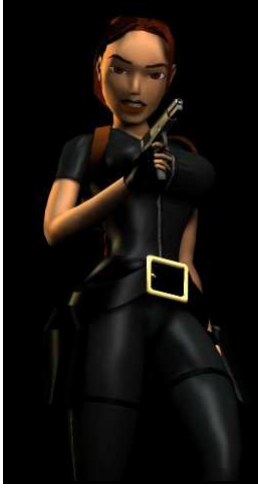
Tópicos Especiais I - Jogos
IA para Jogos



Fernando Osório

23/09/2003

IA para Jogos

Tópicos abordados...

- IA clássica: Jogos de Raciocínio [parte I]
  - Solução de problemas
  - Jogos de Tabuleiro (*Board games*)
  - Busca em Espaço de Estados (*Trees and adversarial search*)
- IA clássica: Jogos de Ação em Labirintos [parte II]
  - Busca de caminhos (Path finding)
  - Planejamento de ações e trajetórias (*Path planning*)
  - Deslocamento / Navegação (*Motion & Navigation*)
- Agentes Inteligentes: reativos, cognitivos/deliberativos e agentes autônomos (Autômatos, Rule-Based, Adaptativos)
- Aprendizado de máquinas em Jogos (ML4Games)

IA para Jogos

Tópicos abordados...

- Agentes Inteligentes
  - Comportamento: “falsa IA” e o “comportamento inteligente”
  - Agentes Reativos
  - Agentes Cognitivos / Deliberativos
  - Agentes com Arquitetura Hierárquica e Híbridos
  - Controle baseado em Autômatos (FSA, RdP)
  - Controle baseado em Regras (RBS)
  - Controle Adaptativo: agentes que aprendem
  - Agentes Autônomos Inteligentes
  - Estratégias em Jogos
- Aprendizado de máquinas em Jogos (ML4Games)

IA para Jogos

Tópicos abordados...

- Aprendizado de máquinas em Jogos
  “Machine Learning for Games”
  - Raciocínio baseado em Casos (RBC / CBR)
  - Redes Neurais Artificiais (RNA / ANN)
  - Árvores de Decisão (AD / IDT)
  - RNA-FSA
  - Aprendizado por Reforço (RL)
  - Meta-Modelos: cinemática, dinâmica
  - Perfil de Usuários
- AI SDKs & CIA:
  - AI Tools, Team AI, PLN, etc.

IA para Jogos

BIBLIOGRAFIA

- IA "clássica"
  AI: A modern approach. Russell & Norvig, 1995
  Artificial Intelligence. Patrick Winston, 1993.
- Aprendizado de Máquina
  Machine Learning. Tom Mitchell, 1998.
  Sistemas Inteligentes. Solange Rezende, 2003.
  Redes Neurais. Simon Haykin, 2001.
  C4.5: Programs for machine learning. Ross Quinlan, 1993.
- IA para Jogos
  AI for Games and Animation. John Funge, 1999.
  AI Game Programming Wisdom. Steve Rabin (Ed), 2002.
  Computational Principles of Mobile Robotics. Dudek & Jenkin, 2000.
- IA para Jogos e Robótica... na Unisinos
  Robótica Autônoma (TC). Farlei Heinen, 2000.
  Sist. de Controle Híbrido para Robôs Móveis Autônomos (Mestrado). Heinen, 2002.
  Ambiente para Simulação de Múltiplos Agentes Autônomos, Cooperativos e competitivos: MAGES (TC). João Bittencourt, 2002.
  Ambiente Virtual Adaptativo (Mestrado em andamento). Cássia dos Santos, 2003.

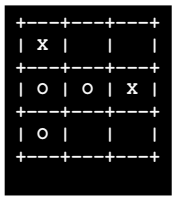
IA para Jogos
Parte I

IA Clássica em Jogos

- **Jogos de Raciocínio**
  - Solução de problemas
    Torre de Hanoi, 8 Puzzles, "brainteasers" (quebra-cabeças), ...
    Busca de Soluções no Espaço de Estados/Configurações
  - Jogos de Tabuleiro:
    Tic-Tac-Toe (Jogo da Velha)
    Connect-4
    Othello
    BackGammon
    Chess
    Go
  - Jogos com Adversários: Game Playing
    "Game as Search Problem"

IA Clássica em Jogos

- **Jogos de Raciocínio**
  - Solução de problemas
    Torre de Hanoi, "brainteasers", ...
    Busca de Soluções no Espaço de Estados/Configurações
  - Jogos de Tabuleiro:
    Tic-Tac-Toe (Jogo da Velha) →



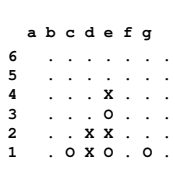
    Connect-4
    Othello
    BackGammon
    Chess
    Go
  - Jogos com Adversários: Game Playing
    "Game as Search Problem"

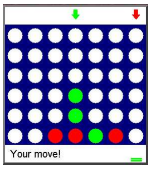
IA Clássica em Jogos

➢ Jogos de Raciocínio

- Solução de problemas
  Torre de Hanoi, "brainteasers, ...
  Busca de Soluções no Espaço de Estados/Configurações
- Jogos de Tabuleiro:
  Tic-Tac-Toe (Jogo da Velha)
  Connect-4
  Othello
  BackGammon
  Chess
  Go

John Tromp





Connect-4 Opening Database - Donated/Created by John Tromp
Contains all legal 8-ply positions in the game of connect-4 in which neither player has won yet, and in which the next move is not forced
Database: 67557 instances, 42 nominal attributes [UCI-ML]

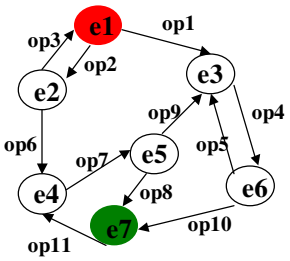
- Jogos com Adversários: Game Playing
  "Game as Search Problem"

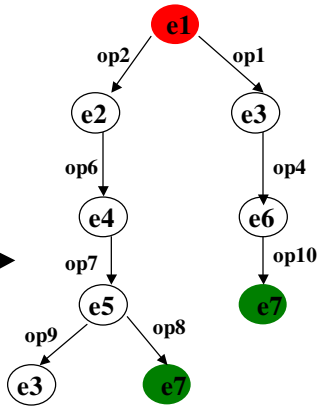
IA Clássica em Jogos

➢ Busca de Soluções no Espaço de Estados

Achar a solução através de uma pesquisa nos possíveis estados do sistema (possíveis estados do sistema = espaço de estados)

- Definição de um problema:
> Estados iniciais (1 ou mais)
> Estados Finais (0 ou mais soluções)
> Operadores que levam de um estado a outro
> Construção de uma "árvore de busca"



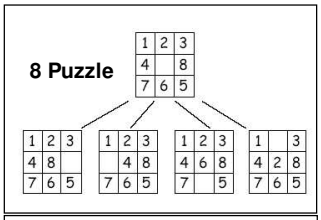


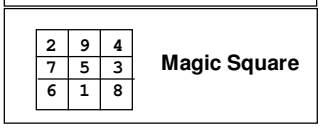
IA Clássica em Jogos

BUSCA EM ESPAÇO DE ESTADOS

Tipos de Busca - *Quanto a estratégia*:

1. **BUSCA CEGA ou NÃO INFORMADA**
   - 1.1. Busca em Largura (Breadth-First)
   - 1.2. Busca em Profundidade (Depth-First)
   - 1.3. Busca Exaustiva (British Museum Search)
2. **BUSCA HEURÍSTICA: A***



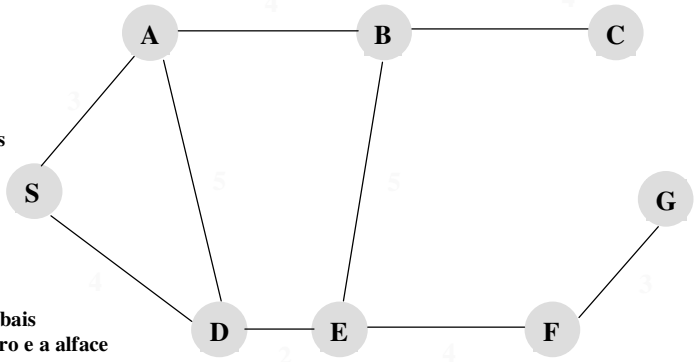


Tipos de Busca - *Quanto ao problema*:

1. **Mecanismo de busca livre: Problemas em geral (quebra-cabeça)**
2. **Mecanismos de busca condicionada: Jogos com mais de 1 jogador (Adversarial Search) Presa as jogadas do oponente**

IA Clássica em Jogos

BUSCA EM ESPAÇO DE ESTADOS:
"No caminho da solução..."



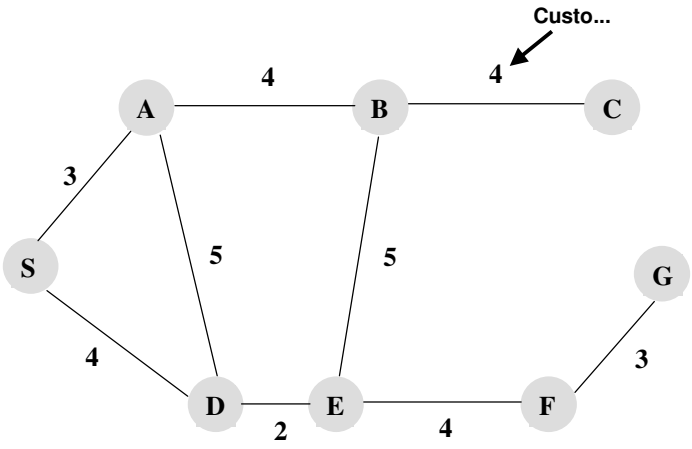
Exemplos de Problemas "Quebra-cabeça":

1. Caixeiro viajante
2. Torre de Hanoi
3. Labirinto
4. Puzzle 8 peças
5. Missionário e os canibais
6. Homem, lobo, carneiro e a alface
7. Problema dos baldes
8. Quadrados mágicos
9. Resta 1
10. Problema do depósito: alocação de espaço

IA Clássica em Jogos

GAME AI

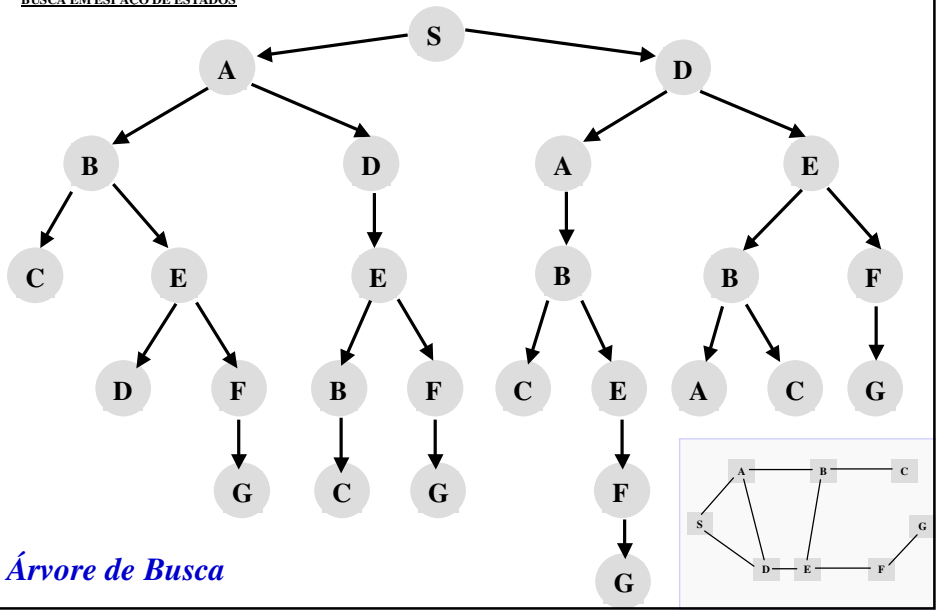
BUSCA EM ESPAÇO DE ESTADOS



IA Clássica em Jogos

GAME AI

BUSCA EM ESPAÇO DE ESTADOS

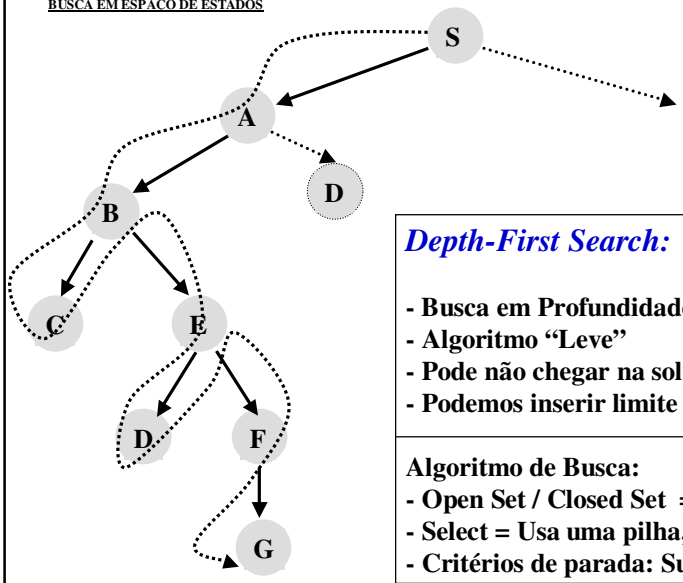


Árvore de Busca

IA Clássica em Jogos

GAME AI

BUSCA EM ESPAÇO DE ESTADOS



Depth-First Search:

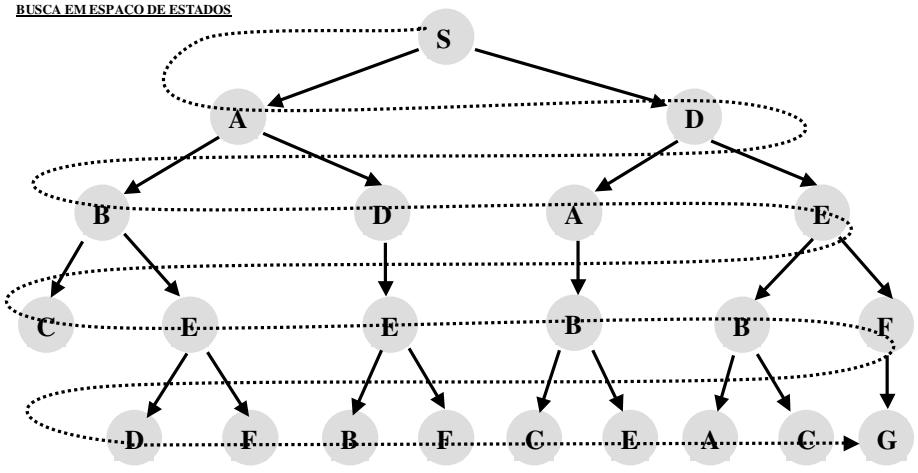
- Busca em Profundidade
- Algoritmo "Leve"
- Pode não chegar na solução
- Podemos inserir limite de profundidade

Algoritmo de Busca:
- Open Set / Closed Set = Pendente / Já visitou
- Select = Usa uma pilha, retira do topo
- Critérios de parada: Sucesso, Profundidade

IA Clássica em Jogos

GAME AI

BUSCA EM ESPAÇO DE ESTADOS



Breadth-First Search:

- Busca em Largura
- Algoritmo "Pesado"
- Deve chegar na solução (não se sabe quando)

Tempo finito?

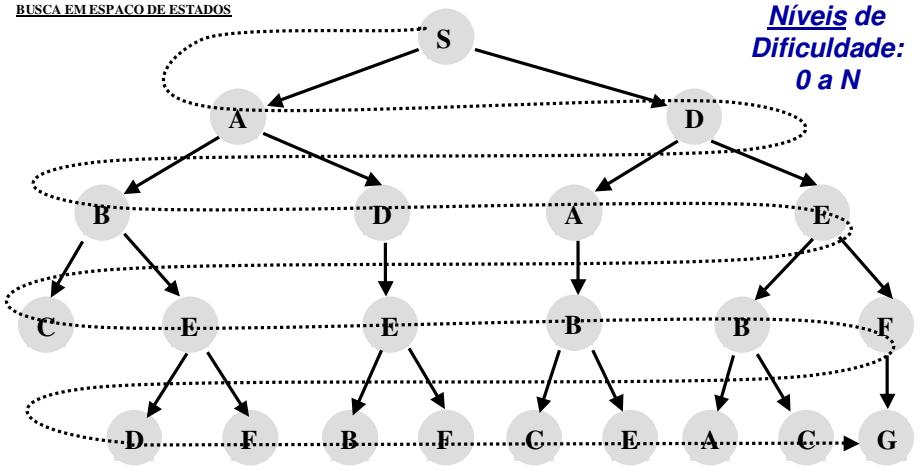
Algoritmo de Busca:
- Open Set / Closed Set
- Select = Usa uma fila, insere no final
- Critérios de parada: Sucesso

IA Clássica em Jogos

GAME AI

BUSCA EM ESPACO DE ESTADOS

Níveis de Dificuldade: 0 a N



Breadth-First Search:

- Busca em Largura
- Algoritmo "Pesado"
- Deve chegar na solução (não se sabe quando)

Tempo finito?

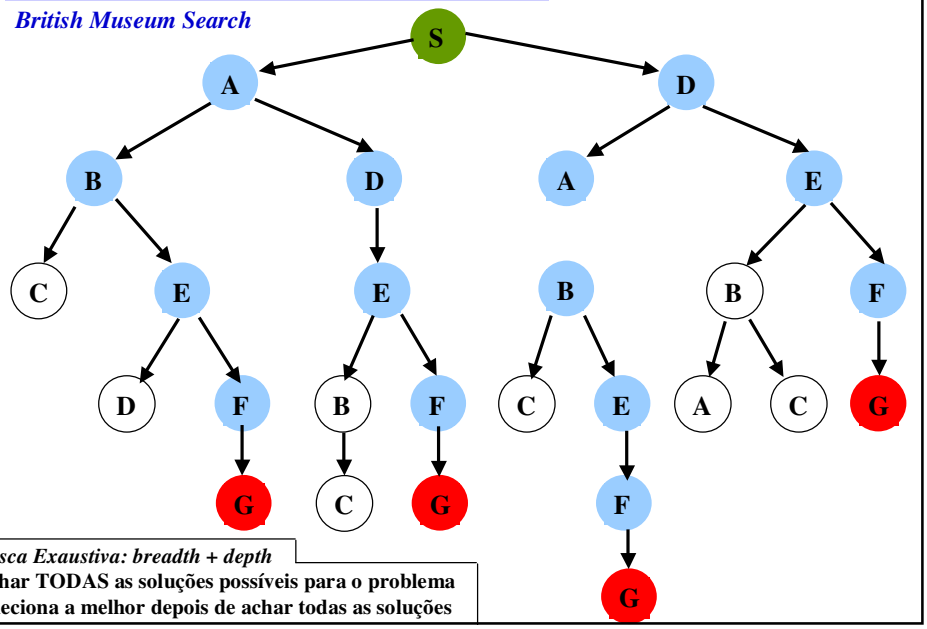
Algoritmo de Busca:

- Open Set / Closed Set
- Select = Usa uma fila, insere no final
- Critérios de parada: Sucesso

IA Clássica em Jogos

GAME AI

British Museum Search



Busca Exaustiva: breadth + depth

Achar TODAS as soluções possíveis para o problema

Seleciona a melhor depois de achar todas as soluções

IA Clássica em Jogos

BUSCA EM ESPAÇO DE ESTADOS

1. Busca Livre em espaço de estados - Problemas / Quebra-cabeças

1.1. Busca Cega
- **1.1.1. Busca em Profundidade (Depth-Search)**
- **1.1.2. Busca em Largura (Breadth-Search)**
- **1.1.3. Busca não determinística (Nondeterministic Search)**
- **1.1.4. Busca exaustiva (British Museum Search - ótima)**

1.2. Busca Heurística
- **1.2.1. Hill Climbing Search**
- **1.2.2. Beam Search**
- **1.2.3. Best-Fisrt Search**
- **1.2.4 Optimal Search**
  - **1.2.4.1. Branch-and-Bound Search**
  - **1.2.4.2. A* Search**

2. Busca Condicionada em espaços de estados - Jogos / Adversário externo

IA Clássica em Jogos

BUSCA EM ESPAÇO DE ESTADOS

➢ **Busca condicionada em Jogos**: ***Trees and Adversarial Search***
"No caminho da vitória..."

- Caminhos possíveis dependem das "reações" do adversário
- Exemplo de jogos tratados pela I.A.:

Jogo da Velha
Gamão
Damas
Xadrez
Go
Othello

- Jogos também são uma procura do caminho em um espaço de estados, onde desejamos seguir o caminho que leva a vitória

- Heurísticas: Avaliar as jogadas (boa, ruim) e a situação/evolução do jogo

Algorítmo mais usados:
- ***Minimax***
- ***Minimax + Alpha-Beta***

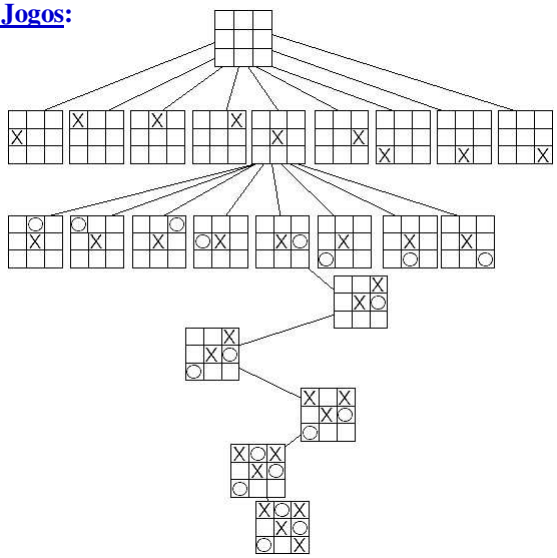
IA Clássica em Jogos

GAME AI

BUSCA EM ESPACO DE ESTADOS

➢ **Busca condicionada em Jogos:**

Jogo da Velha



Algoritmo Minimax

IA Clássica em Jogos

GAME AI

MiniMax Procedure

- Alternância de jogadores

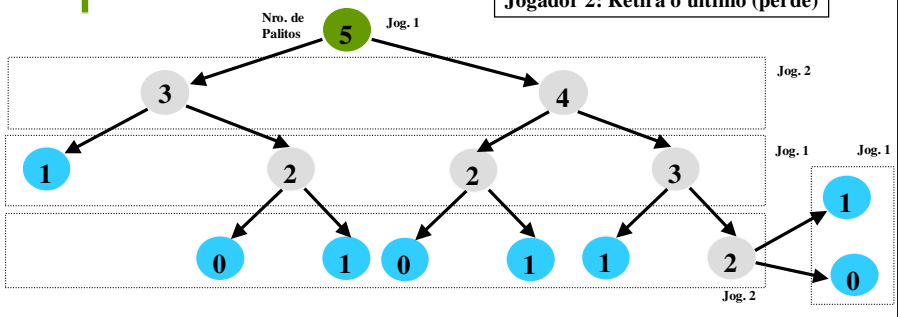
- Construção de uma árvore com camadas alternadas (Mini e Max)

*** Exemplo:**

Jogo dos 5 palitos - Objetivo: pegar 1 ou 2 palitos e não ser o último a jogar

Cenário 1:
Jogador 1: Retira 2 palitos
Jogador 2: Retira 2 palitos
Jogador 1: Retira o último (perde)

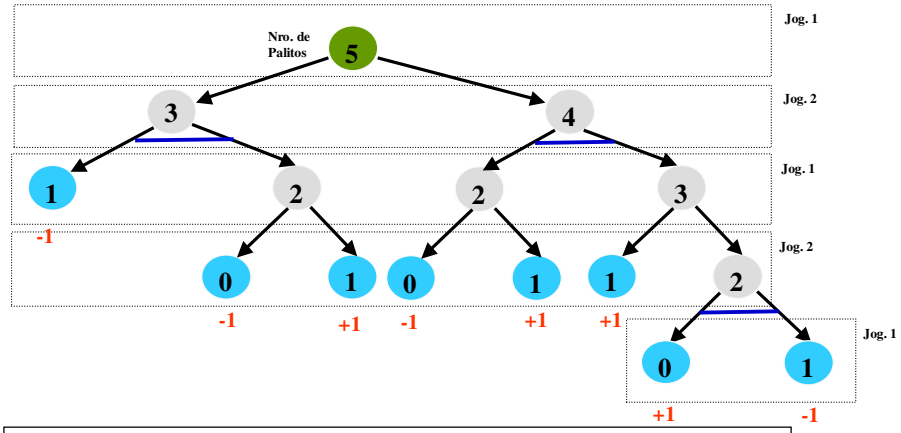
Cenário 2:
Jogador 1: Retira 1 palito
Jogador 2: Retira 2 palitos
Jogador 1: Retira 1 palito
Jogador 2: Retira o último (perde)



IA Clássica em Jogos

GAME AI

MiniMax Procedure

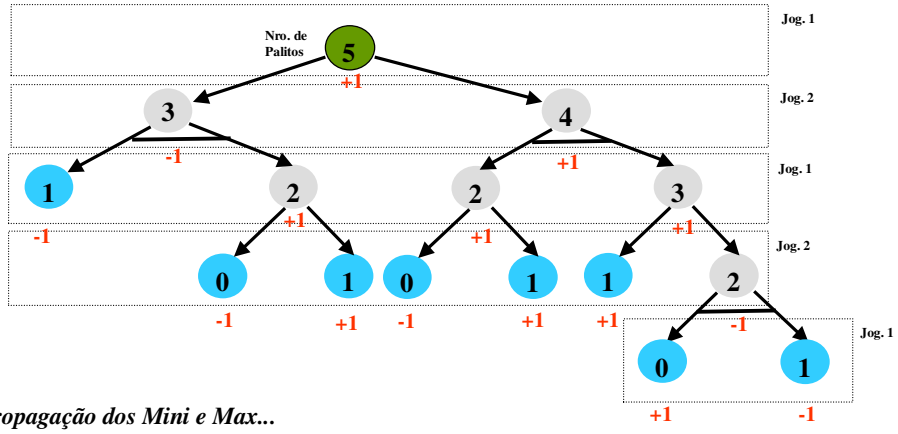


Etapas do algoritmo:
- Pontuação nos nodos terminais da árvore de busca (usualmente: +1, 0, -1)
- Classificar os nodos como do tipo Max (jog. 1 - Livre escolha "ou" =)
ou Mini (jog. 2 - Adversário escolhe =)
- Propagar os mini (menor dos dois filhos) e os max (maior dos dois filhos)

IA Clássica em Jogos

GAME AI

MiniMax Procedure



Propagação dos Mini e Max...

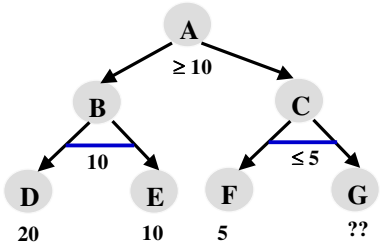
Para obter o melhor caminho, seguir a melhor pontuação!
(Em alguns casos podemos também limitar a profundidade da árvore)

=> Agora você pode fazer o mesmo para o Jogo da Velha!!

IA Clássica em Jogos

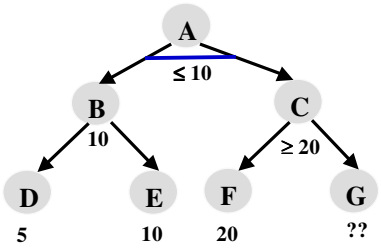
MiniMax Procedure* – *Corte Alpha-Beta

- Limitar a procura tirando proveito das relações existentes entre as sub-árvores
- Procedimentos: Corte Alpha e Corte Beta



Corte Alpha:

Sabendo-se que A recebe o maior entre B e C e que no lado de C existe um valor igual a 5 (logo um valor menor ou igual a 5 será selecionado em C => pois este é um nodo tipo Mini), então não precisamos examinar a sub-arvore G.



Corte Beta:

Sabendo-se que o nodo A é o menor entre B e C, podemos desprezar a sub-árvore G pois esta é certamente maior ou igual a 20, e vamos guardar o menor valor entre B e C

IA Clássica em Jogos

- **Busca condicionada em Jogos:**
  Trees and Adversarial Search

Triunfo da IA:

*** Do Jogo da Velha ao Jogo de Xadrez**
*** Deep Blue / IBM**
http://www.chess.ibm.com/
http://www.inf.unisinos.br/~osorio/protect/iasi/docs/deepblue-faq.pdf

*** Fim do Jogo?**

IA Clássica em Jogos

- **Busca condicionada em Jogos:**
  Trees and Adversarial Search

  Triunfo da IA:

  *** Do Jogo da Velha ao Jogo de Xadrez**
  *** Deep Blue / IBM**
  http://www.chess.ibm.com/
  http://www.inf.unisinos.br/~osorio/protect/iasi/docs/deepblue-faq.pdf

  *** Fim do Jogo?**
  Ainda não...
  - Jogos de Azar e Jogos com Dados (BackGammon)
  - GO: Computer Go tournaments (http://intelligentgo.org/)
  "The most famous, due to its US$1,000,000 prize for winning against a professional player, was the International Go Competition sponsored jointly by Acer Incorporated and the Ing Chang-Ki Wei-Chi (Go) Educational Foundation since 1985. This prize, known as the Ing Prize, unfortunately expired in 2000 and has not been extended."
  - JOGOS DE AÇÃO / INTERATIVOS!

IA para Jogos
Parte II

IA para Jogos

GAME AI

Tópicos abordados... Parte II

- IA clássica: Jogos de Raciocínio [parte I]
  - Solução de problemas
  - Jogos de Tabuleiro (*Board games*)
  - Busca em Espaço de Estados (*Trees and adversarial search*)
- IA clássica: Jogos de Ação em Labirintos [parte II]
  - Busca de caminhos (Path finding)
  - Planejamento de ações e trajetórias (*Path planning*)
  - Deslocamento / Navegação (*Motion & Navigation*)

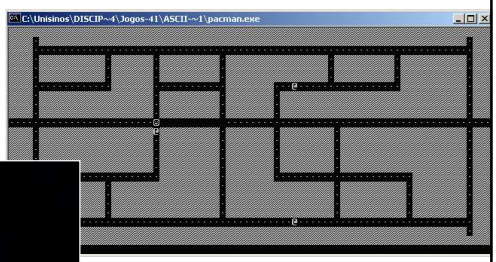
IA para Jogos

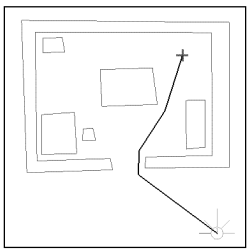
GAME AI

- IA clássica: Jogos de Ação em Labirintos

"Achando o caminho..."

Do Pac-Man a
Robótica Autônoma





Labirinto:
*** Grade**
*** Mapa**

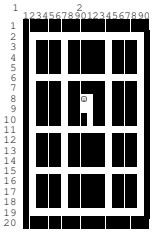
IA para Jogos

- IA clássica: Jogos de Ação em Labirintos

"Achando o caminho..."

Do Pac-Man a
Robótica Autônoma

Path Finding
Path Planning
Motion & Navigation → Algoritmos de Busca



BUSCA CEGA ou NÃO INFORMADA

1. **Busca em Largura (Breadth-First)**
2. **Busca em Profundidade (Depth-First)**
3. **Busca Exaustiva (British Museum Search)**

IA para Jogos

- IA clássica: Jogos de Ação em Labirintos

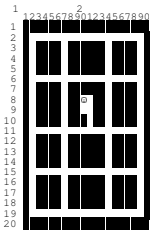
Busca Heurística:

1. Preso no labirinto " atrás das grades": Algoritmo A*

Alguns destes problemas tendem a se tornar intratáveis dependendo do "tamanho" do espaço de estados a ser analisado...

Qual a solução ?

OTIMIZAR = USAR UMA HEURÍSTICA

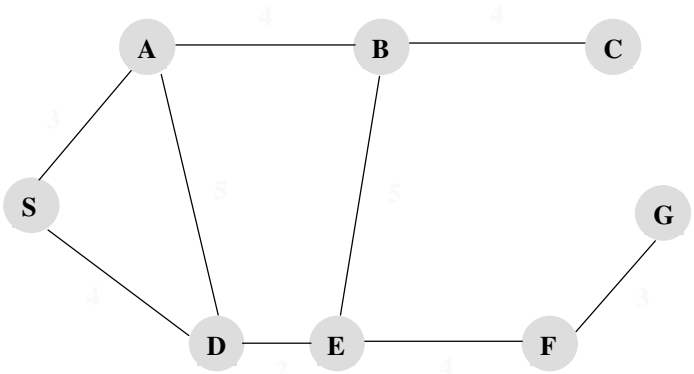


IA para Jogos

GAME AI

- IA clássica: Jogos de Ação em Labirintos

Busca Heurística:

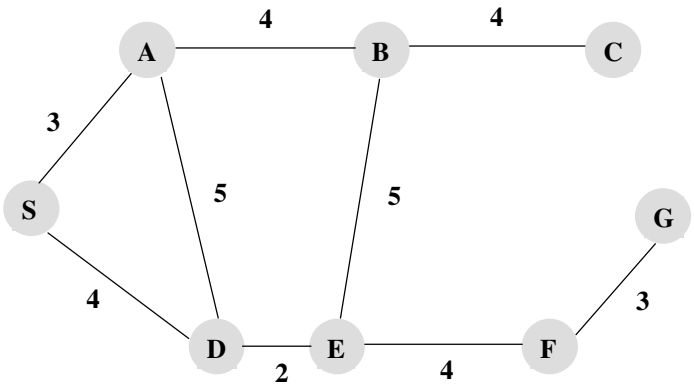


IA para Jogos

GAME AI

- IA clássica: Jogos de Ação em Labirintos

Busca Heurística: **Custo do Caminho**

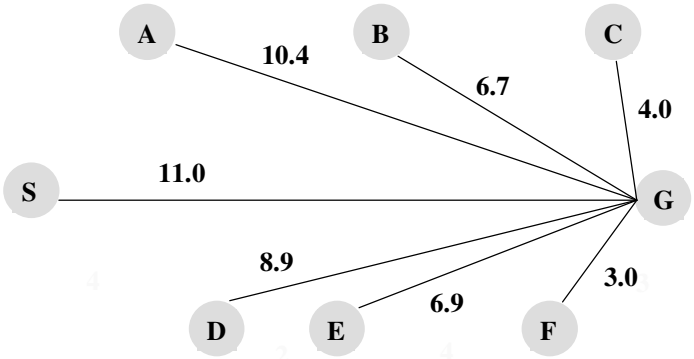


IA para Jogos

GAME AI

- IA clássica: Jogos de Ação em Labirintos

Busca Heurística: **Custo Estimado** (distância em linha reta)



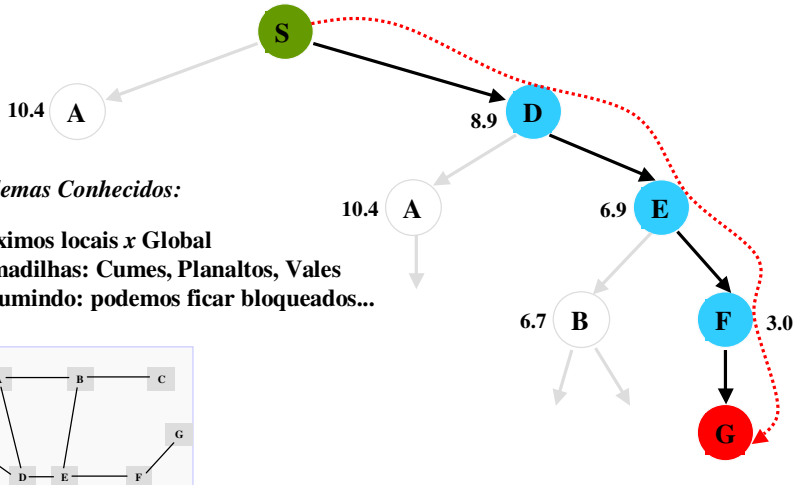
IA para Jogos

GAME AI

Labirinto: Busca Heurística

Hill Climbing Search

- Conhecemos uma informação que permite avaliar os caminhos
- Heurística: depth-first + minimizar o "custo" (distância absoluta)



Problemas Conhecidos:

- Máximos locais *x* Global
- Armadilhas: Cumes, Planaltos, Vales
- Resumindo: podemos ficar bloqueados...

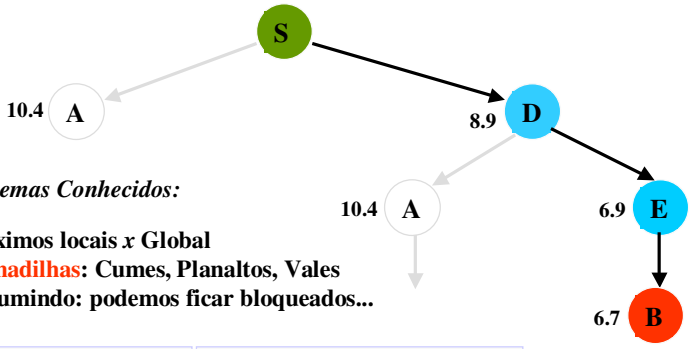
IA para Jogos

Labirinto: Busca Heurística

GAME AI

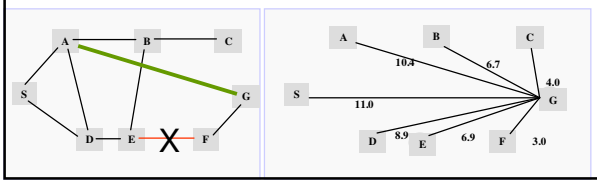
Hill Climbing Search

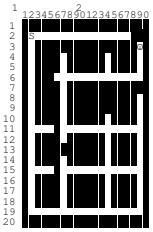
- Conhecemos uma informação que permite avaliar os caminhos
- Heurística: depth-first + minimizar o "custo" (distância absoluta)



Problemas Conhecidos:

- Máximos locais *x* Global
- Armadilhas: Cumes, Planaltos, Vales
- Resumindo: podemos ficar bloqueados...





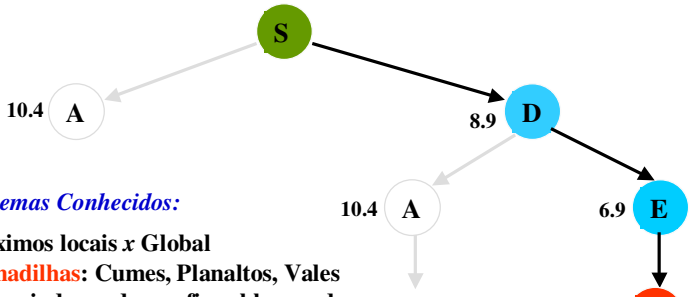
IA para Jogos

Labirinto: Busca Heurística

GAME AI

Hill Climbing Search

- Conhecemos uma informação que permite avaliar os caminhos
- Heurística: depth-first + minimizar o "custo" (distância absoluta)

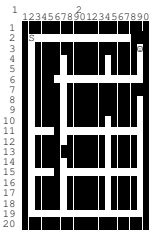


Problemas Conhecidos:

- Máximos locais *x* Global
- Armadilhas: Cumes, Planaltos, Vales
- Resumindo: podemos ficar bloqueados...

Soluções Conhecidas:

- *Beam Search... ("n" melhores)*
- *Best First... (melhor opção)*
- *Branch and Bound* => *A**



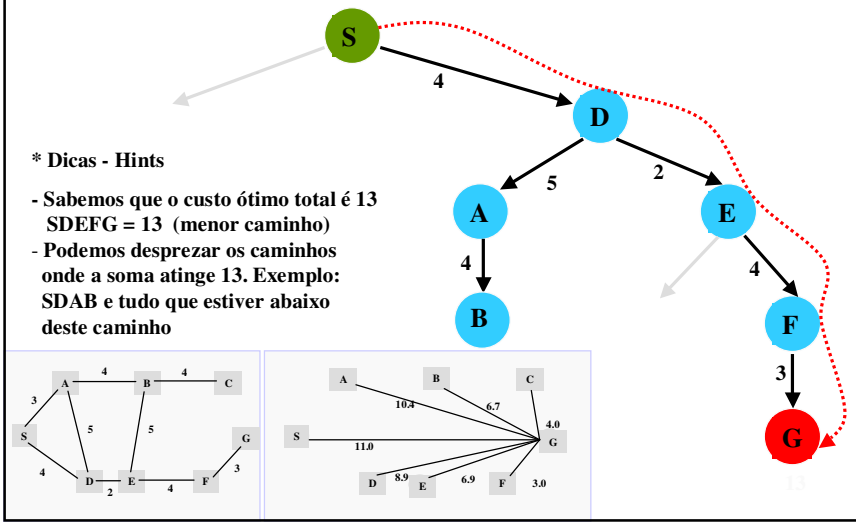
IA para Jogos

Labirinto: Busca Heurística

GAME AI

Branch-and-Bound Search

- Conhecemos uma informação que permite avaliar os caminhos e o custo total
- Heurística: avança e volta caso se "arrependa" do caminho adotado



* Dicas - Hints

- Sabemos que o custo ótimo total é 13
  SDEFG = 13 (menor caminho)
- Podemos desprezar os caminhos
  onde a soma atinge 13. Exemplo:
  SDAB e tudo que estiver abaixo
  deste caminho

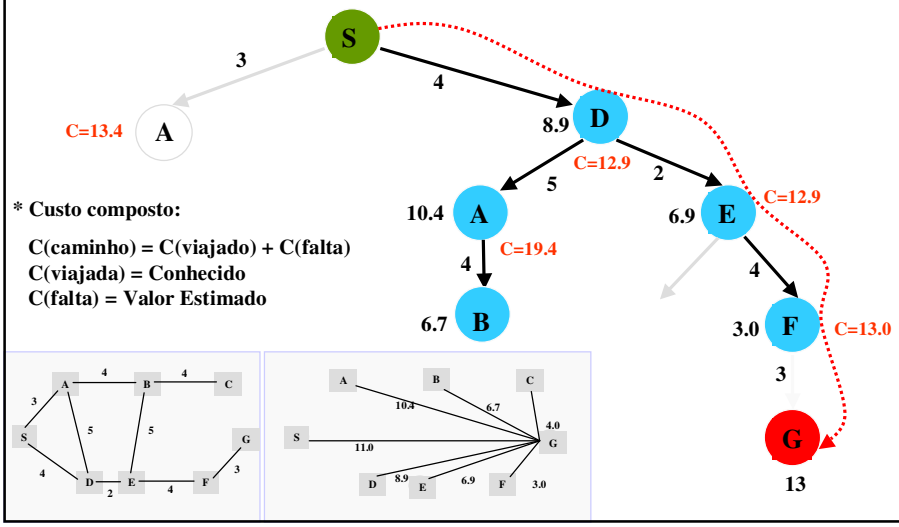
IA para Jogos

Labirinto: Busca Heurística

GAME AI

Branch-and-Bound Search com estimativa

- Conhecemos uma informação que permite avaliar os caminhos e o custo total
- Heurística: avança e volta caso se "arrependa" do caminho adotado



* Custo composto:

C(caminho) = C(viajado) + C(falta)
C(viajada) = Conhecido
C(falta) = Valor Estimado

IA para Jogos

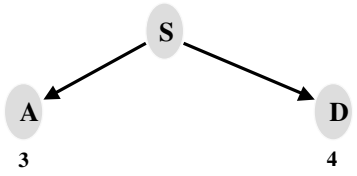
Labirinto: Busca Heurística

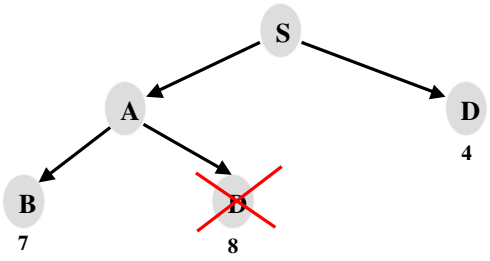
GAME AI

A Search => Select: Sempre busca o melhor da lista Open*

- Heurística: Custo (Caminho) = Custo (Caminho Percorrido)
+ Custo (Caminho Restante)

- Simplificação: eliminar caminhos redundantes
Exemplo => SD... SA...
SD... SAD... (D é novamente usado, caminho maior) SAB...





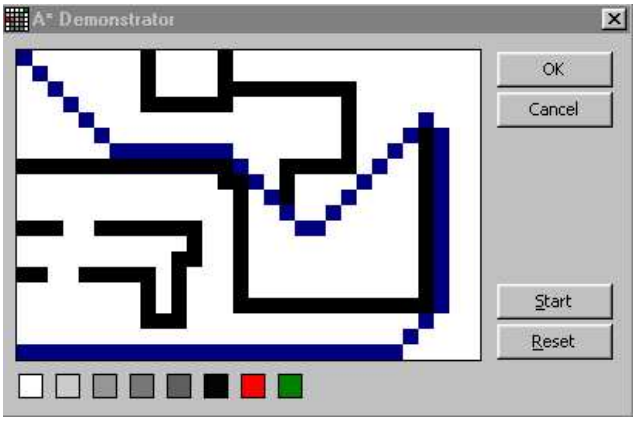
A* Search:
Branch-and-Bound + Simplificação

IA para Jogos

Labirinto: Busca Heurística

GAME AI

A* Search:
O ALGORITMO QUE É A "ESTRELA" DOS JOGOS



http://www.inf.unisinos.br/~osorio/ia.html

IA para Jogos

GAME AI

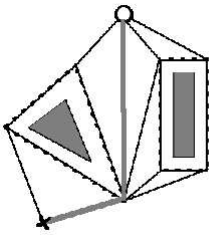
- IA clássica: Jogos de Ação em Labirintos

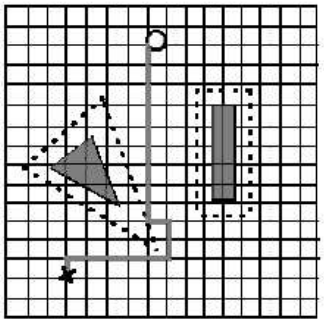
1. Preso no labirinto " atrás das grades":
   Algoritmo A*

2. Explorador com conhecimento do ambiente: "o mapa da mina"
   Grafo de Visibilidade + Caminho Ótimo (Dijkstra)

   * Espaço de configuração
   * Grafo de Visibilidade
   * Caminho ótimo

3. Explorando outros potenciais do mapa...
   - Campos Potenciais
   - Diagramas de Voronoi



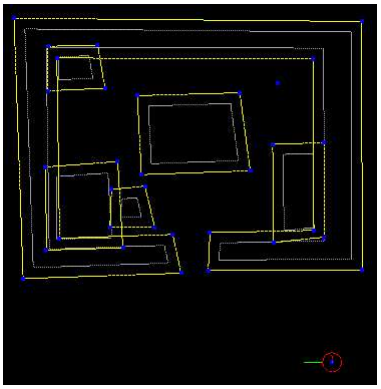


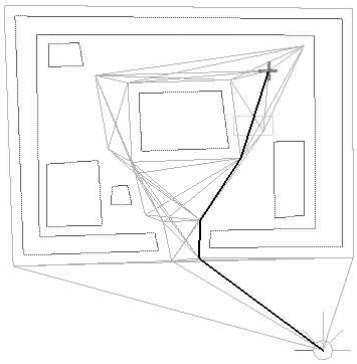
IA para Jogos

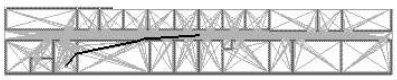
GAME AI

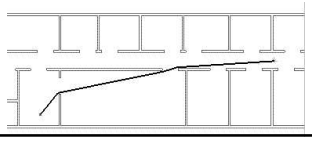
Labirinto: Busca Heurística

Mapa da Geometria do Ambiente:
Grafo de Visibilidade + Caminho Ótimo (Dijkstra)









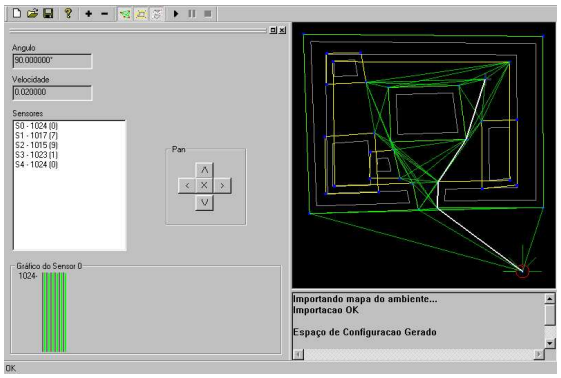
***Referências*: Farlei Heinen**
Robótica Autônoma: A integração entre planificação e comportamento reativo. Editora Unisinos - 2000.
Sist. de Controle Híbrido para RMAs (Mestrado) – 2002.
Web: http://ncg.unisinos.br/robotica/

IA para Jogos

Labirinto: Busca Heurística

GAME AI

Mapa da Geometria do Ambiente:
Grafo de Visibilidade + Caminho Ótimo (Dijkstra)



Jogos:
Grafo de Visibilidade
Pré-calculado

***Referências*: Farlei Heinen**
Robótica Autônoma: A integração entre planificação e comportamento reativo. Editora Unisinos - 2000.
Sist. de Controle Híbrido para RMAs (Mestrado) – 2002.
Web: http://ncg.unisinos.br/robotica/

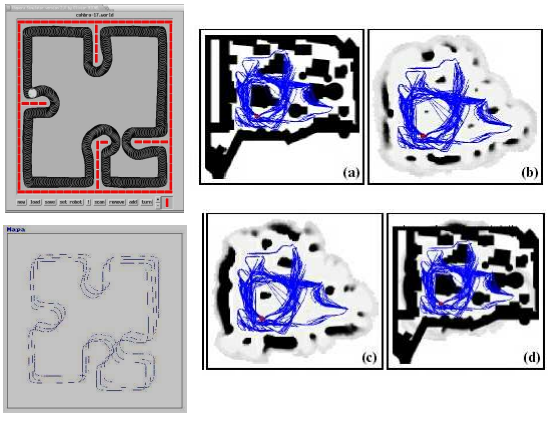
IA para Jogos

Labirinto: Busca Heurística

GAME AI

Ambiente Desconhecido:

SMPA – Sense, Model, Plan, Act **=>** **Jogos: Mundo perfeito**
Construção do Mapa (memória) **Sem ruído, sensores ideais, posição precisa**



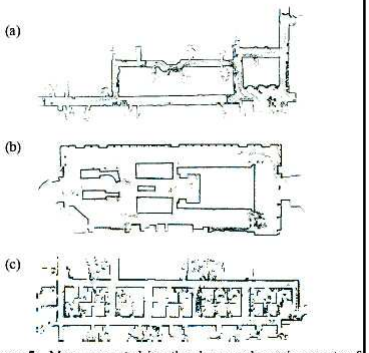


Figure 5: Maps generated in other large-scale environments of sizes (a) 75m, (b) 45m, and (c) 50m. In some of these runs, the cumulative odometric error exceeds 30 meters and 90 degrees.

IA para Jogos

GAME AI

- IA clássica: Jogos de Ação em Labirintos

"Navegar é preciso..." => Deslocando-se nos Labirintos
Execução do planejamento da trajetória.

PROBLEMAS:

*** Desvio de Obstáculos em Robótica**

- Obstáculos conhecidos
- Obstáculos desconhecidos (parados)
- Obstáculos desconhecidos (em movimento)

*** Desvio de Obstáculos em Jogos**

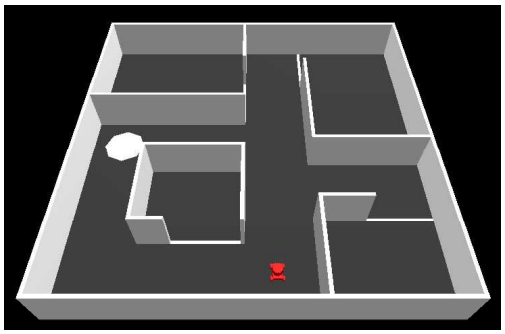
- Obstáculos não definidos no mapa inicial: estáticos (posição conhecida)
- Obstáculos não definidos no mapa inicial: móveis

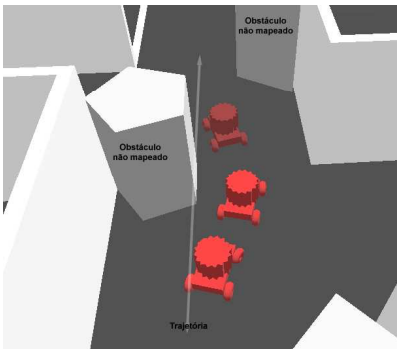
IA para Jogos

GAME AI

- IA clássica: Jogos de Ação em Labirintos

"Navegar é preciso..." => Deslocando-se nos Labirintos
Execução do planejamento da trajetória.





Arquitetura de Controle para Agentes Autônomos...

IA para Jogos

- IA clássica: Jogos de Ação em Labirintos

Volta as origens...

Do Pac-Man a

Robótica Autônoma

Jogos de Raciocínio
- **Solução de problemas**

Jogos de Tabuleiro
- **Jogos com adversários**

Busca no espaço de configurações
- **Busca Cega: Depth, Breadth, British**
- **Busca Condicionada: MiniMax**
- **Busca Heurística: A***

Jogos: Labirintos
- **Grades – A***
- **Mapas – Grafo de visibilidade**
  Caminho ótimo

Jogos e Robótica
- **Planejamento de trajetória**
- **Navegação – Evitar obstáculos**

AGENTES AUTÔNOMOS INTELIGENTES

GAME AI