



TUGAS AKHIR - TF 141581

**PERANCANGAN SISTEM PENGENDALIAN  
LEVEL MENGGUNAKAN METODE *INTERNAL  
MODEL CONTROL* (IMC) PADA *DEAERATOR  
101U PLANT AMONIAK* PT. PETROKIMIA  
GRESIK**

Moammar Arief Hidayat  
NRP. 2411 100 111

Dosen Pembimbing  
Ir. Ya'umar, M.T.

JURUSAN TEKNIK FISIKA  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya  
2016



**FINAL PROJECT - TF 141581**

**DESIGN OF LEVEL CONTROL USING INTERNAL MODEL  
MODEL CONTROL (IMC) ON DEAERATOR 101U PLANT  
AMMONIA PT. PETROKIMIA GRESIK**

Moammar Arief Hidayat  
NRP. 2411 100 111

*Supervisor*  
Ir. Yau'mar, M.T

*DEPARTEMENT OF ENGINEERING PHYSICS  
Faculty Of Industrial Technology  
Sepuluh Nopember Institute Of Technology  
Surabaya  
2016*

**TUGAS AKHIR**

**PERANCANGAN SISTEM PENGENDALIAN LEVEL  
MENGUNAKAN METODE *INTERNAL MODEL  
CONTROL (IMC)* PADA *DEAERATOR 101U PLANT*  
AMONIAK PT. PETROKIMIA GRESIK**

Oleh :  
**Moammar Arief Hidayat**  
**NRP. 2411 100 111**

Surabaya, 28 Januari 2016  
Mengetahui/Menyetujui  
Pembimbing

  
**Ir. Ya'umar, M.T.**  
**NIPN. 19540406 198103 1 003**

**Ketua Jurusan  
Teknik Fisika FTI – ITS**

  
**Agus Muhammad Hatta, S.T., M.Si, Ph.D.**  
**NIPN. 19780902 200312 1 002**

## LEMBAR PENGESAHAN

### PERANCANGAN SISTEM PENGENDALIAN LEVEL MENGUNAKAN METODE *INTERNAL MODEL CONTROL (IMC)* PADA *DEAERATOR 101U PLANT* AMONIAK PT. PETROKIMIA GRESIK

#### TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Teknik  
Pada

Bidang Studi Rekayasa Instrumentasi  
Program Studi S-1 Jurusan Teknik Fisika  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember

Oleh :

**MOAMMAR ARIEF HIDAYAT**

NRP. 2411 100 111

Disetujui Oleh Tim Penguji Tugas Akhir :

1. Ir. Ya'umar, M.T. .... (Pembimbing)
2. Dr. Dhany Arifianto S.T., M.Eng .... (Ketua Penguji)
3. Ir. Matradji, M.Sc ..... (Penguji I)
4. Ir. Tutug Dhanardono, M.T. .... (Penguji II)
5. Arief Abdurrakhman S.T, M.T. .... (Penguji III)

SURABAYA  
JANUARI 2016

**PERANCANGAN SISTEM PENGENDALIAN LEVEL  
MENGUNAKAN METODE *INTERNAL MODEL  
CONTROL (IMC)* PADA *DEAERATOR 101U PLANT*  
AMONIAK PT. PETROKIMIA GRESIK**

**Nama Mahasiswa** : Moammar Arief Hidayat  
**NRP** : 2411 100 111  
**Program Studi** : S1 Teknik Fisika  
**Jurusan** : Teknik Fisika FTI-ITS  
**Dosen Pembimbing** : Ir. Ya'umar, M.T.

**Abstrak**

*Deaerator* adalah alat yang berfungsi untuk menghilangkan kandungan oksigen pada *make up water* sebelum masuk ke *boiler* dan *heat exchanger*. *Deaerator* bekerja dengan memanfaatkan kandungan oksigen yang terdapat pada *steam* untuk pelepasan kandungan oksigen pada *make up water*. Pengendalian level pada *deaerator* perlu dilakukan untuk menjaga kestabilan pasokan air ke *boiler* dan *heat exchanger*. Perancangan sistem pengendalian dengan metode *Internal Model Control* bertujuan agar respon pengendali memiliki performansi yang lebih baik daripada pengendalian PID. Dari hasil simulasi didapatkan bahwa pengendali IMC memiliki respon yang lebih baik pada saat uji *disturbance* dan mampu mencapai *set point* pada saat uji *tracking set point*. Parameter tuning IMC menunjukkan respon terbaik pada saat  $\lambda = 1$ . Parameter respon pengendali IMC yang didapatkan yaitu *settling time* = 14,56 s dan IAE = 20,92.

**Kata kunci :**

*Deaerator, Internal Model Control, PID, Set Point, Disturbance*

**DESIGN OF CONTROL LEVEL USING INTERNAL MODEL  
MODEL CONTROL (IMC) ON DEAERATOR 101U PLANT  
AMMONIA PT. PETROKIMIA GRESIK**

**Name of Student** : Moammad Arief Hidayat  
**Number Of Registration** : 2411 100 111  
**Major** : Bachelor Degree of Engineering  
Physics  
**Department** : Engineering Physics FTI-ITS  
**Supervisor** : Ir. Yau'mar, M.T.

**Abstract**

Deaerator is a tool that serves to eliminate the oxygen content of the make-up water before it enters the boiler and heat exchanger. Deaerator works by utilizing the oxygen content contained in the steam to release oxygen content of the make-up water. Control of the deaerator level necessary to maintain a stable water supply to the boiler and heat exchanger. The design of control systems with the Internal Model Control method aims to control the response has better performance than the PID control. From the simulation results showed that the IMC controller has better response at the time of the disturbance test and is able to achieve set point when the set point tracking test. IMC tuning parameters showed the best response when  $\lambda = 1$ . IMC controller response parameters were obtained, settling time = 14,56 s and IAE = 20,92.

**Keywords:**

Deaerator, Internal Model Control, PID, Set Point, Disturbance

## KATA PENGANTAR

Segala puji dan syukur kehadirat Allah SWT atas limpahan rahmat, hidayah dan kasih sayang-Nya serta shalawat dan salam atas Nabi Muhammad SAW, hingga terselesaikannya tugas akhir dengan judul, “Perancangan Sistem Pengendalian Level Menggunakan Metode *Internal Model Control* (IMC) Pada *Deaerator* 101U *Plant* Amoniak PT. Petrokimia Gresik.”

Sampai terselesaikannya tugas akhir ini, penulis telah banyak memperoleh bantuan dari berbagai pihak, baik langsung maupun tidak langsung. Untuk kesekian kalinya penulis mengucapkan terima kasih kepada :

1. Kedua orang tua yang telah memberikan dorongan secara moril maupun materil.
2. Bapak Agus Muhamad Hatta, S.T., M.Si, Ph.D selaku Ketua Jurusan Teknik Fisika FTI-ITS Surabaya.
3. Bapak Agus Muhammad Hatta, S.T., M.Si, Ph.D, selaku dosen wali dengan sabar memotivasi dan membimbing penulis selama kuliah di teknik fisika.
4. Bapak Ir. Ya,umar, M.T. selaku dosen pembimbing tugas akhir yang dengan sabar memotivasi dan membimbing penulis untuk menyelesaikan tugas akhir ini.
5. Bapak dan Ibu dosen penguji yang telah memberi banyak kritik dan saran
6. Seluruh Dosen dan Karyawan Jurusan Teknik Fisika FTI-ITS Surabaya.
7. PT. Petrokimia Gresik yang telah memberikan bimbingan pada saat pengambilan data tugas akhir.
8. Teman-teman satu angkatan Teknik Fisika F46.
9. Dan semua pihak yang telah mendukung dan tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa tugas akhir ini masih kurang dari sempurna, oleh karenanya saran dan kritik seluruh pihak diharapkan untuk membawa tugas akhir ini menuju pada kesempurnaan. Dan semoga tugas akhir ini bermanfaat bagi penulis khususnya dan bagi semua pihak yang berkepentingan. Aamin...

Surabaya, 28 Januari 2016

Penulis

## DAFTAR ISI

<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PENGESAHAN</b> .....	v
<b>ABSTRAK</b> .....	ix
<b>ABSTRACT</b> .....	xi
<b>KATA PENGANTAR</b> .....	xiii
<b>DAFTAR ISI</b> .....	xv
<b>DAFTAR GAMBAR</b> .....	xvii
<b>DAFTAR TABEL</b> .....	xix
<b>BAB I PENDAHULUAN</b>	
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	2
1.3 Tujuan Tugas Akhir.....	2
1.4 Batasan Masalah.....	2
1.5 Sistematika Laporan.....	3
<b>BAB II DASAR TEORI</b>	
2.1 <i>Deaerator</i> .....	5
2.1.1 <i>Pemodelan Matematis Deaerator</i> .....	7
2.2 <i>Internal Model Control (IMC)</i> .....	9
2.2.1 <i>Model Based IMC</i> .....	10
2.3 <i>Pendekatan FOPDT</i> .....	13
2.4 <i>Respon Transient</i> .....	14
<b>BAB III METODOLOGI PERANCANGAN SISTEM</b>	
3.1 <i>Alur Penelitian</i> .....	17
3.2 <i>Studi Literatur</i> .....	17
3.3 <i>Pengambilan Data</i> .....	18
3.4 <i>Pemodelan Sistem Open Loop</i> .....	19
3.5 <i>Validasi Model</i> .....	24
3.6 <i>Pemodelan Sistem Dengan Uji Disturbance</i> .....	27
3.6.1 <i>Uji Disturbance Perubahan Mass Flow Air Kondensat</i> .....	27

3.6.2 Uji <i>Disturbance</i> Perubahan <i>Mass Flow Steam</i> .....	29
3.7 Uji <i>Open Loop Gain Plant</i> .....	30
3.8 Perancangan <i>IMC Controller</i> .....	31
3.8.1 Uji <i>Controller</i> tanpa <i>Disturbance</i> .....	31
3.8.2 Uji <i>Controller</i> Dengan <i>Disturbance</i> .....	33
3.9 Perbandingan Respon Sistem.....	34

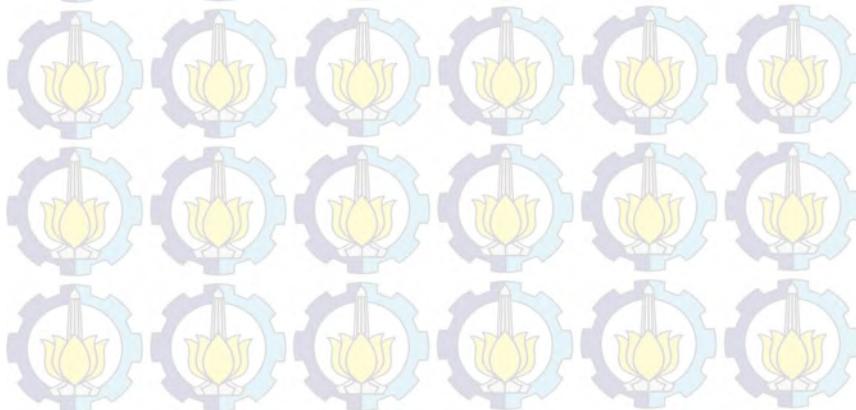
#### **BAB IV ANALISA DATA DAN PEMBAHASAN**

4.1 Hasil Simulasi Uji <i>Open Loop Gain Plant</i> .....	37
4.2 Hasil Perancangan <i>Controller IMC</i> .....	37
4.2.1 Hasil <i>Controller IMC</i> Tanpa <i>Disturbance</i> .....	37
4.2.2 Hasil <i>Controller IMC</i> Dengan <i>Disturbance</i> .....	39
4.3 Uji <i>Disturbance</i> .....	40
4.4 Hasil Perbandingan Performansi Pengendali.....	41
4.5 Pengujian <i>Tracking Set Point</i> Pengendali <i>IMC</i> .....	42
4.6 Pembahasan .....	43

#### **BAB IV KESIMPULAN DAN SARAN**

5.1 Kesimpulan .....	45
5.2 Saran .....	45

#### **DAFTAR PUSTAKA**



## DAFTAR GAMBAR

<b>Gambar 2.1</b>	<i>Deaerator</i> .....	5
<b>Gambar 2.2</b>	Fungsi <i>Deaerator</i> 101U.....	6
<b>Gambar 2.3</b>	Proses <i>Open Loop</i> Kontrol IMC.....	10
<b>Gambar 2.4</b>	Diagram Blok Struktur Kontrol IMC.....	12
<b>Gambar 2.5</b>	Analisa Respon Dengan Metode FOPDT.....	14
<b>Gambar 2.6</b>	Grafik Respon <i>Transient</i> .....	15
<b>Gambar 2.7</b>	Penentuan Nilai <i>Integral Absolute Error</i> (IAE).....	16
<b>Gambar 3.1</b>	<i>Flowchart</i> Penelitian Tugas Akhir.....	17
<b>Gambar 3.2</b>	Proses <i>Input</i> dan <i>Output Deaerator</i> .....	18
<b>Gambar 3.3</b>	<i>Layout</i> Simulasi Hysys.....	19
<b>Gambar 3.4</b>	Grafik Uji <i>Open Loop</i> Hysys.....	20
<b>Gambar 3.5</b>	Grafik Validasi Respon.....	24
<b>Gambar 3.6</b>	Grafik Uji <i>Disturbance</i> Perubahan <i>Mass Flow</i> Air Kondensat.....	28
<b>Gambar 3.7</b>	Grafik Uji <i>Disturbance</i> Perubahan <i>Mass Flow Steam</i> .....	29
<b>Gambar 3.8</b>	Uji <i>Open Loop Gain Plant</i> .....	31
<b>Gambar 3.9</b>	<i>Subsystem Plant</i> .....	31
<b>Gambar 3.10</b>	IMC Controller Tanpa <i>Disturbance</i> .....	33
<b>Gambar 3.11</b>	Uji Controller Dengan <i>Disturbance</i> .....	34
<b>Gambar 3.12</b>	<i>Subsystem Disturbance</i> Air Kondensat.....	34
<b>Gambar 3.13</b>	<i>Subsystem Disturbance Steam</i> .....	34
<b>Gambar 3.14</b>	Simulasi IMC Controller Dan PID Controller.....	35
<b>Gambar 4.1</b>	Grafik Uji <i>Open Loop Gain Plant</i> .....	37
<b>Gambar 4.2</b>	Respon IMC Controller Tanpa <i>Disturbance</i> .....	38
<b>Gambar 4.3</b>	Perbandingan Nilai $\lambda$ Dan IAE.....	39
<b>Gambar 4.4</b>	Respon IMC Controller Dengan <i>Disturbance</i> .....	39
<b>Gambar 4.5</b>	Respon Uji <i>Disturbance</i> Air Kondensat.....	40
<b>Gambar 4.6</b>	Respon Uji <i>Disturbance Steam</i> .....	41
<b>Gambar 4.7</b>	Respon Perbandingan Performansi Pengendali.....	42
<b>Gambar 4.8</b>	Simulasi <i>Tracking Set Point</i> Pengendali IMC.....	43
<b>Gambar 4.9</b>	Respon Uji <i>Tracking Set Point</i> Pengendali IMC.....	43

## DAFTAR TABEL

<b>Tabel 3.1</b>	Data PVT <i>Deaerator Uji Open Loop</i> .....	18
<b>Tabel 3.2</b>	Data Spesifikasi Dimensi <i>Deaerator</i> .....	19
<b>Tabel 3.3</b>	Parameter FOPDT Uji <i>Open Loop Hysys</i> .....	20
<b>Tabel 3.4</b>	Interpolasi 63%.....	22
<b>Tabel 3.5</b>	Interpolasi 28%.....	23
<b>Tabel 3.6</b>	Interpolasi 63% <i>Plant</i> .....	25
<b>Tabel 3.7</b>	Interpolasi 28% <i>Plant</i> .....	25
<b>Tabel 3.8</b>	Interpolasi 63% Model.....	26
<b>Tabel 3.9</b>	Interpolasi 28% Model.....	26
<b>Tabel 3.10</b>	Data PVT <i>Deaerator Uji Disturbance Air Kondensat</i> .....	27
<b>Tabel 3.11</b>	Parameter Pemodelan FOPDT Uji <i>Disturbance Air</i> <i>Kondensat</i> .....	28
<b>Tabel 3.12</b>	Data PVT <i>Deaerator Uji Disturbance Steam</i> .....	29
<b>Tabel 3.13</b>	Parameter Pemodelan FOPDT Uji <i>Disturbance</i> <i>Steam</i> .....	30
<b>Tabel 4.1</b>	Parameter Respon IMC <i>Controller Tanpa</i> <i>Disturbance</i> .....	38
<b>Tabel 4.2</b>	Parameter Respon IMC <i>Controller Dengan</i> <i>Disturbance</i> .....	40
<b>Tabel 4.3</b>	Perbandingan Parameter Respon Pengendali.....	42



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

*Deaerator* merupakan alat yang berfungsi untuk menghilangkan oksigen dan gas-gas yang terkandung dalam air yang akan dipakai di dalam *boiler*[1]. *Deaerator* 101U di PT. Petrokimia Gresik berfungsi untuk menghilangkan kandungan O<sub>2</sub> dari *make up water* yang nantinya akan dipompa masuk ke dalam *heat exchanger* dan *boiler* 101C. Pengendalian yang dilakukan menggunakan teknik pengendali PID. Kendala yang sering dialami adalah respon pengendalian PID lama mencapai *steady state*.

Variabel proses yang dikendalikan pada *deaerator* 101U adalah ketinggian air. Ketinggian air didalam *deaerator* sangat penting untuk dikendalikan. Level air digunakan sebagai acuan untuk mengetahui kadar oksigen yang terkandung di dalam air. Cairan hidrazin akan disuntikkan pada *deaerator* apabila level air terlalu tinggi.

Teknik pengendalian PID sering digunakan dalam industri karena dapat menghasilkan respon sistem yang cepat[2]. Berdasarkan survey, 97% industri menggunakan PID sebagai komponen utama dalam pengontrolannya[3]. Tetapi, masih terdapat kekurangan dalam pengendalian PID tersebut yakni metode tuning dengan *trial and error* untuk mendapatkan parameter PID. Tuning dalam pengendalian PID juga bisa dilakukan dengan cara *autotuning*. Selain itu, tuning PID juga memiliki kekurangan yaitu mempunyai nilai parameter *settling time* dan ketidakstabilan respon apabila *set point* dirubah dengan nilai yang cukup besar.

Dari beberapa kekurangan tersebut, maka diperlukan teknik pengendalian yang mampu mengatasi hal beberapa hal tersebut. Salah satu metode yang dapat digunakan adalah teknik pengendalian *Internal Model Control* (IMC). Kontrol IMC adalah metode pengendalian yang mencari *gain controller* dari model *plant* itu sendiri. IMC memiliki kinerja yang sangat efektif

sebagai *controller* terutama dua parameter tersebut yaitu *maximum overshoot* dan *settling time*[4].

Teknik pengendalian IMC sering diterapkan pada *power plant* dengan melakukan *tuning* terhadap perubahan *set point* dan *tuning* berdasarkan nilai *disturbance*. Pada *tuning disturbance* tersebut dapat mengurangi nilai gangguan yang langsung masuk pada keluaran sebuah proses (Biyanto, 2013). Pada penelitian sebelumnya yang dilakukan oleh Asana Kusnadi dengan judul perancangan pengontrol dan analisis respon pada sistem *Internal Model Control* (IMC) telah dilakukan pengendalian pada *heat exchanger* yang bertujuan untuk menghasilkan respon yang sesuai dengan *set point* dan mengurangi tingkat gangguan yang langsung masuk pada keluaran proses[12]. Oleh karena itu, dalam Tugas Akhir ini akan dirancang sebuah analisa pengendalian level menggunakan metode *internal model control* pada *deaerator* 101U *plant* amoniak PT. Petrokimia Gresik.

## 1.2 Perumusan Masalah

Dari uraian latar belakang tersebut, didapatkan beberapa permasalahan, yaitu :

1. Bagaimana cara merancang sistem pengendalian dengan menggunakan metode *Internal Model Control* (IMC).
2. Bagaimana performansi yang akan dihasilkan dari pengendalian level unit *deaerator* 101U.

## 1.3 Tujuan Tugas Akhir

Tujuan dari penelitian ini adalah menjawab permasalahan yang telah dikemukakan, yakni :

1. Mampu merancang pengendalian level menggunakan metode IMC pada *deaerator* 101U.
2. Menganalisa performansi yang akan dihasilkan dari pengendalian level unit *deaerator* 101U.

## 1.4 Batasan Masalah

Batasan masalah dalam tugas akhir ini antara lain :

1. *Deaerator* pada penelitian kali ini adalah *deaerator* 101U PT. Petrokimia Gresik.
2. Pemodelan *plant* dilakukan dengan *software* Hysys 7.3.
3. Analisa proses berdasarkan kesetimbangan massa saja.
4. Analisa respon dilakukan dengan *software* MatLab 2009a.
5. Pengendalian yang dilakukan adalah pengendalian level saja tanpa adanya analisa mengenai proses pelepasan oksigen yang terdapat pada proses *deaerator* 101U.

### **1.5 Sistematika Laporan**

Laporan tugas akhir ini akan disusun secara sistematis dibagi dalam beberapa bab, dengan perincian sebagai berikut :

#### **BAB I Pendahuluan**

Bab ini berisikan penjelasan latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, metodologi penelitian dan sistematika laporan.

#### **BAB II Tinjauan Pustaka**

Bab ini berisi penjelasan singkat tentang teori-teori yang digunakan dalam pengerjaan Tugas Akhir.

#### **BAB III Metodologi Penelitian**

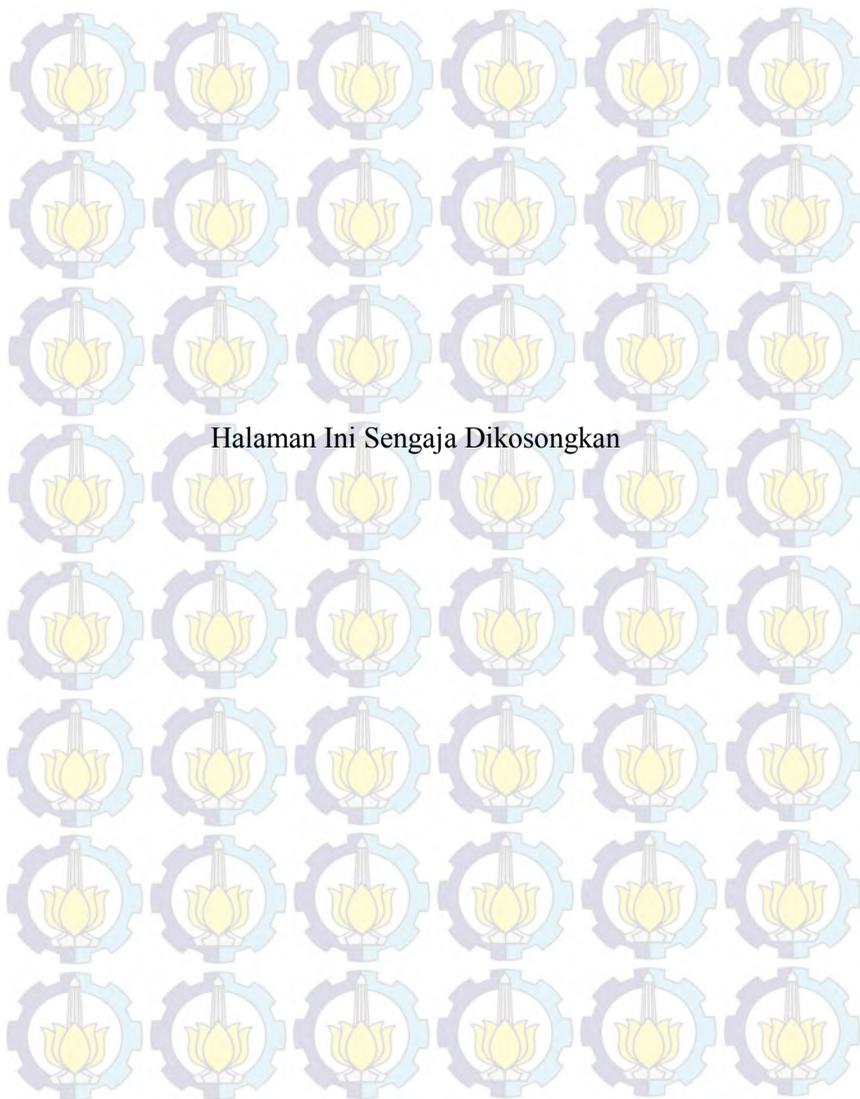
Bab ini berisi penjelasan detail mengenai langkah – langkah yang harus dilalui untuk mencapai tujuan dari penelitian Tugas Akhir ini yang meliputi perhitungan dan perancangan IMC pengendalian level *deaerator* yang telah dikaji.

#### **BAB IV Pengujian dan Analisa Hasil Simulasi**

Bab ini berisi tentang hasil simulasi yang diperoleh dari perancangan IMC pengendalian level *deaerator* yang didapatkan dalam BAB III, kemudian dilakukan analisa-analisa serta pembahasan yang berorientasi pada tujuan penelitian yang telah ditetapkan.

#### **BAB V Kesimpulan dan Saran**

Bab ini berisi tentang kesimpulan pokok dari seluruh rangkaian penelitian yang telah dilakukan dan saran yang dapat dijadikan sebagai pengembangan penelitian selanjutnya



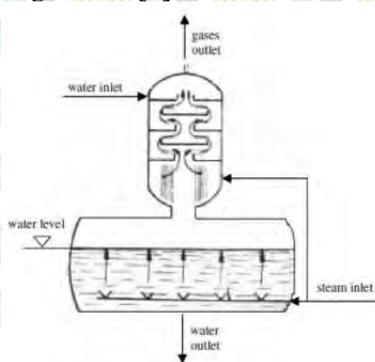
Halaman Ini Sengaja Dikosongkan

## BAB II DASAR TEORI

Pada bab ini berisi teori dasar tentang hal-hal yang berkaitan dengan judul tugas akhir, sehingga nantinya dapat membantu pola berpikir dalam menyelesaikan permasalahan yang terdapat di dalam penulisan. Dasar teori tersebut didapatkan melalui referensi *text book*, *manual book*, jurnal ilmiah yang telah dipublikasikan.

### 2.1 *Deaerator*

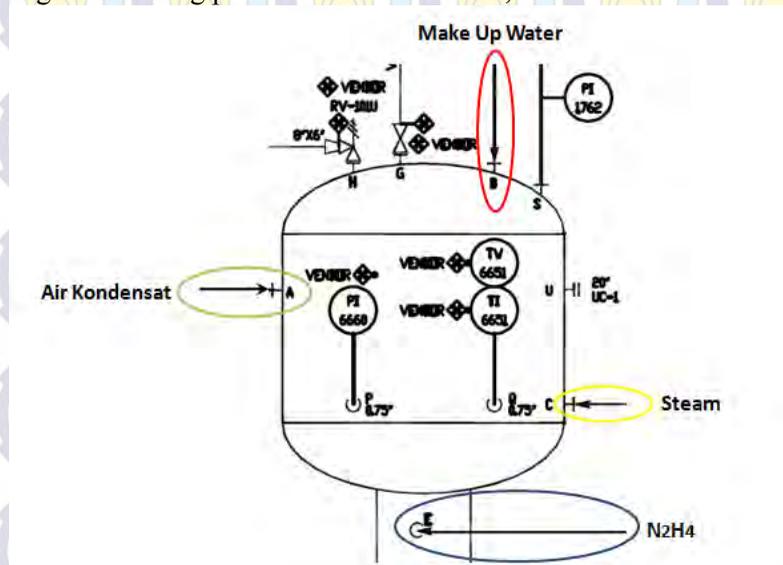
*Deaerator* adalah alat yang berfungsi untuk menghilangkan kandungan oksigen atau gas-gas terlarut lainnya pada air umpan sebelum masuk ke *boiler* untuk menghindari korosi[1]. *Deaerator* terdiri dari 2 bagian utama yaitu *degassing section*, pada bagian atas *deaerator* dan *storage water tank*, pada bagian bawah *deaerator*. Fungsi dari *degassing section* adalah menerima masukan serta menghilangkan  $O_2$  dan gas-gas terlarut lainnya yang nantinya akan dibuang ke atmosfer. Kemudian fungsi dari *storage water tank* adalah untuk menerima air dari proses yang ada dalam *degassing section*[5].



Gambar 2.1 *Deaerator*

Di PT. Petrokimia Gresik, menghilangkan kandungan  $O_2$  pada *deaerator* memiliki tujuan selain untuk menghindari korosi juga bertujuan untuk menghindari kebakaran. Dari P&ID *deaerator* 101U PT. Petrokimia Gresik, *degassing section*

*deaerator* 101U mendapatkan masukan dari *make up water* yang ditunjukkan dengan lingkaran merah pada Gambar 2.2. Terjadi proses pengikatan oksigen dengan memberi masukan *steam* dengan kandungan  $O_2$  kecil yang nantinya akan mengikat kandungan oksigen besar yang terkandung dalam *make up water*. *Steam* akan dibuang seluruhnya ke atmosfer beserta kandungan oksigen yang telah diikat. *Input steam* ditunjukkan dengan lingkaran kuning pada Gambar 2.2 berikut,



**Gambar 2.2** Fungsi *Deaerator* 101U

*Deaerator* juga mempunyai masukan berupa air kondensat seperti yang ditunjukkan oleh lingkaran hijau pada Gambar 2.2. kandungan air kondensat adalah  $H_2O$  tanpa ada campuran dari  $O_2$ . Air keluaran dari *degassing section deaerator* 101U nantinya akan masuk pada *storage water tank deaerator* 101U. Sebelum masuk ke dalam *storage water tank*, *make up water* dengan kandungan oksigen kecil direaksikan dengan cairan hidrazin ( $N_2H_4$ ) seperti pada Gambar 2.2 dengan lingkaran biru. Reaksi tersebut bertujuan untuk menghilangkan kandungan  $O_2$  secara

sempurna sehingga didapatkan cairan murni  $H_2O$  tanpa kandungan  $O_2$ . Persamaan reaksinya adalah sebagai berikut :



Hasil reaksi berupa  $H_2O$  yang murni akan masuk ke dalam *storage water tank* sedangkan  $N_2$  akan dibuang ke atmosfer. Keluaran *storage water tank deaerator* 101U nantinya akan dipompa yang kemudian masuk ke dalam *heat exchanger* dan *boiler*. Pengendalian level pada *deaerator* 101U dilakukan untuk menjaga ketinggian air di *storage water tank*. Level pada *deaerator* ini juga digunakan sebagai dasar untuk menginjeksikan cairan hidrazin. Cairan hidrazin hanya akan diinjeksikan pada saat level pada *deaerator* lebih tinggi dari *setpoint* dan kemudian apabila level *deaerator* sesuai *setpoint*, injeksi cairan hidrazin akan dihentikan.

### 2.1.1 Pemodelan Matematis *Deaerator*

Model matematis *deaerator* mengacu pada proses *deaerator*. *Deaerator* 101U mempunyai tiga masukan yaitu *makeup water*, *steam* dan air kondensat serta satu keluaran yaitu  $H_2O$ . Cairan hidrazin tidak dimasukkan dalam model matematis dikarenakan cairan hidrazin tidak disuntikan secara terus menerus. Model matematis dilakukan dengan menggunakan hukum kesetimbangan massa. Dari proses *deaerator*, didapatkan model matematis yaitu,

Dimana :

<i>MakeUp Water</i>	= MW	
<i>Steam</i>	= ST	
Air Kondensat	= AK	
<i>Steam Out</i>	= vapour	= v

$$\dot{m}_D = \dot{m}_{MW} + \dot{m}_{AK} + \dot{m}_{ST} - \dot{m}_v - \dot{m}_{out} \quad (2.1)$$

$$\rho \frac{dV_D}{dt} = \dot{m}_{MW} + \dot{m}_{AK} + \dot{m}_{ST} - \dot{m}_v - \dot{m}_{out} \quad (2.2)$$

$$\rho A \frac{dh}{dt} = \dot{m}_{MW} + \dot{m}_{AK} + \dot{m}_{ST} - \dot{m}_v - \dot{m}_{out} \quad (2.3)$$

Dimana :

$\rho$  = Massa Jenis Fluida  $\text{kg/m}^3$        $A$  = Luasan *Deaerator*

$\frac{dV_D}{dt}$  = Perubahan Volume Dalam *Deaerator*

Pendekatan model tangki adalah tabung dengan bentuk tangki horizontal seperti gambar 2.1. Perhitungan luasan parsial tabungan dapat dimodelkan dengan persamaan matematis sebagai berikut,

$$A = 2 \left( \pi h \frac{D_i}{2} \right) + 2 \sqrt{R^2 - (R - h)^2} \quad (2.4)$$

Dimana :

$h$  = Ketinggian *Deaerator* (m)

$D_i$  = Diameter *Deaerator* (m)

$R$  = Jari – Jari Tank *Deaerator* (m)

$$\dot{m}_{out} + \rho \left[ 2 \left( \pi h \frac{D_i}{2} \right) + 2 \sqrt{R^2 - (R - h)^2} \right] \frac{dh}{dt} = \dot{m}_{MW}(t) + \dot{m}_{AK}(t) + \dot{m}_{ST}(t) - \dot{m}_v(t) \quad (2.5)$$

Nilai *mass flow output* dapat diperoleh melalui analogi menggunakan resistan yaitu  $Q = h / R$  [11].  $R$  adalah kontrol *valve* yang mempengaruhi laju aliran dari fluida.

Dimana :  $Q = \dot{m}$        $R$  = Resistan

$h$  = Ketinggian Air *Deaerator*

*Mass flow vapour* ( $\dot{m}_v$ ) merupakan selisih dari perubahan fraksi *steam*. Sehingga, dapat diasumsikan bahwa *mass flow vapour* adalah fraksi *steam* dikurangi dengan fraksi *steam* yang berubah menjadi *liquid*. Dengan kata lain *mass flow vapour* digunakan untuk menganalisa perubahan *steam* yang menjadi *liquid*. Dari beberapa asumsi diatas, maka persamaa 2.5 menjadi,

$$\frac{h}{R_e} + \rho \left[ 2 \left( \pi h \frac{D_i}{2} \right) + 2 \sqrt{R^2 - (R - h)^2} \right] \frac{dh}{dt} =$$

$$\dot{m}_{MW}(t) + \dot{m}_{AK}(t) + \dot{m}_{ST}(t) - (1-x)\dot{m}_v(t) \quad (2.6)$$

(1-x) merupakan fraksi vapour yang mempengaruhi perubahan fasa dari *steam*.

Persamaan 2.6 dilaplacekan menjadi,

$$\frac{1}{R_e} H(s) + \rho \left[ 2 \left( \pi h \frac{D_i}{2} \right) + 2\sqrt{R^2 - (R-h)^2} \right] H(s) = M_{MW}(s) + M_{AK}(s) + M_{ST}(s) - (1-x)M_v(s) \quad (2.7)$$

$$\begin{aligned} H(s) &= \frac{M_{MW}(s)}{\frac{1}{R_e} \rho \left[ 2 \left( \pi h \frac{D_i}{2} \right) + 2\sqrt{R^2 - (R-h)^2} \right] s + 1} + \\ &\frac{M_{AK}(s)}{\frac{1}{R_e} \rho \left[ 2 \left( \pi h \frac{D_i}{2} \right) + 2\sqrt{R^2 - (R-h)^2} \right] s + 1} + \\ &\frac{M_{ST}(s)}{\frac{1}{R_e} \rho \left[ 2 \left( \pi h \frac{D_i}{2} \right) + 2\sqrt{R^2 - (R-h)^2} \right] s + 1} - \\ &\frac{(1-x)M_v(s)}{\frac{1}{R_e} \rho \left[ 2 \left( \pi h \frac{D_i}{2} \right) + 2\sqrt{R^2 - (R-h)^2} \right] s + 1} \end{aligned} \quad (2.8)$$

## 2.2 Internal Model Control (IMC)

*Internal model control* (IMC) adalah pengendalian yang hanya dapat dicapai jika sistem memuat parameter dari proses yang akan dikendalikan, baik secara implisit maupun eksplisit[6]. Dapat dikatakan juga bahwa IMC merupakan suatu metode yang dirancang untuk mengendalikan umpan balik dari keluaran suatu proses. Skema pengendali IMC menggunakan fungsi transfer dari *plant* untuk mencari fungsi transfer dari *controller*. Jika skema pengendali yang digunakan berdasarkan pada model dari *plant*, maka memungkinkan untuk mendapatkan pengendalian yang sempurna. IMC merupakan pengembangan dari algoritma kontrol yang memanfaatkan *inverse* dari model proses dengan

mempunyai satu parameter *tuning* yaitu *filter factor* ( $\lambda$ ) (Morari, 1986). Pada pengendalian *internal model control* ini dapat digunakan untuk mengendalikan nilai *disturbance* yang langsung masuk kedalam proses dan metode *tuning* ini bisa digunakan untuk mengendalikan nilai *set point* pada *plant* yang akan dikendalikan.

### 2.2.1 Model Based IMC

Kontrol IMC merupakan *controller* yang menggunakan model dari proses sebagai acuan dari *controller* IMC itu sendiri[7]. Gambar 2.3 merupakan contoh kontrol IMC pada proses *open loop*.



**Gambar 2.3** Proses *Open Loop* Kontrol IMC

$q(s)$  merupakan *controller* dari proses dan  $g_p$  merupakan fungsi transfer dari model proses. Dari Gambar 2.3 didapatkan persamaan,

$$y(s) = g_p(s)q(s)r(s) \quad (2.13)$$

Untuk *first order process*  $g_p(s) = \frac{k_p}{\tau s + 1}$ , maka untuk persamaan 2.13 menjadi,

$$y(s) = \frac{k_p k_q}{\tau s + 1} r(s) \quad (2.14)$$

Persamaan 2.14 adalah persamaan apabila proses dalam keadaan *steady state* atau statis. Pada proses yang dinamik, *controller* IMC yang digunakan adalah,

$$q(s) = \frac{1}{g_p(s)}$$

Untuk *first order process*, maka persamaan dari *controller* menjadi :

$$q(s) = \frac{\tau s + 1}{k_p} \quad (2.15)$$

Pada kontrol IMC, *controller* yang digunakan harus terdapat *denominator polynomial*[7]. *Denominator polynomial* pada IMC adalah berupa filter dengan persamaan,

$$f(s) = \frac{1}{\lambda s + 1} \quad (2.16)$$

Filter ini berguna untuk tuning parameter pada kontrol IMC. Dimana  $\lambda$  adalah parameter tuning filter untuk kontrol IMC. Nilai  $\lambda$  berfungsi untuk mencari respon kontrol IMC yang paling cepat. Untuk mencari respon yang paling cepat, nilai  $\lambda$  harus diperkecil. atau dengan kata lain, mencari nilai  $\lambda$  yang paling kecil untuk mendapatkan respon kontrol IMC yang paling cepat. Dari penambahan filter, didapatkan persamaan *controller*,

$$q(s) = \frac{1}{g_p(s)} = \frac{f(s)}{g_p(s)}$$

$$q(s) = \frac{\tau s + 1}{k_p} \cdot \frac{1}{\lambda s + 1} = \frac{\tau s + 1}{k_p(\lambda s + 1)} \quad (2.17)$$

Dimana :

$q(s)$  = *Controller IMC*

$\tau$  = *Time Constant Plant*

$k_p$  = *Gain Plant*

Dengan  $\lambda$  yang optimal maka akan dihasilkan performa terbaik dari respon proses. Menurut Rivera dalam Seborg (2004 : 307) besar nilai  $\lambda$  yang optimal dapat diperoleh pada persamaan,

$$\lambda > 0,8\theta \quad (2.18)$$

$$\lambda > 0,1\tau \quad (2.19)$$

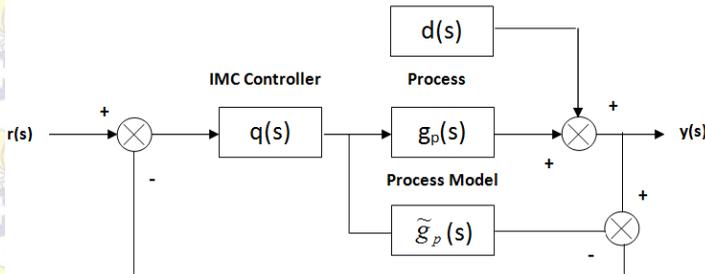
Di dalam sumber lain yang ditulis oleh Wyne Bequette[7], untuk mendapatkan performa yang terbaik maka nilai lamda filter harus sesuai dengan persamaan dibawah 2.20 ini,

$$\lambda > 1,5\theta \quad (2.20)$$

Sedangkan pada jurnal *simple analytic rules for model reduction and PID controller tuning* menurut Sigurd Skogestad[13], besar nilai  $\lambda$  adalah seperti pada persamaan 2.21 dibawah ini,

$$\lambda = 1 \quad (2.21)$$

Struktur pengendalian IMC dapat dirumuskan dalam struktur kontrol umpan balik standar[7]. Struktur IMC menggunakan dua fungsi transfer yaitu fungsi transfer dari proses dan fungsi transfer dari model. Tujuan dari penggunaan dua fungsi transfer ini adalah untuk menghilangkan *disturbance* yang terdapat pada proses. Pada Gambar 2.4 adalah diagram blok struktur IMC,



**Gambar 2.4** Diagram Blok Struktur Kontrol IMC

Dimana :

$\tilde{d}(s)$  = Sinyal Kontrol *Disturbance*       $d(s)$  = *Disturbance*

$g_p(s)$  = Fungsi Transfer Proses       $y(s)$  = *Process Output*

$\tilde{g}_p(s)$  = Fungsi Transfer Proses       $r(s)$  = *Set Point*

$q(s)$  = *Internal Model Controller*

Fungsi transfer model yang digunakan adalah sama dengan fungsi transfer *plant*. Dari gambar struktur IMC dapat dilihat bahwa fungsi transfer model yang digunakan berfungsi untuk merubah sinyal *feedback* pada *controller* menjadi sinyal *disturbance* yang sebelumnya merupakan sinyal proses yang ditambah dengan sinyal *disturbance*. Oleh sebab itu, IMC sangat baik digunakan untuk menghilangkan *disturbance* yang terdapat pada proses.

### 2.3 Pendekatan Model Dengan FOPDT

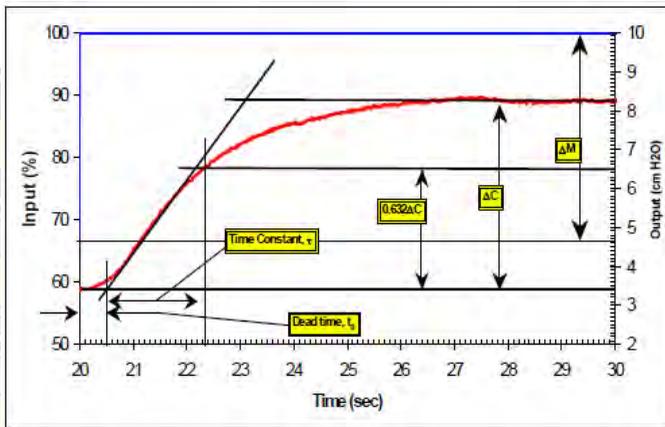
FOPDT (*First Order Plus Dead Time*) adalah sebuah metode untuk mendapatkan fungsi transfer proses dan parameter tuning[7]. Persamaan fungsi transfer dari proses apabila terdapat *dead time* adalah,

$$g_p(s) = \frac{K_p e^{-\theta s}}{\tau_p s + 1} \quad (2.22)$$

Dimana  $\theta$  adalah *time delay* yang terdapat pada respon proses. Pendekatan FOPDT dapat digunakan dengan *software* Ms.Excel dengan menganalisa respon sistem yang telah didapatkan[9]. Langkah – langkah metode FOPDT adalah :

1. Menentukan perubahan *steady state output* ( $\Delta C$ ) dan perubahan *steady state input* ( $\Delta M$ ).
2. Mengitung *gain plant* ( $k$ ) menggunakan parameter  $\Delta C$  dan  $\Delta M$  dengan membagi *steady state output* dengan *steady state input*.
3. Kemudian *time constant* ( $\tau$ ) dari respon sistem merupakan 63% dari respon transien yang terdapat pada sistem setelah *time delay*.

Parameter pada langkah – langkah metode FOPDT dapat digambarkan seperti pada Gambar 2.5,



**Gambar 2.5** Analisa Respon Dengan Metode FOPDT

## 2.4 Respon Transient

Respon *Transient* suatu sistem kontrol secara praktek selalu menunjukkan osilasi teredam sebelum mencapai keadaan tunaknya. Dalam menggolongkan karakteristik tanggapan transient suatu sistem kontrol terhadap masukan tangga satuan yaitu waktu tunda/*time delay* ( $t_d$ ), waktu naik/*rise time* ( $t_r$ ), waktu puncak/*peak time* ( $t_p$ ), maksimum *overshoot* ( $M_p$ ), *settling time* ( $t_s$ ) dimana [8],

1. Waktu tunda/*time delay* ( $t_d$ ) adalah waktu yang digunakan oleh tanggapan untuk mencapai setengah nilai akhir untuk waktu yang pertama.
2. Waktu naik/*rise time* ( $t_r$ ) adalah waktu yang diperlukan oleh tanggapan untuk naik dari 10% menjadi 90%, 5% menjadi 95%, atau 0% menjadi 100% dari nilai akhir yang biasa digunakan. Untuk sistem atas redaman waktu naik yang biasa digunakan 10% menjadi 90%.
3. Waktu puncak/*peak time* ( $t_p$ ) adalah waktu yang diperlukan tanggapan untuk mencapai puncak pertama *overshoot*.
4. Maksimum *overshoot* ( $M_p$ ) adalah nilai puncak kurva tanggapan diukur dari satuan. Apabila nilai akhir keadaan tunak tanggapannya jauh dari satu, maka biasa digunakan

persen maksimum *overshoot* dan didefinisikan sebagai berikut,

$$\%Mp = \frac{c(tp) - c(\infty)}{c(\infty)} \times 100\%$$

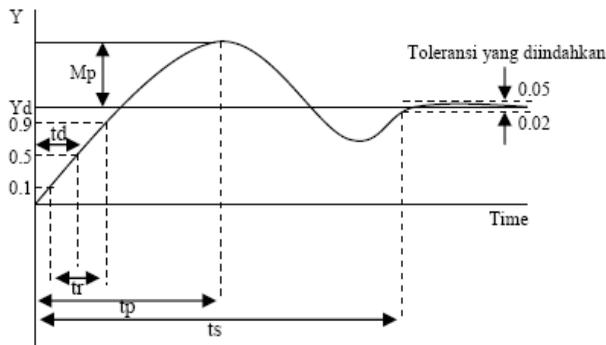
Dimana :

$c(tp)$  : Keadaan pada waktu puncak

$c(\infty)$  : Keadaan pada *setpoint*

Besarnya persen maksimum *overshoot* menunjukkan kestabilan relative dari system.

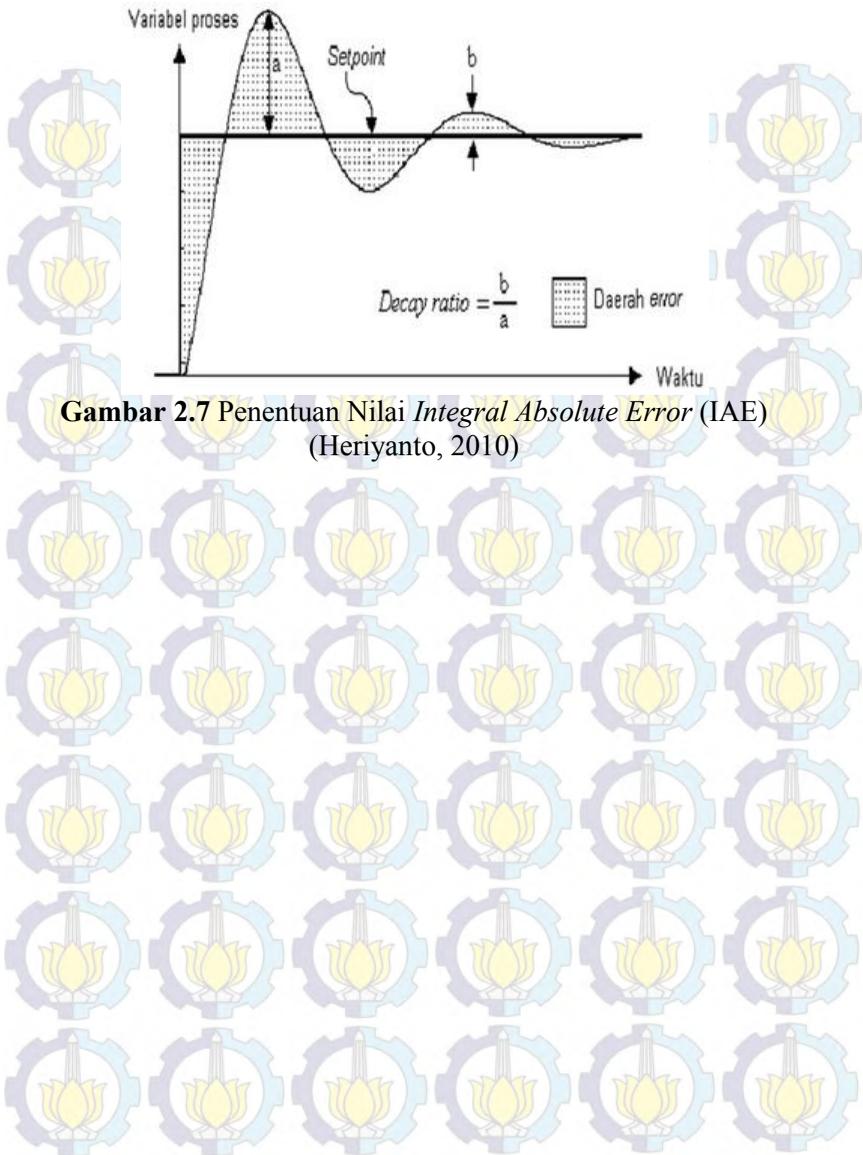
5. *Settling time* ( $t_s$ ) adalah waktu yang diperlukan untuk menanggapi kurva agar dapat mencapai dan tetap berada dalam gugus akhir ukuran yang disederhanakan dengan presentase mutlak harga akhirnya (biasanya 2% atau 5%).



**Gambar 2.6** Grafik Respon *Transient*

6. *Integral absolute error* (IAE) merupakan penjumlahan nilai *error* dari respon *output* yang dihasilkan. Nilai IAE dapat ditentukan berdasarkan luas arsir kurva dari respon sistem yang dihasilkan. Sehingga dapat dikatakan bahwa penjumlahan nilai IAE diawali terhadap respon sistem ketika diberikan sebuah *controller*. Untuk menghitung nilai IAE dapat menggunakan persamaan dibawah ini,

$$IAE = \int_0^{\infty} e(t)dt \quad (2.23)$$

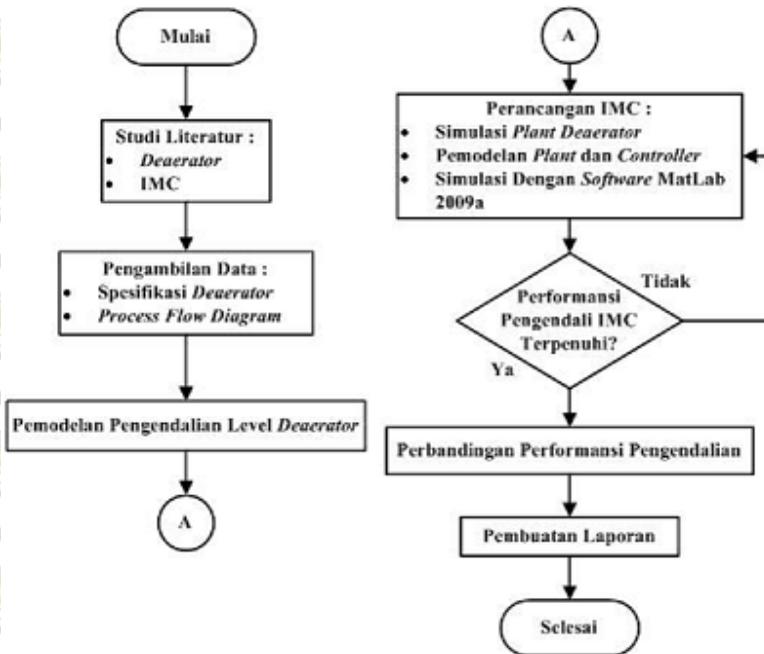


**Gambar 2.7** Penentuan Nilai *Integral Absolute Error* (IAE)  
(Heriyanto, 2010)

## BAB III METODOLOGI PERANCANGAN SISTEM

### 3.1 Alur Penelitian

Pada bab ini akan dijelaskan tentang tahapan pengerjaan tugas akhir mulai dari studi literatur, pemodelan sistem, perancangan sistem kendali *Internal Model Control*, simulasi dan uji performansi, serta penyusunan laporan tugas akhir. Beberapa tahapan tersebut dapat dijelaskan pada Gambar 3.1 seperti di bawah ini,



**Gambar 3.1** Flowchart Penelitian Tugas Akhir

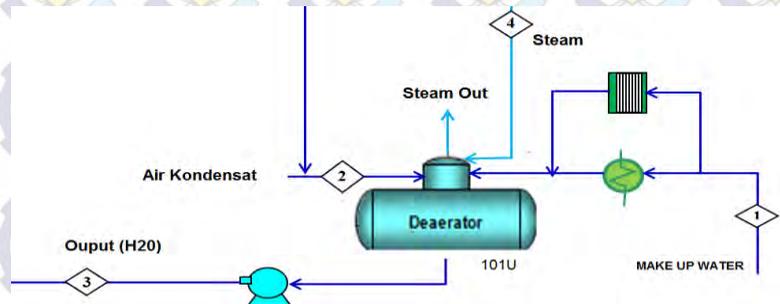
### 3.2 Studi Literatur

Studi literatur dilakukan dengan mencari referensi untuk *dearator* dalam *plant*, proses dalam *dearator*, spesifikasi

*deaerator*, dan kendali IMC. Studi literatur dilakukan dengan mencari jurnal-jurnal dan buku-buku yang berhubungan dengan *deaerator* dan kendali IMC sebagai bahan pengerjaan tugas akhir.

### 3.3 Pengambilan Data

Pengambilan data *deaerator* meliputi PVT pada setiap masukan pada *deaerator*. Data yang diperoleh nantinya akan digunakan sebagai acuan dalam melakukan simulasi dengan menggunakan *software* Hysys. Secara garis besar, *deaerator* 101U berfungsi untuk menghilangkan kandungan oksigen atau gas-gas terlarut lainnya pada air umpan yang nantinya akan masuk ke *boiler* dan *heat exchanger* untuk menghindari korosi.



**Gambar 3.2** Proses Input dan Output Deaerator

Data yang digunakan dalam simulasi untuk setiap *stream* pada ditunjukkan pada Tabel 3.1 berikut,

**Tabel 3.1** Data PVT Deaerator Uji Open Loop

Stream	Keterangan	Mass Flow (kg/h)	Temperature (°C)	Pressure (kg/cm <sup>2</sup> g)
1	Make Up Water	195329	35	5.3
2	Condensate Water	1031	147.5	3.5
3	Output	199794	131	126.5
4	Steam	122244	131	126.5

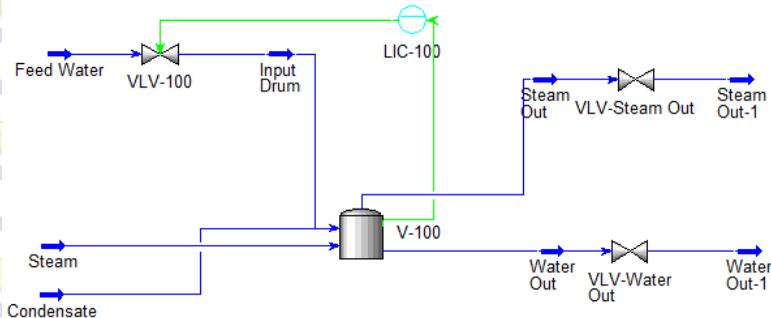
Data spesifikasi dimensi dari *deaerator* 101U diperoleh ditunjukkan pada Tabel 3.2 berikut,

**Tabel 3.2** Data Spesifikasi Dimensi *Deaerator*.

No	Data Operasional	Satuan (m)
1	Diameter	3.048
2	Panjang tabung	9.7536
3	Ketinggian level	1.8

### 3.4 Pemodelan Sistem *Open Loop*

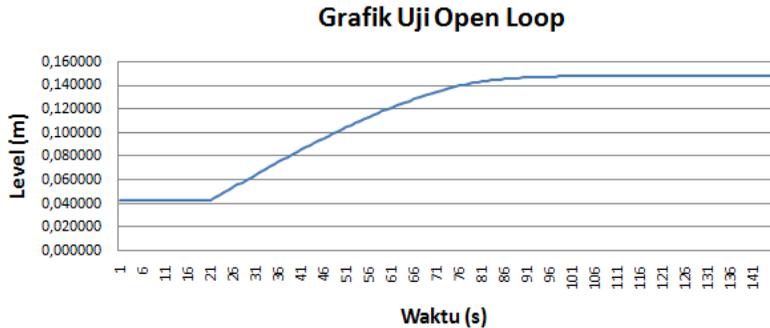
Dari penurunan model matematika pada sub.bab 2.1.1, proses disimulasikan ke dalam *software* Hysys seperti pada Gambar 3.3. Komposisi pada simulasi mengamsumsikan bahwa kandungan semua fluida adalah H<sub>2</sub>O kecuali kandungan pada *make up water* yang mempunyai komposisi H<sub>2</sub>O dan oksigen sebesar 30 ppm.



**Gambar 3.3** Layout Simulasi Hysys

Simulasi pada *software* Hysys seperti pada Gambar 3.3 merupakan uji *open loop* dari proses *deaerator*. Uji *open loop* berfungsi untuk mencari fungsi transfer dari proses. Data yang diambil adalah data dimana respon sistem menunjukkan respon konstan. Data yang diambil sebanyak 145 data. Data tersebut meliputi *mass flow input*, data *operation* (OP) dari *valve*, ketinggian air (level) dan waktu respon. Grafik respon yang digunakan untuk pemodelan dengan metode FOPDT adalah grafik

antara level dan waktu respon. Grafik respon yang didapatkan menggunakan Ms. Excel dengan data *record* dari Hysys.



**Gambar 3.4** Grafik Uji *Open Loop* Hysys

Dari grafik uji *open loop* diperoleh parameter FOPDT untuk mencari fungsi transfer proses seperti pada Tabel 3.3 berikut,

**Tabel 3.3** Parameter FOPDT Uji *Open Loop* Hysys

<i>Settling Time</i>	78,780 s
Perubahan <i>Steady State</i> ( $\Delta$ )	0,105507 kg/h
<i>Gain Steady State</i> (k)	0,55530181
<i>Time Constant</i> ( $\tau$ )	28,614 s
<i>Time Delay</i> ( $\theta$ )	5,106 s
$t_{63\%}$	33,720 s
$t_{28\%}$	14,6444 s

Langkah – langkah untuk mencari parameter dengan metode FOPDT adalah :

1. *Settling time* (ts)

*Settling time* didapatkan dari perubahan respon level sampai menuju *steady*. Dari grafik uji *open loop*, perubahan respon terjadi pada saat waktu 21 s dan mengalami *steady state* pada waktu ke 99 s. Karena data diambil setiap detik, maka *settling time* adalah

waktu dari level pada data ke 21 sampai 99. Dari hasil *record* data ke-21 memiliki waktu respon sebesar 980,890 s dan data ke-99 sebesar 1107,140 s. Maka besar *settling time* 78,780 s.

#### 2. Perubahan *Steady State* ( $\Delta$ )

Perubahan *steady state* didapatkan dari selisih *mass flow* pada waktu ke 21 s dan ke 99 s, yaitu,  
 $0,148044 \text{ kg/h} - 0,042536 \text{ kg/h} = 0,105507 \text{ kg/h}$

#### 3. *Gain Steady State* (k)

*Gain steady state* didapatkan dari perubahan *steady state* dibagi dengan data *operation valve*. *Operation valve* adalah selisih dari dimana bukaan *valve* dengan *span* dari 1% sampai 20%. Maka besar *operation valve* yang diperoleh adalah 19%. Sehingga besar *gain steady state* (k) adalah,

$$k = \frac{0,105507}{0,19} = 0,55530181$$

#### 4. Nilai $t_{63\%}$

Nilai  $t_{63\%}$  merupakan waktu data ke 63% dari perubahan respon level sampai menuju *steady state*. Pertama yang dilakukan untuk mencari nilai  $t_{63\%}$  adalah dengan mencari nilai 63% dari *settling time*, yaitu,

$$78,780 \times 0,63 = 0,066469626 \text{ s.}$$

Kemudian nilai 63% dari *settling time* ditambahkan dengan level pada data ke 21, yaitu,

$$0,066469626 \text{ m} + 0,042536 \text{ m} = 0,109006 \text{ m}$$

Sehingga, ketinggian air pada saat 63% dari perubahan respon level sampai menuju *steady state* sebesar 0,109006 m. Waktu untuk mencapai ketinggian air tersebut dapat dicari dengan cara interpolasi. Interpolasi adalah perbandingan waktu yang dibutuhkan pada saat level tertentu dengan acuan level yang berada diantaranya. Level 0,109006 m berada diantara level dengan ketinggian 0,108331 m dan 0,110077 m. Tabel perhitungan interpolasi ditunjukkan seperti yang pada Tabel 3.4 berikut,

**Tabel. 3.4** Interpolasi 63%

Interpolasi 63%	
Level (m)	Waktu (s)
0,108331	33,330
0,109006	X
0,110077	34,340
x	33,720

Waktu pada saat level 0,108331 m didapatkan dari waktu yang dibutuhkan untuk mencapai level tersebut dikurangi dengan waktu level pada saat data ke 21, yaitu,  $1014,220 \text{ s} - 980,890 \text{ s} = 33,330 \text{ s}$

Kemudian, waktu pada saat level 0,110077 m didapatkan dari waktu yang dibutuhkan untuk mencapai level tersebut dikurangi dengan waktu level pada saat data ke 21, yaitu,  $1015,230 \text{ s} - 980,890 \text{ s} = 34,340 \text{ s}$

Interpolasi 63% dicari dengan menggunakan rumus,

$$\frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1} \quad (3.1)$$

Dimana :

$y$  = Waktu yang belum diketahui ( $x$ )

$y_1 = 33,330 \text{ s}$

$y_2 = 34,340$

$x_2 = 0,110077 \text{ m}$

$x_1 = 0,108331 \text{ m}$

$x = 0,109006 \text{ m}$

Maka, nilai  $y$  dapat diketahui dengan,

$$y = \frac{(y_2 - y_1)(x - x_1)}{x_2 - x_1} + y_1 \quad (3.2)$$

$$y = \frac{(34,340\text{s} - 33,330\text{s})(0,109006\text{m} - 0,108331\text{m})}{0,110077\text{m} - 0,108331\text{m}} + 33,330\text{s}$$

$$y = 33,720\text{s}$$

Jadi, waktu pada saat level mencapai 63% dari perubahan respon level sampai menuju *steady state* sebesar 33,720 s.

5. Nilai  $t_{28\%}$

Nilai  $t_{28\%}$  merupakan waktu data ke 28% dari perubahan respon level sampai menuju *steady state*. Nilai waktu tersebut dapat dicari dengan cara interpolasi yang sama pada saat  $t_{63\%}$ . Hasil interpolasi seperti yang ditunjukkan pada Tabel 3.5 berikut,

**Tabel 3.5** Interpolasi 28%

Interpolasi 28%	
Level (m)	Waktu (s)
0,071014	14,140
0,072078	X
0,073145	15,150
X	14,644

Jadi, waktu pada saat level mencapai 28% dari perubahan respon level sampai menuju *steady state* sebesar 14,644 s.

6. *Time Constant* ( $\tau$ )

*Time Constant* dengan metode FOPDT dapat dicari dengan persamaan,

$$\tau = 1,5(t_{63\%} - t_{28\%})$$

$$\tau = 1,5(33,720 - 14,644) = 28,614 \text{ s}$$

7. *Time Delay* ( $\theta$ )

*Time delay* yang terdapat pada proses, didapatkan dengan persamaan,

$$\theta = t_{63\%} - \tau$$

$$\theta = 33,720 - 28,614 = 5,106 \text{ s}$$



$$Error = \frac{\text{Jumlah Selisih Level Tiap Data}}{\text{Jumlah Data}} = \frac{662,96}{82} = 8,08$$

Jadi, dari validasi respon didapatkan nilai *error* sebesar 8,08%. Perbandingan *time constant real plant* dan model, dilakukan dengan melihat perubahan respon dari level saat level *steady* pada saat *percentage open control valve* sebesar 0% dan 4%. Level *real plant steady* pada ketinggian 63,99% saat *percentage open control valve* sebesar 0% dan 72,35% saat *percentage open control valve* sebesar 4%. Sedangkan level dari model *steady* pada ketinggian 55,41% saat *percentage open control valve* sebesar 0% dan 65,74% saat *percentage open control valve* sebesar 4%. Nilai *time constant* dapat dicari menggunakan rumus *time constant* dari FOPDT dengan asumsi bahwa waktu dirubah ke dalam sekon atau 1 menit sama dengan 60 detik. Tabel 3.6 – 3.9 berikut menunjukkan perhitungan interpolasi untuk mencari nilai *time constant* dari *plant* dan model.

**Tabel. 3.6** Interpolasi 63% *Plant*

Interpolasi 63%	
Level (%)	Waktu (s)
68,74	600
69,26	X
69,34	660
X	652

**Tabel. 3.7** Interpolasi 28% *Plant*

Interpolasi 28%	
Level (%)	Waktu (s)
65,78	300
66,33	X
66,36	360
X	356,8966

Dari Tabel 3.6 dan Tabel 3.7 didapatkan nilai *time constant plant* adalah,

$$\tau = 1,5 (t_{63\%} - t_{28\%})$$

$$\tau = 1,5 (652 \text{ s} - 356,8966 \text{ s}) = 442,655 \text{ s}$$

**Tabel. 3.8** Interpolasi 63% Model

Interpolasi 63%	
Level (%)	Waktu (s)
60,19	540
60,67	X
60,85	600
X	583,6364

**Tabel. 3.9** Interpolasi 28% Model

Interpolasi 28%	
Level (%)	Waktu (s)
57,70	300
57,75	X
58,34	360
X	304,6875

Dari Tabel 3.7 dan 3.8 didapatkan nilai *time constant model* adalah,

$$\tau = 1,5 (t_{63\%} - t_{28\%})$$

$$\tau = 1,5 (583,6364 \text{ s} - 304,6875 \text{ s}) = 418,423 \text{ s}$$

Terdapat perbedaan nilai *time constant* model pada saat validasi dan uji *open loop*. Hal ini dikarenakan, pada saat validasi model, *input* yang digunakan hanya satu yaitu *makeup water*. Berbeda dengan pada saat uji *open loop* yang menggunakan tiga *input* yaitu *steam*, air kondensat dan *makeup water*. Penggunaan satu *input* dilakukan karena mengacu pada data *start up* yang hanya meninjau satu masukan saja yaitu *makeup water*.

### 3.6 Pemodelan Sistem Dengan Uji *Disturbance*

Pemodelan sistem dengan uji *disturbance* adalah melakukan perubahan pada *mass flow* pada *steam* dan air kondensat sebagai *disturbance*. Data PVT yang digunakan sama dengan Tabel 3.1 dengan perubahan nilai *mass flow* pada *steam* dan air kondensat. Komposisi air kondensat adalah murni H<sub>2</sub>O serta data spesifikasi *deaerator* pada Tabel 3.2. Uji *disturbance* berfungsi untuk mengetahui fungsi transfer *disturbance*. Pemodelan dilakukan dengan metode FOPDT. Analisa respon dilakukan pada grafik saat *mass flow* dari *steam* dirubah dan *mass flow* dari air kondensat dirubah.

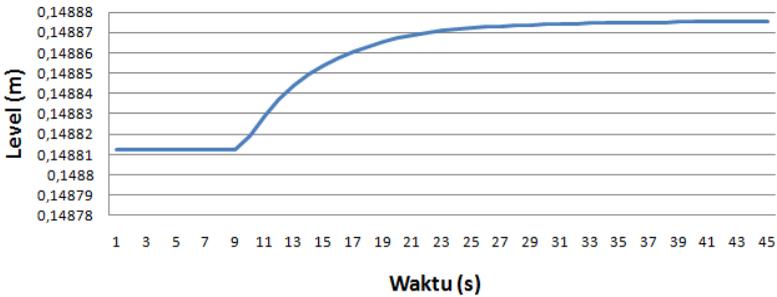
#### 3.6.1 Uji *Disturbance* Perubahan *Mass Flow* Air Kondensat

Data PVT yang digunakan dengan perubahan *mass flow* air kondensat adalah,

**Tabel 3.10** Data PVT *Deaerator* Uji *Disturbance* Air Kondensate

<i>Stream</i>	Keterangan	<i>Mass Flow</i> (kg/h)	<i>Temperature</i> (°C)	<i>Pressure</i> (kg/cm <sup>2</sup> g)
1	<i>Make Up Water</i>	195329	35	5.3
2	<i>Condensate Water</i>	1500	147.5	3.5
3	<i>Output</i>	199794	131	126.5
4	<i>Steam</i>	122244	131	126.5

Dari data diatas disimulasikan dengan *software* Hysys. Data yang dibutuhkan untuk plot grafik pada Ms. Excel adalah *mass flow*, ketinggian air dan waktu. Data *record* yang diambil dari *software* Hysys ke dalam Ms excel sebanyak 45 data dengan waktu per detik. Banyaknya data yang diambil berdasarkan perubahan level dan pada saat level menunjukkan respon yang konstan.



**Gambar 3.6** Grafik Uji *Disturbance* Perubahan *Mass Flow* Air Kondensat

Dari Gambar 3.6, dapat diketahui perubahan respon terjadi pada saat waktu 9 s dan respon menunjukkan nilai level yang *steady* pada waktu 34 s. Fungsi transfer dari air kondensat dicari dengan metode FOPDT dengan cara yang sama pada saat mencari fungsi transfer proses pada uji *open loop*. Dengan metode FOPDT didapatkan parameter model yang ditunjukkan pada Tabel 3.11 berikut,

**Tabel 3.11** Parameter Pemodelan FOPDT Uji *Disturbance* Air Kondensat

<i>Settling Time</i>	25 s
Perubahan Steady State ( $\Delta$ )	0,0000620247 kg/h
<i>Gain Steady State</i> (k)	0,00047610
<i>Time Constant</i> ( $\tau$ )	5,075251086 s
<i>Time Delay</i> ( $\theta$ )	0,444560015 s
$t_{63\%}$	5,519811102 s
$t_{28\%}$	2,136310378 s

Dari data pada Tabel 3.11, didapatkan fungsi transfer air kondensat adalah,

$$g_{d \text{ steam}} = \frac{k e^{-\theta s}}{\tau s + 1} \quad (3.4)$$

$$g_{d \text{ airkondensat}} = \frac{0,00047610 e^{-0,4445600k}}{5,075251086 s + 1}$$

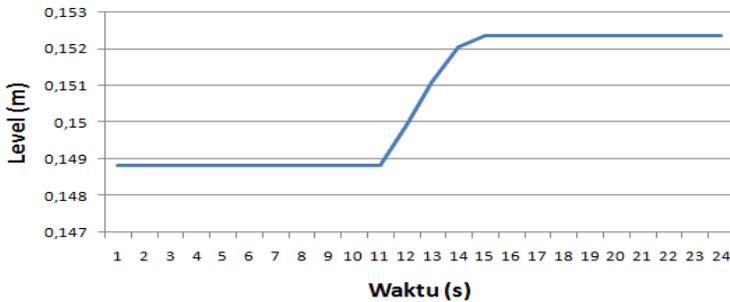
### 3.6.2 Uji *Disturbance Perubahan Mass Flow Steam*

Data PVT yang digunakan dengan perubahan *mass flow steam* ditunjukkan pada Tabel 3.12 berikut,

**Tabel 3.12** Data PVT *Deaerator Uji Disturbance Steam*

<i>Stream</i>	Keterangan	<i>Mass Flow</i> (kg/h)	<i>Temperature</i> (°C)	<i>Pressure</i> (kg/cm <sup>2</sup> g)
1	<i>Make Up Water</i>	195329	35	5.3
2	<i>Condensate Water</i>	1031	147.5	3.5
3	<i>Output</i>	199794	131	126.5
4	<i>Steam</i>	50000	131	126.5

Dari data pada Tabel 3.12 disimulasikan dengan *software* Hysys. Data yang dibutuhkan untuk plot grafik pada Ms. Excel adalah *mass flow*, ketinggian air dan waktu. Data *record* yang diambil dari *software* Hysys ke dalam *software* Ms. Excel sebanyak 24 data. Banyaknya data yang diambil berdasarkan perubahan level dan pada saat level menunjukkan respon yang konstan atau pada saat *steady state*.



**Gambar 3.7** Grafik Uji *Disturbance* Perubahan *Mass Flow Steam*

Dari Gambar 3.7, dapat diketahui perubahan respon terjadi pada waktu 11 s dan respon menunjukkan nilai level yang *steady* pada waktu 15 s. Fungsi transfer dari *steam* dicari dengan metode FOPDT dengan cara yang sama pada saat mencari fungsi transfer proses pada uji *open loop*. Dengan metode FOPDT didapatkan,

**Tabel 3.13** Parameter Pemodelan FOPDT Uji *Disturbance Steam*

<i>Settling Time</i>	4
Perubahan <i>Steady State</i> ( $\Delta$ )	0,003536 kg/h
<i>Gain Steady State</i> ( $k$ )	0,000176
<i>Time Constant</i> ( $\tau$ )	2,922988 s
<i>Time Delay</i> ( $\theta$ )	-0,97335 s
$t_{63\%}$	1,949641 s
$t_{28\%}$	0,000983 s

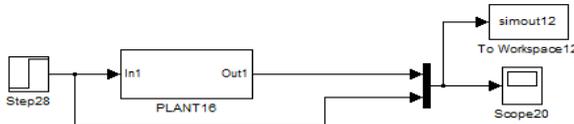
Pada nilai *time delay*, mendapatkan asumsi bahwa nilai *time delay* dianggap nol. Dari data pada Tabel 3.13, didapatkan fungsi transfer *steam* adalah,

$$g_{d \text{ steam}} = \frac{k e^{-\theta s}}{\tau s + 1} \quad (3.5)$$

$$g_{d \text{ steam}} = \frac{0,000176}{2,922988 s + 1}$$

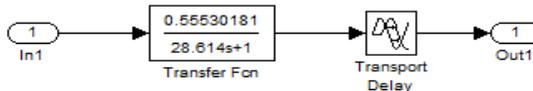
### 3.7 Uji *Open Loop Gain Plant*

Uji *open loop gain plant* dilakukan dengan menggunakan *software* MatLab 2009a. Uji *open loop* bertujuan untuk mengetahui respon dari plant sebelum menggunakan *controller* dan sebelum ada *disturbance*. Penyusunan pada *simulink* dilakukan seperti pada Gambar 3.8 berikut,



**Gambar 3.8** Uji *Open Loop Gain Plant*

Pada *subsystem*, berisi nilai dari fungsi transfer *plant* dan *time delay* menggunakan parameter pemodelan FOPDT pada data Tabel 3.3 seperti pada Gambar 3.9 berikut,



**Gambar 3.9** *Subsystem Plant*

## 3.8 Perancangan IMC *Controller*

### 3.8.1 Uji *Controller* tanpa *Disturbance*

Uji *controller* ini bertujuan untuk melihat respon dari plant pada saat sudah menggunakan *controller*. Kontrol yang digunakan menggunakan kontrol IMC. Perancangan kontrol IMC dilakukan dengan merancang fungsi transfer *controller* dari fungsi transfer *plant* yang telah didapatkan. Persamaan *controller* IMC adalah,

$$q(s) = \tilde{g}_p^{-1}(s)f(s) = \tilde{g}_p^{-1}(s) \frac{1}{\lambda s + 1} \quad (3.6)$$

Dari fungsi transfer *plant* didapatkan nilai *controller* IMC sebesar,

$$\tilde{g}_p^{-1} = \frac{28,614s + 1}{0,55530181} e^{-5,106s}$$

Sehingga persamaan *controller* IMC menjadi,

$$q(s) = \frac{28,614s + 1}{0,55530181} e^{-5.106s} \frac{1}{\lambda s + 1} \quad (3.7)$$

Parameter tuning IMC yaitu  $\lambda$  berfungsi untuk mendapatkan respon yang terbaik. Respon terbaik tersebut yang nantinya akan dibandingkan dengan kontrol PID. Nilai  $\lambda$  yang digunakan sebanyak 4 variasi berdasarkan persamaan 2.11, 2.18, 2.19, dan 2.21. Maka fungsi transfer *control* IMC adalah,

$$\lambda = 1$$

$$q(s) = \frac{28,61s + 1}{0,5553s + 0,5553} e^{-5.106s}$$

$$\lambda = 3$$

$$q(s) = \frac{28,61s + 1}{1,666s + 0,5553} e^{-5.106s}$$

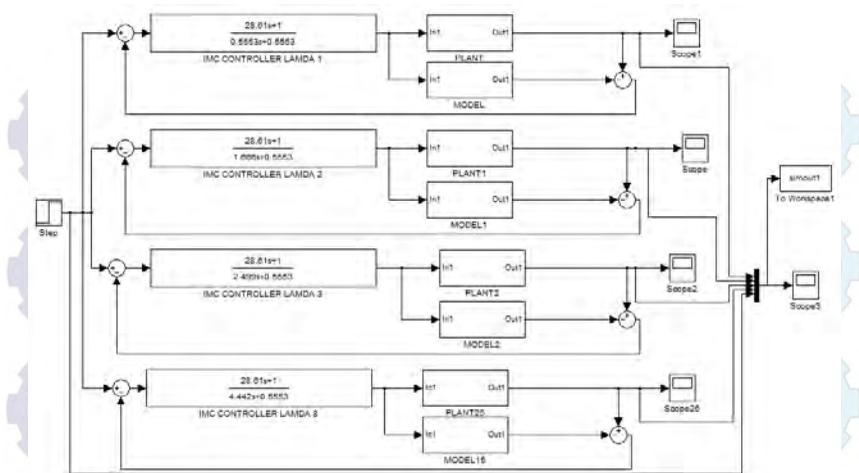
$$\lambda = 4.5$$

$$q(s) = \frac{28,61s + 1}{2,499s + 0,5553} e^{-5.106s}$$

$$\lambda = 8$$

$$q(s) = \frac{28,61s + 1}{4,442s + 0,5553} e^{-5.106s}$$

Simulasi yang dilakukan menggunakan *software* MatLab 2009a untuk mendapatkan grafik respon. Susunan pada *simulink* didasarkan dengan diagram kontrol IMC. Penyusunan pengendali IMC tanpa *disturbance* pada MatLab ditunjukkan seperti pada Gambar 3.10 berikut,

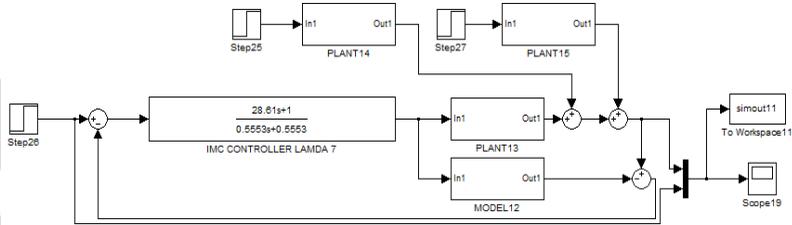


**Gambar 3.10** IMC Controller Tanpa Disturbance

Perbandingan respon sistem dilakukan dengan membandingkan respon dari kontrol IMC. Parameter respon yang digunakan untuk melihat performansi *controller* adalah *maximum overshoot*, *settling time* dan *integral absolute error (IAE)*. Respon terbaik nantinya akan dibandingkan dengan *controller* dengan metode yang lain.

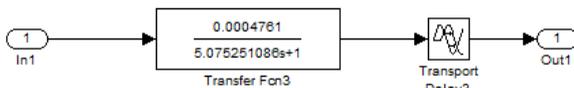
### 3.8.2 Uji Controller Dengan Disturbance

Uji *controller* dengan *disturbance* ini bertujuan untuk melihat respon dari *controller* apabila mendapat *disturbance* dari proses. *Disturbance* yang dimasukan adalah air kondensat dan *steam*. Penyusunan pada *software* MatLab dilakukan seperti pada Gambar 3.11 berikut,

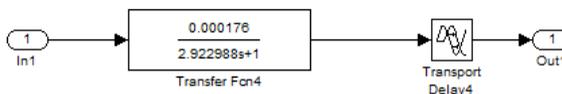


**Gambar 3.11** Uji *Controller* Dengan *Disturbance*

Pada *subsystem disturbance*, berisi nilai dari fungsi transfer *disturbance* dan *time delay* dari *steam* dan air kondensat menggunakan parameter pemodelan FOPDT pada data Tabel 3.7 dan Tabel 3.9 seperti pada Gambar 3.12 dan 3.13 berikut,



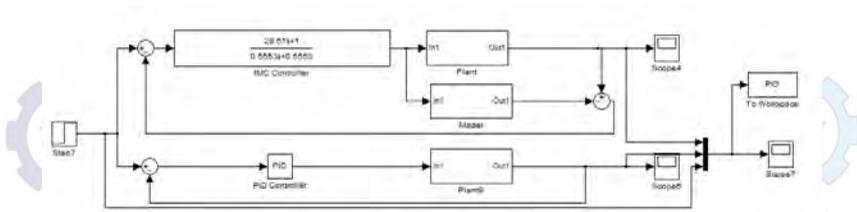
**Gambar 3.12** *Subsystem Disturbance* Air Kondensat



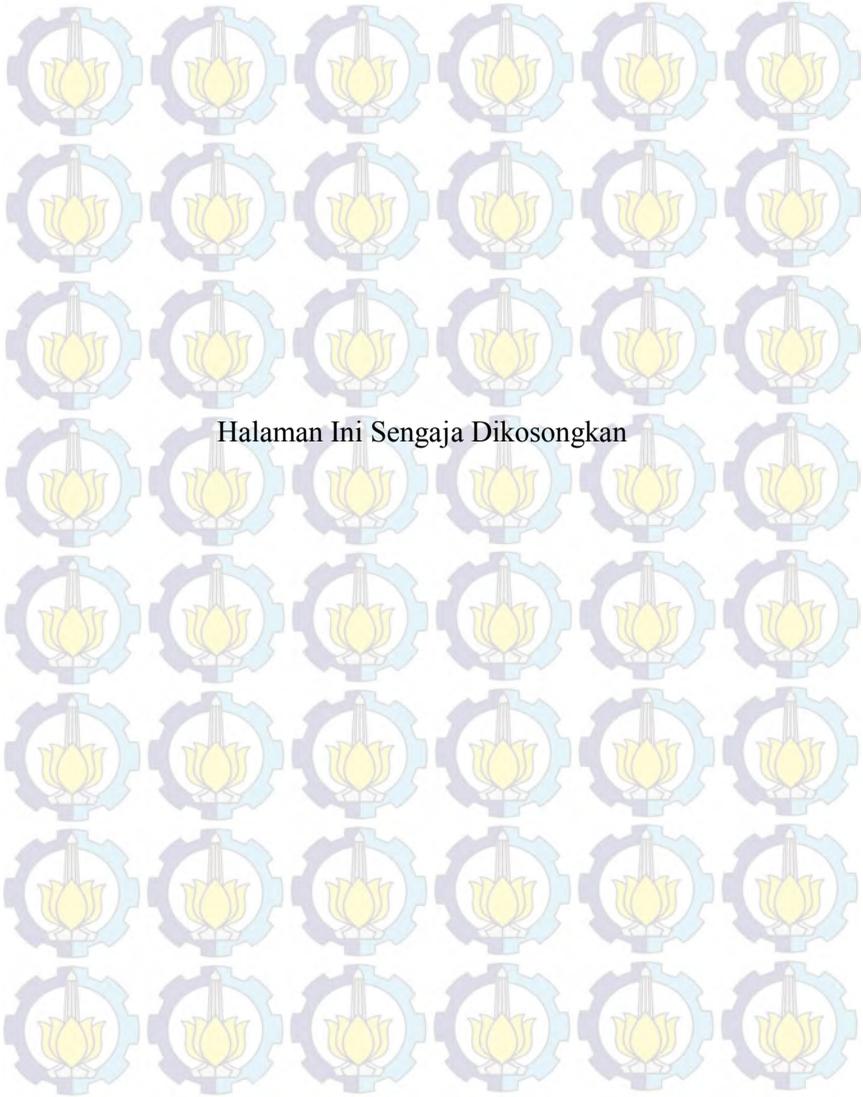
**Gambar 3.13** *Subsystem Disturbance* *Steam*

### 3.9 Perbandingan Respon Sistem

Perbandingan respon sistem dilakukan dengan membandingkan respon dari *controller* IMC dengan kontrol PID yang sudah menggunakan tanpa *disturbance*. Parameter tuning *controller* PID yang digunakan memiliki nilai  $K_c = 1,2$  dan nilai  $T_i = 0,06$ . Untuk mendapatkan respon sistem, simulasi dilakukan dengan menggunakan *software* MatLab 2009a. *Controller* dari IMC yang digunakan adalah *controller* dengan nilai  $\lambda$  terkecil. Rancangan kontrol pada *simulink* ditunjukkan pada Gambar 3.14 berikut,



**Gambar 3.14** Simulasi IMC Controller Dan PID Controller

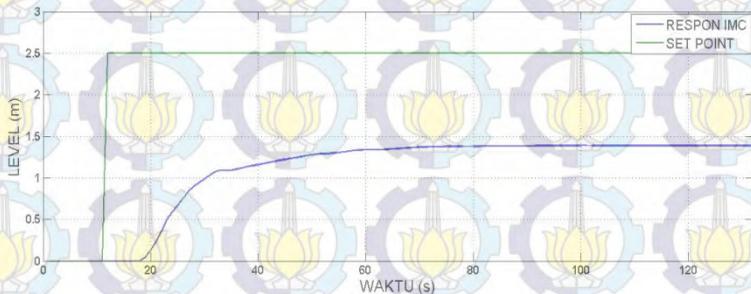


Halaman Ini Sengaja Dikосongkan

## BAB IV ANALISA DATA DAN PEMBAHASAN

### 4.1. Hasil Simulasi Uji *Open Loop Gain Plant*

Uji *open loop* bertujuan untuk mengetahui respon dari *plant* sebelum menggunakan *controller* dan sebelum ada *disturbance*. Simulasi dilakukan dengan menggunakan *software* MatLab. Hasil respon kontrol IMC ditunjukkan pada Gambar 4.1 berikut,



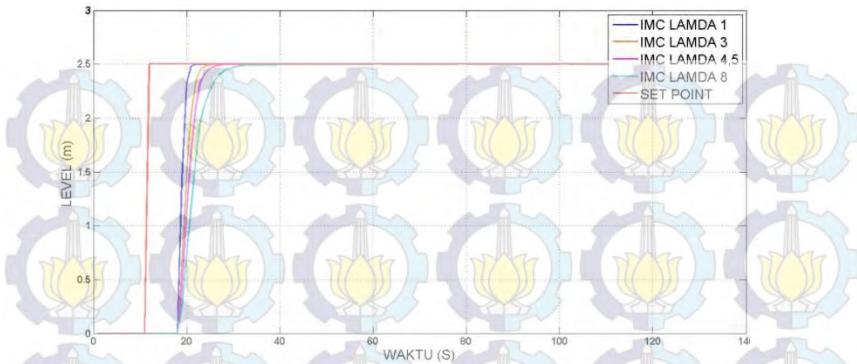
**Gambar 4.1** Grafik Uji *Open Loop Gain Plant*

Dari hasil grafik uji *open loop gain plant*, diketahui bahwa level respon sistem tidak dapat mencapai *set point* yaitu 2,5 m.

### 4.2 Hasil Perancangan *Controller* IMC

#### 4.2.1 Hasil *Controller* IMC Tanpa *Disturbance*

Simulasi kontrol IMC bertujuan untuk mendapatkan respon sistem dengan parameter tuning *controller* terbaik. Simulasi dilakukan dengan menggunakan *software* MatLab. Hasil respon kontrol IMC tanpa *disturbance* ditunjukkan pada Gambar 4.2 berikut,



**Gambar 4.2** Respon IMC Controller Tanpa Disturbance

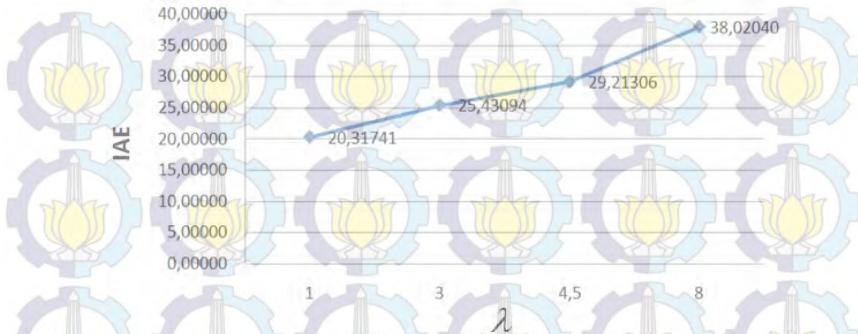
Parameter respon yang digunakan adalah *settling time*, *maximum overshoot* dan IAE. Parameter yang diperoleh ditunjukkan pada Tabel 4.1 berikut,

**Tabel 4.1** Parameter Respon IMC Controller Tanpa Disturbance

Parameter Tuning IMC	Maximum Overshoot (%)	Settling Time (s)	IAE
$\lambda = 1$	0	14,56	20,32
$\lambda = 3$	0	22,33	25,43
$\lambda = 4,5$	0	28,16	29,21
$\lambda = 8$	0	41,75	38,02

Dari hasil parameter respon kontrol IMC dapat diketahui bahwa respon sistem dengan menggunakan kontrol IMC dapat mencapai *set point*. Parameter tuning IMC yang paling baik adalah IMC control dengan nilai  $\lambda = 1$ . Nilai IAE didapatkan dari jumlah selisih antara data respon dan set poin pada *workspace* MatLab yang yang dimutlakkan. Maksud dari dimutlakkan adalah *error* antara respon dan *set point* semuanya dianggap bernilai *plus*.

### PERBANDINGAN NILAI LAMDA DAN IAE

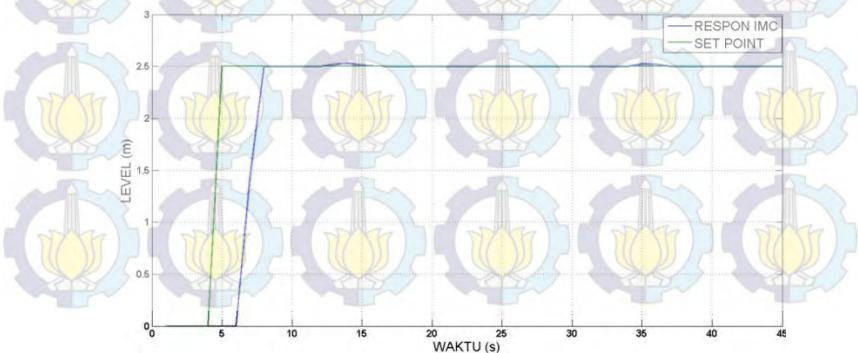


**Gambar 4.3** Perbandingan Nilai  $\lambda$  Dan IAE

Dari grafik pada Gambar 4.3, dapat dilihat bahwa semakin besar nilai  $\lambda$  maka *Integral Absolute Error* (IAE) akan semakin besar. Hal ini dikarenakan nilai *settling time* dari respon juga semakin besar.

#### 4.2.2 Hasil *Controller IMC* Dengan *Disturbance*

Pemasangan *disturbance* dilakukan dengan menggunakan nilai parameter tuning ( $\lambda$ ) sebesar 1. Hal ini dikarenakan nilai  $\lambda = 1$  memiliki respon paling baik. Hasil respon kontrol IMC tanpa *disturbance* ditunjukkan pada Gambar 4.4 berikut,



**Gambar 4.4** Respon IMC *Controller* Dengan *Disturbance*

Parameter respon yang diperoleh dari Respon IMC *controller* dengan *disturbance* ditunjukkan pada Tabel 4.2 berikut,

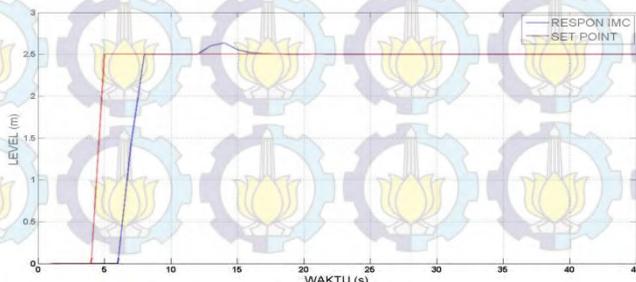
**Tabel 4.2** Parameter Respon IMC *Controller* Dengan *Disturbance*

Settling Time (s)	Maximum Overshoot(%)		IAE
	Steam	Air Kondensat	
14,56	1,748	1,2128	20,93

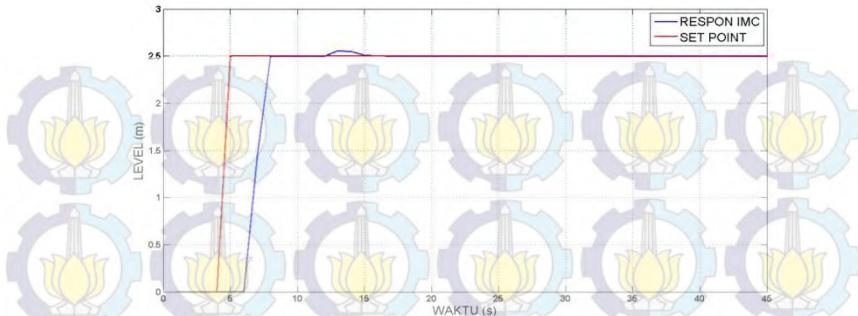
Penambahan *disturbance* dilakukan dengan nilai *mass flow* air kondensat sebesar 50 ton/h dan *mass flow steam* sebesar 300 ton/h. Dari Tabel 4.2, diperoleh bahwa kontrol IMC mampu mengurangi nilai *disturbance* sehingga level dalam *deaerator* dapat kembali dalam keadaan *steady*. Simulasi dilakukan dengan *step time* yang berbeda agar mudah melihat perubahan respon akibat adanya penambahan dua *disturbance*.

### 4.3 Uji *Disturbance*

Uji *disturbance* dilakukan untuk mengetahui seberapa besar pengaruh *disturbance* terhadap level pada *deaerator*. Pengujian *disturbance* dilakukan secara *switch* atau dengan kata lain ketika satu *disturbance* masuk, maka *disturbance* lain dianggap nol. Respon uji *disturbance* ditunjukkan seperti pada Gambar 4.5 berikut,



**Gambar 4.5** Respon Uji *Disturbance* Air Kondensat



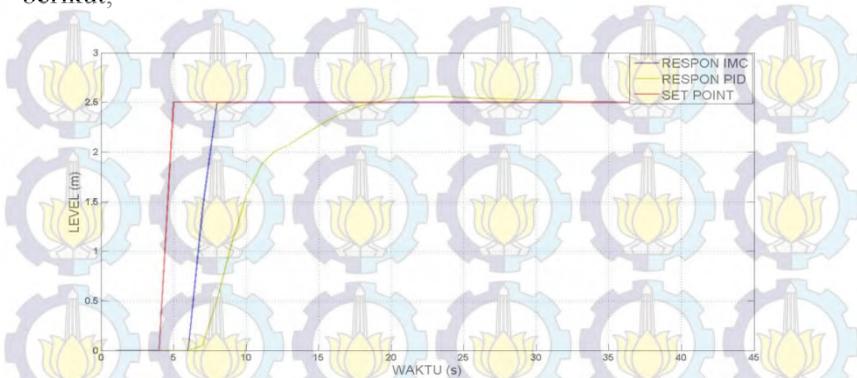
**Gambar 4.6** Respon Uji *Disturbance Steam*

*Mass flow* yang diberikan pada uji *disturbance* adalah sama. Karena tujuan dari uji *disturbance* itu sendiri adalah untuk mengetahui pengaruh dari masing – masing *disturbance* terhadap level *deaerator*. Nilai *maximum overshoot* dari uji *disturbance* air kondensat sebesar 6,148% dan *steam* 2,208%. Dari hasil respon uji *disturbance*, dapat diperoleh bahwa *maximum overshoot* dari air kondensat lebih besar daripada *steam*. Hal tersebut dikarenakan *gain* dari air kondensat lebih besar daripada *gain steam*. Jadi, dengan nilai *mass flow* yang sama, air kondensat lebih mempengaruhi level dari *deaerator*. Selain itu, perbedaan *maximum overshoot* juga dikarenakan *steam* lebih banyak yang keluar mengacu pada proses *deaerator* itu sendiri.

#### 4.4 Hasil Perbandingan Performansi Pengendali

Perbandingan performansi pengendali dilakukan dengan membandingkan kontrol IMC dan kontrol PID. Kontrol IMC yang dibandingkan adalah kontrol IMC dengan nilai  $\lambda = 1$ . Sedangkan nilai kontrol PID menggunakan nilai  $K_c = 1,2$  dan nilai  $T_i = 0,06$ . Parameter tuning PID dicari dengan menggunakan *software* MatLab dengan cara *auto tuning*. Penggunaan *software* MatLab untuk mencari nilai parameter tuning PID dilakukan karena simulasi dilakukan tanpa adanya *disturbance*. Sedangkan data nilai parameter tuning PID yang didapatkan dari pabrik adalah parameter tuning PID yang dengan adanya *disturbance*.

Hasil perbandingan performansi ditunjukkan pada Gambar 4.7 berikut,



**Gambar 4.7** Respon Perbandingan Performansi Pengendali

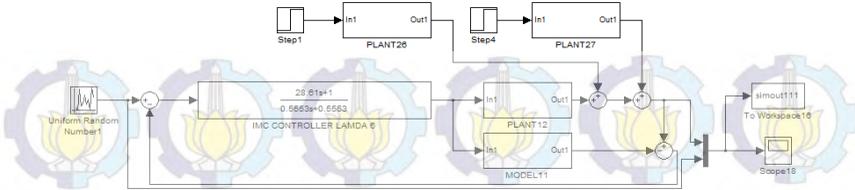
Parameter respon perbandingan performansi pengendali yang diperoleh dari Respon IMC *controller* dengan *disturbance* ditunjukkan pada Tabel 4.3 berikut,

**Tabel 4.3** Perbandingan Parameter Respon Pengendali

<i>Controller</i>	<i>Settling Time</i> (s)	<i>Maximum Overshoot</i> (%)	IAE
IMC	14,56	0	20,32
PID	78,64	2,26	89,55

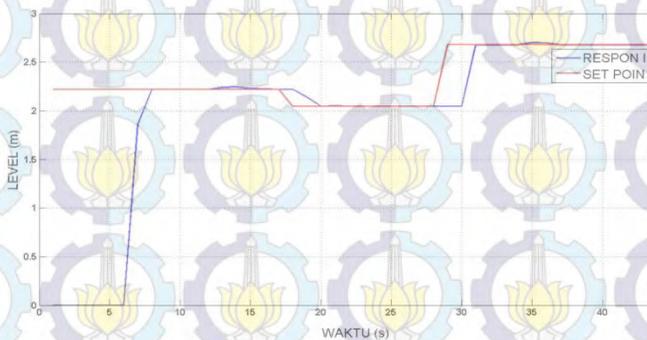
#### 4.5 Pengujian *Tracking Set Point* Pengendali IMC

Uji *tracking set point* bertujuan untuk mengetahui performansi pengendali IMC apabila *set point* pada proses dirubah *set point real plant* adalah 2,5 m. Simulasi *tracking set point* dilakukan dengan *software* Matlab 2009a untuk mendapatkan respon pengendali seperti pada Gambar 4.8 berikut,



**Gambar 4.8** Simulasi *Tracking Set Point* Pengendali IMC

Hasil respon uji *tracking set point* pengendali IMC ditunjukkan pada Gambar 4.9 berikut,



**Gambar 4.9** Respon Uji *Tracking Set Point* Pengendali IMC

Dari hasil respon uji *tracking set point*, didapatkan bahwa pengendali IMC mampu mencapai *set point* pada saat *set point* dirubah. Pengujian *tracking set point* ini dilakukan dengan menggunakan parameter yang sama pada saat perbandingan performansi pengendali.

#### 4.6 Pembahasan

Perancangan pengendalian level *deaerator* pada penelitian ini adalah dengan menggunakan metode *Internal Model Control*. Respon dari pengendalian dengan metode IMC dilakukan dengan menentukan nilai parameter tuning IMC yaitu  $\lambda$ . Simulasi dari model sistem *deaerator* menggunakan *software* Hysys. Pemodelan sistem dilakukan dengan metode FOPDT untuk

mencari *gain process* dan *gain disturbance*. Dari respon kontrol IMC, menunjukkan bahwa *controller* dengan nilai  $\lambda = 1$  mempunyai respon yang paling baik dengan nilai *maximum overshoot* yaitu 0% tanpa adanya *disturbance*. Dasar teori dari parameter tuning IMC menyatakan bahwa untuk mencari respon yang paling cepat, nilai harus diperkecil. Respon dari kontrol IMC dengan nilai  $\lambda$  paling kecil, mempunyai nilai *settling time* yang kecil pula yaitu 14,56 s. Nilai  $\lambda = 1$  digunakan untuk perbandingan performansi pengendalian yang telah digunakan yaitu pengendalian PID. Hasil uji performansi pengendalian menunjukkan bahwa respon pengendalian IMC memiliki respon yang lebih baik dari pengendalian PID dengan nilai *settling time* sebesar 78,64 s. Nilai *maximum overshoot* yang ditunjukkan dari pengendalian PID sebesar 2,26% yang lebih besar dari pengendalian IMC yaitu 0%. *Integral absolute error* pengendalian PID mempunyai nilai yang lebih besar pula 89,55 daripada pengendalian IMC yaitu sebesar 21,32. Uji respon dengan *disturbance* menggunakan kontrol IMC, menunjukkan bahwa kontrol IMC mampu mengurangi nilai *disturbance*. *Maximum overshoot* yang ditimbulkan akibat adanya *disturbance* juga tidak lebih dari nilai toleransi dari *maximum overshoot* yaitu 20%. Perbedaan nilai *maximum overshoot disturbance* dikarenakan perbedaan *gain* dari masing – masing *disturbance*. Selain itu, *mass flow* dari air kondensat juga lebih kecil daripada *steam*. Dengan kata lain, *steam* lebih berpengaruh terhadap level *deaerator*. Dari hal tersebut, dapat menjelaskan tentang suntikkan cairan hidrazin pada *deaerator* untuk membantu menghilangkan. Karena apabila level dari *deaerator* tidak sesuai dengan *set point*, maka *steam* yang berubah fasa menjadi *liquid* semakin banyak. Hal tersebut akan mempengaruhi kandungan  $O_2$  pada *deaerator* itu sendiri. Karena apabila terlalu banyak *steam* yang masuk, maka  $O_2$  yang terbuang oleh *steam* akan sedikit. Oleh karena itu, suntikkan hidrazin perlu dilakukan.

## BAB V KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

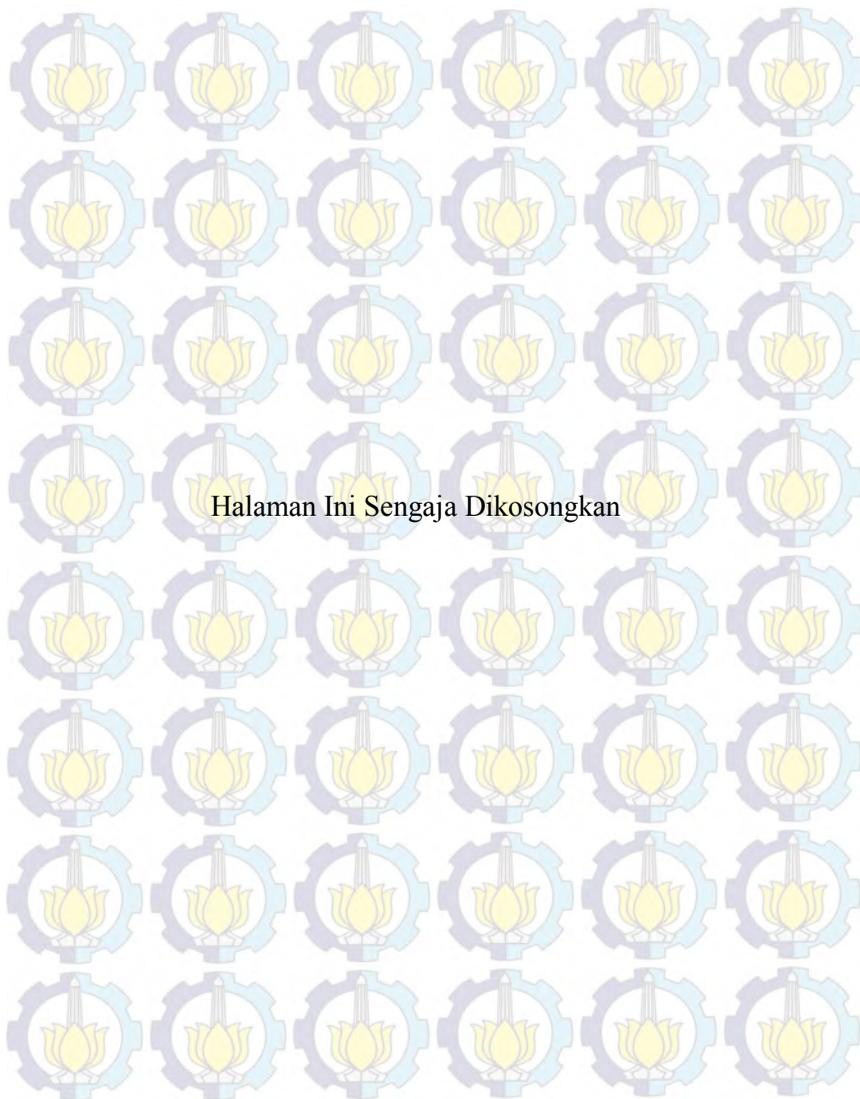
Dari penelitian tugas akhir ini dapat disimpulkan bahwa :

1. Perancangan pengendalian level pada *deaerator* 101U dengan metode *internal model control* (IMC) dapat dilakukan dengan memperkecil nilai parameter *tuning* IMC ( $\lambda$ ) untuk mendapatkan respon yang terbaik.
2. Pengendali IMC dengan nilai parameter *tuning* IMC ( $\lambda$ ) sebesar 1 memiliki performansi nilai *settling time* sebesar 14,56 s dan *Integral Absolute Error* sebesar 20,92.

### 5.2 Saran

Dari hasil tugas akhir terdapat beberapa saran untuk pengembangan penelitian selanjutnya antara lain :

1. Dapat menambahkan nilai komposisi yang lebih banyak karena pada penelitian kali ini komposisi dianggap hanya air dan oksigen.
2. Dapat dilakukan perancangan pengendali IMC sampai mengetahui *tuning* IMC berbasis PID karena pada penelitian kali ini hanya memakai pengendalian IMC.



## DAFTAR PUSTAKA

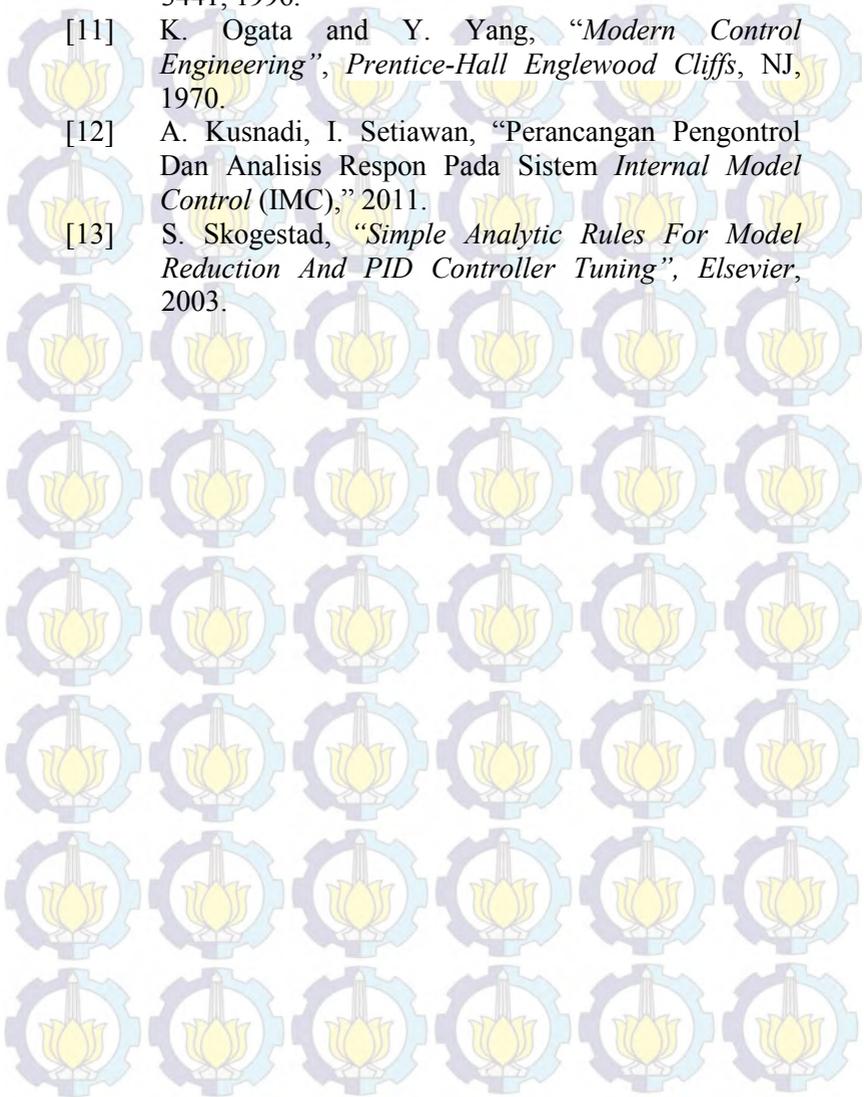
- [1] Ariyanto, "Perancangan Sistem Pengendalian Level Deaerator Menggunakan Fuzzy Gain Scheduling – PI Di PT Petrowidada," 2013.
- [2] A. PORWAL, "*Internal Model Control (IMC) And IMC Based PID Controller*," National Institute of Technology Rourkela, 2010.
- [3] I. Setiawan, "Kontrol PID untuk Proses Industri," 2008.
- [4] K. Rajalakshmi and V. Mangaiyarkarasi, "Design of internal model controller for a heat exchanger system," in *Information Communication and Embedded Systems (ICICES), 2013 International Conference on*, 2013, pp. 899-903.
- [5] I. OPRİŞ, "A Deaerator Model," in *Recent Advances in Continuum Mechanics, Hydrology and Ecology, Energy, Environmental and Structural Engineering Series-14, WSEAS International Conference, Rhodes Island, Greece*, 2013.
- [6] M. T. Tham, "Internal model control," *Introduction To Robust Control-[en ligne]-Disponible sur <<http://lorien.ncl.ac.uk/ming/robust/imc.pdf>>(Consulté le 12.11. 2010)*, 2002.
- [7] B. W. Bequette, *Process control: modeling, design, and simulation*: Prentice Hall Professional, 2003.
- [8] H. Iqbal, "Perancangan Mimo Kontrol (Level Dan Pressure) Unit Deaerator Menggunakan Metode Decoupler," 2008.
- [9] K. Decker, "FOPDT Modeling With MS Excel," 2007.
- [10] I. G. Horn, J. R. Arulandu, C. J. Gombas, J. G. VanAntwerp, and R. D. Braatz, "Improved filter design in internal model control," *Industrial &*

*engineering chemistry research*, vol. 35, pp. 3437-3441, 1996.

[11] K. Ogata and Y. Yang, “*Modern Control Engineering*”, Prentice-Hall Englewood Cliffs, NJ, 1970.

[12] A. Kusnadi, I. Setiawan, “Perancangan Pengontrol Dan Analisis Respon Pada Sistem *Internal Model Control* (IMC),” 2011.

[13] S. Skogestad, “*Simple Analytic Rules For Model Reduction And PID Controller Tuning*”, Elsevier, 2003.



## BIODATA PENULIS



Moammar Arief Hidayat merupakan nama lengkap penulis dengan nama panggilannya, Arief. Penulis dilahirkan di Lamongan, Jawa Timur pada tanggal 2 April 1993 sebagai anak kedua dari pasangan Imam Suprayitno dan Siti Asmikah. Riwayat pendidikan penulis adalah SD Negeri Babat VII (1999 – 2005), SMP Negeri 1 Babat tahun (2005 – 2008), SMA Negeri 1 Babat tahun (2008-2010), SMA Negeri 2 Lamongan tahun (2010 – 2011). Penulis diterima sebagai mahasiswa S1 Teknik Fisika ITS pada tahun 2011, kemudian fokus pada bidang minat rekayasa instrumentasi dan kontrol untuk menyelesaikan tugas akhirnya. Penulis dapat dihubungi melalui email, [moammarhidayat@gmail.com](mailto:moammarhidayat@gmail.com).