

Agents that eat, cooperate and live a long life

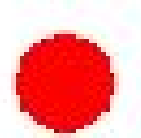
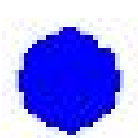
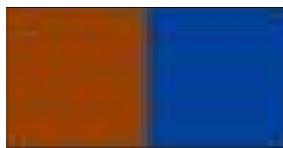
Introdução a Inteligência Artificial

Dezembro 2017





O agente



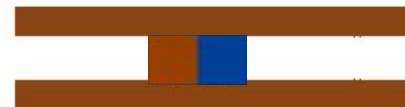
- Agente inteligente
- Com memória do percurso entre ações
- Escolhe sempre a comida mais direta
- Comunicação entre agentes para não ficarem presos
- Consciente da morte do colega
- Pesquisa de percurso rápida e eficaz
- Prioridade da comida que precisa

Pré Processamento:

- Dead Ends



- Dead Locks



- Waypoints para ajudar no pathfinding (desativado por não uso)
- Grafo para representar as ligações entre waypoints (desativado por não uso)

chooseAction()



Lidar com possível mensagem recebida
Cálculo das ações válidas possíveis
Sort da comida em relação à distância e tipo



Vê food



Comida direta?

else



Path guardado?

else

Calcular path para comida mais próxima/Seguir path



Ir para a posição mais próxima/Comer



else

Seguir path atual



Path guardado melhor que path para comida mais próxima?



Actualizar path/Seguir path

else



Path guardado?

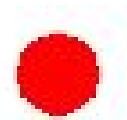
else

Recalcular o path

Seguir path atual

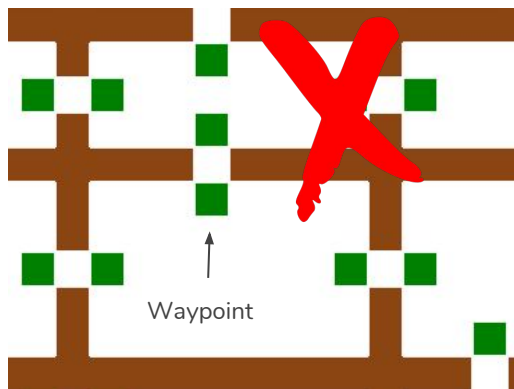


Pathfinding



Algoritmo usado:

- A* usado no início
- Pesquisa em largura com heurística em prioridade (sort de neighbors) foi a mais eficiente
- Tentativa falhada de fazer um search com waypoints e um grafo para representar as ligações



- Uso de dicionários para obter complexidade de acesso $O(1)$
- Verificação de nós já passados através de set





Dead Ends



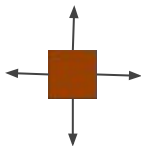
and

Dead Locks



Algoritmo usado:

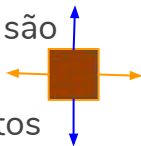
- Percorrer todos os pontos que não são walls
- Se existem 3 walls/dead-ends e 1 bloco livre nestas direções:
 - Colocar nos dead-ends
 - E continuar na direção livre até não existirem 3 walls/dead-ends e um bloco livre



Variáveis - 2 Pontos (borders), Ocupado(bool)

Algoritmo usado:

- Percorrer todos os pontos que não são walls
- Se existe duas paredes nos conjuntos destas direções:
 - Explorar para cada direção enquanto só existir possibilidade de movimento (No backtracking), actualizando as borders
 - Retirar cada ponto percorrido à lista inicial de pontos a percorrer



Comunicação entre agentes

- Usada para os *dead locks*
- Todos os deadlocks têm uma variável *taken* booleana que indica se alguém se encontra no *dead lock*
- Caso receba uma mensagem faz set do valor *taken* a *True* do *dead lock* enviado
- Envia uma mensagem quando entra num *dead lock* e quando sai



Leonardo Costa - 80162

Davide Cruz - 71776

Rodrigo Rocha - 71731

