Combinatorial Aspects of the Card Game War

Atharva Pathak
Mentor: Tanya Khovanova

MIT PRIMES Conference

16 October 2021

Preliminaries

- Two players, Alice and Bob

Preliminaries

- Two players, Alice and Bob
- A finite deck of cards labelled 1 through n

Preliminaries

- Two players, Alice and Bob
- A finite deck of cards labelled 1 through n
- The deck is shuffled and divided among Alice and Bob, which they keep in their **stacks**.

Preliminaries

- Two players, Alice and Bob
- A finite deck of cards labelled 1 through n
- The deck is shuffled and divided among Alice and Bob, which they keep in their **stacks**.
- A **round** consists of the players drawing a card from the top of their stacks. The player with the higher card wins the round and gets both cards.

Preliminaries

- Two players, Alice and Bob
- A finite deck of cards labelled 1 through n
- The deck is shuffled and divided among Alice and Bob, which they keep in their **stacks**.
- A **round** consists of the players drawing a card from the top of their stacks. The player with the higher card wins the round and gets both cards.
- The cards are returned to the bottom of a player's stack according to a *putback* rule

Preliminaries

- Two players, Alice and Bob
- A finite deck of cards labelled 1 through n
- The deck is shuffled and divided among Alice and Bob, which they keep in their **stacks**.
- A **round** consists of the players drawing a card from the top of their stacks. The player with the higher card wins the round and gets both cards.
- The cards are returned to the bottom of a player's stack according to a *putback* rule
  - **random** putback, where the two cards are put back in either order randomly

Preliminaries

- Two players, Alice and Bob
- A finite deck of cards labelled 1 through n
- The deck is shuffled and divided among Alice and Bob, which they keep in their **stacks**.
- A **round** consists of the players drawing a card from the top of their stacks. The player with the higher card wins the round and gets both cards.
- The cards are returned to the bottom of a player's stack according to a *putback* rule
  - **random** putback, where the two cards are put back in either order randomly
  - **WL**-putback, where the winning card is put back before the losing card

Preliminaries

- Two players, Alice and Bob
- A finite deck of cards labelled 1 through n
- The deck is shuffled and divided among Alice and Bob, which they keep in their **stacks**.
- A **round** consists of the players drawing a card from the top of their stacks. The player with the higher card wins the round and gets both cards.
- The cards are returned to the bottom of a player's stack according to a *putback* rule
  - **random** putback, where the two cards are put back in either order randomly
  - **WL**-putback, where the winning card is put back before the losing card
- Game ends when a player has no cards left

Preliminaries

- We represent a *state* of the game as

$$a_1 a_2 \ldots a_i | a_{i+1} a_{i+2} \ldots a_n,$$

where $a_1 \ldots a_i$ represent Alice's stack from top to bottom and $a_{i+1} \ldots a_n$ represents Bob's stack from top to bottom.

Preliminaries

- We represent a *state* of the game as

$$a_1 a_2 \dots a_i | a_{i+1} a_{i+2} \dots a_n,$$

where $a_1 \dots a_i$ represent Alice's stack from top to bottom and $a_{i+1} \dots a_n$ represents Bob's stack from top to bottom.

- E.g. 2|134 with WL-putback:

$$\begin{aligned} & 2 \mid 134 \\ \Longrightarrow\ & 21 \mid 34 \\ \Longrightarrow\ & 1 \mid 432 \\ \Longrightarrow\ & \mid 3241. \end{aligned}$$

Passthroughs

Definition (Passthrough)

Given a state where a player has m cards, their stack undergoes a **passthrough** (PT) after m rounds. These m rounds occur during the passthrough.

Passthroughs

Definition (Passthrough)

Given a state where a player has m cards, their stack undergoes a **passthrough** (PT) after m rounds. These m rounds occur during the passthrough.

Example:

- Again consider $2|134$.

Passthroughs

Definition (Passthrough)

Given a state where a player has m cards, their stack undergoes a **passthrough** (PT) after m rounds. These m rounds occur during the passthrough.

Example:

- Again consider $2|134$.
- $2|134 \implies 21|34 \implies 1|432 \implies |3241$

Passthroughs

Definition (Passthrough)

Given a state where a player has m cards, their stack undergoes a **passthrough** (PT) after m rounds. These m rounds occur during the passthrough.

Example:

- Again consider 2|134.
- $2|134 \implies 21|34 \implies 1|432 \implies |3241$
- Bob has won on his first passthrough

Passthroughs

Definition (Passthrough)

Given a state where a player has m cards, their stack undergoes a **passthrough** (PT) after m rounds. These m rounds occur during the passthrough.

Example:

- Again consider 2|134.
- $2|134 \implies 21|34 \implies 1|432 \implies |3241$
- Bob has won on his first passthrough
- Alice lost on her second passthrough

Passthroughs

Definition (Passthrough)

Given a state where a player has m cards, their stack undergoes a **passthrough** (PT) after m rounds. These m rounds occur during the passthrough.

Example:

- Again consider 2|134.
- $2|134 \implies 21|34 \implies 1|432 \implies |3241$
- Bob has won on his first passthrough
- Alice lost on her second passthrough
- **We only consider games where Bob wins within his first passthrough.**

Visualizing Passthroughs

(Source: commons.wikimedia.org/wiki/File:Customer_divider_bar_1.jpg)

Blocks

Definition (Block)

A block is a subset of cards which may have played in rounds against each other but haven't played in rounds against any other cards.

Blocks

Definition (Block)

A block is a subset of cards which may have played in rounds against each other but haven't played in rounds against any other cards.

Example:

- Take the state 2|134

Blocks

Definition (Block)

A block is a subset of cards which may have played in rounds against each other but haven't played in rounds against any other cards.

Example:

- Take the state 2|134
- Initially each card is in its own block.

Blocks

Definition (Block)

A block is a subset of cards which may have played in rounds against each other but haven't played in rounds against any other cards.

Example:

- Take the state 2|134
- Initially each card is in its own block.
- When Alice wins the first round, cards 1 and 2 form a block, while 3 and 4 are still in their own blocks.

Blocks

Definition (Block)

A block is a subset of cards which may have played in rounds against each other but haven't played in rounds against any other cards.

Example:

- Take the state 2|134
- Initially each card is in its own block.
- When Alice wins the first round, cards 1 and 2 form a block, while 3 and 4 are still in their own blocks.
- When this game ends, there is only one block consisting of all the cards.

Blocks

Definition (Block)

A block is a subset of cards which may have played in rounds against each other but haven't played in rounds against any other cards.

Example:

- Take the state 2|134
- Initially each card is in its own block.
- When Alice wins the first round, cards 1 and 2 form a block, while 3 and 4 are still in their own blocks.
- When this game ends, there is only one block consisting of all the cards.
- If instead the game started as 21|34, there would be two blocks at the end.

Blocks

Definition (Block)

A block is a subset of cards which may have played in rounds against each other but haven't played in rounds against any other cards.

Example:

- Take the state 2|134
- Initially each card is in its own block.
- When Alice wins the first round, cards 1 and 2 form a block, while 3 and 4 are still in their own blocks.
- When this game ends, there is only one block consisting of all the cards.
- If instead the game started as 21|34, there would be two blocks at the end.
- **Each block acts as its own mini-game of War.**

Level-k Single-Use States

Definition (Level-k Single-Use States)

A level-k single-use state is an initial state with WL-putback from which Bob wins during his first passthrough and Alice undergoes at most k passthroughs.

Level-k Single-Use States

Definition (Level-k Single-Use States)

A level-k single-use state is an initial state with WL-putback from which Bob wins during his first passthrough and Alice undergoes at most k passthroughs.

2|134 is level-2 single-use.

Level-k Single-Use States

Definition (Level-k Single-Use States)

A level-k single-use state is an initial state with WL-putback from which Bob wins during his first passthrough and Alice undergoes at most k passthroughs.

2|134 is level-2 single-use.

Note that in these single-use states, the order Bob puts his cards back doesn't matter because they don't show up in the game again. We can regard cards Bob wins as discarded.

Blocks in Single-Use States

- Consider a state $a_1 a_2 \ldots a_m | a_{m+1} \ldots a_n$, where $n \geq m \cdot 2^k$. If this is a level-k single-use state, we have the following:

Blocks in Single-Use States

- Consider a state $a_1 a_2 \ldots a_m | a_{m+1} \ldots a_n$, where $n \geq m \cdot 2^k$. If this is a level-k single-use state, we have the following:

Proposition

No two of Alice's initial cards ever come in a single block.

Blocks in Single-Use States

- Consider a state $a_1 a_2 \dots a_m | a_{m+1} \dots a_n$, where $n \geq m \cdot 2^k$. If this is a level-k single-use state, we have the following:

Proposition

No two of Alice's initial cards ever come in a single block.

- We call a block $a_1 | a_2 \dots a_{2^k}$ a level-k block if it is a level-k single-use state when played as a game of War.

Blocks in Single-Use States

- Consider a state $a_1 a_2 \ldots a_m | a_{m+1} \ldots a_n$, where $n \geq m \cdot 2^k$. If this is a level-k single-use state, we have the following:

Proposition

No two of Alice's initial cards ever come in a single block.

- We call a block $a_1 | a_2 \ldots a_{2^k}$ a level-k block if it is a level-k single-use state when played as a game of War.
- From now on we only discuss blocks where Alice has a single card.

Probability that a State is Level-k Single-Use

Theorem

The chance a random permutation of the 2^k cards in a state $a_1|a_2 \dots a_{2^k}$ is a level-k block is P_k, where $P_1 = \frac{1}{2}$ and recursively

$$P_{k+1} = \frac{1}{2} + \frac{1}{2} P_k^2.$$

Probability that a State is Level-k Single-Use

Theorem

The chance a random permutation of the 2^k cards in a state $a_1|a_2 \dots a_{2^k}$ is a level-k block is P_k, where $P_1 = \frac{1}{2}$ and recursively

$$P_{k+1} = \frac{1}{2} + \frac{1}{2}P_k^2.$$

The first few terms in this sequence are $P_2 = \frac{5}{8}$, $P_3 = \frac{89}{128}$, $P_4 = \frac{24305}{32768}$.

Win-Loss Sequences

Definition (Win-Loss Sequence)

A win-loss sequence is a string of W's and L's describing the progress of the game from the point of view of Alice; W's are wins and L's are losses for her.

Win-Loss Sequences

Definition (Win-Loss Sequence)

A win-loss sequence is a string of W's and L's describing the progress of the game from the point of view of Alice; W's are wins and L's are losses for her.

We can mark off passthroughs of Alice's stack with $/$'s in the win-loss sequence.

Win-Loss Sequences

Definition (Win-Loss Sequence)

A win-loss sequence is a string of W's and L's describing the progress of the game from the point of view of Alice; W's are wins and L's are losses for her.

We can mark off passthroughs of Alice's stack with $/$'s in the win-loss sequence.

Example:

- Take 2|134 with WL-putback.

Win-Loss Sequences

Definition (Win-Loss Sequence)

A win-loss sequence is a string of W's and L's describing the progress of the game from the point of view of Alice; W's are wins and L's are losses for her.

We can mark off passthroughs of Alice's stack with $/$'s in the win-loss sequence.

Example:

- Take 2|134 with WL-putback.
- 2|134 $\Longrightarrow$ 21|34 $\Longrightarrow$ 1|432 $\Longrightarrow$ |3241

Win-Loss Sequences

Definition (Win-Loss Sequence)

A win-loss sequence is a string of W's and L's describing the progress of the game from the point of view of Alice; W's are wins and L's are losses for her.

We can mark off passthroughs of Alice's stack with $/$'s in the win-loss sequence.

Example:

- Take 2|134 with WL-putback.
- $2|134 \implies 21|34 \implies 1|432 \implies |3241$
- So win-loss sequence is W/LL.

More on Win-Loss Sequences

- Before each subsequent passthrough, the number of cards Alice has is twice the number of wins she had in the previous passthrough, because each win yields two cards back to Alice's stack

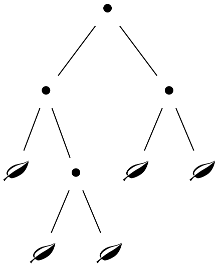
Theorem

Win-loss sequences are in bijection with full binary trees.

Full Binary Trees

Definition (Full Binary Tree)

A full binary tree (FBT) is a binary tree in which every node has either 2 children (left child and right child) or 0 children. A node with 0 children is called a **leaf**, and a node with 2 children is called a **non-leaf**.



Expressing Win-Loss Sequences as Full Binary Trees

Structure of full binary trees and win-loss sequences is the same:

- Each non-leaf yields two nodes in the next level,

Expressing Win-Loss Sequences as Full Binary Trees

Structure of full binary trees and win-loss sequences is the same:

- Each non-leaf yields two nodes in the next level,
- Each win yields two cards in the next passthrough.

Expressing Win-Loss Sequences as Full Binary Trees

Structure of full binary trees and win-loss sequences is the same:

- Each non-leaf yields two nodes in the next level,
- Each win yields two cards in the next passthrough.
- Each leaf yields zero nodes in the next level,

Expressing Win-Loss Sequences as Full Binary Trees

Structure of full binary trees and win-loss sequences is the same:

- Each non-leaf yields two nodes in the next level,
- Each win yields two cards in the next passthrough.
- Each leaf yields zero nodes in the next level,
- Each loss yields zero cards in the next passthrough.

Expressing Win-Loss Sequences as Full Binary Trees

Structure of full binary trees and win-loss sequences is the same:

- Each non-leaf yields two nodes in the next level,
- Each win yields two cards in the next passthrough.
- Each leaf yields zero nodes in the next level,
- Each loss yields zero cards in the next passthrough.
- **non-leaves = W's, leaves = L's, levels of FBT = passthroughs**

From Win-Loss Sequence to Full Binary Tree

From Win-Loss Sequence to Full Binary Tree

Finding FBT for W/WW/LWLL/LL

From Win-Loss Sequence to Full Binary Tree

- initialize with a single unlabelled node

Finding FBT for W/WW/LWLL/LL

From Win-Loss Sequence to Full Binary Tree

- initialize with a single unlabelled node

Finding FBT for W/WW/LWLL/LL

From Win-Loss Sequence to Full Binary Tree

- initialize with a single unlabelled node
- write down the first passthrough in this unlabelled node

Finding FBT for W/WW/LWLL/LL

From Win-Loss Sequence to Full Binary Tree

- initialize with a single unlabelled node
- write down the first passthrough in this unlabelled node

Finding FBT for W/WW/LWLL/LL

From Win-Loss Sequence to Full Binary Tree

- initialize with a single unlabelled node
- write down the first passthrough in this unlabelled node
- for each node labelled W, create two unlabelled children nodes underneath the node

Finding FBT for W/WW/LWLL/LL

From Win-Loss Sequence to Full Binary Tree

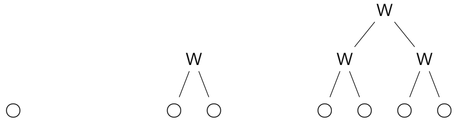
- initialize with a single unlabelled node
- write down the first passthrough in this unlabelled node
- for each node labelled W, create two unlabelled children nodes underneath the node
- write down each subsequent passthrough in the unlabelled nodes of the next level left-to-right

Finding FBT for W/WW/LWLL/LL

From Win-Loss Sequence to Full Binary Tree

- initialize with a single unlabelled node
- write down the first passthrough in this unlabelled node
- for each node labelled W, create two unlabelled children nodes underneath the node
- write down each subsequent passthrough in the unlabelled nodes of the next level left-to-right

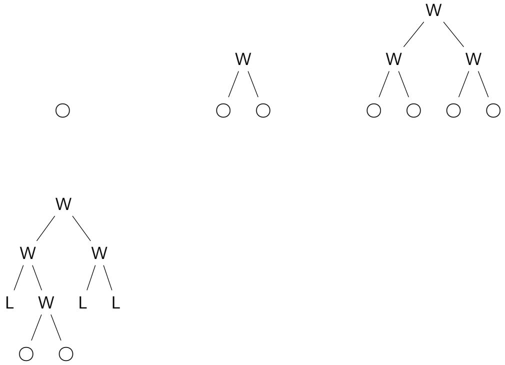
Finding FBT for W/WW/LWLL/LL



From Win-Loss Sequence to Full Binary Tree

- initialize with a single unlabelled node
- write down the first passthrough in this unlabelled node
- for each node labelled W, create two unlabelled children nodes underneath the node
- write down each subsequent passthrough in the unlabelled nodes of the next level left-to-right

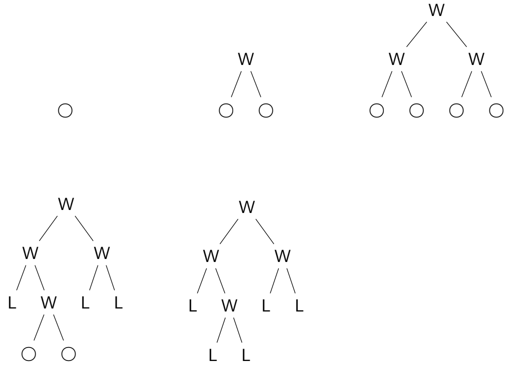
Finding FBT for W/WW/LWLL/LL



From Win-Loss Sequence to Full Binary Tree

- initialize with a single unlabelled node
- write down the first passthrough in this unlabelled node
- for each node labelled W, create two unlabelled children nodes underneath the node
- write down each subsequent passthrough in the unlabelled nodes of the next level left-to-right

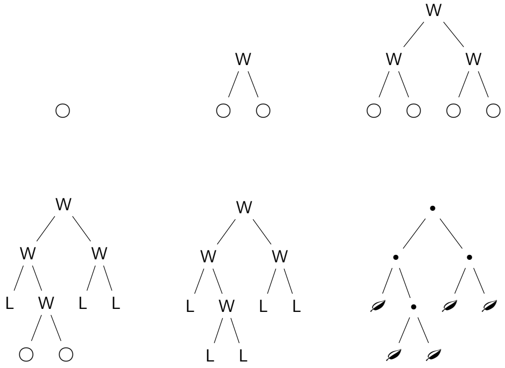
Finding FBT for W/WW/LWLL/LL



From Win-Loss Sequence to Full Binary Tree

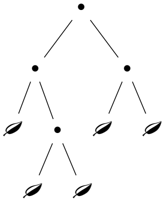
- initialize with a single unlabelled node
- write down the first passthrough in this unlabelled node
- for each node labelled *W*, create two unlabelled children nodes underneath the node
- write down each subsequent passthrough in the unlabelled nodes of the next level left-to-right

Finding FBT for W/WW/LWLL/LL



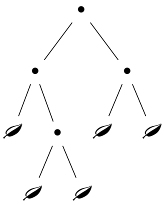
From Full Binary Tree Back To Win-Loss Sequence

From Full Binary Tree Back To Win-Loss Sequence



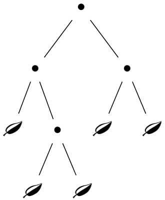
From Full Binary Tree Back To Win-Loss Sequence

- write W in all non-leaves and L in all leaves



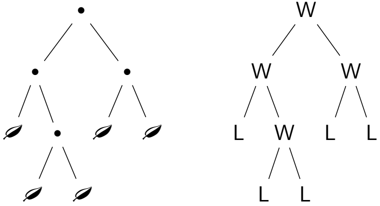
From Full Binary Tree Back To Win-Loss Sequence

- write W in all non-leaves and L in all leaves
- read left-to-right, top-to-bottom



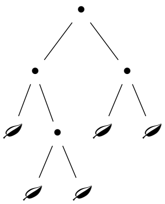
From Full Binary Tree Back To Win-Loss Sequence

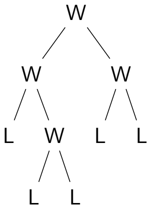
- write W in all non-leaves and L in all leaves
- read left-to-right, top-to-bottom



From Full Binary Tree Back To Win-Loss Sequence

- write W in all non-leaves and L in all leaves
- read left-to-right, top-to-bottom





Reading left-to-right, top-to-bottom, we get $W/WW/LWLL/LL$.

A Couple Consequences of the Bijection

Recall levels of a full binary tree are passthroughs of a win-loss sequence.

Proposition

The number of win-loss sequences A_k that end within k passthroughs for Alice satisfies $A_1 = 1$ and $A_{k+1} = A_k^2 + 1$.

Proposition

The number of win-loss sequences B_k that end in exactly $2k+1$ rounds is C_k, where C_k is the k'th Catalan number $\frac{1}{k+1}\binom{2k}{k}$.

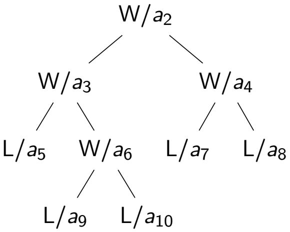
Poset for Necessarily Following a Win-Loss Sequence

Poset for Necessarily Following a Win-Loss Sequence

What relations between the cards in $a_1|a_2a_3\ldots a_{10}$ are there to necessarily follow $W/WW/LWLL/LL$?

Poset for Necessarily Following a Win-Loss Sequence

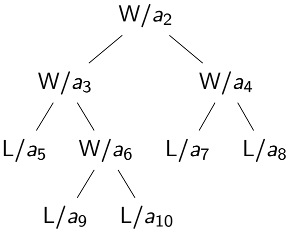
What relations between the cards in $a_1 | a_2 a_3 \dots a_{10}$ are there to necessarily follow $W/WW/LWLL/LL$?



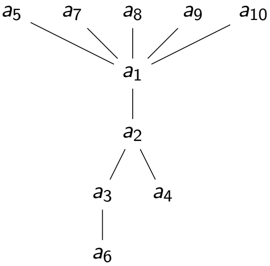
Tree Labelled With W's and L's

Poset for Necessarily Following a Win-Loss Sequence

What relations between the cards in $a_1|a_2a_3\ldots a_{10}$ are there to necessarily follow $W/WW/LWLL/LL$?



Tree Labelled With W's and L's



Poset for Random Putback

Continued Research

- Counting states that end in a certain number of rounds with WL-putback

Acknowledgements

- My mentor Dr. Tanya Khovanova
- Dr. Slava Gerovitch, Prof. Pavel Etingof, and the entire PRIMES program

Bibliography

Boris Alexeev and Jacob Tsimerman. *An analysis of a war-like card game*. 2010. arXiv: 1001.1017 [math.CO].

Eli Ben-Naim and Pavel Krapivsky. "Parity and ruin in a stochastic game". In: *The European Physical Journal B* 25 (Feb. 2002), pp. 239–243. DOI: 10.1140/epjb/e20020027.

Patrick Kellogg and Peter Smith. *A statistical paper on the card game "War"*. URL: http://www.patrickkellogg.com/school/papers/war/index.htm.

Evgeny Lakshtanov and Vera Roshchina. "On Finiteness in the Card Game of War". In: *The American Mathematical Monthly* 119.4 (2017), pp. 318–323. DOI: {http://dx.doi.org/10.4169/amer.math.monthly.119.04.318}.

Jacob Haqq Misra. "Predictability in the Game of War". In: *The Science Creative Quarterly* (2006). URL: https://www.scq.ubc.ca/predictability-in-the-game-of-war/.

Michael Z. Spivey. "Cycles in war.". eng. In: *Integers* 10.6 (2010),