
TRABALHO 1

Aldo Junio Souza Brandão

cc23121@g.unicamp.br

Leonardo Ribeiro Gonzaga Lima

cc23579@g.unicamp.br

Victor Yuji Mimura

cc23158@g.unicamp.br

Sumário

1	Introdução	3
2	Algoritmos	3
3	Metodologia	3
4	Resultados	4
4.1	Metodologia dos Testes	4
5	Conclusão	4
6	Referências	5

1 Introdução

O projeto visa utilizar um algoritmo de busca competitiva, Alpha-Beta Pruning, para implementar o jogo Nim, na variante *misère*, que envolve dois participantes: um ser humano e uma IA. O jogo é organizado em pilhas de 1, 3, 5 e 7 palitos ou contadores.

- Dois participantes irão realizar jogadas alternadamente.
- Em cada jogada, o jogador pode remover qualquer quantidade de contadores, desde que todos sejam da mesma pilha.
- O perdedor é aquele que retirar o último contador dentre todas as pilhas.

2 Algoritmos

O algoritmo Minimax é comumente utilizado em jogos de soma zero. A inteligência artificial assume que o oponente sempre fará a melhor jogada possível, agindo em prol da melhor performance para a máquina ao maximizar suas chances de vitória enquanto minimiza as do oponente.

O Alpha-Beta Pruning é uma otimização do Minimax, pois são descartados os nós de busca que não contribuem para a decisão final, reduzindo assim o tempo de processamento e o número de nós de busca a serem analisados.

3 Metodologia

Para implementar a IA do jogo Nim, seguimos os seguintes passos:

1. Modelagem do jogo e suas regras no código.
2. Implementação do algoritmo Minimax com poda Alpha-Beta.
3. Testes com diferentes cenários para avaliar a IA.
4. Análise dos resultados obtidos e otimização do algoritmo.

4 Resultados

Algoritmo	Oponente	Tempo Médio por Jogada (s)	IA Vencedora
Minimax	IA vs. IA	0.660	IA (Teste)
Minimax + Alpha-Beta	IA vs. IA	0.048	IA (Teste)
Minimax + Alpha-Beta	IA vs. Humano	0.055	IA (Teste)

Tabela 1: Comparação de tempos médios entre os algoritmos Minimax e Minimax com poda Alpha-Beta

4.1 Metodologia dos Testes

- Partidas entre seres humanos e a IA do programa.
- Partidas entre IAs (a IA que começar deve vencer).

5 Conclusão

A implementação do algoritmo Minimax com Alpha-Beta Pruning demonstrou ser eficaz na resolução do jogo Nim, tendo em vista a sua agilidade e sua eficiência. A IA desenvolvida apresentou desempenho superior

ao dos jogadores humanos e mostrou-se consistente em sua jogadas. O uso da poda Alpha-Beta permitiu uma redução significativa no tempo de cálculo, tornando a execução mais eficiente. Na nossa visão como grupo, foi um desafio, tendo em vista que foi nosso primeiro contato com esse algoritmo. Porém acreditamos que conseguimos fazer um bom projeto.

6 Referências

LAGUE, S. Algorithms Explained – minimax and alpha-beta pruning. Disponível em: <https://www.youtube.com/watch?v=l-hh51ncgDI>. Acesso em: 4 abr. 2025.

Python lambda. Disponível em: https://www.w3schools.com/python/python_lambda.asp. Acesso em: 4 abr. 2025.

PYTHON, R. Minimax in Python: Learn how to lose the game of Nim. Disponível em: <https://realpython.com/python-minimax-nim/>. Acesso em: 4 abr. 2025.

Python sorted() function. Disponível em: https://www.w3schools.com/python/ref_func_sorted.asp. Acesso em: 4 abr. 2025.

Alpha-beta pruning. Disponível em: https://en.wikipedia.org/wiki/Alpha%E2%99%A5beta_pruning. Acesso em: 4 abr. 2025.