

ISSN 1035-5421

Volume II, Buku 2 Juni 2015

STIPRO

STINDO PROFESIONAL JURNAL

EKONOMI, HUKUM, PERTANIAN, TEKNIK,
KEDOKTERAN, KESEHATAN, AGAMA,
PENDIDIKAN, KOMPUTER, SOSIAL POLITIK,
PETERNAKAN, PARIWISATA, SENI, BAHASA





STIPRO

STINDO PROFESIONAL JURNAL

Redaksi : Jl. Mesjid Taufik No. 166 Medan, Telp. 0812 6363 020
Jl. Perbaungan No. 2 Medan. Telp. (061) 732 1118

Email : stindoprofesional@yahoo.com

Surat Keterangan

Nomor : 29/Stipro/VI/2015

Yang bertanda tangan dibawah ini :

Nama : Drs. Harkim Simamora, SE.,M.Si
Jabatan : Pimpinan Redaksi Stindo Profesional Jurnal

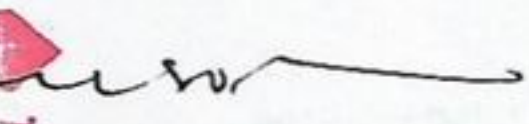
Menerangkan bahwa :

Nama : Timbang Pangaribuan & Parulian Siagian
Pekerjaan : Dosen
Judul Jurnal : Pengkajian Dasar Desain Sistem Kendali Terbarukan
Menggunakan Fuzzy Logic Pada Sistem Kendali Digital Dengan
Bantuan Matlab

Adalah benar yang bersangkutan memuat tulisan jurnalnya pada Stindo Profesional Jurnal dengan nomor : ISSN 2243-0536 yang diterbitkan pada volume II buku 2, Juni 2015.

Demikianlah surat keterangan ini dibuat untuk dapat dipergunakan sebagaimana perlunya, dan bila terdapat kesalahan maupun kekeliruan akan diperbaiki sebagaimana mestinya

Medan, Juni 2015

Pimpinan Redaksi

(Drs. Harkim Simamora, SE., M.Si)

Pengkajian Dasar Desain Sistem Kendali Terbarukan Menggunakan Fuzzy Logic Pada Sistem Kendali Digital Dengan Bantuan Program Matlab

Timbang Pangaribuan
Parulian Siagian

Abstrak

Sistem kendali cerdas yang didukung oleh matematika dan sekelompok teori seperti pemrograman menggunakan linguistik, dikenal dengan sistem kendali logika fuzzy. Dalam memahami logika fuzzy lebih benar dan tepat, seseorang dapat melakukan simulasi numerik untuk mengamati perubahan subset fuzzy dan perilakunya dalam dunia kendali digital. Rancangan sebuah Pengendali Digital berbasis Logika Fuzzy memerlukan sebuah tahapan atau proses pengkajian dan perhitungan yang panjang dan berulang-ulang, dan untuk dapat menghasilkan data fuzzy, data numerik dan data grafik dibutuhkan metoda pemrograman yang cepat. Pemrograman komputer MATLAB adalah salah satu solusi untuk menjawab persoalan membuat program desain kendali logika fuzzy dengan teknik fuzzifikasi dan defuzzifikasi. Perangkat lunak matlab menyediakan fasilitas pembuat grafik yang sangat memadai dan teraplikasi dalam sistem windows. Oleh karena itu solusi perancangan sistem kendali logika fuzzy dengan dengan simulasi dirasa sangat perlu untuk diungkapkan, terutama jika disimulasikan dengan pemrograman matlab.

PENDAHULUAN

Suatu sistem fisik disebut dengan plant, dimana plant adalah suatu sistem yang akan dikendalikan seperti motor listrik arus searah. Persamaan suatu plant dapat diperoleh dengan memberikan pemodelan pada plant dimaksud dengan pemodelan sistem fisik ke dalam bentuk matematik. Bentuk matematis ini dapat dianalisis lebih lanjut sesuai dengan keperluan, misalnya untuk simulasi dalam hal menampilkan respon waktu dalam kawasan waktu kontinyu. Persamaan diferensial dalam bentuk kawasan waktu dapat dibentuk menjadi persamaan diferensi dalam kawasan waktu diskret, dan persamaan diferensi dapat ditransformasi ke dalam bentuk Transformasi-Z. Solusi yang diberikan oleh transformasi-Z dikembalikan juga tetap kedalam kawasan waktu diskret, untuk keperluan simulasi respon.

Salah satu dari bentuk sistem kendali terbaru yang belum banyak digunakan adalah bentuk Fuzzy Logic Controller (FLC) atau disebut dengan Sistem Kendali Logika Fuzzy. Sistem Logika Fuzzy diperkenalkan oleh seorang ilmuwan bernama Zadeh, dan desain dengan fuzzy sudah menjadi terkenal dan aktif digunakan dalam dunia sistem kendali, dan berhasil cukup luas dalam aplikasi teori set fuzzy.

Fuzzy sangat dekat atau erat dengan pola pengambilan keputusan yang dilakukan manusia, dan memiliki bahasa alami dalam sistem logikanya. Sistem kendali fuzzy didasarkan pada logika fuzzy, memiliki paradigma baru sedemikian sehingga pengendali dapat dirancang untuk hal yang kompleks, dan dikenal dengan istilah ill-defined processes, karena tanpa perlu mengetahui analisis data kuantitatif yang berhubungan dengan

proses evaluasi fungsi transfer yang menyangkut hubungan input-output. Penggunaan program simulasi dalam sistem diskret dengan interval waktu k , akan memperoleh iterasi yang cukup banyak jika waktu yang diberikan cukup lama dengan interval waktu atau perioda waktu yang kecil. Oleh karena itu perhitungan numerik tidak dapat dilakukan lagi dengan manual, dan untuk menggambarkan solusi grafis juga tidak dapat lagi dilakukan secara manual. Oleh karena itu, persoalan matematis dan numerik dimaksud dapat diselesaikan menggunakan software khusus pemograman dalam bentuk matematik yang dikenal dengan software matrix laboratory atau dikenal dengan nama **matlab**.

Yang menjadi masalah dalam penelitian ini adalah bagaimana membangun algoritma dalam melakukan desain sistem kendali berbasis logika fuzzy, tetapi tetap mengikutsertakan pola desain dalam sistem kendali konvensional. Tahapan diawali dengan membangun set-fuzzy, dilanjutkan dengan membangun mekanisme inferensi fuzzy, diteruskan dengan membuat fuzzification interface, kemudian membangun decision making logic, dan algoritma fuzzy diakhiri dengan defuzzification interface.

Sebagai hasil final yang diharapkan, bahwa sistem fuzzy itu sendiri terdiri dari 3 hubungan pembagian dikenal dengan *three-fuzzy-partition-terms* atau tiga pembagian set-fuzzy. Set fuzzy pertama untuk error, set fuzzy kedua untuk perubahan error, dan set fuzzy ketiga untuk kendali plant.

Maka yang menjadi masalah utama adalah menyelidiki perilaku respon ditinjau dari tiga hal yaitu : overshoot, osilasi dan error steady state, dengan cara melakukan perubahan pada beberapa jenis parameter

pengendali fuzzy dimaksud yaitu parameter subset fuzzy itu sendiri.

Tujuan dari penelitian ini adalah untuk merancang tahapan-tahapan dalam menyelesaikan desain sistem kendali terbaru dengan Sistem Kendali Logika Fuzzy atau metoda Fuzzy Logic Controller dilengkapi dengan program komputer berbasis matlab, dengan memilih objeknya plant berorde dua dengan redaman 0,29 dan frekuensi alamiah 1,73 rad/s, sehingga akan diperoleh kondisi respon yang lebih baik dengan Sistem Kendali Logika Fuzzy.

TINJAUAN PUSTAKA

Plant

Secara umum plant dalam sistem kendali dibentuk dari susunan bentuk persamaan diferensial orde ke-satu hingga orde ke- n yang bersifat linier, dan dapat dituliskan seperti pada persamaan (2-1).

$$\frac{d^n y}{dt^n} + a_{n-1} \frac{d^{n-1} y}{dt^{n-1}} + \dots + a_1 \frac{dy}{dt} + a_0 y = u$$

dimana, a_{n-1}, \dots, a_1, a_0 , adalah konstanta, y adalah variabel output sebagai fungsi dari waktu, u sebagai variabel input sebagai fungsi dari waktu dan tadalah variabel waktu kontinyu.

Persamaan diferensial di atas dapat lebih mudah diselesaikan dengan menggunakan Transformasi Laplace. Selanjutnya persamaan yang diperoleh dengan Transformasi Laplace akan dikonversi ke bentuk diskret yaitu transformasi-Z, dengan syarat semua kondisi awal untuk setiap perioda waktu yang diberikan diketahui.

Metoda Fuzzy

Konsep fuzzy diawali dengan setumpuk pernyataan yang dikenal dengan istilah **if then rule**. Sebagai contoh, jika udara panas dalam satu ruangan, maka

dibukalah jendela. Jika udara masih tetap panas, maka dihidupkanlah kipas angin. Selanjutnya jika udara masih tetap panas, dimatikanlah kipas dan ditutuplah jendela kemudian dihidupkanlah mair condition. Demikianlah konsep fuzzi *if then rule*, yang akan dicoba dan akan diterapkan dalam desain sistem kendali digital.

Tiga aspek dasar yang diperlukan dalam menentukan model sistem fuzzi, yaitu :

1. Mencari asal-usul gambaran bahasa yang digunakan
2. Menentukan cara implementase dari model fuzzi.
3. Menggunakan secara efektif perpaduan dari data fuzzi dan data numerik.

Model dari sebuah fuzzi terdiri dari sejumlah pengertian dasar dalam bentuk :

Jika u_1 adalah A_1 ,
 u_2 adalah A_2 ,
.....
 u_k adalah A_k
maka

$$y = \square_0 + \square_1 u_1 + \square_2 u_2 + \dots + \square_k u_k \quad (2-1)$$

Untuk memahami sistem fuzzi seperti itu diperhatikan beberapa hal :

1. Susunan variabel dalam implikasi dasar adalah :

$$u_1, u_2, \dots, u_k$$

2. Fungsi keanggotaan set-fuzzi dalam implikasi dasar adalah :

$$A_1, A_2, \dots, A_k$$

3. Parameter dalam bentuk dasar adalah,

$$\square_0, \square_1, \square_2 \dots, \square_k$$

Penentuan ketiga hal di atas sangatlah sulit dalam proses identifikasi parameter.

Fokus desain dalam Fuzzy Logic Controller adalah meniru perilaku operator manusia. Tahap pertama yang dilakukan adalah memodelkan sistem dengan menggunakan persamaan diferensial. Selanjutnya pengendalian dilakukan dengan sistem cerdas berbasis aturan fuzzi, dengan proses model logika berfikir manusia, dimana seorang manusia dapat mahir setelah melalui pelatihan, sehingga manusia mampu memanipulasi sistem. Pergeseran akan terjadi karena proses seseorang yang dilibatkan untuk mengubah pendekatan biasa ke sistem kendali otomatis.

SKLF adalah metodologi kendali digital yang mensimulasikan proses berpikir manusia dengan menggabungkan pengaruh yang melekat (*impression inherent*) dalam semua sistem fisik suatu plant yang akan dikendalikan. SKLF juga merupakan suatu algoritma yang dapat mengubah strategi kendali bahasa (*linguistik*), yang didasarkan pada basis pengetahuan (*expert knowledge*), semuanya kedalam strategi kendali otomatis. Sedangkan susunan dari sebuah SKLF adalah terdiri dari satu set aturan kendali bahasa (*linguistic control rules*) sebagai konsep bersama dari implikasi fuzzi (*fuzzy implications*) dan gubahan aturan dari kesimpulan yang diperoleh (*compositional rule of inference*). Sehingga pada akhirnya SKLF dapat dipandang sebagai suatu pendekatan yang menggabungkan kendali otomatis konvensional presisi dan pembuatan keputusan pola manusia (*human like decision making*), demikian dipaparkan oleh langari & Tomizuka tahun 1990 serta Chin & Chand tahun 1991 dalam analisis kendali robust (*kokoh*) dalam sistem dengan orde rendah.

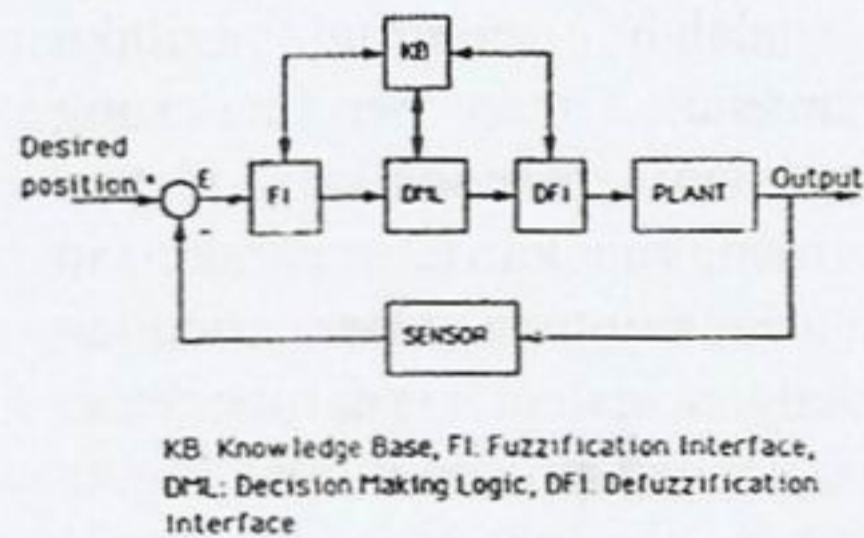


Diagram Blok Sistem Kendali Logika Fuzzi

Basis dari suatu atau FLC memiliki diagram blok seperti yang ditunjukkan pada Gambar 2.1, terdiri dari empat komponen dasar : (1) pertama, sebuah Fuzzification Interface (FI) sebagai perantara fuzzifikasi yang berfungsi seperti sebuah pengubah analog ke digital dalam komputer digital ; (2) kedua, sebuah Decision Making Logic (DML) sebagai logika pengambilan keputusan, berfungsi sebagai sebuah pengendali digital ; (3) ketiga, sebuah Defuzzification Interface (DI) sebagai perantara defuzzifikasi layaknya seperti sebuah pengubah digital ke analog juga dalam komputer digital ; (4) dan keempat, sebuah Knowledge Base (KB) sebagai basis pengetahuan, yang berfungsi seperti teorema kendali digital. Fungsi dari setiap komponen dijelaskan sebagai berikut :

FI : Komponen FI membentuk fungsi berikut ini :

- Mengukur nilai dari variabel-variabel input
- Pemetaan skala, yang mentransfer batas dari nilai input ke nilai tertentu yang bersesuaian yang dikaji secara ilmiah.
- Fuzzifikasi, mengubah data input ke nilai bahasa yang cocok yang ditandai dari set-fuzzi.

DML : Komponen DML adalah inti dari FLC. Komponen ini memiliki

kemampuan mensimulasikan pengambilan keputusan manusia yang didasarkan pada konsep fuzzi. DML dapat menaksir aksi kendali fuzzi dari aksi implikasi fuzzi dan mengatur keputusan dalam logika fuzzi.

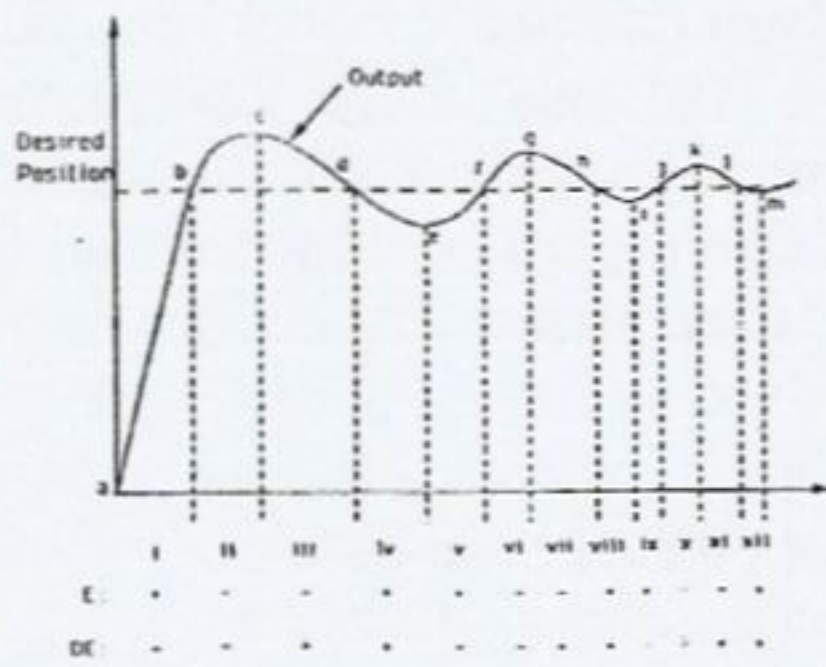
DFI : Komponen DFI membentuk fungsi berikut ini :

- Pemetaan skala, yang mengubah batas nilai variabel output ke nilai tertentu yang bersesuaian yang dikaji secara ilmiah.
- Defuzzifikasi, yang menghasilkan suatu aksi kendali non-fuzzi suatu keputusan aksi kendali fuzzi.

KB : Komponen KB terdiri dari basis pengetahuan tentang domain aplikasi dan goal pengendalian yang akan dicapai. KB terdiri dari basis data suatu basis aturan kendali bahasa fuzzi (linguistic fuzzy control rule base).

- Suatu basis data menyediakan ketentuan yang diperlukan untuk digunakan untuk mendefinisikan aturan-aturan kendali dan manipulasi data fuzzi dalam FLC.
- Karakteristik basis aturan tentang goal kendali dan pengawasan kendali dari domain cerdas dengan menggunakan aturan kendali linguistik.

Dasar berfikir dari logika fuzzi diambil dari respon seperti ditunjukkan oleh suatu respon sistem pada Gambar 2.2. Ada beberapa titik kunci penting yaitu a,b,c,d,e,f,g,h,i,j,k,l,m. Titik-titik tersebut adalah titik-titik berarti dalam pengambilan keputusan sistem fuzzi. Disaat respon disisi atas ($output > input$), respon disisi bawah ($output < input$), respon disisi yang pas ($output = input$).



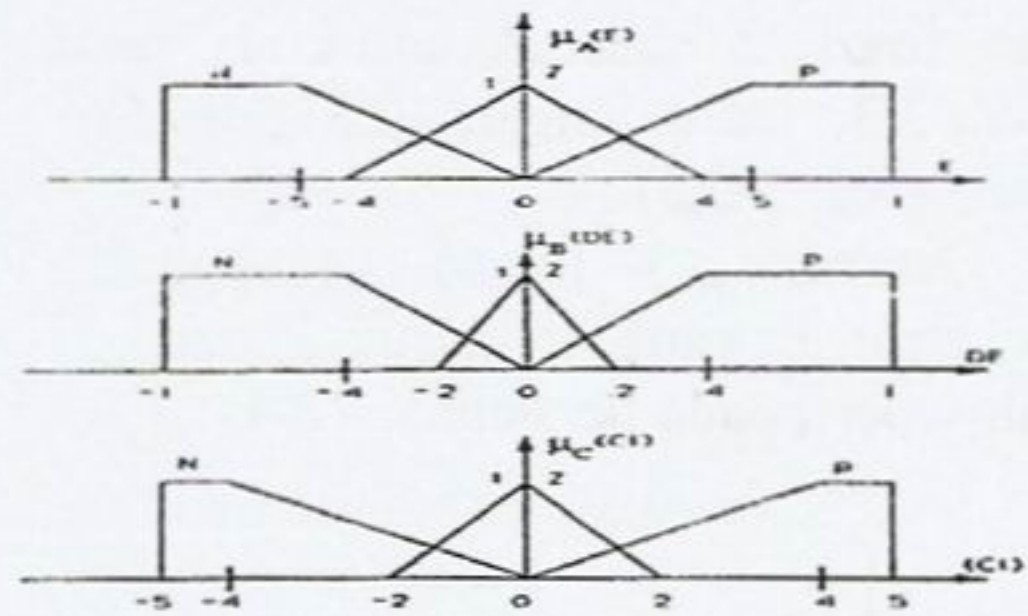
Respon Sistem Sebagai Acuan Berpikir Fuzzi

Grafik yang ditunjukkan pada Gambar 2.2 adalah respon output suatu plant dengan masukan fungsi unit step, memiliki redaman kurang dari satu. Selisih masukan dengan keluaran disebut dengan error E, sedangkan selisih error untuk setiap waktu disebut dengan DE.

Jika besarnya keluaran lebih besar dari besarnya masukan maka error E akan bertanda negatif, dan jika besarnya keluaran lebih kecil dari besarnya masukan maka error E bertanda positif, berarti jika besarnya keluaran sama dengan besarnya masukan error sama dengan zero (nol). Sedangkan jika error sekarang saat waktu t lebih besar dari error sebelumnya saat waktu t- Δ t, maka perubahan error DE akan bertanda positif, dan akan bertanda negatif jika error sekarang lebih kecil dari error sebelumnya, serta akan bertanda zero jika error sekarang sama dengan error sebelumnya.

Fuzzification Interface

Proses Perantara Fuzzifikasi FI adalah suatu prosedur penilaian subjektif yang mengubah hasil pengukuran ke suatu nilai subjektif, dan dapat didefinisikan sebagai pemetaan dari suatu ruang input yang diamati ke suatu set-fuzzi dalam input tertentu sebagai kajian ilmiah universal.



Representasi Fungsi Keanggotaan A, B dan J

Dalam aplikasi kendali, data yang diamati adalah disebut crisp (data yang kena cuplik dalam waktu singkat). Selanjutnya strategi fuzzifikasi menginterpretasikan suatu nilai crisp x sebagai suatu set-fuzzi A melalui fungsi keanggotaan $\mu_A(x)$.

Fungsi keanggotaan yang diujikan pada suatu sistem dengan tiga keanggotaan Error E, Derivative Error DE dan nilai Control Input CI adalah seperti ditunjukkan pada Gambar 2.3. Ketiga E, DE dan CI disatukan kedalam partisi tiga set-fuzzi (three fuzzy partitions), dimana batas E dan DE berada dalam range [-1,1] dan batas CI berada dalam range [-5,5]. Masing-masing set-fuzzi memiliki tiga daerah, daerah subset fuzzi Negatif (N), daerah subset fuzzi Zero (Z) dan daerah subset fuzzi Positif (P).

Error E ditandai dengan set fuzzi A, dan perubahan error DE ditandai dengan set fuzzi B. Baik A ataupun B akan memiliki fungsi anggota μ_A dan μ_B . Kemudian setiap fungsi anggota akan memiliki tiga daerah, daerah negatif N, daerah zero Z dan daerah positif P.

Jika E = 0,3 (error = 0,3) ini disebut sebagai error Z dan error P, karena jika ditarik ke garis ordinat akan menyentuh daerah Z dan P. Derajat kebenaran (degree of truth) untuk Z adalah $\mu_{AZ}(0,3) = 0,25$ dan untuk P $\mu_{AP}(0,3) = 0,6$. Untuk error negatif simbol ditandai dengan μ_{AN} . Dengan demikian untuk nilai error E = 0,3 akan ditransfer nilai crisp kepada dua

subset fuzzy Z dan P untuk E pada Gambar 2.3, yaitu dengan $(0,3, 0,25) \in Z$ dan $(0,3, 0,6) \in P$.

Dari Gambar 2.3 fungsi keanggotaan untuk error E yang termasuk kepada N adalah seperti pada persamaan (2-4).

$$\mu_{AN}(e) = \begin{cases} 1, & -1 \leq e \leq -0.5 \\ -2e, & -0.5 < e < 0 \\ 0, & 0 \leq e \leq 1 \end{cases}$$

Dengan cara yang sama bahwa fungsi keanggotaan untuk error E yang termasuk kepada Z dan P adalah seperti pada persamaan (2-5) dan persamaan (2-6).

$$\mu_{AZ}(e) = \begin{cases} 0, & -1 \leq e \leq -0.4 \\ 2.5(e + 0.4), & -0.4 < e \leq 0 \\ -2.5(e - 0.4), & 0 < e < 0.4 \\ 0, & 0.4 \leq e \leq 1 \end{cases}$$

$$\mu_{AP}(e) = \begin{cases} 0, & -1 \leq e \leq 0 \\ 2e, & 0 < e < 0.5 \\ 1, & 0.5 \leq e \leq 1 \end{cases}$$

Dengan cara yang sama, definisikanlah pada DE tiga variabel lain, yaitu \square_{BN} , \square_{BZ} , \square_{BP} , dan fungsi keanggotaan untuk error DE yang termasuk kepada N, Z dan P adalah seperti pada persamaan (2-7), (2-8) dan persamaan (2-9).

$$\mu_{BN}(d) = \begin{cases} 1, & -1 \leq d \leq -0.4 \\ -2.5d, & -0.4 < d < 0 \\ 0, & 0 \leq d \leq 1 \end{cases}$$

$$\mu_{BZ}(d) = \begin{cases} 0, & -1 \leq d \leq -0.2 \\ 5(d + 0.2), & -0.2 < d \leq 0 \\ -5(d - 0.2), & 0 < d < 0.2 \\ 0, & 0.2 \leq d \leq 1 \end{cases}$$

$$\mu_{BP}(d) = \begin{cases} 0, & -1 \leq d \leq 0 \\ 2.5d, & 0 < d < 0.4 \\ 1, & 0.4 \leq d \leq 1 \end{cases}$$

Dengan cara yang sama juga, definisikanlah pada J pada CI sebagai

variabel baru dan tiga variabel lain, yaitu \square_{JN} , \square_{JZ} , \square_{JP} , dan fungsi keanggotaan untuk error CI yang termasuk kepada N, Z dan P adalah seperti pada persamaan (2-10), (2-11) dan persamaan (2-12).

$$\mu_{JN}(c) = \begin{cases} 1, & -5 \leq c \leq -4 \\ -0.25c, & -4 < c < 0 \\ 0, & 0 \leq c \leq 5 \end{cases}$$

$$\mu_{JZ}(c) = \begin{cases} 0, & -5 \leq c \leq -2 \\ 0.5(c + 2), & -2 < c \leq 0 \\ -0.5(c - 2), & 0 < c < 2 \\ 0, & 2 \leq c \leq 5 \end{cases}$$

$$\mu_{JP}(c) = \begin{cases} 0, & -5 < c \leq 0 \\ 0.25c, & 0 < c < 4 \\ 1, & 4 \leq c \leq 5 \end{cases}$$

Decision Making Logic

Decision making Logic (DML) akan menggunakan dua dari ketiga set fuzzy yang dibentuk yaitu A dan B. Penggambaran Knowledge Base yang akan mengambil keputusan dari fungsi keanggotaan error E dan perubahan error DE harus didasarkan pada basis pengetahuan yang berasal dari respon plant dari Gambar 2.2. Setiap titik referensi pada Gambar 2.2 akan memberi arti yang diuraikan pada basis pengetahuan pada Tabel 2-2 dan Tabel 2-3.

Titik Referensi Basis Pengetahuan 1

PROTOTYPE OF DML RULES WITH THREE FUZZY SUBSETS

Rule no.	E	DE	CI	Reference point
1	P	Z	P	a, e, i
2	Z	N	N	b, f, j
3	N	Z	N	c, g, k
4	Z	P	P	d, h, l
5	Z	Z	Z	Desired position

With term sets (negative, zero, positive).

Titik Referensi Basis Pengetahuan 2

RULE JUSTIFICATION WITH THREE FUZZY SUBSETS

Rule no.	E	DE	CI	Reference region
6	P	N	P	i (rise time), v
7	N	N	N	ii (overshoot), vi
8	N	P	N	iii, vii
9	P	P	P	iv, viii
10	P	N	Z	ix
11	N	P	Z	xi

With term sets (negative, zero, positive).

Jika Tabel 2-1 dan Tabel 2-2 disusun sedemikian rupa sesuai dengan urutannya pada sebuah tabel matrik 3 x 3 maka akan diperoleh sebuah tabel baru untuk DML rule antara E vs DE seperti pada Tabel 2-3.

Terdapat dua keputusan ganda antara E pada P dan DE pada N yaitu menghasilkan CI pada posisi P atau Z, boleh dipilih salah satu tergantung keperluan, dan demikian juga untuk E pada N dan DE pada P yaitu menghasilkan CI pada posisi N atau Z, juga boleh dipilih salah satu tergantung keperluan.

Titik Referensi Basis Pengetahuan Utama

	DE	N	Z	P
E		N	Z	P
	N	N	N	N/Z
	Z	N	Z	P
	P	P/Z	P	P

Misalkan bahwa error $E = 0,3$ maka dari fungsi keanggotaan pada persamaan (2-4) sampai (2-6), diperoleh dua crisp yaitu $(0,3, 0,25) \in AZ$ dan $(0,3, 0,6) \in AP$. Selanjutnya untuk error derivative $DE = -0,3$ maka dari fungsi keanggotaan pada persamaan (2-7) sampai (2-7), diperoleh hanya satu crisp yaitu $(-0,3, 0,75) \in BN$. Untuk mengambil keputusan dengan aturan DML pada E dan DE, diperlukan dua aturan dari Tabel 2-1 dan Tabel 2-2, yaitu :

Jika E memiliki AZ dan DE memiliki BN keputusannya pada rule nomor 2 adalah bahwa CI memiliki JN (negatip).

Sedangkan jika E memiliki AP dan DE memiliki BN keputusannya pada rule nomor 6 adalah bahwa CI memiliki JP (positip).

Aturan DML dimaksud secara universal dituliskan dengan :

$$\begin{aligned} \square_R &= \square_{(E \text{ and } DE \rightarrow CI)}(e, de, ci) = [\\ &\square_{AE}(e) \cap \square_{BDE}(de)] \rightarrow \square_{JC}(ci) \\ &= \square_{AE}(e) \cap \square_{BDE}(de) \equiv \\ &\square_{JC}(ci) \quad (2-13) \end{aligned}$$

(\cap adalah lambang and/irisan)

Selanjutnya untuk itu dibuat DML :

$$\text{Rule 2 : } \square_{JN}(ci) = \min [\square_{AZ}(0,3), \square_{BN}(-0,3)] = 0,25 \cap 0,75 = 0,25$$

$$\text{Rule 6 : } \square_{JP}(ci) = \min [\square_{AP}(0,3), \square_{BN}(-0,3)] = 0,60 \cap 0,75 = 0,60$$

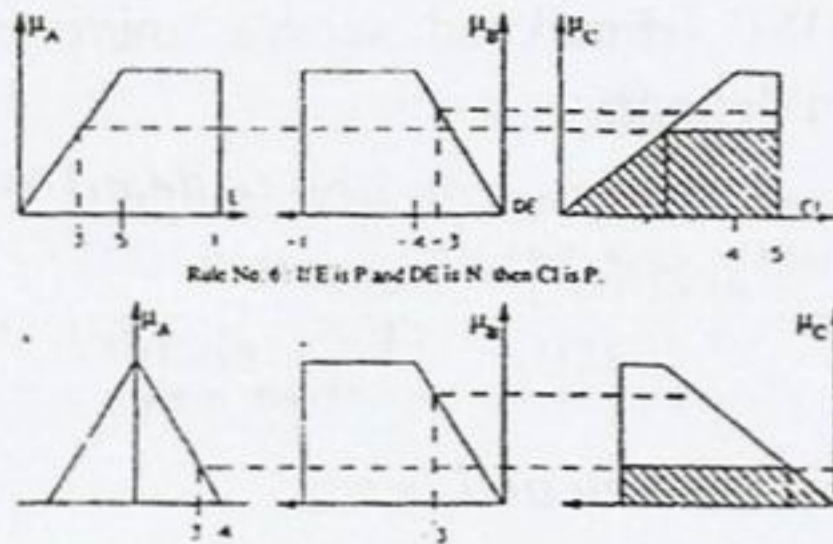
Kedua nilai $\square_{JN}(ci)$ dan $\square_{JP}(ci)$ yang diperoleh akan digunakan dalam proses defuzzifikasi berikut.

Defuzzification Interface

Strategi yang dilakukan pada proses perantara defuzzifikasi DFI adalah suatu pemetaan dari aksi kendali fuzzy yang didefinisikan di atas suatu output universal dari kajian ilmiah kedalam ruang aksi kendali non-fuzzy. Tujuannya adalah menghasilkan aksi kendali non-fuzzy yang memberi hasil terbaik untuk kemungkinan pendistribusian suatu keputusan aksi kendali fuzzy.

Kenyataan yang ada oleh Lee 1990 dapat dijelaskan bahwa ada kriteria yang dapat digunakan untuk melakukan penentuan aksi kendali non-fuzzy yang digunakan adalah kriteria Centre of Area (COA). Strategi COA adalah untuk menentukan titik berat atau titik keseimbangan antara hasil yang diakibatkan oleh J pada N, Z dan P yang akan menghasilkan aksi kendali pada CI.

Ditinjau dari kedua hasil yang diperoleh dari DML sebelumnya, maka interpretasi dari defuzzifikasi kedua \square_{JN} dan \square_{JP} yang diperoleh adalah dengan memilih titik keseimbangan seperti ditunjukkan pada Gambar 2.4.



Konsekuensi Pengambilan Keputusan Fuzzi

Nilai $\mu_{JP} = 0,6$ dari hasil keputusan E dan DE menghasilkan crisp pada daerah positif dari subset fuzzi CI, dan nilai tersebut diberlakukan pada persamaan (2-12) memberi hasil perhitungan,

$$\mu_{JP}(ci) = 0,6 = 0,25 ci$$

Sehingga diperoleh

$$ci = 2,4.$$

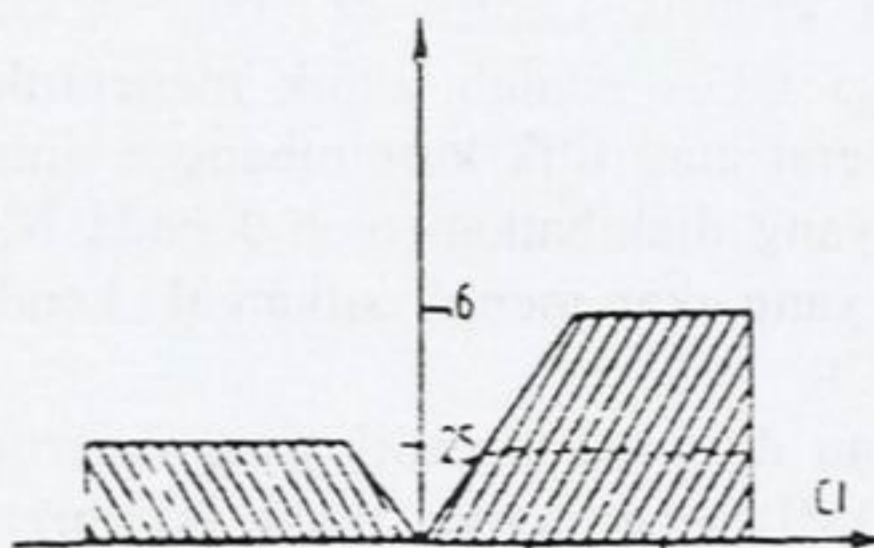
Nilai $\mu_{JN} = 0,25$ dari hasil keputusan E dan DE menghasilkan crisp pada daerah negatif dari subset fuzzi CI, dan nilai tersebut diberlakukan pada persamaan (2-10) memberi hasil perhitungan,

$$\mu_{JN}(ci) = 0,25 = -0,25 ci$$

Sehingga diperoleh

$$ci = -1,0.$$

Kedua hasil ci tersebut memiliki perpaduan crisp seperti pada Gambar 2.5, dan akan memberi hasil input kendali dengan angka tertentu setelah strategi defuzzifikasi, dengan menggunakan persamaan (2-12).



Input Kendali Setelah Strategi Defuzzifikasi

$$ci_{crisp} = \frac{\sum_{j=1}^n \mu_j(ci_j) \cdot ci_j}{\sum_{j=1}^n \mu_j(ci_j)} \quad (2-13)$$

Maka titik berat sebagai hasil strategi COA dengan menggunakan persamaan (2-13) diperoleh dengan:

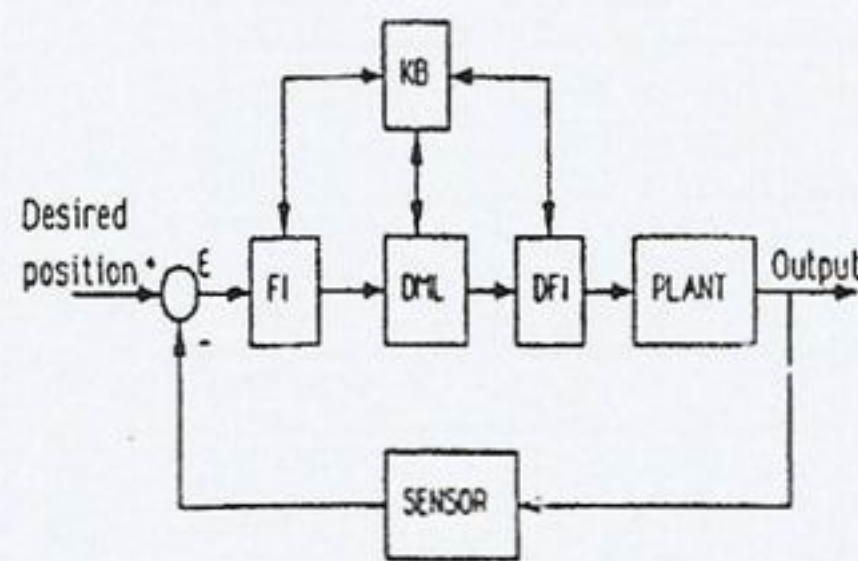
$$ci_{crisp} = \frac{\sum_{j=1}^2 \mu_j(ci_j) \cdot ci_j}{\sum_{j=1}^2 \mu_j(ci_j)} = \frac{(0,6)(2,4) + (0,25)(-1)}{(0,6) + (0,25)} = 1,4$$

Dalam hal ini control input CI yang dihasilkan dengan strategi proses fuzzifikasi adalah $ci_{crisp} = 1,4$.

DESAIN SISTEM KENDALI LOGIKA FUZZI

Teori Dasar Desain

Desain sistem kendali adalah suatu proses dalam pengolahan error sebagai selisih antara input dan output untuk diolah sedemikian dalam membangun sinyal akan akan diberikan kepada suatu plant. Sinyal error diolah dengan proses fuzzifikasi dan menghasilkan nilai yang disebut dengan set fuzzi, selanjutnya diproses dengan algoritma pengambilan keputusan dengan basis data fuzzi, dan hasil proses ini diolah ulang dengan proses defuzzifikasi untuk memberikan sinyal kendali ke suatu plant. Proses itu berlangsung terus menerus hingga satu waktu diperoleh respon steady state.



KB: Knowledge Base, FI: Fuzzification Interface,
DML: Decision Making Logic, DFI: Defuzzification
Interface

Struktur Sistem Pengendali Logika Fuzzi

Langkah-langkah yang perlu dibangun dalam desain sistem kendali untuk proses kendali digital adalah :

1. Langkah pertama adalah perlu membangun persamaan diferensi untuk proses simulasi plant dengan mengacu pada persamaan (2-4). Jika plant memiliki fungsi transfer dengan persamaan,

$$G(z) = \frac{Y(z)}{U(z)} = z^{-d} \frac{B(z)}{A(z)}$$

maka persamaan diferensi untuk suatu plant orde kedua dsiberikan oleh,

$$G(z) = \frac{Y(z)}{U(z)} = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

dan memberikan persamaan diferensi,

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_0 u(k-1) + b_1 u(k-2)$$

$u(k-1)$ dan $u(k-2)$ diperoleh dari nilai $u(k)$ setelah berjalan waktu diskret, demikian juga $y(k-1)$ dan $y(k-2)$ diperoleh dari nilai $y(k)$ juga setelah berjalan waktu diskret, sementara a_1 , a_2 , b_0 dan b_1 adalah parameter dari plant.

2. Langkah kedua adalah membangun persamaan fungsi keanggotaan yang mengacu kepada persamaan (2-4) sampai dengan persamaan (2-9). Fungsi dimaksud harus sudah diterapkan dalam bentuk sub-program yang setiap saat dapat digunakan dalam simulasi sistem kendali.
3. Langkah ketiga adalah membangun program untuk merealisasikan DML yang didasarkan pada KB dengan menggunakan Tabel 2-1 dan Tabel 2-2 atau dengan Tabel 2-3.
4. Langkah keempat adalah proses akhir dari pengolahan sistem kendali yaitu proses defuzzifikasi. Proses ini menggunakan persamaan (2-13) untuk menentukan nilai fungsi keanggotaan control input, dan hasilnya diolah dengan menggunakan persamaan (2-10) sampai dengan persamaan (2-12) untuk memperoleh nilai control input dimaksud.
- 5.

Membangun Program Simulasi Plant

Jika persamaan plant yang digunakan adalah

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_0 u(k-1) + b_1 u(k-2)$$

maka proses simulasi dalam matlab adalah seperti berikut ini :

$y(k)$, $y(k-1)$ dan $y(k-2)$ dalam bentuk matlab dituliskan dengan :

yk, yk1, yk2.

$u(k)$, $u(k-1)$ dan $u(k-2)$ dalam bentuk matlab dituliskan dengan :

uk, uk1, uk2.

Parameter a_1 , a_2 , b_0 dan b_1 dituliskan dengan dengan,

a1, a2, b0, b1.

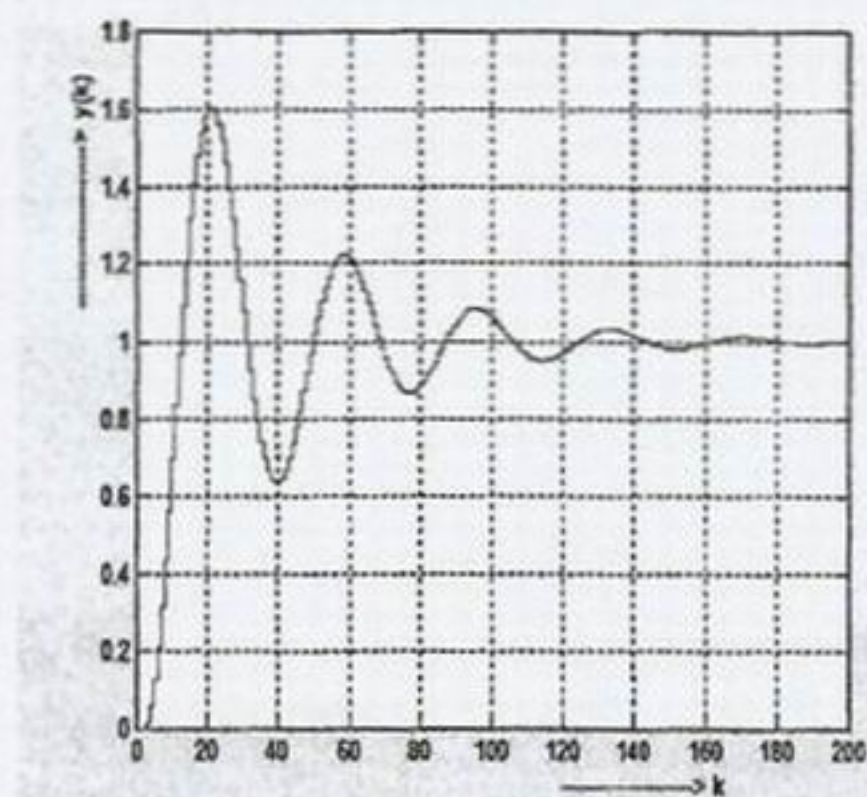
Selanjutnya simulasi plant memerlukan waktu diskrit, dapat dipilih tertentu

sampai dilihat respon keadaan tunak, misalnya dipilih k dari 0 sampai 200.

Untuk proses sistem loop tertutup dalam mensimulasikan plant, maka program yang digunakan adalah sebagai berikut :

```
clear
% Plant :
b = 3 ; a1 = 1 ; a0 = 0 ;
ns = b ; ds = [1 a1 a0] ;
T = 0.1 ;
[nz,dz] = c2dm(ns,ds,T,'zoh') ;
% Simulasi :
% harga awal :
yk = 0 ; yk1 = 0 ; yk2 = 0 ;
rk = 1 ; uk1 = 0 ; uk2 = 0 ;
ek1 = 0 ;
K = 1 ;
% loop :
for i = 1:200
ek = rk - yk ;
uk = K * ek ;
y(i) = yk ;
u(i) = uk ;
yk = -dz(2)*yk1 - dz(3)*yk2 +
nz(2)*uk1 + nz(3)*uk2 ;
yk2 = yk1 ; yk1 = yk ; uk2 =
uk1 ; uk1 = uk ;
ek1 = ek ;
end
% simulasi respon :
stairs(y),grid
xlabel('-----> k')
ylabel('-----> y(k)')
text(150,1.3,'FLC')
text(150,0.9,'Asli')
% Selesai.
```

Hasil uji coba program di atas memberikan respon awal plant seperti pada Gambar 3.2.



Representasi Respon Plant Awal

Spesifikasi dasar, plant memiliki overshoot 60%, respon berosilasi, kondisi keadaan tunak terjadi pada $k = 120$ dengan output $y(k) = 1$ pada input yang diberikan $r(k) = 1$, sehingga error $e(k) = 0$. Diinginkan setelah dilakukan pengendalian, bahwa overshoot dan osilasi harus dikurangi sebaik mungkin, dan error steady state harus dijaga tetap sekecil mungkin.

Membangun Program Fuzzifikasi

Dalam simulasi sistem kendali, setiap satu loop untuk setiap kenaikan waktu k diperlukan perhitungan. Oleh karena itu dalam proses fuzzifikasi diperlukan sub-program perhitungan supaya setiap saat dapat ditentukan subset fuzzy dari nilai e dan de yang dihasilkan dalam setiap loop. Nilai dimaksud dalam bentuk \square_A dan \square_B yang terdiri dari \square_{AN} , \square_{AZ} , \square_{AP} dan \square_{BN} , \square_{BZ} , \square_{BP} .

Error dan Derivative Error dibuat memiliki batas -1 sampai 1, dan mengacu pada persamaan (2-4) sampai persamaan (2-9) dibuat program seperti berikut ini.

```
function [A,B] = fuzzy (e , de)
```

```
% fuzzifikasi untuk e dan de untuk memperoleh miuA dan miuB :
```

```
% parameter untuk e :
```

```
a = 0.4 ; % batas untuk Z
```

```
b = 0.5 ; % batas untuk N-P
```

```
% parameter untuk de :
```

```
c = 0.2 ; % batas untuk Z
```

```

    d = 0.4 ; % batas untuk N-P
    if e >= -1 & e <= -b
        AN = 1;
    end
    if e > -b & e < 0
        AN = -(1/b)*e;
    end
    if e >= 0 & e <= 1
        AN = 0;
    end
    if e >= -1 & e <= -a
        AZ = 0;
    end
    if e > -a & e <= 0
        AZ = (1/a)*(e+a);
    end
    if e > 0 & e < a
        AZ = -(1/a)*(e-a);
    end
    if e >= a & e <= 1
        AZ = 0;
    end
    if e >= -1 & e <= 0
        AP = 0;
    end
    if e > 0 & e < b
        AP = (1/b)*e;
    end
    if e >= b & e <= 1
        AP = 1;
    end
    if de >= -1 & de <= -d
        BN = 1;
    end
    if de > -d & de < 0
        BN = -(1/d)*de;
    end
    if de >= 0 & de <= 1
        BN = 0;
    end
    if de >= -1 & de <= -c
        BZ = 0;
    end
    if de > -c & de <= 0
        BZ = (1/c)*(de+c);

```

```

    end
    if de > 0 & de < c
        BZ = -(1/c)*(de-c);
    end
    if de >= c & de <= 1
        BZ = 0;
    end
    if de >= -1 & de <= 0
        BP = 0;
    end
    if de > 0 & de < d
        BP = (1/d)*de;
    end
    if de >= d & de <= 1
        BP = 1;
    end
    % hail akhir
        A=[AN,AZ,AP] ;
        B=[BN,BZ,BP] ;
    % return.
    Hasil uji coba program di atas dapat
    dilakukan dengan memberikan program
    utama untuk error sebagai berikut :
    % Menggambarkan Membership
    Processors
    % untuk error e :
    b = 0.5; % batasan N dan P
    a = 0.4; % batasan Z
        m = 0.1 ;
    % pembagi antara -1 dan 1
    for i = 1:(1+1)/m+1
        % Memberships of e :
        e = (i - 1) * 0.1 ;
        E(i) = e ;
        % 1. Menggambar AN :
        if e >= -1 & e <= -b
            AN(i) = 1 ;
        end
        if e > -b & e < 0
            AN(i) = -1/b*E(i) ;
        end
        if e >= 0 & e <= 1
            AN(i) = 0 ;
        end
        % 2. Menggambar AZ :
        if e >= -1 & e <= -a

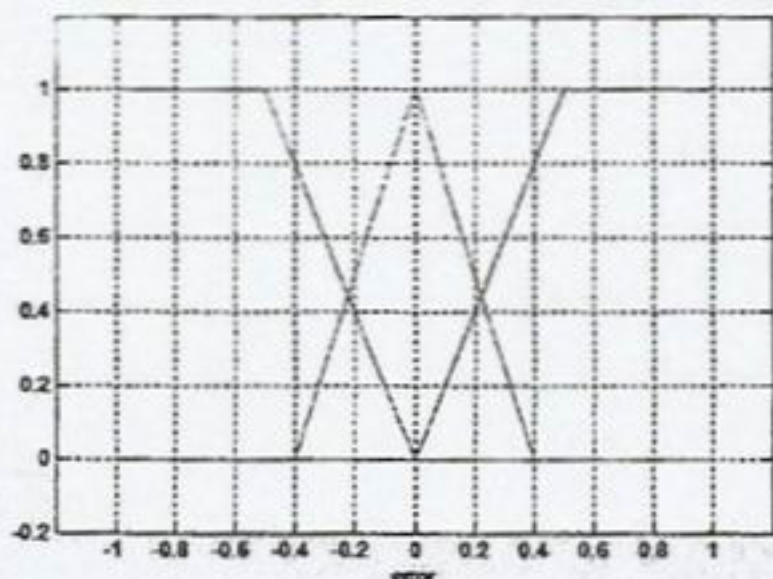
```

```

AZ(i) = 0 ;
end
if e > -a & e <= 0
AZ(i) = 1/a*(E(i)+a) ;
end
if e > 0 & e < 4/10
AZ(i) = -1/a*(E(i)-a) ;
end
if e >= a & e <= 1
AZ(i) = 0 ;
end
% 3. Menggambar AP :
if e >= -1 & e <= 0
AP(i) = 0 ;
end
if e > 0 & e < b
AP(i) = 1/b*E(i) ;
end
if e >= b & e <= 1
AP(i) = 1 ;
end
plot(E,AN,E,AZ,E,AP),grid
xlabel('error'),axis([-1.2 1.2 -0.2 1.2])

```

Dan memberikan hasil gambaran fungsi keanggotaan error seperti gambar 3.3.



Gambar 3.3. Representasi Fungsi Keanggotaan Error

Program utama untuk derivative error adalah sebagai berikut :

```

% Menggambarkan Membership
Processors
% untuk error-dot de :

b = 0.4; % batasan N dan P
a = 0.2; % batasan Z
m = 0.1 ;

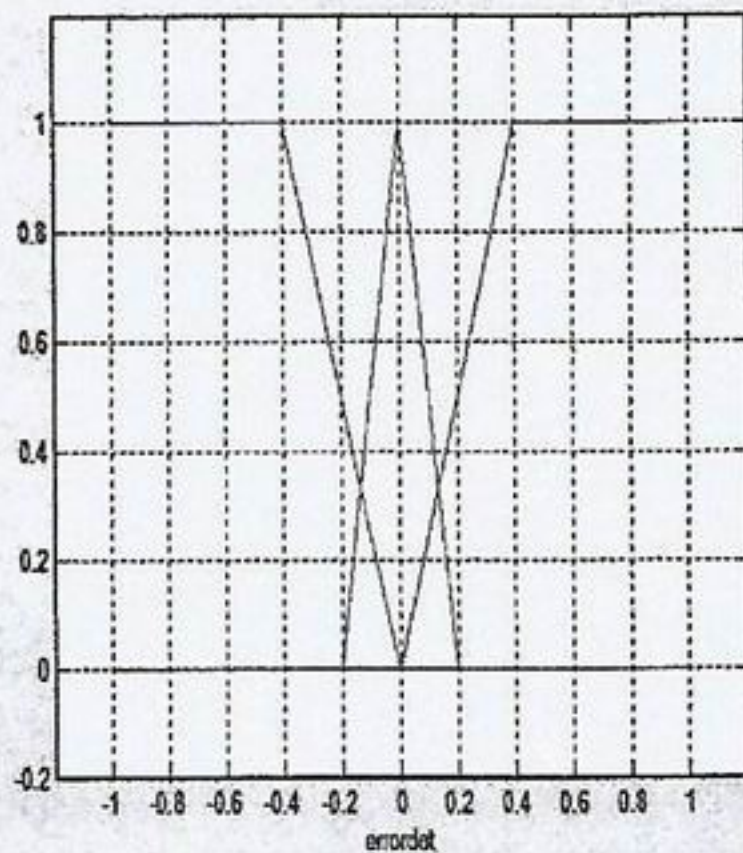
```

```

% pembagi antara -1 dan 1
for i = 1:21
% Memberships of de :
de = (i-11)/10 ;
DE(i) = de ;
% 1. Menggambar BN :
if de >= -1 & de <= -b
BN(i) = 1 ;
end
if de > -4/10 & de < 0
BN(i) = -1/b*DE(i) ;
end
if de >= 0 & de <= 1
BN(i) = 0 ;
end
% 2. Menggambar BZ :
if de >= -1 & de <= -a
BZ(i) = 0 ;
end
if de > -a & de <= 0
BZ(i) = 1/a*(DE(i)+a);
end
if de > 0 & de < 2/10
BZ(i) = -1/a*(DE(i)-a);
end
if de >= 2/10 & de <= 1
BZ(i) = 0 ;
end
% 3. Menggambar BP :
if de >= -1 & de <= 0
BP(i) = 0 ;
end
if de > 0 & de < b
BP(i) = 1/b*DE(i) ;
end
if de >= b & de <= 1
BP(i) = 1 ;
end
end
plot(DE,BN,DE,BZ,DE,BP),grid
xlabel('error-dot'), axis([-1.2 1.2 -0.2 1.2])

```

Dan memberikan hasil gambaran fungsi keanggotaan error seperti gambar 3.3.



Gambar 3.4. Representasi Fungsi

Keanggotaan Derivative Error

Dapat disimpulkan bahwa hasil pemrograman cukup bagus sesuai dengan teori.

III.4. Membangun Program Defuzzifikasi

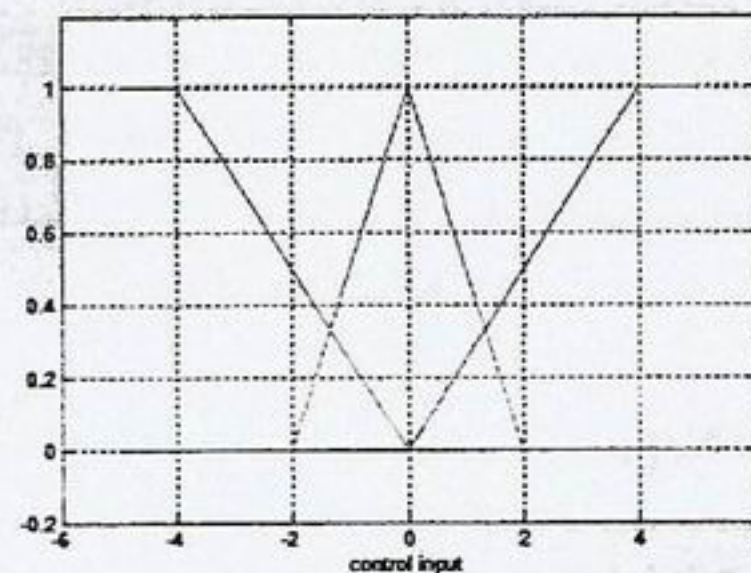
Dalam simulasi diperlukan menentukan nilai control input setiap saat. Program utama untuk control input adalah sebagai berikut :

```
%Menggambarkan Membership Processors
% untuk sinyal kendali CI :

b = 4 ; % batasan N dan P
a = 2 ; % batasan Z
m = 0.1 ;
% pembagi antara -1 dan 1
for i = 1:11
    % Memberships of ci :
    c = (i-6) ;
    CI(i) = c ;
    % 1. Menggambar JN :
    if c >= -5 & c <= -b
        JN(i) = 1 ;
    end
    if c > -4 & c < 0
        JN(i) = -1/b*CI(i) ;
    end
    if c >= 0 & c <= 5
```

```
        JN(i) = 0 ;
    end
    % 2. Menggambar JZ :
    if c >= -5 & c <= -a
        JZ(i) = 0 ;
    end
    if c > -a & c <= 0
        JZ(i) = 1/a*(CI(i)+a) ;
    end
    if c > 0 & c < a
        JZ(i) = -1/a*(CI(i)-a) ;
    end
    if c >= a & c <= 5
        JZ(i) = 0 ;
    end
    % 3. Menggambar JP :
    if c >= -5 & c <= 0
        JP(i) = 0 ;
    end
    if c > 0 & c < b
        JP(i) = 1/b*CI(i) ;
    end
    if c >= b & c <= 5
        JP(i) = 1 ;
    end
end
plot(CI,JN,CI,JZ,CI,JP),grid
xlabel('control input'), axis([-6 6 -0.2 1.2])
```

Dan memberikan hasil gambaran fungsi keanggotaan control input adalah seperti gambar 3.5.



Gambar 3.5. Representasi Fungsi Keanggotaan Control Input

Berbagai cara dilakukan dengan teknik pemrograman dalam matlab, maka

diperoleh program defuzzifikasi yang sudah diuji seperti berikut ini :

```
function Ccrisp = defuzzy (A,B)
% proses defuzzifikasi :

% Basis Pengetahuan matriks ( 3 x 3 ):
    J11N = min(A(1),B(1));  J12N =
min(A(1),B(2));
J13N = min(A(1),B(3));
J21N = min(A(2),B(1));
J22Z = min(A(2),B(2));
J23P = min(A(2),B(3));
J31P = min(A(3),B(1));
J32P = min(A(3),B(2));
J33P = min(A(3),B(3));
% parameter penguatan.
m = 1.0 ; % nilai absolut maksimum dan
minimum u.
g = 1*m ; f = 0.8*m ; e = 0.2*m ;
% standar buku
% penetapan nilai Ccrisp :
if J11N >=1
C11 = -g ;
else
C11 = -J11N/(1/f) ; % N
end
if J12N >=1
C12 = -g ; % N
else
C12 = -J12N/(1/f) ;
end
if J13N >=1
C13 = -g ; % N
else
C13 = -J13N/(1/f) ;
end
if J21N >=1
C21 = -g ; % N
else
C21 = -J21N/(1/f) ;
end
if J22Z ==0
C22 = g ; % Z
else
C22 = e*(1 - J22Z) ; % atau C22 = -
e*(1-J22Z)
```

```
end
if J23P >=1
C23 = g ; % P
else
C23 = J23P/(1/f) ;
end
if J31P >=1
C31 = g ; % P
else
C31 = J31P/(1/f) ;
end
if J32P >=1
C32 = g ; % P
else
C32 = J32P/(1/f) ;
end
if J33P >=1
C33 = g ; % P
else
C33 = J33P/(1/f) ;
end
Cnum = C11*J11N + C12*J12N +
C13*J13N + C21*J21N +
C22*J22Z + C23*J23P + C31*J31P +
C32*J32P + C33*J33P ;
Cden = J11N + J12N + J13N + J21N +
J22Z + J23P + J31P + J32P + J33P
;
Ccrisp = Cnum / Cden ;
% return.
```

Program Lengkap Ssistem kendali Logika Fuzzi

Kemudahan dalam melakukan perubahan paramter diperlukan dalam simulasi, oleh karena itu diperlukan cara yang mudah untuk memperoleh respon output yang diinginkan. Cara yang dilakukan adalah dengan membangun program lengkap untuk sebuah sistem kendali logika fuzzi dengan mengikut sertakan dua sub-program utama yaitu fuzzi dan defuzzi. Program yang dibangun dengan menggunakan semua data pada sub-bab sebelumnya adalah sebagaiberikut :

clear


```

% Plant :
    b = 3 ; a1 = 1 ; a0 = 0 ;
    ns = b ; ds = [1 a1 a0] ;
    T = 0.1 ;
    [nz,dz] = c2dm(ns,ds,T,'zoh') ;
% Simulasi :
% harga awal :
    yk = 0 ; yk1 = 0 ; yk2 = 0 ;
    rk = 1 ; uk1 = 0 ; uk2 = 0 ;
    ek1 = 0 ;
    K = 1.0 ;
% loop :
    for i = 1:150
        ek = rk - yk ;
        dek = ek - ek1 ;
        % membatasi harga e dan de
            if ek > 1
                ek = 1 ;
            end
            if ek < -1
                ek = -1 ;
            end
            if dek > 1
                dek = 1 ;
            end

            if dek < -1
                dek = -1 ;
            end

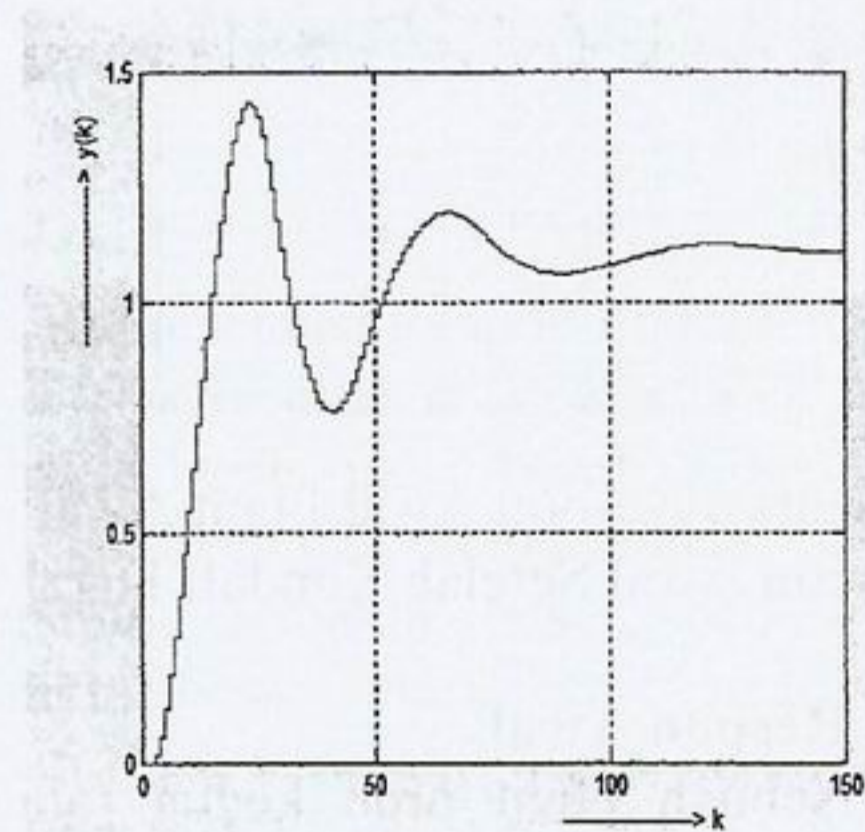
        % menentukan nilai fuzzifikasi dan defuzzifikasi :
            [A,B] = fuzzy(ek,dek) ;
            uk = K * defuzzy(A,B) ;
            y(i) = yk ;
            u(i) = uk ;
            yk = -dz(2)*yk1 - dz(3)*yk2 +
nz(2)*uk1 + nz(3)*uk2 ;
            yk2 = yk1 ; yk1 = yk ; uk2 = uk1 ;
            uk1 = uk ;
            ek1 = ek ;
        end
        % simulasi respon :
            stairs(y)
    xlabel('-----> k')

```

```

ylabel(' y(k)')
grid
% Selesai.

```

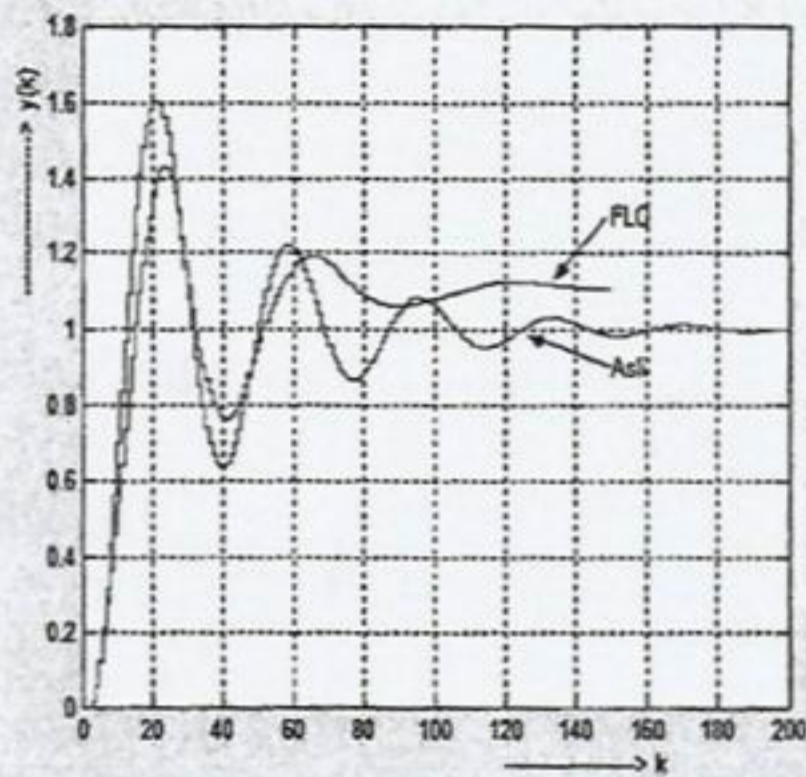


Representasi Respon Plant Awal

Program berhasil dijalankan, dan memberi hasil yang diharapkan. Dan pada tahap awal tanpa merubah spesifikasi awal, bahwa respon yang dihasilkan dengan sistem kendali logika fuzzy lebih baik, ditinjau dari perbandingan respon awa Gambar 3.2 dengan respon setelah dikendali fuzzy gambar 3.6.

Spesifikasi respon setelah dikendali terlihat bahwa overshoot adalah 40 % < dari sebelumnya yang 60 %, osilasi berkurang dan waktu steady state lebih cepat yaitu pada k = 80 sedangkan sebelumnya pada k = 120.

Perbandingan kedua respon dapat dilihat pada Gambar 3.7. Respon Awal ditandai dengan kata Asli, sedang respon setelah kendali fuzzy ditandai dengan FLC.

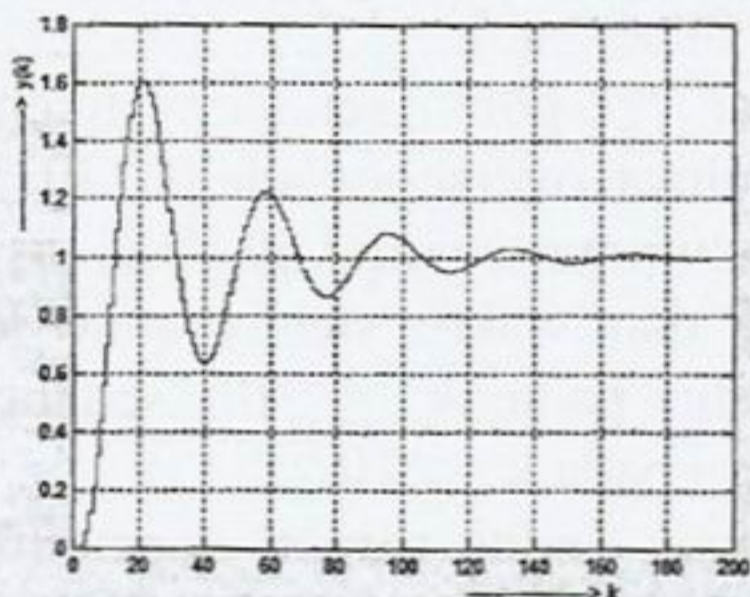


Representasi Respon Awal Plant Awal dan Respon Awal Setelah Kendali Fuzzi.

HASIL

Analisis Respon Awal

Ditinjau sebuah plant orde kedua (atau orde ke satu tipe 1 dengan umpan balik 1) yang memiliki respon awal seperti pada Gambar 4.1. Spesifikasi awal dengan overshoot 60%, berosilasi 3 kali, dan waktu steady state $k = 120$ dengan error adalah nol. Respon seperti ini dianggap kurang baik, maka overshoot haruslah maksimum 25 % dengan error ~ nol.



Representasi Respon Awal Plant

Diinginkan merancang sebuah sistem kendali digital logika fuzzi menggunakan tiga subset fuzzi yaitu error, derivative error dan control input dengan batas tertentu, untuk error dan derivative error adalah minimum dan maksimum $[-1, 1]$, sedangkan untuk control input dibuat batas minimum dan maksimum adalah $m \cdot [-1, 1]$, dimana m dapat diubah sesuai keperluan. Respon output diinginkan

memiliki nilai overshoot dan waktu steady state yang lebih baik, dan sedapat mungkin nilai error dipertahankan nol.

Perubahan Parameter Fungsi Keanggotaan Fuzzi

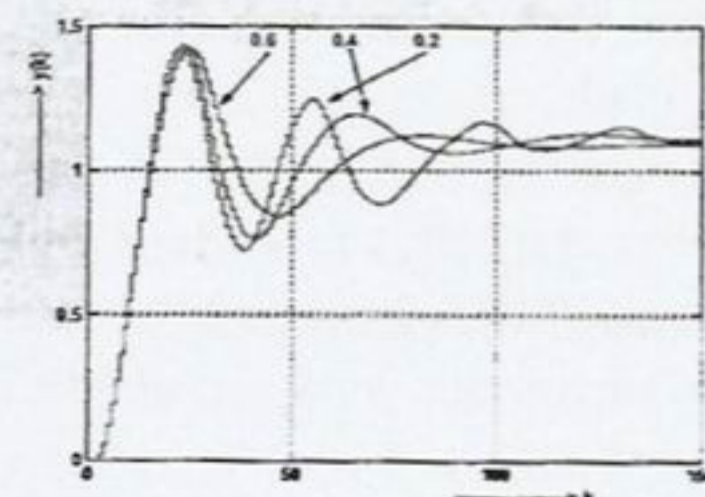
Parameter terdiri dari dua macam, parameter fuzzi dan parameter pengendali. Parameter fuzzi terdiri dari tiga macam, yaitu lebar bidang kurva N, Z dan P untuk derivative error, demikian juga untuk control input.

Kesemua parameter dapat saja diamati sebagai parameter utama fungsi keanggotaan sistem fuzzi. Oleh karena itu pengamatan parameter tersebut akan dilihat dari perubahan respon yang dihasilkan.

Perubahan Parameter Fungsi Keanggotaan Error

Parameter error akan menentukan kemiringan error seperti terlihat pada Gambar 2.3 yaitu kemiringan daerah zero Z, dan kemiringan daerah negatp-positip N-P.

Parameter awal dari 0,4 dibuat berubah untuk 0,2 sampai 0,6 sehingga akan menghasilkan tiga keadaan, dan setelah diujikan dengan mengubah parameter a sebanyak tiga kali pada sub-program fuzzi, maka diperoleh respon seperti pada Gambar 4.2.

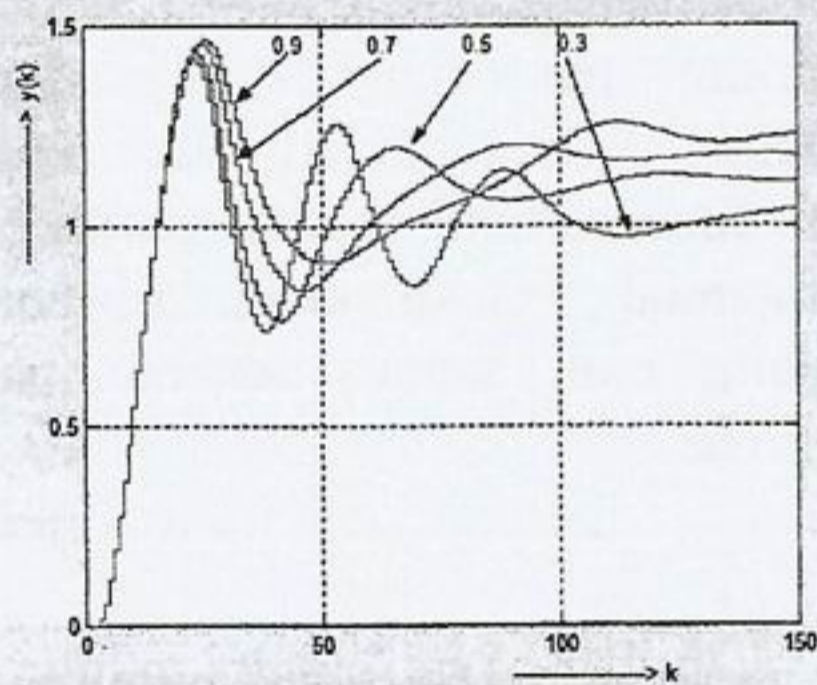


Representasi Respon Plant dengan Perubahan Parameter

Semakin besar nilai a berubah dari 0,2 ke 0,4 dan ke 0,6 menunjukkan bahwa overshoot tidak berubah, osilasi akan semakin kecil dan error steady state

dimati tidak berubah. Maka angka terbaik adalah $a = 0,6$.

Parameter awal dari 0,5 dibuat dari 0,3 sampai 0,9 sehingga akan menghasilkan tiga keadaan, dan setelah diujikan sebanyak empat kali pada sub-program fuzzy, maka diperoleh respon seperti pada Gambar 4.3.

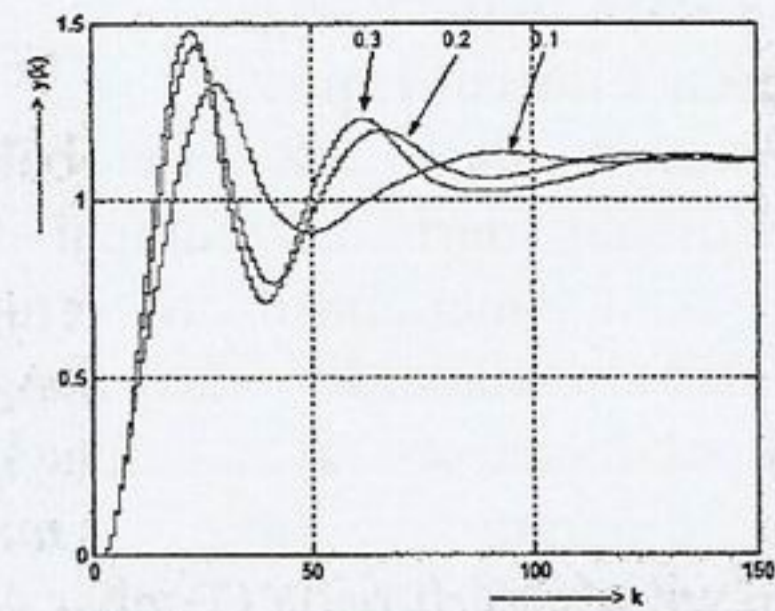


Representasi Respon Plant dengan Perubahan Parameter

Semakin besar nilai b berubah dari 0,3 ke 0,5 ke 0,7 dan ke 0,9 menunjukkan bahwa overshoot hampir tidak banyak berubah, osilasi akan semakin kecil dan error steady state dimati semakin besar. Ada kontroversi antara osilasi dan error steady state. Maka angka terbaik agak sulit ditentukan.

Perubahan Parameter Fungsi Keanggotaan Derivative Error

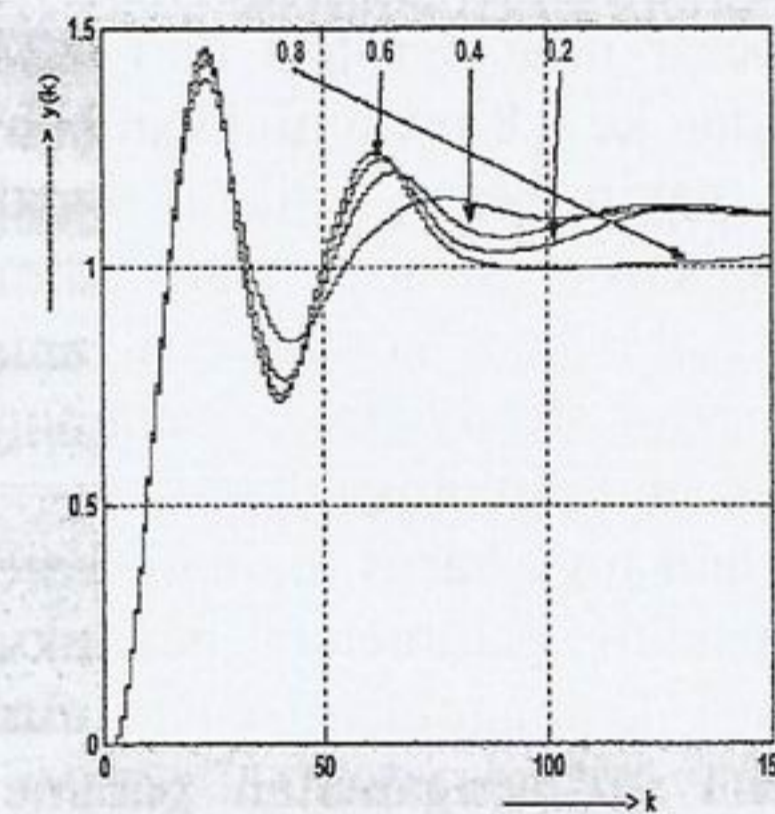
Parameter c dan d akan menentukan kemiringan error seperti terlihat pada Gambar 2.3. Parameter c menentukan batas kemiringan daerah zero Z pada derivative error, sedang parameter d menentukan batas kemiringan daerah negatif-positif $N-P$ pada derivative error. Parameter awal dari 0,2 dibuat dari 0,1 sampai 0,3 sehingga akan menghasilkan tiga keadaan, dan setelah diujikan maka diperoleh respon seperti pada Gambar 4.4.



Representasi Respon Plant dengan Perubahan Parameter

Semakin besar nilai c berubah dari 0,1 ke 0,2 dan ke 0,3 menunjukkan bahwa overshoot semakin besar, osilasi akan semakin besar tetapi error steady state dimati tidak berubah. Maka angka terbaik adalah $a = 0,1$.

Parameter awal dari 0,4 dibuat dari 0,2 sampai 0,8 sehingga akan menghasilkan empat keadaan, dan setelah diujikan maka diperoleh respon seperti pada Gambar 4.5

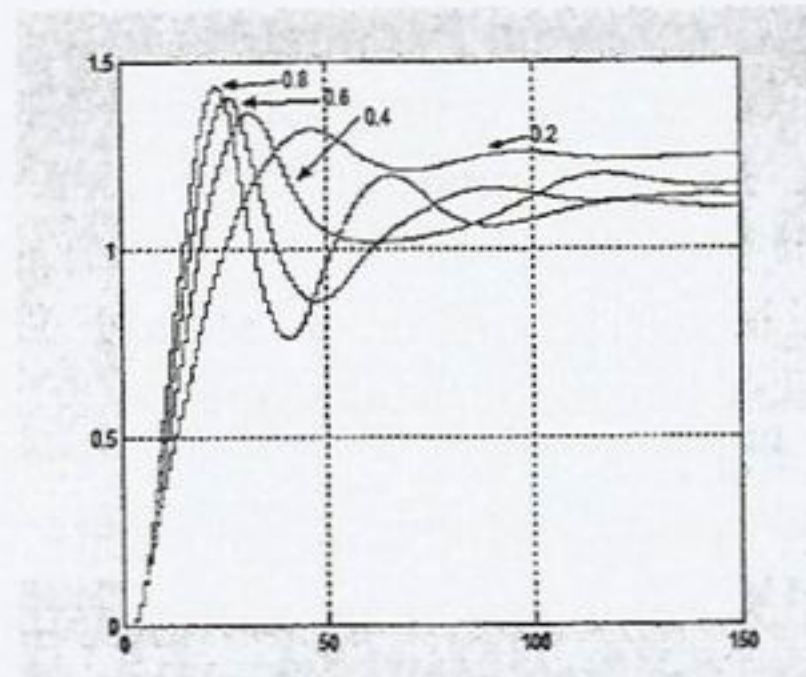


Representasi Respon Plant dengan Perubahan Parameter

Semakin besar nilai d berubah menunjukkan bahwa overshoot hampir tidak banyak berubah, osilasi tidak juga banyak berubah dan error steady state dimati semakinkecil. Maka angka terbaik adalah 0,8.

Perubahan Parameter Fungsi Keanggotaan Control Input

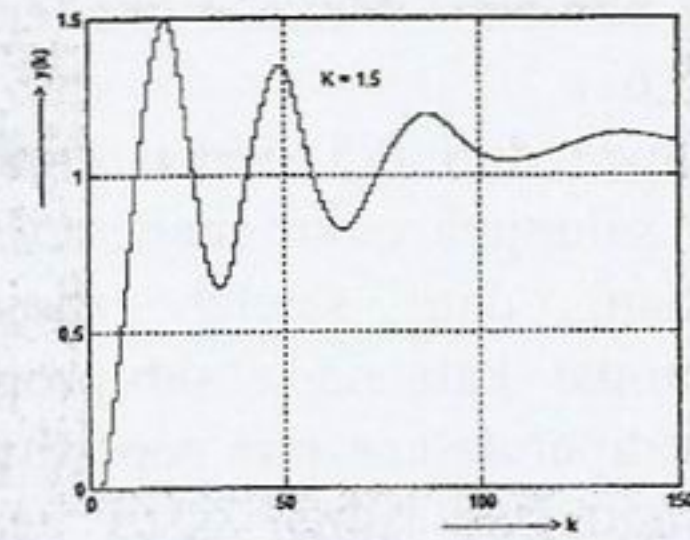
Parameter awal dari 0,8. Perubahan parameter dibuat dari 0,2 sampai 0,8 sehingga akan menghasilkan empat keadaan, dan setelah diujikan dengan mengubah parameter sebanyak empat kali pada sub-program defuzzi, maka diperoleh respon seperti pada Gambar 4.7.



Representasi Respon Plant dengan Perubahan Parameter f

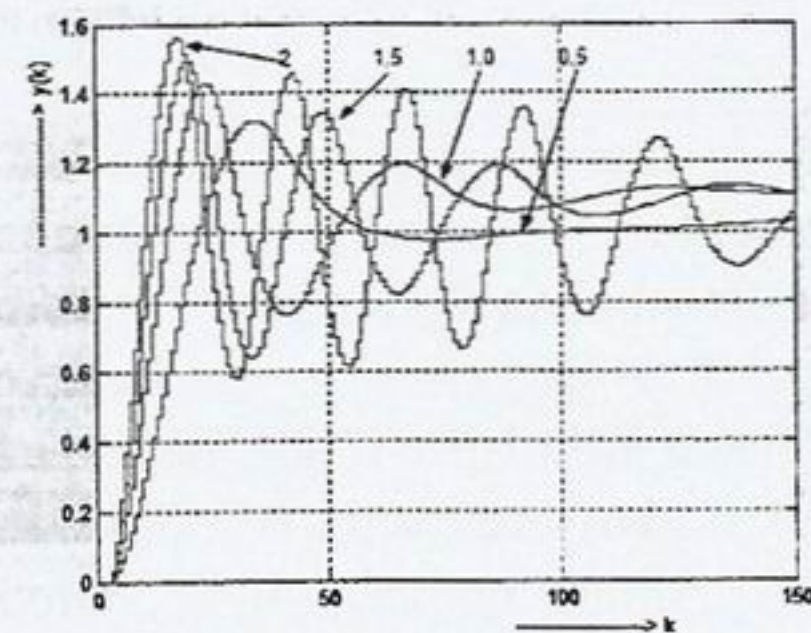
Semakin besar nilai berubah dari 0,2 ke 0,4 ke 0,6 dan ke 0,8 menunjukkan bahwa overshoot makin besar, osilasi semakin besar dan error steady state dimati semakin kecil. Ada kontroversi antara osilasi dan error steady state, maka angka terbaik agak sulit ditentukan.

Parameter lain juga harus diamati, karena ini merupakan parameter pendukung untuk fungsi keanggotaan sistem fuzzy. Oleh karena itu pengamatan parameter tersebut akan dilihat dari perubahan respon yang dihasilkan, akan tetapi perlu diingat bahwa jika satu parameter diamati, maka semua parameter lainnya tidaklah berubah atau dibuat konstan, sehingga setelah diujikan dengan mengubah parameter m pada sub-program defuzzi, maka diperoleh respon seperti pada Gambar 4.14.



Representasi Respon Plant dengan Perubahan Parameter $K = 1,5$

Jika seorang perancang sudah dapat memadukan perilaku dari ke enam parameter fuzzy ditambah dua parameter gain pengendali, maka akan diperoleh respon yang baik seperti terlihat pada Gambar 4.16.



Representasi Respon Plant dengan Perubahan Parameter K

KESIMPULAN

Program teknik kontrol atau teknik kendali belum banyak di wilayah Sumatera Utara, oleh karena itu masih sedikit penelitian tentang sistem kendali fuzzy. Beberapa orang mahasiswa di PSTE telah mencoba membuat tugas akhir mereka dengan awal pengendali fuzzy, tetapi belum sampai analisis tentang fungsi keanggotaan sistem fuzzy.

Dari hasil penelitian yang dilakukan, diberikan kesimpulan dan saran sebagai berikut ini :

1. Sistem Kendali Logika Fuzzy telah berhasil dirumuskan untuk plant dengan single input single output, plant dibuat berparameter tetap, dan sistem set-fuzzy dibatasi hanya sampai

minimum, zero dan maksimum, dengan membangun subprogram fuzzifikasi dan defuzzifikasi berbasis matlab 6,1 untuk sistem atau plant orde kedua. Perilaku dari parameter subset fuzzy telah dapat ditunjukkan, dan penelitian awal ini dianggap berhasil.

2. Titik kunci keberhasilan dari fuzzy harus dipastikan pada pemilihan parameter a,b,c,d,e dan f, karena itulah yang mendasari kendali fuzzy. Oleh karena itu diperlukan pengamatan yang lebih serius dalam menentukan batas kemiringannya, barangkali diasumsi memang berbeda untuk plant yang berbeda.

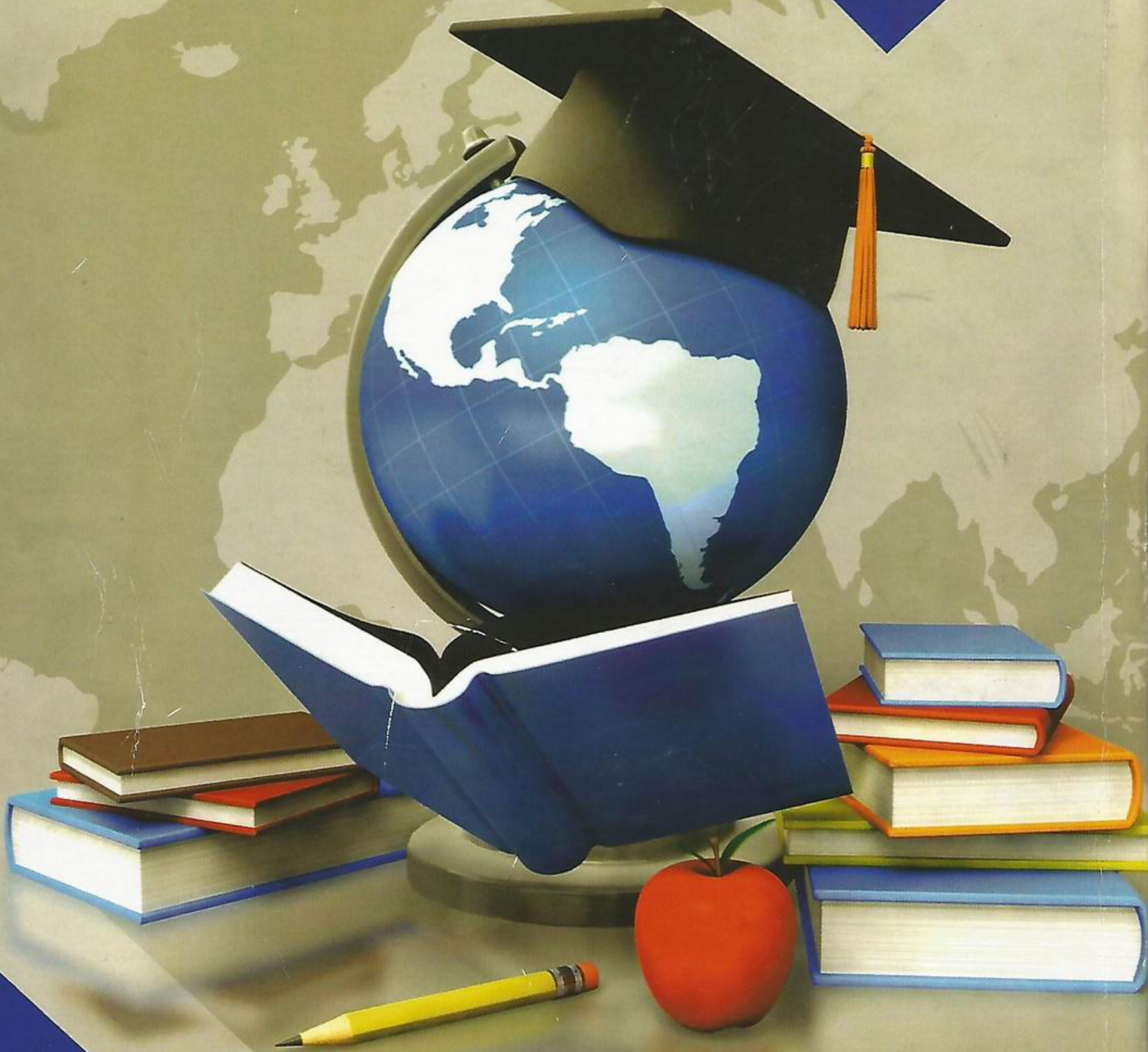
Saran dari peneliti ada tiga hal menyangkut kendali fuzzy yaitu;

1. Penelitian ini harus diteruskan lagi untuk plant dengan orde berbeda, baik stabil maupun tidak stabil.
2. Penelitian ini harus dikembangkan untuk pembagaian subset fuzzy yang lebih banyak misalnya 5 atau 7 atau 9, sehingga akan muncul istilah baru selain minimum-zero-maksimum, menjadi minimum-middle minimum-zero-middle maksimum dan maksimum.
3. Penelitian ini juga harus dikembangkan untuk sistem multi input multi output.

3. Mohammed Jamsidi & Nader Vadiee, Fuzzy Logic and Control, Prentice-Hall International, Inc., New Jersey, 1993.
4. Katsuhiko Ogata, Teknik Kontrol Automatik (Sistem Pengaturan), Jilid 1, Penerbit Erlangga, Jakarta, 1985
5. Katsuhiko Ogata, Teknik Kontrol Automatik (Sistem Pengaturan), Jilid 2, Penerbit Erlangga, Jakarta, 1985
6. Bahram Shahian, Control System Design Using Matlab, Prentice -Hall International Editions, London, 1993
7. Ching-Fang Lin, Advanced Control Systems Design, Prentice-Hall Englewood Clifts, New Jersey, 1993
8. Charles L. Phillips & H. Troy Nagle, Digital Control System Analysis and Design, Prentice-Hall International Editions, London, 1990.

DAFTAR PUSTAKA

1. Sri Kutsumadewi, Analisis Desain Sistem Fuzzy Menggunakan Toolbox Matlab, Penerbit Graha Ilmu, Yogyakarta, 2002.
2. Sohail Iqbal & Nora Boumella, Fuzzy Controllers Recent Advanced In Theory and Applications, Published by Intech, 2012



STINDO PROFESIONAL JURNAL

Redaksi : Jl. Masjid Taufik No. 166 Medan, Telp. 0812 6363 020

Jl. Perbaungan No. 2 Medan. Telp. (061) 732 1118

Email : stindoprofesional@yahoo.com

ISSN 1035-5421



1223488413