



Universidade Federal de Minas Gerais
Departamento de Ciência da Computação

Introdução à Robótica

Trabalho Prático 3

Controle

Grupo 7:

Celina Gomes
Lucas Frossard
Ludmila Villela

Professor:

Mário Fernando Montenegro Campos

Belo Horizonte, 13 de novembro de 2008

1 Objetivos

Familiarizar o aluno com sensores ópticos e de toque, além de técnicas básicas de controle.

2 Etapas do desenvolvimento

- **4 de novembro:**

Na primeira reunião ocorreu a familiarização com os sensores, a sensibilidade e os valores retornados para cada detecção. Foi também estabelecido um planejamento do programa, selecionou-se as funções do tp2 que seriam uteis e deveriam ser mantidas.

- **10 de novembro:**

Foram colocados todos os sensores no Johnnie (ver fotos abaixo). Soldou-se os fios no sensor e os mesmos foram colados com cola quente em peças de lego. Além disso, implementou-se os algoritmos de ajuste de posição e detecção da luz indicadora de início da tarefa.

- **11 de novembro:**

Foi implementado os algoritmos referente a tarefa do shift-encoder e wall-follow. Alguns testes foram feitos, mas o funcionamento das tarefas ainda não estavam de acordo com o esperado.

- **12 de novembro:**

Foram feitos testes e validações para todas as tarefas. Alguns ajustes foram necessários para a tarefa de encoders, como as constantes de proporcionalidade e diferencial.

3 Estrutura Física

A estrutura física do robô foi adaptada para o uso de sensores. Foram inseridos 3 sensores de toque, 1 óptico ativo e 1 LDR.

Um LDR foi colocado na parte inferior central do robô para a tarefa de detecção da luz. Um sensor óptico ativo foi colocado na lateral do robô para realização da tarefa de seguir a parede. Um sensor de toque foi colocado na parte traseira direita, outro na parte traseira esquerda e outro na parte frontal do robô para a tarefa de orientação.

A estrutura do robô foi reduzida nas laterais de forma que este ficou menor e mais compacto. Porém ele manteve os 2 motores, cada um ligado a uma roda do eixo traseiro de forma independente. Também foi mantido o roll on na parte frontal.

Utilizou-se a calibragem do trabalho anterior para determinar qual potência que cada motor deveria receber para que o mesmo andasse reto e fizesse uma rotação de 30, 60 e 90 graus em torno do próprio do eixo.

A Handy Board foi colocada numa estrutura feita para acomodá-la no topo do Johnnie.

4 Estrutura Lógica

4.1 Estrutura de Dados

Utilizou-se as mesmas estruturas de dados do trabalho anterior, Motor e Config. A primeira, Motor, é a abstração de um motor, a qual armazena a porta a qual ele se encontra ligado, bem como sua potência atual. Já a segunda estrutura, Config, possui a configuração de uma trajetória qualquer. Ela armazena a potência desejada de cada motor, além do tempo em que essas potências devem ser mantidas.

4.2 Módulos

O código do trabalho foi dividido em vários módulos, cada um relacionado a uma tarefa ou funcionalidade específica. O método main está contido no arquivo main_tp3.ic, bem como todos os “includes”.

- **main_tp3.ic:** Além de conter o main, esse módulo é responsável pela impressão do menu principal, bem como pela interação com o usuário. Assim que uma opção do menu é escolhida, uma função do módulo correspondente é chamada. Essa função é sempre no formato “startX”, sendo X a tarefa escolhida.
- **motor.ic:** Possui métodos associados aos motores. Foram criadas funções para ajustar uma nova potência de um motor e de dois de forma gradativa, assim como para desligar o mesmo. Dessa forma, evita-se mudanças abruptas de velocidade. Além disso, foi criada uma função execute(), a qual é responsável por executar uma tarefa com uma potência “x” para o motor direito, uma potência “y” para o esquerdo em um intervalo “t” de tempo.
- **wallFollow.ic:** Dentro deste, encontra-se a função de seguir a parede. Utilizando as medidas do sensor óptico para estimar a distância à parede, foi implementado um controle realimentado para que o robô se mantivesse paralelo à parede. É feito o controle de parada da tarefa via o botão “stop”.
- **start.ic:** O módulo é responsável pela realização da tarefa de detecção do instante em que a luz na base do campo é acesa, através das medições do sensor LDR.

- **encoders.ic:** Fazendo uso dos arquivos binários “senchr6.icb” e “senchr5.icb” e de sensores break-beam para construção de encoders, o respectivo módulo é responsável pela implementação de um controlador PD para fazer com que o robô siga uma linha reta. Com os valores das medidas dos encoders de cada roda, foi utilizada a rotina pronta que nos devolve a velocidade de cada roda. Mantendo a velocidade da roda direita como o setPoint o algoritmo atua proporcionalmente entre a diferença de velocidade entre as duas rodas e diferencialmente entre a diferença de velocidade da última medida da roda esquerda e da atual, sempre atuando somente na roda esquerda. O módulo dá a opção de se estabelecer um tempo de realização para a tarefa e qual a potência considerada como setPoint. Caso não seja escolhida um tempo para realização da tarefa o controle de parada é feito via o botão “stop”.
- **findPosition.ic:** Esse módulo é responsável pela tarefa de orientação com sensores de toque.

Além disso, os menus de cada tarefa são gerenciados pelo módulos associados às mesmas.

4.1 Usabilidade

A interação do usuário com o robô foi ajustada para ser feita por meio de um menu, do potenciômetro e dos botões “stop” e “start”. Cada botão tem as seguintes funções:

.start: Sempre utilizado para confirmar/executar a opção impressa no display.

.stop: Utilizado para alternar entre as opções quando no menu, para cancelar o ajuste de alguma variável durante a fase de calibragem e para abortar as tarefas a qualquer momento.

.potenciômetro: Utilizado para setar os valores associados à variáveis no menu de configuração, tempo e potência.

O menu inicial possui 6 opções:

- 0- StartLight:** Inicia a tarefa de detecção do instante em que a luz é acesa.
- 1- Encoder:** Execução da tarefa de controle das rodas para seguir em linha reta.
- 2- Wall Follow:** Executa a tarefa se seguir a parede.
- 3- Set Position:** Chama a tarefa de orientação com sensores de toque.
- 4- Test Sensor:** Permite verificar as medidas dos sensores

5 Desafios

Encontramos os seguintes desafios nesta etapa do trabalho:

- A calibragem não era exata e a variação da carga da bateria contribuía para que esta variação se tornasse maior de maneira que uma calibragem que servia num determinado momento não serviria para um momento posterior.

6 Conclusão

A tarefa de calibrar é difícil e trabalhosa. Várias medições devem ser feitas até ter certeza de que determinados valores são os adequados para uma determinada tarefa. Ainda assim, estes valores podem quando usados diversas vezes podem apresentar diversos resultados diferentes, com uma variação significativa entre os resultados extremos. Estes resultados também variam de acordo com a carga da bateria o que torna a tarefa ainda mais complicada.