MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395 rue Wellington
Ottawa (Ontario)
K1A 0N4

For more information

consult the microfiche

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

FORCE CONTROL OF A FLEXIBLE JOINT ROBOT

by

Zbigniew Jan Borowski M. Eng.

**A thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of
Master of Engineering**

**OTTAWA-CARLETON INSTITUTE FOR MECHANICAL
AND AEROSPACE ENGINEERING
Department of Mechanical and Aerospace Engineering**

**Carleton University
Ottawa, Ontario**

August, 1993

**copyright©
1993, Zbigniew Jan Borowski**

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file - Votre référence

Our file - Notre référence

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-93008-X

Canada

Name _____

Dissertation Abstracts International is arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation. Enter the corresponding four digit code in the spaces provided.

SUBJECT TERM

SUBJECT CODE

U·M·I

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS

Architecture 0529
Art History 0527
Cinema 0508
Dance 0538
Fine Arts 0537
Information Science 0525
Journalism 0523
Library Science 0529
Mass Communications 0508
Music 0519
Speech Communication 0559
Theater 0565

EDUCATION

General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0523
Bilingual and Multicultural 0528
Business 0588
Community College 0525
Curriculum and Instruction 0527
Early Childhood 0518
Elementary 0524
Finance 0527
Guidance and Counseling 0519
Health 0580
Higher 0545
History of 0520
Home Economic 0528
Industrial 0521
Language and Literature 0529
Mathematics 0580
Music 0522
Philosophy of 0598
Physical 0523

Psychology 0525
Reading 0535
Religious 0527
Sciences 0514
Secondary 0532
Social Sciences 0534
Sociology of 0540
Special 0529
Teacher Training 0536
Technology 0510
Tests and Measurements 0588
Vocational 0547

LANGUAGE, LITERATURE AND LINGUISTICS

Language 0679
General 0289
Ancient 0290
Linguistics 0291
Modern
Literature
General 0401
Classical 0294
Comparative 0295
Medieval 0297
Modern 0298
African 0316
American 0591
Asian 0305
Canadian (English) 0352
Canadian (French) 0355
English 0593
Germanic 0311
Latin American 0312
Middle Eastern 0315
Romance 0313
Slavic and East European 0314

PHILOSOPHY, RELIGION AND THEOLOGY

Philosophy 0422
Religion
General 0318
Biblical Studies 0321
Clergy 0319
History of 0320
Philosophy of 0322
Theology 0469

SOCIAL SCIENCES

American Studies 0323
Anthropology
Archaeology 0324
Cultural 0326
Physical 0327
Business Administration
General 0310
Accounting 0272
Banking 0770
Management 0454
Marketing 0338
Canadian Studies 0385
Economics
General 0501
Agricultural 0503
Commerce Business 0505
Finance 0508
History 0509
Labor 0510
Theory 0511
Folklore 0358
Geography 0366
Gerontology 0357
History
General 0578

Ancient 0579
Medieval 0581
Modern 0582
Black 0328
African 0331
Asia, Australia and Oceania 0332
Canadian 0334
European 0335
Latin American 0336
Middle Eastern 0333
United States 0337
History of Science 0585
Law 0398
Political Science
General 0615
International Law and Relations 0616
Public Administration 0617
Recreation 0814
Social Work 0452
Sociology
General 0626
Criminology and Penology 0627
Demography 0938
Ethnic and Racial Studies 0631
Individual and Family Studies 0628
Industrial and Labor Relations 0629
Public and Social Welfare 0630
Social Structure and Development 0700
Theory and Methods 0344
Transportation 0709
Urban and Regional Planning 0999
Women's Studies 0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES

Agriculture 0473
General 0485
Agronomy
Animal Culture and Nutrition 0475
Animal Pathology 0476
Food Science and Technology 0359
Forestry and Wildlife 0478
Plant Culture 0479
Plant Pathology 0480
Plant Physiology 0481
Range Management 0777
Wood Technology 0486
Biology
General 0306
Anatomy 0287
Biostatistics 0308
Botany 0309
Cell 0370
Ecology 0329
Entomology 0353
Genetics 0368
Immunology 0393
Microbiology 0410
Molecular 0371
Neuroscience 0416
Neurophysiology 0416
Physiology 0433
Radiation 0821
Veterinary Science 0778
Zoology 0472
Biophysics
General 0786
Medical 0780

EARTH SCIENCES

Biogeochemistry 0425
Geochemistry 0426

Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoclimatology 0426
Paleontology 0418
Paleozoology 0985
Palynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES

Environmental Sciences 0768
Health Sciences
General 0566
Audiology 0300
Chemotherapy 0992
Dentistry 0567
Education 0350
Hospital Management 0769
Human Development 0758
Immunology 0982
Medicine and Surgery 0564
Mental Health 0427
Nursing 0569
Nutrition 0570
Obstetrics and Gynecology 0380
Occupational Health and Safety 0354
Therapy 0381
Ophthalmology 0571
Pathology 0419
Pharmacology 0572
Pharmacy 0382
Physical Therapy 0573
Public Health 0574
Radiology 0575
Recreation 0575

Speech Pathology 0460
Toxicology 0383
Home Economics 0386

PHYSICAL SCIENCES

Pure Sciences
Chemistry
General 0485
Agricultural 0749
Analytical 0486
Biochemistry 0487
Inorganic 0488
Nuclear 0738
Organic 0490
Pharmaceutical 0491
Physical 0494
Polymer 0495
Radiation 0754
Mathematics 0405
Physics
General 0605
Acoustics 0986
Astronomy and Astrophysics 0606
Atmospheric Science 0608
Atomic 0748
Electronics and Electricity 0607
Elementary Particles and High Energy 0798
Fluid and Plasma 0759
Molecular 0609
Nuclear 0610
Optics 0752
Radiation 0756
Solid State 0611
Statistics 0463
Applied Sciences
Applied Mechanics 0346
Computer Science 0984

Engineering
General 0537
Aerospace 0538
Agricultural 0539
Automotive 0540
Biomedical 0541
Chemical 0542
Civil 0543
Electronics and Electrical 0544
Heat and Thermodynamics 0348
Hydraulic 0545
Industrial 0546
Marine 0547
Materials Science 0794
Mechanical 0548
Metallurgy 0743
Mining 0551
Nuclear 0552
Packaging 0549
Petroleum 0765
Sanitary and Municipal 0554
System Science 0790
Geotechnology 0428
Operations Research 0796
Plastics Technology 0795
Textile Technology 0994

PSYCHOLOGY

General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

THESIS ACCEPTANCE FORM

The undersigned recommended to the Faculty of Graduate Studies
and Research acceptance of the thesis

"FORCE CONTROL OF A FLEXIBLE JOINT ROBOT"

submitted by Zbigniew Jan Borowski, M. Eng.
in partial fulfilment of the requirements for the
degree of Master of Engineering

Thesis Supervisor

Thesis Co-Supervisor

Chair, Department of Mechanical and Aerospace Engineering

Carleton University

August, 1993

ABSTRACT

The fast development of technology in robotics requires the designing of new and advanced control systems. These systems are increasingly complex, their goal providing manipulators with required performance, depending on the task to be performed.

The modelling of the flexible joint manipulator dynamics in constrained environment with the focus on the exerted contact force is an important aspect in Automated Control.

This work presents theoretical and practical issues related to robot force control in constrained environment. Classical and modern control laws are reviewed and implemented. Advanced control schemes are proposed for the given system. The main field of interest is the manipulator joint behaviour in the presence of nonlinear joint parameters such as Coulomb friction and joint flexibility. The sources of contact instability of the manipulator link end are investigated. Experimental results are presented with the single link manipulator SLIM-I. The testing facility is equipped with the modern, high technology hardware.

To my parents.....

Therefore, to explore the above mentioned problems related to the flexible robot joint force control, the major goals of this work are:

- to investigate sources of contact instability of a flexible joint manipulator
- to design an efficient force control scheme for a flexible robot joint

The design process and development of the concepts and guide lines for this work on the flexible robot joint are presented, and steps performed toward the following goals:

- improvement in manipulator mechanical modelling, adequate to the given task
- appropriate choosing of a control algorithm, providing satisfactory system performance.

Other significant aspects of robot force/torque control discussed in this work are related to the following:

- feedback data acquisition
- data filtering and estimation
- signal/ noise interpretation to obtain feedback information
- control system component calibration

Finally, results of various control tests performed on the existing SIM-1 manipulator are presented.

TABLE OF CONTENTS

	Page
THESIS ACCEPTANCE FORM	ii
ABSTRACT	iii
ACKNOWLEDGEMENTS	v
TABLE OF CONTENTS	vi
LIST OF TABLES	ix
LIST OF FIGURES	x
NOMENCLATURE	xiii
1. INTRODUCTION	1
2. FORCE/TORQUE CONTROL THEORY	4
2.1 Force Control Method Classification	4
2.2 Robot Force Control Strategy	5
2.3 Sensory Feedback in Force Control	7
2.4 Force/Torque Control Methods	11
2.5 System Stability	13
2.6 Robot Joint Flexibility	16
2.7 Friction in Force Control	19
2.8 Force on the Manipulator Tip	21
2.8.1 Desired Contact Force	23
2.8.2 Actual Force on the Manipulator Tip	24
3. EXPERIMENTAL SETUP DESCRIPTION	26
3.1 Joint Actuator	31
3.2 Manipulator Link Modelling	33
3.3 Joint Speed and Position Sensing Devices	34
3.4 Servo Motor Amplifier	36
3.5 Force/Torque Sensor	37

3.6 Control System Component Description	39
3.6.1 IBM PC Computer	40
3.6.2 RTI Multifunction Input/Output Board	40
3.6.3 Encoder Interfacing Card	41
4. FLEXIBLE JOINT DYNAMICS MODELLING	41
4.1 Defining the System Dynamics	42
4.1.1 Classical Representation of a Flexible Joint System	43
4.1.2 State-Space System Representation	45
4.1.3 Link and Force Sensor Flexibility	46
4.2 System Response Simulation	47
4.2.1 Time Invariant Simplified Model	49
4.2.2 Discrete Time System with Two Values of Joint Stiffness	55
4.2.3 System Including Joint Friction	62
5. FORCE CONTROL LAW DESIGN	66
5.1 Open Loop Force Control	66
5.1.1 Simplified Model	67
5.1.2 Control Scheme with Motor Friction	74
5.2 Closed loop control	76
5.2.1 Classical PD Compensator, Simplified Model	79
5.2.2 PD Control Scheme with Motor Friction Compensation	88
5.2.3 PID Controller Performance	92
6. ADVANCED FORCE CONTROL LAWS	95
6.1 Optimal Force Control of a Flexible Joint	96
6.1.1 Performance Index Optimization	96
6.1.2 Design Procedure	98
6.1.3 Optimal Control Simulation	99
6.2 Model Reference Adaptive Control	103
6.2.1 MRAC Controller Design	105
6.2.2 MRAC Control Simulation	106
6.2.3 MRAC - Plant without Controller	111

6.3 Force Control Scheme with Friction Estimator	115
6.3.1 State Space Feedback Design	116
6.3.2 Simulation Examples	117
7. EXPERIMENTAL RESULTS	119
8. SUMMARY AND CONCLUSION	136
9. REFERENCES	139
APPENDIX I : SYSTEM COMPONENT SPECIFICATION	143
APPENDIX II : SIMULATION SOFTWARE	147
APPENDIX III : REAL TIME CONTROL SOFTWARE	164
APPENDIX IV : SYSTEM PARAMETER MODELLING	177

LIST OF TABLES

Table 2.6.1	Stiffness of typical robot joint components	16
Table 3.1	Joint actuator parameters - DC harmonic drive motor S6M4H	31
Table 3.3.1	Tachometer parameters	34
Table 3.5.1	JR3 force sensor loads	38
Table 4.2.1	Flexible joint dynamics simulation subroutines	47
Table 4.2.3.1	Simulation result. System with motor friction	62
Table 5.1.1.1	Simulation result. Open loop force control	69
Table 5.2.1.1	PD control - open loop system parameters	80
Table 5.2.1.2	PD controller design parameters	81
Table 5.2.3.1	PID controller parameters	93
Table 6.1.1	Optimal control parameters	99
Table 6.2.3.1	MRAC control simulation parameters	111
Table 7.1	Joint flexibility test results	120

LIST OF FIGURES

Figure 2.3.1	Control system sensitivity	9
Figure 2.6.1	H6D160 harmonic drive flexibility and a standard representation of the joint stiffness	17
Figure 2.7.1	Friction in contact of two rigid bodies	19
Figure 2.3.1	Graphical representation of slip-stick friction	20
Figure 2.8.1	The desired contact force on the manipulator tip	24
Figure 2.8.2	The actual measured contact force on the manipulator tip	25
Figure 3.1	Experimental set up - SLIM-1 manipulator	27
Figure 3.2	Experimental set up - components connection	28
Figure 3.3	Force/torque control test diagram	29
Figure 3.4	Force control test components	30
Figure 3.1.1	DC motor torque characteristic	32
Figure 3.2.1	Manipulator link profiles	33
Figure 3.3.1	Tachometer signal - Low Pass Filter	34
Figure 3.3.2.1	M23 Incremental Encoder Signal	35
Figure 3.3.2.2	Flexible joint compliance measurement	36
Figure 3.5.1	JR3 Force/torque measurement system	37
Figure 3.5.2	JR3 force sensor coordinate system	38
Figure 4.1.1	Flexible joint dynamics representation	42
Figure 4.1.2	Flexible joint representation	43
Figure 4.1.1.1	The generalized flexible joint dynamics	45
Figure 4.2.1.1	Flexible joint step input response-simulation (1)	49
Figure 4.2.1.2	Flexible joint step input response-simulation (2)	50
Figure 4.2.1.3	Flexible joint step input response-simulation (3)	51
Figure 4.2.1.4	Flexible joint variant input response-simulation (4)	52
Figure 4.2.1.5	Flexible joint varying input response-simulation (5)	53
Figure 4.2.1.6	Flexible joint step input response-simulation (6)	54

Figure 4.2.2.1	System with two values of joint stiffness	56
Figure 4.2.2.2	Algorithm to simulate the flexible joint dynamics	56
Figure 4.2.2.3	Flexible joint step input response (1).	57
Figure 4.2.2.4	Flexible joint step input response (2).	58
Figure 4.2.2.5	Flexible joint step input response (3).	59
Figure 4.2.2.6	Flexible joint step input response (4).	60
Figure 4.2.2.7	Flexible joint step input response (5).	61
Figure 4.2.3.1	Flexible joint step input response (1). Motor friction=.0064 [Nm]	63
Figure 4.2.3.2	Flexible joint step input response (2). Motor friction=.0064 [Nm]	64
Figure 4.2.3.1	Flexible joint varying input response (1). Motor friction=.0064 [Nm]	65
Figure 5.1.1	Block diagram of the open loop force control concept	67
Figure 5.1.1.2	Open loop force control of the flexible robot joint (1)	70
Figure 5.1.1.3	Open loop force control of the flexible robot joint (2)	71
Figure 5.1.1.4	Open loop force control of the flexible robot joint (3)	72
Figure 5.1.1.4	Open loop force control of the flexible robot joint (3)	73
Figure 5.2.1	Closed loop force control diagram.	76
Figure 5.2.2	Diagram of the PD force control of a flexible joint	77
Figure 5.2.1.1	Root locus - PD control of the flexible joint	82
Figure 5.2.1.2	PD Force Control, Step response for various gains (1)	83
Figure 5.2.1.3	PD Force Control, Step response for various gains (2)	84
Figure 5.2.1.4	PD Force Control, Step response for various gains (3)	85
Figure 5.2.1.5	PD Force Control, Step response for various gains (4)	86
Figure 5.2.1.6	PD Force Control, Step response for various gains (5)	87
Figure 5.2.2.1	PD control of a flexible joint. Motor friction compensating	91
Figure 5.2.3.1	PID control of a flexible joint. Step input response	94
Figure 6.1.1	Optimal control scheme. State feedback approach.	96

Figure 6.1.2	Optimal control of a flexible robot joint (1)	100
Figure 6.1.3	Optimal control of a flexible robot joint (2)	101
Figure 6.1.4	Optimal control of a flexible robot joint (3)	102
Figure 6.2.1	Diagram of the MRAC control of a flexible joint	103
Figure 6.2.2.1	MRAC adaptive control - Simulation (1,2)	107
Figure 6.2.2.2	MRAC adaptive control - Simulation (3,4)	108
Figure 6.2.2.3	MRAC adaptive control - Simulation (5,6)	109
Figure 6.2.2.4	MRAC adaptive control - Simulation (7)	110
Figure 6.2.3.1	MRAC control simulation (1). Model-first order filter	112
Figure 6.2.3.2	MRAC control simulation (2). Model-first order filter	113
Figure 6.2.3.3	MRAC control simulation (3). Model-first order filter	114
Figure 6.3.2.1	Modern Control Scheme with friction estimator	118
Figure 7.1	Joint flexibility evaluation test	123
Figure 7.2	Flexible joint characteristic. Joint torque versus flexibility	124
Figure 7.3	Open loop force control of a flexible robot joint	125
Figure 7.4	Flexible joint force control. Open loop scheme	126
Figure 7.5	Flexible joint force control (1). PD control scheme	127
Figure 7.6	Flexible joint force control (2). PD control scheme	128
Figure 7.7	Flexible joint force control (3). PD control scheme	129
Figure 7.8	Flexible joint force control (1). PID control scheme	130
Figure 7.9	Flexible joint force control (2). PID control scheme	131
Figure 7.10	Flexible joint force control. MRAC control scheme (1)	132
Figure 7.11	Flexible joint force control. MRAC control scheme (1)	133
Figure 7.12	MRAC control of a flexible joint (2). Experimental results	134
Figure 7.13	MRAC control of a flexible joint (2). Experimental results	135

NOMENCLATURE

A, B, C, D, E	State space design matrices
A_d, B_d, C_d	Discrete time domain system
$A(s), B(s)$	Characteristic polynomials, continuous time domain
α, β, γ	Controller design parameters - P, PD, PID control
F, F_x, F_y, F_z	Force on the manipulator tip and its components
F_d, F_a, F_e	Force: desired, actual, error
F_s, F_k	Friction force: static, kinetic
F_{lm}, F_{lj}, F_{ld}	Coulomb friction forces: motor, load, joint
$\tilde{F}_{lj}, F_{lj}^0$	Observer and initial value value of joint friction
G, G_1	System proportional gain
g_x, g_y, g_z	JR3 force sensor torsional stiffness
$H(s)$	Plant transfer function, general
$H_c(s), H_j(s)$	Transfer function: controller, joint
$H_o(s), H_{cl}(s)$	Open and closed loop system transfer function
J_m, B_m, b_m	Motor inertia, damping
J_l, B_l, b_l	Load inertia, damping
J	Performance index - optimal control
K, K	System gain, vector of optimal gains
K_g, K_{g1}, K_{g2}	Joint stiffness
K_p, K_d, K_i	System gains: proportional, derivative and integral
K_m, K_a	Constants: motor torque and amplifier
k_x, k_y, k_z	JR3 force sensor linear stiffness
$J(q), M(q), C(q)$	Manipulator Jacobian, inertia and damping matrix
λ	Vector of Lagrange multipliers (Page 21)
$\Phi(x)$	Hypersurface equation
M_a, M_d	Joint torque: actual, desired

μ_s, μ_k	Friction coefficient: static, kinetic
N	Harmonic drive gear ratio
η_g, J_g	Harmonic drive gear efficiency and inertia
θ_m, θ_L	Rotational displacement: motor, joint
Q, R	Weighting and penalty matrices
P	Solution of Riccati equation
P, P_x, P_y	Load force
T_L	Load torque
τ, τ	Joint driving torque
$U(s), C(s)$	System input and output - Laplace domain
q, x	Displacement vectors: joint, Cartesian
X_0, X_{INI}	Vector of initial system states
$E(z), \Gamma(z)$	Polynomials in discrete time domain (Page 14)
$W(t)$	External disturbance, Coulomb friction matrix
ξ, ω, ω_n	System damping, frequency and natural frequency
χ, Ψ	Coefficients of characteristic equation
ζ, ψ	Optimal control design parameters
ε, λ	Adaptation gain and control design parameter- MIT Rule
$\hat{u}$	Modern control scheme design parameter
C	Controllability matrix

Chapter 1

INTRODUCTION

The force and torque control of robot manipulators in various operational conditions is a field of increasing interest. This is partially due to manipulator application in tasks requiring higher precision. Attempts to use robot manipulators in all aspects of our life sets new standards in many areas related to the automated industry. Typical tasks, such as *assembly operations, welding, spray painting, picking and placing*, often do not require advanced control concepts. Solutions for some of these problems are relatively well known. Position control devices are common in industrial robots, but often do not provide the required performance and cannot prevent generation of large forces and consequent jamming of assembled parts. However, there is a significant number of operations, in which force/torque control is important. These operations are: *grinding, cutting, deburring, polishing, precision assembling*. The following classification of all issues related to the manipulator force/torque control is proposed:

- rigid model based control
- flexible model based control

The above general classification is based on the dynamical model of the robot joint considered for a particular solution, the type of actuator chosen to drive the joint and the manipulator link structure. Thus the previous classification can be expanded into four more specific subgroups, depending on manipulator link and joint structure: 1) rigid link, rigid joint, 2) rigid link, flexible joint, 3) flexible link, rigid joint, 4) flexible link, flexible joint.

Each of these requires a separate conceptual and design approach and should be provided

with an appropriate dynamical model, which usually involves different phenomena and effects. The first of the given subgroups, manipulator with both rigid link and joint, is relatively simple compared with the other three models. Its system dynamic order is usually smaller than that of a manipulator with flexible structures. The lack of joint and link flexibility simplifies efforts related to the system dynamics modelling. An example of a rigid joint structure is a manipulator equipped with a direct drive motor. Some significant drawbacks in the application of the direct drive motor are increased mass and inertia of the joint, so that a great part of driving torques is consumed to overcome joint and link inertias.

Application of a harmonic drive in the robotic industry reduces robot total mass compared to the weight of a manipulator with direct drive motors.

The overall manipulator efficiency is increased, but simultaneously the *joint flexibility* and often *backlash* are introduced. These two important parameters cause potential problems in the manipulator control analysis and its practical implementation.

This work focuses on a rigid link manipulator equipped with a harmonic drive motor. Many important theoretical and practical aspects of a force/torque control of a manipulator with a rigid link and a flexible joint are explored and examined. Classical control methods are implemented and tested on a one-degree-of freedom manipulator and compared with more advanced controller schemes.

Application of force feedback (explicit force control) can cause contact instability and link "chattering" under constrained conditions. Currently there is little information and practical results on the subject of contact instability or "chattering". The manipulator link "chattering" occurs, when the manipulator end effector breaks the contact with a constrained surface. Such behaviour may cause generating of large forces.

Therefore, to explore the above mentioned problems related to the flexible robot joint force control, the major goals of this work are:

- to investigate sources of contact instability of a flexible joint manipulator
- to design an efficient force control scheme for a flexible robot joint

The design process and development of the concepts and guide lines for this work on the flexible robot joint are presented, and steps performed toward the following goals:

- improvement in manipulator mechanical modelling, adequate to the given task
- appropriate choosing of a control algorithm, providing satisfactory system performance.

Other significant aspects of robot force/torque control discussed in this work are related to the following:

- feedback data acquisition
- data filtering and estimation
- signal/ noise interpretation to obtain feedback information
- control system component calibration

Finally, results of various control tests performed on the existing SIM-1 manipulator are presented.

Chapter 2

FORCE/TORQUE CONTROL THEORY

This chapter is an overview of concepts, terminology and principle problems relevant to the force/torque control topic. It also presents the current state of the art of force/torque control.

2.1 Force Control Method Classification

Two prevalent approaches to force control are ([1]):

- a) *impedance control*
 - active impedance*
 - passive impedance*
- b) *hybrid control*

Impedance control uses the concept of varying the stiffness of the manipulator depending on the external load. Adjusting of the stiffness allows the maintenance of the required value of the contact force.

Active impedance involves feedback control with position, velocity and force measurement. The necessary mechanical impedance of the manipulator is realised by driving joint actuators using feedback data implemented into a control algorithm.

Passive impedance maintains desired forces by using various mechanical structures, such as spring and dampers, which provide the desired mechanical impedance.

Hybrid control causes the manipulator end effector to follow a given trajectory while

simultaneously maintaining a desired contact force. In this method force is controlled in one direction and the position in another (orthogonal).

Some methodologies for force control purposes are, among others:

- *feedback linearization*
- *singular perturbation*
- *integral manifold*
- *adaptive techniques*

The *feedback linearization method* requires the determination of a system model for the application, which may cause several run time problems. Exact feedback linearization of a full-order flexible model is not generally possible. A great part of the current research in force/torque control field is concentrated on adaptive techniques, which have many advantages compared to traditional methods. *Adaptive control* schemes based on rigid manipulator models cannot be directly applicable to manipulators with flexible joints. This is because of the presence of unmodelled high frequency dynamics, leading to stability and robustness problems. Some of these methods use high feedback gains to eliminate the effect of nonlinearities and to increase the adaptation speed. This approach often causes the excitation of unmodelled dynamics. Other adaptive methods lack an independent control input for separate degree of freedom.

2.2 Robot Force Control Strategy

Manipulator force control is a complex problem involving various concepts, which should be taken into consideration. These include:

- manipulator and environment dynamics
- manipulator and environment kinematics
- control system hardware

- feedback data acquisition and processing
- cost and system optimization

An appropriate approach to force control algorithm design and implementation can be based on the following steps:

a) Description of the manipulator dynamics:

establishing of the appropriate model describing the behaviour of the manipulator and the environment during task execution.

b) Dynamic-kinematic dependence:

modelling and predicting of generated forces due to the task execution and trajectory motion tracking. Two types of forces are subject of consideration:

active forces - due to joint actuating torques

passive forces - caused by the deformation of interacting elastic bodies.

c) Task execution methodology:

based on a) and b) and system parameters, a strategy for the given task should be established, eliminating possible unwanted actions and phenomena.

d) Control system design:

establishing and implementing an appropriate control algorithm for the given system providing a high performance and robustness.

e) System stability evaluation:

a complex closed loop control system with the sensory feedback may produce stability problems. Therefore, it is necessary to analyze system stability during the task execution.

f) System simulation:

prior to the real time system implementation, it is necessary to analyze and correct, if required, the system overall behaviour and performance under various conditions. Unexpected and undesired effects should be avoided. System simulation serves these purposes and often allows prediction of manipulator malfunction or damage.

g) **System implementation and testing:**

the final stage of the designing process, involves the concept transfer to the real time system. The specific testing procedure should be designed. The obtained results should be compared with those assumed.

2.3 Sensory Feedback in Force Control

The application of force/torque sensing devices in robot control schemes is a very useful concept. A force/torque sensor is an important information source about the force on the manipulator tip and joint torque. Wrist mounted force sensors used in industrial robots allow performance of manipulations, which require the measurement and control of force. The force/torque sensor in a feedback loop can be also a source of system instability. The reasons for the system instability can be related to the following factors:

- feedback data acquisition, interpretation and filtering
- model of plant dynamics
- applied control law

The force information can be obtained by a number of methods. Depending on the technique providing feedback data, these methods can be classified as follows:

- *motor output torque measurement*
- *motor current measurement*
- *wrist sensors*
- *hand sensors*

There exist various robot control methods which implement force/torque sensor feedback, but the number of papers currently published and explicitly investigating advantages and disadvantages of such systems is relatively small.

In [3] Zheng and Fan examine the impact effect on the robotic system with the wrist

installed force sensor (high sensor stiffness). They propose development of an effective contact detection mechanism based on force sensor signals. In this case the sensor structure, global system dynamics and response time play major role. The operational speed of this system should be decreased to avoid possible sensor damage due to the high force impulse at the impact time.

An universal controller for the transient free-to-constraint motion based on force sensory feedback would solve problems occurring at the collision time.

Mills in [4] developed equations of contact motion based on the Lagrange formulation for a rigid link manipulator with a wrist mounted multi-axis force sensor. In this paper a compliant environment is considered and the sensor is assumed to be relatively stiff. It is suggested that the manipulator can be controlled without taking into consideration the high frequency dynamics due to the force sensor, which confirms common engineering practice. Pfeffer, Khatib and Hake in [6] developed and analyze a model of a flexible joint with joint torque feedback. They suggest compensating of joint torque disturbances by sensing the transmitting torque and closing the torque servo around the motor and transmission system. Their custom designed and built strain gage torque sensor adds flexibility and hysteresis to a system. For example, the force/torque sensor located between the bull gear and link 3 of the PUMA 500 manipulator measures the joint torque, neglecting joint friction. Implementation of a similar concept in a direct-drive manipulator can be found in [35], where the joint torque measuring sensor is characterised by the high mechanical stiffness and high sensitivity.

Gonzales and Widmann in [7] demonstrate a nonlinear robot model for force control investigation. In this paper nonlinearities related to slip-stick friction and control signal saturation are reviewed. An explicit force control concept including plant nonlinearities is presented.

Jankowski and ElMarghy in [18] present the control algorithm based on robust servomechanism theory. They implement a force feedback for a two-link, flexible joint planar manipulator interacting with rough constraining surface.

Eppinger and Seering in [22], assuming a constant contact force of a link with its environment, analyze and simulate the dynamic properties of the robot system. They use the feedback from a sensor mounted close to the tool/workpiece interface. They analyze also sources of contact instability related to robot structural dynamics, sensor compliance, and workpiece dynamics. Hogan in [27] and [28] considers the possibilities of robot impedance control involving measurements of joint angles, actuator torques and interaction forces. He also presents some experimental results of contact force sensing. Robert and Craig in [34] describe another, conceptually simple "hybrid" technique. These authors suggest a control architecture in which a wrist mounted force sensor supplies feedback data to the controller. They provide results of a physical implementation of this control scheme on a modified Scheinman Stanford arm equipped with a Scheinman force-sensing wrist.

The implementation of force feedback has a significant influence on the overall system performance as can be seen by comparing two systems, with- and without sensory feedback. (See Fig. 2.3.1.)

Fig. 2.3.1 Control system sensitivity a) without force feedback, b) with force feedback

The sensitivity to the plant parameter changes for two systems, with- and without feedback is compared below.

case a) - open loop control system:

change of plant parameters $\Delta H(s)$ causes the change in the transfer function $H(s) \rightarrow H(s) + \Delta H(s)$ and the output equals:

$$C(s) + \Delta C(s) = [H(s) + \Delta H(s)] U(s) \quad (2.3.1)$$

and the change in the output equals:

$$\Delta C(s) = \Delta H(s) U(s) \quad (2.3.2)$$

case b) - closed loop:

$$C(s) + \Delta C(s) = \frac{H(s) + \Delta H(s)}{1 + H(s) + \Delta H(s)} U(s) \quad (2.3.3)$$

$$\Delta C(s) = \frac{\Delta H(s)}{1 + H(s)} U(s) \quad (2.3.4)$$

$H(s)$ is usually large. Comparing a) and b), the advantage of the force feedback application in case of plant parameters variation is clear.

2.4 Force/Torque Control Methods

The general classification of force/torque control methods is given by Yoshikawa in [1] (mentioned in chapter 1.1). The growing interest in the force control field is observed in the contemporary robotics and control system literature. Many, sometimes very sophisticated algorithms analytically solve the force/torque control problem in the currently published papers.

There are few general solutions for the force/torque control problem which achieve *stabilisation, robustness, asymptotic reference tracking and elimination of undesired noise*. Not all of them can be easily and successfully implemented into existing flexible joint robot designs. While position controllers for such manipulators are fairly well developed, efficient force/torque control still requires much of effort.

Very few solutions exist for the control of manipulators with mechanical compliance operating under constrained conditions. Below, some of the more interesting force control methods developed in the past few years for flexible joint manipulators are mentioned. In [6] a force compensator with joint torque sensory feedback is designed and tested for the third joint of PUMA 500 manipulator. Some experimental results are provided demonstrating significant reduction in the effective friction (97%).

Khorasani in [14] shows the possibilities of a modification of standard rigid adaptive control schemes, which can be applied to control a flexible robot joint by introducing an appropriate correction. In this paper a robust adaptive controller for a two-link flexible joint manipulator is designed and simulated. Comparison of two standard methods: *adaptive inverse dynamics scheme* and *Slotine and Lee scheme*, which are not directly applicable for the type of problem considered, are also demonstrated.

The objective of the *adaptive inverse dynamic scheme* is the linearization and decoupling of a closed loop system even in the presence of parameter uncertainty.

The *Slotine and Lee scheme* preserves the passivity property of a robot structure without parameter linearization.

Su, Leng and Zhou in [15] propose a *sliding mode* controller for force/motion tracking of a two link revolute manipulator under constrained conditions.

In [16] some experimental results of the adaptation of the rigid control laws to flexible joint robots with a weak elasticity are given. Simple corrections in these schemes allow their successful application and damping of undesired oscillation caused by joint elasticity.

Jankowski and ElMaraghy in [18] examine a version of a nonlinear decoupling and linearizing controller based on robust servomechanism theory. They derive a hybrid control algorithm for a two-link planar manipulator interacting with the environment. They also provide simulation results reflecting the time characteristics of normal and tangent reaction forces and control torque.

In [19] Chen and Fu propose a *two-stage adaptive force controller* for a flexible joint robot with an unknown stiffness value. Simulation results demonstrate the effectiveness of this concept as systems exhibit the stability and zero force error tracking. Some other interesting force control laws are given in [27].

2.5 System Stability

System stability depends on several factors, such as the joint flexibility, motor and transmission mechanism friction, force sensor characteristics, system dynamical model, control system design. It is essential that system stability in any operation condition is assured. Stability problems arise in force feedback control actions of flexible joint robots.

Mills in [23] investigates the influence of manipulator structural compliance on the stability under constrained motion conditions. Readman and Belanger in [24] assuming certain conditions show a gain scheduled velocity control law asymptotically stabilizing fast dynamics related to the joint flexibility. System stability in the continuous and discrete time domains is considered below.

• *Continuous time stability:*

Consider a closed loop system with the characteristic equation given by:

$$1 + KH_c(s)H_o(s) = 0 \quad (2.5.1)$$

where: $H_c(s)$ - controller transfer function, $H_o(s)$ - joint open loop transfer function, defined as follows:

$$H_o(s) = \frac{B(s)}{A(s)} \quad (2.5.2)$$

$A(s)$, $B(s)$ are polynomials with orders of n , m and $n \geq m$. Substituting for $H_c(s) = \alpha/(\alpha + s)$ into 2.5.1 (low pass filter) yields:

$$1 + \alpha \frac{KB(s) + A(s)}{sA(s)} = 0 \quad (2.5.3)$$

For a fixed K and varying α root locus theory can be applied to equation 2.5.3 and the following Lemmas are satisfied

Lemma 2.5.1 The root loci given by 2.5.3 is continuous

Lemma 2.5.2 For $\alpha \rightarrow 0$, poles of equation 2.5.3 converge to zero

Lemma 2.5.3 The departure angle of the root loci from $s = 0$ is π , for α increasing from zero, pole at zero moves along the real axis in the negative direction

Lemma 2.5.4 For $\alpha = \infty$, poles of equation 2.5.3 are equal $-\alpha$ and the closed loop poles of equation $1 + K H_c(s) = 0$ converge to zero

Theorem 2.5.1

For the stable poles of $H_o(s)$ a parameter $\alpha_c > 0$ can be found, that for all $\alpha > \alpha_c$ the closed loop system stability is achieved

• *Discrete time stability.*

Adding to the feedforward path a digital filter with the transfer function defined by

$$H_c(z) = \frac{zT\alpha}{z - e^{-\alpha T}} \quad (2.5.4)$$

the closed loop characteristic equation of the system is given by

$$1 + KH_c(z) z [H_{zoh}(s) H_o(s)] = 0 \quad (2.5.5)$$

or after some standard mapping operations equation 2.5.5 will have the form

$$1 + KH_c(z) (1 - z^{-1}) z \left[\frac{H_o(s)}{s} \right] = \frac{\Gamma(z)}{\mathbf{E}(z)} \quad (2.5.6)$$

Γ and $\mathbf{E}$ are polynomials in z of orders m, n respectively and $m < n$. Because $\alpha T < \pi < 1$, equation 2.5.5 can be written as

$$1 + \alpha \frac{Tz\Gamma(z) + T\mathbf{E}(z)}{(z-1)\mathbf{E}(z)} = 0 \quad ; \quad e^{-\alpha T} = 1 - \alpha T \quad (2.5.7)$$

Theorem 2.5.1 modified to the discrete time domain can be now applied. Knowing that

root locus theory is applicable to equation 2.5.7, the system stability is proved.

Applying the above theory to a second order flexible joint system and assuming the underdamped joint dynamic model as:

$$H_o(s) = \frac{K_g}{J_m s^2 + B_m s + K_g} \quad ; \quad \Delta = B_m^2 - 4J_m K_g < 0 \quad (2.5.8)$$

the transfer function given by equation 2.5.8 can be presented as:

$$H_o(z) = z [H_{zoh}(s) H_o(s)] = \frac{\alpha_1 + \alpha_2 z}{z^2 - 2e^{-\alpha T} \cos(\beta T) z + e^{-2\alpha T}} \quad (2.5.9)$$

where

$$\alpha = \frac{B_m}{2J_m} \quad ; \quad \beta = \frac{\sqrt{(4J_m K_g - B_m^2)}}{2} J_m \quad (2.5.10)$$

$$\begin{aligned} \alpha_1 &= e^{-\alpha T} + \frac{\alpha}{\beta} e^{-\alpha T} \sin(\beta T) - e^{-\alpha T} \cos(\beta T) \\ \alpha_2 &= 1 - \frac{\alpha}{\beta} e^{-\alpha T} \sin(\beta T) - e^{-\alpha T} \cos(\beta T) \end{aligned} \quad (2.5.11)$$

From equations 2.5.6 and 2.5.9 the OLTF (open loop transfer function) is obtained and the root locus can be plotted for given numerical data. For the design parameter α poles of the system should be located in the unit circle.

2.6 Robot Joint Flexibility

Joint flexibility is a phenomenon caused by the elasticity existing in two major manipulator components: joint actuator (gear or harmonic drive elasticity) and transmission mechanism. Traditional robot designs are based on relatively rigid constructions, which simplify efforts related to the control system design.

The elasticity in robot joints has two different meanings in robotics. From one point of view it is highly undesired, because of creating major problems in an efficient manipulator operation. It can cause system instability (end effector "chattering"), limiting the precision and the operation speed. On the other hand a mechanical compliance caused by the joint flexibility allows development of control systems predicting robot collision with an interacting environment. Therefore, the manipulator damage can be often avoided. The stiffness of typical robot components given is in Table 2.6.1.

Table 2.6.1. Stiffness of typical robot joint components.

JOINT COMPONENT	STIFFNESS (Nm/rad)
Steel joint	$\sim 10^4$
Harmonic drive	$\sim 10^4$
Torque transducer	$\sim 10^5$
Flexible coupling	$\sim 10^3$

Joint flexibility is very often modelled as a torsional spring with a linear characteristic. A usual model representation of joint flexibility consist of a set of equations for two coupled dynamic systems: slow dynamics of a standard rigid body and fast dynamics caused by joint flexibility.

In [2] a mathematical model of a single-link flexible joint manipulator is given.

Fig. 2.6.1. H6D160 harmonic drive flexibility and a standard representation of joint stiffness.

In this model joint elasticity is a constant parameter. A conceptually similar model can be found in [19]. Readman and Belanger in [24] present a flexible model of a robot assumed to be an open kinematic chain with only revolute joints. The influence of link and drive parameters in this model on fast subsystem dynamics is reflected. In this work some examples are reviewed, including flexible joint manipulator PUMA 560. Drive inertia effects on control laws and the stabilisation of system fast dynamics are analyzed. Description of the similar problem for a three joint robot design can be found in [37]. In this paper the fast subsystem dynamics and its dependence on the geometric configuration are examined.

Luh in [32] provides a representation of a single joint structural flexibility and formulates its relationship with a restoring torque due to the joint stiffness. He computes the joint structural resonant frequency and gives an example for the Stanford manipulator. The simple dynamic model of a revolute manipulator with flexible joints is given in [44].

This model is an extension of a one-joint, two inertia system to more complex n-joint system. In this model the joint stiffness independency of the gear ratio is assumed.

Hung and Weng in [5] present a modal analysis of a multilink system with flexible joints and links. Their formulation is based on a finite element transfer matrix scheme, where each component is divided into certain number of subsegments (finite elements) connected by transfer states.

Another standard from the conceptual point of view representation of the joint elasticity is given by Lin, Tosunoglu and Tesar in [43].

Spong in [36], using a singular perturbation approach, studies the effect of weak joint elasticity. He examines an adaptive control scheme designed for a rigid robot structure. In this work joint compliance caused by a harmonic drive and an output shaft is measured directly by two high resolution incremental encoders. These encoders are mounted on the input shaft and the harmonic drive output shaft respectively (see Chap.3 for details). Similar joint compliance measuring approach can be found in [16]. Considered in this work harmonic drive H6D160 exhibits a varying joint flexibility K_g , which depends on the actual output torque developed by the joint. Two value of K_g are given. For torques smaller than 20% of the maximum joint load $K_g = 23000$ [lb-in/rad] (2599 Nm/rad) and for greater load torques the stiffness increases rapidly over 5 times and reaches 120000 [lb-in/rad] (13560 Nm/rad) (See Fig. 2.6.1). This fact has a significant meaning for the system behaviour and was a subject of a careful analysis.

2.7 Friction in Force Control

The type of friction often called slip-stick friction is an important, nonlinear parameter causing several problems in control system design and its implementation for robotics application. Among negative effects of friction can be instability and steady state error. Three major sources of friction relevant to the analyzed topic are considered in this work: motor, joint and load friction. Three cases of slip-stick friction existence are shown in Fig. 2.7.1.

Fig. 2.7.1 Friction in contact of two rigid bodies.

a)	b)	c)
$F_s = P_x; F_s \leq \mu_s N; N = P_y + W$	$F_{sm} = P_x; F_{sm} \leq \mu_s N; N = P_y + W$	$F_k = P_x; F_k \leq \mu_k N; N = P_y + W$
F_s - static friction	μ_s - static friction coefficient	
F_{sm} - max static friction	μ_k - kinetic friction coefficient	
F_k - kinetic friction	P - acting force	
W - load force	P_x, P_y - components of the acting force	

The authors of [38] and [39] implement in a control algorithm sliding friction of an end effector constrained by a rigid surface. An approach presented in [39] simplifies the friction phenomenon by neglecting its dependence on the magnitude of a normal reaction force.

Friction forces are manifested by tangential forces on a body with a substantial component of the applied force acting normally against a surface. An interaction of surface layers of two bodies results in friction, which is made up of a number of phenomena, e.g. adhesion of surface atoms.

Properties of friction.

- friction coefficients are independent of the area of interaction
- friction coefficients depend on the nature of two surfaces
- μ_s is nearly independent of the time of the contact
- μ_k is nearly independent of the relative velocity

To achieve the maximum performance of a system, all these properties should be taken into consideration, in fact in control system theory they are very often neglected.

Figure 2.7.2 represents slip-stick friction graphically.

Fig 2.7.2 Graphical representation of slip-stick friction

2.8 Force on the Manipulator Tip

The first force control design step should include a description of forces generated during a manipulator contact with an environment. Two categories of forces are considered: the desired and the actual generated forces, which are a result of several kinematic and dynamic factors. In the problem under consideration (contact force control) the robot end effector is in contact with an ideal rigid surface. The designed force controller should maintain constant contact force with the constraint surface. A manipulator having n degree of freedom (DOF) is described by:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + \gamma(q) + J^T(q)F + W(t) = \tau \quad (2.8.1)$$

where: q is the joint displacement vector, $M(q) \in \mathbb{R}^{n \times n}$ is the inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^n$ is the vector characterizing Coriolis and centrifugal forces, $\tau \in \mathbb{R}^n$ is the applied joint torque, $\gamma(q) \in \mathbb{R}^n$ is the gravitational load of the robot, $W(t)$ is the external disturbance, $J(q) = \partial x(q)/\partial q \in \mathbb{R}^{m \times n}$ is the Jacobian matrix, which is assumed to be non-singular in the finite work space Ω , and $x \in \mathbb{R}^m$ is the position and Euler angles of the end-effector in the Cartesian space. $F \in \mathbb{R}^m$ is the vector of forces/moments exerted on the environment by the robot end-effector.

The environment is described by a set of m rigid, independent hypersurfaces [9]:

$$\Phi(x) = [\Phi_1(x), \dots, \Phi_m(x)]^T \quad m \leq n \quad (2.8.2)$$

$\Phi_i(x)$ is twice differentiable with respect to x .

The contact force F is given by:

$$F = D^T(x) \lambda + v f_t(\mu, v, \lambda) \Delta \Gamma(x, \mu, v) \lambda \quad D(x) = \frac{\partial \Phi(x)}{\partial x} \quad (2.8.3)$$

where $\lambda \in \mathbb{R}^m$ is a vector of Lagrange multipliers associated with the constraints representing normal contact force components, f_t is the friction force dependent on the unknown friction coefficient μ . The direction of f_t is a function of the end-

effector velocity v_e , $D^T(x) \lambda$ represent normal contact force in Cartesian space, v is the unit tangent direction of the surface. For the constrained motion only $(n-m)$ coordinates need to be controlled, and m components f of the contact force F should be controlled according to:

$$f = T(x) F ; \quad f \in R^m ; \quad T \in R^{m \times n} \quad (2.8.4)$$

From equation 2.8.3 we get:

$$\lambda = T(x) \Gamma(x, \mu, v)^{-1} f ; \quad F = \Gamma(T \Gamma)^{-1} f \quad (2.8.5)$$

Introducing a nonlinear transformation described in [9] and substituting into equation 2.8.1, assuming conditions given in 2.8.2, we get the following dynamic equation :

$$M(x) \ddot{x} + C(x, \dot{x}) \dot{x} + \gamma(x) + \bar{F}(x, t) + B(x) f = J_q^{-T} x \quad (2.8.6)$$

$$x = [0 \quad x_p^T]^T$$

and the components of this equation are defined as:

$$M(x) = J_q^{-T}(q) M(q) J_q^{-1}(q) \quad (2.8.7)$$

$$C(x, \dot{x}) = J_q^{-T} C(q, \dot{q}) J_q^{-1} - J_q^{-T} M(q) J_q^{-1} \dot{J}_q J_q^{-1}$$

and the following relationships are valid:

$$\gamma(x) = J_q^{-T} \gamma(q) ; \quad \bar{F}(x, t) = J_q^{-T} \bar{F}(t) ; \quad B(x) = J_x^{-T} \Gamma(T \Gamma)^{-1} \quad (2.8.8)$$

The general relationship between the joint torque $\tau = [\tau_1, \dots, \tau_n]$ and the force on the manipulator tip F_t in the end-effector frame Σ_t and moment n_t with respect to the origin of Σ_t is given by:

$$\tau = J^T [F_t \ n_t]^T$$

For the considered revolute joint manipulator configuration with the angular joint displacement θ the end-effector frame orientation is described by vector r , given by:

$$r = [x_E, y_E, z_E]^T ; \quad q = \theta \quad (2.8.9)$$

$$x_E = \cos(\theta) ; \quad y_E = \sin(\theta) ; \quad z_E = h \quad (2.8.10)$$

The contact force F in equation 2.8.1 can be considered as either the desired force F_d or the actual contact force F_a exerted by the manipulator. A direct expression for the desired joint driving torque τ denoted as M_d can be found in Chap. 2.8.1. The graphical presentation of the desired force F_d and the actual force F_a on the manipulator tip shown are in figure 2.8.1 and 2.8.2 respectively. The designed controller should compensate for the generated force error $F_e = F_d - F_a$.

2.8.1 Desired Contact Force

The relationship between the desired joint torque M_d and the desired contact force F_d is given by (Fig 2.8.1):

$$F_n = F_d \sin \alpha + F_{KL} \cos \alpha ; \quad M_d = [F_d \sin(\alpha) + F_{KL} \cos(\alpha)] L \quad (2.8.11)$$

where F_{KL} - is load friction, α - is the actual angle between the constraint surface and the link axis of symmetry, L is the total link length, F_{KL} is a nonlinear friction, the value of which is difficult to estimate and depends on conditions of the contact. If there is no link motion, F_{KL} is a static load friction and can have a wide range of values. It is not possible in this case to calculate its exact magnitude (See Chap. 2.7).

Fig. 2.8.1 Desired contact force on the manipulator tip

2.8.2 Actual Force on the Manipulator Tip

We denote M_d , M_t , F_x , F_z as the actual applied joint torque, moment about the y axis, and forces measured by the JR3 sensor along x and z axis. Thus, the following relationship is valid (Fig. 2.8.2)

$$F_x = F_z \sin \alpha + F_y \cos \alpha \quad (2.8.1.1)$$

and the actual developed moment equals

$$M_d = M_t + F_x L = F_{yt} L \quad (2.8.1.2)$$

Fig 2.8.2 Actual measured contact force on the manipulator tip.

We can rewrite above equations in the matrix form:

$$\begin{bmatrix} F_x \\ F_y \\ M_y \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ \sin(\alpha) & -\cos(\alpha) & 0 \\ \cos(\alpha)d & \sin(\alpha)d & 1 \end{bmatrix} \begin{bmatrix} F_{KL} \\ F_a \\ 0 \end{bmatrix} \quad (2.8.1.3)$$

From equation 2.8.1.3 the advantage of the force sensor application to measure the contact force is obvious. The ability to measure on line F_x , M_y , F_y , allows calculation of slip-stick friction F_a (equation 2.8.1.4). This property is used further in designed force control laws

$$\begin{cases} F_{KL} = F_x \cos(\alpha) - F_y \sin(\alpha) \\ F_a = \frac{F_x [\sin(\alpha) \cos(\alpha) - F_y \sin(\alpha)] \cos(\alpha)}{\sin(\alpha)} \end{cases} \quad (2.8.1.4)$$

Chapter 3

EXPERIMENTAL SETUP DESCRIPTION

The manipulator and control system components are chosen to allow conducting various tests evaluating system component performance. Conditions are established to implement designed control algorithms and to investigate the manipulator end effector behaviour in both constrained and not constrained environment. The planar revolute joint configuration seemed to be an optimal for these purposes. Such a design allows to neglect the presence of gravity in the system dynamics.

The one-degree-of freedom SLIM-1 test manipulator shown is in Fig. 3.1. It consists of a single revolute robot joint, actuated by a DC servomotor with a harmonic drive.

This configuration can be easily extended to more DOF. To the end of the rigid link is attached the JR3 force/torque sensor. Angular displacements of the motor and joint are measured by two incremental encoders. The placing of an additional encoder on the gear motor output can be highly useful for the joint flexibility and joint friction investigation. Attempts were made to directly implement the encoder output data into the control system. Performed tests on the SLIM-1 manipulator revealed, that an application of such concept to a direct joint flexibility measurement is not recommended as it is difficult to establish a relationship between the joint deflection and measured joint torque.

Some of calibration tests performed on the system components are given in Appendix IV. The system component interconnection is shown in Fig. 3.2.

Fig. 3.1 Experimental set up - SLIM-1 manipulator.

Fig 3.2 Experimental set up - component connection

The control system and data acquisition components are an *IBM PC compatible KINTEAK computer* (80386 microprocessor, 17.5 MHz), *analog-to digital and digital-to analog signal converter, encoder interfacing card, and universal JR3 force torque measurement system*. The joint control simulation, real time control and data acquisition are realised by software designed in Matlab and Microsoft C.

The following paragraphs give detailed description of system components.

Fig. 3.3 presents the diagram of the force/torque test.

Fig. 3.3 Force/torque control test diagram

To simplify the system dynamics, eliminate undesired external disturbances and noise, and focus on the phenomena of the major interest ("chattering", system parameter nonlinearity), the following assumptions related to the experimental set up were made:

- 1 *a rigid link structure*
- 2 *a flexible joint structure*
- 3 *the link rotational moment of inertia is minimized*
- 4 *external sources of vibration and disturbances are eliminated*

Assumptions 1 and 3 are opposed to each other, so that appropriate steps should be done to satisfy the following conditions: a minimum link deflection, a small link mass and a low link rotational moment of inertia

The condition 4 is fulfilled by the fact that the joint motor configuration is placed on a non vibrating Research Series Table, the product of Newport Co

The test system is organized in three major groups (See Fig. 3.4)

- joint/link/JR3 sensor
- control system
- data acquisition and control software

Fig. 3.4 Force control test components

3.1 Joint Actuator

The motor used to actuate the manipulator joint is a high performance DC ServoDisc gear motor, model S6M4H/H6D160, product of PMI Motion Technologies. The motor ratings are given in Table 3.1. This motor is characterized by:

- high torque-to-weight ratio
- smooth constant torque
- fast acceleration and deceleration

The armature of the ServoDisc motor is a low inertia, flat, ironless disc. Full torque is delivered over the entire speed range of zero to 4000 rpm.

Table 3.1. Joint actuator parameters - DC harmonic drive motor S6M4H

PARAMETER	VALUE	UNIT
rated speed	18.8	rpm
continuous torque	16.0	Nm
peak torque	45.0	Nm
gear ratio	160.0	-
motor voltage	17.5	V
motor current	5.0	Amps
torque constant	3.01	N-cm/Amp
friction torque	.64	N-cm
damping constant	.11	N-cm/krpm

The harmonic drive of the gear motor has a virtually zero backlash for high positional accuracy. The gear motor has two values of stiffness, for lower and higher torque ranges. The DC motor is configured with the S6T tachometer and M23 incremental encoder.

Fig 3.1.1 DC motor torque characteristic

The control commands to the motor are send from the VXA 48-8-16 servo amplifier. The amplifier set in torque mode supplies a current i_a proportional to voltage input commands. The motor operating current i_a is given by

$$i_a = \frac{\tau + F_{km} + \frac{B_m \theta_m}{1000}}{K_m} \quad (3.1.1)$$

τ - shaft torque

The maximum motor current is limited to 5 A. The load inertia J_{lm} and the friction torque T_{fm} seen by motor are given by

$$J_{Lm} = J_{Lg} + \frac{J_L}{N^2 v_g} ; \quad T_{fm} = \frac{T_L}{N v_g} \quad (3.1.2)$$

where v_g - gear efficiency, J_{Lg} - gear box inertia (with respect to the input), T_L - load friction torque

3.2 Manipulator Link Modelling

The manipulator link stiffness plays an important function in the considered system, especially for a higher range of applied joint torques. The link should have sufficiently high stiffness so as to allow joint flexibility to dominate elastic effects. Various cross-sectional link designs have been reviewed for both steel and aluminium alloys. The hollow circular cross section was chosen as having a small rotational moment of inertia and high stiffness. Various common manipulator link profiles are shown in Fig. 3.2.1

Fig. 3.2.1
Manipulator
link profiles.

Dynamic and static loads were considered while designing the link. The dynamic force load magnitude, due to the relatively high total inertia of the link, JR3 sensor and load reach significant values and should be controlled to avoid joint hardware damage. The static deflection δ_y of the link end with the profile 4 in Fig. 3.2.1 is given by:

$$\delta_y = \frac{F_{\max} L^3}{3EI_L} \quad ; \quad I_L = \frac{\pi}{64} (D^4 - d^4) \quad (3.2.1)$$

where $E = 9.9 \times 10^6$ [psi] is the Young modulus for an aluminium alloy. The maximum link deflection calculated for the normal force 45 N, acting on the link tip, equals .1 mm. An aluminium alloy material was chosen for the profile No. 4 shown in Fig. 3.2.1, $D = 2"$, $d = 1.87"$, and the thickness $t_h = .065$ (See Appendix 1 for the link design).

3.3 Joint Speed and Position Sensing Devices

The joint angular velocity is measured by means of a tachometer attached to the motor shaft. The tachometer produces an analog signal whose magnitude is proportional to the DC motor velocity. The tachometer constant $K_t = 0.75 \text{ V/krpm}$. The tachometer signal linearity referred to 3600 rpm is 0.06%. The maximum velocity achieved during tests does not exceed 3000 krpm. The tachometer resolution depends on the 12 bits resolution of the analog to digital converter and equals 0.995 rev/s. See Table 3.3.1 for tachometer parameters. The tachometer signal is sent via a shielded wire and filtered by an analog 500 Hz low pass filter (See Fig. 3.3.1).

Table 3.3.1 Tachometer parameters

PARAMETER	VALUE	UNIT
tachometer constant	75	V/krpm
moment of inertia	18	kg-cm ²

Fig. 3.3.1 Tachometer signal Low Pass Filter

The tachometer analog signal, after filtering, is digitized by the RT1815 A/D converter, filtered by a digital 200 Hz low pass filter and implemented into the control scheme. Another means of motor velocity measurement is the integrating of the motor angular position. Both methods were implemented and the noted result discrepancy is minimal.

Position measurement

To measure the angular displacement of the joint and evaluate the joint compliance, two incremental, high resolution (2540 lines/rev) encoders are used. One is attached to the motor shaft, the other is mounted on the output side of the gear motor.

Principle of encoder operation The rotation of the motor shaft is translated onto interruptions of a light beam, which are then amplified and output as electrical pulses. The direction of rotation is determined by observing which of channels is a leading waveform. The encoder board circuitry is counting the rising and falling edges of both the A and B signal channels. Encoder output signal is shown in Fig 3.3.2.1.

Fig. 3.3.2.1
M 23
Incremental
Encoder
signal.

The encoder angular resolution E_k is given by

$$E_k = \frac{360}{Nnr} \quad \text{where } n = \text{number of rising and falling edges per cycle}$$

$$N = \text{gear ratio}$$

$$r = \text{number of electrical cycles}$$

For given encoder $n = 4$, $N = 160$, $r = 2540$, thus the resolution $E_k = 0.00022146$ [deg count] (given with respect to the joint output). Two encoders are used to measure the joint compliance. Attempts were made to calculate the joint stiffness from the difference ΔX between the actual link and motor position, X_l and X_m respectively. The information about the normal force F_n acting on the link measured by the force sensor is given by

$$\Delta X = X_l - X_m = \frac{1}{k} F_n \Delta X$$

Series of tests were performed, but the varying friction in the motor housing bearings, not

considered in the above equation and the difficulty of its measurement make the exact evaluation of the joint stiffness virtually impossible. Figure 3.3.2.2 shows the concept of the joint compliance measurement.

Fig. 3.3.2.2 Flexible joint compliance measurement

3.4 Servo Motor Amplifier

The specifically designed, pulse-width modulated servo amplifier, model VXA 48-8-16, PMI Motion Technologies is used as a driver for the DC gear motor. It may be operated in both velocity and torque mode, with- and without tachometer feedback. The operation mode is selected by means of jumpers on the circuit board. The amplifier is powered by a transformer with a current limiter. A master switch allows to disable the system in case of an emergency. The minimum amplifier gain $K_a = 1.6 \text{ Amp/V}$ and the time constant T_a equals $5.63 \cdot 10^{-4} \text{ s}$. The maximum amplifier current is set on 80% of the motor continuous current (4 Amp). The maximum continuous amplifier current is $I_{amax} = 8 \text{ Amps.}$, for the input command 2 V. For force/torque control tests, the amplifier is modelled in torque mode, providing a linear relationship between an input control voltage and motor current, thus motor torque. Calibration tests performed on the amplifier are presented in Appendix IV.

3.5 Force/Torque Sensor

To obtain an information about the joint torque and the actual contact force on the manipulator tip is mounted a six axis *JR3 force/torque sensor*. It is connected with the JR3 Intelligent Support System, which contains signal conditioning board, data acquisition board and processor board (see Fig 3 5 1). The analog sensor data is digitized by the R11-815 analog-to-digital converter. The speed of data processing and acquisition is given by the time base (data sampling speed). This rate is kept at 200 Hz. Higher rates for this sensor are recommended under some conditions, such as digital filtering disabling. The sensor digital resolution is limited to 12 bits (4.48 mV or 59 N for the sensor z axis).

Fig 3 5 1 JR3 force/torque measurement system.

The sensor requires careful calibration to eliminate random noise. The implementation of several linear noise filters failed, because of the variable sensor dynamics. In the first stage of the experiment a serial communication with the sensor was performed, using the serial port RS 232, set on 9600 baud/no parity/8 data bits/1 stop bit. The use of an analog data sensor option allows to increase the communication speed compared to the serial communication. The analog force information from the sensor is read by the R11-815 A/D input channel, digitized and implemented into the control algorithm. Initialization, calibration and all sensor adjustments are done using the serial port RS 232 by issuing adequate software commands. The orientation of the sensor frame is presented

in Fig. 3.5.2. The 6 degree-of-freedom sensor allows to measure forces and moments about x, y and z axis.

The stiffness vector of the sensor, k , is described by

$$k = [k_x \ k_y \ k_z \ p_x \ p_y \ p_z]$$

where

$$k_x = 1.8 \times 10^{-4} \text{ lb/in}$$

$$k_y = 1.7 \times 10^{-4} \text{ lb/in}$$

$$p_x = 2.3 \times 10^{-4} \text{ lb/in}$$

$$p_y = 8.0 \times 10^{-4} \text{ lb/in}$$

Fig. 3.5.2 JR3 force torque sensor coordinate system

The maximum sensor load ratings are given in Table 3.5.1.

Table 3.5.1 JR3 force sensor loads

F_x	F_y	F_z	M_x	M_y	M_z
[N]			[N-m]		
685.5	685.5	1371	68.4	68.4	68.4

Sensor load limits are sufficient for force control test purposes. The maximum static force applied in control tests is 20 N. To achieve a better measurement accuracy a sensor with 15 bit resolution could be used (0.049 N for z axis). This would require at least 15-bit analog-to-digital signal converter. The calibration procedure of the JR3 sensor is given in Appendix IV.

3.6 Control System Component Description

In this Chapter all control system components are briefly described, including computer and data acquisition devices. The apparatus used in experiments sets the limitation on the controller bandwidth to maximum 250 Hz. The natural frequency ω_n of the open loop dynamics of the considered flexible joint equals 2620 Hz (16453 rad/s). Therefore, the analysis of the joint high frequency behaviour requires a much faster control system components. Due to the above limitation, the proper choice of the sampling time in simulations and experiments was a fundamental problem. For the simulation of a theoretical system assuming a high controller bandwidth, Shannon's sampling theorem was used, and $\omega_s = 2\omega_n$, where ω_s is the sampling natural frequency. For the control simulation and implementation of the existing model the pure signal-processing method was applied and the maximum available sampling frequency for the given control method was used. Such approach allows simply to record a signal and recover it from its sample. Because the signal sampled contains frequencies that are higher than half of the sampling frequency, in simulation results of the real system the effect of aliasing can be observed. Antialiasing prefilters could be used to prevent the aliasing effect, but this would slightly decrease the ability to analyze the high frequency dynamics.

3.6.1 IBM PC Computer

The system control, data storage and data acquisition are provided by an IBM PC compatible KINTFAK computer, equipped with 80386 microprocessor, 92 MB ROM memory HD, 640 Kb RAM base and 3 Mb of extended memory, 17.5 Mhz clock speed. The computer internal time function resolution with the data monitoring $T_{im} \approx 2$ ms. The monitor interrupts significantly decrease the system speed. Therefore, control algorithms are run without instant data monitoring to increase the operational speed. The test data is stored temporarily in the computer RAM memory and saved to a data file after the program run time.

3.6.2 RTI Multifunction Input/Output Board

The data acquisition and controller digital commands conversion to the analog voltage signals are realised by the RTI 815 Multifunction board, plugged into computer expansion slot. This board is mapped into IBM PC input/output structure as a block of 16 consecutive bytes. The RTI 815 board has the following capabilities: analog input/output, digital input/output, and time related input/output. RTI 815 board was chosen for the test purposes because of its fast conversion time, relatively high resolution for the full scale of applied controller commands, short settling time of the digital to analog conversion and voltage range compatibility with all other devices used in experiments. The analog to digital conversion resolution is 12 bits (4096 counts for a full scale, which provides a least significant bit (LSB) value of 4.88 mV when in the ± 10 V range) and conversion time 25 μ s. Input signals in range of ± 10 [V] can be measured, what is consistent with maximum voltage signals produced by the JR3 force/torque sensor. A digital to analog capability of the RTI 815 board allows to convert digital computer commands into analog control signals. The conversion resolution is 12 bits (LSB has value of 4.88 mV in the ± 10 V range) and the settling time 20 μ s.

3.6.3 Encoder Interfacing Card

Encoder signals are input and decoded by the 5312 Quadrature Encoder Input Card, inserted into computer expansion slot. Four incremental encoders can be operated with the card. For each encoder provided is a 24 bit counter, which can be used for quadrature decoding, pulse and direction input counting and pulse input up/down counting. To communicate with this card it was necessary to design customized software based on the software package supplied with encoder card. Output from the card for the PC bus is available in binary or binary coded decimal format. The interface to 24-bit counters from the PC bus is handled by multiple 8-bits reads and writes. The card is powered by the PC bus 5 Volts. With the encoder card the software library is supplied, compiled with the designed control programs.

Chapter 4

FLEXIBLE JOINT DYNAMICS MODELLING

In this Chapter the dynamics of a flexible joint operating under constrained conditions is modelled and simulated. Models of a flexible joint dynamics both with- and without Coulomb friction are presented. The joint dynamics is modelled both as a time invariant system with a constant joint stiffness and system with two values of the joint flexibility. A standard dynamic model of a n-link manipulator is considered and adapted to an one-degree-of freedom case. While the joint stiffness is relatively large compared to other system parameters, the joint damping is very small. The influence of parameter nonlinearities on the system behaviour is investigated.

Three dynamical models of a flexible joint are considered:

Model I - *time invariant, simplified model*

Model II - *system with toggling joint stiffness, the transfer function depends on the actual value of the output torque. Assuming the ideal rigid link, the joint output torque can be calculated and the appropriate joint stiffness k_s is chosen.*

features

- ☐ system transfer function is updated at each sampling time T
- ☐ motor friction is constant, dependent on motor velocity
- ☐ all system states are measurable
- ☐ force feedback provides information about the joint output torque
- ☐ system parameters are known a priori

Model III - *time invariant system with a state observer used to estimate unknown system parameters (e.g. joint friction).*

4.1 Defining the System Dynamics

The general equation of a manipulator dynamics is as follows

$$\mathbf{M}(\mathbf{x}) \ddot{\mathbf{x}} + \mathbf{D}(\dot{\mathbf{x}}) \dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{U}(t) + \mathbf{J}^T \mathbf{F} \pm \mathbf{W} \quad (4.1.1)$$

where:

$\mathbf{M}(\mathbf{x})$ - manipulator inertia matrix

$\mathbf{D}(\dot{\mathbf{x}})$ - damping coefficient matrix

$\mathbf{K}$ - stiffness matrix

$\mathbf{J}$ - manipulator Jacobian

$\mathbf{U}(t)$ - joint driving torque matrix

$\mathbf{W}$ - joint Coulomb friction matrix

$\mathbf{x}$ - vector of joint displacements

$\mathbf{F}$ - force on the manipulator tip

The $\pm$ sign for the Coulomb friction expression reflects the relationship below

$$W_i = |W_i| \operatorname{sgn} \frac{dx_i}{dt} \quad (4.1.2)$$

where: $|W_i|$ is the magnitude of the Coulomb friction for a single joint coordinate, the function sgn evaluates the joint velocity sign. In Fig. 4.1.1 the detailed block diagram representation of the manipulator dynamics is shown. Figure 4.2 provides a schematic concept of the joint flexibility.

Fig. 4.1.1 Flexible joint dynamics representation

Fig. 4.1.2 Flexible joint representation.

4.1.1 Classical Representation of a Flexible Joint System

Based on equation 4.1.1 dynamic equations for a one DOF revolute joint manipulator are developed (assuming that M and D are constant, we linearize the system):

$$J_m \ddot{\theta}_m + B_m \dot{\theta}_m + K_g (N\theta_L - \theta_m) \pm F_{km} = U \quad (4.1.1.1)$$

$$J_l \ddot{\theta}_l + B_l \dot{\theta}_l + NK_g (N\theta_L - \theta_m) \pm F_{kl} + T_L = 0 \quad (4.1.1.2)$$

where

J_l - load inertia	B_l - load damping	T_L - external load torque
J_m - motor inertia	N - gear ratio	F_{kl} - load Coulomb Friction
F_{km} - motor Coulomb friction	U - joint driving torque	
θ_m - motor angular displacement	θ_l - link angular displacement	

The Laplace transformation of the above equations yields

$$s^2 J_L \theta_L(s) + s B_L \theta_L(s) + N K_g (N \theta_L(s) - \theta_m(s)) \pm F_{KL}(s) + T_L(s) = 0 \quad (4.11.3)$$

$$s^2 J_m \theta_m(s) + s B_m \theta_m(s) - K_g (N \theta_L(s) - \theta_m(s)) \pm F_{Km}(s) = U(s) \quad (4.11.4)$$

Transforming the above equations yields

$$\alpha_L(s) \theta_L(s) - N K_g \theta_m(s) \pm F_{KL}(s) = T_L(s) \quad (4.11.5)$$

$$\beta_m(s) \theta_m(s) = U(s) + N K_g \theta_L(s) \pm F_{Km}(s) \quad (4.11.6)$$

where

$$\beta_m(s) = s^2 J_m + s B_m + K_g \quad (4.11.7)$$

$$\alpha_L(s) = s^2 J_L + s B_L + N^2 K_g \quad (4.11.8)$$

Figure 4.11.1 presents the generalized flexible joint dynamics. Figures 4.11.1 and 4.11.1 show the significance of such parameters as motor and load damping, joint stiffness, gear reduction ratio, motor and load friction on the system dynamics. The above equations describe a case when the manipulator is unconstrained and the torque τ developed by the joint actuator equals

$$\tau = J_L \ddot{\theta}_L + B_L \dot{\theta}_L + N K_g (N \theta_L - \theta_m) \pm F_{FL} = T_L + W \quad (4.11.9)$$

When the manipulator tip encounters motion constraints, a collision occurs and the manipulator dynamics changes due to additional external environmental forces generated at this time. Therefore, an appropriate control law should be considered.

The link transient motion between the unconstrained and constrained condition is difficult to control, but approaches exist for this case.

Fig 4.1.1.1 The generalized flexible joint dynamics.

4.1.2 State Space System Representation

It is convenient to represent the considered SISO in a state space form. Introducing the following vector of states. $\mathbf{x} = [x_1, x_2]^T$, where: $x_1 = \theta_m$, $x_2 = \dot{\theta}_m$, the system dynamics can be written as follows:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) + \mathbf{E}v(t) \\ y(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t) \end{aligned} \quad (4.1.2.1)$$

For constrained conditions the adequate matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ are defined by:

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} 0 & 1 \\ \frac{K_g}{J_m} & \frac{B_m}{J_m} \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ \frac{1}{J_m} \end{bmatrix}; & \mathbf{C} &= [NK_g \quad 0]; \\ & & \mathbf{E} &= [0 \quad 1/J_m]^T; \\ & & \mathbf{D} &= [0]; \end{aligned} \quad (4.1.2.2)$$

4.1.3 Link and Force Sensor Flexibility

Equation 4.1.1.2 describing the system dynamic does not include the stiffness of the force sensor or manipulator link. Considering the deflection Δx_s and Δx_e due to the sensor and link deformation respectively along the x axis (in force sensor frame), equations 4.1.1.1 and 4.1.1.2 can be written in the following form

$$\begin{cases} \mathcal{F}(\ddot{\theta}_m, \dot{\theta}_m, \theta_m, \theta_L) = U \\ \mathcal{F}(\ddot{\theta}_L, \dot{\theta}_L, \theta_m, \theta_L) = 0 \end{cases} ; \quad F_x = \Delta x_e k_e \quad (4.1.3.1)$$

This is a set of second order coupled differential equations, where F_x is the force normal to the link, acting on its end, Δx_e is the global link/sensor deflection, U is the joint driving torque and k_e is the effective joint/sensor stiffness

Modelling the link and the sensor as a set of springs connected in series, we can write

$$k_e = \frac{1}{\frac{1}{k_s} + \frac{1}{k_L}} ; \quad k_L = \frac{3EI}{L^3} \quad (4.1.3.2)$$

where: $k_s = 3.153 \times 10^7$ (sensor stiffness [N/m]), $k_L = 9.747 \times 10^4$ (link stiffness),

$k_e = 9.717 \times 10^4$. Assuming a small joint deflection, the following relationship holds

$$F_x L = k_e \theta_L \quad \begin{array}{l} \text{We can substitute this for } \theta_L \text{ into equation 4.1.1.1 and} \\ \text{4.1.1.2 to express the load angle in terms of force } F_x \end{array}$$

Assuming $x_1 = \theta_m$, $x_2 = \dot{\theta}_m$, $x_3 = \theta_L$, $x_4 = \dot{\theta}_L$, the state-space system representation is now

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_g}{J_m} & -\frac{b_m}{J_m} & \frac{k_g N}{J_m} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{N k_g}{J_L} & 0 & -\frac{N^2 k_g}{J_L} - \frac{k_e L^2}{J_L} & -\frac{b_L}{J_L} \end{bmatrix} ; \quad \begin{array}{l} \mathbf{B} = [0 \quad \frac{1}{J_m} \quad 0 \quad 0]^T \\ \mathbf{C} = [0 \quad 0 \quad L k_e \quad 0] \\ \mathbf{E} = \begin{bmatrix} 0 & -\frac{1}{J_m} & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{J_L} \end{bmatrix}^T \end{array} \quad (4.1.3.3)$$

This representation will be used in the control law design in Chapter 6

4.2 System Response Simulation

In this part the model of the system dynamics developed in the previous Chapter is simulated for various values of joint parameters and reference inputs. The behaviour of uncontrolled system dynamics is investigated. The manipulator link is considered to be constrained by an ideal rigid surface. Thus the link constraints are expressed by:

$$\theta_L = 0; \quad \dot{\theta}_L = 0; \quad \ddot{\theta}_L = 0 \quad (4.2.1)$$

where θ_L is the link angular displacement. Assuming that the system output is the actual torque developed by the joint, the actual contact force can be found from equation 2.8.1.4. The following models are designed and simulated in Matlab (See Table 4.2.1):

- A) Time invariant system, motor friction neglected
- B) System with two values of the joint stiffness, motor friction neglected
- C) System with two values of the joint stiffness, motor friction included

Table 4.2.1 Flexible joint dynamic simulation subroutines.

MODEL	SUBROUTINE	TYPE OF INPUT
A	DS_1S	step
	DS_1P	impulse
	DS_F	abs[sin(10t)]
	DS_F2	time varying step
B	DS_VS	step, T=.02[s]
	DS_VS1	step, T=.00002[s]
	DS_V1	abs[sin(10t)]
	DS_V2	time varying step
C	DS_RS	step
	DS_R1	abs[sin(10t)]

4.2.1 Time Invariant Simplified Model

From equations 4.1.1.5 and 4.1.1.6, assuming for now $F_{KL} = 0$, $F_{KM} = 0$ we get

$$NK_g \theta_m(s) = T_L(s) \quad (4.2.1.1)$$

$$s^2 J_m \theta_m(s) + s B_m \theta_m(s) + K_g \theta_m(s) = U(s) \quad (4.2.1.2)$$

combining the above equations yields

$$\frac{T_L(s)}{U(s)} = \frac{NK_g}{s^2 J_m + s B_m + K_g} \quad (4.2.1.3)$$

or transforming the above equation, we get

$$\frac{T_L}{U} = \frac{NK_g/J_m}{s^2 + B_m/J_m + K_g/J_m} \quad (4.2.2.4)$$

Introducing the following definitions

$$2\xi\omega = \frac{B_m}{J_m} ; \quad \omega^2 = \frac{K_g}{J_m} \quad (4.2.2.5)$$

and substituting into equation 4.2.2.4, we obtain

$$\frac{T_L}{U} = \frac{NK_g/J_m}{s^2 + 2\xi\omega s + \omega^2} \quad (4.2.2.6)$$

The system natural frequency ω_n equals 16453 rad/sec (the joint stiffness 2599 Nm/rad) or 37583 rad/sec (the stiffness 13560 Nm/rad). Figures 4.2.1.1 to 4.2.1.6 present simulation results of the joint uncontrolled dynamics, for two numerical values of the joint stiffness $K_g = 2599$ Nm/rad and $K_g = 13560$ Nm/rad, and for various sampling times. The system responses for step, pulse and time varying inputs are simulated.

Flexible joint dynamics simulation (1) TIME INVARIANT SYSTEM

REDUCED ORDER MODEL, MOTOR FRICTION $F_{km} = 0$

step input response, $T = 0.000002$ [s]

Fig. 4.2.1.1 Flexible joint step input response, $k_g = 2599$ Nm/rad.

$T = 0.000002$ [s] (sampling time)

Flexible joint dynamics simulation (2) TIME INVARIANT SYSTEM
SIMPLIFIED MODEL, MOTOR FRICTION $F_{km} = 0$

Fig. 4.2.1.2 Flexible joint step input response, $kg = 13560 \text{ Nm/rad}$
 $T = 0.04 \text{ [s]}$ (sampling time)

Flexible joint dynamics simulation (3) TIME INVARIANT SYSTEM
 SIMPLIFIED MODEL, MOTOR FRICTION $F_{\text{vis}} = 0$

Fig. 4.2.1.3 Flexible joint impulse input response, $k_g = 13560$ Nm/rad.

$T = 0.02$ [s] (sampling time)

Flexible joint dynamics simulation (4) TIME INVARIANT SYSTEM

SIMPLIFIED MODEL. MOTOR FRICTION $F_{fr} = 0$

time varying input $U = \text{abs}[\sin(10kT)]$, $k = 1 \dots N$, N - number of sampling loops

Fig. 4.2.1.4 Flexible joint varying input response kg. 2599 Nm/rad
T = 0.02 (sampling time)

Flexible joint dynamics simulation (5) TIME INVARIANT SYSTEM

SIMPLIFIED MODEL MOTOR FRICTION $F_{fr} = 0$

time varying input $U = \text{abs}[\sin(10kT)]$, $k=1 \dots N$, N - number of sampling loops

Fig. 4.2.1.5 Flexible joint varying input response, $k_g = 13560 \text{ Nm/rad}$.

$T = 0.002 \text{ [s]}$ (sampling time)

Flexible joint dynamics simulation (6) TIME INVARIANT SYSTEM

SIMPLIFIED MODEL, MOTOR FRICTION $F_{km} = 0$

time varying step input $U_1 = 1(t)$, $U_2 = 1.2[1 - 1(t)]$, $U_3 = 1(t)$

Fig. 4.2.1.6 Flexible joint varying input response, $kg = 2599 \text{ Nm/rad}$

$T = 0.02 \text{ [s]}$ (sampling time)

4.2.2 Discrete Time System with Two Values of Joint Stiffness

The common representation of the joint stiffness is given as a spring with a fixed spring constant. In this work joint stiffness depends mainly on the harmonic drive elasticity. Two different values K_{e1} and K_{e2} of the harmonic drive stiffness are given by the manufacturer, which depend on the actual magnitude of the developed output torque (See Chapter 2.6). To include dynamic changes caused by two values of joint stiffness it is proposed to present the system dynamics with a toggling joint stiffness. The force feedback allows to obtain the value of the torque developed by the joint and to substitute the corresponding K_p . The general form of a system in the state-space form is given by the following set of equations

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{Ax}(t) + \mathbf{Bu}(t) \\ \mathbf{y}(t) &= \mathbf{Cx}(t) + \mathbf{Du}(t)\end{aligned}\tag{4.2.2.1}$$

Transforming the above expression to the discrete time representation given by discretized matrices $\mathbf{A}_d$, $\mathbf{B}_d$, $\mathbf{C}_d$, $\mathbf{D}_d$ we obtain

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}_d\mathbf{x}(k) + \mathbf{B}_d\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}_d\mathbf{x}(k) + \mathbf{D}_d\mathbf{u}(k)\end{aligned}\tag{4.2.2.2}$$

This approach assumes the forward evaluation of matrices $\mathbf{A}_d$ and $\mathbf{B}_d$ and the system transfer function updating during the run time, depending on the actual joint output torque $y(k)$. Because for the given configuration $\mathbf{D}=[0]$, we can write

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}_d\mathbf{x}(k) + \mathbf{B}_d\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}_d\mathbf{x}(k)\end{aligned}\tag{4.2.2.3}$$

In Fig. 4.2.2.1 the above concept of the system discretization is presented.

Fig 4.2.2.1 System with two values of the joint stiffness- flexible robot joint

The computer algorithm to update system parameters is shown in Fig 4.2.2.2

- 1 < enter initial conditions X_{01} >
- 2 < calculate the output torque $y(nk)$ >
 - if $y(nk) < \text{abs}(\text{liml})$ $K_p = K_{p1}$
 - else $K_p = K_{p2}$
- 3 < build matrices A_d, B_d, C_d >
- 4 < build the new transfer function >-----go to 2 -----

Fig 4.2.2.2 Algorithm to simulate the flexible joint dynamics Discrete time domain, liml 20% of the max output torque

In Figs 4.2.2.3 to 4.2.2.7 the system responses for various types of inputs are shown

Flexible joint dynamics simulation, (1)

STATE SPACE MODEL, MOTOR FRICTION $F_{km} = 0$;

input response, $T = .02$ (sampling time)

input $U = 2\text{Nm}$

Fig. 4 2 2 3 Flexible joint step input response. System with two values of joint stiffness. Simulation time 7 [s].

Flexible joint dynamics simulation, (2)

STATE SPACE MODEL, MOTOR FRICTION $F_{km} = 0$;

input response, $T = .02$ (sampling time)

input $U = 20\text{Nm}$

Fig. 4.2.2.4 Flexible joint step input response System with two values of joint stiffness Simulation time 7 [s].

Flexible joint dynamics simulation, (3)

STATE SPACE MODEL, MOTOR FRICTION $F_{km} = 0$;

input response, $T = .00002$ (sampling time)

input $U = 2\text{Nm}$

Fig. 4.2.2.5 Flexible joint step input response. System with two values of joint stiffness. Simulation time 7 [s].

Flexible joint dynamics simulation. (4)

STATE SPACE MODEL, MOTOR FRICTION $F_{km} = 0$,

input response, $T = .00002$ (sampling time)

input $U = 20\text{Nm}$

Fig 4 2 2 6 Flexible joint step input response System with two values of joint stiffness Simulation time 7 [s]

Flexible joint dynamics simulation, (5)

STATE SPACE MODEL, MOTOR FRICTION $F_{km} = 0$,

input response, $T = 0.2$ (sampling time)

input $U = \sin(t)$

Fig. 4.2.2.7 Flexible joint response. System with two values of joint stiffness

4.2.3 System Including Motor Friction

In the previous chapter the simplified system dynamics neglecting motor friction is developed. Below is designed and simulated a system with two values of the joint stiffness with the motor friction included in the model. Simulation software "DS- RS" can be found in Appendix II. The system is presented now in the following form

$$\begin{aligned} x(k+1) &= A_d x(k) + B_d \mu(k) + E_d F_{km} \\ y(k) &= C_d x(k) + D_d \mu(k) \end{aligned} \quad (4.2.3.1)$$

or after simple transformation of equation 4.2.3.1

$$x(k+1) = A_d + [B_d \ E_d] \begin{bmatrix} u \\ F_{km} \end{bmatrix} ; \quad E_d \begin{bmatrix} 0 \\ \pm \frac{1}{J_m} \end{bmatrix} \quad (4.2.3.2)$$

In this formulation F_{km} is assumed to have a constant value, where sign depends on the motor velocity direction. Function "TRAF-FM" written in Matlab evaluates system matrices and the function "FRICT_IN" calculates the actual friction value. The average F_{km} given for the S6M motor equals 0.064 Nm. The vector of initial conditions X_0 consist of the motor position θ_{init} and motor angular velocity $\dot{\theta}_{init}$, and $X_0 = [0 \ 0]$. Simulation results of the model developed above for various inputs are given in Figs. 4.2.3.1 and 4.2.3.3. For a time varying input $u = \text{abs}[\sin(t)]$ a high frequency oscillation about the desired input value is observed. The joint maximum output torques, the motor maximum velocity and the measured joint compliance are given in Table 4.2.3.1.

Table 4.2.3.1 Simulation results: System with motor friction

INPUT	MAX. OUTPUT	JOINT COMPLIANCE	MOTOR VELOCITY
2 Nm	4 Nm	$1.6 \cdot 10^{-3}$ rad	13 rad/sec
20 Nm	1300 Nm	$7 \cdot 10^{-3}$ rad	2000 rad/sec
$\text{abs}[\sin(t)]$	1.3 Nm	$5 \cdot 10^{-3}$ rad	1.8 rad/sec

Flexible joint dynamics simulation, (1)

STATE SPACE MODEL, MOTOR FRICTION $F_{\text{fric}} = 0.0064 \text{ Nm}$,

step input response, $T = 0.2$ (sampling time) ,

step input $U = 2 \text{ Nm}$

Measured data

- joint output torque
- contact force F
- motor angle θ_m
- motor velocity

Fig. 4.2.3.1 Flexible joint step input response. System with motor friction included. Model with two values of the joint stiffness.
Matlab simulation, program "DS_RS M"

Flexible joint dynamics simulation. (2)

STATE SPACE MODEL, MOTOR FRICTION $F_{km} = 0.0064 \text{ Nm}$.

step input response, $T = 0.2$ (sampling time) .

step input $U = 20 \text{ Nm}$

Measured data

- joint output torque
- contact force F
- motor angle θ_m
- motor velocity

Fig. 4.2.3.2 Flexible joint: step input response. System with motor friction included. Model with two values of the joint stiffness. Matlab simulation, program "DS_RSM"

Flexible joint dynamics simulation, (3)

STATE SPACE MODEL, MOTOR FRICTION $F_{\text{visc}} = 0.064 \text{ Nm}$,

time varying input, $T = 0.2$ (sampling time),

Measured data

- joint output torque
- contact force F
- motor angle θ_m
- motor velocity

Fig. 4.2.3.3 Flexible joint response. System with motor friction included. Model with two values of the joint stiffness. Matlab simulation, program "DS_R1.M"

Chapter 5

FORCE CONTROL LAW DESIGN

In this Chapter various algorithms designed for the flexible joint force control are presented. The performance of classical control methods based on the developed in Chapter 4 dynamical models is analyzed. A reference model for the MRAC adaptive controller is also designed.

5.1 Open Loop Force Control

We first consider an open loop force control strategy of a single input/single output system. The disturbance rejection problem at this stage will be neglected. The joint friction is not taken into account to simplify the system model. The objective is to design a compensator in such a way that the system output tracks a reference signal. In an open loop control the force sensor feedback information is not used. Therefore, the actual value of the exerted contact force is not known exactly and depends on the accuracy of the chosen model. The steady state error E_{ss} depends on system parameters, controller gains, friction and external disturbances. The steady state error compensation can be realised by an appropriate choice of feedforward gains. For the considered single degree of freedom system joint coupling effect does not occur, thus the system dynamics and control strategy can be simplified.

5.1.1 Simplified Model

Based on the reduced order model designed in Chapter 4.2.1 an open loop force control scheme will be now designed. A proportional controller is implemented and a set of controller gains providing a minimal steady state force error E_{ss} and system stability is chosen. This control scheme does not implement the force feedback information. Based on the manipulator kinematics and the required output (contact force), the desired controller output voltage and corresponding joint driving torque U are calculated. Figure 5.1.1 presents the block diagram of an open loop force control of a flexible joint.

Fig. 5.1.1 Block diagram of the open loop force control

Design of the open loop force control law:

Below the transfer function for an open loop force control system is developed.

Neglecting in this step for the sake of simplicity motor friction F_{bm} , the transfer function between the actual force measured by the JR3 sensor and a desired one equals

$$\frac{F}{F_d} = \frac{M_d}{M_d} K_m K_J H_c(s) H_a(s) H_j(s) \quad ; \quad M_d = J^T F_d \quad (5.1.11)$$

For the constrained link we introduce the following

$$\theta_L = 0; \quad \dot{\theta}_L = 0; \quad \ddot{\theta}_L = 0 \quad (5.1.1.2)$$

The equations describing the system dynamics are

$$NK_g(N\theta_L - \theta_m) = -M_a + F_{KL} \quad (5.1.1.3a)$$

$$s^2 J_m \theta_m + s B_m \theta_m - K_g(N\theta_L - \theta_m) + F_{km} = U \quad (5.1.1.3b)$$

and introducing assumed initial conditions we get

$$NK_g \theta_m = M_a \quad (5.1.1.4a)$$

$$s^2 J_m \theta_m + s B_m \theta_m + K_g = U \quad (5.1.1.4b)$$

After some transformations of equations 5.1.1.4a and 5.1.1.4b we obtain

$$\theta_m = \frac{M_a}{NK_g} \quad (5.1.1.5a)$$

$$(s^2 J_m + s B_m + K_g) \frac{M_a}{NK_g} = U \quad (5.1.1.5b)$$

From the above equations the system open loop transfer function equals

$$\frac{M_a}{U} = \frac{NK_g}{s^2 J_m + s B_m + K_g} \quad (5.1.1.6)$$

where the torque input U is given by

$$U = K_m K_e H_c(s) H_d(s) \quad (5.1.1.7)$$

Finally, the open loop system transfer function equals

$$\frac{M_a}{M_d} \frac{NK_g K_a K_m H_c(s) H_a(s)}{(s^2 J_m + s B_m + K_g)(T_a s + 1)} \quad (5.1.1.8)$$

Now we want to apply a proportional controller, thus the controller transfer function is just a proportional gain K_p . The system transfer function is given by:

$$\frac{M_a}{M_d} \frac{NK_g K_a K_p H_a(s)}{s^2 J_m + s B_m + K_g} ; \quad H_a(s) = \frac{1}{T_a s + 1} \quad (5.1.1.9)$$

$H_a(s)$ is the amplifier transfer function with the time constant T_a .

Computer programs "OLSIM_1" and "OLSIM_2" written in Matlab simulate the dynamics of the above developed system. Simulation results of the developed above system are shown in figures 5.1.1.2 to 5.1.1.5.

Some of simulation results are given in Table 5.1.1.1.

Table 5.1.1.1 Simulation results - Open Loop Force Control.

INPUT	GAIN K_f	JOINT STIFFNESS [Nm/rad]				
		2599		13560		8000
		$E_{ss}\%$	Fig	$E_{ss}\%$	Fig	$E_{ss}\%$
step	10	800	5.1.1.2	800	5.1.1.3	800
	117	0	5.1.1.4	0	-	0
	0588	50	-	-	-	-
sin(10t)	0588	60	5.1.1.5	60	-	-

Force Control of a Flexible Joint Manipulator, Simulation (1)
 TIME INVARIANT SYSTEM,

joint stiffness $K_g = 2599 \text{ Nm/rad}$
 sampling time $T = .02 \text{ [s]}$
 proportional gain $K_p = 1$

-STEP INPUT RESPONSE-

Fig. 5.1.1 2 Open loop control of the flexible robot joint with the PMI S6M4H DC motor, assumed joint stiffness $K_g = 2599$ Step input response

Force Control of a Flexible Joint Manipulator , Simulation (2)

TIME INVARIANT SYSTEM,

joint stiffness $K_g = 13560 \text{ Nm/rad}$
 sampling time $T = 0.02 \text{ [s]}$
 proportional gain $K_p = 1$

-STEP INPUT RESPONSE-

Fig 5.1.1.3 Open loop control of the flexible robot joint with the PMI S6M4H DC motor, assumed joint stiffness $K_g = 13560 \text{ Nm/rad}$. Step input response.

Force Control of a Flexible Joint Manipulator. Simulation (3)

TIME INVARIANT SYSTEM,

joint stiffness $K_g = 2599 \text{ Nm/rad}$
 sampling time $T = .02 \text{ [s]}$
 proportional gain $K_p = .117$

-STEP INPUT RESPONSE-

Fig. 5.1.1.4 Open loop force control of the flexible robot joint with the PMI S6M4H DC motor, assumed joint stiffness $K_g = 2599 \text{ Nm/rad}$ Step input response

Force Control of a Flexible Joint Manipulator, Simulation (4)

TIME INVARIANT SYSTEM,

joint stiffness $K_g = 13560 \text{ Nm/rad}$

sampling time $T = .02 \text{ [s]}$

proportional gain $K_p = .0588$

INPUT $U = \text{abs}[\sin\{10t\}]$

Fig 5.1.1.5 Open loop force control of the flexible robot joint with the PMI S6M4H DC motor, assumed joint stiffness $K_g = 13560 \text{ Nm/rad}$.

5.1.2 Control Scheme with Motor Friction

Including the motor friction component in the system dynamics the following system transfer function is obtained

$$F = F_d H_o(s) H_j(s) + F_{km}(s) H_j(s) \quad (5.1.2.1)$$

where $H_j(s)$ is the joint transmittance and equals

$$H_j(s) = \frac{NK_g}{s^2 J_m + sB_m + K_g} \quad (5.1.2.2)$$

and $H_o(s)$ is defined by

$$H_o(s) = K_m K_a H_c(s) H_a(s) \quad (5.1.2.3)$$

A major problem in the controller design with the friction included is the estimation of the actual friction value of the motor and joint housing bearings. At this point motor friction assumed is as a constant parameter given by the motor manufacturer. The direction of friction depends on the motor velocity sign (See Chapter 2.8)

Although it seems to be a proper approach to neglect the joint bearings friction (due to the constrained motion assumption), physically this parameter influences the system behaviour and is involved in the overall dynamics

The steady state force error equals

$$E_{ss} = \lim_{s \rightarrow 0} sE(s) = \lim_{s \rightarrow 0} s(F_d - F_a) \quad (5.1.2.4)$$

where F_d and F_a are the desired and actual force respectively. The steady state error depends strongly on the friction value F_{km}

Substituting adequate transfer functions into equation 5.2.1.4 yields

$$E_{ss} = sF_d \left[1 - \frac{NK_d H_c(s) K_m K_g}{s^2 J_m + sB_m + K_g} \right] \pm sF_{km} \frac{NK_g}{s^2 J_m + sB_m + K_g} \quad (5.1.2.5)$$

and for a step reference input $F_d(s) = F_d/s$ and a step disturbance $F_{km}(s) = F_{km}/s$, E_{ss} equals:

$$E_{ss} = F_d [1 - NH_c(s) K_d K_m] \pm F_{km} N \quad (5.1.2.6)$$

Neglecting for now the friction component F_{km} in equation 5.1.2.5, we can choose the controller proportional gain K_p to minimize E_{ss}

$$K_p = \frac{1}{NK_d K_m} = \frac{1}{160 \cdot 0.0311 \cdot 76} = .1179 \quad (5.1.2.7)$$

5.2 Closed Loop Control

Closing the control loop with the force feedback information from the JR3 sensor allows to obtain the actual contact force on the manipulator tip. The block diagram of the closed loop control is given in Figure 5.2.1

Fig. 5.2.1 Closed loop force control diagram

To the dynamic model of the flexible joint presented in Chapter 4 the classical PD control will be now applied. PD controllers are designed first neglecting the global joint friction. This concept allows to estimate the steady state error E_{ss} caused by friction.

In the next step it is assumed that values of such parameters as friction and joint flexibility are known. Various values of these parameters are considered.

Finally, assuming friction as a varying and highly nonlinear parameter, a control scheme implementing a state estimator is designed.

The system dynamics is described by (See Figure 5.2.1)

$$F = (U \pm F_{km}) H_f(s) \quad (5.2.4)$$

or

$$F = [F_d(s) - F] H_o(s) H_f(s) \pm F_{km} H_f(s) \quad (5.2.5)$$

after some transformations of equation 5.2.2 we get finally

$$F = F_d \frac{H_f(s) H_o(s)}{1 + H_f(s) H_o(s)} \pm F_{km} \frac{H_f(s)}{1 + H_f(s) H_o(s)} \quad (5.2.6)$$

The force error F_e is defined by

$$F_e = F_d - F = F_d \left[1 - \frac{H_f(s) H_o(s)}{1 + H_f(s) H_o(s)} \right] \pm F_{km} \frac{H_f(s)}{1 + H_f(s) H_o(s)} \quad (5.2.7)$$

and the steady state force error is defined

$$F_{e\infty} = \lim_{s \rightarrow 0} s F_e = \lim_{s \rightarrow 0} s F_d \left[1 - \frac{H_f(s) H_o(s)}{1 + H_f(s) H_o(s)} \right] \pm s F_{km} \frac{H_f(s)}{1 + H_f(s) H_o(s)} \quad (5.2.8)$$

Substituting for $H_f(s)$ and $H_o(s)$ obtained earlier values and after some transformation the transfer function is given by

$$F = F_d \frac{NK_d K_g K_a K_m (s+a)}{s^3 J_m T + s^2 (J_m + B_m T) + s(NK_g K_d K_a K_m + B_m + K_g T) + NK_g K_d K_a K_m a + K_g} \quad (5.2.9)$$

$$\pm F_{km} \frac{NK_g [s^3 J_m T + s^2 (J_m + B_m T) + s(B_m + K_g T) + K_g]}{s^3 J_m T + s^2 (J_m + B_m T) + s(B_m K_g T + NK_g K_d K_a K_m) + NK_g K_d K_a K_m a + K_g}$$

In the following Chapter a classical PD controller based on equation 5.2.9 is designed

5.2.1 Classical PD Compensator, Simplified Model

In this Chapter a classical PD controller for the given system is designed

The output from the system is the actual contact force F on the manipulator tip. The system input is the desired contact force F_d . From equation 5.2.9, neglecting its second part, we get the following open loop transfer function,

$$H_{ol}(s) = \frac{F}{F_d} = \frac{NK_d K_m K_g K_d (s + \alpha)}{s^2 J_m + s B_m + K_g} \quad ; \quad K_p = \alpha K_d \quad (5.2.1.1)$$

the system natural frequency ω_n and damping ratio ξ equal

$$\omega_n^2 = \frac{K_g}{J_m} \quad ; \quad \xi = \frac{B_m}{2} \sqrt{\frac{J_m}{K_g}} \quad (5.2.1.2)$$

From the final value theorem the steady state force error for the step input $F_d(s) = F_d/s$ equals

$$E_{ss} = \lim_{s \rightarrow 0} (F_d(s) - F(s)) = \lim_{s \rightarrow 0} F_d \left[1 - \frac{H_{ol}(s)}{1 + H_{ol}(s)} \right] \quad (5.2.1.3)$$

or

$$E_{ss} = F_d \frac{\omega_n^2}{\omega_n^2 + G K_d \alpha} \quad ; \quad G = NK_d K_g K_m \quad (5.2.1.4)$$

where $G = 22029$, $H_{ol}(s)$ is the system open loop transfer function expressed by equation 5.1.1.9. This system has a zero added at $s = -\alpha$. Poles of the system can be calculated for two different values of the joint stiffness, K_m and K_g .

(5.2.1.5)

$$s_{1,2} = -\omega_n \xi \pm \omega_n \sqrt{\xi^2 - 1}$$

From Table 5.2.1.1 it can be seen that the system is characterized by a very small

damping and two complex poles located close to the imaginary axis

Table 5.2.1.1 Open loop flexible joint system parameters

system parameters	$K_{d1} = 2599$ [Nm/rad]	$K_{d2} = 13560$ [Nm/rad]
p	$-5.25 \times 10^{-3} - 1.64 \times 10^{-3}i$	$-5.25 \times 10^{-3} - 3.72 \times 10^{-3}i$
p	$-5.25 \times 10^{-3} + 1.64 \times 10^{-3}i$	$-5.25 \times 10^{-3} + 3.72 \times 10^{-3}i$
ω_n	1.64×10^3	3.72×10^3
ξ	3.1407×10^{-3}	1.3964×10^{-3}

The root locus is shown in Fig. 5.2.1.1. To design controller gains K_p and K_d , program "PDGAIN" was written in Matlab. Function "RLOC" allows to plot the root locus for various design parameters α . See Appendix II for the program source code.

• PD GAIN DESIGN

The closed loop transfer function $H(s)$ equals

$$H_d(s) = \frac{\frac{K_d G}{J_m}(s + \alpha)}{s^2 + \frac{B_m + K_d G}{J_m}s + \frac{K_g + K_d G \alpha}{J_m}} \quad (5.2.1.6)$$

denominator of the above equation can be written as

$$2\xi\omega = \frac{B_m + K_d G}{J_m} ; \quad \omega^2 = \frac{K_g + K_d G \alpha}{J_m} \quad (5.2.1.7)$$

From equation 5.2.1.7 the following condition arises: $\omega > 16453.8$ rad/sec (2618.7 Hz). Assuming $K = 2599$, $\xi = 7$, response frequency $\omega = 20000$, the corresponding gains are $K = 0.727$, $K = 0.00012261$, $\alpha = 595$. For a critical damping $\xi = 1$ and $\omega = 3290.7$ rad/sec, the controller parameters are $K_p = 354$, $K_d = 0.0002868$, $\alpha = 12349$. The system response was obtained for various design parameters based on the program "PD_RESP" (Table 5.2.1.2).

2

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

Table 5.2.1.2 PD controller design parameters

α	K_p	K_d
8000	2.8	0.0035
5000	.5	0.001
<u>12340</u>	<u>.354</u>	<u>0.000287</u>
5000	6	0.0012
5000	1.0	0.002
6000	6	0.001

Figures 5.2.1.2 to 5.2.1.6 present the system responses for various K_d and α . Plots were obtained using program written in Matlab "PDVOLT.1". The chosen PD controller gains are: $K_p = .354$, $K_d = 0.0002868$, $\alpha = 12340$. This set of gains provides a fast non-oscillatory response without an overshoot. The steady state error $E_{ss} = 0.99$. The system is characterized by a high sensitivity, which causes significant changes in the output for a small gain variation. An important drawback of this design is a possibility of an excitation of high frequency system dynamics due to the joint flexibility. An appropriate choice of gains should be a compromise between the character of response and performance requirements.

• Control Voltage Estimation

The control voltage V is given by: $V = (K_p + K_d s)e$, or after forward discretization,

$$V(k) = K_p e(k) + \frac{e(k) - e(k-1)}{\Delta t} K_d \quad (5.2.1.8)$$

Substituting for $e(0) = F_{sp} / k = 1/N$, where N is the number of sampling loops, the control voltage/time characteristic can be plotted. The control voltage depends strongly on gain values. With inadequate chosen parameters the problem of controller signal saturation may occur (Figs. 5.2.1.5 and 5.2.1.6).

Fig 5.2.1.1 Root locus - PD control of a flexible joint
 a) joint stiffness $K_g = 2599 \text{ Nm/rad}$
 b) joint stiffness $K_g = 13560 \text{ Nm/rad}$
 $\alpha = K_p / K_d$

Fig. 5.2.1.2 PD Force Control- Step response for various gains,
 $K_s = 2599 \text{ Nm/rad}$
 a) transient response -simulation time $2 \times 10^{-4} \text{ [s]}$
 b) steady state response -simulation time 1 [s]

a)

$$\alpha=12340$$

$$K_d=2.868 \cdot 10^{-5}$$

$$\alpha=5000$$

$$K_d=5 \cdot 10^{-5}$$

b)

$$K_d=2.868 \cdot 10^{-5}$$

$$K_d=5 \cdot 10^{-5}$$

Fig. 5.2.1.3 PD Force Control- Step response for various gains,

$$K_p=2599 \text{ Nm/rad}$$

a) transient response -simulation time = $2 \cdot 10^{-4}$ [s]

b) steady state response -simulation time = 1 [s]

$$F_{\text{ref}} = 0.99 \text{ N}$$

a)

$$\alpha = 12340$$

$$K_d = 2.868 \times 10^{-7}$$

b)

$$K_d = 2.868 \times 10^{-7}$$

Fig. 5.2.1.4 PD Force Control- Step response for various gains,

$$K_g = 2599 \text{ Nm/rad}$$

a) transient response -simulation time 2×10^{-4} [s], $T = 4 \times 10^{-4}$ [s]

b) steady state response -simulation time 1 [s], $T = 4 \times 10^{-4}$ [s]

Fig. 5.2.1.5 PD control of a flexible joint, $K_g=2599$ Nm/rad,
 Measured are : the contact force and force error
 2 sampling times a) $T_1=4 \cdot 10^{-7}$ [s], b) $T_2=4 \cdot 10^{-3}$ [s]
 $K_p=354$, $K_d=0.0002868$, $\alpha=12340$

Fig. 5.2.1.6 PD control of a flexible joint, $K_g = 2599$ Nm/rad,
 Measured is the controller voltage [V]
 2 sampling times $T_1 = 4 \cdot 10^{-7}$ [s] and $T_2 = 4 \cdot 10^{-4}$ [s]
 $K_p = 354$, $K_d = 0.0002868$, $\alpha = 12340$

5.2.2 PD Control Scheme with Motor Friction Compensation

To include the motor friction in a control system design, the state space approach will be now considered. Velocity feedback is used to obtain the actual motor velocity. Assuming the following states

$$X_1 = \theta_m; \quad X_2 = \dot{\theta}_m$$

The set of equations

$$NK_g \theta_m = M \quad (5.2.2.1)$$

$$J_m \ddot{\theta}_m + B_m \dot{\theta}_m + K_g \theta_m = U \pm F_{km}$$

where M is the joint output torque, will have the following state space form:

$$\begin{aligned} \dot{X}_1 &= X_2 \\ \dot{X}_2 &= -\frac{B_m}{J_m} X_2 - \frac{K_g}{J_m} X_1 + \frac{1}{J_m} U \pm \frac{1}{J_m} F_{km} \end{aligned} \quad (5.2.2.2)$$

Now we can write the system in the state space form:

$$\begin{aligned} \dot{\mathbf{X}} &= \mathbf{A}\mathbf{X} + \mathbf{B}U \pm \mathbf{E}F_{km} \\ Y &= \mathbf{C}\mathbf{X} \end{aligned} \quad (5.2.2.3)$$

and matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\mathbf{E}$ are defined as follows:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\frac{K_g}{J_m} & -\frac{B_m}{J_m} \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ \frac{1}{J_m} \end{bmatrix} \quad (5.2.2.4)$$

$$\mathbf{C} = [NK_g \quad 0]; \quad \mathbf{D} = [0]; \quad \mathbf{E} = \begin{bmatrix} 0 \\ \frac{1}{J_m} \end{bmatrix}$$

In equation 5.2.2.3 the system input U is defined as

$$U = K_a K_m V; \quad V = (K_p + sK_d) F_e; \quad F_e = F_d - F$$
(5.2.2.5)

For the considered control the following is valid

$$F_e = F_d - CX; \quad \dot{F}_e = -C\dot{X}$$
(5.2.2.6)

Combining above equations we get

$$\dot{X} = AX + BK_m K_a K_p F_e + BK_m K_a K_d \dot{F}_e + EF_{km}$$
(5.2.2.7)

or:

$$\dot{X} = AX + BK_m K_a K_p F_d - BK_m K_a K_p CX - BK_m K_a K_d C\dot{X} + EF_{km}$$
(5.2.2.8)

Transforming equation 5.2.2.8, we obtain

$$(I + BCK_m K_a K_d) \dot{X} = (A - BCK_m K_a K_p) X + BK_m K_a K_p F_d + EF_{km}$$
(5.2.2.9)

The system can be described now as follows

$$\dot{X} = A_c X + B_c F_d + EF_{km}$$
(5.2.2.10)

$$Y = CX + DU$$

$$A_c = (I + BCK_m K_a K_d)^{-1} (A - BCK_m K_a K_p)$$
(5.2.2.11)

$$B_c = (I + BCK_m K_a K_d)^{-1} BK_m K_a K_p F_d$$
(5.2.2.12)

Finally, substituting for $D = [0]$, we get

$$\dot{X} = A_c X + [B_c E_c] \begin{pmatrix} F_d \\ F_{km} \end{pmatrix}$$
(5.2.2.13)

$$Y = CX$$

Substituting matrices A , B , C into equations 5.2.2.11 and 5.2.2.12 and discretizing the obtained system using ZOH technique, the system simulation can be performed. For this purposes the program "PDFRSIM" was written in Matlab. The system discretization is done by the Matlab function "c2d m". The program outputs are: the actual state vector $\mathbf{X}$ and system output Y for each sampling time T . The velocity sign is evaluated and updated in equation 5.2.2.13 at each sampling period.

The simulation algorithm consist of the following steps:

- 1) enter the initial state vector $\mathbf{X}_{INI}=[X_1 \ X_2]$
- 2) for $X_1 = 0$, $F_{km}=0$, the initial input vector $\mathbf{U}(0)$ is given by: $\mathbf{U}(0)= [F_d(0) \ F_{km}(0)]$
- 3) calculate $[Y, \mathbf{X}]$
- 4) evaluate the actual motor velocity sign
- 5) calculate the actual input vector $\mathbf{U}(k) = [F_d(k) \ F_{km}(k)]$

Figure 5.2.2.1 presents simulation results for a PD gain set found in Chapter 5.2.1. The system is characterized by a fast, non-oscillatory response, steady state error is eliminated

PD FORCE CONTROL SIMULATION OF A FLEXIBLE ROBOT JOINT

■ State space model

System parameters $K_p = 354$, $K_d = 0.0002868$, $\alpha = 12340$

joint stiffness $K_x = 2599$ [Nm/rad]

Fig. 5.2.2.1 PD control of a flexible joint The motor friction compensation
 A- contact force F [N], B- motor position [rad]
 C- motor velocity [rad/sec] D- motor friction [Nm]

5.2.3 PID Controller Performance

To compensate the steady state error due to the system nonlinearities a PID control action is now considered. The controller transfer function $H_c(s)$ equals:

$$H_c(s) = K_d(s + \alpha + \frac{\beta}{s}) \quad ; \quad K_p = K_d \alpha \quad ; \quad K_i = K_d \beta \quad (5.2.3.1)$$

The open loop system transfer function $H_o(s)$ is given by (G is defined in Chapter 5.2.1):

$$H_o(s) = \frac{F}{F_d} = \frac{G K_d}{J_m} \frac{(s + \alpha + \frac{\beta}{s})}{(s^2 + 2\xi \omega_n s + \omega_n^2)} \quad (5.2.3.2)$$

• PID GAIN DESIGN

System closed loop transfer function is given by

$$H_d = \frac{G_1 K_d (s^2 + \alpha s + \beta)}{s^3 + (2\xi \omega_n + G_1 K_d) s^2 + (\omega_n^2 + G_1 K_d \alpha) s + G_1 K_d \beta} \quad ; \quad G_1 = \frac{G}{J_m} \quad (5.2.3.3)$$

The closed loop steady state error for the step input equals (after some transformations of equation 5.2.3.3)

$$E_{ss} = \lim_{s \rightarrow 0} sE = \lim_{s \rightarrow 0} F_d \left[\frac{s(s^2 + 2\xi \omega_n s + \omega_n^2)}{s(s^2 + 2\xi \omega_n s + \omega_n^2 + G K_d (s^2 + \alpha s + \beta))} \right] = 0 \quad ; \quad (5.2.3.4)$$

The source code for the program "PIDGAIN" designing controller gains and calculating the system response for a given set of controller gains can be found in Appendix 2. The system is characterized by a high sensitivity and fast response. The system stability is proved using Routh-Hurwitz stability criteria and root locus method. In Table 5.2.3.1 the chosen set of controller gains is given. Simulation results are shown in Fig. 5.2.3.1.

PID CONTROL SIMULATION OF A FLEXIBLE JOINT

Controller parameters

$$K = 43, K_i = 3.48 \times 10^{-4}, K_d = 34.86$$

$$\alpha = 12340, \beta = 10$$

Fig. 5.2.3.1 PID control of the flexible joint Step input response

TOP Transient response, $T=10^{-3}$ [s]

BOTTOM Steady state, $E_{ss} = 0$, $T=5 \times 10^{-3}$ [s]

Chapter 6

ADVANCED FORCE CONTROL LAWS

In this Chapter advanced algorithms to control the contact force on the tip of a flexible joint manipulator are designed. The following control techniques are designed and simulated

- Optimal Force Control
- Model Reference Adaptive Control - MRAC
- Force Control Scheme with Friction Estimator

The first two of the above mentioned methods are adaptations of known control algorithms to the force control scheme

Optimal Force Control uses the state feedback approach. State feedback gains are optimized using *performance index optimization method*. In addition, the control voltage saturation problem is also analyzed. Two schemes of the Model Reference Adaptive Control are considered. While the first MRAC scheme uses the PID controller designed in Chapter 5 as a reference model, the second method has a first lag filter as a reference. The designed algorithm drives the force error between the reference model and plant to zero. Force Control Scheme with Friction Estimator is an approach based on the flexible joint dynamics with two values of joint stiffness, depending on the actual output torque. This method uses a state estimator to evaluate unknown system parameters (e.g. the joint Coulomb friction) and is a result of a search for an appropriate solution for the given force control problem of a flexible joint. In all mentioned methods, the state space approach in discrete time is used.

6.1 Optimal Force Control of a Flexible Joint

In this paragraph an optimal force control approach is presented. The vector $\mathbf{K}$ of optimal controller gains is chosen based on the quadratic performance index optimization. Poles of the system are chosen to minimize the error and provide system stability and desired performance. A comparison is made with previously developed methods. The diagram representing this method is given in Fig. 6.1.1.

Fig. 6.1.1 Optimal control scheme State feedback approach.

Optimal control problem For the system described by equation 6.1.1 find a set of

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} ; \mathbf{y} = \mathbf{Cx} + \mathbf{Du} \quad (6.1.1)$$

optimal state feedback gains to achieve a desired system performance.

The method of performance index minimization is presented below.

6.1.1 Performance Index Optimization

Problem solution

Determine vector $\mathbf{K}$ of optimal feedback control gains to minimize the performance index J given by

$$J = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt \quad (6.1.2)$$

and Q is a positive-definite (or positive semidefinite) real symmetric matrix and R is a positive-definite, real, symmetric matrix. The control given by equation below is an optimal one

$$u(t) = -Kx(t) \quad (6.1.3)$$

Substituting equation 6.1.3 into 6.1.1 and 6.1.2 we get a set of equations

$$\dot{x} = (A - BK)x; \quad J = \int_0^{\infty} (x^T Q x + x^T K^T R K x) dt \quad (6.1.4)$$

The following is valid

$$x^T (Q + K^T R K) x = -\frac{d}{dt} (x^T P x) \quad (6.1.5)$$

$$x^T (Q + K^T R K) x = -\dot{x}^T P x - x^T P \dot{x} = -x^T [(A - BK)^T P + P(A - BK)] x \quad (6.1.6)$$

Equation 6.1.6 is true for any values of x if the below is valid

$$(A - BK)^T P + P(A - BK) + (Q + K^T R K) = 0 \quad (6.1.7)$$

and the performance index J is given by

$$J = x^T(0) P x(0); \quad x^T(\infty) = 0 \quad (6.1.8)$$

From equation 6.1.7, assuming that $R = T^T T$ (T is a nonsingular matrix), we obtain

$$(A^T + K^T B^T) P + P(A - BK) + Q + K^T T^T T K = 0 \quad (6.1.9)$$

After some transformations ([12]) we get the expression for K

$$K = T^{-1} (T^T)^{-1} B^T P + R^{-1} B^T P \quad (6.1.10)$$

and matrix P can be obtained from the following reduced-matrix equation:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (6.1.11)$$

6.1.2 Design Procedure

The design procedure is as follows:

- establish the state space model of the system
- define symmetric positive definite "weighting matrix" Q
- define symmetric semi-positive definite "penalty matrix" R
- solve Ricatti equation 6.1.11 for the matrix P
- substitute P into equation 6.1.10 to get the desired gain vector K

The following states will be chosen for the designed controller:

$$\begin{aligned} x_1 &= NK_g \theta_m ; \quad x_2 = \dot{\theta}_m \\ x_3 &= \int_{t-\Delta t}^t \theta_m dt \Rightarrow \dot{x}_3 = \theta_m \end{aligned} \quad (6.1.2.1)$$

Thus A , B , C , D matrices for the control law $u = -Kx$ are defined as follows:

$$\begin{aligned} A &= \begin{bmatrix} 0 & NK_g & 0 \\ -\frac{1}{J_m N} & -\frac{B_m}{J_m} & 0 \\ 1 & 0 & 0 \end{bmatrix}; \quad B = \begin{bmatrix} 0 \\ \frac{K_a K_m}{J_m} \\ 0 \end{bmatrix} \\ C &= [1 \quad 0 \quad 0]; \quad D = [0] \end{aligned} \quad (6.1.2.2)$$

Q and R matrices for the controller have been chosen as (R is 1×1):

$$Q = \begin{bmatrix} \zeta & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \psi \end{bmatrix}; \quad R = \alpha \quad (6.1.2.3)$$

where: ζ , ψ , α are chosen depending on the system performance and stability criteria. Various values of ζ , ψ , α are considered. Motor velocity and position are measurable states. The magnitude of motor friction F_{im} is assumed to be known and its sign depends on the motor velocity direction.

6.1.3 Optimal Control Simulation

The simulation program "OPTCONT_1" has been written in Matlab. Simulation results for various design parameters are shown in Figs. 6.1.2 and 6.1.3. See Table 6.1.1 for some of considered design parameters.

Table 6.1.1. Optimal control parameters

Controller parameters			Controller gains	Closed Loop poles
α	ζ	ψ		
20	100	10	2.121	$(-4.93 \pm 5.21) \cdot 10^4$ $(-4.93 - 5.21) \cdot 10^4$ $-7.594 \cdot 10^7$
			17.88	
			.707	
20	1000	10	6.95	$(-8.93 \pm 8.21) \cdot 10^4$ $(-8.93 - 8.21) \cdot 10^4$ $2.404 \cdot 10^7$
			32.37	
			.707	
20	10	1	5.99	$(-2.66 \pm 3.01) \cdot 10^4$ $(-2.66 - 3.01) \cdot 10^4$ $7.5 \cdot 10^7$
			9.50	
			.022	

Control limitation - control voltage saturation

A practical limitation in control is caused by a limited driver voltage. From Figs. 6.1.3 to 6.1.4 it can be seen, that for an optimal control process the required voltage is much

higher than the maximum voltage supplied to the VXA 48-8-16 amplifier ($V_{\max} = 2V$). Thus the designer should adjust optimal gains to eliminate this problem. This can be done by changing the adequate parameters ζ , ψ of "weighting" and "penalty" matrices, Q and R , as shown in Fig 6.14. Additional design difficulties arise, because the system behaviour depends on a vector of initial states, desired output and system parameters. All these factors influence the controller voltage

Fig 6.1.2 Optimal control of a flexible robot joint - Simulation (1)
 $\alpha = 20$, $\zeta = 100$, $\psi = 10$ Simulation time $T_s = 10^{-4}$ s,
 Sampling time $T = 10^{-7}$ s Desired contact force = 5[N].

OPTIMAL CONTROL OF A FLEXIBLE JOINT - Simulation(2)

Fig. 6.1.3 Optimal control of a flexible robot joint - Simulation (2)

design parameters $\alpha = 20$, $\zeta = 1000$, $\psi = 10$ Desired force = 6 [N]

OPTIMAL CONTROL OF A FLEXIBLE JOINT - Simulation (3)

Fig 6 1 4 Optimal control of a flexible robot joint - Simulation (3)

design parameters: $\alpha=20$, $\zeta=10$, $\psi=.1$. Desired force = 6 [N].

6.2 Model Reference Adaptive Control - MRAC

To improve the performance of a flexible joint system controlled by a classical method, an adaptive controller based on the Model Reference (MRAC) approach will be designed. Such a controller should compensate for errors due to nonlinear system parameters, provide the system stability and robustness for external disturbances. State space design is considered. The MRAC control diagram is shown in Figure 6.2.1

Fig 6.2.1 Diagram of the MRAC control of a flexible joint

$H_m(s)$ - model TF, $H_p(s)$ - plant TF

$H_{mp}(s)$ - model controller TF, $H_{pc}(s)$ -plant controller TF,

F_m - model output force, F_p - plant output force, e - force error

ϵ - forward adaptation gain

The controller design steps

- establish a *desired model* of the system
- chose an appropriate *adaptation scheme*
- establish the *error* between the desired output from the model and the plant at each sampling period
- apply the chosen adaptation scheme to eliminate the error
- check the *system stability* for assumed control parameters

Following is the implementation of the MRAC control scheme:

- a) as the *reference model* is chosen the PID control scheme presented in Chapter 5.2.3
- b) as the *adaptation scheme* is used the MIT adaptation algorithm, based on a concept of driving the generated output error $e^2(\epsilon)$ to zero

The feedforward gain ϵ is updated at the run time regarding to the rule below:

$$\frac{\delta \epsilon}{\delta t} = -\lambda \frac{\delta(e^2(\epsilon))}{\delta \epsilon} = -\lambda e(\epsilon) \frac{\delta e(\epsilon)}{\delta \epsilon} \quad (6.2.1)$$

In above equation λ is a parameter chosen by the designer. The relationship between the system output and feedforward gain is given by the system sensitivity (equation 6.2.2).

$$\delta e(\epsilon) = \frac{\delta F}{\delta(\epsilon)} \quad (6.2.2)$$

The MIT rule is given by the following:

$$\dot{\epsilon} = -\lambda e \frac{\delta F}{\delta \epsilon} \quad (6.2.3)$$

If $H_{pc}(s)$ is the closed loop plant transfer function, than for this control scheme we get:

$$\frac{F}{F_d} = H_{pc}(s) \epsilon \quad ; \quad H_{pc}(s) = \frac{H_{pc}(s)H_p(s)}{1 + H_{pc}(s)H_p(s)} \quad (6.2.4)$$

transforming the above equations yields

$$F = e F_d H_{pcc}(s) \quad ; \quad \frac{\delta F}{\delta e} = F_d H_{pcc}(s) \quad (6.2.5)$$

$$\frac{\delta F}{\delta e} = \frac{F_d F}{F_d e} \quad \rightarrow \quad \frac{\delta F}{\delta e} = \frac{F}{e} \quad (6.2.6)$$

From equations 6.2.3 and 6.2.5 we obtain

$$\dot{e} = -\lambda e \frac{F}{e} \quad \Rightarrow \quad \dot{e} e = -\lambda e F \quad \rightarrow \quad \frac{1}{2} \frac{\delta e^2}{\delta t} = -\lambda e F \quad (6.2.7)$$

and finally the gain adaptation law can be written as below, where e_0 is an initial value of the feedforward gain and has a significant influence on the system behaviour

$$e = \sqrt{e_0^2 - \int \lambda e(F) dt} \quad (6.2.8)$$

Discretizing equation 6.2.8 with the sampling time T for a digital control process, yields

$$e(k+1) = \sqrt{e^2(k) - \lambda T F(k) e(k)} \quad (6.2.9)$$

As it can be seen from equation 6.2.9 the value of the feedforward gain depends on the sampling time, the actual system output, error and design parameter λ .

6.2.1 MRAC Controller Design

The reference model is given by equation 5.2.3.3. This model consist of the flexible joint dynamics with a PID controller added. For the plant a constant gain output feedback approach is used, given by

$$\begin{aligned} u &= -Ky \quad ; \quad u = -Kx \\ \dot{x} &= (A - BKC)x \end{aligned} \quad (6.2.11)$$

The chosen PID controller gains are $K_p=43$, $K_d=3.47 \cdot 10^{-4}$, $K_i=34.86$

The input to the plant is given by

$$u = F_d e - (K_p + K_d s)y = F_d e - (K_p Cx + K_d C\dot{x}) \quad (6.2.1.2)$$

substituting 6.2.1.2 into 6.2.2.1, yields the plant dynamics

$$\begin{aligned} \dot{x} &= M^{-1}(A - BK_p C)x + M^{-1}[B - E][F_d e - F_k]' \\ M &= (I + BK_d C) \end{aligned} \quad (6.2.1.3)$$

6.2.2 MRAC Control Simulation

The simulation is performed for the following conditions:

⇒ plant response simulation with- and without the adaptation mechanism

⇒ various values of the joint friction F_k are considered

$F_{k1} = 0.064$ Nm (nominal), $F_{k2} = 0.15$ Nm (found experimentally),

$F_{k3} = 2$ Nm (large disturbance)

⇒ the PID controller is used as the reference model

In each sampling time the output of both the plant and reference model is converted to the discrete time form. Various parameters λ and ϵ were considered during the simulation. Simulation results are shown in Figs. 6.2.2.1 to 6.2.2.4. The following are simulation conclusions:

- The proper choice of an initial value of the feedforward gain ϵ is not trivial.

It determines the system response. The chosen $\epsilon = 1$ provides the best system response (Figs. 6.2.2.2 and 6.2.2.3).

- Decreasing λ causes increasing of the adaptation time.

- A large λ causes system instability (i.e., for $\lambda=50$ and $\epsilon=1 \Rightarrow E_{ss} \rightarrow \infty$).

- For disturbances bounded in the range $0 \div 0.15$ Nm/rad designed controller works well (See Fig. 6.2.2.3).

- Large nonlinearities (i.e., 2 Nm/rad seen from the motor side) cause a minimal overshoot, the steady state force error E_{ss} is eliminated (Fig. 6.2.2.4).

10⁻⁴

simulation (1)

simulation (2)

Fig.6.2.2.1 MRAC adaptive control - simulation results for various parameters,
SS - steady state, TR - transient response
LAMBDA = λ , EPS = ϵ ,
Fk -joint friction [Nm/rad]

simulation (3)

simulation (4)

Fig.6.2.2.2 MRAC adaptive control - simulation results for various parameters,
 SS - steady state, TR - transient response
 LAMBDA λ , EPS = ϵ , Kdp - plant controller gain,
 a1 - plant controller parameter, Fk -joint friction [Nm/rad]

simulation (5)

simulation (6)

Fig.6.2.2.3 MRAC adaptive control - simulation results for various parameters,
 SS - steady state, TR - transient response
 LAMBDA = λ , EPS = ϵ , Kdp - plant controller gain,
 a1 - plant controller parameter, Fk - joint friction [Nm/rad]

simulation (7)

Fig.6.2.2.4 MRAC adaptive control - simulation results for various parameters,
 SS - steady state, TR - transient response
 LAMBDA = λ , EPS = ϵ , Kdp - plant controller gain,
 a1 - plant controller parameter, Fk - joint friction [Nm/rad]

6.2.3 MRAC - Plant without Controller

As the next step in MRAC control system design is the system simulation under the following conditions:

- the reference model is a first order lag filter
- two values of the joint stiffness K_g are considered
- a proportional gain K_p is added to the plant transfer function
- the control voltage is calculated and monitored

The "PDMRAC_1" simulation program was written in Matlab. See Table 6.2.3.1 for simulation parameters.

Table 6.2.3.1 MRAC control simulation parameters Unit step input response

K_p	ϵ	λ
1	1	1
1	.5	10
1	5	5
.06	01	10
1	00004	001
1	04	1

Simulation results are presented in Figs. 6.2.3.1 to 6.2.3.3. Simulation conclusions are:

- it is difficult to design an appropriate set of control parameters K_p , λ , and ϵ , satisfying the required response
- for a step input, parameters λ and ϵ , can have a wide range of values and the system response does not change significantly
- for larger values of λ and ϵ , (ϵ a 5 and 1) system is unstable (Fig. 6.2.3.1)
- the system response is characterized by a high frequency oscillation

Summarizing, the analyzed control scheme is not recommended for the considered system. The lack of an additional damping effect from the controller causes system oscillations. The design procedure of control parameters is not trivial.

Fig. 6.2.3.1 MRAC control simulation. Model- first order filter.
Plant without a controller.

Fig. 6.2.3.2 MRAC control simulation. Model- first order filter.
 Plant without a controller.

Fig. 6.2.3.3 MRAC control simulation. Model- first order filter.
Plant without a controller.

6.3 Force Control Scheme with Friction Estimator

The control scheme developed in this Chapter is based on the following assumptions:

- position and velocity of both motor and joint are measurable states
- motor friction has a constant magnitude, its sign depends on the motor velocity
- joint stiffness has two values depending on the joint output torque
- joint initial friction value F_{kj}^0 is assumed by the designer
- sensor and link flexibility are included in the system dynamics

Assuming the nonlinear joint friction F_{kj} as an unmeasurable system state, a state estimator is designed to evaluate its value at any given period of time. Denoting the estimated value of friction by $\tilde{F}_{kj}$, the friction estimation law can be written in the form

$$\dot{\tilde{F}}_{kj} = 0 \cdot x + 0 \cdot U + k(F_{kj}^0 \pm \tilde{F}_{kj}) \quad (6.3.1)$$

where k is observer gain. The control law compensating for friction is in the form

$$U = -K(F_d - y) \pm \theta \tilde{F}_{kj} \quad (6.3.2)$$

F_d is the desired contact force. Assuming a PD type control, where $K = [K_p, K_d]$, we get

$$\dot{x} = M^{-1}(A + BK_p C)x \pm M^{-1}B\theta \tilde{F}_{kj} - M^{-1}BK_p F_d \pm M^{-1}E_1 F_{km} \pm M^{-1}E_2 F_{kj}^0 \quad (6.3.3)$$

Rewriting equation 6.3.3 in matrix form, we get

$$\dot{x} = Ax + [B \quad E_1] \begin{bmatrix} K_p F_d - K_p Cx - K_d C\dot{x} \pm \theta \tilde{F}_{kj} \\ \pm F_{km} \end{bmatrix} \pm E_2 F_{kj}^0 \quad (6.3.4)$$

The final matrix form of the system dynamics is given by the below:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\tilde{F}}_{kj} \end{bmatrix} = \begin{bmatrix} \mathbf{M}^{-1}(\mathbf{A} - \mathbf{B}K_p\mathbf{C}) & -\mathbf{M}^{-1}\mathbf{B}\mathbf{0} \\ \mathbf{0} & -k \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \tilde{F}_{kj} \end{bmatrix} + \begin{bmatrix} \mathbf{M}^{-1}\mathbf{B}K_p & \mathbf{M}^{-1}\mathbf{E}_1 & \mathbf{M}^{-1}\mathbf{E}_2 \\ 0 & 0 & k \end{bmatrix} \begin{bmatrix} F_d \\ \pm F_{km} \\ \pm F_{kj}^0 \end{bmatrix} \quad (6.3.5)$$

where: $\mathbf{M} = (\mathbf{I} - \mathbf{B}K_p\mathbf{C})$, $\mathbf{E}_1 = [0 \ 1/J_m \ 0 \ 0]^T$, $\mathbf{E}_2 = [0 \ 0 \ 0 \ 1/J_l]^T$.

6.3.1 State Space Feedback Design

The controller design problem involves an appropriate choice of the coefficient $\mathfrak{g}$ in equation 6.3.5, assuring the system stability. The value of the friction observer pole k is chosen is in the left hand part of the zero-pole plane. The assumed value of the motor friction $F_{km} = .0064$ N. The initial value of the joint friction $F_{kj}^0 = .0086$ N.

The simulation program "MODFC_3" consist of the following steps:

- establishing a vector of initial conditions $\mathbf{X}_0$ for the system

$$\mathbf{X}_0 = [\theta_m \ \dot{\theta}_m \ \theta_L \ \dot{\theta}_L \ F_{kj}^0]$$

- motor velocity measurement
- joint friction estimation using observer
- controller feedback gain design

Considered is a discrete time system with two values of the joint stiffness. To calculate the parameter $\mathfrak{g}$ and vector $\mathbf{K}$ of feedback gains the *pole placement approach* is used. *Pole placement approach.*

a) built the system open loop characteristic equation:

$$\det(s\mathbf{I} - \mathbf{A}) = s^n + \alpha_{n-1}s^{n-1} + \dots + \alpha_1s + \alpha_0 = 0 \quad (6.3.1.1)$$

b) the desired closed loop system characteristic equation is given by:

$$\det(s\mathbf{I} - \mathbf{A}_d) = s^n + \beta_{n-1}s^{n-1} + \dots + \beta_1s + \beta_0 = 0 \quad (6.3.1.2)$$

$\mathbf{A}_d$ is the matrix of desired closed loop poles.

c) the controllability matrix is defined as:

$$\mathbf{C} = [\mathbf{B} \quad \mathbf{AB} \quad \dots \quad \mathbf{A}^{n-1}\mathbf{B}] \quad (6.3.1.3)$$

d) calculate matrix $\mathbf{P}$ given by:

$$\mathbf{P}_{\text{des}} = (\mathbf{C}^T)^{-1}_{n \times n} \begin{Bmatrix} 0 \\ \vdots \\ 1 \end{Bmatrix} \quad (6.3.1.4)$$

e) find vector $\mathbf{K}$ of controller gains.

$$\mathbf{K}^T = [\mathbf{P} \quad \mathbf{A}^T \mathbf{P} \quad \dots \quad (\mathbf{A}^T)^{n-1} \mathbf{P}] \begin{Bmatrix} \beta_0 & \alpha_0 \\ \vdots \\ \beta_{n-1} & \alpha_{n-1} \end{Bmatrix} \quad (6.3.1.5)$$

6.3.2 Simulation Examples

The following are the input and output data in the program "MODFC_3" simulating the response of the system given by equation 6.3.5

input:

- vector of initial states $\mathbf{X}_0$
- vector of desired closed loop poles $\mathbf{P_DES}$
- friction observer gain k
- ϑ design parameter

output:

- vector of observed friction values
- state vector $\mathbf{X}$
- system output $\mathbf{Y}$

The system simulation was performed for various design parameters. The system output depends on ϑ , k , F_{kf}'' and the vector of desired closed loop system poles. In Fig. 6.3.2.1 some of the system responses for a step input are shown. This system is characterized by a high sensitivity to design parameter changes and oscillatory response. The steady state error E_{ss} was not eliminated for the considered parameters. This control scheme would be recommended for the given system, if an algorithm is established determining a proper choice of design parameters. The source code for the program "MODFC_3" can be found in Appendix II.

Fig 6.3.2.1 Force control scheme with the joint friction estimator. Simulation results for various parameters.

Chapter 7

EXPERIMENTAL RESULTS

The previously developed force control laws are implemented on the SLIM-1 test manipulator, described in Chapter 3. Series of tests investigating the sources of manipulator link "chattering" are performed. Attempts are made to develop the means of the joint flexibility estimating in the real time. This concept would allow to improve the performance of a force controller. The following force control laws are implemented: open loop, PD, PID and adaptive control. Various controller gains and design parameters are considered. The execution time is kept at the minimum level. To control the high frequency oscillation of the system it would be necessary to use a controller with a higher bandwidth than this of the applied one. An analog-to-digital converter with the resolution higher than 12-bits would improve significantly the readability of the feedback data. The small amplitude noise visible in Figs. 7.3, 7.4, 7.5, 7.8, 7.9, 7.10, is due to the insufficient resolution of the analog-to-digital converter. To perform experiments, it was necessary to design several real time communication programs and subroutines. The documentation of the real time control software can be found in Appendix III. The performed tests and obtained results are described below.

■ Joint Flexibility Estimation

Joint flexibility causes an excitation of a high frequency vibration of the manipulator tip. Modelling of the joint flexibility is difficult because it includes the elasticity of all joint components. Joint flexibility is a complex phenomenon and for the considered joint can not be simply presented as a torsional spring with a linear characteristic. The primary

purpose of the tests presented below was an attempt to establish a relationship between the joint output torque and the global joint deflection. Table 7.1 presents sample results of two tests. The joint compliance $\delta\psi$ is measured by two incremental encoders (Chap. 2.2.6). The contact force and joint torque are denoted by F and τ respectively. The link is constrained by a rigid test surface. Various voltage commands corresponding to the joint torque are sent to the motor.

Table 7.1 Joint flexibility test results.

v cmd [V]	F [N]		τ [Nm]		$\delta\psi$ [deg]	
	Test 1	Test 2	Test 1	Test 2	Test 1	Test 2
5	.64	1.93	.60	1.18	.059	.017
6	4.59	2.32	4.03	2.04	.269	.021
7	6.99	7.17	5.32	6.04	.140	.312
8	7.34	7.52	6.84	7.03	.230	.267
9	9.57	8.95	7.59	7.31	.173	.210
10	10.05	9.57	8.17	7.59	.20	.108
12	10.18	10.18	8.88	8.88	.182	.204

Results of several tests prove a high nonlinearity of the joint flexibility. A direct implementation of the joint stiffness measurement technique in a control process can be done, if the system model is very accurate and all other parameters are well defined. It is almost impossible to satisfy all these conditions. Figure 7.1 shows the relationship between the control voltage applied to the motor and the actual joint compliance. For the same voltage input commands the joint compliance can obtain various values. Figure 7.2 reveals the time history of the joint flexibility. During this test a voltage step command is issued to the joint. In the initial period and for a small range of applied torques the joint compliance is very large, in steady state K_g reaches a fixed value. The nonlinearity of the joint flexibility is obvious.

■ Open Loop System Dynamics, Open Loop Force Control

Open loop system dynamics and open loop force control tests were performed based on the designed software "OLCON". The controller output signal necessary to produce the required joint torque is calculated, depending on the desired contact force. The actual joint torque and contact force are measured during the program execution based on the data obtained from the JR3 force torque sensor. Interesting results are shown in Fig. 7.3, for the torque step input $U = 0.032 \text{ Nm}$ (0.6 V). The manipulator link oscillation, characterized by a larger duration time than the noise due to the A/D converter, can be observed. This phenomenon can be explained as follows. To overcome the static friction of the joint it is necessary to apply a voltage command $\approx 0.55 \text{ V}$. Control voltage commands close to 0.55 V cause that the joint is very sensitive to the static friction, which excites periodical oscillation. In the open loop force control various proportional gains are considered. The set of measured data consists of the time, measured contact force, controller voltage and force tracking error. The system responses for inputs 4 [N] and 8 [N] are shown in Fig. 7.4. The open loop controller does not provide a good system performance, as the system is characterized by a large error and overshoot. For identical desired inputs and identical gains different system responses are obtained. Due to the system nonlinearities the controller repeatability is small.

■ PD Force Control Test

The source code for the "PD_T" control program can be found in Appendix III. Several control tests were performed for various gains and desired contact force values. Experimental results are shown in Figs. 7.5 to 7.7. Similar to the open loop control, the steady state error is relatively large. Adding the damping to the system significantly improves its response. The settling time is long ($T_s \approx 1 \text{ s}$) and intensive transient oscillation of the link can be observed (Fig. 7.7). Decreasing proportional gain K_p decreases the settling time ($T_s \approx 0.5 \text{ s}$) and partly eliminates the oscillatory response but the steady state error increases (Fig. 7.7). For proportional gains $K_p > 22$ the link "chattering" can be observed (Fig. 7.6).

■ PID Force Control Test

To eliminate the steady state error resulting in the previously PD control, now the PID controller is implemented. Control results for various inputs and gains are shown in Figs. 7.8 to 7.9. A computer program to perform control process "PID_T" is based on the software designed for the PD control with an integral part added. The controller works well for small force inputs and appropriate chosen gains. The settling time can be long ($T_s \approx 1.5$ s) for inappropriate gains and characterized by transient oscillation (Fig. 7.8). It is difficult to tune up the controller to eliminate the overshoot. For the set of gains $K_p = 0.08$, $K_i = 0.0002868$ and $K_d = 4$ and input 8 N (Fig. 7.9) the steady state error is eliminated, but the force overshoot $\approx 75\%$ and the settling time $T_s \approx 1.3$ s. Similar to the previous examined control schemes, the link "chattering" occurs for inappropriate chosen gains (Fig. 7.8). The random digital noise spikes can be seen in Figs. 7.7, 7.8. This noise can be easily distinguished from the longer in duration link chattering.

■ Adaptive Force Control Test

Two adaptive force control methods were implemented, based on the MRAC technique described in Chapter 6. In the first control scheme a PID controller is added to the plant transfer function. Experimental results are shown in Figs. 7.10 to 7.11. In Fig. 7.10 the system responses are shown for force inputs 4 and 8 N and two sets of design parameters ϵ and λ . The system output is characterized by a relatively fast response. Tuning this controller is a tedious task due to the system response dependency on initial values of parameters ϵ and λ . The second MRAC technique allows to analyze the performance of the adaptation mechanism only, without an additional help from a conventional controller added to the plant. Based on simulation, this controller is not recommended for the considered system. Attempts were made to obtain the system response for a step force input. The MIT rule given by equation 6.2.9 was modified (the absolute value for the expression given by 6.2.9 was substituted) to avoid the run time problems. The expression under the square root obtained negatives values during the execution of the control procedure what caused the program interruption and unpredictable system responses. Experimental results are shown in Figs. 7.12 and 7.13.

Fig 7.1 Flexible robot joint with the PMI harmonic drive Joint flexibility test

Fig 7.2 Flexible joint characteristic Joint torque versus flexibility.

Dynamic tests for a voltage step command. 2 tests for the same inputs.

Fig 7.3 Open loop system dynamics - flexible robot joint, torque step response

Fig. 7.4 Open loop force control of a flexible robot joint
Desired inputs 4 and 8 [N].

Fig. 7.5 Flexible joint force control PD control scheme

Input=4 [N] Tests for various PD gains

Fig 7.6 Flexible joint force control. PD control scheme.
Input= 4 [N] Tests for various gains.

Fig. 7.7 Flexible joint force control. PD control scheme
Tests for various controller gains.

Fig. 7.8 Flexible joint force control. PID control scheme.

Test results for various gains.

Fig. 7.9 Flexible joint force control. PID control scheme
 Test results for various gains

Fig. 7.10 Flexible joint force control. MRAC control scheme.

Plant with a PID controller added. Various design parameters.

Fig. 7.11 Flexible joint force control. MRAC control scheme
 Plant with a PID controller added Various design parameters

Fig 7 12 MRAC control of a flexible joint. Experimental results.

A proportional controller added to the joint transfer function..

Fig. 7.13 MRAC control of a flexible joint Experimental results

A proportional controller added to the joint transfer function

Chapter 8

SUMMARY AND CONCLUSION

The presented flexible joint models and force control schemes vary in the conceptual approach. The most appropriate dynamic model of the considered flexible joint with the harmonic drive is a discrete time model with two values of the joint flexibility. The joint stiffness of this model depends on the joint output torque. Simulation results based on these models provide valuable information for the control system design. The uncontrolled system exposed to the step input is characterized by the high frequency oscillation, due to the small motor inertia, very small damping and relatively high joint stiffness. Presented closed loop force control simulations with force/torque sensory feedback in general did not discover a possibility of the link "chattering". Only the MRAC control described in Chapter 6.2.2 exhibits an oscillatory response, which may be a source of "chattering" (Figs. 6.2.3.2 and 6.2.3.3). Experimental results reveal, that the link "chattering" may occur for all tested controllers and is caused by the discrepancy between the model and the real plant. Major sources of errors between the model and the real system are the joint friction and joint stiffness. The inappropriate approach in the force control system design with the force feedback leads to the manipulator end effector instability and link "chattering". The link "chattering" of a rigid link/flexible joint manipulator can be caused by the following factors

- approach to the control system dynamic modelling
- force/torque feedback
- joint structural compliance
- nonlinear system parameters

- feedback signal acquisition and filtering
- control signal saturation
- sampling time choice

Adaptive control scheme presented in Chapter 6.2.2 is characterized by a fast response and good force trajectory tracking. It is very sensitive for the control parameter variation. For an inappropriate adaptive gain a large tracking error and significant overshoot exist. System may become easily unstable.

The implemented closed loop control with a force/torque sensor proved the advantage over the open loop control for steady state error elimination and reducing existence of nonlinear friction. The open loop control designed in Chapter 5.1 requires precisely designed controller gains. Experimental results have shown, that for the same gains various real time step responses can be obtained. To maintain the constant desired output it was necessary to adjust the proportional gain at each test. The disturbed system is characterized by a limited robustness and a large overshoot.

The PD and PID controlled systems exhibited in some cases a transient oscillation (Figs. 7.7, 7.8, 7.9). The settling time and overshoot become large, if inappropriate gains are applied. System can easily become unstable and "chattering" occurs. In experiments a relatively small sampling time was obtained ($T_s = 6$ ms). A closer investigation of the high frequency related to the joint and sensor flexibility would require a faster control system. To avoid the control signal saturation, the system requires compensating techniques precisely predicting the control voltage, especially for larger desired outputs. An initial idea to implement the joint flexibility obtained on line directly into the control scheme, failed. It seems to be a good concept, but it is difficult to establish a clear relationship between the actual joint torque and joint stiffness. This would be more tedious for a more complex manipulator. Increasing the proportional gain in all presented classical techniques eliminates the effect of system nonlinearities. Simultaneously, the overshoot increases and the link "chattering" occurs for larger values of gains.

Summarizing obtained results, the following are major conclusion arising from this comprehensive studies on flexible joint control

- The application of the force feedback provides valuable information about the actual joint torque, contact force, contact friction and joint flexibility .
- Wrist mounted force sensor allows the estimation of the contact slip-stick friction value and reduces the effect of its existence in control process.
- The high frequency dynamics caused by the force sensor structure does not have a major influence on the system performance. This is due to the natural damping of the high frequency modes
- The "chattering" phenomenon occurs even for a single flexible joint manipulator under specific conditions
- A standard modelling of the joint flexibility as a linear spring can not be applied for the considered joint equipped with the harmonic drive PMI H6D. The model with two values of the joint flexibility is proposed (Chapter 4.2.3).

Following are recommendation and future plans related to the presented research:

- To improve the performance of presented control schemes with sensory feedback it is suggested to use a faster apparatus and improve the feedback signal filtering.
- To investigate the relationship between the joint coupling and link "chattering", the SLIM-1 manipulator should be extended to more degree of freedom. This can be easily done, because all components of the second joint and elements of control system are designed and configured
- It is proposed to build a variable impedance machine to test the developed controllers in various environmental conditions. The concept of such device already exist and can be easily implemented
- Presented in Chapter 6.3 force control scheme requires an improved control algorithm and stability evaluation criteria

REFERENCES

- [1] T. Yoshikawa, *Foundation of Robotics*, 1990 MIT
- [2] H. Sira Ramirez, S. Ahmad, M. Zribi, *Dynamical feedback control of robotic manipulators with joint flexibility*, IEEE Trans. on System, Man and Cybernetics, vol. 22 No. 4, Aug. 1992
- [3] Y. Zheung, Y. Fan, *Robot force sensor interacting with environment*, IEEE Transaction on Robotics and Automotion, vol. 7, No. 1 Feb. 1991
- [4] J. Mills, *Dynamics of Robotic manipulators with wrist mounted force-torque sensor. A singular perturbation approach*, IEEE Transaction on Robotics and Automation, vol. 7, No. 6, Dec. 1991
- [5] C. Hung, C. Weng, *Modal analysis of a controlled multi-link system with flexible links and joints*, J. of Guidance, C and D, vol. 12, No. 3, May-June 1992
- [6] L. Pfeifer, O. Khatib, J. Hake, *Joint torque sensory feedback in the control of a Puma manipulator*, IEEE Transaction on Robotics and Automation, vol. 5, No. 4, August 1989
- [7] J. Gonzales, G. Widmann, *Investigation of nonlinearities in the force control of robots*, IEEE Transaction on System, Man and Cybernetics, vol. 22, No. 5, Sept/Oct. 1992
- [8] S. Chiaverini, B. Siciliano, L. Villiani, *A stable force position controller for robot manipulators*, Proceedings on the 31st Conference on Decision and Control, Tucson, Arizona, Dec. 1992
- [9] B. Yao, S. Chan, D. Wang, *Robust motion and force control of robot manipulators in the presence of environmental, constraint uncertainties*, Proceedings on the 31st Conference on Decision and Control, Tucson, Arizona, Dec. 1992
- [10] H. Qian, J. Schutter, *Stabilizing robot force control through low pass filtering with low cut off frequency*, Proceedings on the 31st Conference on Decision and Control, Tucson, Arizona, Dec. 1992
- [11] C. Wu, *Compliance control of a robot manipulator based on joint torque servo*, Int. Journ. of Robotics and Research, vol. 4, No. 3, Fall 1985
- [12] K. Ogata, *Modern control engineering*, Prentice Hall, Inc. 1970
- [13] D. Whitney, *Historical perspective and state of art in robot force control*, Int. Journal of Robotics Research, vol. 6 No. 1, Spring 1987
- [14] K. Khorasani, *Adaptive control of flexible joint robots*, IEEE Transaction on Robotics and Automation, vol. 8, No. 3, April 1992
- [15] C. Su, T. Leung, Q. Zhou, *Force motion control of constrained robots using sliding mode*, IEEE Transaction on Automatic control, vol. 37, No. 5, May 1992
- [16] F. Ghorbel, J. Hung, M. Spong, *Adaptive control of flexible joint manipulators*, IEEE Transaction on Robotics and Automation, Dec. 1989
- [17] R. Lozano, B. Brogliato, *Adaptive control of robot manipulators with flexible joints*, IEEE Transaction on Automatic Control, vol. 37, No. 2 Feb. 1992
- [18] K. Jankowski, H. ElMaraghy, *Dynamic decoupling for hybrid control of rigid flexible joint robots interacting with the environment*, IEEE Transaction on Robotics and Automation, vol. 8, No. 5, Oct. 1992

- [19] K. L. Lian, J. H. Jean, I. C. Pu, *Adaptive force control of single-link mechanism with joint flexibility*, IEEE Trans. on Robotics and Automat, vol. 7, No. 4, August 1991
- [20] N. Sadegh, K. Guglielmo, *Design and implementation of adaptive and repetitive controllers for mechanical manipulators*, IEEE Trans. on Robot. and Automation, vol. 8, No. 3, June 1992
- [21] J.Z. Sasiadek, R. Srinivasan, *Dynamic modelling and adaptive control of a single link flexible manipulator*, J. Guidance, vol. 12, No. 6, Dec. 1989
- [22] S. Eppinger, W. Seering, *Introduction to dynamic models for robot force control* IEEE Control System Magazine, April 1987
- [23] J. K. Mills, *Stability and control of elastic-joint robotic manipulators during constrained-motion task*, IEEE Transaction on Robot. and Automation, vol. 8, No. 1, February 1992
- [24] M.C. Readman, P.R. Belanger, *Stabilization of the fast modes of a flexible-joint robot*, The Internat. Journal of Robotics and Research, vol. 11, No. 2, April 1992
- [25] H. Kazeroni, *Automated robotic deburring using impedance control*, IEEE Control System Magazine, Feb. 1988
- [26] H. Kazeroni, T. Sheridan, P.K. Houpt, *Robust compliant motion for manipulators, Part I: The fundamental concepts of the compliant motion*, IEEE Journal of Robotics and Automation, vol. ra-2, No. 2, June 1986
- [27] N. Hogan, *Impedance Control: An approach to manipulation: Part I- Theory*, Journal of Dynamics Systems, Measurement, and Control, March 1985, vol. 107/1
- [28] N. Hogan, *Impedance Control: An Approach to Manipulation: Part II- Implementation*, Transactions of the ASME, Journal of Dynamics Systems, Measurement and Control, March 1985, vol. 107/9
- [29] N. Hogan, *Impedance Control: An approach to manipulation: Part III- Applications*, Transactions of the ASME, Journal of Dynamics Systems, Measurement, and Control, March 1985, vol. 107/17
- [30] H. Kazerooni, P.K. Houpt, B. Sheridan, *Robust compliant motion for manipulators, Part II, Design Method*, IEEE Journal of Robotics and Automation, vol. ra-2, No. 2 June 1986
- [31] C.M. Liaw, *Design of a two-degree-of-freedom controller for motor drives*, IEEE Transactions on Automatic Control, vol. 37, No. 8, August 1992
- [32] J.Y. Luh, *Conventional controller design for industrial robots- a tutorial*, IEEE Transactions on Systems, Man and Cybernetics, vol. smc-13, No. 3, May/June 1983
- [33] J. Mills, A. Goldenberg, *Force and position control of manipulators during constrained motion tasks*, IEEE Transactions on Robotics and Automation, vol. 5, No. 1, Feb. 1989
- [34] M. Raibert, J. Craig, *Hybrid position force control of manipulators*, ASME Transaction, Journal of Dynamics, System, measurement and Control, vol. 102 June 1981, pp 126-133
- [35] H. Asada, K. Youcef-Toumi, S.K. Lim, *Joint torque measurement of a direct-drive arm*, Proc. 23rd IEEE Conf. on Decision and Control, pp 1332-1337, Dec. 1984
- [36] M. Spong, *Adaptive control of flexible joint manipulators*, System control Lett., vol. B, pp 15-21, 1989

- [37] De Simone, L. Nicolo, *On the control of elastic robots by feedback decoupling*, Int Journal of Robot. Automat., Feb 1986
- [38] L. Cai, A.A. Goldenberg, *An approach to force and position control of robot manipulators*, Proc IEEE Conf Robotics Automation, May 1989, pp 86-91
- [39] H. Faessler, *Manipulators constrained by the stiff contact: Dynamics, control and experiments*, Int. J. Robotics, vol. 9, pp 40-59, Aug 1990
- [40] K.P. Chen, and L.L. Fu, *Nonlinear, adaptive motion control for the manipulator with flexible joints*, Proc IEEE Int. Conf. Robotics Automat, 1989
- [41] Z. Borowski, *Force/Torque control in robotic applications, Part I, Report*, Department of Mechanical and Aerospace Engineering, Carleton University, Ottawa, Dec. 1991
- [42] Z. Borowski, *Force/torque control of the flexible joint robot*, Presentation material, Carleton University, Department of Mechanical and Aerospace Engineering, Ottawa, Nov 1992
- [43] S. Lin, S. Tosunoglu D. Tesar, *Control of a six-degree of freedom flexible industrial manipulator*, IEEE Transaction on Control System, April 1991
- [44] M. Dawson, F. Lewis, J. Dorsey, *Robust control of a robot manipulator*, Int. Journal of Robotics Research, vol. 11, No. 4, August 1992

APPENDIX I

System Paramters

APPENDIX I/I

- System Parameters -

k_a 1.76,	amplifier gain [A/V]
k_m 0.301,	motor torque constant [Nm/A]
b_l 0,	load damping
b_m 0.000105,	motor damping [Nm/rad/s]
J_t 18×10^{-4}	tachometer inertia [Nms ²]
J_e 18×10^{-4} ,	encoder inertia
J_m 6×10^{-4} ,	motor inertia [Nms ²]
m 72,	sensor mass [kg]
m_l 45,	link mass [kg]
N 160,	gear ratio
l_t 384,	link length
L 545,	link total length (with the sensor tip)
x 24,	contact point x coordinate
T 5.63×10^{-4} ,	amplifier time constant
V_{max} 2,	max control voltage [V]
μ 0.1,	the static friction coefficient
α_0 45.15,	the angle of orientation
J_l 0.47,	link and sensor moment of inertia [Ns ² /m]
K_1 2599,	joint stiffness (1) [Nm/rad]
K_2 13560,	joint stiffness (2) [Nm/rad]
E 9.9×10^4 ,	Young modulus for aluminium alloy [psi]
TM	total test time
h	sampling time

APPENDIX I/II**-Link and sensor moment of inertia-**

Calculation of the link and sensor rotational moment of inertia (see Fig. A1)

Fig. A1 Manipulator link dimensions

J_L - Link moment of inertia, J_s - Sensor moment of inertia, J - Effective inertia

$$J_L = M_L \left(\frac{(L + 1)^2}{12} + X^2 \right) = 0.047$$

$$J_s = M_s (L + 1)^2 = 0.237$$

$$J = J_L + J_s = 0.2856 \text{ [Ns}^2/\text{m]}$$

APPENDIX I/III**- Manipulator link design -**

For profiles 1 to 4 shown in Fig. 3.2.1 the maximum static deflection is calculated. The link with a circular cross-section was chosen (profile No. 4). The material used for the link is an aluminium alloy, Cat. No. 9178, characterized by the Young modulus $E = 9.9 \times 10^{10} \text{ lb/in}^2$. Deflections of the link end for various profiles, for both steel and aluminium, are given in Table A1.

Table A1 Manipulator link design - static deflection for various profiles (See Fig. A1)

No	second moment of area	Material Steel $E=28*10^6 \text{ lb/in}^2$	Material Al. $E=9.9*10^6 \text{ lb/in}^2$
1	$I = (BH^3 - bh^3)/12 = 1.748 \text{ in}^4$	$y = 0.00613 \text{ [in]}$	$y = 0.00172 \text{ [in]}$
2	$I = (BH^3 - bh^3)/12 = 2.07 \text{ in}^4$	$y = 0.00517 \text{ [in]}$	$y = 0.00145 \text{ [in]}$
3	$I = bh^3/12 = 1.125 \text{ in}^4$	$y = 0.0095 \text{ [in]}$	$y = 0.00267 \text{ [in]}$
4	$I = \pi(D^4 - d^4)/64 = 6.37 \text{ in}^4$	-----	$y = 0.00178 \text{ [in]}$

The static deflection y is given by $y = (F_{\max} * L^3) / 3EI$, where: $F_{\max}$ - is the maximum static force acting on the link (here assumed 44 N), I - is the second moment of area given in Table A1

APPENDIX II

Simulation Software

APPENDIX III/I

-Simulation Software-

```

function [Y,X] ds_1s
    %function to simulate the uncontrolled dynamics
    %open loop transfer function without friction
    %simulated is the system step input response
    %master's thesis project
    %written by Zbigniew Borowski, January 93
    %''' program should be run with system parameters
    % given in file "sysparam" in App I '''
    disp('          - PROGRAM DS_1S.M ->'),
    disp('Flexible joint manipulator dynamics simulation');
    % ***** system parameters *****
    m1 = 61;           %static friction coefficient
    alpha = 45/180;    %the angle of orientation (alpha)
    TM = 5;            %total test time
    h = 0.04;          %sampling time
    %-----
    Fmax = (Vmax*km*ka*N)/((sin(alpha)+m1*cos(alpha))*L);
    disp('The maximum contact force [N] =');
    disp(Fmax);
    Fd = input('Enter the desired contact force [N] ->');
    % ----- computation of desired moment and stiffness -----
    Md = Fd*(sin(alpha)+m1*cos(alpha))*L;
    if Md < 8.43         %20°375 in-lbs(42.1875 N-m)
        kg = 2590;      % [N-m/rad]
    else
        kg = 13560;
    end
    %----- built the open loop transfer function Ma/Md-----
    format long
    num = N*kg;          %system tr. func. w/o friction
    den = [Jm bm kg];
    %----- running the system -----
    t = 0:h:TM;
    y = step(num,den,t);
    y_a = sqrt((sin(alpha)+cos(alpha)*m1)*L);
    subplot(211); plot(t,y);
    title('output torque, step response')
    xlabel('time [s]')
    ylabel('output torque [N-m]')
    grid
    subplot(212); plot(t,y_a);
    title('contact force, step response');

```

```

xlabel('time [s]')
ylabel('Contact force [N]')
omn = sqrt(kg/Jm);
disp('natural frequency = ').disp(omn). disp('[rad/sec]');

```

APPENDIX II/II

-Simulation Software-

```

function [Y,X] = ds_1p
%function to simulate the uncontrolled dynamics
%open loop transfer function without friction
%simulated is the system impulse response
%master's thesis project
%written by Zbigniew Borowski, January '93
%''' program should be run with system parameters
% given in file "sysparam" in App 1'''
disp('          PROGRAM DS_1P.M ');
disp('Flexible joint manipulator dynamics simulation');
% ***** system parameters *****
m = 6;
alpha = 45/15;
TM = 5; %total test time
h = 0.02; %sampling time
%-----
Fmax = (Vmax*km*ka*N)/((sin(alpha)+m*cos(alpha))*L);
disp(' the maximum contact force = 26 N ');
disp(Fmax);
Fd = input('Enter the desired contact force [N] = ');
%computation of desired moment and contact friction-----
Md = Fd*(sin(alpha)+m*cos(alpha))*L;
if Md > 8.43 %20*3.75 in-lbs(42.1875N-cm) omt
    kg = 2599; %[Nm/rad]
else
    kg = 13560;
end
%----- built the open loop transfer function Ma/Md-----
format long
num = N*kg %system tr func w/o frict
den = [Jm bm kg]
%----- running the system -----
clf
t = 0:h:TM;
y1 = impulse(num,den,t);
y1_a = y1/((sin(alpha)+cos(alpha)*m)*L);
subplot(211). plot(t,y1)

```



```

title('output torque, impulse response')
xlabel('time [s]')
ylabel('output torque [N-m]')
grid
subplot(212) plot(ty1,a)
title('contact force [N]')
xlabel('time [s]')
ylabel('contact force [N]')
grid

```

APPENDIX II/III

-Simulation Software-

```

function [Y,X] ds_rs(Md)
%function to simulate the uncontrolled dynamics
%open loop time variant transfer function without friction
%simulated is the system step input response
%master's thesis project
%written by Zbigniew Borowski, January '93
%!!!! program should be run with system parameters
% given in file "sysparam" in App I !!!!
disp('          PROGRAM DS_RS.M - '),
disp('Flexible joint manipulator dynamics simulation'),
disp('Time variant system, motor friction included'),
TM=8.4e-02;
%-----
Lmax=(Vmax*km*ka*N)/((sin(alpha)+mi*cos(alpha))*L);
disp('For this configuration the maximum contact force [N] ='),
disp(Lmax);
Md=input('Enter the desired torque input [Nm] ->');
%-----initial values -----
t=0;
time(t)=0;
vd= Md;
xx=[0 0]';
xc=10*xx;
fa(t)=0;
vx(t)=0;
km=(sin(alpha)+cos(alpha)*mi)*L;
clc
%----- running the system -----
while time(t) < TM;
    [AA,BB,CC,FF]=tra_f(yy(t));
    frict=0.0064*sign(xx(2));
    BF=[BB FF];

```

```

[AAd,BFd]=c2d(AA,BF,h); %discretizing the system
U(j)=[yd frict]';
xx=AAd*xx+BFd*U(j); %plant dynamics
yy(j+1)=Kp*CC*xx;
x(j+1)=xx;
Fa(j+1)=yy(j+1)/km;
j=j+1;
time(j)=time(j-1)+h;
end
%----- plot graphs -----
U(j)=[yd frict]';
subplot(221).plot(time,yy,time,U(1,:),'.');
title('torque [Nm].step response')
xlabel('time [s]')
ylabel('output torque [Nm]')
grid
pause
subplot(222).plot(time,Fa)
title('contact force [N].step response')
xlabel('time [s]')
ylabel('Contact force [N]')
grid
pause
subplot(223).plot(time,x(1,:))
title('motor angle [rad]')
xlabel('time [s]')
ylabel('motor angle [rad]')
grid
pause
subplot(224).plot(time,x(2,:),time,i(2,:)*5,'*')
title('motor vel [rad/sec]')
xlabel('time [s]')
ylabel('vel [rad/sec].friction*5')
grid
pause
%----- display -----

```

APPENDIX II/IV

-Simulation Software-

function [Y,X]=olsim_2

```

%function to simulate force control
%open loop transfer function without friction
%simulated is the system step response
%master's thesis project

```

```

%written by Zbigniew Borowski, January 93
disp('          PROGRAM OPSIM 2 M -').
disp('Open loop force control of a flexible joint manipulator').
disp('System step response')
% ***** rs *****
TM (Enter test time) ,          %total test time
h (Enter sampling time) ,       %sampling time
kp (Enter proportional gain  $K_f$ ) , %proportional gain
kg input(Enter joint stiffness ->)
ka 176,                         % A/V amplifiler gain
km 0301,                        % N*cm/A motor torque constant
bl 0,                           %load damping
bm 0000105,                     %motor damping Nm/rad/s
Jt 18*10 (-7),                  %tachometer inertia kg*cm^2
Je Jt,                          %encoder inertia
Jm 6*10 (-6),                   %motor inertia
Jm Je+Jt+Jm,                    %total inertia on motor side
ms 72 ,                         %sensor mass [kg]
ml 45 ,                         %link mass [kg]
N 160,
L 545,                          %link total length (with the sensor tip)
T 5.63*10 (-4)                  %amplifier time constant
% computation of desired moment and contact friction ---
t 0:h:TM,
for i 1:length(t),              %input,
    time t*h,
    Fd(i) abs(sin(10*time)),
end
Fkl Fd*ml,
Md (Fd*sin(alpha)+Fkl*cos(alpha))*L, %desired moment [Nm]
%----- built the open loop transfer function Ma Md-----
num kp*N*ka*kg*km,             %system transf function w/t friction
den [Jm bm kg],
%----- running the system -----
v lsim(num,den,Fd,t),
subplot(211), plot(t,v,t,Fd,'-')
title('contact force, des. force [N]')
xlabel('time [s]')
ylabel('force [N]')
grid
subplot(212), plot(t,v*(sin(alpha)+ml*cos(alpha))*L)
title('output torque')
xlabel('time [s]')
ylabel('torque [Nm]')

```

APPENDIX II/V**-Simulation Software-**

```

function [Kp,Kd,poles] = pdgain(Fd,alpha)
    %function to find gains for the PD controller
    %open loop transfer function without friction
    %master's thesis project
    %written by Zbigniew Borowski, January '93
    %!' include "sysparam" from App I to run this program
    Fd= 1,TM= 1,h= 0.04,
    alpha= input('enter the desired (alpha) - ').
    G= N*ka*kg*km,      %system proportional gain
    %-----
    t= 0:h:TM,
    numol=[1 alpha],    %open loop transf function w/ t fr, PD control
    denol=[Jm bm kg],
    rlocus(numol,denol),
    [gain,poles]= rlocfind(numol,denol),
    title('ROOT LOCUS - PD CONTROL')
    format long
    Kd= gain/G
    Kp= Kd*alpha
    disp('gain = ').disp(gain)
    disp('poles = ').disp(poles)

```

APPENDIX II/VI**-Simulation Software-**

```

function [Kp,Kd,poles] = pd_resp(alpha)
    %function to find gains for the PD controller
    %open loop transfer function without friction
    %master's thesis project
    %written by Zbigniew Borowski, January '93
    %!'To run this program include "sysparam" from App I-----
    Fd= 1,TM= 0.002,h= 0.0000004,TM1= 1,h1= 0.04
    alpha= input('enter the desired (alpha) - ').
    Kg= input('enter joint stiffness - ').
    G= N*ka*Kg*km,      %system proportional gain
    %-----
    Kd= input('Enter Kd')
    Kd1= input('Enter Kd (1)')
    clg
    t= 0:h:TM,
    t1= 0:h1:TM1,
    numol= Kd*G*[1 alpha], %open loop transf function w/ t fr, PD control
    denol= [Jm bm Kg],

```

```

numcl = G*Kd*[1 alpha],
dencl = [Jm G*Kd+bm Kg+G*Kd*alpha],
numcl1 = G*Kd1*[1 alpha],
dencl1 = [Jm G*Kd1+bm Kg+G*Kd1*alpha],
v = step(numcl,dencl,t),
v1 = step(numcl1,dencl1,t),
v2 = step(numcl,dencl,t1),
v3 = step(numcl1,dencl1,t1),
subplot(211), plot(t,y,t,y1,'--')
title('PD FORCE CONTROL (1) - F [Nm]')
xlabel('TIME [s]')
pause
subplot(212), plot(t1,y2,t1,y3,'--')
title('PD FORCE CONTROL (2) - F [Nm], Ess')
xlabel('TIME [s]')

```

APPENDIX II/VII

-Simulation Software-

```

function [Kp,Kd,poles] = pd_resp(alpha)
%function to find gains for the PD controller
%open loop transfer function without friction
%master's thesis project
%written by Zbigniew Borowski, January 93
%!! include file "sysparam" from App I to run this program-----
Fd = 1;
alpha = input('enter the desired (alpha) -> ');
al = 45/15;
TM = 0.002, Jh = 0.0000004, TM1 = 1, h1 = 0.04
G = N*ka*Kg*km, %system proportional gain
Kd = input('Enter Kd')
Kd1 = input('Enter Kd (1)')
Kg = input('enter joint stiffness -> ');
t = 0:h:TM,t1 = 0:h1:TM1,
numol = Kd*G*[1 alpha], %open loop transf function w/t fr, PD control
denol = [Jm+bm Kg],
numcl = G*Kd*[1 alpha],
dencl = [Jm G*Kd+bm Kg+G*Kd*alpha],
numcl1 = G*Kd1*[1 alpha],
dencl1 = [Jm G*Kd1+bm Kg+G*Kd1*alpha],
v = step(numcl,dencl,t),
v1 = step(numcl1,dencl1,t),
v2 = step(numcl,dencl,t1),
v3 = step(numcl1,dencl1,t1),
%----- plot error -----

```

```

Kp=alpha*Kd.
y_err=1-y.
y_err2=1-y2.
for i=1:length(y)-1.
    volt(i)=Kp*y_err(i)+(Kd/h)*(y_err(i+1)-y_err(i)).
    i=i+1.
end
for i=1:length(y2)-1.
    volt2(i)=Kp*y_err2(i)+(Kd/h1)*(y_err2(i+1)-y_err2(i)).
    i=i+1.
end
t=0:h:TM-h.
t1=0:h1:TM1-h1.
subplot(211). plot(t,volt)
title('CONTROL VOLTAGE (1) [V] Ts=0000004 [s]')
xlabel('TIME [s]')
subplot(212). plot(t1,volt2)
title('CONTROL VOLTAGE (2) Ts=0004 [s] [V]')
xlabel('TIME [s]')

```

APPENDIX II/VIII

-Simulation Software-

```

function pdvolt=1(alpha)
%this function has to be included into program "pd_resp"
Kp=alpha*Kd.
y_err=1-y.
y_err2=1-y2.
for i=1:length(y)-1.
    volt(i)=Kp*y_err(i)+(Kd/h)*(y_err(i+1)-y_err(i)).
    i=i+1.
end
for i=1:length(y2)-1.
    volt2(i)=Kp*y_err2(i)+(Kd/h1)*(y_err2(i+1)-y_err2(i)).
    i=i+1.
end

```

APPENDIX II/IX

-Simulation Software-

```

function [Y,X,U,FR] pdfsim(Kg,FR)
%-----
%PD control of a flexible joint, State-space approach
%written by Zbigniew Borowski, Ottawa, March 28,1993
%----- input parameters -----
%xx(1) - vector of initial states

```

```

% FR - initial, estimated value of joint friction
%----- program output -----
% Y - actual contact force, X - state vector .FRIC - friction
%!include file "sysparam " in App 1 to run this program-----
km 0301, bm 0000105, N 160,
TM 0004, h 000004,
FR 0064, % [Nm] motor friction
Fd input('Enter the desired contact force ')
kg input('Enter joint stiffness -->')
Kd input('Enter Kd -->')
alpha input('Enter (alpha) -->')
Kp alpha*Kd,
%-----initial states -----
t 1,x(j) 0
xx [0 0]';x(j) xx;
%-----build the system-----
JAC (cos(alpha)+mi*sin(alpha))*L,
format long
a11 0, a12 1, a21 -kg/Jm, a22 -bm/Jm,
A [a11 a12,a21 a22],
b1 0, b2 1/Jm,
B [b1 b2]',
c1 N*kg, c2 0,
C [c1 c2], F [0 1 Jm]',
%-----run system -----
time(j) 0,
while time(j) < TM
    AA eye(2)+Kd*B*C*ka*km,
    Ac inv(AA)*(A-Kp*B*C*ka*km),
    Bc inv(AA)*Kp*B*ka*km*Fd,
    BB [Bc F],
    [Ad,Bd] c2d(Ac,BB,h), %system discretization
    friction(j) FR*sign(xx(2)),
    U(j) [Fd friction(j)]', %input vector
    xx Ad*xx+Bd*U(j), %system dynamics
    x(j) xx,
    y(j+1) C*xx, %system output
    j j+1,
    time(j) time(j-1)+h,
end
%-----plot results -----
U(j) [Fd friction(j-1)]', %input vector
x(j) xx,
friction(j) friction(j-1),

```

```

subplot(221). plot(time,y)
title('FORCE [N]')
xlabel('TIME [s]')
subplot(222). plot(time,x(1,))
title(' POSITION [RAD]')
xlabel('TIME [s]')
subplot(223). plot(time,x(2,))
title('VELOCITY [RAD SEC]')
xlabel('TIME [s]')
subplot(224). plot(time,friction)
title('FRICTION [Nm]')
xlabel('TIME [s]')

```

APPENDIX II/X

-Simulation Software-

```

function [Kp,Kd,Ki,poles] =pidgain(alpha,beta)
    %function to find gains for the PID controller
    %open loop transfer function without friction
    %master's thesis project
    %written by Zbigniew Borowski, March 93
    %!! include file "sysparam"from App I to run this program-----
    h= 0000004          % sampling time
    TM= 0004            % simulation time
    TM1= 6 , h1= 004 ,
    G1= (km*N*ka*kg)/Jm, %system proportional gain
    gain= input('enter the desired gain G1*Kd')
    alpha= input('enter the desired (alpha) - '),
    beta= input('enter the desired (beta) - '),
    Kd= gain/G1,
    Kp= Kd*alpha,
    Ki= Kd*beta,
    %-----plot the closed loop response-----
    numcl= gain*[1 alpha beta], %transf function without friction
    dencl= [1 2*(bm/Jm)+G1*Kd (kg/Jm)+G1*Kd*alpha G1*Kd*beta]
    t=0:h:TM,
    t1=0:h1:TM1,
    yloop= step(numcl,dencl,t),
    yloop1= step(numcl,dencl,t1),
    subplot(211). plot(t,yloop)
    title('PID CONTROL, FORCE [N], TRANSIENT RESPONSE')
    xlabel('TIME [s]'),
    subplot(212). plot(t1,yloop1)
    title('PID CONTROL, FORCE [N], STEADY STATE'),
    xlabel('TIME [s]'),

```



```

%-----Routh-Hurwitz stability criteria-----
gamma = 1.09375 + 22947443646*Kd;
a1 = (1/gamma)*((gamma + G1*Kd)*((kg/Jm) + G1*Kd*alpha) - (G1*Kd*beta));
disp('----stability-----')
disp('a1 =')
disp(a1)
disp('a3 =')
a3 = (1/a1)*a1*((Jm/kg) + G1*Kd*alpha);
disp(a3)
disp('closed loop poles are ')
format long

```

APPENDIX II/XI

-Simulation Software-

```

function [Y,X,K,P] = optcont_1(Kg,FR)
%Optimal PID Control of a flexible joint, State-space approach
%Designed are Optimal Controller Gains, Q and R matrixes are
%adjusted to satisfy the system performance
%written by Zbigniew Jan Borowski,
% Canadian Space Agency, Ottawa, Jan.,1993
%----- input parameters -----
% xx - vector of initial states,FR- friction
%----- program output -----
% Y - actual contact force, X - state vector ,FRIC - friction
%'include file "sysparam" to run this program friction considered -
Fd = input('Enter the desired contact force ');
kg = input('Enter joint stiffness -->');
xi = input('Enter -xi- (Q- motor position) -->');
psi = input('Enter -psi- ((Q- integral of motor position) -->');
% build the system -----
Md = Fd/JAC; %desired moment
format long
a11 = 0; a12 = N*kg; a13 = 0; a21 = -1/(Jm*N); a22 = -bm/Jm;
a23 = 0;a31 = 1 (N*kg);a32 = 0;a33=0;
A = [a11 a12 a13;a21 a22 a23;a31 a32 a33];
b1 = 0; b2 = (ka*km)/Jm; b3 = 0;
B = [b1 b2 b3]';
c1 = 1; c2 = 0; c3 = 0;
C = [c1 c2 c3]; F = [0 1 Jm 0]';
%-----initial states -----
xini = [Md 0 0]';
%----- find optimal set of controller gains -----
Q = [xi 0 0; 0 1 0; 0 0 psi];
R = 20;

```

```

[Kop,SR] = lqr(A,B,Q,R);
disp('Controller Optimal Gains :')
disp(Kop)
disp('The steady state solution for Ricatti Eq. is :')
disp(SR)
%
% run the system -----
x0 = x_ini;
y(1)=0;
err(1)=Fd;
volt(1)= 0;
AC=A-B*Kop; %closed loop matrix
disp('The system closed loop poles are ')
pol=poly(AC);
roots(pol)
disp('Press any key to continue')
pause
clear x;
time=0;
while time < TM
    xp= expm(AC*time)* x0;
    t(j+1)= time;
    x( j+1) =xp;
    y(j+1)=Fd-(C*xp)/JAC %system output-contact force
    err(j+1) = Fd-y(j+1);
    volt(j+1) = Kop*xp;
    j = j+1;
    time = j*h;
end
%-----plot results -----
subplot(221), plot(t,y)
title('FORCE [N]')
xlabel('TIME [s]')
subplot(222), plot(t,err)
title('ERROR [N]')
xlabel('TIME [s]')
subplot(223), plot(t,(x(1, :)*JAC)/(N*kg))
title(' POS [RAD]')
xlabel('TIME [s]')
subplot(224), plot(t,volt)
title('CONTROL [V]')
xlabel('TIME [s]')

```

APPENDIX II/XII

-Simulation Software-

```

function [ym,yp] adamrac(Kg,Fd)
%-----Discrete time control -----
%Adaptive control of a flexible joint, State-space system model,
%Model Reference Approach and MIT Rule are applied
% written by Zbigniew Borowski, Canadian Space Agency,
% Ottawa, January 1993
%----- program information -----
%PID controller designed in Chap 6 2 3 is a reference model
%Motor Friction is included in the reference model
%Plant is disturbed with an external input acting on the link end
%The problem is to maintain the constant contact force
%'include "sysparam" to run the program-----
h= 0.00005 %input('enter sampling time')
TM= 0.01 %input('Enter simulation time')
TM1= 1.5 , h1= 0.1 ,
kg= 2599,
Fd= input('enter the desired contact force - '),
Md= Fd*JAC,
G1= (km*N*ka*kg)/Jm, %system proportional gain
Fk= input('Enter disturbance - joint friction - '),
Kd1= 0.0002868, %input('Enter Kd1 - plant gain ->')
alpha1= input('Enter alpha1 - plant param - ')
Kp1= Kd1*alpha1,
%----- Enter design parameters and gains-----
gain= 80000, %input('enter the desired gain G1*Kd')
alpha= 12340, %input('enter the desired (alpha) ->'),
beta= 1000000, %input('enter the desired (beta) ->');
xi= input('enter the desired (xi) - '),
lambda= input('enter the desired (lambda) ->'),
Kd= gain/G1,
Kp= Kd*alpha,
Ki= Kd*beta,
%----- build PID Reference Model -----
numcl= gain*[1 alpha beta], %transf function without friction
dencl= [1 2*(bm/Jm)+G1*Kd (kg/Jm)+G1*Kd*alpha G1*Kd*beta],
format long
[A,B,C,D]= tf2ss(numcl,dencl)
%----- plant definition -----
a11= 0, a12= 1, a21= -kg/Jm, a22= -bm/Jm,
AAp= [a11 a12,a21 a22],
b1= 0, b2= 1/Jm,
BBp= [b1 b2],

```

```

c1= N*kg, c2=0,
CCp=[c1 c2], E=[0 1 Jm]';
%-----initial states -----
j=1,time(j)=0,
xx=[0 0 0]', %MRAC initial states
xsp=[ 0 0]'; %plant initial states
x(:,1)=xx,
xp(:,1)=xsp,
yp(j)=0,
y(j)=0,
C1=[1 0]; C2=[0 1], E1=[1 0]',
while time(j)~TM1,
    [Ad, Bd]=c2d(A,B,h1), %model discretization
    M=eye(2)+BBp*Kd1*CCp, % build the plant
    Ap=inv(M)*(AAp-BBp*Kp*CCp), %closed loop plant
    Bp=inv(M)*[BBp E], %input to the plant
    friction(j)=Fk*sign(xx(2)),
    Up=[x1*Fd Fk]', %plant input
    [Apd,Bpd]=c2d(Ap,Bp,h1), %plant discretization
    xsp=Apd*xsp+Bpd*Up,
    xx=Ad*xx+Bd*Fd, %model dynamics
    y(j+1)=C*xx, %model output
    yp(j+1)=CCp*xsp, %plant output,
    %----- error calcul and adaptation scheme -----
    err= yp(j)-y(j), %error between model and plant
    x1=(x1^2 - lambda * h1 * yp(j)*err)^(1/5), %MIT adaptation rule
    j=j+1;
    time(j)=time(j-1)+h1,
end
%-----steady state response graphical output -----
subplot(211), plot(time,y,time,yp,'*')
title('FLEX JOINT MRAC, FORCE [N], SS'),
xlabel('TIME [s]'),
disp('Fig description, pl, mod')
gtext('<-- PLANT (*)')
gtext('<-- MODEL (-)')
%--for transient response case- change the sampling time and total time ---

```

APPENDIX II/XIII

-Simulation Software-

```

function [yp,xp,K,F] modfc(Fd,theta,Kp)
%----- Contact Force Control -----
%Modern Adaptive force control of a flexible joint, State-space system model
%Constant gain output feedback approach is used for the controlled system

```

```

%Disturbance in the form of motor and joint friction is estimated using
% written by Zbigniew Borowski, Canadian Space Agency.
% Ottawa, April 1993
%----- program input and output data-----
%K [K1 K2] observer feedback gains
%teta- design parameter - control signal gain to eliminate disturbance
%'include "sysparam" to run this program-----
%----- Enter design parameters and gains-----
Fd = 1; teta = input('Enter the desired parameter (teta) for input signal -->')
Kp = input('Enter Kp - ');
Kd = input('Enter Kd - ');
h = 0.04; TM = 10; h1 = 0.4;
kg = 2599; kg1 = 2599; kg2 = 13560;
ke = 9.7172*10^4; %effective link/sensor stiffness Nm/rad
JAC = sin(al)*mu*cos(al);
G1 = (km*N*ka*kg)/Jn; %system proportional gain
Fkm = 0.0064; Fkl = 0.0086; %motor and joint initial friction ->)
%----- plant definition -----
format long e
a21 = -kg/Jm; a22 = -bm/Jm; a23 = (kg*N)/Jm; a41 = (N*kg)/Jl; a44 = -bl/Jl;
a43 = -(N*N*kg)/Jl - (ke*L*L)/Jl;
A0 = [0 1 0 0; a21 a22 a23 0; 0 0 0 1; a41 0 a43 a44];
b2 = 1/Jm; e21 = -1/Jm; e42 = -1/Jl; E_1 = [e21 e42];
B0 = [0 b2 0 0]'; C0 = [0 0 L*ke 0]'; E1 = [0 e21 0 0]'; E2 = [0 0 0 e42]';
%----- initial states -----
Fx = Fd*JAC; %Fx is a function of Fd
Md = Fx*L; K = 100; %observer desired gains
Fk = [Fkm Fkl]'; %initial value of friction
%----- ***** RUN THE SYSTEM *****-----
j = 0; F = 0; [0 0 0 0]';
M = eye(4) + Kd*B0*C0; M1 = inv(M);
C = [0 0 L*ke 0]';
%initial value of motor and joint friction
x1d = [0 0 0 0]'; xx0 = [0 0 0 0]';
Fklo = 0; %initial value of the observed load friction
xx = [x1d' Fklo']';
A11 = M1*(A0-B0*Kp*C0);
A12 = -M1*B0*teta; A21 = F_0; A22 = -K;
A = [A11 A12; A21 A22];
B = [M1*B0*Kp M1*E1 M1*E2 0 0 K];
[Ad Bd] = c2d(A,B,h);
while j*h < TM;
    time(j+1) = j*h;
    Fc = [sign(xx(2))*Fkm - sign(xx(4))*Fkl]'; % friction

```

```

    Fklo =sign(xx(4))*xx(5);    %estimated value of friction
    U(j+1)=[M*Ff'];
    xx=[xx0';Fklo];           %new state vector
    xx=Ad*xx + Bd*U(j+1);
    xx0=xx(1:4);
    x(j+1)=xx;
    y(j+1)=C*xx;
    j=j+1;
end
%-----PLOT RESULTS -----
subplot(221). plot(time,y)
title('FORCE [N]')
xlabel('TIME [s]')
subplot(222). plot(time,x(1,:))
title('MOTOR POS [RAD]')
xlabel('TIME [s]')
subplot(223). plot(time,-x(5,:))
title('FRICTION [Nm RAD]')
xlabel('TIME [s]')
%-----to run steady state change h and TM-----

```

APPENDIX III

Real Time Software

APPENDIX III/I**-Real Time Software-****PD force control**

/* Program to control the desired contact force on the link end of the 1 degree of freedom manipulator configuration PD controller is used. Measured are motor and link displacement, the actual force applied to the JR3 sensor and exerted on the environment. Test data is monitored and stored in a file

written by Zbigniew Borowski, February 2, 1993. *

```
#include <dos.h>
#include <stdio.h>
#include <conio.h>
#include <bios.h>
#include <stdlib.h>
#include <math.h>
#include <process.h>
#include <sys/timeb.h>
#include <time.h>
#include <sys/types.h>
#include "C:\ENC\ID_5312.mc"
#include "c:\inc\jr3.h" /*Include file for JR3 sensor*/
#define POS 0
#define NEG 1
#define N 700
#define preset_cnt1 2*1625600
#define preset_cnt2 2*10160 /*number of pulses 2540*/
#define preset_cnt3 2*800000
extern int aot815(),
extern int aot(),
extern int ain(),
extern void init_encoder(int, int, int, int, int, short), /* prototypes */
extern void load_cntr(int, long, short),
extern long read_cntr(int, short),
extern int read_ip(int, short),
int erstat,axis[5],
int board, gain, DA_lchan[1], AD_lchan[15],
int ch, choice,
long cnt[5],
float v_limit, tach_conv,
float enc_resoln 360.0/1625600.0,
float dif_time, step_time,
float j_vel[2],j_vel_filt[2],
float Fx_force,Fy_force,Fz_force,
double rad_to_deg, pi,
double f_time,s_time,
```



```

        double a time, b time, m time, n time,
        double x y,
        double p m,
FILE: *stream,
FILE: *stream1,
char file[12],file1[12],
struct timeb timebuffer,

main()
{
    int i,

    axis[1] = AXIS_A,
    axis[2] = AXIS_B,
    axis[3] = AXIS_C,
    axis[4] = AXIS_D,

main_init();
/* Zero the analog outputs on both D/A physical channels */
    for (ch = 0 ; ch = 1 ; ch++)
        aot (DA_lchan[ch], 0, &erstat),
    for(i = 0; i = 3; i++) _jrz write(COM1, "RO \r", 4),
    for(;;) {
        choice = main_menu(),
        switch(choice)
        {
            case 1
                test(),
                break,
            case 2
                for (ch = 0 ; ch = 1 ; ch++)
                    aot (DA_lchan[ch], 0, &erstat),
                    exit(0),
                    break,
        }
    }

}

test()          *main PD controlprocedure *
{
    char c, ch, done,
    int i, j, result,
    int channel,

```

```

int    i,k,n.
float  tan_theta,theta,dd,ll,yy.
float  v_cmd[N],xolt,xvel[2].
float  flex,GR,jg[3].
float  temp_force,force,torque[N].
float  M_des, M_act,F_act[N],F_des,F_err[N],err,err0.
float  T_k,T_lim,F_lim.
float  T_km,B_m,K_m,K_a.
float  K_p,K_d.
F_lim = 16.
K_m = 0.301,      /* motor torque constant [N-m/Amp] */
K_a = 1.76,      /* [Amp/v] */
T_km = 0.064,    /* motor friction torque [N-m] */
B_m = 0.0001833, /* motor damping [N-cm/deg/s] */
GR=160,ll = 545,yy = 0.92.
j_vel_filt[1]=0.0.
j_vel_filt[2]=0.0.
/*****PROGRAM INPUT *****/
re printf("\n 1) SET THE JOINT INITIAL POSITION ").
printf("\n 2) PRESS - - d IF DONE ! ").
scanf("%s",&done).
if(!(done == 'd'))
    goto re.
printf("\n 3) FILENAME FOR THE DATA STORAGE ").
scanf("%s",file).
stream=fopen(file,"w").
printf("\n 4) FILENAME FOR MATLAB ").
scanf("%s",file1).
stream1=fopen(file1,"w").
enc_init(). /* ENCODER INITIALIZATION*/
printf("\n 5) ENTER JOINT TO BE TESTED ").
scanf("%d",&jnt).
ef printf("\n 6) ENTER DESIRED CONTACT FORCE ").
printf("\n 7) MAXIMUM CONTACT FORCE = 16 [N] ").
scanf("%f",&F_des). /* for static conditions F_des_max = K_m*a*160/ll */
if(F_des<0) /*only + Forces accepted */
    F_des=-F_des.
if(fabs(F_des) > F_lim )
    { printf(" ***** Error entering contact force ! *****").
      goto ef.
    }
M_des=comp_tor(F_des,ll).
printf("\n 7) ENTER K_p - - ").
scanf("%f",&K_p).

```

```

printf(" n 8) ENTER K d - - ").
scanf("%f",&K_d).
printf(" n **** v cmd %6.2f n ",v_cmd). /*
k = 0
    if(jnt == 1) /*test specified channel*/
        channel = 0;
    if(jnt == 2)
        channel = 1;
err0 = 0;
err = M_des;
start();
printf(" n ");
ftime(&timebuffer);
s_time = (double)timebuffer.time*timebuffer.millitm/1000.0;
x = s_time;
while(!kbhit() && (err != 1 || err != -1)){ /*RUN THE SYSTEM */
    ang = jr3();
    F_act[k] = act_for(Fx_force,Fz_force);
    M_act = Fx_force*M;
    F_err[k] = F_des-F_act[k];
    err = M_des-M_act;
    v_cmd[k] = K_p*err + K_d*(err-err0); /*voltage output*/
    err0 = err;
    cnt[0] = read_cnt(axis[2], ENC_BASE); /* read counter */
    cnt[1] = read_cnt(axis[3], ENC_BASE); /* read counter */
    cnt[2] = read_cnt(axis[4], ENC_BASE); /* read counter */
    * Conversion of encoder count to ang displ in degrees: */
    jg[0] = (float) (cnt[0] - preset_cnt1) / 1625600 * 360.0;
    jg[1] = -(float) (cnt[1] - preset_cnt2) / 10140 * 360.0;
    jg[2] = (float) (cnt[2] - preset_cnt3) / 800000 * 360.0;
    flex = jg[2] - jg[1];
    control_output(DA_chan[channel],v_cmd[k]);
    printf(" n Fa %6.2f " F_act[k]);
    n = k;
    k = k+1;
} /* End of while loop */

ftime(&timebuffer);
f_time = (double)timebuffer.time*timebuffer.millitm/1000.0;
x = f_time;
dif_time = x-x;
step_time = dif_time/n;
printf(" n Zero output voltage command ").
for (ch = 0; ch = 1; ch++)

```

```

    aot (DA lchan[ch], 0, &erstat),
    fprintf(stream," Test for the Force command   %0.2f\n",F des),
    fprintf(stream," Test time   %0.4f [s] for %0d loops   ", dif time,n),
    fprintf(stream,"nTime of 1 loop   %0.4f [s]", step time),
    for(k=0, k<= n, k++)    * Store test data *
        fprintf(stream1,"n %0.5f %0.6 2f %0.6 2f %0.6 2f ",k*step time,
            F act[k],F err[k],v cmd[k] ),
    printf("\nloop n   %0d\n",n),
    printf("\n\nTest time   %0.4f [s] for %0d loops   ", dif time,n),
    printf("\nTime of 1 loop   %0.4f [s] ", dif time n),
    fclose(stream),
    fclose(stream1),
} /*****END OF THE MAIN CONTROL LOOP *****/

comp_tor(force,l)    * function to compute the require input torque*
float force,l,        * for given contact force and tip orientation */
{
    float mi,alfa,beta,gamma,    *mi friction coefficient ,a,b,g- orientation angles*/
    float Mom,
    mi=1.2,                /*friction coefficient for alluminium*/
    beta= 44.85,           * arccos(280.395) parameters form trajectory geometry*/
    gamma=0,
    alfa=90-(beta+gamma),
    Mom= force*l*(sin(alfa)+ mi*cos(alfa)),
    return(Mom),
}

act_for(Fxforce,Fzforce)    /* Function to calculate the actual contact force*/
float Fxforce,Fzforce,
{
    float F actual,        /*actual force on the manipulator tip*/
    float beta,alpha,mi,
    mi= 61,
    beta=44.85,            *parameters form trajectory geometry*/
    alpha=90-beta,
    F actual= Fxforce/(sin(alpha)+ mi*cos(alpha)),
    return(F actual),
}

ang_jr3()    *function to read analog data from the JR3 sensor */
{
    float Fx conv,Fy conv,Fz conv,
    float const 1, const 2
    float Fx volt,Fy volt,Fz volt,

```

```

float conv,
int AD_ch[3],value m[3],
conv = 4.448 , /*convertes lbs to N */
Fx_conv = 150/10 , /* 150 lbs per 10 input volt */
Fy_conv = 150/10 , /* 150 lbs per 10 input volt */
Fz_conv = 300/10 , /* 300 lbs per 10 input volt */
const_1 = 10.0 / 2047.0 ,
const_2 = 10.0 / 2048.0 ,
AD_ch[0] = 4 ,
AD_ch[1] = 5 ,
AD_ch[2] = 6 ,
value m[0] = ain (AD_lchan[AD_ch[0]], &erstat),
value m[1] = ain (AD_lchan[AD_ch[1]], &erstat),
value m[2] = ain (AD_lchan[AD_ch[2]], &erstat);

if( value m[0] > 0.0)
    Fx_volt = value m[0] * const_1,
else
    Fx_volt = value m[0] * const_2,
if( value m[1] > 0.0)
    Fy_volt = value m[1] * const_1,
else
    Fy_volt = value m[1] * const_2,
if( value m[2] > 0.0)
    Fz_volt = value m[2] * const_1,
else
    Fz_volt = value m[2] * const_2,

Fx_force = Fx_volt * Fx_conv *conv,
Fy_force = Fy_volt * Fy_conv *conv,
Fz_force = Fz_volt * Fz_conv *conv,
return(Fx_force,Fy_force,Fz_force), /* global variables */
}
main_init() /* function to initialize JR sensor and RTI card */
{
    jr3_init(COM1,A_CODE),
    rti_init(),
}

enc_init() /* function to initialize encoder */
{
    int b_stat, c_stat, d_stat, /* counter statuses */
    int j,k,
    char ans,

```

```

int err,                * returned error values      *
struct Output_Status_Reg
{
    unsigned int ignore      11,
    unsigned int upcount flag  1,
    unsigned int underover flag 1,
    unsigned int preset flag  1,
    unsigned int underflow tog 1,
    unsigned int overflow tog  1,
} a_stat,
err = 0,
for (j=1 ; j<= 4 ; j++) /* Initialize 4 Encoders */
{
    init_encoder(axis[j], MCR, ICR, OCCR, QR, ENC_BASE),
}
printf("\n*****  INITIALIZING ENCODERS  ***** "),
load_cnt1(axis[2], preset cnt1, ENC_BASE), /* load counter value */
load_cnt2(axis[3], preset cnt2, ENC_BASE),
load_cnt3(axis[4], preset cnt3, ENC_BASE),

printf("\n For joint 1 preset cnt= %ld \n\n", preset cnt1),
printf("\n For flex encoder preset cnt = %ld \n\n", preset cnt2),
printf("\n For joint 2 preset cnt = %ld \n\n", preset cnt3),
}

rti_init() /* inititalize RTI card */
{
    board=1, /* RTI-815 Board reference number */
    DA_lchan[0] = 0, /* Parameter initialization */
    DA_lchan[1] = 1,
    AD_lchan[0] = 2,
    AD_lchan[1] = 3,
    AD_lchan[2] = 4,
    AD_lchan[3] = 5,
    AD_lchan[4] = 6,
    AD_lchan[5] = 7,
    AD_lchan[6] = 8,
    gain = 1,
    initialize(&erstat), /* D/A & A/D board initialization */
    if(erstat) {
        printf("***** n"),
        printf("** initialization error %d ** n", erstat),
        printf("***** n n"),
        exit(0),
    }
}

```

```

    }
    for (ch = 0; ch < 1; ch++)
    {
        aot815(DA_lchan[ch], board, ch, &erstat);

        if(erstat)
        {
            printf("*****\n");
            printf("*** D/A setup error %2d **\n", erstat);
            printf("*****\n\n");
            exit(0);
        }
    }
    for (ch = 0; ch < 6; ch++)
    {
        ain815(AD_lchan[ch], board, ch, gain, &erstat);
        if(erstat)
        {
            printf("*****\n");
            printf("*** A/D setup error %2d **\n", erstat);
            printf("*****\n\n");
            exit(0);
        }
    }
}

```

jr3_init(port,code) * Initialization of the JR3 force sensor */

```

int port,
unsigned char code,
{
    union RFGS r;
    r.xdx = port;
    r.hah = 0;
    r.hal = code;
    int86(0x14, &r, &r);
}

```

int main menu()

```

{ int mode;
  printf("\n ***** ");
  printf("\n *      PROGRAM PD TEST      * ");
  printf("\n ***** ");
  printf("\n 1 - FORCE CONTROL TEST");
  printf("\n 2 - PD CONTROL - ");
}

```

```

printf("\n CONTROLLED IS THE CONTACT FORCE ON THE LINK END \n"),
retry printf("\n***** \n"),
    printf(" SELECT TEST MODE \n\n"),
    printf(" 1 --> RUN THE TEST \n"),
    printf(" 2 --> EXIT \n"),
    printf(" ENTER # --> "),
    scanf("%d", &mode),
    if ( !((mode==1)|| (mode==2)))
        goto retry,
    printf("***** \n\n"),
    return(mode),
}

start() /*Subroutine to start the test */
{
    char ans,
    again printf("\n PRESS 's' TO START JOINT MOTION"),
        printf("\n PRESS 'e' TO EXIT --> "),
        scanf("%s",&ans),
        if (ans == 's' || ans == 'e')
            {if (ans == 'e')
                exit(0),
                }
            else goto again,
}

jr3_get_pd(port, processed_data_ptr) /*Read data from the serial port */
int port,
struct jr3_processed_data *processed_data_ptr,
{
    int bytes_read,
    char input_buffer[BUF_SIZE],

    jr3_write(port, "DP S r", 5), /* DP S is a request for processed data */
    bytes_read = jr3_read(port, input_buffer, 81),
    input_buffer[bytes_read] = 0,
    sscanf(input_buffer+4, " Fx %lf Fy %lf Fz %lf Mx %lf My %lf Mz %lf" ,
        &(processed_data_ptr->Fx), &(processed_data_ptr->Fy), &(processed_data_ptr->Fz),
        &(processed_data_ptr->Mx), &(processed_data_ptr->My), &(processed_data_ptr->Mz))
}

int port_putch(port,c) /* Procedure to write character to port */
int port,
char c,

```



```

{
    union REGS r;
    r.x.dx = port;
    r.h.al = c;
    r.h.ah = 1;
    int86(0x14, &r, &r);

    if(r.h.ah & 128) /* check bit 7 */
        return(-1);
    else
        return(0);
}

```

int jr3_write(port, buf, cnt) /* Procedure to write command to JR3 sensor */

```

int port;
char *buf;
int cnt;
{
    char *byte_ptr;
    int loop;
    byte_ptr = buf;
    for(loop = 0; loop < cnt; loop++)
    {
        if(*byte_ptr == '\n')
            *byte_ptr = '\r';
        if(port_putchar(port, (*byte_ptr)))
            return(loop);
        byte_ptr++;
    }
    return(cnt);
}

```

int jr3_read(port, buf, cnt) /* Procedure to read RS 232 port

```

int port;
char *buf;
int cnt;
{
    union REGS r;
    char *byte_ptr;
    int time = cnt;
    byte_ptr = buf;
    for(loop = 0; loop < cnt; loop++)
    {
        r.x.dx = port;

```

```

        r_h ah, 2,
        int8o(0x14,&r,&r),
        *byte_ptr++ = r_h ah,
    }
    return(cnt),
}

```

control_output (logchan, v_out) *Function to limit and generate controller output *

```

int    logchan,
float  v_out,
{
    int    int_v_out,
    float  v_scaled,
    if (v_out > v_limit)
        v_out = v_limit,
    if (v_out < -v_limit)
        v_out = -v_limit,
    v_scaled = 2048/0.10/0 * v_out, /* Digitizing the control voltage */
    int_v_out = (int) v_scaled,
    if ( v_scaled > 0.0) /* Account for chop-off error */
    {
        if (( v_scaled - int_v_out) > 0.5)
            int_v_out += 1,
    }
    else {
        if (( v_scaled - int_v_out) < -0.5)
            int_v_out -= 1,
    }
    aot(logchan, int_v_out, &erstat), /* D/A conversion */
} /******END OF PD CONTROL *****/

```

APPENDIX III/II

-Real Time Software-

PID force control

*Given below function "test" should be substituted into the the PD control program */

/*while loop should be substituted instead of while loop in PD control */

```

test()
{
    /*Data declaration is the same as in PD procedure*/
    float K_f,
    float int_err, del_t,
    float err_temp,
    float v_cmd_temp,

```

```

printf(" n = 10) ENTER del t = : ").
scanf("%d",&del_t);
err[k] = F_des;
n = 1;
while('kbhit() && n = N ) ;

    ang_jr3();
    F_act[k] = act_for(Fx_force,Fz_force);
    err[k] = F_des-F_act[k];
    int_err = int_err+del_t*err[k];
    v_cmd[k] = K_p*err[k] + K_d*(err[k]-err0)+K_i*int_err;
    if (fabs(v_cmd[k]) > 2)
        if(v_cmd[k] > 2)
            v_cmd[k] = 2;
        else v_cmd[k] = -2;
    err0 = err[k];
    control_output(DA_1chan[channel],v_cmd[k]);
    * printf(" n = F_act %6.2f F_err %6.2f", F_act[k], err[k]).*/
if(k == 10)
{
    for(j = 0, j = k, j++)
        fprintf(stream1," n %5f %6.2f %6.2f %6.2f ",n*del_t,
            F_act[j],err[j],v_cmd[j] );
    Fact = F_act[k];
    err_temp = err[k];
    v_cmd_temp = v_cmd[k];
    k = 0;
    F_act[k] = Fact;
    err[k] = err_temp;
    v_cmd[k] = v_cmd_temp;
}
k = k+1;
n = n+1;
}
*****~END OF PID CONTROL *****/

```

APPENDIX IV

System Parameter Modelling

3 of/de 3

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

APPENDIX IV/I

-System parameter modelling-

MODELLING OF THE VXA 48-8-16 AMPLIFIER IN TORQUE MODE

To assure a linear relationship between input commands and amplifier output current, it was necessary to model the amplifier in torque mode. The amplifier voltage current constant K_a is not a constant parameter, thus for the control purposes it is recommended to use the linear part of its characteristic. The found approximate $K_a = 1.76$

A/Volt. Table AIV/I1 presents one of several calibrations performed on the amplifier. In Fig. AIV/I1 is shown the corresponding data plot.

Table AIV/I1 VXA 48-8-16 Amplifier modelling in torque mode

INPUT VOLTAGE [V]	CURRENT [A]	K_a Amp/Volt
25	469	1.876
50	898	1.796
75	1.327	1.769
1.0	1.751	1.751
1.25	2.178	1.74
1.5	2.603	1.735
1.75	3.03	1.731
2.0	3.465	1.732

Fig. AIV/I1 VXA 48-8-16 Amplifier characteristic K_a - amplifier constant

APPENDIX IV/II

-System parameter modelling-

CONTROL INPUT VOLTAGE - OUTPUT TORQUE CHARACTERISTIC OF THE JOINT1- Static conditions.

Voltage commands are sent to the motor. The output torque is measured using the JR3 force sensor. The data is compared with the calculated joint torque. The nonlinearity of the output torque caused by the joint friction can be seen in the presented graph. See Table AIV/II-1 and Fig. AIV/II-1.

Table AIV/II-1 Joint 1 output torque characteristic Static conditions

input command [V]	motor current [A]	TPI control voltage [V]	output torque [lb-in]	computed torque [lb-in]
1.0	53	21	34	13.95
1.5	83	45	14.3	25.46
2.0	1.08	6	17.3	36.97
2.5	1.34	7	23.1	48.48
3.0	1.6	88	33.0	59.98
3.5	1.85	1.03	46.65	71.49
4.0	2.1	1.17	62.40	83.00
4.5	2.36	1.31	82.20	94.51
5.0	2.61	1.45	128.50	106.0
5.5	2.87	1.6	137.40	117.52
6.0	3.13	1.74	162.0	129.03
6.5	3.21	1.78	172.2	140.54
7.0	3.27	1.81	176.55	122.05

CONTROL INPUT VOLTAGE - OUTPUT TORQUE CHARACTERISTIC OF THE JOINT 1- Static conditions

Fig AIV/III/1 Joint 1 control voltage-output torque characteristic Force feedback used to measure the output torque

APPENDIX IV/III

-System parameter modelling-

ESTIMATING OF THE STATIC JOINT FRICTION CONSTANT

The increasing gradually current supplied is to the motor. Motor current measured is by the current measuring circuit. At the transition moment, from the static to the joint motion, the maximum motor current i_m is measured. After multiplication by the motor torque constant, the joint static friction is obtained, including motor and bearings friction. This parameter varies and depends on the actual joint angular displacement. The found average joint static friction torque $F_m = 0.325$ [Nm] (Table IV/III/1).

Table IV/III/1 Joint static friction estimating

Motor current [A]. Average motor current $i_a = 1.04$ [A]. $F_m = i_a \cdot K_m$										
1.07	1.08	1.03	1.43	1.1	1.23	1.17	95	72	1.2	78

APPENDIX IV/IV

-System parameter modelling-

JOINT 1 FLEXIBILITY TEST This test was performed several times to establish a relationship between the joint output torque and joint flexibility. To show the varying joint flexibility the relationship between the controller voltage and the joint internal deflection is presented, caused by elasticity of joint components. Displacement is measured by two encoders, one on the input and one on the output from the joint. The difference between these readings is the actual internal joint deflection (see Fig AIV/IV/1).

Fig AIV/IV/1 Flexibility of Joint 1. Controller input voltage/joint deflection characteristic. Link is constrained by a test surface. Static test.

APPENDIX IV/V

-System parameter modelling-

CALIBRATION OF THE JR3 FORCE/TORQUE SENSOR

The JR3 sensor is initially factory calibrated. The result of this calibration is a 6x6 calibration matrix representing the output of all six channels, when loads are applied to the sensor. Each of channels is equipped with a fourth order Butterworth low pass filter. The disabling of digital filters improves the data acquisition speed, at the same time enhances the chance for the undesirable noise receiving. For the force control test purpose it was necessary to perform the sensor calibration. The calibration scheme is shown on Fig AIV/V/1. Two techniques of sensor data reading are used: *analog data reading* and *serial port communication*. The first method is based on the reading of the output voltage

from the sensor, digitizing by RT1815 A/D converter and decoding using custom written software. The second method uses the RS 232 serial port communication with the sensor. Adequate software commands given by the sensor manufacturer are implemented in the custom written program. Force sensor data can be stored in a file and monitored.

Fig. AIV/V/I Calibration and communication with the JR3 sensor

Following is the description of calibration tests performed on the JR3 sensor

A> Test for small loads, sensor not attached to the manipulator link

Sensor positioned on the even test surface. Data retrieving is accomplished by means of 2.0 UFC software. Single set of data for Fz axis was requested by sending software command "DP S". The average discrepancy between actual applied loads and results obtained by the computer is 1.5%. See Tab. AIV/V/I for sample results.

B> A not loaded sensor test, attached to the link end

Several tests have been done for the sensor, requesting single sets of data ("DP S" command). The "zero offset" (i.e. tare weight or preload removal) can be accomplished by software commands or by adjusting trimpots on the signal conditioning board. The "zero offset" can obtain various values under similar conditions and is influenced by the resolution of the analog-to-digital converter reading the sensor analog data (See Tab. AIV/V/II). The purpose of this calibration was to assure the same initial conditions (zero force offset) for each control test. Practically, it is not possible to reduce the "zero offset".

for all sensor axis to zero, due to the technical limitations. During experiments this parameter was adjusted by issuing the "RO \r" command to the sensor via computer serial port RS 232

Table AIV/V/I JR3 sensor calibration.

Fz applied load [lb]	Fz measured load [lb]
1 049	1.06
2 034	2.09
2 992	2.97
3 5	3.55
5 0	4.91
10 0	9.92

Table AIV/V/II JR3 sensor calibration - "zero offset" for all sensor axis.

No	Fx	Fy	Fz	Mx	My	Mz
	lb			lb-in		
1	.08	.08	.13	-.26	-.15	-.04
2	.01	.08	-.02	-.24	-.22	-.08
3	.08	.01	-.16	-.40	-.04	-.08
4	.04	.08	-.16	.15	-.01	-.04

END

0 9-1 1-9 4

FIN