Kuncheng Feng CSC 466

Project Reflection

Overall Progress

At the conclusion of this battleship strategy research, the progress made was not as much as originally planned, mainly due to the amount of infrastructure work to set up the testing was much more than anticipated, on top of that the actual process of developing game playing machines is also more complicated than expected. But overall a decent amount of progress was made, and a few strategies were analyzed and proved more effective than a random strategy.

Program Friendliness

The function that served as the main goal of this project (getStatistics function) is decently user friendly, it takes no parameters to invoke and only asks the user for 3 integer inputs to start the calculation. During the calculations the terminal will periodically display some text to reflect on the current progress, to let the user know that his program is still functioning as intended. Although any wrong input before the start of the calculation will result in the exit of CLISP, in my opinion not much is lost.

However, the manual testing function (play) that allows users to play out against an AI, is pretty user hostile. At each turn it floods the terminal with text, making it difficult for the user to quickly identify what is the most important event for that turn. And the most criminal of all is that if at any point the user inputted a wrong coordinate to fire at (out of bound, (0, 1) instead of (A, 1)), the program will immediately crash and waste all the player's effort being put in. Normally this is unacceptable, but since this function is not the main focus of the project, and giving the time constraint for the development, this issue is disregarded for now.

Program Performance

Mainly due to the restraint on time, the program was developed with the focus being human readable, so any mistakes can be found quickly and fixed. Due to this it performs very slowly, taking a very long time to simulate games. Also due to the limitation of CLISP, some data structures are not as concise as I liked, and it does not allow easy parallel programming to increase performance.

Findings

The program concludes that some simple tactics will achieve wonders against a random playing tactic. However what is interesting is that some tactics we considered as a positive, placing ships not adjacent to each other, is actually a negative as our normal playing styles counters those tactics.