GGP – Player Improvement

Nicolai, Nera &Irme

1. Multithreading

It was decided to run two instances of the player at the same time when playing a game. One instance uses our alpha-beta search and heuristics to find a solution, the player thread uses Monte-Carlo tree search with UCT to find a solution. Just before the time out is reached, the scores are compared and the best one is chosen. The best solution is determined by the confidence of the solution. For mini-max, this is the best found score value once the search completes, for MCTS-UCT this was implemented as the lower confidence bound. If both are zero, mini-max is used as that one will in that case use our heuristics, if both are 100, mini-max is used as that means it has finished searching the game tree and therefore a winning terminal state has been found.

2. Rule editing

The idea is to speed up the reasoner by rearranging the terms in game description rules. For every rule, we try to find setting which term to be the first one gives us the best speedup. Once we determine that, we find the second optimal term in the body, and so on.

Currently, we are only obtaining a small speedup, as we discard many possible rearrangements to make sure we haven't compromised the reasoning by re-arranging the rules.