Cepheus and Computer Poker Algorithms

Bill Chen, SIG

SIG SUSQUEHANNA™

Certain Disclosures and Disclaimers

- I am an employee of Susquehanna International Group, LLP (together with its affiliated and related entities, "SIG"). However, the views expressed today are my own and do not necessarily reflect the views of SIG. SIG expressly disclaims any liability in connection with the use of this document or its contents by any third party.

- Information presented in this document is for informational, educational and illustrative purposes only. While the information in this document is from sources believed to be reliable, no representations or warranties, express or implied, as to whether the information is accurate or complete are given.

SIG™
SUSQUEHANNA

Outline of Talk

- Explanation of what Cepheus accomplished, and discussions of Game Theory Optimal (GTO) Strategies in Poker,
- Explanation of Regret minimization and Counterfactual Regret (CFR)
- Improvements to CFR, CFR+
- Extension of Computer Solutions to other games, including big bet games and multi-player games

SIG™ SUSQUEHANNA

What did Cepheus Accomplish?

- Cepheus is a Game Theory Optimal (GTO) solution to Heads-up Limit Holdem.
- After 900 CPU-years they have achieved an exploitability of less than 1/1000 of a big blind. This is virtually indistinguishable from a perfect strategy.
- It's definitely a milestone, this is the first time a real poker game has been solved, however given the previous work of Bowling, Burch, Johanson, and Tammelin it was just a matter of time (and lots of CPU).
- What effect does this have on other games? We will visit this later.

SIG™ SUSQUEHANNA

Nash Equilibrium

John F. Nash, Nobel Prize 1994 for “pioneering analysis of equilibria in the theory of non-cooperative games.”

Nash’s work extended earlier idea of John Von Neumann and Oskar Morgenstern.

A (Nash) Equilibrium is a set of strategies:

One strategy for each player such that no player has an incentive to unilaterally change his strategy.

In two player Zero-Sum games, we also refer to Nash Equilibria as *Game Theory Optimal (GTO).*

SIG™ SUSQUEHANNA

Game Theory Poker Example

- The following game is played with Rose and Colin. Each player antes $50 for a $100 pot. Rose looks at a card from a full deck. Rose will win the pot at showdown if the card is a Spade, otherwise she will lose.
- Rose can decide to bet $100 or check.
- If Rose bets, Colin may decide to call $100 or fold. If Colin folds, Rose wins, if Colin calls there is a showdown for the $300 pot.
- What are the best strategies for Rose and Colin?

SIG™ SUSQUEHANNA

Game Theory Example—Poker

- Rose should always bet a spade.
- If Colin calls 100% of the time, Rose will just never bluff (bet when she doesn't have a spade) and net $150 each time she has a spade and -$50 when she does not for a total net gain of zero.
- If Colin never calls , Rose will just bet every time and net $50.
- If Colin calls half the time, Rose will be indifferent to bluffing, she will net -$50 either way without a spade, and $100 with a spade for a net of -$12.50. This is the correct strategy for Colin
- Conversely, if Rose's ratio's of bluffs to spade bets (these are called value bets) is 1:2 then Colin is indifferent to calling. So Rose should bluff on half of the hearts (or similar frequency).
- These are the Nash Equilibrium, and GTO strategies.

SIG™ SUSQUEHANNA

Game Theory Optimal

The set up is that there is a game value function $u(\sigma_x, \sigma_y)$ which takes as arguments a strategy from X and a strategy from Y.

We can take convex linear combinations of strategies, that is if $\sigma_x^k \epsilon X$ and $a_k \geq 0$, with $\sum a_k = 1$ then we can take $\sigma_x = \sum a_k \sigma_x^k$ and also

$$u(\sigma_x, \sigma_y) = \quad a_k u(\sigma_x^K, \sigma_y)$$

We define a pair of strategies σ_x^*, σ_y^* o be in ϵ equilibrium if:

$$\max_{\sigma_x} \; u(\sigma_x, \sigma_y^*) - \min_{\sigma_y} \; u(\sigma_x^*, \sigma_y) \leq \epsilon$$

SIG SUSQUEHANNA™

Regret Minimization

Suppose at each time step *t,* the player has an option of a linear combination of n pure strategies:

$$\sigma^t = \sum_{k=1}^{n} a_k^t \sigma_k$$

Where $\sum_{k=1}^{n} a_k^t = 1 \; and \; a_k^t \geq 0$.

Now at each time step *t* we are given values $u^t\langle\sigma_k\rangle$ perhaps by an adversary. We define the equity of the play as:

$$u^t(\sigma^t) = \sum_{k=1}^{n} a_k^t u^t(\sigma_k)$$

The idea is that the adversary can choose strategy *k* to score well sometimes and badly at other times.

SIG SUSQUEHANNA™

8

Regret Minimization

We now define the total regret for strategy *k:*

$$R_k^T = \sum_{t=1}^{T} u^t(\sigma_k) - u^t(\sigma^t)$$

Note this can be negative or positive, but the idea is that we want the average positive regret to go to zero that is $R_k^T/T < \epsilon^t$ where ϵ^t is a sequence converging to zero. The point is that we can do this if we use the algorithm called regret matching that is if we define $R_k^{T+} = \max(R_k^T, 0)$, and we set

$$a_k^T = \frac{R_k^{T-1+}}{\sum_{k=1}^{n} R_k^{T-1+}}$$

(or a random strategy if all regrets are negative or zero).

SIG SUSQUEHANNA™

Regret minimization example.

Suppose we have n=2 strategies and at each step one of the strategies has equity 1 and the other strategy has equity 0.

1. At t=1 we pick $\sigma^t = \sigma_1$. We are given $u^t(\sigma_1) = 0$ and $u^t(\sigma_2) = 1$. Thus we have $R_1^t = 0$ and $R_2^t = 1$.
2. At t=2 we pick $\sigma^t = \sigma_2$. We are given $u^t(\sigma_1) = 1$ and $u^t(\sigma_2) = 0$. Now we have $R_1^t = 1$ and $R_2^t = 1$.
3. At t=3 we pick $\sigma^t = \sigma_1/2 + \sigma_2/2$. We are given $u^t(\sigma_1) = 0$ and $u^t(\sigma_2) = 1$. Now we have $R_1^t = 0.5$ and $R_2^t = 1.5$
4. At t=4 we pick $\sigma^t = \sigma_1/4 + 3\sigma_2/4$. We are given $u^t(\sigma_1) = 0$ and $u^t(\sigma_2) = 1$. Now we have $R_1^t = -0.25$ and $R_2^t = 1.75$
5. At t=5 we pick $\sigma^t = \sigma_2$.

SIG SUSQUEHANNA™

Regret Minimization

We actually have a quadratic bound in our example $(R_1^{T+})^2 + (R_2^{T+})^2 \leq T$

If both regrets are positive:

$$(R_1^{T+1})^2 + (R_2^{T+1})^2 = \left(R_1^T \pm \frac{R_2^T}{R_1^T + R_2^T}\right)^2 + \left(R_2^T \mp \frac{R_1^T}{R_1^T + R_2^T}\right)^2$$

$$\leq (R_1^T)^2 + (R_2^T)^2 + 1$$

This means that $R_k^T/T \leq 1/\sqrt{T}$.

In fact this is the general bound, we have

$$\frac{R_k^T}{T} \leq \frac{(n-1)}{\sqrt{T}}\Delta$$

where Δ is the maximum deviation of each play.

SIG SUSQUEHANNA™

Regret minimization and GTO

Now what does this have to do with GTO strategies? Suppose the pure strategies for X are $\{\sigma_{x,k}\}_{k=1}^{m}$ and for Y are $\{\sigma_{y,k}\}_{k=1}^{n}$ then if we regret-match using the equity functions $u_x^t(\cdot) = \mu(\cdot, \sigma_y^t)$ and $u_y^t(\cdot) = -\mu(\sigma_x^t, \cdot)$ then if we let $\bar{\sigma}_x^T = \sum_{t=1}^{T} \sigma_x^t/T$ and $\bar{\sigma}_y^T = \sum_{t=1}^{T} \sigma_y^t/T$ then we know that

$$u\left(\sigma_{x,k}, \bar{\sigma}_y^T\right) - u\left(\bar{\sigma}_x^T, \sigma_{y,j}\right)$$

$$= u\left(\sigma_{x,k}, \bar{\sigma}_y^T\right) - \frac{1}{T}\sum_{t=1}^{T} u(\sigma_x^T, \sigma_y^T) + \frac{1}{T}\sum_{t=1}^{T} u(\sigma_x^T, \sigma_y^T) - u\left(\bar{\sigma}_x^T, \sigma_{y,j}\right)$$

$$= R_{x,k}^T/T + R_{y,j}^T/T < 2\epsilon^t$$

We know that the strategies are within $2\epsilon^t$ equilibrium of GTO .

SIG SUSQUEHANNA™

Counterfactual Regret (CFR)

This is the major innovation of the work. First notice that the results will converge if we just have an unbiased sample of *u* instead of the actual value of *u.* That is because the sampled regrets will converge to the true regrets by the Central Limit Theorem.

The main idea is to create a personal tree from the information set of each player, that is cards he was dealt, the community cards, and the actions of each player. In this personal tree, we use regret matching at each node where the player has a decision. It's called counterfactual regret because the weighting is given assuming the player plays to that node, so weights are given by the probability of each community card and the probability of the opponents actions given σ_y^t.

This means the hands can be sampled intrinsically, by playing out a hand, or a few variations of a hands.

SIG™ SUSQUEHANNA

CFR Modifications

- In the final algorithm CFR+, which floored regrets at 0 instead of allowing really negative regrets and used more of an exhaustive search along the tree instead of Monte Carlo
- Branches can be weighted so that the ones with higher average regret can be visited more often.
- There is a lot of flexibility in weighting things in general. For example one could weight later iterations more heavily. On the river, a weighting scheme could help with the fact that call regret could be the whole pot while fold regret could only be one bet.

SIG™ SUSQUEHANNA

Limit Holdem

- Let's try to figure out how big the trees are. Let's concentrate on the river nodes.
- If we assume a 4 bet Cap, there are 9 possible actions on each bettng round (Preflop, Flop , Turn) that gets us to the next street.
- Using symmetries on the Flop we have C(13,3)=286 flops with one suit, 13*C(13,2) = 1014 with two suits, and C(15,3) = 455 three suited flops = 1755 flops.
- There are 49*48 = 2352 turns and rivers, making 1755*9*9*9*2352 = 3 billion possible action sequences to the river.
- There are 47*46 = 2126 hand types on the river, making 6.5x10^12 hand-river types. Each river node should be visited around 1000 times. It's a big computational problem but still tractable, and they used many shortcuts.

SIG SUSQUEHANNA™

Solving other poker games

- Omaha 8—Same structure at LHE, except more hand types. There are C(47,4) hand types instead of C(47,2). That's an 82.5x to the original tree. However there could be efficiencies due to bucketing hand types. That is if X is the space of possible strategies we only consider a subspace $X'CX$ and $Y'CY$ and solve the subgame u(X',Y'). The Alberta group used these methods originally to develop strong LHE programs.
- Razz—definitely the simplest variant of Stud. Only 13 card types, ranks do not matter. Unfortunately there are 13^8 possible ways the upcards can come, and 9^4 action sequences, since there is one more street and still C(15,3)=455 river hand types. That's 2.44x10^15 river-hands, a factor of 374 over Holdem. Although much simplification can probably be made. There are really 78 different games—the A vs 2 game, etc. Also because of the nature of the game, some streets could be trivial.

SIG™ SUSQUEHANNA

Big Bet Games

- The problem with No Limit and Pot Limit games is that there is a continuum of bet sizes.
- In fact there is a continuum of responses to each bet, and then there could be further streets etc. The problem is that one needs some type of interpolation between bet sizes.
- Even if some bet sizes are non-optimal a full strategy needs responses to the bets.
- Simple approximations like the rigid Pot Limit Game where the only bet size is the pot may reveal something interesting.
- Perhaps small stack games are amenable to solutions since many of the actions terminate.

SIG™ SUSQUEHANNA

Multiplayer Games 3+ players.

- Addressed in “Using Counterfactual Regret to Create Competitive Multiplayer Agents”
- Programs were 1st and 2nd in an annual 3-player LHE event
- One problem is there is no guarantee of epsilon-convergence to a Nash equilibrium.
- The bigger problem is that there could be multiple equilibria for multiway games, especially in tournaments/satellites with payouts for other places than first.
- Problems like the Rock-Maniac game in the Math of Poker where players can use a simple strategy and ensure you losing
- This is a rich area of study.

SIG™ SUSQUEHANNA

References

[1] Bowling, M., Birch, N., Johanson, M,, Tammelin, O., 2014. Heads up limit poker is Solved

[2] Risk, N., Szafron, D. 2011. Using Counterfactual Regret Minimization to Create Competitive Multiplayer Poker Agents

[3] Zinkevich, M., Johanson, M., Bowling, M. , Piccione, C. 2008. Regret Minimization in Games with Incomplete Information

SIG™ SUSQUEHANNA

15.S50 Poker Theory and Analytics

January IAP 2015