Computing Science (CMPUT) 657

Algorithms for Combinatorial Games

Martin Müller

Department of Computing Science
University of Alberta
mmueller@ualberta.ca

Winter 2022

Introduction to Impartial Games

Impartial Games

- Impartial games are a special type of combinatorial games
- Restriction: both players always have the same options
- We will look at finite, loop-free impartial games only

Impartial Game - Nim

- Nim is an impartial game
- Both players have the same options
- Take any number of pebbles from a single heap
- Notation: $*n$ = single Nim heap with n pebbles
- By rules of Nim:
- $*n = \{*0, *1, *2, ..., *n-1 | *0, *1, ..., *n-1\}$
- The games $*n$ are called **nimbers** (a pun on numbers)
- Note: $*0 = 0$ (no moves in a heap of 0 pebbles)

Example: Impartial Clobber

- In regular Clobber, players have different moves
- B moves the Black pebbles, W the white ones
- Impartial Clobber: **both** can move with any pebble
- Same types of moves, must clobber the other color
- Example:
  `BWW = {.BW, W.W | .BW, W.W }`
- Compare with regular Clobber:
  `BWW = {.BW | W.W }`

Sprague-Grundy Theorem

- Sprague-Grundy theorem:
- Every finite impartial game is **equal** to some nimber $*n$
- Remember what equal means in CG
- Not too difficult to prove, but I'll skip it
- We'll just do examples
- The theorem is the basis for efficient algorithms to solve impartial games
- For each subgame: find which nimber $*n$ it is equal to
- Then, just add up the nimbers (efficient)

Examples - Impartial Clobber as Nimbers

- $\texttt{WWW} = \star 0$ (no move)
- $\texttt{BW} = \star 1$ (only moves to *0 for each player)
  - Detailed proof:
  - $\texttt{BW} = \{.\texttt{B},\ \texttt{W}.\ |\ .\texttt{B},\ \texttt{W}.\ \} = \{\star 0,\ \star 0\ |\ \star 0,\ \star 0\ \} = \{\star 0\ |\ \star 0\ \} = \star 1$
- $\texttt{BWW} = \{.\texttt{BW},\ \texttt{W}.\texttt{W}\ |\ .\texttt{BW},\ \texttt{W}.\texttt{W}\ \} = \{\star 1,\ \star 0\ |\ \star 1,\ \star 0\ \} = \star 2$

Example: A 2022 Course Project

- J. Dai and X. Chen, Impartial Clobber Solver, CMPUT 655 project report, University of Alberta 2022
- Studied Impartial Clobber computationally
- Solved all $1 \times n$ boards up to 1×16, and most boards up to 1×42
- Not solved: n = 17, 21, 39
- Can you do better?

Nim Addition

- How do nimbers add?
- What is $*n + *m$?
- By Sprague-Grundy, we know the result must be equal to some nimber
- But which one?

Nim Addition Simple Cases

- Two equal heaps add to 0:
- $*n + *n = *0 = 0$
- 0 = second player win, $\mathcal{P}$-position
- Strategy: mimic opponent's move in the other heap
- Can you prove $*n + *n = 0$ formally by induction? (Easy)

Nim Addition - Powers of Two Trick

- Write n in binary to get a decomposition of $*n$
- Example
- n = 39 = 32 + 4 + 2 + 1
- Then: $*39 = *32 + *4 + *2 + *1$
- Can you prove that this equality is always correct? (Not as easy but not too hard)

How to play Nim Perfectly

- Win in Nim: move to *0 = 0
- Break each heap into powers of 2
- Equal heaps cancel
- Move to 0
- Example
- $*3 + *4 + *6 = (*2 + *1) + *4 + (*4 + *2)$
- $= *4 + *4 + *2 + *2 + *1 = *1$
- Winning move: remove *1

How to play Nim Perfectly

- Does this always work?
- Yes, if the sum is different from zero
- What if there is no move directly?
- Insight: Can “convert” a big heap into any smaller one
- Example: sum game $*4 + *2 + *1$

How to play Nim Perfectly

- Example: sum game $*4 + *2 + *1$
- How to win?
- To move to 0, must pair up all three heaps
- Idea: break up the largest unpaired heap (*4 here) to match *all* the other unpaired ones
- Move from *4 to *2 + *1 = *3
- The *3 = *2 + *1 cancels the other two heaps *2 + *1

How to play Nim Perfectly - Relation to binary numbers

- Another equivalent way to view this method
- Write each heap size in binary
- Add all binary numbers bitwise modulo two (xor)
- The result is the nim sum, written in binary

Example - Nim Sum

- $*39 + *31 + *22$
- Convert into binary:
  - $*39$: 100111
  - $*31$: 11111
  - $*22$: 10110

Add each digit modulo 2 (1+1=0)

bit5	bit4	bit3	bit2	bit1	bit0
1	0	0	1	1	1
0	1	1	1	1	1
0	1	0	1	1	0
1	0	1	1	1	0

- 101110 (binary) = 46 (decimal)

- 101110 (binary) = 46 (decimal)
- So $*39 + *31 + *22 = *46$
- How to win? Move from *46 to 0
- How? Well, if we could add a *46 then *46 + *46 = 0
- We can write $*39 + *31 + *22 + *46$ in three different ways
- $(*39 + *46) + *31 + *22$
- $*39 + (*31 + *46) + *22$
- $*39 + *31 + (*22 + *46)$
- Which of those will be legal moves? Do these nim sums to check!

Theorems and Observations

- Any Nim sum of value *0 is a $\mathcal{P}$-position (second player win)
- Any other Nim sum (of value $*n$, $n > 0$ is a $\mathcal{N}$-position (first player win)
- By minimax:
- Any move from a $\mathcal{P}$-position leads to an $\mathcal{N}$-position
  - Any move by the loser leads to a winning position for the opponent
- From any $\mathcal{N}$-position, there is at least one move to a $\mathcal{P}$-position
  - At least one move by the winner leads to a losing position for the opponent

Solving Impartial Games by Computer

- We can solve large sums of large nimbers efficiently
- How to solve real impartial games such as Impartial Clobber?
- We need to find the corresponding nimbers for each subgame
- This can still be a large search problem for complex games
- We will talk about the (really interesting!) algorithms next
- After solving each subgame to a nimber, use the nim addition rule