

EC-16 Tutorial on Computer Poker

Sam Ganzfried and Marc Lanctot

Poker has been studied academically since the founding of game theory and in fact may have even been the inspiration for the field: the only motivating problem described in John Nash’s PhD thesis, which defined and proved existence of the central solution concept, was actually a three-player poker game. Such interest has not waned over time. Last year when a computer program developed at Carnegie Mellon University competed against the strongest human two-player no-limit Texas hold ’em players in the inaugural Brains vs. Artificial Intelligence competition, thousands of poker fans followed with excitement. The entire event was streamed live, and a thread on the most popular poker forum dedicated to the event received over 230,000 views.¹ Earlier that year another monumental breakthrough was attained, as the two-player limit variation of Texas hold ’em (which is one of the smallest variants that is played competitively by humans) was “essentially weakly solved” (i.e., an ϵ -Nash equilibrium was computed for such a small ϵ to be statistically indistinguishable from zero in a human lifetime of played games) by researchers at the University of Alberta. This result was published in the journal *Science*. Poker, and particularly Texas hold ’em, is tremendously popular for humans, and online poker is a multi-billion dollar industry. Computer poker has proved to be one of the most visible applications of research in computational game theory.

In theory, there exists an “optimal” strategy for any two-player zero-sum game, that would guarantee being unbeatable against all opposing agents. This result is due to the Minimax Theorem, one of the fundamental results in game theory, and the “optimal” strategy is called a *Nash equilibrium*. If we were able to compute a Nash equilibrium for two-player no-limit Texas hold ’em, then we would guarantee that against *any* opponent we would either win or tie (in expectation). From a complexity-theoretic perspective, computing this strategy is easy; there exists a polynomial-time algorithm based on a linear programming formulation. However, this algorithm only scales to games with 10^8 states. More recently algorithms have been developed for approximating equilibrium strategies (they converge to equilibrium in the limit) that scale to 10^{15} states. However, even this is a far cry from 10^{165} , the size of the version of two-player no-limit Texas hold ’em played in the AAAI Annual Computer Poker Competition. In addition to scalable algorithms for Nash equilibrium computation, approximating Nash equilibrium strategies in a domain of that magnitude requires approaches for approximating the full 10^{165} game tree with a significantly smaller game of 10^{15} states that retains much of the strategic structure of the original game (aka *automated abstraction algorithms*), approaches for interpreting actions for the opponent that have been removed from the abstraction (aka *action translation mappings*), and additional approaches for extrapolating the equilibrium strategies from the abstract game to the full game (aka *postprocessing techniques*).

The tutorial will cover practical and theoretical aspects of all of the components of the strongest poker agents, including detailed description of the architecture of recent competition champions. While some of these approaches are heuristic, many of them have interesting theoretical aspects and open questions. Unlike in single-agent optimization, it has been shown that abstraction in games can suffer from theoretical pathologies; for instance, solving a finer-grained abstraction can actually increase exploitability. The main abstraction algorithms that have been successful in practice are heuristic and have no theoretical guarantees (it is extremely difficult to prove meaningful theoretical guarantees when approximating a game with 10^{165} states by one with 10^{15} states). Recent work has presented an abstraction algorithm with bounds on the solution quality; however, it only scales to a tiny poker game with a five-card deck. It would be very interesting to bridge this gap between heuristics that work well in practice for large games with no theoretical guarantees, and the approaches with guarantees that have modest scalability. There are also many theoretical questions related to the action translation and post-processing approaches. Another important component

¹<http://forumserver.twoplustwo.com/29/news-views-gossip-sponsored-online-poker-report/wcgrider-dong-kim-jason-les-bjorn-li-play-against-new-hu-bot-1526750/>

of strong poker agents that has been actively studied is called “endgame solving,” where the portion of the game we have reached is solved in real-time to a higher-degree of accuracy. Theoretically this approach can lead to high exploitability, though experimentally it consistently improves performance, and the best human two-player no-limit player said it was the strongest part of the competition agent. We will describe counterfactual regret (CFR) minimization, the most popular equilibrium-finding algorithm for the strongest recent poker agents, and present convergence guarantees of several common variants.

The Minimax Theorem does not apply to games that are not zero sum or have more than two agents. These games can have many equilibria, each assigning different payoffs to the agents; if the opponents do not follow the equilibrium strategy that we have computed, then we can perform arbitrarily poorly. Furthermore, computing a Nash equilibrium in these game classes is challenging computationally (it is PPAD-complete and widely conjectured that no efficient algorithms exist). Despite this worst-case hardness result, several of the techniques described are applicable to these games (though in some cases theoretical guarantees are more limited), and there has already been initial progress in practical algorithms.

Even in two-player zero-sum games, the Nash equilibrium is not the end of the story (even if we are able to compute one exactly without requiring approximation). Against suboptimal opponents who are not playing an equilibrium strategy, we can often obtain a significantly higher payoff than the value of the game by learning to exploit their mistakes as opposed to following a static equilibrium strategy. In addition to the work on approximating Nash equilibrium, there has been significant recent work on opponent modeling and exploitation, which will also be presented in the tutorial. This includes robust approaches for exploiting opponents in large imperfect-information games as well as approaches that exploitatively deviate from stage-game equilibrium while also theoretically guaranteeing safety in the worst-case.

1 Biographies

Sam Ganzfried received a PhD in computer science from Carnegie Mellon University in 2015 for his dissertation “Computing Strong Game-Theoretic Strategies and Exploiting Suboptimal Opponents in Large Games” and holds an A.B. in math from Harvard. His research interests include artificial intelligence, game theory, multiagent systems, multiagent learning, large-scale optimization, large-scale data analysis and analytics, and knowledge representation. He created two-player no-limit Texas hold em agent Claudico that competed in the inaugural 2015 Brains vs. Artificial Intelligence competition against the strongest human specialists in the world (it lost by an amount that was not statistically significant at the 95% confidence level). He also created two-player no-limit Texas hold em agent Tartanian7 that won the 2014 Annual Computer Poker Competition, beating each opposing agent with statistical significance. He organized the AAAI Workshop on Computer Poker and Imperfect Information in 2014 and 2015.

Marc Lanctot received a PhD in Computer Science from University of Alberta in 2013 for his dissertation “Monte Carlo Sampling and Regret Minimization for Equilibrium Computation and Decision-Making in Large Extensive Form Games,” and a BSc and MSc in Computer Science from McGill University. He is interested in artificial intelligence, machine learning, multiagent systems, and computational game theory. He has been involved with development of regret minimization approaches to equilibrium-finding in Poker, particularly the sampling variants. His post-doctoral research at Maastricht University focused on foundations of Monte Carlo tree search, and he proposed new consistent search algorithms for simultaneous move and imperfect information games. He is now at Google DeepMind working on deep reinforcement learning.

2 Prior events

There have been several related workshops at AAAI (e.g., http://www.cs.cmu.edu/~sganzfri/AAAI15_Workshop.html).