Summary of Historical Developments in Planning and Search

By Bryan Travis Smith

In the previous project I wrote a summary of *Mastering the game of Go with deep neural networks and tree search*[1], where I discussed the novel combination of Deep Learning, Reinforcement Learning, and Monte Carlo Tree Search to produce the strongest AI go player to date. In this paper, I will review 3 research events that helped lay the ground work for this work.

The first paper is Bernd Brugmann's 1993 paper, Monte Carlo Go [2]. Brugmann work in this paper is using simulating annealing with simulated to decided which move should be played. The paper describes simulating the play of 10,000 games in such a way the odds of switching a move from the previous simulated game depends on the temperature of simulated annealing. From this simulation, a value of each position is generated from how often it is associated with winning. The highest value position is the one the agent plays. This work generated an agent that is ranked 25 kyu on a 9x9 board.

The second paper is *Bandit based Monte-Carlo Planning* [3] by Furnkranz et. al. where the group proposed using Monte Carlo methods coupled with a selection strategy that was based on statistical convergence to the optimal action instead of uniform sampling or heuristic based bias selection. They proposed using the Upper Confidence Bound (UCB1) applied to the tree search where each state-action pair has an estimated expected reward with a confidence interval. This method will select the path with the state-action pair with the highest confidence bound. The results are that error is reduced in fewer iterations previous Monte-Carlo methods and Alpha-Beta pruning across a wide range of branching factors and problem depths.

The third and final paper in this review is *Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search* [4] by Remi Coulom. This paper acknowledges that the assumption of the central limit theorem holding is not strictly held in tree search, and that Monte Carlo methods can be used to estimate the value and uncertainty in nodes. Selection of the state-action is done with probabilities based on the difference of mean values and the uncertainty of those value. Coulom also integrated the Monte-Carlo phase with the min-max evaluation by alternating the estimates as positive and negative of the estimated values in the simulated games.

Tremendous amount of work has been done with respect to Go and Monte-Carlo Tree search, and these are only three (almost randomly selected) papers on the topics.

[1] *Silver, David; Huang, Aja; Maddison, Chris J.; Guez, Arthur; Sifre, Laurent; Driessche, George van den; Schrittwieser, Julian; Antonoglou, Ioannis; Panneershelvam, Veda. "Mastering the game of Go with deep neural networks and tree search". Nature.* ***529*** *(7587): 484–489.*

[2] *Brügmann, Bernd (1993). Monte Carlo Go (PDF). Technical report, Department of Physics, Syracuse University.*

[3] *Kocsis, Levente; Szepesvári, Csaba (2006). "Bandit based Monte-Carlo Planning". Machine Learning: ECML 2006, 17th European Conference on Machine Learning, Berlin, Germany, September 18–22, 2006, Proceedings. Lecture Notes in Computer Science 4212. Johannes Fürnkranz, Tobias Scheffer, Myra Spiliopoulou (eds.). Springer. pp. 282–293. ISBN 3-540-45375-X.*

[4] *Rémi Coulom (2007). "Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search". Computers and Games, 5th International Conference, CG 2006, Turin, Italy, May 29–31, 2006. Revised Papers. H. Jaap van den Herik, Paolo Ciancarini, H. H. L. M. Donkers (eds.). Springer. pp. 72–83. ISBN 978-3-540-75537-1.*