

Manual Técnico - 2ª Fase do Projeto



**POLITECNICO
SETUBAL**

Unidade Curricular de Inteligência Artificial

Docentes:

- Prof. Joaquim Filipe
- Eng. Filipe Mariano

Jogo do Cavalo

Manual Técnico

Projeto realizado por Grupo 51:

- Miguel Neves - 201900377
- Guilherme Ravasco - 201900646

Índice

1. Introdução
2. Arquitetura do Sistema
3. Entidades
4. Algoritmo
5. Limitações técnicas

1. Introdução

Este projeto tem como objetivo realizar um programa para resolver o **Jogo do Cavalo**. Nesta segunda fase, pegamos na base do primeiro projeto e alterou-se o programa de forma a ser possível um jogador jogar contra outro jogador, jogador jogar contra o computador e o computador a jogar contra computador. O **Jogo do Cavalo** é uma variante do problema matemático conhecido como o **Passeio do Cavalo**, onde o objetivo é, através dos movimentos do cavalo, conseguir visitar todas as casas de um tabuleiro parecido ao de xadrez. Esta versão decorrerá num tabuleiro de 10 linhas e 10 colunas (10x10), onde cada casa possui uma pontuação. Estes movimentos são implementados com operadores em **LISP**.

2. Arquitetura do Sistema

O projeto desenvolvido para a resolução do **Jogo do Cavalo** foi implementado em **Common Lisp** que foi a linguagem lecionada ao longo da Unidade Curricular de Inteligência Artificial, utilizando o **Lisp Works** como ferramenta de desenvolvimento de código

A resolução do **Jogo do Cavalo** está subdividido em 3 ficheiros:

- algoritmo.lisp - Algoritmo Alfabeta, seletores do nós, geração dos sucessores e heurística
- interact.lisp - Interação com o utilizador.
- jogo.lisp - Jogo do Cavalo, onde consta os 3 modos de jogo, Player vs Player, Player vs AI e AI vs AI.

3. Entidades

Tabuleiro

O tabuleiro é representado em forma de lista de listas em **Common LISP**, o tabuleiro é composto por 100 átomos, pois o tabuleiro é composto por 10 linhas e 10 colunas, cada átomo representa uma casa, e estas têm um valor numérico de 0 a 99.

Funções e Regras do Jogo

Nesta parte do manual técnico estão descritas as funções e regras que foram implementadas no projeto, sabendo que as movimentações do cavalo apenas podem ser realizadas para casas disponíveis no tabuleiro.

Regra do Cavalo

Esta regra tem como objetivo principal colocar o cavalo no jogo, a regra é aplicada sempre que se começa a procura pelo tabuleiro. Esta regra começa por validar se o cavalo está posicionado numa casa e verifica se este está na primeira linha do tabuleiro.

O resultado desta regra é o conjunto de casas que estão disponíveis para realizar a jogada, aplicando também as regras do Simétrico e do Duplo que estão descritas abaixo.

Regra do Simétrico

Esta regra coloca o valor simétrico da casa visitada no movimento atual como **NIL**, após isto, essa mesma casa já não poderá ser mais visitada em jogadas futuras. Um número simétrico é por exemplo temos o valor **"34"** o seu simétrico será **"43"**.

Regra do Duplo

Esta regra verifica se o valor da casa que está a ser visitada no movimento da jogada atual é um número duplo. Por exemplo, **"33"**, **"55"**. O que esta regra irá fazer com este número duplo é colocar na casa com o maior número duplo disponível a **NIL**, esta casa após isto não poderá ser visitada em jogadas futuras.

Regra do Jogador

A Regra do Jogador estipula que, em qualquer movimento, um jogador não pode fazer uma jogada se uma casa tiver ameaçada pelo adversário. No caso de não haver mais nenhuma jogada disponível o jogo acaba e o vencedor é quem tiver mais pontos.

Representação de Estados

O **Jogo do Cavalo** é um desafio que envolve encontrar o caminho de um cavalo num tabuleiro de xadrez, passando por todas as casas exatamente uma vez. Para resolver este problema, é fundamental abordá-lo em termos de estados, representando as diferentes configurações do tabuleiro em cada etapa do jogo. Além disso, é preciso definir operadores que permitem a transição entre estes estados, levando assim o cavalo da sua posição inicial até à posição final.

O problema pode ser abordado utilizando técnicas de busca em árvore, explorando assim as várias possibilidades de movimento do cavalo a partir de cada estado atual.

Operadores

Baseando-nos nas oito operações disponíveis em L, criámos oito operadores que executam esses mesmos movimentos. Essas operações podem ser realizadas em relação às linhas e às colunas do tabuleiro.

Na organização da função, a representação é feita por coordenadas (coluna, linha), onde a coluna corresponde aos movimentos para cima e para baixo, e a linha representa os movimentos para a direita e para a esquerda.

Os operadores representam os seguintes movimentos:

- operador-1 (2 1)
- operador-2 (1 2)
- operador-3 (-1 2)
- operador-4 (-2 1)
- operador-5 (-2 -1)
- operador-6 (-1 -2)
- operador-7 (1 -2)
- operador-8 (2 -1)

Nó

O nó é constituído por, (**Tabuleiro|Player|Score|Profundidade | Avaliação do nó | Operadores**). A avaliação do nó é (pontos player maximizante - pontos player minimizante).

Sucessões

A Sucessão é um novo nó gerado a partir do estado do nó anterior.

4. Algoritmo

O algoritmo utilizado nesta segunda fase do projeto foi o **ALFABETA**. O algoritmo **ALFABETA** é uma técnica utilizada em jogos de dois jogadores, como o **Jogo do Cavalo**, **Xadrez**, **Jogo do Galo**, entre outros, para otimizar a busca por movimentos ideais. Derivado do algoritmo **MINIMAX**, o **ALFABETA** é mais eficiente em termos computacionais, pois corta partes da árvore de busca que não influenciam a escolha final. No algoritmo, **ALFA** representa a melhor escolha atual para o maximizante e **BETA** representa a melhor escolha para o minimizante. Durante a busca, se uma escolha for pior do que a melhor opção para o minimizante ou melhor do que a melhor opção para o maximizante, a busca nesse ramo pode ser interrompida, economizando assim recursos computacionais. O **ALFABETA** requer funções recursivas eficientes, estas avaliam as possíveis jogadas, atualizando os valores de **ALFA** e **BETA** conforme necessário.

5. Limitações técnicas

Esta segunda fase do projeto gerou várias dificuldades na realização do programa para resolver o **Jogo do Cavalo**. Existiram dificuldades em implementar o algoritmo, na parte da avaliação e no cálculo do máximo e mínimo de um nó. Não foi possível gerar os resultados para mostrar.