

SISTEM KENDALI MOTOR MENGGUNAKAN KONTROL PID

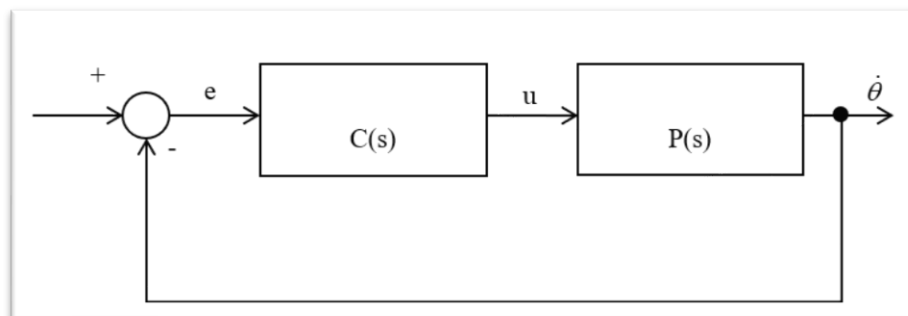
A. Persamaan

Untuk kecepatan motor menggunakan sebuah persamaan, yakni :

$$P(s) = \frac{\dot{\theta}(s)}{V(s)} = \frac{K}{(Js + b)(Ls + R) + K^2} \left[\frac{rad/sec}{V} \right]$$

Gambar 1. Persamaan sistem

Kecepatan motor ini menggunakan system tertutup, artinya adalah akan ada umpan balik yang dikirimkan kembali kepada system. Berikut ini adalah diagramnya



Gambar 2 Struktur sistem kontrol

Struktur diatas, diimplementasikan ke dalam Octave, dengan sourcecode sebagai berikut :

```
j = 0.01;
b = 0.1;
k = 0.01;
r = 1;
l = 0.5;
s = tf('s');
p_motor = k/((j*s+b)*(l*s+r)+k^2);
```

Muslim

21120117140032

Robotika A

Digunakan fungsi kontrol PID sebagai berikut :

$$C(s) = K_p + \frac{K_i}{s} + K_d s = \frac{K_d s^2 + K_p s + K_i}{s}$$

Gambar 3. Fungsi control PID yang digunakan

Untuk mencari kecepatan motor didalam Octave digunakan program sebagai berikut :

```
#Motor speed PID control
#source:http://ctms.engin.umich.edu/CTMS/index.php?example=MotorSpeed&section=ControlPID

pkg load control;

kp = input("Enter Kp value: ");
ki = input("Enter Ki value: ");
kd = input("Enter Kd value: ");

j = 0.01;
b = 0.1;
k = 0.01;
r = 1;
l = 0.5;
s = tf('s');
p_motor = k/((j*s+b)*(l*s+r)+k^2);

c = pid(kp,ki,kd);
sys_cl = feedback(c*p_motor,1);

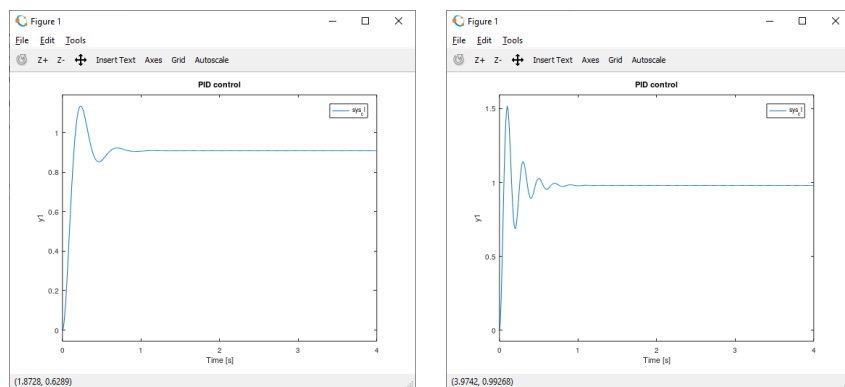
t = 0:0.01:4;
```

```
step(sys_cl,t);
grid;
title('PID control');
```

B. Tuning dan Simulasi

Untuk mengetahui tuning pada sistem, dilakukan simulasi terhadap nilai-nilai percobaan untuk kontrol PID. Hasil simulasi akan menunjukkan nilai kontrol PID optimal untuk sistem. Berikut dilakukan trial n error nilai kontrol PID terhadap unit $\text{step} = 1$.

Tuning Kp atau proporsional :



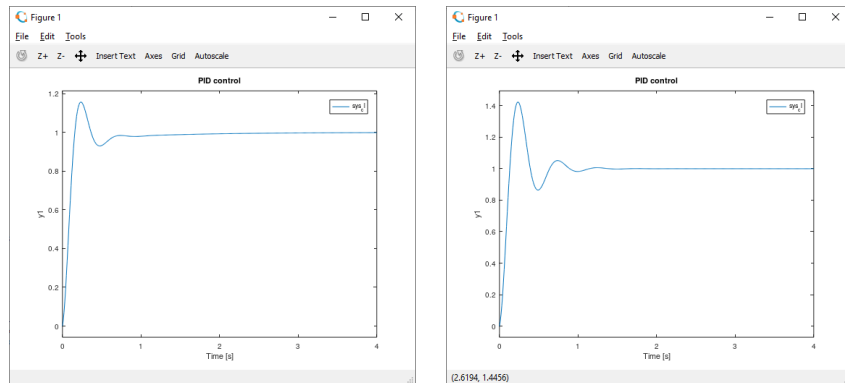
Gambar 4 Simulasi nilai kontrol proporsional tanap kontrol I dan D; (kiri) $K_p = 100$ (kanan) $K_p = 500$

Dari gambar di atas, terlihat bahwa steady state error cukup signifikan. Untuk menyelesaikan masalah ini, nilai K_p dapat diperbesar. Namun, memperbesar nilai K_p akan memperbesar overshoot dan settling time. Untuk menghindari hal tersebut digunakan kontrol Integral. K_i atau Kontrol Integral digunakan untuk memperkecil nilai steady state error. Nilai K_i yang terlalu besar juga dapat meningkatkan overshoot.

Muslim

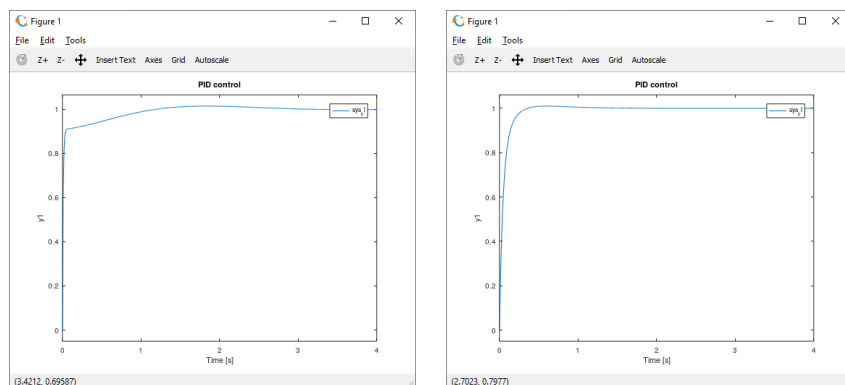
21120117140032

Robotika A



Gambar 5 Kontrol nilai K_i dengan nilai $K_p = 100$ dan nilai $K_d = 1$; (kiri) $K_i = 200$ (kanan) $K_i = 500$

Pada gambar di atas, dapat terlihat bahwa nilai $K_i 500$ menyebabkan peningkatan nilai overshoot. Berarti untuk sistem ini nilai $K_i = 200$ adalah nilai optimal. Overshoot dapat dikurangi dengan menggunakan K_d atau kontrol Derivative. Nilai K_d yang berlebihan juga dapat menyebabkan peningkatan settling time.



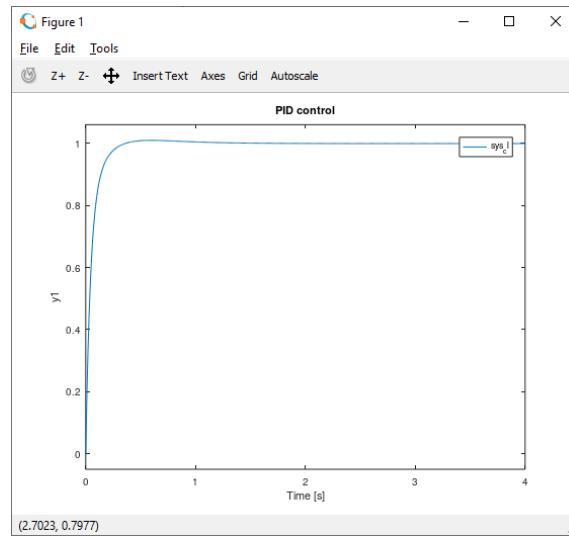
Gambar 6 Kontrol Derivative dengan nilai $K_p = 100$ dan $K_i = 200$; (kiri) $K_d = 50$ (kanan) $K_d = 10$

Berdasarkan hasil simulasi trial n error di atas, dapat diambil kesimpulan bahwa nilai kontrol PID optimal untuk sistem adalah: **$K_p = 100$, $K_i = 200$, dan $K_d = 10$.**

Muslim

21120117140032

Robotika A



Gambar 7 Response sistem dengan nilai kontrol PID optimal