



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



2º RELATÓRIO: Planejadores de Caminhos em Grafos e Campos de Potenciais

CoppeliaSim 4.1.0 (V-REP)

Natal - RN

2021

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

2º Relatório: Planejadores de Caminhos em Grafos e Campos de Potenciais
CoppeliaSim 4.1.0 (V-REP)

Segunda entrega de três do 2º 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	4
3 METODOLOGIA	4
4 PLANEJADOR DE CAMINHOS	6
4.1 GRAFOS DE VISIBILIDADE	7
4.2 CAMPOS DE POTENCIAIS	9
5 CONCLUSÃO	12
REFERÊNCIAS	13

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 com várias ferramentas revolucionárias. 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.

Na Primeira etapa foi criado um modelo simples de robô móvel com a modelagem cinemática, como discutida antes, também criado controladores de posição, seguidor de caminho e trajetória.

Nessa etapa, de planejamento de mapa, aonde criamos nosso espaço de configuração (robô como um ponto), incluindo os obstáculos poligonais na planta baixa e detecção de colisões, e projetar caminho a ser percorrido com o robô.

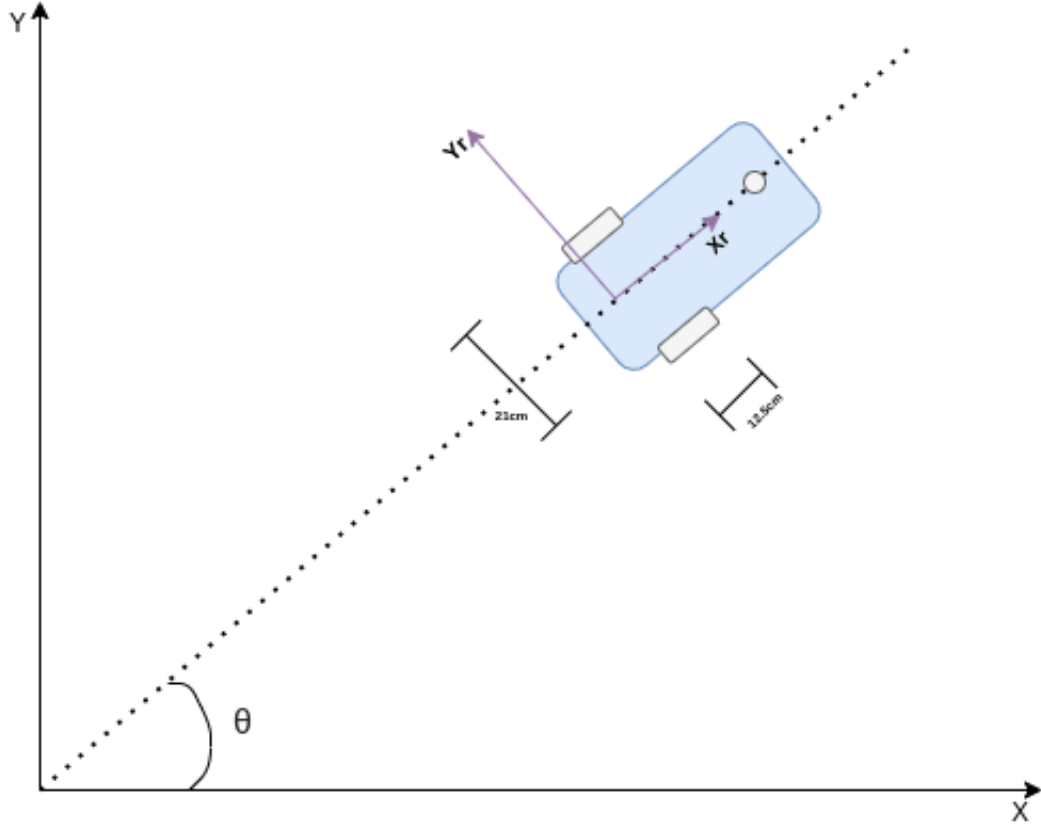
2 OBJETIVO

O propósito desta simulação é aplicar os conhecimentos teóricos aprendidos, desenvolvendo Planejadores de Caminhos para Robô Móvel que permitam ao mesmo executar movimentos especificados em espaço povoado de obstáculos, sem colidir com os mesmos. Nesta segunda entrega são modelados os obstáculos como polígonos e criado o espaço de configuração, onde é possível representar nosso robô como um ponto no mapa para detecção de colisões; além disso, planejar caminho no mapa, usando técnicas de grafos e campos vetoriais.

3 METODOLOGIA

Primeiramente a uma modelagem não holonômica do robô para projetar o controlador e trajeto foi calculado com os parâmetros do robô. 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:

- $\theta_d = -90^\circ$
- $\alpha_d = 180^\circ$
- $l_d = \frac{b}{2}$

Roda Esquerda:

- $\theta_e = 90^\circ$
- $\alpha_e = 0^\circ$
- $l_e = \frac{b}{2}$

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 5 teremos nossa modelagem cinemático 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 $b = 21cm$.

Além da posição e orientação, necessita-se do controle das velocidades linear e angular para concretizar o controlador da trajetória a ser seguida. Dessa forma podemos iniciar a modelagem dos controladores, os quais serão discutidos nos próximos tópicos.

Inicialmente utilizamos a API do V-REp com Python para projetar os controladores, porém para essa segunda etapa que é para planejamento do mapa, teríamos que implementar funções do zero, a qual não é o principal intuito dessa disciplina, por isso optamos com mudar nossa principal ferramenta de implementação para MATLAB (MATrix LABoratory) trata-se de um software interativo de alta performance voltado para o cálculo numérico. O MATLAB integra análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos em ambiente fácil de usar onde problemas e soluções são expressos somente como eles são escritos matematicamente, ao contrário da programação tradicional, onde tem funções já implementadas a partir de tools box como *Robotics System Toolbox*.

Na primeira etapa do projeto foi dividido em três subetapas: a primeira foi a simulação do modelo cinemático, a segunda etapa foi gerado o caminho baseado em um polinômio de 3° aplicado pelo usuário e, a terceira, implementa os controladores cinemáticos do robô móvel.

Nessa segunda etapa do projeto, objetivou-se o planejamento do mapa. Para isso, foram inseridos obstáculos no espaço de trabalho, os quais devem ser reconhecidos pela câmera e considerados pelo robô móvel ao planejar o caminho.

4 PLANEJADOR DE CAMINHOS

Com as informações do modelo cinemático, podemos implementar esse robô e modelar no **V-REP**, aplicando as dimensões da modelagem para testar a simulação e gerar os gráficos necessários, antes de implementar o controle das velocidades via *script*.

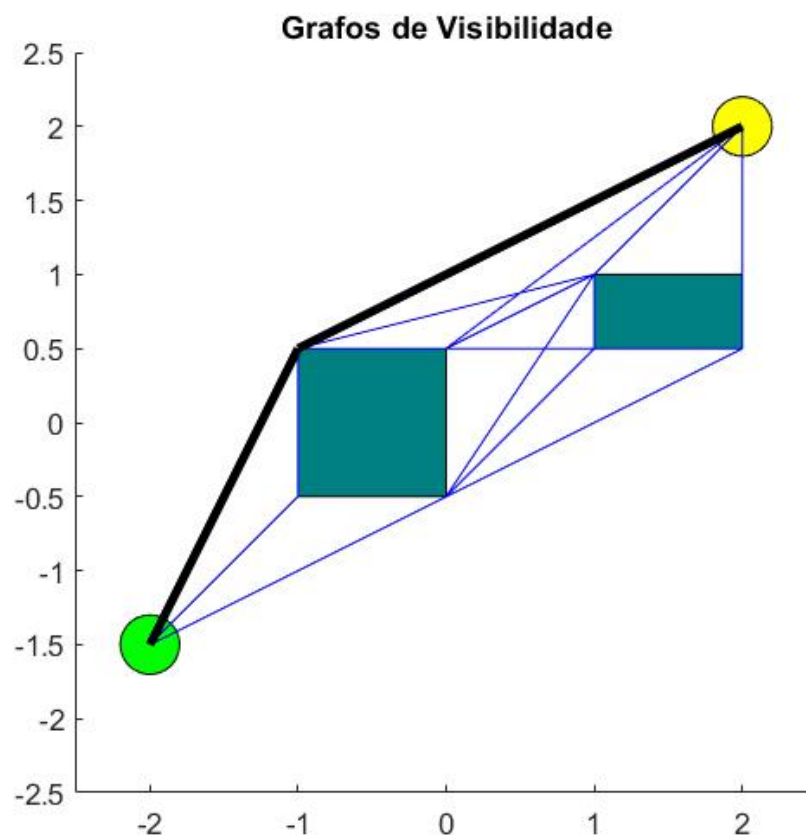
Neste experimento, foram criadas duas simulações para planejamento dos caminhos, uma pelo método de visibilidade de grafos e outra pelos campos potenciais. Ambas possuem o mesmo robô móvel e foram dimensionadas com o mesmo tamanho do espaço de trabalho, porém os obstáculos são distintos. O espaço trabalhado possui dimensões de $5 \times 5m$ e a origem encontra-se no centro.

4.1 GRAFOS DE VISIBILIDADE

A primeira experiência, criou caminhos baseados em grafos. Dessarte, retas foram criadas a partir do ponto de origem ao ponto final, concatenando os vértices dos obstáculos. Ao mapear todas as retas que ligam os pontos aos vértices e estes entre si, o caminho seguido é determinado pela menor distância a ser percorrida, somando-se as retas traçadas.

Na Figura 2 está representado a imagem do planejamento por grafos, onde os círculos em verde e amarelo, representam respectivamente, os pontos inicial e final. As retas em azul, simbolizam os caminhos traçados pelos vértices e pontos; a linha em preto, ilustra o menor caminho a ser seguido, desviando dos obstáculos.

Figura 2. *Grafos de Visibilidade*

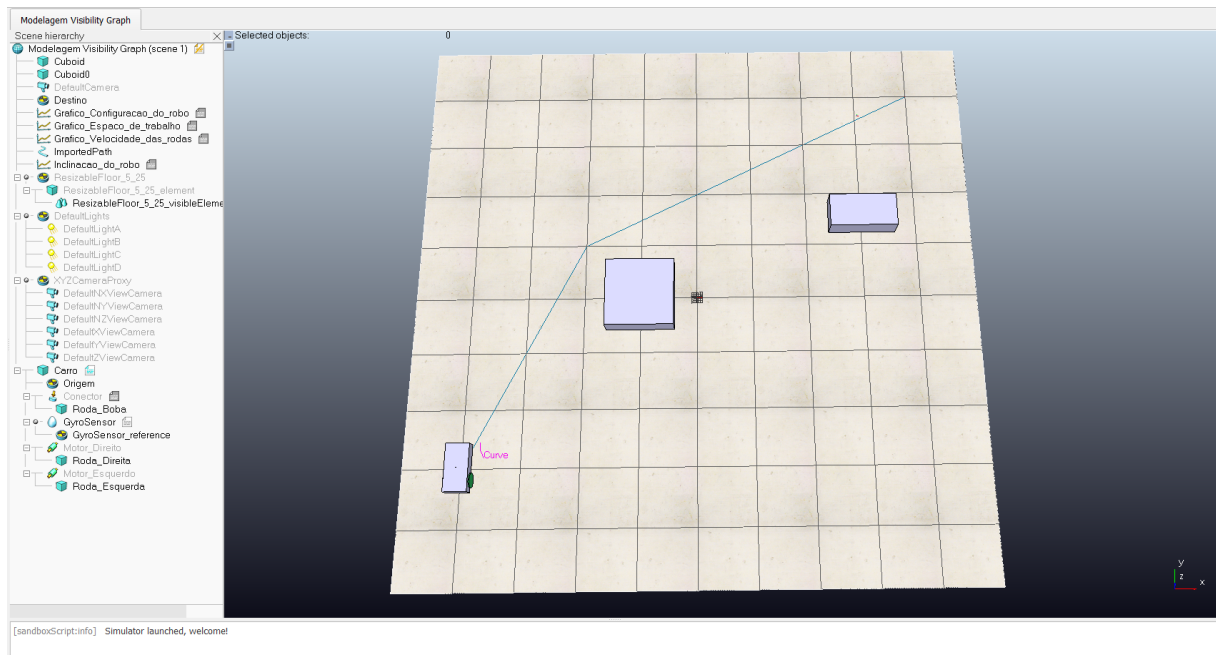


Fonte: Autores.

Após gerar a matriz de pontos a serem seguidos e exportar do *MatLab*, é possível aplicar o controle de velocidades (seguidor de caminho) para simular o trajeto do robô no *V-REP*. Como foi utilizado um ponto para criação do caminho, necessitou-se de um ajuste nas dimensões dos obstáculos, equivalente ao raio de giro do robô utilizado no simulador. Com isso, foi reduzido *40cm* (dobro do raio de giro para ajustar o "diâmetro") dos obstáculos, o que evita colisão do carro ao passar próximo ao vértice do obstáculo.

A Figura 3 demonstra a cena do simulador, com o caminho a ser seguido em azul e os objetos em cinzas, representam proporcionalmente os obstáculos ponderados no gerador de grafos.

Figura 3. *Grafos de Visibilidade no V-REP*

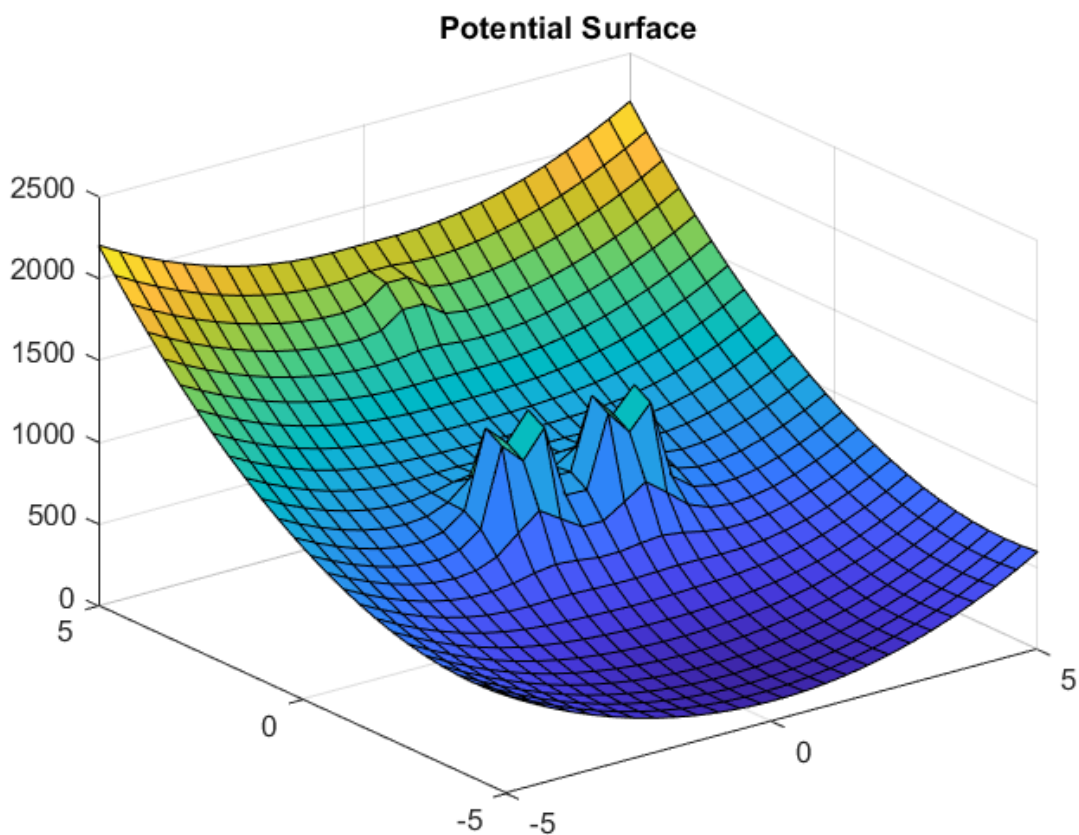


Fonte: Autores.

4.2 CAMPOS DE POTENCIAIS

Na segunda experiência, aplicou-se o método de planejamento de caminhos por campos potenciais. Neste, foram inseridos dois obstáculos idênticos, um ao lado do outro, de formato cilíndrico. O método consiste em "descer" da maior altitude para a menor (ponto inicial mais elevado do que o ponto final), evitando os obstáculos, os quais geram um campo potencial ao redor destes. Na Figura 4 é ilustrado o mapa da superfície gravitacional, onde parte em azul representa a área de posição final e os dois vales ao centro, os obstáculos cilíndricos.

Figura 4. *Mapa de Superfície Gravitacional*

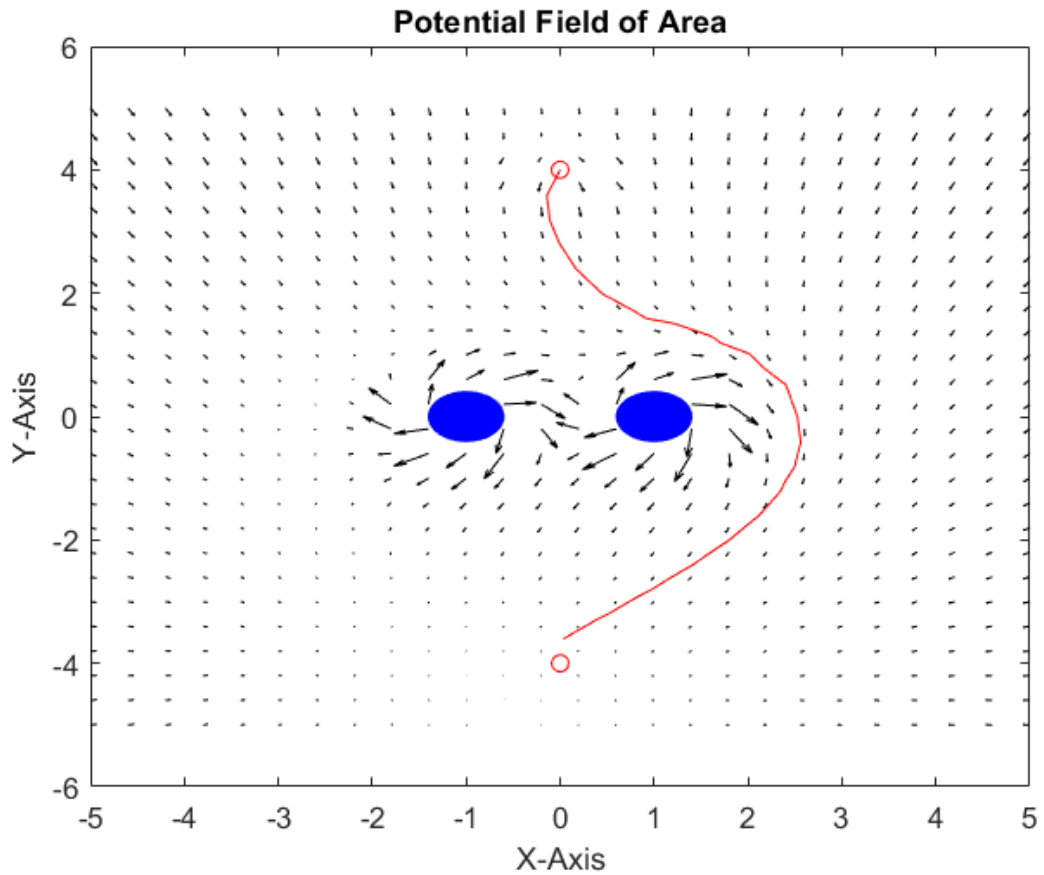


Fonte: Autores.

Assim, quanto mais próximo do objeto, maior será o vetor do campo gerado e o robô móvel deve locomover-se na menor distância necessária, desviando dos vetores de valores altos, próximos ao objeto. Dito isto, o robô percorre apenas vetores até o tamanho limitado, distanciando-se dos obstáculos de acordo com o formato destes. Na Figura 5 está representado o campo potencial com os vetores ilustrados pelas setas pretas (quanto maior a seta, maior a força do campo), os objetos cilíndricos em azul e a curva a ser seguida em vermelho.

Posteriormente, é importado para o simulador a curva a ser seguida pelo controle das velocidades linear e angular. Dessa maneira, o robô percorre o menor caminho até seu destino,

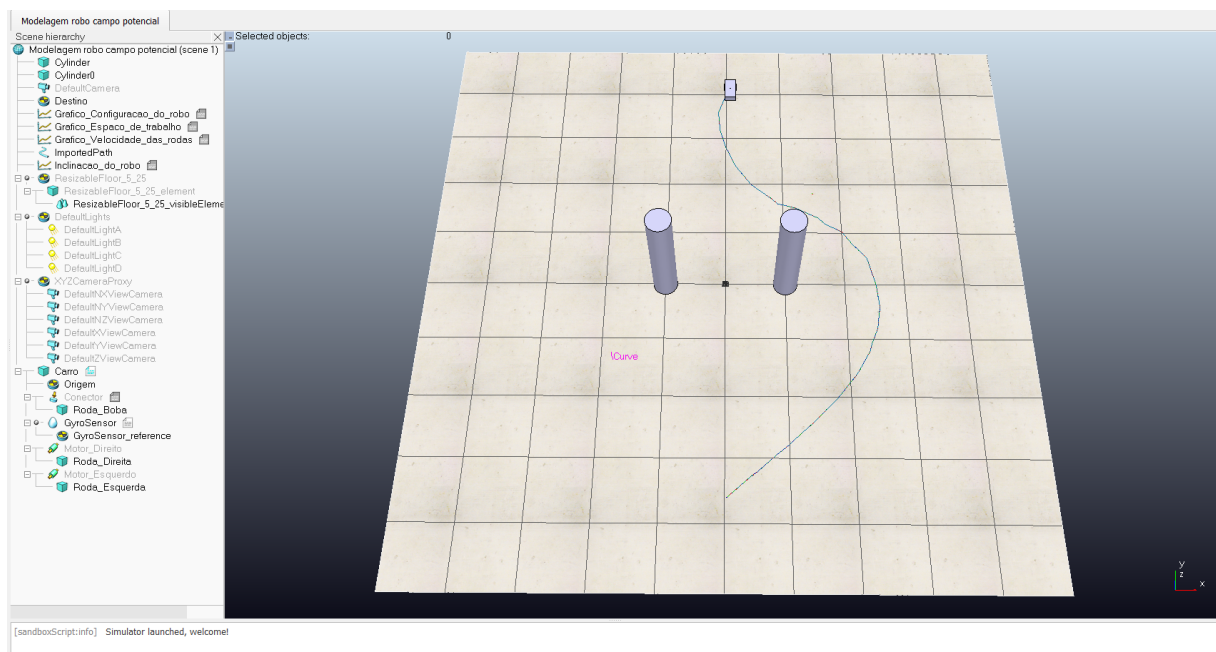
Figura 5. *Área do Campo Potencial*



Fonte: Autores.

nota-se que o mesmo não traçou a curva ao centro, pois o limite determinado pelo usuário foi alto, podendo assim, ajustar para adequar-se ao tamanho real do carro. A Figura 6 demonstra a cena do *V-REP* com o caminho traçado em azul, similar ao planejado.

Figura 6. *Campo Potencial no V-REP*



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 da aplicação 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. Ademais, destaca-se a importância dessas ferramentas simulatórias, para o caso atual de pandemia, onde as aulas são remotas.

O planejamento de caminho foi implementado com sucesso e a precisão é dada pelo limite de pontos escolhidos para traçado das rotas sem colisões. Para tanto, dependendo da aplicação, a precisão pode ser maior, porém demanda proporcionalmente o aumento do gasto computacional.

Como o ajuste da distância do raio de giro foi exatamente o tamanho do robô móvel, ressalta-se que deve-se considerar erros de dimensionamento para aplicações reais, onde deve ser ajustado o aumento dessa distância para garantir a não colisão deste.

Finalmente, o experimento resultou nas proposições desejadas e este serve de base para os próximos, os quais serão implementados, o planejador de caminho baseado em amostragem para o robô móvel. Este será demonstrado pelo simulador e controlado pelo seguidor de caminho desenvolvido no primeiro projeto.

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 da aplicação 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. Ademais, destaca-se a importância dessas ferramentas simulatórias, para o caso atual de pandemia, onde as aulas são remotas.

O controle da trajetória foi implementado com sucesso, porém necessita de ajustes finos para suavizar mais as curvas e aumentar a velocidade, sem perder a precisão do controle de posição.

Analisando os gráficos demonstrados, é notória melhoria na percepção do comportamento de um robô móvel com o ajuste dos ganhos. Com uma melhor clareza, percebe-se que para uma simples trajetória, o controle requer muito processamento para receber e enviar remotamente, as informações de comando para locomoção e estado posição/orientação atual do robô.

Como o processamento é feito em tempo real, os gráficos obtidos pela simulação são muito próximos ao comportamento em um experimento com *hardware* de predefinições similares e em boas condições estruturais. Assim, a certividade nas análises destes é mais pontual e econômica. Outrossim, é a experiência proporcionada em um simulador utilizado tanto no âmbito acadêmico, como comercial.

Finalmente, o experimento resultou nas proposições desejadas e este serve de base para os próximos, os quais serão inseridos obstáculos. Estes serão evitados, onde o caminho gerado

deverá encontrar o melhor trajeto para o destino final, sem nenhuma colisão.

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.