TUGAS AKHIR – TM 184835

APLIKASI PENGGUNAAN METODE OPTIMASI METAHEURISTIK UNTUK *DIESEL ENGINE* BERBASIS KECERDASAN BUATAN

GOZZY BASTIAN ANDREA
NRP 02111540000088

Dosen Pembimbing
M. Khoirul Effendi, ST, MSc.Eng.

DEPARTEMEN TEKNIK MESIN
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2019

TUGAS AKHIR – TM 184835

APLIKASI PENGGUNAAN METODE OPTIMASI METAHEURISTIK UNTUK *DIESEL ENGINE* BERBASIS KECERDASAN BUATAN

GOZZY BASTIAN ANDREA
NRP 02111540000088

Dosen Pembimbing
M. Khoirul Effendi, ST, MSc.Eng,

DEPARTEMEN TEKNIK MESIN
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2019

FINAL PROJECT – TM 184835

APPLICATION FOR THE USE OF METAHEURISTIC OPTIMIZATION METHODS FOR DIESEL ENGINE BASED ON ARTIFICIAL INTELLIGENCE

GOZZY BASTIAN ANDREA
NRP 02111540000088

Supervisor
M. Khoirul Effendi, ST, MSc.Eng.

MECHANICAL ENGINEERING DEPARTMENT
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2019

APLIKASI PENGGUNAAN METODE OPTIMASI METAHEURISTIK UNTUK *DIESEL ENGINE* BERBASIS KECERDASAN BUATAN

TUGAS AKHIR

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

Oleh:
Gozzy Bastian Andrea
NRP. 02111540000088

Disetujui oleh Tim Penguji Tugas Akhir:

1. M. Khoirul Effendi, ST, MSc.Eng. .... (Pembimbing)
   NIP. 198204142010121001
2. Achmad Syaifudin, ST, M.Eng, PhD. .... (Penguji I)
   NIP. 197909262005011001
3. Ir. Julendra B. Ariatedja, M.T. .... (Penguji II)
   NIP. 196807061999031004

DEPARTEMEN TEKNIK MESIN

SURABAYA

JULI, 2019

APLIKASI PENGGUNAAN METODE OPTIMASI METAHEURISTIK UNTUK *DIESEL ENGINE* BERBASIS KECERDASAN BUATAN

Nama Mahasiswa : Gozzy Bastian Andrea
NRP : 02111540000088
Departemen : Teknik Mesin FTI-ITS
Dosen Pembimbing : Mohammad Khoirul Effendi, S.T., M.Sc.Eng

ABSTRAK

Pencemaran udara merupakan permasalahan yang sangat serius di Indonesia. Berdasarkan data yang dilansir dari organisasi lingkungan Greenpeace pada tahun 2018, dua kota di Indonesia masuk dalam kategori kota dengan polusi tertinggi (yaitu, Jakarta (peringkat pertama) dan Denpasar (peringkat kelima)). Oleh karena itu perlu dilakukan upaya untuk mengatasi masalah tersebut salah satunya adalah dengan cara mengurangi emisi gas berbahaya, yang sebagian besar dihasilkan dari mesin dengan bahan bakar fosil. Penelitian ini berfokus pada optimasi mesin diesel CAT 3401 dengan data yang didapat dari *paper* dengan judul "*Parametric Study and Optimization Using RSM of DI Diesel Engine for Lower Emissions*" (2016) oleh Prabhakara Rao Ganji dkk. Penelitian dilakukan dengan menggunakan metode kecerdasan buatan yaitu *Backpropagation Neural Network* (BPNN) sebagai alat untuk memprediksi respon (*peakpressure*, kadar NOx, dan kadar *soot*) dengan parameter input (*Compression Ratio*, *Start of Injection Angle*, *Fuel Injection Pressure*, dan *Exhaust Gas Recirculation*). Prediksi tersebut kemudian digunakan untuk mendapatkan parameter input dengan hasil respon yang optimal dengan menggunakan metode metaheuristik yaitu *Genetic Algorithm* (GA) dan *Particle Swarm Optimization* (PSO) yang berkorelasi dalam mengurangi emisi gas berbahaya pada mesin diesel.

Hasil optimum yang didapat dari paper adalah : CR 14,25 , FIP 1153,15 bar, SOI 13,69 ˚bTDC, dan 16,91% EGR, didapatkan output *peakpressure* sebesar 113,5 bar, NOx 15,8 g / kg bahan bakar, soot 0,2 g / kg bahan bakar, dan *composite desirability* 0,95. Sedangkan hasil optimum yang didapat dari BPNN-PSO adalah : CR 16,5 FIP 1400 bar, SOI 6,376 ˚bTDC, EGR 25%, didapatkan output *peakpressure* sebesar 113,473 bar, NOx 12,08 g / kg bahan bakar, soot 01996 g / kg bahan bakar, dan *composite desirability* 0,962 dan hasil optimum yang didapatkan dari BPNN-GA adalah : CR 15,9, FIP 1396,48 bar, SOI 8,471˚bTDC, dan EGR 25%, didapatkan output *peakpressure* sebesar 113,488 bar, NOx 11,89 g / kg bahan bakar, *soot* 0,1787 g / kg bahan bakar dan *composite desirability* 0,964. Dari hasil tersebut didapat bahwa metode BPNN-GA lebih baik dibandingkan dengan metode BPNN-PSO, sehingga metode GA dipilih sebagai metode untuk mengurangi emisi gas berbahaya.

Kata kunci : *Direct Injection Diesel Engine,* Kecerdasan Buatan, *Back Propagation Neural Network,* Metaheuristik, *Genetic Algorithm, Particle Swarm Optimization, CompositeDesirabilit*

APPLICATION FOR THE USE OF METAHEURISTIC OPTIMIZATION METHODS FOR DIESEL ENGINE BASED ON ARTIFICIAL INTELLIGENCE

Student Name	**: Gozzy Bastian Andrea**
NRP	**: 02111540000088**
Departmen	**: Mechanical Engineering FTI-ITS**
Supervisor	**: Mohammad Khoirul Effendi, S.T., M.Sc.Eng**

ABSTRACT

Air pollution is one of serious problems in Indonesia. Based on released data which is reported by Greenpeace’s environmental organization in 2018, two cities from Indonesia are categorized as cities with highest pollution level (i.e., Jakarta (first rank) and Denpasar (fifth rank)). Therefore, a solution should be overtaken to solve this problem for reducing harmful emission gas, which is mostly produced from fuel engine. This study focuses on optimization of the CAT 3401 diesel engine where the data input is obtained from the paper entitled "Parametric Study and Optimization Using Response Surface Method (RSM) of DI Diesel Engines for Lower Emissions" (2016) by Prabhakara Rao Ganji *et al*. The research was conducted using the Artificial Intelligence Backpropagation Neural Network method to predict the correlation of diesel engines’s responses (*i.e.*, peak pressure, NOx levels, and soot levels) with parameter inputs (*i.e.*, Compression Ratio, Start of Injection Angle, Fuel Injection Pressure, and Exhaust Gas Recirculation). The prediction results are then used to find the parameter inputs with the optimum response using metaheuristic Genetic Algorithm (GA) method and Particle Swarm

Optimization (PSO) which correlated in reducing harmful emission of diesel engine.

The optimum result form the reference paper are: CR 14,25 , FIP 1153,15 bar, SOI 13,69 ˚bTDC, and 16,91% EGR, obtained peak pressure output of 113,5 bar, NOx 15,8 g / kg of fuel, soot 0,2 g / kg of fuel, and composite desirability 0,95. Furthermore the optimum result of BPNN-PSO are : CR 16,5 FIP 1400 bar, SOI 6,376 ˚bTDC, EGR 25%, obtained peak pressure output of 113,473 bar, NOx 12,08 g / kg of fuel, soot 01996 g / kg of fuel, and composite desirability 0,962 and the optimum results of BPNN-GA are : CR 15,9, FIP 1396,48 bar, SOI 8,471˚bTDC, and EGR 25%, obtained peak pressure output of 113,488 bars, NOx 11,89 g / kg of fuel, soot 0,1787 g / kg of fuel and composite desirability of 0,964. Since the performace of GA is better than BPNN-PSO, therefore BPNN-GA is then selected as optimization tools in reducing harmfull emission gas of diesel engine.

Keywords: ***Direct Injection of Diesel Engine, Artificial Intelligence, Back Propagation Neural Network, Metaheuristic, Genetic Algorithm, Particle Swarm Optimization, Composite Desirability***

KATA PENGANTAR

Puji syukur penulis ucapkan kehadirat Allah Yang Maha Kuasa karena atas Rahmat dan Kuasa-Nya, penulis dapat menyelesaikan tugas akhir ini. Tugas akhir ini disusun sebagai syarat kelulusan S1 Teknik Mesin Institut Teknologi Sepuluh Nopember Surabaya.

Penulis juga mengucapkan terimakasih atas segala bantuan dan bimbingan yang telah diberikan kepada penulis selama penyusunan Karya Tulis ini sehingga dapat diselesaikan dengan baik. Secara khusus, penulis mengucapkan terimakasih kepada :

1. Orang tua dan saudara penulis yang selalu memberikan do'a, dukungan dan kasih sayang.
2. Bapak Mohammad Khoirul Effendi, S.T., M.Sc.Eng selaku dosen pembimbing.
3. Segenap Keluarga Besar Teknik Mesin ITS.

Penulis mengharapkan agar Tugas Akhir ini dapat bermanfaat bagi seluruh pembaca dan penelitian berikutnya. Kritik dan saran yang membangun sangat penulis harpkan sehingga Tugas Akhir ini bisa menjadi lebih baik.

Surabaya, Juni 2019

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

ABSTRAK i
ABSTRACT iii
KATA PENGANTAR.. v
DAFTAR ISI vi
DAFTAR GAMBAR x
DAFTAR TABEL xiii
BAB 1PENDAHULUAN... 1
1.1 Latar Belakang 1
1.2 Perumusan Masalah... 5
1.3 Batasan Masalah... 6
1.4 Tujuan Penelitian... 6
1.5 Manfaat Penelitian... 6
BAB 2DASAR TEORI 9
2.1 Motor Diesel.. 9
2.1.1 Siklus Diesel.. 9
2.1.2 Parameter Diesel 11
2.2 Artificial Neural Network (ANN) 13
2.2.1 Prinsip Dasar ANN... 15
2.2.2 Arsitektur Jaringan 16
2.2.3 Fungsi Aktivasi 18
2.2.4 Bias 18
2.2.5 Pelatihan Jaringan Syaraf Tiruan.. 19
2.2.6 Backpropagation 19
2.2.7 Aplikasi ANN... 20
2.2.8 Contoh Aplikasi Backpropagation Neural Network (BPNN) 21
2.3 Genetik Algoritma... 27
2.3.1 Kromosom 27
2.3.2 Fitness.. 28

2.3.3 Elitisme 28
2.3.4 Seleksi dengan *Roulette-wheel* 28
2.3.5 Crossover 29
2.3.6 Mutasi 30
2.3.7 Algoritma GA 30
2.3.8 Contoh Aplikasi GA 34
2.4 Particle Swarm Optimization (PSO) 39
2.4.1 Parameter PSO 39
2.4.2 Algoritma PSO 40
2.4.3 Contoh Aplikasi PSO 42
2.5 Desirability Function 45
2.6 Kajian Pustaka 46
2.6.1 Parametric Study and Optimizing Using RSM of DI Diesel Engine for Lower Emissions (Prabhakara, 2016) ... 46
2.6.2 Evaluation of Artificial Neural Network Performance in Predicting Diesel Engine NO_x Emissions (Obodeh, 2009) 47
2.6.3 Genetic Algorithms Optimization of Diesel Engine Emissions and Fuel Efficiency with Air Swirl, EGR, Injection Timing and Multiple Injections (Hiro, 2003)48
2.6.4 Diesel Engine Emissions Reduction Using Particle Swarm Optimization (Prasanth, 2010) 49
BAB 3METODOLOGI PENELITIAN 51
3.1 Diagram Alir Penelitian 51
3.1.1 Studi Literatur 52
3.1.2 Data 52
3.1.3 Pemodelan BPNN 53
3.1.4 Mencari Nilai Maksimum dan Minimum ... 55
3.1.5 Optimasi dengan Metode GA dan PSO 58
3.1.6 Komparasi Hasil 62
3.1.7 Penarikan Kesimpulan dan Saran 62
BAB 4ANALISA DATA DAN PEMBAHASAN 63

4.1 Pengaruh Setiap Parameter Terhadap Fungsi Tujuan63
4.2 Pemodelan BPNN.. 64
4.3 Nilai Minumum dan Maksimum 68
4.4 Hasil Optimasi dengan metode GA dan PSO........... 79
BAB 5KESIMPULAN DAN SARAN..................................... 83
5.1 Kesimpulan... 83
5.2 Saran... 83
DAFTAR PUSTAKA.. 85
LAMPIRAN... 88
BIODATA PENULIS... 108

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 2.1 Diagram P-v siklus diesel (Moran,2006)................ 9
Gambar 2.2 Siklus motor diesel 4 langkah (Agrawal,2003) 11
Gambar 2.3 Compression Ratio (https://www.mechanicalbooster.com/2017/12/what-is-compression-ratio-petrol-diesel-engine.html) 11
Gambar 2.4 Start of Injection (Agrawal,2003)......................... 12
Gambar 2.5 Exhaust Gas Recirculation (https://www.celtictuning.fi/palvelut/egr-poisto-ohitus) 13
Gambar 2.6 Anatomi neuron (http://psychmuseum.uwgb.org/wp-content/uploads/2016/10/neuron_anatomy.jpg) 14
Gambar 2.7 model jaringan syaraf tiruan (Kusumadewi, 2004)16
Gambar 2.8 *Single Layer Network* (http://neuroph.sourceforge.net/tutorials/Perceptron.html) 17
Gambar 2.9 Multi Layer Network (http://www.swarthmore.edu/NatSci/mzucker1/opencv-2.4.10-docs/modules/ml/doc/neural_networks.html) 17
Gambar 2.10 Macam-macam fungsi aktivasi (Primartha, 2018)18
Gambar 2.11 Contoh jaringan BPNN...................................... 21
Gambar 2.12 Gen, kromosom, dan populasi 27
Gambar 2.13 *Roulette-wheel* (https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_parent_selection.htm) 29
Gambar 2.14 Diagram alir GA 32
Gambar 2.15 Diagram alir PSO 41
Gambar 2.16 Skema penelitian 49
Gambar 3.1 Diagram alir tahapan penelitian............................ 51
Gambar 3.2 Diagram alir pemodelan dengan BPNN 54
Gambar 3.3 Diagram alir pencarian nilai minimum dan maksimum dengan metode GA 57
Gambar 3.4 Diagram alir pencarian nilai minimum dan maksimum dengan metode PSO.. 58

Gambar 3.5 Diagram alir optimasi dengan metode GA 60
Gambar 3.6 Diagram alir optimasi dengan metode PSO.......... 61
Gambar 4.1 Perbandingan nilai maksimum *peakpressure* GA dengan PSO 70
Gambar 4.2 Perbandingan nilai minimum peakpressure GA dengan PSO 72
Gambar 4.3 Perbandingan nilai maksimum NOx GA dengan PSO 73
Gambar 4.4 Perbandingan nilai minimum NOx GA dengan PSO 75
Gambar 4.5 Perbandingan nilai maksimum soot GA dengan PSO 76
Gambar 4.6 Perbandingan nilai minimum soot GA dengan PSO78
Gambar 4.7 Perbandingan nilai composite desirability GA dengan PSO... 80

DAFTAR TABEL

Tabel 1.1 Perbandingan macam-macam metode metaheuristik . 3
Tabel 2.1 Perbandingan hasil simulasi CONVERGE dengan metode regresi 47
Tabel 3.1 Data 52
Tabel 4.1 Perbandingan nilai output dengan nilai hasil prediksi peakpressure 64
Tabel 4.2 Perbandingan nilai output dengan nilai hasil prediksi NOx 66
Tabel 4.3 Perbandingan nilai output dengan nilai hasil prediksi soot 67
Tabel 4.4 Perbandingan optimasi dengan metode GA, PSO,dan RSM 81

Halaman ini sengaja dikosongkan

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Pencemaran udara adalah keadaan dimana udara terkontaminasi oleh zat-zat tertentu yang dapat membahayakan kesehatan manusia dan mahluk hidup lainnya. Organisasi lingkungan Greenpeace pada tahun 2018 merilis data 10 besar tingkat polusi udara di dunia, dimana kota Jakarta menduduki peringkat 1 dunia dan kota Denpasar menduduki peringkat 5 dunia. Penyebab utama tingginya polusi udara di kota Jakarta adalah debu akibat pembangunan infrastruktur dan emisi gas buang kendaraan, sedangkan penyebab utama tingginya polusi udara di kota Denpasar adalah emisi gas buang kendaraan dan kegiatan industri. Pemerintah Indonesia melakukan beberapa upaya untuk mengurangi polusi udara, salah satunya yaitu dengan menerapkan standar emisi Euro 4 untuk mengurangi emisi gas buang kendaraan. Berdasarkan standar tersebut, kandungan Nitrogen Oksida (NO_x) pada kendaraan berbahan bakar bensin tidak boleh lebih dari 80 mg/km sedangkan untuk mesin diesel tidak boleh lebih dari 250 mg/km. Pada kenyataanya, NO_x yang dihasilkan mesin diesel lebih besar daripada mesin berbahan bakar bensin, oleh karena itu apabila kandungan NO_x yang dihasilkan oleh mesin diesel dapat diturunkan, maka hal tersebut secara umum akan dapat menurunkan tingkat pencemaran udara secara signifikan. Metode optimasi yang tepat sangat dibutuhkan untuk dapat mengatasi permasalahan tersebut.

Artificial Neural Network (ANN) merupakan salah satu bagian dari kecerdasan buatan, dimana ANN merupakan representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak

manusia. ANN merupakan sistem yang adaptif, dimana dapat menyesuaikan dirinya untuk dapat menyelesaikan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan tersebut. ANN dapat digunakan untuk memodelkan hubungan yang kompleks antara input dan output untuk menemukan pola-pola pada data. Ada beberapa jenis ANN antara lain adalah *Perceptron, Adaptive Linear Neuron* (ADALINE), *Backpropagation Neural Network* (BPNN), *Radial Basis Function Network* (RBF), *Kohonen Self-Organizing Map,* dan *Hopfield Network.* Setiap jenis ANN memiliki kegunaan masing-masing. *Perceptron,* ADALINE, dan *Kohonen Self-Organizing Map* baik digunakan untuk *clustering* seperti *speech recognition, vector coding,* dan *texture segmentation,* sedangkan BPNN, RBF, dan Hopfield Network dapat digunakan untuk berbagai macam aplikasi yang kompleks dan juga sering digunakan dalam bidang optimasi. BPNN memiliki keunggulan dibandingkan dengan jenis ANN yang lain karena dalam BPNN nilai error yang didapat digunakan kembali untuk memperbaiki setiap *weight* nya, sehingga hasil akhir yang didapat dari jaringan BPNN akan memiliki nilai error yang lebih kecil dibandingkan dengan jenis ANN yang lain.

Metaheuristik merupakan suatu metode untuk mencari solusi yang memadukan interaksi antara prosedur pencarian lokal dan proses pencarian yang mampu keluar dari lokal optimum, dimana pencarian dilakukan di ruang solusi untuk mendapatkan nilai global optimum. ANN bersama dengan metode metaheuristik seperti Genetik Algoritma (GA), *Particle Swarm Optimization* (PSO), *Ant Colony Optimization* (ACO), *Simulated Annealing* (SA), dan *Crossed Entrophy* (CE) saat ini sering digunakan sebagai alat untuk mengoptimasi suatu permasalahan. Masing-masing metode metaheuristik memiliki cirikhas dalam pencarian nilai optimum seperti pada Tabel 1.1.

Tabel 1.1 Perbandingan macam-macam metode metaheuristik

	GA	PSO	ACO	SA	CE
Keluar lokal optimum	√	×	×	√	×
Pencarian menyebar	√	×	×	×	×
Waktu komputasi	lama	cepat	cepat	lama	cepat

Metode GA memiliki keunggulan dibandingkan dengan metode yang lain, dimana GA memiliki kemampuan untuk keluar dari jebakan *local optimum* dan mampu mencapai *global optimum*, namun disisi lain GA memerlukan waktu yang cukup lama untuk mencapai nilai *global optimum* tersebut. PSO memiliki keunggulan dibandingkan dengan metode yang lain dalam hal waktu komputasinya yang cepat, namun memiliki kekurangan yaitu tidak mampu keluar dari jebakan *local optimum*. Untuk memperkecil kemungkinan nilai yang didapat *local optimum*, dapat dilakukan diantaranya dengan cara membatasi kecepatan maksimum dari partikel, namun hal tersebut masih memungkinkan PSO terjebak dalam *local optimum*.

Penelitian mengenai optimasi mesin diesel telah dilakukan oleh beberapa peneliti antara lain Prabhakara Rao Ganji dkk (2016) melakukan penelitian dengan menggunakan software CONVERGE CFD untuk simulasi pembakaran pada mesin diesel CAT 3401 lalu dioptimasi menggunakan metode *Response Surface Methodology* (RSM). Penelitian tersebut bertujuan untuk mencari nilai optimum dari *Compression Ratio, Start of Injection Angle, Fuel Injection Pressure,* dan *Exhaust Gas Recirculation* untuk mendapatkan respon berupa *peak pressure* tinggi, namun dengan kadar NO_x dan *soot* minimum. O Obodeh dan C.I Ajuwa (2009) meneliti mengenai kapabilitas *Artificial Neural Network* (ANN) untuk memprediksi emisi NO_x dari *multi-cylinder diesel engine*. Penelitian tersebut menggunakan 3 *hidden layer* dengan jumlah 3 sampai 15 neuron. Pada hasil akhir didapat bahwa hasil prediksi menggunakan ANN menghasilkan error sebesar 0,68%

sampai dengan 3,34%, hal tersebut membuktikan bahwa ANN mampu memprediksi emisi NO_x dengan baik. Hiro Hiroyasu dkk (2003) melakukan penelitian dengan menggunakan *software* HIDECS untuk menganalisa emisi dan performa dari mesin diesel lalu dioptimasi menggunakan metode Genetik Algoritma. Penelitian tersebut bertujuan untuk mendapatkan nilai yang optimum dari *multiple injections pattern, start of injection timing,* dan *exhaust gas recirculation* untuk mendapatkan respon berupa *Specific Fuel Consumption* (SFC) dan emisi (NO_x dan *soot*). Parameter dikatakan optimum apabila menghasilkan nilai SFC dan emisi yang rendah. Prashanth K. Karra dan Song-Charng Kong melakukan penelitian tentang mesin diesel dengan fungsi objektif yang didapat dari *literature for engine optimization* (Montgomery, 2000) lalu dilakukan optimasi dengan menggunakan metode PSO. Penelitian tersebut bertujuan untuk mendapatkan nilai yang optimum dari *main* SOI, *pilot* SOI, *pilot fuel quantity*, dan *Exhaust Gas Recirculation* untuk mendapatkan respon berupa *Brake Specific Fuel Consumption* (BSFC) dan emisi (NO_x, HC, CO, dan PM). Parameter dikatakan optimum apabila menghasilkan nilai BSFC dan emisi yang rendah.

Penelitian kali ini mengacu pada penelitian yang telah dilakukan oleh Prabhakara Rao Ganji dkk (2016) dengan judul "*Parametric Study and Optimization Using RSM of DI Diesel Engine for Lower Emissions"*. Pada penelitian tersebut menggunakan metode *Response Surface Methodology*(RSM) untuk menentukan kombinasi parameter *Compression Ratio, Start of Injection Angle, Fuel Injection Pressure,* dan *Exhaust Gas Recirculation* yang optimum pada mesin diesel CAT 3401. Parameter respon yang diamati pada penelitian kali ini adalah *Peak Pressure*, kadar NO_x, dan kadar *soot*. Tujuan dari penelitian tersebut adalah untuk mendapatkan nilai *Peak Pressure* maximum, namun dengan kadar NO_x dan *soot* minimum. Parameter dikatakan paling optimum apabila memiliki nilai *composite desirability* tertinggi. Metode RSM sendiri memiliki beberapa kekurangan antara lain waktu komputasinya yang lama,

dan juga hasil yang didapat memungkinkan terjebak dalam lokal optimum, dikarenakan pada metode RSM hanya digunakan persamaan dengan persamaan orde satu dan orde dua saja dalam memprediksi respon, sehingga sangat memungkinkan terjebak dalam lokal optimum. Dengan digunakannya persamaan order satu dan orde dua saja dalam RSM, maka metode RSM memiliki keterbatasan apabila digunakan untuk memprediksi respon yang memiliki fungsi polinomial orde tinggi, maka dari itu metode *Back Propagation Neural Network* diharapkan mampu memprediksi nilai respon lebih baik dibandingkan dengan metode RSM, dikarenakan metode BPNN mampu memprediksi persamaan polinomial orde tinggi dengan cara mengatur jumlah neuron dan jumlah layernya. Dalam hal pencarian nilai optimum, metode GA dan PSO diharapkan mampu mencapai hasil yang lebih baik dibandingkan dengan metode RSM. GA memiliki kelebihan dibandingkan dengan metode RSM yaitu mampu keluar dari lokal optimum, sedangkan PSO sendiri memiliki kelebihan waktu kalkulasi yang lebih cepat dibandingkan dengan RSM. Meskipun PSO dapat terjebak dalam lokal optimum, membatasi kecepatan partikel dapat mengurangi kemungkinan hasil yang didapat adalah lokal optimum sehingga hasil yang didapat akan lebih baik. Hipotesa awal dari penelitian kali ini adalah GA dan PSO mampu mencapai nilai yang lebih baik dibandingkan dengan metode RSM, dimana GA akan mencapai nilai yang lebih optimum sedangkan PSO akan mencapai nilai optimum lebih cepat daripada GA.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka perumusan masalah pada tugas akhir ini adalah sebagai berikut :

1. Bagaimana pemilihan parameter *Neural Network* yang cocok untuk memperoleh tingkat kesalahan prediksi respon yang paling kecil.
2. Bagaimana menentukan desain parameter yang tepat pada mesin diesel CAT 3401 dengan menggunakan BPNN-GA dan

BPNN-PSO untuk mendapatkan variabel respon *Peak Pressure* maksimum dengan kadar NO_x dan *soot* minimum.

1.3 Batasan Masalah

Agar dalam proses penyelesaian tugas akhir ini dapat berjalan dengan fokus dan terarah , maka diberlakukan batasan masalah sebagai berikut :

1. *Hidden layer* yang digunakan pada jaringan BPNN adalah 1 sampai 5 *hidden layer* dan tiap *hidden layer* terdiri dari 2 sampai 10 *neuron*, metode learning yang digunakan adalah *Quasi Newton*
2. Pada Genetik Algoritma dilakukan *selection* dengan metode *Roulette-wheel*, *crossover* dengan metode *simple crossover*, dan mutasi dengan metode *inversion of single bits*.
3. Data yang digunakan diambil dari referensi paper yang telah ada dengan judul "*Parametric study and optimization using RSM of DI diesel engine for lower emissions*" oleh Prabhakara Rao Ganji dkk (Ganji, 2016) .

1.4 Tujuan Penelitian

Berdasarkan perumusan masalah yang sudah ditetapkan, maka tugas akhir ini dilakukan dengan tujuan sebagai berikut :

1. Mendapatkan parameter *Neural Network* yang cocok untuk memperoleh tingkat kesalahan prediksi respon yang paling kecil.
2. Menentukan pengaturan variabel desain yang paling tepat untuk mendapatkan variabel respon *Peak Pressure* maksimum dengan kadar NO_x dan *soot* minimum dengan menggunakan metode BPNN-GA dan BPNN-PSO.

1.5 Manfaat Penelitian

Manfaat yang dapat diperoleh dari tugas akhir ini adalah sebagai berikut :

1. Mendapatkan nilai variabel desain mesin diesel CAT 3401 yang paling optimal.

2. Sebagai referensi untuk penelitian yang sejenis dalam rangka pengembangan pengetahuan dibidang optimasi mesin diesel.

Halaman ini tidak sengaja dikosongkan

BAB 2
DASAR TEORI

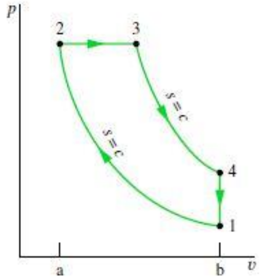
2.1 Motor Diesel

Prinsip kerja motor diesel yaitu mengubah energi kimia (bahan bakar) menjadi energi mekanik. Proses pembakaran antara bahan bakar solar dengan oksigen terjadi pada silinder. Ledakan yang dihasilkan oleh proses pembakaran mampu menggerakkan torak yang dihubungkan dengan poros engkol oleh batang penggerak. Gerak translasi yang terjadi pada torak menyebabkan gerak rotasi pada poros engkol dan sebaliknya gerak rotasi tersebut mengakibatkan gerak bolak-balik torak

Konsep pembakaran pada motor diesel adalah melalui proses kompresi udara pada tekanan tinggi, akibatnya udara akan mempunyai tekanan dan tempereatur melebihi temperatur dan tekanan bakar bahan bakar, sehingga bahan bakar dapat terbakar.

2.1.1 Siklus Diesel

Siklus diesel adalah siklus ideal untuk *compression ignition engine* atau mesin diesel yang mengasumsikan penambahan panas pada tekanan konstan. Siklus diesel dapat digambarkan dalam diagram P-v seperti pada gambar 2.1



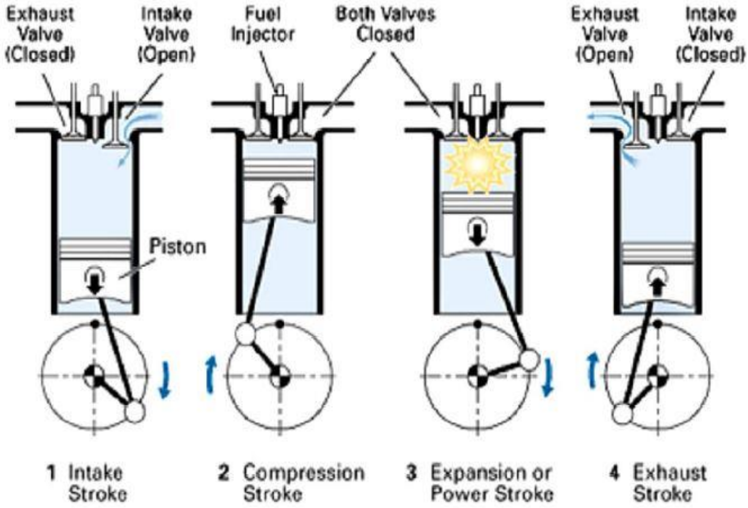
Gambar 2.1 Diagram P-v siklus diesel (Moran,2006)

Proses dari siklus tersebut yaitu :
1-2 = Langkah Kompresi, P bertambah, s = c (isentropik)
2-3 = Pembakaran, pada tekanan tetap (isobarik)

3-4 = Langkah Kerja, P berkurang, s = c (isentropik)
4-1 = Pengeluaran Kalor sisa pada V = c (isokhorik)

Motor diesel empat langkah bekerja bila melakukan empat kali gerakan (dua kali
putaran engkol) menghasilkan satu kali kerja, pada Gambar 2.2 ditunjukkan langkah-langkah yang terjadi pada mesin diesel untuk satu kali kerja. Prinsip kerja motor diesel empat langkah dapat dijelaskan sebagai berikut:

1. Langkah hisap
   Pada langkah ini katup masuk membuka dan katup buang tertutup. Udara
   mengalir ke dalam silinder.
2. Langkah kompresi
   Pada langkah ini kedua katup menutup, piston bergerak dari titik mati bawah (TMB) ke titik mati atas (TMA) menekan udara yang ada dalam silinder. 5^o setelah mencapai TMA, bahan bakar diinjeksikan.
3. Langkah ekspansi
   Karena injeksi bahan bakar kedalam silinder yang bertemperatur tinggi, bahan bakar terbakar dan berekspansi menekan piston untuk melakukan kerja sampai piston mencapai TMB. Kedua katup tertutup pada langkah ini.
4. Langkah buang
   Ketika piston hampir mencapai TMB, katub buang terbuka, katub masuk tetap tertutup. Ketika piston bergerak menuju TMA sisa pembakaran terbuang keluar ruang bakar. Akhir langkah ini adalah ketika piston mencapai TMA. Siklus kemudian berulang lagi.

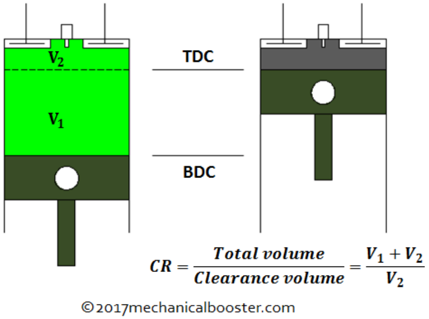


Gambar 2.2 Siklus motor diesel 4 langkah (Agrawal,2003)

2.1.2 Parameter Diesel

Parameter-parameter yang digunakan sebagai acuan untuk melakukan penelitian kali ini antara lain sebagai berikut

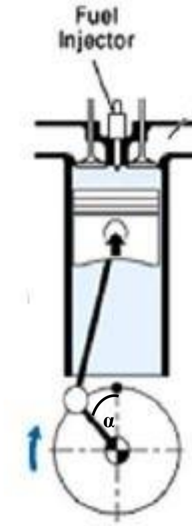
- *Compression Ratio* (CR)



Gambar 2.3 *Compression Ratio*
(https://www.mechanicalbooster.com/2017/12/what-is-compression-ratio-petrol-diesel-engine.html)

Compression Ratio adalah perbandingan volume ruang bakar saat piston berada pada titik mati atas dan pada saat piston berada pada titik mati bawah

- *Start of Injection Angle* (SOI)



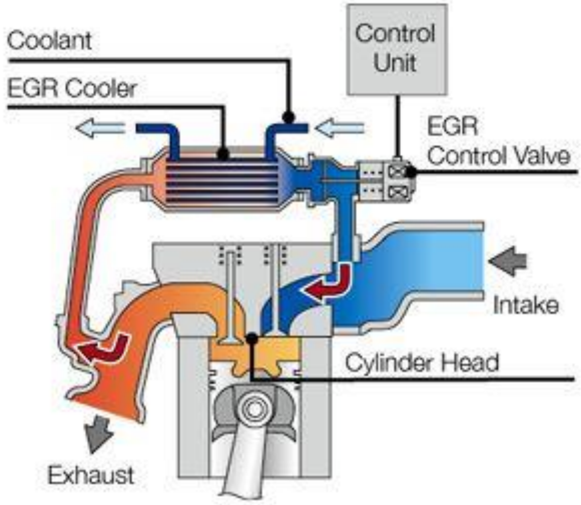
Gambar 2.4 *Start of Injection* (Agrawal,2003)

Start of Injection Angle (α) adalah sudut dimana bahan bakar mulai diinjeksikan

- *Fuel Injection Pressure* (FIP)

  Fuel Injection Pressure adalah tekanan bahan bakar yang diinjeksikan pada ruang bakar

- *Exhaust Gas Recirculation*



Gambar 2.5 *Exhaust Gas Recirculation* (https://www.celtictuning.fi/palvelut/egr-poisto-ohitus)

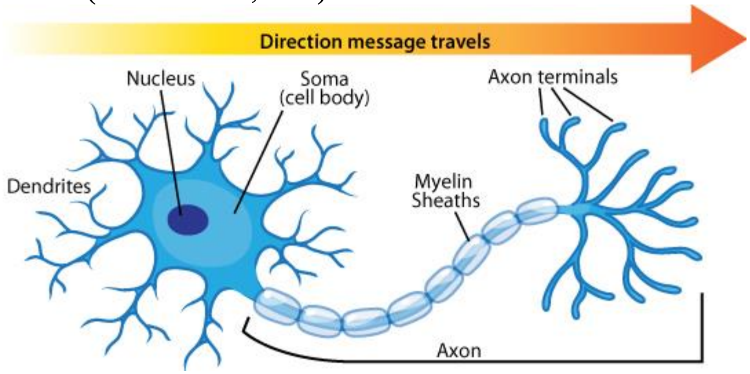
Exhaust Gas Recirculation adalah presentase gas buang yang digunakan kembali untuk proses pembakaran di ruang bakar

- *Peakpressure*
  Peakpressure adalah tekanan maksimum yang dihasilkan di ruang bakar
- Nitrogen Oksida (NO_x)
  Nitrogen oksida adalah gas yang dihasilkan dari pembakaran mesin diesel
- *Soot*
  Soot adalah sisa pembakaran tak sempurna yang berada diruang bakar (jelaga)

2.2 Artificial Neural Network (ANN)

Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses

pembelajaran pada otak manusia. ANN diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran. Ada beberapa tipe ANN antara lain *Back Propagation Neural Network*, *Radial Basis Function Network* (RBF), *Kohonen Self-Organizing Map*, dan *Hopfield Network* dan lain-lain, namun hampir semuanya memiliki komponen komponen yang sama. Seperti halnya otak manusia, ANN juga terdiri dari beberapa neuron, dan ada hubungan antara neuron-neuron tersebut. Neuron-neuron tersebut akan mentransformasikan informasi yang diterima melalui sambungan keluarnya menuju ke neuron-neuron yang lain. Pada *Neural Network*, hubungan ini dikenal dengan nama *weight* (bobot), informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut (Kusumadewi,2004).



Gambar 2.6 Anatomi neuron (http://psychmuseum.uwgb.org/wp-content/uploads/2016/10/neuron_anatomy.jpg)

Neuron memiliki 3 komponen penting yaitu dendrit, soma, dan axon seperti yang ditunjukkan pada Gambar 2.6 . Dendrit berfungsi sebagai input *vector*. Dendrite menerima sinyal dari neuron lainnya. Secara matematika, masing-masing dendrite dapat melakukan operasi pengalian berdasarkan *weight value* dari dendrit. Soma berfungsi sebagai *summation function*. Sinyal positif dan negatif yang datang kepada soma berasal dari dendrit.

Axon menerima sinyal dari soma. Apabila soma mencapai suatu ambang lisrik atau *electrical potential* tertentu maka axon akan meneruskannya ke neuron lainnya (Primartha, 2018).

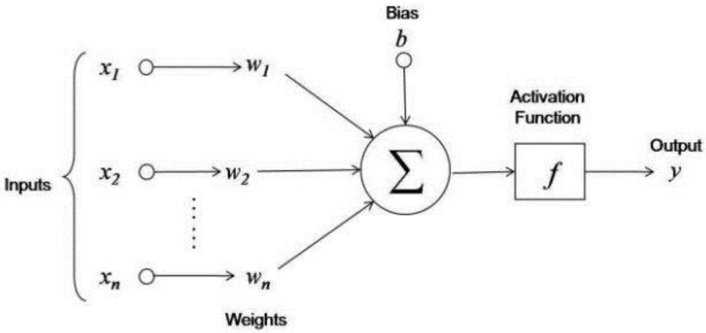
ANN dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi, dengan asumsi bahwa :

- Pemrosesan informasi terjadi pada banyak elemen sederhana(neuron)
- Sinyal dikirim diantara neuron-neuron melalui penghubung-penghubung
- Penghubung antar neuron memiliki bobot yang akan memperkuat atau memperlemah sinyal
- Untuk menentukan output, setiap neuron menggunakan fungsi aktivasi yang dikenakan pada jumlahan input yang diterima. Besarnya output ini selanjutnya dibandingkan dengan suatu batas ambang (Siang, 2009).

ANN ditentukan oleh 3 hal, yaitu pola hubungan antar neuron (arsitektur jaringan), metode untuk menentukan bobot penghubung (metode *training/learning/validation*), dan fungsi aktivasi.

2.2.1 Prinsip Dasar ANN

Pada ANN, neuron-neuron akan dikumpulkan dalam lapisan-lapisan (*layer*) yang disebut dengan lapisan neuron (*neuron layers*). Biasanya neuron-neuron pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya (kecuali lapisan input dan lapisan output). Informasi yang diberikan pada jaringan syaraf akan dirambatkan lapisan ke lapisan, mulai dari lapisan input sampai ke lapisan output melalui lapisan lainnya, yang sering dikenal dengan nama lapisan tersembunyi (*hidden layer*). Tergantung pada algoritma pembelajarnnya, bisa jadi informasi tersebut akan dirambatkan secara mundur pada jaringan.



Gambar 2.7 model jaringan syaraf tiruan (Kusumadewi, 2004)

Gambar 2.7 menunjukkan jaringan syaraf sederhada dengan fungsi aktivasi *f*. Pada gambar tersebut sebuah neuron akan mengolah n input(x_1, x_2,….., x_n) yang masing-masing memiliki bobot w_1,w_2,……,w_n dan bobot bias (b) dengan rumus :

$$a = b + \sum_{i=1}^{n} x_i w_i$$

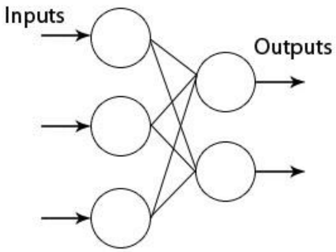
a adalah keluaran unit penjumlahan, kemudian fungsi aktivasi *f* akan mengaktivasi a menjadi output jaringan **y**.

2.2.2 Arsitektur Jaringan

Beberapa arsitektur jaringan yang sering dipakai dalam jaringan syaraf antara lain:

a) *Single Layer Network*

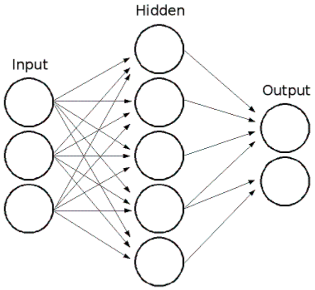
Single layer network adalah jaringan dimana input neuron langsung dihubungkan dengan output neuron, dengan kata lain *single layer network* tidak memiliki *hidden layer* diantara input dan outputnya. Gambar 2.5 adalah gambar *Single layer network* dengan 3 neuron input dan 2 neuron output.



Gambar 2.8 *Single Layer Network*
(http://neuroph.sourceforge.net/tutorials/Perceptron.html)

b) *Multi Layer Network*

Multi Layer Network adalah jaringan dimana input neuron dan output neuron dihubungkan oleh *hidden layer*, seperti yang ditunjukkan pada gambar 2.9



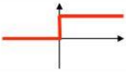
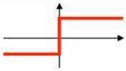
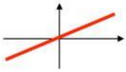
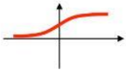
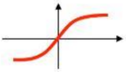
Gambar 2.9 Multi Layer Network
(http://www.swarthmore.edu/NatSci/mzucker1/opencv-2.4.10-docs/modules/ml/doc/neural_networks.html)

c) Jaringan *Reccurent*

Model jaringan *reccurent* mirip dengan jaringan layar tunggal ataupun ganda, hanya saja ada neuron output yang memberikan sinyal pada unit input(sering disebut *feedback loop*) (Siang,2009).

2.2.3 Fungsi Aktivasi

Dalam ANN, fungsi aktivasi dipakai untuk menentukan output dari suatu neuron. Argumen fungsi aktivasi adalah a input (kombinasi linier inputan dan bobotnya). Jika $a = \sum x_i w_i$ maka fungsi aktivasinya adalah $f(a) = f(\sum x_i w_i)$.

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	Multi-layer Neural Networks	

Gambar 2.10 Macam-macam fungsi aktivasi (Primartha, 2018)

Pada Gambar 2.10 ditunjukkan beberapa fungsi aktivasi yang sering digunakan pada ANN.

2.2.4 Bias

Bias adalah sebuah unit masukan yang ditambahkan agar model menjadi lebih fleksibel, khususnya apabila input bernilai 0, tanpa bias maka hanya nilai w yang bisa berubah-ubah. Bias juga bisa menggeser grafik fungsi aktivasi ke kiri atau kanan. Jika melibatkan bias, maka keluaran unit penjumlahan (a) adalah

$$a = b + \sum_{i=1}^{n} x_i w_i$$

dengan b = bias, n = jumlah input, x = input, w = bobot (*weight*).

2.2.5 Pelatihan Jaringan Syaraf Tiruan

Berdasarkan cara memodifikasi bobotnya, ada 2 macam pelatihan yang dikenal yaitu dengan supervisi (*supervised training*) dan tanpa supervisi (*unsupervised training*).

- *Supervised Training*

Pada *Supervised Training*, terdapat sejumlah pasangan data (input-output) yang digunakan untuk melatih jaringan hingga diperoleh output yang mendekati atau sama dengan target. Pada setiap kali pelatihan, suatu input diberikan ke jaringan lalu jaringan akan mengeluarkan output. Selisih antara keluaran jaringan dengan target merupakan kesalahan yang terjadi (*error*), lalu jaringan akan memodifikasi bobot sesuai dengan kesalahan tersebut. Jaringan *percepteron*, ADALINE, dan *backpropagation* merupakan model-model yang menggunakan pelatihan dengan supervisi.

- *Unsupervised Training*

Pada *Unsupervised Training*, perubahan bobot jaringan dilakukan berdasarkan parameter tertentu dan jaringan dimodifikasi menurut ukuran parameter tersebut.

Berdasarkan hasil yang pernah dilaporkan, model pelatihan dengan supervisi lebih banyak digunakan dan terbukti cocok dipakai dalam berbagai aplikasi, akan tetapi kelemahan utama pelatihan dengan supervisi adalah dalam hal pertumbuhan waktu komputasinya yang berorder eksponensial ini berarti untuk data pelatihan yang cukup banyak, prosesnya menjadi sangat lambat .

2.2.6 Backpropagation

Backpropagation merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh percepteron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya (Kusumadewi,2004). Algoritma *backpropagation* menggunakan *error* output juntuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahap perambatan maju (*forward propagation*) harus di kerjakan

terlebih dahulu. Dalam *backpropagation*, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat, yaitu : kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun (Siang, 2009). Beberapa fungsi aktivasi yang sering digunakan dalam *backpropagation* antara lain :

1. Sigmoid

$$y = f(x) = \frac{1}{1 + e^{-\sigma x}}$$

$$f'(x) = \sigma f(x)[1 - f(x)]$$

2. Tansig

$$y = f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

$$f'(x) = [1 + f(x)][1 - f(x)]$$

3. Purelin

$$y = f(x) = x$$

$$f'(x) = 1$$

dimana x merupakan input dari neuron dan y merupakan output setelah neuron dimasukkan ke dalam fungsi aktivasi.

2.2.7 Aplikasi ANN

Beberapa aplikasi ANN adalah sebagai berikut :

a) Pengenalan Pola (*Pattern Recognition*)

ANN dapat dipakai untuk mengenali pola (misal huruf, angka, suara, atau tanda tangan) yang sudah sedikit berubah. Hal ini mirip dengan otak manusia yang masih mampu mengenali orang yang sudah beberapa waktu tidak dijumpainya (mungkin wajah/bentuk tubuhnya sudah sedikit berubah).

b) Signal Processing

ANN (model ADELINE) dapat dipakai untuk menekan *noise* dalam saluran telepon,

c) Peramalan

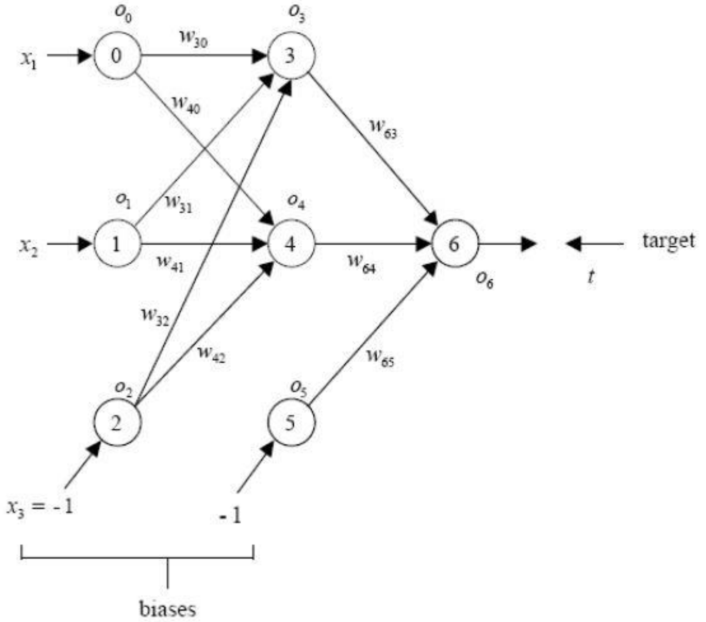
ANN juga dapat dipakai untuk meramalkan apa yang akan terjadi di masa yang akan datang berdasarkan pola kejadian yang ada di masa lampau. Ini dapat dilakukan mengingat

kemampuan ANN untuk mengingat dan membuat generalisasi dari apa yang sudah ada sebelumnya.

Disamping aplikasi-aplikasi tersebut, ANN juga dilaporkan dapat menyelesaikan masalah dalam bidang kontrol, kedokteran, dan lain-lain, meskipun banyak aplikasi menjanjikan yang dapat dilakukan oleh ANN, namun jaringan syaraf tiruan juga memiliki beberapa keterbatasan umum. Pertama adalah ketidak akuratan hasil yang diperoleh. ANN bekerja berdasarkan pola yang terbentuk pada inputnya (Siang,2009).

2.2.8 Contoh Aplikasi Backpropagation Neural Network (BPNN)

Misalkan kita mempunyai sebuah jaringan BPNN seperti pada gambar 2.11



Gambar 2.11 Contoh jaringan BPNN

Dengan nilai input dan target sebagai berikut
$x^{(1)} = (0{,}3\ \ 0{,}4)$, $t^{(1)} = 0{,}88$
$x^{(2)} = (0{,}1\ \ 0{,}6)$, $t^{(2)} = 0{,}82$
$x^{(3)} = (0{,}9\ \ 0{,}4)$, $t^{(3)} = 0{,}57$
bias = -1
Langkah 1 Inisialisasi

- Nilai *weights* diasumsikan 0,2
- Learning rate (η) = 0,2
- E_{max} = 0,01 (*error* maksimum 1%)
- E = 0 (*error* awal)
- k = 1 (*training pattern* awal)

Training Loop – Loop 1
Langkah 2 – Aplikasikan input

- Aplikasikan input pertama pada NN
- $x^{(1)} = (0{,}3\ \ 0{,}4)$, $t^{(1)} = 0{,}88$

$o_0 = x_1 = 0{,}3$
$o_1 = x_2 = 0{,}4$
$o_2 = x_3 = -1$

Langkah 3 – *Forward Propagation*
$o_3 = f(w_{30}o_0 + w_{31}o_1 + w_{32}o_2) = 0{,}4850$
$o_4 = f(w_{40}o_0 + w_{41}o_1 + w_{42}o_2) = 0{,}4850$
$o_5 = -1$
$o_6 = f(w_{63}o_3 + w_{64}o_4 + w_{65}o_5) = 0{,}4985$
Langah 4 – Penghitungan nilai error
$$E = \frac{1}{2}(t - o_6)^2 + E = 0{,}0728$$
$$\delta_6 = f'(tot_6)(t - o_6) = o_6(1 - o_6)(t - o_6) = 0{,}0954$$
Langkah 5 – *Error Backpropagation*
Sebarkan error yang didapat menuju lapisan sebelumnya. Perbarui *weights* dan hitung sinyal error dari lapisan sebelumnya

- Pembaruan *weight* lapisan ke-3

$\Delta w_{63} = \eta\delta_6 o_3 = 0{,}0093$
$w_{63}^{now} = w_{63}^{old} + \Delta w_{63} = 0{,}2093$

$\Delta w_{64} = \eta\delta_6 o_4 = 0{,}0093$
$w_{64}^{now} = w_{64}^{old} + \Delta w_{64} = 0{,}2093$
$\Delta w_{65} = \eta\delta_6 o_5 = -0{,}0191$
$w_{65}^{now} = w_{65}^{old} + \Delta w_{65} = 0{,}1809$

- *Error signals* lapisan ke-2

$$\delta_3 = f_3(tot_3)\sum_{i=6}^{6} w_{i3}\delta_i = o_3(1-o_3)w_{63}\delta_6 = 0{,}0048$$
$$\delta_4 = f_4(tot_4)\sum_{i=6}^{6} w_{i4}\delta_i = o_4(1-o_4)w_{64}\delta_6 = 0{,}0048$$

- Pembaruan weight lapisan ke-2

$\Delta w_{30} = \eta\delta_3 o_0 = 0{,}00028586$
$w_{30}^{now} = w_{30}^{old} + \Delta w_{30} = 0{,}2003$
$\Delta w_{31} = \eta\delta_3 o_1 = 0{,}00038115$
$w_{31}^{now} = w_{31}^{old} + \Delta w_{31} = 0{,}2004$
$\Delta w_{32} = \eta\delta_3 o_2 = -0{,}00095288$
$w_{32}^{now} = w_{32}^{old} + \Delta w_{32} = 0{,}1990$
$\Delta w_{40} = \eta\delta_4 o_0 = 0{,}00028586$
$w_{40}^{now} = w_{40}^{old} + \Delta w_{40} = 0{,}2003$
$\Delta w_{41} = \eta\delta_4 o_1 = 0{,}00038115$
$w_{41}^{now} = w_{41}^{old} + \Delta w_{41} = 0{,}2004$
$\Delta w_{42} = \eta\delta_4 o_2 = -0{,}00095288$
$w_{42}^{now} = w_{42}^{old} + \Delta w_{42} = 0{,}1990$

Training Loop - Loop 2

Langkah 2 – Aplikasikan input

- Aplikasikan input kedua pada NN
- $x^{(2)} = (0{,}1 \;\; 0{,}6)$, $t^{(2)} = 0{,}82$

$o_0 = x_1 = 0{,}1$
$o_1 = x_2 = 0{,}6$
$o_2 = x_3 = -1$

Langkah 3 – *Forward Propagation*

$o_3 = f(w_{30}o_0 + w_{31}o_1 + w_{32}o_2) = 0{,}4853$
$o_4 = f(w_{40}o_0 + w_{41}o_1 + w_{42}o_2) = 0{,}4853$
$o_5 = -1$
$o_6 = f(w_{63}o_3 + w_{64}o_4 + w_{65}o_5) = 0{,}5055$

Langah 4 – Penghitungan nilai error

$$E = \frac{1}{2}(t - o_6)^2 + E = 0{,}1222$$

$$\delta_6 = f'(tot_6)(t - o_6) = o_6(1 - o_6)(t - o_6) = 0{,}0786$$

Langkah 5 – *Error Backpropagation*

Sebarkan error yang didapat menuju lapisan sebelumnya. Perbarui *weights* dan hitung sinyal error dari lapisan sebelumnya

- Pembaruan *weight* lapisan ke-3

$\Delta w_{63} = \eta\delta_6 o_3 = 0{,}0076$
$w_{63}^{now} = w_{63}^{old} + \Delta w_{63} = 0{,}2169$
$\Delta w_{64} = \eta\delta_6 o_4 = 0{,}0076$
$w_{64}^{now} = w_{64}^{old} + \Delta w_{64} = 0{,}2169$
$\Delta w_{65} = \eta\delta_6 o_5 = -0{,}0157$
$w_{65}^{now} = w_{65}^{old} + \Delta w_{65} = 0{,}1652$

- *Error signals* lapisan ke-2

$$\delta_3 = f_3(tot_3)\sum_{i=6}^{6} w_{i3}\delta_i = o_3(1 - o_3)w_{63}\delta_6 = 0{,}0041$$

$$\delta_4 = f_4(tot_4)\sum_{i=6}^{6} w_{i4}\delta_i = o_4(1 - o_4)w_{64}\delta_6 = 0{,}0041$$

- Pembaruan weight lapisan ke-2

$\Delta w_{30} = \eta\delta_3 o_0 = 0{,}00082169$
$w_{30}^{now} = w_{30}^{old} + \Delta w_{30} = 0{,}2004$
$\Delta w_{31} = \eta\delta_3 o_1 = 0{,}00049302$
$w_{31}^{now} = w_{31}^{old} + \Delta w_{31} = 0{,}2009$
$\Delta w_{32} = \eta\delta_3 o_2 = -0{,}000082169$
$w_{32}^{now} = w_{32}^{old} + \Delta w_{32} = 0{,}1982$
$\Delta w_{40} = \eta\delta_4 o_0 = 0{,}000082169$
$w_{40}^{now} = w_{40}^{old} + \Delta w_{40} = 0{,}2004$

$\Delta w_{41} = \eta\delta_4 o_1 = 0{,}00049302$
$w_{41}^{now} = w_{41}^{old} + \Delta w_{41} = 0{,}2009$
$\Delta w_{42} = \eta\delta_4 o_2 = -0{,}00082169$
$w_{42}^{now} = w_{42}^{old} + \Delta w_{42} = 0{,}1982$

Training Loop - *Loop 3*
Langkah 2 – Aplikasikan input

- Aplikasikan input ketiga pada NN
- $x^{(3)} = (0{,}9 \;\; 0{,}4)$, $t^{(3)} = 0{,}57$

$o_0 = x_1 = 0{,}9$
$o_1 = x_2 = 0{,}4$
$o_2 = x_3 = -1$

Langkah 3 – *Forward Propagation*
$o_3 = f(w_{30}o_0 + w_{31}o_1 + w_{32}o_2) = 0{,}5156$
$o_4 = f(w_{40}o_0 + w_{41}o_1 + w_{42}o_2) = 0{,}5156$
$o_5 = -1$
$o_6 = f(w_{63}o_3 + w_{64}o_4 + w_{65}o_5) = 0{,}5146$
Langah 4 – Penghitungan nilai error
$E = \frac{1}{2}(t - o_6)^2 + E = 0{,}1237$
$\delta_6 = f'(tot_6)(t - o_6) = o_6(1 - o_6)(t - o_6) = 0{,}0138$
Langkah 5 – *Error Backpropagation*
Sebarkan error yang didapat menuju lapisan sebelumnya. Perbarui *weights* dan hitung sinyal error dari lapisan sebelumnya

- Pembaruan *weight* lapisan ke-3

$\Delta w_{63} = \eta\delta_6 o_3 = 0{,}0014$
$w_{63}^{now} = w_{63}^{old} + \Delta w_{63} = 0{,}2183$
$\Delta w_{64} = \eta\delta_6 o_4 = 0{,}0014$
$w_{64}^{now} = w_{64}^{old} + \Delta w_{64} = 0{,}2183$
$\Delta w_{65} = \eta\delta_6 o_5 = -0{,}0028$
$w_{65}^{now} = w_{65}^{old} + \Delta w_{65} = 0{,}2183$

- *Error signals* lapisan ke-2

$$\delta_3 = f_3(tot_3)\sum_{i=6}^{6} w_{i3}\delta_i = o_3(1-o_3)w_{63}\delta_6 = 0{,}00074948$$

$$\delta_4 = f_4(tot_4)\sum_{i=6}^{6} w_{i4}\delta_i = o_4(1-o_4)w_{64}\delta_6 = 0{,}00074948$$

- Pembaruan weight lapisan ke-2

$\Delta w_{30} = \eta\delta_3 o_0 = 0{,}00013491$
$w_{30}^{now} = w_{30}^{old} + \Delta w_{30} = 0{,}2005$
$\Delta w_{31} = \eta\delta_3 o_1 = 0{,}000059958$
$w_{31}^{now} = w_{31}^{old} + \Delta w_{31} = 0{,}2009$
$\Delta w_{32} = \eta\delta_3 o_2 = -0{,}00014990$
$w_{32}^{now} = w_{32}^{old} + \Delta w_{32} = 0{,}1981$
$\Delta w_{40} = \eta\delta_4 o_0 = 0{,}00013491$
$w_{40}^{now} = w_{40}^{old} + \Delta w_{40} = 0{,}2005$
$\Delta w_{41} = \eta\delta_4 o_1 = 0{,}000059958$
$w_{41}^{now} = w_{41}^{old} + \Delta w_{41} = 0{,}2009$
$\Delta w_{42} = \eta\delta_4 o_2 = -0{,}00014990$
$w_{42}^{now} = w_{42}^{old} + \Delta w_{42} = 0{,}1981$

Langkah 6 – *One Epoch Looping*

- Cek setiap set data telah di gunakan untuk training, jika belum semua maka kembali ke Langkah 2, jika sudah dilanjutkan ke langkah 7

Langkah 7 – *Total Error Checking*

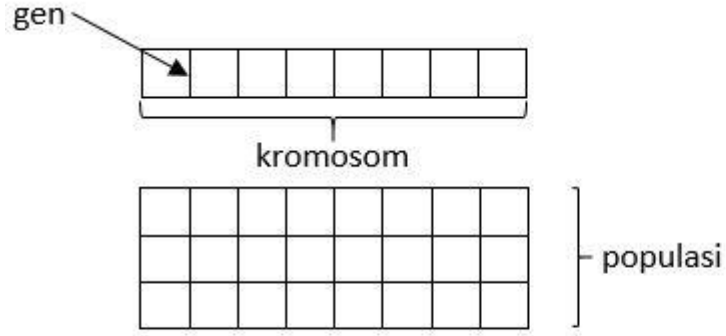
- Kita telah menggunaan 3 pasang data untuk training.
- Cek total error apakah sudah dibawah error maksimum atau belum
    - Jika sudah, akhiri proses training.
    - Jika belum, lakukan training lagi untuk semua data (kembali ke langkah 2)
- Pada kasus ini :

- $E = 0{,}1237$, $E_{max} = 0{,}01$. E masih lebih besar daripada E_{max}, maka lakukan training lagi.

2.3 Genetik Algoritma

Genetik Algoritma (GA) masuk dalam kelompok *Evolutionary Algorithm*. GA didasarkan pada prinsip-prinsip genetika dan seleksi alam. Elemen- elemen dasar dari GA adalah reproduksi, *crossover*, dan mutasi. Algoritma ini banyak dipakai dalam penyelesaian masalah kombinatorial seperti TSP, *crew scheduling*, *airline*, hingga permasalahan kontrol. Dalam *evolution-based approach* biasanya akan dibangkitkan sejumlah populasi awal, lalu dilakukan prosedur tertentu seperti mutasi, seleksi, dan *crossover* akhirnya didapatkan solusi akhir dari problem optimasi yang dihadapi. Dalam GA prosedur pencarian hanya didasarkan pada nilai fungsi tujuan, tidak ada pemakaian *gradient* atau teknik kalkulus (Santosa, 2017) .

2.3.1 Kromosom



Gambar 2.12 Gen, kromosom, dan populasi

Kromosom merupakan bagian penting dari GA. Dalam GA kita akan membangkitkan populasi sebagai kumpulan dari

kromosom, dimana masing-masing kromosom mewakili suatu vektor solusi. Setiap anggota kromosom disusun oleh gen-gen, dimana masing-masing gen mewakili elemen dari vektor solusi, dengan dibangkitkannya populasi ini, maka akan tersedia banyak pilihan solusi.

2.3.2 Fitness

Fungsi *fitness* digunakan untuk mengukur tingkat kesesuaian suatu solusi dengan solusi yang dicari. Fungsi *fitness* bisa berhubungan langsung dengan fungsi tujuan, atau bisa juga sedikit modifikasi terhadap fungsi tujuan. Sejumlah solusi yang dibangkitkan dalam populasi akan dievaluasi menggunakan fungsi *fitness*. Fungsi *fitness* yang biasa digunakan adalah $F(x) = \frac{1}{f(x)+1}$, dimana f(x) adalah fungsi tujuan dari problem yang diselesaikan. Untuk kasus minimasi, jika didapatkan f(x) yang kecil maka nilai *fitness*nya besar. Sebaliknya, untuk kasus maksimasi fungsi *fitness*nya bisa menggunakan nilai f(x) sendiri, jadi $F(x) = f(x)$ (Santosa,2017).

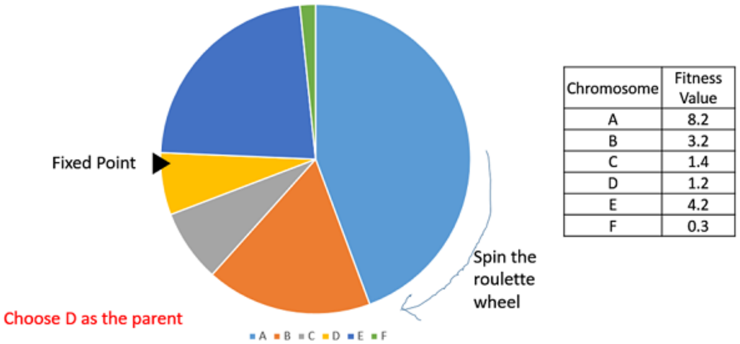
2.3.3 Elitisme

Konsep elitisme dalam GA berarti usaha mempertahankan individu-individu terbaik yang telah diperoleh di suatu generasi ke dalam generasi selanjutnya. Sehingga individu-individu terbaik ini akan tetap muncul di populasi berikutnya. Langkah ini dilakukan dalam berbagai cara, misalnya melalui penyalinan individu terbaik, atau dapat juga melalui kombinasi antara solusi-solusi turunan atau anak dengan induk. Penggunaan operator *elitisme* ini telah terbukti memiliki pengaruh yang sangat penting saat menggunakan GA untuk menyelesaikan masalah persoalan optimasi dengan tujuan tunggal (Santosa,2017).

2.3.4 Seleksi dengan *Roulette-wheel*

Dalam *Roulette-wheel* setiap individu punya kesempatan untuk terpilih. Setiap area dalam lingkaran menunjukkan peluang setiap solusi untuk terpilih. Untuk setiap individu atau solusi akan dibangkitkan bilangan random. Misalkan untuk individu 1, dibangkitkan bilangan random (atau dalam contoh ini bilangan

random dikalikan dengan total *fitness*) berada di area mana dari *Roulette-wheel* tersebut. Misalkan nilai r × tf lebih kecil dari akumulasi *fitness* 1 + *fitness* 2, maka individu 2 yang akan dipilih untuk menggantikan individu 1. Individu dengan *fitness* paling besar akan punya kesempatan terpilih paling besar untuk menjadi induk.



Chromosome	Fitness Value
A	8.2
B	3.2
C	1.4
D	1.2
E	4.2
F	0.3

Gambar 2.13 *Roulette-wheel*
(https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_parent_selection.htm)

2.3.5 Crossover

Istilah *crossover* juga sering disebut kawin silang dilakukan untuk mendapatkan kombinasi yang lebih baik antara satu individu dengan individu yang lain dalam satu populasi. Ada bermacam-macam teknik *crossover* yang bisa digunakan dalam GA. Pada penelitian kali ini teknik *crossover* yang akan digunakan adalah *crossover* dengan memindah sebagian gen induk, ke induk lainnya. Jika ada 2 induk P1 dan P2 dan menghasilkan dua keturunan C1 dan C2 misalkan anggota induk adalah $X=[x_1,x_2,\ldots,x_n]$ dan $Y=[y_1,y_2,\ldots,y_n]$ dan r adalah bilangan random diskrit yang bernilai anatara 1 dan panjang vektor x, maka u dan v mewakili keturunan C1 dan C2, didefinisikan sebagai

$$u_i = \begin{cases} x_i & i < r \\ y_i & sebaliknya \end{cases}$$

$$v_i = \begin{cases} x_i & i < r \\ y_i & sebaliknya \end{cases}$$

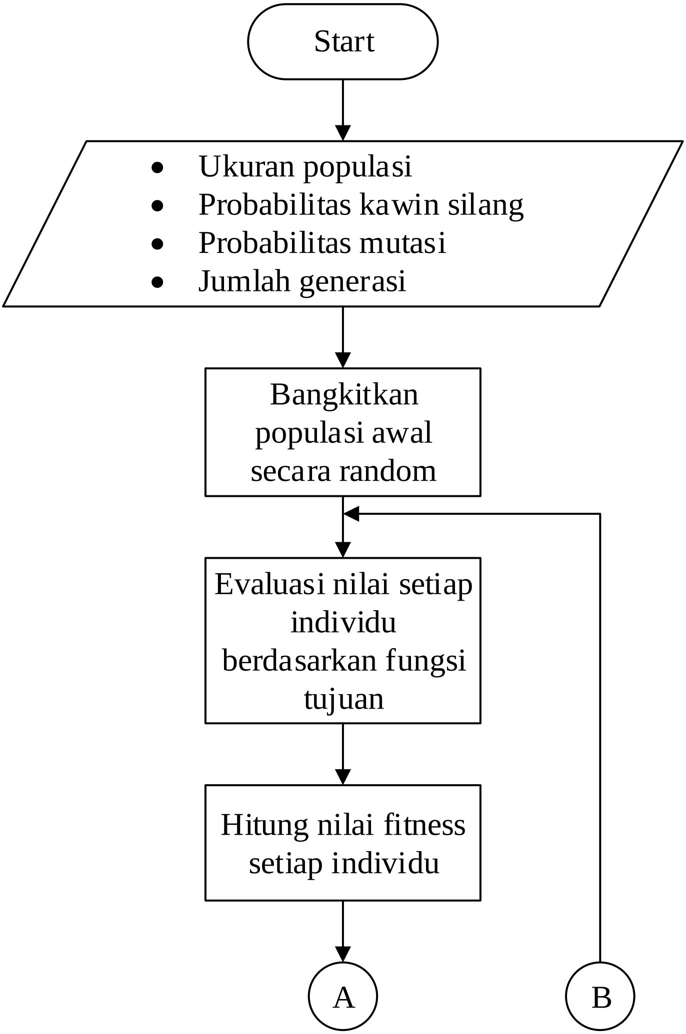
Nilai bilangan random r sering disebut dengan titik potong. Jadi mulai gen keberapa kawin silang akan dilakukan, misalkan dari 8 gen didapat r = 3, maka mulai gen ke 4, gen dari kromosom X akan ditukarkan dengan gen dari kromosom Y. Parameter penting dalam kawin silang adalah probabilitas kawin silang. Jika parameter ini bernilai kecil, maka hanya sedikit kromosom yang aka mengalami kawin silang (Santosa, 2017).

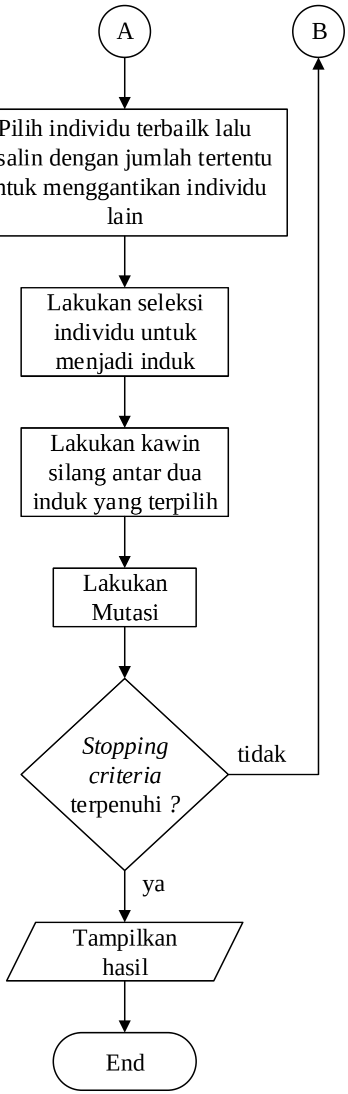
2.3.6 Mutasi

Mutasi dimaksudkan untuk memunculkan individu baru yang berbeda sama sekali dengan individu yang sudah ada. Dalam konteks optimasi memungkinkan munculnya solusi baru untuk bisa keluar dari *local optimum*. Mutasi memunculkan individu-individu baru yang bukan berasal dari hasil kawin silang. Mutasi mengacu pada perubahan urutan atau penggantian elemen dari vektor solusi (pada problem TSP), pemunculan nilai baru (optimasi fungsi). Elemen tersebut juga dipilih secara random. Parameter penting dalam mutasi adalah probabilitas mutasi. Probabilitas ini akan menentukan kromosom mana yang akan mengalami perubahan gen. Semakin besar nilai probabilitas mutasi, semakin banyak kromosom dalam populasi yang akan mengalami mutasi (Santosa,2017).

2.3.7 Algoritma GA

Algoritma GA secara garis besar dapat digambarkan dengan diagram alir seperti pada Gambar 2.10


Start
Ukuran populasi
Probabilitas kawin silang
Probabilitas mutasi
Jumlah generasi
Bangkitkan populasi awal secara random
Evaluasi nilai setiap individu berdasarkan fungsi tujuan
Hitung nilai fitness setiap individu
A
B



Gambar 2.14 Diagram alir GA

Secara garis besar algoritma GA bisa dijelaskan sebagai berikut :

1. Bangkitkan populasi awal
   Tentukan ukuran populasi, probabilitas kawin silang, probabilitas mutasi. Bangkitkan populasi awal atau kromosom-kromosom awal secara random.
2. Evaluasi nilai setiap individu didalam populasi awal ini dengan menggunakan fungsi *fitness*.
3. Pilih salah satu individu terbaik untuk disalin sejumlah tertentu untuk mengganti individu lain (elitisme)
4. Lakukan seleksi individu untuk menjadi induk
   Seleksi dilakukan secara kompetitif untuk memilih 2 anggota populasi sebagai induk.
5. Lakukan kawin siang antar 2 induk yang terpilih
6. Tentukan beberapa individu dalam populasi untuk mengalami proses mutasi
7. Jika belum memenuhi *stopping criteria*, kembali ke langkah 2

Dalam implementasi GA sering digunakan bilangan biner untuk mewakili nilai variabel yang dicari. Umumnya akan ditentukan dulu berapa interval nilai kontinyus yang dicari, sesudah itu akan bisa dicari berapa jumlah bilangan biner yang diperlukan untuk mewakili (berapa jumlah bit). Jika kita punya nilai biner $b_q b_{q-1},\ldots,b_2 b_1 b_0$, dimana b_k=0 atau 1, secara umum untuk mengubah bilangan biner menjadi nilai desimal bisa dilakukan dengan cara

$$y = \sum_{k=0}^{q} 2^k b_k$$

Dengan k = 0,1,2,…,q. Setelah dibangkitkan sejumlah individu dengan nilai biner, maka individu ini bisa dikembalikan nilainya kedalam nilai kontinyus (dekoding) dalam interval batas bawah (BB) dan batas atas (BA) dengan rumus berikut

$$x = BB + \frac{BA - BB}{2^q - 1} \sum_{k=0}^{q} 2^k b_k$$

Jika sebuah variabel x yang batas bawah dan batas atasnya diberikan sebagai BA dan BB akan diwakili dengan menggunakan satu set nilai biner, maka bisa dihitung berapa digit bilangan biner yang diperlukan jika diketahui tingak akurasinya Δx

$$2^q \geq \frac{BA - BB}{\Delta x} + 1$$

(Santosa, 2017)

2.3.8 Contoh Aplikasi GA

Cari nilai minimum dari fungsi berikut :

$$f(x) = ({x_1}^2 + x_2 - 11)^2 + (x_1 + {x_2}^2 - 7)^2$$

Dimana $-5{,}12 \leq x_1 \leq 5{,}12$, $-5{,}12 \leq x_2 \leq 5{,}12$

1. Bangkitkan Populasi

   Misalkan kita bagkitkan populasi berukuran 5, dengan ukuran 16 bit, dimana bit 1-8 merupakan x_1 dan bit 9-16 merupakan x_2. Misalkan diperoleh populasi awal sebagai berikut :

   0 0 1 0 0 1 1 0 0 1 1 0 0 0 0 0
   1 1 0 1 1 1 1 1 0 0 1 0 1 1 0 0
   1 0 0 1 0 0 1 0 1 0 0 1 0 0 0 1
   1 0 0 0 0 0 1 1 0 1 1 1 0 1 1 0
   1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1

   Lakukan proses dekoding untuk masing-masing individu. Contoh dekode untuk individu ke-1

$$x_1 = -5{,}12 + \frac{5{,}12 - (-5{,}12)}{2^7 - 1}(0.2^7 + 0.2^6 + 1.2^5 + 0.2^4 + 0.2^3 + 1.2^2 + 1.2^1 + 0.2^0)$$

$$= -5{,}12 + \frac{10{,}24}{255}(38)$$

$$= -3{,}594$$

$$x_2 = -5{,}12 + \frac{5{,}12-(-5{,}12)}{2^7-1}(0.2^7 + 1.2^6 + 1.2^5 + 0.2^4 + 0.2^3 + 0.2^2 + 0.2^1 + 0.2^0)$$

$$= -5{,}12 + \frac{10{,}24}{255}(96)$$

$$= -1.2649$$

Setelah dilakukan dekoding untuk semua individu, didapat nilai sebagai berikut

x =

-3,5940	-1,2649
3,8350	- 3,3531
0,7429	0,7027
0,1405	-0,3815
2,3893	1,2649

2. Evaluasi tiap individu

Masukkan nilai x terhadap f(x). Didapatkan nilai sebagai berikut

f(x) =

81,3167
65,3834
128,1869
174,1657
25,2735

Untuk melihat nilai fitness gunakan F(x) = 1/f(x). Sehingga dari nilai f(x) diatas diperoleh nilai fitness sebagai berikut

F(x) =

0,0123
0,0153
0,0078
0,0057
0,0396

Dengan nilai maximum 0,0396 dan nilai minimum 0,0057. Nilai maksimum didapat pada x (individu) ke-5.

Nilai maksimum menunjukkan nilai fitness terbaik, dimana nilai x terbaik ini adalah [2,3893 1,2649].

3. Elitisme

Buat populasi sementara untuk menampung individu terbaik hasil seleksi, crossover, dan mutasi. Nilai terbaik didapat pada x ke-5, maka x ke-1 diganti dengan x ke-5, selanjutnya akan didapat populasi sementara sebagai berikut

Populasi sementara =

1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1
1 1 0 1 1 1 1 1 0 0 1 0 1 1 0 0
1 0 0 1 0 0 1 0 1 0 0 1 0 0 0 1
1 0 0 0 0 0 1 1 0 1 1 1 0 1 1 0
1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1

4. Seleksi

Lakukan normalisasi terhadap nilai F(x) dengan cara mengurangi setiap nilai F(x) dengan nilai minimumnya, yaitu 0,0057 sehingga didapatkan

0,0066
0,0096
0,0021
0,0000
0,0339

Jumlah kumulatif dari fitness diatas

cs =

0,0066
0,0162
0,0183
0,0183
0,0522

Jumlah total dari Fitness F adalah 0,0522. Setelah itu proses seleksi bisa dimulai. Bangkitkan bilangan bilangan random, misal r = 0,2893. Jika nilai kumulatif dari F tiap individu dibagi total fitness ≥ bilangan random r, maka

pilih individu yang bersangkutan sebagai induk. Ini bisa dihitung dengan mencari jumlah kumulatif secara bertahap. Jika masing-masing dibagi dengan total fitness akan didapat

csn =

0,1264
0,3102
0,3504
0,3504
1,0000

Masing-masing nilai csn dibandingkan dengan r = 0,2893. Didapat bahwa kumulatif yang ke-2, 0,3102 lebih besar dari r, sehingga yang dipilih sebagai induk 1 (bapak) adalah individu ke-2. Bangkitkan lagi bilangan random,misal r = 0,8120, nilai ini berada pada csn ke-5 yaitu 1, sehingga induk ke-2 (ibu) diambilkan dari individu ke-5.

5. *Crossover* (Kawin silang)

Probabilitas kawin silang ditetapkan, misalnya Ps = 0,8. Jika kita bangkitkan bilangan random dan lebih kecil dari Ps, maka bapak dan ibu terpilih dari individu ke-2 dan ke-5 akan dikawinkan dan keturunannya akan mengganti individu dalam populasi. Jika tidak, maka individu terpilih akan langsung menggantikan induk tanpa kawin silang. Untuk kawin silang disini kita gunakan *simple crossover*. Misal titik potong ditentukan secara random dan ditemukan pada gen 3, maka mulai gen ke-3 akan ditukarsilangkan antara gen pada kromosom bapak dan ibu.

Bapak =

1 1 0 1 1 1 1 1 0 0 1 0 1 1 0 0

Ibu =

1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1

Anak =

1 1 0 | 1 1 0 1 1 1 0 0 1 1 1 1 1

1 0 1 | 1 1 1 1 1 0 0 1 0 1 1 0 0

Populasi sementara akan menjadi

Populasi sementara =

1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1
1 1 0 1 1 0 1 1 1 0 0 1 1 1 1 1
1 0 1 1 1 1 1 1 0 0 1 0 1 1 0 0
1 0 0 0 0 0 1 1 0 1 1 1 0 1 1 0
1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1

Dimana individu ke-2 dan ke-3 digantikan dengan keturunan hasil kawin silang. Jika bilangan random yang dibangkitkan lebih besar dari Ps, maka individu ke-2 dan ke-3 akan digantikan oleh kromosom bapak dan ibu tanpa kawin silang. Ulangi lagi dengan cara yang sama untuk individu ke-4 dan ke-5. Misalkan dari proses pembangkitan bilangan random didapatkan bapak dari individu ke-1 populasi asli dan ibu dari individu ke-5 populasi asli, masing-masing

Bapak =

0 0 1 0 0 1 1 0 0 1 1 0 0 0 0 0

Ibu =

1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1

Lalu dilakukan kawin silang, misalkan secara random didapatkan titik potong terjadi pada gen ke-12, maka anak yang didapat adalah

Anak =

0 0 1 0 0 1 1 0 0 1 1 0 | 1 1 1 1
1 0 1 1 1 0 1 1 1 0 0 1 | 0 0 0 0

Maka populasi sementara menjadi

Populasi sementara =

1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1
1 1 0 1 1 0 1 1 1 0 0 1 1 1 1 1
1 0 1 1 1 1 1 1 0 0 1 0 1 1 0 0
0 0 1 0 0 1 1 0 0 1 1 0 1 1 1 1
1 0 1 1 1 0 1 1 1 0 0 1 0 0 0 0

6. Mutasi

Setelah itu akan dilakukan proses mutasi. Dalam proses ini akan digunakan juga proses stokastik dengan menggunakan bilangan random, lalu dibandingkan dengan probabilitas mutasi yang nilainya ditetapkan cukup kecil, misal 0,05 dengan demikian peluang terjadi mutasi tidak terlalu besar. Untuk setiap gen akan dicek apakah terjadi mutasi atau tidak dengan cara membandingkan bilangan random yang dibangkitkan dengan probabilitas mutasi. Misalnya setelah mutasi didapatkan populasi sementara

Populasi sementara =

1 0 1 1 1 0 1 1 1 0 0 1 1 1 1 1
1 0 0 0 1 0 1 1 1 0 0 1 1 0 1 1
1 0 1 1 1 1 1 1 0 0 1 0 1 1 0 1
0 0 1 0 1 1 1 0 0 1 1 0 1 1 1 1
1 0 1 1 1 0 1 1 1 0 1 1 0 0 0 0

Warna kuning menandakan gen yang mengalami mutasi. Populasi ini yang akan digunakan dala iterasi berikutnya.

2.4 Particle Swarm Optimization (PSO)

PSO didasarkan pada perilaku sebuah kawanan burung atau ikan. Algoritma PSO meniru perilaku sosial organisme ini. Perilaku sosial terdiri dari tindakan individu dan pengaruh dari individu-individu lain dalam suatu kelompok. Kata partikel menunjukkan seekor burung dalam kawanan burung. Setiap individu atau partikel berperilaku dengan cara menggunakan kecerdasannya sendiri dan juga dipengaruhi perilaku kelompok kolektifnya, dengan demikian jika satu partikel atau atau seekor burung menemukan jalan yang tepat atau pendek menuju ke sumber makanan, sisa kelompok yang lain juga akan dapat segera mengikuti jalan tersebut meskipun lokasi mereka jauh di kelompok tersebut (santosa, 2017).

2.4.1 Parameter PSO

$X_i(t) = x_1(t), x_2(t), \dots, x_N(t)$

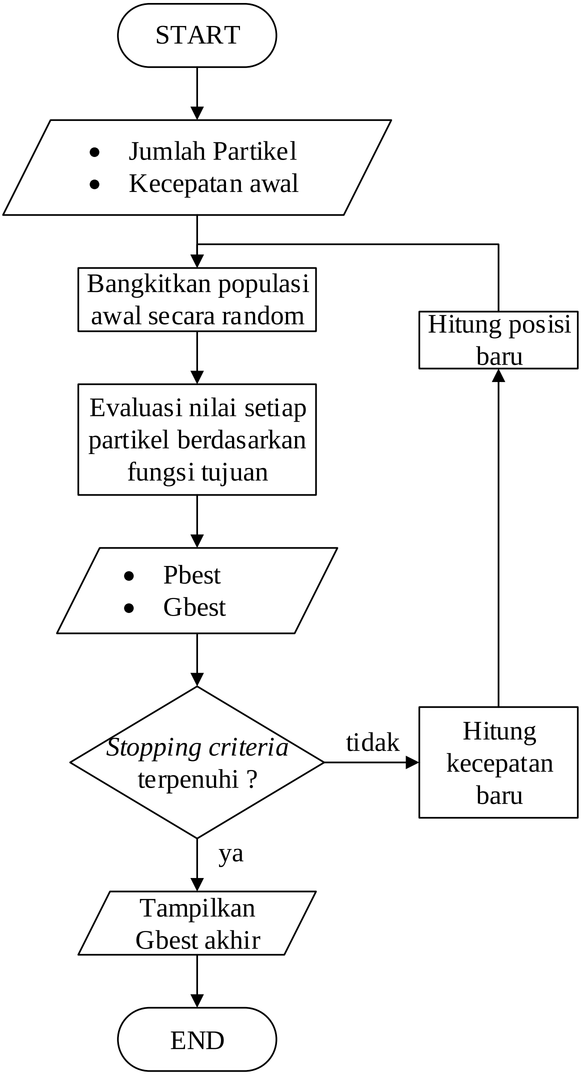
$V_i(t) = v_1(t), v_2(t), \dots, v_N(t)$

Dimana

X	=	posisi partikel
V	=	kecepatan partikel
i	=	indeks
t	=	iterasi ke-t
N	=	ukuran dimensi ruang

2.4.2 Algoritma PSO

Algoritma PSO secara garis besar dapat digambarkan dengan diagram alir seperti pada Gambar 2.11



Gambar 2.15 Diagram alir PSO

Secara garis besar algoritma PSO bisa dijelaskan sebagai berikut :

1. Bangkitkan populasi awal.
   Tentukan ukuran populasi, tentukan kecepatan awal setiap partikel. Bangkitkan populasi awal secara random. Evaluasi nilai fungsi tujuan untuk setiap partikel dan tetapkan P_{best}(Nilai terbaik dari tiap partikel) dan G_{best}(Nilai terbaik dari populasi).
2. Bangkitkan nilai random antara 0 sampai 1, lalu hitung kecepatan baru dengan rumus
   $$V_j(i) = V_j(i-1) + c_1 r_1 \left[P_{best,j} - x_j(i-1)\right] + c_2 r_2 \left[G_{best,j} - x_j(i-1)\right]$$
3. Hitung posisi partikel j pada iterasi ke-i dengan rumus
   $$x_j(i) = x_j(i-1) + v_j(i)$$
4. Evaluasi nilai fungsi tujuan untuk setiap posisi partikel baru
5. Perbarui nilai P_{best} dan G_{best}
   P_{best} dan G_{best} dibandingkan dengan nilai fungsi tujuan baru, apabila nilai fungsi tujuan baru lebih baik daripada nila P_{best} atau G_{best} , maka perbarui nilai P_{best} atau G_{best}
6. Jika belum memenuhi *stopping criteria*, kembali ke langkah 2

2.4.3 Contoh Aplikasi PSO

Cari nilai minimum dari fungsi berikut

$$f(x) = (100 - x)^2$$

Dimana $60 \leq x \leq 120$

1. Tentukan jumlah partikel N =4
   Tentukan populasi awal secara random, misalkan didapat
   $x_1(0)$ = 80
   $x_2(0)$ = 90
   $x_3(0)$ = 110
   $x_4(0)$ = 75
2. Evaluasi nilai fungsi tujuan untuk setiap partikel $x_j(0)$ untuk j = 1, 2, 3, 4, dan nyatakan dengan

$f_1(0) = f(80) = 400$
$f_2(0) = f(90) = 100$
$f_3(0) = f(110) \quad = 100$
$f_4(0) = f(75) = 625$

3. Tentukan kecepatan awal $v_1(0) = v_2(0) = v_3(0) = v_4(0) = 0$. Tetapkan iterasi i = 1,lalu ke langkah 4
4. Temukan

$P_{best, 1} \quad = 80$
$P_{best, 2} \quad = 90$
$P_{best, 3} \quad = 110$
$P_{best, 4} \quad = 75$
$G_{best} \quad = 90$

Hitung v(j) dengan $c_1 = c_2 = 1$. Misalkan nilai random yang didapat, $r_1 = 0,4$, $r_2 = 0,5$, diperoleh

$v_1(1) \quad = 0 + 0,4(80 - 80) + 0,5(90 - 80) = 5$
$v_2(1) \quad = 0 + 0,4(90 - 90) + 0,5(90 - 90) = 0$
$v_3(1) \quad = 0 + 0,4(110 - 110) + 0,5(90 - 110) = -10$
$v_4(1) \quad = 0 + 0,4(75 - 75) + 0,5(90 - 75) = 7.5$

Sedangkan untuk nilai x adalah

$x_1(1) \quad = 80 + 5 = 85$
$x_2(1) \quad = 90 + 0 = 90$
$x_3(1) \quad = 110 – 10 = 100$
$x_4(1) \quad = 75 + 7,5 = 82,5$

5. Evaluasi nilai fungsi tujuan sekarang pada partikel $x_j(1)$,

$f_1(1) = f(85) \quad = 225$
$f_2(1) = f(90) \quad = 100$
$f_1(1) = f(100) \quad = 0$
$f_1(1) = f(82,5) \quad = 306,25$

Sedangkan pada iterasi sebelumnya kita dapatkan

$f_1(0) = f(80) = 400$
$f_2(0) = f(90) = 100$
$f_3(0) = f(110) \quad = 100$
$f_4(0) = f(75) = 625$

Cek apakah solusi x sudah konvergen, dimana nilai x saling dekat. Jika tidak, tingkatkan ke iterasi berikutnya i = 2. Lanjutkan ke langkah 4.

4. $P_{best,\,1}$ = 85
   $P_{best,\,2}$ = 90
   $P_{best,\,3}$ = 100
   $P_{best,\,4}$ = 82,5
   G_{best} = 100
   Hitung kecepatan baru dengan r_1 = 0,3 dan r_2 = 0,6
   $v_1(2)$ = 5 + 0,3(85 - 85) + 0,6(100 - 85) = 14
   $v_2(2)$ = 0 + 0,3(90 - 90) + 0,6(100 - 90) = 6
   $v_3(2)$ = -10 + 0,3(100 - 100) + 0,6(100 - 100) = -10
   $v_4(2)$ = 7,5 + 0,3(82,5 - 82,5) + 0,6(100 - 82,5) = 18
   Sedangkan untuk nilai x adalah
   $x_1(2)$ = 85 + 14 = 99
   $x_2(2)$ = 90 + 6 = 96
   $x_3(2)$ = 100 – 10 = 90
   $x_4(2)$ = 82,5 + 18 = 100,5
5. Evaluasi nilai fungsi tujuan sekarang pada partikel $x_j(2)$
   $f_1(2) = f(99) = 1$
   $f_2(2) = f(96) = 16$
   $f_3(2) = f(90) = 100$
   $f_4(2) = f(100,5) = 0,25$
   Jika dibandingkan dengan nilai f dari iterasi sebelumnya, ada nilai yang lebih baik dari nilai f sekarang yaitu $f_3(1)$ = 0, sehingga P_{best} untuk partikel 3 sama dengan 100, dan G_{best} dicari dari min [1, 16, 0, 0,25] = 0 yang dicapai pada $x_3(1)$ = 100. Sehingga untuk iterasi berikutnya P_{best} = (99, 96, 100, 100,5) dan G_{best} = 100.

Cek apakah solusi sudah konvergen, dimana nilai x saling berdekatan. Jika belum konvergen, set i = 3, masuk ke iterasi berikutnya. Lanjutkan ke langkah berikutnya dengan menghitung

kecepatan v dan ulangi langkah-langkah selanjutnya sampai mencapai konvergen

2.5 Desirability Function

Desirability function adalah sebuah metode optimasi untuk permasalahan yang memiliki respon lebih dari 1 variabel (Manohar, 2013). Konsep dari metode ini adalah membandingkan respon hasil prediksi dengan nilai minimum dan maksimum untuk respon tersebut untuk mendapatkan nilai yang disebut *desirability*. Nilai *desirability* yang didapat lalu digunakan untuk mencari nilai *composite desirability* atau nilai *desirability* gabungan dari beberapa respon, secara matematis dituliskan dengan persamaan sebagai berikut

Jika target T_i adalah nilai maksimum

$$d_i = \begin{cases} 0 & y_i < L_i \\ \left(\dfrac{y_i - L_i}{T_i - L_i}\right)^w & L_i \le y_i \le T_i \\ 1 & y_i > T_i \end{cases}$$

Jika target T_i adalah nilai minimum

$$d_i = \begin{cases} 1 & y_i < L_i \\ \left(\dfrac{T_i - y_i}{T_i - L_i}\right)^w & L_i \le y_i \le T_i \\ 0 & y_i > T_i \end{cases}$$

Jika target T_i berada diantara nilai minimum dan maksimum

$$d_i = \begin{cases} 0 & y_i < L_i \\ \left(\dfrac{y_i - L_i}{T_i - L_i}\right)^w & L_i \le y_i \le T_i \\ \left(\dfrac{U_i - y_i}{U_i - L_i}\right)^w & T_i \le y_i \le U_i \\ 0 & y_i > U_i \end{cases}$$

dimana d_i = nilai *desirability*, y_i = nilai respon, L_i = nilai minimum, U_i = nilai maksimum, T_i = nilai target, setelah itu dicari nilai *composite desirability* seperti pada persamaan berikut

$$D = (d_1 \times d_2 \times d_3 \times \ldots \times d_n)^{\frac{1}{n}} = \left(\prod_{i-1}^{n} d_i\right)^{\frac{1}{n}}$$

dimana D = nilai *composite desirability*, d_i = nilai *desirability*, dan n adalah jumlah variabel respon, apabila tiap respon memiliki bobot berbeda-beda, maka persamaan akan menjadi seperti pada persamaan berikut

$$D = {d_1}^{w1} \times {d_2}^{w2} \times {d_3}^{w3} \times \ldots \times {d_n}^{wn}$$

dimana w adalah bobot masing-masing respon, dan nilai w_i yaitu $0 < w_i < 1$, dan $w_1 + w_2 + \ldots + w_n = 1$.

2.6 Kajian Pustaka

2.6.1 Parametric Study and Optimizing Using RSM of DI Diesel Engine for Lower Emissions (Prabhakara, 2016)

Penelitian ini bertujuan untuk mencari nilai optimum dari multivariabel yang terdiri dari peak pressure, kadar NO_x dan *soot* daro mesin diesel CAT 3401. Variabel bebas pada penelitian ini antara lain *Compression Ratio* (CR), *Start of Injection Angle* (SOI), *Fuel Injection Pressure* (FIP), dan *Exhaust Gas Recirculation* (EGR). *Software* CONVERGE CFD digunakan untuk memprediksi nilai respon dari setiap kombinasi variabel. Data yang didapat dari simulasi digunakan untuk mencari fungsi objektif dengan menggunakan metode regresi, setelah itu dilakukan optimasi dengan metode *Response Surface Methodology* (RSM). Parameter dikatakan optimum apabila memiliki nilai *composite desireability* yang tinggi, dimana nilai *composite desireability* ini berkisar antara 0 sampai 1. Parameter menunjukkan nilai *composite desireability* tertinggi dengan nilai 0.95 yaitu dengan nilai parameter CR 14.25, FIP 1153.15 bar, SOI 13.69 bTDC, dan EGR 16.91 % didapat nilai respon *peak pressure* 114 bar, NO_x 15.24 g/kg *fuel*, *soot* 0.23 g/kg *fuel*. Parameter optimum tersebut disimulasikan dengan *software* CONVERGE didapat respon *peak pressure pressure* 113,5 bar, NO_x 15.8 g/kg *fuel*, *soot* 0.20 g/kg *fuel*. Perbandingan antara

model regresi dengan hasil simulasi menggunakan *software* CONVERGE dapat dilihat pada Tabel 2.1

Tabel 2.1 Perbandingan hasil simulasi CONVERGE dengan metode regresi

	Peak pressure (bar)	NO_x (g/kg fuel)	Soot (g/kg fuel)
CONVERGE simulation	113.5	15.8	0.20
Regression model predicted	114	15.24	0.23
Error (%)	0.43	3.67	9.3

Pada hasil akhir penelitian disimpulkan bahwa meningkatkan nilai CR,FIP, dan SOI akan meningkatkan *peak pressure* dan NO_x, namun mengurangi kadar *soot*. Penulis menambahkan bahwa *Design of Experiment* (DOE) menggunakan RSM dapat memberikan parameter yang optimal didasarkan pada pendekatan *composite desirability* untuk mendapatkan performa yang diinginkan.

2.6.2 Evaluation of Artificial Neural Network Performance in Predicting Diesel Engine NO_x Emissions (Obodeh, 2009)

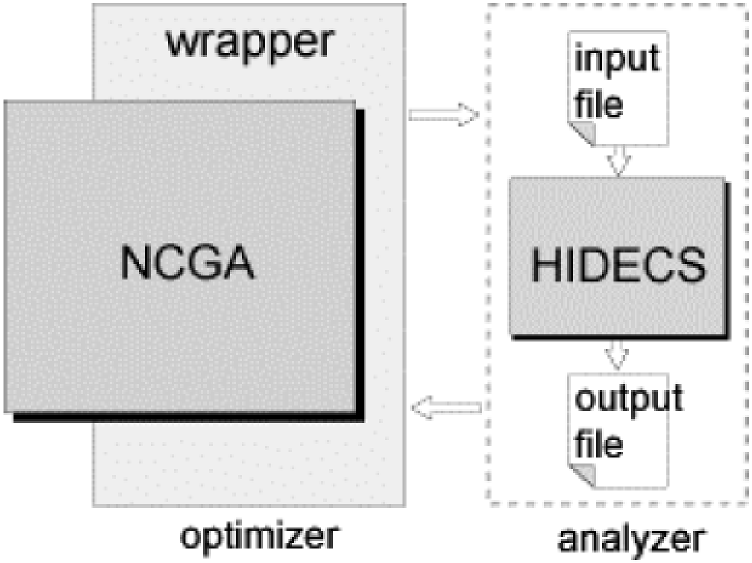
Penelitian ini bertujuan untuk mengevaluasi kapabilitas dari ANN sebagai alat untuk memprediksi emisi NO_x dari *multi-cylinder diesel engine*. Eksperimen dilakukan dengan alat uji *stationary ligh-duty Nissan diesel engine* yang didesain untuk memungkinkan pengujian didalam lingkungan laboratorium. Eksperimen dilakukan dengan 33 kondisi operasi berbeda, dari 33 data yang didapat 23 data digunakan untuk *learning* dan 10 data digunakan untuk *testing*. Model ANN yang digunakan yaitu *Multi Layer Perceptron* (MLP) dengan 3 hidden layer dan jumlah neuron antara 3 sampai 15 neuron. *Fraction of Variance* (R^2) dan *mean absolute percentage error* (ξ) digunakan sebagai pembanding dari setiap ANN yang telah dibuat. ANN dengan 11 neuron memiliki nilai R^2 tertinggi yaitu 0.997 maka dari itu

arsitektur ini dipilih sebagai alat untuk memprediksi emisi NO_x yang dihasilkan dari variasi variabel operasi. *Mean absolute percentage error* dalam prediksi nilai-nilai NO_x berada di kisara 0,68% hingga 3,34%, hal ini menunjukkan bahwa ANN dapat memprediksi nilai NO_x dengan baik.

Pada hasil akhir penelitian disimpulkan bahwa arsitektur yang terlalu kecil (memiliki neuron terlalu sedikit) terlalu sederhana untuk memberikan hasil yang baik dalam masalah yang diteliti. Batas atas diambil di 15 neuron karena semakin banyak jumlah neuron maka dibutuhkan data eksperimen yang lebih banyak, karena dengan banyaknya jumlah neuron semakin banyak *weight* yang harus disesuaikan. Diantara kombinasi *learning criteria* yang digunakan pada arsitektur dari *backpropagation* pada penelitian kali ini akurasi terbaik didapat pada saat *learning rate* 0,05 , momentum 0,05 , dan 0,3 *initial weight.*

2.6.3 Genetic Algorithms Optimization of Diesel Engine Emissions and Fuel Efficiency with Air Swirl, EGR, Injection Timing and Multiple Injections (Hiro, 2003)

Penelitian ini bertujuan untuk mencari nilai optimum dari *multi-objective optimization problem* (MOP) yaitu *Specific Fuel Consumption* (SFC) dan emisi (NO_x dan *soot*) dari mesin diesel. Variabel bebas yang digunakan dalam penelitian ini antara lain *multiple injections pattern*, *start of injection timing,* dan *exhaust gas recirculation*. Model komputasi dari mesin diesel HIDECS digunakan untuk menganalisa emisi dan performa dari mesin diesel, setelah itu dilakukan optimasi dengan metode *Neighborhood Cultivation Genetic Algorithm* (NCGA) untuk mendapatkan nilai parameter yang optimum. Konsep *Pareto Optimum Solution* digunakan pada saat optimasi dengan NCGA karena optimasi dilakukan terhadap multi variabel. Skema penelitian ditunjukkan pada gambar 2.12



Gambar 2.16 Skema penelitian

Pada hasil akhir penelitian disimpulkan bahwa dengan penambahan EGR, emisi NO_x akan berkurang dan pembentukan *soot* terjadi apabila bahan bakar terlalu cepat diinjeksikan. Penulis menambahkan bahwa *phenomenological model* cocok apabila dioptimasi menggunakan metode GA, dan penelitian ini juga menunjukkan bahwa GA dapat digunakan unuk memperoleh *Pareto Optimum Solution*.

2.6.4 Diesel Engine Emissions Reduction Using Particle Swarm Optimization (Prasanth, 2010)

Penelitian ini bertujuan untuk mencari nilai optimum dari *Multi-objective Optimization Problem* (MOP) yaitu *Brake Specific Fuel Consumption* (BSFC) dan emisi (NO_x, HC, CO, dan PM). Variabel bebas yang digunakan dalam peneliian kali ini adalah *main* SOI, *pilot* SOI, *pilot fuel quantity*, dan *Exhaust Gas Recirculation*. Mesin yang digunakan dalam penelitian ini adalah

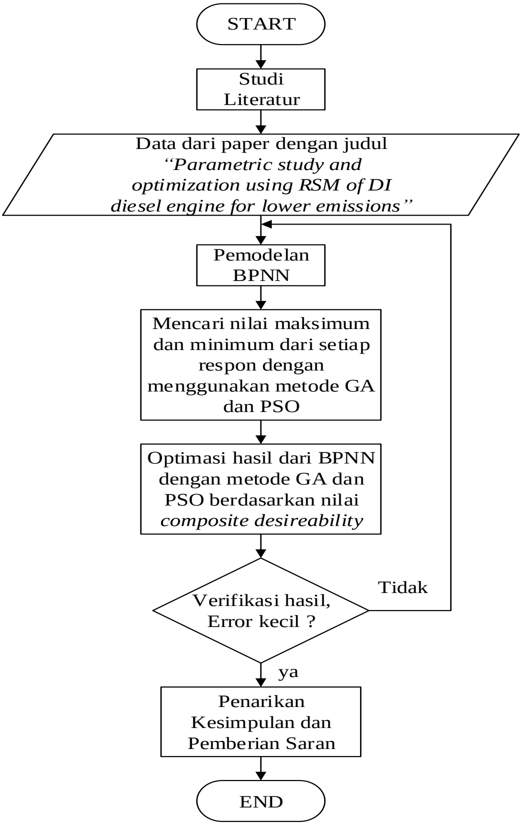
John Deere 4045HF475 nonroad four-cylinder 4.5-L engine yang memiliki daya 129 KW, mesin tersebut oleh *United States Environmental Protection Agency* (U.S EPA) digolongkan kedalam emisi *Tier* 4 (56-130 kW). Fungsi objektif yang akan digunakan dalam optimasi menggunakan metode PSO didapat dari *literature for engine optimization* (Montgomery, 2000). Parameter dikatakan optimum apabila nilai dari *fitness* sudah konvergen. Penelitian ini dilakukan dengan *single-injection optimization* pada temperatur *intake* 23°C, dan dengan *double-injection optimization* dengan temperatur *intake* 23°C dan 40°C. Jumlah populasi yang digunakan untuk optimasi dengan metode PSO yaitu 8 partikel. Pada *single-injection optimization* nilai optimum dicapai saat iterasi ke 7 dengan nilai parameter 31% EGR, 1 ATDC SOI didapat nilai respon 1.174 g/kW-h NOx, 0.223 g/kW-h HC, 0.0196 g/kW-h PM, dan 231 g/kW-h BSFC. Pada *double-injection optimization* dengan 23°C nilai optimum dicapai saat iterasi ke 5 dengan nilai parameter 4% EGR, 4 ATDC main SOI, dan 5% pilot pada 23 ATDC SOI didapat nilai respon 0.37 g/kW-h NO_x, 0.018 g/kW-h PM, 9.59 g/kW-h CO, 0.76 g/kW-h HC, dan 255 g/kW-h BSFC. Pada *double-injection optimization* dengan temperatur *intake* 40°C nilai optimum dicapai saat iterasi ke 8 dengan nilai parameter 34% EGR, 5 ATDC main SOI, dan 5% pilot fuel pada 24 ATDC SOI didapat nilai respon 0.41 g/kW-h NO_x, 0.0092 g/kW-h PM, 8.51 g/kW-h CO, 0.678 g/kW-h HC, dan 249 g/kW-h BSFC.

Pada akhir penelitian disimpulkan bahwa dengan nilai EGR yang sangat tinggi dapat mengurangi NO_x dan PM. SOI yang terlalu cepat dapat meningkatkan emisi PM yang sangat tinggi.

BAB 3
METODOLOGI PENELITIAN

3.1 Diagram Alir Penelitian

Aktivitas yang akan dilakukan dalam penelitian ini digambarkan pada diagram alir penelitian pada Gambar 3.1 sebagai berikut :



Gambar 3.1 Diagram alir tahapan penelitian

3.1.1 Studi Literatur

Langkah pertama pada penelitian ini adalah studi literatur. Studi literatur bertujuan untuk mendalami landasan teori yang akan digunakan pada penelitian ini. Studi literatur dimulai dari mencari jurnal-jurnal penelitian sebelumnya yang meneliti tentang topik yang sejenis, yaitu mengenai optimasi mesin diesel, *Artificial Neural Network* (ANN), Genetik Algoritma (GA), dan *Particle Swarm Optimization* (GA).

3.1.2 Data

Penelitian kali ini menggunakan data yang berasal dari paper dengan judul "*Parametric Study and Optimizing Using RSM of DI Diesel Engine for Lower Emissions*" oleh Prabhakara Rao Ganji, Kiran Prasad Chintala, V.R.K Raju dan Srinivasa Rao Surapaneni (2016). Berikut data yang akan digunakan seperti pada Tabel 3.1

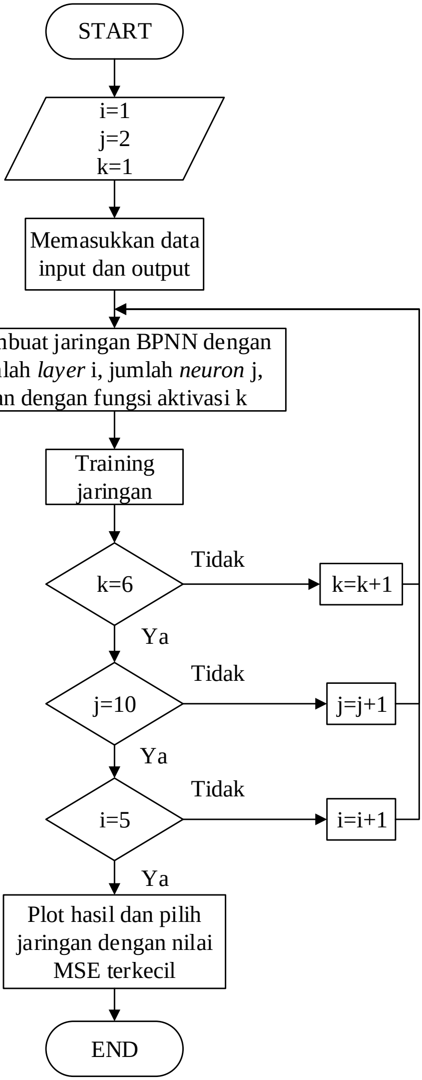
Tabel 3.1

S. no.	Run order	Parameter				Respon		
		CR	FIP(bar)	SOI °bTDC	EGR %	Peak pressure (bar)	Nox(g/kg of fuel)	Soot(g/kg of fuel)
1	4	12	950	15	25	80	4,21	4,83
2	7	12	500	15	12,5	78,8	8,55	4,97
3	21	12	950	15	0	92,7	29,11	0,62
4	22	12	950	0	12,5	61,1	5,78	5,02
5	23	12	950	30	12,5	124,3	71,42	0,06
6	24	12	1400	15	12,5	92,5	17,71	0,91
7	2	14,25	1400	0	12,5	77	7,29	1,71
8	3	14,25	1400	15	25	113,4	11,95	0,49
9	5	14,25	500	15	25	94,7	3,97	3,7
10	6	14,25	950	30	0	168,1	121	0,01
11	9	14,25	500	30	12,5	137,2	42,39	0,77
12	10	14,25	950	0	25	71,3	3,21	5,25

13	11	14,25	500	0	12,5	70,7	3,2	5,97
14	12	14,25	1400	30	12,5	162,4	99,89	0,01
15	15	14,25	1400	15	0	121,9	55,57	0,12
16	16	14,25	950	30	25	144	33,13	0,79
17	17	14,25	500	15	0	98,7	19,48	0,95
18	19	14,25	950	15	12,5	108,6	18,32	0,55
19	20	14,25	950	0	0	74,1	10,46	2,53
20	25	14,25	950	15	12,5	108,6	18,32	0,55
21	26	14,25	950	15	12,5	108,6	18,32	0,56
22	1	16,5	950	0	12,5	86,5	8,13	1,25
23	8	16,5	500	15	12,5	118,6	19,82	0,78
24	13	16,5	950	30	12,5	183,9	88,94	0,03
25	14	16,5	1400	15	12,5	143,2	54,03	0,19
26	18	16,5	950	15	25	129,6	16,72	0,54
27	27	16,5	950	15	0	136,1	59,04	0,21

3.1.3 Pemodelan BPNN

Diagram alir proses pemodelan BPNN untuk mendapatkan jaringan yang akan digunakan dalam proses optimasi BPNN-GA dan BPNN-PSO ditunjukkan pada Gambar 3.2 berikut



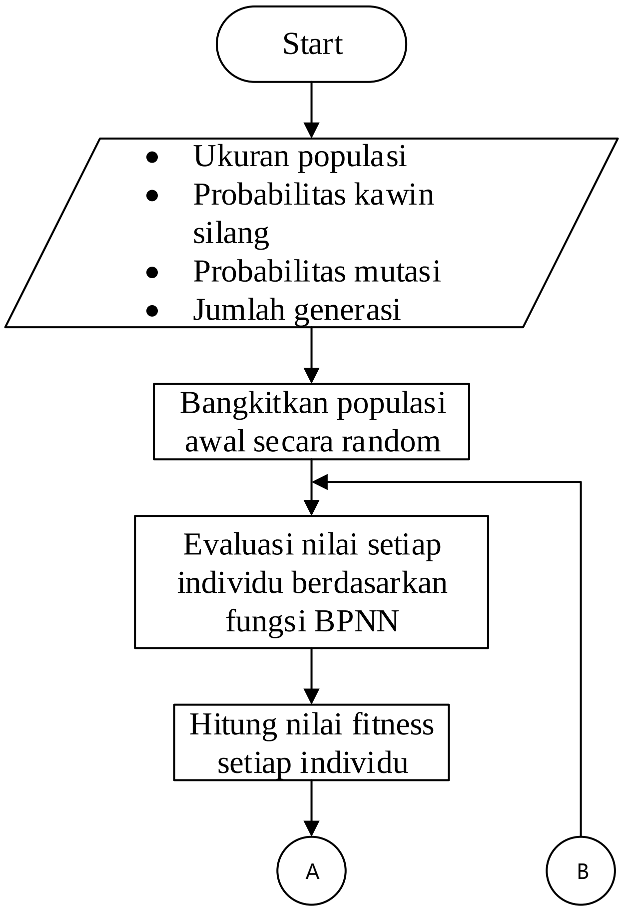
Gambar 3.2 Diagram alir pemodelan dengan BPNN

dimana k dengan nilai 1 s/d 7 adalah fungsi aktifasi dari BPNN yang secara umum bisa dijelaskan sbb: k = 1 (hardlim), k = 2 (hardlims), k = 3 (purelin), k = 4 (satlin), k = 5 (logsig), dan k = 6 (tansig).

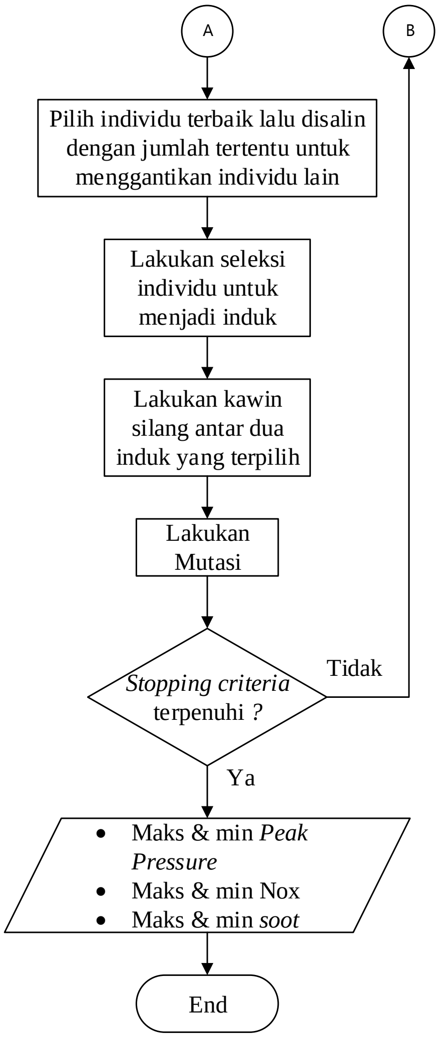
Jaringan yang akan digunakan dalam proses optimasi dengan BPNN-GA dan BPNN-PSO dipilih dari jaringan yang menghasilkan nilai *mean square error* (MSE) terkecil. Pada proses ini, dibuat rancangan jaringan berdasarkan kombinasi parameter jumlah lapisan tersembunyi, jumlah neuron, dan fungsi aktivasi yang digunakan, kemudian dilakukan proses *training*, *testing*, dan validasi untuk seluruh rancangan jaringan. Jaringan dengan nilai MSE terkecil akan digunakan dalam proses optimasi.

3.1.4 Mencari Nilai Maksimum dan Minimum

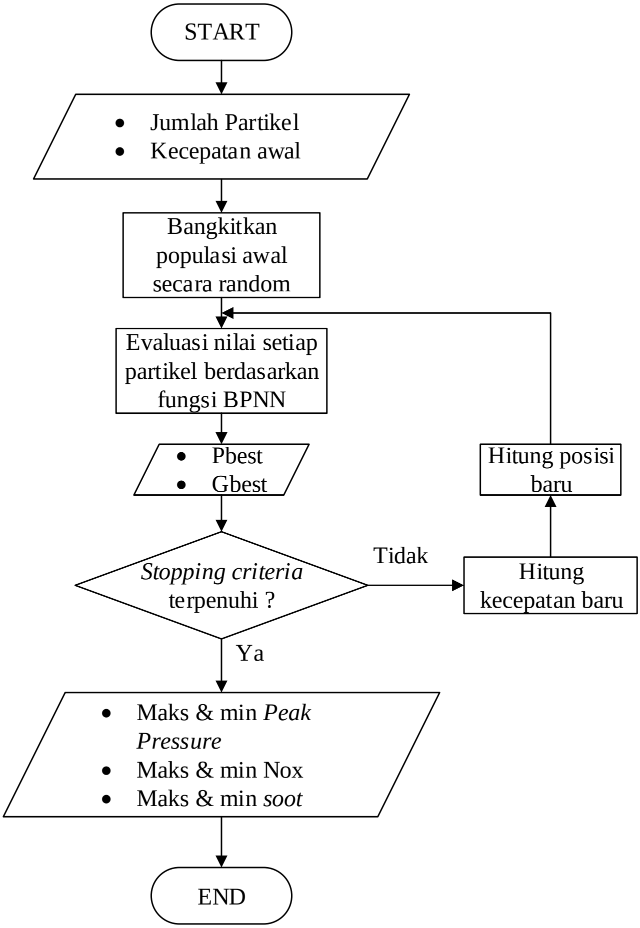
Nilai maksimum dan minimum dicari menggunakan 2 metode yaitu dengan menggunakan Genetik Algoritma (GA) dan *Particle Swarm Optimization* (PSO), pencarian nilai maksimum dan minimum dengan metode GA digambarkan dengan diagram alir seperti pada Gambar 3.3, pencarian nilai maksimum dan minimum dengan metode PSO digambarkan dengan diagram alir seperti pada Gambar 3.4



Start
Ukuran populasi
Probabilitas kawin silang
Probabilitas mutasi
Jumlah generasi
Bangkitkan populasi awal secara random
Evaluasi nilai setiap individu berdasarkan fungsi BPNN
Hitung nilai fitness setiap individu
A
B



Gambar 3.3 Diagram alir pencarian nilai minimum dan maksimum dengan metode GA

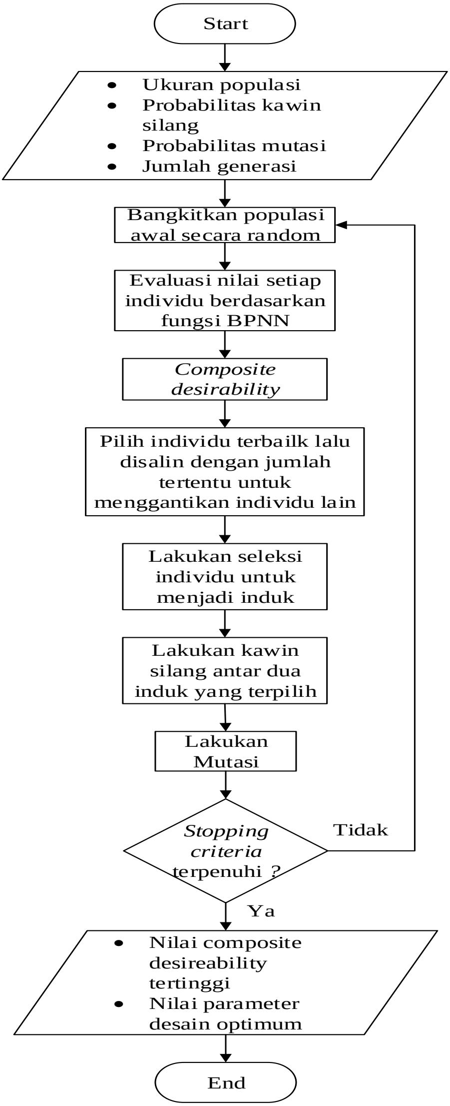


Gambar 3.4 Diagram alir pencarian nilai minimum dan maksimum dengan metode PSO

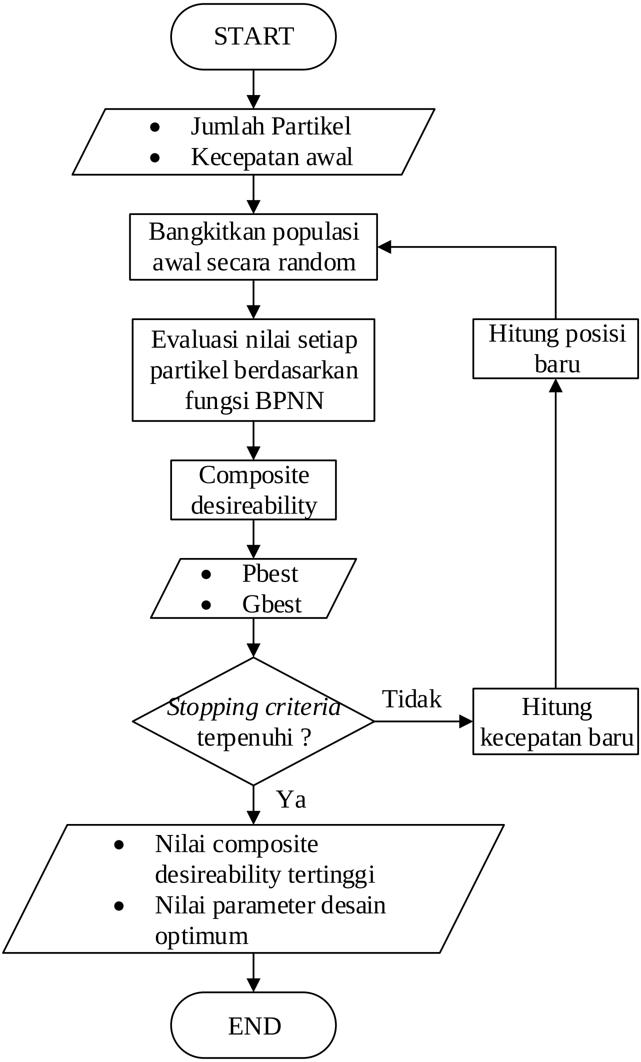
3.1.5 Optimasi dengan Metode GA dan PSO

Nilai maksimum dan minimum yang didapat dari langkah sebelumnya digunakan sebagai parameter untuk mencari nilai *composite desireability*. Nilai *composite desireability* digunakan sebagai acuan untuk mendapatkan parameter desain yang

optimum. Optimasi dengan metode GA digambarkan dengan diagram alir seperti pada gambar 3.5, sedangkan optimasi dengan metode PSO digambarkan dengan diagram alir seperti pada gambar 3.6



Gambar 3.5 Diagram alir optimasi dengan metode GA



Gambar 3.6 Diagram alir optimasi dengan metode PSO

3.1.6 Komparasi Hasil

Hasil yang didapat dari hasil optimasi menggunakan metode GA dan PSO dibandingkan dengan hasil dari penelitian sebelumnya pada *paper* dengan judul "*Parametric Study and Optimizing Using RSM of DI Diesel Engine for Lower Emissions*" yang menggunakan metode optimasi *Response Surface Methodology* (RSM).

3.1.7 Penarikan Kesimpulan dan Saran

Pada tahap ini akan membahas mengenai hasil dari penelitian serta pemberian saran agar dapat membantu penelitian yang akan dilakukan selanjutnya.

BAB 4
ANALISA DATA DAN PEMBAHASAN

4.1 Pengaruh Setiap Parameter Terhadap Fungsi Tujuan

- *Compression Ratio* (CR)

  Semakin besar *compression ratio* maka *peak pressure* dan kadar NOx akan meningkat, sedangkan kadar *soot* akan menurun. Hal tersebut disebabkan karena dengan meningkatnya *compression ratio* maka tekanan dan temperatur dalam ruang bakar akan meningkat sehingga pembakaran yang terjadi akan lebih baik. Pembakaran yang lebih baik akan meningkatkan *peakpressure* dan juga kadar NOx, sedangkan kadar *soot* akan menurun.

- *Fuel Injection Pressure* (FIP)

  Semakin besar *fuel injection pressure* akan membuat evaporasi menjadi lebih baik dan atomisasi bahan bakar akan berlangsung cepat. Hal tersebut akan membuat campuran antara udara dan bahan bakar pada ruang bakar semakin homogen. Semakin homogen campuran bahan bakar maka proses pembakaran akan lebih baik. Pembakaran yang lebih baik akan meningkatkan peakpressure dan juga kadar NOx, sedangkan kadar *soot* akan menurun.

- *Start Of Injection* (SOI)

  Semakin besar *start of injection* (dalam ˚bTDC) maka akan meningkatkan *ignition delay*. Dengan meningkatnya *ignition delay* maka akan memberikan waktu bahan bakar dan udara bercampur, sehingga campuran akan menjadi lebih homogen. Semakin homogen campuran bahan bakar maka proses pembakaran akan lebih baik. Pembakaran

yang lebih baik akan meningkatkan *peakpressure* dan juga kadar NOx, sedangkan kadar *soot* akan menurun.

- *Exhaust Gas Recirculation*

  Semakin besar nilai *exhaust gas recirculation* (dalam %) maka kadar oksigen dalam ruang bakar akan berkurang. Berkurangnya kadar oksigen dalam ruang bakar akan membuat pembakaran kurang sempurna, sehingga temperatur pembakaran akan menurun. Menurunnya temperatur pembakaran menyebabkan pembentukan NOx berkurang, dan juga akan menurunkan *peakpressure*, sedangkan pembakaran yang tidak sempurna akan meyebaban kadar *soot* meningkat.

4.2 Pemodelan BPNN

Hasil pemodelan BPNN dapat dilihat pada Lampiran 1 . Model yang akan digunakan adalah model yang memiliki nilai MSE terkecil, untuk *peakpressure* digunakan model BPNN dengan layer sebanyak 3, jumlah neuron 9, dan fungsi aktivasi tansig dengan nilai MSE 0,014689. Untuk prediksi NOx digunakan model BPNN dengan layer sebanyak 3, jumlah neuron 9, dan fungsi aktivasi tansig dengan nilai MSE 0,020889. Untuk prediksi *soot* digunakan model BPNN dengan layer sebanyak 5, jumlah neuron 10, dan fungsi aktivasi satlin dengan nilai MSE 0.008878. Berikut adalah perbandingan antara nilai output dengan nilai hasil prediksi dari BPNN.

Tabel 4.1 Perbandingan nilai output dengan nilai hasil prediksi peakpressure

PEAKPRESSURE

Input				Output	Prediksi
CR	FIP	SOI	EGR		
12	950	15	25	80	80,00406
12	500	15	12,5	78,8	78,80233
12	950	15	0	92,7	92,70294
12	950	0	12,5	61,1	61,09323
12	950	30	12,5	124,3	124,2991
12	1400	15	12,5	92,5	92,50569
14,25	1400	0	12,5	77	77,0049
14,25	1400	15	25	113,4	113,3983
14,25	500	15	25	94,7	94,69667
14,25	950	30	0	168,1	168,0968
14,25	500	30	12,5	137,2	137,2017
14,25	950	0	25	71,3	71,30193
14,25	500	0	12,5	70,7	70,70141
14,25	1400	30	12,5	162,4	162,4019
14,25	1400	15	0	121,9	121,894
14,25	950	30	25	144	144,0031
14,25	500	15	0	98,7	98,70019
14,25	950	15	12,5	108,6	108,6029
14,25	950	0	0	74,1	74,09936
14,25	950	15	12,5	108,6	108,6029
14,25	950	15	12,5	108,6	108,6029
16,5	950	0	12,5	86,5	86,49809
16,5	500	15	12,5	118,6	118,5996
16,5	950	30	12,5	183,9	183,8979
16,5	1400	15	12,5	143,2	143,2031
16,5	950	15	25	129,6	129,6
16,5	950	15	0	136,1	136,1014

Tabel 4.2 Perbandingan nilai output dengan nilai hasil prediksi NOx

NOX					
Input				Output	Prediksi
CR	FIP	SOI	EGR		
12	950	15	25	4,21	4,215018
12	500	15	12,5	8,55	8,545255
12	950	15	0	29,11	29,11068
12	950	0	12,5	5,78	5,775786
12	950	30	12,5	71,42	71,4182
12	1400	15	12,5	17,71	17,71042
14,25	1400	0	12,5	7,29	7,290943
14,25	1400	15	25	11,95	11,95114
14,25	500	15	25	3,97	3,96792
14,25	950	30	0	121	120,9968
14,25	500	30	12,5	42,39	42,39217
14,25	950	0	25	3,21	3,210037
14,25	500	0	12,5	3,2	3,206178
14,25	1400	30	12,5	99,89	99,89064
14,25	1400	15	0	55,57	55,57187
14,25	950	30	25	33,13	33,13142
14,25	500	15	0	19,48	19,48204
14,25	950	15	12,5	18,32	18,32123
14,25	950	0	0	10,46	10,45865
14,25	950	15	12,5	18,32	18,32123
14,25	950	15	12,5	18,32	18,32123
16,5	950	0	12,5	8,13	8,12996

16,5	500	15	12,5	19,82	19,81898
16,5	950	30	12,5	88,94	88,93779
16,5	1400	15	12,5	54,03	54,02972
16,5	950	15	25	16,72	16,72098
16,5	950	15	0	59,04	59,04011

Tabel 4.3 Perbandingan nilai output dengan nilai hasil prediksi soot

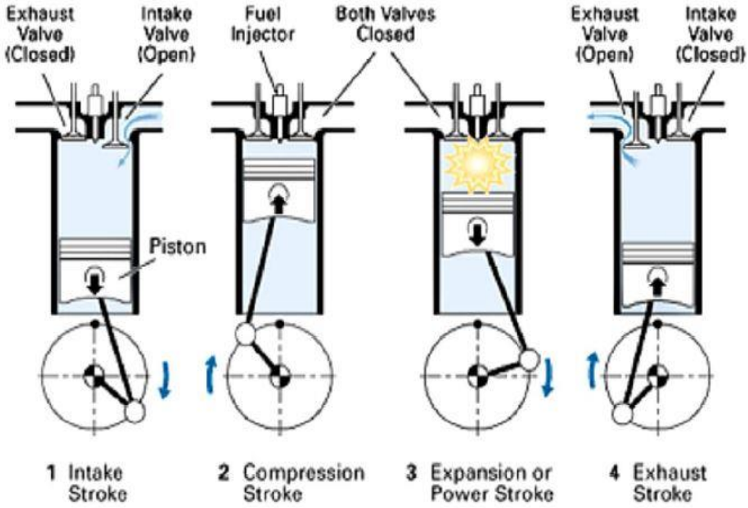
Soot					
Input				Target	Output
CR	FIP	SOI	EGR		
12	950	15	25	4,83	4,826331
12	500	15	12,5	4,97	4,964697
12	950	15	0	0,65	0,660425
12	950	0	12,5	5,02	5,022527
12	950	30	12,5	0,06	0,071931
12	1400	15	12,5	0,91	0,903187
14,25	1400	0	12,5	1,71	1,712676
14,25	1400	15	25	0,49	0,48166
14,25	500	15	25	3,7	3,708288
14,25	950	30	0	0,01	0,017589
14,25	500	30	12,5	0,77	0,768332
14,25	950	0	25	5,25	5,247055
14,25	500	0	12,5	5,97	5,962371
14,25	1400	30	12,5	0,01	0,017589
14,25	1400	15	0	0,12	0,114472
14,25	950	30	25	0,79	0,790157
14,25	500	15	0	0,95	0,952295

14,25	950	15	12,5	0,55	0,545487
14,25	950	0	0	2,53	2,521784
14,25	950	15	12,5	0,55	0,545487
14,25	950	15	12,5	0,56	0,545487
16,5	950	0	12,5	1,25	1,253888
16,5	500	15	12,5	0,78	0,783442
16,5	950	30	12,5	0,03	0,031653
16,5	1400	15	12,5	0,19	0,181513
16,5	950	15	25	0,54	0,539598
16,5	950	15	0	0,21	0,209596

Dari hasil prediksi masing-masing respon didapat bahwa perbedaan dengan nilai aktual cukup kecil, sehingga model BPNN dapat diterima untuk memprediksi respon dari mesin diesel CAT 3401.

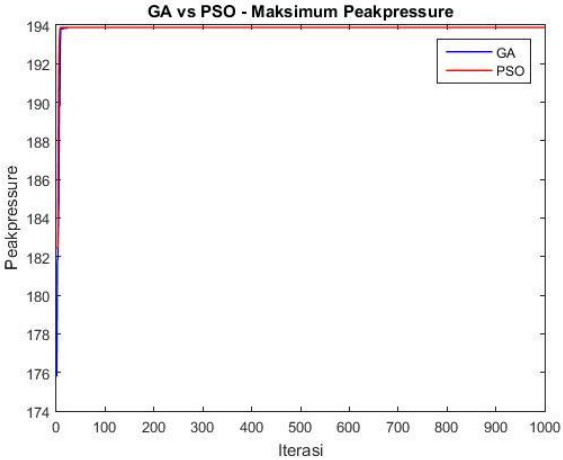
4.3 Nilai Minimum dan Maksimum

Setelah dilakukan pemodelan BPNN langkah selanjutnya adalah mencari nilai minimum dan maksimum dari masing-masing respon menggunakan metode GA dan PSO. Hipotesa awal GA akan memiliki hasil yang lebih optimum dibandingkan dengan PSO, namun membutuhkan waktu yang lebih lama untuk mencapai nilai optimum. Berikut adalah perbandingan pencarian nilai maksimum dan minimum dari masing-masing variabel dengan menggunakan metode GA dan PSO



- ***Peakpressure***

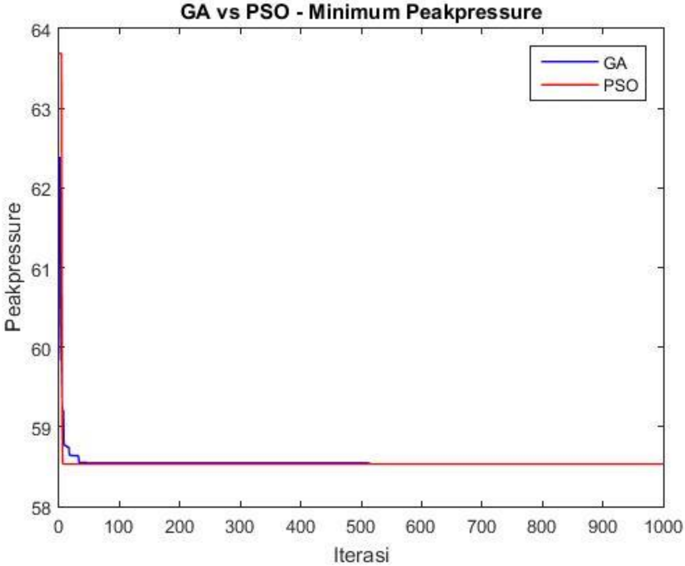
  Peakpressure merupakan tekanan yang dihasilkan pada akhir dari proses pembakaran (langkah 3). Semakin sempurna pembakaran yang terjadi, maka peakpressure yang dihasilkan akan semakin besar.



Gambar 4.1 Perbandingan nilai maksimum *peakpressure* GA dengan PSO

Dalam pencarian nilai maksimum peakpressure diperlukan pengaturan parameter CR, FIP, SOI, dan EGR yang sesuai dimana untuk mendapatkan nilai *peakpressure* yang tinggi maka diperlukan nilai parameter CR, FIP, dan SOI yang tinggi dan nilai parameter EGR yang rendah. Semakin besar *compression ratio* maka *peak pressure* akan meningkat. Hal tersebut disebabkan karena dengan meningkatnya *compression ratio* maka tekanan dan temperatur dalam ruang bakar akan meningkat sehingga pembakaran yang terjadi akan lebih baik. Semakin besar *fuel injection pressure* (pemasukan bahan bakar/langkah 2) akan membuat evaporasi menjadi lebih baik dan atomisasi bahan bakar akan berlangsung cepat. Hal tersebut akan membuat campuran antara udara dan bahan bakar pada ruang bakar semakin homogen. Semakin homogen campuran bahan bakar maka proses pembakaran akan lebih baik sehingga *peakpressure* akan meningkat. Semakin besar *start of injection* (dalam ˚bTDC/pada saat sebelum

langkah 2 berakhir bahan bakar mulai diinjeksikan) maka akan meningkatkan *ignition delay*. Dengan meningkatnya *ignition delay* maka akan memberikan waktu bahan bakar dan udara bercampur, sehingga campuran akan menjadi lebih homogen. Semakin homogen campuran bahan bakar maka proses pembakaran akan lebih baik sehingga *peakpressure* akan meningkat. Semakin kecil nilai *exhaust gas recirculation* (dalam % / udara hasil pembuangan pada langkah 4 yang digunakan kembali pada langkah 1) maka kadar oksigen dalam ruang bakar akan meningkat. Meningkatnya kadar oksigen dalam ruang bakar akan membuat pembakaran menjadi lebih sempurna, sehingga *peakpressure* akan meningkat. Grafik pada Gambar 4.1 menunjukkan perbandingan metode GA dan PSO dalam mencari nilai maksimum *peakpressure*. Pada grafik tersebut didapat bahwa PSO mampu mencapai nilai maksimum pada iterasi 15 dengan nilai 193,880559772805 bar, sedangkan GA mencapai nilai maksimum pada iterasi 24 dengan nilai 193,880554545609 bar. Hasil tersebut menyatakan bahwa PSO mampu mencapai nilai *peakpressure* maksimum lebih cepat daripada GA dengan selisih hasil yang sangat kecil atau mendekati sama.



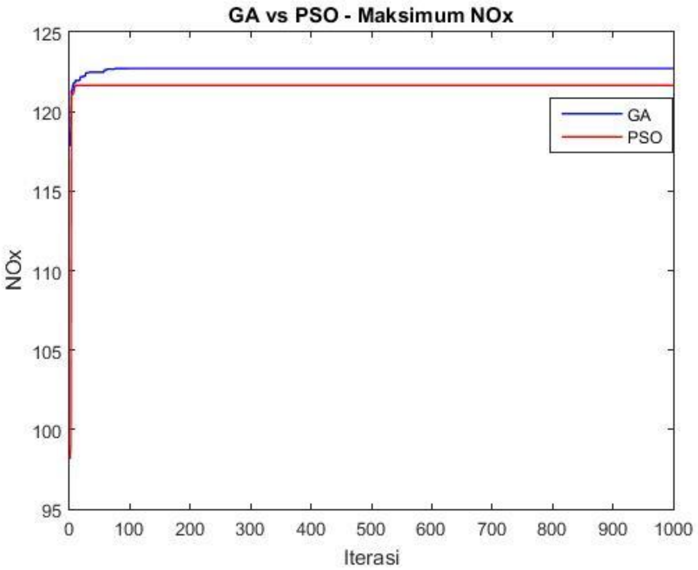
Gambar 4.2 Perbandingan nilai minimum peakpressure GA dengan PSO

Dalam pencarian nilai minimum peakpressure maka perlu dilakukn pengaturan parameter yang berkebalikan dengan pada saat mencari nilai maksimum *peakpressure* dimana dibutuhkan nilai variabel CR, FIP, SOI yang kecil dan nilai EGR yang besar sehingga pembakaran menjadi tidak sempurna. Pembakaran yang tidak sempurna akan membuat *peakpressure* menurun. Grafik pada Gambar 4.2 menunjukkan perbandingan metode GA dan PSO dalam mencari nilai maksimum *peakpressure*. Pada grafik tersebut didapat bahwa PSO mampu mencapai nilai maksimum pada iterasi 200 dengan nilai 58,53138128 bar, sedangkan GA mencapa nilai maksimum pada iterasi 653 dengan nilai 58,53138138 bar. Hasil tersebut menyatakan bahwa PSO mampu mencapai nilai *peakpressure*

maksimum lebih cepat daripada GA dengan selisih hasil yang sangat kecil atau mendekati sama.

- **NOx**

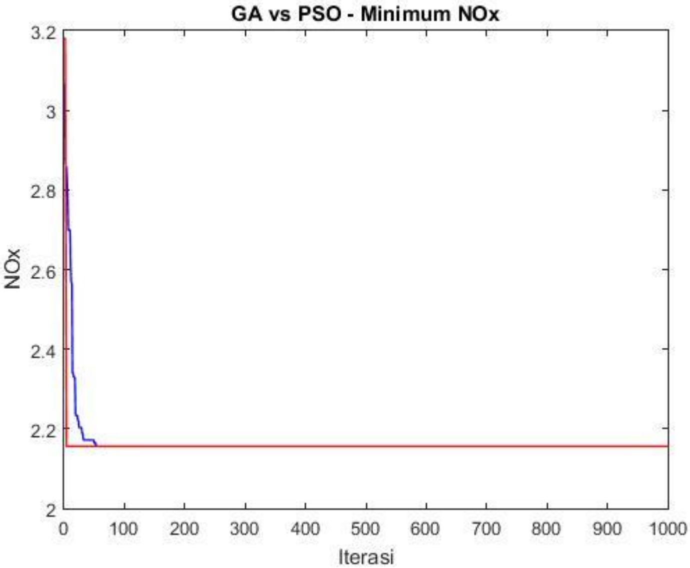
NOx merupakan gas yang dihasilkan pada saat pembakaran (langkah 3), yang selanjutnya akan dibuang pada saat langkah buang (langkah 4). Semakin sempurna pembakaran yang terjadi maka kadar NOx yang dihasilkan akan semakin meningkat, dan sebaliknya.



Gambar 4.3 Perbandingan nilai maksimum NOx GA dengan PSO

Dalam pencarian nilai maksimum NOx diperlukan pengaturan parameter CR, FIP, SOI, dan EGR yang sesuai dimana untuk mendapatkan nilai NOx yang tinggi maka diperlukan nilai parameter CR, FIP, dan SOI yang tinggi dan nilai parameter EGR yang rendah. Semakin besar *compression ratio* maka NOx akan meningkat. Hal

tersebut disebabkan karena dengan meningkatnya *compression ratio* maka tekanan dan temperatur dalam ruang bakar akan meningkat sehingga pembakaran yang terjadi akan lebih baik. Semakin besar *fuel injection pressure* (pemasukan bahan bakar/langkah 2) akan membuat evaporasi menjadi lebih baik dan atomisasi bahan bakar akan berlangsung cepat. Hal tersebut akan membuat campuran antara udara dan bahan bakar pada ruang bakar semakin homogen. Semakin homogen campuran bahan bakar maka proses pembakaran akan lebih baik sehingga NOx akan meningkat. Semakin besar *start of injection* (dalam ˚bTDC/pada saat sebelum langkah 2 berakhir bahan bakar mulai diinjeksikan) maka akan meningkatkan *ignition delay*. Dengan meningkatnya *ignition delay* maka akan memberikan waktu bahan bakar dan udara bercampur, sehingga campuran akan menjadi lebih homogen. Semakin homogen campuran bahan bakar maka proses pembakaran akan lebih baik sehingga NOx akan meningkat. Semakin kecil nilai *exhaust gas recirculation* (dalam % / udara hasil pembuangan pada langkah 4 yang digunakan kembali pada langkah 1) maka kadar oksigen dalam ruang bakar akan meningkat. Meningkatnya kadar oksigen dalam ruang bakar akan membuat pembakaran menjadi lebih sempurna, sehingga NOx akan meningkat. Grafik pada Gambar 4.3 menunjukkan perbandingan metode GA dan PSO dalam mencari nilai maksimum NOx. Pada grafik tersebut didapat bahwa GA memiliki hasil yang lebih baik dibanding PSO, dimana GA mencapai nilai maksimum pada iterasi 174 dengan nilai 122,7142983059 g/kg bahan bakar, sedangkan PSO mencapai nilai maksimum pada iterasi 21 dengan nilai 121,650087115767 g/kg bahan bakar. Hasil tersebut menunjukkan bahwa PSO terjebak dalam lokal optimum sehingga hasil yang didapat kurang optimum.

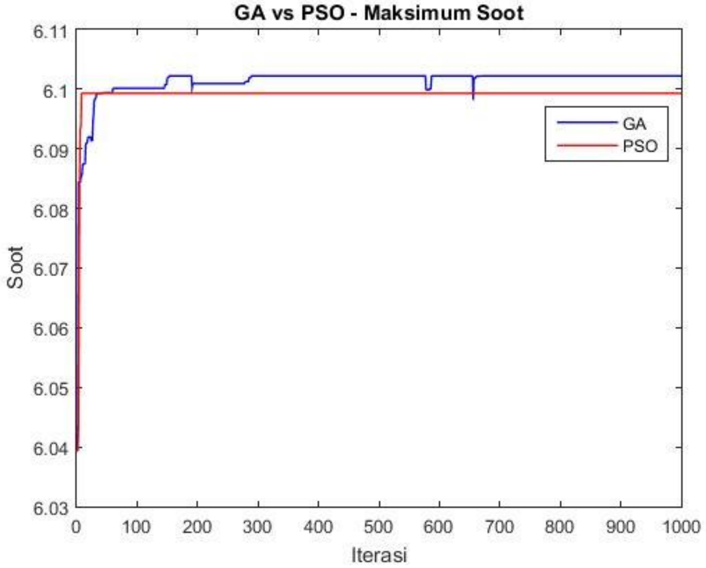


Gambar 4.4 Perbandingan nilai minimum NOx GA dengan PSO

Dalam pencarian nilai minimum NOx maka perlu dilakukn pengaturan parameter yang berkebalikan dengan pada saat mencari nilai maksimum NOx dimana dibutuhkan nilai variabel CR, FIP, SOI yang kecil dan nilai EGR yang besar sehingga pembakaran menjadi tidak sempurna. Pembakaran yang tidak sempurna akan membuat NOx menurun.Grafik pada Gambar 4.4 menunjukkan perbandingan metode GA dan PSO dalam mencari nilai minimum NOx. Pada grafik tersebut didapat bahwa GA dan PSO memiliki nilai minimum yang sama yaitu 2,15595212870506 g/kg bahan bakar, namun PSO mampu mencapai nilai minimum lebih cepat yaitu pada iterasi 4, sedangkan GA mencapai nilai maksimum pada iterasi 54.

- ***Soot***

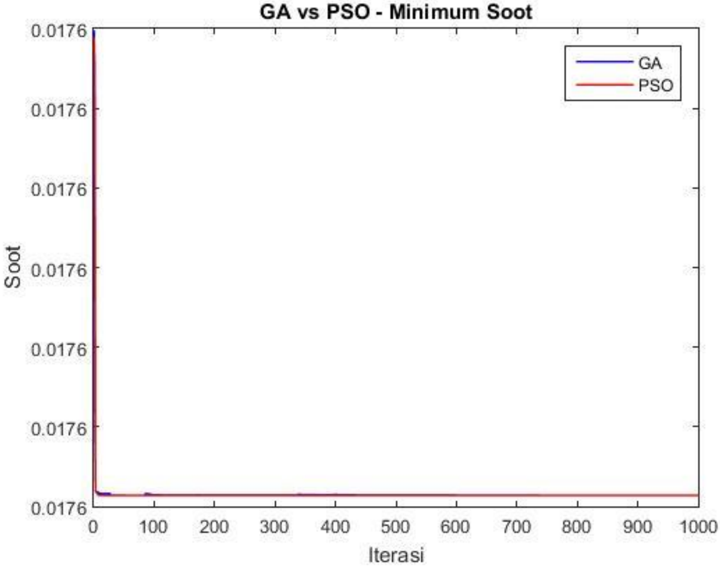
Soot merupakan arang dihasilkan pada ruang bakar pada saat proses pembakaran (langkah 3) yang terjadi tidak sempurna.



Gambar 4.5 Perbandingan nilai maksimum *soot* GA dengan PSO

Dalam pencarian nilai maksimum *soot* diperlukan pengaturan parameter CR, FIP, SOI, dan EGR yang sesuai dimana untuk mendapatkan nilai *soot* yang tinggi maka diperlukan nilai parameter CR, FIP, dan SOI yang rendah dan nilai parameter EGR yang tinggi. Semakin kecil *compression ratio* maka *soot* akan meningkat. Hal tersebut disebabkan karena dengan menurunnya *compression ratio* maka tekanan dan temperatur dalam ruang bakar akan menurun sehingga pembakaran yang terjadi semakin kurang baik. Semakin kecil *fuel injection pressure* (pemasukan bahan bakar/langkah 2) akan membuat evaporasi menjadi lebih buruk dan atomisasi

bahan bakar akan berlangsung lambat. Hal tersebut akan membuat campuran antara udara dan bahan bakar pada ruang bakar semakin tidak homogen(heterogen). Semakin tidak homogen campuran bahan bakar maka proses pembakaran akan lebih buruk sehingga kadar *soot* akan meningkat. Semakin kecil *start of injection* (dalam °bTDC/pada saat sebelum langkah 2 berakhir bahan bakar mulai diinjeksikan) maka akan menurunkan *ignition delay*. Dengan menurunnya *ignition delay* maka waktu bahan bakar dan udara bercampur akan lebih singkat, sehingga campuran akan menjadi tidak homogen. Semakin tidak homogen campuran bahan bakar maka proses pembakaran akan lebih uruk sehingga *soot* akan meningkat. Semakin besar nilai *exhaust gas recirculation* (dalam % / udara hasil pembuangan pada langkah 4 yang digunakan kembali pada langkah 1) maka kadar oksigen dalam ruang bakar akan menurun. Menurunnya kadar oksigen dalam ruang bakar akan membuat pembakaran menjadi kurang sempurna, sehingga *soot* akan meningkat.Grafik pada Gambar 4.5 menunjukkan perbandingan metode GA dan PSO dalam mencari nilai maksimum *soot.* Pada grafik tersebut didapat bahwa GA memiliki hasil yang lebih baik dibanding PSO, dimana GA mencapai nilai maksimum pada iterasi 153 dengan nilai 6,10223646466959 g/kg bahan bakar, sedangkan PSO mencapai nilai maksimum pada iterasi 8 dengan nilai 6,09931856331581 g/kg bahan bakar. Hasil tersebut menunjukkan bahwa PSO terjebak dalam lokal optimum sehingga hasil yang didapat kurang optimum. Pada grafik terlihat bahwa metode GA mengalami fluktuatif dalam mencari nilai optimum. Hal tersebut terjadi karena individu terbaik mengalami mutasi, sehingga hasil yang didapat fluktuatif.



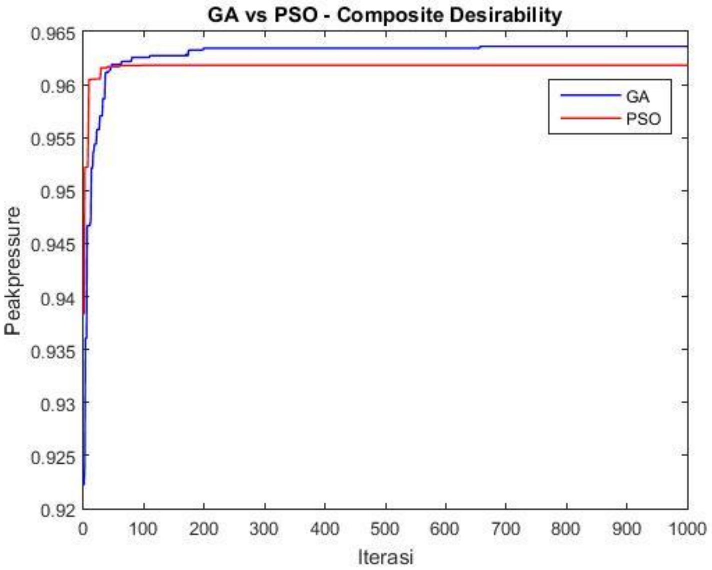
Gambar 4.6 Perbandingan nilai minimum *soot* GA dengan PSO

Dalam pencarian nilai minimum *soot* maka perlu dilakukan pengaturan parameter yang berkebalikan dengan pada saat mencari nilai maksimum *soot* dimana dibutuhkan nilai variabel CR, FIP, SOI yang besar dan nilai EGR yang keci sehingga pembakaran menjadi lebih sempurna. Pembakaran yang sempurna akan membuat *soot* menurun Grafik pada Gambar 4.6 menunjukkan perbandingan metode GA dan PSO dalam mencari nilai minimum *soot.* Pada grafik tersebut didapat bahwa PSO mampu mencapai nilai minimum pada iterasi 13 dengan nilai 0,0175887762775909 g/kg bahan bakar, sedangkan GA mencapa nilai minimum pada iterasi 65 dengan nilai 0,0175887762781884 g/kg bahan bakar. Hasil tersebut menyatakan bahwa PSO mampu mencapai nilai *NOx* maksimum lebih cepat daripada GA dengan selisih hasil yang sangat kecil atau mendekati sama.

Dari hasil pencarian nilai minimum dan maksimum dari respon *peakpressure,* NOx, dan *soot* dapat disimpulkan bahwa GA mampu mencapai hasil yang lebih optimum dibanding dengan PSO, namun untuk menuju nilai optimum GA jauh lebih lambat dibanding dengan PSO. Hal tersebut sesuai dengan teori dimana GA mampu mencapai nilai yang lebih optimum dibanding PSO karena GA dapat keluar dari lokal optimum sedanganKan PSO tidak. GA mencapai nilai optimum lebih lambat dibandingkan dengan PSO karena dalam mencari nilai optimum GA sangat bergantung pada probabilitas.

4.4 Hasil Optimasi dengan metode GA dan PSO

Untuk mendapatkan nilai yang optimum diperlukan pengaturan nilai variabel CR, FIP, SOI, dan EGR yang cocok. Nilai minimum dan maksimum yang didapat dari langkah sebelumnya digunakan untuk mencari nilai *composite desirability* sebagai acuan untuk menentukan nilai optimum dari multivariabel respon. Berikut hasil yang didapat setelah dilakukan optimasi



Gambar 4.7 Perbandingan nilai *composite desirability* GA dengan PSO

Grafik pada Gambar 4.7 menunjukkan perbandingan metode GA dan PSO dalam mencari nilai *composite desirability* . Pada grafik tersebut didapat bahwa metode GA mencapai nilai *composite desirabilty* lebih baik daripada metode PSO dengan nilai 0,963604251124843, sedangkan PSO 0,961827275838624. Hal tersebut kemungkinan disebabkan karena PSO terjebak pada lokal optimum, sehingga tidak bisa mencapai nilai global optimum.

Tabel 4.4 Perbandingan optimasi dengan metode GA, PSO,dan RSM

GA								
CR	FIP	SOI	EGR	PP	NOX	SOOT	CDA	ITERASI
15,9	1396,48	8,471	25	113,488	11,89	0,1787	0,964	662
PSO								
CR	FIP	SOI	EGR	PP	NOX	SOOT	CDA	ITERASI
16,5	1400	6,376	25	113,473	12,08	0,1996	0,962	97
RSM								
CR	FIP	SOI	EGR	PP	NOX	SOOT	CDA	
14,25	1153,15	13,69	16,91	113,5	15,8	0,2	0,95	

Hasil yang didapat sesuai dengan hipotesa awal, dimana GA mencapai hasil yang lebih baik daripada metode PSO dan RSM, namun GA membutuhkan waktu yang jauh lebih lama untuk mencapai nilai optimum. GA lebih optimum dibanding dengan PSO dan RSM disebabkan karena GA mampu keluar dari jebakan lokal optimum, sedangkan PSO dan RSM tidak bisa. Metode PSO sendiri mencapai nilai optimum lebih tinggi dibandingkan metode RSM, karena dalam penelitian kali ini jumlah individu yang digunakan adalah 100 individu, sehingga dapat memperkecil kemungkinan terjebak dalam lokal optimum karena dengan jumlah individu yang lebih banyak pencarian akan lebih menyebar. Metode GA mencapai nilai optimum jauh lebih lama dibandingkan dengan PSO dikarenakan pencarian nilai optimum pada GA sangat tergantung pada probabilitas.

Halaman ini sengaja dikosongkan

BAB 5
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil yang telah didapatkan dari penelitian kali ini dengan judul “Perbandingan Optimasi Multirespon *Direct Injenction Diesel Engine* dengan Metode *Back Propagation Neural Network – Genetic Algorithm* (BPNN-GA) dan *Back Propagation Neural Network – Particle Swarm Optimization* (BPNN-PSO)”, dapat diambil kesimpulan sebagai berikut :

1. Pemodelan BPNN yang cocok untuk prediksi *peakpressure* digunakan model BPNN dengan layer sebanyak 3, jumlah neuron 9, dan fungsi aktivasi tansig dengan nilai MSE 0,014689. Untuk prediksi NOx digunakan model BPNN dengan layer sebanyak 3, jumlah neuron 9, dan fungsi aktivasi tansig dengan nilai MSE 0,020889. Untuk prediksi *soot* digunakan model BPNN dengan layer sebanyak 5, jumlah neuron 10, dan fungsi aktivasi satlin dengan nilai MSE 0,008878.

2. Variabel desain optimum yang didapat dengan menggunakan metode BPNN-GA lebih optimum dibandingkan dengan menggunakan metode BPNN-PSO, yaitu dengan nilai CR 15,9 , FIP 1396.48 bar, SOI 8,471˚bTDC, dan EGR 25%, didapat output *peakpressure* sebesar 113,488 bar, NOx 11,89 g/kg bahan bakar, *soot* 0,1787 g/kg bahan bakar dan *composite desirability* 0,964.

5.2 Saran

Adapun saran yang dapat diberikan setelah melakukan penelitian ini adalah sebagai berikut :

1. Metode optimasi multirespon dalam penelitian ini menggunakan *Back Propagation Neural Network–*

Genetic Algorithm (BPNN-GA) dan *Back Propagation Neural Network – Particle Swarm Optimization* (BPNN-PSO). Pada penelitian selanjutnya , dapat dilakukan studi untuk melakukan optimasi dengan menggunakan metode-metode optimasi yang lain sebagai pembanding.

2. Untuk penelitian selanjutnya mengenai optimasi menggunakan BPNN sebaiknya menggunakan jumlah data yang cukup besar, agar prediksi yang dihasilkan oleh jaringan BPNN cukup akurat. Untuk penelitian dengan permasalahan yang beda dapat digunakan arsitektur BPNN yang berbeda juga, karena arsitektur yang cocok untuk masing-masing permasalahan dapat berbeda.

3. Untuk penelitian selanjutnya mengenai optimasi menggunakan metode GA dan PSO, dapat digunakan nilai parameter yang berbeda, karena parameter yang cocok untuk metode GA dan PSO bisa berbeda-bedea untuk setiap permasalahan.

DAFTAR PUSTAKA

Avinash Kumar Agrawal, Shrawan Kumar Singh, Shailendra Sinha, Mritunjay Kumar Shukla. 2003. *Effect of EGR on the Exhaust Gas Temperature and Exhaust.* India : Indian Istitute of Technology

Hiro Hiroyasu, Haiyan Miao, Tomo Hiroyasu, Mitunori Miki, Jiro Kamiura, Sinya Watanabe. 2003. *Genetic Algorithms Optimization of Diesel Engine Emissions and Fuel Efficiency with Air Swirl, EGR, Injection Timing, and Multiple Injections.* Japan : Society of Automotive Engineers of Japan

Kusumadewi, Sri. 2004. *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB & EXCELLINK*. Yogyakarta : Graha Ilmu

Siang, Jong Jek. 2009. *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan MATLAB.* Yogyakarta : ANDI

M. Manohar, Jomy Joseph, T. Selvaraj, D. Sivakumar. 2013. *Application of Desirability-Function and RSM to Optimise the Multi-Objective while Turning Inconel 718 Using Coated Carbide Tools.* India : Inderscience Enterprises Ltd

Michael J. Moran, Howard N. Shapiro. 2006. *Fundamentals of Engineering Thermodynamics*. England : John Wiley and Sons

O Obodeh, C.I Ajuwa. 2009. *Evaluation of Artificial Neural Network Performance in Predicting Diesel Engine NO_x Emissions.* Nigeria : Ambrose Alli University

Prabhakara Rao Ganji, Kiran Prasad Chintala, V.R K Raju, Srinivasa Rao Surapaneni. 2016. *Parametric Study and Optimization Using RSM of DI Diesel Engine for Lower Emissions*. Brazil : Springer

Prashanth K. Karra, Song-Charng Kong. 2010. *Diesel Engine Emissions Reduction Using Particle Swarm Optimization*. USA : Iowa Stae University

Priambodo, Ir. Bambang. 1995. *Operasi dan Pemeliharaan Mesin Diesel*. Jakarta : Erlangga

Primartha, Rifkie. 2018. *Belajar Machine Learning Teori dan Praktik*. Bandung : Informatika

Santosa, Budi. 2017. *Pengantar Metaheuristik Implementasi dengan Matlab*. Surabaya : ITS TEKNO SAINS

<http://neuroph.sourceforge.net/tutorials/Perceptron.html>(diakses 10 Maret 2019)

<http://psychmuseum.uwgb.org/wp-content/uploads/2016/10/neuron_anatomy.jpg>(diakses 7 Maret 2019)

<https://www.celtictuning.fi/palvelut/egr-poisto-ohitus>(diakses 21 juni 2019)

<https://www.mechanicalbooster.com/2017/12/what-is-compression-ratio-petrol-diesel-engine.html>(diakses 21 juni 2019)

<http://www.swarthmore.edu/NatSci/mzucker1/opencv-2.4.10-docs/modules/ml/doc/neural_networks.html>(diakses 10 Maret 2019)

<https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_parent_selection.htm> (diakses 17 Maret 2019)

Halaman ini sengaja dikosongkan

LAMPIRAN

Lampiran 1 : *Mean Square Error* (MSE) masing-masing model BPNN untuk prediksi *peakpressure*

layer	neuron	fungsi	mse	layer	neuron	fungsi	mse
1	2	hardlim	485,7051	1	6	satlin	1,264072
1	2	hardlims	489,7219	1	6	logsig	2,614617
1	2	purelin	15,02056	1	6	tansig	0,568134
1	2	satlin	4,441116	1	7	hardlim	210,2557
1	2	logsig	3,640307	1	7	hardlims	109,4777
1	2	tansig	2,374558	1	7	purelin	15,04605
1	3	hardlim	255,0043	1	7	satlin	0,043396
1	3	hardlims	359,2435	1	7	logsig	2,687687
1	3	purelin	15,05317	1	7	tansig	0,558461
1	3	satlin	1,448089	1	8	hardlim	155,3417
1	3	logsig	2,214831	1	8	hardlims	143,8354
1	3	tansig	2,105433	1	8	purelin	15,10275
1	4	hardlim	163,4714	1	8	satlin	0,031785
1	4	hardlims	231,0825	1	8	logsig	3,823071
1	4	purelin	14,99193	1	8	tansig	0,636798
1	4	satlin	3,198945	1	9	hardlim	81,56225
1	4	logsig	1,140123	1	9	hardlims	156,1236
1	4	tansig	0,887776	1	9	purelin	14,97501
1	5	hardlim	275,2135	1	9	satlin	0,19346
1	5	hardlims	354,9869	1	9	logsig	3,089313
1	5	purelin	14,98593	1	9	tansig	0,725832
1	5	satlin	0,120178	1	10	hardlim	103,7004
1	5	logsig	3,791456	1	10	hardlims	62,81952
1	5	tansig	0,324221	1	10	purelin	15,09721
1	6	hardlim	231,1641	1	10	satlin	0,025763
1	6	hardlims	208,869	1	10	logsig	3,205615
1	6	purelin	15,06287	1	10	tansig	0,578425

layer	neuron	fungsi	mse	layer	neuron	fungsi	mse
2	2	hardlim	503,6353	2	6	satlin	0,101066
2	2	hardlims	450,944	2	6	logsig	0,214706
2	2	purelin	15,10568	2	6	tansig	0,457501
2	2	satlin	51,0399	2	7	hardlim	265,6338
2	2	logsig	3,18128	2	7	hardlims	210,3896
2	2	tansig	1,022738	2	7	purelin	15,1329
2	3	hardlim	395,0879	2	7	satlin	0,109962
2	3	hardlims	475,3342	2	7	logsig	0,171636
2	3	purelin	15,10675	2	7	tansig	0,030392
2	3	satlin	6,359663	2	8	hardlim	335,7775
2	3	logsig	1,476442	2	8	hardlims	266,1124
2	3	tansig	0,412687	2	8	purelin	15,11945
2	4	hardlim	495,6283	2	8	satlin	0,070361
2	4	hardlims	320,5386	2	8	logsig	0,140801
2	4	purelin	15,24415	2	8	tansig	0,024858
2	4	satlin	1,113044	2	9	hardlim	298,91
2	4	logsig	0,701707	2	9	hardlims	147,2638
2	4	tansig	0,101498	2	9	purelin	15,09181
2	5	hardlim	271,8475	2	9	satlin	29,6032
2	5	hardlims	306,2818	2	9	logsig	0,114639
2	5	purelin	15,09906	2	9	tansig	0,020564
2	5	satlin	1,516127	2	10	hardlim	324,9103
2	5	logsig	0,33023	2	10	hardlims	135,4246
2	5	tansig	0,528744	2	10	purelin	15,1179
2	6	hardlim	103,2041	2	10	satlin	6,08299
2	6	hardlims	327,8661	2	10	logsig	0,100186
2	6	purelin	15,23502	2	10	tansig	0,384815

layer	neuron	fungsi	mse	layer	neuron	fungsi	mse
3	2	hardlim	479,522	3	6	satlin	0,442271
3	2	hardlims	502,7903	3	6	logsig	0,203947
3	2	purelin	15,26867	3	6	tansig	0,435258
3	2	satlin	61,27005	3	7	hardlim	416,4161
3	2	logsig	2,516073	3	7	hardlims	232,5069
3	2	tansig	2,911341	3	7	purelin	15,12298
3	3	hardlim	483,9168	3	7	satlin	0,102158
3	3	hardlims	498,4642	3	7	logsig	0,163062
3	3	purelin	15,08499	3	7	tansig	0,403589
3	3	satlin	17,2348	3	8	hardlim	120,3363
3	3	logsig	1,222079	3	8	hardlims	244,1167
3	3	tansig	2,749929	3	8	purelin	15,10462
3	4	hardlim	345,7836	3	8	satlin	0,107607
3	4	hardlims	370,0598	3	8	logsig	0,126548
3	4	purelin	15,09467	3	8	tansig	0,361685
3	4	satlin	2,453494	3	9	hardlim	141,3795
3	4	logsig	0,629052	3	9	hardlims	362,2095
3	4	tansig	0,05835	3	9	purelin	15,12646
3	5	hardlim	407,1803	3	9	satlin	0,067233
3	5	hardlims	435,9009	3	9	logsig	3,252928
3	5	purelin	15,86151	3	9	tansig	0,014689
3	5	satlin	2,607826	3	10	hardlim	201,1684
3	5	logsig	0,276877	3	10	hardlims	179,9575
3	5	tansig	0,770715	3	10	purelin	15,13146
3	6	hardlim	422,7376	3	10	satlin	0,075472
3	6	hardlims	441,6947	3	10	logsig	0,085535
3	6	purelin	15,19396	3	10	tansig	0,315141

layer	neuron	fungsi	mse	layer	neuron	fungsi	mse
4	2	hardlim	496,4271	4	6	satlin	0,120233
4	2	hardlims	500,8496	4	6	logsig	6,37048
4	2	purelin	15,12613	4	6	tansig	0,421808
4	2	satlin	11,7961	4	7	hardlim	356,4782
4	2	logsig	2,721883	4	7	hardlims	391,6846
4	2	tansig	0,824812	4	7	purelin	15,12196
4	3	hardlim	435,1036	4	7	satlin	5,635848
4	3	hardlims	489,0895	4	7	logsig	0,150132
4	3	purelin	15,14085	4	7	tansig	0,37344
4	3	satlin	505,3204	4	8	hardlim	385,029
4	3	logsig	1,28388	4	8	hardlims	248,1175
4	3	tansig	2,598063	4	8	purelin	15,12868
4	4	hardlim	396,7925	4	8	satlin	0,081473
4	4	hardlims	474,2876	4	8	logsig	0,119778
4	4	purelin	15,18481	4	8	tansig	0,334983
4	4	satlin	505,2076	4	9	hardlim	251,7247
4	4	logsig	0,573196	4	9	hardlims	453,0474
4	4	tansig	0,064948	4	9	purelin	15,10965
4	5	hardlim	482,4116	4	9	satlin	0,065531
4	5	hardlims	387,1496	4	9	logsig	0,097434
4	5	purelin	15,12088	4	9	tansig	0,311198
4	5	satlin	0,162619	4	10	hardlim	284,3513
4	5	logsig	0,295544	4	10	hardlims	355,1519
4	5	tansig	0,569627	4	10	purelin	15,12235
4	6	hardlim	297,0495	4	10	satlin	0,06719
4	6	hardlims	371,297	4	10	logsig	0,08102
4	6	purelin	15,12319	4	10	tansig	0,283962

layer	neuron	fungsi	mse	layer	neuron	fungsi	mse
5	2	hardlim	496,2445	5	6	satlin	1,082607
5	2	hardlims	498,3093	5	6	logsig	5,175189
5	2	purelin	15,45174	5	6	tansig	0,419601
5	2	satlin	5,658661	5	7	hardlim	377,7638
5	2	logsig	2,627127	5	7	hardlims	489,6288
5	2	tansig	3,549914	5	7	purelin	15,14941
5	3	hardlim	336,6718	5	7	satlin	0,095914
5	3	hardlims	254,3723	5	7	logsig	0,150517
5	3	purelin	15,10994	5	7	tansig	0,375292
5	3	satlin	7,030135	5	8	hardlim	468,0929
5	3	logsig	1,199841	5	8	hardlims	282,0917
5	3	tansig	5,962396	5	8	purelin	15,13362
5	4	hardlim	458,7286	5	8	satlin	10,07336
5	4	hardlims	488,2725	5	8	logsig	0,11864
5	4	purelin	15,09854	5	8	tansig	0,323523
5	4	satlin	11,38894	5	9	hardlim	298,628
5	4	logsig	0,622273	5	9	hardlims	327,7425
5	4	tansig	0,926276	5	9	purelin	15,11854
5	5	hardlim	458,8962	5	9	satlin	35,6785
5	5	hardlims	433,2906	5	9	logsig	0,094973
5	5	purelin	15,16255	5	9	tansig	0,292431
5	5	satlin	2,166055	5	10	hardlim	200,7628
5	5	logsig	5,850545	5	10	hardlims	190,8222
5	5	tansig	0,485181	5	10	purelin	15,11549
5	6	hardlim	333,9806	5	10	satlin	2,706838
5	6	hardlims	504,6288	5	10	logsig	0,078678
5	6	purelin	15,18548	5	10	tansig	0,265507

Lampiran 2 : *Mean Square Error* (MSE) masing-masing model BPNN untuk prediksi kadar NOx

layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
1	2	hardlim	451,1187	1	6	satlin	1,803863
1	2	hardlims	351,2955	1	6	logsig	0,723487
1	2	purelin	101,2169	1	6	tansig	0,142571
1	2	satlin	11,39878	1	7	hardlim	279,6141
1	2	logsig	6,904919	1	7	hardlims	81,50029
1	2	tansig	1,135675	1	7	purelin	101,396
1	3	hardlim	493,5676	1	7	satlin	0,620573
1	3	hardlims	200,764	1	7	logsig	2,686893
1	3	purelin	101,1609	1	7	tansig	0,733126
1	3	satlin	10,46024	1	8	hardlim	167,684
1	3	logsig	1,734434	1	8	hardlims	224,688
1	3	tansig	4,619603	1	8	purelin	101,2118
1	4	hardlim	254,5353	1	8	satlin	0,347782
1	4	hardlims	155,3978	1	8	logsig	2,837617
1	4	purelin	101,2157	1	8	tansig	0,59978
1	4	satlin	0,443647	1	9	hardlim	107,156
1	4	logsig	1,144748	1	9	hardlims	133,5461
1	4	tansig	0,37406	1	9	purelin	101,2339
1	5	hardlim	237,3647	1	9	satlin	0,584148
1	5	hardlims	377,2391	1	9	logsig	3,854321
1	5	purelin	101,3981	1	9	tansig	0,790846
1	5	satlin	1,940787	1	10	hardlim	151,7193
1	5	logsig	0,719327	1	10	hardlims	94,80358
1	5	tansig	0,318893	1	10	purelin	101,2481
1	6	hardlim	359,4331	1	10	satlin	0,127137
1	6	hardlims	131,3591	1	10	logsig	0,30676
1	6	purelin	101,1101	1	10	tansig	0,753679

layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
2	2	hardlim	483,3987	2	6	satlin	0,109221
2	2	hardlims	413,1122	2	6	logsig	0,336829
2	2	purelin	101,2278	2	6	tansig	0,058773
2	2	satlin	11,28079	2	7	hardlim	254,4378
2	2	logsig	3,789415	2	7	hardlims	111,8033
2	2	tansig	7,39259	2	7	purelin	101,2186
2	3	hardlim	322,5636	2	7	satlin	0,09766
2	3	hardlims	472,803	2	7	logsig	4,329335
2	3	purelin	101,1292	2	7	tansig	0,443297
2	3	satlin	500,322	2	8	hardlim	281,9553
2	3	logsig	1,310306	2	8	hardlims	209,1811
2	3	tansig	0,337892	2	8	purelin	101,2243
2	4	hardlim	493,1062	2	8	satlin	0,02443
2	4	hardlims	491,9845	2	8	logsig	0,222172
2	4	purelin	101,1166	2	8	tansig	0,038733
2	4	satlin	60,60081	2	9	hardlim	180,3072
2	4	logsig	0,668123	2	9	hardlims	128,9957
2	4	tansig	0,700358	2	9	purelin	101,3025
2	5	hardlim	256,5059	2	9	satlin	15,04947
2	5	hardlims	402,722	2	9	logsig	0,178466
2	5	purelin	101,2274	2	9	tansig	0,410241
2	5	satlin	0,32594	2	10	hardlim	197,991
2	5	logsig	0,581129	2	10	hardlims	224,4545
2	5	tansig	0,580879	2	10	purelin	101,2423
2	6	hardlim	404,575	2	10	satlin	0,088978
2	6	hardlims	440,2983	2	10	logsig	0,158576
2	6	purelin	101,2545	2	10	tansig	0,369613

layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
3	2	hardlim	497,0408	3	6	satlin	0,175045
3	2	hardlims	493,2623	3	6	logsig	0,28081
3	2	purelin	101,1581	3	6	tansig	0,040771
3	2	satlin	3,8388	3	7	hardlim	292,073
3	2	logsig	3,46111	3	7	hardlims	205,2718
3	2	tansig	11,56562	3	7	purelin	101,2078
3	3	hardlim	341,8646	3	7	satlin	13,66893
3	3	hardlims	476,6355	3	7	logsig	1,145351
3	3	purelin	101,2089	3	7	tansig	0,418492
3	3	satlin	500,3954	3	8	hardlim	164,5597
3	3	logsig	1,154587	3	8	hardlims	346,4308
3	3	tansig	0,200521	3	8	purelin	101,2363
3	4	hardlim	498,8038	3	8	satlin	0,083714
3	4	hardlims	378,9324	3	8	logsig	3,542673
3	4	purelin	101,4936	3	8	tansig	0,025802
3	4	satlin	14,60293	3	9	hardlim	229,0398
3	4	logsig	6,373954	3	9	hardlims	159,3168
3	4	tansig	1,127781	3	9	purelin	101,2491
3	5	hardlim	255,9809	3	9	satlin	0,0811
3	5	hardlims	394,1292	3	9	logsig	0,14911
3	5	purelin	101,2508	3	9	tansig	0,020889
3	5	satlin	27,78984	3	10	hardlim	290,1887
3	5	logsig	0,393485	3	10	hardlims	144,8652
3	5	tansig	0,601611	3	10	purelin	101,2459
3	6	hardlim	399,6557	3	10	satlin	0,393098
3	6	hardlims	325,2692	3	10	logsig	0,12275
3	6	purelin	101,2494	3	10	tansig	0,317146

layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
4	2	hardlim	500,1317	4	6	satlin	0,109063
4	2	hardlims	500,236	4	6	logsig	0,258339
4	2	purelin	101,3367	4	6	tansig	0,423194
4	2	satlin	20,22018	4	7	hardlim	337,3493
4	2	logsig	2,103914	4	7	hardlims	406,3049
4	2	tansig	6,790822	4	7	purelin	101,2558
4	3	hardlim	500,2352	4	7	satlin	0,617281
4	3	hardlims	495,8491	4	7	logsig	0,20341
4	3	purelin	101,2546	4	7	tansig	0,419052
4	3	satlin	72,48134	4	8	hardlim	423,2331
4	3	logsig	1,207894	4	8	hardlims	318,1074
4	3	tansig	0,136176	4	8	purelin	101,2529
4	4	hardlim	472,3145	4	8	satlin	0,079335
4	4	hardlims	483,4687	4	8	logsig	0,158549
4	4	purelin	101,266	4	8	tansig	0,340561
4	4	satlin	1,048071	4	9	hardlim	235,5616
4	4	logsig	6,037876	4	9	hardlims	263,4186
4	4	tansig	0,787888	4	9	purelin	101,2628
4	5	hardlim	389,4543	4	9	satlin	4,541269
4	5	hardlims	495,3335	4	9	logsig	0,132821
4	5	purelin	101,2347	4	9	tansig	0,315509
4	5	satlin	7,22808	4	10	hardlim	403,0017
4	5	logsig	0,402744	4	10	hardlims	309,07
4	5	tansig	0,516484	4	10	purelin	101,2513
4	6	hardlim	307,1704	4	10	satlin	0,068702
4	6	hardlims	333,9519	4	10	logsig	0,109159
4	6	purelin	101,21	4	10	tansig	0,288611

layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
5	2	hardlim	445,5309	5	6	satlin	14,02061
5	2	hardlims	495,6262	5	6	logsig	0,251936
5	2	purelin	101,162	5	6	tansig	0,44951
5	2	satlin	35,23327	5	7	hardlim	425,6816
5	2	logsig	2,053909	5	7	hardlims	399,1259
5	2	tansig	11,05504	5	7	purelin	101,2527
5	3	hardlim	492,358	5	7	satlin	5,164443
5	3	hardlims	449,472	5	7	logsig	0,188287
5	3	purelin	101,2771	5	7	tansig	19,16458
5	3	satlin	9,829921	5	8	hardlim	418,4474
5	3	logsig	15,36063	5	8	hardlims	474,5529
5	3	tansig	1,836568	5	8	purelin	101,2374
5	4	hardlim	422,017	5	8	satlin	28,62359
5	4	hardlims	489,1281	5	8	logsig	4,163538
5	4	purelin	101,2315	5	8	tansig	0,318835
5	4	satlin	3,908026	5	9	hardlim	218,5131
5	4	logsig	8,189129	5	9	hardlims	462,5337
5	4	tansig	1,651165	5	9	purelin	101,2548
5	5	hardlim	404,9324	5	9	satlin	0,088163
5	5	hardlims	468,6114	5	9	logsig	0,1208
5	5	purelin	101,2437	5	9	tansig	0,29106
5	5	satlin	19,22599	5	10	hardlim	400,6964
5	5	logsig	0,409091	5	10	hardlims	390,0511
5	5	tansig	3,172015	5	10	purelin	101,2465
5	6	hardlim	446,0553	5	10	satlin	0,194374
5	6	hardlims	479,4938	5	10	logsig	0,100237
5	6	purelin	101,2286	5	10	tansig	0,268177

Lampiran 3 : *Mean Square Error* (MSE) masing-masing model BPNN untuk prediksi kadar *soot*

layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
1	2	hardlim	1,392386	1	6	satlin	0,067002
1	2	hardlims	0,941378	1	6	logsig	0,270446
1	2	purelin	0,479295	1	6	tansig	0,104649
1	2	satlin	0,170918	1	7	hardlim	0,818224
1	2	logsig	0,582198	1	7	hardlims	0,53506
1	2	tansig	0,208161	1	7	purelin	0,431936
1	3	hardlim	1,224072	1	7	satlin	0,061258
1	3	hardlims	1,320559	1	7	logsig	0,24248
1	3	purelin	0,457946	1	7	tansig	0,095349
1	3	satlin	0,12391	1	8	hardlim	0,530636
1	3	logsig	0,438911	1	8	hardlims	0,544137
1	3	tansig	0,16136	1	8	purelin	0,429404
1	4	hardlim	0,929486	1	8	satlin	0,050476
1	4	hardlims	1,168273	1	8	logsig	0,220656
1	4	purelin	0,446788	1	8	tansig	0,088107
1	4	satlin	0,10027	1	9	hardlim	0,810179
1	4	logsig	0,359988	1	9	hardlims	0,46821
1	4	tansig	0,134635	1	9	purelin	0,427422
1	5	hardlim	0,954514	1	9	satlin	0,046339
1	5	hardlims	0,686728	1	9	logsig	0,202164
1	5	purelin	0,43993	1	9	tansig	0,08221
1	5	satlin	0,100646	1	10	hardlim	0,164227
1	5	logsig	0,306823	1	10	hardlims	0,330538
1	5	tansig	0,117077	1	10	purelin	0,42583
1	6	hardlim	1,297923	1	10	satlin	0,041725
1	6	hardlims	1,32046	1	10	logsig	0,188719
1	6	purelin	0,435287	1	10	tansig	0,077323

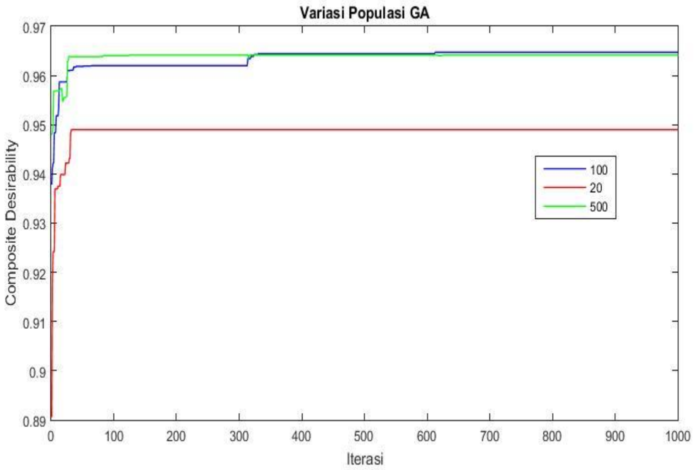
layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
2	2	hardlim	1,691409	2	6	satlin	0,036246
2	2	hardlims	1,784519	2	6	logsig	0,237725
2	2	purelin	0,483003	2	6	tansig	0,053731
2	2	satlin	0,208677	2	7	hardlim	1,542072
2	2	logsig	0,747902	2	7	hardlims	1,35935
2	2	tansig	0,168897	2	7	purelin	0,425036
2	3	hardlim	1,057303	2	7	satlin	0,030494
2	3	hardlims	1,140111	2	7	logsig	0,198703
2	3	purelin	0,455297	2	7	tansig	0,044364
2	3	satlin	0,093651	2	8	hardlim	1,093401
2	3	logsig	0,505982	2	8	hardlims	0,627218
2	3	tansig	0,114352	2	8	purelin	0,422521
2	4	hardlim	1,542377	2	8	satlin	0,02496
2	4	hardlims	1,329576	2	8	logsig	0,167765
2	4	purelin	0,441607	2	8	tansig	0,037322
2	4	satlin	0,068945	2	9	hardlim	1,038201
2	4	logsig	0,373961	2	9	hardlims	1,204275
2	4	tansig	0,08506	2	9	purelin	0,420648
2	5	hardlim	1,205065	2	9	satlin	0,02123
2	5	hardlims	1,250751	2	9	logsig	0,146784
2	5	purelin	0,433643	2	9	tansig	0,032529
2	5	satlin	0,052669	2	10	hardlim	0,999966
2	5	logsig	0,292567	2	10	hardlims	0,246035
2	5	tansig	0,06713	2	10	purelin	0,419211
2	6	hardlim	1,606424	2	10	satlin	0,01958
2	6	hardlims	1,570233	2	10	logsig	0,126961
2	6	purelin	0,428535	2	10	tansig	0,027802

layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
3	2	hardlim	1,649347	3	6	satlin	0,033337
3	2	hardlims	1,331406	3	6	logsig	0,228882
3	2	purelin	0,485343	3	6	tansig	0,04037
3	2	satlin	0,162996	3	7	hardlim	1,208334
3	2	logsig	0,881341	3	7	hardlims	1,168189
3	2	tansig	0,152533	3	7	purelin	0,423186
3	3	hardlim	1,056584	3	7	satlin	0,022434
3	3	hardlims	1,609986	3	7	logsig	0,184565
3	3	purelin	0,454362	3	7	tansig	0,032877
3	3	satlin	0,084112	3	8	hardlim	1,44815
3	3	logsig	0,551446	3	8	hardlims	1,372405
3	3	tansig	0,097176	3	8	purelin	0,420826
3	4	hardlim	1,456561	3	8	satlin	0,019337
3	4	hardlims	1,598924	3	8	logsig	0,153013
3	4	purelin	0,439769	3	8	tansig	0,027149
3	4	satlin	0,055042	3	9	hardlim	1,608355
3	4	logsig	0,388181	3	9	hardlims	1,173998
3	4	tansig	0,069189	3	9	purelin	0,821997
3	5	hardlim	1,402877	3	9	satlin	0,016058
3	5	hardlims	1,266099	3	9	logsig	0,129244
3	5	purelin	0,431612	3	9	tansig	0,022277
3	5	satlin	0,042703	3	10	hardlim	0,876634
3	5	logsig	0,292257	3	10	hardlims	0,63349
3	5	tansig	0,051421	3	10	purelin	0,417811
3	6	hardlim	1,672408	3	10	satlin	0,013024
3	6	hardlims	1,693355	3	10	logsig	0,110861
3	6	purelin	0,426553	3	10	tansig	0,018805

layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
4	2	hardlim	1,793913	4	6	satlin	0,025582
4	2	hardlims	1,650576	4	6	logsig	0,224565
4	2	purelin	0,486914	4	6	tansig	0,03415
4	2	satlin	1,795639	4	7	hardlim	1,359608
4	2	logsig	0,979715	4	7	hardlims	1,428675
4	2	tansig	0,143007	4	7	purelin	0,42234
4	3	hardlim	1,249177	4	7	satlin	0,020685
4	3	hardlims	0,965238	4	7	logsig	0,178991
4	3	purelin	0,453912	4	7	tansig	0,027378
4	3	satlin	1,794133	4	8	hardlim	0,453323
4	3	logsig	0,58778	4	8	hardlims	1,255745
4	3	tansig	1,794133	4	8	purelin	0,420069
4	4	hardlim	1,509474	4	8	satlin	0,015826
4	4	hardlims	1,693343	4	8	logsig	0,145915
4	4	purelin	0,438847	4	8	tansig	0,021405
4	4	satlin	0,052667	4	9	hardlim	1,302646
4	4	logsig	0,398179	4	9	hardlims	0,963317
4	4	tansig	0,059844	4	9	purelin	0,418431
4	5	hardlim	1,492597	4	9	satlin	0,012233
4	5	hardlims	1,508259	4	9	logsig	0,121926
4	5	purelin	0,430624	4	9	tansig	0,018368
4	5	satlin	0,043041	4	10	hardlim	0,921916
4	5	logsig	0,29233	4	10	hardlims	0,787335
4	5	tansig	0,044077	4	10	purelin	0,41721
4	6	hardlim	1,750793	4	10	satlin	0,010951
4	6	hardlims	1,087149	4	10	logsig	0,103857
4	6	purelin	0,42562	4	10	tansig	0,014646

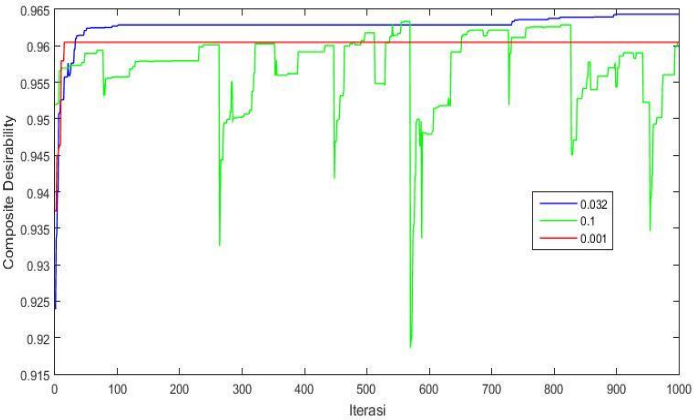
layer	neuron	fungsi	MSE	layer	neuron	fungsi	MSE
5	2	hardlim	1,735579	5	6	satlin	0,025624
5	2	hardlims	1,710579	5	6	logsig	0,221507
5	2	purelin	0,488033	5	6	tansig	0,029965
5	2	satlin	1,79508	5	7	hardlim	1,690888
5	2	logsig	1,79411	5	7	hardlims	1,627423
5	2	tansig	0,135762	5	7	purelin	0,421857
5	3	hardlim	1,234495	5	7	satlin	0,016451
5	3	hardlims	1,328171	5	7	logsig	0,174215
5	3	purelin	0,453656	5	7	tansig	0,024505
5	3	satlin	0,080001	5	8	hardlim	0,516136
5	3	logsig	1,793094	5	8	hardlims	1,105805
5	3	tansig	0,080645	5	8	purelin	0,419642
5	4	hardlim	1,622155	5	8	satlin	0,013062
5	4	hardlims	1,611718	5	8	logsig	0,141594
5	4	purelin	0,438297	5	8	tansig	0,01921
5	4	satlin	0,047765	5	9	hardlim	1,060836
5	4	logsig	1,792691	5	9	hardlims	1,394674
5	4	tansig	0,054192	5	9	purelin	0,688716
5	5	hardlim	1,720285	5	9	satlin	0,010723
5	5	hardlims	1,271437	5	9	logsig	0,117782
5	5	purelin	0,430044	5	9	tansig	0,015034
5	5	satlin	0,030434	5	10	hardlim	0,579705
5	5	logsig	0,292176	5	10	hardlims	1,247488
5	5	tansig	0,039625	5	10	purelin	0,575922
5	6	hardlim	1,638784	5	10	satlin	0,008878
5	6	hardlims	1,336393	5	10	logsig	0,099781
5	6	purelin	0,42508	5	10	tansig	0,012378

Lampiran 4 : Pengaruh Variasi Populasi GA terhadap Nilai Optimum



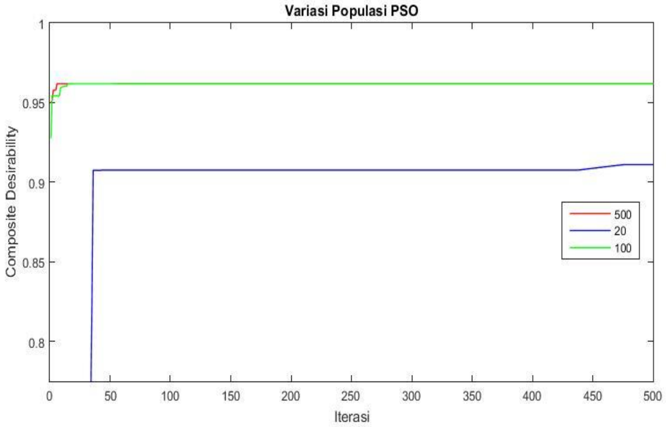
Dari grafik diatas dapat dilihat bahwa semakin tinggi jumlah populasi akan memberikan nilai yang lebih optimum, namun pada saat populasi lebih besar sama dengan 100, hasil yang didapat cukup konvergen sehingga nilai populasi 100 dipilih karena akan memerlukan waktu yang lebih singkat untuk mencapai nilai optimum.

Lampiran 5 : Pengaruh Variasi Probabilitas Mutasi GA Terhadap Nilai Optimum



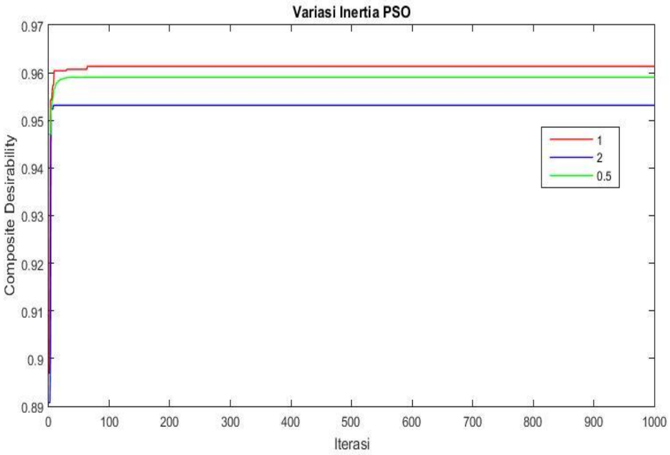
Dari grafik diatas didapat bahwa nilai probabilitas mutasi sangat berpengaruh signifikan terhadap nilai optimum. Apabila probabilitas mutasi terlalu kecil maka hasil yang didapat kurang optimum, namun apabila probabilitas mutasi semakin besar, hasil yang didapat akan sangat fluktuatif.

Lampiran 6 : Pengaruh Variasi Populasi PSO terhadap Nilai Optimum



Dari grafik diatas dapat dilihat bahwa semakin tinggi jumlah populasi akan memberikan nilai yang lebih optimum, namun pada saat populasi lebih besar sama dengan 100, hasil yang didapat cukup konvergen sehingga nilai populasi 100 dipilih karena akan memerlukan waktu yang lebih singkat untuk mencapai nilai optimum

Lampiran 7 : Pengaruh Variasi *Inertia* PSO Terhadap Nilai Optimum



Dari grafik diatas didapat bahwa nilai inertia cukup berpengaruh terhadap nilai optimum. Apabila nilai inertia terlalu besar atau terlalu kecil, maka hasil yang didapat kurang optimum.

Halaman ini sengaja dikosongkan

BIODATA PENULIS

Penulis lahir di Jember pada tanggal 15 Maret 1997, merupakan anak bungsu dari 2 bersaudara dari pasangan Bapak Elkaf dan Ibu Nur. Penulis telah menempuh pendidikan di SDN Kepatihan 6 Jember, SMPN 1 Jember, SMAN 1 Jember, dan masuk di ITS dengan jalur SNMPTN. Apabila pembaca ingin berkorespondensi dengan penulis dapat melalui email : andrea.gozzybastian@gmail.com atau telp : 085331994494.