



UNIVERSIDADE  
ESTADUAL DO CEARÁ

## SEGUNDO TRABALHO DA DISCIPLINA DE INTELIGÊNCIA COMPUTACIONAL

### RELATÓRIO TÉCNICO

**Equipe:**

Alicia Paiva	- 1612479
Rhavy Frota Sá Nogueira Neves Souza	- 1612486

**Professor:**

Gustavo Campos

Fortaleza, Ceará  
16 de janeiro de 2025

## 1. Estudo Dirigido Parte I

01. Defina com suas próprias palavras os seguintes termos: estado, espaço de estados, árvore de busca, nó, objetivo, função sucessora e fator de ramificação.

**Resposta:**

Estado é a situação atual do problema. Em um jogo de xadrez, um estado pode ser a disposição atual das peças no tabuleiro.

Espaço de estados é o conjunto de todos os estados possíveis que podem ser alcançados a partir de um estado inicial, seguindo as regras do problema. Ele é geralmente representado como um grafo, onde os estados são nós, e as transições (ações) são as arestas.

Árvore de busca é uma representação hierárquica dos estados explorados durante a resolução do problema. Começa com o estado inicial como a raiz e expande para outros estados (nós) conforme as ações possíveis são aplicadas. A árvore mostra as escolhas feitas ao longo do caminho.

Um nó na árvore de busca representa um estado específico do problema. Além disso, ele pode conter informações adicionais, como:

- O estado correspondente.
- A ação que levou a esse estado.
- O custo acumulado do caminho até esse estado.
- O nó pai (estado anterior).

O objetivo é o estado ou conjunto de estados que representam a solução do problema. O agente tenta encontrar um caminho para alcançar esse estado. No xadrez, o objetivo pode ser dar xeque-mate no rei adversário.

Função sucessora é a função que define como o agente pode ir de um estado a outro. Dado um estado atual, ela retorna todas as ações possíveis e os estados resultantes dessas ações.

Fator de ramificação é o número médio de sucessores (ações possíveis) gerados a partir de cada nó na árvore de busca. Este valor afeta diretamente a complexidade do problema.

02. Defina para um agente de resolução os seguintes problemas, ou seja, represente o estado inicial do problema, as ações possíveis para o agente, um teste meta/objetivo e uma função custo de caminho,  $g(n)$ .

- a. Usando apenas quatro cores, colorir um mapa plano de tal forma que duas regiões adjacentes não tenham a mesma cor.

**Resposta:**

Estado Inicial: Um mapa onde nenhuma região está colorida.

Ações Possíveis: Escolher uma região não colorida e aplicar uma cor (vermelho, azul, verde ou amarelo) que não conflite com as regiões adjacentes.

Teste de Objetivo: Todas as regiões estão coloridas e nenhuma região adjacente tem a mesma cor.

Função Custo: O número de regiões coloridas até o estado atual.

Função Heurística: O número de regiões restantes que ainda precisam ser coloridas.

Função Avaliação:  $f(n)=g(n)+h(n)$ , onde  $g(n)$  é o número de regiões coloridas e  $h(n)$  é o número de regiões não coloridas.

- b. Um macaco com 30 cm está em uma sala onde tem algumas bananas suspensas em um teto de 80 cm. Ele gostaria de pegar as bananas. A sala contém dois engradados móveis e escaláveis com 30 cm de altura que podem ser empilhados.

**Resposta:**

Estado Inicial:

- Macaco está no chão.
- Bananas estão suspensas a 80 cm.
- Dois engradados de 30 cm estão disponíveis no chão.

Ações Possíveis:

1. Empurrar um engradado.
2. Empilhar os engradados.
3. Subir em um engradado.
4. Pegar as bananas (se a altura for suficiente).

Teste de Objetivo: O macaco alcança as bananas (altura total  $\geq 80$  cm)

Função Custo: O número de ações realizadas até o estado atual.

Função Heurística: A diferença entre a altura atual do macaco e a altura necessária para alcançar as bananas ( $80 - \text{altura atual}$ ).

Função Avaliação:  $f(n)=g(n)+h(n)$ , onde  $g(n)$  é o número de ações realizadas e  $h(n)$  é a altura que ainda falta para alcançar as bananas.

- c. Existe um programa que exibe a mensagem “registro de entrada inválido” ao alimentar determinado arquivo com registros de entrada. Você sabe que o

processamento de cada registro é independente de outros registros e deseja descobrir qual registro é inválido.

**Resposta:**

Estado Inicial: Todos os registros do arquivo estão classificados como "não verificados".

Ações Possíveis: Selecionar um registro e processá-lo no programa.

Teste de Objetivo: Um registro que, quando processado, gera a mensagem "registro de entrada inválido".

Função Custo: O número de registros processados até o estado atual.

Função Heurística: O número de registros restantes que ainda precisam ser processados.

Função Avaliação:  $f(n)=g(n)+h(n)$ , onde  $g(n)$  é o número de registros processados e  $h(n)$  é o número de registros ainda não verificados.

- d. Três missionários e três canibais estão em um lado de um rio, juntamente com um barco que pode levar uma ou duas pessoas. Descubra um meio de fazer todos atravessarem o rio sem deixar que um grupo de missionários de um lado fique em número menor que o número de canibais nesse mesmo lado do rio.

**Resposta:**

Estado Inicial: Todos os três missionários, os três canibais e o barco estão na margem inicial do rio.

Ações Possíveis: Mover o barco com uma ou duas pessoas (missionários e/ou canibais) para a outra margem, respeitando a restrição de não poder existir mais canibais do que missionários em nenhuma margem.

Teste de Objetivo: Todos os missionários e canibais estão na margem oposta do rio.

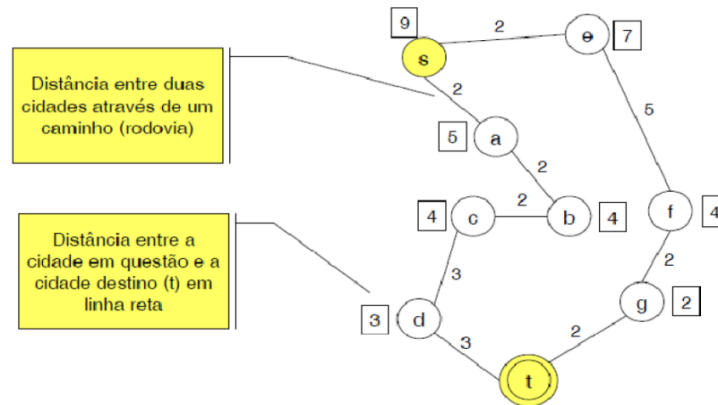
Função Custo: O número de viagens realizadas pelo barco até o estado atual.

Função Heurística: O número de pessoas restantes na margem inicial.

Função Avaliação:  $f(n)=g(n)+h(n)$ , onde  $g(n)$  é o número de viagens realizadas e  $h(n)$  é o número de pessoas ainda na margem inicial.

Em seguida, para cada um dos problemas, especifique uma função avaliação de nó  $n$ ,  $f(n) = g(n) + h(n)$ , no grafo representativo do espaço de estados do problema.

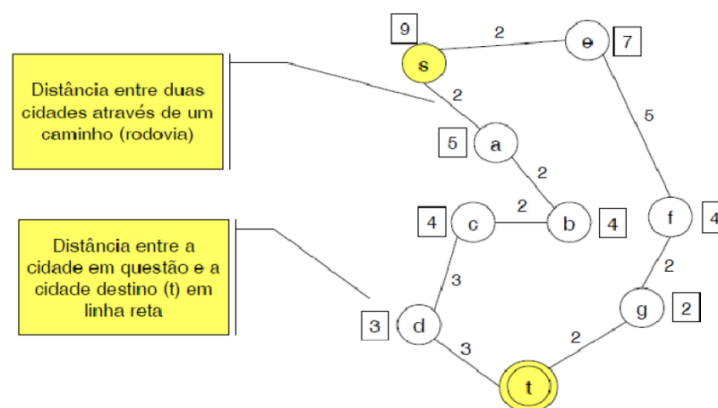
03. Encontre o caminho de s até t empregando as seguintes estratégias de busca: largura, custo uniforme e profundidade. Apresente a evolução da árvore de busca que vai sendo gerada por cada uma das estratégias. Apresente a evolução do conteúdo na borda de nós que ainda não foram explorados (expandidos) e do conteúdo da lista de estados que já foram explorados (lista fechada), quando empregando busca em grafo.



Verifique nos seus resultados: a busca em largura é um caso especial de busca de custo uniforme.

## 2. Estudo Dirigido Parte II

04. Encontre o caminho de s até t empregando as seguintes estratégias de busca: gulosa pela melhor escolha e busca A\*. Apresente a evolução da árvore de busca que vai sendo gerada por cada uma das estratégias. Apresente a evolução do conteúdo na borda de nós que ainda não foram explorados (expandidos) e do conteúdo da lista de estados que já foram explorados (lista fechada), quando empregando busca em grafo.



Verifique nos seus resultados:

- a) busca em largura, busca em profundidade e busca de custo uniforme são casos especiais de busca pela melhor escolha;

**Resposta:**

-----

current node: s of dist: 0

Vertices de borda:

Distancias:

dist: 0, node: s,

Visited nodes:

s,

-----

current node: a of dist: 2

Vertices de borda:

real dist: 2, pred dist: 7, node: e,

Distancias:

dist: 2, node: a,

dist: 2, node: e,

dist: 0, node: s,

Visited nodes:

a, s,

-----

current node: b of dist: 4

Vertices de borda:

real dist: 2, pred dist: 7, node: e,

Distancias:

dist: 2, node: a,

dist: 4, node: b,

dist: 2, node: e,

dist: 0, node: s,

Visited nodes:

a, b, s,

-----

current node: c of dist: 6

Vertices de borda:

real dist: 2, pred dist: 7, node: e,

Distancias:

dist: 2, node: a,

dist: 4, node: b,

dist: 6, node: c,

dist: 2, node: e,

dist: 0, node: s,

Visited nodes:

a, b, c, s,

-----

current node: d of dist: 9

Vertices de borda:

real dist: 2, pred dist: 7, node: e,

Distancias:

dist: 2, node: a,

dist: 4, node: b,

dist: 6, node: c,

dist: 9, node: d,

dist: 2, node: e,

dist: 0, node: s,

Visited nodes:

a, b, c, d, s,

-----

current node: t of dist: 12

Vertices de borda:

real dist: 2, pred dist: 7, node: e,

Distancias:

dist: 2, node: a,  
dist: 4, node: b,  
dist: 6, node: c,  
dist: 9, node: d,  
dist: 2, node: e,  
dist: 0, node: s,  
dist: 12, node: t,

Visited nodes:

a, b, c, d, s, t,

b) busca de custo uniforme é um caso especial de busca A\*.

**Resposta:**

-----

current node: s of dist: 0

Vertices de borda:

Distancias:

dist: 0, node: s,

Visited nodes:

s,

-----

current node: a of dist: 2

Vertices de borda:

real dist: 2, pred dist: 9, node: e,



Distancias:

dist: 2, node: a,

dist: 2, node: e,

dist: 0, node: s,

Visited nodes:

a, s,

-----

current node: b of dist: 4

Vertices de borda:

real dist: 2, pred dist: 9, node: e,

Distancias:

dist: 2, node: a,

dist: 4, node: b,

dist: 2, node: e,

dist: 0, node: s,

Visited nodes:

a, b, s,

-----

current node: e of dist: 2

Vertices de borda:

real dist: 6, pred dist: 10, node: c,

Distancias:

dist: 2, node: a,

dist: 4, node: b,

dist: 6, node: c,

dist: 2, node: e,

dist: 0, node: s,

Visited nodes:

a, b, e, s,

-----

current node: c of dist: 6

Vertices de borda:

real dist: 7, pred dist: 11, node: f,

Distancias:

dist: 2, node: a,

dist: 4, node: b,

dist: 6, node: c,

dist: 2, node: e,

dist: 7, node: f,

dist: 0, node: s,

Visited nodes:

a, b, c, e, s,

-----

current node: f of dist: 7

Vertices de borda:

real dist: 9, pred dist: 12, node: d,

Distancias:

dist: 2, node: a,

dist: 4, node: b,

dist: 6, node: c,

dist: 9, node: d,

dist: 2, node: e,

dist: 7, node: f,

dist: 0, node: s,

Visited nodes:

a, b, c, e, f, s,

-----

current node: g of dist: 9

Vertices de borda:

real dist: 9, pred dist: 12, node: d,

Distancias:

dist: 2, node: a,  
dist: 4, node: b,  
dist: 6, node: c,  
dist: 9, node: d,  
dist: 2, node: e,  
dist: 7, node: f,  
dist: 9, node: g,  
dist: 0, node: s,

Visited nodes:

a, b, c, e, f, g, s,

-----

current node: t of dist: 11

Vertices de borda:

real dist: 9, pred dist: 12, node: d,

Distancias:

dist: 2, node: a,  
dist: 4, node: b,  
dist: 6, node: c,  
dist: 9, node: d,  
dist: 2, node: e,  
dist: 7, node: f,  
dist: 9, node: g,  
dist: 0, node: s,  
dist: 11, node: t,

Visited nodes:

a, b, c, e, f, g, s, t,

05. O algoritmo de busca A\* avalia nós combinando o custo para alcançar cada nó  $n$  ( $g(n)$ ) e o custo para ir do nó  $n$  até o objetivo ( $h(n)$ ):

$$f(n) = g(n) + h(n).$$

A ideia deste algoritmo foi combinar as estratégias de busca de custo uniforme, que considera  $g(n)$ , e a de busca gulosa pela melhor escolha, que considera  $h(n)$ .

a) Diga se a busca A\* é um algoritmo que satisfaz completeza e se encontra a solução ótima. Discuta as afirmações com base em análises feitas sobre  $g(n)$  e  $h(n)$ .

**Resposta:**

Somente em alguns contextos. O algoritmo A\* necessita que certas propriedades de admissibilidade e consistência sejam garantidas para que o resultado encontrado por A\* Seja ótimo.

b) Discuta sobre a complexidade de tempo e de espaço da busca A\*.

**Resposta:**

As complexidades de tempo de A\* dependem da “qualidade” da função heurística, no melhor caso (função heurística representando perfeitamente a realidade) pode ter complexidade de tempo  $O(\text{path\_size})$ , enquanto no pior caso pode ser acabar sendo proporcional ao número total de vértices ou seja  $O(N)$ . Já para complexidade de espaço temos  $O(b^d)$  sendo  $d$  a profundidade da execução e  $b$  a taxa média de ramificação dos estados.

06. Considere o problema do mundo do aspirador simplificado com duas salas. Qual dos algoritmos de busca sistemática seria apropriado para este problema? O algoritmo deveria utilizar busca em grafo ou em árvore? Aplique o algoritmo escolhido para um mundo com três salas. Compare a solução obtida com a solução obtida por um agente reativo aleatório, ou seja, que aspira se houver sujeira e move-se aleatoriamente em caso contrário.

**Resposta:**

É apropriada uma busca em grafo, podendo se modelar o problema de modo que cada posição no “grid” representasse um nó no grafo, cada nó tendo arestas para os nós das posições adjacentes. Simulação e relatório parecido foi realizado no trabalho anterior.

07. Forneça o nome do algoritmo que resulta de cada um dos seguintes casos especiais:

a. Busca em feixe local com  $k=1$ .

**Resposta:**

Seria análogo a uma subida de encosta

b. Busca em feixe local com um estado inicial e nenhum limite sobre o número de estados mantidos.

**Resposta:**

Por simplesmente manter todos os sucessos de todos os elementos gerados, seria análogo a um bfs que, uniformemente, se espalhando pelos filhos dos nós atualmente “selecionados”

c. Têmpera simulada com  $T = 0$  em todos os passos (com omissão de teste de término).

**Resposta:**

Quando  $T = 0$  seria simplesmente selecionado o primeiro nó aleatoriamente gerado.

d. Têmpera simulada com  $T = \infty$  em todos os passos.

**Resposta:**

Com  $T = \infty$ , teremos que NUNCA será tolerado um sucessor com  $\Delta E < 0$ , assim, sendo somente escolhidos sucessores, melhores. Assim, o comportamento será também análogo à subida de encosta.

e. Algoritmo genético com tamanho de população  $N = 1$ .

**Resposta:**

Tendo a população constituída somente de um elemento, um indivíduo reproduz consigo mesmo gerando uma cópia de si e dependendo apenas de mutação para a variabilidade, transitando para estados sucessores de forma completamente aleatória e podendo até retroceder para estados anteriores.