



UNIVERSITAS INDONESIA

PENGENDALIAN PROSES VARIABEL JAMAK *CONTINUOUS STRIRRED TANK REACTOR* (CSTR) MENGGUNAKAN *MODEL PREDICTIVE CONTROL* PADA UNISIM R390.1

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Melengkapi Nilai Mata Kuliah Skripsi

IRA MUTIARA DEWI

0906604205

FAKULTAS TEKNIK

PROGRAM STUDI EKSTENSI TEKNIK KIMIA

DEPOK

JUNI 2012

HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar

Nama : Ira Mutiara Dewi

NPM : 0906604205

Tanda Tangan : 

Tanggal : 29 Juni 2012

LEMBAR PENGESAHAN

Skripsi diajukan oleh:

Nama : Ira Mutiara Dewi

NPM : 0906604205

Program studi : Teknik Kimia

Judul Skripsi : Pengendalian Proses Variabel Jamak *Continuous Stirred Tank Reactor* (CSTR) Menggunakan *Model Predictive Control* pada Unisim R390.1

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Kimia, Fakultas Teknik, Universitas Indonesia.

Dewan Penguji :

Pembimbing : Ir. Abdul Wahid , M.T.

Penguji : Dr. Heri Hermansyah, S.T., M. Eng.

Penguji : Dr. Ing. Ir. Misri Gozan, M. Tech.

Penguji : Dr. Tania Surya Utami, S. T., M. T.



Ditetapkan di : Depok

Tanggal : 29 Juni 2009

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Allah SWT, atas segala rahmat dan karunia-Nya yang telah memberikan segala kelancaran dan izin sehingga penulis dapat menyelesaikan laporan seminar ini. Laporan seminar ini disusun untuk memenuhi salah satu syarat pencapaian gelar Sarjana Teknik di Departemen Teknik Kimia Universitas Indonesia. Kontribusi nyata dari berbagai pihak membantu penulis dalam menyelesaikan perkuliahan hingga saat ini dalam menyelesaikan seminar. Oleh karena itu, penulis mengucapkan terima kasih kepada :

1. Ir. Abdul Wahid, M. T., selaku pembimbing yang telah membantu dan mengarahkan dalam penyusunan laporan seminar ini.
2. Prof. Dr. Ir. Widodo W. Purwanto, DEA selaku Ketua Departemen Teknik Kimia.
3. Ir. Bambang Heru S., M. T., selaku pembimbing akademik penulis.
4. Seluruh dosen dan karyawan Departemen Teknik Kimia yang telah memberikan ilmu serta bantuan dalam penyusunan makalah seminar.
5. Orang tua, kakak Arizona, sahabat dan keluarga atas dukungan moral dan material,
6. Teman – teman seperjuangan Teknik Kimia Ekstensi angkatan 2009 atas dukungan dan kebersamaan yang telah terjalin selama ini.
7. Seluruh pihak yang telah berkontribusi dan membantu proses penyusunan makalah seminar ini.

Akhir kata, penulis mengharapkan semoga laporan seminar ini dapat memberikan manfaat bagi pengembangan ilmu keteknikan dan kepada berbagai pihak yang berkepentingan.

Depok, 29 Juni 2012

Ira Muitara Dewi

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI

TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertandatangan dibawah ini:

Nama : Ira Mutiara Dewi
NPM : 0906604205
Program studi : Teknik Kimia
Departemen : Teknik Kimia
Fakultas : Teknik
Jenis Karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan Universitas Indonesia **Hak Bebas Royalti Noneklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul:

PENGENDALIAN PROSES VARIABEL JAMAK *CONTINUOUS STIRRED TANK REACTOR* (CSTR) MENGGUNAKAN *MODEL PREDICTIVE CONTROL* PADA UNISIM R390.1

Dengan Hak Bebas Royalti Noneklusif ini Universitas Indonesia berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 29 Juni 2009

Yang menyatakan



(Ira Mutiara Dewi)

ABSTRAK

Nama : Ira Mutiara Dewi
Program Studi : Teknik Kimia
Judul : Pengendalian Proses Variabel Jamak *Continuous Stirred Tank Reactor* (CSTR) Menggunakan *Model Predictive Control* pada Unisim R390.1

Model Predictive Control (MPC) merupakan sistem pengendalian yang menggunakan model berdasarkan data hasil pengukuran keluaran (*output*) saat ini atau masa sebelumnya untuk memprediksi nilai dari variabel proses (*input*) pada masa yang akan datang. Pada penelitian ini, sistem pengendalian MPC digunakan untuk menangani pengendalian proses variabel jamak dalam unit operasi *Continuous Stirred Tank Reactor* (CSTR) dengan reaksi pembuatan propylene glycol. Model dinamik sesuai dengan kondisi operasi yang dapat mewakili interaksi antara variabel jamak dibuat untuk diterapkan pada sistem pengendali. Sistem pengendalian proses disimulasikan dengan menggunakan perangkat lunak Unisim R390.1. Simulasi pengendalian proses dilakukan untuk menghasilkan performa pengendalian yang optimum dan untuk mengendalikan variabel jamak yang saling berinteraksi dalam sistem pada CSTR. Optimasi pada sistem pengendalian dilakukan dengan cara *tuning* terhadap parameter-parameter MPC seperti *model horizon* (N), waktu sampel (T), *prediction horizon* (P), dan *control horizon* (M). Hasil dari simulasi menunjukkan Model F sebagai model dinamik terbaik pada pengendali MPC *multivariable* mampu menangani jangkauan perubahan *setpoint* dalam rentang perubahan yang kecil dari 0,33 ke 0,331 dengan IAE sebesar 0,10602. Secara keseluruhan, pengendali MPC belum dapat mengendalikan sistem CSTR secara optimum berdasarkan nilai IAE, namun pengendali MPC lebih mampu menjaga kestabilan sistem dibandingkan dengan pengendali PI.

Kata Kunci : *model predictive control*, variabel jamak, *tuning*, CSTR

ABSTRACT

Name : Ira Mutiara Dewi
Study Program : Chemical Engineering
Title : Continous Stirred Tank Reactor (CSTR) Multivariable Process Control Using Model Predictive Control in Unisim R390.1

Model Predictive Control (MPC) are control system which use model based on value output variable at present or past to predict value of future process variable. In this research, MPC control system use to handle multivariable process control in unit operation Continous Stirred Tank Reactor (CSTR) with propylene glycol reaction system. Dynamics model based on operating condition which representative interaction between multivariable are made to implement in control system. Process control system simulating in Unisim R390.1 software. The simulation of process control aims to achieve optimum performance of controller and to control interaction between multivariable in CSTR system. Optimasion will be doing in system control with MPC parameters tuning such as model horizon (N), time sampling (T), prediction horizon (P), and control horizon (M). The Results show that Model F as the best model in MPC multivariable can control the change of setpoint in short length from 0,33 to 0,331 with 0,10602 IAE. Overall, MPC controller can't controlled CSTR system with optimum result based on IEA value, but MPC can make system more stabile than PI controller.

Keywords : model predictive control, multivariable, tuning, CSTR

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
LEMBAR PENGESAHAN.....	iii
KATA PENGANTAR.....	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS	v
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xii
DAFTAR SINGKATAN	xiii
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Sistematika penulisan	4
BAB 2 TINJAUAN PUSTAKA	5
2.1 <i>Continous Stirred Tank Reactor (CSTR)</i>	5
2.1.1 Pengendalian Pada CSTR	6
2.2 Pengendalian proses	7
2.2.1 Pengendalian konvensional	7
2.2.1.1 Pengendalian Berumpan Balik (<i>Feed Back Control</i>).....	8
2.2.1.2 Mode Pengendalian	10
2.2.2 Pengendalian tingkat lanjut	13
2.2.2.1 Model Predictive Control (MPC)	14
2.3 Identifikasi sistem	23
2.4 <i>State ofThe Art</i>	23
BAB 3. METODOLOGI PENELITIAN	26
3.1 Alur Penelitian	26
3.1.1 Variasi model	27

3.1.2	Tuning MPC	27
3.1.3	Perhitungan IAE	28
3.2	Pemodelan dan Perangkat Lunak yang Digunakan	28
3.3	Variabel Penelitian	29
3.4	Tahap Pelaksanaan Penelitian	29
3.4.1	Prosedur Penelitian	29
3.4.2	Prosedur Pengambilan Sampel	56
3.4.3	Prosedur Analisis	56
3.4.4	Prosedur Perhitungan	57
BAB 4. HASIL DAN PEMBAHASAN		
4.1	Identifikasi sistem	59
4.2	Kinerja pengendali PI	65
4.3	Kinerja Pengendali MPC	67
4.3.1	Kinerja MPC <i>Singlevariable</i>	68
4.3.2	Kinerja MPC <i>Multivariable</i>	70
4.3.2.1	Kinerja MPC <i>Multivariable</i> dengan strategi <i>tuning non-adaptive</i>	70
4.3.2.2	Kinerja MPC <i>Multivariable</i> dengan strategi <i>tuning trial error</i>	73
BAB 5. KESIMPULAN DAN SARAN		83
5.1	Kesimpulan	83
5.2	Saran	83
DAFTAR PUSTAKA		85
LAMPIRAN		87

DAFTAR GAMBAR

Gambar 2.1.	CSTR berjaket dengan reaksi eksotermik	5
Gambar 2.2.	Diagram Blok untuk Sistem Pengendali Berumpan Balik	8
Gambar 2.3.	Pengendalian proporsional: Efek terhadap gain pengendali	11
Gambar 2.4.	Pengendalian Proporsial-Integral :	12
	(a) efek dari waktu integral	
	(b) efek dari gain pengendali	
Gambar 2.5.	Pengendalian PID : efek dari waktu derivative	13
Gambar 2.6.	Respon pada proses dengan pengendalian berumpan balik	13
Gambar 2.7.	Struktur dasar MPC.....	15
Gambar 2.8.	Strategi pada MPC	16
Gambar 2.9.	Profile <i>error</i> yang diprediksikan pada MPC	17
Gambar 2.10.	Diagram blok MPC	19
Gambar 2.11.	<i>Multiple-Input, Multiple-Output process</i>	19
Gambar 3.1.	Alur Penelitian	26
Gambar 3.2.	Skema proses CSTR	30
Gambar 3.3.	Tampilan <i>New Case</i>	30
Gambar 3.4.	Tampilan <i>Simulation Basis Manager</i>	31
Gambar 3.5.	Tampilan <i>Component List View</i>	31
Gambar 3.6.	Tampilan <i>Selected Components List</i>	32
Gambar 3.7.	Tampilan <i>Simulation Basis Manager</i> pada tab <i>Fluid Pkgs</i>	32
Gambar 3.8.	Tampilan <i>Fluid Package</i> pada tab <i>Set Up</i>	33
Gambar 3.9.	Tampilan <i>Fluid Package</i> pada tab <i>Activity Model Interaction Parameters</i>	34
Gambar 3.10.	Tampilan tab <i>Reactions</i>	34
Gambar 3.11.	Tampilan <i>Kinetic Reaction</i>	35
Gambar 3.12.	Tampilan tab <i>Stoichiometry</i>	35
Gambar 3.13.	Tampilan tab <i>Basis</i>	36
Gambar 3.14.	Tampilan <i>Material Stream</i>	37
Gambar 3.15.	Tampilan <i>Mixer</i>	38
Gambar 3.16.	Tampilan <i>worksheet Mixer</i>	39
Gambar 3.17.	Tampilan CSTR	39
Gambar 3.18.	Tampilan <i>worksheet CSTR</i>	40
Gambar 3.19.	Tampilan hasil konversi reaksi pada CSTR	41
Gambar 3.20.	Tampilan PFD dalam kondisi <i>steady state</i>	41
Gambar 3.21.	Dialog <i>Dynamics Assistant</i> pada tab <i>General</i>	42
Gambar 3.22.	Tampilan PFD dalam kondisi dinamik	44
Gambar 3.23.	Spesifikasi komposisi pada <i>Product Block</i>	45
Gambar 3.24.	Tampilan <i>Product Block</i> pada tab <i>Conditions</i>	45

Gambar 3.25.	Tampilan <i>PID Controller</i> TIC pada tab <i>Connections</i>	47
Gambar 3.26.	Tampilan <i>PID Controller</i> XIC pada tab <i>Connections</i>	48
Gambar 3.27.	<i>Object</i> dan <i>Variable</i> pada <i>Data Book</i>	49
Gambar 3.28.	Tampilan <i>Data Book</i> pada tab <i>Strip Chart</i>	49
Gambar 3.29.	Tampilan <i>Model Testing</i>	50
Gambar 3.30.	Tampilan <i>MPC Controller</i> pada tab <i>MPC Setup</i>	52
Gambar 3.31.	Tampilan <i>MPC Controller</i> pada tab <i>Connections</i>	53
Gambar 3.32.	Tampilan <i>MPC Controller</i> pada tab <i>Process Models</i>	54
Gambar 3.33.	Tampilan <i>MPC Controller</i> pada tab <i>Parameters</i> (<i>Operations</i>)	54
Gambar 3.34.	Tampilan <i>MPC Controller</i> pada tab <i>Parameters</i> (<i>Configuration</i>)	55
Gambar 3.35.	Tampilan <i>MPC Controller</i> pada tab <i>MPC Setup</i>	56
Gambar 3.36.	<i>Process Reaction Curve</i>	58
Gambar 3.37.	Grafik Respon CV dengan IAE	59
Gambar 4.1.	Tampilan PFD dengan pengendalian PI pada CSTR	62
Gambar 4.2.	Tampilan PFD dengan pengendalian MPC <i>singlevariable</i> pada CSTR	62
Gambar 4.3.	Tampilan PFD dengan pengendalian MPC <i>multivariable</i> pada CSTR	62
Gambar 4.4.	PRC Pengaruh Perubahan F_c Untuk Model A	65
Gambar 4.5.	PRC Pengaruh Perubahan F_{p0} Untuk Model A	65
Gambar 4.6.	Peringatan dari sistem jika terjadi ketidakstabilan sistem ..	67
Gambar 4.7.	Kinerja pengendali MPC <i>singlevariable</i> dengan model A skenario 1	70
Gambar 4.8.	Kinerja Pengendali MPC <i>multivariabel</i> dengan startegi <i>tuning non-adaptive</i> (model A skenario 1)	71
Gambar 4.9.	Kinerja Pengendali MPC <i>multivariabel</i> dengan startegi <i>tuning non-adaptive</i> (Model D skenario 5)	72
Gambar 4.10.	Tampilan kondisi pengendali hasil simulasi dengan startegi <i>tuning non-adaptive</i>	72
Gambar 4.11.	Kinerja Pengendali MPC <i>multivariabel</i> dengan model A skenario 1	74
Gambar 4.12.	Kinerja Pengendali MPC <i>multivariabel</i> dengan model F skenario 1	74
Gambar 4.13.	Kinerja Pengendali PI dan MPC <i>multivariabel</i> pada skenario 4 model C	77
Gambar 4.14.	Kinerja pengendali PI pada skenario 4	78
	a.) perubahan SP konsentrasi	
	b.) perubahan SP temperatur	
Gambar 4.15.	Kinerja Pengendali PI dan MPC <i>multivariabel</i> pada skenario 7 model F	80

DAFTAR TABEL

Tabel 2.1.	<i>State of The Art</i>	24
Tabel 3.1.	Data untuk tab <i>Basis</i>	36
Tabel 3.2.	Data untuk Aliran Umpan	37
Tabel 3.3.	Data untuk <i>Mixer</i>	38
Tabel 3.4.	Data untuk CSTR	40
Tabel 3.5.	Pemilihan rekomendasi pada <i>Dynamics Assistant</i>	43
Tabel 3.6.	Data untuk Pengendali Temperatur	46
Tabel 3.7.	Data untuk Pengendali Konsentrasi Produk	47
Tabel 3.8.	Data untuk <i>Model Testing</i>	50
Tabel 3.9.	Data untuk <i>MPC Controller</i> pada tab <i>Connections</i>	52
Tabel 3.10.	Formula Integrasi Newton-Cotes	60
Tabel 4.1.	Spesifikasi bukaan valve pada setiap model	64
Tabel 4.2.	Parameter FOPDT pada setiap model	66
Tabel 4.3.	Perubahan <i>set point</i> (SP) pada setiap skenario	67
Tabel 4.4.	Nilai parameter <i>tuning</i> dan IAE pada pengendali PI setiap skenario	68
Tabel 4.5.	Parameter FOPDT untuk <i>MPC singlevariable</i>	69
Tabel 4.6.	Parameter <i>tuning MPC singlevariable</i> sesuai Gambar 4.7	70
Tabel 4.7.	Hasil Perhitungan parameter <i>tuning</i> berdasarkan strategi <i>tuning non-adaptive</i>	71
Tabel 4.8.	Perbandingan respon dari skenario 1 dan 2 pada model 2	75
Tabel 4.9.	Perbandingan nilai IAE pada setiap rangkaian pengendali untuk skenario 1	79
Tabel 4.10.	Perbandingan nilai IAE pada rangkaian pengendali PI dan MPC untuk setiap skenario	80
Tabel 4.11.	Perbandingan nilai IAE pada setiap rangkaian pengendali dengan Model F	81

DAFTAR SINGKATAN

CSTR	<i>Continous Stirred Tank reactor</i>
CO	<i>Controlled Output</i>
CV	<i>Controlled Variable</i>
DMC	<i>Dynamic Matrix Control</i>
FOPDT	<i>First Order Plus Dead Time</i>
IAE	<i>Integral Absolute Error</i>
MV	<i>Manipulated Variable</i>
MPC	<i>Model Predictive Control</i>
MIMO	<i>Multi Input Multi Output</i>
PRC	<i>Process Reaction Curve</i>
PV	<i>Process Variable</i>
PI	<i>Proportional-Integral</i>
PID	<i>Proportional-Integral-Derivatif</i>
SP	<i>Set point</i>

DAFTAR LAMPIRAN

LAMPIRAN A	PRC setiap model dan kondisi awal sistem pada bukaan valve tertentu	87
LAMPIRAN B	Perhitungan parameter FOPDT	96
LAMPIRAN C	Hasil tuning pengendali PI pada setiap skenario	99
LAMPIRAN D	Pengaruh perubahan parameter tuning MPC	101
LAMPIRAN E	Perhitungan parameter tuning MPC menggunakan strategi <i>tuning non-adaptive</i>	111
LAMPIRAN F	Kinerja pengendali MPC <i>multivariable</i> pada setiap model dengan skenario 1 dan parameter tuning <i>default</i>	114
LAMPIRAN G	Respon Xpg Pada Tuning Parameter MPC Skenario 1 dengan Model A dan Model F	115
LAMPIRAN F	Respon Xpg Pada Tuning Parameter MPC Skenario 2 dengan Model A	124
LAMPIRAN I	Respon CV pada pengendali MPC <i>Multivariable</i> dengan pengujian menggunakan skenario dan model yang berbeda	131

BAB 1

PENDAHULUAN

1.1. Latar Belakang

MPC (*Model Predictive Control*) merupakan sistem pengendalian yang menggunakan model berdasarkan data hasil pengukuran keluaran (*output*) saat ini untuk memprediksi nilai dari variabel proses (*input*) pada masa yang akan datang. Dalam perkembangannya, MPC digunakan sebagai pengendalian tingkat lanjut karena kemampuannya dalam menangani pengendalian proses variabel jamak atau sistem dengan *Multi Input Multi Output* (MIMO), menangani kendala pada masukan pengendalian (variabel yang dikendalikan) dan keadaan sistem, dan kebutuhan optimasi dalam perilaku yang sistematis (Christofides, et al., 2011). Keberadaan MPC telah menjawab kekurangan pada pengendalian konvensional seperti PID yang mempunyai keterbatasan dalam pengendalian pada proses variabel jamak yang kompleks dan sistem dengan ketidaklinieran yang kuat (Upadhyay, et al., 2010).

MPC sudah banyak diterapkan pada industri terutama industri petrokimia, hal ini dibuktikan dengan survey oleh Qin dan Badgwell (2003) yang menunjukkan data statistik pada akhir tahun 1999 bahwa aplikasi MPC telah digunakan lebih dari 4500 aplikasi, khususnya pada industri pengolahan minyak bumi dan petrokimia. Namun penerapannya kini meluas, selain pada industri pengolahan minyak, MPC juga digunakan pada industri kimia, proses makanan, *automotive*, perkapalan, metalurgi dan *pulp and paper*, sehingga hal ini menjadikan MPC sebagai teknik pengendalian pada industri yang paling aplikatif setelah PID (Yu, et al., 2005).

Pada penelitian ini, simulasi MPC dilakukan untuk mengendalikan Reaktor berpengaduk secara kontinyu atau *Continous Stirred Tank Reactor* (CSTR) dengan sistem reaksi pembuatan propylene glycol.

Pemilihan CSTR sebagai alat proses pada simulasi dikarenakan reaktor sebagai tempat berlangsungnya reaksi kimia merupakan alat proses yang banyak digunakan dalam industri kimia dan industri makanan (Dotsal et al., 2011). Pengendalian reaktor eksotermis banyak dijadikan sebagai subjek penelitian, terutama karena ketidaklinearan sistemnya. Contohnya, pengendalian temperatur

diperlukan pada reaksi kimia yang terjadi dalam reaktor karena temperatur akan mempengaruhi konversi yang terjadi dan perolehan produk yang diinginkan (Khaniki et al., 2007). Reaktor eksotermis mempunyai perilaku ketidaklinieran yang kuat, yang dipengaruhi oleh adanya interaksi antara sejumlah panas hasil generasi dengan panas yang dipindahkan yang akan berpengaruh pada kinetika reaksi (Aris, 1969). Selain itu, reaktor eksotermis juga mempunyai sistem variabel jamak, dimana terdapat interaksi antara konsentrasi reaktan dengan temperatur reaktor. Sifat interaksi ini menyebabkan jika terdapat perubahan salah satu variabel proses (sebagai variabel yang dimanipulasi, contohnya temperatur aliran pendingin) akan mempengaruhi lebih dari satu variabel proses lainnya (sebagai variabel yang dikontrol, contohnya temperatur reaktor dan komposisi produk). Sifat interaksi seperti ini ditambah waktu tunda yang lama, membuat sistem sangat sulit untuk dikontrol oleh pengendalian konvensional, sehingga diterapkan pengendalian dengan MPC.

Dengan adanya penerapan sistem pengendalian MPC pada CSTR diharapkan temperatur reaktor dan konsentrasi produk sesuai dengan *setpoint*, dan masalah ketidaklinieran dalam reaktor dapat teratasi. Pengendalian sistem disimulasikan dengan menggunakan perangkat lunak proses simulator Unisim R390.1. Simulasi dapat memberikan gambaran proses pada kondisi nyata dan dapat digunakan untuk menilai kinerja sistem pengendalian dalam penanganan gangguan operasional pada proses. Model dinamik sesuai dengan kondisi operasi yang dapat mewakili interaksi antara variabel jamak dibuat untuk diterapkan pada sistem pengendali dalam simulasi proses. Hasil simulasi MPC diuji dengan menghitung *Integral Absolute Error* (IAE), yang kemudian nilainya dibandingkan dengan IAE pada sistem pengendalian *Proportional Integral* (PI).

Penerapan MPC yang semakin meluas, memberikan keyakinan bahwa penelitian mengenai simulasi MPC akan memberikan manfaat teknologi untuk mengatasi masalah ketidaklinearan pada CSTR sehingga dapat diterapkan pada masa yang akan datang.

1.2. Perumusan Masalah

Masalah yang hendak diselesaikan pada penelitian ini adalah bagaimana menerapkan simulasi pengendalian MPC pada sistem CSTR dengan perangkat lunak proses simulator Unisim R390.1 untuk mengendalikan temperatur reaktor dan konsentrasi produk.

1.3. Tujuan Penelitian

Penelitian ini bertujuan untuk :

- 1) Membuat model dinamik variabel jamak pada CSTR,
- 2) Mengendalikan variabel-variabel yang saling berinteraksi pada CSTR (temperatur dan konsentrasi produk) dengan menggunakan sistem pengendalian MPC,
- 3) Menghasilkan performa pengendalian pada sistem CSTR yang optimum berdasarkan kemampuannya mengatasi perubahan *setpoint* dinilai dengan IAE.

1.4. Batasan Masalah

Penelitian ini dibatasi dengan:

- 1) Sistem reaksi yang digunakan yaitu reaksi pembuatan propylene glycol,
- 2) Parameter dalam sistem reaksi, nilai kinetik dan panas reaksi merupakan data dari Fogler, 2006, berdasarkan pada penelitian yang dilakukan Furusawa *et al.*, 1969,
- 3) Sistem yang digunakan yaitu MIMO (*Multi Input Multi Output*) dengan variabel masukan (*input*) dan keluaran (*output*) berukuran 2x2, yaitu dua variabel *input* (*coolant flow* dan *propylene oxide flow/reactan flow*), dan dua variabel *output* (temperatur reaktor dan konsentrasi produk),
- 4) Simulasi proses dalam perangkat lunak proses simulator Unisim R390.1.

1.5. Sistematika Penulisan

Sistematika penulisan seminar ini adalah sebagai berikut:

BAB I PENDAHULUAN

Berisi pendahuluan yang terdiri dari latar belakang, rumusan masalah, tujuan penelitian, batasan masalah dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Berisi tinjauan pustaka yang membahas tentang CSTR; pengendalian proses khususnya pengendalian berumpan balik, pengendalian Proporsional Integral Derivatif (PID), dan pengendalian Model Predictive Control (MPC); pengendalian variabel jamak (MIMO); dan identifikasi sistem model empirik.

BAB III METODOLOGI PENELITIAN

Berisi tentang alur penelitian, model dan perangkat lunak yang digunakan, variabel penelitian, dan tahapan pelaksanaan penelitian.

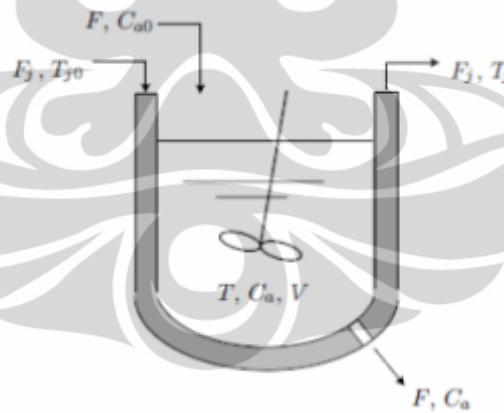
BAB 2

TINJAUAN PUSTAKA

2.1. *Continuous Stirred Tank Reactor (CSTR)*

Reaktor merupakan unit operasi yang banyak digunakan di industri pembuatan bahan-bahan kimia atau industri makanan. Reaktor digunakan sebagai tempat untuk mereaksikan reaktan-reaktan dalam kondisi operasi tertentu sehingga dihasilkan produk dengan spesifikasi yang diinginkan. Dalam pengoperasiannya, reaktor dapat dioperasikan secara *batch*, *semi-batch* dan kontinyu. Jenis reaktor yang paling banyak digunakan dalam industri proses adalah *Continuous Stirred Tank Reactor (CSTR)* atau reaktor berpengaduk yang dioperasikan secara kontinyu.

CSTR biasanya dilengkapi dengan jaket yang berfungsi sebagai pendingin atau pemanas, untuk menopang kebutuhan reaksi eksotermis atau endotermis. Jaket ini digunakan untuk memenuhi sejumlah energi yang akan dipindahkan atau ditambahkan pada reaktor sehingga temperatur proses tetap berada pada *set point*. Skema proses pada CSTR diilustrasikan pada Gambar 2.1.



Gambar 2.1. CSTR berjaket dengan reaksi eksotermik (Bao, et al., 2007)

Berdasarkan Gambar 2.1., panas reaksi dipindahkan dengan menggunakan media pendingin (*coolant*) yang mengalir didalam jaket pada sekeliling reaktor. Sejumlah aliran fluida umpan, F , dengan konsentrasi C_{a0} diumpankan kedalam reaktor. Aliran fluida tercampur sempurna didalam reaktor dan dikeluarkan melalui aliran keluaran dengan konsentrasi C_a . Jaket pada sekeliling reaktor

mempunyai aliran keluaran dan masukan dengan temperatur masukan T_{j0} dan temperatur keluaran T_j . Temperatur pada jaket lebih rendah dibandingkan dengan temperatur pada reaktor.

Model persamaan untuk menggambarkan kondisi CSTR berjaket yang dinamis dengan reaksi orde satu, eksotermis, dan ireversibel, pada reaksi $A \rightarrow B$, dapat ditunjukkan dari kesetimbangan persamaan neraca massa dan energi pada reaktor (Uppal *et al.*, 1974). Asumsi yang digunakan yaitu pencampuran dalam reaktor dan jaket terjadi secara sempurna, reaktor dan jaket berada pada volume yang konstan dan nilai parameter yang konstan.

Neraca massa

(akumulasi massa komponen) = (massa komponen masuk) – (massa komponen keluar) + (penurunan massa komponen)

Neraca energi

(akumulasi U + EP + EK) = (H + EP + KE)_{in} - (H + EP + KE)_{out} + Q - Ws

Persamaan dinamis pada reaktor adalah :

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A0} - C_A) - k_0 \exp\left(\frac{-E_a}{RT}\right) C_A \quad (2.1)$$

$$\frac{dT}{dt} = \frac{F}{V}(T_f - T) - \frac{(-\Delta H)}{\rho C_p} k_0 \exp\left(\frac{-E_a}{RT}\right) C_A - \frac{UA}{V\rho C_p}(T - T_f) \quad (2.2)$$

Neraca kesetimbangan di sekitar jaket ditunjukkan pada persamaan berikut:

$$\frac{dT_j}{dt} = \frac{F_j}{V_j}(T_{j0} - T_j) - \frac{UA}{V_j\rho_j C_{pj}}(T - T_j) \quad (2.3)$$

(Bequette, 2002)

2.1.1. Pengendalian Pada CSTR

Pengendalian pada reaktor banyak dijadikan sebagai subjek penelitian, terutama karena ketidaklinearan sistemnya, khususnya pada reaktor eksotermis yang dilengkapi jaket. Pada reaktor tersebut, ketidaklinieran sistem dipengaruhi oleh adanya interaksi antara sejumlah panas hasil generasi dengan panas yang dipindahkan yang akan berpengaruh pada kinetika reaksi (Aris, 1969). Pada saat temperatur reaktor meningkat, laju reaksi yang terjadi juga akan ikut meningkat. Pada laju reaksi yang tinggi, panas yang digenerasi akan lebih banyak, dan panas

yang dipindahkan karena adanya perbedaan temperatur antara reaktan (reaktor) dan jaket akan semakin banyak. Namun jika tidak terjadi perpindahan panas yang sesuai, maka panas yang berlebih akan terus digenerasikan dan menyebabkan temperatur reaktor akan semakin tinggi (Wade, 2004). Sehingga, temperatur reaksi harus dijaga pada kondisi kesetimbangan antar panas yang di generasi, $G(T)$, dengan panas yang dipindahkan, $R(T)$, karena jika tidak, maka akan terjadi proses “runaway”.

Temperatur reaktor yang semakin tinggi atau temperatur diatas kondisi *steady state* akan sangat berbahaya untuk dioperasikan. Contohnya dapat menyebabkan terjadinya reaksi kedua atau jika dalam produksi propylene glycol dapat terjadi evaporasi pada bahan-bahan yang bereaksi (Fogler, 2006).

Selain itu, reaktor eksotermis juga mempunyai sistem variabel jamak, dimana terdapat interaksi antara konsentrasi reaktan dengan temperatur reaktor. Sifat interaksi ini menyebabkan jika terdapat perubahan salah satu variabel proses (sebagai variabel yang dimanipulasi, contohnya temperatur aliran pendingin atau laju alir reaktan) akan mempengaruhi lebih dari satu variabel proses lainnya (sebagai variabel yang dikontrol, contohnya temperatur reaktor dan komposisi produk). Sistem variabel jamak pada reaktor ini yang kemudian akan dikendalikan dengan menggunakan MPC.

2.2. Pengendalian proses

2.2.1. Pengendalian konvensional

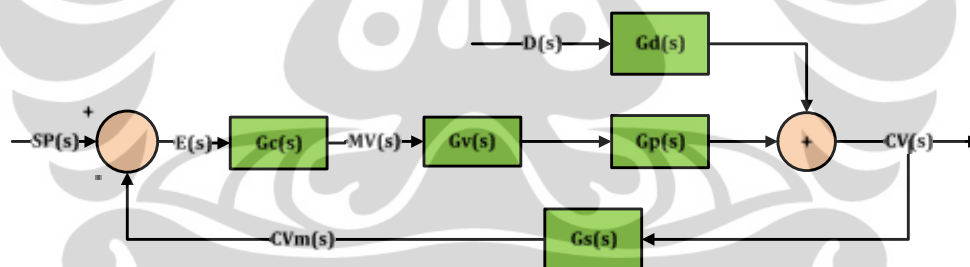
Pembelajaran dan penelitian mengenai pengendalian pada proses telah ada sejak abad ke-19. Pada perkembangannya, pengendalian berumpan balik merupakan generasi awal pada pengendalian proses, dimana dasar-dasar teoritisnya telah dipublikasikan sejak tahun Perang Dunia II. Pengendalian berumpan balik dapat diklasifikasikan sebagai pengendali nilai keluaran (*output*). Salah satu jenis pengendali keluaran yang paling sederhana yaitu *discrete form*, atau biasa disebut pengendali *on/off* atau *two-position control*. Pengendalian berumpan balik merupakan bentuk lama dari jenis pengendalian yang sekarang lebih dikenal dengan nama pengendalian Proportional-Integral-Derivative (PID).

Algoritma pengendalian PID telah berhasil digunakan pada banyak industri-industri proses sejak 1940-an dan tetap menjadi algoritma yang paling banyak digunakan sampai saat ini. Hal ini disebabkan karena strukturnya yang sederhana, kemudahan dalam pengoperasiannya, dapat mencapai *zero offset* dan kemampuannya dalam melakukan tuning secara on-line.

2.2.1.1. Pengendalian Berumpan Balik (*Feed Back Control*)

Konfigurasi pengendalian berumpan balik mengukur secara langsung variabel yang dikendalikan untuk mengatur nilai variabel yang dimanipulasi. Variabel keluaran dari sistem digunakan untuk memutuskan cara yang akan mempengaruhi masukan ke sistem sehingga dihasilkan variabel keluaran yang sedekat mungkin dengan *set point*.

Konfigurasi pengendalian ini digambarkan dalam suatu diagram blok pada Gambar 2.2. Diagram blok digunakan untuk mengkombinasikan elemen-elemen pada proses, instrumentasi, dan pengendali yang mempengaruhi perilaku dinamik sistem. Dengan adanya diagram blok, dapat diketahui masukan dan keluaran pada setiap elemen dan hubungan fungsi transfer individu masukan/keluaran.



Gambar 2.2. Diagram Blok untuk Sistem Pengendali Berumpan Balik (Marlin, 2000)

Keterangan :

Fungsi transfer

$G_c(s)$ = controller

$G_p(s)$ = transmisi, transducer, dan valve

$G_v(s)$ = proses

$G_s(s)$ = sensor, transducer dan transmisi

$G_d(s)$ = Disturbance (gangguan)

Variabel

$CV(s)$ = Controlled Variable

$CV_m(s)$ = Harga terukur dari CV

$D(s)$ = Disturbance (gangguan)

$E(s)$ = Error

$MV(s)$ = Manipulated variable

$SP(s)$ = Set point

Elemen peralatan pada lup berumpan balik dibagi menjadi tiga fungsi alih, yaitu: elemen pengendali akhir, $G_v(s)$; proses, $G_p(s)$; dan sensor, $G_s(s)$. Elemen

pengendali akhir, $G_v(s)$, dapat berupa *valve*, alat pengendali kecepatan motor, *air flow damper*, atau peralatan lain yang menerima sinyal dari pengendali alternatif dan memanipulasi proses, seperti merubah laju alir suatu material. Proses, $G_p(s)$, merespon perubahan variabel yang dimanipulasi (MV) dengan menghasilkan perubahan pada variabel terukur.

Kontroler merupakan elemen penghitung yang dinotasikan dengan $G_c(s)$. Variabel keluaran proses yang dikendalikan dinamakan *controlled variable*, $CV(s)$, dan variabel masukan proses yang diatur oleh sistem kontrol dinamakan *manipulated variable*, $MV(s)$. Harga yang diinginkan, yang harus ditentukan secara independen untuk kontroler, dinamakan *set point*, $SP(s)$. Selisih antara *set point* dengan harga CV terukur disebut sebagai *error*, $E(s)$. Nilai masukan proses yang berubah karena pengaruh dari luar dan mempengaruhi CV dinamakan *disturbance* (gangguan), $D(s)$, dan hubungan antara *disturbance* dan CV adalah fungsi alih *disturbance*, $G_d(s)$.

Pengukur variabel proses, $G_s(s)$, dapat berupa sensor, atau lebih dikenal dengan nama transmitter. Sensor digunakan untuk mengukur variable dalam proses seperti temperatur, level cairan, tekanan atau laju alir, dan kemudian mengkonversikan nilai terukur ke dalam bentuk sinyal untuk transmisi yang akan dikirim ke pengendali alternatif atau sistem pengendali.

Berikut ini merupakan fungsi alih lup tertutup untuk sebuah lup berumpan balik:

Disturbance response:

$$\frac{CV(s)}{D(s)} = \frac{G_d(s)}{1 + G_p(s)G_v(s)G_c(s)G_s(s)} \quad (2.4)$$

Set point response:

$$\frac{CV(s)}{SP(s)} = \frac{G_p(s)G_v(s)G_c(s)}{1 + G_p(s)G_v(s)G_c(s)G_s(s)} \quad (2.5)$$

(Marlin, 2000)

2.2.1.2. Mode Pengendalian

Pengendalian berumpan balik menggunakan metode yang disebut sebagai mode pengendali untuk menentukan nilai dari keluran pengendali. Mode tersebut berupa mode Proporsional (P), Integral (I) dan Derivatif (D).

Mode pengendali dapat digunakan sendiri atau dikombinasikan. Kombinasi yang dapat digunakan yaitu P, PI, PID, I, PD, ID, dan D. Kombinasi yang paling banyak digunakan yaitu P, PI dan PID.

2.2.1.2.1. Pengendalian proporsional

Pada pengendalian dengan mode proporsional (P), aksi kendali (pengaturan terhadap variabel yang dikendalikan) dibuat proporsional terhadap sinyal *error*. Jika *error* meningkat, pengaturan terhadap MV juga harus meningkat. Konsep tersebut ditunjukkan pada persamaan mode proporsional (2.6) dan (2.7) berikut :

$$MV_p(t) = K_c E(t) + I_p \quad (2.6)$$

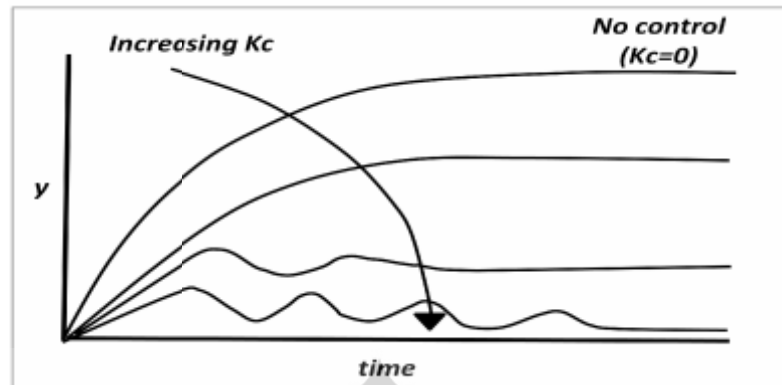
$$G_c(s) = \frac{MV_p(s)}{E(s)} = K_c \quad (2.7)$$

Gain controller, K_c , merupakan parameter pertama dari tiga parameter yang dapat diatur untuk membuat pengendali PID dapat digunakan pada berbagai aplikasi. Pada pengendalian perangkat keras yang komersil, parameter dalam pengendali Proporsional dilambangkan dengan *Proportional Band* (PB).

$$PB = \frac{100 \text{ (kisaran maksimum dari keluaran pada pengendali)}}{K_c \text{ (kisaran maksimum dari variabel terukur)}} \quad (2.8)$$

Dengan adanya mode proporsional (P), osilasi yang terjadi pada variabel proses karena adanya gangguan atau perubahan terhadap *set point* akan lebih cepat diredam dibandingkan dengan skema pengendalian berumpan balik lain.

Pengendalian proporsional cocok diterapkan ketika dibutuhkan respon yang cepat terhadap adanya gangguan dan jika *offset* pada *steady state* tidak penting. Namun perlu diperhatikan bahwa penambahan harga K_c diatas batas kestabilannya (atau terlalu tinggi) akan menyebabkan osilasi yang semakin besar bahkan respon menjadi tidak stabil. Sehingga peningkatan harga K_c perlu dilakukan secara bertahap. Grafik pengaruh peningkatan harga K_c pada pengendalian Proporsional ditunjukkan pada Gambar 2.3.



Gambar 2.3. Pengendalian proporsional: efek terhadap gain pengendali (Seborg, et al., 2003)

2.2.1.2.2. Pengendalian Proporsional dan Integral (PI)

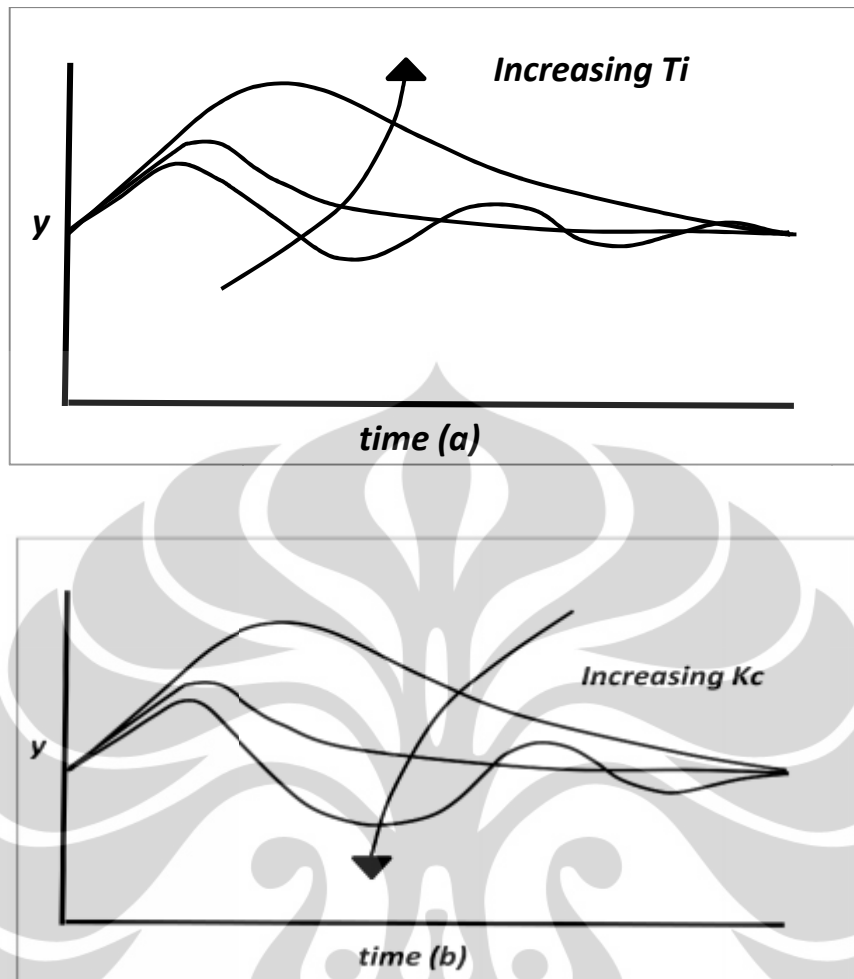
Pada pengendalian PI, osilasi dapat diredam dan dikembalikan ke *set point*. Adanya tidakan Integral membuat skema pengendalian menjadi *zero error*, akan tetapi respon dari pengendali lebih lambat karena terjadi peningkatan periode osilasi.

Keluaran pengendali Proporsional ditambah pengendali Integral didefinisikan pada persamaan (2.9) berikut:

$$MV_p(t) = K_c E(t) + \frac{K_c}{T_i} \int E(t) \quad (2.9)$$

Integral waktu, T_i , didefinisikan sebagai jumlah waktu yang dibutuhkan keluaran pengendali untuk menghilangkan *error*. T_i dan aksi kendali mempunyai hubungan yang resiprokal, jika T_i dinaikkan harganya maka akan menurunkan aksi Integral dan sebaliknya. Kenaikan harga T_i yang terlalu besar setelah adanya gangguan atau perubahan pada *set point*, dapat menyebabkan variabel yang dikendalikan kembali ke *set point* secara lambat.

PI cocok digunakan ketika *offset* yang terjadi tidak dapat di tolerasi. Mayoritas pengendalian dalam *Plant* proses kimia menggunakan PI sebagai pengendalinya.



Gambar 2.4. Pengendalian Proporsional-Integral :
 (a) efek dari waktu integral, (b) efek dari gain pengendali (Seborg, et al., 2003)

2.2.1.2.3. Pengendalian Proporsional Integral Derivatif (PID)

Adanya penambahan tindakan Derivatif pada pengendalian PID dapat mengurangi periode osilasi. Dengan mengukur laju perubahan pada *error*, pengendali dapat mengantisipasi arah *error* dan merespon lebih cepat dibandingkan dengan pengendali tanpa tindakan Derivatif.

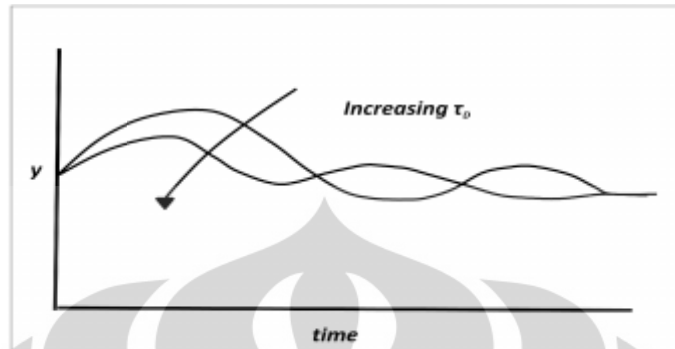
Keluaran pengendali Proporsional + Integral + Derivatif didefinisikan pada persamaan (2.10) berikut:

$$MV_p(t) = K_c E(t) + \frac{K_c}{T_i} \int E(t) + K_c T_d \frac{dE(t)}{dt} \quad (2.10)$$

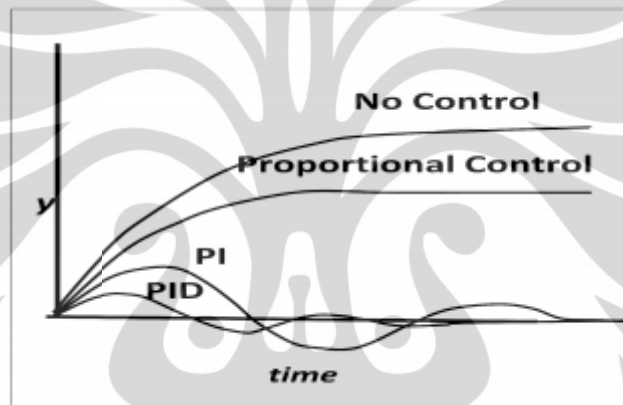
Tindakan pengendali derivatif cocok digunakan pada proses yang mempunyai sedikit atau tidak ada *dead time* dan dengan kapasitas besar.

Dalam skema pengendalian PID, Aksi Integral menjadikan *zero error*, dan aksi Derivatif memperpendek respon pengendalian. Hal ini akan lebih baik

dibandingkan pengendalian dengan mode Proporsional saja. Perbandingan hasil penggunaan kombinasi P, PI dan PID digambarkan pada Gambar 2.6.



Gambar 2.5. Pengendalian PID : efek dari waktu derivatif (Seborg, et al., 2003)



Gambar 2.6. Respon pada proses dengan pengendalian berumpan balik (Seborg, et al., 2003)

2.2.2. Pengendalian tingkat lanjut

Teknik pengendalian proses terus berkembang untuk memenuhi kebutuhan pengendalian suatu proses atau sistem yang memiliki konfigurasi yang berbeda-beda. Teknik tersebut umumnya disebut sebagai pengendalian tingkat lanjut. Generasi awal pada pengendalian tingkat lanjut disebut *Advance Regulatory Control*, yang terdiri dari pengendalian rasio, *cascade*, umpan maju, *override*, pengendalian *multiple input multiple output* (pengendalian *decoupling*), kompensasi waktu-mati dan pengendalian berdasarkan model. Sedangkan generasi terbaru pada pengendalian tingkat lanjut adalah Model Predictive Control (MPC).

2.2.2.1. Model Predictive Control (MPC)

Model Predictive Control (MPC) mulai diaplikasikan pada dunia industri sejak tahun 1970. Pengembangan MPC yang mendapat perhatian besar diantaranya yaitu Dynamic Matrix Control (DMC) oleh Cutler dan Ramaker, Gneralized Predictive Control (GPC) oleh Clarke, dan Model Algorithmic Control (MAC) oleh Richalet.

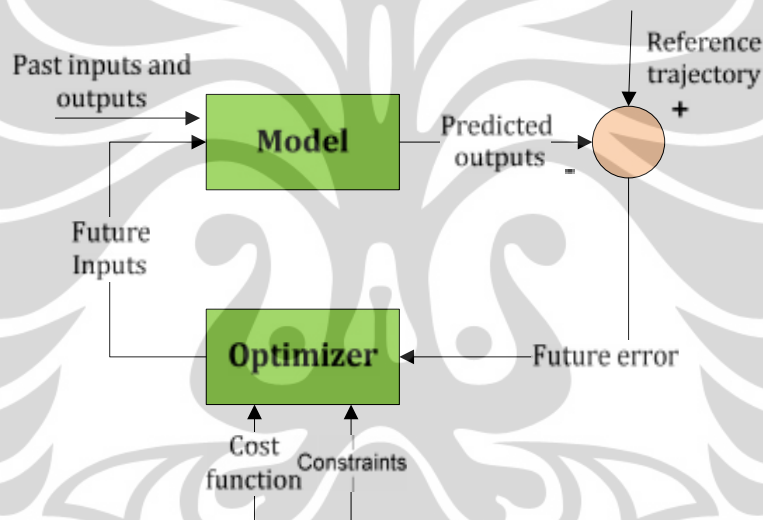
MPC telah diaplikasikan pada sebagian besar industri pengolahan minyak pada unit-unit tertentu antara lain: *fluid catalytic crackers*, *hydrocrackers*, dan *petroleum fractionating towers*. Dimana operasi pada unit-unit tersebut bersifat variabel jamak, memiliki banyak batasan, dan memproses material dalam volume besar (Luyben, 1997). Namun penerapannya kini meluas, selain pada industri pengolahan minyak, MPC juga digunakan pada industri kimia, proses makanan, *automotive*, perkapalan, metalurgi dan *pulp and paper*, sehingga hal ini menjadikan MPC sebagai teknik pengendalian pada industri yang paling aplikatif setelah PID (Yu, et al., 2005).

Keuntungan penggunaan MPC sebagai pengendali dalam proses diantaranya adalah sebagai berikut:

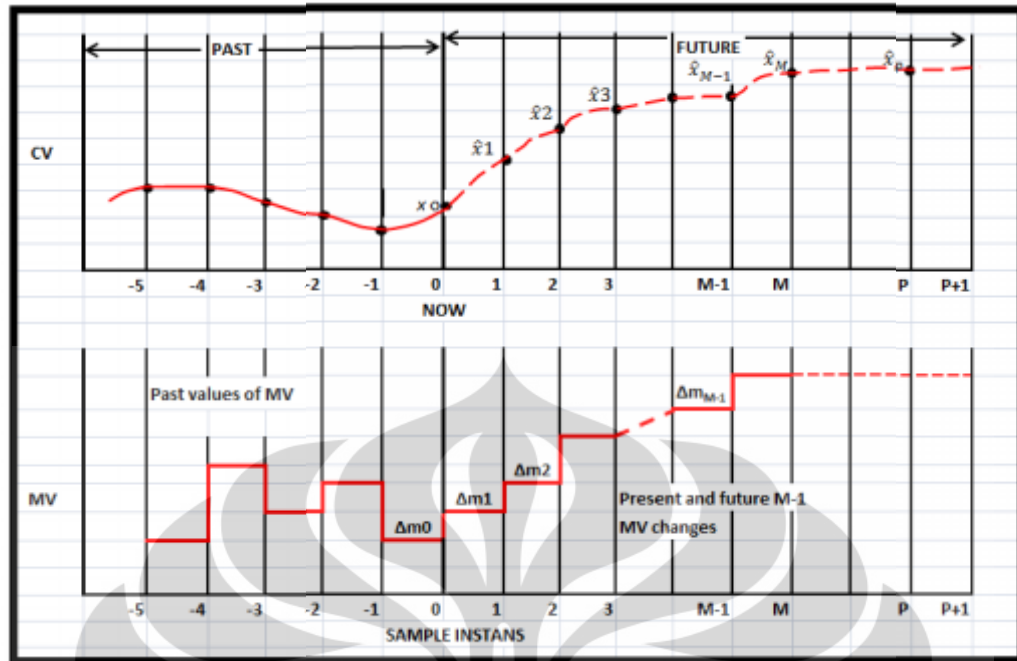
- a. Model dalam proses meliputi interaksi statis dan dinamik antara variabel masukan, keluaran dan gangguan,
- b. Batasan pada masukan dan keluaran diperlakukan dengan perilaku yang sistematis,
- c. Perhitungan pengendalian dapat dikoordinasikan dengan perhitungan pada nilai *set point* yang optimum,
- d. Keakuratan model yang diprediksikan dapat memberikan peringatan awal pada potensial masalah,
- e. Dapat digunakan untuk mengendalikan proses yang beragam, mulai dari proses yang sederhana, hingga sampai proses yang kompleks, seperti proses yang mempunyai waktu tunda besar, *non-minimum phase* atau proses yang tidak stabil,
- f. Dapat menangani sistem variabel jamak,
- g. Mempunyai kompensasi terhadap waktu tunda,

- h. Mempunyai kemampuan dari pengendali umpan maju untuk mengkompensasi gangguan yang terukur.

Pada MPC, sebuah model digunakan untuk memprediksikan keluaran proses yang akan datang (*future inputs*), berdasarkan pada nilai sebelumnya dan saat ini (*past input and output*). Tindakan pengendalian tersebut dikalkulasikan dengan *optimizer* dengan menyertakan kesalahan (*error*) dan batasan proses. Model proses yang digunakan harus mampu membaca dinamika proses sehingga prediksi nilai yang akan datang akan akurat. Gambar 2.7 menunjukkan struktur dasar MPC (Camacho, 2007).



Gambar 2.7. Struktur dasar MPC (Camacho, 2007)



Gambar 2.8. Strategi pada MPC (Wade, 2004)

Strategi pada MPC digambarkan dalam Gambar 2.8. Hasil keluaran di waktu yang akan datang (dinotasikan dengan $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_P$ atau vektor \hat{x}) pada P (*prediction horizon*) diprediksikan setiap t atau waktu pencuplikan menggunakan model proses. Model proses dinotasikan dengan N_1, N_2, \dots, N_P atau vektor N seperti pada persamaan 2.11.

$$N = \begin{bmatrix} N_1 \\ N_2 \\ \vdots \\ N_P \end{bmatrix} \quad (2.11)$$

Keluaran proses terprediksi vektor \hat{x} untuk *control horizon*, $M = 1 \dots P$, bergantung pada nilai masukan dan keluaran lampau, dan serangkaian perubahan sinyal kendali yang dinotasikan dengan $\Delta m_0, \Delta m_1, \dots, \Delta m_{k-1}$ atau vektor Δm .

Nilai CV yang diprediksikan pada *control horizon* (M) ditunjukkan dengan persamaan 2.12 berikut ini:

$$\begin{aligned} \hat{x}_1 &= x_0 + N_1 \Delta m \\ \hat{x}_2 &= x_0 + N_2 \Delta m + N_1 \Delta m \\ \hat{x}_3 &= x_0 + N_3 \Delta m + N_2 \Delta m + N_1 \Delta m \\ &\vdots \\ \hat{x}_M &= x_0 + N_M \Delta m + N_{M-1} \Delta m + \dots + N_1 \Delta m_{M-1} \end{aligned} \quad (2.12)$$

$$\hat{x}_p = x_0 + N_p \Delta m_0 + N_{p-1} \Delta m_1 + \dots + N_{p-M+1} \Delta m_{M-1}$$

Persamaan 2.12 dapat dituliskan dalam bentuk vektor pada persamaan 2.13 dan disederhanakan menjadi persamaan 2.14 berikut ini:

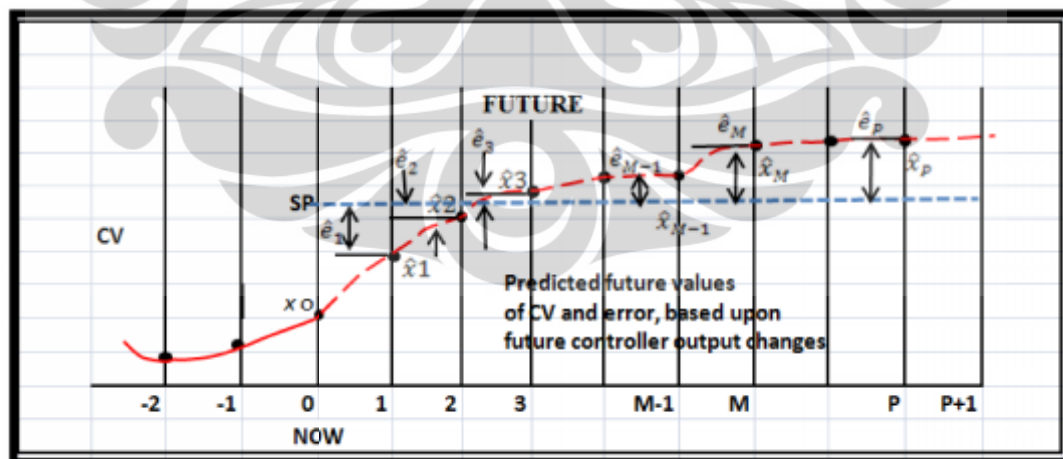
$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \vdots \\ \hat{x}_K \\ \vdots \\ \hat{x}_N \end{bmatrix} = \begin{bmatrix} x_0 \\ x_0 \\ x_0 \\ \vdots \\ x_0 \\ \vdots \\ x_0 \end{bmatrix} + \begin{bmatrix} N1 & 0 & 0 & \dots & 0 \\ N2 & N1 & 0 & \dots & 0 \\ N3 & N2 & N1 & & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ NM & NM-1 & & N2 & N1 \\ \vdots & \vdots & & & \vdots \\ NP & NP-1 & \dots & \dots & NP-M+1 \end{bmatrix} \begin{bmatrix} \Delta m_0 \\ \Delta m_1 \\ \Delta m_2 \\ \vdots \\ \Delta m_{M-1} \end{bmatrix} \quad (2.13)$$

$$\hat{x} = x_0 + N \Delta m \quad (2.14)$$

Serangkaian sinyal kendali yang akan datang dihitung dengan mengoptimasi suatu fungsi kriteria yang ditentukan sebelumnya, dengan tujuan untuk menjaga proses sedekat mungkin terhadap sinyal referensi (yang dapat berupa *setpoint* atau pendekatannya). Fungsi kriteria tersebut umumnya berupa suatu fungsi dari *error* antara sinyal keluaran terprediksi dengan sinyal referensi yang ditunjukkan pada persamaan 2.15.

$$\hat{e} = x_{SP} - x_0 \quad (2.15)$$

Profile CV setelah memperhitungkan *error* digambarkan pada Gambar 2.9 berikut ini:



Gambar 2.9. Profil *error* yang diprediksikan pada MPC (Wade, 2004)

Sinyal kendali Δm_0 dikirim ke proses, sedangkan sinyal kendali selanjutnya seperti $\Delta m_1, \Delta m_2, \dots, \Delta m_{k-1}$ tidak diperlukan lagi. Setelah adanya sinyal kendali dan dilakukannya koreksi pada nilai yang diprediksikan, prosedur perhitungan

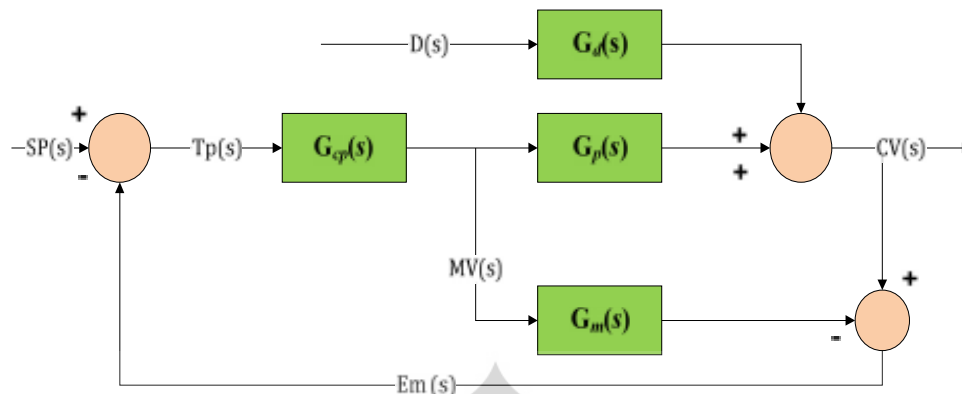
diulangi pada langkah pertama dengan nilai keluaran proses yang baru. Jika terdapat gangguan, maka model step respon menyertakan perhitungan efek gangguan terhadap variabel proses. Gangguan dinotasikan dengan d_1, d_2, \dots, d_p atau vektor d . Persamaan 2.13, 2.14 dan 2.15 mengalami penyesuaian menjadi persamaan 2.16, 2.17, dan 2.18 setelah memperhitungkan gangguan pada proses.

$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \vdots \\ \hat{x}_M \\ \vdots \\ \hat{x}_P \end{bmatrix} = \begin{bmatrix} x_0 \\ x_0 \\ x_0 \\ \vdots \\ x_0 \\ \vdots \\ x_0 \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_M \\ \vdots \\ d_p \end{bmatrix} \Delta u + \begin{bmatrix} N1 & 0 & 0 & \dots & 0 \\ N2 & N1 & 0 & \dots & 0 \\ N3 & N2 & N1 & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ NM & NM-1 & \vdots & N2 & N1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ NP & NP-1 & \dots & \dots & NP-M+1 \end{bmatrix} \begin{bmatrix} \Delta m_0 \\ \Delta m_1 \\ \Delta m_2 \\ \vdots \\ \Delta m_{M-1} \end{bmatrix} \quad (2.16)$$

$$\hat{x} = x_0 + d\Delta u + N\Delta m \quad (2.17)$$

$$e_0 = x_{SP} - x_0 - d\Delta u \quad (2.18)$$

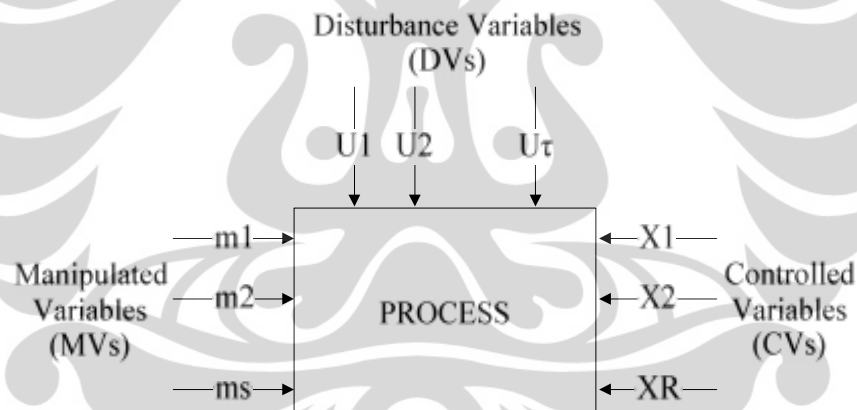
Struktur MPC pada diagram blok dapat dilihat pada Gambar 2.10. Tiga buah fungsi alih merepresentasikan proses sesungguhnya dengan elemen akhir dan sensor, $G_p(s)$; kontroler, $G_{cp}(s)$; dan model dinamik dari proses, $G_m(s)$. Seluruh perhitungan pada sistem kendali prediktif selalu dilakukan setiap kali harga dari elemen final ditentukan. Sinyal *feedback* E_m merupakan perbedaan harga *controlled variable* (CV) terukur dengan prediksinya. Jika model proses sempurna ($G_m(s) = G_p(s)$), nilai variabel E_m sama dengan nilai efek pada gangguan $G_d(s)D(s)$. Sehingga sinyal umpan balik dianggap sebagai koreksi dari model yang digunakan untuk mengoreksi *set point* dalam memperoleh harga target $T_p(s)$ yang lebih baik pada algoritma pengendalian yang diprediksikan (Marlin, 2000).



Gambar 2.10. Diagram blok MPC

2.2.2.1.1. MPC untuk proses dengan variabel jamak

Pada variabel jamak terdapat beberapa variabel yang dikendalikan (*CVs-controlled variable*), beberapa variabel yang dimanipulasi (*MVs-manipulated variable*) dan beberapa gangguan (*DVs-disturbance variables*) pada *set point* yang independen. Skema proses dengan variabel jamak digambarkan pada gambar 2.11 berikut ini:



Gambar 2.11. *Multiple-Input, Multiple-Output process* (Wade, 2004)

Pengendalian sistem proses dengan variabel jamak atau lebih dikenal dengan *Multiple Input-Multiple Output (MIMO) process* memerlukan analisis yang lebih kompleks dibandingkan dengan pengendalian variabel tunggal atau *Single Input-Single Output (SISO) process*. Akan tetapi, konsep sistem variabel tunggal dapat diaplikasikan pada sistem variabel jamak. Pada sistem variabel jamak terdapat interaksi antar variabel seperti perubahan pada satu MV dapat mempengaruhi lebih dari satu CV. Oleh karena itu, model yang digunakan mengalami penyesuaian yang dinotasikan dengan N_{ij} . Notasi “*i*” menunjukkan

CV, sedangkan “ j ” menunjukkan MV. Model proses ditunjukkan pada persamaan 2.19 dan 2.20 berikut ini:

$$N_{ij} = \begin{bmatrix} N_{ij,1} \\ N_{ij,2} \\ \vdots \\ N_{ij,P} \end{bmatrix}, \text{ dengan } i = 1, \dots, R; j = 1, \dots, S \quad (2.19)$$

$$N_{ij} = \begin{bmatrix} N_{ij,1} & 0 & 0 & \dots & 0 \\ N_{ij,2} & N_{ij,1} & 0 & \dots & 0 \\ N_{ij,3} & N_{ij,2} & N_{ij,1} & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ N_{ij,M} & N_{ij,M-1} & \vdots & \dots & N_{ij,1} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ N_{ij,P} & N_{ij,P-1} & \dots & \dots & N_{ij,P-M+1} \end{bmatrix} \quad (2.20)$$

keterangan:

R = banyaknya variabel CVs

S = banyaknya variabel MVs

T = banyaknya variabel DVs

Gangguan pada proses diperhitungkan dengan notasi d_{ik} dalam persamaan 2.21. Notasi “ i ” menunjukkan CV dan “ k ” menunjukkan DV.

$$d_{ik} = \begin{bmatrix} d_{ik,1} \\ d_{ik,2} \\ \vdots \\ d_{ik,P} \end{bmatrix}, \text{ dengan } i = 1, \dots, R; k = 1, \dots, T \quad (2.21)$$

Vektor yang menunjukkan nilai saat ini dan profil CV yang diprediksikan ditunjukkan pada persamaan 2.22 berikut ini:

$$x_{i,0} = \begin{bmatrix} x_{i,0} \\ x_{i,0} \\ \vdots \\ x_{i,0} \end{bmatrix} \quad \hat{x}_i = \begin{bmatrix} \hat{x}_{i,1} \\ \hat{x}_{i,2} \\ \vdots \\ \hat{x}_{i,P} \end{bmatrix}, \text{ dengan } i = 1, \dots, R \quad (2.22)$$

Vektor yang menunjukkan sinyal kendali pada masa yang akan datang ditunjukkan pada persamaan 2.23 berikut ini:

$$\Delta m_j = \begin{bmatrix} \Delta m_{j,0} \\ \Delta m_{j,1} \\ \vdots \\ \Delta m_{j,M-1} \end{bmatrix}, \text{ dengan } j = 1, \dots, S \quad (2.23)$$

Berdasarkan persamaan 2.19-2.23, nilai CV yang diprediksikan pada proses dengan variabel jamak ditunjukkan pada persamaan 2.24 berikut ini:

$$\hat{x}_i = x_{i,0} + \sum_{k=1}^T d_{ik} \Delta u_k + \sum_{j=1}^S N_{ij} \Delta m_j ; \text{ dengan } i = 1, \dots, R \quad (2.24)$$

Bentuk persamaan 2.17 dan 2.18 dapat dianalogikan pada proses dengan variabel jamak dalam persamaan 2.25 dan 2.26 berikut ini:

$$\hat{x} = x_0 + D\Delta u + N\Delta m \quad (2.25)$$

$$e_0 = x_{SP} - x_0 - D\Delta u \quad (2.26)$$

2.2.2.1.2. Parameter tuning dalam MPC

Sejumlah parameter dispesifikasikan untuk merancang sistem MPC. Spesifikasi dilakukan dengan tuning terhadap parameter yang dipilih untuk pengendali MPC seperti *model horizon* (N), waktu sampel (T), *prediction horizon* (P), dan *control horizon* (M).

Parameter tersebut mempunyai efek terhadap kestabilan, *robustness*, dan kinerja pengendali dari algoritma MPC. Pada aplikasinya kriteria yang harus dimiliki oleh pengendali adalah kestabilan dan *robustness*. Sehingga parameter kendali yang mempengaruhi kestabilan dan *robustness* di spesifikasi kisaran nilainya dan dipilih suatu nilai yang akan menghasilkan kinerja pengendalian terbaik. (Agachi, 2006).

Nilai parameter T, P dan M dapat dihitung dengan pendekatan strategi tuning *non-adaptif*. Pendekatan dinamika proses output kontroler untuk pasangan-pasangan variabel proses terukur dengan model FOPDT sebagai berikut (Dougherty, 2003a):

$$\frac{y_r(s)}{u_s(s)} = \frac{K_{rs} e^{-\theta_{rs} s}}{\tau_{rs} s + 1} \quad (2.27)$$

$$(r = 1, 2, \dots, R; s = 1, 2, \dots, S)$$

Penjelasan parameter tuning dalam MPC dijelaskan dalam uraian berikut:

✓ *Model horizon* (N)

merupakan jumlah interval sampel yang diperlukan untuk mencapai kondisi *steady state* ketika step masukan (*input step*) diberikan ke dalam model proses. Umumnya model horizon berada pada nilai $30 < N < 120$ (Ogunaike et al., 1994).

✓ Waktu sampel (T)

merupakan interval waktu yang dipakai dalam pengambilan data. Untuk memastikan kinerja *closed loop* yang bagus, waktu sampel harus cukup kecil untuk menangkap proses yang dinamis dan cukup besar untuk menyediakan kemungkinan untuk implementasi pada waktu nyata.

Waktu sampel dipilih sehingga sedekat mungkin dengan:

$$T_{rs} = \text{Max}(0.1\tau_{rs}, 0.5\theta_{rs}), \quad (r = 1, 2, \dots, R; s = 1, 2, \dots, S) \quad (2.28)$$

$$T = \text{Min}(T_{rs})$$

(Dougherty, 2003a)

✓ *Prediction horizon* (P)

merupakan jarak yang menunjukkan seberapa jauh prediksi yang dilakukan ketika melakukan perhitungan keluaran pengendali alternatif. Terdapat nilai kritis minimum panjang *prediction horizon* untuk mencapai kestabilan dalam sistem *closed loop*. Nilai P yang terlalu kecil akan memicu aksi pengendali yang terlalu agresif, dan menyebabkan ketidakstabilan. Sedangkan pada nilai yang terlalu tinggi, aksi pengendali akan kurang agresif dan respon menjadi lebih lambat (Ogunaike et al., 1994).

Prediction horizon (P) dapat dihitung dengan persamaan sebagai berikut:

$$P = \text{Max} \left(\frac{5\tau_{rs}}{T} + k_{rs} \right) \quad (2.29)$$

$$\text{dengan } k_{rs} = \left(\frac{\theta_{rs}}{T} + 1 \right), \quad (r = 1, 2, \dots, R; s = 1, 2, \dots, S)$$

(Dougherty, 2003a)

✓ *Control Horizon* (M)

merupakan jumlah pergerakan control yang dibuat untuk mencapai *set point* akhir. *Control horizon* digunakan dalam perhitungan optimasi dengan tujuan untuk menurunkan kesalahan yang diprediksikan.

Control horizon (M) dapat dihitung dengan persamaan sebagai berikut:

$$M = \text{Max} \left(\frac{\tau_{rs}}{T} + k_{rs} \right), \quad (r = 1, 2, \dots, R; s = 1, 2, \dots, S) \quad (2.30)$$

(Dougherty, 2003a)

2.3. Identifikasi sistem

Identifikasi proses dibuat untuk menemukan model yang sesuai dengan struktur proses sehingga dapat mendeskripsikan perilaku sistem yang bergantung terhadap waktu. Persamaan matematis dalam model disusun berdasarkan fundamental hukum fisika dan kimia. Dari model matematis yang sudah dibuat, maka dapat dihasilkan hubungan dinamik dalam suatu proses.

Pada proses yang sudah beroperasi, data percobaan dinamik diperoleh dari hasil pengujian *Plant*. Percobaan digunakan ketika proses terlalu kompleks untuk dibuat modelnya. Kebanyakan, percobaan dilakukan untuk mendapatkan parameter-parameter dalam model yang belum diketahui nilainya.

Dalam penentuan model matematis, terdapat klasifikasi model tergantung banyaknya informasi yang terdapat dalam sistem. Dengan adanya konsep model, penentuan identifikasi sistem lebih mudah untuk dilakukan. Klasifikasi model tersebut menggunakan nama kode-warna sebagai berikut:

- ✓ *White Box models*
merupakan sistem dengan informasi atau parameter-parameter yang lengkap.
- ✓ *Grey Box models*
merupakan sistem dengan sebagian parameter yang ditentukan dari data hasil percobaan. *Gray box* terbagi menjadi *physical modeling* dan *semi-physical modeling*.
- ✓ *Black Box models*
merupakan sistem yang tidak mempunyai informasi atau parameter apapun. Pemilihan model berdasarkan parameter yang mempunyai fleksibilitas yang baik. Dalam *black box model*, estimasi dilakukan pada bentuk fungsional dari hubungan antara variabel dan parameter dalam fungsi tersebut.

2.4. State of The Art

Model Predictive Control (MPC) sebagai teknik pengendali terbaru telah banyak diaplikasikan pada industri dalam beberapa dekade terakhir. Keunggulannya dibandingkan pengendali konvensional seperti PI atau PID menyebabkan pengembangan pada MPC yang terus dilakukan untuk lebih memaksimalkan kinerjanya.

Penelitian mengenai MPC telah dimulai sejak tahun 1978 dengan penelitian oleh Richalet et al. yang menunjukkan MPC sebagai teknik *Model Predictive Heuristic Control* (MPHC) dan dinamakan *Model Algorithmic Control* (MAC). Penelitian mengenai MPC terus berkembang hingga muncul istilah DMC, QDMC, GPC, IMC, MAC, NMPC, MMPC, RMPC dan lain sebagainya sebagai bentuk keterbaruan dari teknik pengendali MPC (Aşar, 2004).

Pengendali MPC dapat diterapkan dalam berbagai sistem seperti kolom destilasi, CSTR, *heat exchanger*, tangki dan lain sebagainya. Pengendali MPC memberikan kinerja pengendalian yang lebih baik dibandingkan pengendali PI baik dalam kondisi *steady state* maupun dalam kondisi dinamik pada sistem CSTR dengan reaksi eksotermis yang berorde reaksi nol untuk mengendalikan temperatur dan level (Afonso et al., 1996).

Secara lebih lengkap, *state of the art* penerapan pengendali MPC pada CSTR ditampilkan pada Tabel 2.1 berikut ini:

Tabel 2.1 *State of The Art*

Sistem reaksi	Banyaknya variabel masukan dan keluaran proses	
	SISO	MIMO
eksotermik (tanpa keterangan proses)	(Riggs and Rhinehart et al., 1990)	
		(Santos et al., 2001)
		(Afonso et al., 1996)
proses polimerisasi	(Park et al., 2001)	
	(Cervantes et al., 2002)	
eksotermik (proses produksi propylene glycol)		penelitian yang dilakukan

Pada penelitian ini, pengendalian sistem CSTR disimulasikan dengan menggunakan perangkat lunak proses simulator Unisim R390.1 sebagai perangkat lunak simulator terbaru. Simulasi dalam Unisim R390.1 mengharuskan sistem dalam kondisi konvergen sebelum pengujian pengendalian dilakukan sehingga dapat memberikan gambaran proses pada kondisi nyata dan dapat digunakan untuk menilai kinerja sistem pengendalian dalam penanganan gangguan operasional pada proses.

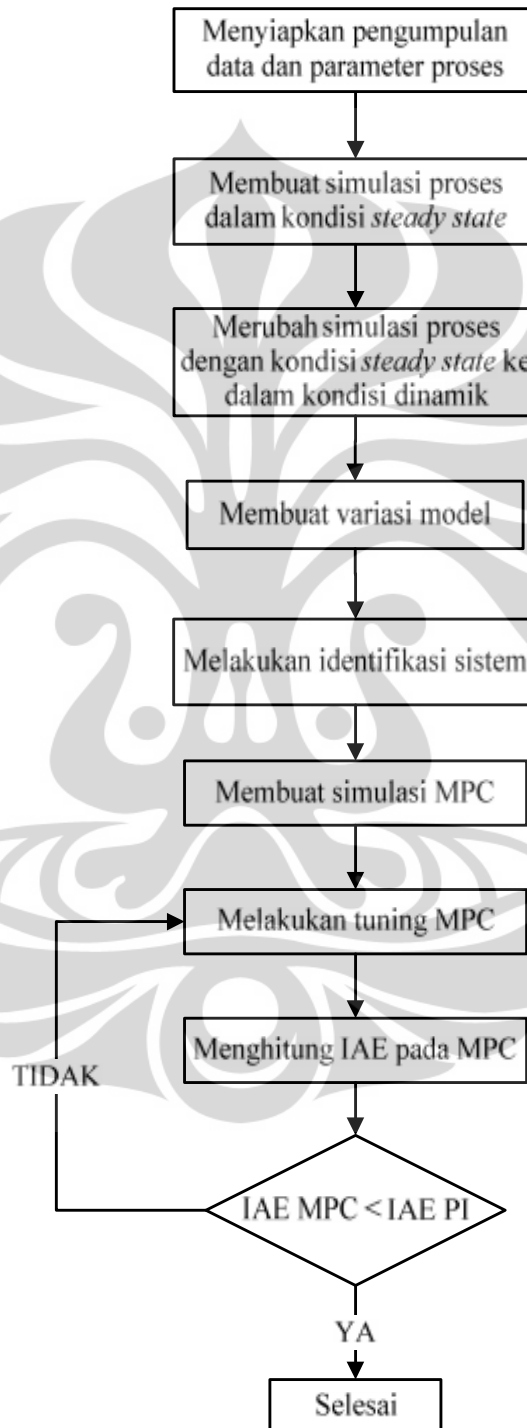
Penelitian dengan menggunakan Unisim R390.1 masih belum banyak dikembangkan. Diharapkan simulasi penerapan pengendali MPC pada sistem CSTR dalam Unisim R390.1 dapat memberikan manfaat teknologi sehingga dapat diterapkan pada kondisi nyatanya.



BAB 3 METODOLOGI PENELITIAN

3.1. Alur Penelitian

Tahapan alur penelitian digambarkan pada Gambar 3.1 berikut ini:



Gambar 3.1. Alur Penelitian

3.1.1 Variasi model

Sesuai dengan namanya, MPC sebagai *Model Predictive Control*, penggunaan model yang sesuai akan mempengaruhi kinerja pengendali tersebut. Dalam penelitian ini, model dibuat dengan perintah *model testing* yang terdapat pada Unisim atau dengan melakukan pengaturan secara manual sehingga sistem dapat diidentifikasi.

Variasi penggunaan model yang dibuat dalam penelitian ini dipaparkan pada penjelasan berikut ini:

- Model A dibuat berdasarkan dengan pengaturan pada Unisim tanpa adanya modifikasi atau disebut sebagai *model testing default*. *Model testing* dilakukan pada kondisi perubahan valve, baik yang mengatur laju alir *coolant* maupun laju alir umpan, dari 50% menjadi 52,5% dengan *signal variation amplitude* sebesar 5%. Tahapan *model testing default* ini akan dipaparkan dalam tahapan identifikasi sistem BAB 3.
- Model B dibuat berdasarkan *model testing* dengan kondisi awal bukaan valve yang masih dapat membuat sistem dalam keadaan stabil. Perbedaan bukaan valve ditentukan pada nilai melebihi 2,5% sehingga terdapat perbedaan bukaan valve dengan *model testing default*.
- Model yang dibuat berdasarkan persentase perbedaan *set point* konsentrasi produk awal dan akhir. Variasi dibuat dengan perbedaan 2,5% , 5%, 10% dan 20% terhadap *set point* akhir yang telah ditentukan, sehingga akan terdapat 4 model pada variasi ini.

3.1.2 Tuning MPC

Tuning parameter pada MPC dilakukan dengan 2 cara, yaitu:

a. Strategi *tuning non-adaptive* (Dougherty, 2002)

Strategi ini dilakukan dengan cara menentukan parameter *tuning* menggunakan persamaan 3.3 – 3.8 yang kemudian parameter tersebut dimasukkan kedalam fasilitas *tuning* MPC pada Unisim.

b. Strategi *tuning trial error*

Strategi *tuning* ini dilakukan dengan cara *trial error* pada nilai parameter MPC. *Tuning* pada parameter MPC dengan parameter yang telah ditetapkan

perangkat lunak dilakukan terlebih dahulu sebagai *tuning default* untuk dijadikan acuan dalam melakukan *trial error*.

3.1.3 Perhitungan IAE

Perhitungan IAE tidak dilakukan pada setiap grafik respon yang dihasilkan dari tuning parameter MPC atau PI. Perhitungan IAE hanya dilakukan pada grafik respon MPC dan PI yang secara visual telah memiliki error yang diperkirakan kecil dan pada batasan parameter tuning yang maksimal.

3.2. Pemodelan dan Perangkat Lunak yang Digunakan

Penelitian ini disimulasikan dengan menggunakan perangkat lunak Unisim R390.1. Sistem yang diidentifikasi adalah alat proses CSTR dengan sistem reaksi pembuatan propylene glycol. Asumsi yang digunakan yaitu pencampuran dalam reaktor dan jaket terjadi secara sempurna, reaktor dan jaket berada pada volume yang konstan dan nilai parameter yang konstan. Parameter dalam sistem reaksi, nilai kinetik dan panas reaksi merupakan data dari Fogler (1992), berdasarkan pada penelitian yang dilakukan Furusawa *et al.* (1969), adalah sebagai berikut :

$$E_a = 32,400 \frac{\text{Btu}}{\text{lbmol}}$$

$$k_0 = 16.96 \times 10^{12} \text{hr}^{-1}$$

$$-\Delta H = 39000 \frac{\text{Btu}}{\text{lbmol}} \text{PO}$$

$$U = 75 \frac{\text{Btu}}{\text{hrft}^2\text{°F}}$$

$$\rho C_p = 53.25 \frac{\text{Btu}}{\text{ft}^3\text{°F}}$$

$$R = 1.987 \frac{\text{Btu}}{\text{lbmol}^{\circ}\text{F}}$$

Desain reaktor diasumsikan berbentuk silinder vertikal dengan rasio ketinggian/diameter 2:1 (Bequette, 2002).

3.3. Variabel Penelitian

- Variabel bebas:
merupakan variabel keluaran pengendali yang dimanipulasi yaitu laju alir *coolant* dan laju alir reaktan (propylene oxide).
- Variabel terikat
merupakan variabel masukan pengendali yang dikendalikan yaitu temperatur reaktor dan konsentrasi produk (propylene glycol).

3.4. Tahapan pelaksanaan penelitian

3.4.1. Prosedur penelitian

1. Menyiapkan pengumpulan data dan parameter proses

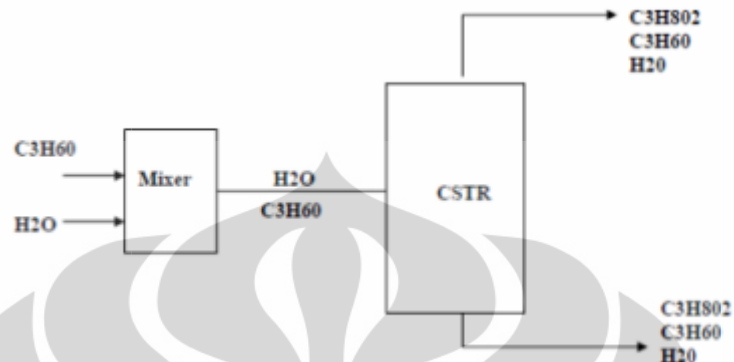
Skema proses CSTR dengan sistem reaksi pembuatan propylene glycol dibuat dalam Unisim. Skema ditampilkan dalam Gambar 3.2. dengan kondisi sebagai berikut:

- ✓ Komponen : propylene oxide, H₂O dan propylene glycol.
- ✓ Fluid Package : Uniquac
- ✓ Reaksi : $C_3H_6O + H_2O \rightarrow C_3H_8O_2$
- ✓ Fasa reaksi : *combined liquid*
- ✓ volume CSTR = 280 ft³
- ✓ liquid volume percent = 85%
- ✓ Faktor A untuk laju reaksi = 16.96×10^{12}
- ✓ Faktor E untuk energi aktivasi = $32,400 \frac{Btu}{lbmol}$

Kondisi umpan adalah sebagai berikut:

- ✓ **Umpan Propylene Oxide**
 - Temperatur = 75°F
 - Tekanan = 16.17 psia
 - Laju alir molar = 150 lbmole/hr
 - Komposisi propylene oxide = 1 (satuan dalam fraksi mol)
- ✓ **Umpan Air**
 - Temperatur = 75°F
 - Tekanan = 16.17 psia

- Laju alir massa = 11000 lb/hr
- Komposisi air = 1 (satuan dalam fraksi mol)



Gambar 3.2. Skema proses CSTR

2. Membuat simulasi proses dalam kondisi *steady state*

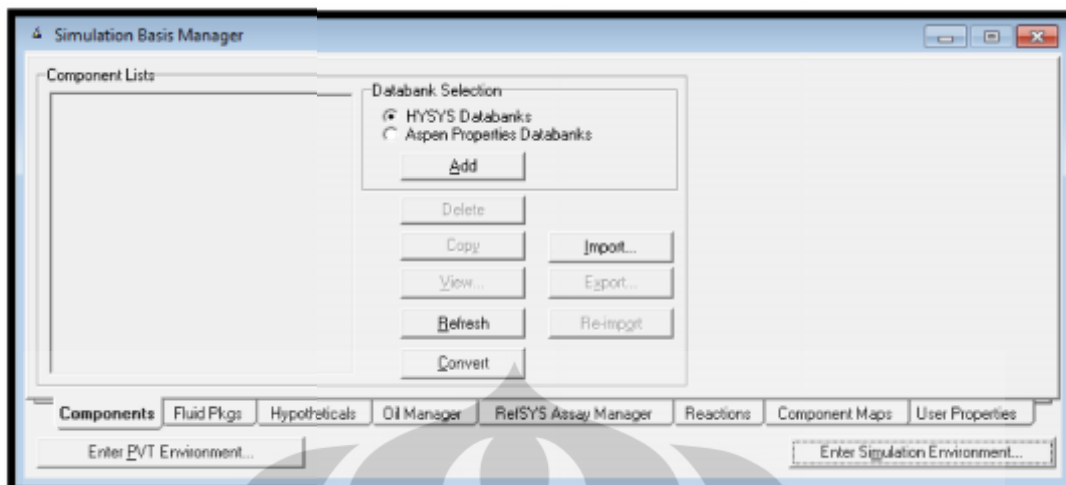
Tahapan pembuatan simulasi proses dalam kondisi *steady state* dilakukan dengan tahapan sebagai berikut:

a) Membuat simulasi baru dalam Unisim

Simulasi baru dalam Unisim dibuat dengan cara klik gambar *New Case* seperti pada Gambar 3.3 sehingga tampil *Simulation Basis Manager* seperti pada Gambar 3.4.



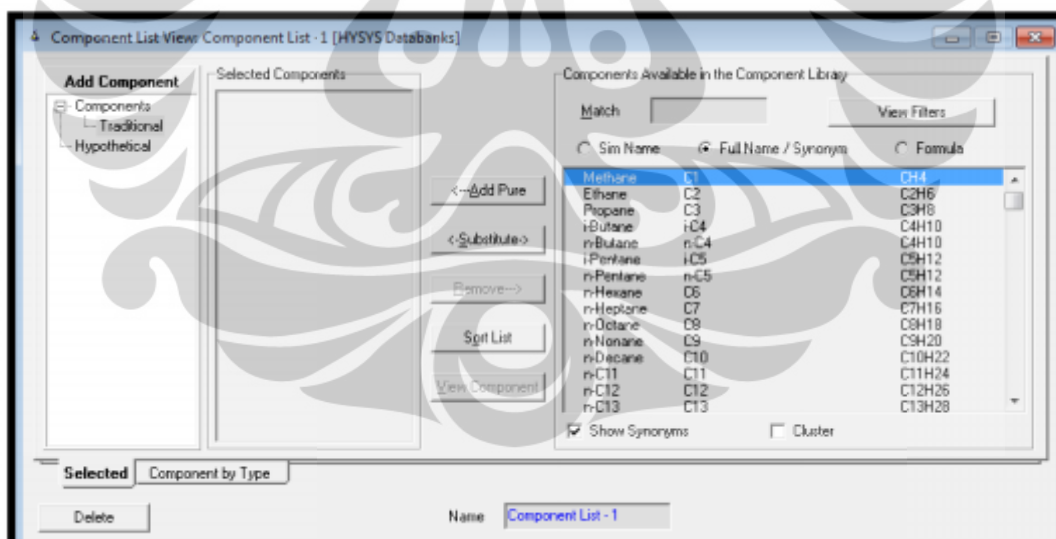
Gambar 3.3. Tampilan *New Case*



Gambar 3.4. Tampilan *Simulation Basis Manager*

b) Membuat rincian komponen

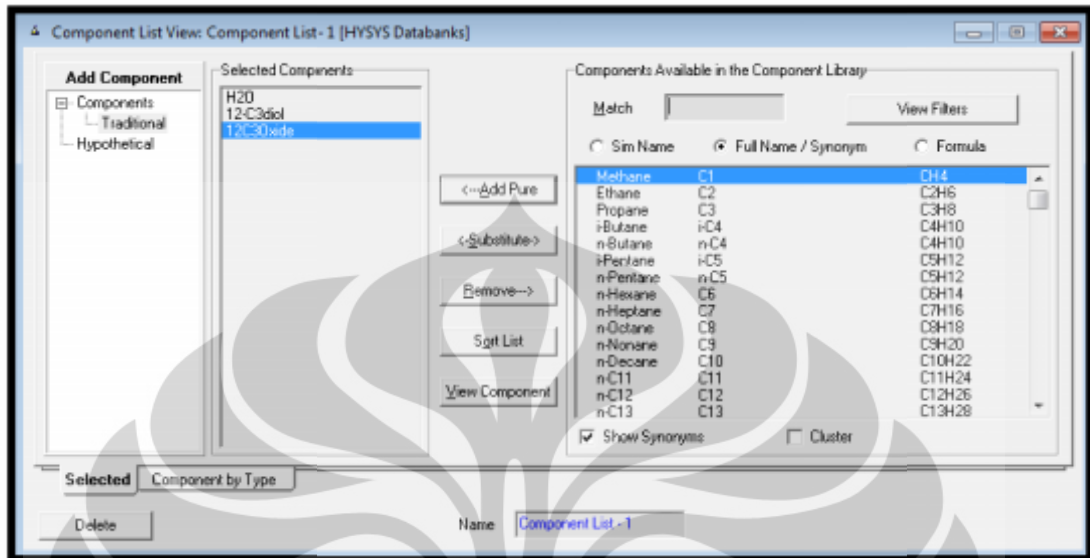
Rincian komponen dibuat dalam *Simulation Basis Manager* dengan cara klik tab *Components*, kemudian klik tombol *Add* dalam *Master Component List* untuk memasukkan komponen yang akan digunakan dalam proses simulasi. Tampilan *Component List View* seperti pada Gambar 3.5 berikut:



Gambar 3.5. Tampilan *Component List View*

Penambahan komponen dapat dilakukan dengan cara menekan *ENTER*, klik tombol *Add Pure* atau melakukan klik dua kali pada komponen yang terpilih. Komponen yang ditambahkan pada *Component List* yaitu : PropyleneOxide (12C3Oxide), Water (H2O) dan PropyleneGlycol (12-C3diol).

Komponen yang telah dipilih akan muncul dalam *Selected Components List* seperti tampilan pada Gambar 3.6 berikut:

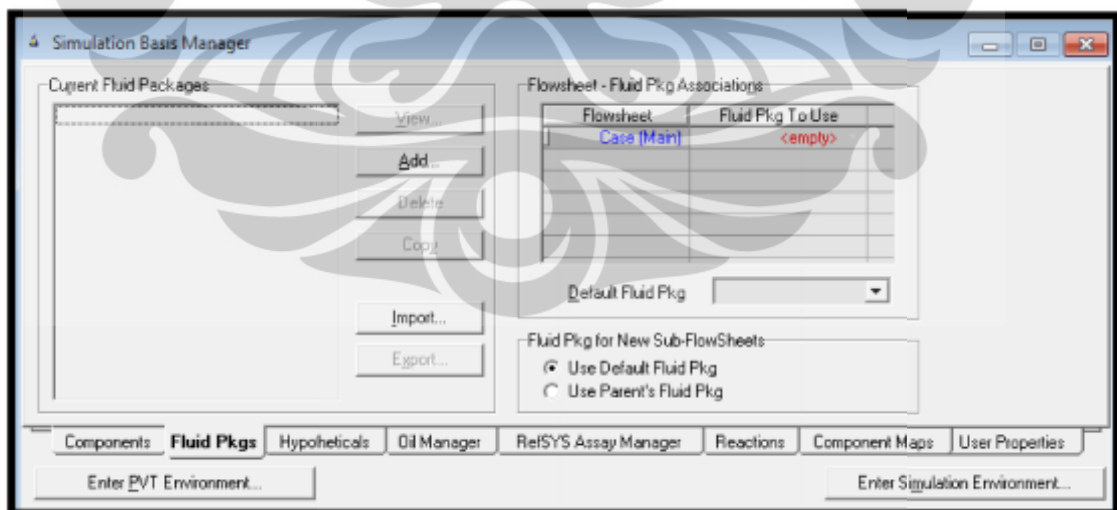


Gambar 3.6. Tampilan *Selected Components List*

c) Membuat *Fluid Package*

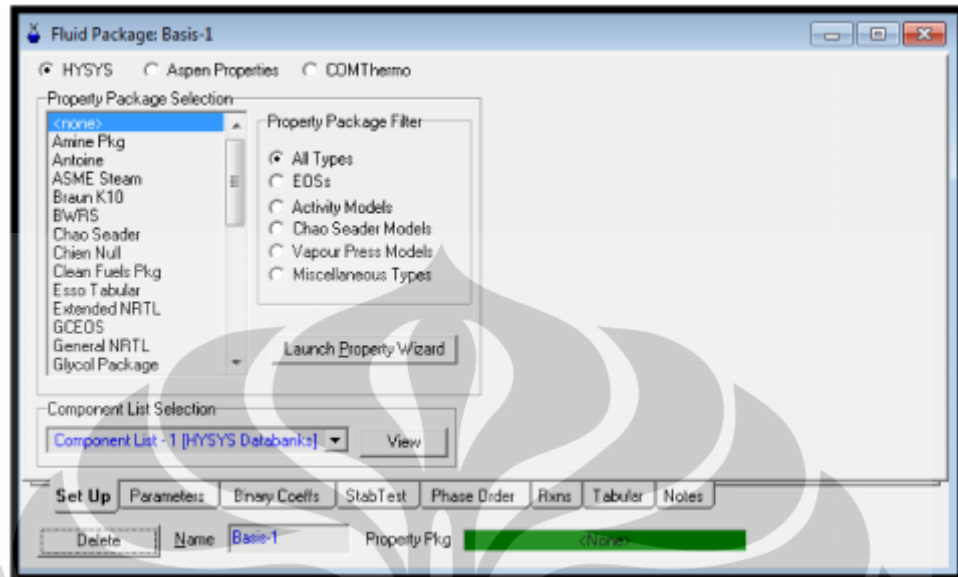
Fluid Package dibuat dalam *Simulation Basis Manager* dengan cara:

- Klik tab *Fluid Pkgs* sehingga tampil Gambar 3.7 seperti berikut:



Gambar 3.7. Tampilan *Simulation Basis Manager* pada tab *Fluid Pkgs*

- Klik tombol *Add*, sehingga tampil Gambar 2.8 seperti berikut:



Gambar 3.8. Tampilan *Fluid Package* pada tab *Set Up*

✓ Tab *Set Up*

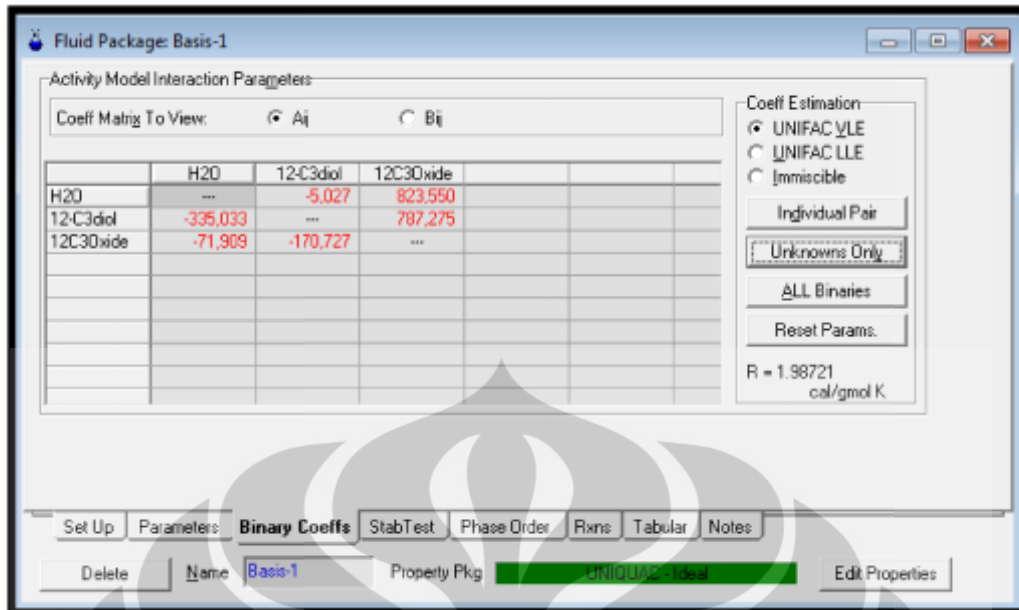
Pada *Property Package Selection* memilih *UNIQUAC* sebagai *Property Package* yang digunakan pada simulasi ini.

✓ Tab *Binary Coeffs*

Pada *Activity Model Interaction Parameters group*, nilai dalam tabel iterasi *Aji* merupakan nilai yang telah dihitung Unisim. Nilai tersebut dapat diubah atau tetap pada nilai tersebut.

Dalam proses simulasi ini, koefisien yang tidak diketahui berada pada tabel *12C3Oxide* atau *12-C3diol*. Koefisien tersebut dapat ditentukan jika terdapat data-data yang sesuai atau ditentukan dengan metode estimasi pada menggunakan *UNIFAC VLE*. Dalam *Coeff Estimation group*, pastikan tombol *UNIFAC VLE* telah dipilih. Kemudian klik tombol *Unknowns Only*.

Tampilan akhir pada tabel *Activity Model Interaction Parameters* untuk koefisien *Aji* seperti pada gambar 3.9 berikut:

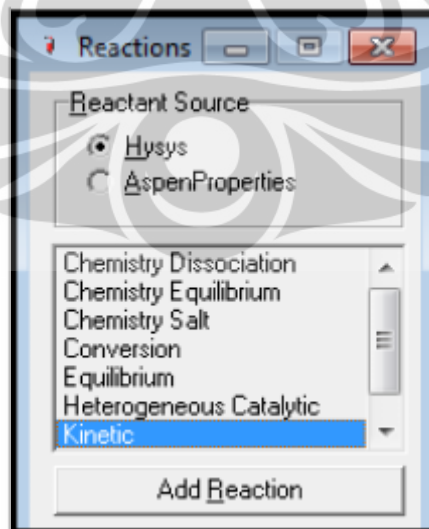


Gambar 3.9. Tampilan *Fluid Package* pada tab *Activity Model Interaction Parameters*

d) Memasukkan reaksi

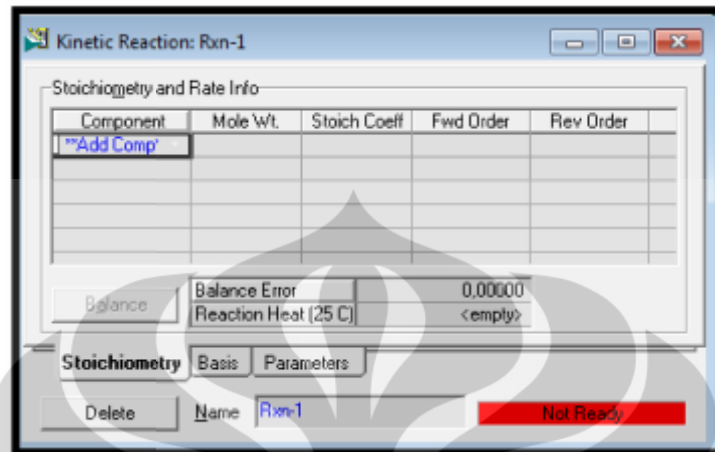
Reaksi yang digunakan pada sistem dimasukkan pada proses simulasi dengan cara sebagai berikut:

- Tampilan dikembalikan pada *Simulation Basis Manager*
- klik tab *Reaction*
- klik *Add Rxn* sehingga muncul tampilan seperti Gambar 3.10 berikut:



Gambar 3.10. Tampilan tab *Reactions*

- memilih tipe reaksi *Kinetic* dan klik tombol *Add Reaction* sehingga muncul tabel *Kinetic Reaction* seperti Gambar 3.11 sebagai berikut:



Gambar 3.11. Tampilan *Kinetic Reaction*

✓ tab *Stoichiometry*

- Pada kolom *Component*, melakukan penambahan komponen yang digunakan dengan cara klik ****Add Comp**** dan memilih komponen yang muncul dalam rincian.
- Pada kolom *Stoich Coeff*, memasukkan nilai koefisien reaksi. Dalam simulasi proses ini terjadi reaksi $H_2O + C_3H_6O \rightarrow C_3H_8O_2$, sehingga koefisien reaksi untuk H_2O dan C_3H_6O bernilai -1 dan $C_3H_8O_2$ bernilai 1.

Tampilan tab *Stoichiometry* setelah dilengkapi data-data tersebut seperti Gambar 3.12 berikut:



Gambar 3.12. Tampilan tab *Stoichiometry*

Nilai negatif pada *Reaction Heat* menunjukkan bahwa reaksi yang terjadi merupakan reaksi eksotermis atau reaksi yang mengeluarkan panas.

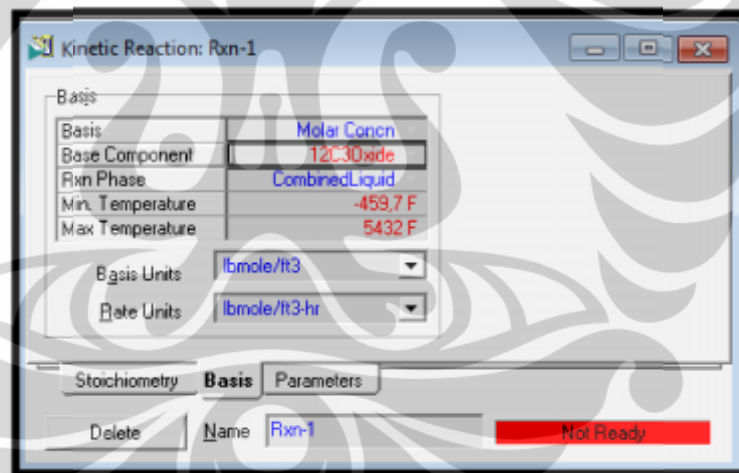
✓ Tab *Basis*

Data yang dimasukkan pada tab *Basis* ditunjukkan pada Tabel 3.1 berikut:

Tabel 3.1. Data untuk tab *Basis*

kolom	
Base Component	12C3Oxide
Rxn Phase	CombinedLiquid

Tampilan tab *Basis* setelah dilengkapi data-data tersebut seperti pada Gambar 3.13 berikut:



Gambar 3.13. Tampilan tab *Basis*

✓ Tab *Parameters*

Memasukkan data parameter pada kolom *Forward Reaction* sebagai berikut:

$$A = 16.96 \times 10^{12} \text{hr}^{-1}$$

$$E = 32,400 \frac{\text{Btu}}{\text{lbmol}}$$

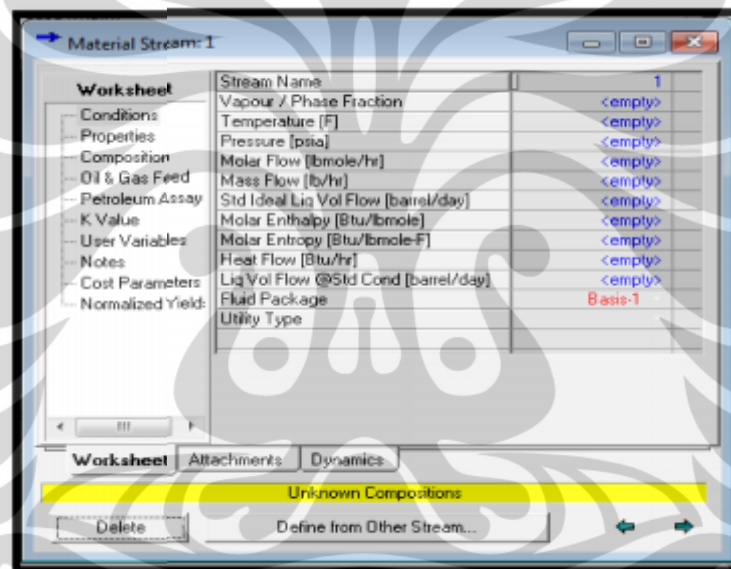
e) Masuk ke dalam lingkungan simulasi proses

Masuk ke dalam lingkungan proses simulasi dilakukan dengan cara klik *Enter Simulation Environment* yang terletak pada bagian kanan bawah *Simulation Basis Manager*.

f) Membuat aliran umpan pada PFD

Aliran umpan pada PFD dibuat dengan cara sebagai berikut:

- Klik gambar *material stream* pada *object pallet*
- Klik dua kali gambar tersebut sehingga muncul tampilan seperti Gambar 3.14 berikut:



Gambar 3.14. Tampilan *Material Stream*

- Membuat aliran umpan dengan ketentuan seperti pada Tabel 3.2 berikut:

Tabel 3.2. Data untuk Aliran Umpan

<i>Worksheet</i>	<i>Kolom</i>	<i>Data yang dimasukkan</i>	
<i>Condition</i>	<i>Stream Name</i>	<i>Propylene Oxide Feed</i>	<i>Water Feed</i>
	<i>Temperature (F)</i>	75	75
	<i>Pressure (psia)</i>	16.17	16.17
	<i>Molar flow (lbmol/hr)</i>	150	-
	<i>Mass flow (lb/hr)</i>	-	11000

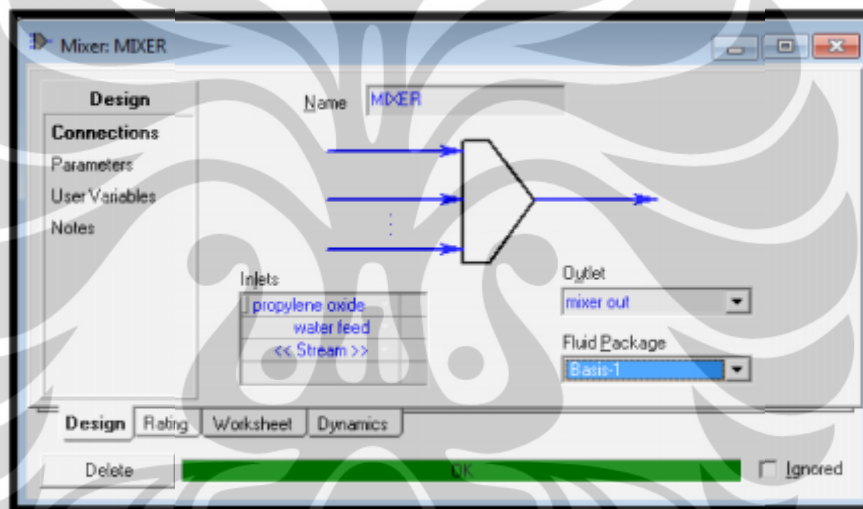
<i>Composition (mole fraction)</i>	12C3Oxide	1	0
	H2O	0	1
	12-C3diol	0	0

g) Membuat Instalasi Unit Operasi

✓ Instalasi Unit Operasi Mixer

Instalasi unit operasi Mixer dilakukan dengan cara sebagai berikut:

- Klik gambar *Mixer* pada *object pallet*, drag kedalam *PFD*
- Klik dua kali pada gambar *Mixer* yang telah berada pada *PFD*, sehingga muncul tampilan seperti Gambar 3.15 berikut:



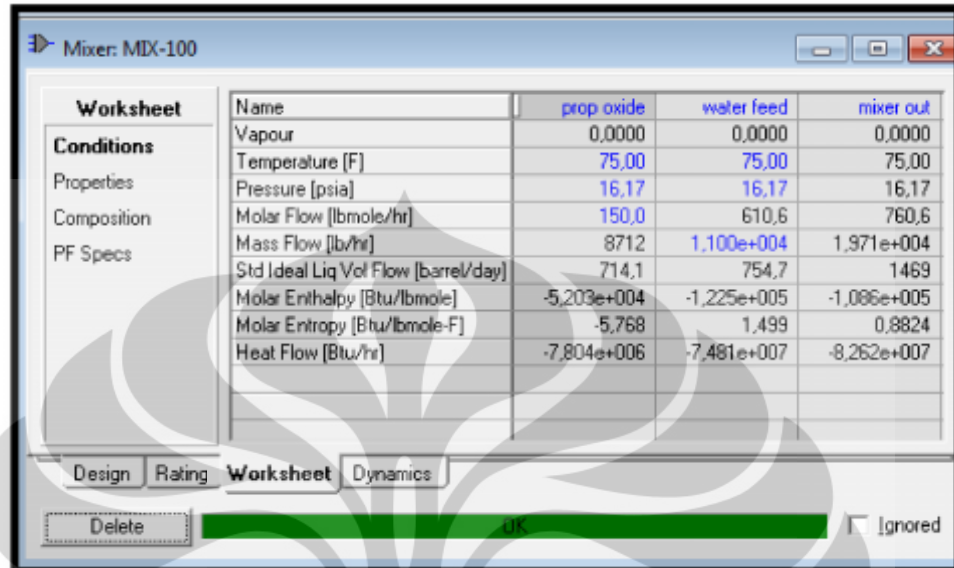
Gambar 3.15. Tampilan *Mixer*

- Ketentuan yang dimasukkan pada Mixer ditunjukkan pada Tabel 3.3 berikut:

Tabel 3.3. Data untuk *Mixer*

Tabs	Kolom	Data yang dimasukkan	
<i>Design</i>	<i>Connection</i>	<i>Name</i>	Mixer
		<i>Inlets</i>	Propylene Oxide Feed, Water Feed
		<i>Outlet</i>	Mixer Out
	<i>Parameters</i>	<i>Set Outlet To Lowest Inlet</i>	

- Tampilan *worksheet Mixer* setelah sistem konfergen ditunjukkan pada Gambar 3.16 berikut:



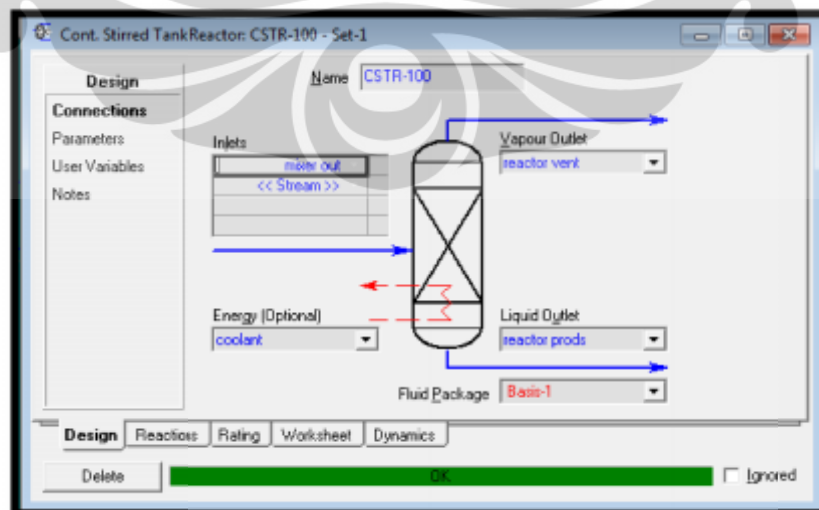
Worksheet	Name	prop oxide	water feed	mixer out
Conditions	Vapour	0,0000	0,0000	0,0000
Properties	Temperature [F]	75,00	75,00	75,00
Properties	Pressure [psia]	16,17	16,17	16,17
Composition	Molar Flow [lbmole/hr]	150,0	610,6	760,6
Composition	Mass Flow [lb/hr]	8712	1,100e+004	1,971e+004
PF Specs	Std Ideal Liq Vol Flow [barrel/day]	714,1	754,7	1469
PF Specs	Molar Enthalpy [Btu/lbmole]	-5,203e+004	-1,225e+005	-1,086e+005
PF Specs	Molar Entropy [Btu/lbmole-F]	-5,768	1,499	0,8824
PF Specs	Heat Flow [Btu/hr]	-7,804e+006	-7,481e+007	-8,262e+007

Gambar 3.16. Tampilan *worksheet Mixer*

- ✓ Instalasi Unit Operasi CSTR

Instalasi unit operasi Mixer dilakukan dengan cara sebagai berikut:

- Klik gambar *CSTR* pada *object pallet*, drag kedalam *PFD*
- Klik dua kali pada gambar *CSTR* yang telah berada pada *PFD*, sehingga muncul tampilan seperti Gambar 3.17 berikut:



Gambar 3.17. Tampilan CSTR

- Ketentuan yang dimasukkan pada Mixer ditunjukkan pada Tabel 3.3 berikut:

Tabel 3.4. Data untuk CSTR

Tabs	Kolom	Data yang dimasukkan	
<i>Design</i>	<i>Connection</i>	<i>Name</i>	CSTR
		<i>Inlets</i>	Mixer out
		<i>Outlet</i>	Vapour outlet = Reactor vent; Liquid outlet = Reactor prods
		<i>Energy</i>	coolant
	<i>Parameters</i>	<i>Volume</i>	280 ft ³
		<i>Liquid volume %</i>	100%
<i>Reaction</i>	<i>Details</i>	<i>Reaction Set</i>	Set-1
		<i>Reaction</i>	Rxn-1
<i>Worksheet</i>	<i>Conditions</i>	<i>Reactor prods</i>	Temperature = 140 °F

- Tampilan *worksheet* CSTR setelah sistem konfergen ditunjukkan pada Gambar 3.18 berikut:

Worksheet	Name	mixer out	reactor prods	reactor vent
Conditions	Vapour	0,0000	0,0000	1,0000
Conditions	Temperature [F]	75,00	140,0	140,0
Properties	Pressure [psia]	16,17	16,17	16,17
Composition	Molar Flow [lbmole/hr]	760,6	618,5	0,0000
Composition	Mass Flow [lb/hr]	1,371e+004	1,371e+004	0,0000
PF Specs	Std Ideal Liq Vol Flow [barrel/day]	1469	1329	0,0000
PF Specs	Molar Enthalpy [Btu/lbmole]	-1,096e+005	-1,409e+005	-7,829e+004
PF Specs	Molar Entropy [Btu/lbmole-F]	0,8624	3,845	29,44
PF Specs	Heat Flow [Btu/hr]	-8,262e+007	-8,714e+007	-0,0000

Gambar 3.18. Tampilan *worksheet* CSTR

Hasil konfersi reaksi dapat dilihat pada unit operasi CSTR, dalam tab *Reaction* (*Result*) seperti tampilan pada Gambar 3.19 berikut ini:

Cont. Stirred Tank Reactor: CSTR-100 - Set-1

Reaction Results Summary

Reaction Extents Reaction Balance

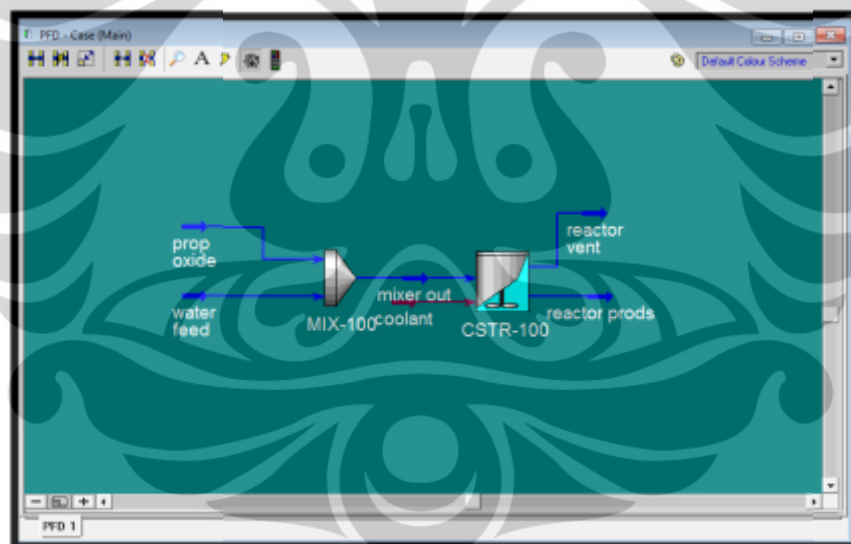
Rxn	Act. % Conv	Base Comp	Rxn Extent
Rxn-1	94.70	12C3Oside	142.0

Design Reactions Rating Worksheet Dynamics

Delete Ignored

Gambar 3.19. Tampilan hasil konversi reaksi pada CSTR

Tampilan PFD simulasi proses dalam kondisi *steady state* ditampilkan pada Gambar 3.20 berikut ini:



Gambar 3.20. Tampilan PFD dalam kondisi *steady state*

3. Merubah simulasi proses dengan kondisi *steady state* ke dalam kondisi dinamik

Perubahan kondisi simulasi proses dari keadaan *steady state* ke dalam bentuk dinamik dilakukan dengan tahapan sebagai berikut:

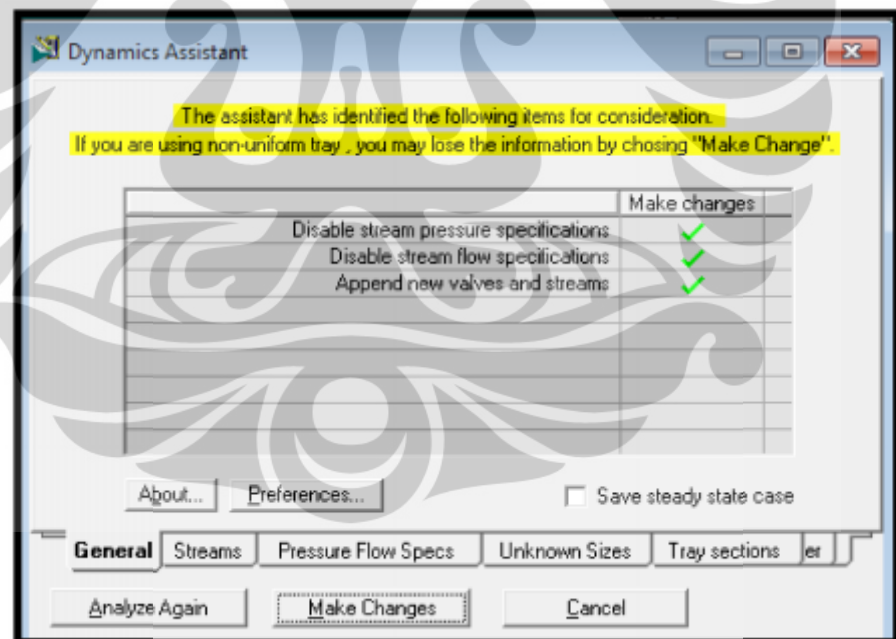
- Menghilangkan spesifikasi tekanan pada aliran *Water Feed*

Penghilangan spesifikasi tekanan dapat dilakukan dengan tahapan sebagai berikut:

- membuka tampilan aliran *Water Feed* dengan cara klik dua kali pada aliran *Water Feed*
- menghapus nilai *Pressure* pada tab *Worksheet*
- menutup tampilan aliran *Water Feed*
- membuka tampilan *Mixer* dengan cara klik dua kali pada unit operasi *Mixer*
- pada tab *Design, Parameter*, mengubah pengaturan *Set Outlet To Lowest Inlet* pada *Automatic pressure Assignment* menjadi *Equalize All*

b) Klik gambar *Dynamics Assistant*

- Tampilan *Dynamics Assistant* pada tab *General* muncul seperti Gambar 3.21 berikut ini:



Gambar 3.21. Dialog *Dynamics Assistant* pada tab *General*

Semua rekomendasi yang diberikan *Dynamics Assistant* akan diimplementasikan walaupun rekomendasi tidak diaktifkan. Pemilihan aktif/tidak aktif rekomendasi dilakukan dengan memberikan ceklis pada kolom *Checkbox OK*.

- Klik tab *Stream*
- Memilih rekomendasi pada sesuai dengan ketentuan yang ditentukan pada Tabel 3.5 berikut:

Tabel 3.5. Pemilihan rekomendasi pada *Dynamics Assistant*

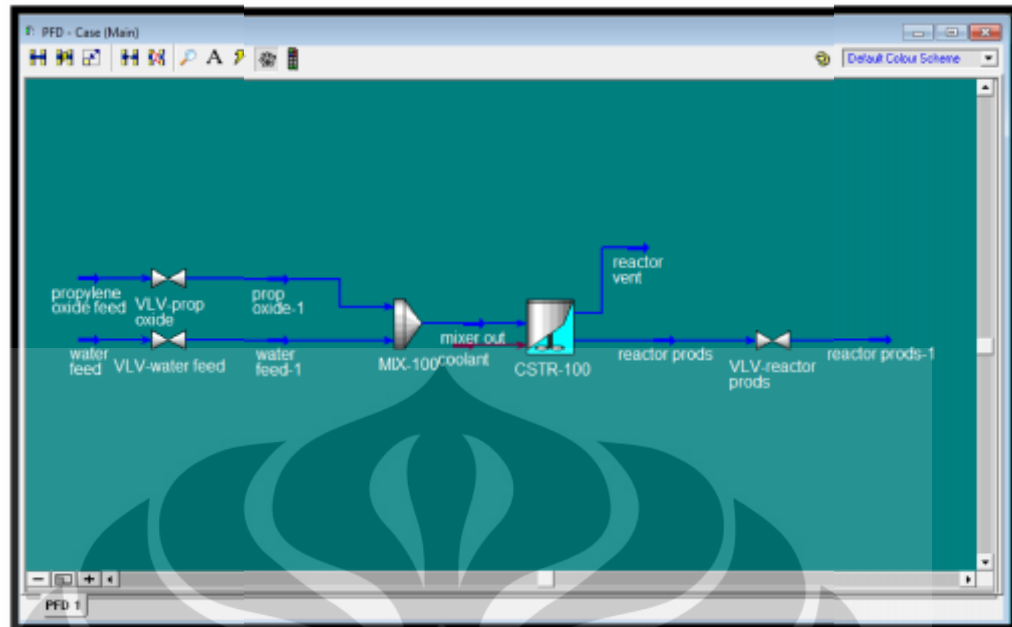
	Rekomendasi	Aliran	OK Checkbox
<i>Pressure Specs</i>	<i>Remove Pressure Specifications</i>	Prop Oxide	<i>Active</i>
<i>Flow Specs</i>	<i>Remove Flow Specifications</i>	Prop Oxide	<i>Active</i>
		Water Feed	<i>Active</i>
<i>Insert Valves</i>	<i>Insert Valves</i>	Prop Oxide	<i>Active</i>
		Reactor Prods	<i>Active</i>
		Reactor Vent	<i>Inactive</i>
		Water Feed	<i>Active</i>

- Klik tombol *Make Changes* satu kali.
Rekomendasi yang diaktifkan secara otomatis diimplementasikan pada simulasi proses.

c) Klik tombol *Dynamic Mode*

Memilih jawaban tidak atau *NO* ketika muncul pertanyaan “Apakah perlu dilakukan perubahan ulang pada item *Dynamic Assistant* sebelum berpindah ke kondisi dinamik?”

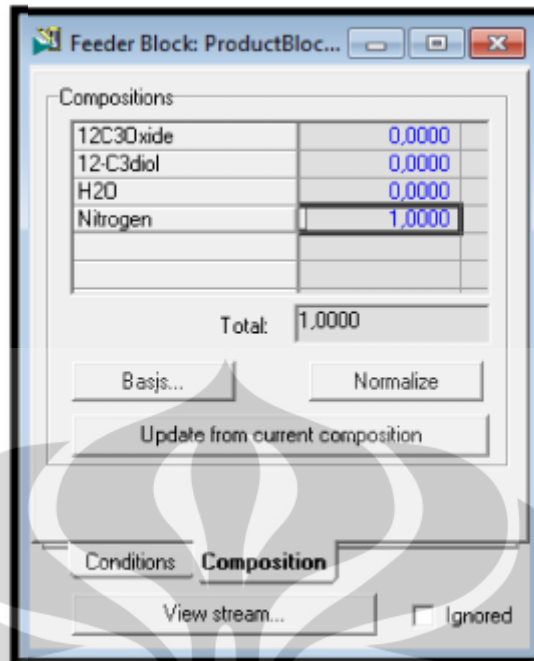
- Membuat spesifikasi tekanan (*Pressure Specification*) pada aliran *Reactor Vent*. Hal ini dilakukan karena rekomendasi untuk memasukkan *valve* pada aliran *Reactor Vent* tidak diaktifkan. Aktivasi dilakukan dengan cara klik dua kali aliran *Reactor Vent* pada PFD. Mengaktifkan *Pressure specification* dalam tab *Dynamics (Specs)* dengan cara klik *Active checkbox*.
- Tampilan PFD simulasi proses dalam kondisi dinamik ditunjukkan pada Gambar 3.22 berikut ini:



Gambar 3.22. Tampilan PFD dalam kondisi dinamik

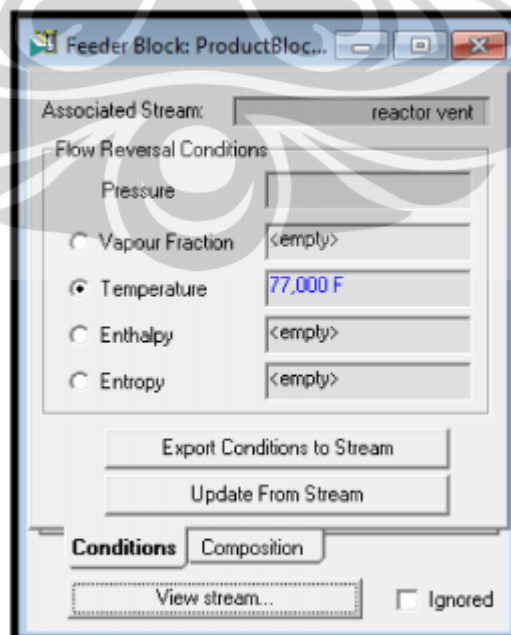
d) Membuat model CSTR yang terbuka ke atmosfer

- Klik *Enter Basis Environment*
- Dalam *Simulation Basis Manager*, klik tab *Components*, kemudian klik tombol *Add* dalam *Master Component List* untuk memasukkan komponen Nitrogen.
- Klik *Return to Simulation Environment* dan jawab *YES* jika tampil pertanyaan
- Pada *PFD*, buka aliran *Reactor Vent* dengan cara klik dua kali pada aliran tersebut
- Klik tombol *Product Block*
- Pada tab *Composition*, memasukkan spesifikasi komposisi pada *Product Block* seperti pada Gambar 3.23 berikut:



Gambar 3.23. Spesifikasi komposisi pada *Product Block*

- Pada tab *Condition*, memasukkan spesifikasi nilai *Temperature* dalam *Flow Reversal Condition* sebesar 77 °F. Kondisi aliran tersebut akan digunakan untuk mengeluarkan aliran Nitrogen murni ketika aliran *Reactor Vent* berbalik arah ke dalam reaktor. Gambar 3.24 menunjukkan tampilan *Product Block* pada tab *Conditions*.



Gambar 3.24. tampilan *Product Block* pada tab *Conditions*

4. Melakukan identifikasi sistem

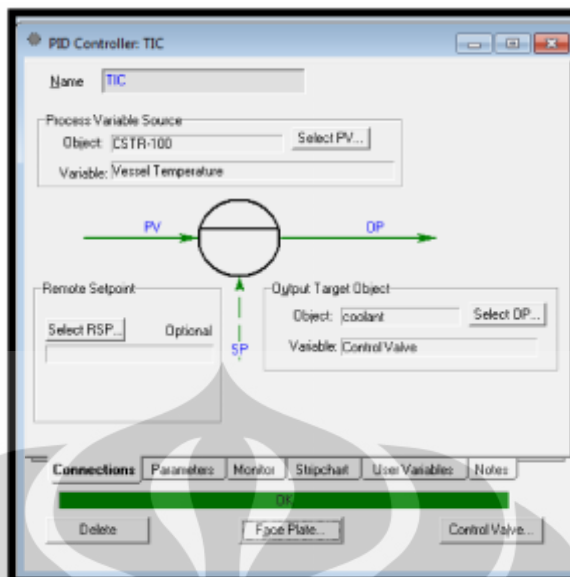
Identifikasi sistem dilakukan dengan tahapan sebagai berikut:

- a. Menambahkan alat pengendali PID pada rangkaian proses.
 - Klik tombol *Integrator Holding* untuk menghentikan simulasi proses
 - Instalasi pengendali PID dengan cara klik gambar *Control Ops* pada *object pallet* dan memilih *PID Controller* dalam pilihan gambar pada *Control Ops*
 - Klik dua kali *PID Controller* dalam PFD untuk menentukan parameter-parameter dalam pengendali dengan ketentuan sebagai berikut:
 - ✓ Pengendali temperatur
 Ketentuan yang dimasukkan pada pengendali temperatur ditunjukkan pada Tabel 3.6 berikut:

Tabel 3.6. Data untuk Pengendali Temperatur

Tab	Kolom	Variabel Yang Dimasukkan
<i>Connection</i>	<i>Name</i>	TC
	<i>Process Variabel</i>	Reaktor, Vessel Temperatur
	<i>Source</i>	
	<i>Output Target Object</i>	Coolant, Control Valve
<i>Parameter</i> (<i>Konfigurasi</i>)	<i>Action</i>	Manual
	<i>PV Minimum</i>	70 °F
	<i>PV Maksimum</i>	300 °F

Tampilan *PID Controller* TIC pada tab *Connections* ditunjukkan Gambar 3.25 berikut ini:



Gambar 3.25. Tampilan *PID Controller TIC* pada tab *Connections*

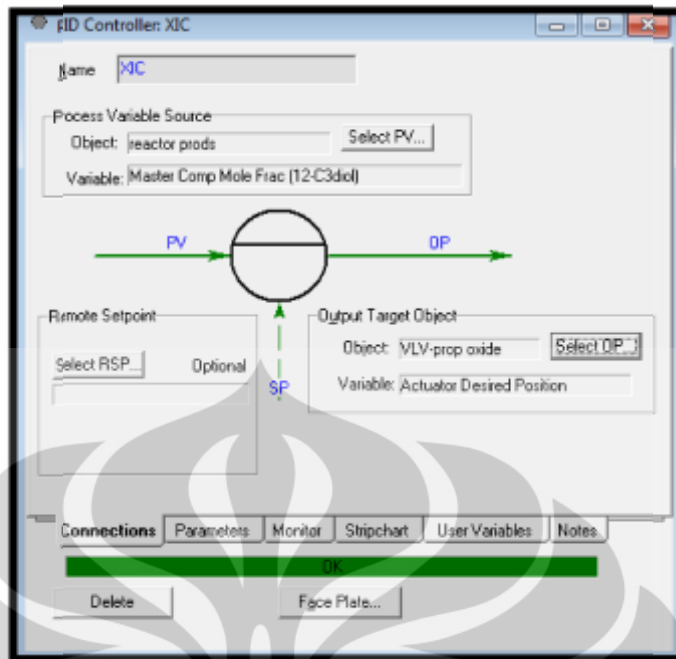
✓ Pengendali laju alir

Ketentuan yang dimasukkan pada pengendali laju alir ditunjukkan pada Tabel 3.7 berikut:

Tabel 3.7. Data untuk Pengendali Konsentrasi Produk

Tab	Kolom	Variabel Yang Dimasukkan
<i>Connection</i>	<i>Name</i>	AC
	<i>Process Variabel Source</i>	Propylene Oxide Feed, Mass Flow
	<i>Output Target Object</i>	VLV-Propylene Oxide Feed, Actuator Desired Position
<i>Parameters (Konfigurasi)</i>	<i>Action</i>	Manual
	<i>PV Minimum</i>	0 lb/hr
	<i>PV Maksimum</i>	18,000 lb/hr

Tampilan *PID Controller XIC* pada tab *Connections* ditunjukkan Gambar 3.26 berikut ini:



Gambar 3.26. Tampilan *PID Controller XIC* pada tab *Connections*

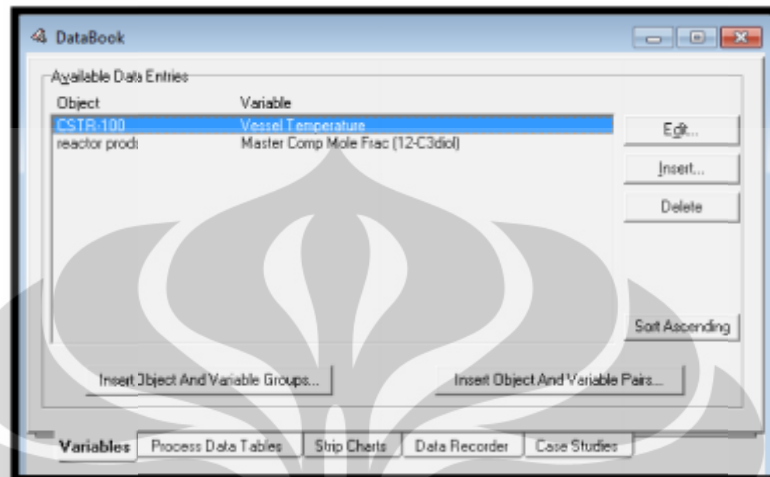
- Klik tombol *Integrator Active* untuk mengaktifkan simulasi proses

b. Melakukan pengawasan dalam kondisi dinamik

Pengawasan dalam kondisi dinamik dapat dilakukan dengan cara membuat grafik atau *Strip Chart*. Grafik tersebut dapat menunjukkan perilaku variabel-variabel yang dikendalikan dan yang dimanipulasi, kondisi aliran umpan dan aliran energi atau variable lain yang dibutuhkan informasinya sehingga modifikasi dalam kondisi dinamik lebih mudah untuk dilakukan. Tahapan pembuatan *Strip Chart* adalah sebagai berikut:

- Membuka *Databook* dengan kombinasi *CTRL + D*
- Tab *Variables*
 - Klik *Insert*
 - Memasukkan *Object* dan *Variable* yang ingin ditampilkan pada *Strip Chart*
 - Klik *OK* sehingga *Object* dan *Variable* yang terpilih akan tampil pada *Available Data Entries*

- Pada penelitian ini *Object* dan *Variable* yang dipilih yaitu *Vessel temperature* pada *CSTR* dan *Master Comp Mole Frac (12-C3diol)* pada *Reactor Prods* seperti Gambar 3.27 berikut ini:



Gambar 3.27. *Object* dan *Variable* pada *Data Book*

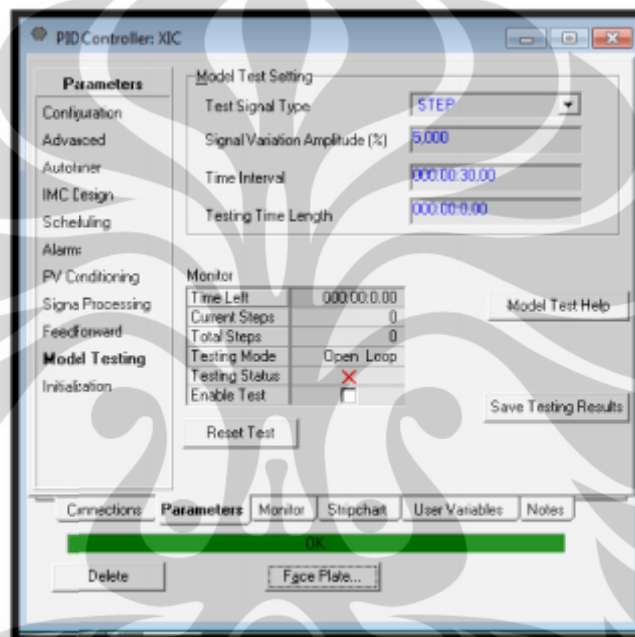
- Tab *Strip Chart*
 - Klik *Checkbox OK* pada kolom *Active* sehingga *Object* dan *Variable* yang terpilih dapat tampil dalam grafik
 - Klik *Strip Chart* pada kolom *View*, sehingga tampil grafik yang menunjukkan *Object* dan *Variable* yang berfungsi terhadap waktu
 - Tampilan pada tab *Strip Chart* ditunjukkan pada Gambar 3.28 berikut ini:



Gambar 3.28. Tampilan *Data Book* pada tab *Strip Chart*

c. Melakukan *Model Testing* pada masing-masing pengendali
Model testing dilakukan pada masing-masing pengendali dengan tahapan sebagai berikut:

- Klik tombol *Integrator Holding* untuk menghentikan simulasi proses
- Klik dua kali *PID Controller* dalam PFD
- Pada tab *Parameters*, klik *Model Testing* sehingga muncul tampilan seperti Gambar 3.29 berikut ini:



Gambar 3.29. Tampilan *Model Testing*

- Menentukan waktu pengujian dengan ketentuan pada Tabel 3.8 berikut ini:

Tabel 3.8. Data untuk *Model Testing*

<i>Model Test Setting</i>	
<i>Test signal type</i>	STEP
<i>Signal Variation Amplitude (%)</i>	5,000
<i>Time interval</i>	10 seconds
<i>Testing time length</i>	250000 seconds

- Klik *Checkbox OK* pada *Enable Test*
- Klik tombol *Integrator Active* untuk mengaktifkan simulasi proses

- *Model Testing* yang dijalankan akan menghasilkan respon dari simulasi proses yang tergambar dalam grafik antara variabel yang di kendalikan (*controlled variabel, CV*) terhadap waktu

d. Membuat model empirik

Model empirik dibuat berdasarkan grafik respon CV hasil *Model Testing* dengan menggunakan metode *II Process Reaction Curve (PRC)*.

e. Memperoleh model empirik untuk digunakan pada simulasi MPC dan PI

5. Membuat simulasi MPC

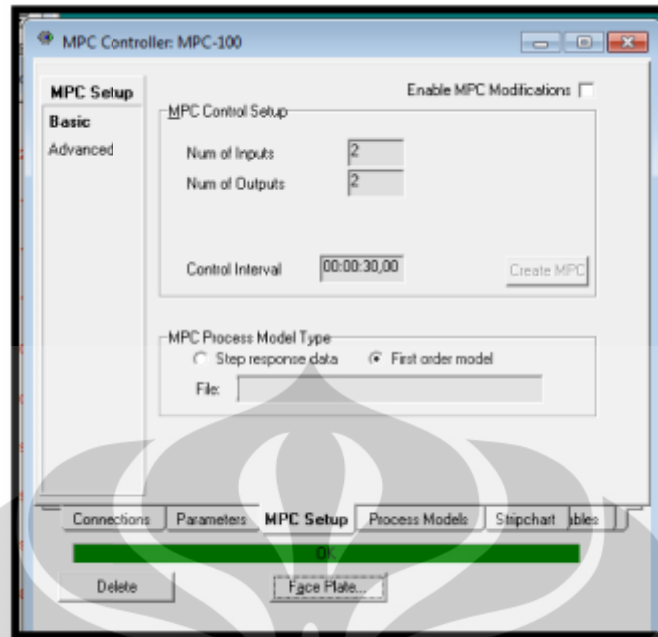
Simulasi MPC dilakukan dengan cara sebagai berikut:

- Klik tombol *Integrator Holding* untuk menghentikan simulasi proses
- Menginstalasi pengendali MPC pada PFD dengan cara klik gambar *Control Ops* pada *object pallet* dan memilih *MPC Controller* dalam pilihan gambar pada *Control Ops*
- Klik dua kali *MPC Controller* dalam PFD untuk menentukan parameter-parameter dalam pengendali

✓ tab *MPC Setup*

- klik *checkbox OK* pada *Enable MPC Modification*
- memasukkan *Num of Inputs* dan *Num of Outputs*. Untuk MPC *singlevariable* nilai yang dimasukkan masing-masing sebesar 1. Sedangkan untuk MPC *multivariable* dimasukkan nilai 2. Nilai 2 menunjukkan sistem yang digunakan merupakan MIMO berukuran 2x2.
- memasukkan *Control Interval* sebesar 30 seconds
- memilih *First order model* pada *MPC Process Model Type*
- klik tombol *Create MPC*

Tampilan pada tab *MPC Setup* ditunjukkan pada Gambar 3.30 berikut:



Gambar 3.30. Tampilan *MPC Controller* pada tab *MPC Setup*

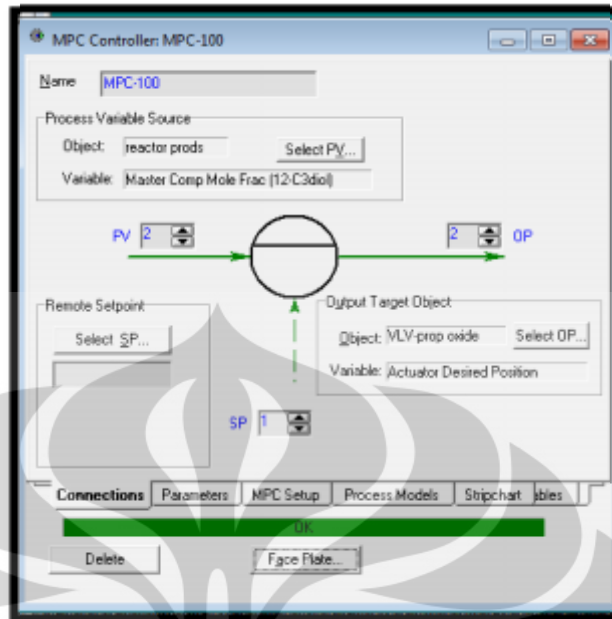
✓ tab *Connections*

- memasukkan nilai 2 pada kotak *PV*, *OP* dan *SP*
- menentukan *Object* dan *Variabel* yang akan dikendalikan dan dimanipulasi dengan ketentuan seperti pada Tabel 3.9 berikut ini:

Tabel 3.9. Data untuk *MPC Controller* pada tab *Connections*

Kotak PV, OP dan SP	<i>Object</i> dan <i>Variable</i>	<i>Process Variable Source</i>	<i>Output Target Object</i>
1	<i>Object</i>	CSTR	Coolant
	<i>Variable</i>	Vessel temperature	Control Valve
2	<i>Object</i>	reactor prods	VLV-propylene oxide
	<i>Variable</i>	Master Comp Mole Frac (12-3Cdiol)	Actuator Desired Position

Tampilan pada tab *Connections* ditunjukkan pada Gambar 3.31 berikut:

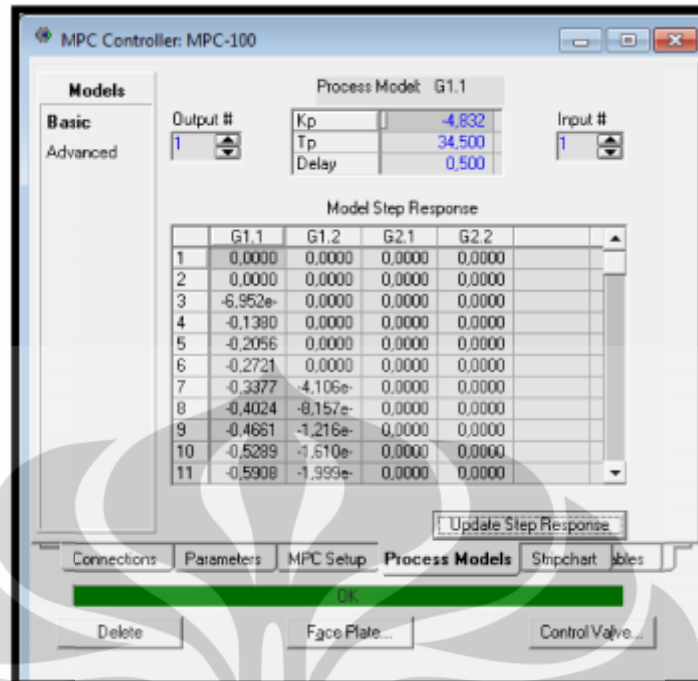


Gambar 3.31. Tampilan *MPC Controller* pada tab *Connections*

✓ tab *Process Models*

- memasukkan nilai K_p , T_p dan $Delay$ pada kolom *Process model* berdasarkan model empirik hasil identifikasi sistem sesuai dengan nomor *Output* dan *Input*
- klik *Update Step Response*

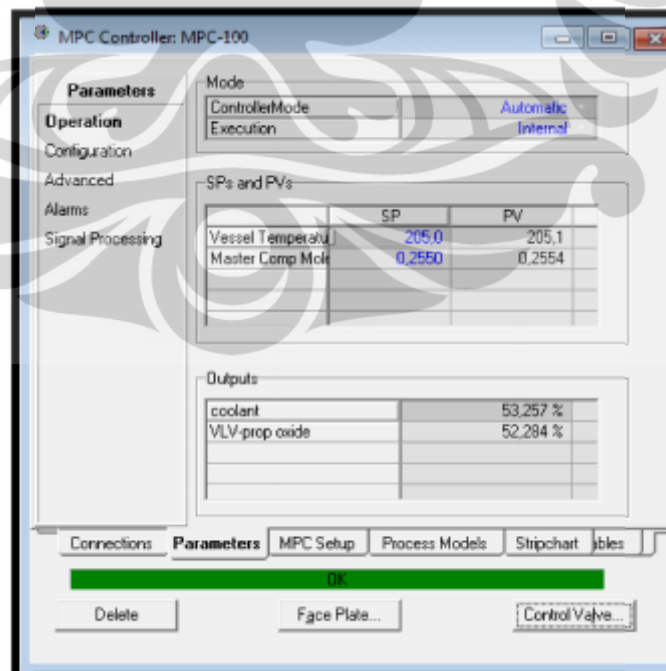
Tampilan pada tab *Process Models* setelah dimasukkan model empirik ditunjukkan pada Gambar 3.32 berikut:



Gambar 3.32. Tampilan MPC Controller pada tab *Process Models*

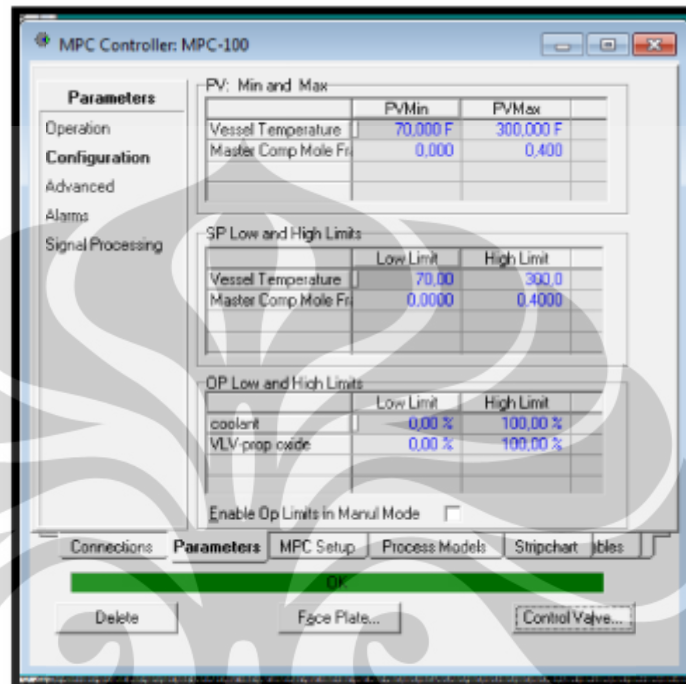
✓ tab *Parameters*

- memasukkan nilai *SP* pada kolom *Operations* berdasarkan *set point* yang diinginkan. Tampilannya ditunjukkan pada Gambar 3.32 berikut:



Gambar 3.33. Tampilan MPC Controller pada tab *Parameters (Operations)*

- memasukkan nilai *PV Min* dan *PV Max* pada kolom *Configuration* pada masing-masing variabel yang dikendalikan. Tampilannya ditunjukkan pada Gambar 3.34 berikut:

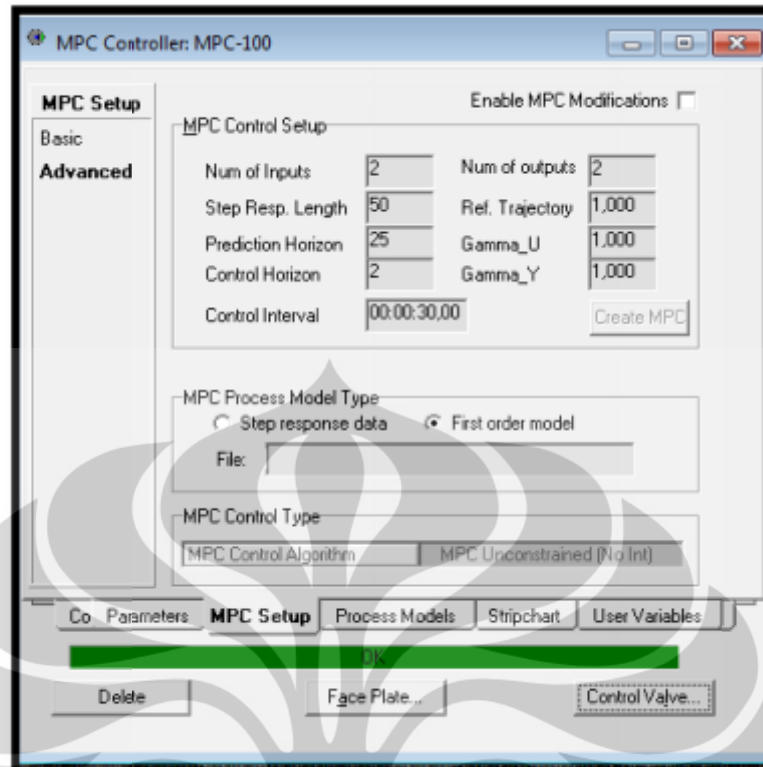


Gambar 3.34. Tampilan *MPC Controller* pada tab *Parameters (Configuration)*

6. Melakukan Tuning MPC

Tuning dilakukan pada parameter MPC, yaitu T (waktu sampel), P (*prediction horizon*), dan M (*control horizon*). Dalam Unisim, tuning dilakukan dengan memasukkan nilai parameter yang telah dihitung dengan pendekatan strategi tuning DMC non-adaptif (Dougherty, 2003a) pada tab *MPC Setup (Advanced)* atau nilai parameter yang diperoleh dengan *trial error*.

Tampilan pada tab *MPC Setup (Advanced)* ditunjukkan pada Gambar 3.35 berikut:



Gambar 3.35. Tampilan *MPC Controller* pada tab *MPC Setup*

3.4.2. Prosedur Pengambilan Sampel

Sampel yang akan diambil dan dianalisis pada penelitian ini adalah grafik respon variabel yang dikendalikan (*controlled variable-CV*). Grafik respon variabel yang dikendalikan diperoleh setelah menjalankan program MPC pada Unisim dan dilakukan tuning pada masing-masing pengendali.

3.4.3 Prosedur Analisis

Prosedur analisa pada sampel dilakukan dengan cara menghitung nilai *Integral Absolute Error* (IAE) dalam grafik respon variabel yang dikendalikan (*controlled variable-CV*) dengan pengendali MPC yang telah dijalankan dalam Unisim. Nilai IAE pada proses dengan pengendali MPC dibandingkan dengan IAE pada proses dengan pengendali PI. Pengendali dengan nilai IAE yang lebih kecil merupakan pengendali dengan kinerja yang lebih baik.

3.4.4 Prosedur Perhitungan

1. Membuat model empirik

Salah satu metode identifikasi yang paling banyak digunakan untuk mengidentifikasi model empirik yang dinamik pada suatu proses adalah *Process Reaction Curve* (PRC). Dalam PRC, dapat ditentukan parameter-parameter (*dead time*, *time constant*, dan *damping coefficient*) yang sesuai dengan data step respon secara eksperimen. Proses yang diidentifikasi merupakan sistem *openloop*, namun pada pengujian secara eksperimen, sistem *closedloop* juga bisa diidentifikasi.

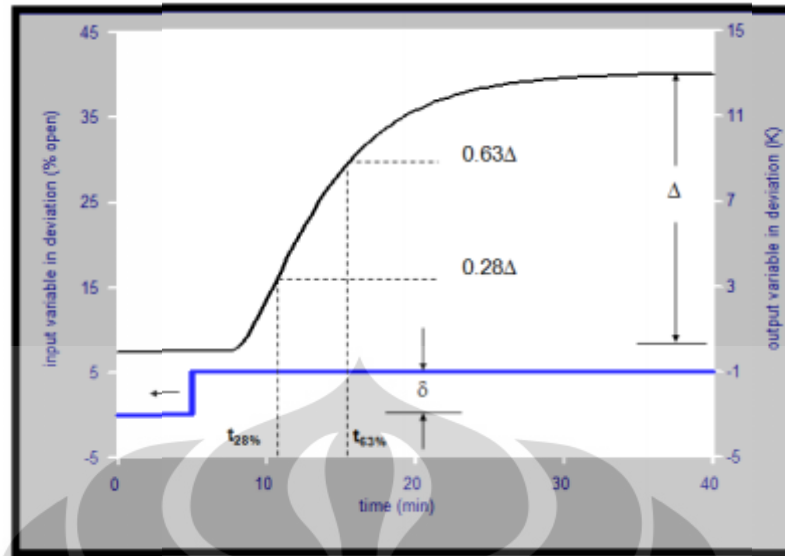
Tahapan dalam *Process reaction curve* meliputi:

1. Pencapaian proses pada kondisi *steady state*,
2. Menunjukkan perubahan *single step* dalam variabel masukan,
3. Mencatat respon masukan dan keluaran sampai proses mencapai kondisi *steady state* nya,
4. Menunjukkan perhitungan secara grafik untuk *process reaction curve*,

Perhitungan grafis dilakukan untuk menentukan parameter pada model *first-order-with-dead-time* (FOPDT). Bentuk model ditunjukkan pada persamaan (2.31) dengan $X(s)$ sebagai masukan dan $Y(s)$ sebagai keluaran.

$$\frac{Y(s)}{X(s)} = \frac{K_p e^{-\theta s}}{\tau s + 1} \quad (3.1)$$

Perhitungan PRC menggunakan Metode II untuk perhitungan grafik PRC ditunjukkan pada Gambar 3.36.



Gambar 3.36. Process Reaction Curve (Marlin, 2000)

Nilai yang ditentukan dari grafik adalah perubahan masukan/*input* (δ), perubahan *steady state* dalam keluran/*outout* (Δ), dan waktu yang dibutuhkan keluaran untuk mencapai 28% dan 63% dari nilai akhirnya. Nilai yang diperoleh dari grafik dapat digunakan untuk menghitung parameter model yang dirumuskan pada persamaan (2.32) berikut ini:

$$Kp = \frac{\Delta}{\delta}$$

$$t_{28\%} = \theta + \frac{\tau}{3} \quad t_{63\%} = \theta + \tau \quad (3.2)$$

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

(Marlin, 2000)

2. Tuning parameter MPC

Parameter MPC dihitung dengan pendekatan strategi tuning DMC non-adaptif (Dougherty, 2003a) sebagai berikut:

- a) Pendekatan dinamika proses output kontroler untuk pasangan-pasangan variabel proses terukur dengan model FOPDT:

$$\frac{y_r(s)}{u_s(s)} = \frac{K_{rs}e^{-\theta_{rs}s}}{\tau_{rs}s+1} \quad (3.3)$$

$$(\tau = 1,2, \dots, R; s = 1,2, \dots, S)$$

b) Memilih waktu sampel sedekat mungkin dengan:

$$T_{rs} = \text{Max}(0.1\tau_{rs}, 0.5\theta_{rs}), \quad (r = 1, 2, \dots, R; s = 1, 2, \dots, S) \quad (3.4)$$

$$T = \text{Min}(T_{rs}) \quad (3.5)$$

c) Menghitung P (*prediction horizon*):

$$P = \text{Max}\left(\frac{5\tau_{rs}}{T} + k_{rs}\right) \quad (3.6)$$

$$\text{dimana } k_{rs} = \left(\frac{\theta_{rs}}{T} + 1\right), \quad (r = 1, 2, \dots, R; s = 1, 2, \dots, S) \quad (3.7)$$

d) Menghitung M (*control horizon*):

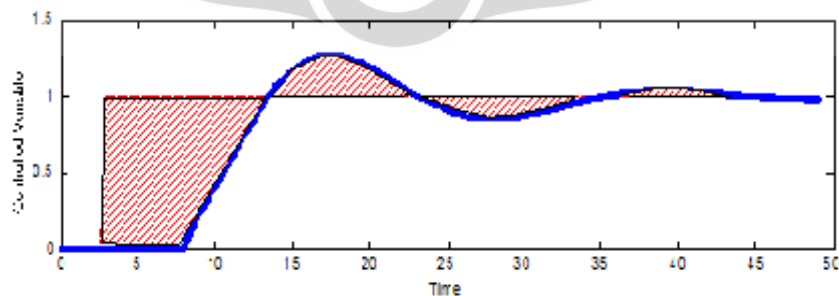
$$M = \text{Max}\left(\frac{\tau_{rs}}{T} + k_{rs}\right), \quad (r = 1, 2, \dots, R; s = 1, 2, \dots, S) \quad (3.8)$$

3. Menghitung *Integral Absolute Error*

Integral Absolute Error (IAE) dihitung dengan menggunakan persamaan:

$$IAE = \int |SP(t) - CV(t)| dt \quad (3.9)$$

Pada grafik respon variabel yang dikendalikan (*controlled variable-CV*), IAE adalah luas daerah absolut dari selisih luas grafik *set point* dengan luas grafik respon CV. Semakin kecil IAE maka CV semakin mendekati *set point*, sehingga mengindikasikan pengendali yang digunakan semakin bagus performanya. Gambar 3.37 menunjukkan contoh grafik respon CV dengan daerah yang diarsir merah adalah besar *Integral Absolute Error*.



Gambar 3.37. Grafik Respon CV dengan IAE

Perhitungan luas dibawah grafik dilakukan dengan menggunakan formula Integrasi Newton-Cotes seperti ditampilkan pada Tabel 3.10.

Tabel 3.10. Formula Integrasi Newton-Cotes

Bias (n)	titik	nama	Formula
1	2	Trapesium	$(b-a) \frac{f(x_0) + f(x_1)}{2}$
2	3	1/3 Simpson	$(b-a) \frac{f(x_0) + 4f(x_1) + f(x_2)}{6}$
3	4	3/8 Simpson	$(b-a) \frac{f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)}{8}$
4	5	Boole	$(b-a) \frac{7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)}{90}$



BAB 4

HASIL DAN PEMBAHASAN

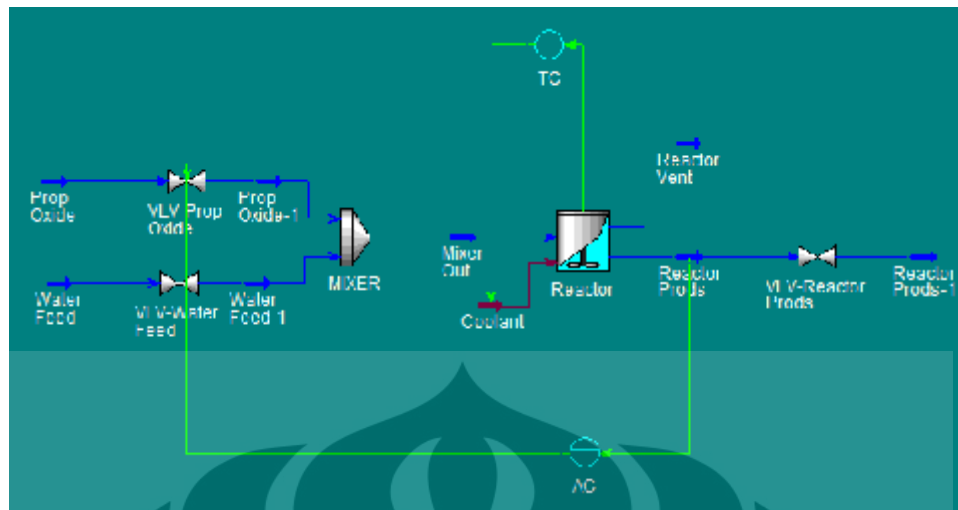
4.1 Identifikasi sistem

Sistem *Continuous Stirred Tank Reactor* (CSTR) yang digunakan sebagai sistem dalam penelitian ini mempunyai dua variabel yang dikendalikan yaitu temperatur reaktor dan komposisi produk utama berupa propylen glycol. Sehingga terdapat dua variabel keluaran pengendali (CO) dan dua variabel proses terukur (PV). Variabel keluaran pengendali berupa laju alir air pendingin atau *coolant* (Fc) dan laju alir umpan (propylene oxide) – (Fpo). Sedangkan variabel proses terukur berupa temperatur reaktor (T) dan konsentrasi produk (Xpg).

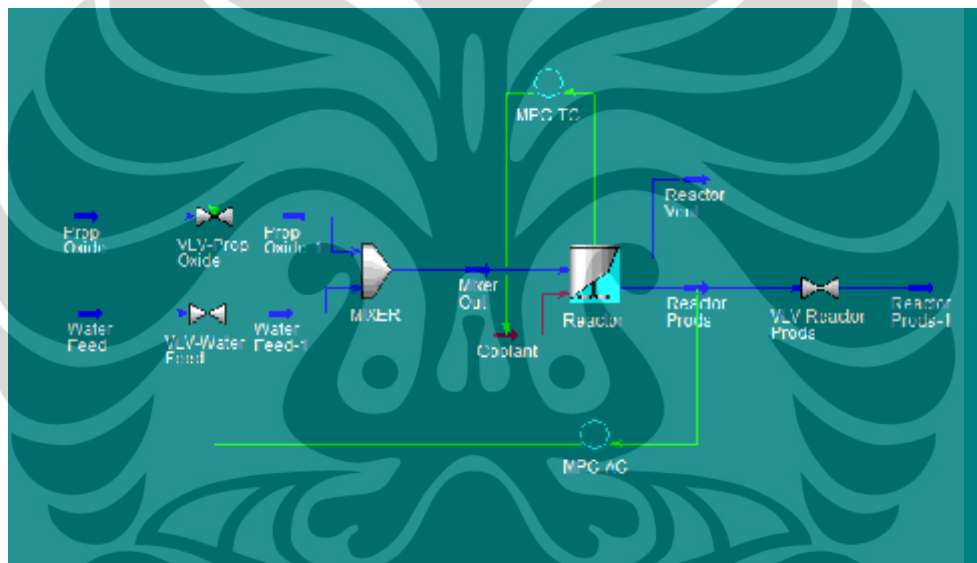
Pada pengendali PI terdapat dua buah pengendali yang dipasang dalam sistem CSTR untuk mengendalikan temperatur dan komposisi produk. Seperti yang ditampilkan pada Gambar 4.1, laju alir *coolant* mengendalikan temperatur reaktor (Fc mengendalikan T) dan laju alir umpan propylene oxide mengendalikan besaran perolehan konsentrasi produk propylene glycol (Fpo mengendalikan Cpg).

Sistem pengendalian pada MPC *singlevariable* sama dengan pengendalian PI, hanya berbeda pada penggunaan jenis pengendali saja. Hal tersebut digambarkan pada Gambar 4.2 yang menunjukkan rangkaian pengendali MPC *singlevariable* pada sistem CSTR. MPC-AC menunjukkan MPC yang mengendalikan konsentrasi produk dengan manipulasi laju alir umpan propylene oxide, dan MPC-TC menunjukkan MPC yang mengendalikan temperatur reaktor dengan manipulasi laju alir *coolant*.

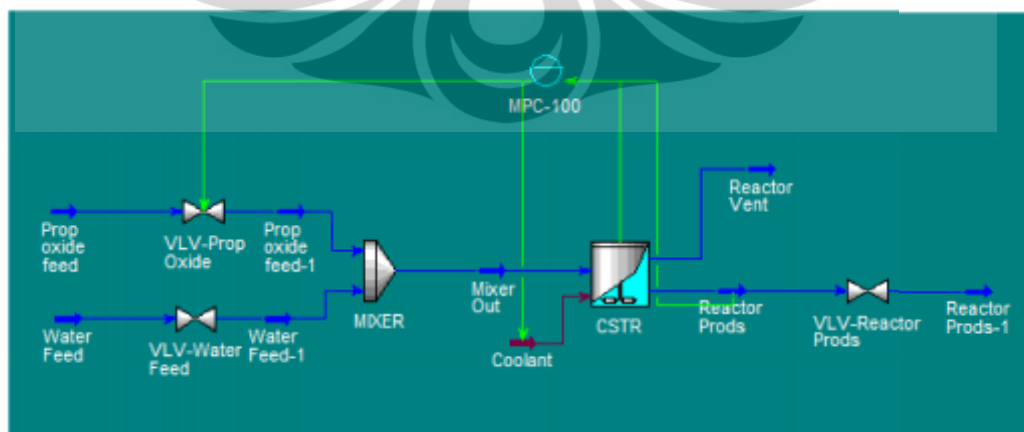
Lain halnya dengan MPC *singlevariable*, pengendalian MPC *multivariable* ditunjukkan dengan satu pengendalian MPC-100 yang berukuran $[2 \times 2]$ pada Gambar 4.3 sesuai dengan jumlah variabel yang dikendalikan.



Gambar 4.1. Tampilan PFD dengan pengendalian PID pada CSTR



Gambar 4.2. Tampilan PFD dengan pengendalian MPC *singlevariable* pada CSTR



Gambar 4.3. Tampilan PFD dengan pengendalian MPC *multivariable* pada CSTR

Pada sistem yang digunakan, dengan adanya interaksi antara kedua variabel yang dikendalikan maka perubahan pada salah satu keluaran pengendali akan mempengaruhi variabel proses terukur. Sehingga identifikasi sistem dilakukan dengan melakukan perubahan pada salah satu variabel keluaran pengendali sedangkan variabel lain dalam keadaan tetap. Contohnya, perubahan dilakukan pada besaran F_c yang masuk ke CSTR sedangkan F_{po} tetap pada kondisi yang sama, begitu sebaliknya.

Variabel proses terukur (T dan X_{pg}) akan berubah dengan adanya perubahan pada salah satu keluaran pengendali sehingga dihasilkan kurva reaksi proses (*Process Reaction Curve/PRC*). Berdasarkan PRC yang dihasilkan, dengan menggunakan persamaan 3.1, dapat diperoleh parameter FOPDT yang mewakili perilaku sistem.

Perubahan laju alir *coolant* menghasilkan respon dari temperatur dalam FOPDT G 1.1, dan dari konsentrasi produk propylene glycol dalam G 1.2. Sedangkan perubahan laju alir umpan propylene oxide menghasilkan respon dari temperatur dalam FOPDT G 2.1, dan dari konsentrasi produk propylene glycol dalam G 2.2. Persamaan model tersebut dalam bentuk matriks berukuran 2 x 2 secara umum adalah sebagai berikut :

$$\begin{bmatrix} T \\ X_{pg} \end{bmatrix} = \begin{bmatrix} G_{1.1} & G_{1.2} \\ G_{2.1} & G_{2.2} \end{bmatrix} \begin{bmatrix} F_c \\ F_{po} \end{bmatrix} \quad (4.1)$$

dengan definisi G_{rs} pada persamaan 3.1.

PRC secara *default* dapat diperoleh dari simulasi dalam perangkat Unisim dengan perintah *model testing* yang terdapat pada pengendali PI, seperti telah dipaparkan cara pengoperasiannya pada BAB 3.

Pada penelitian ini dilakukan perubahan keluaran pengendali yang divariasikan pada nilai tertentu, sehingga dihasilkan variasi model. Model tersebut mempunyai spesifikasi bukaan valve pada kondisi awal dan akhir yang ditampilkan pada Tabel 4.1.

Tabel 4.1. Spesifikasi bukaan valve pada setiap model

Model	bukaan valve awal (%)	bukaan valve akhir (%)
A	50	52,5
B	46	53
C	52,8	53,8
D	52,1	53,9
E	50	54,5
F	45,8	56

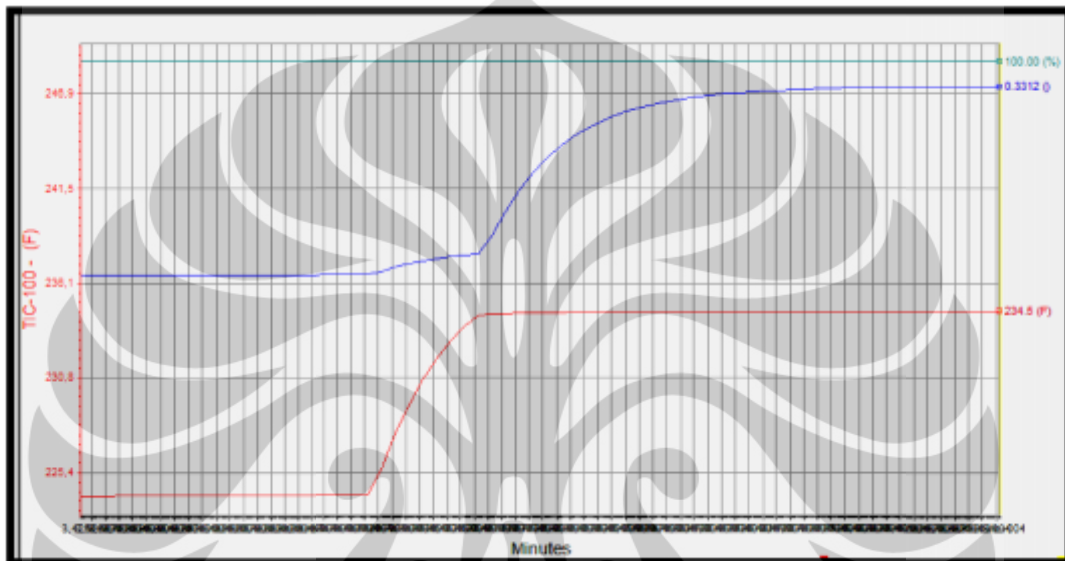
Model A merupakan *model testing default* sebagaimana telah dipaparkan dalam BAB 3. Model B dibuat berdasarkan *model testing* dengan perbedaan bukaan valve sebesar 7% dari kondisi awalnya. Penentuan bukaan valve awal pada *model testing* ini dilakukan dengan mencoba kondisi bukaan valve tertentu sampai diperoleh kondisi sistem yang stabil. Pada kondisi bukaan 40% dan 44%, sistem menjadi tidak stabil ditandai dengan tidak adanya produk yang terbentuk dan temperatur reaktor yang turun melebihi batas CV yang ditentukan. Bukaan valve yang masih dapat membuat sistem stabil adalah pada bukaan 45%, sehingga model B ini dibuat lebih besar dari 45% yaitu pada kondisi awal 46%. Grafik respon sistem terhadap bukaan valve awal ditampilkan pada Lampiran A.

Model C, D, E dan F dibuat dengan menentukan *set point* konsentrasi produk propylene glycol terlebih dahulu, yaitu sebesar 0,37. Nilai ini didasarkan hasil *trial error* pada simulasi sistem CSTR, karena pada kondisi pencapaian *set point* > 0,38 sistem menjadi lebih tidak stabil. Kemudian ditentukan kondisi awal konsentrasi produk hingga mempunyai perbedaan 2,5% , 5%, 10% dan 20% terhadap *set point* akhir yang telah ditentukan. Jadi persentase yang dimaksud tidak berdasarkan bukaan valve tapi berdasarkan perbedaan *set point* konsentrasi produk awal dan akhir.

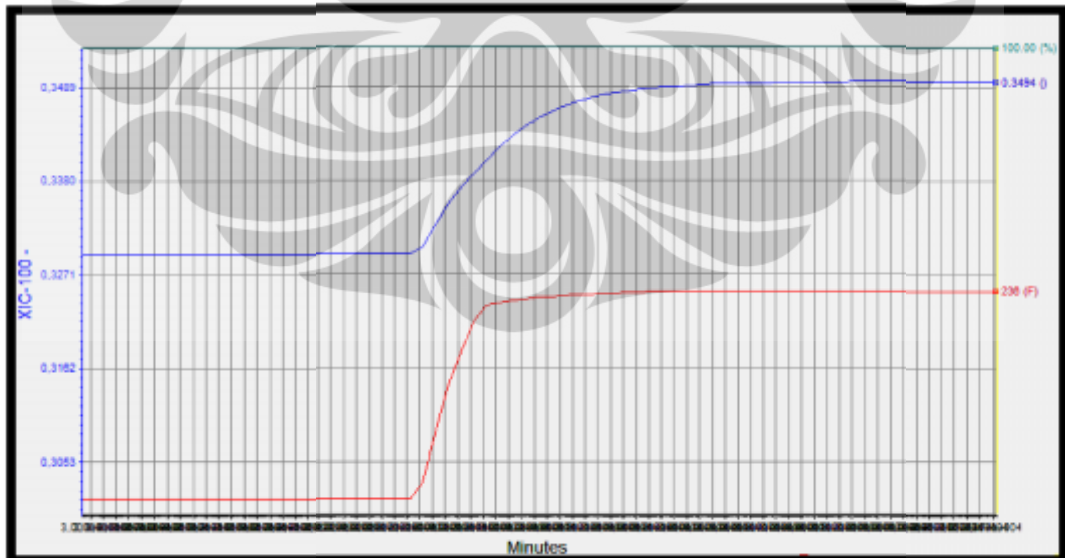
Hal ini dilakukan karena pada penelitian dalam MPC *multivariable* sebelumnya (Dougherty, 2002), variasi model dibuat dengan memvariasikan bukaan valve dari kondisi awal valve tertutup, seperti variasi perbedaan 20%, 50% dan 80% bukaan valve. Namun pada Unisim dalam sistem CSTR, variasi tersebut tidak dapat dilakukan karena sistem CSTR dibuat secara berurutan dari kondisi *steady state* sehingga ketika diubah dalam dinamik sistem akan mempunyai batasan bukaan valve awal.

Hasil *model testing* pada Unisim berupa PRC hanya ditampilkan untuk model A pada Gambar 4.4 dan Gambar 4.5, sedangkan PRC pada model lain ditampilkan pada Lampiran A.

Parameter FOPDT hasil perhitungan secara manual pada setiap model dari PRC ditampilkan pada Tabel 4.2. Perhitungan FOPDT untuk model A dijadikan sebagai contoh dan ditampilkan pada Lampiran B.



Gambar 4.4. PRC Pengaruh Perubahan F_c Untuk Model A



Gambar 4.5. PRC Pengaruh Perubahan F_{po} Untuk Model A

Tabel 4.2. Parameter FOPDT pada setiap model

Parameter		Kp	τ_p	Θ
MODEL A	G 1.1	4,118	34,5	0,5
	G 1.2	0,0008	45	53
	G 2.1	4,759	22,5	7,5
	G 2.2	0,008	57	6
MODEL B	G 1.1	6,232344	45	10
	G 1.2	0,000628	25,5	9,5
	G 2.1	8,299731	45	10
	G 2.2	0,00727	52,5	7,5
MODEL C	G 1.1	0,279	54	11
	G 1.2	0,0031	51	11
	G 2.1	0,971	46,5	5,5
	G 2.2	0,01065	58,5	1,5
MODEL D	G 1.1	0,263333	55,5	4,5
	G 1.2	0,002889	57	1
	G 2.1	0,946111	57	13
	G 2.2	0,010444	60	10
MODEL E	G 1.1	2,439778	19,5	5,5
	G 1.2	0,0016	52,5	37,5
	G 2.1	3,096222	16,5	8,5
	G 2.2	0,009044	60	12
MODEL F	G 1.1	6,110428	55,5	6,5
	G 1.2	0,000513	18	12
	G 2.1	7,383182	40,5	9,5
	G 2.2	0,007546	55,5	4,5

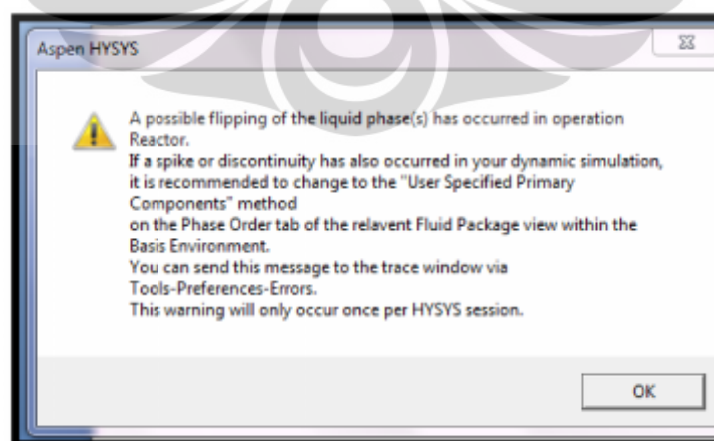
Kinerja pengendali diuji dengan melakukan perubahan *set point* (SP) konsentrasi produk dan temperatur pada nilai tertentu. Perubahan SP yang dilakukan diklasifikasikan menjadi 7 skenario. Skenario 1 merupakan konsentrasi produk yang ditentukan dalam rentang perubahan yang rendah yaitu dari 0,33 menjadi 0,331 dan temperatur dari 220 °F menjadi 221 °F. Sedangkan skenario 2 sampai 7 merupakan perubahan SP sesuai dengan perubahan konsentrasi produk dan temperatur pada PRC setiap model. Perubahan SP pada setiap skenario ditampilkan pada Tabel 4.3.

Tabel 4.3. Perubahan *set point* (SP) pada setiap skenario

Skenario	SP konsentrasi		SP temperatur	
	Xpg awal	Xpg akhir	T awal (⁰ F)	T akhir (⁰ F)
1	0,33	0,331	220	221
2	0,3294	0,3494	224,057	234,352
3	0,2982	0,3491	166,4343	210,0607
4	0,3607	0,37	237,199	237,478
5	0,3514	0,37	236,374	236,848
6	0,3294	0,37	224,123	235,102
7	0,2965	0,37	163,276	225,571

Simulasi yang konvergen diukur dari kestabilan sistem pada skenario pengujian yang dijalankan. Ketidakstabilan sistem dapat dinilai dengan nilai CV yang naik turun karena bukaan valve *coolant* atau umpan beresilasi dari 0% sampai 100% disertai kondisi penurunan persen level cairan. Hal ini akan menyebabkan tidak adanya kesetimbangan massa dan energi dalam sistem yang menyebabkan sistem *error* dan tidak dapat dijalankan kembali.

Selain itu, ketidakstabilan sistem juga dapat dinilai jika terjadi penurunan pada persen level cairan reaktor yang kemudian disertai dengan turunnya konsentrasi produk dan temperatur reaktor. Seperti tidak ada pengaruh dari pengendali, kondisi konsentrasi turun hingga nilai 0 (nol) dengan temperatur kurang dari -0 ⁰F. Pada kondisi tersebut, akan muncul peringatan yang ditampilkan sistem seperti pada Gambar 4.6 dan sistem harus di *hold-off* untuk kemudian dilakukan *tuning* ulang pada pengendali atau dilakuan simulasi ulang.



Gambar 4.6. Peringatan dari sistem jika terjadi ketidakstabilan sistem

4.2 Kinerja pengendali PI

Uji kinerja pengendali PI dilakukan untuk dijadikan pembandingan pada hasil kinerja pengendalian MPC. Pada pengendali PI, *tuning* parameter PI pada nilai K_c dan T_i dilakukan untuk mengoptimalkan kinerja pengendali. Kondisi optimum dinilai dengan nilai IAE dari respon CV yang dihasilkan karena adanya perubahan SP.

Tuning parameter PI dilakukan pada setiap pengendali dengan cara *trial error*. Nilai parameter PI hanya ditetapkan satu nilai untuk diuji pada berbagai skenario. Respon dari CV ditampilkan dalam Lampiran C dengan parameter *tuning* PI dan nilai IAE pada setiap skenario yang ditampilkan dalam Tabel 4.4.

Tabel 4.4. Nilai parameter *tuning* dan IAE pada pengendali PI setiap skenario

Skenario	IAE	
	AC [$K_c = 8,74$; $T_i = 32$]	TC [$K_c = 10$; $T_i = 6,37$]
1	0,00054	2,04
2	0,2002	51,475
3	0,3248	104,2662
4	Tidak terkontrol	Tidak terkontrol
5	Tidak terkontrol	Tidak terkontrol
6	0,34385	Tidak terkontrol
7	0,7991	252,343

4.3 Kinerja pengendali MPC

Berdasarkan teoritis, parameter *tuning* yang mempengaruhi kinerja MPC adalah *model horizon* (N), waktu sampel (T_s), *prediction horizon* (P), dan *control horizon* (M). Sedangkan pada perangkat lunak yang digunakan, seperti pada tampilan Gambar 3.35, parameter untuk *tuning* MPC adalah *Step Resp. Length* (SRL), *Prediction Horizon* (P), *Control Horizon* (M), *Control Interval* (T_s), *Ref. Trajectory*, Γ_U dan Γ_Y . Seperti telah dipaparkan pada BAB 3, Unisim mempunyai fasilitas untuk mengubah parameter *tuning* MPC, sebelum parameter tersebut diubah Unisim menetapkan nilai tertentu yang dapat dijadikan sebagai acuan. Nilai parameter *tuning* pada Gambar 3.35 tersebut dijadikan sebagai nilai parameter *tuning default*.

Perubahan pada masing-masing parameter *tuning* dilakukan untuk melihat pengaruhnya pada kinerja MPC. Hasil dari perubahan tersebut menunjukkan bahwa parameter *tuning* yang sangat berpengaruh terhadap kinerja MPC adalah *Step Resp. Length* (SRL), *Prediction Horizon* (P), *Control Horizon* (M), *Control Interval* (Ts). Sehingga selanjutnya *tuning* MPC hanya dilakukan pada parameter tersebut. Data dan penjelasan pengaruh perubahan nilai pada masing-masing parameter *tuning* ditampilkan pada Lampiran D.

4.3.1 Kinerja MPC *Singlevariable*

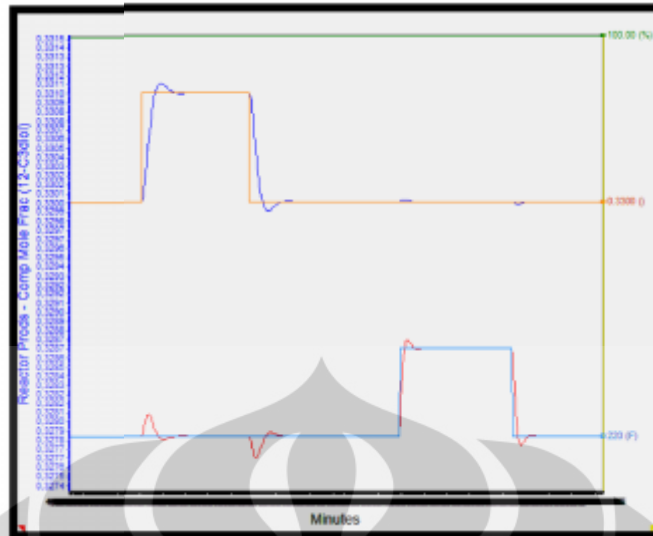
MPC *singlevariable* merupakan MPC dengan ukuran $[1 \times 1]$. *Tuning* parameter MPC *singlevariable* dilakukan pada parameter MPC yang terdapat pada MPC-AC dan MPC-TC. Uji kinerja MPC *singlevariable* hanya dilakukan pada model A sebagai model *default* dengan skenario 1 untuk melihat kemampuan pengendali MPC *singlevariable*. Model FOPDT yang digunakan pada pengendali MPC *singlevariable* ini ditunjukkan pada Tabel 4.5.

Tabel 4.5. Parameter FOPDT untuk MPC *singlevariable*

Parameter	Kp	τp	Θ	
MPC-TC	G 1.1	4,118	34,5	0,5
MPC-AC	G 1.1	0,008	57	6

FOPDT yang digunakan tidak bisa dari model yang berbeda. Misalnya MPC-TC dengan FOPDT G 1.1 dari model A dan MPC-AC dengan FOPDT G 2.1 dari model E. Pemilihan FOPDT dari model yang berbeda menyebabkan tidak adanya kesesuaian pada respon yang dihasilkan dan sistem cenderung langsung menunjukkan ketidakstabilan atau ketidaksesuaian dengan model yang digunakan. Sehingga pemilihan FOPDT untuk MPC *singlevariable* harus pada model yang sama sesuai tahapan identifikasi sistem yang telah dilakukan.

Respon dari *tuning* parameter MPC pada pengendali MPC *singlevariable* ditampilkan pada Gambar 4.7 dengan kondisi *tuning* parameter MPC yang ditampilkan pada Tabel 4.6.



Gambar 4.7. Kinerja pengendali MPC *singlevariable* dengan model A skenario 1

Tabel 4.6. Parameter *tuning* MPC *singlevariable* sesuai Gambar 4.7

Pengendali	Parameter <i>tuning</i>	Nilai
AC	SRL	50
	P	25
	M	1
	Ts	48
	IAE	0,1653
TC	SRL	50
	P	25
	M	1
	Ts	75
	IAE	42,717

4.3.2 Kinerja MPC *Multivariable*

Tuning parameter MPC *multivariable* hanya dilakukan pada MPC-100 yang akan mewakili perilaku sistem pengendali tersebut. Uji kinerja pengendali MPC *multivariable* dilakukan pada setiap model dan setiap skenario. Adanya variasi model dan skenario ini dilakukan untuk melihat pengaruh penggunaan model yang digunakan pada pengendali MPC karena kinerjanya ditentukan oleh penggunaan model yang tepat. Hasil dan pembahasan kinerja MPC *multivariable* adalah sebagai berikut:

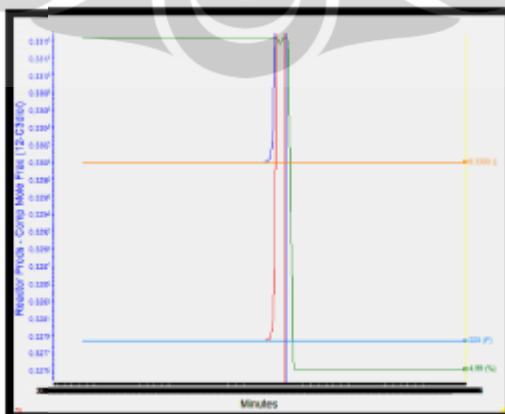
4.3.2.1 Kinerja MPC *Multivariable* dengan startegi *tuning non-adaptive*

Berdasarkan teoritis, parameter MPC *multivariable* dapat diketahui dengan startegi *tuning non-adaptive* (Dougherty, 2003a). Strategi tersebut diuji cobakan pada simulasi MPC dengan semua model. Contoh perhitungan parameter *tuning* pada model A dilampirkan pada Lampiran E dan hasil perhitungan untuk setiap model ditampilkan pada Tabel 4.7.

Tabel 4.7. Hasil Perhitungan parameter *tuning* berdasarkan strategi *tuning non-adaptive*

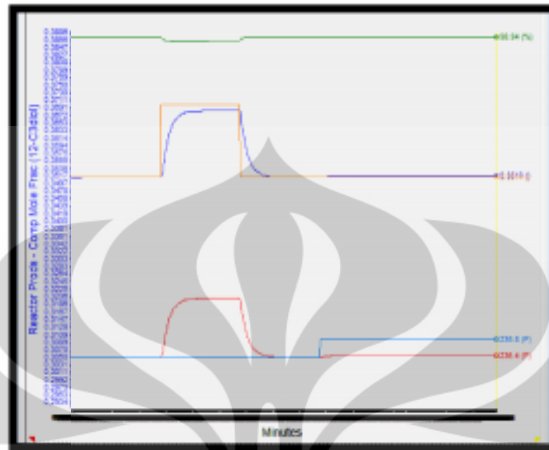
Model	Parameter <i>tuning</i> MPC		
	Ts	P	M
A	3,45	85,34783	29,4058
B	4,75	79,26087	18,3913
C	4,65	86,21739	19,84058
D	5,55	90,85507	21,28986
E	2,75	91,43478	27,08696
F	4,75	83,31884	18,97101

Nilai parameter *tuning* pada Tabel 4.7 dimasukkan pada pengendali MPC, namun pada Unisim, nilai parameter *tuning* P dan M tidak bisa dalam bentuk desimal sehingga harus dalam pembulatangannya. Contohnya pada model A, P dimasukkan dalam simulasi sebesar 85 dan M sebesar 29. Hasil respon dari pengendali MPC *multivariable* dengan menggunakan parameter tersebut untuk skenario 1 model A ditampilkan pada Gambar 4.8.



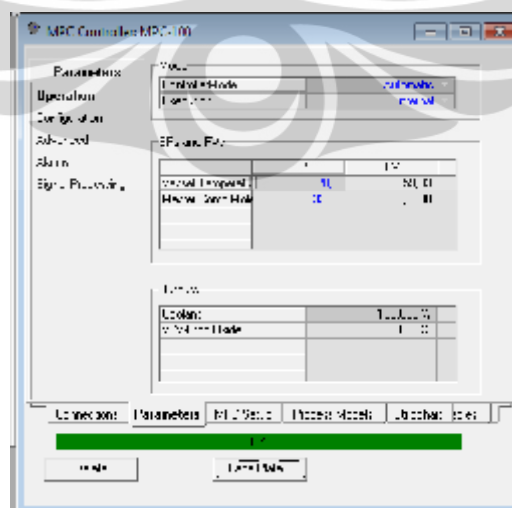
Gambar 4.8. Kinerja Pengendali MPC *multivariable* dengan startegi *tuning non-adaptive* (model A skenario 1)

Hasil simulasi dengan menggunakan model A skenario 1 ini juga terjadi pada setiap model dengan skenario 1 dan pada setiap model dengan setiap skenario, kecuali pada model D skenario 5.



Gambar 4.9. Kinerja Pengendali MPC *multivariabel* dengan strategi *tuning non-adaptive* (Model D skenario 5)

Berdasarkan Gambar 4.8, sistem menunjukkan ketidakstabilan bahkan sebelum dilakukan perubahan nilai SP konsentrasi. Nilai akhir bukaan *coolant* menunjukkan 100% dan bukaan valve umpan propylene oxide menunjukkan 0%, ditunjukkan pada Gambar 4.10 sebagai tampilan simulasi pengendali dalam Unisim. Artinya aliran pendingin masuk dalam jumlah berlebih dan tidak diimbangi oleh masukan umpan sehingga kondisi CSTR dibawah temperatur yang diinginkan.



Gambar 4.10. Tampilan kondisi pengendali hasil simulasi dengan strategi *tuning non-adaptive*

Kondisi yang berbeda ditunjukkan dari pengendali dengan model D skenario 5 yang ditunjukkan pada Gambar 4.9. Sistem masih stabil dengan diterapkannya parameter *tuning non-adaptive*, namun CV yang terukur tidak dapat mengikuti SP yang ditetapkan. Pada waktu pengujian yang lebih lama pun, respon yang dihasilkan akan tetap sama.

4.3.2.2 Kinerja MPC *Multivariable* dengan strategi *tuning trial error*

Uji kinerja pengendali MPC *multivariable* dilakukan secara *trial error* pada setiap model dengan skenario 1 dan pada setiap model dengan skenario yang berbeda. Berikut merupakan hasil dan pembahasannya:

✓ Pengujian pada setiap model dengan skenario 1

Pengujian ini dilakukan dengan parameter *tuning default* pada nilai $T_s = 30$ detik, $SRL = 50$, $P = 25$, dan $M = 2$. Respon dari hasil pengendalian ditampilkan pada Lampiran F.

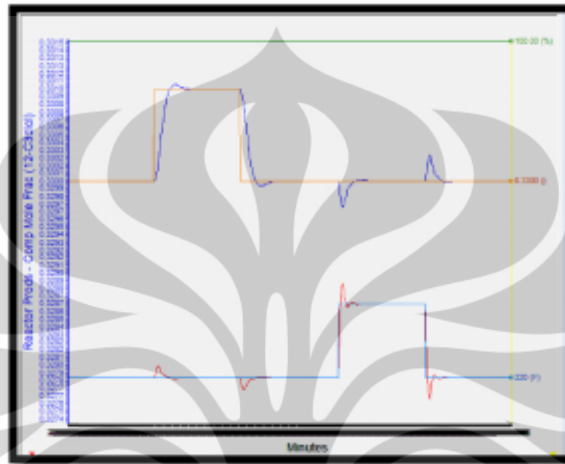
Penggunaan model A, B dan F menunjukkan respon pengendalian yang cukup baik dibandingkan dengan model lain. Pada perubahan SP konsentrasi, model C menunjukkan adanya respon *overshoot* dari PV temperatur yang cukup signifikan, model D menunjukkan waktu pencapaian SP yang lama dan model E menunjukkan tidak adanya respon dari PV konsentrasi.

Analisis dari respon yang dihasilkan ini dapat disebabkan karena perubahan *set point* yang dilakukan tidak berada pada rentang perubahan yang terjadi pada PRC setiap model. Sehingga kemungkinan model yang digunakan tidak sesuai dengan perubahan SP yang dilakukan.

Berdasarkan adanya respon yang dihasilkan pada penggunaan model A dan F yang cukup baik dengan parameter *tuning default*, maka *tuning trial error* skenario 1 hanya dilakukan pada model A dan model F. *Tuning trial error* tidak dilakukan pada model B, karena rentang perubahan temperatur pada model B adalah dari $166,4343\text{ }^{\circ}\text{F} - 210,0607\text{ }^{\circ}\text{F}$ sedangkan pada skenario 1 ini dilakukan perubahan dari $220\text{ }^{\circ}\text{F} - 221\text{ }^{\circ}\text{F}$. Hal ini dilakukan untuk melihat kinerja MPC jika perubahan SP berada pada nilai yang berbeda dari perubahan CV pada PRC tapi nilai perubahan tersebut masih pada rentang perubahan pada PRC tersebut. Hasil

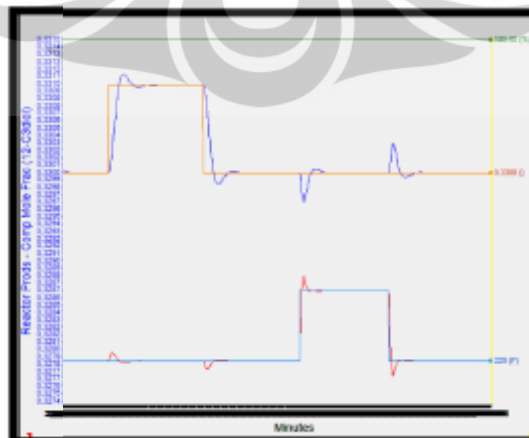
tuning trial error pada skenario 1 dengan model A dan F ditampilkan pada Lampiran G.

Respon yang dihasilkan dari *trial error* parameter *tuning* MPC *multivariable* menggunakan model A skenario 1 dengan IAE sebesar 0,12674 pada perubahan konsentrasi dan 73,558 untuk perubahan temperatur di ditampilkan pada Gambar 4.11.



Gambar 4.11. Kinerja Pengendali MPC *multivariabel* dengan model A skenario 1 [$T_s = 30$; $SRL = 35$; $P = 25$; $M = 1$]

Respon yang dihasilkan dengan model F juga tidak jauh berbeda dengan penggunaan model A. Berdasarkan respon yang ditampilkan pada Lampiran E, penggunaan model F menghasilkan respon terbaik yang ditampilkan pada Gambar 4.12 dengan IAE sebesar 0,10602 untuk perubahan konsentrasi dan 27,525 untuk perubahan temperatur.



Gambar 4.12. Kinerja Pengendali MPC *multivariabel* dengan model F skenario 1 [$T_s = 25$; $SRL = 50$; $P = 25$; $M = 1$]

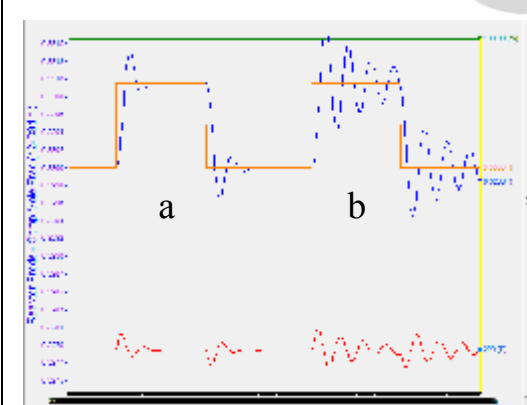
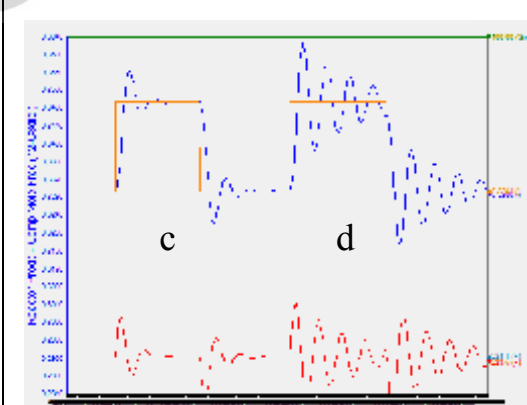
✓ **Pengujian setiap model dengan skenario yang berbeda**

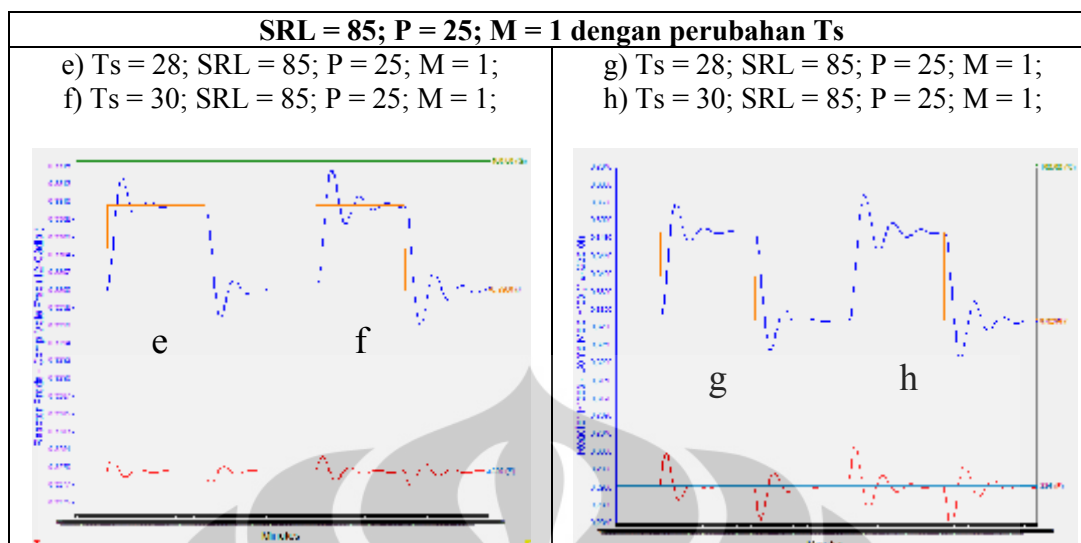
Uji kinerja MPC kemudian dilakukan dengan perubahan SP yang disesuaikan dengan perubahan CV pada PRC setiap model. Hasil respon CV pada parameter *tuning* yang optimum setiap skenario dan model ditampilkan pada Lampiran I.

Respon yang dihasilkan dari *tuning* parameter pada model A dengan skenario 2 cenderung sama dengan skenario 1 dalam perubahan SP konsentrasi. Hal ini ditunjukkan pada Tabel 4.8. Pada tabel tersebut dapat dilihat dengan perubahan konsentrasi yang berbeda pada skenario 1 dan 2, perilaku respon yang dihasilkan cenderung sama pada parameter *tuning* yang sama. Maksudnya jika ditetapkan kondisi $SRL = 50$; $P = 25$; $M = 1$, maka respon kurva a akan diperoleh pada nilai *control interval* sebesar $T_s = 32$ detik. Sedangkan jika ditetapkan kondisi $SRL = 85$; $P = 25$; $M = 1$, kurva serupa akan dihasilkan pada *control interval* sebesar $T_s = 28$, dibuktikan pada kurva e.

Pada proses *trial error*-nya, semakin besar nilai T_s maka respon menjadi tidak stabil dengan bukaan valve yang berosilasi dari 0% hingga 100% jika tidak diimbangi dengan kenaikan parameter *tuning* P, dan semakin kecil nilai T_s maka respon menjadi sangat lambat mencapai *set point*-nya. Hasil respon secara keseluruhan untuk model A skenario 2 ditampilkan pada lampiran H.

Tabel 4.8. Perbandingan respon dari skenario 1 dan 2 pada model 2

Model A, skenario 1	Model A, skenario 2
SRL = 50; P = 25; M = 1 dengan perubahan T_s	
a) $T_s = 32$; SRL = 50; P = 25; M = 1; b) $T_s = 34$; SRL = 50; P = 25; M = 1;	c) $T_s = 32$; SRL = 50; P = 25; M = 1; d) $T_s = 34$; SRL = 50; P = 25; M = 1;
	



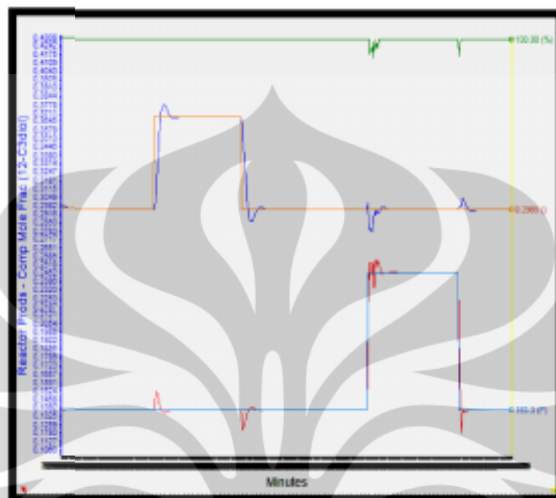
Respon yang dihasilkan dari *tuning* parameter pada model A dengan perubahan SP temperatur tidak menghasilkan perilaku yang sama antara skenario 1 dan 2. Pada skenario 2, dengan adanya perubahan SP temperatur, sistem cenderung tidak stabil.

Berdasarkan respon tersebut, penggunaan model A tidak dapat diaplikasikan pada MPC dengan pengujian skenario 2. Perlu diperhatikan bahwa walaupun perubahan konsentrasi menunjukkan respon yang cenderung sama, belum tentu hal tersebut juga terjadi pada perubahan temperatur. Kondisi operasi juga dapat mempengaruhi kinerja pengendali.

Tuning parameter MPC pada model B juga menghasilkan perilaku yang cenderung sama dengan model A, namun pada model B cenderung lebih mudah mengalami ketidakstabilan sehingga mempunyai batasan parameter MPC yang lebih sempit.

Pengujian dengan skenario 4 menggunakan model C dan skenario 5 menggunakan model D menghasilkan respon CV temperatur tidak mencapai SP yang ditentukan. Penggunaan model C dan D memerlukan *tuning* parameter MPC lebih lanjut. Sedangkan pada skenario 6 dengan model E, pada saat dilakukan perubahan SP temperatur terjadi penurunan persen level cairan reaktor yang dapat menyebabkan ketidakstabilan sistem jika kondisi temperatur tidak dikembalikan ke nilai awal.

Respon yang dihasilkan dari penggunaan model F mempunyai perilaku yang sama seperti model A dan B. Hasil respon pengendalian menggunakan model F skenario 7 ditampilkan pada Gambar 4.13 dengan IAE sebesar 4,02258 untuk perubahan konsentrasi dan 1406,205 untuk perubahan temperatur.

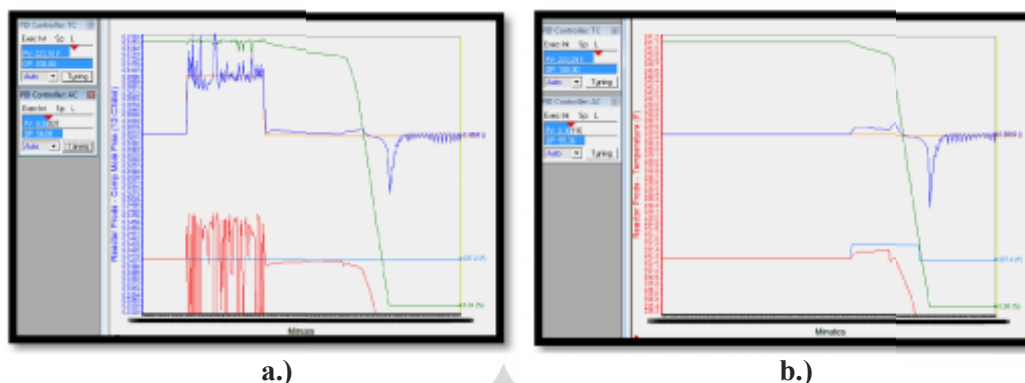


Gambar 4.13. Kinerja Pengendali MPC *multivariabel* pada skenario 7 model F [$T_s = 18$; $SRL = 50$; $P = 35$; $M = 1$]

4.4 Evaluasi

Simulasi yang telah dilakukan adalah mengendalikan sistem CSTR dengan reaksi pembuatan propylen glycol menggunakan 3 rangkaian pengendali yaitu PI, MPC *singlevariable* dan MPC *multivariable*. Hasil respon CV pada setiap skenario pengujian telah ditunjukkan pada bagian 4.2 dan 4.3 dengan data yang lebih lengkap pada Lampiran C, F, G, H dan I.

Hasil uji kinerja pengendali PI pada skenario 1, 3 dan 7 menunjukkan bahwa pengendali PI dapat mengendalikan sistem CSTR dengan nilai IAE yang sangat minim. Namun pada skenario 2, 4, 5 dan 6, sistem menunjukkan respon pengendalian yang tidak dapat dilakukan oleh pengendali PI. Seperti contohnya pada skenario 4 yang ditampilkan pada Gambar 4.14.



Gambar 4.14. Kinerja pengendali PI pada skenario 4
a.) perubahan SP konsentrasi, b.) perubahan SP temperatur

Skenario 4 menunjukkan ketidakstabilan bahkan pada saat simulasi dijalankan. Walaupun pada skenario 2 tidak terjadi kondisi tersebut, skenario 2 dianggap tidak dapat melakukan pengendalian pada saat dilakukan perubahan SP temperatur. Respon yang dihasilkan menunjukkan respon yang bagus dengan IAE sebesar 51,475, namun terjadi penurunan persen level cairan reaktor. Jika temperatur tidak dikembalikan ke harga SP semula maka akan terjadi ketidakstabilan sistem.

Seperti telah disebutkan bahwa perubahan SP pada skenario 2-7 adalah perubahan SP baik konsentrasi produk maupun temperatur sesuai dengan perubahan CV pada PRC. Respon yang ditunjukkan dari skenario 2, 4, 5 dan 6 dapat terjadi karena perubahan temperatur yang terlalu tinggi pada nilai lebih dari 235 °F, walaupun pada saat dilakukan *model testing* kondisi tersebut tidak menyebabkan sistem menjadi tidak stabil.

Pengujian pengendali MPC *singlevariable* hanya dilakukan pada satu model dan skenario untuk melihat kemampuannya dalam mengatasi perubahan SP. Pada penentuan parameter MPC *singlevariable* model A skenario 1 menunjukkan respon dengan hasil IAE yang lebih besar dari pengendalian PI untuk skenario 1. Penggunaan pengendali MPC *singlevariable* ini belum tepat diterapkan karena sistem yang digunakan mempunyai variabel yang saling berinteraksi. Sehingga dalam pengendaliannya diperlukan pengendali yang melibatkan model hasil interaksi antara variabel yang dikendalikan.

Secara teoritis, MPC *multivariable* telah membuktikan kemampuannya dalam mengatasi interaksi antara variabel jamak yang dikendalikan. Pengujian

pengendali MPC *multivariable* dalam penelitian ini dilakukan dengan menggunakan strategi *tuning non-adaptive* dan strategi *trial error*.

Hasil pengujian MPC *multivariable* menggunakan strategi *tuning non-adaptive* menunjukkan tidak adanya pengaruh pengendalian yang diberikan. Hal tersebut dapat disebabkan karena terlalu besarnya jangkauan P sebagai hasil perhitungan dari persamaan strategi *tuning non-adaptive* karena mungkin proses tidak membutuhkan prediksi sebesar itu. Selain itu, waktu sampel hasil perhitungan juga terlalu besar, padahal jika dilakukan *trial error* pada nilai parameter MPC, waktu sampel yang dibutuhkan cukup kecil karena jika ditetapkan waktu yang lama proses menjadi cepat tidak stabil atau terjadi respon yang agresif. Berdasarkan respon tersebut, strategi *tuning non-adaptive* tidak sesuai penggunaannya jika diterapkan pada perangkat lunak Unisim R390.1 ini.

Pengujian pengendali MPC *multivariable* menggunakan strategi *trial error* menghasilkan respon yang lebih baik, baik dalam skenario 1 dengan semua model maupun dalam setiap skenario dengan masing-masing model.

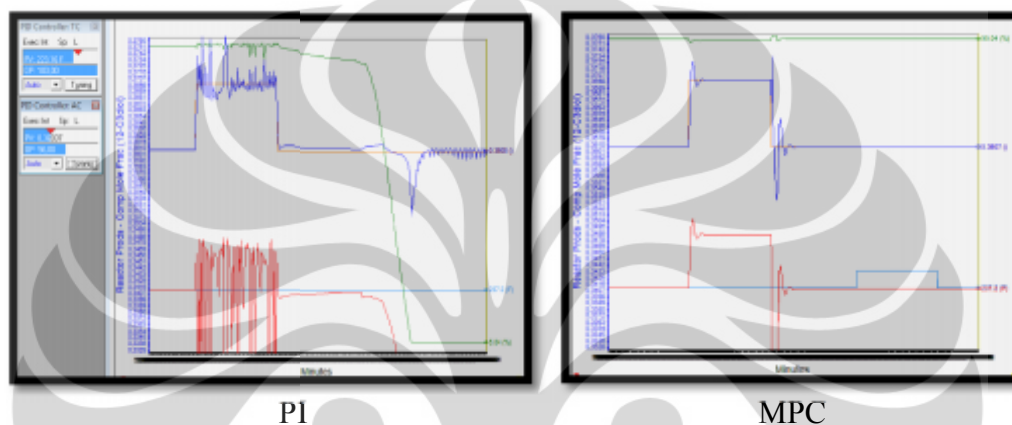
Dalam pengujian dengan skenario 1, dari keseluruhan model yang dibuat, tidak semua model dapat diaplikasikan pada pengendali MPC. Model A dan F memberikan respon paling baik, hal ini didasarkan pada pengujian dengan parameter *tuning default*. Sedangkan hasil dari *tuning trail error* menunjukkan bahwa model F lebih baik untuk diaplikasikan pada skenario 1 dibandingkan model A. Hal tersebut dilihat dari Tabel 4.9. Nilai IAE model F pada perubahan SP Xpg dan T menunjukkan nilai yang lebih kecil dibandingkan dengan model A.

Tabel 4.9. Perbandingan nilai IAE pada setiap rangkaian pengendali untuk skenario 1

Pengendali	IAE				Deviasi terhadap PI (%)			
	model A		model F		model A		model F	
	Xpg	T	Xpg	T	Xpg	T	Xpg	T
PI	0,00054	2,04	0,00054	2,04	-	-	-	-
MPC <i>singlevariable</i>	0,1653	42,717	tidak dilakukan		30.511,11	1.993,97	tidak dilakukan	
MPC <i>multivariable</i>	0,12674	73,558	0,10602	27,525	23.370,37	3.505,78	19.533,33	1.249,26

Pengendali MPC *multivariable* yang diuji pada skenario 4, 5 dan 6 menunjukkan kemampuannya dalam mempertahankan kestabilan sistem

dibandingkan dengan pengendali PI. Contohnya pada skenario 4 dengan model D yang ditampilkan pada Gambar 4.15. Walaupun pada MPC respon CV temperatur tidak mencapai SP yang ditentukan, kurva tersebut menunjukkan MPC lebih dapat menjaga kestabilan sistem dibandingkan dengan pengendali PI. Pada perubahan SP konsentrasi terjadi respon CV yang mengikuti SP dari pengendali MPC, sedangkan dari pengendali PI menghasilkan respon CV yang berosilasi. Respon dari MPC skenario 4 ini dicapai dengan parameter tuning pada $T_s = 30$; $SRL = 50$; $P = 25$; $M = 2$.



Gambar 4.15. Kinerja Pengendali PI dan MPC *multivariabel* pada skenario 4 model C

Rangkuman nilai IAE untuk setiap skenario sesuai dengan model-nya ditampilkan pada Tabel 4.10.

Tabel 4.10. Perbandingan nilai IAE pada rangkaian pengendali PI dan MPC untuk setiap skenario

Skenario	IAE			
	PI, AC [$K_c = 8,74$; $T_i = 32$]	MPC, AC	PI, TC [$K_c = 10$; $T_i = 6,37$]	MPC, TC
2	0,2002	2,3498	51,475	Tidak terkontrol
3	0,3248	6,9577	104,2662	2746,946
4	(Lebih stabil dibanding PI)	Tidak terkontrol	(Lebih stabil dibanding PI)	Tidak terkontrol
5	(Lebih stabil dibanding PI)	Tidak terkontrol	(Lebih stabil dibanding PI)	Tidak terkontrol
6	0,34385	5,727	(Lebih stabil dibanding PI)	Tidak terkontrol
7	0,7991	4,02258	252,343	1406,205

Secara keseluruhan, kinerja pengendali MPC baik *singlevariable* ataupun *multivariable* yang disimulasikan dalam Unisim R390.1 tidak menghasilkan pengendalian yang lebih baik dibandingkan pengendali PI ditinjau dari nilai IAE karena nilai IAE yang dihasilkan dari respon pengendali MPC masih besar nilainya. Namun, ditinjau dari kemampuan dalam mempertahankan kestabilan sistem, pengendali MPC mempunyai keunggulan yang lebih baik dibanding pengendali PI.

Besarnya nilai IAE pada pengendali MPC yang disimulasikan pada perangkat lunak Unisim mengacu pada penggunaan model yang tidak sesuai. Ketepatan pembacaan dalam menentukan nilai K_p , τ_p , dan Θ_p dari PRC hasil *model testing* sangat ditentukan oleh ketelitian pembacaan karena dilakukan secara manual. Sehingga kesalahan pembacaan nilai parameter FOPDT dapat menyebabkan ketidaksesuaian model yang digunakan.

Pada pengujian setiap model dengan skenario yang berbeda, berdasarkan Tabel 4.10, model yang dapat diaplikasikan sesuai dengan skenario-nya adalah model B dengan skenario 3 dan model F dengan skenario 7. Karena pada model B tidak dilakukan pengujian skenario 1 lebih lanjut, maka yang dapat dibandingkan antara model dengan skenario 1 dan model dengan skenario perubahan SP sesuai PRC adalah model F. Tabel 4.11 menunjukkan perbandingan nilai IAE dan deviasi terhadap PI dari penggunaan model F pada skenario 1 dan skenario 7.

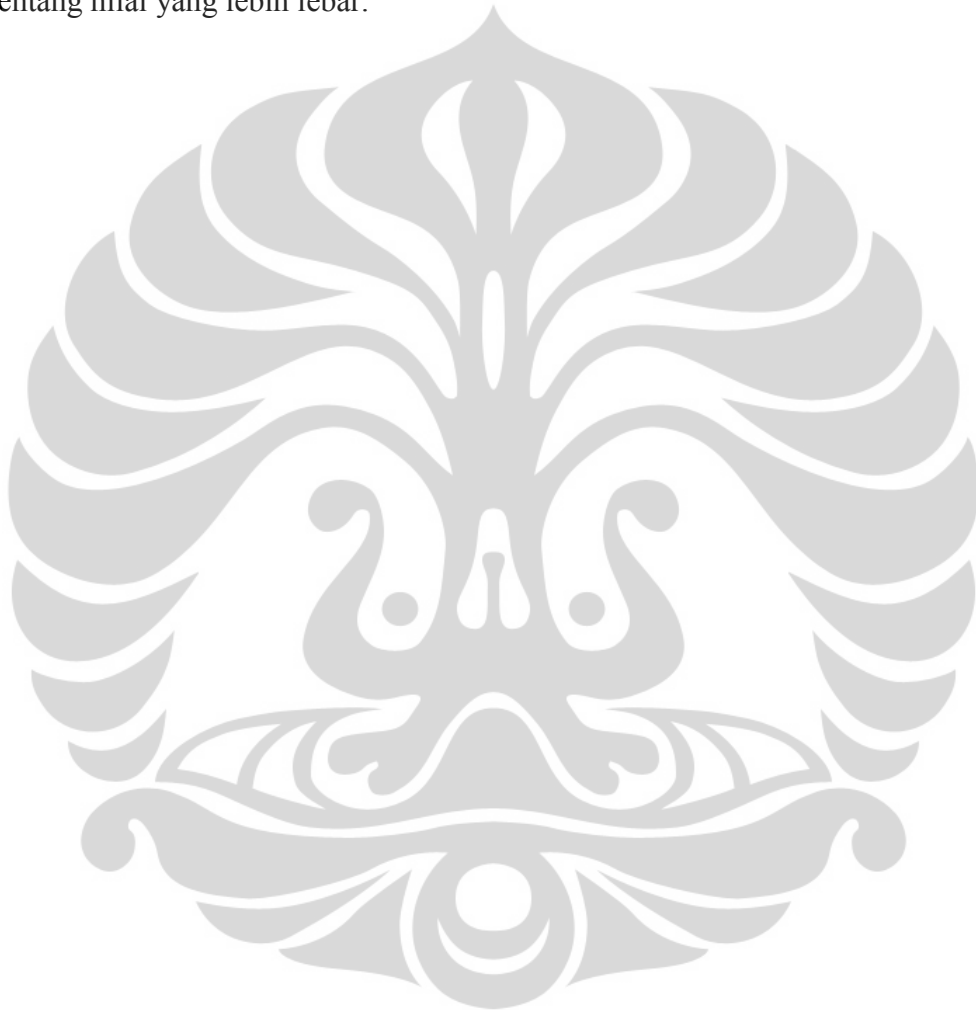
Tabel 4.11. Perbandingan nilai IAE pada setiap rangkaian pengendali dengan Model F

Pengendali	IAE, Model F			
	Skenario 1		Skenario 7	
	Xpg	T	Xpg	T
PI	0,00054	2,04	0,7991	252,343
MPC <i>multivariable</i>	0,10602	27,525	4,02258	1406,205
Deviasi terhadap PI (%)	19.533,33	1.249,26	403,39	457,26

Model F mempunyai rentang perubahan bukaan valve yang cukup besar yaitu dari 45,8% ke 56%. Dengan rentang yang cukup besar tersebut, penggunaan model F lebih sesuai jika diterapkan pada pengujian dengan perubahan CV yang lebih sempit rentangnya atau pada skenario 1 dengan perubahan konsentrasi 0,33 ke 0,331. Hal ini berdasarkan nilai IAE pada skenario 1 menghasilkan IAE lebih

kecil dibandingkan skenario 7 yang mempunyai rentang perubahan CV yang lebih lebar atau dari 0,2965 ke 0,37.

Hal ini dapat menunjukkan bahwa model yang digunakan akan lebih mengoptimalkan pengendali MPC jika digunakan pada perubahan SP dengan rentang perubahan yang lebih sempit. Artinya, diperlukan lebih banyak model atau pengendali MPC dengan multi model jika perubahan SP dilakukan pada rentang nilai yang lebih lebar.



BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Model dinamik variabel jamak (*multivariable*) terbaik untuk diaplikasikan pada sistem CSTR dengan reaksi pembuatan propylene glycol adalah

$$\text{model F, } \begin{pmatrix} T \\ X_{pg} \end{pmatrix} = \begin{pmatrix} \frac{6,110428e^{-6,5s}}{55,5s+1} & \frac{0,000513e^{-12s}}{18s+1} \\ \frac{7,383182e^{-9,5s}}{40,5s+1} & \frac{0,007546e^{-4,5s}}{55,5s+1} \end{pmatrix} \begin{pmatrix} Fc \\ Fco \end{pmatrix}$$

2. Strategi *tuning non-adaptive* tidak dapat diaplikasikan untuk menghitung parameter tuning MPC pada perangkat lunak Unisim R390.1,
3. Skenario 2, 4, 5 dan 6 tidak dapat dikendalikan oleh pengendali PI,
4. Pada skenario 1, hasil terbaik diberikan oleh MPC *multivariable* menggunakan model F pada parameter tuning $T_s = 25$; $SRL = 50$; $P = 25$; $M = 1$, dengan IAE sebesar 0,10602 dan 27,525,
5. Secara keseluruhan, pengendali MPC belum dapat mengendalikan sistem CSTR dengan reaksi pembuatan propylene glycol secara optimum berdasarkan nilai IAE, namun pengendali MPC lebih mampu menjaga kestabilan sistem dibandingkan pengendali PI.

5.2 Saran

Berdasarkan kesimpulan dari hasil penelitian dan pembahasan yang telah penulis lakukan, maka penulis menyarankan untuk adanya penelitian lebih lanjut mengenai :

1. penambahan variasi pengujian perubahan *set point* dengan model yang lebih bervariasi
2. penggunaan MMPC (*Multi Model Predictive Control*) *multivariable* dalam perangkat lunak Unisim R390.1 untuk pengendalian dalam sistem CSTR

3. penambahan pengujian gangguan (*disturbance*) pada sistem dengan pengendali MPC dalam perangkat lunak Unisim R390.1.



DAFTAR PUSTAKA

- Aşar, Işık. (2004). *Model Predictive Control (MPC) Performance for Controlling Reaction Systems*. Haziran: Middle East technical University.
- Aris, R. (1969). *Elementary chemical reactor analysis*.
- Bao, J., & Peter, L.L. (2007). *Process control, the passive sistem approach*. London: Springer-Verlag.
- Bequette, B. W. (1998). *Process dynamics: modeling, analysis, and simulation*. New Jersey: Prentice Hall, Inc.
- Bequette, B. W. (2002). Behavior of a CSTR with a recirculating jacket heat transfer system. *Proceedings of the American Control Conference*. New York: Isermann Department of Chemical Engineering.
- Camacho, E. F., & Carlos, B. (2007). *Nonlinear model predictive control: an introductory review*. Spain: University of Seville.
- Christofides, P. D., J. Liu, & D. Muñoz de la Peña. (2011). *Networked and distributed predictive control, advance industrial control*. London: Springer-Verlag.
- Dostal, P., V. Bobal, & F. Gazdos. (2011). Simulation of nonlinear adaptive control of a continuous stirred tank reactor. *International Journal of Mathematics and Computers in Simulation*, 5, Issue 4, 370-377.
- Dougherty, D., dan D. Cooper. (2003a). A practical multiple model adaptive strategy for multivariable model predictive control. *Control Engineering Practice*, 11, 649-664.
- Dougherty, D., dan D. Cooper. (2003b). A practical multiple model adaptive strategy for single-loop MPC. *Control Engineering Practice*, 11, 141-159.
- Fogler, H.S. (2006). *Element of Chemical Reaction Engineering* (4th ed). Canada: Pearson Education, Inc.
- Huang, H.P., Jyh, C.J., Chih, H.C., & Wen, P. (2003). A direct method for multi-loop PI/PID controller design. *Journal of Process Control*, 13, 769-786. Taiwan: Department of Chemical Engineering, National Taiwan University.

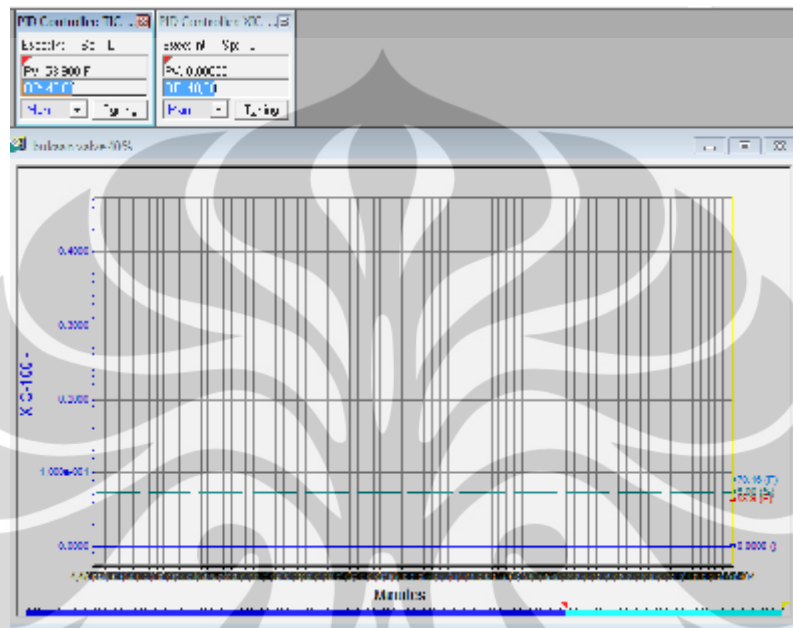
- Khaniki, R., M.B., Menhaj, & H., Eliasi. (2007). Generalized predictive control of batch polymerization reactor. *World Academy of Science, Engineering and Technology* 36.
- Luyben, M.L., & W.L., Luyben. (1997). *Essentials of process control*. Singapore: McGraw-Hill.
- Marlin, T. (2000). *Process control: designing processes and control systems for dynamic performance* (2nd ed). New York: McGraw-Hill.
- Ogunnaike, B.A., & W.H., Ray. (1994). *Process dynamics, modelling and control*. New York : Oxford University Press.
- Qin, S.J., & Thomas, A.B. (2002). A survey of industrial model predictive control technology. *Control Engineering Practice*, 11, 733-764.
- Seborg, D.E., Thomas, F.E., & Dacan, A.M. (2003). *Process dynamics and control* (2nd ed). California: John Wiley & Sons, Inc.
- Unisim design, tutorial and applications*. (2008). Canada: Honeywell.
- Uppal, A., W.H., Ray & A.B., Poore. (1974). On the dynamic behavior of continuous stirred tanks. *Chemical Engineering Science*, 29, 957-985.
- Upadhyay, R., & Rajesh, S. (2010). Analysis of CSTR temperature control with adaptive and PID controller (a comparative study). *IACSIT International Journal of Engineering and Technology*, Vol. 2, 5, 1793-8236.
- Wade, Harold L. (2004). *Basic and advanced regulatory control: system design and application* (2nd ed). Texas: ISA.
- Yu, D.W., & D.L. Yu. (2005). Modeling a multivariable reactor and on-line model predictive control. *ISA Transactions*, 44, 539-559. Qinhuangdao: Automation Department.

LAMPIRAN A

PRC setiap model dan kondisi awal sistem pada bukaan valve tertentu

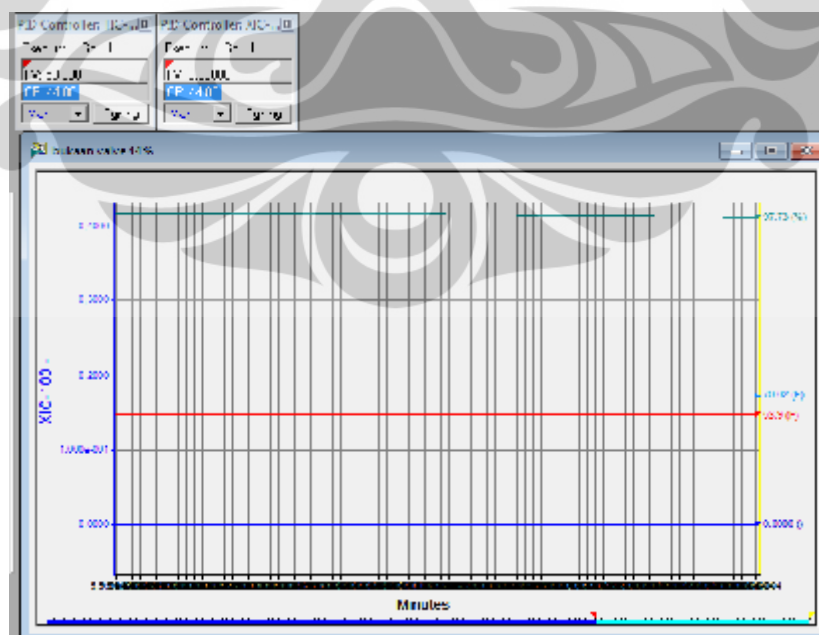
A.1. Kondisi Awal Sistem Pada Bukaan Valve Tertentu

- Bukaan valve 40%



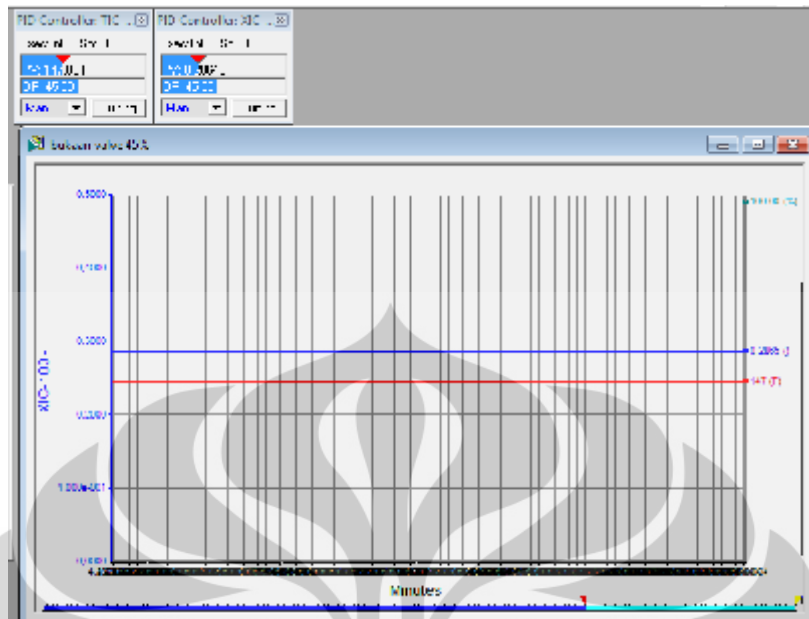
Gambar A.1. Kondisi sistem pada bukaan valve 40%

- Bukaan valve 44%



Gambar A.2. Kondisi sistem pada bukaan valve 44%

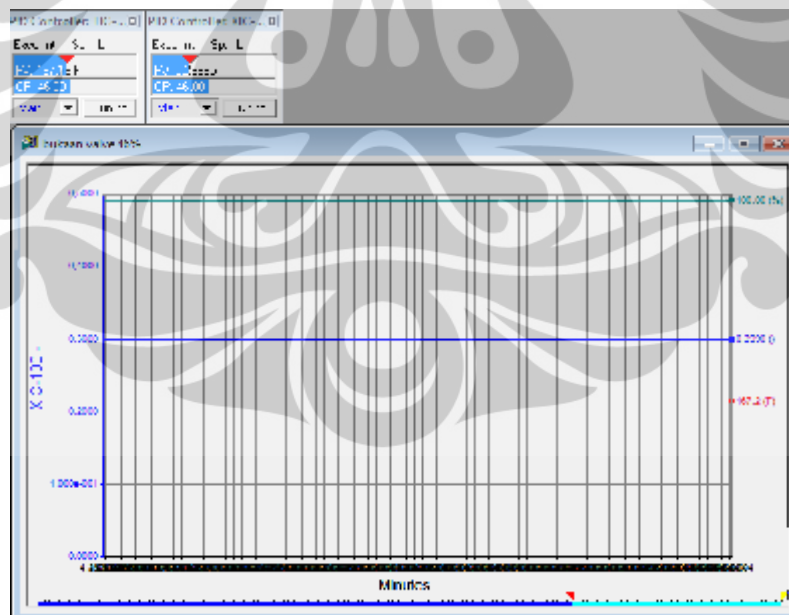
- Buka valve 45%



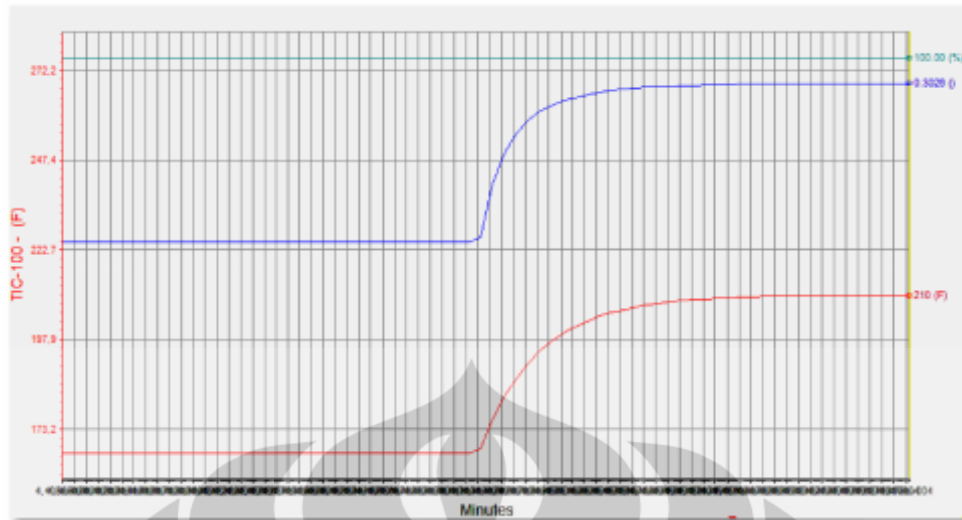
Gambar A.3. Kondisi sistem pada bukaan valve 45%

A.2. PRC dan Kondisi Sistem Pada Kondisi Awal Untuk Setiap Model

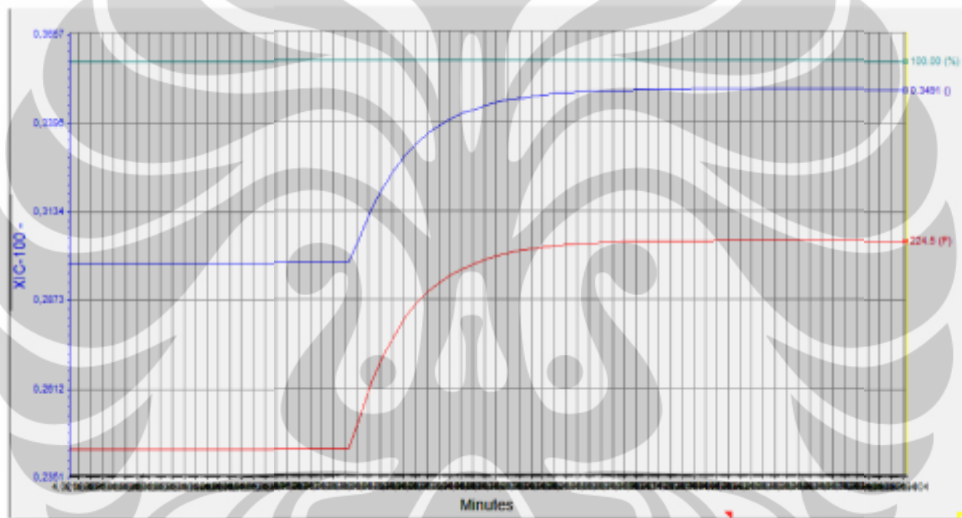
- Model B



Gambar A.4. Kondisi sistem pada bukaan valve 46%

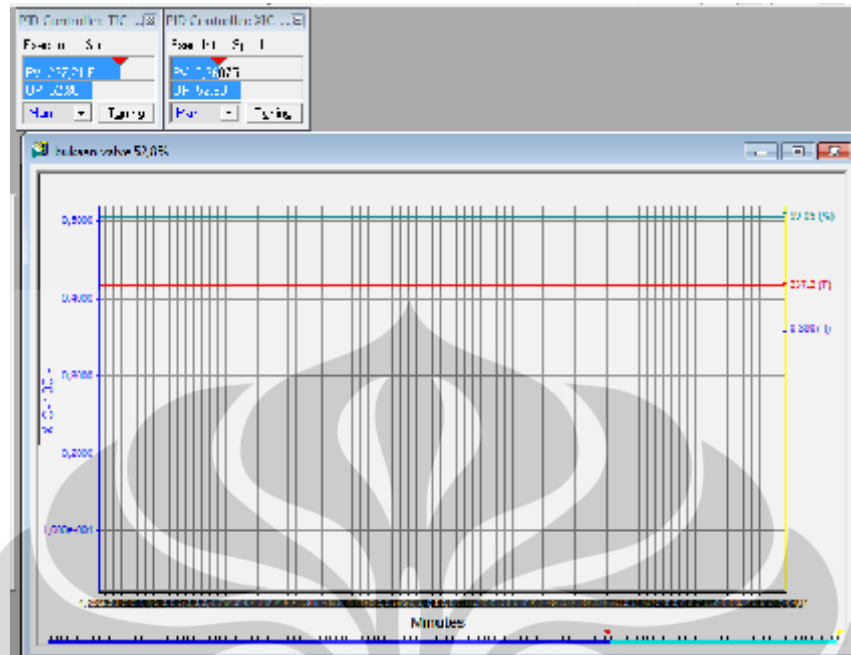


Gambar A.5. PRC Pengaruh Perubahan F_c Untuk Model B

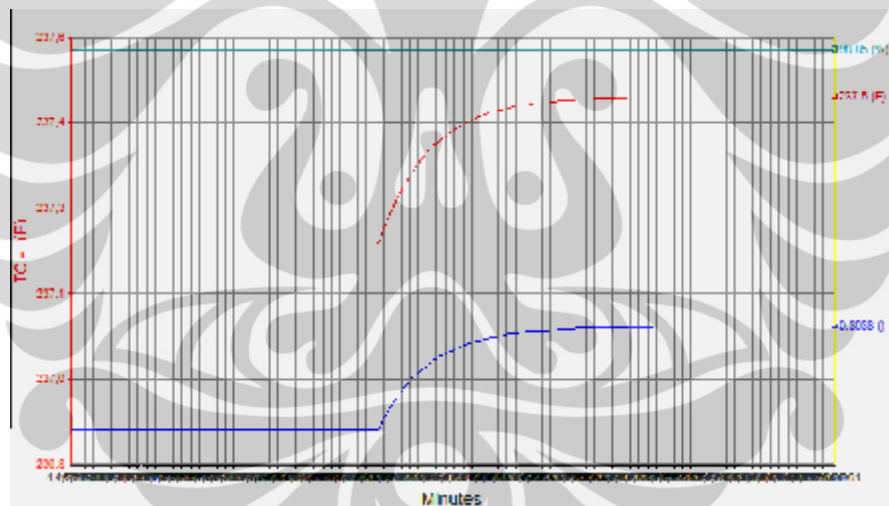


Gambar A.6. PRC Pengaruh Perubahan F_{po} Untuk Model B

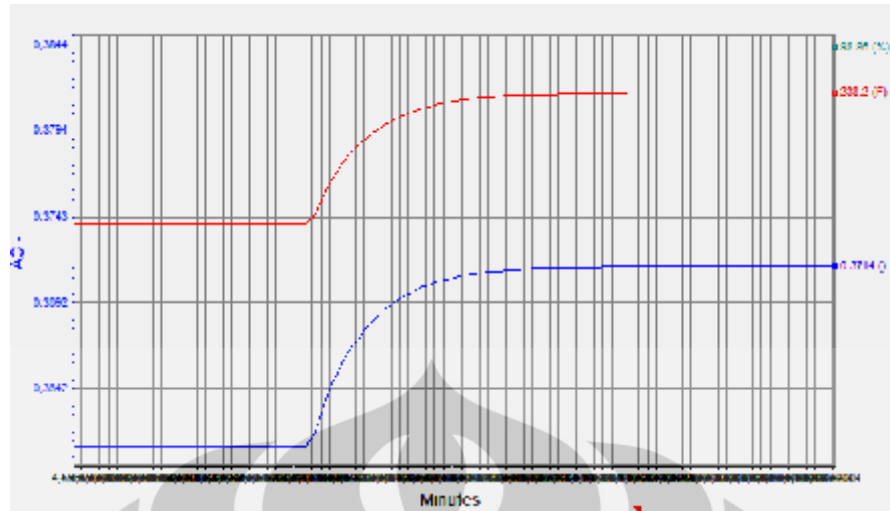
- Model C



Gambar A.7. Kondisi sistem pada bukaan valve 52,8%

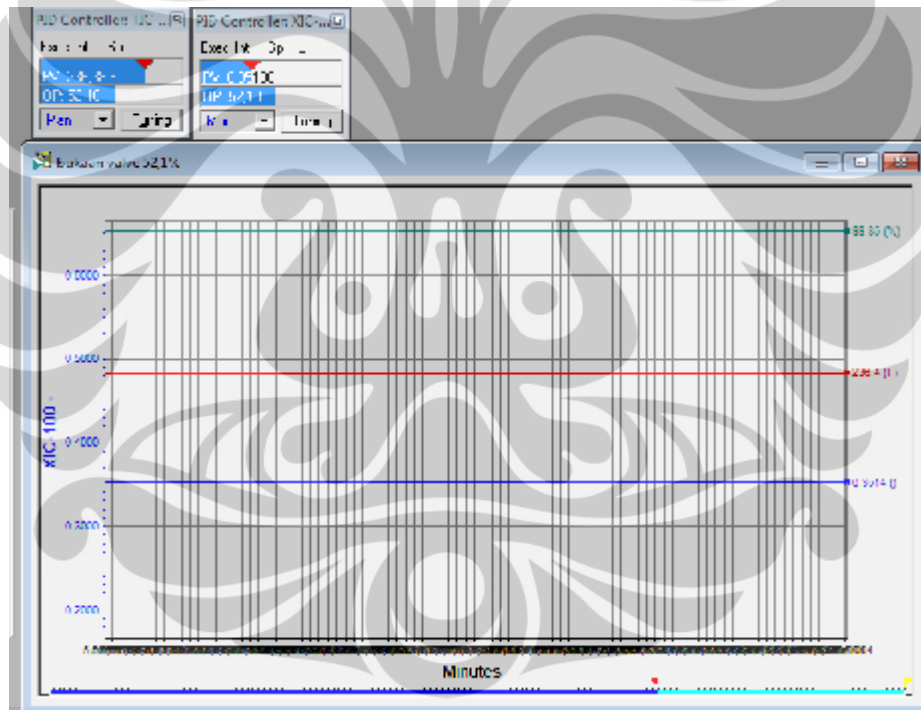


Gambar A.8. PRC Pengaruh Perubahan F_c Untuk Model C

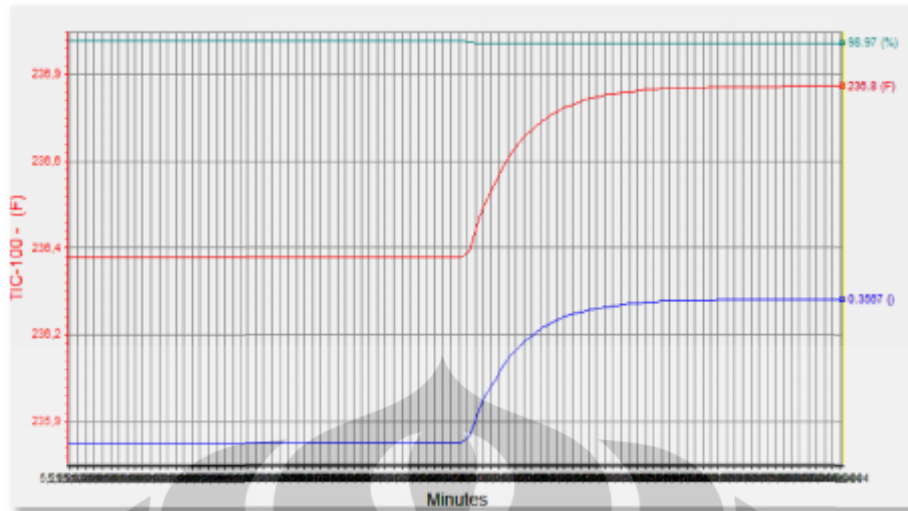


Gambar A.9. PRC Pengaruh Perubahan Fpo Untuk Model C

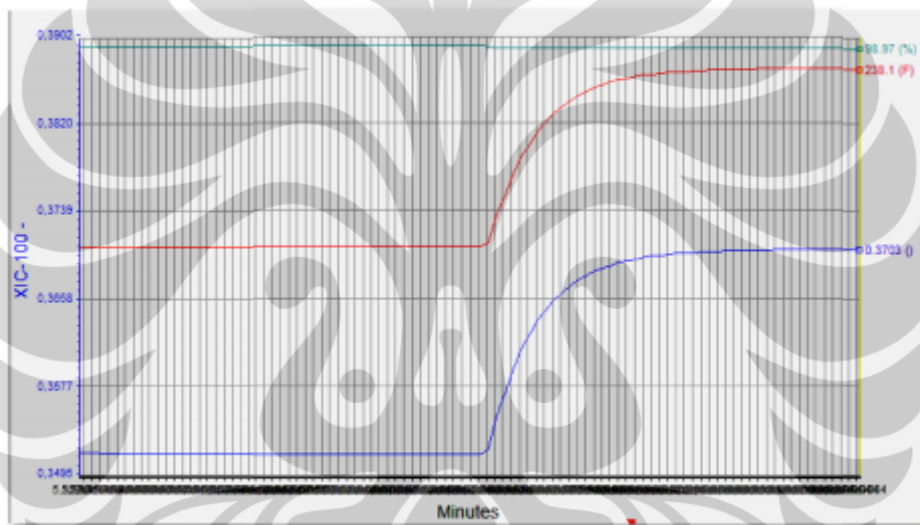
- **Model D**



Gambar A.10. Kondisi sistem pada bukaan valve 52,1%

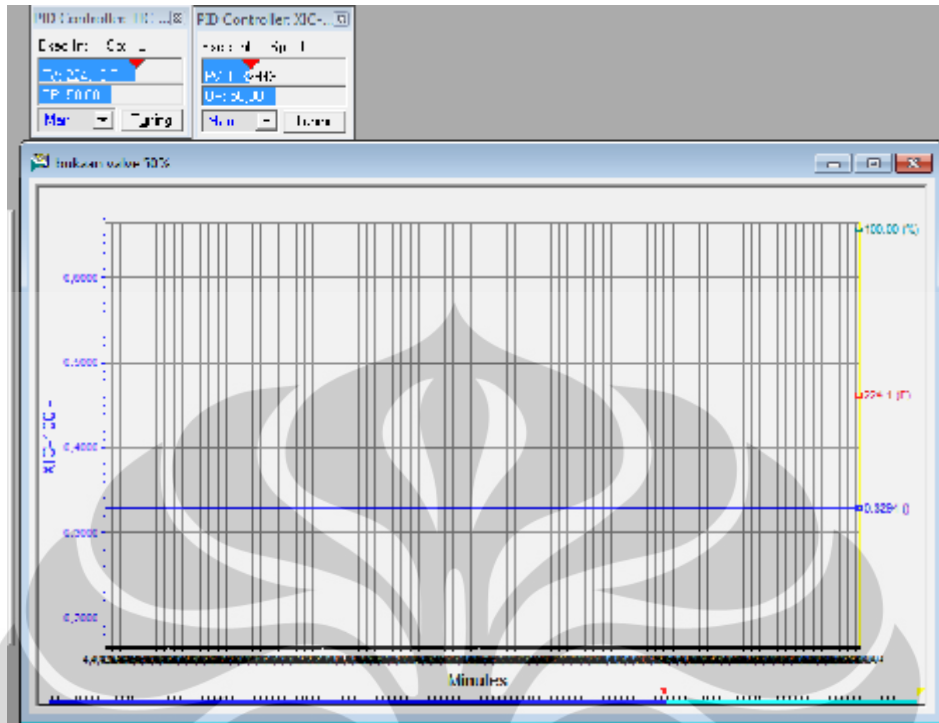


Gambar A.11 PRC Pengaruh Perubahan F_c Untuk Model D

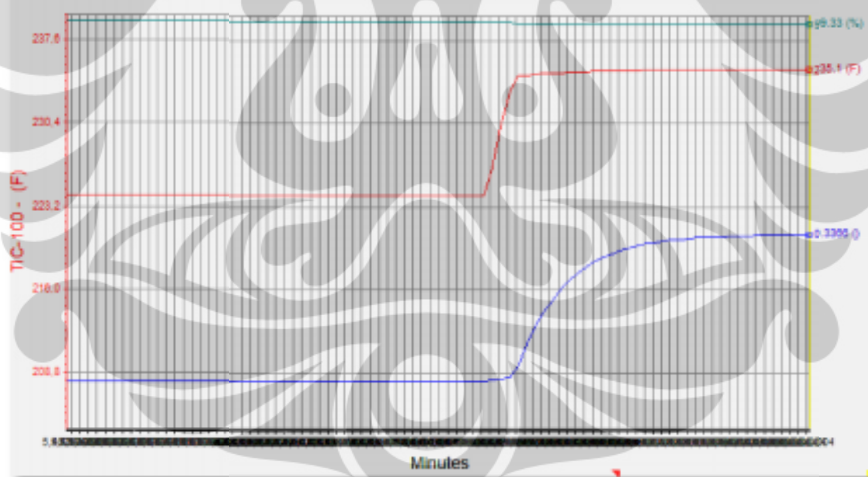


Gambar A.12. PRC Pengaruh Perubahan F_{po} Untuk Model D

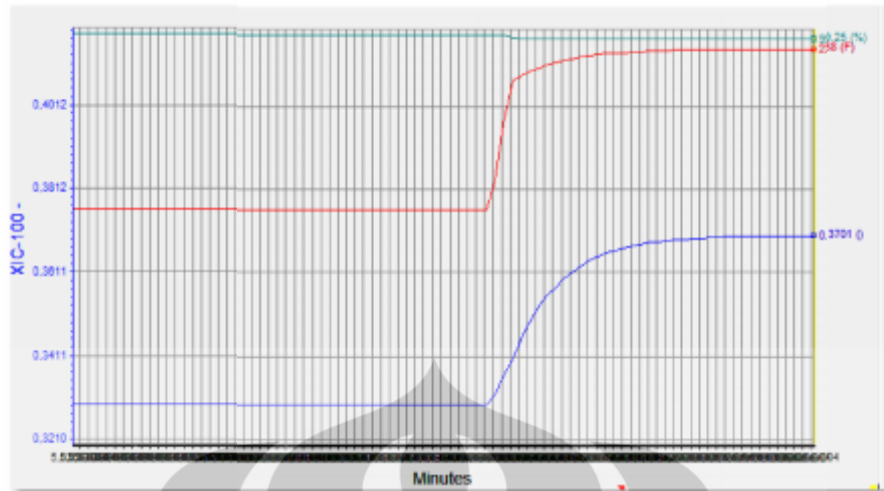
- Model E



Gambar A.13. Kondisi sistem pada bukaan valve 50%

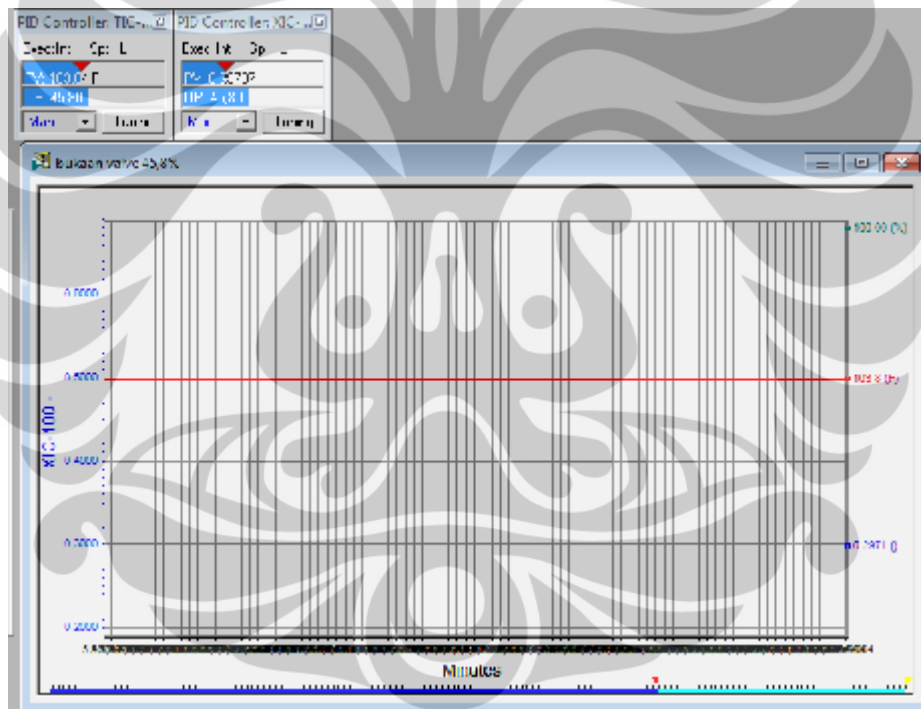


Gambar A.14. PRC Pengaruh Perubahan Fc Untuk Model E

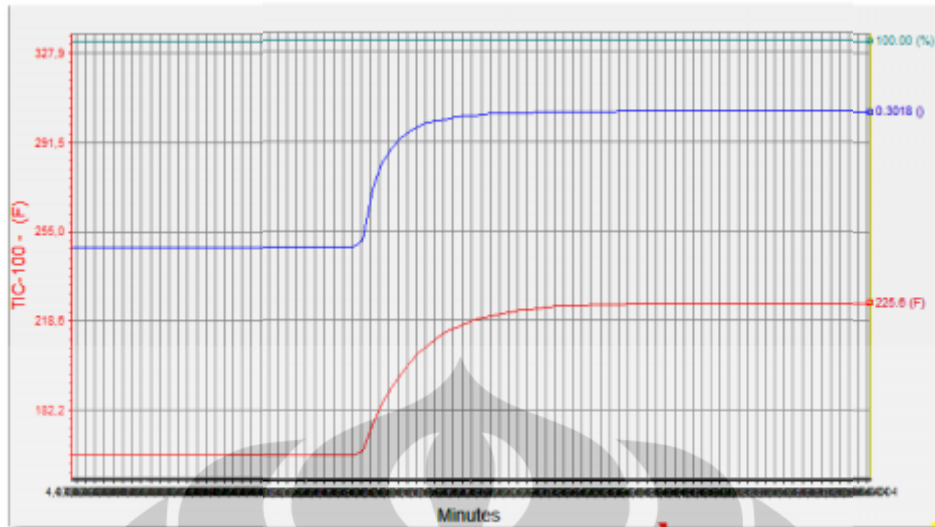


Gambar A.15. PRC Pengaruh Perubahan Fpo Untuk Model E

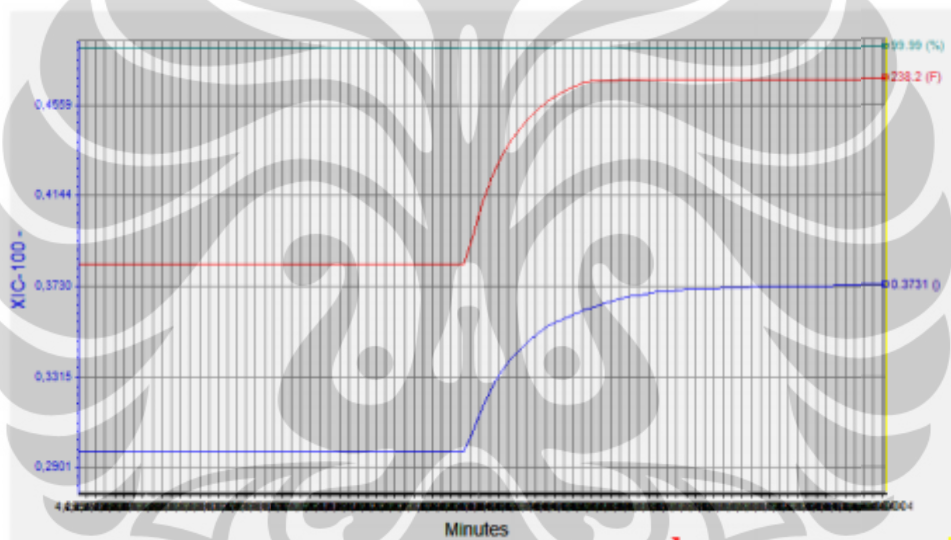
- Model F



Gambar A.16. Kondisi sistem pada bukaan valve 45,8%



Gambar A.17. PRC Pengaruh Perubahan F_c Untuk Model F



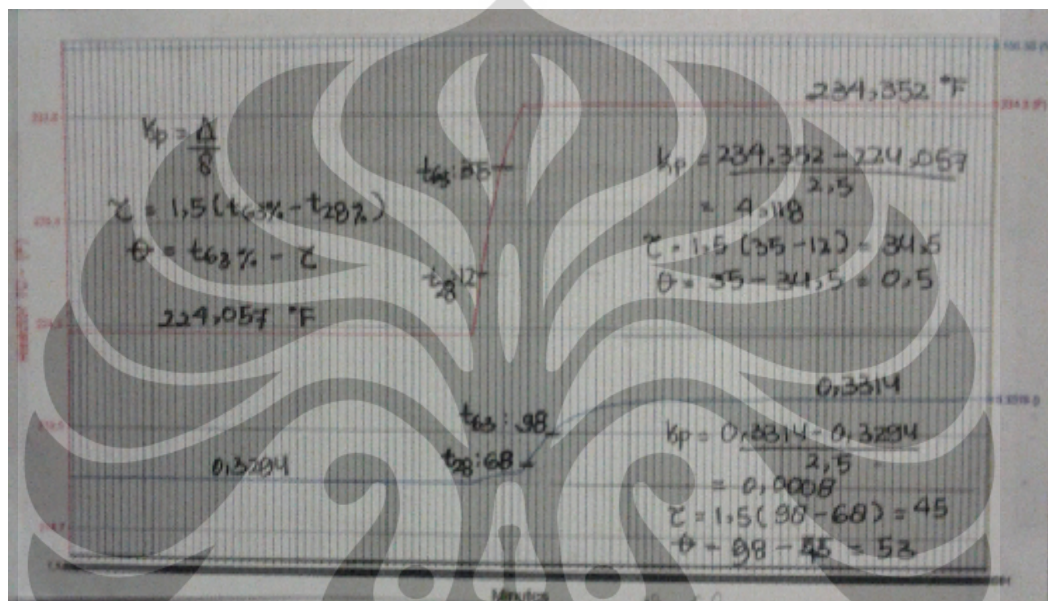
Gambar A.18. PRC Pengaruh Perubahan F_{po} Untuk Model F

LAMPIRAN B

Perhitungan parameter FOPDT

Parameter FOPDT ditentukan secara manual yang dihasilkan dari PRC. Berikut merupakan contoh penentuan parameter FOPDT untuk model A :

✓ Model testing pada TC



Gambar B.1. PRC Pengaruh Perubahan Fc Untuk Model A

- Respon T

Perubahan terhadap temperatur reaktor ditunjukkan pada garis grafik berwarna merah. Pada grafik tersebut diperoleh nilai K_p , τ dan θ sebagai berikut:

$$K_p = \frac{\Delta}{\delta} = \frac{234,352 - 224,057}{2,5} = 4,118$$

$$\tau = 1,5(t_{63\%} - t_{28\%}) = 1,5(35 - 12) = 34,5$$

$$\theta = t_{63\%} - \tau = 35 - 34,5 = 0,5$$

Nilai K_p , τ dan θ dimasukkan pada bentuk model FOPDT sehingga menghasilkan model sebagai berikut:

$$G_{1.1} = \frac{Y(s)}{X(s)} = \frac{4,118 e^{-0,5s}}{34,5 s + 1}$$

- Respon Xpg

Perubahan terhadap komposisi propylene glycol ditunjukkan pada garis grafik berwarna biru. Pada grafik tersebut diperoleh nilai K_p , τ dan θ sebagai berikut:

$$K_p = \frac{\Delta}{\delta} = \frac{0,3314 - 0,3294}{2,5} = 0,0008$$

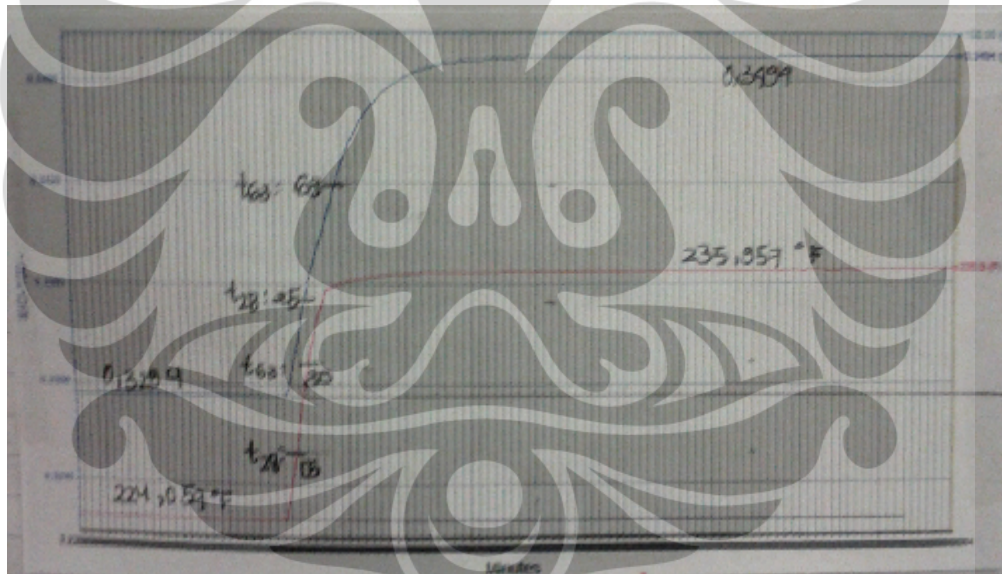
$$\tau = 1,5(t_{63\%} - t_{28\%}) = 1,5(98 - 68) = 45$$

$$\theta = t_{63\%} - \tau = 98 - 45 = 53$$

Nilai K_p , τ dan θ dimasukkan pada bentuk model FOPDT sehingga menghasilkan model sebagai berikut:

$$G_{1.2} = \frac{Y(s)}{X(s)} = \frac{0,0008 e^{-53s}}{45s + 1}$$

✓ Model testing pada AC



Gambar B.2. PRC Pengaruh Perubahan Fpo Untuk Model A

- Respon T

Perubahan terhadap temperatur reaktor ditunjukkan pada grafik berwarna merah. Pada grafik tersebut diperoleh nilai K_p , τ dan θ sebagai berikut:

$$K_p = \frac{\Delta}{\delta} = \frac{235,957 - 224,057}{2,5} = 4,76$$

$$\tau = 1,5(t_{63\%} - t_{28\%}) = 1,5(30 - 15) = 22,5$$

$$\theta = t_{63\%} - \tau = 30 - 22,5 = 7,5$$

Nilai K_p , τ dan θ dimasukkan pada bentuk model FOPDT sehingga menghasilkan model sebagai berikut:

$$G_{2.1} = \frac{Y(s)}{X(s)} = \frac{4,76 e^{-7,5s}}{22,5 s + 1}$$

- Respon X_{pg}

Perubahan terhadap komposisi propylene glycol ditunjukkan pada garis grafik berwarna biru. Pada grafik tersebut diperoleh nilai K_p , τ dan θ sebagai berikut:

$$K_p = \frac{\Delta}{\delta} = \frac{0,3494 - 0,3294}{2,5} = 0,008$$

$$\tau = 1,5(t_{63\%} - t_{28\%}) = 1,5(63 - 25) = 57$$

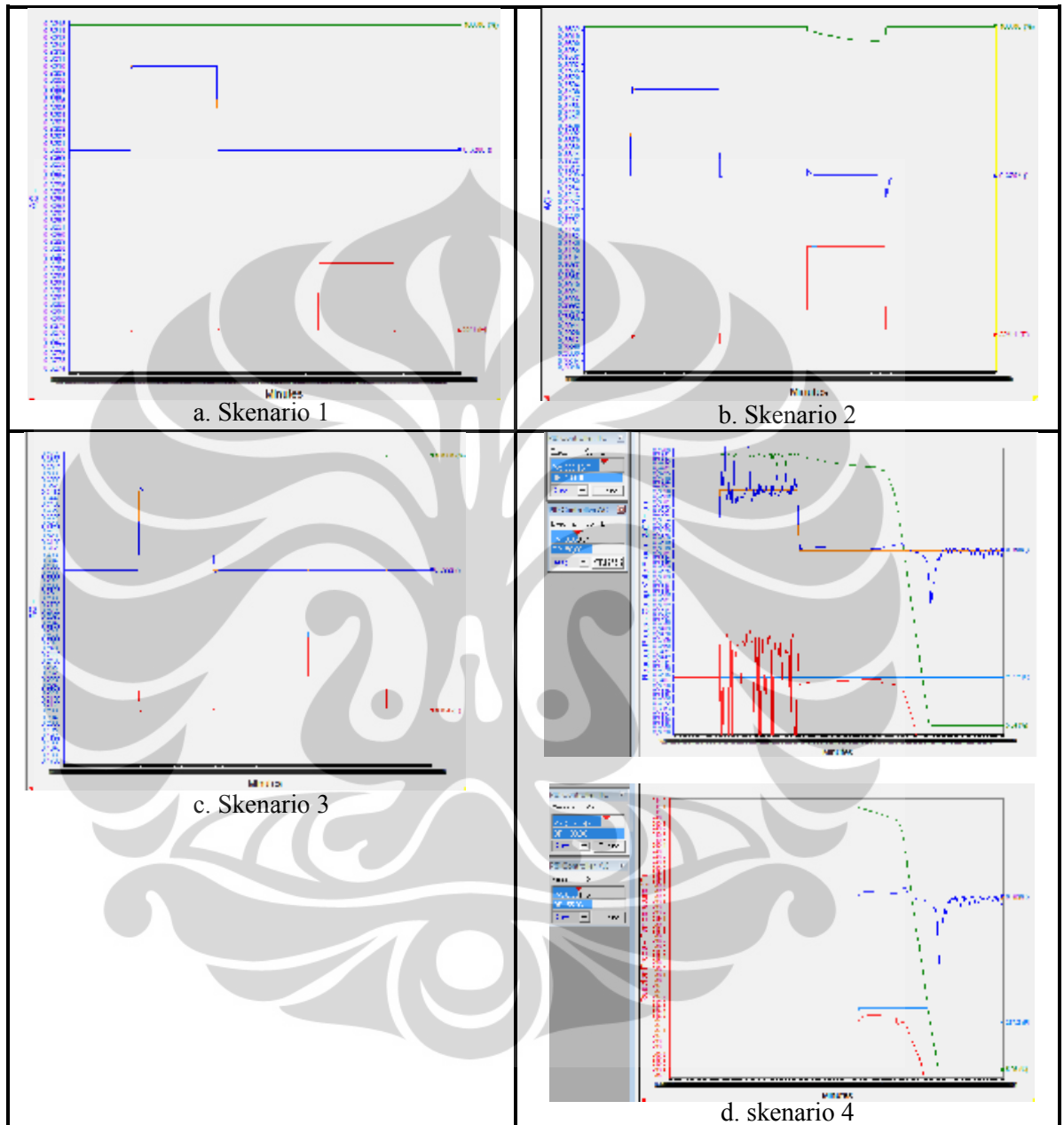
$$\theta = t_{63\%} - \tau = 63 - 25 = 6$$

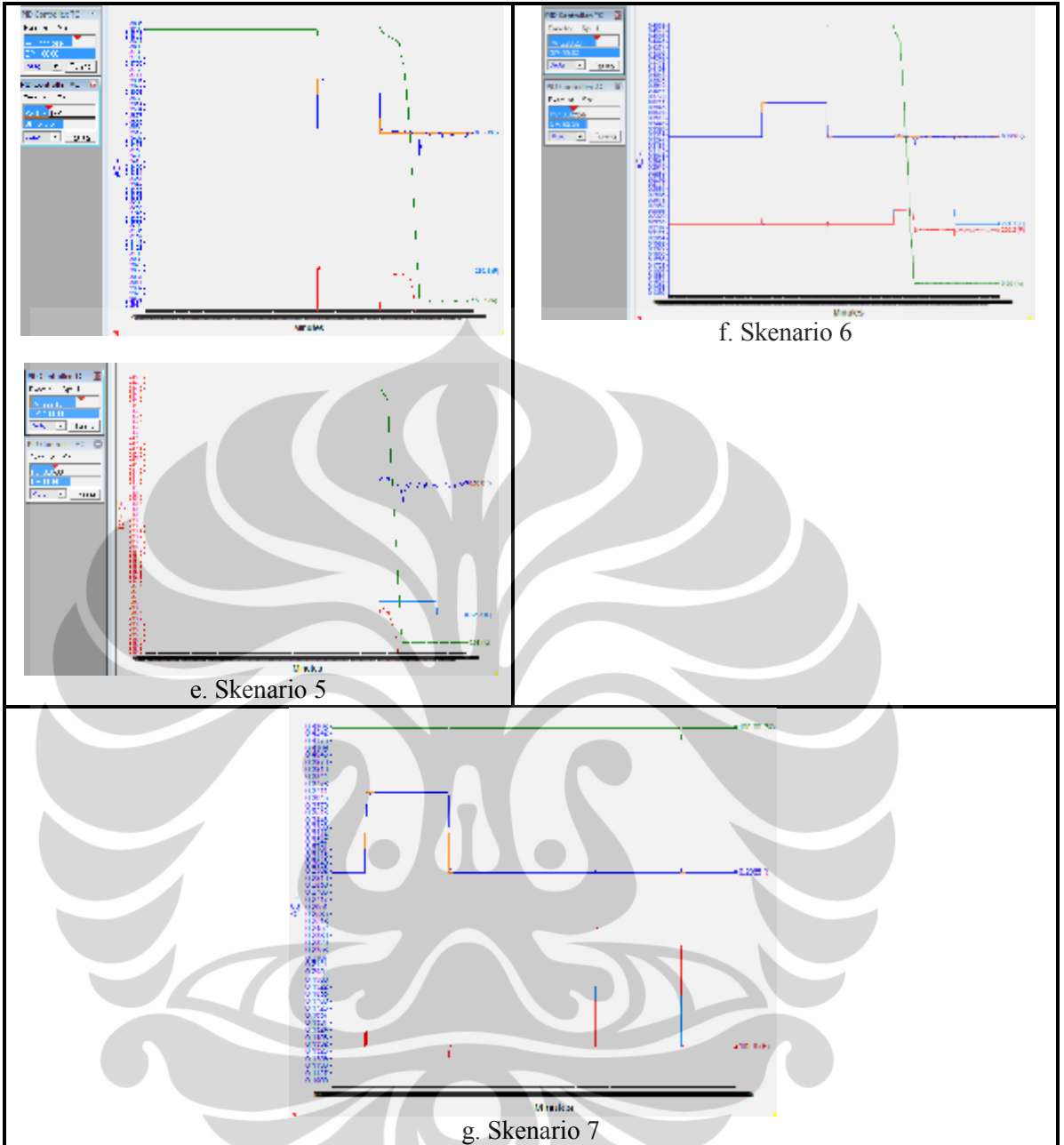
Nilai K_p , τ dan θ dimasukkan pada bentuk model FOPDT sehingga menghasilkan model sebagai berikut

$$G_{1.2} = \frac{Y(s)}{X(s)} = \frac{0,008 e^{-6s}}{57 s + 1}$$

LAMPIRAN C

Hasil tuning pengendali PI pada setiap skenario





LAMPIRAN D

Pengaruh perubahan parameter tuning MPC

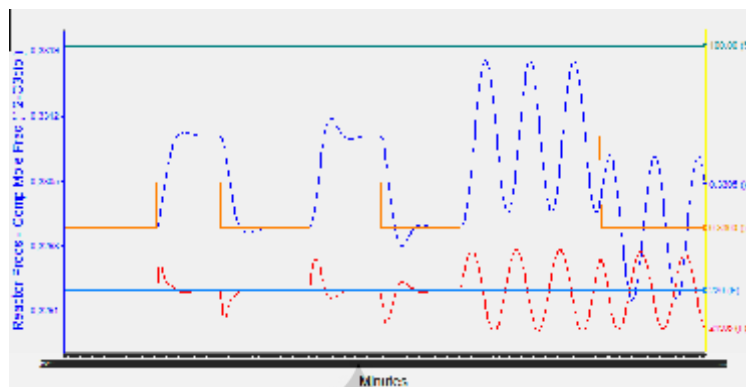
Parameter untuk *tuning* MPC berdasarkan simulasi pada perangkat lunak yang digunakan adalah *Step Resp. Length* (SRL), *Prediction Horizon* (P), *Control Horizon* (M), *Control Interval* (TS), *Ref. Trajectory*, *Gamma_U* dan *Gamma_Y*. Perubahan pada masing-masing parameter *tuning* dilakukan untuk melihat pengaruhnya pada performa MPC yang dihasilkan sehingga memudahkan *tuning* MPC pada langkah selanjutnya. Model pengendalian yang digunakan pada penentuan parameter *tuning* adalah Model A.

D.1. Perubahan parameter *Step Respon Length* (SRL)

Parameter SRL divariasikan pada nilai 25, 40 dan 100. Semakin tinggi nilai SRL, osilasi yang terjadi semakin besar dan pada SRL 100 osilasi berada pada rentang yang tetap. Akan tetapi penentuan parameter SRL ini juga dipengaruhi oleh nilai parameter *tuning* yang lain. Osilasi pada SRL 100 dihasilkan karena kondisi parameter *tuning* pada kondisi *default*-nya. Sedangkan jika parameter *tuning* lain diubah, maka parameter SRL dapat mempunyai batas nilai maksimum 400. Kondisi *tuning* parameter MPC pada perubahan nilai SRL ditampilkan pada Tabel C.1 dan hasil simulasi ditampilkan pada Gambar C.1.

Tabel D.1. Kondisi *tuning* MPC pada perubahan parameter SRL

Parameter <i>tuning</i>	1.1	1.2	1.3
SRL	25	40	100
P	25	25	25
M	2	2	2
Ts	30	30	30
<i>Ref. Trajectory</i>	1	1	1
<i>Gamma_U</i>	1	1	1
<i>Gamma_Y</i>	1	1	1



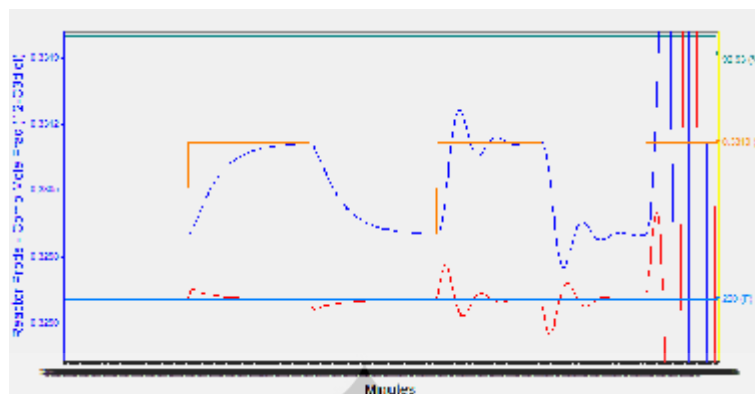
Gambar D.1. Hasil simulasi perubahan parameter SRL

D.2. Perubahan parameter *Prediction Horizon* (P)

Parameter *tuning* P mempunyai nilai maksimum pada nilai SRL yang ditetapkan. Kondisi SRL ditetapkan pada nilai 50 dan nilai P divariasikan pada nilai 20, 25 dan 30. Nilai P terlalu kecil menyebabkan respon yang terlalu lama, sedangkan nilai P yang terlalu besar menyebabkan osilasi yang menimbulkan kondisi tidak stabil. Kondisi tersebut berlaku jika parameter *tuning* lain ditentukan pada kondisi *default*, sedangkan sama seperti SRL, jika parameter lain di ubah kemungkinan respon yang dihasilkan akan berbeda. Kondisi *tuning* parameter MPC pada perubahan nilai P ditampilkan pada Tabel D.2 dan hasil simulasi ditampilkan pada Gambar D.2.

Tabel C.2. Kondisi *tuning* MPC pada perubahan parameter P

Parameter <i>tuning</i>	1.1	1.2	1.3
SRL	50	50	50
P	20	25	30
M	2	2	2
Ts	30	30	30
<i>Ref. Trajectory</i>	1	1	1
Gamma_U	1	1	1
Gamma_Y	1	1	1



Gambar D.2. Hasil simulasi perubahan parameter P

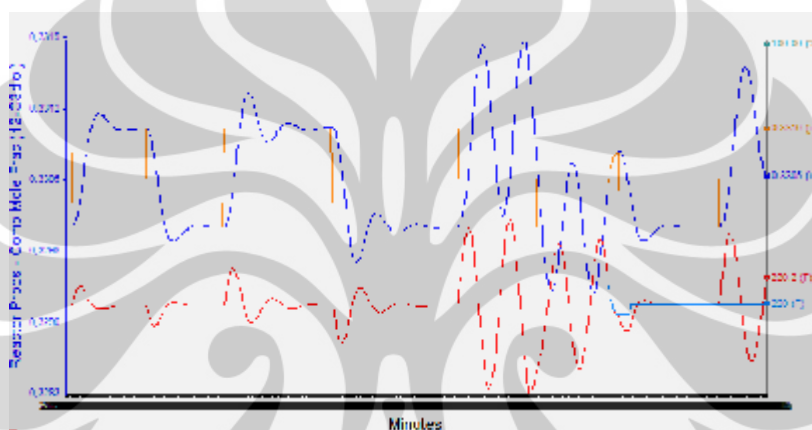
D.3. Perubahan parameter *Control Horizon* (M)

Parameter *tuning* M mempunyai nilai maksimum pada nilai P yang ditetapkan. Dengan kondisi tuning MPC *default*, nilai M divariasikan pada nilai 1, 2, 5, dan 25. Nilai M tidak bisa melebihi nilai 25 karena P ditetapkan pada nilai 25. Parameter *tuning* lainnya ditetapkan pada kondisi *default*-nya.

Hasil simulasi pada Gambar D.3 menunjukkan semakin besar nilai M, respon menghasilkan osilasi yang semakin tinggi, namun pada kondisi nilai M lebih dari 5 simulasi menghasilkan respon yang tetap. Hal ini dibuktikan dengan mensimulasikan kembali perbedaan nilai M dengan parameter *tuning* lain yang ditetapkan pada kondisi seperti ditampilkan pada Tabel D.4. Hasil simulasi ini ditampilkan pada Gambar D.4 yang menunjukkan nilai M sebesar 1 menghasilkan respon yang lebih keTsl *error*-nya dan nilai M sebesar 10 dan 20 menghasilkan respon yang sama atau tidak terjadi perubahan sama sekali namun tetap dengan adanya osilasi yang menghasilkan *error*. Mengacu pada hasil perubahan parameter *tuning* M, perubahan nilai M pada setiap model akan dilakukan karena adanya respon yang berbeda pada setiap model yang digunakan.

Tabel D.3. Kondisi *tuning* MPC pada perubahan parameter M pada kondisi *default*

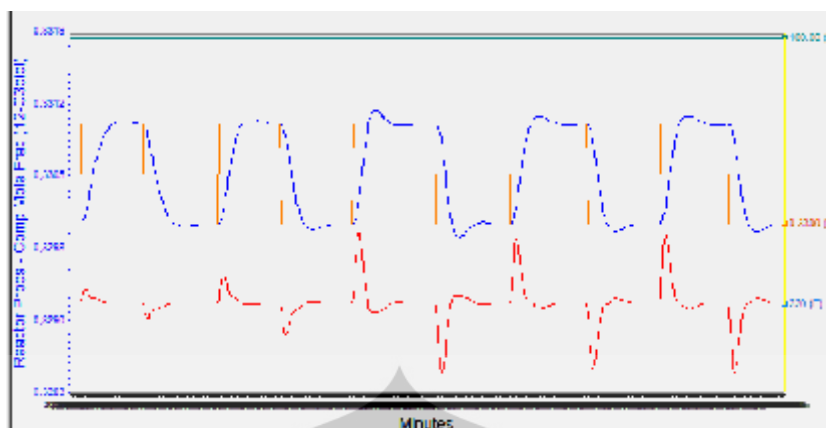
Parameter <i>tuning</i>	1.1	1.2	1.3	1.4
SRL	50	50	50	50
P	25	25	25	25
M	1	2	5	25
Ts	30	30	30	30
<i>Ref. Trajectory</i>	1	1	1	1
Gamma_U	1	1	1	1
Gamma_Y	1	1	1	1



Gambar D.3. Hasil simulasi perubahan parameter M pada kondisi *default*

Tabel D.4. Kondisi *tuning* MPC pada perubahan parameter M pada kondisi modifikasi

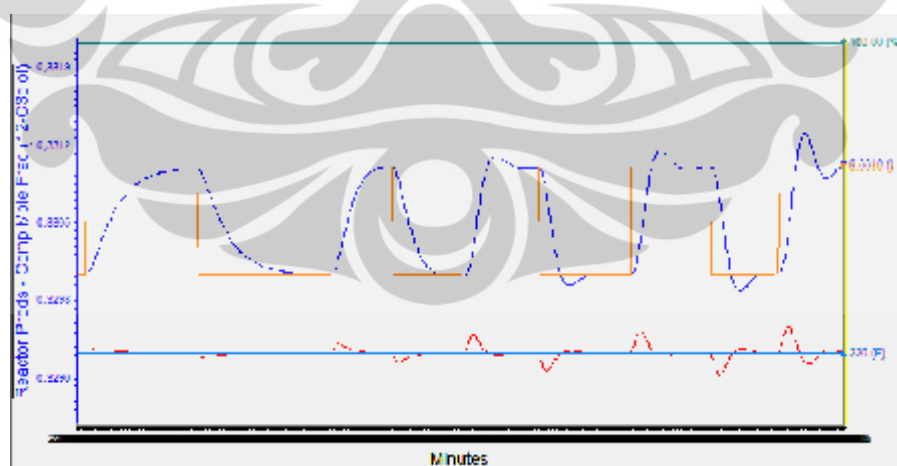
Parameter <i>tuning</i>	1.1	1.2	1.3	1.4	1.5
SRL	25	25	25	25	25
P	25	25	25	25	25
M	1	2	5	10	20
Ts	30	30	30	30	30
<i>Ref. Trajectory</i>	1	1	1	1	1
Gamma_U	1	1	1	1	1
Gamma_Y	1	1	1	1	1



Gambar D.4. Hasil simulasi perubahan parameter M pada kondisi modifikasi

D.4. Perubahan parameter *Control Interval* (T_s)

Parameter *tuning* T_s mempengaruhi seberapa cepat respon yang dihasilkan. Gambar menunjukkan dengan nilai T_s yang semakin ke T_s maka respon yang dihasilkan akan semakin lambat dan menyebabkan *error* yang cukup besar, sedangkan pada kondisi T_s yang semakin tinggi respon menghasilkan osilasi. Sehingga akan terdapat suatu batasan nilai T_s yang sangat dipengaruhi oleh nilai parameter *tuning* MPC lainnya. Kondisi *tuning* parameter MPC pada perubahan nilai T_s ditampilkan pada Tabel D.5 dan hasil simulasi ditampilkan pada Gambar D.5.



Gambar D.5. Hasil simulasi perubahan parameter T_s

Tabel D.5. Kondisi *tuning* MPC pada perubahan parameter TS

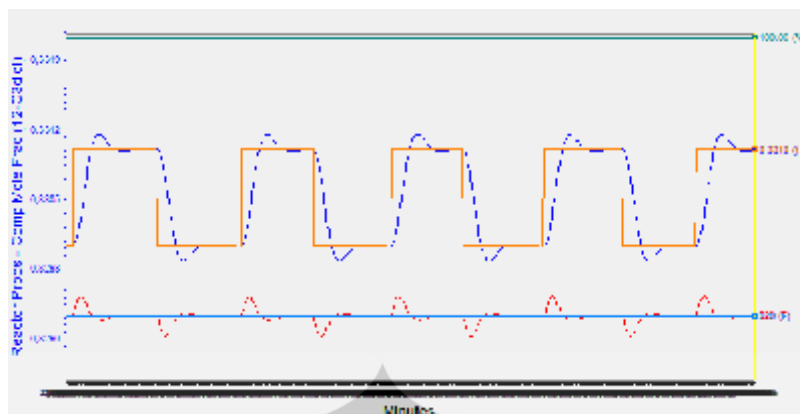
Parameter tuning	1.1	1.2	1.3	1.4	1.5
SRL	50	50	50	50	50
P	25	25	25	25	25
M	1	1	1	1	1
Ts	22	25	28	30	31
<i>Ref. Trajectory</i>	1	1	1	1	1
Gamma_U	1	1	1	1	1
Gamma_Y	1	1	1	1	1

D.5. Perubahan parameter *Ref. Trajectory*

Parameter *tuning Ref. Trajectory* dapat divariasikan dari nilai 1-450, namun variasi nilai *Ref. Trajectory* hanya dilakukan pada nilai 1, 2, 5, 10 dan 20. Hasil simulasi menunjukkan respon yang tidak mengalami perubahan sama sekali, sehingga pada *tuning* parameter MPC selanjutnya, nilai *Ref. Trajectory* tetap berada pada nilai *default* yaitu sebesar 1. Kondisi *tuning* parameter MPC pada perubahan nilai *Ref. Trajectory* ditampilkan pada Tabel D.6 dan hasil simulasi ditampilkan pada Gambar D.6.

Tabel D.6. Kondisi *tuning* MPC pada perubahan parameter *Ref. Trajectory*

Parameter tuning	1.1	1.2	1.3	1.4	1.5
SRL	50	50	50	50	50
P	25	25	25	25	25
M	1	1	1	1	1
Ts	30	30	30	30	30
<i>Ref. Trajectory</i>	1	2	5	10	20
Gamma_U	1	1	1	1	1
Gamma_Y	1	1	1	1	1



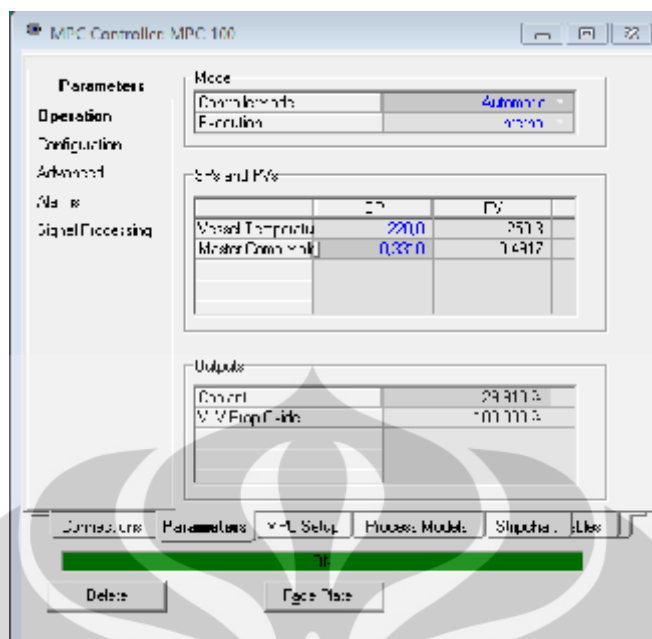
Gambar D.6. Hasil simulasi perubahan parameter *Ref. Trajectory*

D.6. Perubahan parameter Γ_U

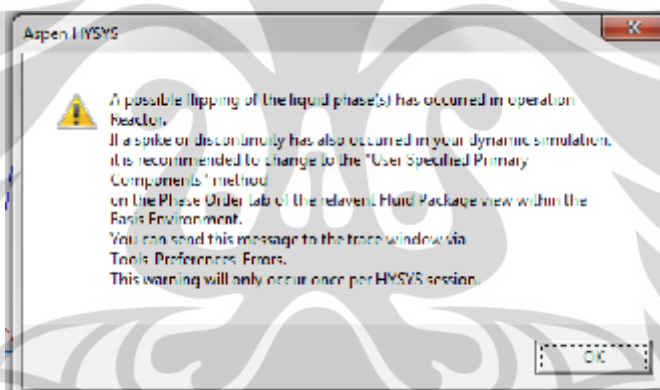
Parameter *tuning* Gamma-U divariasikan dari nilai 0,1-1 karena nilai maksimum berada pada nilai 1. Pada nilai Γ_U sebesar 0,1 sampai 0,5, simulasi menghasilkan keadaan *error* atau tidak stabil. Hal ini ditandai dengan bukaan umpan propylene oxide sudah 100% dan tidak diimbangi oleh bukaan *coolant* sehingga tidak adanya kesetimbangan antara panas yang dibutuhkan oleh CSTR dengan masukan umpan. Kondisi tersebut ditunjukkan pada Gambar D.7. Kondisi tidak stabil ini juga ditandai dengan adanya peringatan pada perangkat lunak yang menyatakan bahwa telah terjadi *flipping* pada fasa cairnya seperti Gambar D.8.

Tabel D.7. Kondisi *tuning* MPC pada perubahan parameter Γ_U

Parameter <i>tuning</i>	1.1	1.2	1.3	1.4	1.5	1.6
SRL	50	50	50	50	50	50
P	25	25	25	25	25	25
M	1	1	1	1	1	1
Ts	30	30	30	30	30	30
<i>Ref. Trajectory</i>	1	1	1	1	1	1
Γ_U	0,1-0,5	0,6	0,7	0,8	0,9	1
Γ_Y	1	1	1	1	1	1

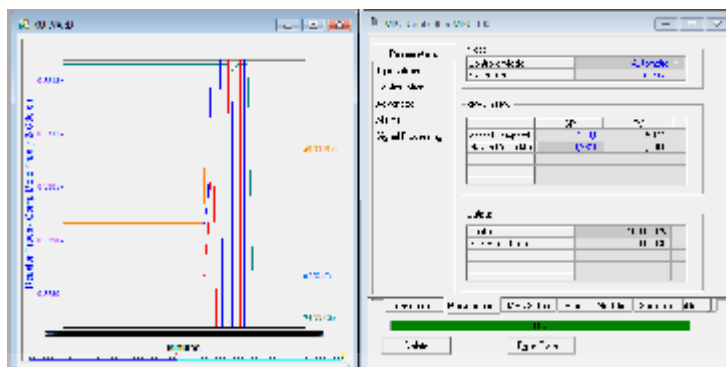


Gambar D.7. Kondisi bukaan *valve* pada perubahan parameter Γ_U [0,1-0,5]



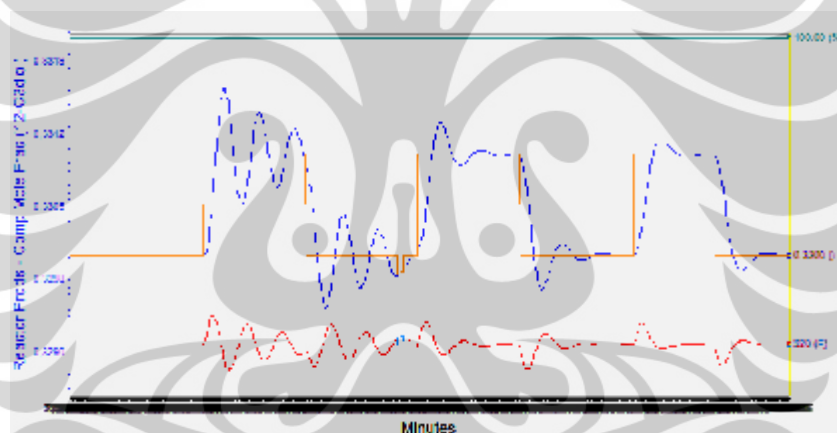
Gambar D.8. Peringatan dari simulasi perangkat lunak

Nilai Γ_U sebesar 0,6 dan 0,7 menghasilkan kondisi PV yang menjauhi SP. Bukaan *valve coolant* menjadi sebesar 100% menyebabkan aliran pendingin masuk melebihi set pointnya sehingga temperatur CSTR turun pada kondisi kurang dari 59,33 °C. Selain itu, bukaan *valve umpan* sebesar 0% menyebabkan umpan propylene oxide tidak ada yang masuk ke dalam CSTR sehingga *liquid percent level* turun menjadi 4,99%. Kedua kondisi tersebut menyebabkan tidak adanya produk yang terbentuk dan terbaca PV konsentrasi mol komponen propylene glycol sebesar 0. Hasil simulasi dengan Γ_U sebesar 0,6 dan 0,7 ditampilkan pada Gambar D.9.



Gambar D.9. Hasil simulasi perubahan parameter Γ_U sebesar 0,6 dan 0,7

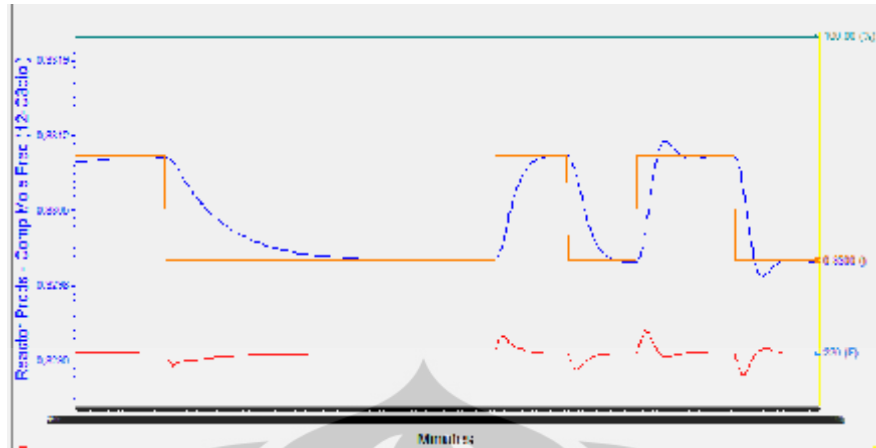
Nilai Γ_U sebesar 0,8-0,9-1 ditampilkan pada Gambar D.10. Semakin besar nilai Γ_U , osilasi yang terjadi semakin kecil, dan paling stabil ditunjukkan pada Grafik dengan Γ_U sebesar 1 sehingga pada *tuning* selanjutnya Γ_U tetap pada kondisi *default*-nya yaitu 1.



Gambar D.10. Hasil simulasi perubahan parameter Γ_U [0,8-0,9-1]

D.7. Perubahan parameter Γ_Y

Tuning parameter Γ_Y hampir sama dengan Γ_U . Nilai Γ_Y mempunyai batasan 0,1 sampai 1. Nilai Γ_Y yang kurang dari 1 menghasilkan respon yang sangat lama dan error-nya besar seperti contoh yang ditampilkan pada Gambar dengan kondisi Γ_Y sebesar 0,5. Sehingga pada *tuning* selanjutnya Γ_Y sama seperti Γ_U yang ditetapkan pada kondisi *default*-nya yaitu 1. Kondisi *tuning* parameter MPC pada perubahan nilai Γ_Y ditampilkan pada Tabel D.8 dan hasil simulasi ditampilkan pada Gambar D.11.



Gambar D.11. Hasil simulasi perubahan parameter Γ_Y

Tabel D.8. Kondisi *tuning* MPC pada perubahan parameter Γ_Y

Parameter tuning	1.1	1.2	1.3
SRL	50	50	50
P	25	25	25
M	1	1	1
Ts	30	30	30
<i>Ref. Trajectory</i>	1	1	1
Gamma_U	1	1	1
Gamma_Y	0,5	0,8	1

LAMPIRAN E

Perhitungan parameter tuning MPC menggunakan strategi *tuning non-adaptive* (Dougherty, 2003a)

E.1. Perhitungan pada Model 1

FOPDT pada model A sesuai PRC hasil *model testing defaults* pada Tabel E.1 berikut ini:

Tabel E.1 Parameter FOPDT model A

FOPDT	Kp	Tp	Delay
G 1.1	4,118	34,5	0,5
G 1.2	0,0008	45	53
G 2.1	4,759	22,5	7,5
G 2.2	0,008	57	6

Waktu sample (T)

$$T_{rs} = \text{Max}(0,1 \quad T_{rs} \quad T_{rs})$$

$$T_{11} = 0,1.34,5 = 3,45 [\text{max}]$$

$$= 0,5.0,5 = 0,25$$

$$T_{12} = 0,1.45 = 4,5$$

$$= 0,5.53 = 26,5 [\text{max}]$$

$$T_{21} = 0,1.22,5 = 2,25$$

$$= 0,5.7,5 = 3,75 [\text{max}]$$

$$T_{22} = 0,1.57 = 5,7 [\text{max}]$$

$$= 0,5.6 = 3$$

$$\text{Jadi, } T = \text{Min}(T_{rs}) = 3,45 \text{ menit}$$

Prediction horizon (P)

$$P = \text{Max} \left(\frac{5}{T} + k_{rs} \right) \text{ dengan nilai } k_{rs} = \left(\frac{rs}{T} + 1 \right)$$

$$k_{11} = \left(\frac{0,5}{3,45} + 1 \right) = 1,1449 \rightarrow P = \left(\frac{5 \cdot 34,5}{3,45} + 1,1449 \right) = 51,1449$$

$$k_{12} = \left(\frac{53}{3,45} + 1 \right) = 16,362 \rightarrow P = \left(\frac{5 \cdot 45}{3,45} + 16,362 \right) = 81,5794$$

$$k_{21} = \left(\frac{7,5}{3,45} + 1 \right) = 3,1739 \rightarrow P = \left(\frac{5 \cdot 22,5}{3,45} + 3,1739 \right) = 35,7826$$

$$k_{22} = \left(\frac{6}{3,45} + 1 \right) = 2,739 \rightarrow P = \left(\frac{5 \cdot 57}{3,45} + 2,739 \right) = 85,3477 [\text{max}]$$

jadi, nilai P adalah 85,3477

Control horizon (P)

$$M = \text{Max} \left(\frac{\tau_{rs}}{T} + k_{rs} \right)$$

$$M_{11} = \left(\frac{34,5}{3,45} + 1,1449 \right) = 11,1449$$

$$M_{12} = \left(\frac{45}{3,45} + 16,362 \right) = 29,405 [\text{max}]$$

$$M_{21} = \left(\frac{22,5}{3,45} + 3,1739 \right) = 9,6956$$

$$M_{22} = \left(\frac{57}{3,45} + 2,739 \right) = 19,260$$

jadi, nilai M adalah 29,405

E.2. Hasil perhitungan parameter tuning pada setiap model

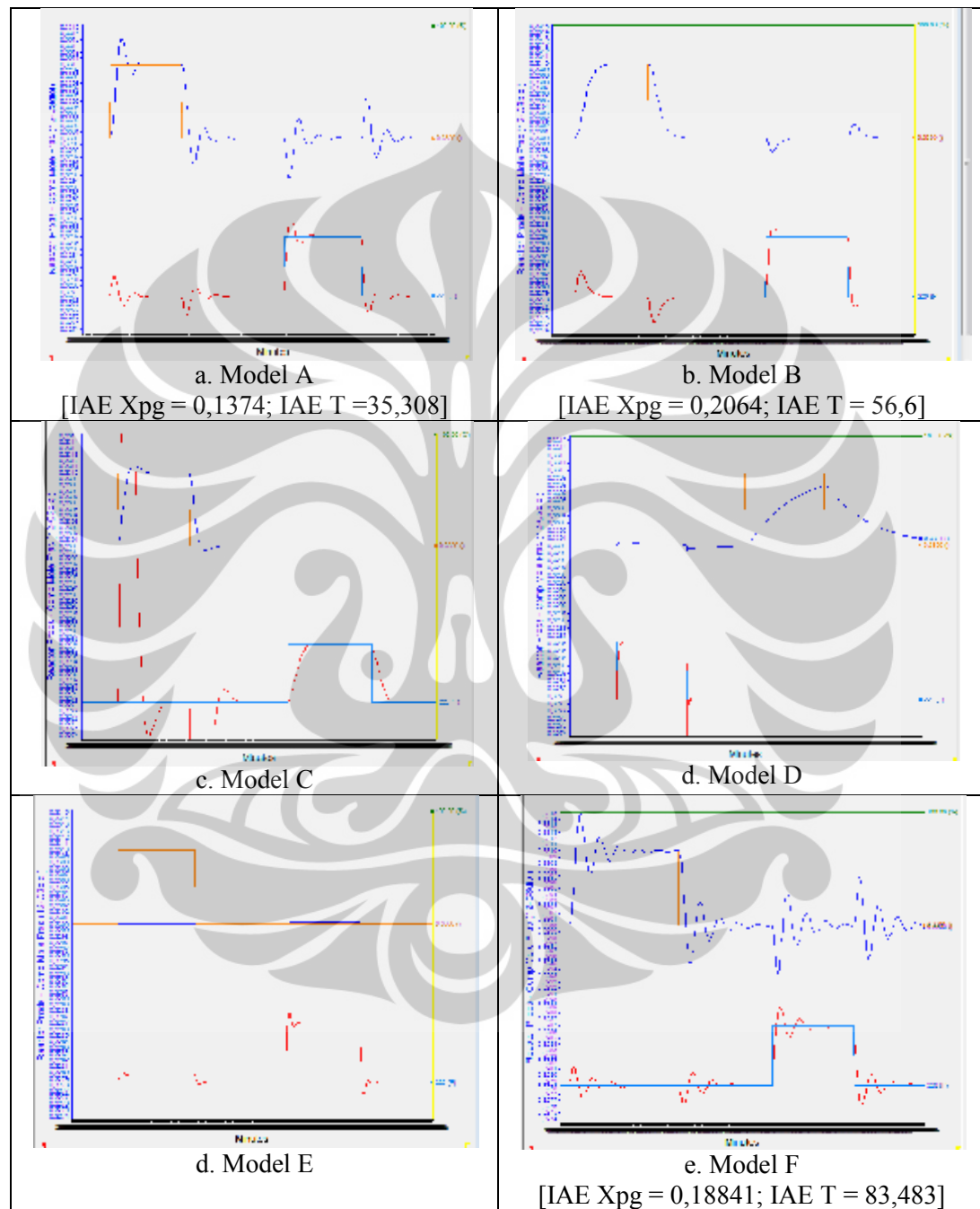
Hasil perhitungan parameter tuning MPC *multivariable* dengan strategi *tuning non-adaptive* DMC ditampilkan pada Tabel E.2 berikut ini:

Tabel E.2. Hasil perhitungan parameter tuning MPC *multivariable* dengan strategi *tuning non-adaptive*

Parameter		Kp	τ_p	Θ	Trs	0,1 x τ_p	0,5 x Θ	Trs	T	Krs	Prs	P	Mrs	M		
MODEL A	G 1.1	4,118	34,5	0,5	T11	3,45	0,25	3,45	3,45	K11	1,144928	51,14493	85,34783	M11	11,14493	29,4058
	G 1.2	0,0008	45	53	T12	4,5	26,5	26,5		K12	16,36232	81,57971		M12	29,4058	
	G 2.1	4,759	22,5	7,5	T21	2,25	3,75	3,75		K21	3,173913	35,78261		M21	9,695652	
	G 2.2	0,008	57	6	T22	5,7	3	5,7		K22	2,73913	85,34783		M22	19,26087	
MODEL B	G 1.1	6,232344	45	10	T11	4,5	5	5	4,75	K11	3,898551	69,11594	79,26087	M11	16,94203	18,3913
	G 1.2	0,000628	25,5	9,5	T12	2,55	4,75	4,75		K12	3,753623	40,71014		M12	11,14493	
	G 2.1	8,299731	45	10	T21	4,5	5	5		K21	3,898551	69,11594		M21	16,94203	
	G 2.2	0,00727	52,5	7,5	T22	5,25	3,75	5,25		K22	3,173913	79,26087		M22	18,3913	
MODEL C	G 1.1	0,279	54	11	T11	5,4	5,5	5,5	4,65	K11	4,188406	82,44928	86,21739	M11	19,84058	19,84058
	G 1.2	0,0031	51	11	T12	5,1	5,5	5,5		K12	4,188406	78,10145		M12	18,97101	
	G 2.1	0,971	46,5	5,5	T21	4,65	2,75	4,65		K21	2,594203	69,98551		M21	16,07246	
	G 2.2	0,01065	58,5	1,5	T22	5,85	0,75	5,85		K22	1,434783	86,21739		M22	18,3913	
MODEL D	G 1.1	0,263333	55,5	4,5	T11	5,55	2,25	5,55	5,55	K11	2,304348	82,73913	90,85507	M11	18,3913	21,28986
	G 1.2	0,002889	57	1	T12	5,7	0,5	5,7		K12	1,289855	83,89855		M12	17,81159	
	G 2.1	0,946111	57	13	T21	5,7	6,5	6,5		K21	4,768116	87,37681		M21	21,28986	
	G 2.2	0,010444	60	10	T22	6	5	6		K22	3,898551	90,85507		M22	21,28986	
MODEL E	G 1.1	2,439778	19,5	5,5	T11	1,95	2,75	2,75	2,75	K11	2,594203	30,85507	91,43478	M11	8,246377	27,08696
	G 1.2	0,0016	52,5	37,5	T12	5,25	18,75	18,75		K12	11,86957	87,95652		M12	27,08696	
	G 2.1	3,096222	16,5	8,5	T21	1,65	4,25	4,25		K21	3,463768	27,37681		M21	8,246377	
	G 2.2	0,009044	60	12	T22	6	6	6		K22	4,478261	91,43478		M22	21,86957	
MODEL F	G 1.1	6,110428	55,5	6,5	T11	5,55	3,25	5,55	4,75	K11	2,884058	83,31884	83,31884	M11	18,97101	18,97101
	G 1.2	0,000513	18	12	T12	1,8	6	6		K12	4,478261	30,56522		M12	9,695652	
	G 2.1	7,383182	40,5	9,5	T21	4,05	4,75	4,75		K21	3,753623	62,44928		M21	15,49275	
	G 2.2	0,007546	55,5	4,5	T22	5,55	2,25	5,55		K22	2,304348	82,73913		M22	18,3913	

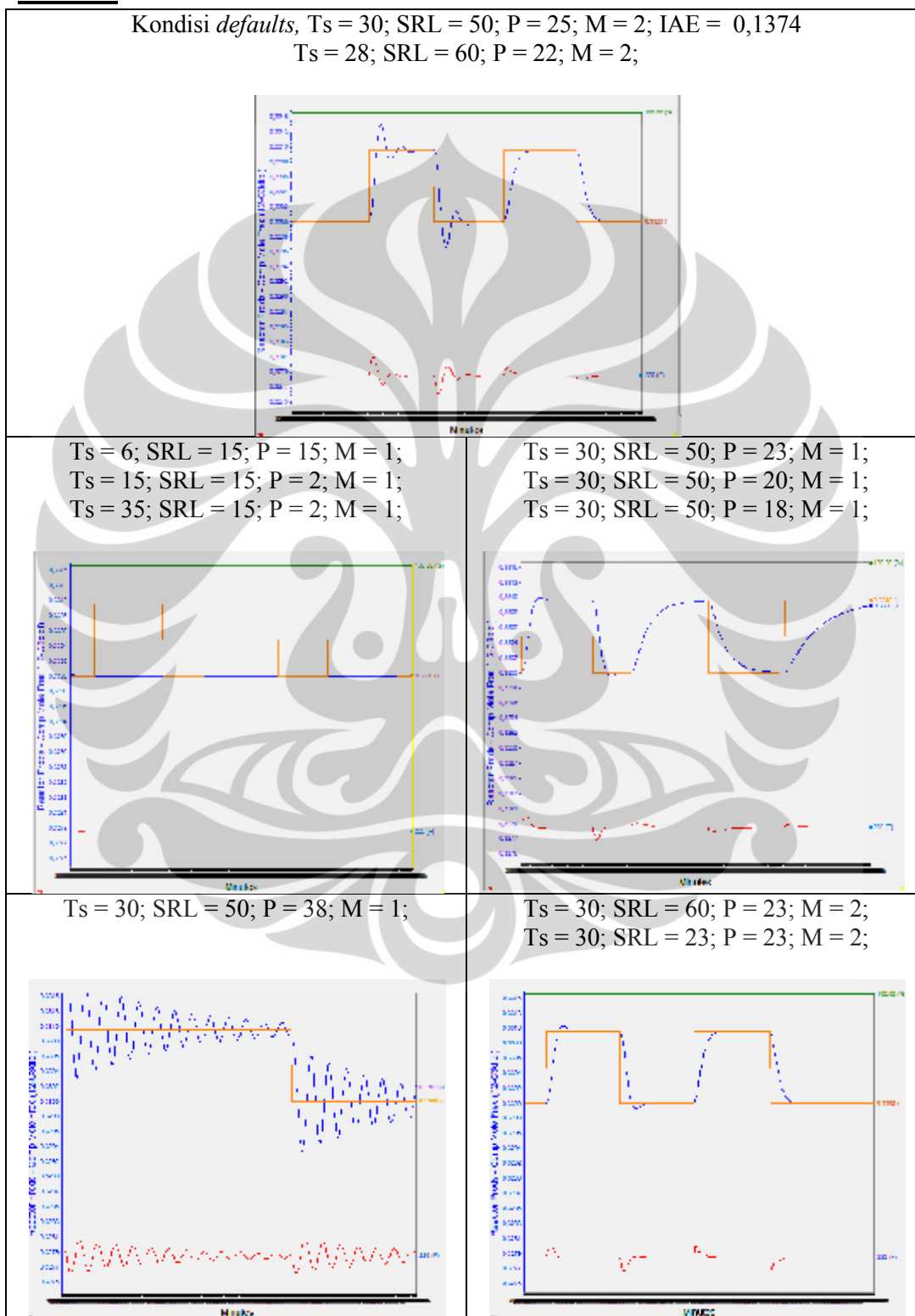
LAMPIRAN F

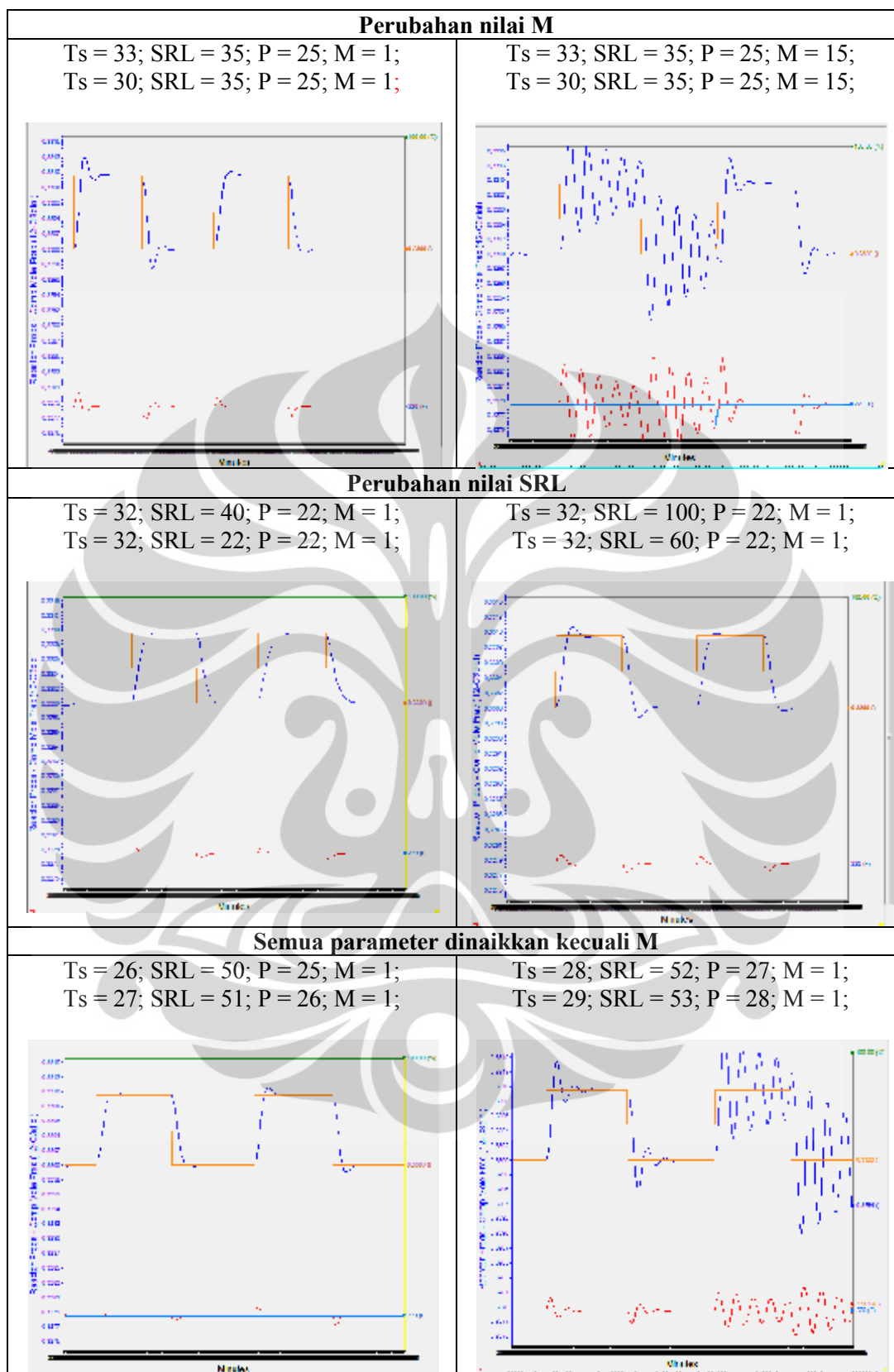
Kinerja Pengendali MPC *multivariable* pada setiap model
dengan skenario 1 dan parameter tuning *default*

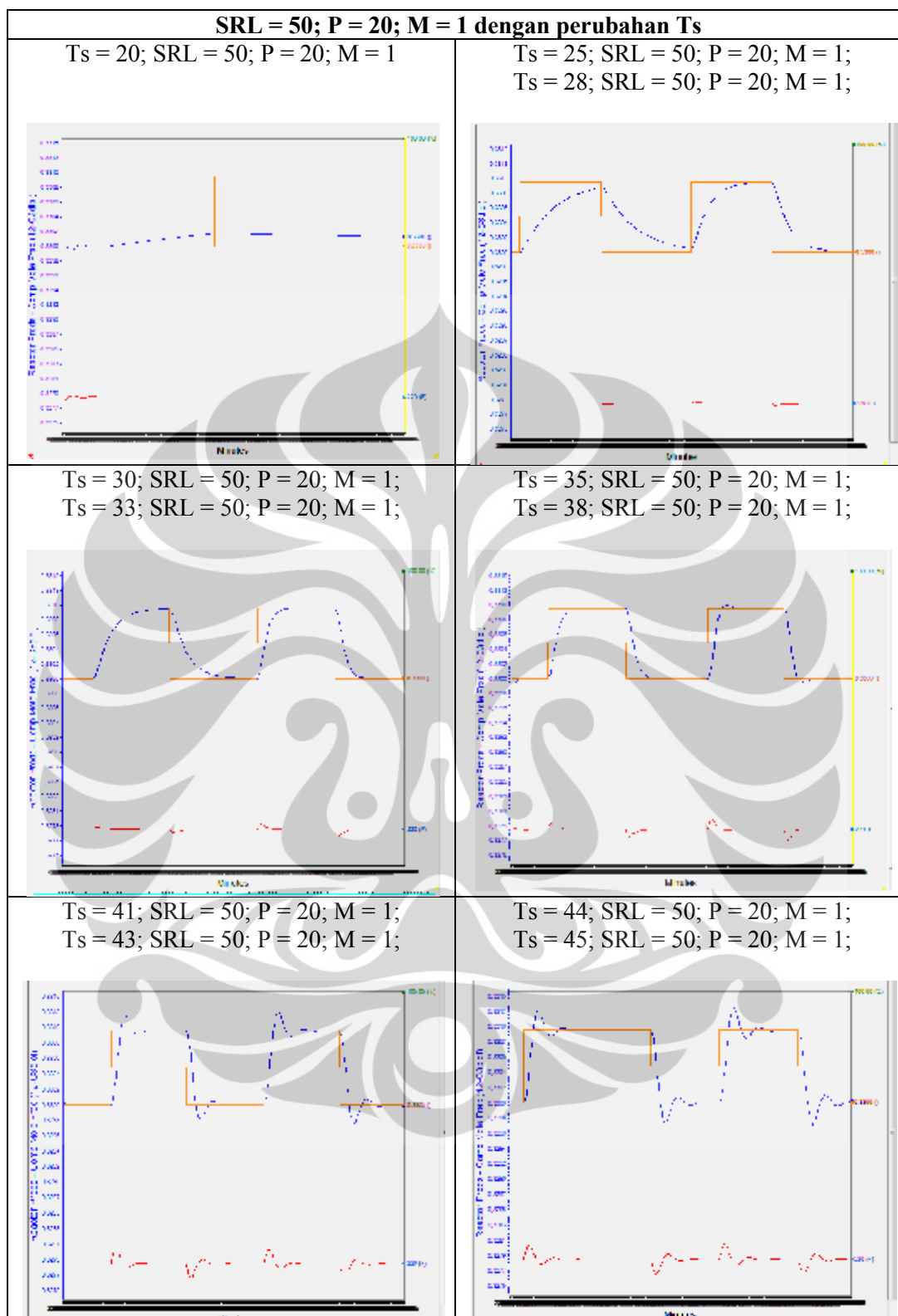


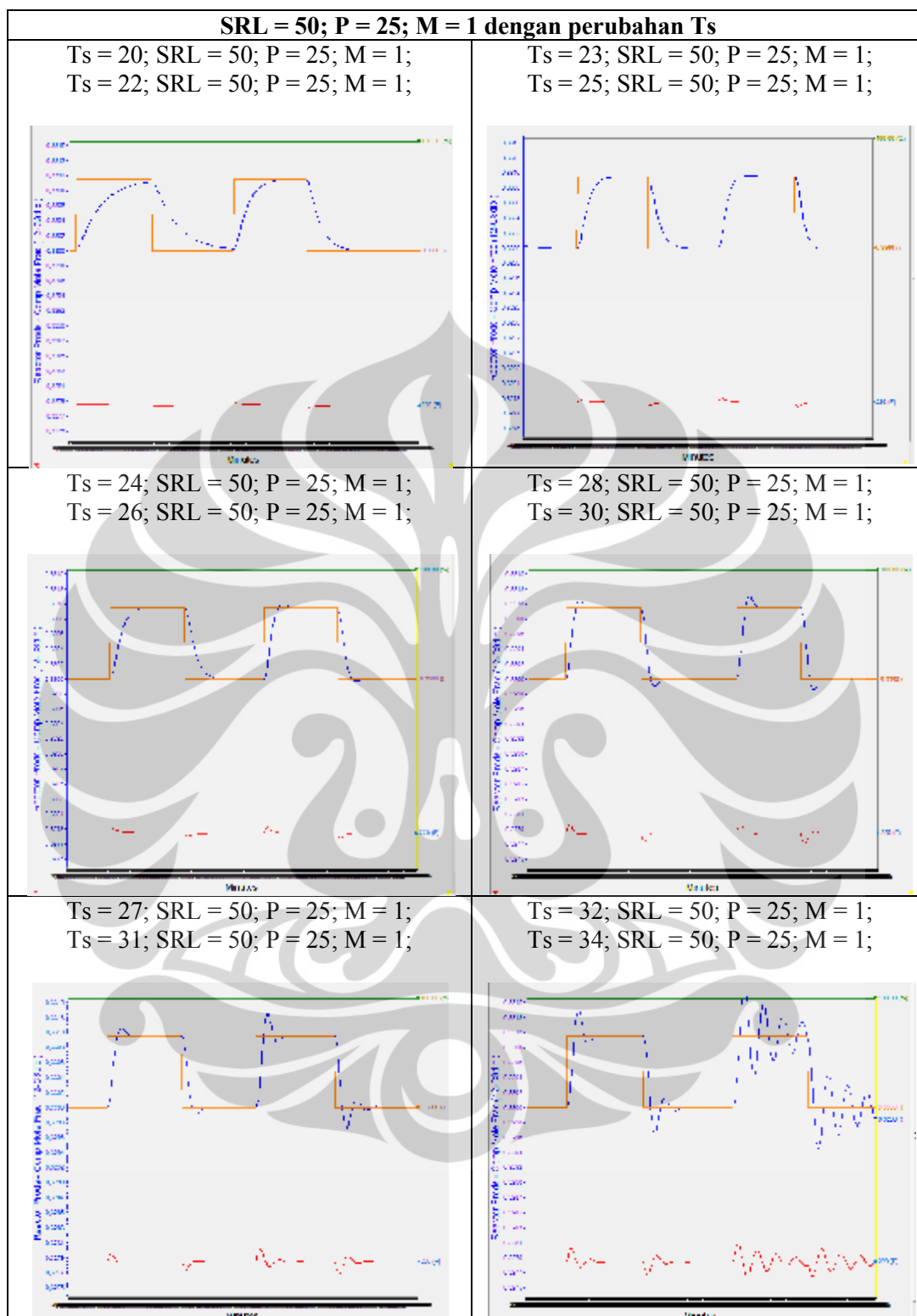
LAMPIRAN G

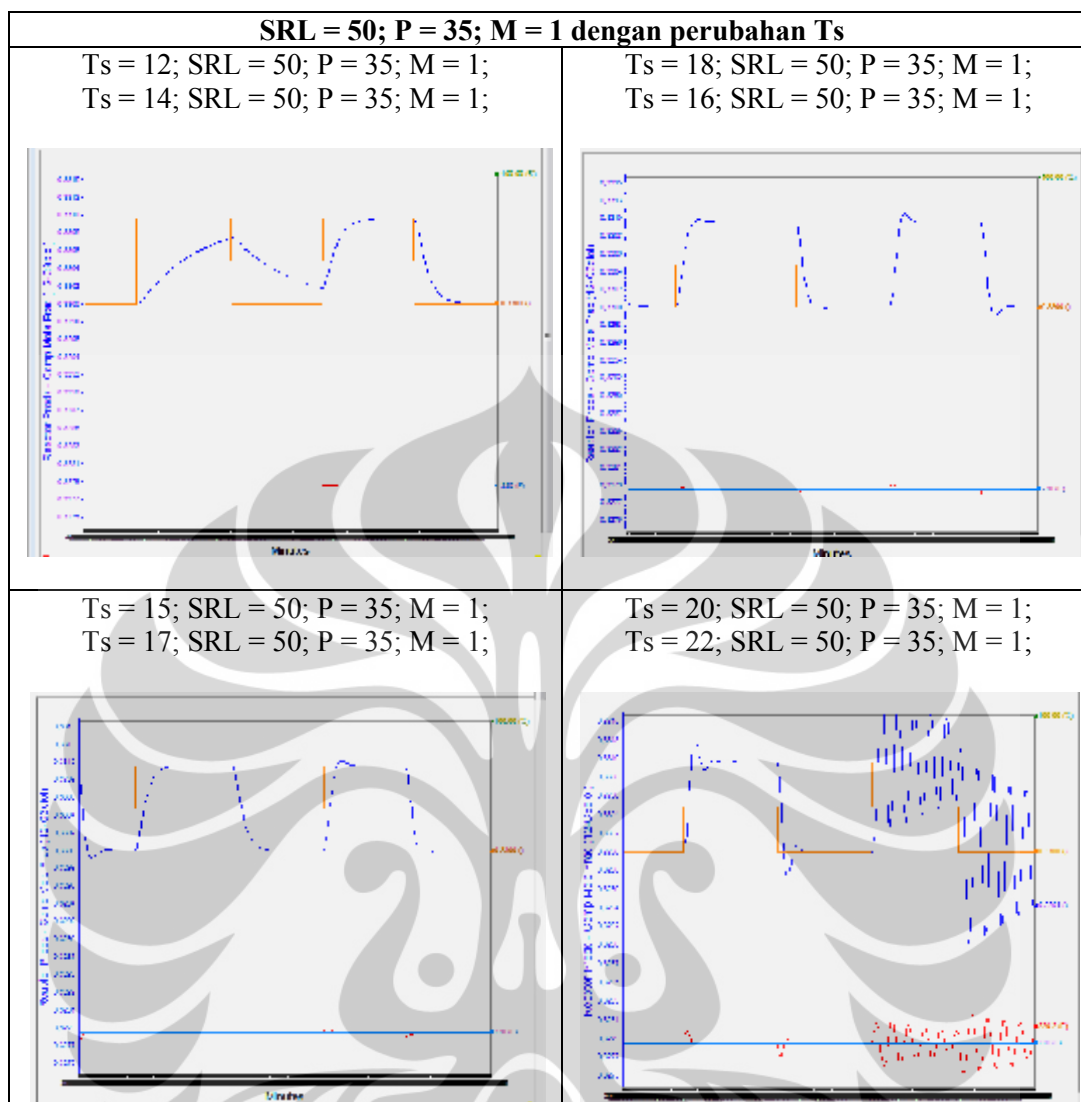
**Respon CV Pada Tuning Parameter MPC Skenario 1
dengan Model A dan Model F**

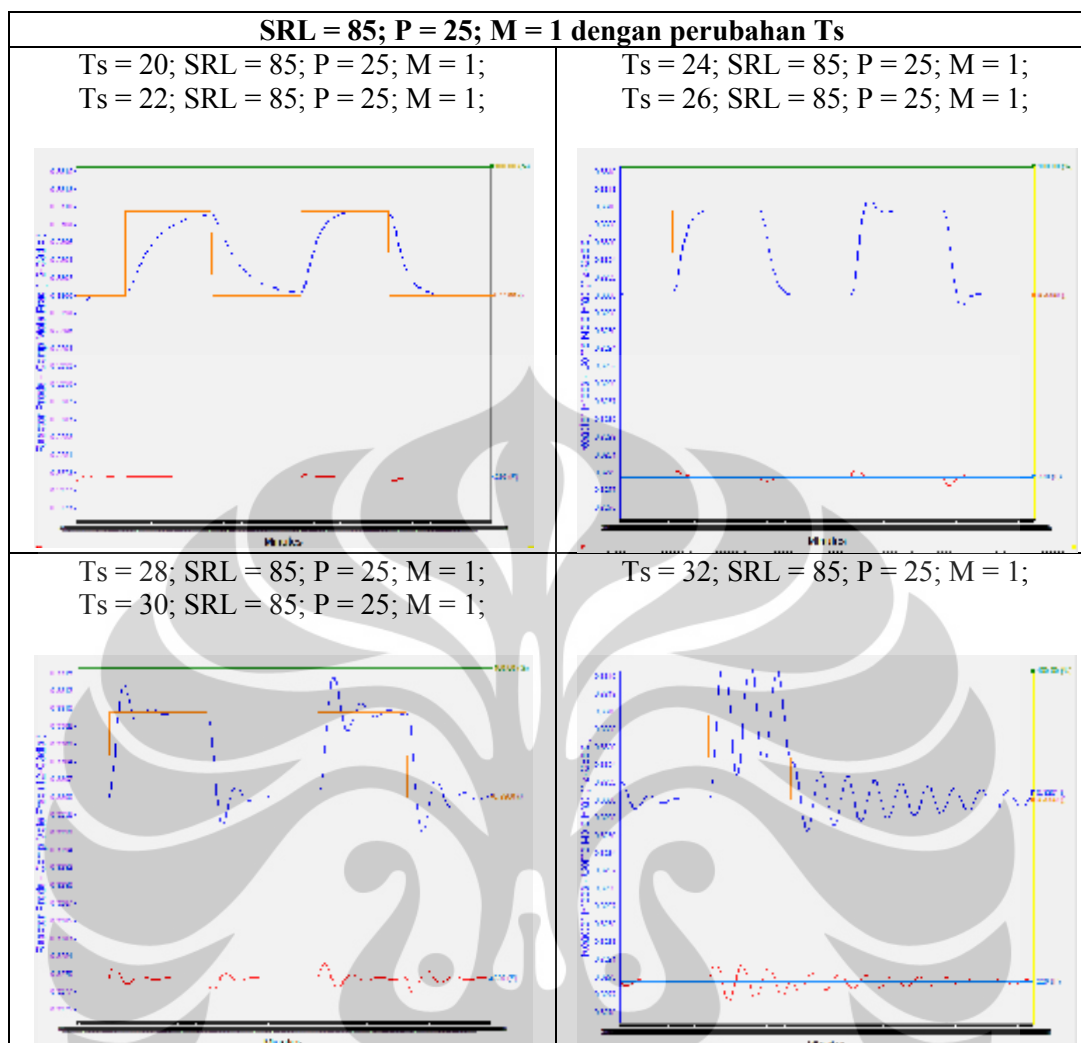
Model A



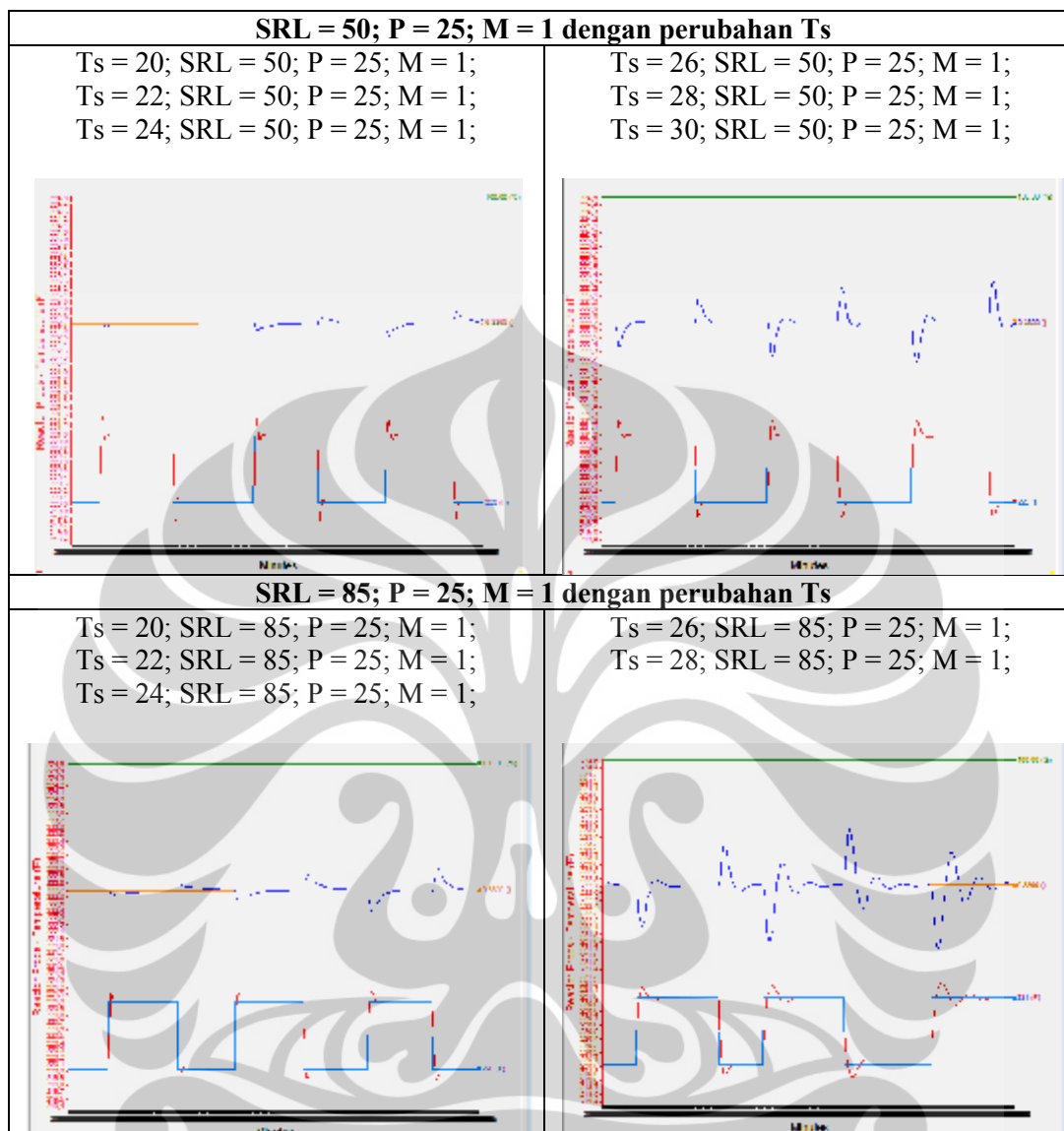


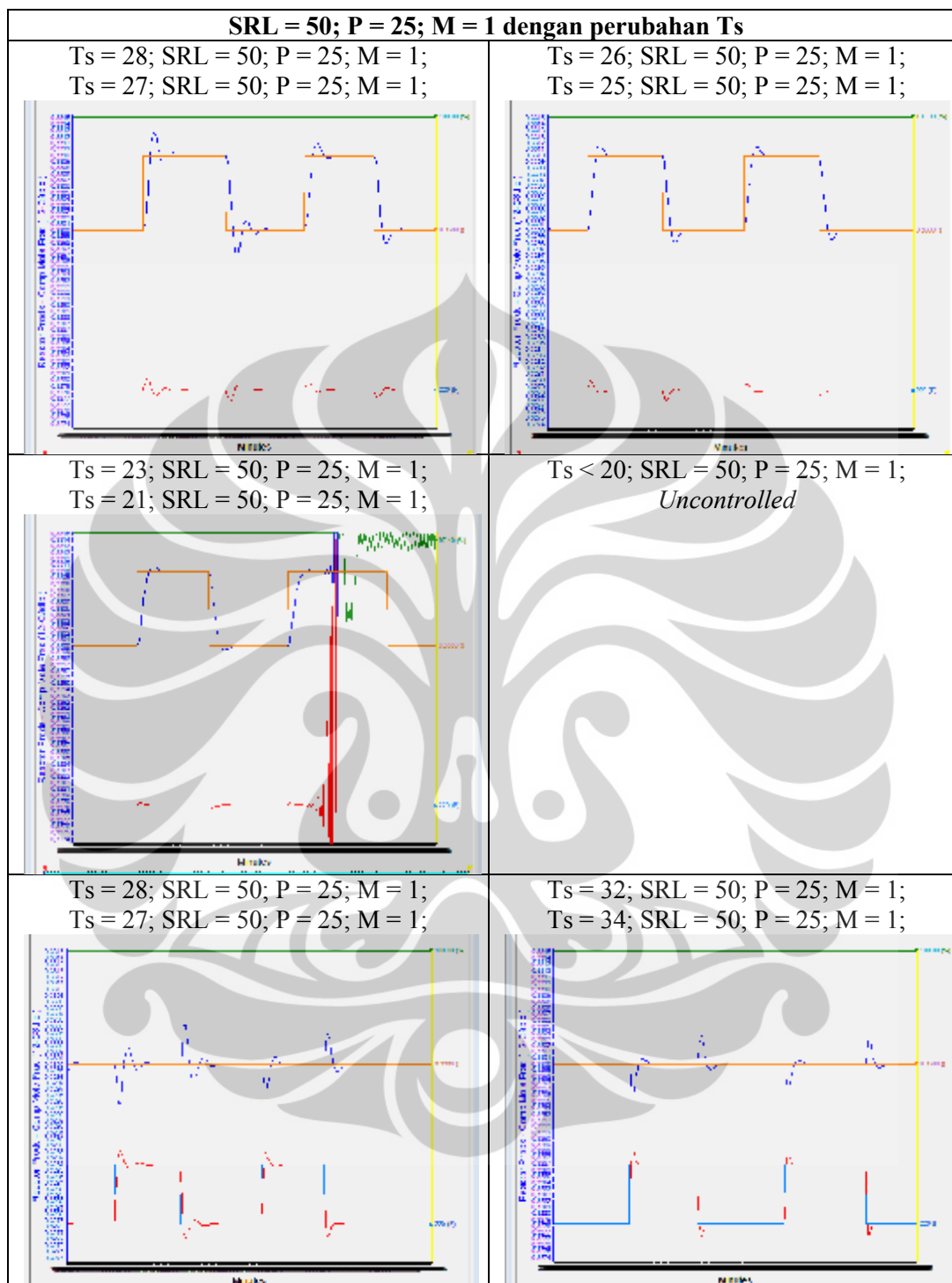


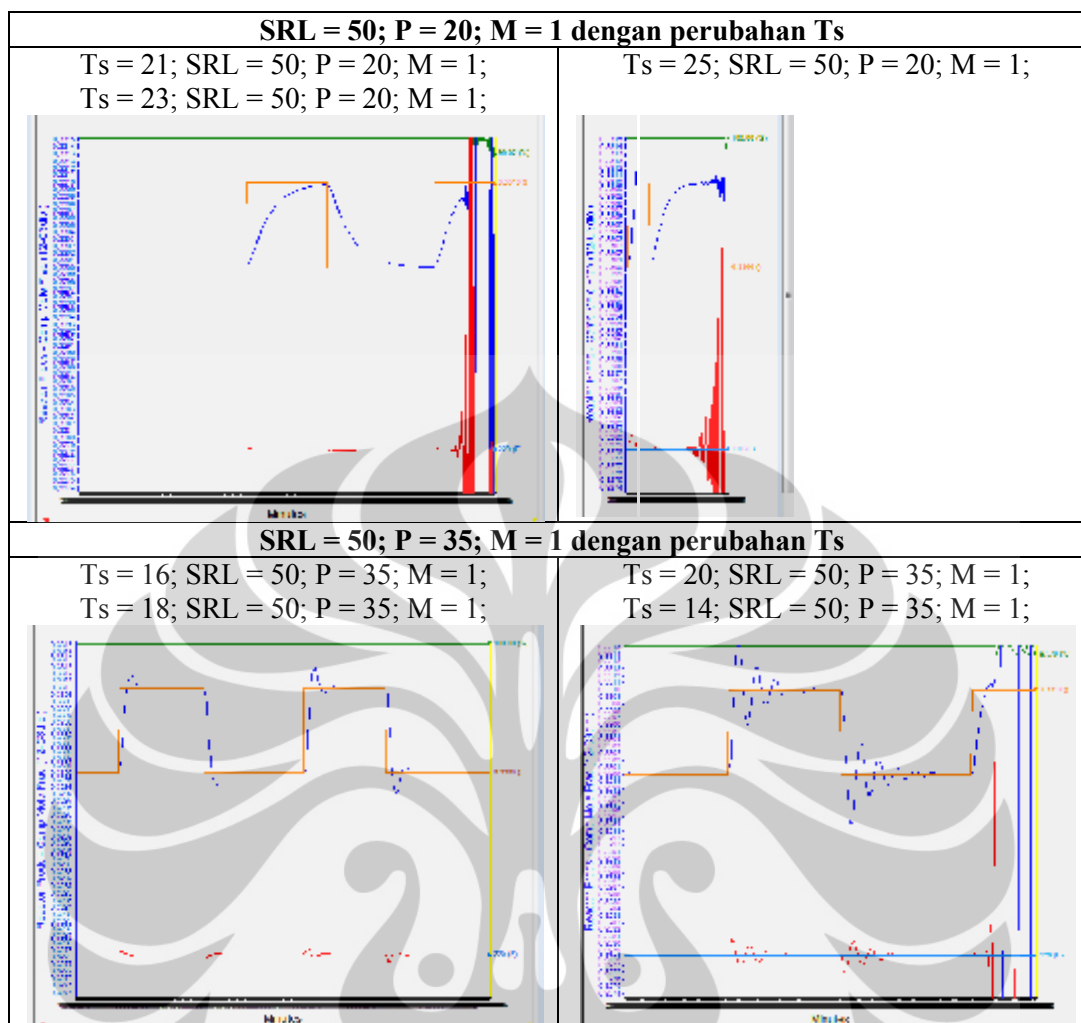




Perubahan *set point* temperatur

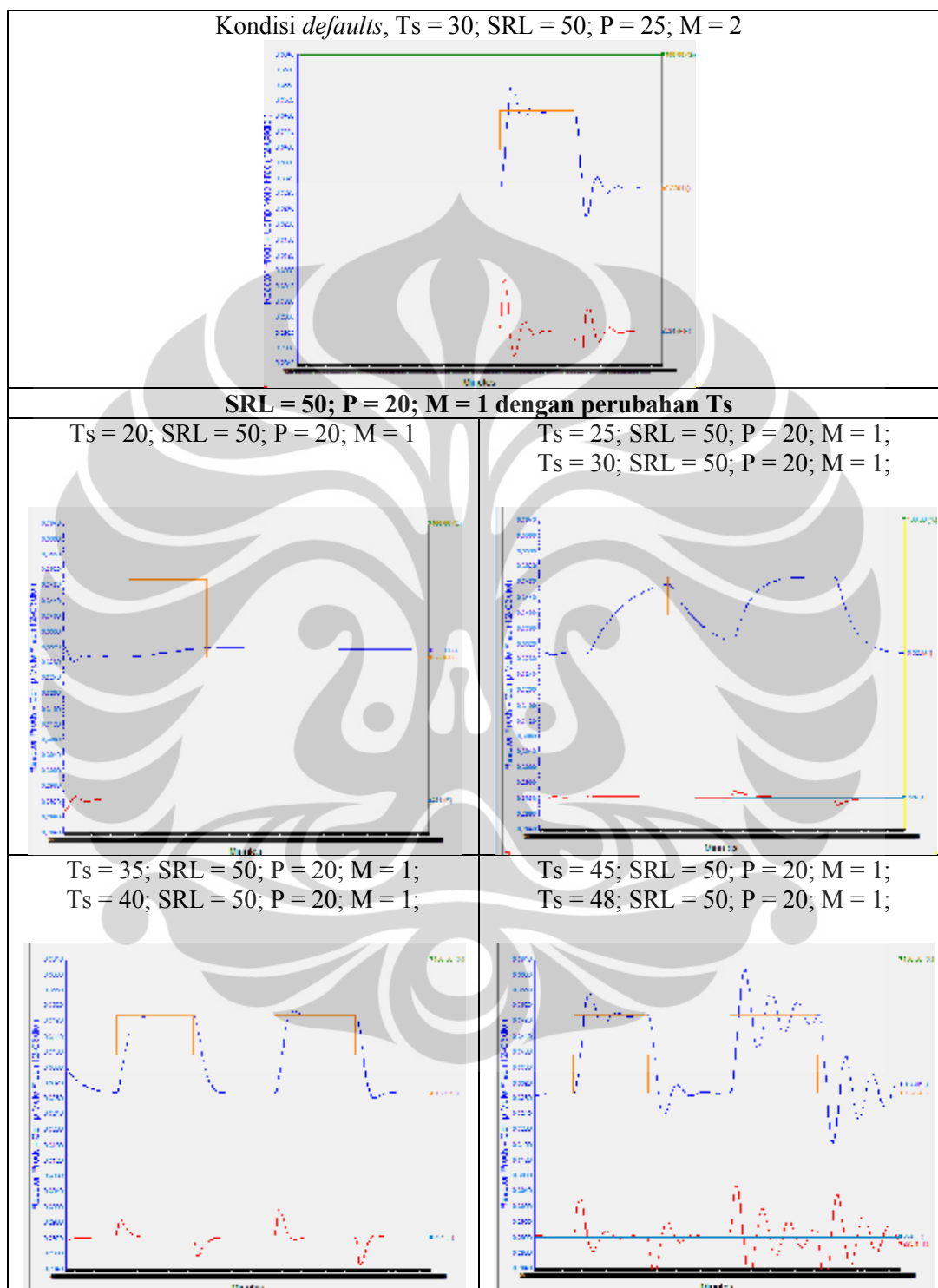


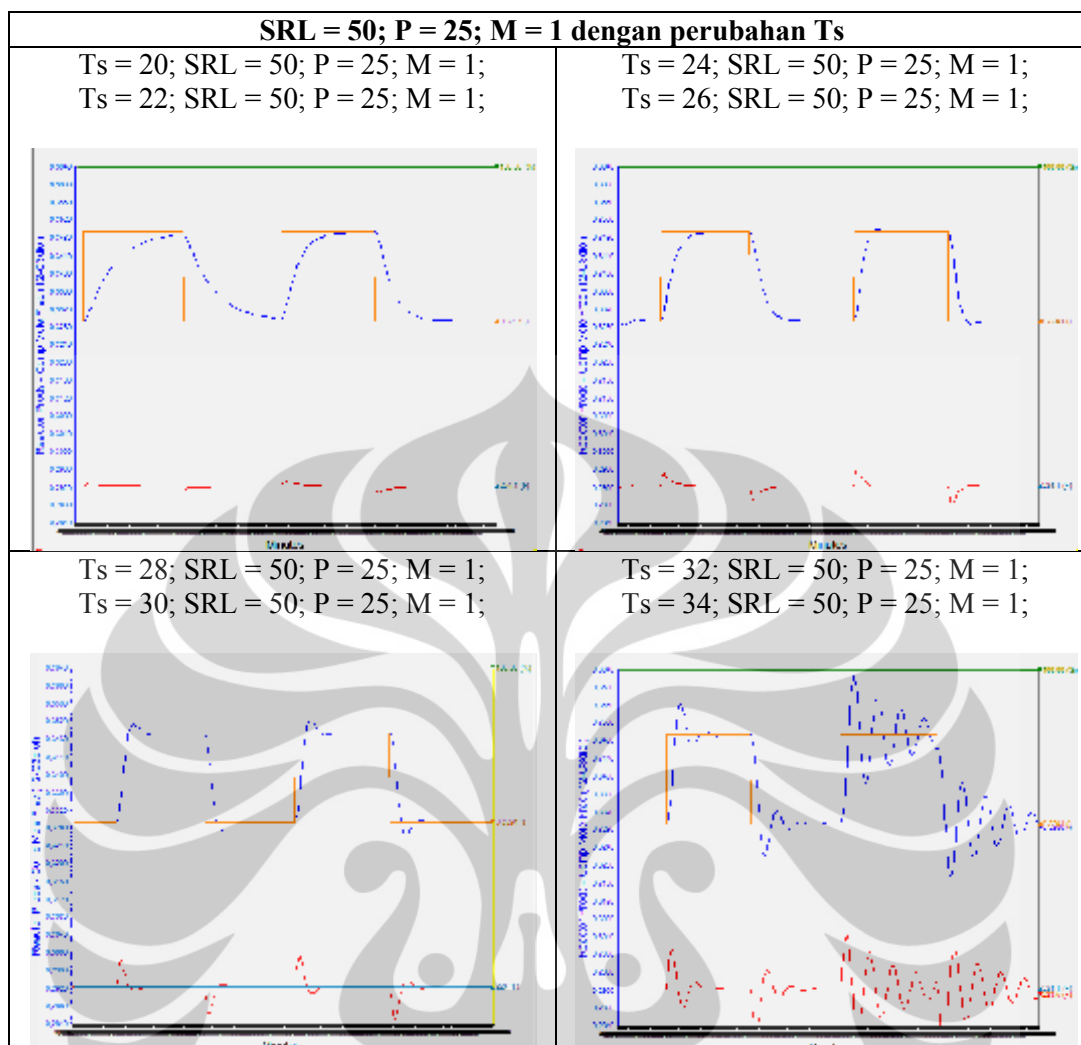
Model F

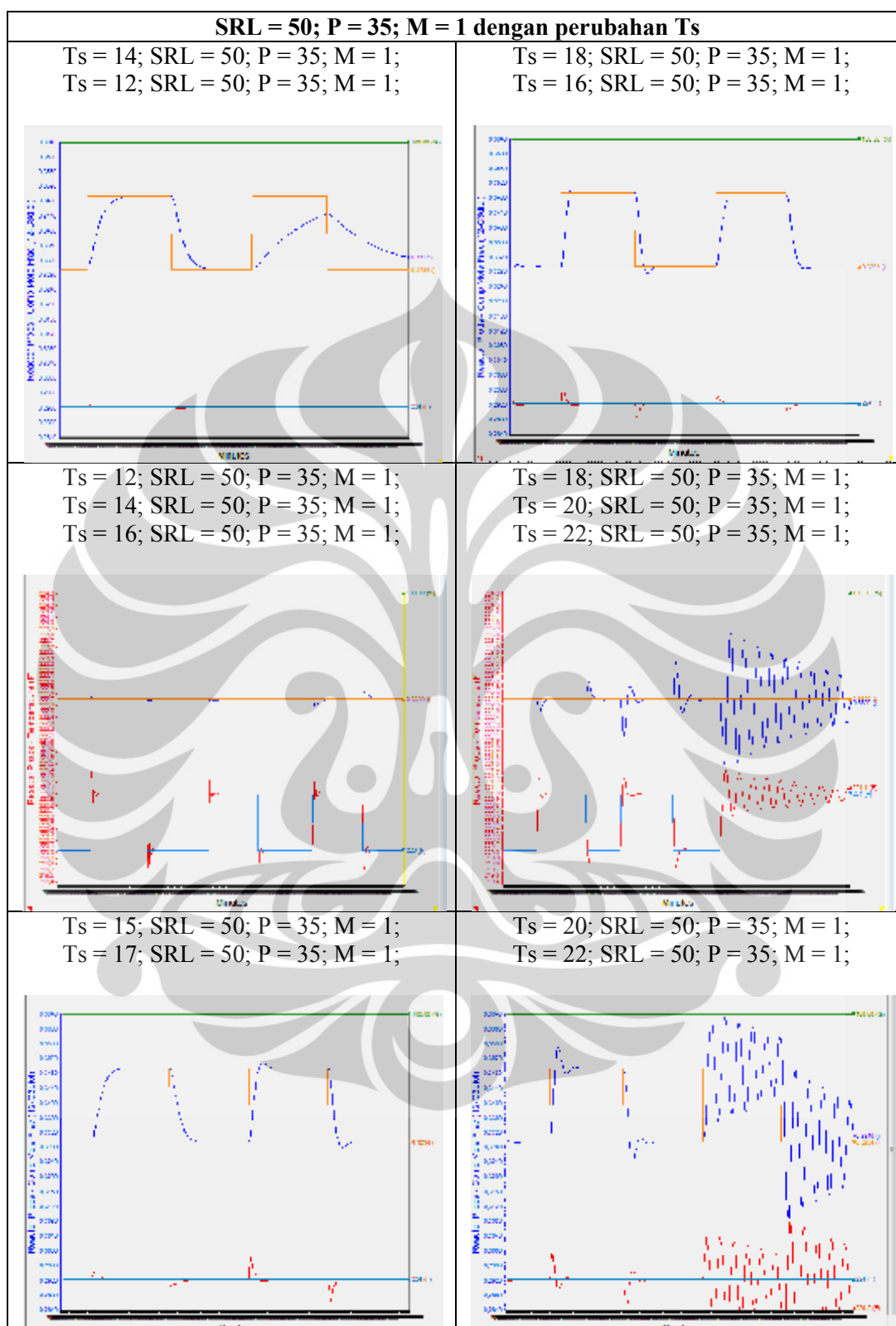


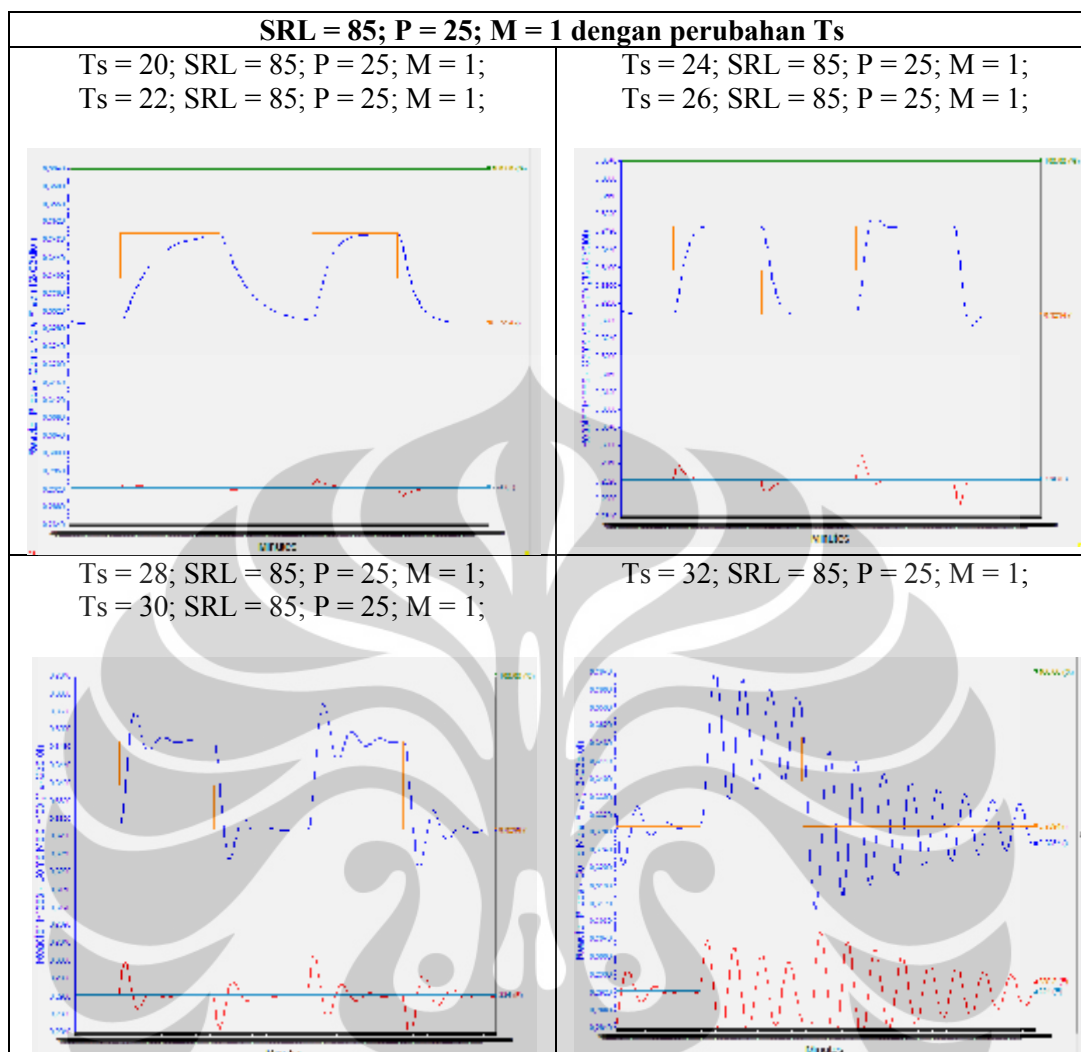
LAMPIRAN H

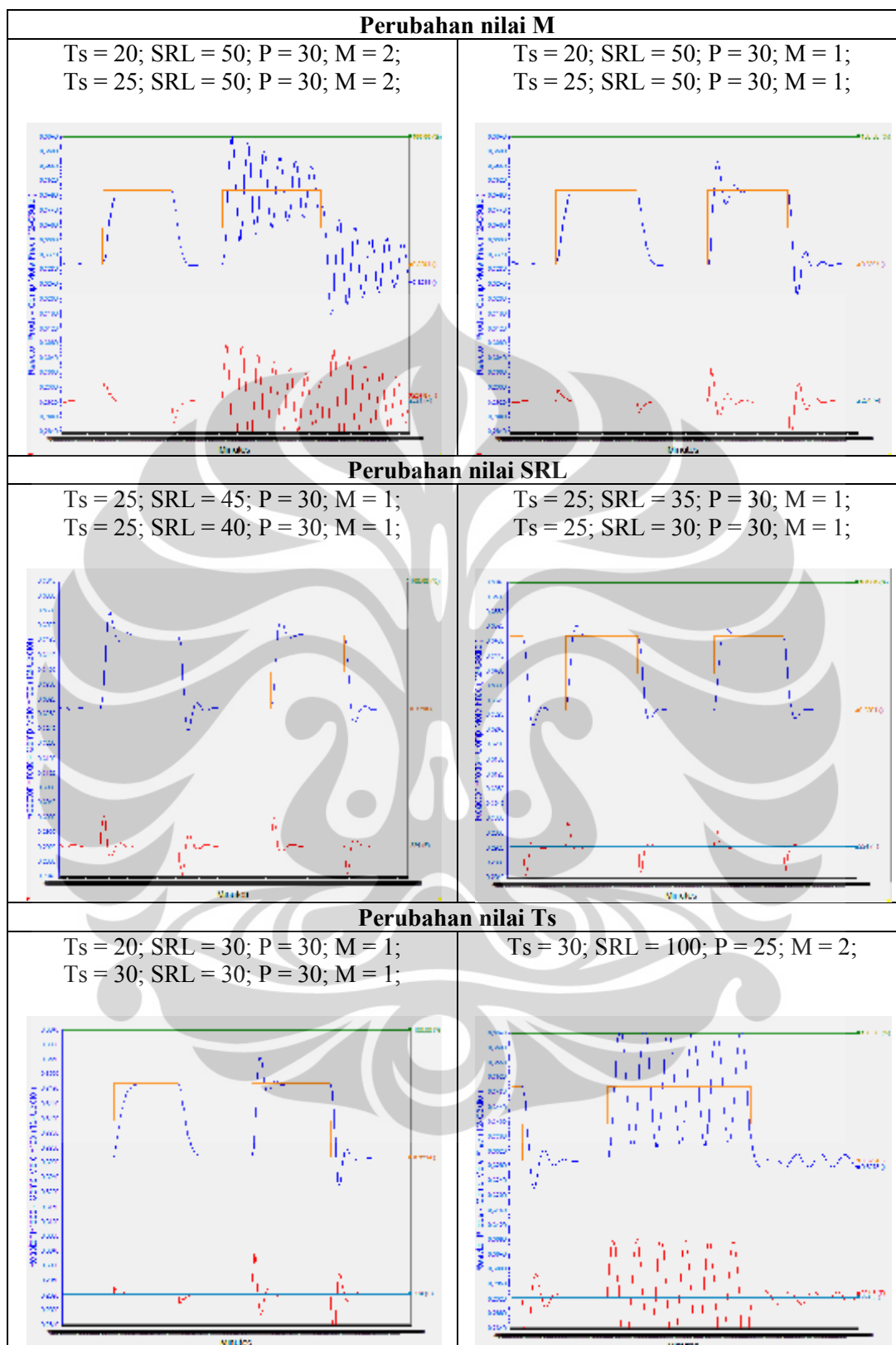
Respon Xpg Pada Tuning Parameter MPC Skenario 2 dengan Model A

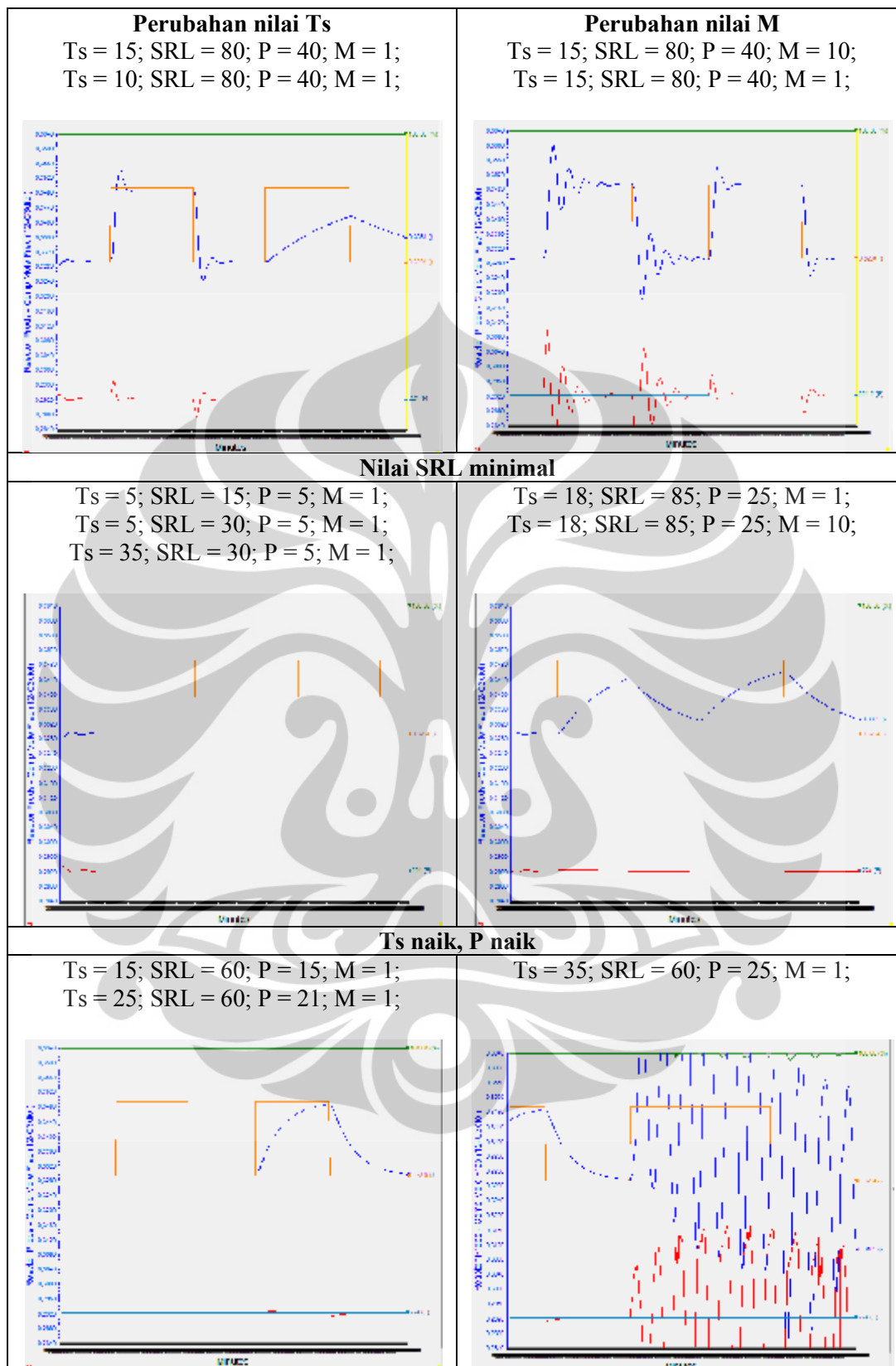


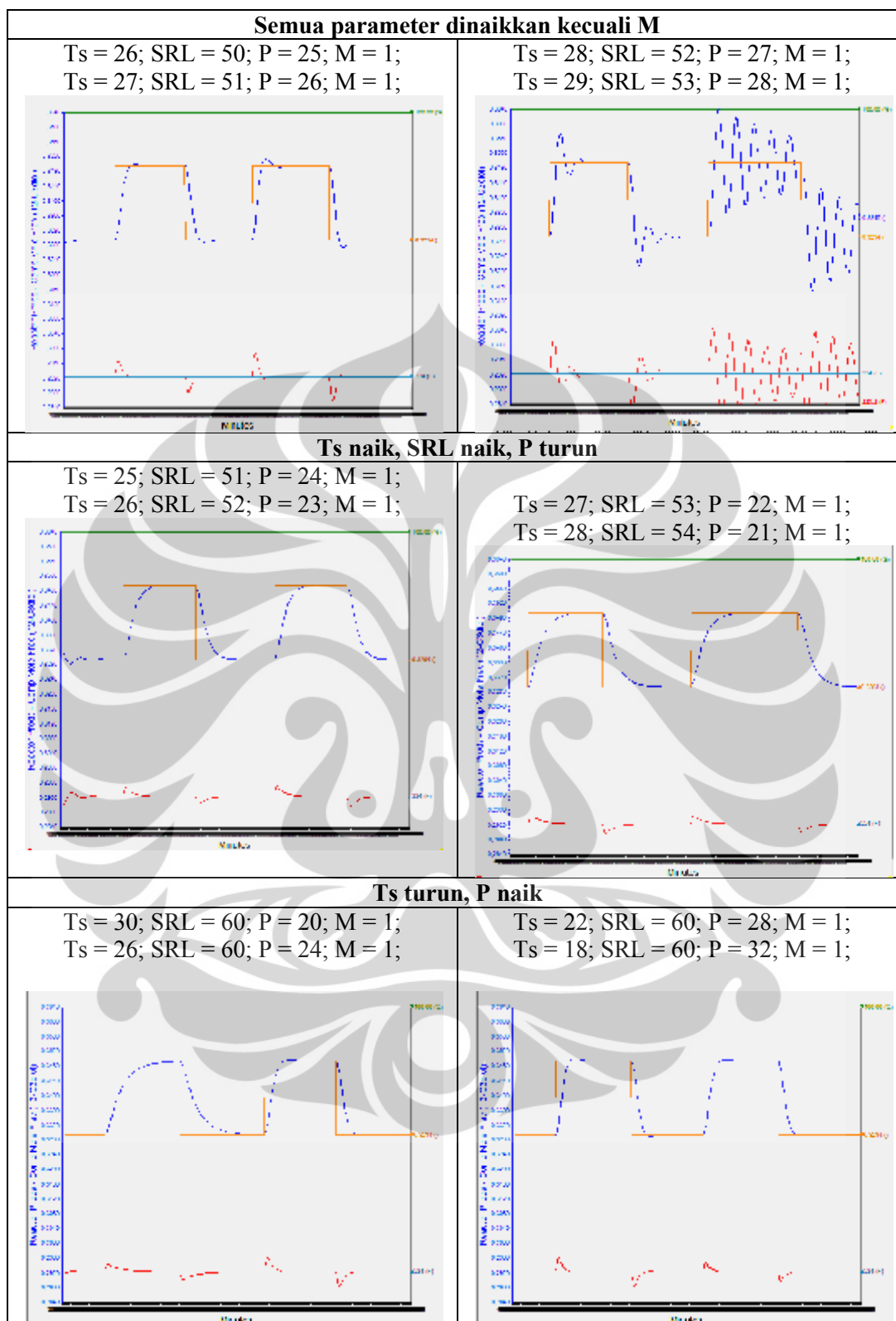












LAMPIRAN I

**Respon CV pada pengendali MPC *Multivariable* dengan
pengujian menggunakan skenario dan model yang berbeda**