

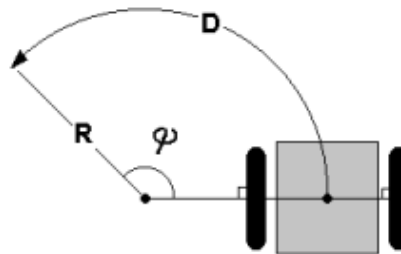
Simulasi pergerakan Mobile Robot saat berbelok dengan GNU Octave

Nama: MUSLIM

NIM: 21120117140032

Kelas: Robotika A

Untuk mencari kecepatan sudut roda robot saat berbelok menggunakan ilustrasi sebagai berikut



Gambar 1. 1 ilustrasi

Untuk mencari kecepatan, menggunakan rumus berikut :

$$R = \frac{L}{2} \frac{V_R + V_L}{V_R - V_L} = \frac{L}{2} \frac{W_R + W_L}{W_R - W_L}$$

$$D = \int \frac{V_L + V_R}{2} dt = \frac{1}{2} r \frac{W_L + W_R}{2} (t_3 - t_0 + t_2 - t_1)$$

$$\varphi = \frac{D}{R} = \frac{r}{2L} (W_R - W_L) (t_3 - t_0 + t_2 - t_1)$$

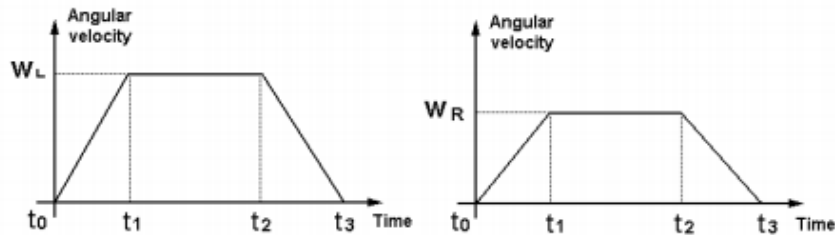
R : Jari2 rotasi

D : Panjang Jalur

φ : Sudut rotasi

Gambar 1. 2 Rumus

Dari pemaparan tadi, kode Octave dibuat untuk simulasi robot berbelok, dengan kecepatan roda kanan dan kiri yang berbeda. Tergantung dari sudut belok, lebar robot dan perpindahan kecepatan robot, maka selisih dari kecepatan sudut roda robot akan berbeda pula. Keluaran dari kode Octave adalah selisih kecepatan sudut roda dan ilustrasi grafik kecepatan sudut roda seperti contoh pada materi yaitu



Gambar 1. 3 Ilustrasi kecepatan

Berikut ini adalah Source code “motion_control.m”

```
pkg load symbolic

function ang_vel = getAngVel (lin_vel, radius, theta)
    ang_vel = abs(lin_vel) * sin(theta) / radius;
endfunction

function ang_vel_diff = getAngVelDiff (mb_len, radius, theta, t0, t1, t2, t3)
    ang_vel_diff = 2 * mb_len / radius * theta * (t3 - t0 + t2 - t1);
endfunction

function save_plot(ang_vel, t0, t1, t2, t3, fname)
    x_coords = [t0, t1, t2, t3];
    y_coords = [0, ang_vel, ang_vel, 0];
    plot(x_coords, y_coords);
    fname = strcat("other/", fname, ".jpg");
    print(fname, "-djpg");
endfunction

function main(verbose=false)
    if(verbose==true)
        disp("\nAll units are on meters, seconds, and radians");
    endif

    primary_ang_vel = 600;

    L = 0.2;
    r = 2;
    th = 135;
    t0 = 0; t1 = 0.1; t2 = 0.5; t3 = 0.6;

    ang_vel_diff = getAngVelDiff(L, r, th, t0, t1, t2, t3);
    secondary_ang_vel = primary_ang_vel + ang_vel_diff;

    disp(["\nPrimary wheel angular velocity (rad/s): ",
num2str(primary_ang_vel)]);
    disp(["Angular velocity difference (rad/s): ", num2str(ang_vel_diff)]);
```

```

    disp(["Secondary wheel angular velocity (rad/s): ",
num2str(secondary_ang_vel)]);

    disp(["\nGenerating primary wheel plot..."]);
    save_plot(primary_ang_vel, t0, t1, t2, t3, "primary_wheel");
    disp(["Generating secondary wheel plot..."]);
    save_plot(secondary_ang_vel, t0, t1, t2, t3, "secondary_wheel");

endfunction

main();

```

Pada source code dapat dilihat ada empat fungsi. Fungsi pertama `getAngVel (lin_vel, radius, theta)` untuk menghitung kecepatan sudut, lalu `getAngVelDiff (mb_len, radius, theta, t0, t1, t2, t3)` untuk mencari selisih kecepatan sudut roda kiri dan kanan, lalu `save_plot(ang_vel, t0, t1, t2, t3, fname)` untuk membuat gambar grafik dari simulasi, lalu `main(verbose=false)` untuk dieksekusi pertama kali saat program berjalan.

Dapat dilihat ada beberapa variabel sebagai masukan didalam fungsi `main()` yaitu masukan untuk kecepatan sudut roda utama, L , r , θ , dan t_0 s.d t_4 sesuai parameter yang diperlukan pada rumus di materi. Kecepatan roda utama maksudnya adalah kecepatan roda yang berjalan lebih lambat saat berbelok. Contohnya jika robot berbelok kearah kanan maka kecepatan roda utama diambil dari roda kanan, lebih tepatnya kecepatan roda yang lebih dekat ke titik pusat belok.

Output dari program adalah selisih kecepatan sudut roda dan ilustrasi grafik kecepatan sudut roda utama dan sekunder.

```
hitungan-stuff other motion_controlM
Project
  hitungan-stuff ~/Codespace/P...
    data
    models
    notebooks
    other
      motion_controlM
      primary_wheel.jpg
      secondary_wheel.jpg
      main.py
      README.md
      requirements.txt
    External Libraries
    Scratches and Consoles
main.py x motion_control.m x
26 L = 0.2;
27 r = 2;
28 th = 135;
29 t0 = 0; t1 = 0.1; t2 = 0.5; t3 = 0.6;
30
31 ang_vel_diff = getAngVelDiff(L, r, th, t0, t1, t2, t3);
32 secondary_ang_vel = primary_ang_vel + ang_vel_diff;
33
34 disp(["\nPrimary wheel angular velocity (rad/s): ", num2str(primary_ang_vel)]);
35 disp(["Angular velocity difference (rad/s): ", num2str(ang_vel_diff)]);
36 disp(["Secondary wheel angular velocity (rad/s): ", num2str(secondary_ang_vel)]);
37
38 disp(["\nGenerating primary wheel plot..."]);
39 save_plot(primary_ang_vel, t0, t1, t2, t3, "primary_wheel");
40 disp(["Generating secondary wheel plot..."]);
41 save_plot(secondary_ang_vel, t0, t1, t2, t3, "secondary_wheel");
42
43 endfunction
44
45 main();
```

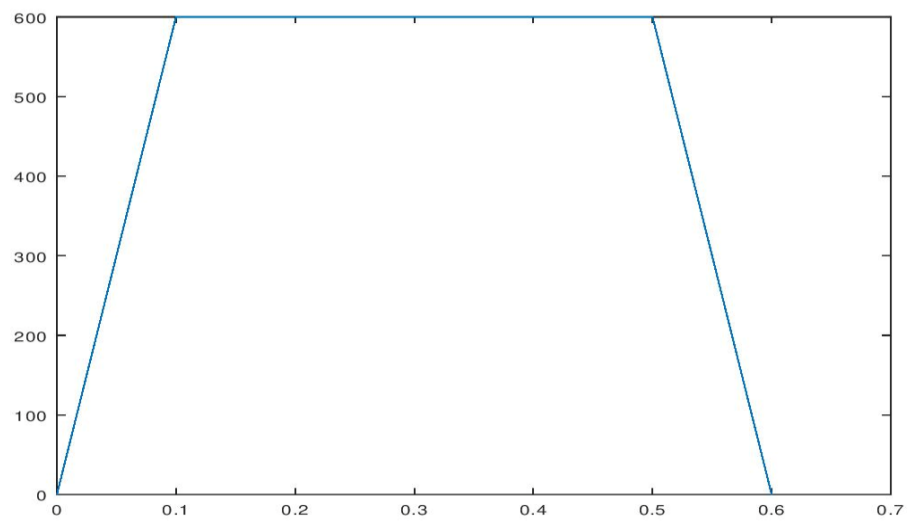
Run: motion_control.m x

```
Primary wheel angular velocity (rad/s): 600
Angular velocity difference (rad/s): 27
Secondary wheel angular velocity (rad/s): 627

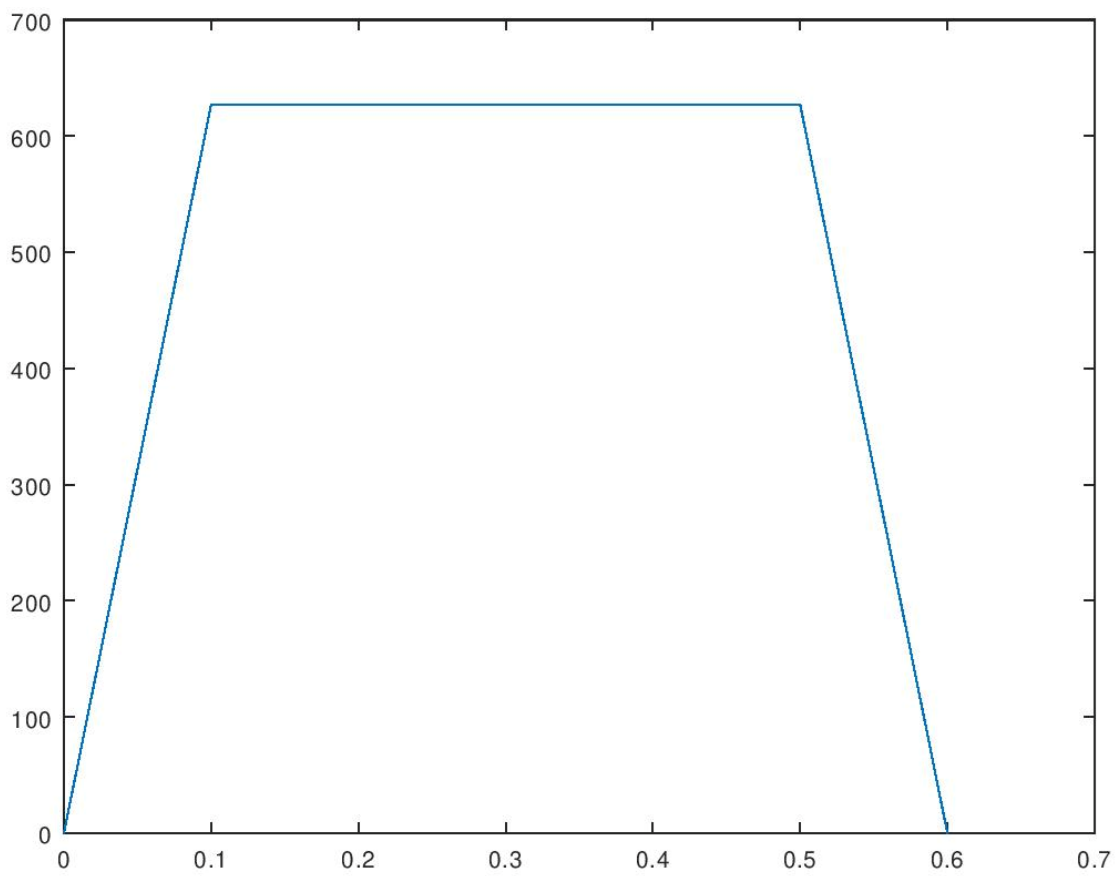
Generating primary wheel plot...
Generating secondary wheel plot...

Process finished with exit code 0
```

Gambar 1. 5 output



Gambar 1. 4 output



Gambar 1. 6 output