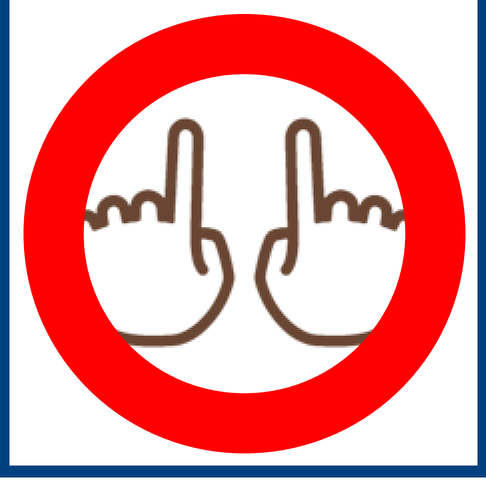
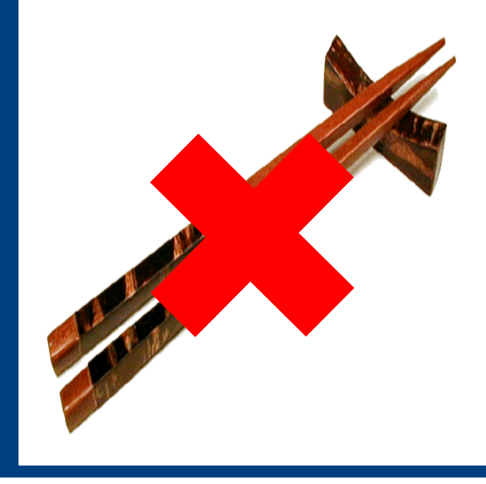


A Study on Winning Strategies of the Generalized Chopsticks Game



Hanseul Cho | KAIST, Math. Sci. (Undergraduate student)

The chopsticks game is a famous traditional two-player finger game. To win the game, you should make the opponent fold all the fingers.

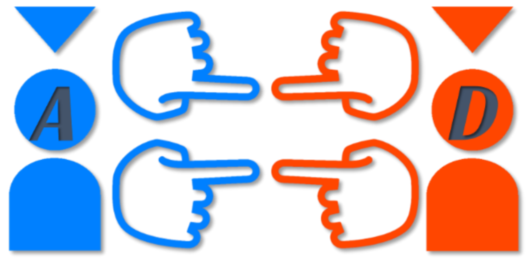
I. Research Objectives

1. We study which of the two players will win in **a conventional game**.
2. We aim to study if there is a winning strategy **as the number of digits is not even five.**
3. We aim to investigate whether there is a winning strategy for players as they **start the game, unlike the traditional rules.**

II. How to play chopsticks game

Conventional (or almost standard) rule of chopsticks game

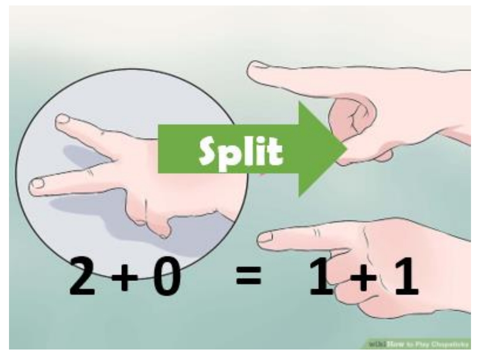
1. When a round starts, both players hold their hands out and extend a finger in each hand. (**(1,1), (1,1)**)



2. Choose a person to go first. Now the person is an **attacker (A)**, and the opponent is a **defender (D).** They will then take turns going back and forth.
3. In every turn, the Attacker has two options: **Hit or Split.**

① **Hit:** Attacker taps one of the defender's non-zero hands with her one hand. **Defender will add the extended finger of the tapping hand and the tapped hand**, and **extend the sum on the tapped hand**. If the sum is 5 or more, the tapped hand becomes a 'dead hand' and will fold all its fingers.

② **Split**: Attacker taps both hands together to **redistribute her own fingers**, but does not just swap both hands. She can even return her dead hand into alive one by splitting.



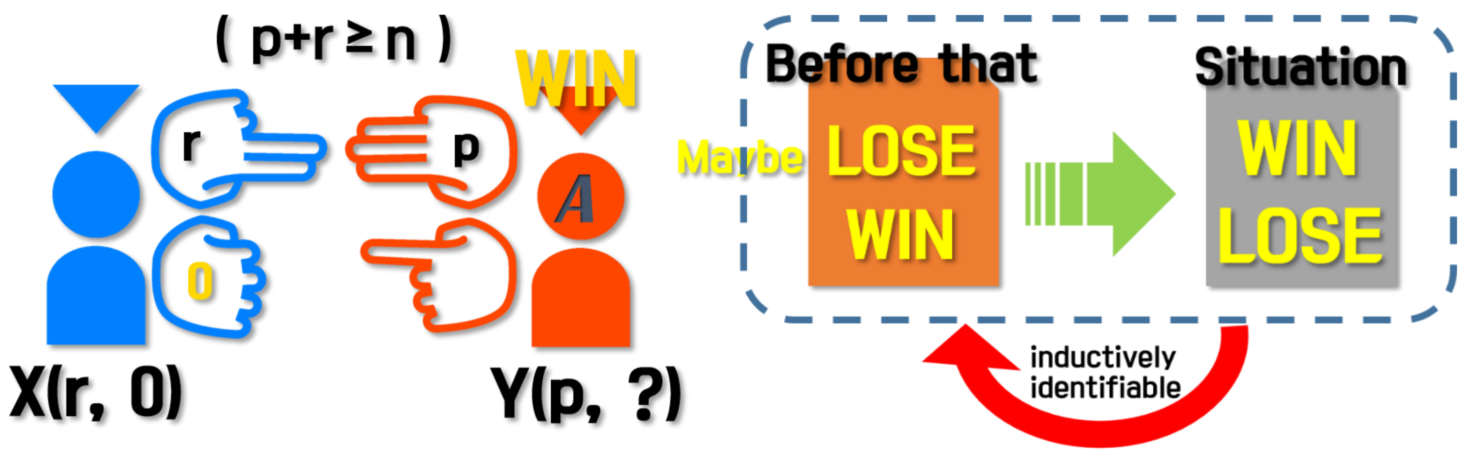
4. To win the game, you should make all of the opponent's hand dead(all fingers are folded).

III. Fundamental Defn. and Thm.

Definitions (No. 1):

- n = minimum number of fingers that makes the hand fold all its fingers. For instance, a conventional game is the game of n=5. …

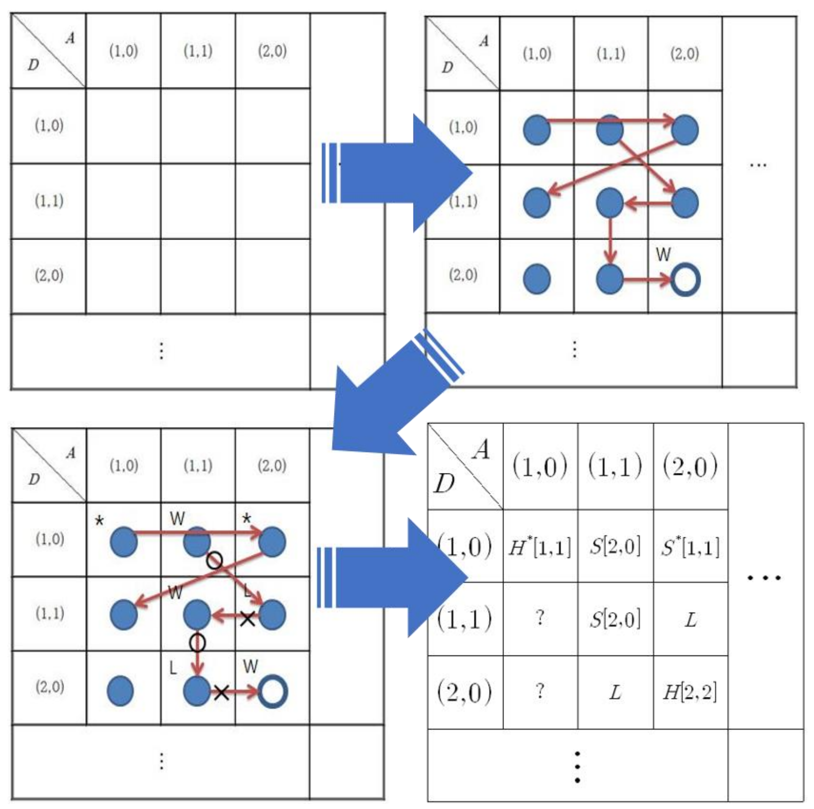
Theorems (No. 1 / No. 2-3):



IV. Strategy Finding Algorithm

- Summary of the Algorithm:

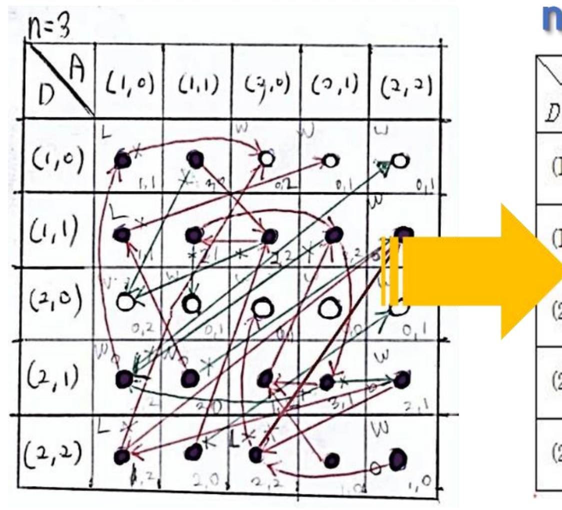
1. Find all possible transitions
2. **Start from the situation that will end in a single turn**
3. Identify the outcome (Win or Lose) of each situation step by step, by **backtracking!**
4. Determine the strategic actions!



(The figure on the left briefly showcases the strategy finding algorithm using a table.)

H[p,q]: 'Hit q with p'; S[p,q]: 'Split into p and q';
L: 'Attacker must Lose', *: 'Tie in the best case'

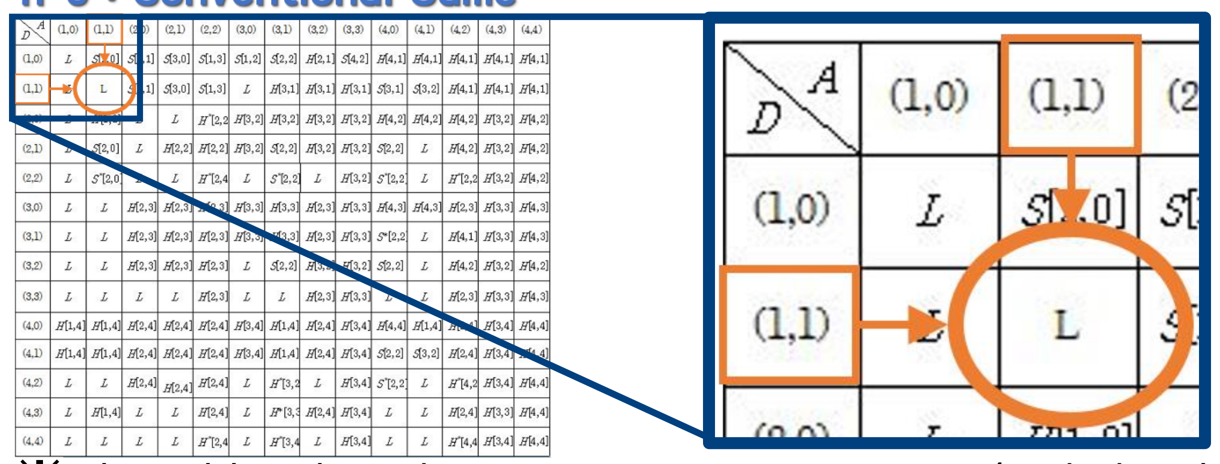
- Examples : n=3, n=5



n=3

D \ A	(1,0)	(1,1)	(2,0)	(2,1)	(2,2)
(1,0)	L	H*[1,1]	H[2,1]	H[2,1]	H[2,1]
(1,1)	L	H*[1,1]	S*[1,1]	H*[1,1]	H[2,1]
(2,0)	H[1,2]	H[1,2]	H[2,2]	H[2,2]	H[2,2]
(2,1)	H[1,2]	H[1,2]	S*[1,1]	H*[2,1]	H[2,1]
(2,2)	L	H*[1,2]	L	H*[2,2]	H[2,2]

n=5 : Conventional Game



※ This table is based on computer programming. (coded with C)

V. Conclusions

1. We showed that the **second player will win in the conventional game** if both players act strategically.
2. We devised an algorithm using a **strategy table** that shows the outcome and strategy of each situation.
3. We observed that there are some rare **situations in which both players are tied**. If you start from those situations, you will enter the infinite loop, which consists of 3 or more situations.
4. (Additional) We rigorously proved some special cases of outcomes with our tabular method.