DOI:10.1145/3131284

Heads-Up Limit Hold’em Poker Is Solved

By Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin

Abstract

Poker is a family of games that exhibit imperfect information, where players do not have full knowledge of past events. While many perfect information games have been solved (e.g., Connect-Four and checkers), no nontrivial imperfect information game played competitively by humans has previously been solved. In this paper, we announce that the smallest variant of poker in-play, heads-up limit Texas hold’em, is now essentially weakly solved. Furthermore, this computation formally proves the common wisdom that the dealer in the game holds a significant advantage. This result was enabled by a new algorithm, CFR$^+$, which is capable of solving extensive-form games three orders of magnitude larger than previously possible. This paper is an extended version of the original 2015 *Science* article,[9] with additional results showing Cepheus’ in-game performance against computer and human opponents.

1. INTRODUCTION

Games have been intertwined with the earliest developments in computation, game theory, and Artificial Intelligence (AI). At the very conception of computing, Babbage had detailed plans for an “automaton” capable of playing tic-tac-toe and dreamt of his Analytical Engine playing chess.[4] Both Alan Turing[46] and Claude Shannon,[40] on paper and in hardware respectively, developed programs to play chess as validation of early ideas in computation and AI. For over a half-century, games have continued to act as testbeds for new ideas and the resulting successes have marked significant milestones in the progress of AI: For example, the checkers-playing computer program Chinook becoming the first to win a world championship title against humans,[38] Deep Blue defeating Kasparov in chess,[14] and Watson defeating Jennings and Rutter on *Jeopardy!*[17] However, defeating top human players is not the same as “solving” a game, that is, computing a game-theoretically optimal solution that is incapable of losing against any opponent in a fair game. Solving games has also served as notable milestones for the advancement of AI, for example, Connect-Four[2] and checkers.[39]

Every nontrivial game played competitively by humans that has been solved to-date is a *perfect information game*.[a] In perfect information games, all players are informed of everything that has occurred in the game prior to making a decision. Chess, checkers, and backgammon are examples of perfect information games. In *imperfect information games*, players do not always have full knowledge of past events (e.g., cards dealt to other players in bridge and poker, or a seller’s knowledge of the value of an item in an auction). These games are more challenging, with theory, computational algorithms, and instances of solved games lagging behind results in the perfect information setting.[b] And, while perfect information may be a common property of parlor games, it is far less common in real-world decision making settings. In a conversation recounted by Bronkowski, John von Neumann, the founder of modern game theory, made the same observation, “Real life is not like that. Real life consists of bluffing, of little tactics of deception, of asking yourself what is the other man going to think I mean to do. And that is what games are about in my theory.”[12]

Von Neumann’s statement hints at the quintessential game of imperfect information: the game of poker. Poker involves each player being dealt private cards, with players taking structured turns making bets on having the strongest hand (possibly bluffing), calling opponent bets, or folding to give up the hand. Poker played an important role in the early developments of the field of game theory. Borel[7] and von Neumann’s[47, 48] foundational works were motivated by developing a mathematical rationale for bluffing in poker, and small synthetic poker games[c] were commonplace in many early papers.[7, 29, 32, 48] Poker is also arguably the most popular card game in the world with over 150mn players worldwide.[1] The most popular variant of poker today is Texas hold’em. When it is played with just two-players (heads-up) and with fixed bet-sizes and number of raises (limit), it is called Heads-Up Limit Hold’Em (HULHE).[10] HULHE was popularized by a series

[a] We use the word trivial to describe a game that can be solved without the use of a machine. The one near-exception to this claim is oshi-zumo, but it is not played competitively by humans and is a simultaneous-move game that otherwise has perfect information.[13] Furthermore, almost all nontrivial games played by humans that have been solved to-date also have no chance elements. The one notable exception is hypergammon, a three-checker variant of backgammon invented by Hugh Sconyers in 1993 which he then strongly solved, that is, the game-theoretic value is known for all board positions. It has seen play in human competitions. See http://www.bkgm.com/variants/HyperBackgammon.html (accessed July 4, 2014).

[b] For example, Zermelo proved the solvability of finite, two-player, zero-sum, perfect information games in 1913,[51] while von Neuman’s more general minimax theorem appeared in 1928.[47] Minimax and alpha-beta pruning, the fundamental computational algorithm for perfect information games, was developed in the 1950s, while Koller and Megiddo’s first polynomial-time technique for imperfect information games was introduced in 1992.[26]

[c] We use the word synthetic to describe a game that was invented for the purpose of being studied or solved rather than played by humans. A synthetic game may be trivial, such as Kuhn poker,[29] or nontrivial such as Rhode Island hold’em.[41]

The original version of this paper was published in *Science 347*, 6218 (Jan. 2015) 145–149. Adapted with permission from AAAS.

of high-stakes games chronicled in the book *The Professor, the Banker, and the Suicide King*.[16] It is also the smallest variant of poker played competitively by humans. HULHE has 3.16×10^{17} possible states the game can reach making it larger than Connect Four and smaller than checkers. However, as an imperfect information game, many of these states cannot be distinguished by the acting player as they involve information about unseen past events (i.e., private cards dealt to the opponent). As a result, the game has 3.19×10^{14} decision points where a player is required to make a decision.

While smaller than checkers, the imperfect information nature of HULHE makes it a far more challenging game for computers to play or solve. It was 17 years after Chinook won its first game against world champion Marion Tinsley in checkers that the computer program Polaris won the first meaningful match against professional poker players.[34] While Schaeffer et al. solved checkers in 2007,[39] heads-up limit Texas hold'em poker, until now, was unsolved. This slow progress is not for lack of effort. Poker has been a challenge problem for artificial intelligence, operations research, and psychology with work going back over 40 years.[6] 17 years ago, Koller and Pfeffer[28] declared, "we are nowhere close to being able to solve huge games such as full-scale poker, and it is unlikely that we will ever be able to do so." The focus on HULHE as one example of "full-scale poker" began in earnest over ten years ago,[5] and became the focus of dozens of research groups and hobbyists after 2006 when it became the inaugural event in the Annual Computer Poker Competition,[53] held in conjunction with the main conference of the Association for the Advancement of Artificial Intelligence (AAAI). This paper is the culmination of this sustained research effort toward solving a "full-scale" poker game.[10]

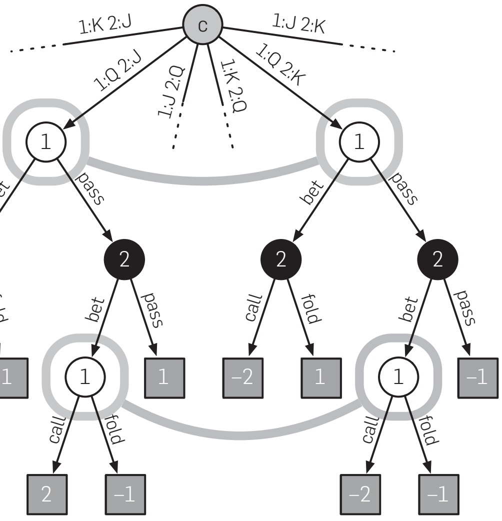
Allis[3] gives three different definitions for solving a game. A game is said to be *ultra-weakly solved* if for the initial position(s), the game-theoretic value has been determined; *weakly solved* if for the initial position(s), a strategy has been determined to obtain at least the game-theoretic value, for both players, under reasonable resources; and *strongly solved* if for all legal positions, a strategy has been determined to obtain the game-theoretic value of the position, for both players, under reasonable resources. In an imperfect information game, where the game-theoretic value of a position beyond the initial position is not unique, Allis's notion of "strongly solved" is not well-defined. Furthermore, imperfect information games, due to stochasticity in the players' strategies or the game itself, typically have game-theoretic values that are real-valued rather than discretely valued (such as "win," "loss," and "draw" in chess and checkers), and only achieved in expectation over many playing of the game. As a result, game-theoretic values are often approximated, and so an additional consideration in solving a game is the degree of approximation in a solution. A natural level of approximation under which a game is *essentially weakly solved* is if a human lifetime of play is not sufficient to establish with statistical significance that the strategy is not an exact solution.

In this paper, we announce that heads-up limit Texas hold'em poker is essentially weakly solved. Furthermore, we bound the game-theoretic value of the game, proving that the game is a winning game for the dealer.

2. SOLVING IMPERFECT INFORMATION GAMES

The classical representation for an imperfect information setting is the *extensive-form game*. Here the word "game" refers to a formal model of interaction between self-interested agents and applies to both recreational games and serious endeavors such as auctions, negotiation, and security. See Figure 1 for a graphical depiction of a portion of a simple poker game in extensive-form. The core of an extensive-form game is a *game tree* specifying branches of possible events, namely player actions or chance outcomes. The branches of the tree split at *game states* and each is associated with one of the players (or chance) who is responsible for determining the result of that event. The leaves of the tree signify the end of the game, and have an associated utility for each player. The states associated with a player are partitioned into *information sets*, which are sets of states which the acting player cannot distinguish between (e.g., corresponding to states where the opponent was dealt different private cards). The branches from states within an information set are the player's available *actions*. A *strategy* for a player specifies for each information set a probability distribution over the

Figure 1. Portion of the extensive-form game representation of three-card Kuhn poker[29] where player 1 is dealt a queen (Q) and the opponent is given either the Jack (J) or King (K). Game states are circles labeled by the player acting at each state ("c" refers to chance, which randomly chooses the initial deal). The arrows show the events the acting player can choose from, labeled with their in-game meaning. The leaves are square vertices labeled with the associated utility for player 1 (player 2's utility is the negation of player 1's). The states connected by thick gray lines are part of the same information set, that is, player 1 cannot distinguish between the states in each pair since they represent a different unobserved card being dealt to the opponent. Player 2's states are also in information sets, containing other states not pictured in this diagram.

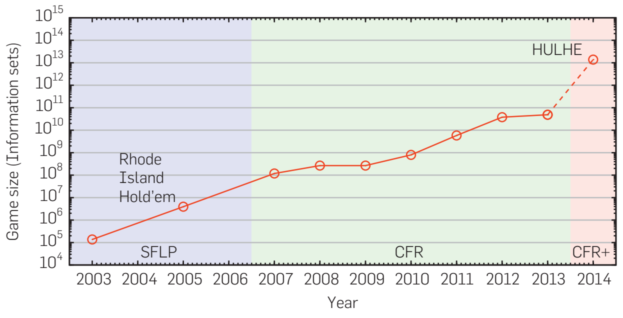


available actions. If the game has exactly two players and the utilities at every leaf sum to zero, the game is called *zero-sum*.

The classical solution concept for games is a *Nash equilibrium*, a strategy for each player such that no player can increase their expected utility by unilaterally choosing a different strategy. All finite extensive-form games have at least one Nash equilibrium. In zero-sum games, all equilibria have the same expected utilities for the players, and this value is called the *game-theoretic value of the game*. An *ϵ-Nash equilibrium* is a strategy for each player where no player can increase their utility by more than ϵ by choosing a different strategy. By Allis's categories, a zero-sum game is ultra-weakly solved if its game-theoretic value is computed, and weakly solved if a Nash equilibrium strategy is computed. We call a game essentially weakly solved if an ϵ-Nash equilibrium is computed for a sufficiently small ϵ to be statistically indistinguishable from zero in a human lifetime of played games. For perfect information games, solving typically involves a (partial) traversal of the game tree. However, the same techniques cannot apply to imperfect information settings. We briefly review the advances in solving imperfect information games, benchmarking the algorithms by their progress in solving increasingly larger synthetic poker games as summarized shown in Figure 2.

Normal-Form Linear Programming. The earliest method for solving extensive-form games involved converting it into a *normal-form game*, represented as a matrix of values for every pair of possible deterministic strategies in the original extensive-form game, and then solving it with a Linear Program (LP). Unfortunately, the number of possible deterministic strategies is exponential in the number information sets of the game. So, while LPs can handle normal-form games with many thousands of strategies, even just a few dozen decision points makes this method impractical. Kuhn poker, a poker game with three cards, one betting round, and a one bet maximum having a total of 12 information sets (see Figure 1), can be solved with this approach. But even Leduc hold'em,[42] with six cards, two betting rounds, and a two bet maximum having a total of only 288 information sets, is intractable having over 10^{86} possible deterministic strategies.

Figure 2. Increasing sizes of imperfect information games solved over time. The shaded regions refer to the technique used to achieve the result with references in the main text. CFR^{+} is the algorithm used in this work and the dashed line shows the result established in this paper.



Sequence-Form Linear Programming. Romanovskii[35] and later Koller et al.[26, 27] established the modern era of solving imperfect information games, introducing the sequence-form representation of a strategy. With this simple change of variables, they showed that the extensive-form game could be solved directly as an LP, without the need for an exponential conversion to normal-form. Sequence-Form Linear Program (SFLP) was the first algorithm to solve imperfect information extensive-form games with computation time that grows as a polynomial of the size of the game representation. In 2003, Billings et al.[5] applied this technique to poker, solving a set of simplifications of HULHE to build the first competitive poker-playing program. In 2005, Gilpin and Sandholm[19] used the approach along with an automated technique for finding game symmetries to solve Rhode Island Hold'em,[41] a synthetic poker game with 3.94×10^6 information sets after symmetries are removed.

Counterfactual Regret Minimization. In 2006, the Annual Computer Poker Competition was started.[53] The competition drove significant advancements in solving larger and larger games, with multiple techniques and refinements being proposed in the years that followed.[36, 37] One of the techniques to emerge, and currently the most widely adopted in the competition, is Counterfactual Regret Minimization (CFR).[d] CFR is an iterative method for approximating a Nash equilibrium of an extensive-form game through the process of repeated self-play between two regret-minimizing algorithms.[10, 52] *Regret* is the loss in utility an algorithm suffers for not having selected the single best deterministic strategy, which can only be known in hindsight. A *regret-minimizing* algorithm is one that guarantees its regret grows sub-linearly over time, and so eventually achieves the same utility as the best deterministic strategy. The key insight of CFR is that instead of storing and minimizing regret for the exponential number of deterministic strategies, CFR stores and minimizes a modified regret for each information set and subsequent action, which can be used to form an upper bound on the regret for any deterministic strategy. An approximate Nash equilibrium is retrieved by averaging each player's strategies over all of the iterations, and the approximation improves as the number of iterations increases. The memory needed for the algorithm is linear in the number of information sets, rather than quadratic, which is the case for efficient LP methods.[25] Since solving large games is usually memory-bound, CFR has resulted in as dramatic an increase in the size of solved games as Koller et al.'s advance. Since its introduction in

[d] Another notable algorithm to emerge from the Annual Computer Poker Competition is an application of Nesterov's excessive gap technique[33] to solving extensive form games.[18] The technique has some desirable properties, including better asymptotic time complexity that what is known for CFR. However, it has not seen widespread use among competition participants due to its lack of flexibility in incorporating sampling schemes and its inability to be used with powerful (but unsound) abstractions that employ imperfect recall. Recently, Waugh and Bagnell[49] have shown that CFR and the excessive gap technique are more alike than different, suggesting that the individual advantages of each approach may be attainable in the other.

2007, CFR has been used to solve increasingly complex simplifications of HULHE, reaching as many as 3.8×10^{10} information sets in 2012.[20]

3. SOLVING HEADS-UP LIMIT HOLD'EM

The full game of HULHE has 3.19×10^{14} information sets. Even after removing game symmetries it has 1.38×10^{13}, that is, three orders of magnitude larger than previously solved games. There are two challenges for established CFR variants to handle games at this scale: memory and computation. During computation CFR must store the resulting solution and the accumulated regret values for each information set. Even with single-precision (four byte) floating point numbers, this requires 262TB of storage. Furthermore, past experience has shown that a three order of magnitude increase in the number of information sets requires at least three orders of magnitude more computation. In order to tackle these two challenges we employ two ideas recently proposed by Tammelin, a co-author of this paper.[44]

To address the memory challenge we store the approximate solution strategy and accumulated regrets using compression. For the solution and regrets we use fixed-point arithmetic by first multiplying all values by a scaling factor and truncating them to integers. The resulting integers are then ordered to maximize compression efficiency, with compression ratios around 13-to-1. Overall, we require under 11TB of storage during the computation, which is distributed across a cluster of computation nodes. This amount is in-feasible to store in main memory, and so we store the compressed strategy and regret values on each node's local disk. Each node is responsible for a set of *subgames*, that is, portions of the game tree partitioned based on publicly observed actions and cards so that each information set is associated with one subgame. The regrets and strategy for a subgame are loaded from disk, updated, and saved back to disk, using a streaming compression technique that decompresses and recompresses portions of the subgame as needed. By making the subgames large enough, the update-time dominates the total time to process a subgame. With disk pre-caching, the inefficiency incurred by disk storage is approximately 5% of the total time.

To address the computation challenge we use a variant of CFR called CFR^+.[10, 44] CFR implementations typically sample only portions of the game tree to update on each iteration. They also employ regret-matching at each information set, which maintains regrets for each action and chooses among actions with positive regret with probability proportional to that regret. Instead, CFR^+ does exhaustive iterations over the entire game tree, and uses regret-matching$^+$, a variant of regret-matching where regrets are constrained to be non-negative. Actions that have appeared poor (with less than zero regret for not having been played) will be chosen again immediately after proving useful (rather than waiting many iterations for the regret to become positive). Finally, in contrast with CFR, we have observed empirically that the exploitability of the players' strategies during the computation regularly converges to zero. Therefore, we skip the step of computing and storing the average strategy, instead using the players' current strategies as the CFR^+ solution. We have empirically observed CFR^+ to require considerably less computation than state-of-the-art sampling CFR,[22] while also being highly suitable for massive parallelization.

THEOREM 1.[e] *Given a set of actions A, and any sequence of T value functions $v^t: A \mapsto \Re$ with a bound L such that $|v^t(a) - v^t(b)| \leq L$ for all t and $a, b \in A$, an agent acting according to the regret-matching$^+$ algorithm will have regret of at most $L\sqrt{|A|T}$.*

Like CFR, CFR^+ is an iterative algorithm that computes successive approximations to a Nash equilibrium solution. The quality of the approximation can be measured by its *exploitability:* the amount less than the game value that the strategy achieves against the worst-case opponent strategy in expectation.[10] Computing the exploitability of a strategy involves computing this worst-case value, traditionally requiring a traversal of the entire game tree. This was long thought to be intractable for games the size of HULHE. Recently it was shown that this calculation could be dramatically accelerated by exploiting the imperfect information structure of the game and regularities in the utilities.[23] This is the technique we use to confirm the approximation quality of our resulting strategy. The technique and implementation has been verified on small games and against independent calculations of the exploitability of simple strategies in HULHE.

A strategy can be exploitable in expectation and yet, due to chance elements in the game and randomization in the strategy, its worst-case opponent still is not guaranteed to be winning after any finite number of hands. We define a game to be *essentially solved* if a lifetime of play is unable to statistically differentiate it from being solved at 95% confidence. Imagine someone playing 200 hands of poker an hour for 12hrs a day without missing a day for 70 years. Furthermore imagine them employing the worst-case, maximally exploitive, opponent strategy, and never making a mistake. Their total winnings, as a sum of many millions of independent outcomes, would be normally distributed. Hence, the observed winnings in this lifetime of poker would be 1.64 standard deviations or more below its expected value (i.e., the strategy's exploitability) at least 1 time out of 20. Using the standard deviation of a single hand of HULHE, which has been reported to be around 5bb/g (big-blinds per game, where the big-blind is the unit of stakes in HULHE),[11] we arrive at a threshold of $1.64 * 5 / \sqrt{200 * 12 * 365 * 70} \approx 0.00105$. So, an approximate solution with an exploitability under 1mbb/g (milli-big-blinds per game) cannot be distinguished with high confidence from an exact solution, and indeed has a 1-in-20 chance of winning against its worst-case adversary even after a human lifetime of games. Hence, 1mbb/g is the threshold for declaring HULHE essentially solved.

4. THE SOLUTION

Our CFR^+ implementation was executed on a cluster of 200 computation nodes each with 24 2.1GHz AMD cores, 32GB

[e] Theorem 1 and others providing the theoretical support for CFR^+ did not appear in the original version of this article, and were published in a subsequent paper.[45]

of Random Access Memory (RAM), and a 1TB local disk. We divided the game into 110,565 subgames (partitioned based on preflop betting, flop cards, and flop betting). The subgames were split among 199 worker nodes, with one parent node responsible for the initial portion of the game-tree. The worker nodes performed their updates in parallel, passing values back to the parent node for it to perform its update, taking 61 min on average to complete one iteration. The computation was then run for 1,579 iterations, taking 68.5 days, and using a total of 900 core years of computation[f] and 10.9TB of disk space, including file system overhead from the large number of files.

Figure 3 shows the exploitability of the computed strategy with increasing computation. The strategy reaches an exploitability of 0.986mbb/g, making HULHE essentially weakly solved. Using the separate exploitability values for each position (as the dealer and non-dealer) we get exact bounds on the game-theoretic value of the game: between 87.7mbb/g and 89.7mbb/g for the dealer, proving the common wisdom that the dealer holds a significant advantage in HULHE.

The final strategy, as a close approximation to a Nash equilibrium, can also answer some fundamental and long-debated questions about game-theoretically optimal play in HULHE. Figure 4 gives a glimpse of the final strategy in two early decisions of the game. Human players have disagreed about whether it may be desirable to "limp," that is, call as the very first action rather than raise, with certain hands. Conventional wisdom is that limping forgoes the opportunity to provoke an immediate fold by the opponent, and so raising is preferred. Our solution emphatically agrees (see the absence of blue in Figure 4a). The strategy limps just 0.06% of the time and with no hand more than 0.5%. In other situations, the strategy gives insights beyond conventional wisdom, indicating areas where humans might improve. The strategy rarely "caps," that is, makes the final allowed raise, in the first round as the dealer, whereas some strong human players cap the betting with a wide range of hands. Even when holding the strongest hand, a pair of aces, the strategy caps the betting less than 0.01%, and the hand most likely to cap is a pair of twos, with probability 0.06%. Perhaps more importantly, the strategy chooses to play, that is, not fold, a broader range of hands as the non-dealer than most human players (see the relatively small amount of red in Figure 4b). It is also much more likely to re-raise when holding a low-rank pair (such as threes or fours).[g]

While these observations are only for one example of game-theoretically optimal play (different Nash equilibria may play differently), they both confirm as well as contradict current human beliefs about equilibria play, and illustrate that humans can learn considerably from such large-scale game-theoretic reasoning.

5. IN-GAME RESULTS

In this extended version of the original paper,[9] we present additional results measuring Cepheus' in-game performance against computer agents and human opponents. HULHE has served as a common testbed for artificial intelligence research for more than a decade, and researchers have produced a long series of computer agents for the domain. This effort was largely coordinated by the Annual Computer Poker Competition (ACPC) which begain in 2006 with HULHE. While each year's top agents outperformed the older agents in the competition, and so appeared to be converging to optimal play, their actual worst-case exploitability was unknown. In 2011, an efficient best response technique was developed that made it feasible to measure a computer agent's exploitability,[23] and for the first time researchers were able to exactly measure their progress towards the goal of solving the game. A key result in that paper was that top ACPC agents only defeated each other by tiny margins, and yet had a wide range of exploitability. Using Cepheus, we can now also evaluate these historical agents through matches against an essentially optimal strategy.

[f] The total time and number of core years is larger than was strictly necessary as it includes computation of an average strategy that was later measured to be more exploitable than the current strategy and so discarded. The total space noted, on the other hand, is without storing the average strategy.

[g] These insights were the result of discussions with Mr. Bryce Paradis, previously a professional poker player who specialized in HULHE.

Figure 3. Exploitability of the approximate solution with increasing computation.

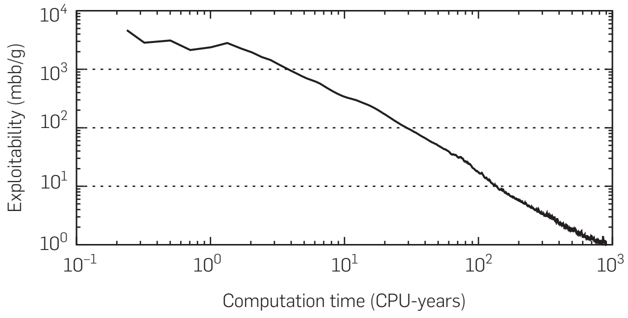
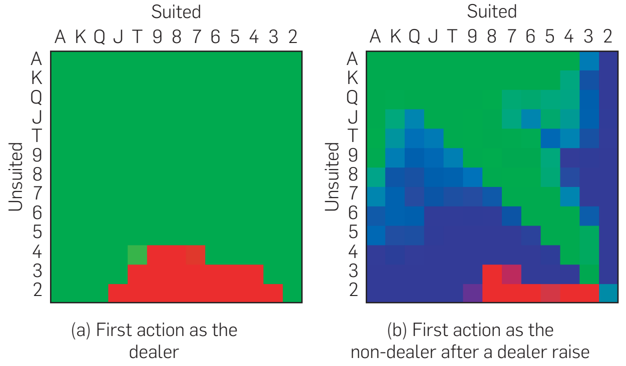


Figure 4. Action probabilities in the solution strategy for two early decisions. Each cell represents one of the possible 169 hands (i.e., two private cards) with the upper diagonal consisting of cards with the same suit and the lower diagonal consisting of cards of different suits. The color of the cell represents the action taken: red for fold, blue for call, and green for raise, with mixtures of colors representing a stochastic decision.



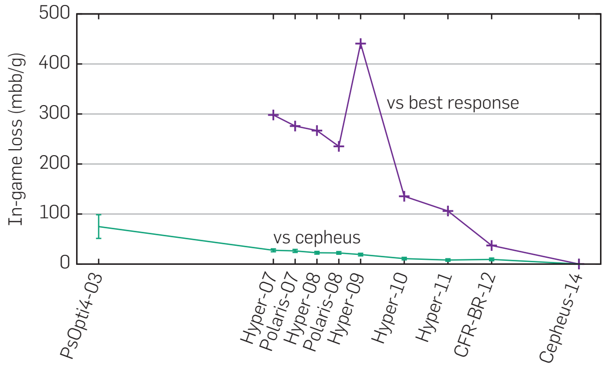
(a) First action as the dealer

(b) First action as the non-dealer after a dealer raise

Figure 5 presents the exploitability of our historical agents and their average loss in games played against Cepheus. To reduce the impact of luck, a duplicate poker format was used where each game is played twice, using the same cards, but with the players in opposite positions. PsOpti4 was the first game theoretic strategy produced for HULHE, and was also the University of Alberta entry to the 2006 ACPC.[5, h] The University of Alberta entries to the ACPC were named Hyperborean, and from 2007 onwards, all were created using variants of CFR.[i] The Polaris 2007 and 2008 agents were created by the University of Alberta for its two Man-vs.-Machine Poker Championship matches, in which Polaris narrowly lost in 2007 and narrowly won in 2008; an analysis of these matches is available in[24] [Chapter 8]. Finally, the CFR-BR agent was our closest equilibrium approximation prior to this work.[21] It used the same abstract game as Hyperborean 2011, but used an algorithm that solved for the abstract strategy with the lowest real game exploitability.

These results show that, with the exception of Hyperborean 2009, each new generation of strategies improved in both exploitability and in loss against an essentially optimal strategy. However, even though many of these strategies were highly exploitable, the rate at which they lose to Cepheus is quite low. This loss is difficult to measure with statistical confidence: a 100,000 game (non-duplicate) match would have a 95% confidence interval of 31mbb/g, larger than the performance difference between Cepheus and every agent but PsOpti4. Further, Hyperborean 2009 did improve over its predecessors in terms of in-game performance against Cepheus, and regressed in exploitability due to its use of "Strategy Grafting," an unsound solving technique that solves an abstraction as a series of fragments.[50] This technique allows for a much larger and finer grained abstraction than would otherwise be feasible, resulting in improved in-game performance, but without theoretical guarantees on exploitability. Together, these results illustrate the difficulty in evaluating a strategy only through its competition performance, instead of calculating its exploitability.

Figure 5. Exploitability and performance against Cepheus for earlier computer strategies. Results are in mbb/g, and indicate the expected winnings by the strategy's opponent (a best response or Cepheus, respectively). The Cepheus matches involved 1mn games of duplicate poker (2mn games total), except for PsOpti4 which played 20,000 duplicate games (40,000 games total).



Name	Year	Exploitability	Cepheus
PsOpti4	2003	–	74.9 ± 23.7
Hyperborean 2007	2007	298.106	27.4 ± 2.9
Polaris 2007	2007	275.880	26.2 ± 3.0
Hyperborean 2008	2008	266.797	22.5 ± 2.7
Polaris 2008	2008	235.294	22.2 ± 2.6
Hyperborean 2009	2009	440.823	18.9 ± 2.6
Hyperborean 2010	2010	135.427	10.8 ± 2.5
Hyperborean 2011	2011	106.035	8.0 ± 2.4
CFR-BR	2012	37.113	9.2 ± 2.6
Cepheus	2014	0.986	0

We can also measure Cepheus' performance against human adversaries. After this article was first published in January 2015, our website allowed visitors to play against Cepheus and inspect its strategy.[8] Each visitor chose a username and played any number of short 100-game matches against Cepheus. Over the last two years, 39,564 unique usernames have played 98,040 matches, with 3,564,094 total games played.[j] Over this set of games, Cepheus is winning at a rate of 169.9 ± 5.2mbb/g with 95% confidence. However, most of the players did not finish a single 100-game match (only 7,878 players did so, with 20,374 completed matches in total), and so this winrate is likely not reflective of Cepheus' performance against strong opponents.

Determining which of these players are strong is nontrivial because of both variance in their matches, and the unequal amount of games played by each player. While both luck and skill contribute to a player's performance, the highest-scoring players are more likely to be the luckiest rather than the strongest. Additionally, bias may be introduced if players keep playing while ahead, but quit if they are losing. In order to limit the impact of bias and evaluate Cepheus' performance against different tiers of humans, we used the following method. First, we eliminated usernames with insufficient data that had played fewer than 500 games, leaving 821 usernames playing 33,752 matches with 1,765,656 games. Next, we divided each username's games into two sets, called Rank and Test.[k] Each username's Rank games were evaluated, and the resulting winrates were used to sort the players by performance. This ordering reflected both their skill and luck. The players were then divided equally into five tiers: the bottom 20% of usernames, 21–40% etc. Within each tier, the Test game results were averaged to produce a winrate for the tier, independent from the luck that affected the Rank games.

These results are shown in Figure 6. Cepheus' estimated winrate varies from 225 to 87mbb/g as we advance through the tiers, decreasing as the quality of the human

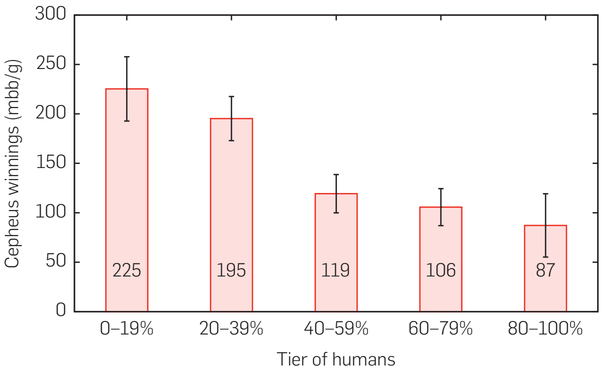
[h] PsOpti4 acts too slowly for an exploitability calculation to be practical, or for a long match against Cepheus.

[i] In the inaugural 2006 ACPC, PsOpti4 was the core component of Hyperborean 2006.

[j] Many players quit before finishing the 100-game match.

[k] In each block of four sequential games, one pair (played in each position) was assigned to each set.

Figure 6. Games played by humans and Cepheus. Humans were divided equally into five skill groups, and the column and error bar indicates the group's average loss to Cepheus in mbb/g.



players improves. Even against the top 20% tier of players in this experiment, Cepheus' winrate of 87mbb/g is higher than against any of our historical agents. It even exceeds 50mbb/g, a commonly cited benchmark for what a professional poker player seeks to win from a weaker opponent.

6. CONCLUSION

In this paper, we announced that heads-up limit Texas hold'em poker is essentially weakly solved. This is the first nontrivial imperfect information game played competitively by humans to be solved. Even still, the reader may ask what is the ultimate significance of solving poker? The breakthroughs behind this result are general algorithmic advances that make game-theoretic reasoning in large-scale models of any sort more tractable. And, while seemingly playful, game theory has always been envisioned to have serious implications, for example, its early impact on cold war politics.[31] More recently, there has been a surge in game-theoretic applications involving security, including systems being deployed for airport checkpoints, air marshall scheduling, and coast guard patrolling.[43] CFR algorithms, based on those described in this paper, have been used for robust decision-making in settings where there is no apparent adversary, with potential application to medical decision support.[15] With real life decision-making settings almost always involving uncertainty and missing information, algorithmic advances, such as those needed to solve poker, are needed to drive future applications. However, we also echo a response attributed to Alan Turing in defense of his own work in games, "It would be disingenuous of us to disguise the fact that the principal motive which prompted the work was the sheer fun of the thing."[30]

Acknowledgments

The author order is alphabetical reflecting equal contribution by the authors. The idea of CFR$^+$ and compressing the regrets and strategy originated with Oskari Tammelin.[44] This research was supported by Natural Sciences and Engineering Research Council (NSERC), Alberta Innovates Technology Futures (AITF) through the Alberta Innovates Centre for Machine Learning (AICML), and was only possible due to computing resources provided by Compute Canada and Calcul Québec. The authors would like to thank all of the current and past members of the University of Alberta Computer Poker Research Group (CPRG), where the idea to solve heads-up limit Texas hold'em was first discussed; Jonathan Schaeffer, Robert Holte, Duane Szafron, and Alex Brown for comments on early drafts of this article; and Bryce Paradis for insights into the conventional wisdom of top human poker players.

References

1. Poker: A big deal. *The Economist*, Economist Newspaper Ltd., London, December 22, 31–38, 2007.
2. Allis, V.L. *A Knowledge-Based Approach to Connect-Four. The Game is Solved: White Wins.* Master's thesis, Vrije Universiteit, Amsterdam, The Netherlands, 1988.
3. Allis, V.L. *Searching for Solutions in Games and Artificial Intelligence.* PhD thesis, Vrije Universiteit Amsterdam, The Netherlands, 1994.
4. Babbage, C. *Passages from the Life of a Philosopher.* Longman, Green, Longman, Roberts, and Green, London, 1864. Chapter 34.
5. Billings, D., Burch, N., Davidson, A., Holte, R.C., Schaeffer, J., Schauenberg, T., Szafron, D. Approximating game-theoretic optimal strategies for full-scale poker. *IJCAI*, (2003), 661–668.
6. Billings, D., Davidson, A., Schaeffer, J., Szafron, D. The challenge of poker. *Artificial Intelligence 134,* 1–2 (2002), 201–240.
7. Borel, E., Ville, J. *Applications de la théorie des probabilités aux jeux de hasard.* Gauthier-Villars, 1938.
8. Bowling, M., Burch, N., Johanson, M., Tammelin, O. The cepheus website, 2015. http://poker.srv.ualberta.ca.
9. Bowling, M., Burch, N., Johanson, M., Tammelin, O. Heads-up limit hold'em poker is solved. *Science 347,* 6218 (Jan. 2015), 145–149.
10. Bowling, M., Burch, N., Johanson, M., Tammelin, O. Heads-up limit hold'em poker is solved: Supplementary online material, January 2015.
11. Bowling, M., Johanson, M., Burch, N., Szafron, D. Strategy evaluation in extensive games with importance sampling. *ICML*, (2008), 72–79.
12. Bronowski, J. Ascent of man. Documentary, 1973. Episode 13.
13. Buro, M. Solving the oshi-zumo game. *Adv. in Comp. Games 135,* (2004) 361–366.
14. Campbell, M., Hoane, A.J. Jr., Hsu, F. Deep blue. *Artificial Intelligence 134,* (Jan. 2002), 57–83.
15. Chen, k., Bowling, M. Tractable objectives for robust policy optimization. *Adv. Neural Inf. Process. Syst. (NIPS) 25,* (2012), 2078–2086.
16. Craig, M. *The Professor, the Banker, and the Suicide King: Inside the Richest Poker Game of All Time.* Grand Central Publishing, New York, NY, 2006.
17. Ferrucci, D. Introduction to "this is watson." *IBM J. Res. Dev 56,* 3.4 (May 2012) 1:1–1:15.
18. Gilpin, A., Hoda, S., Peña, J., Sandholm, T. Gradient-based algorithms for finding nash equilibria in extensive form games. *WINE*, (2007), 57–69.
19. Gilpin, A., Sandholm, T. Lossless abstraction of imperfect information games. *J. ACM 54,* 5 (2007).
20. Jackson, E. Slumbot: An implementation of counterfactual regret minimization on commodity hardware. In *Proceedings of the 2012 Computer Poker Symposium.* (2012).
21. Johanson, M., Bard, N. Burch, N., Bowling, M. Finding optimal abstract strategies in extensive form games. *AAAI*, (2012), 1371–1379.
22. Johanson, M., Bard, N., Lanctot, M., Gibson, R., Bowling, M. Efficient Nash equilibrium approximation through Monte Carlo counterfactual regret minimization. *AAMAS* (2012).
23. Johanson, M., Waugh, K., Bowling, M., Zinkevich, M. Accelerating best response calculation in large extensive games. *IJCAI* (2011), 258–265.
24. Johanson, M.B. *Robust Strategies and Counter-Strategies: From Superhuman to Optimal Play.* PhD thesis, University of Alberta, Edmonton, Alberta, Canada, 2016.
25. Karmarkar, N. A new polynomial-time algorithm for linear programming. In *Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing* (1984), ACM, New York, NY, 302–311.
26. Koller, D., Megiddo, N. The complexity of two-person zero-sum games in extensive form. *Games Econ. Behav 4,* 4 (1992), 528–552.
27. Koller, D., Megiddo, N., von Stengel, B. Efficient computation of equilibria for extensive two-person games. *Games Econ. Behav 14,* 2 (1996).
28. Koller, D., Pfeffer, A. Representations and solutions for game-theoretic problems. *Artificial Intelligence 94,* (1997), 167–215.
29. Kuhn, H. Simplified two-person poker. In *Contributions to the Theory of Games*, volume 1 of *Annals of mathematics studies.* H. Kuhn and A. Tucker, eds. Princeton University Press, Princeton, New Jersey, 1950, 97–103.
30. Mirowski, P. What were von neumann and morgenstern trying to accomplish? In *Toward a History of Game Theory.* Weintraub, ed. Duke University Press, 1992, 113–147. Mirowski cites Turing as author of the paragraph containing this remark. The paragraph appeared in [46], in a chapter with Turing listed as one of three contributors. Which parts of the chapter are the work of which contributor, particularly the introductory material containing this quote, is not made explicit.
31. Morgenstern, O. The cold war is cold poker. *N. Y. Times Mag.* (Feb. 5 1961) pages 21–22.
32. Nash, J.F., Shapley, L.S. A simple 3-person poker game. In *Contributions to the Theory of Games I.* Princeton University Press, Princeton, New Jersey, 1950, 105–116.

33. Nesterov, Y. Excessive gap technique in nonsmooth convex minimization. *SIAM Journal on Optimization 16*, 1 (2005), 233–249.
34. Rehmeyer, J., Fox, N., Rico, R. Ante up, human: The adventures of polaris the poker-playing robot. *Wired 16*, 12 (Dec. 2008), 186–191.
35. Romanovskii, I.V. Reduction of a game with complete memory to a matrix game. *Soviet Mathematics 3*, (1962), 678–681.
36. Rubin, J., Watson, I. Computer poker: a review. *Artificial Intelligence 175*,(2011), 958–987.
37. Sandholm, T. The state of solving large incomplete-information games, and application to poker. *AI Mag. 31*, 4 (2010), 13–32.
38. Schaeffer, J., Lake, R., Lu, P., Bryant, M. Chinook the world man-machine checkers champion. *AI Mag. 17*, 1 (1996), 21–29.
39. Schaeffer, J., Neil Burch, Y.B., Kishimoto, A., Müller, M., Lake, R., Lu, P. Sutphen, S. Checkers is solved. *Science 317*, 5844 (2007), 1518–1522.
40. Shannon, C.E. Programming a computer for playing chess. *Philosophical Magazine, Series 7, 41*, 314 (Mar. 1950), 256–275.
41. Shi, J., Littman, M.L. Abstraction methods for game theoretic poker. In *Comp. Games*, (2000), 333–345.
42. Southey, F., Bowling, M., Larson, B., Piccione, C., Burch, N., Billings, D., Rayner, D.C. Bayes' bluff: Opponent modelling in poker. *UAI*, (2005) 550–558.
43. Tambe, M. *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press, Cambridge, England, 2011.
44. Tammelin, O. Cfr+. *CoRR*, abs/1407.5042, 2014.
45. Tammelin, O., Burch, N., Johanson, M., Bowling, M. Solving heads-up limit texas hold'em. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, 2015, 645–652.
46. Turing, A. Digital computers applied to games. In *Faster Than Thought*. B.V. Bowden, ed. Chapter 25. Pitman, 1976.
47. von Neumann, J. Zur theorie der gesellschaftsspiele. *Mathematische Annalen 100*, 1 (1928), 295–320.
48. von Neumann, J., Morgenstern, O. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, Second Edition, 1947.
49. Waugh, K., Bagnell, J.A. A unified view of large-scale zero-sum equilibrium computation. In *AAAI Workshop on Computer Poker and Imperfect Information*, 2015.
50. Waugh, K., Bard, N., Bowling, M. Strategy grafting in extensive games. In *Advances in Neural Information Processing Systems 22 (NIPS-09)*, 2009. http://webdocs.cs.ualberta.ca/~games/poker/publications/NIPS09-graft.pdf.
51. Zermelo, E. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels. In *Proceedings of the Fifth International Congress Mathematics*. Cambridge University Press, Cambridge, 1913, 501–504.
52. Zinkevich, M., Johanson, M., Bowling, M., Piccione, C. Regret minimization in games with incomplete information. *NIPS* (2008), 905–912.
53. Zinkevich, M., Littman, M. The AAAI computer poker competition. *J. Inter. Comp. Games Association 29*, (2006), News item.

Michael Bowling (bowling@cs.ualberta.ca), Department of Computing, Science, University of Alberta, Edmonton, Alberta, Canada.

Neil Burch and Michael Johanson ({nburch, johanson}@ualberta.ca), Department of Computing, Science, University of Alberta, Edmonton, Alberta, Canada.

Oskari Tammelin ({ot@iki.fi} {http://jeskola.net}).

Foolproof
and Other Mathematical Meditations
Brian Hayes
The Creative Partnership of Humans and Technology
PLATO AND THE NERD
EDWARD ASHFORD LEE
ROBOT SEX
SOCIAL AND ETHICAL IMPLICATIONS
EDITED BY JOHN DANAHER AND NEIL McARTHUR