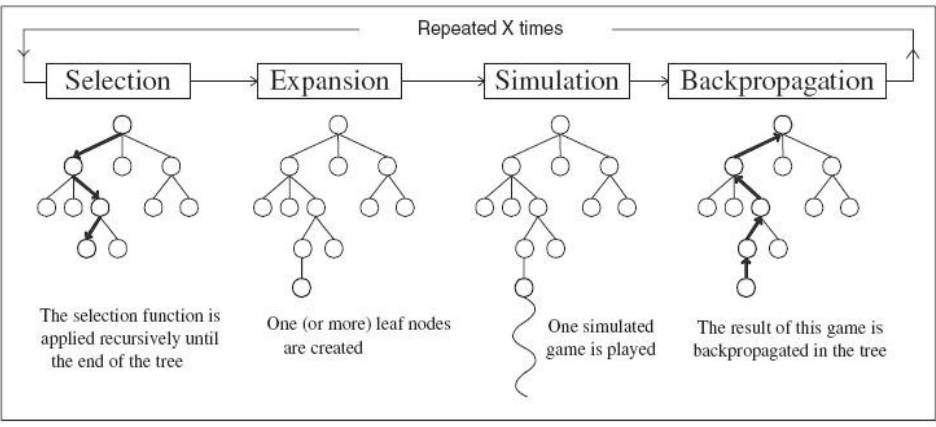
# Monte Carlo Tree Search

https://github.com/vuce/Monte-Carlo-Tree-Search-Gomoku

- Java로 알파 베타와 MCTS 적용된 소스 파일

Selection / Expansion / Playout / Back-propagation



실제 구현을 통해 확인해야 한다. Gomoku에 구현한다.

Simulation과 Playout이 여러 트리 노드에 대해 적용되고 Backpropagation 후에 최종 선택을 하는 것으로 보인다.

# CFR (포커 AI)

전체 게임을 다 플레이 시켜 두는 방법은 없는가? 포커 게임 AI에 적용된 방법을 적는다.

<http://poker.srv.ualberta.ca/>

Cepheus is the result of over a decade of research by the Computer Poker Research Group and a joint effort with Oskari Tammelin, a Finnish software developer. The team was led by Dr. Michael Bowling and included Neil Burch, Michael Johanson, and Oskari Tammelin. Tammelin invented CFR+, an extension of the University of Alberta’s CFR algorithm, which is how Cepheus learns to play poker by playing against itself. While the final steps were taken by a small team of researchers, they are indebted to all current and past members of the Computer Poker Research Group. It was also only possible thanks to nearly a CPU millenium of computation provided by Compute Canada and Calcul Québec

CFR (CounterFactual Regret Minimization 알고리즘)에 기반하여 전체 게임을 플레이 시키고 각 포지션에 대한 대응을 맵으로 갖고 선택하도록 한다.

An Introduction to Counterfactual Regret Minimization

Todd W. Neller\_ Marc Lanctoty

July 9, 2013

위 논문 내용을 기반으로 학습과 테이블 생성을 진행한다. 대규모로 실행해야 하므로 클라우드 서버를 사용한다. 신청해서 해볼 수 있을까? 일단, 여러 프로세스로 작업을 한 다음 정리해서 마무리 한다.

<http://cowboyprogramming.com/2007/01/04/programming-poker-ai/>

기본 구현에 대한 설명이다. 낮은 수준의 AI는 이 정도로 충분할 것으로 보인다. 확률 테이블을 미리 갖고 있는 것이 필요하다. Amazone ebook으로 구매한 것 필독해야 한다.

<https://webdocs.cs.ualberta.ca/~games/poker/>

포커 AI 연구 사이트이다.

<https://webdocs.cs.ualberta.ca/~games/poker/publications/abourisk.msc.pdf>

위 사이트의 위 논문은 CFR을 3인 포커에 적용한 예를 보여준다. 멀티 플레이어 게임의 경우 더 복잡해진다. 쉽게 해결되면 그게 AI이겠는가?