Master's thesis in
Information- and Communication Technology

Title:

Finding optimal rush attacks in real Time strategy (RTS) games

Candidate:
Wenjie Li

Supervisor:
Ole christoffer ， UIA

Introduction

Commercial computer games is an important parts of entertainment industry, it is very popular both in young and old people. RTS is one of the most popular one. RTS is short for real time strategy. It simulates a military section. In general, several players struggle in a resource scattered map. Player commands the game units to developing economy, climbing technology, crafting new troop. The aim is crafting a powerful troop and guiding them into battle. Afterwards, destroy you opponents' base and troop. RTS game is randomly, real time, and unpredictable. Hence, it brings high competition and intensity attention to players, especially when you play with a human player.

However, many players are aware of the AI is so weak and changeless. Why the AI develops didn't follow game step like the relevant arena such as classic board game. There are some reasons:

a) Firstly, **the RTS game world is really complex**. It contains various objects, a huge number of features, partial visible information, micro actions, and real-time decisions. In contrast, current RTS game AI players performance in an opposite direction. Its turn based, perfect information, most action have a predict sequences. The most important is decision making based on a stable manner.

b) **Commercial RTS game doesn't open its source** for commercial reasons. Also, in order to make the graphics of the game more realistic, most time are spend. It means only a few time is cost in AI research. In addition, lack of resource restrains the third party and fans to develop AI.

c) Most people are interested in human competition. We have a lot competition for humans, such as WCG, ESWC, but **lack of AI competition**. It decreases the motivation of those people that are able to develop AI. [1]

We start at the weakness point of currently AI which is the learning ability. In general, the predicted sequence can't handle the situation that would occur in complex RTS world. In this case, a learning ability is a great enhance for AI. It helps AI to learn from the world and change some old fashion and mechanism. However, we can't suppose the AI would do perfectly; it still gives AI more opportunity to make the right decision and win the game. So, we would investigate the learning ability in RTS AI and try to find a suitable one. Cause, commercial RTS game didn't open its resource, our research would base on an open source environment: ORTS.

Solution

In general, we divide research process into two steps. At the first, we analyses the standard game information in ORTS. Afterwards, we design our AI package with rush-attack strategy. Our experience of playing RTS game is a great enhance in package design. Then we could investigate learning methods based on it. As the research going, we have deeper understand of RTS game. As a player, we only got unperfected information and unstable game states. It's really hard to find a decision sequences with universal power. Also it is impossible to presents all possible decision sequences. Cause it would be so huge and infinite. Hence, we decide to design a new method. We notice that the evaluation system in commercial games always present the winner with higher point. In same sense, the point gives some clue of the game process. Based on this inspiration, we design an advanced point system. It gives a point to each game unit. AI tries to make decisions to maximal the points. The higher points AI get the higher probability to win. The point is calculated by units' information, current game states and "experience". Experience is a kind of feature that can influence the game in an indirect manner, such as the lost attack power problem. AI learns the "experience" from game to game. It makes the point more precise and game process would change due to the update of "experience". Format design is described in the following.

It is a quintuple {P, F, S, E, M}.Fig 1 represents its relationship.

- P: the point of units.
- F: the unchangeable game information, like hit point, attack power and so on.
- S: game states which may influence the point. For example the set up of opponent troop
- E: Experience, it the key structure in this system, it's a probability that denotes the performance of units in real conflict. It could have different format.
- M: method to learning, in a word how to make the experience more precise and efficient.

$$P = F * \sum(E*S);$$
$$E_{new} = M(E);$$

Fig1 Relationship of expressions

The expression used in our design is presented in Fig n. we just implement a most simple way to prove our conception. Only three units are considered and opponent troop is predicted. In addition, the structure of each expression can be more complex to show other RTS game features.

$$\frac{HP*AP}{CD} * \sum\left(\frac{SHPi}{SAPi} * \frac{Num\_i}{T\_num}\right) = Force$$

F E S P

Fig 2 Expression of our implementation

Conclusion

Our testing simply proves the effect of the system. In general, the probability of win is increased while the learning process is going. Finally it stops at a certain value. However, we just implement a partial element that could influence the game. So in some special situation, the probability would decrease or still very bad after learning. There are other elements that influence the game result. These would be future works.