



TUGAS AKHIR - TF 141581

PEMODELAN DAN OPTIMISASI *SAFETY INSTRUMENTED SYSTEM* PADA *BURNER* BERDASARKAN *DEPENDABILITY* DAN *COST MEASURE*

NADYA FERIZA
NRP. 2413 100 095

Dosen Pembimbing :
Ir. Ronny Dwi Noriyati, M.Kes
Totok Ruki Biyanto, ST, MT, Ph.D

DEPARTEMEN TEKNIK FISIKA
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2017

“Halaman ini memang dikosongkan”



FINAL PROJECT - TF 141581

**MODELLING AND OPTIMIZATION OF
BURNER SAFETY INSTRUMENTED SYSTEM
BASED ON DEPENDABILITY AND COST
MEASURE**

NADYA FERIZA
NRP. 2413 100 095

Supervisors :
Ir. Ronny Dwi Noriyati, M.Kes
Totok Ruki Biyanto, ST, MT, Ph.D

ENGINEERING PHYSICS DEPARTMENT
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2017

“Halaman ini memang dikosongkan”

PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan di bawah ini

Nama : Nadya Feriza
NRP : 2413100095
Departemen/ Prodi : Teknik Fisika/ S1 Teknik Fisika
Fakultas : Fakultas Teknologi Industri
Perguruan Tinggi : Institut Teknologi Sepuluh Nopember

Dengan ini menyatakan bahwa Tugas Akhir dengan judul “Pemodelan dan Optimisasi *Safety Instrumented System* pada *Burner* berdasarkan *Dependability* dan *Cost Measure*” adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain. Apabila di kemudian hari terbukti terdapat plagiat pada Tugas Akhir ini, maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya.

Surabaya, 11 Juni 2017
Yang membuat pernyataan,

Nadya Feriza
NRP. 2413100095

“Halaman ini memang dikosongkan”

LEMBAR PENGESAHAN
TUGAS AKHIR

PEMODELAN DAN OPTIMISASI *SAFETY*
INSTRUMENTED SYSTEM PADA *BURNER*
BERDASARKAN *DEPENDABILITY* DAN *COST*
MEASURE

Oleh:
Nadya Feriza
NRP. 2413 100 095

Surabaya, 11 Juni 2017

Menyetujui,
Dosen Pembimbing I

Ir. Ronny Dwi Noriyati, M.Kes
NIPN. 19571126 198403 2 002

Menyetujui,
Dosen Pembimbing II

Totok Ruki Biyanto, ST, MT, Ph.D
NIPN. 19710702 199802 1 001

Mengetahui,
Ketua Departemen
Teknik Fisika FTI-ITS



Agus Muhammad Hatta, S.T., M.Si., Ph.D.
NIPN. 197809022003121002

“Halaman ini memang dikosongkan”

**PEMODELAN DAN OPTIMISASI *SAFETY
INSTRUMENTED SYSTEM* PADA *BURNER*
BERDASARKAN *DEPENDABILITY* DAN *COST
MEASURE***

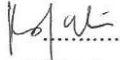



TUGAS AKHIR

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

Oleh:

NADYA FERIZA
NRP. 2413 100 095

Disetujui oleh Tim Penguji Tugas Akhir:

1. Ir. Ronny Dwi Noriyati, M.Kes (Pembimbing I) 
2. Totok Ruki Biyanto, ST, MT, Ph.D (Pembimbing II) 
3. Ir. Ya'umar, M.T (Penguji I) 
4. Hendra Cordova, S.T, M.T (Penguji II) 

**SURABAYA
JUNI, 2017**

“Halaman ini memang dikosongkan”

**PEMODELAN DAN OPTIMISASI SAFETY
INSTRUMENTED SYSTEM PADA BURNER
BERDASARKAN DEPENDABILITY DAN COST
MEASURE**

Nama : Nadya Feriza
NRP : 2413 100 095
Departemen : Teknik Fisika FTI-ITS
Dosen Pembimbing : 1. Ir. Ronny Dwi Noriyati, M.Kes
2. Totok Ruki Biyanto, ST, MT, Ph.D

Abstrak

Safety Instrumented System (SIS) merupakan salah satu tahap pada *layer of protection* yang diimplementasikan sebagai *emergency shutdown system* atau pendeteksi gas dan api. SIS berfungsi untuk mengembalikan suatu keadaan bahaya kembali ke keadaan aman. Penelitian mengenai efek redundansi pada optimisasi multi-objektif desain sistem keamanan untuk meningkatkan tingkat keamanan sangat jarang, sehingga diperlukan penelitian mengenai efek redundansi pada optimisasi desain SIS berdasarkan *reliability*, *safety* dan *cost*. Penelitian mengenai optimisasi SIS berdasarkan *dependability* dan *cost* dilakukan dengan *Genetic Algorithm*. Pemodelan dan optimisasi dilakukan dengan analisa PFD_{avg}, STR dan LCC dengan variasi *MooN voting* dan variasi teknologi pada *sensing element* dan *final element*. Nilai PFD_{avg} dipengaruhi oleh nilai laju kegagalan *danger* yang berpengaruh terhadap nilai SIL sedangkan STR dipengaruhi oleh nilai laju kegagalan *safe* yang berpengaruh terhadap kerugian produksi akibat *shutdown plant*. Tujuan tugas akhir ini adalah mendapatkan nilai LCC terendah. Solusi terbaik yang didapatkan adalah arsitektur 1oo1 dengan *smart technology transmitter* dan *air operated final element*. Dengan solusi tersebut nilai PFD_{avg} sebesar $1,16 \times 10^{-2}$, STR sebesar $1,16 \times 10^{-5}/\text{hour}$, LCC sebesar 225.638 USD, SIL 1 baik untuk SIL PFD_{avg} maupun SIL *final* dan DI 0.

Kata Kunci: *Safety Instrumented System, Burner, Dependability, Cost Measure, PFD_{avg}, STR, LCC.*

“Halaman ini memang dikosongkan.”

MODELLING AND OPTIMIZATION OF SAFETY INSTRUMENTED SYSTEM ON BURNER BASED ON DEPENDABILITY AND COST MEASURE

Name : Nadya Feriza
NRP : 2413 100 095
Department : Teknik Fisika FTI-ITS
Supervisors : 1. Ir. Ronny Dwi Noriyati, M.Kes
2. Totok Ruki Biyanto, ST, MT, Ph.D

Abstract

Safety Instrumented System (SIS) is one step in layer of protection which implemented as Emergency Shutdown System (ESD) and Fire and Gas detection system (F&G). The purposes of the system is to maintain operating condition in the safe regime when the normal operation over the allowable limits. The studies regarding the effect of diverse redundancy for multiple objective optimization of safety related systems design in safety integrity improvement were very rare. Hence, the research to describe the effect of diverse redundancy in the optimization of SIS design based on principle of reliability, safety and cost is very important. In this paper, the research in optimizing SIS based on dependability and cost done using Genetic Algorithm. Modelling and optimization were performed by using PFD_{avg} , STR and LCC with variations of Moon voting and technology for each sensing element and final element. PFD_{avg} value is affected by failure rate danger which has an impact to SIL value, meanwhile STR is affected by failure rate safe which has an impact to asset losses. The aim of this research is to obtain the lowest LCC value. The best solution chosen by optimizer is 1oo1 architecture with smart technology transmitter and air operated final element. This solution provided $1,16 \times 10^{-2}$ for PFD_{avg} value, $1,16 \times 10^{-5}$ /hour for STR value and 225.638 USD for LCC. Finally, The optimization decided the SIL PFD_{avg} and SIL final value is SIL 1 with DI 0.

Keyword: *Safety Instrumented System, Burner, Dependability, Cost Measure, PFD_{avg} , STR, LCC.*

“Halaman ini memang dikosongkan”

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang senantiasa melimpahkan rahmat serta hidayah-Nya, serta shalawat serta salam kepada Nabi Muhammad SAW, hingga terselesaikannya tugas akhir beserta laporan tugas akhir yang berjudul **PEMODELAN DAN OPTIMISASI SAFETY INSTRUMENTED SYSTEM PADA BURNER BERDASARKAN DEPENDABILITY DAN COST MEASURE.**

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

1. Bapak Agus Muhamad Hatta, S.T., M.Si., Ph.D selaku Ketua Departemen Teknik Fisika yang telah memberikan petunjuk, ilmu, serta bimbingan selama menempuh pendidikan di Teknik Fisika.
2. Kedua orang tua (Bapak Faiq Bahfen dan Ibu Tuty Muliaty) kakak (Muhammad Fariz) serta kakak ipar (Licha Permata Amanda). Terima kasih atas segala cinta, kasih sayang, doa, perhatian, serta dukungan moral dan materiil yang telah diberikan.
3. Diken Yus Damara selaku teman yang telah menemani dan memberikan dukungan moral dan semangat serta bimbingan selama 6 tahun ini.
4. Ibu Ir. Ronny Dwi Noriyati, M.Kes dan Bapak Totok Ruki Biyanto, ST, MT, Ph.D selaku dosen pembimbing yang telah dengan sabar memberikan petunjuk, ilmu, serta bimbingan yang sangat bermanfaat.
5. Bapak Totok Ruki Biyanto, S.T., M.T., Ph.D.selaku Kepala Laboratorium Rekayasa Instrumensi yang telah memberikan ilmu, petunjuk, nasihat, serta kemudahan perizinan.
6. Bapak Dr. Bambang Lelono Widjiantoro, ST. MT selaku dosen wali yang telah membimbing penulis selama perkuliahan.

7. Teman – teman HOD (Garin, Nevy, Abel, Pijar, Rara dan Chitra) dan Oriehanna serta Ofi yang telah memberikan semangat dan dukungan selama 4 tahun di Surabaya.
8. Fika, Tyo, Hafisyah, Nugroho, Mirza, dan Adhibya yang telah memberikan semangat dan berjuang bersama selama masa perkuliahan di Teknik Fisika.
9. *Partner* Tugas Akhir (Gita Fitri Sasanti) dan seluruh teman – teman Laboratorium Instrumentasi yang telah banyak membantu selama pengerjaan tugas akhir.
10. Seluruh teman – teman Departemen Teknik Fisika angkatan 2013, terima kasih untuk semuanya.
11. Seluruh dosen, karyawan dan civitas akademik Teknik Fisika, terimakasih atas segala bantuan dan kerjasamanya.
12. Semua pihak yang tidak dapat disebutkan satu persatu, terimakasih atas bantuannya.

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

Surabaya, 2 Juni 2017

Penulis

DAFTAR ISI

Halaman Judul	i
Title Page.....	ii
Lembar Pengesahan I	vii
Lembar Pengesahan II	x
KATA PENGANTAR.....	xvi
DAFTAR ISI.....	xix
DAFTAR GAMBAR.....	xxi
DAFTAR TABEL	xxiii
DAFTAR NOTASI	xxv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan	4
1.4 Lingkup Kerja	4
1.5 Sistematika Laporan.....	5
BAB II TINJAUAN PUSTAKA	6
2.1 <i>Burner</i>	7
2.2 <i>Safety Instrumented System</i>	9
2.3 <i>Dependability</i>	11
2.4 <i>Lifecycle Cost</i>	13
2.5 <i>Diversity</i>	16
2.6 <i>Genetic Algorithm</i>	17
BAB III METODOLOGI PENELITIAN	21
3.1 Pengambilan Data	22
3.2 Penentuan SIS yang Akan Didesain.....	26
3.3 Penentuan Teknik Optimisasi.....	26
3.4 Perhitungan PFD_{avg} dan STR	26
3.5 Perhitungan LCC.....	27
3.6 Optimisasi SIS.....	27
3.7 Analisa Hasil Optimisasi.....	28
BAB IV ANALISIS HASIL DAN PEMBAHASAN.....	31
4.1 Perancangan <i>Safety Instrumented System</i> pada Burner.....	31

4.2	Optimisasi <i>Dependability</i> dan <i>Cost</i>	32
4.3	Analisa Hasil Optimisasi.....	39
BAB V PENUTUP.....		45
5.1	Kesimpulan	45
5.2	Saran.....	45
DAFTAR PUSTAKA.....		47
BIODATA PENULIS.....		50

DAFTAR GAMBAR

Gambar 2.1	Diagram alir GA.....	18
Gambar 3.1	Diagram Alir Tugas Akhir	21
Gambar 3.2	<i>Flowchart Program</i> MATLAB R2016a.....	27
Gambar 4.1	Pemodelan SIS pada <i>Burner</i>	31
Gambar 4.2	Hasil Optimisasi PFD_{avg}	35
Gambar 4.3	Hasil Optimisasi STR.....	35
Gambar 4.4	Optimisasi LCC.....	36
Gambar 4.5	Hasil Optimisasi PFD_{avg} saat LCC Minimum untuk Masing-Masing Iterasi.....	37
Gambar 4.6	Hasil Optimisasi STR saat LCC Minimum untuk Masing-Masing Iterasi.....	37
Gambar 4.7	Hasil Optimisasi LCC Terbaik untuk Masing-Masing Iterasi.....	38

“Halaman ini memang dikosongkan”

DAFTAR TABEL

Tabel 2.1 Korelasi SIL dan PFD_{avg} untuk <i>Low Demand Mode</i>	12
Tabel 2.2 SIL <i>Architectural Constraint</i>	13
Tabel 3.1 Data <i>Dependability Flow Transmitter</i>	23
Tabel 3.2 Data <i>Dependability Final Element</i>	23
Tabel 3.3 Data <i>Dependability Logic Solver</i>	24
Tabel 3.4 Data <i>Lifecycle Cost Instrument</i> [1].....	24
Tabel 3.5 Data <i>Lifecycle Cost PLC</i> [1].....	24
Tabel 3.6 Data Lain [1]	25
Tabel 3.7 Nilai Fraksi Kegagalan <i>Vote Arsitektur</i> [1].....	25
Tabel 4.1 Hasil Optimisasi Maksimum dan Minimum	33
Tabel 4.2 Hasil Variasi Redundansi dan Teknologi	33
Tabel 4.3 Hasil Optimisasi	34
Tabel 4.4 Hasil Variasi Redundansi dan Teknologi	34

“Halaman ini memang dikosongkan”

DAFTAR NOTASI

C_{ACC}	=	<i>Catastrophic loss</i>
C_{CM}	=	<i>Corrective maintenance cost</i>
C_{design}	=	<i>Design cost</i>
C_{hazard}	=	<i>Cost of hazard</i>
$C_{inst/comm}$	=	<i>Installation cost</i>
C_{Moon}	=	<i>Fraksi kegagalan MoonN architecture</i>
$Components_i$	=	<i>Komponen ke-i</i>
C_{op}	=	<i>Operational cost</i>
C_{PM}	=	<i>Predictive maintenance cost</i>
C_{pro}	=	<i>Procurement cost</i>
$C_{purchase}$	=	<i>Purchase cost</i>
C_{risk}	=	<i>Risk cost</i>
C_{SD}	=	<i>Shut down cost</i>
C_{STR}	=	<i>Spurious Trip Rate cost</i>
C_T	=	<i>Test cost</i>
DI	=	<i>Diversity Index</i>
$F(ACC PFD_{avg})$	=	<i>Plant risk without SIS</i>
HFT	=	<i>Hardware Fault Tolerance</i>
N	=	<i>Jumlah MoonN voting</i>
LCC	=	<i>Life Cycle Cost</i>
N_s	=	<i>Total komponen yang digunakan</i>
PFD	=	<i>Probability Failure on Demand</i>
PFD_{avg}	=	<i>Average Probability Failure on Demand</i>
PVF	=	<i>Factor by Present Value</i>
R	=	<i>Discount rate</i>
$Redudancy_i$	=	<i>Redundansi ke-i</i>
SD_{loss}	=	<i>Shut down loss</i>
SD_{time}	=	<i>Shut down time</i>
SFF	=	<i>Safe Failure Fraction</i>
SIF	=	<i>Safety Instrumented Function</i>
SIL	=	<i>Safety Integrity Level</i>
SIS	=	<i>Safety Instrumented Level</i>
STR	=	<i>Spurious Trip Rate</i>

T	=	<i>Years life</i>
$Technologies_i$	=	Teknologi ke-i
TI	=	turunan gaya arah <i>sway</i> terhadap \dot{v} ($Ndet^2/m$)
β_{DD}	=	Beta faktor <i>dangerous detected failures</i>
β_{DU}	=	Beta faktor <i>dangerous undetected failures</i>
β_{SD}	=	Beta faktor <i>safe detected failures</i>
β_{SU}	=	Beta faktor <i>safe undetected failures</i>
λ_{DD}	=	Laju kegagalan <i>dangerous detected failures</i>
λ_{DU}	=	Laju kegagalan <i>dangerous undetected failures</i>
λ_{SD}	=	Laju kegagalan <i>safe detected failures</i>
λ_{SU}	=	Laju kegagalan <i>safe undetected failures</i>
λ_T	=	Total <i>failure rate</i>

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam sistem pengamanan suatu plant, terdapat beberapa lapisan pengamanan, salah satunya adalah *Safety Instrumented System* (SIS). SIS akan bekerja apabila lapisan pengamanan sebelumnya, yaitu *Basic Process Control System* (BPCS) dan *alarm* tidak dapat menangani pelanggaran operasional yang terjadi. SIS diimplementasikan sebagai *Emergency Shutdown System* (ESD) atau pendeteksi gas dan api (F&G) [1]. Ketika BPCS sudah tidak dapat menangani pelanggaran operasional yang terjadi, maka SIS akan segera mematikan seluruh sistem agar menghentikan produksi pada tempat terjadinya kegagalan. SIS tersusun dari *Safety Instrumented Function* (SIF) yang terdiri dari *sensing element*, *logic solver*, dan *final element*. SIS bertujuan untuk mengembalikan sistem ke keadaan aman ketika terjadi suatu keadaan yang menyebabkan perubahan dari nilai *set point* yang telah ditentukan [2].

Terdapat beberapa standar internasional yang digunakan sebagai acuan dalam mendesain sistem keamanan, salah satunya adalah IEC 61508. IEC 61508 merupakan standar internasional yang dipublikasikan oleh *International Electrotechnical Commission* yang membahas fungsi proteksi dari sistem-sistem yang terkait dengan *Electrical/Electronic/Programmable Electronic Safety* (E/E/PE atau E/E/PES) [2]. Dalam memenuhi standar keselamatan IEC 61508, pengaplikasian SIS pada proses industri mulai digunakan.

Distribusi redundansi dan keandalan komponen telah terbukti sebagai metode yang baik dalam memenuhi syarat keamanan berdasarkan IEC 61508 [3]. Penelitian mengenai optimisasi *reliability* telah dibahas sejak 10 tahun yang lalu. Objek utama penelitian tersebut adalah distribusi keandalan dari suatu komponen. Pada penelitian tersebut, untuk meningkatkan keandalan sistem maka dilakukan peningkatan keandalan dari komponen yang digunakan atau dengan menambahkan

redundansi [4]. Selain itu, terdapat penelitian yang membahas penggunaan *Genetic Algorithm* (GA) untuk mendapatkan solusi dari masalah distribusi redundansi untuk sistem *series-parallel* [5]. Penelitian yang telah dilakukan banyak membahas mengenai *reliability*, *availability* dan *cost* dengan batasan *cost* dan *weight* [6, 7, 8]. Selain itu, optimisasi desain sistem keamanan dengan *single objective Genetic Algorithm* juga telah dilakukan [9]. Dari penelitian – penelitian yang telah dilakukan, belum terdapat penelitian yang membahas dampak dari *diverse redundancy* terhadap *multi objectives optimization* pada sistem keamanan untuk meningkatkan tingkat keamanan. Tujuan dari Tugas Akhir ini adalah untuk mengetahui dampak dari *diverse redundancy* pada optimisasi desain SIS berdasarkan *reliability*, *safety* dan *cost*.

Pada Tugas Akhir ini pemodelan dan optimisasi SIS dilakukan pada *burner*. *Burner* merupakan komponen yang digunakan untuk memasok panas ke unit perpindahan panas. *Burner* biasanya digunakan untuk membakar bahan bakar dengan bantuan oksidator, energi kimia yang terkandung dalam bahan bakar akan dikonversikan menjadi energi panas pada prosesnya. Dengan kata lain dapat dikatakan bahwa *burner* merupakan sumber panas pada *boiler*.

Terdapat beberapa keadaan pada *burner* yang dapat menyebabkan bahaya dan kerugian bagi lingkungan maupun perusahaan. Kebocoran yang terjadi pada jalur *feedwater* menuju *burner* dapat menyebabkan temperatur pemanasan yang tidak sesuai dengan *set point* yang diinginkan dan akan berakibat fatal jika tidak segera ditangani. Selain itu, *valve* yang tidak dapat membuka sesuai dengan *set point* dapat menyebabkan temperatur pemanasan yang tidak stabil dan dapat mengakibatkan kebakaran serta ledakan [10]. Oleh karena itu perlu dilakukan pemodelan dan optimisasi SIS pada *burner* untuk meningkatkan keamanan dan meminimalisir peluang terjadinya pelanggaran operasional.

Pada penelitian ini akan dilakukan pemodelan dan optimisasi SIS pada *burner* dengan tinjauan *dependability* dan *cost measure*. *Dependability* merupakan properti sistem yang mengintegrasikan

reliability, availability, safety, security, survivability, maintainability [11]. Namun dalam Tugas Akhir ini, *dependability* hanya diwakilkan dengan perhitungan *reliability* dan *safety*. *Dependability* dapat dikatakan sebagai tingkat kepercayaan pengguna terhadap suatu komponen. Apabila suatu komponen tidak dapat dipercaya oleh pengguna, maka komponen tersebut dapat ditolak oleh penggunanya. Pengguna dapat menolak suatu komponen yang tidak *dependable*, karena resiko yang ditimbulkan dapat menyebabkan kegagalan fatal dan kerugian bagi perusahaan. Oleh karena itu dilakukan penelitian mengenai pemodelan dan optimisasi SIS pada burner berdasarkan *dependability* dengan tujuan mendapatkan sistem keamanan pada burner yang dapat dipercaya oleh pengguna.

Tidak hanya ditinjau dari *dependability*, pada penelitian ini pun akan ditinjau optimisasi SIS pada *burner* dengan mempertimbangkan *cost* yang akan diwakilkan melalui perhitungan *Lifecycle Cost* (LCC). Tiga objek utama yang diperhatikan dalam Tugas Akhir ini adalah *Average of Probability on Demand* (PFD_{avg}), *Spurious Trip Rate* (STR), dan LCC yang merupakan penambahan dari *cost* akibat PFD_{avg} dan STR. Metode optimisasi yang digunakan adalah *Genetic Algorithm* (GA). GA merupakan salah satu metode multi objektif yang digunakan untuk memecahkan masalah berbasis proses seleksi alam dengan meniru evolusi biologi [12]. Secara garis besar, cara kerja GA adalah dengan mengambil secara acak individu di dalam suatu populasi dengan menyocokkan dengan hasil yang diinginkan. Kemudian dilakukan *crossover* atau reproduksi terhadap individu tersebut sehingga menghasilkan keturunan sesuai dengan yang dikehendaki [1, 12]. Metode GA merupakan metode yang dapat memecahkan permasalahan yang rumit, dimensi yang tinggi, diskrit, non-linear, dan diskontinyu [1]. Dengan menggunakan metode GA memungkinkan untuk mendapatkan keseimbangan antara *dependability* dan *cost* [13]. Selain itu, metode GA juga memungkinkan bagi SIS untuk mengeksekusi lebih dari satu SIF.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, rumusan masalah dari penulisan tugas akhir ini adalah:

1. Bagaimana merancang *Safety Instrumented System* pada burner dengan mempertimbangkan *dependability* dan *cost measure* berdasarkan dengan IEC 61508?
2. Bagaimana menggabungkan *safety*, *reliability*, dan *Lifecycle Cost* dalam suatu desain optimisasi?
3. Bagaimana mengintegrasikan *safety* dan *reliability* dengan *Lifecycle Cost* untuk optimisasi dengan menggunakan *Genetic Algorithm*?

1.3 Tujuan

Berdasarkan pemaparan latar belakang dan permasalahan maka tujuan dari penelitian tugas akhir ini adalah sebagai berikut:

1. Memodelkan *Safety Instrumented System* pada burner dengan mempertimbangkan *dependability* dan *cost measure* berdasarkan dengan standar yang ada.
2. Menggabungkan *safety*, *reliability*, dan *Lifecycle Cost* dalam suatu desain optimisasi.
3. Mengintegrasikan *safety* dan *reliability* dengan *Lifecycle Cost* untuk optimisasi dengan menggunakan *Genetic Algorithm*.

1.4 Lingkup Kerja

Adapun lingkup kerja dari Tugas Akhir ini adalah dilakukan pemodelan dan optimisasi SIS berdasarkan *dependability* dan *cost measure* pada burner. *Dependability* diwakilkan dengan perhitungan *reliability* dan *safety* yang ditunjukkan melalui nilai PFD_{avg} dan STR berdasarkan IEC 61508. Objek optimisasi yaitu mendapatkan nilai terendah dari LCC dengan variable yang dioptimisasi adalah *vote* arsitektur dan teknologi untuk *sensing element* dan *final element*. *Vote* arsitektur yang digunakan adalah 1001 hingga 6006. Teknologi yang digunakan untuk *sensing element* adalah *smart transmitter*, *conventional electric transmitter* dan *switch* sedangkan teknologi yang digunakan

untuk *final element* adalah *air operated*, *hydraulic operated* dan *motor operated*. Setiap teknologi memiliki nilai λ_{dd} , λ_{du} , λ_{sd} , λ_{su} , β_{dd} , β_{du} , β_{sd} dan β_{su} . Nilai laju kegagalan adalah konstan dan kegagalan yang terjadi pada masing – masing subsistem tidak mempengaruhi subsistem lainnya (*independent*). SIS diasumsikan untuk keadaan *low demand* dan *time interval* diasumsikan sebesar 1 tahun. Penggunaan data *dependability* dan *lifecycle cost* dapat digunakan pada semua *plant* karena tidak dipengaruhi oleh jenis proses atau tipe peralatan. Teknik optimisasi yang digunakan adalah *Genetic Algorithm* yang dijalankan pada *software* MATLAB R2016a.

1.5 Sistematika Laporan

Sistematika penulisan laporan tugas akhir adalah sebagai berikut:

1. **BAB I PENDAHULUAN**
Bab ini berisi pendahuluan yang menjelaskan latar belakang, rumusan masalah, lingkup kerja, tujuan dan sistematika penulisan.
2. **BAB II TEORI PENUNJANG**
Definisi dan penjelasan pustaka yang dijadikan referensi beserta penelitian sebelumnya yang terkait dalam pembuatan tugas akhir ini akan dijelaskan pada bab dua. Teori yang dipaparkan diantaranya mengenai *Burner*, *Safety Instrumented System*, *Dependability*, *Lifecycle Cost*, *Diversity*, *Genetic Algorithm*, serta konsep-konsep lain yang berkaitan dengan pembuatan tugas akhir.
3. **BAB III METODOLOGI PENELITIAN**
Bab ini menggambarkan uraian dan urutan pekerjaan yang akan dilakukan dalam penyusunan tugas akhir ini.
4. **BAB IV ANALISIS DATA DAN PEMBAHASAN**
Bab ini berisi data, hasil pengerjaan Tugas Akhir dan analisis dari pengerjaan Tugas Akhir yang telah dilakukan.
5. **BAB V KESIMPULAN DAN SARAN**
Bab ini berisi tentang simpulan dari keseluruhan Tugas Akhir dan saran maupun rekomendasi terhadap penelitian

Tugas Akhir ini untuk perbaikan ataupun penelitian lanjutan yang memiliki kesamaan dengan topik yang diangkat.

“Halaman ini memang dikosongkan”

BAB II

TINJAUAN PUSTAKA

2.1 Burner

Fired heaters sudah banyak digunakan di segala sektor industri, seperti perminyakan, petrokimia, pembangkit listrik, semen, dll. *Fired heaters* digunakan untuk memproses bahan mentah menjadi suatu produk. Bahan bakar minyak dan gas sering kali digunakan sebagai media untuk memanaskan produk. Sistem kontrol *burner* merupakan sistem yang sangat penting, sistem ini berfungsi untuk menghindari pemanasan api dan ledakan yang tidak terkontrol akibat kerusakan dan tidak berfungsinya *heat transfer tubes*. *Burner* dikontrol melalui suatu perangkat lunak yang terdapat pada *microprocessor* komputer. Dengan kata lain, *burner* merupakan komponen yang digunakan untuk memasok panas ke unit perpindahan panas. *Burner* biasanya digunakan untuk membakar bahan bakar dengan bantuan oksidator, energi kimia yang terkandung dalam bahan bakar akan dikonversikan menjadi energi panas pada prosesnya.

Pembakaran memerlukan tiga fungsi penting dari desain, yaitu *burner* harus melakukan pencampuran bahan bakar dan oksigen, pemantikan api yang stabil dan mengontrol keadaan api. Untuk mencapai ketiga fungsi tersebut, *burner* harus memenuhi kondisi operasi tertentu. Kondisi operasi tersebut ditentukan saat proses desain, seperti tipe bahan bakar, komposisi bahan bakar, tingkat panas yang dilepaskan karena *mass flow rate*, tekanan bahan bakar maksimum yang diperlukan, tekanan medium atomisasi maksimum untuk pembakaran fluida, suhu bahan bakar, sumber oksidan seperti udara ambien, gas buang, tekanan udara pembakaran yang tersedia baik karena suhu pembakaran (positif) maupun diinduksi (negative), suhu pembakaran (ambien atau pemanasan awal), suhu tungku pembakaran, dimensi *furnace* untuk batasan ukuran api dan tipe dari api (konfigurasi atau bentuknya).

Beberapa kondisi operasi tertentu tidak dapat diaplikasikan pada keadaan sesungguhnya, seperti pada *mass flow rate* dan

tekanan bahan bakar akibat terdapatnya gangguan. Untuk mempertahankan kondisi dari *mass flow rate* dan tekanan bahan bakar, BPCS digunakan untuk mengontrol proses variabel secara otomatis. BPCS dilengkapi dengan sistem *alarm* termasuk *Standard Operating Procedures (SOP)* dan operator yang terlatih. Beberapa keadaan lain pada *burner* yang dapat menyebabkan bahaya terhadap lingkungan sekitarnya dan dapat menyebabkan kerugian bagi perusahaan adalah kebocoran yang terjadi pada jalur *feedwater* menuju *burner*. Kebocoran ini dapat menyebabkan temperatur pemanasan yang tidak sesuai dengan *set point* yang diinginkan sehingga dapat berdampak fatal bagi sistem tersebut. Selain itu, *valve* yang tidak membuka sesuai dengan *set point* juga dapat mengakibatkan temperatur pemanasan yang tidak stabil. Temperatur pemanasan yang tidak stabil dapat memicu terjadinya kebakaran dan ledakan [5]. Sehingga hal ini harus segera ditangani untuk mencegah kegagalan yang lebih parah dan kerugian yang semakin besar. Untuk mencegah bahaya-bahaya yang dapat terjadi maka perlu adanya sistem keamanan pada *burner*. BPCS dan sistem pengontrol sebelumnya memiliki kemungkinan untuk gagal, untuk itu SIS dibutuhkan. *Piping and Instrumentation Diagram (P&ID)* dari *burner* dilampirkan pada lampiran A.

Pada Gambar A.1 pada lampiran A dapat dilihat bahwa terdapat beberapa instrumen pada *node burner*. *Flow Indicator (FI-6217)* berfungsi untuk membaca *flow* dari *fuel inlet burner*. *Flow indicator* akan mendeteksi jika terjadi kekurangan *flow*. Kekurangan *flow* dapat diakibatkan oleh penurunan dari proses, bukaan *valve* yang tidak sesuai dengan *set point* dan menurunnya intensitas *feedwater inlet burner*. Hal ini dapat menyebabkan waktu pemanasan *feedwater* semakin lama, temperatur pemanasan yang tidak stabil dan tidak sesuai dengan *set point*. Jika *flow indicator* membaca adanya kekurangan *flow*, maka *Flow Valve (FV-6217)* akan menjalankan tugasnya untuk membuka bukaan *valve* lebih besar untuk meningkatkan *flow*. Apabila *flow valve* tidak dapat menangani kekurangan *flow* tersebut, maka *Safety Valve (SV-6214)* akan bekerja. Selain *flow indicator* juga

terdapat *Pressure Indicator* (PI-6271). *Pressure indicator* diperlukan untuk membaca *pressure* agar tidak terlalu tinggi. *Pressure* yang terlalu tinggi dapat diakibatkan oleh *volume fuel inlet burner* yang meningkat dan *Pressure Valve* (PV-6217) yang tidak berfungsi secara normal. Kelebihan *pressure* ini dapat diatasi dengan menggunakan *Pressure Switch High* (PSH-6217), *Pressure Valve* (PV-6217) dan *Safety Valve* (SV-6217). Pada Gambar 2.1 dapat dilihat bahwa terdapat SIS yang terdiri dari *flow* dan *pressure transmitter* yang tersambung dengan *valve* dan aktuatur sebagai *final element*. SIS akan berfungsi apabila *flow valve* tidak dapat menangani kekurangan *flow* yang terjadi. SIS juga akan menjalankan tugasnya sebagai sistem keamanan apabila *pressure switch high* dan *pressure valve* tidak dapat menangani kelebihan tekanan yang terjadi.

2.2 *Safety Instrumented System*

Untuk memenuhi sistem keamanan, *layer of protection* perlu dipertimbangkan. *Layer of protection* terdiri dari desain *burner*, *Basic Process Control System* (BPCS) *burner*, sistem alarm, *mechanical safety*, dan *Safety Instrumented System* (SIS). Pemasangan SIS diperlukan apabila tahapan lain pada *layer of protection* tidak dapat menangani keadaan berbahaya yang terjadi. Ketika BPCS sudah tidak dapat mengontrol suatu variabel agar sesuai dengan *set point*, maka SIS akan berfungsi. SIS akan segera mematikan seluruh sistem untuk menghentikan proses produksi pada tempat terjadinya kecelakaan. Hal ini bertujuan untuk meminimalisir dampak bahaya yang dapat disebabkan terhadap lingkungan sistem tersebut. SIS merupakan suatu tahap pada *layer of protection* yang berfungsi untuk mengurangi resiko pada suatu sistem yang berpotensi menyebabkan keadaan berbahaya [14]. SIS diimplementasikan sebagai *Emergency Shutdown System* (ESD) dan sistem pendeteksi *Fire and Gas* (F&G) [1]. SIS terdiri dari satu atau beberapa *Safety Instrumented Function* (SIF). SIF itu sendiri terdiri dari *sensor*, *logic solver* dan *final element* [14]. SIS berfungsi untuk mencapai atau menjaga suatu keadaan yang dapat dikatakan aman. SIS digunakan atau

dipasang pada sistem yang memiliki komponen elektrik, berhubungan langsung dengan orang serta sistem pneumatik dan hidraulik [14]. Karena SIS melibatkan pihak-pihak seperti pabrik, *designer*, kontraktor, operator, pemerintah dan asuransi, diperlukan standar yang menjadi media komunikasi untuk memenuhi sistem keamanan. Salah satu standar yang digunakan adalah IEC 61508.

IEC 61508 merupakan standar internasional yang dipublikasikan oleh *International Electrotechnical Commission* yang membahas fungsi proteksi dari sistem-sistem yang terkait dengan E/E/PE atau E/E/PES [2]. Dalam standar ini dijelaskan metode-metode untuk membangun spesifikasi keamanan yang diperlukan untuk mencapai target *Safety Instrumented Level (SIL)* yang telah ditetapkan. Pendekatan yang digunakan dalam mencapai target SIL adalah dengan pendekatan resiko. IEC 61508 digunakan sebagai acuan untuk keamanan pada suatu proses industri yang menggunakan peralatan-peralatan terkait E/E/PE atau E/E/PES. IEC 61508 membahas seluruh keamanan yang harus *diterapkan* pada suatu proses industri [2]. Selain itu juga dijelaskan seluruh keadaan bahaya yang dapat terjadi pada pada *safety function* yang dijalankan peralatan-peralatan elektrik. Selain membahas sistem keamanan pada suatu peralatan elektrik, dalam standar ini juga dibahas mengenai dampak yang dapat terjadi terhadap seseorang dan lingkungan yang diakibatkan oleh suatu kesalahan dan kegagalan pada proses industri.

Kegagalan yang terjadi pada suatu proses industri dapat berdampak besar bagi suatu perusahaan. Apabila kegagalan terjadi, dapat menyebabkan cedera terhadap orang-orang di sekitar *plant* tersebut. Kegagalan juga menyebabkan kerusakan pada peralatan-peralatan yang terdapat pada proses tersebut. Kegagalan tersebut dapat berdampak besar bagi suatu perusahaan dan menyebabkan kerugian yang berarti. Penting bagi suatu perusahaan untuk memperhatikan sistem keamanan. IEC 61508 diterapkan untuk meminimalisir potensi terjadinya kecelakaan dalam suatu proses industri.

Burner merupakan suatu komponen yang berpotensi *menyebabkan* bahaya. SIS perlu diimplementasikan pada *burner* untuk menjaga sistem agar segera dapat dilakukan eksekusi jika terjadi keadaan bahaya. SIS pada *burner* secara umum berfungsi untuk mematikan sistem jika terdeteksi adanya kecelakaan dan keadaan bahaya.

2.3 Dependability

Dependability merupakan property sistem yang *mengintegrasikan availability, safety, security, survivability, dan maintainability* [11]. *Dependability* merupakan tingkat kepercayaan *user* terhadap suatu sistem. *Dependability* dapat ditentukan dari nilai rata – rata PFD_{avg} dan STR. Perhitungan PFD_{avg} merupakan probabilitas kegagalan dan merupakan perhitungan dasar untuk menentukan SIL dari suatu sistem. STR mempresentasikan kerugian produksi akibat terjadinya kegagalan pada SIF [4]. Oleh karena itu penting untuk memperhatikan kedua nilai tersebut, baik PFD_{avg} dan STR.

Persamaan berikut merupakan persamaan yang digunakan untuk menghitung PFD_{avg} untuk *dangerous independent failure basic event* [15].

$$PFD = (1 - C_{MooN} \times \beta_{DD}) \times \lambda_{DD} \times T_r + (1 - C_{MooN} \times \beta_{DU}) \times \lambda_{DU} \times \left(\frac{TI}{2} + T_r\right) \quad (2.1)$$

Sedangkan persamaan berikut digunakan untuk perhitungan STR untuk *dangerous independent failure basic event* [15].

$$STR = (1 - C_{MooN} \times \beta_{SD}) \times \lambda_{SD} + (1 - C_{MooN} \times \beta_{SU}) \times \lambda_{SU} \quad (2.2)$$

dimana:

PFD = Probability Failure on Demand

STR = Spurious Trip Rate

C_{MooN} = Fraksi kegagalan Moon architecture

λ_{DD}	= Laju kegagalan <i>dangerous detected failures</i>
λ_{DU}	= Laju kegagalan <i>dangerous undetected failures</i>
λ_{SD}	= Laju kegagalan <i>safe detected failures</i>
λ_{SU}	= Laju kegagalan <i>safe undetected failures</i>
β_{DD}	= Beta faktor <i>dangerous detected failures</i>
β_{DU}	= Beta faktor <i>dangerous undetected failures</i>
β_{SD}	= Beta faktor <i>safe detected failures</i>
β_{SU}	= Beta faktor <i>safe undetected failures</i>
T_r	= <i>Repair Time</i>
TI	= <i>Time Interval</i>

Persamaan 2.1 dan 2.2 merupakan persamaan yang digunakan untuk menghitung PFD_{avg} dan STR. Dari kedua persamaan tersebut dapat dilihat bahwa nilai PFD_{avg} dan STR dipengaruhi oleh C_{Moon} , *failure rate* dan fraksi kegagalan. Nilai PFD_{avg} dipengaruhi oleh *danger failure rate* dan fraksi kegagalan bahaya baik *detected* maupun *undetected* sedangkan nilai STR dipengaruhi oleh *safe failure rate* dan fraksi kegagalan aman baik *detected* maupun *undetected*. Selain itu, nilai PFD_{avg} juga dipengaruhi oleh *repair time* dan *Time Interval*. Melalui besarnya nilai *Time Interval* dapat diketahui apakah sistem tersebut termasuk *low demand* atau *high demand*.

Hasil perhitungan PFD_{avg} akan digunakan untuk menentukan SIL berdasarkan PFD_{avg} yang sesuai dengan IEC 61508 *part 1*. Tabel 2.1 merupakan tabel untuk menentukan SIL berdasarkan PFD_{avg} sesuai dengan IEC 61508 *part 1*.

Tabel 2.1 Korelasi SIL dan PFD_{avg} untuk *Low Demand Mode* [2]

SIL	PFD_{avg}
4	$\geq 10^{-5} < 10^{-4}$
3	$\geq 10^{-4} < 10^{-3}$
2	$\geq 10^{-3} < 10^{-2}$
1	$\geq 10^{-2} < 10^{-1}$

SIL dapat ditentukan melalui nilai PFD_{avg} , namun selain itu SIL *final* harus dipertimbangkan. SIL *final* merupakan SIL yang

ditentukan melalui nilai *Hardware Fault Tolerance* (HFT), *Safe Failure Fraction* (SFF) dan tipe untuk masing – masing teknologi [2]. Untuk menentukan SIL *final* berdasarkan SFF dan *fault tolerance* (tipe teknologi dan HFT) dijelaskan pada Tabel 2.2.

Tabel 2.2 SIL *Architectural Constraint* [2]

	Tipe	A			B		
	HFT	0	1	2	0	1	2
SFF (%)	< 60	SIL1	SIL2	SIL3	N/A*	SIL1	SIL2
	60 - < 90	SIL2	SIL3	SIL4	SIL1	SIL2	SIL3
	90 - < 99	SIL3	SIL4	SIL4	SIL1	SIL3	SIL4
	≥ 99	SIL3	SIL4	SIL4	SIL3	SIL4	SIL4

**Not Allowed*

Pada Tabel 2.2 terdapat 2 tipe, yaitu A dan B. Tipe A digunakan untuk komponen dengan mode kegagalan yang diketahui dan data kegagalan dari lapangan dapat diandalkan. Contoh dari tipe A adalah *relay*, *solenoida*, *valve* dan modul elektronik sederhana (*conventional transmitter*). Sedangkan komponen-komponen yang tidak memenuhi syarat pada tipe A termasuk pada tipe B. Contoh dari tipe B adalah komponen yang menggunakan *microprocessor* seperti *smart transmitter*. Tidak hanya ditentukan berdasarkan tipe, SIL *final* juga ditentukan berdasarkan HFT dan SFF. HFT merupakan merupakan batas kegagalan yang dapat ditoleransi sebelum menyebabkan kegagalan sedangkan SFF merupakan persentase kegagalan yang dianggap aman.

2.4 *Lifecycle Cost*

Lifecycle Cost (LCC) merupakan dasar untuk mencapai sistem keamanan yang baik. Suatu sistem keamanan harus memiliki biaya yang efektif [15]. Suatu sistem keamanan dengan biaya yang mahal akan menjadi hambatan bagi perusahaan, karena kerugian yang disebabkan jika adanya kecelakaan bisa saja lebih murah jika dibandingkan dengan biaya sistem keamanan yang harus dipasang. Sehingga suatu perusahaan lebih memilih

untuk tidak memasang sistem keamanan tersebut. LCC perlu diperhatikan agar biaya yang dikeluarkan suatu perusahaan untuk memasang sistem keamanan dapat ditoleransi. Sehingga suatu perusahaan tidak mengambil resiko dengan tidak memasangnya sistem keamanan yang terlalu mahal.

Persamaan berikut merupakan persamaan yang digunakan untuk menghitung LCC [15].

$$LCC = C_{pro} + (C_{op} + C_{risk}) \times PVF \quad (2.3)$$

$$C_{pro} = C_{design} + ((C_{purchase} + \frac{C_{inst}}{comm}) \times N_{valve} + (C_{purchase} + \frac{C_{inst}}{comm}) \times N_{transmitter}) \times jml\ SIF + (C_{purchase} + \frac{C_{inst}}{comm})_{PLC,SIS} \quad (2.4)$$

$$C_{OP} = C_{PM} + C_T + C_{CM} \quad (2.5)$$

$$C_{PM} = \frac{1}{M} \times C_{pm} \times N \quad (2.6)$$

$$C_T = \frac{1}{Ti} \times C_t \times N \quad (2.7)$$

$$C_{CM} = \lambda_t \times C_{cm} \times N \quad (2.8)$$

$$C_{risk} = C_{STR} + C_{hazard} \quad (2.9)$$

$$C_{STR} = STR \times C_{SD} \quad (2.10)$$

$$C_{SD} = SD_{time} \times SD_{loss} \quad (2.11)$$

$$C_{hazard} = C_{ACC} \times F(ACC|PFD_{avg}) \times PFD_{avg} \quad (2.12)$$

$$PVF = \frac{1 - (1 + R)^{-T}}{R} \quad (2.13)$$

dimana:

LCC	= <i>Life Cycle Cost</i>
C_{pro}	= <i>Procurement cost</i>
C_{op}	= <i>Operational cost</i>
C_{risk}	= <i>Risk cost</i>
PVF	= <i>Factor by Present Value</i>
C_{design}	= <i>Design cost</i>
$C_{purchase}$	= <i>Purchase cost</i>
$C_{inst/comm}$	= <i>Installation cost</i>

N	= Jumlah MooN voting
C_{PM}	= Predictive maintenance cost
C_T	= Test cost
C_{CM}	= Corrective maintenance cost
λ_T	= Total failure rate
C_{STR}	= Spurious Trip Rate cost
C_{hazard}	= Cost of hazard
STR	= Spurious Trip Rate
C_{SD}	= Shut down cost
SD_{time}	= Shut down time
SD_{loss}	= Shut down loss
C_{ACC}	= Catastrophic loss
$F(ACC PFD_{avg})$	= Plant risk without SIS
PFD_{avg}	= Average Probability Failure on Demand
R	= Discount rate
T	= Years life

Persamaan 2.3 sampai 2.10 merupakan persamaan yang digunakan untuk menghitung nilai LCC. Dari persamaan 2.3 dapat dilihat bahwa nilai LCC dipengaruhi oleh *procurement cost*, *operational cost*, *risk cost* dan *PVF*. Perhitungan *procurement cost* ditunjukkan melalui persamaan 2.4. Dari persamaan 2.4 dapat dilihat bahwa *procurement cost* dipengaruhi oleh jumlah SIF yang terdapat dalam sistem. *Operational cost* merupakan jumlah dari *maintenance cost* yang ditunjukkan melalui persamaan 2.5. *Maintenance cost* terdiri dari *predictive maintenance cost*, *test cost* dan *corrective maintenance cost* yang ditunjukkan melalui persamaan 2.6, 2.7 dan 2.8. Hubungan antara *dependability* dan *cost* dapat dilihat pada persamaan 2.10 dan 2.12. Melalui kedua persamaan tersebut diketahui bahwa nilai PFD_{avg} dan STR mempengaruhi nilai dari *risk cost*. Perhitungan nilai *PVF* ditunjukkan melalui persamaan 2.13. *PVF* merupakan *Factor by Present Value*. Dari persamaan 2.13 dapat dilihat bahwa nilai *PVF* dipengaruhi oleh *discount rate* dan umur SIS.

2.5 Diversity

Diversity merupakan unit atau komponen yang memiliki beragam desain disusun dalam suatu redundansi. Adanya *diversity* bertujuan untuk mengurangi kerentanan suatu unit atau komponen yang diakibatkan oleh *common cause failure*. Menurut IEC 61508, *diversity* dan redundansi merupakan salah satu cara untuk mencegah terjadinya kegagalan [1].

Diversity Index (DI) digunakan untuk menganalisa pengaruh *diversity* pada sistem berdasarkan *dependability* dan *cost*. DI mencerminkan tingkat redundansi dan *diversity* terhadap jumlah komponen. Persamaan berikut merupakan persamaan yang digunakan untuk menghitung (DI) [15].

$$DI = \frac{1}{Ns} \sum_i^{Ns} \frac{Technologies_i \times Redundancy_i}{Components_i} \quad (2.11)$$

dimana:

DI = *Diversity Index*

Ns = Total komponen yang digunakan

Technologies_i = Teknologi ke-i

Redundancy_i = Redundansi ke-i

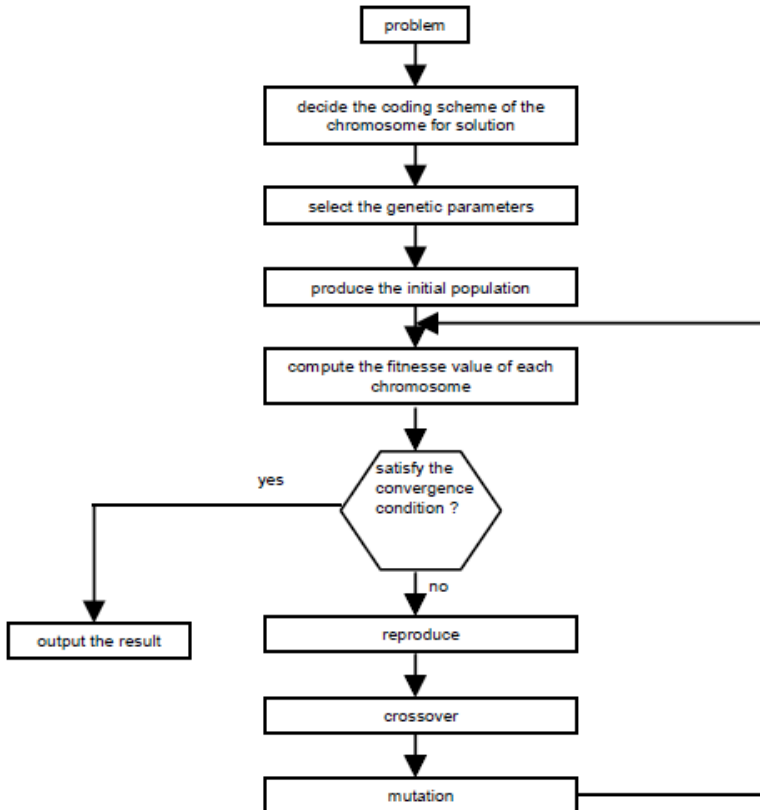
Components_i = Komponen ke-i

Dari persamaan 2.11 dapat dilihat bahwa nilai DI dipengaruhi oleh nilai *Ns*, teknologi, redundansi dan komponen. Nilai *Ns* menunjukkan jumlah subsistem yang ada. Redundansi merupakan adanya penambahan jumlah komponen yang digunakan untuk meningkatkan *dependability*. Nilai redundansi ditentukan melalui *MooN voting* yang digunakan yang dapat diwakilkan dengan perhitungan $n - m$ dimana nilai n menunjukkan jumlah keseluruhan komponen yang digunakan sedangkan nilai m merupakan jumlah komponen yang akan menyebabkan kegagalan terjadi.

2.6 Genetic Algorithm

Genetic Algorithm (GA) merupakan salah satu metode multi objektif yang digunakan untuk memecahkan masalah berbasis proses seleksi alam dengan meniru evolusi biologi [12]. GA merupakan jenis algoritma pencarian dan pengoptimalan metaheuristik yang terinspirasi oleh prinsip seleksi alam Darwin [16]. Metode GA merupakan metode yang dapat memecahkan permasalahan yang rumit, dimensi yang tinggi, diskrit, non-linear, dan diskontinyu [1]. Dengan menggunakan metode GA memungkinkan untuk mendapatkan keseimbangan antara *dependability* dan *cost* [13]. Selain itu, metode GA juga memungkinkan bagi SIS untuk mengeksekusi lebih dari satu SIF. GA digunakan untuk mengoptimisasi suatu sistem dengan cara mengambil secara acak individu di dalam suatu populasi dengan menyocokkan dengan hasil yang diinginkan, kemudian dilakukan *crossover* atau reproduksi terhadap individu tersebut sehingga menghasilkan keturunan sesuai dengan yang dikehendaki [1, 12].

GA yang efektif dipengaruhi oleh operator *crossover* dan komplementer mutasinya. Tingkat keefektifitasan operator *crossover* menentukan tingkat konvergensi, sedangkan operator mutasi berperan dalam mencegah algoritma dalam menghasilkan hasil optimisasi berupa hasil lokal yang optimal. Jumlah anak – anak dan mutan yang dihasilkan pada tiap generasi merupakan suatu parameter yang konstan selama dilakukannya optimisasi [5].



Gambar 2.1 Diagram alir GA

Gambar 2.2 menunjukkan diagram alir GA secara umum. Dari Gambar 2.2 dapat dilihat bahwa tahap pertama dalam melakukan optimisasi dengan menggunakan GA dimulai dengan menentukan fungsi objektif [17]. Tahap selanjutnya adalah menentukan parameter-parameter optimisasi seperti kromosom, *elitism*, *crossover rate* dan *mutation rate*. Kemudian akan dilakukan proses kawin silang terhadap individu-individu yang telah didapatkan melalui proses seleksi yang mirip dengan proses seleksi alam. Individu-individu tersebut direpresentasikan dengan kromosom yang telah ditentukan. Tahap berikutnya

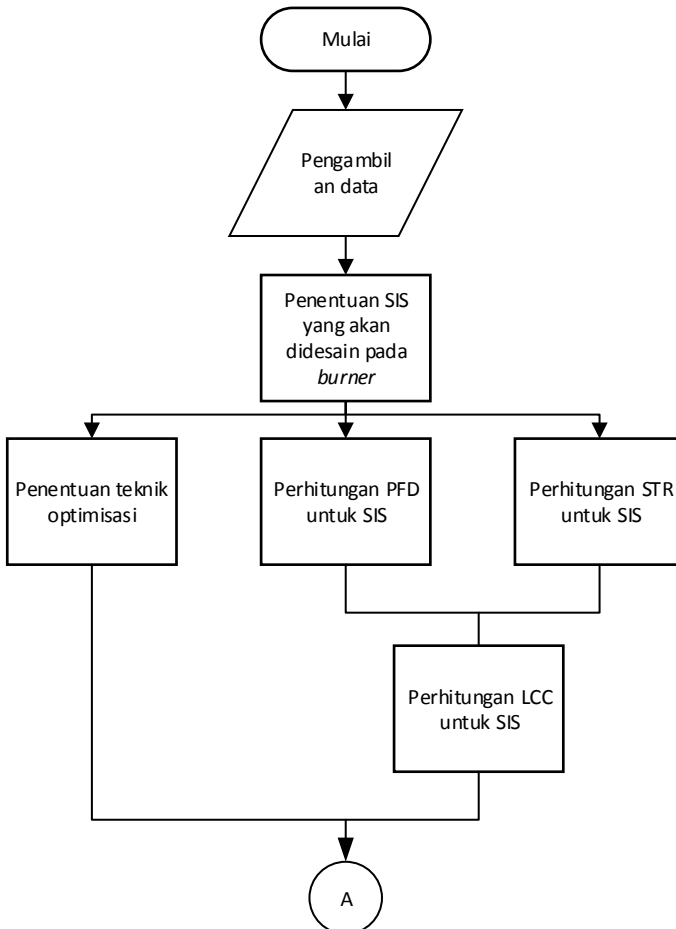
adalah dilakukannya mutasi terhadap individu-individu baru tersebut. Proses tersebut akan berulang hingga telah mencapai iterasi yang telah ditentukan.

Tahap pertama yang dilakukan dalam melakukan optimisasi adalah menentukan fungsi objektif. Variabel yang akan dioptimisasi pada Tugas Akhir ini adalah arsitektur MoonN *voting* dan teknologi dengan objek PFD_{avg} , STR, dan LCC. Sedangkan fungsi objektif pada Tugas Akhir ini adalah LCC. Dari penjelasan sebelumnya, dapat diketahui bahwa hasil optimisasi yang diinginkan adalah terdapat penerunan terhadap tiga nilai tersebut, yaitu PFD_{avg} , STR, dan LCC. Nilai PFD_{avg} yang lebih kecil menunjukkan bahwa probabilitas terjadinya kecelakaan di dalam *plant* tersebut lebih kecil. Nilai STR yang lebih kecil akan mengurangi kerugian produksi yang terjadi akibat kegagalan SIF. Nilai LCC menunjukkan biaya yang dibutuhkan untuk keamanan dari suatu *plant* dan dari hasil optimisasi diharapkan didapatkan nilai LCC yang paling efektif.

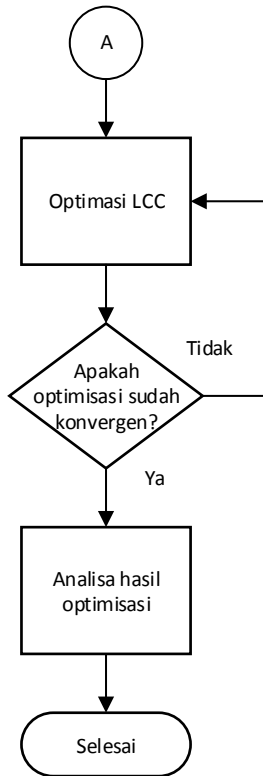
“Halaman ini memang dikosongkan”

BAB III METODOLOGI PENELITIAN

Pada bagian ini akan dijelaskan mengenai metodologi dalam melakukan pengerjaan Tugas Akhir, sehingga langkah-langkah pengerjaan menjadi lebih sistematis dan terorganisir. Berikut merupakan *flowchart* dilakukannya Tugas Akhir



Gambar 3.2 Diagram Alir Tugas Akhir



Gambar 3.1 Diagram Alir Tugas Akhir (Lanjutan)

3.1 Pengambilan Data

Proses pengambilan data akan dilakukan di PT Petrokimia Gresik Pabrik III berupa data HAZOP yang dilampirkan pada lampiran B, P&ID, kapasitas produksi dan *shutdown time* yang dilampirkan pada lampiran C. Karena ketidaksediaan data di pabrik, maka data harga listrik, λ_{DD} , λ_{DU} , λ_{SD} , λ_{SU} , λ_t , β_{DD} , β_{DU} , β_{SD} , β_{SU} , tipe, SFF, *cost purchase*, *cost design*, *cost installation*, *maintenance cost*, *test cost*, *repair cost*, *start up cost*, *catastrophic cost*, *SIS life*, *discount rate*, *failure fraction* MooN, *repair time*, *Test Interval* dan *plant risk without SIS* diambil dari data

sekunder, baik dari buku maupun jurnal. Tabel 3.1 – 3.7 menunjukkan data – data yang telah disebutkan di atas.

Tabel 3.3 Data *Dependability Flow Transmitter* [1]

	<i>Flow Transmitter</i>		
	A	B	C
$\lambda_{DD}(x10^{-6}/hr)$	0,046	0,97	0,41
$\lambda_{DU}(x10^{-6}/hr)$	0,103	0,93	3,70
$\lambda_{SD}(x10^{-6}/hr)$	0,265	1,21	0,68
$\lambda_{SU}(x10^{-6}/hr)$	0,118	0,95	6,13
$\lambda_t(x10^{-6}/hr)$	0,534	4,06	10,92
β_{DD}	0,05	0,1	0,1
β_{DU}	0,1	0,1	0,1
β_{SD}	0,1	0,1	0,1
β_{SU}	0,1	0,1	0,1
Type	B	A	A
SFF (%)	80,7	77,9	66,12
Cost purchase (\$)	4844	3206	500

Tabel 3.4 Data *Dependability Final Element* [1]

	<i>Final Element</i>		
	A	B	C
$\lambda_{DD}(x10^{-6}/hr)$	0,84	1,09	0,79
$\lambda_{DU}(x10^{-6}/hr)$	2,51	4,35	7,11
$\lambda_{SD}(x10^{-6}/hr)$	0	0	0
$\lambda_{SU}(x10^{-6}/hr)$	3,94	3,17	9,17
$\lambda_t(x10^{-6}/hr)$	7,29	8,61	17,07
β_{DD}	0,1	0,1	0,1
β_{DU}	0,1	0,1	0,1
β_{SD}	0,1	0,1	0,1
β_{SU}	0,1	0,1	0,1
Type	A	A	A
SFF (%)	65,57	49,48	58,35
Cost purchase (\$)	6940	6400	6200

Tabel 3.5 Data *Dependability Logic Solver* [1]

	Logic Solver
$\lambda_{DD}(x10^{-6}/hr)$	0,026
$\lambda_{DU}(x10^{-6}/hr)$	0,006
$\lambda_{SD}(x10^{-6}/hr)$	1,73
$\lambda_{SU}(x10^{-6}/hr)$	1,73
$\lambda_i(x10^{-6}/hr)$	3,492
β_{DD}	0,1
β_{DU}	0,1
β_{SD}	0,1
β_{SU}	0,1
Type	B
SFF (%)	99,83
Cost purchase (\$)	40000

Tabel 3.6 Data *Lifecycle Cost Instrument* [1]

	Unit	Cost
<i>Design overall instrumentation</i>	\$	3060
<i>Installation/commisioning per instrument</i>	\$	600
<i>Maintenance per instrument</i>	\$/event	240
<i>Test per instrument</i>	\$/event	60
<i>Repair cost per instrument</i>	\$/event	480
<i>Start up cost</i>	\$	1800
<i>Catastrophic cost</i>	\$	150 x 10 ⁶
<i>SIS life</i>	years	15
<i>Discount rate</i>	%	5

Tabel 3.7 Data *Lifecycle Cost PLC* [1]

	Unit	Cost
<i>Design/install/Commisioning</i>	\$	10320
<i>Repair</i>	\$/event	8000
<i>Maintenance</i>	\$/event	60
<i>Test</i>	\$/event	240

Tabel 3.8 Data Lain [1]

	Unit	Nilai
<i>Repair time</i>	<i>hours</i>	8
<i>Test Interval</i>	<i>year</i>	1
<i>Plant risk without SIS</i>	<i>per year</i>	$8,55 \times 10^{-3}$

Tabel 3.9 Nilai Fraksi Kegagalan Vote Arsitektur [1]

MooN	N	C
1oo1	1	0
1oo2	2	1
1oo3	3	0,3
1oo4	4	0,15
1oo5	5	0,08
1oo6	6	0,04
2oo2	2	0
2oo3	3	2,4
2oo4	4	0,75
2oo5	5	0,45
2oo6	6	0,26
3oo3	3	0
3oo4	4	4,0
3oo5	5	1,2
3oo6	6	0,8
4oo4	4	0
4oo5	5	6,0
4oo6	6	1,6
5oo5	5	0
5oo6	6	8,1
6oo6	6	0

Tabel 3.1 menunjukkan data *dependability* untuk *flow transmitter* dengan 3 macam teknologi. Teknologi A merupakan *smart transmitter*, teknologi B merupakan *conventional electronic transmitter* dan teknologi C merupakan *switch*. Sedangkan Tabel

3.2 menunjukkan data *dependability* untuk *final element* dengan 3 macam teknologi. Teknologi A merupakan *air operated*, teknologi B merupakan *hydraulic operated* dan teknologi C merupakan *motor operated*. Tabel 3.3 menunjukkan data *dependability* untuk *logic solver*. *Logic solver* yang digunakan adalah berupa *safety PLC*. Data mengenai *lifecycle cost* baik untuk *instrument* maupun *PLC* ditunjukkan pada Tabel 3.4 dan 3.5. Sedangkan data lain seperti *repair time*, *shut down time*, *Test Interval* dan *plant risk without SIS* ditunjukkan pada Tabel 3.6. Tabel 3.7 menunjukkan data fraksi kegagalan *MooN voting* yang terdiri dari 21 kominasi *MooN* yaitu 1001 hingga 6006.

3.2 Penentuan SIS yang Akan Didesain

Penentuan SIS dilakukan berdasarkan data HAZOP dan P&ID yang telah didapatkan dari PT Petrokimia Gresik Pabrik III. Berdasarkan data tersebut, dapat ditentukan bahwa terdapat 1 SIF yang terdiri dari *flow control*.

3.3 Penentuan Teknik Optimisasi

Penentuan teknik optimisasi dilakukan berdasarkan penelitian – penelitian yang telah dilakukan. Menurut penelitian yang telah dilakukan, teknik optimisasi terbaik untuk mengoptimasi LCC adalah *Genetic Algorithm*.

3.4 Perhitungan PFD_{avg} dan STR

Setelah menentukan SIS yang akan didesain, maka akan dilakukan analisa dari PFD_{avg} dan STR. Persamaan 3.1 dan 3.2 merupakan persamaan-persamaan untuk mendapatkan nilai-nilai tersebut.

Berikut persamaan PFD untuk *dangerous independent failure basic event*.

$$PFD = (1 - C_{MooN} \times \beta_{DD}) \times \lambda_{DD} \times T_r + (1 - C_{MooN} \times \beta_{DU}) \times \lambda_{DU} \times \left(\frac{TI}{2} + T_r\right) \quad (3.1)$$

Berikut persamaan STR untuk *dangerous independent failure basic event*.

$$STR = (1 - C_{Moon} \times \beta_{SD}) \times \lambda_{SD} + (1 - C_{Moon} \times \beta_{SU}) \times \lambda_{SU} \quad (3.2)$$

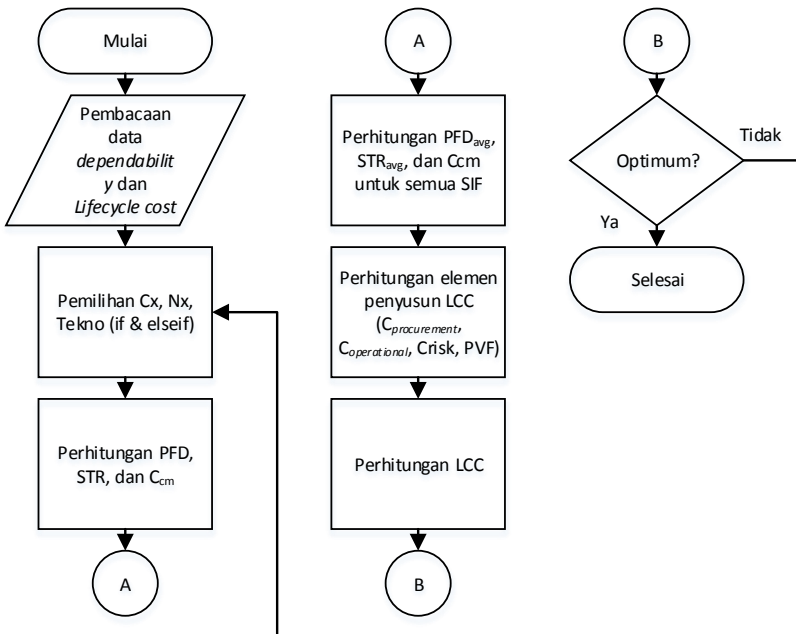
3.5 Perhitungan LCC

Setelah didapatkan nilai PFD_{avg} dan STR kemudian akan dilakukan perhitungan nilai LCC. Persamaan 3.3 menunjukkan persamaan yang digunakan untuk menentukan nilai LCC.

$$LCC = C_{pro} + (C_{op} + C_{risk}) \times PVF \quad (3.3)$$

3.6 Optimisasi SIS

Optimisasi SIS yang dilakukan dengan menggunakan *Genetic Algorithm* dengan fungsi objektif LCC.



Gambar 3.3 Flowchart Program MATLAB R2016a

Optimisasi SIS dilakukan dengan menggunakan *software* MATLAB R2016a dengan constraint MoonN *voting* yang terdiri dari 21 variasi dan teknologi dengan 3 macam variasi untuk masing – masing *transmitter* dan *final element*. Sedangkan *objective function* yang digunakan adalah LCC. Perhitungan performansi dilakukan dengan melakukan *running* program optimisasi pada MATLAB R2016a untuk mendapatkan hasil optimisasi yang optimal dan akurat. Gambar 3.2 menunjukkan *flowchart* program pada MATLAB R2016a.

Optimisasi akan dimulai dengan pembacaan data *dependability* dan *lifecycle cost* yang akan dilanjutkan dengan pemilihan variabel Nx, Cx dan teknologi. Variabel Nx merupakan jumlah komponen yang digunakan dan variabel Cx merupakan nilai C_{Moon} . Penentuan nilai Cx sesuai dengan nilai Nx. Perhitungan PFD dan STR akan dimulai ketika semua variabel penyusun sudah ditentukan. Kemudian akan dilanjutkan dengan perhitungan PFD_{avg} , STR_{avg} , C_{cm} , C_{pm} dan C_t . Perhitungan elemen penyusun LCC akan dilakukan setelahnya. Apabila hasil sudah optimum maka optimisasi telah selesai, jika hasil belum optimum maka proses akan diulang dari pemilihan variabel Nx, Cx dan teknologi. Parameter – parameter yang digunakan dalam mengoptimisasi SIS dengan menggunakan *Genetic Algorithm* [18] adalah sebagai berikut:

- Populasi : 100 individu
- Generasi : 60
- *Elistism* : 5%
- *Crossover rate* : 80%
- *Mutation rate* : 1%

3.7 Analisa Hasil Optimisasi

Analisa hasil optimisasi dengan mengamati nilai PFD_{avg} , STR dan LCC. Perhitungan PFD_{avg} mewakili probabilitas kegagalan dari sistem keamanan dan sebagai dasar dalam menentukan SIL. Perhitungan STR mewakili dampak dari kerugian produksi akibat kegagalan SIF. Perhitungan LCC merupakan perhitungan biaya yang dibutuhkan dalam mendesain, mengoperasikan dan

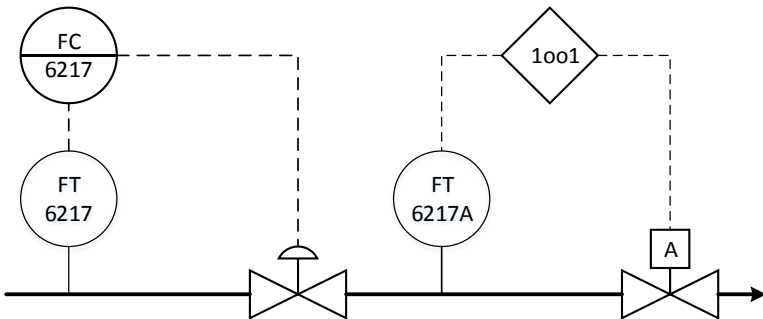
perawatan SIS. Pada Tugas akhir ini terdapat 1 SIF yang terdiri dari *flow transmitter*, *logic solver* berupa *safety PLC* dan *final element* yang terdiri dari aktuator dan *valve*. Tujuan optimisasi ini adalah untuk mendapatkan nilai LCC minimum.

“Halaman ini memang dikosongkan”

BAB IV ANALISIS HASIL DAN PEMBAHASAN

4.1 Perancangan *Safety Instrumented System* pada *Burner*

Berdasarkan P&ID dan HAZOP *burner*, akan dilakukan pemodelan SIS. Pada perancangan SIS untuk *burner* tersebut terdapat 1 SIF yang terdiri dari *flow control*. Pemilihan *flow control* dikarenakan konsekuensi yang dapat terjadi apabila terjadi kekurangan *flow* dari *fuel inlet burner*. Pada pemodelan ini tidak dibuat SIS dengan *pressure control* karena konsekuensi dari kelebihan tekanan yang tidak terlalu berdampak pada sistem. Kelebihan *pressure* tidak terlalu berdampak pada sistem dikarenakan jarang terjadi hal ini. Sehingga pada Tugas Akhir ini akan dirancang SIS untuk *burner* dengan 1 SIS yang terdiri atas *flow transmitter* sebagai *sensing element*, *safety PLC* sebagai *logic solver* serta aktuator dan *valve* sebagai *final element*. Gambar 4.1 menunjukkan pemodelan SIS yang telah dilakukan.



Gambar 4.4 Pemodelan SIS pada *Burner*

Dari Gambar 4.1 dapat dilihat bahwa terdapat 1 SIF yang terdiri dari *sensing element* berupa *flow transmitter*, *logic solver* berupa *safety PLC* dan *final element* yang terdiri atas aktuator dan *valve*. Selain itu dari gambar tersebut dapat dilihat BPCS yang terdiri dari *flow transmitter* dan *flow control*. Pemodelan SIS tersebut dilakukan sebagai sistem proteksi apabila BPCS tidak

dapat menangani kegagalan yang terjadi. Pemilihan teknologi dan redundansi untuk SIF yang telah dirancang akan ditentukan setelah optimisasi dilakukan. *Flow control* pada SIS bertujuan untuk menjaga *flow* dari *fuel inlet burner* agar tetap sesuai dengan *set point*. Kekurangan *flow* dapat diakibatkan oleh penurunan proses yang mengakibatkan turunnya aliran *flow*. Selain itu bukaan *valve* yang tidak sesuai dengan *set point* dan terjadinya penurunan intensitas pada *feedwater inlet burner* dapat menyebabkan turunnya aliran *fuel*. Penurunan aliran *flow* ini dapat menyebabkan waktu pemanasan *feedwater* yang semakin lama, temperatur pemanasan yang tidak stabil dan tidak sesuai dengan *set point*. *Flow Transmitter* (FT-6217) akan mendeteksi apabila terjadi penurunan *flow* yang kemudian sinyal akan diteruskan ke *Flow Control* (FC-6217) sehingga *Flow Valve* (FV-6217) membuka lebih besar untuk menambahkan *flow fuel*. Jika FV-6217 tidak dapat menangani kekurangan *flow* tersebut, maka FV-6217A akan menjalankan tugasnya untuk menambah *flow fuel*. Melalui gambar tersebut dapat dilihat bahwa melalui hasil optimisasi dirancang SIS untuk *burner* dengan *vote* arsitektur yang digunakan adalah 1oo1 untuk kedua *sensing element* dan *final element*. Teknologi yang digunakan untuk *sensing element* adalah *smart transmitter* sedangkan teknologi yang digunakan untuk *final element* adalah *air operated*.

4.2 Optimisasi *Dependability* dan *Cost*

Optimisasi *dependability* dan *cost* direpresentasikan melalui perhitungan PFD_{avg} , STR dan LCC. Optimisasi dilakukan bertujuan untuk mendapatkan nilai LCC terendah. Optimisasi dilakukan dengan memvariasikan redundansi dan teknologi yang digunakan untuk masing – masing subsistem. Terdapat 21 variasi redundansi yang digunakan yaitu *MooN voting architecture* dari 1oo1 hingga 6oo6. Pada penentuan SIS yang telah dilakukan sebelumnya telah ditentukan bahwa terdapat 1 SIF dengan *flow control*. Rangkaian SIS tersebut terdiri dari *sensing element* berupa *flow transmitter*, *logic solver* berupa *safety PLC* dan *final element* berupa aktuator dan *valve*. Variasi teknologi akan

digunakan pada *sensing element* dan *final element*. Teknologi yang digunakan untuk *sensing element* adalah *smart transmitter*, *conventional electronic transmitter* dan *switch*. Sedangkan teknologi yang digunakan untuk *final element* adalah *air operated*, *hydraulic operated* dan *motor operated*.

Perhitungan PFD_{avg} , STR, dan LCC dilakukan untuk masing-masing *vote* dengan variasi kombinasi teknologi dan tes interval sebesar 1 tahun (8760 jam). Optimisasi dilakukan dengan 60 kali iterasi yang menghasilkan 100 solusi untuk masing-masing iterasi. Tabel 4.1 dan 4.2 menunjukkan hasil optimisasi terbaik dan terburuk.

Tabel 4. 1 Hasil Optimisasi Maksimum dan Minimum

Hasil Optimisasi	PFD_{avg}	STR (/hour)	SIL PFD_{avg}	SIL Final	LCC (\$)
Maksimum	$1,07 \times 10^{-1}$	$4,10 \times 10^{-5}$	1	1	1.575.082
Minimum	1.16×10^{-2}	1.16×10^{-5}	1	1	225.638

Dari Tabel 4.1 dapat dilihat nilai maksimum dan minimum dari LCC dan nilai PFD_{avg} , STR, SIL PFD_{avg} dan SIL *final* saat nilai LCC maksimum dan minimum. Dari hasil tersebut kemudian akan didapatkan nilai SIL PFD_{avg} berdasarkan IEC 61508 *part 1* dan SIL *final* berdasarkan IEC 61508 *part 2*.

Tabel 4. 2 Hasil Variasi Redundansi dan Teknologi

	MooN		Subsistem						DI
	FT	FE	FT			FE			
			A	B	C	A	B	C	
Max	4oo4	1oo6	2	2	0	2	4	0	0,5
Min	1oo1	1oo1	1	0	0	1	0	0	0

Tabel 4.2 menunjukkan hasil variasi redundansi, teknologi dan nilai DI pada saat nilai LCC maksimum dan minimum. Tabel 4.3 menunjukkan hasil optimisasi pada beberapa solusi. Dari tabel tersebut dapat dilihat nilai PFD_{avg} , STR, SIL PFD_{avg} dan *final* serta nilai LCC. Tabel 4.4 menunjukkan hasil variasi redundansi,

teknologi dan nilai DI pada solusi-solusi yang ditunjukkan pada Tabel 4.3.

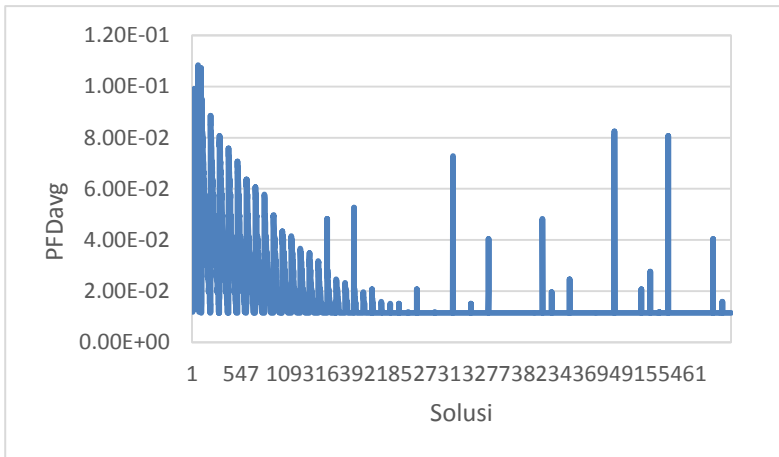
Tabel 4. 3 Hasil Optimisasi

Solusi ke-	PFD _{avg}	STR (hour)	SIL PFD _{avg}	SIL Final	LCC (\$)
6000	$1,16 \times 10^{-2}$	$1,16 \times 10^{-5}$	1	1	225.638
969	$1,19 \times 10^{-2}$	$1,43 \times 10^{-5}$	1	1	232.633
960	$1,60 \times 10^{-2}$	$1,91 \times 10^{-5}$	1	2	284.633
958	$1,65 \times 10^{-2}$	$2,41 \times 10^{-5}$	1	2	293.199
950	$2,11 \times 10^{-2}$	$3,19 \times 10^{-5}$	1	2	351.928
31	$3,13 \times 10^{-2}$	$1,47 \times 10^{-5}$	1	1	497.283
37	$4,04 \times 10^{-2}$	$1,75 \times 10^{-5}$	1	1	627.175
133	$6,35 \times 10^{-2}$	$2,26 \times 10^{-5}$	1	1	941.665
225	$6,68 \times 10^{-2}$	$2,74 \times 10^{-5}$	1	1	993.476
115	$8,25 \times 10^{-2}$	$2,89 \times 10^{-5}$	1	1	1.207.464

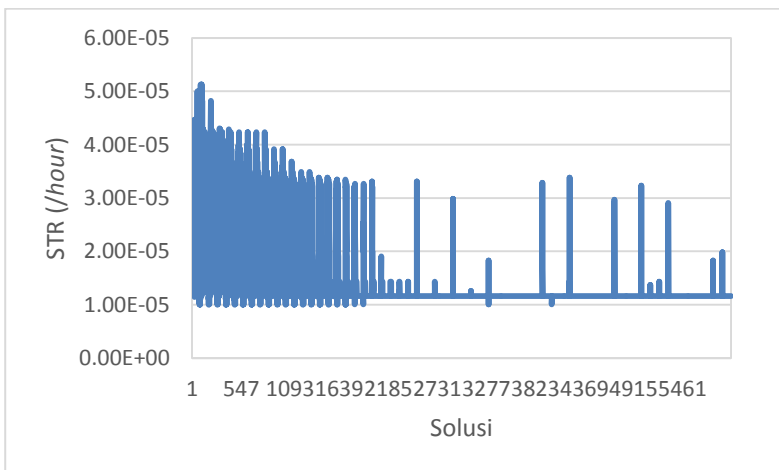
Tabel 4. 4 Hasil Variasi Redundansi dan Teknologi

Solusi ke-	Moon		Subsistem						DI
	FT	FE	FT			FE			
			A	B	C	A	B	C	
6000	1001	1001	1	0	0	1	0	0	0
969	1002	1001	2	0	0	1	0	0	0,167
960	1003	1001	2	1	0	1	0	0	0,5
958	1004	1001	3	1	0	1	0	0	0,6
950	1006	1001	5	1	0	1	0	0	0,71
31	1001	1002	0	1	0	1	1	0	0,33
37	1001	1003	1	0	0	2	1	0	0,5
133	1001	1004	0	1	0	2	2	0	0,6
225	1001	1005	0	1	0	4	1	0	0,67
115	1001	1006	1	0	0	4	2	0	0,71

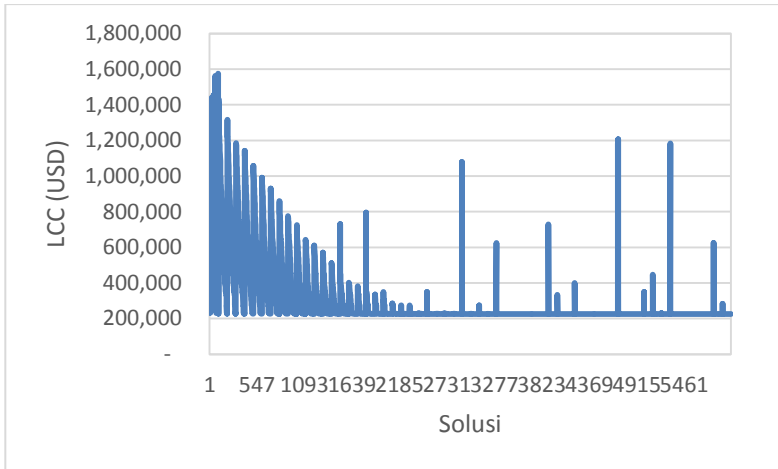
Dari hasil optimisasi yang telah dilakukan dengan 60 kali iterasi, maka didapatkan grafik seperti yang ditunjukkan pada Gambar 4.1, 4.2 dan 4.3.



Gambar 4.5 Hasil Optimisasi PFD_{avg}

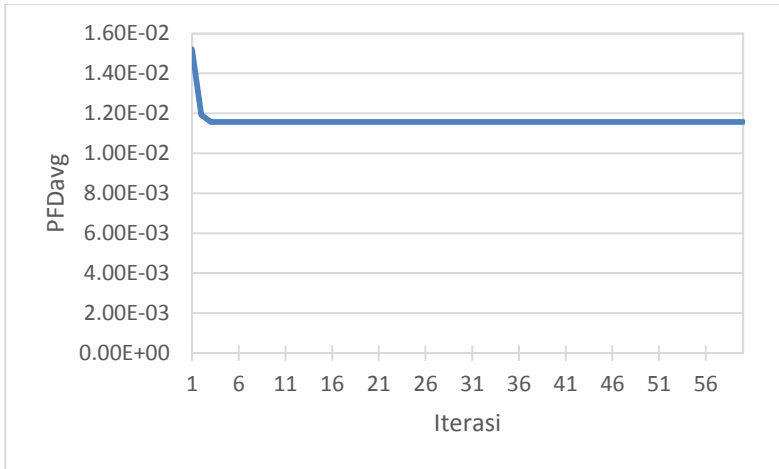


Gambar 4.6 Hasil Optimisasi STR

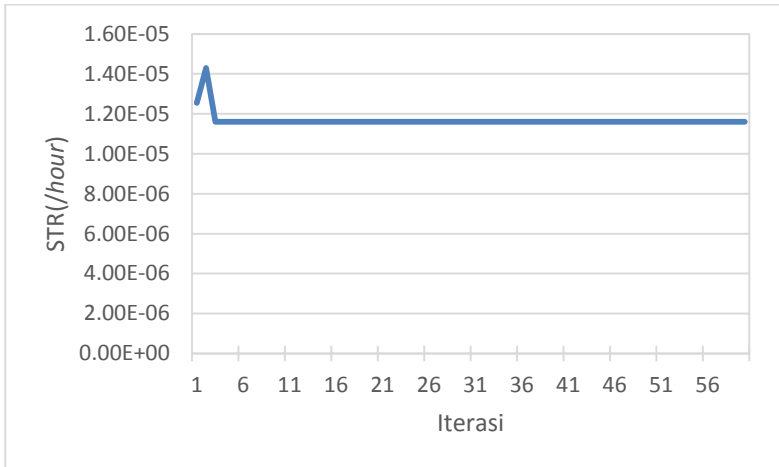


Gambar 4.7 Optimisasi LCC

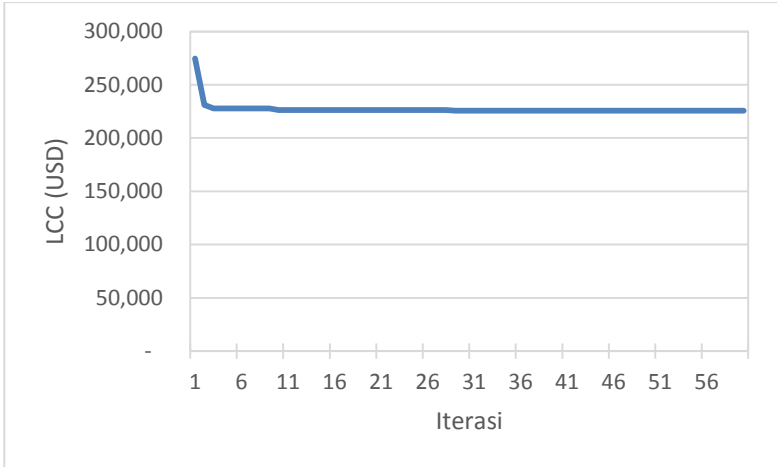
Gambar 4.1, 4.2 dan 4.3 menunjukkan grafik hasil optimisasi PFD_{avg} , STR dan LCC. Dari ketiga gambar di atas dapat dilihat bahwa masing – masing nilai PFD_{avg} , STR dan LCC mengalami penurunan. Namun terdapat beberapa kali kenaikan nilai yang dilihat pada ketiga gambar tersebut. Kenaikan nilai tersebut diakibatkan karena iterasi yang ditentukan belum selesai. Walaupun sudah didapatkan nilai minimum, namun *optimizer* tetap akan mencari nilai yang lain karena bekerja secara *random*.



Gambar 4.8 Hasil Optimisasi PFD_{avg} saat LCC Minimum untuk Masing-Masing Iterasi



Gambar 4.9 Hasil Optimisasi STR saat LCC Minimum untuk Masing-Masing Iterasi



Gambar 4. 10 Hasil Optimisasi LCC Terbaik untuk Masing-Masing Iterasi

Gambar 4.4, 4.5 dan 4.6 menunjukkan grafik hasil optimisasi PF_{avg} , STR dan LCC saat LCC minimum untuk masing-masing iterasi. Dari ketiga gambar tersebut dapat dilihat bahwa terjadi penurunan nilai yang kemudian mulai konvergen. Nilai PF_{avg} pada iterasi pertama adalah sebesar $1,52 \times 10^{-2}$ yang kemudian mengalami penurunan hingga iterasi ketiga. Nilai PF_{avg} pada iterasi ketiga adalah sebesar $1,16 \times 10^{-2}$. Nilai tersebut tetap sampai iterasi terakhir. Nilai STR untuk iterasi pertama adalah sebesar $1,26 \times 10^{-5} /hour$, kemudian terjadi kenaikan nilai STR pada iterasi kedua sehingga didapatkan nilai STR sebesar $1,43 \times 10^{-5} /hour$. Namun kemudian terjadi penurunan pada iterasi ketiga sehingga nilai STR adalah sebesar $1,16 \times 10^{-5} /hour$. Nilai LCC terendah pada iterasi pertama adalah sebesar 274.583 USD yang kemudian menurun pada iterasi kedua sehingga nilai LCC adalah sebesar 230.995 USD. Nilai LCC kembali turun pada iterasi ketiga sehingga didapatkan LCC sebesar 227.816 USD yang kemudian nilai tersebut tetap hingga iterasi ke-28. Pada iterasi ke-30 nilai LCC turun menjadi 225.638 USD, kemudian nilai tersebut tetap hingga iterasi berakhir. Sehingga didapatkan nilai LCC terbaik adalah sebesar 225.638 USD. Kenaikan nilai STR

yang terjadi pada iterasi kedua dikarenakan nilai *shutdown loss* yang jauh lebih kecil dibandingkan nilai *catastrophic loss*, sehingga nilai STR tidak begitu berpengaruh terhadap nilai LCC. Pada iterasi kedua, tetap didapatkan nilai LCC terendah walaupun nilai STR mengalami kenaikan.

4.3 Analisa Hasil Optimisasi

Optimisasi dilakukan dengan tujuan untuk mendapatkan biaya paling efisien untuk pemasangan SIS pada *burner* dengan mempertimbangkan *dependability*. Fungsi objektif dari optimisasi yang telah dilakukan adalah LCC. Dari persamaan 2.7 dan 2.9 dapat dilihat bahwa untuk mendapatkan nilai LCC dipengaruhi oleh nilai PFD_{avg} dan STR, hal ini menunjukkan hubungan antara *cost* dan *dependability*. Tabel 4.1 menunjukkan hasil optimisasi LCC maksimum dan minimum dari 60 iterasi dengan 100 solusi dan Tabel 4.2 menunjukkan hasil variasi redundansi dan teknologi serta nilai DI saat LCC maksimum dan minimum. Sedangkan Tabel 4.3 menunjukkan hasil optimisasi pada beberapa solusi dengan *vote* arsitektur yang berbeda dan Tabel 4.4 menunjukkan hasil variasi redundansi dan teknologi serta nilai DI untuk masing-masing solusi yang ditunjukkan pada Tabel 4.3.

Dari hasil pada Tabel 4.1 dan 4.2 dapat diketahui bahwa kombinasi terbaik untuk desain SIS pada *burner* adalah dengan menggunakan arsitektur 1oo1 untuk masing – masing *sensing element* dan *final element*. Pemilihan redundansi 1oo1 untuk mendapatkan LCC optimal oleh *optimizer* dikarenakan biaya pemasangan komponen yang lebih murah, maka jika dipilih redundansi yang lebih tinggi akan menyebabkan adanya peningkatan biaya. Hal ini ditunjukkan pada Tabel 4.1 dan 4.2, nilai tertinggi LCC didapatkan ketika redundansi 4oo4 untuk *sensing element* dengan menggunakan 2 komponen untuk teknologi *smart* dan 2 komponen untuk teknologi *conventional* dan 1oo6 untuk *final element* dengan menggunakan 2 komponen dengan teknologi *air operated* dan 4 komponen dengan teknologi *hydraulic operated*. Dari Tabel 4.3 dan 4.4 dapat dilihat bahwa nilai LCC terkecil didapatkan saat PFD_{avg} terkecil. Nilai PFD_{avg}

terkecil didapatkan saat *vote* arsitektur adalah 1001 untuk kedua *sensing element* dan *final element*. Dari kedua tabel tersebut juga dapat dilihat bahwa dengan penggunaan komponen yang lebih banyak menghasilkan nilai PFD_{avg} yang semakin besar. Namun nilai PFD_{avg} yang lebih besar didapatkan pada solusi ke-31 dengan *vote* arsitektur 1001 untuk *sensing element* dan 1002 untuk *final element*. Hal ini disebabkan nilai laju kegagalan *danger* pada teknologi yang digunakan untuk *final element* lebih besar dibandingkan dengan nilai laju kegagalan teknologi yang digunakan *sensing element*. Sehingga nilai PFD_{avg} yang lebih besar didapatkan ketika terdapat redundansi pada *final element*. Pada Tabel 4.3 juga dapat dilihat bahwa PFD_{avg} akan semakin besar pada *vote* arsitektur 1001 untuk *sensing element* dan 1006 untuk *final element* sehingga LCC yang didapatkan juga semakin besar. Nilai STR terkecil pada Tabel 4.3 juga didapatkan saat *vote* arsitektur 1001 untuk kedua *sensing element* dan *final element*. Sedangkan nilai STR terbesar pada Tabel 4.3 didapatkan saat *vote* arsitektur 1006 untuk *sensing element* dan 1001 untuk *final element*. Dari tabel tersebut juga dapat diketahui bahwa nilai STR akan semakin besar apabila semakin banyak komponen yang digunakan dan variasi teknologi yang digunakan semakin banyak.

Dari data tersebut dapat dilihat bahwa penambahan redundansi mengakibatkan penambahan biaya dikarenakan semakin banyaknya jumlah komponen yang digunakan. Tabel 3.1 dan 3.2 menunjukkan biaya pembelian teknologi *smart transmitter* untuk *sensing element* dan teknologi *air operated* untuk *final element*. Dari kedua tabel tersebut dapat dilihat bahwa kedua teknologi yang dipilih memiliki biaya tertinggi jika dibandingkan dengan teknologi lainnya. Pemilihan kedua teknologi dengan biaya paling mahal tersebut dilakukan oleh *optimizer* dengan mempertimbangkan tingkat keamanan dan kerugian produksi yang diakibatkan dari kegagalan SIF. Namun LCC yang efisien tetap dapat dicapai dengan pemilihan redundansi yang tepat, yaitu 1001.

Melalui persamaan 2.1 dan 2.2 dapat dilihat bahwa keragaman redundansi dan pemilihan teknologi dapat

mempengaruhi nilai PFD_{avg} dan STR. Nilai PFD_{avg} merepresentasikan tingkat keamanan *plant* sedangkan STR merepresentasikan kerugian produksi yang diakibatkan *shutdown plant* mendadak tanpa adanya kegagalan. Jika dilihat dari Tabel 4.1, dapat diketahui bahwa nilai PFD_{avg} dan STR berbanding lurus dengan nilai LCC. Nilai PFD_{avg} dan STR yang rendah akan menghasilkan nilai LCC yang rendah begitupun sebaliknya. Dilihat pada persamaan 2.1, nilai PFD_{avg} dipengaruhi oleh nilai *danger failure rate* (λ_D), fraksi kegagalan *danger* (β_D) dan fraksi kegagalan *vote* (C_{Moon}). Nilai λ_D dan β_D terbagi menjadi dua kategori yaitu *detected* dan *undetected*. Pemilihan nilai C_{Moon} yang semakin rendah akan menghasilkan nilai PFD_{avg} yang semakin besar. Jika dilihat pada Tabel 4.2, solusi yang menghasilkan LCC terendah memiliki *vote* 1001. Pada Tabel 3.7 dapat dilihat nilai C_{Moon} untuk *vote* 1001 adalah 0. Namun dengan solusi ini didapatkan nilai PFD_{avg} terendah, hal ini dikarenakan nilai PFD_{avg} tidak hanya dipengaruhi oleh nilai C_{Moon} , namun juga dipengaruhi oleh nilai λ_D . Jika dilihat pada Tabel 3.1 dan 3.2, nilai λ_D untuk teknologi yang dipilih oleh *optimizer* memiliki nilai terendah. Pemilihan teknologi dengan nilai λ_D terendah menghasilkan nilai minimum untuk PFD_{avg} . Pada solusi dengan nilai LCC maksimum dapat dilihat bahwa dipilih *vote* 4004 untuk *sensing element* dan 1006 untuk *final element*. Jika dilihat pada Tabel 3.7, nilai C_{Moon} untuk *vote* 4004 adalah sebesar 0 dan untuk *vote* 1006 adalah sebesar 0,04. Hal tersebut dapat menyebabkan nilai PFD_{avg} yang rendah. Namun, nilai PFD_{avg} yang didapat untuk solusi tersebut bukanlah nilai terendah, hal ini dikarenakan teknologi yang digunakan. Pada solusi ini digunakan 4 *sensing element* yang terdiri dari 2 komponen dengan teknologi *smart transmitter* dan 2 dengan teknologi *conventional electronic transmitter* dan digunakan 6 *final element* yang terdiri dari 2 *air operated* dan 4 *hydraulic operated*. Jika dilihat pada Tabel 3.1 dapat dilihat bahwa nilai λ_D untuk *conventional electronic transmitter* lebih tinggi dibandingkan nilai λ_D untuk *smart transmitter*, sehingga dengan kombinasi ini menghasilkan nilai PFD_{avg} yang lebih tinggi.

Persamaan 2.2 menunjukkan bahwa nilai STR dipengaruhi oleh *safe failure rate* (λ_S), fraksi kegagalan *safe* (β_D) dan fraksi kegagalan *vote* (C_{Moon}). Nilai λ_S dan β_S terbagi menjadi dua kategori yaitu *detected* dan *undetected*. Nilai C_{Moon} dan λ_S memiliki pengaruh yang sama terhadap STR seperti pengaruh nilai C_{Moon} dan λ_D untuk nilai PFD_{avg} . Pada Tabel 4.1 dapat dilihat bahwa nilai STR yang didapat bukanlah nilai terendah untuk *vote* 1001. Namun tidak hanya dipengaruhi oleh *vote*, nilai STR juga dipengaruhi oleh nilai λ_S . Pada Tabel 4.2 dapat dilihat bahwa digunakan teknologi *smart transmitter* untuk *sensing element* dan *air operated* untuk *final element*. Jika dilihat pada Tabel 3.1 dan 3.2, nilai λ_S untuk *smart transmitter* merupakan nilai λ_S terendah dibandingkan teknologi yang lain. Sedangkan nilai λ_S untuk *air operated* bukanlah nilai λ_S terendah, sehingga nilai STR yang didapatkan bukanlah nilai terendah.

Nilai LCC dipengaruhi oleh nilai PFD_{avg} dan STR, hal ini dapat dilihat melalui persamaan 2.10 dan 2.12. Semakin rendah nilai PFD_{avg} dan STR maka akan didapatkan nilai LCC yang semakin rendah. Pada kedua solusi dapat dilihat pengaruh PFD_{avg} dan STR terhadap LCC. Pada solusi maksimum, nilai PFD_{avg} yang didapatkan merupakan nilai PFD_{avg} maksimum, sedangkan nilai STR yang didapatkan bukanlah nilai maksimum. Hal yang sama juga dapat dilihat pada solusi minimum, nilai PFD_{avg} yang didapatkan merupakan nilai PFD_{avg} minimum namun nilai STR yang didapatkan bukan nilai STR minimum. Hal ini dikarenakan terdapat faktor lain selain PFD_{avg} dan STR yang dapat mempengaruhi nilai LCC. Dilihat dari persamaan 2.3, nilai LCC dipengaruhi oleh nilai C_{risk} . Nilai C_{risk} dipengaruhi oleh nilai C_{hazard} dan C_{STR} . Nilai C_{hazard} didapatkan dengan mengalikan nilai PFD_{avg} dengan *catastrophic loss* dan nilai resiko *plant* tanpa pemasangan SIS. Jika dilihat pada Tabel 3.6, *catastrophic loss* adalah sebesar 150×10^6 USD. Sehingga dari persamaan 2.12 didapatkan *cost* untuk PFD_{avg} sebesar 137.488 USD untuk solusi maksimum dan 14.834 USD untuk solusi minimum. Nilai C_{STR} didapatkan dengan mengalikan nilai STR dengan C_{SD} . Nilai C_{SD} dipengaruhi oleh harga *shutdown loss*. Jika dilihat pada Tabel 3.4,

nilai *shutdown loss* adalah sebesar 5.161 USD. Dari persamaan 2.11 didapatkan *cost* untuk STR sebesar 211 USD untuk solusi maksimum dan 59 USD untuk solusi minimum. Jika dibandingkan harga *shutdown loss* dan *catastrophic loss*, maka dapat dilihat bahwa harga *catastrophic loss* jauh lebih tinggi. Hal ini mengakibatkan nilai C_{STR} tidak terlalu berpengaruh terhadap nilai LCC tidak seperti nilai C_{hazard} yang memiliki pengaruh besar terhadap nilai LCC. Sehingga dalam kasus ini dapat diketahui bahwa nilai STR tidak terlalu berpengaruh terhadap LCC jika dibandingkan dengan nilai PFD_{avg} . Oleh karena itu, LCC minimum tetap bisa didapatkan walaupun nilai STR bukanlah nilai terendah namun dibutuhkan nilai PFD_{avg} minimum untuk mendapatkan nilai LCC minimum.

Persamaan 2.3 menunjukkan bahwa nilai LCC juga dipengaruhi oleh harga – harga pembelian, operasi hingga *maintenance* instrumen. Dari hal tersebut dapat diketahui bahwa adanya penambahan jumlah komponen yang digunakan akan menyebabkan meningkatnya nilai LCC. Hal ini dipengaruhi oleh redundansi yang digunakan. Pada solusi maksimum, dapat dilihat bahwa digunakan total 3 komponen untuk *sensing element* dan 6 komponen untuk *final element*. Penggunaan banyak komponen ini bertujuan untuk meningkatkan tingkat keamanan dari *plant*. Namun tingkat keamanan yang baik tetap dapat didapatkan dengan penggunaan teknologi yang memiliki laju kegagalan yang rendah walaupun tanpa adanya komponen cadangan. Dari solusi minimum dapat dilihat bahwa hanya digunakan 1 komponen untuk masing – masing *sensing element* dan *final element* namun nilai *failure rate* untuk masing – masing teknologi merupakan nilai *failure rate* terendah.

Dari solusi – solusi yang telah didapatkan akan dilakukan analisa SIL dan DI. Pada Tabel 4.1 dapat dilihat bahwa terdapat 2 macam nilai SIL, yaitu SIL PFD_{avg} yang didapatkan melalui nilai PFD_{avg} berdasarkan IEC 61508 *part 1* dan SIL *final* yang didapatkan berdasarkan IE 61508 *part 2*. SIL *final* dipengaruhi oleh nilai *Hardware Fault Tolerance* (HFT), *Safe Failure Fraction* (SFF), dan tipe dari masing-masing teknologi yang

digunakan. Semakin tinggi nilai SIL maka menunjukkan bahwa *plant* tersebut semakin aman. Namun, pada hasil optimisasi yang didapat, nilai SIL adalah 1 baik untuk SIL PFD_{avg} maupun SIL *final*. Hal ini dikarenakan untuk mendapatkan LCC yang optimal dibutuhkan nilai PFD_{avg} terendah. Nilai SIL *final* pada kombinasi terbaik menurut *optimizer* adalah SIL 1. Dari seluruh hasil optimisasi, nilai SIL *final* tertinggi yang didapat adalah SIL 4, hanya saja untuk mendapatkan SIL terbaik tersebut membutuhkan biaya yang tinggi. Tujuan utama dilakukannya optimisasi ini adalah untuk mendapatkan nilai LCC terendah sehingga dipilih kombinasi yang menghasilkan SIL 1 agar tujuan tersebut dapat tercapai.

Perhitungan nilai DI dipengaruhi oleh redundansi dan jumlah jenis teknologi yang digunakan. Dari persamaan 2.11 dapat dilihat bahwa nilai DI dipengaruhi oleh jumlah subsistem, jumlah komponen, jumlah redundansi dan jumlah teknologi. Melalui hasil optimisasi, dapat diketahui bahwa nilai PFD_{avg} yang semakin besar akan menyebabkan meningkatnya nilai DI. Nilai DI menunjukkan hubungan antara redundansi dengan variasi jumlah komponen. Semakin tinggi nilai redundansi akan menghasilkan nilai DI yang lebih tinggi. Pada solusi maksimum dapat dilihat bahwa DI yang didapat adalah sebesar 0,5 sedangkan nilai DI yang didapat pada solusi minimum adalah 0. Pada solusi maksimum nilai 0,5 untuk DI didapatkan akibat adanya banyaknya jumlah komponen yang digunakan. Nilai DI tertinggi didapatkan pada *vote* 1006 yaitu sebesar 0,83. Nilai DI yang semakin tinggi didapatkan akibat redundansi yang digunakan adalah 1006 untuk *sensing element* dan *final element*. *Vote* 1006 menunjukkan bahwa apabila terjadi kegagalan pada salah 1 komponen maka *shutdown* akan segera terjadi. Hal ini menunjukkan tingkat keamanan yang lebih tinggi namun menghasilkan LCC yang lebih tinggi. Sehingga dipilih solusi dengan DI sebesar 0 karena tujuan dari optimisasi ini adalah untuk mendapatkan *cost* minimum.

BAB V

PENUTUP

5.1 Kesimpulan

Optimisasi dari LCC pada *burner* menggunakan GA dengan variabel optimisasi *vote* arsitektur dan teknologi untuk *sensing element* dan *final element*. Dari optimisasi yang telah dilakukan didapat kesimpulan sebagai berikut:

1. Hasil optimisasi terbaik pada rancangan SIS pada *burner* adalah nilai PFD_{avg} sebesar $1,16 \times 10^{-2}$, STR sebesar $1,16 \times 10^{-5}/hour$, dan LCC sebesar 225.638 USD.
2. Redundansi terbaik untuk mendapatkan LCC terendah adalah dengan *vote* arsitektur 1001 dengan teknologi *smart transmitter* untuk *sensing element* dan *air operated* untuk *final element*.
3. Nilai PFD_{avg} dan STR mempengaruhi nilai LCC sehingga harga – harga yang bersinggungan dengan PFD_{avg} dan STR dapat mempengaruhi hasil optimisasi. Pada Tugas Akhir ini, LCC minimum didapatkan ketika PFD_{avg} minimum dan STR yang relatif rendah namun tidak minimum. Sehingga pada kasus ini dapat diketahui bahwa nilai PFD_{avg} memiliki pengaruh lebih besar terhadap LCC dibandingkan dengan nilai STR.

5.2 Saran

Penulis memberikan saran untuk pengembangan penelitian lebih lanjut sebagai berikut:

1. Melakukan variasi dari nilai *Time Interval*.
2. Melakukan variasi pada teknologi dan jumlah komponen yang digunakan.

“Halaman ini memang dikosongkan”

DAFTAR PUSTAKA

- [1] Torres-Echeverria, A.C. 2009. *Modelling and Optimization of Safety Instrumented System based on Dependability and Cost Measure*. Department of Automatic Control and Systems Engineering, The University of Sheffield.
- [2] **IEC 61508. Functional safety of electrical/electronic/programmable electronic safety related systems.** Parts 1–7, 2nd ed. Geneva: International Electrotechnical Commission; 2010
- [3] Torres-Echeverria AC, Thompson HA. **Multi-objective genetic algorithm for optimization of system safety and reliability based on IEC 61508 requirements: a practical approach.** Proc IMechE PtO: J Risk Reliab 2007;221(O3):193–205.
- [4] Painton, L. Campbell, J. 1995. **Genetic Algorithms in Optimization of System Reliability.** IEEE Trans Reliab.
- [5] Coit D W. Smith A. 1996. **Reliability Optimization of Series-Parallel System Using Genetic Algorithm.** IEEE.
- [6] Busacca PG, Marseguerra M, Zio E. **Multiobjective optimization by genetic algorithms: application to safety systems.** Rel Eng Syst Saf 2001;72(1):59–74.
- [7] Salazar D, Rocco CM, Galva'n B J. **Optimization of constrained multiple-objective reliability problems using evolutionary algorithms.** Rel Eng Syst Saf 2006;91(9):1057–70.
- [8] Elegbede C, Adjallah K. **Availability allocation to repairable systems with genetic algorithms: a multi-objective formulation.** Rel Eng Syst Saf 2003;82(3):319–30.
- [9] Pattison RL, Andrews JD. **Genetic algorithms in optimal safety system design.** Proc IMechE E: J Proc Mech Eng 1999;213:187–97.
- [10] Hari Pradana, Septian. 2014. *Analisis Hazard and Operability (HAZOP) untuk Deteksi Bahaya dan Manajemen Risiko*

pada Unit Boiler (B-6203) di Pabrik III PT. Petrokimia Gresik. Surabaya: Teknik Fisik, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember.

- [11] J.C., Laprie. *Fundamental Concepts of Dependability*. Toulouse, France: LAAS-CNRS Report No. 01-145.
- [12] *Genetic Algorithm*.
<https://www.mathworks.com/discovery/genetic-algorithm.html> (diakses tanggal 23 Januari 2017)
- [13] Torres-Echeverria, A.C, Martorell, S, Thompson, H.A. 2012. **Multi-objective optimization of design and testing of safety instrumented systems with Moon voting architectures using a genetic algorithm**. Department of Automatic Control and System Engineering, The University of Sheffield: Elsevier.
- [14] Rausand, Marvin. Hoyland, Arnjlot. 2004. **System Reliability Theory**. United States of America: John Wiley & Sons, Inc. Second edition, pp 420-426
- [15] Torres-Echeverria, A. C., Martorell S. and Thompson H. A. 2009. **Design optimization of a safety-instrumented system based on RAMS+C addressing IEC 61508 requirements and diverse redundancy** [Journal]. - Sheffield : Elsevier-Reliability Engineering & Safety System
- [16] Biyanto TR. Fibrianto HY. Ramasamy M. 2016. **Thermal and Hydraulic Impacts Consideration in Refinery Crude Preheat Train Cleaning Scheduling Using Recent Stochastic Optimization Methods**. Scindirect.
- [17] Sallak M., Simon C., and Aubry J. **Optimal Design of Safety Instrumented Systems**, HAL (2006).
- [18] Goldberg D. E. 1989. **Genetic Algorithm in Search, Optimization & Machine Learning**. Addison-Wesley Publishing Company, Inc.
- [19] John Varghese, J. Prabhakaran, M. 2015. **Burner Management System for Dual Fired Boiler**. Sri Ramakrishna Engineering College, Coimbatore: IJIRT.

- ^[20] Torres-Echeverria, A.C, Martorell, S, Thompson, H.A. 2009. **Multiobjective optimization of strategies for operation and testing of low-demand safety instrumented systems using a genetic algorithm and fault trees.** Department of Automatic Control and System Engineering, The University of Sheffield. Elsevier.

BIODATA PENULIS



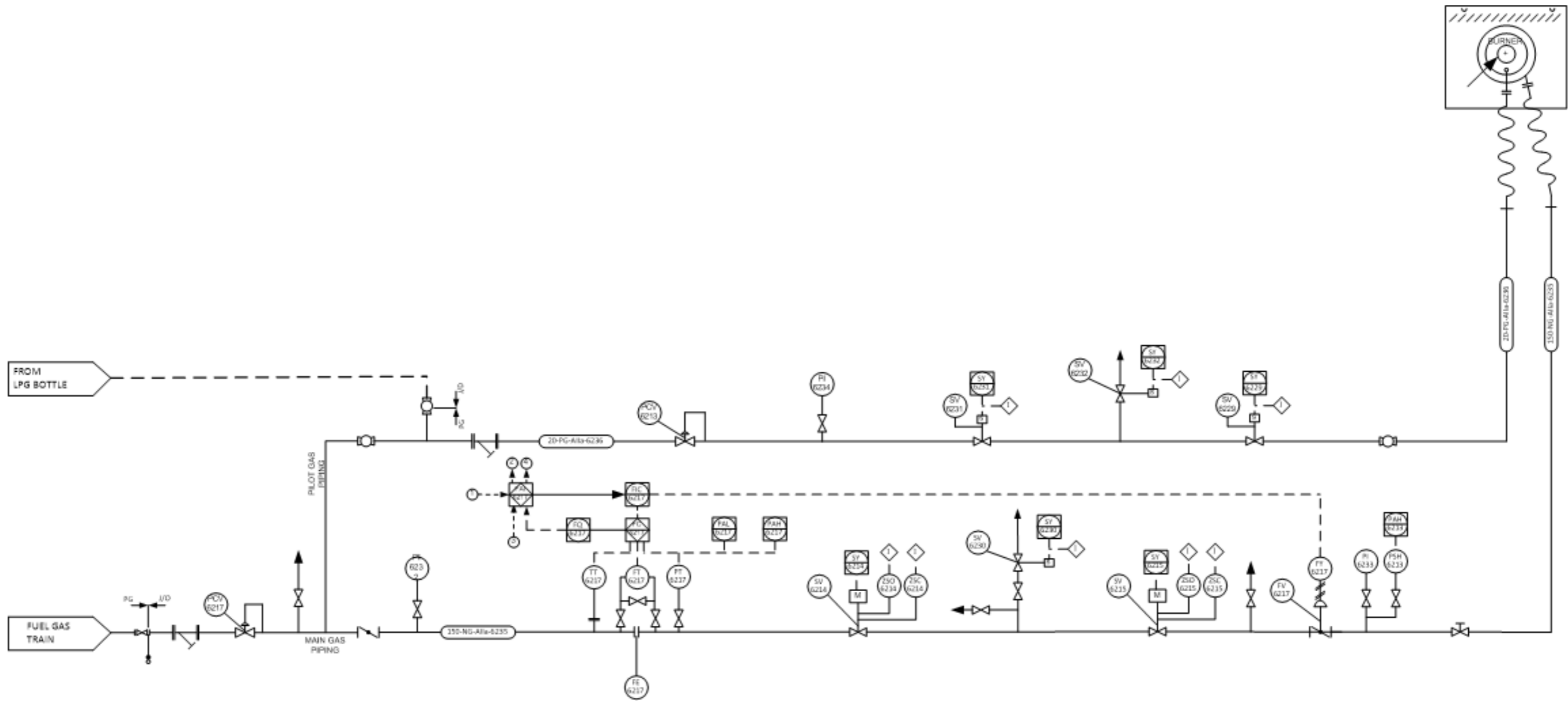
Nama penulis Nadya Feriza dilahirkan di Jakarta, tanggal 5 Mei 1995 dari bapak Faiq Bahfen dan Tuty Muliaty. Saat ini penulis tinggal di Jl Tebet Timur 1 H No 36 Jakarta Selatan. Penulis telah menyelesaikan pendidikan di SDN Menteng 01 Jakarta pada tahun 2007, pendidikan di SMPN 115 Jakarta pada tahun 2010, pendidikan di SMAN 26 Jakarta pada tahun 2013 dan sedang menempuh pendidikan S1 Teknik Fisika FTI di Institut Teknologi Sepuluh Nopember Surabaya hingga

sekarang.

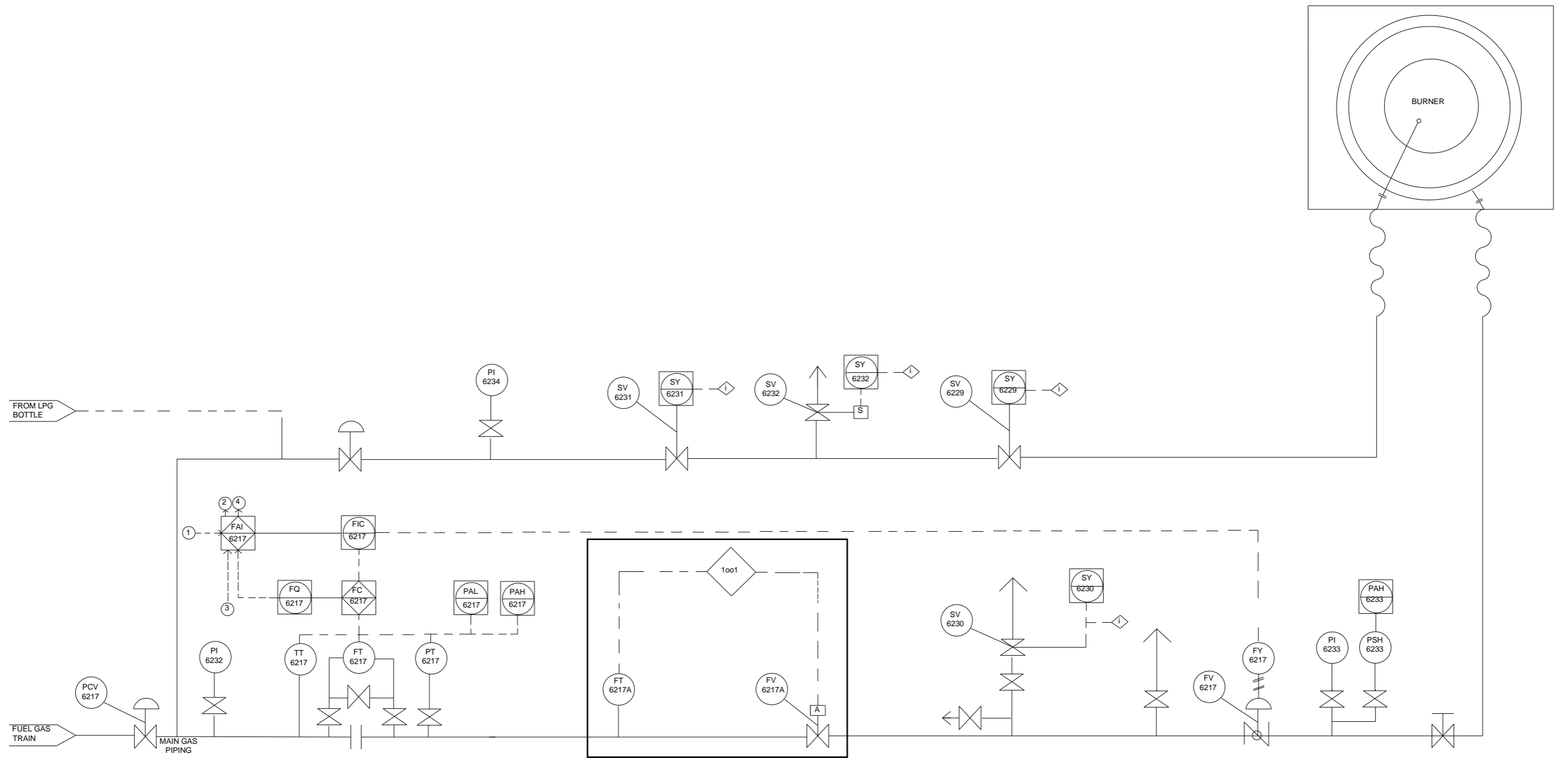
Pada bulan Juni 2017 penulis telah menyelesaikan Tugas Akhir dengan judul **Pemodelan dan Optimisasi *Safety Instrumented System* pada *Burner* berdasarkan *Dependability* dan *Cost Measure***. Bagi pembaca yang memiliki kritik, saran atau ingin berdiskusi lebih lanjut mengenai tugas akhir ini, maka dapat menghubungi penulis melalui *email*: nadyaferiza@gmail.com

LAMPIRAN

LAMPIRAN A. P&ID Burner



Gambar A.1 P&ID Burner Existing



Gambar A.2. P&ID *Burner* Rancangan

Lampiran B. Data HAZOP

Tabel B. 1 Data HAZOP

<i>Instrument</i>	<i>Guide Word</i>	<i>Deviation</i>	<i>Causes</i>	<i>Consequences</i>	<i>Safeguards</i>	<i>Recommendation</i>
<i>Flow Indicator (FI-6217) Fuel Inlet Burner</i>	<i>Less</i>	<i>Less Flow</i>	<ul style="list-style-type: none"> a. Proses mengalami penurunan, sehingga aliran <i>fuel</i> juga turun b. Buka <i>valve</i> tidak sesuai <i>set point</i> c. <i>Feedwater inlet burner</i> menurun intensitasnya 	<ul style="list-style-type: none"> a. Waktu pemanasan <i>feedwater</i> semakin lama b. Temperatur pemanasan tidak stabil c. Temperatur pemanasan tidak sesuai <i>set point</i> 	<i>Flow valve (FV-6217) Safety valve (SV-6214)</i>	<ul style="list-style-type: none"> a. Menjaga kestabilan bahan bakar yang masuk ke <i>burner</i> b. Memantau buka <i>valve</i> FV-6217 setiap jam c. Melakukan perawatan dan kalibrasi secara berkala setiap minimal 3 bulan pada saat non <i>overhaul</i> atau satu tahun sekali pada saat <i>overhaul</i> pada FV-6217
<i>Flow Indicating Controller (FIC-6219A) Feedwater Inlet Burner</i>	<i>Less</i>	<i>Less Flow</i>	<ul style="list-style-type: none"> a. Penurunan <i>rate</i> produksi b. Terjadinya kebocoran (<i>leak through</i>) pada jalur <i>feedwater</i> ke <i>burner</i> c. <i>Valve</i> tidak bias membuka sesuai <i>set point</i> 	<ul style="list-style-type: none"> a. Terjadi kebakaran jika <i>fuel</i> tidak ikut dikurangi b. <i>Steam</i> yang dihasilkan tidak maksimal c. Terjadi <i>explosion</i> apabila tekanan dalam <i>burner</i> meningkat 	<i>Flow Valve (FV-6212)</i>	<ul style="list-style-type: none"> a. Melakukan kalibrasi secara berkala pada FIC-6219A setiap 1 tahun sekali b. Melakukan <i>preventive maintenance</i> secara rutin setiap satu tahun sekali c. Melakukan kontrol rasio antara <i>fuel</i> dengan jumlah fluida yang dipanaskan, sehingga aliran <i>fuel</i> secara otomatis mengikuti intensitas fluida yang dipanaskan
	<i>More</i>	<i>More Flow</i>	<ul style="list-style-type: none"> a. <i>Rate</i> produksi meningkat b. <i>Valve</i> tidak dapat 	<ul style="list-style-type: none"> a. Waktu pemanasan semakin lama b. <i>Steam</i> yang 	<i>Flow Valve (FV-6212)</i>	<ul style="list-style-type: none"> a. Melakukan kalibrasi secara berkala pada FIC-6219A setiap 3 bulan sekali agar tidak

			menutup sesuai <i>set point</i> yang diinginkan	dihasilkan tidak mencapai <i>set point</i> temperatur yang diinginkan		mengganggu proses produksi saat terjadi kalibrasi b. Melakukan <i>preventive maintenance</i> secara rutin setiap satu tahun sekali (<i>overhaul</i>)
<i>Pressure Indicator (PI-6217) Fuel Inlet Burner</i>	<i>High</i>	<i>High Pressure</i>	a. <i>Volume fuel inlet burner</i> meningkat b. <i>Pressure valve (PV-6217)</i> tidak berfungsi secara normal	a. <i>No consequences</i> karena kejadian tidak sering terjadi	a. <i>Pressure Switch High (PSH-6217)</i> b. <i>Pressure Valve (PV-6217)</i> c. <i>Safety Valve (SV-6214)</i>	a. Melakukan kalibrasi secara berkala pada PI-6217 setiap 1 tahun sekali b. Melakukan <i>preventive maintenance</i> secara berkala setiap 1 tahun sekali c. Mengatur bukaan PV-6217 tidak terlalu besar
<i>Temperature Indicator (TI-6217) Fuel Inlet Burner</i>	<i>As Well As</i>	<i>As Well As Temperature</i>	<i>No cause</i>	<i>No Consequences</i>	<i>Safety Valve (SV-6214)</i>	a. Melakukan kalibrasi secara berkala pada TI-6217 setiap satu tahun sekali b. Melakukan <i>preventive maintenance</i> secara berkala setiap satu tahun sekali

Lampiran C. Data Kapasitas Produksi dan *Shutdown Time*

Tabel C. 1 Data Kapasitas Produksi dan *Shutdown Time*

Kapasitas Produksi	70 ton/hour
<i>Pressure Steam</i>	36 kg/cm ²
<i>Shutdown Time</i>	1 hour

Data di atas merupakan data yang didapatkan dari PT Petrokimia Gresik. Data di atas menunjukkan kapasitas per jam, tekanan *steam* dan *shutdown time* untuk *burner*. Dari data-data di atas akan dilakukan perhitungan *loss of production*.

Melalui Tabel A-3 Termodinamika Teknik yang ditulis oleh Michael J. Moran dan Howard N. Shapiro didapatkan nilai entalpi sebesar 2803 kJ/kg.

$$kWh = 70000kg \times 2803 kJ/kg \times 0,000278$$

$$kWh = 54546,38$$

Tarif tenaga listrik untuk bulan April-Juni 2017 menurut PLN adalah sebesar Rp 1.467,28.

$$loss\ of\ production = kWh \times \text{tarif tenaga listrik per kWh}$$

$$loss\ of\ production = 54546,38 \times Rp\ 1.467,28$$

$$loss\ of\ production = Rp\ 80.034.812,45$$

$$loss\ of\ production = US\$ 6019.02$$

Lampiran D. *Objective Function*

Function

```
[LCC, PFDavg, STRavg, Cx, Nx, tekno1, tekno2, tekno3  
, tekno4, tekno5, tekno6, teknoa1, teknoa2, teknoa3  
, teknoa4, teknoa5, teknoa6, i1, i2]=objfun_nadyan  
nnn(x)
```

```
tekno1=x(1);
```

```
tekno2=x(2);
```

```
tekno3=x(3);
```

```
tekno4=x(4);
```

```
tekno5=x(5);
```

```
tekno6=x(6);
```

```
teknoa1=x(7);
```

```
teknoa2=x(8);
```

```
teknoa3=x(9);
```

```
teknoa4=x(10);
```

```
teknoa5=x(11);
```

```
teknoa6=x(12);
```

```
i1=x(13);
```

```
i2=x(14);
```

```
Tr = 8;
```

```
Ti = 8760;
```

```
SDtime=1;
```

```
SDloss=5161;
```

```
R=0.05;
```

```
YL=15;
```

```
Cacc = 150*10^6;
```

```
Facc = 8.55*10^-3;
```

```
Cd = 3060;
```

```
Cinscomm = 600;
```

```
CdPLC = 10320;
```

```
CpmPLC = 960;
```

```
CtPLC = 240;
```

```
CcmPLC = 8000;
```

```
Bddc = 0.1
```

```
Bduc = 0.1
```

```
Bsdc = 0.1
```

```
Bsuc = 0.1  
Xtc = 3.492*10^-6  
Xsdc = 3.46*10^-6  
Xsuc = 0*10^-6  
Cpurchasec =40000;
```

```
if i1==1  
Cx= 0  
Nx= 1  
elseif i1==2  
Cx= 1  
Nx= 2  
elseif i1==3  
Cx= 0.3  
Nx= 3  
elseif i1==4  
Cx= 0.15  
Nx= 4  
elseif i1==5  
Cx= 0.08  
Nx= 5  
elseif i1==6  
Cx= 0.04  
Nx= 6  
elseif i1==7  
Cx= 0  
Nx= 2  
elseif i1==8  
Cx= 2.4  
Nx= 3  
elseif i1==9  
Cx= 0.75  
Nx= 4  
elseif i1==10  
Cx= 0.45  
Nx= 5  
elseif i1==11  
Cx= 0.26  
Nx= 6
```

```
elseif i1==12
Cx= 0
Nx= 3
elseif i1==13
Cx= 4
Nx= 4
elseif i1==14
Cx= 1.2
Nx= 5
elseif i1==15
Cx= 0.8
Nx= 6
elseif i1==16
Cx= 0
Nx= 4
elseif i1==17
Cx= 6.0
Nx= 5
elseif i1==18
Cx= 1.6
Nx= 6
elseif i1==19
Cx= 0
Nx= 5
elseif i1==20
Cx= 8.1
Nx= 6
elseif i1==21
Cx= 0
Nx= 6
end

if i2==1
Cx= 0
Nx= 1
elseif i1==2
Cx= 1
Nx= 2
elseif i1==3
```

```
Cx= 0.3
Nx= 3
elseif i1==4
Cx= 0.15
Nx= 4
elseif i1==5
Cx= 0.08
Nx= 5
elseif i1==6
Cx= 0.04
Nx= 6
elseif i1==7
Cx= 0
Nx= 2
elseif i1==8
Cx= 2.4
Nx= 3
elseif i1==9
Cx= 0.75
Nx= 4
elseif i1==10
Cx= 0.45
Nx= 5
elseif i1==11
Cx= 0.26
Nx= 6
elseif i1==12
Cx= 0
Nx= 3
elseif i1==13
Cx= 4
Nx= 4
elseif i1==14
Cx= 1.2
Nx= 5
elseif i1==15
Cx= 0.8
Nx= 6
elseif i1==16
```

```
Cx= 0
Nx= 4
elseif i1==17
Cx= 6.0
Nx= 5
elseif i1==18
Cx= 1.6
Nx= 6
elseif i1==19
Cx= 0
Nx= 5
elseif i1==20
Cx= 8.1
Nx= 6
elseif i1==21
Cx= 0
Nx= 6
end
```

```
if teknol==1
Xdda1 = 0.048*10^-6      Xdua1 = 0.103*10^-6
%(PT A)
```

```
Xsda1 = 0.265*10^-6%(FT A)
Xsua1 = 0.118*10^-6%(FT A)
Xta1  = 0.534*10^-6%(FT A)
```

```
Xtc1  = 3.492*10^-6%(PLC)
Xsdc1 = 3.46*10^-6 %(PLC)
Xsuc1 = 0*10^-6;    %(PLC)
```

```
Bdda1 = 0.1 %(FT A)
Bdua1 = 0.1 %(FT A)
Bsua1 = 0.1 %(FT A)
Bsda1 = 0.05%(FT A)
```

```
Bddc1 = 0.1
Bduc1 = 0.1
Bsdcl = 0.1
Bsuc1 = 0.1
```

```
Cpurchaseal = 4844;  
Ccma = Xta1*480*Nx;  
Cproa = [Cpurchaseal+Cinscomm]*Nx;
```

```
elseif teknol==2
```

```
Xdda1 = 0.97*10^-6  
Xdual = 0.93*10^-6  
Xsda1 = 1.21*10^-6  
Xsual = 0.95*10^-6  
Xta1 = 4.06*10^-6
```

```
Xtc1 = 3.492*10^-6  
Xsdc1 = 3.46*10^-6  
Xsuc1 = 0*10^-6
```

```
Bdda1 = 0.1 % (FT B)  
Bdual = 0.1 % (FT B)  
Bsual = 0.1 % (FT B)  
Bsda1 = 0.1; % (FT B)
```

```
Bddc1 = 0.1  
Bduc1 = 0.1  
Bsdcl = 0.1  
Bsuc1 = 0.1
```

```
Cpurchaseal = 3206;  
Ccma = Xta1*480*Nx;  
Cproa = [Cpurchaseal+Cinscomm]*Nx;
```

```
elseif teknol==3
```

```
Xdda1 = 0.41*10^-6  
Xdual = 3.70*10^-6  
Xsda1 = 0.68*10^-6  
Xsual = 6.13*10^-6  
Xta1 = 10.92*10^-6
```

```
Xtc1 = 3.492*10^-6
```

```
Xsdc1 = 3.46*10^-6  
Xsuc1 = 0*10^-6
```

```
Bsda1 = 0.1% (FT C)  
Bdda1 = 0.1% (FT C)  
Bdua1 = 0.1% (FT C)  
Bsua1 = 0.1% (FT C)  
Bsda1 = 0.1% (FT C)
```

```
Bddc1 = 0.1  
Bduc1 = 0.1  
Bsdcl = 0.1  
Bsuc1 = 0.1
```

```
Cpurchaseal = 500;  
Ccma = Xta1*480*Nx;  
Cproa = [Cpurchaseal+Cinscomm]*Nx;  
end
```

```
if tekno2==1  
Xdda2 = 0.048*10^-6  
Xdua2 = 0.103*10^-6% (FT A)  
Xsda2 = 0.265*10^-6% (FT A)  
Xsua2 = 0.118*10^-6% (FT A)  
Xta2 = 0.534*10^-6% (FT A)
```

```
Xtc2 = 3.492*10^-6% (PLC)  
Xsdc2 = 3.46*10^-6 % (PLC)  
Xsuc2 = 0*10^-6; % (PLC)
```

```
Bdda2 = 0.1 % (FT A)  
Bdua2 = 0.1 % (FT A)  
Bsua2 = 0.1 % (FT A)  
Bsda2 = 0.05% (FT A)
```

```
Bddc2 = 0.1  
Bduc2 = 0.1  
Bsdcl = 0.1
```



```

Bsuc2 = 0.1

Cpurchasea2 = 4844;
Ccma = Xta2*480*Nx;
Cproa = [Cpurchasea2+Cinscomm]*Nx;

elseif tekno2==2
Xdda2 = 0.97*10^-6
Xdua2 = 0.93*10^-6
Xsda2 = 1.21*10^-6
Xsua2 = 0.95*10^-6
Xta2  = 4.06*10^-6

Xtc2  = 3.492*10^-6
Xsdc2 = 3.46*10^-6
Xsuc2 = 0*10^-6

Bdda2 = 0.1 %(FT B)
Bdua2 = 0.1 %(FT B)
Bsua2 = 0.1 %(FT B)
Bsda2 = 0.1;%(FT B)

Bddc2 = 0.1
Bduc2 = 0.1
Bsdc2 = 0.1
Bsuc2 = 0.1

Cpurchasea2 = 3206;
Ccma = Xta2*480*Nx;
Cproa = [Cpurchasea2+Cinscomm]*Nx;

elseif tekno2==3
Xdda2 = 0.41*10^-6
Xdua2 = 3.70*10^-6
Xsda2 = 0.68*10^-6
Xsua2 = 6.13*10^-6
Xta2  = 10.92*10^-6

```

```
Xtc2 = 3.492*10^-6
Xsdc2 = 3.46*10^-6
Xsuc2 = 0*10^-6
```

```
Bsda2 = 0.1% (FT C)
Bdda2 = 0.1% (FT C)
Bdua2 = 0.1% (FT C)
Bsua2 = 0.1% (FT C)
Bsda2 = 0.1% (FT C)
```

```
Bddc2 = 0.1
Bduc2 = 0.1
Bsd2 = 0.1
Bsuc2 = 0.1
```

```
Cpurchasea2 = 500;
Ccma = Xta2*480*Nx;
Cproa = [Cpurchasea2+Cinscomm]*Nx
end
```

```
if tekno3==1
Xdda3 = 0.048*10^-6
Xdua3 = 0.93*10^-6
Xsda3 = 0.265*10^-6% (FT A)
Xsua3 = 0.118*10^-6% (FT A)
Xta3 = 0.534*10^-6% (FT A)
```

```
Xtc3 = 3.492*10^-6% (PLC)
Xsdc3 = 3.46*10^-6 % (PLC)
Xsuc3 = 0*10^-6; % (PLC)
```

```
Bdda3 = 0.1 % (PT A)
Bdua3 = 0.1 % (PT A)
Bsua3 = 0.1 % (PT A)
Bsda3 = 0.05% (PT A)
```

```
Bddc3 = 0.1
Bduc3 = 0.1
```

```

Bsd3 = 0.1
Bsuc3 = 0.1

Cpurchasea3 = 4844;
Ccma = Xta3*480*Nx;
Cproa = [Cpurchasea3+Cinscomm]*Nx;

elseif tekno3==2
Xdda3 = 0.97*10^-6
Xdua3 = 0.93*10^-6
Xsda3 = 1.21*10^-6
Xsua3 = 0.95*10^-6
Xta3 = 4.06*10^-6

Xtc3 = 3.492*10^-6
Xsdc3 = 3.46*10^-6
Xsuc3 = 0*10^-6

Bdda3 = 0.1 %(FT B)
Bdua3 = 0.1 %(FT B)
Bsua3 = 0.1 %(FT B)
Bsda3 = 0.1;%(FT B)

Bddc3 = 0.1
Bduc3 = 0.1
Bsd3 = 0.1
Bsuc3 = 0.1

Cpurchasea3 = 3206;
Ccma = Xta3*480*Nx;
Cproa = [Cpurchasea3+Cinscomm]*Nx;

elseif tekno3==3
Xdda3 = 0.41*10^-6
Xdua3 = 3.70*10^-6
Xsda3 = 0.68*10^-6
Xsua3 = 6.13*10^-6
Xta3 = 10.92*10^-6

```

```

Xtc3 = 3.492*10^-6
Xsdc3 = 3.46*10^-6
Xsuc3 = 0*10^-6

Bsda3 = 0.1% (FT C)
Bdda3 = 0.1% (FT C)
Bdua3 = 0.1% (FT C)
Bsua3 = 0.1% (FT C)
Bsda3 = 0.1% (FT C)

Bddc3 = 0.1
Bduc3 = 0.1
Bsd3 = 0.1
Bsuc3 = 0.1

Cpurchasea3 = 500;
Ccma = Xta3*480*Nx;
Cproa = [Cpurchasea3+Cinscomm]*Nx;
end

if tekno4==1
Xdda4 = 0.048*10^-6
Xdua4 = 0.103*10^-6% (FT A)
Xsda4 = 0.265*10^-6% (FT A)
Xsua4 = 0.118*10^-6% (FT A)
Xta4 = 0.534*10^-6% (FT A)

Xtc4 = 3.492*10^-6% (PLC)
Xsdc4 = 3.46*10^-6 % (PLC)
Xsuc4 = 0*10^-6; % (PLC)

Bdda4 = 0.1 % (FT A)
Bdua4 = 0.1 % (FT A)
Bsua4 = 0.1 % (FT A)
Bsda4 = 0.05% (FT A)

Bddc4 = 0.1

```

```
Bduc4 = 0.1  
Bsd4 = 0.1  
Bsuc4 = 0.1
```

```
Cpurchasea4 = 4844;  
Ccma = Xta4*480*Nx;  
Cproa = [Cpurchasea4+Cinscomm]*Nx;
```

```
elseif tekno4==2
```

```
Xdda4 = 0.97*10^-6  
Xdua4 = 0.93*10^-6  
Xsda4 = 1.21*10^-6  
Xsua4 = 0.95*10^-6  
Xta4 = 4.06*10^-6
```

```
Xtc4 = 3.492*10^-6  
Xsdc4 = 3.46*10^-6  
Xsuc4 = 0*10^-6
```

```
Bdda4 = 0.1 %(FT B)  
Bdua4 = 0.1 %(FT B)  
Bsua4 = 0.1 %(FT B)  
Bsda4 = 0.1;%(FT B)
```

```
Bddc4 = 0.1  
Bduc4 = 0.1  
Bsd4 = 0.1  
Bsuc4 = 0.1
```

```
Cpurchasea4 = 3206;  
Ccma = Xta4*480*Nx;  
Cproa = [Cpurchasea4+Cinscomm]*Nx;
```

```
elseif tekno4==3
```

```
Xdda4 = 0.41*10^-6  
Xdua4 = 3.70*10^-6  
Xsda4 = 0.68*10^-6  
Xsua4 = 6.13*10^-6
```

```

Xta4 = 10.92*10^-6

Xtc4 = 3.492*10^-6
Xsdc4 = 3.46*10^-6
Xsuc4 = 0*10^-6

Bsda4 = 0.1%(FT C)
Bdda4 = 0.1%(FT C)
Bdua4 = 0.1%(FT C)
Bsua4 = 0.1%(FT C)
Bsda4 = 0.1%(FT C)

Bddc4 = 0.1
Bduc4 = 0.1
Bsd4 = 0.1
Bsuc4 = 0.1

Cpurchasea4 = 500;
Ccma = Xta4*480*Nx;
Cproa = [Cpurchasea4+Cinscomm]*Nx;
end

if tekno5==1
Xdda5 = 0.048*10^-6
Xdua5 = 0.103*10^-6%(FT A)
Xsda5 = 0.265*10^-6%(FT A)
Xsua5 = 0.118*10^-6%(FT A)
Xta5 = 0.534*10^-6%(FT A)

Xtc5 = 3.492*10^-6%(PLC)
Xsdc5 = 3.46*10^-6 %(PLC)
Xsuc5 = 0*10^-6; %(PLC)

Bdda5 = 0.1 %(FT A)
Bdua5 = 0.1 %(FT A)
Bsua5 = 0.1 %(FT A)
Bsda5 = 0.05%(FT A)

```

```
Bddc5 = 0.1  
Bduc5 = 0.1  
Bsd5 = 0.1  
Bsuc5 = 0.1
```

```
Cpurchasea5 = 4844;  
Ccma = Xta5*480*Nx;  
Cproa = [Cpurchasea5+Cinscomm]*Nx;
```

```
elseif tekno5==2  
Xdda5 = 0.97*10^-6  
Xdua5 = 0.93*10^-6  
Xsda5 = 1.21*10^-6  
Xsua5 = 0.95*10^-6  
Xta5 = 4.06*10^-6
```

```
Xtc5 = 3.492*10^-6  
Xsdc5 = 3.46*10^-6  
Xsuc5 = 0*10^-6
```

```
Bdda5 = 0.1 % (FT B)  
Bdua5 = 0.1 % (FT B)  
Bsua5 = 0.1 % (FT B)  
Bsda5 = 0.1; % (FT B)
```

```
Bddc5 = 0.1  
Bduc5 = 0.1  
Bsd5 = 0.1  
Bsuc5 = 0.1
```

```
Cpurchasea5 = 3206;  
Ccma = Xta5*480*Nx;  
Cproa = [Cpurchasea5+Cinscomm]*Nx;
```

```
elseif tekno5==3  
Xdda5 = 0.41*10^-6  
Xdua5 = 3.70*10^-6
```

```
Xsda5 = 0.68*10^-6  
Xsua5 = 6.13*10^-6  
Xta5 = 10.92*10^-6
```

```
Xtc5 = 3.492*10^-6  
Xsdc5 = 3.46*10^-6  
Xsuc5 = 0*10^-6
```

```
Bsda5 = 0.1% (FT C)  
Bdda5 = 0.1% (FT C)  
Bdua5 = 0.1% (FT C)  
Bsua5 = 0.1% (FT C)  
Bsda5 = 0.1% (FT C)
```

```
Bddc5 = 0.1  
Bduc5 = 0.1  
Bsdc5 = 0.1  
Bsuc5 = 0.1
```

```
Cpurchasea5 = 500;  
Ccma = Xta5*480*Nx;  
Cproa = [Cpurchasea5+Cinscomm]*Nx;  
end
```

```
if tekno6==1  
Xdda6 = 0.048*10^-6  
Xdua6 = 0.103*10^-6% (FT A)  
Xsda6 = 0.265*10^-6% (FT A)  
Xsua6 = 0.118*10^-6% (FT A)  
Xta6 = 0.534*10^-6% (FT A)
```

```
Xtc6 = 3.492*10^-6% (PLC)  
Xsdc6 = 3.46*10^-6 % (PLC)  
Xsuc6 = 0*10^-6; % (PLC)
```

```
Bdda6 = 0.1 % (FT A)  
Bdua6 = 0.1 % (FT A)  
Bsua6 = 0.1 % (FT A)
```



```

Bsda6 = 0.05%(FT A)

Bddc6 = 0.1
Bduc6 = 0.1
Bsd6 = 0.1
Bsuc6 = 0.1

Cpurchasea6 = 4844;
Ccma = Xta6*480*Nx;
Cproa = [Cpurchasea6+Cinscomm]*Nx;

elseif tekno6==2
Xdda6 = 0.97*10^-6
Xdua6 = 0.93*10^-6
Xsda6 = 1.21*10^-6
Xsua6 = 0.95*10^-6
Xta6 = 4.06*10^-6

Xtc6 = 3.492*10^-6
Xsdc6 = 3.46*10^-6
Xsuc6 = 0*10^-6

Bdda6 = 0.1 %(FT B)
Bdua6 = 0.1 %(FT B)
Bsua6 = 0.1 %(FT B)
Bsda6 = 0.1;%(FT B)

Bddc6 = 0.1
Bduc6 = 0.1
Bsd6 = 0.1
Bsuc6 = 0.1

Cpurchasea6 = 3206;
Ccma = Xta6*480*Nx;
Cproa = [Cpurchasea6+Cinscomm]*Nx;

elseif tekno6==3
Xdda6 = 0.41*10^-6

```

```
Xdua6 = 3.70*10^-6  
Xsda6 = 0.68*10^-6  
Xsua6 = 6.13*10^-6  
Xta6 = 10.92*10^-6
```

```
Xtc6 = 3.492*10^-6  
Xsdc6 = 3.46*10^-6  
Xsuc6 = 0*10^-6
```

```
Bsda6 = 0.1% (FT C)  
Bdda6 = 0.1% (FT C)  
Bdua6 = 0.1% (FT C)  
Bsua6 = 0.1% (FT C)  
Bsda6 = 0.1% (FT C)
```

```
Bddc6 = 0.1  
Bduc6 = 0.1  
Bsd6 = 0.1  
Bsuc6 = 0.1
```

```
Cpurchasea6 = 500;  
Ccma = Xta6*480*Nx;  
Cproa = [Cpurchasea6+Cinscomm]*Nx;  
end
```

```
if teknoal==1  
Xddb1 = 0.084*10^-6  
Xdub1 = 2.51*10^-6 % (FE A)  
Xsdb1 = 0 % (FE A)  
Xsub1 = 3.94*10^-6 % (FE A)  
Xtb1 = 7.29*10^-6 % (FE A)
```

```
Xtc1 = 3.492*10^-6% (PLC)  
Xsdc1 = 3.46*10^-6 % (PLC)  
Xsuc1 = 0*10^-6; % (PLC)
```

```
Bddb1 = 0.1% (FE A)  
Bdub1 = 0.1% (FE A)
```

```

Bsub1 = 0.1%(FE A)
Bsdb1 = 0.1%(FE A)

Bddc1 = 0.1
Bduc1 = 0.1
Bsdcl = 0.1
Bsuc1 = 0.1

Cpurchaseb1 = 6940;
Ccma = Xtb1*480*Nx;
Cprob = [Cpurchaseb1+Cinscomm]*Nx;

elseif teknoal==2
Xddb1 = 1.09*10^-6
Xdub1 = 4.35*10^-6
Xsdb1 = 0
Xsub1 = 3.17*10^-6
Xtb1 = 8.61*10^-6

Xtc1 = 3.492*10^-6
Xsdcl = 3.46*10^-6
Xsuc1 = 0*10^-6

Bddb1 = 0.1%(FE B)
Bdub1 = 0.1%(FE B)
Bsub1 = 0.1%(FE B)
Bsdb1 = 0.1%(FE B)

Bddc1 = 0.1
Bduc1 = 0.1
Bsdcl = 0.1
Bsuc1 = 0.1

Cpurchaseb1 = 6400;
Ccmb = Xtb1*480*Nx;
Cprob = [Cpurchaseb1+Cinscomm]*Nx;

elseif teknoal==3

```

```

Xddb1 = 0.79*10^-6
Xdub1 = 7.11*10^-6
Xsdb1 = 0
Xsub1 = 9.17*10^-6
Xtb1  = 17.07*10^-6

Xtc1  = 3.492*10^-6
Xsdc1 = 3.46*10^-6
Xsuc1 = 0*10^-6

Bsdb1 = 0.1 % (FE C)
Bddb1 = 0.1 % (FE C)
Bdub1 = 0.1 % (FE C)
Bsub1 = 0.1 % (FE C)
Bsdb  = 0.05% (FE C)

Bddc1 = 0.1
Bduc1 = 0.1
Bsdcl = 0.1
Bsuc1 = 0.1

Cpurchaseb1 = 6200;
Ccmb = Xtb1*480*Nx;
Cprob = [Cpurchaseb1+Cinscomm]*Nx;
end

if teknoa2==1
Xddb2 = 0.084*10^-6
Xdub2 = 2.51*10^-6% (FE A)
Xsdb2 = 0 % (FE A)
Xsub2 = 3.94*10^-6% (FE A)
Xtb2  = 7.29*10^-6% (FE A)

Xtc2  = 3.492*10^-6% (PLC)
Xsdc2 = 3.46*10^-6 % (PLC)
Xsuc2 = 0*10^-6; % (PLC)

Bddb2 = 0.1% (FE A)

```

```
Bdub2 = 0.1%(FE A)
Bsub2 = 0.1%(FE A)
Bsdb2 = 0.1%(FE A)

Bddc2 = 0.1
Bduc2 = 0.1
Bsdc2 = 0.1
Bsuc2 = 0.1

Cpurchaseb2 = 6940;
Ccmb = Xtb2*480*Nx;
Cprob = [Cpurchaseb2+Cinscomm]*Nx;

elseif teknoa2==2
Xddb2 = 1.09*10^-6
Xdub2 = 4.35*10^-6
Xsdb2 = 0
Xsub2 = 3.17*10^-6
Xtb2 = 8.61*10^-6

Xtc2 = 3.492*10^-6
Xsdc2 = 3.46*10^-6
Xsuc2 = 0*10^-6

Bddb2 = 0.1%(FE B)
Bdub2 = 0.1%(FE B)
Bsub2 = 0.1%(FE B)
Bsdb2 = 0.1%(FE B)

Bddc2 = 0.1
Bduc2 = 0.1
Bsdc2 = 0.1
Bsuc2 = 0.1

Cpurchaseb2 = 6400;
Ccmb = Xtb2*480*Nx;
Cprob = [Cpurchaseb2+Cinscomm]*Nx;
```

```

elseif teknoa2==3
Xddb2 = 0.79*10^-6
Xdub2 = 7.11*10^-6
Xsdb2 = 0
Xsub2 = 9.17*10^-6
Xtb2  = 17.07*10^-6

Xtc2  = 3.492*10^-6
Xsdc2 = 3.46*10^-6
Xsuc2 = 0*10^-6

Bsdb2 = 0.1 %(FE C)
Bddb2 = 0.1 %(FE C)
Bdub2 = 0.1 %(FE C)
Bsub2 = 0.1 %(FE C)
Bsdb2 = 0.05%(FE C)

Bddc2 = 0.1
Bduc2 = 0.1
Bsd2  = 0.1
Bsuc2 = 0.1

Cpurchaseb2 = 6200;
Ccmb = Xtb2*480*Nx;
Cprob = [Cpurchaseb2+Cinscomm]*Nx;
end

if teknoa3==1
Xddb3 = 0.084*10^-6
Xdub3 = 2.51*10^-6%(FE A)
Xsdb3 = 0           %(FE A)
Xsub3 = 3.94*10^-6%(FE A)
Xtb3  = 7.29*10^-6%(FE A)

Xtc3 = 3.492*10^-6%(PLC)
Xsdc3 = 3.46*10^-6%(PLC)
Xsuc3 = 0*10^-6;   %(PLC)

```

```
Bddb3 = 0.1% (FE A)
Bdub3 = 0.1% (FE A)
Bsub3 = 0.1% (FE A)
Bsdb3 = 0.1% (FE A)
```

```
Bddc3 = 0.1
Bduc3 = 0.1
Bsdc3 = 0.1
Bsuc3 = 0.1
```

```
Cpurchaseb3 = 6940;
Ccmb = Xtb3*480*Nx;
Cprob = [Cpurchaseb3+Cinscomm]*Nx;
```

```
elseif teknoa3==2
Xddb3 = 1.09*10^-6
Xdub3 = 4.35*10^-6
Xsdb3 = 0
Xsub3 = 3.17*10^-6
Xtb3 = 8.61*10^-6
```

```
Xtc3 = 3.492*10^-6
Xsdc3 = 3.46*10^-6
Xsuc3 = 0*10^-6
```

```
Bddb3 = 0.1% (FE B)
Bdub3 = 0.1% (FE B)
Bsub3 = 0.1% (FE B)
Bsdb3 = 0.1% (FE B)
```

```
Bddc3 = 0.1
Bduc3 = 0.1
Bsdc3 = 0.1
Bsuc3 = 0.1
```

```
Cpurchaseb3 = 6400;
Ccmb = Xtb3*480*Nx;
Cprob = [Cpurchaseb3+Cinscomm]*Nx;
```

```

elseif teknoa3==3
Xddb3 = 0.79*10^-6
Xdub3 = 7.11*10^-6
Xsdb3 = 0
Xsub3 = 9.17*10^-6
Xtb3 = 17.07*10^-6

Xtc3 = 3.492*10^-6
Xsdc3 = 3.46*10^-6
Xsuc3 = 0*10^-6

Bsdb3 = 0.1 %(FE C)
Bddb3 = 0.1 %(FE C)
Bdub3 = 0.1 %(FE C)
Bsub3 = 0.1 %(FE C)
Bsdb3 = 0.05%(FE C)

Bddc3 = 0.1
Bduc3 = 0.1
Bsdc3 = 0.1
Bsuc3 = 0.1

Cpurchaseb3 = 6200;
Ccmb = Xtb3*480*Nx;
Cprob = [Cpurchaseb3+Cinscomm]*Nx;
end

if teknoa4==1
Xddb4 = 0.084*10^-6
Xdub4 = 2.51*10^-6 %(FE A)
Xsdb4 = 0 %(FE A)
Xsub4 = 3.94*10^-6 %(FE A)
Xtb4 = 7.29*10^-6 %(FE A)

Xtc4 = 3.492*10^-6%(PLC)
Xsdc4 = 3.46*10^-6 %(PLC)
Xsuc4 = 0*10^-6; %(PLC)

```



```
Bddb4 = 0.1% (FE A)
Bdub4 = 0.1% (FE A)
Bsub4 = 0.1% (FE A)
Bsdb4 = 0.1% (FE A)

Bddc4 = 0.1
Bduc4 = 0.1
Bsd4 = 0.1
Bsuc4 = 0.1

Cpurchaseb4 = 6940;
Ccmb = Xtb4*480*Nx;
Cprob = [Cpurchaseb4+Cinscomm]*Nx;

elseif teknoa4==2
Xddb4 = 1.09*10^-6
Xdub4 = 4.35*10^-6
Xsdb4 = 0
Xsub4 = 3.17*10^-6
Xtb4 = 8.61*10^-6

Xtc4 = 3.492*10^-6
Xsdc4 = 3.46*10^-6
Xsuc4 = 0*10^-6

Bddb4 = 0.1% (FE B)
Bdub4 = 0.1% (FE B)
Bsub4 = 0.1% (FE B)
Bsdb4 = 0.1% (FE B)

Bddc4 = 0.1
Bduc4 = 0.1
Bsd4 = 0.1
Bsuc4 = 0.1

Cpurchaseb4 = 6400;
Ccmb = Xtb4*480*Nx;
```

```
Cprob = [Cpurchaseb4+Cinscomm]*Nx;
```

```
elseif teknoa4==3
```

```
Xddb4 = 0.79*10^-6
```

```
Xdub4 = 7.11*10^-6
```

```
Xsdb4 = 0
```

```
Xsub4 = 9.17*10^-6
```

```
Xtb4 = 17.07*10^-6
```

```
Xtc4 = 3.492*10^-6
```

```
Xsdc4 = 3.46*10^-6
```

```
Xsuc4 = 0*10^-6
```

```
Bsdb4 = 0.1 % (FE C)
```

```
Bddb4 = 0.1 % (FE C)
```

```
Bdub4 = 0.1 % (FE C)
```

```
Bsub4 = 0.1 % (FE C)
```

```
Bsdb4 = 0.05% (FE C)
```

```
Bddc4 = 0.1
```

```
Bduc4 = 0.1
```

```
Bsdc4 = 0.1
```

```
Bsuc4 = 0.1
```

```
Cpurchaseb4 = 6200;
```

```
Ccmb = Xtb4*480*Nx;
```

```
Cprob = [Cpurchaseb4+Cinscomm]*Nx;
```

```
end
```

```
if teknoa5==1
```

```
Xddb5 = 0.084*10^-6
```

```
Xdub5 = 2.51*10^-6 % (FE A)
```

```
Xsdb5 = 0 % (FE A)
```

```
Xsub5 = 3.94*10^-6 % (FE A)
```

```
Xtb5 = 7.29*10^-6 % (FE A)
```

```
Xtc5 = 3.492*10^-6% (PLC)
```

```
Xsdc5 = 3.46*10^-6 % (PLC)
```

```
Xsuc5 = 0*10^-6;    %(PLC)

Bddb5 = 0.1%(FE A)
Bdub5 = 0.1%(FE A)
Bsub5 = 0.1%(FE A)
Bsdb5 = 0.1%(FE A)

Bddc5 = 0.1
Bduc5 = 0.1
Bsdc5 = 0.1
Bsuc5 = 0.1

Cpurchaseb5 = 6940;
Ccmb = Xtb5*480*Nx;
Cprob = [Cpurchaseb5+Cinscomm]*Nx;

elseif teknoa5==2
Xddb5 = 1.09*10^-6
Xdub5 = 4.35*10^-6
Xsdb5 = 0
Xsub5 = 3.17*10^-6
Xtb5  = 8.61*10^-6

Xtc5  = 3.492*10^-6
Xsdc5 = 3.46*10^-6
Xsuc5 = 0*10^-6

Bddb5 = 0.1%(FE B)
Bdub5 = 0.1%(FE B)
Bsub5 = 0.1%(FE B)
Bsdb5 = 0.1%(FE B)

Bddc5 = 0.1
Bduc5 = 0.1
Bsdc5 = 0.1
Bsuc5 = 0.1

Cpurchaseb5 =6400;
```

```
Ccmb = Xtb5*480*Nx;  
Cprob = [Cpurchaseb5+Cinscomm]*Nx;
```

```
elseif teknoa5==3  
Xddb5 = 0.79*10^-6  
Xdub5 = 7.11*10^-6  
Xsdb5 = 0  
Xsub5 = 9.17*10^-6  
Xtb5 = 17.07*10^-6
```

```
Xtc5 = 3.492*10^-6  
Xsdc5 = 3.46*10^-6  
Xsuc5 = 0*10^-6
```

```
Bsdb5 = 0.1 % (FE C)  
Bddb5 = 0.1 % (FE C)  
Bdub5 = 0.1 % (FE C)  
Bsub5 = 0.1 % (FE C)  
Bsdb5 = 0.05% (FE C)
```

```
Bddc5 = 0.1  
Bduc5 = 0.1  
Bsd5 = 0.1  
Bsuc5 = 0.1
```

```
Cpurchaseb5 =6200;  
Ccmb = Xtb5*480*Nx;  
Cprob = [Cpurchaseb5+Cinscomm]*Nx;  
end
```

```
if teknoa6==1  
Xddb6 = 0.084*10^-6  
Xdub6 = 2.51*10^-6 % (FE A)  
Xsdb6 = 0 % (FE A)  
Xsub6 = 3.94*10^-6 % (FE A)  
Xtb6 = 7.29*10^-6 % (FE A)
```

```
Xtc6 = 3.492*10^-6% (PLC)
```

```

Xsdc6 = 3.46*10^-6 % (PLC)
Xsuc6 = 0*10^-6; % (PLC)

Bddb6 = 0.1% (FE A)
Bdub6 = 0.1% (FE A)
Bsub6 = 0.1% (FE A)
Bsdb6 = 0.1% (FE A)

Bddc6 = 0.1
Bduc6 = 0.1
Bsdc6 = 0.1
Bsuc6 = 0.1

Cpurchaseb6 =6940;
Ccmb = Xtb6*480*Nx;
Cprob = [Cpurchaseb6+Cinscomm]*Nx;

elseif teknoa6==2
Xddb6= 1.09*10^-6
Xdub6= 4.35*10^-6
Xsdb6= 0
Xsub6= 3.17*10^-6
Xtb6 = 8.61*10^-6

Xtc6 = 3.492*10^-6
Xsdc6 = 3.46*10^-6
Xsuc6 = 0*10^-6

Bddb6 = 0.1% (FE B)
Bdub6 = 0.1% (FE B)
Bsub6 = 0.1% (FE B)
Bsdb6 = 0.1% (FE B)

Bddc6 = 0.1
Bduc6 = 0.1
Bsdc6 = 0.1
Bsuc6 = 0.1

```

```
Cpurchaseb6 =6400;  
Ccmb = Xtb6*480*Nx;  
Cprob = [Cpurchaseb6+Cinscomm]*Nx;
```

```
elseif teknoa6==3  
Xddb6 = 0.79*10^-6  
Xdub6 = 7.11*10^-6  
Xsdb6 = 0  
Xsub6 = 9.17*10^-6  
Xtb6 = 17.07*10^-6
```

```
Xtc6 = 3.492*10^-6  
Xsdc6 = 3.46*10^-6  
Xsuc6 = 0*10^-6
```

```
Bsdb6 = 0.1%(FE C)  
Bddb6 = 0.1%(FE C)  
Bdub6 = 0.1%(FE C)  
Bsub6 = 0.1%(FE C)
```

```
Bsdb6 = 0.05%(FE C)  
Bddc6 = 0.1  
Bduc6 = 0.1  
Bsdc6 = 0.1  
Bsuc6 = 0.1
```

```
Cpurchaseb6 =6200;  
Ccmb = Xtb6*480*Nx;  
Cprob = [Cpurchaseb6+Cinscomm]*Nx;  
end
```

```
if i1==1  
Cx= 0;  
Nx= 1;  
PFDa1 =  
funcpfd(Cx,Bddal,Xddal,Bdual,Xdual,Ti,Tr);  
PFDa = PFDa1;
```

```
STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;  
STRa=STRa1;
```

```
elseif i1==2
```

```
Cx= 1;
```

```
Nx= 2;
```

```
PFDa1 =
```

```
funcpfd (Cx, Bdda1, Xdda1, Bdual, Xdual, Ti, Tr) ;
```

```
PFDa2 =
```

```
funcpfd (Cx, Bdda2, Xdda2, Bdual, Xdual, Ti, Tr) ;
```

```
PFDa = PFDa1+PFDa2;
```

```
STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;
```

```
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2) ;
```

```
STRa=STRa1+STRa2;
```

```
elseif i1==3
```

```
Cx= 0.3
```

```
Nx= 3
```

```
PFDa1 =
```

```
funcpfd (Cx, Bdda1, Xdda1, Bdual, Xdual, Ti, Tr) ;
```

```
PFDa2 =
```

```
funcpfd (Cx, Bdda2, Xdda2, Bdual, Xdual, Ti, Tr) ;
```

```
PFDa3 =
```

```
funcpfd (Cx, Bdda3, Xdda3, Bdual, Xdual, Ti, Tr) ;
```

```
PFDa = PFDa1+PFDa2+PFDa3;
```

```
STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;
```

```
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2) ;
```

```
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3) ;
```

```
STRa=STRa1+STRa2+STRa3;
```

```
elseif i1==4
```

```
Cx= 0.15
```

```
Nx= 4
```

```
PFDa1 =
```

```
funcpfd (Cx, Bdda1, Xdda1, Bdual, Xdual, Ti, Tr) ;
```

```
PFDa2 =
funcpfd(Cx,Bdda2,Xdda2,Bdua2,Xdua2,Ti,Tr);
PFDa3 =
funcpfd(Cx,Bdda3,Xdda3,Bdua3,Xdua3,Ti,Tr);
PFDa4 =
funcpfd(Cx,Bdda4,Xdda4,Bdua4,Xdua4,Ti,Tr);
PFDa = PFDa1+PFDa2+PFDa3+PFDa4;
```

```
STRa1=funcstr(Cx,Bsda1,Xsda1,Bsua1,Xsub1);
STRa2=funcstr(Cx,Bsda2,Xsda2,Bsua2,Xsub2);
STRa3=funcstr(Cx,Bsda3,Xsda3,Bsua3,Xsub3);
STRa4=funcstr(Cx,Bsda4,Xsda4,Bsua4,Xsub4);
STRa=STRa1+STRa2+STRa3+STRa4;
```

```
elseif i1==5
```

```
Cx= 0.08
```

```
Nx= 5
```

```
PFDa1 =
```

```
funcpfd(Cx,Bdda1,Xdda1,Bdua1,Xdua1,Ti,Tr);
```

```
PFDa2 =
```

```
funcpfd(Cx,Bdda2,Xdda2,Bdua2,Xdua2,Ti,Tr);
```

```
PFDa3 =
```

```
funcpfd(Cx,Bdda3,Xdda3,Bdua3,Xdua3,Ti,Tr);
```

```
PFDa4 =
```

```
funcpfd(Cx,Bdda4,Xdda4,Bdua4,Xdua4,Ti,Tr);
```

```
PFDa5 =
```

```
funcpfd(Cx,Bdda5,Xdda5,Bdua5,Xdua5,Ti,Tr);
```

```
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5;
```

```
STRa1=funcstr(Cx,Bsda1,Xsda1,Bsua1,Xsub1);
```

```
STRa2=funcstr(Cx,Bsda2,Xsda2,Bsua2,Xsub2);
```

```
STRa3=funcstr(Cx,Bsda3,Xsda3,Bsua3,Xsub3);
```

```
STRa4=funcstr(Cx,Bsda4,Xsda4,Bsua4,Xsub4);
```

```
STRa5=funcstr(Cx,Bsda5,Xsda5,Bsua5,Xsub5);
```

```
STRa=STRa1+STRa2+STRa3+STRa4+STRa5;
```

```
elseif i1==6
```

```
Cx= 0.04
```

```
Nx= 6
```



```

PFDa1 =
funcpfd (Cx, Bdda1, Xdda1, Bdua1, Xdua1, Ti, Tr) ;
PFDa2 =
funcpfd (Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr) ;
PFDa3 =
funcpfd (Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr) ;
PFDa4 =
funcpfd (Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr) ;
PFDa5 =
funcpfd (Cx, Bdda5, Xdda5, Bdua5, Xdua5, Ti, Tr) ;
PFDa6 =
funcpfd (Cx, Bdda6, Xdda6, Bdua6, Xdua6, Ti, Tr) ;
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5+PFDa6;

```

```

STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2) ;
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3) ;
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4) ;
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5) ;
STRa6=funcstr (Cx, Bsda6, Xsda6, Bsua6, Xsub6) ;
STRa=STRa1+STRa2+STRa3+STRa4+STRa5+STRa6;

```

```

elseif i1==7

```

```

Cx= 0
Nx= 2
PFDa1 =
funcpfd (Cx, Bdda1, Xdda1, Bdua1, Xdua1, Ti, Tr) ;
PFDa2 =
funcpfd (Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr) ;
PFDa = PFDa1+PFDa2;

```

```

STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2) ;
STRa=STRa1+STRa2;

```

```

elseif i1==8

```

```

Cx= 2.4
Nx= 3

```

```
PFDa1 =  
funcpfd(Cx, Bdda1, Xdda1, Bdual, Xdua1, Ti, Tr);  
PFDa2 =  
funcpfd(Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr);  
PFDa3 =  
funcpfd(Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr);  
PFDa = PFDa1+PFDa2+PFDa3;
```

```
STRa1=funcstr(Cx, Bsda1, Xsda1, Bsua1, Xsub1);  
STRa2=funcstr(Cx, Bsda2, Xsda2, Bsua2, Xsub2);  
STRa3=funcstr(Cx, Bsda3, Xsda3, Bsua3, Xsub3);  
STRa=STRa1+STRa2+STRa3;
```

```
elseif i1==9
```

```
Cx= 0.75
```

```
Nx= 4
```

```
PFDa1 =
```

```
funcpfd(Cx, Bdda1, Xdda1, Bdual, Xdua1, Ti, Tr);
```

```
PFDa2 =
```

```
funcpfd(Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr);
```

```
PFDa3 =
```

```
funcpfd(Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr);
```

```
PFDa4 =
```

```
funcpfd(Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr);
```

```
PFDa = PFDa1+PFDa2+PFDa3+PFDa4;
```

```
STRa1=funcstr(Cx, Bsda1, Xsda1, Bsua1, Xsub1);
```

```
STRa2=funcstr(Cx, Bsda2, Xsda2, Bsua2, Xsub2);
```

```
STRa3=funcstr(Cx, Bsda3, Xsda3, Bsua3, Xsub3);
```

```
STRa4=funcstr(Cx, Bsda4, Xsda4, Bsua4, Xsub4);
```

```
STRa=STRa1+STRa2+STRa3+STRa4;
```

```
elseif i1==10
```

```
Cx= 0.45
```

```
Nx= 5
```

```
PFDa1 =
```

```
funcpfd(Cx, Bdda1, Xdda1, Bdual, Xdua1, Ti, Tr);
```

```
PFDa2 =
```

```
funcpfd(Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr);
```

```

PFDa3 =
funcpfd (Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr);
PFDa4 =
funcpfd (Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr);
PFDa5 =
funcpfd (Cx, Bdda5, Xdda5, Bdua5, Xdua5, Ti, Tr);
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5;

STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1);
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2);
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3);
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4);
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5);
STRa=STRa1+STRa2+STRa3+STRa4+STRa5;

elseif i1==11
Cx= 0.26
Nx= 6
PFDa1 =
funcpfd (Cx, Bdda1, Xdda1, Bdua1, Xdua1, Ti, Tr);
PFDa2 =
funcpfd (Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr);
PFDa3 =
funcpfd (Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr);
PFDa4 =
funcpfd (Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr);
PFDa5 =
funcpfd (Cx, Bdda5, Xdda5, Bdua5, Xdua5, Ti, Tr);
PFDa6 =
funcpfd (Cx, Bdda6, Xdda6, Bdua6, Xdua6, Ti, Tr);
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5+PFDa6;

STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1);
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2);
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3);
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4);
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5);
STRa6=funcstr (Cx, Bsda6, Xsda6, Bsua6, Xsub6);
STRa=STRa1+STRa2+STRa3+STRa4+STRa5+STRa6;

```

```

elseif i1==12
Cx= 0
Nx= 3
PFDa1 =
funcpfd(Cx,Bdda1,Xdda1,Bdua1,Xdua1,Ti,Tr);
PFDa2 =
funcpfd(Cx,Bdda2,Xdda2,Bdua2,Xdua2,Ti,Tr);
PFDa3 =
funcpfd(Cx,Bdda3,Xdda3,Bdua3,Xdua3,Ti,Tr);
PFDa = PFDa1+PFDa2+PFDa3;

STRa1=funcstr(Cx,Bsda1,Xsda1,Bsua1,Xsub1);
STRa2=funcstr(Cx,Bsda2,Xsda2,Bsua2,Xsub2);
STRa3=funcstr(Cx,Bsda3,Xsda3,Bsua3,Xsub3);
STRa=STRa1+STRa2+STRa3;

elseif i1==13
Cx= 4
Nx= 4
PFDa1 =
funcpfd(Cx,Bdda1,Xdda1,Bdua1,Xdua1,Ti,Tr);
PFDa2 =
funcpfd(Cx,Bdda2,Xdda2,Bdua2,Xdua2,Ti,Tr);
PFDa3 =
funcpfd(Cx,Bdda3,Xdda3,Bdua3,Xdua3,Ti,Tr);
PFDa4 =
funcpfd(Cx,Bdda4,Xdda4,Bdua4,Xdua4,Ti,Tr);
PFDa = PFDa1+PFDa2+PFDa3+PFDa4;

STRa1=funcstr(Cx,Bsda1,Xsda1,Bsua1,Xsub1);
STRa2=funcstr(Cx,Bsda2,Xsda2,Bsua2,Xsub2);
STRa3=funcstr(Cx,Bsda3,Xsda3,Bsua3,Xsub3);
STRa4=funcstr(Cx,Bsda4,Xsda4,Bsua4,Xsub4);
STRa=STRa1+STRa2+STRa3+STRa4;

elseif i1==14
Cx= 1.2

```

```

Nx= 5
PFDa1 =
funcpfd(Cx,Bdda1,Xdda1,Bdua1,Xdua1,Ti,Tr);
PFDa2 =
funcpfd(Cx,Bdda2,Xdda2,Bdua2,Xdua2,Ti,Tr);
PFDa3 =
funcpfd(Cx,Bdda3,Xdda3,Bdua3,Xdua3,Ti,Tr);
PFDa4 =
funcpfd(Cx,Bdda4,Xdda4,Bdua4,Xdua4,Ti,Tr);
PFDa5 =
funcpfd(Cx,Bdda5,Xdda5,Bdua5,Xdua5,Ti,Tr);
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5;

STRa1=funcstr(Cx,Bsda1,Xsda1,Bsua1,Xsub1);
STRa2=funcstr(Cx,Bsda2,Xsda2,Bsua2,Xsub2);
STRa3=funcstr(Cx,Bsda3,Xsda3,Bsua3,Xsub3);
STRa4=funcstr(Cx,Bsda4,Xsda4,Bsua4,Xsub4);
STRa5=funcstr(Cx,Bsda5,Xsda5,Bsua5,Xsub5);
STRa=STRa1+STRa2+STRa3+STRa4+STRa5;

elseif i1==15
Cx= 0.8
Nx= 6
PFDa1 =
funcpfd(Cx,Bdda1,Xdda1,Bdua1,Xdua1,Ti,Tr);
PFDa2 =
funcpfd(Cx,Bdda2,Xdda2,Bdua2,Xdua2,Ti,Tr);
PFDa3 =
funcpfd(Cx,Bdda3,Xdda3,Bdua3,Xdua3,Ti,Tr);
PFDa4 =
funcpfd(Cx,Bdda4,Xdda4,Bdua4,Xdua4,Ti,Tr);
PFDa5 =
funcpfd(Cx,Bdda5,Xdda5,Bdua5,Xdua5,Ti,Tr);
PFDa6 =
funcpfd(Cx,Bdda6,Xdda6,Bdua6,Xdua6,Ti,Tr);
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5+PFDa6;

STRa1=funcstr(Cx,Bsda1,Xsda1,Bsua1,Xsub1);
STRa2=funcstr(Cx,Bsda2,Xsda2,Bsua2,Xsub2);

```

```
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3) ;  
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4) ;  
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5) ;  
STRa6=funcstr (Cx, Bsda6, Xsda6, Bsua6, Xsub6) ;  
STRa=STRa1+STRa2+STRa3+STRa4+STRa5+STRa6;
```

```
elseif i1==16
```

```
Cx= 0
```

```
Nx= 4
```

```
PFDa1 =
```

```
funcpfd (Cx, Bdda1, Xdda1, Bdua1, Xdua1, Ti, Tr) ;
```

```
PFDa2 =
```

```
funcpfd (Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr) ;
```

```
PFDa3 =
```

```
funcpfd (Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr) ;
```

```
PFDa4 =
```

```
funcpfd (Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr) ;
```

```
PFDa = PFDa1+PFDa2+PFDa3+PFDa4;
```

```
STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;
```

```
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2) ;
```

```
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3) ;
```

```
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4) ;
```

```
STRa=STRa1+STRa2+STRa3+STRa4;
```

```
elseif i1==17
```

```
Cx= 6.0
```

```
Nx= 5
```

```
PFDa1 =
```

```
funcpfd (Cx, Bdda1, Xdda1, Bdua1, Xdua1, Ti, Tr) ;
```

```
PFDa2 =
```

```
funcpfd (Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr) ;
```

```
PFDa3 =
```

```
funcpfd (Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr) ;
```

```
PFDa4 =
```

```
funcpfd (Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr) ;
```

```
PFDa5 =
```

```
funcpfd (Cx, Bdda5, Xdda5, Bdua5, Xdua5, Ti, Tr) ;
```

```
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5;
```

```
STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;  
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2) ;  
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3) ;  
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4) ;  
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5) ;  
STRa=STRa1+STRa2+STRa3+STRa4+STRa5;
```

```
elseif i1==18
```

```
Cx= 1.6
```

```
Nx= 6
```

```
PFDa1 =
```

```
funcpfd (Cx, Bdda1, Xdda1, Bdua1, Xdua1, Ti, Tr) ;
```

```
PFDa2 =
```

```
funcpfd (Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr) ;
```

```
PFDa3 =
```

```
funcpfd (Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr) ;
```

```
PFDa4 =
```

```
funcpfd (Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr) ;
```

```
PFDa5 =
```

```
funcpfd (Cx, Bdda5, Xdda5, Bdua5, Xdua5, Ti, Tr) ;
```

```
PFDa6 =
```

```
funcpfd (Cx, Bdda6, Xdda6, Bdua6, Xdua6, Ti, Tr) ;
```

```
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5+PFDa6;
```

```
STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;
```

```
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2) ;
```

```
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3) ;
```

```
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4) ;
```

```
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5) ;
```

```
STRa6=funcstr (Cx, Bsda6, Xsda6, Bsua6, Xsub6) ;
```

```
STRa=STRa1+STRa2+STRa3+STRa4+STRa5+STRa6;
```

```
elseif i1==19
```

```
Cx= 0
```

```
Nx= 5
```

```
PFDa1 =
```

```
funcpfd (Cx, Bdda1, Xdda1, Bdua1, Xdua1, Ti, Tr) ;
```

```
PFDa2 =
funcpfd (Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr);
PFDa3 =
funcpfd (Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr);
PFDa4 =
funcpfd (Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr);
PFDa5 =
funcpfd (Cx, Bdda5, Xdda5, Bdua5, Xdua5, Ti, Tr);
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5;
```

```
STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1);
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2);
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3);
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4);
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5);
STRa=STRa1+STRa2+STRa3+STRa4+STRa5;
```

```
elseif i1==20
```

```
Cx= 8.1
```

```
Nx= 6
```

```
PFDa1 =
funcpfd (Cx, Bdda1, Xdda1, Bdua1, Xdua1, Ti, Tr);
PFDa2 =
funcpfd (Cx, Bdda2, Xdda2, Bdua2, Xdua2, Ti, Tr);
PFDa3 =
funcpfd (Cx, Bdda3, Xdda3, Bdua3, Xdua3, Ti, Tr);
PFDa4 =
funcpfd (Cx, Bdda4, Xdda4, Bdua4, Xdua4, Ti, Tr);
PFDa5 =
funcpfd (Cx, Bdda5, Xdda5, Bdua5, Xdua5, Ti, Tr);
PFDa6 =
funcpfd (Cx, Bdda6, Xdda6, Bdua6, Xdua6, Ti, Tr);
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5+PFDa6;
```

```
STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1);
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2);
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3);
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4);
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5);
```



```

STRa6=funcstr (Cx, Bsda6, Xsda6, Bsua6, Xsub6) ;
STRa=STRa1+STRa2+STRa3+STRa4+STRa5+STRa6;

elseif i1==21
Cx= 0
Nx= 6
PFDa1 =
funcpfd (Cx, Bdda1, Xdda1, Bdual, Xdua1, Ti, Tr) ;
PFDa2 =
funcpfd (Cx, Bdda2, Xdda2, Bdual, Xdua2, Ti, Tr) ;
PFDa3 =
funcpfd (Cx, Bdda3, Xdda3, Bdual, Xdua3, Ti, Tr) ;
PFDa4 =
funcpfd (Cx, Bdda4, Xdda4, Bdual, Xdua4, Ti, Tr) ;
PFDa5 =
funcpfd (Cx, Bdda5, Xdda5, Bdual, Xdua5, Ti, Tr) ;
PFDa6 =
funcpfd (Cx, Bdda6, Xdda6, Bdual, Xdua6, Ti, Tr) ;
PFDa = PFDa1+PFDa2+PFDa3+PFDa4+PFDa5+PFDa6;

STRa1=funcstr (Cx, Bsda1, Xsda1, Bsua1, Xsub1) ;
STRa2=funcstr (Cx, Bsda2, Xsda2, Bsua2, Xsub2) ;
STRa3=funcstr (Cx, Bsda3, Xsda3, Bsua3, Xsub3) ;
STRa4=funcstr (Cx, Bsda4, Xsda4, Bsua4, Xsub4) ;
STRa5=funcstr (Cx, Bsda5, Xsda5, Bsua5, Xsub5) ;
STRa6=funcstr (Cx, Bsda6, Xsda6, Bsua6, Xsub6) ;
STRa=STRa1+STRa2+STRa3+STRa4+STRa5+STRa6;
end

if i2==1
Cx= 0
Nx= 1
PFDb1 =
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;
PFDb=PFDb1

STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;
STRb=STRb1

```

```

elseif i2==2
Cx= 1
Nx= 2
PFDb1 =
funcpfd(Cx,Bddb1,Xddb1,Bdub1,Xdub1,Ti,Tr);
PFDb2 =
funcpfd(Cx,Bddb2,Xddb2,Bdub2,Xdub2,Ti,Tr);
PFDb=PFDb1+PFDb2;

STRb1=funcstr(Cx,Bsdb1,Xsdb1,Bsub1,Xsub1);
STRb2=funcstr(Cx,Bsdb2,Xsdb2,Bsub2,Xsub2);
STRb=STRb1+STRb2;

elseif i2==3
Cx= 0.3
Nx= 3
PFDb1 =
funcpfd(Cx,Bddb1,Xddb1,Bdub1,Xdub1,Ti,Tr);
PFDb2 =
funcpfd(Cx,Bddb2,Xddb2,Bdub2,Xdub2,Ti,Tr);
PFDb3 =
funcpfd(Cx,Bddb3,Xddb3,Bdub3,Xdub3,Ti,Tr);
PFDb=PFDb1+PFDb2+PFDb3;

STRb1=funcstr(Cx,Bsdb1,Xsdb1,Bsub1,Xsub1);
STRb2=funcstr(Cx,Bsdb2,Xsdb2,Bsub2,Xsub2);
STRb3=funcstr(Cx,Bsdb3,Xsdb3,Bsub3,Xsub3);
STRb=STRb1+STRb2+STRb3;

elseif i2==4
Cx= 0.15
Nx= 4
PFDb1 =
funcpfd(Cx,Bddb1,Xddb1,Bdub1,Xdub1,Ti,Tr);
PFDb2 =
funcpfd(Cx,Bddb2,Xddb2,Bdub2,Xdub2,Ti,Tr);

```

```
PFDb3 =  
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr);  
PFDb4 =  
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr);  
PFDb=PFDb1+PFDb2+PFDb3+PFDb4;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1);  
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2);  
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3);  
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4);  
STRb=STRb1+STRb2+STRb3+STRb4;
```

```
elseif i2==5
```

```
Cx= 0.08
```

```
Nx= 5
```

```
PFDb1 =
```

```
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr);
```

```
PFDb2 =
```

```
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr);
```

```
PFDb3 =
```

```
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr);
```

```
PFDb4 =
```

```
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr);
```

```
PFDb5 =
```

```
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr);
```

```
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1);
```

```
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2);
```

```
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3);
```

```
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4);
```

```
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5);
```

```
STRb=STRb1+STRb2+STRb3+STRb4+STRb5;
```

```
elseif i2==6
```

```
Cx= 0.04
```

```
Nx= 6
```

```
PFDb1 =
```

```
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr);
```

```
PFDb2 =
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
PFDb3 =
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;
PFDb4 =
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr) ;
PFDb5 =
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr) ;
PFDb6 =
funcpfd (Cx, Bddb6, Xddb6, Bdub6, Xdub6, Ti, Tr) ;
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5+PFDb6;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3) ;
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4) ;
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5) ;
STRb6=funcstr (Cx, Bsdb6, Xsdb6, Bsub6, Xsub6) ;
STRb=STRb1+STRb2+STRb3+STRb4+STRb5+STRb6;
```

```
elseif i2==7
```

```
Cx= 0
```

```
Nx= 2
```

```
PFDb1 =
```

```
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;
```

```
PFDb2 =
```

```
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
```

```
PFDb=PFDb1+PFDb2;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;
```

```
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;
```

```
STRb=STRb1+STRb2;
```

```
elseif i2==8
```

```
Cx= 2.4
```

```
Nx= 3
```

```
PFDb1 =
```

```
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;
```

```
PFDb2 =
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
PFDb3 =
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;
PFDb=PFDb1+PFDb2+PFDb3;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3) ;
STRb=STRb1+STRb2+STRb3;
```

```
elseif i2==9
```

```
Cx= 0.75
```

```
Nx= 4
```

```
PFDb1 =
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;
PFDb2 =
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
PFDb3 =
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;
PFDb4 =
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr) ;
PFDb=PFDb1+PFDb2+PFDb3+PFDb4;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3) ;
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4) ;
STRb=STRb1+STRb2+STRb3+STRb4;
```

```
elseif i2==10
```

```
Cx= 0.45
```

```
Nx= 5
```

```
PFDb1 =
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;
PFDb2 =
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
PFDb3 =
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;
```

```
PFDb4 =
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr);
PFDb5 =
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr);
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1);
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2);
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3);
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4);
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5);
STRb=STRb1+STRb2+STRb3+STRb4+STRb5;
```

```
elseif i2==11
```

```
Cx= 0.26
```

```
Nx= 6
```

```
PFDb1 =
```

```
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr);
```

```
PFDb2 =
```

```
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr);
```

```
PFDb3 =
```

```
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr);
```

```
PFDb4 =
```

```
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr);
```

```
PFDb5 =
```

```
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr);
```

```
PFDb6 =
```

```
funcpfd (Cx, Bddb6, Xddb6, Bdub6, Xdub6, Ti, Tr);
```

```
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5+PFDb6;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1);
```

```
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2);
```

```
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3);
```

```
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4);
```

```
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5);
```

```
STRb6=funcstr (Cx, Bsdb6, Xsdb6, Bsub6, Xsub6);
```

```
STRb=STRb1+STRb2+STRb3+STRb4+STRb5+STRb6;
```

```
elseif i2==12
```

```
Cx= 0
Nx= 3
PFDb1 =
funcpfd(Cx,Bddb1,Xddb1,Bdub1,Xdub1,Ti,Tr);
PFDb2 =
funcpfd(Cx,Bddb2,Xddb2,Bdub2,Xdub2,Ti,Tr);
PFDb3 =
funcpfd(Cx,Bddb3,Xddb3,Bdub3,Xdub3,Ti,Tr);
PFDb=PFDb1+PFDb2+PFDb3;
```

```
STRb1=funcstr(Cx,Bsdb1,Xsdb1,Bsub1,Xsub1);
STRb2=funcstr(Cx,Bsdb2,Xsdb2,Bsub2,Xsub2);
STRb3=funcstr(Cx,Bsdb3,Xsdb3,Bsub3,Xsub3);
STRb=STRb1+STRb2+STRb3;
```

```
elseif i2==13
```

```
Cx= 4
Nx= 4
PFDb1 =
funcpfd(Cx,Bddb1,Xddb1,Bdub1,Xdub1,Ti,Tr);
PFDb2 =
funcpfd(Cx,Bddb2,Xddb2,Bdub2,Xdub2,Ti,Tr);
PFDb3 =
funcpfd(Cx,Bddb3,Xddb3,Bdub3,Xdub3,Ti,Tr);
PFDb4 =
funcpfd(Cx,Bddb4,Xddb4,Bdub4,Xdub4,Ti,Tr);
PFDb=PFDb1+PFDb2+PFDb3+PFDb4;
```

```
STRb1=funcstr(Cx,Bsdb1,Xsdb1,Bsub1,Xsub1);
STRb2=funcstr(Cx,Bsdb2,Xsdb2,Bsub2,Xsub2);
STRb3=funcstr(Cx,Bsdb3,Xsdb3,Bsub3,Xsub3);
STRb4=funcstr(Cx,Bsdb4,Xsdb4,Bsub4,Xsub4);
STRb=STRb1+STRb2+STRb3+STRb4;
```

```
elseif i2==14
```

```
Cx= 1.2
Nx= 5
PFDb1 =
funcpfd(Cx,Bddb1,Xddb1,Bdub1,Xdub1,Ti,Tr);
```

```
PFDb2 =
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
PFDb3 =
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;
PFDb4 =
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr) ;
PFDb5 =
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr) ;
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3) ;
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4) ;
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5) ;
STRb=STRb1+STRb2+STRb3+STRb4+STRb5;
```

```
elseif i2==15
```

```
Cx= 0.8
```

```
Nx= 6
```

```
PFDb1 =
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;
PFDb2 =
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
PFDb3 =
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;
PFDb4 =
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr) ;
PFDb5 =
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr) ;
PFDb6 =
funcpfd (Cx, Bddb6, Xddb6, Bdub6, Xdub6, Ti, Tr) ;
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5+PFDb6;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3) ;
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4) ;
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5) ;
```



```
STRb6=funcstr (Cx, Bsdb6, Xsdb6, Bsub6, Xsub6) ;  
STRb=STRb1+STRb2+STRb3+STRb4+STRb5+STRb6;
```

```
elseif i2==16
```

```
Cx= 0  
Nx= 4  
PFDb1 =  
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;  
PFDb2 =  
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;  
PFDb3 =  
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;  
PFDb4 =  
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr) ;  
PFDb=PFDb1+PFDb2+PFDb3+PFDb4;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;  
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;  
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3) ;  
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4) ;  
STRb=STRb1+STRb2+STRb3+STRb4;
```

```
elseif i2==17
```

```
Cx= 6.0  
Nx= 5  
PFDb1 =  
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;  
PFDb2 =  
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;  
PFDb3 =  
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;  
PFDb4 =  
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr) ;  
PFDb5 =  
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr) ;  
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;  
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;
```

```
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3) ;  
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4) ;  
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5) ;  
STRb=STRb1+STRb2+STRb3+STRb4+STRb5 ;
```

```
elseif i2==18
```

```
Cx= 1.6
```

```
Nx= 6
```

```
PFDb1 =
```

```
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;
```

```
PFDb2 =
```

```
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
```

```
PFDb3 =
```

```
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;
```

```
PFDb4 =
```

```
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr) ;
```

```
PFDb5 =
```

```
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr) ;
```

```
PFDb6 =
```

```
funcpfd (Cx, Bddb6, Xddb6, Bdub6, Xdub6, Ti, Tr) ;
```

```
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5+PFDb6 ;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1) ;
```

```
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2) ;
```

```
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3) ;
```

```
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4) ;
```

```
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5) ;
```

```
STRb6=funcstr (Cx, Bsdb6, Xsdb6, Bsub6, Xsub6) ;
```

```
STRb=STRb1+STRb2+STRb3+STRb4+STRb5+STRb6 ;
```

```
elseif i2==19
```

```
Cx= 0
```

```
Nx= 5
```

```
PFDb1 =
```

```
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr) ;
```

```
PFDb2 =
```

```
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr) ;
```

```
PFDb3 =
```

```
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr) ;
```

```
PFDb4 =
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr);
PFDb5 =
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr);
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1);
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2);
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3);
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4);
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5);
STRb=STRb1+STRb2+STRb3+STRb4+STRb5;
```

```
elseif i2==20
```

```
Cx= 8.1
```

```
Nx= 6
```

```
PFDb1 =
```

```
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr);
```

```
PFDb2 =
```

```
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr);
```

```
PFDb3 =
```

```
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr);
```

```
PFDb4 =
```

```
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr);
```

```
PFDb5 =
```

```
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr);
```

```
PFDb6 =
```

```
funcpfd (Cx, Bddb6, Xddb6, Bdub6, Xdub6, Ti, Tr);
```

```
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5+PFDb6;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1);
```

```
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2);
```

```
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3);
```

```
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4);
```

```
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5);
```

```
STRb6=funcstr (Cx, Bsdb6, Xsdb6, Bsub6, Xsub6);
```

```
STRb=STRb1+STRb2+STRb3+STRb4+STRb5+STRb6;
```

```
elseif i2==21
```

```
Cx= 0
Nx= 6
PFDb1 =
funcpfd (Cx, Bddb1, Xddb1, Bdub1, Xdub1, Ti, Tr);
PFDb2 =
funcpfd (Cx, Bddb2, Xddb2, Bdub2, Xdub2, Ti, Tr);
PFDb3 =
funcpfd (Cx, Bddb3, Xddb3, Bdub3, Xdub3, Ti, Tr);
PFDb4 =
funcpfd (Cx, Bddb4, Xddb4, Bdub4, Xdub4, Ti, Tr);
PFDb5 =
funcpfd (Cx, Bddb5, Xddb5, Bdub5, Xdub5, Ti, Tr);
PFDb6 =
funcpfd (Cx, Bddb6, Xddb6, Bdub6, Xdub6, Ti, Tr);
PFDb=PFDb1+PFDb2+PFDb3+PFDb4+PFDb5+PFDb6;
```

```
STRb1=funcstr (Cx, Bsdb1, Xsdb1, Bsub1, Xsub1);
STRb2=funcstr (Cx, Bsdb2, Xsdb2, Bsub2, Xsub2);
STRb3=funcstr (Cx, Bsdb3, Xsdb3, Bsub3, Xsub3);
STRb4=funcstr (Cx, Bsdb4, Xsdb4, Bsub4, Xsub4);
STRb5=funcstr (Cx, Bsdb5, Xsdb5, Bsub5, Xsub5);
STRb6=funcstr (Cx, Bsdb6, Xsdb6, Bsub6, Xsub6);
STRb=STRb1+STRb2+STRb3+STRb4+STRb5+STRb6;
```

end

```
Ccmc = Xtcl*480*Nx;
```

```
Cta = 60*Nx/Ti;
```

```
Ctb = 60*Nx/Ti;
```

```
Ctc = 60*Nx/Ti;
```

```
Cpma = 240*Nx;
```

```
Cpmb = 240*Nx;
```

```
Cpmc = 240*Nx;
```

```
%Perhitungan PFD PLC
```

```
PFDC = 0.0001;
```

```

%Perhitungan STR PLC
STRc=funcstr(Cx,Bsdc,Xsdc,Bsuc,Xsuc);

for Ns = 1:1
%Perhitungan PFDavg dan STRavg
if Ns==1
PFDavg1(1,Ns) = PFDa+PFDb+PFDc;
STRavg1(1,Ns) = STRa+STRb+STRc;
Ccm(1,Ns)=Ccma+Ccmb+Ccmc;
Cpm(1,Ns)=Cpma+Cpmb+Cpmc;
Ct(1,Ns)=Cta+Ctb+Ctc;
end
end

PFDavg=PFDavg1(1,1)
STRavg=STRavg1(1,1)
Ccm=Ccm(1,1);
Cpm=Cpm(1,1);
Ct=Ct(1,1);

%Perhitungan Cost procurement masing-masing
instrumen
Cproc = [Cpurchasec+CdPLC];

Csd=SDtime*SDloss;

%Perhitungan Cost STR
Cstr = STRavg*Csd;

%Perhitungan Cost Hazard
Chazard = Cacc*Facc*PFDavg;

%Perhitungan Cost Risk
Crisk = Cstr+Chazard;

%Perhitungan Cost Operasional
Cop = Cpm+Ct+Ccm;

```

```
%Perhitungan keseluruhan Cost procurement
Cproall1 = Cd+Cproa+Cprob+Cproc;
Cproall = Cproall1;

PVF=(1-(1+R)^-YL)/R;

%Perhitungan Lifecycle Cost
LCC = Cproall+[Cop+Crisk]*PVF;
plot3(LCC, PFDavg, STRavg);

save('objfun_nadyaindd.mat')
end
```