

Universidad EIA

Ingeniería Mecatrónica

Robótica Industrial

Actividad de Seguimiento No. 12 y 13

Simulador Vehículo Ruedas Odometría (Simulink)

Profesor: David Rozo Osorio

Arthur Jean Antoine De La Lande

Guillermo Domínguez Sánchez

Oliver Osorio Agudelo

Rodolfo Tonatiuh Silva Martínez

A partir del robot que se muestra en la figura 1 se debe construir un Simulador de Odometría y Control de Posición para un Vehículo con Ruedas en Simulink.

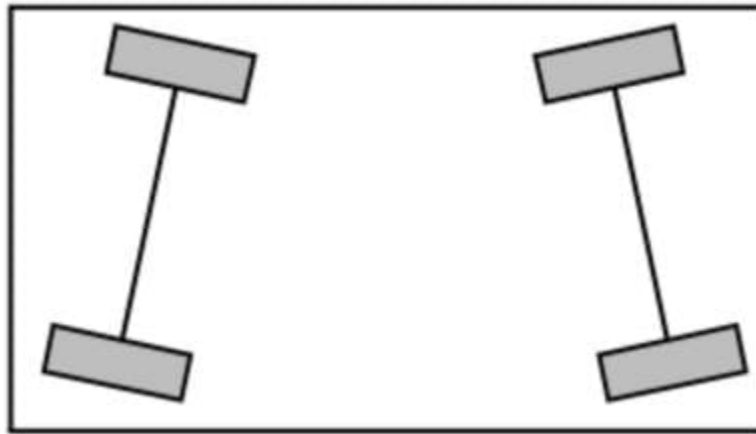
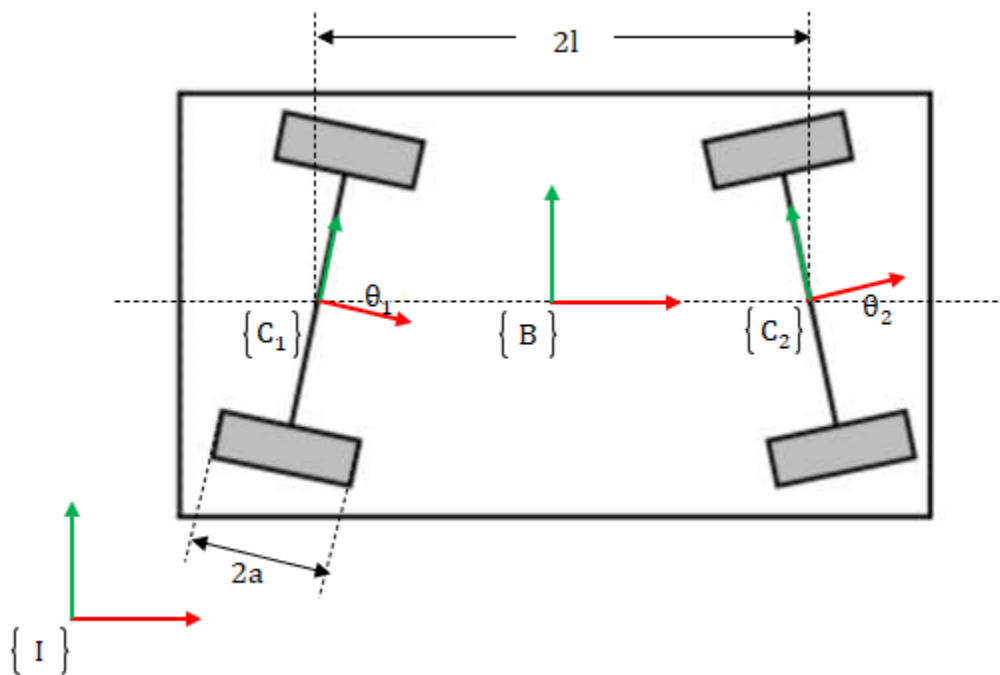


Figura 1

Iniciamos con el modelo cinemático, planteando los sistemas de referencia Inercial, del Body y de las ruedas del robot:



Dado que las ruedas de tracción tanto delanteras como traseras se encuentran unidas y se mueven en conjunto, las podemos suponer como una sola rueda en un punto intermedio equidistante C_1 y C_2 .

Estableciendo los parámetros de cada rueda:

$$\omega_1: a_1 = a \quad \phi_1 = 0^\circ \quad \theta_1 = \theta_1 \quad dx_1 = -l \quad dy_1 = 0$$

$$\omega_2: a_2 = a \quad \phi_2 = 0^\circ \quad \theta_2 = \theta_2 \quad dx_2 = l \quad dy_2 = 0$$

Con estos datos podemos sustituir en la ecuación (15):

$$\omega_i = \begin{bmatrix} \frac{1}{a_i} & \frac{1}{a_i} \tan(\phi_i) \end{bmatrix} \begin{bmatrix} \cos(\theta_{Bi}) & \sin(\theta_{Bi}) \\ -\sin(\theta_{Bi}) & \cos(\theta_{Bi}) \end{bmatrix} \begin{bmatrix} 1 & 0 & -d_{yi} \\ 0 & 1 & d_{xi} \end{bmatrix} \begin{bmatrix} u(t) \\ v(t) \\ r(t) \end{bmatrix} \quad (15)$$

Con ayuda de la función `fn_wi` que creamos previamente en Matlab podemos obtener de forma simbólica ω_1 y ω_2 , agrupando los valores en la matriz Q :

```
Q =  
  
[cos(th1)/a, sin(th1)/a, -(l*sin(th1))/a]  
[cos(th2)/a, sin(th2)/a, (l*sin(th2))/a]
```

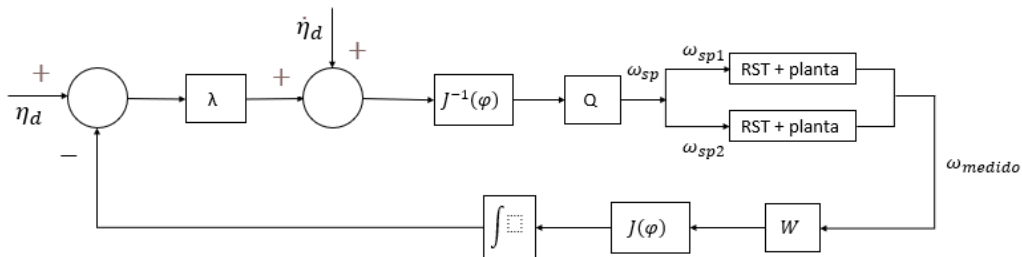
Ahora que tenemos la matriz Q procedemos a calcular su inversa. Debido a que es una matriz 2×3 recurrimos al tercer caso:

$$m < n$$

$$W = Q^T (Q Q^T)^{-1}$$

Con ayuda de Matlab y el `if` planteado en el seguimiento anterior obtenemos una matriz W simbólica muy larga.

A partir de la obtención de estas matrices podemos implementarlas en el diagrama de control planteado:



De esta forma pasamos a Simulink.

Antes de implementarlo tal cual en Simulink, tenemos que reflexionar en la estrategia de control, porque el sistema no es olonómico. Dividimos el control de posición y orientación en 2 fases porque no es olonómico y no nos permite controlar 3 variables de una.

Fase 1: traslación del robot hacia la posición (x_d, y_d) por medio del control de velocidad y de orientación de las llantas.

Restricción: para simplificar, se supone que las 4 llantas tienen misma orientación θ y misma velocidad w en esta primera fase. Además, se tiene que realizar el control de giro de las llantas. Se supone entonces que el modelo del actuador (tipo servo por ejemplo) sigue perfectamente la acción de control, entonces para simular este control de posición angular le agregamos un control proporcional con una pequeña ganancia.

La lógica es la siguiente:

- Se mide el error de θ
- Se corrige θ
- Se cambian las matrices Q y W con este mismo θ
- Se cambian las velocidades de las llantas con respecto al setpoint.

Fase 2: Una vez llegado hasta la posición adecuada (con un cierto radio de tolerancia), nos enfocamos sobre el control de orientación para alcanzar el setpoint de ángulo del body.