

Tetris

Universidade de Aveiro

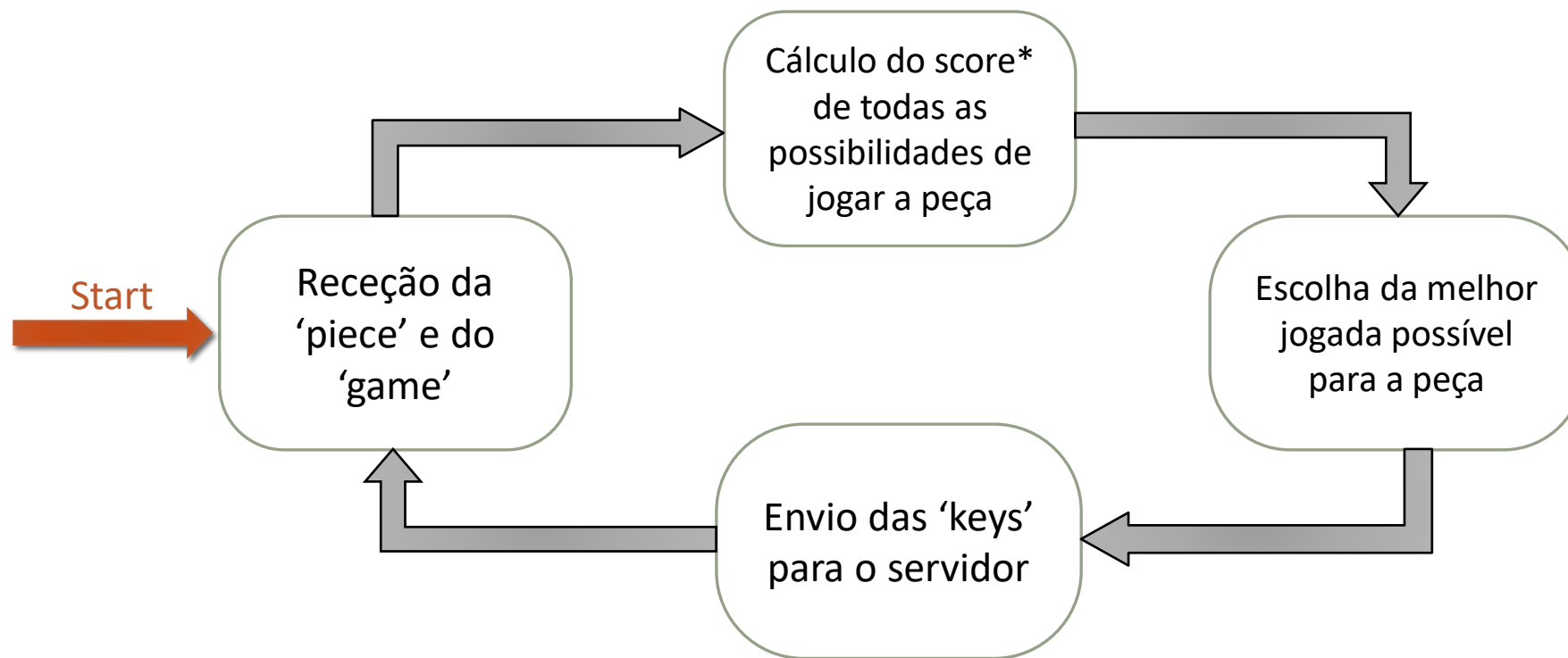
Departamento de Eletrónica, Telecomunicações e Informática
Inteligência Artificial

BRUNO SILVA - 97931

FRANCISCO CARDITA - 97640

PEDRO GUERRA - 98610

Descrição do funcionamento do agente



* O score é calculado através de heurísticas que permitem atribuir uma pontuação a cada jogada possível de uma peça (contando todas as posições e rotações).

Interpretação da 'piece' e do 'game'

- Após a receção da peça a jogar é necessário identifica-la, para isto simulámos todas as posições possíveis para o aparecimento de cada peça (partindo de um exemplo de cada tipo de peça e aplicando vetores de translação a cada uma delas). Tendo uma lista com todas as possibilidades associadas a cada tipo de peça, temos apenas que verificar em qual destas é que se insere a peça recebida e devolver o tipo correspondente.
- O agente deve também ter conhecimento do estado atual do 'game' aquando da receção de uma nova peça. Para tornar a interpretação desta estrutura mais simples foi criada uma função que a transforma numa matriz com as dimensões do jogo e em que cada posição a 1 representa um bloco ocupado e uma posição a 0 o contrário. Este mecanismo torna o cálculo das heurísticas mais simples e eficiente.

Heurísticas

- Após a receção e interpretação da peça a jogar e do estado atual do jogo o agente decide qual a melhor jogada calculando scores para todas as possibilidades, escolhendo no fim a possibilidade com o melhor score.
- Os scores são calculados através de heurísticas, sendo estas:
 - **Aggregate height** -> soma da altura de cada uma das colunas
 - **Complete lines** -> o número de linhas completas
 - **Number of holes** -> o número de buracos, que é um espaço vazio com pelo menos uma posição ocupada acima deste.
 - **Bumpiness** -> soma da variação de alturas entre as várias colunas
- Através da fórmula* apresentada a baixo é calculado o score:
$$-0,510066 * \text{AggregateHeight} + 0,760666 * \text{CompleteLines} - 0,35663 * \text{NumberOfHoles} - 0.184483 * \text{Bumpiness}$$

* Consultada em https://codemyroad.wordpress.com/2013/04/14/tetris-ai-the-near-perfect-player/?fbclid=IwAR0ysYYxA2_IOfirvRI5etTZ6UsEEGKM_c9XfKmimWM9h3hd-NvICGDkTts

Rotação das peças

- Para que seja possível estudar todas as possibilidades de jogar uma peça é necessário que o agente saiba simular a rotação desta.
- Estudámos o centro de rotação das peças e as coordenadas dos blocos de cada peça para cada rotação possível. Chegámos à conclusão que algumas das peças apenas têm dois estados possíveis (contando com a posição original), ou seja, após a segunda rotação voltam ao seu estado inicial (peças I, S, Z) sendo que as peças restantes têm quatro estados possíveis, à exceção da peça O que tem apenas um.

Simulação de todas as possibilidades

- O agente necessita de simular todas as possibilidades para que no fim consiga escolher a melhor. Isto é conseguido através da simulação da queda da peça em todas as posições e rotações possíveis, calculando o score para cada uma destas possibilidades.
- Existe também um mecanismo de “segurança” que evita que sejam formadas “torres”. Quando o agente deteta que existe uma altura e bumpiness elevados procede de forma a obrigar que as peças sejam jogadas nas laterais (apenas calculando os scores destas mesmas posições) de maneira a prevenir que o jogo acabe precocemente.
- De seguida escolhe-se o score mais alto que corresponde à melhor jogada possível, isto é a peça na sua melhor posição e rotação possível.
- Por fim, compara-se a peça original com a melhor solução encontrada pelo agente para que se encontre as **keys** necessárias para enviar ao server.