

## PERANCANGAN DAN IMPLEMENTASI *CONTROLLER KNOWLEDGE BASE* PADA PENGENDALIAN KECEPATAN MOTOR INDUKSI TIGA FASA

Almasri\*) dan Agus Santoso \*\*)

### ABSTRACT

*Designing a controller, the model development is necessary. From the model will shown the relationship between plant response and models response expected. This research will try to make plant of controller which has relationship between plant response and the models response. From the plant response and models response will be made the rule base as the basic plant of knowledge based controller. Before making the rule base, the plant parameters must be identified. The plant of knowledge based will be implemented for controlling three phase motor induction directly. From the relationship between plant response and the expected model. It's easier to use knowledge based controller to mean some models we expected. The aim of this research is to obtain algorithms of knowledge base controller non-linear base plant and model response expected, so that is obtained realtion really between the second response, which can controlled lenear and non-lenear plant and the stability of speed of the three phases induction motor is constant although change of load.*

**Key Word :** knowledge based controller, rule base, three phase motor induction

\*) Jurusan Teknik Elektronika FT Universitas Negeri Padang, e-mail: [al\\_unp@yahoo.co.id](mailto:al_unp@yahoo.co.id)

\*\*) Universitas Muhammadiyah Lampung, e-mail: [agus\\_uml@yahoo.co.id](mailto:agus_uml@yahoo.co.id)

### PENDAHULUAN

Pemakaian motor induksi tiga fasa sebagai tenaga penggerak lebih banyak digunakan dan diminati di industri. Hal ini disebabkan karena motor ini memiliki beberapa kelebihan dibandingkan motor arus searah seperti: konstruksi dan fisiknya yang tertutup sehingga relatif aman dari percikan air dan kotoran seperti debu, dimensi kecil dengan daya melalui medan putar tiga fasa yang memberi transfer energi yang lebih efisien tanpa menggunakan sikat dan komutator sehingga biaya pemeliharaan relatif murah.

Pemakaian tersebut umumnya digunakan untuk pengaturan yang membutuhkan kecepatan yang konstan sesuai dengan frekuensi sumber tegangan dan jumlah kutub. Dalam hal-hal tertentu motor induksi tiga fasa memerlukan pengaturan putaran yang presisi dan range yang luas serta respon dinamis yang baik, maka diperlukan suatu sistem pengaturan putaran.

Karena hubungan antara arus motor dan torsi yang dihasilkan tidak linier seperti pada motor arus searah, maka dibutuhkan suatu metode khusus untuk mengatur putaran motor induksi tersebut.

Pada peneliti terdahulu telah melakukan simulasi untuk pengaturan kecepatan motor induksi 3 fasa dengan menggunakan kontroler *knowledge base* (Lewi, 2001), Bertolak dari simulasi ini penulis mencoba mengimplementasikan kontroler *knowledge base* ini pada *plant* yang sebenarnya yaitu motor induksi tiga fasa.

Permasalahan yang akan dibahas dibatasi sebagai berikut :

1. Perancangan kontroler nonlinier berdasarkan pada perbandingan antara respon model yang diharapkan dengan *plant* yang akan dikontrol.
2. *Plant* yang digunakan motor induksi tiga fasa dengan beban berubah
3. Pengaturan tegangan input pada motor dilakukan melalui pengaturan tegangan inverter, dengan menganggap bahwa tegangan output inverter sebanding dengan sinyal kontrol yang dihasilkan oleh kontroler.
4. Kontroler menggunakan *knowledge base* dan berbasis *PC(personal computer)* dengan bahasa  $C^{++}$

Tujuan penelitian ini adalah untuk mendapatkan algoritma kontroler *knowledge base* nonlinier berdasarkan respon *plant* dan *model*

yang diharapkan, sehingga didapat hubungan yang jelas antara respon *plant* dan *model*, yang dapat mengontrol *plant* orde kesatu atau orde kedua, serta mempertahankan kecepatan putaran motor induksi tiga fasa dalam keadaan beban berubah-ubah.

### Kontroler Knowledge Base

*Knowledge base* merupakan informasi terorganisasi dan teranalisis agar bisa lebih mudah dimengerti dan bisa diterapkan pada pemecahan masalah dan pengambilan keputusan. *Knowledge base* berisi fakta-fakta tentang objek dalam *domain* yang ditentukan, pikiran, teori, prosedur praktis dan hubungannya satu sama lain. Pola yang diterapkan dalam *knowledge base* antara lain:

#### 1. Representasi Pengetahuan

Pola representasi pengetahuan menggambarkan aspek-aspek penting dari suatu masalah. Dari pola representasi yang dibuat, kemudian akan dirancang suatu *knowledge base* berdasarkan pola tersebut. Bahasa representasi haruslah mampu mengekspresikan *knowledge base* yang diperlukan untuk mendapatkan penyelesaian permasalahan. Objek relasi dan inferensi perhitungan yang tersedia dalam *knowledge base* ditentukan oleh representasi pengetahuan yang dipilih (Suparman, 1991).

#### 2. Representasi dengan Rule - based System

Merupakan representasi pengetahuan yang dinyatakan dengan IF-THEN. Alternatif perencanaan untuk rule *IF-THEN* adalah *statement premise-conclusion*. Kumpulan *premise* dapat dihubungkan dengan operator AND yang disebut *conjunctive-clause* atau dengan operator OR yang dinamakan *disjunctive-clause*.

Untuk mendapatkan pengetahuan, ada dua pendekatan teori yang biasa digunakan, yaitu: a. Mendapatkan pengetahuan langsung dari *domain expert*. b. Mendapatkan pengetahuan melalui induksi *rule*.

#### 3. Mekanisme Inferensi

Mekanisme *inferensi* adalah suatu cara untuk mengarahkan sistem pada kesimpulan, yang diperoleh dari aturan dan fakta yang ada. Dengan pendekatan tertentu, mekanisme inferensi ini melibatkan konversi dari representasi pengetahuan yang ada ke dalam bentuk pendekatan mekanisme inferensi yang dipilih (Suparman, 1991).

*Knowledge base* dianggap terdiri dari basis data dan aturan-aturan dasar (*rule base*), dan

representasi pengetahuan yang digunakan adalah representasi dengan sistem *rule-base*.

Kontroler mengolah sinyal error untuk menghasilkan suatu sinyal kontrol yang merepresentasikan tindakan yang harus diambil ketika harga variabel yang dikontrol mengalami penyimpangan dari set-point. Cara bagaimana kontroler menghasilkan sinyal kontrol dinamakan aksi kontrol atau mode kontrol (Lewi, 2001). *Knowledge Base System* (KBS) termasuk kelompok kendali cerdas yang merupakan bagian dari kecerdasan buatan (*Artificial Intelligence*). Pada dasarnya *knowledge base system* merupakan sebuah program komputer yang menggunakan pengetahuan dan teknik inferensi (pengambilan keputusan) untuk memecahkan persoalan seperti yang dilakukan oleh seorang pakar (Fakhreddine O, 2004). Kontrol ini menggunakan model atau proses berdasarkan pengalaman yang umumnya menghasilkan suatu model yang lebih menekankan kepada bagaimana proses kelakuan input/output berlangsung. Dengan keberhasilan kendali *knowledge base* yang diterapkan pada *plant* maka kendali cerdas ini dikembangkan lagi dengan menggabungkannya dengan kendali konvensional seperti kontroler *knowledge based PID* untuk mengendalikan motor DC sebagai penggerak spindel yang prinsipnya juga menjaga kestabilan torsi yang bekerja pada motor tersebut akibat beban yang berubah-ubah (Suryono, 2005),

### Motor Induksi Tiga Fasa

Motor induksi merupakan mesin listrik yang dapat mengkonversi energi listrik bolak-balik menjadi energi mekanik (putaran). Motor ini pada umumnya berputar pada kecepatan konstan, mendekati kecepatan sinkronnya. Pengaturan putaran motor induksi memerlukan biaya yang cukup tinggi. Biasanya pengaturan ini dapat dilakukan dengan beberapa cara antara lain (Zuhail, 1991):

#### 1. Mengubah jumlah kutub motor

$$n_s = \frac{120f}{p} \dots\dots\dots (1)$$

dimana:  $n_r$  = kecepatan putar rotor (rpm),  $f$  = frekuensi jala-jala (Hz),  $p$  = jumlah kutub

Jumlah kutub dapat diubah dengan merencanakan kumparan stator sedemikian rupa sehingga dapat menerima tegangan masuk pada posisi kumparan yang berbeda-beda.

#### 2. Mengubah frekuensi jala-jala

Pengaturan putaran motor induksi dapat dilakukan dengan mengubah-ubah harga frekuensi jala-jala. Hanya saja untuk menjaga kesetimbangan kerapatan fluks, perubahan tegangan harus dilakukan bersamaan dengan perubahan frekuensi. Cara ini sangat efisien dan daerah pengaturannya cukup lebar, meskipun alat pengaturannya lebih rumit dan mahal.

3. Mengatur tegangan jala-jala

$$T = \frac{3}{\omega} V_1^2 \frac{sa^2 R_2}{(a^2 R)^2 + s^2 (a^2 X_2)^2} \dots\dots\dots (2)$$

Bahwa torsi sebanding dengan pangkat dua tegangan yang diberikan. Cara ini menghasilkan daerah pengaturan yang sempit.

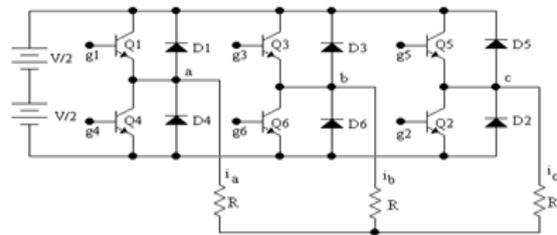
4. Pengaturan tahanan luar

Perubahan tahanan luar motor induksi rotor belitan akan menghasilkan pengaturan putaran. Cara ini mengakibatkan kerugian daya yang cukup besar pada rotor, besarnya:  $\frac{s}{1-s}$  kali daya mekanik. Untuk slip 0,5 kerugiannya sama dengan kerugian mekaniknya.

**Inverter Tiga Fasa**

*Inverter* dipakai untuk mengubah daya arus searah menjadi daya arus bolak balik yang tegangan dan frekuensinya dapat diatur. Tega-

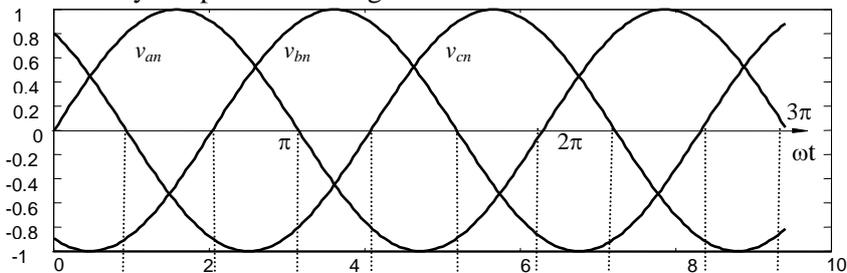
ngan bolak balik yang dihasilkannya berbentuk gelombang persegi dan untuk pemakaian tertentu diperlukan filter untuk menghasilkan bentuk gelombang sinus. Inverter 3 fasa paling sedikit mempunyai 6 buah transistor(Q) dan 6 buah dioda(D), ditata sedemikian rupa sehingga membentuk rangkaian jembatan. Gambar berikut ini(Ramshaw, 1993) secara berturut-turut merupakan Inverter jembatan tiga fasa, bentuk gelombang yang dihasilkan dan tahap kondukt *switching* masing-masing transistor.



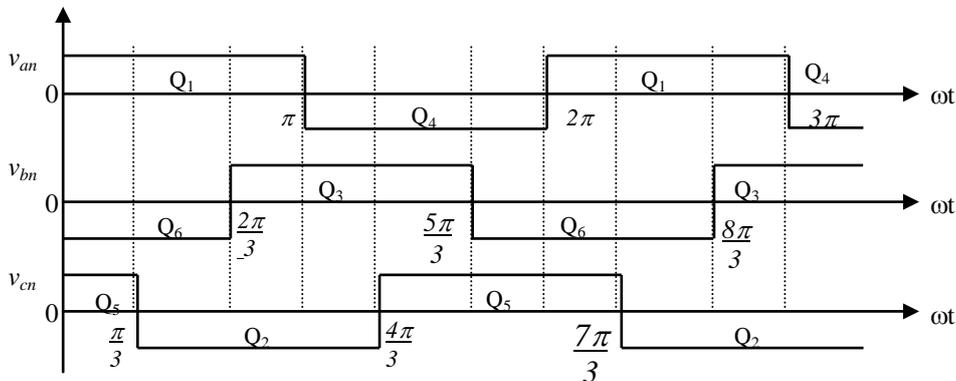
Gambar 1. Rangkaian Jembatan

Dari bentuk gelombang tegangan output pada gambar 3. di atas dapat dijelaskan urutan kerja tiap- tiap transistor (hanya dijelaskan satu perioda).

Tahap 0-  $1/3\pi$  rad, Q1, Q5, Q6 dalam keadaan ON dan Q2, Q3, Q4 dalam keadaan OFF.



Gambar 2. Gelombang Tiga Fasa



Gambar 3. Tahapan Kondukt

Tahap  $1/3\pi - 2/3\pi$  rad, Q1, Q2, Q6 dalam keadaan ON dan Q3, Q4 Q5 dalam keadaan OFF

Tahap  $2/3\pi - \pi$  rad, Q1, Q2, Q3 dalam keadaan ON dan Q4, Q5, Q6 dalam keadaan OFF

Tahap  $\pi - 4/3\pi$  rad, Q2, Q3, Q4 dalam keadaan ON dan Q1, Q5, Q6 dalam keadaan OFF

Tahap  $4/3\pi - 5/3\pi$  rad, Q3, Q4, Q5 dalam keadaan ON dan Q1, Q2 Q6 dalam keadaan OFF

Tahap  $5/3\pi - 2\pi$  rad, Q4, Q5, Q6 dalam keadaan ON dan Q1, Q2, Q3 dalam keadaan OFF

Penjelasan di atas memperlihatkan tahapan konduk switching masing- masing transistor yang ada pada gambar 1, dimana pergeseran urutan kerja transistor sebesar 600 atau  $1/3\pi$  rad secara bergantian.

**Spesifikasi Kontrol**

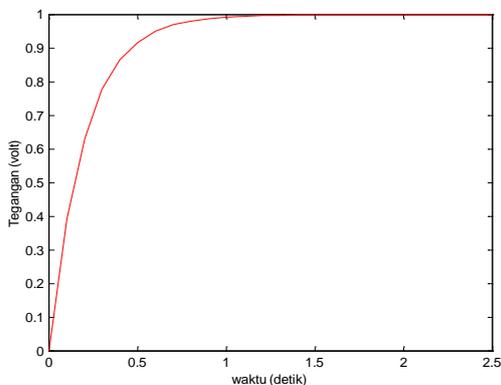
Setiap perancangan kontroler, diperlukan suatu spesifikasi kontrol yang akan digunakan sebagai acuan perancangan dan digunakan untuk melihat kualitas sistem melalui spesifikasi performansi. Berdasarkan respon sistem, biasanya dibagi dalam beberapa *orde*, yakni:

a. Sistem orde kesatu dengan fungsi transfer sebagai berikut :

$$\frac{Y(s)}{X(s)} = \frac{K}{\tau s + 1} \dots\dots\dots (3)$$

di mana :  $K = \text{gain total}$  dan  $\tau = \text{konstanta waktu}$ .

Respon sistem orde kesatu, merupakan respon yang teredam tanpa *overshoot* seperti gambar 4 berikut ini.



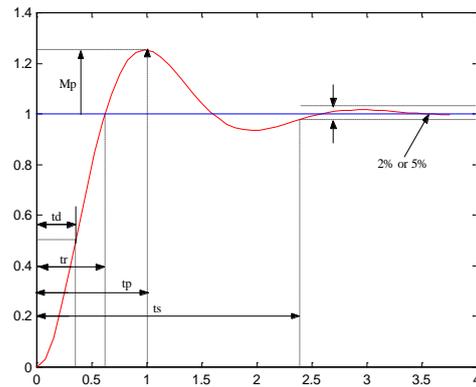
Gambar 4. Respon Orde Kesatu

b. Sistem orde kedua dengan fungsi transfer sebagai berikut :

$$\frac{Y(s)}{X(s)} = \frac{K\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \dots\dots\dots (4)$$

Gambar responnya dapat dilihat pada gambar 5.

Respon transien suatu sistem kontrol secara praktek selalu menunjukkan osilasi teredam sebelum mencapai *steady state* (Ogata, 1997). seperti gambar 5 berikut ini.



Gambar 5. Kurva Spesifikasi Respon Transien

Parameter-parameter yang diperlukan untuk menganalisa karakteristik performans sistem kontrol adalah sebagai berikut (Ogata, 1997):

- a. Waktu tunda (*delay time*),  $t_d$ : adalah waktu yang diperlukan oleh respon untuk mencapai setengah nilai akhir untuk waktu yang pertama.
- b. Waktu naik (*rise time*),  $t_r$ : 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%.
- c. Waktu puncak (*peak time*),  $t_p$ : adalah waktu yang diperlukan oleh respon untuk mencapai puncak pertama lewat (*overshoot*).
- d. Maksimum (persen) overshoot (*overshoot maximum*),  $M_p$ : adalah nilai puncak kurva respon diukur dari satuan. Apabila nilai akhir steady state responnya jauh dari satu, maka biasa digunakan persen overshoot maksimum, didefinisikan oleh: Maksimum(persen) overshoot

$$= \frac{c(t_p) - c(\infty)}{c(\infty)} \times 100\% \dots\dots\dots(5)$$

Besarnya (persen) lewatan maksimum menunjukkan kestabilan relatif dari sistem.

- e. Waktu penetapan atau waktu turun (*settling time*),  $t_s$ : adalah waktu yang diperlukan untuk merespon kurva agar dapat mencapai dan tetap berada dalam gugus nilai akhir ukuran yang disederhanakan dengan persentase mutlak harga akhirnya (biasanya 2% atau 5%).

**METODE PENELITIAN**

**Perancangan Kontroler Knowledge Base (*knowledge Base Controller*)**

Dalam menerapkan kendali *knowledge base* ini digunakan motor induksi tiga fasa sebagai *plant*, *inverter* tiga fasa untuk *driver*, *tacho-generator* sebagai sensor, pengereman secara mekanik untuk pengereman dan voltmeter untuk kontrol tegangan.

Variabel yang dikontrol adalah kecepatan putaran motor induksi tiga fasa. Yang mempengaruhi kecepatan ini adalah tegangan dan frekuensi, maka sinyal kontrol yang dihasilkan oleh kontroler akan digunakan untuk mengatur tegangan output inverter, sedangkan frekuensi diatur sesuai dengan *set-point* yang diinginkan.

Tegangan (v) dan frekuensi (f) dari inverter tersebut, merupakan input untuk motor induksi yang akan diatur putarannya. Gambar 6 berikut adalah blok diagram penerapan kontroler *knowledge base* untuk pengaturan kecepatan motor induksi tiga fasa.

**1. Identifikasi Plant**

Identifikasi *plant* bertujuan untuk mendapatkan karakteristik *plant* yang berguna

untuk merancang kontroler dengan semaksimal mungkin. Peralatan yang digunakan adalah:

Motor induksi tiga fasa sebagai *plant*, inverter tiga fasa tipe VFSX-200UP merek toshiba untuk *driver*, PCL-712 yang merupakan *AD/DA converter* dan satu unit PC (*personal computer*).

Untuk mendapatkan nilai *setpoint* yang baik maka *plant* perlu diidentifikasi yang dioperasikan dalam keadaan *open loop*(Ogata, 1997), beban nominal ditetapkan 0.3 amper, *set-point* dimulai dari kecepatan putaran 900 rpm sampai mencapai 1200 rpm dengan kenaikan 100 rpm. Hasilnya seperti tabel 1. berikut ini.

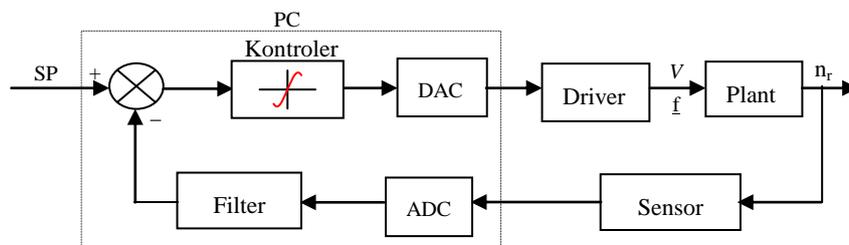
Tabel 1. Menentukan *set-point*

| <i>Set-point</i> (rpm) | 900  | 1000 | 1100 | 1200 |
|------------------------|------|------|------|------|
| Arus beban nominal (A) | 0.60 | 0.40 | 0.32 | 0.30 |

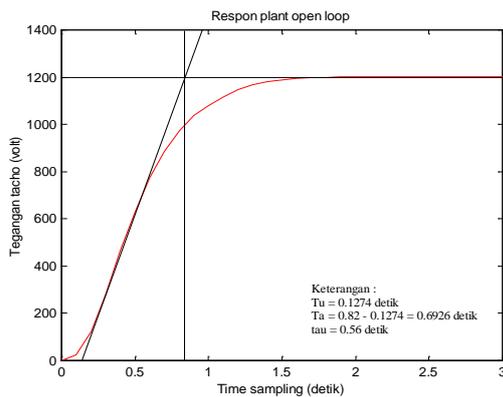
Dari tabel didapat bahwa kecepatan putaran rendah arus motor maksimum, sedangkan kecepatan tinggi arusnya sama dengan arus beban nominal, maka ditetapkan nilai *set-point* 1200 rpm.

Dengan *setpoint* 1200 rpm, motor dioperasikan dalam keadaan *open loop* dengan beban nominal 0.3 ampere, maka dilakukan percobaan. Dengan bantuan *matlab* versi 5.3(Etter, 1993), data hasil identifikasi diolah dengan mengambil 30 titik sampling, dimana responnya seperti gambar 7.

Dengan menggunakan pendekatan praktis model *Strejc*, parameter-parameter kurva respon gambar 7 dapat ditentukan. Berikut ini adalah tabel model *Strejc* (Santoso, 2001).



Gambar 6. Diagram Blok Kontroler Knowledge Base untuk Pengaturan Kecepatan Motor Induksi Tiga Fasa Induksi



Gambar 7. Cara Menentukan Parameter Respon

Tabel 2. Model Strejc

| n | $T_a/T$<br>(T) | $T_u'/T$<br>( $\tau$ ) | $T_u/T_a$<br>(n) |
|---|----------------|------------------------|------------------|
| 1 | 1.000          | 0.000                  | 0.000            |
| 2 | 2.718          | 0.282                  | 0.104            |
| 3 | 3.695          | 0.805                  | 0.218            |
| 4 | 4.463          | 1.425                  | 0.319            |
| 5 | 5.119          | 2.100                  | 0.410            |

Dari penarikan garis didapat:

$$T_u = 0.1274, T_a = 0.6926$$

$$T_u/T_a = 0.1274/0.6926 = 0.1839 \Rightarrow n = 2$$

(dilihat dari tabel 2.)

$$T_a/T = 2.718 \Rightarrow T = T_a/2.718 = 0.6926/2.718 = 0.2548 \text{ detik}$$

$$T_u'/T = 0.282 \Rightarrow T_u' = T \times 0.282 = 0.2548 \times 0.282 = 0.0718$$

$$\tau = T_u - T_u' = 0.1274 - 0.0718 = 0.0556 \text{ detik}$$

Dengan menggunakan *transfer function* model Strejc, yaitu:

$$G(s) = K \frac{e^{-\tau s}}{(Ts + 1)^n} \dots\dots\dots(6)$$

dan *transfer function* sistem orde kedua, yaitu:

$$G(s) = K \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \dots\dots\dots(7)$$

dimana: K = gain total,  $\omega_n$  = frekuensi alamiah dan  $\xi$  = konstanta redaman.

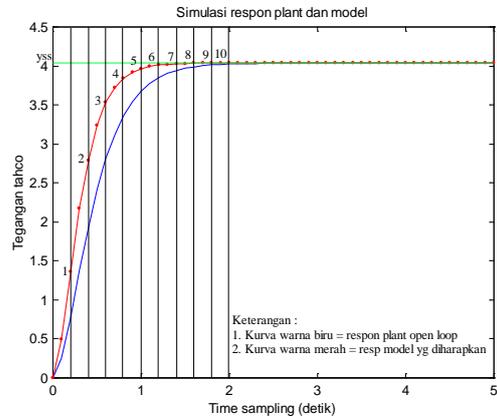
maka didapat parameter respon *plant open loop* dan *model* sebagai berikut:

$$G(s) = \frac{15.4083K}{s^2 + 7.852s + 15.4083} \cdot 3.9253 \text{ dan } \xi = 1 \text{ (plant open loop)}$$

$$2. G(s) = \frac{36K}{s^2 + 12s + 36} \rightarrow \omega_n = 6 \text{ dan } \xi = 1$$

(model yang diharapkan)

Dan gambar responnya seperti gambar 8. berikut ini:



Gambar 8. Respon Plant dan Model

Dari dua respon di atas diambil sepuluh titik yang mewakili seluruh titik yang ada dan didapat data *sampling time* sinyal kontrolnya seperti pada tabel 3.

Tabel 3. Data *Sampling Time* Sinyal Kontrol Plant dan Model

| Titik | Time (detik) | Sinyal kontrol |            |
|-------|--------------|----------------|------------|
|       |              | Plant (yp)     | Model (ym) |
| 1     | 0            | 0.0000         | 0.0000     |
| 2     | 0.2          | 0.7503         | 1.3620     |
| 3     | 0.4          | 1.8787         | 2.7918     |
| 4     | 0.6          | 2.7519         | 3.5296     |
| 5     | 0.8          | 3.3137         | 3.8443     |
| 6     | 1.0          | 3.6446         | 3.9670     |
| 7     | 1.2          | 3.8295         | 4.0123     |
| 8     | 1.4          | 3.9293         | 4.0285     |
| 9     | 1.6          | 3.9820         | 4.0241     |
| 10    | 1.8          | 4.0092         | 4.0360     |

**2. Menentukan Rule Base Kontroler Knowledge Base**

Berdasarkan data *sampling time* pada tabel 3. di atas dapat ditentukan sinyal error (e) dan sinyal kontrol(u) dengan rumus sebagai berikut:

$$a. e_n = y_s - y_{p(n-1)} \dots\dots\dots(8)$$

$$b. u_n = \frac{y_{m(n)}}{y_{p(n)}} \times A \Rightarrow A = 4.0370 \text{ volt} \dots\dots(9)$$

Dari analisa data di atas dapat dibuat aturan *rule base* sebagai berikut:

**if**  $e_1 = 4.0370$  volt, **then**  $u_1 = 4.0370$  volt,

**if**  $e_2 = 3.2867$  volt, **then**  $u_2 = 7.3284$  volt

**if**  $e_3 = 2.1583$  volt, **then**  $u_3 = 5.9992$  volt,

if  $e_4 = 1.2851$  volt, then  $u_4 = 5.1778$  volt  
 if  $e_5 = 0.7233$  volt, then  $u_5 = 4.6834$  volt,  
 if  $e_6 = 0.3924$  volt, then  $u_6 = 4.3941$  volt  
 if  $e_7 = 0.2075$  volt, then  $u_7 = 4.2297$  volt,  
 if  $e_8 = 0.1077$  volt, then  $u_8 = 4.1388$  volt  
 if  $e_9 = 0.0550$  volt, then  $u_9 = 4.0899$  volt,  
 if  $e_{10} = 0.0278$  volt, then  $u_{10} = 4.0640$  volt

**Perancangan Perangkat Lunak (Software)**

1. Program menentukan konstanta *knowledge base*

Nilai konstanta *knowledge base* dapat ditentukan dengan rumus polinomial (Devore, 1982) sampai derajat pangkat ke-n, dalam hal ini menggunakan  $n = 5$ , dengan algoritma sebagai berikut:

- a. Dapatkan *database* hubungan error dan sinyal kontrol sebanyak  $m$  data dari hasil percobaan.
- b. Dari *database* tersebut, dicari statistik data yakni sigma ( $error, error^2, \dots$ ) dan sigma ( $aksi, aksi^2, \dots$ ).
- c. Secara matrik dicari sebanyak 5 konstantanya, dihitung dengan teliti menggunakan *matlab versi 5.3*.

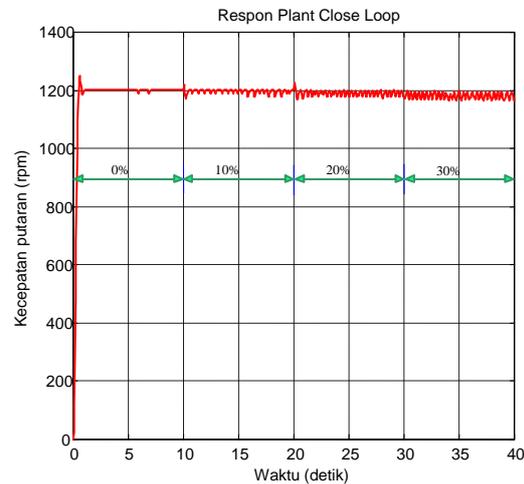
2. Program kontroler *knowledge base*

Algoritma program kontroler adalah sebagai berikut:

- a. Masukkan harga *set-point*
- b. Baca data dari tacho menggunakan ADC dari PCL dan disimpan sebagai *feedback*.
- c. *Feedback* difilter disimpan sebagai *feedback filter*.
- d. Error dicari dengan mengurangi *set-point* dengan *feedback filter*
- e. Error diinputkan untuk menghitung polinomial dengan menggunakan konstanta yang sudah dicari dan hasil polinomial disimpan sebagai *knowledge base*
- f. Harga sinyal kontrol didapat dengan mengalikan *knowledge base* dengan *gain* atau menjumlahkan dengan harga biasa.
- g. Batas harga sinyal kontrol yakni antara 0 sampai 4.999 sesuai karakteristik PCL.
- h. Kirim sinyal kontrol ke *plant* menggunakan DAC dari PCL.
- i. Tampilkan harga-harga yang perlu ditampilkan.
- j. Ulangi langkah no.b sampai diizinkan program keluar.

**HASIL DAN PEMBAHASAN**

Berikut ini adalah kurva respon saat motor bekerja dengan kontroler *knowledge base* beban nominal, kemudian beban dinaikkan 10%, 20%, dan 30% dari beban nominal dengan *set-point* 1200 rpm.



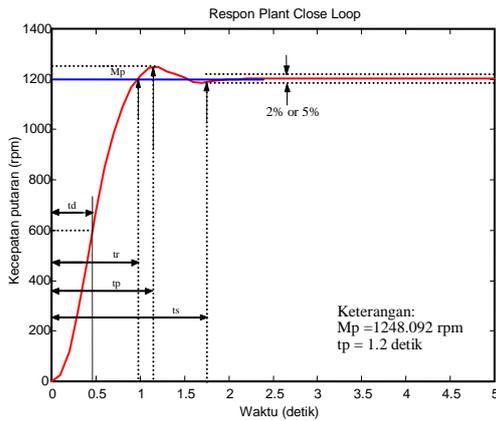
Gambar 9. Respon Plant Berbeban *Closed Loop*

Berdasarkan respon *plant* gambar 9. di atas dapat ditentukan prosentase penurunan kecepatan putaran motor induksi tiga fasa sebagai berikut:

- a. Kecepatan rata-rata putaran motor beban nominal = 1209.079 rpm. Beban dinaikkan 10% dari beban nominal = 1196.4215 rpm. Prosentase penurunan kecepatan putaran motor = 1.059%
- b. Beban dinaikkan 20% dari beban nominal = 1189.515 rpm. Prosentase penurunan kecepatan motor = 1.62%
- c. Beban dinaikkan 30% dari beban nominal = 1180.9768 rpm. Prosentase penurunan kecepatan = 2.32%

Untuk menganalisa karakteristik transien respon *plant* berbeban *close loop* gambar 9, dapat ditampilkan pada gambar 10.

Untuk menganalisa ukuran kualitas respon *plant open loop* dan *model* dipedomani respon sistem orde kesatu (gambar 4) dan respon *plant closed loop* dipedomani respon sistem orde kedua (gambar 5). Secara lengkap ukuran spesifikasi performansinya (Ogata, 1997) dapat dilihat pada tabel 4.



Gambar 10. Respon Plant Transien Closed Loop

Tabel 4. Ukuran Spesifikasi Performansi Berdasarkan Respon Step

| Ukuran Performansi             | Sistem orde kesatu                       | Sistem orde kedua  |
|--------------------------------|--|--|
| Konstanta waktu                | $\tau$                                   | $\frac{1}{\xi \omega_n}$   |
| Delay time: $t_d$              | $\tau \ln 2$<br>(5% ÷ 95%)               | $\frac{0.742}{\xi \omega_n}$   |
| Rise time: $t_r$               | $\rightarrow \tau \ln 19$<br>(10% ÷ 90%) | $\frac{\pi - \tan^{-1} \sqrt{1 - \xi^2}}{\omega_n \sqrt{1 - \xi^2}}$ |
| Settling time : $t_s$<br>(±5%) | $\rightarrow \tau \ln 9$<br>$3 \tau$     | $3 / (\xi \omega_n)$   |
| (±2%)                          | $4 \tau$                                 | $4 / (\xi \omega_n)$   |
| (±0.5%)                        | $5 \tau$                                 | $5 / (\xi \omega_n)$   |
| Time peak : $t_p$              | -  | $\frac{\pi}{\omega_n \sqrt{1 - \xi^2}}$                              |
| % Over-shoot                   | -  | $\exp(-\xi\pi) / \sqrt{1 - \xi^2}$                                   |

1. Respon plant open loop
  - a. *Time constan* ( $\tau$ ) ditentukan secara grafis (dari kurva respon) yaitu: 0,56
  - b. *Delay time* ( $t_d$ ) =  $\tau \ln 2$   
=  $0,56 \ln 2 = 0.3882$
  - c. *Rise time* ( $t_r$ ) =  $t_r(5\% - 95\%)$   
=  $\tau \ln 19 = 0,56 \ln 19 = 1.6489$
  - d. *Settling time* ( $t_s$ ) =  $t_s(\pm 5\%) = 3 \tau$   
=  $3 \times 0,56 = 1.6800$   
 $t_s(\pm 2\%) = 4 \tau = 4 \times 0,56 = 2.240$

2. Respon *model* yang diharapkan
  - a. *Time constan* ( $\tau$ ) ditentukan secara grafis (dari kurva respon) yaitu: 0,1785
  - b. *Delay time* ( $t_d$ ) =  $\tau \ln 2$   
=  $0,1785 \ln 2 = 0.1237$
  - c. *Rise time* ( $t_r$ ) =  $t_r(5\% - 95\%)$   
=  $\tau \ln 19 = 0,1785 \ln 19 = 0.5256$
  - d. *Settling time* ( $t_s$ ) =  $t_s(\pm 5\%) = 3 \tau$   
=  $3 \times 0,1785 = 0.5355$   
 $t_s(\pm 2\%) = 4 \tau = 4 \times 0,1785$   
= 0.7140

3. Respon *plant close loop*  
 Dari gambar 9 di atas dapat ditentukan parameter-parameter spesifikasi performans kontrolnya, yaitu menggunakan rumus spesifikasi respon sistem orde dua sebagai berikut:

- a.  $M_p = 1248.092$  rpm (data hasil percobaan)
- b.  $t_p = 1.2$  detik (data hasil percobaan)
- c.  $y_{ss} = 1200$  rpm

$$d. \xi = \frac{1}{\sqrt{1 + \left[ \frac{\pi}{\ln \left( \frac{M_p - y_{ss}}{y_{ss}} \right)} \right]^2}}$$

$$= \frac{1}{\sqrt{1 + \left[ \frac{\pi}{\ln \left( \frac{1248.092 - 1200}{1200} \right)} \right]^2}}$$

$$\xi = 0.7154$$

$$e. \omega_n = \frac{\pi}{t_p \sqrt{1 - \xi^2}} = \frac{\pi}{1.2 \sqrt{1 - 0.7154^2}}$$

$$\omega_n = 3.7469 \text{ rad} \times \text{s}^{-1}$$

$$f. \text{Time constan,}$$

$$\tau = \frac{1}{\xi \omega_n} = \frac{1}{0.7154 \times 3.7469}$$

$$\tau = 0.3731 \text{ detik}$$

$$h. \text{ Rise time, } t_r = \frac{\pi - \tan^{-1}\left(\frac{\sqrt{1-\xi^2}}{\xi}\right)}{\omega_n \sqrt{1-\xi^2}}$$

$$t_r = \frac{\pi - \tan^{-1}\left(\frac{\sqrt{1-0.7154^2}}{0.7154}\right)}{3.7469\sqrt{1-0.7154^2}}$$

$$t_r = 0.9045 \text{ detik}$$

i. *Settling time,*

$$t_s = \frac{3}{\xi\omega_n} = 1.1192 \text{ detik } (\pm 5\%)$$

$$t_s = \frac{4}{\xi\omega_n} = 1.4922 \text{ detik } (\pm 2\%)$$

$$j. \% \text{Overshoot} = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}} \\ = e^{-\frac{0.7154\pi}{\sqrt{1-0.7154^2}}} \times 100\% = 4\%$$

Hasil lengkapnya seperti tabel berikut ini

### SIMPULAN DAN SARAN

Dari hasil pembahasan dapat diambil beberapa kesimpulan, antara lain:

1. Untuk kontrol *closed loop* bila motor diberi beban secara bertahap kecepatan putaran cenderung stabil hanya saja pada beban 30% kecepatan putaran menjadi 2.32% dari kecepatan putaran beban nominal.
2. Dari ketiga respon (*open loop*, *model* dan *plant closed loop*) tampak respon *plant closed loop* yang diharapkan untuk bisa mencapai respon *model*, tidak tercapai, tetapi kecenderungan respon, mendekati respon model.
3. Kontroler *knowledge base* dapat dikatakan kontroler yang sederhana karena cukup membandingkan respon *plant* dengan respon *model* yang diinginkan, lalu membuat *rule base* dengan parameter *error* dan sinyal kontrol saja.

Agar hasil yang diharapkan tercapai hendaknya hindari memakai motor DC sebagai sensor, karena padanya terjadi *ripple* yang sulit dihilangkan, begitu juga dalam memilih *plant*, gunakanlah *plant* yang baik agar tercapai hasil yang diharapkan.

### DAFTAR RUJUKAN

- Devore, Jay L., (1982), **Probability and Statistics for Engineering and the Sciences**, Wadsworth, Inc, Belmont, California.
- Etter, D.M., (1993), **Engineering Problem Solving with Matlab**, Prentice-Hall, Inc, New Jersey.
- Fakhreddine O, Karray and Clarence de Silva, (2004), **Soft Computing and Intelligent Systems Desain, Theory, Tools and Applications**, Pearson Addison Wesley.
- Lewi, (2001), **Simulasi Perancangan Kontroler Nonlinier Berbasis Pengetahuan untuk Pengaturan Kecepatan Putaran Motor Induksi Tiga Fasa**, Tesis, ITS, Surabaya.
- Ogata, Katsuhiko, (1997), **Teknik Kontrol Automatik**, edisi 2, alih bahasa : Edi Laksono, Erlangga, Jakarta.
- Ramshaw, R.S., (1993), **Power Electronics Semiconductor Switches**, Chapman & Hall
- Suryono, (2005), **Perencanaan Kontroler Knowledge Based PID untuk Mengendalikan Motor DC Penggerak Spindel Pada Prototip Mesin Milling**, Tesis, ITS, Surabaya.
- Santoso, Ari, (2001), **Pengolahan Sinyal & Identifikasi**, Teknik Sistem Pengaturan, ITS, Surabaya.
- Suparman, (1991), **Mengenal Artificial Intelligence**, Andi Offset, Yogyakarta.
- Zuhal, (1990), **Dasar Teknik Tenaga Listrik dan Elektronika Daya**, PT. Gramedia, Jakarta.