

Planejador de Rota baseado na Geometria de Manhattan

Álvaro de Almeida¹¹, Wesley Simmer¹¹, Wagner da Trindade¹¹, Eduardo Amaral²¹

¹Aluno do Curso Técnico em Informática, ²Orientador

¹Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo (IFES)

Rodovia ES 010, Km 6.5, Manguinhos – 29173-084 – Serra – ES – Brasil

alvaro_otto7@hotmail.com, icbm77@gmail.com, wagnercofcof@ig.com.br,
eduardomax@ifes.edu.br

Abstract. *This paper describes a system of generating route for a differential robot based in heuristic of the Manhattan geometry. The experimental results demonstrated that the proposal is viable and can be used in differential robots with restrictions of movement and sensing.*

Resumo. *Este artigo descreve um sistema de geração de rota para um robô diferencial baseado na heurística da geometria de Manhattan. Os resultados experimentais demonstraram que a proposta é viável e pode ser utilizada em robôs diferenciais com restrições de movimento e sensoramento.*

1. Introdução

O desenvolvimento da tecnologia necessária para robôs autônomos engloba vários desafios. Entre eles, encontra-se o planejamento de rotas. A ideia para o planejamento de rotas é deixar o robô sempre se mover em um caminho de custo mínimo até o alvo, sendo este caminho o mais seguro possível.

Este artigo apresenta um sistema de geração de rota para ser utilizado por um robô com acionamento diferencial. Este sistema é baseado na decomposição do ambiente em células exatas e na heurística da geometria do táxi. O objetivo é analisar a eficiência dessas técnicas de navegação de robôs autônomos no contexto de um robô com restrições de movimento e de sensoramento.

2. O robô móvel e suas arquiteturas e o sistema de navegação

O robô móvel utilizado neste trabalho foi desenvolvido em [Amaral, Pin e Oliveira 2012]. É um robô diferencial contendo uma plataforma com servomotores, placa Arduino e um sonar. As funções de movimento disponíveis no robô são: “frente”, “ré”, “pare”, “direita 90°” e “esquerda 90°”. O robô não possui odometria.

A arquitetura física é composta por um controlador (notebook) interligado, pela porta USB, a um Arduino, que por sua vez, está ligado a dois atuadores (servos) pelas portas digitais de sua placa. Já a arquitetura do sistema define a interligação de seus módulos lógicos. A modelagem do espaço e o planejador e gerador de rota foram implementados na linguagem de programação Processing, instalada no notebook. Já o acionamento do robô foi implementado na própria IDE do Arduino.

Neste trabalho, o mapa é previamente construído e para isso utilizou-se o método de decomposição em células. Este método consiste em dividir o espaço livre do robô em

regiões simples (células) [Souza 2008]. O planejador de rota desenvolvido no projeto foi baseado no método de Manhattan. Dados dois pontos “a” e “b”, em um sistema de coordenadas cartesianas fixo, a distância Manhattan é definida como: $d(a,b) = |X_a - X_b| + |Y_b - Y_a|$ [Thompson e Dray 1999].

No início do algoritmo, uma matriz é construída e o usuário define, com o uso do mouse, onde estão os obstáculos e, na sequência o local de destino. O robô possui pré-definidas a origem na célula 0,0 e a orientação, no plano cartesiano, a 270° do eixo x. O planejador de rota então, calcula quantas células o robô terá que percorrer para chegar ao destino, definido pelo usuário. Seu movimento é feito em L, utilizando a heurística de Manhattan. Se houver obstáculo em alguma célula na trajetória, o robô desviará utilizando um padrão preestabelecido, e somente retornará a uma linha ou coluna já percorrida se esse retorno for fundamental ao atingimento da célula destino. Isso garante uma rota com menor consumo de energia e menor de erro acumulado.

3. Experimentos, resultados e considerações finais

Foram executados 30 testes em um ambiente controlado que consistia em uma matriz de 10x10 células com 40cm x 40cm por célula, feitas no chão usando fita isolante, conforme vídeo disponível em (http://www.youtube.com/watch?v=0_Dk5YCQPac). Em cada teste foi medida a distância do centro do robô ao centro da célula de destino (erro linear), e o desalinhamento do robô no momento do término da rota (erro angular).

Após os testes, a média do erro linear ficou em 30,3 mm. Já a média do erro angular ficou em 6,1°. O robô executou sempre a rota de menor custo, considerando suas limitações de movimento. Ocorreram erros na trajetória devido à deformação axial das rodas dianteiras gerada pelo peso do notebook. A roda traseira também ocasionou erros de trajetória durante as curvas de 90° devido à falta de atrito. Porém, a taxa de erro foi satisfatória. Testes precisam ser feitos em ambientes de maior dimensão, pois o erro acumulado pode inviabilizar o método.

Neste trabalho foi apresentado um sistema de geração de rota para ser utilizado por um robô diferencial. Dentre as técnicas de navegação de robôs, o método escolhido foi considerado adequado às características e limitações do robô utilizado no projeto. Além disso, o método também se mostrou eficiente na resolução do problema de locomoção de um autônomo em ambiente com obstáculos. Em projetos futuros deve-se implementar o uso de sensores e de outros métodos de navegação.

4. Referências

- Amaral, E. M. A., Pin, L. B., Oliveira, D. (2012) “LUDEbot: Uma plataforma com Servomotores e Sensoriamento utilizando a Placa Arduino e Controle Externo”. In: III Workshop on Educational Robotics, IEEE SBR/LARS/WRE, Fortaleza, CE.
- Souza, S. C. B. (2008) “Planejamento de trajetória para um robô móvel com duas rodas utilizando um algoritmo A-Estrela modificado”. Dissertação de mestrado, Programa de Engenharia Elétrica, UFRJ/COPPE, Rio de Janeiro, RJ.
- Thompson, K., Dray, T. (1999) “Taxicab Angles and Trigonometry”. Oregon State University, Corvallis, OR 97331.