Analyses of Tapatan and Picaria

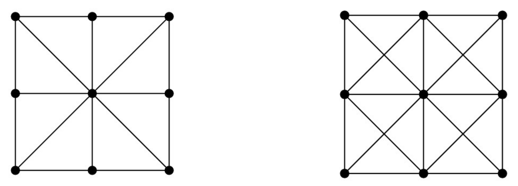
Hope Neveux,[1] Lesley (Jingyi) Wu,[2] and Johanna N.Y. Franklin[3] (advisor)

[1] Marist College, [2] Mount Holyoke College, [3] Hofstra University

Abstract

Markov chains have been used to solve a variety of problems and are especially applicable for investigating abstract game play. We consider the traditional two-player abstract strategy games of Tapatan and Picaria and some new variants from several perspectives: combinatorial exploration of board state spaces, Markov chain analyses of various strategies, and development of optimal strategies where possible.

Game Rules



Standard Tapatan board Standard Picaria board

- Two-player game, 3 stones per player
- Placement phase: alternate placing stones until all 6 are on the board.
- To win, arrange 3 stones in a row.

Board configurations

- All transformations (via rotation and reflection) of a board configuration form a group and are only counted once.
- Given **Burnside's Lemma**, there are $|X/G| = \frac{1}{|G|}\sum_{g\in G}|X^g| = 228$ unique board configurations. Three of those 228 are "double wins," however, these are not possible in Tapatan or Picaria, leaving us **225** possible states.

Markov chains

Markov chains model scenarios with a fixed number of possible *states* that you move between in discrete *steps* such that for any two states σ and τ, the probability of going from σ to τ is constant no matter what happened previously.

A state σ of a Markov chain is *absorbing* if it is impossible to leave.

A Markov chain is absorbing if it has at least one absorbing state and from any state, it is possible to get to at least one absorbing state.

Win rates given minimax algorithm

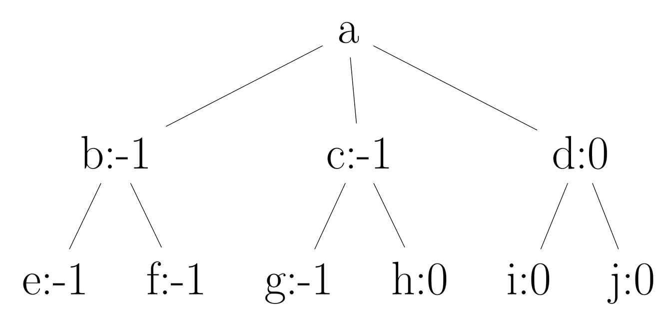
	P1	P2
• Both players playing randomly:	52.7%	47.3%
• P1 thinks ahead 1 move, P2 random:	87.5%	12.5%
• P1 random, P2 thinks ahead 1 move:	22.6%	77.4%
• Both players think ahead 1 move:	65.7%	34.3%

Minimax algorithm

- Inputs: current game state, depth, evaluation function (1 if win, -1 if loss, 0 otherwise)
- Output: Best move(s) from current state

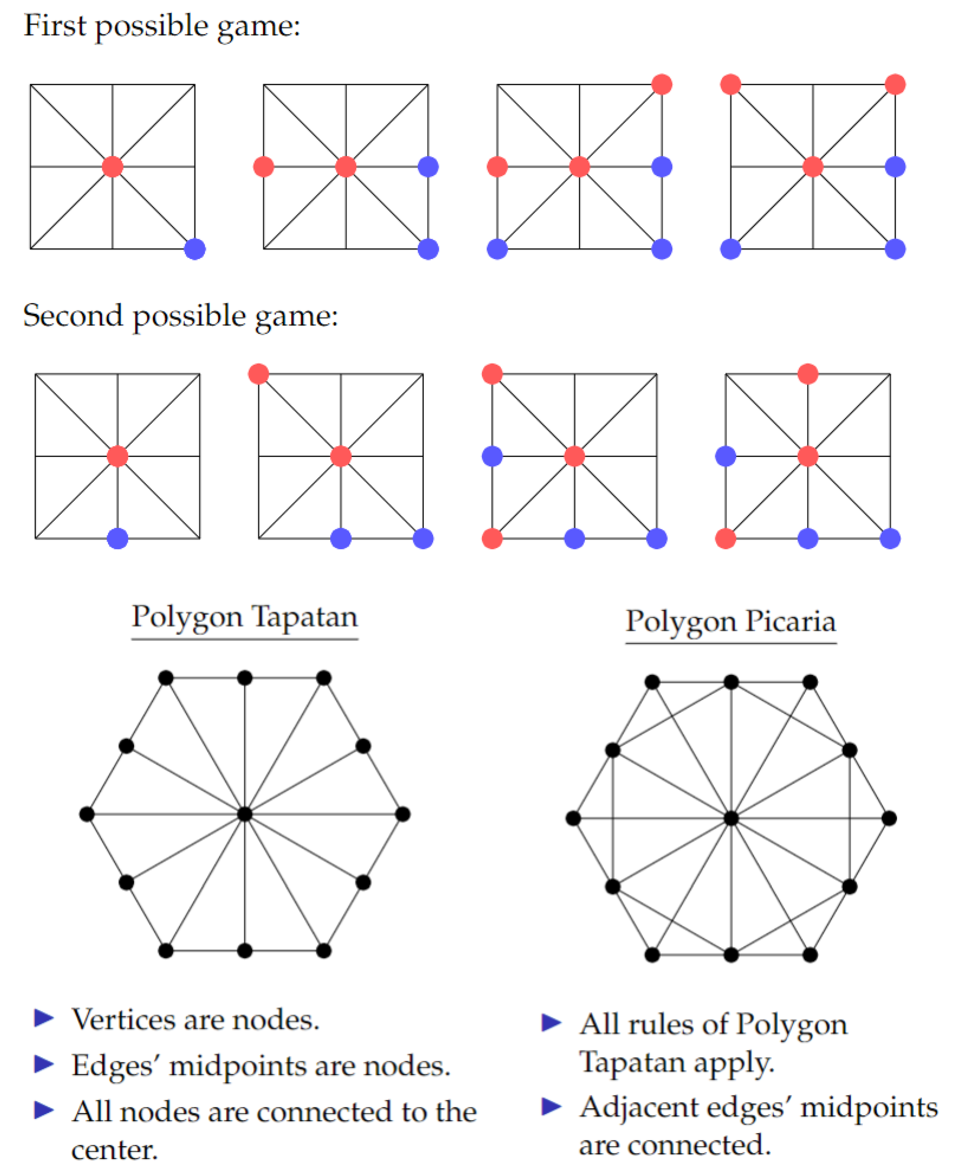
1. Construct search tree of game states to target depth.
2. Assign values to leaves using evaluation function.
3. Assign values to parents using max or min of children (max for player, min for opponent).
4. Return move(s) with maximum value.

Example: Starting from a, and using the default evaluation function and a depth of 2, the algorithm constructs the following search tree. Then the states e, f, g, h, i, and j are evaluated according to the evaluation function.

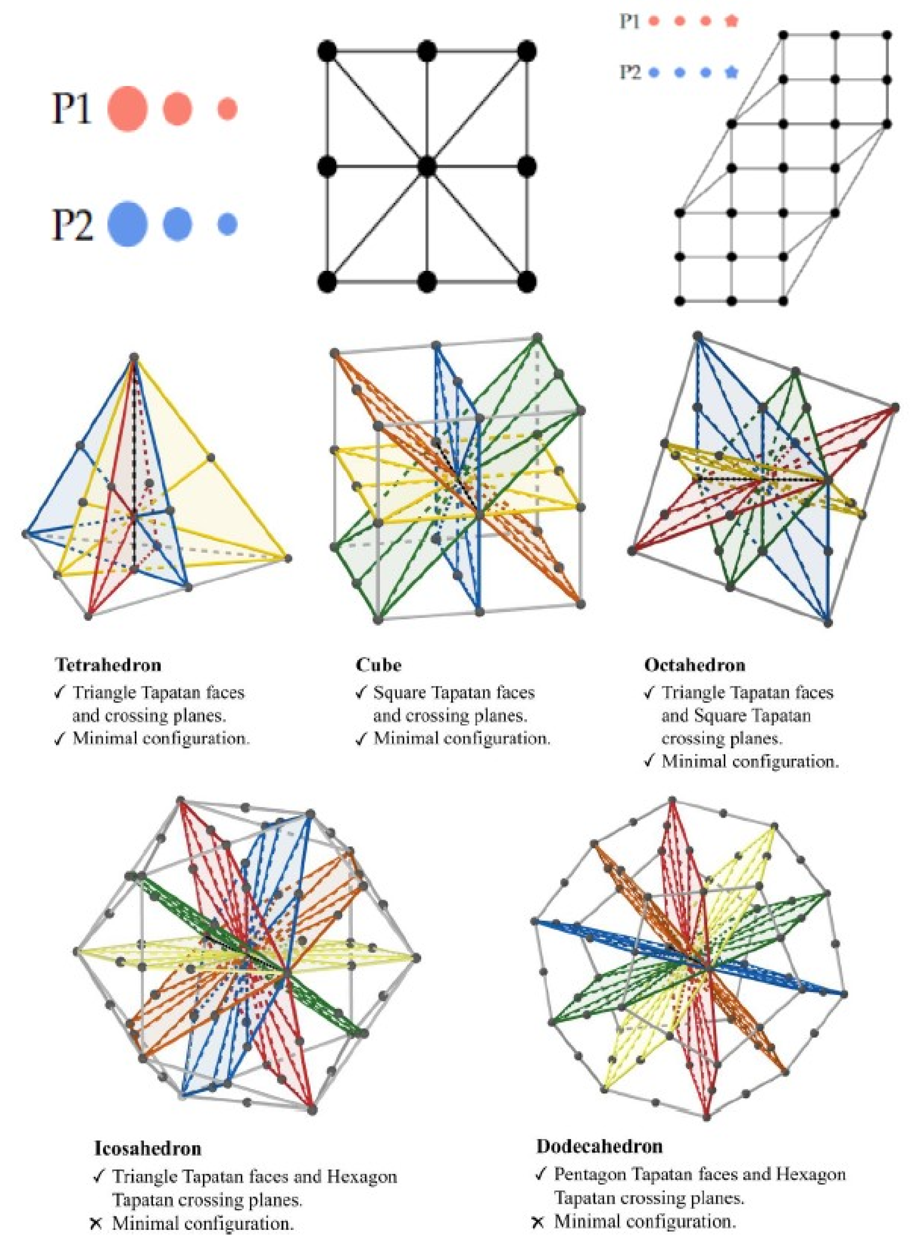


Optimal strategy for polygonal versions

- **Tapatan with $n \geq 3$:** Player 1 has a guaranteed win.
- **Standard (4-)Picaria:** Neither player is guaranteed a win (Larsson and Rocha).
- **Picaria with $n > 4$:** Player 1 has a guaranteed win (originally stated by Larsson and Rocha, proven here).



Alternative versions



Summary

If you have a choice, always choose to be **Player 1**!

- In Tapatan, you're guaranteed a win.
- In Picaria, you're guaranteed at least a draw.

Acknowledgements

This project was collectively carried out by the Markov Chains and Abstract Strategy Games group in the 2021 Polymath Jr. program under the mentorship of Dr. Johanna Franklin.

References

[1] Row games: Tapatan (Achi, Marelle, Tant Fant, Luk tsut k'i), 2010.

[2] Charles M. Grinstead and J. Laurie Snell. Introduction to Probability. American Mathematical Society, 1997.

[3] Urban Larsson and Israel Rocha. Eternal Picaria, 2016.

[4] Stuart Russell and Peter Norvig. Artificial Intelligence: A Modern Approach. Pearson, 4th edition, 2020.