

Projeto 1 - Robótica Móvel

OSVALDO XAVIER DIAS JUNIOR E EDUARDO DE OLIVEIRA MORAES *

*Ciência da Computação - Pós-Graduação

E-mail: osvaldoxdjr@gmail.unicamp.br

eduardomoraesmail@gmail.com

Resumo – Este relatório trata o projeto 1 (P1) da disciplina de robótica móvel, o trabalho é sobre o simulador robótico V-REP, um sistema de cômputo da odometria e extração de características para o robô Pioneer 3-DX, isso é relevante pois abrange os conceitos básicos sobre simulação de robôs em um ambiente de software. Utilizou-se rotinas em Python para comunicação com o simulador V-REP, com o propósito de efetuar a localização do robô e o mapeamento da cena. Foram testados, durante o desenvolvimento, sensores como encoder para odometria, e sonar e laser 2D para mapeamento, os resultados obtidos foram bons, com exceção da odometria para longas distâncias percorridas pelo robô. A parte de extração de características ficou por conta da aplicação das transformadas de Hough, entretando sem êxito na sua aplicação. O desenvolvimento do projeto apresentou uma curva de aprendizado lenta, devido a falta de fluência no simulador, impactando no tocante as metas propostas inicialmente do projeto.

Palavras-chave – V-REP, P3-DX, Robô, odometria, extração de características, transformada de Hough e ground truth

I. INTRODUÇÃO

O desafio de desenvolver robôs é notoriamente grande e complexo, devido a interdisciplinaridade de assuntos que envolvem, majoritariamente, engenharia. Posta essa grande complexidade, sabe-se que para confeccionar um robô é requerido um determinado custo que em alguns casos podem ser muito elevados. Logo, existem softwares que são capazes de fazer a simulação de robôs, como por exemplo o V-REP que foi utilizado nesse projeto 1.

O V-REP, em linhas gerais, é um simulador no qual faz-se a escolha de uma dinâmica de operação, e também pode-se colocar objetos como sensores, atuadores, robôs, anteparos, e etc. na cena (ambiente simulado no qual um protótipo de robô será inserido). Uma das formas tomar ações é através de comunicação remota com os objetos da cena utilizando-se de diferentes linguagens de programação como C/C++, Python, Java, Matlab, Octave e Lua, além disto os objetos podem possuir rotinas de simulação, ou childscripts, que estão codificadas em Lua; estas rotinas são pré-programações dos objetos para tomar determinadas ações sem que seja preciso o usuário do objeto desenvolver.

Um ponto fundamental quando se deseja desenvolver um robô móvel é localização, pois esta é a premissa básica para que o robô possa se mover de um ponto a outro, sem interação humana. A localização consiste de ter conhecimento de onde o robô está localizado com referência ao referencial da cena simulada, dada a importância desse tópico existe um ramo da robótica que estuda uma parte desse problema que é a odometria. Esta consiste no estudo da posição do robô através

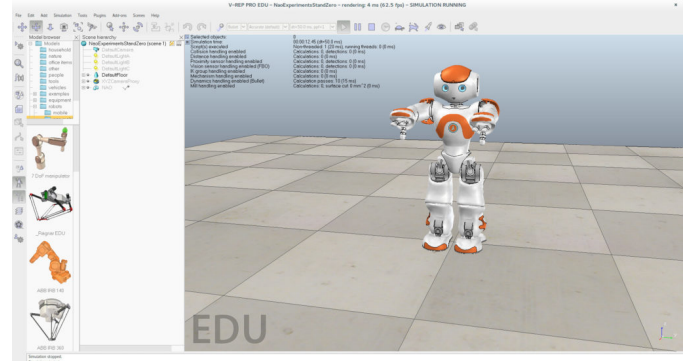


Figura 1. Ambiente de simulação V-REP

do deslocamento incremental de sua mecânica de locomoção, por exemplo através do cômputo da movimentação do robô através da velocidade angular de rodas. Para o robô diferencial (robô que movimenta-se através de duas rodas montadas num eixo comum e controladas por motores independentes, um para cada roda) têm que o cálculo da odometria é dado por:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_t = \begin{bmatrix} x \\ y \\ z \end{bmatrix}_0 + \sum_{k=1}^t \begin{bmatrix} \Delta s \cos(\theta + \frac{\Delta\theta}{2}) \\ \Delta s \sin(\theta + \frac{\Delta\theta}{2}) \\ \Delta\theta \end{bmatrix} \quad (1)$$

$$\Delta s = \frac{R(V_r + V_l)}{2} \Delta t \quad (2)$$

$$\Delta\theta = \frac{R(V_r - V_l)}{2L} \Delta t \quad (3)$$

Onde x, y e z são as coordenadas; Δs é a distância percorrida; $\Delta\theta$ é a variação angular; R é o raio das rodas; L é a distância entre as rodas; V_r é a velocidade da roda direita e V_l é a velocidade da roda esquerda.

Com as informações acima é possível estimar a trajetória do robô ao longo do tempo, porém nota-se pela somatória temporal que existe um acúmulo de erro ao longo do tempo que pode resultar em uma trajetória diferente daquela indicada pelo ground truth.

O mapeamento da cena consiste da elaboração de um mapa do ambiente simulado, e está intimamente atrelado a movimentação do robô, pois através daquele é possível movimentar o robô com maior eficiência evitando assim obstáculos e executar deslocamentos mais rapidamente, uma vez que, sabe-se exatamente a trajetória a ser percorrida. O mapeamento

pode ser realizado utilizando-se diversos sensores como sonares, radares, lasers, cameras e etc. ou também utilizando um sub conjunto dos componentes citados anteriormente.

Contudo, uma questão fundamental em localização, odometria e mapeamento é o comparativo entre referencial inercial ou referencial global e referencial de um objeto qualquer, quando o robô realiza qualquer medição que envolva posicionamento em cena, é preciso sempre tomar a precaução de trabalhar com coordenadas em referencial inercial. Uma ferramenta para solucionar esse problema é utilizar-se de conceitos de álgebra linear - matrizes de rotação e translação.

$$T_{trans} = \begin{bmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$T_{trans} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$Pos_I = T_{trans} * T_{rot} * Pos_r \quad (6)$$

Onde Δx representa a variação da coordenadas x dos referenciais, Δy representa a variação da coordenadas y dos referenciais e α representa o ângulo entre os referenciais. A equação 6 é utilizada para rotacionar e transladar o ponto de um referencial local para o global, respeitando princípio de não comutatividade. Para extrair características do mapeamento de uma cena é possível utilizar a transformada de Hough que é uma técnica matemática que realiza a detecção de formas geométricas em imagens digitais, no caso específico de um mapa 2D seria a detecção de linhas a partir de uma nuvem de pontos.

A partir daqui este trabalho encontra-se organizado da seguinte forma: a seção 2 apresenta "Trabalho Proposto". A seção 3 descreve "Materiais e Métodos". Os "Resultados e Discussão" são apresentados na seção 4, e as "Conclusões" são apresentadas na seção 5.

II. TRABALHO PROPOSTO

As soluções propostas foram baseadas em mapeamento de cena utilizando sonar e laser, cálculo de odometria utilizando o encoder das rodas laterais do robô e extração de característica utilizando transformada de Hough. Segue a lista tarefas executadas, em sequência cronológica, durante o desenvolvimento:

- 1) Ambientação com ambiente de simulação
- 2) Localização via ground truth
- 3) Localização via odometria
- 4) Mapeamento através de sonar
- 5) Mapeamento através de laser 2D
- 6) Extração de característica

III. MATERIAIS E MÉTODOS

Para desenvolver o trabalho foi utilizado o remote API do V-REP em Python, baseado em uma comunicação cliente-servidor. O robô utilizado na simulação foi o P3-DX, um robô diferencial que, portanto, possui duas rodas com um motor

em cada uma delas e 16 sonares já pré-instalados. Este robô possui um chilsdscript em Lua que executa algumas tarefas, entre elas a navegação através do algoritmo de Braitenben. Este código nativo é capaz fazer a navegação do robô e evitar colisões, considerando a distância medida pelo sonar e ativando os motores das rodas de forma ponderada, esta navegação foi adotada para a execução das simulações. Foi inserido uma linha de código no servidor para que este ficasse esperando a conexão com o cliente.

Já no script do cliente, todo desenvolvido em Python, foi criada a lógica para leitura dos sensores e interpretação dos dados. Porém, antes do desenvolvimento foi feita a inserção de mais dois sensores além daqueles originais: um laser 2D no topo do robô e no mesmo referencial x e y deste, além de um giroscópio exatamente no mesmo referencial.

No código Python, para atuar nos objetos da cena foi necessário coletar os handles (referências) do robô, rodas e sensores. A localização ground truth é observada diretamente do cliente, não sendo necessário nenhum cálculo para tal. Entretanto, para cômputo da odometria foram utilizadas as equações 1, 2 e 3. Em linhas gerais, esses cálculos foram implementados usando listas que acumulavam os valores que tinham que ser adicionados as coordenadas x, y e θ , cabe salientar que foi necessário realizar a conversão dos ângulos dos encoders dos motores para representar valores entre 0 e 2π , ao invés π e $-\pi$. Foi tentado utilizar um giroscópio para melhorar a odometria, porém devido a possivelmente um problema da comunicação cliente-servidor os valores apresentados do giroscópio mostraram-se pouco consistentes. A odometria foi comparada com o ground truth verificar se os cálculos estavam sendo obtidos de forma coerente.

O mapeamento através de nuvem de pontos foi feito utilizando-se dois tipos de sensores e duas lógicas de cálculo. Foram utilizados os sensores sonar e laser, e para calcular o valor da posição absoluta, no referencial inercial, foi utilizado como base o ground truth e a odometria; sendo que as mudança de coordenada local para global foram feitas com as equações 4, 5 e 6. Os mapas obtidos nas 4 configurações diferentes foram comparados com a cena do simulador.

COLOCAR SOBRE EXTRAÇÃO DE CARACTERÍSTICAS

IV. RESULTADOS E DISCUSSÃO

Nesta seção você deve apresentar claramente os resultados obtidos para os testes efetuados. Procure organizar os dados utilizando uma linguagem científica. Algumas opções são o uso de tabelas e gráficos, para que a compreensão seja fácil e rápida.

V. CONCLUSÕES

Nesta seção, faça uma análise geral de seu trabalho, levando em conta todo o processo de desenvolvimento e os resultados. Quais os seus pontos fortes? Quais os seus pontos fracos? Quais aspectos de sua metodologia de trabalho foram positivas? Quais foram negativas? O que você recomendaria (ou

não recomendaria) a outras pessoas que estejam realizando trabalhos similares aos seus?

+-----+

REFERÊNCIAS

- [1] R. Siegwart and Illah R. Nourbakhsh *Autonomous Mobile Robots*, 1st ed. Massachusetts Institute of Technology, 2004