УДК 004.8

Liudmyla M. Magas

Michael Ilchenko

ARTIFICIAL INTELLEGENCE IN COMPUTER GAMES

Vinnytsia National Technical University

Abstract

The article describes general principles in game AI development. Several algorithms and methods of AI behavior are presented.

Keywords

Artificial intelligence; AI; Game development.

If you have ever played a video game, you have interacted with artificial intelligence (AI). Regardless of whether you prefer race-car games like *Need for Speed*, strategy games like *Civilization*, or shooting games like *Counter Strike*, you will always find elements controlled by AI. AIs are often behind the characters you typically don't pay much attention to, such as enemy creeps, neutral merchants, or even animals. But how does AI found in gaming relate to the AI that tech giants talk about every day?

Recently Elon Mask has warned the world that the fast development of AI with learning capability by Google and Facebook would put humanity in danger. Such argument has drawn a lot of public attention to the topic of AI. The flashy vision AI described by these tech giants seems to be a program that can teach itself and get stronger and stronger upon being fed more data. This is true to some extent for AI like AlphaGo, which is famous for beating the best human Go players. AlphaGo was trained by observing millions of historical Go matches and is still learning from playing with human players online. However, the term "AI" in video game context is not limited to this self-teaching AI.

The implementation of AI greatly affects the gameplay, system requirements and the budget of the game, and developers balance between these requirements, trying to make an interesting and undemanding to resources AI at a low price. Therefore, the approach to game AI is seriously different from the approach to traditional AI - various kinds of simplifications, deceptions and emulations are widely used. For example: on the one hand, in first-person shooters, the unmistakable movement and instantaneous targeting of bots leave no chance for a human, so these abilities are artificially reduced. Rather than learn how best to beat human players, AI in video games is designed to enhance human players' gaming experience. One of the AI lead developers of Russian game studio Nival said: *The main task of AI is not to win from the player, but to give it up nicely.*

The most common role for AI in video games is controlling non-player characters (NPCs). Designers often use tricks to make these NPCs look intelligent. One of the most widely used tricks, called the Finite State Machine(FSM) algorithm, was introduced to video game design in the 1990s. In a FSM, a designer generalizes all possible situations that an AI could encounter, and then programs a specific reaction for each situation. Basically, a FSM AI would promptly react to the human player's action with its pre-programmed behavior. For example, in a shooting game, AI would attack when human player shows up and then retreat when its own health level is too low.

An obvious drawback of FSM design is its predictability. All NPCs' behaviors are pre-programmed, so after playing an FSM-based game a few times, a player may lose interest.

A more advanced method used to enhance the personalized gaming experience is the Monte Carlo Search Tree (MCST) algorithm. MCST embodies the strategy of using random trials to solve a problem. This is the AI strategy used in Deep Blue, the first computer program to defeat a human chess champion in 1997. For each point in the game, Deep Blue would use the MCST to first consider all the possible moves it could make, then consider all the possible human player moves in response, then consider all its possible responding moves, and so on. You can imagine all of the possible moves expanding like the branches grow from a stem– that is why we call it "search tree". After repeating this process multiple times, the AI would calculate the

payback and then decide the best branch to follow. After taking a real move, the AI would repeat the search tree again based on the outcomes that are still possible. In video games, an AI with MCST design can calculate thousands of possible moves and choose the ones with the best payback (such as more gold). A similar algorithm has also been applied in many strategy games. However, since the possible moves are much more than in chess, it is impossible to consider all of them. Instead, in these games the MCST would randomly choose some of the possible moves to start with [1].

In games in which the creative potential of a player is important, AI can not fight on an equal footing with a human. To equalize the chances, developers apply cheating, or deceptive AI.

Deceptive AI compensates for the lack of strategic thinking with any other advantages over the player. For example: more lives, faster movement or ignorance the fog of war.

Of course, a computer always has an advantage over a person - a person has to rely on vision and hearing with their limitations, while the computer has direct access to the engine abstractions. A "true" game AI should have and use visual processing algorithms, but analog of human vision is now an unreachable goal for computer vision systems.

There is one general example of cheating game AI below, which is present in many racing games. If the AI player is far behind most riders, he suddenly gets a huge speed increase or other parameters that allow him to catch up with other riders and again become a competitive rival. This method is known as Rubber banding or Catch-Up, as it allows the AI character to immediately return to a competitive position. In more advanced games, the competitiveness of non-player characters or bots can be achieved through dynamic game balancing, which can be considered fairer, although still a technical fraud, as AI players still gain advantages, even though they follow the rules of the virtual world [2].

However, some game developers are against of cheating AI in games. Earlier they used complicated algorithms for better fair AI, but nowadays new methods are implemented in the development of AI. The great example is the strategy game *Blitzkrieg 3*, the first game ever which uses neural networks for enemy behavior. This neural network is called Boris and here are his main features: usage of fair data; usage of scouting; skillful dodging; clever usage of terrain (covering himself); valuation of situation and clever respond.

A glimpse of the future comes from Michigan State University, where researchers have deployed AI into a game specifically to learn from each player's behavior. *We use Darwinian evolution to optimize the AI while the game is being played, which hopefully leads to arms races between players and AI, which will present players with new challenges all the time*, the researchers say [3].

Artificial intelligence and gaming are rapidly becoming symbiotic. While it has always operated at the cutting edge of tech to make better games, game theory is also contributing to better AI practice.

REFERENCES

1. AI in Video Games: Toward a More Intelligent Game [Electronic resource] – Access mode: http://sitn.hms.harvard.edu/flash/2017/ai-video-games-toward-intelligent-game/
2. Suryakumar Balakrishnan Nair Learning LibGDX Game Development 2015 // Birmingham-Mumbai: Packt Publishing, 2014 500 p.: ISBN-13: 978-1783554775
3. How artificial intelligence is changing game industry [Electronic resource] – Access mode: https://unbabel.com/blog/ai-changing-gaming-industry/

Liudmyla M. Magas — FLD Teacher, Vinnytsa National Technical University, Vinnytsia.

Michael Ilchenko — student, group 1CS-14b, Faculty of information technology and computer engineering, Vinnytsa National Technical University, Vinnytsia.