Optimizing Small Scale Combat With Neural Networks

Peixian Wang

May 3, 2016

Abstract

A major challenge in real-time strategy games is individual unit control, where the player must issue separate orders and deploy units in different places in order to maximize the effectiveness of a single unit. Starcraft: Brood War (henceforth referred to as Starcraft), a real-time strategy game released in 1998 by Blizzard Entertainment, provides a ripe environment for optimization within combat deployment. We present a backpropagation neural network design that hooks into the Alpha-Beta combat algorithm, which then allows an AI to consistently deploy units in a manner to beat the build-in computer AI.

1 Introduction

Micro-management, or the management of units on an individual scale, is an extremely integral part of modern real-time strategy (RTS) games today. Players are required to move units in order to gather resources, construct buildings, or engage in combat. Within combat alone, effective players must decide: 1. which units to engage with, 2. proper battlefield to engage on, and 3. where the engaging units should be placed. Oftentimes, this level of decision making has happened on the scale of seconds for human players, where quick decision making is oftentimes rewarded with a succesful engagement. Due to the fast-paced nature of combat micro-management, decisions are oftentimes complex despite only having a small number of units.

Modern RTS's provide a limited amount of assistance within micro-mangement, commands can be issued en masse to a group of units. However, effective management of units on an individual scale oftentimes lead to a more favorable outcome, such as the technique called Scourage Cloning[1]. While players are able to devise complex techniques in order to work around the limitations of Starcraft itself and selectively issue batch commands when necessary, AI playing Starcraft do not have access to the same nuances and quick decision making a player does. Consider the following scenario: a player has a heterogeneous distribution of units, with both units that can only attack within a very small melee range (melee units) and units that attack from a distance (ranged units). A good player would know to place melee units in front of the ranged units in order to maximize combat effectiveness when moving the group, as the ranged units can still fire upon the enenmy when the melee units are attacking the enemy, but the melee units would be blocked behind the ranged units if unit positions were switched around. To an AI, however, this distribution of units looks homogeneous, there is no easy way for the AI to realize that ranged units would be more effective if placed at a distance behind the melee units.

This combat-oriented domain of micro-management is extremely well-suited for the applications of neural networks, as neural networks can learn the proper positioning of units based upon repeated trials and a large corpus of training data. In this paper we present an application of a neural network trained through backpropagation, over the course of 3000 games an AI properly learns the correct deployment of a small melee unit called the Zergling. In section 2 we discuss some background on Starcraft and SparCraft, a combat simulation program. In section 3 we discuss the technology stack used to enable this neural network and the AI itself, along with introducing our AI called Ultralisk. Section 4 recounts the technological approach to solving the problem, including the topology of the neural network, obtaining training data through SparCraft, and feeding the data into the

[1] http://wiki.teamliquid.net/starcraft/Scourge_Cloning

Alpha-Beta combat algorithm. Section 5 and 6 detail the results of the simulation and experiment trials, along with concluding and discussing the interpretations of the results.

2 Background

Starcraft is an extremely complex game that has captured an entire country, becoming the first major e-sport. At a high level, Starcraft consists of 3 stages:

1. Use units to gather resources (called *Minerals* and *Vespene Gas*).
2. Spend resources to construct buildings, which enable gathering more resources and constructing more powerful units (called *expanding* and *moving up the tech tree* respectively).
3. Use the accumulated units to engage the enemy units, with the eventual goal of destroying all of the enemy buildings and units.

3 Technology Stack

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

4 Technical Approach

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

5 Results

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

6 Conclusions

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.