

# Thoughts about use of AI in video games

---

Una Benediktsdóttir

## Why?

As the final project of Introduction to AI, it was decided that the aim of the project would be to create an artificially intelligent bot within Unreal Tournament 2004. However, after some futile attempts and frustration due to jumping into the idea without having done too much research on both the coding and thinking behind AI in shooting games, the project was to an extent, abandoned. However, not in terms of complete resignation but rather in the aim becoming more research than execution, driven by the curiosity about the problems and obstacles that need to be overcome in order to create a realistic and reliable artificially intelligent bot.

There have been hundreds of articles and many books written on the subject, so of course it is not an easy matter to sum up in a few pages. However, the aim is to, if nothing more, create some kind of general idea of the main problems that were encountered from the attempts at making a bot, and the research that followed these problems.

## Introduction

The research and exploration for using artificial intelligence in games is far from being a fresh and can be traced back all the way to 1951, when Christopher Strachey made a program for chess and Ddietrich Prinz wrote a program for chess. Of course they were basic and far from being perfect but they were a stepping stone for the future of development in the particular department of AI. For example, inspiring Deep Blue, a computer made for chess playing developed by IBM with the objective of beating the world champion of chess at the time, Garry Kasparov. In 1997 on May 11, Deep Blue succeeded in this task. This success inspired the creation of more programs of the type and despite objections by some stating that there was some sort of human manipulation that aided in the victory, really became an example of the power of artificial intelligence.

This previous example was in terms of a more or less physical game of chess, and does not completely fall into the category of a “video game”. However it is important to note the initial stages of AI emerging within the physical gaming world, outside of the “virtual world”.

Indeed, these initial efforts in 1951 were well before the creation of the first video games, which didn't occur until around 1960s and early 1970s, 10 - 20 years later. Additionally, these games, such as Pong and Spacewar! purely relied on discrete logic and were strictly based on competition between two players without the use of AI. It wasn't until the 1970's that games with Person vs.

computer and the idea of computer enemies began to emerge. These ideas continued being built upon and up to this day research and experimentation continue in order to ameliorate AI in games.

It would take a long time to list every little discovery and improvement that has been made over the years, especially taking into consideration every type of game and gameplay that exists currently. There is one section of games where artificial intelligence can be and is used extensively in order to make the game playing experience more realistic and overall enjoyable for the player. These games fall into a few different categories, such as action, adventure, first person shooter and combat games overall.

## **Combat and shooting games**

The implementation of AI in combat oriented games can be detected in virtually every shooter game since the early mid-nineties. The dependence that designers of FPS games possess in terms of AI increases day by day, as it is a particularly substantial feature to implement in order to make the games stimulating, not to mention playable. However, in the beginning, the evolution of AI was not very fast and remained in a stable, rather unremarkable state for many years. In this state the AI NPC's often lacked the idea of defending themselves and were prone to shambling into the player's gunfire without a second "thought". A noticeable step up in this technology was made by Valve when they released Half-Life. In this game the AI combat took a large step in the right direction. Half-Life quickly began proclaiming itself, and quite deservedly so, for having the most developed AI of any game on the market. This breakthrough was shown through the AI of the Marines who showed, amongst other things, different reactions when shot, well-organized spotting of grenades and additionally in some ways a very realistic attentiveness and response to other players.

Development continues and in the current games AI NPC's can be seen react to their environment in very realistic ways, such as ducking behind objects, tossing grenades back if thrown at them and dodging incoming fire and even sometimes standing in for real players in the form of game bots. There are so many different factors to be considered when attempting to make an AI player. Notably, some skills that human combatants naturally develop and have a much greater understanding over than any AI are the ability to use teamwork efficiently, knowledge of their environment, hunting ability and survival instinct.

## **Interacting with environment**

The ability of dynamically interacting with a terrain is already quite important within games and will become increasingly important in the future. This is one skill that AI do not possess to the same extent as humans do, or at least not by the same methods. If a human is given a good and solid map within a game then they will inevitably be able to find places to hide for cover and alternate their routes to important destinations and even plan and find good spots for ambushes. Although AI bots can to some extent do the same, they are usually specifically programmed and trained to do this in a specific environment, and do not understand why they are doing it.

In many cases the NPC will have some kind of map of nodes or prior knowledge of the environment that enables him to direct himself from some point A to some point B without much worry of obstacles (since these nodes are usually in places that the NPC or bot can reach). The NPC needs to be made aware of the world in order to choose which decision and direction to take. For humans and other biological creatures, this is usually quite simple to do and they are extremely efficient at using their physical sensors such as feeling or seeing to navigate. In most games, navigation of NPC's is almost never achieved in this way, since it would be unnecessary effort on the hand of the programmer. This is because the world is virtual so it can be entirely stored in the computer's memory, and is typically processed by an algorithm in order to extract its structural information. This way, the bot can receive knowledge of the terrain it will be traveling before the game begins.

Moreover, due to the fact that the NPC is technically a part of the simulation of the real world, its position will always be perfectly known, and in addition the NPC will know its position in terms of the entire environment. This is an example of something that the NPC has which a human player doesn't. Even though the player might have a good detailed map of the environment or he is already familiar with it, often, especially in the beginning of games, the player has to figure out the environment and his place in it by exploring. When a bot is given knowledge of the environment, it can use standard path finding algorithms such as A\* to decide which moves and actions to take.

However, as stated before, when programmed this way, the AI does not have the apt *sense* of the world like a human player would, so the bot can be very prone to common problems such as bumping into walls or ignoring dynamic objects. This can happen because the previous knowledge of the environment that the bot is relying on is information which was preprocessed, and sometimes did not take into consideration the dynamics of the environment, and is not being updated throughout the gameplay. This problem has been solved to some extent through changing the problemsolving tactic from purely predictional path finding, to reaction based tasks and rule based systems. Do not be fooled though, the reaction based perspective of the problem contains its problems too, namely the fact that it is very good at dealing with low-level details and travelling accurately across short distances, but often has trouble reaching particular points in space which are further away with higher level details.

## Teamwork

Additionally, an important factor of shooting games that are multiplayer games to some extent is the idea of teamwork. Once a player gets to a certain level they have usually trained themselves to play and think according to some kind well organized team plan. Players will often create teams that are referred to as clans, such as in Counter-Strike. Teamwork is extremely important in this game due to the fact that the idea of the game is to meet certain objectives within a time limit, and most of the time these pursuits will be strained by the presence of enemies.

The difference between the organization and efficiency of plans within clans is a far way away from those who are working as a team outside of a clan. This is because efficient teamwork will often take much planning and learning to move in coordination with other team members, and this can sometimes take some time to master. Once it is mastered it is much more efficient than being a

stand-alone player with no support from other team members. Therefore in order to make a team of NPC's in shooting and combat games more realistic and effective, the idea of teamwork would probably have to be used in some way. It might prove to be difficult, however, to teach the computers to work as a team, as it is a skill that even many humans have a hard time mastering. Bots, (and even some humans) need to be taught to differentiate between when to help a member of a team instead of merely thinking of itself in terms of survival, and when thinking of oneself is more important.

## Hunting and surviving

Finally, the balance between hunting and survival within AI can sometimes be hard to level out. In the first FPS games such as Wolfenstein 3D, AI were usually in "guard mode", and no matter whether the human player is going out for the kill or just trying to sneak past, the AI has to be able to make a quick decision about how to react. The hunter aspect, where it is up to the hunter to try to catch a human player, is very complex and requires many sensors to be used together to create a decision.

A second idea that AI need to have built in is the survival instinct that most people develop naturally. This survival instinct falls into the category of actions that are taken solely for the purpose of self-protection even when there is no attack going on at the time. For example, even though a human player cannot see anyone approaching or no one is attacking them, they will still have the instinct of hiding behind something while they reload their weapon or try to run short distances between objects to hide behind. They do not do this because there is an immediate danger but just the potential of danger. This is an action that must also be thought about in terms of AI in shooting games. Survival instinct differs in level from person to person, of course, and can sometimes even be called cowardice if there is too much hiding, but this is nevertheless a natural reaction that people have to danger. It is also a reaction that will not come naturally to an AI unless it is programmed to, and even if it is programmed to it might still not be as efficient as the human response. This is due to the fact that it is often triggered by an emotion of fear, which would yet again be another level to add on to the AI bots in order to make them human like instead of suicidal; emotions. That is a completely different debate however.

One key element to think about in terms of these ideas is to balance out the seeking and fleeing, pursuing and evading, steering behaviors. That is, moving towards or away from certain destinations or objects. An NPC needs to know which times to seek an object or pursue another player, and which times to flee or evade them. Not all objects should be picked up, such as live grenades (unless it is with the aim to throw them somewhere different), and if the NPC is low on life then he should be seeking for an object and not pursuing an enemy. For a human being this comes as common sense, and the player often has to make decisions between staying in a battle or retreating and looking for life or ammo, but to an NPC it must be laid out and taught to them as they will usually not figure out the right balance by themselves.

## Balancing the skills

NPC's will almost always have one ability that humans don't, and that is the power of being a computer. Most people are aware of the fact that without humans telling computers what to do, they are not very "intelligent" by themselves. They do however possess a computational power that few if no humans possess, which makes them extremely powerful with things such as calculations which, if used correctly, can provide an extremely unfair advantage for NPC's when coupled with AI. Since natural reactions and complex human thought or "intelligence" is basically the one thing that human players have over the computer players, if NPC's are given the same intelligence as well as having their initial computational advantages, the game quickly becomes unfair if not impossible for players. This is why it is important to not only get AI running in the NPC's but also it is important to not go overboard with the NPC's skills.

Workarounds and cheats can be allowed and what very often occurs, especially in terms of shooting games, is that the agents or bots have to be toned down in order to provide fairness for human players. NPC's can have skill such as advanced path finding, unfair ease of ability to find and aim at opponents etc. since in some cases it can be allowed to have direct access to some information that the player cannot access. For example, if allowed, they can ask the game engine for the exact position of the player, which is something that usually players cannot do, unless provided with a map where other players are shown. In the most realistic FPS games, the bot does not need to do calculations to predict because the bullets fly almost instantaneously, and there is a very simple solution to aiming which involves using nothing more than vector algebra. When an enemy or a target is recognized and spotted, a bullet is sent traveling towards it. The bot also needs to be aware of its weapon such as knowing how much damage it makes, if it's loaded or not etc. Additionally, if their aim has been correctly executed and has not been altered to make mistakes, it will almost always be perfect which is probably an impossible mission for human players in most cases.

## Realistic

It may not matter how perfectly AI is built into NPC's or how well they are able to act like a human, if their moves and actions are not realistic. This is another challenge that had to be faced, making the characters look real and move in a realistic way. This is a different part of the issue but still important to note and keep in mind. If an NPC is taking one millisecond to aim perfectly and moving without any pause then the human players may have a difficult time really getting engaged in the game. The point of using AI in these games is partly to create a challenging opponent but also to create the illusion of real people and players as opponents. If these two criteria are not thought about in advance then the game will probably not be too successful, and considering the amount of competition there is out there, it's probably not wise to take any chances.

## Final notes

Using tutorials and looking at codes that were already made by the Pogamut platform of basic players, it was easy to see where the initial challenges above are stemming from. Amongst the tutorial bots there was the Empty bot, which serves mostly at showing the main methods of a bot. A reactive bot, a navigation bot and finally a hunter bot. Looking at the reactive bot really helped show the importance of creating a reaction within NPC's to their environment, and similarly the navigation showed the importance of having a good navigation system.

Finally the hunter bot showed how these two could be coupled together in order to make a very impressive hunter player. However, the hunter lacked the "survival instinct" mentioned above so they did not have many tactics employed to protect themselves, but rather to kill others. Although the initial objective wasn't carried out as planned, it still helped to build a basic understanding of the many difficulties that programmers are faced with in the building of an artificially intelligent bot.

There are so many different ways to solve these problems nowadays, and of course it is much easier now to create an AI bot than it was 10 years ago. Nevertheless, a lot of debugging and thought goes into them as they have to be prepared for every situation, or at least have a possible reaction to unexpected events. Games are hard to predict, even if the basics are clear, especially when preparing a bot to play against a human. Humans develop strategy skills, and navigation skills usually automatically throughout their lives and if an AI bot does not have at least close to similar skills then they will not be very effective in creating a realistic gaming environment for the human players of the game.

## Sources

[http://en.wikipedia.org/wiki/Artificial\\_intelligence\\_\(video\\_games\)](http://en.wikipedia.org/wiki/Artificial_intelligence_(video_games))

<http://www.peachpit.com/articles/article.aspx?p=102090&seqNum=8>

<http://research.microsoft.com/en-us/projects/ijcaiigames/>

<http://www.giantbomb.com/ai-bots/92-597/>

<http://diana.ms.mff.cuni.cz/main/tiki-index.php>

<http://www.emunix.emich.edu/~evett/GameProgramming/LectureNotes/FPS.pdf>

<http://www.enterprise-europe-scotland.com/sct/services/enquire.asp?id=11%20PT%2065BN%203KL7&EnquiryType=BBS>

[http://wiki.beyondunreal.com/Legacy:First\\_Person\\_Shooter\\_AI\\_Architecture](http://wiki.beyondunreal.com/Legacy:First_Person_Shooter_AI_Architecture)

AI Game Development – Synthetic Creatures with Learning and Reactive Behaviour by Alex j Champandard

## One page Report

T-622-ARTI, Introduction to Artificial Intelligence, Spring 2011

Final Project Short Report

1. Title: Thoughts about use of AI in video games
2. Author(s): Una Benediktsdóttir



3. 

4. The initial idea was to create an artificially intelligent bot in Unreal Tournament capable of traveling and navigating by himself using A\*. It developed into more the idea of use of AI in shooting games and a bit of experimentation with it using pogamut and UT2004 in order to become more familiar with the ideas.

b) Area of Research: AI in shooting games.
5. Building on existing code used firstly to try to make our own bot but then more of a playground for experimenting with basic AI bots within Unreal Tournament in order to discover what it is behind the complicated aspects that drives it. It helped to see some of the main problems, especially since the codes were not too complicated and the bots were not extremely "intelligent" so it helped to see the basic issues that one would encounter when beginning to make bots. For research most information came from existing research on the topic.

Main Sources: Main sources were the internet (of course) and a book called AI Game Development, Synthetic Creatures with Learning and Reactive Behaviour by Alex j Champandard

6. As stated earlier the approach began by working with tutorials in order to learn how to code our own bot using A\* search navigation. This proved to be difficult but nevertheless it did teach me the basic problems that programmers face when attempting to make good AI for shooting games.

Then the project moved on to more research on the subject and most of the results obtained through the research supported the problems that I had been facing when trying to code something respectable for unreal tournament.

The ideas researched were mostly about the overall problems with discovering terrain, teamwork, reaction and balancing out hunter and prey “instincts”.

There are plenty of problems that arise when trying to make an AI for games, and shooting games are no exception. As a novice it can prove extremely difficult to make anything satisfactory enough to really be referred to as artificially intelligent. The attempts made in this project were no exception of this and eventually more research on the subject had to be done in order to really understand what is needed. Since coding without understanding of the objective can render projects almost impossible, or at least not as effective as they would be with this understanding, the paper focuses a bit more on the general issues that seemed to come up within the readings and issues that were noticed throughout experimentation.

7. In terms of the initial project, it worked both well and not so well. It was difficult to get the code to work in the way wanted initially, and eventually I found out that A\* search didn't even need to be coded since it was already a built in function with pogamut. So in terms of the initial objective, a bot with A\* was made using tutorials and building on existing code, but it wasn't extremely impressive since the actual path finding algorithm (which would have been the main challenge) was already made for me.

However, the second objective, which was to get more familiar with the making and problems of AI in games, and in particular shooting games was pretty efficiently executed and I definitely learnt a lot of new things. It was also definitely an experience in which you read something and you think, “Oh, wow, of course, I can't believe I didn't take that into consideration”. As it turns out, before coding something that is supposed to show artificial intelligence, you need to get your own intelligence on the subject in order.