T.C.
BAHÇEŞEHİR UNIVERSITY

MOTION AND FORCE CONTROL OF UNDERACTUATED ROBOT MANIPULATORS BASED ON PROJECTED INVERSE DYNAMICS

Master's Thesis

SEDA KORKMAZ

İSTANBUL, 2016

T.C.
BAHÇEŞEHİR UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
MECHATRONICS ENGINEERING

MOTION AND FORCE CONTROL OF UNDERACTUATED ROBOT MANIPULATORS BASED ON PROJECTED INVERSE DYNAMICS

Master's Thesis

SEDA KORKMAZ

ADVISOR: ASST. PROF. MEHMET BERKE GUR

İSTANBUL, 2016

THE REPUBLIC OF TURKEY
BAHCESEHIR UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
MECHATRONICS ENGINEERING

Name of the thesis: Motion and Force control of Underactuated Robot Manipulators Based on Projected Inverse Dynamics
Name/Last Name of the Student: Seda Korkmaz
Date of the Defense of Thesis:

The thesis has been approved by the Graduate School of Natural and Applied Sciences.

Signature
Assoc. Prof. Nafiz ARICA
Program Coordinator

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

Signature
Asst. Prof. Mehmet Berke GÜR
Program Coordinator

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

Examining Comittee Members	Signature
Thesis Supervisor Asst. Prof. Mehmet Berke GÜR	----------------------------------
Member Ph.D. Yongki YOON	----------------------------------
Member Asst. Prof. Uğur TÜMERDEM	----------------------------------

ABSTRACT

MOTION AND FORCE CONTROL OF UNDERACTUATED ROBOT MANIPULATORS BASED ON PROJECTED INVERSE DYNAMICS

Seda Korkmaz

Mechatronics Engineering

Thesis Supervisor: Asst. Prof. Mehmet Berke Gür

April 2016, 68 pages

The control of underactuated robot manipulators is a challenging problem since they have less control input than their degrees of freedom. Underactuated robotic systems are seen in different types in real applications which have similar problems that the uncontrolled degree of freedom makes accurate manipulation difficult.

This thesis focuses on the position-force control of underactuated robots using projected inverse dynamics based control schemes. The projected inverse based control approach considers the constraint forces in addition to the motion. The linear projection matrix decomposes the joint input torque to its two components which creates the motion and constraint force. This fact allows developing two different feedback control loops and implementing them simultaneously. Also, the relation between the acceleration and constraint force yields a formulation that the constraint forces can be estimated by the knowledge of control input.

In order to build the model of dynamics and to design controllers, it is important to observe the behaviors of robots during the simulation. V-REP creates a visualization of all the bodies in mechanical model. This simulator also gives an opportunity to simulate motions of robotic systems by specifying the mass properties of bodies and to initiate and observe body motions.

In this study, in order to develop the controllers, first, the necessary definitions are made and the dynamic formulations are explained which is basis of the control approach. Then, the kinematic and dynamic analysis is demonstrated for the PHANTOM Omni manipulator, which is a RRR articulated robot platform used in simulations and experimental studies. Eventually, the underactuated controller schemes are explained and they are implemented to the robot via Matlab/V-REP and experimentally.

Underactuation is imposed artificially without applying any control input to one of the joints making it passive. In order to compensate the lost torque at passive joint, the active joint that produces torque in the same direction with the passive joint is used. The PHANTOM Omni robot which used in simulation and experimental studies has a structure that only last two joints generate torque in the same direction. Hence, when one of them is chosen as passive, only one joint remains to control. There are also differences between the control performances while robot operates in vertical and horizontal directions. If there is no constraint in vertical axis, the precision of control decreases due to the gravitational affects. Nevertheless, according to the results, it can be asserted that the desired control is achieved with minor steady state errors in an admissible range.

Keywords: Underactuation, Projected Inverse Dynamics, Motion/Force Control

ÖZET

EKSİK TAHRİKLİ ROBOT KOLLARIN TERS İZDÜŞÜM DİNAMİĞİ YÖNTEMİ İLE HAREKET VE KUVVET KONTROLÜ

Seda Korkmaz

Mekatronik Mühendisliği

Tez Danışmanı: Yrd. Doç. Mehmet Berke Gür

Nisan 2016, 68 sayfa

Kontrol giriş sayısının robotun serbestlik derecesinden az olması, eksik tahrikli manipülatörlerin kontrolünü zorlu bir problem haline getirmektedir. Eksik tahrikli robotlar pek çok farklı türde olabileceği gibi, tümünde ortak sorun, kontrol edilmeyen serbestlik derecesi nedeni ile robotun hedeflenen performansı yerine getirmesinin güçleşmesidir.

Bu tez çalışmasında, eksik tahrikli robot manipülatörlerinin kontrolü için izdüşüm operatörü tabanlı dinamik çözüm ve denetleyici tasarımları üzerinde durulmuştur. Bu yöntem ile hareket kontrolü ile birlikte, robotların herhangi bir nesne ile temasları durumdaki kuvvet kontrolü de ele alınmaktadır. Giriş kuvvetinin izdüşüm operatörü ile harekete ve tepki kuvvetlerine neden olan bileşenlerine ayrılması, iki bağımsız geri beslemeli kontrol şemasının geliştirilmesi ve aynı anda uygulanmasına olanak sağlamaktadır. Bununla birlikte hareket ve tepki kuvvetleri arasındaki bu ilişki, hareketin bilinmesi ile kuvvetlerin de öngörülebildiği bir formülasyon sunmaktadır.

Robotların hareket kabiliyetini benzetim sırasında yakından gözlemlemek, dinamiklerinin benzetimi ve denetleyici tasarımında oldukça önemlidir. V-REP, robotun mekanik modelini görselleştirilirken, aynı zamanda modellenen dinamiğin ve tasarlanan denetleyicilerin katı model üzerindeki sonuçlarını da gözlemleme fırsatı vermektedir.

Bu çalışmada, test için kullanılan denetleyicilerin elde edilmesi için öncelikle gerekli tanımlamalar yapılmış, ardından kontrol yönteminin temel aldığı dinamik formülasyon açıklanmıştır. Denetleyicilerin uygulandığı robot manipülatörü olan PHANTOM Omni robot için kinematik ve dinamik analizler yapılmıştır. Son olarak, denetleyiciler açıklanmış ve Matlab ve V-REP benzetim ortamlarında ve deneysel olarak testleri yapılmıştır.

Herhangi bir ekleme kontrol girişi uygulanmaması ile o eklem yapay olarak pasif hale getirilerek, eksik tahrikli robot kontrol şemaları uygulanmıştır. Pasif olan eklemdeki torku karşılamak için, onunla aynı yönde tork üreten aktif eklem kullanılmıştır. PHANTOM Omni robotta ikinci ve üçüncü eklemler aynı yönde tork üretmektedir, bu nedenle herhangi biri pasif olarak seçildiğinde kontrol tek bir eklem ile sağlanmaktadır. Ayrıca robotun düşey ve yatay eksenlerindeki hareketleri arasında da yerçekimi etkisi nedeni ile farklılıklar görülmektedir. Robotun düşey eksendeki hareketini sınırlayıcı bir yüzey olmaması durumunda yerçekimi etkisi ile kontrol hassasiyeti düşmektedir. Bununla birlikte, elde edilen sonuçlar, kontrol hedeflerine kalıcı durumda kabul edilebilir ölçüde küçük hatalarla varılabildiğini göstermektedir.

Anahtar Kelimeler: Eksik Tahrikli Robot Kontrolü, İzdüşüm Operatörü Tabanlı Dinamik Modelleme, Hareket/Kuvvet Kontrolü

TABLE OF CONTENTS

TABLES ix

FIGURES x

ABBREVIATIONS xii

SYMBOLS xiii

1. INTRODUCTION 1

  1.1 PROBLEM DEFINITION..3

  1.2 LITERATURE SURVEY 5

2. DYNAMICS BASED ON ORTHOGONAL PROJECTIONS 7

  2.1 PROJECTION OPERATORS 7

  2.2 INVERSE AND DIRECT DYNAMICS 10

    2.2.1 Constraint Force..11

    2.2.2 Decomposition of Motion and Constraint Force 12

3. UNDERACTUATED ROBOT MANIPULATORS 14

  3.1 INTRODUCTION 14

  3.2 PHANTOM OMNI ROBOT MANIPULATOR...15

    3.2.1 Kinematic Model 16

    3.2.2 Dynamic Analysis 18

4. CONTROL OF UNDERACTUATED MANIPULATORS 23

  4.1 JOINT SPACE CONTROL OF UNDERACTUATED MANIPULATORS 24

    4.1.1 Motion Control 25

    4.1.2 Force Control...27

    4.1.3 Hybrid Control Method..28

  4.2 OPERATIONAL SPACE CONTROL OF UNDERACTUATED MANIPULATORS 32

4.2.1 Motion Control in Operational Space 33

4.2.2 Operational Space Hybrid Motion/Force Control Based on Projected Inverse Dynamics 36

5. SIMULATION AND RESULTS 39

5.1 TEST OF THE CONTROLLER PERFORMANCE IN CASE OF A CONSTRAINT IN VERTICAL AXIS 40

5.1.1 Joint Space Controllers 40

5.1.2 Operational Space Controllers 45

5.2 TEST OF THE CONTROLLER PERFORMANCE IN CASE OF A CONSTRAINT IN HORIZONTAL AXIS 48

5.3 TEST OF THE MOTION/FORCE CONTROLLER PERFORMANCE IN OPERATIONAL SPACE 51

6. EXPERIMENTAL RESULTS 57

6.1 ASSUMPTIONS AND METHODS 58

6.2 SOFTWARE 60

6.3 RESULTS 60

7. CONCLUSION 67

REFERENCES 69

APPENDICES 73

Appendix A.1- Operation Modes for Matlab / V-REP Communication 73

Appendix A.2- Simulink Controllers 76

CURRICULUM VITAE 81

TABLES

Tablo 3.1: DH Table of PHANTOM Omni..17
Table 6.1: Joint limits for PHANTOM Omni...58
Table 6.2: End effector limits..58

YK

FIGURES

Figure 1.1 : Underactuated biped robot..2
Figure 2.1 : Constrained motion of a mechanism 9
Figure 2.2 : Force and acceleration relation with the input force 12
Figure 3.1 : PHANTOM Omni Robot..15
Figure 3.2 : Kinematic model of PHANTOM Omni 16
Figure 3.3 : Inertia Calculation 20
Figure 4.1 : Concept of joint space motion control..24
Figure 4.2 : Projected based joint space motion control 26
Figure 4.3 : Projected based joint space constraint force control 27
Figure 4.4 : Obtaining underactuated control input 30
Figure 4.5 : Concept of operational space motion control 32
Figure 4.6 : Projected based operational space motion control 35
Figure 4.7 : Operational space motion/force control 37
Figure 5.1 : Matlab - V-REP Communication 39
Figure 5.2 : Calculation of Projection Matrix in Case of a Constraint in Z-Axis 41
Figure 5.3 : VREP Simulator Environment 41
Figure 5.4 : Joint positions and velocities...42
Figure 5.5 : Joint input torques 42
Figure 5.6 : Actuator projection matrix for passive 3rd joint 43
Figure 5.7 : Joint positions and velocities while joint 3 is passive 43
Figure 5.8 : Joint input torques while joint 3 is passive...44
Figure 5.9 : Actuator projection matrix for passive 2nd joint...44
Figure 5.10 : Joint positions and velocities while joint 2 is passive 45
Figure 5.11 : End effector positions and velocity 46
Figure 5.12 : Joint input torques 46
Figure 5.13 : End effector positions and velocities while joint 3 is passive 47
Figure 5.14 : Control input and null space torque while joint 3 is passive 47
Figure 5.15 : Calculation of Projection Matrix in Case of a Constraint in X-Axis.........48
Figure 5.16 : End effector position and velocities in case of a constant input................49
Figure 5.17 : Joint input and null space torques...49

Figure 5.18 : Tracking performance in case of a time varying reference input 50
Figure 5.19 : Joint input and null space torques...51
Figure 5.20 : End effector force in z-direction...51
Figure 5.21 : End effector position and velocity..52
Figure 5.22 : End effector forces 52
Figure 5.23 : End effector position and velocity..53
Figure 5.24 : End effector forces 54
Figure 5.25 : End effector position and velocity when joint 2 is passive 55
Figure 5.26 : End effector forces when joint 2 is passive 56
Figure 6.1 : Bode plot of the second order filter..58
Figure 6.2 : Joint positions and velocities when joint 2 is passive 60
Figure 6.3: Joint torques when joint 2 is passive 61
Figure 6.4 : Joint torques when joint 3 is passive 62
Figure 6.5 : Joint torques when joint 3 is passive 62
Figure 6.6 : Joint positions and velocities in case of hybrid motion/force control 63
Figure 6.7 : Joint constraint torques and inputs in case of hybrid motion/force control.63
Figure 6.8 : Joint constraint torques and inputs in case of passive 2nd joint..................64
Figure 6.9 : Joint constraint torques and input in case of passive 3rd joint....................65
Figure 6.10 : End effector position and velocity in case of a constant reference 65
Figure 6.11 : End effector position and velocity in case of a time varying reference 66
Figure App. 1.1: Blocking function calls...74
Figure App. 1.2 : Non-blocking function calls 74
Figure App. 1.3 : Data streaming..75
Figure App. 1.4: Synchronous Operation 76
Figure App. 2 1 : Position Control Scheme 77
Figure App. 2 2 : Position Control Scheme for Underactuated Case..............................78
Figure App. 2 3 : Hybrid Control Scheme for Fully Actuated / Underactuated Cases...79
Figure App. 2 4 : Operational Space Controller 80

ABBREVIATIONS

API : Application Programming Interface
DH : Denavit-Hartenberg
DOF : Degree of Freedom
RRR : Revolute-Revolute-Revolute
SVD : Singular Value Decomposition
URDF : Universal Robot Description File
V-REP : Virtual Robot Experimentation Platform

SYMBOLS

Acting and passive joint torques : $\Gamma_{\parallel},\ \Gamma_{\perp}$

Angular velocities of joints : ω

Center of mass locations : $P_{c_1}, P_{c_2}, P_{c_3}$

Constraint equation : $\phi(q)$

Constraint forces at joints : Γ_c

Constraint forces at the end effector : F_c

Constraint inertia in joint space : M_p

Constraint inertia in operational spaces : Λ_c

Conversion operator from independent to generalized coordinates : ψ

Derivation of conversion operator : Λ

Derivation of projection matrix : D

End effector positions : x

Force control law in joint space : u_f

Force control law in operational space : ζ

Force exerted on link body center : F

Gains for motion control in joint space : G_p, G_D

Gains for motion control in operational space : K_p, K_D

Homogeneous transformation matrix : T

Independent coordinates : θ

Jacobian matrix : J

Joint input torque : Γ

Joint positions : q

Linear velocities of joints : v

Mass matrix : M

Moments of inertia of the links : $I_{xx_i}, I_{yy_i}, I_{zz_i}$

Motion control law in joint space : u_p

Number of degree of freedom : n

Projection matrix : P

Proportional gain for force control in operational space : K_f

Proportional gain for force control : G_f

Rotation matrix : R

The constraint Jacobian : Jc

The vector of centrifugal and Coriolis terms : b

The vector of gravitational terms : g

Torque exerted on link body center : N

Underactuation matrix : B

Vector of position : p

1. INTRODUCTION

Underactuated robots belong to a special class of mechanical systems which have fewer independent input commands than number of generalized coordinates. There are different kinds of mechanisms that exhibit underactuation like free-floating space satellites equipped with robots, underwater vehicles, redundant snake robots and legged humanoids with passive joints. Control of this class of robot manipulators differs from the conventional robot types since passive links cannot be independently controlled.

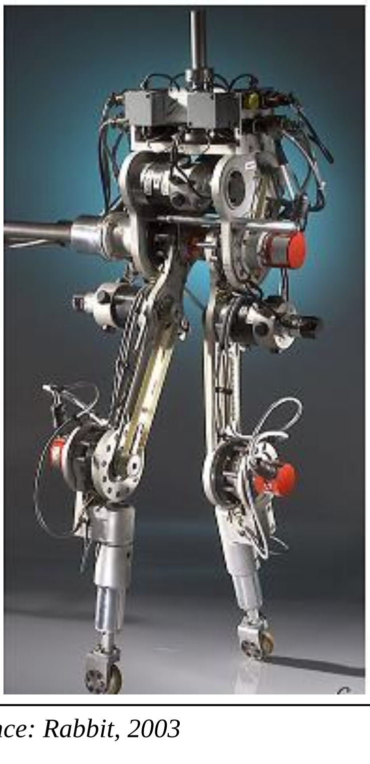
Serial chain underactuated manipulators may be in various types such as robots with flexible links, rigid robots with joint transmission elasticity or rigid robots with passive joints. In the first two cases, a manipulator with flexible links and a rigid robot with joint elasticity are inherently underactuated. While their joints are able to control directly by the actuators, the internal degrees of freedom causing the flexibility of the links or transmission elasticity cannot be controlled. Robots with passive joints connected by the rigid links are designed for reducing the energy consumption or avoiding them to become heavy and bulky. A conventional robot manipulator with failed joints is also a type of underactuated robot with passive joints.

Despite of its wide definition, various types of underactuated systems have different difficulties from the control perspective. Particularly, robots with passive joints give raise to control problems that the classical feedback control approaches turn out to be inadequate.

In addition to manipulation problem of underactuated mechanisms, it is becoming increasingly necessary for robots to interact safely with their environments. As robotic applications become gradually prevalent, robotic systems are increasingly employed in various industrial, medical, military and exploratory applications. In many applications, most of the robots operate in complex environments while interacting closely with environment and performing a variety of tasks. In order to make robot manipulators stable, safe, and robust to disturbances caused by the environment, and to improve their overall performance, it is required to regulate the contact forces in addition to motion control. To this end, joint torque control is used instead of motion control.

Underactuated robotic systems can be found in various types in real life applications. A legged robot is underactuated, since the toe behaves as a pin joint which the robot can rotate around. This additional degree of freedom is not governed directly by actuators. Most of the swimming and flying robots such as underwater vehicles, surface vessels and space craft are underactuated. The underactuation is originated from extra degrees of freedom coming with each control surface. Those vehicles cannot easily be used for interactive control by a manipulator due to the impossibility of handle the generalized forces exchanged with the manipulator’s base. In Figure 1.1, an example of underactuated systems is given.

Figure 1.1 : Underactuated biped robot



Reference: Rabbit, 2003

This work focuses on controlling the motion of an underactuated manipulator while also controlling the contact force between the environment and robot arm, using projected inverse based control schemes. The control method based on projected inverse dynamics directly uses the constraints to control the motion and minimize the instantaneous joint torques.

Projected inverse dynamic method that is used in this study proposes a unified formulation for dynamic equations of motion which gives a solution even in case of linearly dependent constraint equations. This fact is an advantage of the method in the presence of singular configurations. Another plus is the separation of joint input torque into its two orthogonal components which creates motion and constraint forces. Use of the control input in terms of motion and force allows developing two separate control laws for motion and force. The relation between motion and contact force also yields a formulation for obtaining the contact forces by using the information of input torques. This result allows utilizing of controllers without any force/torque sensor. So, the drawbacks such as noise, complexity and instability of sensor usage are eliminated in real applications.

1.1 PROBLEM DEFINITION

The control of underactuated manipulators is crucial for some applications. For example, where failed actuators cannot easily be repaired, it is important to control of the robot arm by the remaining actuators. In addition to manipulation problem of the robots with failed actuators, a robot manipulator can be designed as underactuated for special reasons such as reducing energy consumption or for lightweight construction.

Robot manipulators are described by their degrees of freedom (DOF) that refer to the number of single-axis joints in the arm that indicates capability of positioning. Manipulators require at least six DOF to manipulate an object in any position and orientation in Cartesian space, consisting of three in translation and three in orientation. This study focused on underactuated robot manipulators with passive joints which have less control inputs than their DOF. Although more actuators increase versatility, they also increase the size, weight, complexity, cost, and energy consumption. The tasks where it is required to eliminate these properties, underactuated manipulators may be utilized. Alternatively, for a successful manipulation of an underactuated robot which has at least one failed joint, the remaining actuators should control the robots with failed joints which work at place cannot easily be reached. Underactuated structure offers a number of benefits as being simple and high efficient than their fully actuated alternatives while the control of them are theoretically more complex.

Although many different solutions have been proposed, the control of underactuated robot manipulators is still an active and challenging problem due to completely free joints. The characterization of controllability properties is one of the most basic problems for controlling of underactuated robots. The underactuated robots moving in the horizontal plane, the effect of the gravity may guarantee the local controllability. But there may not controllability condition for underactuated robotic systems with drift terms.

In this thesis, controllers for constrained underactuated systems are investigated by first considering the projected inverse dynamics. Then underactuation is resolved through the addition of dynamically consistent control torques to the active joints. The term of the linear projection operator is used to develop a controller for constrained underactuated robotic manipulators.

Projection of the control input yields an explicit formulation that allows a better understanding of motion and constraint forces of the robot manipulator. In the absence of actuators for some of the joints, remaining actuators should compensate the required control torques in order to manipulate the robot performing a task. For this purpose, another linear transformation matrix is defined called as underactuation matrix. The vector space includes the torques all of the passive and active joints assuming they all produce forces, can be mapped onto space only actuated joints through this underactuation matrix. This mapping operation which is stored by underactuation matrix is used to develop the suitable controllers that can achieve the desired motion and force.

1.2 LITERATURE SURVEY

Classical robot control techniques often rely on the linearly independent spatial constraints and the invertible full rank Jacobian matrix. In the presence of singular configurations and redundancy, the Jacobian turns out to be a non-invertible matrix. The augmented Lagrangian method that solves the dynamic equations with an iterative process was proposed by Blajer (2002) which is able to overcome the singularities and redundancy. Generalized inverse and pseudoinverse of the Jacobian (Hollerbach 1987) have been also used to solve this problem. The projected inverse based control approach used in this study yields a solution that is applicable in case of singular constraints and redundancy.

The control of underactuated systems is an active research area in control of robotic systems. Most of the methods and solutions have been so far obtained by adapting the control approaches to the specific mechanism considered. Despite a number of different methods to control underactuated robotic systems, the number of general principles is not much sufficient. This section of thesis aims to review the literature which reports on the existing underactuated robot control algorithms.

The modeling and analysis of underactuated manipulators for control purposes have been presented in different forms by researchers. In the deficiency of control input in the dynamic equations of motion for such systems, input-output feedback linearization method for controlling becomes non-applicable. In order to overcome this problem, partial feedback linearization control has been widely used proposed by Spong (1994). Partial linearization method makes controlling of highly nonlinear dynamics of fully actuated robots easier with suitable linear control methods. This method can also be applied for underactuated robots provided that mass matrix satisfies the certain criteria.

There are different energy based control methods such as and Lagrangian and direct Lyapunov method. Bloch (2001) used controlled Lagrangian method which is obtained by modifying only the generalized inertia matrix and the potential energy function. Control process is realized by shaping the Lagrangian with suitable equilibrium states and structural features by using control inputs. Direct Lyapunov method used by White et. al. (2007) serves a nonlinear solution by Lyapunov functions consists of kinetic and potential portions.

There are typical differences between a practical control system and its theoretical model due to pre-modeled dynamics, uncertainties and disturbances. Robust control (Bergerman et. al. 1994) and adaptive control (Moore et. Al. 2014) are two main techniques for compensation of parameter uncertainties since they use online identification of either system parameter.

In this work, the control method is used based on projected direct and inverse dynamics proposed in study of Aghili (2005). This dynamic derivation with projected dynamics is applicable to achieve a joint space and an operational space controller. In joint space controller scheme, control input is projected into its two orthogonal components which lead the motion and reaction forces. This approach yields an effective formulation to manipulate an underactuated manipulator and minimize the actuation force. Operational space controller derived using this method decomposes rigid body dynamics into task space and null space dynamics in study of Mistry et. al. (2011). Underactuation is addressed by using null space forces indirectly apply torque at passive DOF such as at active joints performing a task.

2. DYNAMICS BASED ON ORTHOGONAL PROJECTIONS

This chapter of thesis will start from the explanation of inverse and direct dynamic formulation based on projection operator method in order to define the kinematics and dynamic of the system. Then the application of the method on PHANTOM Omni robot arm will be presented.

The development presented in this chapter is based on the work of Aghili (2005). Formulation for the direct and the inverse dynamics of robot manipulators based on the linear projection operator serves a basis for modeling, analysis and control of underactuated manipulators. In order to derive controllers for constrained underactuated manipulators, underactuation is resolved by the addition of suitable control inputs considering the projected inverse dynamic.

The linear projection matrix is an nxn square matrix, while n denotes the DOF of the robot. According to this method, first, constraint forces are annihilated from projection of the dynamic equations by the projection operator into the null space. Then, the direct dynamics is derived that relates the null space of constraint Jacobian to the acceleration by introducing a constraint inertia matrix that is always invertible. Subsequently, the constraint torques/forces are calculated through projecting the dynamics into the null space orthogonal. The approach is applicable also for systems with singular configuration or redundant constraints.

2.1 PROJECTION OPERATORS

In order to achieve dynamic model of mechanical systems, equations of motion need to be written in acceleration level and the applied torques need to be defined which are the projection of the forces along the axes of rotation. For the mechanisms constrained by their environment, orthogonal components of force and acceleration which are in direction of the constraint and motion, helps to describe overall dynamics of the system. Using the linear projection operator which includes the constraint condition, the orthogonal components of kinematic and dynamic parameters can be obtained.

Mathematically, a transformation $A : X \rightarrow Y$ from a vector space X to a vector space Y is a linear transformation by definition, if:

$$A(ax_1 + bx_2) = aA(x_1) + bA(x_2), \quad x_1, x_2 \in X \;\; a, b \in \mathbf{R} \tag{2.1}$$

The projection method of dynamic vectors into the null space and range space may be used in order to understand the behavior of constraint mechanisms. The null ($\mathbb{N}$) and range spaces ($\mathbb{R}$) are subspaces of a linear transformation. Null space of a linear transformation is defined as $x \in \mathbb{N}(A)$, if $Ax = 0$. Range space is defined as $y \in \mathbb{R}(A)$ if $y = A\,x$ for any x.

The linear projection operator can be calculated by the singular value decomposition method. Singular value decomposition (SVD) is defined as factorization of a real or complex matrix. Assuming r is the rank of A, Σ is a diagonal matrix that is equal to diagonal $(\sigma_1, .., \sigma_r)$, SVD of A matrix is given in the Equation (2.2)

$$A = U\,\Sigma\,V \tag{2.2}$$

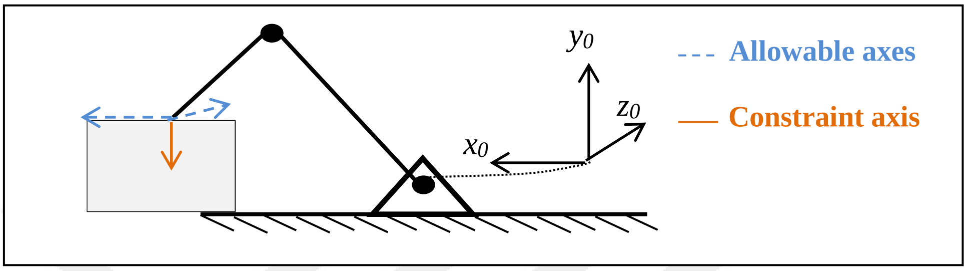
where $U = [U_1\,U_2]$ and $V = [V_1\,V_2]$ are unitary matrices. Each row of a unitary matrix has a norm of one and the Hermitian inner product between rows is zero. A vector has unique orthogonal decompositions, if the elements of vector have homogeneous measure units. The components of the decomposition of any x vector are obtained by using the projector operator as follows:

$$x_{\parallel} = P\,x \;\text{ and }\; x_{\perp} = (I - P)\,x \tag{2.3}$$

where $x_{\parallel}$ and $x_{\perp}$ are in subspaces of null space and the null space orthogonal of A. From the definition of projection operator, one can show that the kinematic and dynamic parameters of a mechanical system can be projected to find out the orthogonal components.

When a mechanism contacts with a rigid object, its motion in the particular direction is constrained and degree of freedom decreases. In order to obtain the projection operator for these mechanisms, constraint condition has to be expressed mathematically. This can be achieved by definition of the kinematic of constrained systems.

Figure 2.1 : Constrained motion of a mechanism



Constraint kinematics is described by a set of nonlinear equations given in Equation (2.4).

$$\phi(q) = \begin{bmatrix} \phi_1(q) \\ \vdots \\ \phi_k(q) \end{bmatrix} = 0 \tag{2.4}$$

where k denotes the number of constrained DOF and q denotes the vector of generalized coordinate represents the joint angles. Differentiation of Equation (2.4) results in a linear equation for the joint velocities

$$J_c\dot{q} = 0 \tag{2.5}$$

where J_c is the constraint Jacobian. The joint velocities are in the null space of the constraint Jacobian. This mathematical equation can be explained physically with the fact that there will be no velocity in the constraint axis. In order to define Equation (2.5) with notion of projector operator, it is possible to express the null-space orthogonal component of velocity as follows:

$$\dot{q}_\perp = (I - P)\dot{q} = 0 \tag{2.6}$$

In the light of these definitions, acceleration in restricted motion direction can be found by time differentiation of the velocity equation. Assuming $D = dP / dt$, acceleration is obtained as following:

$$\ddot{q}_{\perp} = (I - P)\ddot{q} = D\,\dot{q} \tag{2.7}$$

The null space orthogonal component of the acceleration is produced by the constraint. The projection operator is obtained by,

$$P = I - J_c^{+}\, J_c \tag{2.8}$$

where J_c^{+} denotes the pseudoinverse of J_c, D matrix which is the time differentiation of the projection matrix P can be found by differentiation of $J_c\dot{q} = 0$ and substituting Equation (2.7) into resulting equation. Then, D matrix is obtained as following:

$$D = -J_c^{+}\, \dot{J}_c \tag{2.9}$$

2.2 INVERSE AND DIRECT DYNAMICS

Assuming Γ is the joint input torques and Γ_c is the constraint force exerted by robot, dynamic equation of a rigid manipulator with n DOF are expressed as follows,

$$M(q)\ddot{q} + b(q, \dot{q}) + g(q) = \Gamma - \Gamma_c \tag{2.10}$$

where $M(q)$ is the $n \times n$ mass matrix of the manipulator, $b(q, \dot{q})$ is an $n \times 1$ vector of centrifugal and Coriolis terms and $g(q)$ is an $n \times 1$ vector of gravitational terms.

Constraint force Γ_c is exerted by joints in the direction of limitation of motion by the environment. Therefore, the orthogonal null space projection of the force at that point will be zero. As a consequence, the projected inverse dynamic equation can be easily obtained by multiplying the both side of the Equation (2.11) with projection operator.

$$\mathrm{PM}\ddot{\mathrm{q}} = \mathrm{P}(\Gamma - (\mathrm{b} + \mathrm{g})) \tag{2.11}$$

When PM term is singular, acceleration is not able to be calculated through matrix inversion in Equation (2.11). In order to solve this problem, the constraint inertia matrix formulation was developed. A solution can be obtained by multiplying Equation (2.7) with M and adding it to Equation (2.11) and resultant equation is found in the following form:

$$\mathrm{M_p}\ddot{\mathrm{q}} = \mathrm{P}(\Gamma - (\mathrm{b} + \mathrm{g})) + \mathrm{D_p}\dot{\mathrm{q}} \tag{2.12}$$

In the equation $\mathrm{Mp} = \mathrm{P\,M} - \gamma\,(\mathrm{I} - \mathrm{P})$ and $\mathrm{Dp} = \gamma \mathrm{D}$ where γ is an arbitrary scalar, hence, Mp and $\mathrm{D_p}$ are not unique. By adding the constraint inertia term, acceleration can be solved using Equation (2.12) which requires that the constraint inertia matrix be invertible. Since the projection operator in the Equation (2.10) eliminates all forces, it is seen that only the null space component of the generalized input force lead the motion.

Generalized input force can be decomposed into its orthogonal components by projection. The vector summation of these orthogonal components will give the input force, which called as acting ($\Gamma_{\parallel}$) and passive input forces ($\Gamma_{\perp}$). While passive forces produce the constraint forces, acting forces provides the motion of the mechanism. Finally, the dynamic equation of motion is expressed as shown in Equation (2.13).

$$\ddot{\mathrm{q}} = \mathrm{M_p}^{-1}(\Gamma_{\parallel} - (\mathrm{b} + \mathrm{g})_{\parallel} + \mathrm{D_p}\dot{\mathrm{q}}) \tag{2.13}$$

2.2.1 Constraint Force

In order to retrieve the constraint forces, dynamic equation for a rigid manipulator given in Equation (2.10) is projected into the null space orthogonal using $(\mathrm{I} - \mathrm{P})$. The resultant equation is in the following form:

$$\Gamma_c = (\mathrm{I} - \mathrm{P})(\Gamma - (b + g)) - (\mathrm{I} - \mathrm{P})\, M\ddot{q} \tag{2.14}$$

For ease of calculation, the acceleration term in Equation (2.14) can be written with velocity term gives a generalized equation for constraint forces.

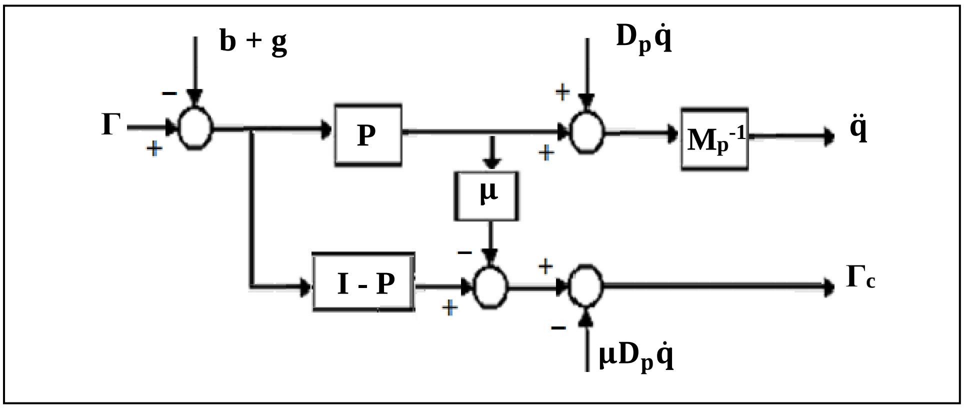
$$\Gamma_c = (\Gamma_{\perp} - (b + g)_{\perp}) - \mu(\Gamma_{\parallel} - (b + g)_{\parallel}) - \mu D_p \dot{q} \tag{2.15}$$

where $\mu = (\mathrm{I} - \mathrm{P}) M M_p^{-1}$.

2.2.2 Decomposition of Motion and Constraint Force

For constrained mechanical systems, acceleration and constraint force can be obtained by decomposing the total input. Inputs channels are decomposed to their acting $\Gamma_{\parallel}$ and passive $\Gamma_{\perp}$ components to obtain the acceleration and the constraint force as given in Figure 2.2.

Figure 2.2 : Force and acceleration relation with the input force



It can be directly seen from the figure that the acceleration is created by acting force which is the null space component of input. However, the constraint force output can be affected by directly from the acting force and indirectly from the passive force which are null space and null space orthogonal components of input channel.

If the inertia matrix satisfies the condition $\mu = 0$, the constraint force is no longer affected by the acting force. The equation of constraint force of such a system can be simplified to the following form given in Equation (2.16).

$$\Gamma_c = (\Gamma_\perp - (b + g)_\perp) - \mu D_p \dot{q} \tag{2.16}$$

In this condition, the constraint force is directly calculated by the passive input force.

3. UNDERACTUATED ROBOT MANIPULATORS

This section of this thesis is devoted to the derivation of the kinematic model and dynamic equations of motion of PHANTOM Omni (Geomagic 2016) robotic arm which is used in of the simulations and experimental studies. The first part provides a review of underactuation and utilization of Omni robot as an underactuated system. Then the kinematic and dynamic properties of system are explained.

3.1 INTRODUCTION

A manipulator is defined as a mechanical articulated arm constructed of links connected through joints that allow relative movements between two successive links. The mechanical structure of the manipulator is controlled to achieve desirable tasks. The information received through sensors about the environment or the robot itself is processed, for calculating the appropriate torques to the actuators. Controlling of these systems has mostly an existence of subjection to physical constraints and strong nonlinearities. With the inclusion of underactuated feature to manipulators, the control problems arise due to internal degrees of freedom caused by the passive joints. A mechanism has underactuation when the number of actuators is less than the number of generalized coordinates. An increasing interest has been focused on the applications of underactuated mechanical systems over the last decades.

Underactuation may arise from one of the following possible ways:

a. **Failure of actuators:** A system may become underactuated when one or more actuators fail to work properly. A robot manipulator which has at least one failed joint is an instance for this type.

b. **The dynamics of the system:** Some mechanical systems e.g. flexible link robots, space crafts and underwater vehicles, underactuated feature are an inherent property. The extra DOF causing from the dynamics of these systems may be overcome modeling them as underactuated to improve appropriate controllers.

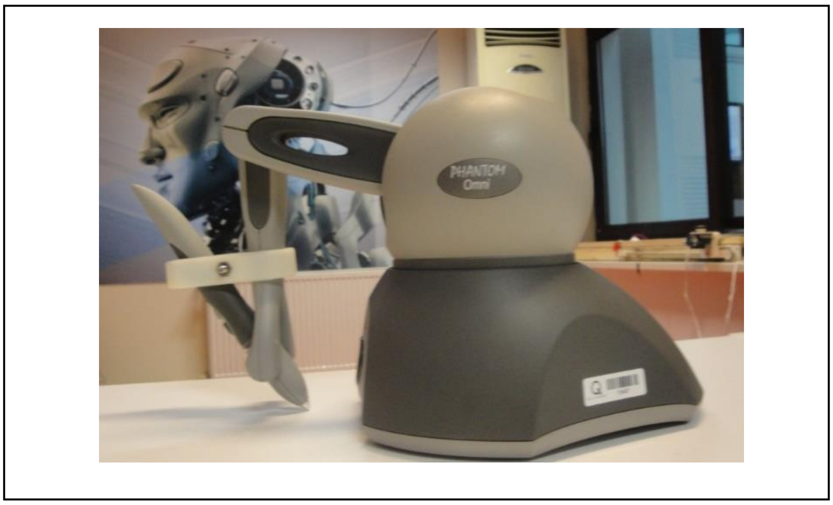
c. **Practical purposes:** A system may be designed specifically underactuated for practical purposes. Designing such a system may aim to achieve a mechanism whose energy consumption is smaller, yet whose dexterity is maintained.

In this study, underactuation is imposed artificially to gain an insight in the underactuated control theory based on projected inverse dynamics. PHANTOM Omni robot arm is the basis of control studies. Although all the joints are equipped with actuators, some of the joints were considered as passive and control inputs are only implemented to active ones, while the others do not have any input. Hybrid motion force control of an underactuated mechanism can be thus studied on this robot.

3.2 PHANTOM OMNI ROBOT MANIPULATOR

The PHANTOM Omni haptic device is a small robot arm that has 3 active revolute joints which are actuated by computer controlled DC electric motors. Besides the active joints, the PHANTOM robot has 3 passive wrist joints. In simulation and experimental studies the first three joints of the device, i.e., the actuated joints are used. Figure 3.1 depicts a PHANTOM Omni haptic device.

Figure 3.1 : PHANTOM Omni Robot

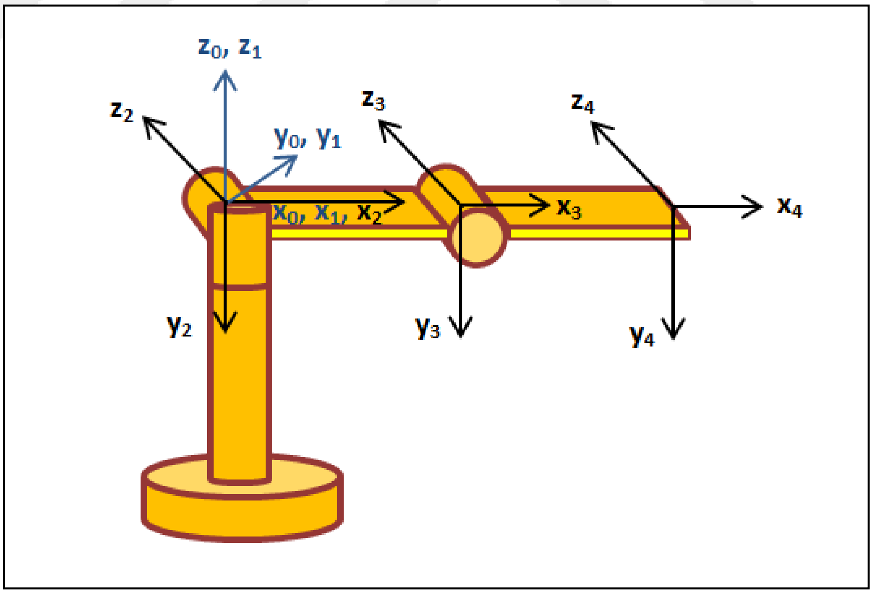


3.2.1 Kinematic Model

In this section of the thesis, the kinematic model of the PHANTOM Omni robot manipulator is explained. In order to obtain the required kinematic equations for developing the controllers based on projected inverse dynamics, it is a necessity to have the explicit expressions for the robot.

The structure of the PHANTOM Omni is represented as a 3-link Revolute-Revolute-Revolute (RRR) manipulator. While the kinematic analysis can be done through purely geometric structure, a more customized method will be used called as Denavit-Hartenberg method (Spong 2005) which includes four parameters (DH parameters) defining the orientation and position of the joints. The starting point is the frame attachment to the joints that is given in Figure 3.2

Figure 3.2 : Kinematic model of PHANTOM Omni



The coordinate frames are defined from the base which defined as frame 0 to the end effector frame defined as frame 4. With the base frame attached to the ground, the architecture can be easily described using the DH parameters shown in Table 3.1.

Table 3.1 : DH Table of PHANTOM Omni Robot

i	Θ_i	d_i	a_i	α_i
0-1	Θ_1	0	0	0
1-2	Θ_2	0	0	$-\pi/2$
2-3	Θ_3	0	l_1	0
3-4	0	0	l_2	0

Assume that $c_i = \cos(q_i)$, $s_i = \sin(q_i)$, $c_{ij} = \cos(q_i + q_j)$ and $s_{ij} = \sin(q_i + q_j)$, $(i = 1, 2, 3 \text{ and } j = 1, 2, 3)$. The transformation matrices corresponding to the each frame attached to the joints is given as the following:

$$ {}^0_1T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^1_2T = \begin{bmatrix} c_2 & -s_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^2_3T = \begin{bmatrix} c_3 & -s_3 & 0 & l_1 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^3_4T = \begin{bmatrix} 1 & 0 & 0 & l_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1} $$

Subsequently, the homogeneous transformation matrix of the end effector with respect to the based coordinate system calculated by,

$$ {}^0_4T = {}^0_1T \; {}^1_2T \; {}^2_3T \; {}^3_4T \tag{3.2} $$

is given by the Equation (3.3)

$$
{}^{0}_{4}\mathrm{T} = \begin{bmatrix} c_1\, c_{23} & -c_1 s_{23} & -s_1 & l_2 c_1\, c_{23} + l_1 c_1 c_2 \\ s_1 c_{23} & -s_1 s_{23} & c_1 & l_2 s_1\, c_{23} + l_1 s_1 c_2 \\ -s_{23} & -c_{23} & 0 & -l_1 s_2 - l_2 s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3}
$$

Differentiation with respect to joint variables of the forward kinematics position equations yields the below equation,

$$
\dot{x} = J(q)\dot{q} \tag{3.4}
$$

$\dot{x}$ is the velocity of end effector, $\dot{q}$ is the vector of joint velocities, $J(q)$ is a 6xn matrix whose elements are in general nonlinear functions is called as Jacobian matrix. End effector velocity vector $\dot{x}$ consist of the two components, linear and angular velocity expressing by Equation (3.5)

$$
\dot{x} = \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{3.5}
$$

Since the first three joint is considered for the simulation studies, Jacobian matrix has dimensions as 6×3. The last three rows that correspond to angular velocity of J are relevant for calculations and it can be directly calculated considering the position equations from the forward kinematics by the Equation (3.6).

$$
J = \begin{bmatrix} -s_1\,(l_1 c_2 + l_2 c_{23}) & -c_1\,(l_1 s_2 + l_2 s_{23}) & -l_2 c_1 s_{23} \\ -c_1\,(l_1 c_2 + l_2 c_{23}) & -s_1\,(l_1 s_2 + l_2 s_{23}) & -l_2 s_1 s_{23} \\ 0 & -(l_1 c_2 + l_2 c_{23}) & -l_2 c_{23} \end{bmatrix} \tag{3.6}
$$

3.2.2 Dynamic Analysis

The dynamic equations give the relationships between input forces and constraint forces acting on joints. In order to analyze and implement the controllers it is necessary to know the dynamic equation of the PHANTOM Omni. These equations governing the motion of this serial 3-RRR manipulator are calculated through the Newton-Euler equations.

Newton - Euler method that computes torques using the motion of the links and joints consists of two parts. First, link velocities and accelerations are computed iteratively from link 1 out to link n, then, forces are derived recursively from link n back to link 1. (Craig 2005)

In order to achieve the force and torque equations, following iterations is utilized.

Outward iterations: i: 0 →2

$$ {}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R\ {}^{i}\omega_{i} + \dot{q}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} \tag{3.7} $$

$$ {}^{i+1}\dot{\omega}_{i+1} = {}^{i+1}_{i}R\ {}^{i}\dot{\omega}_{i} + {}^{i+1}_{i}R\ {}^{i}\omega_{i} \times \dot{q}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} + \ddot{q}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} \tag{3.8} $$

$$ {}^{i+1}\dot{v}_{i+1} = {}^{i+1}_{i}R\,(\ {}^{i}\dot{\omega}_{i} \times\ {}^{i}p_{i+1} +\ {}^{i}\omega_{i} \times \left(\ {}^{i}\omega_{i} \times\ {}^{i}p_{i+1}\right) +\ {}^{i}\dot{v}_{i}) \tag{3.9} $$

$$ {}^{i+1}\dot{v}_{c_{i+1}} = {}^{i+1}\dot{\omega}_{i+1}\ x\ {}^{i+1}P_{c_{i+1}} +\ {}^{i+1}\omega_{i+1} \times \left({}^{i+1}\omega_{i+1} \times\ {}^{i+1}P_{c_{i+1}}\right) + {}^{i+1}\dot{v}_{i+1}) \tag{3.10} $$

$$ {}^{i+1}F_{i+1} = m_{i+1}\,{}^{i+1}\dot{v}_{c_{i+1}} \tag{3.11} $$

$$ {}^{i+1}N_{i+1} = {}^{c_{i+1}}I_{i+1}\,{}^{i+1}\dot{\omega}_{i+1} +\ {}^{i+1}\omega_{i+1} \times {}^{c_{i+1}}I_{i+1}\,{}^{i+1}\omega_{i+1} \tag{3.12} $$

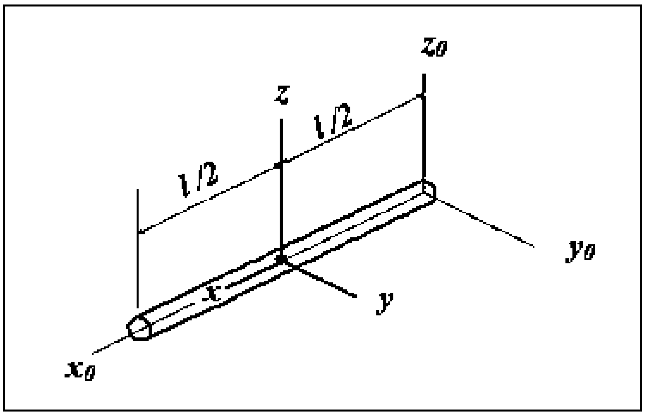
Inward iterations: i: 3 →1

$$ {}^{i}f_{i} = {}^{\ \ i}_{i+1}R^{i+1}\,{}^{i+1}f_{i+1} +\ {}^{i}F_{i} \tag{3.13} $$

$$ {}^{i}\Gamma_{i} =\ {}^{i}N_{i} + {}^{\ \ i}_{i+1}R^{i+1}\,{}^{i+1}\Gamma_{i+1} +\ {}^{i}P_{c_{i}} \times\ {}^{i}F_{i} +\ {}^{i}p_{i+1}\,x\ {}^{\ \ i}_{i+1}R^{i+1}\,{}^{i+1}f_{i+1} \tag{3.14} $$

In order to derive the dynamics of the Omni robot, first step is to define centers of gravity which are the distances from the origins of the link frames to the centers of the gravity of links. Subsequently, the inertial parameters must be defined. In this study, the inertias are calculated assuming the links are cylindrical with the diameter of the cylinder smaller than their lengths.

Figure 3.3 : Inertia Calculation



Moments of inertia of link 2 and link 3 are calculated by Equation (3.16) assuming they are slender rods with the diameter of the cylinder comparatively smaller than their length. I_{xx_i}(i=2,3) is neglected since the radius of the rod is much smaller than the length.

$$I_{c_i} = \begin{bmatrix} Ixx_i & 0 & 0 \\ 0 & Iyy_i & 0 \\ 0 & 0 & Izz_i \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & ml^2/3 & 0 \\ 0 & 0 & ml^2/3 \end{bmatrix} \quad (i=2,3) \qquad (3.15)$$

Mass of the link is denoted by m and length denoted by *l* inertias. Moment of inertia for the first link is calculated assuming that it is a solid sphere where the mass moment inertia matrix is $I_{c_1} = \mathrm{diag}\,(2mr^2/5,\ 2mr^2/5,\ 2mr^2/5)$. Center of mass locations are:

$$P_{c,1} = \{0,0,0\}, \quad P_{c,2} = \{l_1/2\,,0,0\}, \quad P_{c,3} = \{l_2/2\,,0,0\} \qquad (3.16)$$

Since the center of mass is too close to rotation axis for the first link, mass location of it is assumed as zero. With the knowledge of the kinematics and mass distribution of the manipulator, the joint forces required to lead motion is derived by using Newton -Euler outward and inward equations. Joint torque equations are given by:

$$\begin{aligned}\Gamma_1 = {} & \tfrac{1}{8}(\ddot{\theta}_1(8\mathrm{Sin}(\theta_2)^2\mathrm{Ixx}_2 + 8\mathrm{Sin}(\theta_2+\theta_3)^2\mathrm{Ixx}_3 + 4\mathrm{Iyy}_2 + 4\mathrm{Cos}(2\theta_2)\mathrm{Iyy}_2 + \\ & 4\mathrm{Iyy}_3 + 4\mathrm{Cos}(2(\theta_2+\theta_3))\mathrm{Iyy}_3 + 8\mathrm{Izz}_1 + l_1^2m_2 + \mathrm{Cos}(2\theta_2)l_1^2m_2 + 4l_1^2m_3 + \\ & 4\mathrm{Cos}(2\theta_2)l_1^2m_3 + 4\mathrm{Cos}(\theta_3)l_1l_2m_3 + 4\mathrm{Cos}(2\theta_2+\theta_3)l_1l_2m_3 + l_2^2m_3 + \mathrm{Cos}(2(\theta_2 + \\ & \theta_3))l_2^2m_3) + 2\dot{\theta}_1(\dot{\theta}_2(4\mathrm{Sin}(2\theta_2)\mathrm{Ixx}_2 + 4\mathrm{Sin}(2(\theta_2+\theta_3))\mathrm{Ixx}_3 - 4\mathrm{Sin}(2\theta_2)\mathrm{Iyy}_2 - \\ & 4\mathrm{Sin}(2(\theta_2+\theta_3))\mathrm{Iyy}_3 - \mathrm{Sin}(2\theta_2)l_1^2m_2 - 4\mathrm{Sin}(2\theta_2)l_1^2m_3 - 4\mathrm{Sin}(2\theta_2+\theta_3)l_1l_2m_3 - \\ & \mathrm{Sin}(2(\theta_2+\theta_3))l_2^2m_3) + 2\mathrm{Sin}(\theta_2+\theta_3)\dot{\theta}_3(4\mathrm{Cos}(\theta_2+\theta_3)\mathrm{Ixx}_3 - 4\mathrm{Cos}(\theta_2 + \\ & \theta_3)\mathrm{Iyy}_3 - l_2(2\mathrm{Cos}(\theta_2)l_1 + \mathrm{Cos}(\theta_2+\theta_3)l_2)m_3)))\end{aligned}\tag{3.17}$$

$$\begin{aligned}\Gamma_2 = {} & \tfrac{1}{8}(\dot{\theta}_1^2(-4\mathrm{Sin}(2\theta_2)\mathrm{Ixx}_2 - 4\mathrm{Sin}(2(\theta_2+\theta_3))\mathrm{Ixx}_3 + 4\mathrm{Sin}(2\theta_2)\mathrm{Iyy}_2 + \\ & 4\mathrm{Sin}(2(\theta_2+\theta_3))\mathrm{Iyy}_3 + \mathrm{Sin}(2\theta_2)l_1^2m_2 + 4\mathrm{Sin}(2\theta_2)l_1^2m_3 + 4\mathrm{Sin}(2\theta_2+\theta_3)l_1l_2m_3 + \\ & \mathrm{Sin}(2(\theta_2+\theta_3))l_2^2m_3) + 2(\ddot{\theta}_2(4\mathrm{Izz}_2 + 4\mathrm{Izz}_3 + l_1^2m_2 + 4l_1^2m_3 + 4\mathrm{Cos}(\theta_3)l_1l_2m_3 + \\ & l_2^2m_3) + \ddot{\theta}_3(4\mathrm{Izz}_3 + l_2(2\mathrm{Cos}(\theta_3)l_1 + l_2)m_3) - 2(g\mathrm{Cos}(\theta_2+\theta_3)l_2m_3 + \\ & l_1(g\mathrm{Cos}(\theta_2)m_2 + (2g\mathrm{Cos}(\theta_2) + 2\mathrm{Sin}(\theta_3)\dot{\theta}_2\dot{\theta}_3l_2 + \mathrm{Sin}(\theta_3)\dot{\theta}_3^2l_2)m_3))))\end{aligned}\tag{3.18}$$

$$\begin{aligned}\Gamma_3 = {} & \tfrac{1}{4}(4\ddot{\theta}_3\mathrm{Izz}_3 + \ddot{\theta}_3l_2^2m_3 - 2g\mathrm{Cos}(\theta_2+\theta_3)l_2m_3 + 2\mathrm{Sin}(\theta_3)d\theta_2^2l_1l_2m_3 + \\ & \ddot{\theta}_2(4\mathrm{Izz}_3 + l_2(2\mathrm{Cos}(\theta_3)l_1 + l_2)m_3) - \mathrm{Sin}(\theta_2+\theta_3)\dot{\theta}_1^2(4\mathrm{Cos}(\theta_2+\theta_3)\mathrm{Ixx}_3 - \\ & 4\mathrm{Cos}(\theta_2+\theta_3)\mathrm{Iyy}_3 - l_2(2\mathrm{Cos}(\theta_2)l_1 + \mathrm{Cos}(\theta_2+\theta_3)l_2)m_3)\end{aligned}\tag{3.19}$$

The dynamic equations for the manipulator can be written in the form:

$$\begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{bmatrix}\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \end{bmatrix} = \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \end{bmatrix}\tag{3.20}$$

Thus, each element of the mass (M) and centrifugal - coriolis (b) vector and vector of gravitational terms (g) can be written as following:

$$\begin{aligned}m_{11} = {} & \tfrac{1}{8}(8\mathrm{Sin}(\theta_2)^2\mathrm{Ixx}_2 + 8\mathrm{Sin}(\theta_2+\theta_3)^2\mathrm{Ixx}_3 + 4\mathrm{Iyy}_2 + 4\mathrm{Cos}(2\theta_2)\mathrm{Iyy}_2 + 4\mathrm{Iyy}_3 + \\ & 4\mathrm{Cos}(2(\theta_2+\theta_3))\mathrm{Iyy}_3 + 8\mathrm{Izz}_1 + l_1^2m_2 + \mathrm{Cos}(2\theta_2)l_1^2m_2 + 4l_1^2m_3 + \\ & 4\mathrm{Cos}(2\theta_2)l_1^2m_3 + 4\mathrm{Cos}(\theta_3)l_1l_2m_3 + 4\mathrm{Cos}(2\theta_2+\theta_3)l_1l_2m_3 + l_2^2m_3 + \mathrm{Cos}(2(\theta_2 + \\ & \theta_3))l_2^2m_3)\end{aligned}\tag{3.21}$$

$$m_{22} = \frac{1}{4}(4Izz_2 + 4Izz_3 + l_1^2 m_2 + 4l_1^2 m_3 + 4\mathrm{Cos}(\theta_3)l_1 l_2 m_3 + l_2^2 m_3) \tag{3.22}$$

$$m_{23} = \frac{1}{8}(4Izz_3 + l_2(2\mathrm{Cos}(\theta_3)l_1 + l_2)m_3) \tag{3.23}$$

$$m_{32} = \frac{1}{4}(4Izz_3 + l_2(2\mathrm{Cos}(\theta_3)l_1 + l_2)m_3) \tag{3.24}$$

$$m_{33} = \frac{1}{4}(4Izz_3 + l_2^2 m_3) \tag{3.25}$$

$$b_1 = \frac{1}{8}(2\dot{\theta}_1(\dot{\theta}_2(4\mathrm{Sin}(2\theta_2)Ixx_2 + 4\mathrm{Sin}(2(\theta_2 + \theta_3))Ixx_3 - 4\mathrm{Sin}(2\theta_2)Iyy_2 - 4\mathrm{Sin}(2(\theta_2 + \theta_3))Iyy_3 - \mathrm{Sin}(2\theta_2)l_1^2 m_2 - 4\mathrm{Sin}(2\theta_2)l_1^2 m_3 - 4\mathrm{Sin}(2\theta_2 + \theta_3)l_1 l_2 m_3 - \mathrm{Sin}(2(\theta_2 + \theta_3))l_2^2 m_3) + 2\mathrm{Sin}(\theta_2 + \theta_3)\dot{\theta}_3(4\mathrm{Cos}(\theta_2 + \theta_3)Ixx_3 - 4\mathrm{Cos}(\theta_2 + \theta_3)Iyy_3 - l_2(2\mathrm{Cos}(\theta_2)l_1 + \mathrm{Cos}(\theta_2 + \theta_3)l_2)m_3))) \tag{3.26}$$

$$b_2 = \frac{1}{8}(\dot{\theta}_1^2(-4\mathrm{Sin}(2\theta_2)Ixx_2 - 4\mathrm{Sin}(2(\theta_2 + \theta_3))Ixx_3 + 4\mathrm{Sin}(2\theta_2)Iyy_2 + 4\mathrm{Sin}(2(\theta_2 + \theta_3))Iyy_3 + \mathrm{Sin}(2\theta_2)l_1^2 m_2 + 4\mathrm{Sin}(2\theta_2)l_1^2 m_3 + 4\mathrm{Sin}(2\theta_2 + \theta_3)l_1 l_2 m_3 + \mathrm{Sin}(2(\theta_2 + \theta_3))l_2^2 m_3) + l_1(2\mathrm{Sin}(\theta_3)\dot{\theta}_2\dot{\theta}_3 l_2 + \mathrm{Sin}(\theta_3)\dot{\theta}_3^2 l_2)m_3) \tag{3.27}$$

$$b_3 = \frac{1}{4}(2\mathrm{Sin}(\theta_3)\dot{\theta}_2^2 l_1 l_2 m_3 - \mathrm{Sin}(\theta_2 + \theta_3)\dot{\theta}_1^2(4\mathrm{Cos}(\theta_2 + \theta_3)Ixx_3 - 4\mathrm{Cos}(\theta_2 + \theta_3)Iyy_3 - l_2(2\mathrm{Cos}(\theta_2)l_1 + \mathrm{Cos}(\theta_2 + \theta_3)l_2)m_3) \tag{3.28}$$

$$g_2 = -\frac{1}{4}(g\mathrm{Cos}(\theta_2 + \theta_3)l_2 m_3 + l_1(g\mathrm{Cos}(\theta_2)m_2 + 2g\mathrm{Cos}(\theta_2))m_3) \tag{3.29}$$

$$g_3 = -\frac{1}{2}(g\mathrm{Cos}(\theta_2 + \theta_3)l_2 m_3) \tag{3.30}$$

Using the dynamics of the PHANTOM Omni considering approximations of the inertial parameters of each links, a control scheme can be implemented to robot arm.

4. CONTROL OF UNDERACTUATED MANIPULATORS

In a motion control application, it is desired to move a robot manipulator to a goal position in order to perform a task. Robotic tasks are generally defined in the operational space in notion of the end effector, as control schemes are applied in the joint space to reach the desired targets. This fact leads to two different control problems called as joint space control and operational space control (Chung 2007).

For a successful manipulation task, it is required to overcome the interactions of robot and its environment. Only motion control becomes inadequate since the effect of the contact force, causing instability during the interaction. This case makes the force control necessary to perform a stable behavior of a robot manipulator in structured environments.

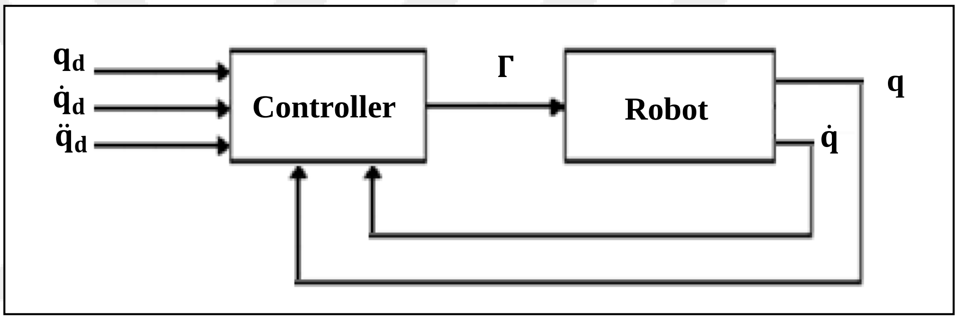
This section of thesis aims to explain projected inverse based control strategy to control of motion and regulate the interaction of the manipulator with environment. The method exhibits the opportunity of deriving and combining two different feedback control law through motion and force estimation.

Regarding to this purpose, the motion/force control of robotic manipulators will be explained based on the projected inverse dynamics. The relationship between control force and direct and inverse dynamics is used as the basis of position/force control structure. By considering the orthogonal components of the input force called as acting and passive forces which lead to the motion and constraint forces, control schemes for position and force are developed. A position control loop is constructed thanks to the inherent of the acceleration channel that is decoupled. In case of force control, the components of input force are both used to derive the constraint force as a feedback and by comparing the desired force; a control feedback loop is built.

4.1 JOINT SPACE CONTROL OF UNDERACTUATED MANIPULATORS

The purpose of the joint space motion control is to develop a controller that the joints follow the desired motion as closely as possible. By choosing the control forces as input, the control of robot manipulators is performed in the joint space. The desired motion may be expressed in terms of end effector coordinates and through the inverse kinematics and Jacobian of robot; they are converted to a joint trajectory. In this case, the controller calculates the joint force required to move the manipulator along the desired trajectory defined in joint coordinates.

Figure 4.1 : Concept of joint space motion control



In case of manipulators which constrained by environment, the number of DOF of the system decreases by presence of independent constraints. Therefore, dependent and independent coordinates have to be considered in solving of control problem. During interaction, the constraint physically limits on the trajectories that can be tracked by the end effector. In this case, the constraint forces have to be regulated.

Controlling the contact force requires force feedback similar to that of motion control. In case joint space control, it is hard to know all the constraint forces at the joints since all the joints have to be equipped with torque sensors. There are different kinds of estimation methods to get the joint torques without sensor. Projected base dynamics method which used in this study, serves an estimation method based on the input torques.

4.1.1 Motion Control

For an n DOF manipulator, assuming k is the number of reduced DOF due to the constraints the following equation gives the generalized coordinates,

$$q = \psi(\Theta) \tag{4.1}$$

where q represents all the joint positions and Θ is the independent ones. The function of independent coordinates must satisfy the constraint condition since,

$$\phi(q) = \phi\big(\psi(\Theta)\big) = 0 \tag{4.2}$$

Differentiation of this equation gives the joint velocities.

$$\dot{q} = \Lambda\dot{\Theta} \tag{4.3}$$

Differentiating Equation (4.3) ones more yields accelerations in terms of generalized coordinates.

$$\ddot{q} = \Lambda\ddot{\Theta} + \dot{\Lambda}\dot{\Theta} \tag{4.4}$$

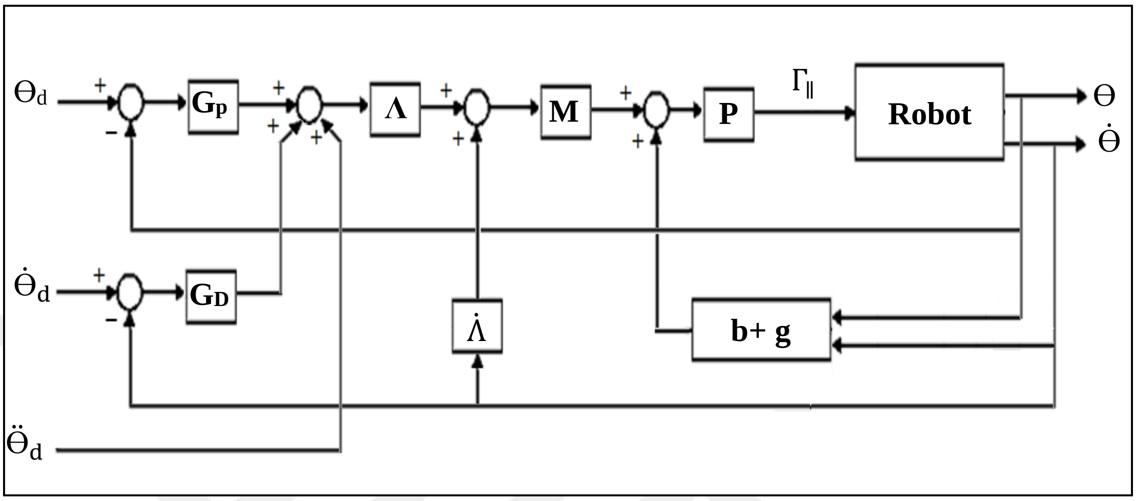
Using the acceleration defined in terms of the reduced dimensional coordinate, projected inverse dynamic equation can be calculated.

Implementing the acceleration into Equation (2.11) is given by:

$$PM(\Lambda\ddot{\Theta} + \dot{\Lambda}\dot{\Theta}) = P(\Gamma - (b + g)) \tag{4.5}$$

As it explained before, the main purpose of designing a controller in joint space is to derive the required joint forces to perform the desired joint motion. In this approach, decomposing the control input into its two orthogonal components yields an independent law to control of the motion. The block diagram scheme of the controller is constructed in Figure 4.2.

Figure 4.2 : Projected based joint space motion control



Projected inversed based motion control law is obtained by retrieving the acting joint torques from the inverse dynamic equation.

$$\Gamma_{\|} = (b+g)_{\|} + P\,M\,u_p \tag{4.6}$$

In the Equation (4.6), u_p is equal to the acceleration defined in terms of the reduced dimensional coordinate which is given in Equation (4.4). If acceleration is written with the desired motion variables, a Proportional Derivative (PD) control structure is obtained.

$$u_p = \dot{\Lambda}\dot{\Theta} + \Lambda(\ddot{\Theta}_d + G_D\dot{e}_p + G_p e_p) \tag{4.7}$$

where $e_p = \Theta_d - \Theta$ position error, $\dot{e}_p = \dot{\Theta}_d - \dot{\Theta}$ velocity error, G_p and G_D are the proportional and derivative gains respectively.

In the case underactuation, the actuated and non-actuated joints can be called as active joints and passive joints, respectively. For an n DOF manipulator, assuming there are a active joints and $n - a$ passive joints, the control force of the robot takes the form given by Equation (4.8)

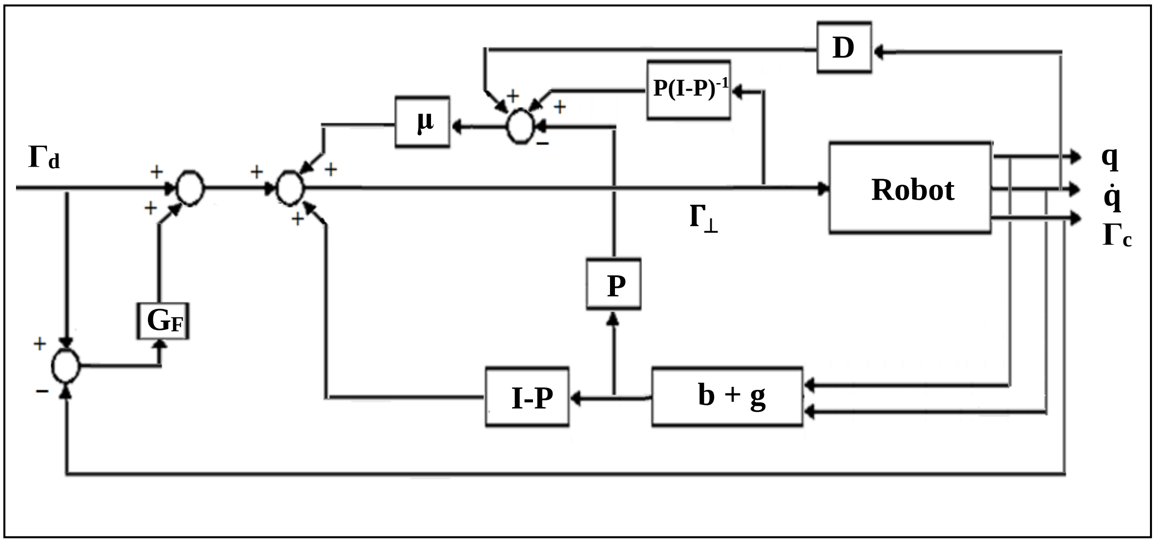
$$\Gamma = \begin{bmatrix} \Gamma_1 \\ \vdots \\ \Gamma_a \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{4.8}$$

The control forces, from 1 to a, denote the generalized input forces of the active joints, while the zero columns represent the passive joints. Choosing an another appropriate projection operator which projects all the joint inputs into the actuator space, the projected inverse dynamic control scheme can be implemented for underactuated systems. Further information about the implementation of the method is given in the Hybrid motion/force control section.

4.1.2 Force Control

During contact of a manipulator with a surface that limits the motion, for ensuring a compliant behavior control accuracy must be provided. The compliant behavior may be realized by controlling the resultant constraint forces in addition to the motion control.

Figure 4.3 : Projected based joint space constraint force control



The projected inverse dynamic formulation decomposes the input force into active and passive components that lead the motion and constraint forces, respectively. So in the cases such that the sign of constraint force does not change, motion control law works well. If not, the constraint condition may not be physically maintained using the motion control law. In this case, the constraint force must be controlled. The block diagram scheme of the controller is given in Figure 4.3.

The force control law can be derived using the null space orthogonal component of dynamic equation. Instead of the constraint force term in Equation (2.15), force control law can be implemented as u_F as follows:

$$\Gamma_{\perp} = (b+g)_{\perp} + \mu(\Gamma_{\parallel} - (b+g)_{\parallel} + D\dot{q}) + u_F \tag{4.9}$$

In above equation, u_F denotes the force control input. Then control law can be implemented through this term to find out the required passive force for a desired constraint force. For most of the cases, P control seems adequate and force control law is written as a P controller given by Equation (4.10)

$$u_F = \Gamma_d + G_f\, e_f \tag{4.10}$$

where $e_f = \Gamma_d - \Gamma_c$ force error and G_f is proportional gain.

4.1.3 Hybrid Control Method

A generic hybrid position/force control law usually requires a separation for force control from motion control. This case may alter for different constraint states and may have to be changed from one constraint configuration to another within the same constraint state. These control schemes yield two different feedback loops which use force error and position error.

A hybrid position/force control scheme may be derived through coupling the motion and the force control formulations based on projected inverse dynamic by following equation as:

$$\Gamma = \Gamma_{\parallel} + \Gamma_{\perp} \tag{4.11}$$

$$\Gamma = (b + g) + \mu D\dot{q} + (I + \mu)PMu_P + u_F \tag{4.12}$$

Projected inverse dynamics control approach leads to two error equations. The motion and force control laws can be applicable simultaneously due to the presence of two error information.

The reduced control input through underactuation was simply expressed by Equation (4.8). A proper projection matrix may be defined to make this transformation assuming I_a is an a × a identity matrix given by Equation (4.13)

$$B = \begin{bmatrix} I_a & 0 \\ 0 & 0 \end{bmatrix}, \quad B\,\Gamma = \Gamma \tag{4.13}$$

As suggested by Mistry et. al. (2011), if $J_cF \neq BJ_cF$ additional null space torques must be used to compensate for underactuated dynamics. The projected inverse control law can be simply transformed to a new form to satisfy this condition. If the null space motion is a subset of the range space of this transformation ($\mathbb{N}(A) \subseteq \mathbb{R}(B)$), the motion control equation may be used as the control input. Assuming the constraint force control law is an orthogonal projection of such a component,

$$\Gamma_{\perp} = (I - P)\Gamma_0, \quad \Gamma_0 \in \mathbb{R} \tag{4.14}$$

Then the control law takes the form given as following:

$$\Gamma = \Gamma_{\parallel} + (I - P)\Gamma_0 \tag{4.15}$$

Since B transforms the input force to the actuator space, I - B projects it to the null space which is zero.

$$(\mathrm{I}-\mathrm{B})\Gamma = 0 \tag{4.16}$$

The same equation can be written as the following form after substitution the control input in Equation (4.16)

$$(\mathrm{I}-\mathrm{B})(\mathrm{I}-\mathrm{P})\Gamma_0 = -(\mathrm{I}-\mathrm{B})\Gamma_{\parallel} \tag{4.17}$$

For linear systems with non-unique solutions, the pseudoinverse is used to get the solution of minimum Euclidean norm. In order to solve the Γ_0 which is given by Equation (4.17), pseudoinverse is used to get a minimal actuation force.

$$\Gamma_0 = -((\mathrm{I}-\mathrm{B})(\mathrm{I}-\mathrm{P}))^{+}(\mathrm{I}-\mathrm{B})\Gamma_{\parallel} \tag{4.18}$$

Substituting Γ_0 into the Equation (4.14) gives,

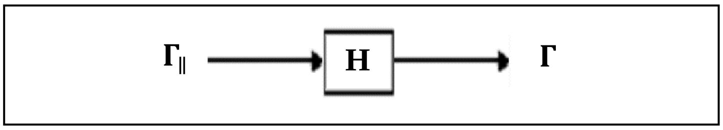
$$\Gamma_{\perp} = -(\mathrm{I}-\mathrm{P})((\mathrm{I}-\mathrm{B})(\mathrm{I}-\mathrm{P}))^{+}(\mathrm{I}-\mathrm{B})\Gamma_{\parallel} \tag{4.18}$$

Eventually, the control input for an underactuated system can be obtained by summation of active and passive control inputs as following:

$$\begin{aligned}\Gamma &= \Gamma_{\parallel} + \Gamma_{\perp}\\ &= (\mathrm{I}+\mathrm{H})\Gamma_{\parallel}\end{aligned} \tag{4.19}$$

where H = $-(\mathrm{I}-\mathrm{P})((\mathrm{I}-\mathrm{B})(\mathrm{I}-\mathrm{P}))^{+}(\mathrm{I}-\mathrm{B})$.

Figure 4.4 : Obtaining underactuated control input



The controllability condition of underactuated robot under the control law is related to existence of a solution.

In order to derive hybrid motion force/control for underactuated case, the force control law must be added to control equation given in Equation (4.19). When the force control law is written as Equation (4.18), the constraint force control law is not performed. If the both side of the Equation (4.19) is multiply with projection operator, the particular equation is taken the form,

$$\mathrm{P}\Gamma = \mathrm{P}(\Gamma_{\parallel} + \Gamma_{\perp}) = \Gamma_{\parallel} \tag{4.20}$$

Finally the hybrid motion/force control law is written as in Equation (4.21)

$$\Gamma = \Gamma_{\parallel} + \Gamma_{\perp} = (\mathrm{I} + \mathrm{H})\, \mathrm{P}\, \Gamma \tag{4.21}$$

4.2 OPERATIONAL SPACE CONTROL OF UNDERACTUATED MANIPULATORS

Since many of the tasks require controlling of the end effector motion, joint space control method may turn out inadequate. This fact motivates an approach that develops direct control method with respect to the end effector dynamics in the operational space.

Operational space dynamics present a solution for the modeling and control of robotic manipulators based on the dynamics of their end effectors. A unified approach for motion and force control is developed by Khatib (1987), based on the fundamentals of the operational space formulation. This method expresses decoupling task space and null space dynamics and constructs the control scheme with respect to the forces at the end effector or only motion within the null space.

The required variables for operational space formulation can be calculated by the Jacobian and forward kinematics using the joint space variables. The Jacobian matrix denoted by J, transforms the joint velocity to the end effector velocity is given in Equation (4.21).

$$\dot{x} = J(q)\dot{q} \tag{4.21}$$

The end effector acceleration is found by:

$$\ddot{x} = \dot{J}(q)\dot{q} + J(q)\ddot{q} \tag{4.22}$$

Figure 4.5 : Concept of operational space motion control

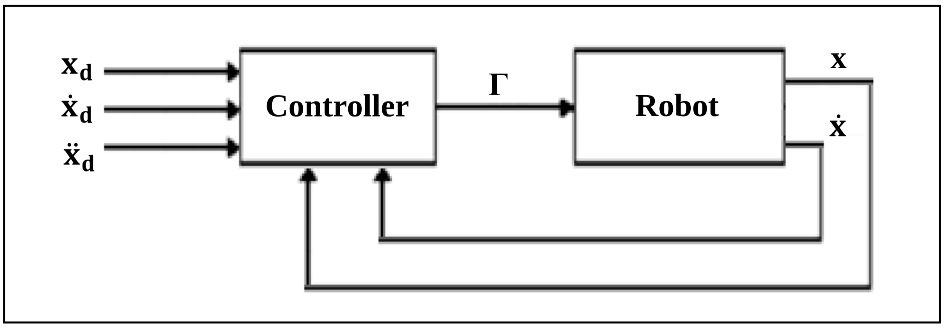


Figure 4.5 depicts the generic block diagram of operational space motion control scheme. For controlling constraint forces, a force feedback loop is also necessary. Operational space dynamics can be expressed with Equation (4.23)

$$\mathrm{M_t(q)\ddot{x} + b_t(q,\dot{q})\dot{q} + g_t(q) = F_t} \tag{4.23}$$

F_t is the control forces in the operational space. Operational space mass matrix is defined by,

$$\mathrm{M_t(q) = J^{-T}(q)M(q)J^{-1}(q)} \tag{4.24}$$

And

$$\mathrm{b_t(q,\dot{q}) = J^{-T}(q)B(q,\dot{q})J^{-1}(q) - M_t(q)\dot{J}(q)J^{-1}(q)} \tag{4.25}$$

$$\mathrm{g_t(q) = \ J^{-T}(q)G(q)} \tag{4.26}$$

End effector motion parameters, i.e position, velocity and acceleration can be derived through the joint space variables.

4.2.1 Motion Control in Operational Space

The main purpose of the operational space control is to obtain a controller which the end effector motion tracks the goal motion as closely as possible. This method offers many advantages since they use a control scheme that directly minimize the operational space errors.

Orthogonal projections method can be utilized for developing constrained operational space dynamics like joint space dynamics (Mistry et. al. 2011). Operational space dynamics is formulated with consideration of motion dynamic equation, when equations of projected inverse dynamics are used. It leads a compact solution of passive DOFs and underactuated manipulators.

The constraint forces caused by the contact are utilized to compensate the torque at non-actuated joint, dynamically convenient null space forces is used to create torque at passive joints to achieve desired operational space dynamics. The forces in null space may be used to produce constraint forces or null-space motion. The identical control method is also valid for unconstrained systems, using null space motion to satisfy for passive joints. This unified formulation is applicable for all the systems even in case of underactuation and unconstrained motion.

Projected inverse dynamic equation is given in the Equation (4.27):

$$\mathrm{M_p\ddot{q} + P(b + g) - D_p\dot{q} = P\Gamma} \tag{4.27}$$

Multiplying two side of the equation with $\mathrm{JM_p}^{-1}$

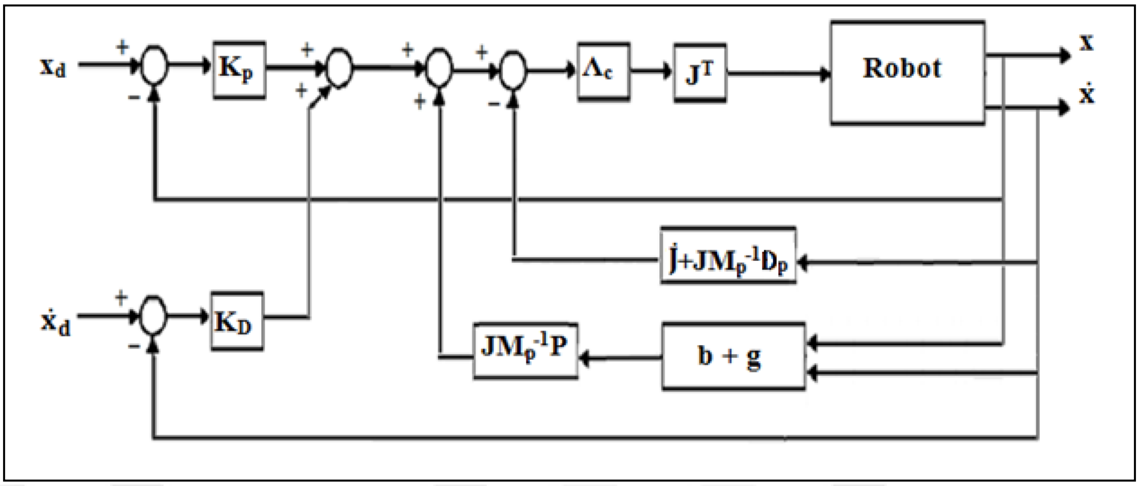
$$\mathrm{\ddot{x} - \dot{J}\dot{q} + JM_p}^{-1}\mathrm{(P(b + g) - D_p\dot{q}) = JM_p}^{-1}\mathrm{P\Gamma} \tag{4.28}$$

where $\mathrm{J\ddot{q} = \ddot{x} - \dot{J}\dot{q}}$. The force of the end effector is transformed by $\mathrm{J^T}$ into joint torques.

$$\mathrm{\Lambda_c\ddot{x} + \Lambda_c(JM_p}^{-1}\mathrm{P(b + g) - \left(\dot{J} + JM_p}^{-1}\mathrm{D_p\right)\dot{q}) = F} \tag{4.29}$$

where $\mathrm{\Lambda_c = J^{-T}P^{-1}M_pJ^{-1}}$.

Figure 4.6 : Projected based operational space motion control



The block diagram scheme in Figure 4.6 shows the controller structure in operational space. Operational space motion control law is developed according to the projected inverse equation by the Equation (4.30) with a PD controller.

$$\Gamma = J^T F = J^T(\Lambda_c \ddot{x}_{des} + \Lambda_c(JM_p{}^{-1}P(b+g) - \left(\dot{J} - JM_p{}^{-1}D_p\right)\dot{q}) \tag{4.30}$$

and,

$$\ddot{x}_{des} = K_p(x_d - x) + K_d(\dot{x}_d - \dot{x}) \tag{4.31}$$

Defining $N = I - J^T J^{T\#}$ and $J^{T\#} = (JM_p{}^{-1}\, PJ^T)^{-1}\, JM_p{}^{-1}\, P$, input torques are expressed by the Equation (4.32),

$$\Gamma = J^T F + N\Gamma_0 \tag{4.32}$$

where Γ_0 denotes the null space torque of a robot manipulator. B matrix, which was defined in Equation (4.13) , should satisfy the following condition:

$$J^T F + N\Gamma_0 = BJ^T F + BN\Gamma_0 \tag{4.33}$$

Retrieving Γ_0 from the below equation to solve the joint torque equation gives,

$$\Gamma_0 = ((I - B)N)^{+}(I - B)J^{T}F \tag{4.34}$$

The solution by pseudoinversion yields the minimum norm of Γ_0. This approach presents a valid solution both constrained and unconstrained systems.

Finally, substituting Γ_0 into the Equation (4.32), the control equation is taken the following form:

$$\Gamma = (I - N[(I - B)N]^{+})J^{T}F \tag{4.35}$$

4.2.2 Operational Space Hybrid Motion/Force Control Based on Projected Inverse Dynamics

In this part, a hybrid motion/force control scheme in operational space has been proposed. The usage of linear projection matrix gives an opportunity to develop two different feedback control loops in operational space and implement them simultaneously to the robot.

According to the error signal regarding to the constraint and desired forces, a force control scheme is built based on the projected inverse dynamics methodology. Using the transpose of Jacobian, desired forces in operational space are mapped into the joint space. By using the null space orthogonal component of input torques in Equation (2.10), constraint force term is derived and it is replaced by the control law defined as ζ. In this way; the force control law is written as in Equation (4.36).

$$(I - P)\Gamma = \Gamma_{\perp} = (I - P)(b + g) + \mu(P\Gamma - P(b + g) + D\dot{q}) + J^{T}\zeta \tag{4.36}$$

where ζ is the operational space force control law described by the following equation:

$$\zeta = F_d + K_f\,(F_d - F_c) \tag{4.37}$$

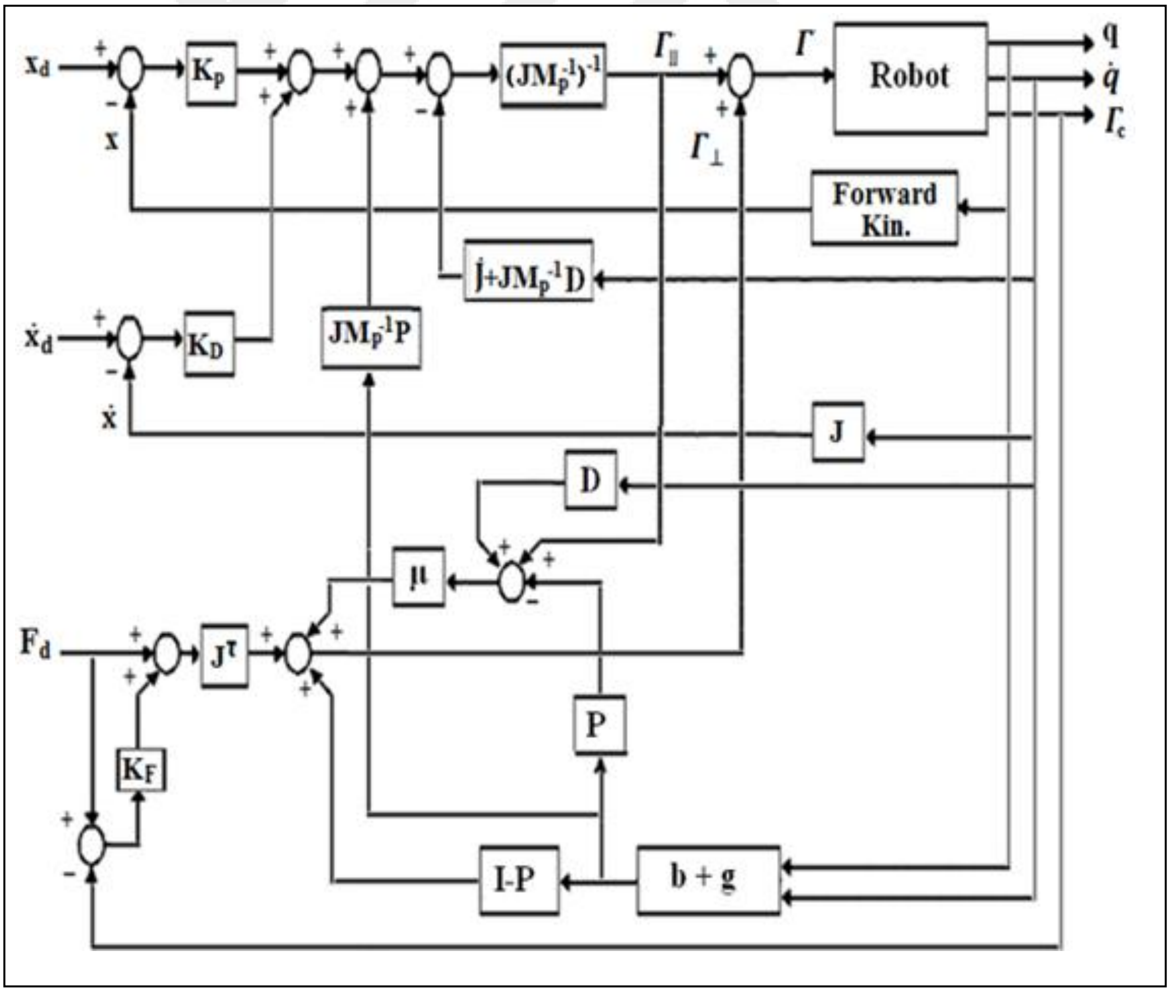
Above equation, F_d is the desired force, F_c is the constraint force at the end effector and K_f is the proportional gain.

Since the projection operator decomposes the joint torques into its two orthogonal components, the forces which create motion and force can be combined to get a hybrid control method. Hence, the motion control law must be written as a component which creates the motion. Similar to the derivation of Equation (4.36), motion control law in operational space is in the following form,

$$P\Gamma = \Gamma_{\parallel} = (JM_p{}^{-1})^{-1}(\ddot{x}_{des} + JM_p{}^{-1}P(b+g) - (\dot{J} + JM_p{}^{-1}D)\dot{q}) \qquad (4.38)$$

The block diagram scheme in Figure 4.7 shows the controller structure.

Figure 4.7 : Operational space motion/force control



Combining Equation (4.36) and (4.37), the motion / force control law in operational space is written in the following form:

$$\Gamma = \Gamma_{\parallel} + \Gamma_{\perp} = P\Gamma + (I - P)\Gamma$$

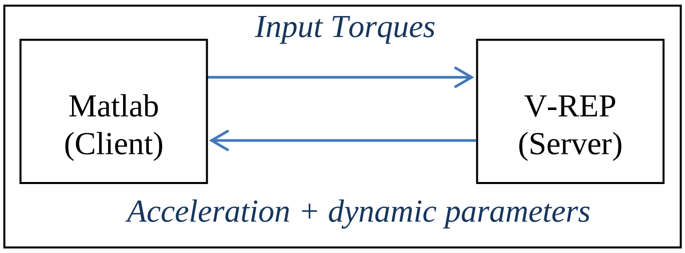
$$= (JM_p{}^{-1})^{-1}\left(\ddot{x}_{des} + JM_p{}^{-1}P(b + g) - \left(\dot{J} + JM_p{}^{-1}D\right)\dot{q}\right) + (I - P)(b + g) + \mu(P\Gamma - P(b + g) + D\dot{q}) + J^T\zeta \qquad (4.39)$$

5. SIMULATION AND RESULTS

Simulation of dynamics and control design of the robotic manipulators are generally complicated tasks. Our problem is to obtain a model of system dynamics fast and reliable enough to be suitable for designing of control mechanism. This may be realized using Matlab (Mathworks 2016) and Virtual Robot Experimentation Platform (V-REP) (Coppelia Robotics 2016) through dynamic modeling and rigid model of the robot manipulator.

V-REP is a robot simulator which is used for a wide range of applications such as rapid modeling and simulation and remote monitoring. This simulator allows developing the scripts with an external language and communicates it through a remote application programming interface (remote API). Matlab runs as a client which can read data from V-REP or send data to it. In classical client - server communication, a request is send by client and it waits till the server processed and replied that takes long time. In order to overcome this problem, the remote API allows choosing the type of operation mode and the way simulation advances by providing different modes to control the simulation. The detailed information is given in Appendix -1.

Figure 5.1 : Matlab - V-REP Communication



The projected inverse based control schemes have been tested in a number of simulations on PHANTOM Omni robot. Controller gains used in simulation studies are obtained by trial and error.

Omni robot manipulator model is imported as a Universal Robot Description Format (URDF) into the simulator. (Ruiz 2015) Since pure shapes are much more stable during dynamic simulation, instead of using the complicated robot model as respondable shape, convex decomposition of the links are used to increase the simulation performance.

5.1 TEST OF THE CONTROLLER PERFORMANCE IN CASE OF A CONSTRAINT IN VERTICAL AXIS

5.1.1 Joint Space Controllers

It is possible to limit the translational motion of the end effector of PHANTOM Omni in Cartesian coordinates. In order to develop a projected inverse based controller, projection matrix is calculated by corresponding to these kinematic constraints. The vertical translational motion of end effector is limited by the constraint equation which was calculated with forward kinematics:

$$z(t) = \phi(t) = -l_1 s_2 - l_2 s_{23} \tag{5.1}$$

Constraint Jacobian (J_c) is calculated by the differentiation of constraint equation with respect to the joint variables. Differentiating Equation (5.1) with respect to time yields the following constraint Jacobian,

$$J_c = [0 \quad (-l_1 c_2 - l_2 c_{23}) \quad -l_2 c_{23}] \tag{5.2}$$

which can be seen as the last row of the Jacobian given in Equation (3.6). The Matlab code for calculating the constraint Jacobian and the projection matrix is given in Figure 5.2. In this code, q(4), q(5), and q(6) corresponds the joint velocity that comes from the time derivation of the joint positions. Then, the projection matrix is obtained by pseudoinverse (pinv) function based on SVD.

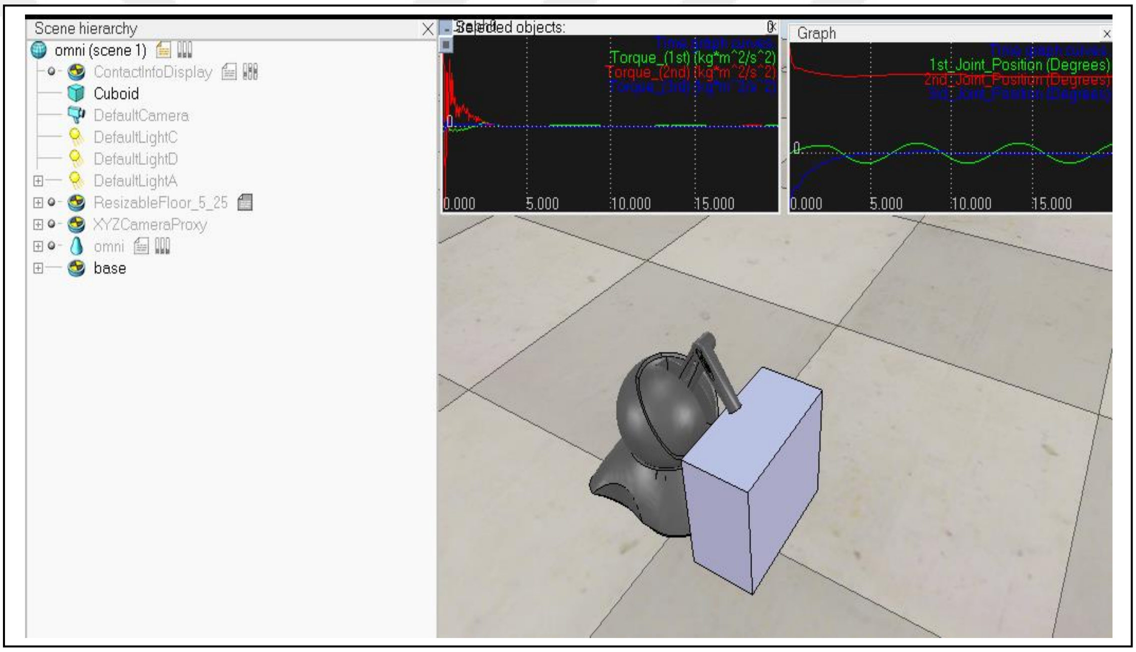
Figure 5.2 : Calculation of Projection Matrix in Case of a Constraint in Z-Axis

```
% Projection matrix in case of a constraint in z direction
Jc = (0, (-l1*c2-l2*c23), -l2*c23);
drv_Jc = (0 (l1*s2*q(5)+l2*s23*(q(5)+q(6))) (l2*s23)*(q(5)+q(6)));

P = eye(3) - pinv(Jc) * Jc;
D = -pinv(Jc) * drv_Jc;
```

In order to test the performance of the controller considering the manipulator as fully actuated, Equation (4.6) is implemented. Figure 5.3 shows the simulation environment.

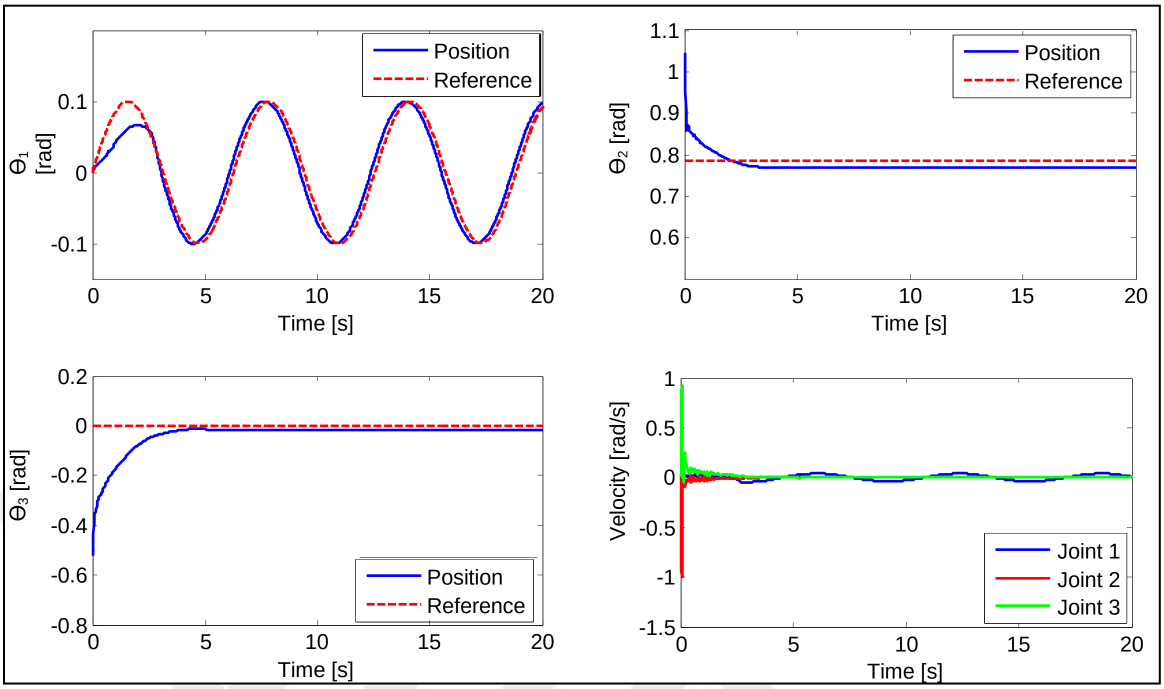
Figure 5.3 : VREP Simulator Environment



Initial joint positions are 0, $\pi/3$ and $-\pi/6$ while the desired joint trajectories are $0.1\sin(t), \pi/4$ and 0 rad for the joint 1, joint 2 and joint 3, respectively. Such a trajectory gives an insight to controller performance for constant and time varying control inputs. PD motion controller gains (see Equation 4.7) are chosen as $G_P = 15$, $G_D = 10$.

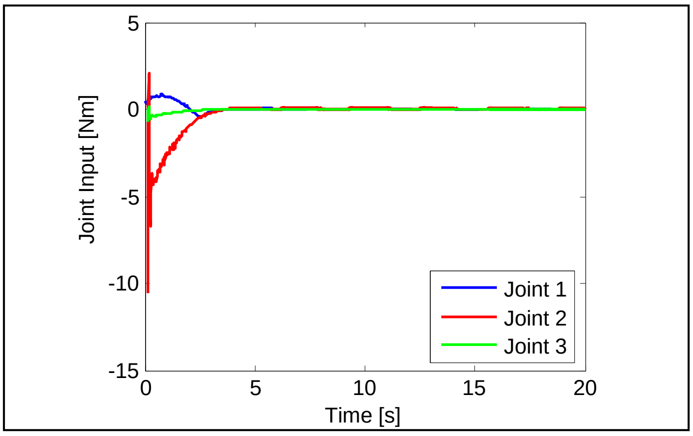
The motion of the fully actuated robot in joint space is presented in Figure 5.4. As can be seen from these plots, the joints are tracking desired trajectories.

Figure 5.4 : Joint positions and velocities



The control input and constraint forces are found as shown in Figure 5.5.

Figure 5.5 : Joint input torques



In the second step, the underactuated motion control is tested. The actuator projection matrix is chosen as follows:

Figure 5.6 : Actuator projection matrix for passive 3rd joint

```
% Actuator projection for underactuated control
B =[1 0 0;0 1 0;0 0 0];
```

This means that joint 3 is a passive joint which has no control input, while others are the active joints. Using the actuator projection matrix information, controller can generate dynamically consistent motion for compensation of lost torque at the passive joint according to the control law given in Equation (4.19).

PD motion controller gains are both 10 for the underactuated joint space motion control problem. Joint positions and velocities are given in Figure 5.7. Input forces are given in Figure 5.6. Although there is no applied torque at the joint 3, constraint forces create the motion to reach the goal position.

Figure 5.7 : Joint positions and velocities while joint 3 is passive

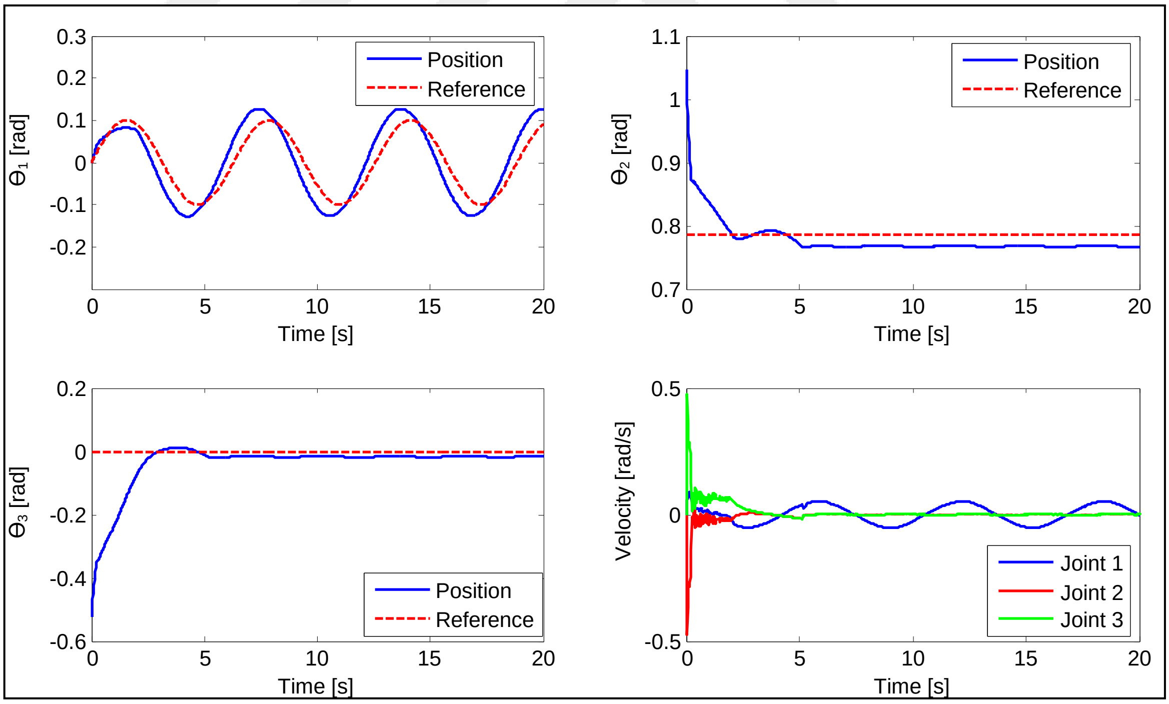
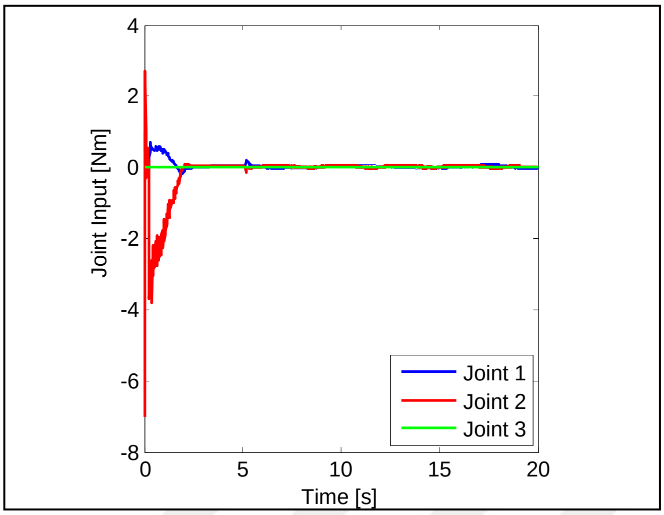


Figure 5.8 : Joint input torques while joint 3 is passive



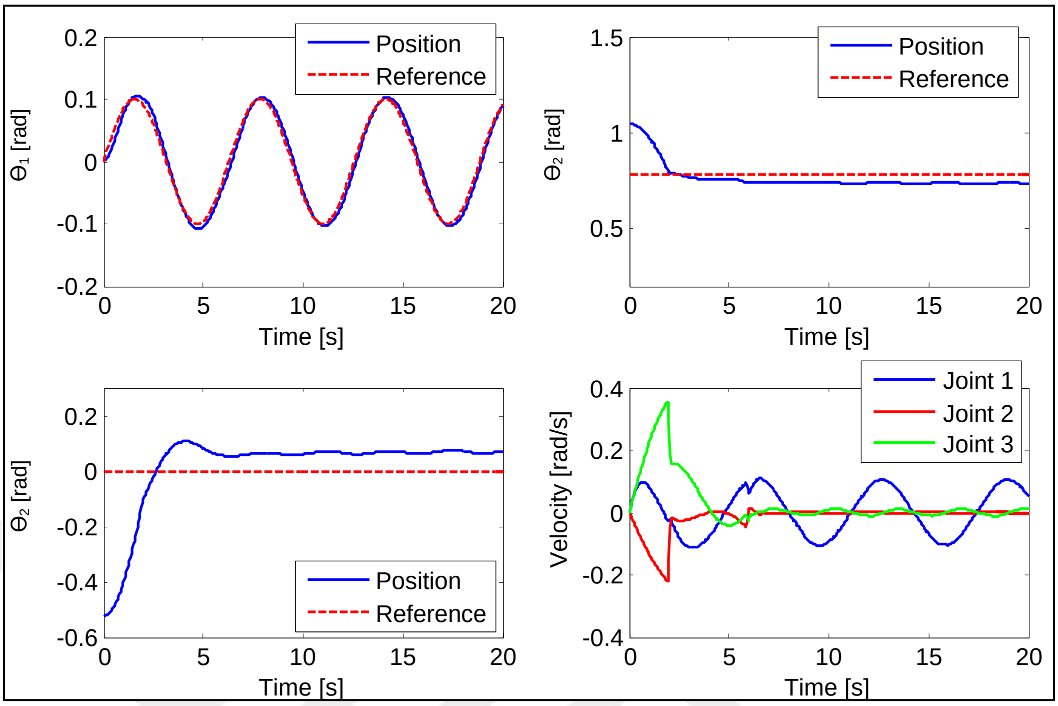
In the third step, underactuated motion control of joint 2 is tested. In order to select 2nd joint as passive, the actuator projection matrix is chosen as following.

Figure 5.9 : Actuator projection matrix for passive 2nd joint

```
% Actuator projection for underactuated control
B =(1  0  0;0  0  0;0  0  1);
```

Using the actuator projection matrix, controller has generated required input commands to compensate lost torque at the joint 2. The joint motions are given in Figure 5.10 while proportional and derivative gains are 10 and 5, respectively.

Figure 5.10 : Joint positions and velocities while joint 2 is passive



5.1.2 Operational Space Controllers

For defining the desired motion in terms of operational space, the motion control law in Equation (4.30) is implemented. Operational space motion control performs considering the initial joint angles as 0, $\pi/3$ and $-2\pi/3$ which makes the end effector initial positions 0.132, 0, 0 m in x, y and z directions, respectively. Set point motion control performance for constant references $x = x(0) + 0.02$ m, where $x(0)$ is the initial position in x (0.132m) direction, $y = 0.1$ m, is given in Figure 5.11. PD motion controller gains are 0.5 and 0.3 respectively.

Figure 5.11 : End effector positions and velocity

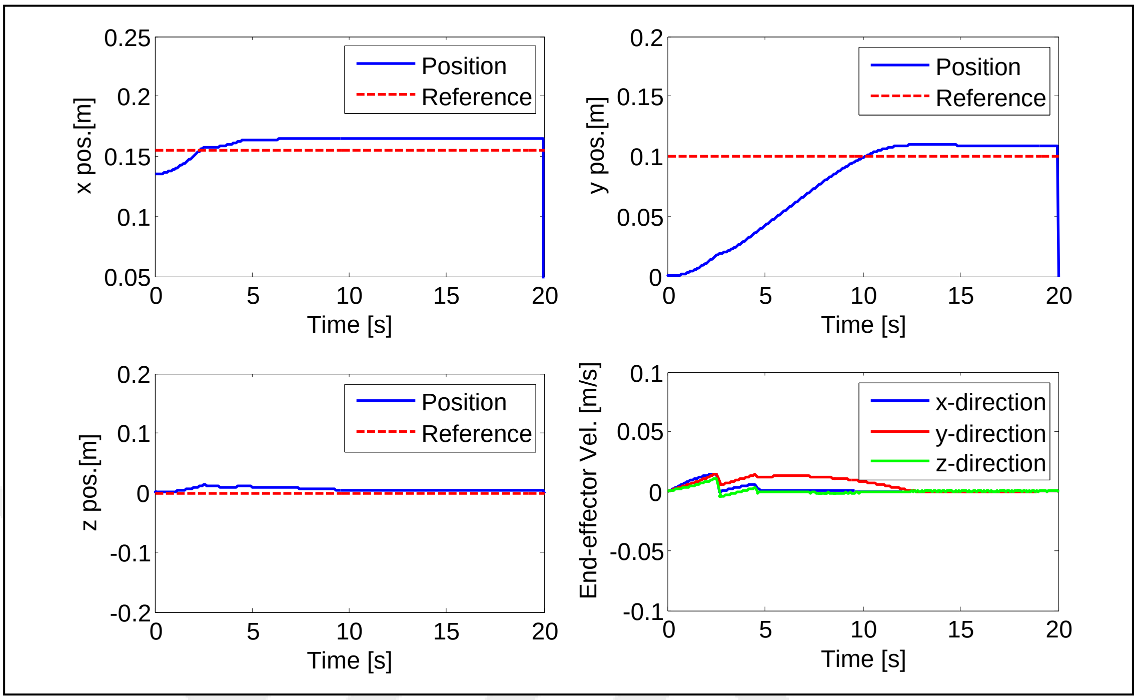
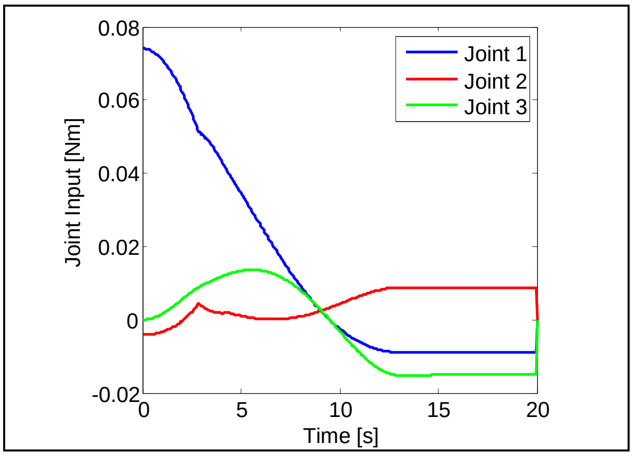


Figure 5.12 shows the input forces while no force feedback control is applied.

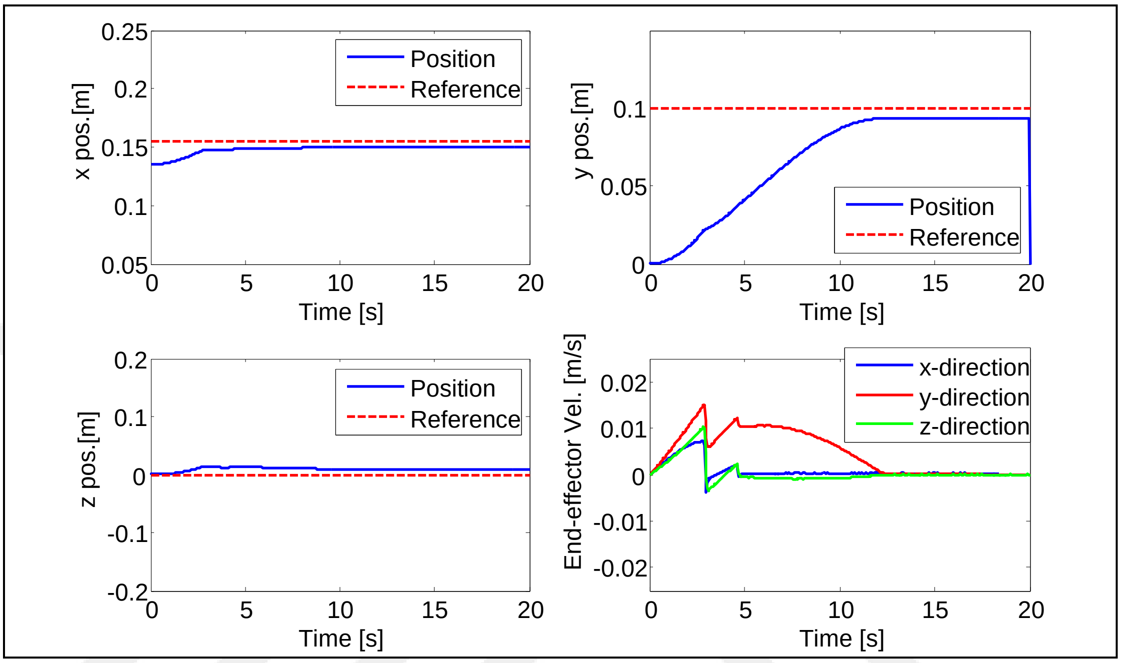
Figure 5.12 : Joint input torques



In case of underactuation, the actuator projection matrix is chosen to set the joint 3 as passive. For this case control input of joint 3 is zero. Initial and desired positions are similar with the previous case, references are $x = x(0) + 0.02$ m and $y = 0.1$ m.

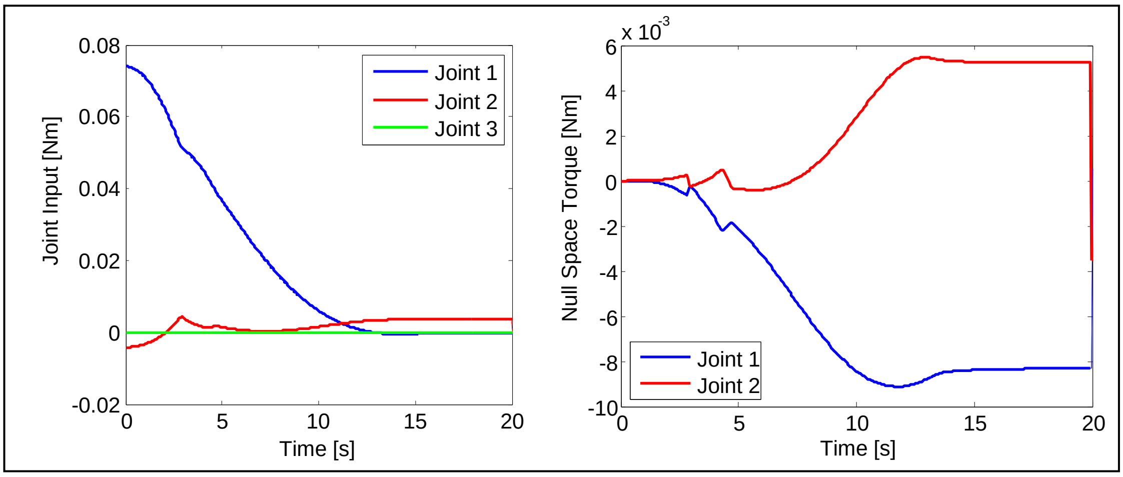
Controller gains are $K_p = 0.1$ and $K_d = 0.5$, motion control performance is given in Figure 5.13.

Figure 5.13 : End effector positions and velocities while joint 3 is passive



Operational space underactuated controllers use null space torque in order to generate the motion at passive joint. Null space torque at each joint given in the Equation (4.32) creates a motion to compensate the lost torque at passive joint.

Figure 5.14 : Control input and null space torque while joint 3 is passive



5.2 TEST OF THE CONTROLLER PERFORMANCE IN CASE OF A CONSTRAINT IN HORIZONTAL AXIS

Only the underactuated case is tested for horizontal constraints. In the horizontal direction the constraint equation is described by the Equation (5.2). In order to derive constraint Jacobian and projection matrix for this case, the required Matlab code is given in Figure 5.15.

$$x(t) = \phi(t) = l_2 c_1 c_{23} + l_1 c_1 c_2 \tag{5.2}$$

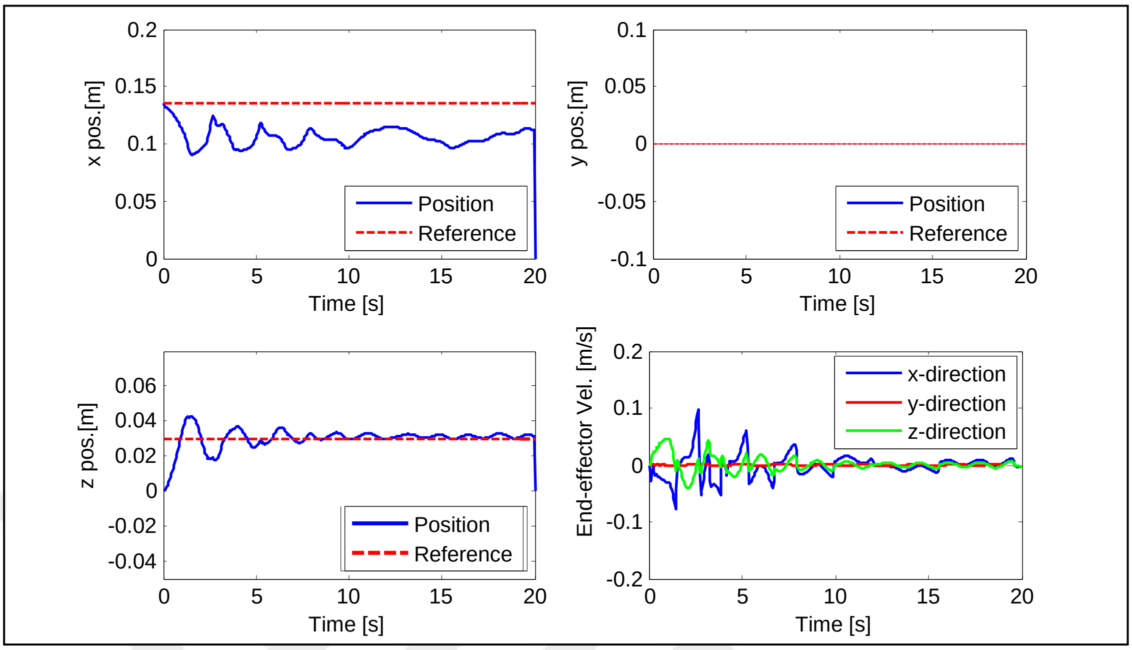
Figure 5.15 : Calculation of Projection Matrix in Case of a Constraint in X-Axis

```
% Projection matrix in case of a constraint in x direction
Jc = (-s1*(l2*c23+l1*c2),-c1*(l2*s23+l1*s2),-l2*c1*s23);
drv_Jc = (-c1*q(4)*(l2*c23+l1*c2)+
s1*(l2*s23*(q(5)+q(6))+l1*s2*q(5)), s1*q(4)*(l2*s23+l1*s2)-
c1*(l2*c23*(q(5)+q(6))+l1*c2*q(5)), l2*s1*q(4)*s23-
l2*c1*c23*(q(5)+q(6)));

P = eye(3) - pinv(Jc) * Jc;
D = -pinv(Jc) * drv_Jc;
```

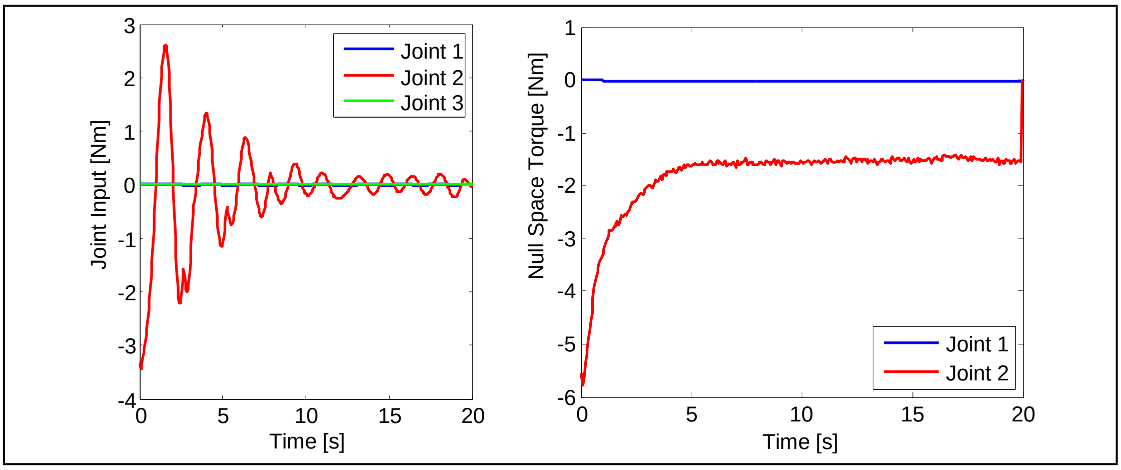
In operational space control scheme, joint 3 is set as passive joint with using underactuation matrix. Initial end effector positions are 0.132, 0, 0 m in x, y, and z directions, respectively. Underactuated motion is tested with PD gains $K_p = 15$ and $K_d = 5$ for a constant reference 0.03 m in z direction. Since the robot has not enough redundancy to reach a target position without changing its current position, slight movements are observed in other axes to achieve the goal. End effector motions are given in Figure 5.16.

Figure 5.16 : End effector position and velocities in case of a constant input



Joint 1 and 2 are active joints, but it is seen that the joint 1 is kept at its initial position to preserve the contact condition as a result of the desired end effector motions. In order to manipulate the passive joint, the null space torque of joints which creates input torques in the same axis of passive joint.

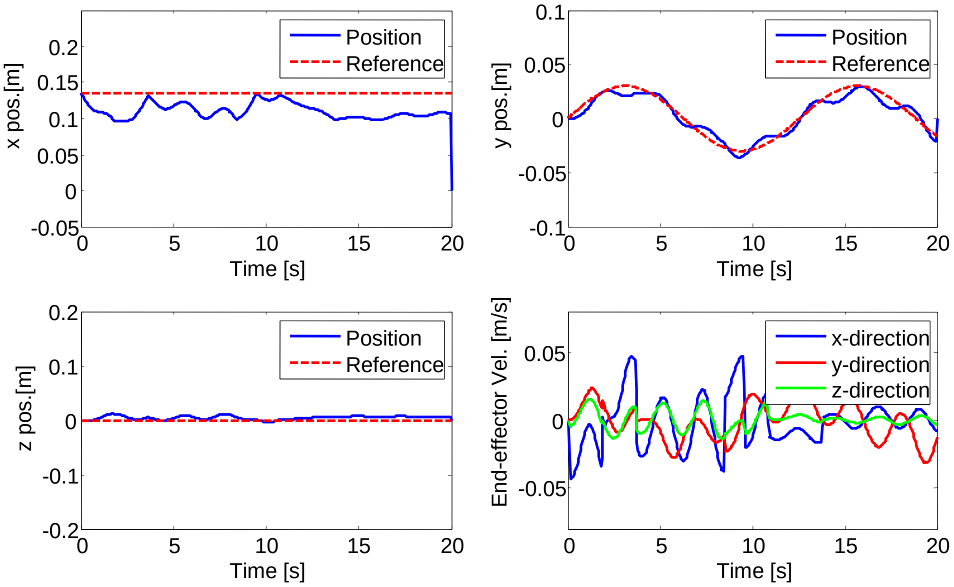
Figure 5.17 : Joint input and null space torques



Another test is performed with time varying reference input is given next while PD controller gains are $K_p = 10$ and $K_d = 0$. Initials end effector positions are same while the desired position is 0.03sin(0.5t) m applied in y direction. The tracking performance of controller is given in Figure 5.18. In this case, in order to reach the desired position, joint 1 and 2 move simultaneously.

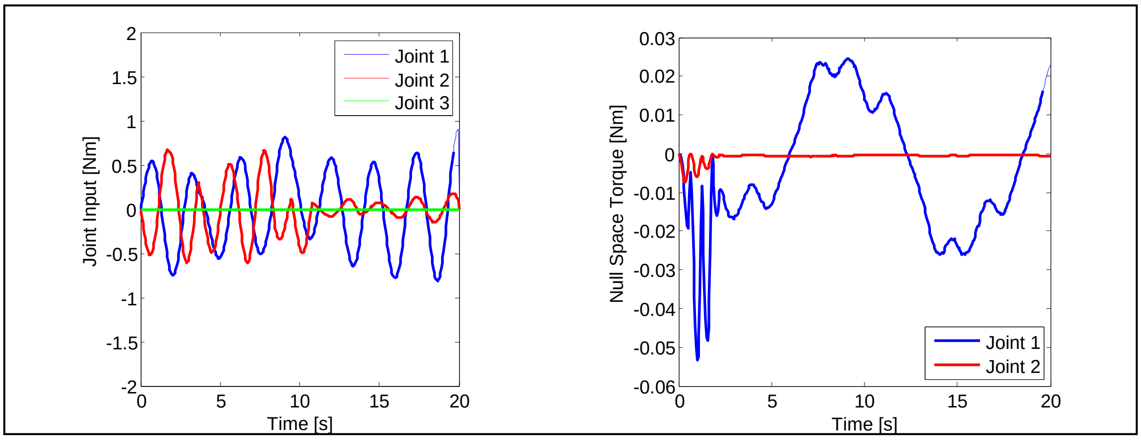
Similar with the previous case, slight movements are seen in other axes although there is no desired movement, since the robot has not enough redundancy.

Figure 5.18 : Tracking performance in case of a time varying reference input



The input torques and null space torques are given in Figure 5.19. The null space torque compensates the passive joint motion.

Figure 5.19 : Joint input and null space torques

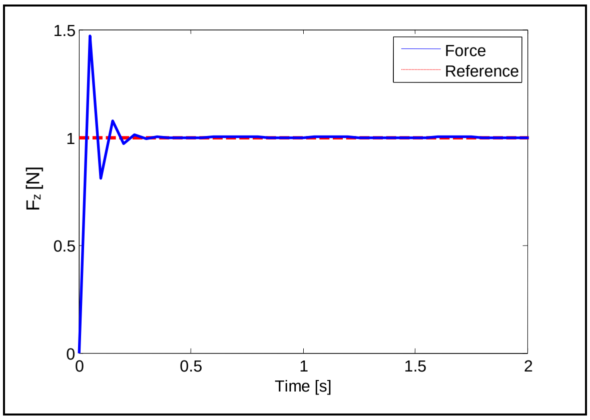


5.3 TEST OF THE MOTION/FORCE CONTROLLER PERFORMANCE IN OPERATIONAL SPACE

Before testing of motion/force control scheme, the pure force control law in Equation (4.36) is performed. In case of a constraint in vertical axis, the contact force can be applied in z-direction.

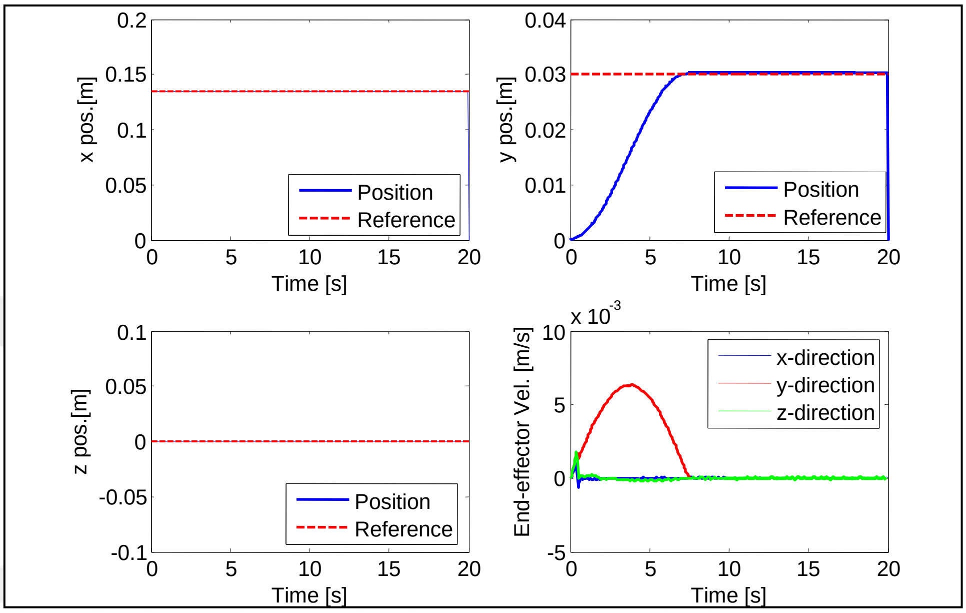
With a controller gain $K_f = 0.4$, the end effector force is reached to desired value within a short period while desired force is applied as $F_z = 1$ N.

Figure 5.20 : End effector force in z-direction



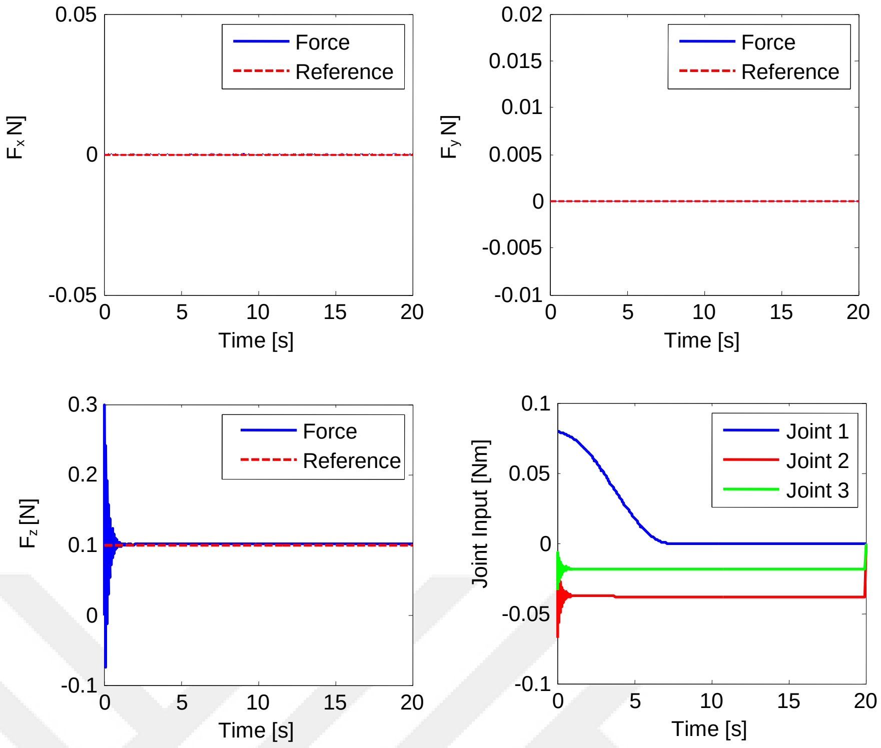
In order to test the performance of the motion/force controller considering the manipulator as fully actuated, Equation (4.39) is implemented. Set point motion control performance for constant references is given in Figure 5.21, while PD motion controller gains are 55 and 10, and force control gain is 0.8, respectively.

Figure 5.21 : End effector position and velocity



Operational space forces are plotted in Figure 5.22.

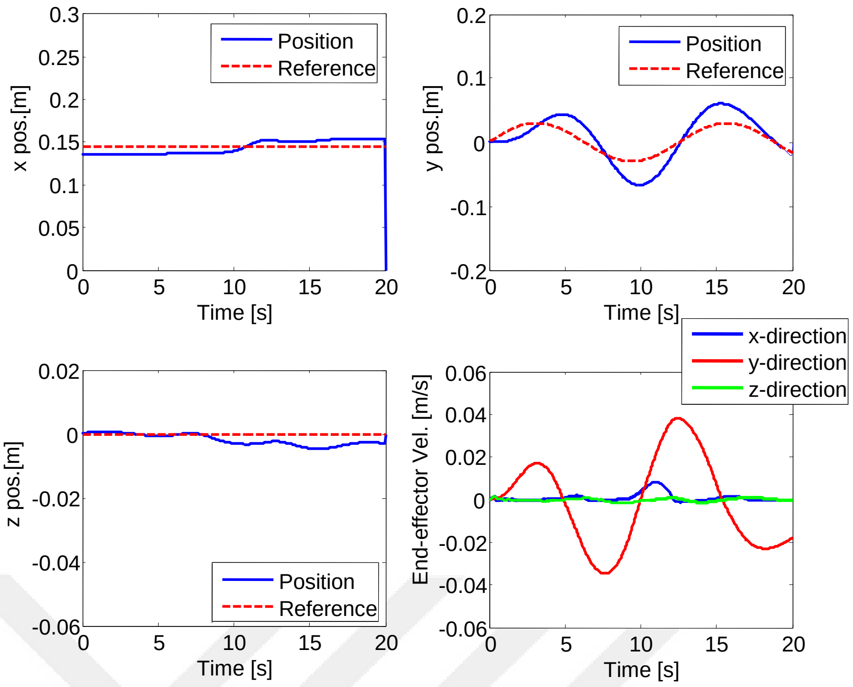
Figure 5.22 : End effector forces



According to the Figures 5.21 and 5.22 which show the end effector position and forces, the tip of the robot manipulator reaches to the desired trajectories with accuracy while the 0.1 N contact force is applied in the z-direction. It can be seen from the Figure 5.22 that although the end effector reached to its desired position, joint 2 and joint 3 continue to produce input torque to achieve the desired force in z direction. End effector forces are calculated by converting joint torques with Jacobian transpose.

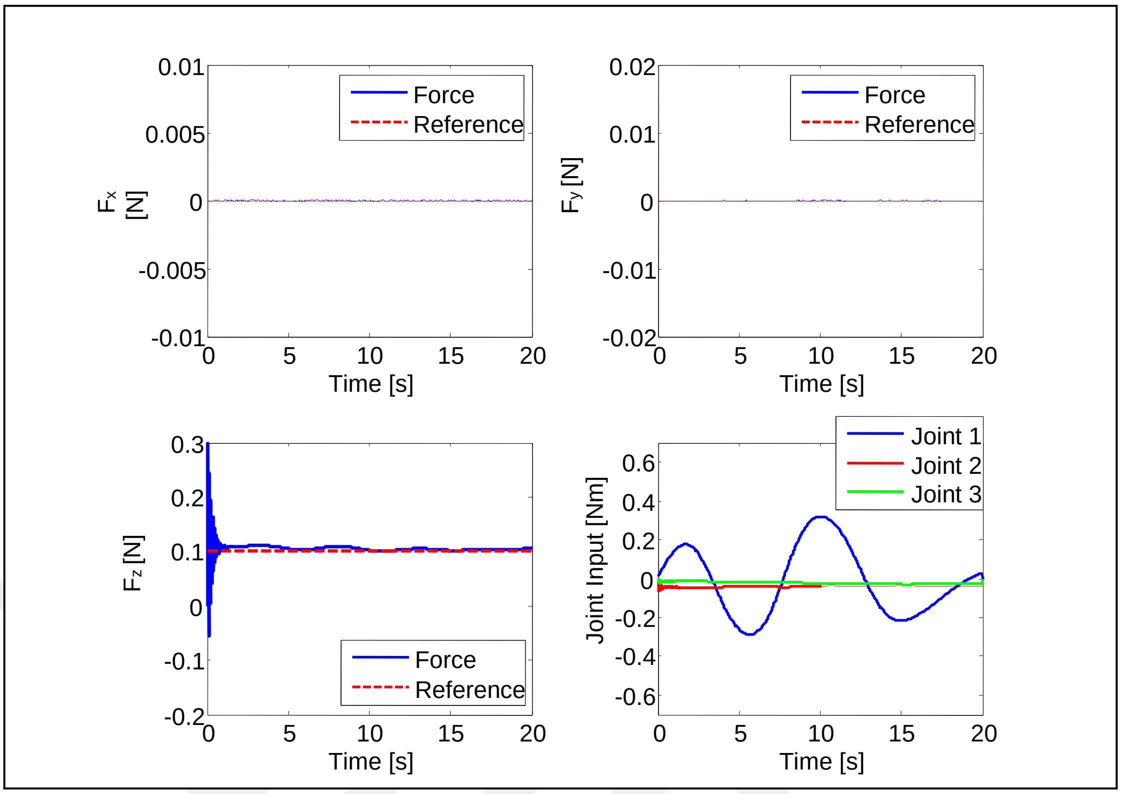
The tracking performance of controller in case of a time varying reference input is given in Figure 5.25 while PD controller gains are $K_p = 65$, $K_d = 5$ and force control gain is $K_f = 0.8$.

Figure 5.23 : End effector position and velocity



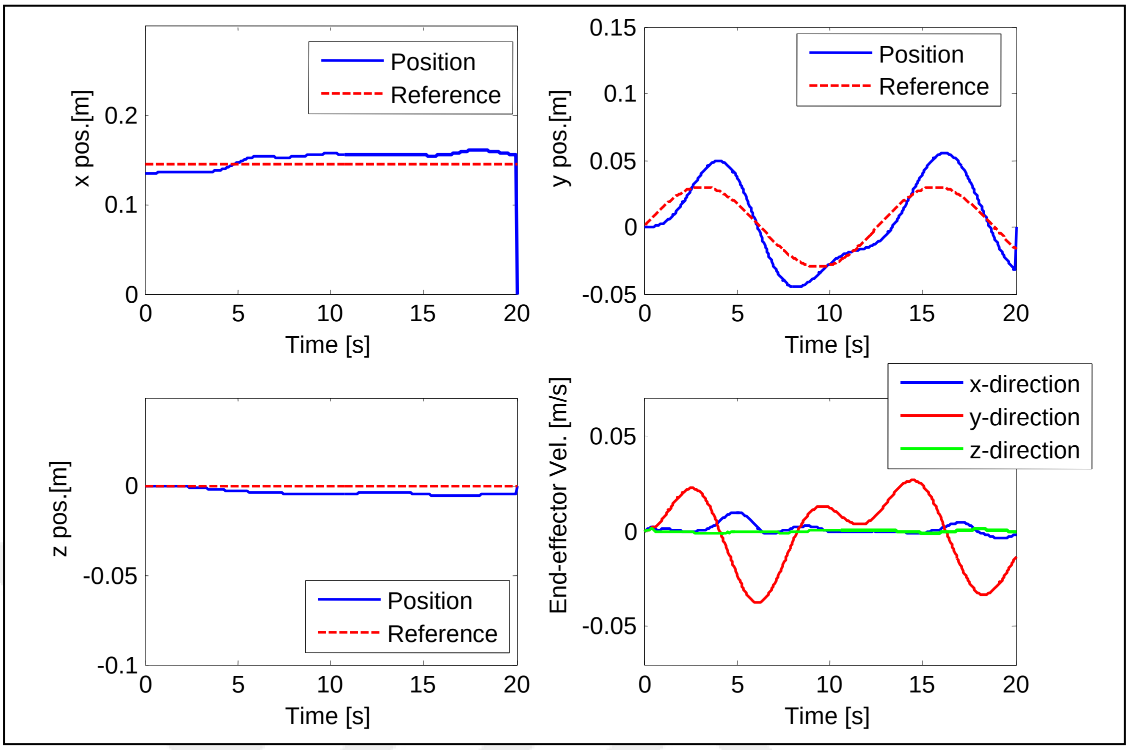
Since the robot has not enough redundancy, reaching to the desired trajectory in y axis causes slight movements in other axes. The force controller performance of the robot for a sinusoidal trajectory is given in Figure 5.23 while the desired end effector force is chosen as 0.1 N. End effector forces are given in Figure 5.24.

Figure 5.24 : End effector forces



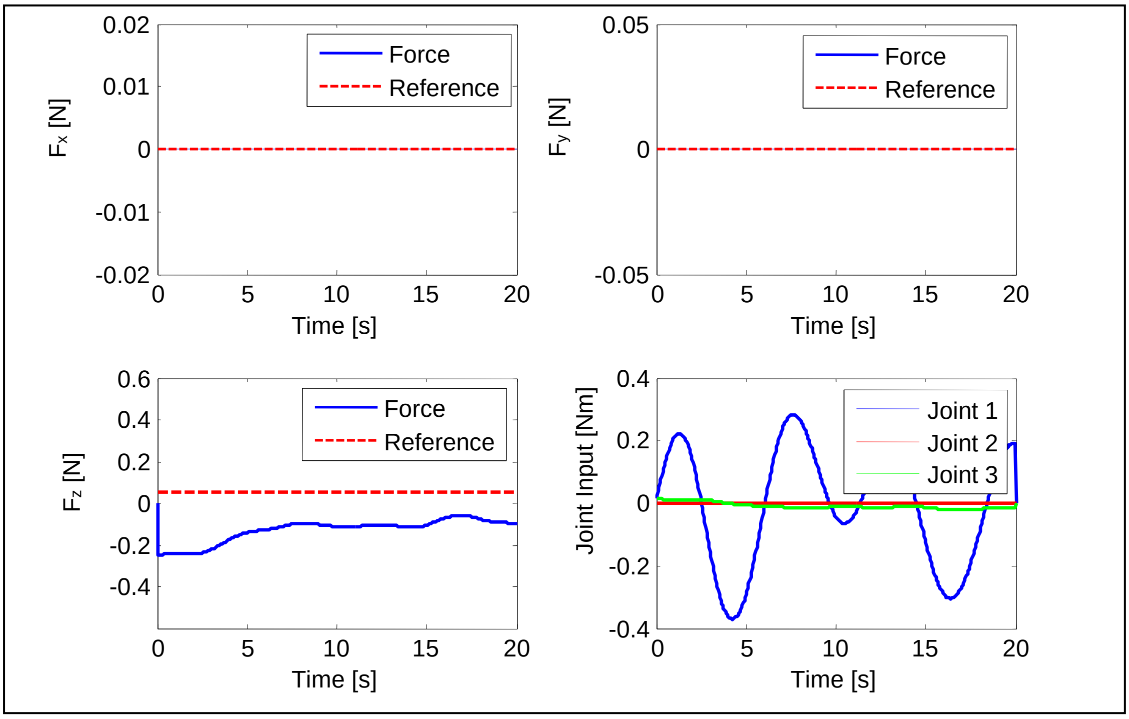
In case of underactuation, the desired end effector forces using mapping to joint variables with Jacobian transpose may not be produced. In this case, the task space dynamics without the addition of null-space torques in general may not be achieved, and task and null space dynamics cannot be decoupled. In order to perform the underactuated case in operational space, joint 2 has been chosen as passive. Tracking performance of end effector is given in Figure 5.25.

Figure 5.25 : End effector position and velocity when joint 2 is passive



Operational space forces are plotted in Figure 5.26.

Figure 5.26 : End effector forces when joint 2 is passive



As a summary, projected inverse based control schemes are applied for both joint space and operational space in simulation studies. In joint space, motion control is performed for fully actuated and underactuated cases for PHANTOM Omni robot and robot is achieved the desired trajectories with accuracy.

In operational space, in addition to the motion control, force control is also implemented. Although Omni is a 3-DOF robot, it has two joints (joint 2 and 3) which generate the torque in same direction. Hybrid motion force/control goal cannot be successfully performed for underactuated case, since only one control input remains due to underactuation.

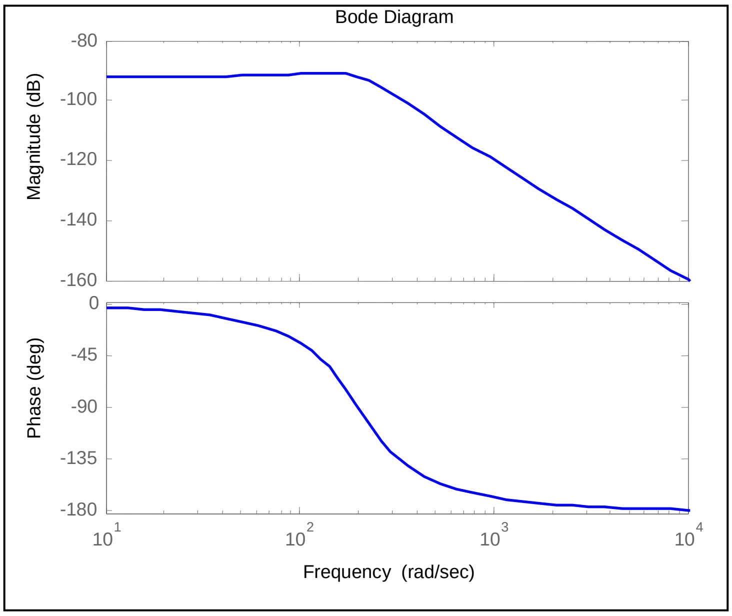
6. EXPERIMENTAL RESULTS

The PHANTOM Omni robot used for experimental evaluation is a 6-DOF robot with three active revolute joints in addition to three wrist joints that are non-actuated. At the tip of the non-actuated joints, there is a stylus which was fixed to link 3 during the experiments. In this position, last 3 joints remain fixed which allow to make analysis assuming only first three joints.

Several experiments are performed by applying the proposed control algorithms to PHANTOM Omni robot. The control scheme is based on the encoder feedback and the estimation of contact forces to move joints through the desired positions of the workspace and providing the required torques at the joint motors. The angular velocities and accelerations of the joints are calculated by using the position information providing from the encoders.

In order to filter out the noise due to differentiation, second order filters are used. To eliminate the noises for joint velocities, bode plot of the filter is given in Figure 6.1.

Figure 6.1 : Bode plot of the second order filter



6.1 ASSUMPTIONS AND METHODS

The following are the assumptions made for implementing the control algorithm:

a. **Link Properties:** Link properties are provided in Chapter 3 but are repeated here for convenience. In the PHANTOM haptic interface hardware installation and technical manual, the masses of required parts were measured using an electronic weighing scale. The length of link 2 and link 3 is given as 0.132 m.

For the calculation of inertial parameters, the first link has been assumed as a sphere while link 2 and link 3 are assumed as cylinders. The calculation of the center of masses and the moments of inertia was done using these approximations and assuming they are homogeneous rigid bodies. In order to implement appropriate desired motion for control study, the reachability of joints is considered as given in Table 6.1 and Table 6.2.

Table 6.1: Joint limits for PHANTOM Omni

Joint	Min (Rad)	Max (Rad)
q1	-1.01	0.954
q2	-1.79	-0.033
q3	3.47	5.58

Table 6.2: End effector limits

Coordinate	Min (m)	Max (m)
x	0.103	0.288
y	-0.249	0.241
z	-0.129	0.189

b. **Torque calculations:** The frictional effects are ignored during the entire analysis. These effects could be accounted for by multiplying the torques by appropriate coefficients.

The total current consumption is limited as 2.2A to prevent over-current of the joint motors. In order to figure out this issue, smart saturation is used to allow channels to go up to 1.6A unless the total exceeds 2.2A in which case all channels are scaled so that the cumulative current is equal to this current value.

c. **Conversion of torque to PWM values:** The assignment of joint torques to the motors was achieved using Digital to Analog (DAC) values. It is assumed that there is a linear relationship between the DAC values and the torques.

6.2 SOFTWARE

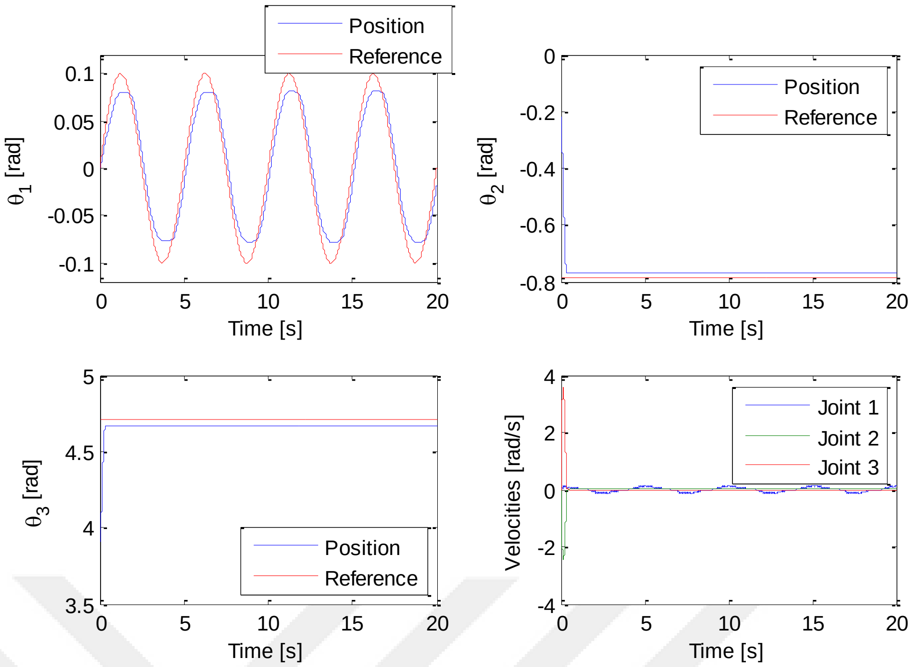
In order to control PHANTOM Omni robot, Quanser’s QUARC software has been used which has powerful tools to Matlab/Simulink to make the development and implement of complicated real-time applications easier. Toolbox generates real-time code directly from Simulink designed controllers and runs it in real-time on the Windows target. The HIL API from the toolbox has been used to access the robot to send joint commands and to read the encoder values.

6.3 RESULTS

First, joint space motion control is tested. Controller gains used in experiments are obtained by trial and error.

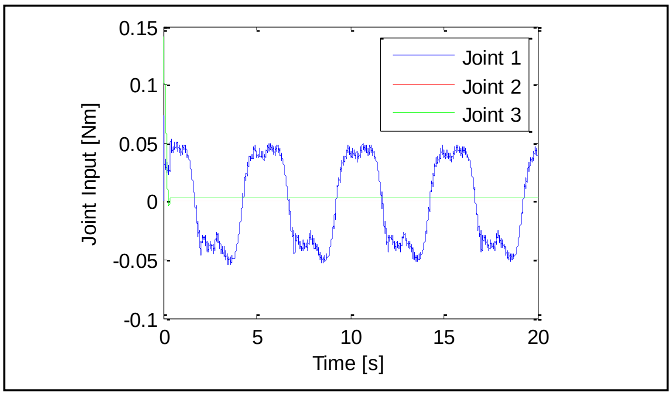
To test the underactuation, the actuator projection matrix is chosen to set the joint 2 as passive that makes the control input of this joint is 0. Desired positions are $\Theta_1 = 0.1\sin(0.4\pi t)$, $\Theta_2 = -\pi/4$ and $\Theta_3 = 3\pi/2$ rad for each joint, where initial positions are 0, -0.26 and 3.78 rad respectively. Controller proportional and derivative gains are Gp = 40 and Gd =1. In case of a constraint in vertical axis, motion control performance is given in Figure 6.2.

Figure 6.2 : Joint positions and velocities when joint 2 is passive



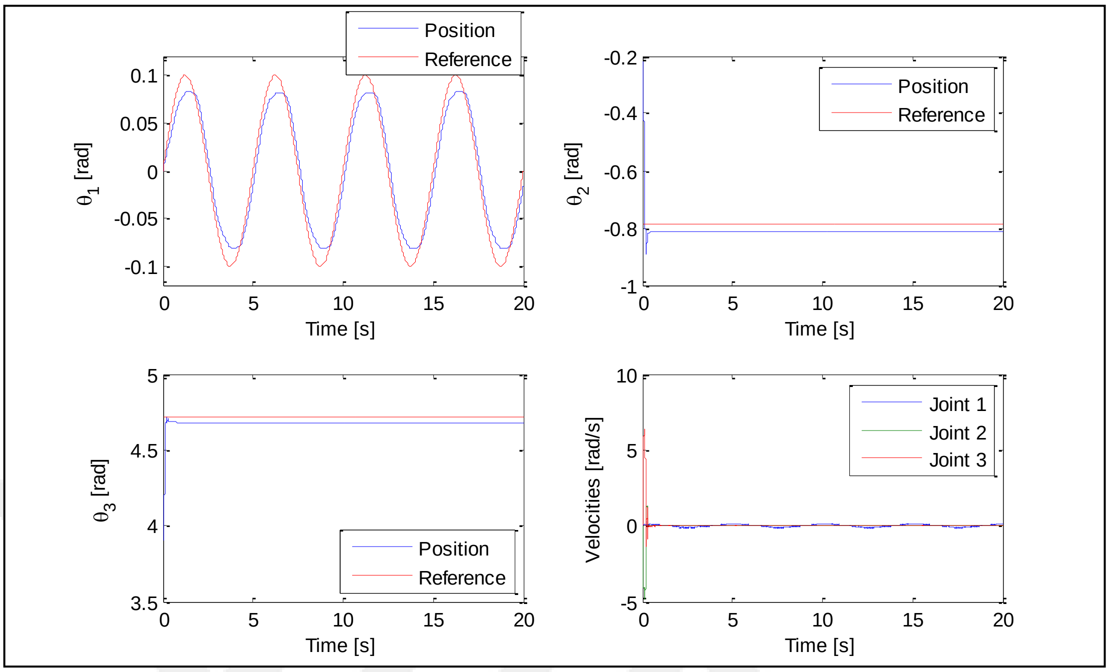
According to the Figure 6.2, the max position error of joint 1 is 0.023 rad, steady state errors are 0.017 rad and 0.041 rad for joint 2 and 3, respectively. The joint inputs are given in Figure 6.3 to make pure motion control.

Figure 6.3: Joint torques when joint 2 is passive



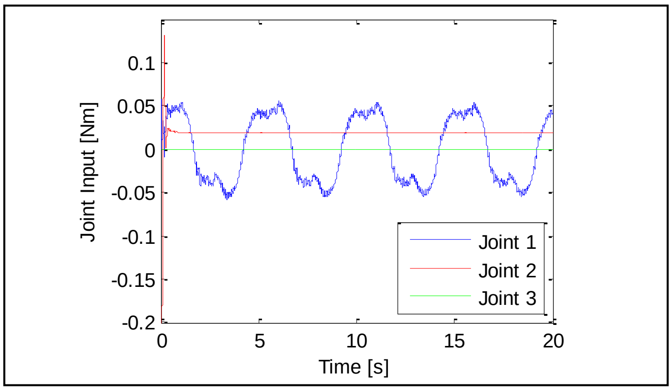
Secondly, motion control is tested while joint 3 is passive with same desired and initial positions. The max position error of joint 1 is 0.021 rad and steady state errors are 0.025 rad 0.032 rad for joint 2 and 3, respectively. Motion control performance is given in Figure 6.4.

Figure 6.4 : Joint torques when joint 3 is passive



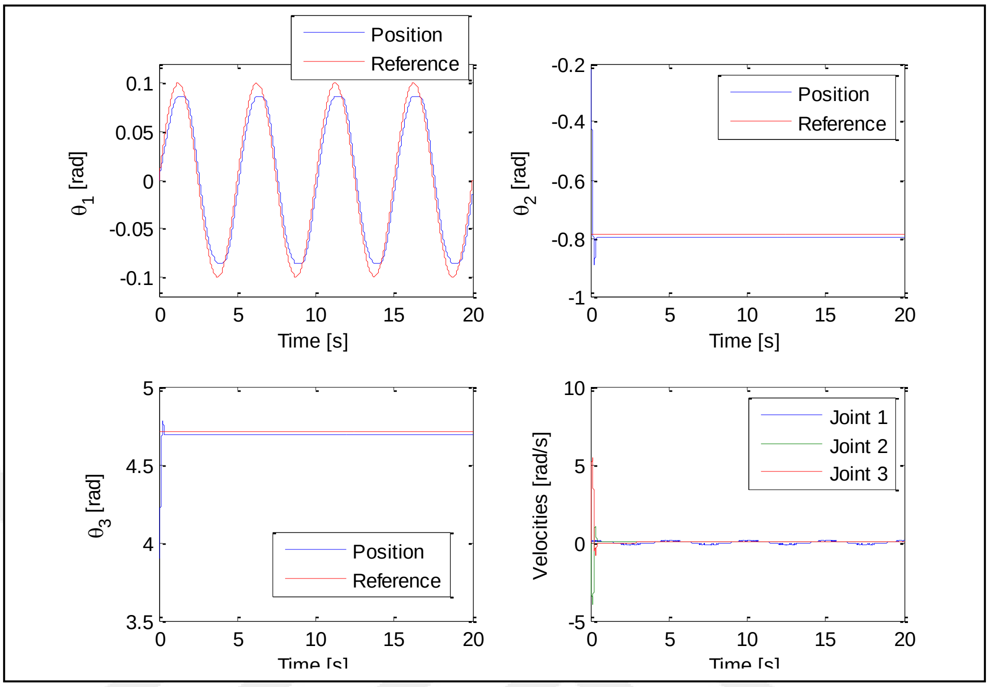
The joint inputs are given in Figure 6.5.

Figure 6.5 : Joint torques when joint 3 is passive



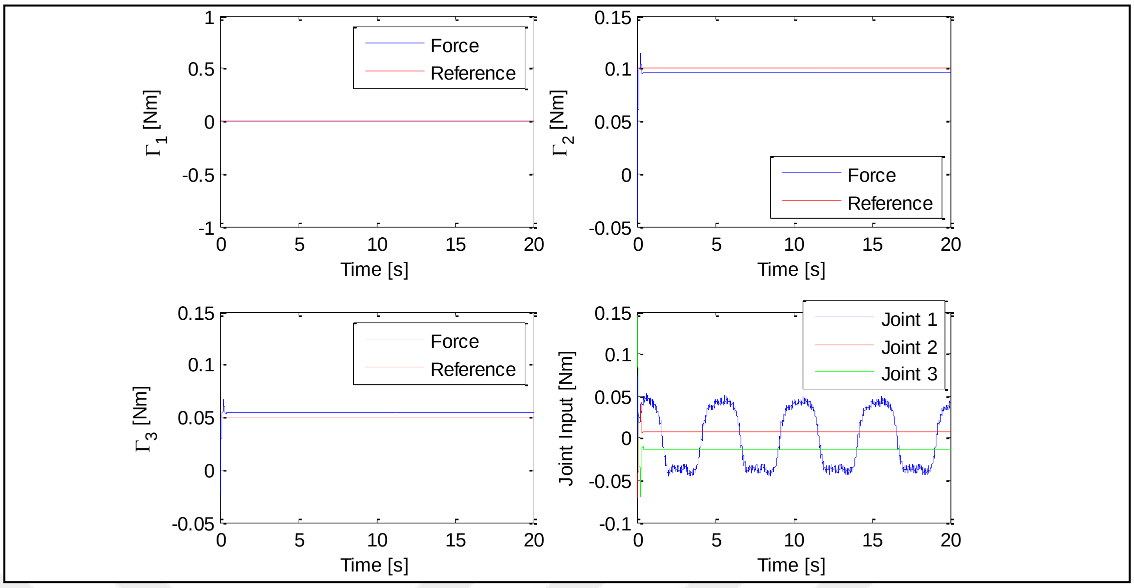
Hybrid motion/force control for fully actuated case is tested next in joint space. In this case, Equation (4.21) is implemented with same desired and initial positions. Desired contact forces are chosen as 0, 0.1 and 0.05 Nm for each joints and force gain is G_f = 0.7. According to the Figure 6.6, the joints are tracking desired path. The max position error of joint 1 is 0.015 rad and steady state errors are 0.013 rad and 0.022 rad for joint 2 and 3, respectively.

Figure 6.6 : Joint positions and velocities in case of hybrid motion/force control



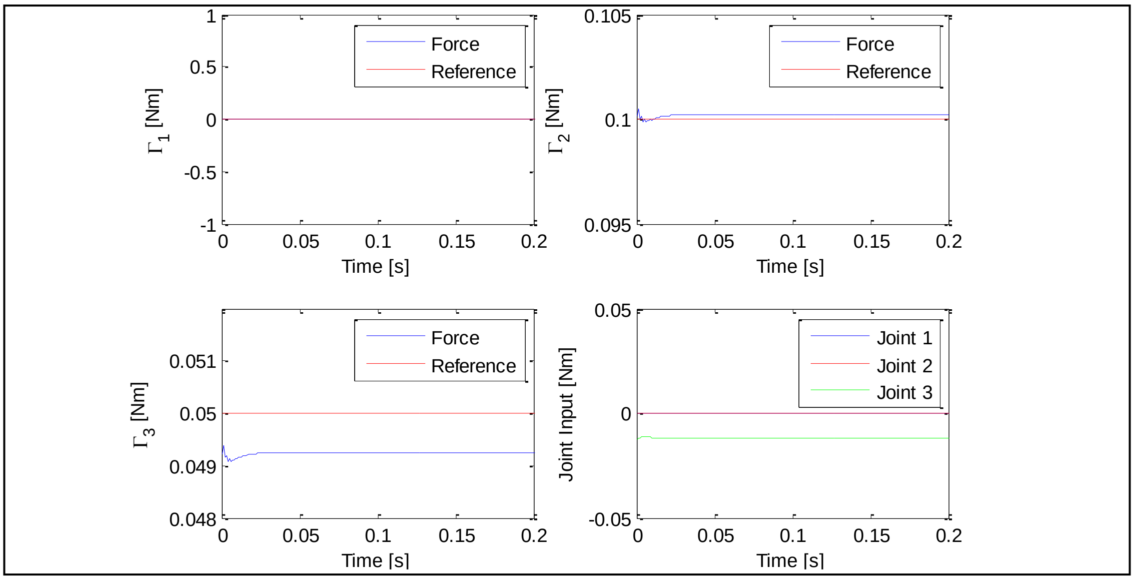
Joint 1 does not produce any torque in constraint axis; hence the contact of the robot with an obstacle in this direction does not produce any constraint force at this joint. The constraint forces at the joints and input forces are given in Figure 6.7. Steady state errors for constraint force are 0.004 Nm and 0.003 Nm for joint 2 and 3, respectively.

Figure 6.7 : Joint constraint torques and inputs in case of hybrid motion/force control



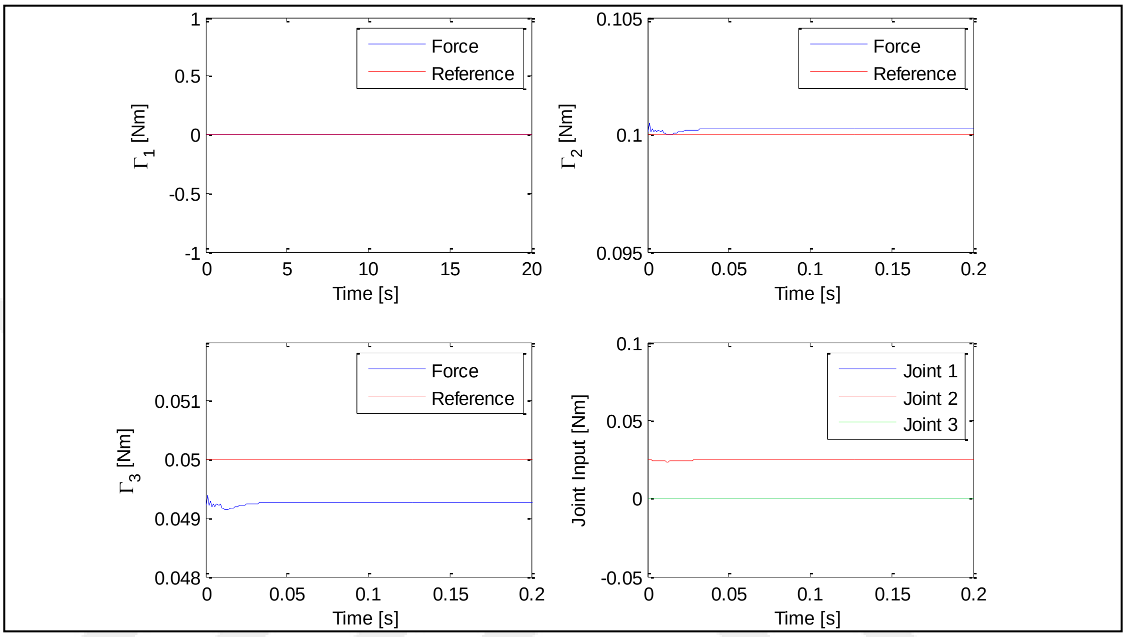
Next, actuator projection matrix is chosen in order to make the joint 2 passive. In case of underactuation, since robot has only one actuated joint which produces torque in constraint direction, only position or constraint force can be controlled. In order to test the force control performance, joints are kept their initial positions. Desired contact forces are chosen as 0, 0.1 and 0.05 Nm. Force control performance is given in Figure 6.8, while the steady state errors are 0.0002 and 0.001 Nm.

Figure 6.8 : Joint constraint torques and inputs in case of passive 2nd joint



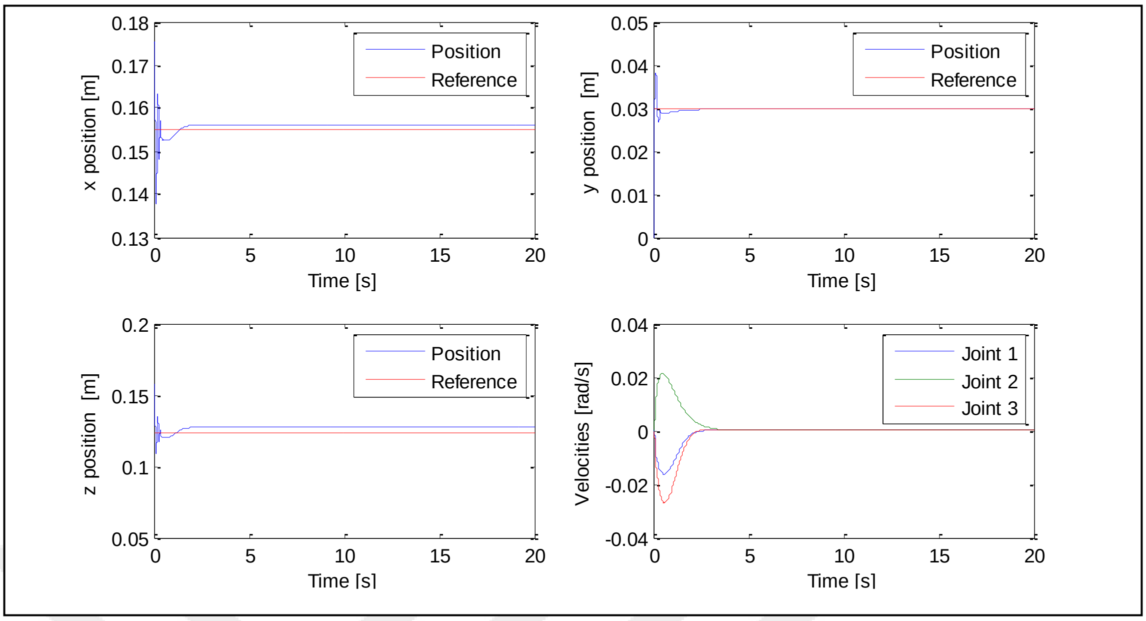
In order to test the force control performance for passive 3rd joint, while joints are kept their initial positions. Desired contact forces are same, force control performance is given in Figure 6.9.

Figure 6.9 : Joint constraint torques and input in case of passive 3rd joint



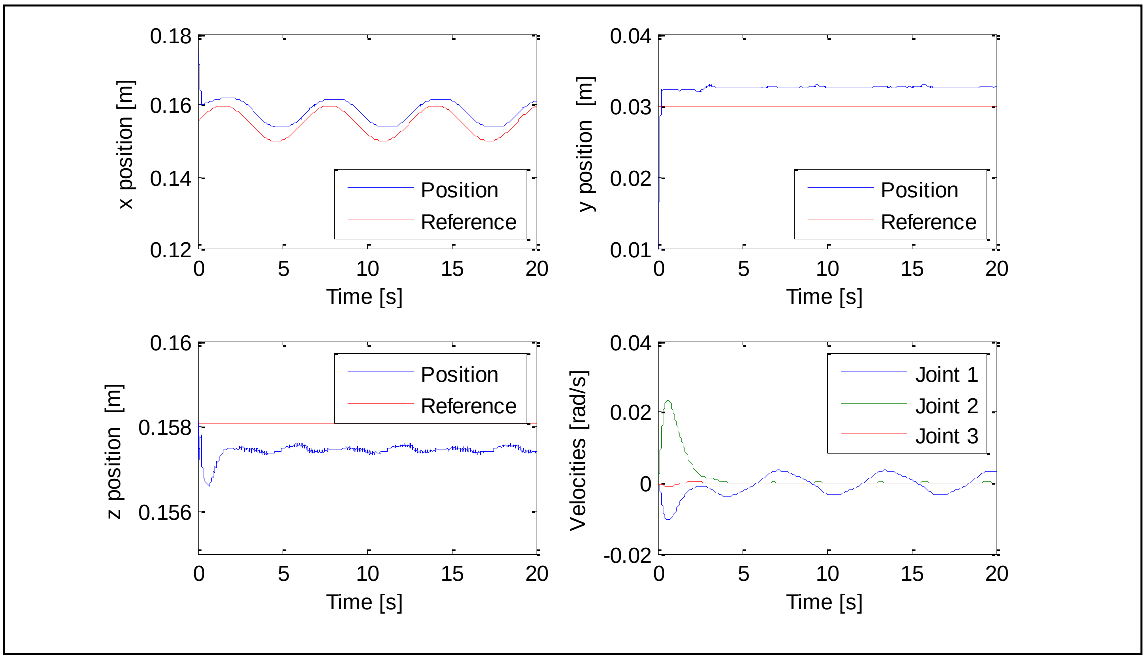
For defining the desired motion in terms of operational space, the motion control law in Equation (4.34) is implemented. Response for constant reference is given in Figure 6.10 with $x(t) = x(0) + 0.02$, $x(t) = x(0) + 0.03$, and $x(t) = x(0) - 0.035$ m desired positions in direction of x, y and z axes.

Figure 6.10 : End effector position and velocity in case of a constant reference



Tracking performance for a time varying reference is given in Figure 6.11. Desired position of x is $x(t) = x(0) + 0.001\sin(2\pi t)$ while desired positions in y and z axes are same with the previous case.

Figure 6.11 : End effector position and velocity in case of a time varying reference



Controller schemes are implemented experimentally for joint space in order to test the motion and force control performance. In case of underactuation, position and constraint force are controlled separately, since the robot has only one control input in direction of underactuation. In operational space, set point and time varying motion control performance is tested for fully actuated case of robot.

7. CONCLUSION

This dissertation describes the process of modeling, simulation and control of an underactuated robot manipulator considering interaction with its environment. Dynamic simulation gives an opportunity to analyze the time varying behavior of the robot by giving a virtual response close to the actual system using mathematical equations. The use of V-REP as a tool to model and simulate the mechanics of the robot allows the possibility to verify control algorithms. Subsequently, the validity of control algorithms is tested experimentally. Simulator and experiment environments accept joint positions as input, thus applying joint torques as control input requires making assumptions.

In this study, controllers for the underactuated robots are modeled by first considering the projected inverse based dynamics and then solving underactuation by the addition of appropriate control torques. In presence of passive joints, the underactuation matrix is used to rebuild controller laws in order to perform a manipulation in a way that robot can achieve the desired motion and force. The controllers based on projected inverse dynamics are developed both in joint space and operational space. In joint space underactuated controller scheme, the controller uses constraint forces to create lost torque at the joints that are passive. In operational space underactuated control method, the null space motion is used besides the constraint force in order to achieve the desired motion.

Motion and force feedback control loops are developed and implemented together through decomposition of the input channel. Another plus is to estimate the constraint forces by using the control input while robot moves on a constraint surface. The estimation of forces in a force feedback control is useful approach, because the usage of sensors increases the complexity of system and decreases the stability of control.

An insight into the use of projected inverse based control shows its adequacy to control the environment interaction of a robot. It can be asserted that a compliant behavior is achieved, although some differences occur in the performance of the control schemes while robot is constrained in vertical or in horizontal direction since the gravitational affects. The number of active joint is important to achieve the desired position accurately, on condition that they have axes of rotation in the same direction with the passive joint. The same axis of rotation is important in order to obtain control torques in the same direction to compensate the lost torque at passive joint. In this study, since there is only one active joint at axis of rotation in passive joint, rise of gravitational effect in horizontal case decreases the accuracy and precision of control.

REFERENCES

Books

Chung W, Fu L, Hsu S, 2008, Motion Control, *Springer Handbook of Robotics*, Springer-Verlag Berlin Heidelberg, pp. 133-159

John J. Craig, 2005, Manipulator Dynamics, *Introduction to Robotics*, *Mechanics and Control*, Third Edition, Pearson Education International, pp. 165-200

Mark W. Spong, Seth Hutchinson, M. Vidyasagar, 2005, Forward and Inverse Kinematics, *Robot Modeling and Control*, John Wiley&Sons, Inc., pp. 65-112

Periodicals

Aghili, F., 2003, Inverse and Direct Dynamics of Constrained Multibody Systems Based on Orthogonal Decomposition of Generalized Force, *International Conference on Robotics & Automation*, pp. 4035-4041

Aghili, F., 2005, A Unified Approach for Inverse and Direct Dynamics of Constrained Multibody Systems Based on Linear Projection Operator: Applications to Control and Simulation, *Proceedings of the IEEE Transactions on Robotics&Automotion*, **21** (5), pp. 834-849

Bergerman, M., Xu, Y.,1994, Robust Control of Underactuated Manipulators: Analysis and Implementation, Systems, Man, and Cybernetics, *Humans, Information and Technology*, **1,** pp. 925-930

Blajer, W., 2002, Augmented Lagrangian Formulation: Geometrical Interpretation and Application Systems with Singularities and Redundancy, *Multibody System Dynamics*, **8** (2), pp 141-159

Bloch, A.M., Chang, D.E., Leonard, N.E., Marsden, J.E., 2001, Controlled Lagrangians and the stabilization of mechanical systems II: potential shaping, IEEE Transactions on Automatic Control, **46** (10), pp. 1556-1571

Hollerbach, J., Suh, K., 1987, Redundancy resolution of manipulators through torque optimization, *IEEE Journal on Robotics and Automation*, **3** (4), pp 308 – 316

Khatib, O., 1987, A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation, *IEEE Journal of Robotics and Automation*, **3** (1), pp. 43-53

Koul, M.H., Kumar, P., Singh, P.K., Manivannan, M., Saha, S.K., 2010, Gravity Compensation for PHANToM Omni Haptic Interface, *The 1st Joint International Conference on Multibody System Dynamics*

Mistry, M., Righetti, L., 2011, Operational Space Control of Constrained and Underactuated Systems, *Proceedings of Robotics: Science and Systems*, pp. 225-232

Moore, J., Tedrake, R., 2014, Adaptive Control Design for Underactuated Systems using Sums-of-Squares Optimization, *American Control Conference*, pp. 721-728

Spong, M.W, 1994, Partial feedback linearization of underactuated mechanical systems, *Intelligent Robots and Systems*, **1**, pp. 314 – 321

White, W. N., Foss, M., Guo, X., 2007, A Direct Lyapunov Approach for Stabilization of Underactuated Mechanical Systems, in *Proceedings of the 2007 American Control Conference*, **1**, pp 4817-4822

YÖK

Other Publications

Geomagic® Touch™ Haptic Device, http://www.geomagic.com/en/products/phantom-omni/specifications/, [accessed 22.04.2016]

Mathworks, 1994, http://www.mathworks.com/, [accessed 22.04.2016]

Ruiz, F. S., ROS Packages for Sensable Phantom Omni Device, Omni Description, URDF, https://github.com/fsuarez6/phantom_omni, [accessed 05.11.2015]

V-REP Virtual Robot Experimentation Platform, http://www.coppeliarobotics.com/, [accessed 22.04.2016]

APPENDICES

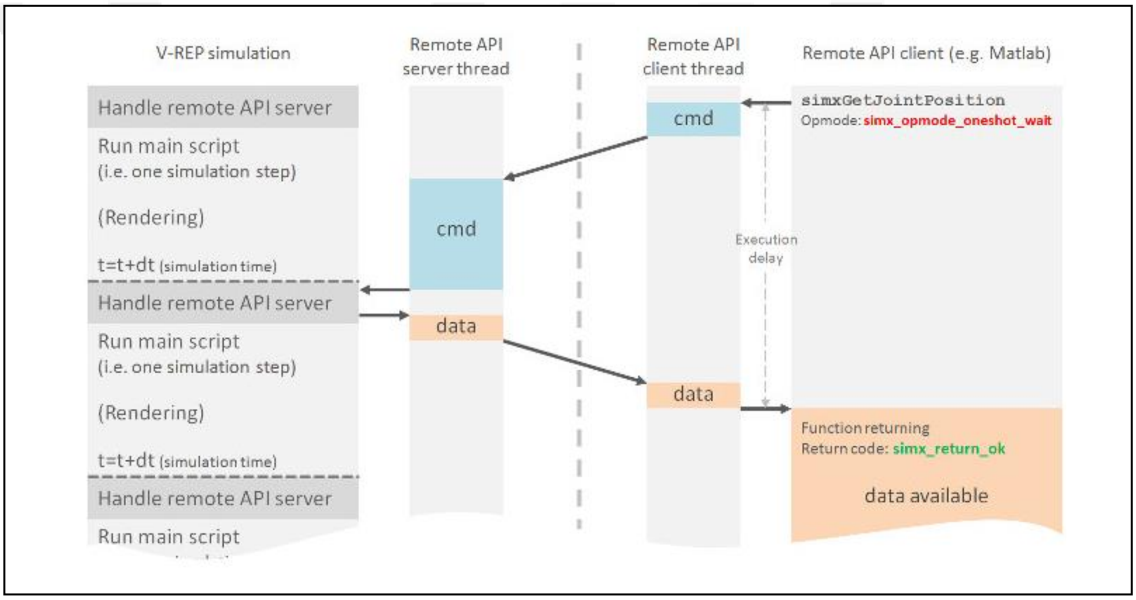
Appendix A.1- Operation Modes for Matlab / V-REP Communication

In order to overcome the time lag problem in server client communication between Matlab and V-REP, there are 4 different modes to control the simulation.

i. Blocking function calls
ii. Non-blocking function calls
iii. Data streaming
iv. Synchronous operation

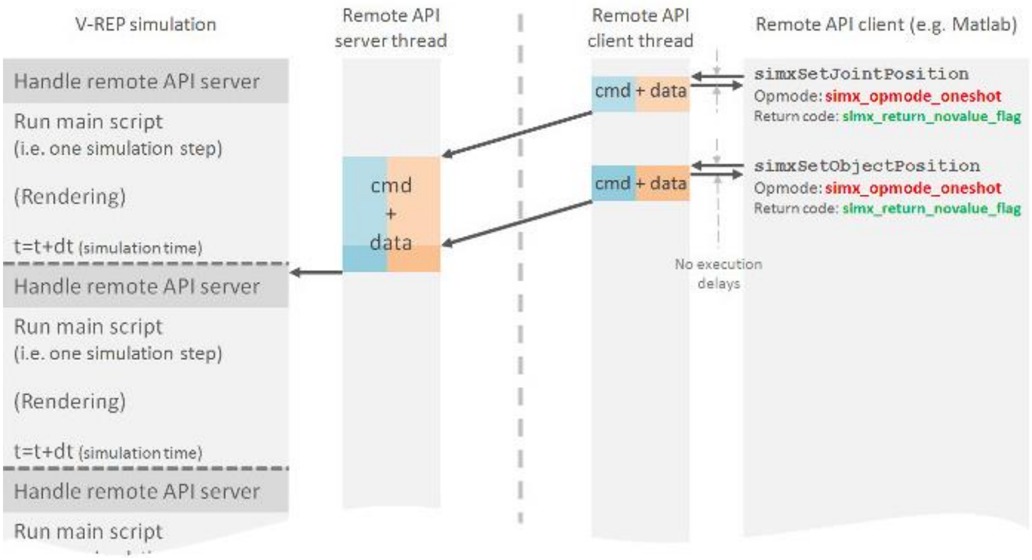
Blocking function call is used for situations when there is no need to wait for a reply from the server. There is an execution delay between sending command and reading data from V-REP Server.

Figure App. 1.1 : Blocking function calls



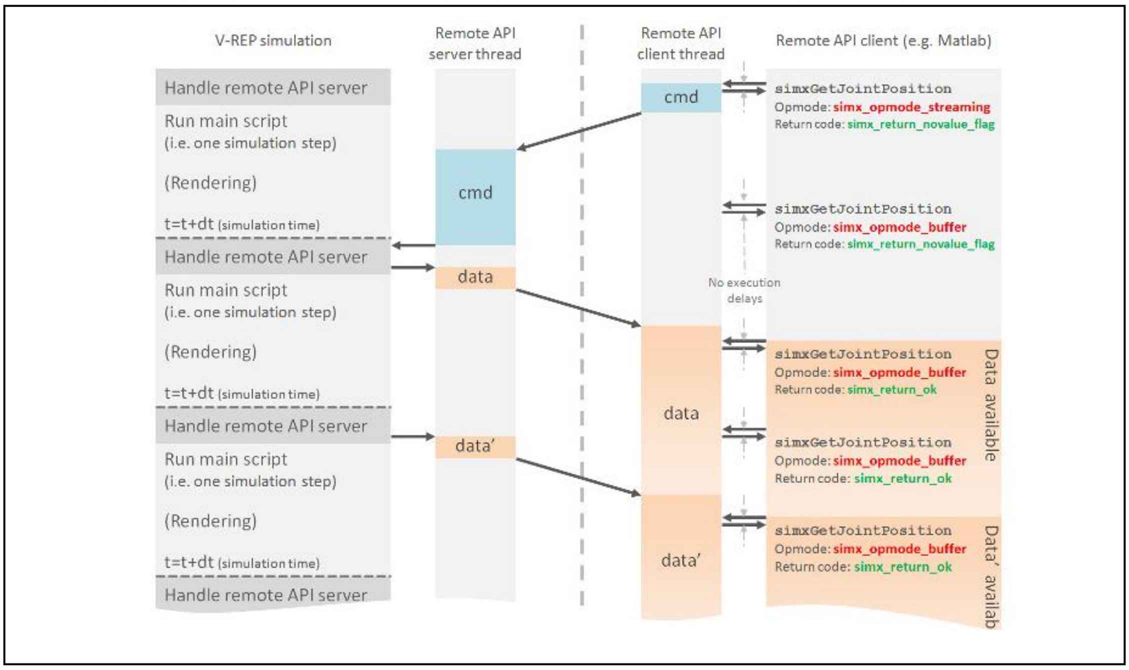
In some situations, there is no need for reply from the server side. In this case non-blocking function call is used to send data to V-REP simply.

Figure App. 1.2 : Non-blocking function calls



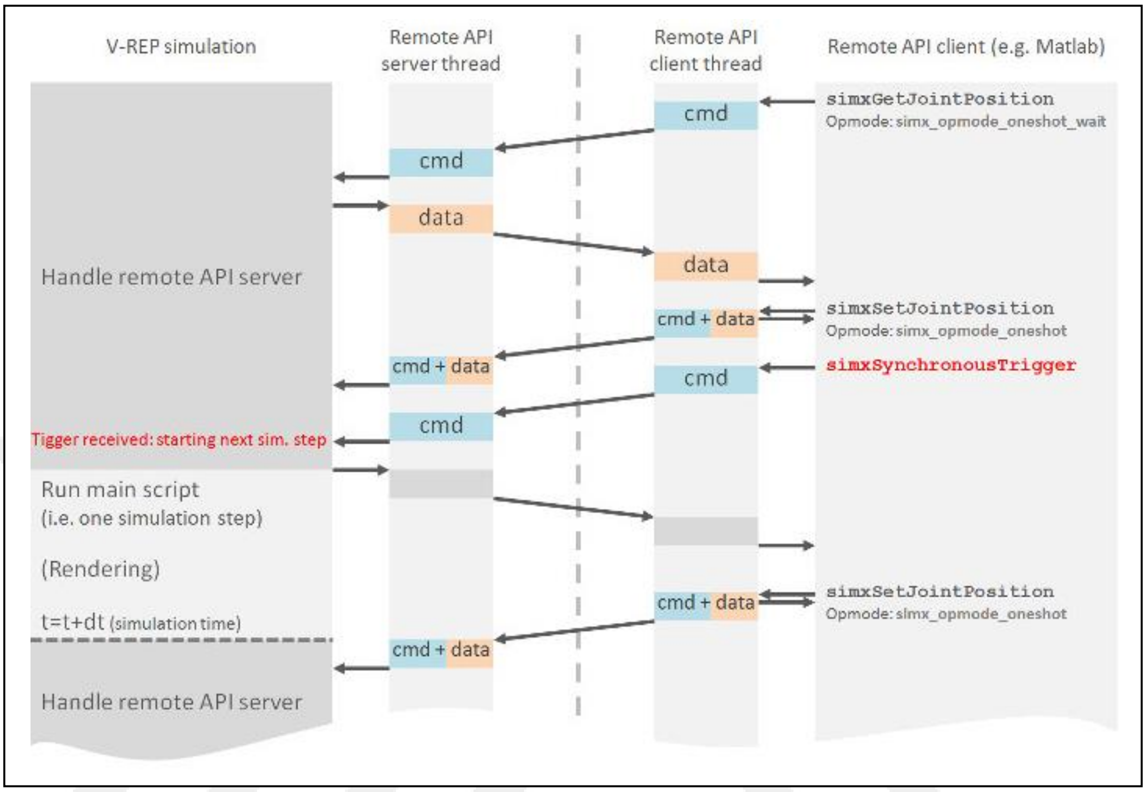
Data streaming is used for continuous data transfer, where the client signals the desire to the server with a "streaming" or "continuous" operation mode flag. This can be regarded as a command/message subscription from the client to the server, where the server will be streaming the data to the client.

Figure App. 1.3 : Data streaming



The situations where the client needs to be synchronized with the simulation progress, by controlling the simulation advance from the client side, synchronous operation mode is used.

Figure App. 1.4: Synchronous Operation



Appendix A.2- Simulink Controllers

Figure App. 2 1 : Position Control Scheme

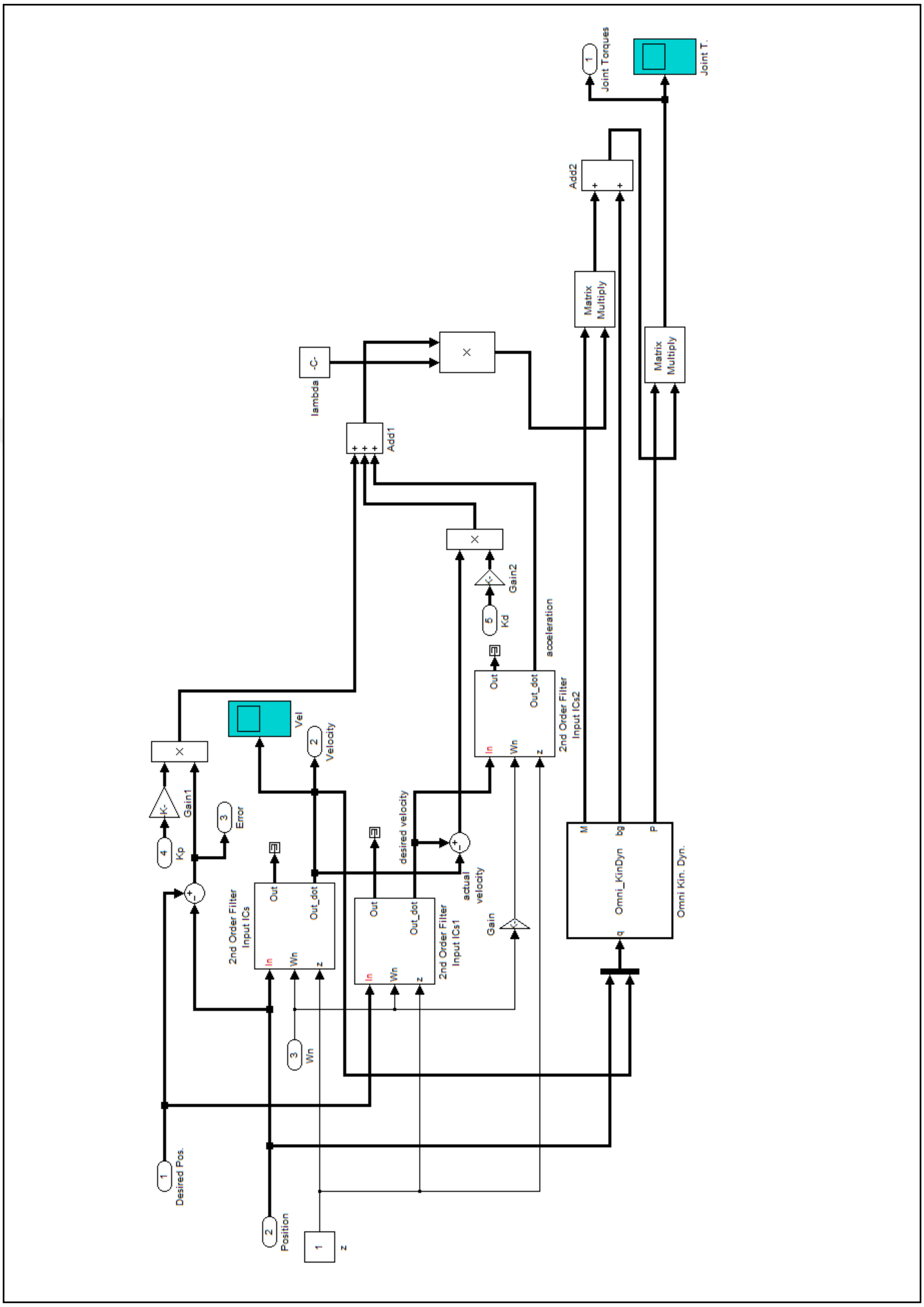


Figure App. 2 2 : Position Control Scheme for Underactuated Case

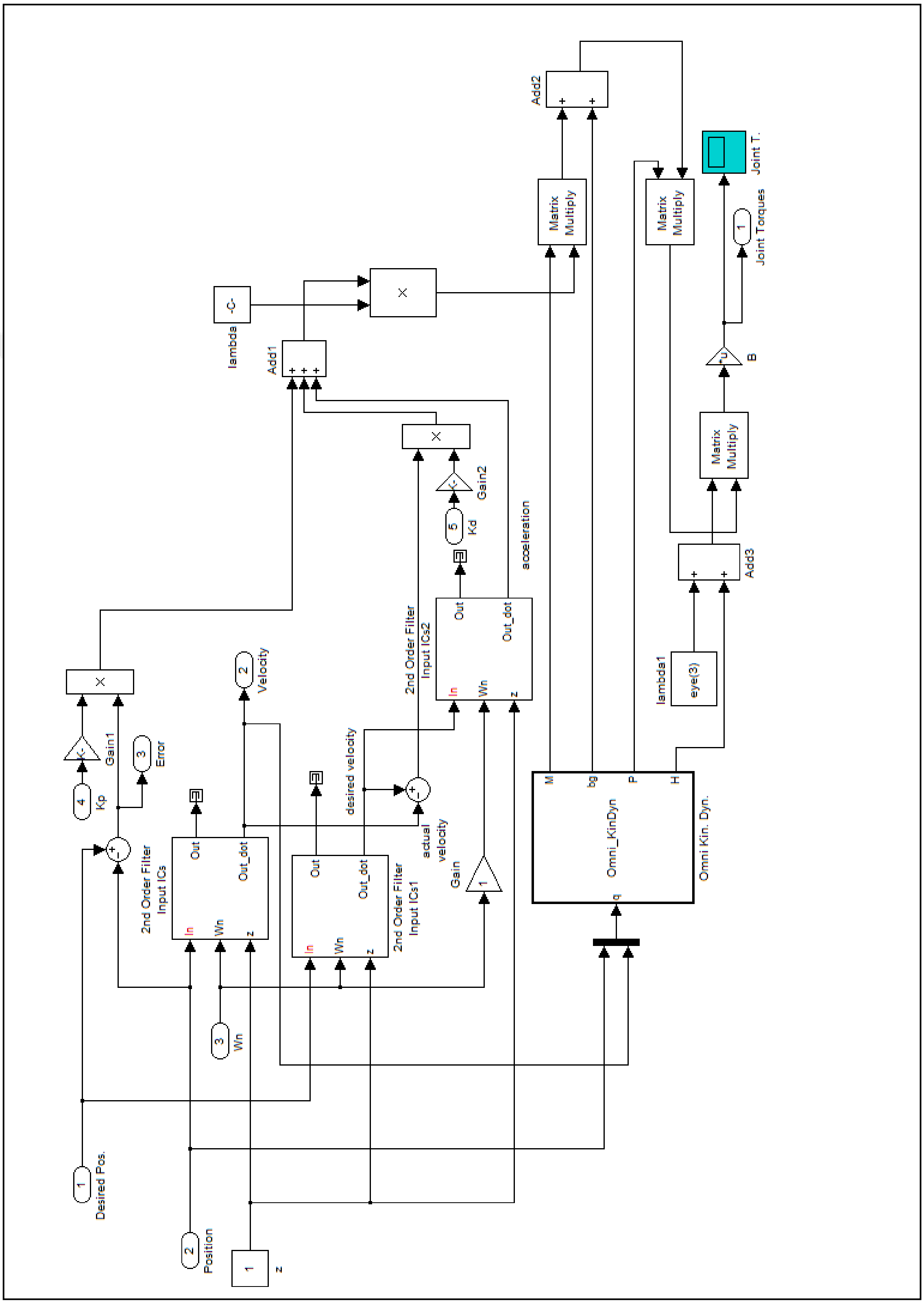


Figure App. 2 3 : Hybrid Control Scheme for Fully Actuated / Underactuated Cases

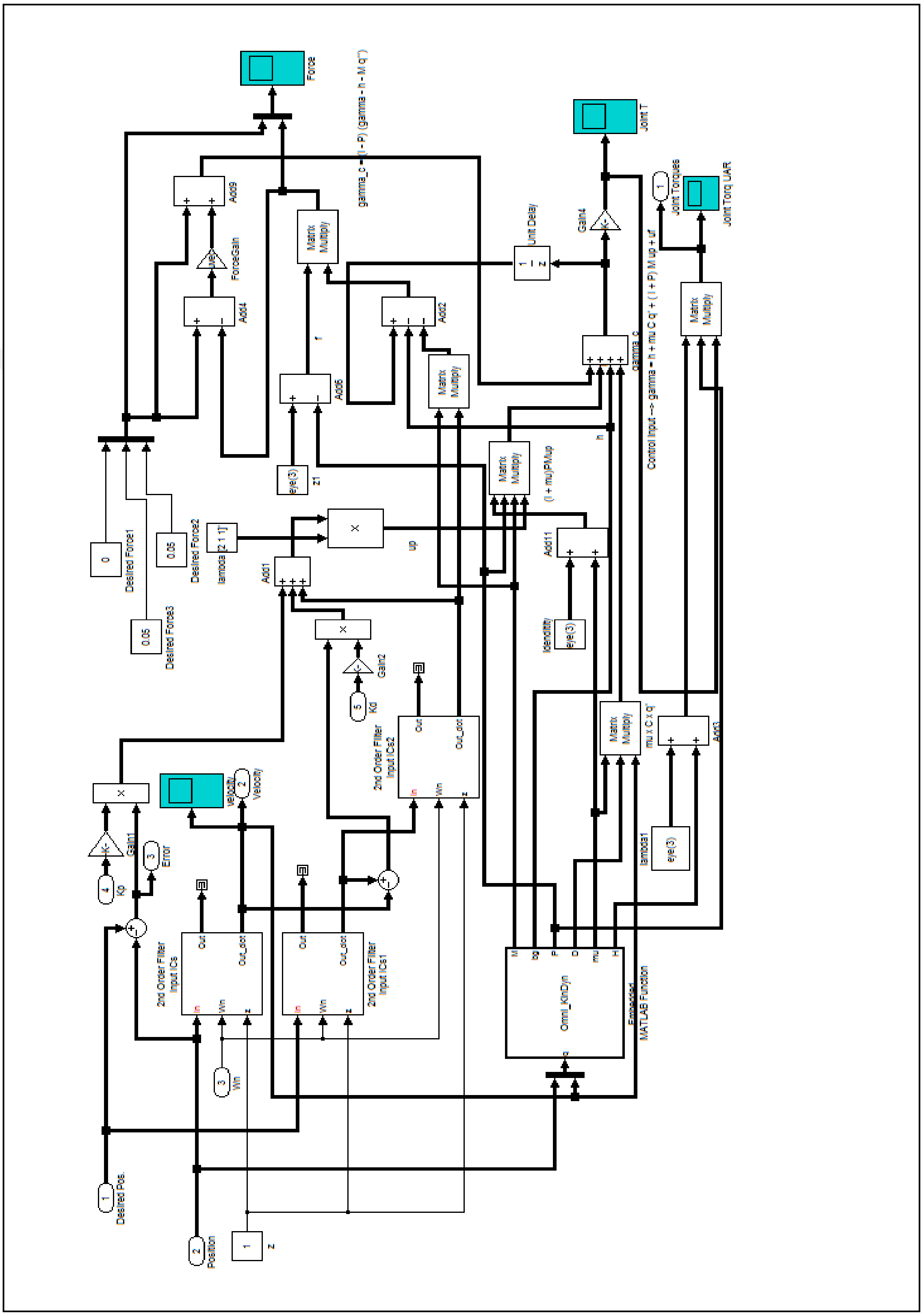
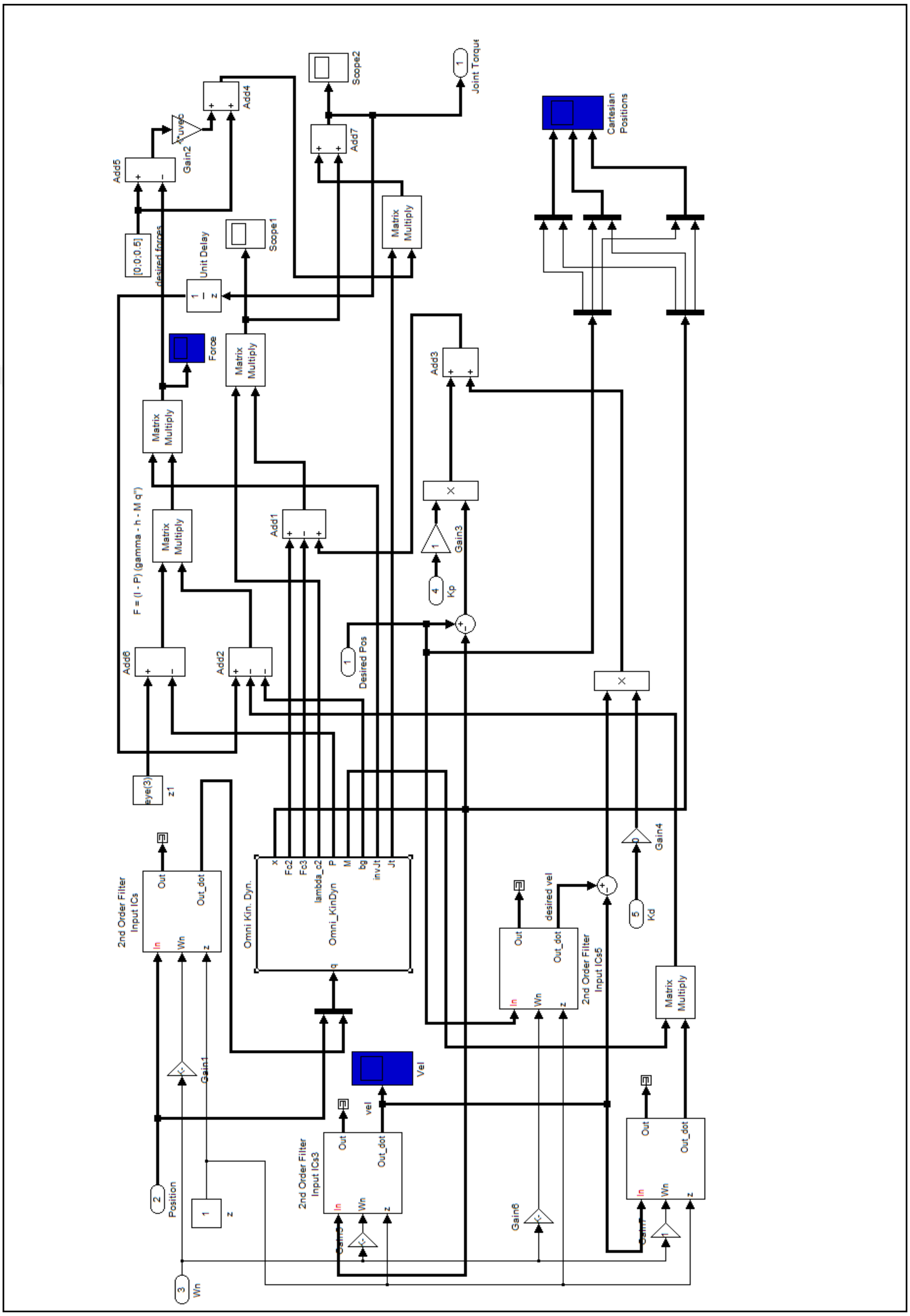


Figure App. 2 4 : Operational Space Controller



CURRICULUM VITAE

Name Surname : Seda Korkmaz

Birth Place / Year : Bursa, 1989

Languages : English

Bachelor of Science : Kocaeli University

Master of Science : Bahçeşehir University

Name of Institute : Graduate School of Natural and Applied Sciences

Name of Program : Mechatronics Engineering

Publications :

S. Korkmaz, B. Gür, Robot Etkileşim Kontrolünün İzdüşüm Operatörü Yöntemi ile Benzetimi, Türkiye Robotbilim Konferansı, 2015, İstanbul

S. Korkmaz, Mekanum Tekerlekli Gezgin Robotun Kinematik Analizi ve Yörünge Planlaması, Otomatik Kontrol Türk Milli Komitesi Ulusal Toplantısı, 2015, Denizli

Work Experience : 2012 -
Research Assistant
Bahçeşehir University, Mechatronics Engineering