



**TUGAS AKHIR - TF 181801**

**PREDIKSI KEANDALAN *REAL-TIME* PADA  
SISTEM PENGENDALIAN *LEVEL* SEPARATOR  
BERDASARKAN PREDIKSI KESALAHAN  
AKTUATOR**

**NUR UMMAMA SOFYANA  
NRP. 0231174500004**

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

**DEPARTEMEN TEKNIK FISIKA  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya  
2019**

*“Halaman ini sengaja dikosongkan”*



**FINAL PROJECT - TF 181801**

**REAL-TIME RELIABILITY PREDICTION ON  
LEVEL SEPARATOR CONTROL SYSTEM BASED  
ON ACTUATOR ERROR PREDICTION**

NUR UMMAMA SOFYANA

NRP. 02311745000004

*Supervisor:*

Ir. Ya'umar, M.T.

**ENGINEERING PHYSICS DEPARTMENT**

*Faculty of Industrial Technology*

*Sepuluh Nopember Institute of Technology*

*Surabaya*

*2019*

*“Halaman ini sengaja dikosongkan”*

## **PERNYATAAN BEBAS PLAGIARISME**

Saya yang bertanda tangan di bawah ini

Nama : Nur Ummama Sofyana  
NRP : 0231174500004  
Departemen/Prodi : Teknik Fisika/S1 Teknik Fisika  
Fakultas : Fakultas Teknologi Industri  
Perguruan Tinggi : Institut Teknologi Sepuluh Nopember

Dengan ini menyatakan bahwa Tugas Akhir dengan judul “Prediksi Keandalan *Real-Time* Pada Sistem Pengendalian *Level* Separator Berdasarkan Prediksi Kesalahan Aktuator” adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain. Apabila di kemudian hari terbukti terdapat plagiat pada Tugas Akhir ini maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya.

Surabaya, 10 Juni 2019  
Yang membuat pernyataan,



Nur Ummama Sofyana  
NRP. 0231174500004

*“Halaman ini sengaja dikosongkan”*

**LEMBAR PENGESAHAN  
TUGAS AKHIR  
PREDIKSI KEANDALAN *REAL-TIME* PADA SISTEM  
PENGENDALIAN *LEVEL* SEPARATOR BERDASARKAN  
PREDIKSI KESALAHAN AKTUATOR**

Oleh:

**Nur Ummama Sofyana**  
NRP.0231174500004

Surabaya, 24 Juli 2019

Menyetujui,  
Dosen Pembimbing



**Ir. Yaumar, M.T.**  
NIP. 19540406 198103 1 003

Mengetahui,  
Kepala Departemen  
Teknik Fisika FTI-ITS



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

*“Halaman ini sengaja dikosongkan”*



**PREDIKSI KEANDALAN *REAL-TIME* PADA SISTEM  
PENGENDALIAN *LEVEL* SEPARATOR BERDASARKAN  
PREDIKSI KESALAHAN AKTUATOR**


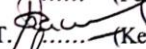


**TUGAS AKHIR**

Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Teknik  
pada  
Program Studi S-1 Departemen Teknik Fisika  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember

Oleh:

**NUR UMMAMA SOFYANA**  
**NRP.0231174500004**

Disetujui oleh Tim Penguji Tugas Akhir :

1. Ir. Ya'umar, M.T.  (Pembimbing)
2. Dr. Bambang L. Widjiantoro, S.T. M.T.  (Ketua Penguji)
3. Dr. Katherin Indriawati, S.T., M.T.  (Penguji I)
4. Lizda Johar Mawarani, S.T., M.T.  (Penguji II)

**SURABAYA**  
**Juli, 2019**

*“Halaman ini sengaja dikosongkan”*

**PREDIKSI KEANDALAN *REAL-TIME* PADA SISTEM  
PENGENDALIAN *LEVEL* SEPARATOR BERDASARKAN  
PREDIKSI KESALAHAN AKTUATOR**

**Nama** : Nur Ummama Sofyana  
**NRP** : 0231174500004  
**Departemen** : Teknik Fisika FTI-ITS  
**Dosen Pembimbing** : Ir. Ya'umar, M.T.

**ABSTRAK**

Kerja dari separator dipengaruhi oleh *level* untuk menjaga kemurnian fluida. Seiring penggunaannya performansi separator akan menurun karena degradasi komponen yang menyebabkan terjadinya kegagalan fungsi dari separator, salah satunya gangguan berupa kesalahan aktuator. Agar separator dapat bekerja secara optimal maka diperlukan peningkatan keandalan pada sistem. Pada tugas akhir ini dirancang sistem prediksi keandalan *real-time* sistem pengendalian *level* separator. Langkah utama yang dilakukan dalam perancangan sistem ini adalah perancangan *observer* untuk mengestimasi kesalahan, perancangan algoritma *exponential smoothing* untuk memprediksi kesalahan dan perancangan algoritma *reliability prediction* untuk menghitung keandalan. Pada *Observer* hasil estimasi kesalahan mengikuti dinamika kesalahan residual atau riil yang terjadi dan terdapat deviasi dengan nilai sebesar 0.064%. Pada algoritma *exponential smoothing* ditemukan bahwa *time interval* terbaik adalah 1s . Pada uji coba algoritma *reliability prediction* ditemukan bahwa keandalan dengan nilai prediksi yang lebih depan dapat lebih dini mengindikasikan terjadinya kegagalan pada sistem.

**kata kunci** : separator, kesalahan aktuator, *observer*, *exponential smoothing*, prediksi keandalan

*“Halaman ini sengaja dikosongkan”*

**REAL-TIME RELIABILITY PREDICTION ON LEVEL  
SEPARATOR CONTROL SYSTEM BASED ON ACTUATOR  
ERROR PREDICTION**

**Name** : Nur Ummama Sofyana  
**NRP** : 02311745000004  
**Departement** : Teknik Fisika FTI-ITS  
**Supervisors** : Ir. Ya'umar, M.T.

**ABSTRACT**

*Work of the separator is affected by the level to maintain fluid purity. Along with the use of the separator, the performance of separator will decrease due to the degradation of the component which causes the failure of the separator, one of which is the interference of an actuator error. In order to work optimally, separator need to increase its reliability system. In this final project, a system for predicting reliability of real-time control system for separator level is designed. The main step taken in designing this system is designing an observer to estimate errors, designing exponential smoothing algorithms to predict errors and designing reliability prediction algorithms to calculate reliability. From the Observer, the results of errors estimation follow the dynamics of residual or real errors and there is a deviation with a value of 0.064%. From the exponential smoothing algorithm it is found that the best time interval is 1 seconds. From reliability prediction algorithm, it is found that reliability with an earlier prediction value could indicate a failure in the system more earlier.*

**Keywords:** *separator, actuator error, observer, exponential smoothing, reliability prediction*

*“Halaman ini sengaja dikosongkan”*

## **KATA PENGANTAR**

Puji syukur kehadiran Allah SWT yang senantiasa melimpahkan rahmat serta hidayah-Nya, serta shalawat serta salam kepada Nabi Muhammad SAW, hingga terselesaikannya tugas akhir beserta laporan yang berjudul.

### **PREDIKSI KEANDALAN *REAL-TIME* PADA SISTEM PENGENDALIAN *LEVEL* SEPARATOR BERDASARKAN PREDIKSI KESALAHAN AKTUATOR**

Penulis telah banyak memperoleh bantuan dari berbagai pihak dalam penyelesaian tugas akhir dan laporan Tugas Akhir ini. Penulis mengucapkan terimakasih kepada :

1. Bapak Agus Muhamad Hatta, S.T.,M.Si., Ph.D. selaku ketua departemen teknik fisika yang telah memberikan petunjuk, ilmu, serta bimbingan selama menempuh pendidikan di teknik fisika.
2. Ir. Ya'umar, M.T.selaku dosen pembimbing yang telah memberikan ilmu, nasihat serta bimbingan yang sangat bermanfaat.
3. Ibu Dr. Katherin Indriawati, S.T., M.T. selaku dosen yang juga telah dengan sabar memberikan petunjuk, ilmu serta bimbingan yang sangat bermanfaat.
4. Ibu Dr.Ir Ronny Dwi Noriyati, M.Kes Selaku dosen wali yang telah membimbing penulis selama perkuliahan.
5. Kedua orang tua. Terimakasih atas segala cinta, kasih sayang, doa, perhatian serta dukungan moril dan materiil yang telah diberikan.
6. Aloys Yantri Martina teman seperjuangan yang selalu bersama mengerjakan tugas akhir ini, terimakasih atas dukungan dan kerja samanya.

7. Teman-teman TAWan Laboratorium Fisika Rekayasa yang selalu memberi semangat dan dukungannya
8. Seluruh teman-teman Lintas Jalur Departemen Teknik Fisika angkatan 2017, terima kasih untuk semuanya.
9. Seluruh dosen karyawan dan civitas akademik Teknik Fisika, terimakasih atas segala bantuan dan kerjasamanya.
10. Semua pihak yang tidak dapat disebutkan satu persatu, terimakasih atas bantuannya.

Penulis sadar bahwa penulisan laporan tugas akhir ini tidak sempurna, namun semoga laporan ini dapat memberikan kontribusi berarti dan menambah wawasan yang bermanfaat bagi pembaca, keluarga besar Teknik Fisika khususnya, dan civitas akademik ITS pada umumnya. Semoga laporan tugas akhir ini dapat bermanfaat sebagai referensi pengerjaan laporan tugas akhir bagi mahasiswa yang lain.

Surabaya, 13 Mei 2019

Penulis



## DAFTAR ISI

HALAMAN JUDUL.....	i
PERNYATAAN BEBAS PLAGIARISME .....	v
LEMBAR PENGESAHAN .....	vii
ABSTRAK .....	xi
<i>ABSTRACT</i> .....	xiii
KATA PENGANTAR.....	xv
DAFTAR ISI.....	xvii
DAFTAR GAMBAR .....	xix
DAFTAR TABEL .....	xxi
DAFTAR NOTASI .....	xxiii
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah .....	3
1.3 Tujuan.....	4
1.4 Batasan Masalah.....	4
BAB II TEORI PENUNJANG.....	5
2.1 Separator.....	5
2.2 Pengendalian <i>Level</i> Pada Separator.....	6
2.3 Persamaan Dinamik Separator.....	7
2.4 Sistem Kendali PI.....	9
2.5 <i>Level Transmitter</i> .....	11
2.6 <i>Control Valve</i> .....	13
2.7 Observer .....	14
2.8 Algoritma <i>Exponential Smoothing</i> .....	18
2.9 Keandalan.....	19
2.10 Prediksi Keandalan <i>Real-Time</i> .....	20
BAB III METODOLOGI PENELITIAN.....	23
3.1 Studi literatur.....	24
3.2 Data Pemodelan <i>Plant</i> .....	24
3.3 Pemodelan Matematis Sistem.....	24

3.4 Uji <i>Open Loop Plant</i> .....	29
3.5 Perancangan sistem kendali PI .....	30
3.6 Perancangan <i>Observer</i> .....	31
3.7 Validasi <i>Observer</i> .....	37
3.8 Perancangan algoritma prediksi kesalahan.....	38
3.9 Pengujian algoritma prediksi kesalahan aktuator .....	39
3.10 Perancangan Algoritma Prediksi Keandalan Real-Time	40
3.11 Pengujian Algoritma Prediksi Keandalan Real-Time ....	41
<b>BAB IV ANALISIS DATA DAN PEMBAHASAN .....</b>	<b>43</b>
4.1. Hasil Uji Observer .....	43
4.2 Hasil Uji Algoritma Prediksi Kesalahan Aktuator .....	45
4.3 Hasil Uji Algoritma Prediksi <i>Keandalan Real-Time</i> .....	51
<b>BAB V PENUTUP .....</b>	<b>55</b>
5.1 Kesimpulan.....	55
5.2 Saran .....	55
<b>DAFTAR PUSTAKA</b>	
<b>LAMPIRAN</b>	
<b>BIODATA PENULIS</b>	

## DAFTAR GAMBAR

Gambar 2.1	Separator 2 fasa .....	5
Gambar 2.2	Diagram blok pengendalian <i>level</i> separator.....	6
Gambar 2.3	Instrument pengendali separator (P&ID separator Saka Pangkah Indonesia Limited) .....	7
Gambar 2.4	Diagram blok sistem pengendalian tertutup .....	10
Gambar 2.5	Diagram Blok Pengendalian PI .....	10
Gambar 2.6	LDT ( <i>Differential Pressure Transmitter</i> ).....	12
Gambar 2.7	ATC dan ATO <i>control valve</i> .....	13
Gambar 2.8	Observer .....	17
Gambar 2.9	Kestabilan <i>pole placement</i> .....	18
Gambar 3.1	Diagram alir penelitian tugas akhir .....	23
Gambar 3.2	Respon <i>open loop</i> .....	30
Gambar 3.3	Hasil Respon <i>Plant</i> .....	31
Gambar 3.4	Diagram blok sistem pengendalian yang telah terintegrasi dengan <i>observer</i> .....	38
Gambar 3.5	Hasil estimasi state level dari <i>observer</i> dibandingkan dengan respon dari <i>plant</i> . .....	39
Gambar 3.6	Diagram blok sistem prediksi keandalan <i>real-time</i>	41
Gambar 4.1	Hasil estimasi kesalahan aktuator.....	43
Gambar 4.2	Hasil estimasi kesalahan aktuator (perbesaran Gambar 4.1) .....	44
Gambar 4.3	Hasil uji prediksi kesalahan aktuator dengan <i>time interval</i> 1 detik.....	47
Gambar 4.4	Hasil uji prediksi kesalahan aktuator dengan <i>time interval</i> 3 detik.....	49
Gambar 4.5	Hasil uji prediksi kesalahan aktuator dengan <i>time interval</i> 5 detik.....	51
Gambar 4.6	Grafik hasil uji prediksi keandalan <i>real-time</i> dengan <i>time interval</i> 1 detik.....	54

*“Halaman ini sengaja dikosongkan”*

## DAFTAR TABEL

Tabel 3.1	Data Spesifikasi Separator.....	24
Tabel 4.1	Perbandingan nilai riil dan hasil estimasi kesalahan aktuator .....	45
Tabel 4.2	Hasil prediksi kesalahan aktuator dengan <i>time interval</i> 1 s .....	46
Tabel 4.3	Hasil prediksi kesalahan aktuator dengan <i>time interval</i> 3 s .....	48
Tabel 4.4	Hasil prediksi kesalahan aktuator dengan <i>time interval</i> 5 s .....	50
Tabel 4.5	Hasil uji prediksi keandalan <i>real-time</i> dengan <i>time interval</i> 1 s.....	53

*“Halaman ini sengaja dikosongkan”*

## DAFTAR NOTASI

$\hat{x}(t)$	: estimasi <i>state</i> $x(t)$
$\hat{y}(t)$	: estimasi $y(t)$
$\hat{f}(t)$	: estimasi kesalahan aktuator $f(t)$
$A$	: matriks keadaan ( <i>state</i> )
$B$	: matriks kendali
$C$	: matriks keluaran
$l$	: step prediksi untuk prediksi kesalahan aktuator
$n$	: <i>discrete time index</i>
$r$	: step prediksi untuk prediksi keandalan <i>real-time</i>
$s_n$	: faktor pertumbuhan linear yang diestimasi
$u_n$	: hasil estimasi yang diperhalus
$x(t)$	: vektor keadaan ( <i>state</i> )
$y(t)$	: vektor keluaran
$\mathcal{M}_0$	: Batas nilai variabel
$a, b$	: <i>Parameter Holt's method</i>
$T$	: <i>Time interval</i>

*“Halaman ini sengaja dikosongkan”*



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Energi memegang peranan yang sangat penting disemua sektor usaha, baik itu sektor industri, bangunan komersial, transportasi, pertambangan dan banyak lagi. Sektor migas adalah sektor energi yang hingga kini masih menguasai diantara sektor-sektor energi lainnya. Dengan semakin pesatnya pertumbuhan dan pembangunan sektor usaha di Indonesia menyebabkan kebutuhan energi terutama migas juga semakin meningkat. Untuk memenuhi kebutuhan tersebut perusahaan yang bergerak di bidang migas baik hulu maupun hilir akan terus bersaing untuk dapat memenuhi kebutuhan pasar. Salah satu perusahaan migas milik pemerintah adalah Saka Indonesia Pangkah Limited (SIPL) yang bergerak di bidang industri hulu migas.

Dalam proses pengolahan minyak terdapat separator yang berfungsi memisahkan memisahkan fluida produksi kedalam fasa liquid dan fasa gas. Prinsip kerja dari separator yaitu fluida yang berasal dari *manifold* akan masuk ke separator melalui lubang *inlet* dan selanjutnya akan menabrak *inlet diverter*. Prinsip gaya centrifugal digunakan dalam proses pemisahannya, gaya centrifugal inilah yang nantinya mendorong fluida ke dinding, sehingga liquid yang memiliki massa jenis lebih berat tertumbuk dan jatuh kebawah sementara gas yang memiliki massa jenis lebih ringan akan keatas. (Wang, 2000)

Kerja dari Separator akan dipengaruhi oleh *Level* dan Tekanan untuk menjaga kemurnian fluida. Jika level liquid pada separator memiliki ketinggian yang melebihi ketinggian yang diizinkan maka liquid akan tercampur dengan gas, tetapi pada saat liquid berada di level yang terendah maka gas akan tercampur ke liquid.

Terjadinya kegagalan fungsi dari separator dapat berdampak pada kinerja sistem menjadi tidak optimal sehingga performansi sistem menjadi kurang baik. Selain itu, juga akan menghambat kegiatan produksi minyak jika terjadi *trip/shut down*. Salah satu

permasalahan yang terjadi yaitu kegagalan sistem pada separator disebabkan adanya gangguan berupa kesalahan aktuator pada sistem pengendalian *level*.

Aktuator dalam hal ini *Level Control Valve* harus di *preventive maintenance* setiap *half year*, kemudian berdasarkan data kesalahan pada LCV-111 belum pernah terjadi. Sehingga hal tersebut yang melatar belakangi penulis melakukan penelitian terkait prediksi kesalahan aktuator guna menanggulangi atau menolerir kesalahan pembacaan aktuator ketika proses berjalan.

Upaya agar separator dapat bekerja secara optimal maka yang diperlukan adalah peningkatan keandalan pada sistem. Keandalan suatu sistem ditentukan secara keseluruhan ketika sistem itu dibuat. Namun, keandalan sistem tersebut akan menurun selama penggunaan karena degradasi komponen dan pengaruh eksternal sistem. Semakin besar kemungkinan sistem akan mengalami kegagalan, semakin rendah tingkat keandalan sistem tersebut. Oleh karena itu, keandalan suatu sistem terkait erat dengan *lifetime* sistem. Jembatan yang menghubungkan kendalan dengan *lifetime* sistem adalah evaluasi dan monitoring keandalan *real-time*. Analisis keandalan *real-time* biasanya mempertimbangkan penurunan performansi, yang ditandai oleh beberapa variabel kinerja. Variabel kinerja tersebut diukur *real-time* dan sistem atau komponen dianggap gagal saat variabel kinerja mencapai beberapa ambang batas yang telah ditentukan sebelumnya (Xu, 2009). Penelitian di bidang ini telah berkembang sejak awal 1990-an diantaranya adalah (Kolarik & Lu, 2001)(Chinnam & Mohan, 2002)(Xu et al., 2008).

Pada penelitian sebelumnya telah diperkenalkan monitoring keandalan *real-time*, yang merupakan usaha mengevaluasi kondisi sistem tersebut secara periodik sehingga sistem dapat melaksanakan fungsinya dengan baik, tanpa adanya kegagalan pada periode operasi yang ditentukan. Metode baru dalam memonitoring keandalan *real-time* untuk sistem dinamik yang digabungkan dengan algoritma estimasi kesalahan. Faktor yang dapat mengurangi keandalan sistem dinamik tersebut dimodelkan sebagai sebuah masukan kesalahan tambahan pada sistem, dan

kesalahan diasumsikan *linear* terhadap waktu (Xu dkk, 2009) Monitoring ini juga bisa digunakan untuk memprediksi keandalan sebuah sistem. Algoritma estimasi kesalahan *real-time* dapat dinyatakan dalam sebuah *observer*. Sebelumnya (Walijah, 2017) dan telah dilakukan penelitian mengenai penggunaan *observer* pada perancangan *Fault Tolerant Control* untuk mengakomodasi kesalahan pada sensor dan aktuator pada sistem pengendalian level separator. Monitoring Keandalan *Real-Time* Berdasarkan Estimasi Kesalahan (Fitria, 2018)

Prediksi keandalan digunakan untuk menjejaki peningkatan keandalan dan dapat juga membantu mengevaluasi signifikansi dari kegagalan yang dilaporkan. Prediksi keandalan ini juga perlu diterapkan untuk sistem pengendalian level pada separator, sehingga kegagalan yang akan terjadi pada separator dapat dideteksi lebih dini. Dengan dilakukannya prediksi keandalan dapat meningkatkan umur peralatan, mengurangi waktu *downtime* dan kerusakan yang lebih parah dapat dihindari.

Oleh karena itu, penelitian ini merancang prediksi keandalan *real-time* pada sistem pengendalian *level* separator berdasarkan estimasi kesalahan aktuator, yang diharapkan mampu menghasilkan nilai prediksi keandalan *real-time* yang akurat.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan diatas, maka didapatkan permasalahan dalam tugas akhir ini, yaitu:

- Bagaimana merancang *observer* yang dapat digunakan untuk mendapatkan estimasi kesalahan pada aktuator pada sistem pengendalian *level* separator?
- Bagaimana merancang prediksi kesalahan aktuator pada sistem pengendalian *level* separator menggunakan algoritma *Exponential Smoothing* ?
- Bagaimana cara menghitung keandalan sistem pengendalian *level* separator menggunakan algoritma *Reliability Prediction* (RP) ?

### 1.3 Tujuan

Berdasarkan rumusan masalah tersebut, maka tujuan pada tugas akhir ini adalah sebagai berikut :

- Merancang *observer* pada sistem pengendalian *level* separator yang mampu mengestimasi kesalahan.
- Memprediksi nilai kesalahan pada sistem pengendalian *level* seapartor menggunakan algoritma *Exponential Smoothing*.
- Menghitung keandalan sistem pengendalian *level* separator secara *real-time* menggunakan algoritma *Reliability Prediction* (RP).

### 1.4 Batasan Masalah

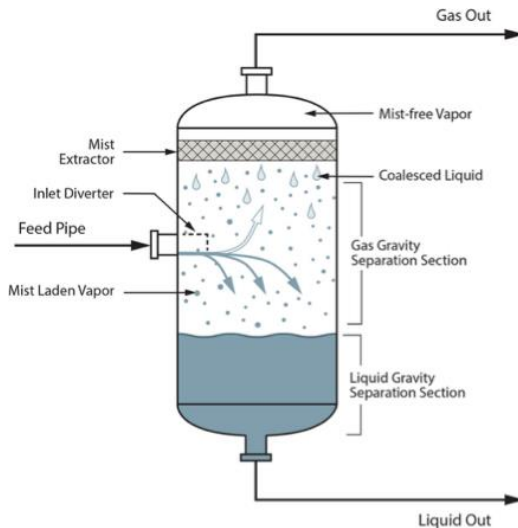
Adapun batasan masalah dalam penelitian ini antara lain:

- *Fault* pada pengendalian *level* separator terjadi pada aktuator.
- Kesalahan yang ditinjau pada aktuator berupa kesalahan bias
- *Controller* yang terpasang pada separator berupa pengendalian PI.
- Pemodelan plant hanya meninjau pada kesetimbangan massa di separator.
- Keandalan yang dimaksud dalam penelitian ini adalah nilai keandalan oleh algoritma RP, bukan keandalan yang diperoleh dari data maintenance
- Sistem dikatakan gagal ketika tinggi *level* telah melebihi batas maksimum *level* separator yaitu 2,025 m.
- Simulasi dilakukan pada Matlab R2014b

## BAB II TEORI PENUNJANG

### 2.1 Separator

Separator adalah suatu tabung bertekanan tinggi yang digunakan untuk memisahkan fluida produksi kedalam fasa liquid dan fasa gas. Prinsip kerja dari separator yaitu fluida yang berasal dari manifold akan masuk ke separator melalui lubang *inlet* dan selanjutnya akan menabrak *inlet diverter*. Prinsip gaya centrifugal digunakan dalam proses pemisahannya, gaya centrifugal inilah yang nantinya mendorong fluida ke dinding, sehingga liquid yang memiliki massa jenis lebih berat tertumbuk dan jatuh kebawah sementara gas yang memiliki massa jenis lebih ringan akan keatas. (Wang, 2000)



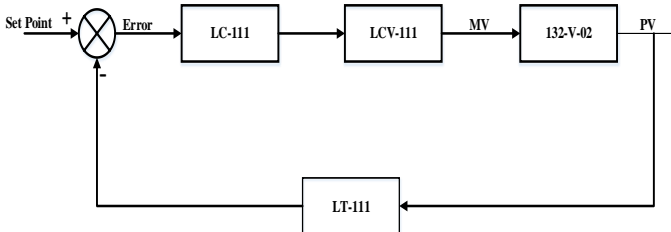
**Gambar 2.1** Separator 2 Fasa (Campbell J, 2014)

## 2.2 Pengendalian *Level* Pada Separator

Prinsip kerja pengendalian *level* pada separator bergantung pada berat jenis fluida. Tujuan pengendalian *level* ini adalah untuk menjaga *level* fluida yang terdapat di dalam separator agar fluida yang tercampur dapat di pisahkan.

Adapun blok diagram proses pengendalian *level* pada separator dapat di lihat pada gambar 2.2. Proses pengendalian *level* pada separator, gas yang bercampur dengan liquid masuk ke dalam separator adalah indikator pertama yang memberikan *input* terhadap sistem pengendalian. Banyak atau sedikitnya fluida yang masuk ke dalam separator menyebabkan *level* fluida yang terdapat di dalam berubah. Gas dan liquid yang masuk ke dalam separator di pisahkan berdasarkan massa jenisnya. Liquid yang memiliki massa jenis yang lebih besar dibandingkan dengan gas menyebabkan liquid berada pada bagian paling bawah separator, sehingga gas berada pada bagian atas liquid.

Berdasarkan PFD (*Process Flow Diagram*) dan P&ID (*Piping and Instrumentation Diagram*) pada *production separator* (132-V-02) terdiri dari beberapa instrument pada sistem pengendalian level yaitu *Level Indicator Controller* (LIC-111) dan *Level Transmitter* (LT-111), dan *Level Valve* (LV-111).

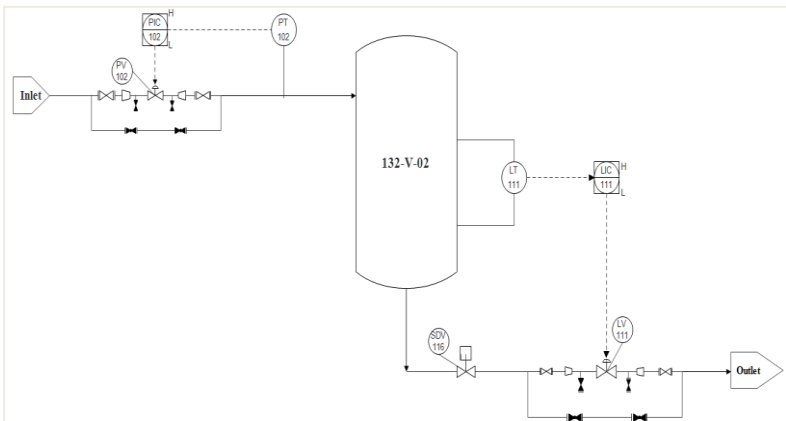


**Gambar 2.2** Diagram Blok Pengendalian *Level* Separator

LIC (*Level Indicator Controller*) berfungsi sebagai media penunjukan level yang masuk ke dalam separator dan juga berfungsi sebagai pemberi sinyal ke actuator (*control valve*) untuk melakukan *open* atau *close*. Pada *controller*, sinyal elektrik pada *transmitter* atau yang biasa disebut juga dengan *process variable*

*value* dibandingkan dengan nilai set point yang sudah ditetapkan. Selisih antara *process variable value* dengan set point disebut juga dengan *error*. *Error* inilah yang dijadikan sebagai input bagi controller. Sedangkan output yang dihasilkan dari aktuator (*control valve*) disebut *manipulated value* (mv).

Pengendalian level pada tangki separator dilakukan untuk menjaga level fluida tetap sesuai dengan nilai *set point*. Ketika *level* fluida yang dibaca lebih besar dari *set point* maka *control valve* akan membuka lebih besar untuk menjaga *level* pada tangki kembali ke *setpoint* dan sebaliknya jika *level* yang dibaca kurang dari set point maka *control valve* akan mengurangi bukaan atau menutup. Kondisi *control valve open* atau *close* terjadi sampai fluida kembali stabil sehingga kondisi *level* tetap terjaga pada nilai set point.



**Gambar 2.3** Instrument pengendali separator (P&ID separator Saka Pangkah Indonesia Limited)

### 2.3 Persamaan Dinamik Separator

Dengan hukum kesetimbangan massa, sistem tangki separator 2 fasa dapat didekati dengan :

$$\begin{bmatrix} \text{laju} \\ \text{perubahan} \\ \text{massa} \end{bmatrix} = \begin{bmatrix} \text{laju} \\ \text{massa} \\ \text{input} \end{bmatrix} - \begin{bmatrix} \text{laju} \\ \text{massa} \\ \text{output} \end{bmatrix} \quad (2.1)$$

Hukum kesetimbangan massa menyatakan bahwa jumlah massa yang masuk ke dalam sistem sebanding dengan jumlah massa yang keluar dari sistem serta massa yang terakumulasi dalam sistem itu sendiri. Dengan menggunakan hukum kesetimbangan massa maka model matematis separator dapat ditulis sebagai berikut :

$$\rho \frac{dv}{dt} = \sum \dot{m}_{in} - \sum \dot{m}_{out} \quad (2.2)$$

Dengan :

$$\begin{aligned} \rho \frac{dv}{dt} &: \text{laju perubahan massa separator} \\ \sum \dot{m}_{in} &: \text{total laju massa masukan} \\ \sum \dot{m}_{out} &: \text{total laju massa keluaran} \end{aligned}$$

Pada input separator berupa fasa campuran yang terdiri dari fasa gas dan fasa air. Kemudian kedua fasa tersebut dipisah berdasarkan densitasnya. Untuk menjaga *level* air dilakukan pengendalian agar air tidak tercampur dengan gas.

$$\text{Apabila } V = A \times h \quad (2.3)$$

Dan

$$\dot{m} = \rho_{liquid} \times Q \quad (2.4)$$

Maka



$$\rho_{liquid} A \frac{dh}{dt} = \sum \rho_{liquid} \cdot Q_{in} - \sum \rho_{liquid} \cdot Q_{out} \quad (2.5)$$

Karena nilai  $\rho_{liquid}$  memiliki nilai yang sama sehingga dapat dihilangkan, persamaan menjadi :

$$A \frac{dh}{dt} = (Q_{in} - Q_{out}) \quad (2.6)$$

Dengan,  $A, H, Q_{in}$  dan  $Q_{out}$  berturut-turut adalah luas penampang separator ( $m^2$ ), ketinggian / level (m), *flowrate liquid input* ( $m^3/s$ ) dan *flowrate liquid output* ( $m^3/s$ ).

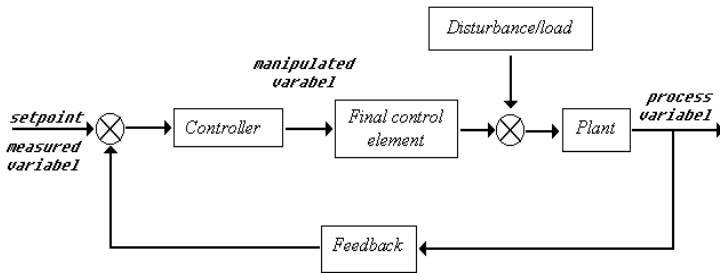
## 2.4 Sistem Kendali PI

Sistem pengendalian adalah gabungan dari kerja suatu komponen yang digunakan untuk mempertahankan variabel yang dikendalikan (*process variable*) pada suatu nilai tertentu (*set point*) sehingga sistem dapat dikatakan stabil. Seiring perkembangan ilmu pengetahuan dan teknologi suatu sistem pengendalian sudah memakai unit kendali otomatis seperti pada Gambar 2.4.

Algoritma pengendali *Proportional-Integral-Derivative* (PID) hingga saat ini dapat dikatakan sebagai algoritma pengendali terbaik karena keefektifannya, sederhana dalam implementasi dan luas penggunaannya. Karakteristik pengendali PID sangat dipengaruhi oleh kontribusi besaran dari ketiga parameter P, I dan D. Penyetelan konstanta  $K_p, T_i$  dan  $T_d$  akan mengakibatkan penonjolan sifat dari masing – masing elemen. Satu atau dua dari ketiga konstanta tersebut dapat diatur lebih menonjol dibandingkan dengan yang lain, konstanta yang menonjol itulah yang akan memberikan pengaruh pada respon sistem secara keseluruhan.

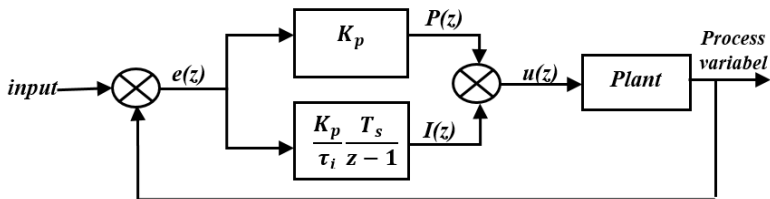
Pengendali PI merupakan penggabungan antara dua macam pengendali, yaitu P (*Propotional*) dan I (*Integral*). Pengendali PI

merupakan jenis pengendali untuk menentukan suatu kestabilan atau kepresisian sistem instrumentasi dengan adanya *feedback* atau umpan balik pada sistem tersebut. Perancangan sistem kendali PI bertujuan untuk mengetahui respon dari sistem jika ditambahkan dengan pengendali yang telah didapatkan nilai *gain*nya serta untuk mengetahui kesesuaian kerja sistem secara keseluruhan. Gambar 2.4 adalah diagram blok sistem pengendali PI.



**Gambar 2.4** Diagram blok sistem pengendalian tertutup (Basilio, 2002)

Pengendali P memiliki kelebihan untuk mempercepat *rise time* sehingga respon dari sistem cepat mencapai *set point*, tetapi pengendali P memiliki kekurangan yaitu meninggalkan *offset*. Pengendali I memiliki kelemahan yaitu menyebabkan lambatnya respon dari sistem, tetapi pengendali I dapat menghilangkan *offset* dan pengendali I ini mampu mengurangi terjadinya *maximum overshoot* yang besar.



**Gambar 2.5** Diagram blok pengendalian PI (Basilio, 2002)

Sifat pengendali P yang meninggalkan *offset* dapat dihilangkan oleh pengendali I, sedangkan sifat pengendali I yang menyebabkan lambatnya respon dapat ditutupi oleh pengendali P, sehingga pengendali PI menghasilkan respon yang lebih cepat dari pengendali I tetapi menghilangkan *offset* yang ditinggalkan pengendali P.

Persamaan pengendali PI adalah sebagai berikut :

$$u(z) = Kp \cdot e(z) + \frac{Ki \cdot Ts}{\tau_i(z - 1)} \quad (2.7)$$

Dengan :

$\tau_i$  : Konstanta waktu Integral

$Ts$  : Waktu cuplik (detik)

Beberapa hal yang perlu diperhatikan pada nilai dari  $Kp$  dan  $Ki$  adalah sebagai berikut :

- Nilai gain  $Kp$  yang terlalu besar akan mengakibatkan sistem menjadi semakin sensitif dan cenderung tidak stabil. Jika nilai  $Kp$  terlalu kecil maka akan menyebabkan *offset* yang besar.
- Nilai dari  $Ti$  yang kecil akan menghilangkan *offset* tetapi cenderung membuat sistem menjadi lebih sensitif atau mudah berosilasi sedangkan  $Ti$  yang besar belum tentu efektif menghilangkan *offset* dan cenderung membuat respon menjadi lambat. (Maulana, 2012)

## 2.5 Level Transmitter

Untuk pengukuran *level* separator yang saya amati ini menggunakan prinsip DP transmitter (*Differential Pressure transmitter*) untuk mengukur level di dalamnya. Pengukuran level jenis differential pressure (DP) didasarkan pada prinsip "*hydrostatic head*". Prinsip ini mengatakan bahwa pada setiap titik di dalam fluida yang diam (static), gaya yang bekerja padanya adalah sama untuk semua arah dan tidak tergantung pada volume fluida maupun bentuk ruang atau tempat dimana fluida berada, tetapi hanya bergantung pada tinggi kolom fluida di atas titik yang

bersangkutan. Oleh karena itu *hydrostatic head* sering dinyatakan dalam satuan tekanan

Prinsip pembacaan tekanan didalam tangki bergantung pada nilai *high pressure* (HP) dan *lowpressure* (LP) pada *tapping point* yang telah ditentukan, sehingga dapat disimpulkan bahwa DP *transmitter* mengukur tekanan hydrostatic dari fluida yang ada di dalam tangki dan *hydrostatichead* dinyatakan dalam tekanan dengan persamaan (2.8) dan (2.9).



**Gambar 2.6** LDT (*Differential Presure Transmitter*)  
(Rosemount, 2019)

$$\Delta P = P_{\text{high}} - P_{\text{low}} \quad (2.8)$$

$$\Delta P = \rho_1 \cdot g_1 \cdot h_1 - \rho_2 \cdot g_2 \cdot h_2 \quad (2.9)$$

Dimana :

$\Delta P$  : Tekanan Hidrostatik (mmH<sub>2</sub>O)

$\rho$  : Densitas fluida (kg/m<sup>3</sup>)

$g$  : Konstanta gravitasi (m/s<sup>2</sup>)

$h$  : Level fluida (m)

*Differential Pressure Transmitter* bekerja berdasarkan prinsip beda tekanan antara sisi *high* dan *low*. Karena tekanan merupakan parameter utama dari *transmitter*, tentu saja *density* yang bervariasi akan mempengaruhi pembacaan *transmitter* yang bekerja berdasarkan prinsip beda tekanan.

## 2.6 Control Valve

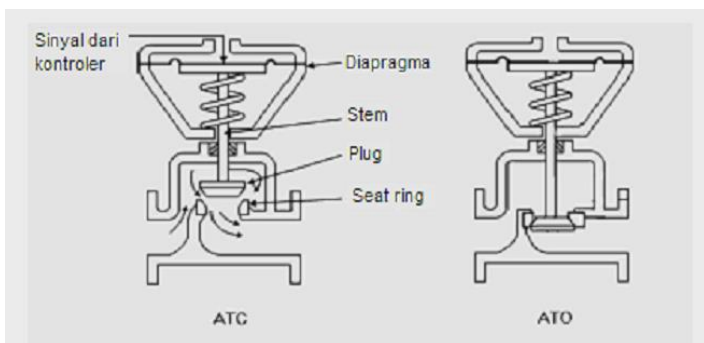
*Control valve* bertugas melakukan langkah koreksi terhadap variabel termanipulasi sebagai hasil akhir dari sistem pengendalian. *Control valve* merupakan salah satu elemen pengendali akhir (*final element control*) yang paling umum digunakan. Berdasarkan aksinya *control valve* dibedakan menjadi dua macam, yaitu : Aksi *control valve* terdiri dari dua macam yaitu:

a. ATO (*Air to Open*)

ATO adalah kondisi posisi *valve* yang secara normal terbuka (*open*) dan ketika mendapatkan sinyal kendali (4 – 20 mA atau 3 – 15 psi) posisi *valve* menutup secara proporsional dengan sinyal kendali tersebut.

b. ATC (*Air to Closed*)

ATC adalah kondisi posisi *valve* yang secara normal tertutup (*close*) dan ketika mendapatkan sinyal kendali (4 – 20 mA atau 3 – 15 psi) posisi *valve* membuka secara proporsional dengan sinyal kendali tersebut.



**Gambar 2.7** ATC dan ATO *control valve* (Prawito,K 2005)

## 2.7 Observer

*Observer* merupakan algoritma yang digunakan untuk mengestimasi variabel keadaan (*state*) berdasarkan model dinamik sistem. Pada umumnya, sistem dinamik dinyatakan dalam bentuk ruang keadaan (*state space*) yang dapat didefinisikan secara umum untuk sistem diskrit seperti di bawah ini (Ogata, 1987).

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (2.10)$$

$$y(t) = Cx(t) + Du(t) \quad (2.11)$$

dengan:

A	: Matriks keadaan ( <i>state</i> )
B	: Matriks masukan
C	: Matriks keluaran
D	: Matriks gangguan
$u(t)$	: Masukan kendali
$y(t)$	: Vektor keluaran
$\dot{x}(t)$	: Vektor keadaan ( <i>state</i> )

Jika di dalam sistem atau plant terjadi kesalahan, maka persamaan 2.10 dan 2.11 akan menjadi persamaan berikut : (Indriawati.Katherin,2015)

$$\dot{x}(t) = Ax(t) + Bu(t) + F_a f_a(t) \quad (2.11)$$

$$y(t) = Cx(t) + Du(t) + F_s f_s(t) \quad (2.12)$$

Dimana:

$F_a$  = matriks kesalahan aktuator

$F_s$  = matriks kesalahan sensor

$f_a$  = vektor kesalahan aktuator

$f_s$  = vector kesalahan sensor

Kemudian state baru  $\dot{z}(t)$  ditambahkan untuk memindahkan kesalahan aktuator dari persamaan keluaran ke persamaan

keadaan, persamaan tersebut dapat didefinisikan persamaan baru sebagai berikut:

$$\dot{z}(t) = Az(y - z) \quad (2.13)$$

$$\dot{z}(t) = Az(Cx(t) - z(t)) \quad (2.14)$$

Untuk  $Az$  merupakan matriks identitas ( $I$ ) untuk  $Az \in I^q$ , sedangkan matriks dari kesalahan sensor dan aktuator didefinisikan sebagai  $f = \begin{bmatrix} f_a \\ f_s \end{bmatrix}$

Dari persamaan di atas, maka dapat didefinisikan matriks yang akan digunakan untuk mendapatkan matriks yang sudah di ekspansi, sebagai berikut.

$$\begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix} = \begin{bmatrix} A & 0 \\ AZ \cdot C & -AZ \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} F_a & 0 \\ 0 & AZ \cdot F_s \end{bmatrix} \begin{bmatrix} f_a \\ f_s \end{bmatrix} \quad (2.15)$$

$$\dot{y}(t) = z(t) = [0 \ 1] \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} \quad (2.16)$$

dengan  $X_a = \begin{bmatrix} x \\ z \end{bmatrix}$

$$Aa = \begin{bmatrix} A & 0 \\ AZ \cdot C & -AZ \end{bmatrix}, Ba = \begin{bmatrix} B \\ 0 \end{bmatrix}, Ca = [0 \ 1], Ea = \begin{bmatrix} F_a & 0 \\ 0 & AZ \cdot F_s \end{bmatrix}$$

Dari persamaan *state space* (2.15) dan (2.16), persamaan tersebut dapat didefinisikan dalam bentuk *standard state space* sebagai berikut.

$$\begin{aligned} \dot{X}_a(t) &= A_a X_a(t) + B_a u(t) + E_a f(t) \\ \dot{y}(t) &= C_a X_a(t) \end{aligned} \quad (2.17)$$

Untuk memperoleh nilai dari *gain observer* maka dipakai ekspansi matrik melalui persamaan ruang keadaan baru yang ditunjukkan pada persamaan berikut.

$$\begin{aligned}\dot{\hat{X}}_a(t) &= A_a \hat{X}_a + B_a u + E_a \hat{f}_s + L_x (Y - \hat{Y}) \\ \dot{\hat{f}}_s &= L_f (Y - \hat{Y}) \\ \dot{\hat{Y}} &= C_a \hat{X}_a\end{aligned}\quad (2.18)$$

Dimana  $\hat{X}$ ,  $\hat{f}$ ,  $\hat{Y}$  merupakan *state estimate*, *fault estimate* dan *output estimate*.  $L_x$  adalah *gain propotional* dari *observer* dan  $L_f$  adalah *gain integral* dari *observer*. Sehingga dari persamaan diatas akan menjadi sebagai berikut.

$$\begin{bmatrix} \dot{\hat{X}}_a(t) \\ \dot{\hat{f}}_s(t) \end{bmatrix} = \begin{bmatrix} A_a & E_a \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{X}_a(t) \\ \hat{f}_s(t) \end{bmatrix} + \begin{bmatrix} B_a \\ 0 \end{bmatrix} u + \begin{bmatrix} L_x \\ L_f \end{bmatrix} [C_a \quad 0] (Y - \hat{Y}) \quad (2.19)$$

$$\dot{\hat{y}} = [C_a \quad 0] \begin{bmatrix} \hat{x} \\ \hat{f} \end{bmatrix} \quad (2.20)$$

Dari bentuk (2.19) dan (2.20) dapat diubah menjadi persamaan keadaan dan persamaan keluaran *observer*, sebagai berikut.

$$\dot{\hat{x}}(t) = \tilde{A}_a \hat{x}(t) + \tilde{B}_a u(t) + \tilde{K}_e (\tilde{y}(t) - \hat{y}(t)) \quad (2.21)$$

$$\tilde{y}(t) = \tilde{C}_a \hat{x}(t) \quad (2.22)$$

dengan  $\hat{x} = \begin{bmatrix} \hat{X}_a \\ \hat{f}_a \end{bmatrix}$

$$\tilde{A}_a = \begin{bmatrix} A_a & E_a \\ 0 & 0 \end{bmatrix}, \quad \tilde{C}_a = [C_a \quad 0]$$

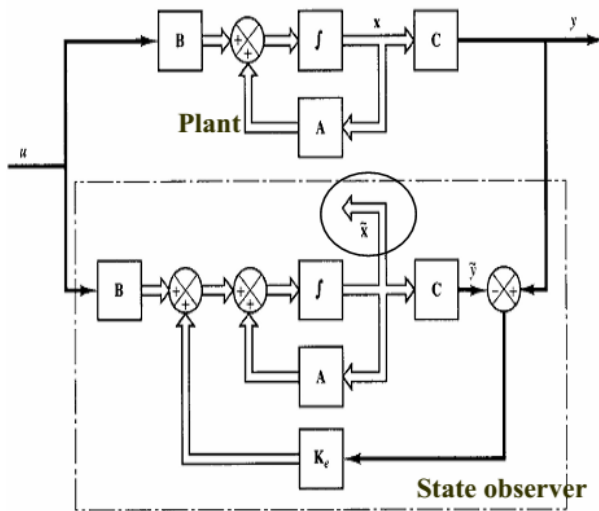
Dimana:

$\hat{x}$  = menunjukkan state estimasi dari  $x(t)$

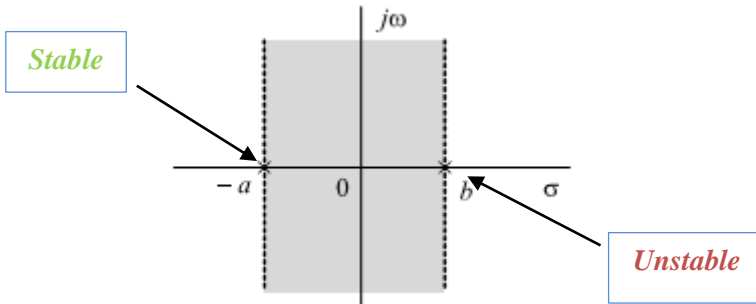
$K_e$  = gain matris *observer*



Berdasarkan Gambar 2.8, *observer* mengandung nilai *gain observer* atau  $K_e$ . *Gain observer* merupakan matriks untuk mengoreksi perbedaan antara keluaran  $y$  aktual dan  $y$  estimasi sehingga dapat digunakan untuk mengestimasi *state* yang tidak terukur (Ogata, 2004). Untuk menentukan nilai  $K_e$  sangat dipengaruhi oleh nilai letak *pole*, seperti pada Gambar 2.9 untuk sistem dengan domain kontinu, posisi nilai *pole* untuk menyatakan sistem stabil berada di sebelah kiri sumbu imajiner, sedangkan sistem tidak stabil berada di sebelah kanan sumbu imajiner.



**Gambar 2.8** Observer (Ogata K. , 1987)



**Gambar 2.9** Kestabilan *pole placement* (Yuriy, S,2007)

## 2.8 Algoritma *exponential smoothing*

Kesalahan yang dimodelkan sebagai *output* aditif dari sistem separator merupakan kesalahan aktuator. Dengan merancang *observer* maka dapat menghasilkan estimasi kesalahan aktuator. Estimasi kesalahan aktuator diasumsikan linier dari indeks waktu diskrit  $n$ . Metode *Holt's* diadopsi untuk memproses prediksi kesalahan. Algoritma prediksi kesalahan dapat dihitung melalui *exponential smoothing algorithm* sebagai berikut (Xu et al., 2009):

$$u_n = a\hat{a}_n + (1 - a)(u_{n-1} + s_{n-1}) \quad (2.22)$$

$$s_n = b(u_n - u_{n-1}) + (1 - b)s_{n-1} \quad (2.23)$$

Misalkan  $u_n$  adalah hasil estimasi yang diperhalus dan  $s_n$  adalah faktor pertumbuhan linear yang diestimasi. Diberikan hasil estimasi kesalahan  $\hat{a}_n$  ( $n = 1, 2, \dots$ ) yang diperoleh dari *observer*. Dimana  $0 < a < 1$  &  $0 < b < 1$  adalah parameter *Holt's method* (Xu et al., 2009). Kondisi awal untuk *Holt's method* sebagai berikut:

$$u_1 = \hat{a}_1 = \hat{f}(k) \quad (2.24)$$

$$u_2 = \hat{a}_2 = \hat{f}(k + 1) \quad (2.25)$$

$$s_2 = u_2 - u_1 \quad (2.26)$$

Setelah menghitung estimasi kesalahan pada saat  $nT$  dan estimasi faktor pertumbuhan, maka prediksi kesalahan dapat dihitung menggunakan algoritma berikut (Xu et al., 2009):

$$\bar{\alpha}_{n+l} = u_n + s_n lT \quad (2.27)$$

Dengan

$l$  : step prediksi

$T$ : *time interval* untuk prediksi kesalahan

Penjelasan mengenai algoritma prediksi kesalahan adalah sebagai berikut :

- 1) Step estimasi kesalahan yang diperhalus: Jika  $n = 1$ , atau  $n = 2$ , menghitung estimasi kesalahan yang diperhalus  $u_n$  menggunakan persamaan (2.24), atau (2.25). Jika  $n \geq 3$  menghitung estimasi kesalahan yang diperhalus  $u_n$  menggunakan persamaan (2.22).
- 2) Step estimasi faktor pertumbuhan: Menghitung  $n$ -step prediksi faktor pertumbuhan  $s_n$  dengan menggunakan (2.25) jika  $n = 2$ , dan menghitung  $n$ -step estimasi faktor pertumbuhan  $s_n$  dengan menggunakan (2.23) jika  $n \geq 3$ .
- 3) Step prediksi kesalahan: Menghitung prediksi kesalahan  $\bar{\alpha}_{n+l}$  dengan menggunakan persamaan (2.27).

## 2.9 Keandalan

Keandalan merupakan kemungkinan dari suatu komponen atau sistem untuk dapat beroperasi atau melaksanakan fungsinya (Ebeling, 1997). Fungsi tersebut telah ditetapkan pada kondisi pengoperasian tertentu dan dalam lingkungan tertentu untuk periode waktu yang telah ditentukan pula. Jadi, keandalan adalah kemungkinan untuk tidak mengalami kegagalan atau dapat melaksanakan fungsinya selama periode waktu ( $t$ ) atau lebih. Keandalan komponen, subsistem atau sistem produksi merupakan salah satu aspek yang dapat mempengaruhi kesuksesan proses produksi. Evaluasi keandalan dapat membantu dalam memperkirakan peluang sebuah sistem atau komponen untuk dapat

melaksanakan fungsinya dalam jangka waktu tertentu dan dalam kondisi tertentu pula. Sistem atau komponen masih dapat beroperasi walaupun tidak dapat memenuhi fungsinya secara maksimal (Dhillion, 2005).

Nilai fungsi *reliability* merupakan nilai probabilitas, maka nilai fungsi *reliability* ( $R$ ) bernilai  $0 \leq R \leq 1$  (Ebeling, 1997). Kegagalan pada separator terjadi ketika fungsi yang dimaksudkan dari komponen separator tidak dapat bekerja dengan baik. Keandalan sistem pengendalian level dipengaruhi oleh keandalan sub-sistemnya, termasuk perangkat keras dan perangkat lunak, dan bagaimana komponen-komponen tersebut terhubung.

## 2.10 Prediksi keandalan *real-Time*

Penelitian mengenai keandalan *real-time* atau yang biasanya disebut juga dengan keandalan *on-line* ini mulai berkembang sejak tahun 1990-an. Pada tahun 1992, Kim dan rekannya meneliti mengenai prediksi kendalan *real-time* kondisional berdasarkan data kinerja alat. Sedangkan pada 1999, Chinnman melakukan penelitian mengenai estimasi keandalan *on-line* dari masing-masing komponen menggunakan sinyal degradasi.

Penelitian terus berkembang pada tahun 2000-an. Pada 2002, Chinnman kembali melanjutkan penelitiannya dengan mengambil topik berupa estimasi keandalan *on-line* dari sistem fisik menggunakan jaringan saraf. Penelitian tersebut terus berkembang hingga pada tahun 2009, Xu dan rekannya memperkenalkan penetiannya mengenai sebuah metode baru dalam memonitoring keandalan *real-time* untuk sistem dinamik yang digabungkan dengan algoritma estimasi kesalahan

Setelah kesalahan dimonitoring, keandalan pada sistem dinamik untuk beberapa waktu kedepan dapat dimonitoring. Untuk memonitoring keandalan sistem, ada beberapa hal penting yang harus diperhatikan. Pertama, selain dari definisi kegagalan sebagai nilai kesalahan yang telah mencapai batas, kegagalan sistem juga didefinisikan sebagai keadaan sistem yang telah memasuki area terlarang  $\mathcal{M}_1$ .

Secara umum, proses kesalahan yang sama memiliki dampak yang berbeda pada sifat sistem untuk sistem yang berbeda. Keadaan sistem merupakan hal yang krusial, dikarenakan berhubungan langsung dengan performansi atau keamanan dari sebuah sistem. Kedua, keadaan sistem tidak dapat diukur secara langsung, harus melalui estimasi dikarenakan kegagalan sistem didefinisikan dalam sebuah keadaan. Keadaan diestimasi menggunakan algoritma OTK-SIR. Pada tugas akhir ini, penulis tidak menggunakan algoritma OTK-SIR, namun menggunakan algoritma *observer*. Ketiga, kunci dari prediksi keandalan adalah prediksi kesalahan (2.27), sehingga algoritma RP (*Reliability Prediction*) yang berdasar pada algoritma *observer* dan prediksi kesalahan.

$$R(nT, (n+r)T) = \Pr(x_{n+r} \in \mathcal{M}_0 | x_n \in \mathcal{M}_0) \quad (2.28)$$

Dimana :

$R(nT, (n+r)T)$  : RP (*Reliability Prediction*)  
 $T$  : *Sampling Interval*  
 $\mathcal{M}_0$  : Batas nilai variable

Penjelasan mengenai algoritma RP (*Reability Prediction*) adalah sebagai berikut :

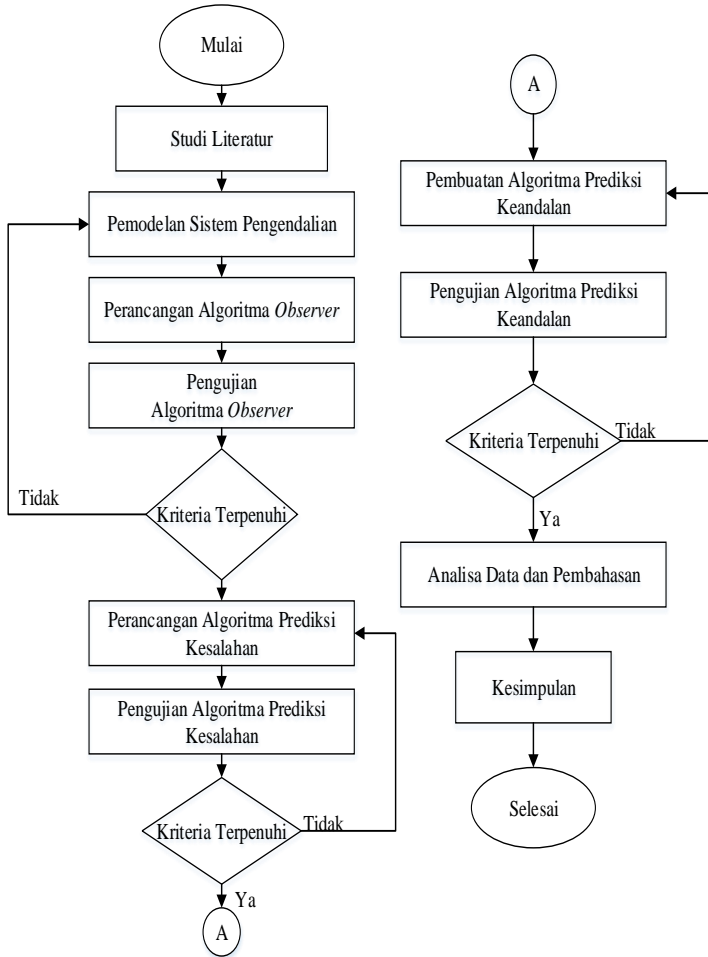
- 1) Step inialisasi :  $l$  yang telah dinotasikan pada algoritma prediksi kesalahan dijadikan  $l = 1$ . Inialisasi partikel menjadi  $\bar{X}_{n+l-1|n}^{(i)} = \bar{X}_{n+0|n}^{(i)} = \hat{X}_{n|n}^{(i)}$ ,  $i = 1, \dots, N$ .
- 2) Step estimasi kesalahan : Jalankan algoritma *observer* untuk menghasilkan estimasi kesalahan  $\hat{a}_1$  dan  $\hat{a}_2$  pada persamaan (2.21) dan (2.22).
- 3) Step prediksi kesalahan: Jalankan algoritma prediksi kesalahan dengan  $l$  langkah untuk menghitung prediksi kesalahan  $\tilde{a}_{n+l}$ .
- 4) Step prediksi partikel : untuk  $i = 1, \dots, N$ , hitung partikel baru  $\bar{X}_{n+l-1|n}^{(i)}$  dari sistem persamaan  $\bar{X}_{n+l-1|n}^{(i)} = f(\bar{X}_{n+l-1|n}^{(i)}) + \Gamma \tilde{a}_{n+l} + W_{n+l}^{(i)}$  dimana  $W_{n+l}^{(i)}$  adalah realisasi dari  $w_{n+l}$  untuk partikel ke- $i$ .

- 5) Step pemilihan partikel : jika  $l < r$ , maka  $l = l + 1$ , dan kembali ke step 2, jika sebaliknya lanjut pada step 6.

Step perhitungan prediksi keandalan : hitung jumlah partikel dari himpunan  $\{\bar{X}_{n+r|n}^{(i)}\} (i = 1, \dots, N)$  yang jatuh pada  $\mathcal{M}_0$  dan dinotasikan oleh  $N_0$ . Maka sistem prediksi keandalan selama  $(nT, (n+r)T|$  dihitung oleh  $N_0/N$

### BAB III METODOLOGI PENELITIAN

Prosedur penelitian secara garis besar digambarkan dalam diagram alir di bawah ini :



**Gambar 3.1** Diagram alir penelitian tugas akhir

### 3.1 Studi literatur

Studi literatur digunakan untuk membangun pemahaman awal tentang metode konsep, serta teori yang menunjang dalam pelaksanaan penelitian. Materi yang digunakan untuk menunjang penelitian antara lain yaitu, antara lain pemahaman mengenai sistem separator, kendali PID, *observer*, algoritma prediksi kesalahan, algoritma prediksi keandalan dan mekanisme perancangan sistem prediksi keandalan pada sistem pengendalian *level* separator secara *real-time* berdasarkan prediksi kesalahan.

### 3.2 Data Pemodelan *Plant*

Data pemodelan *plant* berdasarkan data spesifikasi separator, sensor, dan aktuator meliputi data besaran input dan outputnya. Pengambilan data dilakukan di PT.SIPL (Saka Indonesia Pangkah Limited). Data yang digunakan berupa data spesifikasi separator yang ditunjukkan dalam tabel 3.1.

**Tabel 3. 1** Data Spesifikasi Separator

Notasi	Keterangan	Nilai dan satuan
$H_{\max}$	Level maksimum	2,025 m
$H_{\min}$	Level minimum	0,6 m
$Q_{in}$	Flowrate input liquid	0,001835 m <sup>3</sup> /s
$\rho_{liquid}$	Massa jenis liquid	631,87 Kg/m <sup>3</sup>
D	Diameter Separator	1,22 m
L	Tinggi Tangki	4,978m

### 3.3 Pemodelan Matematis Sistem

Pemodelan matematis sistem diperlukan dalam merancang simulasi suatu sistem, sehingga kita bisa mengetahui respon sistem dari simulasi tersebut. Pemodelan matematis sistem pengendalian *level* separator ini meliputi pemodelan matematis separator (*plant*), pemodelan matematis aktuator, dan pemodelan matematis *level transmitter* atau sensor.

#### 3.3.1 Pemodelan Matematis Separator

Untuk memodelkan separator digunakan hukum kesetimbangan massa. Hukum ini menyatakan bahwa jumlah



massa yang masuk ke dalam sistem sebanding dengan jumlah massa yang keluar dari sistem serta massa yang terakumulasi dalam sistem itu sendiri. Dengan menggunakan persamaan hukum kesetimbangan massa, maka proses pada Separator dapat dimodelkan sebagai berikut:

$$\begin{bmatrix} \text{laju} \\ \text{perubahan} \\ \text{massa} \end{bmatrix} = \begin{bmatrix} \text{Laju} \\ \text{massa} \\ \text{input} \end{bmatrix} - \begin{bmatrix} \text{Laju} \\ \text{massa} \\ \text{output} \end{bmatrix}$$

Dengan menggunakan hukum kesetimbangan massa maka model matematis *separator* dapat ditulis sebagai berikut:

$$\rho \frac{dv}{dt} = \sum \dot{m}_{in} - \sum \dot{m}_{out} \quad (3.1)$$

Dengan :

$\rho \frac{dv}{dt}$  : laju perubahan massa separator

$\sum \dot{m}_{in}$  : total laju massa masukan

$\sum \dot{m}_{out}$  : total laju massa keluaran

Apabila  $V = A \times h$  dan  $\dot{m} = \rho_{liquid} \times Q$  maka :

$$\rho_{liquid} A \frac{dh}{dt} = \sum \rho_{liquid} \cdot Q_{in} - \sum \rho_{liquid} \cdot Q_{out} \quad (3.2)$$

Karena nilai  $\rho_{liquid}$  memiliki nilai yang sama sehingga dapat dihilangkan, persamaan menjadi :

$$A \frac{dh}{dt} = (Q_{in} - Q_{out}) \quad (3.3)$$

Dari persamaan (3.3) didapatkan model matematis separator sebagai berikut:

$$\frac{dV_L}{dt} = Q_{Lin} - Q_{Lout} \quad (3.4)$$

Untuk mencari nilai  $Q_{Lout}$  didapatkan dari persamaan *pressure drop* sebagai berikut:

$$P - P_{Lout} = \frac{C_L \rho_L Q_{Lout}^2 - \rho_L g H}{g_c} + \Delta P_{Lc_v} \quad (3.5)$$

Sehingga,

$$Q_{Lout} = (0,002228) C_v \sqrt{\frac{\Delta P_{Lc_v}}{\gamma_L}} \quad (3.6)$$

Dimana :

$\Delta P_{Lc_v}$  : Tekanan fluida yang keluar

$\gamma_L$  : Specific Gravity

Untuk mendapatkan *state level* dari sistem separator, maka nilai spesifikasi terukur dari diameter separator dimasukkan kedalam persamaan (2.6) sebagai berikut :

$$A \frac{dh}{dt} = Q_{in} - Q_{out} \quad (3.7)$$

$$\frac{dh}{dt} = \frac{4}{\pi d^2} (Q_{in} - Q_{out}) \quad (3.8)$$

$$\frac{dh}{dt} = \frac{4}{\pi 1,22^2} (Q_{in} - Q_{out}) \quad (3.9)$$

$$\frac{dh}{dt} = 0,856 (Q_{in} - Q_{out}) \quad (3.10)$$

### 3.3.2 Pemodelan Matematis Aktuator

Aktuator yang digunakan adalah *Level Valve*. *Level* yang digunakan dilengkapi dengan *I/P converter* yang berfungsi untuk mengubah sinyal input 4-20 mA menjadi sinyal *pneumatic* 3-15 psi. Dalam hal ini, *Level Control Valve* memiliki masukan sinyal berupa arus listrik kemudian diubah menjadi tekanan untuk menggerakkan *control valve*.

Pemodelan aktuator yang digunakan untuk mengubah arus 4-20 mA menjadi tekanan 3-15 psi yang akan menggerakkan aktuator dengan nilai H maksimum sebesar 2,025 m dan H minimum 0,6 m adalah sebagai berikut :

$$\frac{\Delta P_v}{\Delta I} = \frac{P_{vmax} - P_{vmin}}{Arus_{max} - Arus_{min}} \quad (3.11)$$

$$\frac{\Delta P_v}{\Delta I} = \frac{15 - 3}{20 - 4} = 0,75 \quad (3.12)$$

Transfer function dari *pneumatic line delay* adalah sebagai berikut:

$$\frac{\Delta p_c}{\Delta p_v} = \frac{1}{\tau_0 s + 1} \quad (3.13)$$

$$\frac{\Delta p_c}{\Delta p_v} = \frac{1}{0,2s + 1} \quad (3.14)$$

$\tau_0$  adalah onstanta waktu aktuator , dengan nilai sebesar 0,2 sekon.(Wang, 2000)

Untuk mendapatkan *state pneumatic line* dari sistem separator, maka nilai *pline* 0,75 dari persamaan(3.12) dikalikan dengan persamaan (3.14) sehingga persamaannya menjadi sebagai berikut :

$$\frac{Pline(s)}{u(s)} = \frac{0,75}{0,2s + 1} \quad (3.15)$$

$$(0,2 s + 1)Pline = 0,75 u(s) \quad (3.16)$$

$$0,2 Pline(s) = 0,75 u(s) - Pline \quad (3.17)$$

$$Pline(s) = \frac{0,75}{0,2} u(t) - \frac{1}{0,2} Pline \quad (3.18)$$

$$Pline(s) = 3,75 u(s) - 5 Pline \quad (3.19)$$

*Transfer function* dari respon *control valve* adalah sebagai berikut:

$$\frac{\Delta x (s)}{\Delta P(s)} = \frac{100}{\frac{P_{\max} - P_{\min}}{c_0 s + 1}} \quad (3.20)$$

$$\begin{aligned} \frac{\Delta x (s)}{\Delta P(s)} &= \frac{100}{\frac{15 - 3}{2s + 1}} \\ &= \frac{25}{\frac{3}{2s + 1}} \end{aligned}$$

$c_0$  adalah konstanta waktu *control valve* , dengan nilai 2 sekon.(Wang, 2000)

Dari persamaan (3.20) maka didapatkan *state respon control valve* sebagai berikut :

$$\frac{\Delta x}{Pline} = \frac{8,33}{2s + 1} \quad (3.21)$$

$$2\Delta x(s) + \Delta x = 8,33 Pline \quad (3.22)$$

$$\Delta x(s) = \frac{8,33}{2} Pline - \frac{1}{2} \Delta x \quad (3.23)$$

$$\Delta x(s) = 4,165 \text{ Pline} - 0,5 \Delta x \quad (3.24)$$

### 3.3.3 Pemodelan Matematis *Level Transmitter*

Sensor *level* yang digunakan adalah *Differential Pressure Transmitter*, yaitu alat yang berfungsi untuk mengukur tekanan didalam separator dan memanfaatkan perbedaan tekanan untuk mengukur *level* didalam separator. *Transmitter* ini mengubah input berupa *level* menjadi arus dengan persamaan sebagai berikut :

$$\frac{I(s)}{H(s)} = \frac{I_{\max} - I_{\min}}{H_{\max} - H_{\min}} \quad (3.25)$$

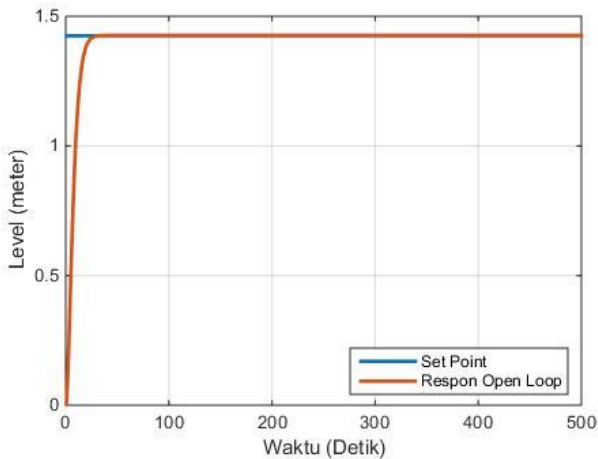
Dimana nilai H maksimum sebesar 2,765 m dan H minimum sebesar 0,6 m, I maksimum sebesar 20 mA dan I minimum sebesar 4 mA sehingga :

$$\frac{I(s)}{H(s)} = \frac{20 - 4}{2,025 - 0,6}$$

$$\frac{I(s)}{H(s)} = 11,22$$

### 3.4 Uji *Open Loop Plant*

Setelah pemodelan matematis di dapatkan, selanjutnya model disimulasikan dengan menggunakan software Matlab (R2014b) untuk dilakukan validasi dengan data hasil pengukuran yang dimiliki oleh PT. Saka Indonesia Pangkah Limited. Hasil Grafik respon *uji open loop* dapat dilihat pada gambar 3.2 dimana pada grafik tersebut dapat diketahui bahwa sistem dikatakan tervalidasi ketika nilai variabel yang terukur sudah sama dengan set point.

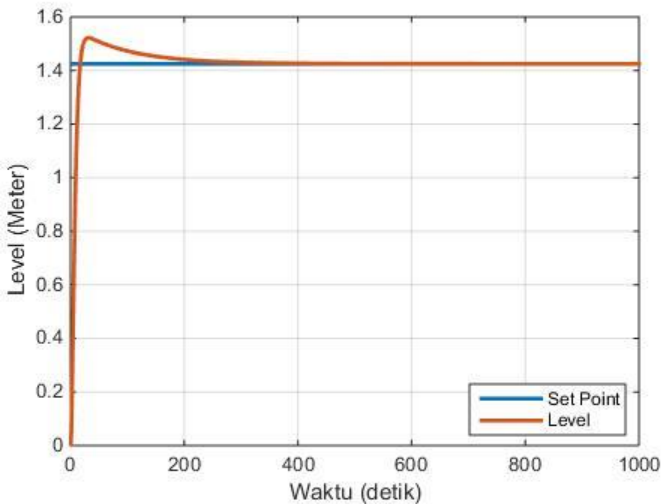


**Gambar 3.2** Respon *open loop*

Berdasarkan gambar tersebut bahwa grafik merupakan sistem berorde satu karena pada saat terjadi penambahan input, maka secara otomatis *level* juga akan bertambah.

### 3.5 Perancangan sistem kendali PI

Pengendalian level pada separator menggunakan pengendalian *feedback* dengan algoritma PI. Pada pengendali jenis *feedback* digunakan *Level Transmitter* untuk mengukur nilai proses yang dikendalikan sebagai pembanding dengan *set point*. Untuk menentukan gain pengendali PI menggunakan metode *trial and error* hingga memenuhi kriteria  $error\ steady\ state \leq 2\%$ . Pada sistem ini didapatkan bahwa nilai  $K_p$  sebesar 1 dan  $T_i$  sebesar 0.01. Setelah itu sistem disimulasikan dengan menggunakan *software* Matlab (2014b) dengan respon seperti gambar 3.3



**Gambar 3.3** Hasil respon *plant*

Berdasarkan respon sistem tersebut dapat diketahui karakteristik respon sistem yakni *rise time* sebesar 10,9 detik, *settling time* sebesar 147 detik dan *maximum overshoot* sebesar 6,71%, dan error steady state sebesar 0,14%

### 3.6 Perancangan *Observer*

Perancangan *observer* dilakukan untuk mengestimasi kesalahan yang ada dengan cara mengembangkan persamaan ruang yang ada, dengan cara mengembangkan persamaan ruang keadaan sistem. Model *state space* dari sistem pengendalian *level* separator terdiri dari 3 *state* yaitu *pneumatic line delay*, respon control valve ( $\Delta x$ ) dan *level*. Persamaan (3.19), (3.24) dan (3.10) merupakan persamaan dari *state pneumatic line delay*, respon control valve ( $\Delta x$ ) dan *level*, kemudian ketiga persamaan tersebut diubah kedalam domain waktu.

$$\frac{dP_{\text{pneumatic\_line}}(t)}{dt} = -5P_{\text{pneumatic\_line}}(t) + 3,75u(t) \quad (3.26)$$

$$\frac{d\Delta x(t)}{dt} = 4,165 P_{\text{neumatic}_{line}}(t) - 0,5\Delta x(t) \quad (3.27)$$

$$\frac{dh(t)}{dt} = 0,856Q_{in} - 0,856Q_{out} \quad (3.28)$$

Ketiga persamaan diatas kemudian dibuat kedalam bentuk matriks ruang keadaan mengacu pada persamaan (2.10) seperti berikut :

$$\begin{bmatrix} \frac{dP_{\text{neumatic}_{line}}}{dt} \\ \frac{d\Delta x}{dt} \\ \frac{dh}{dt} \end{bmatrix} = \begin{bmatrix} -5 & 0 & 0 \\ 4,165 & -0,5 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_{line} \\ \Delta x \\ h \end{bmatrix} \quad (3.29)$$

$$+ \begin{bmatrix} 3,75 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0,856 & -0,856 \end{bmatrix} \begin{bmatrix} u \\ Q_{in} \\ Q_{out} \end{bmatrix}$$

Persamaan matriks output didapatkan berdasarkan persamaan (2.11)

$$\begin{bmatrix} \frac{d\Delta x}{dt} \\ \frac{dh}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_{line} \\ \Delta x \\ h \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ Q_{in} \\ Q_{out} \end{bmatrix} \quad (3.30)$$

Dari bentuk matriks diatas didapatkan nilai-nilai sebagai berikut :

Matriks A adalah matriks yang menyatakan *state* sistem. Berdasarkan penjabaran dari fungsi transfer aktuator, maka matriks A bernilai:

$$A = \begin{bmatrix} -5 & 0 & 0 \\ 4,165 & -0,5 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$



Matriks B merupakan matriks yang menyatakan input kontrol sistem. Untuk mencari nilai matriks B didapatkan dari fungsi transfer aktuator dan plant, sehingga matriks B bernilai:

$$B = \begin{bmatrix} 3,75 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0,856 & -0,856 \end{bmatrix}$$

Matriks C merupakan matriks output sistem, sehingga matriks C bernilai:

$$C = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Matriks D merupakan matriks yang menyatakan *noise* sistem, jika diasumsikan tidak terjadi *noise* maka matriks D bernilai:

$$D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Matriks kesalahan terdapat dua macam, yakni kesalahan aktuator ( $F_a$ ) dan sensor ( $F_s$ ). Dalam tugas akhir ini hanya dianalisa kesalahan aktuator, sehingga matriks  $F_a$  bernilai:

$$F_a = \begin{bmatrix} 3,75 \\ 0 \\ 0 \end{bmatrix}$$

Untuk mengkompensasi kesalahan aktuator maka diperlukan modifikasi algoritma *observer* baru, seperti pada persamaan berikut ini.

$$\dot{x}(t) = Ax(t) + Bu(t) + F_a f_a(t) \quad (3.31)$$

$$y(t) = Cx(t) + Du(t) \quad (3.32)$$

Kemudian *state* baru  $z(t)$  ditambahkan untuk memindahkan kesalahan aktuator dari persamaan keluaran ke persamaan keadaan, persamaan tersebut dapat didefinisikan persamaan baru pada persamaan (3.33) – (3.35) berikut.

$$\dot{z}(t) = Az(y(t) - z(t)) \quad (3.33)$$

$$\dot{z}(t) = Az(Cx(t) - z(t)) \quad (3.34)$$

$$\dot{z}(t) = Az.Cx(t) - Az.z(t) \quad (3.35)$$

Untuk  $Az$  merupakan matriks identitas ( $I$ ),  $Az \in Iq$ . Agar dapat mendefinisikan matriks kesalahan aktuator, maka bentuk dari matriks ruang keadaan dapat diekspansi menjadi seperti berikut.

$$\begin{aligned} \begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix} &= \begin{bmatrix} A & 0 \\ Az \cdot C & -Az \end{bmatrix} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) \\ &+ \begin{bmatrix} F_a \\ 0 \end{bmatrix} f_a(t) \end{aligned} \quad (3.36)$$

$$\dot{y}(t) = z(t) = [0 \ 1] \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} \quad (3.37)$$

dengan  $X_a = \begin{bmatrix} x \\ z \end{bmatrix}$ , Sehingga didapatkan

$$\begin{aligned} Aa &= \begin{bmatrix} A & 0 \\ Az \cdot C & -Az \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 4,165 & -0,5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{bmatrix} \\ Ba &= \begin{bmatrix} B \\ 0 \end{bmatrix} = \begin{bmatrix} 3,75 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -0,0155 & 0,0155 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ Ca &= \begin{bmatrix} 0 \\ C \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

$$Ea = \begin{bmatrix} Fa \\ 0 \end{bmatrix} = \begin{bmatrix} 3,75 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Dari persamaan *state space* (3.36) dan (3.37), persamaan tersebut dapat didefinisikan persamaan baru yang sudah di ekspansi sebagai berikut.

$$\begin{aligned} \dot{X}_a(t) &= A_a x_a(t) + B_a u(t) + E_a f_a(t) \\ \dot{y}(t) &= C_a x_a(t) \end{aligned} \quad (3.38)$$

Untuk memperoleh nilai *gain observer*, maka ditambahkan *state* yang mengandung *gain* agar dapat mengestimasi kesalahan aktuator, sehingga ekspansi matriks ruang keadaan baru yang ditunjukkan pada persamaan berikut.

$$\begin{aligned} \dot{\hat{X}}_a(t) &= A_a \hat{X}_a + B_a u + E_a \hat{f}_a + L_x (Y - \hat{Y}) \\ \hat{f}_a &= L_f (Y - \hat{Y}) \\ \dot{\hat{Y}} &= C_a \hat{X}_a \end{aligned} \quad (3.39)$$

Sehingga dari persamaan diatas jika diubah kedalam bentuk yang lebih ringkas akan menjadi sebagai berikut.

$$\begin{bmatrix} \dot{\hat{X}}_a(t) \\ \hat{f}_a(t) \end{bmatrix} = \begin{bmatrix} A_a & E_a \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{X}_a \\ \hat{f}_a \end{bmatrix} + \begin{bmatrix} B_a \\ 0 \end{bmatrix} u + \begin{bmatrix} L_x \\ L_f \end{bmatrix} [C_a \quad 0] (Y - \hat{Y}) \quad (3.40)$$

$$\dot{\hat{Y}} = [C_a \quad 0] \begin{bmatrix} \hat{X}_a \\ \hat{f}_a \end{bmatrix} \quad (3.41)$$

Dengan mensubstitusikan nilai matriks  $A_a$ ,  $C_a$  dan  $E_a$ , diperoleh nilai matriks  $\tilde{A}_a$  dan  $\tilde{C}_a$  sebagai berikut.

$$\tilde{A}_a = \begin{bmatrix} A & E_a \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} -5 & 0 & 0 & 0 & 0 & 3,75 \\ 4,165 & -0,5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{C}_a = [Ca \quad 0] = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Dari bentuk (3.40) dan (3.41) dapat diubah menjadi persamaan keadaan dan persamaan keluaran *observer*, sebagai berikut.

$$\dot{\hat{x}}(t) = \tilde{A}_a \hat{x}(t) + \tilde{B}_a u(t) + \tilde{K}_e (\check{y}(t) - \hat{y}(t)) \quad (3.42)$$

$$\hat{y}(t) = \tilde{C}_a \hat{x}(t) \quad (3.43)$$

dengan  $\hat{x} = \begin{bmatrix} \hat{X}_a \\ \hat{f}_a \end{bmatrix}$

Langkah selanjutnya, Untuk memperoleh nilai *gain observer* ( $\tilde{K}_e$ ) Ktild digunakan metode *pole-placement*. Nilai pole yang diinginkan ditentukan dengan metode *trial-and-error*, dengan berdasarkan ilmu kestabilan berdasarkan letak *pole*, diketahui bahwa sebuah sistem diskrit didefinisikan stabil jika seluruh akar-akar karakteristik bagian riil atau imajiner bernilai tidak lebih dari 1 atau -1. Nilai *pole-placement* yang dipakai adalah

$$P_1 = -4,2813 + 0,0000i$$

$$P_2 = -1,0068 + 2,8959i$$

$$P_3 = -1,0068 - 2,8959i$$

$$P_4 = -1,4324 + 0,0000i$$

$$P_5 = -0,4506 + 1,0480i$$

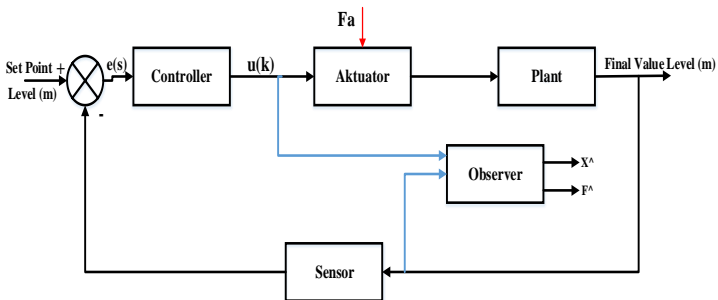
$$P_6 = -0,4506 - 1,0480i$$

Selanjutnya dari nilai *pole-placement* diatas matriks *gain observer* didapatkan dengan menggunakan program perangkat lunak matlab R2014a dengan *syntax* “*place*” sehingga diperoleh matriks *Ktild* atau *Ke* sebagai berikut :

Berikut ini adalah gambar diagram blok sistem pengendalian yang telah diintegrasikan dengan observer.

$$K_e = \begin{bmatrix} -0,2277 & 0 \\ 12,2875 & 0 \\ 0 & 1,3014 \\ 1,2273 & 0 \\ 0 & -0,0988 \end{bmatrix}$$

Untuk skema perancangan observer dapat dilihat di Lampiran dan untuk *syntax* perancangan observer di Lampiran. Berikut ini adalah gambar diagram blok sistem pengendalian yang telah diintegrasikan dengan observer

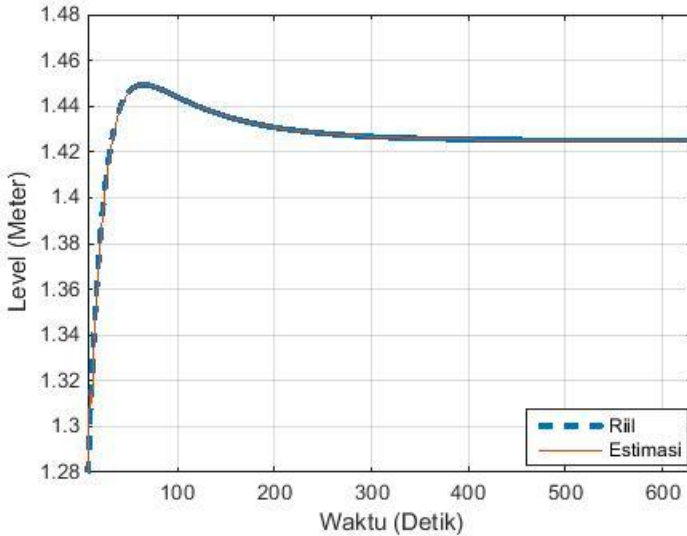


**Gambar 3.4** Diagram blok sistem pengendalian yang telah terintegrasikan dengan observer

### 3.7 Validasi Observer

Dalam memvalidasi *observer* untuk memastikan apakah *observer* hasil estimasi *state* yang dibuat telah sesuai dengan plant

yang telah dirancang ditunjukkan pada gambar 3.5 berikut, dimana dapat dilihat pada grafik dengan garis merah yang merupakan hasil estimasi *state level* dari *observer* telah sesuai dengan grafik garis biru yang merupakan respon dari *plant* yang telah dibuat ketika belum diberi kesalahan.



**Gambar 3.5** Hasil estimasi *state level* dari *observer* dibandingkan dengan respon dari *plant*.

### 3.8 Perancangan algoritma prediksi kesalahan

Setelah perancangan algoritma *observer*, langkah berikutnya perancangan algoritma prediksi kesalahan yang mengacu pada (Xu et al., 2009) yaitu algoritma *exponential smoothing*. Estimasi kesalahan yang diperhalus dapat dihitung dengan persamaan sebagai berikut:

$$u_n = a\hat{\alpha}_n + (1 - a)(u_{n-1} + s_{n-1}) \quad (3.44)$$

$$s_n = b(u_n - u_{n-1}) + (1 - b)s_{n-1} \quad (3.45)$$

Misalkan  $u_n$  adalah hasil estimasi yang diperhalus dan  $s_n$  adalah faktor pertumbuhan linear yang diestimasi. Diberikan hasil

estimasi kesalahan  $\hat{a}_n (n = 1, 2, \dots)$  yang diperoleh dari *observer*. Dimana  $0 < a < 1$  &  $0 < b < 1$  adalah parameter *Holt's method* (Xu et al., 2009). Kondisi awal untuk *Holt's method* sebagai berikut:

$$u_1 = \hat{a}_1 = \hat{f}(k) \quad (3.46)$$

$$u_2 = \hat{a}_2 = \hat{f}(k + 1) \quad (3.47)$$

$$s_2 = u_2 - u_1 \quad (3.48)$$

Setelah menghitung estimasi kesalahan yang diperhalus dan faktor pertumbuhan pada saat  $nT$  prediksi kesalahan kemudian dapat dihitung menggunakan algoritma berikut (Xu et al., 2009):

$$\bar{a}_{n+l} = u_n + s_n l T \quad (3.49)$$

dengan

$l$  : step prediksi untuk prediksi kesalahan

$T$ : *time interval*

Untuk  $n = 1$  atau  $n = 2$  menggunakan persamaan (3.46) atau (3.47) untuk  $n > 2$  menggunakan persamaan (3.44).

### 3.9 Pengujian algoritma prediksi kesalahan aktuator

Pengujian algoritma prediksi kesalahan aktuator dilakukan untuk memastikan apakah algoritma *observer* dapat diterapkan pada algoritma prediksi kesalahan aktuator yang mengacu algoritma *exponential smoothing*. Hasil estimasi kesalahan menggunakan *observer* digunakan untuk menghitung estimasi kesalahan diperhalus ( $u_n$ ) untuk  $n \geq 2$  dan menghitung faktor pertumbuhan linear ( $s_n$ ) untuk  $n \geq 2$ . Dimana ( $u_n$ ) dan ( $s_n$ ) mengandung parameter *Holt's method*  $a$  dan  $b$  dimana  $0 < a < 1$  &  $0 < b < 1$ . Pengujian ini dilakukan dengan memberikan variasi nilai  $T$  (*time interval*) 1 detik, 3 detik, dan 5 detik. Setiap nilai *time interval* diberikan variasi jumlah prediksi kesalahan  $\hat{f}(k + 2)$ ,  $\hat{f}(k + 4)$ ,  $\hat{f}(k + 6)$  dan  $\hat{f}(k + 9)$ . Pengujian dilakukan dengan menggunakan sinyal kesalahan berupa *ramp*.

### 3.10 Perancangan Algoritma Prediksi Keandalan Real-Time

Langkah terakhir yaitu perancangan algoritma prediksi keandalan *real-time* yang mengacu pada (Xu et al., 2009).

- 1) Inisialisasi partikel dimulai dari  $l = 1$ .

$$\bar{X}_{n+l-1|n}^{(i)} = \bar{X}_{n+0|n}^{(i)} = \hat{X}_{n|n}^{(i)}$$

- 2) Estimasi kesalahan menggunakan algoritma *observer* yang akan menghasilkan *state* estimasi mengacu pada (Indriawati et al., 2015).

$$u_1 = \hat{\alpha}_1 = \hat{f}(k)$$

$$u_2 = \hat{\alpha}_2 = \hat{f}(k + 1)$$

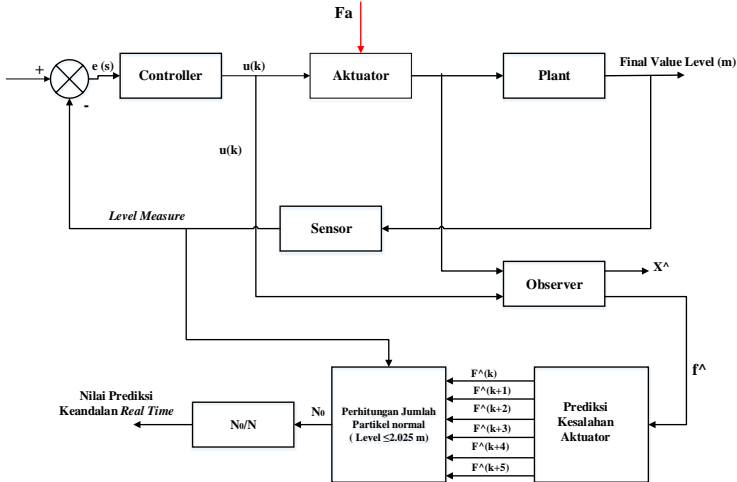
- 3) Prediksi kesalahan menggunakan algoritma *exponential smoothing* pada persamaan (3.30), (3.31), dan (3.35).
- 4) Menghitung partikel baru menggunakan persamaan :

$$\bar{X}_{n+l-1|n}^{(i)} = f(\bar{X}_{n+l-1|n}^{(i)}) + \Gamma \tilde{\alpha}_{n+l} + W_{n+l}^{(i)} \text{ dimana } W_{n+l}^{(i)}$$

adalah realisasi dari  $w_{n+l}$  untuk partikel ke- $i$ .

- 5) Pemilihan partikel jika  $l < r$ , maka  $l = l + 1$ , dan kembali ke step 2, jika sebaliknya lanjut pada step 6.
- 6) Menghitung jumlah partikel dari himpunan  $\{\bar{X}_{n+r|n}^{(i)}\} (i = 1, \dots, N)$  yang jatuh pada  $\mathcal{M}_0$  (area yang diperbolehkan) dan dinotasikan oleh  $N_0$ . Maka sistem prediksi keandalan selama  $(nT, (n+r)T|$  dihitung oleh  $N_0/N$ .





**Gambar 3.6** Diagram blok sistem prediksi keandalan *real-time*

Gambar 3.6 merupakan diagram blok sistem prediksi keandalan *real-time* yang telah diintegrasikan dengan sistem pengendalian *level* separator, *observer*, dan prediksi kesalahan aktuator.  $\hat{f}(k)$  merupakan nilai estimasi kesalahan dari *observer* kemudian digunakan dalam prediksi kesalahan aktuator. Hasil prediksi kesalahan aktuator digunakan matriks untuk mengambil *state* prediksi kesalahan aktuator. Selanjutnya dibandingkan nilai *state* prediksi kesalahan aktuator dengan nilai *level* maksimum sebesar 2,025 m, jika *state* prediksi kesalahan aktuator melebihi nilai 2,025 m maka sistem dikatakan gagal. Lalu pada blok  $N_0/N$  merupakan hasil jumlah partikel yang memasuki area yang diperbolehkan  $\mathcal{M}_0$  yang dinotasikan  $N_0$  (Level tidak melebihi  $\leq 2,025$  m) kemudian dibandingkan jumlah tersebut dengan jumlah seluruh partikel ( $N$ ).

### 3.11 Pengujian Algoritma Prediksi Keandalan Real-Time

Pengujian algoritma prediksi keandalan *real-time* dilakukan untuk menentukan nilai parameter desain yang mempengaruhi prediksi keandalan *real-time* pada sistem pengendalian *level* pada

separator. Hasil prediksi kesalahan aktuator digunakan dalam memprediksi keandalan *real-time*. Pengujian ini dilakukan dengan memberikan nilai  $T$  (*time interval*) 1 detik. Kemudian, diberikan variasi jumlah prediksi yaitu tiga prediksi, lima prediksi, tujuh prediksi dan sepuluh prediksi. Pengujian hasil prediksi keandalan *real-time* dilakukan dengan prosedur sebagai berikut:

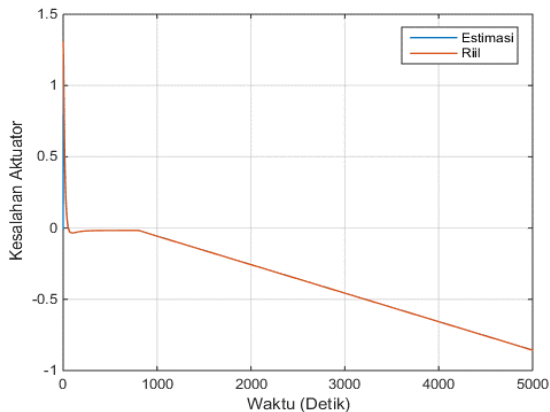
- a. prediksi kesalahan aktuator yang dihasilkan melalui algoritma *exponential smoothing* selanjutnya digunakan matriks untuk mengambil *state* prediksi kesalahan aktuator.
- b. dibandingkan nilai *state* prediksi kesalahan aktuator dengan nilai maksimum level sebesar 2,025 m , jika *state* prediksi kesalahan aktuator melebihi nilai 2,025 m maka sistem dikatakan gagal.

## BAB IV ANALISIS DATA DAN PEMBAHASAN

Bab ini membahas mengenai hasil simulasi dari sistem pengendalian separator menggunakan *software* MATLAB R2014b. Terdapat tiga pengujian yang dilakukan antara lain yaitu pengujian *observer*, pengujian algoritma kesalahan aktuator, dan pengujian prediksi keandalan *real time*

### 4.1. Hasil Uji Observer

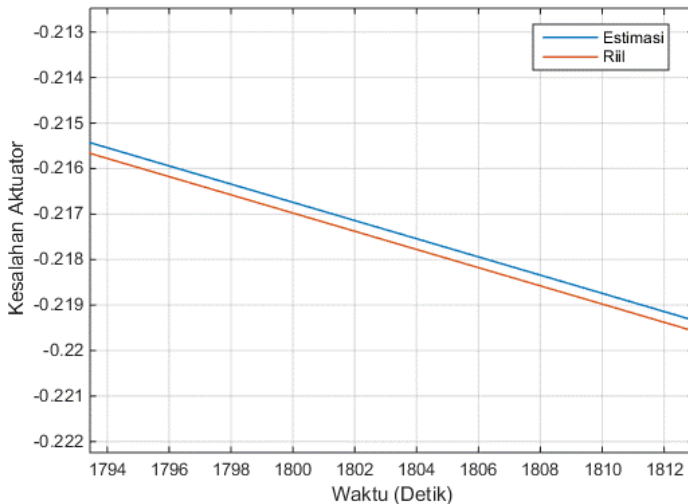
Pengujian *observer* dilakukan dengan tujuan untuk mengetahui kemampuan *observer* dalam mengestimasi kesalahan aktuator. Kesalahan aktuator diasumsikan sebagai kesalahan bias,. Kesalahan bias adalah *Offset* atau penyimpangan konstan pada kondisi normal sinyal aktuator. Kesalahan aktuator yang disimulasikan berupa sinyal *ramp* dengan *slope*  $-0.0001/t$  dan kesalahan diberikan pada detik ke-800. Garis biru tua merupakan hasil estimasi kesalahan aktuator menggunakan *observer* dan garis berwarna merah merupakan hasil residual dari kesalahan aktuator



**Gambar 4.1** Hasil estimasi kesalahan aktuator

Gambar 4.1 menunjukkan *observer* mampu mengestimasi kesalahan aktuator dengan baik. Kontroller PI mampu membuat

respon sistem sesuai dengan kriteria yang diinginkan, seperti yang ditunjukkan pada gambar 4.1 pada saat terjadi perubahan nilai estimasi di awal maupun pada saat detik ke-800, tidak ada *overshoot* dan waktu yang dibutuhkan untuk melakukan estimasi memiliki nilai yang sama dengan nilai riil. Dapat diketahui bahwa hasil estimasi kesalahan aktuator mengikuti dinamika hasil kesalahan residual atau riil yang terjadi.



**Gambar 4. 2** Hasil estimasi kesalahan aktuator (perbesaran gambar 4.1)

Gambar 4.2 merupakan perbesaran dari Gambar 4.1 yang menunjukkan bahwa hasil estimasi kesalahan aktuator dan nilai riilnya terdapat deviasi yang relatif kecil. Selisih antara keduanya dapat dilihat pada Tabel 4.1 dibawah ini pada detik ke 1794 - 1812

**Tabel 4.1** Perbandingan Nilai Riil dan Hasil Estimasi Kesalahan Aktuator

t	Estimasi	Riil	Error (%)
1794	-0,2155	-0,2158	0,1094
1795	-0,2157	-0,2158	0,0167
1796	-0,2159	-0,2162	0,1092
1797	-0,2161	-0,2162	0,0166
1798	-0,2163	-0,2166	0,1090
1799	-0,2165	-0,2166	0,0166
1800	-0,2167	-0,2170	0,1088
1801	-0,2169	-0,2170	0,0166
1802	-0,2171	-0,2174	0,1086
1803	-0,2173	-0,2174	0,0165
1804	-0,2175	-0,2178	0,1084
1805	-0,2177	-0,2178	0,0165
1806	-0,2179	-0,2182	0,1082
1807	-0,2181	-0,2182	0,0165
1808	-0,2183	-0,2186	0,1080
1809	-0,2185	-0,2186	0,0165
1810	-0,2187	-0,2190	0,1078
1811	-0,2189	-0,2190	0,0164
1812	-0,2191	-0,2194	0,1076

Berdasarkan Tabel 4.1 dapat diketahui bahwa deviasi hasil estimasi dengan nilai riil memiliki rata-rata selisih dengan nilai sebesar 0,064%. Dari hasil pengujian diketahui bahwa *observer* yang dirancang dapat memberikan hasil estimasi yang optimal.

#### 4.2 Hasil Uji Algoritma Prediksi Kesalahan Aktuator

Berdasarkan hasil estimasi kesalahan aktuator dari *observer*, kemudian prediksi kesalahan aktuator diterapkan. Algoritma prediksi kesalahan aktuator mengacu pada algoritma *exponential smoothing*. Parameter *Holt's method* pada algoritma *exponential smoothing* diberikan sebesar  $a = 0.8$ , dan  $b = 0.8$ . Pengujian ini dilakukan dengan memberikan variasi nilai  $T$  (*time interval*). Nilai *time interval* dipilih dengan berbagai variasi nilai untuk mengetahui prediksi kesalahan yang dihasilkan. Nilai kesalahan

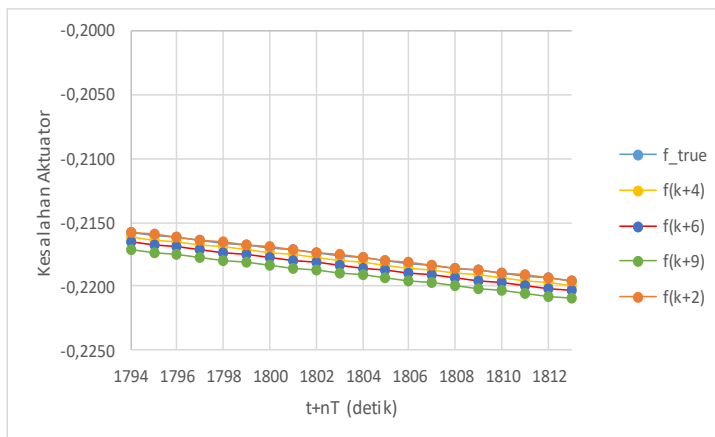
residual atau *riil* akan menjadi validasi dari hasil prediksi algoritma *exponential smoothing*. Nilai sinyal residual setara dengan  $f(k+1)$  dan algoritma prediksi kesalahan *exponential smoothing* memberikan kesalahan  $f(k+n)$ ,  $n \geq 2$ . Pengujian algoritma prediksi kesalahan aktuator ini dilakukan dengan memberikan kesalahan bias pada aktuator berupa sinyal *ramp* dengan *slope*  $-0.0001/t$  dan kesalahan dimulai pada detik ke-800. Berikut ini adalah hasil prediksi kesalahan aktuator dengan time interval 1

**Tabel 4.2** Hasil Prediksi Kesalahan Aktuator dengan *Time Interval* 1 s

t+nT	f <sub>true</sub>	f(k+2)	f(k+4)	f(k+6)	f(k+9)
1794	-0,2158	-0,2157	-0,2161	-0,2165	-0,2171
1795	-0,2160	-0,2159	-0,2163	-0,2167	-0,2173
1796	-0,2162	-0,2161	-0,2165	-0,2169	-0,2175
1797	-0,2164	-0,2163	-0,2167	-0,2171	-0,2177
1798	-0,2166	-0,2165	-0,2169	-0,2173	-0,2179
1799	-0,2168	-0,2167	-0,2171	-0,2175	-0,2181
1800	-0,2170	-0,2169	-0,2173	-0,2177	-0,2183
1801	-0,2172	-0,2171	-0,2175	-0,2179	-0,2185
1802	-0,2174	-0,2173	-0,2177	-0,2181	-0,2187
1803	-0,2176	-0,2175	-0,2179	-0,2183	-0,2189
1804	-0,2178	-0,2177	-0,2181	-0,2185	-0,2191
1805	-0,2180	-0,2179	-0,2183	-0,2187	-0,2193
1806	-0,2182	-0,2181	-0,2185	-0,2189	-0,2195
1807	-0,2184	-0,2183	-0,2187	-0,2191	-0,2197
1808	-0,2186	-0,2185	-0,2189	-0,2193	-0,2199
1809	-0,2188	-0,2187	-0,2191	-0,2195	-0,2201
1810	-0,2190	-0,2189	-0,2193	-0,2197	-0,2203
1811	-0,2192	-0,2191	-0,2195	-0,2199	-0,2205
1812	-0,2194	-0,2193	-0,2197	-0,2201	-0,2207
1813	-0,2196	-0,2195	-0,2199	-0,2203	-0,2209

Pada tabel 4.2 terdapat nilai,  $\hat{f}(k+2)$ ,  $\hat{f}(k+4)$ ,  $\hat{f}(k+6)$  dan  $\hat{f}(k+9)$  yang kemudian dibandingkan dengan hasil *riil* atau  $f_{true}$  dimana Nilai  $f_{true}$  merupakan nilai kesalahan sesungguhnya

pada waktu  $t+nT$ , sedangkan nilai prediksi  $\hat{f}(k+2)$  menunjukkan nilai kesalahan pada  $1 \times \text{time interval}$  lebih cepat, dimana pada prediksi kesalahan tersebut memiliki nilai rata-rata *error* sebesar 0,0165 %.  $\hat{f}(k+4)$  merupakan nilai prediksi kesalahan pada  $3 \times \text{time interval}$  lebih cepat terdapat *error* sebesar 0,1673 %. Nilai Prediksi  $\hat{f}(k+6)$  menunjukkan nilai kesalahan pada  $5 \times \text{time interval}$  lebih cepat *error* yang dihasilkan sebesar 0,3510 % . Nilai prediksi  $\hat{f}(k+9)$  merupakan nilai kesalahan pada  $8 \times \text{time interval}$  lebih cepat dan terdapat *error* sebesar 0,6266 % . Berdasarkan hasil yang telah ditunjukkan, dapat diketahui bahwa dengan *time interval* sebesar 1 s deviasi atau *error* prediksi yang dihasilkan masih minim atau sedikit. Grafik hasil perbandingan antara step prediksi dapat dilihat pada gambar 4.3



**Gambar 4.3** Hasil uji prediksi kesalahan aktuator dengan *time interval* 1 detik

**Tabel 4.3** Hasil prediksi kesalahan aktuator dengan *time interval* 3

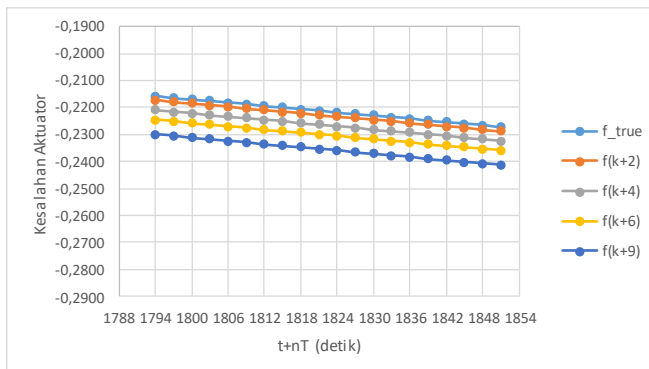
$t+nT$	$f\_true$	$f(k+2)$	$f(k+4)$	$f(k+6)$	$f(k+9)$
1794	-0,2158	-0,2173	-0,2209	-0,2245	-0,2299
1797	-0,2164	-0,2179	-0,2215	-0,2251	-0,2305
1800	-0,2170	-0,2185	-0,2221	-0,2257	-0,2311
1803	-0,2176	-0,2191	-0,2227	-0,2263	-0,2317
1806	-0,2182	-0,2197	-0,2233	-0,2269	-0,2323
1809	-0,2188	-0,2203	-0,2239	-0,2275	-0,2329
1812	-0,2194	-0,2209	-0,2245	-0,2281	-0,2335
1815	-0,2200	-0,2215	-0,2251	-0,2287	-0,2341
1818	-0,2206	-0,2221	-0,2257	-0,2293	-0,2347
1821	-0,2212	-0,2227	-0,2263	-0,2299	-0,2353
1824	-0,2218	-0,2233	-0,2269	-0,2305	-0,2359
1827	-0,2224	-0,2239	-0,2275	-0,2311	-0,2365
1830	-0,2230	-0,2245	-0,2281	-0,2317	-0,2371
1833	-0,2236	-0,2251	-0,2287	-0,2323	-0,2377
1836	-0,2242	-0,2257	-0,2293	-0,2329	-0,2383
1839	-0,2248	-0,2263	-0,2299	-0,2335	-0,2389
1842	-0,2254	-0,2269	-0,2305	-0,2341	-0,2395
1845	-0,2260	-0,2275	-0,2311	-0,2347	-0,2401
1848	-0,2266	-0,2281	-0,2317	-0,2353	-0,2407
1851	-0,2272	-0,2287	-0,2323	-0,2359	-0,2413

Pada tabel 4.3 terdapat nilai,  $\hat{f}(k+2)$ ,  $\hat{f}(k+4)$ ,  $\hat{f}(k+6)$  dan  $\hat{f}(k+9)$  yang kemudian dibandingkan dengan hasil *riil* atau  $f_{true}$  dimana Nilai  $f_{true}$  merupakan nilai kesalahan sesungguhnya pada waktu  $t+nT$ , sedangkan nilai prediksi  $\hat{f}(k+2)$  menunjukkan nilai kesalahan pada  $1 \times \text{time interval}$  lebih cepat, dimana pada prediksi kesalahan tersebut memiliki nilai rata-rata *error* sebesar 0,7064 %.  $\hat{f}(k+4)$  merupakan nilai prediksi kesalahan pada  $3 \times \text{time interval}$  lebih cepat terdapat *error* sebesar 2,3322 %. Nilai  $\hat{f}(k+6)$  menunjukkan nilai kesalahan pada  $5 \times \text{time interval}$  lebih cepat *error* yang dihasilkan sebesar 3,9580 % . Nilai  $\hat{f}(k+9)$



merupakan nilai kesalahan pada  $8 \times \text{time interval}$  lebih cepat dan terdapat *error* sebesar 6,3968 %.

Berdasarkan hasil yang telah ditunjukkan, dapat diketahui bahwa dengan *time interval* sebesar 3s deviasi atau *error* pada prediksi yang dihasilkan lebih besar dari pada *error* pada *time interval* 1. Grafik hasil perbandingan antara step prediksi dapat dilihat pada gambar 4.4, dimana pada gambar 4.4 terdapat jarak antar garis sedikit jauh dikarenakan memiliki deviasi dengan nilai  $f_{true}$ .



**Gambar 4.4** Hasil uji prediksi kesalahan aktuator dengan *time interval* 3 detik

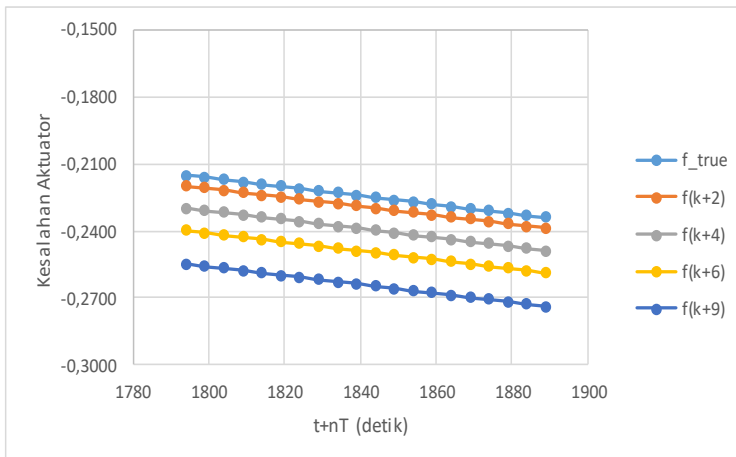
Sehingga untuk membuktikan bahwa dengan penambahan *time interval* mengakibatkan penambahan deviasi, maka dilakukan simulasi dengan peningkatan *time interval* yang lebih besar lagi.

**Tabel 4.4** Hasil Prediksi Kesalahan Aktuator dengan *Time Interval* 5 s

t+nT	f <sub>true</sub>	f(k+2)	f(k+4)	f(k+6)	f(k+9)
1794	-0,2150	-0,2197	-0,2297	-0,2397	-0,2547
1799	-0,2160	-0,2207	-0,2307	-0,2407	-0,2557
1804	-0,2170	-0,2217	-0,2317	-0,2417	-0,2567
1809	-0,2180	-0,2227	-0,2327	-0,2427	-0,2577
1814	-0,2190	-0,2237	-0,2337	-0,2437	-0,2587
1819	-0,2200	-0,2247	-0,2347	-0,2447	-0,2597
1824	-0,2210	-0,2257	-0,2357	-0,2457	-0,2607
1829	-0,2220	-0,2267	-0,2367	-0,2467	-0,2617
1834	-0,2230	-0,2277	-0,2377	-0,2477	-0,2627
1839	-0,2240	-0,2287	-0,2387	-0,2487	-0,2637
1844	-0,2250	-0,2297	-0,2397	-0,2497	-0,2647
1849	-0,2260	-0,2307	-0,2407	-0,2507	-0,2657
1854	-0,2270	-0,2317	-0,2417	-0,2517	-0,2667
1859	-0,2280	-0,2327	-0,2427	-0,2527	-0,2677
1864	-0,2290	-0,2337	-0,2437	-0,2537	-0,2687
1869	-0,2300	-0,2347	-0,2447	-0,2547	-0,2697
1874	-0,2310	-0,2357	-0,2457	-0,2557	-0,2707
1879	-0,2320	-0,2367	-0,2467	-0,2567	-0,2717
1884	-0,2330	-0,2377	-0,2477	-0,2577	-0,2727
1889	-0,2340	-0,2387	-0,2487	-0,2587	-0,2737

Pada tabel 4.4 terdapat nilai  $\hat{f}(k+2)$ ,  $\hat{f}(k+4)$ ,  $\hat{f}(k+6)$  dan  $\hat{f}(k+9)$  yang kemudian dibandingkan dengan hasil *riil* atau  $f_{true}$  dimana Nilai  $f_{true}$  merupakan nilai kesalahan sesungguhnya pada waktu  $t+nT$ , sedangkan nilai prediksi  $\hat{f}(k+2)$  menunjukkan nilai kesalahan pada  $1 \times \text{time interval}$  lebih cepat, dimana pada prediksi kesalahan tersebut memiliki nilai rata-rata *error* sebesar 2,1237 %.  $\hat{f}(k+4)$  merupakan nilai prediksi kesalahan pada  $3 \times \text{time interval}$  lebih cepat terdapat *error* sebesar 6,5814 %. Nilai  $\hat{f}(k+6)$  menunjukkan nilai kesalahan pada  $5 \times \text{time interval}$  lebih cepat *error* yang dihasilkan sebesar 11,0392 % . Nilai  $\hat{f}(k+9)$  merupakan nilai kesalahan pada  $8 \times \text{time interval}$  lebih cepat dan terdapat *error* sebesar 17,7258 %. Berdasarkan hasil yang telah ditunjukkan, dapat diketahui bahwa dengan *time interval* sebesar

3s deviasi atau *error* pada prediksi yang dihasilkan lebih besar dari pada *error* pada *time interval* 1 atau *time interval* 2. Grafik hasil perbandingan antara step prediksi dapat dilihat pada gambar. Dengan *time interval* sebesar 5 s dapat disimpulkan algoritma prediksi kesalahan tidak dapat memberikan hasil prediksi kesalahan yang optimal ketika *time interval* diperbesar karena deviasi yang dihasilkan cukup besar.



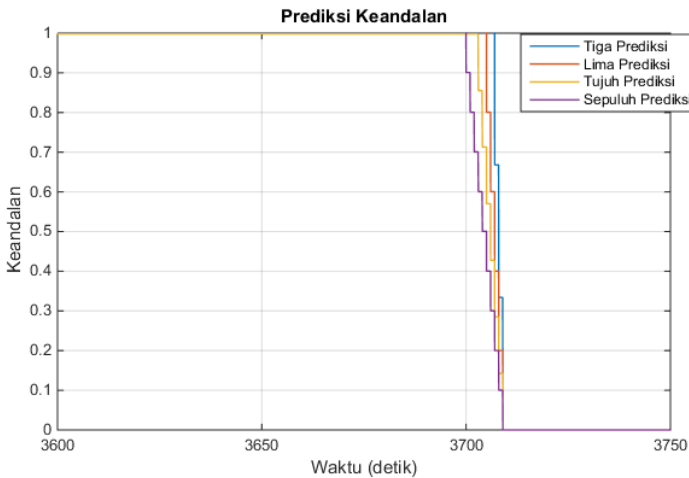
Gambar 4.5 Hasil uji prediksi kesalahan aktuator dengan *time interval* 5 detik

Hasil algoritma prediksi kesalahan aktuator dipengaruhi oleh nilai *time interval* yang diberikan. Semakin kecil nilai *time interval* maka semakin baik performa algoritma prediksi kesalahan aktuator. Sehingga untuk pengujian algoritma prediksi keandalan digunakan *time interval* yang memberikan hasil prediksi kesalahan terbaik yaitu  $T = 1s$ .

#### 4.3 Hasil Uji Algoritma Prediksi Keandalan Real-Time

Algoritma prediksi keandalan *real-time* akan dijalankan dengan menggunakan hasil prediksi kesalahan aktuator yang telah diuji sebelumnya. Ditemukan bahwa algoritma prediksi kesalahan

aktuator bekerja secara optimal dengan *time interval* 1s. Hasil uji dari algoritma prediksi keandalan *real-time* dapat digunakan sebagai acuan tingkat “kesehatan” sistem pengendalian level separator ketika terdapat kesalahan bias pada aktuator. Pengujian algoritma prediksi keandalan ini akan dilakukan untuk prediksi 3 step, 5 step, 7 step dan 10 step kedepan. Kesalahan dimulai pada detik ke-800 dengan batas kesalahan  $\mathcal{M}_0$  sebesar 2,025 m, yang berarti jika nilai level melebihi 2,025 m, sistem dikatakan gagal.



**Gambar 4.6** Grafik hasil uji prediksi keandalan *real-time* dengan *time interval* 1 detik

Tabel 4.5 menunjukkan terdapat perbedaan nilai keandalan *real-time* dengan *time interval* dan jumlah prediksi. *Time interval* sebesar 1 detik dengan jumlah prediksi sebesar 10 menghasilkan prediksi keandalan *real-time* terbaik.

**Tabel 4.5** Hasil uji prediksi keandalan *real-time* dengan *time Interval 1*

Time Interval	Waktu (detik)	Keandalan Tiga Prediksi	Waktu (detik)	Keandalan Lima Prediksi	Waktu (detik)	Keandalan Tujuh Prediksi	Waktu (detik)	Keandalan Sepuluh Prediksi
1	0-3707	1	0-3705	1	0-3703	1	0-3700	1
	3707-3708	0,668	3705-3706	0,8	3703-3704	0,855	3700-3701	0,9
	3708-3709	0,334	3706-3707	0,6	3704-3705	0,7129	3701-3702	0,8
	3709-5000	0	3707-3708	0,4	3705-3706	0,5703	3702-3703	0,7
			3708-3709	0,2	3706-3707	0,4277	3703-3704	0,6
			3709-5000	0	3707-3708	0,2852	3704-3705	0,5
					3708-3709	0,1426	3705-3706	0,4
					3709-5000	0	3706-3707	0,3
							3707-3708	0,2
							3708-3709	0,1
						3709-5000	0	

Hasil uji prediksi keandalan *real-time* dengan *time interval* ditunjukkan Gambar 4.6 Prediksi kesalahan aktuator mempengaruhi hasil dari prediksi keandalan *real-time*. Waktu kegagalan sebenarnya (*riil*) ketika level separator lebih besar dari 2,025 m pada detik ke-3709 untuk setiap jumlah prediksi. Berdasarkan Tabel 4.5 juga dapat diketahui bahwa keandalan mulai turun pada setiap jumlah prediksi keandalan *real-time* yang dihasilkan berbeda-beda. Pada detik ke-3707 untuk tiga prediksi, pada detik ke-3705 untuk lima prediksi, detik ke-3703 untuk tujuh prediksi, dan detik ke-3700 untuk sepuluh prediksi. Prediksi keandalan *real-time* dengan *time interval* 1 detik untuk sepuluh prediksi mempunyai performansi terbaik jika dibandingkan dengan jumlah prediksi lainnya. Hal ini didasarkan pada waktu keandalan mulai turun untuk sepuluh prediksi lebih teliti, tampak bahwa pada detik ke-3700 sudah terdeteksi terjadi kesalahan aktuator. Hasil tersebut sesuai dengan penelitian sebelumnya yang diaplikasikan pada *three-vessel water tank system* bahwa semakin besar jumlah prediksi maka tingkat prediksi keandalan akan semakin teliti (Xu et al., 2009).

*“Halaman ini sengaja dikosongkan”*

## BAB V PENUTUP

### 5.1 Kesimpulan

Berdasarkan hasil analisis data yang telah dilakukan, didapatkan beberapa kesimpulan sebagai berikut :

- Perancangan *Observer* pada sistem pengendalian *level separator* yang telah dilakukan, dapat diketahui bahwa hasil estimasi kesalahan aktuator mengikuti dinamika kesalahan residual atau riil yang terjadi dengan nilai *error observer* sebesar 0.064%.
- Algoritma prediksi kesalahan *Exponential Smoothing* mampu memberikan hasil prediksi kesalahan yang optimal dengan parameter *Holt's method*  $a = 0.8$  dan  $b = 0.8$  dengan *Time Interval* = 1s. Nilai *time interval* 1 detik yang paling optimal dikarenakan *error* prediksi yang dihasilkan kecil yaitu sebesar 0,6266% pada  $\hat{f}(k + 9)$
- Jumlah prediksi mempengaruhi ketelitian prediksi keandalan *real-time* yang dihasilkan, semakin besar jumlah prediksi maka semakin teliti. Dengan *time interval* 1 detik jumlah prediksi sebesar 10 menghasilkan prediksi keandalan *real-time* lebih baik dibandingkan jumlah prediksi sebesar 3,5, dan 7

### 5.2 Saran

Berdasarkan tugas akhir yang telah dilakukan, terdapat beberapa saran yang dapat diberikan untuk penelitian selanjutnya yaitu sebagai berikut :

- Menggunakan metode algoritma kesalahan yang lain untuk mendapatkan hasil yang lebih optimal.
- Prediksi keandalan *real-time* pada sistem pengendalian level separator ini sekiranya dapat dikembangkan dimana penyebab kegagalan dari sistem selain kesalahan bias pada aktuator

*“Halaman ini sengaja dikosongkan”*



## DAFTAR PUSTAKA

- Basilio, J. C. (2002). Design of PI and PID Controlers With Transient Performance Specification. *IEEE Transactions On Education, Vol. 45, No.4* .
- Campbell, J (2014, Desember) . Dipetik April 30, 2019. dari *Troubleshooting Gas-Liquids Separators-Removal of Liquids From The Gas* :<http://www.jmcampbell.com/tip-of-the-month/2014/12/troubleshooting-gas-liquid-separators-removal-of-liquids-from-the-gas/>
- Chinnam, R. B. (1999). *On-line Reliability Estimation of Individual Components, Using Degradation Signals*, 48
- Dhillon, B. S. (2005). *Reliability, Quality, and Safety for Engineers*.
- Ebeling, C. (1997). *An Introduction To Reliability and Maintainability Engineering*. Boston: McGraw-Hill.
- Fitria, M. A (2018) Monitoring Keandalan Real-Time pada Sistem Pengendalian Level Separator Berdasarkan Estimasi Kesalahan. *ITS : Surabaya*
- Indriawati, K., Agustinah, T., & Jazidie, A. (2015). Robust observer-based fault tolerant tracking control for linear systems with simultaneous actuator and sensor faults: Application to a DC motor system. *International Review on Modelling and Simulations (IREMOS)*, 8 (4), 410-417
- Prawito, K (2005) Automasi Switching Sistem Pembakaran Pada Burner Boiler Pg Padjarakan-Probolinggo. Departemen Teknik Fisika, ITS : Surabaya
- Lu.H, Kolarik.W.J ,and S. Lu (2001) "Real Time Performance Reliability Prediction," *IEEE Transactions On Reliability*, pp. 353 - 357.
- Ogata, K. (1970). *Modern Control Engineering*. New Jersey: Prentice-Hall.
- Ogata, K. (2004). *System Dynamics*. New Jersey: Prentice-Hall.
- Rubiandani, R. (2001). *Peralatan Eksplorasi Minyak* : Bandung

- Rosemount. (2019, Juli). Dipetik Juli 02, 2019, dari <http://www.emerson.com/en-us/catalog/rosemount-3051-coplanar-pressure-transmitter>
- Walijah, C. S. (2017). Perancangan Active Fault Tolerant Control Pada Sistem Pengendalian Level Separator Di Petronas Muriah LTD Dengan Kesalahan Pada Sensor dan Aktuator. *ITS : Surabaya*.
- Wang, S. (2000). Dynamic Simulation Experimental Investigation And Control System Design Of Gas Liquid Cylindrical Cyclone Separators. *Tulsa, Oklahoma, Amerika Serikat*.
- Yuriy, S. (2007). *Continuous-Time Systems*. Mexico: *Guanajuato University*, Published by Springer
- Zhegguo Xu, Y. j. (2008). Real-time Reliability Prediction for a Dynamic System Based on the Hidden Degradation Process Identification.
- Zhengguo Xu, Y. J. (2009). A New Real-Time Reliability Prediction Method for Dynamic Systems Based on On-line Fault Prediction. *IEEE Transaction On Reliability*, Vol 58. 3.

## Lampiran A

### A. Mfile pemodelan Separator

```
clear
clear
clc
A=[-5 0 0;4.165 -0.5 0;0 0 0]
B=[3.75 0 0;0 0 0;0 -0.856*11.22*0.0025
0.856*11.22*0.0025]
C=[0 1 0; 0 0 1]
D=zeros(2,3);
Ts = 0.01;
Fa=[3.75;0;0]
Abar=eye(2)
n = length(A)
p = size(C,1)
m = size(B,2)
fa = size(Fa,2)
Aa=[A zeros(3,2);Abar*C -Abar]
Ba=[B;zeros(2,3)]
Ca=[zeros(2,3) eye(2)]
Ea=[Fa;zeros(p,fa)]
Atild = [Aa Ea; zeros(1,5) zeros(1,1)];
Ctild=[Ca zeros(2,1)]
Bo = [zeros(n+p,fa); eye(fa)]
Qf = eye(n+p+fa)
Qe = eye(fa)

setlmis([])
P = lmivar(1,[n+p+fa 1])
G = lmivar(2, [n+p+fa p])
gamma = lmivar(1,[1 0])

lmiterm([1 1 1 P],1,Atild,'s')
lmiterm([1 1 1 G],-1,Ctild,'s')
lmiterm([1 1 1 0],Qf)
lmiterm([1 1 2 P],1,Bo)
lmiterm([1 2 2 gamma],-1,1)
```

```

lmiterm([-2 1 1 P],1,1)

LMIs = getlmis;

[t,xdec]=feasp(LMIs);
P = dec2mat(LMIs,xdec,1);
G = dec2mat(LMIs,xdec,2);
gamma = dec2mat(LMIs,xdec,3);
Ktild = inv(P)*G
K = Ktild(1:n+p, :)
L = Ktild(n+p+1:n+p+fa, :)
ars=eig(Atild-Ktild*Ctild)

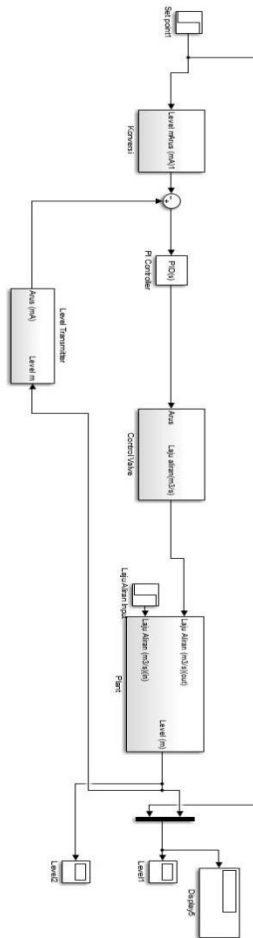
%Exponential Smoothing
a = 0.8;
b = 0.8;
Tp= 2;
T = 0.01;
end

```

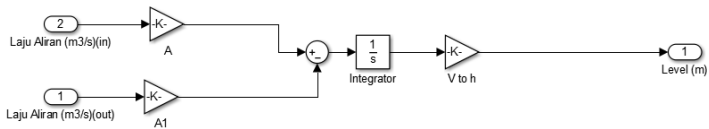
## LAMPIRAN B

- B. Simulink perancangan pemodelan sistem pengendalian *level separator*, *observer* dan prediksi keandalan secara realtime pada perangkat lunak matlab 2014b.

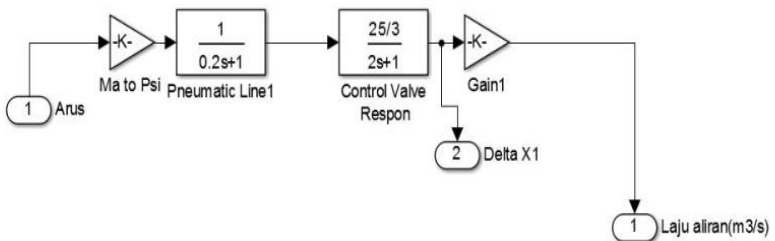
### B.1 Pemodelan sistem pengendalian *level* pada *separator*.



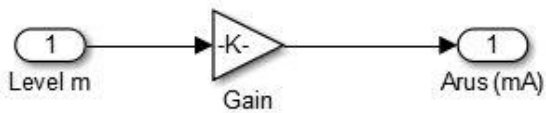
## B.2 Pemodelan Plant (*separator*)



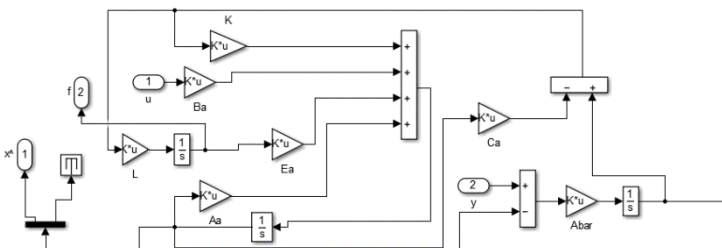
## B.3 Pemodelan Aktuator



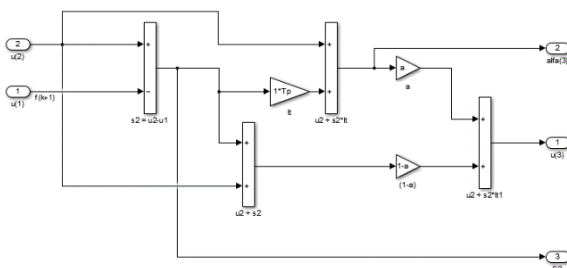
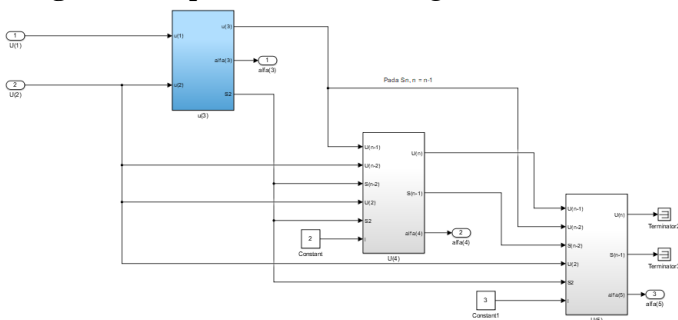
## B.4 Pemodelan *Level Transmitter*



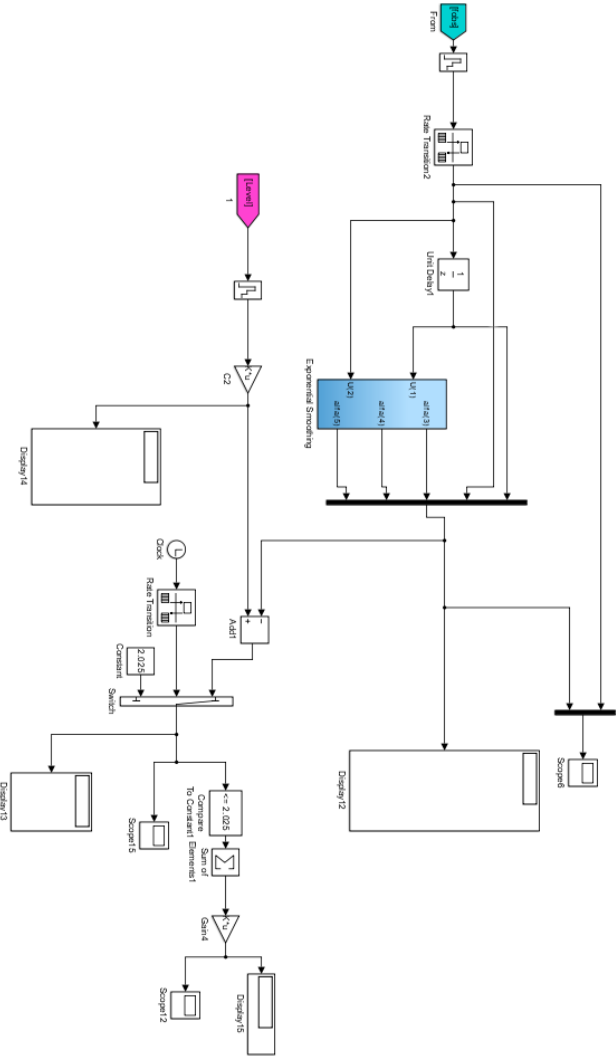
## B.5 Observer



## B.6 Algoritma Exponential Smoothing

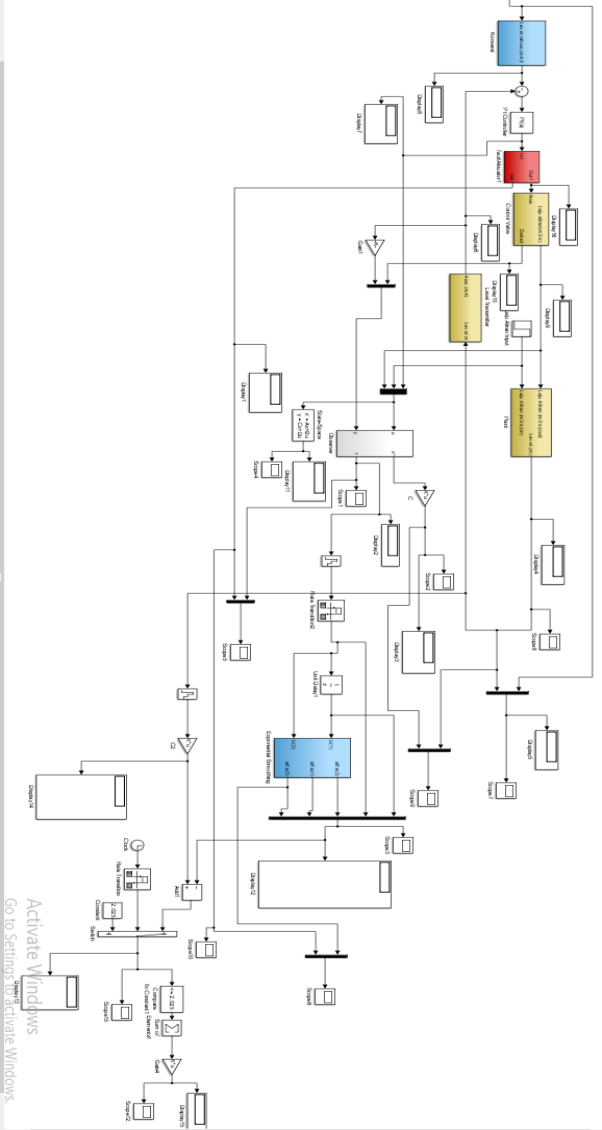


# B.7 Algoritma Reliability Prediction





# B.8 Pemodelan Sistem Pengendalian *Level Separator, Observer* dan Prediksi Keandalan secara *Realtime*.



## **BIODATA PENULIS**



Nama Nur Ummama Sofyana, dilahirkan di Gresik, tanggal 29 Juni 1996. Alamat penulis yaitu di Jalan Sunan Prapen 4 BA RT 07 RW 02 Klangonan. Penulis telah menyelesaikan pendidikan di MI Maarif Al-Hasani, pendidikan di SMPN 4 Gresik, pendidikan di SMAN 1 Kebomas dan sedang menempuh pendidikan S1 Teknik Fisika di ITS hingga sekarang. Bagi pembaca yang memiliki kritik, saran atau ingin berdiskusi lebih lanjut mengenai tugas akhir ini maka dapat menghubungi penulis melalui email : [ummamasofyana96@gmail.com](mailto:ummamasofyana96@gmail.com).