DeepStack

Expert-Level Artificial Intelligence in Heads-Up No-Limit Poker

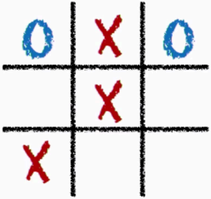
Lasse Becker-Czarnetzki

University of Heidelberg
Artificial Intelligence for Games
SS 2019

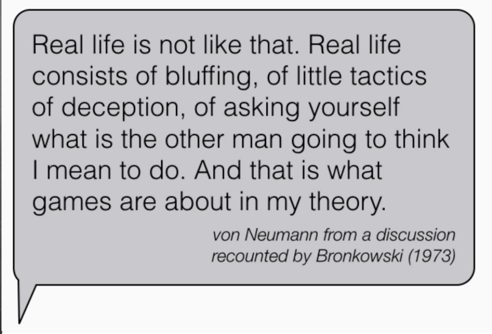
July 11. 2019

1. Perfect vs Imperfect information Games
   - Introduction
   - No-Limit Heads-up Texax Holdem
   - Perfect Information strategies
2. DeepStack
   - Re-solving (CFR)
   - Depth limited search
   - Counterfactual Value Networks
   - Sparse lookahead trees
3. Evaluation
   - Performanve against humans
   - Exploitability (LBR)
   - Nice features
4. Conclusion

Perfect information games



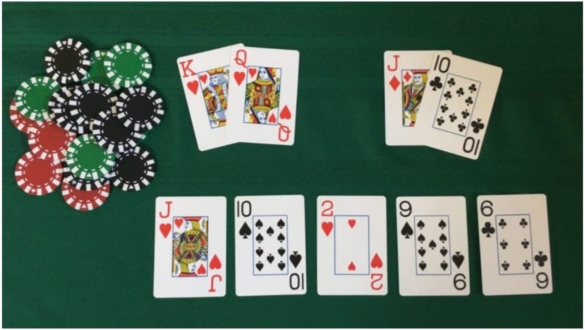
Von Neuman on games



No-Limit Heads-up Texax Holdem

- 2 player zero-sum game
- 4 Betting rounds on "who has the better cards"
- 2 Hold cards (private) (3, 4, 5) public cards.

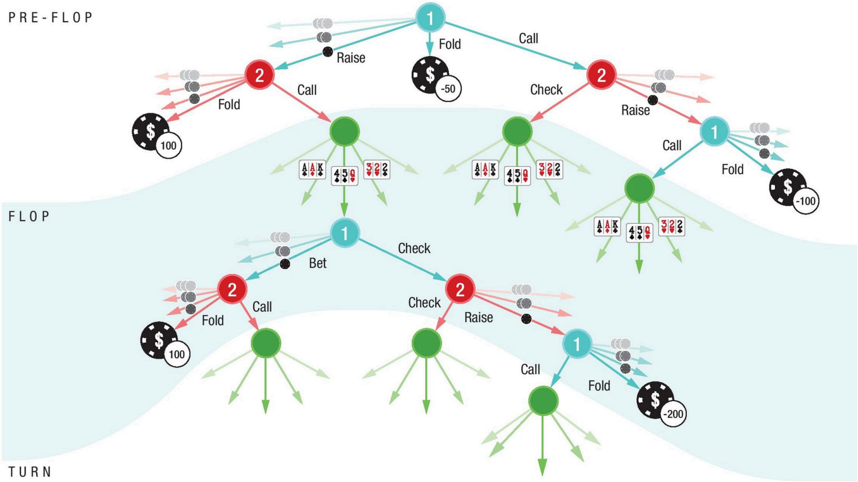
$-> \mathbf{10}^{160} \textit{decisionpoints}$



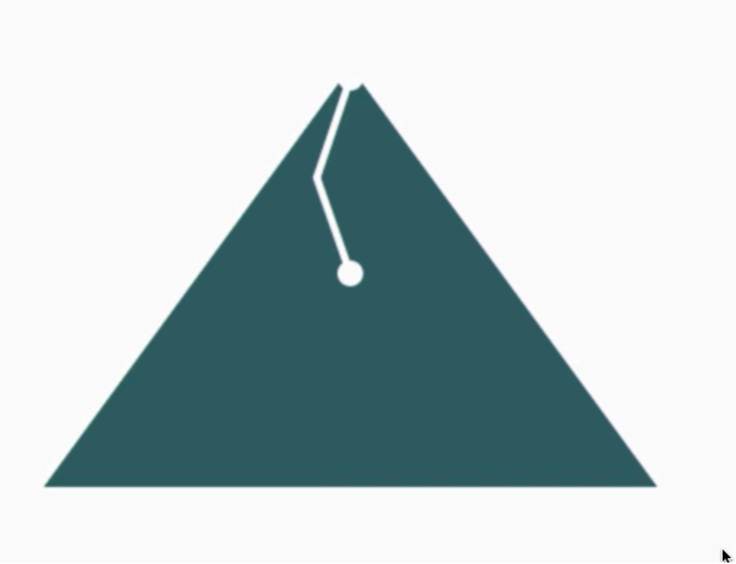
Poker Terms

- Bigblind
- Fold
- Check
- Call
- Bet (raise)
- Flop (Pre-Flop)
- Turn
- River
- range

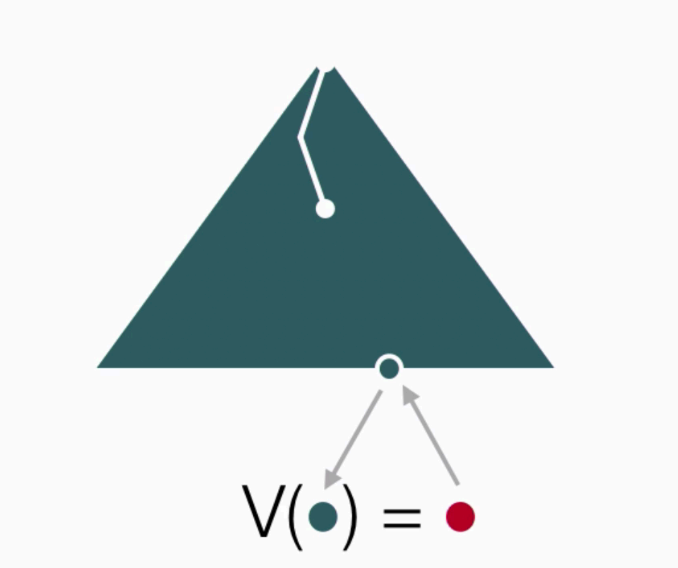
Poker Game Tree



Perfect information game

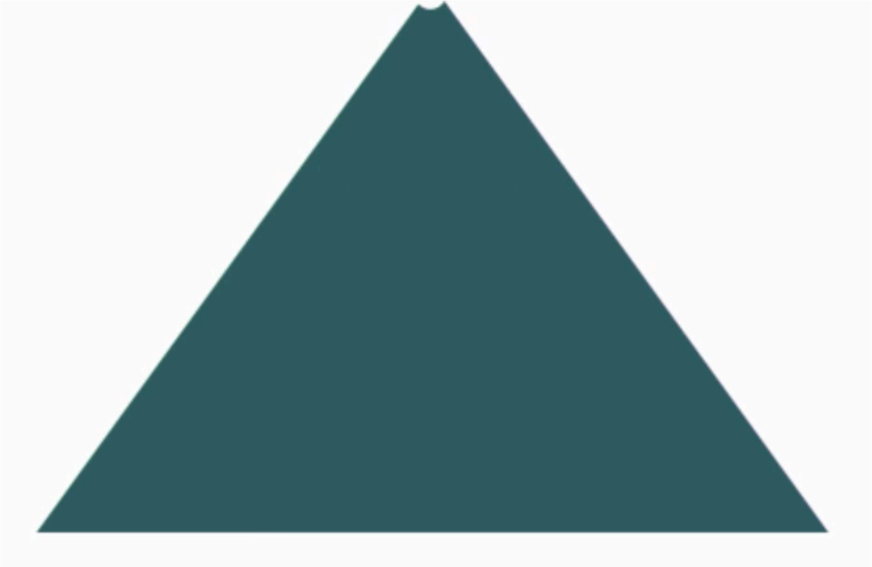


Perfect information game

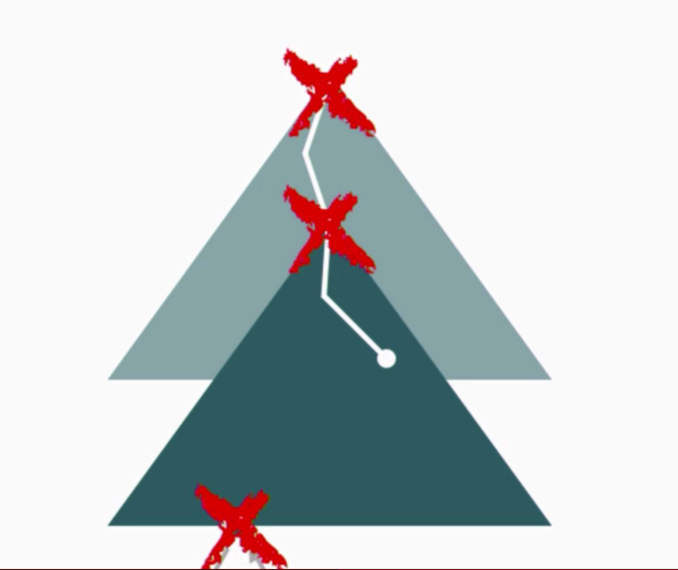


Perfect information game

Perfect information game



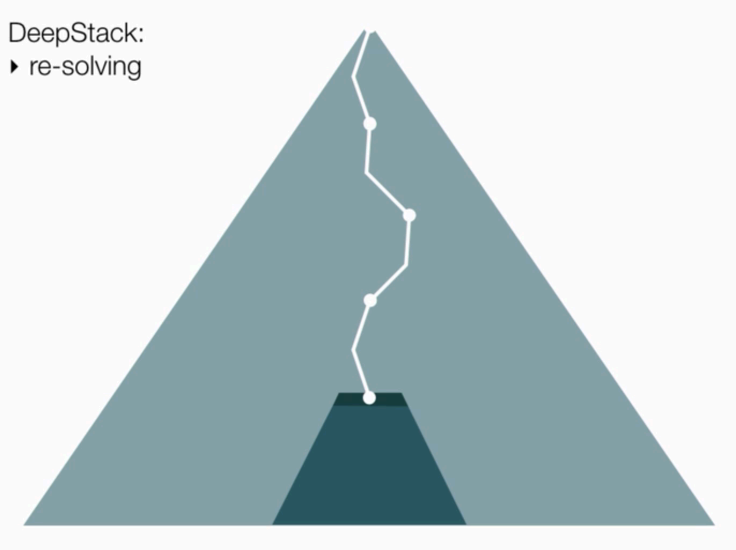
Problems for imperfect information games



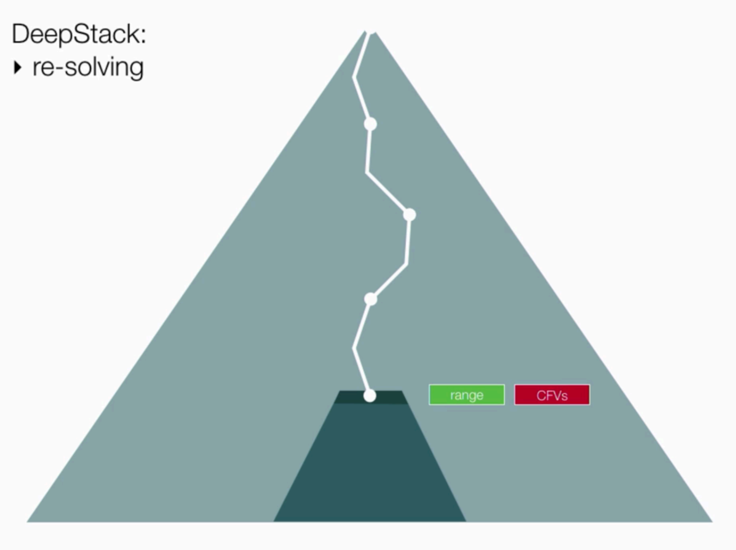
Questions

- How can we forget supergames without using necessary information?
- How do we solve a subgame when their are no definite states to start from?
- How do we evaluate a state, when we can't use a single value to summarize a position?

Re-solving



Re-solving

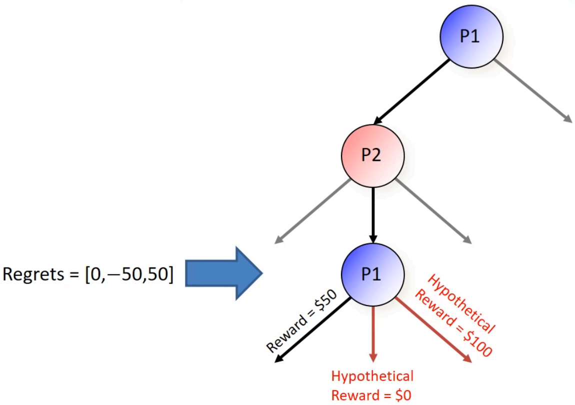


Re-solving

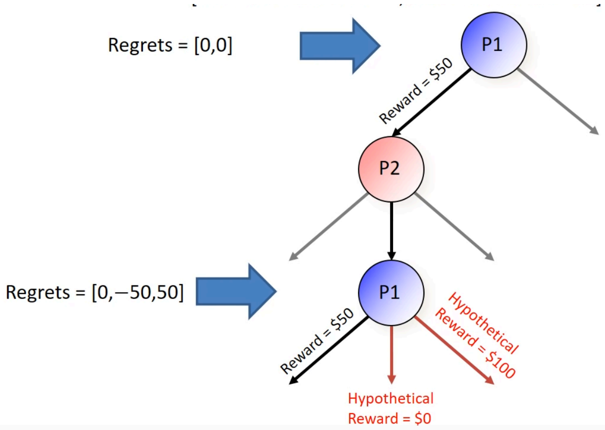
Counterfactual Regret Minimization

- **Counterfactual:** "If i had known"...
- **Regret:** "how much better would i have done if i did something else instead?
- **Minimization:** "what strategy minimizes my overall regret?
- Average strategy over i iterations = approximation to Nash Equilibrium

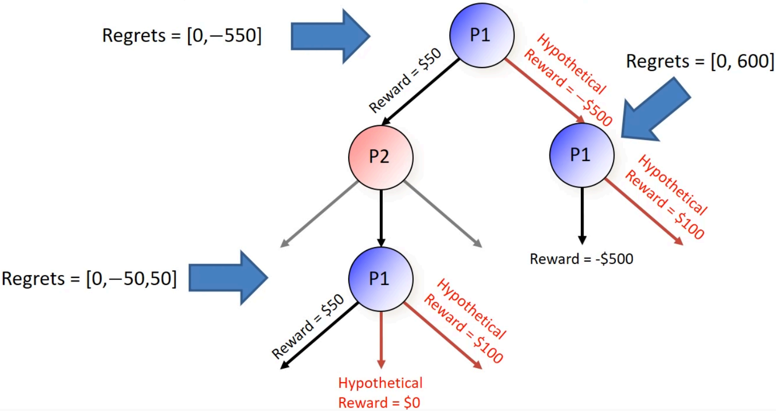
Counterfactual Regret Minimization



Counterfactual Regret Minimization



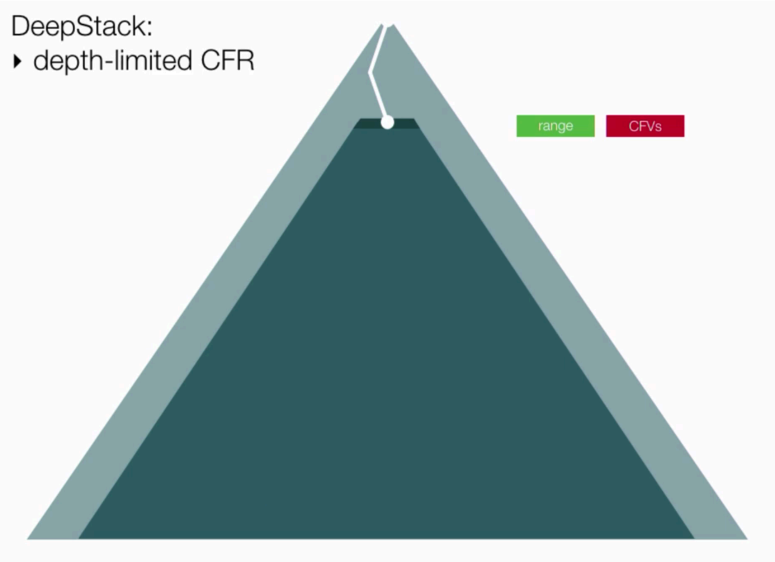
Counterfactual Regret Minimization



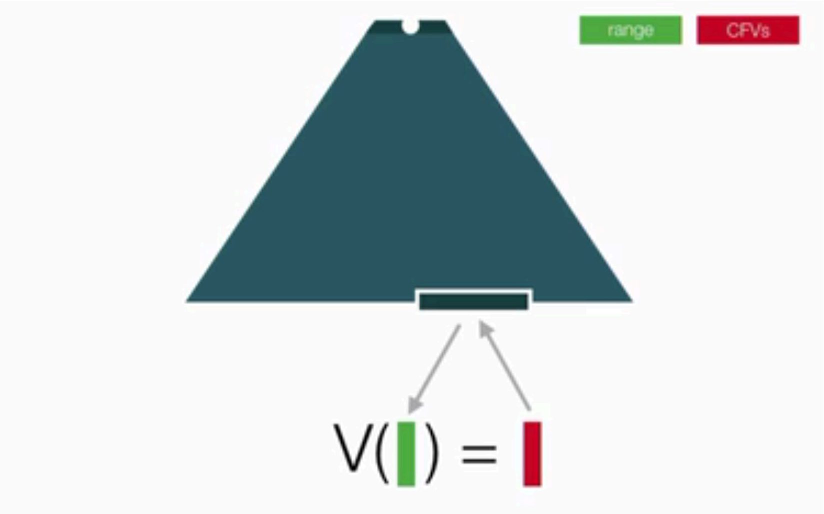
Continual Re-solving

- At every action we re-solve the subgame
- We need our range and opponents counterfactual value
  "What-if" (expected value) opponent reaches public state with hand x.
- 3 scenarios for updating range and CFVs.
  - **own action:** CFVs = CFVs(action) – Update range via Bayes rule
  - **Chance action:** CFVs = CFVs(chance action) – Eliminate impossible card combos.
  - **Opponents action:** Do Nothing

Depth limited search



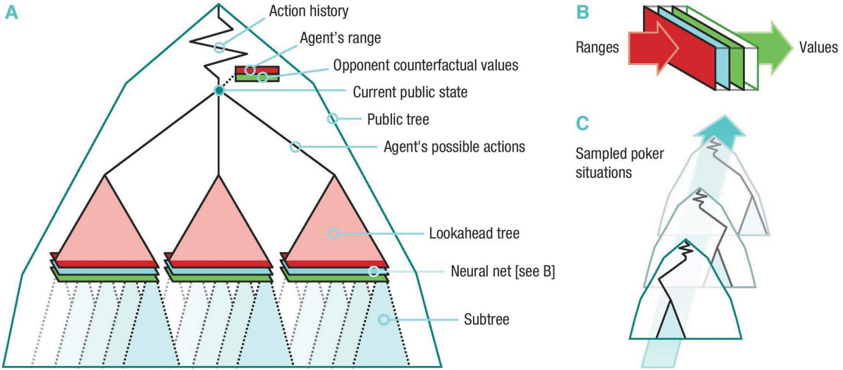
Depth limited search



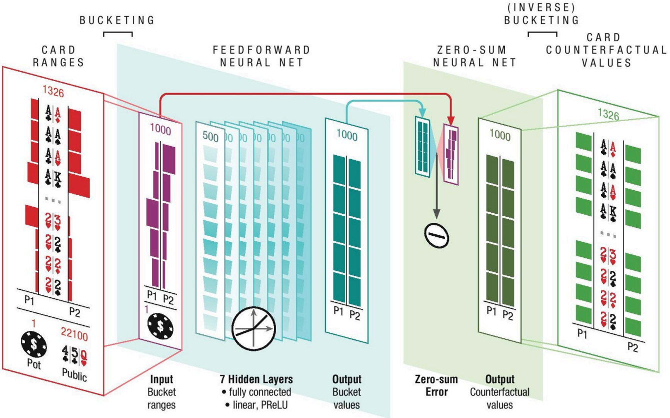
Solutions

- Search from a set of possible states, re-solving multiple times.
- Remember players range and opponents counterfactual values
- Get evaluation through Deep Counterfactual value networks

DeepStack elements summary



Deep Counterfactual Value Networks



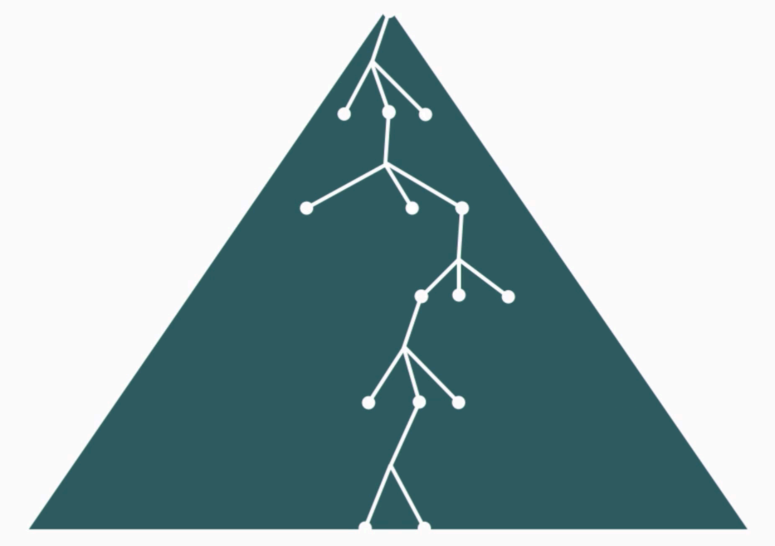
Deep Counterfactual Value Networks

- 2 Networks: Flop Network, Turn Network
- Auxiliary network (Pre-Flop)
- Simple FFNN (7 layers, 500 Nodes, ReLU)
- outer network to fit values for zero-sum game
- **input:** Pot sizes, public cards, players ranges
- **output:** Counterfactual Values (Players, Hands)

Training

- Randomly generated Poker situations.
- Turn network: 10M, Flop network:1M
- Turn network used for depth-limited lookahead in Flop Network training.

Sparse lookahead trees



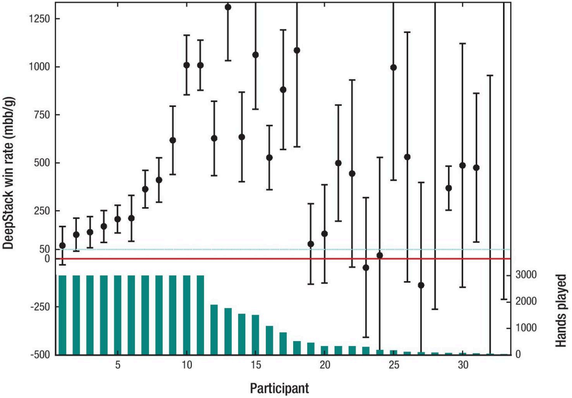
Abstraction?

- Traditionally abstraction was used to simplify the game
- Action abstraction – Card abstraction
  –> Translation Errors
- Deepstack only uses action abstraction in lookahead
- Card clustering is used for NN input.

Evaluation

- Exploitability – Play against humans
- Problems with Variance(Luck) –> 100.000 Hands for statistical significance
  –> AIVAT 3k Hands = 90k normal hands

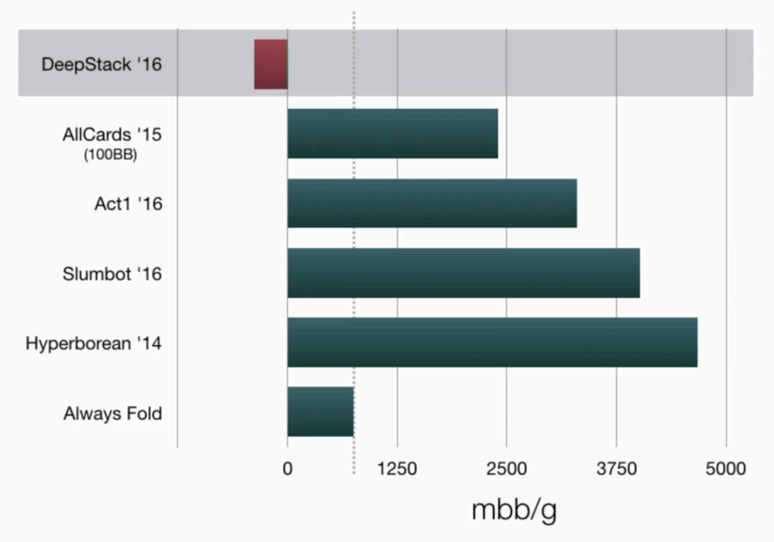
Pro players experimental results



Pro players experimental results

Player	Rank	Hands	Luck Adjusted Win Rate	Unadjusted Win Rate
Martin Sturc	1	3000	70 ± 119	−515 ± 575
Stanislav Voloshin	2	3000	126 ± 103	−65 ± 648
Prakshat Shrimankar	3	3000	139 ± 97	174 ± 667
Ivan Shabalin	4	3000	170 ± 99	153 ± 633
Lucas Schaumann	5	3000	207 ± 87	160 ± 576
Phil Laak	6	3000	212 ± 143	774 ± 677
Kaishi Sun	7	3000	363 ± 116	5 ± 729
Dmitry Lesnoy	8	3000	411 ± 138	−87 ± 753
Antonio Parlavecchio	9	3000	618 ± 212	1096 ± 962
Muskan Sethi	10	3000	1009 ± 184	2144 ± 1019

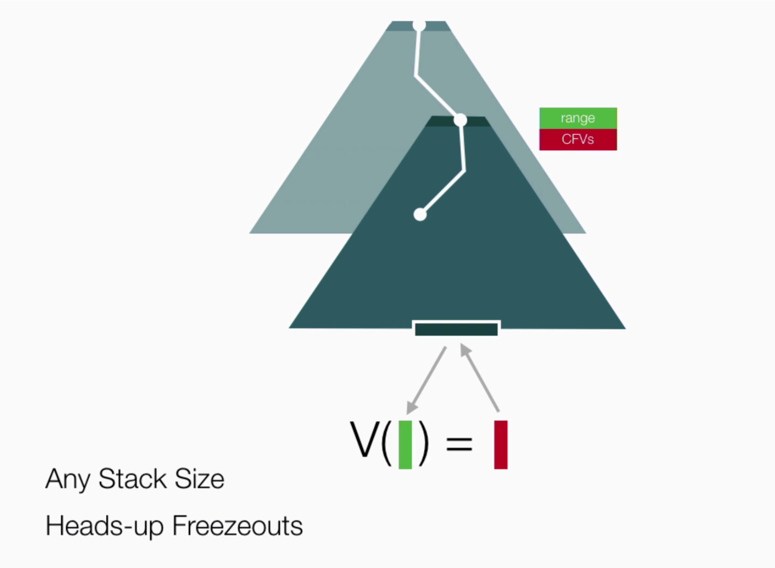
Exploitability



Nice to know

Thinking Time: 3s / action
7.2s / hand

Nice to know



Conclusion

- DeepStack beats Pro Poker player in No-Limit Heads-Up Holdem for the first time
- Connects Perfect information AI heuristical searrch strategy with imperfect information AI
- Plays with Nash Equilibrium approximated strategy
  $-\!\!>$ Doesn't exploit weaker players.
- No Multiplayer
- Can't explain moves but strategy tips can be taken away from DeepStacks play.

References

[1] Matej Moravcík and Martin Schmid and Neil Burch and Viliam Lisý and Dustin Morrill and Nolan Bard and Trevor Davis and Kevin Waugh and Michael Johanson and Michael H. Bowling
DeepStack: Expert-Level Artificial Intelligence in No-Limit Poker
Science, 2017

[2] N. Burch, M. Johanson, M. Bowling
Proceedings of the Twenty-Eighth Conference onArtificial Intelligence(2014)
pp. 602–608

References

[3] M. Bowling, N. Burch, M. Johanson, O. Tammelin
Science 347
145 (2015)

[4] Todd W. Neller and Marc Lancto
An introduction to counterfactual regret minimization.
InProceedings of Model AI Assignments
The Fourth Symposium on Educational Advances inArtificial Intelligence (EAAI-2013), 2013

[5] www.deepstack.ai

[6] www.depthfirstlearning.com/2018/DeepStack

[7] https://www.youtube.com/watch?v=qndXrHcV1sM

Thank You for Listening

Any Questions?