

Primeiro Trabalho de FCR

Nome: Tulio Mariano da Silva Lima
Matrícula: 12/0054337

Objetivos

O robô pioneer deve se deslocar pelo prédio no simulador STAGE. A posição inicial pode ser dada pelo usuário ou pelo simulador. A posição do robô a cada instante de tempo também é conhecida (fornecida pela odometria ou por um sistema de localização). O usuário deve especificar uma posição qualquer no mapa do prédio, em coordenadas do eixo global e o robô deve se deslocar para esse ponto, sem tocar nenhum objeto que ele consiga detectar: paredes, móveis, pessoas, etc.

Introdução

A implementação de uma rota até um objetivo pode envolver uma série de barreiras a serem superadas até a chegada ao destino, e pode ainda acarretar em perceber as alterações em um ambiente que é dinâmico. Embora pequena, a frase acima é um desafio muito grande no campo da robótica e pode ter inúmeras soluções, nas quais os resultados são discutíveis, e o foco desse trabalho é achar pelo menos uma forma de 'resolver' tal problema.

A plataforma usada na atividade será um ambiente simulado, contendo um robô pioneer com uma série de sensores que auxiliam na percepção do mundo simulado e visível pelo Stage. O robô deve ser orientado durante sua trajetória, e não deve encostar em nenhum objeto, podendo ser atribuído um limite de distância para o qual pode permanecer, evitando a todo custo colisões.

Partir de um ponto até outro envolve uma estratégia, nessa estratégia é usado o que se tem disponível ou que pode ser alcançado, ou seja, para sair de um ponto até outro é possível usar um mapa do ambiente, um grafo que simboliza caminhos, um sistema de coordenadas ou mesmo nada. Ambos com implementações diferentes, mas apenas o que envolve o uso de um sistema de coordenadas será o foco por enquanto, dessa forma o usuário deverá passar coordenadas em x e y para que o programa encontre e percorra o caminho calculado.

Aparato Experimental

Foram usados nesse trabalho:

- Um computador com ROS configurado
- Python (produção do programa que orquestra a movimentação e orientação)
- Stage e Rviz (para acompanhamento do robô)

Metodologia

Dado o objetivo descrito no roteiro de trabalho, foi desenvolvido um programa em python que atende as propostas da atividade. Nesse programa, o usuário é questionado pelas posições em x, y e angulo para os quais ele deseja encaminhar o robô.

```
tuliolima@tuliolima-VPCSB35FB:~/catkin_ws$ rosrn fcr2017 roteiro3.py
Iniciando o Programa para mover o Pioneer
Posicao:
('X: 24.4360095583', 'Y: 51.4475629548', 'Graus: 1.14591559026')
Qual a posicao final em x desejada?30
Qual a posicao final em y desejada?51.5
Qual o angulo(Graus) final desejado?0
(5.563990441676388, 0.05243704515365266, -1.1459155902616376)
```

De posse desses dados, o programa calcula em qual direção o destino se encontra e qual a distância este se encontra do tal objetivo, após fazer esse cálculo o robô é orientado na direção na qual deve seguir, e para isso uma velocidade de 0.1 foi fixada para rotações em z e que pode ser negativa ou positiva(a depender do ângulo). Em seguida, e considerando que o robô se considera orientado, tudo está pronto para o robô seguir em frente e aqui começa a rotina de movimentação com velocidade fixa de 0.2 em x. E assim o movimento segue até que um objeto é encontrado a uma distância menor que 0.5, na qual o programa realiza uma parada no robô e faz uma curva de 90° a direita para depois andar 0.2 a frente. Depois disso o robô tenta encontrar o ponto novamente, o quanto possível.

Um simples desvio para direita não é suficiente para salvar o robô de uma colisão um objeto, e por isso esse comportamento é implementado na forma recursiva, até que o robô encontre o seu destino. Ou seja, uma vez começado a movimentação o robô anda até encontrar com um objeto ou com seu objetivo, caso um objeto seja encontrado ele deve desviar para direita e em seguida movimentar-se novamente até achar um novo objeto ou o seu destino, isso é repetido até quando ele encontrar com o seu objetivo. Uma vez no destino o robô está pronto para alinhar com o ângulo final oferecido pelo usuário e assim terminar a sua rotina de trabalho.

Essa implementação recursiva que fez uso de três funções que são basicamente reorientar(afastando-se dos objetos), orientar(alinhar com o eixo que leva até o ponto final) e movimentos(movimentar-se na direção orientada)

Dados experimentais e Resultados

Como entrada, o dado deve ser fornecido deve ser um ponto contido e acessível dentro das coordenadas do mapa do prédio disponível no launch stage disponibilizado em (github.com/fcr2018). Para executar o programa, basta ter a cópia do arquivo '.py', inserir essa cópia na pasta src do projeto fcr2018 e executar o comando rosrún (ex: rosrún fcr2017 roteiro3.py).

Uma vez executado o programa mostra a sua posição dentro das coordenadas, que normalmente é o ($x = 0$, $y = 0$), e pedirá pela posicao final desejada para x, y e o angulo que o robô deve ficar ao final da execução, assim como na imagem detalhada na metodologia. Como saída o programa tem apenas os comandos que envia para os tópicos de velocidade e uma mensagem de "Alinhado" ao final execução.

O comportamento do algoritmo pode ser observado no vídeo disponibilizado junto do arquivo desse simples relatório.

Conclusão

Dado a característica de não ter uma solução única para o problema, uma simplificação foi adotada, e justamente a que encaminha o robô para a direção a sua direita, da mesma forma que poderia ser direcionado para a esquerda. Isso mostrou-se útil para desviar de objetos no meio do caminho, como mostrado no vídeo e passível de observação na execução do código, ilustrando o sucesso da solução. A solução que envolve a recursividade pode acarretar em um estouro de memória e merece uma atenção melhor caso melhorias sejam implemetadas, porém nenhum comportamento desse tipo foi observado durante os testes.

Em um teste rápido, foi possível observar que o programa se comportou bem com paredes. De forma que, se o destino se encontra logo após uma curva e atrás de uma parede(seja para direita ou esquerda), ainda assim o robô alcançou o objetivo.