

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Motor induksi tiga fasa sebagai pengendali telah banyak dilakukan penelitian, seperti pada perancangan dan implementasi pengendali kontroler sliding mode (SMC) pada pengaturan kecepatan motor induksi tiga fasa. Dalam penelitian terkait menyimpulkan bahwa pada saat dilakukan simulasi kontroler sliding mode dapat mempercepat *rise time* dan juga memperkecil *overshoot*. Simulasi beban minimal dan nominal didapatkan hasil respon yang diinginkan sedangkan pada beban maksimal nilai t yang berbeda[1]. Penelitian yang dilakukan mengkombinasikan SMC dan PID untuk mengendalikan kecepatan motor induksi tiga fasa saat terjadi penambahan beban ataupun adanya gangguan. Dalam penggunaan pengendali hybrid SMC dan PID menunjukkan performansi yang paling baik pada motor DC[4].

Penelitian tentang pengendali *Sliding Mode Control* (SMC) motor induksi tiga fasa dengan metode *Direct Torque Control* (DTC) menggunakan algoritma genetika. Pada pengaturan kecepatan motor induksi *Hybrid SMC* dan DTC memiliki respon dinamik yang cukup baik karna perubahan beban yang terjadi rata-rata kesalahan adalah 0,126%. Sistem pengendali SMC robust untuk perubahan beban yang terjadi terlihat dari nilai maksimum overshoot (2,89%) dan settling time (0,08 detik) yang cukup kecil dibanding dengan DTC (MP=12,82% dan TSS=0,84 detik). Penulis mendapatkan hasil motor induksi akurat untuk mengendalikan putaran motor untuk beban dibawah (TL= -30 Nm)[5].

Hasil penelitian menggunakan pengendali *hybrid SMC* dan PID pada sistem *coople tank* untuk mengendalikan level menyimpulkan bahwa pengujian terhadap pengendali SMC *hybrid* PID kemudian menggunakan nilai parameter yang berbeda, dimana untuk nilai parameter $K_p=3$, $K_i=0.1$, $K_d=0.1$, parameter $K=320$ dan $n=0.05$. Hasilnya adalah respon transien pengendali SMC *hybrid* PID lebih cepat dibanding dengan kedua pengendali tersebut[3].

Penelitian menggunakan pengendali PID *hybrid* Fuzzy pada mesin sentrifugal untuk mengatur kecepatan motor induksi tiga fasa mendapatkan hasil analisa pada simulasi dan implementasi dengan nilai rata-rata *error steady state* saat *starting*, *spinning* dan

breaking sebesar 0.51% dan 1.06% dan karakteristik kerja kurva tidak jauh berbeda dengan referensi yang diberikan[6].

Penelitian pada perancangan dan simulasi untuk mengatur kecepatan motor induksi tiga fasa dengan menggunakan metode *Direct Torque Control* (DTC). Hasil respon torsi dari percobaan yang diperlihatkan bahwa respon torsi dapat mengikuti torsi acuan yang diberikan. Pada motor 10 HP respon torsi memiliki daerah *ripple* mencapai 10Nm dan saat kecepatan putaran motor dalam keadaan tunak maka respon torsi juga memiliki keadaan tunak pada 0 Nm namun tetap ada *ripple*. Sedangkan pada motor induksi 200 HP respon torsi memiliki daerah *ripple* yang besar yaitu 150Nm[7].

Berdasarkan penelitian-penelitian di atas motor induksi tiga fasa masih banyak memiliki kekurangan apabila tidak menggunakan pengendali. Oleh sebab itu, penulis tertarik meneliti motor induksi tiga fasa (plant) dan pengendali sangat dibutuhkan untuk mendapatkan hasil yang baik. Penelitian – penelitian sebelumnya sudah menjabarkan bahwa plant menggunakan pengendali sangat mempengaruhi hasil terhadap nilai acuan yang kita dapat. Pada beberapa penelitian sistem yang telah menggunakan pengendali *hybrid* SMC dan PID telah membuktikan bahwa pengendalian ini memiliki keuntungan. Untuk mewujudkan penelitian ini penulis mengangkat judul berdasarkan hasil penelitian sebelumnya dengan judul “Analisa pengendalian kecepatan motor induksi tiga fasa menggunakan pengendali *hybrid* SMC dan PID”.

2.2 Dasar Teori

2.2.1 Motor Induksi Tiga Fasa

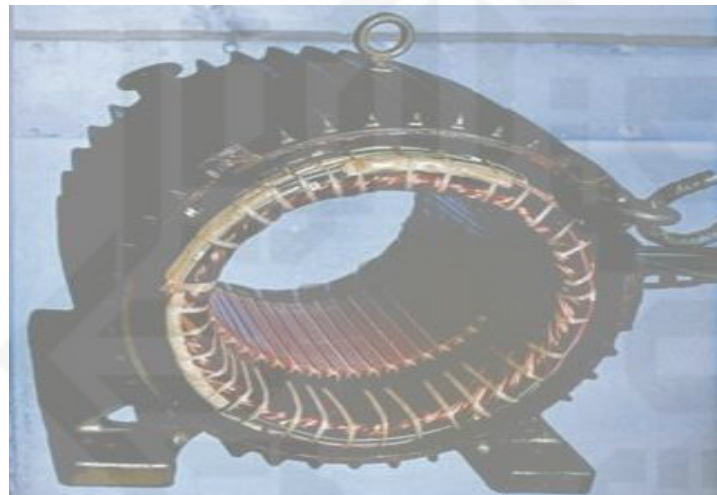
Motor induksi tiga fasa adalah suatu mesin listrik yang merubah energi listrik menjadi energi gerak dengan menggunakan gandengan medan listrik dan mempunyai slip antara medan stator dan medan rotor. Kenyataannya arus rotor motor ini bukan diperoleh dari suatu sumber listrik, tetapi merupakan arus yang terinduksi akibat adanya perbedaan relatif antara putaran rotor dengan medan putar oleh sebab itu, dinamakan sebagai motor induksi [1].

A. Kontruksi Motor Induksi Tiga

Pada kontruksi motor induksi tiga fasa hampir sama dengan motor jenis lainnya yaitu motor jenis ini memiliki bagian rotor dan stator.

a. Stator

Stator adalah bagian dari mesin yang pasif atau tidak berputar dan terletak pada bagian luar. Stator mempunyai alur-alur sebagai tempat meletakkan kumparan dan terbuat dari besi bundar berlaminasi[1].

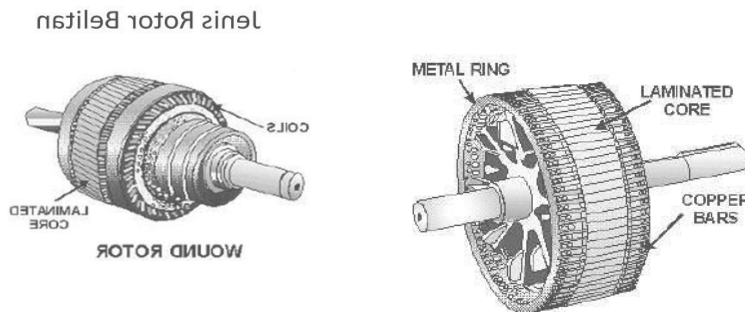


Gambar 2.1 Stator motor induksi tiga fasa

b. Rotor

Rotor adalah bagian dari mesin yang aktif atau berputar dan terletak di bagian dalam stator. Ada dua bagian dari rotor yaitu rotor sangkar dan rotor kumparan. Rotor sangkar merupakan bagian dari mesin yang berputar bebas dan letaknya bagian dalam dan terbuat dari besi laminasi yang mempunyai slot dengan batang aluminium atau tembaga yang dibungkan dengan singkat pada ujungnya. Sedangkan, rotor kumparan adalah kumparan yang dihubungkan ke batang di bagian dalam dan ujung lain dihubungkan dengan *slipring* ke tahanan luar. Kumparan dapat dikembangkan sebagai pengatur kecepatan putaran motor[1].

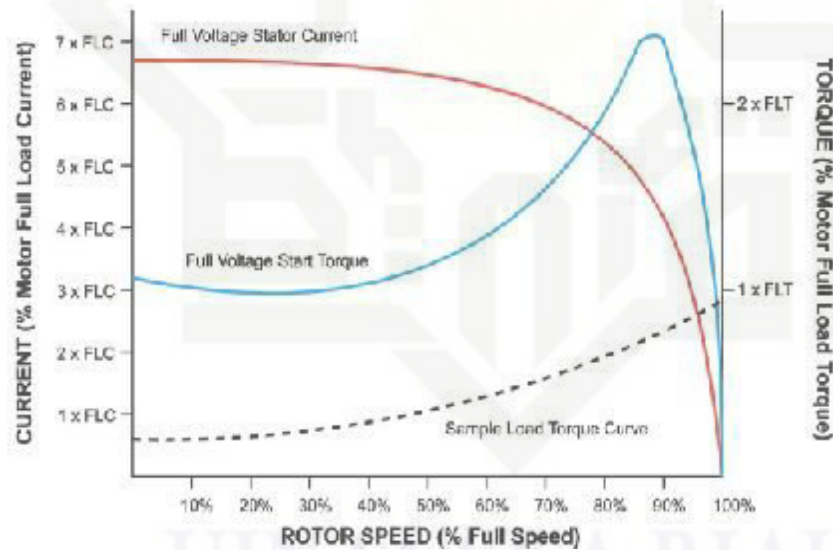
- Hak Cipta Dilindungi Undang-Undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Gambar 2.2 rotor kumparan(kiri) dan rotor sangkar(kanan)

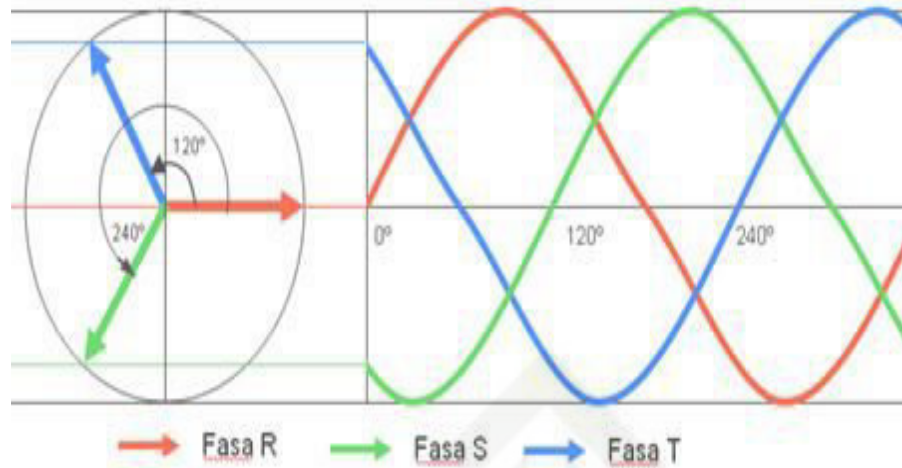
B. Karakteristik Motor Induksi Tiga Fasa

Pada gambar 2.3 dapat dilihat ilustrasi pada motor induksi saat kecepatan rotor mulai nol hingga kecepatan maksimalnya, untuk bergerak dengan kecepatan rendah, arus yang dibutuhkan sangat tinggi hingga mencapai tujuh kali arus kerjanya, sedangkan saat kecepatannya mulai tinggi, arus yang dibutuhkan berkurang hingga arus kerja normalnya[8].



Gambar 2.3 Grafik torsi motor induksi tiga fasa

Pemanfaatan perbedaan fasa sumber untuk menimbulkan gaya putar pada rotor merupakan prinsip motor induksi tiga fasa saat bekerja. Jika pada motor induksi satu fasa untuk menghasilkan beda fasa diperlukan penambahan komponen kapasitor, pada motor induksi tiga fasa perbedaan fasa sudah didapat langsung dari sumber seperti terlihat pada gambar arus fasa berikut ini[8].

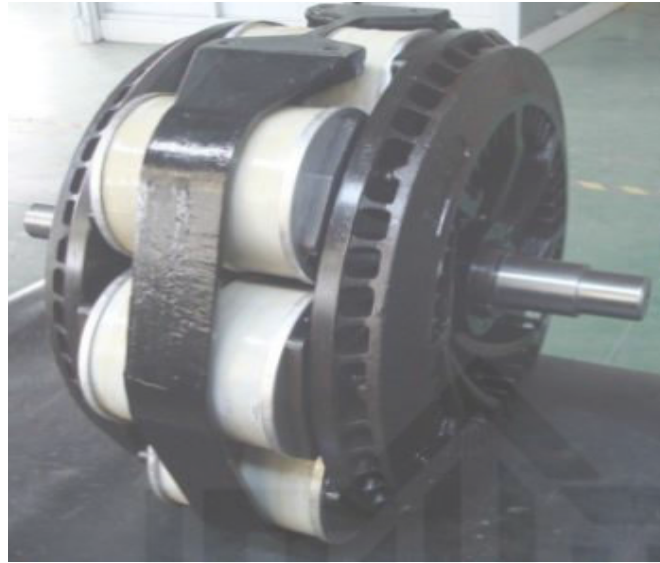


Gambar 2.4 Grafik arus tiga fasa

2.2.2 Rem Elektromagnetik

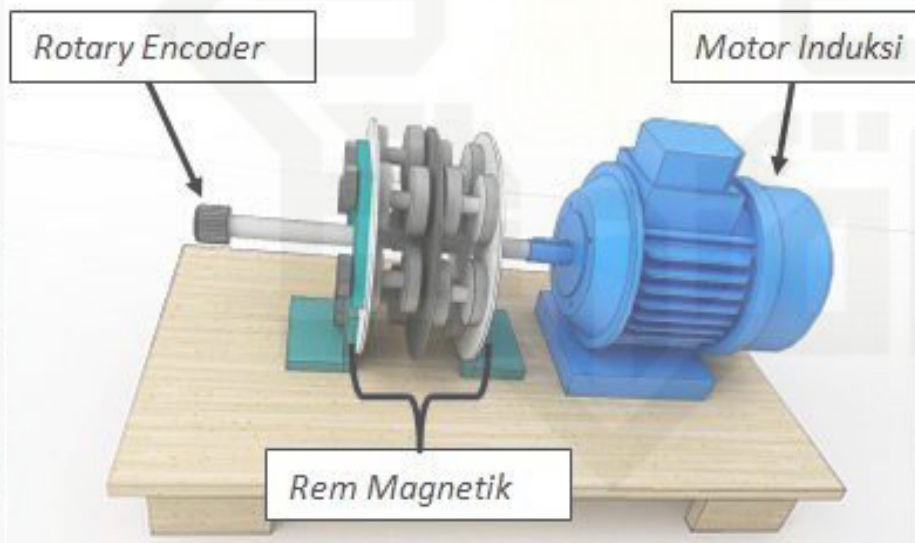
Rem merupakan suatu alat yang digunakan untuk menurunkan kecepatan pada saat aksi deselerasi dalam selang waktu yang ditentukan. Penggunaan gaya gesek memberikan gaya lawan terhadap gaya gerak adalah tipe rem yang umum digunakan. Namun dalam sistem pengereman elektromagnetik menggunakan gaya elektromagnetik untuk memperlambat suatu gerakan, yang biasanya adalah gerakan poros. Sebuah piringan dengan bahan logam *non-ferromagnetik* terpasang dengan poros yang berputar. Piringan tersebut diapit oleh sisi stator berupa sistem lilitan elektromagnetik yang dapat membangkitkan medan magnet dari aliran listrik. Arus listrik menimbulkan medan magnet pada lilitan dan logam piringan yang memotong medan magnet tersebut akan menimbulkan arus *eddy* pada piringan itu sendiri. Arus *eddy* ini akan menimbulkan medan magnet yang arahnya berlawanan dengan medan magnet sebelumnya, sehingga menghambat gerakan putar dari poros tersebut. Rem elektromagnetik akan optimal untuk memberikan penurunan kecepatan, bukan untuk menghentikan gerak suatu objek. Sehingga Rem ini sering diaplikasikan untuk sistem pengereman pada *roller coaster*, kereta api dan juga digunakan pada alat dinamometer untuk pengukuran torsi suatu mesin[1].

Bentuk fisik dari rem elektromagnetik ditunjukkan oleh Gambar 2.6 dan 2.7. Arus *eddy* yang melingkar menyebabkan medan magnet induksi melawan arah medan magnet mula-mula. Hal ini menyebabkan gaya pengereman yang melawan arah kecepatan konduktor yang bergerak memotong medan magnet dari kedua solenoid[1].



Gambar 2.5 Rem elektromagnetik

Dalam pengaplikasiannya pengereman pada motor induksi tiga fasa ditunjukkan pada gambar 2.7 berikut:



Gambar 2.6 Sistem motor induksi tiga fasa dengan pengereman elektromagnetik

2.2.3 Identifikasi Sistem Motor Induksi Tiga Fasa

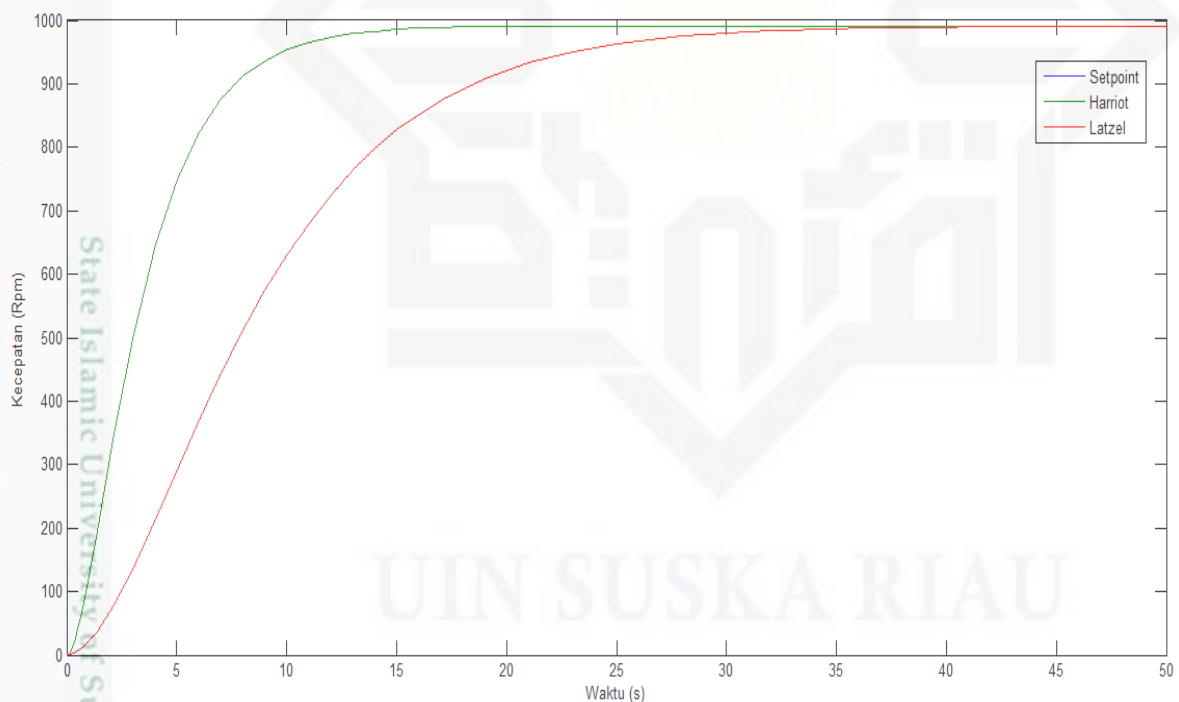
Tahap yang menggambarkan perilaku sistem ataupun mencari pemodelan matematika dari sistem merupakan tahapan identifikasi. Dalam penelitian perancangan dan implementasi kontroler sliding mode pada pengaturan kecepatan motor induksi tiga fasa sistem terlebih dahulu diidentifikasi secara *open loop* dengan memberikan sinyal uji step dan melihat respon berupa kecepatan putar, kemudian hasil dari respon *plant* yang terbaca

oleh *rotary encoder* ditampilkan pada komputer melalui proses akuisisi data dengan PLC *Mitsubishi*[1]. Adapun sinyal uji step dan respon keluaran ditunjukkan pada Tabel 2.1 berikut:

Tabel 2.1 sinyal uji step tanpa beban dan keluaran pada keadaan *open loop*

Sinyal Uji Step (RPM)	Respon Keluaran latzel (RPM)	Respon Keluaran Harriot (RPM)
1000 RPM	991,0754 RPM	991,366548 RPM

Dimana pada tabel 2.1 menunjukkan setpoint 1000 rpm, respon keluaran untuk latzel 991,0754 rpm dan respon keluaran untuk harriot 991.366548 yang dilihat bentuk grafik yang ditunjukkan pada gambar 2.7 berikut:



Gambar 2.7 Respon asli dari hasil uji step (*plant* sebenarnya)

Pada gambar 2.7 adalah respon asli dari *plant* berupa kecepatan putar yang akan dicari fungsi alihnya dengan ketentuan:

$X_{ss} = \text{Setpoint}$

$Y_{ss} = \text{Respon keluaran}$

$K = \text{Gain keadaan tunak}$

Untuk mendapatkan K didapatkan dari X_{ss}/Y_{ss} Pada penelitian ini, fungsi alih dicari dengan pendekatan metode latzel.

Setelah itu salah satu sampel data berupa grafik sebagai respon asli sistem motor induksi tiga fasa diterapkan pada 6 model pendekatan matematika sistem berdasarkan paper penelitian internasional, *Experimental Identification of Stable Nonoscillatory Systems from Step-Responses by Selected Method*[2].

2.2.4 Metode latzel

Metode *latzel* merupakan metode yang umum digunakan pada kasus simulasi, metode yang eksperimental berdasarkan jurnal internasional yang diterbitkan oleh Ing pavel jakoubek[2]. Metode ini didasarkan pada perkiraan sistem dengan model fungsi transfer. Pada metode *latzel* model fungsi alih dilambangkan $G(s)$. Untuk mendapatkan nilai dari model fungsi alih, maka gain waktu keadaan tunak akan berbanding terbalik dengan hasil jumlah dari waktu arus ditambah satu. Sehingga didapatkan persamaan sebagai berikut :

$$G(s) = \frac{K}{(T_s s + 1)^n} \tag{2.1}$$

Dimana :

$G(s)$ = model fungsi alih *latzel*

K = waktu tunak

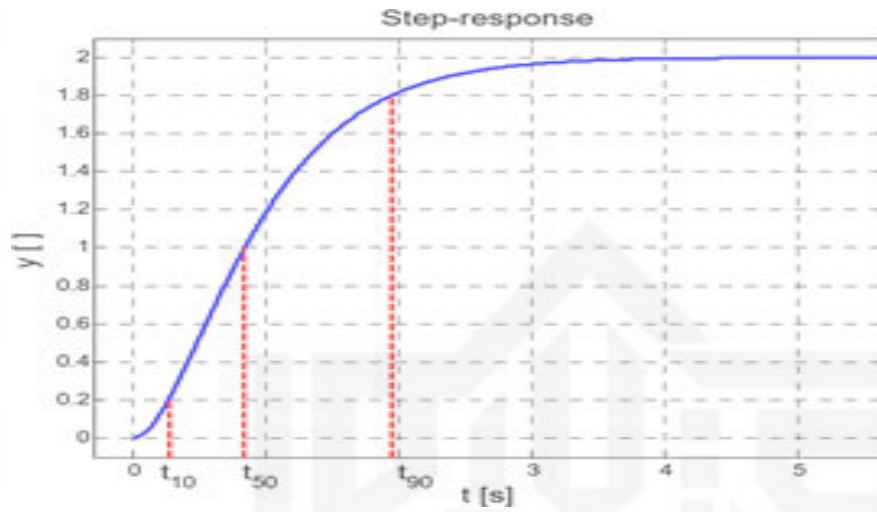
T_s = waktu alih *latzel*

T_1 merupakan waktu konstan sehingga dapat dicari dengan persamaan sebagai berikut:

$$T_1 = (\alpha_{10}t_{10} + \alpha_{50}t_{50} + \alpha_{90}t_{90}) \tag{2.2}$$

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

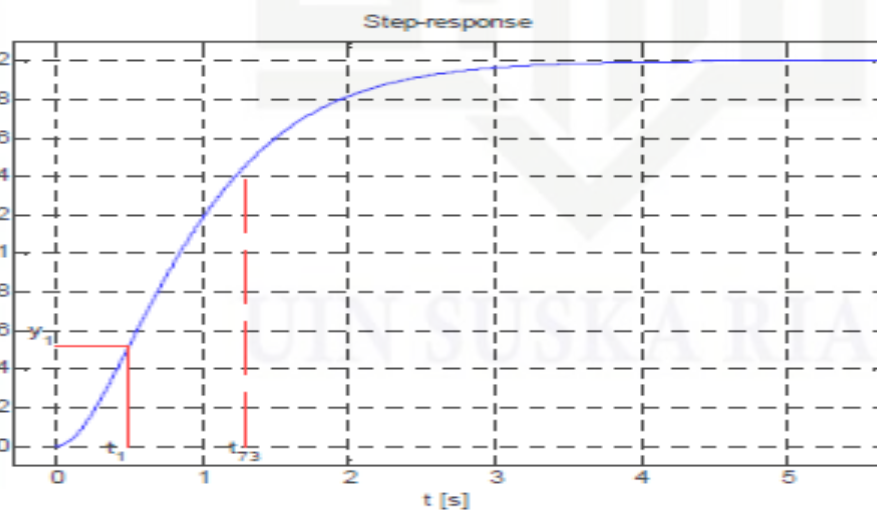
Dimana t_{10} , t_{50} , dan t_{90} adalah waktu, nilai waktu yang dibutuhkan saat respon sistem mencapai 10%, 50%, 70%.



Gambar 2.8 Step Respon dari Sistem dengan waktu t_{10} , t_{50} , dan t_{90}

2.2.5 Metode Harriot

Metode *Harriot* adalah salahsatu metode eksperimental pada jurnal Internasional yang ditulis oleh Ing Pavel Jakoubek[2]. Pada metode *Harriot*, mencari fungsi alih dihitung berdasarkan respon keluaran sistem dalam bentuk grafik. Dengan mula-mula dicari t_1 dan t_{73} sebagai contoh pada Gambar 2.9 berikut:



Gambar 2.9 Step Respon dari Sistem dengan waktu t_1 , dan t_{73}

Pada metode *Harriot* diberikan persamaan:

$$G_H(S) = \frac{K}{(\tau_{H1}s + 1) + (\tau_{H2}s + 1)} e^{-\tau_d Hs} \quad (2.3)$$

Tentukan nilai t_{73} yaitu saat respon bernilai 73 % dari yss.

$$T_{dH} = 1.937 t_{33} - 0.937 t_{70} \quad (2.4)$$

Kemudian hitung nilai t_{dH} , $\tau_{H1} + \tau_{H2}$, dan t_i melalui persamaan:

$$\tau_{H1} + \tau_{H2} = \frac{\tau_{73}}{1.3} \quad (2.5)$$

$$t_i = 0,5 (\tau_{H1} + \tau_{H2}) \quad (2.6)$$

Tentukan nilai y_i yaitu nilai respon mencapai t_i pada Grafik.

Setelah itu didapatkan nilai $\frac{\tau_{H1}}{\tau_{H1} + \tau_{H2}}$ dari hasil y_i/y_{ss} .

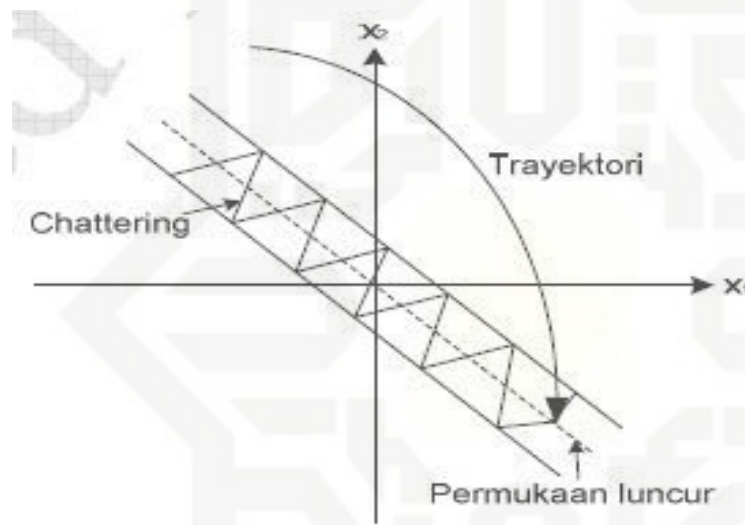
Dengan mensubstitusi nilai $\tau_{H1} + \tau_{H2} = \frac{\tau_{73}}{1.3}$ pada persamaan $\frac{\tau_{H1}}{\tau_{H1} + \tau_{H2}}$ maka didapat nilai τ_{H1} dan τ_{H2} [2].

2.3 Sliding Mode Control (SMC)

Pengendali Modus Luncur (PML) atau istilah populernya sering disebut dengan *Sliding Mode Controller* (SMC) merupakan sebuah kendali umpan balik pensaklaran berkecepatan tinggi yang efektif dan kokoh dalam mengendalikan sistem linear maupun non-linear. Sistem kendali ini kokoh karena menyediakan sebuah metoda perancangan sistem yang tidak peka terhadap ketidakpastian parameter lingkungan dan gangguan dari luar [9].

Pada prinsipnya SMC menggunakan sebuah hukum kendali pensaklaran berkecepatan tinggi untuk membawa trayektori status dari sistem linear atau non-linear ke dalam sebuah permukaan tertentu dalam ruang status yang disebut permukaan luncur atau *sliding surface*, kemudian trayektori status tersebut dipelihara agar tetap meluncur pada permukaan tersebut. Proses pemeliharaan trajektori status pada permukaan luncur mengakibatkan terjadinya osilasi pada permukaan luncur. Osilasi ini sering disebut dengan *chattering*. Fenomena *chattering* pada permukaan luncur akan berdampak pada stabilitas dari sistem kendali [9].

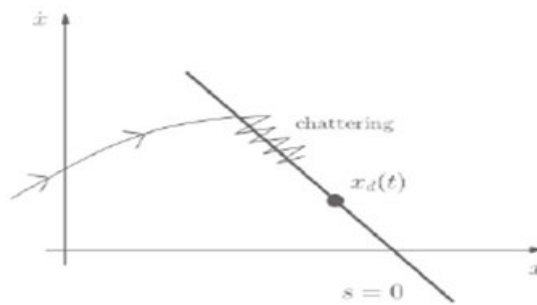
Pada SMC waktu yang dibutuhkan oleh status dalam mencapai permukaan luncur akan berdampak terhadap kecepatan tanggapan sistem terhadap waktu. Waktu yang dibutuhkan oleh status untuk mencapai permukaan luncur sering disebut dengan *hitting time*. Salah satu keuntungan dari SMC adalah ketika sistem mengenai permukaan luncur, maka sistem tersebut tidak akan peka terhadap ketidakpastian parameter lingkungan dan gangguan dari luar. Untuk menjaga perilaku sistem sehingga tidak peka terhadap perubahan lingkungan, maka dibutuhkan *hitting time* yang kecil agar sistem cepat mengenai permukaan luncur. *Hitting time* dan *chattering* yang kecil merupakan dua hal yang sangat penting dalam merancang SMC [9]



Gambar 2.10 Diagram trayektori status

2.3.1 Chattering

SMC merupakan pengendali berumpan balik dengan pensaklaran berkecepatan tinggi (*High Speed Swtitching Feedback*), sehingga dalam proses SMC mengalami *chattering* yang dapat mengganggu kestabilan sistem[3]. *Chattering* adalah sebuah fenomena perubahan kendali dengan frekuensi tinggi ketika trayektori disekitar permukaan bidang luncur dan ketika harga signum sering berubah-ubah. Beberapa cara untuk mengurangi *Chattering* adalah dengan memperluas fungsi signum menjadi saturasi, atau mengganti signum dengan arcuss tangen[1].



Gambar 2.11 Fenomenan *chattering effects*

2.3.2 Perancangan permukaan luncur

Perancangan permukaan luncur dapat dirumuskan dengan persamaan awal sebagai berikut[3] :

$$x^n(t) = f(x) + b(x, t).U + d(t) \quad (2.7)$$

Dimana U merupakan input kendali, x merupakan faktor keadaan, f(x,t) dan b(x,t) berupa fungsi terbatas, d(t) gangguan eksternal. Jika x_d merupakan x maka *tracking error* dapat dinyatakan dengan :

$$e(t) = x(t) - x_d(t)$$

Fungsi *switching* yaitu permukaan S (x,t) didalam ruang keadaan R_n , memenuhi persamaan :

$$S(x, t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} e \quad (2.8)$$

Dengan λ berupa konstanta positif. Dimana fungsi *switching* ini digunakan untuk menentukan besarnya nilai u agar memenuhi kondisi *sliding*. Permukaan luncur (*sliding surface*) merupakan persamaan yang memenuhi :

$$S(x, t) = 0 \quad (2.9)$$

Berdasarkan nilai kendali input pada SMC bergantung pada nilai S, sehingga memenuhi pertidaksamaan yang disebut kondisi *sliding*. Kondisi tersebut ditulis dalam bentuk sebagai berikut :

$$S\dot{S} \leq -\eta|S| \quad (2.10)$$

Keterangan :

- S = Permukaan luncur
- λ = Konstanta positif
- n = Orde sistem
- e = Error

2.3.3 Sinyal Kendali

Untuk mendapatkan suatu sinyal kendali yang mampu membawa trayektori menuju permukaan luncur dan mempertahankan status trayektori tetap berada disekitar permukaan luncur. Dalam kendali SMC terdapat dua sinyal kendali. Yang pertama adalah sinyal kendali ekivalen (u_{eq}) yang berfungsi untuk membawa status trayektori menuju permukaan luncur. Dan sinyal kendali yang kedua adalah sinyal kendali natural (u_n) yang berfungsi untuk mempertahankan status trayektori agar tetap berada pada permukaan luncur [3].

$$u = u_{eq} + u_n \tag{2.11}$$

Keterangan :

- u = Sinyal Kendali
- u_{eq} = Sinyal Kendali ekivalen.
- u_n = Sinyal Kendali Natural

2.3.4 Persamaan Lyapunov

Lyapunov adalah salah satu metode yang dapat digunakan untuk menguji kestabilan suatu sistem proses. *Lyapunov* mengembangkan metode untuk menentukan kestabilan berdasarkan pengamatan energi yang disimpan. Dengan menggunakan persamaan *lyapunov*, maka stabilitas dari sistem proses yang berbentuk linear maupun tidak linear dapat ditentukan.

Alexander Mikhailovich Lyapunov, merupakan seorang ilmuwan dari Rusia, telah mengembangkan metode untuk menentukan stabilitas dari sistem proses didasarkan dari penghematan energi yang disimpan. Menurut *Lyapunov*, hubungan antara kestabilan dan

Hak Cipta Dilindungi Undang-Undang
 1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber.
 a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

energi adalah sebagai berikut: Sistem dikatakan stabil apabila energi yang disimpan makin lama makin besar, maka osilasi yang terjadi semakin lama semakin besar juga. Agar sistem dapat dianalisis kestabilannya maka perlu dibuat modek matematis yang menghubungkan antara masukan, proses dan keluaran.

Pada pengendali, diharapkan agar keluaran suatu respon menuju pada *setpoint* yang di berikan. Hal ini identik pada nilai *error* yang menuju nol. Berdasarkan teori *state space* dimana *error* dapat diambil sebagai variabel state maka dapat juga diambil suatu fungsi saklar:

$$V(x) = \frac{1}{2} [S]^2 \tag{2.12}$$

Yang memenuhi :

$$\dot{V}(x) = S\dot{S} < 0 \tag{2.13}$$

2.4 *Propotional Integral Derivative (PID)*

Pengendali PID adalah pengendali otomatis penggabungan dari ketiga metode kendali proporsional, inetegral, dan derivatif, pada masing-masing memiliki aksi yang khas pada tiap-tiap kekurangan pengendali P, I dan D dapat saling menutupi dengan menggabungkan ke tiganya dan merangkai secara paralel menghasilkan pengendali PID. Pada masing- masing aksi keseluruhan memiliki tujuan untuk mempercepat reaksi sebuah sistem yang mampu menghilangkan *offset* dan menghasilkan perubahan yang besar [3].

Untuk menunjukan hubungan dari ketiga pengendali P,I dan D dapat dinyatakan dengan rumus berikut:

$$u(t) = Kp e(t) + Ki \int_0^t e(t) dt + Kd \frac{de}{dt} \tag{2.14}$$

Persamaan 2.3 dirubah kedalam bentuk *Laplace* menjadi :

$$\frac{U(s)}{E(s)} = Kp + \frac{Ki}{s} + Kd \tag{2.15}$$

Dimana:

$Kp = Proporsional Gain.$

$Ki = Integral Gain.$

$K_d = \text{Derivatif Gain}$.

Pengendali *proporsional* (K_p) akan memberikan efek mengurangi waktu naik, tetapi tidak menghapus *error steady state*. Pengendali *integral* (K_i) akan memberikan efek menghapus *error steady state* tetapi berakibat buruk pada saat menanggapi respon. Pengendali *derivatif* (K_d) berdampak pada stabilitas sistem, mengurangi kesalahan keadaan tunak dan menaikkan respon transien. Untuk hubungan dari ketiga aksi di tunjukkan pada tabel 2.3.

2.4.1 Penelaan Parameter PID

Istilah penalaan (*tunning*) untuk menggambarkan metode-metode yang digunakan dalam memilih pengendalian yang cocok agar mendapat unjuk kerja pada nilai respon sistem yang diinginkan, dengan melakukan penalaan dari masing-masing aksi kendali akan didapatkan respon sistem sesuai dengan setpoint yang diberikan sehingga dapat diterapkan pada sistem tertentu. Pada penelitian ini dilakukan penalaan pada pengendali PID dengan melakukan penalaan pada masing-masing aksi kendali pada parameter PID[3]. Adapun Tabel yang menggambarkan hubungan dari masing-masing aksi kendali pada parameter PID ditunjukkan pada Tabel 2.2 berikut:

Tabel 2.2 Hubungan antara pengendali K_p , K_i , dan K_d [3]

Respon lup tertutup	Waktu naik	<i>Overshoot</i>	Waktu turun	Kesalahan keadaan tunak
K_p	Menurun	Meningkat	Perubahan kecil	Menurun
K_i	Menurun	Meningkat	Meningkat	Hilang
K_d	Perubahan kecil	Meningkat	Menurun	Perubahan kecil

2.4.2 Pengendali PID Metode *Heruistik*

Metode *Heuristik* merupakan sebuah metode pemecahan masalah menggunakan ekspolarasi dan cara coba-coba. *Heuristik* adalah suatu metode untuk bisa menyelesaikan solusi secara penalaan. Rancangan metode *Heuristik* ini diperoleh dengan cara penambahan parameter yang disesuaikan dengan kinerja plant yang akan dikendalikan. Untuk merancang sistem pengendali PID dilakukan pencarian nilai besarnya K_p , K_i , dan K_d Maka pengujian dilakukan dengan beberapa tahap, dengan penalaan (*Heuristic method*).

Penalaan pengendali dimulai hanya dengan menggunakan pengendali 'P', kemudian baru ditambahkan dengan pengendali 'I' dan yang terakhir ditambahkan dengan pengendali 'D'. Pemberian nilai parameter disesuaikan dengan karakteristik respon sistem yang diperoleh.

2.5 Program Matlab

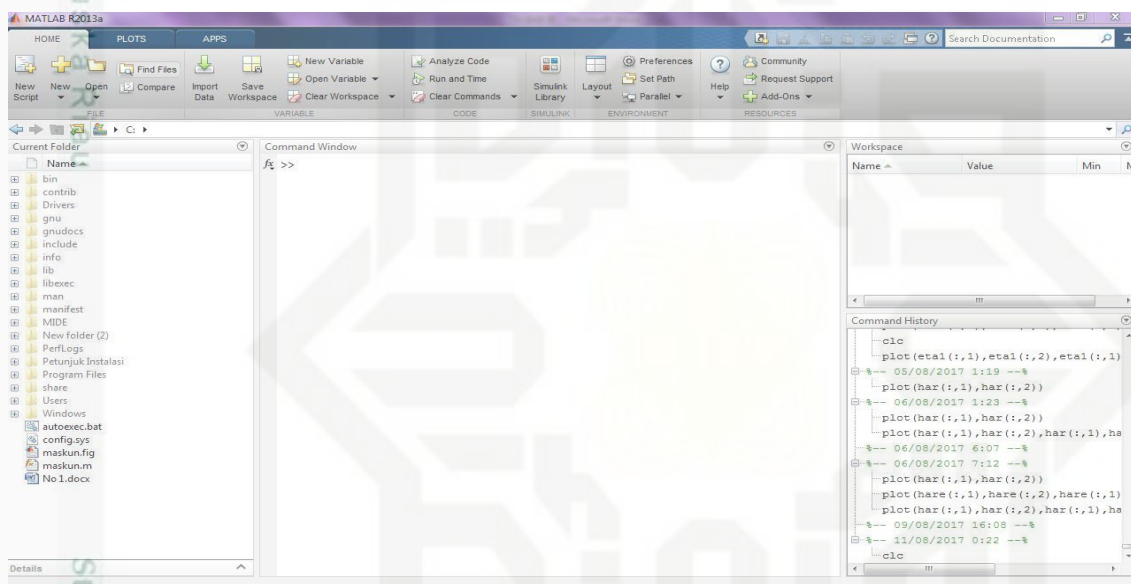
Matlab merupakan sebuah singkatan dari *Matrix Laboratory*, yang pertama kali diperkenalkan oleh *University of New Mexico* dan *University of Stanford* pada tahun 1970. Matlab adalah sebuah program untuk analisis dan komputasi numerik dan merupakan suatu bahasa pemrograman matematika lanjutan yang dibentuk dengan dasar pemikiran menggunakan sifat dan bentuk matriks. Pada awalnya, program ini merupakan *interface* untuk koleksi rutin-rutin numerik dari proyek *Linpack* dan *Eispack*, dan dikembangkan menggunakan bahasa *Fortran* namun sekarang merupakan produk komersial dari perusahaan *Mathworks Inc.* yang dalam perkembangan selanjutnya dikembangkan menggunakan bahasa C++ dan *assembler* (utamanya untuk fungsi-fungsi dasar Matlab).

Matlab telah berkembang menjadi sebuah pemrograman yang canggih yang berisi fungsi-fungsi built-in untuk melakukan tugas pengolahan sinyal, aljabar linier dan kalkulasi matematis lainnya. Matlab juga berisi *toolbox* yang berisi fungsi-fungsi tambahan untuk aplikasi khusus. Matlab bersifat *extensible*, dalam arti bahwa seorang pengguna dapat menulis fungsi baru untuk ditambahkan pada *library* ketika fungsi-fungsi *built-in* yang tersedia tidak dapat melakukan tugas tertentu.

Matlab yang merupakan bahasa pemrograman tingkat tinggi berbasis pada matriks sering digunakan untuk teknik komputasi numerik, yang digunakan untuk menyelesaikan

masalah-masalah yang melibatkan operasi matematika elemen, matrik, optimasi, aproksimasi dll. Sehingga Matlab banyak digunakan pada :

- a. Matematika dan komputansi.
- b. Pengembangan dan algoritma.
- c. Pemrograman modeling, simulasi dan pembuatan prototipe.
- d. Analisa data, eksplorasi, dan visualisasi.
- e. Analisis numerik dan statistik.
- f. Pengembangan aplikasi teknik.



Gambar 2.12 Tampilan dasar matlab

2.5.1 Simulink Matlab

Simulink adalah sebuah kumpulan aplikasi dalam Matlab untuk melakukan *modeling*, simulasi, dan melakukan analisis dinamik pada suatu sistem. Program *simulink* memudahkan *user* untuk membuat suatu simulasi lebih interaktif. Tiruan sistem diharapkan mempunyai perilaku yang sangat mirip dengan sistem fisik. Jika digunakan dengan benar, simulasi akan membantu proses analisis dan desain sistem. *Simulink* mempunyai kumpulan *toolbox* yang dapat kita gunakan untuk analisis sistem linear dan nonlinear.

Simulink dalam matlab juga dapat menunjukkan performasi sistem dalam bentuk grafik dua dimensi ataupun tiga dimensi. Dalam perancangannya *user* dibantu oleh blok-

blok diagram yang dapat dengan mudah diatur sedemikian rupa, sesuai dengan model matematis dari sistem atau *plant* yang dikendalikan.

2.5 Respon Waktu (*time respons*)

Dalam beberapa kasus peraktek, karakteristik kerja yang diinginkan dari suatu sistem kendali dicirikan oleh suku besaran daerah asal waktu. Sistem dengan tenaga tidak dapat memberikan repon seketika dan akan menunjukkan tanggapan transien walaupun diberi masukan atau gangguan. Biasanya, karakteristik kinerja suatu sistem kendali dicirikan oleh suku respon transien terhadap masukan tangga satuan karena hal itu mudah ditimbulkan dan cukup drastis. Jika respon terhadap masukan tangga diketahui, secara matematis dapat dihitung respon untuk sembarang masukan[10].

Respon transien suatu sistem terhadap masukan tangga satuan tergantung transien terhadap variasi sistem, terdapat cara praktis yang biasa digunakan. Yaitu dengan syarat awal standar bahwa sistem mula-mula diam dengan keluaran dan semua turunannya nol. Dengan demikian, karakteristik tanggapannya dapat dengan mudah dibandingkan.

Respon transien suatu sistem kendali secara praktek selalu menunjukkan osilasi teredam sebelum mencapai keadaan tunaknya. Dalam menggolongkan karakteristik tanggapan transien suatu sistem kendali terhadap masukan tangga satuan secara umum dikelompokkan sebagai berikut[10].

1. Konstanta Waktu (t)

Waktu yang dibutuhkan respon mulai dari $t = 0$ sampai respon mencapai 63.2 % dari respon *steady state*.

2. Waktu Tunda (t_d)

Waktu tunda merupakan waktu yang diperlukan oleh tanggapan untuk mencapai setengah nilai akhir untuk waktu yang pertama.

$$t_d(3\%) \approx t \ln 12 \tag{2.16}$$

3. Waktu Naik (t_r)

Waktu naik merupakan waktu yang diperlukan oleh tanggapan untuk naik dari 10% menjadi 90%, 5% menjadi 95%, atau 0 menjadi 100% dari nilai akhir yang biasa digunakan.

$$t_r(5\% - 95\%) \approx t \ln 19 \tag{2.17}$$

$$t_r(10\% - 90\%) \approx t \ln 9 \tag{2.18}$$

4. Waktu Puncak (t_p)

Waktu puncak merupakan waktu yang diperlukan tanggapan untuk mencapai puncak pertama *overshoot*.

5. *Overshoot* Maksimum (M_p)

Overshoot maksimum merupakan nilai puncak kurva tanggapan diukur dari satuan. Apabila nilai akhir keadaan tunak tanggapannya jauh dari satu, maka biasanya digunakan persen *overshoot* maksimum dan didefinisikan oleh :

$$M_p = \frac{c(t_p) - c(\infty)}{c(\infty)} 100\% \tag{2.19}$$

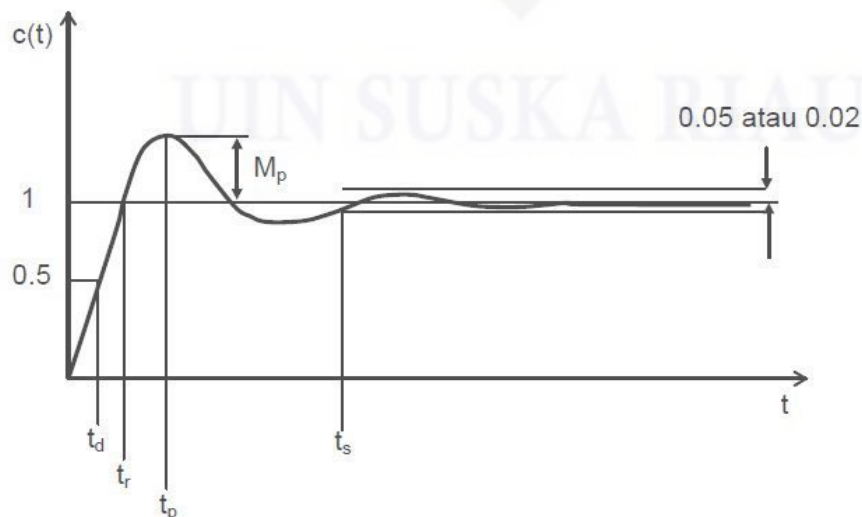
6. Waktu Tunak (t_s)

Waktu tunak merupakan waktu yang diperlukan untuk menanggapi kurva agar dapat mencapai dan tetap berada dalam gugus nilai akhir ukuran yang diserhanakan dengan presentase mutlak harga akhirnya (biasanya 2% atau 5%) waktu tunak tadi dihubungkan dengan sebesar sistem kendali. Apabila kita menemukan kriteria kesalahan presentase untuk sistem, kita boleh menetapkannya dari tujuan desain sistem dalam pertanyaan.

$$t_s(\pm 3\%) \approx 5t \tag{2.20}$$

$$t_s(\pm 4\%) \approx 2t \tag{2.21}$$

$$t_s(\pm 5\%) \approx 0.5t \tag{2.22}$$



Gambar 2.13 Penggolongan respon transien dan respon steady state