



Implementação de Comportamentos Reactivos no Robô *Insónia*

Filipe Cabral Pinto, Pedro Bento Aires, Miguel Aragão, Jorge Aragão

ISR/DEE – Universidade de Coimbra

I. INTRODUÇÃO

Um dos grandes problemas no controlo de robôs é a detecção e desvio de obstáculos em tempo real. A criação de algoritmos para solucionar este problema depende da variedade de sensores utilizados no robô.

A navegação autónoma de um robô prende-se com o desvio de obstáculo em tempo real e com o alcance de um objectivo. Esse objectivo pode ser, por exemplo, um farol a emitir um feixe de luz infravermelha ou uma determinada localização no espaço.

O trajecto que o robô descreve depende do ambiente que o rodeia. No caso de este ser conhecido é possível planear uma trajectória óptima para, a partir da posição em que o robô se encontra, chegar ao seu objectivo, ou seja, há um planeamento global do caminho a percorrer pelo robô. No caso de um ambiente desconhecido ou em mudança aleatória o movimento baseia-se na informação sensorial e em tempo real de que o robô dispõe. Estamos, neste caso, perante um planeamento local da trajectória.

II. NAVEGAÇÃO LOCAL

A navegação local de robôs móveis envolve o desvio de obstáculos em tempo real e o alcance de uma meta.

Existem diversos métodos para o planeamento local do trajecto a seguir pelo robô.

O INSÓNIA actualmente é constituído por seis sensores de infravermelhos a 40 KHz que permitem ao robô fazer o desvio de obstáculos, sete sensores de infravermelhos a 32 KHz utilizados para seguir a luz emitida por um farol, dez bumpers que nos indicam se houve contacto com um objecto e dois fototransistores para detectar a chegada à meta. A electrónica deste robô está distribuída por duas placas de circuito impresso principais: a Mekatronics Robot Controller, MRC11, que permite a execução de programas no robô e a MRX01 que é uma placa de expansão e que fornece os circuitos que suportam a parte relativa aos motores (dois servos alterados que são controlados de uma forma totalmente independente) e sensores (os sensores foram soldados em placas de circuitos impressos de maneira a diminuir a complexidade provocada pelo excesso de fios). Existem ainda pequenos circuitos tais como um sistema de paragem/arranque, uma ligação série e ainda um conversor DC/DC.

Como se pode verificar, o robô não é detentor de qualquer sensor indicador da sua posição, impossibi-

litando desta forma a concretização de qualquer um dos modelos explicados em [1], [2] e [4].

A nossa opção recaiu no método descrito por Brooks em [3] que permite um desenvolvimento modular de um conjunto de comportamentos extremamente simples e independentes que permitem levar a cabo um modelo coerente e eficaz de navegação local.

III. SUBSUMPTION ARCHITECTURE

O modelo desenvolvido por Brooks, denominado Subsumption Architecture, tem grandes vantagens: a sua estrutura é concebida de uma forma bastante simples e eficaz; necessita de baixos requisitos computacionais, pois os comportamentos, de uma forma genérica, não implicam grandes cálculos; não requer grandes memórias devido à inexistência de modelos do mundo; pode ser melhorado de uma forma incremental, o que significa que novas camadas podem ser adicionadas sem que a estrutura do programa sofra grandes alterações.

A ideia subjacente a esta estratégia é a de ter um conjunto de tarefas independentes, que correm em paralelo, implementando cada uma delas um comportamento. Cada comportamento, num determinado instante, pode estar activo ou desactivo. Quando está activo compete com os outros comportamentos no ataque aos motores. No estado desactivo não entra em competição para o controlo dos motores. Obviamente, em cada instante, apenas um desses comportamentos prevalece. A selecção é feita através de um esquema de prioridades. Os comportamentos de mais alto nível, isto é, de maior prioridade, têm a capacidade de suprimir os de menor nível. A comunicação entre os diferentes processos é feita recorrendo a variáveis globais.

O objectivo é pôr o robô a movimentar-se num ambiente desconhecido e em mudança, em direcção a um farol que emite luz infravermelha, desviando-se dos obstáculos.

O farol está situado em cima de um material de cor preta, onde o robô deverá parar ao chegar. O INSÓNIA nem sempre vê o farol, pois este pode estar tapado por obstáculos que impeçam a passagem dos feixes de luz infravermelha.

Caso haja um choque com um obstáculo é necessário que o robô consiga recuperar dessa situação e volte a direccionar-se para o farol.

A solução encontrada foi a criação de seis comportamentos distintos. São eles: Fim_de_Curso, Bateu, Contorna, Evita, Segue_Farol e Forward.

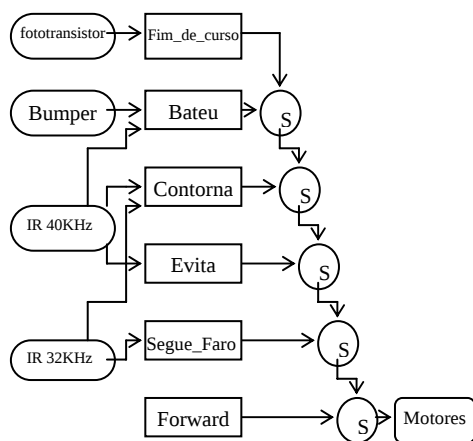


Fig. 1 - Diagrama de blocos

No esquema apresentado em cima, as caixas arredondadas representam tanto os sensores como os actuadores e todo o SW necessário para os controlar. As caixas rectangulares representam os diferentes comportamentos. O símbolo S representa um nó de supressão. Este S significa que o comportamento situado num nível superior, quando está activo, tem prioridade sobre todos os outros situados a um nível inferior ao seu, ou seja, o comando para os actuadores é o do nível superior

IV. COMPORTAMENTOS

A. Forward

O objectivo deste comportamento não é mais do que o de impedir a paragem do robô. Assim, se todos os outros comportamentos estão desactivos, o INSÓNIA desloca-se em frente.

B. Segue_Farol

O Segue_Farol é o comportamento que possibilita que o INSÓNIA se desloque até ao farol.

Para que o Segue_Farol se torne activo é necessário que o robô consiga visualizar o farol e que estes não se encontrem muito afastados

C. Evita

Este comportamento visa impedir que o robô colida com os obstáculos existentes na sua área de trabalho à medida que este se desloca em direcção ao seu alvo.

D. Contorna

O comportamento Contorna tem como finalidade fazer com que o robô siga ao longo de uma parede, tanto pela sua esquerda como pela sua direita, mantendo uma distância mínima de segurança.

E. Bateu

Para que este comportamento se torne activo basta que um dos bumpers seja pressionado, pois isto significa que o robô bateu em algum obstáculo ou que algum objecto em movimento colidiu com ele.

F. Fim_de_curso

O farol situa-se em cima de um material de cor preta. É lá que o INSÓNIA deve parar ao aproximar-se do farol. Para detectar o material preto o robô dispõe de dois sensores direccionados para o chão.

G. Escalonador

O papel do escalonador é fazer a selecção do comportamento que vai actuar sobre as rodas. No fundo o escalonador é a implementação prática do nó de supressão. A escolha é feita tendo em atenção as prioridades de cada um dos comportamentos e o seu estado. Comportamentos que estejam inactivos não competem para o controlo dos motores. O comportamento situado na camada superior é o mais prioritário. A prioridade de cada um deles decresce à medida que se vai descendo nas camadas.

V. CONCLUSÕES

O INSÓNIA demonstrou ser uma poderosa ferramenta para o teste de diversos comportamentos reactivos. A diversidade dos seus sensores possibilita o movimento do robô em ambientes desconhecidos, evitando os obstáculos, até ao alcance do seu objectivo. Foi possível implementar com sucesso os comportamentos de seguimento de um farol a emitir um feixe de luz infravermelha, desvio de obstáculos, contorno de paredes e paragem do robô ao detectar no chão um material de cor preta. Através de uma conjugação destes, o robô ficou capacitado para responder de uma forma eficaz à fuga de situações críticas provocadas por determinadas configurações de obstáculos denominadas de armadilhas. Conjugando todos estes comportamentos foi possível implementar com sucesso o método desenvolvido por Brooks denominado Subsumption Architecture.

BIBLIOGRAFIA

- [1] J. Borenstein, Y. Koren, "Real-Time Obstacle Avoidance for Fast Mobile Robots", IEEE Transactions on Systems, Man, and Cybernetics, Vol 19, Setembro/Outubro, 1989.
- [2] Filipe Cabral Pinto e Pedro Bento Aires, "Implementação de Comportamentos Reactivos em Robôs Móveis", Universidade de Coimbra, 1998.
- [3] R. A. Brooks, "A Robust Layered Control System for a Mobile Robot", IEEE Journal of Robotics and Automation, 1986.
- [4] A. Elfes, "Sonar-Based Real-World Mapping and Navigation", IEEE, Junho, 1987.