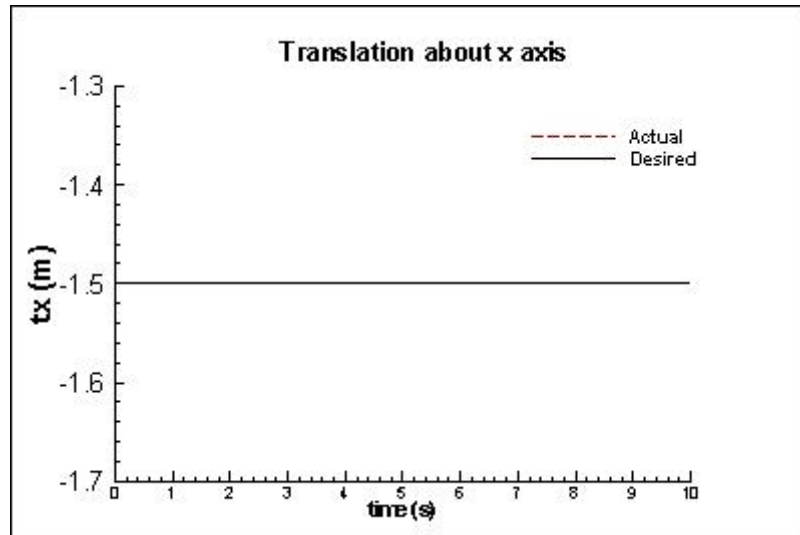
Desired trajectories of the moving platform are given as following;

$$P(t) = \begin{bmatrix} -1.5 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0.35\sin(wt) \end{bmatrix}$$

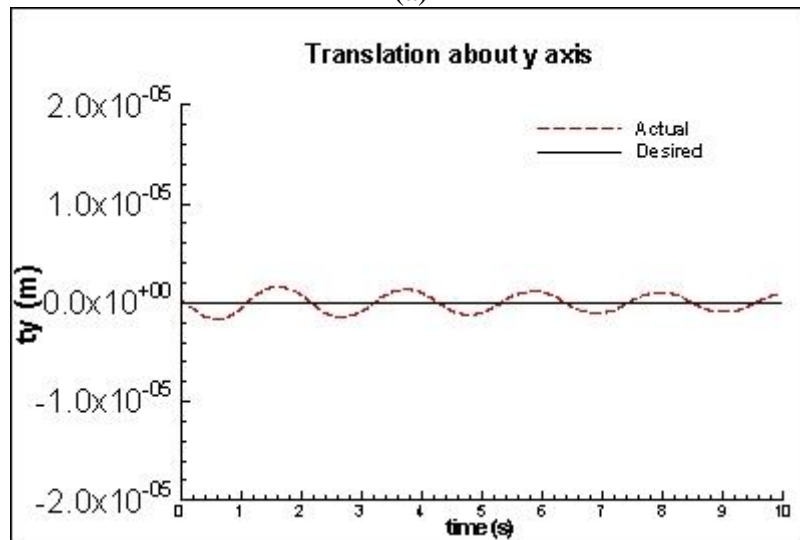
Where $w = 3\text{rad/s}$

5.2.2.1 Results and Discussion of Case Study II

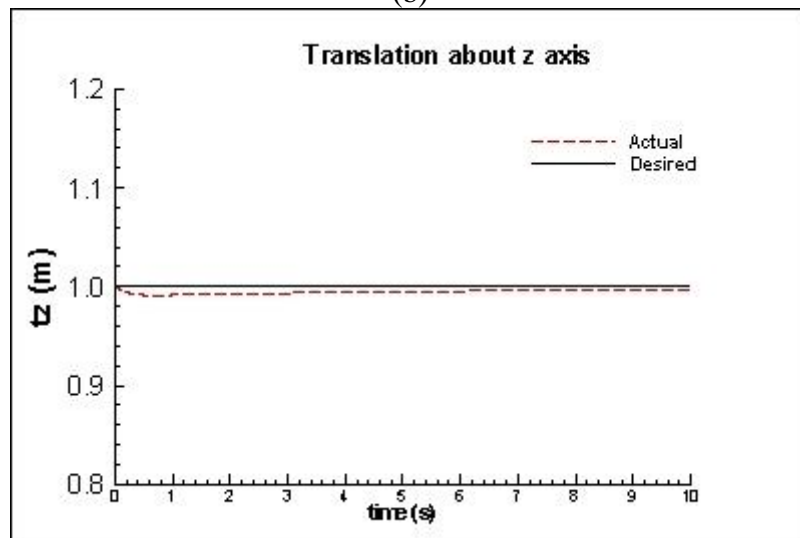
According to control results, it is observed that the rotation error about x, y and z axes are less than 10^{-6} rad. Translation error about x and y axes are less than 10^{-6} m and about z axis is less than 10^{-4} m. PID constants in the second case study have higher than the first case study due to the fact that the mass of the system is increased. The settling time in translation about z axis and in rotation about y axis can be decreased by using high proportional constant value. However, oscillation of the system is increased due to high proportional constant. Orientation and position of the moving platform for the desired and actual trajectories are shown in Figure 5.6 (a-f).



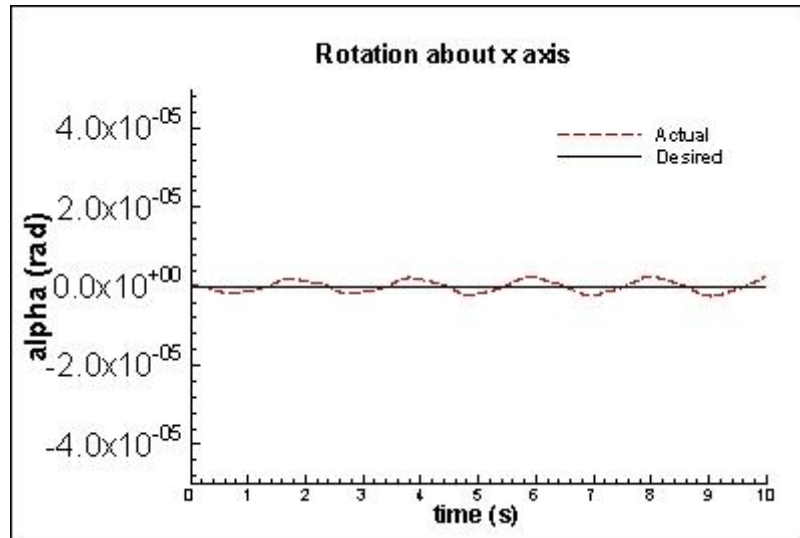
(a)



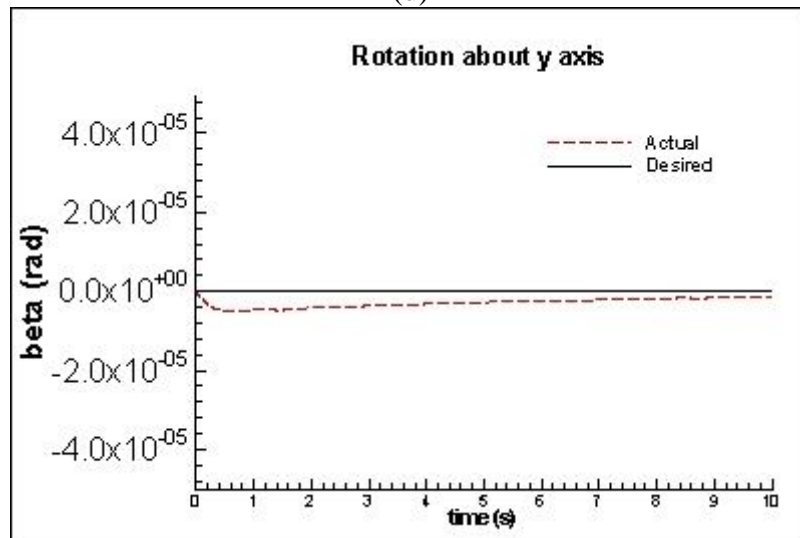
(b)



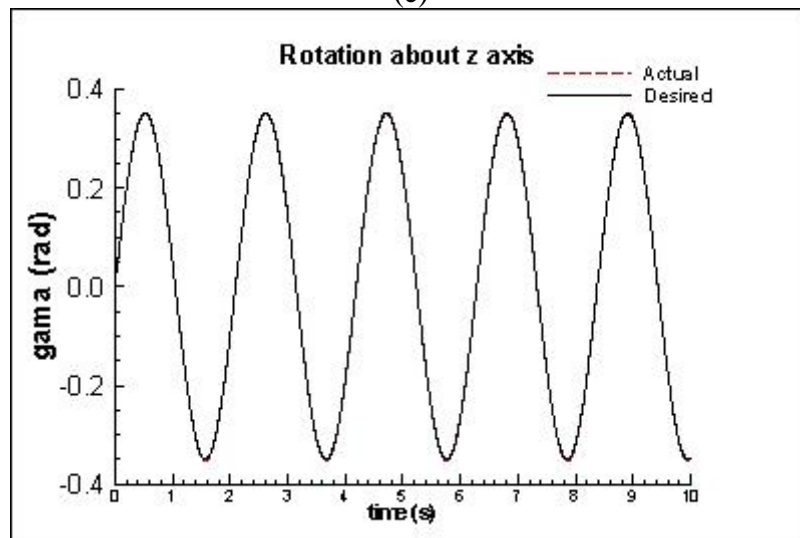
(c)



(d)



(e)



(f)

Figure 5.6 Desired and Actual Trajectories of the System for Case Study II (a-f)

5.2.3 Case Study III

In the case study III, values of mass moment of inertia of the legs and moving platform and their mass are not changed. Only initial positions of the moving platform and the desired trajectories of the moving platform are taken as followings

Initial positions of the moving platform given as following;

$$P(0) = [-1.3 \quad 0.2 \quad 1.2 \quad 0 \quad 0 \quad 0]^T$$

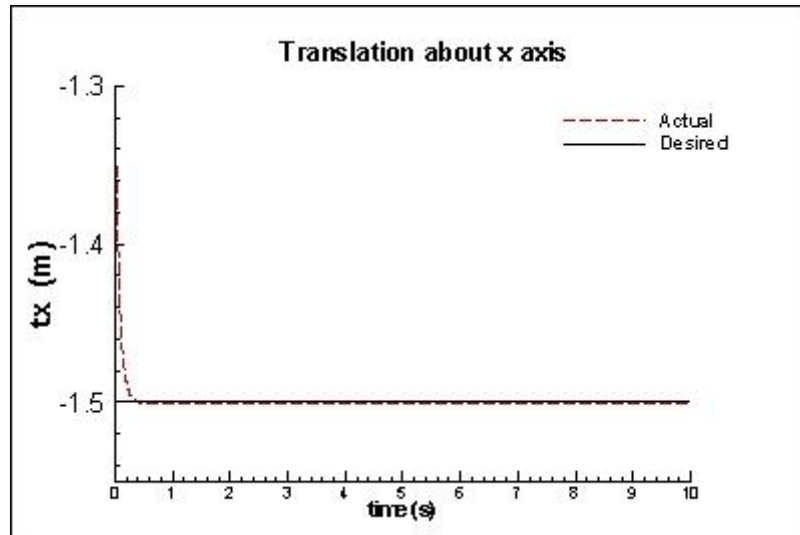
Desired trajectories of the moving platform given as following;

$$P(t) = \begin{bmatrix} -1.5 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0.35\sin(\omega t) \end{bmatrix}$$

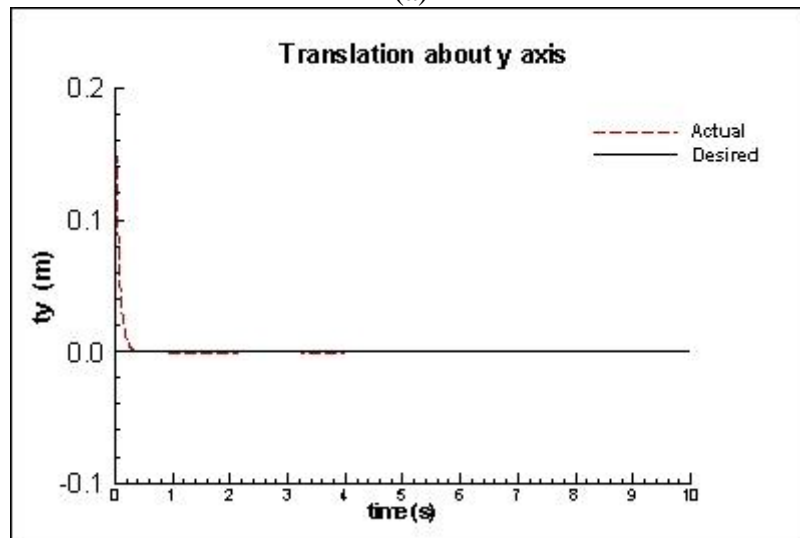
Where $\omega = 3\text{rad/s}$

5.2.3.1 Results and Discussion of Case Study III

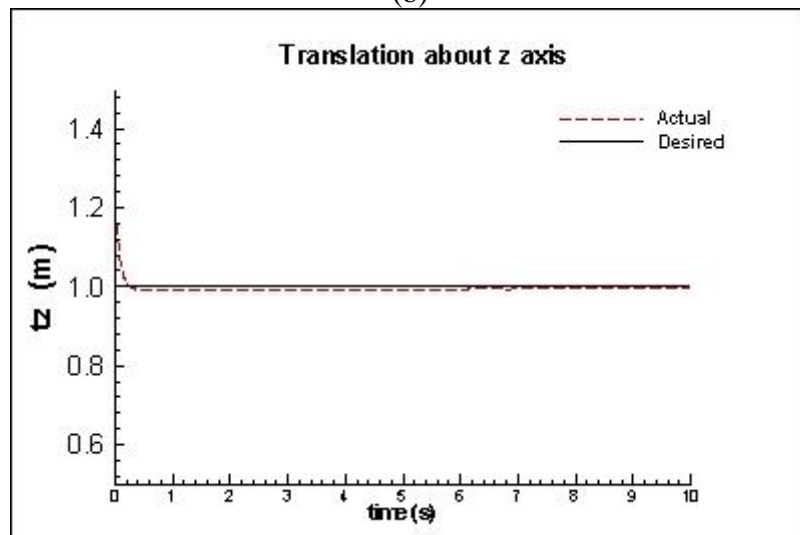
According to control results, it is observed that the rotation error about x, y and z axes are less than 10^{-5} rad. Translation error about x axis is less than 10^{-4} m and about y and z axes are less than 10^{-3} m. By increasing integral and derivative constant of the controller, errors may be decreased and oscillation in rotation about y axis can be eliminated. However, response time of the system can increase. Orientation and position of the moving platform for the desired and actual trajectories are shown in Figure 5.7 (a-f).



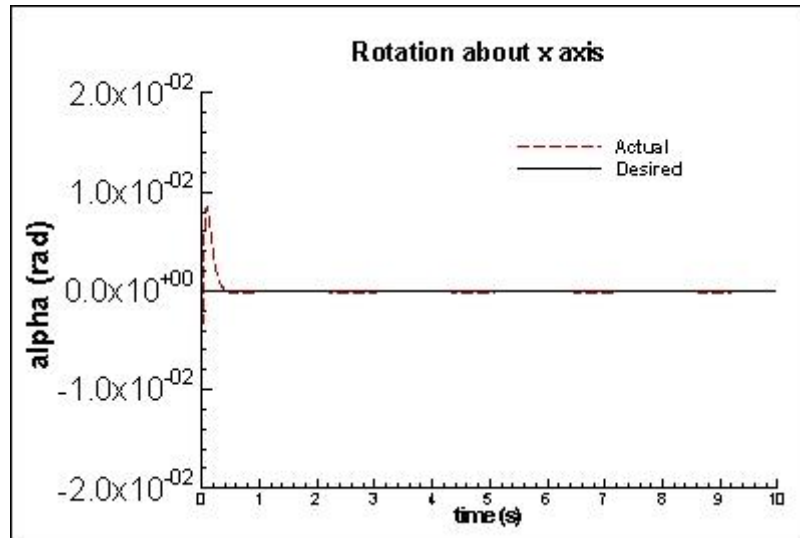
(a)



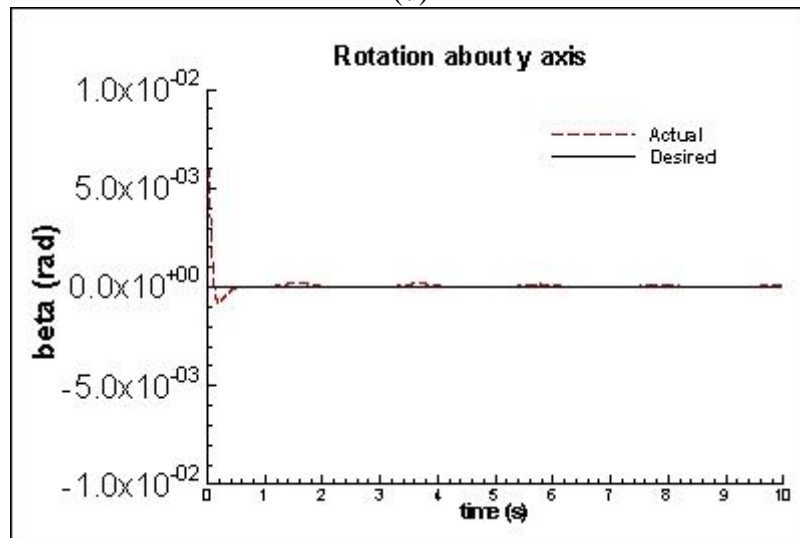
(b)



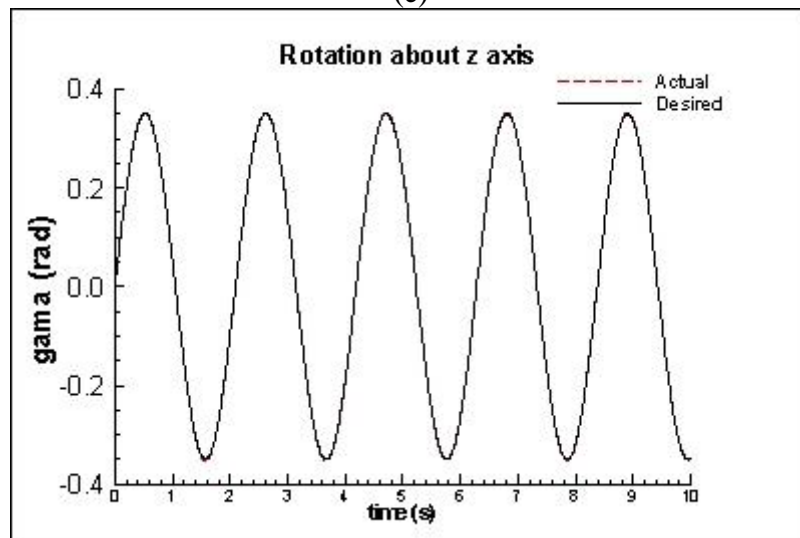
(c)



(d)



(e)



(f)

Figure 5.7 Desired and Actual Trajectories of the System for Case Study III (a-f)

5.2.4 Case Study IV

In the case study IV, values of mass moment of inertia of the legs and moving platform and their mass are not changed. Only initial positions of the moving platform and the desired trajectories of the moving platform are taken as followings

Initial positions of the moving platform are given as following;

$$P(0) = [-1.3 \quad 0.2 \quad 1.2 \quad 0 \quad 0 \quad 0]^T$$

Desired trajectories of the moving platform are given as following;

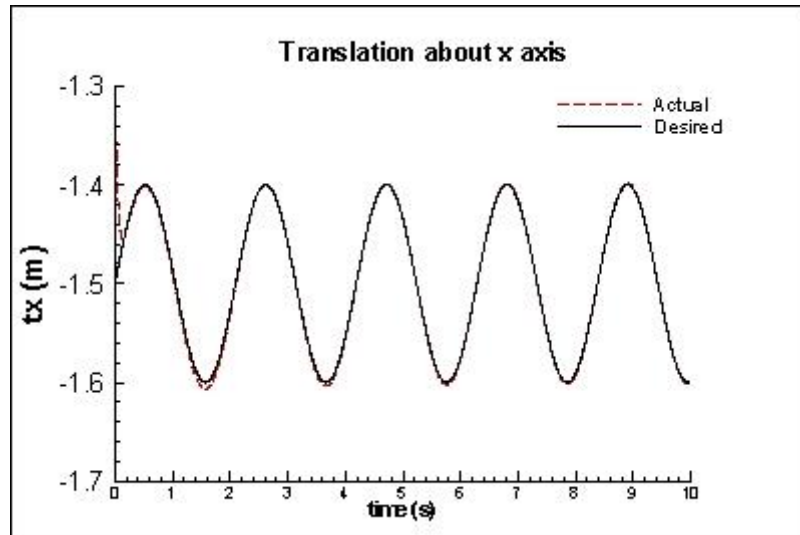
$$P(t) = \begin{bmatrix} -1.5 + 0.1\sin(\omega t) \\ 0 \\ 1 \\ 0.1\sin(\omega t) \\ 0 \\ 1 \end{bmatrix}$$

where $\omega = 3\text{rad/s}$

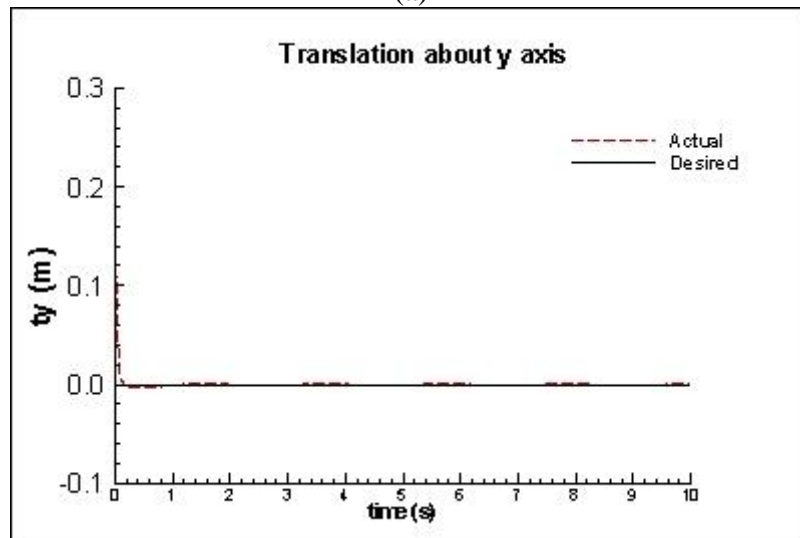
5.2.4.1 Results and Discussion of Case Study IV

According to control results, it is observed that the rotation error about x, y and z axes are less than 10^{-3} rad. Translation error about x axis is less than 10^{-4} m, about y axis is less than 10^{-5} m and z axis is less than 10^{-3} m. In order to provide the system stabilization, integral and derivative constants in rotation about x and y axes can be increased and they can reduce steady state error.

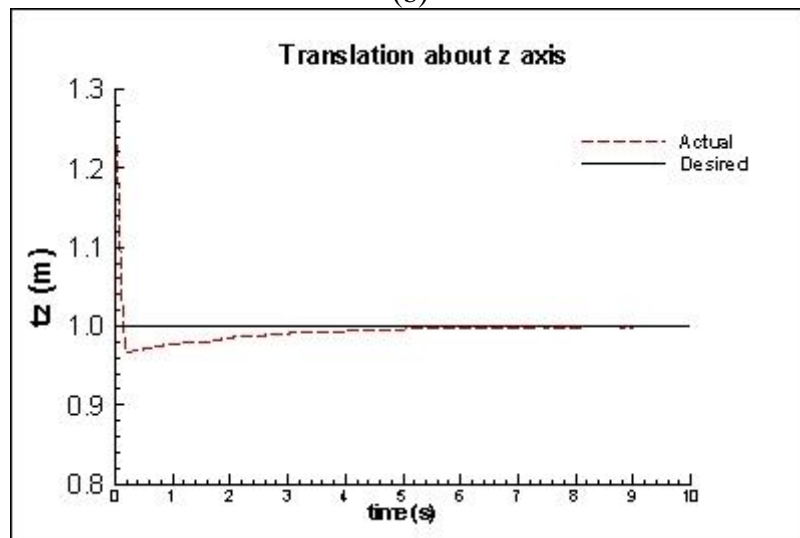
Moreover, to decrease response time in translation and rotation about z, proportional constants can be increased. But, it can cause increasing the oscillation of the system. Therefore, to achieve the control of the system for these trajectories, proper PID constants should be used. Orientation and position of the moving platform for the desired and actual trajectories are shown in Figure 5.8 (a-f).



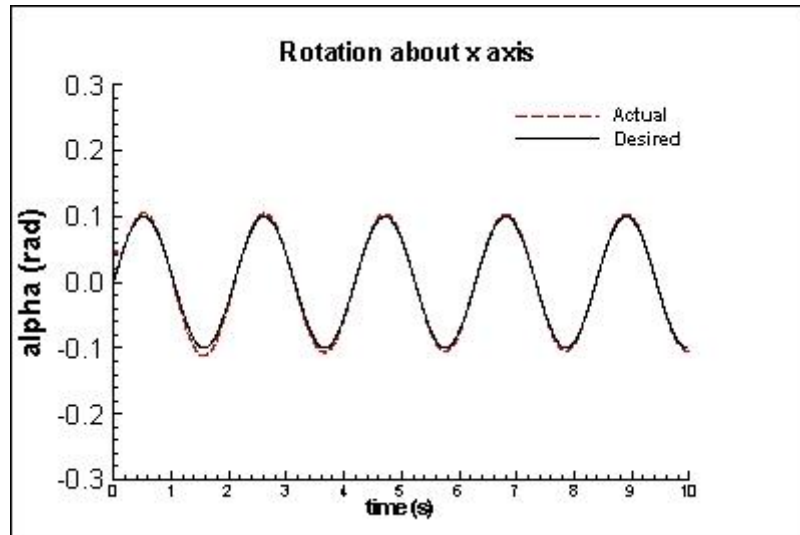
(a)



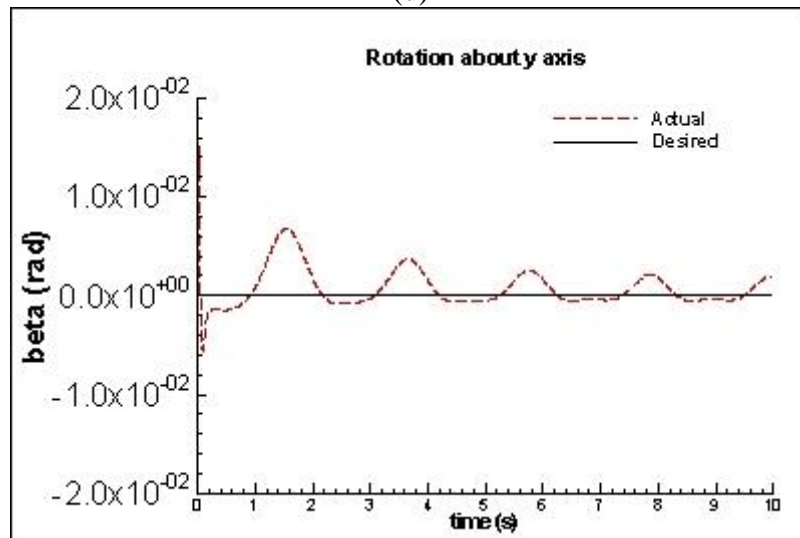
(b)



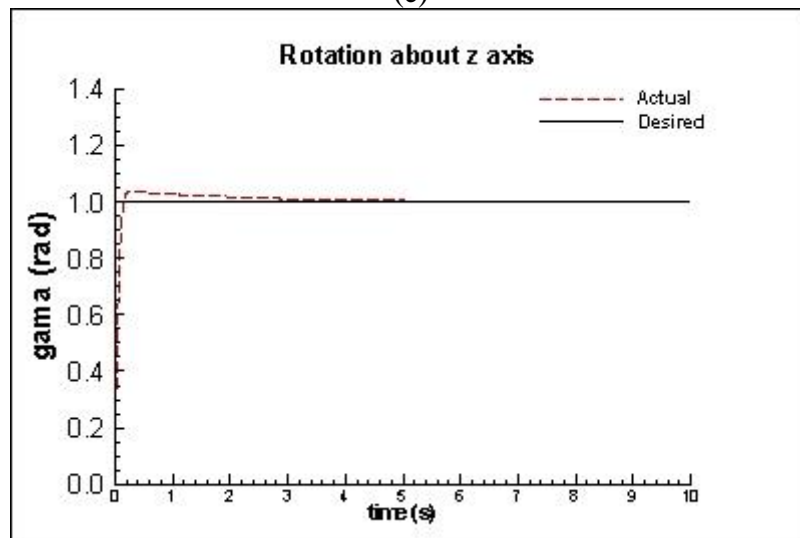
(c)



(d)



(e)



(f)

Figure 5.8 Desired and Actual Trajectories of the System for Case Study IV (a-f)

5.3 PID Control of the System with Linear Motor

In this section, the control of the system with linear motors is carried out using PID control in MATLAB/Simulink. Linear motor transfer function block is obtained using dynamic equations of the linear motor, which are presented in chapter 4 equations (4.27-4.29). PID controller is connected to transfer function block. This block provides necessary forces to the legs (l_1, l_2, \dots, l_6) to perform desired trajectory of the moving platform. System dynamic model as shown in Figure 5.9

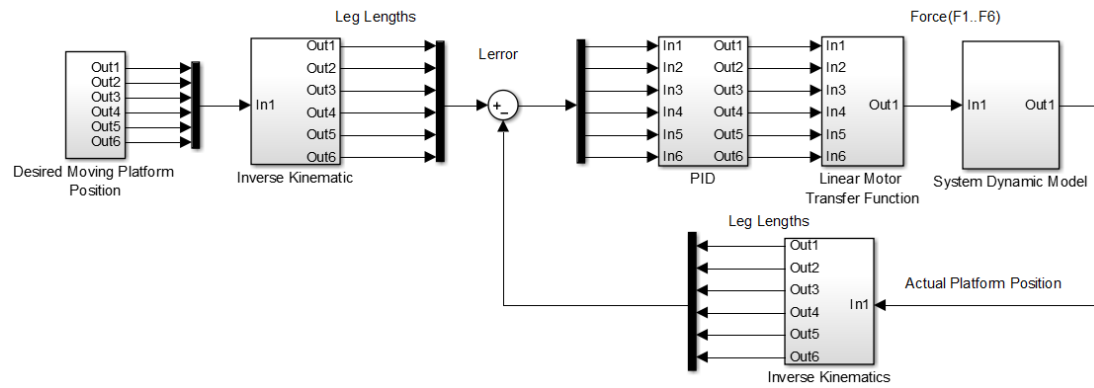


Figure 5.9 System Block Diagram with Linear Motors

Linear motor transfer function can be obtained as follows.

$$F(s) = K_f i(s) \quad (5.3)$$

$$ms^2 x(s) + Bs x(s) = F(s) \quad (5.4)$$

$$K_e s x(s) + (R + Ls) i(s) = V(s) \quad (5.5)$$

$$K_e s x(s) + (R + Ls) \frac{F(s)}{K_f} = V(s)$$

$$F(s) \left[\frac{K_e s}{ms^2 + Bs} + \frac{R + Ls}{K_f} \right] = V(s)$$

Where

$$\frac{F(s)}{V(s)} = \frac{K_f (ms^2 + Bs)}{mLs^3 + (BL + Rm)s^2 + (K_f K_e + RB)s} \quad (5.6)$$

The Simulink block diagram of the function is presented as shown in Figure 5.10.

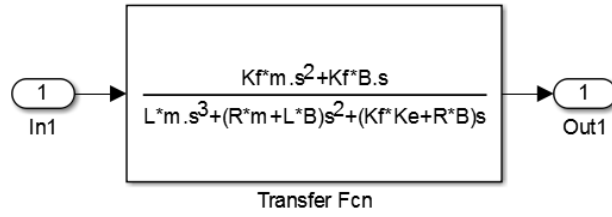


Figure 5.10 Linear Motor Transfer Function Block

Three case studies are implemented to verify the proposed system control with respect to different desired trajectories by using linear motors (with linear motor transfer function) as follows.

5.3.1 Case Study I

In the case study I, mass moment of the inertia of the legs and platform and their mass values, which are presented in chapter 4, are used to carry out control of the system with linear motors.

Initial positions of the moving platform are given as following;

$$P(0) = [-1.5 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0]^T$$

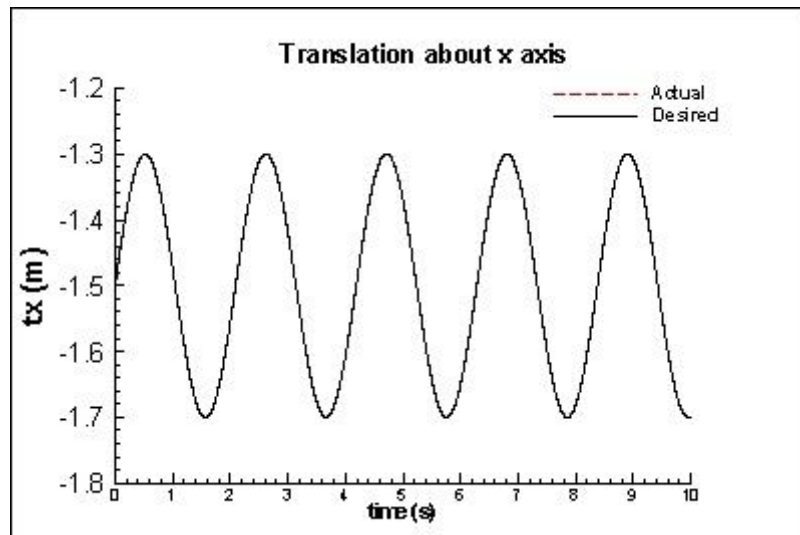
Desired trajectories of the moving platform are given as following;

$$P(t) = \begin{bmatrix} -1.5 + 0.1\sin(\omega t) \\ 0.2\sin(\omega t) \\ 1 + 0.2\sin(\omega t) \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

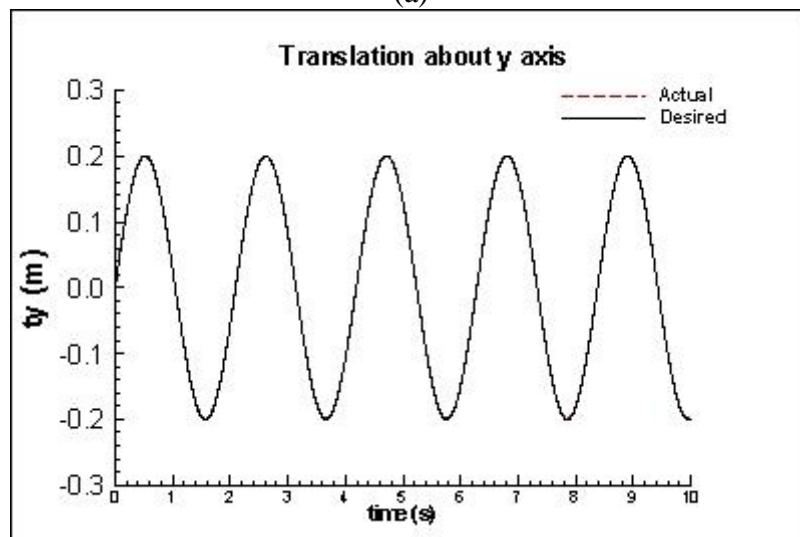
Where $\omega = 3\text{rad/s}$

5.3.1.1 Results and Discussion of Case Study I

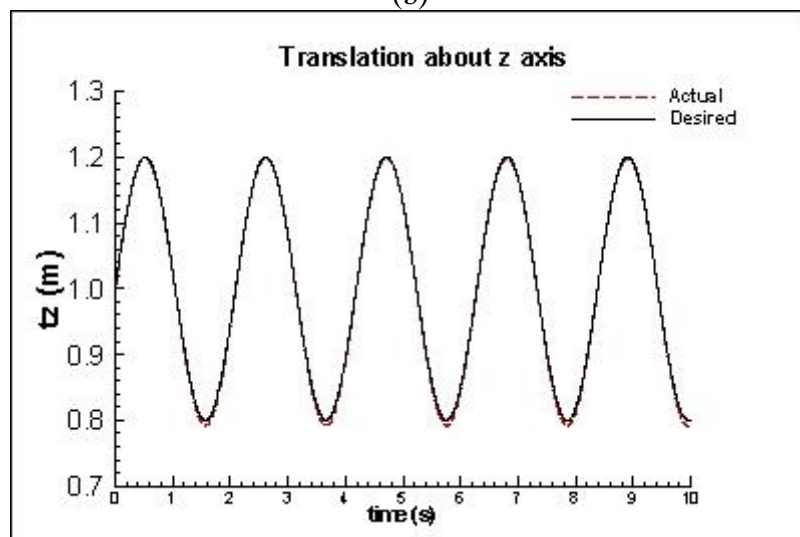
According to control results, it is observed that the rotation error about x axis is less than 10^{-3} rad and about y and z axes are less than 10^{-4} rad. Translation error about x and y axes are less than 10^{-4} m, about z is less than 10^{-3} m. Position and orientation error of the moving platform can be diminished by increasing integral constant value. Oscillation occurs in rotation about x, y and z axes can be prevented by enhancing the derivative constant to create damping effect. It can increase the system response. Orientation and position of the moving platform for the desired and actual trajectories are shown in Figure 5.11 (a-f).



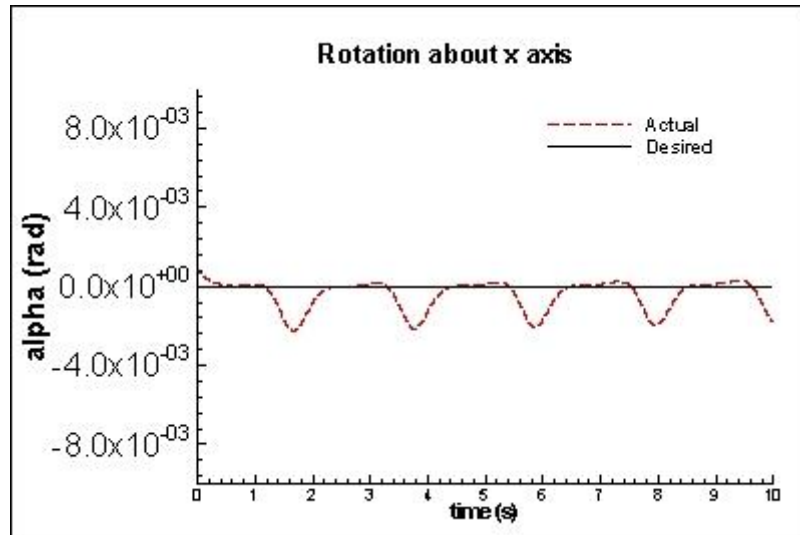
(a)



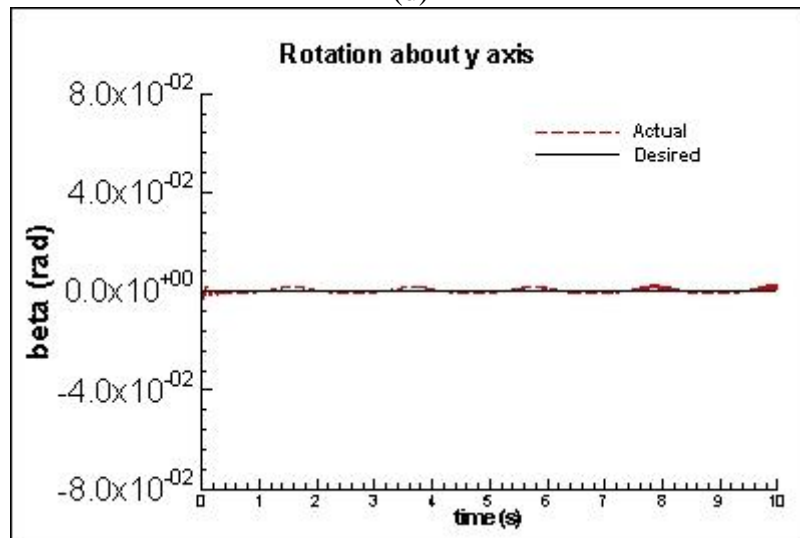
(b)



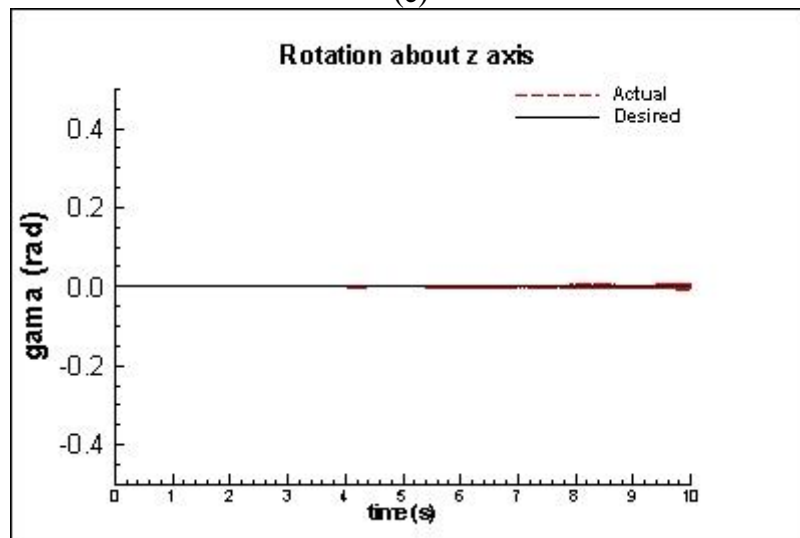
(c)



(d)



(e)



(f)

Figure 5.11 Desired and Actual Trajectories of the System with Linear Motors for Case Study I (a-f)

5.3.2 Case Study II

In the case study II, values of mass moment of inertia of the legs and moving platform and their mass are not changed. Only initial positions of the moving platform and the desired trajectories of the moving platform are taken as followings

Initial positions of the moving platform are given as following;

$$P(0) = [-1.3 \quad 0.2 \quad 1.2 \quad 0 \quad 0 \quad 0]^T$$

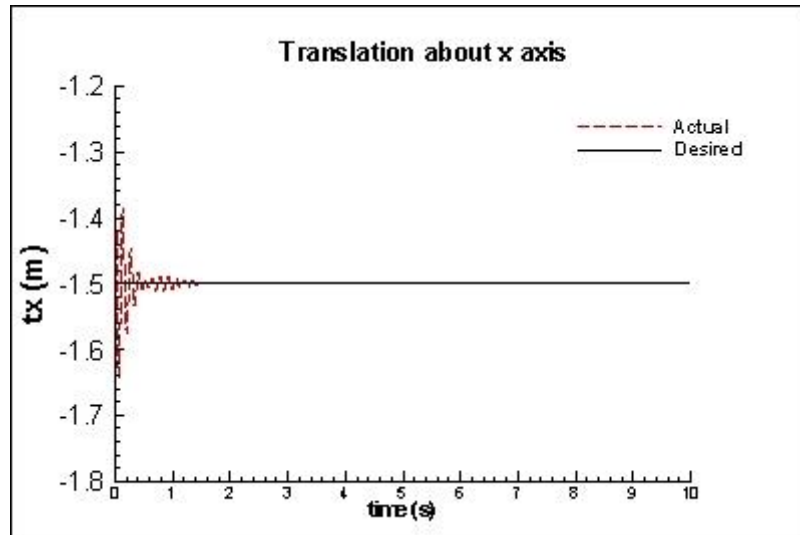
Desired trajectories of the moving platform are given as following;

$$P(t) = \begin{bmatrix} -1.5 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0.35\sin(\omega t) \end{bmatrix}$$

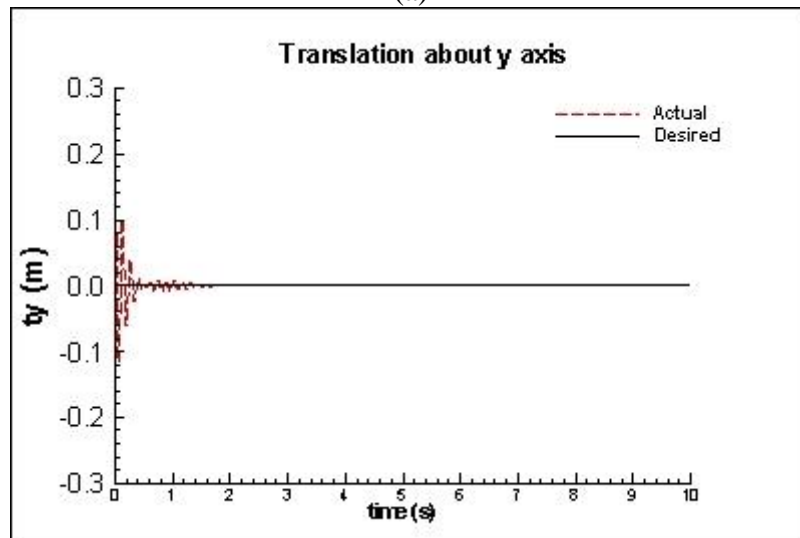
Where $\omega = 3\text{rad/s}$

5.3.2.1 Results and Discussion of Case Study II

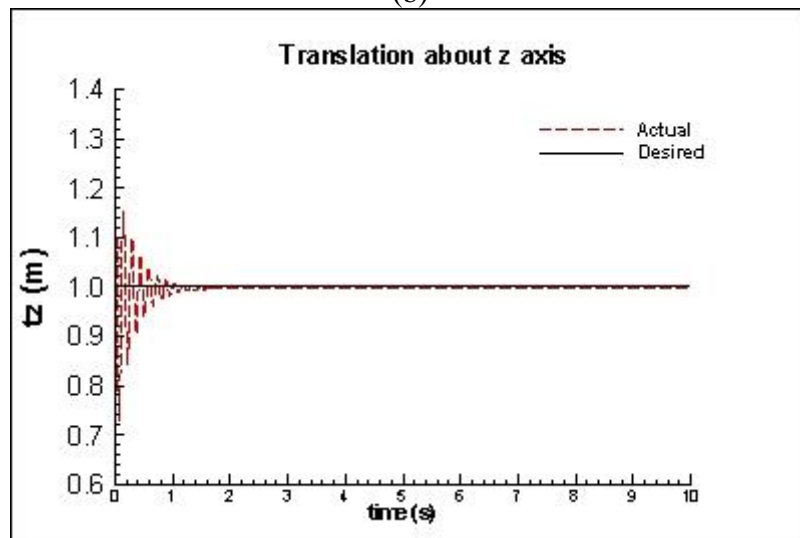
According to control results, it is observed that the rotation error about x and y axes are less than 10^{-7} rad and about z axis is less than 10^{-3} rad. Translation error about x and y axes are less than 10^{-7} m, about z axis is less than 10^{-3} m. Proportional constant can be increased to compensate system error in rotation about z axis and to decrease response time of the system. However, oscillation of the system may increase due to higher proportional constant. Orientation and position of the moving platform for the desired and actual trajectories are shown in Figure 5.12 (a-f).



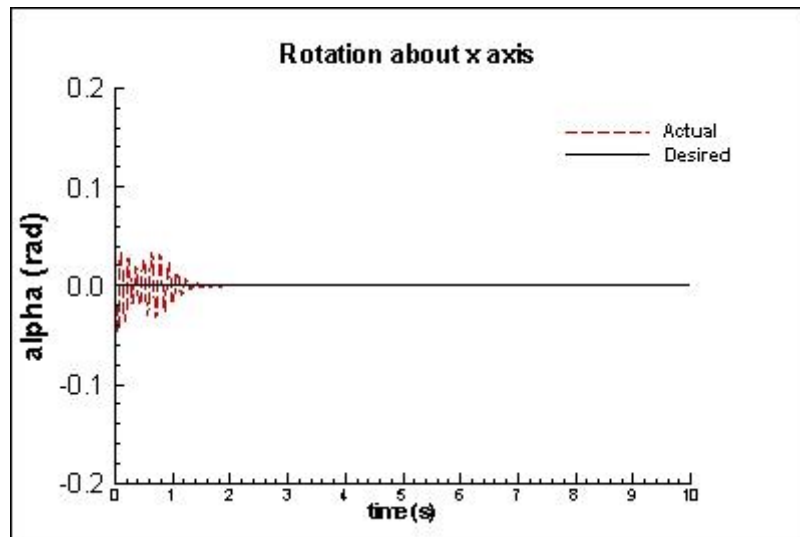
(a)



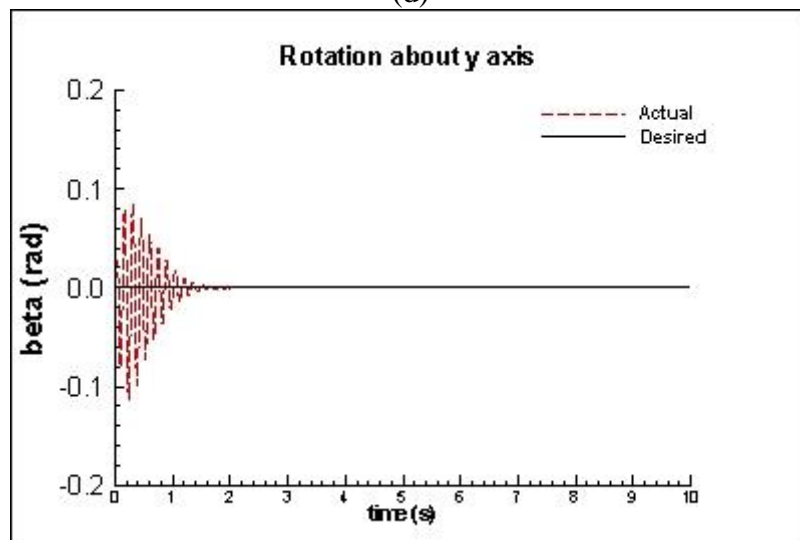
(b)



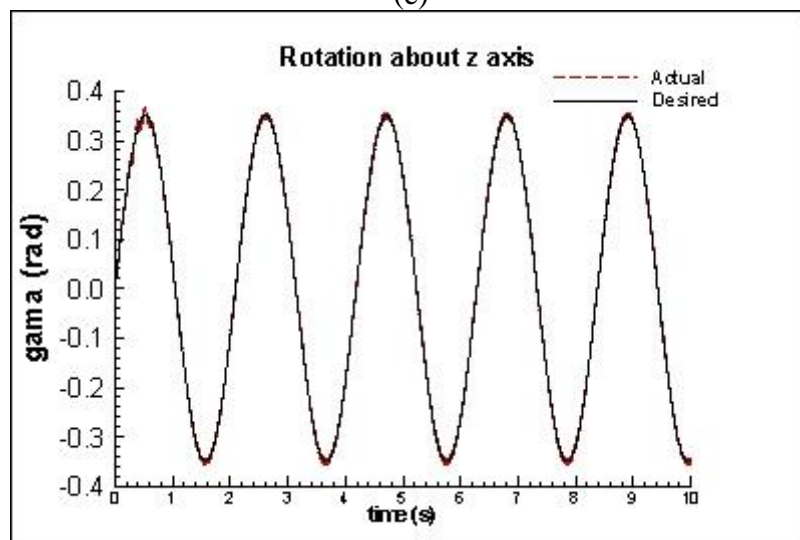
(c)



(d)



(e)



(f)

Figure 5.12 Desired and Actual Trajectories of the System with Linear Motors for Case Study II (a-f)

5.3.3 Case Study III

In the case study III, values of mass moment of inertia of the legs and moving platform and their mass are not changed. Only initial positions of the moving platform and the desired trajectories of the moving platform are taken as followings

Initial positions of the moving platform are given as following;

$$P(0) = [-1.3 \quad 0.2 \quad 1.2 \quad 0 \quad 0 \quad 0]^T$$

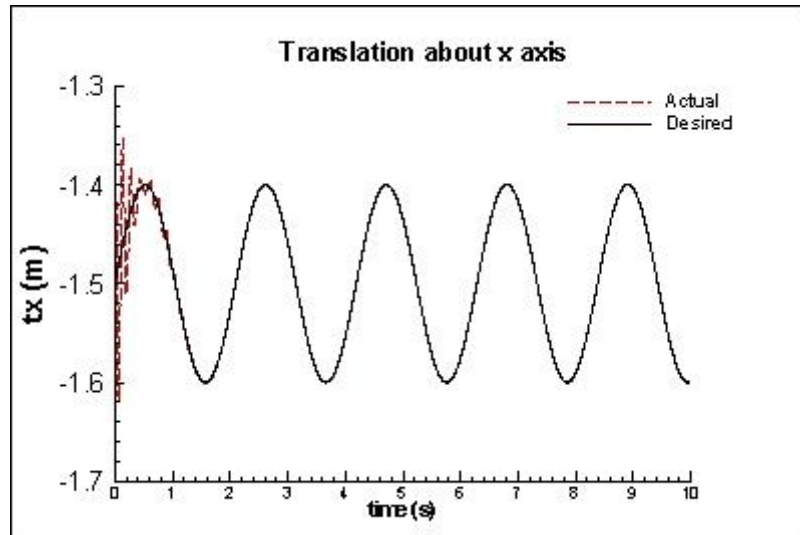
Desired trajectories of the moving platform are given as following;

$$P(t) = \begin{bmatrix} -1.5 + 0.1\sin(\omega t) \\ 0 \\ 1 \\ 0.1\sin(\omega t) \\ 0 \\ 0 \end{bmatrix}$$

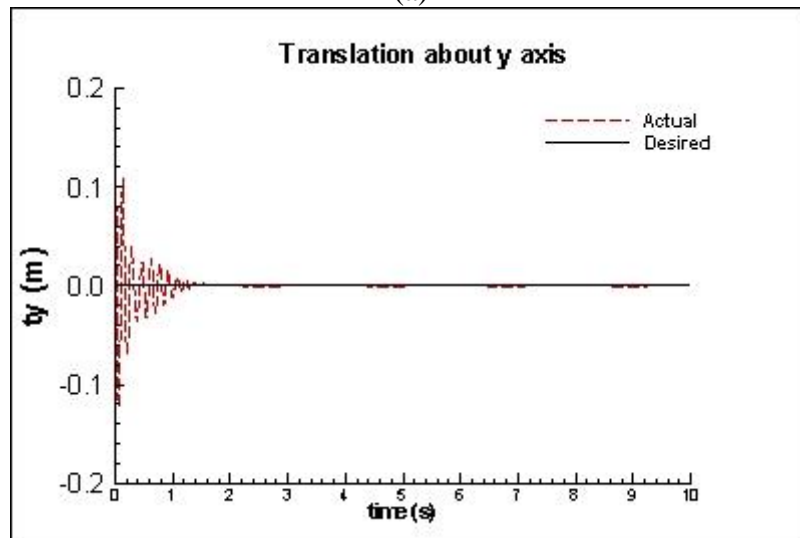
Where $\omega = 3\text{rad/s}$

5.3.3.1 Results and Discussion of Case Study III

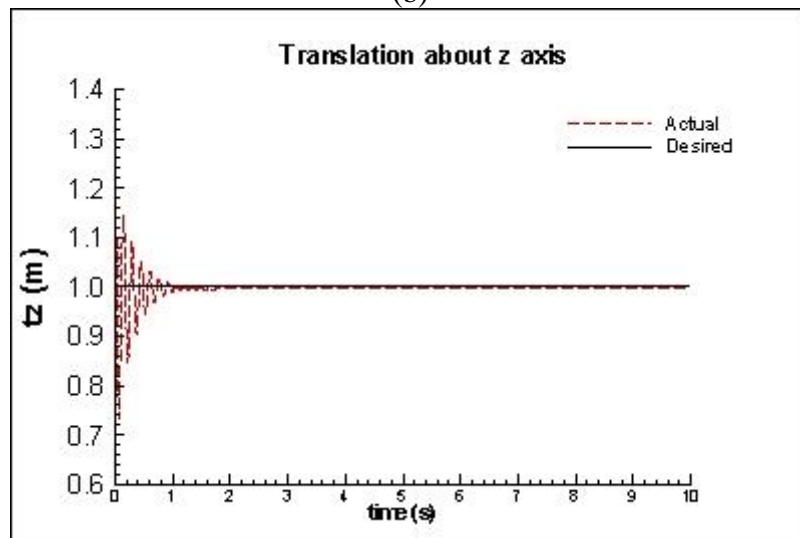
According to control results, it is observed that the rotation error about x, y and z axes are less than 10^{-3} rad. Translation error about x is less than 10^{-5} m, about y axis is less than 4×10^{-4} m and about z axis is less than 10^{-3} m. In rotation about z axis occurs limit cycle due to high stiffness of the mechanism. It can be eliminated by increased damping effect of the system. But, this process may increase response time of the system in other axis. Because of that, in order to provide optimum control of the mechanism, proper PID constants should be used. Orientation and position of the moving platform for the desired and actual trajectories are shown in Figure 5.13 (a-f).



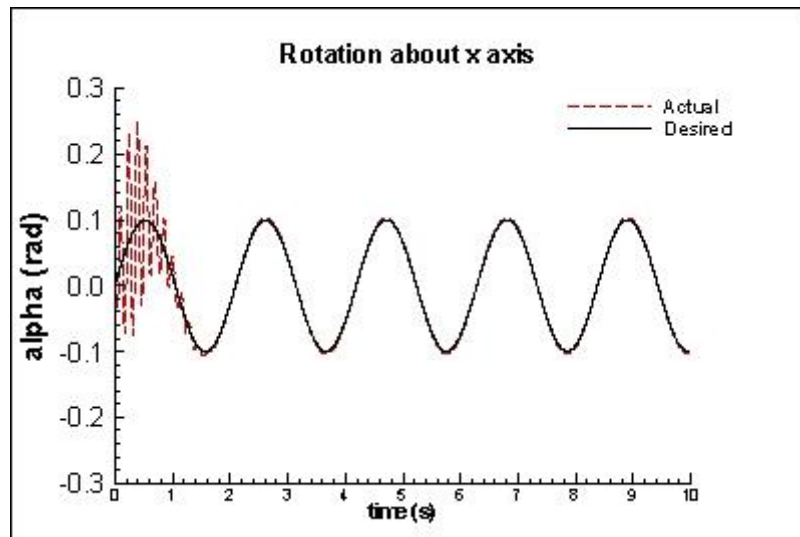
(a)



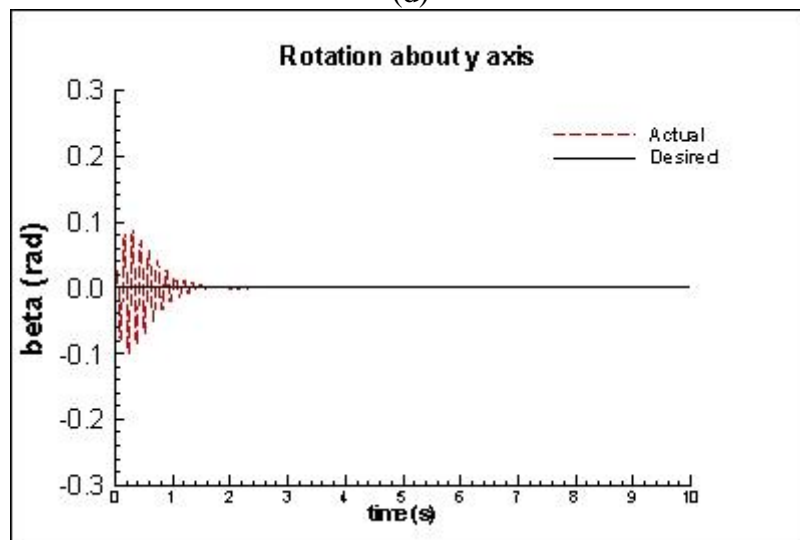
(b)



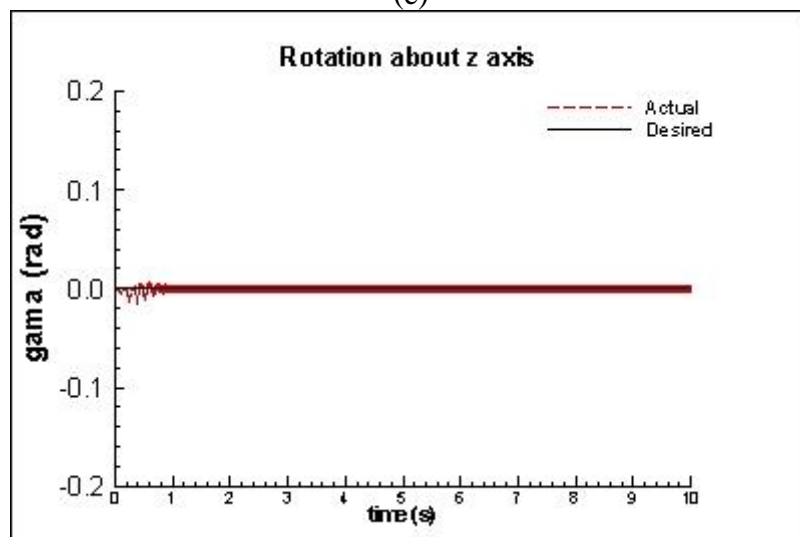
(c)



(d)



(e)



(f)

Figure 5.13 Desired and Actual Trajectories of the System with Linear Motors for Case Study III (a-f)

CHAPTER 6

CONCLUSIONS

In this study, analysis and design of the Stewart platform was carried out using linear motors. In the kinematic analysis, the inverse kinematic method was used to obtain position, velocity and acceleration of the mechanism. Moreover, velocity vector of the links was acquired by using inverse Jacobian matrix. The velocities of the upper and lower parts of leg and angular velocity of the actuator were obtained separately. In the dynamic analysis, the leg dynamic and moving platform dynamic were performed to obtain equation of motion of the mechanism by using Lagrange and Newton-Euler methods. It was concluded that these methods not only give more accurate results but also simple way to solve the system dynamics. In order to perform dynamic simulation, a MATLAB code was developed to compute the force of each leg according to prescribed trajectories. Dynamic simulation was performed both with and without inertia of the lower part of the leg and it was observed that the effect of the lower part inertia increase for high mass values.

In addition, linear motor dynamics was presented to derive transfer function and this function was used for system control. In the control stages, PID control of the mechanism was implemented in two ways in MATLAB/Simulink; the first stage was performed without linear motors and the second stage was performed using linear motors. Mathematical model block was obtained using equation of motion of system and inverse kinematic blocks were developed to compute the legs displacements error. Using proper PID constants, position control of the mechanism was implemented with respect to planned trajectories. According to case studies results, it was observed that the position errors were very small and system indicated good stability and performance in the first stage. The second stage was also carried out and the results are quite accurate. However, PID constants have higher value according to first stage, it was concluded that these higher values may originate because of non-linearity of the linear motor.

RECOMMENDATIONS FOR THE FUTURE WORK

In this study, PID control of the Stewart platform was performed by using linear motors. Application of an adaptive control to the linear motor on each leg can be performed due to change of forces. Comparison of the efficiencies of linear and DC motors can be carried out to drive a Stewart platform. Simmechanics can be used to perform control of the system and to implement the simulation of the mechanism. Further, to simplify the dynamic equation of the linear motor, friction, ripple and disturbance forces were not taken into account but in order to obtain more accurate results, new computations can be performed by taking into account these forces. Workspace and singularity problem is the most important for the parallel mechanism. To prevent singularity and to develop workspace of the system, new algorithms or new design can be improved. Due to the loads on the legs of the mechanism, stress analysis (buckling, bending etc.) can be performed.

REFERENCES

- Abedinnasab, M. H., Zohoor, H., Yoon, Y. J. (2012). Exploiting Higher Kinematic Performance-Using a 4-Legged Redundant PM Rather Than Gough-Stewart Platforms. INTECH Open Access Publisher.
- Akdağ, M., Karagülle, H., Malgaca, L. (2012). An integrated approach for simulation of mechatronic systems applied to a hexapod robot. *Mathematics and computers in simulation*. **82**, 818-835.
- Alrashidi, M., Yıldız, İ., Alrashdan, K., Esat, İ. (2009). Evaluating elbow joint kinematics with the Stewart platform mechanism. *WIT Transactions on Biomedicine and Health*. **13**, 181-189.
- Ay, S., Vatandas, O. E., Hacıoglu, A. (2012). Determination of the reachable workspace of 6-3 Stewart platform mechanism. In *Proc. of the World Congress on Engineering*. 3.
- Bai, X., Turner, J. D., Junkins, J. L. (2006). Dynamic analysis and control of a Stewart platform using a novel automatic differentiation method. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit* (pp. 21-24).
- Bangjun, L., Likun, P., Tingtao, M. (2012). Improving Dynamic Performance of Stewart Platforms through Optimal Design Based on Evolutionary Multi-objective Optimization Algorithms. In *Proceedings of the 1st International Conference on Mechanical Engineering and Material Science*. Atlantis Press.
- Ben-Horin, R., Shoham, M., Djerassi, S. (1998). Kinematics, dynamics and construction of a planarly actuated parallel robot. *Robotics and Computer-Integrated Manufacturing*, **14**, 163-172.
- Bingul, Z., Karahan, O. (2012). Dynamic Modeling and Simulation of Stewart Platform. INTECH Open Access Publisher.
- Bonev, I., A., Ryu, J. (2000). A new method for solving the direct kinematics of general 6-6 Stewart platform using three linear extra sensors. *Mechanism and Machine Theory*. **35**, 423-436.
- Brouwer, D. M., De Jong, B. R., Soemers, H. M. J. R. (2010). Design and modeling of a six DOFs MEMS-based precision manipulator. *Precision Engineering*, **34**, 307-319.
- Chen, H., Chen, W., Liu, J. (2007). Optimal design of Stewart platform safety mechanism. *Chinese Journal of Aeronautics*. **20**, 370-377. .

- Clerc, J. P., Tol, U. A., Wiens, G. J., Lindström, M., Steene, J., Jhaveri, N. K. (2002). Deburring using a micro/macro parallel kinematic machine. *Proceedings of 2002 Florida Conference on Recent Advances in Robotics*. Miami, Florida.
- Dasgupta, B., Mruthyunjaya, T. S. (1998 a). A Newton-Euler formulation for the inverse dynamics of the Stewart platform manipulator. *Mechanism and Machine Theory*, **33**, 1135-1152.
- Dasgupta, B., Mruthyunjaya, T. S. (1998 b). Singularity-free path planning for the Stewart platform manipulator. *Mechanism and Machine Theory*. **33**, 711-725.
- Davliakos, I., Papadopoulos, E. (2008). Model-based control of a 6-dof electrohydraulic Stewart–Gough platform. *Mechanism and Machine Theory*. **43**, 1385-1400.
- Gao, Z., Zhang, D., 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
- Gewald, D. (2006). Dynamics and Control of Hexapod Systems. Joint Advanced Student School (JASS).
- Gough, V. E., Whitehall, S. G. (1962) Universal tyre test machine. *Proceedings Ninth International Technical Congress F.I.S.I.T.A.* pp.117
- Guo, H. B., 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**, 61-72.
- Guo, H., Liu, Y., Liu, G., Li, H. (2008). Cascade control of a hydraulically driven 6-DOF parallel robot manipulator based on a sliding mode. *Control Engineering Practice*. **16**, 1055-1068.
- Güneri, B. (2007). Complete Dynamic Analysis of Stewart Platform Including Singularity Detection. *Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü. İzmir, 213141.*
- Harib, K., Srinivasan, K. (2003). Kinematic and dynamic analysis of Stewart platform-based machine tool structures. *Robotica*. **21**, 541-554.
- HIWIN. (2012) Linear motor system technical information. M99TE06-1202.
- Hsu, C. C., Fong, I. K. (2001). Motion control of a hydraulic Stewart platform with computed force feedback. *Journal of the Chinese Institute of Engineers*. **24**, 709-721.
- Huang, C. I., Chang, C. F., Yu, M. Y., Fu, L. C. (2004). Sliding-mode tracking control of the Stewart platform. IEEE. In *Control Conference, 2004. 5th Asian*. **1**, 562-569.
- Inner, B., Kucuk, S. (2013). A novel kinematic design, analysis and simulation tool for general Stewart platforms. *Simulation*. 0037549713482733, 1-22.
- Jakobovic, D., Budin, L. (2002) Forward kinematics of a Stewart platform mechanism. *Proc. 6th International Conference on. Intelligent Engineering System*.
- Jin, Y., Chen, I. M., Yang, G. (2009). Kinematic design of a family of 6-DOF partially decoupled parallel manipulators. *Mechanism and Machine Theory*. **44**, 912-922.

- Johnson, M. A., Moradi, M. H. (2005). PID control. Springer-Verlag London Limited.
- Kallio, P. (2002). Development of a parallel composite-joint piezohydraulic micromanipulator. Tampere University of Technology Publications.
- Kapur, P., Ranganath, R., Nataraju, B. S. (2007). Analysis of Stewart platform with flexural joints at singular configurations. In *12th IFToMM Congres, Besançon*.
- Karimi, D., Nategh, M. J. (2011). A statistical approach to the forward kinematics nonlinearity analysis of Gough-Stewart mechanism. *Journal of Applied Mathematics*, 2011.
- Kizir, S., Bingül, Z., Oysu, C., Küçük, S. (2011). Development and Control of a High Precision Stewart Platform. *International Journal of Technological Sciences*. **3**, 51-59.
- Korayem, M. H., Shokri, M. (2006). Maximum dynamic load carrying capacity of 6UPS-Stewart platform flexible joint manipulator. In *Robotics and Biomimetics, 2006. ROBIO'06. IEEE International Conference on* (pp. 727-732). IEEE.
- Korayem, M. H., Shokri, M. (2008). Maximum dynamic load carrying capacity of a 6UPS-Stewart platform manipulator. *Scientia Iranica*, **15**, 131-143.
- Korobeynikov, A., V., Turlapov, V., E. (2005). Modelling and evaluating of the Stewart platform. *International Conference Graphicon. Novosibirsk Akademgorodok, Russia*.
- Lee, S. H., Song, J. B., Choi, W. C., Hong, D. (2003). Position control of a Stewart platform using inverse dynamics control with approximate dynamics. *Mechatronics*. **13**, 605-619.
- Lee, T. H., Tan, K. K., Lim, S. Y., Dou, H. F. (2000). Iterative learning control of permanent magnet linear motor with relay automatic tuning. *Mechatronics*. **10**, 169-190.
- Lin, C. L., Jan, H. Y., Lin, J. R., Hwang, T. S. (2008). Singularity characterization and path planning of a new 3 links 6-DOFs parallel manipulator. *European Journal of Control*. **14**, 201-212.
- Liu, K., Fitzgerald, J., M., Lewis, F., L. (1993). Kinematic analysis of a Stewart platform manipulator. *IEEE Transactions on Industrial Electronics*. **40**, 282-293.
- Ma, O., Angeles, J. (1991). Architecture singularities of platform manipulators. In *Robotics and Automation, 1991. Proceedings. 1991 IEEE International Conference on*. 1542-1547.
- Meng, Q., Zhang, T., He, J. F., Song, J. Y., Han, J. W. (2010). Dynamic modelling of a 6-degree-of-freedom Stewart platform driven by a permanent magnet synchronous motor. *Journal of Zhejiang University Science C*. **11**, 751-761.
- Merlet, J. P. (1995). Determination of the orientation workspace of parallel manipulators. *Journal of intelligent and robotic systems*. **13**, 143-160.
- Mishra, A., Omkar, S. N. (2011). Singularity analysis and comparative study of six degree of freedom Stewart platform as a robotic arm by heuristic algorithms and simulated annealing. *International Journal of Engineering Science and Technology*. **3**, 644-659.

- Omran, A., Kassem, A. (2011). Optimal task space control design of a Stewart manipulator for aircraft stall recovery. *Aerospace Science and Technology*. **15**, 353-365.
- Ömürlü, V., Yildiz, İ. (2011). A Stewart platform as a FBW flight control unit. *Journal of Electrical Engineering*. **62**, 213-219.
- Pollard, W. L. V. (1942). Position controlling apparatus. *United State Patent Office*. No: 2,286,571, June 16, 1942.
- Serrano, F., Caballero, A. A., Yen, K. K., Brezina, T. (2007). Control of a Stewart Platform used in biomechanical systems. *Florida International University, Miami, Florida, USA*.
- Staicu, S. (2011). Dynamics of the 6-6 Stewart parallel manipulator. *Robotics and Computer-Integrated Manufacturing*, **27**, 212-220.
- Stewart, D. (1965). A platform with six degrees of freedom. *Proceedings of the Institution of Mechanical Engineers*. **180**, 371-386
- Takeda, Y., Funabashi, H. (1996). Kinematic and static characteristics of in-parallel actuated manipulators at singular points and in their neighborhood. *JSME international journal. Ser. C, Dynamics, control, robotics, design and manufacturing*. **39**, 85-93.
- Toz, M., Kucuk, S. (2013). Dexterous workspace optimization of an asymmetric six-degree of freedom Stewart–Gough platform type manipulator. *Robotics and Autonomous Systems*. **61**, 1516-1528.
- Tsai, L. W. (2000). Solving the inverse dynamics of a Stewart-Gough manipulator by the principle of virtual work. *Journal of Mechanical design*. **122**, 3-9.
- Wang, J., Gosselin, C. M. (1998). A new approach for the dynamic analysis of parallel manipulators. *Multibody System Dynamics*. **2**, 317-334.
- Wang, S. C., Hikita, H., Kubo, H., Zhao, Y. S., Huang, Z., Ifukube, T. (2003). Kinematics and dynamics of a 6 degree-of-freedom fully parallel manipulator with elastic joints. *Mechanism and Machine Theory*, **38**, 439-461.
- Wang, Y. (2008). Symbolic Kinematics and Dynamics Analysis and Control of a General Stewart Parallel Manipulator. ProQuest.
- Williams II, R. L. (2015). NotesBook Supplement for ME 4290/5290 Mechanics and Control of Robotic Manipulators.
- Wu, P., Xiong, H., Kong, J. (2012). Dynamic analysis of 6-SPS parallel mechanism. *International Journal of Mechanics and Materials in Design*, **8**, 121-128.
- Yang, C., Huang, Q., Han, J. (2012). Decoupling control for spatial six-degree-of-freedom electro-hydraulic parallel robot. *Robotics and Computer-Integrated Manufacturing*. **28**, 14-23.
- Yao, B., Xu, L. (2002). Adaptive robust motion control of linear motors for precision manufacturing. *Mechatronics*, **12**, 595-616.
- Yıldız, İ., Ömürlü, V., E., Ekicioğlu, Z., Güney, A. (2010). 6 Serbestlik dereceli paralel mekanizmadaki ileri kinematik analiz yöntemleri. *Otomatik Kontrol Uluslar Toplantısı, Kocaeli*.

Yildiz, I., Omurlu, V. E., Sagirli, A. (2009). A novel visualization technique in Bond-Graph method for modeling of a generalized Stewart platform. In *Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference on*. 780-785.

Yingjie, L., Wenbai, Z., Gexue, R. (2006). Feedback control of a cable-driven Gough-Stewart platform. *Robotics, IEEE Transactions on*. **22**, 198-202.

APPENDICES

Appendix A

Following properties related skew symmetric matrices are used for derivation of the equations.

$$\tilde{u}_i^T = -\tilde{u}_i \quad \text{A. 1}$$

$$\tilde{u}_i^T \tilde{u}_i = \mathbf{I} - u_i u_i^T \quad \text{A. 2}$$

$$u_i^T u_i = 1 \quad \text{A. 3}$$

$$\dot{u}_i = \frac{d}{dt} \left(\frac{L_i}{l_i} \right) = \frac{\dot{L}_i l_i - L_i \dot{l}_i}{l_i^2} = \frac{\dot{q}_i^B l_i - L_i u_i^T \dot{q}_i^B}{l_i^2} = \frac{\dot{q}_i^B (\mathbf{I} - u_i u_i^T)}{l_i} = \frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B \quad \text{A. 4}$$

The following equations are enables to derive M_1 with respect to time.

$$\begin{aligned} M_1 &= m_m \left(\mathbf{I} + \frac{d_p \tilde{u}_i^2}{l_i} \right) \left(\mathbf{I} + \frac{d_p \tilde{u}_i^2}{l_i} \right) \\ &= m_m \left[\mathbf{I} + 2 \frac{d_p \tilde{u}_i^2}{l_i} + \frac{d_p^2 \tilde{u}_i^4}{l_i^2} \right] = m_m \left[\mathbf{I} + 2 \frac{d_p}{l_i} (-\tilde{u}_i^T \tilde{u}_i) \left(\frac{d_p}{l_i} \right)^2 (-\tilde{u}_i^T \tilde{u}_i)^2 \right] \\ &= m_m \left[\mathbf{I} + 2 \frac{d_p}{l_i} (u_i u_i^T - \mathbf{I}) + \left(\frac{d_p}{l_i} \right)^2 (u_i u_i^T - \mathbf{I})^2 \right] \\ &= m_m \left[\mathbf{I} + 2 \frac{d_p}{l_i} u_i u_i^T - 2 \frac{d_p}{l_i} \mathbf{I} + \left(\frac{d_p}{l_i} \right)^2 [\mathbf{I} - 2u_i u_i^T + (u_i u_i^T)^2] \right] \\ &= m_m \left[\mathbf{I} + 2 \frac{d_p}{l_i} u_i u_i^T - 2 \frac{d_p}{l_i} \mathbf{I} + \left(\frac{d_p}{l_i} \right)^2 [\mathbf{I} - 2u_i u_i^T + u_i u_i^T u_i u_i^T] \right] \\ &= m_m \left[\mathbf{I} + 2 \frac{d_p}{l_i} u_i u_i^T - 2 \frac{d_p}{l_i} \mathbf{I} + \left(\frac{d_p}{l_i} \right)^2 [\mathbf{I} - u_i u_i^T] \right] \end{aligned}$$

$$\begin{aligned}
&= m_m \left[\mathbf{I} + 2 \frac{d_p}{l_i} u_i u_i^T - 2 \frac{d_p}{l_i} \mathbf{I} + \left(\frac{d_p}{l_i} \right)^2 \mathbf{I} - \left(\frac{d_p}{l_i} \right)^2 u_i u_i^T \right] \\
&= m_m \left[\left(1 - 2 \frac{d_p}{l_i} + \left(\frac{d_p}{l_i} \right)^2 \right) \mathbf{I} + 2 \frac{d_p}{l_i} u_i u_i^T - \left(\frac{d_p}{l_i} \right)^2 u_i u_i^T \right] \\
&= m_m \left[\left(1 - \frac{d_p}{l_i} \right)^2 \mathbf{I} + 2 \frac{d_p}{l_i} u_i u_i^T - \left(\frac{d_p}{l_i} \right)^2 u_i u_i^T \right] \tag{A.5}
\end{aligned}$$

The equation of M_1 is separated in three part to simplify the arrangement of the equations.

From the following equations (a), (b) and (c), Equation (4.7) is obtained.

$$\frac{d}{dt} \left(1 - \frac{d_p}{l_i} \right)^2 \mathbf{I} = 2 \left(1 - \frac{d_p}{l_i} \right) \frac{d_p \dot{l}_i}{l_i^2} \mathbf{I} = \frac{2 d_p u_i^T \dot{q}_i^B}{l_i^2} \mathbf{I} - \frac{2 d_p^2 u_i^T \dot{q}_i^B}{l_i^3} \mathbf{I} \tag{A.6}$$

$$\frac{d}{dt} \left(2 \frac{d_p}{l_i} u_i u_i^T \right) = - \frac{2 d_p u_i^T \dot{q}_i^B}{l_i^2} u_i u_i^T + 2 \frac{d_p}{l_i} \left(\frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B u_i^T + u_i \left(\frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B \right)^T \right) \tag{A.7}$$

$$\frac{d}{dt} \left(- \left(\frac{d_p}{l_i} \right)^2 u_i u_i^T \right) = \frac{2 d_p^2 u_i^T \dot{q}_i^B}{l_i^3} u_i u_i^T - \left(\frac{d_p}{l_i} \right)^2 \left(\frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B u_i^T + u_i \left(\frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B \right)^T \right) \tag{A.8}$$

Then, we arrange the M_2 to take derivative with respect to time and obtain equation (4.8).

$$\begin{aligned}
M_2 &= \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right)^T m_r \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right) = m_r \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right) \left(\frac{d_b \tilde{u}_i^T \tilde{u}_i}{l_i} \right) \\
&= m_r \left(\frac{d_b}{l_i} \right)^2 (\mathbf{I} - u_i u_i^T)^2 = m_r \left(\frac{d_b}{l_i} \right)^2 (\mathbf{I} - 2 u_i u_i^T + u_i u_i^T u_i u_i^T) \\
&= m_r \left(\frac{d_b}{l_i} \right)^2 (\mathbf{I} - u_i u_i^T) \tag{A.9}
\end{aligned}$$

$$\frac{dM_2}{dt} = -m_r \left[2 \left(\frac{d_b^2}{l_i^3} \right) (u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i) + \left(\frac{d_b}{l_i} \right)^2 \left(\frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B u_i^T + u_i \left(\frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B \right)^T \right) \right] \tag{A.10}$$

Finally, derivative of M_3 is taken with respect to time and equation (4.9) is obtained.

$$M_3 = \frac{\tilde{u}_i^T \tilde{u}_i}{l_i^2} (I_r + I_m) = \frac{(1 - u_i u_i^T)(I_r + I_m)}{l_i^2} \quad \text{A. 11}$$

$$\begin{aligned} \frac{dM_3}{dt} &= \frac{(I_r + I_m)[-(\dot{u}_i u_i^T + u_i \dot{u}_i^T)l_i^2 - 2l_i \dot{l}_i(1 - u_i u_i^T)]}{l_i^4} \\ &= \frac{(I_r + I_m)[-(\dot{u}_i u_i^T + u_i \dot{u}_i^T)l_i - 2u_i^T \dot{q}_i^B(1 - u_i u_i^T)]}{l_i^3} \\ &= -\frac{I_r + I_m}{l_i^3} \left[\left(\frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B u_i^T + u_i \left(\frac{\tilde{u}_i^T \tilde{u}_i}{l_i} \dot{q}_i^B \right)^T \right) l_i + 2u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i \right] \quad \text{A. 12} \end{aligned}$$

Appendix B

Using following equations, Equation (4.11) can be simplified. Equation (4.12) can be obtained.

$$\frac{\partial L_i^T}{\partial q_i^B} = I \quad \text{B.1}$$

$$\frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i} \right) = \frac{\partial}{\partial q_i^B} \left(\frac{1}{\sqrt{L_i^T L_i}} \right) = -\frac{1}{2} (L_i^T L_i)^{-3/2} \frac{\partial (L_i^T L_i)}{\partial q_i^B} = -\frac{L_i}{l_i^3} = -\frac{u_i}{l_i^2} \quad \text{B.2}$$

$$\frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i^2} \right) = \frac{\partial}{\partial q_i^B} \left(\frac{1}{L_i^T L_i} \right) = -(L_i^T L_i)^{-2} \frac{\partial (L_i^T L_i)}{\partial q_i^B} = -2 \frac{u_i}{l_i^3} \quad \text{B.3}$$

$$\begin{aligned} \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T L_i}{l_i^2} \right) &= \frac{\partial L_i^T}{\partial q_i^B} \frac{\dot{q}_i^B}{l_i^2} + \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T}{l_i^2} \right) L_i = \frac{\dot{q}_i^B}{l_i^2} + (\dot{q}_i^B)^T \left(-2 \frac{u_i}{l_i^3} \right) L_i \\ &= \frac{\dot{q}_i^B}{l_i^2} - 2 \frac{u_i u_i^T \dot{q}_i^B}{l_i^2} \end{aligned} \quad \text{B.4}$$

$$\begin{aligned} \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T L_i (\dot{q}_i^B)^T L_i}{l_i^3} \right) &= \frac{\partial}{\partial q_i^B} \left(\frac{1}{l_i} \frac{(\dot{q}_i^B)^T L_i}{l_i} \frac{(\dot{q}_i^B)^T L_i}{l_i} \right) \\ &= -\frac{u}{l_i^2} \left(\frac{(\dot{q}_i^B)^T L_i}{l_i} \frac{(\dot{q}_i^B)^T L_i}{l_i} \right) + \frac{1}{l_i} \frac{\partial}{\partial q_i^B} \left(\frac{(\dot{q}_i^B)^T L_i}{l_i} \frac{(\dot{q}_i^B)^T L_i}{l_i} \right) \\ &= -\frac{1}{l_i^2} (u_i (\dot{q}_i^B)^T u_i u_i^T \dot{q}_i^B) + \frac{2}{l_i} \frac{(\dot{q}_i^B)^T L_i}{l_i} \left(-\frac{(\dot{q}_i^B)^T u_i l_i}{l_i^2} + \frac{\dot{q}_i^B}{l_i} \right) \\ &= -\frac{1}{l_i^2} (u_i (\dot{q}_i^B)^T u_i u_i^T \dot{q}_i^B) + \frac{2}{l_i^2} (\dot{q}_i^B)^T u_i (\dot{q}_i^B - (\dot{q}_i^B)^T u_i u_i) \\ &= \frac{1}{l_i^2} (2u_i^T \dot{q}_i^B \tilde{u}_i^T \tilde{u}_i \dot{q}_i^B - u_i (\dot{q}_i^B)^T u_i u_i^T \dot{q}_i^B) \end{aligned} \quad \text{B.5}$$

Appendix C

MATLAB Codes

```
clear all ;clc
% Kinematic Equations of the Stewart Platform
syms t
a=input('enter the rotation angle about the x axis a=');
adot=diff(a);
addot=diff(adot);
b=input('enter the rotation angle about the y axis b=');
bdot=diff(b);
bddot=diff(bdot);
g=input('enter the rotation angle about the z axis g=');
gdot=diff(g);
gddot=diff(gdot);
% t is the translational vector between the origin of base and platform
tx=input('enter the translational vector in x direction tx=');
txdot=diff(tx);
txddot=diff(txdot);
ty=input('enter the translational vector in y direction ty=');
tydot=diff(ty);
tyddot=diff(tydot);
tz=input('enter the translational vector in z direction tz=');
tzdot=diff(tz);
tzddot=diff(tzdot);
t=input('enter the time value to calculate the functions t=');
txyz=[tx1;ty1;tz1];
tdot=[txdot1;tydot1;tzdot1];
tddot=[txddot1;tyddot1;tzddot1];
% rotational matrix wrt base platform
ax=cos(g1)*cos(b1);
ay=sin(g1)*cos(b1);
az=-sin(b1);
bx=cos(g1)*sin(b1)*sin(a1)-sin(g1)*cos(a1);
```

```

by=sin(g1)*sin(b1)*sin(a1)+cos(g1)*cos(a1);
bz=cos(b1)*sin(a1);
cx=cos(g1)*sin(b1)*cos(a1)+sin(g1)*sin(a1);
cy=sin(g1)*sin(b1)*cos(a1)-cos(g1)*sin(a1);
cz=cos(b1)*cos(a1);
R=[ax bx cx;ay by cy;az bz cz];
% angular velocity of the moving platform wrt base platform
wx=adot1+sin(b1)*gdot1;
wy=bdot1*cos(a1)-sin(a1)*cos(b1)*gdot1;
wz=bdot1*sin(a1)+cos(a1)*cos(b1)*gdot1;
w=[wx;wy;wz];
% skew symmetric form of the angular velocity
ws=[0 -wz wy;wz 0 -wx;-wy wx 0];
% angular acceleration of the moving platform
wxdot=addot1+sin(b1)*gddot1+bdot1*cos(b1)*gdot1;
wydot=bddot1*cos(a1)-sin(a1)*cos(b1)*gddot1-adot1*sin(a1)*bdot1-
adot1*cos(a1)*cos(b1)*gdot1+bdot1*sin(b1)*sin(a1)*gdot1;
wzdot=bddot1*sin(a1)+cos(a1)*cos(b1)*gddot1+adot1*cos(a1)*bdot1-
adot1*sin(a1)*cos(b1)*gdot1-bdot1*sin(b1)*cos(a1)*gdot1;
wddot=[wxdot;wydot;wzdot];
% Generalized velocity and acceleration of the platform frame
P=[txyz;a1;b1;g1];
Pdot=[tdot;w];
Pddot=[tddot;wddot];
% qi/p and bi are the coordinate of the upper and lower junction points, respectively
for i=1:6
    % moving platform coordinates
    fprintf('%d. cordinate ',i)
        x1=input('enter the position of the xp=');
        y1=input('enter the position of the yp=');
        z1=input('enter the position of the zp=');
        theate_p=input('enter the angle of the upper point wrt platform coordinate=');
        x1=rp*cosd(theate_p);
        y1=rp*sind(theate_p);

```

```

    z1=input('enter positon of the zp=');
qp=[x1;y1;z1];
% skew symmetric matrix form of platform coordinate
qps=[0 -z1 y1;z1 0 -x1;-y1 x1 0];
% base platform coodinates
fprintf('%d. cordinate ',i)
    x=input('enter the position of the xb=');
    y=input('enter the position of the yb=');
    z=input('enter the position of the zb=');
bi=[x;y;z];
% qi/B is the position vector wrt base platform
qb=txyz+R*qp;
% Li is the position vector of the leg
Li=qb-bi;
% li is the length of the leg
A=Li(1,1);
B=Li(2,1);
C=Li(3,1);
li=sqrt(A^2+B^2+C^2);
% unit vector of the leg in the direction of Li
u=Li/li;
% skew symmetric form of the unit vector
us=[0 -u(3,1) u(2,1);u(3,1) 0 -u(1,1);-u(2,1) u(1,1) 0];
if i==1
    qp1=qp;
    qps1=qps;
    u1=u;
    us1=us;
    l1=li;
    b1i=bi;
else if i==2
    qp2=qp;
    qps2=qps;
    u2=u;

```

```
us2=us;
l2=li;
b2i=bi;
else if i==3
    qp3=qp;
    qps3=qps;
    u3=u;
    us3=us;
    l3=li;
    b3i=bi;
else if i==4
    qp4=qp;
    qps4=qps;
    u4=u;
    us4=us;
    l4=li;
    b4i=bi;
else if i==5
    qp5=qp;
    qps5=qps;
    u5=u;
    us5=us;
    l5=li;
    b5i=bi;
else if i==6
    qp6=qp;
    qps6=qps;
    u6=u;
    us6=us;
    l6=li;
    b6i=bi;
    break
end
end
```



```

        end
    end
end
end
end
% inverse Jacobian matrix
J=[u1' (R*qps1*(R')*u1)';u2' (R*qps2*(R')*u2)';u3' (R*qps3*(R')*u3)';u4'
(R*qps4*(R')*u4)';u5' (R*qps5*(R')*u5)';u6' (R*qps6*(R')*u6)'];
% velocity of the link is obtained by using Jacobian matrix
Ldot=J*Pdot;
% qi/B velocity vector wrt base platform(generalized coordinate)
I=[1 0 0;0 1 0;0 0 1];
qbdot1=[I R*(qps1')*R']*Pdot;
qbdot2=[I R*(qps2')*R']*Pdot;
qbdot3=[I R*(qps3')*R']*Pdot;
qbdot4=[I R*(qps4')*R']*Pdot;
qbdot5=[I R*(qps5')*R']*Pdot;
qbdot6=[I R*(qps6')*R']*Pdot;
% qi/B acceleration vector wrt base platform(generalized coordinate)
qbddot1=[I R*(qps1')*R']*Pddot+(ws*ws*R*qp1);
qbddot2=[I R*(qps2')*R']*Pddot+(ws*ws*R*qp2);
qbddot3=[I R*(qps3')*R']*Pddot+(ws*ws*R*qp3);
qbddot4=[I R*(qps4')*R']*Pddot+(ws*ws*R*qp4);
qbddot5=[I R*(qps5')*R']*Pddot+(ws*ws*R*qp5);
qbddot6=[I R*(qps6')*R']*Pddot+(ws*ws*R*qp6);
% wact is the velocity of the actuator
wm1=(us1*qbdot1)/l1;
wm2=(us2*qbdot2)/l2;
wm3=(us3*qbdot3)/l3;
wm4=(us4*qbdot4)/l4;
wm5=(us5*qbdot5)/l5;
wm6=(us6*qbdot6)/l6;
% velocity of the moving part of the leg
dp=input('enter the distance of the moving part of the leg=');

```

```

Vm1=(I+((dp*us1*us1)/l1))*qbdot1;
Vm2=(I+((dp*us2*us2)/l2))*qbdot2;
Vm3=(I+((dp*us3*us3)/l3))*qbdot3;
Vm4=(I+((dp*us4*us4)/l4))*qbdot4;
Vm5=(I+((dp*us5*us5)/l5))*qbdot5;
Vm6=(I+((dp*us6*us6)/l6))*qbdot6;
% velocity of the rotating part of the leg
db=input('enter the distance of the rotating part of the leg=');
Vr1=((db*(us1')*us1)/l1)*qbdot1;
Vr2=((db*(us2')*us2)/l2)*qbdot2;
Vr3=((db*(us3')*us3)/l3)*qbdot3;
Vr4=((db*(us4')*us4)/l4)*qbdot4;
Vr5=((db*(us5')*us5)/l5)*qbdot5;
Vr6=((db*(us6')*us6)/l6)*qbdot6;
% Dynamic Equation of the Stewart Platform
Ir=input('enter the mass moment of inertia of rotating part=');
Im=input('enter the mass moment of inertia of moving part=');
Ip=input('enter the mass moment of inertia of platform=');
Mr=input('enter the mass of the rotating part=');
Mm=input('enter the mass of the moving part=');
Mp=input('enter the mass of the platform=');
g=[0;0;-9.81];
% center point coordinate of the platform
theate_cont=input('enter the control point angle wrt reference coordinate=');
rc=input('control point radius=');
xc=rc*cosd(theate_cont);
yc=rc*sind(theate_cont);
zc=0;
qc=[xc;yc;zc];
qcs=[0 -zc yc;zc 0 -yc xc 0];
% gravitational force for rotating and moving part of the leg
FMm1=(I+((dp*us1*us1)/l1))*Mm*g;
FMm2=(I+((dp*us2*us2)/l2))*Mm*g;
FMm3=(I+((dp*us3*us3)/l3))*Mm*g;

```

$$FMm4=(I+((dp*us4*us4)/l4))*Mm*g;$$

$$FMm5=(I+((dp*us5*us5)/l5))*Mm*g;$$

$$FMm6=(I+((dp*us6*us6)/l6))*Mm*g;$$

$$FMr1=((db*(us1')*us1)/l1)*Mr*g;$$

$$FMr2=((db*(us2')*us2)/l2)*Mr*g;$$

$$FMr3=((db*(us3')*us3)/l3)*Mr*g;$$

$$FMr4=((db*(us4')*us4)/l4)*Mr*g;$$

$$FMr5=((db*(us5')*us5)/l5)*Mr*g;$$

$$FMr6=((db*(us6')*us6)/l6)*Mr*g;$$

% Lagrange formulation three (M1,M2,M3) parts

$$M11=((I+((dp*us1*us1)/l1)))*Mm*(I+((dp*us1*us1)/l1));$$

$$M12=((I+((dp*us2*us2)/l2)))*Mm*(I+((dp*us2*us2)/l2));$$

$$M13=((I+((dp*us3*us3)/l3)))*Mm*(I+((dp*us3*us3)/l3));$$

$$M14=((I+((dp*us4*us4)/l4)))*Mm*(I+((dp*us4*us4)/l4));$$

$$M15=((I+((dp*us5*us5)/l5)))*Mm*(I+((dp*us5*us5)/l5));$$

$$M16=((I+((dp*us6*us6)/l6)))*Mm*(I+((dp*us6*us6)/l6));$$

$$M21=(((db*(us1')*us1)/l1)*Mr*((db*(us1')*us1)/l1);$$

$$M22=(((db*(us2')*us2)/l2)*Mr*((db*(us2')*us2)/l2);$$

$$M23=(((db*(us3')*us3)/l3)*Mr*((db*(us3')*us3)/l3);$$

$$M24=(((db*(us4')*us4)/l4)*Mr*((db*(us4')*us4)/l4);$$

$$M25=(((db*(us5')*us5)/l5)*Mr*((db*(us5')*us5)/l5);$$

$$M26=(((db*(us6')*us6)/l6)*Mr*((db*(us6')*us6)/l6);$$

$$M31=(((us1')*us1)/(l1^2))* (Ir+Im);$$

$$M32=(((us2')*us2)/(l2^2))* (Ir+Im);$$

$$M33=(((us3')*us3)/(l3^2))* (Ir+Im);$$

$$M34=(((us4')*us4)/(l4^2))* (Ir+Im);$$

$$M35=(((us5')*us5)/(l5^2))* (Ir+Im);$$

$$M36=(((us6')*us6)/(l6^2))* (Ir+Im);$$

% Coriolis-Centrifugal equation for the legs

$$B1=((dp*Mm)/(l1^2))*(u1*(qbdot1')*(us1')*us1+(u1')*qbdot1*(us1')*us1+(us1')*us1*qbdot1*(u1'))$$

$$(((dp^2)*Mm+(db^2)*Mr)/(l1^3))*((u1')*qbdot1*(us1')*us1+(us1')*us1*qbdot1*(u1'))-((2*(Ir+Im))/(l1^3))*((us1')*us1*qbdot1*(u1'));$$

$$B2 = ((dp * Mm) / (l2^2)) * (u2 * (qbdot2') * (us2') * us2 + (u2') * qbdot2 * (us2) * us2 + (us2') * us2 * qbdot2 * (u2')) -$$

$$(((dp^2) * Mm + (db^2) * Mr) / (l2^3)) * ((u2') * qbdot2 * (us2') * us2 + (us2') * us2 * qbdot2 * (u2')) - ((2 * (Ir + Im)) / (l2^3)) * ((us2') * us2 * qbdot2 * (u2'));$$

$$B3 = ((dp * Mm) / (l3^2)) * (u3 * (qbdot3') * (us3') * us3 + (u3') * qbdot3 * (us3) * us3 + (us3') * us3 * qbdot3 * (u3')) -$$

$$(((dp^2) * Mm + (db^2) * Mr) / (l3^3)) * ((u3') * qbdot3 * (us3') * us3 + (us3') * us3 * qbdot3 * (u3')) - ((2 * (Ir + Im)) / (l3^3)) * ((us3') * us3 * qbdot3 * (u3'));$$

$$B4 = ((dp * Mm) / (l4^2)) * (u4 * (qbdot4') * (us4') * us4 + (u4') * qbdot4 * (us4) * us4 + (us4') * us4 * qbdot4 * (u4')) -$$

$$(((dp^2) * Mm + (db^2) * Mr) / (l4^3)) * ((u4') * qbdot4 * (us4') * us4 + (us4') * us4 * qbdot4 * (u4')) - ((2 * (Ir + Im)) / (l4^3)) * ((us4') * us4 * qbdot4 * (u4'));$$

$$B5 = ((dp * Mm) / (l5^2)) * (u5 * (qbdot5') * (us5') * us5 + (u5') * qbdot5 * (us5) * us5 + (us5') * us5 * qbdot5 * (u5')) -$$

$$(((dp^2) * Mm + (db^2) * Mr) / (l5^3)) * ((u5') * qbdot5 * (us5') * us5 + (us5') * us5 * qbdot5 * (u5')) - ((2 * (Ir + Im)) / (l5^3)) * ((us5') * us5 * qbdot5 * (u5'));$$

$$B6 = ((dp * Mm) / (l6^2)) * (u6 * (qbdot6') * (us6') * us6 + (u6') * qbdot6 * (us6) * us6 + (us6') * us6 * qbdot6 * (u6')) -$$

$$(((dp^2) * Mm + (db^2) * Mr) / (l6^3)) * ((u6') * qbdot6 * (us6') * us6 + (us6') * us6 * qbdot6 * (u6')) - ((2 * (Ir + Im)) / (l6^3)) * ((us6') * us6 * qbdot6 * (u6'));$$

% platform equation Bp for the platform

$$a0 = [0 \ 0 \ 0; 0 \ 0 \ 0; 0 \ 0 \ 0];$$

$$Bp = [a0 \ a0; a0 \ ws * R * Ip * (R)];$$

% total Coriolis-Centrifugal equation (B)

$$B_total = [I; R * qps1 * (R')] * B1 * [I; R * (qps1') * (R')] * Pdot + [I; R * qps2 * (R')] * B2 * [I; R * (qps2') * (R')] * Pdot + [I; R * qps3 * (R')] * B3 * [I; R * (qps3') * (R')] * Pdot + [I; R * qps4 * (R')] * B4 * [I; R * (qps4') * (R')] * Pdot + [I; R * qps5 * (R')] * B5 * [I; R * (qps5') * (R')] * Pdot + [I; R * qps6 * (R')] * B6 * [I; R * (qps6') * (R')] * Pdot;$$

$$M_total = [I; R * qps1 * (R')] * (M11 + M21 + M31) * ws * ws * R * qp1 + [I; R * qps2 * (R')] * (M12 + M22 + M32) * ws * ws * R * qp2 + [I; R * qps3 * (R')] * (M13 + M23 + M33) * ws * ws * R * qp3 + [I; R * qps4 * (R')] * (M14 + M24 + M34) * ws * ws * R * qp4 + [I; R * qps5 * (R')] * (M15 + M25 + M35) * ws * ws * R * qp5 + [I; R * qps6 * (R')] * (M16 + M26 + M36) * ws * ws * R * qp6;$$

$$BPdot = Bp * Pdot + B_total + [Mp * I; Mp * R * (qcs') * (R')] * (ws * ws * R * qc) + M_total;$$

% platform inertia

```

MP=[Mp*I Mp*R*qcs*(R');Mp*R*qcs*(R') Mp*R*qcs*(qcs')*(R')+R*Ip*(R)];
% Leg Inertia (M)
M_tot=[I;R*qps1*(R')]*(M11+M21+M31)*[I R*(qps1')*(R')] + [I;R*qps2*(R')] *
(M12+M22+M32) * [I R*(qps2')*(R')] + [I;R*qps3*(R')]*(M13+M23+M33)*[I
R*(qps3')*(R')] + [I;R*qps4*(R')] * (M14+M24+M34)*[I R*(qps4')*(R')] +
[I;R*qps5*(R')] * (M15+M25+M35)*[I R*(qps5')*(R')] + [I;R*qps6*(R')]
*(M16+M26+M36) * [I R*(qps6')*(R)];
M=MP+M_tot;
% Gravity part of equation of motion
K_tot=[I;R*qps1*(R')]*(FMm1+FMr1)+[I;R*qps2*(R')]*(FMm2+FMr2)+[I;R*qps3
*(R')]*(FMm3+FMr3)+[I;R*qps4*(R')]*(FMm4+FMr4)+[I;R*qps5*(R')]*(FMm5+
FMr5)+[I;R*qps6*(R')]*(FMm6+FMr6);
K=-[Mp*gr;Mp*R*qcs*(R')*gr]-K_tot;
% general equation of motion of the Stewart Platform
JF=M*Pddot+BPdot+K;
fprintf('The force value for %d.t value',k);
F=inv(J)*JF

```