

# Perancangan Kontroler PID-*Fuzzy* untuk Sistem Pengaturan *Cascade Level* dan *Flow* pada *Basic Process Rig 38-100*

Dwi Arkin Pritandi, Joko Susila, Eka Iskandar

Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

*e-mail*: arkin14@mhs.ee.its.ac.id.

**Abstrak**— Pada industri yang melibatkan sebuah proses, sistem pengaturan level dan *flow* sering dijumpai. Permasalahan yang sering muncul pada sistem pengaturan proses adalah munculnya gangguan yang membuat sistem tidak stabil. Sedangkan kontroler PID yang sering digunakan di industri, tidak mampu mengembalikan kestabilan sistem akibat gangguan dengan cepat. Dibutuhkan penalaan ulang pada kontroler ketika terjadi perubahan parameter pada sistem atau saat sistem berubah parameternya. Penalaan tersebut hanya dapat dilakukan oleh operator yang memiliki kemampuan mumpuni. Oleh karena itu, dibutuhkan sebuah metode pengaturan untuk menala parameter kontroler PID secara otomatis, sehingga sistem tetap dapat bekerja dengan baik saat terdapat gangguan. Konfigurasi *cascade* pada sistem pengaturan level dan *flow* menggunakan kontroler PID konvensional memberikan hasil yang kurang baik dengan perubahan level lebih banyak saat perubahan parameter *plant*. Hal ini dikarenakan *inner loop* bekerja sangat cepat dan tidak ada perubahan parameter kontroler saat terjadi gangguan. *Fuzzy Logic* yang digunakan untuk menala parameter PID berdasarkan kesalahan dan perubahan kesalahan pada sistem, mampu menangani permasalahan tersebut. Kontroler PID-*Fuzzy* yang dikonfigurasi *cascade* juga mampu memberikan *recovery time* lebih cepat dari kontroler PID konvensional saat parameter sistem berubah. Hal ini dibuktikan dengan RMSE pada sistem yang menggunakan kontroler PID-*Fuzzy* adalah 16,85 dan sistem yang menggunakan kontroler PID konvensional adalah 16,91.

**Kata Kunci**—*Cascade, Flow, Fuzzy, Level, PID, PID-Fuzzy*

## I. PENDAHULUAN

Salah satu permasalahan dasar pada industri proses adalah pengaturan level suatu cairan. Pada saat proses pengaturan level sedang berlangsung seringkali muncul permasalahan pada instrumen-instrumen kontrol yang ada pada tangki. Misalnya, kerusakan instrumen ukur maupun gangguan-gangguan yang tidak diduga. Hal ini tentu akan mempengaruhi kerja sistem pengaturan level cairan tersebut.

Untuk mendapatkan performa yang baik dalam sistem pengaturan level dan aliran maka dibutuhkan kontroler yang mampu menangani masalah yang muncul pada sistem. Kontroler PID memiliki kelebihan struktur yang sederhana serta mudah didesain dan diaplikasikan[1]. Hal ini mendasari kontroler yang banyak digunakan dalam industri adalah kontrol PID[1]. Kontroler ini memiliki parameter  $K_p$ ,  $\tau_i$ , dan  $\tau_d$  yang nilainya diperoleh dari perhitungan dengan melibatkan parameter sistem. Namun pada kenyataannya, kontroler PID tidak cukup untuk menangani gangguan-gangguan yang muncul pada sistem. Beberapa metode penalaan (*tuning*) praktis kontroler PID seperti yang

diajukan oleh Cohen-Coon pada [2] dan Ziegler-Nichols pada [3] pun masih belum mampu menangani permasalahan tersebut. Hal ini dikarenakan kontroler PID tidak mampu mempertahankan kriteria yang diinginkan ketika terjadi perubahan parameter yang terlalu besar dalam sistem. Kontroler ini memiliki tingkat kerja yang tinggi apabila *gain* kontroler tersebut diatur oleh seseorang yang memiliki tingkat pengalaman yang tinggi pada *plant* yang akan dikontrol.

Oleh karena itu, diperlukan sebuah kontroler yang dapat melakukan proses adaptasi dimana nilai *gain*-nya mampu mengikuti perubahan parameter *plant*, sehingga performansi sistem tetap terjaga dengan baik. Dalam salah satu kasus *autotuning*, parameter PID dapat beradaptasi dengan perubahan parameter *plant* yang melibatkan perhitungan menggunakan logika *fuzzy*.

Pada jurnal ini akan dirancang sebuah kontroler PID-*Fuzzy* dimana kontroler ini mampu menangani permasalahan diatas. Pada kontroler tersebut, *fuzzy* bertugas memilih parameter PID yang tepat berdasarkan error yang disebabkan adanya gangguan pada sistem. Selain itu, juga dilakukan pengaturan pada aliran yang masuk ke tangki. Konfigurasi ini dinamakan sistem pengaturan *cascade*. Ada dua tujuan dari kontroler dengan konfigurasi *cascade*, yaitu untuk mengeliminasi pengaruh gangguan dan meningkatkan kedinamisan performansi *control loop*. [4]

## II. SISTEM PENGATURAN LEVEL DAN FLOW

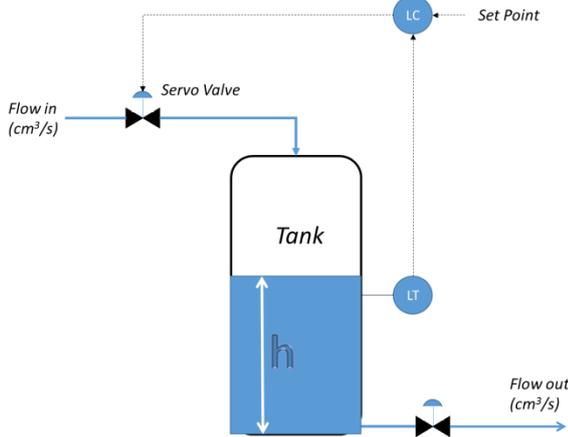
### A. Pemodelan Sistem Level Air Pada Tangki [5]

Gambar 1 menunjukkan *P&ID* dari sebuah tangki yang akan diatur level serta aliran masuknya. Pemodelan yang dilakukan pada penelitian ini menggunakan penurunan hukum-hukum fisik yang ada pada sebuah tangki yang akan diatur level airnya. Tangki diasumsikan seperti kapasitor dikarenakan sifatnya yang mirip. Proses pengisian air pada tangki seperti proses pengisian muatan sedangkan proses pembuangan air seperti proses pembuangan muatan pada kapasitor. Sehingga kapasitansi tangki dapat didekati dengan luas alas tangki yang berupa lingkaran seperti Persamaan 2.1.

$$C = \frac{\text{Perubahan volume air (cm}^3\text{)}}{\text{Perubahan ketinggian air (m)}} \quad (2.1)$$

$$C = \text{Luas penampang tangki} = \frac{\pi d^2}{4} = A$$

Pemodelan tangki dimulai dari hukum kesetimbangan massa pada Persamaan 2.2 dimana debit air didalam



Gambar 1 Sistem Pengaturan Level Air Pada Tangki [5]

merupakan selisih antara debit air yang masuk ke dalam tangki dan debit air yang keluar dari tangki.

$$Q_{in} - Q_{out} = Q \tag{2.2}$$

Hubungan antara debit dengan level air dalam tangki dirumuskan pada Persamaan 2.3.

$$Q = Av = A \frac{dh}{dt} \tag{2.3}$$

Dengan substitusi Persamaan 2.2 dan 2.1 pada Persamaan 2.3 sehingga hubungan antara debit air yang masuk dan yang keluar dituliskan pada Persamaan 2.4.

$$Q_{in} - Q_{out} = C \frac{dh}{dt} \tag{2.4}$$

*Control valve* diasumsikan seperti resistansi atau hambatan pada saat proses pembuangan air. Hal tersebut dikarenakan debit air yang keluar bergantung pada bukaan dari *control valve*, sehingga dapat dirumuskan pada Persamaan 2.5.

$$R = \frac{\text{Perubahan ketinggian air (cm)}}{\text{Perubahan debit air yang keluar (cm³/s)}} \tag{2.5}$$

Debit air yang melewati *control valve* adalah laminar karena tidak terdapat perubahan diameter dari pipa antara pompa dan *control valve* ditunjukkan oleh Persamaan 2.6.

$$R = \frac{h}{Q} \tag{2.6}$$

$$Q = \frac{h}{R} \tag{2.7}$$

Dengan melakukan substitusi hubungan antara debit air yang keluar dengan bukaan *valve* pada Persamaan 2.7 ke Persamaan 2.4, sehingga rumus kesetimbangan massa menjadi Persamaan 2.8.

$$Q_{in} - \frac{h}{R} = C \frac{dh}{dt} \tag{2.8}$$

$$Q_{in}R - h = RC \frac{dh}{dt} \tag{2.9}$$

Dengan menggunakan transformasi *Laplace* pada persamaan 2.10 maka diperoleh fungsi alih sistem pada Persamaan 2.12.

$$Q_{in}(s)R - H(s) = RCsH(s) \tag{2.10}$$

$$Q_{in}(s)R = H(s)(1 + RCs) \tag{2.11}$$

$$\frac{H(s)}{Q_{in}(s)} = \frac{R}{RCs + 1} \tag{2.12}$$

**B. Pemodelan Pompa [6]**

Pemodelan sensor bertujuan untuk mendapatkan nilai penguatan yang dilambangkan dengan  $K_2$ . Nilai  $K_2$  dirumuskan dengan rentang kerja dari pompa terhadap masukan berupa tegangan seperti Persamaan 2.13.

$$K_2 = \frac{\text{Rentang tegangan keluaran (V)}}{\text{Rentang ketinggian air dalam tangki (cm)}} \tag{2.13}$$

**C. Pendekatan Waktu Tunda Sistem Orde Pertama [4]**

Sistem dengan waktu tunda memiliki kelemahan dalam keperluan perancangan kontroler. Karena saat akan dirancang sebuah kontroler seperti PID misalnya, waktu tunda pada sistem akan diabaikan. Hal ini menyebabkan waktu tunda tidak berubah walaupun sistem telah diberi kontroler. Sedangkan di dunia industri, waktu tunda akan sangat mempengaruhi proses produksi.

Oleh karena itu, diperlukan sebuah metode untuk mendekati nilai waktu tunda agar menyatu dengan fungsi alih sistem. Dengan begitu, desain kontroler akan lebih mudah. Salah satu metode pendekatan waktu tunda yang digunakan adalah pendekatan deret orde pertama seperti pada Persamaan 2.14.

$$e^{-\alpha s} \cong \frac{1}{\left(\frac{\alpha s}{N} + 1\right)^N} \tag{2.14}$$

Dimana :

$\alpha$  = Waktu tunda sistem

$N$  = Orde hasil kompensasi yang diinginkan

**D. Kontroler PID (Proportional Integral Derivative) [1]**

Kontroler PID merupakan kontroler yang sudah sering digunakan dalam dunia industri. Keluaran dari kontroler ini dinyatakan pada Persamaan 2.15.

$$u(t) = K_c e(t) + \frac{K_c}{\tau_i} \int_0^t e(t) dt + K_c \tau_d \frac{de(t)}{dt} \tag{2.15}$$

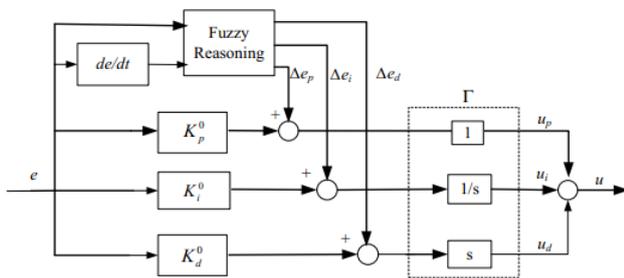
Dengan adanya bagian *derivative*, kontroler PID

mengantisipasi apa yang akan terjadi pada *error* sesaat yang akan datang kemudian melakukan aksi kontrol yang sebanding dengan kecepatan perubahan *error* saat ini. Berdasarkan sifat ini, aksi kontrol *derivative* kadang-kadang bertindak sebagai *anticipatory control*.

Efek dari kontroler PID ini adalah bila pada proses kesalahannya sangat besar, maka kontroler PI akan membutuhkan waktu yang panjang untuk mencapai *setpoint*nya, tetapi untuk kontroler PID akan mempercepat proses pencapaian *setpoint* tersebut.

**E. Kontroler PID-Fuzzy [8]**

Kontroler PID-Fuzzy merupakan koordinasi dari 2 macam kontroler yaitu PID dengan *fuzzy logic*. Fungsi dari *fuzzy logic* adalah menambah atau mengurangi nilai dari parameter PID ketika terdapat gangguan yang dapat merubah parameter sistem.



Gambar 2 Diagram Blok Kontroler PID-Fuzzy

Dengan perubahan parameter PID yang dilakukan oleh *fuzzy logic*, akan terjadi proses adaptasi oleh sistem terhadap gangguan yang terjadi. Nilai  $K_p$ ,  $T_i$  dan  $T_d$  saat diperbarui menjadi Persamaan :

$$\begin{aligned} K_p(t) &= K_{p_0} + \Delta K_p(t) \\ T_i(t) &= T_{i_0} + \Delta T_i(t) \\ T_d(t) &= T_{d_0} + \Delta T_d(t) \end{aligned} \quad (2.19)$$

**III. PERANCANGAN SISTEM**

**A. Pemodelan Sistem**

Pemodelan keseluruhan *plant* dapat dihitung dengan rumus seperti Persamaan 3.21. Untuk sistem pengaturan *flow* dapat dihitung menggunakan Persamaan 3.2. Dengan memasukkan parameter sistem yang didapat dari *plant*, didapatkan fungsi alih pada sistem pengaturan level Persamaan 3.3 dan sistem penagaturan *flow* pada persamaan 3.4.

$$\frac{H(s)_{total}}{Q_{in}(s)_{total}} = K1.e^{-\theta s} \cdot \frac{R}{RCs+1} \cdot K2 \quad (3.1)$$

$$F(s) = \frac{1}{RCs+1} K1.K2.e^{-\theta s} \quad (3.2)$$

$$G(s) = \frac{0,105}{7,96s+1} e^{-1,04s} \quad (3.3)$$

$$G(s) = \frac{1,46}{7,96s+1} e^{-0,56s} \quad (3.4)$$

Jika Waktu Tunda ( $\theta$ ) yang terjadi sangat kecil maka nilai N dapat dipilih 1. Maka, Fungsi Alih dari sistem pengaturan level menjadi orde 2 seperti pada persamaan 3.6.

$$G(s) = \frac{0,748}{7,96s+1} \cdot \frac{1}{\left(\frac{1,04}{1}s+1\right)} \quad (3.5)$$

$$G(s) = \frac{0,91}{8,27s^2+9s+1} \quad (3.6)$$

Sedangkan untuk model sistem pengaturan *flow* ditunjukkan pada persamaan 3.8.

$$G(s) = \frac{1,46}{7,96s+1} \cdot \frac{1}{\left(\frac{0,56}{1}s+1\right)} \quad (3.7)$$

$$G(s) = \frac{1,46}{4,45s^2+8,52s+1} \quad (3.8)$$

**B. Perancangan Kontroler PID**

Berdasarkan fungsi alih yang telah didapat dari persamaan 3.6 dan 3.8, akan dirancang sebuah kontroler PID pada sistem pengaturan *flow* dan sistem pengaturan level.

**1) Sistem Pengaturan Flow**

Berdasarkan fungsi alih dari sistem pengaturan *flow* seperti pada Persamaan 3.8, diketahui nilai  $\omega_n=0,47$  dan nilai  $\xi=2,002$ . Akan dirancang kontroler PID dengan spesifikasi konstanta waktu hasil rancangan 3 kali lebih cepat dari sistem seperti pada Persamaan 3.9.

$$\tau = 3\tau^* \quad (3.9)$$

Walaupun sistem adalah orde kedua, namun respon sistem menyerupai orde pertama. Maka untuk mengetahui nilai konstanta waktu ( $\tau$ ), digunakan rumus sistem orde pertama seperti pada Persamaan 3.10.

$$\tau = 0,632 \cdot X_{ss} \quad (3.10)$$

Dengan analisa dari respon *open loop*, didapat nilai  $\tau$  adalah 9,86 detik. Seperti yang telah dibahas pada BAB II, jika nilai  $\tau_i$  ditentukan seperti pada Persamaan 3.11,  $\tau_d$  pada Persamaan 3.12 dan  $K_p$  pada Persamaan 3.13.

$$\tau_i = \frac{2\xi}{\omega_n} \quad (3.11)$$

$$\tau_d = \frac{1}{\tau_i \omega_n^2} \quad (3.12)$$

$$K_p = \frac{\tau_i}{\tau^* K} \quad (3.13)$$

Maka dihitung nilai  $\tau_d$ ,  $\tau_i$ , dan  $K_p$  seperti pada Persamaan 3.14, 3.15 dan 3.16.

$$\tau_i = \frac{2,2,002}{0,47} = 8,52 \tag{3.14}$$

$$\tau_d = \frac{1}{8,52 \cdot 0,229} = 0,512 \tag{3.15}$$

$$K_p = \frac{8,52}{3,3,146} = 1,18 \tag{3.16}$$

2) Sistem Pengaturan Level

Berdasarkan fungsi alih dari sistem pengaturan level seperti pada Persamaan 3.6, diketahui nilai  $\omega_n = 0,064$  dan nilai  $\xi = 2,42$ . Akan dirancang kontroler PID dengan spesifikasi konstanta waktu hasil rancangan adalah 9,84 detik seperti pada Persamaan 3.17.

$$\tau^* = 9,84 \tag{3.17}$$

Pada sistem pengaturan level, respon sistem juga menyerupai orde pertama. Namun, karena nilai konstanta waktu respon awal tidak diperlukan, maka tidak perlu dicari.

Jika nilai  $\tau_i$  ditentukan seperti pada Persamaan 3.18,  $\tau_d$  pada Persamaan 3.19 dan  $K_p$  pada Persamaan 3.20.

$$\tau_i = \frac{2\xi}{\omega_n} \tag{3.18}$$

$$\tau_d = \frac{1}{\tau_i \omega_n^2} \tag{3.19}$$

$$K_p = \frac{\tau_i}{\tau^* K} \tag{3.20}$$

Maka dihitung nilai  $\tau_i$ ,  $\tau_d$  dan  $K_p$  seperti pada Persamaan 3.21, 3.22 dan 3.23.

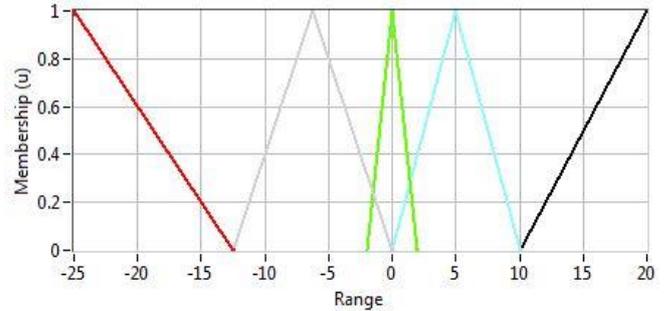
$$\tau_i = \frac{2 \cdot 1,56}{0,34} = 9 \tag{3.21}$$

$$\tau_d = \frac{1}{9 \cdot 0,00496} = 0,52 \tag{3.22}$$

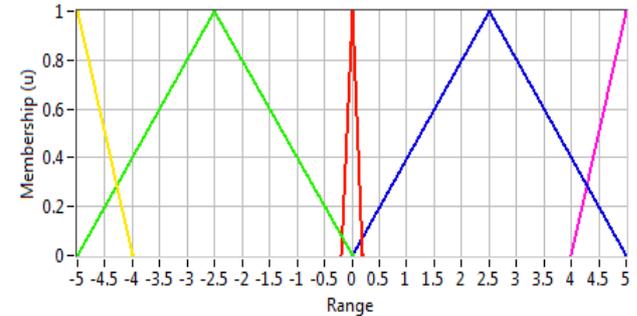
$$K_p = \frac{9}{9,38 \cdot 0,748} = 1,28 \tag{3.23}$$

C. Perancangan Kontroler PID-Fuzzy

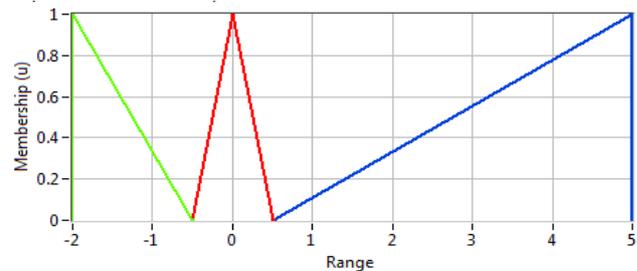
Perancangan *membership function* dari eror dan perubahan eror dibuat berdasarkan penentuan nilai maksimum dan nilai minimum dari eror dan perubahan eror ketika sistem diberikan gangguan. Nilai maksimum dan minimum tersebut dikuantisasikan sehingga menjadi sebuah *membership function*. Prosedur ini juga diterapkan pada penentuan *membership function* untuk sinyal keluaran yang merupakan perubahan nilai  $K_p$ ,  $T_i$  dan  $T_d$ .



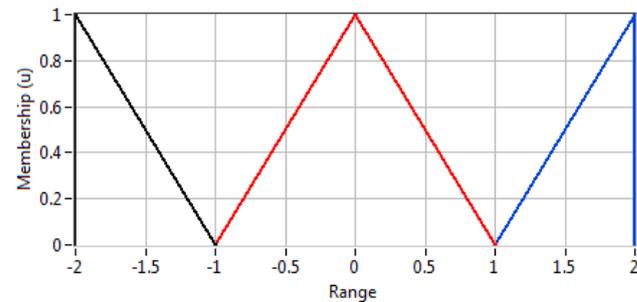
Gambar 3 Fungsi Keanggotaan Kesalahan



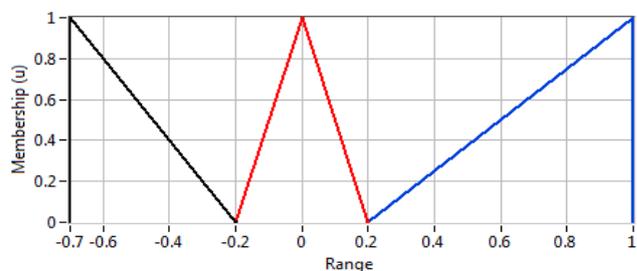
Gambar 4 Fungsi Keanggotaan Perubahan Kesalahan



Gambar 5 Fungsi Keanggotaan



Gambar 6 Fungsi Keanggotaan Fungsi Keanggotaan  $\Delta T_i$



Gambar 7 Fungsi Keanggotaan  $\Delta T_d$

Sedangkan basis aturan *fuzzy* didapat berdasarkan pengamatan dari karakteristik respon sistem ketika diberi gangguan hingga respon mencapai kondisi yang diinginkan. Terdapat 3 basis aturan yang disajikan dalam bentuk tabel. Basis aturan yang dibuat terdiri dari Basis aturan untuk  $\Delta K_p$ ,  $\Delta T_i$ , dan  $\Delta T_d$ .

Tabel 1.  
Basis Aturan  $\Delta Kp$

de \ e	NB	NS	ZE	PS	PB
NB	S	B	B	B	B
NS	S	B	B	B	B
ZE	S	B	ZE	B	B
PS	S	B	ZE	B	B
PB	S	B	ZE	B	B

Tabel 2.  
Basis Aturan  $\Delta Ti$

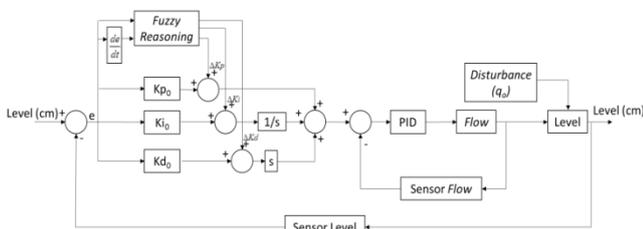
de \ e	NB	NS	ZE	PS	PB
NB	B	S	S	S	ZE
NS	B	S	S	ZE	S
ZE	S	B	ZE	B	S
PS	S	ZE	S	S	B
PB	ZE	S	S	S	B

Tabel 3.  
Basis Aturan  $\Delta Td$

de \ e	NB	NS	ZE	PS	PB
NB	B	ZE	ZE	ZE	B
NS	ZE	ZE	ZE	B	B
ZE	ZE	ZE	ZE	ZE	ZE
PS	B	ZE	ZE	B	ZE
PB	B	ZE	B	ZE	ZE

IV. HASIL PENGUJIAN DAN ANALISA

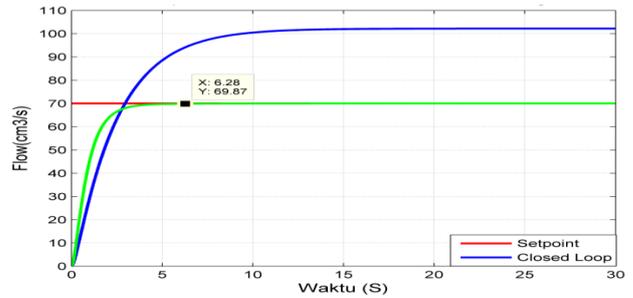
Untuk membuktikan bahwa kontroler PID-Fuzzy yang telah dibuat memiliki *recovery time* yang lebih cepat daripada kontroler PID biasa, maka dilakukan pengujian kontroler PID-fuzzy dengan cara diberi gangguan (*disturbance*) serta respon hasilnya dibandingkan dengan kontroler PID. Sistem pengaturan level dan *flow* akan dikonfigurasi *cascade* sesuai dengan diagram blok dibawah ini.



Gambar 8 Diagram Blok Sistem Keseluruhan

A. Respon Hasil Rancangan Kontroler PID untuk Sistem Pengaturan Flow

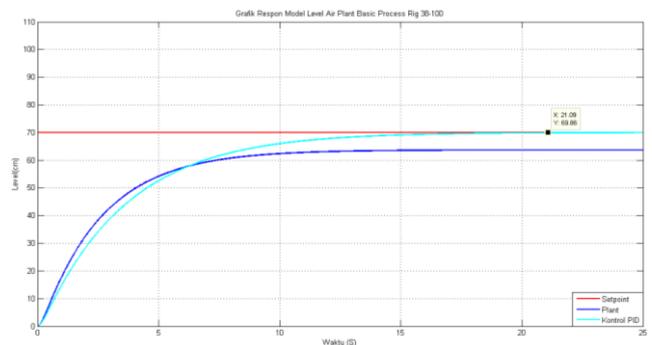
Berdasarkan hasil rancangan pada bagian sebelumnya, diperoleh waktu untuk mencapai kondisi tunak selama 6,28 detik. Artinya, konstanta waktu sistem dapat dihitung dari 63,2% waktu kondisi tunak yaitu sebesar 3,89 detik. Waktu ini sesuai dengan spesifikasi desain bahwa konstanta waktu sistem hasil rancangan 3 kali lebih cepat dari sistem asli.



Gambar 9 Respon Hasil Rancangan Sistem Pengaturan Flow

B. Respon Hasil Rancangan Kontroler PID untuk Sistem Pengaturan Level

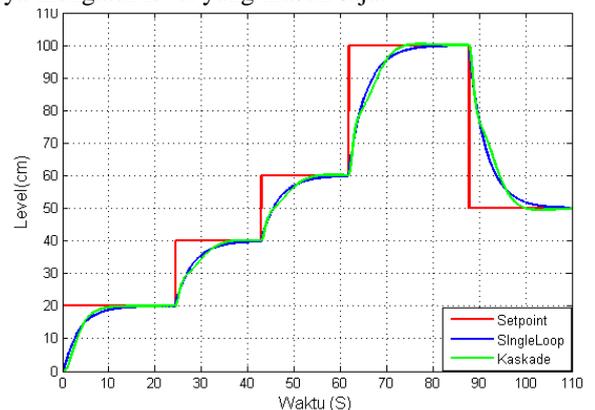
Berdasarkan hasil rancangan pada bagian sebelumnya, spesifikasi desain yang diinginkan adalah *settling time*(*ts*) sebesar 60 detik. Dari Gambar 10, diketahui waktu untuk mencapai kondisi tunak sebesar 21.09 detik. Artinya, sistem memiliki waktu kondisi tunak selama 66,6 detik. Ini membuktikan bahwa hasil rancangan telah sesuai dengan spesifikasi yang diinginkan.



Gambar 10 Respon Hasil Rancangan Sistem Pengaturan Level

C. Pengujian Kontroler PID Konfigurasi Cascade dengan Perubahan Set Point

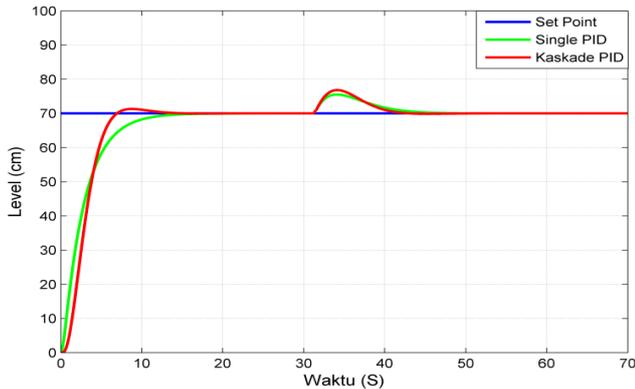
Mula-mula diberikan *set point* sebesar 20 cm selama 18 detik. Setelah itu, *set point* dinaikkan menjadi 40 cm selama 20 detik. Dapat dilihat pada Gambar 11, terdapat perbedaan pada kondisi *transient* respon dengan konfigurasi *cascade* dan respon dengan konfigurasi *single loop*. Sistem dengan konfigurasi *cascade* mampu mencapai keadaan tunak lebih awal daripada *single loop*. Hal ini disebabkan *flow* yang masuk ke tangki untuk mengatur level air juga ikut diatur jumlahnya. Sehingga menjadi kelebihan tersendiri daripada hanya mengatur level yang masuk saja.



Gambar 11 Pengujian Variasi Set Point pada Sistem Konfigurasi Cascade

**D. Pengujian Kontroler PID Konfigurasi *cascade* dengan Perubahan Parameter *Plant***

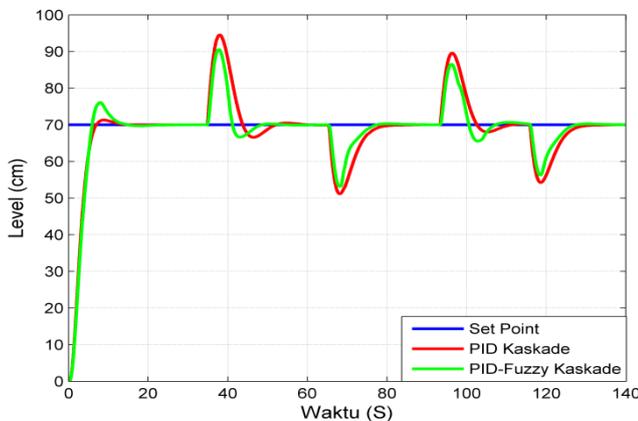
Kontroler PID konfigurasi *cascade* ini diuji dengan cara pada detik ke-52, diberikan gangguan berupa *flow out* sebesar 50cm<sup>3</sup>/s. Hal ini menyebabkan level turun sebesar 20 cm untuk single loop dan sebesar 24 cm untuk konfigurasi *cascade*. Walaupun lebih banyak nilai yang turun, waktu yang dibutuhkan pada konfigurasi *cascade* lebih sedikit dibanding *single loop*. Hal ini membuktikan bahwa konfigurasi *cascade* dapat mempercepat *recovery time* sistem ketika diberikan gangguan.



Gambar 12 Pengujian Sistem Konfigurasi Cascade dengan Perubahan Parameter Plant

**E. Pengujian Kontroler PID-Fuzzy Konfigurasi *Cascade* Dengan Perubahan Parameter *Plant***

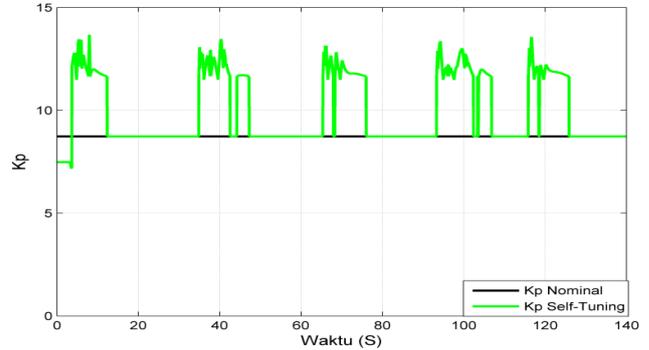
Gangguan yang diberikan pada pengujian ini berupa pengurangan nilai sinyal kontrol yang masuk pada sistem. Dapat dilihat pada Gambar 13 bahwa terdapat perbedaan pada kondisi *transient* respon dengan kontroler PID-fuzzy dan respon dengan kontroler PID. Sistem dengan kontroler PID-fuzzy mampu mencapai keadaan tunak lebih cepat daripada kontroler PID. Hal ini disebabkan ketika terjadi perubahan parameter pada sistem secara tiba-tiba, maka *error* akan menjadi besar dan sistem *fuzzy* menghitung perubahan nilai parameter PID berdasarkan *error* yang masuk. Untuk perubahan parameter PID dapat dilihat pada gambar 14.



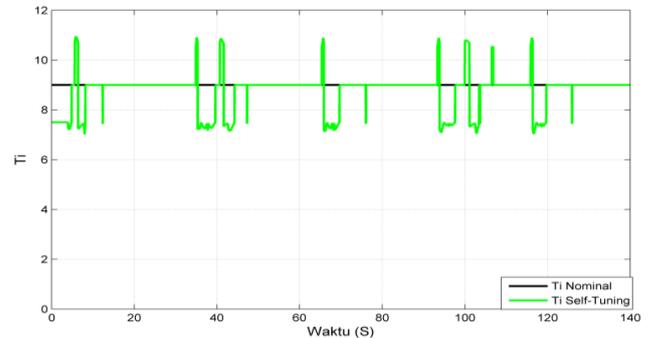
Gambar 13 Respon Sistem dengan Perubahan Parameter Plant

Dengan adanya gangguan pada sistem, sinyal kontrol akan berubah agar sistem dapat kembali menuju *set point*. Pada gambar 14, sinyal kontrol dari *Ti* yang paling dominan

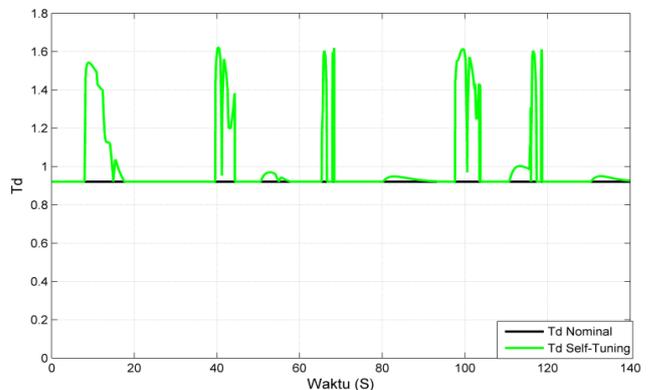
karena fungsi dari *Ti* sendiri adalah menghilangkan kesalahan kondisi tunak (%*Ess*). Begitupun dengan parameter *Kp* yang naik nilainya ketika ada gangguan. Fungsinya untuk mempercepat respon mencapai *set point*. Sedangkan nilai *Td*, berubah tidak terlalu banyak. Karena fungsi *derivative* sendiri adalah meredam *overshoot* dan hanya bekerja pada kondisi *transient*.



Gambar 14 Perubahan Nilai *Kp* terhadap Perubahan Parameter Plant



Gambar 15 Perubahan Nilai *Ti* terhadap Perubahan



Gambar 16 Perubahan Nilai *Td* terhadap Perubahan Parameter Plant

**V. KESIMPULAN**

Kesimpulan yang dapat kami ambil dari pengujian yang telah kami lakukan bahwa konfigurasi *cascade* pada sebuah sistem tidak selalu memberikan performansi lebih baik pada sistem karena tergantung pada jenis kontroler yang digunakan dan spesifikasi desain yang diinginkan pada *primary* dan *secondary controller* pada sistem tersebut. Selain itu, kontroler PID-Fuzzy mampu memberikan *recovery time* lebih cepat dan lebih tahan terhadap perubahan parameter *plant* karena adanya proses adaptasi berupa perubahan parameter kontroler PID terhadap perubahan kesalahan pada sistem. Sedangkan kontroler PID-Fuzzy yang kami buat memiliki *recovery time* yang

lebih cepat dan lebih tahan terhadap perubahan parameter *plant* dibanding dengan PID konvensional dibuktikan dengan nilai RMSE pada sistem dengan menggunakan kontroler PID-*Fuzzy* adalah 16,85 sedangkan sistem dengan menggunakan kontroler PID konvensional adalah 16,91.

## VI. UCAPAN TERIMA KASIH

Ucapan terima kasih penulis ucapkan pada bapak Joko Susila, M.T., Eka Iskandar, M.T., Imam Arifin M.T. dan M. Abdul Hady, M.T. yang banyak memberikan bimbingan, saran dan petunjuk selama pengerjaan penelitian ini. Ucapan terima kasih juga disampaikan kepada semua pihak yang turut membantu terselesaikannya penelitian ini.

## VII. DAFTAR PUSTAKA

- [1] Karl Johan Astrom and Tore Hagglund, "*Advanced PID Control*", USA: ISA-The Instrumentation, Systems, and Automation Society, 2005.
- [2] G.H. Cohen and G.A. Coon, "Theoretical Consideration of Retarded Control", in *Trans. ASME*, Vol. 75, pp. 827-834, 1953.
- [3] J.G. Ziegler and N.B. Nichols, "Optimum Settings for Automatic Controllers", in *Trans. ASME*, Vol. 64, pp. 759-768, 1942.
- [4] B.A. Ogunnaike and W.H. Ray, "*Process Dynamics, Modeling, and Control*", New York: Oxford University Press, 1994.
- [5] Ogata, Katsuhiko, "*Modern Control Engineering*", New Jersey: Prentice Hall, 1970.
- [6] Qashmal D., Rheza, "Desain Autotuning Kontroler PID Berbasis Algoritma Neural-Network untuk Sistem Pengaturan Cascade Level dan Flow Liquid pada Plant Coupled Tanks", *Tugas Akhir: Institut Teknologi Sepuluh Nopember*, 2015.
- [7] Yan, J., Ryan, M., Power, J, "*Using Fuzzy Logic*", UK: Prentice Hall, 1994.
- [8] Yuanhui Yang, Wailing Yang, Mingchun Wu, Qiwen Yang, Yuncan Xue, "A New Type of Adaptive Fuzzy PID Controller", *Intelligent Control and Automation (WCICA) World Congress*, pp. 5306-5310, 2010.
- [9] -----, "*PROCON - Level, Flow & Temperature 38-901-M Feedback Instruments, 1<sup>st</sup> Edition, I38S-P*", Manual Book, England: Feedback Instruments Ltd., 1999.
- [10] Zainal Abidin, Ali, "Perancangan Sistem Fuzzy untuk Autotuning Parameter Kontroler PID pada Sistem Cascade Flow dan Level Plant Model "S-4" Yokogawa", *Tugas Akhir : Institut Teknologi Sepuluh Nopember*, 2014.
- [11] Seborg, D.E., Edgar, T.F. and Mellichamp, D.A, "*Process Dynamic and Control (2<sup>nd</sup> Ed.)*", USA : John Willey & Sons Inc, 1989.