

"Motivate and present planned experiments"

First of all, the project is no longer tied to the game "ORTS". A heavily simplified "game" engine will be made up and used instead, to give us a system to test our AI on while keeping the complexity down.

Background:

Although almost modern strategy game contains some sort of AI - mostly a collection of "if x, do y" statements and predefined strategies - there have been done very little in the direction of "learning" AIs. This has various reasons, like preserving processing power and reducing development time, but in general it's viewed as less important: Most strategy games focus on two areas - the heavily scripted single-player campaigns, where only AI is sparingly used, and multiplayer, where you don't need a central AI at all.

Requirements:

We intend to reach our goal in stages. First, the learning AI should be able to beat a simulated opponent using the same tactic every game. When this is achieved, we know that the AI is, actually, learning by playing. The next step is attempting to learn to beat a simulated opponent that uses different tactics between games. Even for a well-balanced game this should be possible, as long as the AI can adapt to changes while the game is running. The final step is to learn to beat an opponent which not only changes tactics between games, but also while the game is running. While there will be a possible "best" strategy for the two first stages, which the AI can learn, a well-balanced game will make it impossible to achieve a winning strategy completely, as any tactic should have a counter-tactic. If the game is not well-balanced, the learning AI should be able to determine which strategy will yield the best chance of winning. Because we will be using a self-made simulator, we can make the game slightly imbalanced to determine if the learning AI can detect this imbalance and use it to determine a winning strategy.

Design specifications:

The design consists of two parts: The simulator and the AI. The simulator will simply be a tool to update information, both to the human controller and to the AI, every "tick" of gametime. It will also take input, again both from the human controller and the AI, about, for example, what units is produced and destroyed, and in what amounts the simulated opponent should collect various resources, produce various units, etc.