

A Heuristic Evaluation Function for Hand Strength Estimation in Gin Rummy

Aqib Ahmed, Joshua Leppo, Michal Lesniewski, Riken Patel, Jonathan Perez, Jeremy J. Blum

The Pennsylvania State University, Harrisburg
{aja6082, jzl6309, mkl5384, rkp8, jrp5717, jjb24}@psu.edu

Abstract

This paper describes a fast hand strength estimation model for the game of Gin Rummy. The algorithm is computationally inexpensive, and it incorporates not only cards in the player's hand but also cards known to be in the opponent's hand, cards in the discard pile, and the current game stage. This algorithm is used in conjunction with counterfactual regret (CFR) minimization to develop a gin rummy bot. CFR strategies were developed for the knocking strategies. The hand strength estimation algorithm was used to select a discard that balances the goals of maximizing the utility of the player's hand and minimizing the likelihood that a card will be useful to the opponent. A study of the parameterization of this estimation algorithm demonstrates the soundness of approach as well as good performance under a wide range of parameter values.

Introduction

In games of perfect information, players have full knowledge of the game state. Examples of these games include checkers, chess, and go. Game theoretic approaches, like alpha-beta pruning, can explore the game tree to find the best outcomes for a player (Samuel 1959). The size of the game tree is often the constraining factor for solving these types of games.

In games of imperfect information, players may have only partial information about the current game state. Examples of these types of games include poker and gin rummy, in which, for example, the cards in an opponent's hand may be unknown to the player. In these games, nodes in possible game trees are grouped into sets, called information sets, which contain nodes that are indistinguishable to the player due to the partial information available to them (Leyton-Brown and Shoham 2008). Players then play a common strategy for all nodes within an information set. For example, counterfactual regret (CFR) minimization finds an equilibrium strategy by minimizing a weighted regret of selected

actions within an information set, where the weights are based on the likelihood that a node would be reached by an opponent's strategy (Hart and Mas-Colell 2000).

A constraint for nodes within an information set is that they have the same set of available actions. For a discard decision in Gin Rummy, this constraint creates a challenge. The discard decision requires the selection of 1 of 11 cards. Given the more than 60 billion 11-card hands, the explosion in the number of information sets limits the ability to directly develop a CFR strategy for this decision point.

This paper introduces an approach to a Gin Rummy bot that uses CFR-based approach for knocking decisions, a heuristic approach to drawing decisions, and a hand strength estimation model for discarding decisions. The model is used to select a discard that attempts to maximize the current player's chances of winning. The approach incorporates not only the effect of discards on the player's cards, but also an estimate of the effect on the opponent's hand given the knowledge that the player has of the opponent's hand and the current game state. The resulting discard approach is computationally inexpensive, requiring the look ahead of just one turn in the game. Studies of parameterization show that the model's design choices are sound. In addition, the model is not extremely sensitive to the choice of parameter values. Instead, there is a range of values that work well.

Background

Gin Rummy

Gin Rummy is a part of the Rummy family of card games in which, generally speaking, there is a drawing and discarding action involved in making sets of similar cards, called melds. While the origins of Rummy are unclear, speculation of its origins includes the Chinese games Mah-Jong and Kun P'ai, and the Mexican game of Conquian (Parlett 2002).

The Gin Rummy variant of Rummy is thought to have been invented in 1909 in New York by Elwood T. Baker (NYT Archives 1950). Gin Rummy is a two-player game in which each player is dealt ten hidden cards from a 52-card deck. One card from the remaining 32 cards is turned face up to start a discard pile, while the other cards form the stockpile. In this turn-based game, each player first draws either the face-up card at the top of the discard pile, or the top face-down card from the stockpile. They then choose one card from their hand and place it face up on the top of the discard pile. One variation to the normal game play occurs on the first turn. If the first player chooses not to draw the face up card, the other player has an opportunity to draw.

Broadly speaking, the objective of the game is to group all of a player's cards into melds. The melds consist of three or more cards of the same rank, called a set, or three or more cards of the same suit with consecutive ranks, called a sequence. For the purposes of consecutive ranks, Aces are always low, followed by the numbered cards in ascending order, and then Jacks, Queens, and Kings. Each card has a point value, with aces worth one point, numbered cards worth the number of points on the card, and face cards worth ten points. The specific objective of a hand is to minimize the number of deadwood points, which is the sum of the points of the cards that are not in a meld.

A hand ends when there are only two cards left in the stockpile, or either of the two players chooses to end the hand by "knocking." If there are only two cards left in the stockpile, the hand is considered a tie. Once a player has ten or fewer deadwood points, they may knock when discarding. At this point, they reveal their hand and compare their deadwood points with the opponent's points. The non-knocking player has the ability to "lay off" unmelded cards, adding them to the opponent's melds if possible. Any cards that are laid off do not count towards the player's deadwood.

To determine the score of the hand, in the case where the knocker's deadwood is less than the opponent's, the knocker gets points equal to the difference in the deadwood points. In the case where the knocker has a greater or equal deadwood than the opponent, the opponent scores the difference between deadwoods plus a bonus, typically 10 or 25 points. In the case where the knocker has no deadwood, they have gone "gin." In this case, the opponent is not allowed to lay off any cards, and the knocker scores the opponent's deadwood points plus a gin bonus, which is typically either 20 or 25 points. A game consists of multiple hands and continues until one player has a cumulative score of at least 100 points.

Equilibrium Strategies in Imperfect Information Games

An imperfect information game G , in extensive form, can be represented as a tuple $G = (N, A, H, Z, \chi, \rho, \sigma, u, I)$, where (Leyton-Brown and Shoham 2008):

- N is a set of players.
- A is a set of actions available to players.
- H is a set of non-terminal nodes in the game tree. These internal nodes in the game tree represent decision points for the player, and a path from the root to a leaf represents the sequence of decisions in one instance of a game.
- Z is the set of terminal nodes in the game tree. Each node represents an outcome of a single game.
- χ , the action function, maps a non-terminal node in H to the actions available to a player at this node.
- ρ , the player function, maps a non-terminal node in H to the player who makes the decision at this node.
- σ , the successor function, maps a non-terminal node in the game tree and an action taken at that node to a successor node, in either H or Z . This function determines the next node in the game tree based on the current node and the action taken.
- u , is a utility function that maps a terminal node to a vector containing the payoff for each player should the game end at that particular node.
- I is a partition of the nodes in H into sets of nodes. The partition I typically will represent a set of game nodes that are indistinguishable to a player based on the partial information available.

For the game of Gin Rummy, we seek a Nash equilibrium strategy for our player. When players are playing Nash equilibrium strategies, no player can unilaterally change their strategy and improve their payoff (Leyton-Brown and Shoham 2008). In this sense, these strategies are non-exploitable.

Counterfactual Regret Minimization

Regret minimization provides one approach for finding Nash equilibrium strategies (Hart and Mas-Colell 2000). Regret, as defined in game theory, is similar to what humans are familiar with, the feeling of dissatisfaction over making a decision that results in unwanted outcomes, compounded with the knowledge that a better alternative was possible. A regret matching strategy is one where players choose actions proportional to the regret they experienced when not choosing that action in the past.

Counterfactual Regret Minimization (CFR) is a regret minimization algorithm that finds Nash equilibrium strategies in sequential imperfect-information games, where the players take turns taking actions at the decision points in a game (Zinkevich 2008). In the algorithm, the entire game tree is explored, with probabilities that a game node is reached being passed down the tree, and the payoffs for players being passed back up the tree.

As the game tree is explored, CFR calculates the total counterfactual regret of playing each action at every node.

First, the algorithm finds the expected utility of a node under the current strategy based on the probability of reaching each terminal node that is a descendant of this node and the payoff at the terminal node. Then, the algorithm calculates the regret of not playing each strategy by taking the utility at each successor node that is reached by taking an action and subtracting the expected utility of the current node. This regret is weighted by the probability that the other player would play to this node in the current strategy. It is counterfactual in that we assume that the current player is deliberately trying to play to this node. Zinkevitch *et al.* showed that the average counterfactual regret matching strategy converges to a Nash equilibrium strategy for zero-sum games (2008).

Exhaustive search of the game tree is infeasible in large games like Gin Rummy. Lanctot *et al.* (2009) demonstrated how Monte Carlo CFR (MCCFR) could sample actions in the game tree in such a way that the sampled counterfactual regrets approached the actual regrets in expectation.

Hand Strength Estimation

While the agent used MCCFR for the knocking strategy, there was a challenge in applying it to the discard strategies. Regret is calculated on an information set basis, and all nodes in an information set must have the same set of actions available. Since these actions in the discard strategy depend on cards in a player's hand, there is an explosion in the number of information sets for this strategy.

Instead, the discard strategy used a hand strength estimation approach. In previous research, generic approaches to player state estimation heuristics used models based on the expected payoff for a player, the degree to which a player controls the outcome of a game, and the number of turns remaining in the game (Clune 2007). In the strength estimation heuristic described in the next section, these concepts are mapped to observable attributes including the cards held by the player, cards that have been discarded, cards known to be held by the opponent, and the current turn number.

Methodology

As shown in Figure 1, the agent is built on three primary strategies implemented as three different methods for each decision point in the game: drawing, discarding, and knocking. The knocking strategy was developed using an MCCFR approach. A simplified drawing strategy and discard strategy was implemented, and regret captured based on an abstraction of the game state that minimized the number of information sets that needed to be considered. The resulting strategy for knocking was based on the regret matching strategies from the MCCFR training process.

The drawing strategy uses a simple heuristic. If the face-up card does not contribute to making a meld, the player

draws the face-down card. Otherwise, if the face-up card contributes to a meld which reduced the total number of deadwood points, the face-up card is drawn.

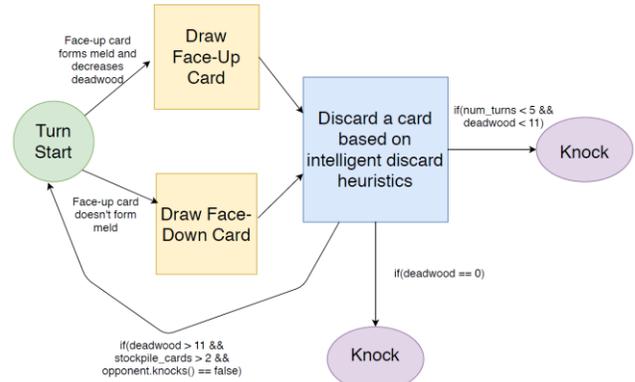


Figure 1. Block diagram of the three primary strategies.

The knocking strategy used an abstraction that included the number of deadwood points and the turn number. If there is no deadwood, the agent always knocks since gin is possible. Otherwise, the agent will knock in its first four turns, if possible. If it is later than the fourth turn, the agent will not knock, but instead try to get gin or undercut the opponent.

The knocking strategy that was implemented based on the strategies found by MCCFR put a large emphasis on creating melds to achieve gin or undercut the opponent. This emphasis helped in the design of the hand strength estimation algorithm used for discarding.

When an agent needs to make a discard decision, it scores each discard that is not participating in a meld and selects the card with the highest score. The formula used to score a discard c is:

$$s(c) = d_0(c) + \alpha\beta^t d_f(c) + \sum \delta_i \omega_i \quad (1)$$

where:

- $d_0(c)$ is the deadwood that is in our hand if we discarded card c .
- α is an initial weighting term that controls the relative value of deadwood at the turn versus future expected deadwood.
- β is a decay weighting term, set to be between 0 and 1, that allows for exponential decay of the value of future expected deadwood as the game progresses.
- t is the turn number in the game. Each time the agent draws a card, this turn number increments.
- $d_f(c)$ is an estimate of future deadwood if we discarded card c , calculated as described below.
- δ_i is an indicator variable which evaluates to 1 if the card contains the i^{th} attribute relating to the usefulness of this card to the opponent, as described below.
- ω_i is a weight for the value of the i^{th} attribute.

To calculate the expected deadwood in the hand, the agent looks one turn in the future. Each card that has not yet been seen is added to the hand, and the card that minimizes deadwood is discarded. The top k deadwood values then are averaged together to calculate $d_f(c)$. Empirically, relatively small values of k , between 4 to 8, seemed to work well. Interestingly this is much smaller than the number of unseen cards in most cases. It is close, though, to the maximum number of cards that could be drawn and made into melds in a typical hand. This result seems consistent with the emphasis on going gin or undercutting an opponent, suggested by the CFR knocking strategy.

The indicator variables in formula 1 are associated with pairs of attributes that reflect whether it is likely that an opponent will be able to use the discarded card in a meld. These attributes include penalties, which indicate that the discard is likely to help the opponent, as well as bonuses, which indicate that the discard is less likely to help the opponent. The penalties are positive values since the card chosen is the one with the minimum score, while the corresponding bonuses are negative values.

1. Sequence: The penalty version of this indicator variable is set to 1 if the opponent is known to have cards that could make a sequence with this card. In other words, the opponent has two cards of the same suit with nearby ranks, such that this card would be in a meld that is a run or sequence. The bonus version of this variable is set if the discard cannot make a sequence/run for the opponent because the suited cards with nearby ranks have either been discarded or are in the player's hand.
2. Set: The penalty indicator variable is set if the opponent has two cards of the same rank as the discard. The bonus variable is set if the discard cannot make a set for the opponent because two or more cards with the same ranks have either been discarded or are in the player's hand.
3. Rank: The penalty variable indicates that the opponent is known to have one card of the same rank as the discard. The variable is set if one other card with the same rank is not available to the opponent, because it has been discarded or is held by the player.
4. Adjacent: The penalty indicator variable is set if the opponent is known to have a card of the same suit as the discard that is one rank away. The bonus variable is set if one card, of the same suit and with an adjacent rank, is not available to the opponent.
5. 2-Adjacent: The penalty indicator variable is set if the opponent is known to have a card of the same suit as the discard that is two ranks away. The bonus variable is set

if one card, of the same suit and with rank that is two away from the discard, is not available to the opponent.

Note that the penalty version of the first two of these variables indicate that the opponent will definitely be able to make a particular type of meld with this card, where the bonus version indicates that the opponent cannot make a particular type of meld with this card. The last three variable pairs represent less certainty. These variables are set if one of the cards needed for a particular 3-card meld is known to be held by the opponent (in the penalty version), or known to be unavailable to the opponent (in the bonus version).

Evaluation of the Hand Strength Estimation Algorithm

To evaluate the hand strength estimation algorithm, the authors searched for a set of parameters which worked well against the base agent from the EAAI competition¹, as well as a set of base agents using a variety of heuristic approaches. Then each of the parameter values was varied to evaluate the soundness of including the parameter in the model and the sensitivity of the model to the parameters' values.

The parameters for the baseline agent were chosen as follows:

- Initial Weighting Term: 20
- Decay Weighting Term: 0.9
- Draw Cards to Consider: 6
- Sequence: +20 points for penalty, -10 point for bonus
- Set: +20 points for penalty, -10 point for bonus
- Rank: +4 points for penalty, -4 point for bonus
- Adjacent: +5 points for penalty, -5 point for bonus
- 2-Adjacent: +3 points for penalty, -3 point for bonus

The agent was then evaluated by playing 50,000 games against itself, with one of the parameter pairs changed.

The first three parameters are intended to capture the effect of a discard on the agent's expected deadwood. These parameters balance immediate deadwood reduction that results in discarding a card and possible future deadwood reduction that would occur if the card were retained to be added to a meld after a obtaining additional draw cards.

Figure 2 shows the results when varying the initial weighting term. Increasing the value of this parameter places less weight on immediate reductions in deadwood value and instead favors larger reductions in deadwood values that would be possible by adding cards to melds. As shown in this figure, the agent performs quite poorly at low values for this parameter, indicating that it is important to

¹ Neller, T.; Heibel, J.; and Francis, P.E. 2019. "Gin-Rummy-EAAI: Gin Rummy software for the Gin Rummy EAAI Undergraduate Research Challenge." <https://github.com/tneller/gin-rummy-eai>. Accessed: 12/1/2020.

weight future deadwood reductions more heavily than immediate ones. However, there is a large range of values greater than 20 that perform well, winning approximately 50% of the time against the baseline agent. The large range of values indicates that for sufficiently large values, the model is not particularly sensitive to the exact value for this parameter. Moreover, while the optimal value appears to be about 40, the fact that this parameter value yielded only a 51% win rate indicates that the chosen value of 20 for this parameter is a fairly good one.

Figure 3 shows the results when varying the decay weighting term. The agent performed best when the decay parameter was set to between 0.9 and 1.0, with the highest win rate of 0.5073 reached at a parameter value of 0.94. This result supports the idea that the value of future deadwood improvements reduces as the game progresses. As before, the performance of the agent is fairly consistent within this range, indicating that the model is not overly sensitive to the exact value of this parameter. Moreover, the value chosen of 0.9 for this parameter seems to be a good one.

In order to determine the potential for future deadwood reduction, only the top k deadwood values resulting after a possible draw card are considered. Figure 4 shows the effect of varying this parameter from the chosen value of 6. The best values occur in the range of 3 to 6, indicating that the model does best when only the largest reductions in deadwood values are considered. The optimal winning rate of 0.5059 occurs at a parameter value of 5.

The remaining parameter values are focused on capturing the potential of a discard to improve the opponent's hand. These parameters include ones that capture whether a card definitely could be or could not be used by opponent in a meld, as well as ones

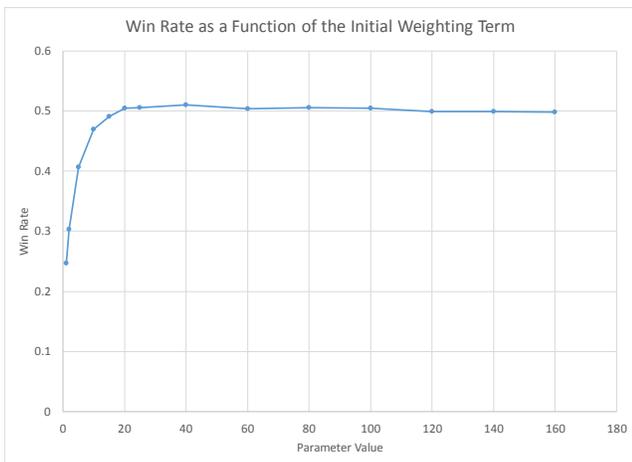


Figure 2. Win rate of agent with varying initial weighting parameters against one with a fixed value of 20.

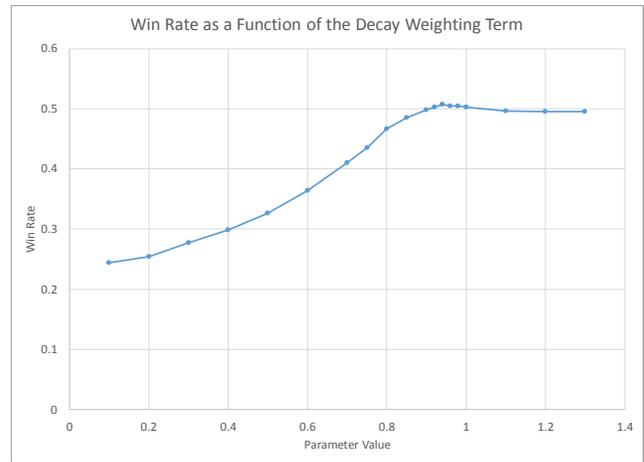


Figure 3. Win rate of agent with varying decay weighting parameters against one with a fixed value of 0.9.

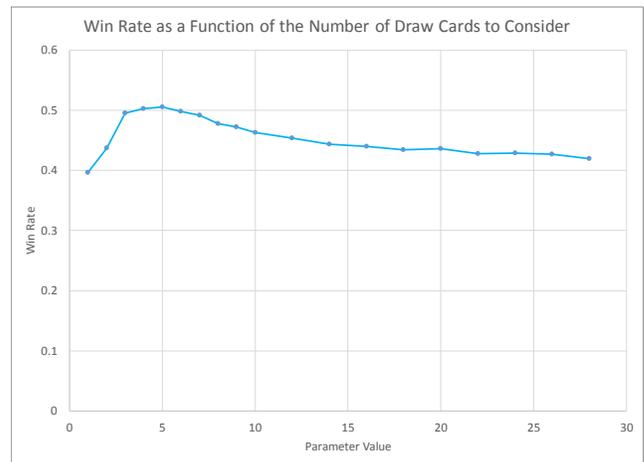


Figure 4. Win rate as a function of varying the number of draw cards considered, when estimating future deadwood reductions.

For example, figure 5 shows the results when varying the definite meld parameters, the sequence and set bonuses and penalties. In these experiments, the bonuses and penalty values for one of the parameters was set to be equal in magnitude. It was then compared against the base line parameterization that used +20 for a penalty value and -10 for a bonus value. As shown in the figure, when these values are set to zero, effectively removing them from the model, the performance of the model drops, with win rate of between 0.4691 and 0.4803. Moreover, for the cases at the left side of the graph, when the sign of the penalties of bonuses are incorrect, the performance of the system degrades as the magnitude of these values increases. There does seem to be a range of parameter values, though, that performs well. It appears that the weights of the set and sequence parameters should be roughly equal. As in the previous experiments, the fact that the win rate of the best parameterizations was close to

0.5 against the agent with the baseline parameters indicates that these baseline parameters are fairly good.

Figure 6 shows the results of varying the possible meld parameters. As with the definite meld parameters, one pair of parameters was changed at a time, with the magnitude of the bonus and penalty parameters equal in each experiment against the baseline agent. As shown in the figure, these parameters are all important to the model. Removing any of these parameter pairs from the model, by setting the values to 0, reduces the agent performance. In addition, setting the sign on these parameters incorrectly, as shown with the tests at the left of the figure also reduces performance.

The vertical axis does not start at 0 in figure 6, in order to be able to examine subtle differences in these parameter values. All have a range of values that performs on par with the agent with baseline parameter values. The results seem to indicate that the adjacent parameter value should be the largest, since it achieves the highest win rate of 0.5045 at a value of 3 for the bonus and -3 for the penalty. Next in terms of importance would be the rank parameter value, which has a highest win rate of 0.5023 at a value of 2 for the bonus and -2 for the penalty. Finally, comes the 2-adjacent parameter, which achieves its best win rate of 0.5002 at a values of ± 1 . Note that since these best win rates are all near 0.5, the parameter values that were chosen for the baseline agent appear to be good ones.

This study of the parameterization has limitations which should be noted. First the parameterization studied the performance of the agent against a fixed baseline agent. Variations in the opponent's play would affect the hand strength estimation model, particularly from the perspective of the parameter values that capture the potential benefit of a discard to the opponent. Moreover, the parameterization study varied just one or two parameter values at a time. Exploring more of the space of parameter values could yield better performance from the agent.

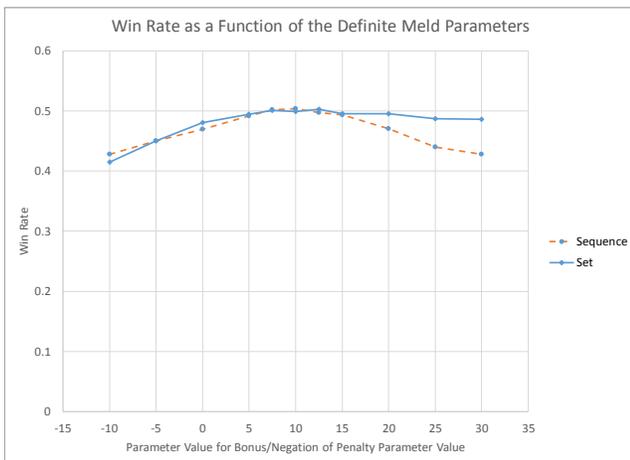


Figure 5. Win rate against baseline agent with varying definite meld parameters.

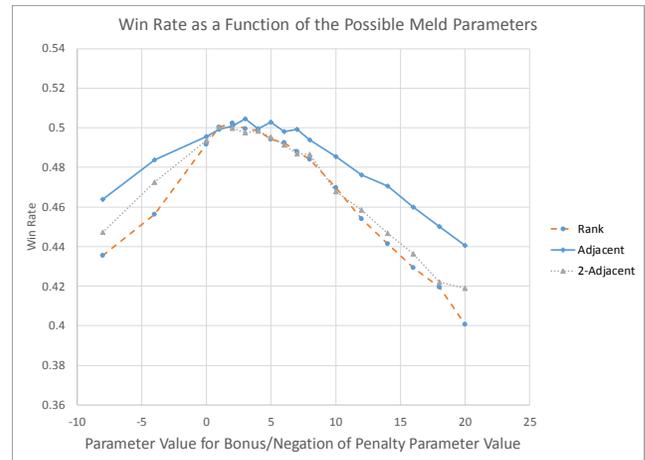


Figure 6: Win rate against agent with baseline parameters, when varying possible meld parameter values

Conclusions

An agent capable of playing gin rummy was developed using strategies based on a counterfactual regret (CFR) minimization and a novel hand-strength estimation algorithm. A simple heuristic approach was used for drawing decisions. CFR was applied to extract a strategy for the knocking decision points for the agent. The hand-strength estimation algorithm was designed with three key aspects: a mechanism to balance short- and long-term improvements in the agent's hand, the ability to discount the worth of long-term improvements as the game progresses, and a capability to estimate the likelihood that a discard would benefit the opponent.

A study of the parameterization of the hand-strength model demonstrated value in this approach. All of the parameters had fairly wide ranges of values for which the model performed well. Moreover, for parameters that could be removed from the model, removal reduced agent performance.

References

- Clune, J. 2007. Heuristic evaluation functions for general game playing, Proceedings of the 22nd national conference on Artificial intelligence (2), pp. 1134-39.
- Hart, S. and Mas-Colell, A. 2000. A Simple Adaptive Procedure Leading to Correlated Equilibrium. *Econometrica* 68(5): 1127-50.
- Lanctot, M.; Waugh, K.; and Bowling, M. 2009. Monte Carlo Sampling for Regret Minimization in Extensive Games. *Advances in Neural Information Processing Systems*, 1078-1086.
- Leyton-Brown, K., and Shoham, Y. 2008. *Essentials of Game Theory: A Concise Multi-Disciplinary Approach*. Morgan & Claypool Publishers.

NYT Archives, "C.G. Baker, Helped Devise Gin Rummy." The New York Times, The New York Times, 17 May 1950, www.nytimes.com/1950/05/17/archives/cg-baker-helped-devise-gin-rummy.html.

Parlett, David. 2020. "Gin Rummy: The Game of the Stars". Historic Card Games. www.parlettgames.uk/histocs/ginrummy.html.

Samuel, A.L. 1959. "Some Studies in Machine Learning Using the Game of Checkers," IBM Journal of Research and Development, 3(3), pp. 210-29.

Zinkevich, M., Johanson, M., Bowling, M., and Piccione, C. 2008. Regret Minimization in Games with Incomplete Information. *Advances in Neural Information Processing Systems* (20): 1729-36.