Artificial Intelligence in Gaming

Mayuresh Anant Shirke[1] Aniket Abaji Rankhambe[2] Prof. Swapna Nikale[3]
[1,2,3]Department of Information Technology
[1,2,3]B. K. Birla, India

Abstract— Computer games are increasingly popular application for Artificial Intelligence (AI) research, and conversely AI is an increasingly popular selling point for commercial games. Although games are typically associated with entertainment, there are many serious applications of gaming, including military, corporate, and advertising applications. There are also so called “human” gaming applications for medical training, educational games, and games that reflect social consciousness or advocate for a cause. Game AI is the effort of going beyond scripted interactions, however complex, into the arena of truly interactive systems that are responsive, adaptive, and intelligent. Such systems learn about the player(s) during game play, adapt their own behaviour beyond the pre-programmed set provided by the game author, and interactively develop and provide a richer experience to the player(s). The long-term goal of our research is to develop artificial intelligence techniques that can have a significant impact in the game industry. In this paper, we present a list of challenges and research opportunities in developing techniques that can be used by computer game developers. We discuss three Case Based Reasoning (CBR) approaches to achieve adaptability in games: automatic behaviour adaptation for believable characters; drama management and user modelling for interactive stories; and strategic behaviour planning for real-time strategy games.

Keywords: Case Based Reasoning (CBR), Artificial Intelligence

I. INTRODUCTION

AI stands differently for different people. The definition of AI varies according to each individual perspective. For some, AI is artificial life which can cross human intelligence, and may cause the destruction of human species and for others, AI is almost any data processing technology.

If you have ever played any video game, you have interacted with artificial intelligence (AI). Regardless of whether you prefer racing car games like Need for Speed, strategy games like Grand theft Auto, or shooting games like P.U.B.G., you will always find elements controlled by AI. AI is always involved in these small small things. AIs are often behind the characters you typically don’t pay much attention, such as enemy creeps, neutral keepers, or even animals.

Most people probably imagine that the majority of games released in the last couple of years have highly complicated A.I. for any non-player controlled characters, creatures, or animals. But in reality, many video game developers are hesitant or afraid to build advanced A.I. into their games in fear of losing control of the overall player experience. In fact, the goal of A.I. in video games is not to create an unbeatable character. Where no matter how much players put an effort to defeat enemy the character will not get defeated, but is instead meant to increase player participation and joy of playing over long period of time.

If you start play a new gaming will you able to enjoy it by getting destroyed each time? Or would you rather be paired against someone/something more your level so that you will put your best effort and able to defeat it and learn new things? The majority of players would likely choose the later. That doesn’t mean there is no place for A.I. in the modern gaming industry, it just means that it’s purpose is different from what we would initially expect. We don’t want to create the best possible A.I which is unbeatable., we want to create the most enjoyable A.I. for players to interact/compete with which will not get players bored.

II. USES OF VIDEO GAME A.I. IN THE REAL WORLD

The same type of reinforcement learning that is being used in the video game industry is also being successfully applied to other industries. For example, the Grand Theft Auto games. The game have pre-programmed “traffic rules, roads, and car physics” (Luzgin, 2018) have been used to provide a safe and realistic environment for testing self-driving car algorithms. Not only is it safe and realistic, but it is also up to 1,000 times faster to collect data in a virtual environment compared to the real world (Luzgin, 2018).

“Video games are a great way of training AI algorithms because they are designed to give human minds timely progression into harder and harder challenges.” (Luzgin, 2018)

One of the latest advancements of A.I. in video games was made by researchers at Open AI. Open AI created a game based on an algorithm whose sole purpose was simply to explore with a sense of natural curiosity. The reward system focused on rewarding exploration over progressing further into the game. The researchers placed this curiosity-driven model into a game of Super Mario Bros. and it successfully passed 11 levels out of pure curiosity. Obviously, there are downsides to this, as it takes immense computing power and the machine can get easily distracted. However, this would also be the same for a human player playing the game for the first time.

“There are many forms of A.I. in use across the video game industry today. Whether it’s a simplistic FSM model or an advanced neural network learning from feedback in it’s the environment, the possibilities that these virtual environments provide to the advancement of A.I. (both in the gaming industry and beyond) are endless.”

A. Playing against an AI:

Recently Elon Musk has warned the world that the fast development of AI with learning capability by Google and Facebook is going to put humanity in danger. This types of statements and arguments have drawn the public attention on 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 is trained by observing many no. of historical Go matches and it 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.

Rather than learn how best to beat human players, AI in the games is designed for increasing the user experience and joy of playing the game. 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 according to the situation he programs specific reaction. Basically, a FSM AI would promptly react to the human player’s action with its pre-programmed behavior. For example, if you are playing a shooting game AI will attack only when a player shows up and will fall back when it’s health gets low. In FSM-oriented game, in response to possible situation, a character can perform 4 basic operations: aid, evade, wander and attack. Many famous games, such as Battle Field, Call of Duty, and Tomb Raider, incorporate successful examples of FSM AI design. Even the turtles in Super Mario have a rudimentary FSM design.

B. Requirements for Game AI

In previous work, Laird and van Lent (2000) analysed different games of different genres, and the challenges of AI that each game presents. In their report, they considered the following types of games: action, role playing, adventure, strategy games, god games, individual and team sports games. In addition to those genres, we would like to consider two additional categories, namely, interactive drama (Mateas & Stern 2003) and educational games (Rieber 1996). Interactive dramas have a strong plot behind them that the author wants to communicate to the player, but where the player may have a strong influence on the plot. A key difference with the classical “adventure” genre is that adventures have a scripted plot, while interactive dramas are more open-ended and adapt to interaction of the player as the story unfolds. Educational games have an additional rhetorical goal of teaching some particular content to the player which may or may not be new to know.

By analysing the possible range of application of computer game AI to different applications and game genres, we identify two different levels at which we can apply AI: 1) individual characters AI, with the goal of producing more intelligent or believable behaviours, and 2) a global AI that watches over the game or game-player interaction, influencing the directions that the game is taking. Thus, we can talk about character-level AI and game-level AI (the second being referred in some papers as the Drama Manager (Nelson et al. 2006a) or as the Director (Magerkoet al. 2004)). Different applications and game genres require a different mix of these two kind of AIs. For instance, real time strategy games rely mainly on a game-level AI that controls all the units, while the individual unit behaviours can be scripted. Role playing games, on the other hand, require believable character-level AI to provide an interesting player experience. Interactive dramas requires a mix of both kinds of AI: individual characters that are believable and a drama manager that leads the plot by guiding the individual characters to take actions that can make the drama advance. Educational applications of gaming also require a game-level AI, similar to the drama manager, that monitors the interaction of the game as it unfolds, easing or complicating the tasks according to the learner’s expertise level, thereby making sure that educational purpose of the game is being met.

Each game genre presents particular requirements for character level and game level AI. For instance, god games usually require the game-level AI to solve resource allocation problems and solve long-term strategy problems, while interactive drama requires the game-level AI to adapt the story according to the player interactions in a way that it is more appealing to the player (thus, the latter requires user modelling and story planning). Moreover, adventures, interactive dramas and other genres with embodied characters usually require believability and natural language generation. In the following section, we have described a list of interesting challenges that computer games pose in general to the AI community.

C. Challenges in Computer Game AI

While developing Artificial Intelligence for Computer games, the issues arrived are briefly described as follows.

- Complex decision spaces: most state-of-the-art computer games involve complex strategic (real time strategy games) or believable behaviours (interactive dramas). Both kind of behaviours share the characteristic of having huge decision spaces, and thus traditional search-based AI techniques cannot be applied. Learning techniques or higher-level representations are required to deal with such complex games. Traditionally, computer games use handcrafted strategies coded by the game developers, but these tend to be repetitive, and players easily find holes and exploit them.
- Knowledge engineering: even assuming that strategies or behaviours are handcrafted, authoring these behaviour sets in a game requires a huge human engineering effort. Game developers have to encode all the knowledge they have about a domain (either to achieve a strategic behaviour or a believable human behaviour) in some sort of behaviour language.
- Authoring support: hand crafted behaviours are, ultimately, software code in a complex programming language, prone to human errors. The behaviour errors could be in the form of program “bugs” or not achieving the desired result. Tools are needed to support story authors, who are typically not artificial intelligence experts, to author behaviours in a computer programming language.
- Unanticipated situations: it is not feasible to anticipate all possible situations and player strategies that can encountered during game play. This makes it difficult to craft believable behaviours that react in an appropriate manner to these unforeseen circumstances and player actions.
- User-specific adaptation: different players may enjoy different strategies to fight against (in the case of real time strategy games), or different styles of story-telling (in the case of interactive dramas), different types of

story development, different kinds of character behaviours and interactions, or different educational problems. As game designers begin to include user modelling capabilities, the AI strategy and behaviour must, in turn, be adaptable based on the user model.

– Replayability and variability: a player might get bored of seeing the same strategies and behaviours again and again. Although simple variability can be achieved through stochastic selection of behaviours or strategies from a large repository, this increases the authoring burden. Furthermore, random selection begs the question of true interestingness.
– Rhetorical objectives: it is possible, even likely, that human-engineered behaviours or strategies do not achieve the game's objectives adequately, especially in realistic, scaled-up domains or applications. These objectives could range from entertainment to education, training, etc. Thus, the game has to realize that the objectives are not being met on a per-use basis, and adapt accordingly. For example, a particular user may be getting bored, or not learning the intended lesson.
– A conclusion that can be drawn from the previous list is that not only can games benefit from better AI techniques (by AI games improve a lot), but AI can also benefit from the challenges that computer games provide. In the remainder of this paper, we will present three research projects that contribute to achieving a subset of these challenges. The three projects have the same underlying theme of (a) easing the effort involved in developing computer games AI and (b) making them more adaptive and appealing to the player.

The first project focuses on a framework for runtime behaviour adaptation that aims at removing the responsibility of the authors to foresee all possible situations that can be encountered during the game. The second project aims also at reducing the effort of defining strategic behaviours for real time strategy games by extracting behavioural knowledge from experts using a system that can automatically learn new behaviours from example of game play sessions. Finally, the third project aims at making the games more adaptive to the players by using player modelling. The approach predicts which sequence of events will be more appealing for each player and influences the game accordingly through the drama manager.

III. TYPES OF ARTIFICIAL INTELLIGENCE

Arend Hintze, an assistant professor of integrative biology and computer science and engineering at Michigan State University, have categorized AI in four types, from AI systems which exist in present and AI systems which are yet to exist. The following are the different categories of AI.

A. *Reactive Machines:*

An example is Garry Kasparov in the 1990s got beaten by Deep Blue, the IBM chess program. Deep Blue can identify pieces on the chess board and make predictions, but it has no memory and cannot use past experiences to inform future ones. It analyses possible moves -- its own and its opponent -- and chooses the most strategic move. Google's AlphaGO and Deep Blue are designed for narrow purposes and cannot be easily applied to another situation.

B. *Limited Memory:*

From the experience which we got from past decisions this type of AI help in future situations. In self-driving cars some of the decision making functions are designed this way. Observations inform actions happening in the not-so-distant future, such as a changing of the car lane from one to another. Also the observations are not stored permanently in the system.

C. *Theory of Mind:*

This psychology term refers to the understanding that others have their own beliefs, desires and intentions that impact the decisions they make. This kind of AI does not yet exist because its very advanced and our technology is not that developed yet.

D. *Self-Awareness:*

In this type AI have awareness about self and consciousness. In this type AI helps machine to understand its current state and can use the information to infer what others are feeling. This is very advanced type of AI t does not yet exist.

IV. CASE BASED PLANNING FOR STRATEGY GAMES

AI techniques have been successfully applied to many computer games such as checkers, chess or Othello. However, because of the characteristics of the vast search spaces many games require, in some computer games traditional AI techniques fail to play at a human level. For that reason, game developers need to invest significant effort in hand-coding specific strategies that play at a reasonable level for each new game.

For instance, previous research has shown that real time strategy games (RTS) such as Wargus (a clone of the popular commercial game Warcraft II) have huge decision spaces (Aha, Molineaux, &Ponsen 2005; Buro 2003). In this section we present an architecture that uses case-based planning (Hammond 1990) to deal with such complex games.

In previous work, we have experienced with applying case-based reasoning (CBR) to RTS games (Sharma et al. 2007a). The idea there was to define a set of high-level actions, and let a CBR system learn when each of the has to be applied. In this section, we discuss a different approach which addresses the complexity of this domain by extract behavioural knowledge from expert demonstrations (i.e., an expert plays the game and our system observes). Then, at performance time, a case-based planning engine retrieves suitable behaviours observed from the expert and adapts them to the current game state. One of the main contributions of this approach is that it enables the game developers to specify the AI behaviour just by demonstration, i.e., instead of having to code the behaviour using a programming language, the behaviour can be specified simply by demonstrating it to the system. If the system shows an incorrect behaviour in any particular situation, instead of having to find the bug in the program and fix it, the game developers can simply demonstrate the correct action in the particular situation. The system will then incorporate that information in its case base and will behave correctly in the future.

A. Case-Based Behaviour Learning in Wargus

Figure 2 shows an overview of the process used to learn behaviours from expert demonstrations. The process is divided into two stages:

- Behaviour acquisition: Where a set of cases are extracted from an expert trace.
- Execution: Where the cases extracted are reused to play the game.

The first step in the process involves an expert providing a demonstration to the system of how to play the game. As a result of that demonstration, the system obtains a game trace consisting of the set of actions executed during the game. The next step is to annotate the trace. For this process, the expert uses a simple annotation tool that allows him to specify which goals was he pursuing with each particular action.

Next, as Figure 2 shows, the annotated trace is processed by the case extractor module, that encodes the strategy of the expert in this particular trace in a series of cases. A case stores a sequence of actions that an expert used in a particular situation to achieve a particular goal. Notice that from a single trace many cases can be extracted. For instance, if the expert destroyed multiple towers of the enemy, we can collect multiple cases on how to destroy a tower.

Once the cases have been extracted from the expert demonstration, the system is ready to use them in actual game play. During performance, the interleaved planning and execution (PE) module keeps track of the open goals. For each open goal, the PE module sends the goal and the current game state to the CBR module. In response, the CBR module selects a case from the case base that suits the goal and game state. That case is adapted to match the current game state (adjusting which units make which actions, and on what coordinates, since in the new map the stored coordinates in the case might not make sense). Once adaptation has taken place, the adapted behaviour is sent to the PE module.

The PE module keeps track of the behaviours that are being executed for each goal, and it ensures that each sub goal of each behaviour is also satisfied by maintaining an execution tree of goals and sub goals. In addition, by using the alive conditions of the behaviours, it monitors whether a particular behaviour is still alive, and whether it is worthwhile to continue executing it. Each time a behaviour is finished (or canceled), the PE module checks whether the goal that behaviour was pursuing has been achieved. If it has not been achieved, then another behaviour must be created by the CBR module to satisfy the goal.

Future Work We plan to incorporate learning during performance by retaining those adapted behaviours that succeeded when applied. This will enable the system to learn with experience. The goal is to use the behaviours extracted from the expert as the starting point, and slowly improve the behaviour library with experience. At any time, if the developers of the game sees that the system is unable to succeed at a particular situation, the demonstration from an expert for that situation can be provided.

V. CONCLUSIONS

In this paper, we discussed a set of challenges that state the-art of computer games pose to the artificial intelligence community. Developing AI techniques that can deal with the complexity of computer games is a big challenge, but has the potential to have a big impact in several areas including entertainment, education and training.

Our main goal is to develop AI techniques that can ease the effort of incorporating AI in computer games to make them more adaptive and appealing to the player. We call such games adaptive games and also to create an AI which is enjoyable rather than creating an AI which is unbeatable. We have to increase the Enjoyment of the game rather than creating a complex AI which decreases the user performance. In this paper, we introduced three of our current research thrusts aimed at creating adaptive games via the application of case-based reasoning techniques.

We believe that computer game AI will be the next revolution in the gaming industry. After the impressive advances in the audio-visual presentation the networking capabilities, the next step in computer games is to incorporate advanced AI techniques that can achieve the goal of having truly adaptive games, increasing the level of believability and immersion. To achieve this goal, the gaming community needs new techniques, approaches and tools that allow them to easily specify, develop, and incorporate AI in their games.

ACKNOWLEDGMENT

I wish to express my wholehearted appreciation and deep sense of heartfelts gratitude to my guide Prof. Swapna Nikale, Department of Information Technology, B.K. Birla College of Arts, Science & Commerce for her generous help, excellent guidance, lucid suggestions and encouragement throughout the course of this work. Her concrete directions and critical views have greatly helped me in successful completion of this work.

I am also thankful to those who helped directly or indirectly in the completion of this work.

REFERENCES

[1] https://course.elementsofai.com/
[2] http://sitn.hms.harvard.edu/flash/2017/ai-video-games-toward-intelligent-game/
[3] https://towardsdatascience.com/artificial-intelligence-in-video-games-3e2566d59c22 etc.

