Generation of interesting instances of incomplete information puzzles, by using computer players of various strength

Hiroshi Omachi (1110012)

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

February 12, 2014

Keywords: Game, Puzzle, Incomplete Information, Automatically Generate.

Puzzle games have been enjoyed as one kind of game since a long time, and were usually played with special tools, books or sheets of paper. With the recent popularization of personal computers, it is now easier than ever for anybody to play incomplete information games. Mahjong Solitaire is a representative incomplete information puzzle, which is played by many players since it is included by default since the Vista version of the Windows operating system from Microsoft Corporation. However, the difficulty of the game may vary greatly depending on the initial arrangement of the tiles (an instance), and players are left unsatisfied by too easy or too difficult instances, or also unfair instances where luck is involved.

Then, in this research, our goal is to create automatically interesting instances for incomplete information puzzle games. Instance generation and classification has already been well researched for games like Sudoku, but the methods developed for such complete information or single-solution puzzles cannot be applied directly to incomplete information puzzles.

In order to generate interesting instances for incomplete information puzzles, we assume that an adequate degree of difficulty and fairness (absence of chance) are important elements that make an instance interesting for a

player. Under this assumption, we expect that interesting instances can be obtained by generating ordinary instances and removing those with an inadequate degree of difficulty, and also the unfair ones involving chance. We use Mahjong Solitaire as the target game in this research.

There is already some existing research to solve instances of Majhong Solitaire, but in this existing research, the solutions of the instances were found by using some information that is normally not available to the human player, like the arrangement of the hidden tiles. By using this information that is normally hidden, it was reported that as much as 97% of the instances of Mahjong Solitaire could be solved. However, in our research, our goal is to generate interesting instances for the human player, so it is preferable to avoid solving methods that have access to more information than the final human player.

For this reason, in this thesis, we use search methods that play the game in the exact same conditions as the human player. The search performance of the computer program can be adjusted through various heuristics and parameters.

First, in order to evaluate the difficulty of the instances, we implement a computer player that solves instances with tree search and Monte-Carlo simulations (called MSMC) under the same conditions as a normal human player. By changing the parameters of the computer player, we obtain several evaluation values (success ratio of solving) for a given instance.

We have implemented a Monte-Carlo search solver of Mahjong Solitaire, and with the best set of parameters, we obtain an average solving ratio of 91%/75%/58% on instances with 36 tiles/72 tiles/144 tiles in the shape of a spider.The instances with 144 tiles are 100 instances taken from the "Mahjong Titans" software bundled in Windows. If we compare to the existing research presented in the previous paragraph, the solving ratio for 144 tiles seems fairly low, but advanced human players including the author can reach only an average 60% solving ratio on the spider shape of "Mahjong Titan", which shows that our program is sufficiently strong to perform the experiments needed for automatic instance generation.

Then, with a two-dimensional plot of the success ratio of two particular computer players, we observe that the characteristics of the instance are well described by the difference of solving ratio between computer players

with different performances. Experienced human players, including the author of this thesis, confirmed that the distribution in the two-dimensional map can be used to distinguish good and bad instances. For example, we picked up an instance that was well solved by the weaker artificial player and hardly solved by the stronger one. Then the instance was played and evaluated by several human players, all of them evaluated it as a frustrating instance, because good and standard moves lead to bad results. Finally, we show that this method can be applied to generate interesting instances.