

Busca Competitiva

Professor: José Eurípedes Ferreira de Jesus Filho
jeferreirajf@gmail.com

Universidade Federal de Goiás – Unidade de Jataí

Agenda

- Introdução.
- Jogos.
- Algoritmo min-max.
- Poda alfa-beta.
- Exercício.

Introdução

- Na aula anterior.
 - PSRs e suas características.
 - Domínios finitos e domínios infinitos.
 - Busca com retrocesso.
 - Retrocesso baseado em conflito.
 - Conjunto de conflito.
 - Várias heurísticas.

Jogos

- Problemas.
 - O que é busca competitiva?
 - Ambiente multi-agente no qual a meta dos agentes estão em conflito.
 - Competição.
 - Jogos.
 - Teoria dos jogos: Ramo da economia.
 - Planejamento com antecedência.
 - Um ou mais agentes também planejando contra nós.
 - O que devemos examinar?
 - Como examinar?
 - Que tipo de busca funcionaria?
 - Como prever o movimento de outros agentes?

Jogos

- Jogos em IA:
 - Determinísticos de revezamento de dois jogadores de **soma zero**.
 - **Informações perfeitas**.
- Quer dizer:
 - Ambientes determinísticos.
 - Completamente observáveis.
 - Dois agentes que alternam ações.
 - Os valores da utilidade no fim do jogo são iguais e opostos.
 - Um deve ganhar (+1) e o outro deve perder (-1).
 - A **competição** vem da oposição da utilidade.

Jogos

- Xadrez;
- Futebol de robôs;
- Damas;
- Cartas;

- Etc...

Jogos

- Precisamos tomar alguma decisão!
 - Em muitos casos o cálculo da decisão ótima é completamente inviável.
 - A árvore de busca para vários jogos possuem fator médio ou alto de ramificação.
 - Problemas difíceis!
 - Ineficiência portanto é extremamente punida.
 - O tempo gasto para a tomada de decisão é um fator importante!
 - Não queremos esperar infinito para que a máquina faça um movimento no xadrez.

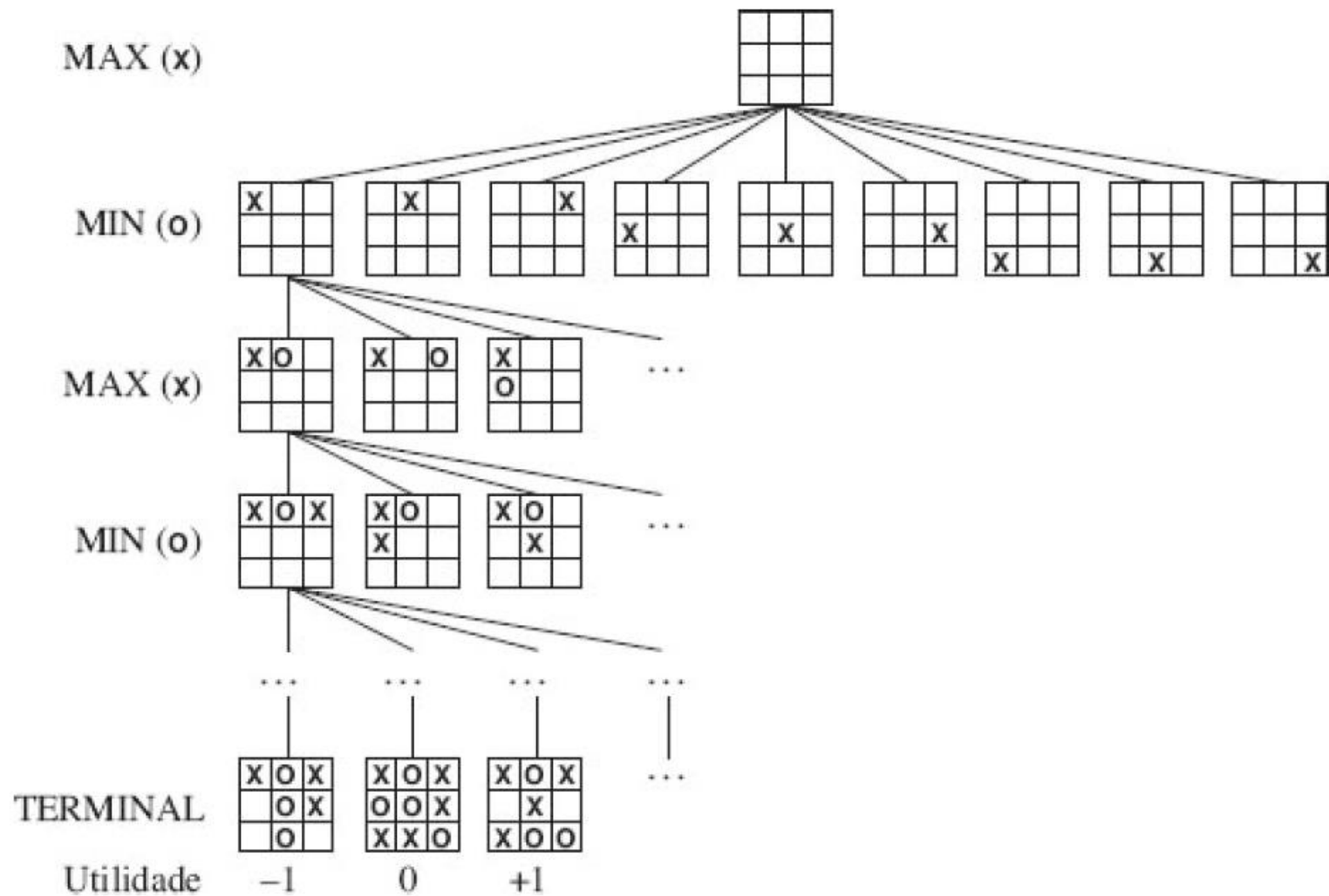
Jogos

- **Definição.**
 - **Estado inicial:** Inclui a posição do tabuleiro e identifica o jogador que fará o primeiro movimento.
 - **Função sucessor:** Retorna uma lista de pares (movimento, estado), cada qual informa um movimento válido e o estado resultante.
 - **Teste de término:** Determina quando o jogo termina. Estados onde o jogo é encerrado são chamados de **estados terminais**.
 - **Função de utilidade:** Dá um valor numérico aos **estados terminais**. Em xadrez, por exemplo, o resultado pode ser vitória (+1), empate (0) ou derrota (-1).

Jogos

- Jogos com dois jogadores.
 - Um se chama **MIN** e o outro **MAX**.
 - Queremos maximizar a recompensa de **MAX** e minimizar a de **MIN**.
 - Desejamos portanto que **MAX** ganhe e **MIN** perca.
 - Queremos encontrar **estratégias** de contingência para MAX em cada estado.
 - Uma **estratégia ótima** é equivalente a um oponente infalível.

Árvore de busca



Jogos

- Definições.

- **Movimento:** Considera-se que **MAX** e **MIN** tenham jogado uma vez.
- **Jogada:** “Meio movimento”.
- **Valor minmax:** Dado a árvore de jogo, é o valor retornado pela árvore de busca que analisa as jogadas da partida. Representa a utilidade para MAX em um dado nó.

$$ValorMinMax(n) = \begin{cases} Utilidade(n), & \text{se } n \text{ é um estado terminal} \\ \max_{a \in Ações(n)} ValorMinMax(Sucessor(a, n)), & \text{se } n \text{ é nó de MAX.} \\ \min_{a \in Ações(n)} ValorMinMax(Sucessor(a, n)), & \text{se } n \text{ é nó de MIN.} \end{cases}$$

Jogos

função DECISÃO-MINIMAX(*estado*) **retorna** *uma ação*
retornar $\arg \max_{a \in \text{Ações}(s)} \text{VALOR-MIN}(\text{RESULTADO}(\text{estado}, a))$

função VALOR-MAX(*estado*) **retorna** *um valor de utilidade*
se TESTE-TERMINAL(*estado*) **então retornar** UTILIDADE(*estado*)
 $v \leftarrow -\infty$
para cada a **em** AÇÕES(*estado*) **faça**
 $v \leftarrow \text{MAX}(v, \text{VALOR-MIN}(\text{RESULTADO}(s, a)))$
retornar v

função VALOR-MIN(*estado*) **retorna** *um valor de utilidade*
se TESTE-TERMINAL(*estado*) **então retornar** UTILIDADE(*estado*)
 $v \leftarrow -\infty$
para cada a **em** AÇÕES(*estado*) **faça**
 $v \leftarrow \text{MIN}(v, \text{VALOR-MAX}(\text{RESULTADO}(s, a)))$
retornar v

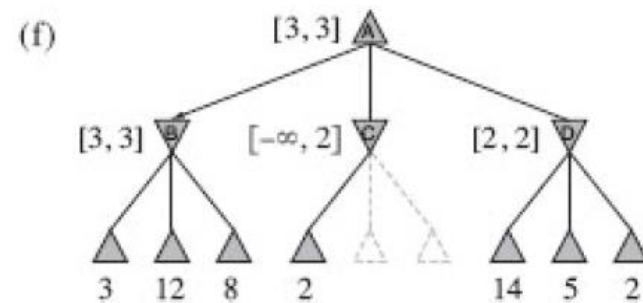
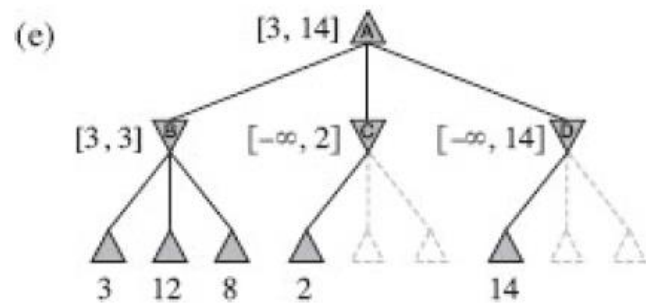
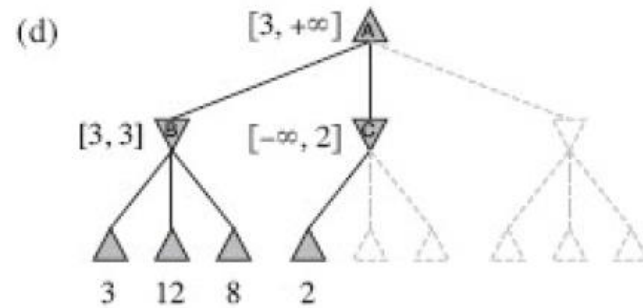
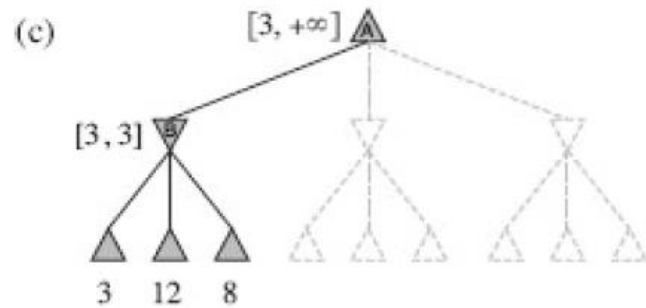
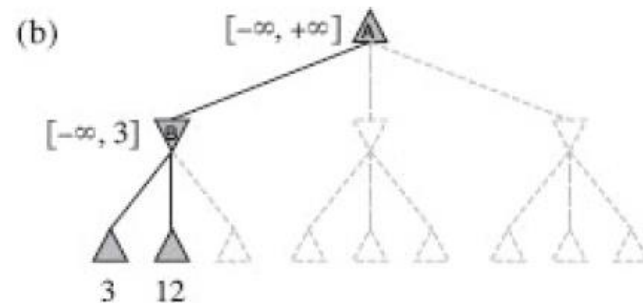
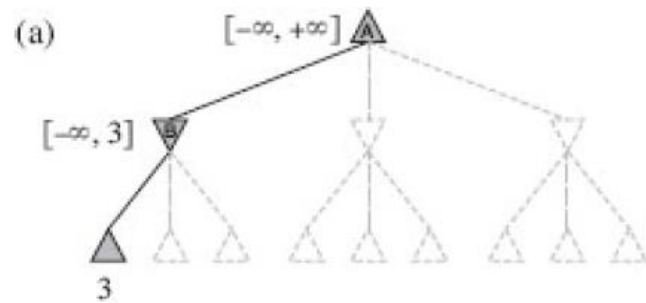
Jogos

- E se tivermos mais de dois jogadores?

Poda alfa-beta

- A poda considera valores máximos e mínimos.
 - Beta: melhor valor possível encontrado pelo caminho para MAX.
 - Alfa: melhor valor possível encontrado pelo caminho para MIN.

Poda alfa-beta



Poda alfa-beta

função BUSCA-ALFA-BETA(*estado*) **retorna** uma ação

$v \leftarrow \text{VALOR-MAX}(\text{estado}, -\infty, +\infty)$

retornar a ação em AÇÕES(*estado*) com valor v

função VALOR-MAX(*estado*, α , β) **retorna** um valor de utilidade

se TESTE-TERMINAL(*estado*) **então retornar** UTILIDADE(*estado*)

$v \leftarrow -\infty$

para cada a , em AÇÕES(*estado*) **faça**

$v \leftarrow \text{MAX}(v, \text{VALOR-MIN}(\text{RESULTADO}(s, a), \alpha, \beta))$

se $v \geq \beta$ **então retornar** v

$\alpha \leftarrow \text{MAX}(\alpha, v)$

retornar v

função VALOR-MIN(*estado*, α , β) **retorna** um valor de utilidade

se TESTE-TERMINAL(*estado*) **então retornar** UTILIDADE(*estado*)

$v \leftarrow +\infty$

para cada a , em AÇÕES(*estado*) **faça**

$v \leftarrow \text{MIN}(v, \text{VALOR-MAX}(\text{RESULTADO}(s, a), \alpha, \beta))$

se $v \leq \alpha$ **então retornar** v

$\beta \leftarrow \text{MIN}(\beta, v)$

retornar v

Exercício

- Qual é o erro no algoritmo busca-alfa-beta do slide anterior?

Próxima aula

- Agentes lógicos.
- **Para casa:**
 - Implemente um algoritmo busca-alfa-beta para o jogo da velha.
 - Entregue no SIGAA.
- **Mini pesquisa:** N/A.