Análise Comparativa de Algoritmos de Busca na Solução de um Problema de Caminho

Lucas Ferreira[1]

[1]Mestrando em Engenharia Eletrônica e Computação
Universidade Católica de Pelotas (UCPel)

Trabalho de Introdução à Inteligência Computacional

Sumário

Sumário

Introdução

- Um labirinto é uma combinação intrincada de passagens ou corredores, da qual é difícil encontrar um meio ou caminho de saída;

Introdução

- Um labirinto é uma combinação intrincada de passagens ou corredores, da qual é difícil encontrar um meio ou caminho de saída;
- Labirintos podem:
  - Ter tamanhos diferentes; e

Introdução

- Um labirinto é uma combinação intrincada de passagens ou corredores, da qual é difícil encontrar um meio ou caminho de saída;
- Labirintos podem:
  - Ter tamanhos diferentes; e
  - Complexidades diferentes.

Introdução

- Um labirinto é uma combinação intrincada de passagens ou corredores, da qual é difícil encontrar um meio ou caminho de saída;
- Labirintos podem:
  - Ter tamanhos diferentes; e
  - Complexidades diferentes.
- Este estudo, na computação, é chamado de Busca de Caminho;

Introdução

- Um labirinto é uma combinação intrincada de passagens ou corredores, da qual é difícil encontrar um meio ou caminho de saída;
- Labirintos podem:
  - Ter tamanhos diferentes; e
  - Complexidades diferentes.
- Este estudo, na computação, é chamado de Busca de Caminho;
- A busca é aplicada com o objetivo de achar a saída, percorrendo as células do labirinto para tal;

Introdução

- Um labirinto é uma combinação intrincada de passagens ou corredores, da qual é difícil encontrar um meio ou caminho de saída;
- Labirintos podem:
  - Ter tamanhos diferentes; e
  - Complexidades diferentes.
- Este estudo, na computação, é chamado de Busca de Caminho;
- A busca é aplicada com o objetivo de achar a saída, percorrendo as células do labirinto para tal;
- Serão aplicados os algoritmos de Busca em Largura (BFS) e A*.

Sumário

Trabalhos Relacionados

- Foram comparados 2 trabalhos relacionados implementando os algoritmos escolhidos;

Trabalhos Relacionados

- Foram comparados 2 trabalhos relacionados implementando os algoritmos escolhidos;
- O primeiro faz uso de 5 algoritmos de busca de caminho e compara apenas dois deles sem detalhes de como foram usados ou feita sua aplicação no jogo;

Trabalhos Relacionados

- Foram comparados 2 trabalhos relacionados implementando os algoritmos escolhidos;
- O primeiro faz uso de 5 algoritmos de busca de caminho e compara apenas dois deles sem detalhes de como foram usados ou feita sua aplicação no jogo;
- O segundo utiliza algoritmos NPC para competir contra um ser humano em um jogo de corrida de carros, tendo a implementação do algoritmo A* satisfatória para obstáculos estáticos e pouco satisfatória para obstáculos dinâmicos.

Sumário

Metodologia

- Testes comparativos empregando diversos tabuleiros, com as mesmas posições de obstáculos, para cada um dos algoritmos;

Metodologia

- Testes comparativos empregando diversos tabuleiros, com as mesmas posições de obstáculos, para cada um dos algoritmos;
- Foram computados tempo de execução, resultado dos nodos visitados e o número total de nodos armazenados na memória.

Sumário

Busca em Largura (BFS)

- Busca por objetos em largura;

Busca em Largura (BFS)

- Busca por objetos em largura;
- Examina todos os nós de um nível abaixo do nó raiz até encontrar o nó objetivo dentro de um grafo ou uma árvore;

Busca em Largura (BFS)

- Busca por objetos em largura;
- Examina todos os nós de um nível abaixo do nó raiz até encontrar o nó objetivo dentro de um grafo ou uma árvore;
- É bom para ser utilizado em situações em que a árvore pode ter caminhos muito profundos, principalmente se o nó estiver em uma parte mais rasa da árvore.

Busca em Largura (BFS)

- Busca por objetos em largura;
- Examina todos os nós de um nível abaixo do nó raiz até encontrar o nó objetivo dentro de um grafo ou uma árvore;
- É bom para ser utilizado em situações em que a árvore pode ter caminhos muito profundos, principalmente se o nó estiver em uma parte mais rasa da árvore.
- Não funciona muito bem quando o fator de ramificação da árvore é muito alto (Xadrez ou Go) ou quando todos os caminhos levam a um nó objetivo com caminhos de comprimentos parecidos.

A*

- Algoritmo de busca genérica;

A*

- Algoritmo de busca genérica;
- Se baseia na análise dos nós através de uma combinação de *g(nó)*, que é o custo para alcançar o nó, e *h(nó)*, que é o custo para ir do nó ao objetivo;

A*

- Algoritmo de busca genérica;
- Se baseia na análise dos nós através de uma combinação de *g(nó)*, que é o custo para alcançar o nó, e *h(nó)*, que é o custo para ir do nó ao objetivo;
- *h(nó)* deve ser uma subestimativa da distância de um nó a um nó objetivo para que ele seja ótimo;

A*

- Algoritmo de busca genérica;
- Se baseia na análise dos nós através de uma combinação de *g(nó)*, que é o custo para alcançar o nó, e *h(nó)*, que é o custo para ir do nó ao objetivo;
- *h(nó)* deve ser uma subestimativa da distância de um nó a um nó objetivo para que ele seja ótimo;
- É otimamente eficiente pois, para encontrar o nó objetivo, ele expandirá o mínimo de caminhos possível;

A*

- Algoritmo de busca genérica;
- Se baseia na análise dos nós através de uma combinação de *g(nó)*, que é o custo para alcançar o nó, e *h(nó)*, que é o custo para ir do nó ao objetivo;
- *h(nó)* deve ser uma subestimativa da distância de um nó a um nó objetivo para que ele seja ótimo;
- É otimamente eficiente pois, para encontrar o nó objetivo, ele expandirá o mínimo de caminhos possível;
- A heurística utilizada pode não ser admissível se os valores estimados *h(nó)* não forem todos subestimativas.

Sumário

Resultados

- Foram utilizados tabuleiros de tamanho 25x25, 125x125 e 525x525;

Resultados

- Foram utilizados tabuleiros de tamanho 25x25, 125x125 e 525x525;
- Três níveis de complexidade, sendo eles fácil, médio e difícil;

Resultados

- Foram utilizados tabuleiros de tamanho 25x25, 125x125 e 525x525;
- Três níveis de complexidade, sendo eles fácil, médio e difícil;
- Cada gráfico gerado do tabuleiro mostra o percurso de um algoritmo de uma cor:

Resultados

- Foram utilizados tabuleiros de tamanho 25x25, 125x125 e 525x525;
- Três níveis de complexidade, sendo eles fácil, médio e difícil;
- Cada gráfico gerado do tabuleiro mostra o percurso de um algoritmo de uma cor:
  - Vermelho para BFS; e

Resultados

- Foram utilizados tabuleiros de tamanho 25x25, 125x125 e 525x525;
- Três níveis de complexidade, sendo eles fácil, médio e difícil;
- Cada gráfico gerado do tabuleiro mostra o percurso de um algoritmo de uma cor:
  - Vermelho para BFS; e
  - Verde para A*; e

Resultados

- Foram utilizados tabuleiros de tamanho 25x25, 125x125 e 525x525;
- Três níveis de complexidade, sendo eles fácil, médio e difícil;
- Cada gráfico gerado do tabuleiro mostra o percurso de um algoritmo de uma cor:
  - Vermelho para BFS; e
  - Verde para A*; e
  - Amarelo para nodos visitados durante a busca pelos diferentes algoritmos.

Resultados

- Foram utilizados tabuleiros de tamanho 25x25, 125x125 e 525x525;
- Três níveis de complexidade, sendo eles fácil, médio e difícil;
- Cada gráfico gerado do tabuleiro mostra o percurso de um algoritmo de uma cor:
  - Vermelho para BFS; e
  - Verde para A*; e
  - Amarelo para nodos visitados durante a busca pelos diferentes algoritmos.

Resultados

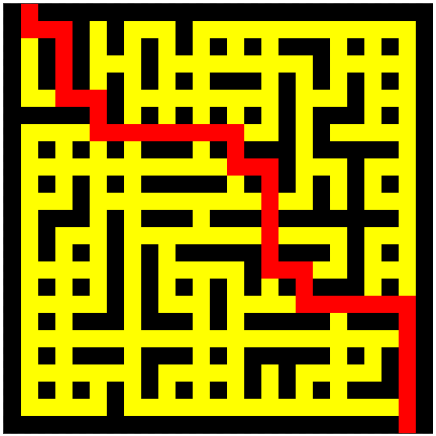

Figura 1: BFS

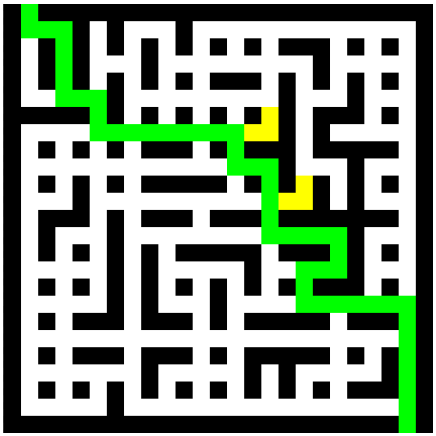

Figura 2: A*

Resultados

Componentes	BFS	A*
Nodos Visitados	346	57
Nodos na Memória	17	26
Tempo de Execução (em segundos)	0.03899	0.01100

Tabela 1: Resultados em tabuleiro de tamanho 25x25 com complexidade baixa.

Resultados

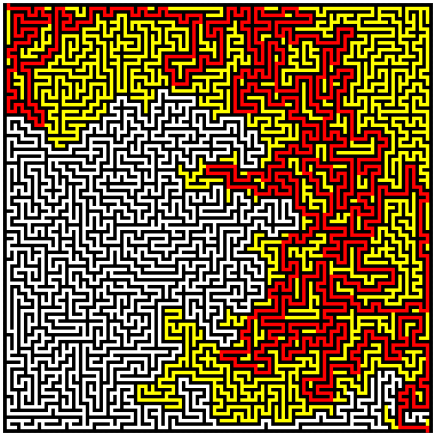

Figura 3: BFS

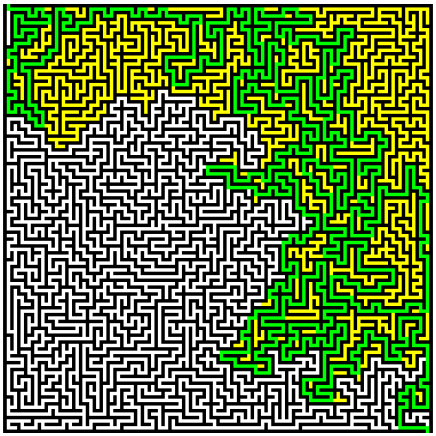

Figura 4: A*

Resultados

Componentes	BFS	A*
Nodos Visitados	4502	3897
Nodos na Memória	9	17
Tempo de Execução (em segundos)	0.36498	0.38502

Tabela 2: Resultados em tabuleiro de tamanho 125x125 com complexidade alta.

Resultados

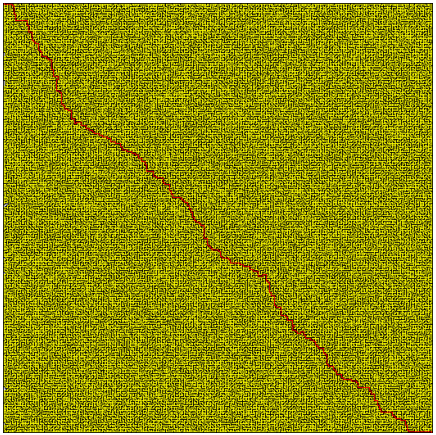

Figura 5: BFS

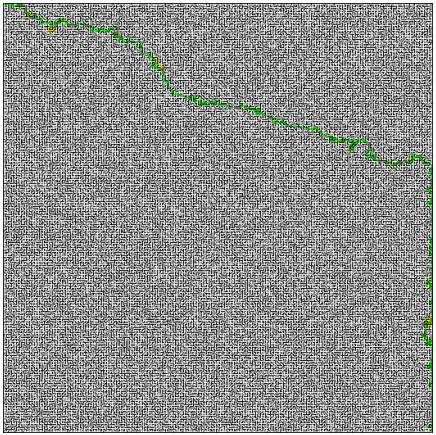

Figura 6: A*

Resultados

Componentes	BFS	A*
Nodos Visitados	154226	1604
Nodos na Memória	366	438
Tempo de Execução (em segundos)	13.5030	0.22418

Tabela 3: Resultados em tabuleiro de tamanho 525x525 com complexidade média.

Resultados

- O algoritmo A* se mostrou a melhor opção no problema discutido;

Resultados

- O algoritmo A* se mostrou a melhor opção no problema discutido;
- O BFS mostra que, independente do tamanho do tabuleiro, seus resultados são praticamente idênticos, havendo divergências grandes apenas em seus tempos de execução;

Sumário

Considerações Finais

- O grande problema é como usar os algoritmos para resolver problemas difíceis;

Considerações Finais

- O grande problema é como usar os algoritmos para resolver problemas difíceis;
- A performance geral do BFS foi satisfatória e se mostrou capaz de atingir o objetivo;

Considerações Finais

- O grande problema é como usar os algoritmos para resolver problemas difíceis;
- A performance geral do BFS foi satisfatória e se mostrou capaz de atingir o objetivo;
- O uso do algoritmo correto pode fazer com que o desempenho em termos de poder computacional, uso de memória e tempo sejam reduzidos, tornando a aplicação mais robusta e estável.

Análise Comparativa de Algoritmos de Busca na Solução de um Problema de Caminho

Lucas Ferreira[1]

[1]Mestrando em Engenharia Eletrônica e Computação
Universidade Católica de Pelotas (UCPel)

Trabalho de Introdução à Inteligência Computacional