(c) 2001 Australian Journal of Intelligent Information Processing Systems, 7, 61-68.

Computer Games with Intelligence

Daniel Johnson[1] and Janet Wiles[1,2]
[1]School of Information Technology and Electrical Engineering
[2]School of Psychology
University of Queensland

Abstract—Artificial Intelligence (AI) is playing an increasingly important role in the success or failure of computer games and the quality and complexity of the AI techniques employed in games is steadily increasing. The paper explores the use of AI in games focusing on difficulties faced by game developers and techniques that have proven useful. Particular attention is given to a recently released popular game that makes extensive use of AI. The paper highlights the potential benefits of collaboration between academic AI researchers and the games industry and specific focuses for potential collaboration are suggested.

***Keywords*—artificial intelligence, fuzzy logic, video games**

Introduction

IN recent years the video game industry has experienced exponential growth. Hardware changes have meant a great deal more raw-processing power is available to game-players in the home and increasingly, partly as a result of this trend, artificial intelligence (AI) is playing an important role in the success or failure of a game [3,6]

The goal of the present paper is to facilitate collaboration between academic AI researchers and games developers by highlighting the potential benefits to both parties. To this end, the paper explores the manner in which AI has been used in games, the challenges games developers face when incorporating AI into games, the relative successes and failures of different AI methods which have been applied in particular games, and consideration is given to a particular game that is an excellent example of the successful use of AI techniques to make a game more appealing to users. Finally, the paper focuses on specific potential areas of collaboration for the academic AI community and the games AI community[1].

The intended audience for this paper is people from a variety of backgrounds (particularly academic AI researchers and people from the video game industry). Thus, to facilitate understanding of domain specific terms, two appendices are provided; Appendix A defines terms associated with AI techniques and research and Appendix B provides a breakdown of game genres with brief descriptions of each category of game.

[1] Readers interested in critical reviews focusing in greater detail on specific games AI issues are directed to the reference list of this paper (specifically references 35 and 36) and also to the publications listed on the AI wisdom website (http://www.AIWisdom.com/bydate_all.html)

A Working Definition of AI

Definitions of AI have been debated since the term's inception. For some researchers it is important that the definition of AI include only those techniques that actually reflect human cognitive processes [2]. For others, the defining factor is that AI techniques are able to solve problems that would require intelligence if solved by humans, irrespective of whether the techniques employed reflect human cognition [2,19]. It has also been suggested that for a definition of AI to be applicable, the most important criterion is that the system can adapt to or learn from its environment [19].

It seems that AI has progressed to the point where it cannot be considered to be a binary concept. Rather, in practice, the term refers to a spectrum of ideas ranging from a simple system that can perform only basic tasks to a fully adaptive system that is able to solve highly complex problems by using techniques that reflect the nature of human intelligence.

It has been argued that the important criterion for successful AI in a game is that the non-player characters (non-player characters are all entities in the game which are not controlled by the user) display life-like attributes and thereby provide a challenge or an appearance of reality and intellect [6]. Thus, a working definition of AI for the purposes of the current paper need not include the ability to learn or the requirement that techniques employed reflect human processing. Rather, AI can be considered to include all techniques employed in an effort to make game elements appear 'smarter', more 'aware' or more 'lifelike'.

Changes in the Importance of AI in Games

Improvements in Graphics and Sound

In recent years the quality of graphics and sound in games has improved exponentially. The improved graphical representation of characters on screen has meant that it is far easier for the user to discern nonsensical or questionable actions on the part of non-player characters [6]. Thus, it has become far more important that the actions of game controlled characters reflect an appropriate level of intelligence.

More CPU Power Available for Implementing AI

In the past, game AI programmers often felt that they were given insufficient CPU resources to complete their tasks [25]. The percentage of CPU power devoted to AI (as

opposed to graphics) was often insufficient for any significant AI techniques to be employed [25,26]. Recent trends have led to a situation where that is rarely the case any longer. As the specifications of the average home PC increase, the same percentage of CPU processing power allows far more advanced techniques that it did three years ago. Moreover, with the introduction of the graphics processing unit (GPU) the CPU is no longer responsible for the majority of the work associated with providing visuals for a game. This development has allowed a larger proportion of CPU power to be devoted to the implementation of AI [25].

Competition in the Games Market

The games market is highly competitive and while in the past excellent graphics were enough to increase the likelihood of success for a game, in the present climate a high standard of graphics is assumed or expected [22]. Thus, the use of AI techniques has increasingly become a necessity in order for a game to stand out in the market. Moreover, as more games incorporate functional and impressive AI techniques, users' expectations and demands increase and a game that fails to employ effective AI is less likely to be well received [25].

The Use of AI in Games

In the interest of exploring the use of AI in games this section considers some of the major AI techniques, the manner in which they have been used in specific games and some of the inherent strengths and weaknesses. The techniques are divided into those that are purely 'rules based' and those that make an attempt at learning or adapting to the player's behaviour. Discussion of games in which the player can modify the AI is also undertaken.

Rules-Based Techniques

Finite State Machines and Fuzzy Logic

FSMs have often been used for controlling enemies in first person shooters (e.g., Doom, Quake) [29]. The FSM checks the environment and if, for example, it discovers that the player is in the room and that the player has not fired a weapon then it might decide to attack. FSMs remain common in games as they are familiar to games developers (particularly relative to more advanced AI technologies such as neural networks) and they are easy to test, modify and customize [25].

Increasingly, however there is a trend towards FuSMs that is perhaps unsurprising given that the use of fuzzy logic allows for recognition of non-binary conditions. FuSMs are used in the FPS game 'Unreal' to make the enemies appear reasonably intelligent. Based on elements of the battle situation, fuzzy logic is used such that enemy characters in Unreal can decide to run away when losing a battle, summon reinforcements, hide if wounded, and lead the player into ambushes [25].

Fuzzy logic has also been employed in the action game, 'S.W.A.T. 2'. In the game, fuzzy logic is used to determine the tactical responses of enemy units based not only on the situation but also the 'personality' of the enemy unit. When fired upon, an enemy unit has a variety of alternative behaviours to choose from (flee, take cover, shoot back, shoot a hostage). Rather than determine the unit's behaviour at random, the AI mechanism considers the character's pre-determined personality traits (aggression, courage, intelligence and cooperation) and depending on the various levels of these traits chooses the 'appropriate' reaction [27].

Another application of FuSMs can be in seen in 'Civilisation: Call to Power' (CCTP). CCTP is a real time strategy game in which the player will encounter a number of different cultural groups. In order to imbue each different group with its own personality, the game developers implemented cascading FuSMs. The developers built a core AI engine but changed the decision thresholds depending on the personality of the group being represented [25]. This allows for the expression of varying approaches across groups without the need for separate lengthy code for each group.

Techniques that Allow for Learning

The type of problems involved in creating an AI system that 'learns' can vary greatly across genres. In fighting games the player generally only has a choice of up to 6 buttons combined with different choices of direction. For these types of games it is feasible to create a database of all possible 'valid' button combinations as well as appropriate responses to each combination. Once the AI system notices a particular button combination is being repeated it can consult the database for the best response. In this way it is possible to create a character in a fighting game that is able to 'adapt' to the user's style of play. The simplicity of this sort of mechanism stems largely from the fact that the AI system need only be aware of the raw input commands that the player is entering. Other game genres present more of a difficulty. In a real time strategy game, for example, the raw input commands (on their own and without context) provide very little information about the player's behaviour and strategies [7] and it is impossible to create a database, such as the one mentioned above, which lists all possible player behaviours in the form of raw input commands. In strategy games, an effective AI system needs to be able to learn information such as the player's preferred style of unit or dominant method of attack. Such information can only be learned by the AI system if it is able to store knowledge of the player's behaviour at a more abstract level [7].

Neural Networks

Neural Networks can be used as a means of updating the AI system as the player progresses through the game [29]. The major advantage of a neural-network-based AI is that, theoretically, the network can improve continuously, such that the player will constantly be challenged to change their style of play and must avoid reusing the same strategy repeatedly.

Neural networks have been successfully implemented in a variety of games, including adventure games (e.g., Battlecruiser 3000AD), racing games (Dirt Track Racing), and strategy games (Fields of Battle). One interesting

example of an AI system that relies on neural networks is 'Battlecruiser 3000AD', in which every non-player character in the game is controlled by a neural network. Another example is Heavy Gear, an action game in which the player controls a large mechanized robot. According to the game developers, neural networks are employed as a means of providing the robots with a 'brain'. In practice the neural network functions by improving the skills of the robot depending on the actions the player performs (e.g., if the player shoots a particular weapon regularly that weapon may begin to reload faster over time) [27].

Although used successfully in the past, neural networks provide a particular set of challenges to game developers and as a result, to date, their use has been limited [26]. Neural networks require the specification of valid inputs and outputs, and this can be a difficult task within the context of a game. Moreover, when unsupervised learning techniques are employed in a game, there is always the chance that the neural network will be trained into a state in which it performs badly. Thus, for game developers, even if they can successfully define the elements of the network, the possibility exists that the AI system will end up directing the non-player characters to behave in an obtuse or unintelligent manner. One solution to this problem is to include the option for the player to 'reset' the neural network parameters (i.e., return the parameters to their original, default values) [25].

Evolutionary Algorithms

With the exception of Artificial Life technologies (considered below), genetic algorithms per se have not been used a great deal by game developers. Although they offer the benefits of an AI system that can evolve over time, many developers have suggested that genetic algorithms required too much CPU power and were too slow to produce useful results [25]. One noteworthy exception is the real time strategy game 'Cloak, Dagger, and DNA' (CDD). In CDD both the player and the non-player characters have an associated 'DNA' strand that monitors and stores performance. A player has the option of evolving the DNA strands by placing them in tournaments with other DNA strands [27].

Artificial Life

Traditionally, developers have reported difficulties as a result of the unpredictability inherent in artificial life techniques [25]. However, in late 1999 and 2000 a number of very successful games were released which employed artificial life techniques [26]. The value of artificial life techniques lie in the fact that they allow game developers to break down larger AI tasks into smaller sub-tasks. The core of artificial life techniques is the phenomena whereby more complex (non-explicitly programmed) global behaviours can been seen to result from the interaction of simple lower-level pre-programmed rules [26,31]. This phenomenon is referred to as 'emergence' [31]. An example is the use of 'flocking' techniques in which characters controlled by flocking algorithms simulate the movement behaviours of large groups of animals, the reasonably complex movement patterns emerge from three simple steering behaviours: separation, alignment and cohesion [31].

In the past, games that have used artificial life have tended to reflect the engines that drove them. That is, they focused on the breeding and evolution of small virtual creatures. Indeed, some of the more successful examples of this sort of game come from a series titled 'Creatures'. Games in the 'Creatures' series require the player to hatch eggs that contain 'Norns'. The 'Norns' must be raised as virtual pets (with their own needs and the ability to respond to reinforcement) and if they survive they are able to reproduce. 'Beasts', a game currently being developed by the group who created the 'Creatures' series, promises to take the technology further by incorporating dominance hierarchies, realistic mating habits and complex social structures [34].

Increasingly, artificial life techniques are being incorporated in games other than creature-evolution style games (a sub-genre of simulation games). 'Nooks and Crannies' is a real time strategy game in which a player not only breeds and evolves creatures, they then use them to fight 'battles' against other evolved creatures [34]. Even further from the traditional use of artificial life technologies, are some of the games developed by 'Maxis', including 'The Sims' and 'Theme Hospital'. 'Theme Hospital' is a simulation style game in which the player must manage a hospital. Artificial life algorithms are used to simulate phenomena such as the movements and behaviours of staff and patients in the hospital [33].

Although, artificial life techniques are being incorporated into an increasing variety of games, it has been suggested that the more complicated techniques are too processor intensive for use in games that already require a moderate amount of power to run [27]. Virtual-creature-style games have, to date, tended not to be too graphics intensive, and thus there is processing power available for implementation of the more advanced techniques. However, with the aforementioned trends of increasing power and the use of graphical processing units it seems possible that artificial life techniques will continue to be adopted across more resource intensive genres.

Extensible AI

The term 'extensible AI' refers to games in which the player can modify the existing AI implemented in the game. For example, in a first-person shooter, the player may be given the ability to determine how non-player characters in the game react to enemies. Extensible AI has proven very popular among users [20], however it can be a challenging feature to implement for game designers. Once users have the ability to change the AI programming in the game, several support and security issues arise. For example, how much support can realistically be provided, given that the ability to modify the AI may rely on programming knowledge? How do game companies prevent a situation in which a script is released which damages the user's computer? [25].

The first game to incorporate extensible AI was Quake [25,28,32] and so far the majority of games that feature extensible AI have been first person shooters (Unreal, Half

Life, Quake) [25]. In these games the user is given control over the 'bots' (non-player characters) in the game; setting parameters such as level of aggressiveness or tweaking reactions so that, for example, the bots will jump over shots fired at their feet [32]. More recently, games from other genres (e.g., real-time strategy and role playing games) have incorporated extensible AI and generally it is a feature that is well received [25,27].

Recent Innovations in Games AI : Lionhead Studios' Black and White

Black and White (BAW) has been described as the latest big step forward for the art of game development [21]. Topping the sales charts for a number of weeks [24] and receiving remarkably positive reviews from a variety of sources, Black and White has proven to be a highly popular and successful game. Much of the game's popularity and success has been attributed to the AI system implemented throughout the game [21].

Through consideration of the nature of the game and the AI techniques employed it is possible to gain insight regarding what is currently possible using AI in games and which techniques have proven successful. Moreover, consideration of BAW, a game that makes a number of significant advances in the field of games-AI, leads to the realization of previously unconsidered ideas concerning new directions and potential applications for AI in games.

Description of Game Content

Lionhead Studios (the designers of the game) describe BAW as a god game packed with strategy [30]. BAW can be considered to fall within the simulation genre. The idea of the game is that the player takes control of the destiny of an entire world. Acting as the 'god' for the world, the player must rule over and look after the people inhabiting the world ('villagers'). As the player's rule progresses the villagers' prayer and worship increases the player's power, which in turn allows them to extend their influence further. The content of the prayers and requests the villagers offer determine the 'missions' that the player can undertake (e.g., to make crops grow, to harm enemies). The player can elect whether or not to help the villagers and more broadly can determine whether they will be a cruel or a kind god [21,30].

As a god, the player acquires a 'creature' that will carry out their bidding. The creature is capable of doing anything the player can do in the game and must be taught to behave in the manner that the player desires [16]. The creature can be thought of as the player's physical manifestation in the game world. Ultimately, the player must battle other gods who inhabit the world, and the player's creature must battle other creatures [30].

Description of Some of the AI Techniques Employed and Their Impact on Gameplay

Creatures

One of the goals mentioned by BAW developers was that the creatures in the game would be both plausible and malleable (i.e., be able to learn) [1]. In order to facilitate the realization of these goals the AI programmers for the game took into consideration the nature of real-life learning, practice, and reinforcement [16].

In an effort to make the creature plausible, BAW incorporates the Belief-Desire-Intention (BDI) architecture of an agent [1], a variation of the standard artificial life model [21]. In a BDI architecture, the beliefs of an agent are its model of the domain, its desires are the manner in which the agent prioritises possible states, and its intentions are the things the agent has decided to do [17]. In BAW, beliefs about objects are represented symbolically as attribute-value pairs for individual items, and as decision trees for classes of items, and desires are represented as perceptrons. To further strengthen the plausibility of the creatures in BAW, a creature cannot form a belief about an object unless it is possible based on the creature's perceptions of the object. That is, the creatures in BAW do not 'cheat', in the sense that they cannot access any information they have not gathered from their own 'senses' [1].

The malleability of the creatures in BAW is quite extensive as they are able to learn a variety of concepts in a variety of ways. The creatures in BAW are able to learn facts (e.g., the location of other villages), how to do things (e.g., how to fish), which desires to prioritise (e.g., eating is more important that attacking enemy villagers), which objects are most important for satisfying particular desires (e.g., which objects are good to eat), and which methods to apply in which situations (e.g., the best way to attack another creature). The sources of learning for creatures in BAW include the following: player feedback (a player can stroke or slap their creatures in order reinforce or punish their actions), player commands (if the creature is ordered to kill a certain villager it can extrapolate over time which kinds of villagers should be killed), observation of other characters (creatures can learn from observations of the player, other creatures, or villagers), and reflection (the creature can 'reflect' on an experience and consider how well a particular action satisfied the desire which motivated it) [1].

Villagers

Villagers, unlike creatures, are unable to learn, but they have still been imbued with a reasonable degree of 'intelligence'. The challenge for BAW developers was to ensure that the villagers behaved intelligently without compromising the speed of the game by using too much CPU power [16]. This was achieved by giving each individual villager unique desires, but placing them under the control of a larger 'group-mind'. The group-mind ascertains what needs to be done for the sake of the village and delegates the appropriate actions to specific villagers [21].

Thus, by employing artificial intelligence techniques and seeking to make characters that are both believable and able to learn, Lionhead studios were able to develop a notably popular and successful game.

Future Directions for Games-AI

Since the release of BAW, despite the games' success, there has been speculation on how the game might have been improved. Unsurprisingly, Richard Evans, one of BAW's AI programmers has suggested some of the most interesting ideas. In BAW, the creatures have a finite number of goals. The possibility exists for imbuing characters with the ability to construct goals themselves [1]. Moreover, the creatures in BAW had only pre-programmed means of satisfying their goals. Although more computationally expensive, it would be possible for agents to generate novel ways of satisfying their goals by exploring various options and considering the results of their actions [1]. Finally, although the creatures in BAW are reactive (they respond to events) and proactive (they form intentions based on their own desires) they are not able to anticipate. They never 'plan ahead' or consider the consequences of their own or others' actions [21]. An agent with the ability to 'anticipate' or consider the consequences of their own or others' actions would be more flexible and most likely would appear to be more intelligent than the creatures in BAW.

Conclusions: Games AI and The Role of Academia

It is clear that games-AI techniques have undergone marked changes and improvements in recent years. However, bottlenecks still exist and obviously, a great deal more is still possible. Concurrently, academic AI research continues to progress, albeit at a slightly less breakneck speed than the developments occurring in Games-AI. There is little reason to doubt that both communities could benefit greatly from collaboration.

As Laird and van Lent point out [12], collaboration offers academic AI researchers an opportunity to conduct simulations in an increasingly realistic and powerful, ready-made environment. Computer games are inexpensive and reliable. Moreover, with more games incorporating extensibility, it is increasingly easy for researchers to adapt computer games to their needs.

For games' AI developers, collaboration could be a means of dealing with the pressure to continually improve the standard of AI in games. Many of the AI techniques currently employed are unlikely to be suitable for larger-scale, more ambitious projects [12]. In the past, games AI has often 'cheated' and provided only the illusion of human-like behaviour. Increasingly, the game-playing public expects more advanced AI techniques in games and they are less likely to be satisfied with the illusion of an intelligent character. In general, the game-playing public is starting to demand better AI [18]. However, due to the time-pressure inherent in games production, developers rarely have time to explore alternative methods or implement more risky techniques. Not only do academics have more time to research alternative techniques, they often have a very different perspective and a large knowledge base upon which to draw.

As Pottinger and Laird [18] point out, games that include extensible AI are an easy entry point for academic researchers into games AI. Research by Laird and colleagues [10,11,12,18] has led to the creation of a 'bot' (a non-player character) capable not only of playing the FPS, Quake 2, but also of anticipating an opponent's actions. (The fact that members of the academic community have been exploring and implementing one of the ideas highlighted by BAW developer Richard Evans as an important advance, is in itself evidence of the advantages of collaboration). As more games begin to incorporate extensible AI, similar academic research on games will be further facilitated.

Another potential area of collaboration is the use of neural networks and evolutionary algorithms in games. Many games' developers have reported difficulties as a result of the amount of processing power these techniques require. Given the wealth of research conducted on such techniques in academia, it is possible that less 'greedy' techniques could be found that are suitable for use in games.

The examples of collaboration mentioned above are drawn from a much larger pool of potential ideas. Given the advances already made in the game industry, the potential future developments, the wealth of knowledge that exists in academia, and the advantage to academics of ready-made simulation environments, it is clear that collaboration will benefit both parties. Given the relative dearth of collaboration thus far, this paper is designed not only to collate the available information and ideas, but also to provide a platform from which further collaboration can begin.

Appendix A. Artificial Intelligence Related Terms

It should be noted that disagreement exists regarding the definition of many of the terms listed below and what follows is not intended to be directive, rather it is an attempt to provide working knowledge for the reader less familiar with the area.

Artificial Intelligence Related Terms

Expert Systems

Sometimes referred to as the greatest commercial success of AI, expert systems are AI programs that use knowledge to solve problems that would normally require a human expert [2]. Expert systems require access to a substantial domain-specific knowledge base in combination with one or more reasoning mechanisms with which to solve problems [19]. Theoretically, the AI employed in most games can be considered an expert system – the more commonly used control structures are based on finite state machines (FSMs) or fuzzy state machines (see below).

Learning

Learning involves changes to a system's internal parameters and/or structure, resulting from interactions between the system and 'the outside world' that lead to a change in the system's performance in the future [15]. It has been suggested that a machine cannot be considered to have intelligence unless it is able to learn [19].

Fuzzy Logic

Fuzzy set theory was originally devised to provide a mathematical tool for dealing with concepts described in natural language. Fuzzy logic is a superset of conventional (Boolean) logic, which has been extended beyond the binary notion of true or false, to include the concept of partial truth [8,9]. For those unfamiliar with Boolean logic, fuzzy logic can be thought of as a system able to deal with vague concepts such as 'fairly tall' or 'almost flat'.

Finite State Machines

FSMs consist of a set of states, a set of inputs, a set of outputs, and a state transition function. Based on the input and the initial state, the state transition function calculates a new state and a set of outputs [5]. The term 'finite' in FSM refers to a finite number of states which the system can occupy. (Fuzzy state machines (FuSM) perform the same function, but whether and how the status of the object is changed is based on Fuzzy Logic).

Evolutionary Algorithms

Evolutionary algorithms (EAs) comprise a family of algorithms that take inspiration from the principles of biological evolution. They make use of adaptive change in a population of computational agents to solve problems or model complex systems. Each agent is represented by a set of genes, which specify features of the agent. Populations of agents are evolved by selecting and replicating fitter agents, and recombining and/or mutating their genes to create new agents. A variety of selection and recombination operators have been studied. For optimization problems, EAs can be viewed as search procedures to find the fittest agents. One type of EA commonly used in games is the Genetic Algorithm (GA), which primarily uses crossover to create new agents in the population [14].

Neural Networks

Artificial neural networks (NNs) are massively parallel networks of simple processing units. They were inspired by the massive connectivity of neural networks in biological brains. Nodes in a network represent units, connected by links. Knowledge is stored in weights (analogous to synapses) on these links. These weights are the modifiable parameters in the NN and are typically learned from patterns in its environment. The ability to learn gives NNs the power to acquire new behaviours, even to the extent of discovering ways of solving problems that their designers had not anticipated [4,15].

Artificial Life (A-Life)

Artificial life (Alife) is the study of synthetic systems that show behaviours characteristic of natural living systems. Alife systems typically consist of populations of entities that interact using simple rules defined at a local level. Such systems exhibit 'emergent behaviour', whereby complex dynamics or higher-level functionality develops as a result of the interaction between lower-level mechanisms [13].

Appendix B. Games Related Terms

A huge variety of games are currently available. To facilitate understanding for readers unfamiliar with the game industry, this section offers definitions of a number of game genres. The classifications offered are not intended to be exhaustive; depending on the criteria employed for classification different genre definitions could be generated).

Games Related Terms

Action Games

Action games generally involve the player guiding a protagonist through a series of levels (set in a virtual world) containing a variety of enemies. These games usually focus on the need for quick reflexes and refined hand-eye coordination. Perhaps the most prevalent form of action games is the first person shooter (FPS) in which the player views the virtual world from the perspective of the protagonist and uses a variety of weapons to overpower increasingly resilient enemies.

Fighting Games

Fighting games are based around the concept of physical combat between two characters. The plot is generally not an integral part of the game and all interaction takes place within a closed arena in which the two characters fight.

Sport Games

Sport games include virtual representations of a variety of sports (from basketball to skydiving). In team sports the player is generally only able to control one character at a time during gameplay, but this role is supplemented with input similar to that which would normally be provided by a coach. The degree of realism incorporated into sports games is highly variable.

Racing Games

Racing games include all games in which the player drives a vehicle in an effort to finish ahead of the other characters. Some racing games are modeled on real world racing events (e.g., car or motorbike racing) whereas others are far less realistic and incorporate elements such as the ability to shoot other competitors during the race.

Adventure Games

Adventure games tend to be plot-focused and involve activities such as exploration, information gathering and problem solving. They often require good hand-eye coordination and fast reflexes.

Role Playing Games (RPGs)

Derived from board games such as Dungeons and Dragons, RPGs generally focus on the player undertaking a quest. The quest may require the collection of items or information and battles with other characters. Often the improvement of the characters' skills and experience is an integral part of the game.

Strategy Games

Strategy games are based on the idea of a military-type battle scenario. The focus of the game is often the acquiring of resources, the building of defensive and offensive units (including both buildings and troops), and the organization and management of battles. The player almost always views the game from an overhead perspective.

Simulation Games

Simulation games involve the player taking control of a virtual environment in which a number of organisms (often humans) are interacting (e.g., the running of a hospital, the building and management of a city). The player is responsible for managing the environment and thereby the environment's inhabitants.

References

[1] Evans, R. "AI in Games: A Personal View." [Online]. Available: http://www.gameai.com/blackandwhite.html

[2] Finlay, J. and Dix, A. "An Introduction to Artificial Intelligence." London: UCL Press, 1996.

[3] Gustavson, S. "Multi-player 3-D graphics computer games: history, state of the art and development trends." [Online]. Available: http://www.itn.liu.se/~bjogu/TNM008-2001/games.pdf. April, 2001.

[4] Haykin, S. "Neural Networks: A Comprehensive Foundation." New Jersey: Prentice Hall, 1999.

[5] Hopcroft, J.E. and Ullman, J.D. "Introduction to Automata Theory, Languages, and Computation." Reading, Mass: Addison-Wesley. 1979.

[6] Howland, G. "Basics of Game AI." [Online]. Available: http://www.lupinegames.com/articles/basicai.htm

[7] Howland, G. "Introduction to Learning in Games." [Online]. Available: http://www.lupinegames.com/articles/introlearn.htm

[8] Kantrowitz, M. "Fuzzy Logic and Fuzzy Expert Systems." [Online]. Available: http://www.faqs.org/faqs/fuzzy-logic/part1/ August, 1997.

[9] Kosko, B. "Fuzzy Thinking: The New Science of Fuzzy Logic." New York, NY: Hyperion, 1993.

[10] Laird, J. "It Knows What You're Going To Do: Adding Anticipation to a Quakebot." Proceedings of the fifth international conference on Autonomous agents: 385-392. March, 2000.

[11] Laird, J. and Duchi, J. "Creating Human-like Synthetic Characters with Multiple Skill Levels: A Case Study using the Soar Quakebot." [Online]. Available: http://ai.eecs.umich.edu/people/laird/papers/human-level-quake.pdf 2000.

[12] Laird, J. and van Lent, M. "Human-level AI's Killer Application: Interactive Computer Games." Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on on Innovative Applications of Artificial Intelligence: 1171-1178. March, 2000.

[13] Langton, C. "Artificial Life" in Artificial Live, vol. 6, C. Langton, Ed. Santa Fe: Addison-Wesley, 1989.

[14] Mitchell, M. "An Introduction to Genetic Algorithms." Cambridge, MA: MIT Press, 1996.

[15] Mitchell, T. "Machine Learning." New York: McGraw-Hill, 1997.

[16] Molyneux, P. "Postmortem: Lionhead Studios' Black & White." [Online]. Available: http://www.gamasutra.com/ features/20010613/molyneux_01.htm June, 2001.

[17] Norman, T. "Belief, Desire and Intention Architectures." [Online]. Available: http://www.csd.abdn.ac.uk/~tnorman/publications/atal95/bdi.html November, 1995.

[18] Pottinger, D and Laird, J. "Game AI: The State of the Industry, Part Two." [Online]. Available: http://www.gamasutra.com/features/20001108/laird_pfv.htm November, 2000.

[19] Rich, E. and Knight, K. "Artificial Intelligence." New York: McGraw-Hill, 1991.

[20] Shelley, B. "Guidelines for Developing Successful Games." [Online]. Available: http://www.gamasutra.com/features/20010815/shelley_01.htm August, 2001.

[21] van de Walle, M. "Playing God." [Online]. Available: http://www.feedmag.com/templates/default.php3?a_id=1694

[22] van Lent, M and Laird, J. "Developing an Artificial Intelligence Engine." Proceedings of the Game Developers Conference 1999, 577-588. March, 1999.

[23] van Lent, M., Laird, J., Buckman, J., Hartford, J., Houchard, S., Steinkraus, K., and Tedrake, R. "Intelligent Agents in Computer Games." Proceedings of the National Conference on Artificial Intelligence, 929-930. July, 1999.

[24] Walker, T. "Black and White Dethrones the Sims." [Online]. Available: http://gamespot.com/gamespot/stories/news/0,10870,2707519,00.html April, 2001.

[25] Woodcock, S. "Game AI: The State of the Industry." [Online]. Available: http://www.gamasutra.com/features/19990820/game_ai_01.htm August, 1999.

[26] Woodcock, S. "Game AI: The State of the Industry." [Online]. Available: http://www.gamasutra.com/features/20001101/woodcock_pfv.htm November, 2000.

[27] Woodcock, S. "Games Making Interesting use of Artificial Intelligence Techniques." [Online]. Available: http://www.gameai.com/games.html

[28] Woodcock, S. "Games with Extensible AIs." [Online]. Available: http://www.gameai.com/exaigames.html

[29] "AI in Gaming." [Online]. Available: http://www.generation5.org/app_game.shtml

[30] "Black and White Homepage." [Online]. Available: http://main.bwgame.com/game.shtml

[31] "Get Artificial Life!" [Online]. Available: http://ai.about.com/library/weekly/aa032700a.htm

[32] "How to Get Started with Gaming AI." [Online]. Available: http://www.generation5.org/aisolutions/howto00.shtml

[33] ivan@mastergamer.com, "Master Gamer's 1997 Video Game Awards." [Online]. Available: http://www.mastergamer.com/features1997awards.html 2000.

[34] "Nooks and Crannies." [Online]. Available: http://www.andnow.net/nooks/index.html

[35] Deloura, M. & Rabin, S. (eds.). "AI Game Programming Wisdom." Boston, Charles River Media, 2002.

[36] Funge, J. "AI for Computer Games and Animation: A Cognitive Modeling Approach." Boston, AK Peters Ltd., 1999.