Artificial Intelligence in video games

Xabier Lekunberri

Iker Ortiz

Summary

History of AI in video games

Artificial intelligence have a long history on video games. A lot of research on AI for games is about constructing agents for playing games, with or without a learning component. This has been the first and, for a long time, the only approach to using AI in games.

The first software that mastered a game was programmed by Alexander Shafto Douglas in 1952. It was a digital version of the Tic-Tac-Toe game. A few years later, Arthur Samuel was the first to invent the form of machine learning that is now called reinforcement learning (a program that learn playing against itself).

For decades Chess was seen as “standard”, because of that, uncountable new AI methods were tested on. The first software that exhibited superhuman Chess capability was the IBM’s Deep Blue. It consisted on a Minimax algorithm with numerous Chess-specific modifications and a very highly tuned board evaluation function running on a custom supercomputer. For comparison, today, it is possible to download public domain software plays better than any human player when running on a regular laptop.

Image 1: Deep Blue vs Gary Kasparov (1997)

Other uses of AI in video games have come to be very important as well. One of these is procedural content generation. Starting in the 1980s , certain video games created some of their content algorithmically during runtime, instead of having it designed by humans. Two games that became very influential were *Rogue*, where dungeons and the placement of creatures and items in them are generated every time a new game starts, and Elite. The influence of these games can be seen in recent successes such as Diablo III, No Man’s Sky and *Bloodborne*.

Why videogames are so appropriate for AI

There are some reasons for the video games to be so appropriate for using AI in them, these 4 being some of the most important ones.

Difficulty and interesting problems of games

Many people like videogames due to their effort and skills to complete or solve them in case of puzzles. It is that difficulty the one which encourages to use AI in them, as they operate in a finite space and the agent/s in them has/have many possible strategies to follow.

The way to measure the level of accuracy of AI in this field is through some milestones games. A common game to measure it has always been Chess and sometimes Checkers. Some computers as Deep Blue and Chinook got to play in a very competent way, and since then, the level of the algorithms used to learn playing Chess have just but improved.

Nowadays one of the most dauntless games to programme for is Go, having been researched for decades. One of the complexities of this game is the number of state, nothing less than 10^{170} states. Even so, *AlphaGo*, developed Google, won against Lee Sedol for a price of 1 million $. Moreover, in May 2017, it also won against Ke Jie, the number 1 ranked player three times in a row.

Image 2: AlphaGo vs Ke Jie (2017)

Human-Computer Interaction

Video Games are at least at this time one of the better examples of human-computer interaction, as it usually gives the player a plenty of possible actions to perform at any given time, which can change the environment. Anyway, the range of action that can be done vary depending on the game, as a Sandbox game (*Grand Theft Auto*) where the world is enormous and the player's possible actions are nearly unlimited differ much from a Sidescroll game (*Super Mario Bros*.).

Obviously, every game features a player (or more) and the interaction between the game and the player is the one of the most important area to examine in order to learn from the player's stimuli and emotional manifestations (e.g. having a fright in a horror game and getting sudden input from controllers).

Popularity of games

Video games were first introduced in the society as some video consoles (or arcades) in just some locals or some people who could afford one. Nowadays it is a multi-billionaire industry, generating more revenue than other creative industries combined as film and music.

One of the main reasons for videogames to become so popular was the ability to motivate the player by offering many ways to interact with it, and over the last 4 decades the technological advances have made possible to play in any platform (as PCs, video consoles, smartphones...) anytime, as the portability of many of the platforms. One example of the huge popularity of videogames was *Pokémon Go*, a game for smartphones of the extremely well-known franchise Pokémon published in the summer of 2016, which encouraged millions of people around the globe to play it and going for a walk while playing it.

As more people play more games, more content is required to satisfy the need of game mechanics more innovative. This massive demand for new gaming experiences from a huge community constantly makes software developers to improve human and computational creativity, therefore improving AI as well.

Cover of any AI field by many challenges

Video games have proven to be worth of using any kind of Ai in them, as taking some areas of AI and trying to find a challenge of that field in games is done with ease. Some major milestones were achieved thanks to Crackers, Chess, Jeopardy!, Go and Arcade games:

- Go and Arcade games helped on machine learning
- Tree search turned out to be very useful in Checkers and Chess.
- Jeopardy! was a more complex game than the others, but the advances in knowledge representation, reasoning and NLR (Natural Language Processing) made it possible to play properly.

While machine learning is waiting for improving in real-time strategy video games such as *StarCraft*, NLP is being tested and optimized in storyteller video games, where an interactive story is trying to be developed.

As far as planning and navigation problems are concerned, algorithms of high complexity are being used (e.g. games as *StarCraft*). An advantage to keep in mind over a handmade environment for robots is that video games offer a realistic environment to test too, although it remains cheaper than a handmade one.

Image 3: StarCraft II (2010)

Why AI fits so well in videogames

As video games can improve AI, AI can help in designing better video games as well. These following reasons will explain why:

AI Plays and Improves Your Game

By just playing the game, AI can add much value to the game and the game industry is pleased with its results, even if it is based on a simple behaviour tree, a few heuristics or a powerful machine learning algorithm as long as it achieves its main purpose.

AI in games has 2 main goals: To play well and to play the most human-like as possible. Moreover, AI is able to control the player character or a non-playable agents.

- AI that plays well as a player character can be of huge importance for automatic game testing and for the evaluation of the game design.
- AI that plays well as a non-player character, instead, can dynamically adjust the difficulty and game-balancing mechanisms that will enhance the experience for the player.
- AI that play human-like can serve as player experience for difficulty adjustments.
- A game that has interaction with NPCs can only benefit from AI if the controlled NPCs are expressive have a human-like behaviour.

More Content, Better Content

There are several reasons for game designers and developers to be interested in AI.

On the one hand, most typical reason is memory usage. A good example is the classic space-adventure game Elite from 1984, which managed to keep hundreds of star systems in few tens of kilobytes of memory available on the hardware.

Image 4: Elite (1984)

On the other hand, content generation is a very useful tool, as new content is generated with sufficient diversity, quality and quantity, then it may become possible to create endless games. Obviously, games that have a powerful content generation systems can even build an entire marketing campaign on content generation like No Man's Sky from the year 2016, whose AI can theoretically build more than 10^{19} different planets.

Player Experience and Behavioural Data Analytics

The use of AI for the understanding of player experience can drive and improve can make much easier and faster the design of games. Depending on the genre of the game, player's emotions, such as fear and stress, frustration, anticipation help to improve the player experience and personalized adjust to each player. Further, as a direct consequence of better and faster design, the whole game development process is sped up and improved as well.

Any AI-informed decisions about the future of a game's design or development are based on evidence rather than intuition, which shows the potential of AI via game analytics for better design, development and guaranteeing quality.

In the end, we could say that game design with AI and data analysis means better games.

Learning methods / Types of AI

There are three popular class of AI methods for game development. Finite state machines, behaviour trees and utility-based AI. Their dominance is evident by the fact that the term game AI in the game development scene is still nowadays synonymous with the use of these methods.

Finite State Machines

A Finite State Machine is the AI method that dominated processes of AI in games until the mid-2000s. Finite State Machines belong to the expert systems area and are represented as directed graphs. The graph contains states as nodes and transitions as edges. A Finite State Machine can only be in one state at a time.

Finite State Machines are very simple to design, implement and visualize. However, they can be extremely complex to design on a large scale and are computationally limited to certain tasks within game AI. An additional critical limitation of Finite State Machines is that they are not flexible and dynamic. After their design is completed and tested there is limited room for adaptivity and evolution.

Tree Search

A Tree is an expert system which contains transitions between a finite set of states. The strength of Trees compared to Finite State Machines is their modularity. Compared to Finite State Machines, Trees are more flexible to design and easier to test, however, they have same problems.

The functioning of a Tree it is simple. We start from the root and then activate the execution of the children. A child may return three different values to the parent node: run if the behaviour is still active, success if the behaviour is completed, failure if the behaviour failed. Trees are composed of three node types: the sequence, the selector, and the decorator.

- Sequence: If one child succeeds, the parent continues to the next child, and when there is no more children, if all child succeed, the parent also succeeds.
- Selector: There are two type of this node: Probability and Priority. On the probability ones, each of the children has one probability of being chosen. On the other hand, the priority ones, are ordered based on their priority and executed in this order.
- Decorator: It may execute more than one time each child, or for a specific period of time.

Utility-based AI

Utility in games is a measure of the "goodness" of a choice when playing a game. It can be viewed as a function that is able to assist a search algorithm to decide which path to take. This utility function takes characteristics of the search space and returns information about the "goodness" of areas in the space. Essentially, a utility function is an approximation to the solution we try to find. The utility functions are also known as heuristic functions, and there are three main types of functions that finds one solution. Hill climbing, Beam search and Best first.

- Hill climbing: It is an algorithm that starts with a random solution and iteratively tries to find a better solution. This kind of algorithms are very fast reaching their objectives, but in the other hand, they get stuck in local maximum, couldn't been able to reach the global optimum.
- Beam search: This kind of algorithms are basically identical to the Hill climbing ones, but in this case, they search an arbitrary number of best nodes instead of only one.
- Best first: As the name says, calculates the best option of the ones that are available at each iteration and takes its path.

Bibliography

- Game based learning how to delight and instruct in the 21st century
- Artificial Intelligence
- Brain_Evolution of Artificial Intelligence In Video Games
- Game AI Book
- Early history of video games
- Museum of play, video game history and timeline