Incorporation of the Articulated-Body Equations into a Model-Based Sliding-Mode Controller for the Reduction of Dynamic Coupling Effect in Underwater-Manipulator Systems

by

Serdar Soylu
B. Eng., Dokuz Eylul University, 2002

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Mechanical Engineering.

Co-supervisors: Dr. Bradley J. Buckham, and Dr. Ronald P. Podhorodeski

Abstract

A control scheme is presented for reducing dynamic coupling between an underwater robotic vehicle (URV) and a manipulator. During task execution the torques commanded at the manipulator joints lead to reactions at the junction point of the manipulator and vehicle. These reactions disturb the vehicle position and orientation and are the source of the vehicle-manipulator coupling. In underwater robotic vehicle-manipulator (URVM) applications, the URV serves as a base while the manipulator performs a required task. Therefore, it is necessary to hold the URV as stationary as possible. In the current work, URV thrusters are used to compensate for the dynamic coupling forces. SlotineŠs sliding mode control approach is used to reduce the dynamic coupling present in URVM systems. The articulated body (AB) algorithm is used both for the time-domain simulation of the system and for the dynamic equations within the model-based sliding-mode controller. Finally, the results of time-domain numerical simulation of the proposed control scheme on a URVM system are presented.

Examiners:

Dr. B. J. Buckham, Co-Supervisor (Department of Mechanical Engineering)

Dr. R. P. Podhorodeski, Co-Supervisor (Department of Mechanical Engineering)

Dr. Z. Dong (Department of Mechanical Engineering)

D. M. Lokhurst, M.A.Sc., External Examiner (Tactex Controls, Inc.)

Table of Contents

Abstract ii

Table of Contents iv

List of Figures x

List of Tables xiv

Acknowledgements xv

1 Introduction 1

1.1 Underwater Robotic Vehicles 1

1.2 URV-Manipulator Systems 2

1.3 Problem Description 3

1.4 Previous Work 4

1.4.1 Time-Domain URVM Dynamic Simulations 4

1.4.2 URV Dynamic Literature Survey 5

1.4.3 Thruster Dynamics and Control 6

1.4.4 Manipulator Dynamics Literature Survey 9

1.4.5 URVM Control Literature Survey 13

1.5 Objective of the Work 17
1.6 Thesis Contributions 18
1.7 Thesis Outline 19

2 Kinematics **21**

2.1 Overview 21
2.2 Vehicle Kinematics 21
2.2.1 Position and Velocity State Vectors 21
2.2.2 Linear Velocity Transformations 22
2.2.3 Angular Velocity Transformation 24
2.2.4 Kinematic Equations 26
2.3 Manipulator Kinematics in the URVM Serial Chain 28
2.3.1 Overview 28
2.3.2 Link Transformations 29
2.3.3 Extension of the Aforementioned Equations to the Kinematics of Serial-Chain Manipulators 30
2.3.4 Kinematic Equations for Serial-Chain Manipulators 31

3 Articulated-Body Equations **34**

3.1 Overview 34
3.2 Spatial Kinematics of Manipulator Links 34
3.3 Spatial Dynamic Equations for a Link 38
3.3.1 Link Translational Dynamics 38
3.3.2 Link Rotational Dynamics 39
3.3.3 Spatial Force-Balance Equation 41
3.4 Derivation of Articulated Body Inertias 43

3.5 Gravity 48
3.6 Mobile Base (URV) 49
3.7 Extension of the Articulated Body Algorithm to Underwater Vehicle Systems with Dual-Manipulator 51
3.8 Recovery of the Dynamic Coupling Effect 51
3.9 Summary of Solution of the Forward Dynamic Problem of a Mobile-Base Manipulator System 53

4 Forces Acting on an Underwater-Manipulator System **55**
4.1 Overview 55
4.2 Dominant Hydrodynamic Effects 56
4.2.1 Assumptions 56
4.2.2 Added Mass 57
4.2.3 Drag and Lift Forces 60
4.2.4 Buoyancy and Fluid Acceleration Forces 63
4.2.5 Incorporation of the Hydrodynamic Forces into the Articulated-Body Equations 64
4.3 Thruster Dynamics 66
4.3.1 Assumed Model 66
4.3.2 Motor Modelling 66
4.3.3 Hydrodynamic Propeller Loads 68
4.3.4 Fluid Modelling 72
4.3.5 Thruster Simulations 72
4.3.6 Thruster Mapping 75

4.3.7 Incorporation of the Thruster Dynamics in the Articulated-Body Equations 77

5 Complete Articulated-Body Algorithm for the Time-Domain Simulation of URVM Systems and Case Studies 78

5.1 Overview 78

5.2 Articulated-Body Algorithm for the Time-Domain Simulation of URVM Systems 79

5.2.1 Overview 79

5.2.2 Forward Kinematics: 79

5.2.3 Backward Dynamics: 82

5.2.4 Forward Accelerations: 85

5.3 Simulation Studies for the Demonstration of the Dynamic Coupling . 86

5.3.1 System Description and Frame Attachment 87

5.3.2 Fixed-Base Case (The Ideal Scenario) 88

5.3.3 Mobile-Base Case (The Worst Scenario) 91

6 Control of the URVM System 100

6.1 Overview 100

6.2 Sliding Mode Control Based on Articulated-Body Formulation 101

6.2.1 Linear Control Technique Disadvantages 101

6.2.2 Advantages of Incorporating the Articulated-Body Equations in the Controller 102

6.2.3 Preliminaries 103

6.2.4 Sliding-Mode Control Design 107

6.3 Thruster Controller 116

6.4 Reduction of the Dynamic Coupling in URVM Systems 121
6.5 Comparative Study Between Newton-Euler Formulation and Articulated-Body Algorithm 139

7 Conclusions and Recommendations for Future Work 143
7.1 Overview 143
7.2 Conclusions 143
7.3 Consideration of the Proposed Control Scheme on Real World Applications 145
7.4 Recommendations for Future Work 147
7.4.1 Overview 147
7.4.2 Exploitation of Kinematic Redundancy in URVM Systems . . 147
7.4.3 Centralized Multi-Input Multi-Output Controller Design . . . 149

References 151

A Kinematic Equations 156

B Cross Products of Vectors Using Skew- Symmetric Form 162

C Derivative of a Rotation Matrix 163

D Derivation of Added Mass Coefficients for the URV 164
D.1 Added Mass Matrix 164
D.2 Derivation of Added Mass Coefficients for Translational Motion . . . 165
D.3 Derivation of Added Mass Coefficients for Rotational Motion 167

E Simulation Parameters 168

E.1 URV Simulation Parameters: 168
E.2 Manipulator Simulation Parameters 169
E.3 Thruster Simulation Parameters 169
E.4 Sliding Mode Control Parameters 170
E.5 Thruster Control Parameters 170
E.6 Environmental Parameters 170

F An Example of Centralized Model-Based Sliding-Mode Controller Design **171**

List of Figures

1.1 A Typical URVM System (Seaeye Marine Limited: http://seaeye.com) 2

1.2 Block Diagram of Fixed-Feedforward Thruster Control 8

1.3 Block Diagram of Feedback Velocity Control and Model-Based Velocity Control Thruster Control 9

2.1 Frames and URV's Motion 22

2.2 Euler Angles 23

3.1 Points on the Manipulator Links 36

3.2 Link in a Serial-Manipulator for the Force-Balance Equation 39

3.3 Link in a Serial-Manipulator for the Moment-Balance Equation . . . 40

3.4 Spatial Quantities for a Link 42

3.5 Multi-Body System 43

3.6 Articulated Bodies in a Serial-Chain 44

3.7 URVM System 50

3.8 Reaction Forces That Causes the Dynamic Coupling Effect 52

4.1 URV is Divided into Strips. 62

4.2 A Cylinder is Composed of a Series of Circular Strips 63

4.3 Airfoil for a Propeller Section 69
4.4 Four Quadrant Mapping 70
4.5 Time History of Propeller Rate 74
4.6 Time History of Thrust 75
4.7 Time History of Incoming Fluid Velocity 76
4.8 Time History of Thrust and Incoming Fluid Velocity 77

5.1 Frame Attachments for the URVM system 88
5.2 End-Effector Path 90
5.3 a) Time History of Joint Position for Joint 1 b) Time History of Joint Velocity for Joint 1 94
5.4 a) Force Acting Along Xo Direction Caused by Dynamic Coupling b) Force Acting Along Yo Direction Caused by Dynamic Coupling . . . 95
5.5 Moment Acting Around Zo Direction at the Center of Mass of the URV Caused by Dynamic Coupling 96
5.6 a)Vehicle Position and Orientation When Released b) Vehicle Translational and Rotational Velocities When Released 97
5.7 a) Time History of Joint 1 Position When the URV Released b) Time History of Joint 1 Velocity When the URV Released 98
5.8 End-Effector Path with the URV Floating 99

6.1 $n-1$ Filters Between s and $\breve{x}$ 105
6.2 The Sliding Surface for $n = 2$ Represents a Line. The State Variable Hits the Desired Point with a Time-Constant. 106

6.3 If $\mathbf{x}(t = 0) \neq \mathbf{x}_d(t = 0)$, the Controlled Variable State Reaches to the Sliding Surface in a Finite Time and Then Hits the Desired Point Exponentially. 107
6.4 URV's Controlled State Variable on the State Space 109
6.5 Boundary Layers 117
6.6 Thruster Control Scheme 119
6.7 Thrust History 120
6.8 Time History of the Controlled Propeller Rate 121
6.9 Time History of the Incoming Fluid Velocity 122
6.10 Thruster Configuration of the URV 123
6.11 Simulation Scheme 124
6.12 End Effector Path on X-Y Plane 125
6.13 a) Vehicle Translational and Rotational Motion History without Control Activity b) Vehicle Translational and Rotational Motion History with Control Activity 126
6.14 a) Boundary Layer versus s b) s Graph with a Smaller Scale 128
6.15 Resulting Thruster Force and Moments Acting at the Center of Mass of the URV 129
6.16 a) Manipulator Joint Displacement History b) Manipulator Joint Velocity History 130
6.17 a) URV Translational Velocity History b) URV Rotational Velocity History 131
6.18 a) Incoming Fluid Velocity b) Propeller Shaft Velocity 132
6.19 a) Force Acting Along Xo Direction Caused by Dynamic Coupling b) Compensation Force Acting Along Xo Direction of the URV by Thrusters 133

6.20 a) Difference Between the Disturbance Force and the Control Force Along Xo Axis of the URV b) Same Graph with a Different Scale . . 134
6.21 a) Force Acting Along Yo Direction Caused by Dynamic Coupling b) Compensation Force Acting Along Yo Direction of the URV by Thrusters 135
6.22 a) Difference Between the Disturbance Force and the Control Force Along Yo Axis of the URV b) Same Graph with a Different Scale . . 136
6.23 a) Moment Acting Around Zo Direction Caused by Dynamic Coupling b) Compensation Moment Acting Around Zo Direction by Thrusters 137
6.24 a) Difference Between the Disturbance Moment and the Control Moment Around Zo Axis of the URV b) Same Graph with a Different Scale 138
6.25 End-Effector Absolute Errors When Fluid Velocity 0.1 m/sec Along X and Y Directions of the Inertial Frame 141
6.26 URV Positionand Orientation Errors When Fluid Velocity 1 m/sec Along X and Y Directions of the Inertial Frame 142

A.1 Position Relative to a Fixed or Moving Reference Frame 157

D.1 URV is Considered To Be Composed of a Series of Rectangular Strips 166
D.2 2D Added Mass Coefficients for a Rectangular Cross-Section 167

List of Tables

5.1 Denavit-Hartenberg Parameters for the URVM system 89

Acknowledgements

First I would like to express my deepest gratitude to my supervisors Dr. Bradley J. Buckham and Dr. Ronald P. Podhorodeski for their excellent guidance and support during the completion of this thesis, and for giving me the opportunity to undertake this work. I would also like to extend thanks to my friends and fellow RAM members Alp Zibil and Flavio Firmani for their helpful discussions regarding my studies, and for their invaluable computer expertise. Finally, I would like thank my family and friends, who have encouraged me greatly throughout the course of my studies.

Anneme bana vermis oldugu karsiliksiz sevgi icin. Seni Seviyorum Anne...

Babama bana vermis oldugu iyi insan olma ideali icin...

Kizkardeslerim Tuba, Zuleyha ve yegenim Olcayto'ya kendileri oldugu icin...

Amanda'ya bana gostermis oldugu sevgisi ve anlayisi icin...

Chapter 1

Introduction

1.1 Underwater Robotic Vehicles

The oceans of the earth cover about two-thirds of the world's surfaces. However, very limited attention has been given to ocean exploration compared to the attention given to terrestrial and atmospheric matters. The oceans hold enormous natural resources of gas, oil, and mineral deposits, that have not been explored yet. However, the majority of the earth's vast ocean resources are not utilized. If these natural resources can be utilized, it should positively affect human society. In addition, the oceans also hold tremendous information about the origin of life and the distribution of species on earth. Underwater robotic vehicles (**URVs**) are the primary means to explore the deep ocean's scientific potential.

URVs equipped with robotic manipulators have an important role in many deep-water missions. These systems are deployed to subsea work sites where human beings can not survive due to the enormous pressure existing in the underwater environment. In these applications, URVs are used as mobile platforms that deliver robotic tools

to a subsea work site. Along with this function, the URV serves as a base while the manipulator performs a required task.

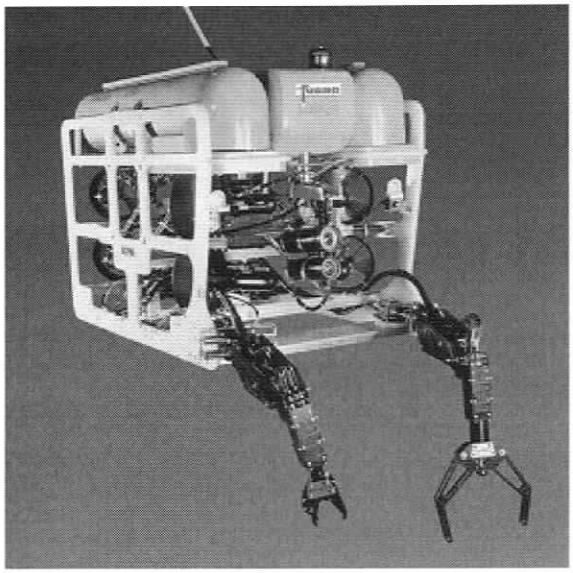


Figure 1.1: A Typical URVM System (Seaeye Marine Limited: http://seaeye.com)

To facilitate the human presence in the URV applications, URV systems are tethered. Information associated with the task between the URV and the human operator flows through the tethers. A typical URV system is illustrated in Figure 1.1.

1.2 URV-Manipulator Systems

In many URV applications, the teleoperated master-slave configuration is used to execute underwater tasks. In this configuration, the larger slave-arm duplicates the movement of a smaller master-arm driven by a human operator on the surface vessel [1]. In a typical underwater robotic vehicle-manipulator (**URVM**) application, the URV's position information along with slave-arm joint positions are provided to the

pilot of the system. The pilot uses two-dimensional video images to judge where he is in a work site so that he can operate the master-arm based on this visual information.

However, during a typical deployment, the pilot encounters enormous difficulties. Firstly, since the two dimensional image of the work-site does not provide a depth cue, the pilot may misinterpret the visual information and in turn possibly fail to accomplish the task. Moreover, as conventional thrusters rely on momentum transfer to the surrounding fluid, their response time is slower than that of the manipulator joints and this makes it difficult to synchronize the thrusters and manipulator commands. Furthermore, when the movement is replicated by the larger-slave arm, the inertial and hydrodynamic drag associated with the manipulator links create reactions at the manipulator-URV junction. The reaction loads act as disturbances to the URV position which then in turn disturbs the placement of the end effector.

The aforementioned difficulties associated with the URV-manipulator scheme reduce the efficiency of the human pilot. In addition, these difficulties limit the range of sea states in which the URVM can operate efficiently. A control system that provides solutions to these difficulties is needed. In the current work, the problem of the reduction of the dynamic interactions between the URV and the manipulator is addressed within the development of a supervisory control system.

1.3 Problem Description

During URVM operation, it is usually necessary to operate the manipulator in a confined space. In this type of task, the URV must provide a fixed-base for the manipulator. However, as mentioned earlier, when the movement of the master-arm is duplicated by the submerged slave-arm, reaction forces at the junction point

between the URV and manipulator occur. The reaction loads perturb the URV attitude, heading, and position, and these changes in the URV's state adversely affect the end-effector position and orientation. The pilot must compensate the manipulator dynamical loads with thruster inputs to place the end-effector in a desired position and orientation. This dynamic interaction between URVM subsystems is called dynamic coupling and the dynamic coupling effect varies in magnitude depending on the manipulator trajectory specifications. In order to obtain better system performance, the reduction of the dynamic coupling becomes a significant issue [2]. The most direct solution is to hold the URV fixed during manipulator operation by intelligently commanding the URV thrusters via an automatic controller. This greatly simplifies the task of the human pilot.

1.4 Previous Work

1.4.1 Time-Domain URVM Dynamic Simulations

Any attempt to solve the problem described in Section 1.3 requires a dynamics model of the system. The dynamic model of the system is the basis for time-domain dynamic simulations of the URVM systems. The time-domain simulation schemes provide a test-tool to validate several design concepts as well as the control laws to be implemented. This precludes the high costs of the URVM prototyping process since erroneous controller designs are never constructed and deployed. The URVM simulations are also used to develop a better understanding of the system dynamics, which can be used to improve design concepts.

However, obtaining a dynamic model of the system is very challenging in the sense

that the system dynamics are highly nonlinear and vary greatly with changes in arm and URV orientation [3], [4], [5], [6]. The non-linearity in the system is largely caused by the hydrodynamic forces and moments, which dominate the dynamics of the URVM system. Furthermore, the hydrodynamic parameters are poorly known. Also, a variety of unmeasurable effects inherent in underwater environments exists. Moreover, the URV and the manipulator have their own unique set of dynamic properties.

1.4.2 URV Dynamic Literature Survey

The equations governing the motion of URVs in six degrees of freedom can be derived from the Newton-Euler formulation in which the URV is treated as a single rigid body with constant inertia and hydrodynamic coefficients. The Newton-Euler formulation relates the resultant forces and moments to the time derivative of the linear and angular momentum. One of the notable works in the implementation of the Newton-Euler formulation to the URVs is done by Fossen [7]. Fossen derived the equations of motion for URVs based on the Newton-Euler formulation in a matrix form. He refers the quantities to a body-fixed frame using the body-fixed linear and angular velocities. In [7], the forces applied to the URVs include: added mass forces, weight and buoyancy, drag forces, wave forces, and current forces. All of these forces can be analytically expressed under certain assumptions. The assumptions that are made to derive the hydrodynamic forces include: the fluid has constant and uniform density and is unbounded, incompressible, inviscid, irrotational, and of infinite extent as mentioned in [7] and [8]. A representative work of the incorporation of the aforementioned hydrodynamic forces into the Newton-Euler formulation is demonstrated

in [7]. The detailed discussions about these hydrodynamic forces can be also be found in [8] and [9].

The derivation of the dynamic equations of motion on the URVs can also be made using Lagrangian dynamics. Lagrangian dynamics leads to an easier derivation of the equations of motion since it is based on two scalar energy equations namely the rigid-body kinetic energy and potential energy. The application of the Lagrangian approach to the marine dynamics can be found in [10].

There are many works on the dynamics of Autonomous Underwater Vehicles (AUVs) [11], [12], [13]. In [11], the AUV is broken up into different components. Since AUVs are streamlined, the hydrodynamic characteristics of the AUV are found using the characteristic dimensions of the each component of the AUV such as port/starboard, upper/lower fore/aft planes, hull and instrumentation bulges. The hydrodynamic characteristics using the characteristic dimensions can not be determined for URVs, as they are not streamlined. According to [11], the overall effects are determined by superimposing every individual dynamic effect on the AUV. It is shown that the resulting dynamics is successful in capturing the non-linear nature of the AUV.

In the field of URV dynamics, the treatment of the thruster dynamics is paramount. Thruster dynamics models are most often created on their own and then are inserted into URV dynamics model.

1.4.3 Thruster Dynamics and Control

It has been established that thruster dynamics are a significant effect within a URV dynamics model. Yoerger *et al.* [14] developed a dynamic thruster model for use

in underwater vehicles. Although the model accurately captures the time delay between motor control signals and developed thrust, it does not account for all possible combinations of propeller rotational rate and flow direction through the thruster's shroud. Van Lammaren *et al.* [15] came up with a model that characterizes all of the possible combinations through a four-quadrant mapping of the apparent angle of approach of water particles on the propeller blades. Healey *et al.* [16] combined an armature controlled motor model, a theoretical propeller mapping using airfoil theory, and a fluid momentum model applied within the finite volume of the shrouded region. Healey *et al.* [16] only considered the axial component of the fluid flow and used sinusoidal lift/drag curves. Bachmayer *et al.* [17] incorporated the effects of the rotational fluid motion on thruster response and further proposed a method to experimentally determine non-sinusoidal lift/drag curves. In the present work, Healey *et al.*'s [16] approach is followed.

In Yoerger *et al.*'s [14] paper , three different thruster control methods on a torque-controlled thruster were proposed to compensate the lag between commanded and developed thrust and limit cycle. In the first method, a lead compensator has been used. However, it is concluded that changes in the operation conditions result in the reduction of the performance of the controller. The second method is a "pole" cancellation method. In this method, the effect of thruster dynamics are compensated by canceling the apparent lag pole with a corresponding zero. The downside of this method is its dependence on good knowledge of the thruster dynamics properties. The adaptive sliding-mode control methodology was the third approach that applied to the control problem of the thrusters in [14]. In the adaptive sliding-mode methodology, the sliding-mode controller made adaptive to parametric uncertainty by coupling it to an on-line parameter estimator. In [14], it is shown on a hybrid

simulation setup that the adaptive sliding-mode controller is effective over the entire operating range and is capable of compensating the uncertainties or degradation of the thrusters.

Another three different thruster controllers on a current-controller thruster were presented in [18]. Comparative experiments with the three different thruster control methodologies were reported in [18]. The first control strategy was called fixed-feedforward thrust control. This method is the most commonly used thrust-control algorithm on today's underwater vehicles. In this method, the commanded thrust is fed into the controller in a feedforward manner as illustrated in Figure 1.2. This method is also called open-loop proportional control. The second method is called

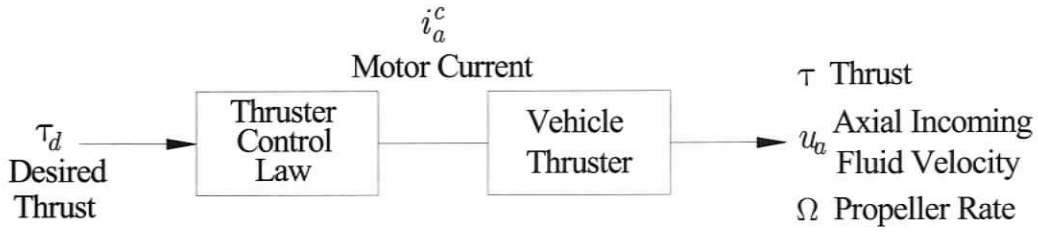


Figure 1.2: Block Diagram of Fixed-Feedforward Thruster Control

feedback velocity control. In the feedback velocity control, the measured state of propeller rotational rate is fed back to the controller as illustrated in Figure 1.3. The controller uses the thruster dynamic model proposed in [14]. The third method is called model-based velocity control. The block diagram of the model-based velocity control is the same as that of the feedback velocity control. However, in the model-based velocity control, the work of Healey *et al.* [16] is followed to model the dynamics of thrusters. According to [16], two state variables are needed to define the thruster's state, *i.e.*, the axial incoming fluid velocity and the propeller rotational rate. In

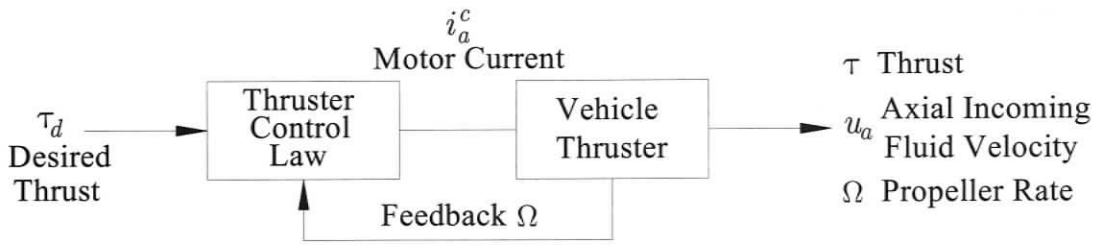


Figure 1.3: Block Diagram of Feedback Velocity Control and Model-Based Velocity Control Thruster Control

the model-based velocity control method, the propeller rotational rate is directly measured through sensors and is fed into the controller. However, since it is difficult or impossible to measure the axial incoming fluid velocity, an ad hoc open-loop estimator is used to estimate the axial velocity. The model-based velocity control method is chosen in this thesis considering that the controller is consistent with the modelling approach of Healey *et al.* [16]. Moreover, in [14], it has been shown that the model-based velocity control method offers better thrust performance over a wide range of operating conditions in comparison to the other two methods in [14].

Note that for the aforementioned thruster controller, the desired thrust is obtained from a high-level vehicle position control.

1.4.4 Manipulator Dynamics Literature Survey

In manipulator dynamics, there are two main problems: the inverse dynamics problem and the forward dynamics problem. While the inverse dynamics problem involves finding required joint torques for a given set of desired joint positions, velocities, and accelerations, the forward dynamics problem deals with obtaining joint accelerations for a given set of joint positions, velocities, and applied torques.

Inverse Dynamics Problem

Inverse dynamics provides the values of required generalized forces necessary to achieve a desired trajectory. These values are fed into controllers, which drive the manipulator through a task. Moreover, the inverse dynamics can also be used to provide some feedforward knowledge to controllers depending on the controller methodology employed. In most of the inverse dynamic applications, the inverse dynamics must be computed on-line [19]. This leads to the need for a computationally efficient computation scheme for the inverse dynamics problem.

There are two standard approaches to formulating the robot dynamics, namely Lagrangian dynamic formulation and the Newton-Euler dynamic formulation. The Lagrangian dynamic formulation is an energy-based approach. The classic implementations of the Lagrangian dynamic formulation yield an algorithm whose computational complexity is $O(N^4)$[1][20]. Hollerbach [21] derived a recursive manipulator dynamic formulation based on the Lagrangian formulation with the computational complexity $O(N)$. However, Hollerbach [21] also found that his recursive Lagrangian formulation is not as computationally efficient as its Newton-Euler-based counterpart in terms of the number of multiplications and additions/subtractions. [21]

The first $O(N)$ algorithm developed for the inverse dynamics problem was based on the Newton-Euler formulation. Stepanenko and Vukobratovic [22] derived a recursive Newton-Euler method for human limb dynamics. Afterwards, Orin *et al.* [23] made an improvement over the work of [22] by formulating the forces and moments in local frames. Luh *et al.* [19] developed a very efficient recursive Newton-Euler al-

[1] N represents the number of degree of freedom. $O(N^n)$ means that the computatinal complexity increases by the order of n with degree of freedom.

gorithm by formulating the quantities in local frames. Further improvements on the Newton-Euler Algorithm have been made in efficiency over the years. For instance, Balafoutis *et al.* [24] attained faster algorithms by using orthogonal second-order Cartesian tensors to formulate the Newton-Euler dynamic equations. McMillan [25] formulated the recursive Newton-Euler equations using spatial matrices and obtained a computational efficiency that is comparable to the work of Balafoutis *et al.* [24].

Forward Dynamics Problem

The forward dynamics problem is primarily encountered in the simulation of a robotic mechanism. For a simulation purposes, as opposed to real-time control applications, it is not required for the forward dynamics to meet high-speed requirements. However, the computational efficiency is still a significant issue since it is desired to minimize the computational cost of the simulation.

Since the inverse dynamics algorithm is computationally efficient, it is used as a basis for the forward dynamics problem of robotic manipulators. In the early 1980s, Walker and Orin [26] proposed four different methods for the computation of the forward dynamics problem. These methods were based on the recursive Newton-Euler method proposed in [19]. In [26], *Method 1* rested on the physical interpretation of the robot inertia matrix. That is, in a system where all of the velocities and acceleration-independent forces are zero, the i^{th} column of the inertia matrix corresponds to the generalized force vector that causes a unit acceleration of joint i. Therefore, *Method 1* states that the application of the recursive Newton-Euler algorithm successively for each joint in such conditions forms the inertia matrix for open-chain robot mechanisms. Having obtained the inertia matrix and knowing the forces acting through the manipulator, the forward dynamics problem reduces to solving a linear system of

equations. The resulting algorithm has a $O(N^3)$ computational complexity.

Method 2 was the same as *Method 1* except that the symmetric nature of the manipulator inertia matrix was exploited. In *Method 2* only the diagonal and the bottom half of the off-diagonal terms are computed, and thus better efficiency over *Method 1* is obtained. The resulting algorithm also has a $O(N^3)$ computational complexity.

Method 3, later named the Composite-Rigid-Body-Algorithm by Featherstone [27], involves the computation of the inertia properties of a series of composite rigid bodies within the open-chain robotic mechanism from the end-effector to the base. This algorithm works based on *Method 1.* In *Method 1,* a unit acceleration is given to the zero-velocity robotic system at joint i, and in turn the whole robotic system from link i to n behaves like a single composite rigid body while the rest of the system stays in static equilibrium. Therefore, a composite rigid body accelerates around joint i and the forces, which correspond to the i^{th} column of the inertia matrix, are calculated using the inertia properties of the composite-rigid body. The resulting algorithm has a $O(N^3)$ computational complexity. *Method 4* uses an iterative procedure to solve for the joint accelerations based on the conjugate gradient method.

Featherstone [27] first developed the Articulated-Body Algorithm. The advantage of the articulated-body algorithm lies in its $O(N)$ computational complexity meaning the amount of computation grows linearly with the number of joint degrees of freedom. Further improvements over Featherstone's articulated-body algorithm were made by Brandl *et al.* [28] by using efficient transformations and link coordinates. The resulting Articulated-Body Algorithm was comparable to the composite-rigid body method for $N = 6$. McMillan and Orin [29] further improved the work of Brandl *et al.* [28] and reduced the amount of computation by 15%. A comparison between the

composite-rigid body method and the articulated-body method was made by [25]. In [25], it was shown that the articulated-body requires less computation than that of the composite-rigid body method for $N > 3$.

Finally, the computational complexity of the Articulated-Body Algorithm is $O(N)$ meaning that the computational complexity grows linearly with the number of degree of freedom. However, the Composite-Rigid Algorithm has a cubic complexity *i.e.*, $O(N^3)$. In the current work, the articulated-body algorithm is chosen considering that it is the most computationally efficient simulation algorithm for serial-chain structures with the number of degree of freedom more than 3.

1.4.5 URVM Control Literature Survey

There are two conventional methods to hold the URV reasonably stationary during manipulator operation. Additional manipulators or an attachment system such as suction feet are often mounted on the front of the URV and are used to latch onto the surroundings or the workpiece itself. This requires additional mechanical units and the coordination of these multiple devices. Using a controller to keep the URV stationary via the URV thrusters can eliminate the need to implement additional mechanisms. A controller thus not only reduces the complexity of the system structure and preserves valuable payload space on the vehicle but also eliminates time required for the attachment units to latch onto the surrounding objects. In addition, an automatic controller operating over the URV thrusters can perform in the middle of the water column.

In the literature several control methods have been applied to control URV motion. Kazerooni and Sheridan [30] developed a control system using pole placement and

observer method. Since they used an observer, their model can be used when the full state feedback is not available. However, the robustness of the control system with respect to parameter uncertainties cannot be guaranteed.

Yoerger and Slotine [31] applied Slotine's [32] sliding-mode control methodology to the control problem of underwater vehicles. The control problem in [31] is to make the vehicle states follow desired trajectory values with a prescribed dynamic characteristic in the presence of dynamic parameter uncertainties and disturbances. The strength of their approach was that it did not require perfect dynamic parameters knowledge. If the bounds are known on the dynamics parameters, then the satisfaction of the performance criteria is guaranteed. In [31], a series of single-input single-output controllers were used, and the robustness of the proposed control scheme to uncertainties in the dynamic model of a vehicle was demonstrated. High control activity or "chattering" is eliminated by introducing a boundary layer into the control technique. The sliding-mode technique has been successively implemented on the Jason vehicle operated by Woods Hole Oceanographic Institution [33].

Cristi *et al.* [34] proposed a control technique that combines the robustness property of the sliding-mode controller with the adaptivity of a adaptive controller. In [34], the sliding surface is based on system state and state estimators instead of on output error.

The adaptive approach was also employed by Goheen and Jeffery [35]. In [35], a method that does not require knowing precise knowledge of the dynamics of an underwater vehicle is proposed. It is highlighted that system dynamics identification requires expensive full-scale testing and still involves uncertainty. Goheen and Jeffery [35] linearized the dynamics of an underwater vehicle around typical operating points and the adaptation law is constructed from the linearized model. Yuh [36]

demonstrated the application of neural networks to an underwater robotic control system. In neural net application, the goal is to find a proper way to train the controller such that the controller generates control signals that accomplish the desired task. Simulation studies were carried out in [36] and it was concluded that a neural net controller provides robust control technique in case the exact dynamic knowledge is not available. However, the questions about stability and convergence remain unanswered.

Multivariable sliding-mode control is developed for the control problem of underwater vehicles by Healey and Lienard [37]. In [37] a set of separate designs for the steering, diving and speed control is used for the control of unmanned underwater vehicles. Among the various approaches in the literature, the H_∞ approach is used by Conte and Serrani [38]. In [38], it is shown that H_∞ approach provides a robust control technique to the control of underwater vehicles.

The control of the URVM systems can involve the control of both components; the URV and the manipulator. In the control problem of the URVM system, the goal is to make the URV and the manipulator state values follow desired trajectory values. A discrete adaptive control strategy for coordinated control of URVMs is presented in [39]. It is shown that a centralized controller gives better performance over two separate controllers applied separately for the vehicle control and the manipulator control. In [4], a closed-form symbolic expression URVMs motion equations are derived using Newton-Euler dynamic equations. The feedback linearization technique was then applied using these expressions to the tracking problem of URVMs. However, the feedback linearization technique requires perfect dynamic knowledge, which is not attainable in the underwater environment. This problem was not addressed in [4]. A similar model-based control approach is also presented in [40]. In [40], the

closed-form symbolic model were obtained using Kane's equations [5].

An adaptive control law based on a sliding mode approach for the tracking problem of URVMs is developed in [41]. The control law is free of representation singularities of the orientation due to the use of quaternions in the vehicle attitude control. The numerical simulation of an URVM system has been carried out and the results have been presented.

In URVM systems, the vehicle and the manipulator have their own dynamic characteristics *e.g.*, their response time to control inputs differ. In [42], the singular perturbation theory has been used to model the difference between the two time operation scales between the vehicle and manipulator. The computed torque control method and a robust nonlinear control technique have been demonstrated using the singular perturbation model.

The problem of tracking a desired motion trajectory for an underwater vehicle-manipulator system without using direct velocity feedback has been addressed in [43]. In the direct velocity feedback, the problem is that numerical differentiation of noisy position/ orientation measurements lead to chattering of the control inputs and in turn cause high-energy consumption at the actuators. This creates the need for a control technique that does not need direct velocity feedback. In [43], this problem were solved by using a velocity observer. The numerical simulation of an URVM system has been carried out, and it is demonstrated using the controller-observer strategy leads to the reduction of chattering and in turn lowers the energy consumption.

Dunnigan and Russell [44] address the dynamic coupling problem for URVM systems, and propose the sliding-mode approach with a feedforward compensation term to minimize the vehicle-manipulator dynamic interactions. It is also shown that

such a feedforward term significantly improves the end-effector trajectory tracking performance of the system. Ryu *et al.* [3] considers a force/torque sensor mounted at the base of the manipulator in order to detect & compensate the dynamic coupling effect. It is also shown by Ryu *et al.* [3] that in the case that such a sensor is not available, a disturbance observer is feasible. However, the practical implementation of such algorithms is not trivial [45].

The URVM dynamics are dominated by hydrodynamic terms and it is difficult to accurately measure or estimate the hydrodynamic coefficients. This calls for a robust controller: one that is insensitive to inaccuracies in the dynamic model of the URVM. As mentioned before, the sliding mode approach is an effective technique for the control of URVM systems [31], [33] and [44]. As such, in the present work the sliding-mode approach is chosen.

1.5 Objective of the Work

The primary objective of this work is to propose a control law that reduces the dynamic coupling effect between the URV and the manipulator. As opposed to the vast majority of URVM systems in which the human pilot runs thrusters and arms independently, it is proposed that the arm and the thrusters be coordinated such that the URV thrusters are used to compensate the dynamic coupling effect due to the manipulator motion. This requires employing a control strategy that intelligently commands the thrusters such that the disturbances caused by the dynamic coupling are compensated. In order to accomplish this goal, it is required to have a complete nonlinear time-domain simulation of the URVM that provides not only a test-tool for validation of the control laws to be implemented, but also a basis from which the

controller model is constructed.

1.6 Thesis Contributions

The contributions of this thesis are spread over the areas of dynamic simulation and the control of URVM systems. The existing articulated-body algorithm has been extended to include the thruster dynamics and has been applied for dynamic simulation of a URVM system. A generic URVM dynamic simulation program has been created using the articulated-body algorithm. Simulation results for a URVM system that has a manipulator with a $R \perp R \,//\, R$ configuration are presented for a planar task. The incorporation of the articulated-body algorithm into a model-based sliding-mode controller for the reduction of the dynamic coupling between the URV and the manipulator has been accomplished and simulations are used to demonstrate its effectiveness in the reduction of dynamic coupling effect.

The dynamics models that were previously used to solve the dynamic coupling reduction problem require inertial measurement units to accurately predict the dynamic coupling forces. In this work, it is shown that using the articulated-body algorithm in the model-based sliding-mode controller eliminates the need for the inertial measurement units to predict the dynamic coupling forces. In addition, the articulated-body equations directly capture the effect of the added mass inertia of the manipulator arm. Consequently, using the articulated-body form of the URV dynamics equations within a URVM system model allows for a more precise URV dynamics representation in the model-based controller, and in turn provides better trajectory control performance of the URV.

The inclusion of the articulated-body algorithm and its performance in reducing

the dynamic coupling effect have not been presented before in literature and its inclusion in the sliding-mode controller is considered to be a unique contribution of this thesis to the underwater-robotics field. This theoretical work culminates in a comparative study in Section 6.5. The comparative study shows that the use of articulated-body algorithm within the model-based controller improves the controller performance in accomplishing a coordinated task in comparison to the Newton-Euler based controllers. Consequently, this work improves the coordinated control of the URVM systems in response to disturbances.

1.7 Thesis Outline

The remainder of the thesis proceeds as follows: Chapter 2 describes the vehicle kinematics and manipulator kinematics. In Chapter 2, the state variables associated with the vehicle and the manipulator are defined. The frame attachment procedure is also explained in this chapter. Necessary transformation matrices from the earth frame to the URV's body-fixed frame, and the transformation matrices from link to link within the manipulator, are presented in this section. The velocity and acceleration equations governing the vehicle and the manipulator are also given in Chapter 2.

In Chapter 3, a spatial matrix form of the kinematic and dynamic equations for a single rigid link is presented. In this section, the derivation of the articulated-body equations are presented using the kinematic definitions of Chapter 2. The incorporation of a mobile base and gravity into the articulated-body equations is also demonstrated in Chapter 3. The dynamic coupling spatial force equations are also presented in this section.

In Chapter 4, the most significant hydrodynamic forces that are considered in this thesis are presented. In addition to the hydrodynamic forces, a non-linear thruster dynamic modelling technique is introduced. A thruster simulation is performed to compare actual thruster performance to that predicted by the modelling technique used.

In Chapter 5, the results obtained from Chapters 3 and 4 will be combined and a complete articulated-body algorithm for the time-domain simulation of URVM systems will be presented. The solution procedure will be also laid out step by step based on the derived equations from Chapters 3 and 4. Finally, the problem of dynamic coupling present in URVM systems will be demonstrated based on the simulation results obtained.

In Chapter 6, the model-based sliding-mode control approach for general nonlinear systems and model-based velocity control method for the thrusters are given. The incorporation of the articulated-body equations into the model-based sliding-mode controller is presented. A control scheme is presented for reducing the dynamic coupling between the manipulator and the vehicle. To demonstrate the effectiveness of the proposed control scheme, a time-domain numerical simulation is carried out and the results are presented. A comparative study between the Newton-Euler and the articulated-body based sliding-mode controllers are also carried out in this section.

Finally, in Chapter 7, the topic of the exploitation of the redundancy in URVM systems as well as the full motion control of URVM systems are brought up as a future research topic. Chapter 7 finishes with conclusions.

Chapter 2

Kinematics

2.1 Overview

In this chapter, the kinematic model of the URV and the manipulator are presented separately.

2.2 Vehicle Kinematics

2.2.1 Position and Velocity State Vectors

As shown in Figure 2.1, the Z axis of the earth-fixed inertial frame {E} is assumed in the gravity direction as is consistent with traditional marine mechanics. The URV is modeled as another manipulator link in the serial chain and numbered as 0. The body-fixed frame {0} is attached to the centre of mass of the URV as shown in Figure 2.1. The URV's absolute spatial velocity state vector in terms of the body-fixed frame is considered to be $\underline{\mathbf{v}}_0 = \begin{bmatrix} \boldsymbol{\omega}_0^T & \mathbf{v}_0^T \end{bmatrix}^T = \begin{bmatrix} p & q & r & u & v & w \end{bmatrix}^T$, and the position and orientation state vector of the URV in terms of the inertial frame is given by

$\underline{\mathbf{x}}_0 = \begin{bmatrix} \boldsymbol{\eta}_2^T & \boldsymbol{\eta}_1^T \end{bmatrix}^T = \begin{bmatrix} \phi & \theta & \psi & x & y & z \end{bmatrix}^T$. Note that the notation used is based on the SNAME [46] notation.

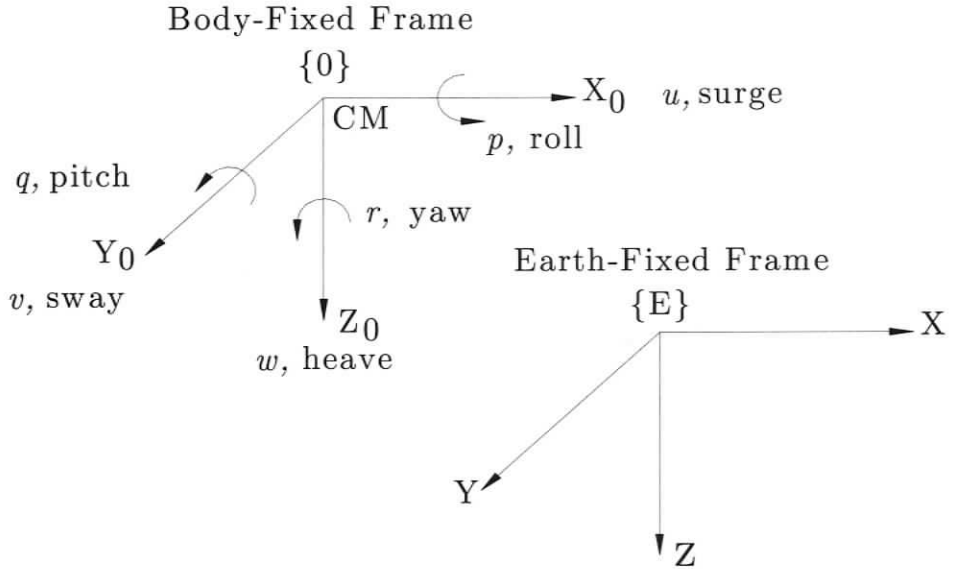


Figure 2.1: Frames and URV's Motion

In time-domain simulations, it is desired to know the evolution of the position and orientation of the vehicle, with respect to an inertial frame, in time. In Chapter 3, the evolution of the vehicle dynamics will be completed by using kinematic parameters specific to the body-fixed frame. In order to evolve the vehicle state in terms of the inertial frame coordinates, transformations between the body-fixed and the inertial frame must be defined.

2.2.2 Linear Velocity Transformations

The transformation matrix between the body-fixed frame and the earth-fixed frame can be obtained by the Euler sequence of rotations, *i.e.*, the first rotation ψ is about

the Z axis of the inertial frame (yaw), the second rotation θ is about the new Y$'$ axis (pitch), and finally, the last rotation ϕ is about the new X$''$ axis (roll), as illustrated in Figure 2.2.

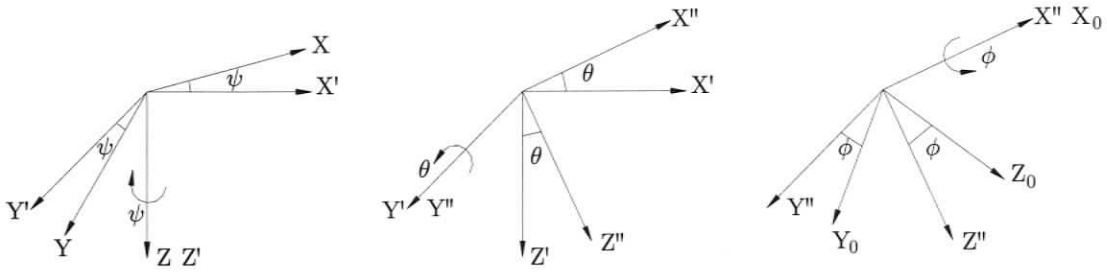


Figure 2.2: Euler Angles

The transformation matrices corresponding to each of the individual single-axis rotations are given by:

$$\mathbf{R}_{z,\psi} = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{R}_{y,\theta} = \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix}, \mathbf{R}_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \tag{2.1}$$

where $\mathbf{R}_{z,\psi}$ is the transformation matrix from intermediate frame $\{'\}$ to the inertial frame, $\mathbf{R}_{y,\theta}$ is from intermediate frame $\{''\}$ to intermediate frame $\{'\}$ and lastly, $\mathbf{R}_{x,\phi}$ is from the URV's body-fixed frame to intermediate frame $\{''\}$. The resulting transformation matrix, $\mathbf{T}_1$, is the transformation matrix from the URV's body-fixed frame to the inertial frame and can be found by:

$$\mathbf{T}_1 = \mathbf{R}_{z,\psi}\mathbf{R}_{y,\theta}\mathbf{R}_{x,\phi} \tag{2.2}$$

yielding

$$\mathbf{T}_1 = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi s\theta c\phi \\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & -c\psi s\phi + s\psi s\theta c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{2.3}$$

where

$$s \equiv \sin(\cdot), \quad c \equiv \cos(\cdot), \quad t \equiv \tan(\cdot)$$

The linear velocity of the URV in terms of the inertial frame is given by:

$$\dot{\boldsymbol{\eta}}_1 = \mathbf{T}_1 \mathbf{v}_0 \tag{2.4}$$

2.2.3 Angular Velocity Transformation

The relation between body-fixed angular velocity and earth-fixed angular velocity is

$$\dot{\boldsymbol{\eta}}_2 = \mathbf{T}_2 \boldsymbol{\omega}_0 \tag{2.5}$$

In order to find the transformation matrix $\mathbf{T}_2$ an infinitesimal rotation can be considered. An infinitesimal rotation can be written as the vector sum of a series of infinitesimal Euler angles [47]. Considering Figure 2.2, an infinitesimal rotation, $\Delta\mathbf{r}$, can be given by:

$$\Delta\mathbf{r} = \Delta\phi\mathbf{i}_0 + \Delta\theta\mathbf{j}' + \Delta\psi\mathbf{k}'' \tag{2.6}$$

Dividing Equation 2.6 by Δt and taking the limit as $\Delta t \to 0$, yields the time rate of change of $\Delta\mathbf{r}$ and is given by:

$$\lim_{t\to 0} \frac{\Delta\mathbf{r}}{\Delta t} = \dot{\mathbf{r}} = \dot{\phi}\mathbf{i}_0 + \dot{\theta}\mathbf{j}' + \dot{\psi}\mathbf{k}'' \tag{2.7}$$

Based on the frame definitions, the following transformations can be written:

$$\mathbf{j}' = c\phi\mathbf{j}_0 - s\phi\mathbf{k}_0 \tag{2.8a}$$

$$\mathbf{k}'' = -s\theta\mathbf{i}_0 + c\theta s\phi\mathbf{j}_0 + c\theta c\phi\mathbf{k}_0 \tag{2.8b}$$

Substituting Equations 2.8a and 2.8b into Equation 2.7 yields the following equation:

$$\dot{\mathbf{r}} = (\dot{\phi} - \dot{\psi}s\theta)\mathbf{i}_0 + (\dot{\theta}c\phi + \dot{\psi}c\theta s\phi)\mathbf{j}_0 + (-\dot{\theta}s\phi + \dot{\psi}c\theta c\phi)\mathbf{k}_0 \tag{2.9}$$

Note that $\dot{\mathbf{r}}$ corresponds to the angular velocity of the vehicle $\boldsymbol{\omega}_0 = p\mathbf{i}_0 + q\mathbf{j}_0 + r\mathbf{k}_0$. Therefore,

$$\dot{\mathbf{r}} = \boldsymbol{\omega}_0 \tag{2.10}$$

Thus, the following equations hold:

$$\begin{aligned} p &= \dot{\phi} - \dot{\psi}s\theta \\ q &= \dot{\theta}c\phi + \dot{\psi}c\theta s\phi \\ r &= -\dot{\theta}s\phi + \dot{\psi}c\theta c\phi \end{aligned} \tag{2.11}$$

Equation 2.11 can be reorganized into a non-orthogonal transformation that maps the time rate of change of Euler angles into the body-fixed frame angular velocity components as follows:

$$\boldsymbol{\omega}_0 = \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & c\theta s\phi \\ 0 & -s\phi & c\theta c\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{2.12}$$

$$\boldsymbol{\omega}_0 = \mathbf{T}_2^{-1}\dot{\boldsymbol{\eta}}_2 \tag{2.13}$$

Inverting $\mathbf{T}_2^{-1}$ gives the transformation matrix between the vehicle's body-fixed rotational velocity and the time rate of change of the Euler angles,

$$\mathbf{T}_2 = (\mathbf{T}_2^{-1})^{-1} \tag{2.14}$$

Consequently, $\mathbf{T}_2$ is obtained as:

$$\mathbf{T}_2 = \begin{bmatrix} 1 & -s\psi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \tag{2.15}$$

Note that the transformation matrix $\mathbf{T}_2$ is not an orthonogal matrix because the ψ, θ, ϕ entries do not form a vector entity, that is, they are not associated with orthogonal directions.

2.2.4 Kinematic Equations

In this section, the transformation matrices $\mathbf{T}_1$ and $\mathbf{T}_2$ are used to transform the vehicle's absolute velocity and acceleration between body-fixed and inertial frame representations.

Velocity Equations

The absolute spatial velocity of the vehicle in terms of the inertial frame can be given according to:

$$\begin{bmatrix} \dot{\boldsymbol{\eta}}_2 \\ \dot{\boldsymbol{\eta}}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{T}_2 & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{T}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega}_0 \\ \mathbf{v}_0 \end{bmatrix} \tag{2.16}$$

Equation 2.16 can be written as follows:

$$\dot{\underline{\mathbf{x}}} = \underline{\mathbf{T}}\,\underline{\mathbf{v}} \tag{2.17}$$

where the transformation matrix $\underline{\mathbf{T}}$ is given as:

$$\underline{\mathbf{T}} = \begin{bmatrix} \mathbf{T}_2 & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{T}_1 \end{bmatrix} \tag{2.18}$$

and Equation 2.17 relates both translational and angular velocities expressed in the vehicle's body-fixed frame to the quantities in terms of the inertial frame.

Acceleration Equations

The acceleration equations can be obtained by taking the derivative of Equation 2.17:

$$\ddot{\underline{\mathbf{x}}} = \dot{\underline{\mathbf{T}}}\,\underline{\mathbf{v}} + \underline{\mathbf{T}}\,\dot{\underline{\mathbf{v}}} \tag{2.19}$$

where $\dot{\underline{\mathbf{T}}}$ is comprised of $\dot{\mathbf{T}}_1$ and $\dot{\mathbf{T}}_2$. The transformation matrix $\dot{\mathbf{T}}_1$ is computed as:

$$\dot{\mathbf{T}}_1 = \mathbf{T}_1\tilde{\boldsymbol{\omega}}_0 \tag{2.20}$$

where $\sim$ refers to a skew symmetric matrix formed from the components of the vector entity as described in Appendix B. The proof for the time derivative of a rotation matrix is given in Appendix C.

As for $\dot{\mathbf{T}}_2$, it is given by:

$$\dot{\mathbf{T}}_2 = \begin{bmatrix} 0 & c\phi\dot{\phi}t\theta + (1+t^2\theta)s\phi\dot{\theta} & -s\phi\dot{\phi}t\theta + (1+t^2\theta)s\phi\dot{\theta} \\ 0 & -s\phi\dot{\phi} & -c\phi\dot{\phi} \\ 0 & \left(c\phi c\theta\dot{\phi} + s\phi s\theta\dot{\theta}\right)/c^2\theta & \left(-s\phi c\theta\dot{\phi} + c\phi s\theta\dot{\theta}\right)/c^2\theta \end{bmatrix} \tag{2.21}$$

2.3 Manipulator Kinematics in the URVM Serial Chain

2.3.1 Overview

In this section, the velocity and the acceleration of the URV at the junction point with the manipulator will be considered to be the manipulator base velocity and acceleration. As such, the kinematic effect of the URV throughout the serial chain of the URVM system can be incorporated.

The outward iterations of the Newton-Euler Recursive Algorithm [48] are used to compute the rotational velocity and the linear and rotational acceleration of the centre of mass of each link of the manipulator for a given instant of time. The kinematic parameters for each link are defined in terms of the link's body-fixed frame. The body-fixed frames are attached to the initial joint of each successive link using the Denavit-Hartenberg convention [48]. This step is followed by numbering the links starting from 1 through n, with n representing the end-effector. The base is identified as link 0. Again, in this work the base is considered to be the URV. The position state vectors for serial-manipulators containing only revolute joints are defined as $\mathbf{q} = \begin{bmatrix} q_1 & q_2 & \dots & q_n \end{bmatrix}^T$, where q_i is the rotation angle of the i^{th} revolute joint. For serial-manipulators containing prismatic joints, the corresponding state vector component is replaced by d_i, where d_i is the translational displacement of the i^{th} prismatic joint.

2.3.2 Link Transformations

A homogeneous transformation matrix ${}^{i-1}_{i}\mathbf{T}$ describes the location and orientation of link frame i with respect to link-frame $i-1$. The homogeneous transformation matrix is constructed based on the modified Denavit&Hartenberg parameters [48] and is expressed as shown below:

$$ {}^{i-1}_{i}\mathbf{T} = \mathbf{R}_{\mathbf{x}_{i-1}}(\alpha_{i-1})\,\mathbf{D}_{\mathbf{x}_{i-1}}(a_{i-1})\,\mathbf{D}_{\mathbf{z}_i}(d_i)\,\mathbf{R}_{\mathbf{z}_i}(\theta_i) \tag{2.22} $$

yielding [48]:

$$ {}^{i-1}_{i}\mathbf{T} = \begin{bmatrix} c\theta i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.23} $$

This expression describes the necessary successive motions to transform from the link frame $i-1$ to the link frame i. In this expression, $\mathbf{R}_{\mathbf{x}_{i-1}}(\alpha_{i-1})$, $\mathbf{D}_{\mathbf{x}_{i-1}}(a_{i-1})$, $\mathbf{D}_{\mathbf{z}_i}(d_i)$ and $\mathbf{R}_{\mathbf{z}_i}(\theta_i)$ represent a rotation about the $\mathbf{x}_{i-1}$ axis, a translation along the $\mathbf{x}_{i-1}$, a translation along the $\mathbf{z}_i$ axis, and lastly a rotation about $\mathbf{z}_i$, respectively. The components of the homogeneous transformation matrix are

$$ {}^{i-1}_{i}\mathbf{T} = \begin{bmatrix} {}^{i-1}_{i}\mathbf{R} & {}^{i-1}\mathbf{p}_{i-1\to i} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{2.24} $$

where ${}^{i-1}_{i}\mathbf{R}$ is a 3×3 rotation matrix that transforms representations in the i^{th} link frame to representations in the *i-1*th link frame, and ${}^{i-1}\mathbf{p}_{i-1\to i}$ is a vector denoting the translation from the origin of the link frame $i-1$ to the origin of the link frame i.

2.3.3 Extension of the Aforementioned Equations to the Kinematics of Serial-Chain Manipulators

The principles described in Appendix A can be applied to the kinetics of serial-manipulators. A serial-manipulator is composed of successive links. Each link has a 1 degree of freedom motion contribution to its outboard links due to its physical structure. This implies that the computation of the velocity and the acceleration of each link requires knowing the velocity and the acceleration of its inboard (preceding) links. Therefore, the calculations must be initiated starting from the base, the URV, through the structure.

Let's consider a land-based n-link serial-manipulator, which is composed of only revolute joints. The equations of motion of link 1 can be derived by means of Figure A.1. Frame $\{E\}$ represents the inertial frame. Frame $\{0\}$ is attached to the base of link 1. The linear velocity[1] of link 1,$\mathbf{v}_1$, is equal to the linear velocity of the base, $\mathbf{v}_0$, *i.e.*,$\mathbf{v}_0$=$\mathbf{v}_1$. This statement can be extended to the linear accelerations, *i.e.*, $\mathbf{a}_0$=$\mathbf{a}_1$. Let's calculate the linear velocity of link 2. In Figure A.1, vector $\mathbf{p}_{b/0} = \mathbf{p}_{b/1}$ corresponds to the position vector from the base joint of link 1 to the point where the body-fixed frame of the outboard link (link 2) is attached. Considering the recursive nature of serial-manipulators, the calculations for link 2 must start from the base of link 1.

Since joint 1 is revolute, Equation A.14 is used to calculate the linear velocity of link 2. Note that the velocity of link 1 is equal to the velocity of the URV. In Equation A.14, the term $\boldsymbol{\omega}_0 = \boldsymbol{\omega}_1$ corresponds to the absolute angular velocity of

[1]The linear velocity or acceleration of a link refers to the absolute linear velocity or acceleration of the origin of the link frame.

link 1. In order to calculate the linear acceleration of link 2, Equation A.15 is used. In Equation A.15, $\mathbf{a}_0 = \mathbf{a}_1$ and $\boldsymbol{\alpha}_0 = \boldsymbol{\alpha}_1$, as they represent the base motion. In order to calculate the linear velocity of link 3, the angular velocity of link 3 must be known. According to Theorem 2 of Appendix A, this can be found by adding each rotational contribution of preceding links to the joint rate of link 3 expressed as a vector quantity. The angular acceleration is computed by taking the time derivative of the angular velocity of link 3 using Theorem 1 of Appendix A. This process carries out throughout the rest of the links successively.

2.3.4 Kinematic Equations for Serial-Chain Manipulators

The outward equations of the Newton-Euler recursive dynamic algorithm are presented in this subsection. These equations establish the translational and rotational velocities of the links based on a known base velocity.

Starting from $i = 0$ through $i = n - 1$, the angular rotational velocity from link to link is given based on Theorem 2 of Appendix A as:

$$^{i+1}\boldsymbol{\omega}_{i+1} = {}^{i+1}_{i}\mathbf{R}\ {}^{i}\boldsymbol{\omega}_i + \dot{q}_{i+1}\ {}^{i+1}\mathbf{z}_{i+1} \tag{2.25}$$

Since there is no rotational motion contribution of prismatic joints to the serial-manipulator, for a prismatic joint Equation 2.25 simplifies to:

$$^{i+1}\boldsymbol{\omega}_{i+1} = {}^{i+1}_{i}\mathbf{R}\ {}^{i}\boldsymbol{\omega}_i \tag{2.26}$$

where

$^{i+1}\boldsymbol{\omega}_{i+1}$: angular velocity vector of link $i+1$ in terms of frame $i+1$

${}^{i+1}_{i}\mathbf{R}$: rotation matrix from link frame i to frame $i+1$

${}^{i}\boldsymbol{\omega}_{i}$: angular velocity vector of link i in terms of frame i

$\dot{q}_{i+1}$: angular velocity of joint $i+1$

${}^{i+1}\mathbf{z}_{i+1}$: unit vector in z direction of link frame $i+1$

Taking the time derivative of the angular velocity yields the angular acceleration of the link. For a rotational joint, the time derivative of Equation 2.25 is given by:

$$ {}^{i+1}\boldsymbol{\alpha}_{i+1} = {}^{i+1}_{i}\mathbf{R}\,{}^{i}_{i}\boldsymbol{\alpha} + {}^{i+1}_{i}\mathbf{R}\,{}^{i}\boldsymbol{\omega}_{i} \times \dot{q}_{i+1}\,{}^{i+1}\mathbf{z}_{i+1} + \ddot{q}_{i+1}\,{}^{i+1}\mathbf{z}_{i+1} \tag{2.27} $$

Since there is no rotational velocity between link i and link $i+1$ in case of prismatic joints, for a prismatic joint Equation 2.27 simplifies to:

$$ {}^{i+1}\boldsymbol{\alpha}_{i+1} = {}^{i+1}_{i}\mathbf{R}\,{}^{i}\boldsymbol{\alpha}_{i} \tag{2.28} $$

where

${}^{i+1}\boldsymbol{\alpha}_{i+1}$: angular acceleration vector of link $i+1$ in terms of frame $i+1$

${}^{i}\boldsymbol{\alpha}_{i}$: angular acceleration vector of link i in terms of frame i

$\ddot{q}_{i+1}$: angular acceleration of joint $i+1$.

The linear acceleration of each link frame origin is given by:

$$ {}^{i+1}\mathbf{a}_{i+1} = {}^{i+1}_{i}\mathbf{R}\left({}^{i}\boldsymbol{\alpha}_{i} \times {}^{i}\mathbf{p}_{i\rightarrow i+1} + {}^{i}\,\boldsymbol{\omega}_{i} \times \left({}^{i}\boldsymbol{\omega}_{i} \times {}^{i}\mathbf{p}_{i\rightarrow i+1}\right) + {}^{i}\mathbf{a}_{i}\right) \tag{2.29} $$

Since the joint rate of a prismatic joint changes in time with respect to its body-fixed frame, the Coriolis and the centripetal acceleration terms appear in the derivative of the linear velocity expression:

$$ \begin{aligned} {}^{i+1}\mathbf{a}_{i+1} \;=\; & {}^{i+1}_{i}\mathbf{R}\left({}^{i}\boldsymbol{\alpha}_{i} \times {}^{i}\mathbf{p}_{i\rightarrow i+1} + {}^{i}\,\boldsymbol{\omega}_{i} \times \left({}^{i}\boldsymbol{\omega}_{i} \times {}^{i}\mathbf{p}_{i\rightarrow i+1}\right) + {}^{i}\mathbf{a}_{i}\right) + \\ & 2\,{}^{i+1}\boldsymbol{\omega}_{i+1} \times \dot{d}_{i+1}\,{}^{i+1}\mathbf{z}_{i+1} + \ddot{d}_{i+1}\,{}^{i+1}\mathbf{z}_{i+1} \end{aligned} \tag{2.30} $$

where

${}^{i+1}\mathbf{a}_{i+1}$: absolute translational acceleration of link frame $i+1$ in terms of frame $i+1$

${}^{i}\mathbf{p}_{i\to i+1}$: position vector from link frame i to link frame $i+1$ in terms of frame i

${}^{i}\mathbf{a}_{i}$: absolute translational acceleration of link frame i in terms of frame i

$\dot{d}_{i+1}$: translational velocity of joint $i+1$

$\ddot{d}_{i+1}$: absolute translational acceleration of joint $i+1$.

Finally, the linear acceleration of the centre of mass of each link is given by

$$ {}^{i+1}\mathbf{a}_{C_{i+1}} = {}^{i+1}\boldsymbol{\alpha}_{i+1} \times {}^{i+1}\mathbf{p}_{i+1\to C_{i+1}} + {}^{i+1}\boldsymbol{\omega}_{i+1} \times \left({}^{i+1}\boldsymbol{\omega}_{i+1} \times {}^{i+1}\mathbf{p}_{i+1\to C_{i+1}}\right) + {}^{i+1}\mathbf{a}_{i+1} \tag{2.31} $$

where

${}^{i+1}\mathbf{p}_{i+1\to C_{i+1}}$: position vector of centre of mass of link $i+1$ relative to the origin of link frame $i+1$ in terms of frame $i+1$.

Note that Equation 2.31 is the same for both prismatic and revolute joints.

Chapter 3

Articulated-Body Equations

3.1 Overview

The forward dynamic problem is the calculation of joint accelerations for a given set of joint positions, velocities and torques. In this chapter, the equations of motion for a link within a serial-manipulator that were derived in Chapter 2 are expressed in terms of spatial vector notation. These equations form the basis for the derivation of the articulated-body equations. After the derivation of the articulated-body equations, the incorporation of the mobile base effect and the gravity forces into the derived equations are presented. Finally, a methodology to solve the forward dynamic problem is given at the end of the chapter based on [25].

3.2 Spatial Kinematics of Manipulator Links

The dynamic equations of the articulated-body algorithm are in terms of the linear and angular kinematic parameters of each link. The definitions of the linear and

angular kinematic quantities were already given in Chapter 2. In this section, the spatial vector representation of linear and angular kinematic quantities are written in a spatial vector format.

A spatial vector is a 6 dimensional vector, which combines the associated 3×1 vector of linear kinematic quantities and the 3×1 vector of angular kinematic quantities. The spatial representation of vectors simplifies the dynamics equations of multi-body systems, as it reduces the number of equations [49]. In the articulated-body algorithm, the kinematic quantities are represented by 6×1 spatial vectors and as a result of the reduction in the dynamic equation set, the articulated-body algorithm provides better computational efficiency over its 3- dimensional-vector-based counterparts [49].

In the following articulated-body dynamic equations, the force and moment equations are written with respect to the origin of each link frame. This is in contrast to the Newton-Euler recursive dynamic equations where the dynamic force-moment balance equations are written with respect to the centre of mass of each link.

Let's consider the manipulator link shown in Figure 3.1. In Figure 3.1, joint $i-1$ and i are at the base of links $i-1$ and i. The body-fixed frame is attached to joint $i-1$ of link $i-1$, which is labeled as a. The body-fixed frame of the outboard link i is attached to joint i, which is labeled as b. Point b is coincident with the junction point between the two successive links and considered to exist on link $i-1$ and exhibits the same angular velocity as point a. The spatial velocity vector of point a and b is defined as $\underline{\mathbf{v}}_a = \begin{bmatrix} {}^{i-1}\boldsymbol{\omega}_{i-1} & {}^{i-1}\mathbf{v}_{i-1} \end{bmatrix}^T$ and $\underline{\mathbf{v}}_b = \begin{bmatrix} {}^{i}\boldsymbol{\omega}_{i-1} & {}^{i}\mathbf{v}_{i} \end{bmatrix}^T$, respectively. The position vector from point a to point b is defined as ${}^{i-1}\mathbf{p}_{i-1\to i}$. The 6×6 spatial transformation matrix ${}^{i}\underline{\mathbf{X}}_{i-1}$ that maps these quantities from link frame $i-1$ to link

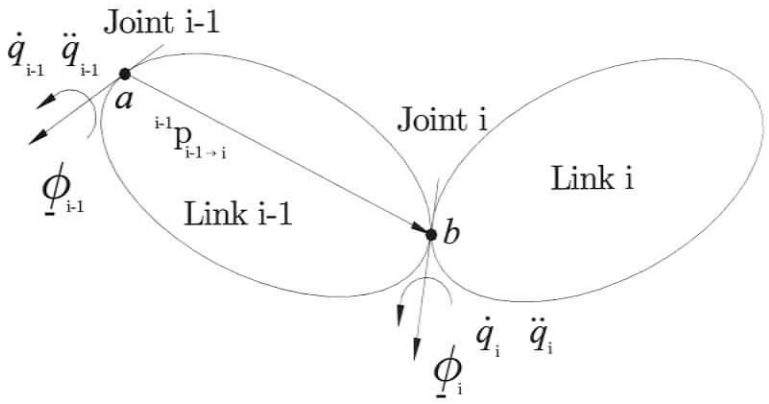


Figure 3.1: Points on the Manipulator Links

frame i is given by:

$$\begin{bmatrix} {}^{i}\boldsymbol{\omega}_{i-1} \\ {}^{i}\mathbf{v}_{i} \end{bmatrix} = \begin{bmatrix} {}^{i}_{i-1}\mathbf{R} & \mathbf{0}_{3\times 3} \\ {}^{i}_{i-1}\mathbf{R}\ {}^{i-1}\tilde{\mathbf{p}}^{T}_{i-1\to i} & {}^{i}_{i-1}\mathbf{R} \end{bmatrix} \begin{bmatrix} {}^{i-1}\boldsymbol{\omega}_{i-1} \\ {}^{i-1}\mathbf{v}_{i-1} \end{bmatrix} \tag{3.1}$$

$$\underline{\mathbf{v}}_b = {}^{i}\underline{\mathbf{X}}_{i-1}\underline{\mathbf{v}}_a \tag{3.2}$$

Now, the spatial velocity vector of link i is seeked. The motion afforded by a single joint is defined by a spatial unit vector ${}^{i}\underline{\boldsymbol{\phi}}_i$ The vector ${}^{i}\underline{\boldsymbol{\phi}}_i$ maps the contribution of the i^{th} joint, a scalar rotation rate, to the spatial motion of the manipulator. The vector ${}^{i}\underline{\boldsymbol{\phi}}_i$ is defined as ${}^{i}\underline{\boldsymbol{\phi}}_i = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T$ for revolute joints and ${}^{i}\underline{\boldsymbol{\phi}}_i = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$ for prismatic joints in terms of the body-fixed frame attached to the i^{th} link. In order to find the spatial velocity vector of the outboard link i, the motion contribution of link i must be considered. For the sake of simplicity, the joint rate of the i^{th} link is defined by the same symbol $\dot{\xi}_i$ for revolute joints and prismatic joints, *i.e.*, $\dot{\xi}_i$ corresponds to $\dot{q}_i$ for revolute joints and $\dot{d}_i$ for prismatic joints.

The spatial velocity of link i, regardless of whether joint i is prismatic or revolute, is given by:

$$\begin{bmatrix} {}^{i}\boldsymbol{\omega}_i \\ {}^{i}\mathbf{v}_i \end{bmatrix} = \begin{bmatrix} {}^{i}_{i-1}\mathbf{R} & \mathbf{0}_{3\times3} \\ {}^{i}_{i-1}\mathbf{R}\ {}^{i-1}\tilde{\mathbf{p}}^{T}_{i-1\to i} & {}^{i}_{i-1}\mathbf{R} \end{bmatrix} \begin{bmatrix} {}^{i-1}\boldsymbol{\omega}_{i-1} \\ {}^{i-1}\mathbf{v}_{i-1} \end{bmatrix} + {}^{i}\underline{\boldsymbol{\phi}}_i\dot{\xi}_i \tag{3.3}$$

With regards to the spatial accelerations, Equations 2.27 and 2.29 can be written in a spatial vector form for a revolute joint as follows:

$$\begin{bmatrix} {}^{i}\boldsymbol{\alpha}_i \\ {}^{i}\mathbf{a}_i \end{bmatrix} = \begin{bmatrix} {}^{i}_{i-1}\mathbf{R} & \mathbf{0}_{3\times3} \\ {}^{i}_{i-1}\mathbf{R}\ {}^{i-1}\tilde{\mathbf{p}}^{T}_{i-1\to i} & {}^{i}_{i-1}\mathbf{R} \end{bmatrix} \begin{bmatrix} {}^{i-1}\boldsymbol{\alpha}_{i-1} \\ {}^{i-1}\mathbf{a}_{i-1} \end{bmatrix} + {}^{i}\underline{\boldsymbol{\phi}}_i\ddot{\xi}_i + \begin{bmatrix} \mathbf{0}_{3\times1} \\ {}^{i}_{i-1}\mathbf{R}\ \left[{}^{i-1}\boldsymbol{\omega}_{i-1} \times \left({}^{i-1}\boldsymbol{\omega}_{i-1} \times {}^{i-1}\mathbf{p}_{i-1\to i}\right)\right] \end{bmatrix} + \begin{bmatrix} {}^{i}\boldsymbol{\omega}_i \times \dot{\xi}_i\ \mathbf{z}_i \\ \mathbf{0}_{3\times1} \end{bmatrix} \tag{3.4}$$

where $\mathbf{z}_i$ is called joint unit vector and is defined for both revolute and prismatic joints as:

$$\mathbf{z}_i = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{3.5}$$

For a prismatic joint, Equations 2.28 and 2.30 are combined in a spatial vector form as follows:

$$\begin{bmatrix} {}^{i}\boldsymbol{\alpha}_i \\ {}^{i}\mathbf{a}_i \end{bmatrix} = \begin{bmatrix} {}^{i}_{i-1}\mathbf{R} & \mathbf{0}_{3\times3} \\ {}^{i}_{i-1}\mathbf{R}\ {}^{i-1}\tilde{\mathbf{p}}^{T}_{i-1\to i} & {}^{i}_{i-1}\mathbf{R} \end{bmatrix} \begin{bmatrix} {}^{i-1}\boldsymbol{\alpha}_{i-1} \\ {}^{i-1}\mathbf{a}_{i-1} \end{bmatrix} + {}^{i}\underline{\boldsymbol{\phi}}_i\ddot{\xi}_i + \begin{bmatrix} \mathbf{0}_{3\times1} \\ {}^{i}_{i-1}\mathbf{R}\ \left[{}^{i-1}\boldsymbol{\omega}_{i-1} \times \left({}^{i-1}\boldsymbol{\omega}_{i-1} \times {}^{i-1}\mathbf{p}_{i-1\to i}\right)\right] \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times1} \\ 2\left({}^{i}\boldsymbol{\omega}_i \times \dot{\xi}_i\mathbf{z}_i\right) \end{bmatrix} \tag{3.6}$$

Equations 3.4 and 3.6 can be written in a much more compact form as follows:

$${}^{i}\underline{\mathbf{a}}_i = {}^{i}\underline{\mathbf{X}}_{i-1}\ {}^{i-1}\underline{\mathbf{a}}_{i-1} + {}^{i}\underline{\boldsymbol{\phi}}_i\ddot{\xi}_i + {}^{i}\underline{\boldsymbol{\zeta}}_i \tag{3.7}$$

where ${}^{i}\underline{\zeta}_{i}$ is the vector of Coriolis and centripetal accelerations, which differs between the revolute and prismatic joint cases.

3.3 Spatial Dynamic Equations for a Link

3.3.1 Link Translational Dynamics

According to Newton's second law, the resultant of the external forces, $\mathbf{f}$, applied to a rigid body, whose total mass is m, causes the absolute acceleration $\mathbf{a}_c$ at the centre of mass of the rigid body. The sum of external forces is related to the acceleration of the rigid body according to:

$$\mathbf{f} = \mathbf{m}\mathbf{a}_c \tag{3.8}$$

where m is defined as:

$$\mathbf{m} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & m \end{bmatrix} \tag{3.9}$$

Equation 3.8 is applied to the manipulator links in a recursive manner. Figure 3.2 shows a free-body diagram of a manipulator link in a serial-chain. In the figure, $\mathbf{f}_i$ is the total force exerted onto link i at the joint i by its inboard link $i-1$ and $\mathbf{f}_{i+1}$ is the force exerted by link i onto the next outboard link $i+1$ at the joint $i+1$.

The force-balance equation can be written in terms of the acceleration of the joint i, the link base, as follows:

$$\mathbf{f}_i - {}^{i}_{i+1}\mathbf{R}\ {}^{i+1}\mathbf{f}_{i+1} = \mathbf{m}_i\ {}^{i}\mathbf{a}_{C_i} \tag{3.10}$$

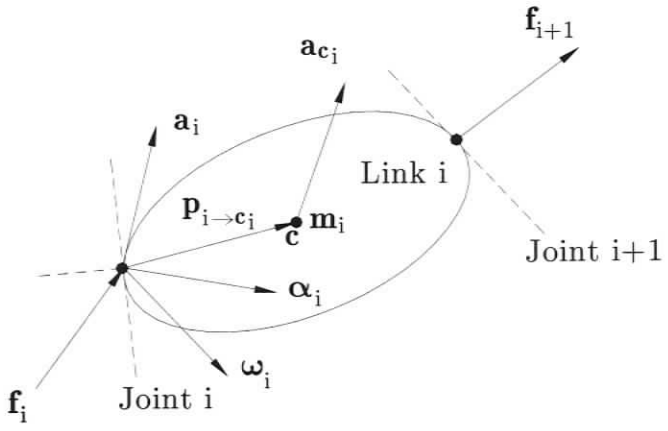


Figure 3.2: Link in a Serial-Manipulator for the Force-Balance Equation

Substituting Equation 2.31 into Equation 3.10 yields:

$$ {}^{i}\mathbf{f}_i - {}^{i}_{i+1}\mathbf{R}\ {}^{i+1}\mathbf{f}_{i+1} = \mathbf{m}_i\ {}^{i}\mathbf{a}_i - \mathbf{m}_i\ {}^{i}\mathbf{p}_{i\to C_i} \times {}^{i}\boldsymbol{\alpha}_i + {}^{i}\boldsymbol{\omega}_i \times \left({}^{i}\boldsymbol{\omega}_i \times \mathbf{m}_i\ {}^{i}\mathbf{p}_{i\to C_i}\right) \quad (3.11) $$

where superscript i represents link's body-fixed frame. The term $\mathbf{m}_i\ {}^{i}\mathbf{p}_{i\to C_i}$ is called the first mass moment of link i and is defined by $\mathbf{h}_i$. Using the skew-symmetric definition in Appendix B, Equation 3.11 becomes:

$$ \mathbf{f}_i - {}^{i}_{i+1}\mathbf{R}\ {}^{i+1}\mathbf{f}_{i+1} = \mathbf{m}_i\ {}^{i}\mathbf{a}_i - \tilde{\mathbf{h}}_i\ {}^{i}\boldsymbol{\alpha}_i + {}^{i}\boldsymbol{\omega}_i \times \left({}^{i}\boldsymbol{\omega}_i \times \mathbf{h}_i\right) \quad (3.12) $$

The skew-symmetric property ,$(-\tilde{\mathbf{h}}_i = \tilde{\mathbf{h}}_i^T)$, produces:

$$ \mathbf{f}_i - {}^{i}_{i+1}\mathbf{R}\ {}^{i+1}\mathbf{f}_{i+1} = \mathbf{m}_i\ {}^{i}\mathbf{a}_i + \tilde{\mathbf{h}}_i^T\ {}^{i}\boldsymbol{\alpha}_i + {}^{i}\boldsymbol{\omega}_i \times \left({}^{i}\boldsymbol{\omega}_i \times \mathbf{h}_i\right) \quad (3.13) $$

Equation 3.13 will be used to form a spatial force-balance equation for a rigid body.

3.3.2 Link Rotational Dynamics

Referring to Figure 3.3, a rigid manipulator link is moving such that the linear acceleration at point a on joint i is $\mathbf{a}_i$, the angular velocity is $\boldsymbol{\omega}_i$, and the angular

acceleration is $\boldsymbol{\alpha}_i$. The total moment $\mathbf{n}$, which must be exerted on the body to cause the motion is given by:

$$\mathbf{n} = \mathbf{m}\ \mathbf{p}_{a\rightarrow C} \times \mathbf{a} + \mathbf{I}_a\ \boldsymbol{\alpha} + \boldsymbol{\omega} \times \mathbf{I}_a\ \boldsymbol{\omega} \tag{3.14}$$

where $\mathbf{p}_{a\rightarrow C}$ is the position vector from point a on joint i to the centre of mass of the body. The inertia tensor is expressed in terms of link i's body-fixed frame which originates at the point a on joint i:

$$\mathbf{I}_a = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \tag{3.15}$$

Note that the first term in Equation 3.14 accounts for the use of "a" as a reference point. The last two terms in Equation 3.14 are due to the time derivative of the angular momentum of the rigid body.

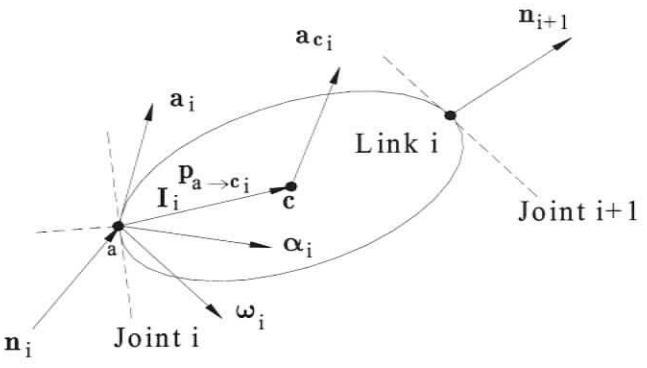


Figure 3.3: Link in a Serial-Manipulator for the Moment-Balance Equation

Equation 3.14 can be extended to the manipulator dynamics by defining the overall moment applied to the link. In Figure 3.3, $\mathbf{n}_i$ is the moment in terms of link frame i

exerted onto link i by its inboard link $i-1$ at its base and $\mathbf{n}_{i+1}$ is the moment exerted by link i onto the next outboard link $i+1$ at the junction point between link i and link $i+1$, the force $\mathbf{f}_{i+1}$ is exerted by link i on the outboard $i+1$ link at the joint $i+1$. The moment equation with respect to the point a can be written in terms of link's body-fixed frame as follows:

$$ {}^{i}\mathbf{n}_i - {}^{i}_{i+1}\mathbf{R}\,{}^{i+1}\mathbf{n}_{i+1} - {}^{i}\mathbf{p}_{i\to i+1} \times {}^{i}_{i+1}\mathbf{R}\,{}^{i+1}\mathbf{f}_{i+1} = \mathbf{m}_i\,{}^{i}\mathbf{p}_{a\to C_i} \times {}^{i}\mathbf{a}_i + {}^{i}\mathbf{I}_i\,{}^{i}\boldsymbol{\alpha}_i + {}^{i}\boldsymbol{\omega}_i \times {}^{i}\mathbf{I}_i\,{}^{i}\boldsymbol{\omega}_i \tag{3.16} $$

By taking the origin of the frame of joint i as a reference point, *i.e.*, ${}^{i}\mathbf{p}_{a\to C_i} = {}^{i}\mathbf{p}_{i\to C_i}$, Equation 3.16 can be rearranged by replacing term ${}^{i}\mathbf{p}_{i\to C_i} \times$ with its skew symmetric form ${}^{i}\tilde{\mathbf{p}}_{i\to C_i}$ and using the definition of the mass moment given in Section 3.3.1 as follows:

$$ {}^{i}\mathbf{n}_i - {}^{i}_{i+1}\mathbf{R}\,{}^{i+1}\mathbf{n}_{i+1} - {}^{i}\tilde{\mathbf{p}}_{i\to i+1}\,{}^{i}_{i+1}\mathbf{R}\,{}^{i+1}\mathbf{f}_{i+1} = \tilde{\mathbf{h}}_i \times {}^{i}\mathbf{a}_i + {}^{i}\mathbf{I}_i\,{}^{i}\boldsymbol{\alpha}_i + {}^{i}\boldsymbol{\omega}_i \times {}^{i}\mathbf{I}_i\,{}^{i}\boldsymbol{\omega}_i \tag{3.17} $$

3.3.3 Spatial Force-Balance Equation

Equations 3.17 and 3.13 can be combined to yield into a single spatial vector dynamics equation as follows:

$$ \begin{bmatrix} {}^{i}\mathbf{n}_i \\ {}^{i}\mathbf{f}_i \end{bmatrix} - \begin{bmatrix} {}^{i}_{i+1}\mathbf{R} & {}^{i}\tilde{\mathbf{p}}_{i\to i+1}\,{}^{i}_{i+1}\mathbf{R} \\ \mathbf{0}_{3\times 3} & {}^{i}_{i+1}\mathbf{R} \end{bmatrix} \begin{bmatrix} {}^{i+1}\mathbf{n}_{i+1} \\ {}^{i+1}\mathbf{f}_{i+1} \end{bmatrix} = \begin{bmatrix} {}^{i}\mathbf{I}_i & \tilde{\mathbf{h}}_i \\ \tilde{\mathbf{h}}_i^T & \mathbf{m} \end{bmatrix} \begin{bmatrix} {}^{i}\boldsymbol{\alpha}_i \\ {}^{i}\mathbf{a}_i \end{bmatrix} + \begin{bmatrix} {}^{i}\boldsymbol{\omega}_i \times {}^{i}\mathbf{I}_i\,{}^{i}\boldsymbol{\omega}_i \\ {}^{i}\boldsymbol{\omega}_i \times ({}^{i}\boldsymbol{\omega}_i \times \mathbf{h}_i) \end{bmatrix} \tag{3.18} $$

Equation 3.18 can be written in a more compact form as follows:

$$ {}^{i}\underline{\mathbf{f}}_i - {}^{i+1}\underline{\mathbf{X}}_i^T\,{}^{i+1}\underline{\mathbf{f}}_{i+1} = {}^{i}\underline{\mathbf{I}}_i\,{}^{i}\underline{\mathbf{a}}_i - {}^{i}\underline{\boldsymbol{\beta}}_i \tag{3.19} $$

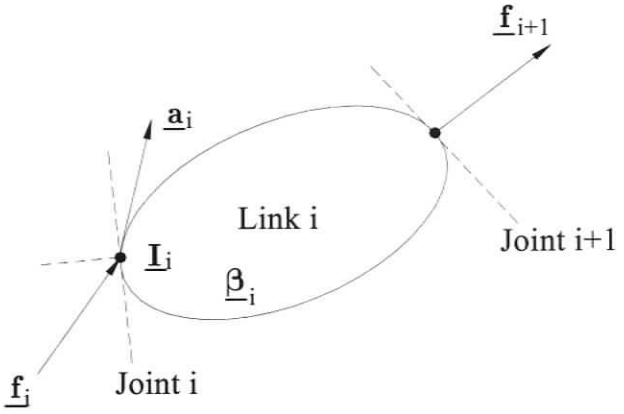


Figure 3.4: Spatial Quantities for a Link

Equation 3.19 is the spatial form of the force balance equation for a rigid body as illustrated in Figure 3.4. The vector $\underline{\mathbf{I}}_i$ is called spatial inertia tensor. The vector $\underline{\boldsymbol{\beta}}_i$ is called bias force that contains all velocity dependent terms. Bias force can be described as a velocity-dependent force that must be exerted onto link i to maintain its current link velocities. This equation along with Equation 3.7 provides the basis for the derivation of the articulated-body algorithm. The gravitational effects are excluded to ease the computational burden and will be incorporated into the articulated-body equations in subsequent chapters.

In the next section, the interaction forces between the link will be the main focus. By introducing the concept of an articulated body, it will be shown how Equation 3.19 for $i = 1...n$ can be reformulated in a recursive set of equations that allow fast evaluation of the forward dynamics.

3.4 Derivation of Articulated Body Inertias

It is known that there is a linear relationship between the acceleration of a body in a rigid body system and the applied forces causing that motion [50]. This linear relationship is given by Equation 3.19. Now, referring to Figure 3.5, the question is "is there any linear relationship between the force $\underline{\mathbf{f}}_i$ applied to a particular member of a multi-body system and the acceleration $\underline{\mathbf{a}}_i$ of that member, taking into account the effect of the rest of the multi-body system, which is defined by $\underline{\mathbf{I}}_i^*$ and $\underline{\boldsymbol{\beta}}_i^*$." Therefore, the following relationship will be seeked in the subsequent derivations:

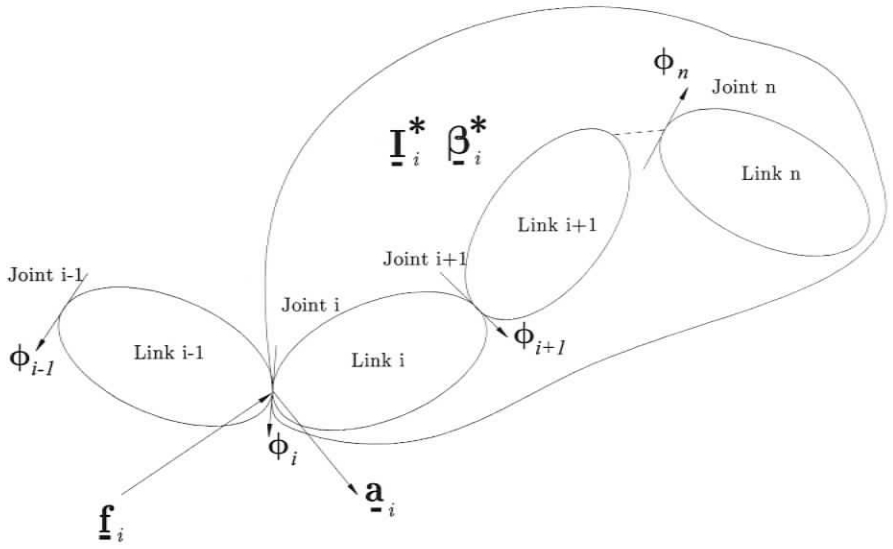


Figure 3.5: Multi-Body System

$$^i\underline{\mathbf{f}}_i = {}^i\underline{\mathbf{I}}_i^{*\,i}\underline{\mathbf{a}}_i - {}^i\underline{\boldsymbol{\beta}}_i^* \tag{3.20}$$

In Equation 3.20, the term $\underline{\mathbf{I}}_i^*$ is called articulated inertia, which is the accumulated inertia of effect of the articulated bodies that is defined as a collection of rigid links

connected by joints. The term $\underline{\boldsymbol{\beta}}_i^*$ is called articulated bias force [49]. Obtaining the articulated properties $\underline{\mathbf{I}}_i^*$ and $\underline{\boldsymbol{\beta}}_i^*$, leads to a definition of a spatial force $\underline{\mathbf{f}}_i$ in terms of the dynamic properties of the articulated body. In what follows, a recursive equation for the terms $\underline{\mathbf{I}}_i^*$ and $\underline{\boldsymbol{\beta}}_i^*$ will be found.

Let's consider the outermost link (link n) of the manipulator in Figure 3.6. If the force balance equation, Equation 3.19, is written for the outermost link (n^{th} articulated body) and then is equalized to Equation 3.20, one can see that the definitions of the spatial inertia tensor and the spatial bias force coincide with the definitions of the articulated inertia and articulated bias force, *i.e.*, $\underline{\mathbf{I}}_i = \underline{\mathbf{I}}_i^*$ and $\underline{\boldsymbol{\beta}}_i = \underline{\boldsymbol{\beta}}_i^*$.

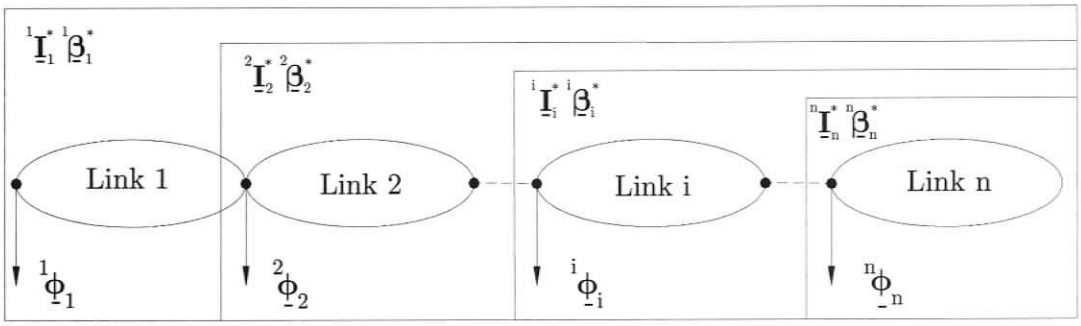


Figure 3.6: Articulated Bodies in a Serial-Chain

Let's take Equation 3.20 as a starting point of the derivation of the articulated inertias and the bias forces. Substituting Equation 3.7, which is the spatial acceleration of the i^{th} link, into Equation 3.20 yields the following equation:

$$ {}^{i}\underline{\mathbf{f}}_i = {}^{i}\underline{\mathbf{I}}_i^* \left({}^{i}\underline{\mathbf{X}}_{i-1} \, {}^{i-1}\underline{\mathbf{a}}_{i-1} + {}^{i}\underline{\boldsymbol{\phi}}_i \ddot{\xi}_i + {}^{i}\underline{\boldsymbol{\zeta}}_i \right) - {}^{i}\underline{\boldsymbol{\beta}}_i^* \tag{3.21} $$

The scalar joint force τ_i can be found by projecting the spatial joint force onto the associated spatial joint unit vector ${}^{i}\underline{\boldsymbol{\phi}}_i$. Therefore, the scalar joint force equation

is given by:

$$\tau_i = {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\mathbf{f}_i \tag{3.22}$$

Substituting Equation 3.21 into Equation 3.22 yields:

$$\tau_i = {}^i\underline{\boldsymbol{\phi}}_i^T \left({}^i\underline{\mathbf{I}}_i^* \left({}^i\underline{\mathbf{X}}_{i-1} \, {}^{i-1}\underline{\mathbf{a}}_{i-1} + {}^i\underline{\boldsymbol{\phi}}_i \ddot{\xi}_i + {}^i\underline{\boldsymbol{\zeta}}_i \right) - {}^i\underline{\boldsymbol{\beta}}_i^* \right) \tag{3.23}$$

Expanding Equation 3.23 gives:

$$\tau_i = {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \left({}^i\underline{\mathbf{X}}_{i-1} \, {}^{i-1}\underline{\mathbf{a}}_{i-1} + {}^i\underline{\boldsymbol{\zeta}}_i \right) - {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\boldsymbol{\beta}}_i^* + {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \, {}^i\underline{\boldsymbol{\phi}}_i \ddot{\xi}_i \tag{3.24}$$

Solving Equation 3.24 for unknown joint acceleration requires the following steps:

$${}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \underline{\boldsymbol{\phi}}_i \ddot{\xi}_i = \tau_i + \underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\boldsymbol{\beta}}_i^* - {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \left({}^i\underline{\mathbf{X}}_{i-1} \, {}^{i-1}\underline{\mathbf{a}}_{i-1} + {}^i\underline{\boldsymbol{\zeta}}_i \right)$$

$$\ddot{\xi}_i = \left({}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \, {}^i\underline{\boldsymbol{\phi}}_i \right)^{-1} \left[\tau_i + {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\boldsymbol{\beta}}_i^* - {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \left({}^i\underline{\mathbf{X}}_{i-1} \, {}^{i-1}\underline{\mathbf{a}}_{i-1} + {}^i\underline{\boldsymbol{\zeta}}_i \right) \right] \tag{3.25}$$

By the following definitions;

$$\begin{aligned} m_i^* &= {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \, {}^i\underline{\boldsymbol{\phi}}_i \\ \tau_i^* &= \tau_i + {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\boldsymbol{\beta}}_i^* \end{aligned} \tag{3.26}$$

Equation 3.25 simplifies to:

$$\ddot{\xi}_i = (m_i^*)^{-1} \left[\tau_i^* - {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \left({}^i\underline{\mathbf{X}}_{i-1} \, {}^{i-1}\underline{\mathbf{a}}_{i-1} + {}^i\underline{\boldsymbol{\zeta}}_i \right) \right] \tag{3.27}$$

Substituting Equation 3.27 back into Equation 3.21 requires the following steps:

$$\begin{aligned} {}^i\underline{\mathbf{f}}_i = {}^i\underline{\mathbf{I}}_i^* \, (& {}^i\underline{\mathbf{X}}_{i-1} \, {}^{i-1}\underline{\mathbf{a}}_{i-1} + \\ {}^i\underline{\boldsymbol{\phi}}_i (& (m_i^*)^{-1} \left[\tau_i^* - {}^i\underline{\boldsymbol{\phi}}_i^T \, {}^i\underline{\mathbf{I}}_i^* \, ({}^i\underline{\mathbf{X}}_{i-1} \, {}^{i-1}\underline{\mathbf{a}}_{i-1} + {}^i\underline{\boldsymbol{\zeta}}_i) \right]) + {}^i\underline{\boldsymbol{\zeta}}_i) - {}^i\underline{\boldsymbol{\beta}}_i^* \end{aligned} \tag{3.28}$$

$$
\begin{aligned}
{}^{i}\underline{\mathbf{f}}_{i} = & \left({}^{i}\underline{\mathbf{I}}_{i}^{*} - {}^{i}\underline{\mathbf{I}}_{i}^{*}\,\underline{\boldsymbol{\phi}}_{i} m_{i}^{-1} \underline{\boldsymbol{\phi}}_{i}^{T}\,{}^{i}\underline{\mathbf{I}}_{i}^{*}\right)^{i}\underline{\mathbf{X}}_{i-1}\,{}^{i-1}\underline{\mathbf{a}}_{i-1} + \\
& ({}^{i}\underline{\mathbf{I}}_{i}^{*} - {}^{i}\underline{\mathbf{I}}_{i}^{*}\,\underline{\boldsymbol{\phi}}_{i} m_{i}^{-1} \underline{\boldsymbol{\phi}}_{i}^{T}\,{}^{i}\underline{\mathbf{I}}_{i}^{*})\,{}^{i}\underline{\boldsymbol{\zeta}}_{i} + {}^{i}\underline{\mathbf{I}}_{i}^{*}\,{}^{i}\underline{\boldsymbol{\phi}}_{i}\, m_{i}^{-1}\tau_{i}^{*} - {}^{i}\underline{\boldsymbol{\beta}}_{i}^{*}
\end{aligned}
\tag{3.29}
$$

By the following definitions:

$$
{}^{i}\underline{\mathbf{k}}_{i} = {}^{i}\underline{\mathbf{I}}_{i}^{*}\,{}^{i}\underline{\boldsymbol{\phi}}_{i} \tag{3.30}
$$

$$
\begin{aligned}
{}^{i}\underline{\mathbf{N}}_{i} &= {}^{i}\underline{\mathbf{I}}_{i}^{*} - {}^{i}\underline{\mathbf{I}}_{i}^{*}\,{}^{i}\underline{\boldsymbol{\phi}}_{i} m_{i}^{-1}\,{}^{i}\underline{\boldsymbol{\phi}}_{i}^{T}\,{}^{i}\underline{\mathbf{I}}_{i}^{*} \\
&= {}^{i}\underline{\mathbf{I}}_{i}^{*} - {}^{i}\underline{\mathbf{k}}_{i}\, m_{i}^{-1}\,{}^{i}\underline{\mathbf{k}}_{i}^{T}
\end{aligned}
\tag{3.31}
$$

Equation 3.29 simplifies to:

$$
{}^{i}\underline{\mathbf{f}}_{i} = {}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\mathbf{X}}_{i-1} - \left[{}^{i}\underline{\boldsymbol{\beta}}_{i}^{*} - {}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\boldsymbol{\zeta}}_{i} - {}^{i}\underline{\mathbf{k}}_{i}\,(m_{i}^{*})^{-1}\,\tau_{i}^{*}\right] \tag{3.32}
$$

Finally, Equation 3.32 is substituted into Equation 3.19 for link $i-1$, which is the rigid-body force equation:

$$
\begin{aligned}
& {}^{i-1}\underline{\mathbf{f}}_{i-1} - {}^{i}\underline{\mathbf{X}}_{i-1}^{T}\left({}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\mathbf{X}}_{i-1} - \left[{}^{i}\underline{\boldsymbol{\beta}}_{i}^{*} - {}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\boldsymbol{\zeta}}_{i} - {}^{i}\underline{\mathbf{k}}_{i}\,(m_{i}^{*})^{-1}\,\tau_{i}^{*}\right]\right) = \\
& {}^{i-1}\underline{\mathbf{I}}_{i-1}\,{}^{i-1}\underline{\mathbf{a}}_{i-1} - {}^{i-1}\underline{\boldsymbol{\beta}}_{i-1}
\end{aligned}
\tag{3.33}
$$

The above equation can be put into the same format as Equation 3.20 as follows:

$$
\begin{aligned}
{}^{i-1}\underline{\mathbf{f}}_{i-1} = & \left({}^{i-1}\underline{\mathbf{I}}_{i-1} + {}^{i}\underline{\mathbf{X}}_{i-1}^{T}\,{}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\mathbf{X}}_{i-1}\right)^{i-1}\underline{\mathbf{a}}_{i-1} - \\
& \left({}^{i-1}\underline{\boldsymbol{\beta}}_{i-1} + {}^{i}\underline{\mathbf{X}}_{i-1}^{T}\left({}^{i}\underline{\boldsymbol{\beta}}_{i}^{*} - {}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\boldsymbol{\zeta}}_{i} - {}^{i}\underline{\mathbf{k}}_{i}\,(m_{i}^{*})^{-1}\,\tau_{i}^{*}\right)\right)
\end{aligned}
\tag{3.34}
$$

By the following definitions:

$$
\begin{aligned}
{}^{i-1}\underline{\mathbf{I}}_{i-1}^{*} &= {}^{i-1}\underline{\mathbf{I}}_{i-1} + {}^{i}\underline{\mathbf{X}}_{i-1}^{T}\,{}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\mathbf{X}}_{i-1} \\
{}^{i-1}\underline{\boldsymbol{\beta}}_{i-1}^{*} &= {}^{i-1}\underline{\boldsymbol{\beta}}_{i-1} + {}^{i}\underline{\mathbf{X}}_{i-1}^{T}\left({}^{i}\underline{\boldsymbol{\beta}}_{i}^{*} - {}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\boldsymbol{\zeta}}_{i} - {}^{i}\underline{\mathbf{k}}_{i}\,(m_{i}^{*})^{-1}\,\tau_{i}^{*}\right)
\end{aligned}
\tag{3.35}
$$

The recursive equation has been finally found as:

$$
{}^{i-1}\underline{\mathbf{f}}_{i-1} = {}^{i-1}\underline{\mathbf{I}}_{i-1}^{*}\,{}^{i-1}\underline{\mathbf{a}}_{i-1} - {}^{i-1}\underline{\boldsymbol{\beta}}_{i-1}^{*} \tag{3.36}
$$

where the terms ${}^{i-1}\underline{\underline{\mathbf{I}}}^{*}_{i-1}$ and ${}^{i-1}\underline{\boldsymbol{\beta}}^{*}_{i-1}$ are called the articulated-body inertia and bias force of the $(i-1)^{th}$ articulated body, respectively. In Equation 3.36, the force term ${}^{i-1}\underline{\mathbf{f}}_{i-1}$ is a known quantity, and represents applied force to the link *i-1*. Knowing ${}^{i-1}\underline{\mathbf{f}}_{i-1}$, it is obvious that Equation 3.36 can be solved for accelerations ${}^{i-1}\underline{\mathbf{a}}_{i-1}$ once the associated articulated inertias and bias forces are computed. Since the articulated inertia is always symmetric and positive definite, it is always invertible.

Equation 3.36 implies a recursive relationship starting from the end-effector to the base, as the $(i-1)^{th}$ articulated-body inertia is a function of the articulated-body inertia of the outboard links. The same statement also applies to the bias forces in the serial-chain. Therefore, the initial values of the articulated-body inertia and bias force for the end-effector n (last link) must be known to propagate Equation 3.20 through the serial-chain. For the end-effector, as explained before , the force balance equation, Equation 3.19, coincides with the articulated-body equation, Equation 3.20, *i.e.*, ${}^{n}\underline{\underline{\mathbf{I}}}^{*}_{n} = {}^{n}\underline{\underline{\mathbf{I}}}_{n}$ and ${}^{n}\underline{\boldsymbol{\beta}}^{*}_{n} = {}^{n}\underline{\boldsymbol{\beta}}_{n} - {}^{n+1}\underline{\underline{\mathbf{X}}}^{T}_{n}\,{}^{n+1}\underline{\mathbf{f}}_{n+1}$ where ${}^{n+1}\underline{\mathbf{f}}_{n+1}$ represents the external force on the manipulator's end-effector.

The power of the articulated-body equations is embedded in Equation 3.36. This equation allows us to treat the whole serial-structure as a sequence of rigid bodies through the calculation of articulated inertias and bias forces. This simplifies the dynamics of serial-manipulators. As it can be seen from the derived equations, articulated-body inertias are a function of only the rigid-body inertias along with the instantaneous kinematics of the connections between the links in serial manipulators. The $i-1^{th}$ articulated-body inertia is felt by joint $i-1$ because of not only its own inertia $\underline{\underline{\mathbf{I}}}_{i-1}$, but also the accumulated inertial effect of the rest of the outboard links $\underline{\underline{\mathbf{N}}}_{i}$. The term $\underline{\underline{\mathbf{N}}}_{i}$ is called the reflected inertia. The reflected inertia is the inertia effect of the outboard links on joint $i-1$. The transformation of the reflected inertia

from link frame i to link frame $i-1$ in Equation 3.35 is the single most significant portion of the computational burden in the articulated-body equations. A method to ease this computational cost has been proposed in [29]. As for the bias force, it captures all the velocity-dependent terms within the articulated body.

3.5 Gravity

The incorporation of the gravity effect can be accomplished by introducing a gravitational force into the force balance equation, Equation 3.19, as follows:

$$ {}^{i}\underline{\mathbf{f}}_i + {}^{i}\underline{\mathbf{f}}_i^g - {}^{i+1}\underline{\mathbf{X}}_i^T \, {}^{i+1}\underline{\mathbf{f}}_{i+1} = {}^{i}\underline{\mathbf{I}}_i \, {}^{i}\underline{\mathbf{a}}_i - {}^{i}\underline{\boldsymbol{\beta}}_i \tag{3.37} $$

where $\underline{\mathbf{f}}_i^g$ is the gravitational force, and is defined as:

$$ {}^{i}\underline{\mathbf{f}}_i^g = {}^{i}\underline{\mathbf{I}}_i \begin{bmatrix} \mathbf{0}_{3\times 1} \\ {}^{i}\mathbf{a}_g \end{bmatrix} = {}^{i}\underline{\mathbf{I}}_i \, {}^{i}\underline{\mathbf{a}}_g \tag{3.38} $$

By taking the gravitational force term to the right side of Equation 3.37 and then grouping terms multiplying the spatial inertia yields:

$$ \begin{aligned} {}^{i}\underline{\mathbf{f}}_i + {}^{i+1}\underline{\mathbf{X}}_i^T \, {}^{i+1}\underline{\mathbf{f}}_{i+1} &= {}^{i}\underline{\mathbf{I}}_i \left({}^{i}\underline{\mathbf{a}}_i - {}^{i}\underline{\mathbf{a}}_g \right) - {}^{i}\underline{\boldsymbol{\beta}}_i \\ &= {}^{i}\underline{\mathbf{I}}_i \, {}^{i}\underline{\acute{\mathbf{a}}}_i - {}^{i}\underline{\boldsymbol{\beta}}_i \end{aligned} \tag{3.39} $$

where $\underline{\acute{\mathbf{a}}}_i$ is called biased acceleration and is defined as:

$$ \underline{\acute{\mathbf{a}}}_i = \underline{\mathbf{a}}_i - \underline{\mathbf{a}}_g \tag{3.40} $$

The biased acceleration will be used in the subsequent sections, as it allows incorporating the gravity effect into the articulated-body equations in a computationally

more efficient way [25]. This can be seen in Equation 3.39. Instead of computing the gravitational force, the biased acceleration can be used through the articulated-body equations. In other words, by using the biased acceleration, the computation required to calculate the gravitational force can be avoided while accounting for the gravitational effect. Therefore, the acceleration propagates according to:

$$ {}^{i}\underline{\acute{\mathbf{a}}}_i = {}^{i}\underline{\mathbf{X}}_{i-1}\ {}^{i-1}\underline{\acute{\mathbf{a}}}_{i-1} + \underline{\phi}_i \ddot{\xi}_i + {}^{i}\underline{\zeta}_i \tag{3.41} $$

Note that if the absolute acceleration is required, the gravitational acceleration must be added to the biased acceleration:

$$ \underline{\mathbf{a}}_i = \underline{\acute{\mathbf{a}}}_i + \underline{\mathbf{a}}_g $$

3.6 Mobile Base (URV)

In the previous chapters, it is assumed that the base is stationary, *i.e.*, the spatial velocity and acceleration of the base (link 0) are set to a constant zero value. However, in URVM systems, the base link is an URV that is not stationary with respect to the inertial frame. Furthermore, the URV's motion is not known. Therefore, the previously derived equations should be extended to account for a mobile-base link.

The dynamic effect of the moving vehicle can be incorporated into the previously derived articulated-body equations by modelling the vehicle as another link (link 0) in the serial-chain as illustrated in Figure 3.7. However, since the URV has a 6 degree of freedom motion contribution to the 0^{th} articulated-body, its contribution is given by a constant 6×6 identity matrix $\underline{\phi}_0$. Every column of this identity matrix

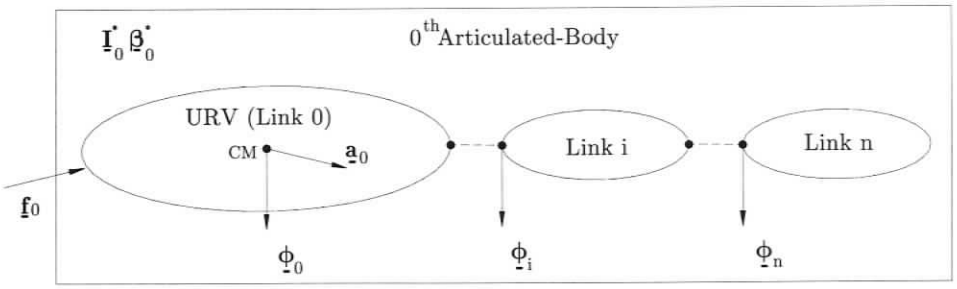


Figure 3.7: URVM System

corresponds to one degree of motion contribution along or about the URV's body-fixed frame. This motion contribution is defined explicitly by the vector $\underline{\mathbf{v}}_0$, which is the URV's spatial velocity vector in terms of the vehicle's body-fixed frame.

$$\underline{\mathbf{v}}_0 = {}^0\underline{\boldsymbol{\phi}}_0 \begin{bmatrix} p \\ q \\ r \\ u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \\ u \\ v \\ w \end{bmatrix} = \begin{bmatrix} p \\ q \\ r \\ u \\ v \\ w \end{bmatrix} \tag{3.42}$$

According to Equation 3.36, the motion of the base can be written as:

$${}^0\underline{\mathbf{f}}_0 = {}^0\underline{\mathbf{I}}_0^* {}^0\underline{\acute{\mathbf{a}}}_0 - {}^0\underline{\boldsymbol{\beta}}_0^* \tag{3.43}$$

In order to simulate the underwater-manipulator system, the accelerations must be calculated. Therefore, Equation 3.43 is solved for acceleration as in [6]:

$${}^0\underline{\acute{\mathbf{a}}}_0 = \left({}^0\underline{\mathbf{I}}_0^*\right)^{-1} \left({}^0\underline{\mathbf{f}}_0 + {}^0\underline{\boldsymbol{\beta}}_0^*\right) \tag{3.44}$$

where $\underline{\mathbf{I}}_0^*$ is the articulated-body of the whole URVM system at the URV's body-fixed frame, $\underline{\mathbf{f}}_0$ is the resultant thruster force at the URV's body-fixed frame, $\underline{\boldsymbol{\beta}}_0^*$ is the bias

force containing all the external forces that the system experiences, excluding gravity, and lastly the term $\underline{\acute{\mathbf{a}}}_0$ is the biased spatial acceleration of the URV. The acceleration of the vehicle is the sum of the biased acceleration and the gravitational acceleration:

$$\underline{\mathbf{a}}_0 = \underline{\acute{\mathbf{a}}}_0 + \underline{\mathbf{a}}_g \tag{3.45}$$

3.7 Extension of the Articulated Body Algorithm to Underwater Vehicle Systems with Dual-Manipulator

The articulated-body algorithm can also be used to define the dynamics of the underwater vehicles with dual-manipulator. Let's consider the URVM system in Figure 1.1. If the robotic arms are expanded to their maximum extension, one can easily see the serial structure of the dual-manipulator URVM system. By numbering the rigid links within the serial-chain of the system starting from the end-effector of the first manipulator (the one on the left of the figure) through the URV and finally all the way to the end-effector of the second manipulator (the one on the right of the figure), the same articulated-body equations can be implemented for a freely floating dual-manipulator URVM system [27].

3.8 Recovery of the Dynamic Coupling Effect

The dynamic coupling effect on the URV is caused by the reactions forces at the mounting point between the URV and the manipulator as illustrated in Figure 3.8. As far as the URV's motion is concerned, this force can be considered to be a force

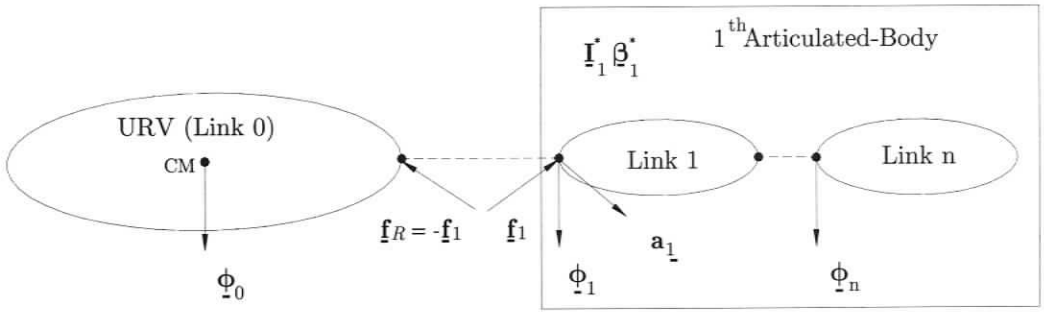


Figure 3.8: Reaction Forces That Causes the Dynamic Coupling Effect

that is exerted by the manipulator on the URV. Referring to Figure 3.8, the force that is experienced by the link 1 in the URVM serial-chain can be given according to the articulated-body equations as:

$$\underline{f}_1 = \underline{I}_1^* \underline{\acute{a}}_1 - \underline{\beta}_1^* \tag{3.46}$$

Therefore, the force exerted by the manipulator on the URV is equal and opposite and can be written as:

$$\underline{f}_R = -\underline{f}_1 \tag{3.47}$$

The moment created by this force is found according to:

$$\underline{N}_R = \begin{bmatrix} 0_{3\times 1} \\ p_{0\to 1} \times \underline{f}_R \end{bmatrix} \tag{3.48}$$

where $p_{0\to 1}$ is the position vector from the center of gravity to the manipulator mounting point on the URV. Finally, the spatial force vector acting on the URV due to the dynamic coupling is given by:

$$\underline{f}_R + \underline{N}_R \tag{3.49}$$

3.9 Summary of Solution of the Forward Dynamic Problem of a Mobile-Base Manipulator System

In this chapter, the main articulated-body equations have been derived. The first step, called forward kinematics, starts with the calculation of velocities ${}^i\boldsymbol{\omega}_i$ and ${}^i\mathbf{v}_i$, using Equation 3.3, and velocity-dependent terms ${}^i\underline{\boldsymbol{\zeta}}_i$ and ${}^i\underline{\boldsymbol{\beta}}_i$ in Equations 3.7 and 3.19 based on the vehicle state vectors, $\underline{\mathbf{x}}$ and $\underline{\mathbf{v}}$, and the manipulator state vectors, $\mathbf{q}$ and $\dot{\mathbf{q}}$. The first step carries out starting from the base ($i = 0$) to the end-effector ($i = n$), and produces the needed terms to calculate the articulated inertias and bias forces. The second step, called backward dynamics, involves the calculation of articulated inertias ${}^i\underline{\mathbf{I}}^*_i$, and bias forces ${}^i\underline{\boldsymbol{\beta}}^*_i$, based on Equation 3.35. As discussed earlier, the articulated inertias and bias forces are computed starting from the end-effector to the base (vehicle). The third step, called forward acceleration, is the calculation of accelerations of the vehicle ${}^0\underline{\acute{\mathbf{a}}}_0$, based on Equation 3.44, and of the manipulator $\ddot{\xi}_i$, based on Equation 3.27. For the URV dynamics, since the unbiased acceleration is needed, the gravitation vector ${}^0\underline{\mathbf{a}}_g$ is added to ${}^0\underline{\acute{\mathbf{a}}}_0$, *i.e.*, ${}^0\underline{\mathbf{a}}_0 = {}^0\underline{\acute{\mathbf{a}}}_0 + {}^0\underline{\mathbf{a}}_g$. The resulting spatial acceleration vector is composed of the second time derivative of Euler angles and the absolute translational accelerations of the vehicle. This is a quick review on the articulated-body dynamic algorithm. More detailed explanations will be given after all the necessary equations are defined.

The main goal of the two previous chapters is to derive the necessary equations to obtain a solution to the forward dynamic problem of underwater-manipulator systems based on the articulated-body dynamic formulation. However, the hydrodynamic

forces inherent in the underwater environment have not been addressed yet. In the next chapter, the most dominant hydrodynamic forces will be defined and then incorporated into the derived equations.

Chapter 4

Forces Acting on an Underwater-Manipulator System

4.1 Overview

In this chapter, the non-linear hydrodynamic forces that incur over the manipulator and the URV are presented. In addition to the hydrodynamic forces, the URV motion is dominated by pilot inputs to propeller type thrusters. In this chapter, a non-linear thruster dynamics modelling technique is introduced and the model is included within the URV dynamics to afford control of the URV motion via thruster voltage inputs. A thruster simulation is performed to analyze the thruster model used.

4.2 Dominant Hydrodynamic Effects

4.2.1 Assumptions

When a rigid body moves in a fluid environment, it is subject to various hydrodynamic effects that are fundamentally nonlinear in nature, which the dominate the rigid-body's dynamic equations. The hydrodynamic effects are added mass, drag and lift, fluid acceleration, buoyancy and the vortex shedding. Vortices may build up behind a body translating in a fluid. These vortices may eventually cause periodic forces [51]. In this work, this effect is assumed to be small enough to neglect. Likewise, the lift forces are also assumed to be small enough to ignore. Therefore, only added mass, drag, fluid acceleration and buoyancy effects are considered in the current work. These nonlinear hydrodynamic effects were found to be dominant in the dynamic simulation of submerged rigid bodies by Yuh [2] and Ioi and Itoh [52]. Lumped approximation to these forces is employed within the articulated-body dynamics equations.

For the derivation of the all of the hydrodynamic forces considered, many assumptions are employed. Firstly, it is assumed that the fluid flow is irrotational. This is a reasonable assumption, as the rotation due to any vortices in the fluid would be negligible in comparison to the rotational motion of URVM systems. In addition, the fluid is assumed to be unbounded and steady. Another assumption is that the added mass coefficients and drag coefficients are constant. In reality, these coefficients vary nonlinearly with respect to velocity and other parameters [53]. However, it is assumed that over the range of operating conditions, these coefficients vary very little and therefore can be considered to be constant. Lumped approximation to

these forces is employed within the articulated-body dynamics equations.

4.2.2 Added Mass

When a submerged body accelerates through a fluid, an additional inertial resistance occurs. The additional inertial effect can be characterized by a 6×6 symmetric, positive definitive matrix. Each component of the added mass matrix is a proportional constant that defines an added mass contribution due to an acceleration in one of the body's degree of freedom. The added mass matrix is defined as [7]:

$$\underline{\mathbf{I}}^A = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \tag{4.1}$$

For completely submerged bodies, the components of the added mass matrix can be assumed to be constant and their approximation can be derived by using the energy approach in terms of Kirchhoff's equations [7]. When the submerged body moves slowly, and three planes of symmetry exist for the body, the contribution of the off-diagonal elements in the added mass matrix can be neglected due to the fact that the diagonal elements are much larger in comparison to their off-diagonal counterparts [7]. This means that the added mass effect in each direction is not dependent on the body's motion in the other directions. In the event that experimental values are not available, the diagonal components of the added mass matrix can be estimated by using strip theory. In this thesis, the vehicle is approximated as a rectangular box and the manipulator links are approximated as cylinders [5], [6], [52]. The derivation of the added mass coefficients using strip theory for the cubic vehicle approximation is made based on the work of Fossen [7] and is demonstrated in Appendix D. It is noted that strip theory is a very crude approximation. In order to obtain better

system simulation results, the experimental values should be used in the dynamic simulations of the URVM systems.

The coefficients for the cylindrical manipulator links are taken from the work of [54] as:

$$\underline{\mathbf{I}}_i^A = diag\left\{ 0 \quad \frac{k_i\ L_i^2}{3} \quad \frac{k_i\ L_i^2}{3} \quad 0 \quad k_i \quad k_i \right\}, \quad k_i = \rho\pi r_i^2 L_i \tag{4.2}$$

where ρ is the density of the surrounding fluid, and r_i and L_i are the radius and length of the i^{th} link, respectively. The added mass effect can be neglected in land-based robotic applications, as the density of the air is negligible in comparison to that of the robotic devices. However, in underwater applications, the difference in density between the water and URVMs is not significant, and therefore this additional inertial resistance must be considered. In Equation 4.2, the first three components represent the added mass force contribution due to the cylinder's rotational motion while the last three components represent the added mass force contribution due to the linear motion of the cylindric body. Note that added mass contribution due to accelerations along the length of the cylinder is assumed to be negligible. This is consistent with an infinitely long cylinder.

Newman [8] derived added mass force and moment equations for a rigid body moving through an unbounded, still water and these expressions have been written in a form consistent with the articulated-body algorithm as proposed by McMillan [25]:

$$\begin{aligned} \underline{\mathbf{f}}_i^A = & -\underline{\mathbf{I}}_i^A \begin{bmatrix} \boldsymbol{\alpha}_i \\ \mathbf{a}_i \end{bmatrix} + \underline{\mathbf{I}}_i^A \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \boldsymbol{\omega}_i \times \mathbf{v}_i \end{bmatrix} - \\ & \begin{bmatrix} \tilde{\boldsymbol{\omega}}_i & \tilde{\mathbf{v}}_i \\ \mathbf{0}_{3\times 3} & \tilde{\boldsymbol{\omega}}_i \end{bmatrix} \underline{\mathbf{I}}_i^A \begin{bmatrix} \boldsymbol{\omega}_i \\ \mathbf{v}_i \end{bmatrix} \qquad i = 0...n \end{aligned} \tag{4.3}$$

where $\underline{\mathbf{I}}_i^A$ is the added mass matrix of the i^{th} link. Note that the terms $\boldsymbol{\alpha}_i$, $\mathbf{a}_i$, $\boldsymbol{\omega}_i$ and $\mathbf{v}_i$ are not the velocity and acceleration of the centre of mass of link i. Rather, they are the velocity and acceleration of joint i as explained in Chapter 3. Equation 4.3 represents a resultant force of distributed added mass forces over the link at joint i.

The effect of the translational velocity and acceleration of the surrounding fluid are accounted for by replacing the terms ${}^i\mathbf{a}_i$ and ${}^i\mathbf{v}_i$ with the corresponding measures:

$$\begin{aligned} \mathbf{a}_i^r &= \mathbf{a}_i - \mathbf{a}^f \\ \mathbf{v}_i^r &= \mathbf{v}_i - \mathbf{v}^f \end{aligned} \tag{4.4}$$

where $\mathbf{v}^f$ and $\mathbf{a}^f$ are the translational velocity and acceleration of the surrounding fluid, respectively. The fluid flow is assumed to be irrotational. For a rigid link moving through an unsteady flow, Equation 4.3 is adjusted to give:

$$\begin{aligned} \underline{\mathbf{f}}_i^A = -\underline{\mathbf{I}}_i^A \begin{bmatrix} \boldsymbol{\alpha}_i \\ \mathbf{a}_i \end{bmatrix} + \underline{\mathbf{I}}_i^A \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \mathbf{a}_f + \boldsymbol{\omega}_i \times \mathbf{v}_i^r \end{bmatrix} \\ - \begin{bmatrix} \tilde{\boldsymbol{\omega}}_i & \tilde{\mathbf{v}}_i^r \\ \mathbf{0}_{3\times 3} & \tilde{\boldsymbol{\omega}}_i \end{bmatrix} \underline{\mathbf{I}}_i^A \begin{bmatrix} \boldsymbol{\omega}_i \\ \mathbf{v}_i^r \end{bmatrix} \qquad i = 0...n \end{aligned} \tag{4.5}$$

Equation 4.5 can be written in a more compact form to facilitate its incorporation into the articulated-body equations as follows:

$$\underline{\mathbf{f}}_i^A = -\underline{\mathbf{I}}_i^A \, \underline{\mathbf{a}}_i + \underline{\boldsymbol{\beta}}_i^A \tag{4.6}$$

where $\underline{\mathbf{f}}_i^A$ is a 6×1 vector expressed in the link-fixed frame combining a 3×1 force and a 3×1 torque vector. The term $\underline{\boldsymbol{\beta}}_i^A$ is a 6×1 bias force vector combining all

velocity dependent terms of the added mass force and is given by:

$$\underline{\boldsymbol{\beta}}_i^A = \underline{\mathbf{I}}_i^A \begin{bmatrix} \mathbf{0}_{3\times1} \\ \mathbf{a}_f + \boldsymbol{\omega}_i \times \mathbf{v}_i^r \end{bmatrix} - \begin{bmatrix} \tilde{\boldsymbol{\omega}}_i & \tilde{\mathbf{v}}_i^r \\ \mathbf{0}_{3\times3} & \tilde{\boldsymbol{\omega}}_i \end{bmatrix} \underline{\mathbf{I}}_i^A \begin{bmatrix} \boldsymbol{\omega}_i \\ \mathbf{v}_i^r \end{bmatrix} \qquad i = 0...n \tag{4.7}$$

4.2.3 Drag and Lift Forces

In a viscous environment, a moving rigid body experiences drag and lift forces. Lift forces are created by differences in pressure distribution over the surface of the body and act in a direction normal to the fluid flow. In this paper, it is assumed that lift and lift-related forces are negligible. Drag force can be described as a resistive force acting on a body that moves through a viscous fluid due to the relative velocity of the fluid with respect to the body. This force can be broken up into two components: frictional drag (this force is also called in literature as linear, shear, skin), which acts along the tangential direction of the surface and depends primarily on the smoothness of the surface of the body, and pressure drag (this force is also called in literature as profile), which acts in an opposite direction to the link's velocity relative to the fluid and depends mainly on the frontal area of the body. Frictional drag is assumed to be small and is neglected. The computation of drag forces for the URV and manipulator have been conducted separately.

Vehicle Translational and Rotational Drag

Translational drag force incurred along a particular axis of the URV body fixed frame can be given by:

$$f_{0_j}^D = -\frac{1}{2}\rho A_j C_{0_j}^D \; \mathbf{v}_{0,\,j}^r \; ||\mathbf{v}_0^r|| \qquad j = 4...6 \tag{4.8}$$

where ρ is the density of the surrounding fluid and subscript j indicates a particular DOF of the vehicle. For example, $j \in \{1,\ 2,\ 3\}$ and $j \in \{4,\ 5,\ 6\}$ represents rotations about and translations along the $\mathbf{x}_0$, $\mathbf{y}_0$ and $\mathbf{z}_0$ axes of the URV's body-fixed frame or the link 0 frame, respectively. With regards to the remaining terms, A_j is the frontal area of the URV face perpendicular to the j direction, $\mathbf{v}_0^r$ is the relative velocity with respect to the fluid, and $C_{0_j}^D$ is the drag coefficient attributed to the perpendicular face. The drag coefficients for the translational directions can be estimated to be similar to the empirically determined drag coefficients for a cube or rectangular plate. This approach is employed to the URV. Referring to [55] the coefficient of drag was estimated to be 1.07 for each direction. The characteristic area is taken as the area of the face perpendicular to the fluid flow. It is assumed that drag forces act at the centre of pressure[1] of each face of the vehicle.

To compute rotational drag effects, the URV is again approximated as a rectangular box. Each side length of the rectangular box is divided into small strips starting at the central axis of the rotational motion and moving outward as illustrated in Figure 4.1. The drag force on the face of each strip is calculated using Equation 4.8 based on the strips velocity relative to the surrounding fluid. Two-dimensional drag coefficients are found by interpolating the data presented in [55] using strip dimensions. The collective moment of these translational drag forces over the entire body can be obtained by integrating the vector product of each strip's distance from the central axis r_{strip} and the translational drag force accumulated over the strip, df_{strip}^D.

[1]The centre of pressure is a point on the URV face that lies under the centroid of the distributed hydrodynamic load.

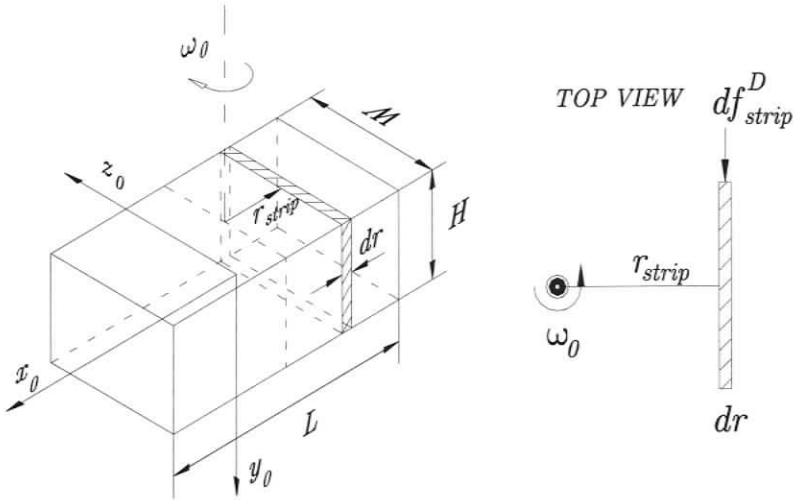


Figure 4.1: URV is Divided into Strips.

Rotational drag with respect to the vehicle's centre of mass is given by:

$$n^D_{0_j} = 2 \int r_{strip}\, df^D_{strip} \qquad j = 1...3 \tag{4.9}$$

The spatial drag force acting on the URV is given by:

$$\underline{\mathbf{f}}^D_0 = \begin{bmatrix} n^D_{0_1} & n^D_{0_2} & n^D_{0_3} & f^D_{0_4} & f^D_{0_5} & f^D_{0_6} \end{bmatrix}^T \tag{4.10}$$

Translational and Rotational Drag Acting on a Manipulator Link

The manipulator links are approximated as a series of circular strips as illustrated in Figure 4.2. The resultant drag force f^D_i and moment exerted n^D_i onto the i^{th} link can be found by integrating the drag force and moment experienced on each strip

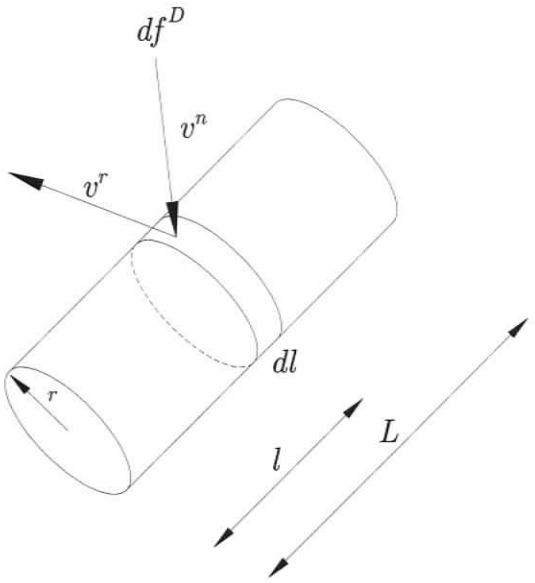


Figure 4.2: A Cylinder is Composed of a Series of Circular Strips

over the length of the cylindrical link [56],

$$\underline{\mathbf{f}}_i^D = \begin{bmatrix} \mathbf{n}_i^D \\ \mathbf{f}_i^D \end{bmatrix} = \begin{bmatrix} -0.5\rho C_i^D \int_0^{L_i} \|\mathbf{v}_i^n\| \left(\begin{bmatrix} l_i & 0 & 0 \end{bmatrix}^T \times \mathbf{v}_i^n \right) 2r_i \cdot dl_i \\ -0.5\rho C_i^D \int_0^{L_i} \|\mathbf{v}_i^n\| \ \mathbf{v}_i^n \ 2r_i \cdot dl_i \end{bmatrix} \qquad i = 1...n \tag{4.11}$$

where r_i and L_i are the radius and length of the cylindrical link i, respectively, and $\mathbf{v}_i^n$ is the normal velocity of the link i at the location l_i relative to the surrounding fluid.

4.2.4 Buoyancy and Fluid Acceleration Forces

The buoyant force of a body opposes gravity and is equal to the weight of the displaced fluid. The fluid acceleration force is proportional to the acceleration of the fluid.

These two forces are combined in the following calculation:

$$\mathbf{f}_i^{TB} = m_i^f \left(\mathbf{a}_f - \mathbf{a}_g\right) \qquad i = 0...n \tag{4.12}$$

where m_i^f is the mass of displaced fluid by the i^{th} link. The buoyancy and fluid acceleration forces are exerted at the body's centre of buoyancy. The spatial force is given by:

$$\underline{\mathbf{f}}_i^{TB} = \begin{bmatrix} \mathbf{b}_i \times \mathbf{f}_i^{TB} \\ \mathbf{f}_i^{TB} \end{bmatrix} \qquad i = 0...n \tag{4.13}$$

where ${}^i\mathbf{b}_i$ is the position vector from the body-fixed frame to the centre of buoyancy of the body.

4.2.5 Incorporation of the Hydrodynamic Forces into the Articulated-Body Equations

In order to derive URVM dynamics algorithm, the hydrodynamic force equations for the rectangular URV and the cylindrical manipulator links must be incorporated into the articulated-body equations.

The collective effect of these forces can be found by superimposing each force individually. Therefore, the force spatial balance equation for a single rigid body that is submerged in a fluid, including the gravitational force, is given in terms of its body-fixed frame as:

$${}^i\underline{\mathbf{f}}_i - {}^{i+1}\underline{\mathbf{X}}_i^T \, {}^{i+1}\underline{\mathbf{f}}_{i+1} + {}^i\underline{\mathbf{f}}_i^A + {}^i\underline{\mathbf{f}}_i^D + {}^i\underline{\mathbf{f}}_i^{TB} = {}^i\underline{\mathbf{I}}_i \, {}^i\underline{\acute{\mathbf{a}}}_i - {}^i\underline{\boldsymbol{\beta}}_i \tag{4.14}$$

where ${}^i\underline{\acute{\mathbf{a}}}_i = {}^i\underline{\mathbf{a}}_i - {}^i\underline{\mathbf{a}}_g$. Substituting the added mass force, Equation 4.6, into Equation 4.14, and grouping terms yields:

$${}^i\underline{\mathbf{f}}_i - {}^{i+1}\underline{\mathbf{X}}_i^T \, {}^{i+1}\underline{\mathbf{f}}_{i+1} = {}^i\underline{\mathbf{I}}_i^A \, {}^i\underline{\mathbf{a}}_i + {}^i\underline{\mathbf{I}}_i \, {}^i\underline{\acute{\mathbf{a}}}_i - {}^i\underline{\boldsymbol{\beta}}_i - {}^i\underline{\boldsymbol{\beta}}_i^A - {}^i\underline{\mathbf{f}}_i^D - {}^i\underline{\mathbf{f}} \tag{4.15}$$

In order to express accelerations in terms of the biased accelerations, the absolute (unbiased) acceleration term ${}^{i}\underline{\mathbf{a}}_i$ is replaced by ${}^{i}\underline{\acute{\mathbf{a}}}_i + {}^{i}\underline{\mathbf{a}}_g$. This step results in the following equation:

$$ {}^{i}\underline{\mathbf{f}}_i - {}^{i+1}\underline{\mathbf{X}}_i^T \, {}^{i+1}\underline{\mathbf{f}}_{i+1} = {}^{i}\underline{\mathbf{I}}_i^A \left({}^{i}\underline{\acute{\mathbf{a}}}_i + {}^{i}\underline{\mathbf{a}}_g \right) + {}^{i}\underline{\mathbf{I}}_i \, {}^{i}\underline{\acute{\mathbf{a}}}_i - {}^{i}\underline{\boldsymbol{\beta}}_i - {}^{i}\underline{\boldsymbol{\beta}}_i^A - {}^{i}\underline{\mathbf{f}}_i^D - {}^{i}\underline{\mathbf{f}}_i^{TB} \quad (4.16) $$

In Equation 4.16, the terms multiplying ${}^{i}\underline{\acute{\mathbf{a}}}_i$ can be grouped together. The remaining term ${}^{i}\mathbf{I}_i^A \, {}^{i}\underline{\mathbf{a}}_g$ can be inserted into the term ${}^{i}\underline{\boldsymbol{\beta}}_i^A$. Therefore, the following equation is obtained:

$$ {}^{i}\underline{\mathbf{f}}_i - {}^{i+1}\underline{\mathbf{X}}_i^T \, {}^{i+1}\underline{\mathbf{f}}_{i+1} = \left({}^{i}\underline{\mathbf{I}}_i + {}^{i}\underline{\mathbf{I}}_i^A \right) {}^{i}\underline{\acute{\mathbf{a}}}_i - {}^{i}\underline{\boldsymbol{\beta}}_i - {}^{i}\underline{\boldsymbol{\beta}}_i^{\acute{A}} - {}^{i}\underline{\mathbf{f}}_i^D - {}^{i}\underline{\mathbf{f}}_i^{TB} \quad (4.17) $$

where ${}^{i}\underline{\boldsymbol{\beta}}_i^{\acute{A}}$ is given as:

$$ {}^{i}\underline{\boldsymbol{\beta}}_i^{\acute{A}} = {}^{i}\underline{\mathbf{I}}_i^A \begin{bmatrix} \mathbf{0}_{3\times 1} \\ (\, {}^{i}\mathbf{a}_f - {}^{i}\mathbf{a}_g \,) + {}^{i}\boldsymbol{\omega}_i \times {}^{i}\mathbf{v}_i^r \end{bmatrix} - \begin{bmatrix} {}^{i}\tilde{\boldsymbol{\omega}}_i & {}^{i}\tilde{\mathbf{v}}_i^r \\ \mathbf{0}_{3\times 3} & {}^{i}\tilde{\boldsymbol{\omega}}_i \end{bmatrix} {}^{i}\underline{\mathbf{I}}_i^A \begin{bmatrix} {}^{i}\boldsymbol{\omega}_i \\ {}^{i}\mathbf{v}_i^r \end{bmatrix} \quad (4.18) $$

Rewriting Equation 4.17 in the form of Equation 3.19 yields:

$$ {}^{i}\underline{\mathbf{f}}_i - {}^{i+1}\underline{\mathbf{X}}_i^T \, {}^{i+1}\underline{\mathbf{f}}_{i+1} = {}^{i}\underline{\mathbf{I}}_i^H \, {}^{i}\underline{\acute{\mathbf{a}}}_i - {}^{i}\underline{\boldsymbol{\beta}}_i^H \quad (4.19) $$

where

$$ {}^{i}\underline{\mathbf{I}}_i^H = {}^{i}\underline{\mathbf{I}}_i + {}^{i}\underline{\mathbf{I}}_i^A \quad (4.20) $$

$$ {}^{i}\underline{\boldsymbol{\beta}}_i^H = {}^{i}\underline{\boldsymbol{\beta}}_i + {}^{i}\underline{\boldsymbol{\beta}}_i^{\acute{A}} + {}^{i}\underline{\mathbf{f}}_i^D + {}^{i}\underline{\mathbf{f}}_i^{TB} \quad (4.21) $$

The terms $\underline{\mathbf{I}}_i^H$ and $\underline{\boldsymbol{\beta}}_i^H$ are called hydrodynamic inertia and hydrodynamic bias acceleration, respectively [6]. Equation 4.19 is the spatial force-balance equation that is used in the articulated-body equations.

4.3 Thruster Dynamics

4.3.1 Assumed Model

It is shown in [14] that for a successful hovering and low-speed trajectory of URVs, thruster dynamics dominate the URV dynamics. Therefore, in order to be able to precisely predict the hovering and low-speed trajectory of the URVM systems, the non-linear thruster dynamics must be accounted for in the dynamic simulation URVMs.

In this thesis, the work of Healey *et al.* [16] is followed. Healey *et al.* [16] combined an armature controlled motor model, a theoretical propeller mapping using airfoil theory, and a fluid momentum model applied within the finite volume of the shrouded region. Healey *et al.* [16] only considered the axial component of the fluid flow and used sinusoidal lift/drag curves, which suits to our needs well as we consider a hypothetical URVM structure for which experimental thruster data is not available.

4.3.2 Motor Modelling

Healey *et al.*'s [16] model uses a standard dc servo-motor permanent magnet type of motor whose control signal is voltage. The differential equation for the armature current i_a inside the motor is governed by the following equation [57]:

$$l \frac{di_a}{dt} + r\, i_a = v_a - v_b \tag{4.22}$$

where l is the armature inductance, r is the armature resistance, v_a is the armature voltage, and lastly v_b is the back emf of the motor. The torque τ_m developed by the

motor is given by:

$$\tau_m = k_m \, i_a \tag{4.23}$$

where k_m is the torque constant in N-m/amp. The back emf voltage is given by the following equation:

$$v_b = k_b \, \Omega \tag{4.24}$$

where k_b and Ω are the back emf constant and the angular velocity of the rotor, respectively. Assuming that the motor inductance is negligible Equation 4.22 simplifies to:

$$r \, i_a = v_a - v_b \tag{4.25}$$

Substituting Equation 4.24 into Equation 4.25 yields the following equation for steady-state armature current:

$$i_a = \frac{v_a - k_b \, \Omega}{r} \tag{4.26}$$

In order to model the motor dynamics, the shaft dynamics must be considered. While the thrusters operate in the underwater environment, the water medium around the thruster's propellers creates hydrodynamic loading on the propeller blades and in turn on the motor shaft. This hydrodynamic load results from the lift and drag forces that develop on the propeller blades. The development of the hydrodynamic loading will be explained in Section 4.3.3. The equation of motion of a motor under hydrodynamic loading τ_L is given by:

$$j_m \, \dot{\Omega} + b_m \, \Omega = \tau_m - \tau_L \tag{4.27}$$

where j_m is the motor inertia, $\dot{\Omega}$ is the angular acceleration of the rotor, and lastly b_m is the motor damping constant. Substituting Equations 4.23 and 4.26 into Equation 4.27 respectively, yields the following equation for the rotational acceleration of the rotor in terms of the commanded armature voltage, v_a:

$$\dot{\Omega} = -j_m^{-1} \left[b_m + r^{-1} k_m \, k_b\right] \, \Omega + r^{-1} \, j_m^{-1} \, k_m \, v_a - j_m^{-1} \, \tau_L \tag{4.28}$$

Making the following definitions:

$$\begin{aligned} k_1 &= j_m^{-1} \left[b_m + r^{-1} k_m \, k_b\right] \\ k_2 &= r^{-1} \, j_m^{-1} \, k_m \\ k_h &= j_m^{-1} \end{aligned} \tag{4.29}$$

the motor dynamics to be expressed as:

$$\dot{\Omega} = -k_1 \Omega + k_2 \, v_a - k_h \tau_L \tag{4.30}$$

k_1 and k_2 relates the motor shaft acceleration to the back emf developed in the motor and the motor torque to the current or voltage applied to the motor, respectively. The term k_h is the parameter accounting for motor deceleration caused by hydrodynamic torque loading, and can be approximated as the inverse of the motor shaft inertia.

4.3.3 Hydrodynamic Propeller Loads

In order to find the thrust and torque generated by the fluid flow over the propeller blades, Healey *et al.* [16] applied airfoil theory to the propeller. His theory involves estimating the relative direction of the incoming fluid particles, which is quantified by the effective angle of attack α_e. This effective angle of attack is the angle between

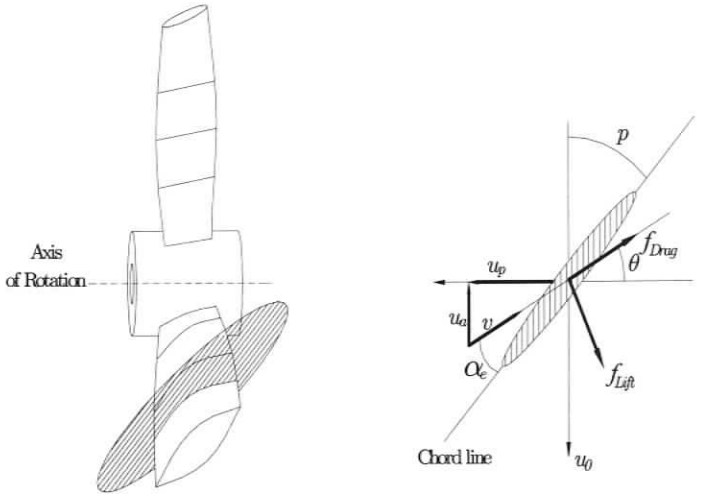


Figure 4.3: Airfoil for a Propeller Section

the relative velocity vector of the incoming particles v at the propeller blade and the chord line of the airfoil as illustrated in Figure 4.3.

In Figure 4.3, v is dependent on the velocity of the axial fluid passing through the propeller shroud u_a, and the translational velocity of the propeller blade u_p at a radial position of 0.7 R, where R is the radius of the propeller.

An airfoil's hydrodynamic force can be separated into lift and drag components that intersect with its chord line at its centre of pressure. As illustrated in Figure 4.3, lift force is developed on the propeller blade perpendicular to the instantaneous line of action of the incoming fluid path while the drag force is developed along the direction of the flow. Calculation of the drag and lift forces requires the effective angle of attack along with squared magnitude of the velocity of incoming fluid particles. The squared magnitude of velocity v is given by the following equation:

$$v^2 = u_p^2 + u_a^2 \tag{4.31}$$

where, with n_{gr} representing the gear reduction ratio,

$$u_p = 0.7\ r\ \Omega\ /\ n_{gr} \tag{4.32}$$

Using the sense of u_a and u_p, the effective angle of attack is defined over all four quadrants of α_e as illustrated in Figure 4.4.

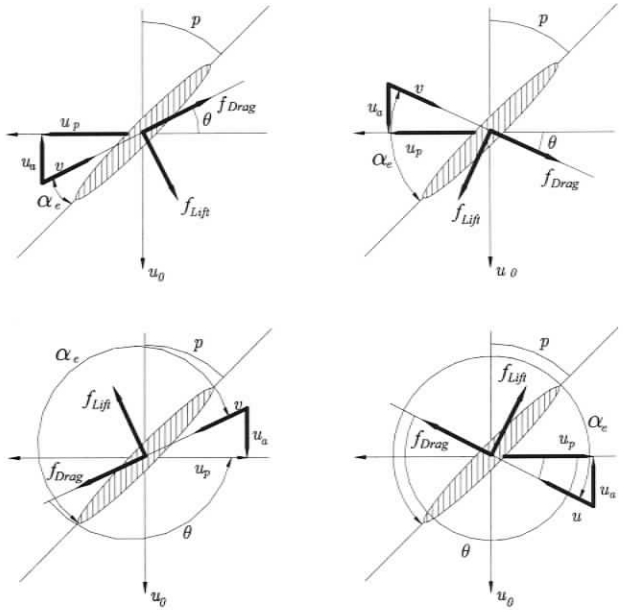


Figure 4.4: Four Quadrant Mapping

As it can be seen from Figure 4.4 that the effective angle of attack for quadrant 1 and 2 is given by:

$$\alpha_e = (\pi/2 - p) - \arctan\left(\frac{u_a}{u_p}\right) \tag{4.33}$$

where p represents the pitch angle, which is unique to the propeller blade. The effective angle of attack for quadrant 3 and 4 is given by:

$$\alpha_e = \left(\frac{3\pi}{2} - p\right) - \arctan\left(\frac{u_a}{u_p}\right) \tag{4.34}$$

Lift and drag forces are proportional to dimensionless lift and drag coefficients that are themselves a function of α_e and of the propeller blade geometry. Healey *et al.* [16] proposed that the lift and drag coefficients be obtained using sinusoidal lift and drag curves over the domain α_e. Using this approach, only the maximum drag coefficient $c_{D\max}$ and the maximum lift coefficient $c_{L\max}$ are required to establish these sinusoidal variations of the drag and lift coefficients.

Having found the angle of attack, the resultant lift and drag forces are obtained from the following equation:

$$f_{Lift} = 0.5\rho\, v^2\, a\, c_{L\max} \sin(2\alpha_e) \tag{4.35}$$

$$f_{Drag} = 0.5\rho\, v^2\, a\, c_{D\max} (1 - \cos(2\alpha_e)) \tag{4.36}$$

where a is the tunnel cross-sectional area. The axial thrust force and torque on the motor can be found by resolving the lift and drag forces using the following rotation matrix:

$$\mathbf{R}_\theta = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{4.37}$$

where the angle θ is determined for all quadrants by:

$$\theta = \frac{\pi}{2} - p - \alpha_e \tag{4.38}$$

The output thrust τ_h, and force on the propeller f_p, can be determined by:

$$\begin{bmatrix} \tau_h \\ f_p \end{bmatrix} = R_\theta \begin{bmatrix} f_{Lift} \\ f_{Drag} \end{bmatrix} \tag{4.39}$$

The torque load on the motor can be found using:

$$\tau_L = 0.7r\ f_p \tag{4.40}$$

This torque acts as a hydrodynamic loading on the rotor shaft and can be substituted into the motor shaft dynamics equation; Equation 4.30.

4.3.4 Fluid Modelling

Healey *et al.* [16] connected the motor model and the propeller maps by applying the momentum equation to a control volume enveloping the propeller shroud. The changing flow velocity through the control volume is governed by:

$$\dot{u}_a = -k_4 k_3^{-1} \bar{u}_a \left| \bar{u}_a \right| + k_3^{-1} \tau_h \tag{4.41}$$

where

$$k_3 = \rho \; a \; l_\gamma \tag{4.42}$$

$$k_4 = \rho \; a \; \Delta\beta \tag{4.43}$$

$$\bar{u}_a = (u_a - u_0) \tag{4.44}$$

with u_0 being the velocity of propeller shroud due to the rigid-body motion of the URV, and l_γ and $\Delta\beta$ being experimentally determined coefficients.

Consequently, the collective state of thrusters is defined by the two state vectors:

$$\Omega = \begin{bmatrix} \Omega_1 & \Omega_2 & \dots & \Omega_h \end{bmatrix}^T, \; \mathbf{U} = \begin{bmatrix} u_1 & u_2 & \dots & u_h \end{bmatrix}^T \tag{4.45}$$

where h is the number of thrusters.

4.3.5 Thruster Simulations

Having found the rotation rate of the propeller's rotor using Equation 4.30 and the fluid rate passing through the shroud using Equation 4.41, the state vector at a

later time can be obtained through integrating these terms forward in time. The corresponding thrust can be recovered from the state vector by applying Equation 4.31 through Equation 4.39 based on the new state vector.

A time-domain simulation program based on the above thruster modelling approach was created using MatLab®'s numerical integration command *ode45*. This simulation program is used solely to test the thruster dynamics and allows the examination of the thruster dynamic modelling technique's capability of capturing the known overshoot in thrust and the time lag in changes in u_a.

The following graphs illustrate the time history of a thruster's state. The initial state values $\Omega_1 = 0$ and $u_1 = 0$ were used and the voltage v_a is stepped up to 10V. In the simulation, the vehicle velocity u_0 is also considered to be zero, and all the associated simulation parameters are taken from the work of Healey *et al.* [16] are provided in Appendix E

When the voltage is stepped, the propeller rotation Ω is subject to the shaft inertia and the inertia of the fluid in the shroud and so a time lag is seen in the transient response of the rotor.

The experimental results in [58] exhibit close correspondence to the experimental result of Figure 4.5. Likewise, it is also shown in [58] that the thruster modelling technique of Healey *et al.* [16] accurately predicts the steady-state behavior of the rotor.

As can be seen in Figure 4.6, the model captures the overshoot in the thrust transient response, which is experienced by most typical thrusters [16]. This result is the product of fluid inertial effects on the propeller blade when the interaction between the surrounding fluid medium and the propeller blade motion occur.

The other dynamic characteristic of thrusters is the time lag in the development

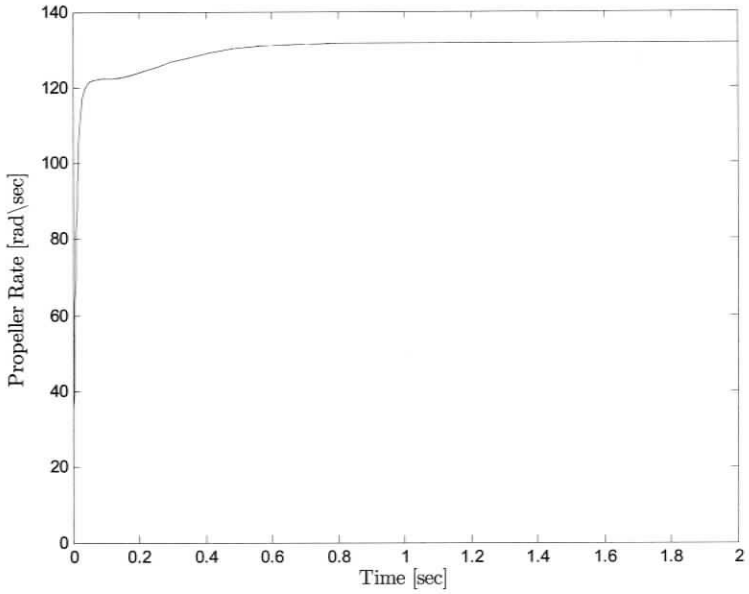


Figure 4.5: Time History of Propeller Rate

of changes in the axial component of the flow. In Figure 4.7, time history of the incoming fluid history is demonstrated. As it can be seen in Figure 4.7, the time lag is captured by the dynamic model. This time lag is due to the first-order nonlinear nature of the incoming fluid in view of Equation 4.41 and is consistent with the experimental results obtained in [16].

Thrust generally decreases as the propeller advances through the water [16]. The dynamic model of the thruster is also successful in demonstrating this nature of typical thrusters as illustrated in Figure 4.8.

It is concluded that thruster dynamics inherently involves time lags due to its nonlinear nature and in turn imposes bandwidth constraint.

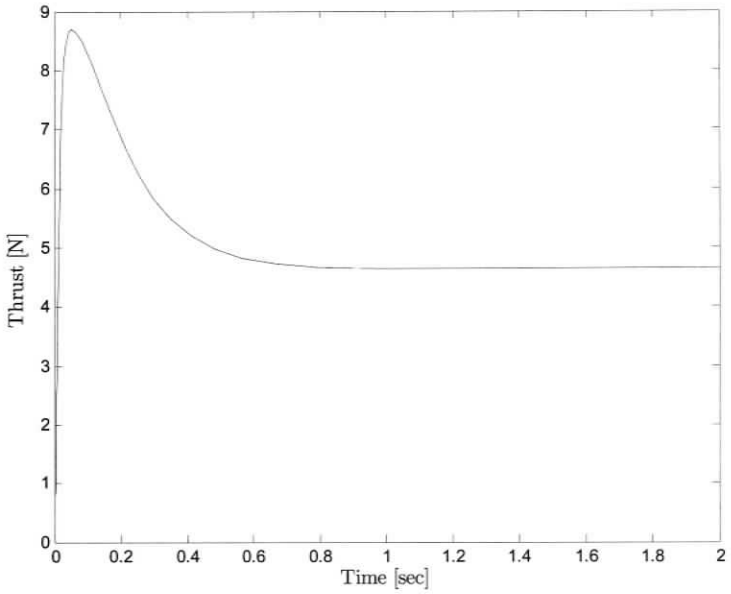


Figure 4.6: Time History of Thrust

4.3.6 Thruster Mapping

Thrusters are usually used to control the motion of URVM systems. URVM controllers usually give required forces and moments with respect to a point (usually the URV's centre of mass) on the URV. However, since the thrusters are placed at the specific positions on the URV, these required forces and moments must be distributed over the thrusters of the URVM system

The commanded forces and moments ${}^0\underline{\mathbf{f}}_0$ at a point is translated into thruster space thrust through the mapping matrix. For a URV having h thrusters, the corresponding relationship between the commanded forces/torques and the thrust is given by:

$$ {}^0\underline{\mathbf{f}}_0 = \Lambda \begin{bmatrix} \tau_1 & \tau_2 & \dots & \tau_h \end{bmatrix}^T \tag{4.46} $$

where Λ is a $6 \times h$ thruster mapping matrix, which is dependent on the thruster

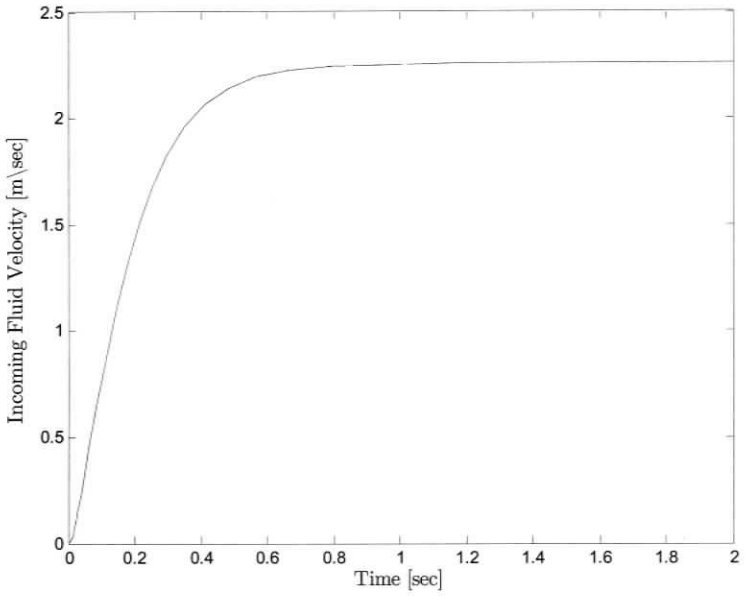


Figure 4.7: Time History of Incoming Fluid Velocity

position and orientation on the URV, and τ_h is the commanded thrust for the h^{th} thruster. Commanded thrust τ_h is fed into the thruster controller as an input thrust. Output thrusts are translated back into the forces and moments at the centre of mass using Equation 4.46.

If the matrix Λ is square or low rectangular, *i.e.*, $h \geq 6$, then the URV's 6 degrees of motion can be controlled fully. In case of $h = 6$, there will be only one solution to the distribution problem of thrusters. However, in case of $h > 6$, there will be a infinite number of solutions, and therefore the problem of optimum distribution of forces over the thrusters arises. In case of $h < 6$ then the URV's 6 degrees of motion can not be controlled. In case there is no thruster controller, the thrusts are translated into the forces and moments at the center of mass directly from Equation 4.46.

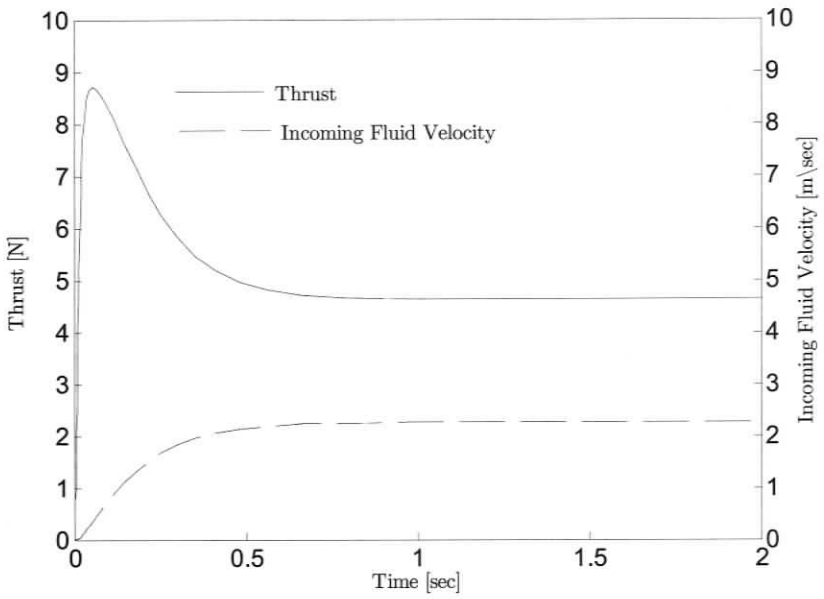


Figure 4.8: Time History of Thrust and Incoming Fluid Velocity

4.3.7 Incorporation of the Thruster Dynamics in the Articulated-Body Equations

For a given state vectors of Equation 4.45, Equations 4.31 through 4.40 are used to obtain the corresponding thrust for each thruster. After this step, the forces and moments at the centre of mass produced by thrusters are attained by using Equation 4.46. These resulting forces and moments are incorporated into the articulated-body equations by substituting ${}^0\underline{\mathbf{f}}_0$ into Equation 3.44. The next thruster state-vector values are obtained by numerically integrating Equations 4.27 and 4.41 forward in time.

Chapter 5

Complete Articulated-Body Algorithm for the Time-Domain Simulation of URVM Systems and Case Studies

5.1 Overview

In Chapter 3, basic articulated-body equations for a floating-base manipulator have been derived. In Chapter 4, the most important hydrodynamic effects experienced by the URVM systems along with the thruster dynamics have been addressed and their incorporation into the articulated-body equations was presented. In this chapter, a complete articulated-body algorithm for the time-domain simulation of URVM systems will be presented. The solution procedure will be laid out step by step with reference to the equations derived in Chapters 3 and 4. Finally, the problem

of dynamic coupling present in URVM systems will be demonstrated based on the simulation of a small URV with a 3-DOF manipulator.

5.2 Articulated-Body Algorithm for the Time-Domain Simulation of URVM Systems

5.2.1 Overview

In this section, the articulated-body algorithm for URVM system proposed by McMillan *et al.*, [6] is extended to include the thruster dynamics. In contrast to the work of McMillan *et al.*, [6], the state variables for the URV are given in terms of the inertial frame. The complete hydrodynamic articulated-body algorithm is comprised of three steps: forward kinematics, backward dynamics and forward acceleration. Each individual step along with the relevant equation are presented below:

5.2.2 Forward Kinematics:

The Articulated-body algorithm consists of three steps based on the URVM state vector $\mathbf{u}_s = \begin{bmatrix} {}^{E}\underline{\mathbf{x}}_0^T & {}^{E}\underline{\mathbf{v}}_0^T & \mathbf{q}^T & \dot{\mathbf{q}}^T & \Omega^T & \mathbf{u}^T \end{bmatrix}^T$, which is formed by the URV's state vectors ${}^{E}\underline{\mathbf{x}}_0^T$ and ${}^{E}\underline{\mathbf{v}}_0^T$, the manipulator's state vectors $\mathbf{q}^T$, $\dot{\mathbf{q}}^T$, and finally the thruster's state vectors $\Omega^T, \mathbf{u}^T$. The first step, called forward kinematics, is the calculation of the velocity-dependent terms from the URV to the end-effector. The forward kinematics step starts computing the values for ${}^{0}\mathbf{v}_0^r$ and ${}^{0}\mathbf{a}^{f-g}$, which are the URV's body fixed velocity and the combined acceleration with respect to the fluid, respectively. In order to perform this calculation, the rotation matrix between the inertial frame and the URV's body-fixed frame ${}^{0}_{E}\mathbf{R}$, the fluid's velocity ${}^{E}\mathbf{v}^f$, the fluid's acceleration

${}^{E}\mathbf{a}^{f}$, and lastly the gravitational acceleration ${}^{E}\mathbf{a}^{g}$ in terms of the inertial frame must be given. Transformation matrix ${}^{0}_{E}\mathbf{R}$ can be found by using Equation 2.3 based on the URV's orientation. However, this is not the only representation of the URV's orientation with respect to the inertial frame, as there are more than one Euler sequence of rotations resulting in different transformation matrices. Given ${}^{0}_{E}\mathbf{R}$, ${}^{E}\mathbf{v}^{f}$,${}^{E}\mathbf{a}^{f}$ and ${}^{E}\mathbf{a}^{g}$, ${}^{0}\mathbf{v}^{r}_{0}$ and ${}^{0}\mathbf{a}^{f-g}$ are computed using Equations 5.1 through 5.3. The URV's hydrodynamic bias force, ${}^{0}\underline{\boldsymbol{\beta}}^{H}_{0} = {}^{0}\underline{\boldsymbol{\beta}}_{0} + {}^{0}\underline{\boldsymbol{\beta}}^{\acute{A}}_{0} + {}^{0}\underline{\mathbf{f}}^{D}_{0} + {}^{0}\underline{\mathbf{f}}^{TB}_{0}$, contains the velocity-dependent terms of the added mass forces (Equation 4.5), the drag forces (Equation 4.10), and the fluid acceleration and buoyancy forces (Equation 4.13), for the URV and is given in Equations 5.4 through 5.5 in the complete articulated-body algorithm.

In order to perform the calculations associated with the manipulator dynamics, it is required to define the rotation matrices ${}^{i}_{i-1}\mathbf{R}$ along the serial-chain of the manipulator as well as the position vectors ${}^{i-1}\mathbf{p}_{i-1\rightarrow i}$ from link to link. The terms ${}^{i}_{i-1}\mathbf{R}$ and ${}^{i-1}\mathbf{p}_{i-1\rightarrow i}$ can be obtained from Equation 2.24, which is derived based on Craig's Denavit-Hartenberg parameters [48]. Note that for prismatic joints ${}^{i-1}\mathbf{p}_{i-1\rightarrow i}$ is a function of $\dot{\xi}_i$, and ${}^{i}_{i-1}\mathbf{R}$ is constant and can be computed off-line for the sake of computational efficiency. Likewise, for revolute joints, ${}^{i}_{i-1}\mathbf{R}$ is a function of $\dot{\xi}_i$, and vector ${}^{i-1}\mathbf{p}_{i-1\rightarrow i}$ can be computed off-line. Specifying ${}^{i}_{i-1}\mathbf{R}$ and ${}^{i-1}\mathbf{p}_{i-1\rightarrow i}$, the spatial transformation matrix ${}^{i}_{i-1}\mathbf{X}$ can be formed and is substituted in Equation 5.6, which is used to propagate the link velocities starting from link 1 to n.

The manipulator's vector of Coriolis and centripetal accelerations ${}^{i}\underline{\boldsymbol{\zeta}}_{i}$ are computed from link $i = 1$ to link n in Equation 5.8a. Likewise, the hydrodynamic bias forces ${}^{i}\underline{\boldsymbol{\beta}}^{H}_{i} = {}^{i}\underline{\boldsymbol{\beta}}_{i} + {}^{i}\underline{\boldsymbol{\beta}}^{\acute{A}}_{i} + {}^{i}\underline{\mathbf{f}}^{D}_{i} + {}^{i}\underline{\mathbf{f}}^{TB}_{i}$ containing the velocity-dependent terms of the added mass forces (Equation 4.5), the drag forces (Equation 4.11), and fluid acceleration and

buoyancy forces (Equation 4.13), for the manipulator are computed using Equations 5.9 through 5.10 in the complete articulated-body algorithm. The equations used in the forward kinematics step are presented below.

Forward Kinematics:

For a given URVM state vector $\mathbf{u}_s = \begin{bmatrix} {}^E\underline{\mathbf{x}}_0^T & {}^E\underline{\mathbf{v}}_0^T & \mathbf{q}^T & \dot{\mathbf{q}}^T & \Omega^T & \mathbf{u}^T \end{bmatrix}^T$ and parameters ${}^E\mathbf{v}^f$,${}^E\mathbf{a}^f$,

$$
\begin{aligned}
{}^0\mathbf{v}_0^r &= {}^0\mathbf{v}_0 - {}^0_E\mathbf{R}\ {}^E\mathbf{v}^f && (5.1)\\
{}^0\mathbf{a}^g &= {}^0_E\mathbf{R}\ {}^E\mathbf{a}^g \qquad \text{where } {}^E\mathbf{a}^g = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T && (5.2)\\
{}^0\mathbf{a}^{f-g} &= {}^0_E\mathbf{R}\ \left({}^E\mathbf{a}^f - {}^E\mathbf{a}^g \right) && (5.3)
\end{aligned}
$$

URV Kinematics:

$$
{}^0\underline{\boldsymbol{\beta}}_0 = -\begin{bmatrix} {}^0\boldsymbol{\omega}_0 \times {}^0\mathbf{I}_0\ {}^0\boldsymbol{\omega}_0 \\ {}^0\boldsymbol{\omega}_0 \times ({}^0\boldsymbol{\omega}_0 \times {}^0\mathbf{h}_0) \end{bmatrix} \tag{5.4}
$$

$$
\begin{aligned}
{}^0\underline{\boldsymbol{\beta}}_0^H = {}^0\underline{\boldsymbol{\beta}}_0 + {}^0\underline{\mathbf{I}}_0^A \begin{bmatrix} \mathbf{0}_{3\times1} \\ ({}^0\mathbf{a}_f - {}^0\mathbf{a}_g) + {}^0\boldsymbol{\omega}_0 \times {}^0\mathbf{v}_0^r \end{bmatrix} - \qquad (5.5)\\
\begin{bmatrix} {}^0\tilde{\boldsymbol{\omega}}_0 & {}^0\tilde{\mathbf{v}}_0^r \\ \mathbf{0}_{3\times3} & {}^0\tilde{\boldsymbol{\omega}}_0 \end{bmatrix} {}^0\underline{\mathbf{I}}_0^A \begin{bmatrix} {}^0\boldsymbol{\omega}_0 \\ {}^0\mathbf{v}_0^r \end{bmatrix} + {}^0\underline{\mathbf{f}}_0^D \left({}^0\boldsymbol{\omega}_0, {}^0\mathbf{v}_0^r \right) + m_0^f \begin{bmatrix} {}^0\mathbf{b}_0 \times {}^0\mathbf{a}^{f-g} \\ {}^0\mathbf{a}_{f-g} \end{bmatrix}
\end{aligned}
$$

Manipulator Kinematics:

for $i = 1...n$

$$
\begin{bmatrix} {}^i\boldsymbol{\omega}_i \\ {}^i\mathbf{v}_i \end{bmatrix} = {}^i_{i-1}\mathbf{X} \begin{bmatrix} {}^{i-1}\boldsymbol{\omega}_{i-1} \\ {}^{i-1}\mathbf{v}_{i-1} \end{bmatrix} + {}^i\underline{\boldsymbol{\phi}}_i \dot{\xi}_i \tag{5.6}
$$

$$
{}^i\mathbf{a}_{f-g} = {}^i_{i-1}\mathbf{R}\ {}^{i-1}\mathbf{a}_{f-g} \tag{5.7}
$$

For a revolute joint:

$$ {}^{i}\underline{\boldsymbol{\zeta}}_i = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ {}^{i}_{i-1}\mathbf{R}\ \left[{}^{i-1}\boldsymbol{\omega}_{i-1} \times \left({}^{i-1}\boldsymbol{\omega}_{i-1} \times {}^{i-1}\mathbf{p}_{i-1\to i}\right)\right] \end{bmatrix} + \begin{bmatrix} {}^{i}\boldsymbol{\omega}_i \times \dot{\xi}_i\ \mathbf{z}_i \\ \mathbf{0}_{3\times 1} \end{bmatrix} \tag{5.8a} $$

For a prismatic joint:

$$ {}^{i}\underline{\boldsymbol{\zeta}}_i = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ {}^{i}_{i-1}\mathbf{R}\ \left[{}^{i-1}\boldsymbol{\omega}_{i-1} \times \left({}^{i-1}\boldsymbol{\omega}_{i-1} \times {}^{i-1}\mathbf{p}_{i-1\to i}\right)\right] \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times 1} \\ 2\left({}^{i}\boldsymbol{\omega}_i \times \dot{\xi}_i \mathbf{z}_i\right) \end{bmatrix} \tag{5.8b} $$

$$ {}^{i}\underline{\boldsymbol{\beta}}_i = \begin{bmatrix} {}^{i}\boldsymbol{\omega}_i \times {}^{i}\mathbf{I}_i\, {}^{i}\boldsymbol{\omega}_i \\ {}^{i}\boldsymbol{\omega}_i \times \left({}^{i}\boldsymbol{\omega}_i \times {}^{i}\mathbf{h}_i\right) \end{bmatrix} \tag{5.9} $$

$$ \begin{aligned} {}^{i}\underline{\boldsymbol{\beta}}^{H}_i \;=\; & {}^{i}\underline{\boldsymbol{\beta}}_i + {}^{i}\underline{\mathbf{I}}^{A}_i \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \left({}^{i}\mathbf{a}_f - {}^{i}\mathbf{a}_g\right) + {}^{i}\boldsymbol{\omega}_i \times {}^{i}\mathbf{v}^r_i \end{bmatrix} - \\ & \begin{bmatrix} {}^{i}\tilde{\boldsymbol{\omega}}_i & {}^{i}\tilde{\mathbf{v}}^r_i \\ \mathbf{0}_{3\times 3} & {}^{i}\tilde{\boldsymbol{\omega}}_i \end{bmatrix} {}^{i}\underline{\mathbf{I}}^{A}_i \begin{bmatrix} {}^{i}\boldsymbol{\omega}_i \\ {}^{i}\mathbf{v}^r_i \end{bmatrix} + {}^{i}\underline{\mathbf{f}}^{D}_i \left({}^{i}\boldsymbol{\omega}_i, {}^{i}\mathbf{v}^r_i\right) + m^f_i \begin{bmatrix} {}^{i}\mathbf{b}_i \times {}^{i}\mathbf{a}_{f-g} \\ {}^{i}\mathbf{a}_{f-g} \end{bmatrix} \end{aligned} \tag{5.10} $$

end for i

5.2.3 Backward Dynamics:

The second step containing Equations 5.11 through 5.16 is called backward dynamics and involves the calculation of articulated inertias ${}^{i}\underline{\mathbf{I}}^{*}_i$, and of articulated bias forces ${}^{i}\underline{\boldsymbol{\beta}}^{*}_i$, based on previously derived Equation 3.35. Applied joint torques τ_i and external spatial tip force ${}^{n+1}\underline{\mathbf{f}}_{n+1}$ are accounted for in this step and are provided to the algorithm. This step proceeds starting from link n to link 0, which is defined as the URV.

The second step also involves the thruster dynamics. The computation starts computing the velocity of the squared magnitude of incoming fluid particles on the propeller blade based on Equation 5.17. The quadrant through which the angle of attack is determined is found by checking the sign of the propeller's rotor rate. Having found the angle of attack α_{e_k} using Equations 5.18a through 5.18c, and the transformation matrix $\mathbf{R}_{\theta_k}$ using Equation 5.21, the drag and lift forces can be estimated based on Equation 5.19. The thrust contribution τ_k of each thruster within the system along with the hydrodynamic loads τ_{L_k} on each thruster rotor caused by the lift and drag forces is determined using Equations 5.22 and 5.23, respectively. Finally, the velocity of the incoming fluid particles relative to the URV's advance velocity is found using Equation 5.24. The equations used in the backward dynamics step are presented below:

Backward Dynamics:

Given: ${}^{n}\underline{\mathbf{I}}_{n}^{*} = {}^{n}\underline{\mathbf{I}}_{n}$, ${}^{n}\underline{\boldsymbol{\beta}}_{n}^{*} = {}^{n}\underline{\boldsymbol{\beta}}_{n} - {}^{n+1}\underline{\mathbf{X}}_{n}^{T}\,{}^{n+1}\underline{\mathbf{f}}_{n+1}$ and τ_i,

for $i = n...1$

$$ {}^{i}\underline{\mathbf{k}}_{i} = {}^{i}\underline{\mathbf{I}}_{i}^{*}\,{}^{i}\underline{\boldsymbol{\phi}}_{i} \tag{5.11} $$

$$ m_{i}^{*} = {}^{i}\underline{\boldsymbol{\phi}}_{i}^{T}\,{}^{i}\underline{\mathbf{I}}_{i}^{*}\,{}^{i}\underline{\boldsymbol{\phi}}_{i} \tag{5.12} $$

$$ {}^{i}\underline{\mathbf{N}}_{i} = {}^{i}\underline{\mathbf{I}}_{i}^{*} - {}^{i}\underline{\mathbf{k}}_{i}\, m_{i}^{-1}\, {}^{i}\underline{\mathbf{k}}_{i}^{T} \tag{5.13} $$

$$ \tau_{i}^{*} = \tau_{i} + {}^{i}\underline{\boldsymbol{\phi}}_{i}^{T}\,{}^{i}\underline{\boldsymbol{\beta}}_{i}^{*} \tag{5.14} $$

$$ {}^{i-1}\underline{\mathbf{I}}_{i-1}^{*} = {}^{i-1}\underline{\mathbf{I}}_{i-1} + {}^{i}\underline{\mathbf{X}}_{i-1}^{T}\,{}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\mathbf{X}}_{i-1} \tag{5.15} $$

$$ {}^{i-1}\underline{\boldsymbol{\beta}}_{i-1}^{*} = {}^{i-1}\underline{\boldsymbol{\beta}}_{i-1} + {}^{i}\underline{\mathbf{X}}_{i-1}^{T}\left({}^{i}\underline{\boldsymbol{\beta}}_{i}^{*} - {}^{i}\underline{\mathbf{N}}_{i}\,{}^{i}\underline{\boldsymbol{\zeta}}_{i} - {}^{i}\underline{\mathbf{k}}_{i}\,(m_{i}^{*})^{-1}\,\tau_{i}^{*}\right) \tag{5.16} $$

end for i

Calculation of thruster dynamic parameters:

for $k = 1...h$

$$u_{p_k} = 0.7 R_k \ \Omega_k / n_{gr_k}, \quad u_k^2 = u_{p_k}^2 + u_{a_k}^2 \tag{5.17}$$

if $\Omega_k > 0$

$$\alpha_{e_k} = (\pi/2 - p_k) - \arctan\left(\frac{u_{a_k}}{u_{p_k}}\right) \tag{5.18a}$$

if $\Omega_k < 0$

$$\alpha_{e_k} = \left(\frac{3\pi}{2} - p_k\right) - \arctan\left(\frac{u_{a_k}}{u_{p_k}}\right) \tag{5.18b}$$

if $\Omega_k = 0$

$$\alpha_{e_k} = 0 \tag{5.18c}$$

$$f_{(Lift)_k} = 0.5 \ \rho \ v_k^2 \ a \ c_{L\max} \ \sin(2\alpha_e) \tag{5.19}$$

$$f_{(Drag)_k} = 0.5 \ \rho \ v_k^2 \ a \ c_{D\max} (1 - \cos(2\alpha_e)) \tag{5.20}$$

$$\theta_k = \frac{\pi}{2} - p_k - \alpha_{e_k} \quad \mathbf{R}_{\theta_k} = \begin{bmatrix} \cos(\theta_k) & -\sin(\theta_k) \\ \sin(\theta_k) & \cos(\theta_k) \end{bmatrix} \tag{5.21}$$

$$\begin{bmatrix} \tau_k \\ f_{p_k} \end{bmatrix} = \mathbf{R}_{\theta_k} \begin{bmatrix} f_{(Lift)_k} \\ f_{(Drag)_k} \end{bmatrix} \tag{5.22}$$

$$\tau_{L_k} = 0.7 \ r_k \ f_{p_k} \tag{5.23}$$

$$\bar{u}_{a_k} = (u_{a_k} - u_0) \tag{5.24}$$

end for k

5.2.4 Forward Accelerations:

All the associated accelerations are computed in the last step called forward accelerations. The angular acceleration of the thruster motor's rotor and the acceleration of the incoming fluid particles are computed using Equations 5.25 and 5.26. The collective effect of each thrust ${}^0\underline{\mathbf{f}}_0$ at the centre of mass of the URV is found based on Equation 5.27. The total thrust ${}^0\underline{\mathbf{f}}_0$ is treated as an external force action on the URV and is substituted into Equation 5.28 along with the articulated-body inertia ${}^0\underline{\mathbf{I}}_0^*$ and the bias force ${}^0\underline{\boldsymbol{\beta}}_0^*$ of the whole URVM system. This step gives the biased acceleration of the URV ${}^0\underline{\acute{\mathbf{a}}}_0$. In simulation, the absolute acceleration of the URV is needed. This can be found by adding the gravitational acceleration ${}^0\mathbf{a}^g$ to the biased acceleration as given in Equation 5.29. The absolute acceleration of the URV in terms of the inertial frame ${}^E\underline{\mathbf{a}}_0$ is computed using Equation 5.30. Finally, the joint accelerations $\ddot{\xi}_i$ are calculated starting from link 1 to link n using Equations 5.31 through 5.33.

In order to find state vector of the URVM at a later time, Equations 5.30 and 5.32 are integrated forward in time. The equations used in the backward dynamics step are presented below:

Forward Accelerations:

Thruster Accelerations:

for $k = 1...h$

$$\dot{\Omega}_k = -k_{1_k}\ \Omega_k + k_{2_k}\ u_{a_k} - k_{h_k}\ \tau_{L_k} \tag{5.25}$$

$$\dot{u}_{a_k} = -k_{4_k}\ k_{3_k}^{-1}\bar{u}_{a_k}\ |\bar{u}_{a_k}| + k_{3_k}^{-1}\ \tau_k \tag{5.26}$$

end for k

$${}^0\underline{\mathbf{f}}_0 = \Lambda \begin{bmatrix} \tau_1 & \tau_2 & \dots & \tau_h \end{bmatrix}^T \tag{5.27}$$

URV Accelerations:

$$ {}^{0}\underline{\acute{\mathbf{a}}}_{0} = \left({}^{0}\underline{\mathbf{I}}^{*}_{0}\right)^{-1}\left({}^{0}\underline{\mathbf{f}}_{0} + {}^{0}\underline{\boldsymbol{\beta}}^{*}_{0}\right) \tag{5.28} $$

$$ {}^{0}\underline{\mathbf{a}}_{0} = {}^{0}\underline{\acute{\mathbf{a}}}_{0} + \begin{bmatrix} \mathbf{0}_{3\times 1} \\ {}^{0}\mathbf{a}_{g} \end{bmatrix} \tag{5.29} $$

In terms of the inertial reference frame

$$ {}^{E}\underline{\mathbf{a}}_{0} = \underline{\mathbf{T}}\ {}^{0}\underline{\mathbf{a}}_{0} + \begin{bmatrix} \dot{\mathbf{T}}_{2} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \end{bmatrix} {}^{0}\underline{\mathbf{v}}_{0} \tag{5.30} $$

Manipulator Accelerations:

for $i = 1...n$

$$ {}^{i}\underline{\bar{\mathbf{a}}}_{i} = {}^{i}\underline{\mathbf{X}}_{i-1}\ {}^{i-1}\underline{\acute{\mathbf{a}}}_{i-1} + {}^{i}\underline{\boldsymbol{\zeta}}_{i} \tag{5.31} $$

$$ \ddot{\xi}_{i} = \left(m^{*}_{i}\right)^{-1}\ \tau^{*}_{i} - \left[{}^{i}\underline{\mathbf{k}}_{i}\ \left(m^{*}_{i}\right)^{-1}\right]^{T}\ {}^{i}\underline{\bar{\mathbf{a}}}_{i} \tag{5.32} $$

$$ {}^{i}\underline{\acute{\mathbf{a}}}_{i} = {}^{i}\underline{\bar{\mathbf{a}}}_{i} + {}^{i}\underline{\boldsymbol{\phi}}_{i}\ \ddot{\xi}_{i} \tag{5.33} $$

end for i

5.3 Simulation Studies for the Demonstration of the Dynamic Coupling

As mentioned in Chapter 1, when the slave arm attempts to follow the desired joint motions set by the master-arm, reaction forces and moments develop at the junction

point between the URV and manipulator. The reaction loads act as disturbances to the URV motion and in turn adversely affect the placement of the end-effector, as the end-effector absolute motion is dependent on that of the URV.

In order to demonstrate the extent of the disturbance caused by the dynamic coupling on the URV position and orientation and in turn on the end-effector position and orientation, a small URV maneuver is simulated. Two cases are considered. In the first, the URV is held fixed and the manipulator joint motors are activated, is considered. By holding the URV fixed, the URVM motions are eliminated, and the URVM system reduces to a land-based manipulator that exhibits perfect trajectory control -the ideal case. The ideal case gives a benchmark for comparison. In the second case, the URV is released. Comparison of the two cases demonstrates the level of the dynamic coupling inherent in URVM systems.

The simulation code that numerically integrates the dynamics of a URVM system forward in time is created based on the complete articulated-body algorithm presented in Section 5.2. MatLab®'s numerical integration command *ode45,* Runge-Kutta variable step integration method, is used to numerically integrate the acceleration equations forward in time. This simulation also provides a basis for the validation of the control law to be implemented in the next chapter. In the control section, the control law will attempt to reduce the dynamic coupling on the URV by automatically commanding the URV's thrusters.

5.3.1 System Description and Frame Attachment

The URVM system considered in this work is shown in Figure 5.1, and all the associated parameters are given in Appendix E. A 3 degree of freedom $R \perp R \setminus\setminus R$ serial

manipulator is chosen for the manipulator part of the URVM system. The Z axis of

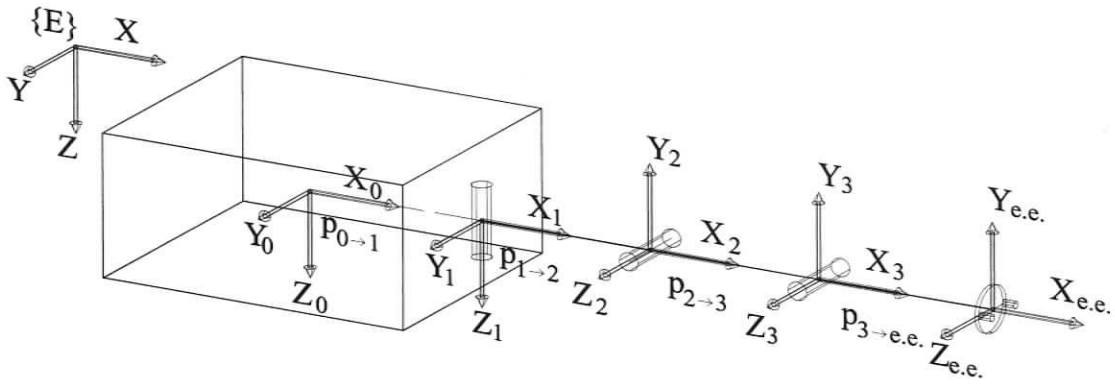


Figure 5.1: Frame Attachments for the URVM system

the earth-fixed inertial frame {E} is assumed in the gravity direction as is consistent with traditional marine mechanics. The origin of the earth-fixed frame is coincident with the initial position of the URV's centre of mass. The URV is modeled as another manipulator link in the serial chain and numbered as 0. The body-fixed frame {0} is attached to the centre of mass of the URV as shown in Figure 5.1. With regards to the manipulator part of the URVM, the body-fixed frames are attached to the initial base joint of each successive link using the modified Denavit-Hartenberg convention [48] and as shown in Figure 5.1 and Table 5.1.

5.3.2 Fixed-Base Case (The Ideal Scenario)

A simulation study for a fixed-base underwater manipulator is performed in this section. As discussed earlier, the fixed-base underwater manipulator simulation represents the ideal case and therefore generates data to be compared to that of the

Table 5.1: Denavit-Hartenberg Parameters for the URVM system

$i-1$	α_{i-1}	a_{i-1}	d_i	θ_i	i
0	0	$p_{0\to1}$	0	q_1	1
1	$-\pi/2$	$p_{1\to2}$	0	q_2	2
2	0	$p_{2\to3}$	0	q_3	3
3	0	$p_{3\to e.e.}$	0	0	*e.e.*

uncontrolled URVM system.

Some modifications to the complete articulated-body algorithm must be made to be able to implement it to the fixed-base underwater manipulators. Given that the URV is stationary, ${}^0\boldsymbol{\omega}_0$ and ${}^0\mathbf{v}_0$ are taken as zero. The forward kinematics proceeds from Equations 5.6 through 5.10. Since the URV is eliminated from the serial-chain, the thruster dynamics are omitted from the backward dynamics equations. Therefore, only Equations 5.11 through 5.16 are employed for the backward dynamics of the fixed-based underwater manipulator. Likewise, the equations associated with the thrusters and the URV are omitted from the forward accelerations step. Finally, Equations 5.31 through 5.33 are used to obtain the joint accelerations with ${}^0\underline{\acute{\mathbf{a}}}_0 = \begin{bmatrix} 0 & -{}^0\mathbf{a}_g^T \end{bmatrix}^T$, which is the biased gravitational acceleration.

In the simulation, the initial state vectors $\mathbf{q} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ and $\dot{\mathbf{q}} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ are taken for the manipulator. Randomly selected torque vectors of $\boldsymbol{\tau} = \begin{bmatrix} 2.5 & 0 & 0 \end{bmatrix}^T$ Nm and $\boldsymbol{\tau} = \begin{bmatrix} 0.01 & 0 & 0 \end{bmatrix}^T$ Nm are applied to the manipulator joints for $t \leq 2$ sec. and for $2 < t \leq 10$ sec., respectively. In the ideal case, these joints torques produce an end-effector tracking path that is considered to be the pilot's true intent. The linear velocity and acceleration of the surrounding fluid are assumed to be zero. The

system is also assumed to be neutrally buoyant.

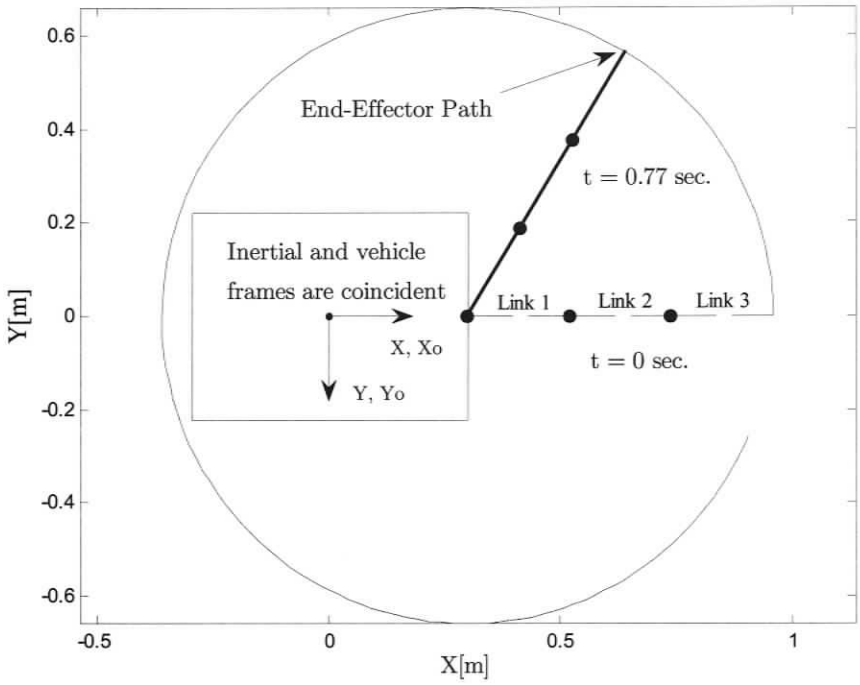


Figure 5.2: End-Effector Path

Figure 5.2 demonstrates the resulting end effector tracking path on the X-Y plane. Figure 5.3 illustrates the time history of the manipulator's state variables of the fixed-base underwater manipulator.

When the first joint is actuated, the whole manipulator arm moves in accordance with the first joint. As a result, the end-effector draws a circle whose centre is coincident with the first joint. Joint displacements for joint 2 and 3 are not observed due to the layout of the manipulator and the hydrodynamic loading involved. The hydrodynamic loading does not produce spatial forces that are capable of moving the second and third joint about their axes.

When the whole arm moves, associated added mass, drag and buoyancy forces

develop over each link of the manipulator. Since the manipulator is assumed to be neutrally buoyant, the buoyancy effect is compensated by the gravity force. For the three joints, the added mass force develops only within the X-Y plane of the links, as the added mass matrix of each joint is composed of only diagonal elements. All the joints within the manipulator experience hydrodynamic load due to the added mass, as they all move. However, only the first joint moves around its joint axis, as only the added mass forces acting on the first joint are capable of producing moments around the joint axis. This result would have been different if the off-diagonal elements of the added mass matrices were different than zero.

Likewise, all the joints experience drag forces and moments but only the motion of the first joint around its joint axis is affected by the drag forces.

5.3.3 Mobile-Base Case (The Worst Scenario)

In the mobile-base case simulation, the URV's initial orientation and position state vector and the URV's spatial velocity state vector are taken as $\underline{\mathbf{x}}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$ rad, m and $\underline{\mathbf{v}}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$, respectively. The initial state vectors $\mathbf{q} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ and $\dot{\mathbf{q}} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ are taken for the manipulator as in the fixed-base case. The same torque values are applied to the system for the same time intervals. However, it is assumed that there is no facility nearby the subsea worksite on which the URVM system can be docked; therefore the URV is no longer fixed and floats in the water in contrast to the fixed-based case. This situation corresponds to the worst scenario, as even small disturbances acting on the URV are capable of disturbing the URV position and orientation and thus, those of the end-effector.

In this simulation case, the effect of dynamic coupling on the URV and manip-

ulator state variables and the end effector trajectory path is demonstrated. The articulated-body algorithm, excluding the thruster dynamic equations, is used in the simulation.

As explained before, during task execution, reaction forces and moments occur at the junction point of the URV and the manipulator. These reaction forces and moments are exerted on the URV via the dynamic coupling of the manipulator and URV as illustrated in Figures 5.4 and 5.5. As a result, the URV motion is adversely affected by the reaction forces as illustrated in Figure 5.6.

Note that since the dynamic coupling causes disturbances only in the URV's horizontal plane, only the state variables associated with the URV's horizontal motion are shown in Figure 5.6.

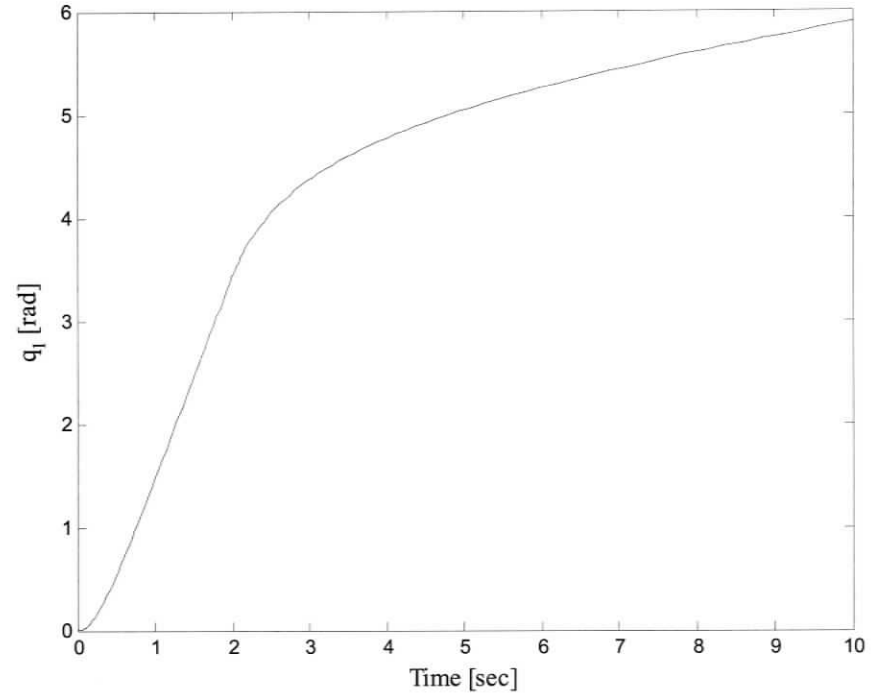
When the URV attitude is disturbed by the dynamic coupling, the dynamical load coupled with the hydrodynamic load is transmitted from the URV to the manipulator. As such, the manipulator joint motions diverge from what the pilot intended to set, which is determined by the applied joint torques. As a result, the pilot can not operate the manipulator properly. The effect of dynamic coupling on the state variables of joint 1 is illustrated in Figure 5.7. Note that since the URV moves only in its horizontal plane, it does not cause any motion around the joint axes 2 and 3, and therefore joint displacements for joints 2 and 3 are not demonstrated in Figure 5.7.

The resulting end-effector path is given in Figure 5.8. In Figure 5.8, the trajectory path of the ideal scenario is considered to be the pilot's true intent. The discrepancy between the path of the ideal scenario and the worst scenario results from the dynamic coupling effect.

Consequently, it can be concluded that the dynamic coupling effect on the URV is caused by the reaction forces at the junction point between the URV and the

manipulator. These forces act on the URV and in turn cause changes in the attitude of the URV. These effects leads to the reduction of task effectiveness, and it must be compensated to obtain better system performance in underwater robotic applications. In the next chapter, the compensation of the effect of the reaction forces on the URV will be attempted by automatically commanding the URV's thrusters.

a)



b)

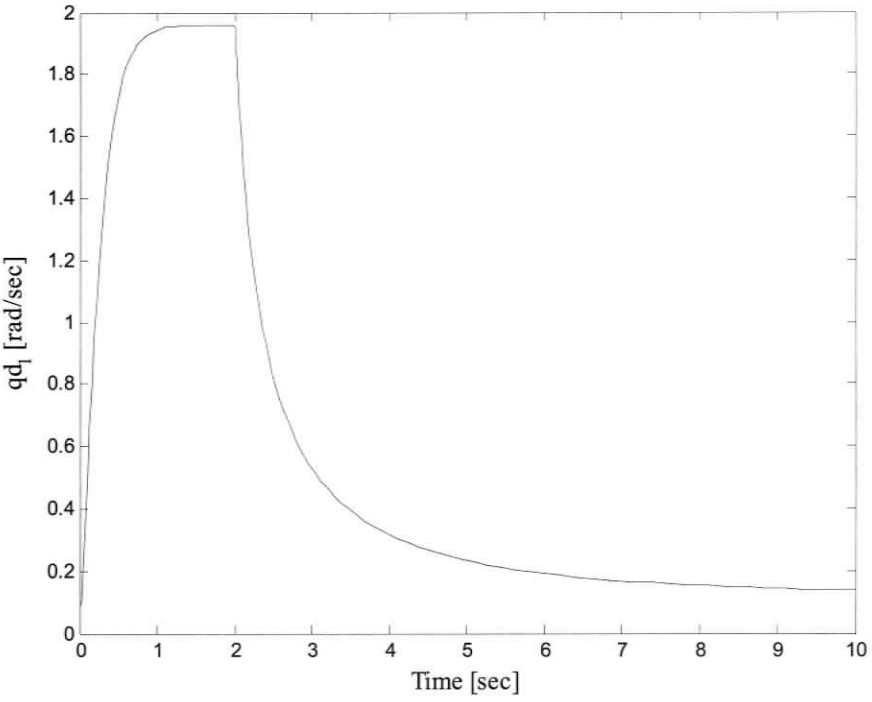
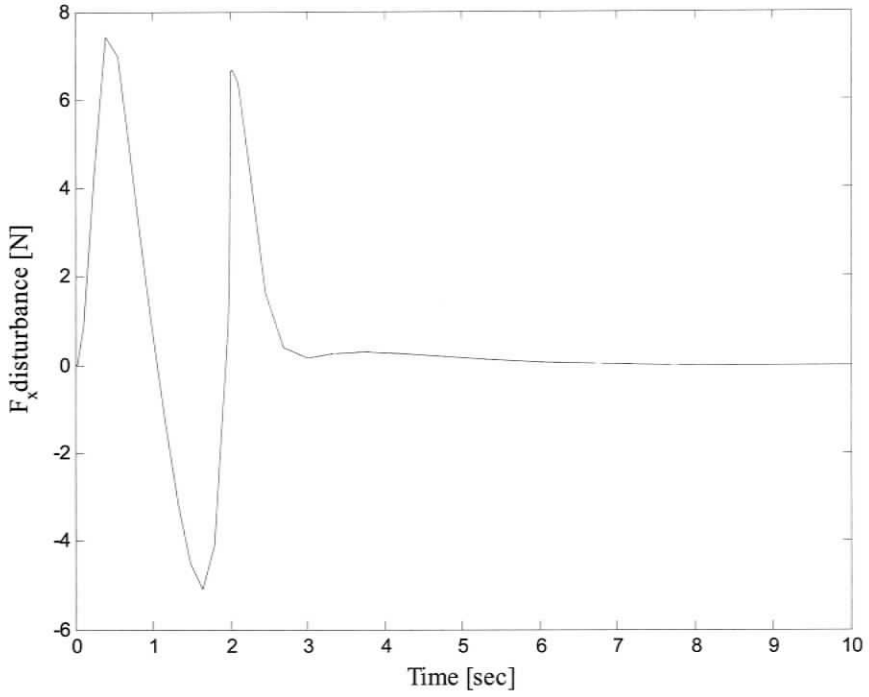


Figure 5.3: a) Time History of Joint Position for Joint 1 b) Time History of Joint Velocity for Joint 1

a)



b)

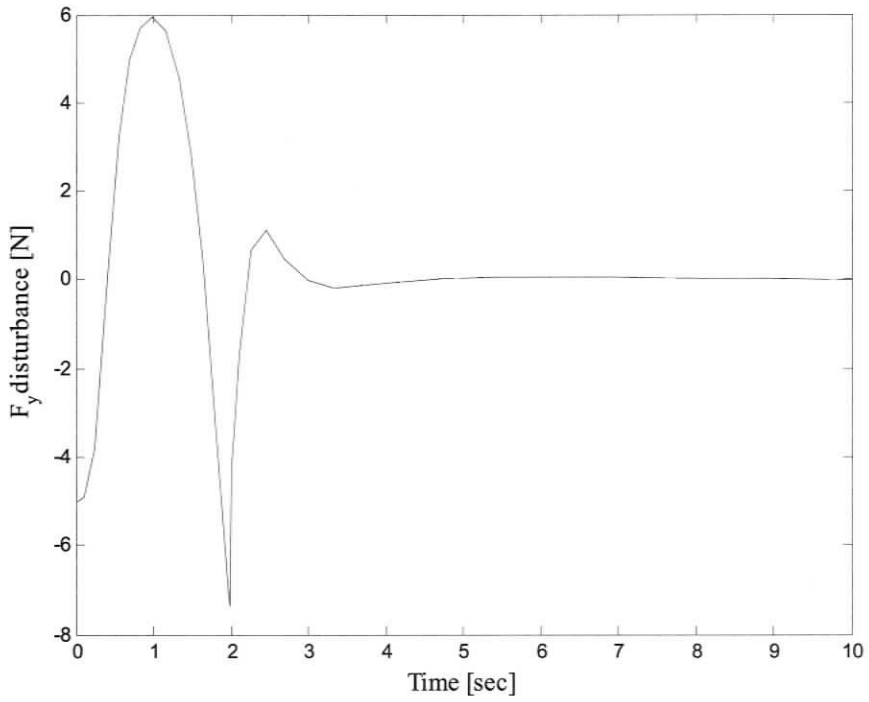


Figure 5.4: a) Force Acting Along Xo Direction Caused by Dynamic Coupling b) Force Acting Along Yo Direction Caused by Dynamic Coupling

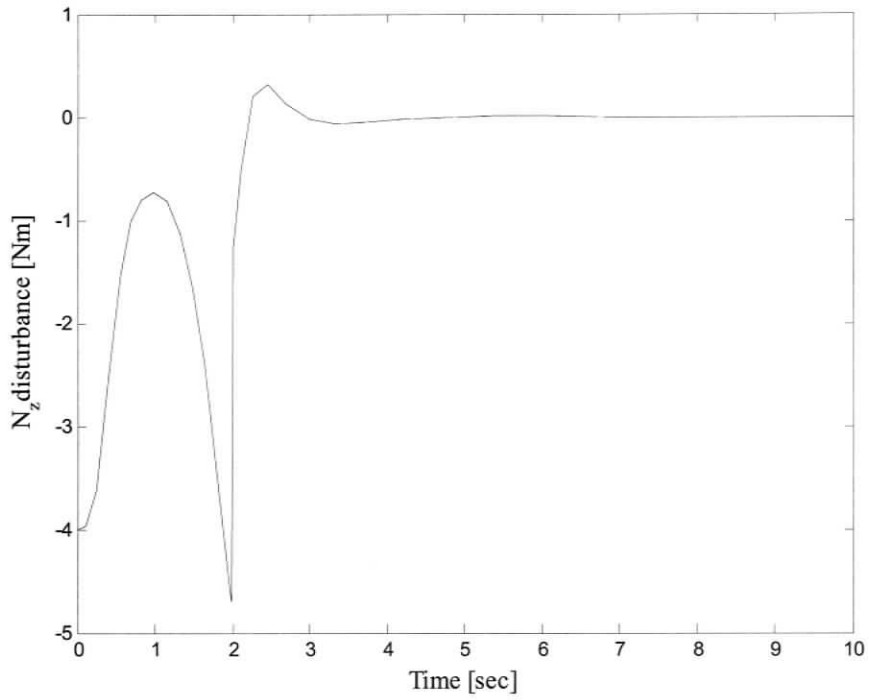


Figure 5.5: Moment Acting Around Zo Direction at the Center of Mass of the URV Caused by Dynamic Coupling

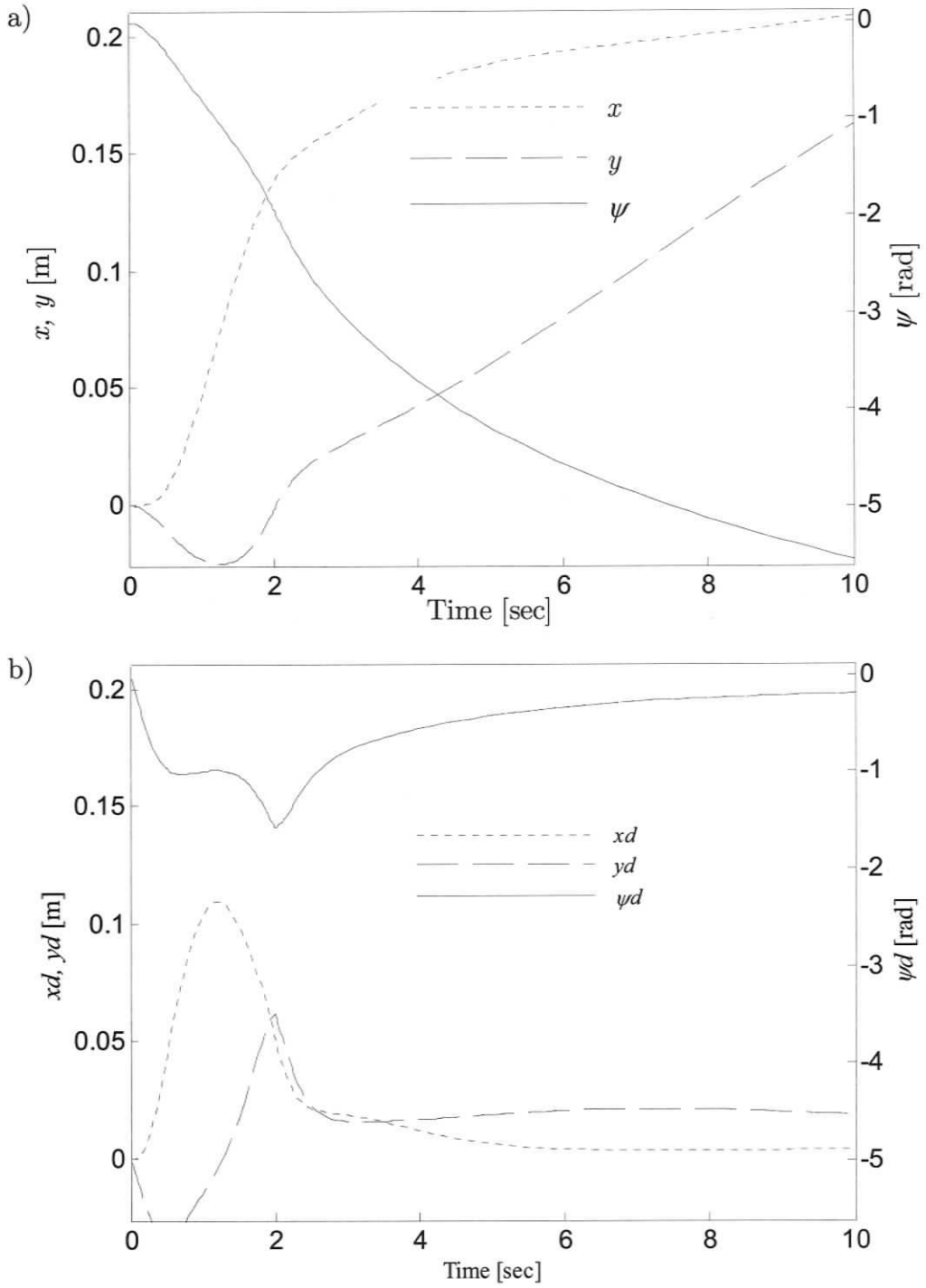


Figure 5.6: a)Vehicle Position and Orientation When Released b) Vehicle Translational and Rotational Velocities When Released

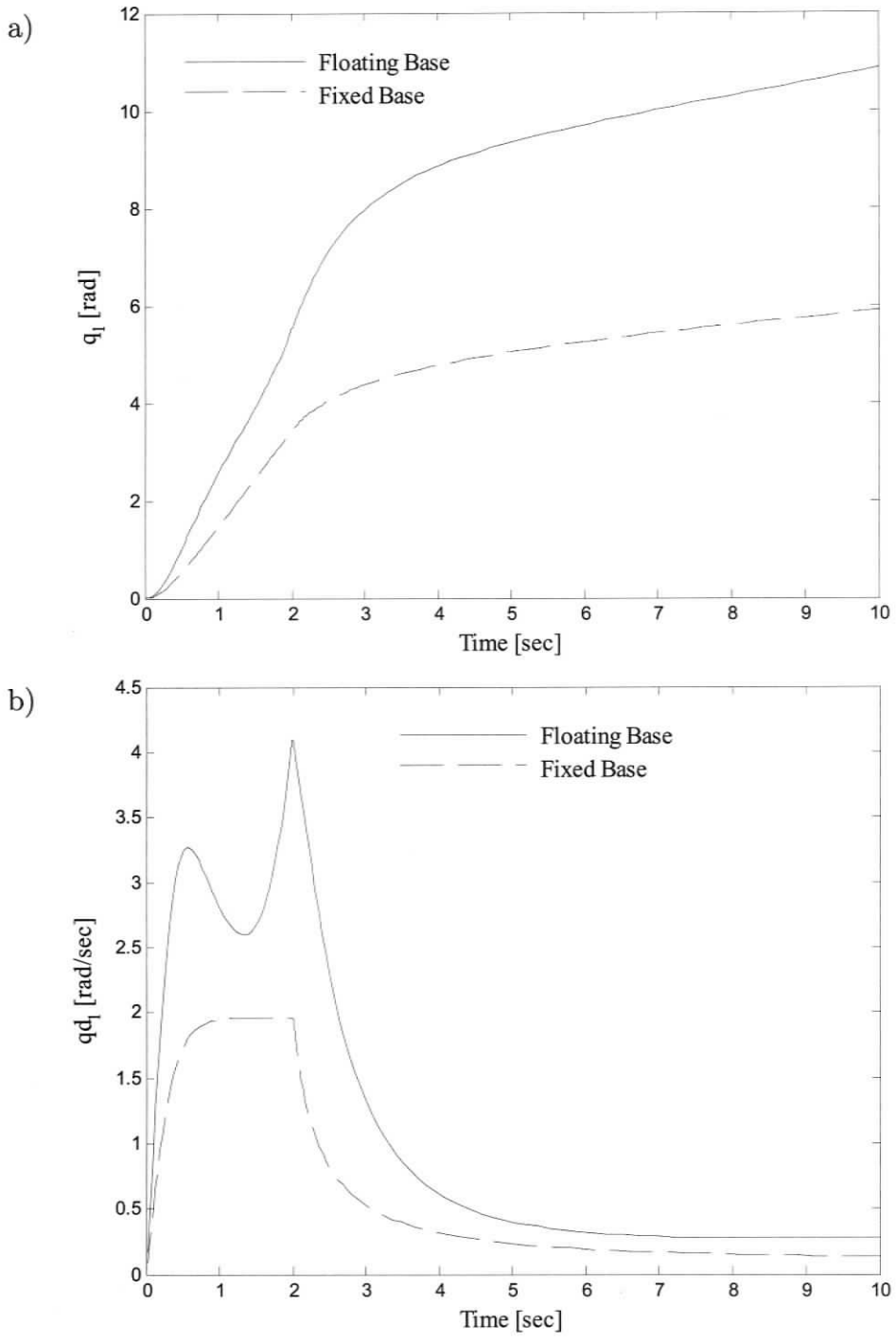


Figure 5.7: a) Time History of Joint 1 Position When the URV Released b) Time History of Joint 1 Velocity When the URV Released

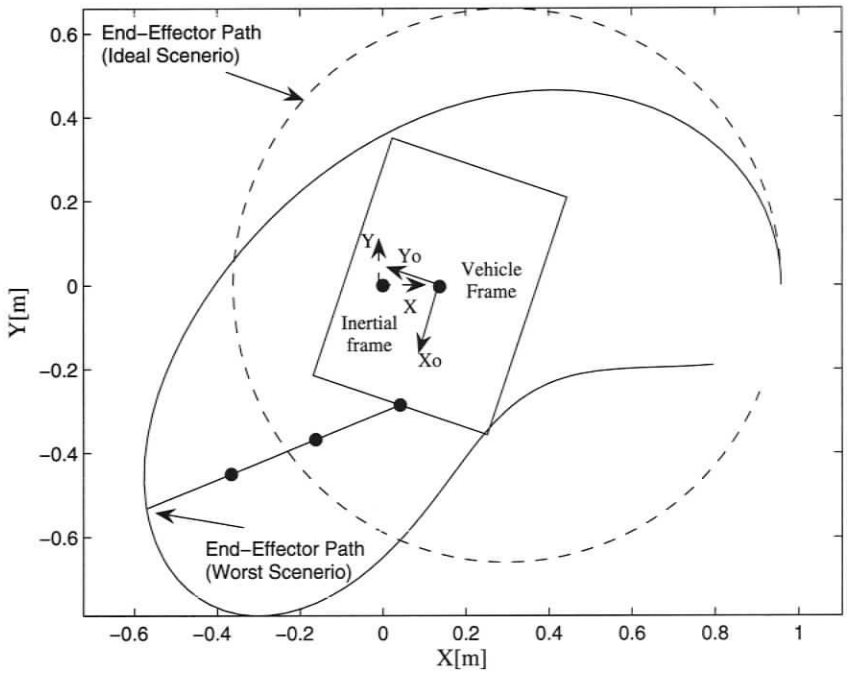


Figure 5.8: End-Effector Path with the URV Floating

Chapter 6

Control of the URVM System

6.1 Overview

In Chapter 5, it has been concluded that the dynamic coupling effect reduces the effectiveness of the human pilot, and therefore must be eliminated to increase the efficiency of any URVM deployment. In order to eliminate the dynamic coupling effect, the URV motion caused by the manipulator joint motions must be predicted on-line and compensated with the URVM thrusters. This problem is equivalent to an automatic control of the URV in the presence of disturbances created by the manipulator joint motions.

In order to eliminate the dynamic coupling, a model-based non-linear control technique, namely sliding-mode control technique [59], will be implemented to the URV under the dynamic and hydrodynamic load due to the manipulator motion. The articulated-body dynamic equations will be incorporated into this controller. To the author's best knowledge, the incorporation of the articulated-body equations into the sliding-mode controller has not been demonstrated in literature before. It is expected

that better end-effector trajectory following control performance will be obtained by using the articulated-body equations in the controller since it affords the dynamic effect of the manipulator on the URV and thus gives a better approximation of the URVM system. The thrusters are to be used to mitigate the disturbances. However, generating the desired thrust commanded by the sliding-mode URVM controller requires active control of the URV's thrusters. In order to control the individual URVM thruster, the model-based velocity control technique [18] will be used.

Simulations will be performed to show the effectiveness of the proposed control scheme in the reduction of the dynamic coupling effect on the URV motion.

6.2 Sliding Mode Control Based on Articulated-Body Formulation

6.2.1 Linear Control Technique Disadvantages

The linear control techniques are generally used for the trajectory control problem of URVs. In the linear control techniques, the dynamics of underwater systems are linearized about specific operating points and a series of linear controllers is constructed based on the linearized models for each specific operating point. However, this method involves several control designs, *i.e.,* one controller for each operation point. Furthermore, the underwater systems may progress away from the operating point due to their non-linear nature, and as a result, the linear controllers fail. In addition, since the dynamics model is discarded in the course of linearizing, detailed dynamics models can not be utilized. Using a non-linear control technique can eliminate the aforementioned problems about the use of linear control techniques.

6.2.2 Advantages of Incorporating the Articulated-Body Equations in the Controller

The URVM dynamics are dominated by hydrodynamic terms and it is difficult to accurately measure or estimate the hydrodynamic coefficients. This calls for a robust controller: one that is insensitive to inaccuracies in the dynamic model of the URVM. The sliding mode strategy has been successfully applied to the robust control of URVs [31], [60], due to its fundamental property of insensitivity to parameter variations in sliding mode. The same control approach has also been implemented to URVM systems by considering the dynamic coupling between the two systems as disturbances in the URV control loop [44].

In [44], the URV and manipulator are modelled separately within the controller. The disturbances caused by the manipulator motion have been incorporated into the control law by introducing a feedforward compensation term that is obtained using the Newton-Euler recursive algorithm for manipulator dynamics. It has been shown that the dynamical loads transmitted from the URV to the manipulator can be incorporated in the recursive Newton-Euler equations by using the URV's velocities and accelerations as the base motions of the manipulator [44]. However, this requires knowing the acceleration state knowledge of the URV, which is difficult to measure accurately during the URV operation. It is proposed that the articulated-body algorithm be used in the control law equations since it affords direct calculation of the URV accelerations based on a known URVM state vector $\mathbf{u}_s$. This eliminates the need to estimate the URV accelerations using inertial measurement sensors. Furthermore, since the dynamic coupling effects are embedded in the URV's articulated-body equations, it is expected to have a better approximation of the URV dynamics at the

expense of a relatively small increase in the computational complexity of the controller.

6.2.3 Preliminaries

Consider a single-input single-output system whose equations of motion are given by:

$$x^{(n)} = f(\mathbf{x}, t) + b(\mathbf{x}, t)\, u_c \tag{6.1}$$

where x represents the output of interest, and $\mathbf{x}$ represents the state vector defined as $\mathbf{x} = \begin{bmatrix} x & \dot{x} & \ldots & x^{(n-1)} \end{bmatrix}^T$, and u_c is the control input. The superscripts n is the order of differentiation. In Equation 6.1, the function $f(\mathbf{x}, t)$ represents the dynamics of the system, which is possibly non-linear and time-varying. The function $f(\mathbf{x}, t)$ is not exactly known but is estimated as $\hat{f}(\mathbf{x}, t)$. The estimation error on $f(\mathbf{x}, t)$ is upper bounded by a known function $F(\mathbf{x}, t)$;

$$\left| f(\mathbf{x}, t) - \hat{f}(\mathbf{x}, t) \right| \leq F(\mathbf{x}, t) \tag{6.2}$$

Likewise, the control gain term $b(\mathbf{x}, t)$ is not exactly known but is estimated as $\hat{b}(\mathbf{x}, t)$. Moreover, $b(\mathbf{x}, t)$ is known to be a constant sign and is bounded by a function of $\hat{b}(\mathbf{x}, t)$ according to:

$$1/\beta \leq \hat{b}(\mathbf{x}, t)/b(\mathbf{x}, t) \leq \beta \tag{6.3}$$

The parameter β can be called the gain margin of the control system, as it has the same effect on the non-linear system as in a linear control system.

For a trajectory controller, the control problem is to design a control law u_c such that the state $\mathbf{x}$ follows a desired state $\mathbf{x}_d = \begin{bmatrix} x_d & \dot{x}_d & \ldots & x_d^{(n-1)} \end{bmatrix}^T$ with a specified dynamic characteristic under the presence of uncertainties on $f(\mathbf{x}, t)$ and $b(\mathbf{x}, t)$.

The tracking error vector is given by:

$$\tilde{\mathbf{x}} = \begin{bmatrix} \breve{x} & \dot{\breve{x}} & \dots & \breve{x}^{(n-1)} \end{bmatrix}^T \tag{6.4}$$

where $\breve{x}$ is defined as:

$$\breve{x} = x - x_d \tag{6.5}$$

A time-varying surface is defined in the state space $\mathbf{R}^n$ by the scalar equation $s(\mathbf{x}, t) = 0$ where:

$$s(\mathbf{x}, t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} \breve{x} \qquad \lambda > 0 \tag{6.6}$$

and λ is a strictly positive constant defining the bandwidth of the error dynamics and will be explained later. Equation 6.6 defines desired error dynamics of order $n-1$. Given that Equation 6.6 sets the desired dynamics of the system, verifying Equation 6.6 means that the system dynamics behaves consistently with the desired dynamics. Therefore it can be concluded that by constructing the controller from Equation 6.6, the control design can be simplified since a system of order $n-1$ is dealt with instead of one with order n. Moreover, the problem of tracking the n-dimensional vector $\mathbf{x}_d$ is reduced to a first-order stabilization problem in s.

There are some conclusions that can be drawn based on the definition of the error dynamics. For instance, bounds on the error metric s can be associated with the bounds on the tracking performance, assuming $\tilde{\mathbf{x}} = \mathbf{0}$ at $t = 0$, as follows:

$$t \geq 0, \quad |s(t)| \leq \Phi \quad \Longrightarrow t \geq 0, \quad |\breve{x}(t)| \leq \frac{\Phi}{\lambda^{n-1}} \tag{6.7}$$

where Φ will be called "boundary layer" and will be explained later. The proof of Equation 6.7 is given in [59]. This result stems from the fact that the tracking error

$\check{x}$ results from sequentially applying first-order low-pass filters in view of Equation 6.6 as illustrated in Figure 6.1. Note that in Figure 6.1, $p = \frac{d}{dt}$ is the Laplace operator. The variable s can be considered to be a scalar quantity that indicates the magnitude of the discrepancy between the desired state and the current state. For instance, in the case of $s = 0$, the system state variables are on the sliding surface, and thus the system acts in accordance with the desired dynamics set by Equation 6.6. Note that once the error dynamics is specified in terms of error values, it can also be defined in terms of state variables. This implies that for each desired state value, there is a corresponding sliding surface that moves with the desired state in the state space. The error metric s being equal to zero implies that $\check{\mathbf{x}}$ converges to zero with a time constant $(n-1)/\lambda$ if the system initial state variables originally start on the sliding surface as illustrated in Figure 6.2 for a second-order system. Note that the time

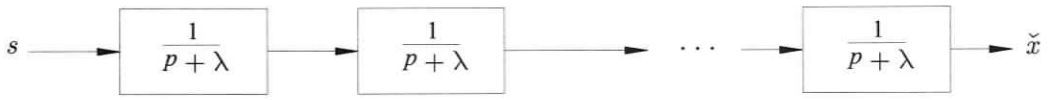


Figure 6.1: $n-1$ Filters Between s and $\check{x}$

constant $(n-1)/\lambda$ is due to the existence of $n-1$ filters with a time constant $1/\lambda$ between $\check{x}$ and s. However, if $s < 0$ or $s > 0$, then the state variables are not on the sliding surface meaning the desired dynamics is not accomplished. Consequently, the problem of tracking $\mathbf{x}_d$ becomes equivalent to that of keeping s at zero for all $t > 0$.

The error metric s can be kept at zero by constructing a control law u of Equation

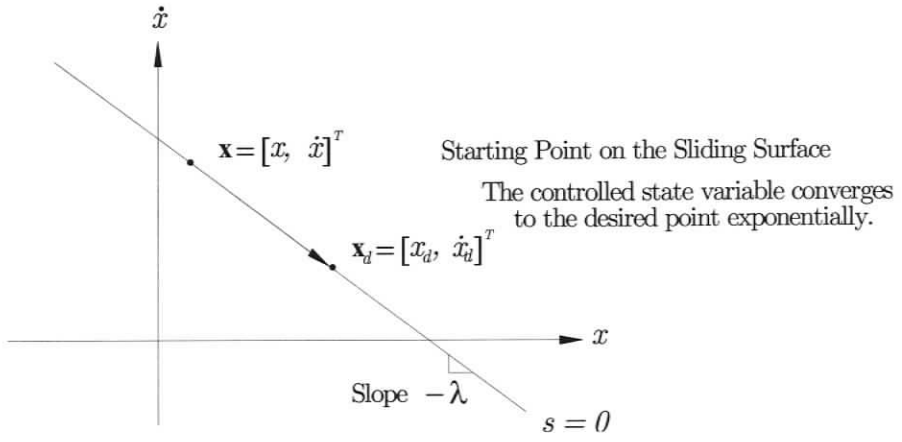


Figure 6.2: The Sliding Surface for $n = 2$ Represents a Line. The State Variable Hits the Desired Point with a Time-Constant.

6.1 based on the sliding condition:

$$\frac{1}{2}\frac{d}{dt}s^2 \leq -\eta\,|s| \tag{6.8}$$

where η is a positive constant. In Equation 6.8, s^2 can be interpreted as the squared norm of the distance to the sliding surface. Given that the right hand side of Equation 6.8 is always negative, as long as Equation 6.8 is satisfied, s^2 decreases along the system state trajectories. This constrains the system state trajectories to direct towards the sliding surface and guarantees the convergence of s to zero in a finite time smaller than $|s\,(t=0)|\,/\eta$ [59]. In cases where the system state variables don't originally start on the sliding surface, $\mathbf{x}(t=0) \neq \mathbf{x}_d(t=0)$ as illustrated in Figure 6.3, the convergence of s to zero also means the exponential convergence of $\tilde{\mathbf{x}}$ to zero which translates to an asymptomatic progression towards the sliding surface. The sliding condition also ensures that once the system state trajectory is on the sliding surface ($s = 0$), it stays on the sliding surface in spite of parameter changes and disturbances.

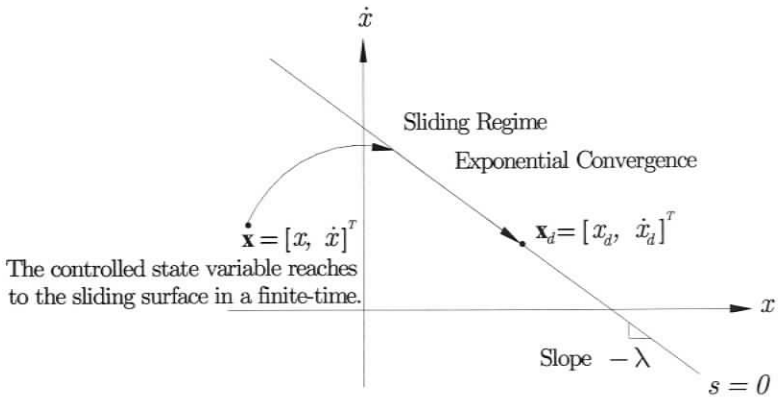


Figure 6.3: If $\mathbf{x}(t = 0) \neq \mathbf{x}_d(t = 0)$, the Controlled Variable State Reaches to the Sliding Surface in a Finite Time and Then Hits the Desired Point Exponentially.

To sum up, the controller design is based on the sliding-mode methodology and involves two main steps. The first step is to define the desired dynamics in the form of a sliding surface s, Equation 6.6, and the second step is to find a control law u such that the sliding condition, Equation 6.8, is satisfied. It is this second step that is now presented.

6.2.4 Sliding-Mode Control Design

In this section, the sliding-mode controller of [59] will be applied to the control of the URV in the presence of manipulator disturbances. The control law design involves breaking the control system up into a set of low-order controllers so that each axis of the URV has a separate controller. This was done in [31]. Each controller includes cross-coupling terms due to the multiaxis motion of the URV and the manipulator arm. These coupling effects are treated as disturbances within each controller.

The dynamic model of the URV within the controller is derived from Equation 3.44. The sliding-mode control strategy seeks to control motion of the URV as seen by an observer in the URV's body-fixed frame. Since Equation 3.44 gives the absolute URV accelerations, it is necessary to remove the component of this result that is due to the rotating body fixed frame. As such, the controller model is given by:

$$\underline{\mathbf{a}} = \left({}^{0}\underline{\mathbf{I}}_{0}^{*}\right)^{-1}\left({}^{0}\underline{\mathbf{f}}_{0} + {}^{0}\underline{\boldsymbol{\beta}}_{0}^{*}\right) + \begin{bmatrix} \mathbf{0}_{1\times 3} & ({}^{0}\mathbf{a}^{g})^{T} \end{bmatrix}^{T} - \begin{bmatrix} \mathbf{0}_{1\times 3} & (\boldsymbol{\omega}_{0}\times\mathbf{v}_{0})^{T} \end{bmatrix}^{T} \tag{6.9}$$

In Equation 6.9, the dynamic effect of the manipulator on the dynamics of the URV is embedded in the articulated inertia and the bias force of link 0 as is presented in Equation 3.44. To illustrate the control law design, only the URV sway motion is described in detail. The model of the dynamics of the sway axis in the sliding-mode controller can be given as follow:

$$\dot{\hat{v}} = \hat{f}\left(\underline{\mathbf{x}}_{0},\ \underline{\mathbf{v}}_{0},\ \mathbf{q},\ \dot{\mathbf{q}}\right) \tag{6.10}$$

where $\dot{\hat{v}}$ is the estimation of the time derivative of the URV's swaying velocity when $\underline{\mathbf{f}}_{0}$ is equal to zero, the term $\hat{f}\left(\underline{\mathbf{x}}_{0},\ \underline{\mathbf{v}}_{0},\ \mathbf{q},\ \dot{\mathbf{q}}\right)$, is the estimation of this same quantity based on Equation 6.9. The closed-loop dynamics is given by:

$$\dot{\hat{v}} = \hat{f}\left(\underline{\mathbf{x}}_{0},\ \underline{\mathbf{v}}_{0},\ \mathbf{q},\ \dot{\mathbf{q}}\right) + \hat{b}\ u_{c} \tag{6.11}$$

where u_c is the control input, which is equivalent to $\underline{\mathbf{f}}_{0}$ since the system is controlled by the thrusters. The first step in the design of the controller is the definition of the sliding surface s in the state space based on Equation 6.6. As explained before, this surface is used to characterize the desired dynamics. The URV's sway dynamics is represented by a second-order nonlinear differential equation, and therefore the

corresponding sliding surface is a line in the state space as shown in Figure 6.4 The sliding line s for the sway motion of the URV is given for $s = 0$ and $n = 2$ as follows:

$$s = \left(\frac{d}{dt} + \lambda\right)^{2-1} \check{y} = \check{v} + \lambda \check{y}_0, \quad \check{y}_0 = y_0 - y_{0,d} \tag{6.12}$$

where subscript d represents the desired value of the state variable. The term y_0 is the Y_0 coordinate of the URVs absolute position in terms of the URV frame. Given that the sliding surface is a function of the desired state values, the sliding surface moves on the state space with the desired state values.

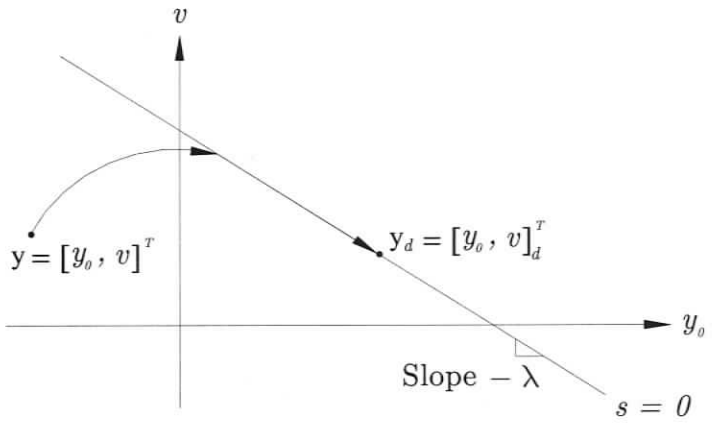


Figure 6.4: URV's Controlled State Variable on the State Space

As explained before, the construction of a sliding-mode controller requires knowing the bounds on f and b. The dynamics of the parameter of interest and the control gain are bounded according to:

$$\left| f - \hat{f} \right| \leq F \tag{6.13}$$

$$0 < b_{\min} < b < b_{\max} \tag{6.14}$$

respectively. The estimate $\hat{b}$ of control gain b is defined as the geometric mean of the lower bound $b_{\min}$ and the upper bound $b_{\max}$ of b:

$$\hat{b} = (b_{\min} b_{\max})^{1/2} \tag{6.15}$$

Equation 6.14 can be written as follows:

$$\beta^{-1} \leq \frac{\hat{b}}{b} \leq \beta \tag{6.16}$$

where

$$\beta = (b_{\max}/b_{\min})^{1/2} \tag{6.17}$$

Once the s is zero, ideally it must be kept at zero. This implies that the derivative of $\dot{s}$ is equal to zero. Therefore, taking the derivative of Equation 6.12;

$$\dot{s} = \dot{\breve{v}} + \lambda \breve{v} \tag{6.18}$$

Substituting $\dot{\breve{v}} = \dot{v} - \dot{v}_d$ into Equation 6.18 yields:

$$\dot{s} = \dot{v} - \dot{v}_d + \lambda \breve{v} \tag{6.19}$$

The exact dynamics of $\dot{v}$ is substituted into Equation 6.19:

$$\dot{s} = f + b\, u_c - \dot{v}_d + \lambda \breve{v} \tag{6.20}$$

Solving Equation 6.20 for u_c with $\dot{s} = 0$ yields:

$$u_c = b^{-1}\ (-f + \dot{v}_d - \lambda \breve{v}) \tag{6.21}$$

However, since f and b are not known exactly, only the best approximation of u_c can be made. Therefore, the best approximation of u_c, $u_{c_{best}}$, that would achieve $\dot{s} = 0$ is:

$$u_{c_{best}} = \hat{b}^{-1}\ \hat{u}_c, \quad \hat{u}_c = -\hat{f} + \dot{v}_d - \lambda \breve{v} \tag{6.22}$$

If exact dynamic knowledge was available, Equation 6.21 would ensure that the URV sway dynamics satisfied desired dynamics s. However, Equation 6.22 does not meet the sliding condition yet, as $\hat{f}$ is not the exact dynamics, and the robustness to uncertainties is not guaranteed. However, a discontinuous term can be introduced into the continuous control law (Equation 6.22) can provide robustness to the imperfect dynamic model. The role of this discontinuous term will be switching the control law in order to compensate the discrepancy between the exact dynamics f and the estimated dynamics $\hat{f}$. This discontinuous term is the source of the robustness of the sliding-mode control law to the imprecisions involved [59].

The discontinuous term k is incorporated into Equation 6.22 as follows:

$$u_c = \hat{b}^{-1} \ [\hat{u}_c - ksgn(s)] = \hat{b}^{-1} \left[-\hat{f} + \dot{v}_d - \lambda\tilde{v} - ksgn(s)\right] \tag{6.23}$$

where $sgn\,(*)$ is called "sign function" and is defined as:

$$sgn\,(x) = 1, \quad \text{for } x > 0$$

$$sgn\,(x) = -1, \quad \text{for } x \leq 0$$

In order to guarantee the convergence of s to zero, the value of k must be chosen such that the sliding condition is satisfied. The value of k is calculated as follows. The sliding condition (Equation 6.8) can also written as:

$$s\ \dot{s} \leq -\eta\,|s| \tag{6.24}$$

Multiplying Equation 6.20 by s gives:

$$s\ \dot{s} = s\ \ [f + b\ u_c - \dot{v}_d + \lambda\tilde{v}] \tag{6.25}$$

Substituting Equation 6.23 into Equation 6.25 yields:

$$s\ \dot{s} = \left[\left(f - b\hat{b}^{-1}\hat{f}\right) + \left(1 - b\hat{b}^{-1}\right)\left(-\dot{v}_d + \lambda\breve{v}\right) - b\hat{b}^{-1}ksgn(s)\right] s \tag{6.26}$$

Since $sgn(s)s = |s|$, Equation 6.26 simplifies to:

$$s\ \dot{s} = \left(f - b\hat{b}^{-1}\hat{f}\right) s + \left(1 - b\hat{b}^{-1}\right)\left(-\dot{v}_d + \lambda\breve{v}\right) s - b\hat{b}^{-1}k\,|s| \tag{6.27}$$

In order to find the value of k that makes Equation 6.27 satisfy the sliding condition, Equation 6.27 is substituted into Equation 6.24 as follows:

$$\left(f - b\hat{b}^{-1}\hat{f}\right) s + \left(1 - b\hat{b}^{-1}\right)\left(-\dot{v}_d + \lambda\breve{v}\right) s - b\hat{b}^{-1}k\,|s| \leq -\eta\,|s| \tag{6.28}$$

Rearranging Equation 6.28 yields:

$$k\,|s| \geq \hat{b}b^{-1}\left[\eta\,|s| + \left(f - b\hat{b}^{-1}\hat{f}\right) s + \left(1 - b\hat{b}^{-1}\right)\left(-\dot{v}_d + \lambda\breve{v}\right) s\right] \tag{6.29}$$

The maximum value of the right hand side of Equation 6.29 must be obtained to find the bound on the value of k. This can be done as follows with $\hat{b}b^{-1} > 0$ and $\eta > 0$ in mind:

$$k\,|s| \geq \hat{b}b^{-1}\eta\ \ |s| + \hat{b}b^{-1}\left|f - b\hat{b}^{-1}\hat{f}\right|\ \ |s| + \left|1 - b\hat{b}^{-1}\right|\ \ |-\dot{v}_d + \lambda\breve{v}|\ \ |s| \tag{6.30}$$

The term $|s|$ cancels out in both sides of Equation 6.30:

$$k \geq \hat{b}b^{-1}\eta + \hat{b}b^{-1}\left|f - b\hat{b}^{-1}\hat{f}\right| + \left|1 - b\hat{b}^{-1}\right|\ \ |-\dot{v}_d + \lambda\breve{v}| \tag{6.31}$$

Substituting $f = \hat{f} + \left(f - \hat{f}\right)$ into Equation 6.31 yields:

$$k \geq \hat{b}b^{-1}\eta + \hat{b}b^{-1}\left|f - \hat{f}\right| + \left|\hat{f}\right|\ \ \left|\hat{b}b^{-1} - 1\right| + \left|\hat{b}b^{-1} - 1\right|\ \ |-\dot{v}_d + \lambda\breve{v}| \tag{6.32}$$

Since $\hat{b}b^{-1} \geq 1$, this leads to:

$$k \geq \hat{b}b^{-1}\eta + \hat{b}b^{-1}\left|f - \hat{f}\right| + \left|\hat{f}\right|\ \ \left(\hat{b}b^{-1} - 1\right) + \left(\hat{b}b^{-1} - 1\right)\ \ |-\dot{v}_d + \lambda\breve{v}| \tag{6.33}$$

Rearranging Equation 6.33 yields:

$$k \geq \hat{b}b^{-1}\eta + \hat{b}b^{-1}\left|f - \hat{f}\right| + \left(\hat{b}b^{-1} - 1\right)\left(\left|\hat{f}\right| + |-\dot{v}_d + \lambda\breve{v}|\right) \tag{6.34}$$

Since $\left|f - \hat{f}\right| \leq F$ and $\beta^{-1} \leq \hat{b}b^{-1} \leq \beta$, this leads to:

$$k \geq \beta\left(F + \eta\right) + \left(\beta - 1\right)\left(\left|\hat{f}\right| + |-\dot{v}_d + \lambda\breve{v}|\right) \tag{6.35}$$

By choosing k depending on Equation 6.35, the control law (Equation 6.23) guarantees the stability and perfect tracking for the URV sway motion as long as the bounds on the dynamic parameters are not exceeded.

The control law drives the system state variable onto the sliding line and keeps it on the line in spite of parameter changes and disturbances. Once the system state is trapped on this line, it is constrained to move along the sliding line provided that the sliding condition is satisfied [31].

Sliding-Mode Control Tuning

The control law is composed of two parts; the continuous part $\hat{u}_c$, which is model based and is the best approximation that would achieve $\dot{s} = 0$, and the discontinuous part, $ksgn(s)$, which is a nonlinear feedback component. If the dynamic model is perfect, the continuous part alone is able to keep the state on the sliding line and the state value converges to the desired point exponentially with a time constant equal to $1/\lambda$. The discontinuous term works only when the system state variable moves away from the sliding line due to the presence of model uncertainty or any disturbance effect. It basically pushes the system state back on the sliding line in the event the system state variable go away from the sliding line, *i.e.*, $s \neq 0$.

The control bandwidth λ contributes directly to the tracking performance of the sliding-mode controller in view of Equation 6.7. Higher values provide better tracking

performance. However, there are some limiting factors to the choice of the value of λ. First of all, it must be smaller than the smallest unmodelled structural mode in the system. Likewise, the frequency of sampling rate should not be smaller than λ. Lastly, the frequency of neglected time-delay in actuators should not be smaller than λ. In case one of these constraints is not met, the system can be driven to an unstable dynamic behavior.

The gain k is a function of the system state and possibly time. The value of k is dependent primarily on the degree of the mismatch between the actual and estimated system dynamics. If the accuracy of parameter estimates are high, then the value of k can be smaller. The smaller k results in the reduction of control energy. The gain contributes directly to the control input $\hat{u}_c$ and thus is also related to the energy consumed by the actuators.

The value of η sets the maximum time required for the system state variables to hit the sliding surface. It is chosen to be small compared to the average value of k. As with k, the term η is associated with the control activity, *i.e.,* larger η values result in higher control activity and hence control energy.

Chattering Reduction

The term $ksgn(s)$ in the control law equation implies switching control action across the sliding surface due to the nature of the sign function. However, in order to obtain a good performance from the sliding-mode controller, infinitely fast switching rates may be needed. However, infinitely fast switching actions are not possible due to the limited actuator slew. In addition, the switching rate of the control signal is limited by the sample rate of the state measurements The control signal stays constant during the sample time, and the system state can diverge from the sliding surface due to the

model uncertainty or any disturbance effect. However, this is not an issue in this thesis since the URV state data can be pulled at any time from the simulator. The switching control slew leads to the oscillations within a neighborhood of the sliding surface and is called chattering. The chattering is not desirable since it may cause the actuators to wear out prematurely. Furthermore, it may excite unmodelled high frequency dynamics of actual dynamics f. Chattering must be eliminated to obtain optimal control performance.

Chattering can be eliminated by allowing the continuous control action to work within a wider region around the sliding surface. This region is defined by a boundary layer. In the boundary layer, the switching term does not work, and therefore relaxes the control law. The effect of boundary layer can be accounted for by replacing the $sgn(s)$ function in the control law with the saturation function $sat(s)$, which is defined as:

$$sat\,(s) = sgn(s), \quad \text{for } |s| > \Phi \tag{6.36}$$

$$sat\,(s) = \frac{s}{\Phi}, \quad \text{for } |s| \leq \Phi \tag{6.37}$$

where Φ is the width of the boundary layer as illustrated in Figure 6.5 for the URV sway state. Outside the region defined by the boundary layer, the switching term is activated as dictated by Equation 6.36. Since the sliding condition is satisfied, the system state is drawn into the region and finally is trapped in the region as long as the foreseen bounds on the parameter estimations and disturbances are not exceeded. Replacing $sgn(s)$ with $sat(s)$ and introducing the boundary layers leads to a tracking performance worse than what the control law with $sgn(s)$ alone offers. This implies a trade-off between the tracking precision and the chattering in view of Equation 6.7.

The guaranteed precision that is found for a second-order system using Equation 6.7 as

$$\epsilon = |\check{y}_0|_{\max} = \frac{\Phi}{\lambda} \tag{6.38}$$

Given that the tracking performance is bounded by Equation 6.38, the quantitative approach to making a trade-off can be made, *i.e.*, for a given control bandwidth λ, using a bigger boundary width Φ results in a reduction of the control performance while improving the elimination of the chattering problem. It is obvious that the tracking precision can be increased by setting the control bandwidth to higher values. However, this adjustment is not always possible due to the constraints on the choice of the value of the control bandwidth as explained before.

To summarize, with the reference to Equation 6.23, the control input is given by:

$$u_c = \hat{b}^{-1}\left[-\hat{f} + \dot{v}_d - \lambda\check{v} - ksat(s/\Phi)\right] \tag{6.39}$$

where $\hat{b}$ is obtained from Equation 6.17 , k is obtained from Equation 6.35. The resulting control law gives a guaranteed precision that is given by Equation 6.38. In Figure 6.5, the theoretical maximum deviation of the sway state from the sliding line is labelled as ϵ.

6.3 Thruster Controller

Once the control input is obtained according to Equation 6.39, it remains to create this control input using the URV thrusters. However, since the thrusters are used to provide the control forces and torques, the control forces and torques must be distributed over the thrusters in a way that the net force and moment effect due to

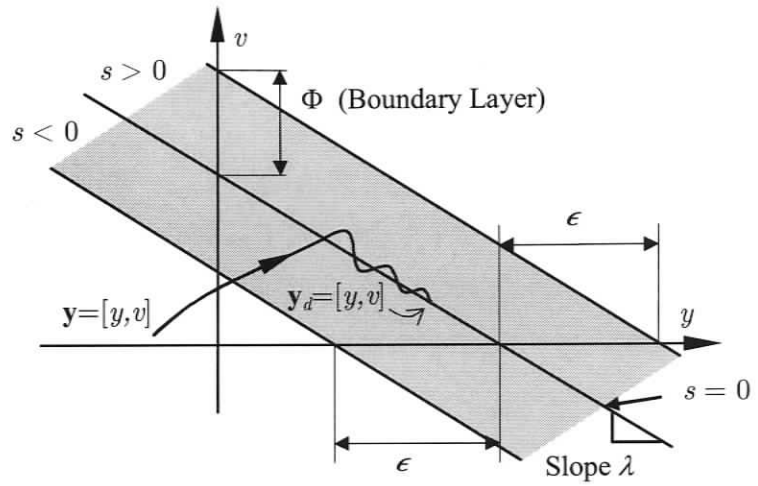


Figure 6.5: Boundary Layers

the thrusters are the same as the required ones. This distribution is carried out by the thruster mapping matrix. As a result of the thruster mapping process, the desired thrust command for each thruster is generated. Every thruster has its own controller that is responsible for generating the desired thrust.

A model-based non-linear thruster control algorithm called model-based velocity control is used in [18] to generate the control signal that makes the actual thrust follow the desired thrust. In the model-based velocity control, the work of Healey *et al.* [16] is followed to model the thrusters. The control methodology requires state feedback. The thruster controller specifies the motor voltage v_a as a function of the desired thrust τ_d, the measured propeller rate Ω, and observed axial incoming fluid particle speed u_a. The controller ensures that the actual trust follows the desired thrust, *i.e.*, $\lim_{t\to\infty} \tau = \tau_d$.

In practical applications, the measurement of state u_a is extremely difficult or impossible [18]. Therefore, an ad hoc open-loop observer is used to estimate u_a.

The observer estimates u_a based on Equation 4.41 as follows:

$$\dot{u}_a^e = -k_4 k_3^{-1} \bar{u}_a^e \left| \bar{u}_a^e \right| + k_3^{-1} \tau_d \tag{6.40}$$

where u_a^e is the estimate of u_a.

According to Healey *et al.* [16], thrust is a non-linear function of axial incoming fluid velocity and the propeller rate. Based on the explanations in section 4.3, the developed thrust is defined by:

$$\tau = g_1 \left(\Omega, u_a^e \right) \tag{6.41}$$

The hydrodynamic load on the propeller shaft is given by:

$$\tau_L = g_2 \left(\Omega, u_a^e \right) \tag{6.42}$$

The corresponding desired propeller rate Ω_d for a given desired thrust is obtained by inverting function g_1:

$$\Omega_d = g_1^{-1} \left(\tau_d, u_a^e \right) \tag{6.43}$$

In order to solve Equation 6.43 for Ω_d for a given desired thrust, the MatLab® command *fsolve* can be used. This command solves nonlinear equations numerically. Having found the desired propeller rate and measured the current axial incoming fluid velocity and the current propeller rate, the control law i_a^c is given by:

$$i_a^c = k_m^{-1} \; g_2 \left(\Omega, u_a^e \right) + k_m^{-1} \; k_{fb} \left(\Omega - \Omega_d \right) \tag{6.44}$$

where $k_{fb} < 0$ is the empirically tuned feedback gain. Equation 6.44 combines a model-based open-loop feedforward control, $k_t^{-1} \; g_2 \left(\Omega, u_a^e \right)$ and high-gain feedback of

propeller rotation rate error, $k_t^{-1}\ k_{fb}\,(\Omega - \Omega_d)$. Since the control input is voltage, the corresponding control voltage must be found. This is found by:

$$v_a^c = r\ i_a^c + k_b\ \Omega_d \tag{6.45}$$

The control scheme is illustrated in Figure 6.6. The desired thrust and the esti-

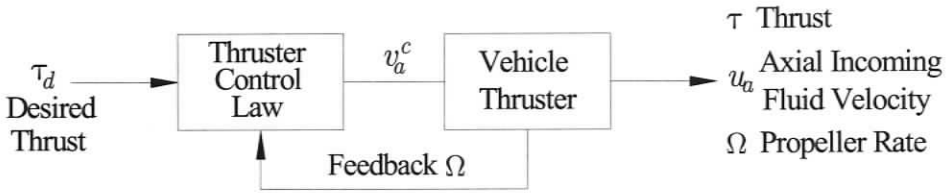


Figure 6.6: Thruster Control Scheme

mate of the axial incoming fluid velocity are provided to the model-based velocity thruster controller. In the thruster controller, as a first step, the corresponding desired propeller rate is found based on Equation 6.43 for the given desired thrust. Having found the desired propeller rate and provided the estimate of u_a, the control voltage is generated using Equations 6.44 and 6.45. The estimate of u_a at a later time is found by numerically integrating Equation 6.40 forward in time.

In order to verify the thruster control methodology, a simulation program was created. In the thruster simulation, the state variables are Ω, u_a and u_a^e. Note that u_a represent the axial fluid velocity while u_a^e is the estimate of u_a, which is computed by the observer within the propeller shroud. The control input is formed based on Equations 6.44 and 6.45. The resulting control input is fed into the thruster dynamic equations described in Section 4.3.

In Figure 6.7, the corresponding thrust response to the commanded thrust value of 7 N is illustrated. All the initial state variables are taken as zero. The feedback

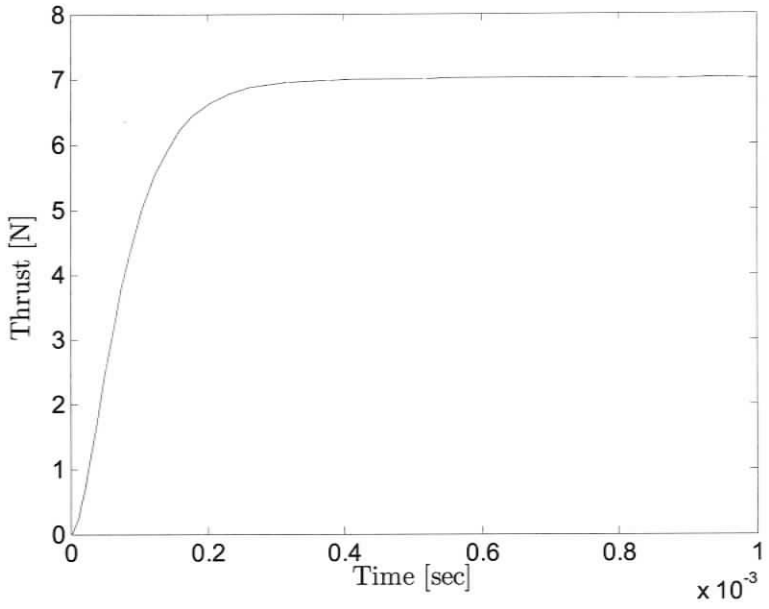


Figure 6.7: Thrust History

gain k_{fb} is found by trial and error as -1 and all the associated thruster parameters are given in Appendix E. As it can be seen in the figure, there is a time delay in obtaining the desired thrust. This result reaffirms the need for a robust controller, as a robust controller can compensate the errors due to the time delay.

Figure 6.8 shows the time history of the propeller rate of the thruster. As expected, once the desired thrust is obtained, the propeller rate stays constant.

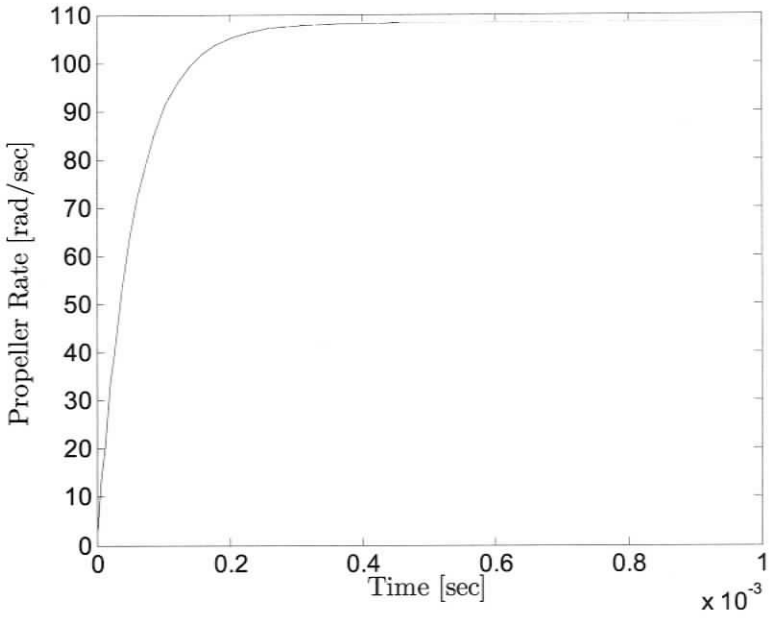


Figure 6.8: Time History of the Controlled Propeller Rate

Figure 6.9 demonstrates the time history of u_a and u_a^e. As expected, these two quantities are different. However, the observer is quite successful in predicting the actual value of u_a.

6.4 Reduction of the Dynamic Coupling in URVM Systems

A number of simulations have been performed to demonstrate the ability of the proposed control scheme to reduce the dynamic coupling between the URV and the manipulator. The URVM system considered in the simulations is the same as the one considered in Section 5.3. The thruster configuration uses two surge thrusters,

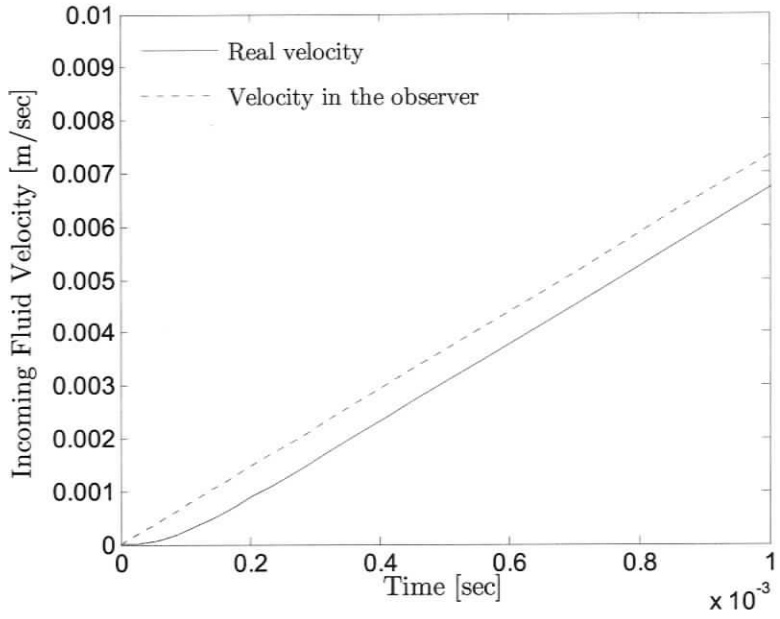


Figure 6.9: Time History of the Incoming Fluid Velocity

that can be operated differentially for yaw motion, and two sway thrusters as illustrated in Figure 6.10. In order to avoid the problem of the allocation of thruster forces among the thrusters (see section 4.46), the commanded moments around the yaw axis of the vehicle are evenly distributed between the surge and sway thrusters. The compensation force along the surge axis of the vehicle is evenly distributed over the two surge thrusters. Likewise, the compensation force along the sway axis of the vehicle is evenly allocated between the two sway thrusters. However, this allocation of the commanded forces/moments does not exploit the redundancy that can be used to enhance the ability of the URV to accomplished a desired task. The exploitation of redundancy in thruster systems is addressed in [61]. The simulation parameters and the state vector structure for the URV and the manipulator are the same as in Section 5.3.3. In the simulation, the same station keeping task is considered to demonstrate

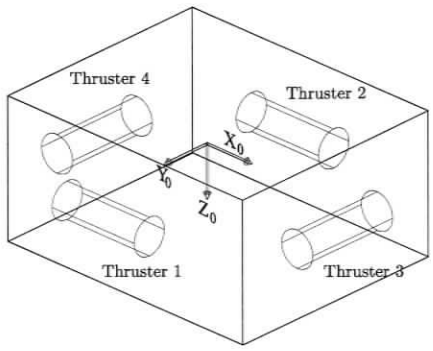


Figure 6.10: Thruster Configuration of the URV

the reduction of the dynamic coupling and to afford comparison to the uncontrolled case of Section 5.3. In the simulation, a station keeping task is considered for the URV to demonstrate the reduction of the dynamic coupling. In the station keeping task, the sliding-mode controller's objective is to keep the URV's state vectors zero, *i.e.*, $\underline{\mathbf{x}}_0 = 0$ and $\underline{\mathbf{v}}_0 = 0$, in the presence of dynamic modelling uncertainties and disturbances caused by the dynamic coupling due to the manipulator motion. In order to model uncertainties, drag and added mass coefficients for the URV differed from the real system by 60%. Furthermore, the thruster dynamics are not included into the model-based sliding-mode controllers. Only the URV's motion in the horizontal plane, *i.e.*, $\begin{bmatrix} x & y & \psi \end{bmatrix}^T$ is controlled.

The simulation scheme is illustrated in Figure 6.11. In the simulation, the URVM state vector $\mathbf{u}_s = \begin{bmatrix} {}^E\underline{\mathbf{x}}_0^T & {}^E\underline{\mathbf{v}}_0^T & \mathbf{q}^T & \dot{\mathbf{q}}^T & \Omega^T & \mathbf{u}^T \end{bmatrix}^T$ proceeds through the articulated-body equations with estimated parameters. Since the horizontal motion is controlled, only the URV's surge $\dot{u}$, sway $\dot{v}$ and yaw $\dot{r}$ equations of motion are needed for the controllers. Given that the single-input single-output sliding mode

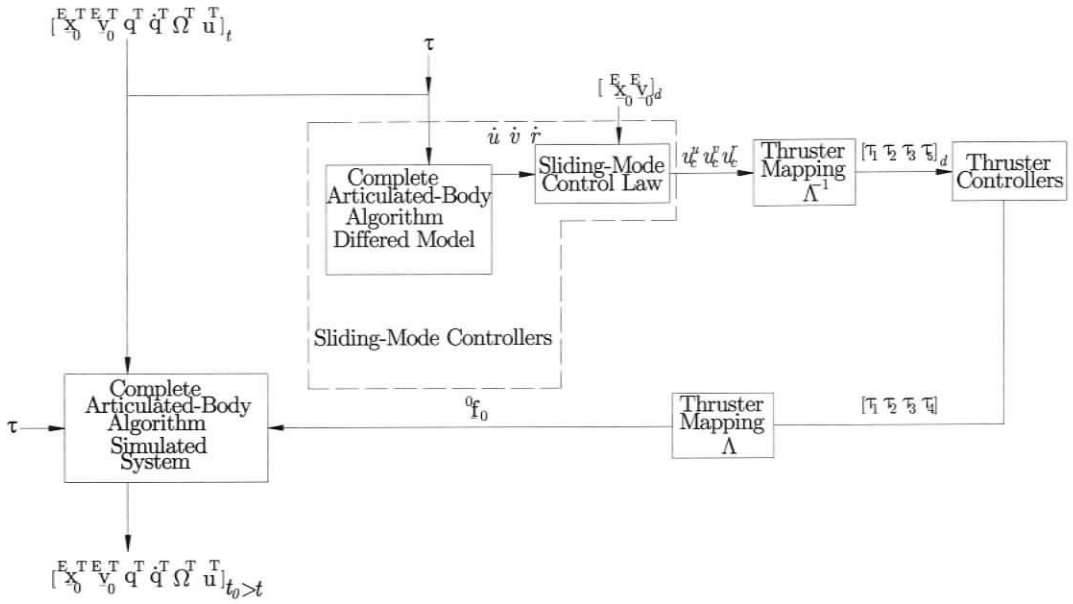


Figure 6.11: Simulation Scheme

approach is employed, it is required to have three controllers, *i.e.*, every degree of freedom of the URV has its own controller. Note that since the system dynamics are highly coupled, every degree of freedom of the URV is a function of the other degrees of freedom, and any coupling effects due to the motion of the other degrees of freedom are treated as a disturbance in each controller. The control signals u_c^u, u_c^v and u_c^r are generated by the sliding-mode controllers to control the URV's horizontal motion. Using the thruster mapping, a desired thrust for each thruster is calculated and fed to the individual thruster controllers. The actual thrust developed by each thruster controller is mapped back into the forces and moments at the centre of mass of the URV, ${}^0\underline{\mathbf{f}}_0$. The force and moment vector ${}^0\underline{\mathbf{f}}_0$ along with the URVM state vector $\mathbf{u}_s$ are then inserted into the articulated-body equations that represent the simulated system. The state vector of the URVM system at a later time is found by integrating

forward in time.

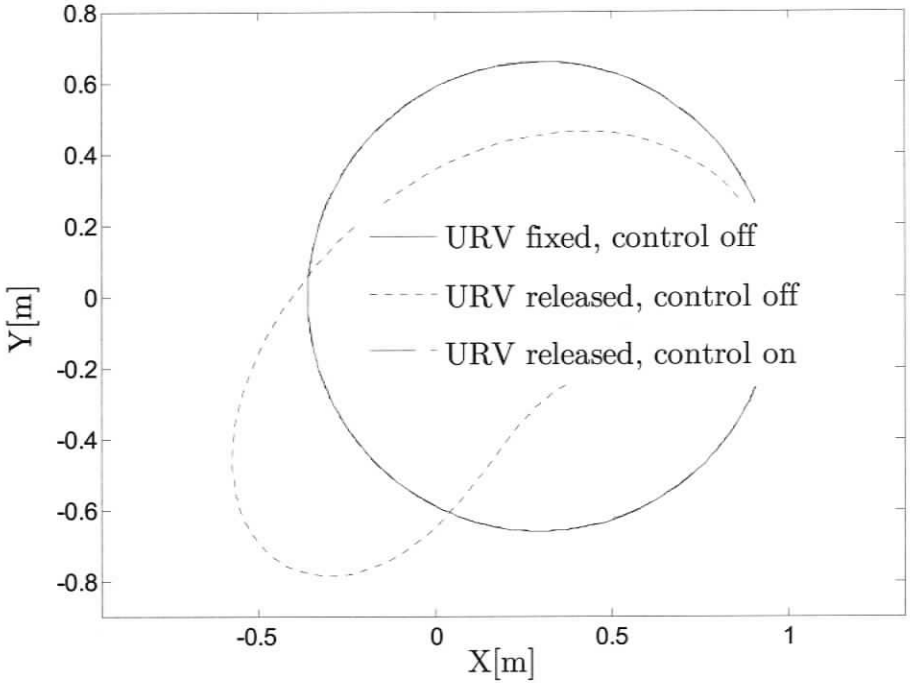


Figure 6.12: End Effector Path on X-Y Plane

Figure 6.12 demonstrates the resulting end-effector tracking path on the X-Y plane under the controlled and uncontrolled conditions. The first case in Figure 6.12, URV fixed control off, corresponds to the ideal case in which the URV is held fixed and the system behaves as a land-based manipulator. The end-effector path of this case is considered to represent the pilots true intent. In the second case, URV released-control off, the URV is released with no active control. In the third case, URV released-control on, the URV is released and the control is turned on. From Figure 6.12, one can see that the controller works to provide a fixed-base for the manipulator by reducing the dynamic coupling effect between the URV and the manipulator. The control-on case almost overlaps the ideal case.

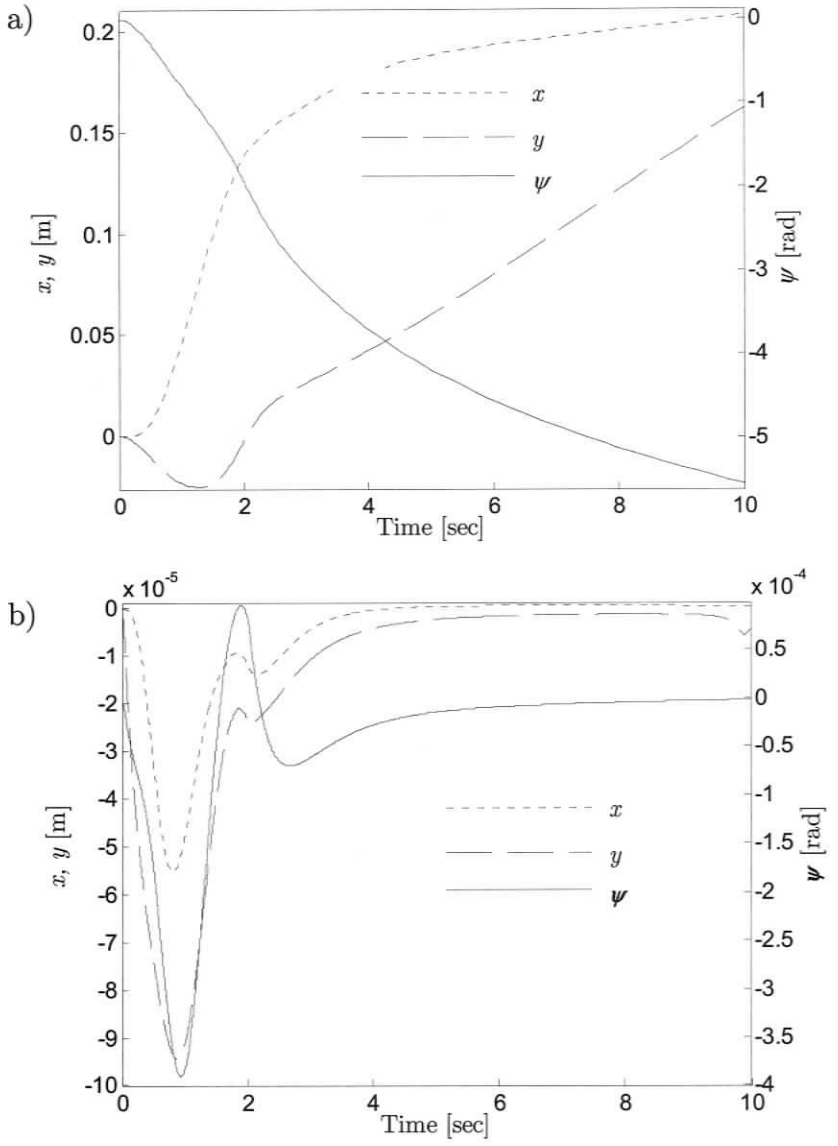


Figure 6.13: a) Vehicle Translational and Rotational Motion History without Control Activity b) Vehicle Translational and Rotational Motion History with Control Activity

The URV position error caused by the dynamic coupling is demonstrated in Figure 6.13. While Figure 6.13a shows the URV position error for the uncontrolled system,

Figure 6.13b shows it for the controlled system. Note that the displacements are in terms of the URVs body-fixed frame. As can be seen in Figure 6.13, the sliding-mode controller reduces the error scale from 10^{-1} to 10^{-5}. This stems from the fact that the controller is successful in reducing the dynamic coupling between the URV and the manipulator.

Figure 6.14a and 6.14b are the plot of the boundary layers and s, which indicates the discrepancy between the resulting motion and the desired motion. Note that Figure 6.14b is the same as Figure 6.14a except for the scale. As predicted by the sliding-mode control theory, the value of s does not exceed the boundary layer value of 0.02, *i.e.*, $|s| \leq \Phi$. Figure 6.14 also demonstrates that s remains inside the boundary layers throughout the operation. This also indicates that the bounds on the dynamic parameters are not exceeded.

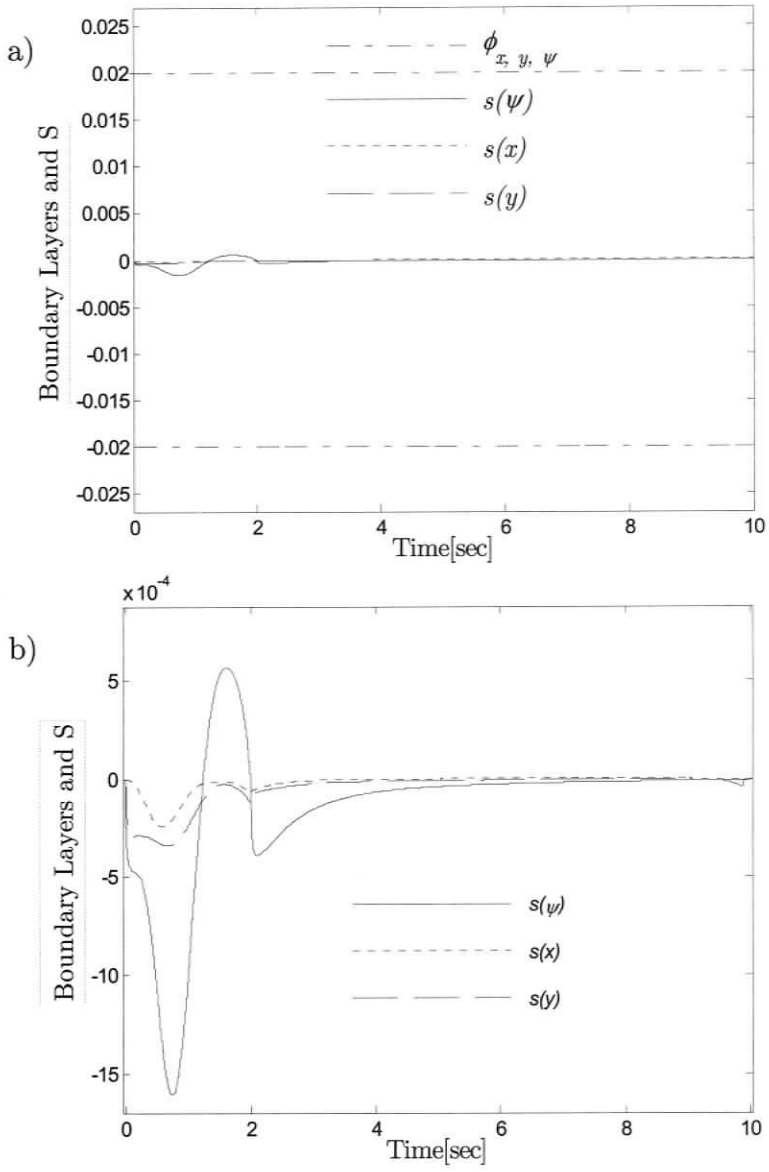


Figure 6.14: a) Boundary Layer versus s b) s Graph with a Smaller Scale

Figure 6.15 shows the forces and moments at the center of mass of the URV commanded by the controller. One can see that the commanded forces and moments are smooth and free of chattering. This is attributed to the presence of the boundary

layer, which relaxes the control law within the boundary layer and in turn provides smooth control outputs. The boundary layer thickness was chosen using trial and error to ensure that chattering was not encountered by the actuators. Smaller boundary layer values can cause chattering of the thrusters shafts.

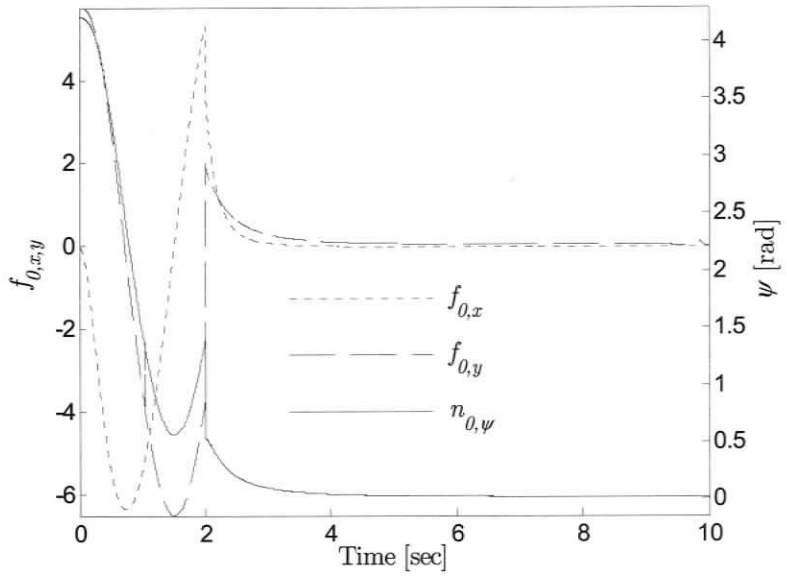


Figure 6.15: Resulting Thruster Force and Moments Acting at the Center of Mass of the URV

Time histories of the state variables are given in Figures 6.16 through 6.18.

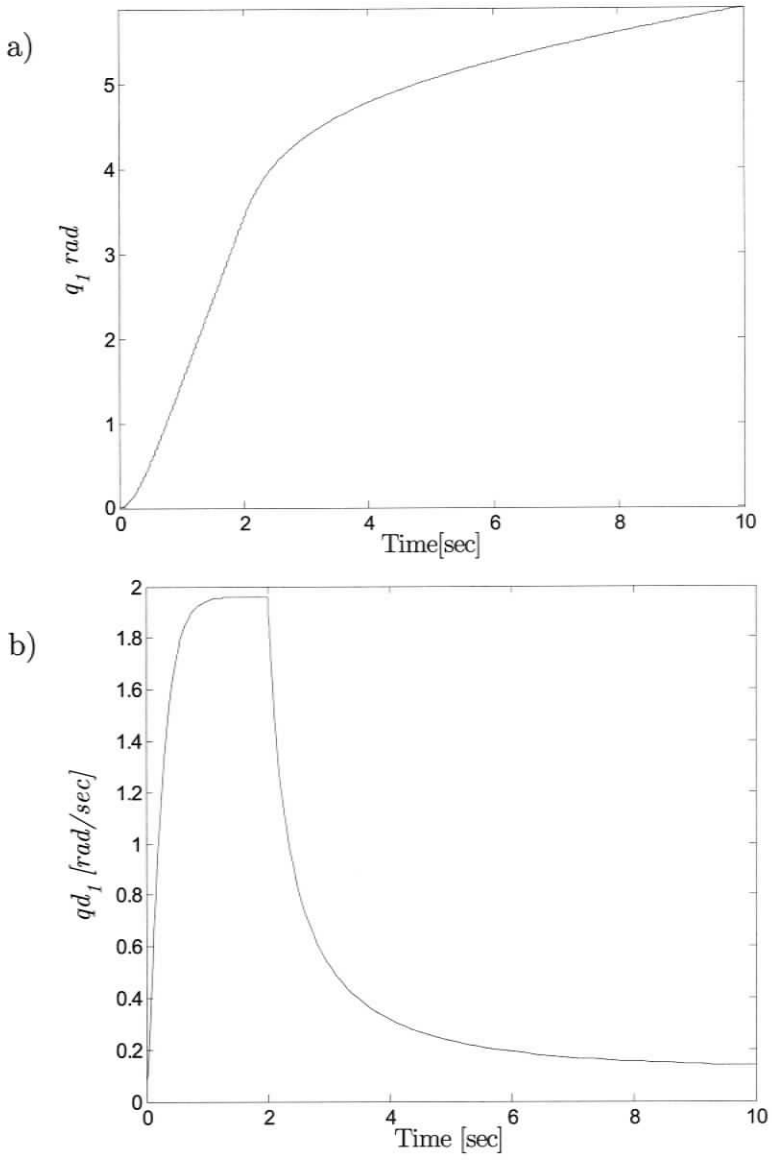


Figure 6.16: a) Manipulator Joint Displacement History b) Manipulator Joint Velocity History

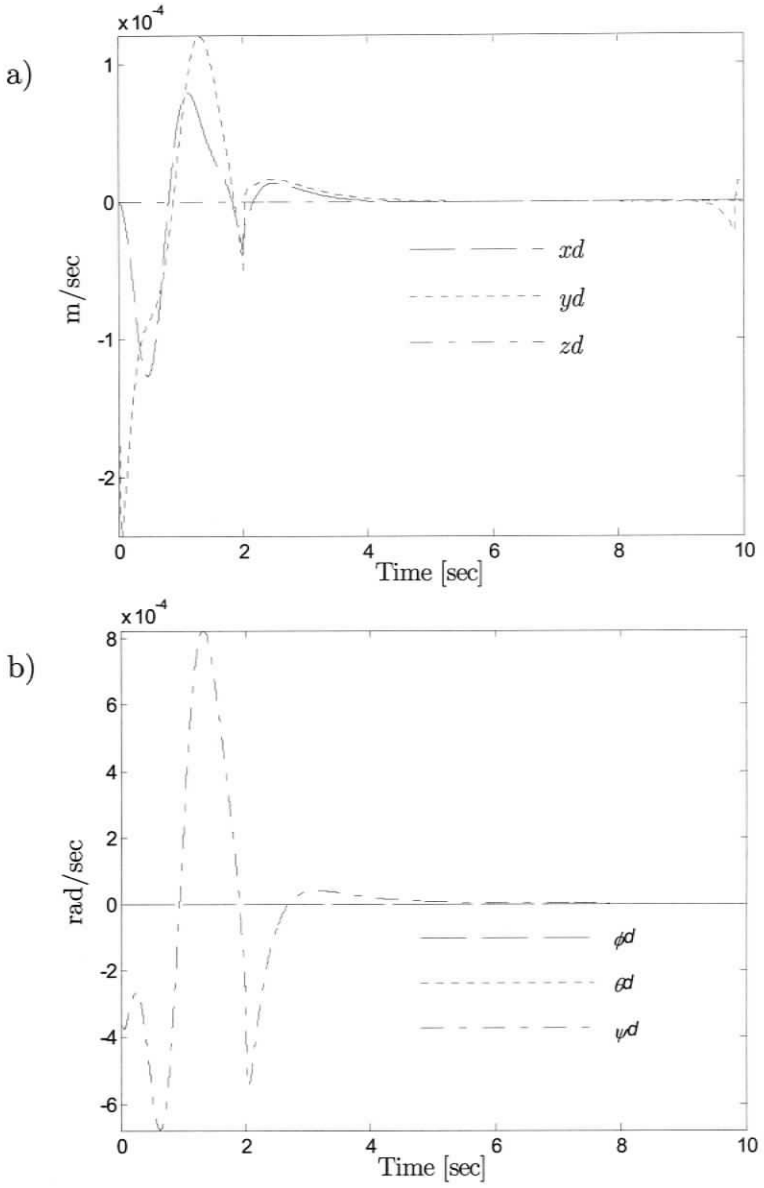


Figure 6.17: a) URV Translational Velocity History b) URV Rotational Velocity History

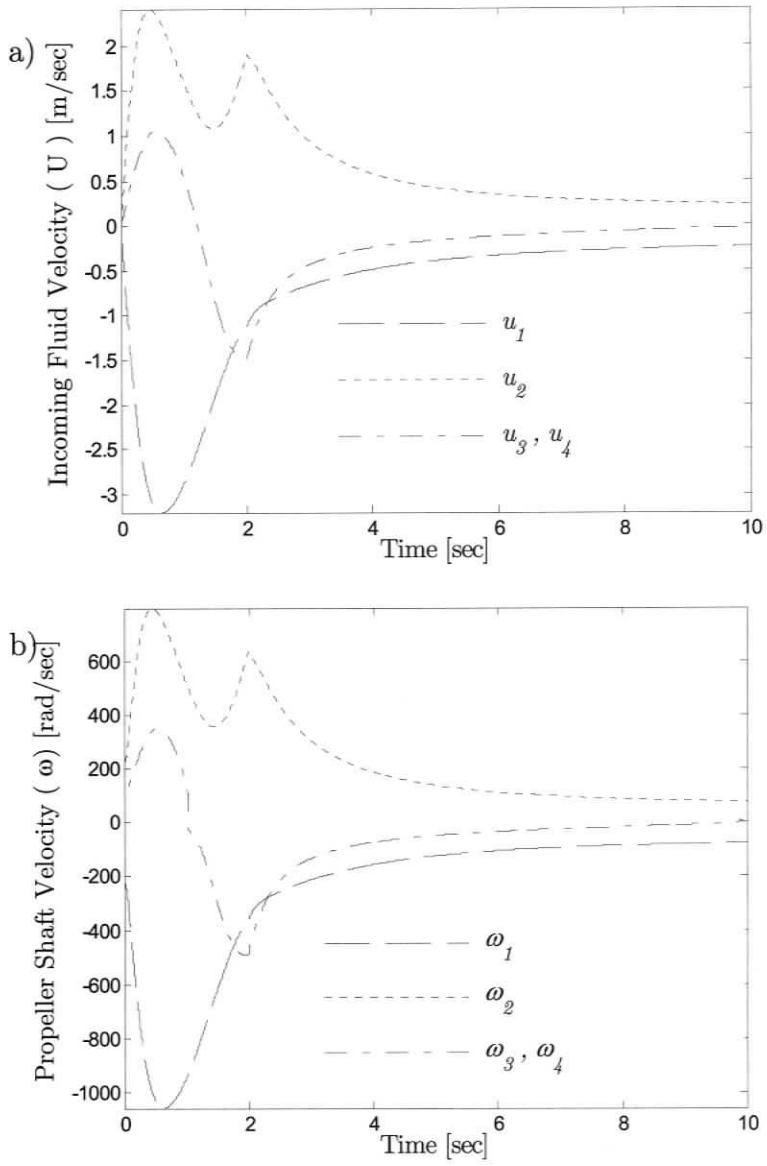


Figure 6.18: a) Incoming Fluid Velocity b) Propeller Shaft Velocity

Figures 6.19, 6.21 and 6.23, demonstrate the reaction forces that develop at the mounting point between the URV and the manipulator according to Equation 3.46. As shown in Figures 6.20, 6.22 and 6.24, these reaction forces are actively compen-

sated by the thruster forces developed by the three model-based sliding-mode controllers.(See Figure 6.11 for the control hierarchy). As a result, the URV can maintain its position and orientation in the presence of the manipulator joint motions. By using the automatic thruster controller, the human pilot can operate the manipulator without providing thruster inputs simultaneously.

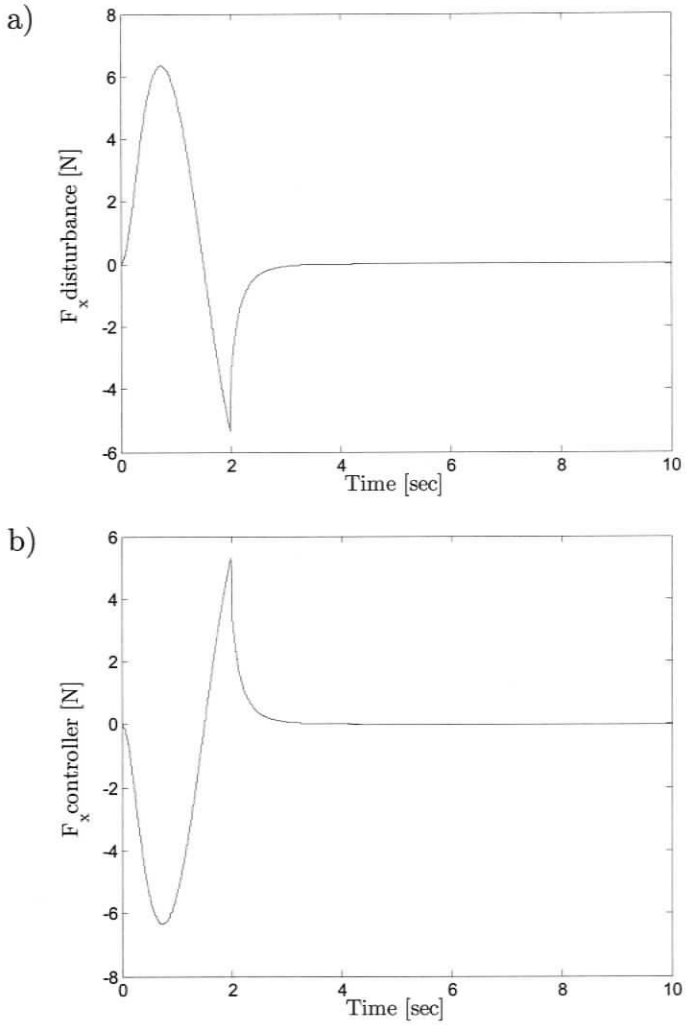


Figure 6.19: a) Force Acting Along Xo Direction Caused by Dynamic Coupling b) Compensation Force Acting Along Xo Direction of the URV by Thrusters

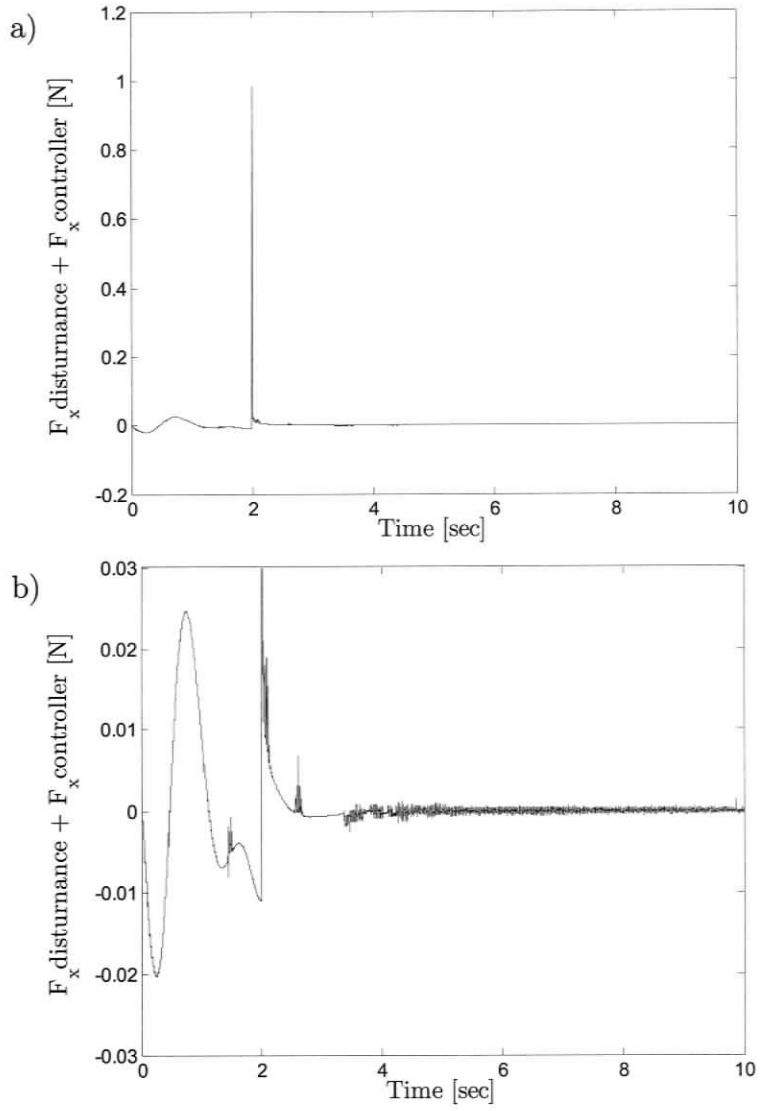


Figure 6.20: a) Difference Between the Disturbance Force and the Control Force Along Xo Axis of the URV b) Same Graph with a Different Scale

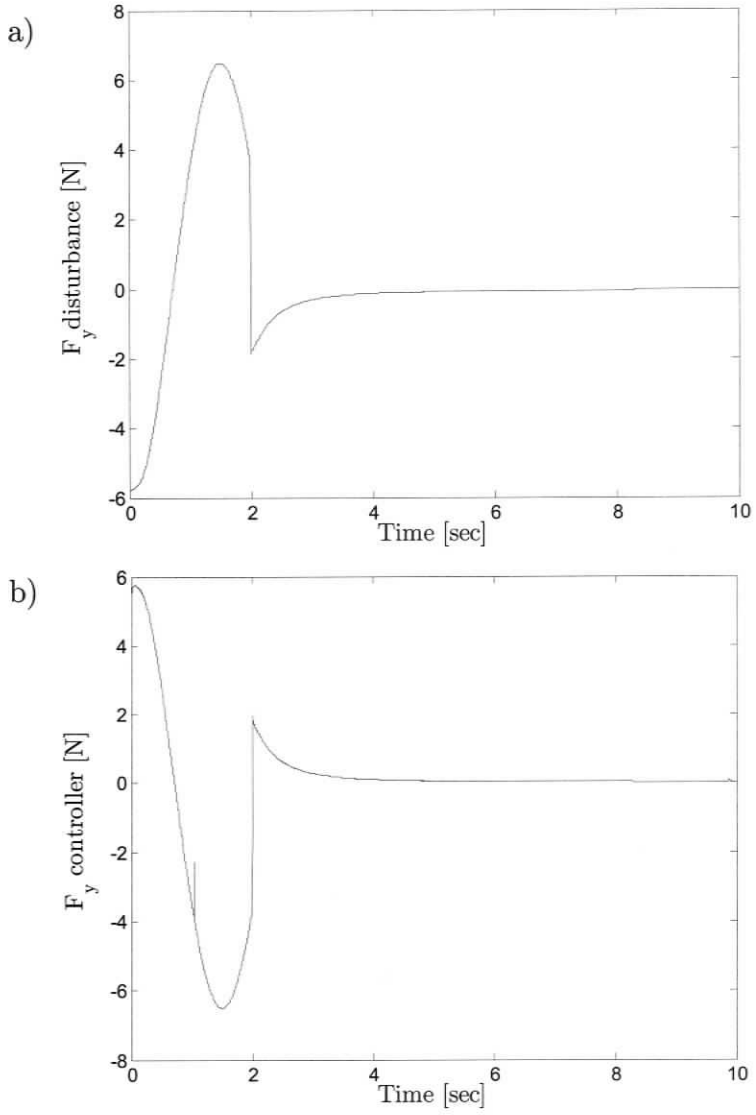


Figure 6.21: a) Force Acting Along Yo Direction Caused by Dynamic Coupling b) Compensation Force Acting Along Yo Direction of the URV by Thrusters

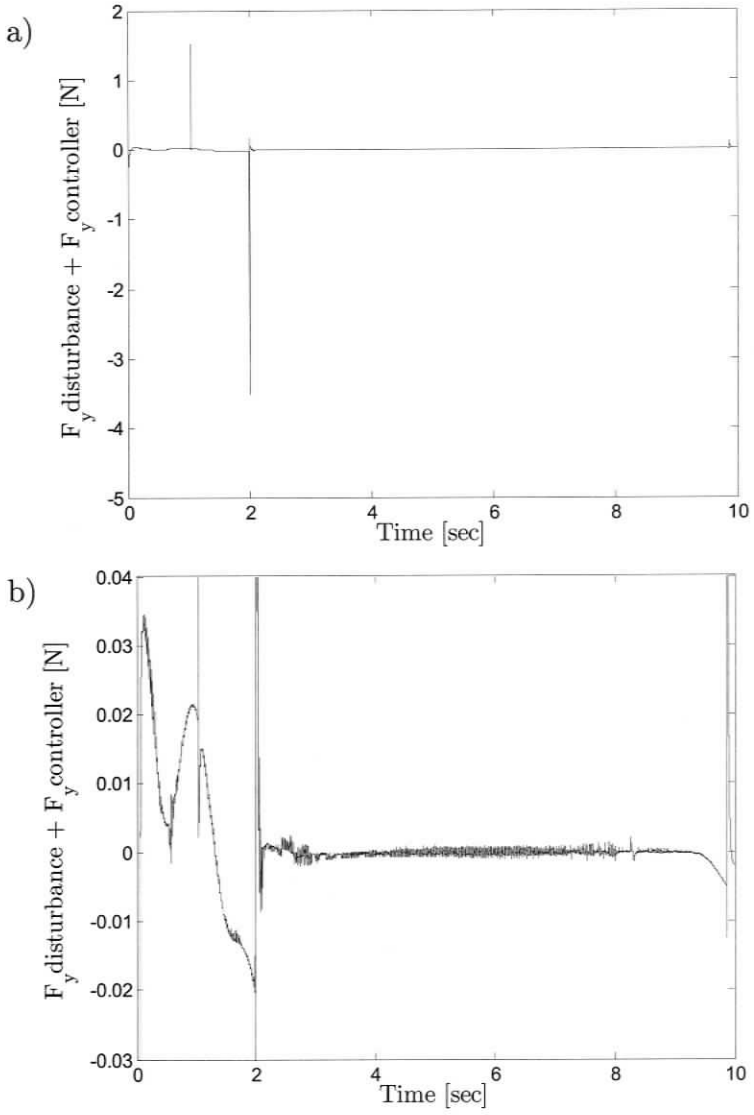


Figure 6.22: a) Difference Between the Disturbance Force and the Control Force Along Yo Axis of the URV b) Same Graph with a Different Scale

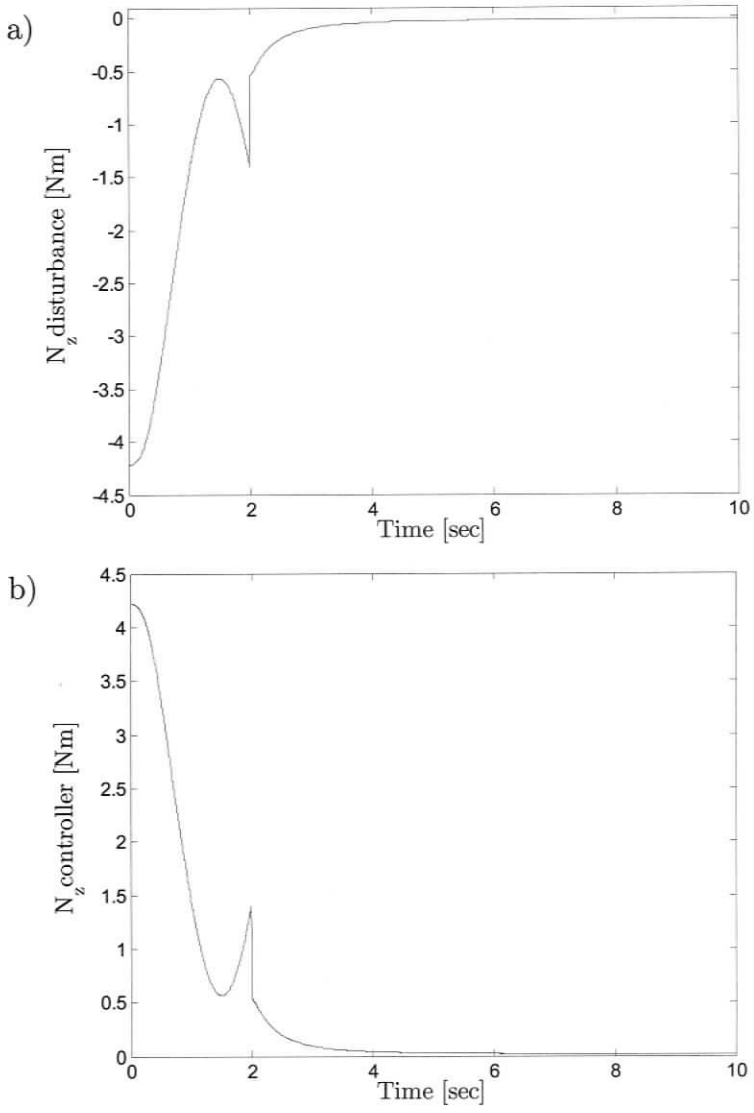


Figure 6.23: a) Moment Acting Around Zo Direction Caused by Dynamic Coupling b) Compensation Moment Acting Around Zo Direction by Thrusters

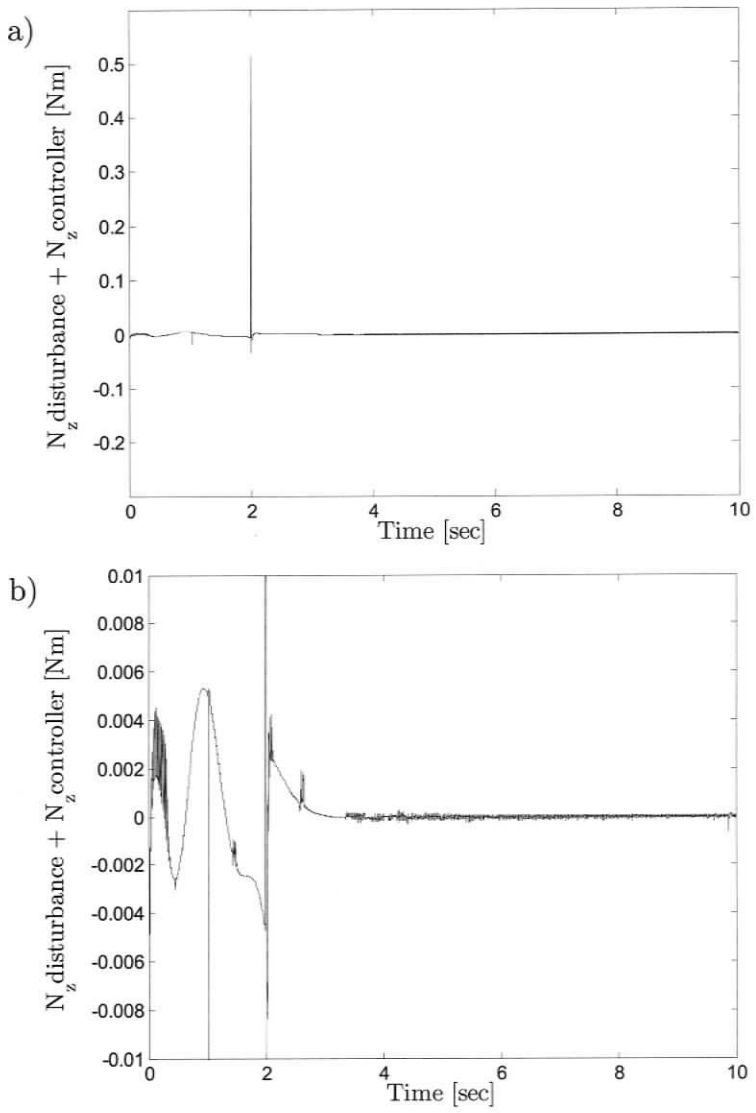


Figure 6.24: a) Difference Between the Disturbance Moment and the Control Moment Around Zo Axis of the URV b) Same Graph with a Different Scale

6.5 Comparative Study Between Newton-Euler Formulation and Articulated-Body Algorithm

A comparative study has been done to demonstrate the performance of two different dynamic modelling methods in the sliding-mode controllers, *i.e.,* the articulated-body approach of Section 6.2 and a Newton-Euler model of [44]. To do that, a model-based sliding-mode controllers was constructed based on the Newton-Euler algorithm. In both controllers, drag and added mass coefficients for the URV and the manipulator differed from the real system by 60%. In addition, the added mass Coriolis effect was not included in both controllers to simplify the dynamics equations. It was assumed that inertial measurement units were not available for the measurement of the URV acceleration. The Newton-Euler and the articulated-body based sliding-mode controllers were run for the case of Section 6.4. To emphasize the better ability of the articulated-body based controller to predict the manipulator added mass load, fluid velocities of 1 m/sec. along the X and Y directions of the inertial frame was added. This impulsive load can also be considered to be a tether jerk. This abrupt load causes a URV acceleration which is not captured by the Newton-Euler method.

In the first case, the surrounding fluid has a velocity of 1 m/sec. along X and Y directions of the inertial frame and is acting on the system for 0.2 sec. This creates impact effect on the URV, and in turn causes the URV to accelerate along the X and Y directions. End-effector absolute errors for the both controllers are shown in Figure 6.25. Using the area underneath the curves as a comparison method, Figure 6.25 indicates that the articulated-body based controller provides 10% improvement over its Newton-Euler counterpart. The range of the improvement changes with the extent of disturbances acting on the system. The improvement can be attributed

to the fact that the effect of the URV's acceleration on the manipulator dynamics and the effect of added mass of the manipulator on the URV are captured by the articulated-body algorithm during the first 0.2 sec. of the simulation when the URV is subjected to an impulsive load.

Figure 6.25 exhibits constant end-effector errors even though the URV moves back to its original position after the impulsive loads are applied to the URVM system as illustrated in Figure 6.26. This is due to the fact that applied torques to the manipulator joints remain the same as those in the ideal case of Section 5.3.2 during the system motion even though the dynamic loads on the manipulator after the perturbation are different. As a result, the end-effector of the URVM produces a different trajectory path. In URVM applications, this difference is compensated by the human pilot in a open-loop control structure. Using the articulated-body equations in the controller reduces the human pilot's effort to compensate for the deviations from the desired end-effector path.

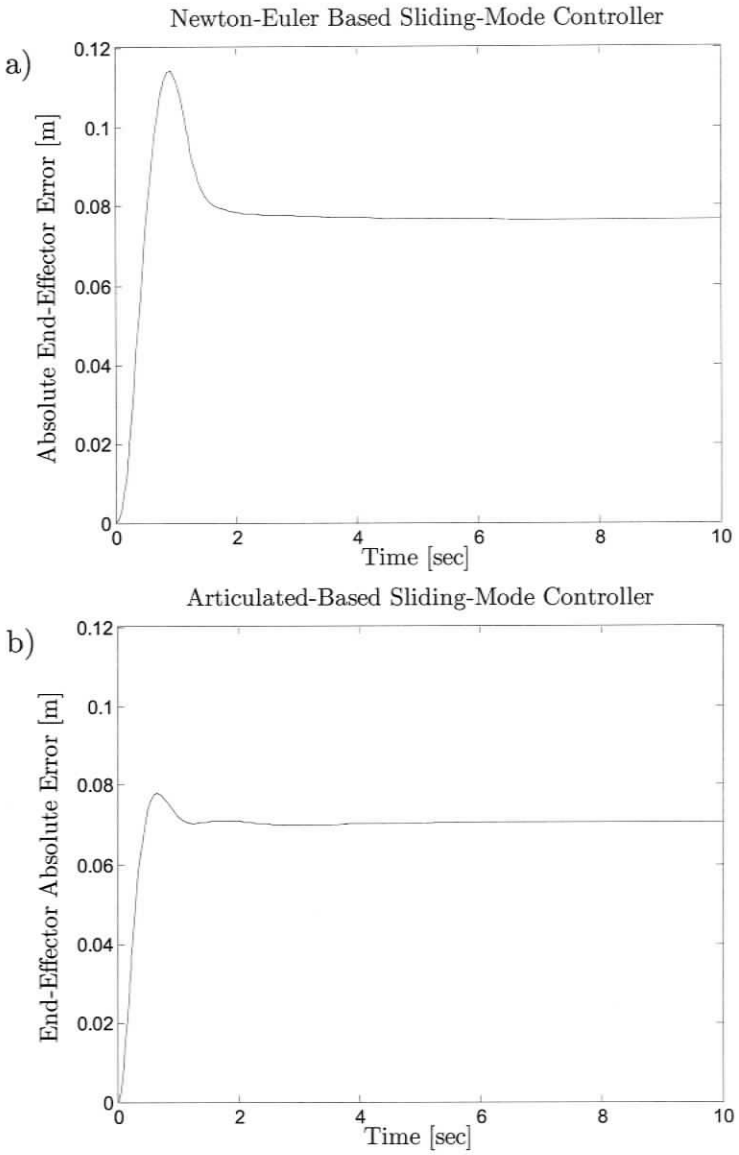


Figure 6.25: End-Effector Absolute Errors When Fluid Velocity 0.1 m/sec Along X and Y Directions of the Inertial Frame

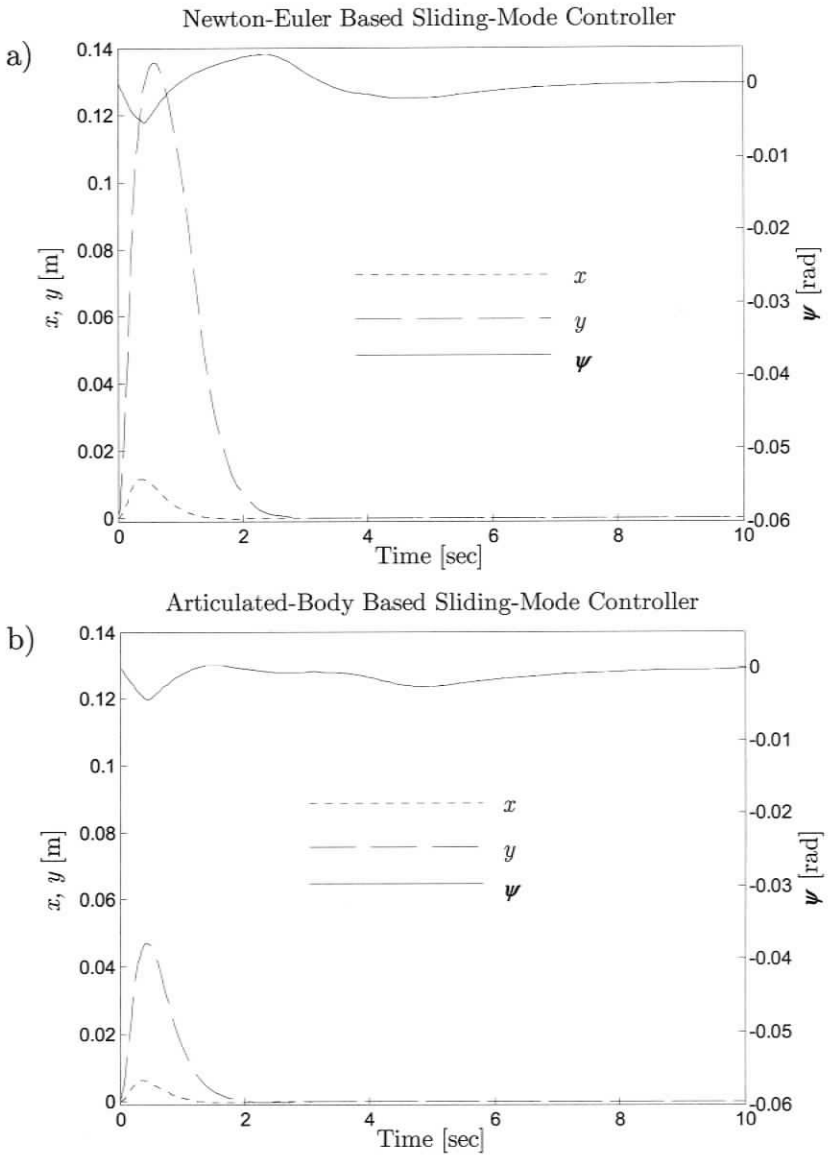


Figure 6.26: URV Positionand Orientation Errors When Fluid Velocity 1 m/sec Along X and Y Directions of the Inertial Frame

Chapter 7

Conclusions and Recommendations for Future Work

7.1 Overview

In this chapter, conclusions about the thesis will be presented. In addition, recommendations will be made about possible areas where future research could be conducted.

7.2 Conclusions

A generic time-domain dynamics simulator for URVM systems has been created based on the articulated-body algorithm. The thruster dynamics has been incorporated into the existing articulated-body equations. For a specific URVM system, a number of simulations have been performed and the results indicating the extent of the dynamic coupling problem have been presented. It is proposed that the thrusters be used

to offset the dynamic coupling effect on the URV motion by employing intelligent automatic control of the URV thrusters using Slotine's sliding-mode control strategy.

It is also proposed that the articulated-body algorithm be used within the model-based sliding-mode controller. The incorporation of the articulated-body algorithm into a model-based sliding-mode controller for the reduction of the dynamic coupling between the URV and the manipulator has been accomplished. The use of the articulated-body equations within the sliding-mode controller has not been introduced before in literature.

In order to show the effectiveness of the proposed control scheme, the simulation of a URVM system has been performed and the results have been demonstrated. The simulation results have indicated that the tracking performance of the URVM system is improved by the inclusion of the articulated-body algorithm into the controller with respect to the sliding-mode controllers that use Newton-Euler dynamics formulation for the URV and the manipulator separately. In the Newton-Euler formulation, the URV acceleration is ignored in the added mass force calculation of manipulator, unless expensive inertial measurement units are used to predict the acceleration at the mounting point of the manipulator. However, the omission of the URV acceleration within the manipulator dynamic equations leads to a reduction in the quality of the estimation of the URV dynamics within the controller. It is shown that using the articulated-body equations for the URV equations of motion can eliminate the need for the inertial measurement units while improving the quality of the estimation of the URV dynamics. This results from the two facts *i.e.*, the articulated-body algorithm affords the direct calculation of the URV accelerations based on a known URVM state vector, and it accounts for the added mass effect of the manipulator on the URV.

Finally, a comparative study between the two modelling approaches has been

done, and the improvement in the reduction of the dynamic coupling effect by using the articulated-body equations within the controller has been shown.

7.3 Consideration of the Proposed Control Scheme on Real World Applications

The proposed control scheme has some limitations in its application to real world URVM tasks. These limitations arise because the model employed does not account for all the external inputs that have the potential to affect the dynamics of the system. In particular, contact of the URVM with the environment and the forces exerted by the payload on the end effector are not accounted for in the sliding mode controller. In the first case, if the system collides with an object at the subsea worksite, the proposed dynamics model fails to provide an accurate prediction of the resultant system motion, as the contact creates persisting forces and moments that are not modelled within the controller. Consequently, it may mislead the controller. Therefore, in order for the URVM system to effectively complete the pilot's desired task, the proposed model should be implemented in obstacle-free sites or for tasks that are completed at mid-depth.

During task execution, it is the end-effector that contacts the workpiece. The forces and moments caused by this contact can be incorporated into the articulated body algorithm. However because the system can experience three different contact scenarios depending on the extent of the payload, the incorporation of these forces and moments occurs in three different ways.

Negligible Payload

In this scenario, the contact that the end-effecter has with its workpiece has no significant effect on the dynamics of the system, because there is only negligible contact force applied to the end-effector. When a negligeable payload exists, such as in welding or pick-and-place tasks, the end-effector does not experience excessive force. In these cases, the articulated body based controller can give acceptable predictions of the system dynamics without the necessity of additional contact force measurements being added to the controller.

Medium Payload

In this scenario, the contact that the end-effector has with its workpiece impacts and changes the system dynamics. Therefore, to be able to accurately predict the system dynamics, these forces and moments must be inserted into the system equations within the controller. This requires using force-torque sensors to measure the force and moments on the end-effector. These sensors will feedback into the system equation in the controller, and will allow the controller to maintain acceptable performance.

Large Payload

In this scenario, the contact that the end-effector has with the workpiece dominates the URVM dynamics. In these cases, motions of the master arm by the human pilot may not be reproducible by the slave arm of the URVM. As such, the path followed by the end-effector will deviate from the pilot's intent regardless of the performance of the controller proposed in this work - even with the addition of measured contact forces and moments. One way to solve this problem is to limit the inputs that the

human pilot makes to the end-effector motion. This would have to involve some on-line was of discerning if the intended motion was breaching the capability of the URVM. An indicator that could be used to do this is the measure of s in each of the single-input single output sliding mode controllers. The human pilot can monitor changes in s to discern if the actual system motion is consistent with the desired motion. If an s value shows constant growth, the pilot must understand that the system can not follow the current commands and consequently the pilot must slow down the commanded end-effector motion.

7.4 Recommendations for Future Work

7.4.1 Overview

This work has addressed the problem of the reduction of dynamic coupling present in URVM system and has presented a control strategy to reduce the effect of the dynamic coupling on the attitude of the URV. Problems that have not been addressed in this work and are recommended for future research include:

1. Exploitation of Kinematic Redundancy in URVM Systems,
2. Centralized Multi-Input Multi-Output Controller Design.

7.4.2 Exploitation of Kinematic Redundancy in URVM Systems

URVM systems are kinematically redundant, as they possesses more degrees of freedom than those required for the end-effector to execute a task. This is due to the

fact that the URV part contributes six degrees of freedom in addition to the number of degrees of freedom provided by the manipulator joints. When a task is performed with URVM, the current strategy is to keep the URV stationary. A contribution of this thesis is the development of an automatic controller that keeps the URV stationary. However, when the end-effector performs at the boundary of its workspace or close to a kinematic singularity, reconfiguration of the whole system is needed. In this situation, the URVM redundancy can be exploited to achieve more dexterous of the URVM end-effector. This would allow URVM systems to be operated more efficiently in work sites that have many obstacles, with the redundant system able to attain any given pose in an optimal manner.

A kinematically redundant system contains an infinite number of joint-space configurations for specific task-space coordinates and thus requires redundancy resolution schemes to determine which joint-space configuration is used. In other words, a redundancy resolution scheme generates the desired trajectories for both the URV and the manipulator in such a way that predefined constraints are met, including for example:

- Minimize the kinetic energy of the system;
- Minimize the joint torques required for the motion. By satisfying this criteria, saturation of the actuators can be avoided. Moreover, it leads the task to be performed with minimum effort and in turn reduces the energy consumption;
- Maximize the manipulability of the system, *i.e.,* give it the best ability to move and /or exert forces in all directions;
- Avoid obstacles in subsea worksites;

- Avoid joints reaching their mechanical limits;
- Avoid singularities. Singularity-free trajectories are very crucial in generating the desired trajectory values since at a singularity, an arbitrary motion of the end-effector is not possible when the URV is held fixed and the manipulator is not redundant. Furthermore, joint velocities can become large for small task velocities.

One major point that should be addressed is the fundamental difference between the redundant URVM system and a more typical land-based redundant system: a land-based device has very well known dynamic parameters. Therefore, a redundant land-based device can be controlled purely based on kinematic knowledge. However, for the undersea vehicle phenomenon such as added mass make it necessary to involve the URVM dynamics.

7.4.3 Centralized Multi-Input Multi-Output Controller Design

So far, a series of model-based sliding-mode controllers for each active degree of freedom of the URV has been employed to control the URV part of the system. Since every individual active degree of freedom is controlled indepedently from each other, this type of controllers is called the decentralized controller. A decentralized controller is composed of a series of single-input single-output controllers. Alternative approach to the decentralized controllers is the centralized controllers. In the centralized controllers, instead of a series of single-input single output controllers, there is only one multi-input multi-output controller. In this type of controller, the controller output is commanded by one centralized controller, taking into account of the

whole system dynamics. In Appendix F, an example of the centralized control design process is outlined based on the work of [45]. Note that in [45], not only is the URV's motion trajectory controlled, but also the manipulator joints are controlled as opposed to the thesis in which only the URV trajectory motion is controlled.

References

[1] D. M. Lane, M. W. Dunnigan, P. J. Knightbridge, and A. W. Quinn, "Planning and control for coordination of underwater manipulators," *Proc. IEE Control 91*, pp. 493–498, 1991.

[2] J. Yuh, "Modelling and control of underwater robotic vehicles," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, no. 4, pp. 1475–1483, 1990.

[3] J. H. Ryu, D. S. Kwon, and P. M. Lee, "Control of underwater manipulators mounted on an rov using base force information," *IEEE International Conference on Robotics and Autimation*, pp. 3238–3243, 2001.

[4] I. Schjolberg and T. I. Fossen, "Modelling and control of underwater vehicle-manipulator systems," in *3th Conference on Marine Craft Manoeuvring and Control*, pp. 45–47, 1994.

[5] T. J. Tarn, G. A. Shoults, and S. P. Yang, "A dynamic model for an underwater vehicle with a robotic manipulator using kane's method," *Autonomous Robots*, vol. 3, pp. 269–283, 1996.

[6] S. McMillan, D. E. Orin, and R. B. McGhee, "Efficient dynamic simulation of an underwater vehicle with a robotic manipulator," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, pp. 1194–1206, 1994.

[7] T. I. Fossen, *Guidence and Control of Ocean Vehicles.* John Wiley and Sons, 1994.

[8] J. N. Newman, *Marine Hydrodynamics.* MIT Press, 1977.

[9] O. M. Faltinsen, *Sea Loads on Ships and Offshore Structures.* Cambridge University Press, 1990.

[10] S. Sagatun and T. Fossen, "Lagrangian formulation of underwater vehicles' dynamics," in *IEEE International Conference on Systems, Man and Cybern.*, (Charlottesville, VA), pp. 1029–1034, 1991.

[11] M. Nahon, "A simplified dynamics model for autonomous underwater vehicles," in *IEEE Symposium on Autonomous Underwater Vehicle Technology*, pp. 373–379, 1996.

[12] X. Chen, D. Marco, S. Smith, E. An, K. Ganesan, and T. Healey, "6 dof nonlinear auv simulation toolbox," in *IEEE Ocean 97*, 1997.

[13] A. Serrani and S. Zanoli, "Designing guidance and control schemes for auvs using an integrated simulation environment," in *IEEE International Conference on Robotics and Automation*, pp. 1079–1083, 1998.

[14] D. R. Yoerger, J. G. Cooke, and J. J. E. Slotine, "The influence of thruster dynamics on underwater vehicle behaviour and their incorporation into control system design," *IEEE J. Oceanic Eng.*, vol. 15, pp. 167–177, 1990.

[15] W. P. A. Van Lammeren, J. D. Van Manen, and M. W. C. Oosterveld, "The wageningen b-secrew series," *Trans., ASME*, pp. 269–317, 1969.

[16] J. Healey, R. S. M., S. Cody, D. Miles, and J. P. Brown, "Toward an improved understanding of thruster dynamics for underwater vehicles," *IEEE J. Oceanic Eng.*, vol. 20, pp. 354–361, 1995.

[17] R. Bachmayer, L. L. Whitcomb, and M. Grosenbaugh, "An cccurate four-quadrant nonlinear dynamical model for marine thrusters: theory and experimental validation," *IEEE J. Oceanic Eng.*, vol. 1, no. 25, pp. 146–159, 2000.

[18] L. L. Whitcomb and D. R. Yoerger, "Preliminary experiments in model-based thruster control for underwater vehicle positioning," *IEEE J. Oceanic Eng.*, vol. 24, no. 4, pp. 495–506, 1999.

[19] J. Y. S. Luh, M. W. Walker, and R. P. Paul, "Resolved acceleration control for mechanical manipulators," *ASME J. Dyn., Sys., Meas. and Contr.*, vol. 102, pp. 69–76, 1980.

[20] M. E. Kahn and B. Roth, "The near minumum time control of open-loop articulated kinematic chains," *ASME J. Dyn., Sys., Meas. and Contr.*, vol. 93, pp. 164–172, 1971.

[21] J. M. Hollerbach, "A recursive lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity," *IEEE Trans. Syst., Man, Cybern.*, no. 11, pp. 730–736, 1980.

[22] Y. Stepanenko and M. Vukobratovic, "Dynamics of articulated open-chain active mechanisms," *Math. Biosciences*, vol. 28, pp. 137–170, 1976.

[23] D. E. Orin, R. B. McGhee, M. Vukobratovic, and G. Hartoch, "Kinematic and kinetic analysis of open-chain linkages utilizing newton-euler methods," *Math. Biosciences*, vol. 43, pp. 107–130, 1979.

[24] C. A. Balafoutis, R. V. Patel, and P. Misra, "Efficient modeling and computation of manipulator dynamics using orthogonal cartesian tensors," *IEEE Trans. of Robotics and Automation*, vol. 4, pp. 665–676, 1988.

[25] S. McMillan, *Computational dynamics for robotic systems on land and underwater.* Ph.D. Dissertation, The Ohio State University, Colombus, Summer 1994.

[26] M. W. Walker and D. E. Orin, "Efficient dynamic computer simulation of robotic mechanisms," *Trans. ASME J. Dyn., Sys., Meas. and Contr.*, vol. 104, pp. 205–211, 1982.

[27] R. Featherstone, "The calculation of robot dynamics using articulated-body inertias," *Int. J. Robotics Research*, vol. 2, no. 1, pp. 13–30, 1983.

[28] H. Brandl, R. Johanni, and M. Otter, "A very efficient algorithm for the simulation of robots and similar multibody systems without inversion of the mass matrix," in *in Proc. IFAC/IFIP/IMACS Int. Symp. Theory of Robots*, 1986.

[29] S. McMillan and D. E. Orin, "Efficient computation of articulated-body inertias using successive axial screws," *IEEE Trans. on Robotics and Automation*, vol. 11, pp. 606–615, August 1995.

[30] H. Kazerooni and T. B. Sheridan, "Computer simulation and control of underwater vehicles," tech. rep., MIT Sea grant college program Report No:MITSG-8219, October 1982.

[31] D. Yoerger and J. J. E. Slotine, "Robust trajectory control of underwater vehicles," *IEEE J. Oceanic Eng.*, pp. 462–470, 1985.

[32] J. J. E. Slotine, *Tracking control of nonlinear systems using sliding surfaces.* Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, may 1983.

[33] D. R. Yoerger and J. B. Newman, "Demonstration of supervisory control for rov's and manipulators," in *ROV'85*, pp. 82–87, 1985.

[34] R. Cristi, F. Papoulias, and A. Healey, "Adaptive sliding- mode control of autonomous underwater vehicles in the dive plane," *IEEE J. Oceanic Eng.*, vol. 15, no. 3, pp. 152–160, 1990.

[35] K. Goheen and E. R. Jeffery, "Multivariable self-tuning autopilots for autonomous and remotely operated underwater vehicle," *IEEE J. Oceanic Eng.*, vol. 15, no. 3, pp. 144–151, 1990.

[36] J. Yuh, "A neural net controller for underwater robotic vehicles," *IEEE J. Oceanic Eng.*, vol. 15, no. 3, pp. 161–166, 1990.

[37] A. J. Healey and D. Lienard, "Multivariable sliding-mode control for autonomous diving and steering of unmanned underwater vehicles," *IEEE J. Oceanic Eng.*, vol. 18, no. 3, pp. 327–339, 1993.

[38] G. Conte and A. Serrani, "Robust control of a remotely operated underwater vehicle," *Automatica*, vol. 34, no. 2, pp. 193–198, 1998.

[39] H. Mahesh and J. Yuh, "A coordinated control of underwater vehicle and robotic manipulator," *Journal of Robotic System*, vol. 8, pp. 339–370, 1991.

[40] T. J. Tarn and S. P. Yang, "Modelling and control of underwater robotic manipulators," in *IEEE International Conference Robotics and Automation*, pp. 2166–2171, 1997.

[41] G. Antonelli and S. Chiaverini, "Adaptive tracking control of underwater vehicle-manipulator systems," in *IEEE International Conference on Control Applications*, pp. 1089–1093, 1998.

[42] C. Canucas de Wit, D. Olguin Diaz, and M. Perrier, "Nonlinear control of an underwater vehicle/manipulator system with composite dynamics," *IEEE Transactions on Control System Technology*, vol. 8, pp. 948–960, 2000.

[43] F. Antonelli, G. Caccavale, S. Chiaverini, and L. Villani, "Tracking control for underwater-vehicle manipulator systems with velocity estimation," *IEEE Journal of Oceanic Engineering*, vol. 25, pp. 399–413, 2000.

[44] M. W. Dunnigan and G. T. Russell, "Evaluation and reduction of the dynamic coupling between a manipulator and an underwater vehicle," *IEEE J. Oceanic Eng.*, vol. 23, no. 3, pp. 260–273, 1998.

[45] G. Antonelli, *Underwater Robots: Motion and Force Control of Vehicle-Manipulator Systems.* Springer-Verlag, 2003.

[46] SNAME, "Nomenclature for treating the moton of a submerged body through a fluid," tech. rep., The Society of Naval Architecs and Marine Engineers, 1950.

[47] J. H. Ginsberg, *Advanced Engineering Dynamics, Second Edition.* Cambridge University Press, 1998.

[48] J. J. Craig, *Introduction to Robotics, Mechanism and Control, Second Edition.* Addison-Wesley Publishing Company, 1987.

[49] R. Featherstone, *Robot Dynamics Algorithms.* Kluwer Academic Publishers, 1987.

[50] R. Featherstone and D. E. Orin, "Robot dynamics: equations and algorithms," in *IEEE Int. Conf. Robotics and Automation*, pp. 826–834, 2000.

[51] O. M. Griffin and S. E. Ramberg, "Some recent studies of vortex shedding with application to marine tubularsand risers," *Trans. ASME*, vol. 104, pp. 2–13, 1982.

[52] K. Ioi and K. Itoh, "Modeling and simulation of an underwater manipulator," *Advanced Robotics*, vol. 4, no. 4, pp. 303–317, 1990.

[53] T. Sarpkaya and M. Isaacson, *Mechanics of Wave Forces on Offshore Structures.* Van Nostrand Reinhold, 1981.

[54] M. H. Patel, *Dynamics of Offshore Structures.* Butterworths, 1989.

[55] F. M. White, *Fluid Mechanics.* McGraw Hill, 1999.

[56] B. Levesque and M. J. Richard, "Dynamic analysis of a manipulator in a fluid environment," *Int J. Robotics Res.,*, vol. 13, pp. 221–213, 1994.

[57] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control.* John Wiley and Sons, 1989.

[58] L. L. Whitcomb and D. R. Yoerger, "Development, comparison, and preliminary experimental validation of nonlinear dynamic thruster models," *IEEE J. Oceanic Eng.*, vol. 24, pp. 481–494, 1999.

[59] J. J. E. Slotine and L. Weiping, *Applied Nonlinear Control.* Prentice-Hall, 1991.

[60] R. Cristi, F. A. Papoulias, and A. J. Healey, "Adaptive sliding mode control of autonomous underwater vehicles in the dive plane," *IEEE J. Oceanic Eng.*, vol. 15, no. 3, pp. 152–160, 1990.

[61] N. Sarkar, T. K. Podder, and G. Antonelli, "Fault-accommodating thruster force allocation of an auv considering thruster redundancy and saturation," *IEEE Trans. on Robotics and Automation*, vol. 18, no. 2, pp. 223–233, 2002.

[62] H. J. Marquez, *Nonlinear Control Systems.* John Wiley and Sons, 2003.

Appendix A

Kinematic Equations

In this appendix, the equations governing the kinematics of serial-manipulators are given. Before this discussion, the kinematic equations will be derived for a rigid body and then the discussion will be extended to include the most general cases in robot kinematics. There are two basic theorems that are used frequently in the kinetics of rigid bodies. These two theorems provide a basis for the derivation of equations of motion of rigid bodies.

Theorem 1 *"Let* $\mathbf{a}$ *be a vector whose components relative to a moving reference frame are constant, and let* $\boldsymbol{\omega}$ *be the angular velocity of that reference frame. Then the rate of change of* $\mathbf{a}$ *is the cross product of* $\boldsymbol{\omega}$ *and* $\mathbf{a}$*." [47]*

Therefore, for the rate of change of vector $\mathbf{a}$ is given by:

$$\dot{\mathbf{a}} = \boldsymbol{\omega} \times \mathbf{a} \tag{A.1}$$

Theorem 1 is used to differentiate vector quantities with respect to a moving reference frame and it encompasses the effects the rotation of frame.

Theorem 2 *"Let xyz be a coordinate system that is undergoing a spatial rotation. Let* $\mathbf{e}_i$ *be a unit vector parallel to the* i^{th} *axis of rotation in the sense of the rotation according to the right-hand rule, and let* ω_i *be the corresponding rate of rotation in radians per second. Then the angular velocity of xyz is given by* $\boldsymbol{\omega} = \sum_i \omega_i \mathbf{e}_i$*."* *[47]*

The above theorem allows us to write the angular velocity of rigid bodies as the vector sum of each individual rotational effects.

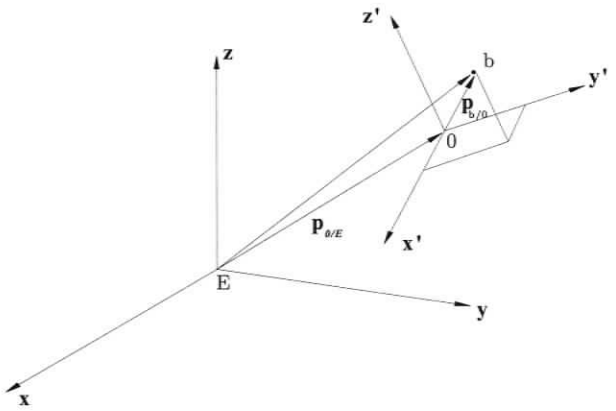


Figure A.1: Position Relative to a Fixed or Moving Reference Frame

The general situation is demonstrated in Figure A.1. According to Figure A.1, the reference frame $\{E\}$ represents the inertial frame. Point b can be either fixed or moving with respect to the moving reference frame $\{0\}$. If the position of point b changes in time, this situation corresponds to the prismatic joint case in which the joint displacement has translational velocity with respect to its own body-fixed frame. If point b is fixed to the moving reference frame, then the situation matches with the rotational joint case in which the joint displacement does not have velocity with respect to its own body-fixed frame.

Point Translates With Respect to the Reference Frame

The position vector can be constructed from the fixed and moving reference frame by the following equation:

$$\mathbf{p}_{b/E} = \mathbf{p}_{0/E} + \mathbf{p}_{b/0} \tag{A.2}$$

where $\mathbf{p}_{0/E}$ is the position vector from the origin of frame $\{E\}$ to the origin of frame $\{0\}$. $\mathbf{p}_{b/0}$ is the position vector from the origin of frame $\{0\}$ to point b.

The absolute velocity is defined as the time derivative of the position with respect to the fixed inertial reference frame. Therefore, the absolute velocity of point b is computed by differentiating Equation A.2 as follows:

$$\frac{d}{dt}\mathbf{p}_{b/E} = \frac{d}{dt}\mathbf{p}_{0/E} + \frac{d}{dt}\mathbf{p}_{b/0} \tag{A.3}$$

The term $\frac{d}{dt}\mathbf{p}_{0/E}$ corresponds to the absolute velocity of the origin of frame $\{0\}$. Given that $\mathbf{p}_{0/E}$ is defined as $\mathbf{p}_{0/E} = x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$ in terms of the inertial frame, its time derivative is calculated based on Theorem 1 and 2 as:

$$\frac{d}{dt}\mathbf{p}_{0/E} = \dot{x}\mathbf{i} + \dot{y}\mathbf{j} + \dot{z}\mathbf{k} + \boldsymbol{\omega}_E \times \mathbf{p}_{0/E} \tag{A.4}$$

The first three components of Equation A.4 are due to the changes in the position of vector $\mathbf{p}_{0/E}$ with respect to the inertial frame. These three components are defined as the absolute velocity of the origin of frame $\{0\}$. The last term accounts for changes in the rotation of the frame. Since the inertial frame has no rotational velocity, $\boldsymbol{\omega}_E$ is equal to zero. Therefore Equation A.4 simplifies to:

$$\frac{d}{dt}\mathbf{p}_{0/E} = \dot{x}\mathbf{i} + \dot{y}\mathbf{j} + \dot{z}\mathbf{k} = \mathbf{v}_0 \tag{A.5}$$

The time derivative of $\mathbf{p}_{b/0}$ contains the effects of the changing coordinate systems along with the rotation of frame $\{0\}$ itself. The overall effects can be found by

superimposing these two effects. Given that $\mathbf{p}_{b/0}$ is defined as $\mathbf{p}_{O/E} = x\mathbf{i}' + y\mathbf{j}' + z\mathbf{k}'$ in terms of frame $\{0\}$, its time derivative is calculated based on Theorem 1 and 2 as:

$$\frac{d}{dt}\mathbf{p}_{b/0} = \underbrace{\dot{x}\mathbf{i}' + \dot{y}\mathbf{j}' + \dot{z}\mathbf{k}'}_{(\mathbf{v}_b)_0} + \boldsymbol{\omega}_0 \times \mathbf{p}_{b/0} \tag{A.6}$$

It is known that frame $\{0\}$ is moving. This fact is accounted for by the term $\boldsymbol{\omega}_0$, which is the rotational velocity of frame $\{0\}$. The term $(\mathbf{v}_b)_0$ represents the velocity of point b observed by an observer sitting on the origin of the reference frame $\{0\}$. In other words, it corresponds to the relative velocity of point b with respect to frame $\{0\}$.

Finally, the absolute velocity of point b is found by substituting Equations A.5 and A.6 into Equation A.3 yielding:

$$\mathbf{v}_b = \mathbf{v}_0 + (\mathbf{v}_b)_0 + \boldsymbol{\omega}_0 \times \mathbf{p}_{b/0} \tag{A.7}$$

A relation for the absolute acceleration of point b can be found by differentiating Equation A.7 and is given by:

$$\frac{d}{dt}\mathbf{v}_b = \frac{d}{dt}\mathbf{v}_0 + \frac{d}{dt}(\dot{\mathbf{v}}_b)_0 + \frac{d}{dt}(\boldsymbol{\omega}_0 \times \mathbf{p}_{b/0}) = \mathbf{a}_b \tag{A.8}$$

The time derivative of the first term gives the acceleration of the origin of frame $\{0\}$ and is given, in view of Equation A.5, by:

$$\frac{d}{dt}\mathbf{v}_0 = \ddot{x}\mathbf{i} + \ddot{y}\mathbf{j} + \ddot{z}\mathbf{k} = \mathbf{a}_0 \tag{A.9}$$

The relative acceleration of point b with respect to the origin of frame $\{0\}$ is the time derivative of the second term in Equation A.7:

$$\frac{d}{dt}(\mathbf{v}_b)_0 = \underbrace{\ddot{x}\mathbf{i}' + \ddot{y}\mathbf{j}' + \ddot{z}\mathbf{k}'}_{(\dot{\mathbf{v}}_b)_0} + \boldsymbol{\omega}_0 \times (\mathbf{v}_b)_0 \tag{A.10}$$

Lastly, using Equation A.6, the time derivative of the third term in Equation A.7 is given by:

$$\begin{aligned} \frac{d}{dt}(\boldsymbol{\omega}_0 \times \mathbf{p}_{b/0}) &= \dot{\boldsymbol{\omega}}_0 \times \mathbf{p}_{b/0} + \boldsymbol{\omega}_0 \times \dot{\mathbf{p}}_{b/0} \\ &= \boldsymbol{\alpha}_0 \times \mathbf{p}_{b/0} + \boldsymbol{\omega}_0 \times \left((\mathbf{v}_b)_0 + \boldsymbol{\omega}_0 \times \mathbf{p}_{b/0}\right) \end{aligned} \tag{A.11}$$

Finally, substituting Equations A.9, A.10 and A.11 into Equation A.8 yields:

$$\frac{d}{dt}\mathbf{v}_b = \mathbf{a}_0 + (\dot{\mathbf{v}}_b)_0 + \boldsymbol{\omega}_0 \times (\mathbf{v}_b)_0 + \boldsymbol{\alpha}_0 \times \mathbf{p}_{b/0} + \boldsymbol{\omega}_0 \times \left((\mathbf{v}_b)_0 + \boldsymbol{\omega}_0 \times \mathbf{p}_{b/0}\right) = \mathbf{a}_b \tag{A.12}$$

Rearranging Equation A.12 yields the following equation:

$$\mathbf{a}_b = \mathbf{a}_0 + (\dot{\mathbf{v}}_b)_0 + \boldsymbol{\alpha}_0 \times \mathbf{p}_{b/0} + \boldsymbol{\omega}_0 \times \boldsymbol{\omega}_0 \times \mathbf{p}_{b/0} + 2\boldsymbol{\omega}_0 \times (\mathbf{v}_b)_0 \tag{A.13}$$

The last two terms in Equation A.13 are called the centripetal and the Coriolis acceleration, respectively. The Coriolis effect is produced by the rotation of frame $\{0\}$ along with the change in the components of the vector $\mathbf{p}_{b/0}$.

Point is Fixed with Respect to the Reference Frame $\{0\}$

In this case, as opposed to the first case, point b is fixed to the moving frame $\{0\}$. Since, point b moves with the frame $\{0\}$, the observer does not observe any changes in the position of point b. Rather, the velocity of point b with respect to the observer is zero. Therefore, $(\mathbf{v}_b)_0 = 0$ and Equation A.7 simplifies to:

$$\mathbf{v}_b = \mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{p}_{b/0} \tag{A.14}$$

A relation for the absolute acceleration of point b can be found by differentiating the time derivative of the absolute velocity of point b. Using Equation A.6 in view of $(\mathbf{v}_b)_0 = 0$, the absolute acceleration of point b is obtained as:

$$\begin{aligned} \mathbf{a}_b &= \mathbf{a}_0 + \boldsymbol{\alpha}_0 \times \mathbf{p}_{b/0} + \boldsymbol{\omega}_0 \times \dot{\mathbf{p}}_{b/0} \\ &= \mathbf{a}_0 + \boldsymbol{\alpha}_0 \times \mathbf{p}_{b/0} + \boldsymbol{\omega}_0 \times \left(\boldsymbol{\omega}_0 \times \mathbf{p}_{b/0}\right) \end{aligned} \tag{A.15}$$

The derivations of angular velocities and accelerations are explained in Section 2.3.4.

Appendix B

Cross Products of Vectors Using Skew- Symmetric Form

Let a be a vector whose components are $a = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}^T$ and b be a vector whose components are $b = \begin{bmatrix} b_x & b_y & b_z \end{bmatrix}^T$. The cross products of two vectors are described by:

$$a \times b = \begin{bmatrix} a_y\, b_z - a_z\, b_y \\ a_z\, b_x - a_x\, b_z \\ a_x\, b_y - a_y\, b_x \end{bmatrix} \tag{B.1}$$

Equation B.1 can be written in the following form:

$$a \times b = \tilde{a} b \tag{B.2}$$

where $\tilde{a}$ is the 3×3 skew-symmetric matrix:

$$\tilde{a} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \tag{B.3}$$

Appendix C

Derivative of a Rotation Matrix

Let $\{A\}$ be a fixed coordinate frame and $\{B\}$ be a coordinate frame rotation with the angular velocity ω relative to frame $\{A\}$. Let ω_B is the representation of ω in $\{B\}$. Let r be a vector that is fixed with respect to frame $\{B\}$, *i.e.*, vector r rotates with frame $\{B\}$. Let $\mathbf{R}$ be the rotation matrix that transforms vector quantities in $\{B\}$ to $\{A\}$. If r_A and r_B are the representation of vector r in terms of frame $\{A\}$ and frame $\{B\}$, respectively. The time derivative of r_A can be written as follows:

$$\dot{r}_A = \frac{d}{dt}(\mathbf{R}\, r_B) = \dot{\mathbf{R}}\, r_B + \mathbf{R}\, \dot{r}_B = \dot{\mathbf{R}}\, r_B \tag{C.1}$$

The time derivative of r can also be written as follows:

$$\dot{r}_A = \mathbf{R}\ (\omega_B \times r_B) = \mathbf{R}\, \tilde{\omega}_B\, r_B \tag{C.2}$$

Equalizing Equations C.1 and C.2 yields an expression for the time derivative of rotation matrices:

$$\dot{\mathbf{R}}\, r_B = \mathbf{R}\, \tilde{\omega}_B\, r_B \tag{C.3}$$

$$\dot{\mathbf{R}} = \mathbf{R}\, \tilde{\omega}_B \tag{C.4}$$

Appendix D

Derivation of Added Mass Coefficients for the URV

D.1 Added Mass Matrix

The added mass matrix is defined as:

$$\underline{\mathbf{I}}^A = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} = \begin{bmatrix} k_{\dot{p}} & k_{\dot{q}} & k_{\dot{r}} & k_{\dot{u}} & k_{\dot{v}} & k_{\dot{w}} \\ m_{\dot{p}} & m_{\dot{q}} & m_{\dot{r}} & m_{\dot{u}} & m_{\dot{v}} & m_{\dot{w}} \\ n_{\dot{p}} & n_{\dot{q}} & n_{\dot{r}} & n_{\dot{u}} & n_{\dot{v}} & n_{\dot{w}} \\ x_{\dot{p}} & x_{\dot{q}} & x_{\dot{r}} & x_{\dot{u}} & x_{\dot{v}} & x_{\dot{w}} \\ y_{\dot{p}} & y_{\dot{q}} & y_{\dot{r}} & y_{\dot{u}} & y_{\dot{v}} & y_{\dot{w}} \\ z_{\dot{p}} & z_{\dot{q}} & z_{\dot{r}} & z_{\dot{u}} & z_{\dot{v}} & z_{\dot{w}} \end{bmatrix} \tag{D.1}$$

Note that added mass matrix $\underline{\mathbf{I}}^A$ is the swapped version of that of [7]. This adjustment is needed to simplify the added mass force equation into the recursive articulated-body equations in which the angular components of the spatial vectors are placed above the translational components as opposed to [7]. In the added mass matrix,

each component of the added mass matrix is a proportional constant that defines an added mass force contribution due to an acceleration in one of the body's degree of freedom. For instance, added mass force contribution the hydrodynamic force f_A along y_0 caused by the linear acceleration $\dot{w}$ in the direction of z_0 is defined as:

$$f^A = -y_{\dot{w}}\ \dot{w} \text{ where } y_{\dot{w}} = \frac{\partial y_A}{\partial \dot{w}} \tag{D.2}$$

Note that notation similar to the SNAME [46] is used in the above matrix.

D.2 Derivation of Added Mass Coefficients for Translational Motion

The derivation of the added mass coefficients are made based on the strip theory [7]. In this theory, a rigid-body is divided into a number of small strips. Depending on the geometric shape of each strip, two-dimensional added mass coefficients are found for each strip. In order to obtain three-dimensional added mass coefficients, the integration is carried out over the length of the rigid body.

The URV is approximated as a rectangular box and is divided into a series of rectangular strips as illustrated in Figure D.1. The origin of the URV's frame is attached to the centre of mass of the URV. Two-dimensional added mass coefficient for the rectangular shape is found based on the dimensions of the strip as well as the density of water as follows:

$$\rho = 998\ \frac{kg}{m^3}, \quad W = 0.444\ m, \quad H = 0.154\ m \quad L = 0.596\ m \tag{D.3}$$

$$\frac{L}{H} = 3.87 \tag{D.4}$$

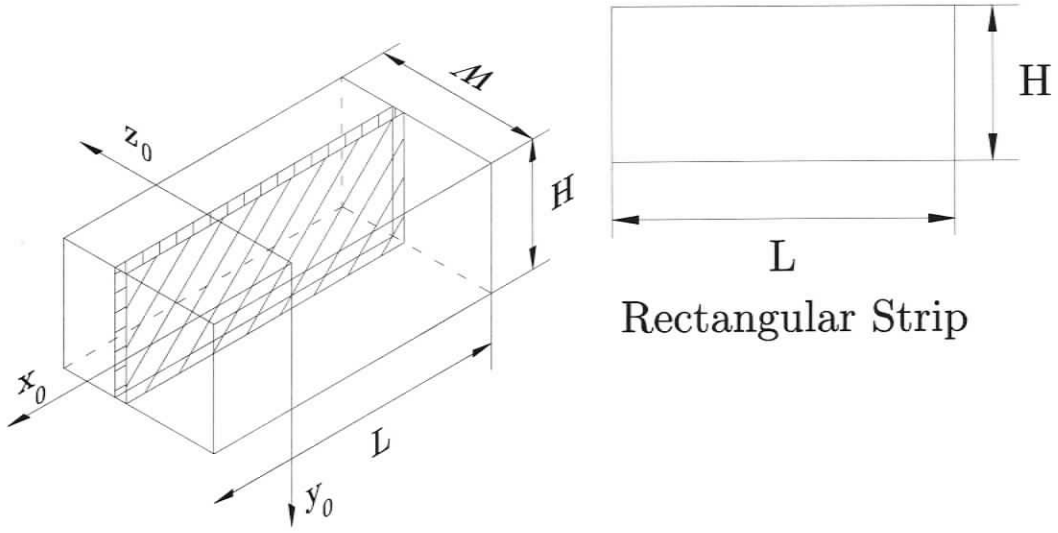


Figure D.1: URV is Considered To Be Composed of a Series of Rectangular Strips

Using Figure D.2, with $a = \frac{H}{2}$,

$$\frac{A_{11}^{(2D)}}{\rho\ \pi\ \left[\frac{H}{2}\right]^2} = 2 \Longrightarrow A_{11}^{(2D)} = 2\rho\ \pi\ \left[\frac{H}{2}\right]^2 = 37.179$$

Two-dimensional added mass coefficient $A_{11}^{(2D)}$ is found to be 37.179. The added mass coefficient for the whole URV body along the x (surge) axis can be given as [7]:

$$x_{\dot{u}} = -\int_{-W/2}^{W/2} A_{11}^{(2D)} dW \tag{D.5}$$

Substituting the known values into Equation D.5, yields:

$$x_{\dot{u}} = -\int_{-0.222}^{0.222} 37.179 dW = -16.507\ kg \tag{D.6}$$

The same process is followed for the added mass coefficients along $y_{\dot{v}}$ and $z_{\dot{w}}$.

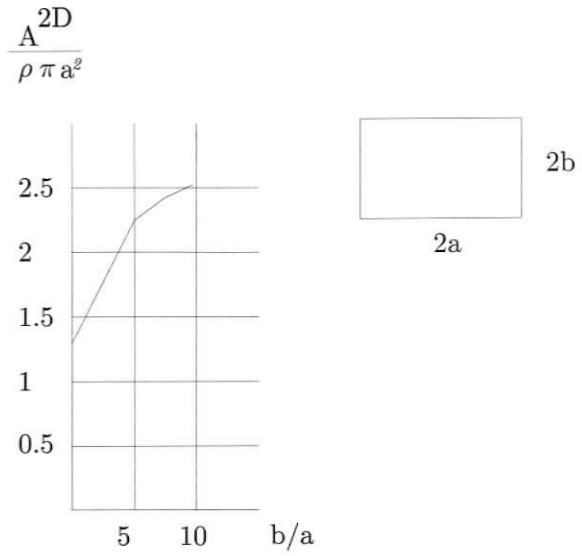


Figure D.2: 2D Added Mass Coefficients for a Rectangular Cross-Section

D.3 Derivation of Added Mass Coefficients for Rotational Motion

The derivation of the rotational added mass coefficients can be made based on the translational added mass coefficients. For instance, the added mass coefficients for the rotational motion around the roll axis of the URV is given in [7] as:

$$k_{\dot{p}} = \int_{-L/2}^{L/2} A_{44}^{(2D)} dL = -\int_{-W/2}^{W/2} W^2 A_{33}^{(2D)} dW + \int_{-H/2}^{H/2} H^2 A_{22}^{(2D)} dH \qquad \text{(D.7)}$$

Similar expressions for the derivation of the rotational added mass coefficients can be found in [7].

Appendix E

Simulation Parameters

E.1 URV Simulation Parameters:

$L = 0.596m,\ W = 0.444m,\ H = 0.154m$

$${}^0p_{0\to1} = \begin{bmatrix} 0.298 & 0 & 0 \end{bmatrix}^T m$$

$${}^0b_0 = \begin{bmatrix} 0 & 0 & -0.0254 \end{bmatrix}^T m$$

$$r_{G_0} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T m$$

$m_0 = 32kg, \quad m_0^f = 32kg$

$${}^0\mathbf{I}_0 = diag\left\{ \begin{array}{ccc} 0.498 & 0.878 & 1.254 \end{array} \right\} Nm\sec^2$$

$${}^0\underline{\mathbf{I}}_0^A = diag\left\{ \begin{array}{cccccc} 2.654 & 3.438 & 0.249 & 16.54 & 18 & 115.8 \end{array} \right\} Nm\sec^2$$

$A_{x_0} = 0.061m^2, \quad A_{y_0} = 0.85m^2$

$C_{0_x}^D = 1.07, \quad C_{0_y}^D = 1.07$ For Linear Drag Force

Rotation around Z_0 : $C_{0_{X-Z}}^D = 1.555,\ C_{0_{Y-Z}}^D = 2.646$

Rotation around X_0 : $C_{0_{X-Y}}^D = 2.093,\ C_{0_{Y-Z}}^D = 2.646$

Rotation around Y_0 : $C_{0_{X-Y}}^D = 2.205,\ C_{0_{Y-Z}}^D = 2.046$

E.2 Manipulator Simulation Parameters

$$^1p_{1\to2} = {}^2p_{2\to3} = {}^3p_{3\to e.e} = \begin{bmatrix} 0.298 & 0 & 0 \end{bmatrix}^T m$$

$$^1b_1 = {}^2b_2 = {}^3b_3 = \begin{bmatrix} 0.11 & 0 & 0 \end{bmatrix}^T m$$

$$^1r_{G_1} = {}^2r_{G_2} = {}^3r_{G_3} = \begin{bmatrix} 0.11 & 0 & 0 \end{bmatrix}^T m$$

$$m_{1,2,3} = 1,\ m^f_{1,2,3} = 1kg$$

$$^1\mathbf{I}_1 = {}^2\mathbf{I}_2 = {}^3\mathbf{I}_3 = diag\left\{ \begin{array}{ccc} 0 & 0.0137 & 0.0137 \end{array} \right\} Nm\sec^2$$

$$^1\underline{\mathbf{I}}^A_1 = {}^2\underline{\mathbf{I}}^A_2 = {}^3\underline{\mathbf{I}}^A_3 = diag\left\{ \begin{array}{cccccc} 0 & 0.0017 & 0.0017 & 0 & 0.1078 & 0.1078 \end{array} \right\} Nm\sec^2$$

$$C^D_{1,2,3} = 1.1$$

E.3 Thruster Simulation Parameters

$r = 0.0381 ohms$	$k_h = 17790$
$k_b = 0.0619 V\sec/rad$	$c_{L_{\max}} = 1.75$
$j_m = 5.6211e-005 kgm^2$	$c_{D_{\max}} = 1.2$
$b_m = 0 Nm\sec/rad$	$n_{gr} = 2$
$k_1 = 70.15$	$p = \pi/6 rad$
$k_2 = 1133.2$	$a = 0.00445 m^2$
$k_3 = 0.954$	$l_\gamma = 0.5$
$k_4 = 0.910$	$\Delta\beta = 0.2$

E.4 Sliding Mode Control Parameters

$b_{r_{\min}} = 1.2870,\ b_{r_{\max}} = 1.5889,\ b_{u,v_{\min}} = 36,\ b_{u,v_{\max}} = 44.44$

$b_{r_{\min}} = 0.5828,\ b_{r_{\max}} = 0.8392, b_{u,v_{\min}} = 0.0208\ ,\ b_{u,v_{\max}} = 0.03$ for the comparative study.

$\lambda = 3.14 rad/\sec$

$\eta = 0.5$

$\Phi_{r,u,v} = 0.02$, and $\Phi_{r,u,v} = 0.015$ for the comparative study.

$F_{r,u,v} = 1$,and $F_{r,u,v} = \left|\hat{f}_{r,u,v} - (1.6) \times \hat{f}_{r,u,v}\right|$ for the comparative study.

E.5 Thruster Control Parameters

$k_{fb} = -1$

E.6 Environmental Parameters

$${}^{E}a^{g} = \begin{bmatrix} 0 & 0 & 9.81 \end{bmatrix}^{T} m/\sec^2$$

$${}^{E}a^{f} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^{T} m/\sec^2$$

$\rho = 998 kg/m^3$

Appendix F

An Example of Centralized Model-Based Sliding-Mode Controller Design

Closed-Form Model of URVM Dynamic Equation

The equations of motions of an URVM system can be written in a closed-form as follows:

$$\mathbf{M}\,\dot{\boldsymbol{\zeta}} + \mathbf{C}\,\boldsymbol{\zeta} + \mathbf{D}\,\boldsymbol{\zeta} + \mathbf{g} = \mathbf{B}\mathbf{u} \tag{F.1}$$

where $\boldsymbol{\zeta}$ is a $(6+n)$ state vector defined as $\boldsymbol{\zeta} = \begin{bmatrix} \mathbf{v}_0^T & \boldsymbol{\omega}_0^T & \dot{\mathbf{q}}^T \end{bmatrix}^T$ and $\mathbf{M}$ is the $(6+n)\times(6+n)$ inertia matrix including added mass terms multiplying the acceleration terms, $\mathbf{C}$ is the $(6+n)$ vector of Coriolis and centripetal terms, $\mathbf{D}$ is the $(6+n)$ vector of dissipative effects, $\mathbf{g}$ is the $(6+n)$ vector of gravity and buoyancy effects, $\mathbf{B}$ is a $(6+n)\times h$ matrix containing the thruster mapping matrix, and lastly $\mathbf{u}$ is the $(h\times 1)$ vector of control inputs.

The following properties are assumed to hold:

- The inertia matrix is symmetric and positive definitive, *i.e.*, $\mathbf{M} = \mathbf{M}^T > 0$;
- the damping matrix is positive definitive, *i.e.*, $\mathbf{D} > 0$;
- the matrix $\mathbf{C}$ is skew-symmetric, *i.e.*, $\mathbf{C} = -\mathbf{C}^T$;
- for a suitable choice of the parametrization of $\mathbf{C}, \dot{\mathbf{M}} - 2\mathbf{C}$ is skew symmetric.

Centralized Model-Based Sliding Mode Controller Design

A multi-input multi-output model-based sliding mode controller allows using one centralized controller instead of many, *i.e.*, one for each controlled axis. The centralized controller coordinates the URV and the manipulator motion in such a way that the desired task is accomplished. As mentioned before, desired values for a specific task can be generated by the redundancy resolution scheme.

The vector of sliding surfaces $\mathbf{s}$, is defined as:

$$\mathbf{s} = \Upsilon \begin{bmatrix} {}^{0}_{\mathbf{E}}\mathbf{R}\ \check{\boldsymbol{\eta}}_1 \\ \check{\boldsymbol{\varepsilon}} \\ \check{\mathbf{q}} \end{bmatrix} - \begin{bmatrix} \mathbf{v}_0^T \\ \boldsymbol{\omega}_0^T \\ \dot{\mathbf{q}}^T \end{bmatrix} = \mathbf{r} - \boldsymbol{\zeta} \tag{F.2}$$

where Υ is a positive definite matrix containing the control bandwidth value for each sliding surface, $\check{\boldsymbol{\varepsilon}}$ is the quaternion representation of the URV[45] and the rest of the terms in Equation F.2 are defined as:

$$\check{\boldsymbol{\eta}}_1 = \begin{bmatrix} x_d - x & y_d - y & z_d - z \end{bmatrix}^T \tag{F.3}$$

$$\check{\mathbf{q}} = \begin{bmatrix} q_{d1} - q_1 & \cdots & q_{dn} - q_n \end{bmatrix}^T \tag{F.4}$$

Equation F.2 can be written as:

$$\mathbf{s} = \mathbf{r} - \boldsymbol{\zeta} \tag{F.5}$$

Differentiating Equation F.2 with respect to time yields:

$$\dot{\mathbf{s}} = \dot{\mathbf{r}} - \dot{\boldsymbol{\zeta}} \tag{F.6}$$

Equations F.5 and F.6 can be written as:

$$\boldsymbol{\zeta} = \mathbf{r} - \mathbf{s}, \quad \textit{and} \quad \dot{\boldsymbol{\zeta}} = \dot{\mathbf{r}} - \dot{\mathbf{s}} \tag{F.7}$$

Substituting Equation F.7 into Equation F.1 yields:

$$\mathbf{M}(\mathbf{q})\,\dot{\mathbf{s}} = \mathbf{M}\dot{\mathbf{r}} + \mathbf{C}(\mathbf{r} - \mathbf{s}) + \mathbf{D}(\mathbf{r} - \mathbf{s}) + \mathbf{g} - \mathbf{B}\mathbf{u} \tag{F.8}$$

Control Law Design based on Lyapunov Stability Analysis Control law design can be made based on a Lyapunov function. The idea is to formulate a scalar positive function of the system states, and choose a control law to make this function decrease [59]. The following positive definite Lyapunov function is taken, with $\mathbf{M}$ being positive definitive in mind, as:

$$V = \frac{1}{2}\mathbf{s}^T\mathbf{M}\mathbf{s} \tag{F.9}$$

Differentiating Equation F.9 with respect to time yields:

$$\dot{V} = \frac{1}{2}\left(\mathbf{s}^T\dot{\mathbf{M}}\mathbf{s} + \mathbf{s}^T\mathbf{M}\dot{\mathbf{s}} + \dot{\mathbf{s}}^T\mathbf{M}\mathbf{s}\right) \tag{F.10}$$

since $\dot{\mathbf{s}}^T\mathbf{M}\mathbf{s} = \mathbf{s}^T\mathbf{M}\dot{\mathbf{s}}$;

$$\dot{V} = \frac{1}{2}\mathbf{s}^T\dot{\mathbf{M}}\mathbf{s} + \mathbf{s}^T\mathbf{M}\dot{\mathbf{s}} \tag{F.11}$$

Substituting Equation F.8 into Equation F.11 and through the following steps the expression for $\dot{V}$ is obtained as follows:

$$\dot{V} = \frac{1}{2}\mathbf{s}^T\dot{\mathbf{M}}\mathbf{s} + \mathbf{s}^T\left(\mathbf{M}\dot{\mathbf{r}} + \mathbf{C}(\mathbf{r}-\mathbf{s}) + \mathbf{D}(\mathbf{r}-\mathbf{s}) + \mathbf{g} - \mathbf{B}\mathbf{u}\right) \tag{F.12}$$

$$\dot{V} = \frac{1}{2}\mathbf{s}^T\dot{\mathbf{M}}\mathbf{s} + \mathbf{s}^T\left(\mathbf{M}\dot{\mathbf{r}} + (\mathbf{C}+\mathbf{D})\mathbf{r} + \mathbf{g} - \mathbf{B}\mathbf{u}\right) - \mathbf{s}^T(\mathbf{C}+\mathbf{D})\,\mathbf{s} \tag{F.13}$$

$$\dot{V} = \frac{1}{2}\mathbf{s}^T\left(\dot{\mathbf{M}} - 2\mathbf{C}\right)\mathbf{s} + \mathbf{s}^T\left(\mathbf{M}\dot{\mathbf{r}} + (\mathbf{C}+\mathbf{D})\mathbf{r} + \mathbf{g} - \mathbf{B}\mathbf{u}\right) - \mathbf{s}^T\mathbf{D}\mathbf{s} \tag{F.14}$$

Since $\dot{\mathbf{M}} - 2\mathbf{C}$ gives a skew-symmetric matrix, $\mathbf{s}^T\left(\dot{\mathbf{M}} - 2\mathbf{C}\right)\mathbf{s}$ is equal to zero. Therefore, Equation F.14 simplifies to:

$$\dot{V} = \mathbf{s}^T\left(\mathbf{M}\dot{\mathbf{r}} + (\mathbf{C}+\mathbf{D})\mathbf{r} + \mathbf{g} - \mathbf{B}\mathbf{u}\right) - \mathbf{s}^T\mathbf{D}\mathbf{s} \tag{F.15}$$

Theorem 3 *Let x be an equilibrium point* [1] *of time invariant function $\dot{\mathbf{x}} = f(\mathbf{x})$ and let V be a continuously differentiable scalar function such that*

- $V(0) = 0$,
- $V(\mathbf{x}) > 0$ *in* D-$\{0\}$,
- $\dot{V}(\mathbf{x}) < 0$ *in* D-$\{0\}$,

where D is represents an open subset of $\Re^n$, thus $x = 0$ is asymptotically stable.[62]

[1] "A point $x = x_e$ in the state space is said to be an equilibrium point of the time-invariant system $\dot{x} = f(x)$, if it has the property that whenever the state of the system starts at x_e, it remains at x_e for all future time." [62]

Above theorem states that asymptotic stability is obtained if the conditions of Theorem 3 are satisfied. Therefore, for global asymptotic stability, control input $\mathbf{u}$ must be chosen such that $\dot{V} < 0$. The control input ,$\mathbf{u}$, is chosen as:

$$\mathbf{u} = \mathbf{B}^{\dagger}[\mathbf{K}_D s + \hat{\mathbf{M}}\dot{\mathbf{r}} \ + (\hat{\mathbf{C}} + \hat{\mathbf{D}})\mathbf{r} + \check{\mathbf{g}} + \mathbf{K}_s sgn(\mathbf{s})] \tag{F.16}$$

where $\mathbf{B}^{\dagger}$ is the pseudoinverse of matrix $\mathbf{B}$, $\mathbf{K}_D$ is a positive definite matrix of gains, $\hat{\mathbf{M}}$, $\hat{\mathbf{C}}$, $\hat{\mathbf{D}}$ and $\check{\mathbf{g}}$ are the estimates of inertia matrix $\mathbf{M}$, the vector of Coriolis and centripetal acceleration $\mathbf{C}$, the vector of dissipative effects, and gravitational and buoyant force $\mathbf{g}$, respectively. The term $\mathbf{K}_s$ is a positive definite matrix, and $sgn(\mathbf{s})$ is the vector function whose i^{th} component is defined as:

$$sgn\left(\mathbf{s}\right)_i = 1, \quad \text{for } s_i > 0 \tag{F.17}$$

$$sgn\left(\mathbf{s}\right)_i = -1, \quad \text{for } s_i \leq 0 \tag{F.18}$$

Substituting Equation F.16 into Equation F.15 yields:

$$\dot{V} = \mathbf{s}^T \left(\check{\mathbf{M}}\, \dot{\mathbf{r}} \ + (\check{\mathbf{C}} + \check{\mathbf{D}})\mathbf{r} + \check{\mathbf{g}} - \mathbf{K}_s sgn(\mathbf{s})\right) - \mathbf{s}^T \left(\mathbf{D} + \mathbf{K}_D\right)\mathbf{s} \tag{F.19}$$

where $\check{\mathbf{M}}$, $\check{\mathbf{C}}$ and $\check{\mathbf{D}}$ are defined as:

$$\begin{aligned} \check{\mathbf{M}} &= \mathbf{M} - \hat{\mathbf{M}} \\ \check{\mathbf{C}} &= \mathbf{C} - \hat{\mathbf{C}} \\ \check{\mathbf{D}} &= \mathbf{D} - \hat{\mathbf{D}} \end{aligned} \tag{F.20}$$

Since $\mathbf{D}$ and $\mathbf{K}_D$ are positive definite matrices, the term $\mathbf{s}^T \left(\mathbf{D} + \mathbf{K}_D\right)\mathbf{s}$ is always positive.

Theorem 4 *Consider a nonsingular symmetric matrix* $\mathbf{Q} \in \Re^{n \times n}$, *and let* $\lambda_{\min}$ *and* $\lambda_{\max}$ *be respectively the minimum and maximum eigenvalues of* $\mathbf{Q}$. *Under these conditions, for any* $\mathbf{x} \in \Re^{n \times n}$, *with* $||\mathbf{x}||$ *being 2-norm of the vector* $\mathbf{x}$, $\lambda_{\min}(\mathbf{Q})\,||\mathbf{x}||^2 \leq \mathbf{x}^T\mathbf{Q}\mathbf{x} \leq \lambda_{\max}(Q)\,||\mathbf{x}||^2$ *[62].*

In view of Theorem 4 and the positive definiteness of $\mathbf{K}_D$ and $\mathbf{D}$, Equation F.19 can be upper bounded as follows:

$$\dot{V} \leq -\lambda_{\min}\left(\mathbf{K}_D + \mathbf{D}\right)||\mathbf{s}||^2 - \lambda_{\min}\mathbf{K}_s\,||\mathbf{s}|| + \left|\left|\check{\mathbf{M}}\,\dot{\mathbf{r}} + (\check{\mathbf{C}} + \check{\mathbf{D}})\mathbf{r} + \check{\mathbf{g}}\right|\right|\,||\mathbf{s}|| \quad \text{(F.21)}$$

By choosing $\mathbf{K}_s$ such that:

$$\lambda_{\min}\mathbf{K}_s \geq \left|\left|\check{\mathbf{M}}\,\dot{\mathbf{r}} + (\check{\mathbf{C}} + \check{\mathbf{D}})\mathbf{r} + \check{\mathbf{g}}\right|\right| \quad \text{(F.22)}$$

the negative definiteness of the time derivative of V is guaranteed meaning s converges to zero asymptotically. It must be also demonstrated that once the error vectors are on the sliding surface, they converge to zero. The proof can be found in [45].