

Desain PID Controller

Untuk Temperature Heating Furnace Menggunakan Metode PSO

Machrus Ali
Hidayatul Nurohmah
Rukslin
Dwi Ajiatmo
Muhammad Agil Haikal

Editor: Khoiruddin, S.Sos, M.E.



Desain PID Controller

Untuk Temperature Heating Furnace Menggunakan Metode PSO

Furnace merupakan sebuah alat untuk memanaskan bahan, minyak, dan sebagainya. Furnace biasanya menggunakan bahan bakar gas, batubara, dan minyak. Temperatur merupakan parameter utama yang perlu dikontrol agar tetap stabil, presisi, dan tentunya meningkatkan efisiensi terhadap bahan bakar. PID controller sudah terbukti dan banyak digunakan pada industri, namun untuk menentukan gain dari nilai PID tersebut masih kurang tepat. Maka dengan buku ini untuk mensimulasikan nilai gain PID yang lebih baik dengan menggunakan kecerdasan buatan. Metode kecerdasan buatan tersebut adalah Particle Swarm Optimization (PSO) dan Adaptive Neuro Fuzzy Inference System (ANFIS). Dalam buku ini dijelaskan secara mendalam tentang Desain PID Controller untuk Temperature Heating Furnace Menggunakan Metode PSO. Adapun topik yang dibahas dalam buku ini yaitu furnace, furnace pada industri, kontroler, Fuzzy Inference System (FIS), cara kerja ANFIS, PSO (Particle Swarm Optimization), parameter PSO, metode implementasi PID-PSO, analisa dan PID-PSO, simpulan PID-PSO, dan ANFIS-PSO.

Machrus Ali, dkk.

Desain PID Controller Untuk Temperatur Heating Furnace Menggunakan Metode PSO



CV. Global Aksara Pers
Anggota IKAPI, Jawa Timur, 2021.
No. 282/JTI/2021
Jl. Wonocolo Utara V/18 Surabaya
Telp: +628977416123 / +628573269334
www.globalaksarapers.com

ISBN: 978-623-462-166-2



9 786234 621662

**Desain PID Controller untuk
Temperatur Heating Furnace
Menggunakan Metode PSO**

Machrus Ali
Hidayatul Nurohma
Rukslin
Dwi Ajiatmo
Muhammad Agil Haikal

**UNDANG-UNDANG REPUBLIK INDONESIA
NOMOR 28 TAHUN 2014
TENTANG HAK CIPTA**

**PASAL 113
KETENTUAN PIDANA
SANKSI PELANGGARAN**

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah).
3. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf a, huruf b, huruf e, dan/atau huruf g untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 4 (empat) tahun dan/atau pidana denda paling banyak Rp1.000.000.000,00 (satu miliar rupiah).
4. Setiap Orang yang memenuhi unsur sebagaimana dimaksud pada ayat (3) yang dilakukan dalam bentuk pembajakan, dipidana dengan pidana penjara paling lama 10 (sepuluh) tahun dan/atau pidana denda paling banyak Rp4.000.000.000,00 (empat miliar rupiah).

Machrus Ali
Hidayatul Nurohmah
Rukslin
Dwi Ajiatmo
Muhammad Agil Haikal

**Desain PID Controller untuk
Temperatur Heating Furnace
Menggunakan Metode PSO**



Desain PID Controller untuk *Temperatur Heating Furnace* Menggunakan Metode PSO

*Diterbitkan pertama kali dalam bahasa Indonesia
oleh Penerbit Global Aksara Pers*

ISBN: **978-623-462-166-2**

viii + 92 hal; 14,8 x 21 cm

Cetakan Pertama, Oktober 2022

copyright © Oktober 2022 Global Aksara Pers

- Penulis** : Machrus Ali
Hidayatul Nurohmah
Rukslin
Dwi Ajiatmo
Muhammad Agil Haikal
- Penyunting** : Khoiruddin, S. Sos., ME
- Desain Sampul** : Hamim Thohari Mahfudhillah, M.Pd
- Layouter** : Ilil Ni'matul M

Hak Cipta dilindungi undang-undang.

Dilarang memperbanyak sebagian atau seluruh isi buku ini dengan bentuk dan cara apapun tanpa izin tertulis dari penulis dan penerbit.

Diterbitkan oleh:



CV. Global Aksara Pers
Anggota IKAPI, Jawa Timur, 2021,
No. 282/JTI/2021
Jl. Wonocolo Utara V/18 Surabaya
+628977416123/+628573269334
globalaksarapers@gmail.com

MOTTO

“Ikatlah ilmu dengan dengan menulisnya”
(Rasulullah, SAW)

Ilmu adalah buruan dan tulisan adalah ikatannya, ikatlah buruanmu dengan tali yang kuat. Termasuk kebodohan kalau engkau memburu kijang, setelah itu kamu tinggalkan terlepas begitu saja

(Imam Asy Syafi’i, Rahimahullah)

PRAKATA

Furnace merupakan sebuah alat untuk memanaskan bahan, minyak, dan sebagainya. *Furnace* biasanya menggunakan bahan bakar gas, batubara, dan minyak. Temperatur merupakan parameter utama yang perlu dikontrol agar tetap stabil, presisi, dan tentunya meningkatkan efisiensi terhadap bahan bakar. *PID controller* sudah terbukti dan banyak digunakan pada industri, namun untuk menentukan gain dari nilai *PID* tersebut masih kurang tepat, Maka dengan buku ini untuk mensimulasikan nilai gain *PID* yang lebih baik dengan menggunakan kecerdasan buatan. Metode kecerdasan buatan tersebut adalah *Particle Swarm Optimization* (PSO) dan *Adaptive Neuro Fuzzy Inference System* (ANFIS).

Penyusun,

DAFTAR ISI

Motto	v
Prakata	vi
Daftar Isi	vii
BAB I Furnace	1
BAB II Furnace pada Industri	3
A. Ilustrasi Furnace	3
B. Fungsi Furnace	5
C. Sistem Kontrol Cascade Ifurnance	6
D. Tipe-tipe Furnace	8
E. Burner	10
F. Sensor dan Instrumen	10
BAB III Kontroler	18
A. Proportional Integral Derivatif (PID).....	18
B. Parameter Sistem Kontrol	20
C. Komponen PID.....	22
BAB IV Fuzzy Inference System (FIS)	35
BAB V Cara Kerja ANFIS	39
A. Proses Training data ANFIS.....	42
B. Defuzzyfikasi.....	43
BAB VI PSO (Particle Swarm Optimzation)	44
A. Algoritma	45
B. Parameter Selection	47
C. Neighbourhoods and Topologies	47
D. Inner Workings.....	48
E. Konvergensi.....	49

F. Mekanisme Adaptif	51
G. Varian.....	51
H. Hibridisasi.....	52
I. Meringankan Konvergensi Dini.....	52
J. Penyederhanaan.....	53
K. Optimalisasi Multi-Objektif.....	54
L. Biner, Diskrit, dan Kombinatorial.....	54
BAB VII Parameter PSO.....	56
A. Parameter Particle Swarm Optimization (PSO)	56
B. Pseudocode.....	56
C. Setting PID Controller	57
BAB VIII Metode.....	58
A. Flow Chart.....	58
B. Sensor Thermocouple.....	59
C. Pemodelan Sistem.....	60
D. Parameter Kontrol PID.....	62
BAB IX Implementasi PID-PSO	65
BAB X Analisa PID-PSO.....	68
BAB XI Simpulan PID-PSO.....	78
BAB XII ANFIS-PSO	79
A. Desain ANFIS-PSO	79
B. Hasil Respon Simulink.....	80
C. Simpulan ANFIS-PSO	84
Daftar Pustaka.....	85
Index.....	89
Biografi Penulis	91

BAB I

FURNACE

Furnace merupakan sebuah peralatan yang digunakan untuk memanaskan bahan atau mengubah bentuknya (misalnya *rolling*/penggulungan, *annealing process*) atau merubah sifat-sifatnya (perlakuan panas), biasa disebut juga sebagai *oven*. Transfer energi pada tungku terjadi dalam tahapan pembangkitan energi panas oleh *element heater* yang energinya disuplai dari energi listrik ataupun menggunakan gas.

Kontrol proses menjadi perhatian penting dalam pengembangan proses yang lebih fleksibel dan lebih kompleks untuk pembuatan produk agar bernilai tambah tinggi. Salah satu kontrol yang kompleks dalam suatu proses adalah kontrol *tuning*. Kontrol *tuning* merupakan kunci utama untuk mengoperasikan sebuah kontrol industri. Proses *tuning* berperan dalam memastikan bahwa kinerja sebuah sistem sudah memenuhi tujuan operasi, sehingga perlu optimasi dan pengujian tersendiri pada bagian sistem kontrol PID untuk mendapatkan hasil pengontrolan temperatur yang lebih stabil dan efisien dibanding dengan pengontrolan konvensional. Kontrol cerdas berbasis *Artificial Intelligent* sudah banyak berkembang untuk memperbaiki kontrol konvensional. Salah satunya yaitu penelitian yang pernah dilakukan oleh YV Pava Kumar, Arvapalli Rajesh, Cascaded PID

Controller Design for Heating Furnace Temperature Control. Pada penelitian tersebut, membahas mengenai PID dengan metode cascade dan dibandingkan dengan metode lain[1][2]. Oleh sebab itu, pada tugas penelitian ini penulis akan mendesain model temperatur pada *heating furnace* menggunakan kontrol PID yang dituning dengan *Artificial Intelegent (AI)* dengan metode *Particle Swarm Optimization (PSO)* untuk mengontrol temperatur *heating furnace* agar lebih stabil dan efisiensi. Hasil performance dari temperatur yang dikontrol menggunakan PID dan dituning dengan PSO diharapkan dapat memiliki tingkat efisiensi sebuah sistem dan kestabilan temperatur yang lebih baik sehingga *heating furnace* dapat bekerja dengan temperatur yang di inginkan sesuai *set point* dan dapat mengurangi konsumsi gas maupun udara yang merupakan sebagai bahan utama dari *furnace*[3][4].

BAB II

FURNACE PADA INDUSTRI

A. Ilustrasi Furnace



Gambar 2.1 Ilustrasi *furnace* pada industri
(sumber : industrialheating.com)

Seperti yang kita ketahui bersama, *furnace* adalah alat yang digunakan untuk menaikkan suhu pada suatu objek dengan menggunakan panas dari pembakaran bahan bakar. Bahan bakar yang digunakan adalah bahan bakar cair dan bahan bakar gas, yang dicampur oleh burner dan menghasilkan pembakaran. Pemanasan dilakukan dengan melewati bahan bakar melalui tabung atau pipa yang diatur sedemikian rupa di dalam tungku, perpindahan panas terjadi melalui tiga cara yaitu konveksi, konduksi dan radiasi.

Tahun demi tahun, *furnace* telah meningkat baik dalam proses dan peralatan. Bahan bakar yang digunakan juga berubah. Jika sebelum tahun 1800 mayoritas penggunaan bahan bakar adalah arang, setelah tahun itu kokas, gas, dan listrik mulai berperan. Tentunya dengan berkembangnya *furnace*, terjadi peningkatan efisiensi proses. Idealnya, *furnace* harus memanaskan material sebanyak mungkin ke suhu yang seragam dengan bahan bakar sesedikit mungkin. Kunci untuk operasi *furnace* yang efisien adalah pada saat pembakaran bahan bakar yang sempurna dengan udara berlebih yang minim. *Furnace* beroperasi pada efisiensi yang relatif rendah (sekitar 7%) dibandingkan dengan perangkat pembakaran lain seperti boiler (dengan efisiensi lebih dari 90%), hal ini disebabkan suhu operasi yang tinggi dalam ruang bakar *furnace*.^[5]

Karena emisi bahan bakar bersentuhan langsung dengan bahan baku, pemilihan jenis bahan bakar menjadi sangat penting. Misalnya, beberapa bahan tidak akan mentolerir belerang dalam bahan bakar. Bahan bakar padat akan menciptakan partikel yang mengganggu bahan baku yang diumpankan ke *furnace*.

Untuk itu, hampir semua tungku menggunakan bahan bakar cair, bahan bakar gas atau listrik sebagai sumber bahan bakarnya. Panas yg dihasilkan oleh pembakaran bahan bakar, dipindahkan dengan 3 cara, yaitu:

- 1) Radiasi, adalah perpindahan panas transfer panas. Ketika tabung menerima panas burner api, panas yang dinyatakan ditransmisikan dengan radiasi dan tubenya yang disebut tabung radiasi. Panas ditransmisikan ke bagian radiasi kotak api adalah 60% - 70%.

- 2) Konveksi, merupakan panas yang ditransfer melalui konveksi diperoleh dari panas yang terkandung dalam asap, yang mengalir dari perapian sebelum keluar dari cerobong asap. Kontak antara gas buang dan dinding pipa akan menyebabkan perpindahan panas. Tabung yang menerima panas secara konveksi disebut tabung konveksi.
- 3) Konduksi, perpindahan panas secara konduksi merupakan proses perpindahan panas antara tube dengan aliran fluida yang ada didalamnya.
- Pada umumnya, pemanas terdiri atas:
- Benda-benda penerima panas
 - Bahan pemanas (sumber panas)
 - Media yang akan diubah temperaturnya karena adanya penyerapan panas.

B. Fungsi Furnace

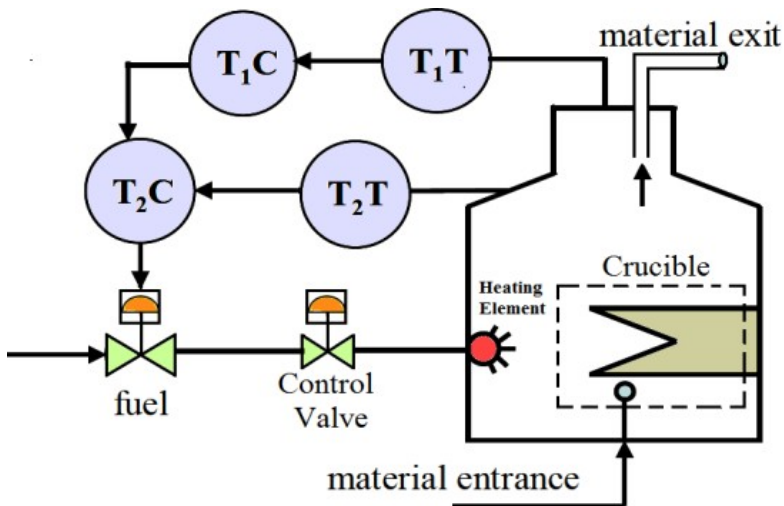


Gambar 2.2 *furnace* untuk proses pembakaran

Fungsi *furnace* adalah untuk menghasilkan energi panas dan mencapai pembakaran. Alat ini bekerja dengan cara menggunakan komposisi gas dan udara yang tercampur dengan seimbang. Bahan bakar yang digunakan berupa bahan bakar cair dan gas menyala yang terdapat di dalam burner. Temperatur yang dihasilkan oleh *furnace* bisa mencapai lebih dari 1000°C.

C. Sistem Kontrol Cascade Ifurnance

Konstituen dari sistem kontrol suhu tungku ditunjukkan pada gambar.1. Sedangkan T1C adalah pengontrol utama, T2C adalah pengontrol sekunder, T1T mewakili suhu pengukuran untuk ekspor mentah bahan, dan T2T merupakan ukuran suhu tungku perapian. Operasi dasar sistem kontrol suhu tungku adalah sebagai berikut.



Gambar 2.3 Kontrol bertingkat sistem suhu tungku

Output dari kontroler primer (T1C) diberikan sebagai set point ke kontroler sekunder (T2C). T2C mengontrol aliran bahan bakar. Dalam proses pemanasan, bahan yang ditempatkan di dalam wadah dipanaskan sampai suhu tertentu dari pintu masuk sampai keluar. Dari ruang bakar bahan bakar hingga ekspor bahan mentah, ada tiga komponen kapasitas dalam suhu Sistem. Mereka adalah tungku, perapian dan bahan baku yang dipanaskan. Gangguan beban dan di sisi bahan baku adalah gangguan sistem di satu sisi dan di sisi lain gangguan di sisi pembakaran, seperti bahan bakar, tekanan, dan aliran udara.

Dalam sistem kontrol suhu tungku, objek utama adalah bahan baku dan objek sekunder adalah perapian di tungku. Variabel kontrol utama adalah suhu bahan baku di ekspor dan variabel sub kontrol adalah suhu perapian di tungku. Gangguan utama adalah aliran bahan baku dan suhu bahan di inlet. Gangguan sekunder adalah tekanan bahan bakar sebelum katup, nilai panas campuran bahan bakar-udara yang dipasok ke tungku perapian dll.

Salah satu tindakan pencegahan penting yang harus diambil dalam sistem kontrol PID berjenjang adalah bahwa, dalam pemilihan parameter desain, pastikan bahwa tidak ada masalah pencocokan konstanta waktu loop utama dan sub loop sehingga operasi yang aman mencegah resonansi dapat dimungkinkan.

D. Tipe-tipe Furnace

Furnace memiliki beberapa *type*, antara lain yaitu:

a) Silinder Tipe vertikal

Bentuk tungku ini adalah tegak dan terbuat dari baja, tabung penerima panas radiasi diatur sedemikian rupa secara vertikal dan setiap sambungan selang digunakan sebagai bentuk U di bagian bawah sehingga nyala api sejajar dengan susunan tabung di dapur. Jenis tungku ini dapat dirancang tanpa zona konveksi, tergantung pada kebutuhan pemanasan. Jika dirancang dengan zona konveksi, tabung yang dapat digunakan adalah *bare tube* dan *finned tube*, tetapi pada umumnya *bare tube* sering digunakan karena dapat mempercepat perpindahan panas secara konveksi[5].

b) *Type Box*

Furnace ini memiliki bentuk box atau kotak atau sering juga disebut tipe kabin, area pemanasan secara radiasi dan konveksi dipisahkan oleh dinding. Umumnya tipe ini digunakan pada pemanasan destilasi minyak bumi dan destilasi *vacum*. Tube pada area radiasi tersusun horizontal sepanjang sisi *vertical wall*, burner dipasang pada sisi *furnace*.

Pemanas jenis ini terdiri dari ruang pembakaran dan ruang konveksi yang dipisahkan oleh dinding penyekat yang disebut dengan *bridge wall*. Tube yang digunakan pada *furnace* ini ialah tube yang terbuat dari material *high chrome content alloy* dengan 25% cr yang tahan terhadap

temperatur yang tinggi tetapi harga dari jenis furnace ini memiliki harga yang cukup tinggi[6].

c) *Visbreaker Charge Furnace*

Furnace jenis ini juga hadir dalam bentuk kotak, namun posisi burner berada di lantai yang dipanaskan dan sering menggunakan *flux* dan ada juga yang menggunakan *multiple pass*. Zona radiasi meliputi dinding samping dan sekat dan biasanya tidak dilengkapi dengan zona konveksi. Karena area burner terletak di bawah lantai *furnace*, lantai *furnace* dirancang hingga 6 kaki di atas tanah. Untuk hasil terbaik dan kontrol proses yang mudah, tungku ini dilengkapi dengan peredam cerobong asap, peniup uap, pengukur *draft*, pengukur suhu, dan termokopel[7]. Pemisahan kental biasanya dilakukan untuk pemanasan suhu tinggi seperti pemanasan minyak bumi fraksinasi berat dengan suhu operasi sekitar 930 ° F.

d) *Type high pressure box*

Jenis ini biasanya digunakan pada pengolahan lanjutan fraksi minyak bumi seperti *reforming* dan *hydrocracking unit*. *Furnace* ini beroperasi dalam tekanan tinggi yaitu 2200 psi dan suhu 700°F.. Area burner terletak di lantai lantai *furnace*, pembakaran terletak di dua tempat, yaitu proses pembakaran besar di tengah, sementara getaran kecil terletak di tepi. Untuk menghindari kehilangan panas, orang menggunakan insulasi bata tahan api berkekuatan tinggi.

E. Burner

Burner adalah suatu alat yang mensuplai atau mengatur bahan bakar dan udara yang akan dibakar. Salah satu kegunaannya adalah untuk mencampur udara dan bahan bakar untuk menciptakan pembakaran yang baik dan meningkatkan efisiensi pembakaran. Secara umum bagian burner terbagi menjadi dua bagian yaitu pilot burner dan main burner. *Pilot burner* digunakan sebagai titik awal proses penyalaan dari pemantik api (*ignition*), kemudian setelah penyalaan akan dilanjutkan dengan main burner hingga memiliki keadaan dimana burner mengkonsumsi lebih banyak gas dan udara.[6] Gambar 2.3 merupakan contoh dari bentuk burner[8].



Gambar 2.4 Ilustrasi Burner

F. Sensor dan instrumen

Untuk memenuhi kebutuhan alat dan mesin dalam pengujian dan pengukuran, diperlukanlah perlengkapan yang mendukung kelengkapan alat dan mesin tersebut. Mengenai hal tersebut diperlukan alat sensor dan

instrumen yang menjadi faktor pendukung alat tersebut. Berikut deskripsi tentang sensor dan instrumen :

1. Sensor

Sensor adalah perangkat yang digunakan untuk mendeteksi perubahan lingkungan fisik atau kimia. Besaran keluaran sensor yang diubah menjadi besaran listrik disebut transduser. Sensor yang digunakan untuk mengukur perubahan yang sangat kecil harus sangat sensitif.[7]

a) Klasifikasi sensor

Sensor memiliki macam-macam jenis dan memiliki fungsi masing-masing sesuai dengan kegunaannya. Sensor dibagi menjadi empat kategori utama, yaitu :

1) Sensor Analog

Sensor analog adalah sensor yang menghasilkan sinyal keluaran yang kontinu atau berkelanjutan. Sinyal keluaran kontinu yang dihasilkan oleh sensor analog ini sebanding dengan nilai yang terukur. Berbagai parameter analog ini meliputi suhu, tegangan, tekanan, gerakan, dan sebagainya. Contoh sensor analog ini antara lain akselerometer, sensor kecepatan, sensor tekanan, sensor optik, level transmitter, dan sensor suhu.

2) Sensor Digital

Sensor Digital adalah sensor yang menghasilkan sinyal keluaran diskrit. Sinyal diskrit akan non-kontinu dengan waktu dan dapat direpresentasikan dalam "bit". Sensor digital

biasanya terdiri dari sensor, kabel, dan pemancar. Sinyal yang diukur ditampilkan dalam format digital. Keluaran digital dapat berupa logika 1 atau logika 0 (*ON* atau *OFF*). Sinyal fisik yang diterima diubah menjadi sinyal digital oleh sensor itu sendiri tanpa komponen eksternal. Kabel digunakan untuk transmisi jarak jauh. Contoh dari sensor digital tersebut antara lain akselerometer digital (*digital accelerometers*), *limit switch*, *proximity sensor*, *optic sensor*, sensor cahaya[9].

3) Sensor Aktif

Sensor aktif adalah jenis sensor yang membutuhkan sumber daya eksternal untuk beroperasi. Sifat fisik sensor aktif berubah dalam kaitannya dengan efek eksternal yang mereka berikan. Sensor aktif ini juga dikenal sebagai sensor yang dihasilkan sendiri. Misalnya yaitu sensor *proximity*, sensor *proximity* harus memiliki tegangan input terlebih dahulu untuk memberikan *feedback* ke *controller*[10].

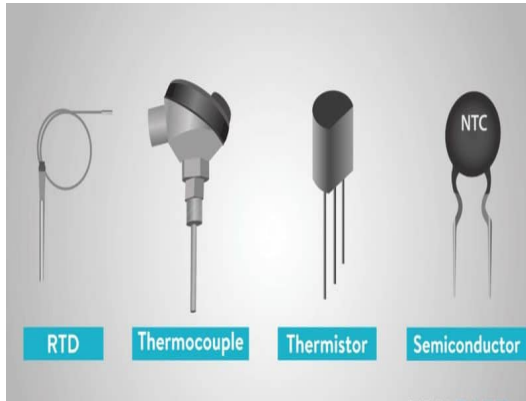
4) Sensor pasif

Sensor pasif adalah sensor yang dapat menghasilkan sinyal keluaran tanpa memerlukan sumber daya eksternal. Misalnya termokopel (*thermocouple*) yang menghasilkan nilai tegangan tergantung dari panas atau temperatur yang diterimanya.

b) Jenis-jenis sensor

Setiap kategori memiliki jenis sensor yang berbeda sebagai berikut:

1) **Sensor Suhu**



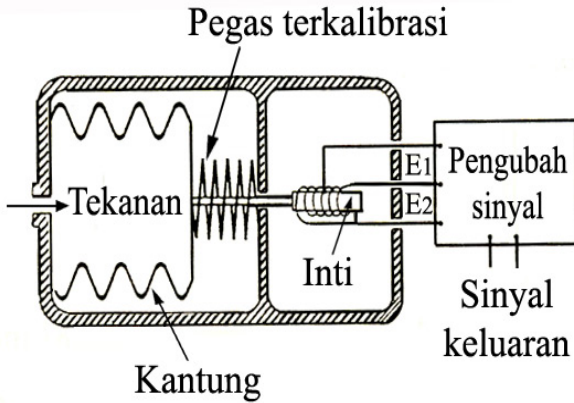
Gambar 2.5 Sensor suhu

(sumber : <https://pintarelektro.com/sensor-suhu/>)

Sensor suhu digunakan untuk mendeteksi suhu dan energi panas dan mengubahnya menjadi sinyal listrik (berupa tegangan atau arus). Ada beberapa jenis sensor suhu yang digunakan.[11]

- Thermometer
- Thermocouple
- Resistance Temperature Detector (RTD)
- Thermistor Temperature Sensor
- Semiconductor Temperature Sensor
- Vibrating Wire Temperature Sensor

2) Sensor Tekanan



Gambar 2.6 Prinsip kerja sensor tekanan
(sumber : *blogspot.com*)

Perubahan tekanan dalam kantong mengubah posisi inti kumparan yang mengakibatkan perubahan induksi magnet pada kumparan. Kumparan yang digunakan adalah kumparan CT (*Center-Tap*), jadi jika inti memiliki ketidaksesuaian, induktansi pada satu kumparan meningkat sedangkan induktansi pada kumparan lainnya berkurang. Kemudian konverter sinyal memiliki efek mengubah induktansi yang muncul pada kumparan menjadi tegangan yang proporsional[12].

3) Sensor Proximity

Sensor *proximity* adalah sensor yang digunakan untuk mendeteksi suatu objek berdasarkan jaraknya dari sensor. Sensor jarak ini akan mendeteksi objek pada jarak yang cukup

dekat, mulai dari 1 mm hingga beberapa sentimeter dari sensor.



Gambar 2.7 Sensor *proximity*

Sensor ini biasa diimplementasikan di pabrik industri, perkantoran, dunia robot dan tempat lainnya. Berdasarkan penggunaannya, sensor *proximity* merupakan sensor yang mampu mendeteksi keberadaan suatu obyek logam maupun non logam tanpa menggunakan kontak fisik.

4) Sensor level

Peran utama sensor level adalah untuk mengukur level atau ketinggian material yang berbeda seperti padat, cair, dan gas. Sensor level diklasifikasikan menjadi beberapa jenis, antara lain yaitu :

- Sensor Level Laser
- *Float* Sensor

- Sensor Level *Capacitive*
- Sensor Level *Resistive*
- Sensor Level *Ultrasonic*
- Sensor Level *Hydrostatic*
- Sensor Level *Optical*
- Sensor Level *Electromagnetic*

Sensor ini banyak digunakan di kapal, wadah, tangki air, tempat tangki bahan bakar, dll untuk mengetahui tingkat level suatu benda cair pada tanki atau wadah tertentu.

5) Sensor Posisi

Sensor posisi atau *Position sensor* merupakan sensor yang menentukan perpindahan dan posisi (seperti linier dan rotasi). Pada dasarnya, sensor posisi tersedia dalam berbagai jenis, yaitu :

- Sensor Posisi Optik
- Sensor Posisi Linier
- Sensor Posisi *Rotary*
- Sensor Posisi Induktif
- Sensor Posisi Kapasitif
- Sensor Posisi *Fiber-Optic*
- Sensor Posisi Ultrasonik

Sensor posisi digunakan dalam aplikasi domestik dan industri seperti penutupan / pembukaan pintu, *monitoring valve*, pengontrol motor, *throttle* untuk mengontrol aliran bahan bakar atau daya ke mesin.

2. Instrumentasi pengukuran

Instrumen adalah alat dan perangkat yang digunakan untuk pengukuran dan kontrol dalam sistem

yang lebih besar dan lebih kompleks. Hal inilah yang membuat ilmu instrumentasi menjadi sangat penting dalam sebuah industri. Secara umum, sebuah instrumentasi memiliki fungsi utama sebagai alat ukur, alat analisis, alat kontrol dan alat perekam untuk suatu instrumen (dalam hal ini juga dikenal sebagai data *trend*).

Alat ukur sebagai alat ukur meliputi alat survey/statistik, alat ukur suhu dan lain-lain. Contoh instrumentasi sebagai alat analisis banyak dijumpai dalam bidang kimia, seperti pH meter, konduktivitas meter, *turbidity meter*, dll. Sementara itu, contoh instrumentasi sebagai alat kontrol banyak dijumpai pada elektronika, industri, dan pabrik. Sistem pengukuran, analisis dan kontrol alat ini dapat dilakukan secara manual (hasilnya dibaca dan dicatat secara manual), tetapi juga dapat dilakukan secara otomatis oleh komputer. Untuk kategori yang terakhir ini, instrumentasi tidak dapat dipisahkan dari bidang elektronika dan dari instrumentasi itu sendiri.

BAB III

KONTROLER

A. Proportional Integral Derivatif (PID)

Pengendali PID (*Proportional Integral Derivative*) adalah pengendali mekanisme umpan balik yang biasa digunakan dalam sistem kendali industri. Kontrol PID secara terus menerus menghitung nilai kesalahan sebagai perbedaan antara setpoint yang diinginkan dan variabel proses yang diukur. Kontroler mencoba meminimalkan nilai kesalahan setiap kali dengan mengatur variabel kontrol, seperti posisi katup kontrol, catu daya peredam atau pemanas, ke nilai baru yang ditentukan oleh jumlahan. Berikut merupakan rumus dari PID *controller* :

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

Dengan K_p , K_i , dan K_d semuanya positif, menandakan koefisien untuk term *proporsional*, *integral*, dan *derivatif*, secara berurutan (atau *P*, *I*, dan *D*). Pada model ini,

- *P* bertanggung jawab untuk nilai kesalahan saat ini. Contohnya, jika nilai kesalahan besar dan positif, maka keluaran kontrol juga besar dan positif.
- *I* bertanggung jawab untuk nilai kesalahan sebelumnya. Contoh, jika keluaran saat ini kurang besar, maka kesalahan akan terhitung terus menerus, dan kontroler akan merespon dengan keluaran lebih tinggi.

- D bertanggung jawab untuk kemungkinan nilai kesalahan akan terjadi, berdasarkan pada rate perubahan tiap waktu.

Karena pengontrol PID hanya bergantung pada variabel proses yang dapat diukur dan bukan mengenai pengetahuan prosesnya, maka pengontrol PID dapat digunakan secara luas[2][13]. Dengan menyesuaikan ketiga parameter model, kontroler PID dapat memenuhi kebutuhan proses. Respon pengontrol dapat dijelaskan dengan cara bereaksi terhadap kesalahan, kelebihan setpoint, dan osilasi sistem. penggunaan Algoritma PID tidak menjamin kontrol sistem yang optimal atau bahkan tingkat stabilitasnya [14][15]

Kontrol PID adalah sistem kontrol yang bersifat continue, khususnya adalah sistem kontrol yang beroperasi dalam rentang 0 hingga 100%. Karena kontrol PID bersifat continue, jenis kontrol ini sangat baik untuk mengontrol proses dengan fluktuasi atau perubahan proses yang cepat. Pengendali tipe PID sangat cocok untuk pengaturan suhu karena memiliki keunggulan sebagai berikut:

- a) Waktu naik yang cepat.
- b) Dapat memperkecil kesalahan.
- c) Dapat meredam kekurangan tanggapan atau kelebihan tanggapan yang mengakibatkan kestabilan yang baik dan cepatnya tanggapan terhadap perubahan yang dikendalikan.[2]

B. Parameter Sistem Kontrol

Parameter-parameter yang diperlukan untuk menganalisa karakteristik performa pada sistem kontrol adalah sebagai berikut :

- 1) Waktu tunda (*delay time*): adalah ukuran atau durasi waktu yang diperlukan untuk mencapai setengah nilai akhir atau 50% dari respon *steady state*. Persamaan berikut menyatakan besarnya waktu tunda dari respon :

$$t_d = \frac{0.742}{\xi \omega_n}$$

- 2) Waktu naik (*rise time*): adalah waktu yang diperlukan oleh respon untuk naik dari 10% menjadi 90%, 5% menjadi 95%, atau 0% menjadi 100% dari nilai akhir yang biasa digunakan. Untuk sistem atas redaman waktu naik yang biasa digunakan 10% menjadi 90%. Besarnya nilai pada saat naik dinyatakan dengan pernyataan sebagai berikut :

$$t_r = \frac{1}{\omega_n \sqrt{1 - \xi^2}} \left(\pi - \tan^{-1} \frac{\sqrt{1 - \xi^2}}{\xi} \right)$$

- 3) Waktu puncak (*peak time*), t_p : adalah waktu yang diperlukan oleh respon untuk mencapai puncak pertama lewat (*overshoot*). Maksimum dari *overshoot* adalah nilai puncak kurva, respons 19 ukur dari satuan. Besarnya persen lewat maksimum menunjukkan kestabilan relatif dari sistem. Waktu puncak dapat dinyatakan sebagai berikut :

$$t_p = \frac{1}{\omega_n \sqrt{1 - \xi^2}}$$

Agar sistem kendali mempunyai performansi yang baik, maka diusahakan nilai waktu puncak sekecil mungkin.

- 4) Waktu penetapan atau waktu turun (*settling time*) : adalah waktu yang diperlukan untuk merespon kurva agar dapat mencapai dan tetap berada dalam gugus nilai akhir ukuran yang disederhanakan dengan presentase mutlak harga akhirnya (biasanya 2% atau 5%). *Settling time* dinyatakan dengan persamaan sebagai berikut :

$$t_s(\pm 5\%) = \frac{3}{\xi\omega_n} \quad t_s(\pm 2\%) = \frac{4}{\xi\omega_n} \quad t_s(\pm 0.5\%) = \frac{5}{\xi\omega_n}$$

- 5) Waktu keadaan *steady* adalah waktu yang diperlukan kurva respons untuk mencapai dan menstabilkan di daerah sekitar nilai akhir yang ukurannya ditentukan sebagai persentase mutlak dari nilai akhir. Terakhir, biasanya 5%, 2% atau 0,5 dengan laju redaman mulai dari 0.9 detik.

Untuk kriteria lebih kurang dari 2%, waktu keadaan *steady* dihitung dengan menggunakan persamaan :

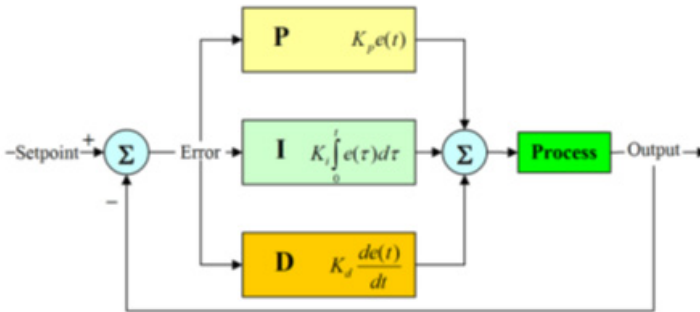
Untuk kriteria lebih kurang dari 5%, waktu *steady* dihitung dengan menggunakan persamaan:

Untuk kriteria kurang dari 0,5%, waktu yang keadaan mantap *steady* dihitung dengan menggunakan persamaan:

Agar sistem kendali mempunyai performansi yang baik, maka diusahakan nilai waktu keadaan *steady* sekecil mungkin.[2]

C. Komponen PID

PID (*Proportional Integral Derivative*) *Controller* merupakan *controller* untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik / *feed back* pada sistem tersebut. Komponen PID terdiri dari 3 jenis, yaitu *Proportional*, *Integratif*, dan *Derivatif*. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri, tergantung dari respon yang diinginkan terhadap suatu *plant*.



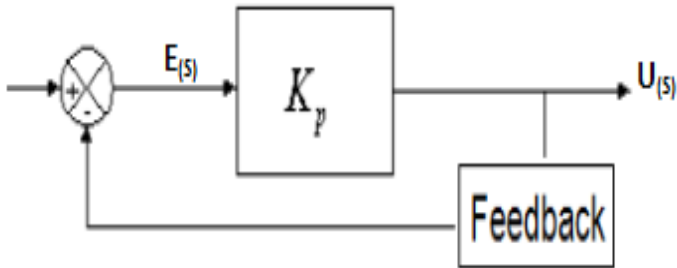
Gambar 3.1 Blok Diagram PID Controller

Ada 3 macam control PID yaitu *control Proportional*, *Integratif*, dan *Derivatif*. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon terhadap suatu sistem.

1. Kontrol Proportional (P)

Kontrol *proportional* merupakan sebuah penguat input sehingga hasil pada *output* tidak semakin menjadi kecil pada sebuah sistem. *Output proportional* adalah hasil perkalian antara konstanta proposional dengan nilai *error* nya. Perubahan yang terjadi pada

sinyal *input* akan menyebabkan sistem secara langsung mengubah *output* sebesar konstanta pengalinya.



Gambar 3.2 Blok Diagram Untuk Kontrol Proporsional

Persamaan matematis untuk pengendali Proporsional adalah :

$$U(t) = K_p e(t)$$

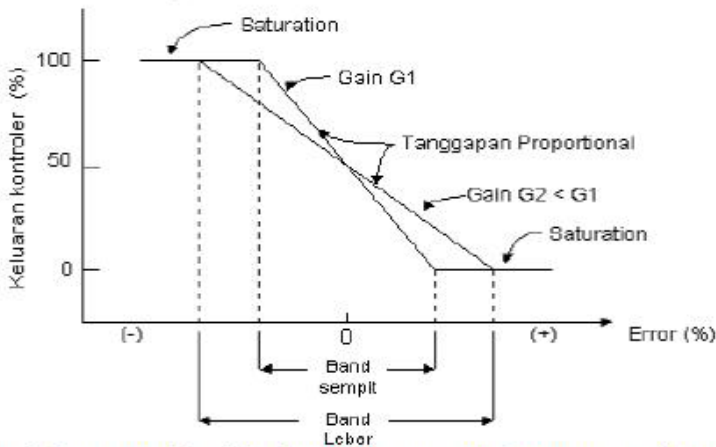
Fungsi alih untuk pengendali proporsional adalah

$$\frac{U(s)}{E(s)} = K_P$$

K_p = adalah kepekaan proporsional atau penguatan

Pengendali *proportional* memiliki 2 parameter yaitu: pita proporsional (*band proportional*) dan konstanta proporsional. Daerah kerja efektif kontroler dicerminkan oleh pita *proportional*, sedangkan konstanta *proportional* menunjukkan nilai faktor penguatan terhadap sinyal kesalahan, K_p .

Hubungan antara pita proporsional (PB) dengan konstanta proporsional (KP) ditunjukkan sbb : $PB = (1 / KP) \times 100\%$ Gambar berikut ini merupakan grafik hubungan antara PB, keluaran controller dan kesalahan yang merupakan masukan controller. Ketika konstanta proporsional semakin tinggi, pita proporsional menunjukkan penurunan yang semakin kecil, sehingga lingkup kerja yang dikuatkan akan semakin sempit.

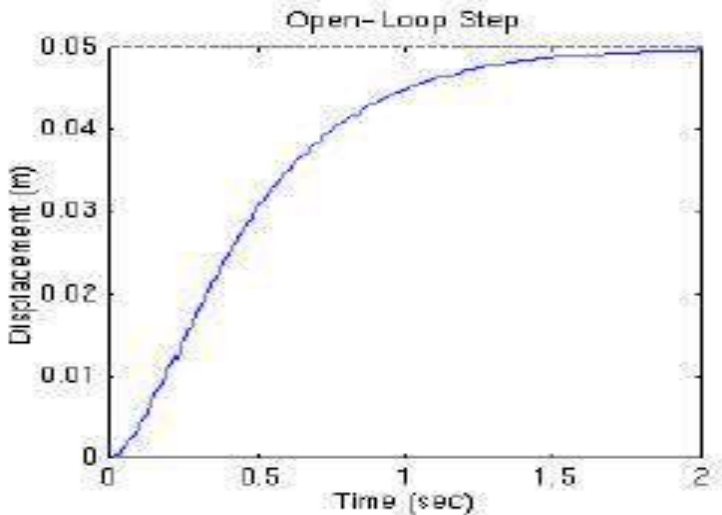


Gambar 3.3 Proportional band dari controller tergantung pada penguatan [11]

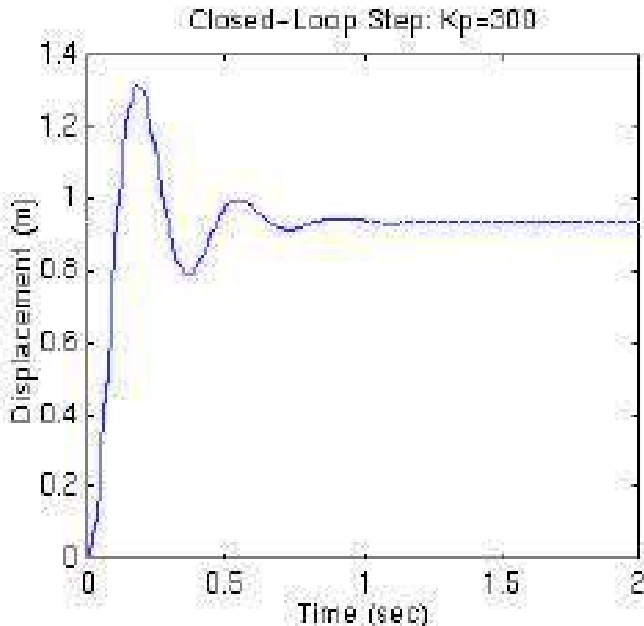
Penambahan aksi kontrol P mempunyai pengaruh mengurangi waktu naik dan kesalahan tunak, tetapi konsekuensinya *overshoot* naik cukup besar. Kenaikan *overshoot* ini sebanding dengan nilai parameter K_p , waktu turun juga menunjukkan kecenderungan yang membesar.

Karakteristik kontrol proporsional :

- a) Jika nilai K_p kecil, pengontrol proporsional hanya mampu melakukan koreksi kesalahan yang kecil, sehingga akan menghasilkan respon sistem yang lambat (menambah *rise time*).
- b) Jika nilai K_p dinaikkan, respon/tanggapan sistem akan semakin cepat mencapai keadaan stabilnya (mengurangi *rise time*).
- c) Namun jika nilai K_p diperbesar sehingga mencapai harga yang berlebihan, akan mengakibatkan sistem bekerja tidak stabil atau respon sistem akan berosilasi.
- d) Nilai K_p dapat diset sedemikian sehingga mengurangi *steady state error*, tetapi tidak menghilangkannya.[11]



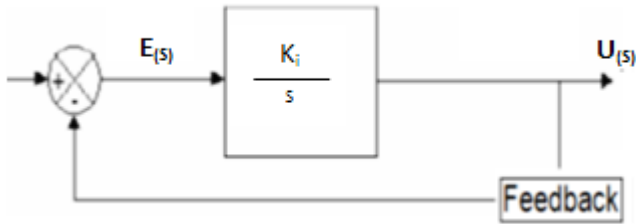
Gambar 3.4 Respon Sistem sebelum diberi K_p



Gambar 3.5 Respon Sistem setelah diberi K_p . [11]

2. Kontrol Integral

Controller proporsional tidak akan mampu menjamin *output* dari sistem akan menuju ke keadaan yang diinginkan kalau sebuah plant tidak memiliki unsur *integrator*. Pada *controller integral*, respon kepada sistem akan meningkat secara *continue* terus-menerus kecuali nilai *error* yang diintegrasikan dengan batasan atas t dan batasan bawah 0 (nol).



Gambar 3.6 Blok Diagram Untuk Pengendali Integral

Persamaan matematis untuk pengendali Integral adalah

$$U(t) = K_i \int_0^t e(t) dt$$

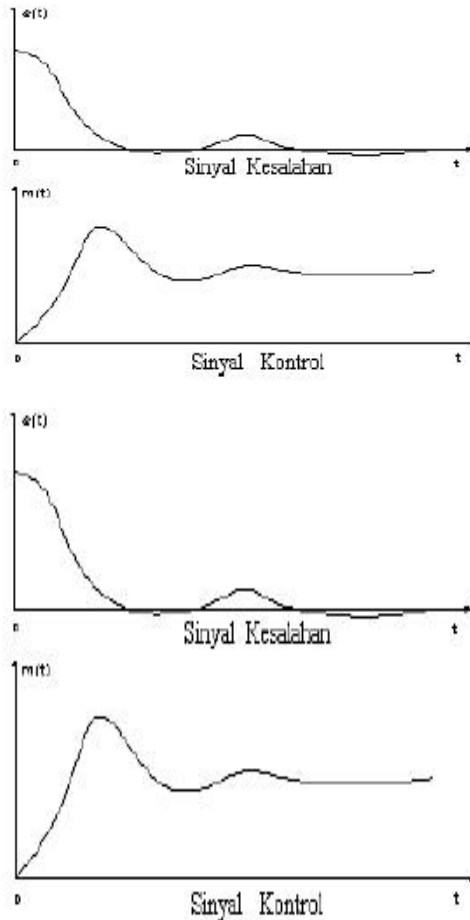
fungsi alih untuk pengendali Integral adalah :

$$\frac{U(s)}{E(s)} = \frac{K_i}{s}$$

Controller integral memiliki karakteristik seperti halnya sebuah integral. Keluaran controller sangat dipengaruhi oleh perubahan yang sebanding dengan nilai sinyal kesalahan. Keluaran controller ini merupakan jumlahan yang terus menerus dari perubahan masukannya. Kalau sinyal kesalahan tidak mengalami perubahan, maka keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan.

Sinyal keluaran *controller* integral merupakan luas bidang yang dibentuk oleh kurva kesalahan

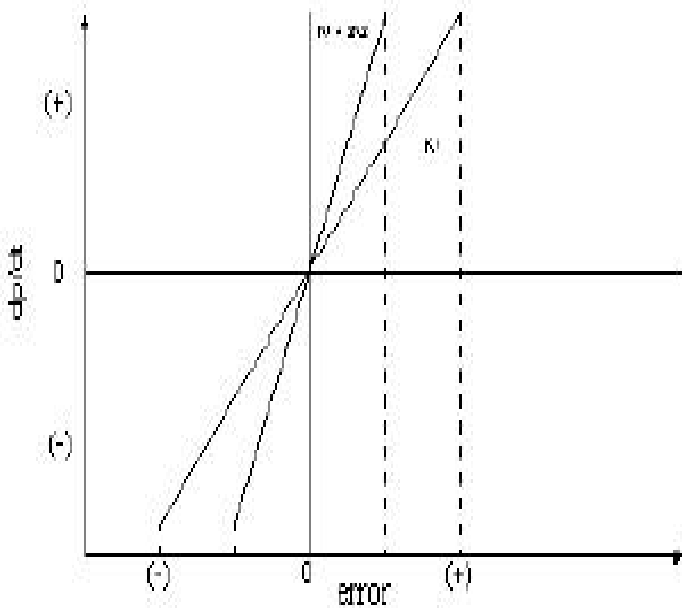
penggerak. Sinyal keluaran akan berharga sama dengan harga sebelumnya ketika sinyal kesalahan berharga nol. Gambar 2.12 menunjukkan contoh sinyal kesalahan yang yang disulutkan kedalam kontroler integral terhadap perubahan sinyal kesalahan tersebut.



Gambar 3.7 Kurva sinyal kesalahan terhadap t dan kurva $u(t)$ terhadap t pada pembangkit kesalahan nol

K_i = adalah konstanta pengendali Integral.

Pengaruh perubahan konstanta integral terhadap keluaran integral ditunjukkan pada gambar dibawah ini. Bila nilai $e(t)$ naik 2 kali, maka laju perubahan $u(t)$ terhadap waktu menjadi 2 kali lebih cepat. Bila $e(t)$ tetap maka nilai $u(t)$ akan tetap seperti semula. Aksi reset setelah ada perubahan beban. Jika nilai konstanta integrator berubah menjadi lebih besar, sinyal kesalahan yang relatif lebih kecil dapat mengakibatkan laju keluaran menjadi besar.[11]



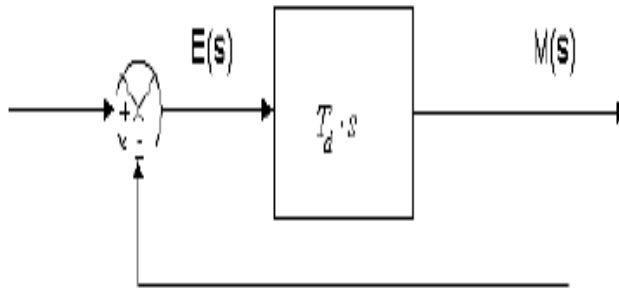
Gambar 3.8 Perubahan keluaran sebagai akibat penguatan kesalahan

Karakteristik kontrol integral :

- a) Keluaran pengontrol integral membutuhkan selang waktu tertentu, sehingga pengontrol integral cenderung memperlambat respon.
- b) Ketika sinyal kesalahan berharga nol, keluaran pengontrol akan bertahan pada nilai sebelumnya.
- c) Jika sinyal kesalahan tidak berharga nol, keluaran akan menunjukkan kenaikan atau penurunan yang dipengaruhi oleh besarnya sinyal kesalahan dan nilai K_i .
- d) Konstanta integral K_i yang berharga besar akan mempercepat hilangnya offset. Tetapi semakin besar nilai konstanta K_i akan mengakibatkan peningkatan osilasi dari sinyal keluaran pengontrol.[11]

3. Kontrol diferensial

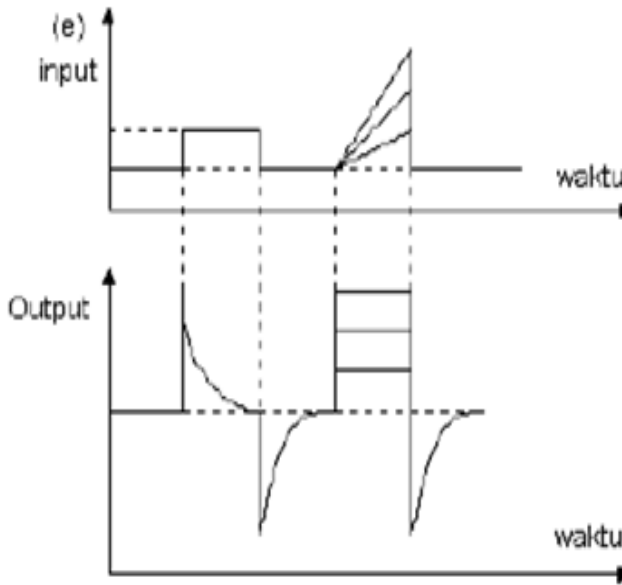
Keluaran kontroler *diferensial* atau kontroler turunan memiliki sifat seperti halnya suatu operasi diferensial. Perubahan yang mendadak pada masukan *controller*, akan mengakibatkan perubahan yang sangat besar dan cepat. Aksi *controller*, turunan, sering disebut juga kontrol laju (*rate control*), karena besar keluaran kontroler sebanding dengan laju perubahan sinyal kesalahan penggerak.[14]



Gambar 3.9 Blok Diagram Sistem kontrol differential

Ketika masukannya tidak mengalami perubahan, maka keluaran controller juga tidak mengalami perubahan, sedangkan apabila sinyal masukan berubah mendadak dan menaik, maka keluaran menghasilkan sinyal berbentuk impulse. Jika sinyal masukan berubah naik secara perlahan, maka keluarannya justru merupakan fungsi step yang besar magnitudnya sangat dipengaruhi oleh kecepatan naik dari fungsi ramp dan faktor konstanta difrensialnya.

Gambar berikut ini menyatakan hubungan anatar sinyal masukan dengan sinyal keluaran *controller diferensial*



Gambar 3.10 Kurva waktu hubungan *input output controller diferensial*

Keluaran pengontrol diferensial memiliki sifat seperti halnya suatu operasi *derivatif*. Perubahan yang mendadak pada masukan pengontrol akan mengakibatkan perubahan yang sangat besar dan cepat. Ketika masukannya tidak mengalami perubahan, keluaran pengontrol juga tidak mengalami perubahan, sedangkan apabila sinyal masukan berubah mendadak dan menaik (berbentuk fungsi *step*), keluaran menghasilkan sinyal berbentuk impuls. Jika sinyal masukan berubah naik secara perlahan (fungsi *ramp*),

keluarannya justru merupakan fungsi step yang besar magnitudenya sangat dipengaruhi oleh kecepatan naik dari fungsi *ramp* dan *factor* konstanta K_d .

Karakteristik kontrol Diferensial :

- a) Pengontrol tidak dapat menghasilkan keluaran jika tidak ada perubahan pada masukannya (berupa perubahan sinyal kesalahan)
- b) Jika sinyal kesalahan berubah terhadap waktu, maka keluaran yang dihasilkan pengontrol tergantung pada nilai K_d dan laju perubahan sinyal kesalahan.
- c) Pengontrol diferensial mempunyai suatu karakter untuk mendahului, sehingga pengontrol ini dapat menghasilkan koreksi yang signifikan sebelum pembangkit kesalahan menjadi sangat besar. Jadi pengontrol diferensial dapat mengantisipasi pembangkit kesalahan, memberikan aksi yang bersifat korektif dan cenderung meningkatkan stabilitas sistem.
- d) Dengan meningkatkan nilai K_d , dapat meningkatkan stabilitas sistem dan mengurangi *overshoot*. [11]

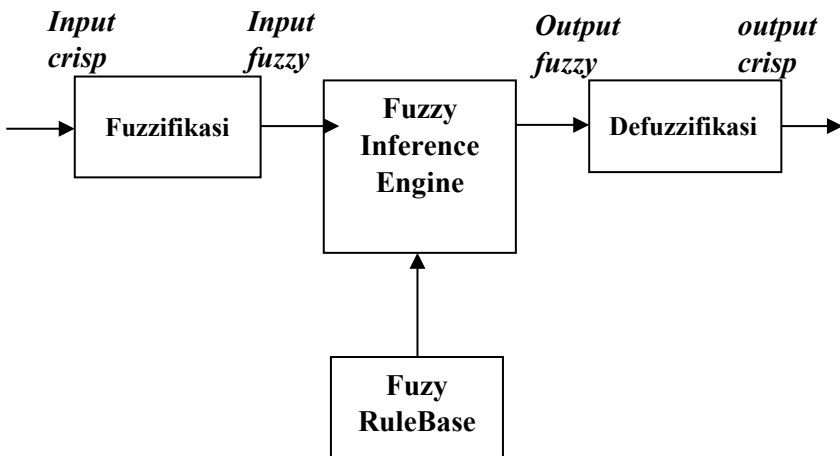
Berdasarkan karakteristik pengontrol ini, pengontrol diferensial umumnya dipakai untuk mempercepat respon awal suatu sistem, tetapi tidak memperkecil kesalahan pada keadaan tunaknya. Kerja pengontrol diferensial hanyalah efektif pada lingkup yang sempit, yaitu pada periode peralihan. Oleh sebab itu pengontrol

diferensial tidak pernah digunakan tanpa ada *controller* lainnya.

BAB IV

FUZZY INFERENCE SYSTEM (FIS)

Fuzzy inference yaitu melakukan penalaran menggunakan fuzzy input dan aturan fuzzy yang telah ditentukan sehingga menghasilkan fuzzy output. Struktur utama dari *Fuzzy logic system* adalah dapat dilihat pada gambar 4.1:

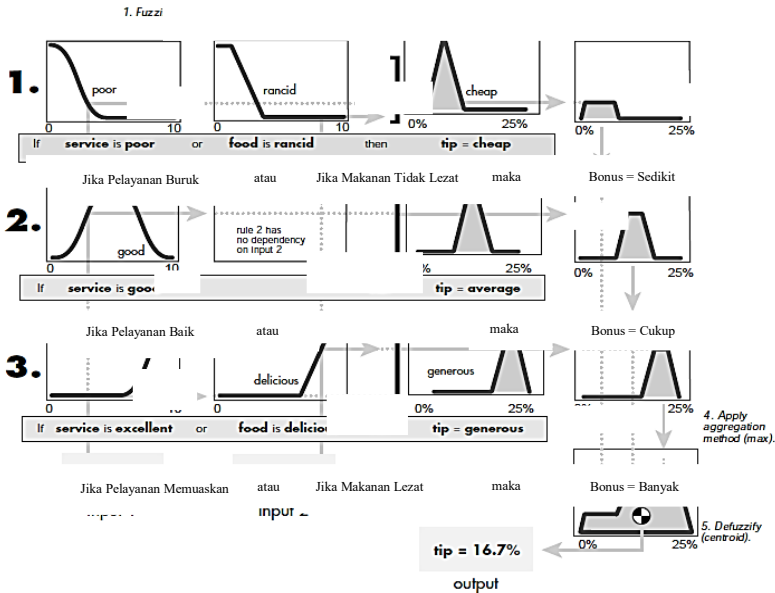


Gambar 4.1 Struktur *Fuzzy Logic System*

Metode aturan fuzzy pertama kali dikenalkan oleh Ebrahim Mamdani tahun 1975 yang dikenal dengan metode mamdani atau Max-Min dimana terdapat lima tahapan untuk mendapatkan output yaitu:[16]

- a. Pembentukan himpunan Fuzzy (variabel input dan output)
- b. Operasi *membership function*
- c. Aplikasi fungsi implikasi (*implication*), pada umumnya digunakan fungsi Min
- d. Komposisi aturan (*agregation*)
- e. Penegasan (*defuzzification*)

Kelima proses diatas dapat dilihat pada contoh kasus pemberian "Bonus" berdasarkan pelayanan dan makanan yang diberikan pada gambar 4.2.



Gambar 4.2 Fuzzy Inference System

Defuzifikasi adalah proses pemetaan dari hasil daerah inferensi fuzzy ke daerah non Fuzzy (*Crisp*). Pada

implementasi kontrol *real time fuzzy*, proses defuzzifikasi digunakan *Center of Area (COA)*.

Defuzzifikasi *center of area* menghasilkan pusat gravitasi dari distribusi aksi kontrol yang dinyatakan pada persamaan :

$$z^* = \frac{\sum_{k=1}^m V_k \mu_v(V_k)}{\sum_{k=1}^m \mu_v(V_k)}$$

z^* = nilai output

m = tingkat kuantisasi

V_k = elemen ke -k

μ_v = *membership degree* elemen pada fuzzy set v

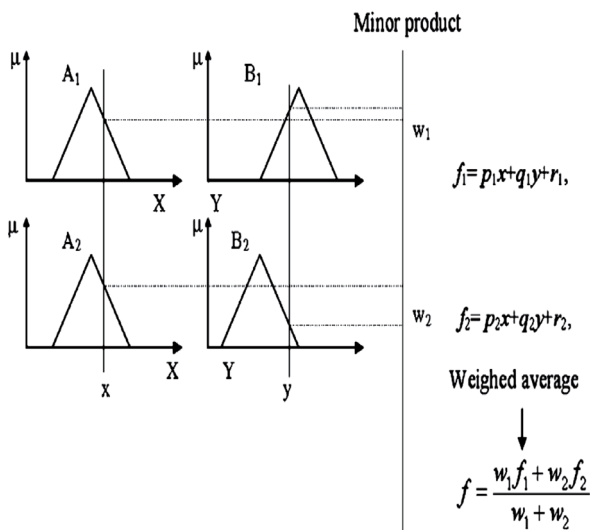
A. Mekanisme Inferensi Fuzzy TSK

Mekanisme inferensi fuzzy TSK orde satu dengan dua masukan x dan y (Gambar 9.6). Basis aturan dengan dua aturan fuzzy if-then seperti dibawah ini :

Rule 1 : if x is A_1 and y is B_1 then $f_1 = p_1x + q_1y + r_1$

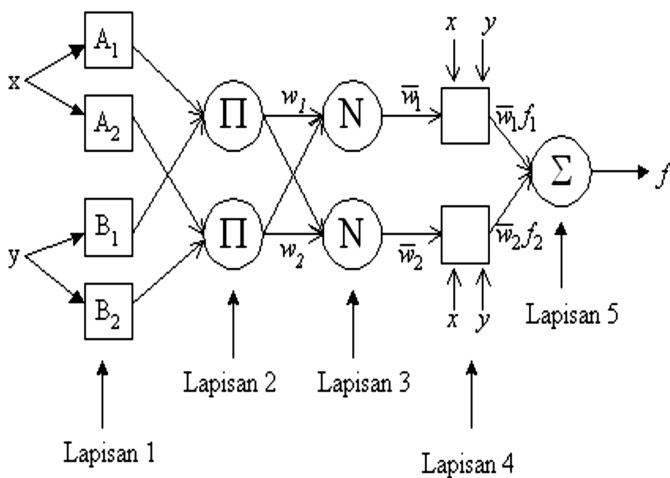
Rule 2 : if x is A_2 and y is B_2 then $f_2 = p_2x + q_2y + r_2$

Input : x dan y . Consequent-nya adalah f



Gambar 4.3 System inferensi fuzzy TSK

Sedangkan untuk Struktur ANFIS bisa dilihat pada Gambar 4.4.



Gambar 4.4 Struktur ANFIS

BAB V

CARA KERJA ANFIS

Cara kerjanya seperti system FIS biasa cuma cara perhitungannya (Algoritmanya) yang beda. Jelasnya gambar diatas memperlihatkan suatu masukan crisp (tidak fuzzy) x dan y , hmmm.. gini deh supaya jelas misalnya kita ingin mengontrol kecepatan motor listrik dengan mengatur tegangannya, jadi x itu pengukuran harga variabel yang dikontrol yaitu kecepatan pada saat ke t , dan misalnya y pengukuran pada saat ke $t+1$ sedangkan f adalah nilai tegangan yang diberikan sebagai sinyal kontrol. Harga x dan y tersebut jelaskan bukan fuzzy. Lalu nilai x dan y tersebut dipetakan pada fungsi keanggotaannya.

Dalam gambar diatas tiap-tiap input tersebut dibagi jadi 2 fungsi keanggotaan, x dibagi dalam $A1$ dan $A2$ anggap misalnya $A1$ menyatakan small dan $A2$ menyatakan big. Begitu juga y dibagi dalam fungsi keanggotaan $B1$ yang menyatakan small dan $B2$ yang menyatakan big.

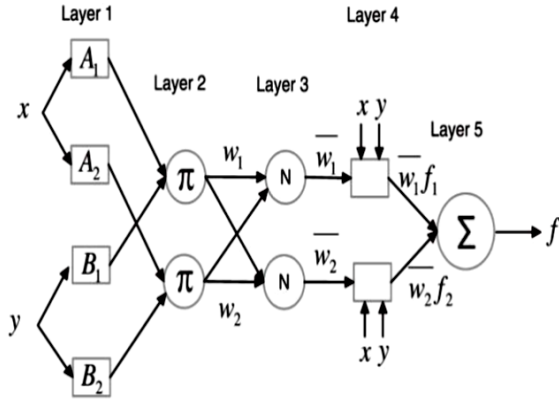
Dari pemetaan tersebut x dan y sudah jadi variabel fuzzy yang masing-masing punya nilai m small dan big tertentu. x mempunyai nilai m_{A1} dan m_{A2} sedangkan y punya nilai m_{B1} dan m_{B2} . Nilai masing-masing pasangan input tersebut lalu diagregasi dengan operasi T-norm, misalnya operasi ini adalah operasi AND. Jadi $w1 = (m_{A1} \text{ AND } m_{A2})$ sedangkan $w2 = (m_{B1} \text{ AND } m_{B2})$.

Dari basis aturan yang udah dibuat kita tau
if $w=w_1$ then $f_1 = p_1x + q_1y + r_1$
if $w=w_2$ then $f_2 = p_2x + q_2y + r_2$

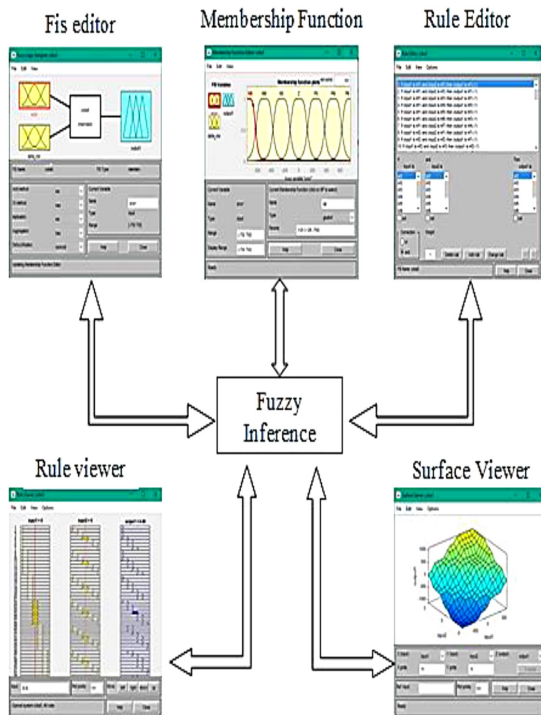
Hasil akhir f_1 dan f_2 . Ini merupakan nilai output sinyal kontrol, yaitu tegangan. Perhatikan kita telah loncat dari domain input x dan y (kecepatan) ke domain output f (tegangan).

Tapi itu nilai p_1 , q_1 , r_1 , p_2 , q_2 , dan r_2 dari mana, siapa yang nentuin? Itu namanya parameter konsekuen yang ditentukan dengan nilai awal tertentu dan akan berubah dengan pembelajaran (Algoritma belajar). Pada bagian pembelajaran parameter konsekuen hal ini akan dibicarakan lebih detail. Sekarang yang penting kita udah punya f_1 dan f_2 . Selanjutnya dari nilai f_1 dan f_2 ini kita perlu mendapatkan satu nilai tegangan sebagai sinyal kontrol. Nah nilai akhir tersebut dihitung dengan persamaan:

Ini namanya defuzzyfikasi. Rumus tersebut sebenarnya diperoleh dari salah satu metode defuzzyfikasi yaitu metode rata-rata tengah



Gambar 5.1 ANFIS structure



Gambar 5.2 The FIS flow

A. Proses Training data ANFIS

Pada struktur ANFIS, simpul adaptif terdapat pada lapisan pertama dan keempat. Simpul pada lapisan pertama mengandung parameter premis yang nonlinier sedangkan pada lapisan keempat mengandung parameter konsekuen yang linier. Nah untuk memperbaharui parameter parameter itu atau dalam kata lain si jaringan saraf itu belajar, maka perlu metoda atau Algoritma untuk itu. Metoda pembelajaran jaringan syaraf tiruan banyak macamnya, dan masing masing punya kelebihan dan kekurangan.

Disini akan dijelaskan pembelajaran hibrid untuk ANFIS. Artinya penggunaan/penyatuan dua metoda pembelajaran pada ANFIS. Pembelajaran hibrid terdiri atas dua bagian yaitu arah maju (forward pass) dan arah mundur (*backward pass*).

Pada arah maju, parameter premis dibuat tetap. Dengan menggunakan metode *Recursive Least Square Estimator* (RLSE), parameter konsekuen diperbaiki berdasarkan pasangan data masukan-keluaran. Metode RLSE dapat diterapkan karena parameter konsekuen yang diperbaiki adalah parameter linier. Metode RLSE akan mempercepat proses belajar hybrid. Kemudian setelah parameter konsekuen didapatkan, data masukan dilewatkan jaringan adaptif kembali dan hasil keluaran jaringan adaptif ini dibandingkan dengan keluaran yang sebenarnya[17].

Pada arah mundur, parameter konsekuen dibuat tetap. Kesalahan yang terjadi antara keluaran jaringan adaptif dan keluaran sebenarnya dipropagasikan balik dengan

menggunakan gradient descent untuk memperbaiki parameter premis. Pembelajaran ini dikenal sebagai Algoritma *Backpropagation-error*.

Satu tahap arah pembelajaran maju-mundur dinamakan satu epoch. Tabel 5.1 menerangkan proses pembelajaran hibrid ANFIS.

Tabel 5.1 Proses pembelajaran hibrid ANFIS

	Arah maju	Arah mundur
Parameter premis	Tetap	Gradient descent
Parameter konsekuen	RLSE	Tetap
Sinyal	Keluaran simpul	Laju kesalahan

B. Defuzzifikasi

Pada *Fuzzy* proses defuzzifikasi kontrol logika melalui *type-reducer* yang memiliki beberapa metode Algoritma seperti *Kernik Mendel Algoritm (KMA)*, *Enhance Kernik Mendel Algoritm (EKMA)*, *Enhance Kernik Mendel Algoritm with Initialization (EKMANI)*, *Iterative Algoritm with Stop Condition (IASC)*, *Enhance Iterative Algoritm with Stop Condition (EIASC)* [13]. Proses defuzzifikasi dengan menggunakan *centroid* pada FLS.

BAB VI

PSO

(Particle Swarm Optimzation)

Dalam ilmu komputasi, particle swarm optimization (PSO) adalah metode komputasi yang mengoptimalkan masalah dengan secara iterative mencoba meningkatkan solusi kandidat terkait dengan ukuran kualitas yang diberikan. Ini memecahkan masalah dengan memiliki populasi kandidat solusi, di sini dijuluki partikel, dan menggerakkan partikel-partikel ini di dalam ruang pencarian sesuai dengan rumus matematika sederhana atas posisi dan kecepatan partikel. Setiap gerakan partikel dipengaruhi oleh posisi lokalnya yang paling dikenal, tetapi juga diarahkan ke posisi yang paling dikenal di ruang pencarian, yang diperbarui sebagai posisi yang lebih baik ditemukan oleh partikel lain. Ini diharapkan untuk menggerakkan kawanan menuju solusi terbaik[18][19][20][21][18].

PSO awalnya dikaitkan dengan Kennedy, Eberhart, dan Shi dan pertama kali dimaksudkan untuk mensimulasikan perilaku sosial, sebagai representasi bergaya gerakan organisme dalam kawanan burung atau sekolah ikan. Algoritma disederhanakan dan diamati melakukan optimasi. Buku karya Kennedy dan Eberhart menggambarkan banyak aspek filosofis dari PSO dan intelijen yang berkerumun. Survei ekstensif aplikasi

PSO dilakukan oleh Poli. Baru-baru ini, ulasan komprehensif tentang karya teoritis dan eksperimental pada PSO diterbitkan oleh Bonyadi dan Michalewicz[21].

PSO adalah metaheuristik karena membuat sedikit atau tidak ada asumsi tentang masalah yang dioptimalkan dan dapat mencari ruang solusi kandidat yang sangat besar. Namun, metaheuristik seperti PSO tidak menjamin solusi optimal yang pernah ditemukan. Juga, PSO tidak menggunakan gradient masalah yang sedang dioptimalkan, yang berarti PSO tidak mengharuskan masalah optimisasi dapat dibedakan seperti yang disyaratkan oleh metode optimasi klasik seperti penurunan gradient dan metode quasi-newton

A. Algoritma

Varian dasar dari Algoritma PSO bekerja dengan memiliki populasi (disebut segerombolan) solusi kandidat (disebut partikel). Partikel-partikel ini dipindahkan di ruang pencarian menurut beberapa rumus sederhana. Pergerakan partikel dipandu oleh posisi mereka sendiri yang paling dikenal di ruang pencarian serta posisi segerombolan yang paling terkenal. Ketika posisi yang lebih baik sedang ditemukan, ini akan datang untuk memandu pergerakan kawanannya. Proses ini diulangi dan dengan demikian diharapkan, tetapi tidak dijamin, bahwa solusi yang memuaskan pada akhirnya akan ditemukan.

Secara formal, misalkan $f: \mathbb{R}^n \rightarrow \mathbb{R}$ menjadi fungsi biaya yang harus diminimalkan. Fungsi mengambil solusi kandidat sebagai argumen dalam bentuk vector dari bilangan real dan menghasilkan sejumlah nyata sebagai

output yang menunjukkan nilai fungsi tujuan dari kandidat solusi yang diberikan. The gradient dari f tidak diketahui. Tujuannya adalah untuk menemukan solusi \mathbf{a} yang $f(\mathbf{a}) \leq f(\mathbf{b})$ untuk semua \mathbf{b} dalam ruang pencarian, yang berarti \mathbf{a} adalah minimum global.

Misalkan S adalah jumlah partikel dalam kawanan, masing-masing memiliki posisi $\mathbf{x}_i \in \mathbb{R}^n$ dalam ruang pencarian dan kecepatan $\mathbf{v}_i \in \mathbb{R}^n$. Biarkan \mathbf{p}_i menjadi posisi partikel i yang paling dikenal dan biarkan \mathbf{g} menjadi posisi paling dikenal dari seluruh kawanan. Algoritma PSO dasar adalah:

```

for each particle  $i = 1, \dots, S$  do
  Initialize the particle's position with a uniformly distributed
  random vector:  $\mathbf{x}_i \sim U(\text{blo}, \text{bup})$ 
  Initialize the particle's best known position to its initial
  position:  $\mathbf{p}_i \leftarrow \mathbf{x}_i$ 
  if  $f(\mathbf{p}_i) < f(\mathbf{g})$  then
    update the swarm's best known position:  $\mathbf{g} \leftarrow \mathbf{p}_i$ 
  Initialize the particle's velocity:  $\mathbf{v}_i \sim U(-|\text{bup}-\text{blo}|, |\text{bup}-\text{blo}|)$ 
while a termination criterion is not met do:
  for each particle  $i = 1, \dots, S$  do
    for each dimension  $d = 1, \dots, n$  do
      Pick random numbers:  $r_p, r_g \sim U(0,1)$ 
      Update the particle's velocity:  $v_{i,d} \leftarrow \omega v_{i,d} + \phi_p r_p (p_{i,d} - x_{i,d}) + \phi_g r_g (g_d - x_{i,d})$ 
      Update the particle's position:  $x_i \leftarrow x_i + v_i$ 
    if  $f(x_i) < f(p_i)$  then
      Update the particle's best known position:  $p_i \leftarrow x_i$ 
    if  $f(p_i) < f(g)$  then
      Update the swarm's best known position:  $g \leftarrow p_i$ 

```

Nilai b_{lo} dan b_{up} mewakili batas bawah dan atas ruang pencarian masing-masing. Kriteria terminasi dapat berupa jumlah iterasi yang dilakukan, atau solusi di mana nilai fungsi objektif yang memadai ditemukan. Parameter ω , φ_p , dan φ_g dipilih oleh praktisi dan mengontrol perilaku dan kemandirian metode PSO.

B. Parameter selection

Pilihan parameter PSO dapat memiliki dampak besar pada kinerja pengoptimalan. Memilih parameter PSO yang menghasilkan kinerja yang baik telah menjadi subjek dari banyak penelitian.

Parameter PSO juga dapat disetel dengan menggunakan pengoptimal pelapis lainnya, sebuah konsep yang dikenal sebagai meta-optimisasi, atau bahkan disetel dengan baik selama optimasi, misalnya, dengan menggunakan logika fuzzy.

C. Neighbourhoods and topologies

Topologi gerombolan mendefinisikan subset partikel yang dengannya setiap partikel dapat bertukar informasi. Versi dasar dari Algoritma ini menggunakan topologi global sebagai struktur komunikasi swarm. Topologi ini memungkinkan semua partikel untuk berkomunikasi dengan semua partikel lain, sehingga seluruh kawanan berbagi posisi terbaik yang sama g dari satu partikel tunggal. Namun, pendekatan ini dapat menyebabkan gerombolan itu terjebak ke minimum lokal, sehingga topologi yang berbeda telah digunakan untuk mengontrol aliran informasi antar partikel. Misalnya, dalam topologi

lokal, partikel hanya berbagi informasi dengan subset partikel. Subset ini bisa berupa geometris, misalnya " m partikel terdekat" - atau, lebih sering, partikel sosial, yaitu sekumpulan partikel yang tidak tergantung pada jarak apa pun. Dalam kasus seperti itu, varian PSO dikatakan terbaik lokal (vs global terbaik untuk PSO dasar).

Topologi gerombolan yang umum digunakan adalah cincin, di mana setiap partikel hanya memiliki dua tetangga, tetapi ada banyak lainnya. Topologi belum tentu statis. Faktanya, karena topologi terkait dengan keragaman komunikasi partikel, beberapa upaya telah dilakukan untuk membuat topologi adaptif (SPSO, APSO, bintang stokastik, TRIBES, Cyber Swarm, dan C-PSO).

D. Inner workings

Ada beberapa aliran pemikiran mengapa dan bagaimana Algoritma PSO dapat melakukan optimasi. Keyakinan umum di antara para peneliti adalah bahwa perilaku gerombolan bervariasi antara perilaku eksplorasi, yaitu, mencari wilayah yang lebih luas dari ruang pencarian, dan perilaku eksploitatif, yaitu, pencarian yang berorientasi lokal sehingga lebih dekat ke (mungkin lokal) optimal. Aliran pemikiran ini telah lazim sejak awal PSO. Kelompok pemikiran ini berpendapat bahwa Algoritma PSO dan parameternya harus dipilih sehingga dapat menyeimbangkan antara eksplorasi dan eksploitasi untuk menghindari konvergensi premature ke optimum lokal namun tetap memastikan tingkat yang baik dari konvergensi dengan optimal. Keyakinan ini adalah pendahulu dari banyak varian PSO, Pemikiran lain adalah

bahwa perilaku gerombolan PSO tidak dipahami dengan baik dalam hal bagaimana hal itu memengaruhi kinerja pengoptimalan aktual, terutama untuk ruang pencarian berdimensi lebih tinggi dan masalah pengoptimalan yang mungkin terputus-putus, berisik, dan beragam waktu. Kelompok pemikiran ini hanya mencoba menemukan Algoritma dan parameter PSO yang menyebabkan kinerja yang baik terlepas dari bagaimana perilaku swarm dapat ditafsirkan dalam kaitannya dengan misalnya eksplorasi dan eksploitasi. Studi semacam itu telah mengarah pada penyederhanaan Algoritma PSO.

E. Konvergensi

Sehubungan dengan PSO, kata *konvergensi* biasanya merujuk pada dua definisi yang berbeda:

- Konvergensi urutan solusi (alias, analisis stabilitas, konvergen) di mana semua partikel telah berkumpul ke titik di ruang pencarian, yang mungkin atau mungkin tidak optimal,
- Konvergensi ke optimal lokal di mana semua Bests pribadi p atau sebaliknya, kawanan terbaik dikenal posisi g , mendekati optimum lokal dari masalah, terlepas dari bagaimana berperilaku swarm.

Konvergensi urutan solusi telah diselidiki untuk PSO. Analisis ini menghasilkan pedoman untuk memilih parameter PSO yang diyakini menyebabkan konvergensi ke titik dan mencegah divergensi partikel swarm (partikel tidak bergerak tanpa batas dan akan bertemu di suatu tempat). Namun, analisis tersebut dikritik oleh

Pedersen karena terlalu disederhanakan karena mereka menganggap swarm hanya memiliki satu partikel, bahwa ia tidak menggunakan variabel stokastik dan bahwa titik-titik tarikan, yaitu posisi p partikel yang paling dikenal dan swarm posisi paling dikenal g , tetap konstan selama proses optimasi. Namun, itu ditunjukkan bahwa penyederhanaan ini tidak mempengaruhi batas-batas yang ditemukan oleh penelitian ini untuk parameter di mana segerombolan adalah konvergen. Upaya yang cukup besar telah dilakukan dalam beberapa tahun terakhir untuk melemahkan asumsi pemodelan yang digunakan selama analisis stabilitas PSO, dengan hasil generalisasi terbaru yang berlaku untuk berbagai varian PSO dan memanfaatkan apa yang ditunjukkan sebagai asumsi pemodelan minimal yang diperlukan.

Konvergensi ke optimum lokal telah dianalisis untuk PSO di dan. Telah terbukti bahwa PSO memerlukan beberapa modifikasi untuk menjamin menemukan yang optimal secara lokal. Ini berarti bahwa menentukan kemampuan konvergensi dari berbagai Algoritma dan parameter PSO karena itu masih tergantung pada hasil empiris. Salah satu upaya untuk mengatasi masalah ini adalah pengembangan strategi "pembelajaran orthogonal" untuk peningkatan penggunaan informasi yang sudah ada dalam hubungan antara p dan g , sehingga membentuk contoh konvergen terkemuka dan menjadi efektif dengan topologi PSO. Tujuannya adalah untuk meningkatkan kinerja PSO secara keseluruhan, termasuk konvergensi global yang lebih cepat, kualitas solusi yang lebih tinggi, dan ketahanan yang lebih kuat. Namun,

penelitian tersebut tidak memberikan bukti teoritis untuk benar-benar membuktikan klaim mereka.

F. Mekanisme adaptif

Tanpa perlunya pertukaran antara konvergensi ('eksploitasi') dan divergensi ('eksplorasi'), mekanisme adaptif dapat diperkenalkan. Adaptive Particle Swarm Optimization (APSO) menghadirkan efisiensi pencarian yang lebih baik daripada PSO standar. APSO dapat melakukan pencarian global di seluruh ruang pencarian dengan kecepatan konvergensi yang lebih tinggi. Ini memungkinkan kontrol otomatis bobot inersia, koefisien akselerasi, dan parameter algoritmik lainnya pada waktu berjalan, sehingga meningkatkan efektivitas dan efisiensi pencarian pada saat yang sama. Juga, APSO dapat bertindak pada partikel terbaik global untuk melompat keluar dari kemungkinan lokal. Namun, APSO akan memperkenalkan parameter algoritme baru, namun tidak memperkenalkan desain tambahan atau kompleksitas implementasi.

G. Varian

Banyak varian bahkan Algoritma PSO dasar dimungkinkan. Misalnya, ada berbagai cara untuk menginisialisasi partikel dan kecepatan (misalnya mulai dengan nol kecepatan sebagai gantinya), cara mengurangi kecepatan, hanya memperbarui p_1 dan g setelah seluruh kawanan telah diperbarui, dll. Beberapa dari pilihan ini dan kemungkinan dampak kinerja telah dibahas dalam literatur.

Serangkaian implementasi standar telah dibuat oleh para peneliti terkemuka, "dimaksudkan untuk digunakan baik sebagai dasar untuk pengujian kinerja perbaikan teknik, serta untuk mewakili PSO ke komunitas optimasi yang lebih luas. Memiliki yang terkenal, terdefinisi dengan ketat Algoritma standar memberikan titik perbandingan yang berharga yang dapat digunakan di seluruh bidang penelitian untuk menguji kemajuan baru dengan lebih baik. "Yang terbaru adalah Standard PSO 2011

H. Hibridisasi

Varian PSO yang baru dan lebih canggih juga terus diperkenalkan dalam upaya untuk meningkatkan kinerja pengoptimalan. Ada tren tertentu dalam penelitian itu; salah satunya adalah membuat metode optimasi hibrida menggunakan PSO yang dikombinasikan dengan pengoptimal lain, misalnya, kombinasi PSO dengan optimisasi berbasis biogeografi, dan penggabungan metode pembelajaran yang efektif.

I. Meringankan konvergensi dini

Tren penelitian lain adalah mencoba dan mengurangi konvergensi prematur (yaitu, stagnasi optimasi), misalnya dengan membalikkan atau mengganggu pergerakan partikel PSO, pendekatan lain untuk menangani konvergensi prematur adalah penggunaan banyak gerombolan (optimisasi multi-gerombolan). Pendekatan multi-swarm juga dapat digunakan untuk mengimplementasikan optimasi multi-objektif. Akhirnya,

ada perkembangan dalam mengadaptasi parameter perilaku PSO selama optimisasi.

J. Penyederhanaan

Aliran pemikiran lain adalah bahwa PSO harus disederhanakan sebanyak mungkin tanpa mengganggu kinerjanya; sebuah konsep umum yang sering disebut sebagai Occam's razor . Penyederhanaan PSO pada awalnya disarankan oleh Kennedy dan telah dipelajari lebih luas, dimana tampak bahwa kinerja pengoptimalan ditingkatkan, dan parameter lebih mudah untuk disetel dan kinerjanya lebih konsisten di berbagai masalah optimisasi.

Argumen lain yang mendukung penyederhanaan PSO adalah bahwa metaheuristik hanya dapat menunjukkan kemanjurannya secara empiris dengan melakukan eksperimen komputasi pada sejumlah masalah optimasi yang terbatas. Ini berarti metaheuristik seperti PSO tidak dapat dibuktikan benar dan ini meningkatkan risiko membuat kesalahan dalam deskripsi dan implementasinya. Sebuah contoh yang baik dari ini menyajikan varian menjanjikan dari Algoritma genetika (metaheuristik populer lainnya) tetapi kemudian ditemukan cacat karena sangat bias dalam pencarian optimasi terhadap nilai yang sama untuk dimensi yang berbeda di ruang pencarian, yang kebetulan merupakan optimum dari masalah patokan yang dipertimbangkan. Bias ini karena kesalahan pemrograman, dan sekarang telah diperbaiki.

Inisialisasi kecepatan mungkin memerlukan input tambahan. Varian Bare Bones PSO telah diusulkan pada

tahun 2003 oleh James Kennedy, dan tidak perlu menggunakan kecepatan sama sekali. Varian lain yang lebih sederhana adalah akselerasi optimasi partikel (APSO), yang juga tidak perlu menggunakan kecepatan dan dapat mempercepat konvergensi di banyak aplikasi. Kode demo sederhana APSO tersedia.

K. Optimalisasi multi-objektif

PSO juga telah diterapkan pada masalah multi-objektif, dimana perbandingan fungsi tujuan memperhitungkan dominasi pareto ketika menggerakkan partikel PSO dan solusi yang tidak didominasi disimpan untuk memperkirakan bagian depan pareto .

L. Biner, diskrit, dan kombinatorial

Karena persamaan PSO yang diberikan di atas bekerja pada bilangan real, metode yang umum digunakan untuk memecahkan masalah diskrit adalah memetakan ruang pencarian diskrit ke domain kontinu, untuk menerapkan PSO klasik, dan kemudian memetakan hasilnya. Pemetaan semacam itu bisa sangat sederhana (misalnya dengan hanya menggunakan nilai bulat) atau lebih canggih.

Namun, dapat dicatat bahwa persamaan pergerakan menggunakan operator yang melakukan empat tindakan:

- menghitung perbedaan dua posisi. Hasilnya adalah kecepatan (lebih tepatnya perpindahan)
- mengalikan kecepatan dengan koefisien numerik
- menambahkan dua kecepatan
- menerapkan kecepatan ke suatu posisi

Biasanya posisi dan kecepatan diwakili oleh n bilangan real, dan operator ini hanya $-$, $*$, $+$, dan lagi $+$. Tetapi semua objek matematika ini dapat didefinisikan dengan cara yang sama sekali berbeda, untuk mengatasi masalah biner (atau yang lebih umum diskrit), atau bahkan yang kombinatorial. Salah satu pendekatan adalah mendefinisikan kembali operator berdasarkan set.

BAB VII

PARAMETER PSO

A. Parameter Particle Swarm Optimization (PSO)

Parameter parameter sangat diperlukan untuk memberi batasan batasan agar program sesuai dengan standar yang direncanakan. Parameter PSO dapat dilihat pada table 7.1.

Table 7.1. PSO Parameters

Parameters	Value
Number of Particles	30
Maximum iteration	50
Number of Variables	3
C2 (Social Constant)	2
C1 (Cognitive Constant)	2
W (Momentum Inertia)	0.9
Kp_pso	0 - 600
Ki_pso	0 - 10
Kd_pso	0 - 10

B. Pseudocode

Dari table parameter diatas, maka program pseudocode dapat dinyatakan menjadi:

- Parameter dalam metode

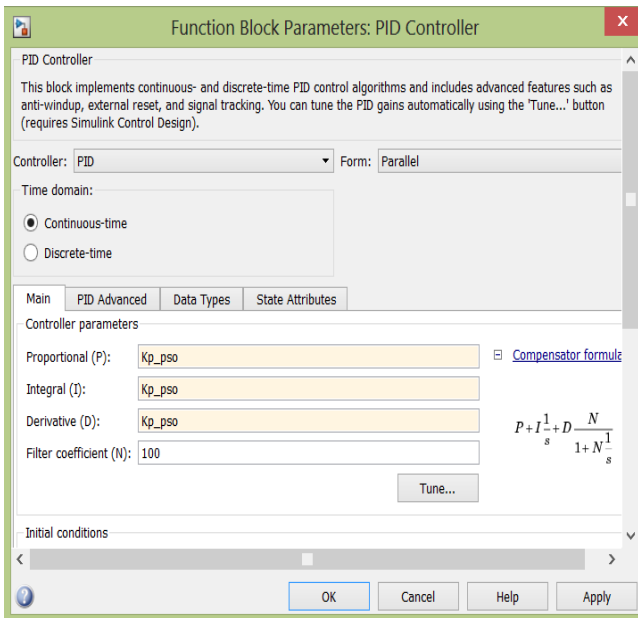
```
Juml_Particles = 30;
MaxIt          = 50;
Juml_Variabel = 3;
c2            = 2 ;
c1            = 2 ;
w             = 0.9;
```

- Parameter dalam PID Kontroller

```
Rb(1) = 0;   Ra(1) = 600;
Rb(2) = 0;   Ra(2) = 10;
Rb(3) = 0;   Ra(3) = 10;
```

C. Setting PID Controller

Gambar setting Simulink PID controller yang detuning dengan PSO dapat dilihat pada gambar 7.1.



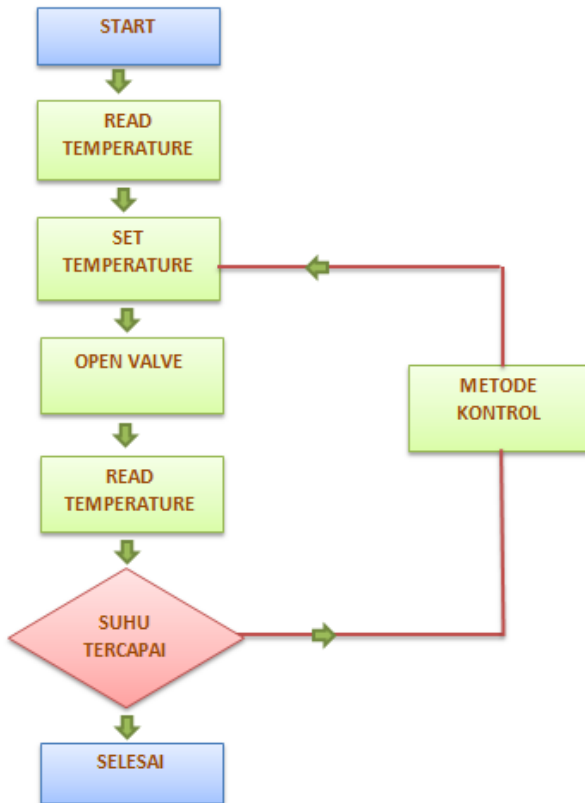
Gambar 7.1 Function Blok Parameter PID dituning PSO

BAB VIII

METODE

A. Flow chart

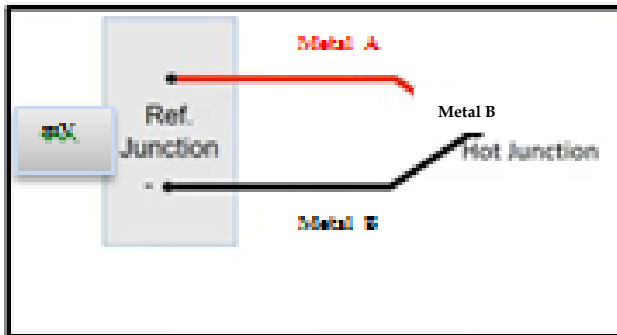
Berikut adalah flow Chart desain PID Kontroler sesuai dengan gambar :



Gambar 8.1 Gambar flowchart

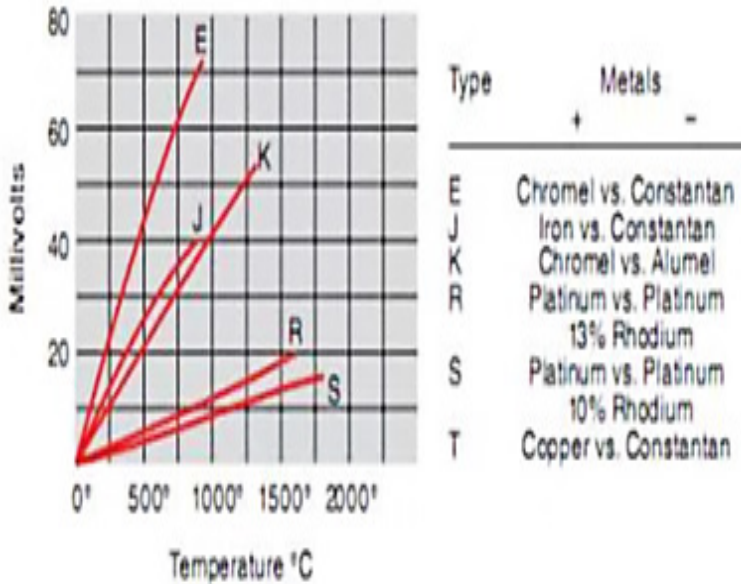
B. Sensor Thermocouple

Sensor thermocouple merupakan sensor untuk pembacaan temperatur yang tersusun dari dua bahan jenis logam yang hampir sama jenisnya dan ujungnya digabung menjadi satu, yang bilamana jika terkena panas akan menghasilkan efek *thermal electric*.



Gambar 8.2 Prinsip kerja *thermocouple*

Gambar 8.2 merupakan prinsip kerja dari *thermocouple*, dimana ketika ada suhu panas secara *gradient* pada konduktor logam, maka akan timbul perbedaan suhu yang kemudian dapat menghasilkan tegangan. Perbedaan tegangan yang terjadi antara dua titik perpotongan ini dikenal sebagai efek "*see back*".

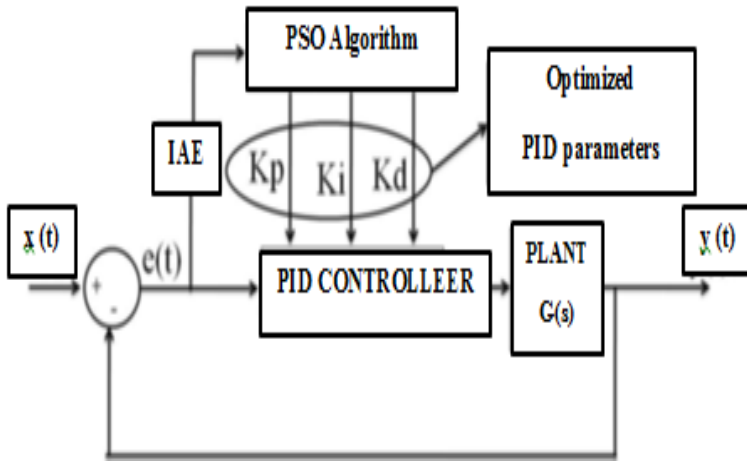


Gambar 8.3 *type-type thermocouple* sesuai spesifikasinya [1]

Thermocouple memiliki beberapa jenis *type* yang dalam penggunaannya dapat disesuaikan dengan media yang diukur, jika tidak sesuai dengan spesifikasi maka tingkat akurasi pembacaan akan berkurang.

C. Pemodelan sistem

Pada Buku ini menggunakan sistem software simulink matlab 2013a. setelah mengetahui alur, maka berikut adalah pemodelan sistem yang sesuai dengan *Gambar 3.3*.



Gambar 8.4 Pemodelan sistem pada PID - PSO

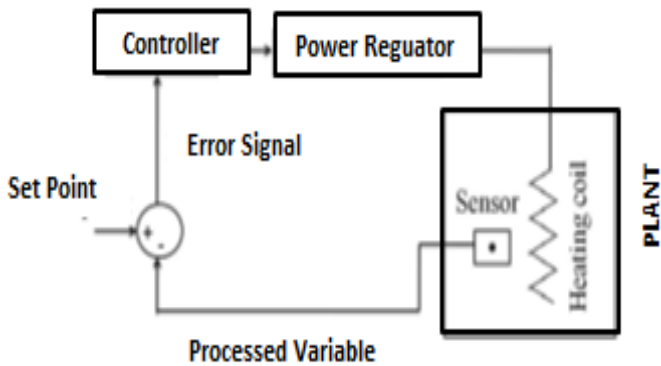
IAE = *Integral Absolute Error*

$x(t)$ = *set temperature*

$e(t)$ = *generated signal error*

$y(t)$ = *output temperature*

$G(s)$ = *Plant*



Gambar 8.5 Control temperature system

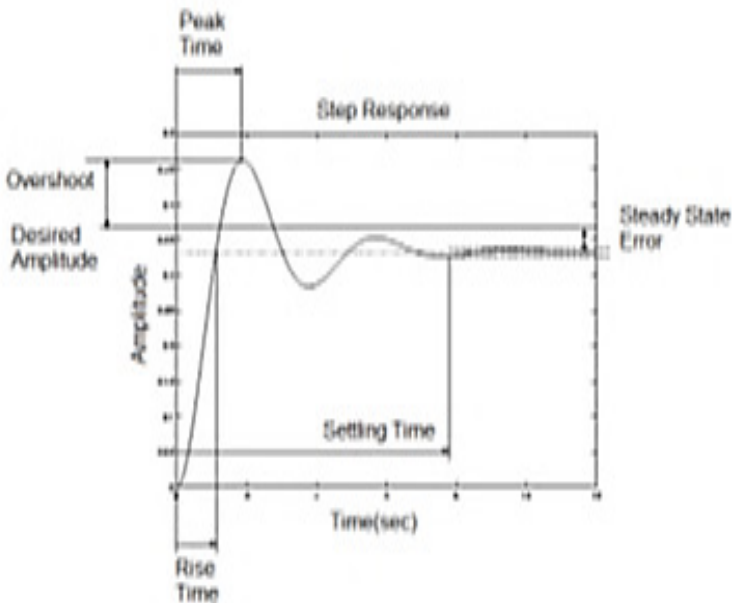
D. Parameter Kontrol PID

Ada beberapa parameter dalam menentukan suatu sistem *close loop*, yaitu *rise time*, *overshoot*, *settling time*, dan *steady state error*. *Rise time* adalah waktu yang dibutuhkan oleh *output plant* yang melebihi 90% dari tingkat yang diinginkan saat pertama kali sistem dijalankan. *Overshoot* adalah seberapa besar peak level lebih tinggi dari *steady state*, untuk membuat normal lagi *steady state*. *Settling time* adalah waktu yang dibutuhkan sistem untuk mengkonvergenkan *steady state*. *Steady state error* adalah perbedaan antara *steady state* output dengan output yang diinginkan.[4]

Tabel 8.1 Tabel pengaruh K_p , K_i dan K_d

Respon	Waktu Naik	Overshoot	Waktu Turun	S-S Error
K_p	Turun	Naik	Perub	Turun
K_i	Turun	Naik	Naik	Hilang
K_d	Kecil	Menurun	Menurun	kecil

K_p berguna untuk mengurangi *rise time*. Sedangkan K_i berguna untuk menghapuskan *steady state error*, K_d dapat mengurangi *overshoot* dan waktu turun.[4]



Gambar 8.6 Respon sistem

Kontrol signal $u(t)$ adalah kombinasi *linear* dari *error* $e(t)$, yaitu *Integral* dan *Derivatif*.

Persamaan dari kontrol PID adalah sebagai berikut:

$$u(t) = K_p \cdot e(t) + K_I \int e(t) dt + K_D \frac{de(t)}{dt}$$

$$u(t) = K_p \cdot e(t) + \frac{1}{T_I} \int e(t) dt + T_D \frac{de(t)}{dt}$$

Dimana :

K_p = Proporsional Gain

K_i = Integral Gain

K_D = Derivatif Gain

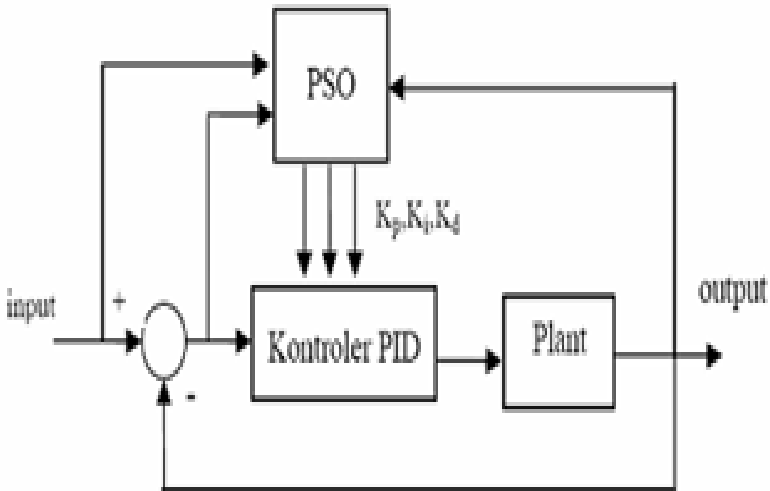
$T_i = \text{Integral Time}$

$T_D = \text{Derivatif Time}$

BAB IX

IMPLEMENTASI PID-PSO

Controller PID-PSO untuk *plant* ditunjukkan pada gambar 9.1, yang merupakan gabungan dari PID *controller* dan dituning dengan PSO



Gambar 9.1 Struktur controller PID-PSO

Dimana indeks performansi IAE yang dipakai untuk mengestimasi parameter-parameter PID diberikan sebagai berikut:

Konsep utama *tuning* kontroler PID secara *on line* adalah *tuning* parameter PD tiap *sampling* waktu. Fungsi

obyektif atau *fitness function* yang akan dioptimasi dinyatakan sebagai berikut :

$$J(i) = \alpha.IAE(i) + \beta.O(i)$$

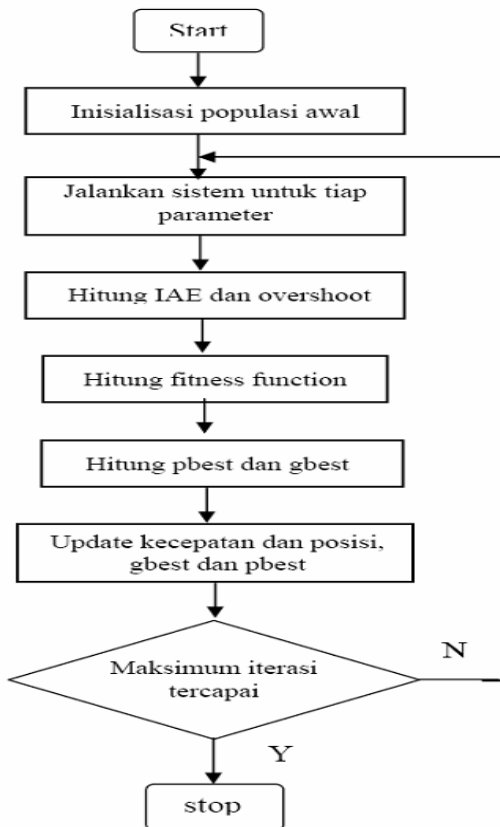
di mana :

α, β : faktor *improvement*

O : *overshoot*

Flowchart kontroler PID-PSO ditunjukkan pada Gambar

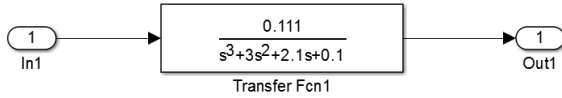
9.2



Gambar 9.2 *flowchart* controller PID- PSO

A. Transfer function

$$G_1(s) = \frac{1/90}{(s+1/30)(s+1/3)} \quad (3.7)$$



Gambar 9.3. Transfer Function

Dari *transfer function* tersebut akan diimplementasikan ke Simulink Matlab 2013.[10]

B. Parameter PSO

Parameter yang digunakan untuk PSO sebagai berikut

:

<i>PARAMETERS</i>	<i>VALUE</i>
<i>Number of particle</i>	30
<i>Maximum iteration</i>	50
<i>Number of Variables</i>	3
<i>C2 (Social Constant)</i>	2
<i>C1 (Cognitive Constant)</i>	2
<i>W (Momentum Inertia)</i>	0.1
Kp-PSO	0-40
Ki-PSO	0-10
Kd-PSO	0-1

Gambar 9.4 Parameter input PSO

BAB X

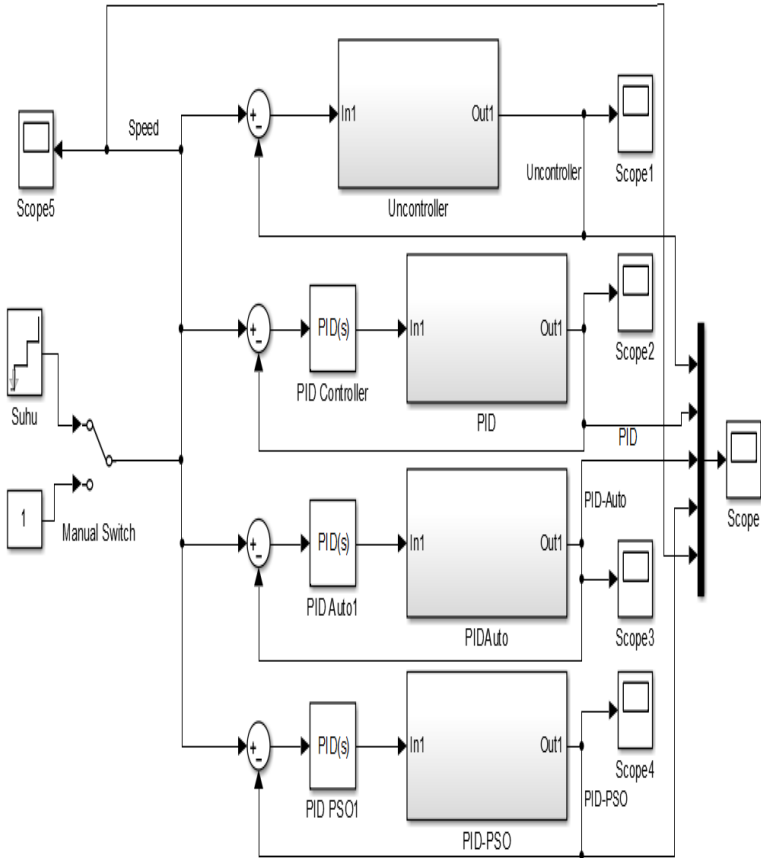
ANALISA DAN PID-PSO

A. Hasil Dan Analisa

Simulasi yang dilakukan untuk mengoptimalkan sistem PID dengan metode *Particle Swarm Optomozation (PSO)* yaitu menggunakan Matlab Simulink 2013. Respon sistem yang dihasilkan akan dibandingkan dengan berbagai kondisi, seperti PID tanpa kontrol, PID standar/konvensional, PID auto tuning, dan PID dengan metode PSO. Dengan perbandingan tersebut, maka akan didapatkan karakteristik dari masing-masing PID kontroler beserta metode optimasinya.

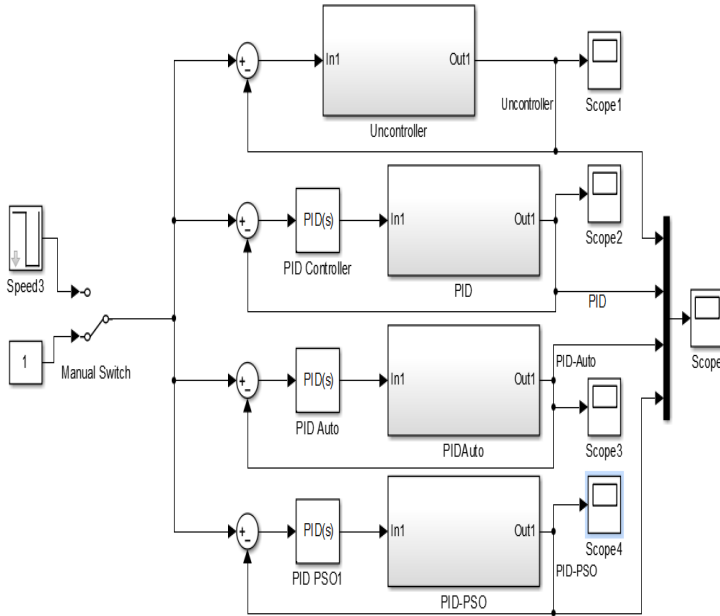
B. Desain Controller

Desain yang digunakan merupakan desain yang dibuat menggunakan aplikasi simulink matlab 2013a. Desain yang disusun sudah mencakup berbagai kondisi *controller* tersebut, yaitu PID tanpa kontrol, PID standar/konvensional, PID *auto tuning*, dan PID dengan metode PSO. Berikut adalah gambar desain *controller* :



Gambar 10.1 Desain beberapa *controller* suhu pada *furnace* dengan *reference*

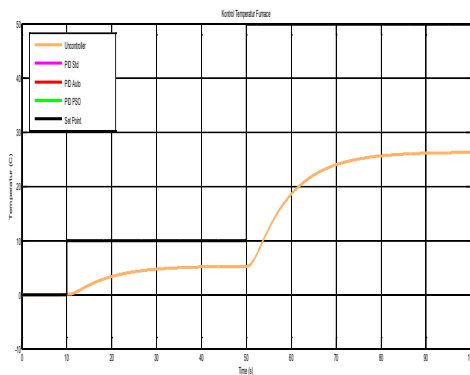
Gambar 10.1 menunjukkan desain beberapa *controller* beserta *reference* untuk mengatur *set point* yang akan diinginkan terhadap temperatur *furnace*. Berikut adalah desain beberapa *controller* tanpa menggunakan *reference* untuk menentukan *set point*.



Gambar 10.2 Desain beberapa *controller* suhu pada *furnace* tanpa menggunakan *reference*

C. Respon sistem

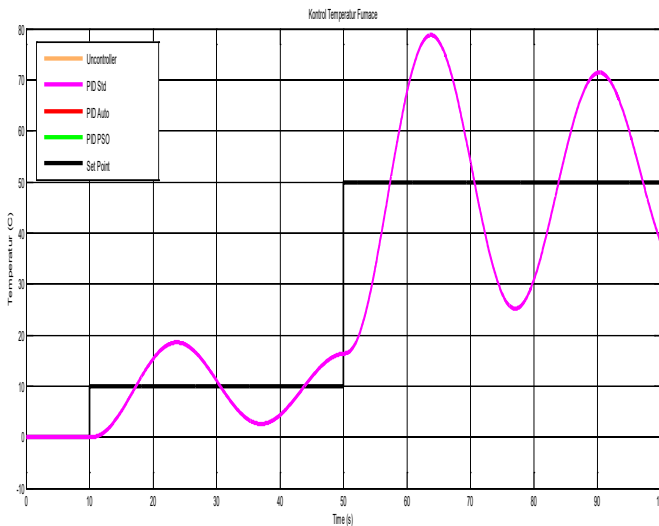
1. Respon sistem PID tanpa *controller*



Gambar 10.3 Gambar respon PID Tanpa Kontrol

Dari gambar 10.3 merupakan hasil dari simulasi PID tanpa kontroler. Perubahan suhu menunjukkan bahwa model tanpa kontroler tidak memiliki nilai K_p , K_i , dan K_d dengan nilai *overshoot* nol (0), *undershoot* 0.474, dan tidak memiliki *settling time* karena tidak pernah mencapai titik atau *setpoint* yang diinginkan sampai pada detik ke 100. Dengan hasil tersebut, PID tanpa *controller* masih kurang tepat jika digunakan sebagai pengontrol temperatur furnace.

D. Respon sistem PID konvensional

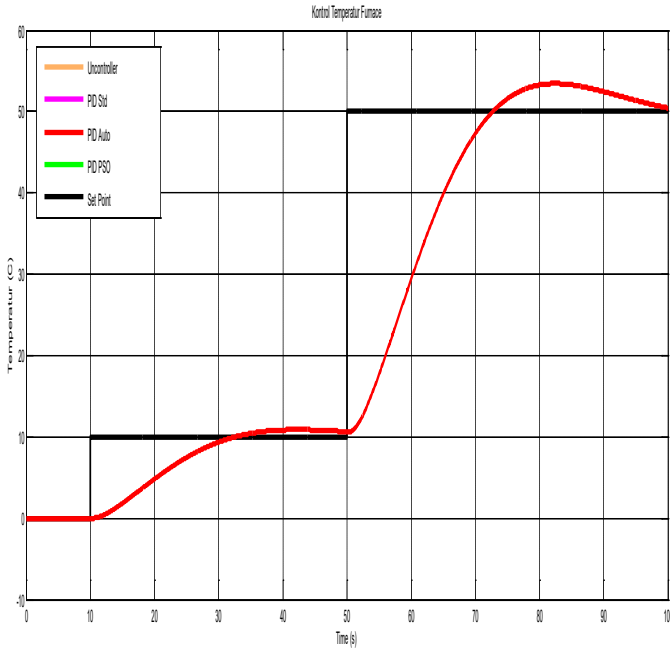


Gambar 10.4 Gambar respon PID standart/konvensional

Dari gambar 4.4 merupakan hasil dari Metode PID-Konv (konvensional) dengan nilai $K_p=1$, $K_i=1$, dan $K_d= 0$,

dengan hasil *overshoot* = 1.2872, *undershoot* = 0.0161, dan *settling time* pada 38.4422 detik.

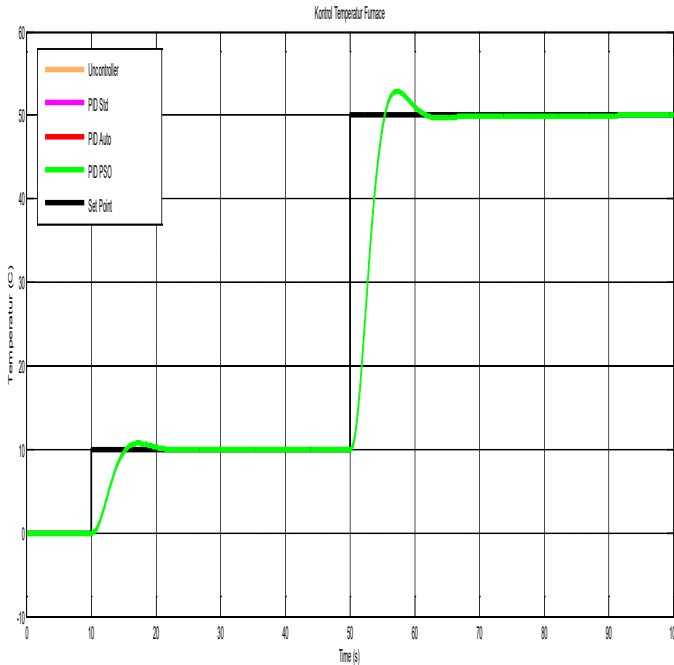
E. Respon sistem PID Auto tuning



Gambar 10.5 Gambar respon PID *Auto Tuning*

Dari gambar 10.5 merupakan hasil dari Metode PID-Auto (tuning matlab 2013a) dengan $K_p= 4.2875$, $K_i= 0.2875$, $K_d= 1.1256$, dengan hasil *overshoot* = 1.0716, *undershoot* = 0.0277, dan *settling time* pada 36.5342 detik.

F. Respon PID tuning PSO

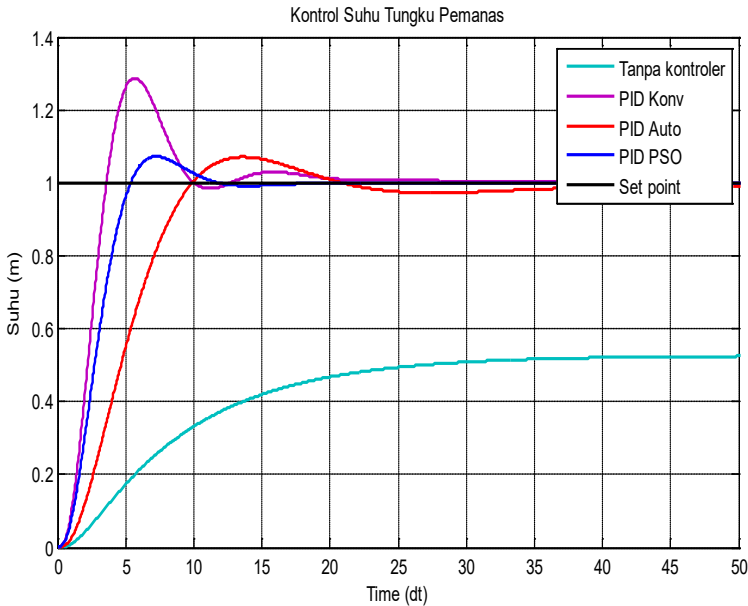


Gambar 10.6 Gambar PID dengan Tuning PSO

Dari gambar 4.6 merupakan hasil dari metode PID-PSO dengan nilai konstanta $K_p = 7.0177$, $K_i = 0.3486$, $K_d = 0.4323$ menghasilkan *overshoot* = 0.0721, *undershoot* 0.0081, dan *settling time* pada 30.4283 detik.

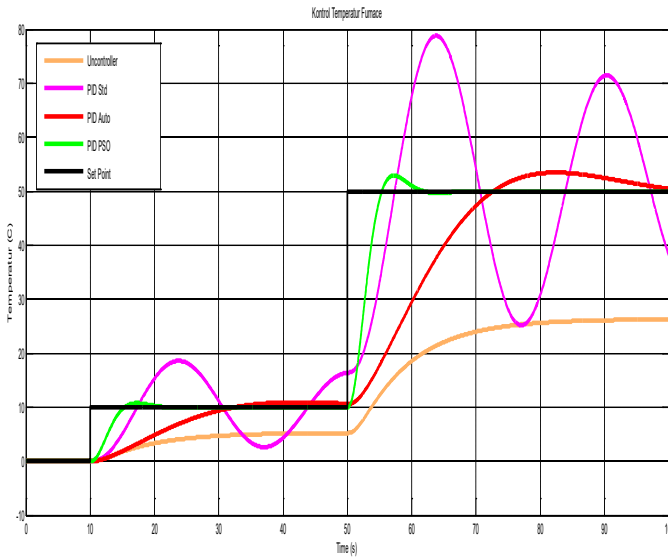
G. Hasil perbandingan untuk semua tanggapan

Respon *transien*/karakteristik kinerja dinamis dihitung dan ditabulasi seperti yang ditunjukkan pada *Gambar 10.7* sebagai berikut :



Gambar 10.7 Respon sistem berbagai kondisi tuning pada switch 1

Dari *Gambar 10.7* menunjukkan bahwa kontroler PID-Auto mempunyai *overshoot* terkecil hampir sama dengan PID-PSO, *undershoot* terkecil pada PID-PSO dan *settling time* tercepat juga pada PID-PSO. Jika diterapkan pada *setpoint* kenaikan suhu sesungguhnya dapat dilihat pada *Gambar 10.7*.



Gambar 10.8 Respon sistem dengan berbagai kondisi tuning menggunakan reference atau switch 2

Dari *Gambar 10.8* menunjukkan bahwa pada setiap pergantian tahapan kenaikan *set poin* temperatur terjadi lonjakan temperatur untuk sebagian *controller*. Pada kontroler PID-Auto mempunyai *overshoot* terkecil yang nilainya hampir sama dengan PID-PSO, *undershoot* terkecil terletak pada kontroler PID-PSO dan *settling time* tercepat juga terdapat pada PID-PSO. Sedangkan untuk PID standart dan PID tanpa *controller* menghasilkan nilai yang masih kurang baik. *Overshoot* dan *undershoot* yang terjadi dapat dilihat pada table berikut :

Tabel 10.1 Tabel respon masing-masing *controller*

Parameter	Uncontroller	PID-Konv	PID-Auto	PID-PSO
Kp	-	1	42.875	70.177
Ki	-	1	0.1769	0.3486
Kd	-	0	11.256	0.4323
Overshoot	0	0.2872	0.0720	0.0721
Undershoot	0.474	0.0161	0.0277	0.0081
Settling time	~	384.422	365.342	304.283

Dari tabel diatas menunjukkan respon masing-masing *controller* terhadap perubahan suhu menunjukkan bahwa :

- Model tanpa kontroler tidak memiliki nilai kp, ki, dan kd dengan nilai *overshoot* nol (0), *undershoot* 0.474, dan tidak memiliki *settling time* karena tidak pernah mencapai titik yang diinginkan sampai pada detik ke 100.
- Metode PID-Konv (konvensional) dengan nilai Kp = 1, Ki = 1, dan Kd = 0, dengan hasil *overshoot* = 1.2872, *undershoot* = 0.0161, dan *settling time* pada 38.4422 detik.
- Metode *PID-Auto* (tuning matlab 2013a) dengan Kp = 4.2875, Ki = 0.2875, Kd = 11.256, dengan hasil *overshoot* = 10.716, *undershoot* = 0.0277, dan *settling time* pada 36.5342 detik.
- Sedangkan pada metode PID-PSO dengan nilai konstanta Kp = 7.0177, Ki= 0.3486, Kd= 0.4323

menghasilkan *overshoot* = 0.0721, *undershoot* 0.0081, dan *settling time* pada 30.4283 detik.

BAB XI

SIMPULAN PID-PSO

Dari hasil simulasi yang ditunjukkan oleh *Gambar 10.1* dan *Table 10.1* menunjukkan bahwa :

1. Metode optimasi terbaik dihasilkan oleh metode PID-PSO dengan menghasilkan *overshoot* = 0.0721, *undershoot* 0.0081, dan *settling time* pada 30.4283 detik. Meskipun pada PID-Auto menghasilkan nilai *overshoot* yang sedikit lebih kecil, akan tetapi *settling time* dan *undershoot*nya cukup besar.
2. Dari hasil simulasi menunjukkan, PID tanpa *controller* memiliki hasil yang sangat jauh dari temperatur yang diinginkan, sehingga kurang efisien jika digunakan untuk control proses pada temperatur *furnace*.

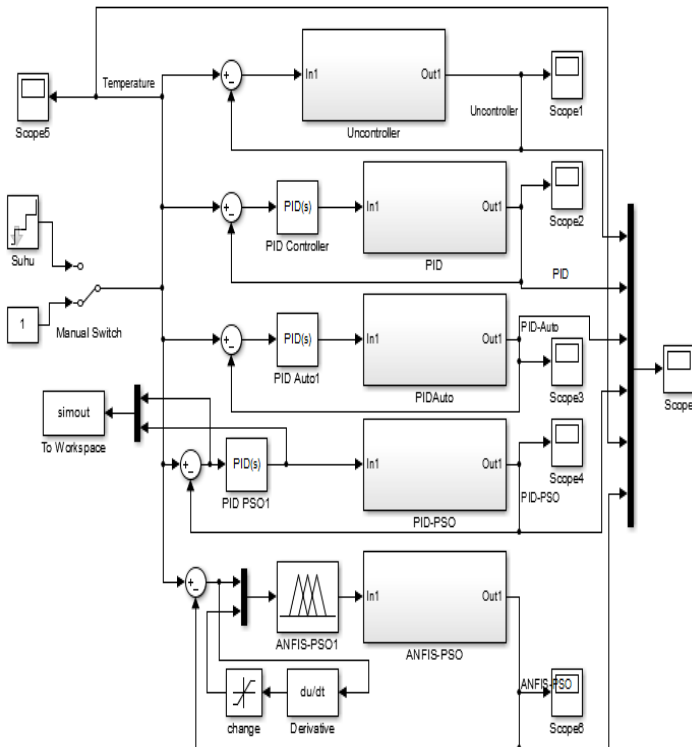
Dengan diterapkan metode PID-PSO terhadap *heating furnace*, maka akan dapat menghasilkan temperatur yang stabil, dan pada saat ada perubahan *setpoint* temperatur, maka metode PID-PSO akan cepat tercapai pada perubahan temperatur yang di *setting* sehingga dari segi penggunaan bahan bakar dapat memiliki tingkat efisiensi yang tinggi.

BAB XII

ANFIS-PSO

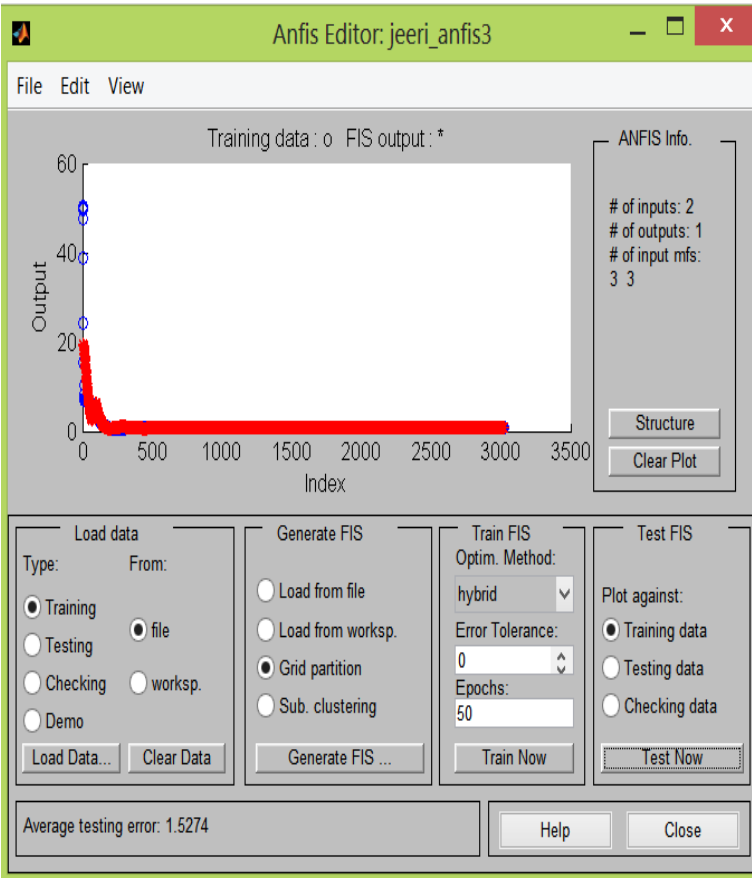
A. Desain ANFIS-PSO

Desain beberapa pengaturan suhu pada tungku pemanas diberikan pada Gambar 12.1, dengan sakelar pada posisi 1.



Gambar 12.1 Desain beberapa pengaturan suhu pada tungku pemanas

Data hasil pelatihan ANFIS dapat dilihat pada Gambar 12.2.

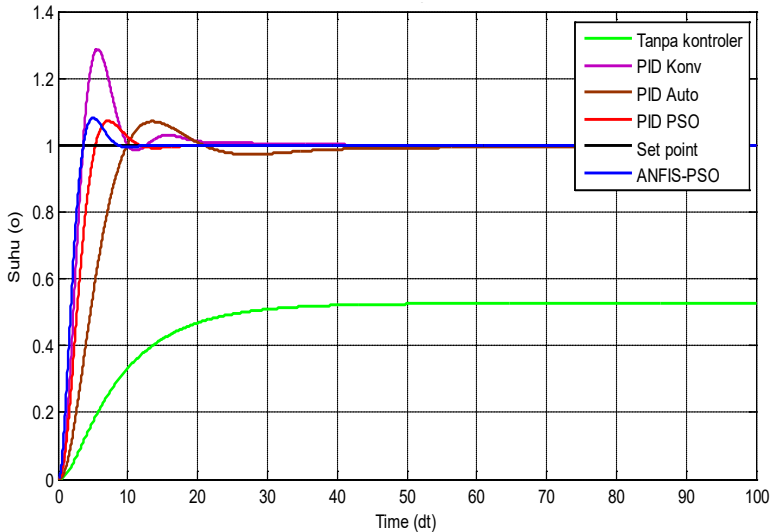


Gambar 12.2 Data hasil pelatihan ANFIS

B. Hasil Respon Simulink

Hasil simulasi metode Hybrid ANFIS-PSO yang diusulkan akan dibandingkan dengan beberapa metode lainnya, yaitu Conventional PID (PID-Konv), Matlab Autotuning PID (PID-Auto), dan PSO Tuned PID (PID-

PSO). Hasil perbandingan untuk semua tanggapan sementara dari karakteristik kinerja dinamis dihitung dan ditabulasi seperti yang ditunjukkan pada Gambar.11.2.

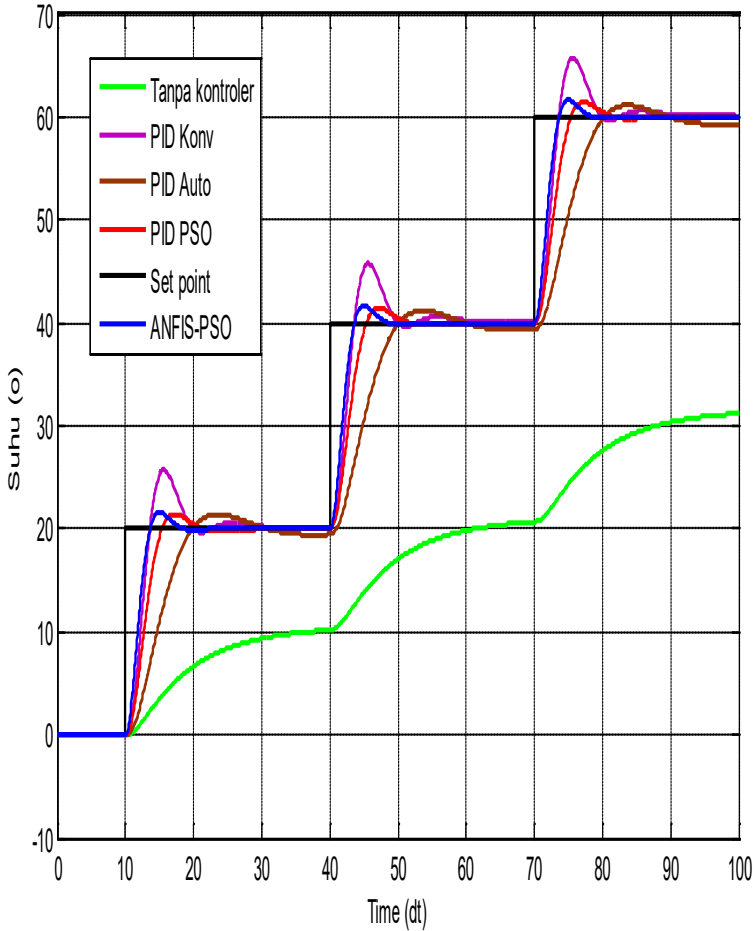


Gambar 11.2 Respon kontrol suhu tungku pemanas pada posisi sakelar 1

Seperti terlihat pada Gambar 11.2, kontroler PID-Auto memiliki overshoot terkecil yang hampir sama dengan kontroler PID-PSO dan ANFIS-PSO. Undershoot terkecil terdapat pada PID-PSO dan settling time tercepat juga terdapat pada ANFIS-PSO. Jika diterapkan pada titik setel, kenaikan suhu aktual pada posisi sakelar 2 dapat dilihat pada Gbr.11.3.

Seperti ditunjukkan pada Gambar 11.3, kontroler PID-Auto memiliki overshoot terkecil yang hampir sama dengan kontroler PID-PSO dan ANFIS-PSO. Undershoot terkecil diperoleh dengan menggunakan PID-PSO dan

settling time tercepat juga pada ANFIS-PSO. Overshoot dan undershoot yang terjadi dapat dilihat pada Tabel 1.



Gambar 12.3 Respon kontrol suhu tungku pemanas pada posisi sakelar 2

Tabel 12.1. Respon setiap pengontrol terhadap perubahan suhu

	Un control led	PID- Konv	PID- Auto	PID- PSO	Anfis- PSO
Kp	-	1	4.2875	7.0177	-
Ki	-	1	0.1769	0.3486	-
Kd	-	0	1.1256	0.4323	-
Over shoot	0	0.2872	0.0720	0.0721	0.0722
Under shoot	0.4742	0.0161	0.0277	0.0081	0.0085
Settling time	~	38.4422	36.5342	30.4283	18.8789

Seperti terlihat pada Gambar 12.3. dan Tabel 12.1, overshoot terkecil pada PID-Auto adalah 0,0720, undershoot terkecil pada PID-PSO adalah 0,0081, dan settling time tercepat pada ANFIS-PSO adalah 18,8789 detik. Perbedaan antara overshoot dan undershoot relatif kecil. Semua metode menunjukkan kesiapannya untuk implementasi. Namun, settling time ANFIS-PSO jauh lebih singkat dengan nilai overshoot dan undershoot yang relatif sebanding, sehingga ini menjadi metode optimasi terbaik untuk dipilih. Meskipun PID-Auto menghasilkan nilai overshoot yang sedikit lebih kecil, settling time dan undershoot cukup besar, sehingga PID-PSO dipilih sebagai metode terbaik untuk mengontrol suhu tungku pemanas.

C. Simpulan ANFIS-PSO

Perancangan dan implementasi metode ANFIS-PSO yang diusulkan untuk mengontrol suhu tungku pemanas dalam penelitian ini menghasilkan beberapa kesimpulan yang dibandingkan dengan metode lain yang dipertimbangkan (PID Konvensional (PID-Konv), Matlab Autotuning PID (PID-Auto), dan PID yang disetel PSO (PID-PSO)), ANFIS-PSO menawarkan pilihan terbaik. Meskipun metode PID-Auto memberikan overshoot terkecil dan PID-PSO menawarkan undershoot terkecil, tetapi ANFIS-PSO menghasilkan waktu penyelesaian tercepat dengan nilai overshoot dan undershoot yang hampir sama, dan memastikan metode terbaik untuk mengoptimalkan tungku. suhu.

DAFTAR PUSTAKA

- [1] M. Ali, A. Raikhani, H. Sopian, and I. Umami, "Optimasi Pengaturan Kecepatan Motor Shunt Berbasis Imperialist Competitive Algorithm (ICA)," *J. Intake J. Penelit. Ilmu Tek. dan Terap.*, vol. 9, no. 1, pp. 26–31, Apr. 2020, doi: 10.32492/jintake.v9i1.756.
- [2] M. Ali, A. Raikhani, B. Budiman, and H. Sopian, "Algoritma Persaingan Imperialis Sebagai Optimasi Kontroler PID dan ANFIS Pada Mesin Sinkron Magnet Permanen," *JEEE-U (Journal Electr. Electron. Eng.*, vol. 3, no. 1, pp. 57–81, Apr. 2019, doi: 10.21070/jeee-u.v3i1.2023.
- [3] M. Z. Bogatova and S. I. Chibizova, "Statistical modeling of temperature operating modes of heating furnaces for hot strip mills," *Izv. Ferr. Metall.*, vol. 64, no. 5, pp. 374–381, 2021, doi: 10.17073/0368-0797-2021-5-374-381.
- [4] V. Feliu-Batlle and R. Rivas-Perez, "Control of the temperature in a petroleum refinery heating furnace based on a robust modified Smith predictor," *ISA Trans.*, vol. 112, pp. 251–270, 2021, doi: 10.1016/j.isatra.2020.12.006.
- [5] T. Hidayat, "DESAIN UNTUK MANUFaktur HUSK FURNACE PADA MESIN PENERING GABAH TIPE VERTICAL DRYING," *ENGINEERING*, 2015.

- [6] B. Sun, H. Wang, and J. Zhou, "Analysis on quenching oxidation of workpiece in a controlled atmosphere box-type furnace," *Jinshu Rechuli/Heat Treat. Met.*, vol. 39, no. 4, pp. 153–155, 2014, doi: 10.13251/j.issn.0254-6051.2014.04.040.
- [7] H. K. Lemke and W. K. Stephenson, "Deposit formation in industrial delayed coker/visbreaker heaters," *Pet. Sci. Technol.*, vol. 16, no. 3–4, pp. 335–360, 1998, doi: 10.1080/10916469808949787.
- [8] D. Froud, A. Beale, T. O'Doherty, and N. Syred, "Studies of helmholtz resonance in a swirl burner/furnace system," *Symp. Combust.*, vol. 26, no. 2, pp. 3355–3362, 1996, doi: 10.1016/S0082-0784(96)80183-4.
- [9] Y. Kwon *et al.*, "Low noise dual-mode sensor analog front-end for capacitive and resistive microsensors," 2020, doi: 10.1109/ICEIC49074.2020.9051379.
- [10] G. R. P. Yudha, "Perekaman Sensor Aktif Dan Pasif Untuk Klasifikasi Hutan-Non Hutan," *Perbandingan Pemanfaat. Citra Satelit Has. Perekaman Sens. Aktif Dan Pasif Untuk Klasifikasi Hutan-Non Hutan*, 2018.
- [11] N. R. Pohan and R. Rasyid, "Rancang Bangun Sistem Kipas Otomatis Menggunakan Sensor PIR dan Sensor Suhu LM35," *J. Fis. Unand*, vol. 10, no. 1, pp. 104–110, 2021, doi: 10.25077/jfu.10.1.104-110.2021.
- [12] I. Islahudin, F. Riadi, R. Rosdaniah, and Y. Y. Yustina Yuyun, "RANCANG BANGUN SENSOR TEKANAN BERBASIS KOIL DATAR UNTUK MENGUKUR TEKANAN HIDROSTATIS AIR BENDUNGAN RAWAN BANJIR DI WILAYAH

- LOMBOK NUSA TENGGARA BARAT," *ORBITA J. Kajian, Inov. dan Apl. Pendidik. Fis.*, vol. 5, no. 1, p. 1, 2019, doi: 10.31764/orbita.v5i1.893.
- [13] M. Arrohman, R. Fajardika, M. Muhlasin, and M. Ali, "Optimasi Frekuensi Kontrol pada Sistem Hybrid Wind-Diesel Menggunakan PID Kontroler Berbasis ACO dan MFA," *J. Rekayasa Mesin*, vol. 9, no. 1, pp. 65–68, May 2018, doi: 10.21776/ub.jrm.2018.009.01.10.
- [14] Budiman and M. Ali, "Modifikasi Firefly Algorithm Untuk Partial Shading pada Photovoltaic," *ALINIER J. Artif. Intell. Appl.*, vol. 2, no. 1, pp. 22–27, 2021, doi: 10.36040/aliner.v2i1.3534.
- [15] M. Ali, A. N. Afandi, A. Parwati, R. Hidayat, and C. Hasyim, "DESIGN OF WATER LEVEL CONTROL SYSTEMS USING PID AND ANFIS BASED ON FIREFLY ALGORITHM," *JEEMECs (Journal Electr. Eng. Mechatron. Comput. Sci.)*, vol. 2, no. 1, 2019, doi: 10.26905/jeemecs.v2i1.2804.
- [16] A. Raikhani, M. Ali, D. Ajiatmo, and Budiman, "Desain Optimal Automatic Voltage Regulator Pada Pembangkit Listrik Mikro Hidro Menggunakan Fuzzy Logic Controller," *J. Intake*, vol. 7, no. 1, pp. 30–39, 2016, [Online]. Available: <http://ejournal.undar.ac.id/index.php/intake/article/view/384>.
- [17] M. Farul, R. Rukslin, M. Andrik, and I. Cahyono, "Optimasi Load Frequency Control (LFC) Pada Sistem Pembangkit Listrik Tenaga Mikro Hidro Berbasis PID-ANFIS," *J. Rekayasa Mesin*, vol. 9, no. 1,

- pp. 61-64, 2018, doi: 10.21776/ub.jrm.2018.009.01.9.
- [18] M. Ali, Muhlasin, H. Nurohmah, A. Raikhani, H. Sopian, and N. Sutantra, "Combined ANFIS method with FA, PSO, and ICA as Steering Control Optimization on Electric Car," in *2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)*, Oct. 2018, pp. 299-304, doi: 10.1109/EECCIS.2018.8692885.
- [19] M. Ali *et al.*, "The comparison of dual axis photovoltaic tracking system using artificial intelligence techniques," *IAES Int. J. Artif. Intell.*, vol. 10, no. 4, pp. 901-909, 2021, doi: 10.11591/IJAI.V10.I4.PP901-909.
- [20] M. Ali, Budiman, A. R. Sujatmika, and A. A. Firdaus, "Optimization of controller frequency in wind-turbine based on hybrid PSO-ANFIS," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1034, no. 1, p. 12070, Feb. 2021, doi: 10.1088/1757-899X/1034/1/012070.
- [21] H. Hartanto, A. Rochmansyah, Kadaryono, and H. Nurohmah, "Desain Optimasi Kecepatan Motor Induksi Menggunakan Hybrid ANFIS (Adaptive Neuro Fuzzy Inference System) Dengan PID-PSO (Proportional Integral Derivative-Particle Swarm Optimization)," *SENTIA-2016, Polinema, Malang*, vol. 8, no. 2, pp. 44-48, 2016.

INDEX

A

Algorithm · 52
algoritma · 48, 51, 52
Algoritma · 52
anfis · 107, *See*
Anfis · 46
ANFIS · 47, 50, 51, 52
ANFIS structure · 49
Artificial Intelligent · 1

B

Burner · 11, 12

C

center of area · 44
Controller · i, ii, 2, 27, 32,
33, 69, 78, 82

D

Data · 107
Defuzzifikasi · 44
distribusi · 44

E

EIASC · 52
EKMANI · 52

F

FLC · 107
FLS · 52

Fuzzy · 42, 43, 44, 52
Fuzzy Logic Controller ·
107

G

gravitasi · 44

H

Heating · i, 2
hybrid · 51

K

KMA · 52
Konvergensi · 59, 60, 61

L

logic · 107

M

membership function · 42

O

Overshoot · 75, 76, 91

P

parameter · 48, 50, 51
Parameter · 24, 52, 56, 57,
68, 69, 70, 75, 81, 91
Particle Swarm
Optimization · 53
PID · i, ii, 1, 22, 23, 27, 69,
70, 71, 74, 75, 77, 78, 79,

80, 82, 84, 85, 86, 87, 88,
89, 90, 91, 92, 93
PID-Auto · 105
PID-Korv · 105
PID-PSO · 105
PSO · i, ii, 2, 53, 54, 55, 56,
57, 58, 59, 60, 61, 62, 63,
64, 65, 66, 68, 69, 70, 74,
78, 79, 80, 81, 82, 88, 89,
90, 91, 92, 93

R

Recursive Least Square
Estimator · 51

RLSE · 51, 52

Rule · 44

S

Sensor · 12, 13, 14, 15, 16,
17, 18, 19, 20, 72

Settling time · 26, 75, 91

sinyal · 47, 48

Sistem kontrol · 38

Struktur · 42, 45, 46

system · 42, 47

BIOGRAFI PENULIS



Machrus Ali

Machrus Ali adalah Dosen Teknik Elektro Univ. Darul Ulum, Jombang sejak Tahun 1998, Lulus S1 tahun 1996 di Teknik Elektro Univ. Darul Ulum Jombang, Lulus S2 tahun 2015 Magister Teknik Elektro di ITS

Surabaya, memperoleh gelar IPM dari PII tahun 2017. Aktif meneliti di bidang Kecerdasan Buatan, Energi, Sistem Kontrol, dan Sistem Tenaga Listrik. Aktif publikasi ilmiah internasional bereputasi, nasional terakreditasi, baik sebagai penulis, reviewer. Aktif dan menjadi pengurus organisasi Forum Pendidikan Tinggi Teknik Elektro (FORTEI) Reg. VII Jawa Timur dan Persatuan Insinyur Indonesia (PII). Sekarang sedang menjabat sebagai Kepala LPPM UN DAR.



Hidayatul Nurohmah

Hidayatul Nurohmah adalah Dosen Teknik Elektro Univ. Darul Ulum, Jombang. Lulus S1 tahun 1993 di Teknik Elektro di Universitas Darul Ulum, Jombang, Lulus S2 tahun 2015 di Magister Teknik Elektro di ITS

Surabaya. Aktif meneliti dan menulis di bidang Sistem Kontrol, Renewable Energi, Pembangkit Tenaga Listrik, dan Sistem Proteksi.



Rukslin

Rukslin adalah Dosen Teknik Elektro Univ. Darul Ulum, Jombang. Lulus D3 Telkom PENS-ITS tahun 2001, Lulus S1 Teknik Elektro Universitas Darul Ulum, Jombang tahun 2013, Lulus S2 Teknik Elektro, Unisula,

Semarang tahun 2016. Aktif meneliti di bidang Elektronika, Sistem Kontrol, Fisika, dan Sistem Tenaga Listrik.



Dwi Ajiatmo

Dwi Ajiatmo adalah Dosen Teknik Elektro Univ. Darul Ulum tahun 1996, lulus S1 Tekni Elektro Univ. Darul Ulum, Lulus S2 Univ. Gadjah Mada Yogyakarta tahun 2004, Lulus S3 Teknik Elektro di ITS urabaya.

Aktif meneliti di bidang Sistem Kontrol, Sistem Distribusi Tenaga Listrik, dan Analisa Sistem Tenaga Listrik



Muhammad Agil Haikal

Muhammad Agil Haikal adalah Dosen Teknik Elektro Univ. Hang Tuah Surabaya Lulus S1 pada Tahun 2008, Lulus S2 tahun 2015 Magister Teknik Elektro di ITS Surabaya, Aktif meneliti di bidang Energi, Sistem Kontrol, dan Sistem Tenaga Listrik.