



Modelling and Feed-Forward Control of Robot Arms with Flexible Joints and Flexible Links

Master's thesis in Systems, Control and Mechatronics

Research conducted and thesis written by:

Mazin Hamad (890403-6137)

Main Supervisor:

Yiannis Karayiannidis, Chalmers University of Technology, Sweden

External Supervisor:

Torgny Brogårdh, Retired Corporate Executive Engineer, ABB AB, Sweden

MASTER'S THESIS 2016:EX047

**Modelling and Feed-Forward Control of Robot Arms
with Flexible Joints and Flexible Links**

MAZIN HAMAD



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Signals and Systems
Division of Automatic Control, Automation and Mechatronics
Mechatronics Research Group
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2016

Modelling and Feed-Forward Control of Robot Arms with Flexible Joints and Flexible Links

MAZIN HAMAD

© MAZIN HAMAD, 2016.

Supervisor: Yiannis Karayiannidis, Signals and Systems Department

External Supervisor: Torgny Brogårdh, ABB Group

Examiner: Yiannis Karayiannidis, Signals and Systems Department

Master's Thesis 2016:EX047

Department of Signals and Systems

Division of Automatic Control, Automation and Mechatronics

Mechatronics Research Group

Chalmers University of Technology

SE-412 96 Gothenburg

Telephone +46 31 772 1000

Cover: A teaser for the unsettlingly realistic Canadarm3 announced by Canadian Space Agency (Photo source: The Syrup Trap Magazine).

Typeset in L^AT_EX

Printed by [Reproservice (Chalmers printing services)]

Gothenburg, Sweden 2016

Modelling and Feed-Forward Control of Robot Arms with Flexible Joints and Flexible Links

MAZIN HAMAD

Department of Signals and Systems

Chalmers University of Technology

Abstract

Reducing the weight of robotic manipulators has been identified as one of the important factors that can reduce the production and installation cost and allow for safe physical human-robot interaction but it results to flexible robotic manipulators. Motion control of flexible manipulators is a challenging task, particularly when the flexibility is arising not only because of flexible joints but also due to flexible links. To address this problem, more advanced real-time control strategies that effectively use more accurate mathematical models are required.

This thesis deals with the various aspects of modelling, design, simulation and control of robotic manipulators that have both flexible joints and flexible links. For a manipulator with long-reach elastic arms (links), the simulations show that a lumped mass-spring-damper model with enough number of elements can adequately capture the critical modes of the distributed link flexibility of a robot arm. The main contributions of this work is the use of lumped inverse dynamic models to generate the required actuator references in addition to feedforward computed torque for high performance end-effector's trajectory tracking. To pursue this target, the dynamic differences between lumped models and elastic links are compensated both in the frequency- and time-domain. The compensation in frequency-domain relies on frequency response matching of zeros and poles between lumped models and models of the manipulator arms and joints. By using lumped model in the inverse dynamic models for feedforward control, real-time implementation will be possible in robot controllers used today. The compensation in the time-domain is based on feed-forward compensation of estimated link position errors obtained from measurements or FEM model simulations. Trajectory tracking results using different model implementations are presented and analysed and they show that the performance of the proposed modelling and control methodologies is promising.

Keywords: Flexible robotic manipulators, light-weight robotics, safe robotics, trajectory tracking, PID control, inverse dynamic models, feed-forward compensation.

Acknowledgements

A journey is easier when traveled together. Interdependence is certainly more valuable than independence. This thesis report is the result of the Master's work carried out during the final year of my scholarship period at Chalmers University of Technology, thanks to a Swedish Institute scholarship.

I am very grateful to my adviser and supervisor Torgny Brogårdh for suggesting this thesis work, at the first place, and for his considerable contribution to the (readability of the) contents of this thesis work. He was an encouraging companion in my research activities and has always provided me with lots of support, given me great enthusiasm and a lot of inspiration in many aspects.

It is my great pleasure to be especially thankful to my mentor Yiannis Karayianidis, who gave me the opportunity to do this research project and supervised it with very useful suggestions and motivating comments. I appreciate the latitude and confidence he gave me along the thesis journey, his knowledgeable advice, our positive interactions and his persistent commitment to high-quality research. All these together have paved the way for the excellent milestones reached.

Furthermore, I like to express and bestow many thanks due to Professor Thomas Abrahamsson for his advice and enriching couple of our weekly meetings by his fruitful comments and joyful scientific discussion.

I also express my gratitude to Giuseppe Giordano, the Ph.D. researcher at the Signals and Systems department at Chalmers, for his caring spirit, distinctive style and fierce academic mind he showed during our instructive discussions through my Master's studies and project, in addition to all the assistance and mental support he provided me with.

The contact with my student colleagues and with the researchers at Chalmers has always been enjoyable. My awesome Sudanese brothers, Alaaddin and Ahmed, my beautiful ambitious sister, Amal, and all my study colleagues, especially my dear friend Vishnu.

I am forever thankful to the kind soul of my great father, Dr. Ahmed Elhag, who encouraged and supported me by all possible and even impossible means in his own way. Special thanks to my family, my brothers and sisters, for their good-natured forbearance with the process and for their pride in this accomplishment.

No amount of words can adequately express the debt I owe to my wonderful mother, the noble Professor Gawahir Mohammed, for her love and affection and especially her courage that pushes me always forward following my dreams.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of the project. It was a team effort and it is dedicated to all those who believe in the richness of learning.

Mazin Hamad, Gothenburg, June 2016

Contents

List of Figures	xiii
List of Tables	xx
1 Introduction	1
1.1 Introduction	1
1.2 Motivation and Problem Justification	2
1.3 Literature Review	3
1.3.1 Earlier Studies in the Field of Flexible Robotic Manipulators .	3
1.3.2 Recent and Relevant Work on Flexible Manipulators	4
1.4 Objectives	7
1.5 Thesis Layout	8
2 Modelling of Flexible Robotic Manipulators	10
2.1 Sources of Flexibility	10
2.1.1 Joint Flexibility	10
2.1.2 Link Flexibility	12
2.1.3 Manipulators with both Joint and Link Flexibilities	13
2.2 Dynamical Models for the Study of Flexible Robotic Arms	14
2.2.1 2-Mass System Dynamic Model of a Motor with Flexible Gear- box Drive Train	14
2.2.2 3-Mass System Dynamic Model of a Motor with Flexible Gear- box Drive Train and a Rigid Link	15
2.2.3 Multi-Mass System Dynamic Model of a Motor with Flexi- ble Gearbox Drive Train and Lumped Mass-Spring-Damper Model of a Flexible Link	19
2.2.4 ADAMS Models of Gearbox Drive Train with FEM Model of a Flexible Link	24
2.3 Modes and Anti-Modes Matching in the Frequency Plane	27
2.3.1 Frequency Response Matching of a 2-Mass Plant System Using an Identical 2-Mass Inverse Dynamics Model	27
2.3.2 Frequency Response Matching of a 3-Mass Plant System . . .	28
2.3.2.1 Using a 2-Mass Inverse Dynamics Model	28
2.3.2.2 Using an Identical 3-Mass Inverse Dynamics Model .	30
2.3.3 Frequency Response Matching of Multi-Mass Plant Systems .	30
2.3.3.1 A 2-Mass Plant System Case	30

2.3.3.2	A 3-Mass Plant System Case	32
2.3.4	Frequency Response Matching of ADAMS Plant Systems	34
2.3.4.1	A 2-Mass Plant System Case	34
2.3.4.2	A 3-Mass Plant System Case	34
3	Problem Statement, Control System Design and Compensation Techniques	36
3.1	Problem Statement	36
3.2	Reference Trajectory Planning for Feedforward Motion Control	37
3.2.1	Higher-order trajectory planning:	38
3.2.1.1	Fourth-order trajectory planning	39
3.3	Servo PID Controller	44
3.4	Inverse Dynamics (Feed-forward) Computed Torque Controller	45
3.5	Smoothing Filters for Reference Profile Signals and Inverse Dynamics Derivations	48
3.6	Compensation of Differences between ADAMS FEM/Lumped-Masses Model and Flexible Link Model	49
4	Simulation Results and Discussion	54
4.1	Simulation Parameters and Scenario	54
4.2	Simulation Results with Comments	55
5	Conclusions and Future Work	59
5.1	Future Work	60
	Bibliography	63
A	ADAMS-MATLAB/Simulink Interface and Performing Co-Simulations	I
B	Frequency Responses of Additional 2-Mass Models Used in Simulations	VI
C	Frequency Responses of Additional 3-Mass Models Used in Simulations	VIII
D	Frequency Responses of Additional Multi-Mass (3-Mass) Models Used in Simulations	X
E	Steps for Mode/Anti-Mode Matching in the Frequency Plane Using Inverse Dynamic Models	XIII
E.1	Matching procedure of a 2-mass (3-mass) plant mode(s) and anti-mode(s) using a 2-mass inverse dynamic model	XIII
E.2	Matching procedure of a 3-mass plant modes and anti-modes using a 3-mass inverse dynamic model	XIV
F	Detailed Results and Comparisons	XV
F.1	A 2-Mass Plant System	XV
F.1.1	With A Simple Servo PID Controller Results	XV

F.1.2	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XVI
F.2	A 3-Mass Plant System	XVIII
F.2.1	With A Simple Servo PID Controller Results	XVIII
F.2.2	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XVIII
F.2.3	With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XXI
F.3	Multi-Mass Plant Systems	XXIII
F.3.1	A 2-Mass Plant System Case	XXIII
F.3.1.1	With A Simple Servo PID Controller Results	XXIII
F.3.1.2	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XXIII
F.3.2	A 3-Mass Plant System Case	XXV
F.3.2.1	With A Simple Servo PID Controller Results	XXV
F.3.2.2	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XXVI
F.3.2.3	With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XXVII
F.4	ADAMS Plant Systems	XXIX
F.4.1	A 2-Mass Plant System Case	XXIX
F.4.1.1	With A Simple Servo PID Controller Results	XXIX
F.4.1.2	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XXIX
F.4.1.3	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position Difference Error Compensation Results	XXX
F.4.1.4	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position And Angular Speed Difference Errors Compensation Results	XXXI
F.4.2	A 3-Mass Plant System Case	XXXIII
F.4.2.1	With A Simple Servo PID Controller Results	XXXIII
F.4.2.2	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XXXIII
F.4.2.3	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position Difference Error Compensation Results	XXXIV
F.4.2.4	With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position And Angular Speed Difference Errors Compensation Results	XXXV
F.4.2.5	With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results	XXXVI

F.4.2.6	With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position Difference Error Compensation Results . . .	XXXVII
F.4.2.7	With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position And Angular Speed Difference Errors Compensation Results	XXXVIII

List of Figures

2.1	Flexible transmission elements [1].	11
2.2	A schematic representation of a manipulator model with a motor, a flexible gearbox and rigid link [2].	12
2.3	Different types of beam-like arm flexibilities in three dimensions (twisting and bending) [68].	13
2.4	A schematic representation of a manipulator model with a motor, a rigid gearbox and flexible link [2].	13
2.5	Mass-spring-damper model for a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox.	14
2.6	State-space model for a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox.	16
2.7	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox (2-mass system).	16
2.8	Mass-spring-damper model for a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and a flexible link.	16
2.9	State-space model a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and a flexible link.	18
2.10	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox , an inertia on the output of the gearbox and a flexible link (3-mass system).	19
2.11	Mass-spring-damper model for a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link.	20
2.12	State-space model for a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link.	21
2.13	State-space model for a lumped (multi-mass) model of a flexible link (with n lumped elements).	22

2.14	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (2-mass system case).	23
2.15	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case).	23
2.16	ADAMS model for a flexible manipulator gearbox with a relatively small inertia on the output of the gearbox and FEM model of a flexible link with a load at its end (2-mass system case).	24
2.17	ADAMS model for a flexible manipulator gearbox with a large inertia on the output of the gearbox and FEM model of a flexible link with a load at its end (3-mass system case).	25
2.18	State-space model for a flexible manipulator integrating a motor, and gearbox model in Simulink with an ADAMS FEM model of an inertia on the output of the gearbox and a flexible link.	25
2.19	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of 2-mass system resulting from a motor and a gearbox flexibility driving an ADAMS model with low inertia on gearbox output and FEM model of a flexible link.	26
2.20	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a 3-mass system resulting from a motor and gearbox flexibility driving ADAMS model with large inertia on the output of the gearbox and FEM model of a flexible link.	26
2.21	Frequency response matching of a 2-mass plant system by an identical 2-mass inverse dynamic model.	28
2.22	Frequency response matching of a 3-mass plant system by a 2-mass inverse dynamic model (lower pair of mode and anti-mode matching).	28
2.23	Frequency response matching of a 3-mass plant system by a 2-mass inverse dynamic model (terminal pair of mode and anti-mode matching).	29
2.24	Frequency response matching of a 3-mass plant system by a 2-mass inverse dynamic model (higher pair of mode and anti-mode matching).	29
2.25	Frequency response matching of a 2-mass plant system by an identical 2-mass inverse dynamic model.	30
2.26	Frequency response matching of an ADAMS 2-mass plant system by a multi-mass inverse dynamic model (a 2-mass system).	31
2.27	Frequency response matching of a multi-mass plant system (2-mass system) by an identical 2-mass inverse dynamic model.	31
2.28	Frequency response matching of an ADAMS 3-mass plant system by a multi-mass inverse dynamic model (a 3-mass system).	32
2.29	Frequency response matching of a multi-mass plant system (3-mass system) by a 2-mass inverse dynamic model (lower pair of mode and anti-mode matching).	33

2.30	Frequency response matching of a multi-mass plant system (3-mass system) by an identical 3-mass inverse dynamic model.	33
2.31	Frequency response matching of an ADAMS plant system (2-mass system) by an identical 2-mass inverse dynamic model.	34
2.32	Frequency response matching of an ADAMS plant system (2-mass system) by a 2-mass inverse dynamic model (lower pair of mode and anti-mode matching).	35
2.33	Frequency response matching of an ADAMS plant system (3-mass system) by an identical 3-mass inverse dynamic model.	35
3.1	Scheme of a robotic arm manipulator with two (flexible) links.	37
3.2	Continuous jerk profile in a fifteen-segments “double S” trajectory [3].	41
3.3	Conceptual scheme for the numerical computation of lower-order (jerk, acceleration, velocity and position) profiles of the fifteen-segments trajectory starting from the derivative of jerk “snap” profile [3].	41
3.4	Fifteen segments trajectory (position, velocity, acceleration, jerk and snap).	42
3.5	Iterated (twice) fifteen-segments trajectory profiles (position, velocity, acceleration, jerk, snap and its derivatives up to the second derivative).	43
3.6	PID control based on PI feedback of velocity error.	44
3.7	Scheme for reference trajectory generation and inverse dynamics plus FFW computed compensation torque controller.	46
3.8	A schematic representation of the 3-mass manipulator inverse dynamic model (comprised of a motor, a flexible gearbox and a rigid link) for compensation purposes, together with its system I/O abstraction.	49
3.9	A schematic representation of a manipulator plant model with a motor, a flexible gearbox and a flexible link with the motor and “fake” link angular positions demonstrated.	50
3.10	A flexible link mounted on a motor inertia via a flexible gear and how the inverse dynamics can be used to control it in a feed-forward fashion.	50
3.11	A compensation scheme and its usage.	51
3.12	A two-links manipulator with a flexible link mounted on a rigid link.	51
3.13	A two-links manipulator with a flexible link mounted with a right angle on a rigid link.	52
3.14	A manipulator with two flexible links with some important definitions illustrated.	52
4.1	Flexible FEM link’s tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller, in addition to both angular position and angular speed difference errors compensation (a 3-mass plant system).	58
5.1	A manipulator link with a rigid part having the same length as the outer flexible part. The output of a flexible gearbox (implemented in Simulink) is connected to the rigid part at its left end.	61

5.2	Proposal of how to connect the modules developed in the thesis to control a manipulator of two degrees of freedom.	62
A.1	The ADAMS/Controls block diagram of the FEM flexible robotic link with the gearbox inertia in MATLAB/Simulink.	IV
B.1	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox (2-mass system with $k_g = 32$).	VI
B.2	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox (2-mass system with $k_g = 45$).	VII
B.3	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox (2-mass system with $k_g = 65$).	VII
C.1	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox , an inertia on the output of the gear box and a flexible link (3-mass system with $k_g = 32$ and $k_l = 20$).	VIII
C.2	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox , an inertia on the output of the gear box and a flexible link (3-mass system with $k_g = 45$ and $k_l = 20$).	IX
C.3	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox , an inertia on the output of the gear box and a flexible link (3-mass system with $k_g = 65$ and $k_l = 20$).	IX
D.1	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case with $n = 5$).	X
D.2	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case with $n = 10$).	XI
D.3	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case with $n = 15$).	XI

D.4	Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case with $[n = 5, 10 \text{ and } 15]$).	XII
F.1	Flexible Gearbox angular position, angular speed and angular acceleration reference tracking using PID controller only (a 2-mass plant system with $k_g = 5N/m$).	XVI
F.2	Flexible Gearbox angular position, angular speed and angular acceleration reference tracking using identical 2-mass inverse dynamics with PID controller (a 2-mass plant system with $k_g = 5N/m$).	XVII
F.3	Comparison of gearbox's angle, speed and acceleration tracking results obtained using various controllers for a 2-mass plant system. . .	XVII
F.4	Flexible link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (a 3-mass plant system with $k_g = 5N/m$ and $k_l = 5N/m$).	XVIII
F.5	Flexible link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [lower pair of mode and anti-mode matching] with PID controller (a 3-mass plant system with $k_g = k_l = 5N/m$).	XIX
F.6	Flexible link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [terminal pair of mode and anti-mode matching] with PID controller (a 3-mass plant system with $k_g = k_l = 5N/m$).	XX
F.7	Flexible link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [higher pair of mode and anti-mode matching] with PID controller (a 3-mass plant system with $k_g = k_l = 5N/m$).	XXI
F.8	Flexible link's tip angular position, angular speed and angular acceleration reference tracking using identical 3-mass inverse dynamics with PID controller (a 3-mass plant system with $k_g = k_l = 5N/m$). . .	XXII
F.9	Comparison of link's angle, speed and acceleration tracking results obtained using various controllers for a 3-mass plant system.	XXII
F.10	Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 2-mass plant system).	XXIII
F.11	Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics with PID controller (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 2-mass plant system).	XXIV
F.12	Comparison of link's angle, speed and acceleration tracking results obtained using various controllers for a multi-mass plant with a lumped link parameters that results in a 2-mass system.	XXV

F.13 Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 3-mass plant system).	XXVI
F.14 Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [lower pair of mode and anti-mode matching] with PID controller (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 3-mass plant system).	XXVII
F.15 Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 3-mass plant system).	XXVIII
F.16 Comparison of link's angle, speed and acceleration tracking results obtained using various controllers for a multi-mass plant with a lumped link parameters that results in a 3-mass system.	XXVIII
F.17 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (a 2-mass plant system).	XXIX
F.18 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics with PID controller (a 2-mass plant system).	XXX
F.19 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics with PID controller, in addition to angular position difference error compensation (a 2-mass plant system).	XXXI
F.20 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics with PID controller, in addition to both angular position and angular speed difference errors compensation (a 2-mass plant system).	XXXII
F.21 Comparison of link's angle, speed and acceleration tracking results obtained using various controllers for an ADAMS plant with a FEM link parameters that results in a 2-mass system.	XXXII
F.22 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (a 3-mass plant system).	XXXIII
F.23 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [lower pair of mode and anti-mode matching] with PID controller (a 3-mass plant system).	XXXIV
F.24 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics [lower pair of mode and anti-mode matching] with PID controller, in addition to angular position difference error compensation (a 3-mass plant system).	XXXV

F.25 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics[lower pair of mode and anti-mode matching] with PID controller, in addition to both angular position and angular speed difference errors compensation (a 3-mass plant system).	XXXVI
F.26 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller (a 3-mass plant system).	XXXVII
F.27 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller, in addition to angular position difference error compensation (a 3-mass plant system).	XXXVIII
F.28 Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller, in addition to both angular position and angular speed difference errors compensation (a 3-mass plant system).	XXXIX
F.29 Comparison of link's angle, speed and acceleration tracking results obtained using various controllers for an ADAMS plant with a FEM link parameters that results in a 3-mass system.	XL

List of Tables

4.1	Angular errors and speed errors when different inverse dynamic model types are used for feed forward control. The errors are recorded for the different link dynamic models developed in the thesis.	55
4.2	A rearrangement of Table 4.1 to easier see which inverse dynamic model type that best suits a given link dynamic model type.	56
4.3	Results from different matching strategies for the inverse dynamic 2-mass model when the link model has two resonances. In this case, the link model is a lumped 3-mass model.	57
4.4	An ADAMS system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller with angular position and angular speed difference errors compensations results	58
F.1	A 2-mass plant system tracking with a PID controller only results . .	XV
F.2	A 2-mass plant system tracking with an identical 2-mass inverse dynamics and PID controller results.	XVI
F.3	A 3-mass plant system tracking with a PID controller only results . .	XVIII
F.4	A 3-mass plant system tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller results . .	XIX
F.5	A 3-mass plant system tracking with a 2-mass inverse dynamics (terminal pair of mode and anti-mode matching) and PID controller results	XIX
F.6	A 3-mass plant system tracking with a 2-mass inverse dynamics (higher pair of mode and anti-mode matching) and PID controller results . .	XX
F.7	A 3-mass plant system tracking with an identical 3-mass inverse dynamics and PID controller results	XXI
F.8	A multi-mass system (2-mass plant case) tracking with a PID controller only results	XXIII
F.9	A multi-mass system (2-mass plant case) tracking with a 2-mass inverse dynamics and PID controller results	XXIV
F.10	A multi-mass system (3-mass plant case) tracking with a PID controller only results	XXV
F.11	A multi-mass system (3-mass plant case) tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller results	XXVI
F.12	A multi-mass system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller results	XXVII

F.13	An ADAMS system (2-mass plant case) tracking with a PID controller only results	XXIX
F.14	An ADAMS system (2-mass plant case) tracking with a 2-mass inverse dynamics and PID controller results	XXX
F.15	An ADAMS system (2-mass plant case) tracking with a 2-mass inverse dynamics and PID controller with angular position difference error compensation results	XXX
F.16	An ADAMS system (2-mass plant case) tracking with a 2-mass inverse dynamics and PID controller with angular position and angular speed difference errors compensations results	XXXI
F.17	An ADAMS system (3-mass plant case) tracking with a PID controller only results	XXXIII
F.18	An ADAMS system (3-mass plant case) tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller results	XXXIV
F.19	An ADAMS system (3-mass plant case) tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller with angular position difference error compensation results	XXXIV
F.20	An ADAMS system (3-mass plant case) tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller with angular position and angular speed difference errors compensations results	XXXV
F.21	An ADAMS system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller results	XXXVI
F.22	An ADAMS system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller with angular position difference error compensation results	XXXVII
F.23	An ADAMS system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller with angular position and angular speed difference errors compensations results	XXXVIII

1

Introduction

1.1 Introduction

Nowadays, robotic manipulators are widely used to efficiently carry out different tasks in dangerous, monotonous and/or tedious jobs. They are being utilised extensively in various industrial applications such as painting, welding, assembling, laser cutting, etc. Those tasks naturally have different performance requirements. Most of industrial robotic manipulators are designed for high performance and long lifetime and are implemented with high stiffness links. The main flexibility is then usually located in the gearboxes and joint bearings. The control schemes are designed in order to obtain good position tracking accuracy in case of these flexibilities are influential. With the growing demand for “cheap” mass-production and robot safety, lighter robot structures are needed, which means experiencing performance limitations because of the excitation of elastic modes in the arm system. With also more requirements on higher speed and acceleration to obtain shorter manufacturing cycle times, the effects of elastic dynamics of the arm structures cannot be neglected anymore.

A robotic manipulator with only flexibilities in gears and bearings can only have a finite number of degrees of motion freedom (DOFs), that are usually described by ordinary differential equations (ODEs). On the contrary, a manipulator with flexible links has an infinite number of DOFs, which are preferably specified by partial differential equations (PDEs) [4]. However, the flexible links will usually have a geometry such that only one vibration mode is excited for each link during the manipulator motion and it is therefore sufficient to use models with finite number of degrees of freedom when controlling a manipulator with flexible links. The problem is not to model the vibration modes but is to do so while simultaneously taking care of the change in the kinematic structure of the manipulator with dynamically bending and twisting links.

When the motion programming of the manipulator is made, the dynamic deformation of the links during the programmed movements is not known. Hence, it is not possible to compensate for the link deflections without running the robot or a model of the robot at first. One possibility to solve this problem when the program can be run in advance is to simultaneously calculate the differential equations and the kinematics of the robot by solving a system of Differential Algebraic Equations (DAE) [66]. The problems with this solution are that the whole trajectory must be known in advance and that the calculations are too heavy to be possible to run

in real time for manipulator control. The requirement to know the trajectory in advance makes it impossible to use, for example, sensor-based control.

A new alternative method to solve the problem with the influence of link deflections on the kinematics of a manipulator is presented in this thesis. The basic idea is to use a model of the dependence of the so called fake angle of a bending and/or twisting link on the deflection of a rigid link with a local flexibility and with the same frequency response. The compensation is then made in real time from the calculated inverse model used for feed-forward control.

The suggested inverse dynamics control and compensation schemes are tested and verified in a co-simulation environment comprising both Finite Element Method's (FEM) models including the flexible links in ADAMS/View and Matlab/Simulink. The results of the tracking performance are then compared to the reference case without using feed-forward control or compensation of link deflections.

1.2 Motivation and Problem Justification

Lightweight manipulators with flexible links offer several advantages in comparison to their traditional rigid and bulky counterparts. The advantages include the requirements of relatively smaller actuators, lower overall mass, lower energy consumption and lower overall cost in general. Due to these characteristics, this class of high-productivity manipulators are especially suitable for a number of non-conventional robotic applications such as: space exploration missions robotic manipulators (for example NASA Remote Manipulator System), fence-less robots for physical human-robot interaction, mobile robotics, robots on tracks, etc. Besides, link flexibility study is enforced also for heavy manipulators such as robotic excavators used in mining operations, robotic crane systems used in construction applications, tunnel inspection robots, oil/gas platforms robots, big robots in unstructured environments and the like. In either case, size, mass, speed payload, environmental or task-related constraint factors will motivate the use of more or less flexible link systems. It should also be pointed out that traditional industrial robots with relatively rigid links are used in high precision and high speed applications as laser cutting, plasma cutting, water jet cutting and deburring. To increase the performance of industrial robots in such applications compensation of small link deflections could be necessary. In such a case, it should be emphasised that the elastodynamics of the robotic manipulator system must be considered and incorporated at the early design stages to obtain resonant link modes and link deflections that will be well adapted to the control requirements with respect to, for example feed-forward control using inverse dynamic models including flexible links.

During the latest 20 years, subtle design, exact modelling, critical analysis and precise control and effective and efficient operation of flexible manipulators are hot topics in both the robotics and control research communities. The investigation of dynamics properties of a robotic manipulator with flexible links is sometimes carried out using absolutely rigid mechanical models for both the links and the joints. This simplification, however, leads to intolerable problems in the execution

of operational tasks (for example bad accuracy of a laser-cutting tool mounted at the end-effector of a high-speed robotic arm). Thus, in many cases the elasticity of the manipulator with flexible links needs to be taken into account when analysing its dynamics, especially when the task's precision is largely influenced. For a wide range of industrial manipulators with flexible links, the structural elastic compliance is concentrated in its joint, gears and (harmonic) drives, and that is why most of today's manipulators with flexible links can only handle the flexibility in the joints but not in the links. To be able to fully exploit the advantages of this class of mechatronic systems and generalise them to cover other application areas, very accurate models that can also capture the elastic behaviours of compliant links, together with effective control strategies are needed [5].

1.3 Literature Review

Substantial work has been done on both the modelling and the control of flexible robotic manipulators, and hence plenty of relevant publications in both areas are available. Benosman and Vey [11] presented a survey that covers the control of flexible manipulators in the period from 1990 to 2002, citing mostly those dealing with the multi-link manipulators. Additionally, S. K. Dwivedy and P. Eberhard [12] compiled a more recent and richer review work of literature around dynamic analysis of flexible manipulators, with many other publications ranging from 1974 to 2005 and classifications based mainly on the number of flexible links used in the study together with the control methods.

1.3.1 Earlier Studies in the Field of Flexible Robotic Manipulators

In this section, a survey of the literature related primarily to dynamical analysis and control design of flexible robotic manipulators is carried out, considering both joint and link flexibility.

Pioneering research in the area of modelling and control of flexible link manipulators was driven by the interest in utilising flexible manipulators for space exploration missions in the late 1970s. The motive behind this was the fact that a space robotic manipulator should be the lightest possible in order to reduce costs attributed to/associated with launching process. Many researchers had studied various aspects (Modelling, Control, Simulation, Experimental Testing and Mechatronic Design) of flexible manipulators used for space applications. In 1979, Book [13] analysed massless elastic chains with servo controlled joints by using the lumping approximation to account for the deflection of elastic elements under load, and later in 1984, Book [14] investigated flexible manipulator dynamics using a recursive Lagrangian formulation. At around the same time (in 1984), Cannon and Schmitz published one of the first papers on the flexible manipulators control [15]. The problem of feedback control innate in flexible links was considered by Sakawa et al. [16] in 1985. Yang

and Donath [17] developed a dynamic model of a flexible arm with joints flexibility integrated in 1988.

The period from the late 1980s to 1990s saw a great number of publications on control models of flexible manipulators. For example, De Luca and Siciliano [18] addressed the trajectory control of nonlinear single-link flexible arm by considering the additional gravity factor in their control model. In 1990, a complete study of different equations of flexible manipulators with various boundary conditions was conducted by Bellezze et al. [19]. Asada et al. [20] have used a finite-element model to study the optimal design problem of flexible arm by increasing its fundamental vibration frequency through optimal beam tapering.

Most of the research work until the 1990s beginnings was focused primarily on the optimisation control of flexible manipulators without any practical consideration for the problem from structural dynamics or shape optimisation perspective. However, some researchers such as Belvin and Park [21] used structural tailoring, and Asada Park and and Rai [20] included shape design integrally in the overall control scheme. In 1991, Wang et al. [22] formulated an optimisation model for a flexible manipulator. They then produced its first optimum mode shapes and modal frequencies using a modified iteration algorithm they developed. In 1994, an excellent paper treating different models of flexible manipulators while examining the effect of load tip-masses on their fundamental frequencies was published by Wang and Guan [23]. They demonstrated how both shear deformation and rotary inertia will not affect so much the fundamental vibration frequency. Wang and Russel, based on Euler-Bernoulli model of flexible manipulators with tip loads, generated diverse examples of tip load design vectors [24]. They continued investigating the matter and extending their work by applying optimality of shape design [24] in their mechatronic approach [25], where they presented an innovative new segmentised flexible manipulator model opening the problem of design optimality to numerous techniques and algorithms of mathematical programming.

1.3.2 Recent and Relevant Work on Flexible Manipulators

In this part of the literature review, recent research work, that has inspired the author in some way or another is cited. It represents a focused overview to help in systematically building the required theoretical concepts and also serve as a practical guideline to facilitate following the rest of the thesis.

Numerous researchers have addressed the control problem associated with flexible manipulators. Initial work was focusing on state-space model. For example, Cannon and Schmitz [15] first studied the possibility of using the linear quadratic Gaussian (LQG) controller with states reconstructed via estimation. Kalman filtering was used for estimation of both rigid and flexible systems needed for control by Lin and Lewis [20]. Geniele et al. [26] used two loops controller with the inner one for stabilisation and the outer one for tracking of joint angle and some point on the link, using input-output inversion method. Madhavan and Sing [27] used the same approach, in addition to a sliding mode control for a robust controller design that can handle wide range of uncertainties.

The transfer function approach is adopted in other studies. For instance, Wang and Vidyasagar [28] used the tip deflection as output and then deduced a passivity-based output feedback of the model's transfer function. Siciliano and Book separated the flexible manipulator into two time constants yielding a slow system and a fast system, with the help of perturbation theory. They argued that the slow system is nonlinear and the fast one controller is linear. Besides, they successfully designed both a nonlinear controller and linear state-feedback controller for the two systems respectively. Aoustin et al. [29] did similar work, and revised slow and fast systems were utilised by Moallem et al. (1997).

A novel lumped-parameter dynamic model is presented for a distributed flexible manipulator system by S.S. Ge et al [67]. The suggested lumping method is motivated by the Finite Element Method (FEM) and associated with the modal analysis, where the distributed system is lumped to a cascade system of weightless linear angular springs and concentrated point masses. It was shown that increasing the number of lumped basic elements, equivalent springs and concentrated masses, improves the accuracy of the lumped dynamic model in reflecting the dominant dynamic behaviour of the original flexible manipulator system. A comparison, for both time and frequency-domain responses, between the newly introduced model and the generally used Assumed Modes Model (AMM) is provided based on open-loop simulations, which serve as a verification for the resulting model.

Some computational intelligence control approaches have been also used for the purpose of flexible manipulators control. Guiterrez et al. [30] suggested a control scheme that uses a neural network (NN) for tracking. A robust H_∞ controller for a flexible beam that is subject to perturbations and delay is proposed by Lenz et al. [31]. Similar concept is used by Tchernychev et al. [32] with constraints added and treated directly in the time-domain. In the same year (1997), a back-stepping technique using lumped mass-spring system to model the beam for the purpose of tip tracking is examined by Zhu et al. [33] in an attempt to develop a robust controller that can tolerate the system uncertainties/disturbances. Jnifene and Fahim [34] introduced active damping into the flexible manipulator system to tackle its nonminimum phase characteristics, and proposed a delayed deflection/computed torque approach depending on the deflection sensing point. In 1998, Liang et al. [35] discussed the free-floating space manipulator problem and represented it as a conventional manipulator with a fixed-base. This mapping opened the doors toward using well-understood methods in building manipulator systems, especially after Ghanekar et al. [36] have produced the large robotic manipulator scaling laws for linear controllers, which resulted in allowing the use of small-scale prototypes to perform the design task.

Concerning simulation work, Tang and Lu [37] utilised a video system to predict the model shapes and states of vibration. Hastings and Book [38] and Fukuda [39] used strain gauges to measure the amount of deflection. An accelerometer is used to measure the beam vibration by Chalhoub et al. [40]. Additionally, external photo-sensors are used to assist detecting the displacement.

Simplifying modelling of a flexible link by using a port-Hamiltonian-based approach has also been proposed by A. Machelli et al. [41] in 2007. In such a case that is very

different from classical modelling approaches that are based on either Euler-Bernoulli or Timoshenko beam theories, the suggested model has the ability to describe large deflection in 3D space without relying on any finite dimensional (e.g. modal) approximation. By combining rigid bodies, springs, dampers, joints and flexible links at the end, modelling and mathematical description of whatever complex mechanical structure formed by the beams is possible.

A vision-based controller for a flexible beam tip point with two-time-scale is used by Xu and Ritz [42] to track a desired tip point reference while mitigating its vibration. The controller showed better performance than other tested controllers. A very similar study, but using visual servoing of eye-in-hand flexible manipulators is also presented by L. Bascetta et al. [43]. In this work, both rigid and flexible motion modes of the manipulator are fully considered in the control solution to the two-time scale problem. The visual information from the sensor was used for task-space-oriented control law computation in the “slow” part, where the relatively more computationally expensive operations, such as inverting the Jacobian or taking its time derivative, are avoided. A constructive Lyapunov-based stability proof is also given, and the controller effectiveness is supported by means of numerical simulation for the trajectory tracking problem. Furthermore, some experimental tests are demonstrated with their results presenting the precision improvement upon using the suggested algorithm for flexible single-link manipulator.

An adaptive controller for flexible single-link manipulators is constructed by Becedas et al. [44]. A generalised proportional integral (GP)-type output feedback controller combined with a fast online closed-loop identification method is designed to control an uncertain flexible robotic arm with unknown tip mass load, integrating a Coulomb friction model in the motor dynamics. M. Hosaka and T. Murakami [45] carried out simulation studies and experiments on the vibration control of flexible arm by multiple observer structure; using the acceleration feedback, and a multiple-observer structure for disturbance rejection.

An augmented stable fuzzy control for a flexible robotic arm using a neuro-fuzzy state-space modelling and a linear matrix inequality (LMI) approach is proposed by Chatterjee et al. [46]. Issues related to nonlinear controller design for flexible robotic manipulators with an adaptive model are discussed by Dogan and Stefanopoulos [47]. They used a dynamics state-feedback controller to robustly regulate the rigid modes as well as suppress elastic vibrations. Since highly nonlinear flexible multi-link arms are largely uncertain, due to backlash, variable payload and external disturbances, the internal model has to be tuned up adaptively for unknown disturbances, in parallel with a robust stabiliser, which is optimised with a new revolutionary algorithm. C. A. Monje et al. [48] developed a fractional order controller to control a lightweight flexible manipulator’s tip position. Some of the interesting features of this work are: the controlled system overshoot is not dependent on the tip mass load allowing a constant safety zone to be delimited always (advantage in planning collision avoidance), noise and motor saturation considerations.

W. J. O’Connor [49] provided a foundation for wave-based analysis and control of lump-modelled flexible robots, by exploring the nature and validity of wave concepts in lumped flexible robotic systems. A novel exact, but not unique, wave-based model

of uniform mass-spring systems is proposed and verified, leading to simple, powerful, robust and generic control strategies for flexible robotic systems.

Causal inversion-based output tracking control of flexible one-link manipulator, a non-minimum phase system, is investigated in [50]. The suggested new controller achieves stable real-time reference trajectory profile tracking via causal inversion approach. The simulation results of controller testing on tip trajectory tracking of a flexible one-link manipulator demonstrated the effectiveness in output tracking.

Active control of flexible manipulators using a nonlinear “conventional” motor with gearbox actuators is studied by Gomez et al [51], where a friction compensation using neural networks is used to solve the problem of existence of dead-zone in the torque caused by the nonlinear friction inside the actuator.

After the initial studies in optimal shape design by Wang [24, 25, 52–56], no much progress has been noticed in the research area of optimal structures of flexible manipulators. A new method for computing optimal mass and rigidity distributions has been presented by Wang et al. [22] for a flexible manipulator with tip-mass load. The goal is to consider robustness with respect to design specifications and known constraints (focused on open-loop design), hence other practical aspects can be treated. He concluded that following a proper design guideline for the flexible beam shape can reduce the vibration. But, in reality, all manipulators are operated in a closed-loop fashion to achieve high performance, and unfortunately answering the question of “how to design a flexible beam shape from a control perspective?” is still a challenging task!

A very recent work on inverse dynamics of flexible manipulator is carried out by Stig Moberg and Sven Hanssen [66]. They have investigated different methods for the inverse dynamics of the extended flexible joint model (a general manipulator model that includes joint flexibility), whereas they have used the inverse dynamics to obtain the required feed-forward control action, that is often used for high-precision control of robotic manipulators. The inverse dynamics solution was computed by solving the a high-index differential algebraic equation (DAE) using a constant step-size constant-order backwards differentiation formula (BDF), while it is also concluded that scaling of the algebraic equations and their derivatives is important. The inverse dynamics can be treated as an initial-value problem in case of minimum phase systems (stable systems with stable zero dynamics). On the other hand, an optimisation approach based on the discretised DAE is suggested for unstable zero dynamics. Also, another optimisation method, using a continuous DAE formulation, is suggested and evaluated. The solvers are illustrated by simulation, using a manipulator with two actuators and five degrees-of-freedom.

1.4 Objectives

The objective of this Master’s thesis research is to study one promising possibility for modelling, angular position tracking control and simulation of the end-effector’s tip of a flexible multi-link robotic manipulator system, in order to obtain an efficient controller that can be implemented in real-time settings for industrial robotics

applications. Because of the overall problem complexity, a modular work scheme is adopted, whereas the modelling and control of flexible single-link manipulators are studied thoroughly serving as a starting point and a basis that can later be extended to the multiple links case.

Utilising feedback from sensors on the arm system for control purposes of a manipulator (non-collocated control) may give rise to performance problems, due to the non-minimum phase nature of the resulting close-loop system. To avoid such problems, this thesis work targets only the case with joint-motor sensor feedback. Thus, the work is based on feed-forward control using the inverse dynamics of the robot arm including the gearbox. Frequency response matching was made for the tip-position tracking of the flexible link and implemented in a co-simulation environment including ADAMS and Matlab/Simulink. Therefore, this research is composed of two main parts/objectives: (1) the frequency-based inverse dynamics collocated tracking control of minimum phase systems, and (2) modelling and control of flexible single-link manipulators.

1.5 Thesis Layout

The rest of this thesis is organised as follows:

Chapter 2 (Theory): This chapter describes the sources of flexibility in robotic manipulators, the application areas of flexible robotic arms, and theoretical background of the suggested modelling and control techniques used, including all assumptions and considerations for the system's operation.

Chapter 3 (Problem Statement, Control System Design and Compensation Techniques): This chapter details the objectives of the project, and focuses on how those objectives were achieved at each step in a modular way to allow testing, modifications and correction after simulating the controlled planar robotic manipulator (benchmark) system in real-time settings. Also, it presents the proposed control and compensation schemes are implemented within the complete controlled (close-loop) system.

Chapter 4 (Simulation Results and Discussion): This chapter discusses the results obtained after implementing the system, the evaluation of the system according to the results obtained and the discussion of those results.

Chapter 5 (Conclusions and Future Work): The project is summarised in this chapter, discussing the achieved objectives. In addition to the further developments that can be studied to improve the resulting system.

Appendix A (ADAMS-MATLAB/Simulink Interface and Performing Co-Simulations) describes the detailed steps for carrying out a co-simulation between ADAMS and MATLAB environments.

Appendix B (Frequency Responses of Additional 2-Mass Models Used in Simulations) presents the frequency responses for additional 2-mass plant models that are generated during simulations work throughout the thesis work.

Appendix C (Frequency Responses of Additional 3-Mass Models Used in Simulations) presents the frequency responses for additional 3-mass plant models that are generated during simulations work throughout the thesis work.

Appendix D (Frequency Responses of Additional Multi-Mass (3-Mass) Models Used in Simulations) presents the frequency responses for additional multi-mass plants (3-mass models) that are generated during simulations work throughout the thesis work.

Appendix E (Steps for Mode/Anti-Mode Matching in the Frequency Plane Using Inverse Dynamics Models) illustrates the recipes followed to tune the 2-mass and 3-mass inverse dynamic models frequency responses to match those of the plant models controlled.

Appendix F (Detailed Results and Comparisons) shows additional results and comparative charts that are generated during the simulation work throughout the thesis studies, but not included in the main text of the thesis.

2

Modelling of Flexible Robotic Manipulators

In this chapter, various aspects related to the models used for simulating the behaviour of a flexible robotic arm are discussed. First, different sources of both joint and link flexibilities in a robotic manipulator are presented highlighting the benefits and problems associated with the introduction of those flexibilities in the manipulator system. After that, the dynamical models needed to study the flexible robotic arms are presented. To facilitate understanding of the flexible manipulator models used for the (inverse dynamics) control and compensation design at a later stage of the thesis work, the simplest 2-mass and 3-mass models are first introduced and the model complexity is then gradually increased. This is done by approximating the flexible link by a series lumped mass-spring-damper elements that can adequately capture the interesting flexibility phenomena of a FEM model of the real flexible link with the coupled gearbox inertia. This FEM model is developed in ADAMS/View with the help of ADAMS/Flex, in addition to gearbox flexibility part modelled in Matlab/Simulink. It represents our best approximation for the real flexible manipulator plant. At each step, the transfer function, state-space representation and frequency response of the manipulator model under study are presented. At the end of the chapter, the possibility for matching the resonant and ant-resonant frequencies is investigated, with the results of the matching process for different plant/inverse dynamic models combinations shown.

2.1 Sources of Flexibility

2.1.1 Joint Flexibility

Effective modelling, real-time simulation and control of flexible manipulators is presently still an open problem. Most of the contemporary scientific research and experimentation in academia or industry has only considered flexibility at joints. This seems acceptable for the majority of the manipulators available nowadays, especially in application where the joints flexibility effects dominate those of the links flexibility. To elaborate more on this, one can notice that lots of today's industrial robotic manipulators are driven by harmonic drive gears. Such kind of gears are used frequently because of their low weight, compactness and ability for large torque transmission with almost negligible backlash. Nonetheless, flexibility present

in the gears and joint bearings of the drive can dramatically affect the manipulator behaviour. It has been well established both theoretically and experimentally (for example in [57–59]) that, for a large class of manipulators, including joint flexibility in the modelling and control steps has a great impact on the resulting tracking performance. Figure 2.2 shows a schematic representation of flexible joint of a manipulator.

Joint flexibility (or, in a more precise term, drive elasticity) stems from several sources, such as various deformations in gears, belts, shafts, hydraulic lines, bearings, transmissions, ... etc (see Subfigure 2.1a). Among flexible gears, compact gears as the Harmonic Drive gears (see Subfigure 2.1b) are commonly used elements in robotics because they allow obtaining high reduction ratios at low backlash levels. The Harmonic Drive gears utilise a deformable component (called “flexspline”) that forms the basis of its operating principle.

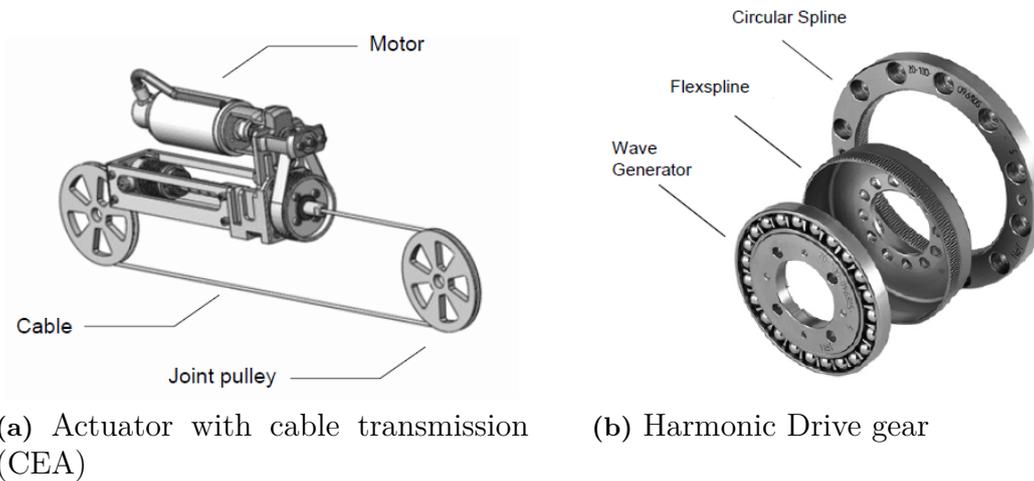


Figure 2.1: Flexible transmission elements [1].

Neglecting joint flexibilities in control laws leads to degraded accuracy of the tracking and may even cause instability if applied to real manipulators. For this reason, the flexibility in joints should be integrated when designing controllers for fast manipulators. Including joint flexibility in the manipulator model alters the dynamical equations of motions of the manipulator in addition to increasing their number (more DOFs, and hence more dynamical constraints to be satisfied than in a rigid model). In many cases, flexibility at a joint can be modelled by a linear, torsional spring with some inherent damping. As a consequence of the increase in the number of degrees of freedoms, flexible-joint manipulators control is more complicated. Though, tackling joint flexibility is much easier than handling the distributed flexibility of a link. One interesting practical aspect to be noted here is that link flexibility can approximately be modelled using a chain of rigid sub-links that are connected together by intermediate flexible joints. Thus tackling the distributed link flexibility problem can be made easier by such incorporation of joint flexibility as a first step [2].

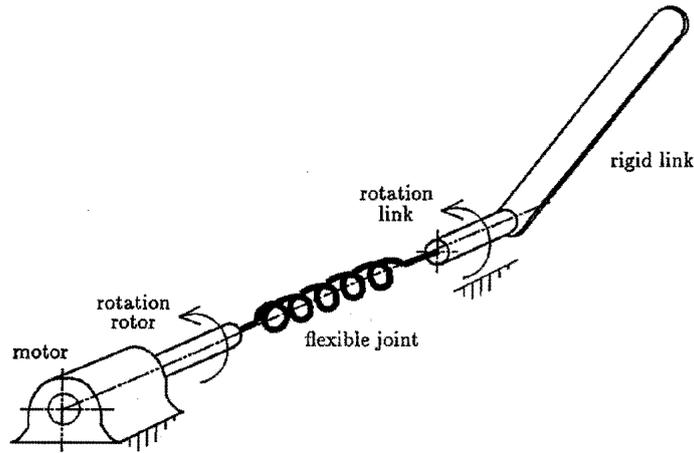


Figure 2.2: A schematic representation of a manipulator model with a motor, a flexible gearbox and rigid link [2].

2.1.2 Link Flexibility

The downsides and shortcomings encountered upon using a stiff (and hence relatively heavy) manipulator can be circumvented, to some degree, if lightweight links and drives are used instead. As a consequence of such a choice, the ratio between payload and manipulator weight is relatively larger in comparison to the stiff, bulky manipulator. Additionally, motor power is reduced and the resulting light weight manipulator may have shorter response time (i.e. faster). Out of the numerous advantages are: improved mobility, safer operation and lower construction costs. On the other hand, usually lightweight manipulators suffer from large link deformations, imposing lots of constraints and introducing very challenging modelling and control tasks, if accurate trajectory tracking is needed, which often is the case.

It turns out that the design of controllers for such lightweight systems is very complicated, since their resulting dynamical models are highly nonlinear with (far) more degrees of freedom than the usually available number of actuator inputs. Theoretically, to describe the motion of a flexible link, infinite number of degrees of freedom is needed [60].

Different type of flexibilities that can occur in a beam-like manipulator arm are depicted in Figure 2.3. Subfigure 2.3a shows twisting around the vertical z -axis, Subfigure 2.3b shows bending around the y -axis, Subfigure 2.3c shows twisting around a z -axis + bending around the y -axis, and Subfigure 2.3d shows bending around both the x - and the y -axis. In all these plots, the z -axis is going through the beams, and the beams are located on the (xy) -plane. Furthermore, it is important to note that since the workspace of the manipulator test-bench model adopted for this study (shown later in Figure 3.1) is defined in \mathbb{R}^2 , its motion must also be evaluated in this two-dimensional Euclidean space only (i.e. only planar motions are possible), the link oscillations can only occur around one axis. Therefore, only bending as shown by Subfigure 2.3b is dominantly possible [68].

Figure 2.3 shows different types of beam-like arm flexibilities in three dimensions.

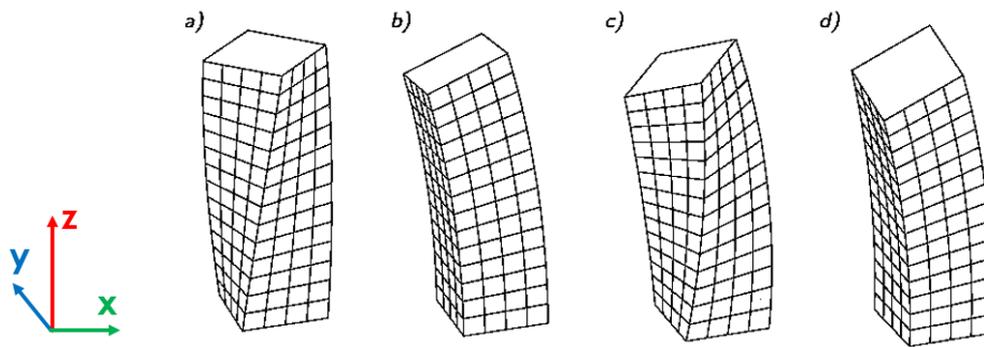


Figure 2.3: Different types of beam-like arm flexibilities in three dimensions (twisting and bending) [68].

Figure 2.4 shows a schematic representation of flexible link of a manipulator.

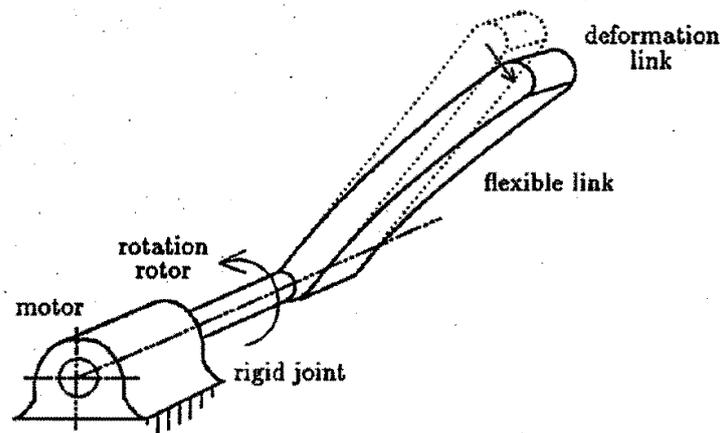


Figure 2.4: A schematic representation of a manipulator model with a motor, a rigid gearbox and flexible link [2].

Using finite element methods (FEMS) or boundary integral methods (for example in [61–64]), flexible links can be approximately modelled by finite dimensional systems, which adequately meets the necessities of a model from a control perspective. Otherwise, some model reduction techniques have to be used, for example the Guyan reduction method [65].

2.1.3 Manipulators with both Joint and Link Flexibilities

The dynamical equations that governs that motion of multi-link manipulators under consideration of both joint and link flexibilities are unduly complex, and the number of actuators available for control purposes is much smaller than the number of degrees of freedom. In addition, the associated solution of kinematical problems to determine the desired reference motor trajectory corresponding to a given end-effector trajectory is not unique. Further demanding is the fact that the usual control objective of such manipulators is to track a prescribed end-effector's trajectory, while limiting the vibrations as much as possible. The problem of trajectory

tracking in manipulators with the combined flexibility of both joints and links is rarely tackled in literature. Therefore the thesis contribution presented here is an attempt to enrich the research around the topic in both industry and academia.

2.2 Dynamical Models for the Study of Flexible Robotic Arms

In this part of the thesis report, an extensive study for the modelling and control of flexible robotic manipulator arm is presented. In order to perform analysis of control schemes for a manipulator arm with compliant links, it is necessary to understand the dynamics of elastic systems in question. Since the targeted new control method will approximate a link with bending, or twisting, flexibility with a stiff arm having a joint flexibility, it is important to understand the dynamics of these kind of systems. Thus, models of 2-mass and 3-mass systems will be analysed as well as models of multi mass systems and FEM models. For each model the equation of motion, the transfer function from motor torque to motor acceleration, the state space representation and the frequency response from motor torque to motor acceleration are presented. The relations between motor torque and acceleration are shown since these are used when tuning the inverse models for feed-forward control.

2.2.1 2-Mass System Dynamic Model of a Motor with Flexible Gearbox Drive Train

The simplest model used in the thesis is the 2-mass model consisting of a first inertia (motor inertia) connected to a second inertia (for example the load on a gearbox) by a spring in parallel with a damper. An example of such a simplified mass-spring-damper model of a drive train of a flexible robotic manipulator consisting of only a motor (torque T_m , inertia J_m) and a flexible gearbox (stiffness k_g , damping d_g) and a rigid link (with inertia J_g on the output of the gearbox) is shown in Figure 2.5.

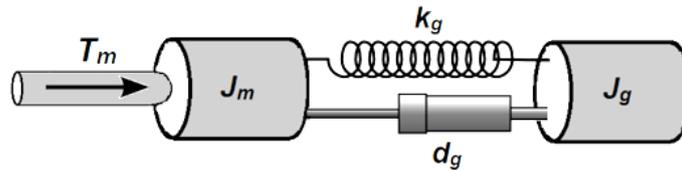


Figure 2.5: Mass-spring-damper model for a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox.

The corresponding equations of motion resulting from applying Newton's second law (in differential equations form) are:

$$\begin{aligned} 0 &= J_g \ddot{\varphi}_g + d_g(\dot{\varphi}_g - \dot{\varphi}_m) + k_g(\varphi_g - \varphi_m) \\ \tau_m &= J_m \ddot{\varphi}_m - d_g(\dot{\varphi}_g - \dot{\varphi}_m) - k_g(\varphi_g - \varphi_m) \end{aligned} \quad (2.1)$$

where (respecting order of subscripts):

φ_m, φ_g : Motor angular displacement and gearbox angular displacement (rad).

$\dot{\varphi}_m, \dot{\varphi}_g$: Motor angular speed and gearbox angular speed (rad/s).

$\ddot{\varphi}_m, \ddot{\varphi}_g$: Motor angular acceleration and gearbox angular acceleration (rad/s^2).

J_m, J_g : Motor inertia and gearbox inertia ($(kg.m^2)$).

k_g : Gearbox stiffness (N/m).

d_g : Gearbox damping (N/m^2).

The above set of motion equations can be described by the following transfer function:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{s^2/\omega_z^2 + 2\zeta_z s/\omega_z + 1}{(J_m + J_g) \left[s^2/\omega_p^2 + 2\zeta_p s/\omega_p + 1 \right]} \quad (2.2)$$

where:

$$\omega_z = \sqrt{\frac{k_g}{J_g}}, \quad \omega_p = \sqrt{\frac{k_g(J_m + J_g)}{J_m J_g}}, \quad \zeta_z = \frac{d_g}{2} \sqrt{\frac{1}{J_g k_g}}, \quad \zeta_p = \frac{d_g}{2} \sqrt{\frac{J_m + J_g}{J_m J_g k_g}} \quad (2.3)$$

The input u is the motor torque (τ_m) and the output y is the motor acceleration ($\ddot{\varphi}_m$).

Figure 2.6 shows a snapshot of the state-space model corresponding to the dynamical model in (2.1).

Figure 2.7 shows the frequency response of the transfer function model in (2.2); with the following parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$ and $k_g = 5$. In Appendix B, frequency responses of three more 2-mass model versions used in the analysis are shown.

2.2.2 3-Mass System Dynamic Model of a Motor with Flexible Gearbox Drive Train and a Rigid Link

In cases where an inertia is situated between the gearbox output shaft and the flexible link, a 3-mass system is needed to describe the dynamics. This is, for example, the case for axis 1 of an anthropomorphic robot when the upper robot link is flexible and the lower robot link is rigid. Then the inertia of the lower robot link will introduce a large inertia between the gearbox output and the flexible upper link. A simplified mass-spring-damper model of a flexible robotic manipulator consisting of a drive train of a motor (torque T_m , inertia J_m), a flexible gearbox (stiffness k_g , damping d_g), an inertia on its output (J_g) and a flexible link (stiffness k_l , damping d_l , inertia J_l) is shown in Figure 2.8.

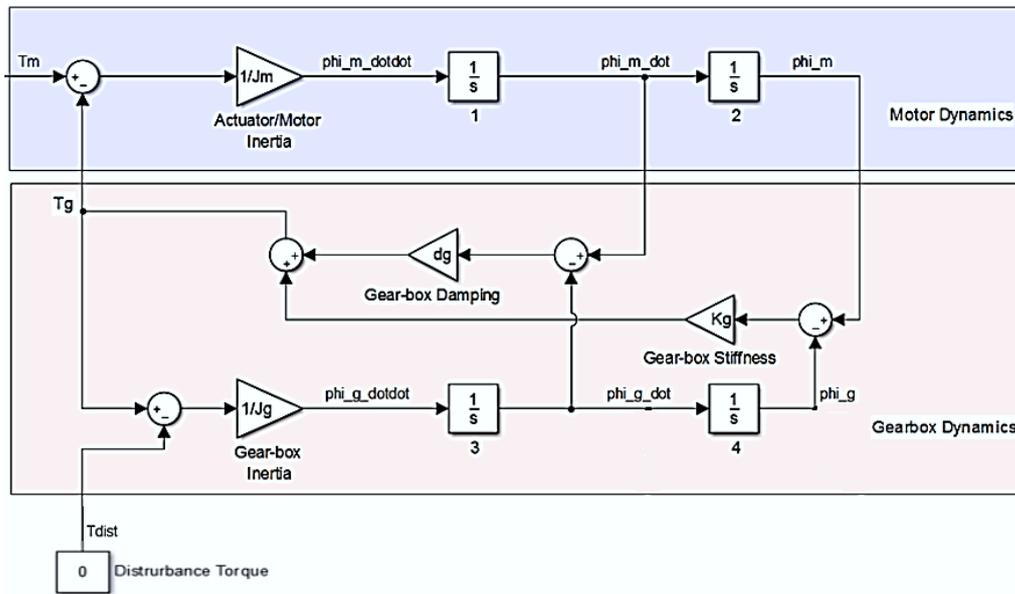


Figure 2.6: State-space model for a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox.

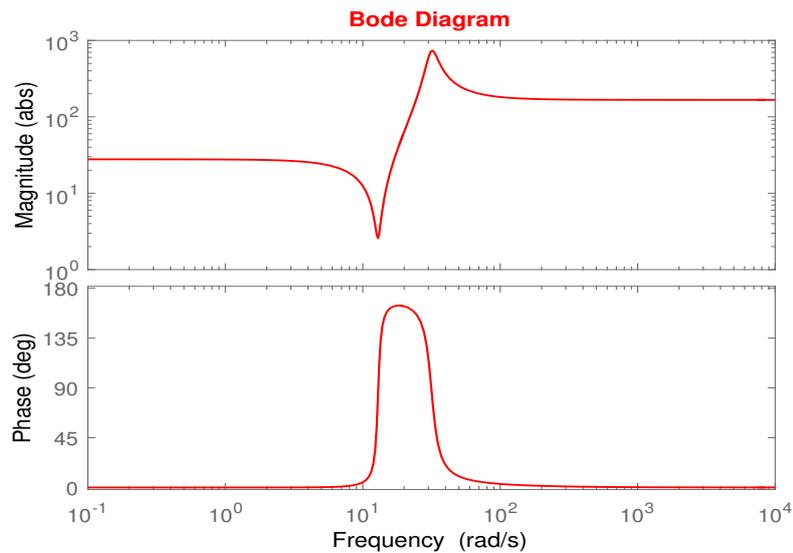


Figure 2.7: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox (2-mass system).

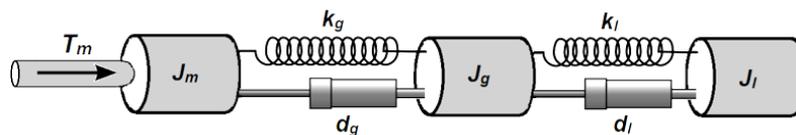


Figure 2.8: Mass-spring-damper model for a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and a flexible link.

The corresponding equations of motion resulting from applying Newton's second law (in differential equations form) are:

$$\begin{aligned}
 0 &= J_l \ddot{\varphi}_g + d_l(\dot{\varphi}_l - \dot{\varphi}_g) + k_l(\varphi_l - \varphi_g) \\
 0 &= J_m \ddot{\varphi}_m + d_g(\dot{\varphi}_g - \dot{\varphi}_m) + k_g(\varphi_g - \varphi_m) \\
 &\quad - d_g(\dot{\varphi}_l - \dot{\varphi}_g) - k_g(\varphi_l - \varphi_g) \\
 \tau_m &= J_m \ddot{\varphi}_m - d_g(\dot{\varphi}_g - \dot{\varphi}_m) - k_g(\varphi_g - \varphi_m)
 \end{aligned} \tag{2.4}$$

where $(\varphi_m, \varphi_g, J_m, J_g, k_g$ and $d_g)$ are as previous, in addition to:

φ_l : Link angular displacement (*rad*).

$\dot{\varphi}_l$: Link angular speed (*rad/s*).

$\ddot{\varphi}_l$: Link angular acceleration (*rad/s²*).

J_l : Link Inertia (*kg.m²*).

k_l : Link stiffness (*N/m*).

d_l : Link damping (*N/m²*).

The above set of motion equations can be described by the following transfer function:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\left[\begin{aligned} &J_g J_l s^4 + (J_g d_l + J_l d_g + J_l d_l) s^3 \\ &+ (J_g k_l + J_l k_g + J_l k_l + d_g d_l) s^2 \\ &+ (k_g d_l + k_l d_g) s + k_g k_l \end{aligned} \right]}{\left[\begin{aligned} &J_m J_g J_l s^4 + (J_m J_g d_l + J_m J_l d_g + J_m J_l d_l + J_g J_l d_g) s^3 \\ &+ (J_m J_g k_l + J_m J_l k_g + J_m J_l k_l + J_m d_g d_l + J_g J_l k_g + J_g d_g d_l + J_l d_g d_l) s^2 \\ &+ (J_m k_g d_l + J_m k_l d_g + J_g k_g d_l + J_g k_l d_g + J_l k_g d_l + J_l k_l d_g) s \\ &+ J_m k_g k_l + J_g k_g k_l + J_l k_g k_l \end{aligned} \right]} \tag{2.5}$$

where the following pairs of mode and anti-resonant frequencies can be obtained (using a symbolic solver in, for e.g. *Mathematica*, *Matlab* or *Maple*):

$$\omega_{z_1, z_2} = + \sqrt{\frac{\left[J_g k_l + J_l k_g + J_l k_l \right] \mp \sqrt{J_g^2 k_l^2 + J_l^2 k_l^2 + J_l^2 k_g^2 - 2J_g J_l k_g k_l + 2J_g J_l k_l^2 + 2J_l^2 k_g k_l}}{2J_g J_l}}$$

$$\omega_{p_1, p_2} = + \sqrt{\frac{\left[J_m J_g k_l + J_m J_l k_g \right] + J_m J_l k_l + J_g J_l k_g \mp \sqrt{J_m^2 J_g^2 k_l^2 + J_m^2 J_l^2 k_g^2 + J_m^2 J_l^2 k_l^2 + J_g^2 J_l^2 k_g^2 + 2J_m^2 J_l^2 k_g k_l + 2J_m^2 J_g J_l k_l^2 - 2J_m^2 J_g J_l k_g k_l - 2J_m^2 J_g^2 J_l k_g k_l - 2J_m J_g J_l^2 k_g k_l + 2J_m J_g J_l^2 k_g^2}}{2J_m J_g J_l}}$$
(2.6)

The input u is the motor torque (τ_m) and the output y is the motor acceleration ($\ddot{\psi}_m$).

Figure 2.9 shows a snapshot of the state-space model corresponding to the dynamical model in (2.4).

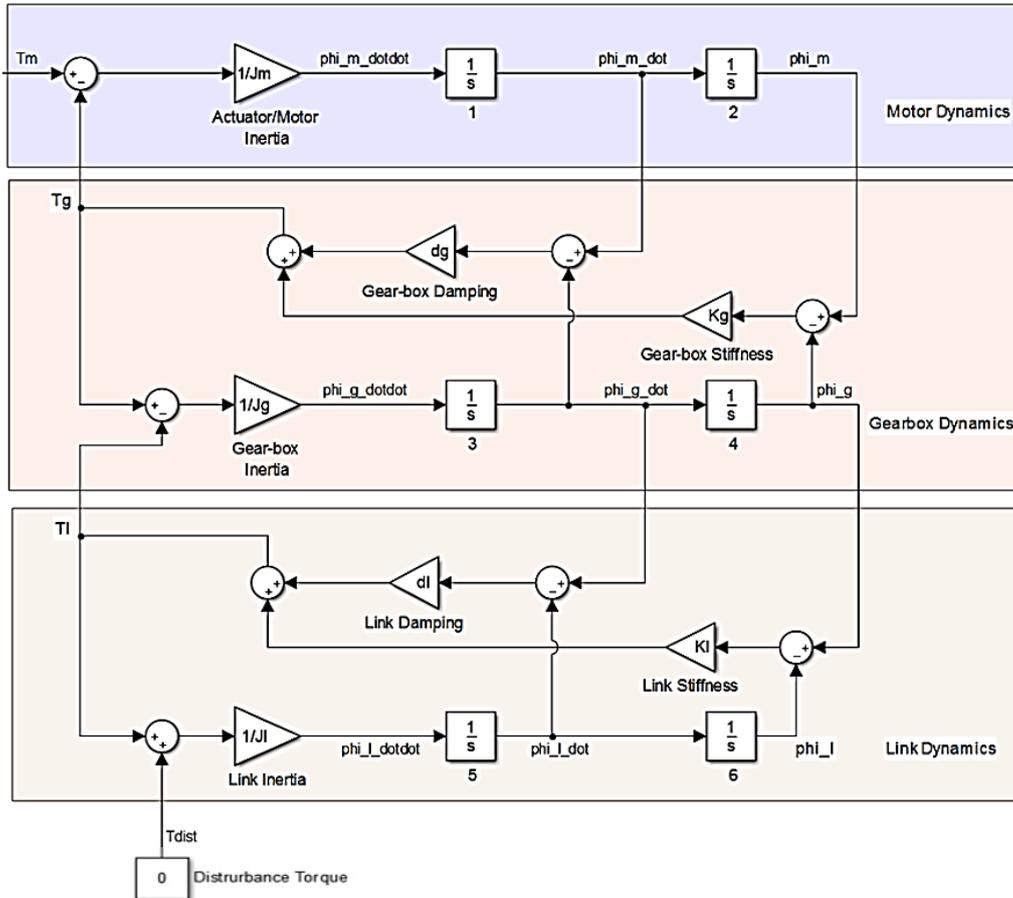


Figure 2.9: State-space model a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and a flexible link.

Figure 2.10 shows the frequency response of the transfer function model in (2.5);

with the following parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 5$, $J_l = 60 \times 10^{-3}$, $d_l = 30 \times 10^{-3}$ and $k_l = 5$. In Appendix C, frequency responses of other versions of the 3-mass system used in the analysis are shown.

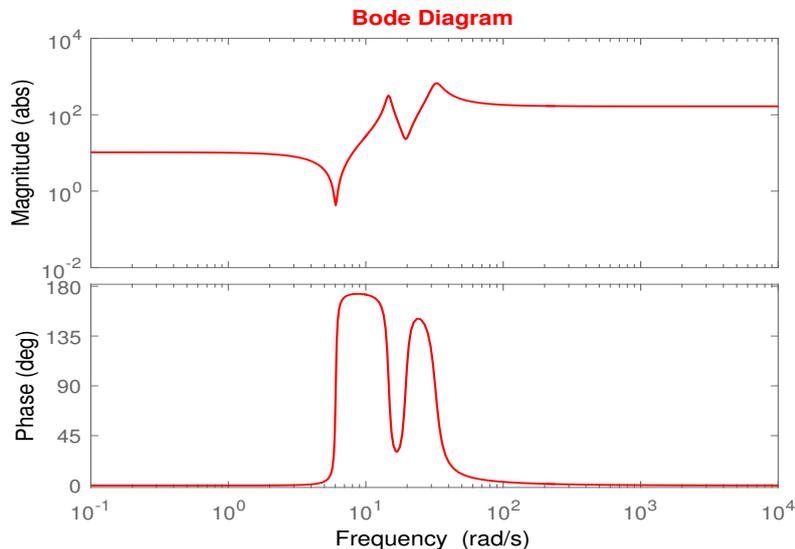


Figure 2.10: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and a flexible link (3-mass system).

2.2.3 Multi-Mass System Dynamic Model of a Motor with Flexible Gearbox Drive Train and Lumped Mass-Spring-Damper Model of a Flexible Link

According to [67], an equivalent dynamic equation of motion for a distributed flexible manipulator system link can be obtained using a lumped parameter system. Such a cascaded system consists of a chain of weightless linear angular springs separated by concentrated point-masses, and it is described by a set of ordinary differential equations. Furthermore, using enough number of lumped elements N , the frequency responses of the equivalent lumped parameter system approach the frequency response of the clamped-free continuously flexible assumed-mode method (AMM) model from the left and the calculated natural frequencies will be adequately close to experimental results. Thus the lumped model can be regarded as a good approximation for the real model when N increases.

Figure 2.11 shows a mass-spring-damper model of a flexible robotic manipulator consisting of a drive train of a motor (torque T_m , inertia J_m) and a flexible gearbox (stiffness k_g , damping d_g), an inertia on its output (J_g), together with a lumped (multi-mass) mass-spring-damper model of a flexible link (stiffness k_l , damping d_l , inertia J_l).

The corresponding numerical values for the equivalent lumped parameter model can be calculated by using the following set of derivations and equations:

2. Modelling of Flexible Robotic Manipulators

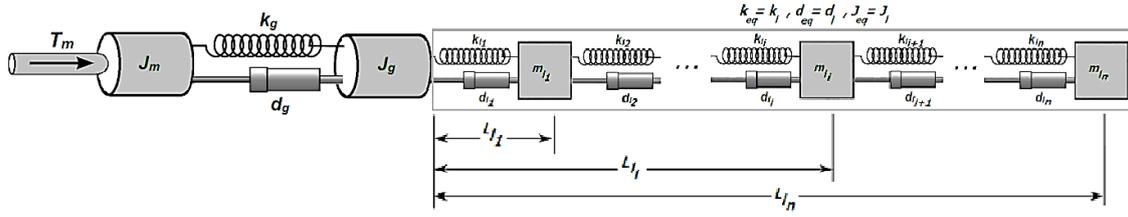


Figure 2.11: Mass-spring-damper model for a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link.

Assuming that there are n_e lumped masses (each has a uniform mass m_i , stiffness k_i and damping d_i) in the flexible link that is approximated by a beam of a total length L_{beam} [m], total mass M_{beam} [kg] and inertia J_{beam} [$kg.m^2$], total stiffness k_{beam} [N/m] and total damping d_{beam} [$N.s/m$], then:

$$\begin{aligned}
 m_i &= \left(\frac{1}{n_e} \times M_{beam}\right) \text{ [kg]}; \quad i = 1, \dots, n_e \\
 J_{a_1} &= m_i \times \left(1 \times \frac{L_{beam}}{n_e}\right)^2 \\
 J_{a_2} &= m_i \times \left(2 \times \frac{L_{beam}}{n_e}\right)^2 \\
 &\vdots \\
 J_{a_{n_e}} &= m_i \times \left(n_e \times \frac{L_{beam}}{n_e}\right)^2
 \end{aligned} \tag{2.7}$$

And hence the mass of each individual element (m_i) of the lumped link can be calculated by solving:

$$J_{beam}(\text{at the motor side}) = N^2 J_{beam}(\text{at the link side}) = \sum_{i=0}^{n_e} m_i \times \left(\frac{L_{beam}}{n_e} + i \times \frac{L_{beam}}{n_e}\right)^2 \tag{2.8}$$

In addition, the stiffness and damping of individual mass elements (k_i and d_i respectively) of the lumped link can be calculated via:

$$\begin{aligned}
 k_{beam}(\text{at the motor side}) &= N^2 k_{beam}(\text{at the link side}) = \frac{k_i}{n_e} \\
 \Rightarrow k_i &= n_e N^2 k_{beam}(\text{at the link side}) \\
 &= n_e k_{beam}(\text{at the motor side}) [N/m]; \quad i = 1, \dots, n_e \\
 \frac{1}{d_{beam}}(\text{at the motor side}) &= N^2 \times \frac{1}{d_{beam}}(\text{at the link side}) = \frac{n_e}{d_i} \\
 \Rightarrow d_i &= n_e N^2 d_{beam}(\text{at the link side}) \\
 &= n_e d_{beam}(\text{at the motor side}) [N.s/m]; \quad i = 1, \dots, n_e
 \end{aligned}
 \tag{2.9}$$

where N is the gear ratio between the low-speed motor side and the high-speed flexible link side.

Figure 2.12 shows a snapshot of the state-space model corresponding to the model shown in Figure 2.11, followed a detailed state-space representation of the lumped link model dynamics shown in Figure 2.13.

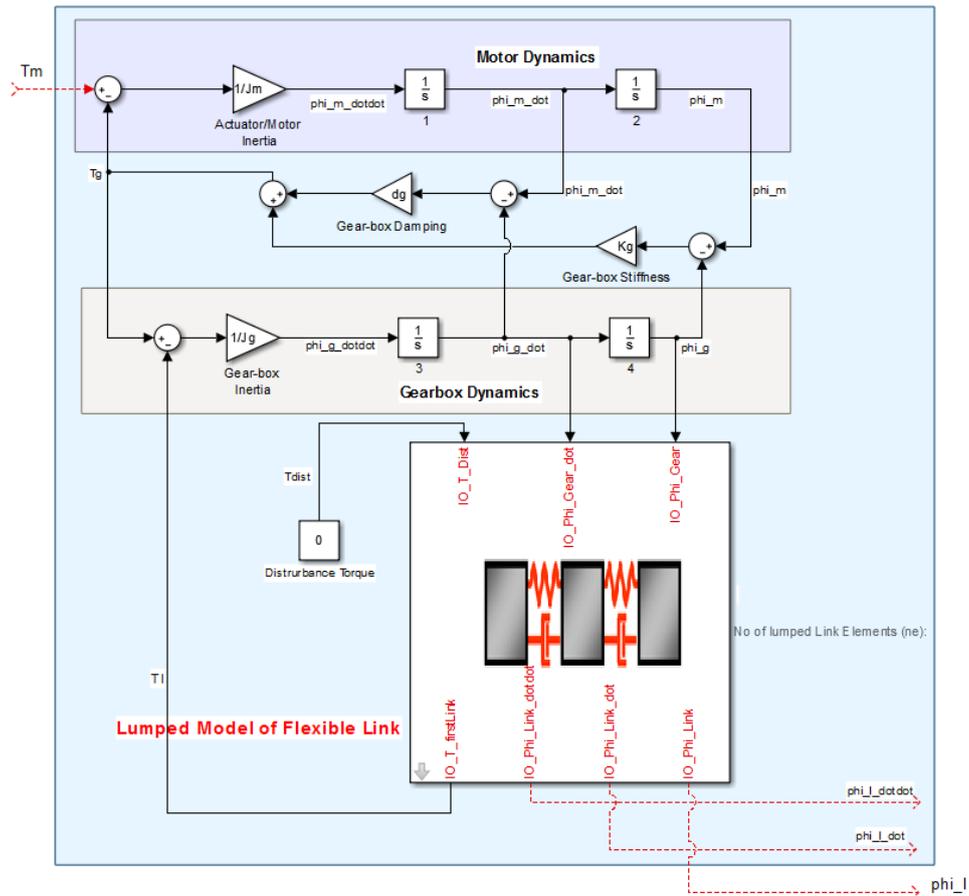


Figure 2.12: State-space model for a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link.

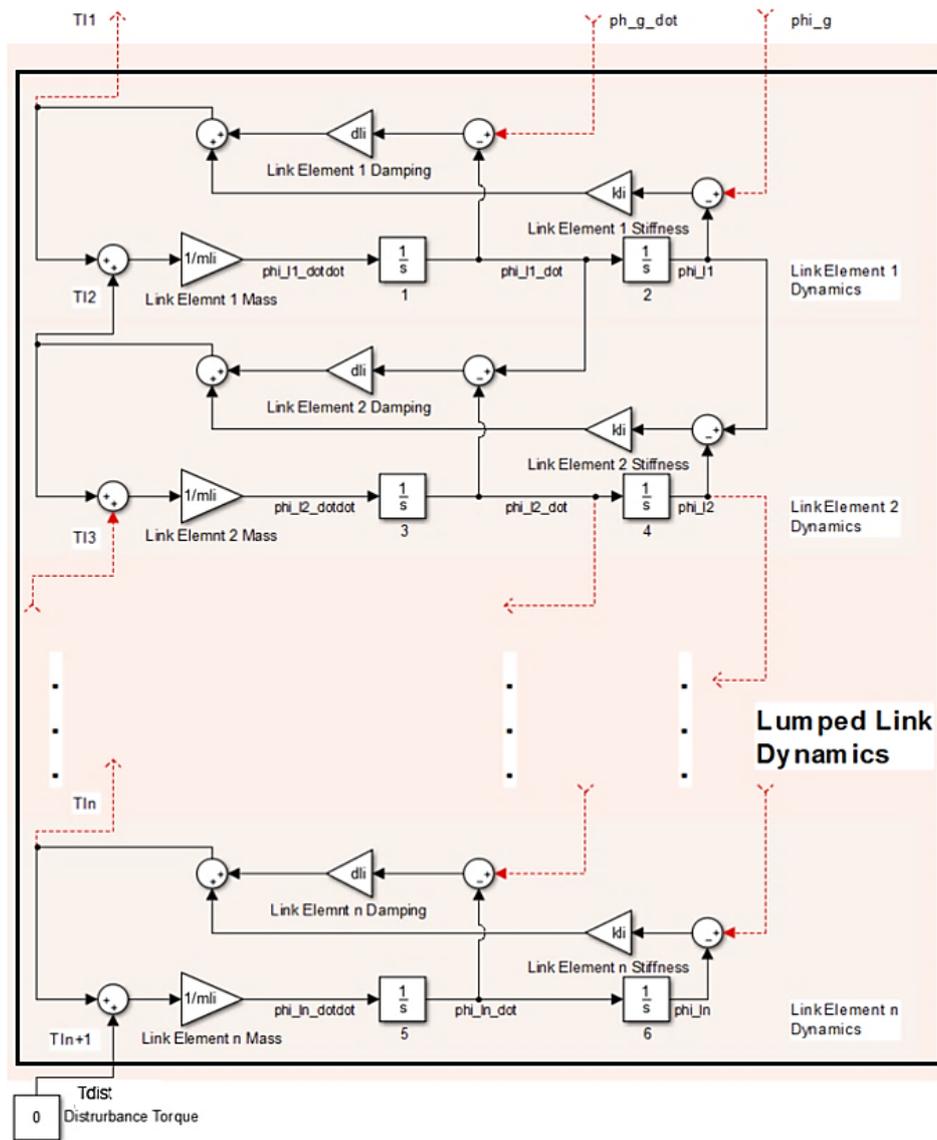


Figure 2.13: State-space model for a lumped (multi-mass) model of a flexible link (with n lumped elements).

To obtain a very realistic simulation of the first axis motion of a real robot (a 2-mass system) and also its upper link (a 3-mass system), a tip-mass load with a big inertia around the base first and then the first axis of motion is consequently added to the end of the lumped link ($M_{load} = 250 \text{ kg}$).

Figure 2.14 shows the identified (using a chirp input signal) frequency response of the resulting 2-mass system; with the following set of discrete model parameters: $J_m = 6 \times 10^{-3}$, $J_g = 45.952 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 5$, $J_l = 11.376 \times 10^{-3}$, $d_l = 100 \times 10^{-3}$, $k_l = 5$ and $n = 15$.

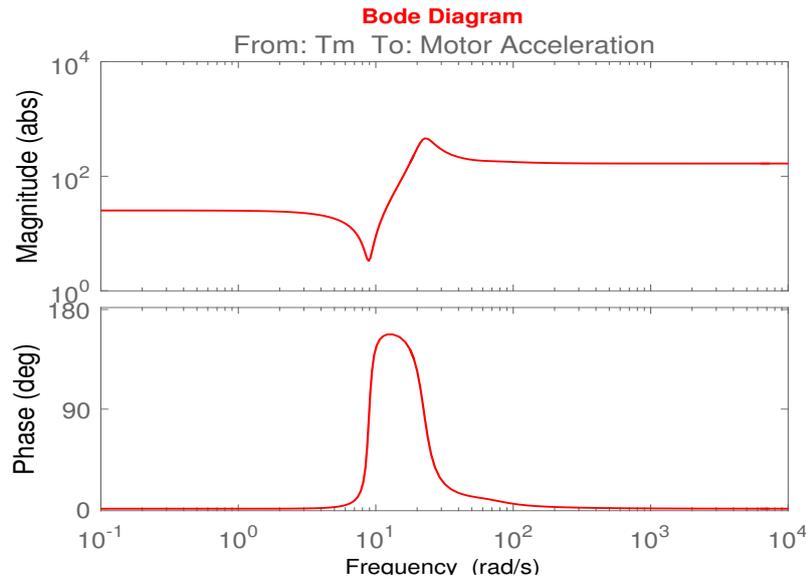


Figure 2.14: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (2-mass system case).

On the other hand, Figure 2.15 shows the identified (using a chirp input signal) frequency response of the resulting 3-mass system; with the following set of discrete model parameters: $J_m = 6 \times 10^{-3}$, $J_g = 10.2 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 5$, $J_l = 11.376 \times 10^{-3}$, $d_l = 30 \times 10^{-3}$, $k_l = 5$ and $n = 15$.

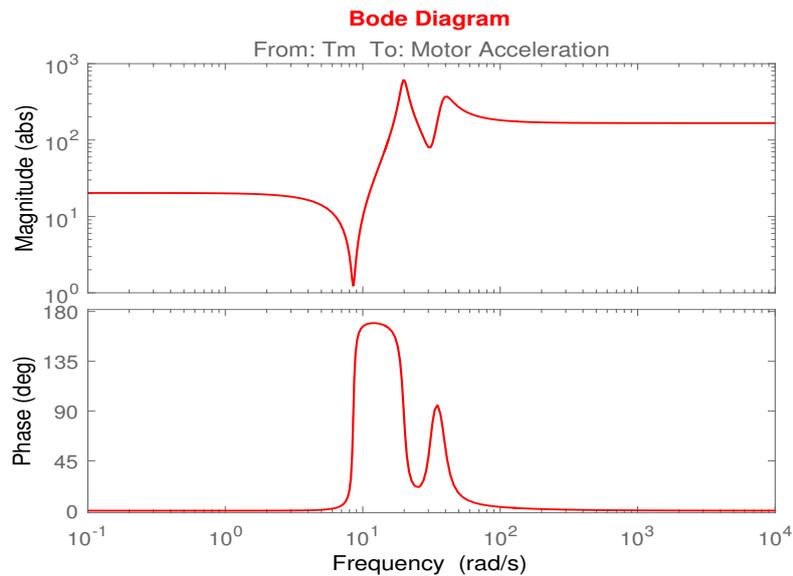


Figure 2.15: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case).

In Appendix D, frequency responses of other versions of the multi-mass (3-mass) systems used in the analysis are shown, together with an illustrative comparison explaining the effect of the number of lumped elements (in the multi-mass link model) on the frequency response.

2.2.4 ADAMS Models of Gearbox Drive Train with FEM Model of a Flexible Link

In order to make it possible to understand the problems with flexible link control in a more realistic set-up, a simplified robot arm was designed with a FEM model in the ADAMS software package. The model corresponds to a flexible upper link of a large robot handling 250 *kg* load. The link is connected to a gearbox, which is simulated in Simulink. The main bending mode of the ADAMS model is exported to Simulink and all the model analysis is then made in Simulink. The ADAMS model consists of a flexible link, a rigid mass mounted on the tip of the link and a rigid mass mounted on the output of the gearbox shaft. The flexible link is mounted on the rigid joint mass and this mass can be designed to obtain a 3-mass system, which at low mass approaches a 2-mass system.

Figure 2.16 shows an ADAMS model for a flexible manipulator link model integrating an ADAMS FEM model for the flexible link with a relatively small inertia on the output of the gearbox, which will result in a 2-mass system.

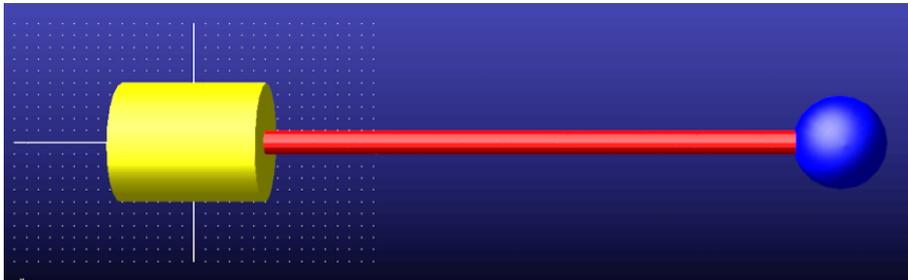


Figure 2.16: ADAMS model for a flexible manipulator gearbox with a relatively small inertia on the output of the gearbox and FEM model of a flexible link with a load at its end (2-mass system case).

Figure 2.17 shows an ADAMS model for a flexible manipulator model integrating an ADAMS FEM model for the flexible link with a large inertia on the output of the gearbox, which will give rise to a 3-mass system. This may be the case for axis 1 of an industrial robot.

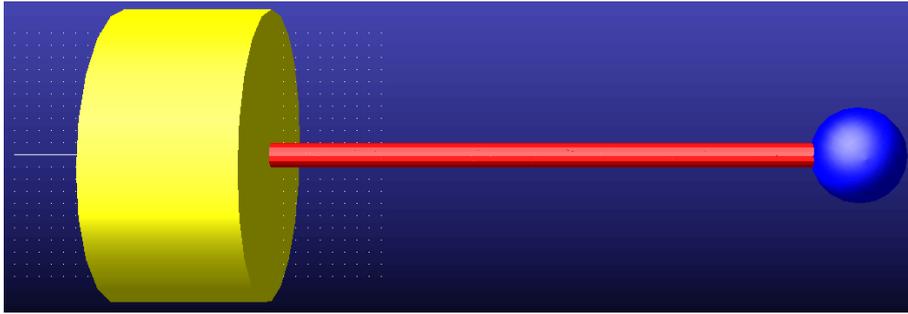


Figure 2.17: ADAMS model for a flexible manipulator gearbox with a large inertia on the output of the gearbox and FEM model of a flexible link with a load at its end (3-mass system case).

Figure 2.18 shows a snapshot of the state-space model corresponding to a flexible manipulator integrating an ADAMS FEM model for the flexible link.

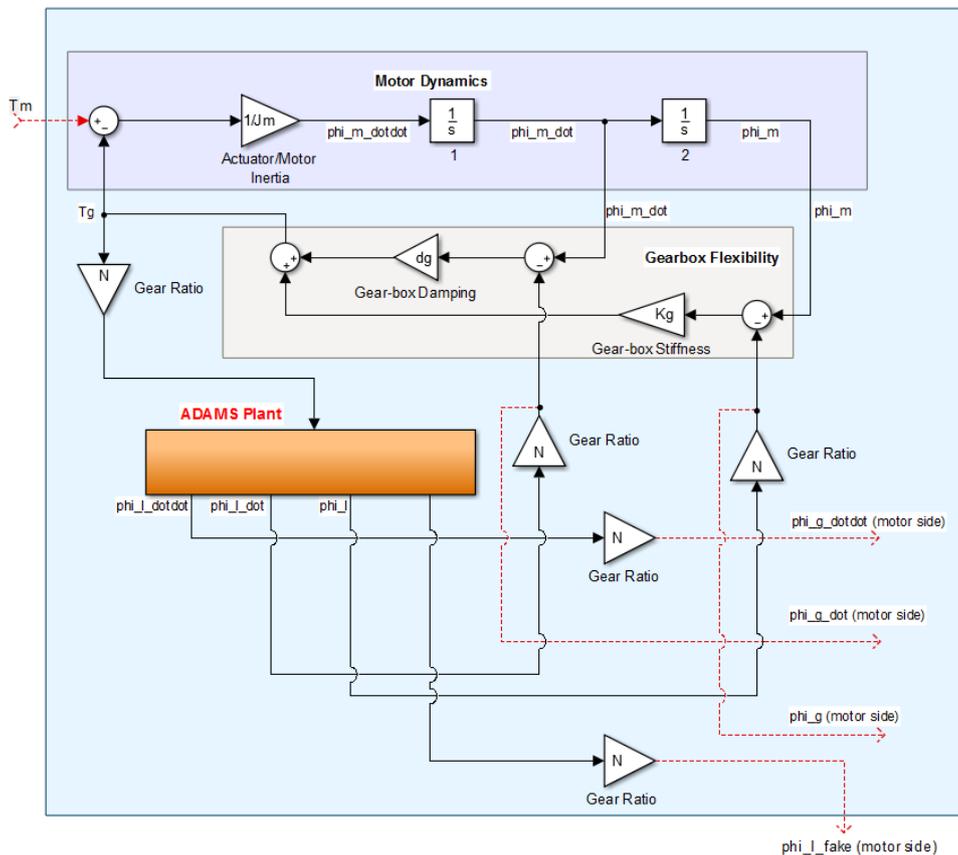


Figure 2.18: State-space model for a flexible manipulator integrating a motor, and gearbox model in Simulink with an ADAMS FEM model of an inertia on the output of the gearbox and a flexible link.

Figure 2.19 shows the identified (using a chirp input signal) frequency response of the state-space model shown in Figure 2.18 using the ADAMS model in Figure 2.16 (2-mass model).

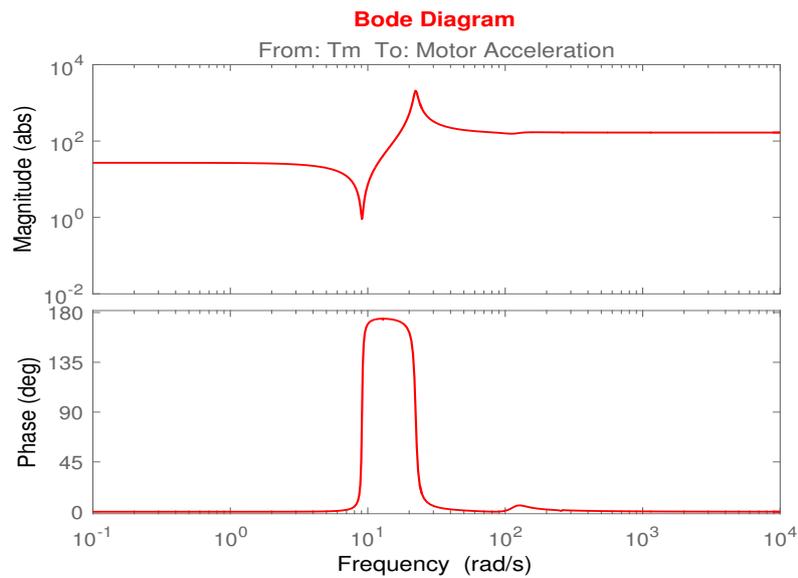


Figure 2.19: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of 2-mass system resulting from a motor and a gearbox flexibility driving an ADAMS model with low inertia on gearbox output and FEM model of a flexible link.

On the other hand, Figure 2.20 shows the identified (using a chirp input signal) frequency response of the state-space model shown in Figure 2.18 using the ADAMS model in Figure 2.17 (3-mass model).

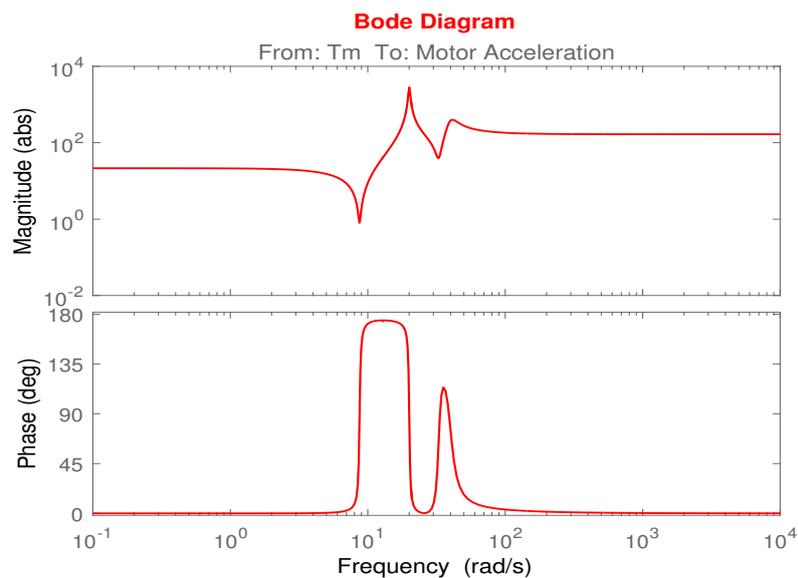


Figure 2.20: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a 3-mass system resulting from a motor and gearbox flexibility driving ADAMS model with large inertia on the output of the gearbox and FEM model of a flexible link.

2.3 Modes and Anti-Modes Matching in the Frequency Plane

In order to avoid using the inverse of the complex FEM ADAMS model for feed-forward control (it takes relatively longer time to simulate even simple cases), the idea is to use the inverse of a model with lumped rigid masses having the same frequency response as the ADAMS model. This means that given the frequency response of the FEM model, a method must be developed to match the zeros and poles of the simpler lumped-mass model with the zeros and poles of the ADAMS model. This frequency response matching is made in the following way:

- The frequency response of the ADAMS model is obtained after exciting the ADAMS plant system with a swept-frequency sinusoidal (chirp) signal and recording both gain and phase-shift for each pair of amplitude and frequency of the sinusoidal input signal while sweeping. Thus, an approximate estimation of the plant's frequency response can be obtained. Even though the frequency response can be obtained directly from the ADAMS model, such frequency response identification experiments are very useful as they are necessary to be carried out, in practice, on the real manipulator.
- The zeros and poles are found from the frequency response (Bode diagram).
- A lumped model with the same zeros and poles is determined. This model may either be a 2-mass model or a 3-mass model.

In this section it is shown by means of frequency responses that it is possible to make accurate matching of lumped models to FEM models. The methodology followed to achieve this goal is also described. When the matched lumped model is obtained, then the inverse of this model can be derived and used for feed-forward control. Matching of simple 2-mass and 3-mass models is provided first in order to build the practical experience before diving into the more complex cases. The matching procedure and the precise steps for mode/anti-mode matching in the frequency plane using 2-mass and 3-mass inverse dynamic models are described in details in Appendix E.

2.3.1 Frequency Response Matching of a 2-Mass Plant System Using an Identical 2-Mass Inverse Dynamics Model

Figure 2.21 portrays the results of the pair of mode and anti-mode frequency matching process of the 2-mass plant system shown in Figure 2.7 by an identical 2-mass inverse dynamic model.

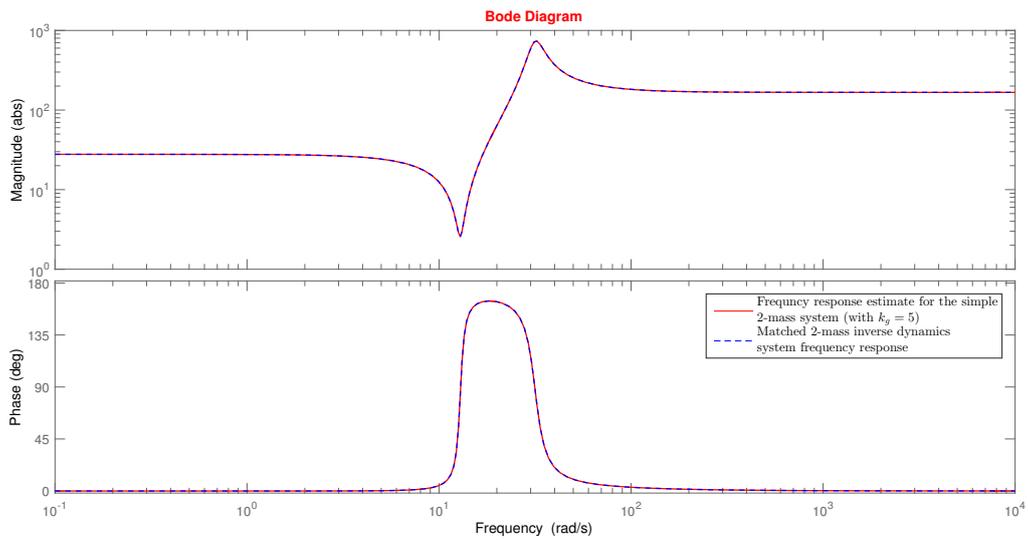


Figure 2.21: Frequency response matching of a 2-mass plant system by an identical 2-mass inverse dynamic model.

2.3.2 Frequency Response Matching of a 3-Mass Plant System

2.3.2.1 Using a 2-Mass Inverse Dynamics Model

Figure 2.22 portrays the results of lower pair of mode and anti-mode frequency matching process of the 3-mass plant system shown in Figure 2.10 by a 2-mass inverse dynamic model.

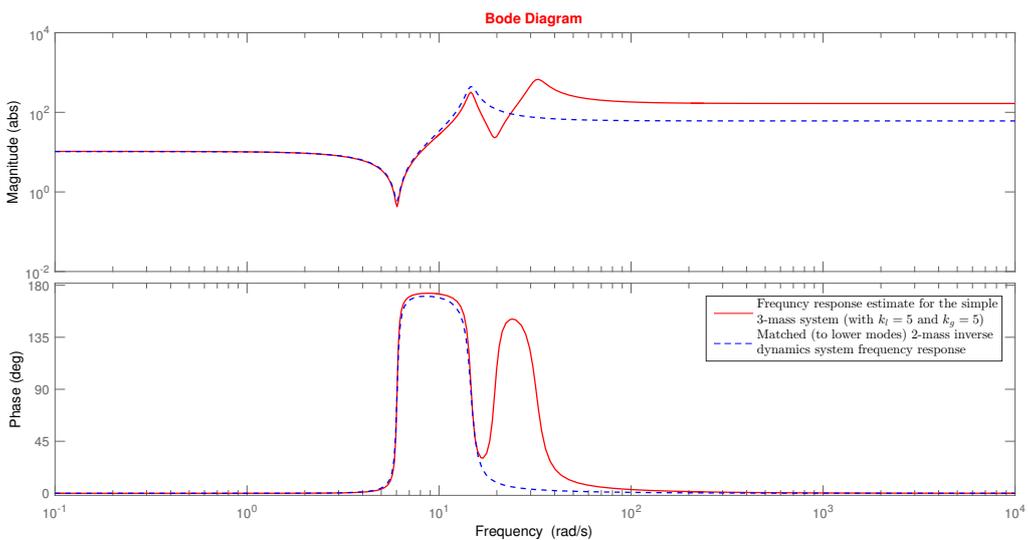


Figure 2.22: Frequency response matching of a 3-mass plant system by a 2-mass inverse dynamic model (lower pair of mode and anti-mode matching).

Figure 2.23 portrays the results of terminal pair of mode and anti-mode frequency matching process of the 3-mass plant system shown in Figure 2.10 by a 2-mass inverse dynamic model.

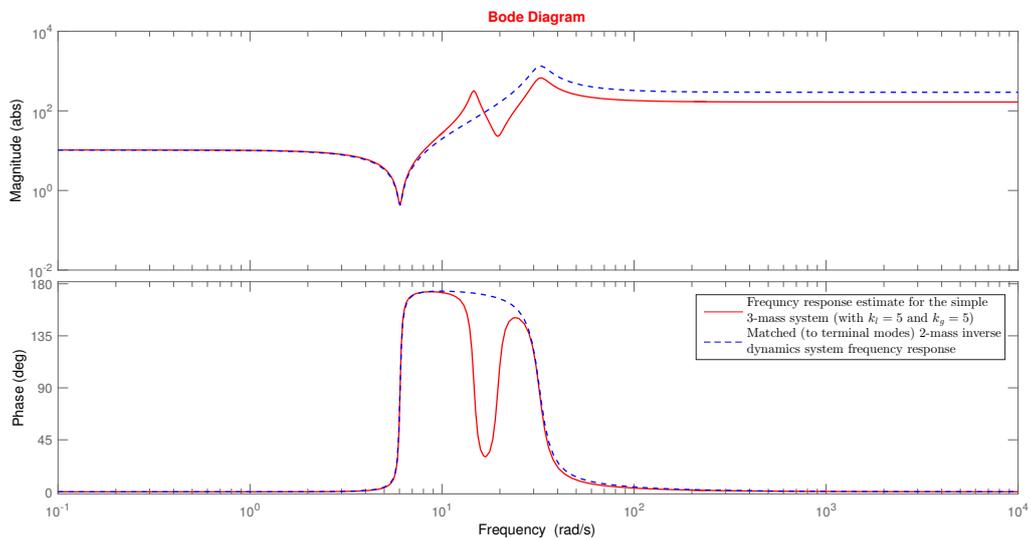


Figure 2.23: Frequency response matching of a 3-mass plant system by a 2-mass inverse dynamic model (terminal pair of mode and anti-mode matching).

Figure 2.24 portrays the results of higher pair of mode and anti-mode frequency matching process of the 3-mass plant system shown in Figure 2.10 by a 2-mass inverse dynamic model.

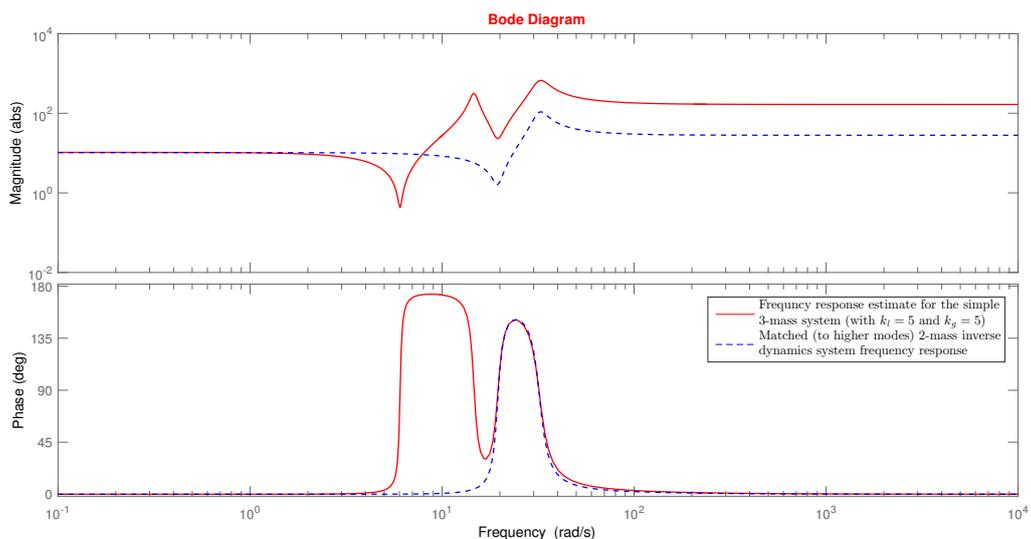


Figure 2.24: Frequency response matching of a 3-mass plant system by a 2-mass inverse dynamic model (higher pair of mode and anti-mode matching).

2.3.2.2 Using an Identical 3-Mass Inverse Dynamics Model

Figure 2.25 portrays the results of the pair of mode and anti-mode frequency matching process of the 3-mass plant system shown in Figure 2.10 by an identical 3-mass inverse dynamic model.

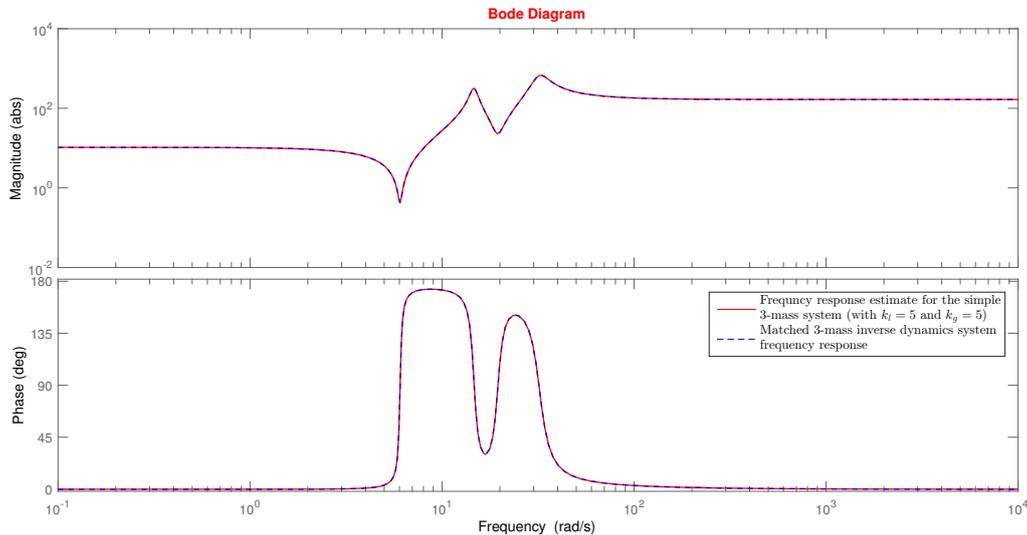


Figure 2.25: Frequency response matching of a 2-mass plant system by an identical 2-mass inverse dynamic model.

2.3.3 Frequency Response Matching of Multi-Mass Plant Systems

2.3.3.1 A 2-Mass Plant System Case

Figure 2.26 shows the results of the pair of mode and anti-mode frequency matching process of the 2-mass ADAMS plant system frequency response shown in Figure 2.19 by a multi-mass inverse dynamic model (2-mass system).

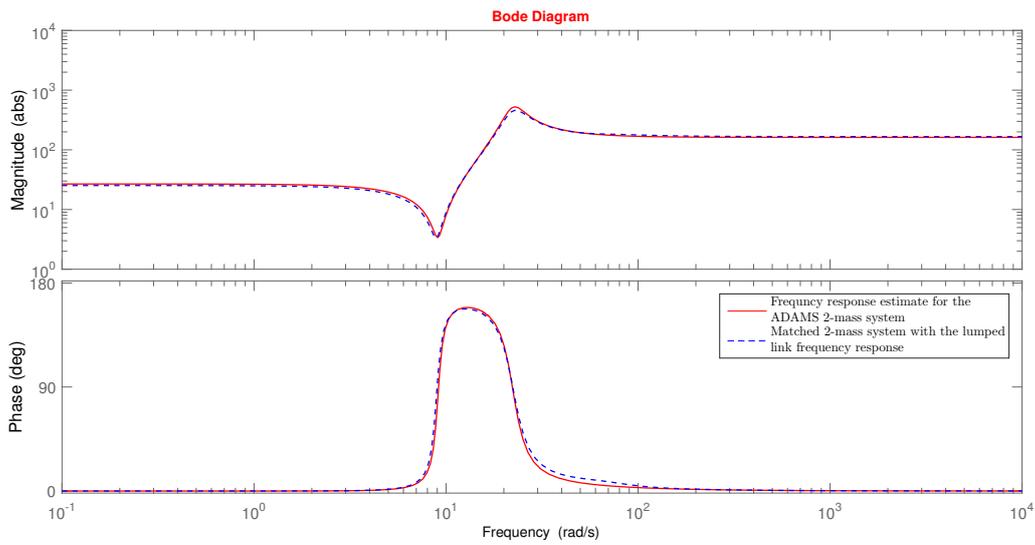


Figure 2.26: Frequency response matching of an ADAMS 2-mass plant system by a multi-mass inverse dynamic model (a 2-mass system).

Using a 2-Mass Inverse Dynamics Model

Figure 2.27 portrays the results of the pair of mode and anti-mode frequency matching process of the 2-mass plant system shown in Figure 2.14 by an identical 2-mass inverse dynamic model.

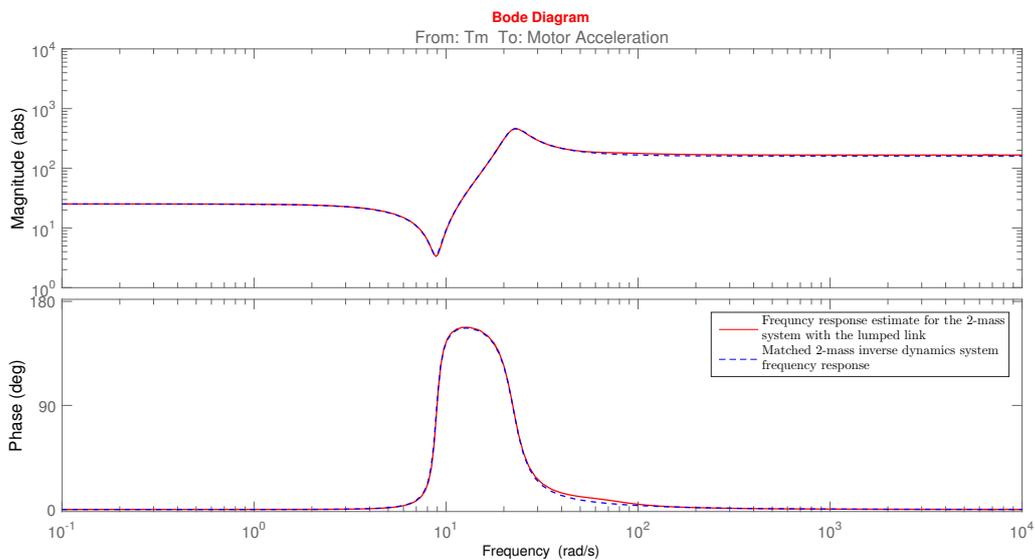


Figure 2.27: Frequency response matching of a multi-mass plant system (2-mass system) by an identical 2-mass inverse dynamic model.

2.3.3.2 A 3-Mass Plant System Case

Figure 2.28 shows the results of the pair of mode and anti-mode frequency matching process of the 3-mass ADAMS plant system frequency response shown in Figure 2.19 by a multi-mass inverse dynamic model (3-mass system).

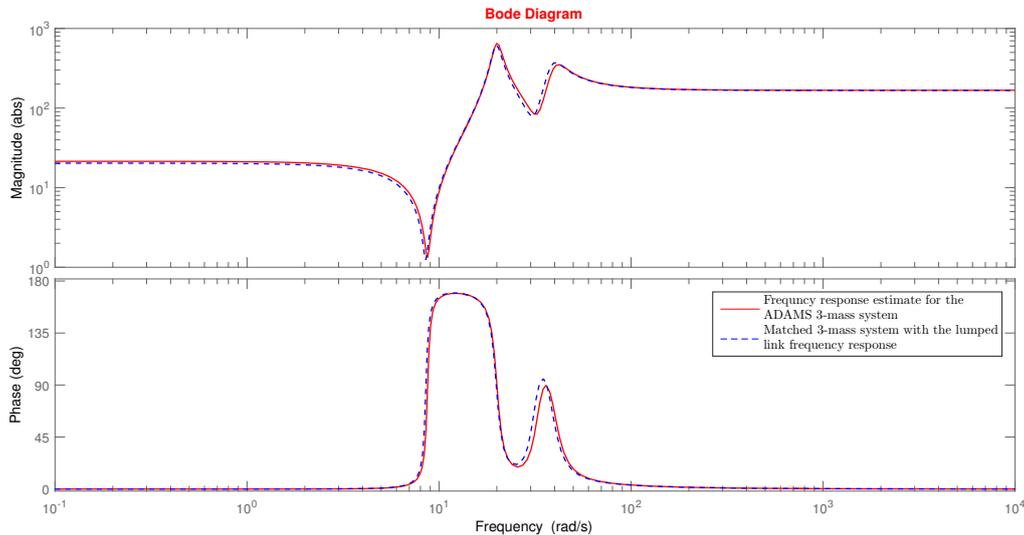


Figure 2.28: Frequency response matching of an ADAMS 3-mass plant system by a multi-mass inverse dynamic model (a 3-mass system).

Using a 2-Mass Inverse Dynamics Model (lower pair of mode and anti-mode matching)

Figure 2.29 portrays the results of lower pair of mode and anti-mode frequency matching process of the 3-mass plant system shown in Figure 2.15 by a 2-mass inverse dynamic model.

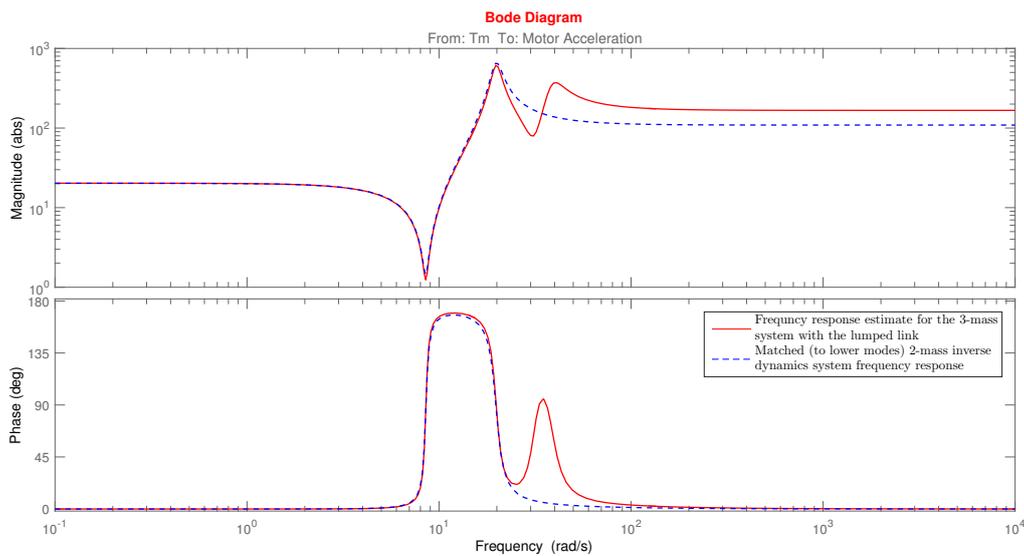


Figure 2.29: Frequency response matching of a multi-mass plant system (3-mass system) by a 2-mass inverse dynamic model (lower pair of mode and anti-mode matching).

Using a 3-Mass Inverse Dynamics Model

Figure 2.30 portrays the results of the pair of mode and anti-mode frequency matching process of the 3-mass plant system shown in Figure 2.10 by an identical 3-mass inverse dynamic model.

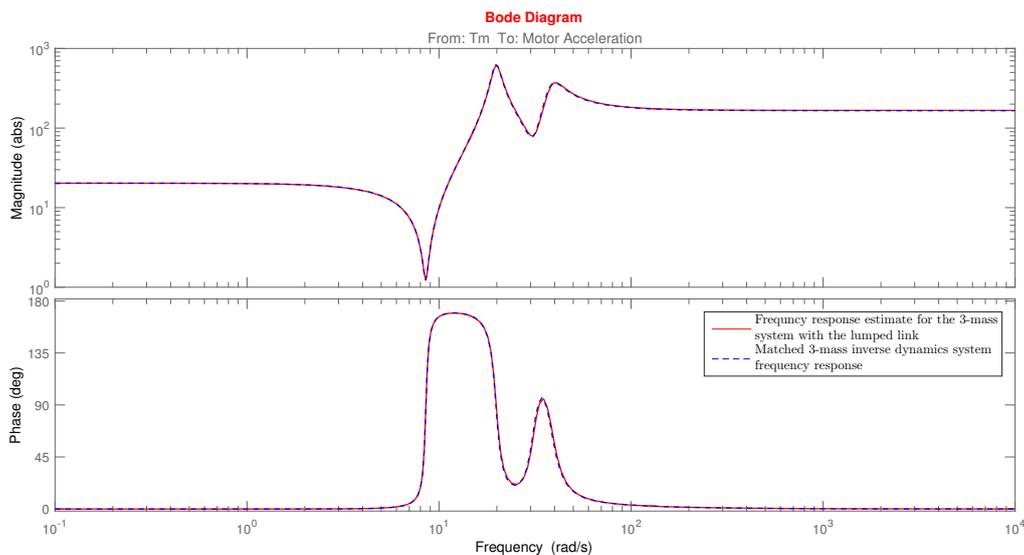


Figure 2.30: Frequency response matching of a multi-mass plant system (3-mass system) by an identical 3-mass inverse dynamic model.

2.3.4 Frequency Response Matching of ADAMS Plant Systems

2.3.4.1 A 2-Mass Plant System Case

Using a 2-Mass Inverse Dynamics Model

Figure 2.31 portrays the results of the pair of mode and anti-mode frequency matching process of the 2-mass plant system shown in Figure 2.19 by an identical 2-mass inverse dynamic model.

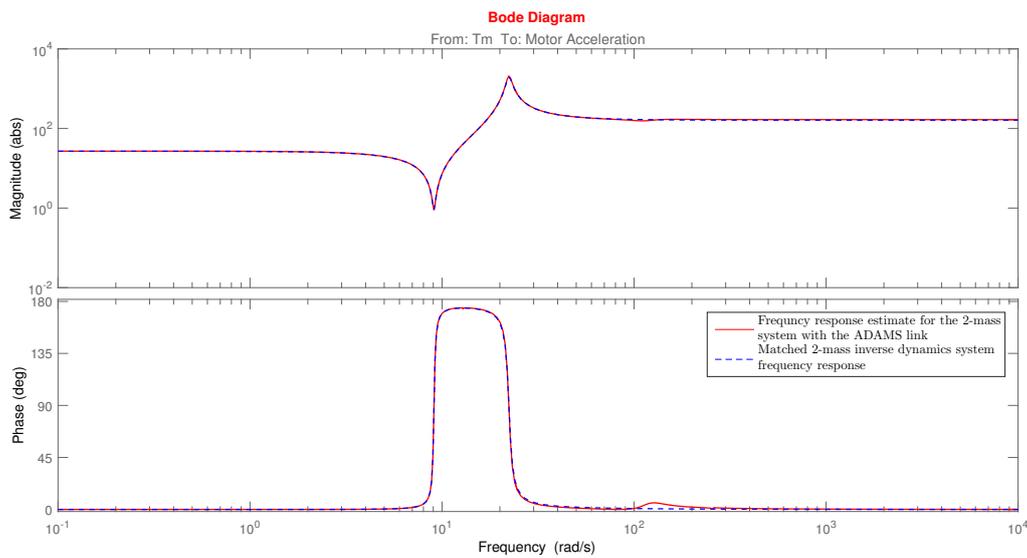


Figure 2.31: Frequency response matching of an ADAMS plant system (2-mass system) by an identical 2-mass inverse dynamic model.

2.3.4.2 A 3-Mass Plant System Case

Using a 2-Mass Inverse Dynamics Model (lower pair of mode and anti-mode matching)

Figure 2.32 portrays the results of the pair of mode and anti-mode frequency matching process of the 2-mass plant system shown in Figure 2.20 by an identical 2-mass inverse dynamic model.

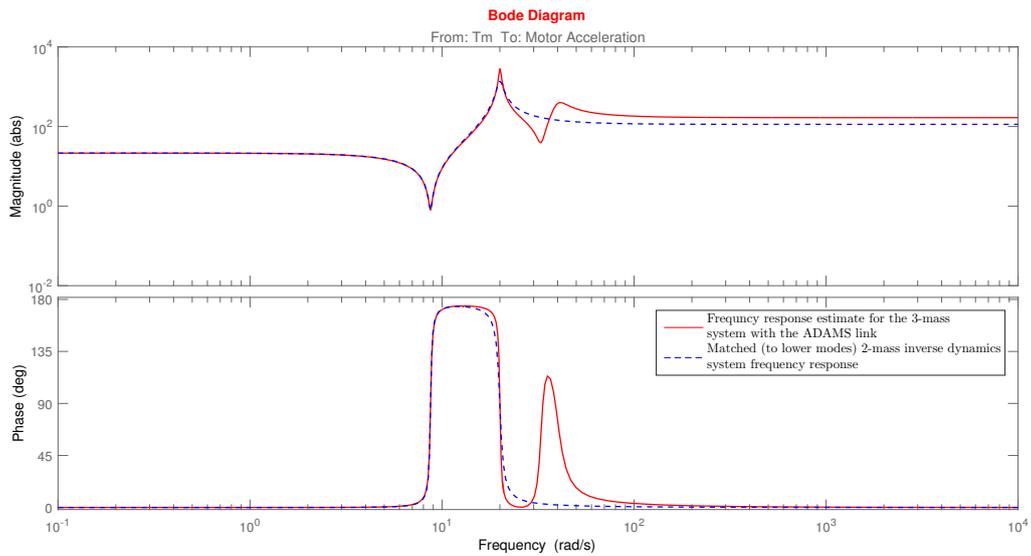


Figure 2.32: Frequency response matching of an ADAMS plant system (2-mass system) by a 2-mass inverse dynamic model (lower pair of mode and anti-mode matching).

Using a 3-Mass Inverse Dynamics Model

Figure 2.33 portrays the results of the pair of mode and anti-mode frequency matching process of the 2-mass plant system shown in Figure 2.20 by an identical 2-mass inverse dynamic model.

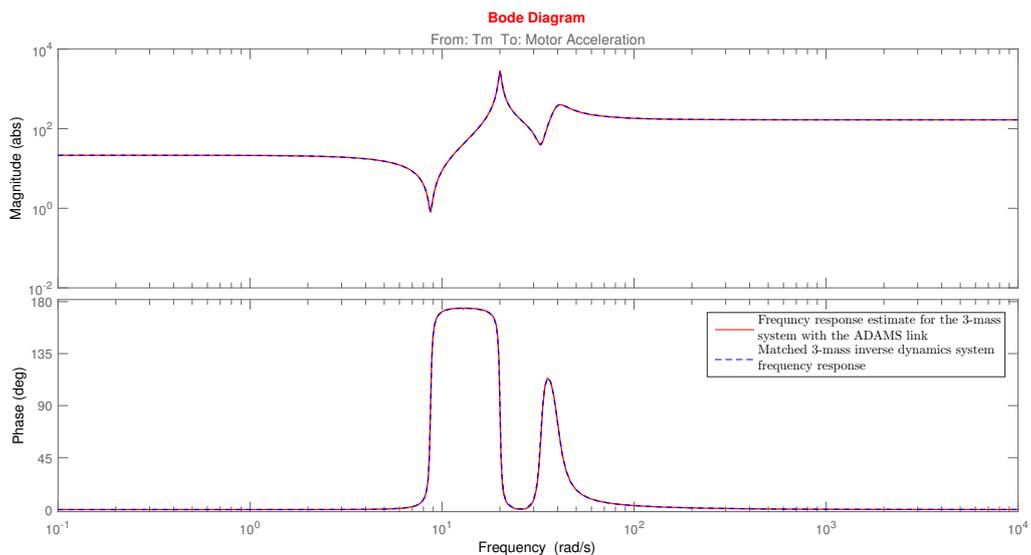


Figure 2.33: Frequency response matching of an ADAMS plant system (3-mass system) by an identical 3-mass inverse dynamic model.

3

Problem Statement, Control System Design and Compensation Techniques

As mentioned earlier, the goal of this thesis work is to accurately track a prescribed angular position reference trajectory profile by the end-effector's tip of an elastic robotic arm (with both a flexible joint and a flexible link). To achieve this task, we need to develop a realistic model for the elastic arm (lumped multi-mass-spring-damper and FEM models are used to capture the interesting link's "bending" flexibility). The frequency response of this "nonlinear" model is then identified and inverted using a feed-forward computed torque method of a 4th-order (2-mass inverse dynamics) or a 6th-order (3-mass inverse dynamics). The result of this input-output linearisation process is a linear system that can be easily controlled by a simple PID feedback controller. Additionally, a smooth position reference trajectory that can be differentiated up to the required order of the inverse dynamic model should be generated. Due to the difficulties encountered as a result of noisy derivatives in the inverse dynamic models, some smoothing is needed to avoid running into numerical problems, hence some smoothing filters are utilised. Furthermore, a scheme for compensation of the residual tracking errors (for position and speed), that arise because of the model-mismatch with the real plant, is introduced and implemented.

In this chapter, the trajectory tracking problem for the manipulator's end-effector is formally introduced. Beside that, a thorough discussion regarding the different issues related to smooth reference trajectory generation is given. The theory and motivation behind the PID controller variant adopted for feedback control in this thesis work are also introduced, in addition to the computed torque scheme used for feed-forward compensation with all filtering needed to avoid numerical problems. At the end, compensation of dynamical residual errors are also discussed in details studying various flexible manipulator designs that are mostly encountered in practice.

3.1 Problem Statement

The scheme of a two-link planar robotic manipulator arm (in a horizontal plane) is shown in Figure 3.1. The manipulator arm is actuated by two motors (with

their gearbox drive-trains) and have the ability to rotate, by construction, in the horizontal plane. Each of these two motors resides at one of the flexible joints (A or B). For a robot arm with two flexible links, the control problem is to find the input torques (τ_{m_1}, τ_{m_2}) that guarantee transferring the load mass at the tip of the arm (the manipulator's end-effector) from its initial configuration

$$\varphi_{l_1}(0) = \varphi_{l_1}^0, \quad \varphi_{l_2}(0) = \varphi_{l_2}^0, \quad \dot{\varphi}_{l_1}(0) = \dot{\varphi}_{l_2}(0) = 0 \quad (3.1)$$

to any admissible arbitrary final configuration

$$\varphi_{l_1}(T) = \varphi_{l_1}^{ref}, \quad \varphi_{l_2}(T) = \varphi_{l_2}^{ref}, \quad \dot{\varphi}_{l_1}^{ref}(T) = \dot{\varphi}_{l_2}^{ref}(T) = 0 \quad (3.2)$$

and keep it in this final configuration until new desired trajectory settings are executed. Here, $\varphi_{l_1}^0$ is the first link's end-point initial angle (usually it starts from rest at 0), $\varphi_{l_2}^0$ is the initial interlink angle and $\varphi_{l_1}^{ref}$ and $\varphi_{l_2}^{ref}$ are the desired angle rotations of the link; where as T is the motion time that is not given but necessarily as short as possible for this class of manipulators' tasks (and hence excitation of flexible modes). Additionally, initial and final elastic deformations are desirably zero too.

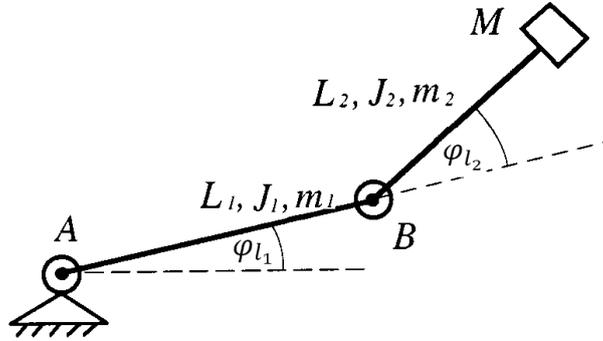


Figure 3.1: Scheme of a robotic arm manipulator with two (flexible) links.

3.2 Reference Trajectory Planning for Feedforward Motion Control

High-performance industrial robotic manipulator systems, such as pick-and-place units and positioning systems, involve short motion times (small cycle times, and hence high accelerations), little settling times and relatively very high accuracies in reference tracking. Moreover, these systems are often set to be used in rest to rest motion fashion, which makes it harder to meet these strict demands in real-time setting. Therefore it is not only enough to utilise even a well- designed controller together with accurate dynamic models, but also an efficient, dedicated motion

controller is needed. A motion controller is usually a computer hardware that is responsible for the details of trajectory planning and motion online execution at each sample instant, and its tasks are mainly comprised of [6]:

- Reference trajectory planning: calculations of an allowable trajectory to be tracked.
- Profile generation: the representation of the reference trajectory as a sampled time sequence.
- Feed-forward control: the control process that calculates the actuating signals to obtain an intended reference trajectory.
- System compensation (includes feedback control): to minimise or mitigate unwanted behaviour like sensed disturbances or unmodelled nonlinearities.
- Internal checks, fault notifications, diagnostics, safety issues, communications, and so.

It is apparent that there is quite a high burden for the motion controller with a high sampling rate requirements. One straightforward, practical implementation of a simple motion controller is known as mass feed-forward (or rigid body feed-forward). In mass feed-forward, the first three tasks (reference trajectory planning, profile generation and feed-forward control) are usually done for each actuator (considered acting on a simple mass object and hence represents a single degree of freedom) separately. As a result of this simplification, the feed-forward control problem is just to generate the required force to accelerate the associate mass according to the desired (allowable) trajectory. The disadvantage of this simple method is its dependence on feedback to deal with unmeasured disturbance or unmodelled behaviour aforementioned, which cause undesired and even dangerous high trajectory tracking inaccuracy, in addition to (indefinite) time delays. To overcome these deficiencies, many academic and practical approaches are suggested and used with the intention to simplify these tasks. These approaches include trajectory smoothing or shaping, plant-inversion feed-forward control and feed-forward control based on optimisation plus system compensation.

A purposeful and promising implementable extension of ‘rigid-body feed-forward’ for higher-order reference trajectory planning that can be incorporated with all of the alternatives mentioned above is the “fourth-order feed-forward”. Furthermore, it is well-matched with the behaviour of electromechanical systems in general (and industrial robotic manipulator systems used in factory automation in particular) and it obtains time optimality within given physical actuator bounds and motion system limits.

3.2.1 Higher-order trajectory planning:

A trapezoidal (or triangular) velocity motion profile produces a discontinuous acceleration. As a consequence, the mechanical system that is fed by this acceleration input may suffer efforts and stresses (due to the saturated actuating acceleration), which may cause permanent damages or undesired vibrational effects. Therefore, a smoother motion profile must be defined, for example by adopting a continuous,

linear piece-wise, acceleration profile. In this case, the resulting velocity is composed by linear segments connected by parabolic blends, hence the name double S for this trajectory profile [3].

Generally speaking, second- and third-order trajectories are reasonably known both in industry and academia with numerous approaches to give a valid solution for the trajectory planning problem. However, extensions to fourth-order trajectory planning are not that direct and trivial. Before digging into the details of a generic algorithmic solution for obtaining the required higher-order trajectory (irrespective of the order of the resulting trajectory), the primary objectives of trajectory planning is summarised and reviewed. Those objectives work as general guidelines for applicability of any trajectory planning solution, and briefly, they are (as mentioned in [6]):

- **Timing:** A time-optimal trajectory is always required or at least getting closer to it with the consequences of trajectory execution time specified to assess the effect of boundary conditions.
- **Actuator effort:** This usually forms the basis for selecting limits on velocity, acceleration and jerks, though they sometimes attributed to bounds on mechanical construction or safety aspects.
- **Accuracy:** For point-to-point motion commands, the end-effector position of the trajectory must be set to the desired position at each instant (within acceptable measurement accuracy).
- **Complexity:** Although the trajectory planning is usually treated as being done off-line, however, practically the desired end-effector position may become available only at the moment when the trajectory should be started. Thus, the required time for trajectory planning is lost and this delay should be minimised.
- **Reliability:** The planning algorithm should always yield a valid solution.
- **Implementation:** Trajectory planning algorithm is implemented in computer hardware, making it subject to discretisation and quantisation (digitalisation) characteristics.

3.2.1.1 Fourth-order trajectory planning

As discussed earlier, it is required for our simulation studies of flexible robotic manipulators to have a trajectory with a continuous derivative of acceleration (i.e. jerk). For this reason, a variation of the double S trajectory that has a trapezoidal jerk profile can be used. The overall trajectory is comprised of fifteen segments, where the jerk increases or decreases linearly with time, or is constant at some level. The bound on the first derivative of jerk is necessary to be defined (called jounce, ping, or more formally snap).

Given that the positions at the start and end points are q_0 and q_1 respectively, the corresponding initial and final values of velocity are v_0 and v_1 , and the lower and

upper saturation limits on velocity v , acceleration a , jerk j and snap s are specified as:

$$v_{min} = -v_{max}, \quad a_{min} = -a_{max}, \quad j_{min} = -j_{max}, \quad s_{min} = -s_{max}; \quad (3.3)$$

The fifteen-segments “double S” trajectory consists of an acceleration phase (the first seven segments), a constant velocity phase (the eighth segment) and a deceleration phase (the phases from 9 to 15), all together constituting the whole trajectory with a continuous jerk. The complete trajectory is thus determined by specifying the time length expressions for different component segments (detailed analysis with complete derivations can be found, for e.g., in [3]), according to:

$$\begin{cases} T_s & ; & \frac{j_{max}}{s_{max}} \\ T_j & ; & T_s + \frac{a_{max}}{j_{max}} \\ T_a & ; & T_j + \frac{v_{max}-v_1}{a_{max}} \\ T_d & ; & T_j + \frac{v_{max}-v_0}{a_{max}} \\ T_v & ; & \frac{q_1-q_0}{v_{max}} - \frac{T_0}{2} \left(1 + \frac{v_0}{v_{max}}\right) - \frac{T_d}{2} \left(1 + \frac{v_1}{v_{max}}\right) \end{cases} \quad (3.4)$$

Alternatively, to obtain the jerk, acceleration, velocity and position profiles, one can obtain the corresponding snap profile and integrate it numerically to get the required set of lower-order profiles. The snap profile’s expression is particularly rather simple and is given by:

$$q^{(4)}(t) = \begin{cases} +s_{max} & ; & t \in [0, T_s] \\ 0 & ; & t \in [T_s, T_j - T_s] \\ -s_{max} & ; & t \in [T_j - T_s, T_j] \\ 0 & ; & t \in [T_j, T_a - T_j] \\ -s_{max} & ; & t \in [T_a - T_j, T_a - T_j + T_s] \\ 0 & ; & t \in [T_a - T_j + T_s, T_a - T_s] \\ +s_{max} & ; & t \in [T_a - T_s, T_a] \\ 0 & ; & t \in [T_a, T_a + T_v] \\ -s_{max} & ; & t \in [T_a + T_v, T_a + T_v + T_s] \\ 0 & ; & t \in [T_a + T_v + T_s, T_a + T_v + T_j - T_s] \\ +s_{max} & ; & t \in [T_a + T_v + T_j - T_s, T_a + T_v + T_j] \\ 0 & ; & t \in [T_a + T_v + T_j, T - T_j] \\ +s_{max} & ; & t \in [T - T_j, T - T_j + T_s] \\ 0 & ; & t \in [T - T_j + T_s, T - T_s] \\ -s_{max} & ; & t \in [T - T_s, T] \end{cases} \quad (3.5)$$

where T represents the total trajectory duration, T_s represents the numerical integration step size of the trapezoidal rule used in the bilinear transform (or Tustin’s discretisation method); i.e. the sampling period. The meanings of T_j , T_a , T_d and

T_v are best explained graphically as in Figure 3.2.

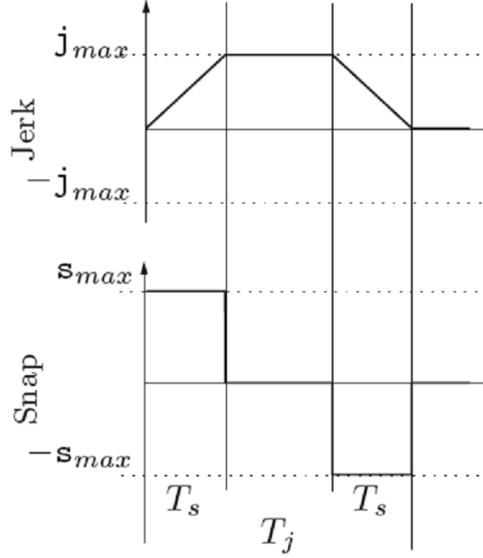


Figure 3.2: Continuous jerk profile in a fifteen-segments “double S” trajectory [3].

Although generating position, speed, acceleration and jerk profiles using this approach is conceptually quite simple (see Figure 3.3 below), care must be taken to avoid numerical problems that can possibly arise and cause unacceptable position/velocity/acceleration drifts.

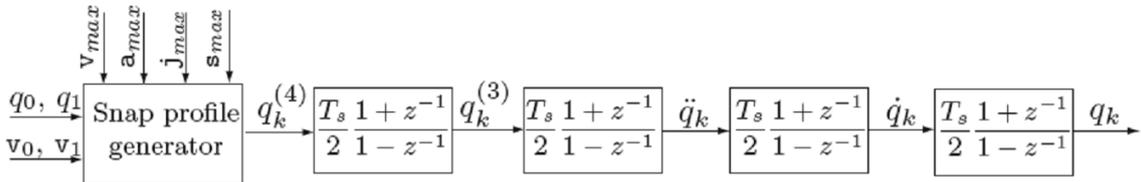


Figure 3.3: Conceptual scheme for the numerical computation of lower-order (jerk, acceleration, velocity and position) profiles of the fifteen-segments trajectory starting from the derivative of jerk “snap” profile [3].

Figure 3.4 reports the position, velocity, acceleration, jerk and snap profiles of a fifteen-segments trajectory concerning a double S motion; with the boundary conditions: $q_0 = 0$, $q_1 = 1$, $v_0 = 1$, $v_1 = 0$, and the constraints: $v_{max} = 1$, $a_{max} = 5$, $j_{max} = 50$, $s_{max} = 1000$. It is worth mentioning that this trajectory profile represents our first trial to design a reference input to be used with our inverse dynamics feed-forward computed torque controller.

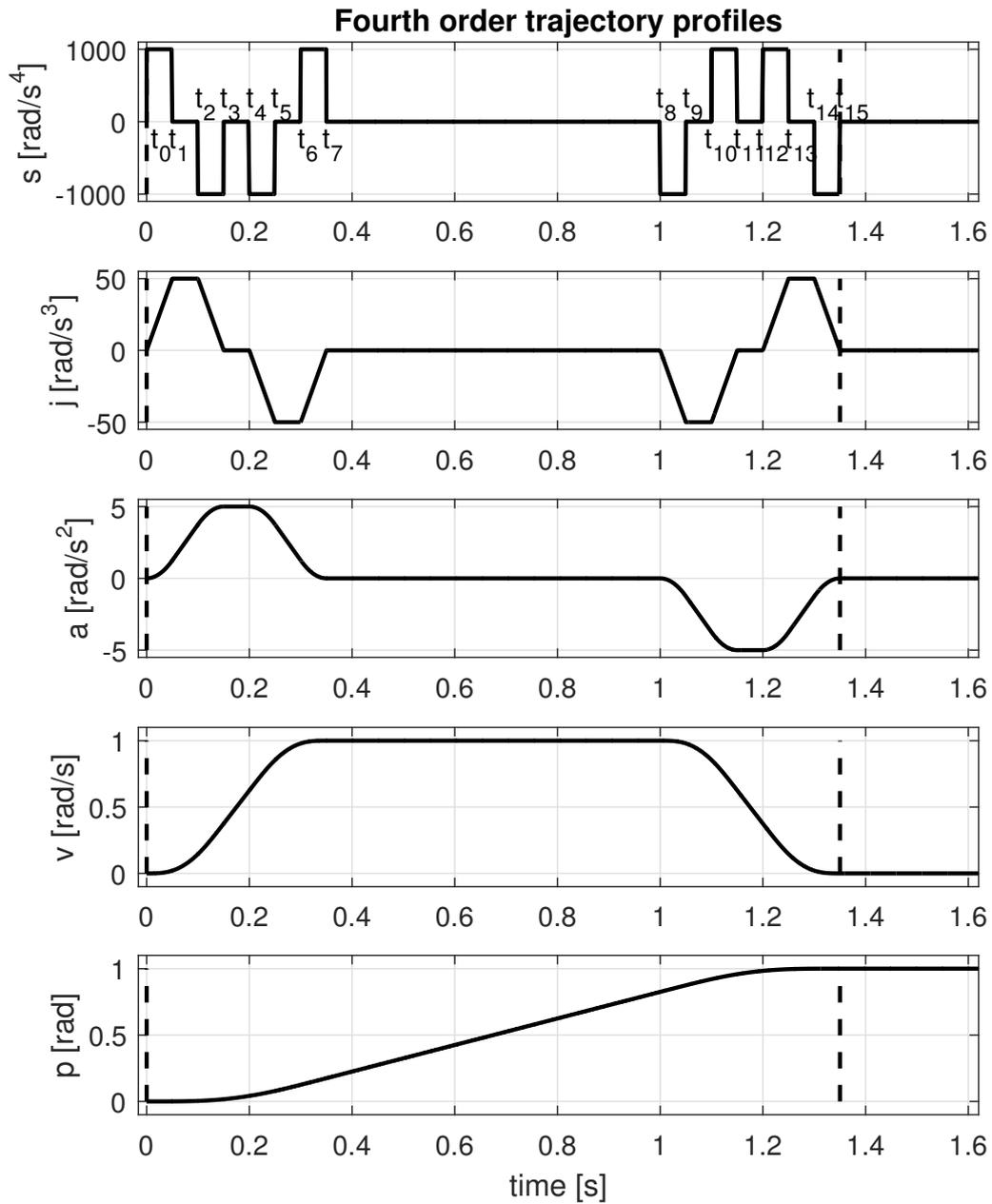


Figure 3.4: Fifteen segments trajectory (position, velocity, acceleration, jerk and snap).

It is sometimes desirable to have even higher derivatives of jerk ($q^{(5)}(t)$, $q^{(6)}(t)$, $q^{(7)}(t)$, ...etc) so as to reduce the filtering needed in the inverse dynamic models which leads to a reduction in the manipulator bandwidth and hence its maximum speed. For this purpose, one can iterate the above mentioned algorithmic solution for trajectory generation to design the higher derivative of jerk profile (up to the order required smooth) firstly, and then use a chain of integrators (and Tustin's discretisers if required) to get the rest of lower-order trajectories profiles as previous. Figure 3.5 reports the position, velocity, acceleration, jerk and snap, together with first and second derivatives of snap of a continuous second derivative of jerk profile, generated using the technique that is just described (i.e. after iterating the algorithm of generating the fifteen-segments trajectory twice). The boundary conditions and constraints imposed on the reference motion are the same as before. This reference was used in the simulations, meaning that the position reference can be differentiated six times in the inverse dynamics calculations used for feed-forward control.

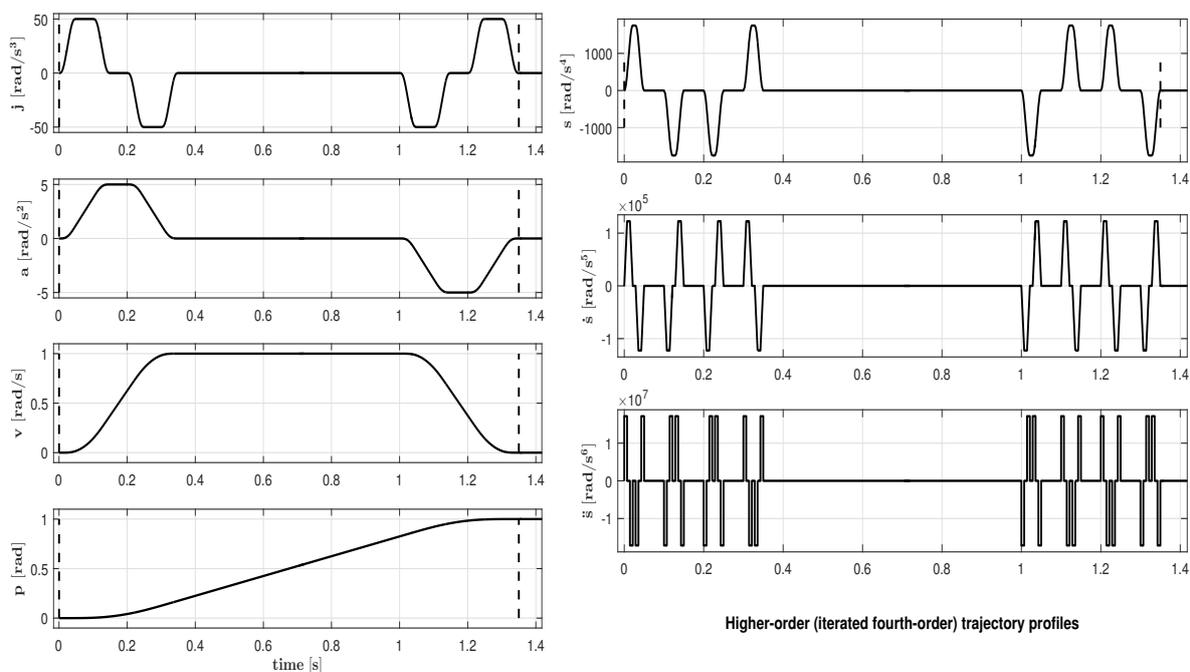


Figure 3.5: Iterated (twice) fifteen-segments trajectory profiles (position, velocity, acceleration, jerk, snap and its derivatives up to the second derivative).

3.3 Servo PID Controller

It is well-known, both in academia and industry, that the proportional-integral-derivative (PID) controller is the simplest, yet the most useful control algorithm in many practical solutions (for both linear and nonlinear systems). Many structures are available when it comes to the implementation of a PID controller, depending on the sensors available, all with the primary goal of meeting the position and speed regulation aim to ensure:

$$\lim_{t \rightarrow \infty} q(t) = q_d, \quad \lim_{t \rightarrow \infty} \dot{q}(t) = \dot{q}_d \quad (3.6)$$

where q_d and \dot{q}_d are constants specifying desired reference set-points to be tracked at each sampling instant. For our simulation studies here, q_d and \dot{q}_d will represent the angular position and angular speed reference trajectory profiles, respectively, for a flexible link's actuator (a servo motor in our case).

From a stability perspective, the hierarchical structure shown in Figure 3.6 is preferred [7]; where p is the differential operator, (τ_{min}, τ_{ffw}) are the actual motor torque and the feed-forward computed compensation torque, and (k_p, k_v, k_i) are the proportional, integral, and derivative gains, respectively, with $k_i = \frac{1}{T_i}$ (T_i is the integrator reset time). This specific PID variant is based in a velocity inner loop plus a position outer loop to keep the closed-loop system stability less stringent than those required when the PID control is implemented as the other PI position loop plus velocity feedback classic variant.

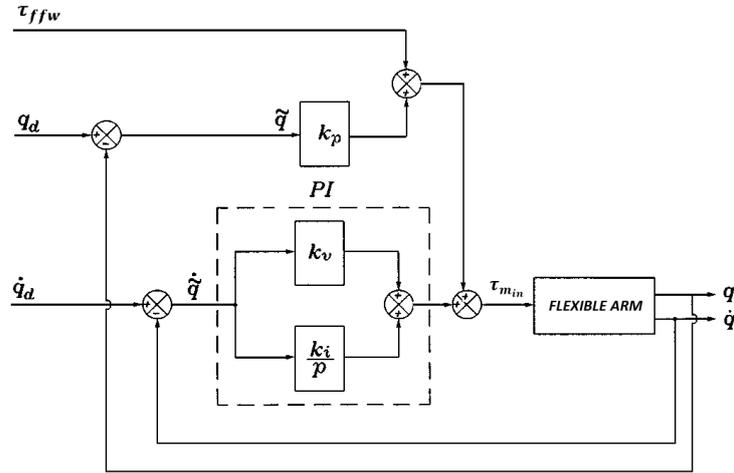


Figure 3.6: PID control based on PI feedback of velocity error.

3.4 Inverse Dynamics (Feed-forward) Computed Torque Controller

Most of the time, the end-effector of some industrial manipulator is required to track a specific trajectory (a continuous function of time) instead of a discrete set of set-points. Trajectory tracking is needed in applications such as plasma cutting, water jet cutting, laser-cutting, arc welding and paint spraying, where both the position and the velocity of the end-effector have to be controlled at all points along the course of the specified trajectory. Since the underlying nonlinear nature of a manipulator becomes more significant under high-speed trajectory tracking, linear control strategies (like PD controllers and so) then fail to satisfy the required performance metrics anymore and a more sophisticated control approach is then required. Therefore, modern model-based control techniques have to be proposed and designed in order to minimise both the position and velocity tracking errors in all the trajectory points in space. However, with all these new schemes, a feedback part is still incorporated to take care of model errors, gravity and other disturbances, and is usually done by a PD, or a PID, controller [2].

Many modern model-based control schemes have been proposed to handle the trajectory tracking problem in rigid manipulators. Among those, the computed torque control (CTC) method - sometimes called inverse dynamics control - is perhaps the best known control concept for rigid manipulators. It has been proved to be very successful in practice and promising for extensions to be used with larger classes of manipulators (including various flexibilities). CTC is explicitly based on a model of the rigid manipulator and it compensates, via inverse dynamics, the model's nonlinear dynamic terms, decoupling the interactions between the degrees of freedom. This is mainly because the inverse dynamics (or computed torque) control approach accounts explicitly for and compensate most of the nonlinear dynamics of the robotic manipulator robustly. Furthermore, ideally under no model uncertainties, the closed-loop system is both linear and decoupled and can be controlled by simple controller (commonly based on classical linear control theory, for instance, PD or PID). The computed torque method is also known as the static nonlinear state feedback approach, because the tracking objective is achieved through a special type of feedback linearisation [2].

A CTC scheme is usually comprised of two parts: (1) a compensation part to take care of the nonlinear dynamic terms (i.e. a kind of feed-forward control) plus (2) an "internal" or "external" PD or PID-term. In the ideal case when the model matches the actual underlying manipulator dynamics exactly (for example in frequency response sense as used for this thesis work), the CTC method theoretically guarantees globally, asymptotically stable trajectory tracking. However, in practice the tracking performance substantially decreases if the model structure is imperfect or if the parameters are unknown or time-varying (need for robust -/or adaptive control schemes).

In the rest of this section, the two very similar control concepts, inverse dynamics and computed torque, are combined together (as classically done in industry) and

termed as “inverse dynamics computed torque control (ID CTC)”. This suggested inverse dynamics computed torque control approach leads to the familiar second-order, linear error dynamics, which can easily be shown to yield a stable system under certain conditions for the controller gains.

Figure 3.7 shows a global sketch of the reference generator together with the developed controller within the closed-loop system. The role of ID CTC block is to take the desired trajectory profiles (as the ones shown in Figures 3.4 and 3.5) for the link’s tip of the flexible robotic manipulator (q_l^{ref} , \dot{q}_l^{ref} and \ddot{q}_l^{ref}), which are generated by the reference generator shown, and provides the corresponding references trajectories to be sent to the joint actuators (motors) (q_m^{ref} , \dot{q}_m^{ref} and \ddot{q}_m^{ref}).

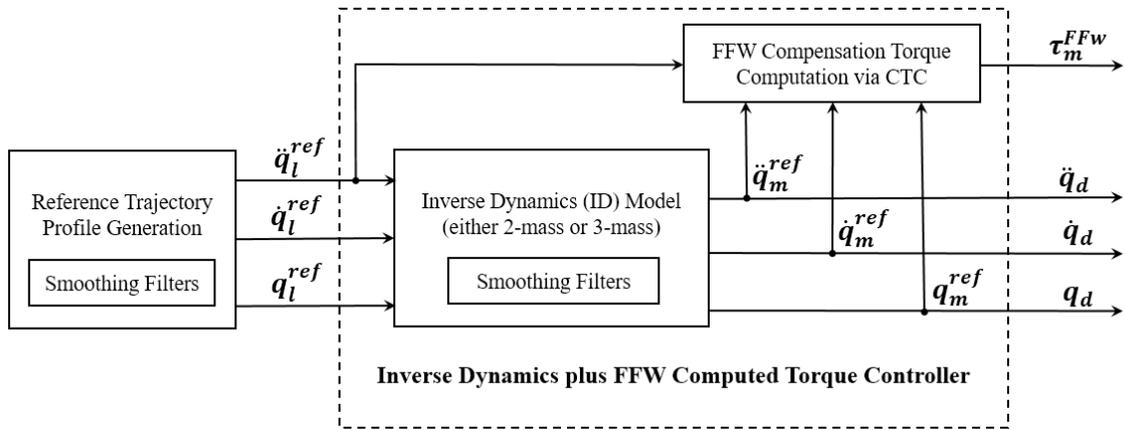


Figure 3.7: Scheme for reference trajectory generation and inverse dynamics plus FFW computed compensation torque controller.

The inverse dynamic (ID) model and the feed-forward computed compensation torque (CTC) for the 2-mass system can be obtained from (2.1) by taking the Laplace transform (in the complex frequency variable s) as:

$$\begin{aligned}
 \varphi_m^{ref}(s) &= \left[\frac{J_g s^2 + d_g s + k_g}{d_g s + k_g} \right] \varphi_l^{ref}(s) \\
 \dot{\varphi}_m^{ref}(s) &= s \varphi_m^{ref}(s) \\
 \ddot{\varphi}_m^{ref}(s) &= s \dot{\varphi}_m^{ref}(s) \\
 \tau_m^{FFW}(s) &= J_m \ddot{\varphi}_m^{ref}(s) + J_l \ddot{\varphi}_l^{ref}(s) \\
 &= J_m \ddot{\varphi}_m^{ref}(s) + d_g (\dot{\varphi}_m^{ref} - \dot{\varphi}_l^{ref})(s) + k_g (\varphi_m^{ref} - \varphi_l^{ref})(s)
 \end{aligned} \tag{3.7}$$

Whereas the inverse dynamic (ID) model and the feed-forward computed compensation torque (CTC) for the 3-mass system can be obtained as two serially-cascaded

2-mass inverse dynamics, as in (3.7) above, from (2.4) as

$$\begin{aligned}
 \varphi_g^{ref}(s) &= \left[\frac{J_g s^2 + d_g s + k_g}{d_g s + k_g} \right] \varphi_l^{ref}(s) \\
 \dot{\varphi}_g^{ref}(s) &= s \varphi_g^{ref}(s) \\
 \ddot{\varphi}_g^{ref}(s) &= s \dot{\varphi}_g^{ref}(s) \\
 \varphi_m^{ref}(s) &= \left[\frac{(J_l s^2 + d_l s + k_l)(J_g s^2 + d_g s + k_g) + J_l(d_l s + k_l)s^2}{(J_l s^2 + d_l s + k_l)(d_g s + k_g)} \right] \varphi_g^{ref}(s) \\
 &= \left\{ \left[\frac{(J_l s^2 + d_l s + k_l)(J_g s^2 + d_g s + k_g) + J_l(d_l s + k_l)s^2}{(J_l s^2 + d_l s + k_l)(d_g s + k_g)} \right] \times \right. \\
 &\quad \left. \left[\frac{J_g s^2 + d_g s + k_g}{d_g s + k_g} \right] \right\} \varphi_l^{ref}(s) \\
 \dot{\varphi}_m^{ref}(s) &= s \varphi_m^{ref}(s) \\
 \ddot{\varphi}_m^{ref}(s) &= s \dot{\varphi}_m^{ref}(s) \\
 \tau_g^{FFW}(s) &= d_l(\dot{\varphi}_g^{ref} - \dot{\varphi}_l^{ref})(s) + k_l(\varphi_g^{ref} - \varphi_l^{ref})(s) \\
 \tau_m^{FFW}(s) &= J_m \ddot{\varphi}_m^{ref}(s) + J_g \ddot{\varphi}_g^{ref}(s) + J_l \ddot{\varphi}_l^{ref}(s) \\
 &= J_m \ddot{\varphi}_m^{ref}(s) + d_g(\dot{\varphi}_m^{ref} - \dot{\varphi}_g^{ref})(s) + k_g(\varphi_m^{ref} - \varphi_g^{ref})(s)
 \end{aligned} \tag{3.8}$$

In general, taking the derivative of a discontinuous or noisy signal tends to exacerbate the influence of that noise, and differentiators (even in Simulink or more importantly in practise) exhibit this behaviour. Therefore, one should make sure that all signals in the reference generator and the inverse dynamics plus feed-forward controller are continuous in time and amplitude with a small derivative at $t = 0$ and relatively low level of noise. Since there is need for taking derivatives of the motor reference position signal (obtained via inverse dynamics) in order to get the corresponding velocity, acceleration and also then the feed-forward compensating torque, it is evident that the reference trajectory should be appropriately defined to avoid running into numerical problems or instability. To this end we can employ the following tricks: (1) rearranging the dynamic equations so as to describe the inverse dynamic system in its best-form mathematical model exclusively that involves only integrators (i.e. no differentiators at all), (2) including the differentiator into a transfer function that directly precedes it whenever that is possible, and (3) always filter the signal before taking the derivative and afterwards as much as possible. It is clear that the first remedy is not valid with our 2-mass and 3-mass inverse dynamic models aforementioned. On the other hand, the second approach can always be used. Then it is important that the reference to the inverse dynamic model is possible to differentiate the number of times needed in the inverse dynamic model, see Section 3.5.

3.5 Smoothing Filters for Reference Profile Signals and Inverse Dynamics Derivations

In general, digital filtering may indicate any form of digital signal processing. In the narrow context of robotics manipulators that we are currently discussing, it refers to discrete-time low-pass (LP) smoothing, that has been an essential tool in designing inverse dynamics-based feed-forward (FFW) compensators in industrial robotics manipulators.

Practically speaking, a good practice that is encouraged before feeding a reference trajectory profile signal to an inverse dynamic model is to filter/smooth it, to avoid discontinuities that may arise upon successive differentiations encountered later along the inverse dynamics path. It is also known that the resulting position reference signal for motor movement must be sufficiently low-pass filtered in order to prevent excessive noise that can occur when its derivative is taken to obtain the corresponding velocity, acceleration or even more difficult the computed torque for FFW compensation. However, low-pass filtering of kinematic data may remove high frequency components of the actual movement of the controlled link, and the effects of such filter-induced errors lead to an overall slower response and possibly tracking inaccuracies (a critical compromise that should be cautiously addressed). It is the purpose, as for digital filtering, that the cut-off frequency must be decided carefully (or equivalently, the frequency bandwidth or operation window of the controlled manipulator). In order not to largely affect this bandwidth, since the flexibility is introduced in the first place in order to enable the industrial robotic working at higher speeds, the reference signal is generated adequately very smooth by using the newly developed iteration scheme for the higher-derivative of jerk trajectory profile, as discussed in the previous sections. As a result, multiple copies of a very simple second-order low-pass filter that is formed by cascading two first-order low pass filter serially are used as just needed, and each of this second-order smoothers is given by:

$$H_{LP}(s) = \frac{\omega_c^2}{s + \omega_c^2} \frac{\omega_c^2}{s + \omega_c^2} \quad (3.9)$$

It is worth-mentioning here that this low-pass filter pair (two first-order low-pass filters cascaded serially to give a second-order smoother) can also be used to smooth the derivative of the position difference error compensation signal. Such a compensating signal is obtained as the output of the compensation scheme and its derivative (the corresponding velocity difference error compensation signal) may be needed while controlling the ADAMS FEM models. Otherwise, one can tune these filters as required since they are inside the exogenous compensation scheme added on top of the controller but is not really a part of it, thus none of these compensation signals affect the closed-loop system bandwidth.

3.6 Compensation of Differences between ADAMS FEM/Lumped-Masses Model and Flexible Link Model

Before discussing the theoretical grounds and the practical considerations for the proposed compensation schemes, some definitions and conventions are first presented.

Relying on (3.10) and Figure 3.8, the following necessary definitions are first introduced:

$$\begin{aligned}\Delta_{error} &= (\varphi_m - \varphi_l^{fake}) - (\varphi_m^{ref} - \varphi_l^{ref}) \\ \varphi_m = \varphi_m^{ref} &\Rightarrow \Delta_{error} = \varphi_l^{ref} - \varphi_l^{fake} \\ Quota_{error} &= \frac{\varphi_m - \varphi_l^{fake}}{\varphi_m^{ref} - \varphi_l^{ref}}\end{aligned}\quad (3.10)$$

where Δ_{error} is the dynamic difference error between a FEM/lumped model reference angle φ_l^{ref} and the real flexible plant angle φ_l^{fake} (when $\varphi_m = \varphi_m^{ref}$), $Quota_{error}$ is the corresponding ratio.

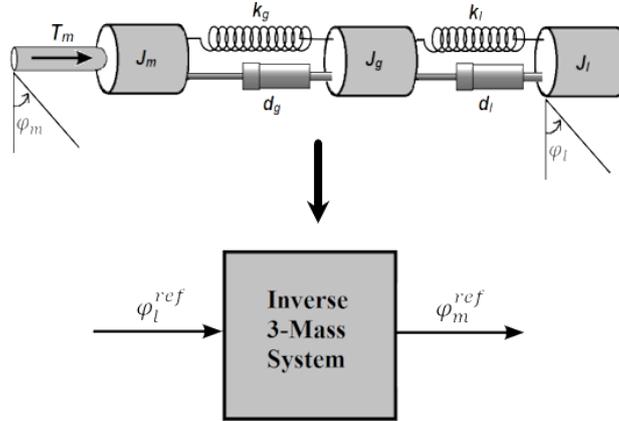


Figure 3.8: A schematic representation of the 3-mass manipulator inverse dynamic model (comprised of a motor, a flexible gearbox and a rigid link) for compensation purposes, together with its system I/O abstraction.

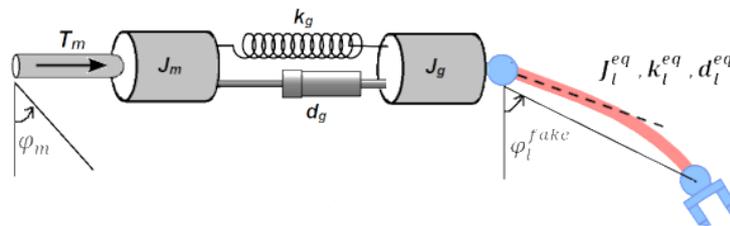


Figure 3.9: A schematic representation of a manipulator plant model with a motor, a flexible gearbox and a flexible link with the motor and “fake” link angular positions demonstrated.

Using a more generalised notation for the motor, gearbox and link angular positions, their angular speeds and angular accelerations, the upper part of Figure 3.10 outlines an exaggerated situation for a flexible link mounted on a motor inertia via a flexible gear. q_m is the motor angle, q_g is the gearbox angle and q_l is the link fake angle. In the lower part of the drawing, the inverse dynamic model used for feed-forward control is shown. In this case, this model is a 2-mass model with a motor inertia connected via a spring to a rigid link inertia.

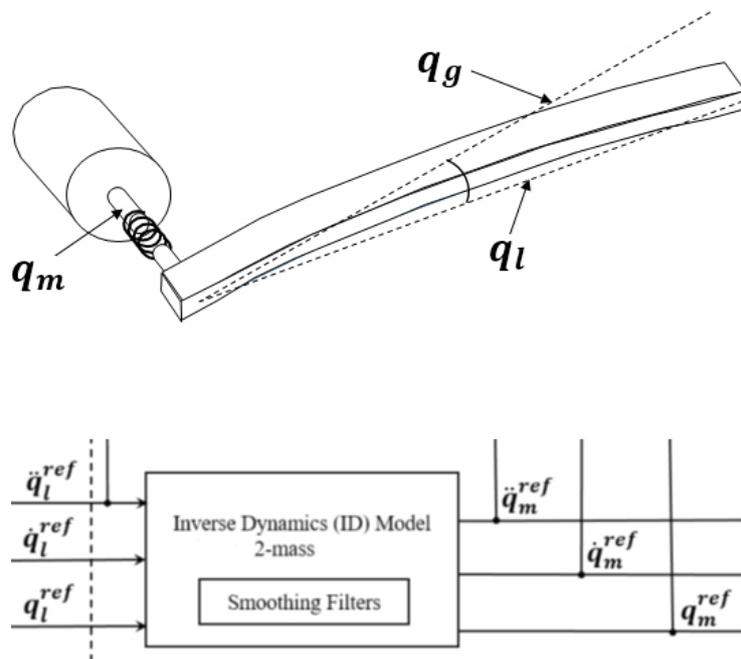


Figure 3.10: A flexible link mounted on a motor inertia via a flexible gear and how the inverse dynamics can be used to control it in a feed-forward fashion.

The upper part of Figure 3.11 shows an example of a compensation relation obtained by theoretical calculations or experimental measurements. The lower part shows how this compensation relation is used.

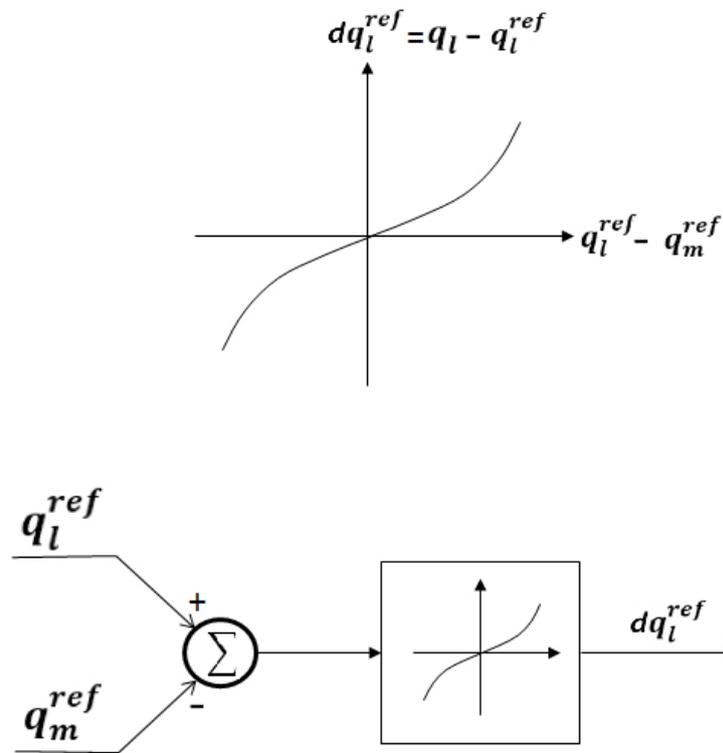


Figure 3.11: A compensation scheme and its usage.

In some cases the link consists of a rigid part, on which a flexible part is mounted. The fake link angle q_l can here be defined according to Figure 3.12. However, this will give another compensation relation than for the case in Figure 3.10.

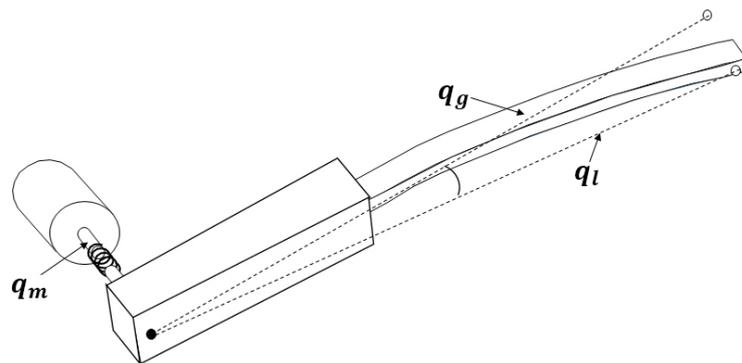


Figure 3.12: A two-links manipulator with a flexible link mounted on a rigid link.

Another way than the one illustrated in Figure 3.12 is to connect a rigid link part with a flexible part. Here the rigid part is mounted with a right angle to the rigid part giving an offset, which will introduce a kinematic error when compensation for the angle of the flexible part is made.

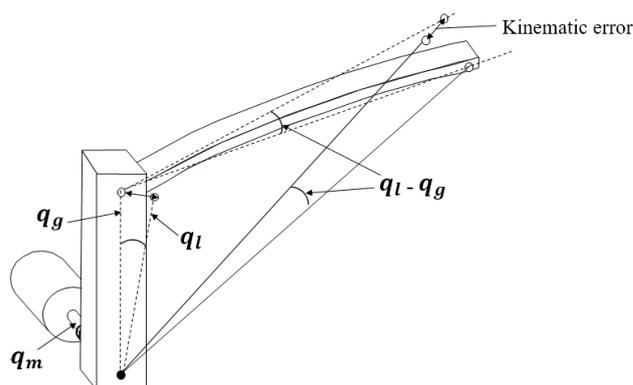


Figure 3.13: A two-links manipulator with a flexible link mounted with a right angle on a rigid link.

When two compliant links form a 2-DOF robot, it is necessary to compensate for the bending angle of the first arm (lower link) when the reference angle of the second arm (upper link) is used as the link angle reference. This case is shown in Figure 3.14.

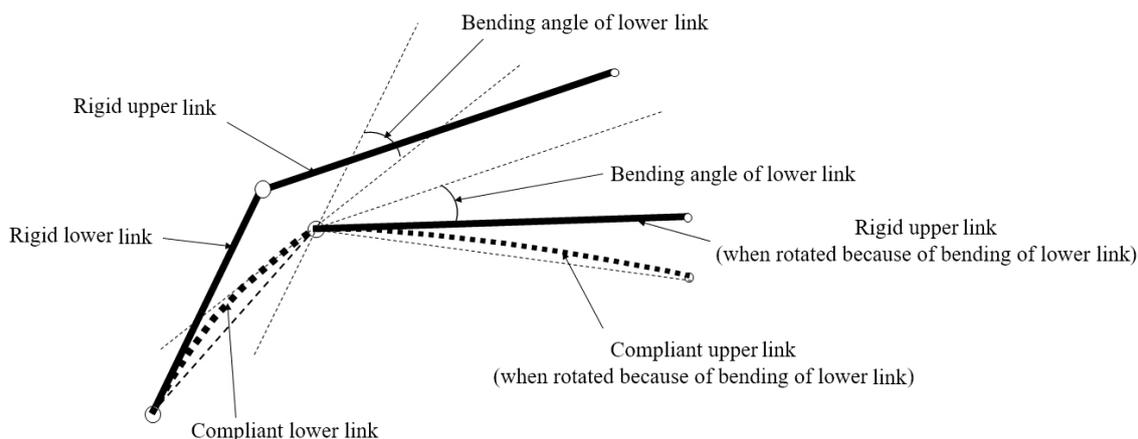


Figure 3.14: A manipulator with two flexible links with some important definitions illustrated.

Now we are ready to give the constructive discussion behind the compensation schemes developed, with the help of the various above shown cases. The idea to use lumped models in the inverse dynamics for the feed-forward control makes it necessary to develop a method to take care of the differences between the lumped models and the real plants with elastic links. This difference will depend both on the angular errors, when a fake angle is approximated with angles between lumped masses, and on kinematic parameters that cannot be taken care of by a simple 2- or 3-mass model. The upper drawing in Figure 3.10 outlines a flexible link connected via a gearbox compliance to a motor inertia. Since in this case the inertia on the output of the gearbox is small, this case will appear as a 2-mass system with the gearbox stiffness in series with the link stiffness. The frequency response of this system can be matched by the frequency response of a lumped-mass system with 2

masses separated by a spring and a damper. The inverse of such a matched 2-mass system is implemented in the inverse dynamic model with inputs and outputs as shown in the lower drawing of Figure 3.10. Here q_m should excite the servo in such a way that q_l^{ref} should be as close as possible to the fake angle q_l in the real link. However, because of the differences between the real system and the model, there will be a difference $(q_l - q_l^{ref})$. Now it is possible to calculate or measure this difference and relate it to $(q_l^{ref} - q_m^{ref})$ as exemplified in the upper drawing in Figure 3.11. This relation is then stored as a table or a mathematical expression and used for compensating the input to the inverse dynamic model. Of course the same concept can be used also for a 3-mass system.

Figure 3.12 shows a case when the link consists of one rigid part and one compliant part. In this case, the fake angle will contain the rigid part and another relation will be achieved between $(q_l^{ref} - q_m^{ref})$ and q_l and q_l^{ref} . If the rigid part has low inertia in relation to the link (link+load in a real case) a 2-mass inverse dynamic model can be used, otherwise a 3-mass model is needed or a 2 mass model matched to the lowest zero and pole in the frequency response of the real system.

Figure 3.13 illustrates a case where full accuracy cannot be obtained using the inverse dynamics for feed-forward, and the compensation scheme in Figure 3.12 will not be able to compensate for this kinematic error. However, the arm design should consider this and keep offsets in this direction as small as possible. In order to compensate for this error at least one more DOF is needed, and then it is possible to use a compensation scheme including the kinematics of the whole manipulator.

Figure 3.14 shows a case that has to be handled when the manipulator has more than one link. Here the compliance of the first link gives a bending in the tip of this link, which will add angle to the second link. This can be compensated for by a table generated from experiments or from FEM models; obtaining the bending angle of the lower link as a function of the difference between the link angle reference and the motor angle reference of the inverse dynamic model of the lower link. The estimated bending angle can then be added to the link angle reference of the inverse dynamic model of the upper link.

If a manipulator is designed without any larger offsets as in Figure 3.13, it will be possible to use the proposed compensation scheme for a multi DOF manipulator. If the situation as in Figure 3.13 or other situations that need a kinematic manipulator model, it is possible to input the angle compensations obtained from tables or mathematical expressions based on the compliances from the inverse dynamic models to a kinematic compensation model generating the necessary compensations on the inputs of the inverse dynamic models of the robot joints.

One point that must be considered with this concept is that compensations will be made on old calculations of the inverse dynamics. However, the errors caused by a lag in compensation will be small if the sampling time is high in relation to the time constants of the manipulator dynamics. Otherwise, it is always possible to make double inverse dynamic model calculations or having two inverse dynamic models running in parallel. In this Master Thesis work, two parallel inverse models were used motivated by implementation reasons in Simulink.

4

Simulation Results and Discussion

4.1 Simulation Parameters and Scenario

Before delving into the results, their analysis and discussion, some numerical values for the simulation parameters are given.

Following the rules of thumb for tuning of the three PID parameters (k_p , k_v and k_i) to get an oscillation-free accurate tracking results, and after further fine tuning process depending on the end-effector's tracking errors, the following two set of PID parameters gave the best tracking results:

$$k_p = 0.75, k_v = 1.5, T_i = 0.25 \quad (4.1)$$

$$k_p = 0.3, k_v = 0.3, T_i = 3 \quad (4.2)$$

the first PID parameters set can be used with simple 2-mass and 3-mass plant systems, while the second one can be used for controlling the lumped multi-mass and ADAMS FEM models of the flexible link.

In addition, ($\omega_c = 20 \text{ rad/s}$) is reached for the filter in (3.9) after some tuning to get both a fast response and an oscillation-free tracking for velocity and acceleration that is numerically stable. In addition, this filter's cut-off frequency respects the high smoothness requirements discussed earlier while still preserves a reasonable controlled manipulator's bandwidth to fully achieve the tasks requested within the demanded performance specifications.

The first target of this thesis is to obtain a basic understanding of the relationships between flexible manipulator arm model types and feed-forward inverse dynamic model types. The flexible manipulator arm model dynamics should be as close to reality as possible and the inverse model should be as simple as possible. Manipulator arm models that have been studied are 2-mass and 3-mass lumped-mass models, multi-mass lumped-mass models and FEM models, where the FEM models should have the dynamics and compliance geometry closest to a real manipulator arm. In order to make it possible to run the inverse dynamic models in real-time in a real industrial robot controller, only 2-mass and 3-mass lumped-mass models have been used. The biggest problem with the inverse models is the necessary chain of derivatives and hence, the inverse 2-mass model should be the first choice. An important finding therefore is that even if the manipulator arm model corresponds to a 3-mass model it is still possible to obtain good results when using a 2-mass inverse model,

especially if its frequency response is matched to the lowest pair of anti-resonance and resonance frequencies.

The second target of the thesis is to test a new concept to compensate for differences between the dynamics and compliance geometry of a 2-mass/3-mass inverse model and a more accurate arm model. The concept used is to save the error between the real link fake angle and the lumped-mass link angle as a function of the compliance in the lumped-mass system. When running the simulation, the link position reference to the inverse dynamic model is compensated for this error using the saved error as a function of the actual compliance of the inverse dynamic model.

4.2 Simulation Results with Comments

In Table 4.1, the results from different control and arm model combinations are shown, focusing on the inverse model type. The values depicted are the maximum deviation between the actual link angle/speed and the reference angle/speed. For comparison, no inverse model at all is used as a first option and the results rely on the tuning of the PID controller. The simulations show that as expected, the 2-mass lumped model is easiest to tune with a PID controller while 3-mass, multi-mass and ADAMS models will give 4 – 5 times worse results.

Inverse Model Type	Link Model Type	Compensation Type	Angular Error (rad)	Speed Error (rad/s)
No (only PID)	2-Mass lumped model	No	0.020	0.080
	3-Mass lumped model	No	0.110	0.360
	Multi-mass one resonance	No	0.090	0.290
	Multi-mass two resonances	No	0.090	0.300
	ADAMS one resonance	No	0.110	0.360
	ADAMS two resonances	No	0.110	0.310
2-Mass lumped	2-Mass lumped model	No	0.004	0.008
	3-Mass lumped model	No	0.006	0.026
	Multi-mass one resonance	No	0.003	0.007
	Multi-mass two resonances	No	0.004	0.012
	ADAMS one resonance	No	0.006	0.035
	ADAMS one resonance	Yes, angle	0.003	0.047
	ADAMS one resonance	Yes, angle and speed	0.003	0.047
	ADAMS two resonances	No	0.004	0.030
	ADAMS two resonances	Yes, angle	0.003	0.054
	ADAMS two resonances	Yes, angle and speed	0.007	0.055
3-Mass lumped	3-Mass lumped model	No	0.004	0.007
	Multi-mass two resonances	No	0.004	0.010
	ADAMS two resonances	No	0.006	0.034
	ADAMS two resonances	Yes, angle	0.003	0.046
	ADAMS two resonances	Yes, angle and speed	0.003	0.046

Table 4.1: Angular errors and speed errors when different inverse dynamic model types are used for feed forward control. The errors are recorded for the different link dynamic models developed in the thesis.

In the case when a 2-mass lumped inverse model is used for feed forward control, it can be seen that the results of angle errors are gathered in a smaller interval (0.003 – 0.0065) and now the tuning of the PID is no longer as critical as without feed-forward control. As expected the best results are obtained when the link model has only one resonance (including the 2-mass lumped link model). In the case of

4. Simulation Results and Discussion

links with two resonances (including the 3-mass lumped link model), the error is not much larger. In the cases when the link model has two resonances, the inverse lumped 2-mass model is matched to the lowest pair of anti-resonance and resonance frequencies of the link model. This will give the best results as can be seen in Table 4.1.

Using a lumped 2-mass inverse model was also tested on the ADAMS two resonances link model with the compensation method described in Section 3.6. It was shown that compensating the link reference angle reduced the angle error while the speed error was increased. The compensation of both the link reference angle and the speed gave worse results. Hence, more investigative studies are needed here.

Feed forward control with lumped 3-mass model on links with two resonances (including 3-mass lumped model) gave about the same results as with feed-forward using the lumped 2-mass system. One reason for this could be that the inverse 3-mass model includes more differentiations and filters, whereby the feed-forward signals will not be that accurate. The compensation method described in Section 3.6 was also used for the ADAMS two-resonances model with a 3-mass inverse model. It was shown that the compensation reduced the angle error a factor of 2; both for angle compensation and combined angle and speed compensation. The speed error was in both cases increased somewhat.

In Table 4.2, the data has been arranged to easier find which inverse model (including only PID control) that offers the best performance for a given link model. As can be seen, no feed-forward always gives the worst result. For the 3-mass lumped link model, a 3-mass inverse model gives the best result, especially interesting is the very low-speed error. It is also interesting to note that for the multi-mass model with two resonances, the 2-mass and 3-mass inverse models give very close results.

Link Model Type	Inverse Model Type	Compensation Type	Angular Error (rad)	Speed Error (rad/s)
2-Mass lumped model	No (only PID)	No	0.020	0.080
	2-Mass lumped model	No	0.004	0.008
3-Mass lumped model	No (only PID)	No	0.107	0.360
	2-Mass lumped model	No	0.006	0.026
	3-Mass lumped model	No	0.004	0.007
Multi-mass one resonance	No (only PID)	No	0.090	0.290
	2-Mass lumped model	No	0.003	0.007
Multi-Mass two resonances	No (only PID)	No	0.091	0.300
	2-Mass lumped model	No	0.004	0.012
	3-Mass lumped model	No	0.004	0.010
ADAMS one resonance	No (only PID)	No	0.110	0.360
	2-Mass lumped model	No	0.006	0.035
	2-Mass lumped model	Yes, angle	0.003	0.047
	2-Mass lumped model	Yes, angle and speed	0.003	0.047
ADAMS two resonances	No (only PID)	No	0.109	0.312
	2-Mass lumped model	No	0.004	0.030
	2-Mass lumped model	Yes, angle	0.003	0.054
	2-Mass lumped model	Yes, angle and speed	0.007	0.055
	3-Mass lumped model	No	0.006	0.034
	3-Mass lumped model	Yes, angle	0.003	0.046
	3-Mass lumped model	Yes, angle and speed	0.003	0.046

Table 4.2: A rearrangement of Table 4.1 to easier see which inverse dynamic model type that best suits a given link dynamic model type.

For both the ADAMS link models, the angle compensation scheme yields about half

the angle error, which is very encouraging for future analysis of this concept.

Table 4.3 shows the importance of how a 2-mass lumped inverse model for feed-forward control is matched to a link with two resonances, in this case a 3-mass lumped-mass link. As a reference, the angle- and speed errors are shown for a 3-mass lumped inverse model. As can be seen, the best results are obtained when matching the 2-mass lumped inverse model to the first zero- and pole frequencies (the lowest pair of anti-resonance and resonance frequencies). However, matching the zero of the 2-mass model to the first zero of the 3-mass link model and the pole to the second pole of the 3-mass link model will just give slightly worse results. This means that the most important interesting frequency to match with the feed-forward inverse dynamics model is the first zero (the lowest anti-resonance). This conclusion is further verified by the case when matching is made for the highest pair of anti-resonance and resonance frequencies, where the worst tracking results are obtained. However, PID control without feed-forward will still give even worse results.

Link Model Type	Inverse Model Type	Angular Error (rad)	Speed Error (rad/s)
3-Mass lumped model	3-Mass lumped model	0.004	0.007
	2-Mass lumped model lowest anti-resonance and lowest resonance	0.006	0.026
	2-Mass lumped model lowest anti-resonance and highest resonance	0.007	0.028
	2-Mass lumped model highest anti-resonance and highest resonance	0.070	0.250
	No (only PID)	0.107	0.360

Table 4.3: Results from different matching strategies for the inverse dynamic 2-mass model when the link model has two resonances. In this case, the link model is a lumped 3-mass model.

Summing up, from Tables 4.1, 4.2 and 4.3, we can make the following conclusions:

- A lumped 2-mass inverse model for feed-forward matched to the lowest pair of anti-resonance and resonance frequencies will give good results for all the link models studied. This is a very promising finding since a 2-mass inverse model will need less number of differentiations and therefore, the requirements on the reference generator and filtering of differentiation steps will be much lower.
- The new compensation scheme presented in Section 3.6 works and gave in the case of the tested ADAMS FEM link model a reduced angle error of a factor of 2.
- Using feed-forward control with frequency response matching gives about the same results for all the link model types studied. This means that it will be possible to make simulations with simple dynamic models when new control concepts are tested, for example when a manipulator with two or more links are analysed or when concepts for friction compensation is added to the feed-forward control.

Figure F.28 and Table F.28 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link's tip tracking performance. The plant system whose frequency response is shown in Figure 2.20 is in effect a 3-mass plant system. The tracking results shown are obtained when this system is controlled by a 3-mass

4. Simulation Results and Discussion

inverse dynamics with a simple servo PID controller, in addition to both angular position and angular speed difference errors compensation.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.001066	0.002929	-0.039955	0.046281	-0.249029	0.228360

Table 4.4: An ADAMS system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller with angular position and angular speed difference errors compensations results

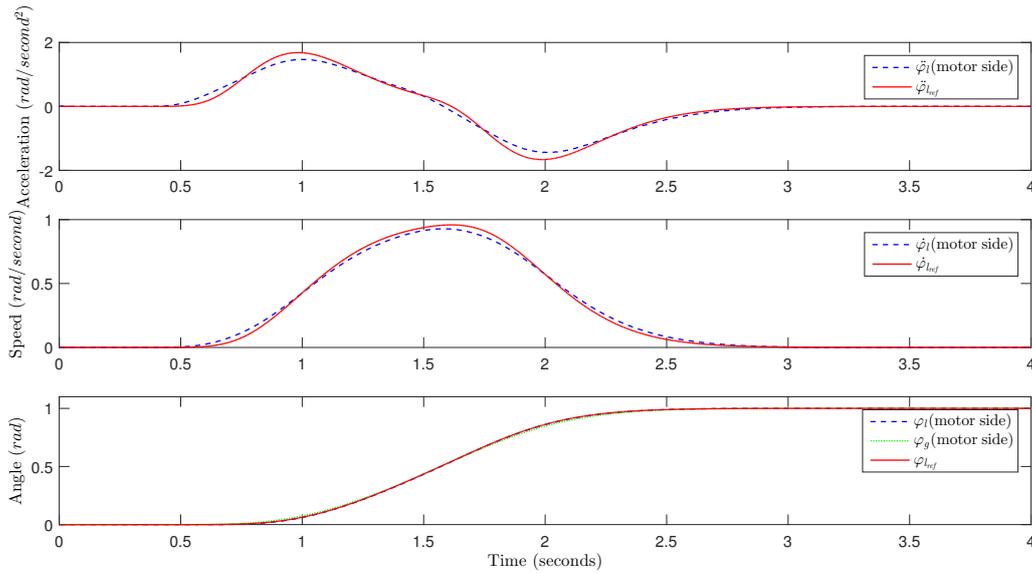


Figure 4.1: Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller, in addition to both angular position and angular speed difference errors compensation (a 3-mass system).

The rest of the results discussed and summarised here together with their qualitative tracking plots and numerical error values are given in Appendix F.

5

Conclusions and Future Work

The dynamic analysis and control of flexible manipulators is an emerging research area in both academia and industry fields (such as manufacturing, automation, and robotics), where a lot of work is required to control the effects of flexibility. Robotic manipulators with flexible links became more widely used in many applications (starting from simple pick-and-place operations of an industrial manipulator to micro-surgery and maintenance of nuclear plants). Combining both joint and link flexibilities in the manipulator model and compensating for their effects in closed-loop systems in real time is a challenging problem to solve. In this thesis, a modelling- and control technique for manipulators with both flexible joints and flexible links has been presented.

The inverse dynamics “feed-forward” computed torque control scheme has been introduced to derive a model-based real-time controller for various manipulator plants. This model considers both concentrated joint flexibilities as well as distributed link flexibilities, while only using joint sensors (collocated control). To perform realistic simulations, ADAMS FEM models for the flexible link together with the inertia on the gearbox output shaft are developed and integrated in a cosimulation environment with MATLAB/Simulink. Furthermore, models of flexible links using different numbers of lumped masses have been developed and their dynamic behaviours are compared with those of the corresponding ADAMS FEM models. The frequency responses of 2-mass and 3-mass models were successfully matched to all the types of link models used. The inverse matched 2-mass and 3-mass systems were then derived and used for feed-forward control. Simulations with feed-forward control using the lumped or FEM versions of the link model have shown that the inclusion of both joint and link flexibilities in the inverse feed-forward control model achieves satisfactory tracking accuracy within the desired boundaries. Both joint and link vibrations are suppressed effectively, even though the robotic manipulator system is underactuated (the number of degrees of freedom exceeds the number of control inputs).

In order for the inverse dynamics (or computed torque) control method to work, a desired reference trajectory that is smooth-enough (i.e. can be differentiated the required number of times depending on the complexity of the inverse dynamics model used, either 2-mass or 3-mass) is generated. That is done by iterating the process of fourth-order trajectory planning (smooth derivative of jerk profile) twice to yield a position trajectory profile with a continuous second derivative of snap. Additionally, a number of second-order smoothers, comprising two serially-cascaded

identical copies of simple low-pass filters, are placed whenever a derivative is taken and the cut-off frequency of each simple low-pass filter is determined after careful tuning so as not to slow the overall system response, i.e. reduce its bandwidth, too much. However, one part of the residual path tracking error was probably due to missing bandwidth in the differentiation steps of the inverse model.

The proposed model-based controller can be regarded as a conventional inverse dynamics computed torque controller (ID CTC) plus a compensation unit that accounts for the effects of errors in inverse dynamic models due to neglected coupled inertia between the lumped masses and neglected link flexibility dynamics (in case the plant has higher order, for example when a 2-mass inverse dynamic model is used to match a 3-mass plant resonance and anti-resonance frequencies). The required inverse dynamics schemes were developed based on the frequency-domain analysis of the plant models used. Additionally, a concept for compensation of difference errors has also been introduced and used to compensate for residual tracking errors. This compensation made it possible to reduce the residual error by 50%. In particular, we obtain the relation between the residual error and the dynamic lag in the inverse dynamic model and then we use this relation to generate a compensation signal to the inverse dynamic model during the link control.

5.1 Future Work

One issue that should be studied in future is how to improve the compensation in the time-domain of the residual tracking errors. Even if the obtained 50% reduction in the tracking error is a good result, it should be possible to reach further reduction. Here one question regards whether it is possible to reduce residual errors because of lacking differentiation bandwidth in the inverse model. Another question concerns whether it is possible to reduce the residual errors because of differences between the dynamics of the lumped-mass models used for feed-forward control and the controlled link model. A closer study of the dynamic influence of the time-domain compensation is then needed. It will be probably possible to increase the efficiency of the compensation by model-based compensation filters. Whether to introduce the compensation signal and where in addition to how to tune the timing should also be studied.

An example of a link structure that gave larger residual errors with the lumped-mass models used for feed-forward control is the link structure outlined in Figure 3.12. An ADAMS implementation of this model can be seen in Figure 5.1. Here the link consists of one rigid part with about the same length as the flexible part mounted on the tip of the rigid part. In order to obtain a better path tracking using this link structure a 3-mass model better adapted to the link structure is probably needed. A future research possibility in this respect is to find out if specialised 2- or 3-mass lumped-mass dynamic models can be derived to obtain the same good path tracking performance for structures like the one in Figure 5.1 as for the structures studied in this thesis work.

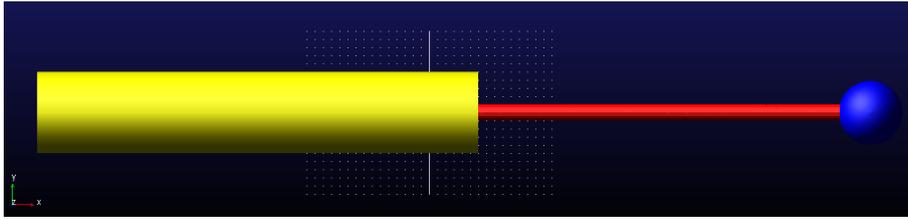


Figure 5.1: A manipulator link with a rigid part having the same length as the outer flexible part. The output of a flexible gearbox (implemented in Simulink) is connected to the rigid part at its left end.

The proposed model-based strategy followed during modelling, simulation and control of various mechanical design of the planar robotic manipulator with a flexible joint and a flexible link is more complete than other methods used earlier, in the sense that it considers both the effects of structural damping in joints and links in addition to payload. It provides a promising scheme to study “nonlinear” under-actuated robotic manipulator systems, and to integrate other sources of additional phenomena than joint and link flexibilities, such as friction, clearances and backlash. In addition, using a modular approach for the model formulation offers the opportunity to exploit it in order to obtain dynamic models (approximate frequency responses) for practical flexible manipulators that incorporate any number of joints and links, building upon the models and results obtained. The only thing that should be taken care of then is the specific compensation scheme for residual tracking errors of the specific controlled plant.

As an example, Figure 5.2 shows a proposal for a scheme to control a manipulator with two links using the results of the present thesis. The scheme can, of course, be expanded to manipulators with more degrees of freedom. A trajectory is generated in Cartesian space and a dynamic filter generates the dynamic path profile with speed, acceleration and the number of derivatives of the acceleration needed for proper use of the feed-forward inverse dynamic models. The allowable acceleration and speed levels are obtained from a rigid multi-body dynamic model. The dynamically-filtered trajectory in Cartesian coordinates is then transformed to joint coordinates by a kinematic model. The joint value outputs of this model are used by the rigid multi-body dynamic model to calculate the parameters of the dynamic equations of motion for the manipulator. The joint values (angle, speed, acceleration) are also used as link angle reference values to the inverse dynamic models, one model for each joint. These models are based on lumped 2- or 3-mass models and the output of the inverse dynamic models form the references (angle, speed) and feed-forward torques to the PID servo units. The output and input of the inverse dynamic models are used as inputs to the angle compensation units. These units calculate from calibration curves (usually a linear relationship) the difference between the link reference angle input to the inverse dynamic model and the real link angle. This difference is buffered to avoid numerical loops and then fed into the inverse dynamic model for error compensation.

It is assumed that joint 2 is mounted on joint 1 and to compensate for the bending of the first link, the bending angle is calculated from the input and output of the

inverse dynamic model of joint 1 together with a calibration curve.

In order to compensate for coupled inertia, Coriolis torques and centrifugal torques, dynamic coupling calculations are made based on the actual parameters from the rigid multi-body dynamic model given the actual speed and acceleration values of the link as calculated by the inverse dynamic models.

There are of course many ways to build up a system for the control of manipulators with compliant links. For example, the outputs from the angle compensation units and the bending compensation units could go to a kinematic compensation unit, which with the help of the kinematic model of the whole manipulator can calculate the angle compensation signals to the inverse dynamic models. It could also be possible to use the outputs from the angle compensation units to compensate the trajectory via inverse kinematics calculations. In this case, the compensations could also be stored and used in an iterative learning control concept.

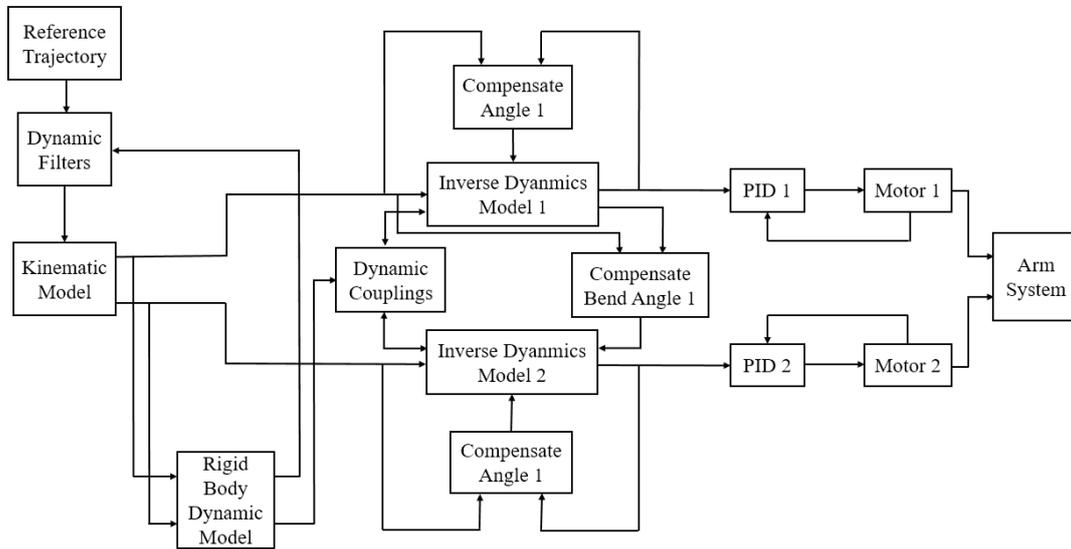


Figure 5.2: Proposal of how to connect the modules developed in the thesis to control a manipulator of two degrees of freedom.

One more important issue to reflect on here is that linearised models of the distributed link flexibility are considered in this Master’s thesis study, which reduced the complexity of the model-based controllers employed. Accordingly, the resulting models can not take care of large elastic deflections of the manipulator links and more general 3D motion. Fortunately, these cases are unlikely to happen in a real industrial manipulators for the application areas discussed, due to their limited payload weight usually.

Another aspect that is very crucial and needs further investigation is the energy consumption associated with the modelling, control and compensation techniques developed, studying the possibility of incorporating some active elements like springs or dampers. Such topic and others related to the operation and implementation can be addressed after testing the designed reference generation and proposed control and compensation schemes on a real robotic manipulator to verify the results and validate the conclusions drawn.

Bibliography

- [1] M. Grossard, S. Régnier, and N. Chaillet. *Flexible robotics: applications to multiscale manipulations*. Wiley, 2013.
- [2] I.M.M. Lammerts. *Adaptive computed reference computed torque control of flexible manipulators*. PhD thesis, University of Technology Eindhoven, Eindhoven, The Netherlands, 1993.
- [3] L. Biagiotti and C. Melchiorri. *Trajectory planning for automatic machines and robots*. Springer-Verlag Berlin Heidelberg, Germany, 1st edition, 107-116, 2008.
- [4] G. Yanqing, F.-Y. Wang and Z.-Q. Xiao. *Flexible manipulators: modeling, analysis and optimum design*. Academic Press, Waltham MA, Hangzhou, China, 1st edition, 1-1, 2012.
- [5] K.P. Jankowski. *Inverse dynamics control in Robotics applications*. Trafford Publishing, Victoria, British Columbia, Canada, 1st edition, 67, 2004.
- [6] P. Lambrechts, M. Boerlage and M. Steinbuch. Trajectory planning and feedforward design for high performance motion systems. *Proceedings of the American Control Conference*, 5:4637-4642, June 2004.
- [7] R. Kelly and J. Moreno. Learning pid structures in an introductory course of automatic control. *IEEE Transactions on Education*, 44(4):373-376, November 2001.
- [8] Catalin Alexandru, Claudiu Pozna, N.E. Mastorakis, O. Martin and X.J. Zheng. A mechatronic model for the dynamic analysis of the windshield wiper linkages used for the motor vehicles. *Proceedings of the 2nd WSEAS International Conference on Engineering Mechanics, Structures and Engineering Geology (EMESG)*, 60-65, 2009.
- [9] Farzad Cheraghpour, Masoud Vaezi, Hesam E. Shoori Jazeh and S.A.A. Moosavian. Dynamic modeling and kinematic simulation of stäubli tx40 robot using matlab/adams co-simulation. *Proceedings of the IEEE International Conference on Mechatronics*, 386-391, April 2011.
- [10] Yongxian Liu, Chunxia Zhu and Zhao Jinfu. Research on co-simulation of rigid-flexible coupling system of parallel robot. *Proceedings of the IEEE International Conference on Industrial Technology (ICIT), IEEE Computer Society*, 1-6, 2009.
- [11] M. Benosman, G.L. Vey. Control of flexible manipulators: a survey. *Robotica*, 22:533-545, 2004.

- [12] S.K. Dwivedy and P. Eberhard. Dynamic analysis of flexible manipulators, a literature review. *Mechanism and Machine Theory*, 41(7):749-777, 2006.
- [13] W.J. Book. Analysis of massless elastic chains with servo controlled joints. *ASME Journal of Dynamic Systems, Measurement, and Control*, 101:187-192, 1979.
- [14] W.J. Book. Recursive Lagrangian dynamics of flexible manipulator arms. *The International Journal of Robotics Research*, 3(3):87-101, 1984.
- [15] R.H. Cannon Jr. and E. Schmitz. Precise control of flexible manipulators. *Proceedings of First International Symposium on Robotics Research*, MIT Press, Cambridge, Massachusetts, USA, 841-861, 1984.
- [16] T. Sakawa, F. Matsuno and S. Fukushima. Modelling and feedback control of a flexible arm. *Journal of Robotic Systems*, 2:453-472, 1985.
- [17] G.B. Yang and M. Donath. Dynamic model of a one link manipulator with both structural and joint flexibility. *Proceedings of IEEE International Conference on Robotics and Automation*, 6(3):454-461, 1988.
- [18] A. De Luca and B. Siciliano. Trajectory control of non-linear one link flexible arm. *International Journal of Control*, 50:1699-1715, 1989.
- [19] F. Bellezze, L. Lanari and G. Ulivi. Exact modeling of the flexible slewing link. *Proceedings of IEEE 1990 International Conference on Robotics and Automation*, 2:734-739, 1990.
- [20] H. Asada, J.-H. Park and S. Rai. A control-configured flexible arm: Integrated structure/control design. *Proceedings of IEEE Conference on Robotics and Automation*, Sacramento, California, USA, 3:2356-2362, 1991.
- [21] W.K. Belvin and K.C. Park. Structural tailoring and feedback control synthesis: An interdisciplinary approach. *Journal of Guidance and Control*, 13(3):424-429, 1990.
- [22] F.-Y. Wang, P.X. Zhou and P. Lever. Dynamic effects of rotary inertia and shear deformation on flexible manipulators. *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, Beijing, China, 2315-2320, 1996.
- [23] F.-Y. Wang and G.G. Guan. Influences of rotary inertia, shear deformation and tip load on vibrations of flexible manipulators. *Journal of Sound and Vibrations*, 171(4):433-452, 1994.
- [24] F.-Y. Wang and J.L. Russell. Optimum shape construction of flexible manipulators with total weight constraint. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(4):605-614, 1994.
- [25] F.-Y. Wang and J.L. Russell. A new approach to optimum flexible link design. *Proceedings of IEEE International Conference on Robotics and Automation*, Nagoya, Japan, 931-936, 1995.
- [26] H. Geniele, R.V. Patel and K. Khorasani. End-point control of a flexible-link manipulator: Theory and experiments. *IEEE Transactions on Control System Technology*, 5(6):556-570, 1997.

-
- [27] S.K. Madhavan and S.N. Singh. Inverse trajectory control and zero dynamic sensitivity of an elastic manipulator. *International Journal of Robotics and Automation*, 6(4):179-192, 1991.
- [28] D. Wang and M. Vidyasagar. Modelling a class of multi-link manipulators with the last link flexible. *IEEE Transactions on Robotics and Automation*, 8(1):33-41, 1992.
- [29] Y. Aoustin and C. Chevallereau. The singular perturbation control of a two-flexible link robot. *Proceedings of IEEE Conference on Robotics and Automation*, Atlanta, Georgia, USA, 737-742, 1993.
- [30] L.B. Gutierrez, F.L. Lewis and J.A. Lowe. Implementation of a neural network tracking controller for a single flexible link: Comparison with PD and PID controller. *IEEE Transactions on Industrial Electronics*, 45(2):307-318, 1998.
- [31] K. Lenz, H. Ozbay, A. Tannenbaum, J. Turi and B. Morton. Frequency domain analysis and robust control design for an ideal flexible beam. *Automatica*, 27:947-961, 1991.
- [32] A. Tchernychev, A. Sideris and J. Yu. Constrained HN control of an experimental flexible link. *Transactions of the ASME Journal*, 119(2):206-211, 1997.
- [33] G. Zhu, T. Lee and S. Ge. Tip tracking control of a single-link flexible robot: A backstepping approach. *Dynamics and Control*, 7:341-360, 1997.
- [34] A. Jnifene and A. Fahim. A computed torque/time delay approach to the endpoint control of a one-link flexible manipulator. *Dynamics and Control*, 7:177-189, 1997.
- [35] B. Liang, Y. Xu and M. Bergernan. Mapping a space manipulator to a dynamically equivalent manipulator. *Journal of Dynamic Systems, Measurement, and Control*, 120(3):1-7, 1998.
- [36] M. Ghanekar, D.W.L. Wang and R. Heppler. Scaling laws for linear controller of flexible link manipulators characterized by nondimensional groups. *IEEE Transactions on Robotics and Automation*, 13(1):117-127, 1997.
- [37] P. Tang and S. Lu. A video signal processing technique for vibration measurement of flexible beam. *International Journal of Robotics Research*, 11(6):1610-1622, 1992.
- [38] G.G. Hastings and W.J. Book. A linear dynamic model for flexible robotic manipulators. *IEEE Control System Magazine*, 7(1):61-64, 1987.
- [39] T. Fukuda. Control of flexible robot arm. *Bulletin of the JSME*, 29(250):1269-1273, 1986.
- [40] N.G. Chalhoub and A.G. Ulsoy. Control of a flexible robot arm: Experiments and theoretical results. *Journal of Dynamic System Measurement Control*, 109(4):299-309, 1987.
- [41] A. Macchelli, C. Melchiorri and S. Stramigioli. Port-based modeling of a flexible link. *IEEE Transactions on Robotics*, 23(2):650-660, 2007.
- [42] Y. Xu and E. Ritz. Vision-based flexible beam tip point control. *IEEE Transactions on Control Systems Technology*, 17(5):1220-1227, 2009.

- [43] L. Bascetta and P. Rocco. Two-time scale visual servoing of eye-in-hand flexible manipulators. *IEEE Transaction on Robotics*, 22(4):818-830, 2006.
- [44] J. Becedas, J.R. Trapero, V. Feliu and H. Sira-Ramirez. Adaptive controller for single-link flexible manipulators based on algebraic identification and generalized proportional integral control. *IEEE Transactions on Systems, Man, and Cybernetics, Part B Cybernetics*, 39(3):735-751, 2009.
- [45] M. Hosaka and T. Murakami. Vibration control of flexible arm by multiple observer structure. *Electrical Engineering in Japan*, 154(2):68-75, 2006.
- [46] A. Chatterjee, R. Chatterjee, F. Matsuno and T. Endo. Augmented stable fuzzy control for flexible robotic arm using LMI approach and neuro-fuzzy state space modeling. *IEEE Transactions on Industrial Electronics*, 55(3):1256-1270, 2008.
- [47] M. Dogan and Y. Istefanopulos. Optimal nonlinear controller design for flexible robot manipulators with adaptive internal model. *IET Control Theory and Applications*, 1(3):770-778, 2007.
- [48] C.A. Monje, F. Ramos, V. Feliu and B.M. Vinagre. Tip position control of a lightweight flexible manipulator using a fractional order controller. *IET Control Theory and Applications*, 1(5):1451-1460, 2007.
- [49] W.J. O'Connor. Wave-based analysis and control of lump-modeled flexible robots. *IEEE Transactions on Robotics*, 23(2):342-352, 2007.
- [50] X. Wang and D. Chen. Output tracking control of a one-link flexible manipulator via causal inversion. *IEEE Transactions on Control Systems Technology*, 14(1):141-148, 2006.
- [51] S.C.P. Gomes, V. Santos da Rosa and B. de Carvalho Albertini. Active control to flexible manipulators. *IEEE/ASME Transactions on Mechatronics*, 11(1):75-83, 2006.
- [52] F.-Y. Wang. Optimum design of vibrating cantilevers: A classical problem revisited. *Journal of Optimization Theory and Applications*, 84(3):635-652, 1995.
- [53] F.-Y. Wang and J.L. Russell. Optimum shape construction of flexible manipulators with tip loads. *Proceedings of IEEE International Conference on Decision and Control*, Tucson, Arizona, USA, 1:311-316, 1992.
- [54] F.-Y. Wang. On the extremal fundamental frequencies of one-link flexible manipulators. *International Journal of Robotics Research*, 13(2):162-170, 1994.
- [55] F.-Y. Wang and J.L. Russell. Minimum-weight robot arm for a specified fundamental frequency. *Proceedings of IEEE Conference on Robotics and Automation*, Atlanta, USA, 490-495, 1993.
- [56] F.Y. Wang. *Optimum design of flexible manipulators: Integration of control and construction* [Working Paper], 40-91, SIE Department, University of Arizona, Tucson, Arizona, USA, 1991.
- [57] L.M. Sweet and M.C. Good. Redefinition of the robot motion control problem: effects of plant dynamics, drive system constraints, and user requirements. *Proceedings of IEEE Conference on Decision and Control*, Las Vegas, Nevada, USA, 724-791, December 1984.

-
- [58] E.I. Rivin. Effective rigidity of robot structure: analysis and enhancement. *Proceedings of American Control Conference*, Boston, USA, 381-382, June 1985.
- [59] G. Widmann and S. Ahmad. Control of industrial robots with flexible joints. *Proceedings of IEEE Conference on Robotics and Automation*, Raleigh, North Carolina, USA, 1561-1566, May 1987.
- [60] W.J. Book. New concepts in lightweight arms. *Proceedings of the 2nd International Symposium of Robotics Research*, Kyoto, Japan, 203-205, August 1985.
- [61] O.C. Zienkiewicz. *The finite element method*, Mc Graw-Hill, New York, USA, 1979.
- [62] P.B. Usoro, R. Nadira and S.S. Mahil. A finite element/Lagrange approach to modeling lightweight flexible manipulators. *ASME Journal of Dynamic Systems, Measurement, and Control*, 108(9):198-205, September 1986.
- [63] W.C. Hurty and M.F. Rubinstein. *Dynamics of structures*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, USA, 1964.
- [64] A.A. Shabana. *Dynamics of multibody systems*. John Wiley & Sons Inc., New York, USA, 1989.
- [65] R.J. Guyan. Reduction of stiffness and mass matrices. *A/AA Journal*, 9(2):980, May 1965.
- [66] S. Moberg and S. Hanssen. *Inverse Dynamics of Flexible Manipulators* [PhD Dissertation], Linköpings universitet, Reglerteknik, Institutionen för systemteknik, Tekniska högskolan, 2010.
- [67] S.S. Ge, T.H. Lee and G. Zhu. A new lumping method of a flexible manipulator. *Proceedings of the American Control Conference*, 3:1412-1416, 1997.
- [68] Rasmus B. B. Olesen. *Controlling Of Industrial Robotic Manipulators Operating With Flexible Tools* [Master's thesis report], Aalborg University, 2012.
- [69] Z. Yi, X. Min-min, Q. Jin-yi and Z. Hu. Research on co-simulation using ADAMS and MATLAB for automobile active suspension system. *International Conference on Computer Application and System Modeling (ICCASM)*, 14:366-370, 2010.
- [70] Z. Affi and L. Romdhane. ADAMS/SIMULINK interface for Dynamic Modeling and Control of Closed Loop Mechanisms. *Proceedings of the 7th WSEAS International Conference on automatic control, modeling and simulation*, Prague, Czech Republic, 353-356, 2006.
- [71] Z.Z. Naing and M.P. Al-Mamun. Integrated ADAMS/MATLAB environment for design of an autonomous single wheel robot, Industrial Electronics. *Proceedings of IECON 35th Annual Conference of IEEE*, 2253-2258, 2009.
- [72] R. Rahmfeld. *Modelling of Hydraulic-Mechanical Systems Co-Simulation between ADAMS and MATLAB/Simulink*, Hamburg, Germany, February 2003.
- [73] MSC Company. *Getting Stared Using ADAMS/Control*, 2003.

A

ADAMS-MATLAB/Simulink Interface and Performing Co-Simulations

ADAMS is a dynamic modelling software package that has the capability of simulating any mechanical system's kinematics and kinetics behaviour, resulting in a dynamic behaviour for the simulated system with an acceptable error in comparison to the real models. ADAMS/View is a very useful tool within ADAMS for exact modelling and accurate calculation of both dynamical and kinematical parameters (for example inertial couplings of a robotic manipulator system, bending of its flexible links, . . . etc.) of mechanical systems. ADAMS/Flex is another tool within the ADAMS modelling and simulation environment, that can be used alone for modelling and simulation of flexible structures, or can be integrated as an extension inside ADAMS/View for modelling of the flexible components of a mechanical system [8]. On the other hand, MATLAB, together with its companion Simulink simulation environment, is a well-known software package and powerful tool for efficiently simulating control algorithms and applying control commands to (for e.g., a robotic manipulator model). Hence, interfacing ADAMS and MATLAB can be very useful for having an efficient simulation package that can be used to design and test suggested modelling techniques with control strategies, therefore the ability to establish and execute a co-simulation between ADAMS and MATLAB is noticeably useful and encouraged [9].

To carry out a co-simulation between ADAMS and MATLAB, acceptable formats for inputs and outputs for the both programs should be defined. The co-simulation objective is then to provide a communication channel that guarantees that any change is mutually exchanged back and forth between the two programs affecting both ADAMS and MATLAB simultaneously [69]. In other words, as long as this interface is established and active, ADAMS outputs are matched with MATLAB inputs and the other way around.

To provide a simulation that enables ADMAS to understand the exported output from MATLAB, a "control" plug-in inside ADAMS needs to be activated via the "Plug-in Manager" wizard in ADAMS, beside defining the robotic manipulator as a plant. After activation, a new window appears for the determination of plant and its inputs and outputs. Afterwards, ADAMS saves dynamic model of the robotic manipulator motion as a complex set of seven matrices that can be sent to MAT-

LAB as a Simulink block called *Adams-sub*. The first four are A , B , C , D matrices of state-space representation of the “linearised” plant model. The fifth and sixth matrices are associated with predefined inputs and outputs. And the last one includes information about state variables of the “linearised” plant [70,71]. By having *Adams-sub* block in SIMULINK environment, the robot model is suitable for control and motion simulation as a well-defined system in MATLAB.

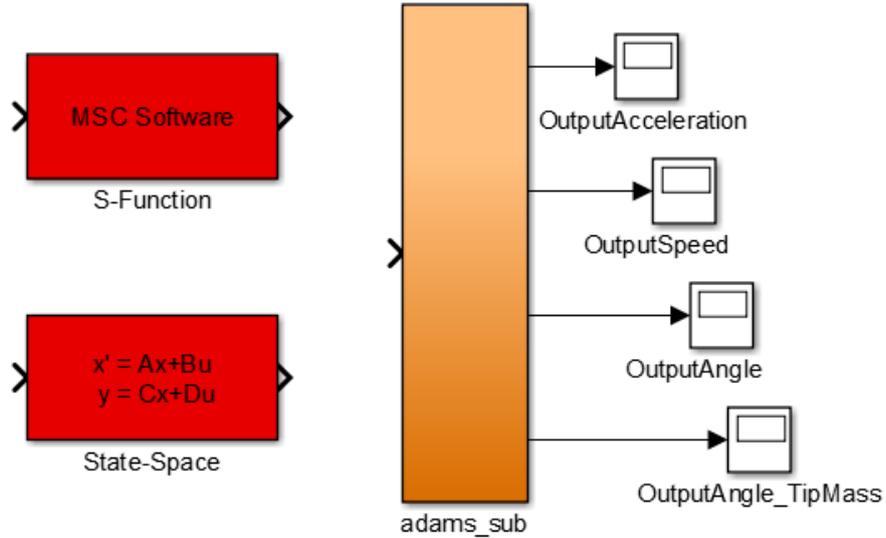
For our thesis tasks specifically, in order to obtain the measurements for angular position, angular speed and angular acceleration of the end-effector at the flexible link’s tip (and sometimes the gearbox’s angular position too), in addition to the torque that move the joint motor, a control system was developed using ADAMS/Controls and MATLAB/Simulink. ADAMS/Controls is an extra plug-in that can be activated inside ADAMS/View. It allows integrating motion simulation and controlling system design in the virtual model. It also allows connecting the ADAMS model to block diagrams that are developed with the control application (MATLAB/Simulink in our case).

To keep it simple and yet precise, the process to combine ADAMS with MATLAB; providing a co-simulation of the “linearised” robotic manipulator system under study involves the following concrete steps [10, 72, 73]:

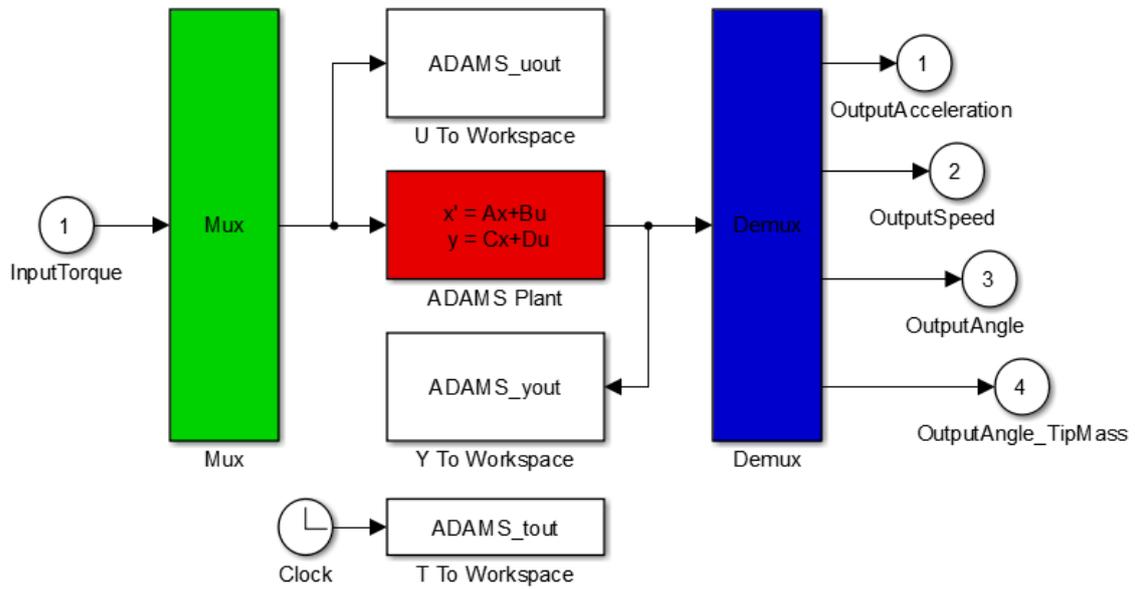
1. Build the mechanical model of the (sub-)system to be controlled in ADAMS/View (the plant model), including all the necessary geometry, forces and constraints.
2. Transform rigid parts that are required to be compliant into their flexible counterpart representations by utilising ADAMS/Flex. To do so, right-click on the specific part of the plant model and choose “Make Flexible”, then define the number of flexible modes to be exported, the attachments nodes, ... etc.
3. Define the required actuation signals (torques and forces) and sensing (measures) for the quantities that are to be imported from and exported to, respectively, MATLAB for control and simulation purposes.
4. To import mechanical model created in ADAMS/View into ADAMS/Controls, a CMD file has to be exported by ADAMS/View. Next point is to deactivate the motions in the control system. The motions will be replaced by torques and forces in a next step.
5. Define the input and output variables of the ADAMS developed plant model: the inputs represent the variables to be exported to MATLAB (measurements/sensors), whereas the outputs are the variables to be imported from MATLAB (torques and forces) at each time sample of the communication channel. For the ADAMS outputs, simply state variables can be used by definition and the state variable can be simply set equal to a measure which was defined before. A useful name for the state variable should be given so that it can be simply recognised in MATLAB. For the ADAMS inputs, also state variables have to be created and must be initialised to zero upon definition. Later, the corresponding ADAMS input is edited and its value is set equal to the *VARVAL* function of the previously defined state variable: $T_{in} = VARVAL(T_{MATLAB})$; where T_{in} is the input torque, and T_{MATLAB}

is the corresponding state variable equal to zero. $VARVAL(variable)$ is the ADAMS function that returns the value of the given variable.

6. Pick a useful name for the plant model to be generated by ADAMS/Controls. Then use the variable browser to insert the input and output state variables, respecting the order in which you want them to be imported in the block diagram exported to MATLAB/Simulink. An important issue to take care of here is making sure that the units are uniform by using unit conversions as needed (note: MATLAB uses SI units). It should be made sure that latest at this point that ADAMS and MATLAB are working in the same file directory. The export process will create some files in the common ADAMS-MATLAB working directory to be used for information exchange between the two software packages.
7. Close the ADAMS/Controls window and start the MATLAB/Simulink, where the controller scheme (block diagram) is built, and include the ADAMS plant block into the MATLAB/Simulink control system simply by copy and paste. The *m-file* created by ADAMS has to be started first. A good practice then is to merge it together with the *m-file* of the control system, so that before a simulation only one common *m-file* has to be executed. The ADAMS/Controls block diagram of the robotic manipulator FEM flexible link with the gearbox inertia in MATLAB/Simulink for co-simulation is showed in Figure A.1, and Subfigure A.1a is obtained by double clicking the $adams_{sub}$ in Subfigure A.1b. Then the input and output connections have to be realised. $Adams_{uout}$, $Adams_{yout}$, $Adams_{tout}$ are the input and output variables and the simulation time, respectively. The block ADAMS Plant is the encapsulated dynamic model of the robotic manipulator FEM flexible link with the gearbox inertia.



(a) The encapsulated model generated by ADAMS and shown under Simulink



(b) Architecture of the *adams_{sub}* block

Figure A.1: The ADAMS/Controls block diagram of the FEM flexible robotic link with the gearbox inertia in MATLAB/Simulink.

8. Execute initialisations and set all the parameters used in the co-simulation (including calling of the plant model generated by ADAMS, to define the state-space matrices and inputs and outputs variables of the shared interface between ADAMS and MATLAB).
9. Call the ADAMS/Controls generated plant in MATLAB, the plant name should be entered in the “command window” so that the corresponding input and output information will appear. By executing *adams_sys* command, the blocks that contain information about the dynamic model of the robotic manipulator link with the gearbox inertia in MATLAB/Simulink environment, are loaded.
10. Perform the simulation of the combined mechanical model (including the ADAMS plant) and control system (developed in MATLAB/Simulink) in MATLAB. Here, a crucial thing is the choice of integration step size. It is then recommended to choose the *ODE45* integration method in MATLAB with variable step size when using continuous mode. The simulation precision can be defined among the relative accuracy and the absolute accuracy. These individual values have to be adjusted in case the simulation is problematic. It is recommended to set the relative tolerance to maximum while the absolute tolerance is set to auto (In our case, the relative tolerance is set to 10^{-9}). Additionally, the initial conditions for the *ADAMS Plant* block are all set to 0. When all the parameters are set, the co-simulation can be carried out.

B

Frequency Responses of Additional 2-Mass Models Used in Simulations

Figure B.1 shows the frequency response of the transfer function model in (2.2); with the following parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$ and $k_g = 32$.

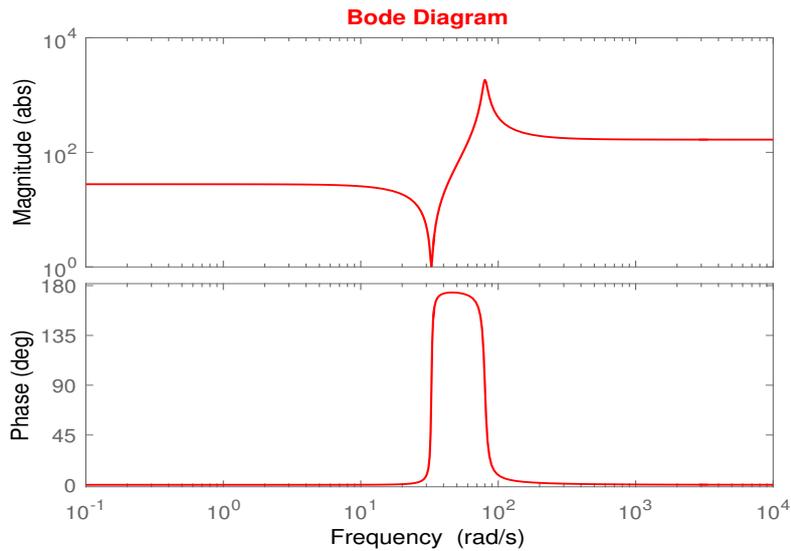


Figure B.1: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox (2-mass system with $k_g = 32$).

Figure B.2 shows the frequency response of the transfer function model in (2.2); with the following parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$ and $k_g = 45$.

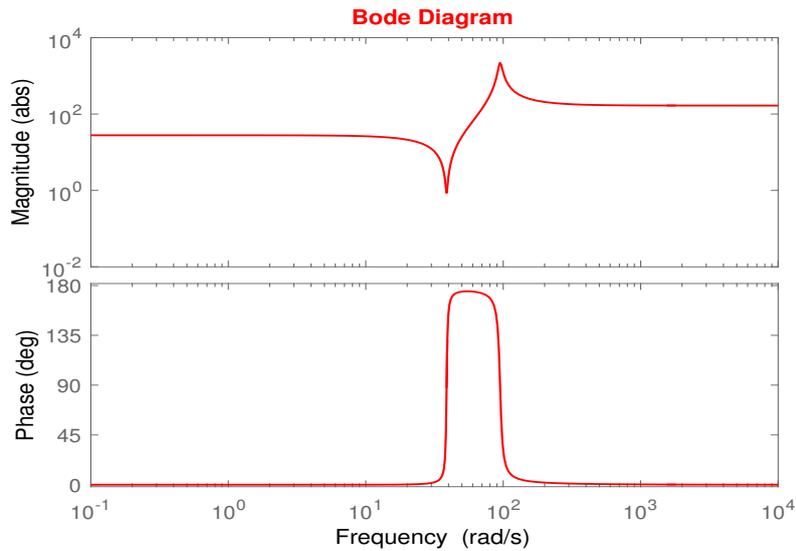


Figure B.2: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox (2-mass system with $k_g = 45$).

Figure B.3 shows the frequency response of the transfer function model in (2.2); with the following parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$ and $k_g = 65$.

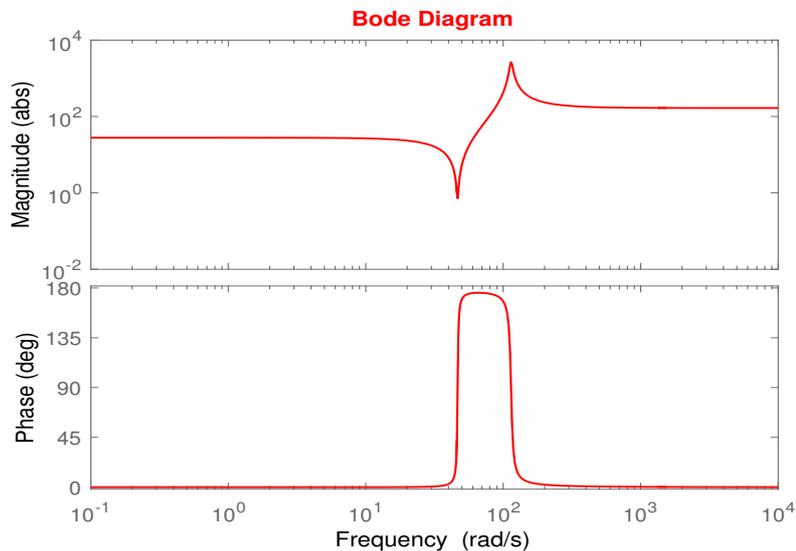


Figure B.3: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox and an inertia on the output of the gearbox (2-mass system with $k_g = 65$).

C

Frequency Responses of Additional 3-Mass Models Used in Simulations

Figure C.1 shows the frequency response of the transfer function model in (2.5); with the following parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 32$, $J_l = 60 \times 10^{-3}$, $d_l = 30 \times 10^{-3}$ and $k_l = 20$.

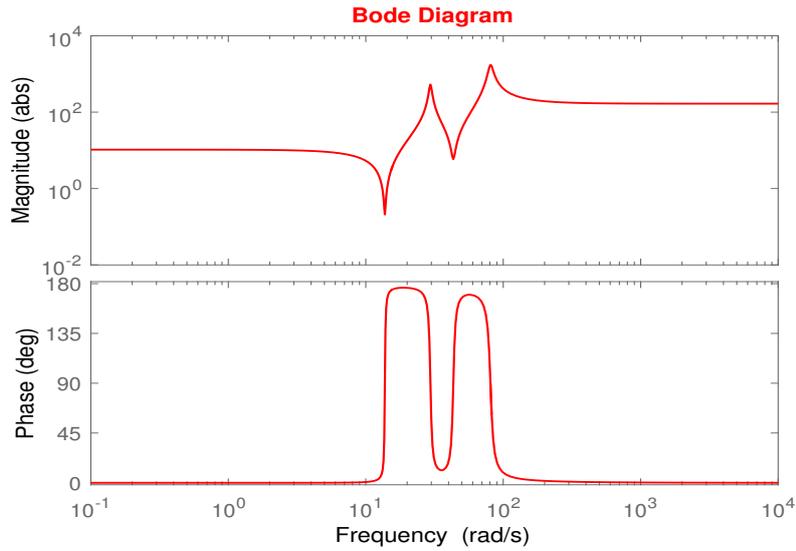


Figure C.1: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gear box and a flexible link (3-mass system with $k_g = 32$ and $k_l = 20$).

Figure C.2 shows the frequency response of the transfer function model in (2.5); with the following parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 45$, $J_l = 60 \times 10^{-3}$, $d_l = 30 \times 10^{-3}$ and $k_l = 20$.

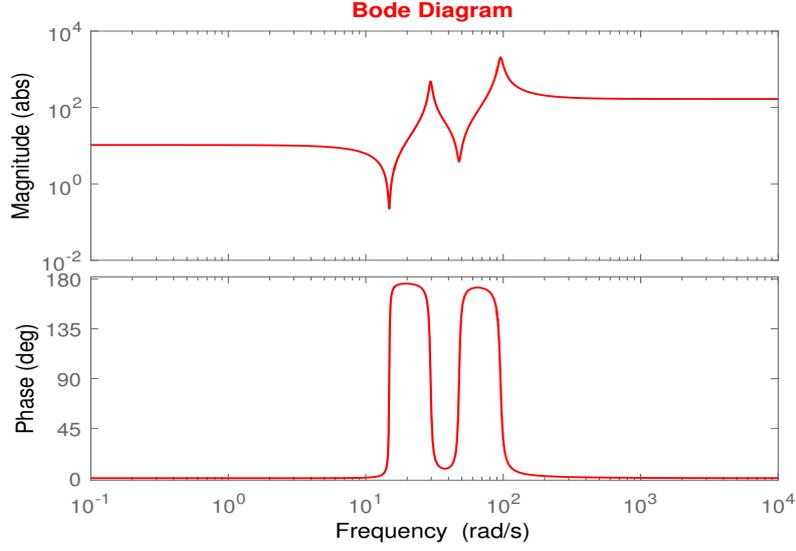


Figure C.2: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gear box and a flexible link (3-mass system with $k_g = 45$ and $k_l = 20$).

Figure C.3 shows the frequency response of the transfer function model in (2.5); with the following parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 65$, $J_l = 60 \times 10^{-3}$, $d_l = 30 \times 10^{-3}$ and $k_l = 20$.

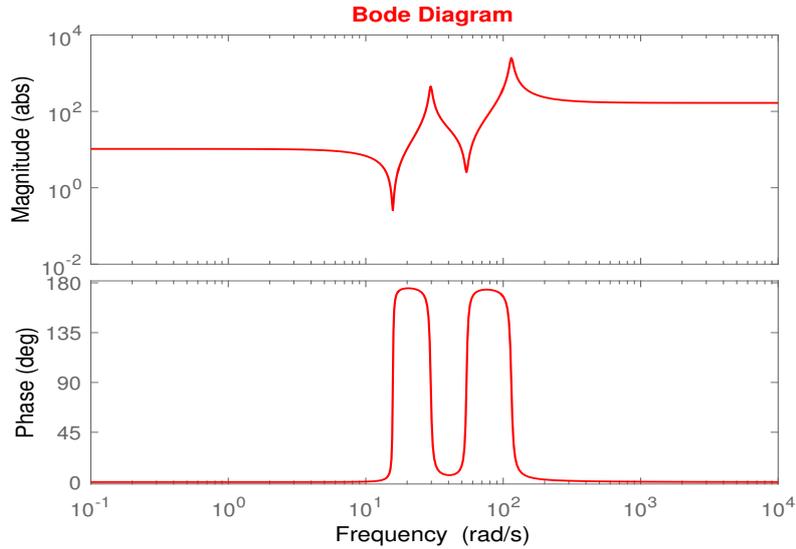


Figure C.3: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gear box and a flexible link (3-mass system with $k_g = 65$ and $k_l = 20$).

D

Frequency Responses of Additional Multi-Mass (3-Mass) Models Used in Simulations

Figure D.1 shows the identified (using a chirp input signal) frequency response of the multi-mass link and gearbox resulting in a 3-mass system; with the following set of discrete model parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 65$, $J_l = 60 \times 10^{-3}$, $d_l = 30 \times 10^{-3}$, $k_l = 20$ and $n = 5$.

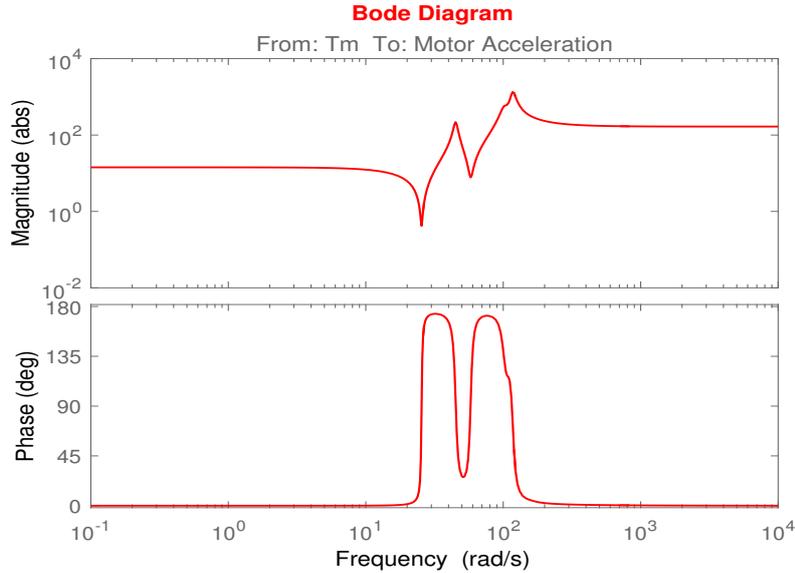


Figure D.1: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case with $n = 5$).

Figure D.2 shows the identified (using a chirp input signal) frequency response of the multi-mass link and gearbox resulting in a 3-mass system; with the following set of discrete model parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 65$, $J_l = 60 \times 10^{-3}$, $d_l = 30 \times 10^{-3}$, $k_l = 20$ and $n = 10$.

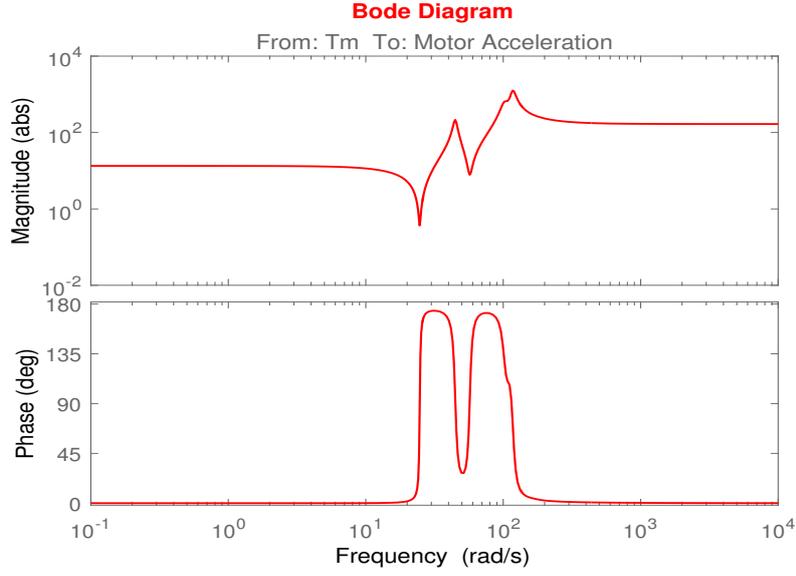


Figure D.2: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case with $n = 10$).

Figure D.3 shows the identified (using a chirp input signal) frequency response of the multi-mass link and gearbox resulting in a 3-mass system; with the following set of discrete model parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 65$, $J_l = 60 \times 10^{-3}$, $d_l = 30 \times 10^{-3}$, $k_l = 20$ and $n = 15$.

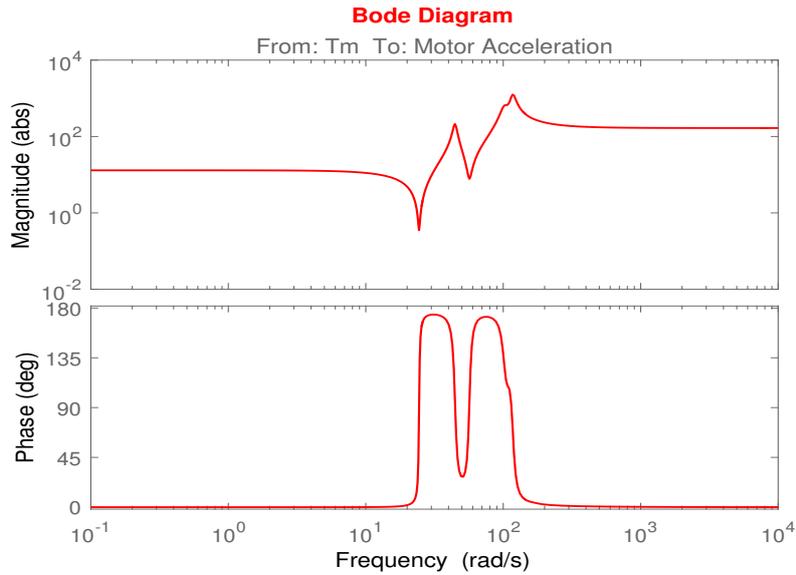


Figure D.3: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case with $n = 15$).

Figure D.4 shows a visual illustrative comparison of the identified (using a chirp input signal) frequency responses of the multi-mass link and gearbox systems resulting in a 3-mass system, explaining the effect of the number of lumped elements N on the frequency response. All these frequency responses are obtained with the following set of discrete model parameters: $J_m = 6 \times 10^{-3}$, $J_g = 30 \times 10^{-3}$, $d_g = 30 \times 10^{-3}$, $k_g = 65$, $J_l = 60^{-3}$, $d_l = 30 \times 10^{-3}$ and $k_l = 20$ [$n = 5, 10$ and 15].

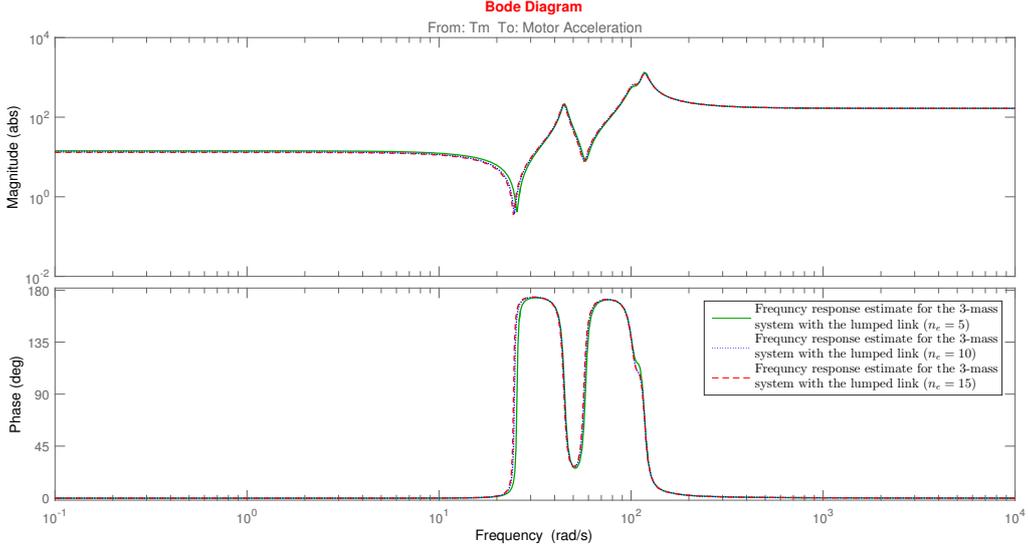


Figure D.4: Frequency response for transfer function from motor torque (τ_m) to motor acceleration ($\ddot{\varphi}_m$) of a flexible manipulator with a motor, a flexible gearbox, an inertia on the output of the gearbox and lumped (multi-mass) model of a flexible link with a big-inertia tip-mass load (3-mass system case with [$n = 5, 10$ and 15]).

E

Steps for Mode/Anti-Mode Matching in the Frequency Plane Using Inverse Dynamic Models

The required formulas for frequency response matching are based on the modes and anti-modes expressions and the inverse dynamics equations given in the main thesis text (see Subsections 2.2.1, 2.2.2 and 2.2.3, in addition to Section 3.4). Here, the precise steps for mode/anti-mode matching in the frequency plane using 2-mass and 3-mass inverse dynamic models are provided.

E.1 Matching procedure of a 2-mass (3-mass) plant mode(s) and anti-mode(s) using a 2-mass inverse dynamic model

1. Enter the plant's estimated frequency response information:
 - The anti-resonance frequency (in case of a 2-mass plant), or either the lower or higher anti-resonance frequency (in case of a 3-mass plant) $[\omega_z]$.
 - The resonance frequency (in case of a 2-mass plant), or either the lower or higher resonance frequency (in case of a 3-mass plant) $[\omega_p]$.
 - The low frequency region (or DC) gain $[H_{DC}]$.
2. Pick a reasonable initial value for the motor inertia of the 2-mass inverse dynamic model $[J_{m2-mass}]$. A good guess is then the plant model nominal value $[J_m]$. Hence:

$$J_{m2-mass} = J_m$$

3. Calculate the value of the gearbox inertia of the 2-mass inverse dynamic model $[J_{g2-mass}]$ using the relation:

$$J_{g2-mass} = \frac{1}{H_{DC}} - J_{m2-mass}$$

4. Calculate the value of the gearbox stiffness of the 2-mass inverse dynamic model $[K_{g2-mass}]$ using the relation:

$$k_{g2-mass} = J_{g2-mass} \omega_z^2$$

5. Refine the guess for $[J_{m2-mass}]$:

$$J_{m2-mass} = k_{g2-mass} / \left(\omega_p^2 - \frac{k_{g2-mass}}{J_{g2-mass}} \right)$$

6. Tune the damping value of the gearbox damping of the 2-mass inverse dynamic model $[d_{g2-mass}]$ until the frequency response of the matched 2-mass inverse dynamic model matches/gets as closest as possible to that of the plant model (in case of a 2-mass plant) or best fit the lower part of it (in case of a 3-mass plant) . A good initial guess to start with is then the plant model nominal value $[d_g]$. Hence:

$$d_{g2-mass} = d_g$$

E.2 Matching procedure of a 3-mass plant modes and anti-modes using a 3-mass inverse dynamic model

1. Enter the plant's estimated frequency response information:
 - The lower and the higher anti-resonance frequencies $[\omega_{z1}, \omega_{z2}]$.
 - The lower and the higher resonance frequencies $[\omega_{p1}, \omega_{p2}]$.
 - The low frequency region (or DC) gain $[H_{DC}]$.
2. Pick a reasonable initial value for the motor inertia of the 3-mass inverse dynamic model $[J_{m3-mass}]$. A good guess is then the plant model nominal value $[J_m]$. Hence:

$$J_{m3-mass} = J_m$$

3. Solve the four equations that give the resonance and anti-resonance frequencies for the inverse dynamic model (given by (2.6)) simultaneously after substituting the value of $[J_{m3-mass}]$ and equating them to the numerical values entered above. This step can be done using a symbolic solver, for example the one integrated in Matlab. As a result, the values of the gearbox inertia $[J_{g3-mass}]$, the gearbox stiffness $[k_{g3-mass}]$, the link inertia $[J_{l3-mass}]$ and the link stiffness $[k_{l3-mass}]$ of the 3-mass inverse dynamic model are obtained.
4. Refine the guess for $[J_{m3-mass}]$:

$$J_{m3-mass} = \frac{1}{H_{DC}} - J_{g3-mass} - J_{l3-mass}$$

5. Tune the values of the gearbox damping and the link damping of the 2-mass inverse dynamic model $[d_{g3-mass}, d_{l3-mass}]$ until the frequency response of the matched 3-mass inverse dynamic model matches/gets as closest as possible to that of the plant model (best fit) . A good initial guess to start with is then the plant model nominal values $[d_g, d_l]$. Hence:

$$d_{g3-mass} = d_g$$

$$d_{l3-mass} = d_l$$

F

Detailed Results and Comparisons

In this Appendix, the detailed results obtained after implementing each of the fore-mentioned control and compensation strategies are provided, together with some comparative bar charts for the results of each plant model used. The plant model complexity together with the controller complexity were stepwise increased and each of the results are noted and also compared with the previous ones.

F.1 A 2-Mass Plant System

F.1.1 With A Simple Servo PID Controller Results

Figure F.1 and Table F.1 illustrate both qualitatively and quantitatively, respectively, the flexible gearbox tracking performance of the 2-mass plant system whose frequency response is shown in Figure 2.7, when it is controlled by a simple servo PID controller only.

$\min(\varphi_{gref} - \varphi_g)$	$\max(\varphi_{gref} - \varphi_g)$	$\min(\dot{\varphi}_{gref} - \dot{\varphi}_g)$	$\max(\dot{\varphi}_{gref} - \dot{\varphi}_g)$	$\min(\ddot{\varphi}_{gref} - \ddot{\varphi}_g)$	$\max(\ddot{\varphi}_{gref} - \ddot{\varphi}_g)$
-0.019896	0.021330	-0.084104	0.075255	0.556559	0.547807

Table F.1: A 2-mass plant system tracking with a PID controller only results

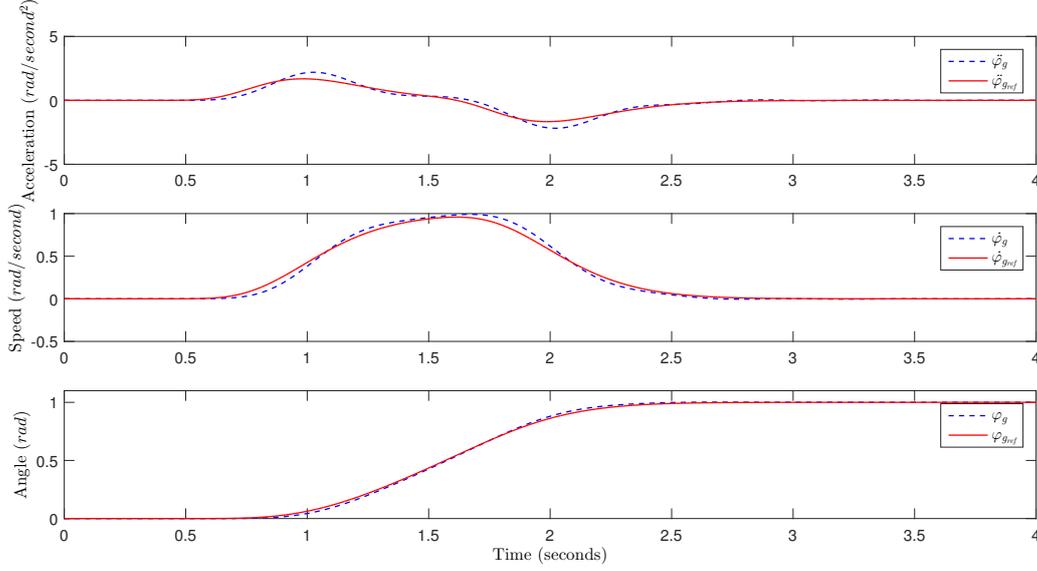


Figure F.1: Flexible Gearbox angular position, angular speed and angular acceleration reference tracking using PID controller only (a 2-mass plant system with $k_g = 5N/m$).

F.1.2 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.2 and Table F.2 illustrate both qualitatively and quantitatively, respectively, the flexible gearbox tracking performance of the 2-mass plant system whose frequency response is shown in Figure 2.7, when it is controlled by a 2-mass inverse dynamics with a simple servo PID controller.

$\min(\varphi_{gref} - \varphi_g)$	$\max(\varphi_{gref} - \varphi_g)$	$\min(\dot{\varphi}_{gref} - \dot{\varphi}_g)$	$\max(\dot{\varphi}_{gref} - \dot{\varphi}_g)$	$\min(\ddot{\varphi}_{gref} - \ddot{\varphi}_g)$	$\max(\ddot{\varphi}_{gref} - \ddot{\varphi}_g)$
-0.000105	0.004331	-0.008351	0.008135	-0.028569	0.0262159

Table F.2: A 2-mass plant system tracking with an identical 2-mass inverse dynamics and PID controller results.

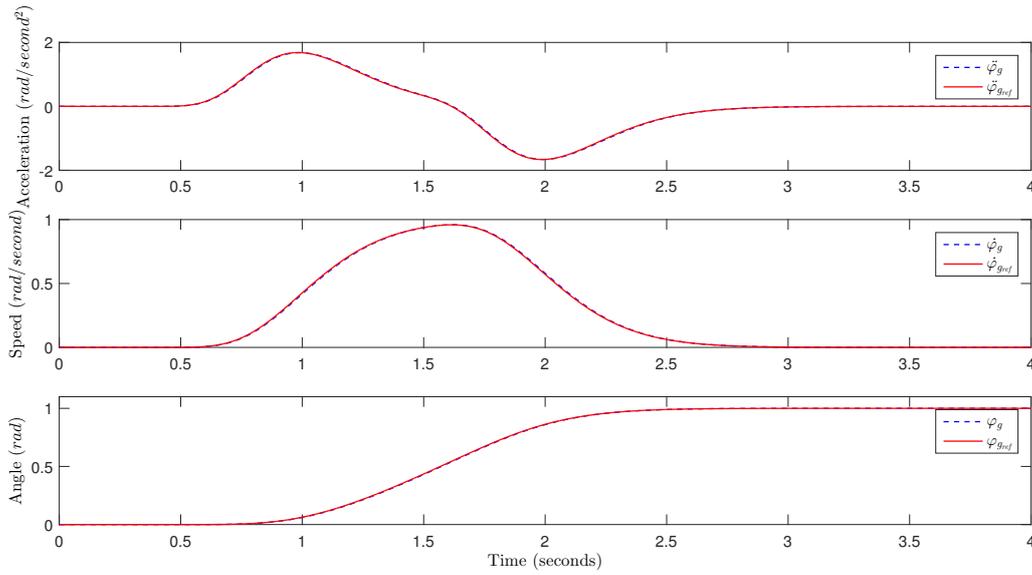


Figure F.2: Flexible Gearbox angular position, angular speed and angular acceleration reference tracking using identical 2-mass inverse dynamics with PID controller (a 2-mass plant system with $k_g = 5N/m$).

Figure F.3 shows a bar chart comparison of the 2-mass plant system gearbox's tracking errors using various controllers.

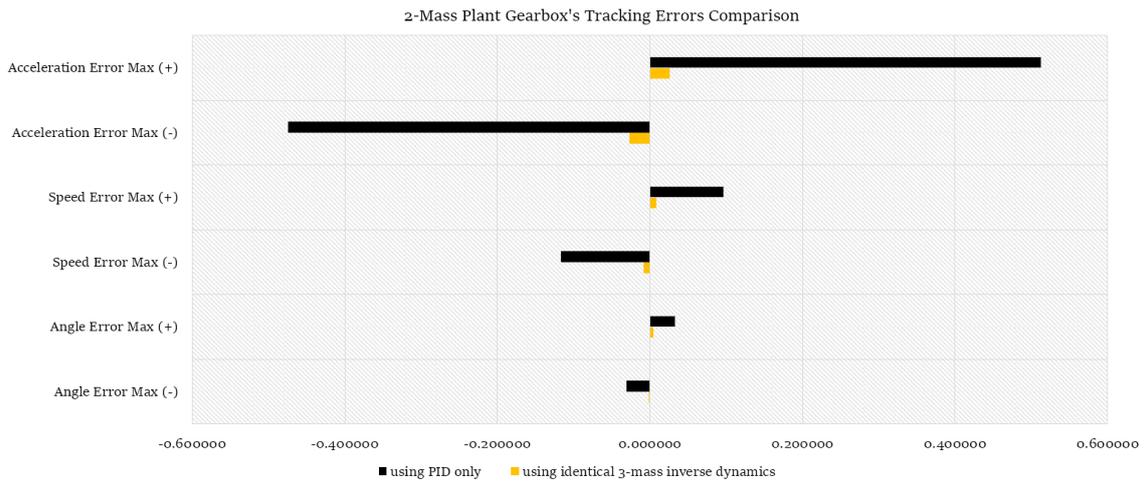


Figure F.3: Comparison of gearbox's angle, speed and acceleration tracking results obtained using various controllers for a 2-mass plant system.

F.2 A 3-Mass Plant System

F.2.1 With A Simple Servo PID Controller Results

Figure F.4 and Table F.4 illustrate both qualitatively and quantitatively, respectively, the flexible link's Tip tracking performance of the 3-mass plant system whose frequency response is shown in Figure 2.10, when it is controlled by a simple servo PID controller only.

$\min(\varphi_{lref} - \varphi_l)$	$\max(\varphi_{lref} - \varphi_l)$	$\min(\dot{\varphi}_{lref} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{lref} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{lref} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{lref} - \ddot{\varphi}_l)$
-0.107109	0.105483	-0.360957	0.316790	-1.772321	1.423061

Table F.3: A 3-mass plant system tracking with a PID controller only results

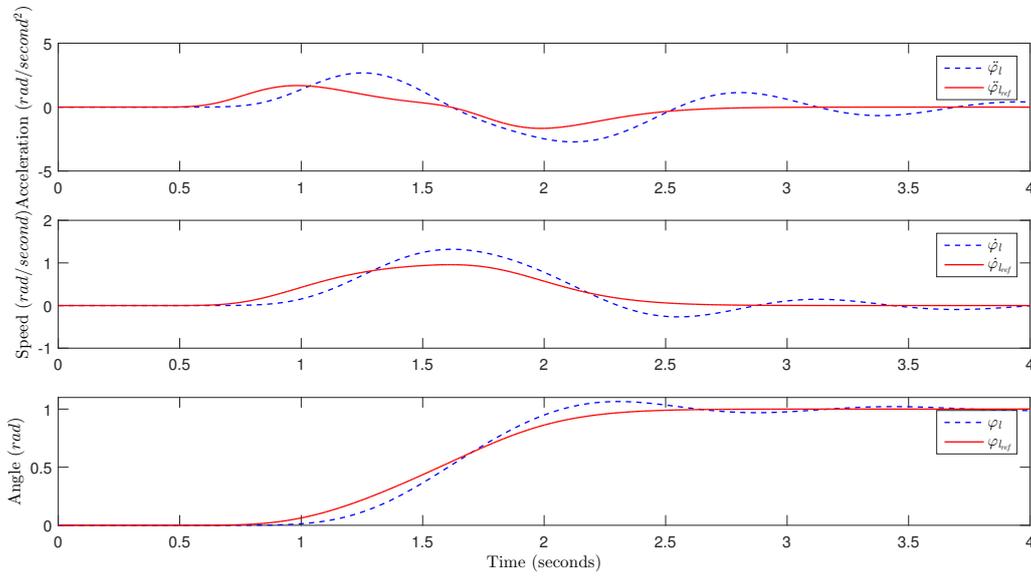


Figure F.4: Flexible link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (a 3-mass plant system with $k_g = 5N/m$ and $k_l = 5N/m$).

F.2.2 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.5 and Table F.5 illustrate both qualitatively and quantitatively, respectively, the flexible link's tip tracking performance of the 3-mass plant system whose frequency response is shown in Figure 2.10, when it is controlled by a 2-mass inverse dynamics (by frequency matching of the lower pair of mode and anti-mode) with a simple servo PID controller.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.002540	0.006514	-0.026986	0.024382	-0.143983	0.145532

Table F.4: A 3-mass plant system tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller results

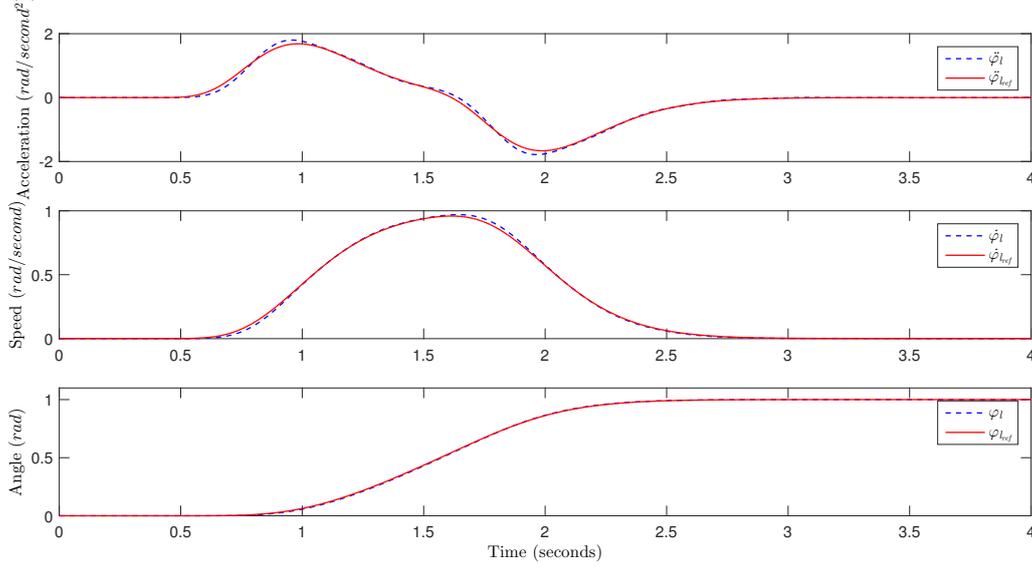


Figure F.5: Flexible link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [lower pair of mode and anti-mode matching] with PID controller (a 3-mass plant system with $k_g = k_l = 5N/m$).

Figure F.6 and Table F.6 illustrate both qualitatively and quantitatively, respectively, the flexible link's tip tracking performance of the 3-mass plant system whose frequency response is shown in Figure 2.10, when it is controlled by a 2-mass inverse dynamics (by frequency matching of the terminal pair of mode and anti-mode) with a simple servo PID controller.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.001705	0.007232	-0.023203	0.028303	-0.189479	0.155322

Table F.5: A 3-mass plant system tracking with a 2-mass inverse dynamics (terminal pair of mode and anti-mode matching) and PID controller results

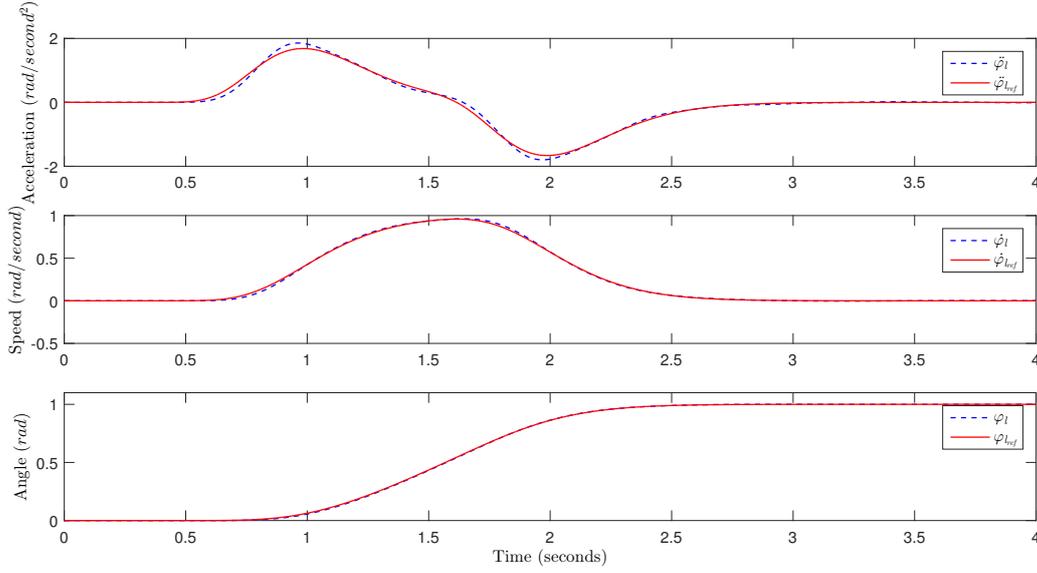


Figure F.6: Flexible link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [terminal pair of mode and anti-mode matching] with PID controller (a 3-mass plant system with $k_g = k_l = 5N/m$).

Figure F.7 and Table F.7 illustrate both qualitatively and quantitatively, respectively, the flexible link's tip tracking performance of the 3-mass plant system whose frequency response is shown in Figure 2.10, when it is controlled by a 2-mass inverse dynamics (by frequency matching of the higher pair of mode and anti-mode) with a simple servo PID controller.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.069053	0.071797	-0.256397	0.225889	-1.307012	0.962013

Table F.6: A 3-mass plant system tracking with a 2-mass inverse dynamics (higher pair of mode and anti-mode matching) and PID controller results

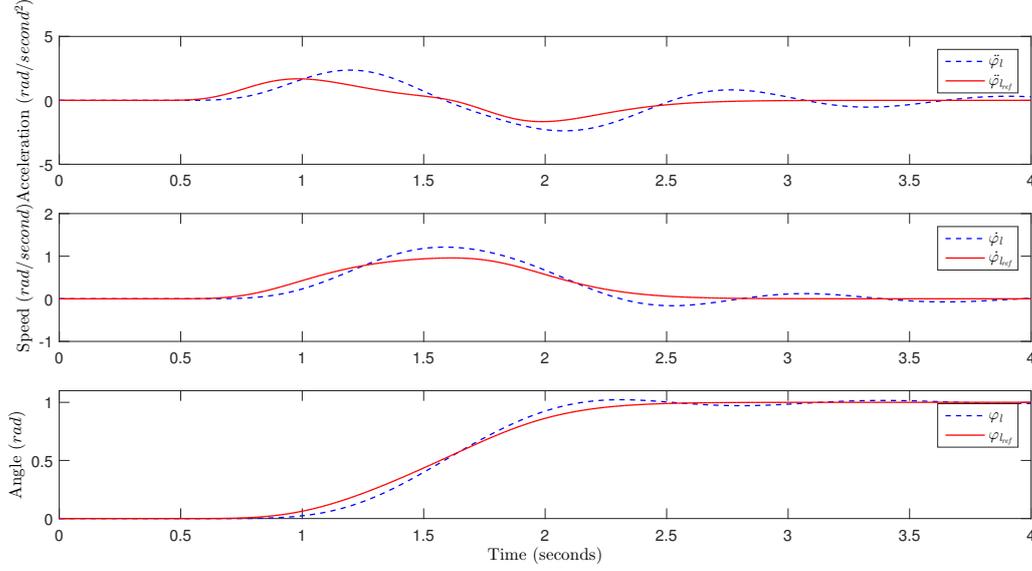


Figure F.7: Flexible link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [higher pair of mode and anti-mode matching] with PID controller (a 3-mass plant system with $k_g = k_l = 5N/m$).

F.2.3 With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.8 and Table F.8 illustrate both qualitatively and quantitatively, respectively, the flexible gearbox tracking performance of the 3-mass plant system whose frequency response is shown in Figure 2.10, when it is controlled by an identical 3-mass inverse dynamics with a simple servo PID controller.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.000142	0.004071	-0.006748	0.007138	-0.028290	0.031018

Table F.7: A 3-mass plant system tracking with an identical 3-mass inverse dynamics and PID controller results

F. Detailed Results and Comparisons

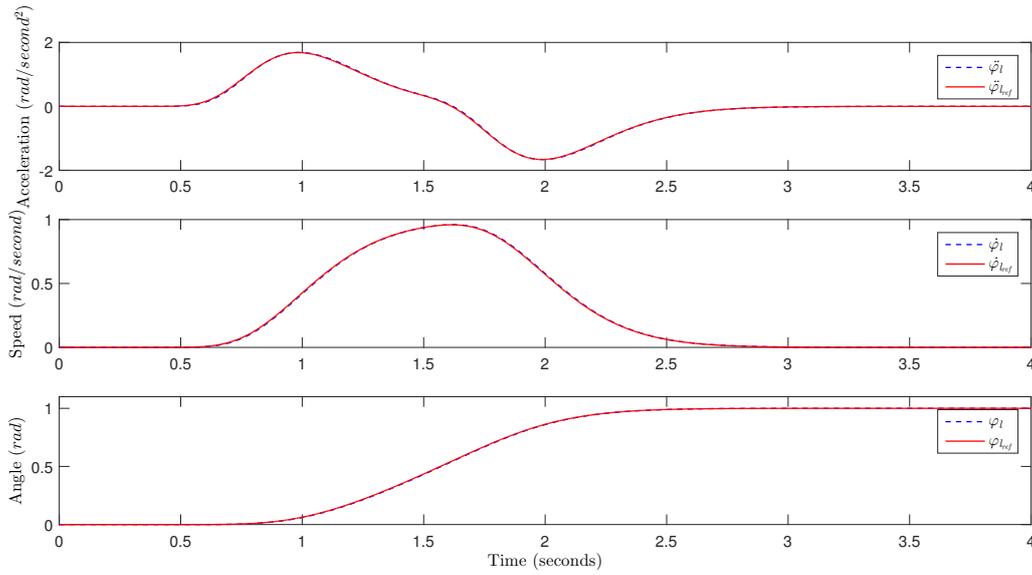


Figure F.8: Flexible link's tip angular position, angular speed and angular acceleration reference tracking using identical 3-mass inverse dynamics with PID controller (a 3-mass plant system with $k_g = k_l = 5N/m$).

Figure F.9 shows a bar chart comparison of the 3-mass plant system link's tracking errors using various controllers.

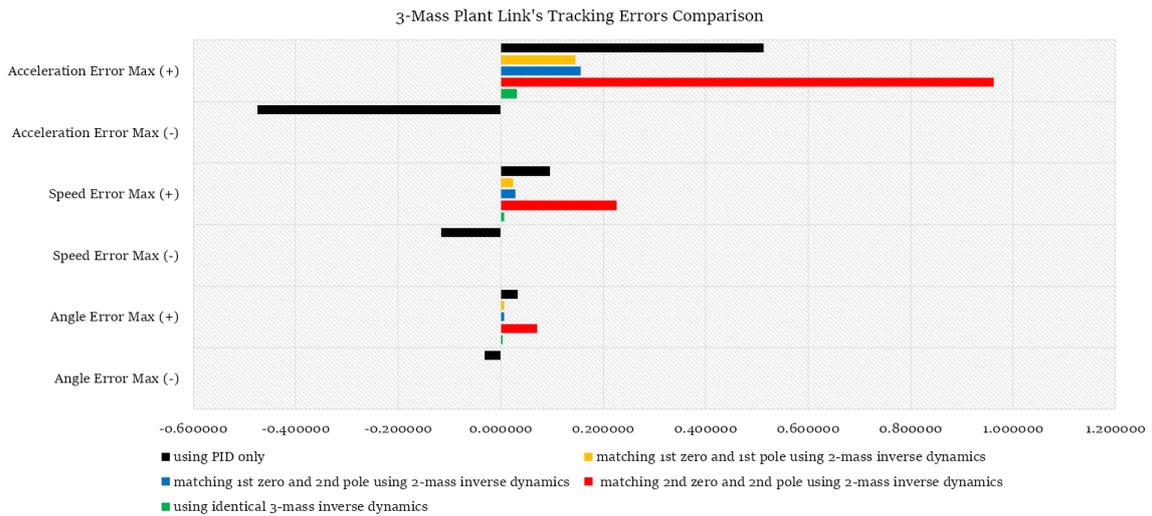


Figure F.9: Comparison of link's angle, speed and acceleration tracking results obtained using various controllers for a 3-mass plant system.

F.3 Multi-Mass Plant Systems

F.3.1 A 2-Mass Plant System Case

F.3.1.1 With A Simple Servo PID Controller Results

Figure F.10 and Table F.10 illustrate both qualitatively and quantitatively, respectively, the flexible lumped link's Tip tracking performance of the resulting 2-mass plant system whose frequency response is shown in Figure 2.14, when it is controlled by a simple servo PID controller only.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.085512	0.091353	-0.296367	0.220420	-0.957521	1.072504

Table F.8: A multi-mass system (2-mass plant case) tracking with a PID controller only results

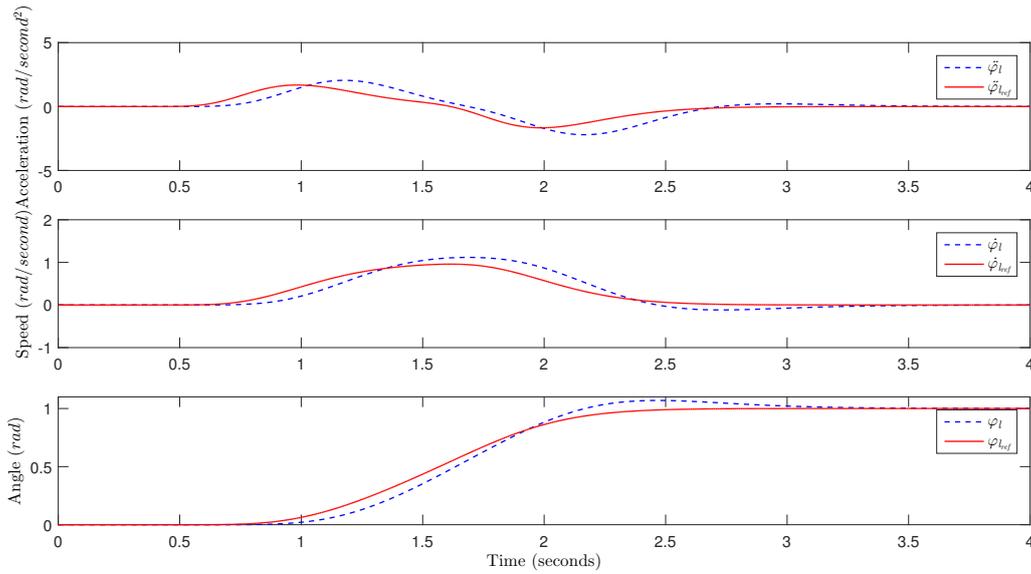


Figure F.10: Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 2-mass plant system).

F.3.1.2 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.11 and Table F.11 illustrate both qualitatively and quantitatively, respectively, the flexible lumped link's Tip tracking performance of the resulting 2-mass plant system whose frequency response is shown in Figure 2.14, when it is controlled by a 2-mass inverse dynamics with a simple servo PID controller.

F. Detailed Results and Comparisons

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.001045	0.003733	-0.007637	0.005950	-0.036361	0.026958

Table F.9: A multi-mass system (2-mass plant case) tracking with a 2-mass inverse dynamics and PID controller results

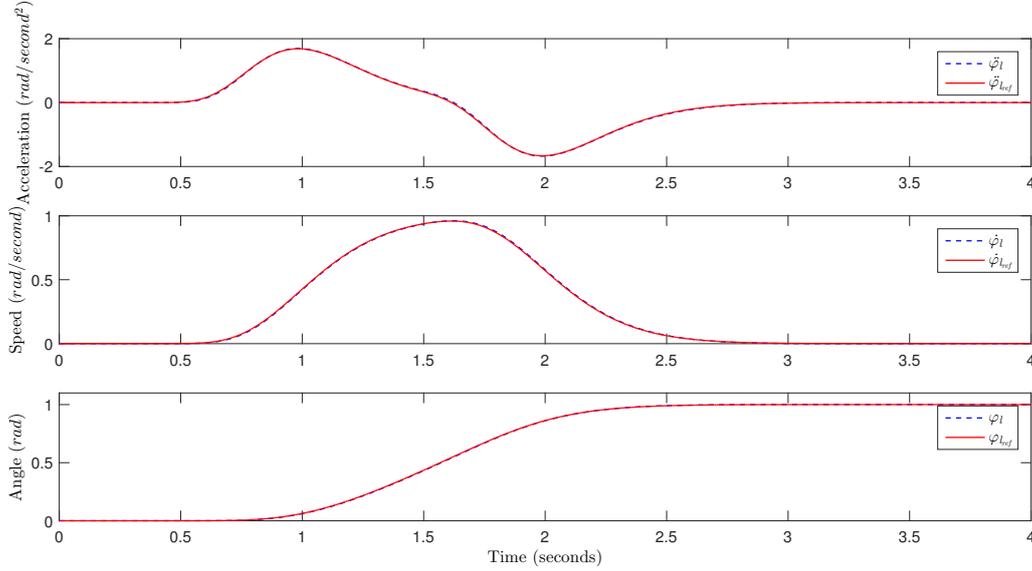


Figure F.11: Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics with PID controller (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 2-mass plant system).

Figure F.12 shows a bar chart comparison of the link's tracking errors using various controllers for a multi-mass plant with a lumped link having parameters that result in a 2-mass system.

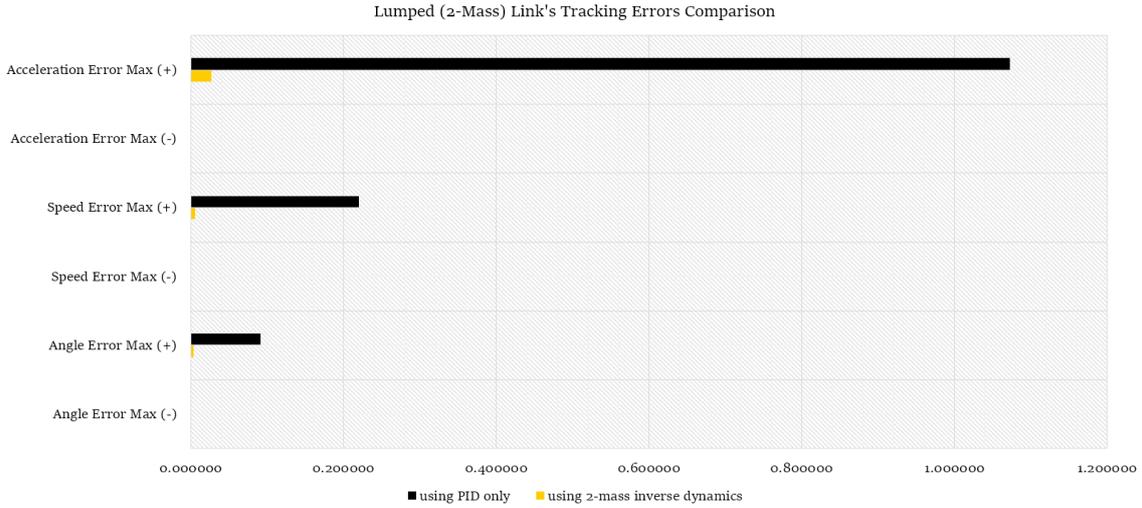


Figure F.12: Comparison of link’s angle, speed and acceleration tracking results obtained using various controllers for a multi-mass plant with a lumped link parameters that results in a 2-mass system.

F.3.2 A 3-Mass Plant System Case

F.3.2.1 With A Simple Servo PID Controller Results

Figure F.13 and Table F.13 illustrate both qualitatively and quantitatively, respectively, the flexible lumped link’s Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.15, when it is controlled by a simple servo PID controller only.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.085512	0.091353	-0.296367	0.220420	-0.957521	1.072504

Table F.10: A multi-mass system (3-mass plant case) tracking with a PID controller only results

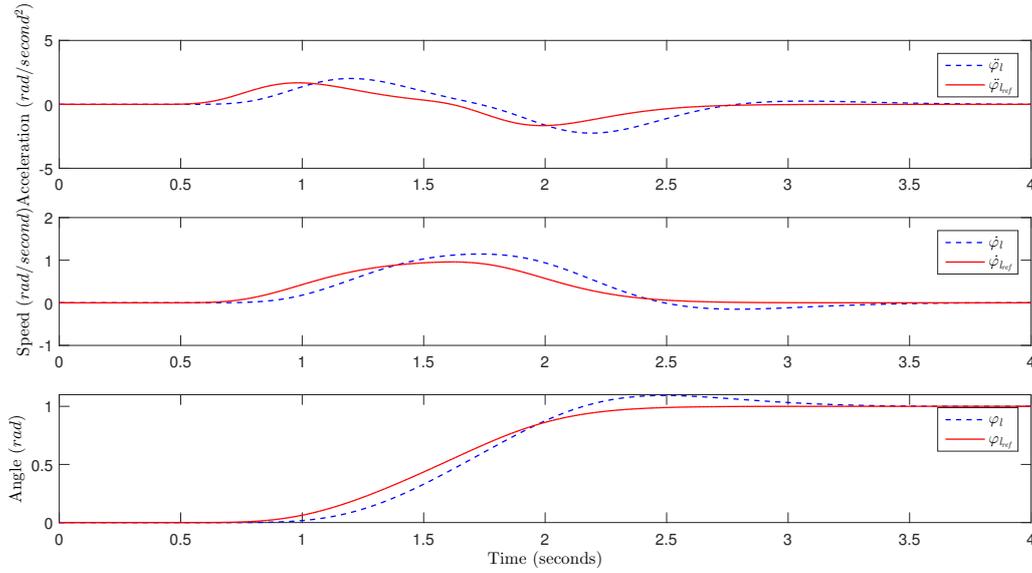


Figure F.13: Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 3-mass plant system).

F.3.2.2 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.14 and Table F.14 illustrate both qualitatively and quantitatively, respectively, the flexible lumped link's Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.14, when it is controlled by a 2-mass inverse dynamics (by frequency matching of the lower pair of mode and anti-mode) with a simple servo PID controller.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.001632	0.004376	-0.012736	0.010208	-0.059452	0.053705

Table F.11: A multi-mass system (3-mass plant case) tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller results

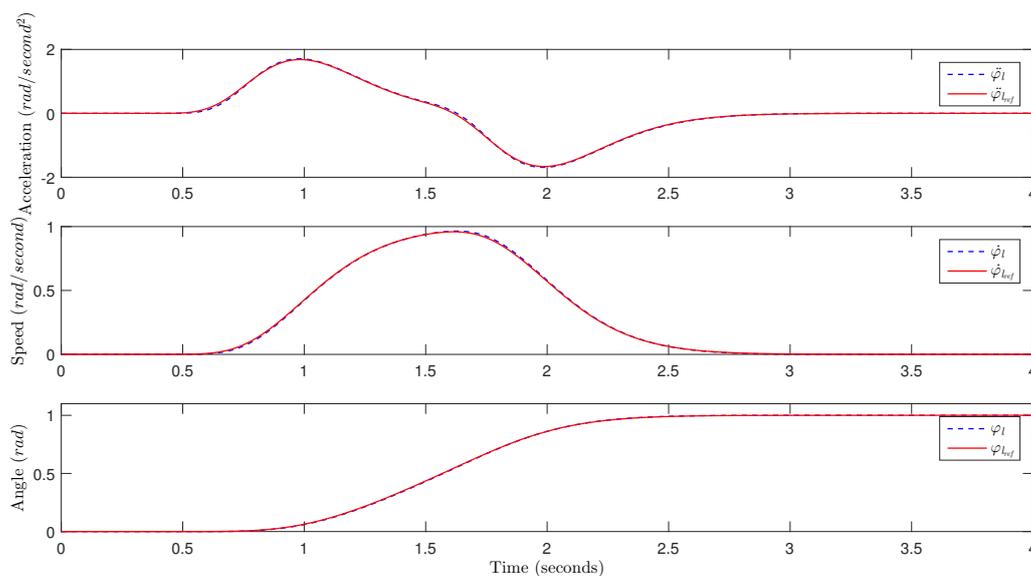


Figure F.14: Flexible lumped link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [lower pair of mode and anti-mode matching] with PID controller (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 3-mass plant system).

F.3.2.3 With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.15 and Table F.15 illustrate both qualitatively and quantitatively, respectively, the flexible lumped link's Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.15, when it is controlled by a 2-mass inverse dynamics with a simple servo PID controller.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.001639	0.004377	-0.009809	0.007408	-0.033830	0.026480

Table F.12: A multi-mass system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller results

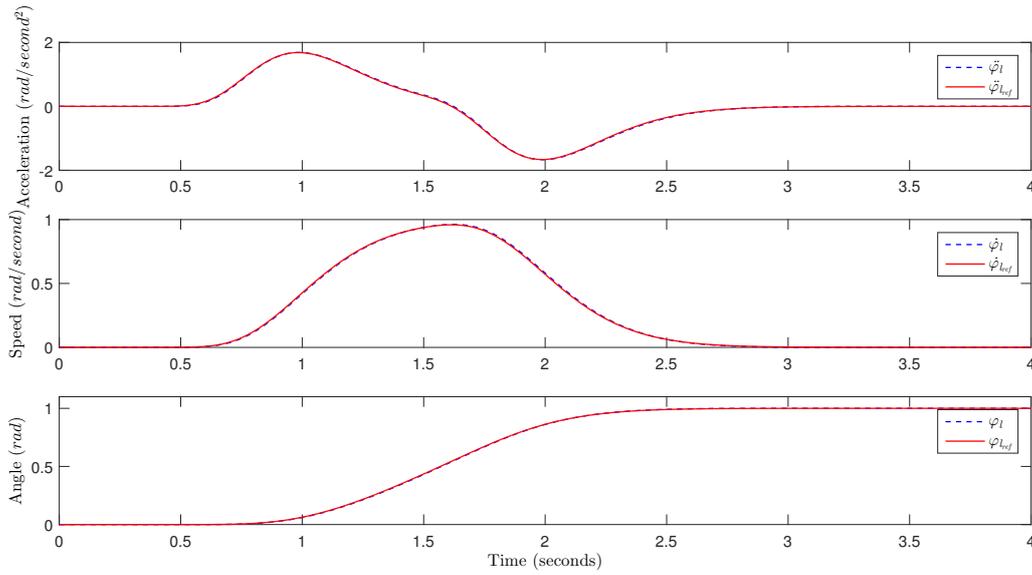


Figure F.15: Flexible lumped link’s tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller (multi-mass plant with $n = 15$ and a set of parameter values resulting in a 3-mass plant system).

Figure F.16 shows a bar chart comparison of the link’s tracking errors using various controllers for a multi-mass plant with a lumped link having parameters that result in a 3-mass system.

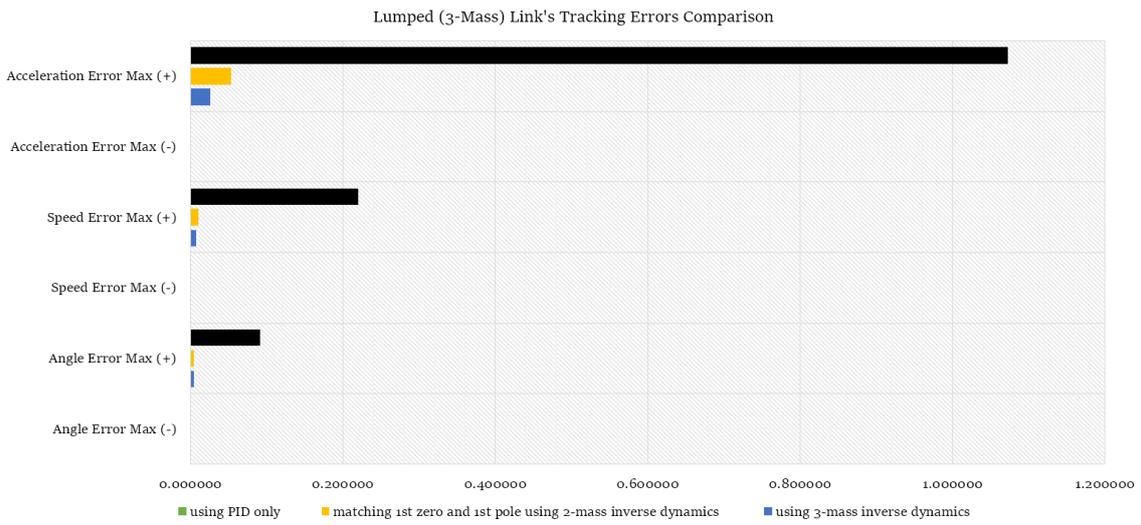


Figure F.16: Comparison of link’s angle, speed and acceleration tracking results obtained using various controllers for a multi-mass plant with a lumped link parameters that results in a 3-mass system.

F.4 ADAMS Plant Systems

F.4.1 A 2-Mass Plant System Case

F.4.1.1 With A Simple Servo PID Controller Results

Figure F.17 and Table F.17 illustrate both qualitatively and quantitatively, respectively, the flexible ADAMS link's Tip tracking performance of the resulting 2-mass plant system whose frequency response is shown in Figure 2.19, when it is controlled by a simple servo PID controller only.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.106301	0.110699	-0.366976	0.2546190	-1.024695	1.208989

Table F.13: An ADAMS system (2-mass plant case) tracking with a PID controller only results

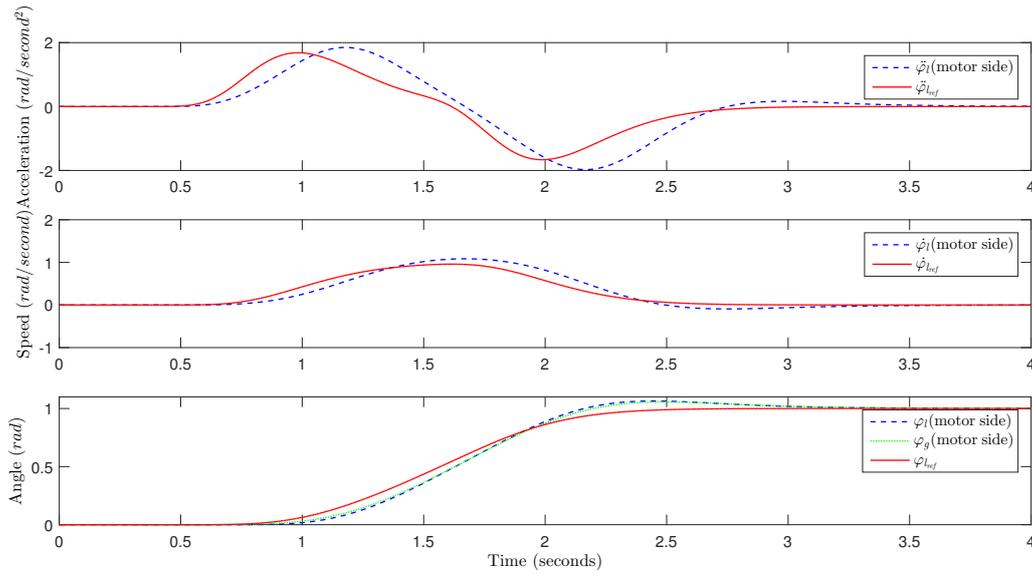


Figure F.17: Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (a 2-mass plant system).

F.4.1.2 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.18 and Table F.18 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link's Tip tracking performance of the resulting 2-mass plant system whose frequency response is shown in Figure 2.19, when it is controlled by a 2-mass inverse dynamics with a simple servo PID controller.

F. Detailed Results and Comparisons

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.002652	0.005685	-0.032380	0.034617	-0.223495	0.216481

Table F.14: An ADAMS system (2-mass plant case) tracking with a 2-mass inverse dynamics and PID controller results

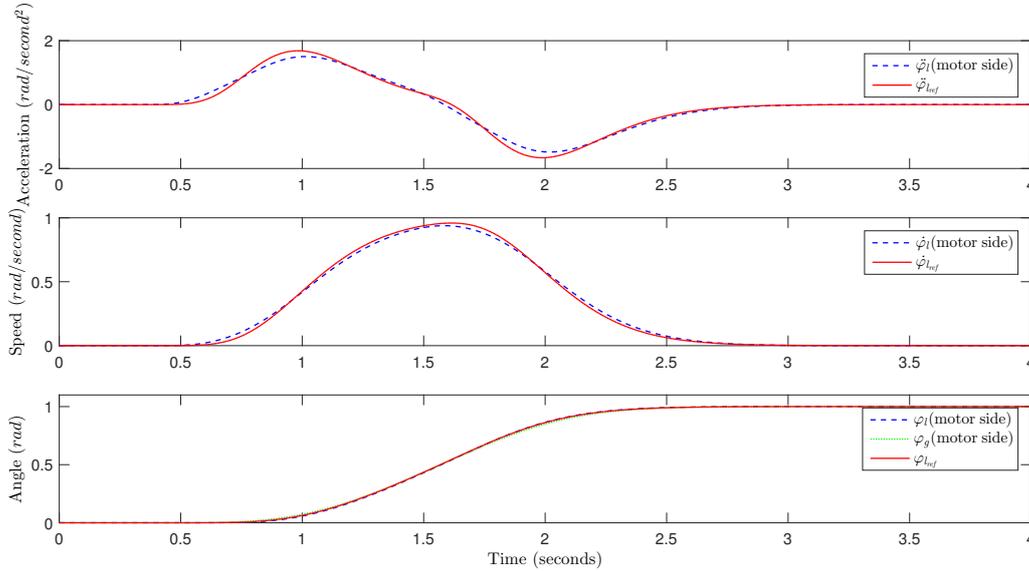


Figure F.18: Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics with PID controller (a 2-mass plant system).

F.4.1.3 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position Difference Error Compensation Results

Figure F.19 and Table F.19 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link's Tip tracking performance of the resulting 2-mass plant system whose frequency response is shown in Figure 2.19, when it is controlled by a 2-mass inverse dynamics with a simple servo PID controller, in addition to angular position difference error compensation.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.001211	0.003144	-0.040898	0.046700	-0.240849	0.224323

Table F.15: An ADAMS system (2-mass plant case) tracking with a 2-mass inverse dynamics and PID controller with angular position difference error compensation results

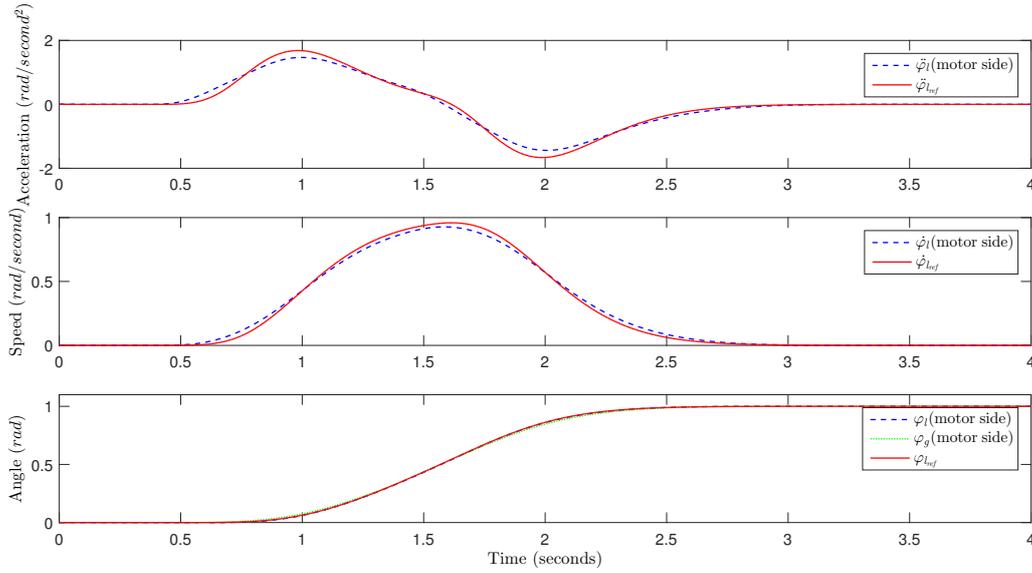


Figure F.19: Flexible FEM link’s tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics with PID controller, in addition to angular position difference error compensation (a 2-mass plant system).

F.4.1.4 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position And Angular Speed Difference Errors Compensation Results

Figure F.20 and Table F.20 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link’s Tip tracking performance of the resulting 2-mass plant system whose frequency response is shown in Figure 2.19, when it is controlled by a 2-mass inverse dynamics with a simple servo PID controller, in addition to both angular position and angular speed difference errors compensation.

$\min(\varphi_{lref} - \varphi_l)$	$\max(\varphi_{lref} - \varphi_l)$	$\min(\dot{\varphi}_{lref} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{lref} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{lref} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{lref} - \ddot{\varphi}_l)$
-0.001230	0.003210	-0.041135	0.046756	-0.242779	0.227029

Table F.16: An ADAMS system (2-mass plant case) tracking with a 2-mass inverse dynamics and PID controller with angular position and angular speed difference errors compensations results

F. Detailed Results and Comparisons

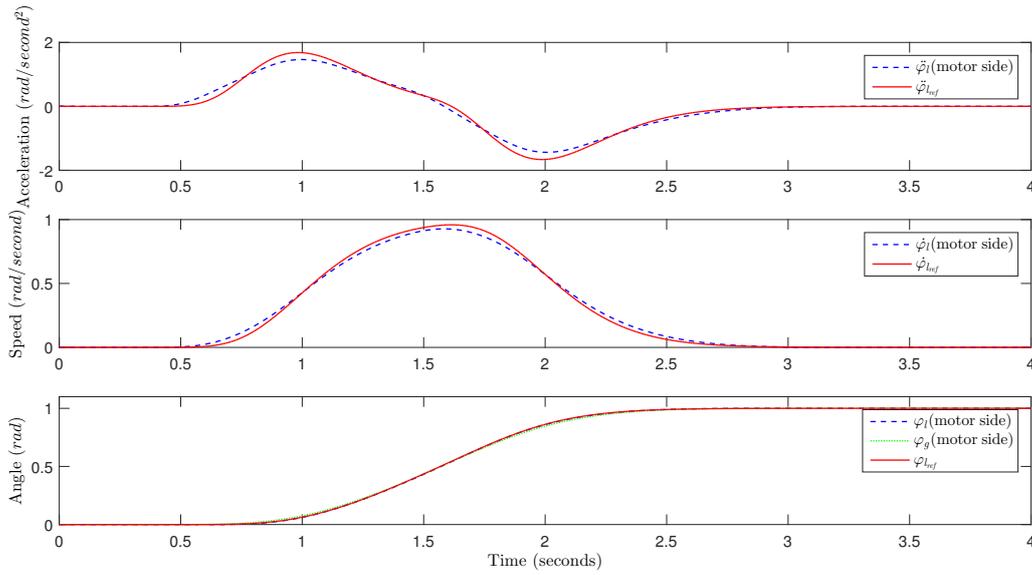


Figure F.20: Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics with PID controller, in addition to both angular position and angular speed difference errors compensation (a 2-mass plant system).

Figure F.21 shows a bar chart comparison of the link's tracking errors using various controllers for an ADAMS plant with a FEM link having parameters that result in a 2-mass system.

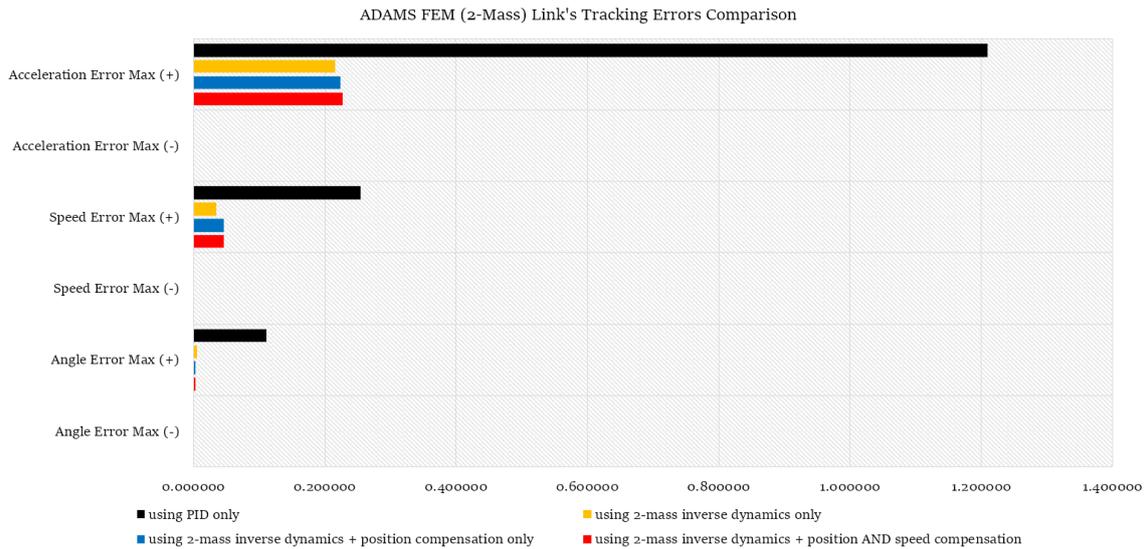


Figure F.21: Comparison of link's angle, speed and acceleration tracking results obtained using various controllers for an ADAMS plant with a FEM link parameters that results in a 2-mass system.

F.4.2 A 3-Mass Plant System Case

F.4.2.1 With A Simple Servo PID Controller Results

Figure F.22 and Table F.22 illustrate both qualitatively and quantitatively, respectively, the flexible ADAMS link's Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.20, when it is controlled by a simple servo PID controller only.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.103417	0.108585	-0.311751	0.216740	-0.881969	1.032136

Table F.17: An ADAMS system (3-mass plant case) tracking with a PID controller only results

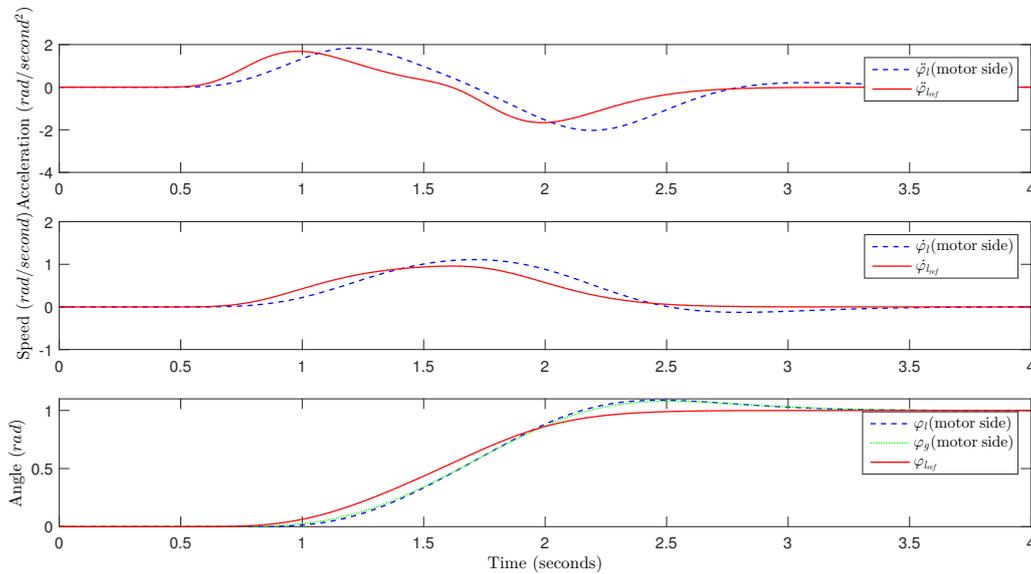


Figure F.22: Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using PID controller only (a 3-mass plant system).

F.4.2.2 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.23 and Table F.23 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link's Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.20, when it is controlled by a 2-mass inverse dynamics (by frequency matching of the lower pair of mode and anti-mode) with a simple servo PID controller.

F. Detailed Results and Comparisons

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.003980	0.007131	-0.029652	0.030330	-0.207301	0.201920

Table F.18: An ADAMS system (3-mass plant case) tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller results

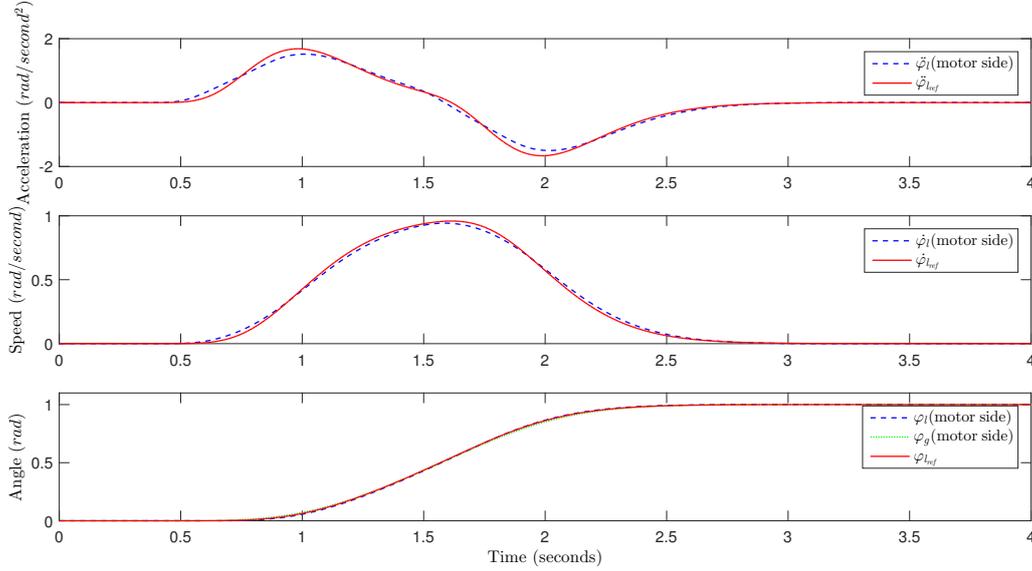


Figure F.23: Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using 2-mass inverse dynamics [lower pair of mode and anti-mode matching] with PID controller (a 3-mass plant system).

F.4.2.3 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position Difference Error Compensation Results

Figure F.24 and Table F.24 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link's Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.20, when it is controlled by a 2-mass inverse dynamics with a simple servo PID controller, in addition to angular position difference error compensation.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.002678	0.006474	-0.044735	0.054429	-0.242938	0.214651

Table F.19: An ADAMS system (3-mass plant case) tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller with angular position difference error compensation results

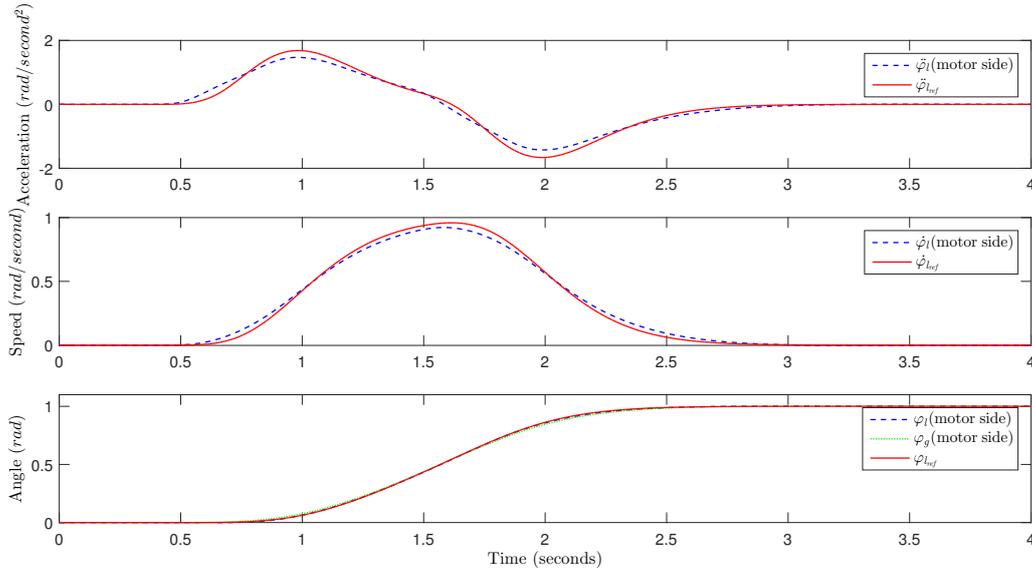


Figure F.24: Flexible FEM link’s tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics [lower pair of mode and anti-mode matching] with PID controller, in addition to angular position difference error compensation (a 3-mass plant system).

F.4.2.4 With A 2-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position And Angular Speed Difference Errors Compensation Results

Figure F.25 and Table F.25 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link’s Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.19, when it is controlled by a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) with a simple servo PID controller, in addition to both angular position and angular speed difference errors compensation.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.002667	0.006471	0.044960	0.0544955	0.243793	0.217596

Table F.20: An ADAMS system (3-mass plant case) tracking with a 2-mass inverse dynamics (lower pair of mode and anti-mode matching) and PID controller with angular position and angular speed difference errors compensations results

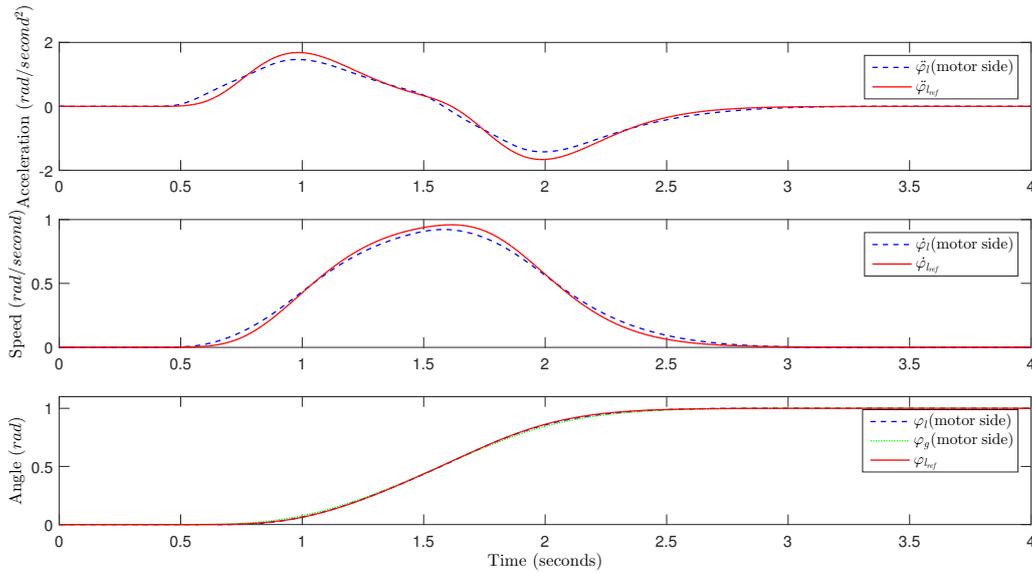


Figure F.25: Flexible FEM link’s tip angular position, angular speed and angular acceleration reference tracking using a 2-mass inverse dynamics[lower pair of mode and anti-mode matching] with PID controller, in addition to both angular position and angular speed difference errors compensation (a 3-mass plant system).

F.4.2.5 With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller Results

Figure F.26 and Table F.26 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link’s Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.20, when it is controlled by a 3-mass inverse dynamics with a simple servo PID controller.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.003073	0.006110	-0.031644	0.033624	-0.221832	0.215012

Table F.21: An ADAMS system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller results

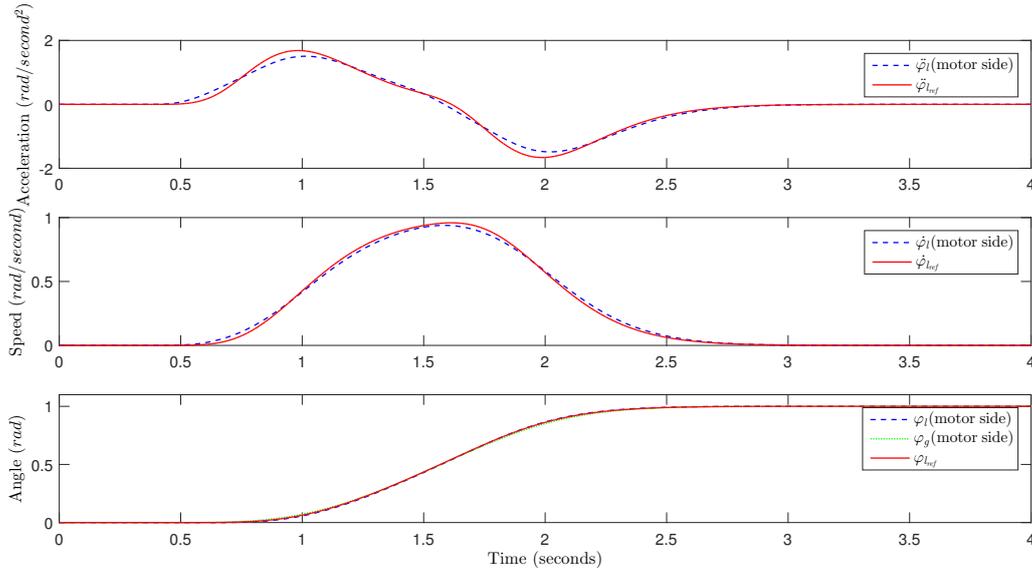


Figure F.26: Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller (a 3-mass plant system).

F.4.2.6 With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position Difference Error Compensation Results

Figure F.27 and Table F.27 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link's Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.20, when it is controlled by a 3-mass inverse dynamics with a simple servo PID controller, in addition to angular position difference error compensation.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.001066	0.002925	-0.039944	0.046284	-0.248577	0.228556

Table F.22: An ADAMS system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller with angular position difference error compensation results

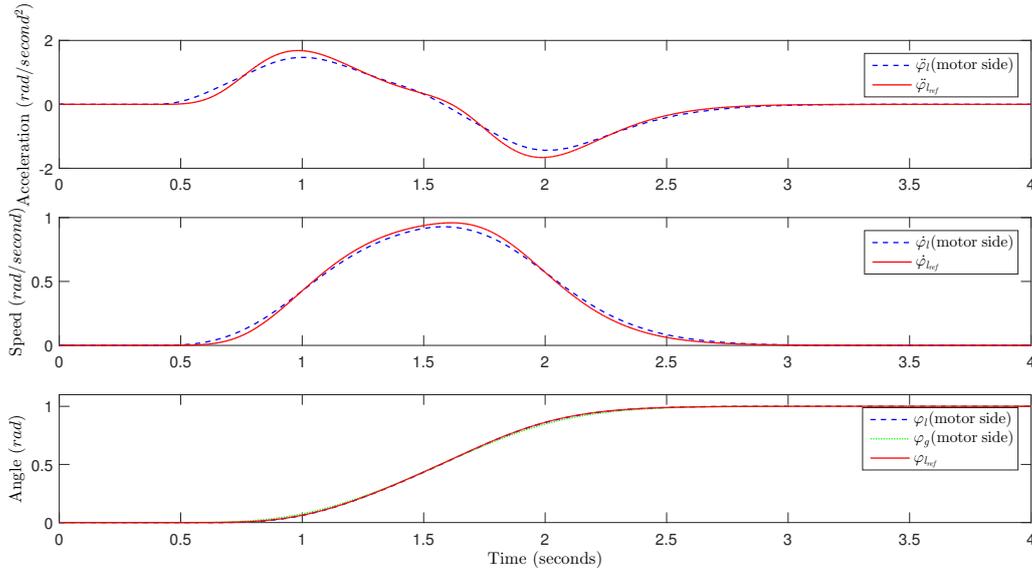


Figure F.27: Flexible FEM link’s tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller, in addition to angular position difference error compensation (a 3-mass plant system).

F.4.2.7 With A 3-Mass Inverse Dynamics Model Feed-forward Computed Torque with PID Controller and Angular Position And Angular Speed Difference Errors Compensation Results

Figure F.28 and Table F.28 illustrate both qualitatively and quantitatively, respectively, the flexible FEM link’s Tip tracking performance of the resulting 3-mass plant system whose frequency response is shown in Figure 2.20, when it is controlled by a 3-mass inverse dynamics with a simple servo PID controller, in addition to both angular position and angular speed difference errors compensation.

$\min(\varphi_{l_{ref}} - \varphi_l)$	$\max(\varphi_{l_{ref}} - \varphi_l)$	$\min(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\max(\dot{\varphi}_{l_{ref}} - \dot{\varphi}_l)$	$\min(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$	$\max(\ddot{\varphi}_{l_{ref}} - \ddot{\varphi}_l)$
-0.001066	0.002929	-0.039955	0.046281	-0.249029	0.228360

Table F.23: An ADAMS system (3-mass plant case) tracking with a 3-mass inverse dynamics and PID controller with angular position and angular speed difference errors compensations results

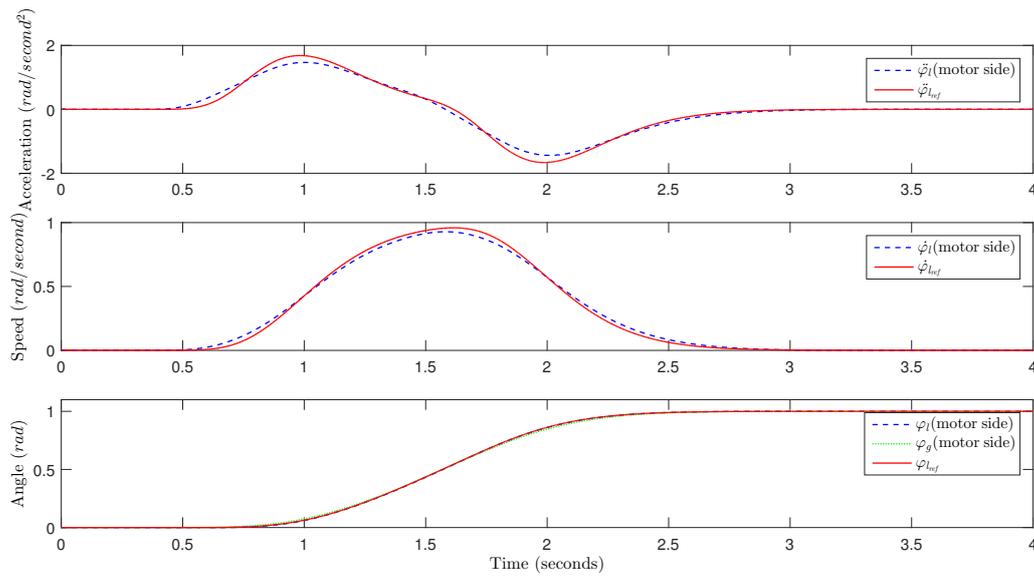


Figure F.28: Flexible FEM link's tip angular position, angular speed and angular acceleration reference tracking using a 3-mass inverse dynamics with PID controller, in addition to both angular position and angular speed difference errors compensation (a 3-mass plant system).

Figure F.29 shows a bar chart comparison of the link's tracking errors using various controllers for an ADAMS plant with a FEM link having parameters that result in a 3-mass system.

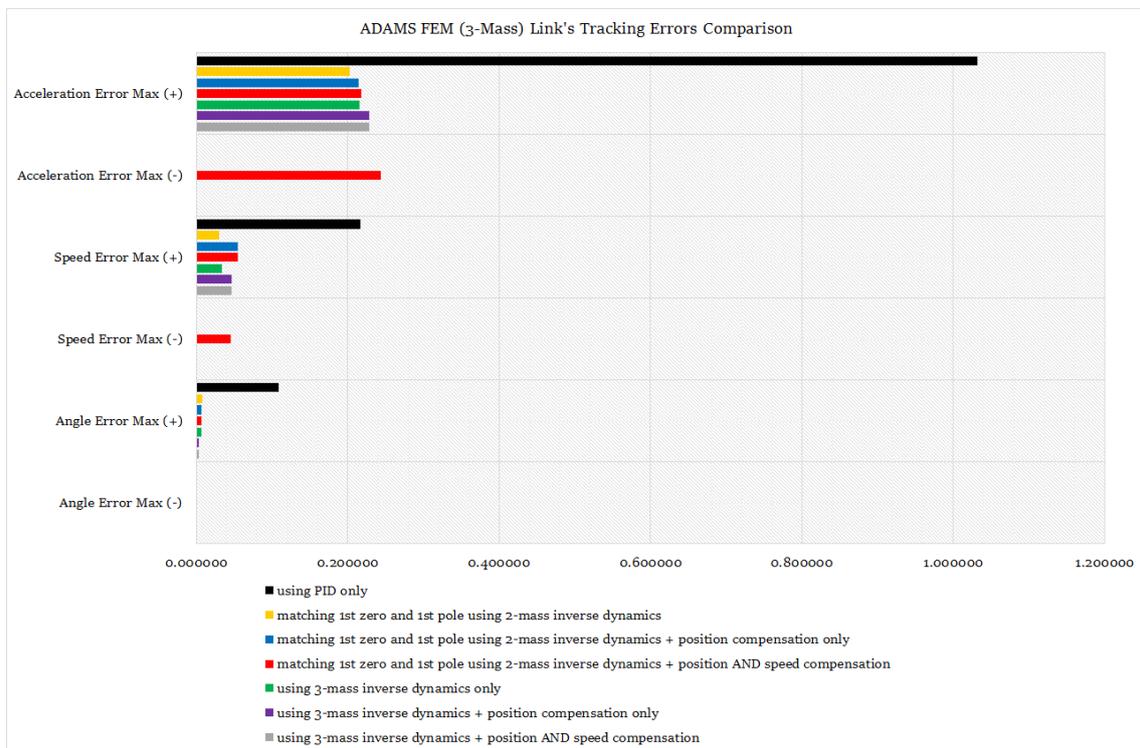


Figure F.29: Comparison of link's angle, speed and acceleration tracking results obtained using various controllers for an ADAMS plant with a FEM link parameters that results in a 3-mass system.