



Universidade do Porto
Faculdade de Engenharia

FEUP

Ricochet Robots

Relatório Intercalar

Inteligência Artificial

3º ano do Mestrado Integrado em Engenharia Informática e Computação

Elementos do Grupo:

Carlos Miguel da Silva Pereira – 201200678– ei12078@fe.up.pt

Luis Torres Aguiar da Costa – 201203872– ei12008@fe.up.pt

15 de abril de 2016

Objetivo

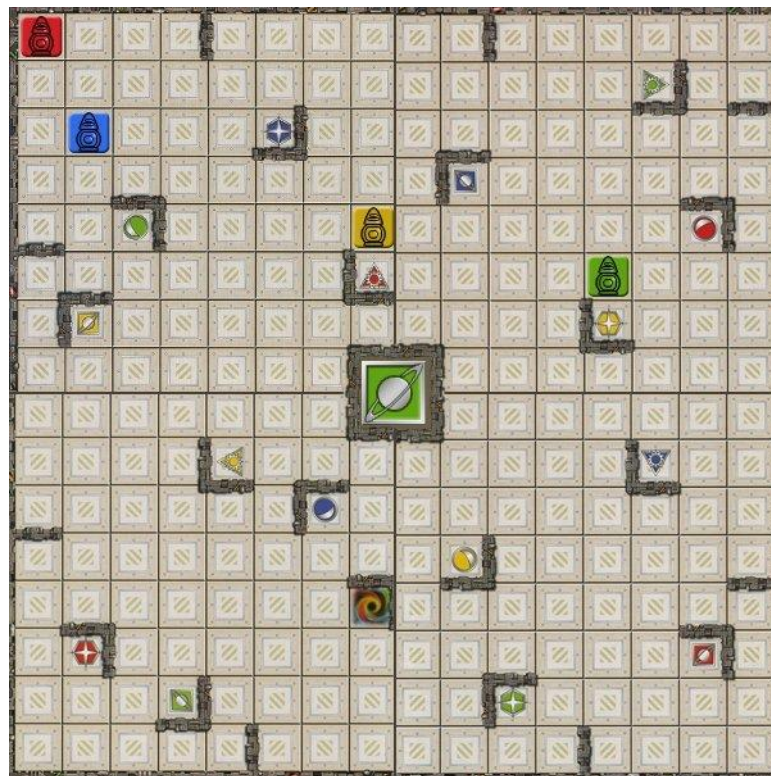
O trabalho *Ricochet Robots*, desenvolvido no âmbito da unidade curricular de Inteligência Artificial tem como objetivo a familiarização, estudo e implementação de algoritmos expostos nas aulas teóricas.

Sendo este projeto um jogo de tabuleiro, é pretendida a aplicação de algoritmos para Pesquisa Sistemática/Informada de Soluções à resolução do mesmo de forma autónoma por parte do computador, recorrendo a várias estratégias para esse mesmo objetivo como A*.

Descrição

- Especificação

O jogo *Ricochet Robots* conta com um tabuleiro inicial com paredes nas extremidades, células quadrangulares, algumas delas com paredes próprias em duas das suas quatro bordas, bem como 4 pontos para cada uma de 4 cores (azul, verde, vermelho e amarelo) espalhados pelo tabuleiro.



Inicialmente são colocados 4 robôs com as mesmas cores dos pontos já mencionados, em posições aleatoriamente escolhidas. O objetivo do jogo, com qualquer número de jogadores, passa por observar o estado do tabuleiro e chegar ao número mínimo de movimentos necessário para chegar com o robô de uma cor, determinada de forma aleatória, até um dos pontos dessa mesma cor presentes no tabuleiro. Um movimento consiste em mover um robot numa direção até este bater numa parede ou noutro robot.

No que toca à implementação do trabalho, a linguagem escolhida é C++, devido à familiarização do grupo com a mesma. Para o desenvolvimento do jogo são implementadas várias classes, *Board* para a representação do tabuleiro, *Robot* para referenciar os robôs a mover e *Wall* para as paredes onde os robôs fazem “ricochete”.

Na classe *Board* serão inicializados os 4 robôs e os alvos onde estes devem chegar serão representados através de pares de *ints*. A localização das células do tabuleiro é feito através de “coordenadas”.

Depois de completa a representação gráfica do tabuleiro, os “múltiplos jogadores” serão retratados por métodos de pesquisa informada como A*. Este método ao terminar o seu processamento indicará que encontrou uma solução e o número de movimentos válidos da mesma.

- **Trabalho Efetuado**

Até este momento estão implementadas, ainda que de forma parcial, as classes *Board*, *Wall* e *Robot*. Estas 3 classes já têm implementados os seus construtores e métodos de acesso (e modificação, caso seja pertinente) aos respetivos membros.

A classe *Board* já dispõe de um método para mover um robot numa direção, assim como um método para obter a distância entre um robot e o alvo de que se encontra mais perto e outro para somar essas distâncias de todos os robots de modo a obter uma espécie de heurística.

Para além disso, está construído um tabuleiro inicial ao qual pretendemos adicionar, pelo menos, mais uma alternativa.

- **Resultados esperados e forma de avaliação**

Uma das consequências diretas da implementação do algoritmo A* é a redução de custos tanto em espaço como em tempo.

Para averiguar se o algoritmo A* cumpre o esperado a diversos níveis, para efeitos de teste e forma de avaliação iremos comparar a sua execução com outros algoritmos, para isso mediremos o tempo e o espaço que o mesmo necessita na sua execução.

Conclusões

Após uma primeira abordagem ao trabalho pedido, foram tiradas elações relacionadas com o funcionamento e implementação do jogo pedido, bem como sobre os métodos a utilizar para encontrar a solução, nomeadamente A^* .

Estas primeiras conclusões terão um papel preponderante na implementação integral da solução pretendida.

Recursos

- Bibliografia
 - "Artificial Intelligence: A Modern Approach" - Stuart Russell and Peter Norvig
- Software
 - Visual Studio 2015
 - GitHub