

# Robot móvil omnidireccional de 4 ruedas, para la detección de distancia mínima en colas de espera

Jhayro Chávez. Juan Alva y Renzo Meza

Universidad Ricardo Palma  
Lima, Perú

jhayro.chavez@urp.edu.pe

juan.alvam@urp.edu.pe

renzo.meza@urp.edu.pe

**Abstract—** This work includes the calculations carried out to obtain a kinematic model of an omnidirectional 4-wheel robot and its programming in Matlab, the mechanical design of the robot and the wheels type mecanum in the Inventor software, and the simulation in CoppeliaSim, previously called V-REP, since it is a common virtual environment for testing engineered robots.

**Keywords -** kinematic model, omnidirectional 4-wheel robot

## I. INTRODUCCIÓN

La pandemia trajo varias dificultades para la población, pero con las vacunas ya disponibles, las personas no suelen cuidarse en las colas, en varios casos no se respeta la distancia mínima, teniendo en cuenta que la COVID-19 se propaga principalmente entre personas que están en contacto cercano (dentro de 6 pies aproximadamente) por un período prolongado.

## II. DEFINICIÓN DEL PROBLEMA

Las vacunas del COVID 19, no significan invulnerabilidad ante el virus, aunque la tasa de infección tras la vacuna es mínima, se sigue recomendando las medidas de seguridad, como son la mascarilla y la distancia mínima. Según lo comentado, nuestra propuesta abarca un robot móvil omnidireccional de 4 ruedas, como sistema de locomoción, además de un sistema que detecte si la distancia entre las personas puede significar un riesgo para su salud, emitiendo una alarma.

## III. DESARROLLO

Existen varios tipos de robots móviles terrestres, se suelen clasificar según su tipo de movimiento. Entre los que usan ruedas los 2 más comunes son el unicyclo y el omnidireccional, además de ello otra característica es el tipo de rueda escogido para el proyecto, el cual será “mecanum”.

### A. Cálculos

Por su capacidad de desplazarse en cualquier dirección sin la necesidad de alcanzar previamente una orientación específica en toda clase de entornos superficiales, se realizaron los cálculos del modelo de cinemática del robot omnidireccional de

4 ruedas. La característica de este tipo de robot, es que a diferencia de la mayoría de móviles, cuenta con 2 velocidades, comúnmente llamadas frontal y lateral, pues su tipo de llanta peculiar le permite hacer giros sin necesidad de orientarse previamente.

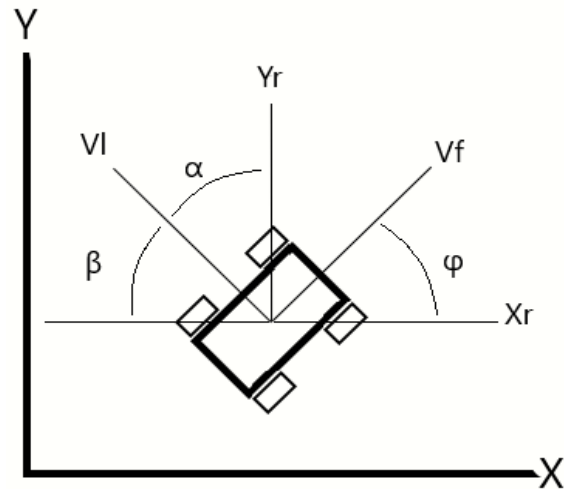


Figura 1: coordenadas del robot en el plano XY

$$X_r = X_c$$

$$Y_r = Y_c$$

$$X_{r'} = X_{c'}$$

$$Y_{r'} = Y_{c'}$$

$$X_{c'} = V_f \cdot \cos(\varphi) - V_l \cdot \sin(\alpha)$$

$$Y_{c'} = V_f \cdot \sin(\varphi) + V_l \cdot \cos(\alpha)$$

$$\beta + \alpha = 90^\circ$$

$$\varphi + 90 + \beta = 180^\circ$$

$$\beta = 90^\circ - \varphi$$

$$\alpha = \varphi$$

Modelo cinemático

$$Xr' = Vf \cdot \cos(\varphi) - Vl \cdot \sin(\varphi)$$

$$Yr' = Vf \cdot \sin(\varphi) + Vl \cdot \cos(\varphi)$$

$$\varphi' = \omega$$

Modelo matricial

$$\begin{bmatrix} Xr' \\ Yr' \\ \varphi' \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Vf \\ Vl \\ \omega \end{bmatrix}$$

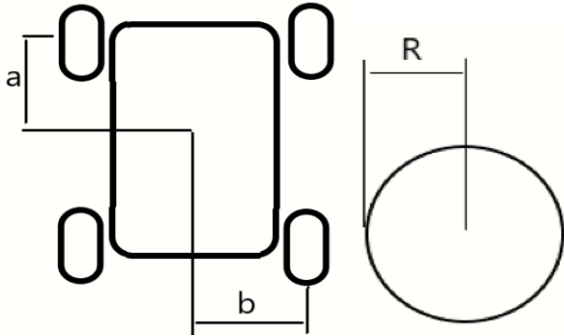


Figura 2: dimensiones de las ruedas del robot.

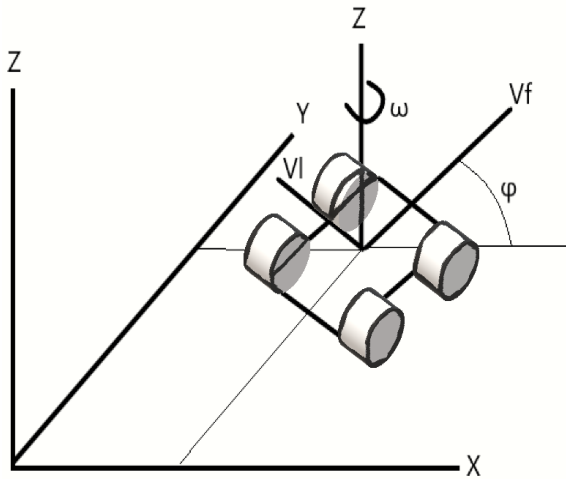


Figura 3: relación de ruedas y velocidades.

$$\begin{bmatrix} Wd1 \\ Wd2 \\ Wd3 \\ Wd4 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & -1 & -(a+b) \\ 1 & 1 & (a+b) \\ 1 & 1 & -(a+b) \\ 1 & -1 & (a+b) \end{bmatrix} \begin{bmatrix} Vf \\ Vl \\ \omega \end{bmatrix}$$

### B. Cinemática

Los cálculos obtenidos a través del modelo cinemático son necesarios para poder controlar el robot. Según lo visto anteriormente se puede mover un robot, dando velocidades

lineales y velocidad angular. Para probar el desenvolvimiento del modelo, se utilizó el software Matlab, por su capacidad de procesar cálculos matriciales.

% Cinemática de un robot móvil omnidireccional

clc; clear all; close all;

% Tiempo de simulación

ts=0.01;

t=0:ts:10;

% Velocidades

vf=-0.4\*ones(1,length(t)); % Velocidad frontal

vl=0.5\*ones(1,length(t)); % Velocidad lateral

w=0.4\*ones(1,length(t)); % Velocidad angular

% Posición de llantas respecto al centro

a=0.15;

b=0.10;

% Radio de la llanta

R=0.02;

T=(1/R)\*[1 -1 -(a+b);...

1 1 (a+b);...

1 1 -(a+b);...

1 -1 (a+b)];

% Posición inicial

xr(1)=0;

yr(1)=0;

phi(1)=0;

for k=1:length(t)

% Movimiento del robot

xrp(k)=vf(k)\*cos(phi(k))-vl(k)\*sin(phi(k));

yrp(k)=vf(k)\*sin(phi(k))+vl(k)\*cos(phi(k));

Wd=T\*([vf(k);vl(k);w(k)]);

Wd1(k) = Wd(1);

```

Wd2(k) = Wd(2);
Wd3(k) = Wd(3);
Wd4(k) = Wd(4);

% Integración numérica
xr(k+1) = xr(k)+ts*xrp(k);
yr(k+1) = yr(k)+ts*yrp(k);
phi(k+1) = phi(k)+ts*w(k);
end

```

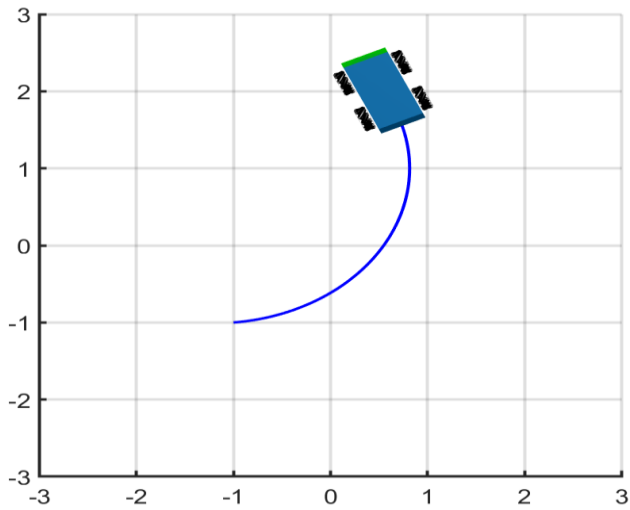


Figura 4: trayectoria realizada por el robot.

*C. Diseño*

Se realizó el diseño mecánico del robot y las ruedas mecanum, en Inventor. Lo cual posteriormente se exportó en stl, para hacer uso de ello en CoppeliaSim y poder simular el robot diseñado.

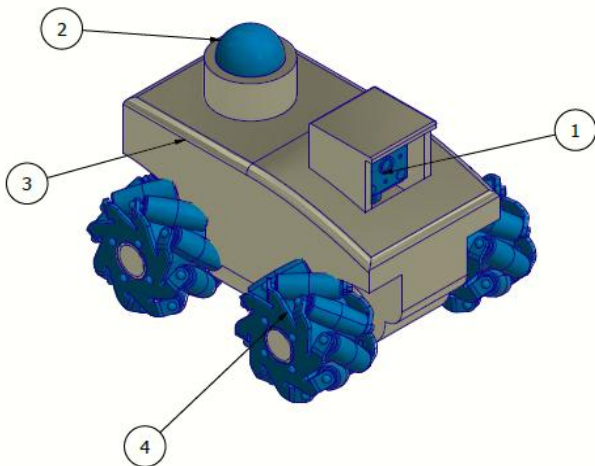


Figura 5: Sistema de visión y ultrasonido (1), Sensor PIR (2), Estructura (3) y Rueda mecanum (4)

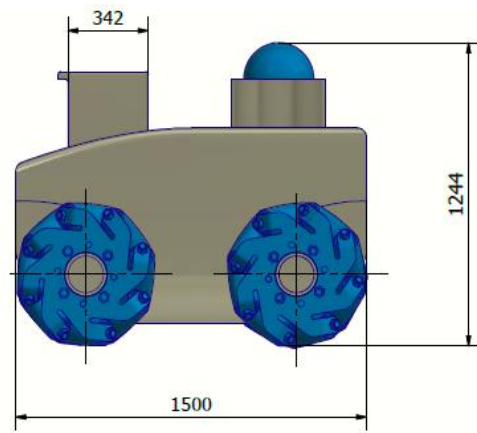


Figura 6: vista de perfil

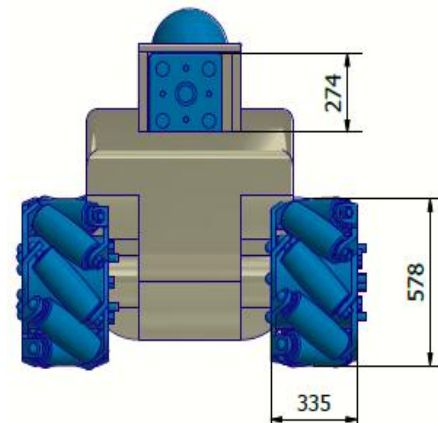


Figura 7: vista frontal

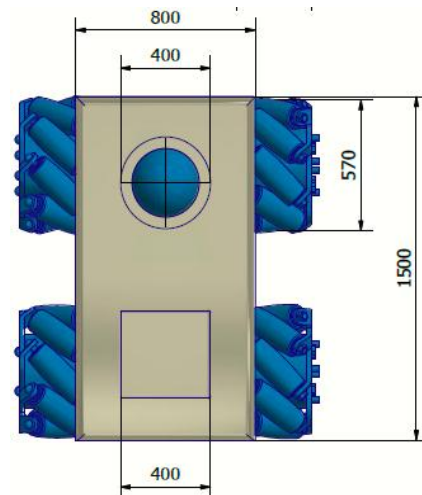


Figura 8: vista superior

**IV. RESULTADOS**

Este robot móvil no logró implementarse, pero con la ayuda del programa Coppelia Sim, se pudo observar su comportamiento dinámico. Los resultados obtenidos en la simulación fueron los deseados debido a que el programa no cuenta con una forma de simular las ruedas mecanum. Esto se

solucionó usando una forma esférica y con el eje girado 45°, con ello pudimos obtener un comportamiento más realista de las ruedas mecanum

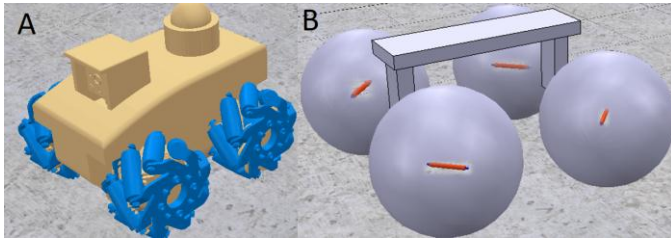


Figura 9: En la figura 'A' está el robot en Coppelia Sim y en la 'B' la forma cinemática del comportamiento del robot.

Los resultados fueron los deseados, se logró detectar la distancia entre personas y si no se respeta el distanciamiento social se activa la alarma.

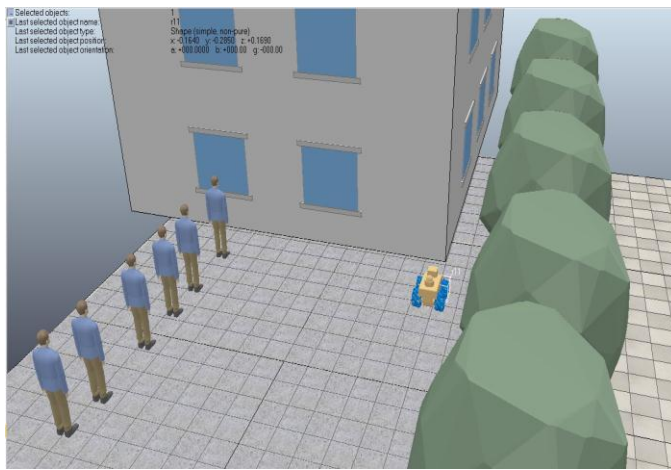


Figura 10: Simulación del funcionamiento del robot en un entorno creado en Coppelia Sim

## V. CONCLUSIONES

Para poder apreciar el funcionamiento de este robot, se optó por usar el Software Coppelia Sim, el cual nos brinda la facilidad de tener una simulación más realista al tener la opción de extraer un sólido (cuadrado, cilindro o esfera) de la pieza seleccionada, con lo cuales podemos definir su peso y es más fácil saber dónde se encuentra su centro de masa.

## REFERENCIAS

- [1] A. Saénz, V. Santibáñez y E. Bugarin, *Control de velocidad de un robot omnidireccional con dinámica de actuadores*, AMRob Journal, Robotics: Theory and Applications, 2016
- [2] R. S. Santiago, *Control y Comportamiento de Robots Omnidireccionales*, Universidad de la República Montevideo - Uruguay, 2009.
- [3] M. I. Aguilera, M. A. bautista y J. Iruegas, *Diseño y Control de Robots Móviles*, Instituto Tecnológico de Nuevo Laredo Reforma Sur 2007
- [4] Silva Ortigoza, R.; García Sánchez, J. R.; Barrientos Sotelo, V. R.; Molina Vilchis, M. A.; Hernández Guzmán, V. M.; Silva Ortigoza, G. Una panorámica de los robots móviles Télématique, vol. 6, núm. 3,

2007, pp. 1-14 Universidad Privada Dr. Rafael Beloso Chacín Zulia, Venezuela

- [5] Barrientos Sotelo, Víctor Ricardo; García Sánchez, José Rafael; Silva Ortigoza, Ramón Robots Móviles: Evolución y Estado del Arte Polibits, núm. 35, 2007, pp. 12-17 Instituto Politécnico Nacional Distrito Federal, México