



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECATRÔNICA



**1º RELATÓRIO: Simulação cinemática de um robô móvel com
acionamento diferencial**
CoppeliaSim 4.1.0 (V-REP)

Iago Lucas Batista Galvão
Jhonat Heberston Avelino de Souza
Thiago de Araújo Brito

1º Relatório: Simulação cinemática de um robô móvel com acionamento
diferencial

CoppeliaSim 4.1.0 (V-REP)

Primeira entrega do 1º Relatório da disciplina EGM0007
- Sistemas Robóticos Autônomos do Mestrado em
Engenharia Mecatrônica da Universidade Federal do Rio
Grande do Norte (UFRN), referente a nota parcial da
segunda unidade.

Professor(a): Pablo Javier Alsina.

Sumário

	Páginas
1 INTRODUÇÃO	4
2 OBJETIVO	5
3 METODOLOGIA	5
4 SIMULAÇÃO DO MODELO CINEMÁTICO	6
4.1 SIMULAÇÃO	7
4.2 RESULTADOS	10
5 CONCLUSÃO	14
6 REFERÊNCIAS	14

1 INTRODUÇÃO

Com o aumento do poder computacional, houve um crescimento no número de pesquisas na área de robótica e, atualmente, o pesquisador é capaz de realizar simulações com qualidade em cenários 2D/3D. Além disso, os simuladores possibilitam a realização de experimentos, sem a necessidade de construir o *hardware* do robô para desempenhar vários tipos de testes. Como tal desenvolvimento custa caro, esta alternativa possibilita a pesquisa em robôs mais viável e difundida.

Existem muitos *softwares* de simulação robótica disponíveis como, por exemplo, Open HPR, Gazebo, Webots, V-REP etc (1). A plataforma utilizada nesse estudo foi a *Virtual Robot Experimentation Platform* (V-REP), pois atende muitos requisitos, como estrutura de simulação versátil e escalável, arquitetura de controle distribuída, controle por *Script*, *Plugins*, ou API cliente remota entre outras funcionalidades (1), ademais, esta é uma das mais empregadas nos estudos robóticos de todo o mundo.

A versatilidade do simulador contribui para diferentes plataformas robóticas e várias aplicações, não atentando-se somente ao fato de simular robôs com resultados iniciais. Outrossim, isenta o gasto e a necessidade da aquisição do *hardware* para resultados mais realísticos e adaptáveis.

No experimento discutido, foi criado um modelo simples de robô móvel no formato retangular ($20 \times 40 \times 10$ cm), com três rodas, sendo estas: duas rodas traseiras tracionadas com $12,5$ cm de diâmetro cada e uma roda boba frontal de apoio. Apenas um sensor fora aplicado nesta simulação, o giroscópio e, dois atuadores *joints* (um em cada roda), os quais retornam as respectivas velocidades nos eixos.

Por tratar-se da primeira de três etapas do projeto, nesta fase inicial, não há necessidade do sensor GPS (*Global Positioning System* ou Sistema de Posicionamento Global) nem do giroscópio, para aferir a posição e orientação em relação a um referencial global, respectivamente. É possível medir a velocidade e a orientação do robô, apenas com a leitura dos atuadores nas rodas traseiras, onde retornam em radianos por segundos, as rotações em cada eixo. Assim, sabe-se as rotações de cada motor, logo, é possível medir a velocidade do corpo como um todo; bem como, as curvas realizadas pelas distintas rotações em cada roda, promovendo a orientação e posição do mesmo a um referencial global.

Entretanto, a nível de simplificação da simulação, o giroscópio foi aplicado, pois o *software* permite apenas um ponto a ser lido dos *joints* nas rodas, onde foi escolhido a velocidade. Com isso, a orientação é dada por este sensor e as demais aferições são realizadas a partir das leituras dos atuadores nos eixos. Ressalta-se também, que foram desprezados os erros de atuação, posição, atrito, derrapagem lateral etc. Onde foi determinado um ponto inicial fixo, aplicando as velocidades desejadas como entrada e simulando a movimentação do robô móvel ideal.

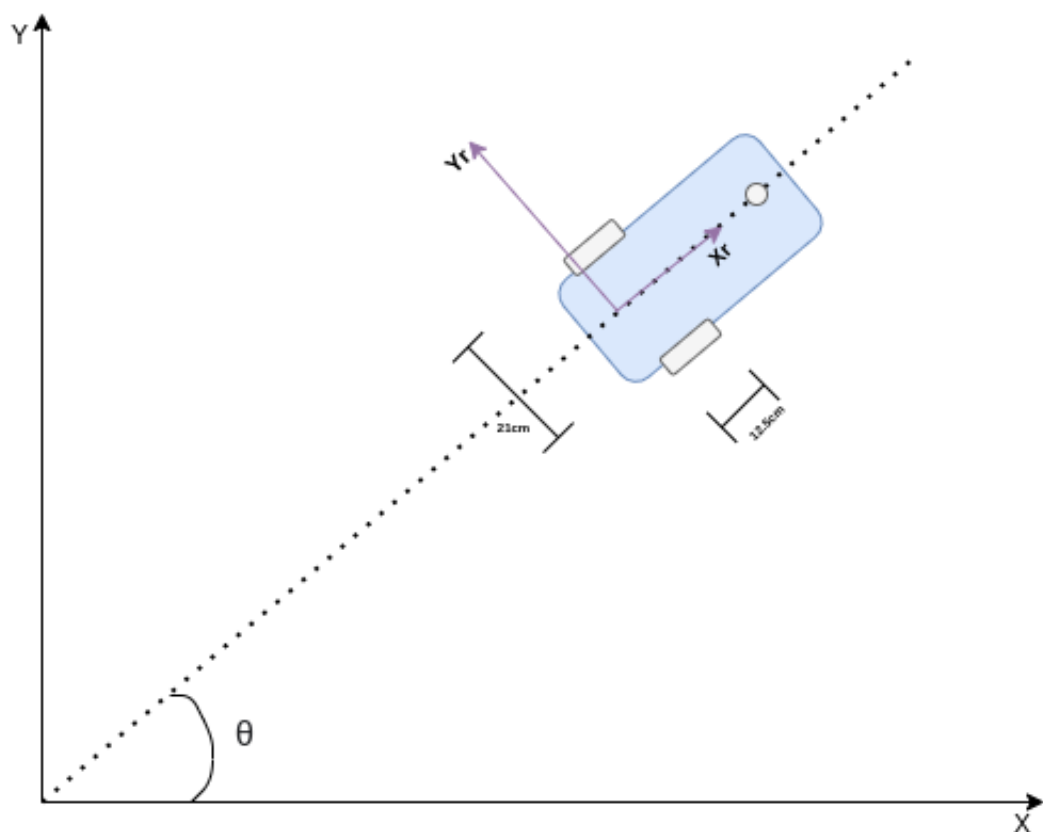
2 OBJETIVO

O propósito desta simulação é aplicar os conhecimentos teóricos aprendidos, adotando o modelo cinemático em um robô móvel com acionamento diferencial, onde comanda-se as velocidades das rodas, bem como, sua referência; retornando sua posição e orientação em relação a um referencial global.

3 METODOLOGIA

Foi realizado primeiramente a uma modelagem não holonômica do robô para projetar o controlador e trajeto. Dessa forma, empregou-se os conceitos aprendidos de modelagem cinemática e não holonômica de robôs, como demonstrada na Figura 1.

Figura 1. Modelagem do robô



Fonte: Autores.

Roda Direita:

- $\theta_d = -90^\circ$
- $\alpha_d = 180^\circ$

- $l_d = b/2$

Roda Esquerda:

- $\theta_e = 90^\circ$
- $\alpha_e = 0^\circ$
- $l_e = b/2$

Com base nessas informações podemos obter as equações derrapagem lateral e rolamento:

$$\begin{bmatrix} 1 & 0 & \frac{b}{2} \end{bmatrix} * ({}^iR_R(\theta))^T * q' = w_d * r_d \quad (1)$$

$$\begin{bmatrix} 1 & 0 & -\frac{b}{2} \end{bmatrix} * ({}^iR_R(\theta))^T * q' = w_e * r_e \quad (2)$$

Restrição de derrapagem lateral:

$$\begin{bmatrix} 0 & 1 & 0 \end{bmatrix} * ({}^iR_R(\theta))^T * q' = 0 \quad (3)$$

Ao agruparmos as Equações 1, 2 e 3 teremos nossa modelagem cinemática do robô:

$$\begin{bmatrix} 1 & 0 & \frac{b}{2} \\ 1 & 0 & -\frac{b}{2} \\ 0 & 1 & 0 \end{bmatrix} * ({}^iR_R(\theta))^T * q' = \begin{bmatrix} r_d & 0 \\ 0 & r_e \end{bmatrix} * \begin{bmatrix} w_d \\ w_e \end{bmatrix} \quad (4)$$

para encontrar q' :

$$\begin{bmatrix} X \\ Y \\ \theta \end{bmatrix} = \begin{bmatrix} \frac{\cos(\theta) * (r_d * w_d + r_e * w_e)}{2} \\ \frac{\sin(\theta) * (r_d * w_d + r_e * w_e)}{2} \\ \frac{(r_d * w_d - r_e * w_e)}{b} \end{bmatrix} \quad (5)$$

onde $r_d = r_e = 6,25cm$ e $b = 21cm$.

Dessa forma podemos modelar um controlador que queremos qual a coordenado e orientação do robô no mundo, e passamos o velocidade angular das rodas podemos ter a posição orientação.

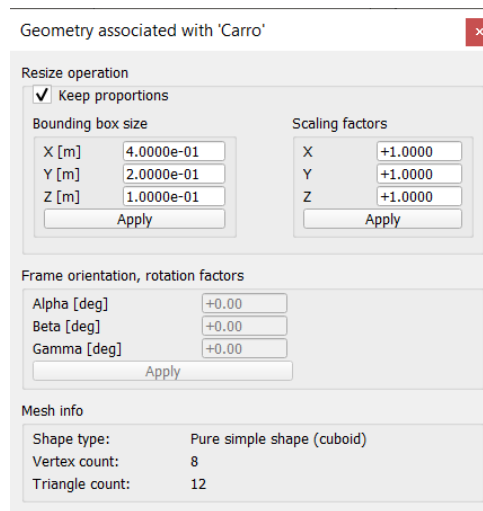
4 SIMULAÇÃO DO MODELO CINEMÁTICO

Com as informações do modelo cinemático, podemos implementar esse robô e modelar no **V-REP**, aplicando as dimensões da modelagem.

4.1 SIMULAÇÃO

No experimento, foram inseridas as entradas das velocidades em cada motor, sendo estas, para roda direita 4 rad/s e para roda esquerda 2 rad/s , os parâmetros definidos estão ilustrados na Figura 2.

Figura 2. Entradas no Robô Móvel



Geometry associated with 'Carro'

Resize operation

☒ Keep proportions

Bounding box size

X [m]	4.0000e-01
Y [m]	2.0000e-01
Z [m]	1.0000e-01

Apply

Scaling factors

X	+1.0000
Y	+1.0000
Z	+1.0000

Apply

Frame orientation, rotation factors

Alpha [deg]	+0.00
Beta [deg]	+0.00
Gamma [deg]	+0.00

Apply

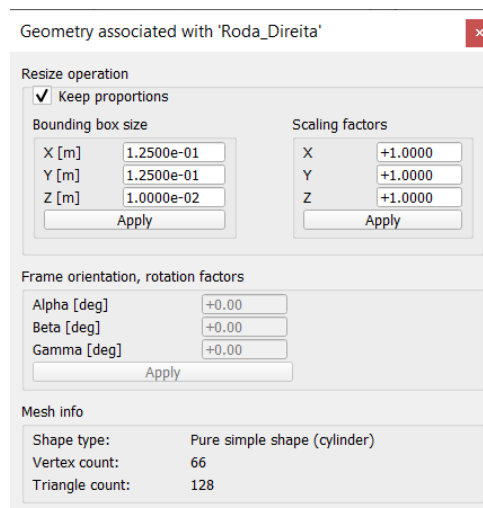
Mesh info

Shape type:	Pure simple shape (cuboid)
Vertex count:	8
Triangle count:	12

Fonte: Autores.

Similarmente, foi definido os parâmetros para cada roda. Onde são demonstrados nas Figuras 3 e 4.

Figura 3. Entradas na Roda Direita



Geometry associated with 'Roda_Direita'

Resize operation

☒ Keep proportions

Bounding box size

X [m]	1.2500e-01
Y [m]	1.2500e-01
Z [m]	1.0000e-02

Apply

Scaling factors

X	+1.0000
Y	+1.0000
Z	+1.0000

Apply

Frame orientation, rotation factors

Alpha [deg]	+0.00
Beta [deg]	+0.00
Gamma [deg]	+0.00

Apply

Mesh info

Shape type:	Pure simple shape (cylinder)
Vertex count:	66
Triangle count:	128

Fonte: Autores.

Como mencionado anteriormente, também é possível fornecer as entradas via *Script*. Assim, o código a seguir fornece as mesmas velocidades, onde são definidos os valores para cada

Figura 4. Entradas na Roda Esquerda

Geometry associated with 'Roda_Esquerda'

Resize operation

☒ Keep proportions

Bounding box size

X [m] 1.2500e-01

Y [m] 1.2500e-01

Z [m] 1.0000e-02

Apply

Scaling factors

X +1.0000

Y +1.0000

Z +1.0000

Apply

Frame orientation, rotation factors

Alpha [deg] +0.00

Beta [deg] +0.00

Gamma [deg] +0.00

Apply

Mesh info

Shape type: Pure simple shape (cylinder)

Vertex count: 66

Triangle count: 128

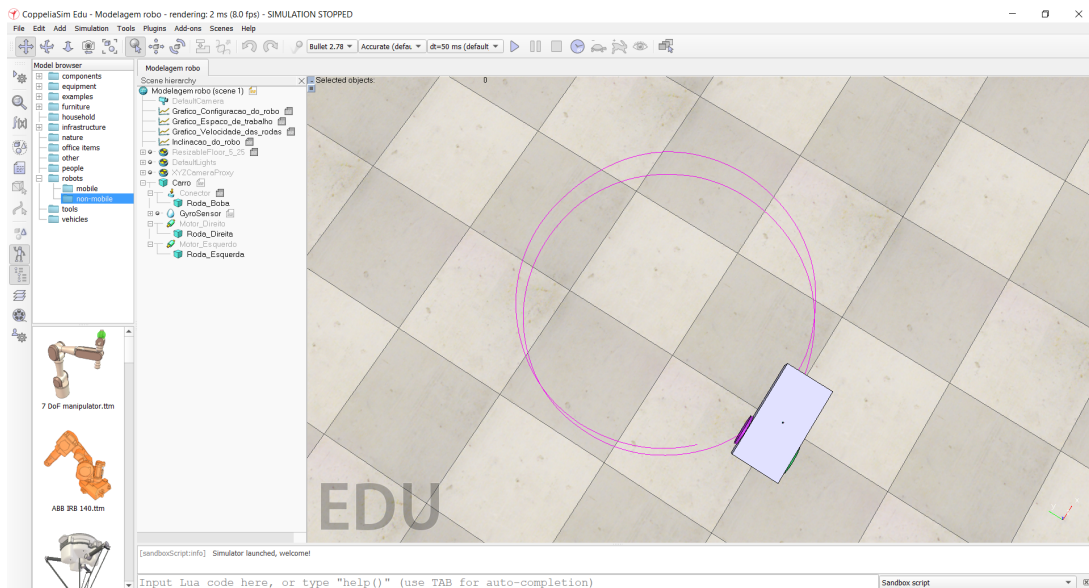
Fonte: Autores.

motor, atribuindo a variável a roda específica com os respectivos parâmetros, resultando em uma simulação idêntica.

```
1 if (sim_call_type==sim.syscb_init) then
2     motorLeft=sim.getObjectHandle("Motor_Esquerdo")
3     motorRight=sim.getObjectHandle("Motor_Direito")
4 end
5
6 if (sim_call_type==sim.syscb_actuation) then
7     sim.setJointTargetVelocity(motorLeft,2)
8     sim.setJointTargetVelocity(motorRight,4)
9 end
```

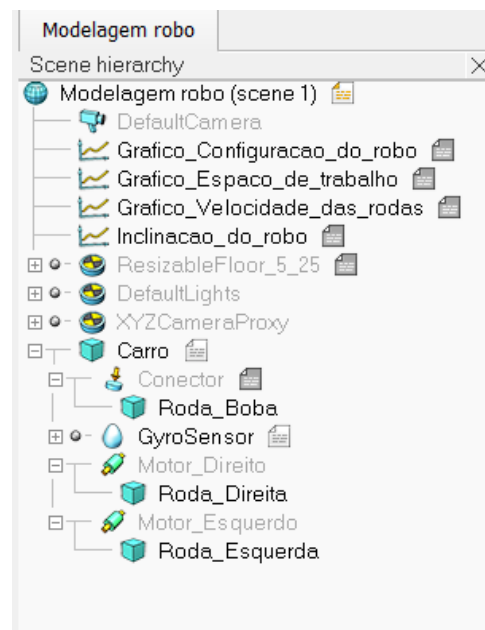
Dessa forma, o robô realiza um movimento circular anti-horário contínuo, como ilustrado na Figura 5. A hierarquização do simulador, é ilustrada na Figura 6.

Figura 5. Trajetória do Robô



Fonte: Autores.

Figura 6. Hierarquização do Simulador

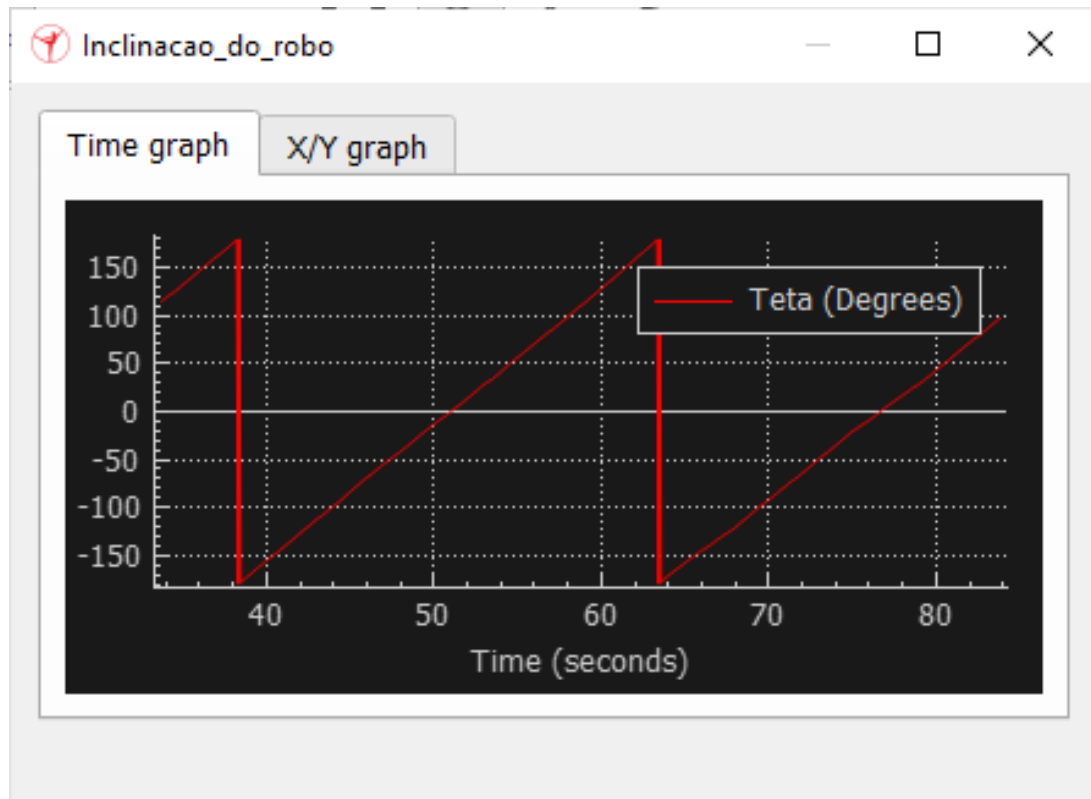


Fonte: Autores.

4.2 RESULTADOS

Posteriormente, foram definidas as variáveis a serem medidas e *plotadas* em gráficos. O gráfico da inclinação do robô em graus por segundos está esboçado na Figura 7.

Figura 7. Gráfico da Inclinação em Graus do Robô no Tempo

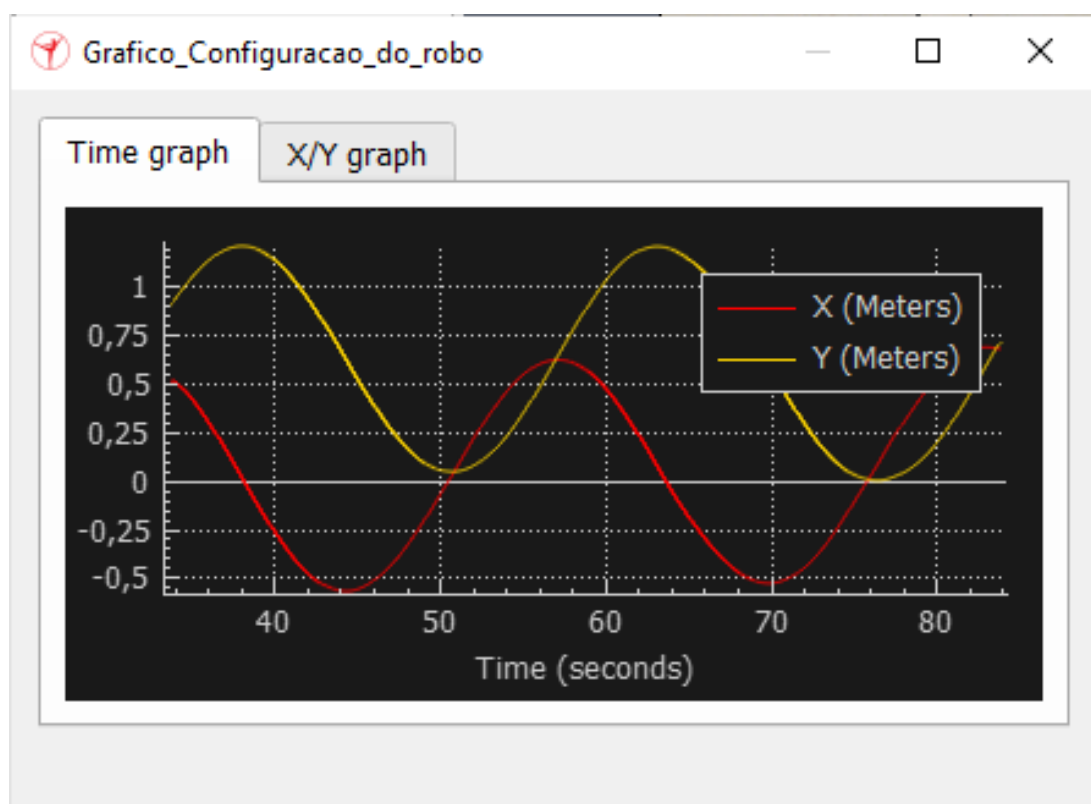


Fonte: Autores.

Os gráficos da configuração do robô em metro por segundo e o da velocidade de cada motor em graus por segundos, são esboçados nas Figuras 8 e 9, respectivamente.

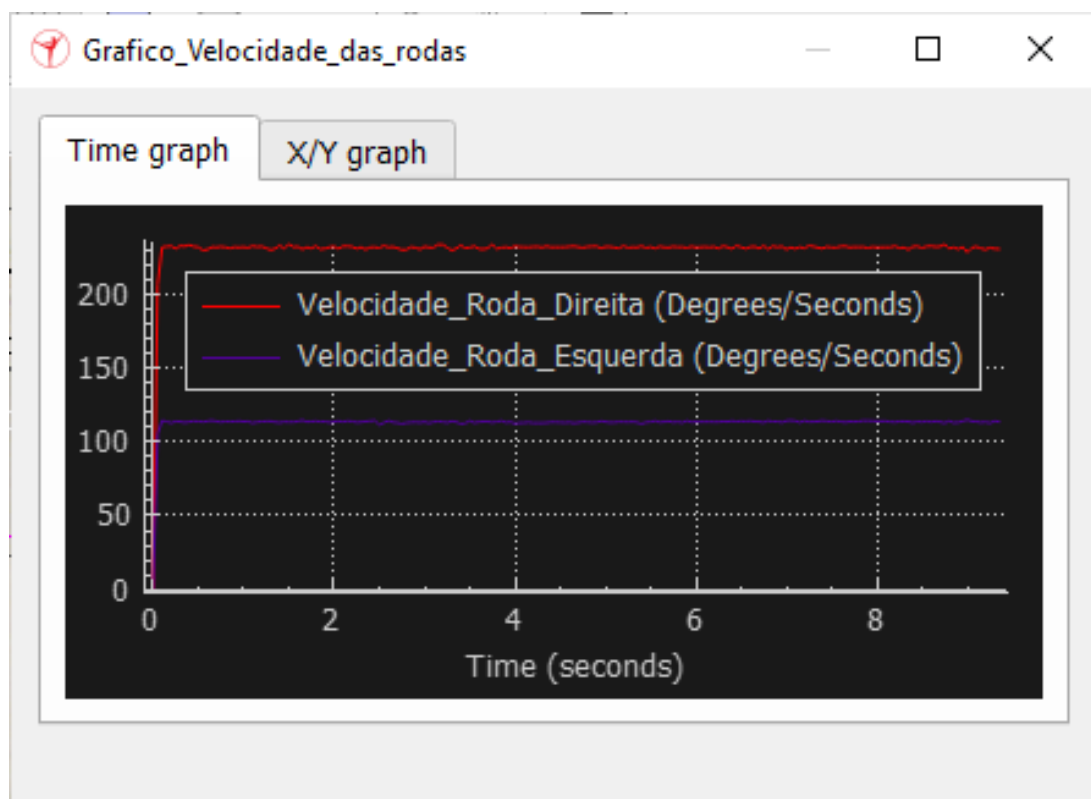
Na Figura 10 é ilustrado o gráfico da trajetória percorrida no plano de trabalho. Por fim, a Figura 11 esboça um *frame* da simulação em andamento com os descritos gráficos em tempo real.

Figura 8. *Gráfico da Configuração do Robô*



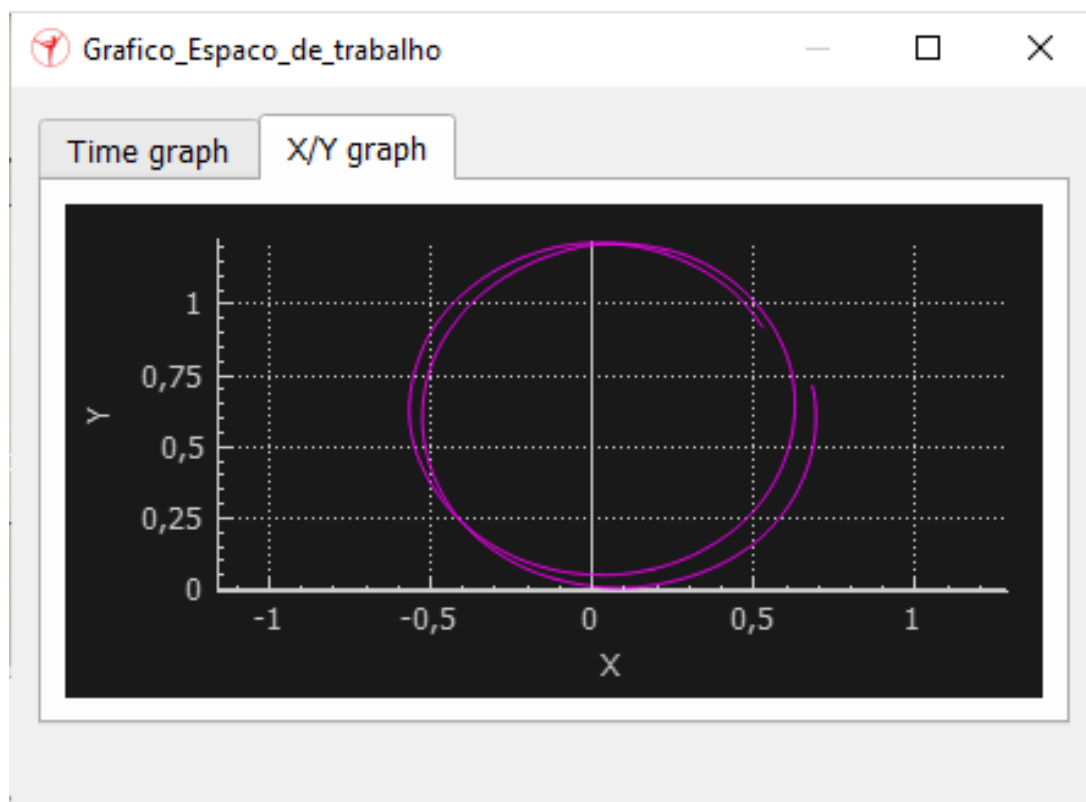
Fonte: Autores.

Figura 9. *Gráfico da Velocidade dos Motores*



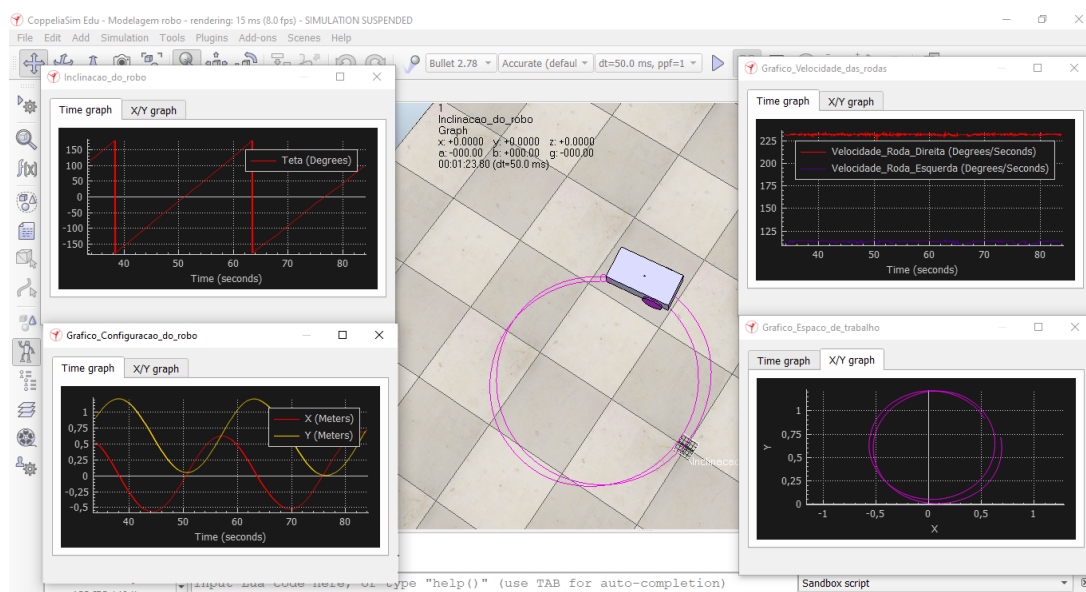
Fonte: Autores.

Figura 10. Gráfico da Trajetória Percorrida



Fonte: Autores.

Figura 11. Frame da Simulação com os Gráficos



Fonte: Autores.

5 CONCLUSÃO

Com a inovação de simuladores como este, pesquisadores de todos os níveis podem realizar suas pesquisas sem a necessidade física do dispositivo robótico. Assim, reduzindo ao máximo os possíveis erros de programação e custos, antes de aplicar no robô físico.

A partir desta, também facilitou a compreensão da teoria mostrada em aula, principalmente, nos casos de escassez ou ausência dos materiais necessários para tal finalidade. Outro sim, destaca-se a importância dessas ferramentas simulatórias, para o caso atual de pandemia, onde as aulas são remotas.

Analisando os gráficos demonstrados, é notória melhoria na percepção do comportamento de um robô móvel. Com uma melhor clareza, percebe-se que para uma simples trajetória, a disposição para esse tipo de robô segue uma lógica cíclica. Nos próximos experimentos, serão inseridos obstáculos, os quais devem ser evitados no caminho previamente definido pelo usuário.

6 REFERÊNCIAS

- [1] ROHMER, E.; SINGH, S. P. N.; FREESE, M. V-rep: A versatile and scalable robot simulation framework. In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. [S.l.: s.n.], 2013. p. 1321–1326. 4
- [2] NASCIMENTO, L. B. P. et al. Introdução ao v-rep: Uma plataforma virtual para simulação de robôs. In: *Alex Oliveira Barradas Filho; Pedro Porfirio Muniz Farias; Ricardo de Andrade Lira Rabêlo. (Org.). Minicursos da ERCEMAPI e EAComp 2019. 1ed. Porto Alegre: Sociedade Brasileira de Computação*. [S.l.: s.n.], 2019. p. 49–68.