

OPTIMASI PREVENTIVE MAINTENANCE DENGAN PSO (PARTICLE SWARM OPTIMIZATION) PADA SEMI LEAN SOLUTION PUMP 107-JC DI PABRIK I PT. PETROKIMIA GRESIK

Nama Mahasiswa : Widdhi Purwo Pudyastuti

NRP : 2410100040

Jurusan : Teknik Fisika FTI - ITS

Pembimbing I : Ir. Ya'umar, M.T.

Pembimbing II : Dr. Bambang Lelono W., S.T.,M.T.

Abstrak

Dalam memproduksi amonia, terdapat tahapan *CO₂ Removal* dimana pada proses ini terdapat 3 buah pompa untuk mengalirkan larutan *benfield* menuju *CO₂ Absorber*. Pada salah satu *Semi Lean Solution Pump*, yaitu 107-JC sering mengalami *breakdown* karena kebocoran pada *bearing*, *mechanical seal*, korosi, vibrasi tinggi, keausan, dan permasalahan proses. Dalam penelitian ini dilakukan metode optimasi menggunakan PSO (*Particle Swarm Optimization*) untuk menentukan penjadwalan *preventive maintenance* sehingga dapat meningkatkan nilai *reliability* pada sistem pompa 107-JC dan meminimalisir biaya yang dibutuhkan untuk melakukan *maintenance*. Pemilihan komponen pompa berdasarkan *record* data *maintenance* selama 4 tahun (2010-2013) dengan intensitas kerusakan lebih dari 5 kali. Dalam proses optimasi ini, dicari nilai waktu yang tepat untuk penjadwalan *preventive maintenance*. Dari hasil optimasi, didapat penjadwalan PM adalah 110 jam. Perlakuan *maintenance* tiap komponen pompa berbeda pada tiap interval. Untuk rentang waktu 110 jam, nilai *reliability* kritis ketika akan dilakukan *maintenance* adalah 0,645. Selama rentang waktu 1430 jam atau selama 2 bulan, perawatan pompa dengan *preventive maintenance* dapat menjaga nilai *reliability* pompa selalu pada rentang 0,6 dan 1. Serta biaya yang dibutuhkan selama 1430 jam atau 2 bulan adalah sebesar US\$2.115,95.

Kata Kunci : Preventive Maintenance, PSO (Particle Swarm Optimization), Reliability, Semi Lean Solution Pump

***OPTIMIZATION OF PREVENTIVE MAINTENANCE
WITH PSO (PARTICLE SWARM OPTIMIZATION) ON
SEMI LEAN SOLUTION PUMP 107-JC IN FACTORY I
PT. PETROKIMIA GRESIK***

Name of Student : Widdhi Purwo Pudyastuti
NRP : 2410100040
Department : Engineering Physics FTI - ITS
1st Supervisor : Ir. Ya'umar, M.T.
2nd Supervisor : Dr. Bambang Lelono W., S.T.,M.T.

Abstract

In producing ammonia, there are stages of CO₂ Removal in the process where there are 3 pumps to drain the solution of Benfield to the CO₂ Absorber. At one Semi Lean Solution Pump, ie 107-JC is often breakdown because a leak in the bearing, mechanical seal, corrosion, high vibration, wear, and process problems. In this research, the optimization method using PSO (Particle Swarm Optimization) to determine the scheduling of preventive maintenance that can increase the value of reliability to the system pumps 107-JC and minimize the costs required to perform maintenance. Selection of pump components is based on the data record maintenance for 4 years (2010-2013) with the intensity of damage more than 5 times. In this optimization process, look for the value of the right time for scheduling preventive maintenance. From the optimization results, obtained scheduling PM is 110 hours. Maintenance treatment of each component pumps is different at each interval. For a span of 110 hours, the value of critical reliability when will be maintenance is 0.645. Over the span of 1430 hours or as long as 2 months, treatment pump with preventive maintenance can keep the pump reliability is always in the range of 0.6 and 1. As well as the costs required for 1430 hours or 2 months is US \$ 2,115.95.

Keywords : Preventive Maintenance, PSO (Particle Swarm Optimization), Reliability, Semi Lean Solution Pump

DAFTAR TABEL

Tabel 2.1. Ilustrasi data kerusakan	10
Tabel 3.1. Jumlah kerusakan komponen	21
Tabel 4.1. Aktivitas PM tiap komponen	31
Tabel 4.2. Biaya <i>maintenance</i> tiap <i>preventive maintenance</i>	33

BAB II

TINJAUAN PUSTAKA

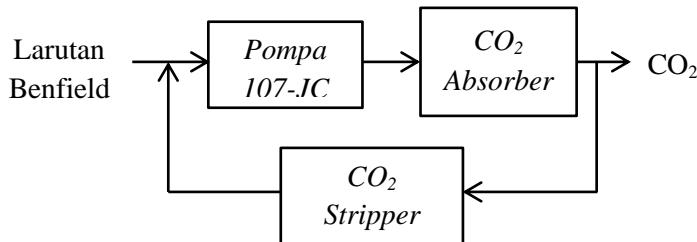
2.1 Pompa Sentrifugal

Pompa memiliki fungsi utama yaitu memindahkan cairan dari satu tempat ke tempat lainnya. Selain itu, pompa juga berfungsi untuk mensirkulasi cairan sekitar sistem, misalnya air pendingin atau pelumas yang melewati mesin dan peralatan. Dalam tugas akhir ini, jenis pompa yang dipakai adalah pompa dinamik tipe sentrifugal. Pompa sentrifugal dapat beroperasi dengan cairan didorong menuju sebuah *impeller* oleh tekanan atmosfir, atau *jet pump* oleh tekanan buatan. Kemudian baling-baling *impeller* meneruskan energi kinetik ke cairan yang menyebabkan cairan juga ikut berputar. Cairan meninggalkan *impeller* pada kecepatan tinggi. *Impeller* yang dikelilingi oleh *volute casting* atau dalam pompa turbin menggunakan cincin *diffuser* stasioner. *Volute* atau cincin *diffuser* stasioner ini berfungsi untuk mengubah energi kinetik menjadi energi tekanan. (Peralatan Energi Listrik: Pompa dan Sistim Pemompaan, 2006)



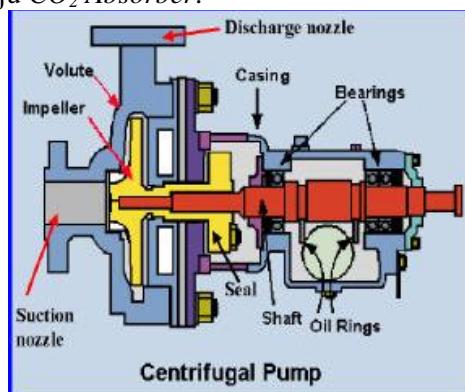
Gambar 2.1. Pompa 107-JC

Pada tugas akhir ini digunakan pompa sentrifugal *Semi Lean Solution Pump* 107-JC yang berfungsi untuk memindahkan larutan benfield pada tahap pemurnian gas sintesa proses pembentukan ammonia. Proses pemurnian gas sintesa terdiri dari 2 tahap, yaitu CO_2 Removal dan methanasi. Oleh karena gas CO_2 yang tidak diperlukan untuk proses pembentukan ammonia, maka penghilangan gas CO_2 dilakukan dengan cara absorpsi gas CO_2 oleh media K_2CO_3 (larutan benfield). Diagram fungsional untuk proses penghilangan gas CO_2 ditunjukkan pada Gambar 2.2.



Gambar 2.2. Diagram fungsional pompa 10-JC

Pompa 107-JC yang digerakkan oleh motor, sering terjadi *trouble* pada bagian mekanik. Sehingga, pompa 107-JC harus di *switch* ke pompa lainnya. Pada saat *switch* ini memerlukan waktu cukup lama yang berakibat pada proses yaitu berkurangnya aliran (*flow*) menuju *CO₂ Absorber*.

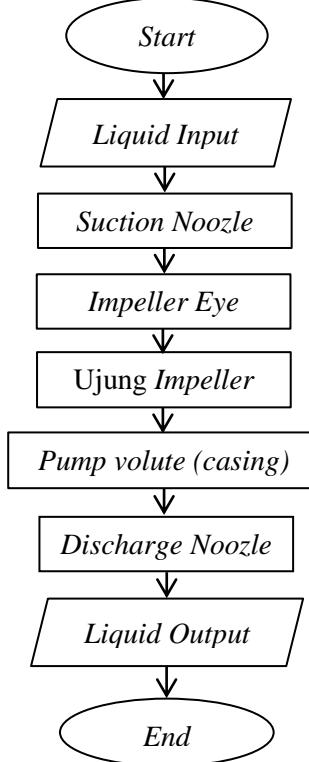


Gambar 2.3. Komponen utama pompa sentrifugal (Sahdev, M)

Komponen dari pompa sentrifugal adalah sebagai berikut :

- Komponen berputar : *impeller* yang disambungkan ke sebuah poros
- Komponen statis : *casing*, penutup *casing*, dan *bearings*. (Sahdev, M)

Proses masuknya cairan ke pompa yaitu, cairan dihisap oleh *suction noozle* kemudian masuk ke poros (*center*) dari bagian pompa sebagai *impeller*. Proses masuknya cairan sampai keluar dari pompa, dapat digambarkan pada diagram berikut :



Gambar 2.4. Diagram blok aliran fluida dalam pompa

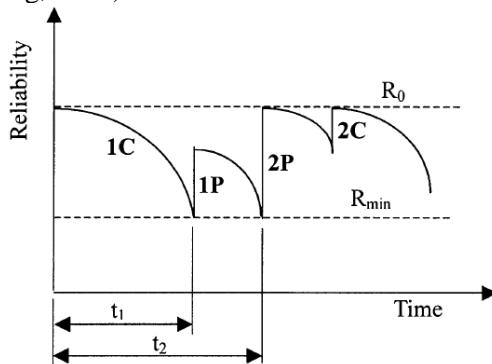
Ketika *impeller* berotasi, menyebabkan cairan teraduk diantara *vanes* kemudian cairan keluar dan menjadi gaya sentrifugal. *Vanes* yang berputar memberikan kecepatan sentrifugal. Energi kinetik yang keluar dari *impeller* menimbulkan sebuah resistansi aliran. Resistansi pertama dihasilkan oleh *pump volute (casing)* yang mengurangi kecepatan dan mengubahnya menjadi tekanan sesuai prinsip Bernoulli. Pada *head* (tekanan pada ketinggian cairan) mengalir kurang lebih sama dengan kecepatan energi pada pinggiran *impeller*. Pada *shaft*, terdapat *seal* dan *bearing*. *Seal* berfungsi untuk mencegah kebocoran cairan pada poros pompa. Pada *bearing housing*, berfungsi membungkus *bearing* yang terpasang pada *shaft*. *Bearing* menjaga poros rotor agar sama dengan bagian stasioner dibawah beban radial dan transversal. Pada *bearing housing* juga terdapat wadah minyak untuk pelumasan, permukaan *level* minyak yang konstant, jaket untuk air pendingin.

2.2 Preventive Maintenance

Maintenance atau perawatan adalah kegiatan yang bertujuan untuk menjamin fungsi komponen atau sistem agar dapat berjalan sesuai kondisi standar atau kondisi yang diharapkan. Perawatan adalah suatu konsepsi dari semua aktivitas yang diperlukan untuk menjaga atau mempertahankan kualitas agar tetap dapat berfungsi dengan baik seperti dalam kondisi sebelumnya. (Supandi, 990:5). Perawatan memiliki peranan yang sangat penting dalam tindakan pencegahan kerusakan (*preventive*) dan perbaikan kerusakan (*corrective*). Sesuai dengan tindakan tersebut, strategi pemeliharaan terdiri dari dua macam, yaitu pemeliharaan terencana (*Planned Maintenance*) dan pemeliharaan tidak terencana (*Unplanned Maintenance*). Pemeliharaan terencana terdiri dari dua macam, yaitu pemeliharaan secara rutin (*Preventive Maintenance*) yang telah terjadwal (*Scheduled Maintenance*) atau diprediksi sebelumnya (*Predictive Maintenance*). Pemeliharaan yang tidak terencana adalah pemeliharaan darurat (*Emergency Maintenance*) yang terjadi

diluar perkiraan dan harus segera diperbaiki kembali (*Corrective Maintenance*).

Pemeliharaan secara rutin (*Routine Maintenance*) merupakan tindakan perawatan untuk mencegah kerusakan atau mengurangi penurunan nilai *reliability* komponen. Kegiatan ini dilakukan secara berkala (*Time Based*) dan peramalan dengan pengukuran (*Condition Based*). Pemeliharaan secara rutin ini dapat dilakukan dengan pengecekan mesin, pengecekan vibrasi, penggantian oli, penggantian suku cadang, dan pembersihan. Menurut Lie and Chun (1986), aktivitas CM (*Corrective Maintenance*) dikategorikan ke dalam *minimal repair* (1C) dan *corrective replacement* (2C). Aktivitas *preventive maintenance* terdiri dari *simple preventive maintenance* (1P) dan *preventive replacement* (2P). Gambar 2.1 merepresentasikan karakter *reliability* sistem dengan berbagai jenis *maintenance*. (Tsai, Wang, & Teng, 2001)



Gambar 2.5. Perubahan *reliability* sistem dengan berbagai tipe *maintenance* (Tsai, Wang, & Teng, 2001)

Aktivitas *simple repair* (1C-maintenance) tidak merubah waktu dan mengembalikan *reliability* sistem ketika terjadi kegagalan. *Corrective replacement* (2C-maintenance) mengembalikan waktu sistem kembali nol dan memperbarui kurva *reliability* sistem. *Simple preventive maintenance* (1P-maintenance) merubah *reliability* sistem hanya untuk beberapa waktu. Sedangkan 2P-maintenance seperti *corrective replacement*

mengembalikan kurva *reliability* menjadi baru lagi. (Tsai, Wang, & Teng, 2001)

2.3 Pemodelan *Reliability*

Reliability atau keandalan merupakan keadaan suatu komponen atau sistem akan bekerja dengan baik pada kurun waktu tertentu dan pada kondisi tertentu. Konsep keandalan adalah untuk meningkatkan kinerja komponen dan memberikan *redundant*. Terdapat dua metode untuk menganalisa *reliability*, yaitu *time failure distribution* dan *empirical method*. Umumnya, digunakan *time failure distribution* untuk menghitung *reliability* dengan jumlah data banyak. Namun, karena jumlah data terbatas, digunakan metode empiris untuk menentukan *reliability*. (Ebeling, 1997)

2.3.1 Metode Empiris

Analisa metode empiris (*empirical method*) disebut juga *nonparametric methods* atau *distribution-free method*. Tujuannya adalah untuk memperoleh secara langsung *failure times*, *the failure distribution*, *reliability function*, dan *hazard rate function*. (Ebeling, 1997)

Tabel 2.1. Ilustrasi Data Kerusakan (Asrori, Widjiantoro, & Cordova, 2014)

i	TTF	Reliability	Failure Rate
1	t_1	R_1	f_1
2	t_2	R_2	f_2
...
n	t_n	R_n	f_n

Apabila diperoleh data kerusakan berupa TTF (*Time to Failure*) dari suatu komponen, maka *reliability* dan *failure rate* setiap waktu kerusakan dapat diperoleh dari Persamaan (2.1) dan Persamaan (2.2). (Asrori, Widjiantoro, & Cordova, 2014)

$$R(t_i) = 1 - \frac{i}{n+1} = \frac{n+1-i}{n+1} \quad (2.1)$$

$$\lambda(t) = \frac{1}{(t_{i+1}-t_i)(n+1-i)} \text{ untuk } t_i < t < t_{i+1} \quad (2.2)$$

Keterangan : $R(t_i)$: reliability pada TTF ke-i

n : jumlah kerusakan

$\lambda(t)$: failure rate pada TTF ke-i

t_i : nilai TTF ke-i

2.3.2 Dynamic Reliability

Umumnya, hazard rate function dari beberapa komponen dapat didefinisikan sebagai fungsi reliability seperti pada Persamaan (2.3).

$$h(t) = -\frac{1}{R(t)} \frac{dR(t)}{dt} \quad (2.3)$$

Dua tipe reliability tergantung failure rate models yang tepat dalam mendeskripsikan masalah kegagalan kumulatif, seperti *fatigue*, *wear*, *corrosion*, *thermal creep*, dan lain sebagainya. Sehingga, hazard function (h) dapat didefinisikan pada Persamaan (2.4). (Tsai, Wang, & Teng, 2001)

$$h(t) = A_0 + A_1(R_0 - R(t)) \quad (2.4)$$

Dimana R_0 adalah initial reliability, A_0 adalah failure rate, dan A_1 adalah degradation factor. Dari Persamaan (2.3) dan Persamaan (2.4), dynamic reliability suatu sistem tanpa preventive maintenance dapat direpresentasikan dalam Persamaan (2.5). (Asrori, Widjiantoro, & Cordova, 2014)

$$R(t) = \frac{R_0(A_0 + A_1 R_0)}{A_1 R_0 + A_0 e^{[(A_0 + A_1 R_0)t]}} \quad (2.5)$$

Apabila dalam suatu sistem dilakukan preventive maintenance, maka Persamaan (2.5) diubah dalam Persamaan (2.6).

$$R(t - nT) = \frac{R_0(A_0 + A_1 R_0)}{A_1 R_0 + A_0 e^{[(A_0 + A_1 R_0)(t - nT)]}} \quad (2.6)$$

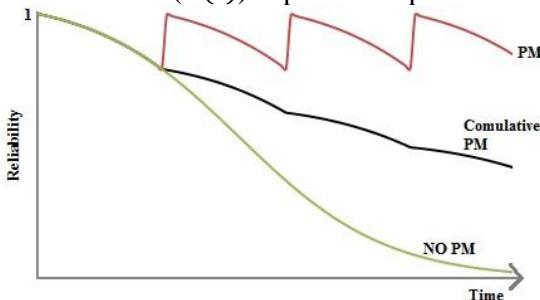
Ketika suatu sistem terus dilakukan preventive maintenance, aktualnya sistem tersebut terus mengalami penurunan reliability.

Sehingga, pemodelan sistem harus dinyatakan dalam *Cummulative Preventive Maintenance* seperti Persamaan (2.7) dan Persamaan (2.8). (Ebeling, 1997)

$$R_m(t) = R(T)^n R(t - nT) \quad (2.7)$$

$$R_m(t) = \left(\frac{R_0(A_0 + A_1 R_0)}{A_1 R_0 + A_0 e^{[(A_0 + A_1 R_0)T]}} \right)^n \left(\frac{R_0(A_0 + A_1 R_0)}{A_1 R_0 + A_0 e^{[(A_0 + A_1 R_0)(t-nT)]}} \right) \quad (2.8)$$

Grafik yang membedakan kurva *preventive maintenance* ($R(t - nT)$), *Cummulative preventive maintenance* ($R_m(t)$), dan *no preventive maintenance* ($R(t)$) dapat dilihat pada Gambar 2.4.



Gambar 2.6. Kurva *reliability* penurunan *failure rate* (Asrori, Widjiantoro, & Cordova, 2014)

2.4 Biaya Total *Maintenance*

Untuk menekan biaya total *maintenance*, maka diperlukan penjadwalan *preventive maintenance* yang tepat dan sesuai kebutuhan. Karena biaya maintenance akan sangat banyak ketika terjadi *breakdown* secara tiba-tiba yang mengakibatkan penggantian komponen. Meskipun dalam kondisi aktual sudah terdapat *redundant* yang dapat menggantikan komponen yang rusak, proses *switch* dari satu komponen ke komponen lainnya memerlukan waktu yang lama dan dapat berpengaruh terhadap proses produksi. Apabila suatu komponen tidak terdapat *redundant*, dapat mengakibatkan *downtime* dan lebih banyak mengeluarkan biaya total *maintenance*.

Biaya total *maintenance* dapat dianalisa melalui tiga faktor, yaitu *downtime*, *man power*, dan biaya komponen pengganti jika harus diganti. (Chen, Yang, & Weng, 2004) Sehingga, biaya total *maintenance* dapat dirumuskan pada Persamaan (2.9).

$$C_{TM} = C_c + C_d + (TTR \times C_m) \quad (2.9)$$

Dengan C_{TM} : Cost total Maintenance

C_c : Cost komponen pengganti

C_d : Cost downtime

C_m : Cost tenaga pelaksana (*man power*)

TTR : time to repair (hours)

2.5 Optimasi PM (*Preventive Maintenance*) dengan Metode PSO (*Particle Swarm Optimization*)

Optimasi merupakan aktivitas untuk mendapatkan hasil yang terbaik atau optimal (nilai efektif yang dapat dicapai) dari pilihan yang tersedia. Tujuan dari setiap keputusan adalah untuk meminimumkan usaha yang dilakukan atau memaksimumkan keuntungan yang diperoleh. Usaha atau keuntungan tersebut secara praktek dinyatakan sebagai fungsi dengan variabel keputusan yang akan dicari nilai optimumnya. Teknik optimasi secara umum dapat dibagi menjadi dua bagian, yang pertama adalah *mathematical programming* dan yang kedua adalah *combinatorial optimization*. (Zerda, 2009)

Tujuan *preventive maintenance* adalah untuk meng-upgrade kembali nilai *reliability* sistem yang menurun. Namun, apabila kegiatan penjadwalan *maintenance* yang dilakukan kurang tepat, maka biaya perawatan yang dikeluarkan perusahaan dapat lebih besar. Oleh karena itu, Kamran (2008) menurunkan beberapa *obyektif function* yang dapat digunakan untuk meningkatkan nilai *reliability* namun meminimalisasi nilai *cost*. (Moghaddam, 2010)

$$Fitness_1 = w_1 \left(\frac{cost_{total}}{cost_{max}} \right) + w_2 (-Reliability) \quad (2.10)$$

$$Fitness_2 = (-Reliability) + \left(\frac{1}{cost_{max}} \right) \times |Total Cost - Given Budget| \quad (2.11)$$

$$Fitness_3 = \left(\frac{\text{Total Cost}}{\text{Cost}_{\max}} \right) \times |\text{Reliability} - \text{Required reliability}| \quad (2.12)$$

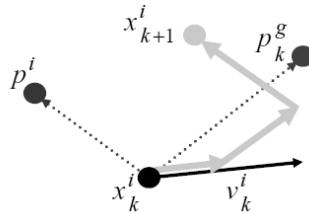
Objective function pada Persamaan (2.10) merupakan tipe penyetaraan orde *total cost maintenance* dengan *reliability*. Sedangkan pada Persamaan (2.11) digunakan ketika *budget* atau biaya *maintenance* telah ditentukan sebelumnya. Sementara Persamaan (2.12) merupakan model yang digunakan jika diberlakukan target *reliability* yang harus dicapai. (Asrori, Widjiantoro, & Cordova, 2014)

Algoritma PSO (*Particle Swarm Optimization*) merupakan algoritma optimasi yang meniru perilaku suatu kawanan binatang dalam melakukan pencarian tempat hidup yang aman dan memiliki suplai makanan yang banyak. Algoritma ini diperkenalkan pertama kali oleh James Kennedy and Russel Eberhart pada tahun 1995. (Wati, 2011) Algoritma *particle swarm optimization* adalah metode pencarian yang didasarkan populasi dan merupakan algoritma optimasi global yang dihubungkan dengan masalah yang mana solusi terbaik dapat direpresentasikan sebagai titik atau *surface* diarea *n-dimensional*. (Zerda, 2009)

2.5.1 Proses Algoritma PSO (*Particle Swarm Optimization*)

Pada algoritma PSO ini, suatu individu disebut partikel. Setiap partikel memiliki posisi $X_i = (x_1, \dots, x_n)$ dan tersebar secara acak pada ruang solusi berdimensi n (n=banyaknya variabel). Posisi partikel i pada saat waktu t, yaitu $x_i(t)$ akan diperbaiki menurut Persamaan (2.13).

$$x_i(t + 1) = x_i(t) + v_i(t + 1) \quad (2.13)$$



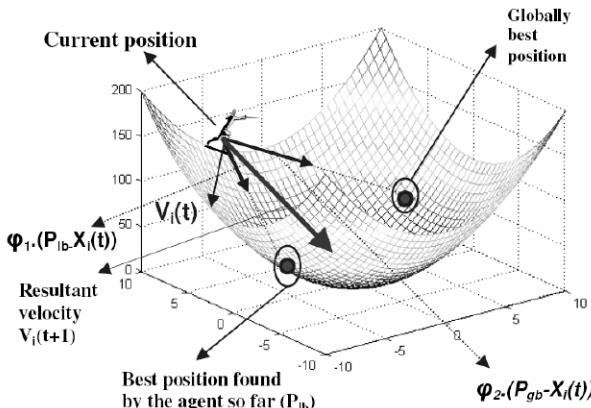
Gambar 2.7. Ilustrasi perubahan posisi (Zerda, 2009)

Setiap iterasi partikel tersebut, akan memperbarui posisinya dengan kecepatan $V_j = (v_1, \dots, v_n)$. Kecepatan partikel $v_j(i)$ dapat dihitung dengan menggunakan Persamaan (2.14).

$$\begin{aligned} v_j(i) &= w v_j(i-1) + c_1 r_1 (P_{best,j} - x_j(i-1)) + \\ &c_2 r_2 (Gbest - x_j(i-1)) \end{aligned} \quad (2.14)$$

Keterangan :

- $p_{best,1}(t)$ adalah posisi partikel terbaik yang dicapai oleh partikel i sampai waktu t dan merepresentasikan kontribusi individu terhadap vektor $v_i(t+1)$
- $g_{best}(t)$ adalah solusi global terbaik yang dicapai dari semua partikel sampai waktu t dan merepresentasikan kontribusi sosial terhadap vektor kecepatan
- r_1 dan r_2 adalah bilangan acak terdistribusi seragam pada interval $[0, 1]$
- c adalah faktor skala kognitif (individu) umumnya bernilai 2
- c_2 adalah faktor skala sosial umumnya bernilai 2
- w adalah bobot inersia



Gambar 2.8. Ilustrasi perubahan kecepatan (Zerda, 2009)

Nilai bobot inersia w memberikan hasil yang baik apabila memiliki nilai dari tinggi ke rendah. Penentuan nilai w mulai dari 0,9 dan menurun secara linier sampai 0,4 dengan *decrement factor* a (Uysal dan Bulkan, 2008). Secara matematis, nilai w pada iterasi (i) dirumuskan pada Persamaan (2.15).

$$\theta(i) = \theta_{max} - \left(\frac{\theta_{max} - \theta_{min}}{i_{max}} \right) i \quad (2.15)$$

Untuk metode optimasi dengan fungsi obyektif f dan variabel x_1, \dots, x_n dapat dituliskan pada Persamaan (2.16).

$$\min_{x_1, \dots, x_n} f(x_1, \dots, x_n) \quad (2.16)$$

$$\text{dengan } x_1, \dots, x_n = [x_{lower}, x_{upper}]$$

Dengan ukuran populasi: n_{pop} dan ukuran partikel: n , prosedur PSO dapat dilakukan sebagai berikut :

- Inisiasi ukuran kelompok (jumlah partikel) adalah n . Sebaiknya nilai n tidak terlalu besar dan tidak terlalu kecil agar lebih banyak kemungkinan menemukan posisi partikel yang baik. Biasanya ukuran n antara 20 sampai 30 partikel.
- Populasi ini dibangkitkan secara acak pada ruang solusi $[0,1]$. Evaluasi nilai fungsi tujuan untuk setiap partikel dan nyatakan dengan Persamaan (2.17).

$$f[X_1(0)], f[X_2(0)], \dots, f[X_n(0)] \quad (2.17)$$

- Menghitung kecepatan dari semua partikel secara acak pada ruang $[0,1]$.
- Perhitungan nilai fitness partikel, seperti pada Persamaan (2.14).
- Menentukan p_{best} (terbaik lokal), yaitu hasil evaluasi fungsi $f(x_1, \dots, x_n)$ didapat nilai $cost$ sebanyak n_{pop} , yaitu $f_1, \dots, f_{n_{pop}}$. Nilai p_{best} ditentukan apabila nilai $cost$ dari sebuah partikel lebih baik (lebih kecil) dari nilai $cost$ sebelumnya, maka p_{best} diset ke nilai $cost$ yang baru.

- Menentukan g_{best} (terbaik global) ditentukan apabila nilai minimum $cost$ dari semua partikel lebih baik (lebih kecil) dari nilai minimum $cost$ semua partikel pada iterasi sebelumnya, maka nilai g_{best} diset sebagai nilai minimum $cost$ yang baru.
- Mengecek apakah solusi yang sekarang sudah konvergen. Jika posisi semua partikel menuju ke satu nilai yang sama, maka kondisi ini disebut konvergen. Jika belum konvergen maka diulang dengan memperbarui iterasi $i = i + 1$, dengan cara menghitung nilai baru dari p_{best} dan g_{best} . Proses iterasi ini dilanjutkan sampai semua partikel menuju ke satu titik solusi yang sama. Apabila masih belum konvergen, dilanjutkan perhitungan kecepatan dan posisi baru untuk masing-masing partikel.

2.5.2 Parameter Algoritma PSO (*Particle Swarm Optimization*)

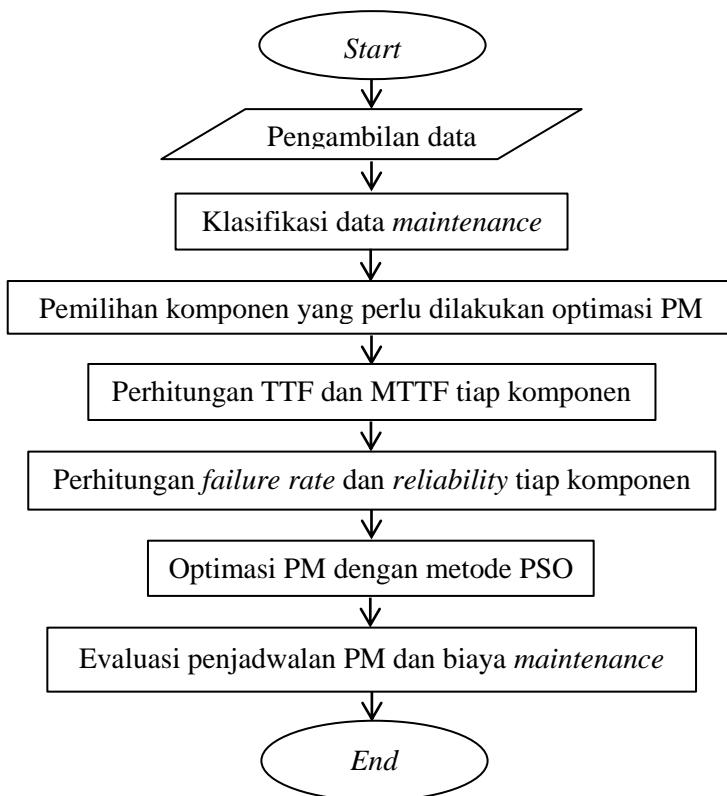
Beberapa parameter yang dibutuhkan dalam algoritma PSO (*Particle Swarm Optimization*) adalah sebagai berikut :

- Jumlah *particle*
Pada umumnya, jumlah partikel adalah 20 sampai 30 partikel. Biasanya, jumlah 10 partikel sudah cukup untuk mendapatkan nilai optimal. Namun, untuk masalah yang lebih rumit, baik untuk menggunakan 100 hingga 200 partikel.
- Dimensi partikel
Dimensi partikel atau banyaknya variabel merupakan permasalahan yang akan dioptimasi.
- Bobot inersia (w)
Di algoritma *particle swarm optimization* keseimbangan antara kemampuan eksplorasi *global* dan *local* secara utama di kontrol oleh *inertia weight* dan merupakan parameter penurunan kecepatan untuk menghindari *stagnasi particle* di lokal optimum.

- Faktor skala (c_1 dan c)
Faktor skala kognitif (c_1) faktor skala sosial (c_2) umumnya bernilai 2. Namun, biasanya nilai c_1 dan c_2 berada pada rentang [0, 4].
(<http://www.swarmintelligence.org/tutorials.php>, 2014)
- Konvergensi
Apabila posisi partikel menuju ke suatu titik yang sama atau proses iterasi menuju ke suatu nilai yang sama, maka proses optimasi sudah mencapai konvergen.

BAB III METODOLOGI

Adapun beberapa tahapan dalam penyelesaian tugas akhir ini yang direpresentasikan dalam diagram alir berikut ini :



Gambar 3.1 Diagram alir realisasi pelaksanaan tugas akhir

Adapun penjelasan tahapan dalam menyelesaikan tugas akhir berdasarkan diagram alir pada Gambar 3.1 adalah sebagai berikut :

- Pengambilan data
Data yang diperlukan dalam tugas akhir ini meliputi data *maintenance* selama 4 tahun mulai bulan Januari tahun 2010 sampai bulan Desember tahun 2013. Selain itu, juga diperlukan daftar harga tiap komponen, biaya *man power* per jam, jumlah tenaga *man power* untuk proses perbaikan atau penggantian tiap komponen, dan P&ID dari *Semi Lean Solution Pump* 107-JA di Pabrik I PT. Petrokimia Gresik.
- Klasifikasi data *maintenance*
Tahap ini merupakan tahapan pemilihan data *maintenance* yang mengklasifikasikan jumlah kerusakan berdasarkan jenis komponen.
- Pemilihan komponen yang perlu dilakukan optimasi *preventive maintenance*
Setelah data *maintenance* selama 4 tahun tersebut diklasifikasikan, kemudian dipilih komponen berdasarkan jumlah kerusakan lebih dari 4 kali. Sehingga didapat 6 buah komponen yang akan dioptimasi.
- Perhitungan TTF dan MTTF tiap komponen
Setelah didapat 6 komponen yang akan dioptimasi, dilakukan perhitungan TTR, TTF dan MTTF dari masing-masing komponen.
- Menghitung *failure rate* dan *reliability* tiap komponen
Setelah didapat nilai MTTF, dapat diketahui nilai *failure rate*, dan kemudian dapat dihitung nilai *reliability* tiap komponen. Untuk mendapatkan nilai *failure rate* dan nilai *reliability*, digunakan *empirical method* karena jumlah data yang terbatas. Pada tahapan ini juga dilakukan pengeplotan grafik *reliability* pada kondisi aktual. Sehingga nantinya dapat dibandingkan dengan hasil *reliability* setelah dilakukan optimasi *preventive maintenance*.
- Optimasi dengan metode PSO (*Particle Swarm Optimization*)
Setelah diketahui *reliability* aktual pada pompa, kemudian dilakukan optimasi penjadwalan *preventive maintenance*. Metode optimasi yang digunakan adalah PSO dengan tujuan

mendapatkan hasil penjadwalan PM yang sama pada masing-masing komponen. Langkah-langkah optimasi PSO akan dijelaskan pada sub bab selanjutnya.

- Evaluasi penjadwalan PM (*Preventive Maintenance*) dan biaya *maintenance*

Setelah didapat hasil optimasi dengan metode PSO, maka dilakukan perbandingan hasil PM semula dengan hasil optimasi PM. Untuk mengetahui perbedaannya, PM awal dan hasil optimasi PM diplot dalam satu grafik *reliability*. Selain itu, dapat dihitung selisih total biaya *maintenance* yang dikeluarkan sebelum dan setelah dilakukan optimasi PM.

3.1 Pemilihan Komponen Pompa

Pengambilan data *maintenance* pada *Semi Lean Solution Pump* 107-JA yaitu selama 4 tahun mulai awal bulan Januari 2010 hingga akhir bulan Desember 2013. Data tersebut kemudian dikelompokkan berdasar jumlah kerusakan masing-masing komponen.

Tabel 3.1 Jumlah kerusakan komponen

No	Jenis Komponen	Jumlah Kerusakan
1	<i>Mechanical Seal</i>	14
2	Rotor	9
3	<i>Throttle Bushing</i>	8
4	<i>Journal Bearing</i>	8
5	<i>O-Ring</i>	8
6	<i>Wearing Chasing</i>	6
7	<i>Thrust Bearing</i>	6
8	<i>Shaft Sleeve</i>	5
9	<i>Wearing Impeller</i>	2
10	<i>Gasket Line Flushing</i>	1

(Sumber : Bagian Mekanik I, Pabrik I PT. Petrokimia Gresik)

Dari Tabel 3.1 dapat diketahui terdapat 8 buah komponen dengan jumlah kerusakan lebih dari 4 kali. Sehingga, komponen

yang akan dioptimasi jadwal *preventive maintenance*-nya adalah *Mechanical Seal*, *Rotor*, *Throttle Bushing*, *Journal Bearing*, *O-Ring*, *Wearing Chasing*, *Thrust Bearing*, dan *Shaft Sleeve*.

3.2 Pemodelan Matematis

Pemodelan matematis bertujuan untuk mendapatkan fungsi obyektif yang merupakan fungsi tujuan dari metode optimasi. Setelah didapat fungsi obyektif, kemudian dilakukan evaluasi nilai *fitness* yang merupakan rumusan dari fungsi tujuan. Untuk mendapatkan *fitness function*, diperlukan *dynamic reliability model* dan *total cost maintenance model*. Sebelum menentukan *dynamic reliability model*, ditentukan nilai TTF dari masing-masing komponen yang terpilih untuk dilakukan optimasi. Setelah diketahui nilai TTF, kemudian dapat dicari nilai *reliability* dan laju kerusakan (*failure rate*) menggunakan metode empiris (*empirical method*). Menurut You-Tern Sai (2001), hasil perhitungan nilai *reliability* dengan *empirical method* dapat didekati dengan *dynamic reliability model*.

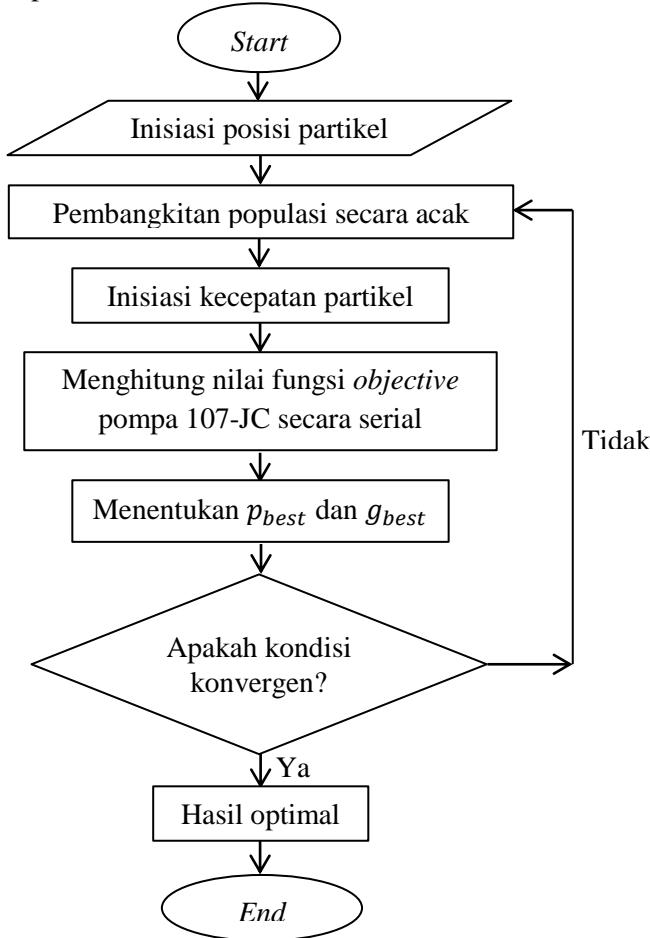
Dengan tujuan memaksimalkan nilai *reliability* sistem dan meminimalkan total biaya *maintenance*, maka menurut penelitian Ahmad Asrori (2014) pemodelan fungsi obyektif dapat dirumuskan hasil pengurangan *reliability* sistem terhadap fraksi *total cost maintenance* sistem.

$$f_{\text{objective}} = \prod R_c(t) - \frac{\sum C_{TM}}{C_{max}} \quad (3.1)$$

Dengan $R_c(t)$ adalah *reliability* tiap komponen, C_{TM} adalah *total cost maintenance* tiap komponen, dan C_{max} adalah maksimal *cost maintenance* dalam sistem. Dalam fungsi obyektif ini, digunakan nilai *reliability* secara sistem. Oleh karena itu, perhitungan *reliability* sistem merupakan hasil perkalian antara nilai *reliability* tiap komponen. Sedangkan *fraksi total cost maintenance* sistem adalah perbandingan jumlah *total cost maintenance* tiap komponen dengan maksimal *cost maintenance* sistem.

3.3 Penerapan Metode PSO (*Particle Swarm Optimization*)

Teknik optimasi penjadwalan PM (*Preventive Maintenance*) dengan PSO (*Particle Swarm Optimization*) dilakukan sesuai diagram alir pada Gambar 3.2.



Gambar 3.2 Diagram alir metode optimasi PSO

Langkah-langkah yang dilakukan untuk mendapatkan hasil optimasi ini adalah sebagai berikut :

- Inisiasi posisi partikel

Inisiasi posisi partikel ini ditentukan sebanyak 20 partikel (n) dan dibangkitkan secara acak. Pembangkitan populasi awal ini dilakukan dengan rentang antara nilai minimal dan maksimal dari populasi x yaitu berupa *time* untuk melakukan penjadwalan *preventive maintenance* secara random. Inisiasi partikel awal dinotasikan dalam $x_1(0), x_2(0), x_3(0), \dots, x_n(0)$.

- Inisiasi kecepatan partikel

Inisiasi kecepatan partikel dengan kecepatan awal 0 yang dinotasikan dalam $v_j(0)$, dengan $j = 1, 2, 3, \dots, n$. Setelah itu, dilakukan evaluasi nilai fungsi *objective* untuk setiap partikel.

- Menghitung nilai fungsi *objective* partikel

Untuk menghitung nilai fungsi *objective* ini harus diketahui variabel berupa komponen yang akan dioptimasi penjadwalan PM nya. Setelah didapat komponen yang akan dioptimasi, kemudian ditentukan jadwal PM selama 3 tahun. Evaluasi nilai fungsi *objective* tiap partikel berdasar posisinya didapat dari fungsi *objective*.

Setelah didapat nilai fungsi *objective* awal, kemudian dilakukan perhitungan kecepatan partikel pada iterasi awal ($i = 1$) yang dirumuskan pada persamaan (3.2).

$$v_j(i) = \theta v_j(i-1) + c_1 r_1 [P_{best,j} - x_j(i-1)] + c_2 r_2 [G_{best} - x_j(i-1)], \text{ dengan } j = 1, 2, 3, \dots, n \quad (3.2)$$

- Menentukan p_{best} dan g_{best}

Penentuan p_{best} (posisi awal terbaik untuk tiap partikel) dan g_{best} (posisi awal terbaik untuk keseluruhan partikel) didapat dari fungsi *objective* yang paling rendah (maksimum) dari keseluruhan partikel. Untuk setiap partikel ditentukan dengan membandingkan nilai p_{best} lama dengan p_{best} baru dengan iterasi maksimal 1000.

- Apabila hasil iterasi sudah mencapai konvergen atau setelah selisih posisi semua partikel menuju ke satu nilai yang sama, maka proses optimasi berhenti. Apabila belum mencapai konvergen, diulang lagi dengan mengevaluasi nilai fungsi

objective partikel dan membandingkan nilai p_{best} lama dan p_{best} baru dengan kecepatan kecepatan partikel yang berbeda.

3.4 Evaluasi *Preventive Maintenance* dengan PSO (*Particle Swarm Optimization*)

Dari data maintenance pompa selama 4 tahun (2010 – 2013) dapat dihitung TTF dan MTTF masing-masing komponen. Dari perhitungan TTF dan MTTF, kemudian diketahui nilai *reliability* aktual dan *failure rate* masing-masing komponen. Perhitungan *reliability* dan *failure rate* dihitung dengan metode empiris pada Persamaan (3.3) dan Persamaan (3.4).

$$R(t_i) = 1 - \frac{i}{n+1} = \frac{n+1-i}{n+1} \quad (3.3)$$

$$\lambda(t) = \frac{1}{(t_{i+1}-t_i)(n+1-i)} \text{ untuk } t_i < t < t_{i+1} \quad (3.4)$$

Kemudian plotting nilai *reliability* didapat dengan rumus pendekatan *dynamic reliability* pada Persamaan (3.5) untuk *reliability* tanpa *preventive maintenance*.

$$R(t) = \frac{R_0(A_0 + A_1 R_0)}{A_1 R_0 + A_0 e^{[(A_0 + A_1 R_0)t]}} \quad (3.5)$$

Nilai *reliability* sistem dihitung secara seri dari nilai *reliability* masing-masing komponen. Metode optimasi PSO (*Particle Swarm Optimization*) digunakan untuk mendapatkan penjadwalan *preventive maintenance* pompa. Dengan algoritma PSO, didapat penjadwalan yang optimal melalui iterasi program. Setelah didapat waktu yang paling optimal untuk penjadwalan *preventive maintenance* pompa secara sistem, dapat diketahui nilai *reliability* kritis dan biaya *maintenance*. Nilai *reliability* sistem setelah dilakukan *preventive maintenance* dihitung dengan *dynamic reliability* pada Persamaan (3.6).

$$R(t - nT) = \frac{R_0(A_0 + A_1 R_0)}{A_1 R_0 + A_0 e^{[(A_0 + A_1 R_0)(t-nT)]}} \quad (3.6)$$

Hasil perhitungan nilai *reliability* awal sebelum dilakukan *preventive maintenance* dengan nilai *reliability* setelah dilakukan optimasi penjadwalan *preventive maintenance* diplot pada Gambar 4.3. Dari grafik tersebut, dapat diketahui perbandingan nilai *reliability* kritis ketika akan dilakukan *preventive maintenance* dengan tanpa penjadwalan *preventive maintenance*.

BAB IV

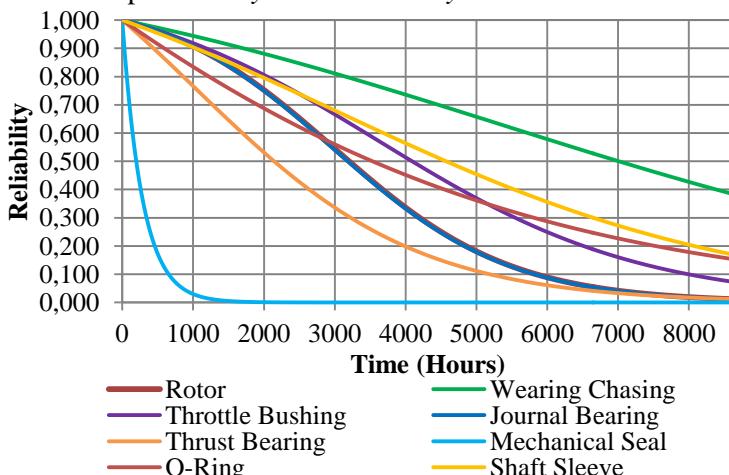
HASIL DAN PEMBAHASAN

4.1 Model *Dynamic Reliability*

Setelah dilakukan perhitungan nilai *reliability* dan *failure rate* dengan metode empiris, selanjutnya dilakukan perhitungan *dynamic reliability*. Parameter pada perhitungan *dynamic reliability* yang berupa A_1 didapat berdasarkan hasil pendekatan atau *fitting* nilai *reliability*. Nilai pendekatan yang diperoleh tersebut dibandingkan sehingga didapat nilai MSE terkecil antara nilai *reliability* hasil metode empiris dengan *dynamic reliability*. Perhitungan *dynamic reliability* tanpa *preventive maintenance* menggunakan persamaan berikut:

$$R(t) = \frac{R_0(A_0 + A_1 R_0)}{A_1 R_0 + A_0 e^{[(A_0 + A_1 R_0)t]}} \quad (4.1)$$

Dengan nilai R_0 merupakan nilai *reliability* ke-0, A_0 merupakan nilai *failure rate* ke-0, dan A_1 merupakan *degradation factor* yang diperoleh dari *fitting reliability*. Perolehan nilai A_1 dipilih berdasarkan nilai MSE terkecil dari selisih nilai *reliability* dengan metode empiris dan *dynamic reliability*.



Gambar 4.1. Grafik reliabilitas tiap komponen pompa

Semua komponen dalam pompa 107-JC ini tidak pernah dilakukan *preventive maintenance* secara rutin. Hal ini berpengaruh terhadapa nilai *reliability* untuk beberapa komponen yang mengalami penurunan *reliability* secara cepat pada rentang waktu 12000 jam. Beberapa komponen yang mengalami penurunan nilai *reliability* secara cepat diantaranya adalah rotor, *throttle bushing*, *thrust bearing*, *jornal bearing*, dan *mechanical seal*. Untuk komponen *mechanical seal* merupakan komponen yang paling cepat mengalami penurunan nilai *reliability*. *Mechanical seal* merupakan salah satu komponen dengan intensitas kerusakan paling tinggi selama 4 tahun. Pada **Gambar 4.1** merupakan grafik *reliability actual* tiap komponen pompa yang dihitung dengan *dynamic reliability*. Oleh sebab pada semua komponen pompa ini tidak dilakukan *preventive maintenance* secara berkala, maka dampaknya nilai reliability aktual selalu menurun seiring berjalananya waktu. Setiap maintenance yang dilakukan pada pompa ini, pasti terdapat beberapa komponen yang harus diganti khususnya yang paling sering diganti adalah *mechanical seal*. Dimana penggantian komponen ini memiliki harga tertinggi dibanding komponen lainnya. Setiap kegiatan maintenance yang dilakukan pada pompa, pasti menghentikan proses pada pompa tersebut dan dilakuakn pembongkaran komponen lainnya. Oleh karena itu, pada optimasi penjadwalan *preventive maintenance* dilakukan secara sistem agar mampu meningkatkan reliability tiap komponen sehingga meningkatkan *reliability* secara sistem. Selain itu, dapat menghemat biaya penggantian komponen khususnya komponen yang memiliki harga tinggi.

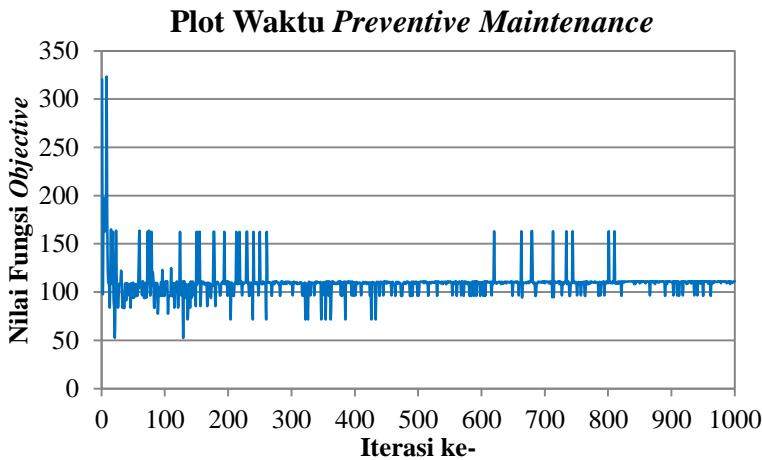
4.2 PSO (*Particle Swarm Optimization*)

Setelah diketahui penurunan nilai *reliability* untuk masing-masing komponen dengan model *dynamic reliability*, maka dapat dicari nilai *objective function* sebagai nilai *fitness* pada proses optimasi. Nilai *fitness* ini sebagai fungsi tujuan pada PSO (*Particle Swarm Optimization*). Adapun beberapa parameter dalam proses PSO adalah jumlah partikel (jumlah *swarm*), nilai

c1 dan c2, nilai r1 dan r2, maksimum iterasi, dan bobot inersia (*inertia weight*). Jumlah partikel yang digunakan dalam tahap optimasi ini adalah sebanyak 20 *swarm*. Jika digunakan jumlah *swarm* terlalu kecil, sedikit kemungkinan mendapatkan posisi terbaik. Begitu pula jika jumlah *swarm* terlalu besar, maka proses perhitungan semakin panjang dan sulit untuk menyampaikan informasi tiap *swarm*. Nilai parameter c1 dan c2 yang digunakan adalah 2, sedangkan r1 dan r2 dibangkitkan secara random untuk mendapatkan nilai *swarm* terbaik. Maksimum iterasi yang digunakan sebanyak 1000 kali. Bobot inersia memiliki nilai pada rentang 0,4 sampai 0,9 dan dihitung dengan Persamaan (2.15). Untuk inisiasi awal, nilai *swarm* dan *velocity* bernilai nol kemudian dibangkitkan secara random untuk masing-masing partikel. Kemudian dilakukan evaluasi nilai fungsi *objective* masing-masing partikel berdasar posisinya. Nilai terbaik fungsi *objective* ditetapkan sebagai Gbest. Nilai Gbest ini didapat dari kombinasi nilai posisi partikel terbaik atau Pbest. Nilai Gbest dipilih berdasarkan nilai fungsi *objective* maksimum karena fungsi *objective* merupakan fungsi maksimum untuk mendapatkan nilai optimal. Iterasi maksimum dilakukan sebanyak 1000 kali untuk mendapatkan nilai Gbest terbaik. Penentuan Pbest tiap partikel dengan membandingkan nilai Pbest awal (sebelum iterasi) dengan Pbest akhir. Jika tiap partikel sudah mencapai nilai yang sama pada tiap iterasi, maka proses iterasi akan berhenti dan didapat nilai Pbest dan Gbest untuk nilai yang paling optimal.

Untuk metode optimasi dengan PSO (*Particle Swarm Optimization*) ini, dapat menghasilkan nilai fungsi *objective* maksimum yang sama untuk hasil yang paling optimal. Namun untuk setiap kali running, waktu penjadwalan preventive maintenance yang dihasilkan tidak terlalu jauh. Hal ini dikarenakan proses optimasi dengan PSO (*Particle Swarm Optimization*) ini terdapat beberapa parameter yang dibangkitkan secara random. Begitu pula dengan parameter nilai r1 dan r2. Ketika semua parameter random telah diganti dengan nilai parameter terbaik selama pross running beberapa kali, maka

dilakukan evaluasi penjadwalan interval preventive maintenance. Evaluasi penjadwalan interval preventive maintenance dilakukan berkali-kali, sehingga didapat penjadwalan waktu preventive maintenance yang tepat sebagai hasil dari proses optimasi.



Gambar 4.2 Plot waktu preventive maintenance

Pada **Gambar 4.2** merupakan hasil running penjadwalan *preventive maintenance* pompa secara sistem menggunakan PSO (*Particle Swarm Optimization*). Hasil penjadwalan preventive maintenance untuk pompa untuk 1000 kali iterasi didapat hasil yang fluktuatif untuk iterasi awal sampai pada akhirnya didapat *time* yang paling optimal untuk dilakukan penjadwalan *preventive maintenance*. Hal ini ditunjukkan dengan hasil iterasi yang semakin lama semakin konvergen menuju di satu titik, yaitu, 111,091. Untuk mempermudah penjadwalan *preventive maintenance*, dilakukan selama rentang waktu 110 jam. Artinya, penjadwalan *preventive maintenance* yang tepat untuk pompa 107-JC adalah selama rentang waktu 110 jam. Hal ini dilakukan secara berkala untuk rentang waktu terebut untuk menjamin *reliability* pompa agar selalu terjaga secara maksimal untuk proses produksi.

Tabel 4.1 Aktivitas PM tiap komponen

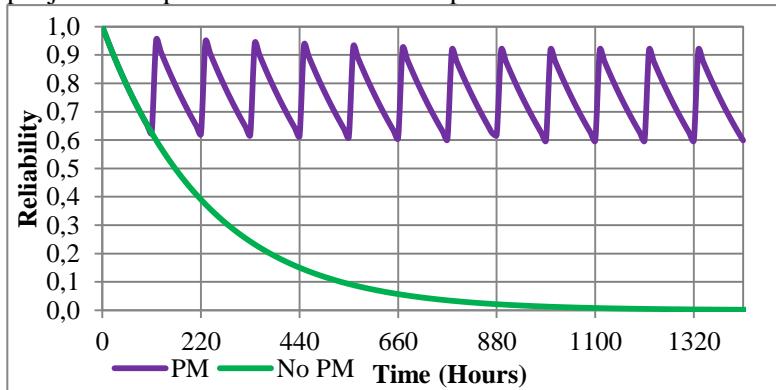
Waktu ke- (Jam)	Komponen							
	A	B	C	D	E	F	G	H
110	0	0	0	0	0	1	0	0
220	0	0	0	0	0	1	0	0
330	0	0	0	0	0	1	0	0
440	0	0	0	0	0	1	0	0
550	0	0	0	1	1	1	0	0
660	0	0	0	0	1	1	1	0
770	0	0	0	1	1	1	1	0
880	0	0	0	0	1	1	1	0
990	0	0	0	1	1	1	1	0
1100	1	0	0	1	1	1	1	1
1210	1	0	1	1	1	1	1	1
1320	1	0	1	1	2	1	1	1
1430	1	0	1	1	1	1	1	1

Untuk penjadwalan *preventive maintenance* selama rentang waktu 110 jam, perlakuan *maintenance* untuk tiap komponen pompa adalah berbeda. Perlakukan *maintenance* tiap komponen berdasarkan perhitungan reliability aktual yang kemudian dilakukan evaluasi sehingga didapat reliability sistem agar selalu terjaga diantara nilai 0,6 dan 1. Masing-masing komponen A sampai H adalah rotor, wearing chasing, throttle bushing, journal bearing, thrust bearing, mechanical seal, o-ring, dan shaft sleeve. Masing-masing komponen tersebut mendapat perlakuan berbeda pada masing-masing interval PM. Angka 0 artinya, tidak dilakukan *maintenance*, angka 1 artinya dilakukan *repair*, dan angka 2 artinya dilakukan penggantian komponen.

4.3 Penjadwalan *Preventive Maintenance*

Dari hasil optimasi, didapat rentang waktu terbaik untuk dilakukan preventive maintenance adalah 110 jam. Sebelum dilakukan optimasi, didapat grafik nilai reliability untuk masing-masing komponen pompa. Dari nilai reliability yang berbeda tiap komponen, dapat diketahui nilai reliability untuk satu sistem

pompa. Untuk mempertimbangkan nilai cost yang seminimal mungkin dan mendapatkan reliability sistem dengan semaksimal mungkin, maka didapat plotting hasil optimasi selama 110 jam penjadwalan preventive maintenance pada **Gambar 4.3** berikut.



Gambar 4.3 Grafik perbandingan reliability PM dan no PM

Dari grafik tersebut, nilai *reliability* sistem secara aktual di lapangan selalu menurun dikarenakan kondisi tanpa *preventive maintenance* secara berkala. Setelah dilakukan optimasi *preventive maintenance* selama 110 jam, didapat nilai *reliability* baru sebesar 0,947 yang sebelumnya memiliki nilai *reliability* terendah 0,653. Sebagai hasil plotting dari nilai *reliability* sistem secara aktual, ditunjukkan pada garis berwarna hijau pada **Gambar 4.3**. Dari *plotting* tersebut, terlihat bahwa nilai *reliability* aktual selalu mengalami penurunan sepanjang waktu dikarenakan tidak diberlakukannya sistem *preventive maintenance*. Sebagai perbaikan dari nilai *reliability* tersebut, dilakukan metode PSO (*Particle Swarm Optimization*) untuk memperbaiki sistem pompa agar selalu berjalan optimal. Selain itu, dapat meminimalkan biaya *maintenance* dengan hanya dilakukan *service* ringan daripada mengganti komponen pompa yang dapat mengeluarkan biaya lebih banyak. Dengan dilakukan *preventive maintenance* setiap 110 jam, sistem pompa ini dapat berjalan dengan baik untuk suatu proses produksi dengan menjaga nilai *reliability* selalu pada rentang antara 0,6 dan 1.

4.4 Total Biaya *Maintenance*

Sebagai tujuan dari optimasi *preventive maintenance* pada pompa ini adalah untuk mendapatkan nilai *reliability* yang maksimal dan nilai *cost* minimal. Sesuai hasil nilai fungsi *objective* untuk interval waktu *preventive maintenance* 110 jam, didapat biaya *maintenance* tiap interval waktu penjadwalan. Dari hasil *running* program PSO, dengan harga fraksi total *cost maintenance*, didapat perlakukan *maintenance* untuk masing-masing komponen. Perlakuan tersebut terdapat beberapa komponen yang dilakukan *repair*, tidak dilakukan *maintenance*, dan dilakukan pergantian komponen. Sesuai biaya *simple repair* dan harga penggantian komponen pada lampiran, sehingga didapat biaya total *maintenance* untuk rentang waktu 110 jam selama 1430 jam atau selama kurang lebih 2 bulan. Dari tiap interval ini, didapat total biaya selama 1430 jam atau 2 bulan.

Tabel 4.2 Biaya *maintenance* tiap *preventive maintenance*

Waktu ke- (Jam)	Biaya Perawatan (Rp)
110	600.000
220	600.000
330	600.000
440	600.000
550	1.020.000
660	790.020
770	1.030.020
880	790.020
990	1.030.020
1100	2.830.020
1210	4.030.020
1320	6.030.020
1430	4.030.020
Total	23.980.160

Dalam keadaan aktual, untuk sistem pompa ini tidak diberlakukan *preventive maintenance*. Sehingga, ketika terjadi

breakdown secara tiba-tiba, menyebabkan terhambatnya proses produksi. Pada proses CO₂ *removal* yang mengalirkan larutan *benfield* ini terdapat 2 pompa yang selalu *running* dan terdapat 1 pompa sebagai *redundant*. Ketika salah satu pompa mengalami *breakdown*, sehingga aliran larutan *benfield* ini harus di *switch* ke pompa lainnya yang sebagai *redundant*. Proses *switch* ini yang memerlukan waktu lama sekitar 1 jam, sehingga *flow* larutan *benfield* dapat berkurang, dan akibatnya menurun pula hasil proses produksi. Selain itu, *breakdown* secara tiba-tiba sebagai akibat dari tidak diberlakukannya sistem *preventive maintenance* dapat merusak komponen pompa. Akibatnya, sekali *overhaul* dapat dilakukan *repair* untuk semua komponen bahkan mengganti komponen dengan yang baru. Hal ini yang menyebabkan banyak mengeluarkan biaya *maintenance* terutama penggantian *mechanical seal* yang memiliki harga tertinggi dibanding komponen lainnya. Dari hasil optimasi selama rentang waktu 1430 jam, didapat biaya *maintenance* sebesar Rp 23.980.160,00 atau US\$2.115,95. Total biaya ini dapat menghemat biaya *maintenance* seperti yang dilakukan perusahaan selama ini. Aktivitas *maintenance* ketika pompa mengalami *overhaul* atau *breakdown*, sering dilakukannya penggantian komponen *mechanical seal* yang memiliki harga termahal untuk komponen pompa. Sehingga dapat dipastikan total biaya *maintenance* yang dibutuhkan untuk interval 110 jam jauh lebih hemat dari total biaya *maintenance* tanpa *preventive maintenance*.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, dapat disimpulkan bahwa:

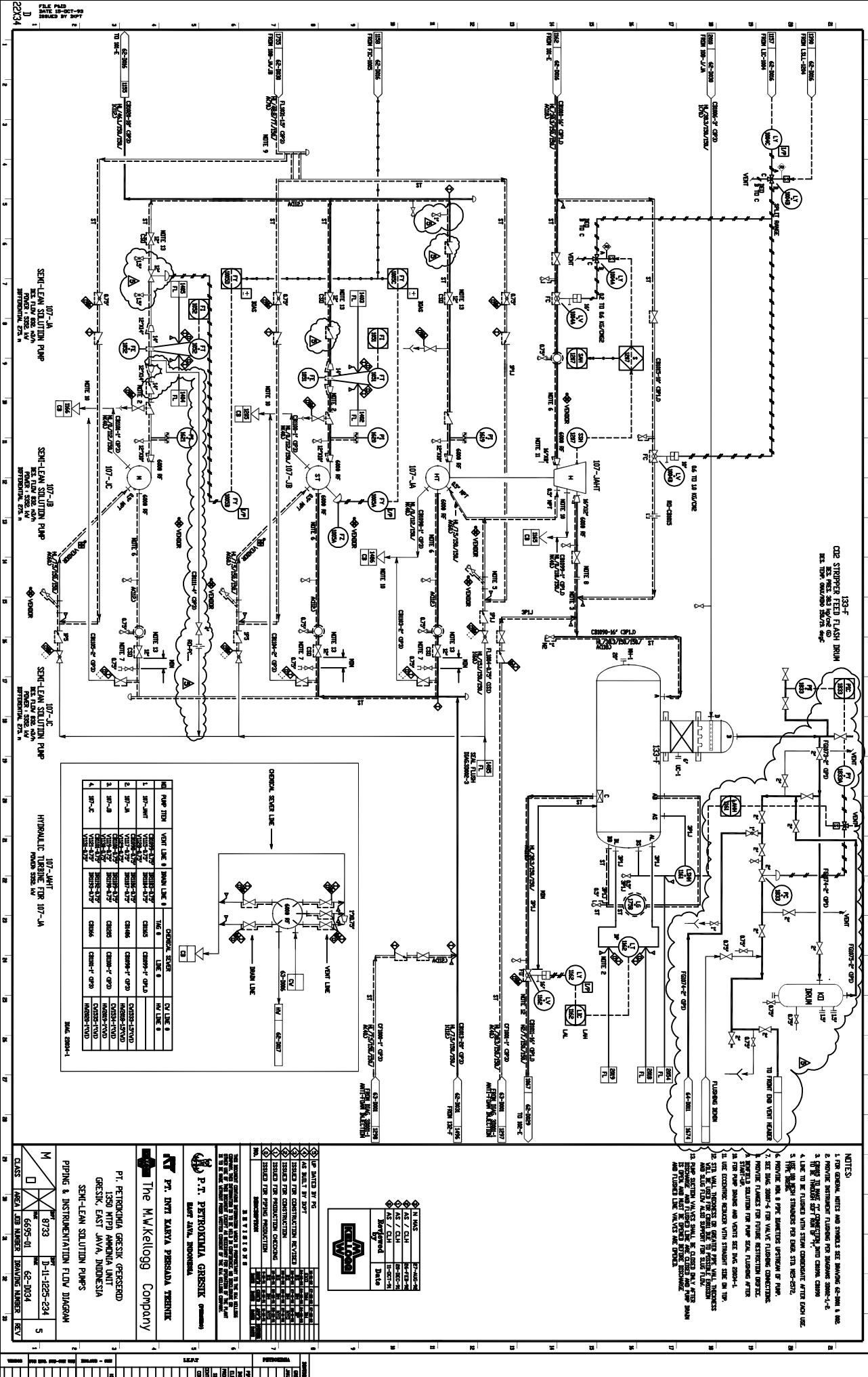
- Optimasi *preventive maintenance* menggunakan metode PSO (*Particle Swarm Optimization*) menghasilkan penjadwalan *preventive maintenance* secara sistem pompa untuk interval 110 jam.
- Optimasi *preventive maintenance* yang dilakukan pada pompa 107-JC dengan interval 110 jam, menghasilkan nilai *reliability* kritis yaitu 0,653. Dimana selama rentang waktu 1430 jam, kegiatan *preventive maintenance* yang dilakukan mampu menjaga nilai *reliability* antara 0,6 sampai 1.
- Selama rentang waktu 1430 jam, hasil optimasi membutuhkan biaya total *maintenance* sebesar Rp 23.980.160,00

5.2. Saran

Dalam penelitian selanjutnya, diharapkan dapat dilakukan optimasi pada sistem pompa yang sama dengan metode yang berbeda. Selain itu, dapat mempertimbangkan kapasitas proses produksi ketika sebelum dilakukan *maintenance* dengan setelah dilakukan *maintenance*. Penjadwalan *predictive maintenance* pada pompa ini juga dapat dicari untuk lebih meminimalisasi *cost maintenance*.

Halaman ini memang dikosongkan

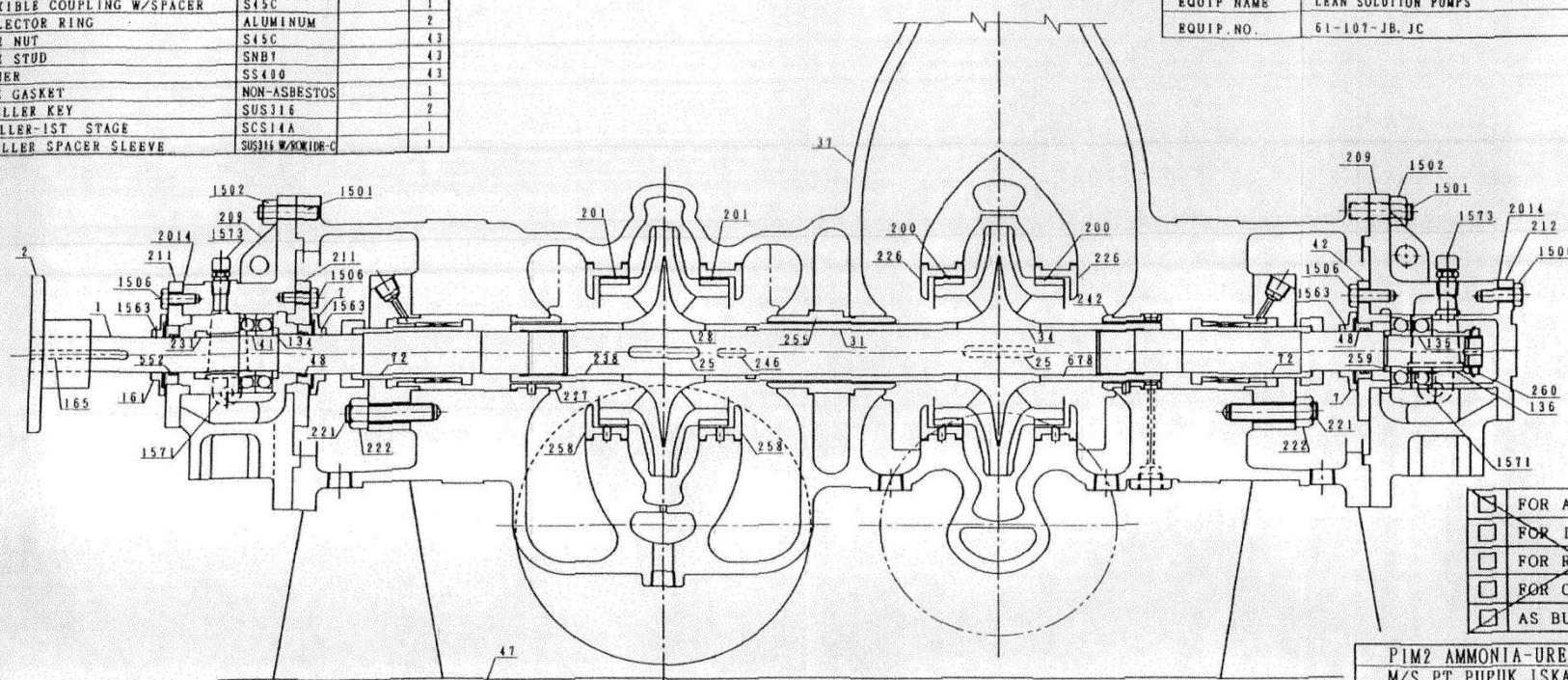
Lampfah A. F&ID semi Lean Solution Pump 10-JC



Lampiran B. Bagian-Bagian Pompa 107-JC

PART NO	NAME OF PART	MATERIAL	REMARKS	Q'TY
1	PUMP SHAFT	17-1PH		1
2	FLEXIBLE COUPLING W/SPACER	S45C		1
7	DEFLECTOR RING	ALUMINUM		2
18	CASE NUT	S45C		43
19	CASE STUD	SNB1		43
20	WASHER	SS400		43
22	CASE GASKET	NON-ASBESTOS		1
25	IMPELLER KEY	SUS316		2
28	IMPELLER-1ST STAGE	SCS14A		1
31	IMPELLER SPACER SLEEVE	SUS316 W/RONIDR-C		1

CLIENT	TBC/PT.PUPUK ISKANDAR MUDA
EQUIP NAME	LEAN SOLUTION PUMPS
EQUIP.NO.	61-107-JB. JC



<input type="checkbox"/>	FOR APPROVAL
<input type="checkbox"/>	FOR INFORMATION
<input type="checkbox"/>	FOR REVIEW
<input type="checkbox"/>	FOR CONSTRUCTION
<input type="checkbox"/>	AS BUILT/FINAL

PIM2 AMMONIA-UREA PROJECT
M/S PT PUPUK ISKANDAR MUDA
LHOKSUMAWE, NORTH ACEH, INDONESIA
Toyo Engineering Corp.
TEC WORK NO.: BA-0456
PIM2 PROJECT

REQ. NO. AXGA003
ITEM NO. 61-107-JB, JC

SWN MFG NO.	MUDA-6P-61526		
SIZE & TYPE	8 x 15 RHC		
API MAT'L CODE	A-B		
REV. NO.	18	19	20
DATE	06/27/03	FOR APPROVAL	
NOTE			
REVIEWED			
APPROVED			

SHIN NIPPON MACHINERY CO., LTD.

PUMP DIVISION

CROSS SECTIONAL DRAWING

DRAWING NO. 3PS-61526 (2)

PART NO	NAME OF PART	MATERIAL	REMARKS	Q'TY
34	IMPELLER-2ND STAGE	SCS14A		1
37	CASE	SCS14A		1
41	RADIAL BALL BEARING		6311D2C3	1
42	BALL BRG.HSG.COVER-THR-INNER	SCPH2		1
47	BASE	SS400		1
48	STATIONARY OIL BAFFLE	FC250		2
72	SHAFT SLEEVE	SUS316		2
134	BALL BEARING SPACER RING	SS400		1
135	THRUST BALL BEARING		T311BDB	1
136	BALL BEARING THRUST COLLAR	SS400		1
161	DEFLECTOR RING-INTERNAL	ALUMINUM		1
165	FLEXIBLE COUPLING KEY	S45C		1
200	IMPELLER WEARING 2ND STAGE	SUS316 W/RONIDR-C HB2400		2
201	IMPELLER WEARING 1ST STAGE	SUS316 W/RONIDR-C HB2400		2
209	BALL BEARING HOUSING	SCPH2		2
211	BALL BRG.HSG.COVER-RADIAL	SS400		2
212	BALL BRG.HSG.COVER-THR-OUTER	SCPH2		1
221	PACKING GLAND STUD	SUS316		8
222	PACKING GLAND NUT	SUS316		8
226	CASE WEARING 2ND STAGE	SUS316 W/RONIDR-C HB2400		2
227	PACKING BOX BUSHING	SUS316 W/NITRIDING		1

THE DETAIL OF CASING BOLT
AND JACKET BOLT

Lampiran C



PT. PETROKIMIA GRESIK DEPARTEMEN PEMELIHARAAN I BAGIAN MEKANIK I

KARTU SEJARAH PERALATAN

Unit / Pabrik : Amoniak / Pabrik I
No. Peralatan : 107 JC
Tahun Dioperasikan : 1994
Pabrik Pembuat : Ingersoll Rand

No	Actual Start	Kondisi	Perbaikan	Keterangan
1	15-Jun-10	Low Capacity/Flow tidak mencukupi	Over-Haul 107-JC a. Ganti Rotor baru (PT Mahita) b. Ganti wearing casing sisi outboard c. Ganti throttle bushing sisi outboard d. Ganti Journal Bearing e. Ganti Thrust bearing QJ 312 FAG f. Ganti Mech. Seal (in & outboard) g. Ganti semua O'ring h. Alignment	Clearance Journal Bearing : InBoard : 0,12 mm OutBoard : 0,13 mm Clearance Wearing : InBoard : 0,80 mm OutBoard : 0,80 mm Axl. Clearance : 0,11 mm
2	15-Jun-10	Mech seal bocor sisi in board	Modifikasi mech seal dengan memasang seal dari teflon pada sisi mech yang bocor	

No	Actual Start	Kondisi	Perbaikan	Keterangan
3	22-Sep-10	<ul style="list-style-type: none"> a. Mech seal pecah sisi out board b. Throttle bushing sisi out hancur (tersisa 1/3 bagian) c. Throttle bushing sisi in aus/ lubang pin rusak d. Pin patah pada posisi throttle bushing sisi out e. Shaft sleeve sisi out aus 	<ul style="list-style-type: none"> a. Ganti rotor b. Ganti mech seal sisi in dan outboard c. Ganti Thrust bearing pompa d. Ganti throttle bushing sisi IN dan OUT e. Repair shaft sleeve f. Ganti Pin g. Ganti O-ring baru: Wearing impeller : kalrez 270 Throttle bushing : kalrez 249 Lock nut impeller : kalrez 239 (152) Ganti gasket line flushing dll (3/4 in x 600; 1/2 in x 600) 	
4	24-Sep-10	Start 107 JC		Hasil pengukuran vib. H 0.80 0.75 1.46 1.62 V 0.74 0.85 2.11 0.83 A 0.85 1.24 2.33 1.79
5	05-Agust-11	Saat start, shaft modot keluar (sisi out), vibrasi tinggi (dugaan shaft patah, ternyata tidak)	<ul style="list-style-type: none"> a. Ganti rotor spare ex. 107 JA b. Spare part semua baru, kecuali bearing inboard dan shaft sleeve sisi inboard c. Data clearance: <ul style="list-style-type: none"> • Wearing casing IN: 0.8 mm, OUT : 0.8 mm • Throttle bushing IN/OUT : 0.7 mm • Shaft sleeve IN/OUT dipotong 3 mm dari dimensi semula 	

No	Actual Start	Kondisi	Perbaikan	Keterangan
5			<ul style="list-style-type: none"> • Journal bearing IN/OUT : 0.18/ 0.12 (dengan bearing baru, housing bearing tidak bisa ditutup) • Axial bearing 0.19 (0.1- 0.15 mm) (pernah mencapai 0,24 mm) • Gap Coupling 185 mm d. Ganti basic seal baru inboard dan outboard e. Matl shaft SS 630 	
6	07-Nop-11	a. Babbit Journal bearing mengelupas (coupling side) b. Thrust side kondisinya bagus (clearance 0,18) c. Mech seal inboard pecah	a. Ganti journal bearing (Inboard) b. Ganti thrust bearing, 0,9 mm c. Ganti basic seal inboard d. Ganti shaft sleeve inboard	Clearance journal 0,15
7	09-Des-11	a. Vibrasi tinggi ($H_2 = 7$ mm/s) b. Laporan ISUS : Indikasi blade pass, temp benfield desain indikasi missalignment c. Motor unbalance	Over haul, cek rotor a. Ganti thrust bearing b. Cek alignment c. Insitu balancing	

No	Actual Start	Kondisi	Perbaikan	Keterangan
8	17-Feb-12	a. Vibrasi tinggi (Lihat data ISUS) b. Lock nut in dan out kendor	a. Ganti rotor ex repair b. Ganti journal bearing inboard dan outboard (clea. 0,12) c. Ganti thrust bearing (0,11) d. Ganti basic seal ex. Repair inboard dan outboard	Tebal gasket 3 mm
9	03-Jun-12	Basic seal pecah (Inboard)	Ganti basic seal baru	
10	05-Jun-12	Basic seal pecah (Inboard)	Ganti basic seal baru	
11	06-Jun-12	a. Mechanical seal pecah b. Journal bearing/babbitt mengelupas c. Throttle bushing aus d. Shaft sleeve kepanjangan 2 mm	a. Ganti basic seal ex repair b. Ganti journal bearing c. Ganti throttle bushing d. Sesuaikan panjang shaft sleeve	a. Tidak ganti rotor b. ASTM 23 - Grade 2
12	10-Nop-12	a. Pompa low capacity b. Wearing impeller sisi inboard lepas	a. Modifikasi wearing impeller dengan menambah lock atau baut pengunci b. Repair stoper wearing	
13	14-Nop-12	Mech seal inboard pecah	Ganti mech seal 1 set	
14	10-Des-12	a. Vibrasi tinggi V3p = 1,5 mm/s V3c = 9,1 mm/s H1 = 5 mm/s H2 = 3,5 mm/s	a. Ganti rotor eks repair (shaft baru) panjang hub = 75 mm centre pasak = 618,5 mm b. Ganti basic seal (lapping) c. Ganti shaft sleeve	

No	Actual Start	Kondisi	Perbaikan	Keterangan
14		b. Low capacity 1180 T/j (JA dan JC) c. Throttle bushing Inboard aus/miring d. O-ring wearing casing dan O-ring busing putus e. Gasket pada area centre volute mengelupas/mlesat (gasket 3 mm)	Inboard = 143 mm Outboar = 150 mm d. Ganti wearing casing Inboard = Outboard = e. Ganti throttle bushing Inboard = 129,40 mm Outboar = 129,40 mm	
15	23-Apr-13	Overhaul (Throttle Bushing lepas)	a. Ganti mechanical seal b. Ganti rotor c. Ganti journal bearing d. Ganti wearing chasing e. Ganti throttle bushing	
16	24-Sep-13	Overhaul	a. Ganti mechanical seal b. Ganti rotor c. Ganti journal bearing d. Ganti wearing chasing e. Ganti throttle bushing	
17	11-Des-13	Overhaul (Mechanical seal inboard pompa bocor)	a. Ganti mechanical seal b. Ganti rotor c. Ganti journal bearing d. Ganti wearing chasing e. Ganti throttle bushing	

Lampiran D

Waktu Perbaikan Komponen

Rotor : Perbaikan selama ± 8 jam

No	Tanggal	Waktu Perbaikan		Jenis Perbaikan	TTF
		Mulai	Selesai		
1	15 Juni 2010	08:10	11:50	Ganti	0
		13:15	17:10		
2	22 September 2010	07:55	11:45	Ganti	2366,75
		13:10	17:25		
3	5 Agustus 2011	08:15	11:52	Ganti	7598,83
		13:17	17:28		
4	9 Desember 2011	07:58	11:40	Ganti	3014,50
		13:15	16:55		
5	17 Februari 2012	08:05	11:28	Ganti	1671,17
		13:10	17:25		
6	10 Desember 2012	08:17	11:26	Ganti	7118,87
		13:30	16:38		
7	23 April 2013	07:45	11:30	Ganti	3207,12
		13:12	16:48		
8	24 September 2013	08:05	11:47	Ganti	3687,28
		13:15	16:40		
9	11 Desember 2013	07:55	11:27	Ganti	1863,25
		13:25	16:58		

Wearing Chasing (Perbaikan selama ± 8 jam)

No	Tanggal	Waktu Perbaikan		Jenis Perbaikan	TTF
		Mulai	Selesai		
1	15 Juni 2010	08:10	11:50	Ganti	0
		13:15	17:10		
2	5 Agustus 2011	08:15	11:52	Ganti	9975,08
		13:17	17:28		
3	10 Desember 2012	08:17	11:26	Ganti	11822,82
		13:30	16:38		
4	23 April 2013	07:45	11:30	Ganti	3207,12
		13:12	16:48		

Wearing Chasing (Lanjutan)

No	Tanggal	Waktu Perbaikan		Jenis Perbaikan	TTF
		Mulai	Selesai		
5	24 September 2013	08:05	11:47	Ganti	3687,28
		13:15	16:40		
6	11 Desember 2013	07:55	11:27	Ganti	7173,08
		13:25	16:58		

Throttle Bushing (Perbaikan selama ± 8 jam)

No	Tanggal	Waktu Perbaikan		Jenis Perbaikan	TTF
		Mulai	Selesai		
1	15 Juni 2010	08:10	11:50	Ganti	0
		13:15	17:10		
2	22 September 2010	07:55	11:45	Ganti	2366,75
		13:10	17:25		
3	5 Agustus 2011	08:15	11:52	Ganti	7598,83
		13:17	17:28		
4	6 Juni 2012	08:12	11:28	Ganti	7334,73
		13:11	17:24		
5	10 Desember 2012	08:17	11:26	Ganti	4478,88
		13:30	16:38		
6	23 April 2013	07:45	11:30	Ganti	3207,12
		13:12	16:48		
7	24 September 2013	08:05	11:47	Ganti	3687,28
		13:15	16:40		
8	11 Desember 2013	07:55	11:27	Ganti	1863,25
		13:25	16:58		

Journal Bearing (Perbaikan selama ± 2 jam)

No	Tanggal	Waktu Perbaikan		Jenis Perbaikan	TTF
		Mulai	Selesai		
1	15 Juni 2010	09:15	11:37	Ganti	0
2	5 Agustus 2011	08:45	10:25	Ganti	9981,13
3	7 Nopember 2011	09:10	11:20	Ganti	2254,75
4	17 Februari 2012	08:55	11:07	Ganti	2445,58

Journal Bearing (Lanjutan)

No	Tanggal	Waktu Perbaikan		Jenis Perbaikan	TTF
		Mulai	Selesai		
5	6 Juni 2012	09:46	11:45	Ganti	2638,65
6	23 April 2013	08:45	11:40	Ganti	7701,00
7	24 September 2013	08:50	11:58	Ganti	3693,17
8	11 Desember 2013	09:06	11:15	Ganti	1869,13

Thrust Bearing (Perbaikan selama ±2 jam)

No	Tanggal	Waktu Perbaikan		Jenis perbaikan	TTF
		Mulai	Selesai		
1	15 Juni 2010	13:20	15:10	Ganti	0
2	22 September 2010	13:25	15:33	Ganti	2374,25
3	5 Agustus 2011	13:41	15:45	Ganti	7606,13
4	7 Nopember 2011	08:47	10:55	Ganti	2249,03
5	9 Desember 2011	08:30	10:17	Ganti	765,58
6	17 Februari 2012	13:25	15:20	Ganti	1683,13

Mechanical Seal (Perbaikan selama ±4 jam)

No	Tanggal	Waktu Perbaikan		Jenis perbaikan	TTF
		Mulai	Selesai		
1	15 Juni 2010	08:10	11:50	Ganti	0
2	22 September 2010	07:55	11:45	Ganti	2372,08
3	5 Agustus 2011	08:15	11:52	Ganti	7604,50
4	7 Nopember 2011	07:53	11:37	Ganti	2252,02
5	9 Desember 2011	08:13	11:57	Ganti	764,60
6	17 Februari 2012	08:05	11:28	Ganti	1676,13
7	3 Juni 2012	07:47	11:55	Ganti	2564,32
8	5 Juni 2012	07:55	11:32	Ganti	44,00
9	6 Juni 2012	07:51	11:43	Ganti	20,32
10	14 Nopember 2012	08:05	11:46	Ganti	3860,37
11	10 Desember 2012	07:56	11:54	Ganti	620,17
12	23 April 2013	08:03	11:47	Ganti	3212,15
13	24 September 2013	08:13	11:30	Ganti	3692,43
14	11 Desember 2013	07:58	11:32	Ganti	1868,47

O-Ring (Perbaikan Selama ± 10 menit)

No	Tanggal	Waktu Perbaikan		Jenis perbaikan	TTF
		Mulai	Selesai		
1	15 Juni 2010	10:12	10:20	Ganti	0
2	22 September 2010	09:55	10:09	Ganti	2375,58
3	5 Agustus 2011	10:34	10:45	Ganti	7608,42
4	10 Nopember 2012	10:23	10:35	Ganti	11111,63
5	10 Desember 2012	09:46	09:55	Ganti	719,18
6	23 April 2013	09:30	09:42	Ganti	3215,58
7	24 September 2013	10:11	10:19	Ganti	3696,48
8	11 Desember 2013	10:02	10:13	Ganti	1871,72

Shaft Sleeve (Perbaikan Selama ± 4 jam)

No	Tanggal	Waktu Perbaikan		Jenis perbaikan	TTF
		Mulai	Selesai		
1	22 September 2010	07:55	11:45	Repair	0
2	5 Agustus 2011	08:15	11:52	Repair	7604,50
3	7 Nopember 2011	07:53	11:37	Ganti	2252,02
4	6 Juni 2012	07:51	11:43	Repair	5084,23
5	10 Desember 2012	07:56	11:54	Ganti	4484,22

Lampiran E

Metode Empiris *Reliability* dan *Failure Rate*

❖ Rotor

Sebelum dilakukan perhitungan *reliability* dan *failure rate* dengan metode empiris, TTF dari data kerusakan harus diurutkan dari yang terkecil hingga terbesar.

i	TTF
0	0
1	1671,17
2	1863,25
3	2366,75
4	3014,50
5	3207,12
6	3687,28
7	7118,87
8	7598,83

Untuk menghitung *reliability* dan *failure rate* tiap kerusakan, diperoleh dari Persamaan (2.1) dan Persamaan (2.2) dengan contoh perhitungan TTF ke $i = 1$.

- Perhitungan *reliability*

$$R(t_i) = \frac{n + 1 - i}{n + 1}$$

$$R(t_1) = \frac{8 + 1 - 1}{8 + 1}$$

$$R(t_1) = 0,889$$

- Perhitungan *failure rate*

$$\lambda(t_i) = \frac{1}{(t_{i+1} - t_i)(n + 1 - i)}$$

$$\lambda(t_1) = \frac{1}{(1863,25 - 1671,17) \times (8 + 1 - 1)}$$

$$\lambda(t_1) = 0,000651$$

Dengan cara yang sama, perhitungan *reliability* dan *failure rate* tiap kerusakan dapat diperoleh pada tabel berikut.

i	TTF	R	λ
0	0	1,000	0,000066
1	1671,17	0,889	0,000651
2	1863,25	0,778	0,000284
3	2366,75	0,667	0,000257
4	3014,50	0,556	0,001038
5	3207,12	0,444	0,000521
6	3687,28	0,333	0,000097
7	7118,87	0,222	0,001042
8	7598,83	0,111	

❖ *Wearing Chasing*

Sebelum dilakukan perhitungan *reliability* dan *failure rate* dengan metode empiris, TTF dari data kerusakan harus diurutkan dari yang terkecil hingga terbesar.

i	TTF
0	0
1	3207,12
2	3687,28
3	7173,08
4	9975,08
5	11822,82

Untuk menghitung *reliability* dan *failure rate* tiap kerusakan, diperoleh dari Persamaan (2.1) dan Persamaan (2.2) dengan contoh perhitungan TTF ke $i = 1$.

- Perhitungan *reliability*

$$R(t_i) = \frac{n + 1 - i}{n + 1}$$

$$R(t_1) = \frac{5 + 1 - 1}{5 + 1}$$

$$R(t_1) = 0,833$$

- Perhitungan *failure rate*

$$\lambda(t_i) = \frac{1}{(t_{i+1} - t_i)(n + 1 - i)}$$

$$\lambda(t_1) = \frac{1}{(3687,28 - 3207,12) \times (5 + 1 - 1)}$$

$$\lambda(t_1) = 0,0004165$$

Dengan cara yang sama, perhitungan *reliability* dan *failure rate* tiap kerusakan dapat diperoleh pada tabel berikut.

i	TTF	R	λ
0	0	1,000	0,0000520
1	3207,12	0,833	0,0004165
2	3687,28	0,667	0,0000717
3	7173,08	0,500	0,0001190
4	9975,08	0,333	0,0002706
5	11822,82	0,167	

➔ ***Throttle Bushing***

Sebelum dilakukan perhitungan *reliability* dan *failure rate* dengan metode empiris, TTF dari data kerusakan harus diurutkan dari yang terkecil hingga terbesar.

i	TTF
0	0
1	1863,25
2	2366,75
3	3207,12
4	3687,28
5	4478,88
6	7334,73
7	7598,83

Untuk menghitung *reliability* dan *failure rate* tiap kerusakan, diperoleh dari Persamaan (2.1) dan Persamaan (2.2) dengan contoh perhitungan TTF ke i = 1.

- Perhitungan *reliability*

$$R(t_i) = \frac{n + 1 - i}{n + 1}$$

$$R(t_1) = \frac{7 + 1 - 1}{7 + 1}$$

$$R(t_1) = 0,875$$

- Perhitungan *failure rate*

$$\lambda(t_i) = \frac{1}{(t_{i+1} - t_i)(n + 1 - i)}$$

$$\lambda(t_1) = \frac{1}{(2366,75 - 1863,25) \times (7 + 1 - 1)}$$

$$\lambda(t_1) = 0,0002837$$

Dengan cara yang sama, perhitungan *reliability* dan *failure rate* tiap kerusakan dapat diperoleh pada tabel berikut.

i	TTF	R	λ
0	0	1,000	0,0000671
1	1863,25	0,875	0,0002837
2	2366,75	0,750	0,0001983
3	3207,12	0,625	0,0004165
4	3687,28	0,500	0,0003158
5	4478,88	0,375	0,0001167
6	7334,73	0,250	0,0018932
7	7598,83	0,125	

❖ *Journal Bearing*

Sebelum dilakukan perhitungan *reliability* dan *failure rate* dengan metode empiris, TTF dari data kerusakan harus diurutkan dari yang terkecil hingga terbesar.

i	TTF
0	0
1	1869,13
2	2254,75
3	2445,58
4	2638,65
5	3693,17
6	7701,00
7	9981,13

Untuk menghitung *reliability* dan *failure rate* tiap kerusakan, diperoleh dari Persamaan (2.1) dan Persamaan (2.2) dengan contoh perhitungan TTF ke $i = 1$.

- Perhitungan *reliability*

$$R(t_i) = \frac{n+1-i}{n+1}$$

$$R(t_1) = \frac{7+1-1}{7+1}$$

$$R(t_1) = 0,875$$

- Perhitungan *failure rate*

$$\lambda(t_i) = \frac{1}{(t_{i+1} - t_i)(n+1-i)}$$

$$\lambda(t_1) = \frac{1}{(2254,75 - 1869,13) \times (7+1-1)}$$

$$\lambda(t_1) = 0,0003705$$

Dengan cara yang sama, perhitungan *reliability* dan *failure rate* tiap kerusakan dapat diperoleh pada tabel berikut.

i	TTF	R	λ
0	0	1,000	0,0000669
1	1869,13	0,875	0,0003705
2	2254,75	0,750	0,0008734
3	2445,58	0,625	0,0010359
4	2638,65	0,500	0,0002371
5	3693,17	0,375	0,0000832
6	7701,00	0,250	0,0002193
7	9981,13	0,125	

➔ **Thrust Bearing**

Sebelum dilakukan perhitungan *reliability* dan *failure rate* dengan metode empiris, TTF dari data kerusakan harus diurutkan dari yang terkecil hingga terbesar.

i	TTF
0	0
1	765,58
2	1683,13
3	2249,03
4	2374,25
5	7606,13

Untuk menghitung *reliability* dan *failure rate* tiap kerusakan, diperoleh dari Persamaan (2.1) dan Persamaan (2.2) dengan contoh perhitungan TTF ke i = 1.

- Perhitungan *reliability*

$$R(t_i) = \frac{n+1-i}{n+1}$$

$$R(t_1) = \frac{5+1-1}{5+1}$$

$$R(t_1) = 0,833$$

- Perhitungan *failure rate*

$$\lambda(t_i) = \frac{1}{(t_{i+1} - t_i)(n+1-i)}$$

$$\lambda(t_1) = \frac{1}{(1683,13 - 765,58) \times (5+1-1)}$$

$$\lambda(t_1) = 0,000218$$

Dengan cara yang sama, perhitungan *reliability* dan *failure rate* tiap kerusakan dapat diperoleh pada tabel berikut.

i	TTF	R	λ
0	0	1,000	0,000218
1	765,58	0,833	0,000218

i	TTF	R	λ
2	1683,13	0,667	0,000442
3	2249,03	0,500	0,002662
4	2374,25	0,333	0,000096
5	7606,13	0,167	

❖ Mechanical Seal

Sebelum dilakukan perhitungan *reliability* dan *failure rate* dengan metode empiris, TTF dari data kerusakan harus diurutkan dari yang terkecil hingga terbesar.

i	TTF
0	0,00
1	20,32
2	44,00
3	620,17
4	764,60
5	1676,13
6	1868,47
7	2252,02
8	2372,08
9	2564,32
10	3212,15
11	3692,43
12	3860,37
13	7604,50

Untuk menghitung *reliability* dan *failure rate* tiap kerusakan, diperoleh dari Persamaan (2.1) dan Persamaan (2.2) dengan contoh perhitungan TTF ke i = 1.

- Perhitungan *reliability*

$$R(t_i) = \frac{n+1-i}{n+1}$$

$$R(t_1) = \frac{13+1-1}{13+1}$$

$$R(t_1) = 0,929$$

- Perhitungan *failure rate*

$$\lambda(t_i) = \frac{1}{(t_{i+1} - t_i)(n+1-i)}$$

$$\lambda(t_1) = \frac{1}{(44,00 - 20,32) \times (13 + 1 - 1)}$$

$$\lambda(t_1) = 0,0032484$$

Dengan cara yang sama, perhitungan *reliability* dan *failure rate* tiap kerusakan dapat diperoleh pada tabel berikut.

i	TTF	R	λ
0	0,00	1,000	0,0035152
1	20,32	0,929	0,0032484
2	44,00	0,857	0,0001446
3	620,17	0,786	0,0006294
4	764,60	0,714	0,0001097
5	1676,13	0,643	0,0005777
6	1868,47	0,571	0,0003259
7	2252,02	0,500	0,0011899
8	2372,08	0,429	0,0008670
9	2564,32	0,357	0,0003087
10	3212,15	0,286	0,0005205
11	3692,43	0,214	0,0019848

i	TTF	R	λ
12	3860,37	0,143	0,0001335
13	7604,50	0,071	

❖ **O-Ring**

Sebelum dilakukan perhitungan *reliability* dan *failure rate* dengan metode empiris, TTF dari data kerusakan harus diurutkan dari yang terkecil hingga terbesar.

i	TTF
0	0
1	719,183
2	1871,717
3	2375,583
4	3215,583
5	3696,483
6	7608,417
7	11111,633

Untuk menghitung *reliability* dan *failure rate* tiap kerusakan, diperoleh dari Persamaan (2.1) dan Persamaan (2.2) dengan contoh perhitungan TTF ke i = 1.

- Perhitungan *reliability*

$$R(t_i) = \frac{n + 1 - i}{n + 1}$$

$$R(t_1) = \frac{7 + 1 - 1}{7 + 1}$$

$$R(t_1) = 0,875$$

- Perhitungan *failure rate*

$$\lambda(t_i) = \frac{1}{(t_{i+1} - t_i)(n + 1 - i)}$$

$$\lambda(t_1) = \frac{1}{(1871,717 - 719,183) \times (7 + 1 - 1)}$$

$$\lambda(t_1) = 0,000124$$

Dengan cara yang sama, perhitungan *reliability* dan *failure rate* tiap kerusakan dapat diperoleh pada tabel berikut.

i	TTF	R	λ
0	0	1,000	0,0001738
1	719,183	0,875	0,0001240
2	1871,717	0,750	0,0003308
3	2375,583	0,625	0,0002381
4	3215,583	0,500	0,0005199
5	3696,483	0,375	0,0000852
6	7608,417	0,250	0,0001427
7	11111,633	0,125	

➔ *Shaft Sleeve*

Sebelum dilakukan perhitungan *reliability* dan *failure rate* dengan metode empiris, TTF dari data kerusakan harus diurutkan dari yang terkecil hingga terbesar.

i	TTF
0	0
1	2252,02
2	4484,22
3	5084,23
4	7604,50

Untuk menghitung *reliability* dan *failure rate* tiap kerusakan, diperoleh dari Persamaan (2.1) dan Persamaan (2.2) dengan contoh perhitungan TTF ke i = 1.

- Perhitungan *reliability*

$$R(t_i) = \frac{n+1-i}{n+1}$$

$$R(t_1) = \frac{4+1-1}{4+1}$$

$$R(t_1) = 0,800$$

- Perhitungan *failure rate*

$$\lambda(t_i) = \frac{1}{(t_{i+1} - t_i)(n+1-i)}$$

$$\lambda(t_1) = \frac{1}{(4484,22 - 2252,02) \times (4+1-1)}$$

$$\lambda(t_1) = 0,000112$$

Dengan cara yang sama, perhitungan *reliability* dan *failure rate* tiap kerusakan dapat diperoleh pada tabel berikut.

i	TTF	R	λ
0	0	1,000	0,0000888
1	2252,02	0,800	0,0001120
2	4484,22	0,600	0,0005555
3	5084,23	0,400	0,0001984
4	7604,50	0,200	

Lampiran F

Model Dynamic Reliability

Hasil perhitungan *reliability* dengan metode empiris kemudian didekti dengan model *dynamic reliability* yang divalidasi dengan MSE. Untuk konstanta R_0 dan A_0 diperoleh dari perhitungan *reliability* dan *failure rate* dengan metode empiris. Tabel berikut merupakan perbandingan nilai A_1 yang dicari sehingga diperoleh MSE terkecil.

Komponen	A_1	MSE
Rotor	0,00073	0,0060948
	0,00074	0,0060786
	0,00075	0,0060836
<i>Wearing Chasing</i>	0,00019	0,0023118
	0,00020	0,0021598
	0,00021	0,0021971
<i>Throttle Bushing</i>	0,00046	0,0030414
	0,00047	0,0029977
	0,00048	0,0029980
<i>Journal Bearing</i>	0,00073	0,0118133
	0,00074	0,0118066
	0,00075	0,0118153
<i>Thrust Bearing</i>	0,00041	0,0065914
	0,00042	0,0065855
	0,00043	0,0065872
<i>Mechanical Seal</i>	$9,9 \times 10^{-8}$	0,1656004
	$1,0 \times 10^{-9}$	0,1656004
	$1,1 \times 10^{-9}$	0,1656004
<i>O-Ring</i>	0,000082	0,0024577
	0,000083	0,0024574
	0,000084	0,0024575
<i>Shaft Sleeve</i>	0,00025	0,0024529
	0,00026	0,0024358
	0,00027	0,0025054

Nilai A_1 yang tepat dengan MSE terkecil ditunjukkan dengan warna biru muda. Diatas dan dibawah nilai A_1 terpilih menghasilkan MSE yang lebih besar. Nilai MSE diketahui dari perbandingan nilai *reliability* metode empiris (R) dengan metode *dynamic reliability* ($R(t)$). Perhitungan MSE masing-masing komponen ditunjukkan pada tabel berikut.

➔ Rotor ($A_1 = 0,00074$)

R	R(t)	E	E²
1,000	1,000	0,000	0,0000000
0,889	0,811	0,078	0,0060819
0,778	0,778	0,000	0,0000001
0,667	0,680	-0,013	0,0001790
0,556	0,541	0,015	0,0002209
0,444	0,499	-0,055	0,0030218
0,333	0,397	-0,064	0,0041274
0,222	0,038	0,184	0,0338657
0,111	0,026	0,085	0,0072109
MSE		0,0060786	

➔ *Wearing chasing* ($A_1 = 0,00020$)

R	R(t)	E	E²
1,000	1,000	0,000	0,0000000
0,833	0,796	0,037	0,0014038
0,667	0,760	-0,093	0,0086878
0,500	0,488	0,012	0,0001530
0,333	0,299	0,034	0,0011529
0,167	0,206	-0,040	0,0015613
MSE		0,0021598	

→ *Throttle Bushing* ($A_1 = 0,00047$)

R	R(t)	E	E^2
1,000	1,000	0,000	0,0000000
0,875	0,822	0,053	0,0028609
0,750	0,755	-0,005	0,0000219
0,625	0,630	-0,005	0,0000287
0,500	0,556	-0,056	0,0031063
0,375	0,435	-0,060	0,0035924
0,250	0,131	0,119	0,0142691
0,125	0,115	0,010	0,0001050
MSE			0,0029980

→ *Journal Bearing* ($A_1 = 0,00074$)

R	R(t)	E	E^2
1,000	1,000	0,000	0,0000000
0,875	0,776	0,099	0,0097656
0,750	0,703	0,047	0,0022010
0,625	0,664	-0,039	0,0015408
0,500	0,624	-0,124	0,0152874
0,375	0,399	-0,024	0,0005557
0,250	0,025	0,225	0,0505554
0,125	0,004	0,121	0,0146003
MSE			0,0118133

→ *Thrust Bearing* ($A_1 = 0,00042$)

R	R(t)	E	E^2
1,000	1,000	0,000	0,0000000
0,833	0,823	0,010	0,0001085
0,667	0,603	0,064	0,0040500
0,500	0,478	0,022	0,0004949
0,333	0,452	-0,119	0,0140785

➔ *Thrust Bearing* (Lanjutan)

R	R(t)	E	E^2
0,167	0,023	0,144	0,0207812
MSE		0,0065855	

➔ *Mechanical Seal* ($A_1 = 1,0 \times 10^{-9}$)

R	R(t)	E	E^2
1,000	1,000	0,000	0,0000000
0,929	0,931	-0,002	0,0000062
0,857	0,857	0,000	0,0000002
0,786	0,113	0,673	0,4524906
0,714	0,068	0,646	0,4176381
0,643	0,003	0,640	0,4097222
0,571	0,001	0,570	0,3249274
0,500	0,000	0,500	0,2496354
0,429	0,000	0,428	0,1834685
0,357	0,000	0,357	0,1274641
0,286	0,000	0,286	0,0816255
0,214	0,000	0,214	0,0459174
0,143	0,000	0,143	0,0204078
0,071	0,000	0,071	0,0051020
MSE		0,1656004	

➔ *O-Ring* ($A_1 = 0,000083$)

R	R(t)	E	E^2
1,000	1,000	0,000	0,0000000
0,875	0,879	-0,004	0,0000181
0,750	0,705	0,045	0,0020096
0,625	0,637	-0,012	0,0001465
0,500	0,535	-0,035	0,0012016
0,375	0,482	-0,107	0,0114739

❖ *O-Ring* (Lanjutan)

R	R(t)	E	E^2
0,250	0,195	0,055	0,0029861
0,125	0,082	0,043	0,0018240
MSE		0,0024575	

❖ *Shaft Sleeve* ($A_1 = 0,00026$)

R	R(t)	E	E^2
1,000	1,000	0,000	0,0000000
0,800	0,765	0,035	0,0012495
0,600	0,503	0,097	0,0094780
0,400	0,437	-0,037	0,0013932
0,200	0,220	-0,020	0,0004065
MSE		0,0025054	

Lampiran G
Jumlah Perbaikan dan Cost Replace Component

No	Komponen	Jumlah Perbaikan	Harga Komponen Satuan	
			Rupiah	US\$
1	Rotor	9	7.500.000	661,78
2	<i>Wearing Chasing</i>	6	2.000.000	176,48
3	<i>Throttle Bushing</i>	8	1.130.000	99,71
4	<i>Journal Bearing</i>	8	8.000.000	705,90
5	<i>Thrust Bearing</i>	6	2.000.000	176,48
6	<i>Mechanical Seal</i>	14	78.077.520	6889,40
7	<i>O-ring</i>	8	5.000.000	441,19
8	<i>Shaft Sleeve</i>	5	2.000.000	176,48
9	<i>Wearing Impeller</i>	2	2.000.000	176,48
10	<i>Gasket Line Flushing</i>	1	100.000	8,82

Ket : warna abu-abu merupakan komponen yang dioptimasi
Kurs US\$ yang digunakan berdasar kurs Bank Mandiri per tanggal 21 April 2014 08:58 WIB, dengan harga Rp 11.333,00.

Lampiran H

Total Biaya Maintenance

Dengan mengacu pada Lampiran D, maka dapat diketahui total biaya *repair* dan *replace* tiap komponen. Biaya man power adalah Rp 30.000,00 atau US\$2,65 per jam. Berikut adalah total *cost repair* dan *replace* komponen.

Komponen	TTR (Hours)	Man Power	Cost Repair Component		Cost Replace Component	
			(Rp)	US\$	(Rp)	US\$
Rotor	8	5	1.200.000	105,88	8.700.000	767,67
<i>Wearing Chasing</i>	8	5	1.200.000	105,88	3.200.000	282,36
<i>Throttle Bushing</i>	8	5	1.200.000	105,88	2.330.000	205,59
<i>Journal Bearing</i>	2	4	240.000	21,18	8.240.000	727,08
<i>Thrust Bearing</i>	2	3	180.000	15,88	2.180.000	192,36
<i>Mechanical Seal</i>	4	5	600.000	52,94	78.667.500	6.941,45
<i>O-ring</i>	0,167	2	10.020	0,88	5.010.020	442,07
<i>Shaft Sleeve</i>	4	5	600.000	52,94	2.600.000	229,42

Total *cost* secara sistem, yaitu total biaya *replacement* semua komponen pada pompa selama kurang lebih 8 jam dengan *man power* sebanyak 5 orang. Perhitungan total *cost* secara sistem adalah sebagai berikut :

$$Cost_{sistem} = (TTR \times man\ power \times cost\ man\ power) + \\ Total\ cost\ replacement$$

$$Cost_{sistem} = (8 \times 5 \times Rp\ 30.000,00) + (Rp\ 105.707.500,00) \\ = Rp\ 1.200.000,00 + Rp\ 105.707.500,00 \\ = Rp\ 106.907.500,00 \text{ atau } US\$9.433,29$$

Lampiran I

Particle Swarm Optimization

I1. Inisialisasi Objective Function

```
function [Realibility, Cost, fitness] = baru(t)

R01=1;
R02=1;
R03=1;
R04=1;
R05=1;
R06=1;
R07=1;
R08=1;
iter=1;
Cost=[];
Realibility=[];

for t = t:t:1000;
    ... Rotor Component
    R1 = (R01*(0.000066+(0.00074*R01)))./((0.00074*R01)+(0.000066*exp...
        ((0.000066+(0.00074*R01)).*t)));
    Re(iter)=R1;
```

```

if (R1 >= 0.9 && R1 <= 1)
R01=R1;
C1 = 0;
pm1 = 0;
elseif (R1 > 0.7 && R1 <= 0.9)
R01=R1+0.07;
C1 = 0.011224657;
pm1 = 1;
else
R01 = 1;
C1 = 0.081378762;
pm1 = 2;
end
... Wearing Chasing Component
R2 = (R02*(0.000052+(0.0002*R02)))./((0.0002*R02)+(0.000052*exp...
((0.000052+(0.0002*R02)).*t)));
Re(iter)=R2;
if (R2 > 0.9 && R2 <= 1)
R02=R2;
C2 = 0;
pm2 = 0;
elseif (R2 > 0.7 && R2 <= 0.9)
R02=R2+0.07;
C2 = 0.011224657;
pm2 = 1;

```

```

else
    R02 = 1;
    C2 = 0.029932418;
    pm2 = 2;
end
... Throttle Bushing Component
R3 = (R03*(0.0000671+(0.00047*R03)))./((0.00047*R03)+(0.0000671*exp...
    ((0.0000671+(0.00047*R03)).*t)));
Re(iter)=R3;
if (R3 > 0.9 && R3 <= 1)
    R03=R3;
    C3 = 0;
    pm3 = 0;
elseif (R3 > 0.7 && R3 <= 0.9)
    R03=R3+0.07;
    C3 = 0.011224657;
    pm3 = 1;
else
    R03 = 1;
    C3 = 0.021794542;
    pm3 = 2;
end
... Journal Bearing Component
R4 = (R04*(0.0000669+(0.00074*R04)))./((0.00074*R04)+(0.0000669*exp...
    ((0.0000669+(0.00074*R04)).*t)));

```

```

Re(iter)=R4;
if (R4 > 0.9 && R4 <= 1)
    R04=R4;
    C4 = 0;
    pm4 = 0;
elseif (R4 > 0.7 && R4 <= 0.9)
    R04=R4+0.07;
    C4 = 0.002244931;
    pm4 = 1;
else
    R04=1;
    C4 = 0.077075977;
    pm4 = 2;
end
... Thrust Bearing Component
R5 = (R05*(0.000218+(0.00042*R05)))./((0.00042*R05)+(0.000218*exp...
    ((0.000218+(0.00042*R05)).*t)));
Re(iter)=R5;
if (R5 > 0.9 && R5 <= 1)
    R05=R5;
    C5 = 0;
    pm5 = 0;
elseif (R5 > 0.7 && R5 <= 0.9)
    R05=R5+0.07;
    C5 = 0.001683699;

```

```

        pm5 = 1;
else
    R05=1;
    C5 = 0.02039146;
    pm5 = 2;
end
... Mechanical Seal Component
R6 = (R06*(0.0035152+((1*10^(-9))*R06)))./(((1*10^(-9)*R06) +
    (0.0035152*exp((0.0035152+((1*10^(-9))*R06)).*t)) ));
Re(iter)=R6;
if (R6 > 0.9 && R6 <= 1)
    R06=R6;
    C6 = 0;
    pm6 = 0;
elseif (R6 > 0.7 && R6 <= 0.9)
    R06=R6+0.07;
    C6 = 0.005612328;
    pm6 = 1;
else
    R06=1;
    C6 = 0.735939948;
    pm6 = 2;
end
... O-Ring Component
R7 = (R07*(0.0001738+(0.000083*R07)))./((0.000083*R07)+(0.0001738*exp...

```

```

        ((0.0001738+(0.000083*R07)).*t)));
Re(iter)=R7;
if (R7 > 0.9 && R7 <= 1)
    R07=R7;
    C7 = 0;
    pm7 = 0;
elseif (R7 > 0.7 && R7 <= 0.9)
    R07=R7+0.07;
    C7 = 0.000093726;
    pm7 = 1;
else
    R07=1;
    C7 = 0.046863129;
    pm7 = 2;
end
... Shaft Sleeve Component
R8 = (R08*(0.0000888+(0.00026*R08)))./((0.00026*R08)+(0.0000888*exp...
        ((0.0000888+(0.00026*R08)).*t)));
Re(iter)=R8;
if (R8 > 0.9 && R8 <= 1)
    R08=R8;
    C8 = 0;
    pm8 = 0;
elseif (R8 > 0.7 && R8 <= 0.9)
    R08=R8+0.07;

```

```

C8    = 0.005612328;
pm8   = 1;
else
    R08=1;
    C8   = 0.02432009;
    pm8   = 2;
end

CC1(iter)=C1
RR1(iter)=R1;
PM1(iter)=pml;
iter=iter+1;
end

Realibility = R1*R2*R3*R4*R5*R6*R7*R8;
Cost   = C1+C2+C3+C4+C5+C6+C7+C8;
fitness = (Realibility-Cost);

```

12. Running Program

```

clear all;
close all;
clc;

```

```

% Inisialisasi
swarmsize = 20; %jumlah partikel
dim = 1;          %dimensi masalah
maxit = 1000;     %iterasi maksimum
c1 = 2;           %konstanta c1
c2 = 2;           %konstanta c2
maxR = 1;
minR = 0.6;
maxC = 1.037696326;
minC = 0.000093726;
max_pm = 0;
min_pm = 2;
min_t = 50;
max_t = 1000;

% Inisialisasi swarm and velocity
swarm = [(max_t-min_t)*rand(swarmsize,dim) + min_t]; %Waktu

vswarm = rand(swarmsize,dim);
[baris,kolom] = size(swarm);

for i = 1:baris
    [Realibility(i,:),Cost(i,:),fitness(i,:)] = baru(swarm(i,:));
end

```

```

iter = 1;
% Inisialisasi Pbest tiap parkitel
pbest = swarm;
bestfitness = fitness;

% Menemukan partikel terbaik pada inisialisasi populasi
[maxfitness,indx] = max(fitness);
gbest = swarm(indx,:);
lastbest = [];
maxftot = [];

%%%%% ----- START ITERASI ----- %%%%
while iter < maxit

% Update Velocity
max_theta = 0.9; %maksimum theta
min_theta = 0.4; %minimum theta
theta = max_theta - ((max_theta-min_theta)/maxit)*iter;
r1 = rand; %konstanta r1
r2 = rand; %konstanta r2
for c = 1:baris
    vswarm(c,:) = (theta.*vswarm(c,:)) + c1*r1.* (pbest(c,:)-...
                  swarm(c,:)) + c2*r2.* (gbest - swarm(c,:));
end

```

```

% Update posisi partikel
swarm(c,:) = swarm(c,:) + vswarm(c,:);
while swarm(c,:)<min_t || swarm(c,:)>max_t
    swarm(c,:)=(max_t-min_t)*rand(1,1)+min_t;
end
% Evaluasi Fungsi Tujuan
[Realibility(c,:),Cost(c,:),fitness(c,:)] = baru(swarm(c,:));
end

changerow = fitness < bestfitness;
bestfitness = bestfitness.* (1-changerow) + fitness.* changerow;
pbest(find(changerow),:) = swarm(find(changerow),:);
[maxfitness,indx] = min(fitness);
maxftot = [maxftot;maxfitness];
gbest = pbest(indx,:);

if sum(var(pbest)) < 1e-8
    break
end

bbest(iter)=gbest;
lastbest = gbest;
iter = iter + 1;

```

```
bbest(iter)=gbest;
lastbest = gbest;
end
```

```
xopt = gbest;
fmax = maxfitness;
```

```
figure
plot(bbest)
```