



**Tecnológico
de Monterrey**

**Instituto Tecnológico y de
Estudios Superiores de
Monterrey**

TE3002B.502

Implementación de robótica Inteligente (Gpo 502)

Semestre: febrero - junio 2023

Evaluación 7.1 (Trayectorias en lazo abierto)

Alumno:

Fredy Yahir Canseco Santos

A01735589

Profesor: Dr. Alfredo García Suárez

Fecha de entrega: 16 de Mayo del 2023

Responder las siguientes preguntas en base al procedimiento empleado para obtener las trayectorias propuestas:

a) ¿Cuál fué el o los parámetros que se modifican para obtener una trayectoria recta? El parámetro que se modifica para una trayectoria recta es el vector “V” ¿Porqué? Esto porque el vector “V” se encarga de la velocidad lineal, para una trayectoria lineal tendríamos que multiplicar por 1 el vector “V”.

b) ¿Cuál fué el o los parámetros que se modifican para obtener una trayectoria curva? El parámetro que se modifica es el vector “W” ¿Porqué? Esto porque el vector “W” se encarga de la velocidad angular, para una trayectoria angular tendríamos que multiplicar por un ángulo el vector “W”.

c) ¿Cuál fué el o los parámetros que se modifican para obtener un giro? Los parámetros que se modifican para una trayectoria recta son ambos vectores “V” y “W” ¿Porqué? Esto porque el vector “V” se encarga de la velocidad lineal y el vector “W” se encarga de la velocidad angular, para una trayectoria que sea un giro tendríamos que multiplicar por un ángulo deseado el vector “W” y por 1 el vector “V”.

d) ¿Qué papel desempeña el vector del tiempo en la generación de la trayectoria? El vector de tiempo nos sirve para poder simular el movimiento de robot, con base en el tiempo es que el robot podrá moverse, esto debido a que se toman las muestras necesarias con base en el mismo.

e) ¿Cuáles fueron los parámetros que se ajustaron para obtener las dimensiones de las trayectorias deseadas?

Se ajustó el tiempo de simulación, el vector de tiempo y el vector de velocidad lineal y angular.

Código Implementado

```
%Limpieza de pantalla
clear all
close all
clc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% TIEMPO %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tf=72;           % Tiempo de simulación en segundos (s)
ts=0.1;          % Tiempo de muestreo en segundos (s)
t=0:ts:tf;       % Vector de tiempo
N= length(t);    % Muestras

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CONDICIONES INICIALES %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Inicializamos las variables que se van a emplear
x1= zeros (1, N+1); % Posición (X) en el centro del eje que une las ruedas en metros (m)
y1= zeros (1, N+1); % Posición (Y) en el centro del eje que une las ruedas en metros (m)
phi= zeros (1, N+1); % Orientación del robot en radianes (rad)
%Damos valores a nuestro punto inicial de posición y orientación
x1(1)=0; %Posición inicial eje x
y1(1)=0; %Posición inicial eje y
phi(1)=0; %Orientación inicial del robot

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PUNTO DE CONTROL %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Inicializamos el punto de control
hx= zeros (1, N+1); % Posición en el eje (X) del punto de control en metros (m)
hy= zeros (1, N+1); % Posición en el eje (Y) del punto de control en metros (m)
%Igualamos el punto de control con las proyecciones X1 y Y1 por su
%coincidencia
hx(1)= x1(1); % Posición del punto de control en el eje (X) metros (m)
hy(1)= y1(1); % Posición del punto de control en el eje (Y) metros (m)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% VELOCIDADES DE REFERENCIA %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
v = [1*ones(1,N)]; % Velocidad lineal de referencia (m/s)
w = [0*ones(1,N)]; % Velocidad angular de referencia (rad/s)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% LETRA F %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
v(1:10) = 0;
w(1:10) = deg2rad(90);

v(41:50) = 0;
w(41:50) = deg2rad(-90);
```

```

v(61:70) = 0;
w(61:70) = deg2rad(180);

v(81:90) = 0;
w(81:90) = deg2rad(90);

v(111:120) = 0;
w(111:120) = deg2rad(90);

v(131:140) = 0;
w(131:140) = deg2rad(-180);

v(151:160) = 0;
w(151:160) = deg2rad(90);

v(171:180) = 0;
w(171:180) = deg2rad(90);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% LETRA R

```

```

v(201:210) = 0;
w(201:210) = deg2rad(90);

v(241:250) = 0;
w(241:250) = deg2rad(-90);

v(261:270) = 0;
w(261:270) = deg2rad(-90);

v(281:290) = 0;
w(281:290) = deg2rad(-90);

v(300:310) = 0;
w(300:310) = deg2rad(100);

v(331:340) = 0;
w(331:340) = deg2rad(70);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% LETRA E

```

```

v(351:360) = 0;
w(351:360) = deg2rad(90);

v(391:400) = 0;
w(391:400) = deg2rad(-90);

v(421:430) = 0;
w(421:430) = deg2rad(180);

v(451:460) = 0;
w(451:460) = deg2rad(90);

v(481:490) = 0;
w(481:490) = deg2rad(90);

v(511:520) = 0;
w(511:520) = deg2rad(180);

```

```

v(541:550) = 0;
w(541:550) = deg2rad(90);

v(561:570) = 0;
w(561:570) = deg2rad(90);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% LETRA D

v(601:610) = 0;
w(601:610) = deg2rad(90);

v(641:650) = 0;
w(641:650) = deg2rad(-125);

v(671:680) = 0;
w(671:680) = deg2rad(-55);

v(690:700) = 0;
w(691:700) = deg2rad(-60);

for k=1:N

    %Aplico la integral a la velocidad angular para obtener el angulo "phi" de la orientación
    phi(k+1)=phi(k)+w(k)*ts; % Integral numérica (método de Euler)

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% MODELO CINEMATICO %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    xp1=v(k)*cos(phi(k));
    yp1=v(k)*sin(phi(k));

    %Aplico la integral a la velocidad lineal para obtener las cordenadas
    %"x1" y "y1" de la posición
    x1(k+1)=x1(k)+ ts*xp1; % Integral numérica (método de Euler)
    y1(k+1)=y1(k)+ ts*yp1; % Integral numérica (método de Euler)

    % Posicion del robot con respecto al punto de control
    hx(k+1)=x1(k+1);
    hy(k+1)=y1(k+1);

end

scene=figure; % Crear figura (Escena)
set(scene,'Color','white'); % Color del fondo de la escena
set(gca,'FontWeight','bold'); % Negrilla en los ejes y etiquetas
sizeScreen=get(0,'ScreenSize'); % Retorna el tamaño de la pantalla del computador
set(scene,'position',sizeScreen); % Configurar tamaño de la figura
camlight('headlight'); % Luz para la escena
axis equal; % Establece la relación de aspecto para que las unidades de datos sean las mismas en todas las direcciones.
grid on; % Mostrar líneas de cuadrícula en los ejes
box on; % Mostrar contorno de ejes
xlabel('x(m)'); ylabel('y(m)'); zlabel('z(m)'); % Etiqueta de los eje

view([-0.1 35]); % Orientacion de la figura
axis([-2 12 -1 6 0 1]); % Ingresar limites minimos y maximos en los ejes x y z [minX maxX minY maxY minZ maxZ]

% b) Graficar robots en la posicion inicial
scale = 4;
MobileRobot;
H1=MobilePlot(x1(1),y1(1),phi(1),scale);hold on;

```

```

% c) Graficar Trayectorias
H2=plot3(hx(1),hy(1),0,'g','linewidth',2);

% d) Bucle de simulacion de movimiento del robot

step=15; % pasos para simulacion

for k=1:step:N

    delete(H1);
    delete(H2);

    H1=MobilePlot(x1(k),y1(k),phi(k),scale);
    H2=plot3(hx(1:k),hy(1:k),zeros(1,k),'b','linewidth',2);

    pause(ts);

end

```

Trayectoria Resultante

