

Title	ターン制ストラテジーのための効率的な探索アルゴリズムの構築
Author(s)	藤木, 翼
Citation	
Issue Date	2016-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/13617
Rights	
Description	Supervisor:池田 心, 情報科学研究科, 修士

Depth-Limited Monte-Carlo Tree Search for Turn-Based Strategy Games

Tsubasa Fujiki (1310062)

School of Information Science,
Japan Advanced Institute of Science and Technology

February 10, 2016

Keywords: Turn-Based Strategy Game, Monte-Carlo, UCT, Evaluation Function, Depth-Limited Monte-Carlo.

A lot of research in the artificial intelligence field has already been done on games such as Chess, Shogi or Go, leading to the development of many search algorithms and machine learning methods. Thanks to these algorithmic developments as well as hardware improvements, it was possible for DeepBlue, a program from IBM, to beat the world chess champion in 1997. In Computer Shogi, programs also reached the professional level, with the breakthrough of the Bonanza method consisting in the machine learning of an evaluation function. Even in the game of Go, “AlphaGo” of Google Deep Mind reached professional level performance in 2016 with the use of Deep-learning technique and Monte-Carlo Tree Search. AlphaGo was able to win 5 official games against a human professional player. Such level of play in Chess, Shogi or Go is sufficient for almost all human players.

However, the research is still limited in the field of strategy games with “Multiple pieces move”. “Multiple pieces move” is the ability to move more than one piece during the turn of the player. There are many game genres where this ability is found such as Turn-Based Strategy Games, Simulation Role-Playing Games and Real-Time Strategy Games. This research focuses on Turn-based Strategy Games, with a limited but representative set of game elements. Example of existing commercial Turn-Based Strategy Games include *Civilization*, *Fami-com Wars* and *Daisenryaku*. The

computer players of these game are still weak compared to human players, so some handicap is needed to compensate the difference of strength even against middle-level human players. However, this solution is not satisfying for human players that prefer to play an equal game without handicap. Thus, this research focuses on the “strength” of computer player, which is one important factor to consider before other more difficult goals, like the entertainment or naturalness aspect of the computer player style.

In this research we used the Turn-Based Strategy Platform ‘TUBSTAP’, which was developed for academic research. Existing Turn-based Strategy Games often contain many small elements specific to the game which is somewhat an unneeded difficulty for research about algorithms. In comparison, TUBSTAP is a two-player zero-sum perfect information game without the most advanced components of commercial Turn-Based Strategy Games. It retains only the main components of commercial games, especially the “Multiple pieces move” ability. Thus, it is more complex than classical games like Shogi, but simpler than other existing commercial Turn-based Strategy Games. The TUBSTAP platform was developed for academic research purpose, and so it is easy to implement and compare different algorithms inside it. This is why we used this platform as the target of this research.

The main difference between Turn-Based Strategy Games and classical games such as Shogi is the possibility for the player to move and make actions like attacks with multiple pieces in a single turn. The number of possible actions for one player during one turn is very large. This is an important difficulty for usual search algorithms such as mini-max search, because there are too many legal actions to search. The computation time in a real game is limited, so a full search of the legal actions is hard to perform. A shallow search to a limited depth, or a search of only some representative actions is needed. However, the optimal search depth depends on the situation. For example, in a situation where some aggressive action is needed, the search can be limited to a small depth, but a detailed search of the possible actions is needed. On the contrary, in a situation where a defensive action is needed, search of some representative moves is sufficient, but they need to be evaluated to a more profound depth. Because Turn-Based Strategy Games contain such very different situations,

search algorithms will usually be strong in some situations but weak in some others.

Upper Confidence bounds Tree (UCT) and Attack Action Search are promising algorithms to handle the problem of the large number of available actions in games with multiple pieces moves. UCT is one of the most used Monte-Carlo tree search algorithm, where the formula for guiding the search is the Upper Confidence Bounds (UCB) formula. The UCT method is used in various games with a large number of actions, such as Go, so it is likely to be a promising approach also for Turn-Based Strategy Games. Attack Action Search is a special search method which was developed specifically for Turn-Based Strategy Games that have a large number of available actions. These existing methods are able to explore and find good aggressive moves, but on the other hand they cannot find effectively more defensive moves.

Therefore, in this research we propose a Depth-Limited Monte-Carlo (DLMC) search method with an emphasis on defensive moves. DLMC does not search all the legal moves. Instead, it limits the number of searched moves with a random sampling. For the sampled moves, simulations are done, but only for a limited number of turns, after which the state is evaluated with an evaluation function. Because of these particularities, DLMC is specialized in the search of defensive moves. In this research, we test first with experiments if DLMC is able to find correct defensive moves. The assessment is done by using DLMC only for the first move of the game from some given start position. All the other moves of the game are played with a different fixed algorithm. The experiment showed that UCT could reach a winning ratio 52.3%, whereas the DLMC performance was significantly better, with a 59.5% of winning ratio. However, DLMC is unable to find effective aggressive move. A simple performance evaluation showed it is weaker than UCT. DLMC could not exhibit sufficient performance alone.

Therefore, we propose a simple method which solves each of the problems by combining DLMC to find good defensive moves and UCT to find good aggressive moves. The best move of each algorithms is first computed, then these moves are evaluated by simulations and finally the move with the best evaluation is played. As a result, the winning ratio of our program against UCT (enhanced with a method called progressive widen-

ing) reached a winning ratio of 59.5% in battle experiments. We also show the efficiency of our program at handling tactical formation. The best performance is obtained when DLMC is combined with an algorithm that handles correctly attacking moves.