TUGAS AKHIR - TF 181801

PERANCANGAN ESTIMATOR KECEPATAN BERBASIS *PARTICLE FILTER* UNTUK *SPEED SENSORLESS CONTROL* PADA MOTOR INDUKSI

EGY JOSUA SIMBOLON
NRP. 02311640000128

Dosen Pembimbing:
Dr. Katherin Indriawati, S.T.,M.T.

Departemen Teknik Fisika
Fakultas Teknologi Industri Dan Rekayasa Sistem
Institut Teknologi Sepuluh Nopember
Surabaya
2020

TUGAS AKHIR - TF 181801

PERANCANGAN ESTIMATOR KECEPATAN BERBASIS *PARTICLE FILTER* UNTUK *SPEED SENSORLESS CONTROL* PADA MOTOR INDUKSI

EGY JOSUA SIMBOLON
NRP. 02311640000128

Dosen Pembimbing:
Dr. Katherin Indriawati, S.T.,M.T.

Departemen Teknik Fisika
Fakultas Teknologi Industri Dan Rekayasa Sistem
Institut Teknologi Sepuluh Nopember
Surabaya
2020

Halaman ini sengaja dikosongkan

FINAL PROJECT - **TF 181801**

PARTICLE FILTER BASED SPEED ESTIMATOR FOR SPEED SENSORLESS CONTROL IN INDUCTION MOTOR

EGY JOSUA SIMBOLON
NRP. 02311640000128

Supervisors:
Dr. Katherin Indriawati, S.T.,M.T.

Department Of Engineering Physics
Faculty of Industrial Technology and System Engineering
Institut Teknologi Sepuluh Nopember
Surabaya
2020

Halaman ini sengaja dikosongkan

PERNYATAAN BEBAS PLAGIASI

Saya yang bertanda tangan di bawah ini.

Nama	: Egy Josua Simbolon
NRP	: 02311640000128
Departemen / Prodi	: Teknik Fisika / S1 Teknik Fisika
Fakultas	: Fakultas Teknologi Industri & Rekayasa Sistem (FT-IRS)
Perguruan Tinggi	: Institut Teknologi Sepuluh Nopember

Dengan ini menyatakan bahwa Tugas Akhir dengan judul "***PERANCANGAN ESTIMATOR KECEPATAN BERBASIS PARTICLE FILTER UNTUK SPEED SENSORLESS CONTROL PADA MOTOR INDUKSI***" adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain. Apabila di kemudian hari terbukti terdapat plagiat pada Tugas Akhir ini, maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenarnya-benarnya.

Surabaya, 2 Agustus 2020

Yang membuat pernyataan,

METERAI TEMPEL

2064BAHF520523615

6000

ENAM RIBU RUPIAH

Egy Josua Simbolon

NRP. 02311640000128

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

TUGAS AKHIR

PERANCANGAN ESTIMATOR KECEPATAN BERBASIS *PARTICLE FILTER* UNTUK *SPEED SENSORLESS CONTROL* PADA MOTOR INDUKSI

Oleh:

Egy Josua Simbolon

NRP. 02311640000128

Surabaya, 4 Agustus 2020

Menyetujui,
Pembimbing I

Dr. Katherin Indriawati, S.T., M.T.
NIP. 19760523 200012 2 001

Mengetahui,
Kepala Departemen
Teknik Fisika FT-IRS

Dr. Suyanto, S.T., M.T.
NIP. 19711113 199512 1 002

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

PERANCANGAN ESTIMATOR KECEPATAN BERBASIS *PARTICLE FILTER* UNTUK *SPEED SENSORLESS CONTROL* PADA MOTOR INDUKSI

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
pada
Progam Studi S-1 Departemen Teknik Fisika
Fakultas Teknologi Industri & Rekayasa Sistem (FT-IRS)
Institut Teknologi Sepuluh Nopember

Oleh:

EGY JOSUA SIMBOLON

NRP. 02311640000128

Disetujui oleh Tim Penguji Tugas Akhir:

1. Dr. Katherin Indriawati, S.T., M.T. …… (Pembimbing I)
2. Dr. Bambang Lelono W., S.T., M.T. …… (Ketua Penguji)
3. Moh. Kamalul Wafi, S.T., M.Sc.DIC …… (Penguji I)
4. Detak Yan Pratama, S.T., M.Sc. …… (Penguji II)
5. Dr. Suyanto, S.T., M.T …… (Penguji III)

SURABAYA

2020

Halaman ini sengaja dikosongkan

PERANCANGAN ESTIMATOR KECEPATAN BERBASIS *PARTICLE FILTER* UNTUK *SPEED SENSORLESS CONTROL* PADA MOTOR INDUKSI

Nama	**:**	**Egy Josua Simbolon**
NRP	**:**	**02311640000128**
Departemen	**:**	**Teknik Fisika FTIRS - ITS**
Dosen Pembimbing	**:**	**Dr. Katherin Indriawati, S.T., M.T**

ABSTRAK

Pada penilitian ini dilakukan perancangan speed sensorless control untuk motor induksi dengan menggunakan algoritma *Particle Filter* (PF) sebagai estimator kecepatan dan *Direct Torque Control* (DTC) sebagai skema kontrol yang dipakai untuk pengendalian kecepatan motor induksi. Algoritma *Particle Filter* bekerja dengan cara memanfaatkan suatu set partikel yang memiliki bobot disebut dengan sampel untuk mendapatkan hasil estimasi dari suatu sistem yang bersifat nonlinier. Pada perancangan ini DTC digunakan dengan nilai Kp sebesar 0,1 dan Ki sebesar 0,01. Pada uji coba dengan sistem *open loop* didapatkan respon kecepatan motor induksi sebesar 157.1 rad/s dalam keadaan *steady state*, dan keadaan ini dicapai pada detik ke 0,2. Hasil yang sama ditunjukkan juga oleh algortima *Particle Filter*, yang diamana hal ini menunjukkan bahwa *Particle Filter* dapat digunakan sebagai estimator kecepatan. Untuk pengujian dengan sistem kontrol *close loop*, rentang pengujian yaitu pada kecepatan 50 hingga 500 rpm, dengan error steady state yang didapatkan lebih kecil dari 5%. Berdasarkan nilai *error* yang diperoleh dapat dikatakann bahwa perancangan *speed sensorless control* untuk motor induksi sudah berjalan dengan baik, namun demikian pada kecepatan tinggi seperti 350 dan 500 rpm hasil yang diapat memiliki nilai *overshoot* yang tinggi dan *settling time* yang lama.

Kata Kunci: *Direct Torque Control*, Motor Induksi, *Particle Filter*, *Speed Sensorless Control*

Halaman ini sengaja dikosongkan

PARTICLE FILTER BASED SPEED ESTIMATOR FOR SPEED SENSORLESS CONTROL IN INDUCTION MOTOR

Name	**:**	**Egy Josua Simbolon**
NRP	**:**	**02311640000128**
Department	**:**	**Engineering Physics FT-IRS - ITS**
Supervisors	**:**	**Dr. Katherin Indriawati, S.T., M.T.**

ABSTRACT

In this research, speed sensorless control design for induction motors is done by using the Particle Filter algorithm (PF) as a speed estimator and Direct Torque Control (DTC) as a control scheme used to control the speed of an induction motor. The Particle Filter algorithm works by utilizing a set of particles that have a weight called a sample to get the estimation results from a nonlinear system. In this design DTC is used with a Kp value of 0.1 and Ki of 0.01. In trials with the open loop system, the speed of the induction motor response is 157.1 rad / s in a steady state, and this state is reached in seconds to 0.2. The same result was also shown by the Particle Filter algorithm, which indicates that the Particle Filter can be used as a speed estimator. For testing with a close loop control system, the test range is at speeds of 50 to 500 rpm, with a steady state error obtained less than 5%. Based on the value of the error obtained it can be said that the design of the speed sensorless control for the induction motor is running well, however at high speeds such as 350 and 500 rpm the results obtained have a high overshoot value and a long settling time.

Keywords: Direct Torque Control, Induction Motor, Particle Filter, Speed Sensorless Control

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji dan syukur kita panjatkan kepada Tuhan Yang Maha Esa karen berkat dan rahmat-Nya sehingga dapat terselesaikan Tugas Akhir dengan judul "Perancangan Estimator Kecepatan Berbasis *Particle Filter* untuk *Speed Sensorless Control* pada Motor Induksi"

Ucapan terima kasih juga disampaikan kepada semua pihak yang telah membantu dalam menyelesaikan tugas akhir ini:

1. Bapak Dr. Suyanto, S.T., M.T. selaku Kepala Departemen Teknik Fisika ITS dan selaku dosen wali yang senantiasa memberikan bimbingan kepada penulis
2. Ibu Dr. Katherin Indriawati, S.T., M.T. selaku pembimbing Tugas Akhir
3. Bapak dann Ibu dosen Teknik Fisika yang telah memberikan ilmu selama kuliah
4. Kedua orang tua (Bapak Dapot Simbolon dan Ibu Juita Nainggolan) dan adik (Gabriel Simbolon, Michael Simbolon, dan Petrik Simbolon) serta segenap keluarga atas kasih sayang dan dukungan dalam segala bentuk.
5. Teman seperjuangan dalam mengerjakan Tugas Akhir bidang minat Rekayasa Instrumentasi.
6. Kepada Mas Syahrul Munir, Mas A. Ibnu Hija, Nimran Saladin, Bariz Irfan Ridho, Agung Prastowo, Faishal A., Viko Dian Nano, yang telah memberikan dukungan dan semangat tanpa henti.
7. Teman-teman asisten Laboratorium Fisika Rekayasa.
8. Seluruh teman-teman Angkatan 2016 yang turur memberikan dukungannya.
9. Semya pihak yang terlibat dalam pengerjaan tugas akhir ini.

Saran dan kritik yang membangun sangat diharapkan guna penyempuranaan laporan tugas akhir ini. Semoga laporan in dapat menambah wawasan yang bermanfaat bagi pembacanya.

Surabaya, 21 Juni 2020

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

COVER PAGE iii

PERNYATAAN BEBAS PLAGIASI v

LEMBAR PENGESAHAN vii

LEMBAR PENGESAHAN ix

ABSTRAK xi

ABSTRACT xiii

KATA PENGANTAR xv

DAFTAR ISI xvii

DAFTAR GAMBAR xix

DAFTAR TABEL xxi

BAB I PENDAHULUAN 1

1.1 Latar Belakang 1

1.2 Rumusan Masalah 2

1.3 Tujuan 2

1.4 Batasan Masalah 3

1.5 Sistematika Laporan 3

BAB II TINJAUAN PUSTAKA DAN DASAR TEORI 5

2.1 Motor Induksi 5

2.1.1 Konstruksi Motor Induksi 6

2.1.2 Model Matematis Motor Induksi 8

2.2 *Particle Filter* 9

2.3 Direct Torque Control 10

BAB III METODOLOGI PENELITIAN 13

3.1 Pemodelan motor induksi 14

3.2 Pemodelan *Particle Filter* 16
3.3 Perancangan *Speed Sensorless Control* 19
BAB IV HASIL DAN PEMBAHASAN 21
4.1 Hasil Pemodelan Motor Induksi 21
4.2 Pengujian *Particle Filter* 23
4.3 Pengujian *Speed Sensorless Control* 26
4.4 Hasil Perbandingan Model dan *Controller* 31
BAB V KESIMPULAN DAN SARAN 33
5.1 Kesimpulan 33
5.2 Saran 33
DAFTAR PUSTAKA 35
LAMPIRAN 37
BIODATA PENULIS 43

DAFTAR GAMBAR

Gambar 2. 1 Motor Induksi (Thomson & Culbert, 2017) 5
Gambar 2. 2 Contoh stator tanpa lilitan (Thomson & Culbert, 2017)..... 6
Gambar 2. 3 Contoh stator lengkap dengan lilitan (Thomson & Culbert, 2017) .. 6
Gambar 2. 4 Skema dari tambaga *squirrel cage* rotor (Thomson & Culbert, 2017) 7
Gambar 2. 5 Rotor *squirrel cage* tembaga (Thomson & Culbert, 2017) 7
Gambar 2. 6 Rangkaian ekivalen 'Γ' motor induksi (Chan & Shi, 2011)..... 8
Gambar 2. 7 Rangkaian ekivalen 'T' motor induksi (Chan & Shi, 2011)..... 9
Gambar 2. 8 Skema kontrol DTC untuk motor induksi tanpa sensor kecepatan (Abu-Rub, Iqbal, & Guzinski, 2012) 11
Gambar 3. 1 Diagram Alir Penelitian……13
Gambar 3. 2 Dagram block *particle filter* 17
Gambar 3. 3 Diagram Blok *Speed Sensorless Control*..... 19
Gambar 4. 1 Respon Arus Motor Induksi……21
Gambar 4. 2 Respon Fluks Motor Induksi 22
Gambar 4. 3 Respon Kecepatan Motor Induksi 22
Gambar 4. 4 Respon Arus Motor Induksi dan Estimasi PF 23
Gambar 4. 5 Respon Fluks Motor Induksi dan Estimasi PF 24
Gambar 4. 6 Respon Kecepatan Motor Induksi dan Estimasi PF 24
Gambar 4. 7 Respon kecepatan motor induksi (Firdausi, 2020)..... 25
Gambar 4. 8 Respon Kecepatan Set Point 50 dengan *Particle Filter* 26
Gambar 4. 9 Perbandingan Respon Kecepatan *Plant* dan PF Set Point 50..... 26
Gambar 4. 10 Respon Kecepatan Set Point 100 dengan Particle Filter 27
Gambar 4. 11 Perbandingan Respon Kecepatan Plant dan PF Set Point 100 27
Gambar 4. 12 Respon Kecepatan Set Point 150 dengan *Particle Filter* 27
Gambar 4. 13 Perbandingan Respon Kecepatan Plant dan PF Set Point 150 28
Gambar 4. 14 Respon Kecepatan Set Point 350 dengan Particle Filter 28
Gambar 4. 15 Perbandingan Respon Kecepatan Plant dan PF Set Point 350 28
Gambar 4. 16 Respon Kecepatan Set Point 500 dengan Particle Filter 29
Gambar 4. 17 Perbandingan Respon Kecepatan Plant dan PF Set Point 500 29

Gambar 4. 18 Respon kecepatan motor dengan *FPGA hardware* untuk motor induks (Mohammedi, Kabache, Moulahoum, & Houassine, 2015) 31
Gambar 4. 19 Perbandingan DTC Sensor dengan *sensorless* 32

DAFTAR TABEL

Tabel 3. 1 Parameter Motor Induksi 16
Tabel 3. 2 Parameter *Particle Filter* 17
Tabel 4. 1 Data kecepatan model motor dan pengukuran..... 23
Tabel 4. 2 Hasil uji performansi masing-masing kecepatan 29
Tabel 4. 3 Beban Komputasi dengan Jumlah Partikel Berbeda..... 30

Halaman ini sengaja dikosongkan

BAB I
PENDAHULUAN

1.1 Latar Belakang

Motor induksi tiga fasa sudah sangat luas digunakan di sektor perindustrian, terutama untuk mengoperasikan pompa, kompresor, dan kipas. Secara umum sebuah motor beroperasi dibawah setidaknya 60% dari beban nominalnya (Guedes & al., 2018). Motor induksi merupakan motor elektrik yang sangat baik, memiliki struktur yang simple, kuat, dan sangat mudah untuk dirawat.

Pada saat ini aplikasi dari driver motor induksi dengan inverter sudah sangat jelas meningkat. Pada desain control *driver*, keduanya metode *scalar* dan *vector control* telah diaplikasikan secara luas. Terutama respon kecepatan, ruang lingkup operasi, dan akurasi kecepatan dari perkembangan terbaru lebih baik dibandingkan dengan sebelumnya (Wang & Fang, 2003). Tenkik *vector control* telah digunakan secara luas untuk *driver* motor induksi pada performansi yang tinggi, teknik mencapai *decoupling* yang efektif antara torsi dan fluks, yang memperbolehkan control kecepatan dan torsi pada performansi dinamis maupun statis yang tinggi, dan control yang baik pada keadaan transien dan *steady state* (Zerikat, Mechernene, & Chekroun, 2010). DTC (*Direct Torque Control*) merupakan salah satu jenis teknik *vector control* yang digunakan luas dalam industry, diantara banyaknya proposal mengenai algoritma kontrol pada motor induksi, DTC telah diterima secara luas, perusahaan penjual *driver* motor induksi yang menggunakan DTC sudah beredar di pasaran selama beberapa tahun. Berbeda dengan jenis *vector control* lainnya seperti FOC (*Field Oriented Control*), DTC tidak cenderung mereproduksi sifat elektromekanikal dari *drive* motor dc, tetapi langsung ditujukan pada eksploitasi menyeluruh dari kemampuan produksi flux dan torsi dari motor induksi yang diberi input PWM inverter. Dalam artian, DTC dapat dipandang sebagai konsep yang baru dalam mengontrol *drive* motor induksi (Buja, Casadei, & Serra, 1997).

Ada permintaan yang luas dari industri untuk teknologi *sensorless control* yang tidak memerlukan sensor kecepatan ataupun sensor posisi, untuk mendapatakan

beberapa keuntungan seperti meningkatkan keandalan dari *driving system*, penghilangan kabel sensor, mengurangi ukuran motor dll. (Shinnaka, 2002). Pada *Sensorless Induction Motor* diperlukan algoritma untuk mendapatkan *state* dari motor induksi tersebut yang nantinya digunakan untuk masuk ke sistem kontrol DTC. Dalam (Barut, Bogosyan, & Gokasan, 2007) telah digunakan EKF (*Extende Kalman Filter*), dimana algoritma EKF ini digunakan sebagai estimator dan menggunakan teknik kontrol FOC dan DTC. Pada penelitian kali ini estimator yang digunakan adalah algoritma *Particle filter*. *Particle filter* merupakan metode komputasional yang membuka peluang untuk interferensi sistematis dalam nonlinier/non-Gaussian *state space models*. *Particle Filters* merupakan metode *Sequential Monte Carlo* (*SMC*) yang paling terkenal (Schon, 2014). Oleh karena motor induksi merupakan sistem yang non linier maka pada tugas akhir ini akan dirancang *Particle Filter* sebagai estimator *state* dari motor induksi dan nantinya untuk kontrol kecepatannya akan digunakan metode DTC.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, maka rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

a) Bagaimana merancang sistem *speed sensorless control* untuk motor induksi dengan *particle filter* sebagai estimator kecepatan?
b) Bagaimana perbandingan performansi dari skema kontrol *Direct Torque Control*?

1.3 Tujuan

Berdasarkan rumusan masalah tersebut, maka tujuan pada tugas akhir ini adalah sebagai berikut

a) Merancang sistem *speed sensorless control* untuk motor induksi dengan *particle filter* sebagai estimator kecepatan.
b) Mendapatkan hasil perbandingan performansi dari skema kontrol *Direct Torque Control*.

1.4 Batasan Masalah

Berdasarkan identifikasi masalah, maka Batasan masalah pada penelitian ini adalah sebagai berikut:

a) Data yang digunakan adalah data arus dan tegangan masukan motor induksi
b) Validasi hasil estimasi *Particle Filter* mengacu pada hasil model motor induksi saat motor induksi beroperasi pada *close loop.*
c) Parameter motor induksi yang digunakan adalah parameter dari motor induksi penelitian sebelumnya.

1.5 Sistematika Laporan

Laporan terdiri dari 5 BAB. BAB I berisikan latar belakang, rumusan masalah, tujuan, batasan masalah, dan sistematika laporan. BAB II berisi tinjauan pustaka dan dasar teori yang mendukung analisa dan pembahasan. BAB III berisi metodologi penelitian yang digunakan dalam tugas akhir ini. BAB IV berisi analisa data dan pembahasan data yang didapatkan. BAB V berisi hasil dan kesimpulan yang dapat diambil dari tugas akhir ini serta saran yang dapat diberikan untuk membantu penelitian selanjutnya.

Halaman ini sengaja dikosongkan

BAB II
TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Motor Induksi

Motor Induksi tiga fasa digunakan secara luas dalam hampir semua aplikasi terutama dikarenakan harganya yang murah dan kekuatannya, serta harga perawatan yang rendah, membuat motor induksi menjadi menjadikan motor induksi sebagai solusi yang menarik dibandingkan dengan alternatif lainnya (Pereira, et al., 2019).

Motor induksi 3 fasa menghasilkan torsi dan *power* pada output *shaft* akibat dari kombinasi 3 fenomena dasar yaitu:

a) Hasil dari medan magnet yang berotasi yang berasal dari 3 lilitan fasa yang di-*supply* oleh tegangan 3 fasa.
b) Gaya gerak listrik (emf) diinduksi dalam koil yang ditempatkan dalam medan magnet yang berubah, dalam hal ini, medan magnet yang berputar.
c) Koil yang membawa arus dan berada dalam medan magnet akan menghasilkan gaya (Thomson & Culbert, 2017).

Gambar 2. 1 Motor Induksi (Thomson & Culbert, 2017)

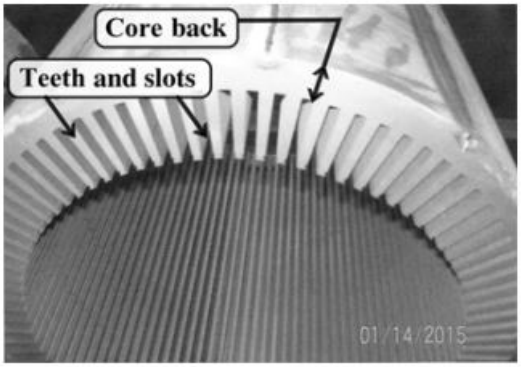
Motor induksi tiga fasa (Gambar 2.1) memiliki konstruksi yang hampir sama dengan motor listrik jenis lainnya. Motor ini memiliki dua bagian utama, yaitu stator yang merupakan bagian yang diam, dan rotor sebagai bagian yang berputar.

Antara bagian stator dan rotor dipisahkan oleh celah udara yang sempit, dengan jarak yang hanya berkisar dari 0,4 mm sampai 4 mm (disesuaikan dengan daya motor).

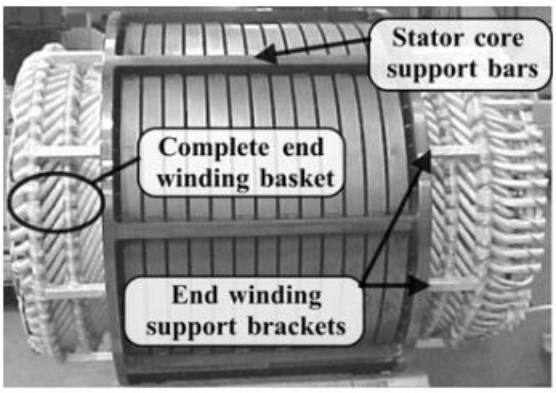
2.1.1 Konstruksi Motor Induksi

Stator adalah bagian dari motor yang tidak bergerak atau statis. tegangan seimbang 3 fase (setiap fase berjarak 120 derajat) disuplai ke lilitan stator 3 fase yang seimbang (dengan setiap fase didistribusikan secara spasial 120 derajat terpisah) menghasilkan medan magnet berputar hanya dalam satu arah. Inti stator, dimana lilitan 3-fase dimasukkan, terbuat dari laminasi baja listrik berkualitas tinggi untuk mengurangi kerugian arus eddy (Thomson & Culbert, 2017).

Berikut dapat dilihat pada Gambar 2.3 dan 2.4 contoh dari stator motor yang belum dimasukkan lilitannya (Gambar 2.3) dan setelah stator dimasukkan lilitan (Gambar 2.4)

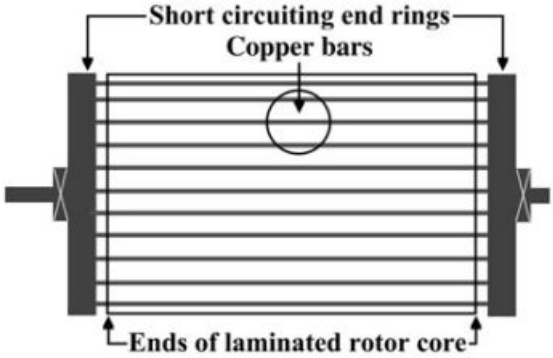


Gambar 2. 2 Contoh stator tanpa lilitan (Thomson & Culbert, 2017)

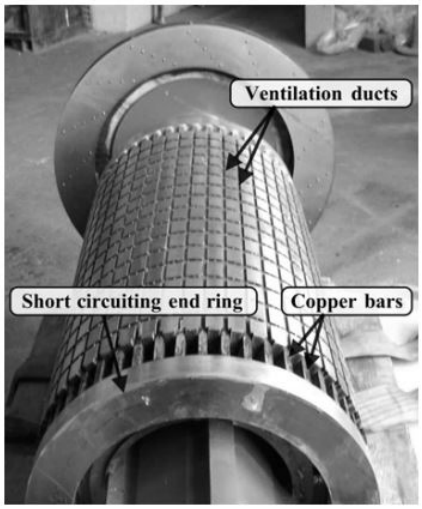


Gambar 2. 3 Contoh stator lengkap dengan lilitan (Thomson & Culbert, 2017)

Rotor merupakan bagian yang bergerak di dalam motori induksi. Rotor in berbentuk sangkar dimana sangkar ini terdiri dari batang tembaga yang digabungkan pada cincin dikedua ujungnya. Pada rotor terdapat arus yang mengalir, hal ini dapat dicapai hanya dengan melakukan hubungan arus pendek pada konduktor dan emf yang diinduksi dan akan mendorong arus melaluinya. Hasil dari satu set konduktor pembawa arus ini dicapai dengan menggunakan satu set batang tembaga, yang disatukan (melalui pemanasan atau dipatri) untuk membuat arus pendek pada cincin penghubung diilustrasikan dalam Gambar 2.4, dan gambar 2.5 adalah contoh dari rotor.



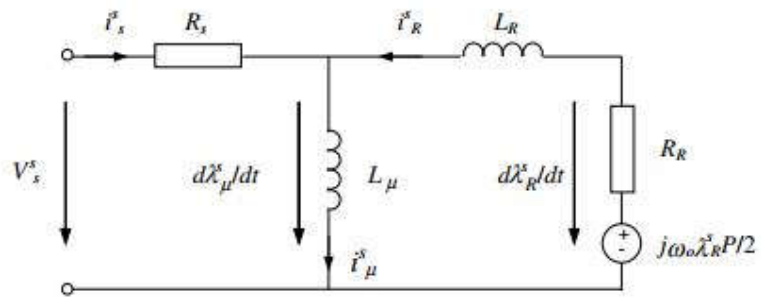
Gambar 2. 4 Skema dari tambaga *squirrel cage* rotor (Thomson & Culbert, 2017)



Gambar 2. 5 Rotor *squirrel cage* tembaga (Thomson & Culbert, 2017)

2.1.2 Model Matematis Motor Induksi

Model motor induksi berdasarkan pada rangkaian ekivalen 'Γ' dalam kerangka referensi stator dapat dinyatakan sebagai persamaan nonlinear orde lima (Trzynadlowski, 1994):



Gambar 2. 6 Rangkaian ekivalen 'Γ' motor induksi (Chan & Shi, 2011)

$$\frac{d\omega_o}{dt} = \frac{2}{3}\frac{P}{2J}\left(\lambda^s_{d\mu} i^s_{qs} - \lambda^s_{q\mu} i^s_{ds}\right) - \frac{T_L}{J} \tag{2.5}$$

$$\frac{d\lambda^s_{d\mu}}{dt} = -R_s i^s_{ds} + V^s_{ds} \tag{2.6}$$

$$\frac{d\lambda^s_{q\mu}}{dt} = -R_s i^s_{qs} + V^s_{qs} \tag{2.7}$$

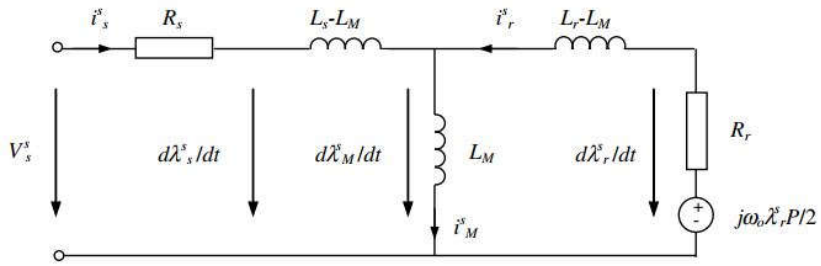
$$\begin{aligned}\frac{di^s_{ds}}{dt} = &-k_\Gamma\left(L_\mu R_R + L_R R_S\right) i^s_{ds} + \frac{P}{2}\omega_o i^s_{qs} + k_\Gamma R_R \lambda^s_{d\mu} \\ &+ \frac{P}{2} k_\Gamma \omega_o \lambda^s_{q\mu} + k_\Gamma L_R V^s_{ds}\end{aligned} \tag{2.8}$$

$$\begin{aligned}\frac{di^s_{qs}}{dt} = &-k_\Gamma\left(L_\mu R_R + L_R R_S\right) i^s_{qs} - \frac{P}{2}\omega_o i^s_{ds} + k_\Gamma R_R \lambda^s_{q\mu} \\ &- \frac{P}{2} k_\Gamma \omega_o \lambda^s_{d\mu} + k_\Gamma L_R V^s_{qs}\end{aligned} \tag{2.9}$$

Dimana $k_\Gamma = \frac{1}{L_\mu^2 - L_\mu L_R}$, dan torsi elektromagnetik $T = \frac{2}{3}\frac{P}{2}\left(\lambda^s_{d\mu} i^s_{qs} - \lambda^s_{q\mu} i^s_{ds}\right)$.

Pada persamaan (2.5) hingga persamaan (2.9), kecepatan rotor ω_o dan fluks stator ($\lambda^s_{d\mu}, \lambda^s_{q\mu}$), dan arus stator ($i^s_{ds}, i^s_{qs}$) merupakan *states*, sementar inersia rotor J, induktansi rotor dan stator (L_μ, L_R), resistansi stator dan rotor (R_S, R_R), dan jumlah *pole* P merupakan parameter. Tegangan (V^s_{ds}, V^s_{qs}) merupakan input, dan T_L adalah beban torsi.

Untuk motor induksi dengan rangkaian ekuivalen 'T' dapat ditunjukkan melalui persamaan orde lima yang lain dengan referensi stator:



Gambar 2. 7 Rangkaian ekivalen 'T' motor induksi (Chan & Shi, 2011)

$$\frac{d\omega_o}{dt} = \frac{2}{3}\frac{PL_M}{2JL_r}\left(\lambda^s_{dr} i^s_{qs} - \lambda^s_{qr} i^s_{ds}\right) - \frac{T_L}{J} \tag{2.10}$$

$$\frac{d\lambda^s_{dr}}{dt} = -\frac{R_r}{L_r}\lambda^s_{dr} - \frac{P}{2}\omega_o\lambda^s_{qr} + \frac{R_r L_M}{L_r} i^s_{ds} \tag{2.11}$$

$$\frac{d\lambda^s_{qr}}{dt} = -\frac{R_r}{L_r}\lambda^s_{qr} + \frac{P}{2}\omega_o\lambda^s_{dr} + \frac{R_r L_M}{L_r} i^s_{qs} \tag{2.12}$$

$$\begin{aligned}\frac{di^s_{ds}}{dt} = {} & \frac{L_M R_r}{(L_r L_s - L_M^2)}\lambda^s_{dr} + \frac{PL_M}{2(L_r L_s - L_M^2)}\omega_o\lambda^s_{qr} \\ & - \frac{L_M^2 R_r + L_r^2 R_s}{(L_r L_s - L_M^2)L_r} i^s_{ds} + \frac{L_r}{(L_r L_s - L_M^2)} V^s_{ds}\end{aligned} \tag{2.13}$$

$$\begin{aligned}\frac{di^s_{qs}}{dt} = {} & \frac{L_M R_r}{(L_r L_s - L_M^2)}\lambda^s_{qr} - \frac{PL_M}{2(L_r L_s - L_M^2)}\omega_o\lambda^s_{dr} \\ & - \frac{L_M^2 R_r + L_r^2 R_s}{(L_r L_s - L_M^2)L_r} i^s_{qs} + \frac{L_r}{(L_r L_s - L_M^2)} V^s_{qs}\end{aligned} \tag{2.14}$$

Pada persamaan (2.10) hingga persamaan (2.14), kecepatan rotor ω_o dan fluks rotor ($\lambda^s_{d\mu}, \lambda^s_{q\mu}$), dan arus stator ($i^s_{ds}, i^s_{qs}$) merupakan *states*, sementara inersia rotor J, induktansi stator dan rotor ($L_\mathrm{s}, L_\mathrm{r}$), induktansi mutual L_M, resistansi stator dan rotor ($R_\mathrm{S}, R_\mathrm{R}$), dan jumlah *pole* P merupakan parameter. Tegangan (V^s_{ds}, V^s_{qs}) merupakan input, dan T_L adalah beban torsi (Chan & Shi, 2011)

2.2 *Particle Filter*

Algoritma *Particle Filter* adalah implementasi rekursif dari metode pemrosesan sinyal statistik berdasarkan Monte Carlo, *Monte Carlo Method* merupakan suatu metode algoritma komputasi yang mengandalkan sampel acak secara berulang untuk medapatkan hasil numerik. Metode *particle filter* ini mendekati fungsi kepadatan probabilitas posterior Bayesian (pdf) dengan

seperangkat sampel berbobot yang dipilih secara acak. Setiap sampel dari *state vector* disebut sebagai partikel. Sejumlah partikel yang cukup besar menjamin konvergensi hampir mendekati ke fungsi distribusi probabilitas sebenarnya (Gyorgy, Kelemen, & David, 2014). Metode *Sequential Importance Sampling* (SIS) adalah kerangka kerja dasar untuk *particle filter*. Ide utamanya adalah untuk mewakili *posterior density* yang dibutuhkan oleh sekelompok sampel random dengan berat terkait dan menghitung estimasi berdasarkan sampel dan bobot ini. Karena jumlah sampel menjadi sangat besar, karakterisasi MC ini menjadi representasi yang setara dengan deskripsi fungsional biasa dari posterior pdf, dan filter SIS mendekati perkiraan Bayesian optimal (Arulampalam, Maskell, Gordon, & Clapp, 2002).

Algoritma dari *particle filter ini* dapat diringkas pada bagian bawah ini. Algoritma ini di presentasikan di (Gordon, Salmond, & Smith, 1993), dengan nama, *Bayesian bootstrap filter*:

a) Inisiasi N sampel partikel $\{x_{0|-1}^{(i)}\}_{i=1}^{N}$ dari $p(x_0)$.

b) Menghitung berat partikel

$$q_t^i = p(y_t|x_t^{(i)}) \tag{2.1}$$

dan normalisasi berat, *i.e.,*

$$q'^{(i)}_t = \frac{q_t^{(i)}}{\sum_{j=1}^{N} q_t^{(j)}}. \tag{2.2}$$

c) *Resample* dengan penggantian, $P(x_{0|-1}^{(i)})$

$$P\left(x_{t|t}^{(i)} = x_{t|t-1}^{(i)}\right) = q'^{(i)}_t \tag{2.3}$$

d) Memprediksi (*i.e.,* mensimulasikan) partikel baru dengan

$$x_{t+1|t}^{(i)} = f\left(x_{t|t}^{(i)}\right) + v_t^{(i)} + w_t^{(i)} \tag{2.4}$$

e) Diulangi mulai dari langkah b).

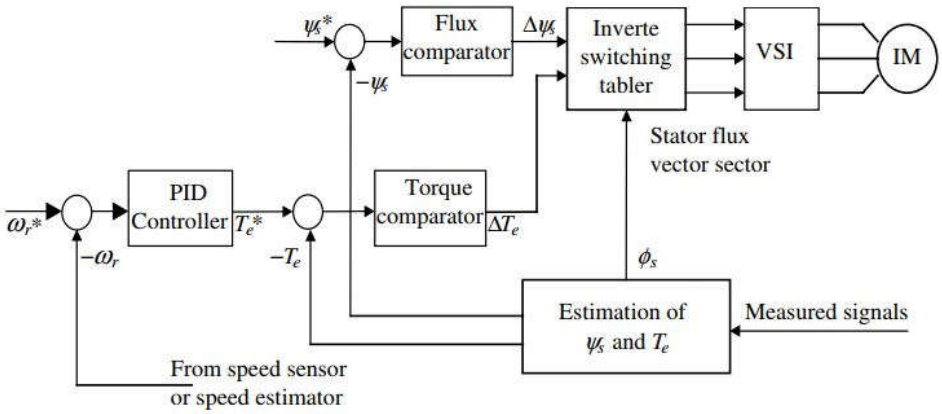
Particle filter dapat diinterpretasikan sebagai metode berbasis simulasi, *i.e.,* N jumlah lintasan yang memungkinkan, $x_t^{(i)}, i = 1, \dots, N$ disimulasikan. Berdasarkan pengukuran, masing-masing lintasan diberi bobot, $q'^{(i)}_t$.

2.3 Direct Torque Control

Inti dari sistem kontrol DTC adalah memiliki subsistem yang memiliki *torque* dan *flux hysteresis controller* dan logika *switching* inverter yang optimal. Model

motor induksi yang akurat juga penting, karena estimasi fluks stator dan torsi motor didasarkan pada model mesin dan pengukuran tegangan dan arus stator input motor induksi (Abu-Rub, Iqbal, & Guzinski, 2012).

Pada metode DTC ini, fluks stator dan torsi stator dikontrol secara langsung dengan memilih kondisi *switching* inverter yang sesuai. Pada pengontrol fluks stator yang memaksakan durasi waktu dari vektor tegangan aktif, yang memutar fluks stator bersama dengan lintasan referensi dan pada pengontrol torsi stator yang menentukan durasi waktu vektor tegangan nol yang menjaga torsi motor dalam dalam pita toleransi histeresis yang telah ditetapkan (Arya & Wadhani, 2011).



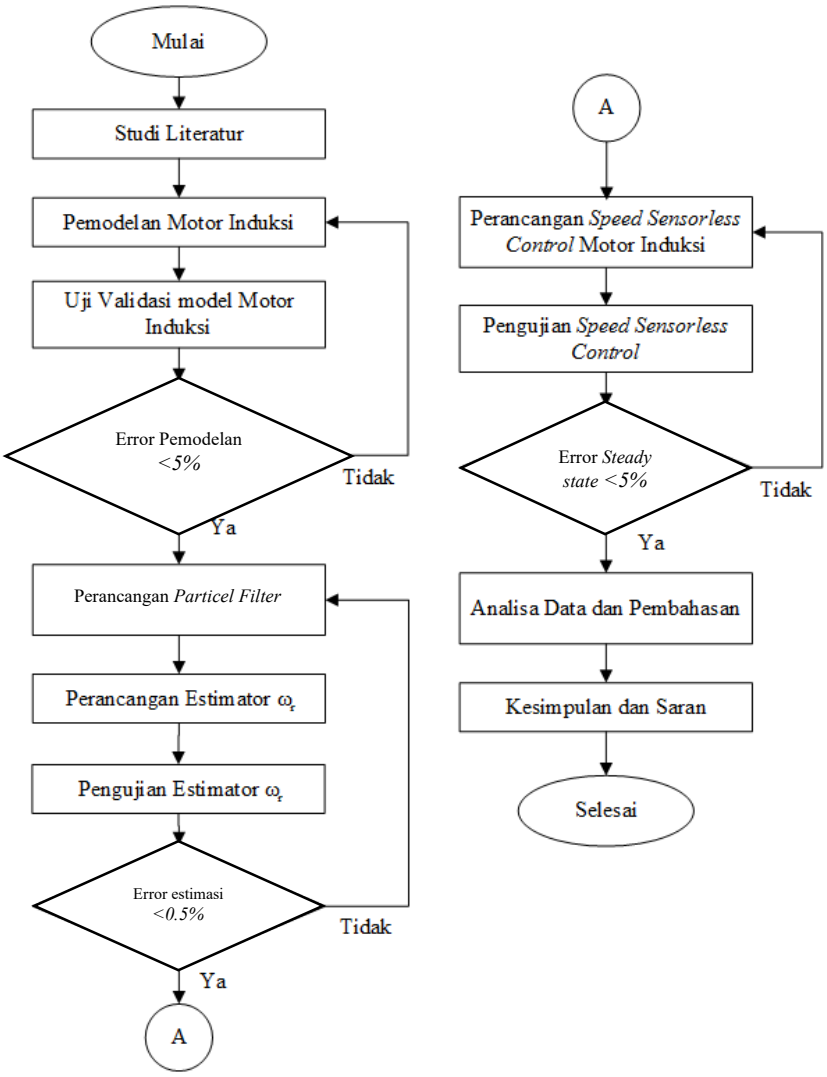
Gambar 2. 8 Skema kontrol DTC untuk motor induksi tanpa sensor kecepatan (Abu-Rub, Iqbal, & Guzinski, 2012)

Skema kontrol dari DTC untuk motor induksi dapat dilihat pada Gambar 2.8 untuk mode operasi kecepatan. Pada gambar dapat dilihat dua cabang parallel, masing-masing untuk amplitudo fluks stator dan kontrol torsi. Torsi referensi didapat dari output *speed controller*. Keduanya fluks stator dan *torque controller* adalah dalam bentuk tipe histerisis. *Drive* membutuhkan pengukuran yang sesuai yang akan memungkinkan estimasi fluks dan torsi stator untuk kontrol *loop* tertutup dari kedua kuantitas ini (Abu-Rub, Iqbal, & Guzinski, 2012).

Halaman ini sengaja dikosongkan

BAB III
METODOLOGI PENELITIAN

Metodologi penelitian yang digunakan dalam Tugas Akhir ini dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Diagram Alir Penelitian

Metodologi yang digunakan untuk mendapatkan data kecepatan motor induksi dengan input arus dan tegangan adalah dengan menggunakan estimator *particle filter*, karena motor induksi dijalankan dengan memakai estimator kecepatan dan bukannya sensor kecepatan, maka motor berjalan dengan sistem

sensorless. Untuk kontrol motor induksi digunakan DTC yang dimana DTC ini akan mememakai data yang didapat dari *particle filter* melalui *speed controller*.

3.1 Pemodelan motor induksi

Pemodelan motor induksi untuk penelitian tugas akhir ini menggunakan kerangka acuan stator dengan berdasarkan model "T" seperti pada Gambar 2.7. *States* yang didapatkan dari model ini adalah arus stator ($i_{\alpha s}^{s}, i_{\beta s}^{s}$), fluks rotor ($\lambda_{\alpha r}^{s}, \lambda_{\beta r}^{s}$) dan kecepatan motor. Sehingga persamaan untuk motor induksi berdasarkan persamaan (2.10) hingga persamaan (2.14) yang sebelumnya berada pada sumbu dq dapat ditulis ulang dengan menggunakan Transformasi Clarke dengan sumbu αβ sebagai berikut:

$$\frac{d\omega_o}{dt} = \frac{2}{3}\frac{PL_M}{2JL_r}\left(\lambda_{\alpha r}^{s} i_{\beta s}^{s} - \lambda_{\beta r}^{s} i_{\alpha s}^{s}\right) - \frac{T_L}{J} \tag{3.1}$$

$$\frac{d\lambda_{\alpha r}^{s}}{dt} = -\frac{R_r}{L_r}\lambda_{\alpha r}^{s} - \frac{P}{2}\omega_o \lambda_{\beta r}^{s} + \frac{R_r L_M}{L_r} i_{\alpha s}^{s} \tag{3.2}$$

$$\frac{d\lambda_{\beta r}^{s}}{dt} = -\frac{R_r}{L_r}\lambda_{\beta r}^{s} + \frac{P}{2}\omega_o \lambda_{\alpha r}^{s} + \frac{R_r L_M}{L_r} i_{\beta s}^{s} \tag{3.3}$$

$$\begin{aligned}\frac{di_{\alpha s}^{s}}{dt} = {} & \frac{L_M R_r}{(L_r L_s - L_M^2)}\lambda_{\alpha r}^{s} + \frac{PL_M}{2(L_r L_s - L_M^2)}\omega_o \lambda_{\beta r}^{s} \\ & - \frac{L_M^2 R_r + L_r^2 R_s}{(L_r L_s - L_M^2)L_r} i_{\alpha s}^{s} + \frac{L_r}{(L_r L_s - L_M^2)} V_{\alpha s}^{s}\end{aligned} \tag{3.4}$$

$$\begin{aligned}\frac{di_{\beta s}^{s}}{dt} = {} & \frac{L_M R_r}{(L_r L_s - L_M^2)}\lambda_{\boldsymbol{\beta r}}^{s} - \frac{PL_M}{2(L_r L_s - L_M^2)}\omega_o \lambda_{\alpha r}^{s} \\ & - \frac{L_M^2 R_r + L_r^2 R_s}{(L_r L_s - L_M^2)L_r} i_{\beta s}^{s} + \frac{L_r}{(L_r L_s - L_M^2)} V_{\beta s}^{s}\end{aligned} \tag{3.5}$$

Untuk pemodelan dengan mengguanakan simulink MATLAB, maka persamaan (3.1) – (3.5) akan diubah kedalam bentuk *state space*, untuk bentuk umum dari persamaan *state space* adalah sebagai berikut

$$\dot{x} = Ax + Bu \tag{3.6}$$

$$y = Cx + Du \tag{3.7}$$

Kemudian untuk matriks A, B, C, D, $\dot{x}$ dan u dapat dijadikan seperti dibawah ini:

$$A = \begin{bmatrix} -A_1 & 0 & A_2 & A_3 & 0 \\ 0 & -A_4 & -A_5 & A_6 & 0 \\ A_7 & 0 & -A_8 & -P * \omega_r/2 & 0 \\ 0 & A_9 & -P * \omega_r/2 & -A_{10} & 0 \\ -A_{11} * \lambda_{\beta r} & A_{12} * \lambda_{\alpha r} & 0 & 0 & 0 \end{bmatrix} \tag{3.8}$$

$$B = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{3.9}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{3.10}$$

$$D = [0] \tag{3.11}$$

Dimana:

$$A_1 = A_4 = \frac{Lm^2Rr + Lr^2Rs}{\sigma Lr^2Ls} \tag{3.12}$$

$$A_2 = A_6 = \frac{LmRr}{\sigma Lr^2Ls} \tag{3.13}$$

$$A_3 = A_5 = \frac{P\ Lm\ \omega r}{2\sigma LrLs} \tag{3.14}$$

$$A_7 = A_9 = \frac{LmRr}{Lr} \tag{3.15}$$

$$A_8 = A_{10} = \frac{Rr}{Lr} \tag{3.16}$$

$$A_{11} = A_{12} = \frac{2\ P\ Lm}{3\ 2J\ Lr} \tag{3.17}$$

$$\sigma = 1 - \frac{Lm^2}{Lr\ Ls} \tag{3.18}$$

$$B_1 = B_2 = \frac{1}{\sigma\ Ls} \tag{3.19}$$

Pada model algoritma *particle filter* yang akan digunakan menggunakan nilai masukan berupa arus dan tegangan, dimana nilai tegangan akan diambil langsung dari sumber yang nantinya terdapat pada model, sementara untuk arus akan didapat dari *state* yang dihasilkan oleh stator oleh karena itu bentuk matriks C seperti pada persamaan (2.6) yang disesuaikan dengan bentuk matriks x dibawah ini:

$$x = \begin{bmatrix} i_{\alpha s} \\ i_{\beta s} \\ \lambda_{ar} \\ \lambda_{\beta r} \\ \omega_r \end{bmatrix} \tag{3.20}$$

$$u = \begin{bmatrix} V_{\alpha s}^{s} \\ V_{\beta s}^{s} \end{bmatrix} \tag{3.21}$$

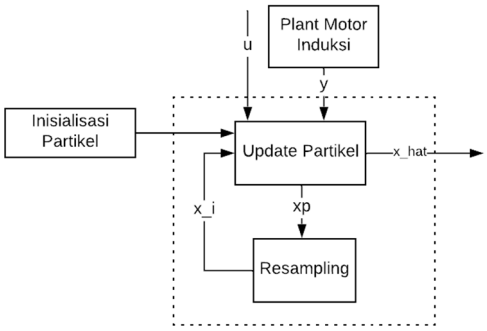
Pemodelan motor induksi diatas menggunakan parameter-parameter dari motor induksi yang nantinya akan dimasukkan kedalam MATLAB terdapat pada Table 3.1 dibawah ini.

Tabel 3. 1 Parameter Motor Induksi

No.	Parameter	Nilai
1	Indukstansi Mutual (L_M)	0.258 H
2	Induktansi Stator (L_s)	0.274 H
3	Induktansi Rotor (L_r)	0.274 H
4	Hambatan Stator (R_s)	4.850 Ω
5	Hambatan Rotor (R_r)	3.850 Ω
6	Inersia Rotor (J)	0.031 Kgm^2
7	*Pole* (P)	4
8	*Sampling Time* (T_s)	10^{-6}
9	Frekuensi (F)	50 Hz

3.2 Perancangan *Particle Filter*

Particle filter dirancang sebagai estimator kecepatan untuk motor induksi dengan menggunakan input dari tegangan dari inverter dan arus stator dari motor induksi. Proses yang terdapat dalam *particle filter* terjadi dalam bentuk domain diskrit sehingga tegangan dan arus input harus diubah terlebih dahulu menjadi bentuk diskrit, serta seluruh *process model* yang terdapat dalam algoritma *particle filter* diubah kedalam bentuk diskrit, diagram block *particle filter* dapat dilihat pada Gambar 3.2. Untuk perancangan *particle filter* dapat dilihat pada langkah-langkah berikut:



Gambar 3. 2 Dagram block *particle filter*

a) Inisialisasi partikel

Partikel yang digunakan pada algoritma ini diinisialisi secara random berdasarkan distribusi normal atau gaussian. Parameter yang digunakan pada tahap ini adalah sebagai berikut:

Tabel 3. 2 Parameter *Particle Filter*

No.	Parameter	Nilai
1	Jumlah Partikel (N)	500
2	*Initial state* (x_0)	[0 0 0 0 0]'
3	Kovarian (P_0)	0.1*I_5
4	*Process Noise* (Q)	1e-11
5	*Measurement Noise* (R)	[0.05^2 0; 0 0.05^2

Pada inisialisasi ini N partikel di-*generate* x_u dari distribusi normal Gaussian, yang dimana berarti *proposal density* adalah distribusi gaussian.

b) Update partikel

Untuk update partikel pada langkah selanjutnya berupa *predicted particle* xp yang didapat dari *proposal density* $q(\{x_p\}_t^i)$ bisa didapat dengan mendistribusikan setiap partikel *xp* dari waktu sebelumnya x_i melalui *process dynimanic* pada persamaan berikut ini:

$$x_p(i) = A_b * x_i(:,t) + B_b * u + sqrt(Q) * randn(n,1) \quad (3.23)$$

$$y_p(i) = C_b * x_i(:,t) \quad (3.24)$$

Kemudian untuk bobot partikel adalah perbandingan probabilitas distribusi dari *proposal density* dengan *target density*, atau secara sederhana perbandingan probabilitas antara hasil pengamatan dan pengukuran dari proses dalam persamaan berikut

$$\widetilde{w}(i) = e^{-\frac{1}{2}(\frac{y-y_p(i)}{R})^2} \quad (3.25)$$

Dimana:

$$y = C_d x + v \quad (3.26)$$

Kemudian untuk normalisasi bobot partikel digunakan persamaan berikut:

$$w(i) = \frac{\widetilde{w}(i)}{\sum_{i=1}^{N} \widetilde{w}(i)} \quad (3.27)$$

Kemudian untuk hasil estimasi didapatkan dengan menjumlahkan semua perkalian antara partikel dengan bobotnya masing-masing dengan persamaan berikut:

$$\hat{x} = \sum_{i=1}^{N} w(i).xp(i) \quad (3.28)$$

c) *Resampling* Partikel

Secara umum tahap *resampling* bertujuan untuk menghilangkan partikel dengan bobot yang rendah dan kemudian menghasilkan partikel baru dengan bobot w yang sama dan partikel baru tersebar sesuai dengan bobot w sebelumnya. Teknik *resampling* partikel yang digunakan dalam algortima ini adalah *Systematic Resample* yang pertama kali digunakan dalam (Kitagawa, 1996), konsep umum dari algoritma ini adalah dengan membandingkan *cumulative weight* (qc_s) yang didefenisikan sebagai ($qc(k)$) dengan sampel acak yang terdistribusi seragam $u(j)$, dimana $j = 1, \dots, M$ dimana M merupakan jumlah partikel yang di-*resample* dan juga merupakan merupakan panjang vektor w. Sampel $u(j)$ dan $qc(k)$ didapat dari:

$$u_j = \frac{(j-1)+\tilde{u}}{N} \ , \tilde{u} \sim U(0,1) \quad (3.29)$$

$$qc(k) = qc(k+1) + k \quad (3.30)$$

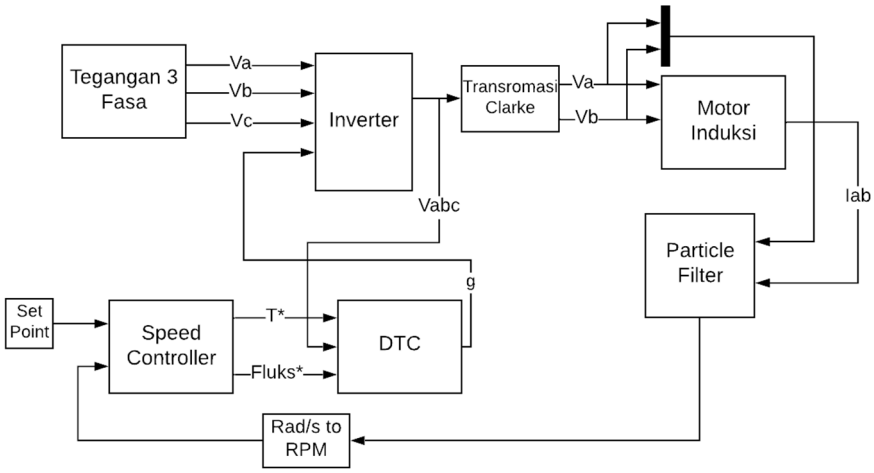
Apabila nilai dari $qc(k)$ lebih kecil dibandingkan dengan $u(j)$ maka di-*loop* ke persamaan (3.29) dengan penambahan nilai k, namun jika sebaliknya maka maka didapat nilai k yang nantinya nilai ini akan diapakai memilih partikel yang akan diestimasi, dan nilai j akan bertambah kemudian didapat nilai baru $u(j)$ dari persamaan (3.28). Hal ini akan terus di-*loop* hingga $j = M$. Algoritma ini memungkinkan partikel dengan nilai $qc(k)$ yang besar akan memiliki probabilitas tinggi untuk di-*resample*, namun tidak memungkinkan juga $qc(k)$ yang bernilai kecil di-*resample*. Apabila proses *resampling* selesai maka akan didapat set partikel baru x_i:

$$x_i(i) = xp(k) \tag{3.31}$$

Kemudian set partikel x_i yang nantinya melalui *block time* delay dan masuk kembali ke bagian *update partikel*.

3.3 Perancangan *Speed Sensorless Control*

Pada simulasi penelitian ini skema kontrol yang digunakan adalah *Direct Torque Control* (DTC). DTC sendiri memanfaatkan perhitungan fluks dan torsi elektromagnetik yang dihasilkan oleh motor induksi.



Gambar 3. 3 Diagram Blok *Speed Sensorless Control*

Gambar 3.2 diatas merupakan diagram blok *plant* secara keseluruhan. *Particle filter* bertindak sebagai *soft sensor* yang dimana output kecepatan rotornya sebelum memasuki *speed* controller akan dikonversi dari rad/s menjadi RPM. Input dari *speed controller* adalah *set point* dan kecepatan rotor dari *particle filter*, di dalam *speed controller* ini merupakan PI *controller* konvensional yang dimana konfigurasi gain proporsional adalah 0,1 dan gain integral adalah 0,01. *Speed controller* akan mengeluarkan fluks dan torsi referensi yang nantinya akan menjadi input untuk DTC. Selain torsi dan fluks referensi, input DTC juga termasuk Vabc yang bisa didapat dari inverter 3 fasa. Fungsi Vabc ini dipakai untuk mengestimasi besar torsi dan fluks, sehingga akan didapat perbedaan antara torsi dan fluks refrensi dan hasil estimasi, kemudian dimasukkan ke blok *hysteresis control* dan digunakan untuk pemilihan *switching* yang tepat pada *switching table*. Output dari DTC

berupa pulsa yang digunakan untuk mengontrol tegangan yang masuk kedalam inverter 3 fasa.

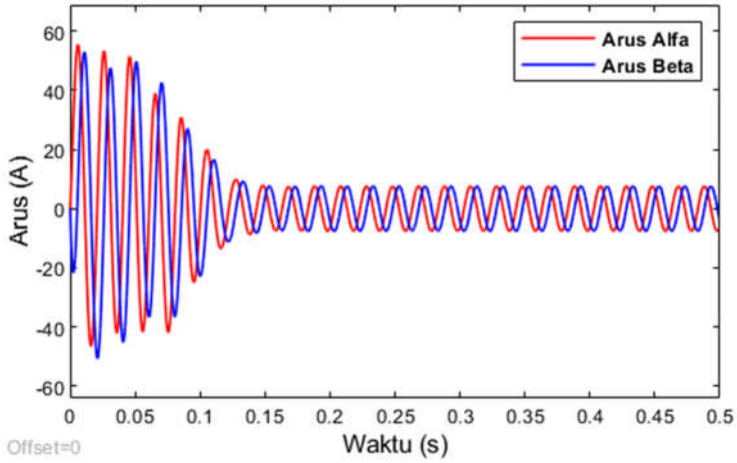
Output tegangan inverter 3 fasa akan diubah dari fasa a-b-c menjadi fasa αβ yang kemudian tegangan αβ ini akan dipakai sebagai input untuk motor induksi dan juga input untu blok *particle filter* untuk mengestimasi kecepatan rotor motor induksi.

Pengujian akan dilakukan pada kecepatan 50, 100, 150, 350 dan 500 RPM. Untuk aplikasi dari motor induksi, pada kecepatan rendah seperti 100 RPM digunakan dalam mesin *conveyor*, dalam hal ini motor induksi dapat digunakan pada rentang kecepatan yang diinginkan seperti seberapa cepat mesin *conveyor*-nya bekerja. Kecepatan motor induksi yang digunakan untuk mesin *conveyor* ini berada pada rentang kecepatan 10 RPM hingga 300 RPM. Aplikasi motor induksi yang lainnya digunakan dalam mesin kompresor, elevator, pompa, mesin penggiling, transportasi kerta api, dsb.

BAB IV
HASIL DAN PEMBAHASAN

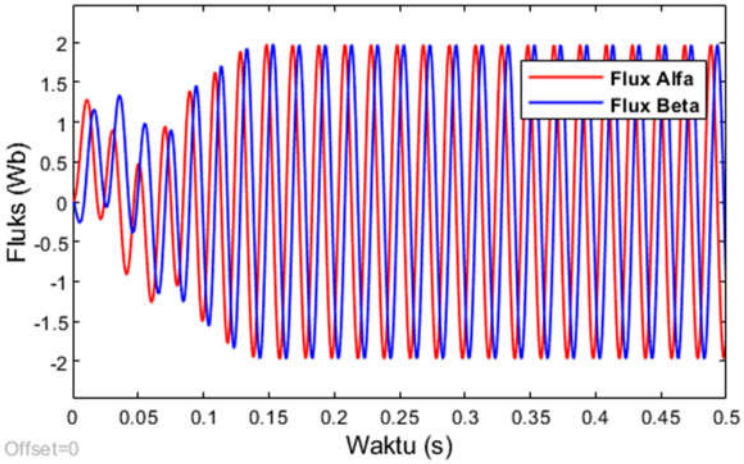
4.1 Hasil Pemodelan Motor Induksi

Pemodelan motor induksi dilakukan pada *software* MATLAB 2019b. Pemodelan motor induksi menggunakan parameter-parameter yang terdapat pada Tabel 3.1. Berikut grafik dari arus, fluks, dan kecepatan dari model motor induksi yang telah dibuat.



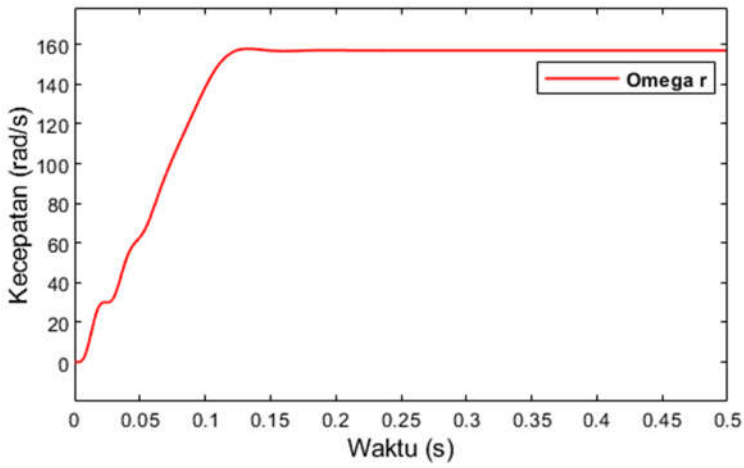
Gambar 4. 1 Respon Arus Motor Induksi

Pada Gambar 4.1 diatas dapat dilihat respon arus dari motor induksi, arusnya dibagai menjadi dua jenis yaitu arus α ($i_{\alpha s}$) dan arus β ($i_{\beta s}$). Kedua arus ini sesuai dengan pemodelan dari motor induksi dan merupakan *state* pertama dan kedua dari pemodelan motor induksi. Dapat dilihat dari Gambar 4.1 awalnya arus mengalami keadaan *transient* sebelum akhirnya mencapai keadaan *steady state*. Keadaan *transient* ini berlangsung dari 0 detik hinga 0,15 detik, hingga akhirnya setelah 0,15 detik keadaan sudah menjadi *steady*. Nilai awal untuk arus α ($i_{\alpha s}$) adalah sebesar 55,38A dan untuk arus β ($i_{\beta s}$) adalah 53,19 A, kemudian untuk kedua arus tersebut menurun dan menuju keadaan *steady state* dengan besar arus 7,67A



Gambar 4. 2 Respon Fluks Motor Induksi

Kemudaian *state* ketiga dan keempat dari pemodelan motor induksi adalah yaitu fluks α ($\lambda_{\alpha s}$) dan arus β ($\lambda_{\beta s}$). Sama dengan respon arus sebelumnya pada Gambar 4.2 dapat diilihat bahwa fluks juga mengalami keadaan *transient* yaitu dari 0 detik hingga 0,15 detik hingga akhirnya menuju kedaan *steady*. Gambar 4.1 dan Gambar 4.2 menunjukkan bahwa respon arus dan respon fluks mengalami keadaan *transient* dengan rentang waktu yang sama yaitu 0,15 detik dengan nilai tertinggi daru kedua fluks adalah sebesar 1,68 Wb, dari kesamaan ini dapat dikatakan bahwa arus dan fluks saling berkaitan satu sama lain.



Gambar 4. 3 Respon Kecepatan Motor Induksi

State terkahir yaitu *state* kelima dari pemodelan motor induksi adalah kecepatan. Pada Gambar 4.3 dapat dilihat bahwa kecepatan motor induksi berada dalam keadaan *transient* selama 0,20 detik, rentang yang sama dengan arus dan

fluks, dengan nilai kecepatan rotor motor induksi adalah sebesar 157,1 rad/s. Hal ini menunjukkan bahwa kelima *state* dari pemodelan motor induksi ini saling berkaitan satu dengan yang lainnya.

Untuk validasi model motor induksi dilakukan perbandingan besar kecepatan motor induksi (ω_r) berdasarkan penelitian sebelumnya (Ardini, 2019) yang membuat model motor induksi yang terdapat pada PT. INKA. Hasil yang didapat adalah sebagai berikut.

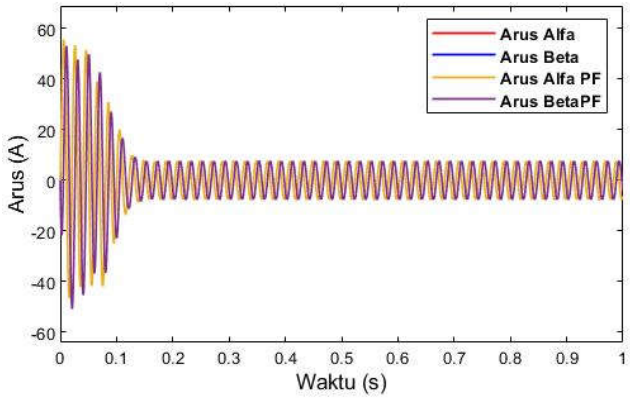
Tabel 4. 1 Data kecepatan model motor dan pengukuran

No.	Tegangan (V)	ω_r *Plant* (rad/s)	ω_r Pengukuran (rad/s)	*Error* (%)
1	263	157,1	156,9	0,127
2	296	157,1	157,07	0,019
3	378	157,1	157,3	0,127
Rata-rata				0,09

Berdasarkan hasil yang didapat dari tabel 4.1 diatas nilai rata-rata *error*-nya adalah 0.09%, yang dimana nilai *error* pemodelan yang dapat ini sudah kurang dari 5% yang dimana merupakan batas atas dari *error* pemodelan dari motor induksi, sehingga dapat dilanjutkan pada perancangan *particle filter*.

4.2 Pengujian *Particle Filter*

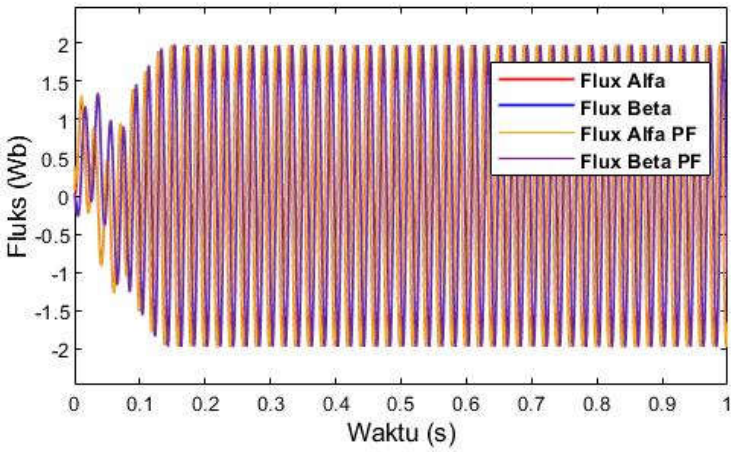
Algoritma *particle filter* yang sebelumnya telah dirancang sesuai penjelasan pada bab tiga. Dengan menggunakan *input* arus dan tegangan maka akan didapat kelima *state* dari model motor induksi. Pengujian *particle filter* ini dilakukan pada keadaan sistem *open loop* dan jumlah partikel yang digunkan pada pengujian ini sebanyak 250 partikel.



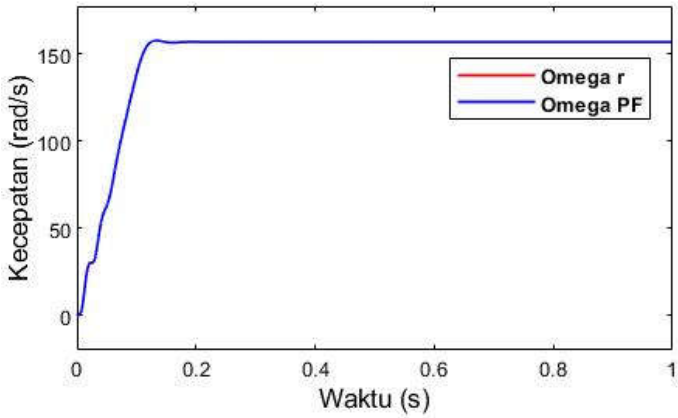
Gambar 4. 4 Respon Arus Motor Induksi dan Estimasi PF

Gambar 4.4 menunjukkan perbandingan respon *state* pertama dan kedua yaitu arus α ($i_{\alpha s}$) dan arus β ($i_{\beta s}$) *plant* motor induksi dengan hasil estimasi dari *particle filter*, dapat dilihat bahwa grafik dari hasil estimasi *particle filter* berhimpitan dengan grafik respon motor induksi, hasil respon keduanya mencapai keadaan transien pada detik yang sama yaitu detik ke 1,5 dengan besar arus *steady state* yang sama dengan hasil sebelumnya yaitu 7,67A, hal ini menunjukkan bahwa *state* arus dari motor induksi sudah dapat di-*track* oleh *particle filter*.

Kemudian untuk respon dari *state* ketiga dan keempat yaitu fluks α ($\lambda_{\alpha s}$) dan arus β ($\lambda_{\beta s}$) ditunjukkan pada Gambar 4.5 dibawah. Hasil yang didapat adalah nilai fluks dari *particle filter* dan motor induksi hampir sama yaitu 1,96 Wb, dengan keadaan *steady state* dicapai pada detik ke 1,5.



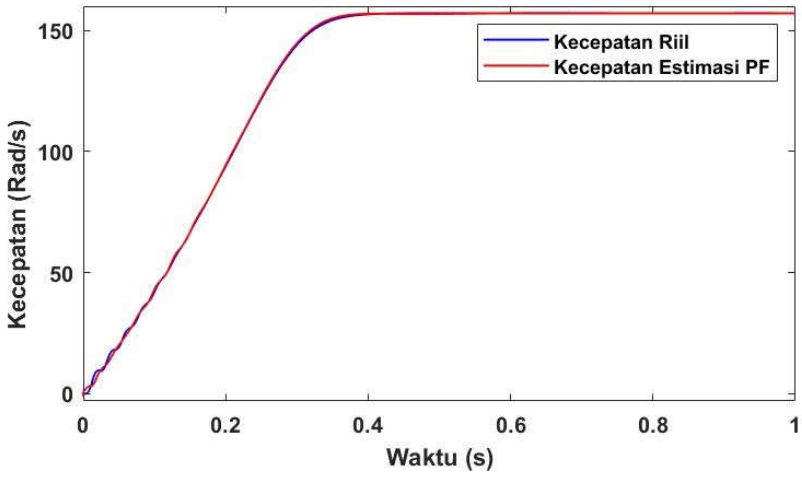
Gambar 4. 5 Respon Fluks Motor Induksi dan Estimasi PF



Gambar 4. 6 Respon Kecepatan Motor Induksi dan Estimasi PF

Pada Gambar 4.5 dan Gambar 4.6 dapat dilihat perbandingan dari fluks dan kecepatan motor induksi dengan hasil estimasi dari *particle filter*. Besarnya kecepatan rotor yang dihasilkan oleh plant dan *particle filter* sama yaitu 157,1 rad/s. Sama dengan grafik perbandingan respon arus yang sebelumnya grafik yang dihasilkan baik pada respon fluks dan respon kecepatan berhimpitan dan juga memiliki besar yang sama dengan hasil estimasi *particle filter*. Hal ini menunjukkan bahwa algoritma *particle filter* yang telah dirancang sudah bisa mengikuti kelima *state* dari motor induksi dan sudah layak untuk dijadikan estimator dan lanjut ke tahap berikutnya yaitu tahap sistem *close loop* dengan menggunakan DTC.

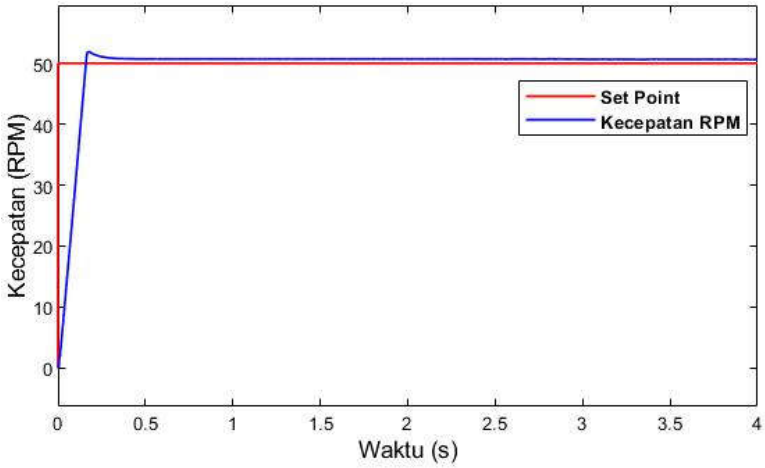
Pada Gambar 4.18 dalam (Firdausi, 2020) dapat dilihat perbandingan hasil dari respon kecepatan motor indkuksi dengan estimasi *particle filter*. Jumlah partikel yang dipakai adalah 250 partikel, dan hasilnya dapat dibandingkan dengan respon kecepatan yang ada pada Gambar 4.6. Hasil yang didapat dari jurnal validasi bahwa kecepatan motor berada pada nilai 157,07 rad/s dan hasil estimasi 157,08 rad/s, hasil yang hampir mendekati dengan yang didapat pada Gambar 4.6 dimana kecepatan dan hasil estimasi bernilai 157,1 rad/s. Hal ini dapat dikatakan bahwa algortima *particle filter* yang digunakan pada penelitian tugas akhir ini sudah berjalan dengan baik



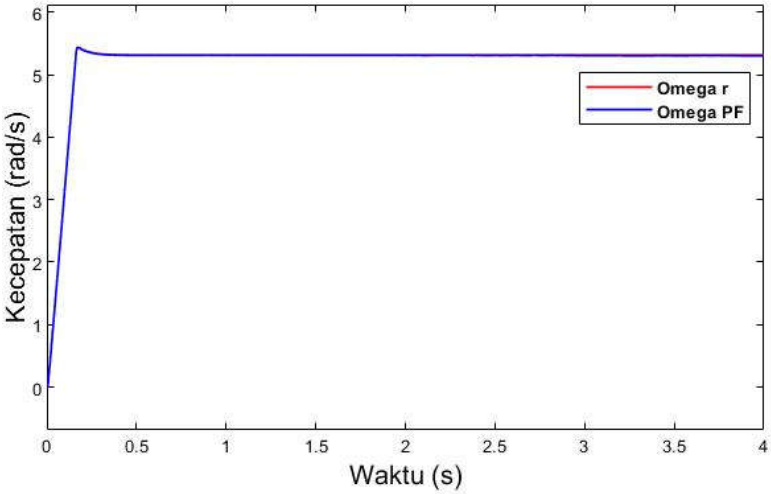
Gambar 4. 7 Respon kecepatan motor induksi (Firdausi, 2020)

4.3 Pengujian *Speed Sensorless Control*

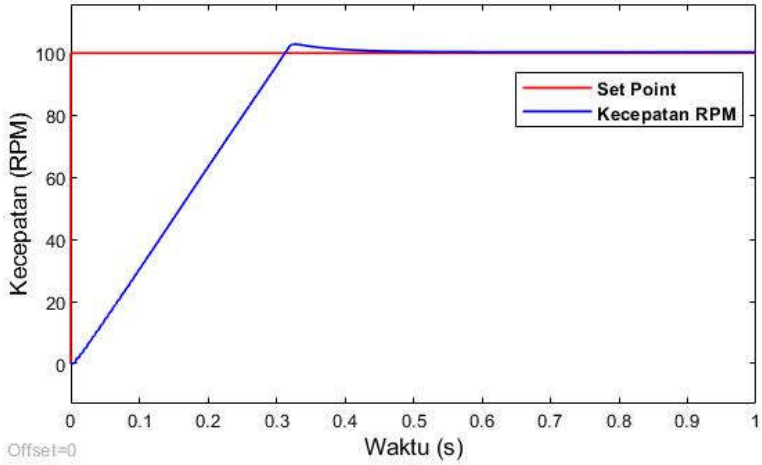
Pengujian sistem kontrol ini dilakukan dengan variasi *set point* dari 50 rpm hingga 500 rpm. Secara terpenrinci besar *set point* yang akan digunakan adalah 50 rpm, 100 rpm, 350 rpm dan 550 rpm. Jumlah partikel yang digunakan *particle filter* pada pengujian ini adalah 500 partikel dengan parameter *particle filter* yang sama dengan Tabel 3.2. Hasil respon dapat dilihat pada Gambar 4.7 hingga Gambar 4.16.



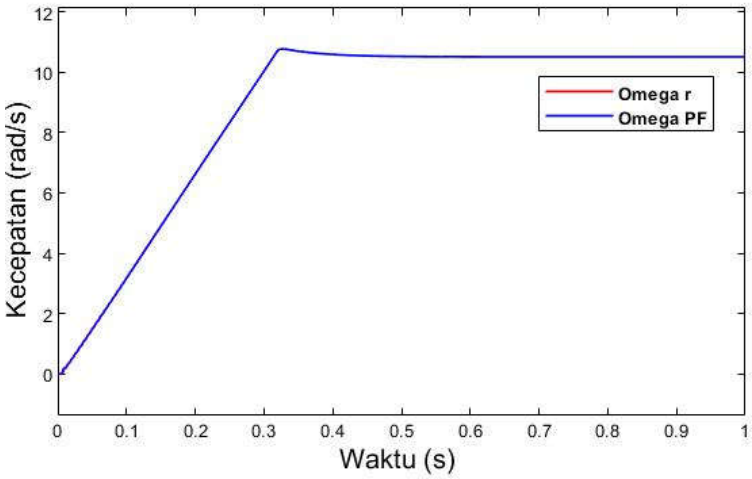
Gambar 4. 8 Respon Kecepatan Set Point 50 dengan *Particle Filter*



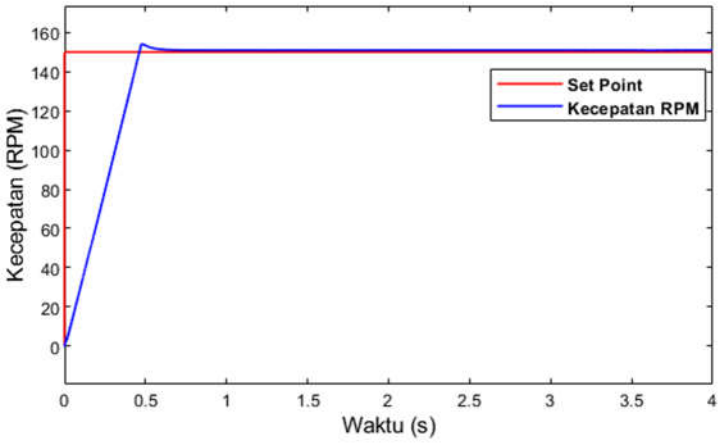
Gambar 4. 9 Perbandingan Respon Kecepatan *Plant* dan PF Set Point 50



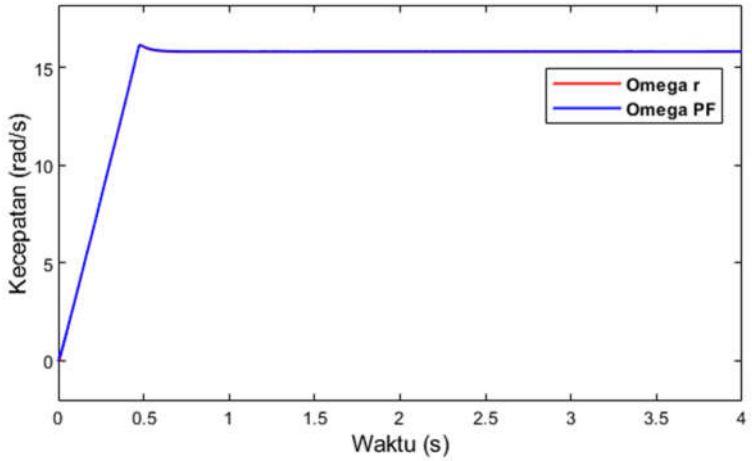
Gambar 4. 10 Respon Kecepatan Set Point 100 dengan Particle Filter



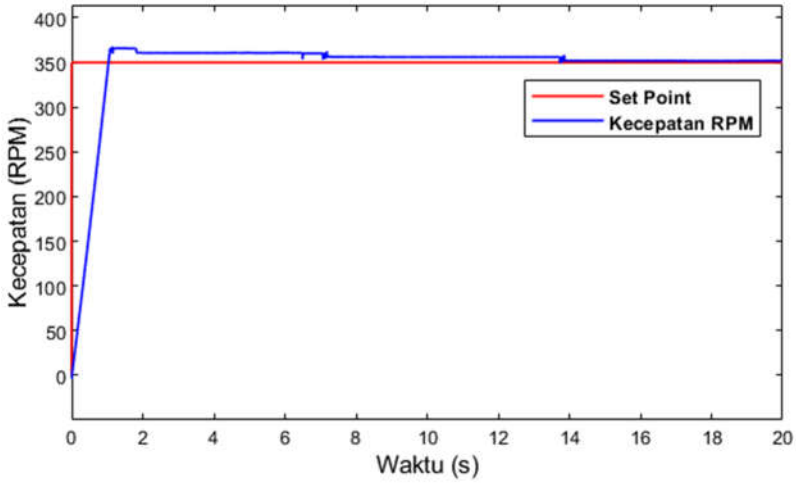
Gambar 4. 11 Perbandingan Respon Kecepatan Plant dan PF Set Point 100



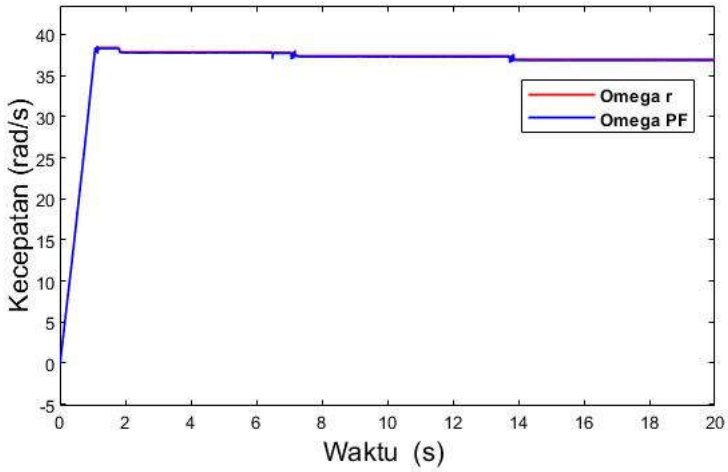
Gambar 4. 12 Respon Kecepatan Set Point 150 dengan *Particle Filter*



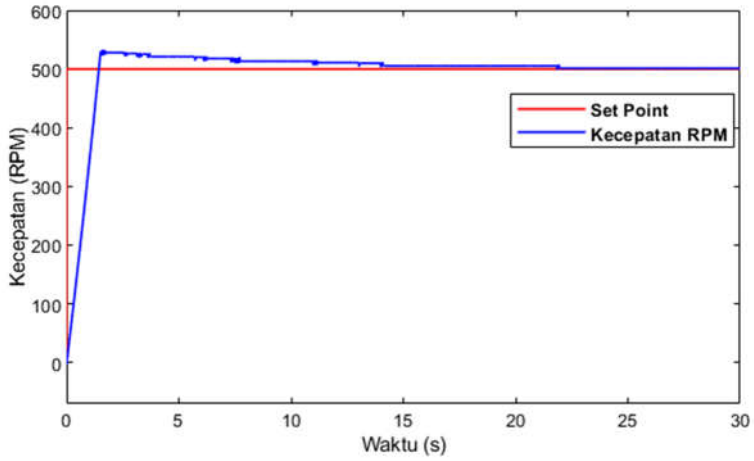
Gambar 4. 13 Perbandingan Respon Kecepatan Plant dan PF Set Point 150



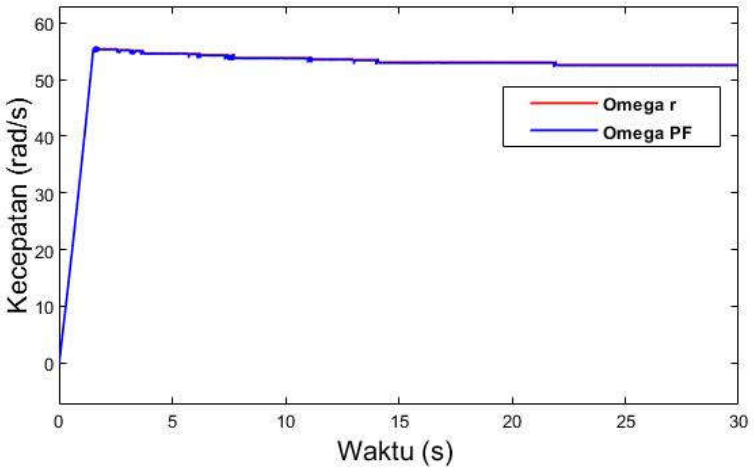
Gambar 4. 14 Respon Kecepatan Set Point 350 dengan Particle Filter



Gambar 4. 15 Perbandingan Respon Kecepatan Plant dan PF Set Point 350



Gambar 4. 16 Respon Kecepatan Set Point 500 dengan Particle Filter



Gambar 4. 17 Perbandingan Respon Kecepatan Plant dan PF Set Point 500

Dari grafik-grafik perbandingan respon *set point* dengan kecepatan hasil estimasi diatas maka bisa didapat nilai deari *error steady state*-nya seperti pada Tabel 4.1 berikut

Tabel 4. 2 Hasil uji performansi masing-masing kecepatan

Set Point (RPM)	Kecepatan PF (RPM)	*Error Steady State* (%)	*Maximum Overshoot* (%)	*Rise time (s)*
50	50,65	1,30	3,60	0,09
100	100,3	0,30	2,80	0,25
150	151	0,67	2,87	0,37
350	351,9	0,54	4,57	0,82
500	501,4	0,28\\	6,00	0,14

Pada Tabel 4.1 diatas dapat dilihat bahwa setiap set point memiliki nilai *error steady state* yang sangat kecil dan sudah memenuhi kriteria besar *error* yang diinginkan yaitu lebih kecil dari 5%. Namun apabila dilihat pada Gambar 4.13 dan Gambar 4.15 *settling time* yang dihasilkan sangat besar begitu juga dengan *maximum overshoot*, hal ini diakibatkan jenis PI *controller* yang digunakan pada *speed controller* adalah jenis yang konvensional sehingga menimbulkan fenomena *windup*. Berdasarkan (Choi & Lee, 2009) Penurunan kinerja ini mengacu pada integrator *windup* yang ada pada PI *controller*, yang dapat secara signifikan menurunkan kinerja kontrol meghasilkan *onvershoot* yang tinggi dan *settling time* yang lama. *Windup* terjadi dikarenakan setiap *controller* bekerja pada kisaran terbatas (*limited range*). Sehingga dengan batas tertentu tersebut *plant* sesungguhnya dapat berbeda dengan output *controller.* Ketika ini terjadi ketika *controller* dirancang diantara rentang linier maka akan kinerja sistem *close-loop* akan menimbulkan penurunan secara signifikan. Oleh karena itu untuk mengatasi fenomena ini pada (Choi & Lee, 2009) dilakukan penelitian *Antiwindup* PI *controller* pada *speed controller*, dan hasil yang didapat adalah penurunan nilai overshoot pada respon kecepatan.

Kemudian berikut adalah data beban komputasi dengan menggunakan jumlah partikel yang berbeda dengan waktu besar kecepatan dan waktu simulasi yang sama yaitu 100 RPM dan 1 detik. Variasi jumlah partikelnya yaitu 50, 100, 250 dan 500 partikel, komputasi dilakukan menggunakan perangkat yang sama dengan spesifikasi *processor* Intel Core i5-9300H @2.4 GHz (8CPUs). Beban komputasi akan direpresentasikan lamanya waktu yang dibutuhkan selama *running* simulasi, dapat dilihat pada Tabel 4.3 berikut

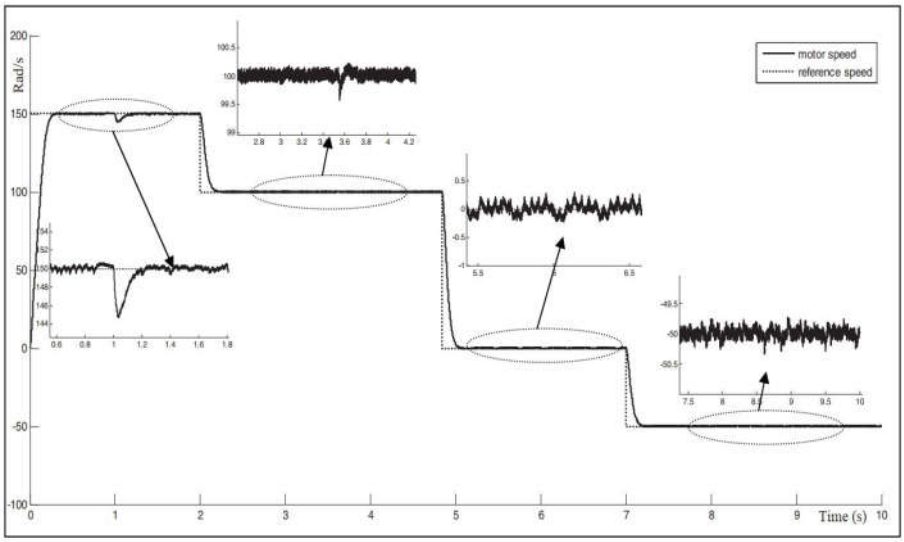
Tabel 4. 3 Beban Komputasi dengan Jumlah Partikel Berbeda

Jumlah Partikel	Waktu *Running*
50	2 Menit 9 Detik
100	2 Menit 31 Detik
250	3 Menit 52 Detik
500	6 Menit 9 Detik

Pada Tabel 4.3 diatas dapat dilihat bahwa semakin banyak jumlah partikel yang dipakai akan memiliki waktu *running* yang semakin lama, hal ini juga dapat diartikan beban komputasi yang dimiliki juga semakin tinggi.

4.4 Hasil Perbandingan Desain *Sensorless* DTC Secara Simulasi dan DTC

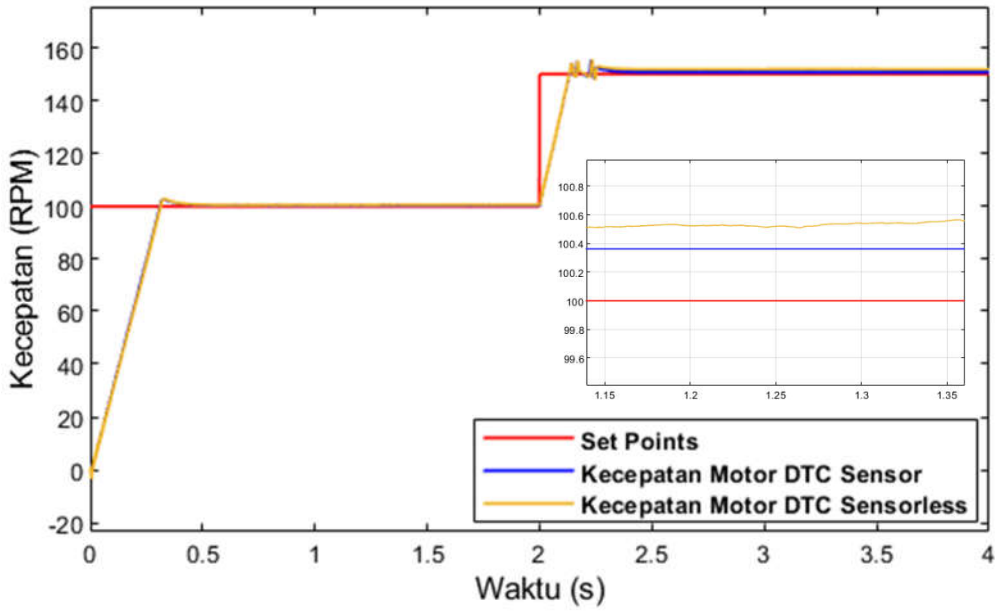
Berikut adalah validasi dari hasil model yang didaptakan penggunaan skema kontrol DTC. Dalam jurnal oleh (Mohammedi, Kabache, Moulahoum, & Houassine, 2015) sudah dilakukan *FPGA hardware* untuk skema kontrol DTC, pengujian dilakukan pada kecepatan, 150 RPM, 100RPM, 0 RPM, dan -50 RPM. Namun untuk validasi dengan hasil yang didapat pada tugas akhir ini akan diambil pada kecepatan 100 RPM dan 150 RPM, hasil dari *FPGA hardware* dari skema kontrol DTC dapat dilihat pada Gambar 4.17, perlu diketahui bahwa pada jurnal validasi ini tidak menggunakan mode *speed sensorless*, tetapi berfokus pada performansi dari skema kontrol DTC.



Gambar 4. 18 Respon kecepatan motor dengan *FPGA hardware* untuk motor induks (Mohammedi, Kabache, Moulahoum, & Houassine, 2015)

Pada Gambar 4.17 dapat dilihat bahwa DTC bekerja dengan baik disetiap *set point* yang diberikan, terutama untuk *set point* 100 RPM dan 150 RPM yang menghasilkan respon seperti *rise time* yang sama sebesar 0.2 detik dan tidak memiliki *maximum overshoot* namun secara keseluruhan bentuk grafik yang didapat hampir sama dengan Gambar 4.10 dan Gambar 4.12, dengan ini dapat dikatakan bahwa skema DTC yang dipakai pada tugas akhir ini, berjalan dengan baik. Meskipun pada dalam jurnal validasi tidak menggunakan mode *speed senorlees* namun pada bagian kesimpulan dikatakan dalam (Mohammedi, Kabache, Moulahoum, & Houassine, 2015) bahwa dengan hasil yang didapat, DTC dapat diimplementasikan lebih jauh untuk mode *speed senorless* untuk motor induksi.

Secara simulasi perbandingan antara DTC dengan menggunakan sensor dan *sensorless* dapat dilihat pada Gambar 4.19 berikut



Gambar 4. 19 Perbandingan DTC Sensor dengan *sensorless*

Pada Gambar 4.19 diatas merupakan hasil uji dari DTC dengan dan tanpa sensor, pada simulasi DTC menggunakan sensor, nilai kecepatan langsung diambil dari *plant* motor induksi. Hasil simulasi yang didapat antara DTC sensor dan *sensorless* jika dilihat dari grafik secara langsung memiliki bentuk yang sama. Perbandingan data *maximum overshoot* pada kecepatan 100 RPM dari DTC dengan sensor dan *sensorless* adalah 2,7% dan 2,8%, kemudian untuk data *settling time* adalah 0,522 dan 0,525. Kemudian perbandingan data *maximum overshoot* pada kecepatan 150 RPM dari DTC dengan sensor dan *sensorless* adalah 2,67% dan 2.73%, selanjutnya data *settling time*-nya adalah 0,448 dan 0,378.

Hasil yang didapat pada kedua mode DTC cukup berdekatan hal ini dapat diartikan DTC dengan mode *sensorless* dapat mendekati hasil dari DTC dengan menggunakan sensor, namun pada DTC dengan mode *sensorless* memiliki keunggulan *set* up lebih sederhana karena penghilangan sensor, oleh karena itu DTC dengan *sensorless* bisa dikatakan lebih baik dibandingkan DTC dengan menggunakan sensor.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengujian dan analisa data yang telah dilakukan maka dapat diambil kesimpulan sebagai berikut:

a) Rancangan *speed sensorless control* dengan menggunakan skema *Direct Torque Control* diuji pada rentang 50-500 RPM dan menghasilkan nilai *error steady state* sesuai dengan yang diharapkan yaitu dibawah 5%, maka dapat dikatakan bahwa rancangan sudah bekerja dengan baik. Walaupun memiliki nilai *overshoot* yang tinggi dan rentang *settling time* yang lama pada kecepatan 350 dan 500 RPM.

b) Hasil perbandingna skema kontrol DTC dengan mode *senosrless* dan DTC dengan memakai sensor memiliki hasil yang hampir sama terutama pada kecepatan 100 dan 150 RPM, perbedaanya hanya terletak pada nilai *overshoot* pada skema *sensorless* yaitu sebesar 2,8% untuk 100 RPM dan 2,9% untuk 150RPM serta nilai *rise time* sebesar 0,25 dan 0,37 detik. Hasil simulasi performa DTC dengan mode *sensorless* menunjukkan hasil yang cukup mirip dengan DTC yang menggunakan sensor, namun kelebihan DTC dengan *sensorless* terletak pada *set up* yang lebih sederhana karena penghilangan sensor, oleh karena itu DTC *sensorless* dapat dikatakan lebih baik.

5.2 Saran

Saran yang dapat diberikan adalah diharapkan untuk dapat diimplementasikan kedalam bentuk *hardware* dan jenis PI *controller* yang digunakan sudah dalam bentuk *anti-windup* PI *controller* sehingga didapatkan hasil yang lebih baik.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

Abu-Rub, H., Iqbal, A., & Guzinski, J. (2012). *High Performance Control of AC Drives With MATLAB/Simulink Models.* West Sussex, UK: John Wiley & Sons.

Arulampalam, M. S., Maskell, S., Gordon, N., & Clapp, T. (2002). A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 175-188.

Arya, M., & Wadhani, S. (2011). Development of DIrect Torque Control Model With Using SVI for Three Phase Induction Motor. *International Journal of Engineering Science and Technology (IJEST)*, 6650-6655.

Barut, M., Bogosyan, S., & Gokasan, M. (2007). Speed-Sensorless Estimation for Induction Motors Using Extended Kalman Filters. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*, 272-280.

Buja, G., Casadei, D., & Serra, D. (1997). Direct Torque Control of Induction Motor Drives. *International Symposium on Industrial Electronics* (hal. -). Guimaraes: IEEE.

Chan, T.-F., & Shi, K. (2011). *Applied Intelligent Control Of Induction Motor Drives.* Singapore: John Wiley & Sons.

Choi, J.-W., & Lee, S.-C. (2009). Antiwindup Strategy for PI-Type Speed Controller. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*, 2039-2046.

Firdausi, D. C. (2020). *Perancangan Particle Filter untuk Speed Sensorless Induction Motor di PT. INKA.* Surabaya: ITS.

Gordon, N., Salmond, D., & Smith, A. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings on Radar and Signal Processing*, 107-113.

Guedes, J. J., & al., e. (2018). Parameters estimation of three-phase induction motors using differential evolution. *Electric Power Systems Research*, 204-212.

Gyorgy, K., Kelemen, A., & David, L. (2014). Unscented Kalman filters and Particle Filter methods for nonlinear state estimation. *The 7th International Conference Interdisciplinarity in Engineering* (hal. 65-74). Tirgu Mures, Romania: -.

Kitagawa, G. (1996). Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models. *Journal of Computational and Graphical Statistics*, 1-25.

Mohammedi, A., Kabache, N., Moulahoum, S., & Houassine, H. (2015). FPGA Hardware In the Loop Validation of Direct Torque Control for Induction Motor. *20th International Conference on Methods and Models in Automation and Robotics (MMAR)*, 812-816.

Pereira, L., Perin, M., Pereira, L., Ruthes, J., de Sousa, F., & de Oliveira, E. (2019). Performance estimation of three-phase induction motors from no-load startup test without speed acquisition. *ISA Transactions*, 376-389.

Schon, T. B. (2014). NONLINEAR SYSTEM IDENTIFICATION USING PARTICLE FILTERS. *Encyclopedia of Systems and Control*, -.

Shinnaka, S. (2002). Usefulness of a Frequency-Hybrid Vector Control for a Sensorless Induction Motor Drive. *Electrical Engineering in Japan*, 1143-1154.

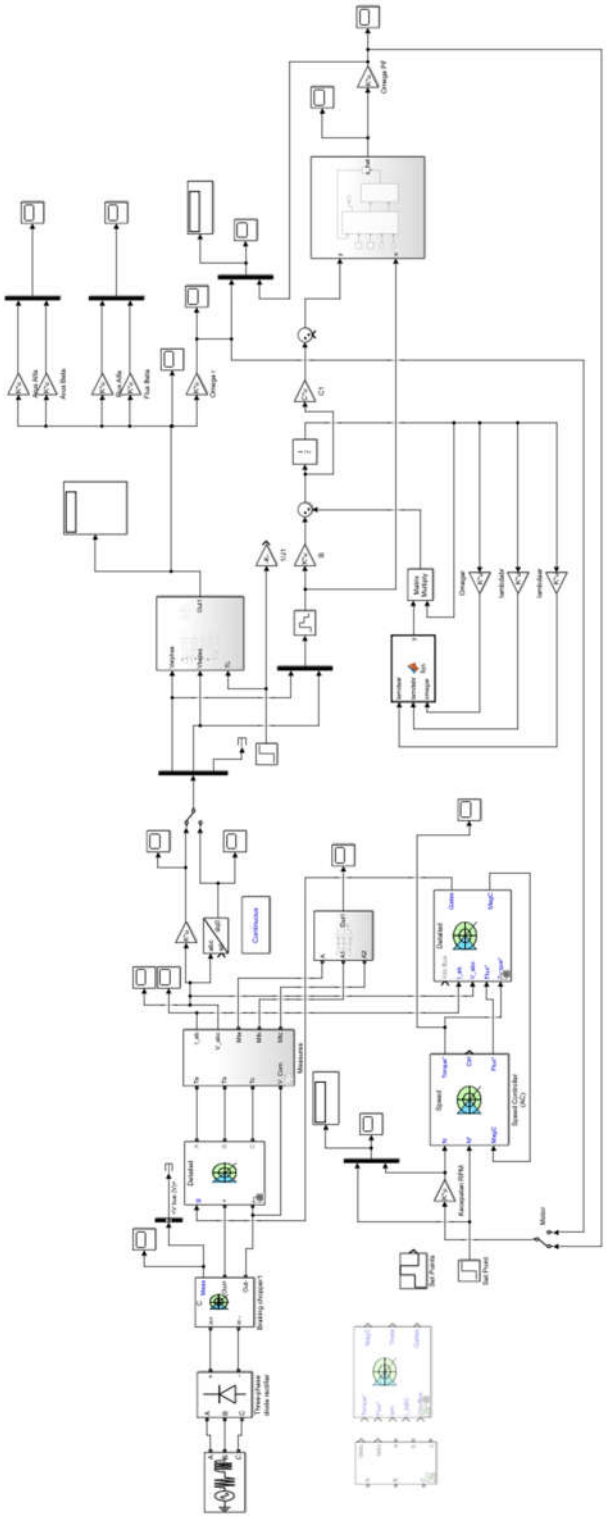
Thomson, W. T., & Culbert, I. (2017). *CURRENT SIGNATURE ANALYSIS FOR CONDITION MONITORING OF CAGE INDUCTION MOTORS Industrial Application and Case Histories.* New Jersey: John Wiley & Sons.

Wang, C.-C., & Fang, C.-H. (2003). Sensorless Scalar-Controlled Induction Motor Drives With Modified Flux Observer. *IEEE TRANSACTIONS ON ENERGY CONVERSION*, -.

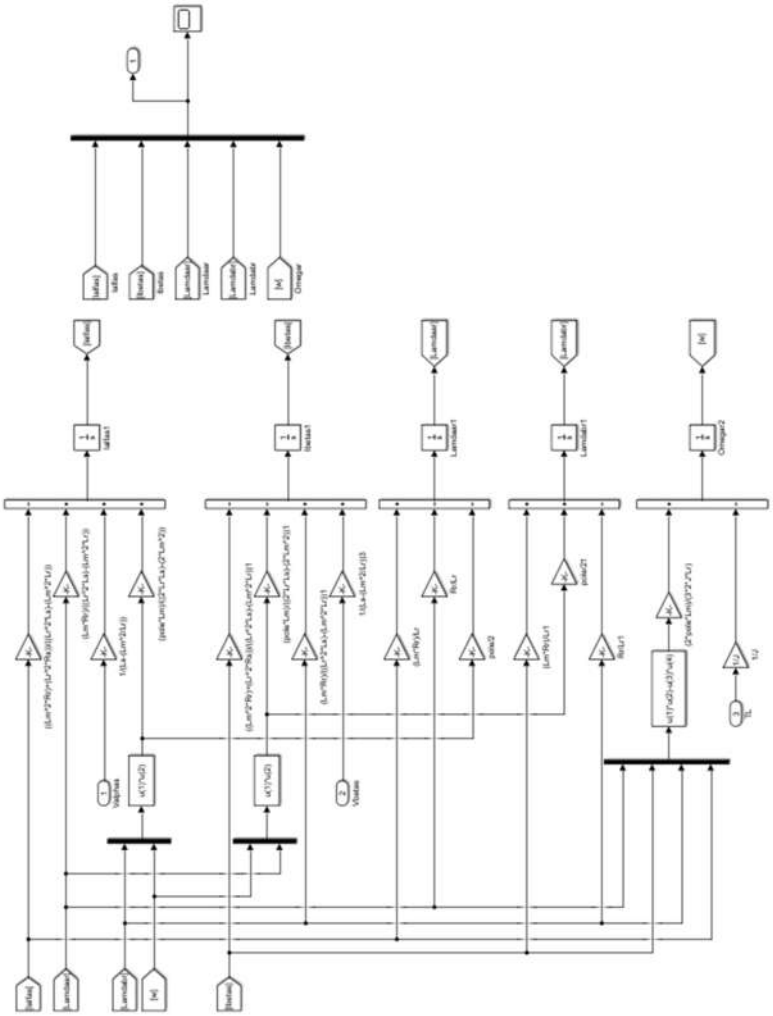
Zerikat, M., Mechernene, A., & Chekroun, S. (2010). High-performance sensorless vector control of induction motor drives using artificial intelligent technique. *EUROPEAN TRANSACTIONS ON ELECTRICAL POWER*, 787-800.

LAMPIRAN

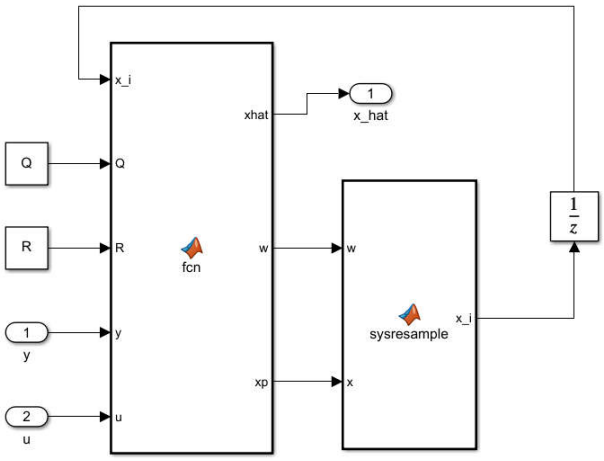
A. Plant Keseluruhan



B. Model Motor induksi



C. Model Particle Filter



D. M File Parameter Plant Motor Induksi

```
clear; clc;
Lm = 0.258;
Ls = 0.274;
Lr = 0.274;
Rr = 3.805;
Rs = 4.85;
%Tr = Lr/Rr;
J = 0.031;
Ts = 0.000001;
pole = 4;
%Tr = Lr/Rr;
tho = 1-(Lm^2/(Lr*Ls));

lamdaar=0;
lamdabr=0;
omegar=0;

a1 = (Lm^2*Rr+Lr^2*Rs)/(tho*Ls*Lr^2);
a2 = Lm*Rr/(tho*Ls*Lr^2);
a3 = pole*Lm/(2*tho*Ls*Lr);
a4 = Lm*Rr/Lr ;
a5 = Rr/Lr;
a6 = 2*pole*Lm/(2*3*J*Lr);

A = [-a1 0 a2 a3*omegar 0; 0 -a1 -a3*omegar a2 0;
a4 0 -a5 -pole*omegar/2 0; 0 a4 pole*omegar/2 -a5
0; -a6*lamdabr a6*lamdaar 0 0 0];
B = [1/(tho*Ls) 0;0 1/(tho*Ls);0 0;0 0;0 0];
C = [1 0 0 0 0; 0 1 0 0 0];
D = zeros(2);

Q  = 1e-11;
R  = [0.05^2 0; 0 0.05^2];

N  = 500;

x0 = [0 0 0 0 0]';
P0 = 0.1*eye(5);
nx = size(P0,2);
x_u = repmat(x0,1,N)+sqrt(P0)*randn(nx,N);
```

E. M File Particle Filter (*Update Particle*)

```
function [xhat, w, xp] = fcn(x_i, Q, R, y, u)
N  = 500;
n = size (x_i,1);
xp = zeros(n,N);

Lm = 0.258;
Ls = 0.274;
Lr = 0.274;
Rr = 3.805;
Rs = 4.85;
J = 0.031;

Ts = 0.000001;
pole = 4;
tho = 1-(Lm^2/(Lr*Ls));

a1 = (Lm^2*Rr+Lr^2*Rs)/(tho*Ls*Lr^2);
a2 = Lm*Rr/(tho*Ls*Lr^2);
a3 = pole*Lm/(2*tho*Ls*Lr);
a4 = Lm*Rr/Lr ;
a5 = Rr/Lr;
a6 = 2*pole*Lm/(2*3*J*Lr);

for t=1:N

    lamdaar =[0 0 1 0 0]*x_i(:,t);
    lamdabr =[0 0 0 1 0]*x_i(:,t);
    omegar =[0 0 0 0 1]*x_i(:,t);
    Ab = eye(5)+ Ts*[-a1 0 a2 a3*omegar 0; 0 -a1 -
a3*omegar a2 0; a4 0 -a5 -pole*omegar/2 0; 0 a4
pole*omegar/2 -a5 0; -a6*lamdabr a6*lamdaar 0 0
0];
    Bb = Ts*[1/(tho*Ls) 0;0 1/(tho*Ls);0 0;0 0;0
0];
    C = [1 0 0 0 0; 0 1 0 0 0];
    xp(:,t) = Ab*x_i(:,t)+Bb*u+sqrt(Q)*randn(n,1);
end
    e = repmat(y,1,N) - C*xp;
    w = exp(-(1/2)*(sum(e.*(R\e))));
    w = w/sum(w);
    nx = size(Ab,2);

xhat = real(sum (repmat(w,nx,1).*xp,2));
```

F. M File Particle Filter (*Systematic Resampling*)

```
function x_i = sysresample(w,x)
qc = cumsum(w);      M=length(w);
u = ((0:M-1)+rand(1))/M;
i = zeros(1,M);      k=1;
for j=1:M
    while (qc(k)<u(j))
        k=k+1;
    end
    i(j)= k;
end
x_i = real(x(:,i));
```

G. Kode Matlab

Halaman ini sengaja dikosongkan

BIODATA PENULIS

Nama penulis Egy Josua Simbolon. Penulis dilahirkan di Hariaratolu, pada tanggal 9 Februari 1999 sebagai anak pertama dari empat bersaudara. Saat ini penulis tinggal di Samosir, Sumatera Utara. Penulis telah menyelesaikan Pendidikan tingkat dasar di SD. Sw. Santo Michael Pangururan hingga tahun 2010, pendidikan menengah pertama di SMP Sw. Budi Mulia Pangururan hingga tahun 2013, pendidikan menengah atas di SMA Unggul Del Laguboti hingga menempuh pendidikan S1 Teknik Fisika FTIRS di Institut Teknologi Sepuluh Nopember Surabaya hingga saat ini.

Pada bulan Juni 2020, penulis telah menyelesaikan tugas akhir dengan judul **Perancangan Estimator Kecepatan Berbasis *Particle Filter* untuk *Speed Sensorless Control* pada Motor Induksi**. Bagi pembaca yang memiliki kritik serta saran atau ingin berdiskusi lebih lanjut mengenai tugas akhir ini maka dapat menghubungi penulis melalui *email*: egysimbolon@gmail.com.