

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
CAMPUS CAMPINA GRANDE
CURSO BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

DHIEGO VINICIUS DA COSTA
EMILIENY DE SOUZA SILVA
MARINO PAULINO MOUZINHO DA SILVA

RELATÓRIO: Agente Caçador de Tesouros

Campina Grande - PB
2025

DHIEGO VINICIUS DA COSTA
EMILIENY DE SOUZA SILVA
MARINO PAULINO MOUZINHO DA SILVA

RELATÓRIO: Agente Caçador de Tesouros

Relatório de Análise de Agente Inteligente
para componente curricular obrigatório da
disciplina de Inteligência Artificial do
Curso Bacharelado em Engenharia de
Computação do IFPB - Campus Campina
Grande.

Professor Orientador: Marcelo José Siqueira Coutinho de Almeida

Campina Grande - PB
2025

SUMÁRIO

| | |
|---|----------|
| 1. INTRODUÇÃO..... | 4 |
| 2. DESCRIÇÃO DO AMBIENTE..... | 5 |
| 3. LÓGICA E FUNCIONAMENTO DO AGENTE..... | 6 |
| 4. TÉCNICAS DE IA UTILIZADAS..... | 7 |
| 4.1. Heurística..... | 7 |
| 5. IMPLEMENTAÇÃO..... | 8 |
| 6. CONCLUSÃO..... | 9 |

1. INTRODUÇÃO

A Inteligência Artificial é um ramo da Ciência Computacional que busca desenvolver sistemas capazes de realizar tarefas que demandam inteligência humana para realizar certas tarefas. Ela tem se permeado multidisciplinarmente com o passar dos anos e nos últimos anos houve um grande holofote sobre a área em decorrência das IAs generativas. Dentre seus pilares fundamentais, há o Agente Inteligente, que nada mais é do que entidades autônomas que percebem seus ambientes através de sensores e agem sobre ele por meio de atuadores, utilizando processos de tomada de decisão e inferências, podendo ser baseados em conhecimento, sem conhecimento e/ou aprendizagem.

Diante disso, há cenários em que os agentes são aplicados constantemente, como no caso dos ambientes desconhecidos, onde o agente deve explorar eficientemente um espaço de estados a fim de cumprir com o seu objetivo. Para auxiliar o agente na tomada de decisões e ações são aplicadas diversas técnicas - isoladas ou conjuntamente - como por exemplo, DFS (), BFS (), Busca Gulosa, entre outros.

O presente relatório busca detalhar o processo de desenvolvimento de um agente inteligente caçador de tesouros, capaz de explorar um ambiente parcialmente desconhecido - representado por uma matriz bidimensional NxM - com o objetivo de localizar um tesouro escondido (T) em uma de suas posições. Podendo se encontrar em diversas posições desta matriz obstáculos (X), Poços (P), indícios de ouro (+) ou de poço (-). Para este desenvolvimento serão aplicadas diversas técnicas de inteligência artificial de modo que se espera obter um resultado satisfatório do agente em seu ambiente.

Figura 1 - Representação gráfica da matriz ambiente NxM

| | | | | | | | | | | |
|---|---|---|--|---|---|---|---|---|---|---|
| | | | | | | | | - | P | - |
| | | | | | | | | - | - | - |
| - | - | | | | | | | | | X |
| P | - | | | + | + | + | | | | |
| - | - | | | + | T | + | | | | X |
| | | | | + | + | + | | | | |
| | | | | | | | | | | |
| - | - | X | | | | | | | X | X |
| P | - | | | | | | | | | |
| - | - | | | | X | X | X | | | |

Fonte: Almeida; Marcelo J. S. C. (2025)

2. DESCRIÇÃO DO AMBIENTE

O ambiente é representado por uma matriz $N \times M$, onde cada célula pode conter:

- E ou vazio: Espaço livre
- X: Obstáculo (não atravessável)
- P: Poço (leva à morte do agente)
- + : Indício de tesouro (em célula adjacente)
- - : Indício de poço (em célula adjacente)
- T: Tesouro

O agente começa na posição fixa (0,0) (canto inferior esquerdo) e possui percepção limitada às células adjacentes (norte, sul, leste, oeste).

3. LÓGICA E FUNCIONAMENTO DO AGENTE

O agente segue o ciclo PDA (Percepção → Decisão → Ação), sendo guiado por uma estratégia baseada em busca heurística.

A cada ciclo, ele:

1. Percebe o conteúdo das células adjacentes.
2. Decide o próximo movimento com base nas percepções e no histórico.
3. Executa a ação e atualiza seu estado interno.

O agente armazena um mapa parcial do ambiente, evita revisitar posições já exploradas e evita perigos com base em percepções locais.

4. TÉCNICAS DE IA UTILIZADAS

A estratégia principal adotada foi uma implementação robusta do algoritmo A* (A-estrela), complementada por mecanismos de prevenção de loops e exploração adaptativa. Esta abordagem demonstrou-se eficaz tanto para navegação em ambientes desconhecidos quanto para tratamento de incertezas, atendendo aos requisitos de otimização de movimentos mesmo com percepção limitada.

4.1. Heurística

A heurística adotada foi a distância de Manhattan até as posições com indícios de tesouro (representados por "+"):

$$h(n) = |x_1 - x_2| + |y_1 - y_2|$$

A função heurística utiliza a distância de Manhattan até o tesouro ou, quando este não é visível, até os indícios mais próximos ('+'), calculada como a soma das diferenças absolutas das coordenadas. Para evitar comportamentos cíclicos, o sistema incorpora um mecanismo de penalização que aumenta o custo de células revisitadas, garantindo a progressão da exploração.

A escolha da busca heurística com A* foi motivada por sua capacidade de guiar o agente de forma eficiente por caminhos promissores, reduzindo o número de movimentos necessários para atingir o objetivo. A percepção limitada foi compensada pela manutenção de um mapa interno e pela aplicação da heurística, que ajudou a evitar zonas perigosas e a concentrar a busca em regiões potencialmente relevantes.

5. IMPLEMENTAÇÃO

O código-fonte pode ser encontrado na plataforma GitHub¹. Utilizamos a linguagem Python e o paradigma de orientação à objetos para implementar o cenário. Temos uma classe Ambiente que guarda as informações do ambiente como localização do agente, dos poços, obstáculos e do tesouro, além das consequentes dicas (“+” e “-”) nas células adjacentes ao tesouro e poços. Ela é responsável pela geração procedural do grid 10x10, garantindo que o tesouro seja sempre acessível e que a posição inicial (0,0) esteja livre de perigos. Ela distribui aleatoriamente os elementos: 1 a 3 poços ('P'), 4 a 6 obstáculos ('X'), e os respectivos indícios ('+' para tesouro e '-' para poços).

A classe Agente implementa a lógica de decisão, combinando busca planejada com comportamentos reativos para lidar com situações imprevistas.

Durante a execução, o sistema segue um ciclo Percepção-Decisão-Ação (PDA) discreto. Em cada iteração, os agentes coletam informações das células adjacentes, atualizam seu mapa interno, decidem o próximo movimento utilizando o algoritmo A* modificado e executam a ação. O processo termina quando algum dos agentes encontra o tesouro, cai em um poço ou atinge o limite de 100 iterações.

¹ <https://github.com/dhiegov/wumpus>

6. CONCLUSÃO

A implementação desenvolvida demonstrou a eficácia da combinação entre algoritmos de busca heurística e técnicas reativas para navegação autônoma em ambientes parcialmente observáveis. O sistema atende plenamente aos requisitos do projeto.

Como aprimoramentos futuros, sugerimos a implementação de aprendizagem por reforço para refinamento da heurística e extensão para ambientes dinâmicos com obstáculos móveis.