

Utilização de APF para Trajetória e DFL para Controle de Trajetória do Turtlebot 3- Simulação no ROS

Jéssica Lima Motta¹, Flávio Fabrício Ferreira² e Takaaki Oda Junior³

Abstract—Artigo desenvolvido para a disciplina Tópicos de Engenharia Elétrica II (ENG735)- Robótica Móvel. Usou-se um robô diferencial- Turtlebot versão 3, no ROS (Robot Operating System) e com simulação de mapa no Gazebo e no RViz onde se pode observar a navegação do Turtlebot no mapa evitando os obstáculos e através do Rviz determinar a posição desejada para o robô se deslocar até esse ponto.

I. INTRODUÇÃO

Este artigo tratará da aplicação de Campos Potenciais Artificiais- APF (Artificial Potential Fields) para realizar a trajetória do robô Turtlebot 3 simulado no ambiente ROS (Robot Operating System) e da utilização da Linearização por Realimentação Dinâmica- DFL(Dynamic Feedback Linearization) para controle da trajetória, juntamente como o sistema de localização AMCL que faz parte do pacote de simulação do Turtlebot 3 no ROS.

II. REVISÃO BIBLIOGRÁFICA

A. Robô Móvel com Acionamento Diferencial

O robô com acionamento diferencial permite que suas rodas sejam acionadas independentemente para realizar um determinado movimento, pois para cada roda há um motor. Este tipo de robô realiza a trajetória não-holonômica, ou seja, possui certas restrições no movimento, como não permitir movimentação lateral que seja perpendicular a direção que o robô esteja, este é o caso do Turtlebot versão 3, usado na simulação.

B. Campos Potenciais Artificiais- APF

Este tipo de técnica é usada para a trajetória de um robô móvel e consiste em somar os potenciais repulsivos, que afastará o robô dos obstáculos, aos potenciais atrativos, que colocará o robô na trajetória que o mesmo deve seguir para alcançar a localização desejada/alvo, ilustrado na Figura 1 [3]. O planejamento da trajetória acontece a cada iteração, ou seja, o CPA define a trajetória a cada movimentação do robô na direção desejada fazendo os devidos ajustes mesmo sem ter o mapa do ambiente *a priori*, ou seja, permite a identificação de obstáculos on-line.

As equações que descrevem os cálculos para encontrar o campo potencial de atração, de repulsão e o campo potencial total estão descritas das equações 1, 2 e 3. [1].

A equação 1 demonstra que o campo potencial é formado a medida que o robô é atraído para a posição final enquanto está sendo repelido pelos obstáculos, por isso ocorre a soma do campo potencial de atração, gerado pela posição final, com o campo potencial de repulsão:

$$U(x_d, y_d) = U_{att}(x_d, y_d) + U_{rep}(x_d, y_d). \quad (1)$$

Para o cálculo do campo potencial de atração a equação 2 mostra que será o quadrado da diferença da posição de referência, x_{ref} e y_{ref} , é a localização no espaço em que se deseja que o robô alcance, pela posição do robô no instante da verificação x_d e y_d . O β é um fator escalar/ganho positivo

$$U_{att}(x_d, y_d) = \frac{\beta}{2}((x_{ref} - x_d)^2 + (y_{ref} - y_d)^2) \quad (2)$$

Os campos potenciais de repulsão, calculados pela equação 3 manterão o robô a uma distância, definida em código, dos obstáculos encontrados no ambiente.

$$U_{rep}(x_d, y_d) = \sum_{i=1}^j U_{rep_i}(x_d, y_d) \quad (3)$$

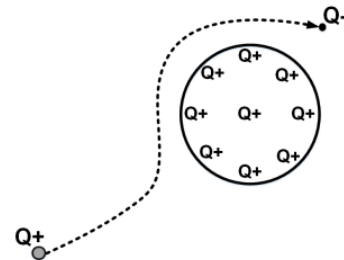


Fig. 1. Trajetória Resultante do Uso do Método APF

O método APF possui uma limitação que é o mínimo local que, dependendo da configuração inicial do robô, onde ele se encontra no mapa, os parâmetros de repulsão e atração escolhidos e a forma como os obstáculos se encontram no ambiente, a trajetória planejada pode não alcançar a referência, mesmo havendo um caminho factível [2].

C. Controle de Trajetória

Existem vários métodos usados para realizar o controle cinemático, ou também chamado de controle de trajetória, de um robô móvel diferencial, que é o caso do Turtlebot versão 3. Foi escolhido então o DFL- Linearização por Realimentação Dinâmica, que tem por objetivo atingir a linearização do modelo cinemático do robô a cada mudança de estado que ocorra, dessa forma a linearização é a todo momento com o objetivo de percorrer a melhor trajetória.

Nas Equações 4 e 5 estão descritas a lei de controle do DFL, onde tem-se na Equação 4 a velocidade linear do robô e na Equação 5 tem-se a velocidade angular do robô.

$$v_s(k) = v_0 + a(k).t_s. \quad (4)$$

$$w_s(k) = -(u_1(k)/v_0) \cdot \sin(\theta(k)) + (u_2(k)/v_0) \cdot \cos(\theta(k)) \quad (5)$$

Nas Equações 6 e 7 estão descritas a forma usada para calcular a aceleração linear no eixo x (u_1) e no eixo y (u_2), que somadas resultará na aceleração linear a na Equação 8.

$$u_1(k) = \Delta^2 x_d(k) + K_{p1} \cdot (x_d(k) - x(k)) + K_{d1} (\Delta x_d(k) - \Delta x(k)) \quad (6)$$

$$u_2(k) = \Delta^2 y_d(k) + K_{p2} \cdot (y_d(k) - y(k)) + K_{d2} (\Delta y_d(k) - \Delta y(k)) \quad (7)$$

$$a(k) = u_1(k) \cdot \cos(\theta(k)) + u_2(k) \cdot \sin(\theta(k)) \quad (8)$$

Onde cada variável significa:

- v_0 - Velocidade linear do robô no instante anterior.
- a - aceleração linear resultante
- u_1 - aceleração linear horizontal- eixo x
- u_2 - aceleração linear vertical- eixo y
- K_{p1}, K_{p2}, K_{d1} e K_{d2} - parâmetros de controle

Na Figura 2 [3] está o diagrama de blocos do sistema de controle, no qual o controlador, no caso o DFL, atuará de acordo com o erro identificado que será a diferença entre os pontos de referência enviados pelo cálculo da trajetória: x_d e y_d que determinam a posição e θ_d que determina a orientação para o qual o robô deverá se deslocar e x, y e θ que é o ponto executado pelo robô, então o controlador fará os ajustes enviando como sinal de controle a velocidade linear v_s e a velocidade angular w_s para o robô corrigir a trajetória.

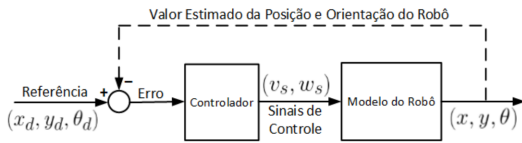


Fig. 2. Diagrama de Blocos do Sistema de Controle

D. AMCL

AMCL- Adaptive Monte Carlo Localization que é um sistema de localização probabilística no qual o robô se desloca em 2D. O AMCL é a adaptação do sistema de localização Monte Carlo que é amplamente aplicado a robôs móveis.

III. DESENVOLVIMENTO

Para o sistema de localização do Turtlebot 3 foi usado o AMCL que faz parte do pacote do Turtlebot 3 e foi usado na simulação. Os algoritmos de geração de trajetória- APF e controle de trajetória- DFL foram desenvolvidos em C++,

TABLE I
PARÂMETROS DEFINIDOS PARA O CONTROLADOR DFL

Parâmetros	Valor Definido
K_{p1}	1
K_{p2}	1
K_{d1}	0.5
K_{d2}	0.5

compatível com o ROS para que pudesse simular a aplicação desses métodos no Turtlebot 3.

Os valores dos parâmetros de controle definidos para o controlador DFL estão descritos na Tabela I.

IV. RESULTADOS- SIMULAÇÃO

A simulação foi realizada no ROS e na Figura 3 é possível verificar o Turtlebot 3 no ambiente Gazebo inserido no mapa no qual ele já fez a varredura (scan) identificando todos os obstáculos e evitando-os durante sua trajetória através do algoritmo do APF, onde o robô, os obstáculos e o ponto de destino estão com a configuração apresentada na Figura 4.

Na Figura 5 mostra a posição de destino escolhida que o robô deverá alcançar e a trajetória que ele deverá percorrer, identificada como uma trajetória factível pelo APF, no ambiente RViz.

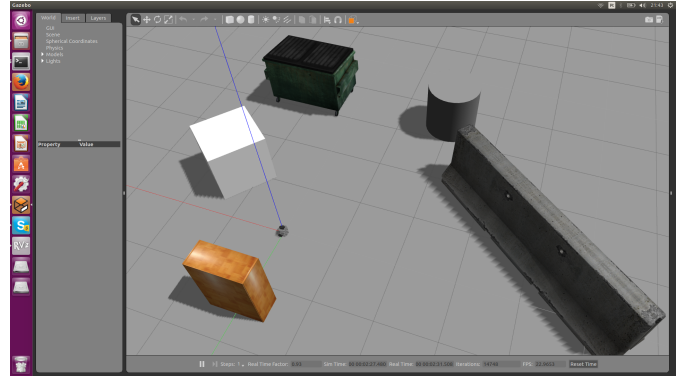


Fig. 3. Turtlebot 3 no Gazebo

Através do algoritmo do APF é possível identificar os obstáculos como campos potenciais que possuem a mesma carga do robô, logo se repelem, e ponto de referência como uma carga negativa logo irá atrair o robô para este ponto. Na Figura 6 está apresentada a trajetória que o robô irá realizar para alcançar o ponto de referência evitando o obstáculo.

A Figura 7 exemplifica uma situação de mínimo local, que é uma limitação já citada do método APF, e por esse motivo o robô não consegue traçar uma trajetória factível até o ponto de referência.

V. CONCLUSÃO

Os métodos empregados para geração de trajetória- APF e controle de trajetória- DFL apresentaram resultados satisfatórios como puderam ser observados na simulação no ROS, Gazebo e RViz. Logo, o APF e o DFL são opções que atendem a robôs diferenciais e podem ser empregados em robôs diferenciais reais.

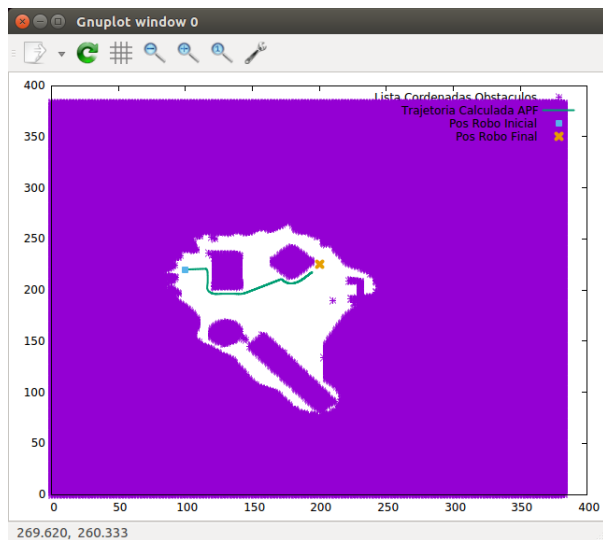


Fig. 4. Trajetória Definida a Partir do APF

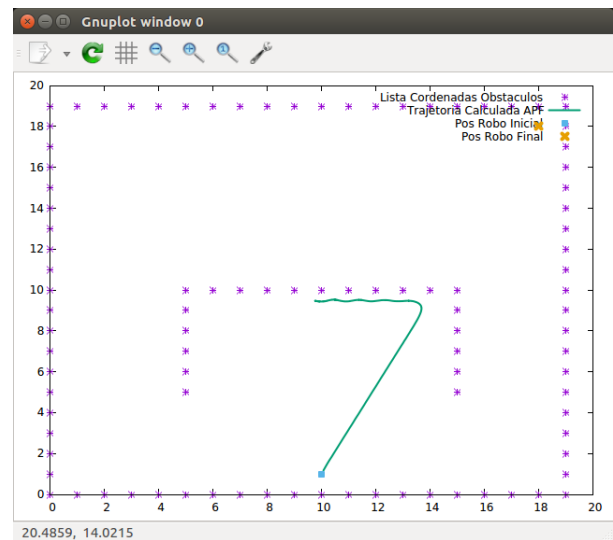


Fig. 7.

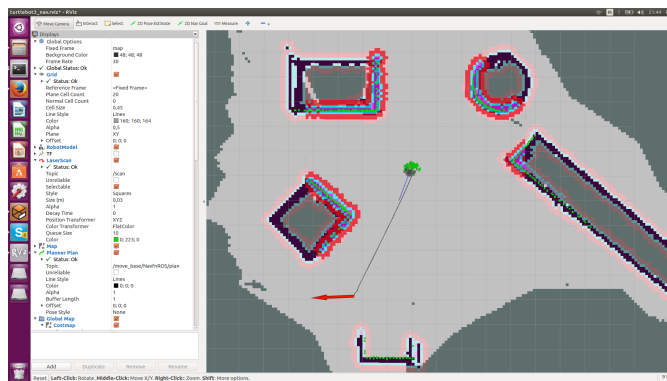


Fig. 5. Turtlebot 3 Realizando Trajetória no RViz

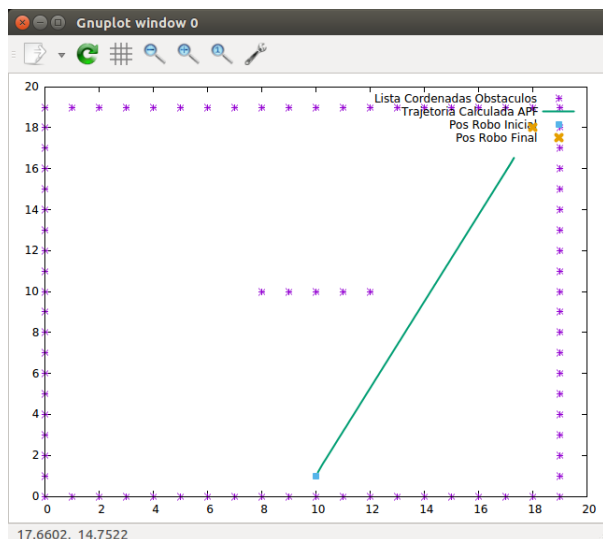


Fig. 6. Turtlebot 3 no Gazebo

na execução dos testes no ROS.

REFERENCES

- [1] Latombe, Jean-Claude. Robot motion planning. Vol. 124. Springer Science & Business Media, 2012.
- [2] Choset, Howie M. Principles of robot motion: theory, algorithms, and implementation. MIT press, 2005.
- [3] CERQUEIRA, TADEU ABREU. "Navegação De Veículos Diferenciais Baseada Em Campos Potenciais Artificiais."
- [4] Fox, Dieter, et al. "Monte carlo localization: Efficient position estimation for mobile robots." AAAI/IAAI 1999.343-349 (1999): 2-2.

AGRADECIMENTOS

Agradecemos aos profs. Dr. André Scolari e Dr. Paulo César Farias pelo suporte dado na produção deste artigo e