Partial N-Player Hackenbush: Combinatorial Theory, Approximation, and Implementation

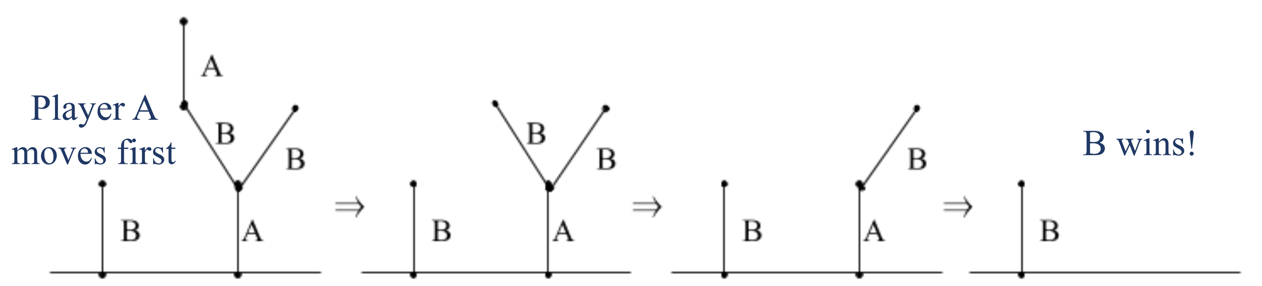
Zachary DeStefano[UG]

Department of Computing Sciences, Villanova University

Department of Computing Sciences, Villanova University, PA, USA

Abstract

This project generalizes game sums (the main tool used for analyzing positions in combinatorial games) in 2-Player Hackenbush and proposes a theory for analyzing N-Player combinatorial games including N-Player Hackenbush. This project proposes a new way to quantify the advantages to each of the players given any game position, provides a fast method for the approximation of this sum, and demonstrates an example implementation of this algorithm on arbitrary input. This algorithm is demonstrated by an application to N-Player Hackenbush.



Example 2 Player Hackenbush Gameplay

Background on 2-Player Game Sums

John H. Conway defines partial games as an abelian group with the following recursive definitions of an arbitrary game G:

The base case is the empty game G={∅|∅} (often written as) G={|}

While the general case is defined as follows:

$G = \{G^L \mid G^R\}$ where G^L and G^R are sets of games one move from G

$G + H = \{G^L + H, G + H^L \mid G^R + H, G + H^R\}$

$-G = \{-G^L \mid -G^R\}$

He defines a game sum as a so called surreal number which is calculated from the rules above and represents the outcome of both players playing a game optimally. In the case of finite games, these sums are calculated from the following formulas:

$0 \stackrel{\text{def}}{=} \{|\}$

$n \stackrel{\text{def}}{=} \{n-1|\}$ and by rules of negation $n \stackrel{\text{def}}{=} \{|1-n\}$

$\frac{m}{2^j} \stackrel{\text{def}}{=} \{\frac{m-1}{2^j} | \frac{m+1}{2^j}\}$

Pitfalls and Difficulties in Generalization to N Players

There are various difficulties in generalizing partial 2-Player game sums to arbitrary numbers of players which prevent a direct generalization of the definitions above.

Problem 1: Game sums with N>2 players cannot be represented by N-dimensional space.

Solution 1: Use constants "a," "b," etc. to represent a move for players "A," "B," etc.

Problem 2: It is possible for players to team to defeat another player, so that the calculation of the sum value will then depend more on the teaming of the players and not their respective advantages at any given time.

Solution 2: Prevent teaming by incentivizing players (as discussed in def. 6 below)

Problem 3: There is no clear definition of negation of a game sum with more than two players which prevents such Games with more than 2 players to form an Abelian group. Also, the sum of two separate games is not guaranteed to produce the same value as when both games are played as one game.

Solution 3: Represent games with N players as a monoid and use the sum of separate games as an approximation bounded from below by $O(n)$ and above by $O(N^n)$ when the game sum coefficients are not close and complete the full sum calculation with an efficiency of $O(N^n)$.

N-Player Game Sums Definitions

Here are my definitions which are extensions of the two-player ones provided by Conway and logically proceed from the definition of a game with N number of players:

1: A game with N players is defined by $G=\{G_1|G_2|\ldots|G_N\}$ where G_n is a subset of games that are one move for player n away from G.
For optimal play this can be expressed merely as the best move (defined later) for each player respectively as opposed to each option in each set

2: Games form a Monoid with addition and equivalence defined below:
Addition: $G + H \stackrel{\text{def}}{=} \{(G_1+H) \cup (G + H_1)|(G_2+H) \cup (G + H_2)| \ldots |(G_N+H) \cup (G + H_N)\}$
Although this does not imply: $S_{G+H} \stackrel{\text{def}}{=} \{S_{G_1+H}\ or\ S_{G+H_1}|S_{G_2+H}\ or\ S_{G+H_2}| \ldots |S_{G_N+H}\ or\ S_{G+H_N}\}$
Note: Addition is communitive and associative
Equivalence (from Conway Directly): $G = H\ if\ (\forall X)\ G + X\ has\ the\ same\ outcome\ as\ H + X$

3: One move advantage for a player m has a game sum of D_m

4: The game sum is 0 when no players have any move (G_1 through G_N are empty) $\{|| \ldots |\} \stackrel{\text{def}}{=} 0$
Corollary 1: No advantage for any player has a game sum of 0 as no players having a move is equal to all players having one move

$$0 = \sum_{n=1}^{N} D_n$$

Corollary 2: One move advantage for a player m is always equal to a one move disadvantage for every other player

$$D_m = -\left(\sum_{n=1}^{N} D_n\right) + D_m \stackrel{\text{def}}{=} -\left(\sum_{n=1}^{N} D_n\,[n \neq m]\right)$$

5: The game sum S of a game G is the sum of the game sums S_{G_1} through S_{G_N} of G_1 through G_N plus one move advantage for each player that has one move available, all divided by the number of players that have a move:

$$S_G = \frac{\sum_{n=1}^{N}\left(D_n + S_{G_n}\right)[G_n \neq \emptyset]}{|G|}$$

6: The best move for each player is defined as the move which gives the highest advantage to them directly and indirectly. The advantage for player m is calculated (using the simplest form expression) by the maximum value for the coefficient of D_m with ties resolved by the smallest next coefficient ex. D_{m+1}, D_{m+2}… looping around in player order

7: The simplest form expression is defined by taking the sum and subtracting the value:

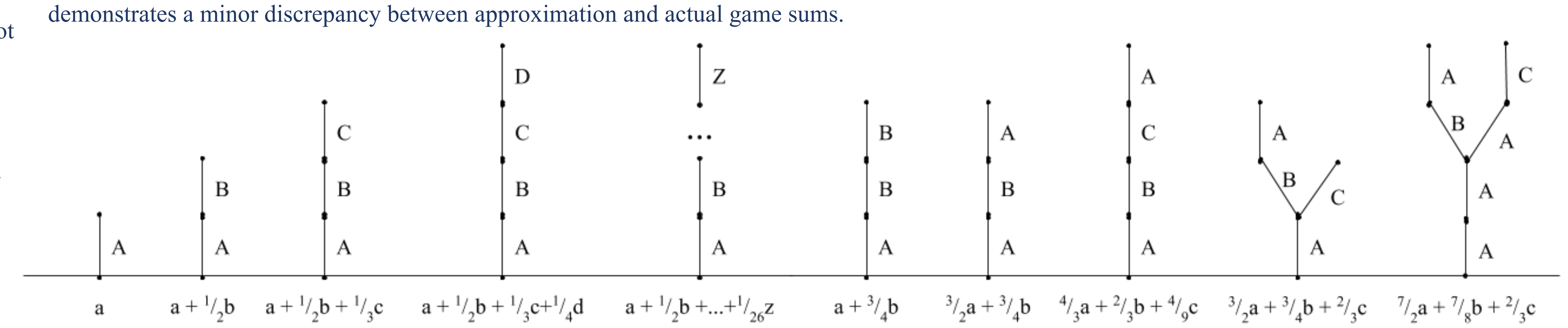
$$M \sum_{n=1}^{N} D_n$$

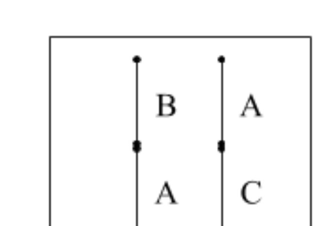
Where M is the smallest coefficient in the sum. (This does not change the overall value of the expression due to definition 4)

$$\sum_{n=1}^{N} D_n = 0$$

Examples

The following are examples of the application of this algorithm to various Hackenbush games with N-Players as well as a short discussion on the "AB-CA Problem" which demonstrates a minor discrepancy between approximation and actual game sums.





Simplified Sums:
Actual: $\frac{5}{6}a + \frac{1}{2}c$
Approx: $a + \frac{1}{2}c$

The "AB-CA Problem" is the smallest game where the approximation differs from the actual sum. This difference is a mere a/6 and does not affect the final outcome class.
Sometimes error is predictable: e.g. games following the pattern AB + AC + AD + … have error coefficients starting with $\frac{1}{6}$ for c and continuing to (n-1)/n – 1/2.

Algorithm and Approximation

Here is an example output obtained by my program for a complicated game with an accompanying visualization of the game:

Input Tree: ABAC<<DBA<B@AB<C<D@ (as represented in the computer)

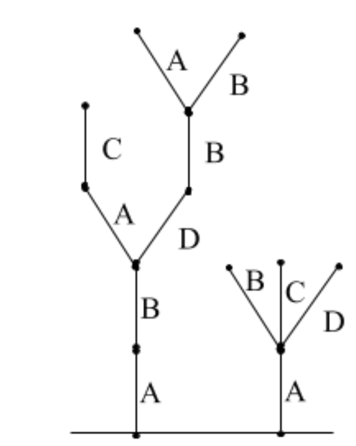
Number of Players: 4

Real Sum: 1327529/663552 a + 3286079/3981312 b + 3281161/11943936 d

Estimate Sum: 217/108 a + 767/864 b + 691/1944 d

Estimate Error: -5719/663552 a + -248257/3981312 b + -964343/11943936 d

Real Sum Running Time: 1747ms **Estimate Sum Running Time**: <1ms



Conclusion

Using my definitions, I implemented the algorithm in Java and verified the results by hand. In the majority of cases, the winner obtained by the approximation was the same as the winner obtained from the complete calculation. Additionally, this theory produced the same results for 2-Player games as Conway's theory specifically for 2-Player games.

This method allows for the generalization of 2-Player combinatorial game theory for cold non-loopy partial games to the case of N players. This method of calculating game sums is an effective and practical generalization of Conway's original theory.

Future Research

This project provides a foundation for N-player game analysis with game sums which is efficient and compatible with existing 2-Player combinatorial game theory. Further research may be conducted in the following areas:

1: To design more efficient algorithms for full calculation and more accurate algorithms for approximation.

2: To extend this theory to other classes of games (impartial, loopy, and hot).

3: To determine a method of reversing the process and generating an N-player Hackenbush game from a game sum.

References

[1] Albert, M. H., Nowakowski, R. J., & Wolfe, D. (2009). *Lessons In Play: An Introduction To Combinatorial Game Theory*. Wellesley, MA: A K Peters.

[2] Cincotti, A., Komori, S., & Iida, H. (2008). The Game of Synchronized Triomineering and Synchronized Tridomineering. *World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering,2*(7).

[3] Conway, J. H., & Guy, R. K. (1996). *The Book Of Numbers*. New York, NY: Springer-Verlag.

[4] Conway, J. H. (2001). *On Numbers And Games*. Natick, MA: A.K. Peters.

[5] Greene, K. A., & Mason, S. K., Dr. (2017). *Exploration of the three-player partizan game of Rhombination*(Unpublished master's thesis). Wake Forest University.

[6] Krawec, W. O. (2015). N-Player impartial combinatorial games with random players. *Theoretical Computer Science,569*, 1-12. doi:10.1016/j.tcs.2014.12.003

[7] Propp, J. (2000). Three-player impartial games. *Theoretical Computer Science,233*(1-2), 263-278. doi:10.1016/s0304-3975(99)00128-0

Acknowledgements

These research began in an Honors Computer Science independent Study supervised by Anany Levitin. Thanks to Anany Levitin for his valuable help and guidance in this project.