Efficient Learning for AlphaZero via Path Consistency

Dengwei Zhao[1], Shikui Tu[1], Lei Xu[1]

[1] Department of Computer Science and Engineering, Shanghai Jiao Tong University

{zdwccc, tushikui, leixu}@sjtu.edu.cn

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

ICML International Conference On Machine Learning

Introduction

- **Background**
  - AlphaZero [1] and its reproductions usually require huge computational power for learning. Thousands of GPUs or TPUs are needed.
  - A* search [2] expands node according to the evaluation function f and claims that $f(s) = f(s_0)$ for all node s on optimal path.
  - CNneim-A [3] relies on A* search's optimality to make a lookahead scouting to guide search process.
  - Although Path consistency (PC) was already schematically proposed [4], it is yet unknown whether it works well.
- **Our work**
  - Our paper develops CNNEIM-A's PC from three aspects: (a) Neural network is used to estimate f; (b) A* search is replaced with MCTS to incorporate with AlphaZero; (c) Moving average a sliding window of the estimated optimal path is considered.
  - We propose PCZero by incorporating AlphaZero with PC. For 13 × 13 Hex, PCZero obtains 94.1% winrate against MoHex 2.0, higher than AlphaZero's 84.3%.
  - We propose to combine MCTS simulated path with historical trajectory to estimate PC target and extend PC from the consistency of state values to the consistency of feature maps.

Path Consistency

- Board games have delayed reward ($g = \sum r = 0$). PC is turned into that "*values on one optimal search path should be identical*".

$$f(s) = \underbrace{g(s)}_{\text{accumulated cost from } s_0 \text{ to } s} + \underbrace{h(s)}_{\text{future cost from } s \text{ to preferred goal}}$$

- PCZero is realized by adding a weighted penalty to the loss function.

$$L(\theta) = L_{RL}(\theta) + \lambda L_{PC}(\theta)$$

- PC loss is $L2$ deviation from the average value within a sliding window.

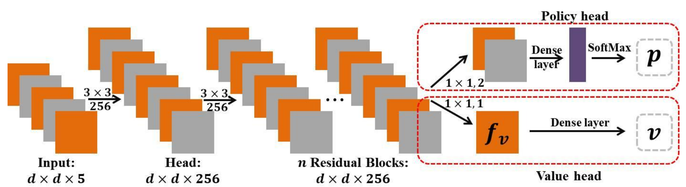
$$L_{PC}(\theta) = (v - \overline{v})^2$$

- PC can also be imposed on the high-dimensional feature map f_v, which contains more information about the game situation.

$$L_{PC}^{f}(\theta) = (f_v - \overline{f}_v)^2$$

- Based on the RL loss of AlphaZero, PCZero's loss function is

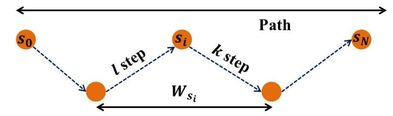
$$L(\theta) = -\pi^T log p + \gamma(v - z)^2 + \lambda L_{PC}(\theta) + \beta L_{PC}^{f}(\theta) + c||\theta||^2$$



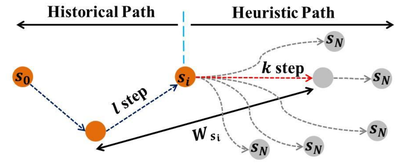
Detailed information about the architecture of policy-value network.

Implementation

- $\overline{v}$ is averaged over the l upstream nodes and k downstream nodes on a terminated game.



- MCTS-PCZero extends PCZero to calculate $\overline{v}$ using not only historical trajectory but also scouted heuristic path provided by MCTS while doing self-play, which can be seen as a trade-off between exploration and exploitation.



Experiment

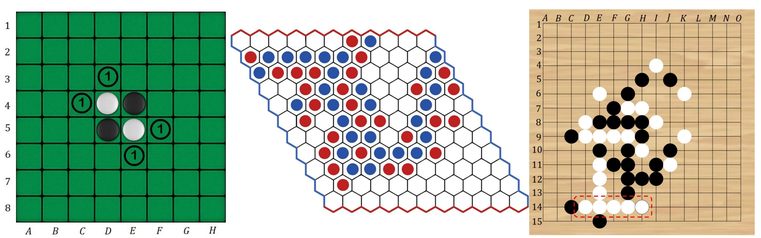
- Effectiveness of PCZero on Online Reinforcement Learning
  - For 13 × 13 Hex game, PCZero obtains **94.1%** winning rate, much higher than AlphaZero's **84.3%**, when competing with MoHex 2.0, the champion of Computer Olympiad in 2015.
  - PCZero consumes only 900K self-play games, which is a small-scale data that humans can make in a lifetime.

$0.9M \approx$ 2 games per hour × 12 hours per day × 365 days per year × 100 years in a lifetime

  - For Othello, PCZero obtains **74.8%** winrate, higher than AlphaZero's **69.5%**, while competing with 3-ply Edax.
- Effectiveness of PCZero on Offline Reinforcement Learning: MCTS player's performance is improved greatly due to the better generalization ability brought by PC.

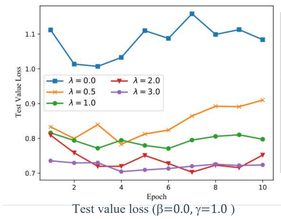
Game	Greedy Player	MCTS Player
Hex (8 × 8)	51.6%	58.6%
Hex (9 × 9)	53.1%	59.9%
Hex (13 × 13)	52.1%	61.5%
Othello	50.5%	80.5%
Gomoku	56.8%	64.0%

Winning rate of Offline PCZero against Offline AlphaZero (λ=2.0, β=0.0, γ=1.0).



Three board games: Othello, Hex and Gomoku

- PCZero gets much lower test value loss. Backup values provided by PCZero is more reliable. Increasing value loss's weight γ cannot replace the role of PC.

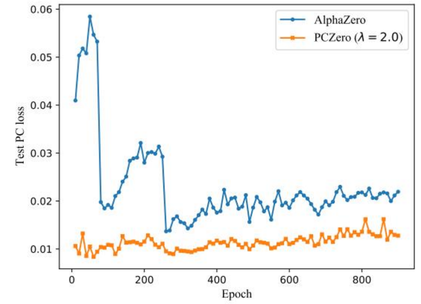


Test value loss (β=0.0, γ=1.0)

Game	γ	Greedy Player	MCTS Player
Hex (13 × 13)	2.0	48.8%	45.9%
Hex (13 × 13)	3.0	55.9%	55.0%
Othello	2.0	49.2%	34.8%
Othello	3.0	42.8%	42.8%

Winning rate against AlphaZero with γ=1.0.

- While training with self-play, AlphaZero's test PC loss also decreases, suggesting that PC is a nature required for strong value predictors.



Test value loss

Future Work

- It is interested to investigate the theoretical foundations under the PCZero scenario.
- It deserves to study the relative strengths between A* and MCTS when incorporating the PC nature.
- There is still room to consider better methods for the window selection.
- We plan to generalize the PCZero framework to more applications in combinatorial nature.

Reference

[1] Silver, David, et al. "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play." Science 362.6419 (2018): 1140-1144.

[2] Hart, Peter E., Nils J. Nilsson, and Bertram Raphael. "A formal basis for the heuristic determination of minimum cost paths." IEEE transactions on Systems Science and Cybernetics 4.2 (1968): 100-107.

[3] Xu, Lei, Pingfan Yan, and Tong Chang. "Algorithm cnneim-a and its mean complexity." Proc. of 2nd international conference on computers and applications. IEEE Press, Beijing. 1987.

[4] Xu, Lei. "Deep bidirectional intelligence: AlphaZero, deep IA-search, deep IA-infer, and TPC causal learning." Applied Informatics. Vol. 5. No. 1. SpringerOpen, 2018.

认知机器和健康计算研究中心 Centre for Cognitive Machines and Computational Health