# 

# Computer Games Development CW208

# Research Document

# Year IV

## Dylan Curran

## C00217112

## 03/05/2020

**Faculty of Science**

**Open-Book and Remote Assessment Cover Page**

**Student Name: Dylan Curran**

**Student Number: C00217112**

**Lecturer Name: Lei Shi**

**Module: Project II Y4**

**Stage/Year: 4**

**Date: 03/05/2020**

**Declaration**

This examination/assessment will be submitted using Turnitin as the online submission tool. By submitting my examination/assessment to Turnitin, I am declaring that this examination/assessment is my own work. I understand that I may be required to orally defend any of my answers, to the lecturer, at a given time after the examination/assessment has been completed, as outlined in the student regulations.

[**Computer Games Development CW208**](#_y1109jl4peni) **1**

[**Research Document**](#_sbowdgo0u95y) **1**

[**Year IV**](#_53ewi7wpg3ja) **1**

[Dylan Curran](#_873rjmb5lxsc) 1

[C00217112](#_h0kkxbvl5rcy) 1

[03/05/2020](#_mz2vd86hpa2l) 1

[Project Abstract](#_qlthlmqnntk6) 2

[Project Introduction and Research Question](#_gn9k3zfmz6e) 2

[Background](#_et42crx1ry2o) 3

[Literature Review](#_czzdftgjw74t) 3

[Key Concepts](#_8ai0tu7d8kf6) 3

[What others have done](#_6fckojjsazfb) 4

[Study](#_zgiy93nq84b8) 4

[Project Description](#_6pr3apy080p5) 5

[Project Milestones](#_4vpwlskmp62w) 5

[Milestone 1 December 18th](#_izcjv573pxok) 5

[Milestone 2: March 15th](#_vaokzqqz99z7) 5

[Milestone 3: April 20th](#_b1cio58q9npo) 5

[Milestone 4: April 27th](#_p4k3bthqhu6d) 5

[Results and Discussions](#_6q8qd42y3dhu) 6

[Project Review and Conclusions](#_kcx8uycf9ahq) 6

[What went right?](#_nsnubegz75gw) 6

[What went wrong?](#_m1u36dt9htgf) 6

[What is missing?](#_faqxrfrletce) 6

[What would I do differently?](#_bvio482j0lmp) 6

[Advice for people doing this in this future](#_1u7x5bw3dsex) 7

[What to research if you were to study something like this next year](#_nsuyns4unir3) 7

## Project Abstract

For this project I chose to look at AI Learning. The reason I chose this is because I felt that simulating an AI to make it look realistic would be a very difficult job, henceforth why I chose this. I specifically chose Q Learning as my Ai learning type due to the fact my AI would be searching for something and Q Learning gives positive and negative reinforcement for actions so I felt it was perfect for this project.

## Project Introduction and Research Question

The game I have in mind is a 3D stealth strategy game where you (the player) must escape the house locked by the murderer hunting you in the house. Your aim is to gather collectibles before the AI hunts you down. This game could have potential to have many more levels created over time with an adaptive AI that will hunt the player. Any audience can play this game but it would be more suited to those who love stealth games.

My research question for this is “How does Q Learning apply to a stealth based game?”

The reason machine learning is so important to study and understand is because everyone is trying to make AI as realistic as possible to simulate as much realism as possible within a game, within the community it is seen as extremely difficult to make a human-like AI as the human mind is so complex to understand, so coding all of it would even be more difficult as a result. This is an ever expanding field of research and I would like to complete some research on it as a result. After having a look at Q Learning I decided this was the type of machine Learning that I wanted to attempt for my game.

A few examples I have seen that gave me a guideline would be from the bot AlphaGo created by Google. AlphaGo was a bot created to compete at the famous board game “Go”.

A later bot AlphaZero would become even greater than the bot AlphaGo and AlphaZero would go on to beat world champions at chess, shogi and go.

## Background

A beginning part for this background would be looking into the company Deepmind which is owned by Google .A few examples from the company deepmind I have seen that gave me a guideline for my research would be from the bot AlphaGo created by Deepmind. AlphaGo was a bot created to compete at the famous board game “Go”. AlphaGo through lots of training against amateurs, then against itself moved on to beat world champions such as Mr. Lee Sedol who is an 18 time world champion at Go. AplhaGo was given a 9th dan rank in Go (highest rank achievable) and was considered the best player until an upgraded version called AlphaZero learned Go randomly by playing against itself with random input by itself and learning from it’s mistakes.

AlphaZero went on to win other games such as shogi against world champions based on teaching itself input from learning about it’s mistakes. AplhaGo did the same thing in a game of chess against a former AI that used to be the best called Stockfish which was developed by open source developers. AplhaZero (the upgraded AplhaGo) only lost 6 out of 1000 games against stockfish, winning 155 and drawing the rest.

The next thing I saw from Deepmind was the AI they created to move a character forward, they just started the bot off with no input and only 1 goal which was forward progress. Over time the AI taught itself how to walk forward and when obstacles came into play the AI learned how to swerve left and right. After this the AI was also applied to other models such as a spider-like model with only 4 legs that learned to climb and jump even though it only started off with the same goal as the AI which was forward progress.

## Literature Review

### Key Concepts

The main idea of machine learning is that an AI with no prior domain knowledge will learn from it’s mistakes to better itself and seem like it is learning like an AI would.

The AI does this based on data analysis as the AI takes in the data and uses it’s systems to make decisions based on that input. The AI does not depend on human intervention or at least tries to prevent it as much as possible for a better AI in that domain.

For reinforcement learning the AI learns via positive and negative input based upon the actions the AI carries out, the environment for the AI to adapt to. For this the AI has what is called a weighting system where actions can be considered more valuable to the AI than others, thus they are given more importance in weighting e.g. Taking the Deepmind AI for forward movement weighing for walking forward could be a plus of 0.1 and this could be added every movement forward, however falling is worse so a negative 0.5 could be given due to the importance of not falling over.

### What others have done

For this example I took a look at some of the workers at Deepmind and their opinion on Q learning.

In their study they talked about what reinforcement learning actually is and gave a definition for it. The main problem they looked at was how the AI may overestimate it’s goal and deepmind wondered if this actually affected the AI in a good or a bad way, they felt it depended on if the goal was not uniform and not concentrated the AI would suffer as a result.

They said the idea of Double Q Learning (added deep neural networks and function approximation) to give better estimates and overall do better on the games it’s applied to.

Later on in the document they talk about the formulae that go with the deep neural networks, Q learning and double Q learning.

After this they begin to graph their results based on Q learning vs double Q learning and graph estimates based on the learning types with values and true inputs with a comparison to see how close they are.

Later they compare the two types of learning against 6 Atari games they played under the same training conditions. After this they begin to summarise the main points they have discovered based upon the research they did.

### Study

To try and answer my research into Q Learning I have to set up some clear things for my AI. The AI must have one simple goal, which is to find the player, however the AI must also have negative feedback and the AI cannot remain stationary so there will have to be an area for the Ai to wander around getting negative feedback if they are unable to find the player. However the weighting must be increased if the AI actually catches the player. The discount factor for this will not really change as the AI is searching for the AI from the start so the discount factor is not too relevant here.

To try and achieve this I will need to store AI feedback to see what is considered a good or a bad decision and have that information carry across over multiple games so that the AI is learning over time and not just per game.

The aim for this AI in the future is that they will be able to find the PLayer in a big level in a short time frame due to the AI’s understanding of the player movement they have learned from the various games against the human player.

### Project Description

For my project some of the specifications do not match up to what I designed based upon my own bad input during the course of the project, the AI does not work as I wished it would due to the fact that I made a mistake with an array I was using and as a result my Ai was no longer working. Another change that had to be made was that some of my walls were supposed to have doors and due to my poor time management I was unable to implement the doors successfully, apart from this I did not make as complex levels as I had wished to make also due to my poor timing skills.

For my technical learning achievements I learned how to successfully use a navmesh and adapt it to any environment. I also learned how to make a camera follow a player despite how the player and camera would not rotate in sync.

As for personal achievements I don’t really have any I’d mention off of the top of my head.

## Project Milestones

### Milestone 1 December 18th

The first level was designed with a player and a very basic Ai that could move around the house but not well.

### Milestone 2: March 15th

This milestone was having a better working level rather than the older level that was not very efficient.

### Milestone 3: April 20th

This milestone was having an updated player movement that would rotate and move in first person with the camera.

### Milestone 4: April 27th

This milestone is where I made the first tutorial level with obstacles for the player so the player would understand the basic movement system for the game.

## Results and Discussions

Due to the fact I was unable to complete my AI due toa mistake I made the only results I had based on the information I had was that the AI responds well to positive reinforcement when moving towards the player after positive feedback is received for the first time and the AI realises this is the correct decision to make, otherwise the AI knows moving from one location to another is a bad decision due to the fact that they cannot find the player, thus the AI begins to learn that the player is their only source of positive feedback and whenever the AI sees the player the Ai will now actively chase the player.

## Project Review and Conclusions

### What went right?

I was happy with the ease of the level design as I could make another level in a couple of minutes, I also thought the player movement went well as the first person movement was more difficult than anticipated and I was able to get it working in the end. Lastly I was happy that I was able to effectively learn how to use and adapt a navmesh per level as I has unsure about the navmesh and how to work it all so it was a learning experience for me.

### What went wrong?

First and foremost the AI did not go near as well as planned due to an error I made with an array and as a result my AI didn’t actually work, nor was I able to test it. My initial level that I made never had collisions on it for some reason so I had to scrap the level due to the problems associated with that. I also had trouble pushing large files at the start due to me needing a large asset file for the level that I no longer used but that set me back a bit.

### What is missing?

What is missing from this project is a working AI due to the array problem. The completion of a level is missing due to the fact I only have a debug log saying you completed the level. I would add doors to the level too so they are missing too.

### What would I do differently?

If I was to do anything differently I would definitely work on the project earlier and adhere to a more scheduled layout rather than have lackadaisical layouts for time, if I kept up the motivation I could have had a much better project.

### Advice for people doing this in this future

Start with the AI first going to a stationary location, then create more locations and have the AI wander, after that, create a player that can move and have that AI try to find the player, after this maybe try creating some levels and see how the AI adapts and finally adhere to a good schedule or you will fall behind.

### What to research if you were to study something like this next year

What I would highly recommend studying from this based on what I have done along with my research would be trying to compare Q Learning AI to Double Q Learning AI in a stealth like game and maybe see if there can be evidence proven that the Double Q learning bot would be more efficient in finding the player and try to get an accurate scale for the difference between the two AI. I would recommend making a large map that would be maze-like with a player, thus allowing the AI to perform many actions which will allow the observer to distinguish the scaleable difference between the two methods of Q learning.

## 

## 

## 

## 

## 

## 

## 

## References

|  |  |  |
| --- | --- | --- |
| Referenced Publication | Citation | Reference |
| Website | Hado van Hasselt , Arthur Guez, and David Silver 2016 | van Hasselt, H. ,Guez,A., and Silver, D. (02/03/2016) Deep Reinforcement Learning with Double Q-Learning [Online] URL: <https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/viewPaper/12389>  AAAI Publications (Accessed March 20th and May 3rd 2020) |
| Website | Nicolas Heess  Josh Merel  Ziyu Wang 2017 | Heess, N., Merel, J., Wang, Z.(10/07/2017) Producing Flexible behaviours in simulated environments [Online] URL: <https://deepmind.com/blog/article/producing-flexible-behaviours-simulated-environments>  Deepmind Blog (accessed Novemeber 12th, January 10th and May 3rd) |
| Website | Unknown | Unknown(Date unknown)  AlphaGo is the first computer program to defeat a professional human Go player, the first to defeat a Go world champion, and is arguably the strongest Go player in history. [online]  URL: <https://deepmind.com/research/case-studies/alphago-the-story-so-far>  (accessed 5th November, February 9th and May 3rd) |