Generation and Analysis of Optimal Strategies for Hearthstone Match Formats

Dominic Calkosz
James McCann
dcalkosz@andrew.cmu.edu
jmccann@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania

ABSTRACT

Hearthstone [3] is a 1-vs-1 digital card game that is played competitively in a variety of formats. In a tournament, each player preconstructs several decks and then chooses at each point in the match what deck they will play, according to the restrictions of a chosen match format. Such tournaments measure players' skill and are exciting for viewers, but can take place in a variety of match formats which fans claim drastically affect the competitiveness and viewer engagement [9]. Given that different deck archetypes have certain expected winrates against other archetypes, an optimal strategy exists for each player that determines with what probability they should pick each action available to them. We developed software that can solve for optimal strategies under fixed conditions and apply this process to winrate sets to generate metrics that can be compared across formats. We then used a large dataset collected from live Hearthstone players to generate metrics that reflect the competitive deck ecosystem, and used these results to draw qualitative conclusions about various match formats.

CCS CONCEPTS

• **Mathematics of computing → Solvers**; • **Applied computing → Computer games**; Economics.

KEYWORDS

Game Theory, Card Games, Hearthstone

1 INTRODUCTION

In professional Hearthstone tournaments, choosing which deck to play in the next game is a key strategic component. Since this decision is independent of gameplay decisions once the game has started, we can view each game as a probabilistic win/lose outcome based on the winrate between the two decks being played, and the relative skill of each player with their respective deck. After abstracting each game to a randomized outcome, the match can be modeled as a zero-sum game with a sequence of simultaneous decisions made by both players corresponding to protecting, banning, or choosing to play certain decks. This is a well studied structure in game theory known as a finite game tree [5], which indicates that an optimal strategy exists for each player. Thus, with input data on the winrates between each deck, and since a finite number of possible matches could play out, the optimal strategy for each player at each step of the match can be computed. But even though such winrate data is widely available through deck tracking services such as HSReplay [6], no match solver previously existed to generate the match tree and compute optimal strategies.

Another gap in game theory for Hearthstone is a comparative analysis of different match formats. The match format chosen determines how many decks each player brings, how many decks they can protect and ban, how many games they need to win, whether a deck is eliminated when you win or lose with it, and whether the winner or loser of the previous game must choose the same deck for the next game. Depending on each of these factors, matches will tend to last a different number of games, reflect player skill in different ways, show more or less variety in deck matchups, and lead to different levels of viewer excitement. When comparing certain formats, one can intuit how certain qualities will differ, but this is the first formal research to verify or refute those notions. Other formats and qualities seem more difficult to reason about *a priori*, but could be determined through empirical observation of professional tournaments, or by simulating and analyzing a large number of matches, which we have done here.

1.1 Our Approach

We have applied computational game theory to these two problems by developing HearthNash, a match tree generation and analysis tool. It takes input parameters for match format settings and deck matchup winrates, and uses them to generate a directed acyclic graph structure containing every possible sequence of decisions and outcomes for a given match. As the structure is created, our system computes the expected match victory probability and optimal mixed strategy for each player at every node in the tree. This allows the system to solve any match and describe optimal strategies given sufficient knowledge of the initial state.

We are able to address the second problem, comparing different match formats, by analyzing properties of generated match trees. Importantly, we were able to run these analyses using matchup winrate data from a real meta state in Hearthstone thanks to HSReplay [6], a free deck tracker for players which doubles as a data aggregate and analysis service by collecting the results of games played by their users. HSReplay was able to provide us with matchup winrate data collected by tracking millions of games by high ranking Hearthstone players.

1.2 Contributions

We analyzed several metrics of each match format which allowed us to identify certain differentiating qualities. By measuring the likelihood of a match spanning a certain number of games, we concluded that the match length distribution is quite consistent within each class of formats (based on number of games required to win). By measuring the sensitivity of a player's victory probability to their skill across all decks, we observed that formats which require *more* games to win are more reflective of "wide" player skill. Conversely, we observed that a player's victory probability is more sensitive to adjustments in "tall" skill adjustments (skill with a single deck) for formats which require *less* games to win. Our analysis on tall skill sensitivity also indicated that Last Hero Standing formats are each more sensitive to improvements with a single deck than their corresponding Conquest counterparts. Finally, we observed that as the magnitude of the skill adjustment for a single deck increases, only those formats which allow the most powerful deck to dominate show an increase in sensitivity.

Our software, HearthNash, also stands on its own as a computational game theory tool to solve and analyze individual matches in Hearthstone or other games with a similar match structure. In order to make it more accessible, we developed an interactive web interface (https://dominic-calkosz.com/HearthNash/web-interface.html) which allows anyone to test potential match scenarios and outcomes, and thus to better understand the game theoretical properties of Hearthstone matches and their assorted formats.

2 RELATED WORK

Within the area of optimal strategies for Hearthstone matches, we found 2 prior studies, both of which are public posts on Reddit.com from the subreddit r/CompetitiveHS (standing for "Competitive Hearthstone"). In April of 2015, user *jmc999* detailed how he calculated the optimal strategies for a no-bans Conquest best-of-5 format by brute-forcing a large number of possible strategies at each point in the match and then picking the best one [8]. It illustrated how deck matchup winrates lead directly to optimal strategies and victory probabilities, but was limited to a single format with no bans or protects. User *BryPye* was inspired by that post to run his own Monte Carlo simulation on the same format and deck matchup winrates, which succeeded in replicating *jmc999*'s results [7]. They noted an additional limitation of their work, that "it assumes your opponent is choosing their deck at random."

More general match and tournament strategy research has been conducted and formally published. "Backward induction and common knowledge of rationality" [1] showed that backwards induction is valid, meaning it can be used to find the optimal strategy for each player, assuming rational players. Although this paper focuses on perfect information games (e.g. chess), it also proves this result for probabilistic games. This is critical to our research because our method of generating and solving match trees for optimal strategies relies heavily on backwards induction.

"Exploring the Hearthstone Deck Space" [2] examines computational deck building using an evolution strategy. This paper overlaps with several of the concepts that are core to our research here, including deck archetype matchups and fine adjustments in winrate probabilities. The implications of this work related to balancing Hearthstone algorithmically could potentially integrate match strategies and victory probabilities as they are computed in our research.

3 IMPLEMENTATION

HearthNash is implemented in JavaScript, and a mirror of its source code can be found at https://github.com/Dmcdominic/HearthNash-Mirror.

3.1 Hearthstone Match Structure

The sequence of decisions and games that are played out in a Hearthstone match are determined by the format settings, which are the rules that the match will adhere to. The flow of a match is visualized in Figure 1. Some matches begin with a "shield phase", in which each player is allowed to protect some number of their own decks from being banned (in the following ban phase). Players choose simultaneously which deck(s) to protect, meaning that one player can not use information about the other's decision to improve their own. Many formats do not include a shield phase and instead go straight to the ban phase. In the ban phase, each player is allowed to ban some number of their opponent's decks from being played in any of the subsequent games. Players make this choice simultaneously, and then proceed to the first deck choice phase. Some formats do not include a ban phase at all, and instead begin with the first deck choice phase. Before each game, players are allowed to choose which of their remaining decks they want to play with. Players once again make this choice simultaneously. This constitutes a deck choice phase. However, certain additional rules may apply to this phase. If this is not the first deck choice phase, then the format settings may specify that the winner and/or loser of the previous game is required to continue playing with the same deck. Once each player has chosen their deck, the game is played out, and the first player to destroy the enemy hero earns 1 win. Following this game, the format settings may specify that the winning and/or losing deck be removed from the decks available to the corresponding player. Then players return to the deck choice phase. The cycle of deck choice and game phases repeats until one player has earned a certain number of wins, determined by the format settings, to be crowned the victor of the match.

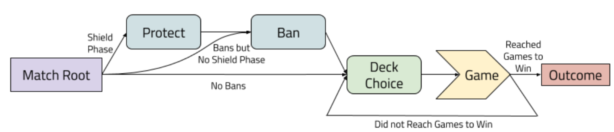

Figure 1: The flow of a Hearthstone match.

As an example, the format "Conquest Best-of-3" is defined such that each player brings 3 decks, skips the shield phase, may ban 1 of their opponent's decks, and must win 2 games to win the match. In addition, it specifies that a player's deck may not be used again once they have won a game with it, and that players are free to choose from any of their remaining decks before each game. All formats used in this research are defined in Table 1.

3.2 How HearthNash Models a Match

To model a single match, HearthNash takes in three parameters. The first input is the format settings, as described in the previous section. Second, the meta settings specify the winrate probabilities

Table 1: Hearthstone Match Format Definitions

Format	Protects	Bans	Games to Win	Decks/Player	Deck Removed	Can Switch Decks
1 Game No Bans	0	0	1	1	N/A	N/A
1 Game One Ban	0	1	1	2	N/A	N/A
Conquest BO3	0	1	2	3	Winner	Both
Conquest BO5	0	1	3	4	Winner	Both
Shield Phase Conquest BO3	1	1	2	3	Winner	Both
Shield Phase Conquest BO5	1	1	3	4	Winner	Both
Conquest No Bans BO3	0	0	2	2	Winner	Both
Conquest No Bans BO5	0	0	3	3	Winner	Both
Last Hero Standing BO3	0	1	2	3	Loser	Loser
Last Hero Standing BO5	0	1	3	4	Loser	Loser
Shield Phase Last Hero Standing BO3	1	1	2	3	Loser	Loser
Shield Phase Last Hero Standing BO5	1	1	3	4	Loser	Loser
Last Hero Standing No Bans BO3	0	0	2	2	Loser	Loser
Last Hero Standing No Bans BO5	0	0	3	3	Loser	Loser

for all possible deck matchups which could come up based on the decks that each player brings to the match. These are needed in order to compute the probability of each possible outcome and to determine optimal strategies at each decision point. Third, each player has a predetermined set of decks that they start the match with. The details of the decks (what cards they contain) are not necessary, as games are simply represented by a probabilistic outcome. Decks in each list are simply represented by indices into the winrate data from the meta settings (although they may also contain supplemental data for usability such as the name of the archetype).

The match tree structure generated from this input adheres closely to the phases described in the previous section. The tree is made up of nodes, which each include an indexed array of child nodes corresponding to all possible events that could come next in the match. A diagram of an example match tree structure for the format 1 Game 1 Ban is shown in Figure 2. The top of the tree is a single “match root” node, with a single child node depending on how the match starts. If there is a shield phase, then it starts with a protect node. If there is no shield phase but there are bans, it starts with a ban node. Otherwise, it starts with a deck choice node. The protect node creates a child for each possible combination of deck protects that the players could make. For instance, if each player has 2 decks and they are allowed 1 protect each, then there will be 4 child nodes, each leading to a different version of the following ban phase where different decks have been protected. A ban phase node, similarly, will generate a child (deck choice) node for each possible combination of deck bans that the players could make. A deck choice node will generate a child (game) node for each possible combination of decks that the players could choose to play against each other. A game node will generate two children nodes, one for each possible winner of the game. Sometimes a child of a game node will be an outcome node, which dictates that a certain player is the final victor of the match, and has no children. Otherwise, a child of a game node is a deck choice node to determine the decks used in the following game.

Once a node has generated all of its children, it completes its own generation by computing the optimal strategy for each player (if this is a decision point, i.e. a protect, ban, or deck choice node), and the probability of each player winning from this point in the match (assuming optimal play). The optimal strategy at a decision point can be computed using a payoff matrix, where the payoff for each outcome is equivalent to the victory probability calculated at the corresponding child node. We use the pivot method, which we implemented according to the description in the textbook Game Theory [5] [4], to extract the optimal strategy for each player based on the payoff matrix. This algorithm also conveniently outputs the expected value when players play optimally, which defines the victory probability of each player. The victory probabilities for the match root are trivial as it only has one child. The victory probabilities for an outcome node (the base case) are simply 1 and 0 for the victorious and losing playing respectively. The victory probabilities for a game node are also determined using the victory probabilities of its children, but in this case, the weight of each child is determined by the probability of either player winning the match, which simply requires indexing into the winrate matrix for this particular matchup.

3.3 Data Analysis Pipeline

Our data analysis pipeline consists of three main stages. The first stage is to convert the raw HSReplay data into usable meta info. The data was provided to us in two sets, one based on games played between ranks 20-11, and the other based on games played between ranks 10-Legend (the latter being the higher-ranked pool of players). This is according to the original rank system, as the data was collected between February 27, 2020 and March 5, 2020. The data included empirical winrates for every archetype matchup that was recorded, as well as the sample size on that particular matchup. A large portion of this data includes archetype matchups with very small (less than 100) samples, which does not give us the desired winrate accuracy. Part of this first stage is reducing it down to 10 archetypes, such that the minimum number of samples between

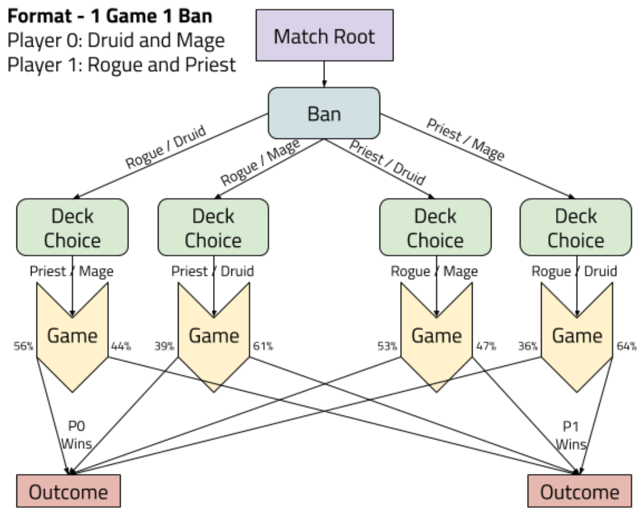

Figure 2: Example diagram of a Hearthstone match tree, for the 1 Game 1 Ban format.

any two archetypes is maximized. The rest of this stage is concerned with formatting and saving the meta info to file.

The second stage generates a fixed number of match trees for each format using the meta info from the first stage. After each tree is generated, it is also written to file so that it can later be analyzed for several different metrics. Every tree corresponds to a pair of randomly chosen subsets of the archetypes available, which are then used as each player's starting decks for the match. Importantly, the archetype set pairs chosen for the match trees of a given format are exactly the same pairs used for the other formats which require the same number of decks for each player. For example, Conquest best-of-3 and Last Hero Standing best-of-3 both require players to bring 3 decks each, so every match tree generated for one of these formats will correspond to a match tree in the other format with precisely the same deck inputs. This gives us more confidence when making comparisons between formats in the analysis stage. In addition, we use a pseudorandom number generator so that all tests can be replicated.

Finally, the third stage conducts analyses of each match tree that has been generated, and outputs the results into .csv files alongside corresponding .json metadata files. The details of each metric and how it is analyzed are described below.

3.4 Details of Metric Analyses

The first metric is distribution of match length, which is the number of games that are played before one player wins. This is computed for each tree by taking a weighted average of the game-depth of all leaves (outcome nodes). The weight for each node is determined by assuming that players play optimally and then evaluating the probability that the node is reached. This can be recursively computed according to the probability that the parent node is reached, multiplied by how likely the node itself will be chosen based on the players' optimal strategies (if it is a decision point), or based on the winrate probability if it is a game node.

The second and third metrics both aim to measure how responsive each format is to adjustments in player skill. In our match model, player skill is reflected by their matchup winrates. So to test skill sensitivity, we make small adjustments to the winrates of a given match tree, generate the corresponding adjusted tree, and then compare the victory probabilities of the new tree with the original.

The difference between these two metrics pertains to which winrates are adjusted. The second metric, called *wide* skill sensitivity, tests how responsive a format is to increasing the winrates of *all* decks for one player. For each match tree, the system makes a duplicate of the winrate matrix, boosts the winrates for player 0, generates a new match tree with the same initial decks for each player, and records the amount by which player 0's win probability increased (if at all). It then repeats this for player 1 as well, and for progressively larger increments in skill. The third metric, called *tall* skill sensitivity, tests how responsive a format is to increasing the winrates of a *single* deck for one player. The process is identical to wide skill sensitivity, except that a new tree is generated for each individual deck, such that only the winrates for that deck of that player are boosted.

4 RESULTS AND DISCUSSION

4.1 Match Length Distribution

From our first metric, we observed that match length distribution is quite consistent within each class of formats, based on the number of games required to win (best-of-1, best-of-3, and best-of-5). This is visualized in Figure 3. Best-of-1 is the obvious case, where the formats strictly allow a single game to determine the victor. Our results for best-of-3 indicate that it is *slightly* more common for a match to go to a third game (meaning each player won one of the first two games), for all six BO3 formats that we analyzed. Best-of-5 is similarly consistent, showing less than 3% difference between any two formats for a certain match length. From these results, it is clear that the probability of a sweep (one player winning the first 3 games) is significantly lower than going to 4 or 5 games total. This may be unsurprising, though, considering that out of the 8 possible outcomes on the first 3 games, only 2 of them lead to a sweep. This suggests $2/8 = 25\%$ as an intuitive estimate for the number of sweeps (assuming a relatively even player skill distribution), which lines up well with our results.

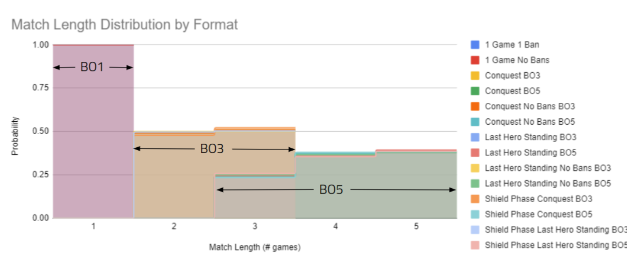

Figure 3: Results from analysis of match length distribution.

4.2 Wide Skill Sensitivity

Our second metric, wide skill sensitivity, yielded some interesting results regarding how responsive each format is to improvements in a player's skill across all of their decks. Figure 4 shows that, as with match length distribution, each class of formats (based on number of games to win) is quite consistent. Between these classes, though, it is interesting to observe that the more games required to win, the more sensitive a format is to skill adjustments. This seems fairly intuitive, because when a single game is played, changes in victory probability should be 1-to-1 with changes in the winrate probability of that single game, and the more skilled player still has a reasonable chance of losing. But as the number of games required to win approaches infinity, the likelihood of the more skilled player being victorious will approach 100%.

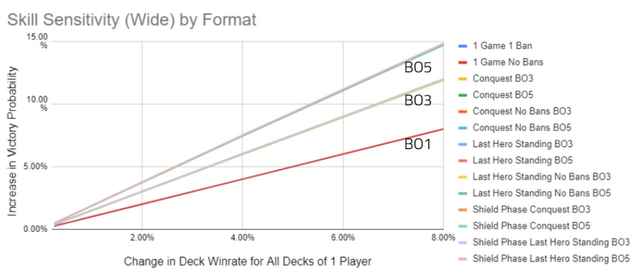

Figure 4: Results from analysis of wide skill sensitivity.

4.3 Tall Skill Sensitivity

Our third metric, tall skill sensitivity, yielded a much greater variety of results across the different formats, seen in Figure 5. There are several interesting observations to be made here. For one, best-of-3 formats (lighter colors) are each more sensitive to tall skill improvements than their best-of-5 counterparts (darker colors). This at first seemed surprising to us when compared with the wide skill sensitivity, which shows the opposite relationship. However, it can be explained by the fact that formats with fewer games lead to greater impact of each individual game, which means greater impact of a player's skill with each individual deck.

Another observation is that Last Hero Standing formats (dotted) are all more sensitive to tall skill improvements than their Conquest counterparts (dashed). The core difference between these two formats is that after each game in Last Hero Standing, the *loser's* deck is removed from their options, as opposed to Conquest, in which the *winner's* deck is removed. This is relevant to tall skill sensitivity because it means that a single powerful deck is able to win several games for you in Last Hero Standing.

Figure 6 shows the same skill sensitivity (tall) results as in Figure 5, but instead plotted by the increase in victory probability *over the change in deck winrate*. This way, the skill sensitivity is normalized to show what proportion of the winrate improvement translates to victory probability improvement. This illuminates another interesting observation: as the magnitude of the skill adjustment for a single deck increases, most of the formats have a *decrease* in tall skill sensitivity ratio, but there are 4 formats that

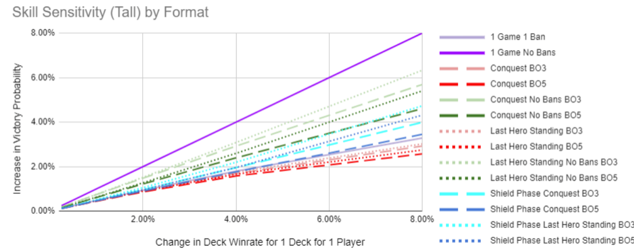

Figure 5: Results from analysis of tall skill sensitivity. Conquest formats are plotted with long dashes. Last Hero Standing formats are plotted with dots. Shield phase formats are in blue, no-ban formats are in green, and standard (single-ban) formats are in red. BO5 formats use darker colors, while BO3 formats use lighter colors. 1 game formats are plotted in purple with solid lines.

actually show an *increase* in this sensitivity. These particular formats are Last Hero Standing No Bans (both BO3 and BO5), and Shield Phase Last Hero Standing (both BO3 and BO5). What makes these 4 formats unique is that a single powerful deck can dominate the match, *and* your opponent is unable to ban the most powerful deck. As the change in deck winrate increases, it also increases the expected number of games which you will play with that deck, leading to a multiplicative effect. The only Last Hero Standing formats which do *not* increase like this are the default formats, with a single ban and no deck protection. Since each player is able to ban the other player's most powerful deck, an exceedingly large increase in winrate for a particular deck is increasingly more likely to be banned and nullified by the opponent.

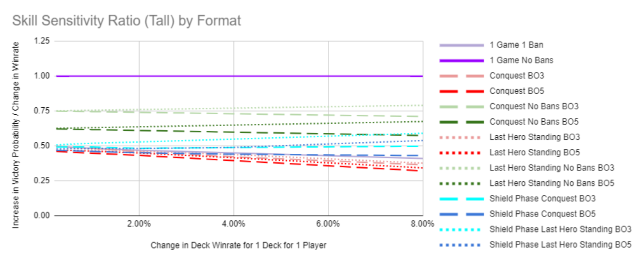

Figure 6: Results from analysis of tall skill sensitivity, plotted by increase in victory probability over the change in deck winrate.

5 EVALUATION

5.1 Sample Size and Variance

In order to ensure that enough samples were taken for our metrics, we ran the same analyses over several different samples sizes. Figure 7 shows how the results of a particular analysis converge from trials with a single sample to trials with 50 samples. We ultimately decided on a sample size of 30 to use for our final analysis, which is covered in the Results and Discussion section, because it converged to a point of less than 5% deviation from the mean, but keeps computation time to a manageable level.

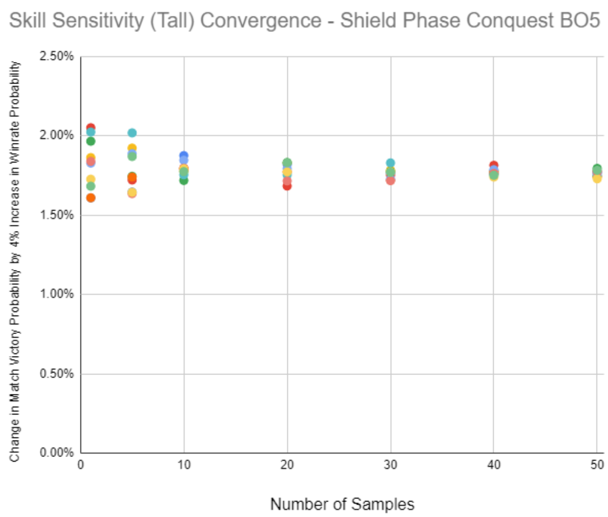

Figure 7: Results from several trials of skill sensitivity (tall) across different sample sizes. Specifically plots the change in match victory probability after a 4% increase in winrate probability for Shield Phase Conquest BO5. Data used for Results and Discussion is based on 30 samples.

5.2 Branching Factor and Memoization

At each phase in the match, there are several possible events that could occur (based on the players' choices, or the winner of a game). This means that each node can have 2 or more children, and each of its children can have 2 or more children, etc. This leads to exponential blowup of the size of the tree structure, especially for the larger formats such as shield phase best-of-5 matches which have a node-depth of up to 14. However, not every node is unique. For most match states, there are several different sequences of events which can lead to that particular state. For example, there may be hundreds of possible ways for the two players to end up in a final game which comes down to two particular decks, and so it is unnecessary to have a separate node object to represent every instance of that state.

We apply memoization to the match tree generation by wrapping each node constructor with a function that first checks a hashtable of all the previously generated nodes of that type. On a hit, we simply return the found node. On a miss, we generate the new node as normal, then place it in the hashmap before returning it to the caller. We also apply memoization to each of the metrics in the analysis pipeline. For example, the match length distribution of a particular node need not be computed more than once. As HearthNash traverses the tree, it adds a property to each node indicating that it has already been computed, along with the value itself. Then before each new node is analyzed, the system first checks if the property exists, and skips the computation if so.

6 FUTURE WORK

6.1 Other Possible Uses and Extensions of HearthNash

There are several potential uses for HearthNash beyond what we have explored here. The most straightforward extension would be to implement additional metrics for analyzing match trees and their respective formats. Optimal deck practice strategies is a natural followup to our analysis of both wide and tall skill sensitivity. A player might wonder, given a certain set of decks that they intend to bring to a match, which deck they should prioritize practicing and improving their winrates with. Another metric that could be analyzed is matchup disparity distribution: Are most games expected to be even matchups (near 50% winrate) or blowouts (one deck is strongly favored over the other). This factor might be important in terms of preventing luck-based outcomes (in which the match has a huge swing in victory probabilities based on a single unlucky deck ban or pick). It also has significance for viewers who find even matchups more exciting than games which may seem decided from the outset.

One might also consider the phase that comes before a match even begins, in which players decide which decks they will bring to the match. This is a simultaneous decision, meaning that each player makes their selection without any knowledge of their opponent's decision. Solving for this deck selection phase would enable users of HearthNash to not only play by an optimal strategy throughout the match itself, but also to select optimal decks from the beginning.

Extending HearthNash to solve the deck selection phase would also enable us to analyze a new metric for archetype diversity. One might wonder if, in a given meta state, certain formats lead to only 3 or 4 different archetypes being used predominantly (over many matches), whereas other formats lead to a variety of 6 or more archetypes appearing regularly. A similar metric could be defined according to matchup diversity, which is interested in how many different archetype matchups are likely to get played out over many matches. This is an important factor in viewer excitement, because fans watching weeks of Hearthstone tournaments are more likely to stay engaged by an assortment of archetypes, rather than by watching the same matchups play out repeatedly.

6.2 Empirical Research

A natural question that arises from the theoretical results determined by our research here is whether or not professional players tend to play optimally. This could be studied through empirical observation of professional tournaments. If players tend to play sub-optimally in certain formats, it would indicate significant room for strategic improvement at even the highest level of play.

One might also wonder how engaging each format is to viewers on game streaming platforms such as Twitch and YouTube. By comparing total viewing hours or viewer retention for tournaments with different match formats, we might observe which ones are more popular for entertainment. This would also allow for a multi-dimensional analysis that could illuminate which characteristics of match formats (such as matchup disparity distribution or archetype/matchup diversity) actually lead to better viewer engagement.

6.3 Beyond Hearthstone

Finally, the match structure that HearthNash uses may be applicable to more than just Hearthstone. Games such as *Magic: The Gathering* use similar match formats, especially if the deck selection phase is integrated. This could illuminate interesting results about Magic as it has about Hearthstone. It would also allow for mathematical and objective comparisons between these two games which have not been possible before. Even games outside of the card game genre could utilize the same structure for determining optimal strategies, including fantasy sports drafting.

HearthNash may also be applicable to drafting strategies in multiplayer online battle arenas (MOBAs) and other games with similar pre-gameplay draft phases. For example, the Captains Mode in *DotA 2* follows a predetermined sequence of picks and bans for each team as they assemble 5 heroes each, from a pool of over 100 possible heroes. By modifying HearthNash to account for allied hero synergies/anti-synergies and opposing hero matchups, optimal strategies for the draft process could be solved. The predetermined sequence for this mode, however, has undergone several adjustments over the years. The modified version of HearthNash would also enable a game theoretical analysis of each variation, indicating key differences and ultimately recommending a certain optimized sequence.

7 CONCLUSION

Hearthstone match formats and their relative strengths and weaknesses have been the subject of much discussion among fans and professional players alike. Until now, this discussion has been limited to anecdotal observation and small, isolated simulations. With HearthNash, we have been able to make direct comparisons between match formats using computational game theory and real world data on winrates at high ranks. We determined that match length distribution and wide skill sensitivity are both consistent within each class of formats. We also determined that tall skill sensitivity is stronger in each best-of-3 format than its best-of-5 counterpart, and stronger in each Last Hero Standing format than its Conquest counterpart. Finally, we observed that as the magnitude of the skill adjustment for a single deck increases, only those formats which allow the most powerful deck to dominate show an increase in sensitivity. HearthNash opens the door to further research, including optimal strategies and implications of the pre-match deck selection phase, as well as extension beyond Hearthstone to games such as Magic: The Gathering, fantasy sports drafting, and MOBAs. We hope that this research strengthens the broader understanding of match formats and their optimal strategies, and enables other research on game theory within digital games.

ACKNOWLEDGMENTS

Thank you to HSReplay, especially Dereck Toker, for providing us with the matchup winrate data that made it possible to analyze formats based on a real Hearthstone meta state.

Thank you to Jonathan Aldrich and Bogdan Vasilescu, the professors for *15-400: Research Practicum in Computer Science, Spring 2020* at Carnegie Mellon University, for providing high-level support throughout the research process.

REFERENCES

[1] Robert J. Aumann. 1995. *Backward Induction and Common Knowledge of Rationality.* Institute of Mathematics, The Hebrew University.

[2] Aditya Bhatt, Scott Lee, Fernando de Mesentier Silva, Connor W. Watson, Julian Togelius, and Amy K. Hoover. 2018. *Exploring the Hearthstone Deck Space.*

[3] Blizzard Entertainment. 2014. Hearthstone. video game.

[4] Thomas S. Ferguson. [n.d.]. Calculating the Solution of a Matrix Game. Retrieved December 10, 2019 from https://www.math.ucla.edu/~tom/gamesolve.html

[5] Thomas S. Ferguson. [n.d.]. *Game Theory.* Retrieved December 10, 2019 from https://www.math.ucla.edu/~tom/Game_Theory/mat.pdf

[6] LLC HearthSim. 2020. HSReplay.net. Retrieved March 5, 2012 from https://hsreplay.net/meta/#tab=matchups

[7] u/BryPye. 2015. r/hearthstone - A small Monte Carlo simulation of deck selection in conquest format. Retrieved December 2, 2019 from https://www.reddit.com/r/CompetitiveHS/comments/33btr8/a_small_monte_carlo_simulation_of_deck_selection/

[8] u/jmc999. 2015. r/hearthstone - A game theory approach to the Conquest tournament format. Retrieved December 2, 2019 from https://www.reddit.com/r/CompetitiveHS/comments/32mw19/a_game_theory_approach_to_the_conquest_tournament/

[9] u/Joppiee. 2016. r/hearthstone - Seatstory Cup shows why the last hero standing format is superior to conquest. Retrieved May 8, 2020 from https://www.reddit.com/r/hearthstone/comments/4ug5r5/seatstory_cup_shows_why_the_last_hero_standing/