



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE141599

**PERANCANGAN PENGATURAN POSISI ROBOT
MANIPULATOR BERBASIS PD *FUZZY MAMDANI*
*COMPUTED TORQUE CONTROL (PD FUZZY CTC)***

Duli Ridlo Istrianono
NRP 2212106087

Dosen Pembimbing
Dr. Trihastuti Agustinah, S.T., M.T.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015



FINAL PROJECT - TE 141599

**CONTROL DESIGN OF ROBOT MANIPULATOR
POSITION BASED ON PD-FUZZY MAMDANI
CONTROLLED WITH COMPUTED TORQUE CONTROL
(PD-FUZZY-CTC)**

Duli Ridlo Istrianono
NRP 2212106087

Advisor
Dr. Trihastuti Agustinah, S.T., M.T.

ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2015

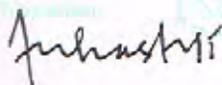
**PERANCANGAN PENGATURAN POSISI
ROBOT MANIPULATOR BERBASIS PD FUZZY MAMDANI
COMPUTED TORQUE CONTROL (PD FUZZY CTC)**

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Teknik Sistem Pengaturan
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing.



Dr. Trihastuti Agustinah, S.T., M.T.
NIP.1968 08 12 1994 03 2001



PERANCANGAN PENGATURAN POSISI ROBOT MANIPULATOR BERBASIS PD FUZZY MAMDANI COMPUTED TORQUE CONTROL (PD FUZZY CTC)

Nama : Duli Ridlo Istriantono
Pembimbing : Dr. Trihastuti Agustinah, S.T., M.T.

ABSTRAK

Ilmu robotika telah banyak berkembang, didorong oleh kemajuan pesat dalam komputer dan teknologi sensor serta kemajuan teoritis dalam kontrol dan *computer vision*. Perkembangan ini membuat maraknya penggunaan robot manipulator dalam lingkungan industri. Penggunaan robot manipulator tidak hanya berdampak pada penurunan biaya tenaga kerja, peningkatan presisi dan produktivitas, peningkatan fleksibilitas, tetapi juga dapat mengatasi kondisi kerja yang membosankan, berulang-ulang, atau pekerjaan berbahaya.

Masalah utama dalam mengendalikan manipulator robot adalah untuk mengontrol robot untuk mencapai posisi yang diinginkan. Oleh karena itu masalah desain kontrol robot adalah untuk memilih jenis kontroler yang tepat. *Computed Torque Controller* (CTC) merupakan kontroler nonlinier yang handal yang secara luas digunakan dalam pengendalian robot manipulator. Kontroler CTC didesain berdasarkan pada linierisasi umpan balik dan penghitungan besar torsi lengan robot yang diperlukan dengan menggunakan hukum kontrol *feedback* nonlinier. Kontroler ini menggunakan analisis gerak dinamik sehingga sangat memperhatikan energi dan gaya-gaya yang terjadi pada pergerakan robot manipulator.

Simulasi dilakukan dengan memberikan trayektori *joint* dari satu titik ke titik yang lain. Hasil simulasi menunjukkan bahwa kontroler PD *Fuzzy* CTC mampu mengikuti trayektori *joint* dengan nilai RMSE posisi sudut *joint* dari kontroler PD *Fuzzy* CTC adalah 10 kali lebih kecil dibandingkan dengan kontroler PD CTC dengan ketelitian posisi *end-effector* dengan ketelitian mencapai 0,1 mm.

Kata Kunci: Dinamik, PD *Fuzzy* CTC, Robot Manipulator, Trayektori



CONTROL DESIGN OF ROBOT MANIPULATOR POSITION BASED ON PD-FUZZY MAMDANI CONTROLLED WITH COMPUTED TORQUE CONTROL (PD-FUZZY-CTC)

Name : Duli Ridlo Istrianono
Supervisor : Dr. Trihastuti Agustinah, S.T., M.T.

ABSTRACT

Robotics science has evolved significantly, driven by rapid advances in computer and sensor technology; and theoretical advances in control and computer vision. These development make widespread use of robot manipulators in industrial environments. Among the advantages of using robots is the impact on the reduction of labor costs, improving precision and productivity, increase flexibility, and address the working conditions such as boring, repetitive, or hazardous work.

Major problem in controlling a robot manipulator is to control the robot in order to achieve the desired position. Therefore the design issue of the robot control is to choose the right type controller. Computed Torque Controller (CTC) is a powerful nonlinear controllers are widely used in the control of robot manipulators. CTC controller is designed based on feedback linearization and the required torque of the robot arm by using a nonlinear feedback control law. This controller uses dynamic motion analysis, so that it is very concern about the energy and the forces in the movement of the robot manipulator.

Simulation is done by providing joint trajectory from point to point. The simulation results show that the PD-Fuzzy-CTC controller is able to follow the joint trajectory with The RMSE value of the joint angle position of PD-Fuzzy-CTC controller is 10 times smaller than that of the PD-CTC controller with the end-effector position accuracy is 0.1 mm.

Keywords: *Dynamic, PD-Fuzzy-CTC, Robot Manipulator, Trajectory*



KATA PENGANTAR

Puji syukur atas kehadiran Allah subhanahu wa ta'ala atas segala limpahan rahmat, taufik, hidayah dan inayah-Nya sehingga Tugas Akhir ini dapat diselesaikan dengan baik. Shalawat serta salam semoga selalu tercurahkan kepada Rasulullah Muhammad shallallaahu 'alaihi wa sallam, juga kepada keluarga beliau, para sahabatnya yang memberikan teladan bagi penulis.

Tugas Akhir ini disusun guna memenuhi sebagian persyaratan menyelesaikan pendidikan Strata-1 pada Bidang Studi Teknik Sistem Pengaturan, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya, dengan judul:

PERANCANGAN PENGATURAN POSISI ROBOT MANIPULATOR BERBASIS PD *FUZZY MAMDANI COMPUTED TORQUE CONTROL (PD FUZZY CTC)*

Penulis menyadari bahwa dalam penyelesaian pengerjaan Tugas Akhir ini tidak terlepas dari bantuan dan dukungan dari berbagai pihak. Untuk itu dengan segala hormat, penulis mengucapkan terima kasih kepada semua pihak yang telah berperan serta dalam penyelesaian Tugas Akhir ini, khususnya kepada dosen pembimbing, Ibu Dr. Trihastuti Agustinah, S.T., M.T. yang selalu sabar membimbing dan memberikan kepercayaan dan motivasi, kedua orang tua yang senantiasa memberikan dukungan baik secara materiil maupun nonmateriil, keluarga, teman-teman LJ 2012, teman-teman asisten Lab. B105, dan teman-teman lainnya yang senantiasa memberikan dukungan penuh kepada penulis, serta kepada semua pihak yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari bahwa masih banyak kekurangan dalam penelitian maupun dalam penulisan buku ini. Untuk itu, penulis memohon maaf atas segala kekurangan yang ada. Semoga Tugas Akhir ini dapat bermanfaat untuk pengembangan ilmu pengetahuan dan teknologi selanjutnya.

Surabaya, Januari 2015

Penulis



DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAN TUGAS AKHIR	iii
HALAMAN PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Metodologi Penelitian	2
1.6 Sistematika Penulisan	3
1.7 Relevansi	4
BAB II TEORI PENUNJANG	5
2.1 Tinjauan Pustaka	5
2.2 Robot Manipulator	5
2.2.1 Bagian-bagian Robot Manipulator	5
2.2.2 Komponen dan Struktur Robot	6
2.2.3 Konfigurasi Robot	7
2.2.4 Gerakan Kaku dan Transformasi Homogen	9
2.3 Kinematika Robot	11
2.3.1 <i>Forward Kinematics</i>	12
2.3.2 <i>Inverse Kinematics</i>	14
2.4 Dinamik Robot Manipulator	15
2.4.1 Perumusan <i>Lagrange</i>	15
2.4.2 Perhitungan Energi Kinetik	16
2.4.3 Perhitungan Energi Potensial	17
2.4.4 Persamaan Gerakan	18
2.5 Kontroler	18
2.5.1 Kontroler <i>Computed Torque Control (CTC)</i>	19

2.5.2	Kontroler PID.....	20
2.5.3	Kontroler <i>Fuzzy</i>	22
2.5.3.1	Himpunan <i>Fuzzy</i>	23
2.5.3.2	Operasi Himpunan <i>Fuzzy</i>	26
2.5.3.3	Fuzzifikasi.....	28
2.5.3.4	Defuzzifikasi.....	29
2.5.3.5	Sistem Inferensi <i>Fuzzy</i>	32
BAB III PERANCANGAN SISTEM.....		35
3.1	Perancangan Blok Diagram Kontrol.....	35
3.2	Perancangan Parameter <i>Denavit-Hartenberg</i>	35
3.3	Perancangan Kinematik.....	38
3.4	Perancangan Model Dinamik.....	42
3.5	Perancangan Trayektori untuk Gerakan <i>Point to Point</i> ...	46
3.6	Perancangan Kontroler.....	48
3.6.1	Desain Kontroler PD CTC.....	48
3.6.2	Desain Kontroler PD <i>Fuzzy</i> CTC.....	50
3.7	Perancangan Sistem pada Diagram Simulink.....	54
BAB IV PENGUJIAN DAN ANALISIS SISTEM		57
4.1	Pengujian Posisi Sudut Setiap <i>Joint</i>	57
4.2	Pengujian Posisi <i>End-Effector</i>	64
4.3	Analisis Sistem dengan Kontroler PD CTC.....	65
4.4	Analisis Sistem dengan Kontroler PD <i>Fuzzy</i> CTC.....	71
BAB V PENUTUP.....		77
5.1	Kesimpulan.....	77
5.2	Saran.....	77
DAFTAR PUSTAKA		79
LAMPIRAN A.....		81
LAMPIRAN B.....		93
RIWAYAT PENULIS.....		103

DAFTAR TABEL

Tabel 3.1 Parameter <i>Link</i> untuk Robot Manipulator PUMA-560.....	37
Tabel 3.2 Parameter <i>Link</i> untuk Robot Manipulator <i>Two-Link</i>	38
Tabel 3.3 Parameter <i>Link</i> untuk Robot Manipulator <i>Articulated</i>	38
Tabel 3.4 Konstanta Inersia	46
Tabel 3.5 Konstanta Gravitasi	46
Tabel 3.6 Metode <i>Ziegler-Nichols</i>	49
Tabel 3.7 Aturan Dasar <i>Mack Vicar Whelan</i> untuk Kontroler <i>Fuzzy</i> ...	53
Tabel 4.1 Hasil Pengujian Posisi Sudut <i>Joint 1, Joint 2</i> dan <i>Joint 3</i>	57
Tabel 4.2 Hasil Pengujian Posisi Sudut <i>Joint 4, Joint 5</i> dan <i>Joint 6</i>	57
Tabel 4.3 Nilai RMSE tanpa Kontroler	63
Tabel 4.4 Koordinat Posisi <i>End-effector</i> Robot Manipulator	64
Tabel 4.5 Data Hasil <i>Tuning Gain</i> kontroler PD	66
Tabel 4.6 Nilai RMSE Kontroler PD CTC.....	70
Tabel 4.7 Koordinat Posisi <i>End-effector</i> Robot Manipulator (PD CTC)	71
Tabel 4.8 Nilai RMSE Kontroler PD <i>Fuzzy</i> CTC	75
Tabel 4.9 Koordinat Posisi <i>End-effector</i> Robot Manipulator (PD <i>Fuzzy</i> CTC)	76

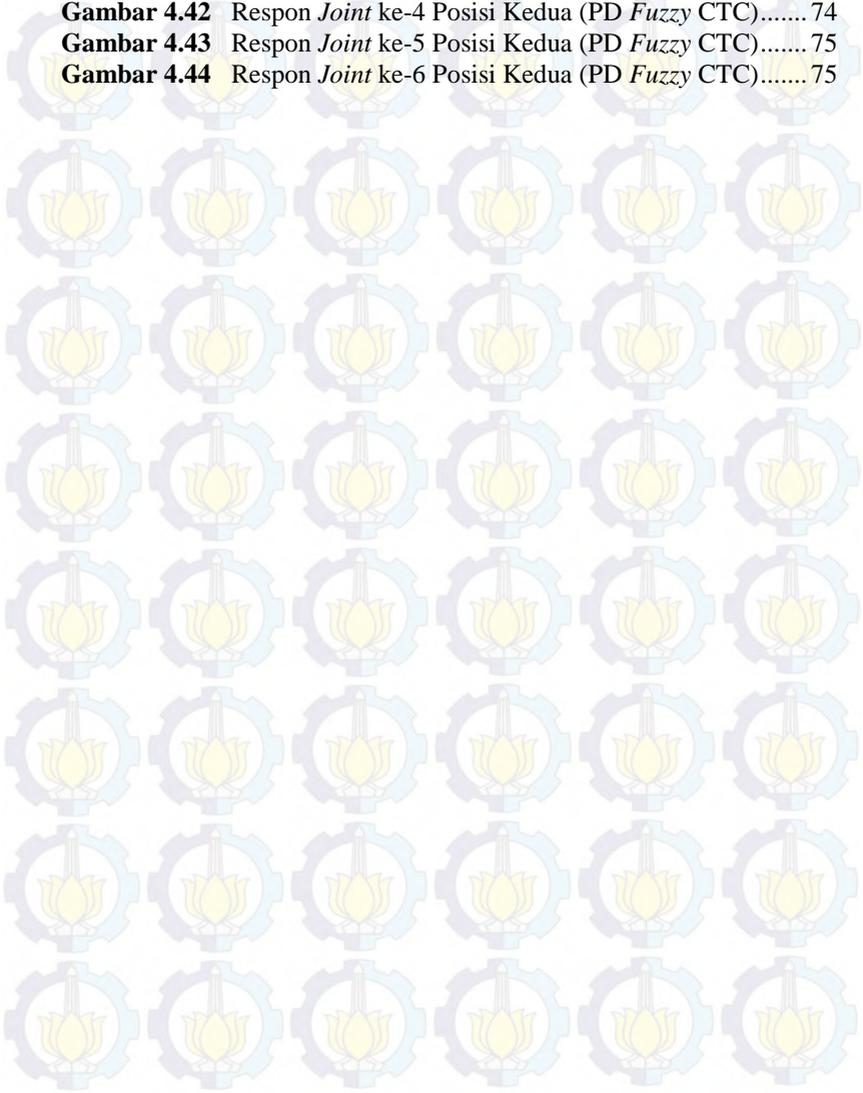


DAFTAR GAMBAR

Gambar 2.1	Bagian-bagian Manipulator	5
Gambar 2.2	Representasi Sendi Robot	6
Gambar 2.3	Konfigurasi <i>Articulated</i> (RRR).....	7
Gambar 2.4	Konfigurasi <i>Spherical</i> (RRP).....	7
Gambar 2.5	Konfigurasi SCARA (RRP).....	8
Gambar 2.6	Konfigurasi <i>Cylindrical</i> (RPP)	8
Gambar 2.7	Konfigurasi <i>Cartesian</i> (PPP)	8
Gambar 2.8	Sistem Koordinat Tangan dan n, s, a, p	11
Gambar 2.9	Representasi <i>Forward dan Inverse Kinematics</i>	11
Gambar 2.10	Penetapan <i>Frame Denavit-Hartenberg</i>	13
Gambar 2.11	Diagram Blok Kontroler PID.....	21
Gambar 2.12	Diagram Blok Kontroler <i>Fuzzy</i>	22
Gambar 2.13	Fungsi Keanggotaan Tinggi Badan.....	23
Gambar 2.14	Fungsi Keanggotaan Tinggi Badan Asia, Eropa, NBA	24
Gambar 2.15	Tipe Fungsi Keanggotaan <i>Fuzzy</i>	25
Gambar 2.16	<i>Plot</i> Fungsi Keanggotaan Himpunan <i>Fuzzy</i> A dan B ..	27
Gambar 2.17	Operasi Himpunan <i>Fuzzy</i>	28
Gambar 2.18	Struktur Fungsi Keanggotaan <i>Fuzzy</i>	29
Gambar 2.19	Defuzzifikasi <i>Center of Area</i>	30
Gambar 2.20	Himpunan <i>Fuzzy</i> Tidak Saling Tumpang Tindih	31
Gambar 2.21	Defuzzifikasi <i>First of Maxima dan Last of Maxima</i>	32
Gambar 2.22	Defuzzifikasi <i>Middle of Maxima</i>	32
Gambar 2.23	Inferensi <i>Fuzzy Mamdani</i>	34
Gambar 3.1	Blok Diagram Sistem Kontrol Robot Manipulator	35
Gambar 3.2	Robot Manipulator PUMA-560	36
Gambar 3.3	Penetapan <i>End-Effector Frame</i> pada $o_n x_n y_n z_n$	37
Gambar 3.4	Bentuk Geometris Robot <i>Two-Link Planar</i>	42
Gambar 3.5	Bentuk Geometris Robot <i>Articulated Elbow</i>	44
Gambar 3.6	Lintasan Trayektori Kubik.....	47
Gambar 3.7	Struktur Kontroler PD CTC.....	49
Gambar 3.8	Struktur Kontroler PD <i>Fuzzy</i> CTC.....	50
Gambar 3.9	Fungsi Keanggotaan <i>Input</i> berupa Sinyal <i>Error</i>	51
Gambar 3.10	Fungsi Keanggotaan <i>Input</i> berupa Sinyal <i>dError</i>	52
Gambar 3.11	Fungsi Keanggotaan untuk <i>Output</i>	52
Gambar 3.12	Diagram Simulink Sistem Kontrol Robot Manipulator	55

Gambar 4.1	Respon Posisi <i>Set Point</i> Pertama	58
Gambar 4.2	Respon Posisi Aktual Pertama	58
Gambar 4.3	Respon Posisi <i>Set Point</i> Kedua	58
Gambar 4.4	Respon Posisi Aktual Kedua	59
Gambar 4.5	Respon <i>Joint</i> ke-1 Posisi Pertama	59
Gambar 4.6	Respon <i>Joint</i> ke-2 Posisi Pertama	60
Gambar 4.7	Respon <i>Joint</i> ke-3 Posisi Pertama	60
Gambar 4.8	Respon <i>Joint</i> ke-4 Posisi Pertama	60
Gambar 4.9	Respon <i>Joint</i> ke-5 Posisi Pertama	61
Gambar 4.10	Respon <i>Joint</i> ke-6 Posisi Pertama	61
Gambar 4.11	Respon <i>Joint</i> ke-1 Posisi Kedua	61
Gambar 4.12	Respon <i>Joint</i> ke-2 Posisi Kedua	62
Gambar 4.13	Respon <i>Joint</i> ke-3 Posisi Kedua	62
Gambar 4.14	Respon <i>Joint</i> ke-4 Posisi Kedua	62
Gambar 4.15	Respon <i>Joint</i> ke-5 Posisi Kedua	63
Gambar 4.16	Respon <i>Joint</i> ke-6 Posisi Kedua	63
Gambar 4.17	Posisi Awal Robot Manipulator	64
Gambar 4.18	Posisi Akhir Robot Manipulator Pertama	65
Gambar 4.19	Posisi Akhir Robot Manipulator Kedua	65
Gambar 4.20	Respon Output <i>Tuning</i> Kontroler PD	66
Gambar 4.21	Respon <i>Joint</i> ke-1 Posisi Pertama (PD CTC)	66
Gambar 4.22	Respon <i>Joint</i> ke-2 Posisi Pertama (PD CTC)	67
Gambar 4.23	Respon <i>Joint</i> ke-3 Posisi Pertama (PD CTC)	67
Gambar 4.24	Respon <i>Joint</i> ke-4 Posisi Pertama (PD CTC)	67
Gambar 4.25	Respon <i>Joint</i> ke-5 Posisi Pertama (PD CTC)	68
Gambar 4.26	Respon <i>Joint</i> ke-6 Posisi Pertama (PD CTC)	68
Gambar 4.27	Respon <i>Joint</i> ke-1 Posisi Kedua (PD CTC)	68
Gambar 4.28	Respon <i>Joint</i> ke-2 Posisi Kedua (PD CTC)	69
Gambar 4.29	Respon <i>Joint</i> ke-3 Posisi Kedua (PD CTC)	69
Gambar 4.30	Respon <i>Joint</i> ke-4 Posisi Kedua (PD CTC)	69
Gambar 4.31	Respon <i>Joint</i> ke-5 Posisi Kedua (PD CTC)	70
Gambar 4.32	Respon <i>Joint</i> ke-6 Posisi Kedua (PD CTC)	70
Gambar 4.33	Respon <i>Joint</i> ke-1 Posisi Pertama (PD <i>Fuzzy</i> CTC)	71
Gambar 4.34	Respon <i>Joint</i> ke-2 Posisi Pertama (PD <i>Fuzzy</i> CTC)	72
Gambar 4.35	Respon <i>Joint</i> ke-3 Posisi Pertama (PD <i>Fuzzy</i> CTC)	72
Gambar 4.36	Respon <i>Joint</i> ke-4 Posisi Pertama (PD <i>Fuzzy</i> CTC)	72
Gambar 4.37	Respon <i>Joint</i> ke-5 Posisi Pertama (PD <i>Fuzzy</i> CTC)	73
Gambar 4.38	Respon <i>Joint</i> ke-6 Posisi Pertama (PD <i>Fuzzy</i> CTC)	73
Gambar 4.39	Respon <i>Joint</i> ke-1 Posisi Kedua (PD <i>Fuzzy</i> CTC)	73

Gambar 4.40 Respon *Joint* ke-2 Posisi Kedua (PD *Fuzzy* CTC)..... 74
Gambar 4.41 Respon *Joint* ke-3 Posisi Kedua (PD *Fuzzy* CTC)..... 74
Gambar 4.42 Respon *Joint* ke-4 Posisi Kedua (PD *Fuzzy* CTC)..... 74
Gambar 4.43 Respon *Joint* ke-5 Posisi Kedua (PD *Fuzzy* CTC)..... 75
Gambar 4.44 Respon *Joint* ke-6 Posisi Kedua (PD *Fuzzy* CTC)..... 75







BAB I

PENDAHULUAN

Bab ini menjelaskan tentang latar belakang, perumusan masalah, batasan masalah dan tujuan dari penelitian pada Tugas Akhir ini. Selain itu, dijelaskan juga mengenai metodologi dan sistematika dalam penyelesaian Tugas Akhir, serta relevansi untuk penelitian selanjutnya.

1.1 Latar Belakang

Ilmu robotika telah banyak berkembang, didorong oleh kemajuan pesat dalam komputer dan teknologi sensor serta kemajuan teoritis dalam kontrol dan *computer vision*. Perkembangan ini membuat maraknya penggunaan robot manipulator dalam lingkungan industri. Di antara keuntungan menggunakan robot manipulator adalah berdampak pada penurunan biaya tenaga kerja, meningkatkan presisi dan produktivitas, meningkatkan fleksibilitas, dan mengatasi kondisi kerja seperti membosankan, berulang-ulang, atau pekerjaan berbahaya [1].

Permasalahan yang sering terjadi adalah kendali posisi robot manipulator sesuai dengan referensi yang diinginkan. Dalam kinematik, pergerakan robot tidak terlalu bermasalah di mana gaya-gaya yang terjadi pada robot tidak diperhatikan. Akan tetapi jika pembahasannya dalam dinamik, gaya-gaya yang menyebabkan gerakan harus diperhatikan.

Pengaturan posisi yang sering menjadi kendala pada robot perlu ditingkatkan agar lebih baik. Untuk itu perlu dilakukan pemilihan jenis kontroler yang tepat, yang bisa digunakan untuk mengendalikan gerak robot manipulator. *Computed Torque Controller* (CTC) merupakan kontroler nonlinier yang handal yang secara luas digunakan dalam pengendalian robot manipulator. Kontroler CTC berdasarkan pada linierisasi umpan balik dan menghitung besar torsi lengan robot yang diperlukan dengan menggunakan hukum kontrol *feedback* nonlinier. Kontroler ini bekerja sangat baik ketika semua parameter fisik dan dinamik diketahui, tetapi ketika robot manipulator memiliki variasi dalam parameter dinamik, dalam situasi seperti ini kontroler tidak mampu memberikan kinerja yang diinginkan [2]. Untuk mendapatkan kinerja sistem robot manipulator yang sesuai, agar setiap *joint* dapat mengikuti trayektori posisi yang diinginkan, pada Tugas Akhir ini dirancang kontroler PD *Fuzzy* CTC yang digunakan untuk mengoreksi kekurangan respon yang ada pada kontroler PD CTC.

1.2 Rumusan Masalah

Permasalahan yang menjadi dasar dalam Tugas Akhir ini adalah kendali posisi robot manipulator dengan menggunakan dinamik di mana gaya-gaya yang menyebabkan terjadinya gerakan pada robot manipulator harus diperhatikan, berbeda dengan kinematik di mana gaya-gaya yang menyebabkan gerakan tidak perlu diperhatikan. Dengan memberikan trayektori pada tiap-tiap *joint* pada robot manipulator, diharapkan robot manipulator dapat diaplikasikan untuk pekerjaan-pekerjaan tertentu dengan memberikan kondisi-kondisi berupa posisi, kecepatan dan percepatan yang telah ditentukan yang memiliki tingkat akurasi posisi *joint* dan *end-effector* yang baik.

1.3 Batasan Masalah

Dalam penelitian Tugas Akhir ini masalah dibatasi pada pengaturan posisi *joint* robot manipulator PUMA-560 dengan trayektori kubik. Selanjutnya untuk mendapatkan respon yang diinginkan masalah juga dibatasi pada perancangan dinamik dari robot manipulator yang memiliki parameter-parameter dinamik tertentu. Juga perancangan kontroler *Fuzzy* yang pengaturannya bergantung dari nilai parameter yang didapatkan dari kontroler PD CTC.

Dalam Tugas Akhir ini perancangan kontroler dan robot manipulator dilakukan dengan menggunakan bantuan *software* Matlab dan *robotics toolbox* Peter Corke.

1.4 Tujuan Penelitian

Penelitian pada Tugas Akhir ini mengarah pada perancangan kontroler PD *Fuzzy* CTC untuk mendapatkan posisi *joint* dan *end-effector* robot manipulator sesuai dengan *set point* berdasarkan trayektori yang diinginkan.

1.5 Metodologi Penelitian

Tahap penyelesaian atau metodologi yang dilakukan pada Tugas Akhir ini, yaitu:

a. Studi Literatur

Pada tahap ini dilakukan kajian teori yang berkaitan dengan gerak kinematik dan pemodelan dinamik robot manipulator, perancangan kontroler PD *Fuzzy* CTC dari jurnal ilmiah, *paper* maupun dari buku referensi.

b. Pemodelan Sistem

Model matematika sistem diperoleh melalui metode pemodelan fisik dengan memperhatikan energi-energi dan gaya-gaya yang bekerja pada robot manipulator, hasil tersebut kemudian dibentuk dalam persamaan *state-space* yang merepresentasikan sistem sebenarnya.

c. Desain Kontroler

Pada tahap ini dilakukan perancangan kontroler PD *Fuzzy* CTC untuk mendapatkan respon yang lebih baik dibandingkan menggunakan kontroler PD CTC.

d. Perancangan Simulasi

Hasil perancangan sistem kontrol robot manipulator selanjutnya dibuat dengan bantuan *software* Matlab dan *toolbox* untuk robot dari Peter Corke.

e. Penulisan Buku

Tahap penyusunan buku sebagai laporan dari Tugas Akhir, meliputi pendahuluan, teori penunjang, perancangan sistem, pengujian dan analisis sistem, serta kesimpulan sebagai penutup.

1.6 Sistematika Penulisan

Pembahasan pada Tugas Akhir ini dibagi menjadi lima Bab dengan sistematika penulisan sebagai berikut

Bab I : Pendahuluan

Bab ini menjelaskan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, sistematika penulisan, serta relevansi.

Bab II : Teori Penunjang

Bab ini menjelaskan mengenai tinjauan pustaka, teori-teori tentang robot manipulator seperti gerak kinematik dan dinamik, pemodelan sistem, serta teori-teori yang berhubungan dengan kontroler PD *Fuzzy* CTC.

Bab III : Perancangan Sistem

Bagian ini berisi pembahasan mengenai perancangan kontrol robot manipulator berdasarkan teori dasar pada Bab II.

Bab IV : Pengujian dan Analisis Sistem

Bab ini memuat hasil pengujian dan analisis data yang didapatkan dari perancangan pada Bab III.

Bab V : Penutup

Analisis yang dilakukan pada Bab IV akan diambil sebagai suatu kesimpulan. Saran diberikan sebagai bahan evaluasi dan pengembangan untuk penelitian selanjutnya.

1.7 Relevansi

Hasil akhir yang diperoleh dari penelitian pada Tugas Akhir ini dapat dijadikan sebagai referensi untuk mempelajari analisis pergerakan robot baik secara kinematik maupun dinamik, serta pengembangan metode kontrol untuk penelitian robot manipulator selanjutnya.

BAB II TEORI PENUNJANG

Bab berisi tentang materi-materi dasar yang digunakan dalam penyelesaian Tugas Akhir ini. Materi-materi dasar yang dibahas dalam Bab ini mencakup pengenalan mengenai robot, kinematika robot, pemodelan dinamik robot dan teori mengenai kontroler yang digunakan.

2.1 Tinjauan Pustaka

Kata robot pertama kali diperkenalkan oleh dramawan dari Ceko Karel Capek pada tahun 1920, kata “robot” merupakan kata Ceko yang memiliki arti untuk bekerja. Berdasarkan RIA (*Robotics Institute of America*), robot merupakan sebuah manipulator multifungsi yang dapat diprogram ulang, didesain untuk memindahkan material, benda-benda, alat-alat, atau dikhususkan dengan gerakan-gerakan yang bisa diprogram untuk melaksanakan bermacam-macam tugas [1].

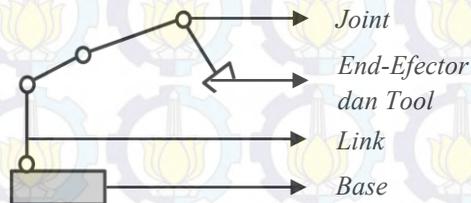
Model matematika sistem diperoleh dengan melakukan penurunan kinematik dan dinamik berdasarkan gaya dan energi yang bekerja melalui persamaan *Euler-Lagrange* sehingga pada akhirnya diperoleh persamaan *state-space* yang merepresentasikan dinamik sistem robot. Model sistem ini kemudian digunakan dalam desain kontroler PD *Fuzzy* CTC untuk mengendalikan pergerakan posisi dari robot.

2.2 Robot Manipulator

Subbab ini membahas bagian-bagian, struktur, konfigurasi dan gerakan robot manipulator secara umum.

2.2.1 Bagian-bagian Robot Manipulator

Robot manipulator memiliki bagian-bagian tertentu secara fisik. Gambar 2.1 menunjukkan bagian-bagian dari robot manipulator.



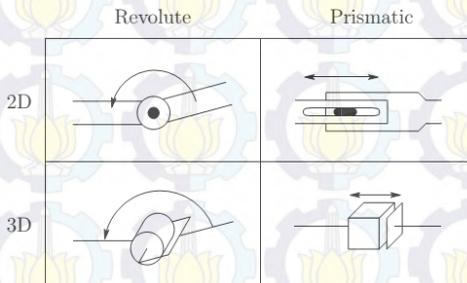
Gambar 2.1 Bagian-bagian Manipulator

Berikut ini penjelasan lebih detail mengenai bagian-bagian robot manipulator yang ditunjukkan pada Gambar 2.1.

- Base* : bagian robot yang diam.
- Link* : bagian robot yang bentuknya tetap dan dapat bergerak.
- Joint* : penghubung antar *link* yang bergerak aktif.
- End-Effector* : ujung yang dapat berinteraksi dengan objek.
- Tool* : alat yang ada pada *end-effector* yang berkaitan dengan fungsi robot.

2.2.2 Komponen dan Struktur Robot

Robot manipulator tersusun dari *link-link* yang terhubung dalam sebuah rantai kinematik terbuka. Secara khusus sendi dibagi menjadi dua yaitu sendi putar (*revolute*) dan sendi linier (prismatik). Sendi *revolute* seperti sebuah engsel dan memungkinkan rotasi relatif antara dua *link*. Sendi prismatik memungkinkan sebuah gerakan relatif linier antara dua *link*. Gambar 2.2 menunjukkan representasi sendi *revolute* dan prismatik.



Gambar 2.2 Representasi Sendi Robot [1]

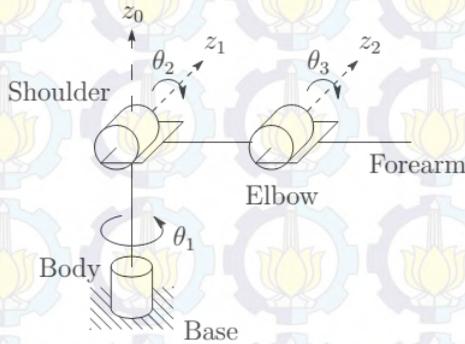
Selanjutnya untuk memudahkan dalam penulisan, sendi *revolute* ditulis dalam “R” dan sendi prismatik ditulis dalam “P”.

Setiap robot manipulator memiliki *degree-of-freedom* masing-masing. *Degree-of-freedom* atau sering disingkat DOF memiliki arti bahwa setiap robot manipulator memiliki tingkat kebebasan dalam bergerak. Banyaknya sendi menentukan jumlah DOF dari robot manipulator, di mana jumlah DOF tersebut menentukan posisi dan orientasi dari robot manipulator [1].

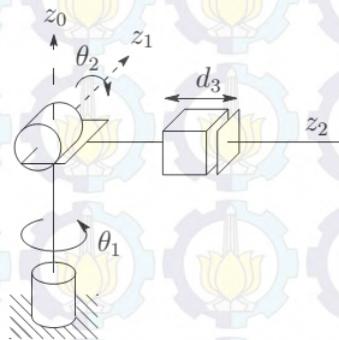
2.2.3 Konfigurasi Robot

Robot manipulator dapat diklasifikasikan menjadi beberapa kriteria, seperti geometri, atau struktur kinematik, tipe dari aplikasi untuk desain robot, pola robot yang dikontrol dan yang lainnya [1].

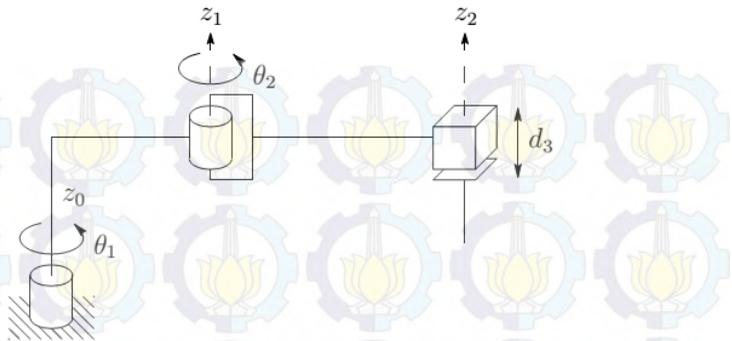
Berdasarkan geometri robot manipulator dapat diklasifikasikan menjadi beberapa konfigurasi. Setidaknya ada lima tipe geometri yang akan dibahas. Berikut ini jenis konfigurasi dari robot manipulator yang ditunjukkan pada Gambar 2.3-2.7.



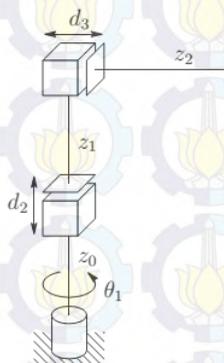
Gambar 2.3 Konfigurasi *Articulated* (RRR) [1]



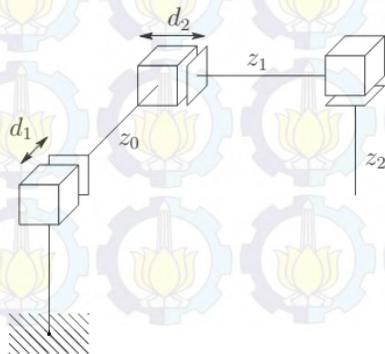
Gambar 2.4 Konfigurasi *Spherical* (RRP) [1]



Gambar 2.5 Konfigurasi SCARA (RRP) [1]



Gambar 2.6 Konfigurasi *Cylindrical* (RPP) [1]



Gambar 2.7 Konfigurasi *Cartesian* (PPP) [1]

2.2.4 Gerakan Kaku dan Transformasi Homogen

Bagian penting dalam memahami pergerakan robot berkaitan dengan pembentukan beragam sistem koordinat untuk merepresentasikan posisi dan orientasi dari objek kaku dan dengan transformasi di antara sistem koordinat [1].

Transformasi homogen mengkombinasikan gerak rotasi dan translasi dari objek kaku menjadi sebuah perkalian matriks tunggal yang nantinya akan digunakan dalam melakukan perhitungan *forward kinematics*.

Dalam perhitungan pergerakan atau perpindahan objek kaku diperlukan vektor yang merupakan besaran yang memiliki nilai dan arah. Pada sistem koordinat terdapat beberapa sumbu koordinat yang mewakili sebuah objek kaku yaitu sumbu x , y dan z . Maka dari itu terdapat beberapa variasi gerakan baik untuk gerak rotasi maupun gerak translasi terhadap sumbu-sumbu tersebut. Dengan perhitungan melalui vektor yang dikaitkan dengan sumbu-sumbu yang ada dibentuk persamaan rotasi dan translasi dalam bentuk matriks. Persamaan matriks rotasi terhadap sumbu x , y dan z ditunjukkan pada Persamaan (2.1-2.3), sedangkan persamaan matriks translasi terhadap sumbu x , y dan z ditunjukkan pada Persamaan (2.4-2.6).

$$R_{x,\theta} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

$$R_{y,\theta} = \begin{bmatrix} \cos \theta & 0 & \sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

$$R_{z,\theta} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

$$Trans_{x,a} = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

$$Trans_{y,b} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

$$Trans_{z,c} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

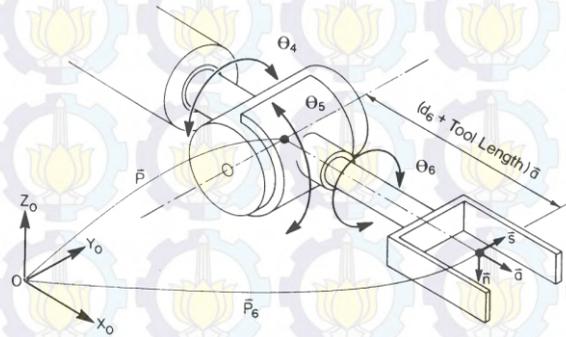
Dari matriks rotasi dan translasi di atas dapat dibentuk matriks transformasi homogen 4x4 yang memetakan sebuah vektor posisi yang diekspresikan dalam koordinat homogen dari satu sistem koordinat ke sistem koordinat yang lain. Matriks transformasi homogen dapat dibagi menjadi empat submatriks, ditunjukkan pada Persamaan (2.7) [3].

$$H = \begin{bmatrix} R_{3 \times 3} & d_{3 \times 1} \\ f_{1 \times 3} & s_{1 \times 1} \end{bmatrix} = \begin{bmatrix} \text{Rotation} & \text{Translation} \\ \text{Perspective} & \text{Scale Factor} \end{bmatrix} \quad (2.7)$$

Transformasi homogen paling umum dituliskan seperti Persamaan (2.8)

$$T = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n & s & a & p \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

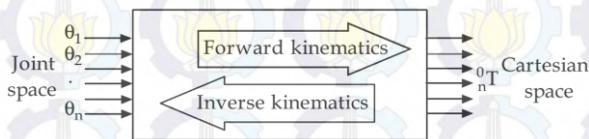
Di mana n adalah vektor normal dari tangan, s adalah vektor geser dari tangan yang menunjuk dalam arah gerak jari sebagai *gripper* membuka dan menutup, a adalah vektor pendekatan dari tangan yang menunjuk ke arah normal terhadap telapak tangan dan p adalah vektor posisi dari tangan yang menunjuk dari titik origin dari sistem koordinat dasar menuju sistem koordinat tangan. Untuk lebih jelasnya Gambar 2.8 menunjukkan sistem koordinat tangan dan n, s, a, p .



Gambar 2.8 Sistem Koordinat Tangan dan n, s, a, p [4]

2.3 Kinematika Robot

Kinematika lengan robot merupakan studi analisis geometri gerak lengan robot terhadap referensi tetap sistem koordinat tanpa memperhatikan kekuatan/gaya atau momen-momen yang menyebabkan gerak [5]. Kinematika lengan robot dibagi menjadi dua yaitu *forward* dan *inverse kinematics*. Gambar 2.9 menunjukkan hubungan antara *forward* dan *inverse kinematics*.



Gambar 2.9 Representasi *Forward* dan *Inverse Kinematics* [6]

Gambar 2.9 menjelaskan perbedaan dalam hal prinsip kerja antara *forward* dan *inverse kinematics*. Bagian kiri merupakan *joint space* (ruang sendi) yang berisi variabel-variabel sendi, sedangkan bagian kanan merupakan *cartesian space* (ruang kartesian) yang merupakan representasi dari transformasi posisi dan orientasi dari *end-effector* robot manipulator.

2.3.1 Forward Kinematics

Forward kinematics berkaitan dengan hubungan antara sendi dari robot manipulator, posisi dan orientasi dari *tool* atau *end-effector*. Masalah *forward kinematics* adalah bagaimana menentukan posisi dan orientasi dari *end-effector*, di mana diketahui nilai untuk variabel sendi dari robot manipulator [1].

Dalam kinematika, khususnya *forward kinematics* untuk melakukan pemodelan robot terdapat standar yang harus dipenuhi. *Denavit-Hartenberg* atau DH merupakan standar yang digunakan dalam pemodelan dan pergerakan robot.

Parameter DH dipublikasikan pertama kali oleh R.S Denavit dan J. Hartenberg pada tahun 1947. Setiap robot dapat digambarkan secara kinematika dengan memberikan empat parameter untuk setiap *link*. Berikut ini merupakan parameter *link* yang digunakan dalam kaidah DH.

a_i = jarak sepanjang x_i dari o_i ke perpotongan dari sumbu x_i dan z_{i-1} .

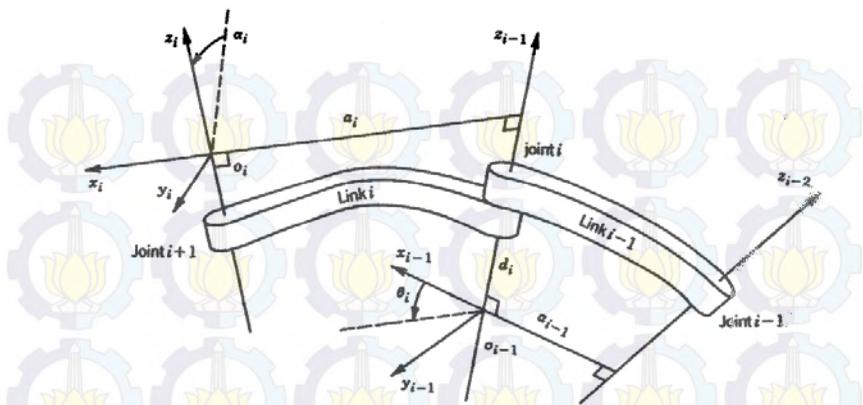
d_i = jarak sepanjang z_{i-1} dari o_{i-1} ke perpotongan sumbu x_i dan z_{i-1} .

d_i adalah variabel jika sendi i adalah sendi prismatic.

α_i = sudut antara z_{i-1} dan z_i diukur terhadap x_i .

θ_i = sudut antara x_{i-1} dan x_i diukur terhadap z_i . θ_i adalah variabel jika sendi i adalah *revolute*.

Agar memudahkan dalam mendapatkan standar DH untuk *link* robot manipulator, Gambar 2.10 menunjukkan penetapan frame *Denavit-Hartenberg* yang menggunakan parameter *link* di atas.



Gambar 2.10 Penetapan *Frame Denavit-Hartenberg* [4]

Untuk memperoleh *forward kinematics* dari setiap robot manipulator terdapat prosedur berdasarkan kaidah DH. Berikut urutan algoritma yang harus dilakukan [1].

Langkah 1: Menempatkan dan memberikan label pada sendi berupa sumbu z_0, \dots, z_{n-1} .

Langkah 2: Menetapkan *base frame*. Menentukan origin pada sumbu z_0 . Sumbu x_0 dan z_0 dipilih secara sembarang untuk membentuk *right-hand frame*.

Langkah 3: Menempatkan origin o_i ke z_i dan z_{i-1} memotong z_i .

Langkah 4: Menetapkan x_i sepanjang common normal antara z_{i-1} dan z_i melalui o_i .

Langkah 5: Menetapkan y_i untuk melengkapi *right-hand frame*.

Langkah 6: Menetapkan *end-effector frame* pada $o_n x_n y_n z_n$.

Langkah 7: Membuat sebuah tabel dari parameter link $a_i, d_i, \alpha_i, \theta_i$.

Langkah 8: Membentuk matriks transformasi homogen A_n dengan melakukan substitusi parameter.

Langkah 9: Membentuk matriks *forward kinematics* $T_0^n = A_1 \dots A_n$.

Matriks ini memberikan posisi dan orientansi dari *tool frame* yang diekspresikan dalam koordinat dasar.

Berdasarkan kaidah DH, setiap transformasi homogen direpresentasikan sebagai sebuah hasil dari empat transformasi dasar.

$$\begin{aligned}
 {}^i T_{i-1} &= Rot_{z,\theta_i} \cdot Trans_{z,d_i} \cdot Trans_{x,a_i} \cdot Rot_{x,\alpha_i} \\
 &= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9) \\
 &= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} c_{\alpha_i} & s_{\theta_i} s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i} c_{\alpha_i} & -c_{\theta_i} s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_i & p_i \\ 0 & 1 \end{bmatrix}
 \end{aligned}$$

2.3.2 Inverse Kinematics

Inverse kinematics memiliki pemahaman yang berkebalikan dengan *forward kinematics*, digunakan untuk menentukan nilai pada variabel sendi dari robot manipulator, di mana diketahui posisi dan orientasi dari *end-effector*.

Inverse kinematic memiliki masalah yang banyak dan kompleks jika dibandingkan dengan *forward kinematic* dikarenakan oleh beberapa alasan berikut

- Persamaan untuk memecahkan permasalahan nonlinier umum, dengan demikian tidak selalu mungkin untuk menemukan solusi *closed-form* (bentuk tertutup).
- Ada beberapa solusi.
- Mungkin tidak ada solusi yang dapat diterima, mengingat struktur kinematik manipulator.

Eksistensi dari setiap solusi dijamin jika posisi *end-effector* dan orientasi yang diberikan termasuk dalam ruang kerja manipulator.

Pada sisi yang lain, masalah banyaknya solusi bergantung tidak hanya pada jumlah DOF saja, melainkan juga pada jumlah parameter DH yang tidak nol.

Perhitungan dari solusi *closed-form* memerlukan intuisi aljabar untuk mencari persamaan yang signifikan dari variabel yang tidak diketahui atau intuisi geometri untuk mencari titik yang signifikan pada struktur yang berkaitan, yang akan lebih mudah untuk mengekspresikan posisi dan atau orientasi sebagai fungsi dari berkurangnya jumlah variabel yang tidak diketahui [3].

2.4 Dinamik Robot Manipulator

Jika kinematika menggambarkan gerakan robot manipulator tanpa memperhatikan kekuatan/gaya dan momen-momen yang menghasilkan gerakan, sedangkan dinamik secara eksplisit menggambarkan hubungan antara kekuatan/gaya dan momen-momen yang bekerja. Persamaan gerakan sangatlah penting digunakan sebagai bahan pertimbangan dalam desain robot, sekaligus dalam simulasi dan animasi, serta dalam desain algoritma kontrol [1].

Untuk mendapatkan persamaan dinamik dari gerakan robot digunakan teknik berdasarkan dinamik *Lagrange* secara sistematis.

2.4.1 Perumusan *Lagrange*

Model dinamik dari sebuah manipulator menyediakan sebuah deskripsi dari hubungan antara torsi aktuator sendi dan gerakan dari struktur manipulator.

Dengan rumus *Lagrange*, persamaan dari gerakan dapat diperoleh dalam sebuah cara sistematis secara independen dari frame koordinat referensi. Ketika sejumlah variabel q_i , $i = 1, \dots, n$, diistilahkan koordinat umum, dipilih yang secara efektif menggambarkan posisi link dari sejumlah n -DOF manipulator, Lagrangian dari sistem mekanik dapat ditentukan sebagai sebuah fungsi dari koordinat umum

$$L = K - P \quad (2.10)$$

Di mana K menunjukkan total energi kinetik dan P menunjukkan energi potensial dari sistem.

Persamaan *Lagrange* diekspresikan dengan

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \xi_i \quad i = 1, \dots, n \quad (2.11)$$

Di mana ξ_i adalah gaya umum yang berkaitan dengan koordinat umum q_i . Persamaan (2.11) dapat ditulis dalam bentuk ringkas seperti

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right)^T - \left(\frac{\partial L}{\partial q} \right)^T = \xi \quad (2.12)$$

Di mana untuk sebuah manipulator dengan sebuah rantai kinematik terbuka, koordinat umum dikumpulkan dalam vektor variabel sendi q . Kontribusi pada gaya umum diberikan oleh gaya nonkonservatif, dengan kata lain, torsi aktuator sendi, torsi friksi sendi, sekaligus torsi sendi diinduksi oleh gaya *end-effector* pada kondisi hubungan dengan lingkungan [5].

Persamaan (2.11) menetapkan hubungan yang ada antara gaya umum yang diaplikasikan pada manipulator dan posisi sendi, kecepatan dan percepatan. Maka dari itu persamaan tersebut mengijinkan penurunan dari energi kinetik dan energi potensial dari sistem mekanik.

2.4.2 Perhitungan Energi Kinetik

Energi kinetik dari sebuah objek merupakan jumlah dari dua kondisi, energi translasi diperoleh dengan memusatkan keseluruhan massa pada bagian benda terhadap pusat massa, dan energi rotasi dari benda terhadap pusat massa [5].

Dengan menganggap sebuah objek terdiri dari sebuah rangkaian kesatuan dari partikel, dan ρ menunjukkan *density* dari massa objek. Lalu menganggap bahwa B menunjukkan tingkat dari objek tersebut, sehingga

$$\int_B \rho(x, y, z) dx dy dz = m \quad (2.13)$$

Di mana m adalah massa dari objek. Kemudian energi kinetik dari objek menjadi

$$\begin{aligned}
 K &= \frac{1}{2} \int_B \mathbf{v}^T(x, y, z) \mathbf{v}(x, y, z) \rho(x, y, z) dx dy dz \\
 &= \frac{1}{2} \int_B \mathbf{v}^T(x, y, z) \mathbf{v}(x, y, z) dm
 \end{aligned}
 \tag{2.14}$$

lalu dengan menganggap *frame* koordinat dipasang pada pusat dari massa, dengan *origin*-nya pada pusat massa tersebut. Ketika objek berpindah, objek memiliki kecepatan

$$\mathbf{v} = \mathbf{v}_c + \boldsymbol{\omega} r
 \tag{2.15}$$

dengan melakukan substitusi Persamaan (2.15) ke dalam Persamaan (2.14) maka

$$K = \frac{1}{2} m \mathbf{v}_c^T \mathbf{v}_c + \frac{1}{2} \boldsymbol{\omega}^T I \boldsymbol{\omega}
 \tag{2.16}$$

Di mana I merupakan matriks inersia

$$I = \begin{bmatrix} \int (y^2 + z^2) dm & -\int xy dm & -\int xz dm \\ -\int xy dm & \int (x^2 + z^2) dm & -\int yz dm \\ -\int xz dm & -\int yz dm & \int (x^2 + y^2) dm \end{bmatrix}
 \tag{2.17}$$

Kecepatan sudut pada Persamaan (2.15) didapatkan melalui vektor S yang merupakan sebuah matriks miring yang memiliki fungsi terhadap kecepatan sudut [1].

$$S(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}
 \tag{2.18}$$

2.4.3 Perhitungan Energi Potensial

Pembahasan selanjutnya mengenai energi potensial. Dalam kasus dinamik kaku, satu-satunya yang menjadi sumber energi potensial adalah

gaya gravitasi. Dengan asumsi bahwa g menunjukkan vektor gravitasi yang diekspresikan dalam *base frame*. Energi potensial merupakan partikel yang sangat kecil yang terletak pada jarak r dari objek [1]. Karenanya energi potensial secara keseluruhan sebagai berikut

$$P = \int_B g^T r dm = \int_B g^T r_c m \quad (2.19)$$

Dengan kata lain, energi potensial dari objek sama seperti massa dari keseluruhan objek yang difokuskan pada pusat massa. Dan hal yang perlu diketahui bahwa energi potensial dari manipulator bergantung hanya pada vektor q .

2.4.4 Persamaan Gerakan

Persamaan gerakan adalah persamaan dinamik dari manipulator yang merupakan jumlah dari gaya-gaya yang bekerja. Total dari gaya-gaya yang bekerja tersebut merupakan torsi yang dimiliki tiap lengan manipulator [1].

Dengan terhitungnya total dari energi kinetik dan potensial, maka persamaan *Lagrange* bisa dikalkulasi, sesuai dengan Persamaan (2.10). Persamaan gerakan bisa didapatkan dengan formulasi sesuai Persamaan (2.11) dan (2.12) sehingga didapatkan persamaan yang lebih khusus seperti

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau \quad (2.18)$$

Persamaan (2.18) di atas berupa persamaan matriks yang memiliki tiga bagian subpersamaan. Subpersamaan yang pertama merupakan turunan kedua dari koordinat umum yaitu matriks inersia. Subpersamaan yang kedua merupakan gabungan dari matriks sentrifugal dan *coriolis*. Dan subpersamaan yang terakhir merupakan gaya gravitasi dari manipulator.

2.5 Kontroler

Kontroler merupakan salah satu komponen dari sistem yang digunakan untuk mengolah hasil jumlah dari sinyal umpan balik dan sinyal referensi sehingga menghasilkan *output* berupa sinyal kontrol sehingga performansi dari sistem yang dikendalikan sesuai dengan

spesifikasi performansi yang diinginkan. Keberadaan kontroler dalam sebuah sistem kontrol mempunyai kontribusi yang sangat besar terhadap perilaku sistem. Pada prinsipnya hal tersebut disebabkan karena tidak bisa diubahnya komponen-komponen penyusun dalam sistem tersebut sehingga karakteristik *plant* harus diterima sebagaimana adanya. Oleh karena itu sebuah kontroler sangat dibutuhkan untuk melakukan perubahan pada perilaku sistem dengan menyisipkan kontroler sebagai subsistem. Salah satu fungsi kontroler adalah untuk meminimalisir sinyal kesalahan yang didapatkan dari selisih antara masukan sinyal referensi dan keluaran sinyal aktual. Tujuannya adalah agar nilai dari sinyal aktual sesuai dengan nilai dari sinyal referensi (*set point*) atau sinyal yang diinginkan. Sehingga semakin cepat respon sistem mengikuti sinyal aktual dan semakin kecil sinyal kesalahan yang terjadi, maka semakin baik performansi sebuah sistem kontrol.

2.5.1 Kontroler *Computed Torque Control* (CTC)

Kontroler CTC berdasarkan pada linearisasi umpan balik dan menghitung besar torsi lengan robot yang diperlukan dengan menggunakan hukum kontrol *feedback* nonlinier. Kontroler ini bekerja dengan sangat baik ketika semua parameter fisik dan dinamik diketahui, tetapi ketika robot manipulator memiliki variasi dalam parameter dinamik, dalam situasi seperti ini kontroler tidak mampu menangani kinerja yang diinginkan [2].

Peran utama dari *Computed Torque Controller* (CTC) digunakan untuk linierisasi umpan balik sehingga algoritma CTC disebut juga kontroler *feedback linearization controller*. Diasumsikan bahwa gerakan trayektori yang diinginkan untuk robot manipulator q_d ditentukan untuk mengikuti *path* (lintasan), dengan menentukan *tracking error* sebagai berikut

$$e(t) = q_d(t) - q_a(t) \quad (2.19)$$

Di mana $e(t)$ merupakan *error* dari *plant*, $q_d(t)$ merupakan variabel *input* yang diinginkan, dan $q_a(t)$ merupakan perpindahan posisi aktual. Jika sebuah persamaan *state-space* linier alternatif didefinisikan dalam bentuk berikut

$$\dot{x} = Ax + Bu \quad (2.20)$$

Maka Persamaan (2.20) dapat didefinisikan seperti berikut

$$\dot{x} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ I \end{bmatrix} u \quad (2.21)$$

Di mana,

$$u = -M^{-1}(q)N(q, \dot{q}) + M^{-1}(q)\tau \quad (2.22)$$

Dan bentuk ini dikenal sebagai *Brunousky canonical form*. Dengan menggunakan Persamaan (2.19) dan Persamaan (2.21) maka *Brunousky canonical form* dapat ditulis seperti berikut

$$\frac{d}{dt} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} u \quad (2.23)$$

Dengan,

$$u = \ddot{q}_d + M^{-1}(q)[N(q, \dot{q}) - \tau] \quad (2.24)$$

Sehingga didapatkan formulasi *inverse* untuk mendapatkan besar torsi dari robot manipulator sebagai berikut

$$\tau = M(q)(\ddot{q}_d - u) + N(q, \dot{q}) \quad (2.25)$$

2.5.2 Kontroler PID

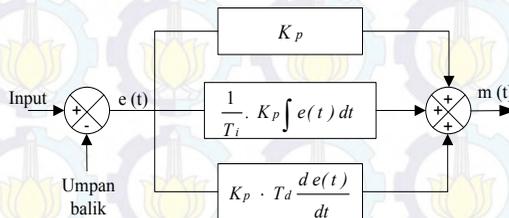
Kontroler PID merupakan kontroler yang paling banyak digunakan di dunia industri karena kemudahan, kesederhanaan, kehandalan dan penggunaannya yang aplikatif. Kontroler PID terdiri dari tiga komponen yaitu proporsional, integral dan *derivative*. Kontroler dapat digunakan secara terpisah maupun dengan melakukan kombinasi tertentu seperti P, PI, PD dan PID.

Setiap kekurangan dari tiap kontroler proporsional, integral, dan *derivative* (kontroler P, I, dan D) dapat diminimalkan dengan cara mengkombinasikan secara paralel agar dapat saling melengkapi dengan

kelebihan masing-masing. Blok diagram kontroler PID ditunjukkan pada Gambar 2.11 menghasilkan keluaran yang bergantung pada *magnitude* dan laju perubahan dari sinyal kesalahan. Karakteristik kontroler PID dipengaruhi oleh ketiga parameter penyusunnya, yaitu K_p , K_i , dan K_d .

Dari Persamaan (2.26), nilai K_p berlaku untuk penguatan saja tanpa memberikan efek dinamik terhadap kinerja kontroler. Penggunaan kontroler P memiliki keterbatasan karena sifat kontrol yang tidak dinamis, sehingga tidak sesuai digunakan untuk *plant* dengan karakter dinamis.

Kontroler integral lebih banyak digunakan untuk meminimalkan kesalahan (*error*). Konstanta integral K_i , dapat memperbaiki sekaligus menghilangkan *error steady-state* (E_{ss}), namun pemilihan K_i yang tidak tepat dapat menyebabkan respon *transient* yang tinggi sehingga dapat menyebabkan ketidakstabilan sistem. Selain itu pemilihan K_i yang sangat tinggi justru dapat menyebabkan timbulnya osilasi karena bertambahnya orde sistem [7].



Gambar 2.11 Diagram Blok Kontroler PID

Kontroler *derivative* tidak dapat berdiri sendiri karena sifat dari kontroler ini hanya berubah saat ada perubahan kesalahan sehingga saat terjadi kesalahan statis, kontroler ini tidak akan bereaksi. Dibalik kelemahan tersebut, kontroler *derivative* mempunyai kecepatan respon yang baik untuk mengatasi adanya sinyal kesalahan yang terjadi, sehingga dapat digunakan untuk memperbaiki respon transien.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (2.26)$$

Di mana,

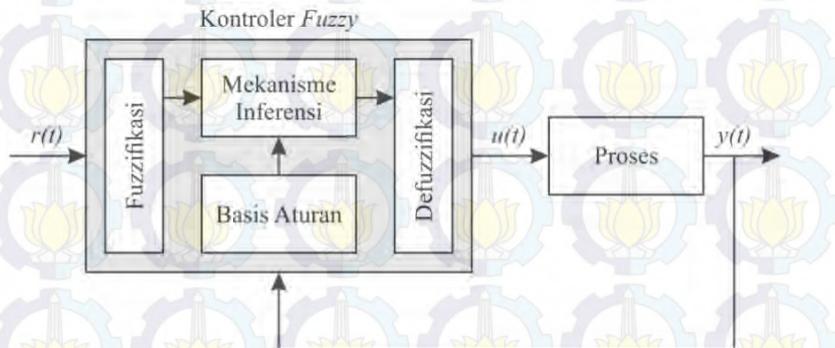
$u(t)$ = sinyal keluaran kontroler PID

- K_p = konstanta proporsional
- T_i = waktu integral
- T_d = waktu *derivative*
- K_i = konstanta integral (K_p/T_i)
- K_d = konstanta *derivative* ($K_p \times T_d$)
- $e(t)$ = sinyal kesalahan

2.5.3 Kontroler *Fuzzy*

Kontroler *Fuzzy* merupakan kontroler alternatif yang praktis untuk berbagai macam aplikasi kontrol. Kontroler ini memiliki metode yang memberikan kemudahan dalam menyusun kontroler-kontroler nonlinier dengan menggunakan informasi heuristik. Informasi tersebut dapat berasal dari seorang operator yang bekerja sebagai kontroler “human in the loop” untuk suatu proses [9].

Di dalam metodologi desain kontroler *Fuzzy*, operator akan menuliskan aturan-aturan mengenai cara-cara mengontrol suatu proses. Informasi tersebut selanjutnya akan dimasukkan ke dalam kontroler *Fuzzy* sebagai dasar bagi kontroler untuk membuat suatu keputusan, kontroler ini memiliki cara yang hampir mirip dengan cara yang dimiliki oleh seorang operator. Gambar 2.12 menunjukkan diagram kontroler *Fuzzy*. Gambar tersebut menunjukkan bahwa kontroler *Fuzzy* terdiri dari empat buah elemen, yaitu basis aturan, mekanisme inferensi, fuzzifikasi, dan defuzzifikasi.



Gambar 2.12 Diagram Blok Kontroler *Fuzzy*

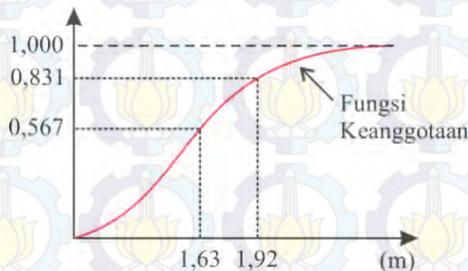
Basis aturan (berisi aturan-aturan *If-Then*), berisi perhitungan-perhitungan (*quantification*) dari deskripsi linguistik dari seorang ahli (contohnya operator) untuk menghasilkan kontrol yang baik. Mekanisme inferensi berisi mekanisme-mekanisme yang menyerupai seorang ahli dalam membuat keputusan dalam menginterpretasikan dan menerapkan pengetahuan tentang cara terbaik untuk mengontrol proses. Pada blok fuzzifikasi masukan kontroler akan dikonversi menjadi informasi yang mudah digunakan untuk mekanisme inferensi, dan keputusan akhir dari mekanisme inferensi dikonversi menjadi masukan yang sebenarnya untuk proses.

2.5.3.1 Himpunan Fuzzy

Langkah pertama yang harus dilakukan untuk menggunakan logika *Fuzzy* adalah menentukan arti dari tiap nilai-nilai linguistik dengan menggunakan fungsi keanggotaan. Nilai-nilai linguistik tersebut pada umumnya merupakan suatu bahasa yang menunjukkan tingkatan tertentu.

Gambar 2.13 menunjukkan plot fungsi keanggotaan tinggi badan manusia. Dengan fungsi keanggotaan tersebut tinggi badan seseorang akan diukur nilai derajat keanggotaannya. Derajat keanggotaan tersebut bernilai dari “0” sampai dengan “1”. Apabila diketahui tinggi badan seseorang adalah 1,63 m, maka derajat keanggotaannya dalam himpunan keanggotaan tinggi badan manusia adalah 0,567. Apabila diketahui tinggi badan yang lain adalah 1,92 m maka derajat keanggotaan orang tersebut dalam himpunan adalah 0,831.

Fungsi keanggotaan merupakan pengukuran yang subjektif [9]. Fungsi keanggotaan tinggi badan menurut orang Asia, Eropa, dan pemain NBA (*National Basketball Association*) dapat berbeda-beda.



Gambar 2.13 Fungsi Keanggotaan Tinggi Badan

Menurut orang Asia, seseorang dengan tinggi badan 1,8 m sudah termasuk sangat tinggi, namun bagi orang Eropa tinggi tersebut merupakan tinggi rata-rata. Berbeda lagi dengan pemain NBA, baginya 1,8 m tidak terlalu tinggi. Plot fungsi keanggotaan tinggi badan menurut orang Asia, Eropa, dan pemain NBA dapat dilihat pada Gambar 2.14.

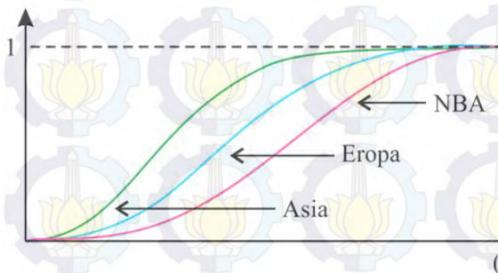
Definisi formal himpunan *Fuzzy* ditunjukkan oleh Persamaan (2.27), dengan X merupakan semesta pembicaraan, A adalah himpunan *Fuzzy*, dan μ_A merupakan fungsi keanggotaan *Fuzzy* untuk himpunan A .

$$A = \{(x, \mu_A(x)) \mid x \in X\} \quad (2.27)$$

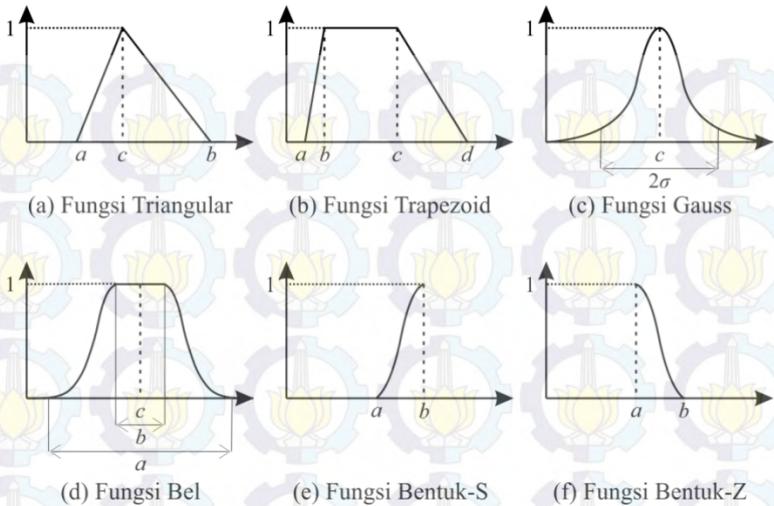
Tipe-tipe fungsi keanggotaan *Fuzzy* secara umum dapat dilihat pada Gambar 2.15, dengan (a) adalah fungsi keanggotaan Triangular, (b) adalah fungsi keanggotaan Trapezoid, (c) adalah fungsi keanggotaan Gauss, (d) adalah fungsi keanggotaan Bel, (e) adalah fungsi keanggotaan bentuk S, dan (f) adalah fungsi keanggotaan bentuk Z [10].

Fungsi ini dideskripsikan oleh tiga parameter, yaitu parameter a , b , dan c , dan diekspresikan dengan formulasi yang ditunjukkan oleh Persamaan (2.28). Parameter a dan c terletak pada “kaki-kaki” segitiga, dan parameter b menunjukkan “puncak” dari segitiga.

$$f(x; a, b, c) = \left\{ \begin{array}{ll} 0 & , \quad x \leq a \\ \frac{x-a}{b-a} & , \quad a \leq x \leq b \\ \frac{c-x}{c-b} & , \quad b \leq x \leq c \\ 0 & , \quad c \leq x \end{array} \right\} \quad (2.28)$$



Gambar 2.14 Fungsi Keanggotaan Tinggi Badan Asia, Eropa, NBA



Gambar 2.15 Tipe Fungsi Keanggotaan *Fuzzy*

Fungsi keanggotaan Gauss diekspresikan oleh formulasi pada Persamaan (2.29), dengan parameter c menunjukkan jarak dari titik origin dan σ merupakan lebar pita dari lengkungan Gauss. Fungsi ini memiliki bentuk simetris dan memiliki nilai tidak sama dengan nol pada setiap titik x .

$$f(x; \sigma, c) = e^{-\frac{(x-c)^2}{2\sigma^2}} \quad (2.29)$$

Fungsi keanggotaan Bel memiliki bentuk simetris dan diekspresikan oleh formulasi pada Persamaan (2.30), dengan parameter b umumnya bernilai positif, parameter c terletak di tengah lengkungan Bel, dan parameter a merupakan lebar pita lengkungan Bel. Sama seperti fungsi keanggotaan Gauss, fungsi ini memiliki nilai tidak sama dengan nol pada setiap titik x .

$$f(x; \sigma, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (2.30)$$

Fungsi keanggotaan bentuk S dan fungsi keanggotaan bentuk Z memiliki bentuk yang asimetris dan diekspresikan oleh formulasi pada Persamaan (2.31) dan Persamaan (2.32), dengan parameter a dan b menunjukkan titik ekstrim lekukan (*slope*) dari lengkungan bentuk S dan bentuk Z.

$$f(x; a, b) = \left. \begin{array}{l} 0, \quad x \leq a \\ 2\left(\frac{x-a}{b-a}\right)^2, \quad a \leq x \leq \frac{a+(b-a)}{2} \\ 1-2\left(\frac{x-b}{b-a}\right)^2, \quad \frac{a+(b-a)}{2} \leq x \leq b \\ 1, \quad b \leq x \end{array} \right\} \quad (2.31)$$

$$f(x; a, b) = \left. \begin{array}{l} 1, \quad x \leq a \\ 1-2\left(\frac{x-a}{b-a}\right)^2, \quad a \leq x \leq \frac{(a+B)}{2} \\ 2\left(b-\frac{x}{b-a}\right)^2, \quad \frac{(a+b)}{2} \leq x \leq b \\ 1, \quad b \leq x \end{array} \right\} \quad (2.32)$$

2.5.3.2 Operasi Himpunan Fuzzy

Bilangan Fuzzy merupakan konveks himpunan Fuzzy A pada garis bilangan *real*. Min dan Max merupakan operasi sederhana yang digunakan untuk memberi perintah terhadap bilangan-bilangan Fuzzy. Operasi dasar pada himpunan Fuzzy adalah union (OR), interseksi (AND), dan komplemen (NOT).

Union (gabungan) dari himpunan Fuzzy A dan B dengan fungsi keanggotaan $\mu_A(x)$ dan $\mu_B(x)$ adalah $\mu_{A \cup B}(x)$. Untuk semua $x | x \in X$, operasi maksimum himpunan Fuzzy A dan B dinyatakan dalam Persamaan (2.33).

$$\mu_{A \cup B}(x) = \max\{\mu_A(x), \mu_B(x), x \in X \quad (2.33)$$

Interseksi (irisan) dari himpunan *Fuzzy* A dan B dengan fungsi keanggotaan $\mu_A(x)$ dan $\mu_B(x)$ adalah $\mu_{A \cap B}(x)$. Untuk semua $x | x \in X$, operasi minimum himpunan *Fuzzy* A dan B dinyatakan dalam Persamaan (2.34).

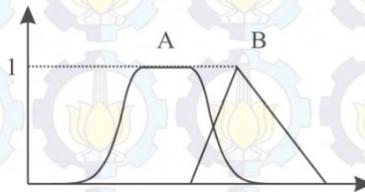
$$\mu_{A \cap B}(x) = \min\{\mu_A(x), \mu_B(x), x \in X \quad (2.34)$$

Komplemen dari himpunan *Fuzzy* A dan B dengan fungsi keanggotaan $\mu_A(x)$ dan $\mu_B(x)$ adalah $\mu_A'(x)$ dan $\mu_B'(x)$. Untuk semua operasi $x | x \in X$, komplemen himpunan *Fuzzy* A dan B dinyatakan dalam Persamaan (2.35) dan (2.36).

$$\mu_A'(x) = 1 - \mu_A(x), x \in X \quad (2.35)$$

$$\mu_B'(x) = 1 - \mu_B(x), x \in X \quad (2.36)$$

Sebagai contoh, diketahui dua buah himpunan *Fuzzy* yaitu himpunan *Fuzzy* A dan himpunan *Fuzzy* B. Fungsi keanggotaan himpunan *Fuzzy* A merupakan fungsi keanggotaan Bel dan fungsi keanggotaan himpunan *Fuzzy* B merupakan fungsi keanggotaan Triangular. *Plot* fungsi keanggotaan himpunan A dan B dapat dilihat pada Gambar 2.16.



Gambar 2.16 Plot Fungsi Keanggotaan Himpunan *Fuzzy* A dan B

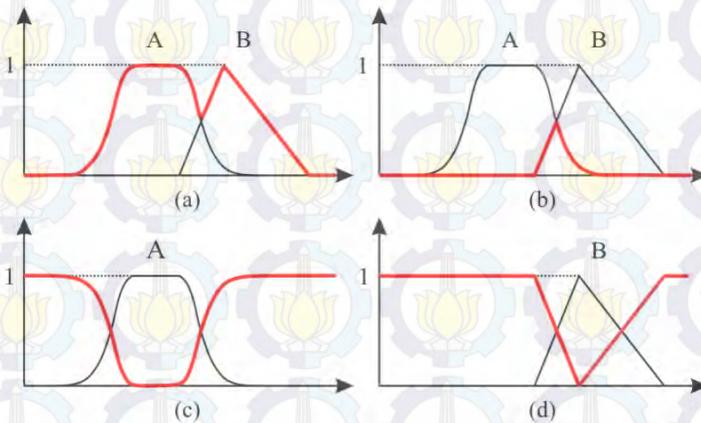
Hasil operasi dasar himpunan *Fuzzy* (OR, AND, NOT) dari himpunan *Fuzzy* A dan himpunan *Fuzzy* B dapat dilihat pada Gambar 2.17

(lihat garis yang berwarna merah). Gambar 2.17 (a) adalah operasi union himpunan *Fuzzy* A dan B ($A \text{ OR } B$), (b) adalah operasi interseksi himpunan *Fuzzy* A dan B ($A \text{ AND } B$), (c) adalah operasi komplemen himpunan *Fuzzy* A ($\text{NOT } A$), dan (d) adalah operasi komplemen himpunan *Fuzzy* B ($\text{NOT } B$). Operasi himpunan *Fuzzy* digunakan untuk memanipulasi fungsi keanggotaan *Fuzzy* satu dengan fungsi lainnya.

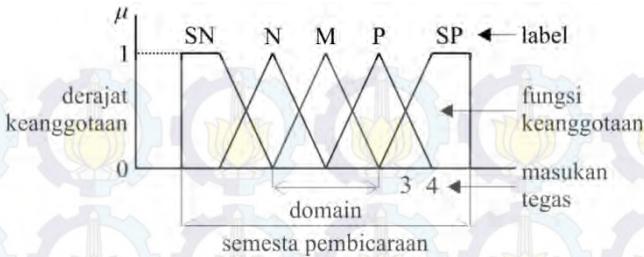
2.5.3.3 Fuzzifikasi

Fuzzifikasi adalah proses mengubah nilai tegas menjadi nilai *Fuzzy* dengan menggunakan fungsi keanggotaan *Fuzzy*. Misal masukan tegas berasal dari hasil pengukuran suatu sensor, maka nilai pengukuran tersebut akan diubah menjadi nilai *Fuzzy* dengan menggunakan fungsi keanggotaan *Fuzzy* yang telah ditentukan. Fungsi tersebut harus mencakup seluruh semesta pembicaraan dari masukan tegas (*range* hasil pengukuran sensor yang mungkin).

Struktur fungsi keanggotaan *Fuzzy* dapat dilihat pada Gambar 2.17. Derajat keanggotaan adalah derajat dari masukan tegas pada sebuah fungsi keanggotaan, memiliki nilai 0 sampai 1. Fungsi keanggotaan mendefinisikan nilai *Fuzzy* dengan melakukan pemetaan nilai tegas berdasarkan daerahnya untuk diasosiasikan dengan derajat keanggotaan.



Gambar 2.17 Operasi Himpunan *Fuzzy*



Gambar 2.18 Struktur Fungsi Keanggotaan *Fuzzy*

Masukan tegas pada umumnya merupakan hasil pengukuran parameter eksternal dari sistem kontrol. Label merupakan deskripsi nama untuk menunjukkan suatu fungsi keanggotaan *Fuzzy* yang memiliki domain (lebar fungsi keanggotaan *Fuzzy*) tertentu. Semesta pembicaraan memiliki jarak yang mencakup seluruh masukan tegas yang mungkin ada. Bentuk fungsi keanggotaan harus mewaliki variabel masukan tegas, namun bentuk yang digunakan dibatasi oleh kemampuan *tool* dalam melakukan perhitungan. Bentuk fungsi yang rumit membutuhkan persamaan fungsi yang lebih kompleks.

2.5.3.4 Defuzzifikasi

Defuzzifikasi merupakan proses mengubah konklusi dari inferensi *Fuzzy* ke dalam bentuk aktual untuk men-*drive* proses. Terdapat beberapa prosedur defuzzifikasi, yaitu *Center of Area*, *Center of Largest Area*, *First of Maxima*, *Middle of Maxima*, dan *Mean of Maxima* [11].

Center of Area

Metode *Center of Area* merupakan prosedur defuzzifikasi yang paling umum digunakan. Metode ini menentukan titik tengah area yang merupakan titik pusat massa dari kombinasi fungsi-fungsi keanggotaan. Formulasi perhitungan titik tersebut, yang nantinya akan menjadi keluaran tegas (u^*), ditunjukkan oleh Persamaan (2.37), dengan b_i merupakan titik tengah area tiap fungsi keanggotaan dan $\int \mu_{(i)}$ merupakan area di bawah fungsi keanggotaan $\mu_{(i)}$.

$$u^* = \frac{\sum_i b_i \int \mu_{(i)}}{\sum_i \int \mu_{(i)}} \quad (2.37)$$

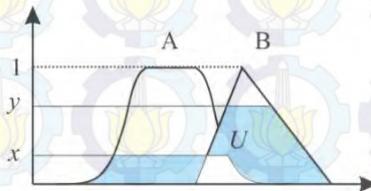
Gambar 2.19 menunjukkan kombinasi fungsi keanggotaan himpunan A dan B dengan derajat keanggotaan x untuk himpunan A dan y untuk himpunan B. Pada contoh ini area kombinasi fungsi keanggotaan himpunan A dan B dinamakan U , seluas area berwarna biru. Irisan area himpunan A dan B tidak termasuk area kombinasi fungsi keanggotaan himpunan A dan B, karenanya area tersebut tidak memberikan partisipasi dalam perhitungan keluaran tegas.

Center of Largest Area

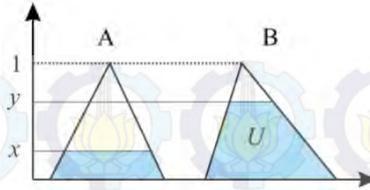
Metode *Center of Largest Area* digunakan ketika dua himpunan *Fuzzy* tidak saling tumpang tindih. Metode ini menggunakan area himpunan *Fuzzy* yang paling besar untuk menghitung nilai keluaran tegas, titik tengah area. Gambar 2.20 menunjukkan himpunan *Fuzzy* yang tidak tumpang tindih. Titik tengah area berwarna biru pada himpunan B digunakan untuk mencari nilai keluaran tegas (u^*).

First of Maxima

Metode *First of Maxima* menggunakan gabungan fungsi keanggotaan dan mengambil nilai paling kecil dari area fungsi keanggotaan dengan derajat keanggotaan paling besar. Metode ini dapat diekspresikan dalam bentuk persamaan, bisa dilihat Persamaan (2.38).



Gambar 2.19 Defuzzifikasi *Center of Area*



Gambar 2.20 Himpunan *Fuzzy* Tidak Saling Tumpang Tindih

$$u^* = \inf_{u \in U} \{u \in U \mid \mu_U(u) = \text{hgt}(u)\} \quad (2.38)$$

Gabungan himpunan *Fuzzy* A dan B dengan derajat keanggotaan x dan y ditandai dengan area berwarna biru pada Gambar 2.21 (a). Nilai terendah dengan derajat keanggotaan tertinggi merupakan u^* .

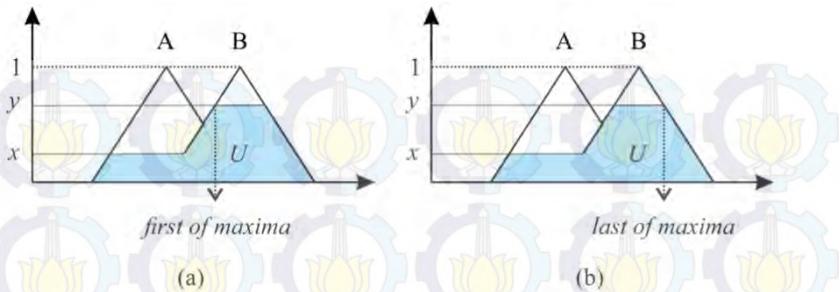
Versi lain dari metode *First of Maxima* adalah metode *Last of Maxima*. Metode menggunakan gabungan fungsi keanggotaan *Fuzzy* dan mengambil nilai paling besar pada derajat keanggotaan yang tertinggi, ditunjukkan oleh Gambar 2.21 (b). Metode ini secara formal ditulis dengan,

$$u^* = \sup_{u \in U} \{u \in U \mid \mu_U(u) = \text{hgt}(u)\} \quad (2.39)$$

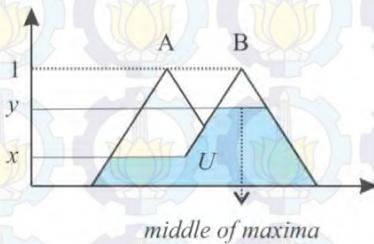
Middle of Maxima

Metode *Middle of Maxima*, ditunjukkan oleh Gambar 2.22, hampir mirip dengan metode *First of Maxima* ataupun *Last of Maxima*. Metode ini mengambil semua nilai pada gabungan himpunan-himpunan *Fuzzy* dengan nilai derajat keanggotaan tertinggi. Keluaran tegas dari metode ini menggunakan nilai rata-rata dari nilai-nilai yang memiliki derajat keanggotaan tertinggi.

$$u^* = \left(\frac{\inf_{u \in U} \{u \in U \mid \mu_U(u) = \text{hgt}(u)\}}{\sup_{u \in U} \{u \in U \mid \mu_U(u) = \text{hgt}(u)\}} \right) \quad (2.40)$$



Gambar 2.21 Defuzzifikasi *First of Maxima* dan *Last of Maxima*



Gambar 2.22 Defuzzifikasi *Middle of Maxima*

Mean of Maxima

Metode *Mean of Maxima* mengambil semua nilai tiap fungsi keanggotaan dengan derajat keanggotaan maksimum dan menghitung rata-rata dari nilai-nilai tersebut sebagai keluaran tegas. Secara formal metode ini ditulis dengan,

$$u^* = \frac{\sum_{i=1}^n \{u \in U \mid \mu_U(u) = \text{hgt}(u)\}}{n} \quad (2.41)$$

2.5.3.5 Sistem Inferensi Fuzzy

Pada bagian inferensi Fuzzy ini akan dijelaskan mengenai aturan Fuzzy. Aturan Fuzzy menyatakan pemetaan masukan Fuzzy terhadap keluaran Fuzzy, dapat dinyatakan sebagai,

If premise Then consequent

Aturan-aturan *Fuzzy* yang telah tersusun akan digunakan pada mekanisme inferensi *Fuzzy*. Mekanisme inferensi *Fuzzy* adalah proses pengambilan keputusan dari masing-masing premis sesuai dengan aturan *Fuzzy* yang ditentukan.

Pengambilan keputusan yang dilakukan oleh mekanisme inferensi *Fuzzy* terdiri atas lima langkah, yaitu fuzzifikasi masukan, aplikasi operator *Fuzzy*, aplikasi metode implikasi, agregrasi semua keluaran, dan defuzzifikasi.

Pada fuzzifikasi masukan dilakukan pemetaan masukan tegas menjadi nilai *Fuzzy* yang berkaitan, sehingga tiap masukan tegas memiliki derajat keanggotaannya. Masukan tersebut selanjutnya diolah dengan menggunakan operator *Fuzzy*, seperti AND, OR, dan NOT. Selanjutnya dilakukan perhitungan kuat penyulutan pada himpunan *Fuzzy* yang terdefinisi untuk setiap variabel keluaran pada bagian konsekuen setiap aturan, ini disebut aplikasi metode implikasi. Semua hasil implikasi *Fuzzy* dari setiap aturan digabungkan untuk menghasilkan keluaran *Fuzzy*. Operasi yang dilakukan menggunakan *max-min* atau *max-product*. Hasil operasi tersebut difuzzifikasi sehingga diperoleh keluaran tegas.

Berikut akan diberikan contoh mekanisme inferensi *Fuzzy*. Dimisalkan suatu aturan *Fuzzy*,

If x is A Then u is B

A dan B merupakan nilai linguistik yang didefinisikan oleh himpunan *Fuzzy* dari variabel linguistik x dan u . Consequent “ u is B” adalah kesimpulan yang didapat ketika suatu premis “ x is A” terpenuhi. Jika masukan *Fuzzy* lebih dari satu, pernyataan premis pada aturan *Fuzzy* dapat menggunakan operator OR atau AND untuk masing-masing masukan *Fuzzy*.

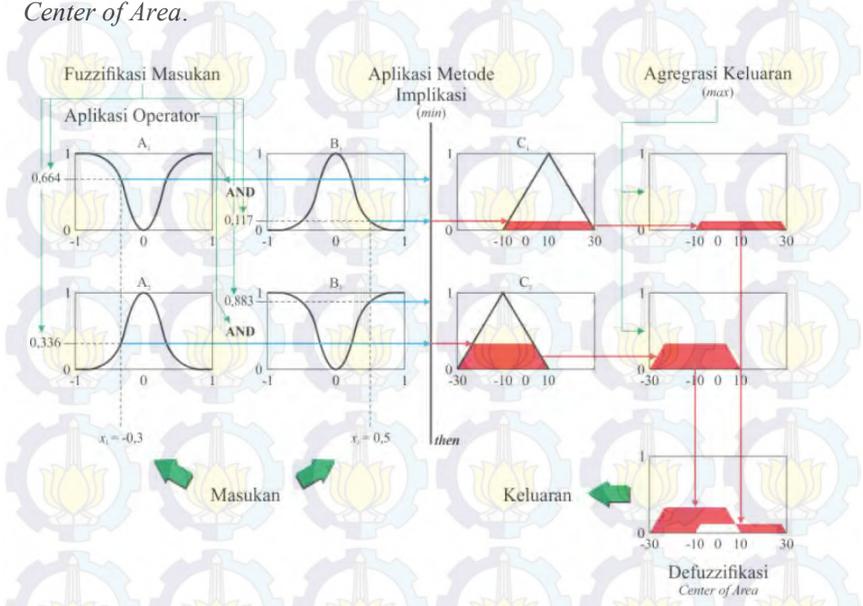
Sebagai contoh model *Mamdani* dengan dua premis dan dua aturan, dipilih operator AND maka aturan *Fuzzy* menjadi,

Aturan 1 : *If x_1 is A_1 AND x_2 is B_1 then u is C_1*

Aturan 2 : *If x_2 is A_2 AND x_2 is B_2 then u is C_2*

Gambar 2.23 menunjukkan ilustrasi dari mekanisme inferensi *Fuzzy*. Keluaran tegas hasil proses inferensi *Fuzzy* Mamdani dapat diperoleh dengan menerapkan operasi *max-min* atau *max-product*. Pada ilustrasi

tersebut keluaran tegas hasil inferensi *Fuzzy* didapat dengan menerapkan operasi *max-min* dan metode defuzzifikasi yang digunakan adalah metode *Center of Area*.



Gambar 2.23 Inferensi Fuzzy Mamdani

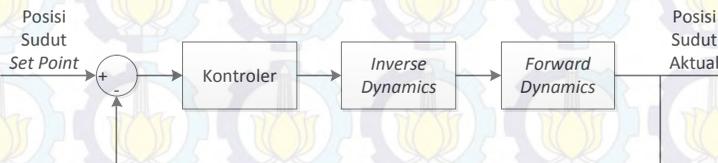
BAB III PERANCANGAN SISTEM

Bab ini berisi pembahasan yang berkaitan dengan perancangan sistem robot manipulator secara mendetail meliputi perancangan parameter *Denavit-Hartenberg*, perancangan gerak kinematik, perancangan model dinamik, perancangan kontroler dan perancangan simulasi melalui perangkat lunak.

Untuk mengetahui tipe konfigurasi dari robot dengan lebih jelas, dalam Bab ini juga akan dibahas mengenai beberapa jenis robot yang merupakan bentuk awal yang mendasari untuk membentuk model dinamik dari *plant* sebenarnya, yaitu robot *two-link planar 2-DOF* dan robot *articulated elbow 3-DOF*. Pada akhirnya akan dijelaskan perancangan robot PUMA-560 yang merupakan *plant* utama dalam Tugas Akhir ini.

3.1 Perancangan Blok Diagram Kontrol

Untuk memudahkan dalam perancangan pengaturan robot manipulator perlu dirancang sebuah blok diagram sistem. Gambar 3.1 menunjukkan blok diagram kontrol robot manipulator yang merepresentasikan sistem secara umum.



Gambar 3.1 Blok Diagram Sistem Kontrol Robot Manipulator

3.2 Perancangan Parameter *Denavit-Hartenberg*

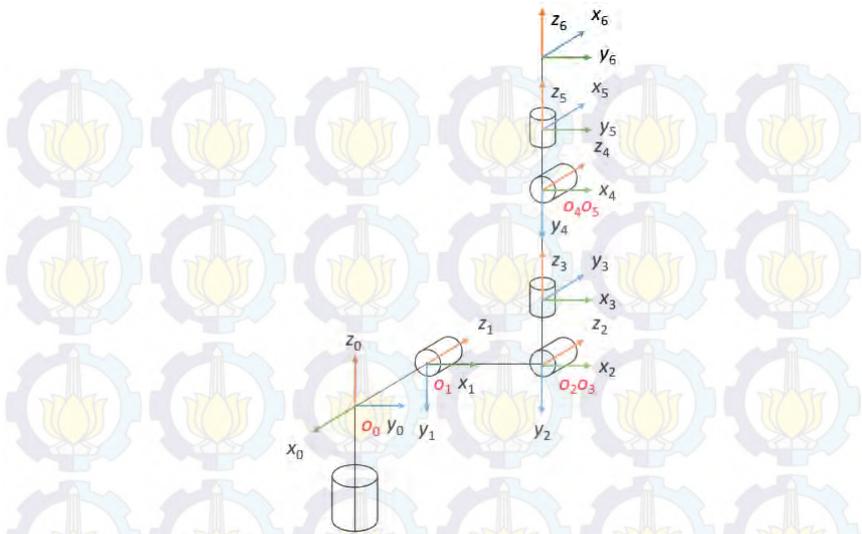
Gambar 3.2 menunjukkan bentuk fisik dari robot PUMA-560 yang akan dicari nilai parameter dari setiap *link-link* robot.



Gambar 3.2 Robot Manipulator PUMA-560

Berdasarkan kaidah *Denavit-Hartenberg* terdapat beberapa prosedur yang harus dilakukan dalam menempatkan *frame*. Dari penempatan *frame* tersebut akan didapatkan parameter setiap *link* sehingga robot dapat digambarkan secara kinematik.

Sesuai dengan algoritma yang telah dijelaskan pada Bab II, maka didapatkan penempatan *frame* robot manipulator PUMA-560. Gambar 3.3 menunjukkan prosedur akhir dari penempatan *frame* robot manipulator.



Gambar 3.3 Penetapan *End-Effector Frame* pada $o_n x_n y_n z_n$

Setelah penetapan *frame* akhir diperoleh, selanjutnya membuat data parameter *link* dari robot manipulator. Tabel 3.1 merupakan data parameter tiap *link* yang didapatkan berdasarkan penetapan *frame*.

Tabel 3.1 Parameter *Link* untuk Robot Manipulator PUMA-560

<i>Link ke-i</i>	a_i (mm)	α_i (deg)	d_i (mm)	θ_i (deg)
1	0	-90	0	θ_1^*
2	431,8	0	149,09	θ_2^*
3	-20,32	90	0	θ_3^*
4	0	-90	433,07	θ_4^*
5	0	90	0	θ_5^*
6	0	0	56,25	θ_6^*

Berdasarkan [3], robot *two-link planar* dan *articulated elbow* memiliki data parameter *link* sebagai berikut.

Tabel 3.2 Parameter *Link* untuk *Two-Link Planar*

Link ke- <i>i</i>	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1^*
2	a_2	0	0	θ_2^*

Tabel 3.3 Parameter *Link* untuk *Articulated Elbow*

Link ke- <i>i</i>	a_i	α_i	d_i	θ_i
1	0	90	0	θ_1^*
2	a_2	0	0	θ_2^*
3	a_3	0	0	θ_3^*

3.3 Perancangan Kinematik

Perancangan kinematik ini membahas bagaimana memperoleh persamaan *forward kinematics* dari setiap robot manipulator. Perancangan ini digunakan untuk menentukan posisi dan orientasi *end-effector* dalam ruang kartesian. Berdasarkan [5], persamaan transformasi homogen adalah sebagai berikut

$${}^i T = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} c_{\alpha_i} & s_{\theta_i} s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i} c_{\alpha_i} & -c_{\theta_i} s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_i & P_i \\ 0 & 1 \end{bmatrix} \quad (3.1)$$

Dengan menggunakan Persamaan (3.1), secara berurutan diperoleh transformasi homogen dari robot *two-link planar*, robot *articulated elbow* dan robot PUMA-560.

Berikut ini transformasi homogen dari robot *two-link planar*.

$$\begin{aligned}
 {}^1_0T &= \begin{bmatrix} c_1 & -s_1 & 0 & a_1c_1 \\ s_1 & c_1 & 0 & a_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^2_1T &= \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)
 \end{aligned}$$

Berikut ini merupakan komposisi transformasi homogen dari robot *two-link planar* berupa matriks 2_0T .

$$\begin{aligned}
 {}^2_0T &= {}^1_0T \cdot {}^2_1T \\
 &= \begin{bmatrix} c_1 & -s_1 & 0 & a_1c_1 \\ s_1 & c_1 & 0 & a_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3) \\
 &= \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1c_1 + a_2c_{12} \\ s_{12} & c_{12} & 0 & a_1s_1 + a_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

Berikut ini transformasi homogen dari robot *articulated elbow*.

$${}^1_0T = \begin{bmatrix} c_1 & 0 & s_1 & a_1c_1 \\ s_1 & 0 & -c_1 & a_1s_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned}
 {}^2_1T &= \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^3_2T &= \begin{bmatrix} c_3 & -s_3 & 0 & a_3c_3 \\ s_3 & c_3 & 0 & a_3s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.4)
 \end{aligned}$$

Berikut ini merupakan komposisi transformasi homogen dari robot *articulated elbow* berupa matriks 3_0T .

$$\begin{aligned}
 {}^3_0T &= {}^1_0T \cdot {}^2_1T \cdot {}^3_2T \\
 &= \begin{bmatrix} c_1 & 0 & s_1 & a_1c_1 \\ s_1 & 0 & -c_1 & a_1s_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_3 & -s_3 & 0 & a_3c_3 \\ s_3 & c_3 & 0 & a_3s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} c_1c_{23} & -c_1s_{23} & s_1 & c_1(a_2c_2 + a_3c_{23}) \\ s_1c_{23} & -s_1s_{23} & -c_1 & s_1(a_2c_2 + a_3c_{23}) \\ s_{23} & c_{23} & 0 & a_2s_2 + a_3s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.5)
 \end{aligned}$$

Berikut ini transformasi homogen dari robot PUMA-560.

$${}^1_0T = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned}
{}^2_1T &= \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
{}^3_2T &= \begin{bmatrix} c_3 & 0 & s_3 & a_3c_3 \\ s_3 & 0 & -c_3 & a_3s_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
{}^4_3T &= \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
{}^5_4T &= \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
{}^6_5T &= \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.6}
\end{aligned}$$

Berikut ini merupakan komposisi transformasi homogen dari robot *articulated elbow* berupa matriks 6_0T .

$$\begin{aligned}
{}^6_0T &= {}^1_0T {}^2_1T {}^3_2T {}^4_3T {}^5_4T {}^6_5T \\
&= \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_3 & 0 & s_3 & a_3c_3 \\ s_3 & 0 & -c_3 & a_3s_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}$$

$$\begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

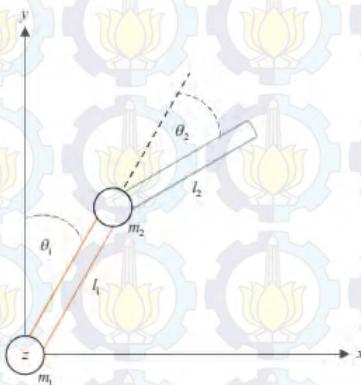
(3.7)

3.4 Perancangan Model Dinamik

Pada perancangan ini akan dibahas perhitungan untuk menentukan model dinamik dari robot manipulator. Perhitungan sangat berhubungan dengan besarnya energi kinetik dan energi potensial yang bekerja pada *link-link* robot manipulator ketika bergerak.

Lebih dalam lagi perancangan ini akan membahas bagaimana persamaan mengenai hukum mekanika dinamik yang berkaitan dengan momen inersia dan gaya-gaya yang bekerja pada robot manipulator, sampai akhirnya didapatkan besar torsi dari robot manipulator sehingga nilai torsi yang dihasilkan tersebut nantinya akan dikendalikan oleh kontroler yang akan dibahas pada Subbab berikutnya.

Gambar 3.4 menunjukkan bentuk sederhana dari robot *two-link planar* yang digunakan untuk mencari model dinamik.



Gambar 3.4 Bentuk Geometris Robot *Two-Link Planar*

Dengan menggunakan bentuk geometris pada Gambar 3.4, dilakukan perancangan dinamik untuk robot *two-link planar*. Berdasarkan

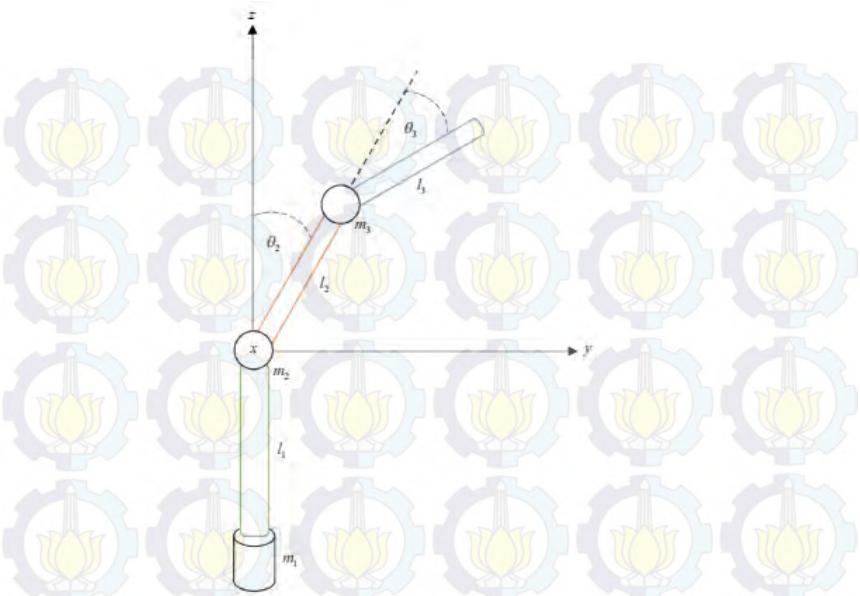
penurunan model dinamik yang dijelaskan pada Lampiran A.1 torsi dapat dijabarkan menjadi beberapa representasi vektor/matriks seperti berikut

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \quad (3.8)$$

Dari Persamaan 3.8 diketahui bahwa torsi robot manipulator memiliki vektor h yang merupakan inersia, vektor c yang merupakan gaya sentrifugal dan *coriolis*, serta vektor g yang merupakan gaya gravitasi. Berikut ini penjabaran lebih detil matriks-matriks dari Persamaan 3.8

$$\begin{aligned} h_{11} &= (m_1 + m_2)l_1^2 + m_2l_2^2\dot{\theta}_1 + 2m_2l_1l_2 \cos \theta_2 \\ h_{12} &= m_2l_2^2 + m_2l_1l_2 \cos \theta_2 \\ h_{21} &= m_2(l_2^2 + l_1l_2 \cos \theta_2) \\ h_{22} &= m_2l_2^2 \\ c_{11} &= -2m_2l_1l_2 \sin \theta_2 \dot{\theta}_2 \\ c_{12} &= -m_2l_1l_2 \sin \theta_2 \dot{\theta}_2 \\ c_{21} &= m_2l_1l_2 \sin \theta_2 \dot{\theta}_1 \\ c_{22} &= 0 \\ g_1 &= -m_1gl_1 \sin \theta_1 - m_2gl_1 \sin \theta_1 - m_2gl_2 \sin(\theta_1 + \theta_2) \\ g_2 &= -m_2gl_2 \sin(\theta_1 + \theta_2) \end{aligned} \quad (3.9)$$

Selanjutnya perancangan dinamik untuk robot *articulated elbow*. Secara geometris Gambar 3.5 menunjukkan bentuk sederhana dari robot *articulated elbow*.



Gambar 3.5 Bentuk Geometris Robot *Articulated Elbow*

Berdasarkan penurunan model dinamik yang dijelaskan pada Lampiran A.2 dapat dijabarkan menjadi beberapa representasi vektor/matriks seperti berikut.

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \\ \dot{\theta}_3^2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} \quad (3.10)$$

Berikut ini merupakan penjabaran detail matriks inersia, matriks sentrifugal dan *coriolis*, serta matriks gravitasi dari Persamaan 3.10.

$$\begin{aligned} h_{11} &= m_2 l_2^2 S_2^2 + m_3 (l_2 S_2 + l_3 S_{23})^2 \\ h_{12} &= 0 \\ h_{13} &= 0 \end{aligned}$$

$$h_{21} = 0$$

$$h_{22} = m_2 l_2^2 + m_3 (l_2^2 + 2l_2 l_3 C_3 + l_3^2)$$

$$h_{23} = m_3 l_3 (l_2 C_3 + l_3)$$

$$h_{31} = 0$$

$$h_{32} = m_3 l_3 (l_2 C_3 + l_3)$$

$$h_{33} = m_3 l_3^2$$

$$c_{11} = 2 \left[m_2 l_2^2 S_2 C_2 + m_3 (l_2 S_2 + l_3 S_{23}) (l_2 C_2 + l_3 C_{23}) \right] \dot{\theta}_2 \\ + 2m_3 l_3 C_{23} (l_2 S_2 + l_3 S_{23}) \dot{\theta}_3$$

$$c_{12} = 0$$

$$c_{13} = 0$$

$$c_{21} = - \left[m_2 l_2^2 S_2 C_2 + m_3 (l_2 S_2 + l_3 S_{23}) (l_2 C_2 + l_3 C_{23}) \right] \dot{\theta}_1$$

$$c_{22} = -2m_3 l_2 l_3 S_3 \dot{\theta}_3$$

$$c_{23} = -m_3 l_2 l_3 S_3 \dot{\theta}_3$$

$$c_{31} = -m_3 (l_2 S_2 + l_3 S_{23}) l_3 C_{23} \dot{\theta}_1$$

$$c_{32} = m_3 l_2 l_3 S_3 \dot{\theta}_2$$

$$c_{33} = 0$$

$$g_1 = 0$$

$$g_2 = -m_2 g l_2 S_2 - m_3 g (l_2 S_2 + l_3 S_{23})$$

$$g_3 = -m_3 g l_3 S_{23}$$

(3.11)

Berdasarkan Amstrong [11], model dinamik yang digunakan untuk mengatur posisi robot manipulator PUMA-560 didapatkan dari penurunan persamaan *lagrange* yang bergantung dari tiga *link* pertama dari total enam *link*. Secara lengkap dan detail model dinamik robot manipulator PUMA-560 dijelaskan pada Lampiran A.3.

Berdasar pada pembelajaran yang dilakukan oleh Amstrong [11], Robot PUMA-560 memiliki parameter-parameter yang digunakan dalam pemodelan dinamik. Tabel 3.4 dan Tabel 3.5 menunjukkan informasi

parameter-parameter yang dimiliki robot PUMA-560 berupa konstanta inersia dan konstanta gravitasi.

Tabel 3.4 Konstanta Inersia

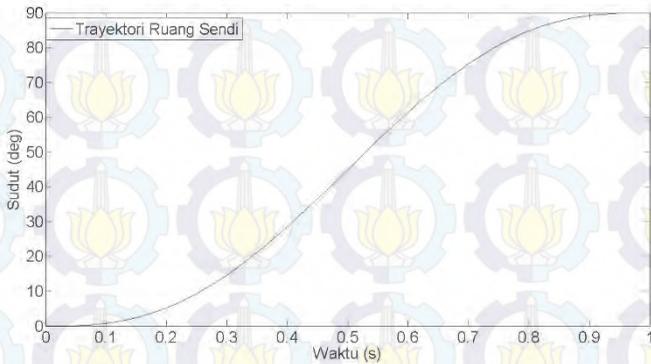
Konstanta Inersia ($Kg.m^2$)	
$I_1 = 1,43 \pm 0,05$	$I_2 = 1,75 \pm 0,07$
$I_1 = 1,38 \pm 0,05$	$I_4 = 0,69 \pm 0,02$
$I_5 = 0,372 \pm 0,031$	$I_6 = 0,333 \pm 0,016$
$I_7 = 0,298 \pm 0,029$	$I_8 = -0,134 \pm 0,014$
$I_9 = 0,0238 \pm 0,012$	$I_{10} = -0,0213 \pm 0,0022$
$I_{11} = -0,0142 \pm 0,007$	$I_{12} = -0,011 \pm 0,0011$
$I_{13} = 0,00379 \pm 0,0009$	$I_{14} = 0,00164 \pm 0,00007$
$I_{15} = 0,00125 \pm 0,0003$	$I_{16} = 0,00124 \pm 0,0003$
$I_{17} = 0,000642 \pm 0,0003$	$I_{18} = 0,000431 \pm 0,00013$
$I_{19} = 0,0003 \pm 0,0014$	$I_{20} = -0,000202 \pm 0,0008$
$I_{21} = -0,0001 \pm 0,0006$	$I_{22} = -0,000058 \pm 0,00001$
$I_{23} = 0,00004 \pm 0,00002$	$I_{m1} = 1,14 \pm 0,27$
$I_{m2} = 4,71 \pm 0,54$	$I_{m3} = 0,827 \pm 0,093$
$I_{m4} = 0,2 \pm 0,016$	$I_{m5} = 0,179 \pm 0,014$
$I_{m6} = 0,193 \pm 0,016$	

Tabel 3.5 Konstanta Gravitasi

Konstanta Gravitasi ($N.m$)	
$g_1 = -37,2 \pm 0,5$	$g_4 = -8,44 \pm 0,2$
$g_2 = 1,02 \pm 0,5$	$g_5 = 0,249 \pm 0,025$
$g_3 = -0,0282 \pm 0,0056$	

3.5 Perancangan Trayektori untuk Gerakan *Point to Point*

Perancangan ini dilakukan untuk membuat lintasan *joint* dari titik ke titik yang membentuk lintasan kubik seperti Gambar 3.6.



Gambar 3.6 Lintasan Trayektori Kubik

Gambar 3.6 adalah fungsi polinomial waktu t dengan empat koefisien independen yang dapat dipilih untuk memenuhi batasan-batasan. Lintasan kubik memiliki bentuk persamaan berikut

$$q(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \quad (3.12)$$

Sehingga didapatkan persamaan kecepatan yang diinginkan seperti berikut

$$\dot{q}(t) = a_1 + 2a_2t + 3a_3t^2 + 4a_4t^3 + 5a_5t^4 \quad (3.13)$$

Dengan melakukan kombinasi dari Persamaan 3.12 dan 3.13, dengan enam batasan dihasilkan enam persamaan.

$$q_0 = a_0 + a_1t_0 + a_2t_0^2 + a_3t_0^3 + a_4t_0^4 + a_5t_0^5 \quad (3.14)$$

$$v_0 = a_1 + 2a_2t_0 + 3a_3t_0^2 + 4a_4t_0^3 + 5a_5t_0^4 \quad (3.15)$$

$$\alpha_0 = 2a_2 + 6a_3t_0 + 12a_4t_0^2 + 20a_5t_0^3 \quad (3.16)$$

$$q_f = a_0 + a_1t_f + a_2t_f^2 + a_3t_f^3 + a_4t_f^4 + a_5t_f^5 \quad (3.17)$$

$$v_f = a_1 + 2a_2t_f + 3a_3t_f^2 + 4a_4t_f^3 + 5a_5t_f^4 \quad (3.18)$$

$$\alpha_f = 2a_2 + 6a_3t_f + 12a_4t_f^2 + 20a_5t_f^3 \quad (3.19)$$

Di mana,

q_0 : posisi awal

q_f : posisi akhir

v_0 : kecepatan awal

v_f : kecepatan akhir

α_0 : Percepatan awal

α_f : Percepatan akhir

Pada trayektori dirancang untuk kecepatan awal dan akhir serta percepatan awal dan akhir dengan inisial nol.

3.6 Perancangan Kontroler

Pada Tugas Akhir ini dirancang kontroler *Proporsional Derivative* (PD), kontroler *Fuzzy* dan kontroler *Computed Torque Controller* (CTC). Kontroler PD digunakan untuk memperbaiki respon *steady state* dari sistem kontrol robot manipulator, dirancang melalui pendekatan karakteristik respon sistem. Lalu kontroler *Fuzzy* digunakan sebagai koreksi terhadap *gain* kontroler PD yang digunakan untuk memperbaiki respon transien dari sistem kontrol robot manipulator. Sedangkan kontroler CTC merupakan kontroler yang digunakan dalam pemodelan dinamik robot manipulator yang menggunakan besaran torsi sebagai metode kendali robot manipulator.

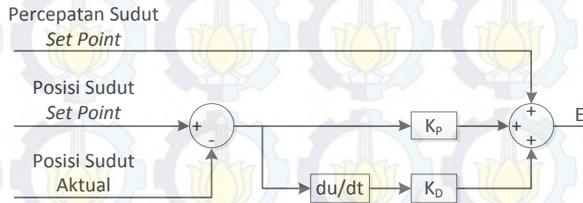
3.6.1 Desain Kontroler PD CTC

Kontroler PD dirancang untuk memperbaiki respon *steady state* dari sistem kontrol manipulator, digunakan untuk mengolah data *error* dan perubahan *error* dari posisi sudut yang diinginkan terhadap posisi sudut aktual.

Dalam perancangan, model dinamik robot manipulator dikombinasikan dengan kontroler PD sehingga didapatkan persamaan untuk kontroler PD CTC seperti berikut

$$\tau = M(\theta)(\ddot{\theta}_d + K_p e + K_d \dot{e}) + B(\theta)(\dot{\theta}\dot{\theta}) + C(\theta)\dot{\theta}^2 + G(\theta) \quad (3.20)$$

Sehingga didapatkan struktur kontroler PD CTC seperti yang ditunjukkan pada Gambar 3.7.



Gambar 3.7 Struktur Kontroler PD CTC

Untuk mendapatkan nilai *gain* K_P dan K_D , kontroler PD didesain dengan menerapkan metode *tuning Ziegler-Nichols* menggunakan parameter yang sudah ditentukan. Tabel 3.6 menunjukkan nilai parameter yang bisa digunakan dengan beberapa kemungkinan kontroler yang bisa dipilih.

Tabel 3.6 Metode *Ziegler-Nichols* [12]

Type Kontrol	K_P	K_I	K_D
P	$0,5 K_U$	-	-
PI	$0,45 K_U$	$1,2 K_P/T_U$	-
PD	$0,8 K_U$	-	$K_P T_U/8$
PID	$0,6 K_U$	$2 K_P/T_U$	$K_P T_U/8$

Di mana K_U merupakan *gain ultimate* dan T_U merupakan periode (*time ultimate*). Nilai K_U ditentukan dengan memasukkan nilai secara sembarang sehingga menghasilkan respon *output* yang memiliki tingkat osilasi tertentu sehingga nilai T_U bisa didapatkan dari respon tersebut.

Berdasarkan Tabel 3.6 bahwa untuk tipe kontroler PD nilai K_P dan K_D bisa didapatkan dengan besar $K_P = 0,8K_U$ dan $K_D = K_P T_U/8$. Untuk mendapatkan nilai *gain* K_P dan K_D yang diinginkan, dilakukan pengujian dengan $K_U = 100$. Selanjutnya nilai K_U tersebut akan menghasilkan osilasi pada respon *output* dan didapatkan nilai T_U dari respon *output* sehingga nilai *gain* K_P dan K_D bisa didapatkan melalui perhitungan sesuai dengan Tabel 3.6.

3.6.2 Kontroler PD Fuzzy CTC

Kontroler *Fuzzy* yang digunakan adalah kontroler *Fuzzy* tipe *Mamdani*. Tipe *Mamdani* digunakan karena tipe ini memiliki masukan yang sifatnya *human input* sehingga sangat sesuai dengan sistem kontrol robot manipulator yang nonlinier. Kontroler ini digunakan untuk memperbaiki respon transien sebagai koreksi dari respon sinyal kontrol dari kontroler PD.

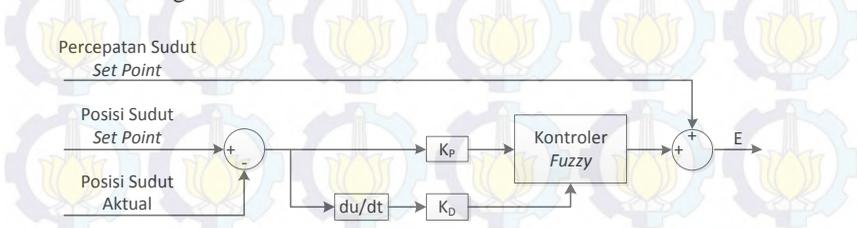
Dalam perancangan kontroler ini digunakan fungsi *toolbox Fuzzy* yang diintegrasikan dengan Simulink dalam *software* Matlab. Toolbox tersebut memberikan fasilitas *Graphical User Interface* (GUI) untuk mempermudah dalam membangun sistem *Fuzzy*.

Terdapat lima GUI yang digunakan untuk merancang kontroler *Fuzzy*, yaitu

- Fuzzy Inference System* (FIS) Editor
- Membership Function* Editor
- Rule* Editor
- Rule* Viewer
- Surface* Viewer

Dengan menggunakan tiga fungsi GUI yang pertama, dapat dirancang dan dibuat data tentang sistem *Fuzzy* yang diinginkan.

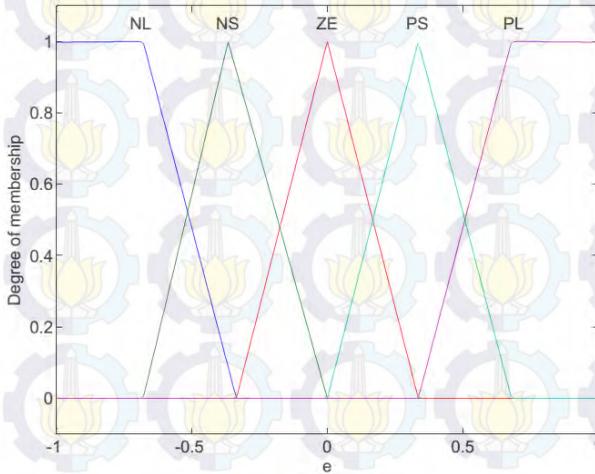
Perancangan kontroler *Fuzzy* dilakukan dengan mempertimbangkan dua sinyal masukan berupa *error* dan *delta error* serta satu *output* berupa sinyal kontrol. Sinyal *error* dan *delta error* didapatkan dari *gain* kontroler PD yang nilainya dibuat sedemikian lebih sederhana sehingga sinyal *input* baik *error* maupun *delta error* memiliki *range* daerah *Fuzzy* (-1,+1). Lalu sinyal *output* dirancang dengan *range* daerah *Fuzzy* (-2,+2). Nilai *gain* K_P dan K_D pada sistem kontroler *Fuzzy* berdasarkan data *tuning* kontroler PD, lalu untuk nilai *gain* sinyal kontrol (K_V) ditentukan dengan metode *tuning* manual. Gambar 3.8 menunjukkan struktur kontroler PD *Fuzzy* CTC yang akan dirancang.



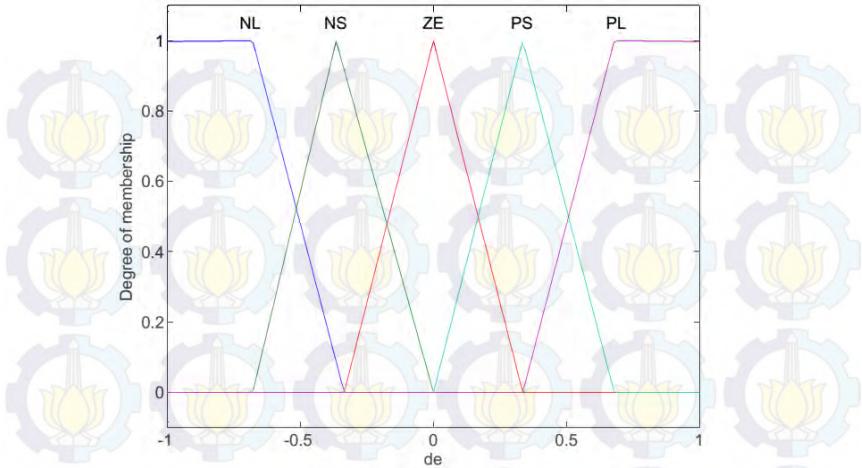
Gambar 3.8 Struktur Kontroler PD *Fuzzy* CTC

FIS *Editor* merupakan editor sistem inferensi *Fuzzy*, digunakan untuk mendesain jumlah *input* dan *output* yang digunakan dalam kontroler *Fuzzy*. Dalam FIS editor terdapat setingan metode untuk proses fuzifikasi, implikasi, agregasi dan defuzifikasi.

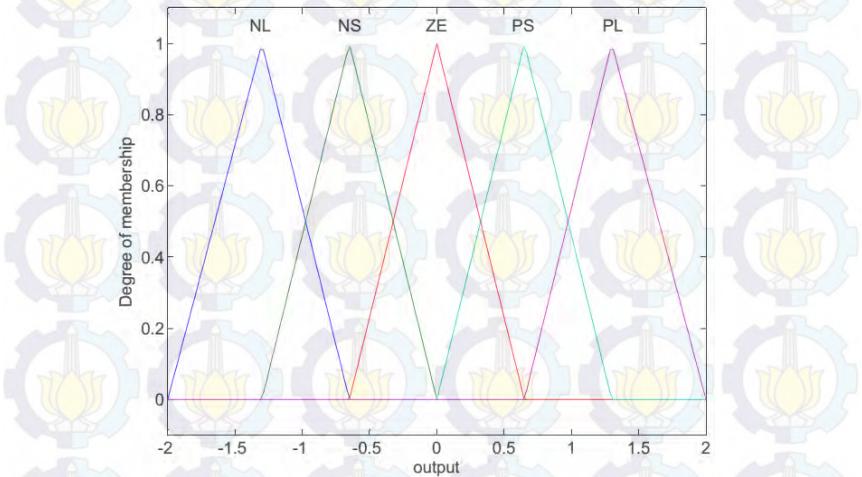
Membership Function Editor merupakan editor untuk mengatur fungsi keanggotaan untuk *input* dan *output*. Ada tiga fungsi keanggotaan, dua untuk *input* dan satu untuk *output*. Untuk *input* berupa *error* dan *delta error* serta untuk *output* berupa kontrol. Dalam perancangan fungsi keanggotaan dirancang dengan tipe segitiga (*trimf*) baik untuk *input* maupun *output* dengan masing-masing memiliki lima himpunan pendukung. Gambar 3.9-3.11 menunjukkan fungsi keanggotaan *input* dan *output* dalam perancangan kontroler *Fuzzy*.



Gambar 3.9 Fungsi Keanggotaan untuk *Input* berupa *Error*



Gambar 3.10 Fungsi Keanggotaan untuk *Input* berupa *Delta Error*



Gambar 3.11 Fungsi Keanggotaan untuk *Output* berupa Sinyal Kontrol

Dari fungsi keanggotaan yang dirancang pada *Membership Function Editor* dapat dibuat suatu aturan yang memungkinkan kondisi

antara satu premis dengan premis yang lain sehingga didapatkan sebuah konsekuen. *Rule Editor* merupakan fasilitas GUI yang digunakan untuk membuat aturan-aturan *Fuzzy* lebih mudah. Untuk memudahkan dalam membuat aturan tersebut Tabel 3.7 menunjukkan aturan-aturan *Fuzzy* dalam bentuk *Mack Vicar Whelan* dengan lima himpunan pendukung pada masing-masing *input*.

Tabel 3.7 Aturan Dasar *Mack Vicar Whelan* untuk Kontroler *Fuzzy*

<i>Output</i>		<i>Delta Error</i>				
		NL	NS	ZE	PS	PL
<i>Error</i>	NL	NL	NL	NS	NS	ZE
	NS	NL	NS	NS	ZE	PS
	ZE	NS	NS	ZE	PS	PS
	PS	NS	ZE	PS	PS	PL
	PL	ZE	PS	PS	PL	PL

Di mana deskripsi linguistik dari fungsi keanggotaan *input* dan *output* adalah *Negative Large* (NL), *Negative Small* (NS), *Zero* (ZE), *Positive Small* (PS) dan *Positive Large* (PL) sehingga didapatkan 25 buah kaidah yang berbentuk seperti berikut

$$IF x_1 \text{ is } A_1^k \text{ AND } x_2 \text{ is } A_2^k \text{ then } y^k \text{ is } B^k$$

Di mana x adalah *input* dan y adalah *output*.

Dari aturan dasar yang sudah dibuat dilakukan proses inferensi untuk menarik kesimpulan sinyal kontrol yang digunakan. Aturan inferensi menggunakan metode *Mamdani* yang menentukan nilai maksimum sinyal kontrol dari minimum *input sinyal error* dan *delta error* dengan persamaan berikut

$$\mu_B(y) = \max[\min[\mu_A(\text{input}(i)), \mu_A(\text{input}(j))]] \quad (3.21)$$

Agar sistem mendapatkan besaran yang *real* maka dilakukan defuzzifikasi untuk memetakan besaran dari himpunan *Fuzzy* ke dalam bentuk nilai *crisp*. Metode yang digunakan dalam defuzzifikasi adalah metode *centroid* (*center of gravity* atau *center of area*) dengan persamaan berikut.

$$z^* = \frac{\int z \mu_c(z) dz}{\int \mu_c(z)} \quad (3.22)$$

3.7 Perancangan Sistem pada Diagram Simulink

Hasil perancangan parameter DH, gerak kinematik, model dinamik dan kontroler pada Subbab sebelumnya disusun dan dibuat menjadi sebuah simulasi menggunakan *toolbox* Simulink pada *software* Matlab 2010a dan *robotics toolbox* Peter Corke versi 8.

Simulasi dengan diagram Simulink disusun dari beberapa blok diagram seperti blok *input*, blok dinamik, blok kinematik, blok kontroler, blok plot robot manipulator, dan blok pendukung lainnya.

Blok *input* berupa sinyal step yang merepresentasikan sinyal posisi sudut, turunan pertama yang merupakan representasi dari sinyal kecepatan sudut, dan juga turunan kedua yang merupakan representasi dari sinyal percepatan sudut dari robot manipulator.

Blok Matriks A serta Matriks B, C dan G merupakan model *inverse dynamic* dari robot manipulator. Blok tersebut berupa persamaan yang mengkonversi masukan berupa posisi sudut, kecepatan sudut dan percepatan sudut menjadi keluaran berupa torsi.

Blok kontroler berupa kontroler PD *Fuzzy* CTC yang mengendalikan masukan torsi dari Matriks A yang di dalamnya terdapat selisih *error* dan selisih perubahan *error* yang dikendalikan oleh kontroler PD. Selanjutnya keluaran dari kontroler PD tersebut akan digunakan sebagai masukan untuk kontroler *Fuzzy*. Kemudian keluaran dari kontroler berupa sinyal kontrol akan digabung dengan menjumlahkannya dengan sinyal percepatan sudut yang selanjutnya akan digunakan untuk menghitung besarnya torsi. Terdapat enam buah *joint* yang akan dikendalikan dengan tiap-tiap *joint* memiliki kontrolernya masing-masing.

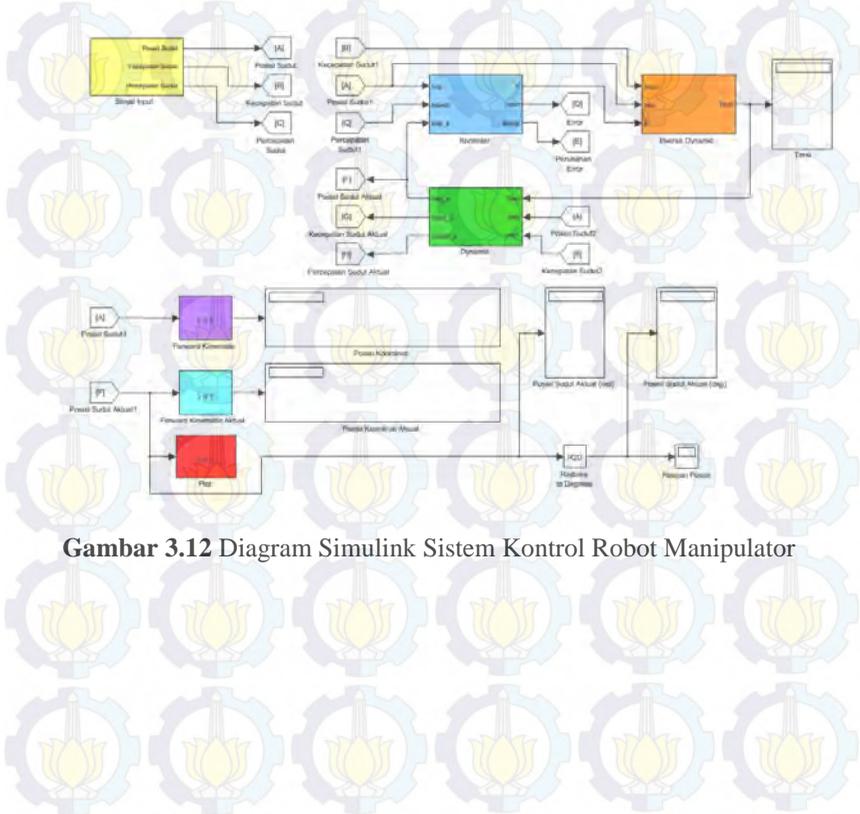
Blok dinamik merupakan model *forward dynamic* dari robot manipulator. Blok tersebut berisi persamaan yang mengkonversi masukan berupa torsi menjadi keluaran berupa percepatan. Di dalamnya terdapat fungsi *inverse* yang mengkonversi masukan torsi menjadi keluaran percepatan, selanjutnya digunakan fungsi integral untuk mendapatkan keluaran berupa posisi sudut.

Blok *forward kinematics* merupakan blok fungsi yang didapatkan dari *robotics toolbox* Peter Corke versi 8, digunakan untuk mengkonversi posisi sudut dalam ruang sendi menjadi posisi koordinat *end-effector*

dalam ruang kartesian. Terdapat dua blok *forward kinematics*, yang pertama untuk masukan posisi sudut yang diinginkan, dan yang lainnya untuk masukan posisi sudut aktual.

Blok plot merupakan blok fungsi khusus yang didapatkan dari *toolbox* Peter Corke versi 8. Digunakan untuk membuat animasi gerak robot secara grafis sehingga gerakan robot manipulator dapat digambarkan secara kinematik.

Setelah semua blok dirancang dan dibuat, kemudian blok-blok tersebut disusun dan dihubungkan menjadi satu kesatuan blok diagram sistem kontrol. Diagram Simulink sistem kontrol robot manipulator ditunjukkan dengan Gambar 3.12.



Gambar 3.12 Diagram Simulink Sistem Kontrol Robot Manipulator



--- Halaman ini sengaja dikosongkan ---

BAB IV PENGUJIAN DAN ANALISIS SISTEM

Pada Bab ini akan dibahas mengenai pengujian data antara respon masukan posisi sudut dalam ruang sendi yang diinginkan dengan respon keluaran posisi sudut dalam ruang sendi yang aktual. Kedua posisi sudut tersebut nantinya akan dibandingkan dan dianalisis lebih lanjut, di mana nilai keduanya akan menentukan posisi koordinat *end-effector* dari robot manipulator dalam ruang kartesian.

4.1 Pengujian Posisi Sudut Setiap *Joint*

Pengujian posisi sudut setiap *joint* perlu dilakukan untuk mengetahui perbandingan antara posisi sudut yang diinginkan dengan posisi sudut aktual sebelum diberikan kontroler dalam sistem kontrol robot manipulator. Pada pengujian ini ada dua masukan posisi sudut yang diberikan untuk enam *joint* dari robot manipulator. Sinyal *input* yang diberikan pada tiap-tiap *joint* berupa trayektori kubik. Tabel 4.1 dan Tabel 4.2 menunjukkan hasil pengujian posisi sudut setiap *joint* robot manipulator dengan masukan dua posisi sudut di mana q_x merupakan posisi robot manipulator ke- x , θ_d merupakan posisi sudut *joint* yang diinginkan, dan θ_a merupakan posisi sudut *joint* aktual.

Tabel 4.1 Hasil Pengujian Posisi Sudut *Joint* 1, *Joint* 2 dan *Joint* 3

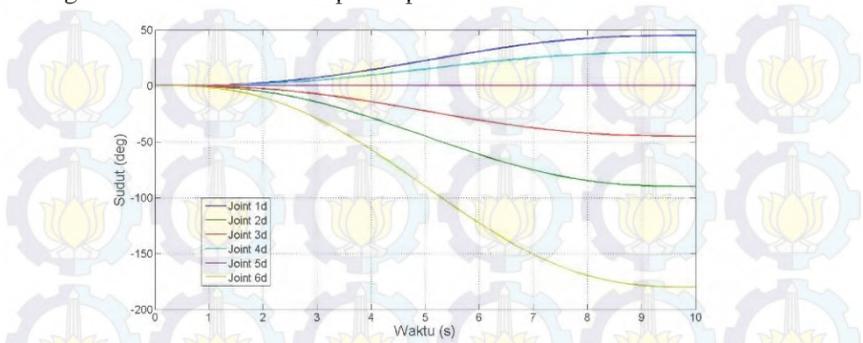
$q_0=0$	<i>Joint</i> 1		<i>Joint</i> 2		<i>Joint</i> 3	
	θ_d (deg)	θ_a (deg)	θ_d (deg)	θ_a (deg)	θ_d (deg)	θ_a (deg)
q_1	45	68,4	-90	-136,8	-45	-68,4
q_2	90	136,8	-45	-68,4	-30	-45,5

Tabel 4.2 Hasil Pengujian Posisi Sudut *Joint* 4, *Joint* 5 dan *Joint* 6

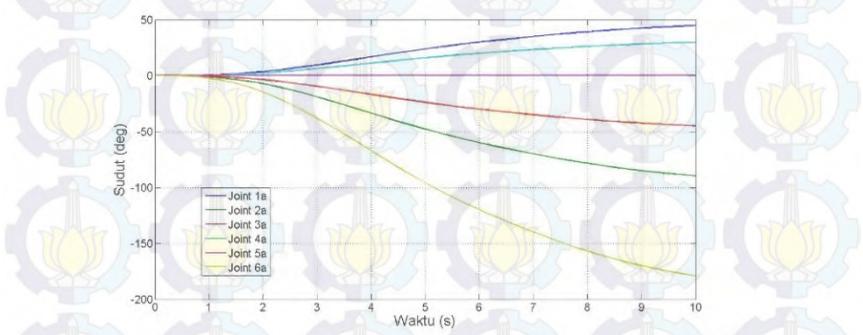
$q_0=0$	<i>Joint</i> 4		<i>Joint</i> 5		<i>Joint</i> 6	
	θ_d (deg)	θ_a (deg)	θ_d (deg)	θ_a (deg)	θ_d (deg)	θ_a (deg)
q_1	30	45,55	0	0	-180	-273,6
q_2	60	91,1	45	68,4	-90	-136,8

Untuk lebih jelasnya Gambar 4.1 dan 4.3 menunjukkan respon posisi *set point* serta Gambar 4.2 dan 4.4 menunjukkan respon posisi aktual enam

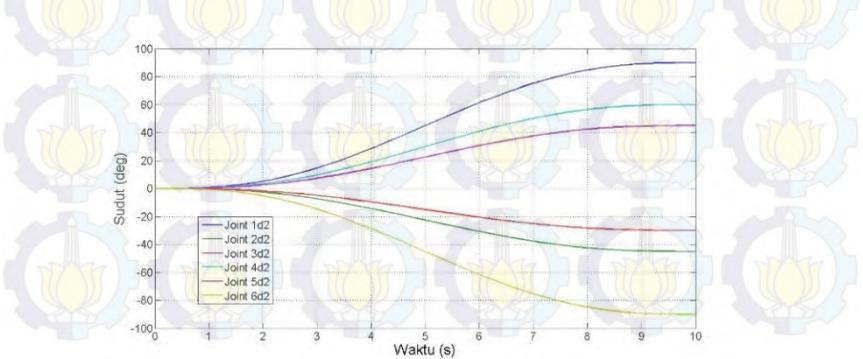
joint dalam sebuah grafik, di mana *Joint xd* merupakan posisi ke-x yang diinginkan dan *Joint xa* merupakan posisi ke-x aktual.



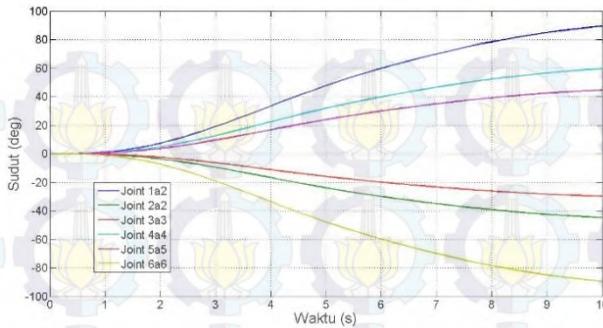
Gambar 4.1 Respon Posisi *Set Point* Pertama



Gambar 4.2 Respon Posisi Aktual Pertama

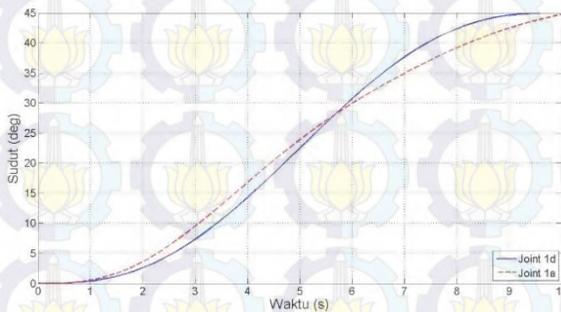


Gambar 4.3 Respon Posisi *Set Point* Kedua

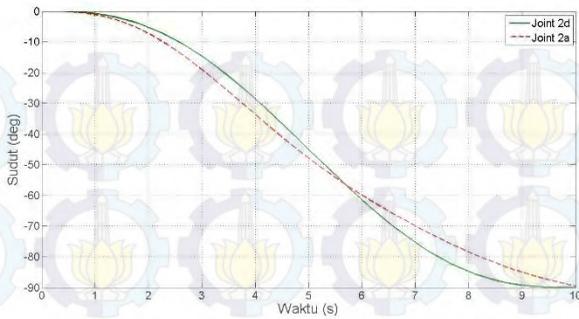


Gambar 4.4 Respon Posisi Aktual Kedua

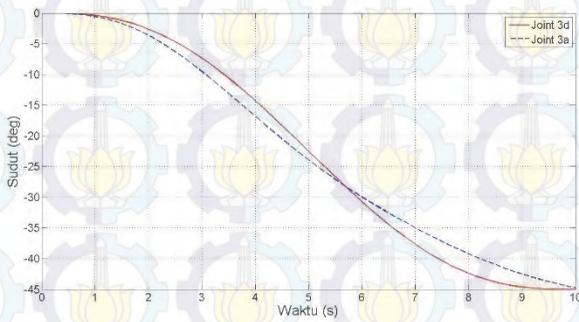
Dapat dilihat bahwa ada perbedaan antara Gambar 4.1 dan 4.2 serta Gambar 4.3 dan 4.4 di mana posisi aktual belum dapat mengikuti trayektori. Agar lebih jelas untuk mengetahui besarnya perbedaan antara nilai *set point* dan aktual, Gambar 4.5-4.10 menunjukkan respon enam *joint* yang membandingkan posisi *set point* dan aktual tiap-tiap *joint* untuk posisi pertama.



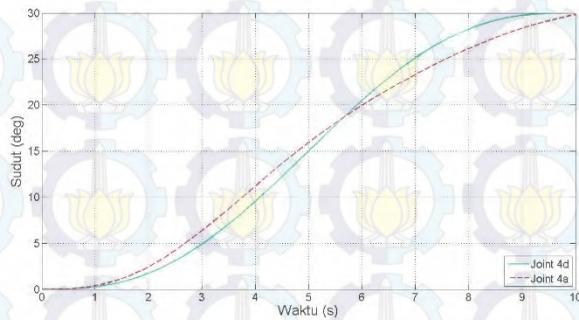
Gambar 4.5 Respon *Joint* ke-1 Posisi Pertama



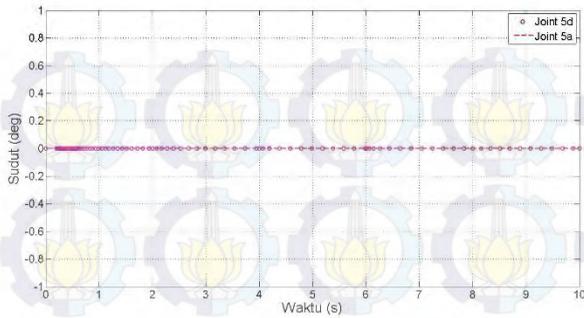
Gambar 4.6 Respon *Joint* ke-2 Posisi Pertama



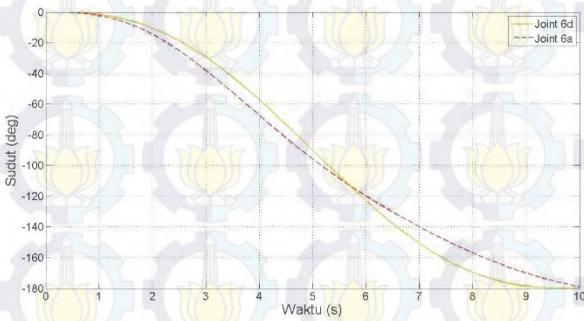
Gambar 4.7 Respon *Joint* ke-3 Posisi Pertama



Gambar 4.8 Respon *Joint* ke-4 Posisi Pertama

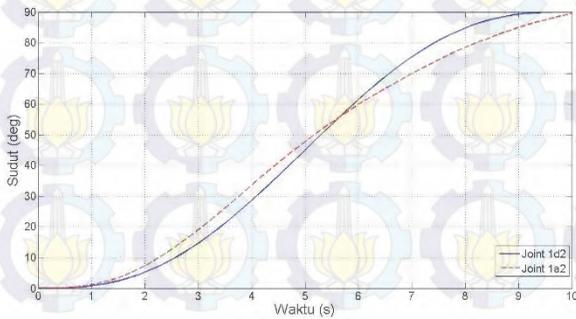


Gambar 4.9 Respon *Joint* ke-5 Posisi Pertama

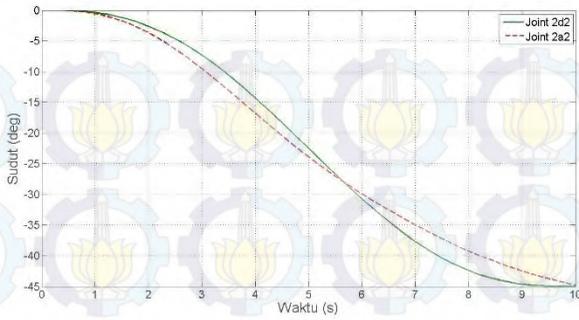


Gambar 4.10 Respon *Joint* ke-6 Posisi Pertama

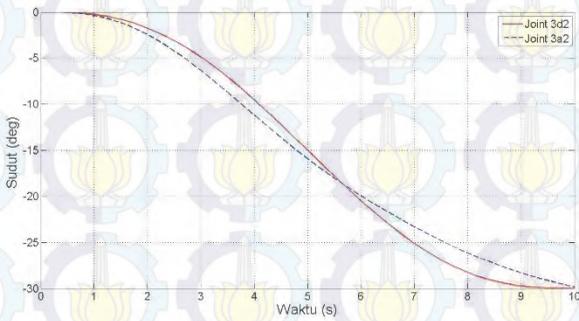
Lalu Gambar 4.11-4.16 menunjukkan respon enam *joint* untuk posisi kedua.



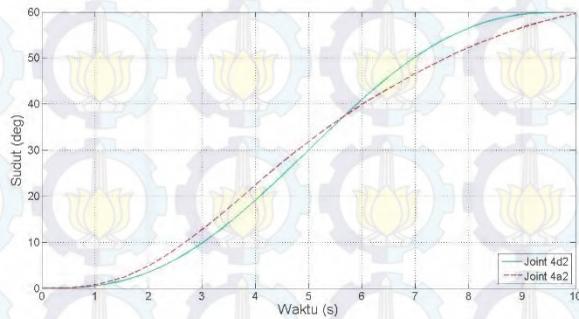
Gambar 4.11 Respon *Joint* ke-1 Posisi Kedua



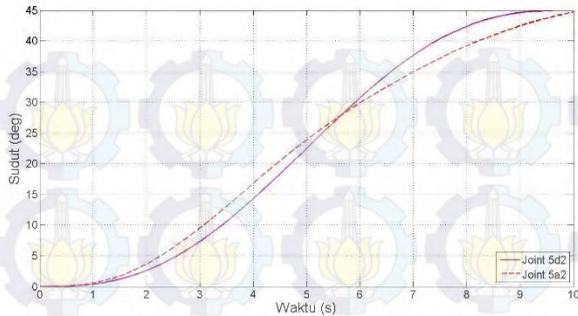
Gambar 4.12 Respon *Joint* ke-2 Posisi Kedua



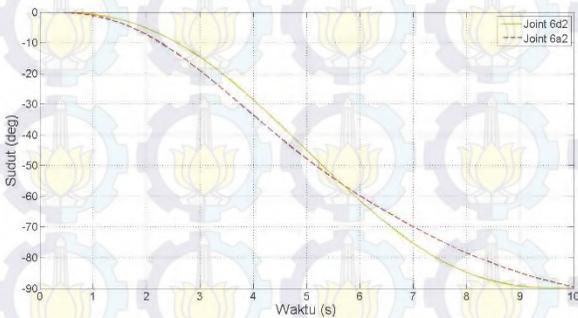
Gambar 4.13 Respon *Joint* ke-3 Posisi Kedua



Gambar 4.14 Respon *Joint* ke-4 Posisi Kedua



Gambar 4.15 Respon *Joint* ke-5 Posisi Kedua



Gambar 4.16 Respon *Joint* ke-6 Posisi Kedua

Dari respon posisi *joint* pada Gambar 4.5-4.16 didapatkan nilai RMSE antara nilai *set point* dan aktual, ditunjukkan pada Tabel 4.3.

Tabel 4.3 Nilai RMSE tanpa Kontroler

Nilai RMSE	Posisi 1 (q_1)	Posisi 2 (q_2)
<i>Joint 1</i>	1,14437	2,612463
<i>Joint 2</i>	2,288741	1,306231
<i>Joint 3</i>	1,14437	0,8708
<i>Joint 4</i>	0,761025	1,7416
<i>Joint 5</i>	0	1,306231
<i>Joint 6</i>	4,577482	2,612463

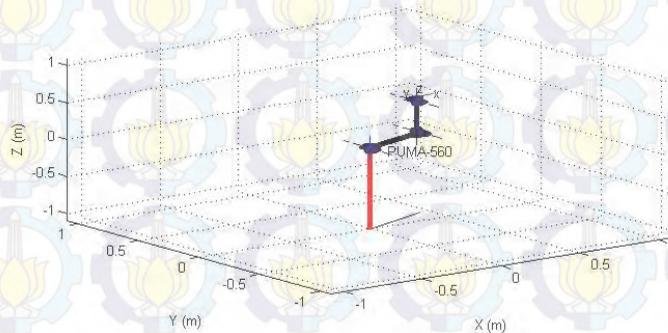
4.2 Pengujian Posisi *End-effector* Robot Manipulator

Pengujian ini dimaksudkan untuk mengetahui koordinat posisi dari *end-effector* robot manipulator pada awal sebelum diberikan kontroler dalam sistem robot manipulator. Koordinat dari posisi yang diinginkan dibandingkan dengan koordinat dari posisi aktual. Koordinat posisi didapatkan dengan menggunakan persamaan transformasi homogen dalam *forward kinematics*. Untuk besarnya posisi sudut *set point* dan posisi aktual sama dengan Subbab 4.1. Tabel 4.4 menunjukkan posisi koordinat sumbu x , y dan z dari *end-effector*.

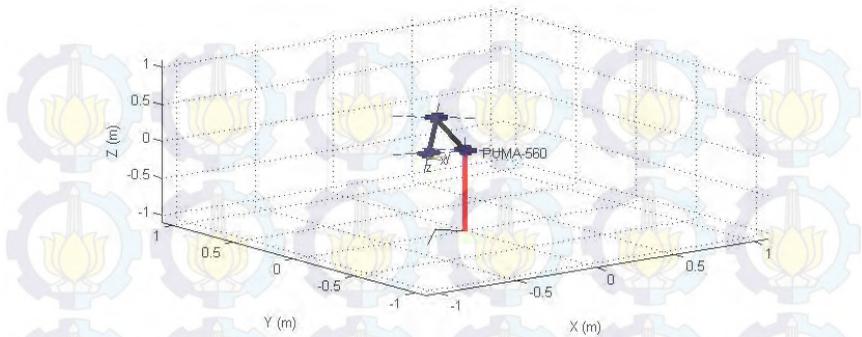
Tabel 4.4 Koordinat Posisi *End-effector* Robot Manipulator

Koordinat		x_i	y_i	z_i
q_1	<i>Set Point</i>	-0,3399	-0,1291	0,07144
	Aktual	-0,3412	-0,1282	-0,07613
q_2	<i>Set Point</i>	-0,1835	-0,1515	0,4273
	Aktual	-0,1846	-0,1477	0,4295

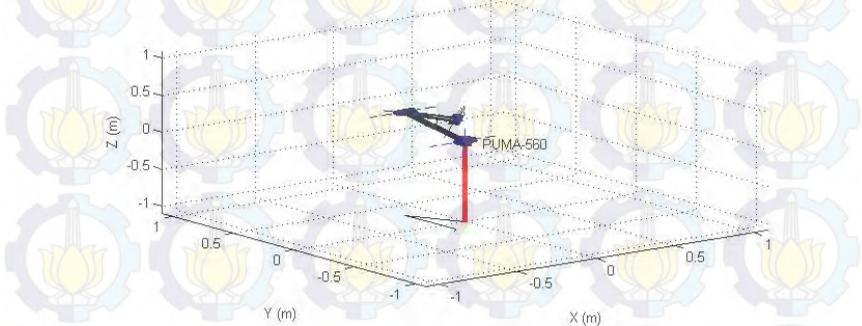
Gambar 4.17 menunjukkan posisi awal robot manipulator dan Gambar 4.18-4.19 menunjukkan posisi akhir robot manipulator dalam bentuk *plot* gambar.



Gambar 4.17 Posisi Awal Robot Manipulator



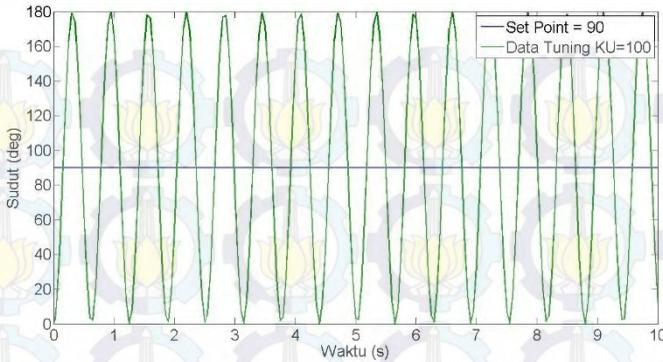
Gambar 4.18 Posisi Akhir Robot Manipulator Pertama



Gambar 4.19 Posisi Akhir Robot Manipulator Kedua

4.3 Analisis Sistem dengan Kontroler PD CTC

Dari persamaan berdasarkan Tabel 3.1 didapatkan beberapa nilai T_U yang akan menentukan nilai *gain* K_P dan K_D . Sinyal *step* yang diberikan sebesar 90° . Gambar 4.20 menunjukkan grafik respon osilasi dengan data *tuning* dengan $K_U = 100$ sehingga didapatkan $T_U = 0,625$.



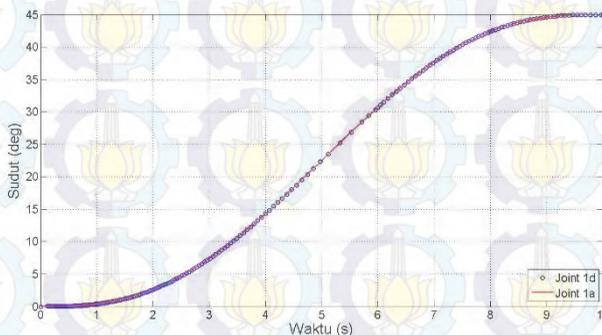
Gambar 4.20 Respon *Tuning* Kontroler PD

Tabel 4.5 merupakan data hasil *tuning gain* K_P dan K_D yang didapatkan melalui metode *ziegler-nichols*.

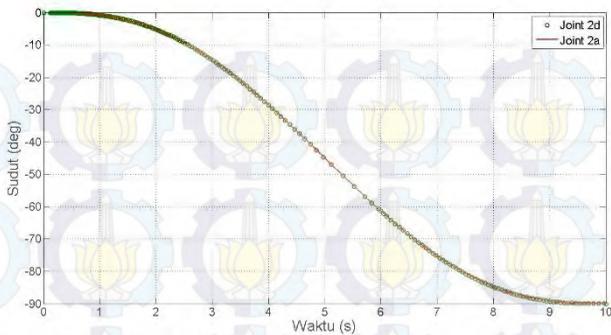
Tabel 4.5 Data Hasil *Tuning Gain* kontroler PD

Data Tuning	K_U	T_U	K_P	K_D
Set Point = 90	100	0,625	80	6,25

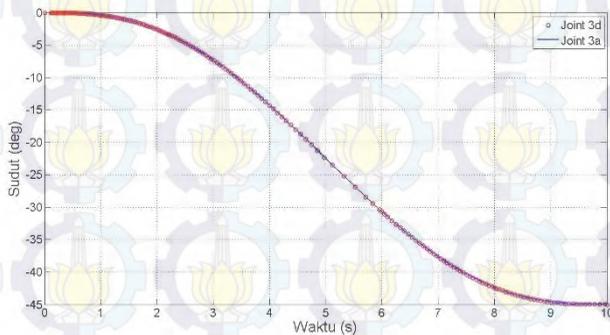
Nilai K_P dan K_D yang didapatkan, digunakan dalam sistem kontrol robot manipulator sehingga didapatkan respon tiap-tiap *joint* untuk posisi pertama yang ditunjukkan pada Gambar 4.21-4.26.



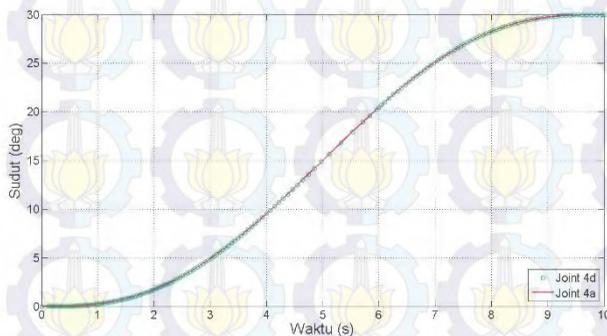
Gambar 4.21 Respon *Joint* ke-1 Posisi Pertama (PD CTC)



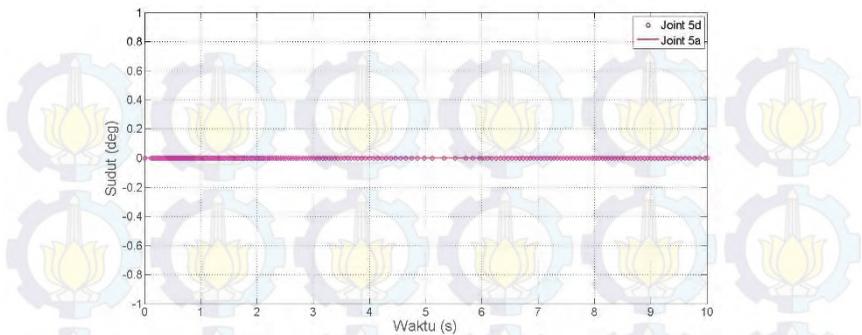
Gambar 4.22 Respon *Joint* ke-2 Posisi Pertama (PD CTC)



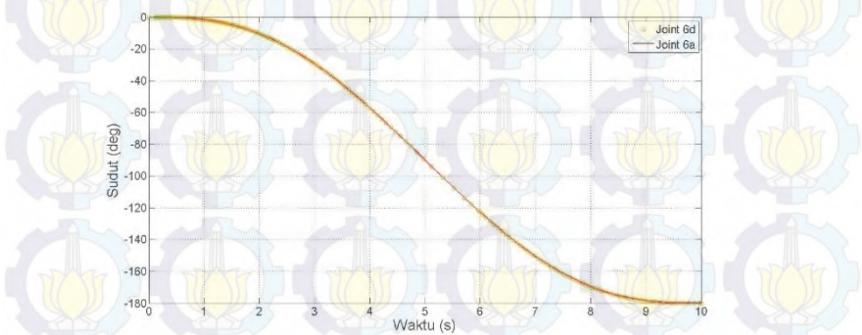
Gambar 4.23 Respon *Joint* ke-3 Posisi Pertama (PD CTC)



Gambar 4.24 Respon *Joint* ke-4 Posisi Pertama (PD CTC)

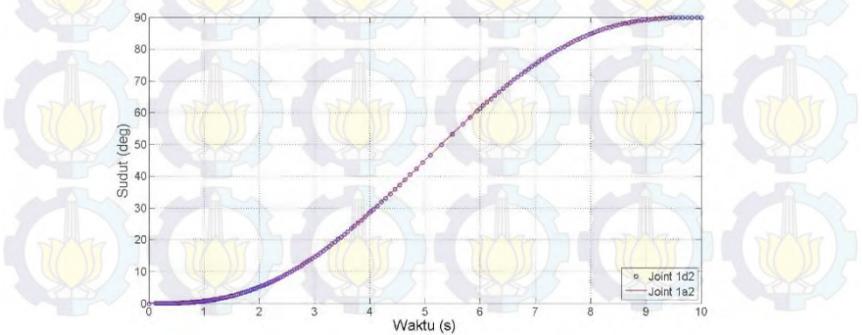


Gambar 4.25 Respon *Joint* ke-5 Posisi Pertama (PD CTC)

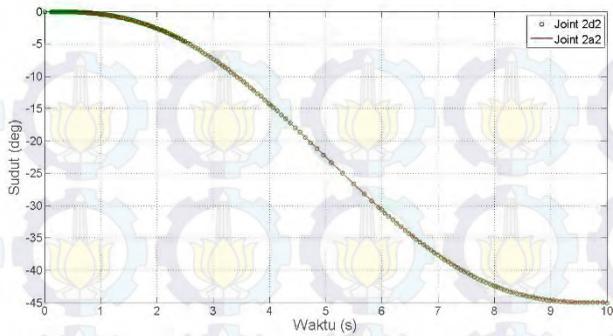


Gambar 4.26 Respon *Joint* ke-6 Posisi Pertama (PD CTC)

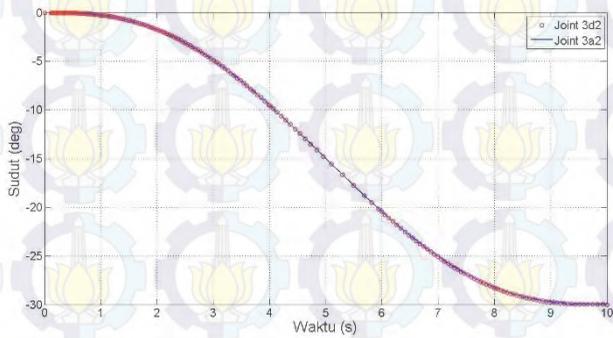
Selanjutnya respon tiap-tiap *joint* untuk posisi kedua sebagai berikut.



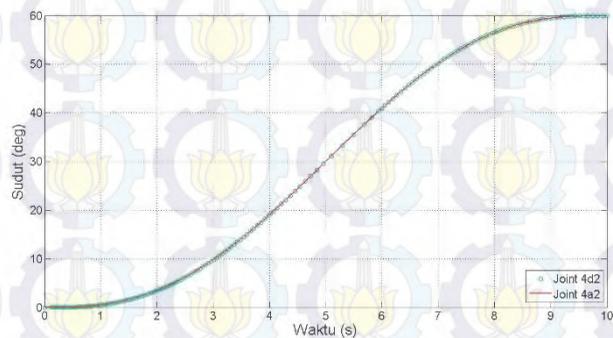
Gambar 4.27 Respon *Joint* ke-1 Posisi Kedua (PD CTC)



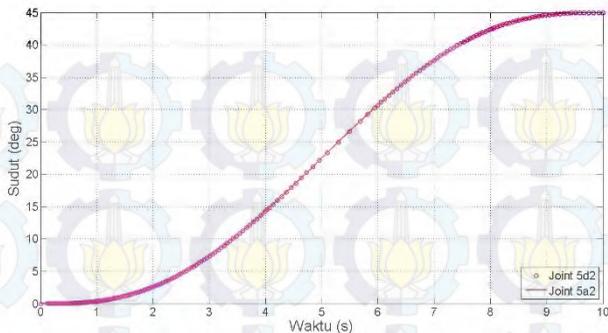
Gambar 4.28 Respon *Joint* ke-2 Posisi Kedua (PD CTC)



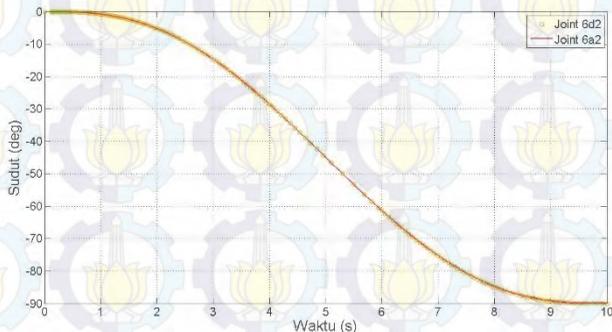
Gambar 4.29 Respon *Joint* ke-3 Posisi Kedua (PD CTC)



Gambar 4.30 Respon *Joint* ke-4 Posisi Kedua (PD CTC)



Gambar 4.31 Respon *Joint* ke-5 Posisi Kedua (PD CTC)



Gambar 4.32 Respon *Joint* ke-6 Posisi Kedua (PD CTC)

Dari respon posisi *joint* Gambar 4.21-4.32 didapatkan nilai RMSE antara nilai *set point* dan aktual, ditunjukkan pada Tabel 4.6.

Tabel 4.6 Nilai RMSE Kontroler PD CTC

Nilai RMSE	Posisi 1 (q_1)	Posisi 2 (q_2)
Joint 1	0,012926	0,025912
Joint 2	0,025852	0,012956
Joint 3	0,012926	0,008638
Joint 4	0,008619	0,017276
Joint 5	0	0,012956
Joint 6	0,051705	0,025912

Dengan posisi *end-effector* ditunjukkan pada Tabel 4.7 berikut.

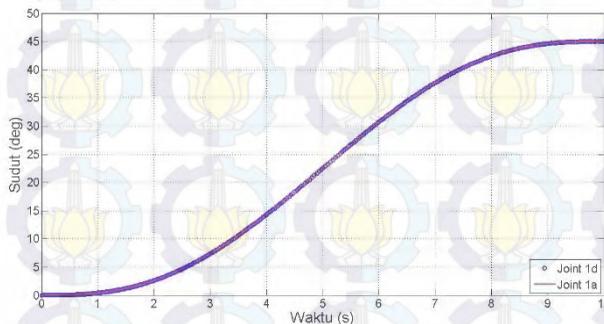
Tabel 4.7 Koordinat Posisi *End-effector* Robot Manipulator (PD CTC)

Koordinat		x_i	y_i	z_i
q_1	Set Point	-0,3399	-0,1291	0,07144
	Aktual	-0,34	-0,1291	0,07153
q_2	Set Point	-0,1835	-0,1515	0,4273
	Aktual	-0,1836	-0,1514	0,4274

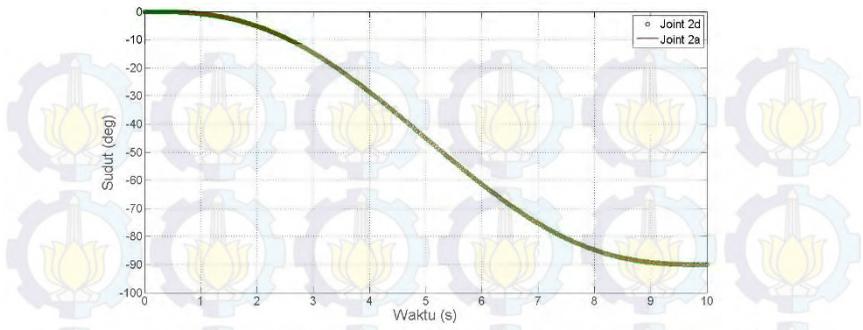
Dari Tabel 4.6 dapat dianalisis bahwa kontroler PD CTC mampu memperbaiki posisi sudut *joint* dengan nilai RMSE yang dihasilkan lebih kecil dibandingkan dengan posisi sudut *joint* tanpa kontroler sehingga kontroler PD CTC juga dapat memperbaiki posisi *end-effector* seperti yang ditunjukkan pada Tabel 4.7.

4.4 Analisis Sistem dengan Kontroler PD Fuzzy CTC

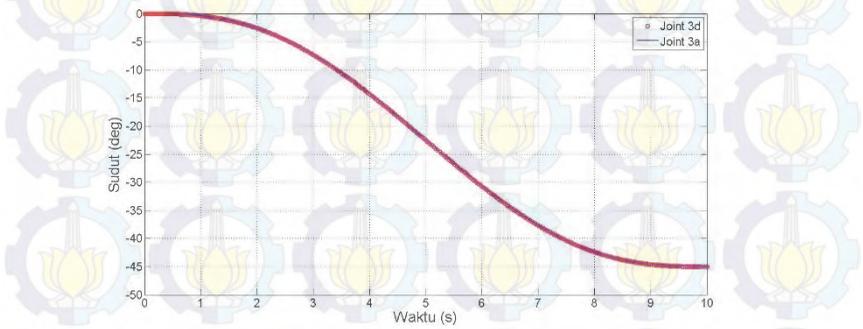
Selanjutnya digunakan kontroler PD Fuzzy CTC untuk memperbaiki posisi sudut *joint* dan *end-effector* pada kontroler PD CTC. Dari perancangan yang telah dilakukan didapatkan *gain* K_U sebesar 650, di mana respon posisi tiap-tiap *joint* pada posisi pertama dan kedua, ditunjukkan pada Gambar 4.33-4.41.



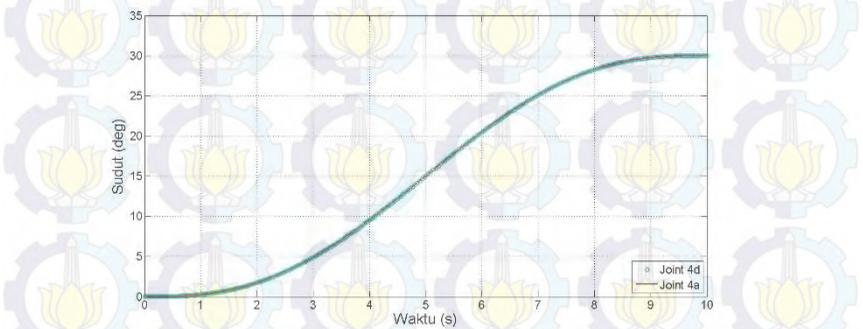
Gambar 4.33 Respon *Joint* ke-1 Posisi Pertama (PD Fuzzy CTC)



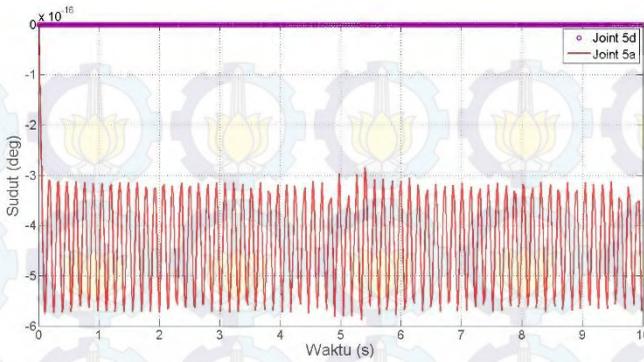
Gambar 4.34 Respon *Joint* ke-2 Posisi Pertama (PD Fuzzy CTC)



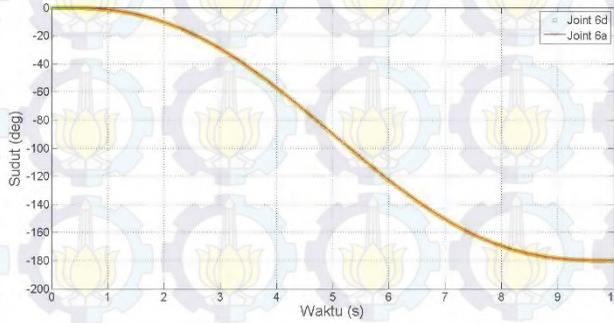
Gambar 4.35 Respon *Joint* ke-3 Posisi Pertama (PD Fuzzy CTC)



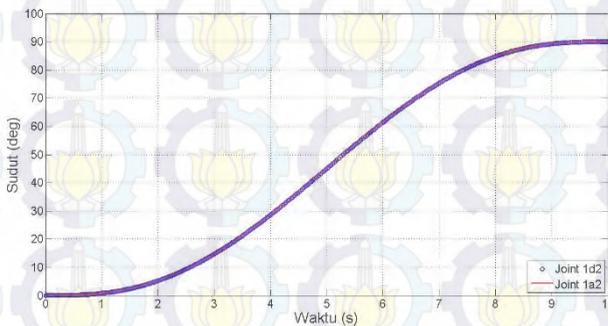
Gambar 4.36 Respon *Joint* ke-4 Posisi Pertama (PD Fuzzy CTC)



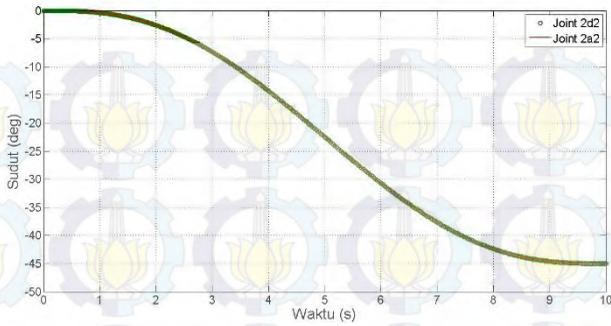
Gambar 4.37 Respon *Joint* ke-5 Posisi Pertama (PD Fuzzy CTC)



Gambar 4.38 Respon *Joint* ke-6 Posisi Pertama (PD Fuzzy CTC)



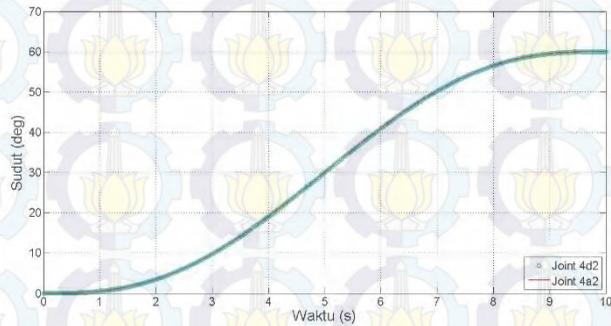
Gambar 4.39 Respon *Joint* ke-1 Posisi Kedua (PD Fuzzy CTC)



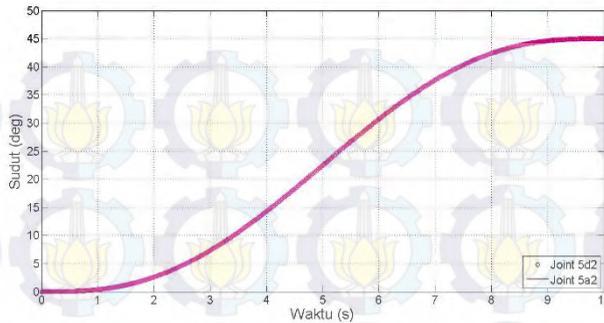
Gambar 4.40 Respon *Joint* ke-2 Posisi Kedua (PD Fuzzy CTC)



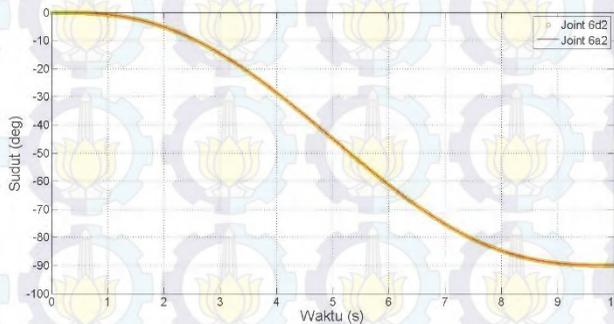
Gambar 4.41 Respon *Joint* ke-3 Posisi Kedua (PD Fuzzy CTC)



Gambar 4.42 Respon *Joint* ke-4 Posisi Kedua (PD Fuzzy CTC)



Gambar 4.43 Respon *Joint* ke-5 Posisi Kedua (PD *Fuzzy* CTC)



Gambar 4.44 Respon *Joint* ke-6 Posisi Kedua (PD *Fuzzy* CTC)

Dari respon posisi *joint* Gambar 4.33-4.44 didapatkan nilai RMSE antara nilai *set point* dan aktual, ditunjukkan pada Tabel 4.8.

Tabel 4.8 Nilai RMSE Kontroler PD *Fuzzy* CTC

Nilai RMSE	Posisi 1 (q_1)	Posisi 2 (q_2)
<i>Joint 1</i>	0,001167805	0,002304274
<i>Joint 2</i>	0,00228027	0,001124882
<i>Joint 3</i>	0,00111588	0,000751249
<i>Joint 4</i>	0,000778844	0,001538788
<i>Joint 5</i>	4,5289E-16	0,001157585
<i>Joint 6</i>	0,004420558	0,002259417

Dengan posisi *end-effector* ditunjukkan pada Tabel 4.9 berikut.

Tabel 4.9 Koordinat Posisi *End-effector* Robot Manipulator (PD *Fuzzy* CTC)

Koordinat		x_i	y_i	z_i
q_1	Set Point	-0,3399	-0,1291	0,07144
	Aktual	-0,3399	-0,1291	0,07145
q_2	Set Point	-0,1835	-0,1515	0,4273
	Aktual	-0,1835	-0,1515	0,4273

Dari Tabel 4.8 dapat dianalisis bahwa kontroler PD *Fuzzy* CTC mampu memperbaiki posisi sudut *joint* dengan nilai RMSE yang dihasilkan lebih kecil dibandingkan dengan posisi sudut *joint* dengan kontroler PD CTC.

Dari data yang ditunjukkan pada Tabel 4.9, kontroler PD *Fuzzy* CTC dapat memperbaiki posisi *end-effector* lebih baik dibandingkan dengan kontroler PD CTC.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil percobaan dan analisis data, ada beberapa hal yang dapat disimpulkan dari penelitian pada Tugas Akhir ini, yaitu:

1. Kontroler PD CTC mampu memperbaiki respon posisi tanpa kontroler, di mana posisi *joint* robot manipulator dapat mengikuti trayektori yang diinginkan.
2. Kontroler PD *Fuzzy* CTC mampu memperbaiki kekurangan respon pada kontroler PD CTC, di mana posisi *joint* robot manipulator dapat mengikuti trayektori yang diinginkan dengan nilai RMSE sekitar 10 kali lebih kecil pada tiap-tiap *joint*.
3. Kontroler PD *Fuzzy* CTC dapat memperbaiki posisi *end-effector* lebih baik dibandingkan dengan kontroler PD CTC dengan ketelitian mencapai 0,1 mm.

5.2 Saran

Berdasarkan pengamatan dan pembelajaran dari penelitian yang dilakukan pada Tugas Akhir ini, ada beberapa hal yang dapat digunakan untuk pengembangan robotika selanjutnya, yaitu:

1. Metode kontroler yang digunakan bisa dengan menerapkan tipe *Fuzzy* yang lain atau dengan menerapkan tipe kontroler yang lain yang berhubungan dengan optimalisasi pergerakan robot manipulator sehingga kontroler yang digunakan bisa lebih bervariasi dan bisa digunakan sebagai perbandingan serta analisis untuk pengembangan selanjutnya.
2. Untuk pengaturan robot selanjutnya tidak hanya dibatasi pada pengaturan posisi, tetapi bisa juga untuk pengaturan *path planning* yang lebih kompleks. Dengan skema yang jelas, robot manipulator dapat dirancang ke dalam pekerjaan-pekerjaan yang lebih spesifik dan aplikatif, seperti yang ada di dunia industri.



--- Halaman ini sengaja dikosongkan ---

DAFTAR PUSTAKA

- [1] Spong, M.W., Hutchinson, S., dan Vidyasagar M., "*Robot Dynamics and Control - Second Edition*", John Wiley & Sons, 2004.
- [2] Piltan, F., Yarmahmoudi, M.H., Shamsodini, M., Mazlomian, E., dan Hosainpour, A., "PUMA-560 Robot Manipulator Position Computed Torque Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinier Control and MATLAB COURSE", *International Journal of Robotics and Automation*, (IJRA), Vol. 3, No. 3, 2012.
- [3] Siciliano, B., Sciavicco, L., Villani, L., dan Oriolo, G., "*Robotics Modelling, Planning and Control*", Springer, London, 2009.
- [4] Fu, K.S., "Geometric Approach in Solving Inverse Kinematics of PUMA Robots", *IEEE Trans. Aero and Elect Sys*, Vol. AES-20, No. 6, pp. 695-706.
- [5] Fu, K.S., Gonzalez, R.C., dan Lee, C.S.G., "*Robotics: Control, Sensing, Vision, and Intelligence*", McGraw-Hill, New York, 1987.
- [6] Cubero, S., "*Industrial Robotics: Theory, Modelling and Control*", Advanced Robotics Systems International, Croatia, 2007.
- [7] Gamayanti, N., "*Diktat Kuliah Dasar Sistem Pengaturan*", Jurusan Teknik Elektro FTI-ITS, Surabaya, 2012.
- [8] P.P, Nyoman, "Desain Kontrol Fuzzy untuk Distribusi Torsi pada Hybrid Electric Vehicle", *Tugas Akhir*, Jurusan Teknik Elektro FTI-ITS, Surabaya, 2013.
- [9] Passino, K.M., dan Yurkovich, S., "*Fuzzy Control*", Ohio State University, Addison Wesley Longman, 1998.
- [10] Jang, R., Sun, T., dan Mizutani, E., "*Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*", Upper Saddle River, 1997.
- [11] Armstrong, B., Khatib, O., dan Burdick, J., "The Explicit Dynamic Model and Inertial Parameters of the PUMA 560 Arm", *IEEE Conference*, 2002, pp. 510-518.
- [12] Ziegler, J.G., dan Nichols, N.B., "Optimum Setting for Automatic Controller", *Transaction of the ASME*, No. 64, pp. 759-768, 1942.
- [13] Craig, J.J., "*Introduction to Robotics Mechanics and Control-Second Edition*", Addison-Wesley Publishing Company, New York, 1955.



RIWAYAT PENULIS



Duli Ridlo Istriantono, dilahirkan di Purworejo, Jawa Tengah pada tanggal 25 Januari 1991 sebagai putra bungsu dari Bapak Sumaryoto dan Ibu Sri Winarti. Setelah lulus dari pendidikan formal di SD Negeri Bragolan, SMP Negeri 2 Purworejo dan SMA Negeri 1 Purworejo, penulis melanjutkan pendidikan tinggi di program studi Diploma III Teknik Elektro UGM dan lulus pada tahun 2012. Penulis melanjutkan pendidikan sarjana di Teknik Elektro ITS pada tahun 2012 dan mengambil konsentrasi bidang studi Teknik Sistem Pengaturan. Selama menjadi mahasiswa, penulis aktif dalam kegiatan di Laboratorium Teknik Sistem Pengaturan sebagai asisten praktikum Sistem Pengaturan Analog serta Sistem Pengaturan Digital dan Otomasi Sistem. Pada bulan Januari 2015, penulis mengikuti seminar dan ujian Tugas Akhir sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Teknik Elektro. Email penulis adalah duliridlo@gmail.com.

LAMPIRAN A

A.1 Penurunan Model Dinamik *Two-Link Planar*

Dengan menggunakan bentuk geometris pada Gambar 3.3, dilakukan penurunan model dinamik untuk robot *two-link planar*. Pada *link-1* didapatkan koordinasi posisi sebagai berikut

$$\begin{aligned}x_1 &= l_1 \sin \theta_1 \\y_1 &= l_1 \cos \theta_1\end{aligned}\quad (\text{A.1})$$

Sehingga didapatkan energi kinetik dan energi potensial dengan formulasi sebagai berikut

$$\begin{aligned}K_1 &= \frac{1}{2} m_1 l_1^2 \dot{\theta}_1^2 \\P_1 &= m_1 g l_1 \cos \theta_1\end{aligned}\quad (\text{A.2})$$

Koordinat posisi dari *link-2* sebagai berikut

$$\begin{aligned}x_2 &= l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) \\y_2 &= l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2)\end{aligned}\quad (\text{A.3})$$

Kecepatan untuk *link-2* yang didapatkan dari penurunan koordinat posisi sebagai berikut

$$\begin{aligned}\dot{x}_2 &= l_1 \dot{\theta}_1 \cos \theta_1 + l_2 (\dot{\theta}_1 + \dot{\theta}_2) \cos(\theta_1 + \theta_2) \\ \dot{y}_2 &= -l_1 \dot{\theta}_1 \sin \theta_1 - l_2 (\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_1 + \theta_2) \\ v_2^2 &= \dot{x}_2^2 + \dot{y}_2^2 \\ &= l_1^2 \dot{\theta}_1^2 + l_2^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + 2l_1 l_2 \dot{\theta}_1 (\dot{\theta}_1 + \dot{\theta}_2) \cos \theta_2\end{aligned}\quad (\text{A.4})$$

Sehingga didapatkan persamaan energi kinetik dan potensial pada *link-2* sebagai berikut

$$K_2 = \frac{1}{2} m_1 l_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 l_2^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2 l_1 l_2 \dot{\theta}_1 (\dot{\theta}_1 + \dot{\theta}_2) \cos \theta_2$$

$$P_2 = m_2 g l_1 \cos \theta_1 + m_2 g l_2 \cos(\theta_1 + \theta_2) \quad (\text{A.5})$$

Dengan menggunakan persamaan *lagrange* yang merupakan selisih antara total energi kinetik dan energi potensial, maka didapatkan persamaan dinamik yang merupakan penurunan dari persamaan *lagrange* tersebut. Berikut ini merupakan penurunan persamaan *lagrange* untuk *link-1*.

$$\frac{\partial K}{\partial \dot{\theta}_1} = (m_1 + m_2) l_1^2 \dot{\theta}_1 + m_2 l_2^2 \dot{\theta}_1 + m_2 l_2^2 \dot{\theta}_2$$

$$+ 2m_2 l_1 l_2 \dot{\theta}_1 \cos \theta_2 + m_2 l_1 l_2 \dot{\theta}_2 \cos \theta_2$$

$$\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{\theta}_1} \right) = \left[(m_1 + m_2) l_1^2 + m_2 l_2^2 \dot{\theta}_1 + 2m_2 l_1 l_2 \cos \theta_2 \right] \ddot{\theta}_1$$

$$+ (m_2 l_2^2 + m_2 l_1 l_2 \cos \theta_2) \ddot{\theta}_2 - 2m_2 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2$$

$$- m_2 l_1 l_2 \dot{\theta}_2^2 \sin \theta_2$$

$$\frac{\partial P}{\partial \theta_1} = -m_1 g l_1 \sin \theta_1 - m_2 g l_1 \sin \theta_1 - m_2 g l_2 \sin(\theta_1 + \theta_2) \quad (\text{A.6})$$

Sehingga didapatkan formulasi torsi untuk *link-1* yang diperoleh dari persamaan *lagrange* sebagai berikut

$$\tau_1 = \frac{d}{dt} \frac{\partial K}{\partial \dot{\theta}_1} - \frac{\partial K}{\partial \theta_1} + \frac{\partial P}{\partial \theta_1}$$

$$\tau_1 = \left[(m_1 + m_2) l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 \cos \theta_2 \right] \ddot{\theta}_1$$

$$+ (m_2 l_2^2 + m_2 l_1 l_2 \cos \theta_2) \ddot{\theta}_2 - 2m_2 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 - m_2 l_1 l_2 \dot{\theta}_2^2 \sin \theta_2 \quad (\text{A.7})$$

$$- m_1 g l_1 \sin \theta_1 - m_2 g l_1 \sin \theta_1 - m_2 g l_2 \sin(\theta_1 + \theta_2)$$

Sedangkan penurunan persamaan *lagrange* untuk *link-2* sebagai berikut

$$\frac{\partial K}{\partial \dot{\theta}_2} = m_2 l_1^2 \dot{\theta}_1 + m_2 l_2^2 (\dot{\theta}_1 + \dot{\theta}_2) + m_2 l_1 l_2 (2\dot{\theta}_1 + \dot{\theta}_2) \cos \theta_2$$

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial K}{\partial \dot{\theta}_2} \right) &= m_2 (l_1^2 + l_2^2 + 2l_1 l_2 \cos \theta_2) \ddot{\theta}_1 \\ &\quad + m_2 l_2^2 \ddot{\theta}_2 - m_2 l_1 l_2 (2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) \sin \theta_2 \end{aligned}$$

$$\frac{\partial K}{\partial \theta_2} = -m_2 l_1 l_2 \dot{\theta}_1 (\dot{\theta}_1 + \dot{\theta}_2) \sin \theta_2$$

$$\frac{\partial P}{\partial \theta_2} = -m_2 g l_2 \sin(\theta_1 + \theta_2) \quad (\text{A.8})$$

Sehingga didapatkan formulasi torsi untuk *link-2* sebagai berikut

$$\begin{aligned} \tau_2 &= m_2 (l_2^2 + l_1 l_2 \cos \theta_2) \ddot{\theta}_1 + m_2 l_2^2 \ddot{\theta}_2 \\ &\quad + m_2 l_1 l_2 \dot{\theta}_1^2 \sin \theta_2 - m_2 g l_2 \sin(\theta_1 + \theta_2) \end{aligned} \quad (\text{A.9})$$

Di mana torsi pertama dan torsi kedua dapat dijabarkan menjadi beberapa representasi vektor/matriks seperti berikut

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \quad (\text{A.10})$$

Dari Persamaan 3.5 diketahui bahwa torsi robot manipulator memiliki vektor h yang merupakan inersia, vektor c yang merupakan gaya sentrifugal dan *coriolis*, serta vektor g yang merupakan gaya gravitasi.

Berikut ini penjabaran lebih detil matriks-matriks dari Persamaan (A.10)

$$h_{11} = (m_1 + m_2) l_1^2 + m_2 l_2^2 \dot{\theta}_1 + 2m_2 l_1 l_2 \cos \theta_2$$

$$h_{12} = m_2 l_2^2 + m_2 l_1 l_2 \cos \theta_2$$

$$h_{21} = m_2 (l_2^2 + l_1 l_2 \cos \theta_2)$$

$$h_{22} = m_2 l_2^2$$

(A.11)

$$\begin{aligned}
c_{11} &= -2m_2 l_1 l_2 \sin \theta_2 \dot{\theta}_2 \\
c_{12} &= -m_2 l_1 l_2 \sin \theta_2 \dot{\theta}_2 \\
c_{21} &= m_2 l_1 l_2 \sin \theta_2 \dot{\theta}_1 \\
c_{22} &= 0
\end{aligned} \tag{A.12}$$

$$\begin{aligned}
g_1 &= -m_1 g l_1 \sin \theta_1 - m_2 g l_1 \sin \theta_1 - m_2 g l_2 \sin(\theta_1 + \theta_2) \\
g_2 &= -m_2 g l_2 \sin(\theta_1 + \theta_2)
\end{aligned} \tag{A.13}$$

A.2 Penurunan Model Dinamik *Articulated Elbow*

Selanjutnya penurunan model dinamik untuk robot *articulated elbow*. Penurunan dilakukan dengan menggunakan geometris dari robot seperti yang ditunjukkan pada Gambar 3.4.

Koordinat posisi untuk *link-2* dengan tiga sumbu koordinat sebagai berikut

$$\begin{aligned}
x_2 &= l_2 C_1 S_2 \\
y_2 &= l_2 S_1 S_2 \\
z_2 &= l_2 C_2
\end{aligned} \tag{A.14}$$

Di mana agar lebih sederhana, $S_i = \sin \theta_i$ dan $C_i = \cos \theta_i$.

Kecepatan dari *link-2* yang diperoleh dari penurunan koordinat posisi sebagai berikut

$$\begin{aligned}
\dot{x}_2 &= l_2 (-S_1 \dot{\theta}_1 S_2 + C_1 C_2 \dot{\theta}_2) \\
\dot{y}_2 &= l_2 (C_1 \dot{\theta}_1 C_2 + S_1 C_2 \dot{\theta}_2) \\
\dot{z}_2 &= l_2 (-S_2 \dot{\theta}_2) \\
v_2^2 &= \dot{x}_2^2 + \dot{y}_2^2 + \dot{z}_2^2 \\
&= l_2^2 \left[(-S_1 \dot{\theta}_1 S_2 + C_1 C_2 \dot{\theta}_2)^2 + (C_1 \dot{\theta}_1 C_2 + S_1 C_2 \dot{\theta}_2)^2 + (S_2 \dot{\theta}_2)^2 \right] \\
&= l_2^2 (S_2^2 \dot{\theta}_1^2 + \dot{\theta}_2^2)
\end{aligned} \tag{A.15}$$

Energi kinetik dan energi potensial dari *link-2* sebagai berikut

$$\begin{aligned}
 K_2 &= \frac{1}{2} m_2 l_2^2 (\dot{S}_2^2 \dot{\theta}_1^2 + \dot{\theta}_2^2) \\
 P_2 &= m_2 g l_2 C_2
 \end{aligned} \tag{A.16}$$

Koordinat posisi untuk *link-3* sebagai berikut

$$\begin{aligned}
 x_3 &= C_1 (l_2 S_2 + l_3 S_{23}) \\
 y_3 &= S_1 (l_2 S_2 + l_3 S_{23}) \\
 z_3 &= l_2 C_2 + l_3 C_{23}
 \end{aligned} \tag{A.17}$$

Kecepatan dari *link-3* yang diperoleh dari penurunan koordinat posisi sebagai berikut

$$\begin{aligned}
 \dot{x}_3 &= -S_1 \dot{\theta}_1 (l_2 S_2 + l_3 S_{23}) + C_1 [l_2 C_2 \dot{\theta}_2 + l_3 C_{23} (\dot{\theta}_2 + \dot{\theta}_3)] \\
 \dot{y}_3 &= C_1 \dot{\theta}_1 (l_2 S_2 + l_3 S_{23}) + S_1 [l_2 C_2 \dot{\theta}_2 + l_3 C_{23} (\dot{\theta}_2 + \dot{\theta}_3)] \\
 \dot{z}_3 &= -l_2 S_2 \dot{\theta}_2 - l_3 S_{23} (\dot{\theta}_2 + \dot{\theta}_3) \\
 v_3^2 &= \dot{x}_3^2 + \dot{y}_3^2 + \dot{z}_3^2 \\
 &= \dot{\theta}_1^2 (l_2 S_2 + l_3 S_{23})^2 + [l_2 C_2 \dot{\theta}_2 + l_3 C_{23} (\dot{\theta}_2 + \dot{\theta}_3)]^2 \\
 &\quad + [-l_2 S_2 \dot{\theta}_2 - l_3 S_{23} (\dot{\theta}_2 + \dot{\theta}_3)]^2 \\
 &= (l_2 S_2 + l_3 S_{23})^2 \dot{\theta}_1^2 + l_2^2 \dot{\theta}_2^2 + 2l_2 l_3 C_3 \dot{\theta}_2 (\dot{\theta}_2 + \dot{\theta}_3) + l_3^2 (\dot{\theta}_2 + \dot{\theta}_3)^2
 \end{aligned} \tag{A.18}$$

Energi kinetik dan energi potensial dari *link-3* sebagai berikut

$$\begin{aligned}
 K_3 &= \frac{1}{2} m_3 (l_2 S_2 + l_3 S_{23})^2 \dot{\theta}_1^2 + \frac{1}{2} m_3 (l_2^2 + 2l_2 l_3 C_3 + l_3^2) \dot{\theta}_2^2 \\
 &\quad + m_3 l_3 (l_2 C_2 + l_3) \dot{\theta}_2 \dot{\theta}_3 + \frac{1}{2} m_3 l_3^2 \dot{\theta}_3^2 \\
 P_3 &= m_3 g (l_2 C_2 + l_3 C_{23})
 \end{aligned} \tag{A.19}$$

Persamaan *lagrange* sebagai berikut

$$\begin{aligned}
 L = & \frac{1}{2} m_2 l_2^2 (S_2^2 \dot{\theta}_1^2 + \dot{\theta}_2^2) + \frac{1}{2} m_3 (l_2 S_2 + l_3 S_{23})^2 \dot{\theta}_1^2 \\
 & + \frac{1}{2} m_3 (l_2^2 + 2l_2 l_3 C_3 + l_3^2) \dot{\theta}_2^2 + m_3 l_3 (l_2 C_3 + l_3) \dot{\theta}_2 \dot{\theta}_3 \\
 & + \frac{1}{2} m_3 l_3^2 \dot{\theta}_3^2 - m_2 g l_2 C_2 - m_3 g (l_2 C_2 + l_3 C_{23})
 \end{aligned} \tag{A.20}$$

Penurunan dari persamaan *lagrange* untuk *link-1*, *link-2* dan *link-3* sebagai berikut

$$\frac{\partial L}{\partial \dot{\theta}_1} = m_2 l_2^2 S_2^2 \dot{\theta}_1 + m_3 (l_2 S_2 + l_3 S_{23})^2 \dot{\theta}_1$$

$$\begin{aligned}
 \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_1} \right) = & m_2 l_2^2 S_2^2 \ddot{\theta}_1 + m_3 (l_2 S_2 + l_3 S_{23})^2 \ddot{\theta}_1 \\
 & + 2m_2 l_2^2 S_2 C_2 \dot{\theta}_2 + 2m_3 (l_2 S_2 + l_3 S_{23}) \\
 & [l_2 C_2 \dot{\theta}_2 + l_3 C_{23} (\dot{\theta}_2 + \dot{\theta}_3)] \dot{\theta}_1
 \end{aligned}$$

$$\frac{\partial L}{\partial \theta_1} = 0$$

$$\frac{\partial L}{\partial \dot{\theta}_2} = m_2 l_2^2 \dot{\theta}_2 + m_3 (l_2^2 + 2l_2 l_3 C_3 + l_3^2) \dot{\theta}_2 + m_3 l_3 (l_2 C_3 + l_3) \dot{\theta}_3$$

$$\begin{aligned}
 \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_2} \right) = & m_2 l_2^2 \ddot{\theta}_2 + m_3 (l_2 + 2l_2 l_3 C_3 + C_3) \ddot{\theta}_2 + m_3 l_3 (l_2 C_3 + l_3) \ddot{\theta}_3 \\
 & - 2m_3 l_2 l_3 S_3 \dot{\theta}_3 \dot{\theta}_2 - m_3 l_2 l_3 S_3 \dot{\theta}_3^2
 \end{aligned}$$

$$\begin{aligned}
 \frac{\partial L}{\partial \theta_2} = & m_2 l_2^2 S_2 C_2 \dot{\theta}_1^2 + m_3 (l_2 S_2 + l_3 S_{23}) (l_2 C_2 + l_3 C_{23}) \dot{\theta}_1^2 \\
 & + m_2 g l_2 S_2 + m_3 g (l_2 S_2 + l_3 S_{23})
 \end{aligned}$$

$$\frac{\partial L}{\partial \dot{\theta}_3} = m_3 l_3 (l_2 C_3 + l_3) \dot{\theta}_2 + m_3 l_3^2 \dot{\theta}_3$$

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_3} \right) &= m_3 l_3 (l_2 C_3 + l_3) \ddot{\theta}_2 + m_3 l_3^2 \ddot{\theta}_2 + m_3 l_3 (-l_2 S_3) \dot{\theta}_3 \dot{\theta}_2 \\ \frac{\partial L}{\partial \theta_3} &= m_3 (l_2 S_2 + l_3 S_{23}) l_3 C_{23} \dot{\theta}_1^2 - m_3 l_2 l_3 S_3 \dot{\theta}_2^2 \\ &\quad - m_3 l_1 l_2 S_3 \dot{\theta}_2 \dot{\theta}_3 + m_3 g (l_3 S_{23}) \end{aligned} \quad (\text{A.21})$$

Sehingga didapatkan formulasi torsi untuk *link-1*, *link-2* dan *link-3* sebagai berikut

$$\begin{aligned} \tau_1 &= \left[m_2 l_2^2 S_2^2 + m_3 (l_2 S_2 + l_3 S_{23})^2 \right] \ddot{\theta}_1 \\ &\quad + 2 \left[m_2 l_2^2 S_2 C_2 + m_3 (l_2 S_2 + l_3 S_{23}) (l_2 C_2 + l_3 C_{23}) \right] \dot{\theta}_1 \dot{\theta}_2 \\ &\quad + 2 m_3 l_3 C_{23} (l_2 S_2 + l_3 S_{23}) \dot{\theta}_3 \dot{\theta}_1 \\ \tau_2 &= \left[m_2 l_2^2 + m_3 (l_2^2 + 2 l_2 l_3 C_3 + l_3^2) \right] \ddot{\theta}_2 \\ &\quad + m_3 l_3 (l_2 C_3 + l_3) \ddot{\theta}_3 - 2 m_3 l_2 l_3 S_3 \dot{\theta}_3 \dot{\theta}_2 - m_3 l_1 l_2 S_3 \dot{\theta}_3^2 \\ &\quad - \left[m_2 l_2^2 S_2 C_2 + m_3 (l_2 S_2 + l_3 S_{23}) (l_2 C_2 + l_3 C_{23}) \right] \dot{\theta}_1^2 \\ &\quad - m_2 g l_2 S_2 - m_3 g (l_2 S_2 + l_3 S_{23}) \\ \tau_3 &= m_3 l_3 (l_2 C_3 + l_3) \ddot{\theta}_2 + m_3 l_3^2 \ddot{\theta}_3 - m_3 l_3 \dot{\theta}_2 \dot{\theta}_3 \\ &\quad - m_3 (l_2 S_2 + l_3 S_{23}) l_3 C_{23} \dot{\theta}_1^2 \\ &\quad + m_3 l_2 l_3 S_3 \dot{\theta}_2^2 + m_3 l_3 \dot{\theta}_2 \dot{\theta}_3 - m_3 g l_3 S_{23} \end{aligned} \quad (\text{A.22})$$

Berdasarkan torsi yang diperoleh dari Persamaan (A.22) dapat dijabarkan menjadi beberapa representasi vektor/matriks seperti yang dilakukan pada robot *two-link planar*.

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \\ \dot{\theta}_3^2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} \quad (\text{A.23})$$

Berikut ini merupakan penjabaran detil matriks inersia, matriks sentrifugal dan *coriolis*, serta matriks gravitasi dari Persamaan (A.23)

$$\begin{aligned}
h_{11} &= m_2 l_2^2 S_2^2 + m_3 (l_2 S_2 + l_3 S_{23})^2 \\
h_{12} &= 0 \\
h_{13} &= 0 \\
h_{21} &= 0 \\
h_{22} &= m_2 l_2^2 + m_3 (l_2^2 + 2l_2 l_3 C_3 + l_3^2) \\
h_{23} &= m_3 l_3 (l_2 C_3 + l_3) \\
h_{31} &= 0 \\
h_{32} &= m_3 l_3 (l_2 C_3 + l_3) \\
h_{33} &= m_3 l_3^2
\end{aligned} \tag{A.24}$$

$$\begin{aligned}
c_{11} &= 2 \left[m_2 l_2^2 S_2 C_2 + m_3 (l_2 S_2 + l_3 S_{23}) (l_2 C_2 + l_3 C_{23}) \right] \dot{\theta}_2 \\
&\quad + 2m_3 l_3 C_{23} (l_2 S_2 + l_3 S_{23}) \dot{\theta}_3
\end{aligned}$$

$$c_{12} = 0$$

$$c_{13} = 0$$

$$c_{21} = - \left[m_2 l_2^2 S_2 C_2 + m_3 (l_2 S_2 + l_3 S_{23}) (l_2 C_2 + l_3 C_{23}) \right] \dot{\theta}_1$$

$$c_{22} = -2m_3 l_2 l_3 S_3 \dot{\theta}_3$$

$$c_{23} = -m_3 l_2 l_3 S_3 \dot{\theta}_3$$

$$c_{31} = -m_3 (l_2 S_2 + l_3 S_{23}) l_3 C_{23} \dot{\theta}_1$$

$$c_{32} = m_3 l_2 l_3 S_3 \dot{\theta}_2$$

$$c_{33} = 0$$

$$g_1 = 0$$

$$g_2 = -m_2 g l_2 S_2 - m_3 g (l_2 S_2 + l_3 S_{23})$$

$$g_3 = -m_3 g l_3 S_{23}$$

$$\tag{A.26}$$

A.3 Model Dinamik PUMA-560

Model dinamik PUMA-560 didapatkan dari pembelajaran yang dilakukan oleh Armstrong [11] yang menggunakan program LISP bernama EMDEG menghasilkan model dinamik dari sebuah mekanisme

articulated. Model dinamik tersebut didapatkan dengan menggunakan persamaan *lagrange*. Pengaturan posisi PUMA-560 bergantung pada tiga *link* pertama dari total enam *link*, berikut ini persamaan dinamik dari PUMA-560

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = M(\theta) \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + B(\theta) \begin{bmatrix} \dot{\theta}_1 \dot{\theta}_2 \\ \dot{\theta}_1 \dot{\theta}_3 \\ \dot{\theta}_2 \dot{\theta}_3 \end{bmatrix} + C(\theta) \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \\ \dot{\theta}_3^2 \end{bmatrix} + G(\theta) \quad (\text{A.27})$$

Di mana,

$$M(\theta) = \begin{bmatrix} M_{11} & M_{12} & M_{13} & 0 & 0 & 0 \\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0 \\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0 \\ 0 & 0 & 0 & M_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & M_{66} \end{bmatrix} \quad (\text{A.28})$$

$$\begin{aligned} M_{11} = & I_{m1} + I_1 + I_3 c_2^2 + I_7 s_{23}^2 + I_{10} s_{23} c_{23} \\ & + I_{11} s_2 c_2 + I_{21} s_{23}^2 + 2(I_5 c_2 s_{23} + I_{12} c_2 c_{23}) \\ & + I_{15} s_{23} s_{23} + I_{16} c_2 s_{23} + I_{22} s_{23} c_{23} \end{aligned}$$

$$M_{12} = I_4 s_2 + I_8 c_{23} + I_9 c_2 + I_{13} s_{23} - I_{18} c_{23}$$

$$M_{13} = I_8 c_{23} + I_{13} s_{23} - I_{18} c_{23}$$

$$M_{22} = I_{m2} + I_2 + I_6 + 2(I_5 s_3 + I_{12} c_2 + I_{15} + I_{16} s_3)$$

$$M_{23} = I_5 s_3 + I_6 + I_{12} c_3 + I_{16} s_3 + 2I_{15}$$

$$M_{33} = I_{m3} + I_6 + 2^* I_{15}$$

$$M_{35} = I_{15} + I_{17}$$

$$M_{44} = I_{m4} + I_{14}$$

$$M_{55} = I_{m5} + I_{17}$$

$$M_{66} = I_{m6} + I_{23}$$

$$M_{21} = M_{12}, M_{31} = M_{13}, M_{32} = M_{23}$$

$$M_{14} = 0, M_{15} = 0, M_{16} = 0$$

$$M_{24} = 0, M_{25} = 0, M_{26} = 0$$

$$M_{34} = 0, M_{36} = 0$$

$$M_{41} = 0, M_{42} = 0, M_{43} = 0, M_{45} = 0, M_{46} = 0$$

$$M_{51} = 0, M_{52} = 0, M_{53} = 0, M_{54} = 0, M_{56} = 0$$

$$M_{61} = 0, M_{62} = 0, M_{63} = 0, M_{64} = 0, M_{65} = 0$$

(A.29)

$$B(\theta) = \begin{bmatrix} b_{112} & b_{113} & 0 & b_{115} & 0 & b_{123} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b_{214} & 0 & 0 & b_{223} & 0 & b_{225} & 0 & 0 & b_{235} & 0 & 0 & 0 \\ 0 & 0 & b_{314} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ b_{412} & b_{412} & 0 & b_{415} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b_{514} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(A.30)

$$b_{112} = 2(-I_3 s_2 c_2 + I_5 c_{223} + I_7 s_{23} c_{23} - I_{12} s_{223} + I_{15} 2s_{23} c_{23} + I_{16} c_{223} + I_{21} s_{23} c_{23} + I_{22} (1 - 2s_{23}^2)) + I_{10} (1 - 2s_{23}^2) + I_{11} (1 - 2s_2^2)$$

$$b_{113} = 2(I_5 c_{23} c_2 + I_7 s_{23} c_{23} - I_{12} s_{23} c_2 + I_{15} 2s_{23} c_{23} + I_{16} c_2 c_{23} + I_{21} s_{23} c_{23} + I_{22} (1 - 2s_{23}^2)) + I_{10} (1 - 2s_{23}^2)$$

$$b_{115} = 2(-s_{23} c_{23} + I_{15} s_{23} c_{23} + I_{16} c_2 c_{23} + I_{22} c_{23}^2)$$

$$b_{123} = 2(-I_8 s_{23} + I_{13} c_{23} + I_{18} s_{23})$$

$$b_{214} = I_{14} s_{23} + I_{19} s_{23} + I_{20} s_{23}$$

$$b_{223} = 2(-I_{12} s_3 + I_5 c_3 + I_{16} c_3)$$

$$b_{225} = 2(I_{16} c_3 + I_{22})$$

$$b_{235} = b_{225}$$

$$b_{314} = (I_{20} + I_{14} + I_{19}) s_{23}$$

$$b_{412} = -b_{214}$$

$$b_{413} = -b_{314}$$

$$b_{415} = -I_{20} s_{23} - I_{17} s_{23}$$

$$b_{514} = -b_{415}$$

(A.31)

$$C(\theta) = \begin{bmatrix} 0 & c_{12} & c_{13} & 0 & 0 & 0 \\ c_{21} & 0 & c_{23} & 0 & 0 & 0 \\ c_{31} & c_{32} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ c_{51} & c_{52} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{A.32})$$

$$\begin{aligned} c_{12} &= I_4 c_2 - I_8 s_{23} - I_9 s_2 + I_{13} c_{23} + I_{18} s_{23} \\ c_{13} &= 0.5b_{123} \\ c_{21} &= -0.5b_{112} \\ c_{23} &= 0.5b_{223} \\ c_{31} &= -0.5b_{113} \\ c_{32} &= -c_{23} \\ c_{51} &= -0.5b_{115} \\ c_{52} &= -0.5b_{225} \end{aligned} \quad (\text{A.33})$$

$$G(\theta) = \begin{bmatrix} 0 \\ G_2 \\ G_3 \\ 0 \\ G_5 \\ 0 \end{bmatrix} \quad (\text{A.34})$$

$$\begin{aligned} G_2 &= g_1 c_2 + g_2 s_{23} + g_3 s_2 + g_4 c_{23} + g_5 s_{23} \\ G_3 &= g_2 s_{23} + g_4 c_{23} + g_5 s_{23} \\ G_5 &= g_5 s_{23} \end{aligned} \quad (\text{A.35})$$

$$\ddot{\theta} = M^{-1}(\theta) [\tau - (B(\theta)\dot{\theta}\dot{\theta} + C(\theta)\dot{\theta}^2 + G(\theta))] \quad (\text{A.36})$$



LAMPIRAN B

B.1 Program Trayektori Kubik

```
% syms q0 v0 ac0 q1 v1 ac1 t0 tf
function qd=fcn(d)
global q0 v0 ac0 q1 v1 ac1 t0 tf

% d=[0,0,0,90,0.5,0.5,0,10]';
q0=d(1); v0= d(2); ac0 = d(3); q1 = d(4);v1 =
d(5);ac1=d(6);t0=d(7); tf=d(8); t=d(9); %% t=d(9), untuk
fungsi Step dalam Clock

% t = linspace(t0,tf,100*(tf-t0)); untuk fungsi Plot
c = ones(size(t));

% Matriks Trayektori Kubik
M = [1 t0 t0^2 t0^3 t0^4 t0^5;
      0 1 2*t0 3*t0^2 4*t0^3 5*t0^4;
      0 0 2 6*t0 12*t0^2 20*t0^3;
      1 tf tf^2 tf^3 tf^4 tf^5;
      0 1 2*tf 3*tf^2 4*tf^3 5*tf^4;
      0 0 2 6*tf 12*tf^2 20*tf^3];

b=[q0; v0; ac0; q1; v1; ac1];
a = M\b;

% qd = Trayektori Posisi
% vd = Trayektori Kecepatan
% ad = Trayektori Percepatan
qd = a(1).*c + a(2).*t + a(3).*t.^2 + a(4).*t.^3
+a(5).*t.^4 + a(6).*t.^5;
% vd = a(2).*c + 2*a(3).*t + 3*a(4).*t.^2 + 4*a(5).*t.^3
+5*a(6).*t.^4;
% ad = 2*a(3).*c + 6*a(4).*t + 12*a(5).*t.^2
+20*a(6).*t.^3;
%plot(t,qd)
```

B.2 Inverse Dynamic

B.2.1 Program Matriks A

```
function A = fcn(teta)

% Inisialisasi
c2=cos(teta(2));
s2=sin(teta(2));
```

```

c3=cos(teta(3));
s3=sin(teta(3));

c23=cos(teta(2)+teta(3));
s23=sin(teta(2)+teta(3));

% c223=cos(2*teta(2)+teta(3));
% s223=sin(2*teta(2)+teta(3));

% Konstanta Inersia (kg.m^2)
I1=1.43; %+- .05
I2=1.75;
I3=1.38;
I4=.69;
I5=.372;
I6=.333;
I7=.298;
I8=-.134;
I9=.0238;
I10=-.0213;
I11=-.0142;
I12=-.011;
I13=-.00379;
I14=.00164;
I15=.00125;
I16=.00124;
I17=.000642;
I18=.000431;
% I19=.0003;
% I20=-.000202;
I21=-.0001;
I22=-.000058;
I23=.00004;

Im1=1.14;
Im2=4.71;
Im3=.827;
Im4=.2;
Im5=.179;
Im6=.193;

% Konstanta Gravitasi (N.m)
% g1=-37.2;
% g2=-8.44;
% g3=1.02;
% g4=.249;
% g5=-.0282;

```

```
% Matriks A
```

```
a11=Im1+I1+I3*c2^2+I7*s23^2+I10*s23*c23+I11*s2*c2+I21*s23^2+2*(I5*c2*s23+I12*c2*c23+I15*s23*s23+I16*c2*s23+I22*s23*c23);
```

```
a12=I4*s2+I8*c23+I9*c2+I13*s23-I18*c23;
```

```
a13=I8*c23+I13*s23-I18*c23;
```

```
a22=Im2+I2+I6+2*(I5*s3+I12*c2+I15+I16*s3);
```

```
a23=I5*s3+I6+I12*c3+I16*s3+2*I15;
```

```
a33=Im3+I6+2*I15;
```

```
a35=I15+I17;
```

```
a44=Im4+I14;
```

```
a55=Im5+I17;
```

```
a66=Im6+I23;
```

```
a21=a12; a31=a13; a32=a23;
```

```
a14=0;
```

```
a15=0;
```

```
a16=0;
```

```
a24=0;
```

```
a25=0;
```

```
a26=0;
```

```
a34=0;
```

```
a36=0;
```

```
a41=0;
```

```
a42=0;
```

```
a43=0;
```

```
a45=0;
```

```
a46=0;
```

```
a51=0;
```

```
a52=0;
```

```
a53=0;
```

```
a54=0;
```

```
a56=0;
```

```
a61=0;
```

```
a62=0;
```

```
a63=0;
```

```
a64=0;
```

```
a65=0;
```

```

% Matriks A
A=[a11 a12 a13 a14 a15 a16;
   a21 a22 a23 a24 a25 a26;
   a31 a32 a33 a34 a35 a36;
   a41 a42 a43 a44 a45 a46;
   a51 a52 a53 a54 a55 a56;
   a61 a62 a63 a64 a65 a66];

```

B.2.2 Program Matriks B, C dan G

```

function TB = fcn(teta, tetad)

% Inisialisasi
c2=cos(teta(2));
s2=sin(teta(2));

c3=cos(teta(3));
s3=sin(teta(3));

c23=cos(teta(2)+teta(3));
s23=sin(teta(2)+teta(3));

c223=cos(2*teta(2)+teta(3));
s223=sin(2*teta(2)+teta(3));

% Konstanta Inersia (kg.m^2)
% I1=1.43; %+- .05
% I2=1.75;
I3=1.38;
I4=.69;
I5=.372;
% I6=.333;
I7=.298;
I8=-.134;
I9=.0238;
I10=-.0213;
I11=-.0142;
I12=-.011;
I13=-.00379;
I14=.00164;
I15=.00125;
I16=.00124;
I17=.000642;
I18=.000431;
I19=.0003;
I20=-.000202;
I21=-.0001;

```

```

I22=-.000058;
% I23=.00004;

% Im1=1.14;
% Im2=4.71;
% Im3=.827;
% Im4=.2;
% Im5=.179;
% Im6=.193;

% Konstanta Gravitasi (N.m)
g1=-37.2;
g2=-8.44;
g3=1.02;
g4=.249;
g5=-.0282;

% Matriks B
b112=2*(-I3*s2*c2+I5*c223+I7*s23*c23-
I12*s223+I15*2*s23*c23+I16*c223+I21*s23*c23+I22*(1-
2*s23^2))+I10*(1-2*s23^2)+I11*(1-2*s2^2);
b113=2*(I5*c23*c2+I7*s23*c23-
I12*s23*c2+I15*2*s23*c23+I16*c2*c23+I21*s23*c23+I22*(1-
2*s23^2))+I10*(1-2*s23^2);
b115=2*(-s23*c23+I15*s23*c23+I16*c2*c23+I22*c23^2);
b123=2*(-I8*s23+I13*c23+I18*s23);
b214=I14*s23+I19*s23+I20*s23;
b223=2*(-I12*s3+I5*c3+I16*c3);
b225=2*(I16*c3+I22);
b235=b225;
b314=(I20*+I14+I19)*s23;
b412=-b214;
b413=-b314;
b415=-I20*s23-I17*s23;
b514=-b415;

% Matriks B
B= [b112 b113 0 b115 0 b123 0 0 0 0 0 0 0 0 0 0;
    0 0 b214 0 0 b223 0 b225 0 0 b235 0 0 0 0 0;
    0 0 b314 0 0 0 0 0 0 0 0 0 0 0 0;
    b412 b413 0 b415 0 0 0 0 0 0 0 0 0 0 0;
    0 0 b514 0 0 0 0 0 0 0 0 0 0 0 0;
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];

% Matriks C
c12=I4*c2-I8*s23-I9*s2+I13*c23+I18*s23;
c13=.5*b123;
c21=-.5*b112;
c23=.5*b223;

```

```

c31=-.5*b113;
c32=-c23;
c51=-.5*b115;
c52=-.5*b225;

% Matriks C
C= [0 c12 c13 0 0 0;
    c21 0 c23 0 0 0;
    c31 c32 0 0 0 0;
    0 0 0 0 0 0;
    c51 c52 0 0 0 0;
    0 0 0 0 0 0];

% Matriks G
G2=g1*c2+g2*s23+g3*s2+g4*c23+g5*s23;
G3=g2*s23+g4*c23+g5*s23;
G5=g5*s23;

% Matriks G
G=[0;G2;G3;0;G5;0];

% Vektor Kecepatan Sudut, q4=q5=q6=0
qdt=[tetad(1)*tetad(2);tetad(1)*tetad(3);0;0;0;tetad(2)*te
tetad(3);0;0;0;0;0;0;0];
qsq=[tetad(1)^2;tetad(2)^2;tetad(3)^2;0;0;0];

% Torsi Robot Manipulator Tiap Lengan
TB=B*qdt+C*qsq+G;

```

B.3 Program Dinamik

```

function tetadda = fcn(T, teta, tetad)

% Inisialisasi
c2=cos(teta(2));
s2=sin(teta(2));

c3=cos(teta(3));
s3=sin(teta(3));

c23=cos(teta(2)+teta(3));
s23=sin(teta(2)+teta(3));

c223=cos(2*teta(2)+teta(3));
s223=sin(2*teta(2)+teta(3));

% Konstanta Inersia (kg.m^2)
I1=1.43; %+-.05

```

```

I2=1.75;
I3=1.38;
I4=.69;
I5=.372;
I6=.333;
I7=.298;
I8=-.134;
I9=.0238;
I10=-.0213;
I11=-.0142;
I12=-.011;
I13=-.00379;
I14=.00164;
I15=.00125;
I16=.00124;
I17=.000642;
I18=.000431;
I19=.0003;
I20=-.000202;
I21=-.0001;
I22=-.000058;
I23=.00004;

Im1=1.14;
Im2=4.71;
Im3=.827;
Im4=.2;
Im5=.179;
Im6=.193;

% Konstanta Gravitasi (N.m)
g1=-37.2;
g2=-8.44;
g3=1.02;
g4=.249;
g5=-.0282;

% Matriks A
a11=Im1+I1+I3*c2^2+I7*s23^2+I10*s23*c23+I11*s2*c2+I21*s23^
2+2*(I5*c2*s23+I12*c2*c23+I15*s23*s23+I16*c2*s23+I22*s23*c
23);
a12=I4*s2+I8*c23+I9*c2+I13*s23-I18*c23;
a13=I8*c23+I13*s23-I18*c23;

a22=Im2+I2+I6+2*(I5*s3+I12*c2+I15+I16*s3);
a23=I5*s3+I6+I12*c3+I16*s3+2*I15;

a33=Im3+I6+2*I15;
a35=I15+I17;

```

```

a44=Im4+I14;
a55=Im5+I17;
a66=Im6+I23;
a21=a12; a31=a13; a32=a23;

a14=0;
a15=0;
a16=0;

a24=0;
a25=0;
a26=0;

a34=0;
a36=0;

a41=0;
a42=0;
a43=0;
a45=0;
a46=0;

a51=0;
a52=0;
a53=0;
a54=0;
a56=0;

a61=0;
a62=0;
a63=0;
a64=0;
a65=0;

% Matriks A
A= [a11 a12 a13 a14 a15 a16;
    a21 a22 a23 a24 a25 a26;
    a31 a32 a33 a34 a35 a36;
    a41 a42 a43 a44 a45 a46;
    a51 a52 a53 a54 a55 a56;
    a61 a62 a63 a64 a65 a66];

% Matriks B

```

```

b112=2*(-I3*s2*c2+I5*c223+I7*s23*c23-
I12*s223+I15*2*s23*c23+I16*c223+I21*s23*c23+I22*(1-
2*s23^2))+I10*(1-2*s23^2)+I11*(1-2*s2^2);
b113=2*(I5*c23*c2+I7*s23*c23-
I12*s23*c2+I15*2*s23*c23+I16*c2*c23+I21*s23*c23+I22*(1-
2*s23^2))+I10*(1-2*s23^2);
b115=2*(-s23*c23+I15*s23*c23+I16*c2*c23+I22*c23^2);
b123=2*(-I8*s23+I13*c23+I18*s23);
b214=I14*s23+I19*s23+I20*s23;
b223=2*(-I12*s3+I5*c3+I16*c3);
b225=2*(I16*c3+I22);
b235=b225;
b314=(I20*+I14+I19)*s23;
b412=-b214;
b413=-b314;
b415=-I20*s23-I17*s23;
b514=-b415;

```

```

% Matriks B

```

```

B= [b112 b113 0 b115 0 b123 0 0 0 0 0 0 0 0 0 0;
    0 0 b214 0 0 b223 0 b225 0 0 b235 0 0 0 0 0;
    0 0 b314 0 0 0 0 0 0 0 0 0 0 0 0 0;
    b412 b413 0 b415 0 0 0 0 0 0 0 0 0 0 0 0;
    0 0 b514 0 0 0 0 0 0 0 0 0 0 0 0 0;
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];

```

```

% Matriks C

```

```

c12=I4*c2-I8*s23-I9*s2+I13*c23+I18*s23;
c13=.5*b123;
c21=-.5*b112;
c23=.5*b223;
c31=-.5*b113;
c32=-c23;
c51=-.5*b115;
c52=-.5*b225;

```

```

% Matriks C

```

```

C= [0 c12 c13 0 0 0;
    c21 0 c23 0 0 0;
    c31 c32 0 0 0 0;
    0 0 0 0 0 0;
    c51 c52 0 0 0 0;
    0 0 0 0 0 0];

```

```

% Matriks G

```

```

G2=g1*c2+g2*s23+g3*s2+g4*c23+g5*s23;
G3=g2*s23+g4*c23+g5*s23;
G5=g5*s23;

```

```

% Matriks G
G=[0;G2;G3;0;G5;0];

% Vektor Kecepatan Sudut, q4=q5=q6=0
qdt=[tetad(1)*tetad(2);tetad(1)*tetad(3);0;0;0;tetad(2)*te
tad(3);0;0;0;0;0;0;0;0;0];
qsq=[tetad(1)^2;tetad(2)^2;tetad(3)^2;0;0;0];

% Dynamic
tetadda=A\ (T- (B*qdt+C*qsq+G));

```

B.4 Program *Plotting* Gerakan Robot Manipulator

```

% Inialisai DH Parameter dari Tiap Link &
alpha      a      theta      d      sigma
L1 = link([-pi/2      0      0      0      0      ],
'standard');
L2 = link([ 0      0.4318      0      0.14909      0      ],
'standard');
L3 = link([ pi/2      -0.0203      0      0      0      ],
'standard');
L4 = link([-pi/2      0      0      0.43307      0      ],
'standard');
L5 = link([ pi/2      0      0      0      0      ],
'standard');
L6 = link([ 0      0      0      0.05625      0      ],
'standard');

% Penggabungan antar Link &
R=robot({L1 L2 L3 L4 L5 L6});

% Nama Robot Manipulator dalam Simulasi &
R.name='PUMA-560';

```