

40352/H/10



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

RSI  
658.312 5  
Han  
P-1  
2010

TUGAS AKHIR - TI 091324

## PERENCANAAN MANHOURS C-CHECK MAINTENANCE PADA UNIT BASE MAINTENANCE PT. GMF AEROASIA

RITA UMMI HANIK  
NRP 2506 100 131

Dosen Pembimbing  
Ir. Budi Santosa, M.S., Ph.D.  
Dosen Co-Pembimbing  
Nani Kurniati, S.T., M.T.

JURUSAN TEKNIK INDUSTRI  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2010

PERPUSTAKAAN ITS	
Tgl Terima	2-8-2010
Terima oleh	H
No Agenda Pro	



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**FINAL PROJECT - TI 091324**

# **MANHOURS PLANNING OF C-CHECK MAINTENANCE ON PT. GMF AEROASIA BASE MAINTENANCE UNIT**

**TA UMMI HANIK**  
NRP. 2506 100 131

Supervisor  
Budi Santosa, M.S., Ph.D.  
Co-Supervisor  
Ani Kurniati, S.T., M.T.

DEPARTMENT OF INDUSTRIAL ENGINEERING  
Faculty of Industrial Technology  
Sepuluh Nopember Institute of Technology  
Surabaya 2010

**PERENCANAAN *MANHOURS C-CHECK MAINTENANCE*  
PADA UNIT *BASE MAINTENANCE*  
PT. GMF AEROASIA**

**TUGAS AKHIR**

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

Oleh :

**RITA UMMI HANIK**

Nrp. 2506 100 131

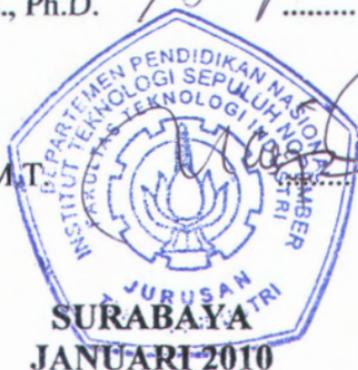
Disetujui oleh Dosen Pembimbing Tugas Akhir :

Ir. Budi Santosa, M.S., Ph.D.

..... (Pembimbing I)

Nani Kurniati, S.T., M.T.

..... (Pembimbing II)



**PERENCANAAN *MANHOURS C-CHECK*  
*MAINTENANCE* PADA UNIT *BASE MAINTENANCE*  
PT. GMF AEROASIA**

**Nama** : Rita Ummi Hanik  
**NRP** : 2506 100 131  
**Jurusan** : Teknik Industri FTI-ITS  
**Dosen Pembimbing** : Ir. Budi Santosa, M.S., Ph.D.  
**Dosen Ko Pembimbing** : Nani Kurniati, S.T., M.T.

**Abstrak**

PT. GMF AeroAsia merupakan salah satu perusahaan *maintenance* pesawat terbang terbesar di Asia yang selalu berupaya se-efisien dan se-efektif mungkin menyelesaikan *order* yang diberikan oleh *customer*. Efisiensi *manhours* merupakan salah satu faktor yang perlu diperhatikan dalam melakukan upaya tersebut. Tingkat efisiensi *manhours* merupakan perbandingan antara *manhours* yang direncanakan terhadap *manhours* aktual. Berdasarkan pengalaman diketahui bahwa efisiensi *manhours* mencapai 249% pada tipe *maintenance C-Check* di unit *Base Maintenance*, sehingga perlu dilakukan evaluasi dalam perencanaan *manhours*-nya. Selama ini perencanaan *manhours* dilakukan dengan memberikan nilai faktor pengali pada *Maintenance Planning Data* (MPD). Faktor pengali ini memiliki unsur subjektifitas *planner* yang ditentukan berdasarkan pengalaman di lapangan dan terdapat perbedaan antara *planner* yang satu dengan yang lain sehingga diperlukan suatu standard faktor pengali baru. Untuk mendapatkan standard faktor pengali tersebut digunakan metode *linear programming* dan untuk mengetahui performansinya digunakan metode *Artificial Neural Network*. Hasil penelitian ini menunjukkan bahwa alternatif terbaik adalah alternatif A yang memberikan peningkatan efisiensi terhadap kondisi *existing* dari 83% menjadi 103%.

**Kata kunci:** MRO, *planning manhours*, efisiensi, *linear programming*, *Artificial Neural Network*

# MANHOURS PLANNING OF C-CHECK MAINTENANCE ON PT. GMF AEROASIA BASE MAINTENANCE UNIT

**Name Of Student** : Rita Ummi Hanik  
**NRP** : 2506 100 131  
**Department** : Teknik Industri FTI-ITS  
**Supervisor** : Ir. Budi Santosa, M.S., Ph.D.  
**Co-Supervisor** : Nani Kurniati, S.T., M.T.

## Abstract

GMF Aero Asia is one of the leading airship service and maintenance company in Asia. It has always been this company's purpose to pursue the more efficient and effective ways to fulfill it's customers orders, and manhours efficiency is one of the factors that needs to be closely monitores in order to do that. Manhours efficiency level/rate is the ratio between planned manhours and actual manhours. Based on the historial data, manhours efficiency level/rate had reach 249% for C-Check maintenance at Base Maintenance Unit, which indicates a need of evaluation on the manhours planning. Currently manhours planning were done by giving a multiplying factor to the Maintenance Planning Data (MPD). The multiplying factor is determined by the planner subjectively based on their field experience and that there is no standard procedure to do so. This brought problems for the company because it often generates differently among planners. This research aims to generate a standard multiplying factor by using linear programming methods. Further on, the standard multypling factor will be tested it's performance by estimating the actual manhours with Artificial Neural Network. The results shows that the best alternative is alternative A, which improve the efficiency rate/level from 83% to 103%.

**Keywords:** MRO, *planning manhours*, efficiency, *linear programming*, *Artificial Neural Network*

## KATA PENGANTAR

Puji syukur saya panjatkan kepada Allah SWT atas berkat dan rahmat-Nya, sehingga laporan penelitian Tugas Akhir ini dapat terselesaikan dengan baik. Laporan penelitian Tugas Akhir ini ditulis sebagai bukti telah dilakukannya kegiatan penelitian Tugas Akhir yang ditujukan untuk pemenuhan syarat lulus menempuh jenjang pendidikan S1, Jurusan Teknik Industri ITS Surabaya.

Ucapan terima kasih dan puji syukur, saya sampaikan kepada seluruh pihak yang telah membantu dalam penyelesaian dan kelancaraan selama penelitian Tugas Akhir ini, seperti :

1. Orang tua penulis (Bapak dan Ibu) serta saudara yang selalu memberikan doa dan motivasi selama ini.
2. Bapak Ir. Budi Santosa, M.S., Ph.D., selaku dosen pembimbing yang telah banyak sekali memberikan pengarahan, masukan, nasehat serta bimbingan dalam mengerjakan Tugas Akhir ini.
3. Ibu Nani Kurniati, S.T.,M.T., selaku dosen co-pembimbing yang telah banyak memberikan masukan yang berharga bagi pengerjaan Tugas Akhir ini.
4. Bapak Agung Triadi Budianda selaku *General Manager Corporate Strategic Resources Management (TDC)*, Pak Irfan dan Mas Rendra selaku pembimbing di perusahaan yang telah banyak membantu, memberikan masukan dan ilmu dalam pelaksanaan penelitian di PT. GMF AA.
5. Bapak Iwan Joeniarto selaku *General Manager Base Maintenance Support (TBS)* dan seluruh elemen unit TBS-6 dan TBS-7 terutama Pak Winarso, Pak Bambang, Pak Tito, Pak Zahrul, Pak Tri, dan Mbak Novi atas ilmu dan bantuan yang diberikan selama di PT. GMF AA.
6. Bapak Yudha Prasetyawan, S.T., M.Eng., Bapak Ir. M. Suf, M.Sc., dan Bapak Aditya Sudiarno, S.T., M.T. selaku dosen penguji yang telah memberikan evaluasi dan saran membangun dalam penelitian Tugas Akhir ini.

7. Ibu Dr. Ir. Sri Gunani Partiw, M.T., selaku Ketua Jurusan Teknik Industri ITS.
8. Ibu Syarif Hanoum, S.T., M.T., selaku koordinator Tugas Akhir Jurusan Teknik Industri ITS.
9. Ganies Soebintang dan Nurul A. Lamik, atas bantuan dan dukungan yang telah diberikan selama ini.
10. Teman sesama peserta program magang PT.GMF AA (Mas Himawan, Mas Andik, Mas Teguh, Mas Wirawan, Mas Irawan, Mas Putu, Amel, dan Mbak Fitri) untuk semua kebersamaan dan motivasi yang diberikan.
11. Teman-teman angkatan 2006 Teknik Industri ITS (Nanda Kiswanto selaku komting TI'06, Shinta Windiani, Wilda, Kristika dkk) dimana telah memberikan banyak hal selama menjalani perkuliahan di Teknik Industri.
12. Himpunan Mahasiswa Teknik Industri (HMTI-ITS), sebagai tempat menempa *soft skill* dan "*secondary hard skill*" serta teman-teman fungsionaris HMTI ITS 2008-2009 (Nanda, Yogi, Susan, Andiani, Anta, Agra, Arie, Mahfud, Abi Laksono, Aditya Hadi, Fara, dan Lia).
13. Rekan-rekan administrator Laboratorium Komputasi dan Optimasi Industri (Fahmi, Agra, Nita, Satya, Kholis, Hardy, Irmaduta, Gladiez, Guntur, Eka Angga, dan Mas Putu) atas dukungan yang diberikan.
14. Seluruh pihak yang tidak dapat disebutkan semua dalam tulisan ini, dimana telah membantu dalam penyelesaian tugas akhir ini.

Semoga Laporan Tugas Akhir ini dapat bermanfaat dan dapat digunakan sebagaimana mestinya oleh pihak yang berkepentingan.

Terima Kasih.

Surabaya, Januari 2010

**Rita Ummi Hanik**  
(2506.100.131)

## DAFTAR ISI

HALAMAN JUDUL	
LEMBAR PENGESAHAN	
Abstrak.....	i
Abstract.....	iii
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	5
1.3 Tujuan Penelitian.....	6
1.4 Manfaat Penelitian.....	6
1.5 Ruang lingkup Penelitian.....	6
1.6 Sistematika Penulisan.....	6
BAB II TINJAUAN PUSTAKA.....	9
2.1 <i>Maintenance</i> .....	9
2.2 <i>Artificial Neural Network (ANN)</i> .....	14
2.3 <i>Productivity, Utility, Efficiency (PEU)</i> .....	22
2.4 Uji Hipotesis Komparatif.....	24
2.4.1 <i>Paired-t sample t-test</i> .....	24
2.5 <i>Operational Research</i> .....	25
2.7 Penelitian Terdahulu.....	27
BAB III METODOLOGI PENELITIAN.....	29
3.1 <i>Flowchart</i> Penelitian.....	29
3.2 Penjelasan <i>Flowchart</i> Penelitian.....	30
3.2.1 Tahap Awal.....	30
3.2.2 Tahap Pengumpulan Data.....	31
3.2.3 Tahap Pengolahan Data.....	32
3.2.4 Tahap Analisis.....	35

3.2.5	Tahap Kesimpulan dan Saran .....	36
<b>BAB IV PENGUMPULAN DAN PENGOLAHAN DATA .... 37</b>		
4.1	Pengumpulan Data .....	37
4.1.1	<i>Data Sample Project C-Check Maintenance B 747 ..</i>	37
4.1.2	<i>Maintenance Planning Data (MPD) .....</i>	39
4.1.3	<i>Data Master Planning Manhours .....</i>	40
4.1.4	<i>Data Actual Manhours .....</i>	40
4.2	Pengolahan Data .....	41
4.2.1.	<i>Linear Programming (LP) .....</i>	41
4.2.2.	<i>Artificial Neural Network (ANN) .....</i>	49
<b>BAB V ANALISIS DAN INTERPRETASI DATA..... 61</b>		
5.1	Analisis Hasil <i>Linear Programming (LP)</i> .....	61
5.2	Analisis Hasil Artificial Neural Network (ANN) .....	63
<b>BAB VI KESIMPULAN DAN SARAN ..... 69</b>		
6.1	Kesimpulan .....	69
6.2	Saran .....	70
<b>DAFTAR PUSTAKA .....</b>		<b>71</b>
<b>LAMPIRAN.....</b>		<b>73</b>
<b>Biodata Penulis</b>		

## DAFTAR GAMBAR

Gambar 1.1 <i>Actual revenue</i> unit <i>Base Maintenance</i> .....	2
Gambar 2.1 Model Neuron ANN (Santosa, 2007) .....	15
Gambar 2.2 Grafik Fungsi Threshold .....	17
Gambar 2. 3 Grafik Fungsi Linear-piecewise.....	17
Gambar 2. 4 Grafik Fungsi Sigmoid.....	18
Gambar 2. 5 Contoh Diagram <i>Back Propagation Network</i> .....	20
Gambar 2. 6 <i>Operational Research Category</i> .....	26
Gambar 3.1 <i>Flowchart</i> penelitian .....	29
Gambar 3.1 <i>Flowchart</i> penelitian (lanjutan).....	30
Gambar 4.1 Prosentase hasil <i>paired-t test</i> kondisi <i>existing</i> .....	43
Gambar 4.2 Prosentase selisih <i>plan</i> dan <i>actual</i> kondisi <i>existing</i> .....	43
Gambar 4.3 Prosentase hasil <i>paired-t test</i> alternatif terbaik .....	59
Gambar 4.4 Prosentase selisih <i>plan</i> dan <i>actual</i> alternatif terbaik .....	59
Gambar 5.1 Perbandingan selisih faktor pengali <i>actual</i> dan <i>planning</i> .....	61
Gambar 5.2 Perbandingan nilai faktor pengali .....	62
Gambar 5.3 Perbandingan <i>manhours</i> .....	65
Gambar 5.4 Perbandingan selisih <i>manhours</i> .....	66
Gambar 5. 5 Perbandingan kondisi <i>existing</i> dengan alternatif A....	67
.....	67

(halaman ini sengaja dikosongkan)

12	Gambar 1.1. Aliran berawal dari arus Mainw...
13	Gambar 1.2 Model Neron ANI (Neron, 2007)
14	Gambar 1.3 Grafik Fungsi Transfer
17	Gambar 1.4 Grafik Fungsi Transfer
18	Gambar 1.5 Grafik Fungsi Transfer
20	Gambar 1.6 Contoh Diagram Blok Pengontrolan Motor
26	Gambar 1.7 Koneksi Kelemahan Kelemahan
28	Gambar 1.8 Fungsi Transfer
30	Gambar 1.9 Fungsi Transfer (lanjutan)
31	Gambar 1.10 Fungsi Transfer
37	Gambar 1.11 Fungsi Transfer
37	Gambar 1.12 Fungsi Transfer
39	Gambar 1.13 Fungsi Transfer
39	Gambar 1.14 Fungsi Transfer
41	Gambar 1.15 Fungsi Transfer
41	Gambar 1.16 Fungsi Transfer
43	Gambar 1.17 Fungsi Transfer
43	Gambar 1.18 Fungsi Transfer
44	Gambar 1.19 Fungsi Transfer
44	Gambar 1.20 Fungsi Transfer
47	Gambar 1.21 Fungsi Transfer

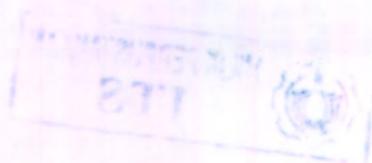
## DAFTAR TABEL

Tabel 1.1 <i>Actual revenue</i> unit <i>Base Maintenance</i> .....	2
Tabel 2.1 <i>Aircraft Maintenance Check Schedule (Typical A/L Example)</i> .....	13
Tabel 2.2 Contoh hubungan antara <i>Check Type</i> dengan Interval	14
Tabel 2.3 Jenis uji statistik.....	24
Tabel 2.4 Perbandingan antara metode estimasi yang digunakan dalam penelitian sebelumnya.....	28
Tabel 4.1 Rekap data <i>barcoding</i> pada project B747.....	38
Tabel 4.2 <i>Record</i> data <i>barcoding</i> pada project B747.....	38
Tabel 4.3 Data MPD ( <i>J/C No, Interval, Manhours</i> ).....	39
Tabel 4.4 Data <i>master planning manhours</i> .....	40
Tabel 4.5 Data <i>actual manhours</i> .....	40
Tabel 4.6 Komposisi <i>routine jobcard</i> .....	41
Tabel 4.7 Rekap hasil selisih dan <i>paired-t test</i> antara <i>planning</i> dan <i>actual</i> .....	42
Tabel 4.8 <i>Decision variable</i> .....	44
Tabel 4.9 <i>Constrain</i> nilai faktor pengali MPD.....	45
Tabel 4.10 Faktor pengali hasil <i>running</i> LINDO.....	46
Tabel 4.11 Performansi MSE dengan metode ANN.....	55
Tabel 4.12 Performansi MSE dengan kombinasi baru.....	56
Tabel 4.13 Perbandingan total <i>manhours</i> dan hasil <i>paired-t test</i> antara kondisi <i>existing</i> dan alternatif.....	57
Tabel 4.14 Perbandingan <i>efficiency</i> kondisi <i>existing</i> dengan alternatif A.....	58
Tabel 4.15 Selisih faktor pengali dan hasil <i>paired-t test</i> pada alternatif terbaik.....	58



(halaman ini sengaja dikosongkan)

1	Tabel 1.1. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
11	Tabel 1.2. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
13	Tabel 1.3. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
14	Tabel 1.4. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
24	Tabel 1.5. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
24	Tabel 1.6. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
28	Tabel 1.7. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
38	Tabel 1.8. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
38	Tabel 1.9. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
39	Tabel 1.10. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
40	Tabel 1.11. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
40	Tabel 1.12. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
41	Tabel 1.13. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
41	Tabel 1.14. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
42	Tabel 1.15. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
44	Tabel 1.16. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
45	Tabel 1.17. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
48	Tabel 1.18. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
52	Tabel 1.19. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
55	Tabel 1.20. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
55	Tabel 1.21. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
57	Tabel 1.22. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
58	Tabel 1.23. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi
58	Tabel 1.24. Kisi-kisi instrumen untuk mengukur kemampuan literasi numerasi



# BAB I

## PENDAHULUAN

Bab ini merupakan pendahuluan dari penelitian yang berisi tentang latar belakang masalah, perumusan masalah, tujuan penelitian, manfaat penelitian, ruang lingkup penelitian dan sistematika penulisan.

### 1.1 Latar Belakang

Industri penerbangan mengalami perkembangan yang sangat pesat, hal ini menyebabkan meningkatnya kompetisi perusahaan-perusahaan *Maintenance, Repair, and Overhaul (MRO)*. PT Garuda *Maintenance Facility Aero Asia* (PT. GMF AA), yang merupakan anak perusahaan dari PT. Garuda Indonesia adalah salah satu perusahaan MRO berskala Internasional yang selalu melakukan *improvement* untuk mendapatkan *revenue* yang telah ditargetkan setiap tahunnya dan mampu bersaing di pasar Internasional.

PT. GMF AA beroperasi dalam bidang perawatan dan perbaikan pesawat, mulai dari perawatan *line maintenance, engine maintenance, component maintenance, cabin refurbishment, redelivery*, sampai dengan *overhaul*. Untuk dapat memberikan pelayanan tersebut, PT GMF AA didukung oleh beberapa unit produksi, diantaranya *Line Maintenance (TL), Base Maintenance (TB), Componen Maintenance (TC), Engine Maintenance (TR)* dan *Trade and Asset Management*.

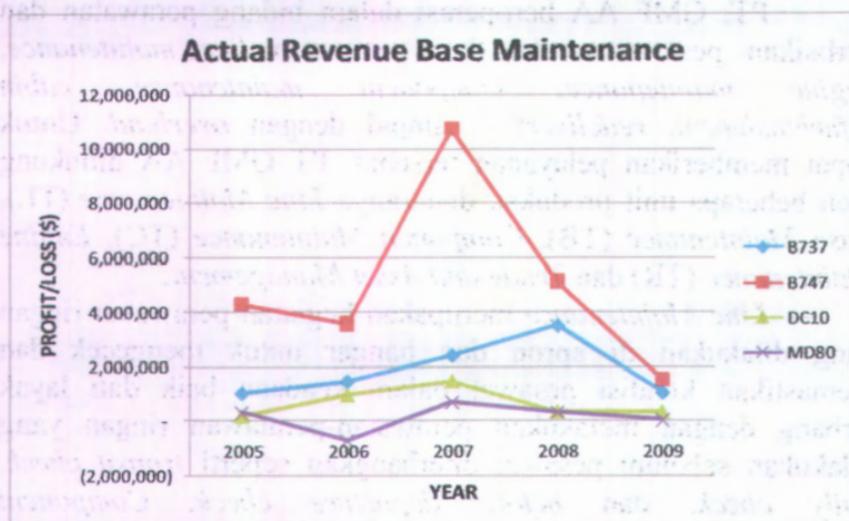
*Line Maintenance* merupakan kegiatan perawatan ringan yang dilakukan di apron dan hangar untuk mengecek dan memastikan kondisi pesawat dalam keadaan baik dan layak terbang dengan melakukan perawatan-perawatan ringan yang dilakukan sebelum pesawat diterbangkan seperti *transit check, daily check*, dan *before departure check*. *Component Maintenance* merupakan kegiatan perawatan pada komponen-komponen yang meliputi AC, electric, electronic, dll. *Engine Maintenance* merupakan kegiatan perbaikan pada *engine* yang

meliputi kegiatan *overhaul*, *repair*, dan *split build up*. Kegiatan tambahan terdiri dari kegiatan pelengkap untuk solusi perawatan terpadu (*total maintenance solution*), yaitu jasa *engineering*, dan jasa material logistik.

*Base Maintenance* merupakan kegiatan perawatan yang dilakukan di hangar yang meliputi kegiatan perawatan struktur *aircraft*, *overhaul*, dan *repair* yang biasanya dikenal dengan istilah *C-Check* dan *D-Check*. Dari data *all project* tahun 2005 hingga bulan Juni tahun 2009 untuk pesawat Non-Garuda, diketahui bahwa *revenue* unit ini dapat dilihat pada Tabel 1.1 dan Gambar 1.1.

Tabel 1.1 Actual revenue unit Base Maintenance (\$)

A/C TYPE	YEAR				
	2005	2006	2007	2008	2009
B737	1,001,403	1,447,196	2,405,337	3,511,192	1,030,203
B747	4,302,906	3,602,788	10,715,406	5,124,824	1,563,691
DC10	161,815	954,813	1,470,300	341,200	328,066
MD80	296,654	(752,575)	688,611	311,581	27,161



Gambar 1.1 Actual revenue unit Base Maintenance

Dari Tabel 1.1 Gambar 1.1 diketahui bahwa tipe pesawat yang memberikan pendapatan tertinggi selama lima tahun terakhir adalah pesawat Non-Garuda dengan tipe B 747. Untuk meningkatkan *revenue* di setiap tahunnya, maka perlu dilakukan evaluasi dan *continous improvement* dalam proses bisnis yang dilakukan. Renoir Group merupakan salah satu konsultan terkemuka di dunia dan PT. GMF AA menggunakan jasa Konsultan Renoir ini untuk melakukan *improvement*. Salah satu hasil analisis Konsultan Renoir di unit *Base Maintenance* untuk jenis pesawat Max Air 5N-BMG (Boeing 747-200) dengan tipe *maintenance* C-check yang dilakukan pada tanggal 5 sampai dengan 25 September 2009 diketahui bahwa *total plan hours* (PL) yakni total *manhours* yang direncanakan untuk menyelesaikan proyek ini adalah 3.402,99 *manhours*. *Total available manhours* (AV) yakni total *manhours* yang tersedia untuk menyelesaikan proyek ini adalah 1.803,00 *manhours*. Sedangkan *total actual manhours* (AC) yakni *total actual hours* yang dihabiskan untuk menyelesaikan proyek adalah 1.365,97 *manhours*. Sehingga nilai performansi P.E.U nya adalah sbb :

- a) *Productivity* =  $PL/AV = 3.402,99 / 1.803,00 = 189 \%$
- b) *Efficiency* =  $PL/AC = 3.402,99 / 1.365,97 = 249 \%$
- c) *Utilization* =  $AC/AV = 1.365,97 / 1.803,00 = 76 \%$

Nilai PEU ini merupakan nilai yang berorientasi pada kualitas perencanaan *manhours* terhadap suatu *maintenance task*, bukan merupakan kualitas dari suatu *jobcard* tertentu. Nilai *productivity* dan *efficiency* pada kasus tersebut cukup besar dan terdapat dua kemungkinan dalam hal ini, yaitu kemungkinan mekanik PT. GMF AA sangat handal sehingga bisa menyelesaikan pekerjaan lebih cepat dari *planning*-nya. Sedangkan kemungkinan yang kedua yakni adanya kesalahan *planning* dalam penentuan *manhours* di masing-masing *jobcard* yang dikerjakan. Dari proyek *maintenance* 5N-BMG ini bisa diketahui bahwa sebenarnya mekanik mampu menyelesaikan pekerjaan yang diberikan lebih cepat dari perencanaannya. Namun hal seperti ini jarang terjadi karena terdapat suatu

anggapan apabila mekanik menyelesaikan pekerjaan yang ada lebih cepat dari perencanaannya, maka hasil pekerjaan tersebut akan diragukan prosedur dan kualitasnya. Selain itu, tidak terdapat *reward* tertentu yang diberikan kepada mekanik ketika mereka mampu mengerjakan pekerjaan lebih cepat dari *manhours* yang direncanakan oleh *planner* dengan hasil kualitas pekerjaan yang baik. Hal ini yang menyebabkan perilaku mekanik secara umum menjadikan *planning manhours* sebagai acuan, mereka cenderung menyamakan dengan *planning* atau sesuai dengan waktu aktual mereka jika kondisi aktualnya menghabiskan waktu lebih banyak dari *planning manhours* yang ada. Kenyataan ini merupakan efek dari *hidden safety*, *student syndrome* dan *Parkinson's Law* (Santosa, 2007). Melihat fenomena seperti itu, maka perlu dilakukan evaluasi dan *improvement* terhadap *planning manhours*-nya.

Pekerjaan *maintenance* meliputi pekerjaan rutin dan pekerjaan non-rutin. *Base Maintenance Support* (TBS) merupakan bagian dari unit TB yang bertugas menentukan perencanaan *manhours* untuk pekerjaan rutin dan pekerjaan non-rutin pada tipe *maintenance C-Check* dan *D-Check* pesawat Non-Garuda. Selama ini, *planner* membuat *planning manhours* dengan mengacu terhadap MPD (*Maintenance Planning Data*) yang dikeluarkan oleh pabrik pesawat. Jika jenis pesawat Boeing, maka yang memiliki otoritas mengeluarkan MPD adalah FAA. Sedangkan untuk jenis pesawat Airbus, maka yang memiliki otoritas mengeluarkan MPD adalah EASA/French DGCA. Perencanaan *manhours* yang ditentukan oleh *planner* merupakan hasil perkalian MPD dengan *faktor* pengali tertentu seperti 2, 2.5, 2.75, dsb. Sedangkan untuk *non-routine jobcard* juga diberikan faktor pengali tertentu terhadap standard *Corrosion Prevention and Control Programme (CPCP)* dan atau *Airworthiness Directive and Service Bulletins (AD/SB)*, namun faktor pengali ini tidak terlalu besar dan cenderung sama dengan standard *manhours* yang ada. *Non-routine jobcard* ini tidak selalu

dilakukan dalam setiap *event maintenance*, pekerjaan-pekerjaan tersebut dilakukan jika terdapat permintaan dari *customer*.

*Manhours* yang ada pada MPD menunjukkan *manhours* yang dikonsumsi untuk menyelesaikan suatu *jobcard* tertentu dalam kondisi lingkungan kerja yang ideal, dimana semua peralatan, objek *maintenance* dan semua kebutuhan *maintenance* lainnya sudah tersedia dan siap untuk dikerjakan. Pada kenyataannya di perusahaan, kondisi ideal ini sulit dicapai sehingga perlu dibuat faktor pengali untuk memberikan estimasi dalam penyelesaian suatu *jobcard*. Di perusahaan, penentuan nilai faktor pengali berdasarkan nilai MPD pada *jobcard* yang bersangkutan. Misalnya untuk *jobcard* A dan *jobcard* B memiliki perbedaan jenis pekerjaan namun memiliki nilai MPD yang sama, maka akan diberikan nilai faktor pengali yang sama. Namun, besarnya nilai faktor pengali tersebut memiliki sifat subjektifitas yang tinggi sehingga kemungkinan terjadi kesalahan perencanaan cukup besar. Dikatakan memiliki unsur subjektifitas karena penentuannya berdasarkan pengalaman *planner* yang sebagian besar adalah mantan mekanik, sehingga faktor pengali yang dibuat oleh *planner* yang satu dengan yang lain masih dijumpai perbedaan. Hal ini yang mendasari adanya kebutuhan penentuan nilai faktor pengali baru yang dijadikan standard referensi bagi *planner* agar mampu memperbaiki kesalahan *planning*, mampu meningkatkan *revenue* perusahaan dan mampu meningkatkan kompetisi perusahaan di industri MRO.

## 1.2 Perumusan Masalah

Berdasarkan uraian latar belakang tersebut di atas maka dapat dirumuskan permasalahan yang akan diteliti adalah bagaimana menentukan standard faktor pengali untuk proses pengerjaan *C-Check maintenance* pada tipe pesawat Boeing 747 Classic dengan menerapkan metode *Linear Programming(LP)* dan *Artificial Neural Network (ANN)* dimana data yang diperoleh berdasarkan data *manhours* MPD, *manhours planning* dan

## BAB V ANALISIS DAN INTERPRETASI DATA

Berisi analisis dan interpretasi data dari pengolahan data yang telah dilakukan pada tahap sebelumnya.

## BAB VI KESIMPULAN DAN SARAN

Berisi penarikan kesimpulan dan pemberian saran kepada perusahaan serta untuk kemajuan penelitian selanjutnya.

## BAB II

### TINJAUAN PUSTAKA

Pada bab ini akan dibahas mengenai konsep dan teori yang digunakan sebagai acuan proses pemecahan masalah dalam penelitian ini.

#### 2.1 *Maintenance*

Menurut (Dhillon, 1895) dalam Sugiharto.RM, 2007, perawatan adalah suatu kombinasi dari berbagai tindakan yang dilakukan untuk menjaga suatu barang atau memperbaiki suatu kondisi yang bisa diterima. Sedangkan menurut British Standard Institute (BS 3811,1974), perawatan adalah kombinasi dari beberapa tindakan yang ditujukan untuk mempertahankan kinerja fasilitas atau mesin.

Dari definisi di atas, dapat ditarik kesimpulan bahwa perawatan mempunyai kaitan yang erat dengan tindakan pencegahan dan pembaharuan. Dalam perawatan, tindakan-tindakan yang dapat dilakukan antara lain ;

1. Pemeriksaan, merupakan tindakan yang ditujukan terhadap sistem untuk mengetahui apakah sistem masih berada dalam keadaan yang memenuhi persyaratan yang diinginkan.
2. Penggantian komponen, yaitu tindakan penggantian komponen sistem yang sudah tidak berfungsi dimana tindakan penggantian komponen system dilakukan dapat bersifat terencana dan tidak terencana.
3. *Repair* dan *Overhaul*, yaitu melakukan pemeriksaan secara cermat serta melakukan perbaikan dimana dilakukan *set-up* ulang.
4. Penggantian sistem, yaitu tindakan yang diambil apabila tindakan-tindakan yang lain sudah tidak memungkinkan lagi.

Fungsi perawatan adalah memperbaiki mesin atau peralatan yang rusak dan menjaga agar selalu dalam kondisi siap

dioperasikan. Bagian perawatan berkaitan erat dengan proses produksi karena kegagalan kegiatan perawatan sangat mengganggu kelancaran proses produksi. Dengan adanya kegiatan perawatan yang baik dan efektif akan mencegah timbulnya *breakdown* pada waktu yang telah diperkirakan terlebih dahulu.

Aktivitas pemeliharaan suatu fasilitas atau mesin produksi yang dilakukan dapat dibedakan menjadi dua yaitu:

❖ Perawatan Tidak Terencana (*Unplanned Maintenance*)

Merupakan perawatan yang tidak direncanakan terlebih dahulu, disebabkan peralatan atau fasilitas produksi tidak memiliki rencana serta jadwal perawatan. Kegiatan perawatan ini disebut juga perawatan darurat (*breakdown maintenance* atau *emergency maintenance*) yang didefinisikan sebagai perawatan yang perlu dilaksanakan tindakan untuk mencegah akibat yang fatal seperti kerusakan besar peralatan, hilangnya produksi, dan keselamatan kerja.

❖ Perawatan Terencana (*Planned Maintenance*)

Merupakan kegiatan perencanaan yang mengacu pada rencana yang telah disusun dan dilaksanakan serta didokumentasikan. Perawatan ini terbagi 2 yaitu :

1. Perawatan Pencegahan (*Preventive Maintenance*)

Kegiatan pemeliharaan dan perawatan untuk mencegah timbulnya kerusakan-kerusakan tidak terduga dan menemukan kondisi atau keadaan yang menyebabkan fasilitas produksi mengalami kerusakan pada waktu proses produksi dan mencegah menurunnya fungsi peralatan dan fasilitas. Perawatan ini dibagi menjadi 2 yaitu:

a) Perawatan Rutin

Perawatan rutin adalah kegiatan pemeliharaan dan perawatan yang dilakukan secara rutin setiap hari yaitu dengan pembersihan peralatan, pelumasan, pengecekan oli, dan pengecekan bahan bakar.

## b) Perawatan Periodik

Perawatan periodik merupakan kegiatan pemeliharaan yang dilakukan secara periodik atau dalam jangka waktu tertentu seperti memeriksa komponen-komponen peralatan.

### 2. Perawatan Perbaikan

Kegiatan perawatan yang sudah direncanakan berupa penggantian komponen yang sudah tidak berfungsi. Perawatan perbaikan dapat berupa perbaikan yang tidak ditemukan pada saat pemeriksaan seperti penggantian komponen secara serentak juga *overhaul* yang terencana.

Pada mulanya *maintenance* pada diartikan sebagai *fixed it when broke*, artinya diperbaiki ketika rusak saja. Ketika kebutuhan terhadap faktor *safety* mulai diperhitungkan, maka muncul *preventive maintenance*, dengan sistem *overhaul* yang dikenal dengan istilah *hardtime (HT)*. *Hardtime* ini dilakukan ketika suatu komponen sudah mencapai umurnya untuk diganti dan tidak memperdulikan kondisi komponen seandainya kondisinya masih bagus. *Hardtime* ini membuat maskapai penerbangan mengeluarkan banyak biaya untuk perawatan.

Ketika B 747 akan di-*release*, maskapai penerbangan dituntut untuk membuat suatu konsep *maintenance* yang lebih jelas dan lebih menghemat biaya, sehingga terbentuk *Maintenance Steering Group 1 (MSG-1)* yang merumuskan konsep *maintenance program* untuk B 747. Dalam konsep ini dikenal juga istilah *On Condition (OC)* yang merupakan suatu inspeksi berkala untuk menentukan apakah komponen tersebut masih bisa berfungsi atau tidak. Konsep MSG-1 ini masih memiliki banyak kekurangan, sehingga dibentuk MSG-2. Konsep yang dihasilkan kemudian diterapkan pada pesawat yang lebih banyak dimulai pada DC-10 dan beberapa pesawat yang masih beroperasi hingga sekarang seperti B737CL dan MD80.

Pada konsep MSG-2 ini dikenal juga istilah *Condition Monitoring (CM)*, CM ini tidak seperti HT atau OC yang merupakan *preventive maintenance*, tetapi berupa pengecekan apabila komponen bermasalah. Salah satu pemantaunya adalah

dengan istilah *Mean Time Between Unscheduled Removals (MTBUR)*. Komponen yang termasuk dalam CM dimonitor umurnya, apabila mendekati perkiraan umurnya maka harus siap dilakukan penggantian. Namun tidak ada usia wajib untuk menggantikan komponen tersebut atau melakukan pengecekan kondisinya.

Sekitar tahun 78, *United Airlines (UAL)* bersama *Department of Defense (DoD)* Amerika bekerjasama untuk menyusun suatu konsep *maintenance management* yang lebih baik yaitu RCM. Karena konsep yang dibuat lebih baik dari MSG-2, maka mulai dibakukan di dunia penerbangan yang dikenal sebagai MSG-3. Jika MSG-2 dikenal dengan istilah *process oriented* karena untuk suatu komponen ditentukan berdasarkan tipe *maintenance* yang berupa prosesnya saja (HT, OC, dan CM), maka MSG-3 dikenal dengan istilah *task oriented* dimana bentuk *maintenance* yang dilakukan langsung berdasarkan tipe pekerjaannya seperti *servicing, lubrication, cleaning, replace* dan *discard* sehingga pada MSG-3 ini, konsep HT, OC, dan CM tidak digunakan lagi.

Saat ini banyak *maintenance program* yang disusun berdasarkan MSG-3, dimana pada konsep ini tidak ada lagi istilah A-check, C-check, dsb yang biasa dikenal dengan istilah *letter check* atau *block maintenance*. Semua *taskcard* hanya diberikan interval berdasarkan *Flight Hour (FH)*, *Flight Cycle (FC)*, atau waktu kalender (DY = *Day*, MO = *Month*, YR = *Year*). *Maintenance check* ini terbagi dalam 3 kategori:

1. *Block Maintenance*, atau istilah yang dikenal dengan sebutan *Letter Check (ABCD Check)*. Pengelompokan ini berasal dari pabrik, setiap *taskcard* dikelompokkan langsung menjadi kelompok masing-masing. A-Check biasanya terdiri dari A1, A2, A3, sampai dengan A8. C-check terdiri dari C1, C2 sampai dengan C8. *Maintenance program* yang dibuat dengan konsep MSG-2 biasanya menggunakan *letter check*. Tabel 2.1 merupakan contoh *aircraft maintenance check schedule* dan Tabel 2.2 merupakan contoh hubungan antara *check type* dengan *interval*.

Tabel 2.1 *Aircraft Maintenance Check Schedule (Example)*

	747-400	747-200/300	DC-10-30	A330B4	F50
<b>Transit Check</b>	At each stop whenever aircraft is in transit				
<b>Daily Check</b>	Before first flight or whenever aircraft is on ground more than 4 hours				
<b>"A" Check</b>	Every 600 FH	Every 500 FH or 7 Weeks	In 3 parts A1, A2, A3 465 FH or 9 weeks	In 4 parts A1, A2, A3, A4 Every 385 FH or 11 weeks	Every 650 FH or 4 months
<b>"B" Check</b>	In 2 Parts B1, B2 Every 1200 FH	In 2 Parts B1, B2 Every 1000 FH	None	None	Every 1300 FH or 8 months
<b>"C" Check</b>	In 2 Parts C1, C2 Every 5000 FH or 18 months	every 4650 FH or 24 months	In 2 Parts C1, C2 Every 4500 FH or 20 months	In 2 Parts C1, C2 Every 3000 FH or 18 months	In 2 Parts C1, C2 Every 4000 FH or 25 months
<b>"D/HMV" check</b>	First Check done between 25K & 27.5K FH Subsequent every 25K FH or 6 years	First Check at 25K FH or 6 years Subsequent every 20K FH or 5 years	Every 20 K FH or 6 years	Every 12 K FH or 4 years	In 2 Parts H1, H2 Every 12K FH or 6 years

Tabel 2.2 Contoh hubungan antara *Check Type* dengan Interval

Check Type	Equal value
A1	1A
A2	1A + 2A
A3	1A + 3A
A4	1A + 2A + 4A
A5	1A + 5A
A6	1A + 2A + 3A + 6A
A7	1A + 7A
A8	1A + 2A + 4A + 8A

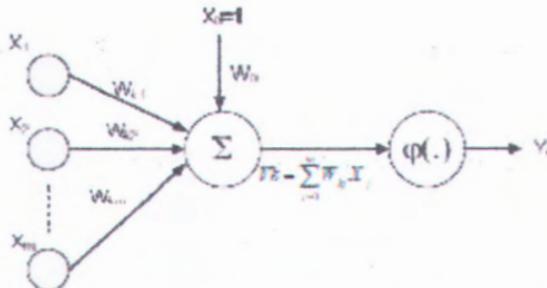
2. *Progressive Maintenance*, setiap *taskcard* dikerjakan berdasarkan intervalnya sendiri yang biasanya dikenal dengan *day by day maintenance*. Pengerjaan dengan sistem ini lebih rumit, dan butuh upaya yang lebih fokus karena yang dimonitor adalah *taskcard* bukan kelompok. Beberapa maskapai penerbangan ada yang menggunakan ini untuk maintenance programnya karena menggunakan MSG3.
3. *Equalized maintenance*, sistem ini dikenal dengan sebutan *phase check*. Seluruh *maintenance check* untuk pesawat tersebut dilist dan dihitung *workload*-nya, lalu dibagi menjadi beberapa *phase* dengan interval yang tetap. Interval yang bisa dipilih meliputi FH, FC, atau kalender.

## 2.2 Artificial Neural Network (ANN)

*Artificial neural networks* (ANN) atau jaringan saraf buatan, selanjutnya disebut ANN, awalnya mendapat inspirasi dari sistem jaringan saraf makhluk hidup. Observasi bahwa sistem belajar dari makhluk hidup terutama manusia terdiri dari jaringan yang sangat kompleks yang terdiri dari neuron yang saling terhubung telah memberi inspirasi ini. Dalam ANN, jaringan saraf makhluk hidup ingin ditiru susunannya. ANN muncul sebagai alternatif pendekatan konvensional yang biasanya kurang fleksibel terhadap perubahan struktur masalah. ANN menawarkan kelebihanannya dimana bisa mengatasi beberapa persoalan tanpa mengadakan perubahan drastis terhadap modelnya.

*Artificial Neural Network* (ANN), dengan kemampuannya yang sangat baik untuk mendapatkan *meaning* di dalam data yang kompleks, dapat digunakan untuk mengekstraksi pola dan mendeteksi trends yang terlalu kompleks untuk dapat dikenali oleh manusia maupun teknik-teknik yang lain. ANN yang telah terlatih dengan baik, dapat disamakan dengan seorang *experts*, yang kemudian dapat digunakan untuk menyediakan proyeksi bila diberikan data baru.

ANN terdiri atas komponen-komponen yang terstruktur dengan suatu kegiatan yang analogi dengan fungsi biologis neuron (sel saraf). Dengan serangkaian inputan di luar sistem yang diberikan kepadanya, jaringan ini dapat memodifikasi bobot yang akan dihasilkannya, sehingga akan menghasilkan output yang konsisten sesuai dengan input yang telah diberikan kepadanya. Maka dari itu respon jaringan ini akan mengabaikan perbedaan kecil dari inputan yang diberikan kepadanya sampai tingkat toleransi tertentu. Struktur jaringan ini ditunjukkan pada Gambar 2.1.



Gambar 2.1 Model Neuron ANN (Santosa, 2007)

Elemen-elemen dasar neuron (Santosa, 2007) adalah:

1. Set sinapsis atau *link* penghubung, ditandai dengan adanya bobot atau kekuatan dari *link* ini. Suatu signal  $x_j$  pada sinapsis  $j$  dihubungkan ke neuron  $k$  dikalikan dengan bobot  $w_{kj}$ . Perlu dicatat bagaimana indeks pada bobot sinapsis  $w_{kj}$  ini dituliskan. Indeks pertama yaitu  $k$



menunjukkan neuron dan indeks kedua  $j$  menunjukkan input ke berapa.

2. Penambah, yaitu untuk menjumlahkan signal input yang diberi bobot. Operasi ini adalah kombinasi linear.
3. Fungsi aktivasi (*activation function*) untuk membatasi besarnya output dari suatu neuron.

Pada Gambar 2.1 terdapat bias yang dinyatakan sebagai  $b_k$  atau  $w_{k0}$ . Bias mempunyai fungsi untuk menaikkan atau menurunkan net input untuk fungsi aktivasi, tergantung nilainya positif atau negatif. Neuron  $k$  bisa didiskripsikan secara matematis sebagai (Santosa, 2007):

$$u_k = \sum_{j=1}^p w_{kj} \cdot x_j \quad (2.1)$$

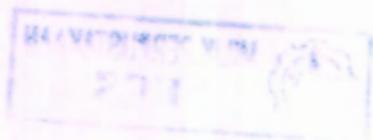
dan untuk kemudian hasil dari persamaan tersebut akan menjadi inputan bagi fungsi aktivasi untuk mendapatkan tingkat derajat signal keluaran pada neuron, dimana terdapat bermacam-macam jenis fungsi aktivasi.

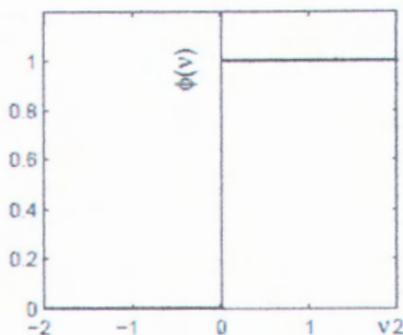
Jenis-jenis fungsi aktivasi  $\phi(\cdot)$  yang bisa dipakai dalam *neural network* adalah :

#### 1. Fungsi Threshold

Untuk fungsi ini kita punya dua output:

$$\phi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases} \quad (2.2)$$

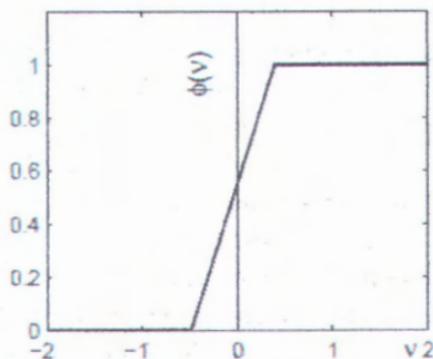




Gambar 2.2 Grafik Fungsi Threshold

## 2. Fungsi Linear Piecewise

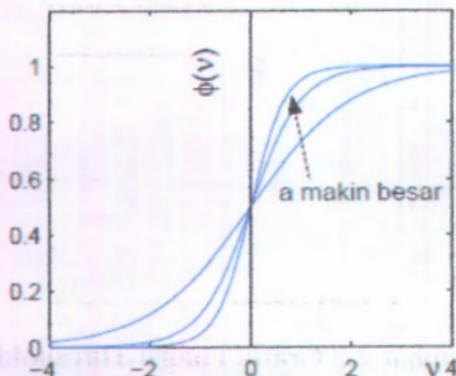
$$\varphi(v) = \begin{cases} 1, & v \geq \frac{1}{2} \\ v + \frac{1}{2}, & -\frac{1}{2} < v < \frac{1}{2} \\ 0, & v \leq -\frac{1}{2} \end{cases} \quad (2.3)$$



Gambar 2.3 Grafik Fungsi Linear-piecewise

## 3. Fungsi Sigmoid

$$\varphi(v) = \frac{1}{1 + \exp(-av)} \quad (2.4)$$



Gambar 2. 4 Grafik Fungsi Sigmoid

Ada beberapa syarat suatu fungsi bisa menjadi fungsi aktivasi  $\phi(\cdot)$ .

1. *Nonlinear*

Dengan memakai fungsi aktivasi yang tidak linier akan memperbaiki kemampuan *network* dalam melakukan tugasnya.

2. *Saturate*

Suatu fungsi dinamakan saturate bila mempunyai output dengan nilai minimum dan maksimum. Dengan demikian akan menjaga nilai bobot  $w$  dan bias  $b$  *bounded* (terbatas). Sehingga, waktu yang dibutuhkan untuk training menjadi terbatas juga.

3. Kontinuitas dan *smoothness*

Dengan syarat ini suatu fungsi aktivasi  $\phi(\cdot)$  dan  $\phi'(\cdot)$  terdefinisi dalam *range* dari argumennya. Ini sangat penting seperti dalam *back-propagasi* dimana kita memerlukan turunan  $\phi'(\cdot)$ .

Macam-macam fungsi aktivasi yang sering digunakan dalam ANN adalah sebagai berikut :

a. Fungsi Sigmoid

$$f(x) = \frac{1}{1+e^{-ax}}, \text{ dimana } a > 0 \quad (2.5)$$

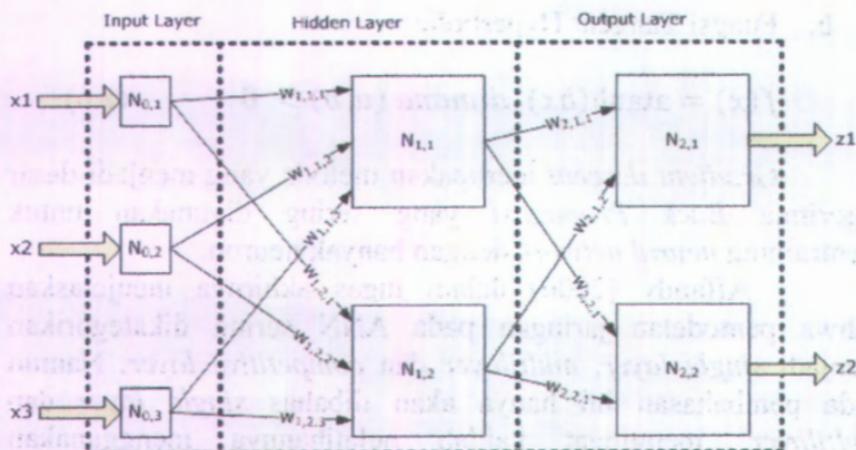
b. Fungsi Tangent Hyperbolic

$$f(x) = \operatorname{atanh}(bx), \text{ dimana } (a, b) > 0 \quad (2.6)$$

*Gradient descent* merupakan metoda yang menjadi dasar algoritma *Back Propagasi* yang sering digunakan untuk mentraining *neural network* dengan banyak neuron.

Affandy (2000) dalam tugas akhirnya menjelaskan bahwa pemodelan jaringan pada ANN sering dikategorikan menjadi *single layer*, *multilayer* dan *competitive layer*. Namun pada pembahasan ini hanya akan dibahas *single layer* dan *multilayer*, mengingat kaidah pelatihnannya menggunakan algoritma *backpropagation*. Secara umum, tiap unit pada *layer* (lapisan) yang sama (dalam pembahasan ini dapat disebut sebagai *neuron*) mempunyai tingkah laku yang sama untuk pemrosesan signal data. Hanya hal terpenting yang perlu diperhatikan ialah penentuan penggunaan jenis aktivasi pada masing-masing unit pada lapisan tersebut dan pola koneksi pembobotan antar lapisan. Namun biasanya unit pada *layer* yang sama mempunyai jenis fungsi aktivasi yang sama dan pola koneksi pembobot yang sama pula. Untuk pemilihan jumlah *layer* bukan berarti pemilihan *layer* untuk *neuron*, namun pemilihan *layer* untuk penghubung jalur pembobot antar *neuron*. Jadi variabel terpenting untuk pengenalan pola ialah pembobotnya.

Salah satu teknik untuk men-*training multi layer neural network* adalah algoritma *feed-forward neural network* atau sering juga disebut *back propagation network* (BPN). Algoritma ini sangat bermanfaat, cukup handal, mudah dipahami, dan banyak algoritma lain mendasarkan prosesnya pada BPN. Dari Gambar 2.5, dapat dilihat bahwa BPN terdiri dari tiga *layer*. *Layer* pertama disebut sebagai input, yang terdiri dari-tiga neuron. Untuk *layer* kedua disebut *hidden layer* yang terdiri dari dua neuron. Dan yang terakhir disebut *output layer* yang terdiri dari dua neuron.



Gambar 2. 5 Contoh Diagram *Back Propagation Network* (Beale & Jackson, 1990)

Terdapat beberapa catatan bahwa :

1. Output dari sebuah neuron di dalam sebuah layer, menjadi input bagi seluruh neuron di layer berikutnya.
2. Tiap-tiap neuron memiliki bobot input masing-masing.
3. Bobot pada input layer, diasumsikan bernilai satu untuk masing-masing input atau bisa dikatakan bahwa nilai dari input tidak berubah.
4. BPN paling tidak harus memiliki satu *input* dan satu *output layer*. Untuk *hidden layer* dapat berjumlah nol maupun lebih.

Berikut adalah ringkasan bagaimana *back propagation network* bekerja (Santosa, 2007) :

1. *Feedforward*

- Hitung signal input.

$$z_{in} = \text{bias} + \sum wx \quad (2.7)$$

- Terapkan fungsi aktivasi, dan hasilnya merupakan input bagi unit untuk layer berikutnya.

$$z_n = \varphi(z_{in}) \quad (2.8)$$

- Hitung signal input.

$$zz\_inj = v0j + v0j + \sum_{h=1}^q z_h v_{hj} \quad (2.9)$$

- Terapkan fungsi aktivasi, dan hasilnya digunakan untuk input bagi simpul output.

$$zzj = \varphi(zz\_inj) \quad (2.10)$$

- Jumlahkan signal output

$$y\_in = w0 + w_0 + \sum_{j=1}^p zzj w_j \quad (2.11)$$

- Terapkan fungsi aktivasi

$$y = \varphi(y\_in) \quad (2.12)$$

## 2. Back Propagation

- a. Untuk setiap unit Yk

$$e = d - y \quad (2.13)$$

- b. Untuk titik data yang sekarang

- Kalikan dengan turunan dari fungsi aktivasi.

$$\delta = e\varphi'(y\_in) \quad (2.14)$$

- Hitung perubahan bobot.

$$\Delta wj = \eta \delta zzj \quad (2.15)$$

- Hitung perubahan bias.

$$\Delta w0 = \eta \delta \quad (2.16)$$

- Dan  $\delta$  dikirim ke unit hidden ZZj, j = 1, ..., p.

- c. Untuk setiap unit hidden ZZj, j = 1, ..., p.

- Jumlahkan input dengan bobot dari layer di atas.

$$\delta\_inj = \sum_{j=1}^p \delta w_j \quad (2.17)$$

- Lalu dikalikan dengan turunan fungsi aktivasi, hasilnya

$$\delta_j = \delta\_inj \varphi'(zz\_inj) \quad (2.18)$$

- Hitung koreksi bobot untuk meng-update vij

$$\Delta vij = \eta \delta_j x_i \quad (2.19)$$

- Hitung koreksi bias untuk meng-update v0j

$$\Delta v0j = \eta \delta_j \quad (2.20)$$

- Dan masukkan  $\delta_j$  ke unit hidden ( $Zh'$  h = 1, ..., q)

- d. Untuk setiap unit hidden ( $Zh'$  h = 1, ..., q)

- a. Jumlahkan input berbobot dari unit-unit di layer di atas untuk mendapatkan

$$\Delta\_inh = \sum_{j=1}^p \delta v_{hj} \quad (2.21)$$

b. Kalikan dengan turunan fungsi aktivasinya  

$$\delta h = \delta_{inh} \phi'(z_{inh}) \quad (2.22)$$

- Hitung koreksi bobot untuk meng-update  $w_{ij}$   

$$\Delta w_{ij} = \eta \delta h x_i \quad (2.23)$$

- Hitung koreksi bias untuk meng-update  $v_{0j}$   

$$\Delta v_{0j} = \eta \delta h \quad (2.24)$$

### 3. Update untuk bobot dan bias

- Untuk setiap unit output ( $j = 0, \dots, p$ ;  $k = 1, \dots, m$ )  

$$W_{jk}(\text{new}) = w_{jk}(\text{old}) + \delta w_{jk} \quad (2.25)$$

- Untuk setiap unit hidden  $ZZ_j$  ( $h = 0, \dots, q$ ;  $j = 1, \dots, p$ )

$$h_j(\text{new}) = h_j(\text{old}) + \delta h_j \quad (2.26)$$

- Untuk setiap unit hidden  $Zh$  ( $i = 0, \dots, n$ ;  $h = 1, \dots, q$ )

$$ih(\text{new}) = ih(\text{old}) + \delta ih \quad (2.27)$$

## 2.3 Productivity, Efficiency, Utility (PEU)

Menurut Renoir Consultant (2009), produktifitas merupakan perbandingan antara total perencanaan manhours yang dihabiskan untuk menyelesaikan suatu pekerjaan terhadap total ketersediaan manhours. Efisiensi merupakan kemampuan untuk menyelesaikan suatu pekerjaan dengan waktu dan usaha seminimal mungkin, bisa juga didefinisikan sebagai perbandingan antara total perencanaan manhours yang dihabiskan untuk menyelesaikan suatu pekerjaan terhadap total manhours actual yang digunakan. Sedangkan Utilitas merupakan suatu pengaplikasian sumberdaya yang dimiliki untuk menyelesaikan suatu pekerjaan, dengan kata lain diartikan sebagai perbandingan antara total manhours actual yang digunakan untuk menyelesaikan suatu pekerjaan terhadap total ketersediaan manhours.

$$\text{Productivity} = PL/AV \quad (2.28)$$

$$\text{Efficiency} = PL/AC \quad (2.29)$$

$$\text{Utilization} = AC/AV \quad (2.30)$$

Keterangan :

PL : *Total plan hours for task*

AV: *Total available man hours*

AC: *Total actual man hours*

Produktifitas merupakan nisbah atau rasio antara hasil kegiatan (output, keluaran) dan segala pengorbanan (biaya) untuk mewujudkan hasil tersebut (input, masukan) (Kussriyanto, 1984, p.1). Input bisa mencakup biaya produksi dan biaya peralatan. Sedangkan output bisa terdiri dari penjualan, pendapatan, dan kerusakan (Gomes, 1995, p.157). Pengertian lain dari produktifitas adalah suatu konsep universal yang menciptakan lebih banyak barang dan jasa bagi kehidupan manusia, dengan menggunakan sumber daya yang serba terbatas (Tarwaka, Bakri, dan Sudiajeng, 2004, p.137). Menurut Manuaba (1992) peningkatan produktifitas dapat dicapai dengan menekan sekecil-kecilnya segala macam biaya termasuk dalam memanfaatkan sumber daya manusia (do the right thing) dan meningkatkan keluaran sebesar-besarnya (do the thing right). Dengan kata lain bahwa produktifitas merupakan pencerminan dari tingkat efisiensi dan efektivitas kerja secara total (Tarwaka, Bakri, dan Sudiajeng, 2004, p.138). Menurut Sinungan, (2003, p.12), secara umum produktifitas diartikan sebagai hubungan antara hasil nyata maupun fisik (barang-barang atau jasa) dengan masuknya yang sebenarnya. Produktifitas juga diartikan sebagai tingkatan efisiensi dalam memproduksi barang-barang atau jasa-jasa. Produktifitas juga diartikan sebagai:

- a. Perbandingan ukuran harga bagi masukan dan hasil.
- b. Perbedaan antara kumpulan jumlah pengeluaran dan masukan yang dinyatakan dalam satuan-satuan (unit) umum.

Menurut Wignjosoebroto, (2000, p.25), produktifitas secara umum akan dapat diformulasikan sebagai berikut:

Produktifitas =  $\text{Output/input (measurable)} + \text{input (invisible)}$

Invisible input meliputi tingkat pengetahuan, kemampuan teknis, metodologi kerja dan pengaturan organisasi, dan motivasi kerja.

Sedangkan pengertian efisiensi menurut Mulyamah (1987), menyatakan bahwa “Efisiensi merupakan suatu ukuran dalam membandingkan rencana penggunaan masukan dengan penggunaan yang direalisasikan atau perkataan lain penggunaan yang sebenarnya”. Menurut SP.Hasibuan (1984) yang mengutip pernyataan H. Emerson menyatakan bahwa “Efisiensi adalah perbandingan yang terbaik antara input (masukan) dan output (hasil antara keuntungan dengan sumber-sumber yang dipergunakan), seperti halnya juga hasil optimal yang dicapai dengan penggunaan sumber yang terbatas. Dengan kata lain hubungan antara apa yang telah diselesaikan.” Adapun untuk mencari tingkat efisiensi dapat digunakan rumus sebagai berikut :

$$\text{Efisiensi} = \frac{\text{Input Target}}{\text{Input Aktual}} \quad (2.31)$$

- 1) Jika input yang ditargetkan berbanding input aktual lebih besar atau sama dengan 1 (satu), maka akan terjadi efisiensi.
- 2) Jika input yang ditargetkan berbanding input aktual kurang daripada 1 (satu), maka efisiensi tidak tercapai.

## 2.4 Uji Hipotesis Komparatif

Tabel 2.3 berikut ini adalah tabel yang menunjukkan karakteristik uji statistik yang dikelompokkan menjadi *parametrik* dan *non-parametric*.

Tabel 2.3 Jenis uji statistik

No.	Jenis Uji Statistik	Jenis Data	Jenis Statistik
1.	Parametrik	2 Sampel Independen	<i>Independent sample t-test</i>
		2 Sampel Berhubungan	<i>Paired sample t-test</i>
		2 < Sampel	Anava
2.	Non Parametrik	2 Sampel Independen	<i>Mann Whitney U test</i>
		2 Sampel Berhubungan	<i>Wilcoxon Sign Rank test</i>
		2 < Sampel Independen	<i>Anava ranking Friedman</i>
		2 < Sampel Berhubungan	

### 2.4.1 *Paired-t sample t-test*

*Paired-t sample t-test* merupakan jenis statistik yang digunakan untuk jenis data dua sampel yang berhubungan (*between treatment*). Sampel berkorelasi adalah sampel yang didapatkan dari data yang berasal dari subjek yang sama. Misalnya:

1. Dalam suatu eksperimen, dibandingkan skor *pre-test* dan *post-test* kelompok eksperimen.
2. Perbandingan agresi yang diakibatkan oleh efek kopi dan efek teh pada satu orang.

Pada contoh diatas, terlihat bahwa ada dua jenis subjek yang sama. Untuk contoh langkah-langkah pengujiannya adalah sebagai berikut:

- Hipotesa

$$H_0: \mu_1 - \mu_2 = 0$$

$$H_1: \mu_1 - \mu_2 \neq 0$$

- Masing-masing populasi saling bebas dan berdistribusi normal dalam populasi.
- Masing-masing populasi tidak harus saling bebas antar populasi
- Variansi antara populasi 1 dengan populasi 2 adalah sama ( $\sigma_1 = \sigma_2 = \sigma$ )
- Perhitungan dilakukan dengan menggunakan rumus rata-rata dan standard selisih:

$$\bar{x}_{(1-2)} = \frac{\sum_{j=1}^n x_{(1-2)j}}{n} \quad (2.32)$$

$$s = \sqrt{\frac{\sum_{j=1}^n [x_{(1-2)j} - \bar{x}_{(1-2)}]^2}{n-1}} \quad (2.33)$$

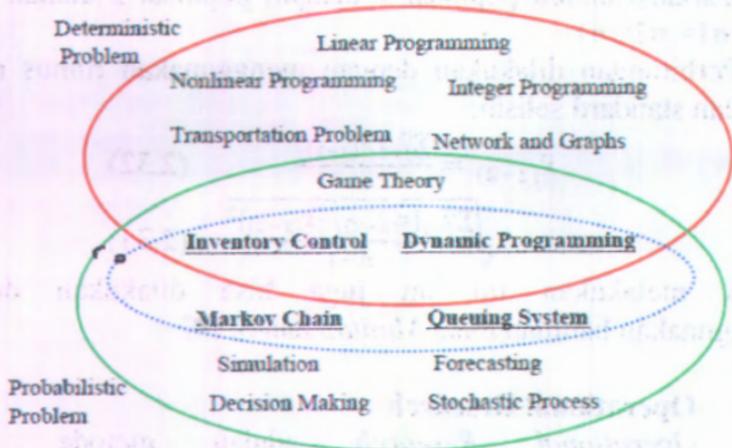
Untuk melakukan uji ini juga bisa dilakukan dengan menggunakan bantuan *tools Minitab* dan *SPSS*.

## 2.5 Operational Research

*Operational Research* adalah metode untuk memformulasikan dan merumuskan permasalahan sehari-hari baik mengenai bisnis, ekonomi, sosial, maupun bidang lainnya kedalam permodelan matematis untuk mendapatkan solusi yang optimal. Tiga hal yang perlu diperhatikan dalam mengimplementasikannya adalah pokok permasalahan, faktor-faktor yang mempengaruhi, dan permodelan. Dalam membangun model matematis dari persoalan, dapat digunakan beberapa permodelan seperti permodelan *linear programming* (PL), *integer programming*, dll. *Operational Research* bisa dikategorikan

seperti pada Gambar 2.6. Dalam *linear programming* terdapat beberapa karakteristik diantaranya:

- Variabel keputusan ( $X$ ), yaitu variabel yang akan menguraikan secara lengkap keputusan-keputusan yang akan dibuat.
- Fungsi tujuan ( $Z$ ), yaitu fungsi dari variabel keputusan yang akan dimaksimumkan/diminimumkan tergantung dari tujuan penyelesaian masalah.
- Pembatas (konstrains), batasan dari sumber daya yang akan dialokasikan.
- Pembatas tanda, yaitu pembatas yang menjelaskan apakah variabel keputusan diasumsikan hanya bernilai nonnegatif atau sebaliknya.



Gambar 2.6 *Operational Research Category*

Bentuk baku dari model matematis yang digunakan adalah :

$$\begin{aligned} \text{Fungsi tujuan :} \quad & \text{Max } Z = C_1X_1 + C_2X_2 + \dots + C_nX_n \\ \text{S.T.} \quad & a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n \leq b_1 \\ & a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n \leq b_2 \\ & a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n \leq b_m \\ & X_1, X_2, \dots, X_n \geq 0 \end{aligned}$$

Dalam menyelesaikan problem optimasi ini, bisa menggunakan *software* LINDO. LINDO (*Linear Interactive and Discrete Optimizer*) adalah paket komputer yang digunakan untuk menyelesaikan *Linear Programming*, *Integer Programming* dan *Kuadratik Programming*. Untuk menggunakan teknik pembangkit kolom dengan bantuan LINDO, ide dasarnya dijelaskan dengan langkah-langkah sebagai berikut (Schrage, 1995):

1. Bentuk dan selesaikan *Linear Programming* awal yang memiliki semua baris dari model yang terdefinisi secara utuh, tetapi dengan sejumlah kecil kolom yang dinyatakan secara eksplisit.
2. Dengan nilai dual solusi kini, bentuk kolom (pola) yang menguntungkan; yaitu, jika  $c_j$  adalah biaya kolom  $j$ ,  $a_{ij}$  adalah koefisien kolom  $j$  pada baris  $i$  untuk  $i = 1, 2, \dots, m$ , dan  $d_i$  adalah harga dual baris  $i$ , tentukan kolom  $j$  yang baru sedemikian sehingga  $c_j + d_1 a_{1j} + d_2 a_{2j} + \dots + d_m a_{mj} < 0$ . Jika tidak ada kolom sedemikian, lalu berhenti.
3. Selesaikan Program Linear dengan kolom baru dari (2) yang telah ditambahkan.
4. Kembali ke (2).

## 2.7 Penelitian Terdahulu

Secara ringkas, penelitian-penelitian yang terkait dengan penerapan data mining untuk estimasi dapat dirangkum pada Tabel 2.4.

Tabel 2.4 Perbandingan antara metode estimasi yang digunakan dalam penelitian sebelumnya

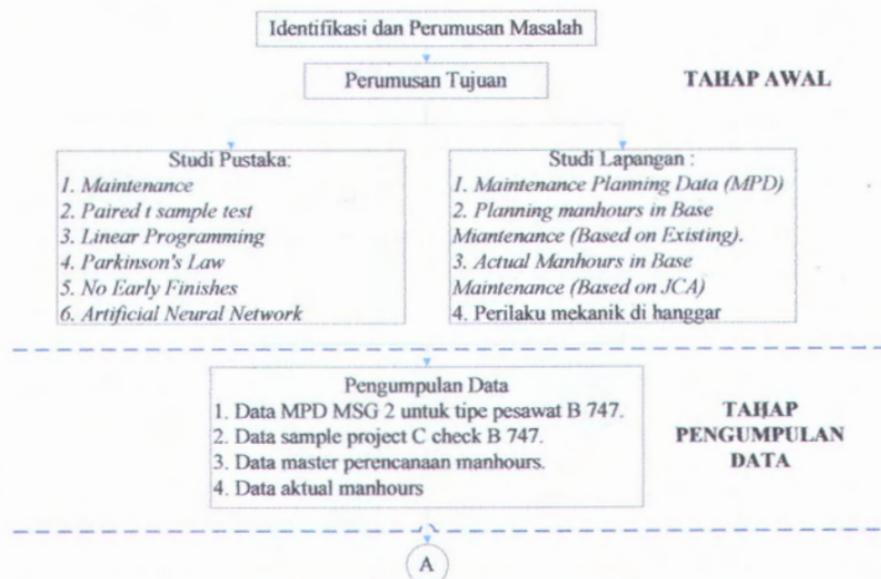
No	Penulis (Tahun)	Judul	Permasalahan (Spesifikasi)	Pendekatan / Metode	Hasil Penelitian
1	Bin Liu and Zu-hua Jiang (Januari 2005)	<i>The Man-hour Estimation Models &amp; Its Comparison of Interim Products Assembly for Shipbuilding</i>	Estimate (Data mining)	<i>Simple linear regression model</i>	Nilai <i>error Linear Regression model</i> dan <i>Artificial Neural Network Model</i> sangat kecil.
				<i>Multiple linear regression model</i>	Tingkat ketelitian <i>Artificial Neural Network Model</i> lebih baik dari <i>Multiple Linear Regression Model</i> .
				<i>Artificial neural network model</i>	Kapabilitas <i>Artificial Neural Network Model</i> lebih baik dibanding kedua model lainnya
2	Ozgun Kisi (Agustus 2005)	<i>Suspended Sediment Estimation Using Neuro-Fuzzy and Neural Network Approaches</i>	Estimate (Data mining)	<i>Neuro-fuzzy model</i>	The Neuro-Fuzzy model memberikan estimasi terbaik dibanding metode lainnya
				<i>Neural networks model</i>	
3	H.Karaborka, O.K. Baykanb, C. Altuntasa, F. Yildza (2008)	<i>Estimation of unknown height with Artificial Neural Network on Digital Terrain Model</i>	Estimate (Data mining)	<i>Modified Shepard's Method</i>	Keakuratan ANN bukan yang terbaik, tetapi sedikit lebih buruk dari <i>radial basis function method</i> dan <i>modified shepard's method</i>
				<i>Radial Basis Function Methods (Multiquadric)</i>	
				<i>Artificial neural network model.</i>	
4	John D. Dowler (June 2008)	<i>Using Neural Networks with Limited Data to Estimate Manufacturing Cost</i>	Estimate (Data mining)	<i>Linear regression model</i>	Kemampuan <i>neural network</i> untuk menampilkan pendekatan <i>cost-efficient design</i> adalah lebih baik dari <i>linear regression model</i>
				<i>Neural networks model</i>	

### BAB III METODOLOGI PENELITIAN

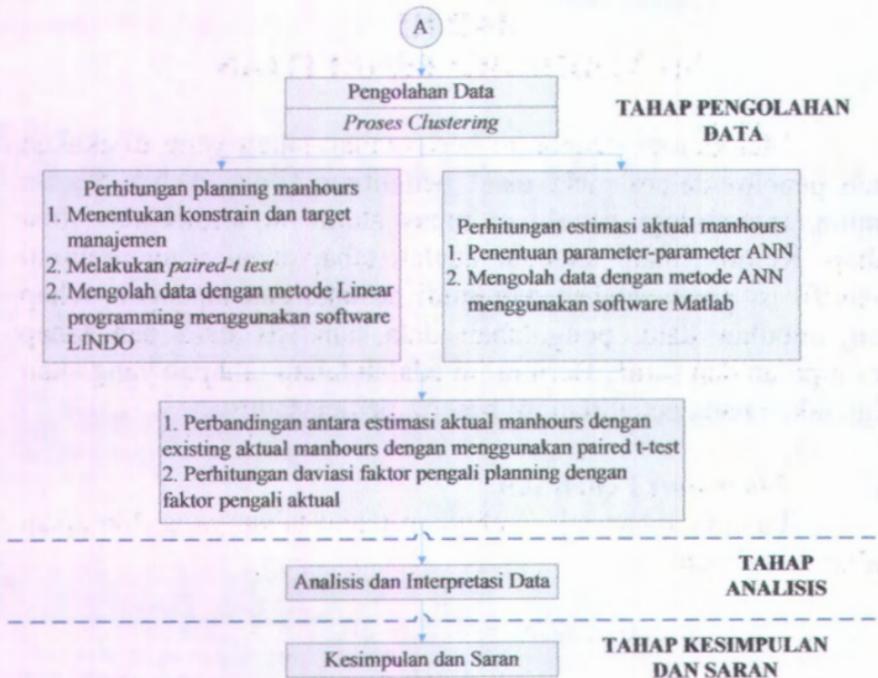
Metodologi penelitian berisi tahap-tahap yang dilakukan oleh peneliti dalam melakukan penelitian Tugas Akhir. Secara umum, metodologi penelitian tugas akhir ini terdiri atas lima tahap. Kelima tahap tersebut adalah tahap awal yang meliputi identifikasi permasalahan dan studi pustaka dan lapangan, tahap pengumpulan data, pengolahan data, analisis data dan tahap kesimpulan dan saran. Berikut ini adalah tahap-tahapan yang akan digunakan pada penelitian ini beserta penjelasannya.

#### 3.1 *Flowchart* Penelitian

Pada Gambar 3.1 disebutkan tahap-tahap yang dilakukan dalam penelitian.



Gambar 3.1 *Flowchart* penelitian



Gambar 3.1 *Flowchart* penelitian (lanjutan)

## 3.2 Penjelasan *Flowchart* Penelitian

Berikut ini penjelasan mengenai langkah-langkah pengerjaan penelitian.

### 3.2.1 Tahap Awal

Tahap awal ini berisi mengenai identifikasi masalah dalam penelitian. Dalam tahap ini, peneliti mengidentifikasi permasalahan yang ada di dalam perusahaan kemudian menyusun kerangka penyelesaian secara umum.

#### 3.2.1.1 Identifikasi dan Perumusan Masalah

Identifikasi masalah adalah kegiatan yang bertujuan untuk merumuskan permasalahan yang dihadapi oleh perusahaan. Sedangkan perumusan masalah adalah tahap untuk menetapkan masalah yang akan menjadi fokus penelitian dalam tugas akhir ini sehingga masalah yang dibahas dapat dibatasi dan menjadi lebih

jelas. Pada tahap ini akan dipelajari mengenai proses penentuan faktor pengali perencanaan manhours untuk *C-Check* pada jenis pesawat B 747 yang menggunakan MSG 2.

### 3.2.1.2 Perumusan Tujuan

Pada tahap ini akan ditetapkan tujuan yang ingin dicapai dari penelitian atas masalah yang dihadapi oleh perusahaan.

### 3.2.1.3 Studi Pustaka dan Studi Lapangan.

Studi pustaka adalah kegiatan untuk mempelajari teoriteori yang berkaitan dengan permasalahan yang akan diselesaikan pada penelitian tugas akhir ini. Studi pustaka berfungsi sebagai pedoman secara teoritis mengenai metode-metode yang dapat digunakan untuk menyelesaikan permasalahan dalam penelitian tugas akhir. Sementara itu studi lapangan diperlukan untuk mendapatkan gambaran mengenai kondisi nyata di perusahaan sehingga dapat dijadikan pertimbangan dalam menyelesaikan masalah yang dihadapi oleh perusahaan.

## 3.2.2 Tahap Pengumpulan Data

Pada langkah ini dilakukan pengumpulan data-data yang diperlukan dalam melaksanakan penelitian. Data yang diperlukan antara lain adalah :

1. Data MPD MSG 2 untuk tipe pesawat B 747.  
Data ini diperoleh dari *maintenance program* yang merupakan salah satu *database* perusahaan dan ketepatan datanya sudah divalidasi oleh *planner* di TBS 6 dan TBS 7 berdasarkan kesesuaiannya dengan data MPD asli yang diberikan oleh perusahaan manufaktur pesawat.
2. Data *sample project C check* B 747.  
*Record* data *project* di perusahaan bisa diambil dari SAP dan *Jobcard Automation (barcoding)*. Dalam penelitian ini tidak digunakan data yang berasal dari SAP dikarenakan *record* data tersebut memiliki tingkat keakuratan yang rendah. Sehingga digunakan data yang berasal dari sistem *Jobcard Automation (barcoding)*.

Pemilihan *sample project* berdasarkan *record project* yang dimiliki oleh TBS 7 dengan penyesuaian waktu selama sistem *barcoding* ini diimplementasikan di unit *Base Maintenance*.

3. Data master perencanaan *manhours*.  
Data ini diperoleh dari *planner* di TBS 6 yang ter-*record* di sistem *Jobcard Automation (barcoding)*. Master perencanaan *manhours* ini diturunkan oleh *planner* ke bagian *produksi* ketika suatu *project* akan dikerjakan.
4. Data aktual *manhours*.  
Data ini merupakan *record* keseluruhan waktu aktual penyelesaian setiap *jobcard* yang diturunkan pada suatu *project maintenance* tertentu yang telah selesai dikerjakan. Data ini terrekam dalam sistem *Jobcard Automation*.

### 3.2.3 Tahap Pengolahan Data

Pada tahap pengolahan data ini dilakukan dua jenis perhitungan, yakni perhitungan penentuan faktor pengali untuk perencanaan *manhours* dengan menggunakan *linear programming* dan perhitungan untuk mengetahui estimasi *manhours* aktual dengan menggunakan *Artificial Neural Network* sehingga bisa diketahui faktor pengali untuk kondisi aktual. Berdasarkan jurnal yang terangkum pada Tabel 2.4 diketahui bahwa dalam penentuan estimasi dapat digunakan metode *Simple Linear Regression*, *Multiple Linear Regression*, dan *Artificial Neural Network*, namun *Artificial Neural Network* merupakan metode yang lebih baik dari yang lain. Selanjutnya dilakukan perbandingan yang terjadi dari perubahan faktor pengali yang ditetapkan terhadap hasil estimasi *actual manhours*.

#### 3.2.3.1 Linear Programming (LP)

Berikut ini beberapa karakteristik *linear programming* yang digunakan dalam penelitian ini:

- Variabel keputusan ( $x$ ) yang merupakan besar faktor pengali yang ingin diketahui.

- Fungsi tujuan ( $Z$ ), yaitu *minimize* total faktor pengali.
- Pembatas (konstrain) yaitu:
  - ❖ Nilai MPD
  - ❖ Target faktor pengali
  - ❖ Hasil perkalian MPD dengan faktor pengali
- a. Pembatas tanda, yaitu pembatas  $=$ ;  $\leq$ ; atau  $\geq$ .

Penentuan nilai dari karakteristik-karakteristik tersebut, berdasarkan kondisi sistem di lapangan dengan tiga alternatif perbaikan yakni alternatif A, alternatif B, dan alternatif C. Dalam melakukan proses pengolahan datanya digunakan *software* LINDO.

### 3.2.3.2 Artificial Neural Network (ANN)

Data yang digunakan sebagai data *training* adalah 80% dari data yang ada, sedangkan 20% sisanya akan digunakan sebagai data *testing*. Aturan untuk menentukan berapa jumlah data *training* dan data *testing* sesuai saran Haykin (1994), yaitu jumlah data *testing* 10% sampai dengan 20% dari data yang ada. Fungsi yang digunakan dalam melakukan estimasi dengan *Artificial Neural Network* adalah fungsi-fungsi yang sudah ada dalam Matlab, yakni dengan menggunakan Matlab *Neural Network toolbox*. Dalam mendesain *Artificial Neural Network*, perlu dipertimbangkan beberapa hal (Santosa, 2007), yaitu:

#### 1. Tipe *network*

Tipe *network* yang digunakan dalam penelitian ini adalah *back-propagation (feed forward network)*, yang digunakan untuk *multilayer Neural Network*.

#### 2. Jumlah *layer*

Terdapat tiga *layer* yang digunakan dalam penelitian ini yaitu *input layer*, *hidden layer* dan *output layer*. Jumlah *layer* dipengaruhi oleh kompleksitas dari permasalahan yang diselesaikan. Semakin banyak atribut dan pola dari data yang digunakan menyebabkan jumlah *layer* semakin banyak.

3. Banyak simpul/*node*

Jumlah *node* untuk setiap *layer* berbeda. Pada awal pengolahan data akan dilakukan *running* dengan beberapa jumlah *node* ( $S_i$ ) yang berbeda pada *hidden layer* untuk melihat pengaruh yang dihasilkan terhadap hasil estimasi, yaitu 3, 5, 7, dan 10. Sedangkan pada *output layer* hanya terdapat satu *node*, yaitu hasil estimasi aktual *manhours*.

4. Fungsi transfer/*activation function*

Fungsi aktivasi  $\varphi(\cdot)$  yang digunakan dalam *hidden layer* adalah fungsi *sigmoid* dan fungsi yang digunakan pada *output layer* adalah *linear function*. Ini merupakan jenis fungsi aktivasi yang umum digunakan (Santosa, 2007; Bajimaya, 2007). Pada proses pembelajarannya, akan digunakan *gradient descent* untuk menemukan bobot  $w$  dan bias  $b$ . Bobot nilai bias awal yang digunakan merupakan bilangan random. *Learning rate* ( $\eta$ ) yang digunakan dalam proses pembelajaran telah ditentukan terlebih dahulu. Dalam penelitian ini digunakan beberapa *learning rate* ( $\eta$ ) yang berbeda yaitu 0.1, 0.3, 0.5, 0.7, dan 0.9 untuk melihat pengaruhnya terhadap hasil estimasi aktual *manhours*. Dalam Krose and van der Snagt (1996) dikatakan bahwa hasil *error* yang minimum akan sangat lama untuk dapat tercapai bila menggunakan *learning rate* yang kecil. Namun bila digunakan *learning rate* yang besar hasil *error* yang minimum tidak akan pernah tercapai, sehingga diberikan nilai momentum (*smoothing*) yang dapat membantu dalam menghasilkan nilai *error* yang minimum dengan lebih cepat, dikenal dengan *gradient descent with momentum*. Besarnya nilai momentum ( $mc$ ) yang digunakan adalah 0.5, 0.6 dan 0.7 untuk melihat pengaruhnya terhadap estimasi *actual manhours*.

5. Jumlah *epoch*/iterasi yang digunakan dalam *training*.

Jumlah iterasi telah ditetapkan pada saat awal, dimana terdapat beberapa jumlah iterasi yang berbeda untuk dibandingkan hasilnya yaitu 50, 100, dan 250. Iterasi dapat berhenti apabila *error* yang dihasilkan tidak berubah lagi antara iterasi sebelumnya dengan iterasi saat ini. Penentuan iterasi ini sangat berpengaruh dalam *training* suatu ANN. Iterasi yang terlalu banyak mungkin akan menghasilkan *error* yang sangat kecil pada tahap *training* tetapi akan menghasilkan *error* yang tinggi dalam validasi, dikenal dengan istilah *overfitting*.

*Artificial Neural Network* memiliki sifat *local optimum* dimana hasil *running* tiap replikasi akan berbeda sehingga pada pengolahan data ini dilakukan perhitungan jumlah sample/replikasi dengan persamaan :

$$n = \left( \frac{z_{\alpha/2}\sigma}{g} \right)^2 \dots\dots(3.1)$$

### 3.2.4 Tahap Analisis

Data yang telah dikumpulkan dan diolah pada tahap sebelumnya akan dianalisis dan diinterpretasikan pada tahap ini. Proses analisis dan interpretasi data ini dilakukan pada hasil *Linear Programming* dan hasil *Artificial Neural Network*.

#### 3.2.4.1 Analisis Hasil *Linear Programming* (LP)

Dalam hal ini dilakukan perbandingan antara deviasi faktor pengali *actual* dan *planning* untuk mengetahui performansi sistem *existing* dan dilakukan perbandingan nilai faktor pengali yang dihasilkan pada alternatif A, alternatif B, dan alternatif C. Selanjutnya dibuat analisisnya beserta fungsi IF-AND yang dimasukkan di *excel* untuk mempermudah proses penginputan data bagi *Artificial Neural Network* (ANN).

#### 3.2.4.2 Analisis Hasil *Artificial Neural Network* (ANN)

Dalam hal ini dilakukan analisa terhadap performansi MSE yang dihasilkan oleh *Artificial Neural Network* (ANN) serta

pengaruh parameter-parameter ANN seperti *epoch*, *momentum*, *learning rate*, dan jumlah node terhadap MSE yang dihasilkan. Selanjutnya dilakukan analisa terhadap kombinasi terbaik dari parameter ANN yang menghasilkan MSE terkecil. Setelah diperoleh hasil estimasi *manhours actual* dengan ANN maka dilakukan analisis terhadap perbandingan *manhours actual* pada kondisi *existing* dengan kondisi pada alternatif A serta efek yang ditimbulkan pada perusahaan.

### 3.2.5 Tahap Kesimpulan dan Saran

Tahap kesimpulan dan saran merupakan tahap akhir dari penelitian tugas akhir. Tahap ini berisikan semua kesimpulan yang diperoleh dari hasil pengumpulan, pengolahan dan analisis terhadap data penelitian. Pada tahap saran berisikan rekomendasi-rekomendasi bagi perusahaan dan saran untuk penelitian selanjutnya.

## BAB IV

### PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini akan ditampilkan proses pengumpulan dan pengolahan data secara sistematis sesuai dengan kerangka penelitian yang telah dibuat.

#### 4.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini meliputi data *Maintenance Planning Data (MPD) MSG-2*, data *master planning manhours*, data *actual manhours* dan data *sample project C Check maintenance B 747*.

##### 4.1.1 Data Sample Project C-Check Maintenance B 747

Terdapat tiga jenis sistem *record* data yang digunakan di unit *Base Miantenance* yakni sistem (*Systems, Applications, and Product*) SAP, sistem *Productivity, Efficiency and Utility (PEU)*, dan sistem *Job Card Automation (JCA / Barcoding)*. Ketiga sistem ini berjalan sendiri-sendiri, tidak terintegrasi menjadi satu sehingga ketiganya memiliki input, proses dan output data yang berbeda. Sistem SAP sudah lama diterapkan namun tingkat akurasi dan validitasnya sangat rendah, hal ini disebabkan oleh penginputan data yang masih manual oleh mekanik terhadap *plan slip jobcard* dan penginputan secara manual oleh operator penginput data terhadap *software* SAP di komputer. Untuk sistem PEU, peneliti memiliki keterbatasan untuk mengaksesnya sehingga tidak diperoleh *record* datanya. Sedangkan sistem *Barcoding* merupakan sistem yang baru diterapkan oleh unit *Base Maintenance* mulai bulan Juli 2009. Sistem ini memiliki tingkat akurasi dan validitas yang lebih baik dibanding SAP karena sistem ini berusaha meminimumkan penginputan data secara manual baik penginputan data yang dilakukan oleh mekanik maupun penginputan data yang dilakukan oleh operator penginput data yang biasa disebut sebagai *time keeper*. Sehingga dalam

penelitian ini digunakan data *sample project* yang berasal dari *barcoding*.

Dari bulan Juli 2009 hingga Desember 2009, terdapat empat *project maintenance C-Check* tipe pesawat Non-Garuda Boeing 747 yang berhasil terrekap datanya dan ditunjukkan pada Tabel 4.1, sedangkan *record* datanya melalui *barcoding* ditunjukkan pada Tabel 4.2 berikut ini.

Tabel 4.1 Rekap data *barcoding* pada project B747

NO	A/C Type	Reg Type	MPD	SO No	Maint type	Month
1	B 747-200/300	5N-BMG	MSG 2	10006149	C05-Check	SEPT - OKT
2	B 747-200/301	5N-MBB	MSG 2	10005649	C05-Check	JUL
3	B 747-200/302	5N-DDK	MSG 2	10006186	C06-Check	SEPT - OKT
4	B 747-400	N789SA	MSG 3	10006021	C01-Check	AGSTS

Tabel 4.2 *Record* data *barcoding* pada project B747

Reg Type	Routine J/C Ter-	Routine J/C Tidak	Total Rutin J/C	Data Ter-record (%)
5N-BMG	591	354	945	63%
5N-MBB	724	221	945	77%
5N-DDK	925	206	1131	82%
N789SA	56	466	522	11%

*Record* data N789SA terlalu kecil dan tidak bisa mewakili data keseluruhan pada *project* ini. Sedangkan pada *record* data 5N-BMG dan 5N-MBB belum cukup mewakili data keseluruhan karena masih kurang dari 80%, selain itu pada hasil output *record* datanya terdapat keanehan data karena kesalahan *time keeper* di lapangan sehingga kedua data ini tidak bisa digunakan. Prosentase sebesar 80% ini merupakan penilaian yang digunakan *planner* agar suatu *project* bisa dikatakan layak untuk dianalisa. Sedangkan *record* data 5N-DDK sudah lebih dari 80% dan hasil output *record* datanya tidak ditemukan keanehan yang berarti, selain itu pada *project* ini tipe *maintenance* yang dilakukan adalah yang terbesar diantara yang lain sehingga *record*

*jobcard* yang ada juga paling lengkap. Oleh karena itu, digunakan *sample project* 5N-DDK C06 Check dalam penelitian ini.

#### 4.1.2 Maintenance Planning Data (MPD)

Pada penelitian ini, data yang digunakan adalah data *Maintenance Planning Data (MPD)* dengan konsep *Maintenance Streering Group 2 (MSG-2)* untuk pesawat Non-Garuda dengan Tipe B747-200/300 yang dikeluarkan oleh FAA. Pemilihan jenis MPD ini disesuaikan dengan *sample project* yang digunakan yakni *project* 5N-DDK yang merupakan tipe pesawat Boeing 747-200 yang menggunakan MPD MSG 2.

Data MPD terdiri dari ribuan *J/C No* yang masing-masing menentukan jenis pekerjaan yang harus dikerjakan, jenis interval dan *standard manhours* yang diberikan untuk menyelesaikan pekerjaan tersebut. Tidak semua *J/C No* tersebut akan dikerjakan ketika C-Check *maintenance* dilakukan. Hal ini mengikuti aturan yang ditetapkan dalam konsep MSG 2. Ketika jenis *maintenance* yang dilakukan untuk pesawat adalah *C06-Check*, maka *J/C No* yang diturunkan meliputi *J/C No* yang berada pada interval *Preflight*, *Preflight/transit*, 1A, 2A, 4A, 1C, 2C, 3C dan 6C. Berikut ini adalah contoh sebagian data MPD yang terdiri dari *jobcard no (J/C No)*, *Interval* dan *manhours (mhrs)* yang digunakan sebagai bahan penelitian ditunjukkan pada Tabel 4.3.

Tabel 4.3 Data MPD (*J/C No*, *Interval*, *Manhours*)

J/C No	Interval	Mhrs
1-10-001	PREFLIGHT	0.1
1-25-002	PREFLIGHT/TRANSIT	0.2
2-21-001-01	1A	0.2
2-21-002-01	2A	1
2-21-003-01	4A	0.3
4-00-012-01	1C	3
4-21-002-01-1	3C	1
4-21-012-01-1	2C	1
4-24-010	6C	0.5
4-53-016	7C	48
4-53-017-01-1	4C	5

#### 4.1.3 Data Master Planning Manhours

Data *master planning manhours* dibuat oleh *planner* ketika ada *project maintenance* yang harus dilakukan. Tabel 4.4 berikut ini adalah sebagian data *master planning manhours* yang diturunkan untuk *project 5N-DDK C06-Check* yang digunakan sebagai bahan penelitian.

Tabel 4.4 Data *master planning manhours*

No	J/C NO	Interval	MPD (Mhrs)	Faktor Pengali	Plan (Mhrs)
1	1-10-001	Preflight	0.1	5	0.5
2	1-10-002	Preflight	0.1	5	0.5
3	1-25-001	Preflight	0.4	2.5	1
4	1-25-002	Preflight	0.2	2.5	0.5
5	1-25-003	Preflight	0.2	2.5	0.5
6	1-25-004	Preflight	0.2	2.5	0.5
7	1-28-001	Preflight	0.2	2.5	0.5
8	1-29-001	Preflight	0.1	5	0.5
9	1-32-001	Preflight	0.3	2.5	0.75

#### 4.1.4 Data Actual Manhours

Data *actual manhours* yang digunakan berasal dari *barcoding* Portal GMF untuk SO No 10006186 yang merupakan *project 5N-DDK C06-Check*. Tabel 4.5 berikut ini adalah sebagian *record* data *actual manhours* yang digunakan sebagai bahan penelitian.

Tabel 4.5 Data *actual manhours*

PACKAGE 5N-DDK SO 10006186							
NO	J/C No	Insp Type	Status	Plan (Mhrs)	Act (Mhrs)	Date Start	Date Acc
1	4-57-021-01-1	2C	CLOSE	2.625	1.283	2009-09-19 09:13	2009-09-19 10:30
2	4-57-021-01-2	2C	CLOSE	2.625	0.816	2009-09-18 13:43	2009-09-18 14:32
3	4-57-022-01-1	1C	CLOSE	2.55	1.316	2009-09-16 09:45	2009-09-16 11:04
4	4-57-022-01-2	1C	OPEN	2.55	0	2009-09-03 10:55	
5	4-57-023-01-1	2C	CLOSE	3	5.383	2009-09-19 07:46	2009-09-19 10:33

Jenis *maintenance* yang dilakukan untuk pesawat 5N-DDK adalah *C06-Check*, maka *J/C No* yang diturunkan meliputi *J/C No* yang berada pada interval *Preflight*, *Preflight/transit*, 1A, 2A, 4A, 1C, 2C, 3C dan 6C dengan komposisi jumlah *jobcard* seperti yang ditampilkan dalam Tabel 4.6 berikut.

Tabel 4.6 Komposisi *routine jobcard*

Interval	Jumlah Jobcard
Preflight	22
1A	75
2A	83
4A	139
1C	604
2C	161
3C	40
6C	7
Total J/C	1131
Missing value	206
Jumlah data yang digunakan	925

## 4.2 Pengolahan Data

Pada bagian ini akan dilakukan pengolahan data untuk menghitung perencanaan faktor pengali baru dengan menggunakan metode *Linear Programming* (LP) dan melakukan perhitungan estimasi *actual manhours* dengan menggunakan metode *Artificial Neural Network* (ANN).

### 4.2.1. *Linear Programming* (LP)

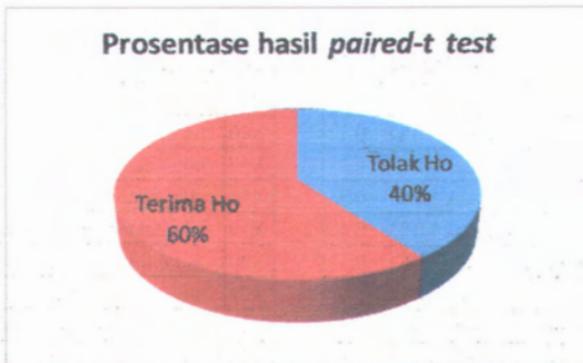
Seperti yang telah dijelaskan pada bab 3, digunakan metode *linear programming* dalam menentukan *planning manhours* yang akan diturunkan pada *C-Check* tipe pesawat B747-200/300. Sebelum dilakukan pengolahan data menggunakan *linear programming*, maka terlebih dahulu ditentukan konstrainnya dengan mengolah data project 5N-DDK

yang sebagian contoh datanya bisa dilihat di Lampiran 1. Penentuan konstrain tersebut berdasarkan perhitungan selisih rata-rata faktor pengali *plan* dan *actual* serta hasil *paired-t test*. Proses perhitungan *paired-t test* dilakukan dengan menggunakan *software Minitab* dan contoh perhitungannya ditampilkan di Lampiran 2. Hasil rekapan perhitungannya diberikan pada Tabel 4.7.

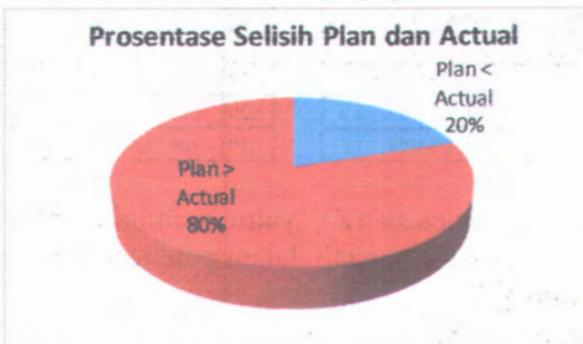
Tabel 4.7 Rekapan hasil selisih dan *paired-t test* antara *planning* dan *actual*

MPD (mhrs)	Faktor Pengali		AC - PL	Hasil paired-t test	Alternatif A	Alternatif B	Alternatif C
	Plan (PL)	Actual (AC)					
0.1	4.85	4.11	-0.74	tolak Ho	tolak Ho	tolak Ho	tolak Ho
0.2	2.41	2.46	0.05	terima Ho	terima Ho	terima Ho	terima Ho
0.3	2.37	2.30	-0.07	terima Ho	terima Ho	terima Ho	terima Ho
0.4	2.16	2.25	0.09	terima Ho	terima Ho	terima Ho	terima Ho
0.5	1.73	1.94	0.21	tolak Ho	tolak Ho	tolak Ho	tolak Ho
0.6	1.68	1.70	0.02	terima Ho	terima Ho	terima Ho	terima Ho
0.7	1.83	1.82	-0.01	tolak Ho	terima Ho	terima Ho	terima Ho
0.8	1.45	1.23	-0.22	tolak Ho	terima Ho	terima Ho	terima Ho
0.9	2.22	1.88	-0.34	terima Ho	terima Ho	terima Ho	terima Ho
1	1.56	1.46	-0.1	terima Ho	tolak Ho	tolak Ho	tolak Ho
1.2	1.67	1.39	-0.27	terima Ho	terima Ho	terima Ho	terima Ho
1.3	1.60	1.25	-0.35	terima Ho	terima Ho	terima Ho	terima Ho
1.5	1.56	1.40	-0.16	terima Ho	terima Ho	terima Ho	terima Ho
1.6	1.69	1.30	-0.39	terima Ho	terima Ho	terima Ho	terima Ho
1.8	1.37	0.66	-0.71	tolak Ho	terima Ho	terima Ho	terima Ho
1.9	1.68	1.24	-0.45	terima Ho	terima Ho	terima Ho	terima Ho
2	1.21	1.35	0.14	tolak Ho	terima Ho	terima Ho	terima Ho
2.1	1.33	1.22	-0.11	tolak Ho	terima Ho	terima Ho	terima Ho
2.5	1.02	0.92	-0.10	terima Ho	terima Ho	terima Ho	terima Ho
2.6	0.94	0.58	-0.36	tolak Ho	terima Ho	terima Ho	terima Ho
2.7	1.06	0.39	-0.67	tolak Ho	terima Ho	terima Ho	terima Ho
2.9	0.92	0.47	-0.45	tolak Ho	terima Ho	terima Ho	terima Ho
3	1.05	1.01	-0.04	terima Ho	terima Ho	terima Ho	terima Ho

Statistik deskriptif untuk hasil *paired-t test* antara *planning man hours* dan *actual manhours* ditampilkan dalam Gambar 4.1, sedangkan statistik deskriptif untuk hasil perhitungan selisih faktor pengali antara *actual* dan *plan* ditampilkan pada Gambar 4.2.



Gambar 4.1 Prosentase hasil *paired-t test* kondisi *existing*



Gambar 4.2 Prosentase selisih *plan* dan *actual* kondisi *existing*

Dari Gambar 4.1 dan Gambar 4.2 diatas diketahui bahwa 60% perencanaan yang dilakukan tidak berbeda secara signifikan namun terdapat beberapa selisih antara *planning* dan *actual* yakni 80% dari data menunjukkan bahwa faktor pengali *planning* lebih besar dari *actual*. Dari preferensi manajemen dan dengan memperhatikan Tabel 4.7 bisa ditentukan beberapa karakteristik *linear programming* yang digunakan dalam penelitian ini meliputi:

1. *Decision variable* ( $x$ ), yaitu besar nilai faktor pengali yang ingin ditentukan untuk setiap nilai MPD yang ada.  
 $X_i$  = nilai faktor pengali untuk nilai MPD ke  $i$ .  
 Tabel 4.8 berikut ini merupakan pendefinisian *decision variable*  $X_i$  untuk  $1 \leq i \leq 30$ .

Tabel 4.8 *Decision variable*

$X_i$	Keterangan
X1	nilai faktor pengali untuk MPD = 0.1
X2	nilai faktor pengali untuk MPD = 0.2
X3	nilai faktor pengali untuk MPD = 0.3
X4	nilai faktor pengali untuk MPD = 0.4
X5	nilai faktor pengali untuk MPD = 0.5
X6	nilai faktor pengali untuk MPD = 0.6
X7	nilai faktor pengali untuk MPD = 0.7
X8	nilai faktor pengali untuk MPD = 0.8
X9	nilai faktor pengali untuk MPD = 0.9
X10	nilai faktor pengali untuk MPD = 1
X11	nilai faktor pengali untuk MPD = 1.1
X12	nilai faktor pengali untuk MPD = 1.2
X13	nilai faktor pengali untuk MPD = 1.3
X14	nilai faktor pengali untuk MPD = 1.4
X15	nilai faktor pengali untuk MPD = 1.5

$X_i$	Keterangan
X16	nilai faktor pengali untuk MPD = 1.6
X17	nilai faktor pengali untuk MPD = 1.7
X18	nilai faktor pengali untuk MPD = 1.8
X19	nilai faktor pengali untuk MPD = 1.9
X20	nilai faktor pengali untuk MPD = 2
X21	nilai faktor pengali untuk MPD = 2.1
X22	nilai faktor pengali untuk MPD = 2.2
X23	nilai faktor pengali untuk MPD = 2.3
X24	nilai faktor pengali untuk MPD = 2.4
X25	nilai faktor pengali untuk MPD = 2.5
X26	nilai faktor pengali untuk MPD = 2.6
X27	nilai faktor pengali untuk MPD = 2.7
X28	nilai faktor pengali untuk MPD = 2.8
X29	nilai faktor pengali untuk MPD = 2.9
X30	nilai faktor pengali untuk MPD = 3

2. *Objective function (Z)*, yaitu meminimumkan besarnya total nilai faktor pengali dalam perencanaan *manhours*.

$$Z = \min \sum X_i.$$

$$Z = \min X_1 + X_2 + \dots + X_{30}$$

3. *Constrain*, fungsi pembatas yang terdiri dari :

- a) Nilai MPD 0.1; 0.5; 1; 2; dan 3 digunakan sebagai fungsi pembatas dalam *linear programming* dimana besar faktor pengalinya akan langsung ditentukan oleh manajemen sesuai dengan target yang ingin dicapai. Dibuat tiga alternatif batas nilai faktor pengali untuk MPD berdasarkan preferensi *expert* di perusahaan seperti yang ditunjukkan pada Tabel 4.9 Jadi nilai faktor pengali untuk X1, X5, X10, X20 dan X30 sudah ditentukan terlebih dahulu.

Tabel 4.9 *Constrain* nilai faktor pengali MPD

MPD	Alternatif		
	A	B	C
0.1	4	3	2.5
0.5	2	2	2
1	1.5	1.5	1.5
2	1.3	1.3	1.3
3	1.01	1.01	1.01

- b) Nilai faktor pengali harus semakin menurun seiring dengan naiknya nilai MPD yang ditetapkan dan besar selisih antara faktor pengali yang berdekatan harus lebih besar dari 0.01. Nilai 0.01 ini merupakan nilai toleransi untuk besarnya factor pengali maksimal terdapat dua angka dibelakang tanda koma. Secara matematis bisa dituliskan persamaannya sbb:

$$X_1 - X_2 > 0.01$$

$$X_2 - X_3 > 0.01$$

.....

.....

$$X_{29} - X_{30} > 0.01$$

- c) Hasil perkalian nilai MPD dengan faktor pengali harus semakin meningkat seiring dengan meningkatnya nilai manhours MPD dari 0.1 hingga 3 dan besar selisih peningkatannya diusahakan sama.

$$(0.2X_2 - 0.1X_1) = (0.3X_3 - 0.2X_2)$$

$$(0.2X_2 - 0.1X_1) - (0.3X_3 - 0.2X_2) = 0$$

$$0.2X_2 - 0.1X_1 - 0.3X_3 + 0.2X_2 = 0$$

$$0.4X_2 - 0.1X_1 - 0.3X_3 = 0$$

Sehingga bisa dituliskan fungsi pembatasnya adalah sbb:

$$0.4X_2 - 0.1X_1 - 0.3X_3 = 0$$

$$0.6X_3 - 0.2X_2 - 0.4X_4 = 0$$

$$5.8X29-2.8X28-3X30=0$$

4. Pembatas tanda, yaitu pembatas = ; ≤ ; atau ≥ .

Proses pengolahan datanya secara lengkap dengan menggunakan *software* LINDO diberikan pada Lampiran 3. Sedangkan hasil *running* program tersebut dirangkum pada Tabel 4.10.

Tabel 4. 10 Faktor pengali hasil *running* LINDO

MPD	Alternatif Faktor Pengali			MPD	Alternatif Faktor Pengali		
	A	B	C		A	B	C
0.1	4.00	3.00	2.50	1.6	1.35	1.35	1.35
0.2	2.75	2.38	2.18	1.7	1.34	1.34	1.34
0.3	2.33	2.10	2.08	1.8	1.32	1.32	1.32
0.4	2.13	2.06	2.03	1.9	1.31	1.31	1.31
0.5	2.00	2.00	2.00	2	1.30	1.30	1.30
0.6	1.83	1.83	1.83	2.1	1.26	1.26	1.26
0.7	1.71	1.71	1.71	2.2	1.22	1.22	1.22
0.8	1.63	1.63	1.63	2.3	1.19	1.19	1.19
0.9	1.56	1.56	1.56	2.4	1.16	1.16	1.16
1	1.50	1.50	1.50	2.5	1.13	1.13	1.13
1.1	1.46	1.46	1.46	2.6	1.10	1.10	1.10
1.2	1.43	1.43	1.43	2.7	1.07	1.07	1.07
1.3	1.41	1.41	1.41	2.8	1.05	1.05	1.05
1.4	1.39	1.39	1.39	2.9	1.03	1.03	1.03
1.5	1.37	1.37	1.37	3	1.01	1.01	1.01

Dari hasil tersebut maka bisa ditentukan fungsi IF-AND pada Excel untuk dimasukkan pada record data 5N-DDK dengan kolom J3 sebagai nilai MPD yakni sebagai berikut:

1. Skenario A

Artinya Jika nilai MPD kurang dari atau sama dengan 0.1, maka nilai faktor pengalinya 2.5. Dan jika nilai MPD lebih besar dari 0.1 dan kurang dari 0.2 maka nilai faktor pengalinya adalah 2.18. Begitu seterusnya hingga sampai pada pengertian jika nilai MPD lebih besar dari 3 maka nilai faktor pengalinya 1 dan nilai MPD tidak ada dalam persamaan tersebut maka nilainya FALSE.

#### 4.2.2. *Artificial Neural Network* (ANN)

Seperti yang telah dijelaskan pada bab 3, untuk melakukan estimasi *actual manhours* dilakukan dengan menggunakan pendekatan *Artificial Neural Network* dengan menggunakan software Matlab 7.0.4.

##### 4.2.2.1 Pengujian Bahasa Program Matlab

Fungsi yang digunakan dalam melakukan estimasi *actual manhours* telah terdapat dalam Matlab. Berikut ini merupakan fungsi-fungsi yang digunakan dalam *Neural Network Toolbox* untuk estimasi:

#### 1. *Pre-processing*

Fungsi yang digunakan untuk melakukan *pre-processing* adalah:

$[pn, minp, maxp, tn, mint, maxt] = \text{premnmx}(p', t')$

$[an, minap, maxa, sn, mins, maxs] = \text{premnmx}(a', s')$

Keterangan:

p = data *training*

t = target *training*

a = data *testing*

s = target *testing*

pn = data *training* hasil *pre-processing*

tn = target *training* hasil *pre-processing*

an = data *testing* hasil *pre-processing*

sn = target *testing* hasil *pre-processing*

#### 2. *Feedforward Network*

Fungsi yang digunakan untuk membentuk *feedforward network* dengan 2 layer (*hidden layer* dan *output layer*):

```
net=newff(PR,[S1 S2],{TF1 TF2},BTF,BLF)
```

Keterangan:

PR = minmax(pn)

Si = Jumlah node tiap layer

Tfi = Fungsi aktivasi

BTF = Fungsi training

BLF = Fungsi learning

### 3. Training Network dan Post-Processing

Fungsi yang digunakan untuk melakukan *training* terhadap data yang ada:

```
net.trainParam.epochs='epoch'
```

```
net.trainParam.lr=' $\eta$ '
```

```
net.trainParam.mc='mc'
```

```
net=train(net,pn,tn)
```

```
y=sim(net,pn)
```

```
x=postmnmx(y',mint,maxt)
```

Keterangan:

epoch = jumlah iterasi

$\eta$  = besar *learning rate*

mc = besar momentum

### 4. Testing dan Post-Processing

Fungsi yang digunakan dalam *testing*:

```
y=sim(net,an)
```

```
x=postmnmx(y',mins,maxs)
```

Keterangan:

x=hasil prediksi

Sebelum dilakukan pengolahan dengan data project 5N-DDK yang berasal dari *barcoding*, akan dilakukan pengujian dahulu untuk mengetahui apakah *syntax* untuk fungsi-fungsi tersebut dapat berjalan dengan semestinya menggunakan data yang terdapat dalam Santosa (2007, 137). Data tersebut berisi 22 data dimana 11 data pertama merupakan input, sedangkan 11 data

kedua merupakan output. Dari 22 data ini, 18 data diantaranya digunakan untuk *training* dan sisanya sebagai *testing*. Data tersebut sebelumnya akan dinamai coba, berikut ini adalah data yang digunakan:

$$p = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9 \ 10]$$

$$t = [0 \ 1 \ 2 \ 3 \ 4 \ 3 \ 2 \ 1 \ 2 \ 3 \ 4]$$

dimana  $p$  adalah vektor input dan vektor  $t$  adalah targetnya.

Pengujian yang dilakukan, menggunakan rincian sebagai berikut:

$$p = \text{coba} (1:9,1)$$

$$t = \text{coba} (1:9,2)$$

$$a = \text{coba} (10:11,1)$$

$$s = \text{coba} (10:11,2)$$

$$S1 = 5$$

$$S2 = 1$$

$$TF1 = \text{tansig atau tangent sigmoid}$$

$$TF2 = \text{linear atau purelin}$$

$$BTF = \text{gradient descent with momentum}$$

$$BLF = \text{gradient descent with momentum}$$

$$\text{epoch} = 50$$

$$\eta = 0.3$$

$$mc = 0.6$$

Dari pengujian yang dilakukan, diperoleh bahwa *syntax Artificial Neural Network* yang terdapat dalam Matlab dapat berjalan dan hasil *running* untuk data diatas dapat dilihat pada Lampiran 4.

#### 4.2.2.2 Rancangan Awal Artificial Neural Network (ANN)

Berikut ini merupakan rancangan awal yang digunakan dalam penelitian untuk melakukan pengolahan data. Rancangan ini akan disesuaikan dengan kondisi yang ada yakni hasil *running* yang berupa nilai *error*. Nilai *error* ini diperoleh dengan melihat nilai *Mean Square Error* (MSE),  $MSE = \frac{\sum(y-\bar{y})^2}{N}$  yang dihasilkan oleh setiap kombinasi parameter ANN. Kombinasi parameter yang menghasilkan nilai *error* terkecil merupakan kombinasi

terbaik yang akan digunakan untuk estimasi manhours aktual. Dalam menghitung MSE ini, dimasukkan skenario penalti sbb: “Jika hasil estimasi  $>$  *actual target (error positif)* maka akan dikenakan penalti sebesar 40 dan jika hasil estimasi  $<$  *actual target (error negatif)* juga akan dikenakan penalti sebesar 40.”

Penentuan ini berdasarkan pertimbangan bahwa perusahaan akan mengalami kerugian \$40 untuk setiap manhours yang berlebih ataupun kekurangan manhours karena biaya untuk tiap manhours sebesar \$40. *Syntax MSE* ini bisa dilihat di Lampiran 5. Berikut ini adalah rancangan parameter ANN yang digunakan dalam penelitian ini:

### Pre-Processing

#### Input pre-processing

Data training

Target training

Data testing

Target testing

#### Output pre-processing

Data training hasil pre-processing

Target training hasil pre-processing

Data testing hasil pre-processing

Target testing hasil pre-processing

Input Matlab

= p

= t

= s

= y

Output Matlab

= pn

= tn

= an

= sn

### Feedforward network

#### Input feedforward network

PR = minmax(pn)

Jumlah layer = 3

Jumlah node tiap layer (Si)

▪ Hidden Layer (S1) = 3, 5, 7, 10

▪ Output Layer (S2) = 1

Fungsi aktivasi (TFi)

▪ Hidden function (TF1) = tansig

▪ Output function (TF2) = purelin

Fungsi training (BTF)

- *Gradient descent with momentum=traingdm*
- Fungsi learning (BLF)
- *Gradient descent with momentum=learngdm*

### Training Network and Post-Processing

#### Input training

Momentum (*mc*)

Learning rate ( $\eta$ )

Iterasi (epoch)

#### Output training

Hasil estimasi

### Testing and Post-Processing

#### Output testing

Hasil estimasi

Hasil Error

*Input Matlab*

= 0.5, 0.6, 0.7

= 0.1, 0.3, 0.5,  
0.7, 0.9

= 50, 100, 250

*Output Matlab*

= x

*Output Matlab*

= x

= mse

Data yang digunakan dalam pengolahan data menggunakan metode ANN ini merupakan data project 5N-DDK tipe pesawat B747-200 C06-Check yang berupa MPD *manhours*, *planning manhours*, dan *actual manhours* yang bisa dilihat pada Lampiran 1. Proses pengolahan datanya adalah sebagai berikut :

```
>> load mh747.txt;
>> p=mh747(1:740,1:2);
>> t=mh747(1:740,3);
>> a=mh747(741:925,1:2);
>> s=mh747(741:925,3);
>> x=nn(p,t,a,s,epoch,lr,mc);
>> mse=mserror(x,s,'l')
```

Rincian *Matlab code* untuk *nn(neural network)* dapat dilihat pada Lampiran 6, sedangkan contoh hasil pengolahan data secara detail untuk hasil estimasi terbaik dapat dilihat pada Lampiran 7. Untuk mendapatkan nilai MSE akhir dilakukan 5 kali running berdasarkan perhitungan jumlah sampel pada persamaan

3.1. Dari data aktual *manhours* diperoleh bahwa simpangan baku populasi  $\sigma = 0.058$ . Maka jika diinginkan tingkat kepercayaan sebesar 95%, maka besar replikasi atau *running* yang diperlukan adalah:

$$n = \left( \frac{1.96 \times 0.058}{0.05} \right)^2 = 5.16926$$

Sedangkan hasil resume estimasinya dapat dilihat pada Tabel 4.11.

Tabel 4.11 Performansi MSE dengan metode ANN

Epoch		50				100				250			
mc	Si	3	5	7	10	3	5	7	10	3	5	7	10
	$\eta$												
0.5	0.1	50.40	59.01	62.79	69.08	39.56	53.61	59.12	61.23	34.33	35.29	46.35	52.73
	0.3	25.77	50.73	49.33	56.71	35.13	33.19	40.98	59.16	21.71	37.49	39.80	53.97
	0.5	46.00	68.10	Inf	Inf	25.83	49.99	inf	inf	29.11	42.78	inf	Inf
	0.7	67.23	Inf	Inf	Inf	46.33	Inf	Inf	inf	141.09	Inf	inf	inf
	0.9	Inf	Inf	Inf	Inf	Inf	Inf	inf	inf	Inf	Inf	inf	inf
0.6	0.1	42.47	58.99	46.36	70.03	33.23	38.64	49.61	66.26	25.10	31.66	39.56	50.22
	0.3	32.61	62.88	78.92	297.44	32.12	32.06	37.38	47.59	31.52	30.90	57.77	296.40
	0.5	43.82	200.57	372.08	Inf	38.68	35.95	215.30	inf	35.87	72.11	1195.81	Inf
	0.7	45.72	Inf	Inf	Inf	43.18	Inf	inf	inf	31.54	Inf	inf	inf
	0.9	48.42	Inf	Inf	Inf	57.06	Inf	inf	inf	123.82	Inf	inf	inf
0.7	0.1	33.44	40.44	42.51	66.66	39.98	37.95	56.06	65.93	28.67	43.93	33.73	57.83
	0.3	39.72	61.93	66.47	199.56	25.98	71.24	82.35	88.43	28.69	31.91	78.86	66.26
	0.5	90.90	26.90	93.94	119.67	29.43	34.12	1346.70	8179.38	29.77	30.70	122.65	352.42
	0.7	36.48	28.51	2690.03	Inf	61.27	85.66	3989.26	inf	259.22	134.83	9553.51	inf
	0.9	87.50	2314.32	Inf	Inf	33.15	716.35	Inf	inf	44.18	2824.97	Inf	inf

Tabel 4.11 menunjukkan bahwa dengan semakin bertambahnya *epoch* maka nilai MSE akan semakin kecil, namun dalam penelitian ini tidak akan digunakan 250 iterasi karena memungkinkan terjadi *overfitting* yakni menghasilkan *error* yang sangat kecil pada tahap *training* tetapi akan menghasilkan *error* yang tinggi dalam validasi. *Error* yang dihasilkan oleh 100 iterasi lebih kecil daripada *error* yang dihasilkan oleh 50 iterasi, dan dari grafik yang ditunjukkan pada lampiran 7 menunjukkan bahwa grafik cenderung konstan disekitar 100 iterasi sehingga dalam penelitian ini digunakan iterasi yang berjumlah 100. Jika dilihat berdasarkan nilai momentumnya diketahui bahwa *error* terkecil yang dihasilkan berada pada momentum 0.5, sehingga digunakan momentum 0.5 pada kombinasi selanjutnya. Semakin besar *learning rate*, maka semakin besar pula *error* yang dihasilkan sehingga untuk penelitian selanjutnya diberikan nilai *learning rate* 0.1 hingga 0.5 serta dilakukan penambahan jumlah node untuk mengetahui pengaruhnya terhadap *error* yang dihasilkan. Rincian perhitungan MSE untuk kombinasi baru dengan 5 replikasi diperlihatkan pada Lampiran 8, sedangkan rekapan performansi MSE-nya ditunjukkan pada Tabel 4.12.

Tabel 4.12 Performansi MSE dengan kombinasi baru

Epoch		100				
mc	$\eta$	0.1	0.2	0.3	0.4	0.5
	Si					
0.5	3	28.00	27.65	25.30	42.94	35.56
	5	38.90	41.44	36.37	30.72	47.69
	7	40.30	44.00	41.55	Inf	Inf
	10	67.79	52.49	Inf	Inf	Inf
	15	94.54	58.68	Inf	Inf	Inf
	20	110.42	Inf	Inf	Inf	Inf

Dari Tabel 4.12 diatas, maka kombinasi terbaik yang bisa digunakan untuk melakukan testing data pada *project* 5N-DDK adalah 100 kali iterasi, jumlah node 3 buah, *learning rate* 0.3 dan momentum sebesar 0.5. Dari kombinasi tersebut dilakukan

*running* untuk mendapatkan estimasi aktual *manhours* dengan perencanaan *manhours* yang dibuat pada alternatif A, alternatif B, dan alternatif C. Dari nilai estimasi *manhours* tersebut dibuat perbandingan terhadap total *plan*, *actual*, selisih keduanya serta pengujian *paired-t test* antara kondisi existing dengan ketiga alternatif tersebut. Contoh rincian perhitungan *paired-t test* bisa dilihat pada Lampiran 2 dan hasil rekapan perhitungannya diberikan pada Tabel 4.13.

Tabel 4.13 Parbandingan total *manhours* dan hasil *paired-t test* antara kondisi *existing* dan alternatif

		Existing	Alternatif		
			A	B	C
Total Manhours	MPD	1185	1185	1185	1185
	Plan (PL)	1679	1535	1512	1503
	Actual (AC)	2020	1490	1508	1509
	AC-PL	341	-45	-4	6
Paired-t test	Ex-A	Terima Ho			
	Ex-B	Tolak Ho			
	Ex-C	Tolak Ho			

Dari Tabel 4.13 diatas diketahui bahwa alternatif terbaik adalah alternatif A karena tidak cukup bukti untuk mengatakan bahwa kondisi *existing* dan kondisi alternatif A memiliki perbedaan yang signifikan dengan kata lain terima Ho. Dilakukan perhitungan nilai efisiensi untuk mengetahui performansi kondisi *existing* dengan alternatif A seperti yang ditunjukkan pada Tabel 4.14. Efisiensi yang didefinisikan oleh Renoir adalah perbandingan antara *planning manhours (PL)* dengan *actual manhours (AC)*.

Tabel 4.14 Perbandingan *efficiency* kondisi *existing* dengan alternatif A

	Existing	Alternatif A
Plan (PL)	1679	1535
Actual (AC)	2020	1490
Efficiency (PL/AC)	83%	103%

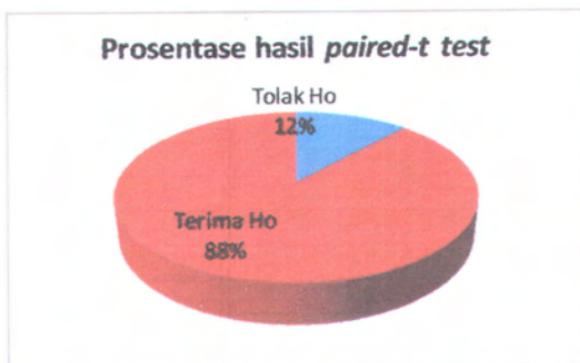
Selanjutnya dilakukan proses pengujian validasi dengan menggunakan *paired-t test* antara *planning manhours* dan *actual manhours* untuk alternatif A dengan menggunakan *software minitab 14* terhadap setiap MPD yang ada sehingga diperoleh hasil sebagai berikut:

Tabel 4.15 Selisih faktor pengali dan hasil *paired-t test* pada alternatif terbaik

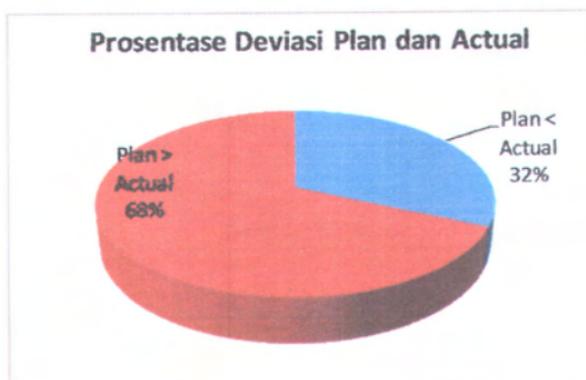
MPD	Faktor Pengali			Hasil Paired-t Test
	Plan (PL)	Actual (AC)	AC-PL	
0.1	4.00	3.70	-0.30	tolak Ho
0.2	2.75	2.42	-0.33	terima Ho
0.3	2.33	2.19	-0.14	terima Ho
0.4	2.13	2.08	-0.04	terima Ho
0.5	2.00	2.09	0.09	tolak Ho
0.6	1.83	1.69	-0.14	terima Ho
0.7	1.71	1.28	-0.43	terima Ho
0.8	1.63	1.58	-0.05	terima Ho
0.9	1.56	1.86	0.30	terima Ho
1	1.50	1.52	0.02	tolak Ho
1.2	1.43	1.21	-0.22	terima Ho

MPD	Faktor Pengali			Hasil Paired-t Test
	Plan (PL)	Actual (AC)	AC-PL	
1.3	1.41	1.30	-0.11	terima Ho
1.5	1.37	1.58	0.21	terima Ho
1.6	1.35	1.66	0.31	terima Ho
1.8	1.32	0.73	-0.59	terima Ho
1.9	1.31	1.10	-0.21	terima Ho
2	1.30	1.27	-0.03	terima Ho
2.1	1.26	1.46	0.20	terima Ho
2.5	1.13	1.36	0.23	terima Ho
2.6	1.10	0.79	-0.31	terima Ho
2.7	1.07	0.44	-0.63	terima Ho
2.9	1.03	0.51	-0.52	terima Ho
3	1.01	1.01	0.00	terima Ho

Dari hasil perhitungan dan uji validasi pada Tabel 4.15 diatas maka bisa ditampilkan statistik deskriptifnya pada Gambar 4.3 dan Gambar 4.4 berikut ini.



Gambar 4.3 Prosentase hasil *paired-t* test alternatif terbaik



Gambar 4.4 Prosentase selisih *plan* dan *actual* alternatif terbaik

(halaman ini sengaja dikosongkan)

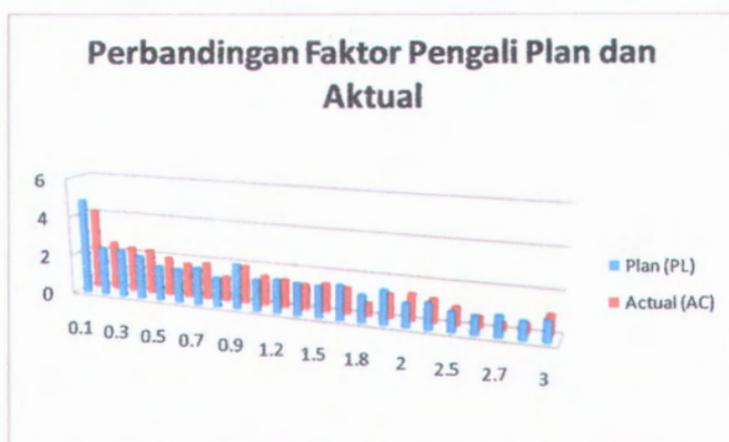
## BAB V

### ANALISIS DAN INTERPRETASI DATA

Pada tahap analisis dan interpretasi data ini akan dijelaskan mengenai interpretasi dari pengolahan data pada bab sebelumnya. Hasil interpretasi ini selanjutnya dapat digunakan untuk menarik kesimpulan serta mendapatkan saran perbaikan.

#### 5.1 Analisis Hasil *Linear Programming* (LP)

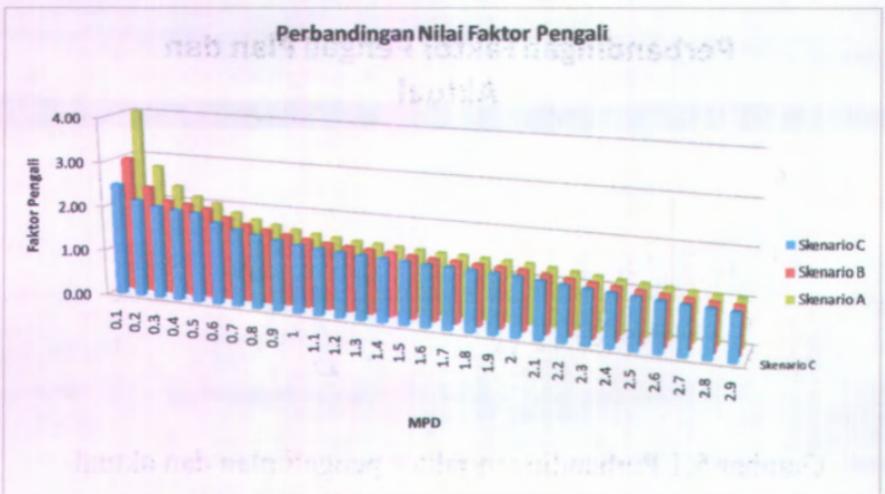
Pada awal pengolahan data menggunakan metode *Linear Programming* (LP) dilakukan terlebih dahulu penentuan konstrain melalui perhitungan selisih antara faktor pengali aktual terhadap faktor pengali yang direncanakan serta melihat hasil *paired-t test* antara *planning manhours* dan *actual manhours*. Gambar 5.1 menunjukkan perbandingan antara faktor pengali *plan* dan aktual.



Gambar 5.1 Perbandingan faktor pengali plan dan aktual

Pada Gambar 5.1 menunjukkan bahwa lebih banyak MPD yang diberikan perencanaan faktor pengali lebih besar dari faktor pengali aktual yang terjadi di lapangan. Jika dilihat pada Gambar 4.2 diketahui bahwa 80 % dari jenis MPD mengalami kelebihan *manhours* pada perencanaannya. Angka ini setara dengan 68% dari seluruh *jobcard* yang diturunkan ternyata mengalami

perencanaan yang lebih besar dari aktualnya. Artinya dalam 925 *jobcard* yang ada ternyata 629 *jobcard* diantaranya mengalami kelebihan manhours planning. Sedangkan dari Gambar 4.1 diketahui bahwa 60% dari jenis MPD yang dilakukan tidak dijumpai perbedaan yang signifikan terhadap kondisi aktual di lapangan. Artinya 60% dari perencanaan manhours yang dilakukan sudah sesuai dengan konsumsi manhours aktual di lapangan. Dengan berdasarkan preferensi *expert* perusahaan dan dengan mempertimbangkan hasil rekapan pada Tabel 4.7 maka konstrain untuk pengolahan data menggunakan *linear programming* bisa ditentukan dan dari hasil pengolahan data tersebut diketahui bahwa terdapat beberapa perbedaan dan persamaan terhadap nilai faktor pengali yang dihasilkan. Gambar 5.2 mendeskripsikan perbandingan antara ketiga alternatif faktor pengali tersebut.



Gambar 5.2 Perbandingan nilai faktor pengali antara alternatif A, alternatif B, dan alternatif C

Berdasarkan Gambar 5.2 dapat diketahui bahwa semakin besar nilai MPD maka nilai faktor pengali yang dihasilkan semakin kecil. Terdapat perbedaan nilai faktor pengali untuk *range* MPD 0.1 s/d 0.4 pada ketiga alternatif faktor pengali A,

alternatif B dan alternatif C, perbedaan ini muncul karena konstrain yang ada pada masing-masing alternatif pada *range* MPD 0.1 s/d 0.4 tidak sama. Sedangkan untuk *range* MPD 0.5 s/d 3 tidak terdapat perbedaan nilai faktor pengali dari ketiga alternatif tersebut karena konstrain yang ada antara ketiganya adalah sama. Hal yang diinginkan perusahaan adalah semakin kecil nilai faktor pengali, maka semakin baik karena semakin kecil waktu yang direncanakan untuk menyelesaikan suatu pekerjaan. Namun dalam penentuan perencanaan *manhours* ini harus disesuaikan dengan kondisi masing-masing *jobcard*, memperhatikan performansi mekanik terhadap perencanaan *manhours existing* yang telah dibuat oleh *planner* pada *project* sebelumnya dan terus dilakukan perbaikan-perbaikan dalam perencanaan tersebut jika diketahui terdapat selisih yang cukup besar antara *planning manhours* dan *actual manhours*.

Dari hasil tersebut dibuat fungsi IF-AND untuk mempermudah melakukan pengolahan data karena *manhours* yang dicantumkan di *planslip* ketika mekanik bekerja adalah *planning manhours* yang merupakan hasil perkalian antara MPD dan faktor pengali, sedangkan *jobcard* yang ada pada *C-Check Maintenance* sangat banyak. Selain itu, *planning manhours* ini juga digunakan sebagai bahan pengolahan data dalam menentukan estimasi *actual manhours* pada tahap selanjutnya yakni sebagai inputan data *Artificial Neural Network (ANN)*.

## 5.2 Analisis Hasil Artificial Neural Network (ANN)

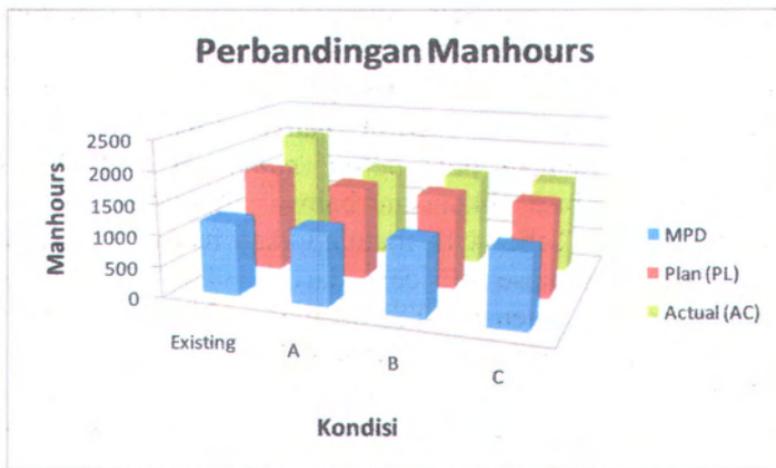
Dari hasil perhitungan performansi MSE yang diperoleh diawal penelitian yang ditampilkan pada Tabel 4.11 diketahui bahwa MSE terkecil pada kombinasi-kombinasi yang ada dicapai pada *learning rate* yang kecil yakni 0.1, 0.3 dan 0.5, dengan node yang kecil pula yakni 3 dan 5. Semakin besar *iterasi/epoch* yang diberikan maka semakin kecil nilai MSE yang dihasilkan. Namun, semakin besar *epoch* yang diberikan akan memungkinkan terjadinya *overfitting* yakni menghasilkan *error* yang sangat kecil pada tahap *training* tetapi akan menghasilkan *error* yang tinggi

dalam validasi. Sehingga tidak digunakan *epoch* yang berjumlah 250. Sedangkan MSE yang dihasilkan pada *epoch* 50 dan 100 tidak terlalu berbeda jauh. Namun jika dilihat pada grafik yang dihasilkan (lihat Lampiran 7) diketahui bahwa pada epoch 50 masih mengalami penurunan nilai MSE sedangkan pada epoch 100 nilai MSE yang dihasilkan cenderung konstan dan tidak mengalami penurunan yang drastis. Semakin tinggi nilai momentum, maka MSE yang dihasilkan semakin kecil. Hal ini berbeda dengan pengaruh nilai *learning rate*, semakin besar *learning rate* maka semakin besar juga MSE yang dihasilkan. Hal ini terjadi dikarenakan pada *learning rate* yang besar banyak dicapai titik optimum pada saat *training* sehingga *error* yang dihasilkan juga semakin besar. Sedangkan pengaruh node yang terlihat adalah semakin besar node yang diberikan maka semakin besar pula MSE yang dihasilkan.

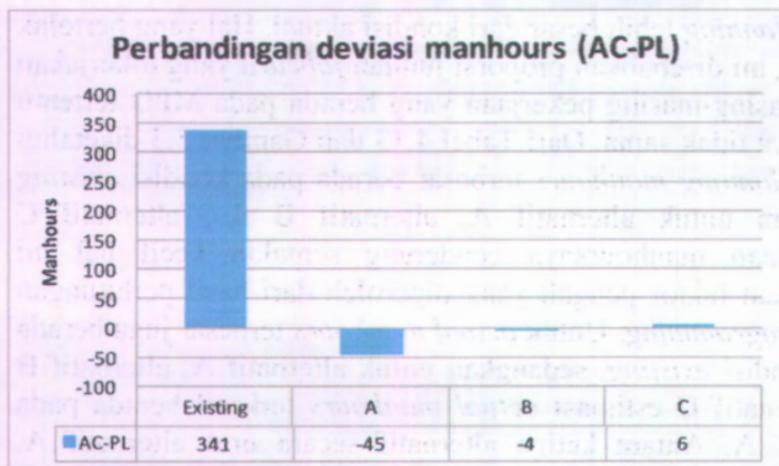
Melihat pengaruh parameter-parameter ANN terhadap MSE yang dihasilkan maka dilakukan perhitungan dengan kombinasi baru seperti yang ditunjukkan pada Tabel 4.12 sehingga diperoleh hasil bahwa kombinasi terbaiknya adalah 100 kali iterasi, jumlah node 3, learning 0.3 dan momentum 0.5 dengan nilai MSE sebesar 25.30. Kombinasi tersebut diterapkan pada model *Artificial Neural Network* (ANN) dengan memasukkan nilai MPD dan *planning manhours* yang dihasilkan dari alternatif A, alternatif B dan alternatif C untuk mendapatkan nilai estimasi *manhours* aktual untuk masing-masing *jobcard* yang ada.

Setelah diperoleh nilai estimasi *actual manhours* maka dilakukan perbandingan terhadap total *manhours* dan dilakukan uji validasi *paired-t test* antara kondisi *existing* dengan ketiga alternatif yang ada. Pada kondisi *existing* menunjukkan bahwa selisih antara *manhours* yang direncanakan dengan *actual manhours* sebesar 341 *manhours* dimana *actual manhours* yang dihabiskan untuk menyelesaikan *project* lebih besar daripada *manhours* yang direncanakan, namun jika dilihat selisih faktor pengalinya 80% dari total jenis MPD yang dianalisa menunjukkan

bahwa *planning* lebih besar dari kondisi aktual. Hal yang bertolak belakang ini disebabkan proporsi jumlah *jobcard* yang dikerjakan untuk masing-masing pekerjaan yang berada pada MPD tertentu jumlahnya tidak sama. Dari Tabel 4.13 dan Gambar 5.3 diketahui bahwa *planning manhours* terbesar berada pada kondisi *existing* sedangkan untuk alternatif A, alternatif B dan alternatif C perencanaan *manhours*nya cenderung semakin kecil hal ini diakibatkan faktor pengali yang diperoleh dari hasil perhitungan *linear programming*. Untuk *actual manhours* terbesar juga berada pada kondisi *existing*, sedangkan untuk alternatif A, alternatif B dan alternatif C estimasi *actual manhours* terkecil berada pada alternatif A. Antara ketiga alternatif secara urut alternatif A, alternatif B dan alternatif C diketahui bahwa semakin kecil *planning manhours* yang diberikan maka estimasi *actual manhours* yang terjadi semakin besar.



Gambar 5.3 Perbandingan *manhours*

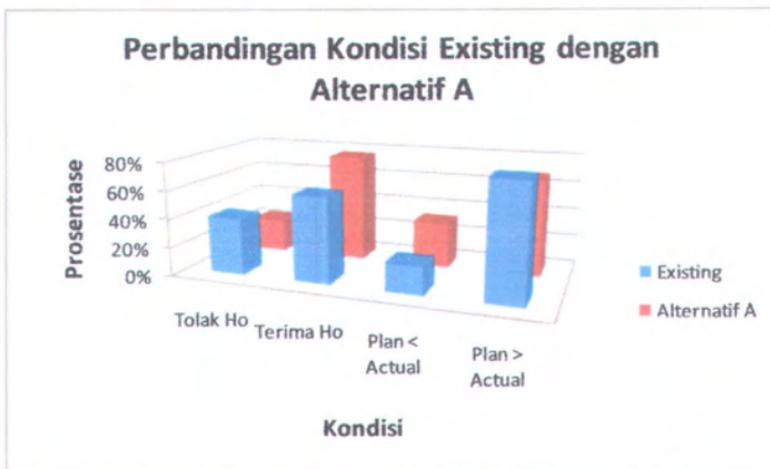


Gambar 5.4 Perbandingan selisih *manhours*

Berdasarkan Tabel 4.13 dan Gambar 5.4 diketahui bahwa selisih *planning manhours* dan *actual manhours* terkecil adalah alternatif B, diikuti alternatif C, alternatif A dan kondisi *existing*. Namun ketika dilakukan uji validasi pada model estimasi *actual manhours* diperoleh hasil bahwa hanya alternatif A yang dikatakan sesuai dengan pola *actual manhours* pada kondisi *existing* sehingga bisa dikatakan bahwa alternatif terbaiknya adalah alternatif A. Tabel 4.14 menunjukkan bahwa alternatif A meningkatkan prosentase efisiensi dari kondisi *existing* yang semula 83% (rasio efisiensi = 0.8) menjadi 103% (rasio efisiensi = 1.03). Kondisi *existing* menunjukkan bahwa rasio efisiensi yang dihasilkan kurang dari 1 yang artinya efisiensi yang diharapkan belum tercapai. Sedangkan alterbnatif A menghasilkan rasio efisiensi yang sudah lebih dari 1 yang artinya efisiensi yang diharapkan telah tercapai. Hal ini cukup berbeda dengan hasil perhitungan efisiensi oleh Renoir, yakni efisiensi pada proyek 5N-BMG adalah 249%. Hal ini menunjukkan bahwa hasil analisa suatu *project maintenance* belum tentu bisa mewakili karakter *maintenance* satu tipe pesawat. Terbukti dengan hasil yang diperoleh dari penelitian ini bahwa pada tipe *maintenance* dan

registrasi pesawat yang berbeda juga akan mendapatkan hasil yang berbeda walaupun jenis pesawatnya sama.

Setelah diketahui bahwa alternatif A adalah alternatif terbaik maka dilakukan uji validasi berdasarkan pengelompokan MPD yang hasilnya ditampilkan dalam Tabel 4.15. Dari tabel tersebut masih dijumpai di tiga titik MPD yang dikatakan memiliki perbedaan antara model estimasi *actual manhours* yang dibuat dengan kondisi *actual manhours* pada kondisi *existing*, ketiga MPD tersebut yakni 0.1, 0.5 dan 1. Namun secara umum jika dilihat pada Gambar 4.3 dan Gambar 4.4 diketahui bahwa 88% dari model yang dibuat telah sesuai dengan kondisi *existing* dan kondisi *planning manhours* lebih besar dari *actual manhours* juga masih terjadi dengan besar prosentase 68%.



Gambar 5. 5 Perbandingan kondisi *existing* dengan alternatif A

Dari Tabel 4.13 dan Gambar 5.5 diketahui bahwa alternatif A meningkatkan prosentase terima Ho dari 60% menjadi 80% yang menunjukkan bahwa tidak cukup bukti untuk mengatakan bahwa terdapat perbedaan yang signifikan antara *planning manhours* dengan *actual manhours*. Pada kondisi *existing*, 60% terima Ho tersebut menyebabkan adanya deviasi 341 *manhours* yang setara dengan \$13.640 jika menggunakan

harga satu satuan *manhours* sebesar \$40. Nilai \$13.640 merupakan biaya yang dikeluarkan perusahaan untuk menggaji *manhours* yang menganggur sebagai akibat dari perencanaan *manhours* yang melebihi *manhours actual*. Dan dari alternatif A dihasilkan 80% terima  $H_0$  yang menghasilkan deviasi antara *manhours planning* dan *manhours actual* sebesar 45 *manhours* dimana *manhours actual* lebih besar dari *manhours planning*. 45 *manhours* ini setara dengan biaya \$1800 yang harus dikeluarkan untuk biaya kekurangan *manhours* jika satu satuan *manhours* seharga \$40. Hal ini akan membawa manfaat bagi perusahaan sehingga bisa mengurangi biaya yang dikeluarkan dari yang semula \$13.640 menjadi \$1800 atau bisa dikatakan perusahaan dapat lebih menghemat biaya sebesar \$11.840.

## BAB VI KESIMPULAN DAN SARAN

Pada tahap ini akan diuraikan mengenai kesimpulan dan saran berdasarkan perhitungan dan analisis data yang dilakukan.

### 6.1 Kesimpulan

Kesimpulan yang dapat diambil pada penelitian ini diantaranya adalah :

1. MSE terkecil dicapai pada nilai 25.30 dengan kombinasi parameter ANN pada *learning rate* 0.3, *momentum* 0.5, jumlah *node* 3 dan 100 kali *epoch*.
2. Hasil *paired-t test* antara alternatif A, alternatif B dan alternatif C terhadap kondisi *existing* menunjukkan bahwa hanya alternatif A yang memiliki persamaan pola dengan kondisi *existing*.
3. Standard faktor pengali yang diperoleh untuk tiga puluh nilai MPD mulai dari 0.1 s/d 3 adalah 4.00, 2.75, 2.33, 2.13, 2.00, 1.83, 1.71, 1.63, 1.56, 1.50, 1.46, 1.43, 1.41, 1.39, 1.37, 1.35, 1.34, 1.32, 1.31, 1.30, 1.26, 1.22, 1.19, 1.16, 1.13, 1.10, 1.07, 1.05, 1.03, dan 1.01.
4. Alternatif A merupakan alternatif terbaik dalam perencanaan faktor pengali *manhours* dan memberikan performansi lebih baik dari kondisi *existing* karena meningkatkan efisiensi yang semula 83% menjadi 103%.
5. Alternatif A meningkatkan prosentase terima  $H_0$  dari 60% menjadi 80% dan menurunkan kelebihan prosentase *planning manhours* yang lebih besar dari *actual manhours* dari 80% menjadi 68%.
6. Hasil *paired-t test* antara *planning manhours* dan *actual manhours* pada alternatif A menunjukkan bahwa masih dijumpai perbedaan yang signifikan pada MPD 0.1, 0,5 dan 1.

7. Hasil analisa suatu *project* dengan tipe *maintenance* dan registrasi pesawat tertentu belum bisa mewakili kepentingan satu jenis pesawat.

## 6.2 Saran

Berikut adalah saran penulis terkait dengan penelitian selanjutnya:

1. Pada proses pengambilan data hendaknya dilakukan pada saat proses *maintenance* amatan sedang berjalan bukan hanya data historis sehingga peneliti bisa mengikuti satu *project* secara utuh dan bisa mengetahui secara pasti sistem yang berjalan saat itu seperti apa.
2. Perlu dilakukan pendisiplinan pada proses penginputan data agar tidak banyak *missing value* dan memenuhi persyaratan untuk dilakukan analisis.
3. Perlu dilakukan analisis terhadap kesesuaian *planning manhours* dan *actual manhours* setiap kali suatu *project* selesai dilaksanakan, sehingga bisa dilakukan *update-an* informasi dan kebijakan.

## DAFTAR PUSTAKA

- Angkiriwang Reina. 2009. "Peramalan Kebutuhan Bahan Baku dengan Mempertimbangkan Tingkat Commonality dan Resiko". Thesis Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember, Surabaya.
- Dowler, D. John. 2008. "Using Neural Network with Limited Data to Estimate Manufacturing Cost". Thesis of the Russ College of Engineering and Technology of Ohio University.
- Karaborka, H, Baykanb, O.K., Altuntasa, C., dan Yildza F. 2008. "Estimation of unknown height with Artificial Neural Network on Digital Terrain Model". The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol. XXXVII. Part B3b. Beijing, 2008.
- Kisi Ozgur. 2005. "Suspended sediment estimation using neuro-fuzzy". Hydrological Sciences–Journal–des Sciences Hydrologiques, 50(4), August 2005.
- Liu Bin dan Jiang Zu-Hua. 2005. "The Man-hour Estimation Models & Its Comparison of Interim Products Assembly for Shipbuilding". International Journal of Operations Research, Vol. 2, No. 1, 9-14, 2005.
- Multiply. (2010). Uji Hipotesis Komparatif. <URL:<http://images.dedi1968.multiply.multiplycontent.com/attachment/0/SNSPkwoKCEEAAADD4NQY1/Membaca%2520t-tes%5B1%5D.pdf?nmid=116333237>>, diakses pada tanggal 6 Januari 2010.
- Santosa, Budi. (2007). Data Mining Terapan dengan Matlab. Yogyakarta: Graha Ilmu.
- Santosa, Budi. (2007). Data Mining : Teknik Pemanfaatan Data untuk Keperluan Bisnis. Yogyakarta: Graha Ilmu.

Santosa, Budi. (2008). *Manajemen Proyek, Konsep dan Implementasi*. Yogyakarta: Graha Ilmu.

Siswanto. (2007). *Operations Research*. Erlangga.

Thirunavukarasu. P. 2009. " Estimation of Return on Investment in Share Market through ANN". *Global Journal of Finance and Management*. ISSN 0975 - 6477 Volume 1, Number 2 (2009), pp. 113-122.

## LAMPIRAN

### Lampiran 1 : Contoh Data project 5N-DDK C06-Check Maintenance

No	J/C No	MPD (mhrs)	Plan (mhrs)	Actual (mhrs)
1	1-25-001	0.4	1	1
2	1-25-002	0.2	0.5	0.5
3	1-25-004	0.2	0.5	0.5
4	1-28-001	0.2	0.5	0.5
5	1-33-002	1	2	1.15
6	1-71-001-3	0.1	0.5	0.383
7	1-71-001-4	0.1	0.5	0.45
8	2-21-001-01	0.2	0.5	0.566
9	2-21-002-01	1	1.7	2.55
10	2-21-003-01	0.3	0.75	0.5
11	2-21-003-02	0.3	0.3	0.3
12	2-21-005-01	2.55	2.55	1
13	2-21-006-01	0.3	0.75	0.37
14	2-21-007-01	0.8	0.8	0.683
15	2-21-007-02	0.2	0.25	0.25
16	2-21-009-01	0.2	0.5	0.5
17	2-21-010-01	0.2	0.5	0.5
18	2-21-011-02	1	1.7	0.85
19	2-21-012-01	0.2	0.5	0.5
20	2-22-002-01	1	0.85	0.85
21	2-23-002-01	0.5	0.5	0.5
22	2-24-001-01	0.2	0.5	0.5
23	2-24-003-01-3	0.1	0.5	0.383

(halaman ini sengaja dikosongkan)

Anda dapat mengisi bagian ini dengan nama dan nomor :

No. Urut	Nama	Nilai	Nilai Rata-rata	Nilai Akhir
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
21				
22				
23				
24				
25				
26				
27				
28				
29				
30				
31				
32				
33				
34				
35				
36				
37				
38				
39				
40				
41				
42				
43				
44				
45				
46				
47				
48				
49				
50				
51				
52				
53				
54				
55				
56				
57				
58				
59				
60				
61				
62				
63				
64				
65				
66				
67				
68				
69				
70				
71				
72				
73				
74				
75				
76				
77				
78				
79				
80				
81				
82				
83				
84				
85				
86				
87				
88				
89				
90				
91				
92				
93				
94				
95				
96				
97				
98				
99				
100				

## Lampiran 2 : Contoh hasil *paired-t test*

Ho : Kolom *plan* dan *actual* memiliki rata-rata yang bernilai sama.

H1 : Terdapat kolom antara *plan* dan *actual* yang memiliki rata-rata yang bernilai tidak sama (berbeda).

➤ MPD = 0.1 mhrrs

P value (0.000) <  $\alpha(0.0500)$ , maka tolak Ho.

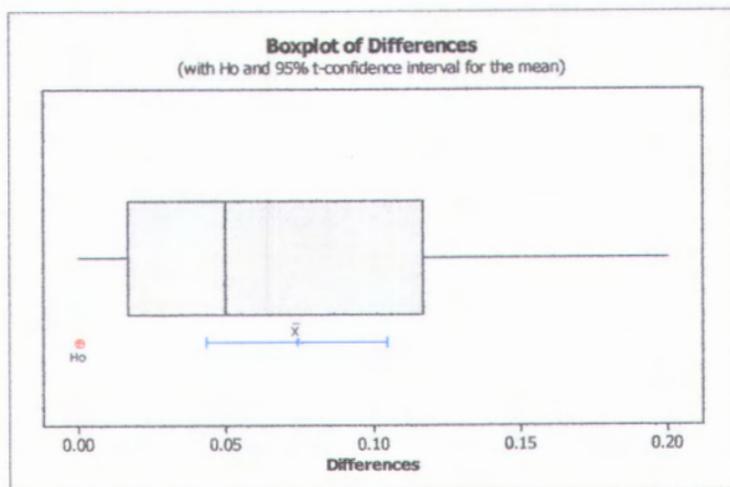
Kesimpulan : Terdapat kolom antara *plan* dan *actual* yang memiliki rata-rata yang bernilai tidak sama (berbeda).

Paired T for plan - actual

	N	Mean	StDev	SE Mean
plan	17	0.485294	0.060634	0.014706
actual	17	0.411471	0.070326	0.017057
Difference	17	0.073824	0.059802	0.014504

95% CI for mean difference: (0.043076, 0.104571)

T-Test of mean difference = 0 (vs not = 0): T-Value = 5.09 P-Value = 0.000



- Pengujian validasi kondisi *existing* dengan alternatif A  
 P value (0.421) >  $\alpha(0.0500)$ , maka terima  $H_0$ .

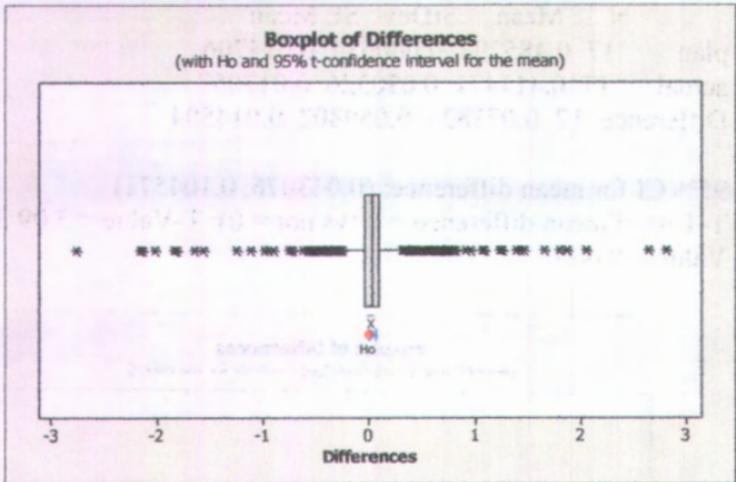
Kesimpulan : Kolom *plan* dan *actual* memiliki rata-rata yang bernilai sama.

Paired T for C1 - C2

	N	Mean	StDev	SE Mean
C1	925	1.61091	1.71483	0.05638
C2	925	1.58864	1.76479	0.05803
Difference	925	0.022263	0.443849	0.014594

95% CI for mean difference: (-0.006377, 0.050904)

T-Test of mean difference = 0 (vs not = 0): T-Value = 1.53 P-Value = 0.127



### Lampiran 3 : Pengolahan data penentuan faktor pengali dengan menggunakan *software* LINDO

✚ Skenario  $0.1 \leq MPD \leq 0.5$

D:\TATA-TAL

Min  $x_1+x_2+x_3+x_4+x_5$

ST

$x_1 = 4$   
 $x_5 = 2$

$x_1-x_2 > 0.01$   
 $x_2-x_3 > 0.01$   
 $x_3-x_4 > 0.01$   
 $x_4-x_5 > 0.01$

$0.4x_2 - 0.1x_1 - 0.3x_3 = 0$   
 $0.6x_3 - 0.2x_2 - 0.4x_4 = 0$   
 $0.8x_4 - 0.3x_3 - 0.5x_5 = 0$

END

LINDO

DO RANGE(SENSITIVITY) ANALYSIS?

LINDO Solver Status

Optimizer Status	
Status:	Optimal
Iterations:	0
Infeasibility:	0
Objective:	13.2083
Best IP:	N/A
IP Bound:	N/A
Branches:	N/A
Elapsed Time:	00:00:01

Update Interval:

Reports Window

LP OPTIMUM FOUND AT STEP 0

OBJECTIVE FUNCTION VALUE

1) 13.20833

VARIABLE	VALUE	REDUCED COST
X1	4.000000	0.000000
X2	2.750000	0.000000
X3	2.333333	0.000000
X4	2.125000	0.000000
X5	2.000000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	-1.604167
3)	0.000000	-3.395833
4)	1.240000	0.000000
5)	0.406667	0.000000
6)	0.198333	0.000000
7)	0.115000	0.000000
8)	0.000000	-6.041667
9)	0.000000	-7.083333
10)	0.000000	-4.791667

NO. ITERATIONS= 0

D:\DATA-1\TA...

Min  $x_1 + x_2 + x_3 + x_4 + x_5$

ST  $x_1 = 3$   
 $x_5 = 2$

$x_1 - x_2 > 0.01$   
 $x_2 - x_3 > 0.01$   
 $x_3 - x_4 > 0.01$   
 $x_4 - x_5 > 0.01$

$0.4x_2 - 0.1x_1 - 0.3x_3 = 0$   
 $0.6x_3 - 0.2x_2 - 0.4x_4 = 0$   
 $0.8x_4 - 0.3x_3 - 0.5x_5 = 0$

END

LINDO

DO RANGE(SENSITIVITY) ANALYSIS?

Yes No

LINDO Solver Status

Optimizer Status

Status:	Optimal
Iterations:	0
Infeasibility:	0
Objective:	11.6042
Best IP:	N/A
IP Bound:	N/A
Branches:	N/A
Elapsed Time:	00:00:02

Update Interval:

Reports Window

LP OPTIMUM FOUND AT STEP 0

OBJECTIVE FUNCTION VALUE

1) 11.60417

VARIABLE	VALUE	REDUCED COST
X1	3.000000	0.000000
X2	2.375000	0.000000
X3	2.166667	0.000000
X4	2.062500	0.000000
X5	2.000000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	-1.604167
3)	0.000000	-3.395833
4)	0.615000	0.000000
5)	0.198333	0.000000
6)	0.094167	0.000000
7)	0.052500	0.000000
8)	0.000000	-6.041667
9)	0.000000	-7.083333
10)	0.000000	-4.791667

NO. ITERATIONS= 0

```

D:\TATA-TA...
Min      x1+x2+x3+x4+x5

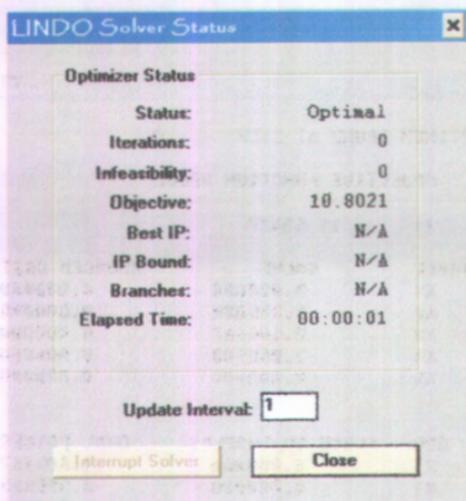
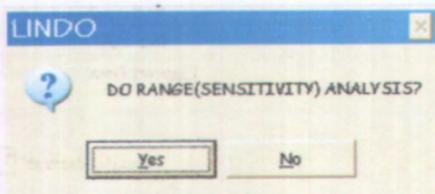
ST       x1 = 2.5
         x5 = 2

         x1-x2>0.01
         x2-x3>0.01
         x3-x4>0.01
         x4-x5>0.01

         0.4x2-0.1x1-0.3x3=0
         0.6x3-0.2x2-0.4x4=0
         0.8x4-0.3x3-0.5x5=0

END|
4

```



 Reports Window

LP OPTIMUM FOUND AT STEP 0

## OBJECTIVE FUNCTION VALUE

1) 10.80200

VARIABLE	VALUE	REDUCED COST
X1	2.500000	0.000000
X2	2.187500	0.000000
X3	2.083333	0.000000
X4	2.031250	0.000000
X5	2.000000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	-1.604167
3)	0.000000	-3.395833
4)	0.302500	0.000000
5)	0.094167	0.000000
6)	0.042083	0.000000
7)	0.021250	0.000000
8)	0.000000	-6.041667
9)	0.000000	-7.083333
10)	0.000000	-4.791667

NO. ITERATIONS= 0

 Skenario  $0.5 \leq MPD \leq 1$ 
 D:\TATA-1.TA\...
Min  $x5+x6+x7+x8+x9+x10$ ST  $x5 = 2$  $x10 = 1.5$  $x5-x6 > 0.01$  $x6-x7 > 0.01$  $x7-x8 > 0.01$  $x8-x9 > 0.01$  $x9-x10 > 0.01$  $1.2x6-0.5x5-0.7x7=0$  $1.4x7-0.6x6-0.8x8=0$  $1.6x8-0.7x7-0.9x9=0$  $1.8x9-0.8x8-x10=0$ 

END

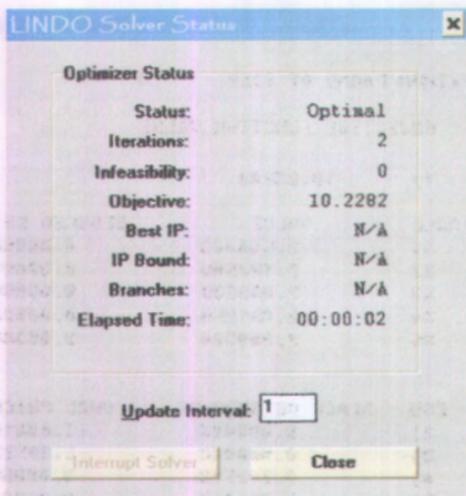
LINDO



DO RANGE(SENSITIVITY) ANALYSIS?

Yes

No



Reports Window \_ □ X

LP OPTIMUM FOUND AT STEP 2

OBJECTIVE FUNCTION VALUE

1) 10.22817

VARIABLE	VALUE	REDUCED COST
X5	2.000000	0.000000
X6	1.833333	0.000000
X7	1.714286	0.000000
X8	1.625000	0.000000
X9	1.555556	0.000000
X10	1.500000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	-2.456349
3)	0.000000	-3.543651
4)	0.156667	0.000000
5)	0.109048	0.000000
6)	0.079286	0.000000
7)	0.059444	0.000000
8)	0.045556	0.000000
9)	0.000000	-2.912699
10)	0.000000	-4.158730
11)	0.000000	-3.976191
12)	0.000000	-2.543651

NO. ITERATIONS= 2

✚ Skenario  $1 \leq MPD \leq 2$

D:\TATA-i-TAVOLA

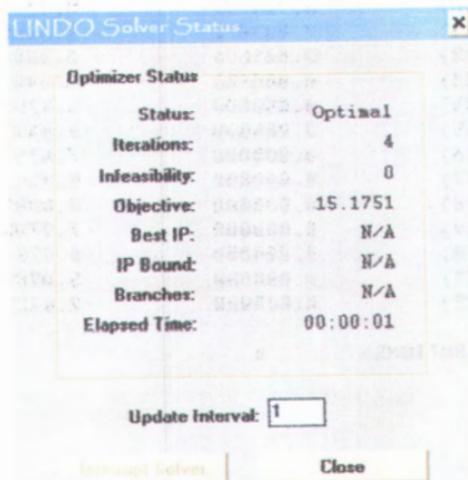
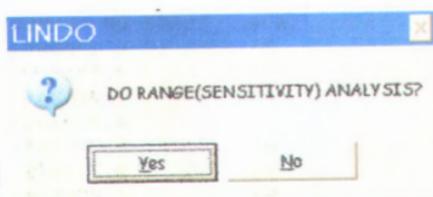
Min  $x_{10}+x_{11}+x_{12}+x_{13}+x_{14}+x_{15}$   
 $+x_{16}+x_{17}+x_{18}+x_{19}+x_{20}$

ST  $x_{10} = 1.5$   
 $x_{20} = 1.3$

$x_{10}-x_{11}>0.01$   
 $x_{11}-x_{12}>0.01$   
 $x_{12}-x_{13}>0.01$   
 $x_{13}-x_{14}>0.01$   
 $x_{14}-x_{15}>0.01$   
 $x_{15}-x_{16}>0.01$   
 $x_{16}-x_{17}>0.01$   
 $x_{17}-x_{18}>0.01$   
 $x_{18}-x_{19}>0.01$   
 $x_{19}-x_{20}>0.01$

$2.2x_{11}-x_{10}-1.2x_{12}=0$   
 $2.4x_{12}-1.1x_{11}-1.3x_{13}=0$   
 $2.6x_{13}-1.2x_{12}-1.4x_{14}=0$   
 $2.8x_{14}-1.3x_{13}-1.5x_{15}=0$   
 $3x_{15}-1.4x_{14}-1.6x_{16}=0$   
 $3.2x_{16}-1.5x_{15}-1.7x_{17}=0$   
 $3.4x_{17}-1.6x_{16}-1.8x_{18}=0$   
 $3.6x_{18}-1.7x_{17}-1.9x_{19}=0$   
 $3.8x_{19}-1.8x_{18}-2x_{20}=0$

END  
 4




 Reports Window

LP OPTIMUM FOUND AT STEP 4

## OBJECTIVE FUNCTION VALUE

1) 15.17509

VARIABLE	VALUE	REDUCED COST
X10	1.500000	0.000000
X11	1.463636	0.000000
X12	1.433333	0.000000
X13	1.407692	0.000000
X14	1.385714	0.000000
X15	1.366667	0.000000
X16	1.350000	0.000000
X17	1.335294	0.000000
X18	1.322222	0.000000
X19	1.310526	0.000000
X20	1.300000	0.000000
ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	-4.375428
3)	0.000000	-6.624572
4)	0.026364	0.000000
5)	0.020303	0.000000
6)	0.015641	0.000000
7)	0.011978	0.000000
8)	0.009048	0.000000
9)	0.006667	0.000000
10)	0.004706	0.000000
11)	0.003072	0.000000
12)	0.001696	0.000000
13)	0.000526	0.000000
14)	0.000000	-3.375428
15)	0.000000	-5.841765
16)	0.000000	-7.474769
17)	0.000000	-8.338542
18)	0.000000	-8.488029
19)	0.000000	-7.970050
20)	0.000000	-6.828671
21)	0.000000	-5.098256
22)	0.000000	-2.812286

NO. ITERATIONS= 4

✚ Skenario  $1 \leq MPD \leq 2$

D:\TATA-IT\LOL

Min  $x_{20} + x_{21} + x_{22} + x_{23} + x_{24} + x_{25} + x_{26} + x_{27} + x_{28} + x_{29} + x_{30}$

ST  $x_{20} = 1.3$   
 $x_{30} = 1.01$

$x_{20} - x_{21} > 0.01$   
 $x_{21} - x_{22} > 0.01$   
 $x_{22} - x_{23} > 0.01$   
 $x_{23} - x_{24} > 0.01$   
 $x_{24} - x_{25} > 0.01$   
 $x_{25} - x_{26} > 0.01$   
 $x_{26} - x_{27} > 0.01$   
 $x_{27} - x_{28} > 0.01$   
 $x_{28} - x_{29} > 0.01$   
 $x_{29} - x_{30} > 0.01$

$4.2x_{21} - 2x_{20} - 2.2x_{22} = 0$   
 $4.4x_{22} - 2.1x_{21} - 2.3x_{23} = 0$   
 $4.6x_{23} - 2.2x_{22} - 2.4x_{24} = 0$   
 $4.8x_{24} - 2.3x_{23} - 2.5x_{25} = 0$   
 $5x_{25} - 2.4x_{24} - 2.6x_{26} = 0$   
 $5.2x_{26} - 2.5x_{25} - 2.7x_{27} = 0$   
 $5.4x_{27} - 2.6x_{26} - 2.8x_{28} = 0$   
 $5.6x_{28} - 2.7x_{27} - 2.9x_{29} = 0$   
 $5.8x_{29} - 2.8x_{28} - 3x_{30} = 0$

END

LINDO

DO RANGE(SENSITIVITY) ANALYSIS?

Yes  No

LINDO Solver Status

Optimizer Status	
Status:	Optimal
Iterations:	0
Infeasibility:	0
Objective:	12.5121
Best IP:	N/A
IP Bound:	N/A
Branches:	N/A
Elapsed Time:	00:00:01

Update Interval:

Reports Window			
LP OPTIMUM FOUND AT STEP 0			
OBJECTIVE FUNCTION VALUE			
1)	12.51211		
VARIABLE	VALUE	REDUCED COST	
X20	1.300000	0.000000	
X21	1.258571	0.000000	
X22	1.220909	0.000000	
X23	1.186522	0.000000	
X24	1.155000	0.000000	
X25	1.126000	0.000000	
X26	1.099231	0.000000	
X27	1.074444	0.000000	
X28	1.051429	0.000000	
X29	1.030000	0.000000	
X30	1.010000	0.000000	
ROW	SLACK OR SURPLUS	DUAL PRICES	
2)	0.000000	-4.834848	
3)	0.000000	-6.165152	
4)	0.031429	0.000000	
5)	0.027662	0.000000	
6)	0.024387	0.000000	
7)	0.021522	0.000000	
8)	0.019000	0.000000	
9)	0.016769	0.000000	
10)	0.014786	0.000000	
11)	0.013016	0.000000	
12)	0.011429	0.000000	
13)	0.010000	0.000000	
14)	0.000000	-1.917424	
15)	0.000000	-3.358658	
16)	0.000000	-4.345346	
17)	0.000000	-4.897252	
18)	0.000000	-5.032491	
19)	0.000000	-4.767730	
20)	0.000000	-4.118353	
21)	0.000000	-3.098607	
22)	0.000000	-1.721717	
NO. ITERATIONS=		0	

#### Lampiran 4. Pengolahan Data Uji Coba *Syntax Artificial Neural Network*

```
>> load coba.txt
>> coba
```

```
coba =
```

```
0 0
1 1
2 2
3 3
4 4
5 3
6 2
7 1
8 2
9 3
10 4
```

```
>> p=coba(1:9,1);
>> t=coba(1:9,2);
>> a=coba(10:11,1);
>> s=coba(10:11,2);
>> [pn,minp,maxp,tn,mint,maxt]=premnmx(p',t')
```

```
pn =
```

```
Columns 1 through 6
```

```
-1.0000 -0.7500 -0.5000 -0.2500    0 0.2500
```

```
Columns 7 through 9
```

```
0.5000 0.7500 1.0000
```

```
minp =
```

```
0
```

```
maxp =
```

```
8
```

```
tr =
Columns 1 through 6
-1.0000 -0.5000 0 0.5000 1.0000 0.5000
```

```
Columns 7 through 9
0 -0.5000 0
```

```
mint =
0
```

```
maxt =
4
```

```
>> [an,minap,maxa,sn,mins,maxs]=premnmx(a',s')
```

```
an =
-1 1
```

```
minap =
9
```

```
maxa =
10
```

```
sn =
-1 1
```

```
mins =
3
```

```
maxs =
4
```

```
>> net=newff(minmax(pn),[5
1],{'tansig','purelin'},'traingdm','learnqdm')
```

```
net =
```

Neural Network object:

architecture:

```

numInputs      : 1
numLayers      : 2
biasConnect    : [1; 1]
inputConnect   : [1; 0]
layerConnect   : [0 0; 1 0]
outputConnect  : [0 1]
targetConnect  : [0 1]

```

```

numOutputs : 1 (read-only)
numTargets : 1 (read-only)
numInputDelays : 0 (read-only)
numLayerDelays : 0 (read-only)

```

subobject structures :

```

inputs      : {1x1 cell} of inputs
layers      : {2x1 cell} of layers
outputs     : {1x2 cell} containing 1 output
targets     : {1x2 cell} containing 1 target
biases      : {2x1 cell} containing 2 biases
inputWeights : {2x1 cell} containing 1 input weight
layerWeights : {2x2 cell} containing 1 layer weight

```

functions:

```

adaptFcn      : 'trains'
initFcn       : 'initlay'
performFcn    : 'mse'
trainFcn      : 'traingdm'

```

parameters:

```

adaptParam : .passes
initParam  : (none)

```

```

performParam      : (none)
trainParam       : .epochs, .goal, .lr, .max_fail,
                  .mc, .min_grad, .show, .time

```

weight and bias values:

```

IW      : {2x1 cell} containing 1 input weight matrix
LW      : {2x2 cell} containing 1 layer weight matrix
b       : {2x1 cell} containing 2 bias vectors

```

other:

```

userdata        : (user stuff)

```

```
>> net.trainParam.epochs=50
```

```
net =
```

Neural Network object:

```

architecture      : [1 1]
numInputs         : 1
numLayers         : 2
biasConnect       : [1; 1]
inputConnect      : [1; 0]
layerConnect      : [0 0; 1 0]
outputConnect     : [0 1]
targetConnect     : [0 1]

numOutputs        : 1 (read-only)
numTargets        : 1 (read-only)
numInputDelays    : 0 (read-only)
numLayerDelays    : 0 (read-only)

```

subobject structures:

```

inputs           : {1x1 cell} of inputs
layers           : {2x1 cell} of layers

```

```

    outputs          : {1x2 cell} containing 1 output
    targets         : {1x2 cell} containing 1 target
    biases          : {2x1 cell} containing 2 biases
inputWeights : {2x1 cell} containing 1 input weight
layerWeights : {2x2 cell} containing 1 layer weight

```

functions:

```

    adaptFcn        : 'trains'
    initFcn         : 'initlay'
performFcn : 'mse'
    trainFcn       : 'traingdm'

```

parameters:

```

    adaptParam : .passes
    initParam  : (none)
performParam  : (none)
    trainParam : .epochs, .goal, .lr, .max_fail,
                .mc, .min_grad, .show, .time

```

weight and bias values:

```

    IW          : {2x1 cell} containing 1 input weight matrix
    LW          : {2x2 cell} containing 1 layer weight matrix
    b           : {2x1 cell} containing 2 bias vectors

```

other:

```

    userdata: (user stuff)

```

```
>> net.trainParam.lr=0.3
```

```
net =
```

Neural Network object:

```
architecture :
```

```

numInputs : 1 (read-only)
numLayers : 2
biasConnect : [1; 1]
inputConnect : [1; 0]
layerConnect : [0 0; 1 0]
outputConnect : [0 1]
targetConnect : [0 1]

```

```

numOutputs : 1 (read-only)
numTargets : 1 (read-only)
numInputDelays : 0 (read-only)
numLayerDelays : 0 (read-only)

```

#### subobject structures:

```

inputs      : {1x1 cell} of inputs
layers      : {2x1 cell} of layers
outputs     : {1x2 cell} containing 1 output
targets     : {1x2 cell} containing 1 target
biases      : {2x1 cell} containing 2 biases
inputWeights : {2x1 cell} containing 1 input weight
layerWeights : {2x2 cell} containing 1 layer weight

```

```

functions
  adaptFcn : 'train'
  initFcn  : 'initlay'
  performFcn : 'mse'
  trainFcn : 'traingdm'

```

```

parameters
  adaptParam : .passes
  initParam  : (none)
  performParam : (none)
  trainParam : .epochs, .goal, .lr, .max_fail, .mc, .min_grad, .show, .time

```

#### weight and bias values:

```

    IW      : {2x1 cell} containing 1 input weight matrix
    LW      : {2x2 cell} containing 1 layer weight matrix
    b       : {2x1 cell} containing 2 bias vectors

```

```
other :
```

```
    userdata : (user stuff)
```

```
>> net.trainParam.mc=0.6
```

```
net =
```

```
Neural Network object:
```

```
architecture :
```

```

    numInputs      : 1
    numLayers      : 2
    biasConnect    : [1; 1]
    inputConnect   : [1; 0]
    layerConnect   : [0 0; 1 0]
    outputConnect  : [0 1]
    targetConnect  : [0 1]

```

```
    numOutputs : 1 (read-only)
```

```
    numTargets : 1 (read-only)
```

```
numInputDelays : 0 (read-only)
```

```
numLayerDelays : 0 (read-only)
```

```
subobject structures :
```

```
    inputs      : {1x1 cell} of inputs
```

```
    layers      : {2x1 cell} of layers
```

```
    outputs     : {1x2 cell} containing 1 output
```

```
    targets     : {1x2 cell} containing 1 target
```

```
    biases      : {2x1 cell} containing 2 biases
```

```
inputWeights : {2x1 cell} containing 1 input weight
```

```
layerWeights : {2x2 cell} containing 1 layer weight
```

```

functions : 'train' : 'performFcn' : 'mse'
           : 'initFcn' : 'initlay'
           : 'trainFcn' : 'traingdm'

parameters :

adaptParam : .passes
initParam   : (none)
performParam : (none)
trainParam  : .epochs, .goal, .lr, .max_fail,
              .mc, .min_grad, .show, .time

```

weight and bias values:

```

IW : {2x1 cell} containing 1 input weight matrix
LW : {2x2 cell} containing 1 layer weight matrix
B  : {2x1 cell} containing 2 bias vectors

other :

userdata : (user stuff)

```

```
>> net=train(net,pr,tn)
```

```

TRAININGDM, Epoch 0/50, MSE 3.41953/0, Gradient 4.61297/1e-010
TRAININGDM, Epoch 25/50, MSE 0.0385329/0, Gradient 0.047941/1e-010
TRAININGDM, Epoch 50/50, MSE 0.0246631/0, Gradient 0.0365669/1e-010
TRAININGDM, Maximum epochs reached, performance goal was not met

net =
  Neural Network object:
    input : 1 neurons (1x1x1)
    hidden : 2 neurons (1x2x1)
    output : 1 neurons (1x1x1)
    layer : 3 layers (1x2x1)
    weights : 1 parameters (1x2x1)
    bias : 2 parameters (1x2x1)

```

```

architecture      :
    numInputs      : 1
    numLayers      : 2
    biasConnect    : [1; 1]
    inputConnect   : [1; 0]
    layerConnect   : [0 0; 1 0]
    outputConnect  : [0 1]
    targetConnect  : [0 1]

```

```

    numOutputs : 1 (read-only)
    numTargets : 1 (read-only)
numInputDelays : 0 (read-only)
numLayerDelays : 0 (read-only)

```

subobject structures:

```

    inputs      : {1x1 cell} of inputs
    layers      : {2x1 cell} of layers
    outputs     : {1x2 cell} containing 1 output
    targets     : {1x2 cell} containing 1 target
    biases      : {2x1 cell} containing 2 biases
inputWeights  : {2x1 cell} containing 1 input weight
layerWeights  : {2x2 cell} containing 1 layer weight

```

functions :

```

    adaptFcn      : 'trains'
    initFcn       : 'initlay'
performFcn    : 'mse'
    trainFcn      : 'traingdm'

```

parameters :

```

    adaptParam : .passes
    initParam  : (none)
performParam  : (none)
    trainParam : .epochs, .goal, .lr, .max_fail,
                .mc, .min_grad, .show, .time

```

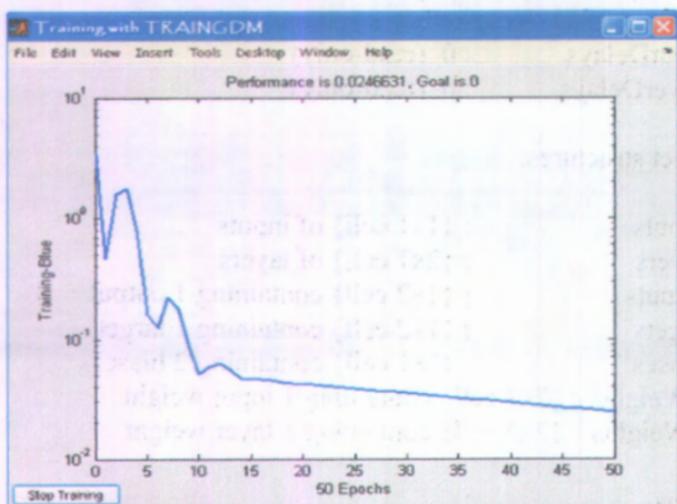


weight and bias values:

`IW` : {2x1 cell} containing 1 input weight matrix  
`LW` : {2x2 cell} containing 1 layer weight matrix  
`b` : {2x1 cell} containing 2 bias vectors

other

`userdata` : (user stuff)



```
>> y=sim(net,pn)
```

```
y =
```

Columns 1 through 6

```
-1.0311 -0.5558 0.1406 0.5871 0.6430 0.6376
```

Columns 7 through 9

```
0.0787 -0.3930 -0.1628
```

```
>> x=postmnmx(y',mint,maxt)
```

```
x =
```

```
-0.0622  
0.8885  
2.2812  
3.1742  
3.2859  
3.2752  
2.1575  
1.2139  
1.6745
```

```
>> y=sim(net,an)
```

```
y =
```

```
-1.0311 -0.1628
```

```
>> x=postmnmx(y',mins,maxs)
```

```
x =
```

```
2.9845  
3.4186
```

```
>> [x s]
```

```
ans =
```

```
2.9845 3.0000  
3.4186 4.0000
```

(halaman ini sengaja dikosongkan)

509-0

509-1

509-2

509-3

509-4

509-5

509-6

509-7

509-8

509-9

509-10

509-11

509-12

509-13

509-14

509-15

509-16

509-17

## Lampiran 5 : Syntax Mean Square Error

```

function mse=mserror(x,s,penalty)
%MSE Mean Square Error
%
%Usage: mse = mserror(x,s,penalty)
%
%parameter: x - Predictions (forecast)
%          s - Actual target
%          penalty - skenario
%
%skenario type: '0' - without penalty
%              '1' - penalty 40 if x>y and 40 if x<y

if (nargin ~=3) %check correct number of arguments
    help mserror
else
    selisih=x-s; %selisih antara prediksi dan actual target
    beda1=find(selisih>0); %mencari data yang bernilai positif
    beda2=find(selisih<0); %mencari data yang bernilai negatif
    positif=selisih(beda1); %data dengan nilai positif
    negatif=selisih(beda2); %data dengan nilai negatif
    a=size(x,1);

    switch lower (penalty)
        case '1'
            %total kuadrat selisih positif data aktual dan prediksi
            i=40*(sum(positif.^2));
            %total kuadrat selisih negatif data aktual dan prediksi
            j=40*(sum(negatif.^2));
            %rata-rata kuadrat selisih data aktual dan prediksi
            mse=(i+j)/a;
        otherwise
            mse = mean (selisih.^2);
    end
end
end

```

(halaman ini sengaja dikosongkan)

## Lampiran 6 : Matlab code of nn (neural network)

```

function x=nn(p,t,a,s,S1,epoch,lr,mc)
% usage: x=nn(p,t,a,s,S1,epoch,lr,mc)
% input parameter: S1 - Jumlah node pada hidden layer
%          epoch - Jumlah epoch/iterasi
%          lr - Learning rate
%          mc - besar gradient descent with momentum
% output: x - hasil estimasi

if ( nargin ~=8)
    help nn
else

% pre-processing
[pn,minp,maxp,tn,mint,maxt]=premnmx(p',t');
[an,mina,maxa,sn,mins,maxs]=premnmx(a',s');

% Feedforward Network
net=newff(minmax(pn),[S1 1],{'tansig','purelin'},'traingdm','learnqdm');

% Training dan Post-processing
net.trainParam.epochs=epoch;
net.trainParam.lr=lr;
net.trainParam.mc=mc;
net=train(net,pn,tn);
y=sim(net,pn);
x=postmnmx(y',mint,maxt);

% Testing dan Post-processing
y=sim(net,an);
x=postmnmx(y',mins,maxs);
[x s];
end

```

(halaman ini sengaja dikosongkan)

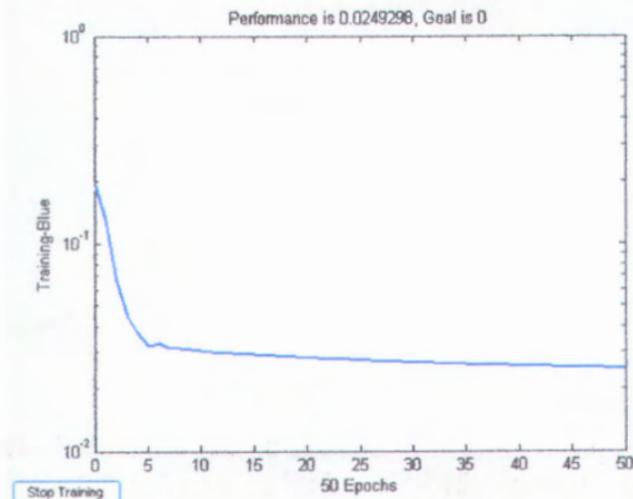
## Lampiran 7 : Contoh Hasil Proses Pengolahan Data Menggunakan ANN

```
>> x=nn(p,t,a,s,3,50,0.7,0.7);
TRAINGDM, Epoch 0/50, MSE 0.18808/0, Gradient 0.726276/1e-010
TRAINGDM, Epoch 25/50, MSE 0.0274113/0, Gradient 0.0217561/1e-010
TRAINGDM, Epoch 50/50, MSE 0.0249298/0, Gradient 0.0145543/1e-010
TRAINGDM, Maximum epoch reached, performance goal was not met.
```

```
>> mse=mserror(x,s,'l')
```

```
mse =
```

```
20.3042
```



```
>> x=nn(p,t,a,s,3,100,0.5,0.5);
TRAINGDM, Epoch 0/100, MSE 7.33859/0, Gradient 12.5596/1e-010
TRAINGDM, Epoch 25/100, MSE 0.0297775/0, Gradient 0.0107256/1e-010
TRAINGDM, Epoch 50/100, MSE 0.0290548/0, Gradient 0.0094055/1e-010
```

TRAININGM, Epoch 75/100, MSE 0.0284294/0, Gradient 0.0088539/1e-010

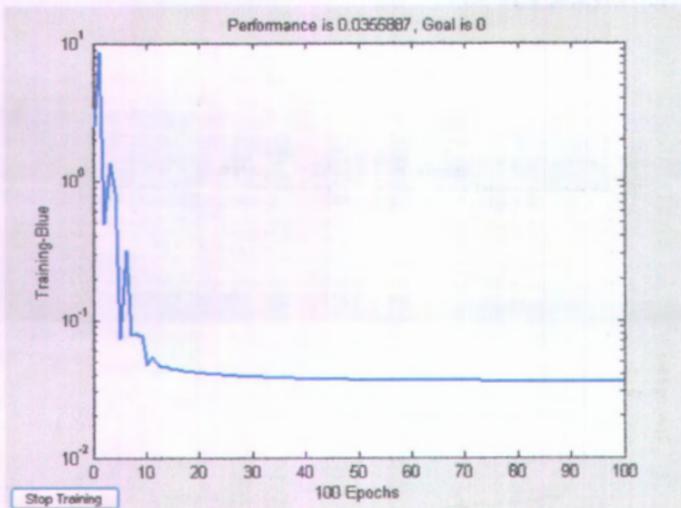
TRAININGM, Epoch 100/100, MSE 0.0278678/0, Gradient 0.00844359/1e-010

TRAININGM, Maximum epoch reached, performance goal was not met.

```
>> mse=mserror(x,s,'1')
```

```
mse =
```

```
20.3595
```



```
>> x=nn(p,t,a,s,7,250,0.3,0.5);
```

TRAININGM, Epoch 0/250, MSE 2.23336/0, Gradient 5.66679/1e-010

TRAININGM, Epoch 25/250, MSE 0.0370646/0, Gradient 0.0304877/1e-010

TRAININGM, Epoch 50/250, MSE 0.0325765/0, Gradient 0.0189622/1e-010

TRAININGM, Epoch 75/250, MSE 0.0305747/0, Gradient 0.0139212/1e-010

TRAININGM, Epoch 100/250, MSE 0.0293587/0, Gradient 0.0116882/1e-010

```

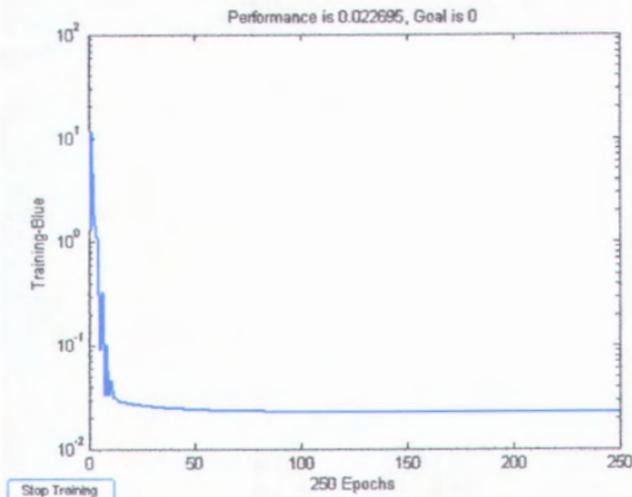
TRAINGDM, Epoch 125/250, MSE 0.0284208/0, Gradient
0.0107413/1e-010
TRAINGDM, Epoch 150/250, MSE 0.0275949/0, Gradient
0.0102473/1e-010
TRAINGDM, Epoch 175/250, MSE 0.0268384/0, Gradient
0.00980242/1e-010
TRAINGDM, Epoch 200/250, MSE 0.026153/0, Gradient
0.00926296/1e-010
TRAINGDM, Epoch 225/250, MSE 0.0255496/0, Gradient
0.00861993/1e-010
TRAINGDM, Epoch 250/250, MSE 0.0250336/0, Gradient
0.00791677/1e-010
TRAINGDM, Maximum epoch reached, performance goal was not met.

```

```
>> mse=mserror(x,s,'l')
```

```
mse =
```

```
19.3682
```



```
>> x=nn(p,t,a,s,10,250,0.9,0.7);
```

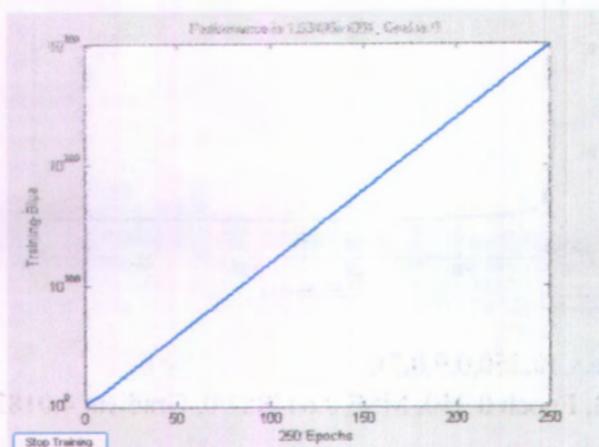
```
TRAINGDM, Epoch 0/250, MSE 0.644887/0, Gradient 4.01825/1e-010
```

TRAININGM, Epoch 25/250, MSE: 3.73995e+028/0, Gradient: 1.27936e+015/1e-010  
 TRAININGM, Epoch 50/250, MSE: 7.29835e+058/0, Gradient: 1.78719e+030/1e-010  
 TRAININGM, Epoch 75/250, MSE: 1.42424e+089/0, Gradient: 2.49661e+045/1e-010  
 TRAININGM, Epoch 100/250, MSE: 2.77934e+119/0, Gradient: 3.48763e+060/1e-010  
 TRAININGM, Epoch 125/250, MSE: 5.42376e+149/0, Gradient: 4.87202e+075/1e-010  
 TRAININGM, Epoch 150/250, MSE: 1.05842e+180/0, Gradient: 6.80595e+090/1e-010  
 TRAININGM, Epoch 175/250, MSE: 2.06546e+210/0, Gradient: 9.50753e+105/1e-010  
 TRAININGM, Epoch 200/250, MSE: 4.03066e+240/0, Gradient: 1.32815e+121/1e-010  
 TRAININGM, Epoch 225/250, MSE: 7.86565e+270/0, Gradient: 1.85535e+136/1e-010  
 TRAININGM, Epoch 250/250, MSE: 1.53495e+301/0, Gradient: 2.59183e+151/1e-010  
 TRAININGM, Maximum epoch reached, performance goal was not met.

```
>> mse = merror(s,a,'1')
```

```
mse =
```

```
4.7409e+303
```



Lampiran 8 : Contoh Rincian Hasil Perhitungan MSE

Epoch	mc	$\eta$	Si	Replikasi					Rata-rata
				1	2	3	4	5	
100	0.5	0.1	3	30.01	24.28	25.51	31.35	28.86	28.00
100	0.5	0.1	5	31.28	30.42	24.72	63.26	44.82	38.90
100	0.5	0.1	7	64.19	28.48	41.19	28.62	39.02	40.30
100	0.5	0.1	10	111.41	51.31	61.96	39.11	75.15	67.79
100	0.5	0.1	15	170.25	86.01	50.65	54.88	110.91	94.54
100	0.5	0.1	20	253.41	33.88	33.40	178.61	52.82	110.42
100	0.5	0.2	3	22.14	31.91	25.97	25.64	32.59	27.65
100	0.5	0.2	5	35.64	23.89	98.86	23.81	25.00	41.44
100	0.5	0.2	7	75.10	52.01	22.07	30.16	40.64	44.00
100	0.5	0.2	10	48.35	47.92	35.93	46.85	83.40	52.49
100	0.5	0.2	15	42.85	48.93	84.69	76.25	40.67	58.68
100	0.5	0.2	20	Inf	Inf	Inf	Inf	Inf	Inf
100	0.5	0.3	3	21.53	25.36	23.42	24.33	31.84	25.30
100	0.5	0.3	5	24.32	33.60	24.79	31.28	67.85	36.37



## BIODATA PENULIS



**Rita Ummi Hanik**, dilahirkan di Jombang, 3 November 1988, merupakan anak kedua dari tiga bersaudara. Riwayat pendidikan formal yang telah ditempuh penulis antara lain adalah : RA Dewi Sartika Megaluh, MI Mambaul Ulum Megaluh, MTsN Megaluh, SMAN 2 Jombang. Setelah lulus dari pendidikan SMA pada tahun 2006, Penulis mengikuti seleksi PMDK Ber-beasiswa dan diterima di Jurusan Teknik Industri, FTI-ITS, Surabaya, dengan NRP 2506.100.131.

Selama menjalani masa perkuliahan di Jurusan Teknik Industri, Penulis aktif dalam beberapa organisasi dan pelatihan. Riwayat organisasi yang pernah diikuti adalah : Staff Departemen PSDM HMTI-ITS (2007/2008), Staff Divisi Pendidikan IECC BEM ITS, Sekretaris II Himpunan Mahasiswa Teknik Industri-ITS (2008/2009). Sedangkan pelatihan yang pernah diikuti adalah LKMM Pra-TD (2006), LKMM TD (2007), dan P3MTI (2008). Selain itu, Penulis juga menjadi Administrator Laboratorium Komputasi dan Optimasi Industri (Lab KOI) TI-ITS, dengan pengalaman sebagai *Staff Maintenance Department Maintenance* (2009-2010).

Pengalaman aplikasi ilmu pernah penulis dapatkan ketika Kerja Praktek di PT. Alstom Power Energy Systems Indonesia. Selama menyelesaikan Tugas Akhir ini, penulis mengikuti Program Magang Industri oleh Jurusan Teknik Industri ITS pada PT. Garuda Maintenance Facility Aero Asia (PT. GMF Aero Asia). Penulis dapat dihubungi melalui email [rita.ummi.hanik@gmail.com](mailto:rita.ummi.hanik@gmail.com).