ARTIFICIAL INTELLIGENCE IN GAMING AND GAME DESIGN

Meet.S.Sanghvi

1Deptartment of Electronics and Communication, B.M.S Institute of Technology, Bangalore, India

ABSTRACT:

ARTIFICIAL INTELLIGENCE (AI) is the ability of a digital computer or computer-controlled robot to perform tasks commonly associated with intelligent beings. The term is frequently applied to the project of developing systems endowed with the intellectual processes characteristic of humans, such as the ability to reason, discover meaning, generalize, or learn from past experience. A computer game is an electronic game that involves human interaction with a user interface to generate visual feedback on a video device. In video games, artificial intelligence is used to generate intelligent behaviours primarily in non-player characters (NPCs), often simulating human-like intelligence. At its most basic level, artificial intelligence consists of emulating the behaviour of other players or the entities they represent. The real goal of AI in games is that the behaviour is simulated. The purpose of AI in games is decision making, perceptions and prediction. There are many methodologies used in AI for the creation of a game. In this paper the most efficient and compelling technique for the conception and designing of a digital game is examined and determined.

KEYWORDS: - games, artificial intelligence, A* algorithm, FPS and RTS types.

[1]. INTRODUCTION

The application of AI in game design now days is becoming more better by implementing the incredible complexity of advanced AI engines which has been developed by the efforts and
research of programming crowds. There are many types of computer programs that use AI. Market simulators, logic systems, and economic planners are some of the different fields of computer software that rely heavily on elements of artificial intelligence. These elements include situation calculus, tree searching, problem solving, and decision-making. But one genre of software programming has been slowly borrowing more and more from the field of AI is video gaming. The most common forms of Game AI in modern computers are those that select animations for Non-Playing character’s (NPCs) and allow the NPCs to navigate through the virtual environment without failure. Video games are no longer just a distraction from work or a thirty minute escape from reality. They are becoming an artistic form of expression for the programmers and developers and a serious hobby and undertaking for the players. General game-playing systems are a quintessential example of a new generation of software that end users can customise for their own specific tasks. This makes general game playing an interesting and challenging problem for AI, involving many fundamental issues such as reasoning, learning, planning and decision making. Consequently, general game playing can, and in fact should, be of interest to researchers in a variety of AI disciplines beyond conventional computer game playing. At the same time and for the same reasons, general game playing provides a new anchor for AI education as a unique framework for teaching multiple basic AI topics, such as problem solving by search, propositional and first-order logic, logic programming and planning. Game AI as practiced in industry encompasses a larger class of algorithms, data representations, hacks, and workarounds used to convey this illusion of intelligence. This paper attempts to give an overview of what has been achieved, and what are the methods that can be implemented to achieve a game that is artificially intelligent.

[2]. TYPES OF AI

The major types of AI are

1. Intelligent Systems
2. Knowledge
3. Demons
4. Expert Systems
5. Agents
6. Neural Networks
   i. Intelligent systems

Intelligent systems are a new wave of embedded and real-time systems that are highly connected, with massive processing power and performing complex applications. Their pervasiveness is reshaping the real world and how we interact with our digital life. These intelligent systems are creating new opportunities for industry and business, and new experiences for users and consumers. They can be found in all domains: automotive, rail, aerospace, and defence, energy, healthcare, telecoms and consumer electronics. An intelligent system is a system that has its own main objective, as well as senses and actuators to reach its objective. It chooses an action based on its experiences. It can learn by the experiences that have occurred to it and stores in its memories. Examples of intelligent systems are: people, higher animals, robots, computers, extra-terrestrials, a business, etc.

ii. Knowledge

Knowledge is an abstract term that attempts to capture an individual’s understanding of a given subject. In the world of intelligent systems the domain-specific knowledge is captured. Domain is a well-focused subject area. Cognitive psychologists have formed a number of theories
to explain how humans solve problems. Knowledge is the information about a domain that can be used to solve problems in that domain. To solve many problems requires much knowledge, and this knowledge must be represented in the computer. As part of designing a program to solve problems, we must define how the knowledge will be represented. Representation scheme is the form of the knowledge that is used in an agent. A representation of some piece of knowledge is the internal representation of the knowledge. A representation scheme specifies the form of the knowledge. A knowledge base is the representation of all of the knowledge that is stored by an agent. The technology of knowledge-based agents includes the syntax, semantics, proof theory of propositional and first order logic, and the implementation of agents that use these logics. Large scale knowledge representation requires general purpose ontology to organise and tie together the various specific domains of knowledge. General purpose ontology needs to cover a wide variety of knowledge and should be capable in principle of handling any domain.

iii. Demons

Demons are an artificial process that attends to various tasks without human interaction. This is normally done in the background of the system or in programs. This is useful for sorting information that the user does not need to be worried about or to automate programs e.g. Emails sending back the reply that the email cannot be sent. In multitasking computer operating systems, a demon is a computer program that runs as a background process, rather than being under the direct control of an interactive user. Traditionally demon names end with the letter d: for example, syslogd is the demon that implements the system logging facility and Systems often start demons at boot time and serve the function of responding to network requests, hardware activity, or other programs by performing some task. Demons can also configure hardware run scheduled tasks and perform a variety of other tasks. The word daemon is an alternative spelling of demon, Alternate terms for daemon are service started task and ghost job.

iv. Expert Systems

An expert is said to be a person who possess specialised skill, experience and knowledge that most people don’t have. Expert Systems are artificial intelligence systems which can provide expert information to decision-makers when a human expert is not available. Basically, an expert system works in a way to replace a human. An expert system is a computer program that contains stored knowledge and solves problems in a specific field. For example, A farmer could use an expert system to find out what sort of soil is best for his farm. The expert system consists of two major components: knowledge base and inference engine. Knowledge base contains the domain knowledge which is used by the inference engine to draw conclusions. The inference engine is the generic control mechanism that applies the axiomatic knowledge to the task-specific data to arrive at some conclusion. When a user supplies facts or relevant information of query to the expert system he receives advice or expertise in response. That is given the facts it uses the inference engine which in turn uses the knowledge base to infer the solution.

v. Agents

An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that. We see that the word ‘agent’ covers humans (where the sensors are the senses and the effectors are the physical body parts) as well as robots (where the sensors are things like cameras and touch pads and the effectors are various motors) and computers (where the sensors are the keyboard and mouse and the effectors are the monitor and speakers).environment through effectors.
In artificial intelligence, an intelligent agent (IA) is an autonomous entity which observes through sensors and acts upon an environment using actuators (i.e. it is an agent) and directs its activity towards achieving goals. Intelligent agents may also learn or use knowledge to achieve their goals.

vi. Neural Networks

Neural networks simulate the human nervous system. The concepts that guide neural network research and practice stem from studies of biological systems. These systems model the interaction between nerve cells. Components of a neural network include neurons (sometimes called "processing elements"), input lines to the neurons (called dendrites), and output lines from the neurons (called axons). Neural networks are composed of richly connected sets of neurons forming layers. The neural network architecture consists of an input layer, which inputs data to the network; an output layer, which produces the resulting guess of the network; and a series of one or more hidden layers, which assist in propagating. During processing, each neuron performs a weighted sum of inputs from the neurons connecting to it; this is called activation. The neuron chooses to fire if the sum of inputs exceeds some previously set threshold value; this is called transfer. Inputs with high weights tend to give greater activation to a neuron than inputs with low weights. The weight of an input is analogous to the strength of a synapse in a biological system. In biological systems, learning occurs by strengthening or weakening the synaptic connections between nerve cells. An artificial neural network simulates synaptic connection strength by increasing or decreasing the weight of input lines into neurons. Neural networks are trained with a series of data points. The networks guess which response should be given, and the guess is compared against the correct answer for each data point. If errors occur, the weights into the neurons are adjusted and the process repeats itself. This learning approach is called back propagation, and is similar to statistical regression. Neural networks are used in a wide variety of business problems.

[3]. ARTIFICIAL INTELLIGENCE HIERARCHY
[4]. APPLICATIONS OF AI

i. Game Playing

You can buy machines that can play master level chess for a few hundred dollars. There is some AI in them, but they play well against people mainly through brute force computation—looking at hundreds of thousands of positions. To beat a world champion by brute force and known reliable heuristics requires being able to look at 200 million positions per second.

ii. Speech Recognition

In the 1990s, computer speech recognition reached a practical level for limited purposes. Thus United Airlines has replaced its keyboard tree for flight information by a system using speech recognition of flight numbers and city names. It is quite convenient. On the other hand, while it is possible to instruct some computers using speech, most users have gone back to the keyboard and the mouse as still more convenient.

iii. Understanding Natural Language

Just getting a sequence of words into a computer is not enough. Parsing sentences is not enough either. The computer has to be provided with an understanding of the domain the text is about, and this is presently possible only for very limited domains.
iv. Computer Vision

The world is composed of three-dimensional objects, but the inputs to the human eye and computers' TV cameras are two dimensional. Some useful programs can work solely in two dimensions, but full computer vision requires partial three-dimensional information that is not just a set of two-dimensional views. At present there are only limited ways of representing three-dimensional information directly, and they are not as good as what humans evidently use.

AI IN GAMES

[5]. AI IN FPS-TYPE GAMES

FPS-type games usually implement the layered structure of the artificial intelligence system. Layers located at the very bottom handle the most elementary tasks, such as determining the optimal path to the target (determined by a layer higher up in the hierarchy) or playing appropriate sequences of character animation. The higher levels are responsible for tactical reasoning and selecting the behaviour which an AI agent should assume in accordance with its present strategy. Path-finding systems are usually based on graphs describing the world. Each vertex of a graph represents a logical location (such as a room in the building, or a fragment of the battlefield). When ordered to travel to a given point, the AI agent acquires, using the graphs, subsequent navigation points it should consecutively head towards in order to reach the specified target location. Moving between navigation points, the AI system can also use local paths which make it possible to determine an exact path between two navigation points, as well as to avoid dynamically appearing obstacles. The animation system plays an appropriate sequence of animation at the chosen speed. It should also be able to play different animation sequences for different body parts: for example, a soldier can run and aim at the enemy, and shoot and reload the weapon while still running. Games of this kind often employ the inverted kinematics system. An IK animation system can appropriately calculate the parameters of arm positioning animation so that the hand can grab an object located on, e.g., a table or a shelf. The task of modules from higher layers is to choose the behaviour appropriate for the situation – for instance, whether the agent should patrol the area, enter combat, or run through the map in search of an opponent. Once the AI
system has decided which behaviour is the most appropriate for the given situation, a lower-level module has to select the best tactics for fulfilling that task. Having received information that the agent should, for instance, fight, it tries to determine the approach that is the best at the moment – e.g., whether we should sneak up on the opponent, hide in a corner and wait for the opponent to present a target of itself, or perhaps just run at him, shooting blindly.

[6]. AI IN RTS-TYPE GAMES

In RTS-type games, it is possible to distinguish several modules of the artificial intelligence system and its layered structure. One of the basic modules is an effective path-finding system – sometimes, it has to find a movement solution for hundreds of units on the map, in split seconds – and there is more to it than merely finding a path from point A to point B, as it is also important to detect collisions and handle the units in the battlefield avoid each other. Such algorithms are typically based on the game map being represented by a rectangular grid, with its mesh representing fixed-sized elements of the area. On higher levels of the AI system's hierarchy, there are modules responsible for economy, development or, very importantly, a module to analyse the game map. It
is that module which analyses the properties of the terrain, and a settlement is built based on the assessment, e.g., whether the settlement is located on an island, thus requiring higher pressure on building a navy. The terrain analyser decides when cities should be built and how fortifications should be placed.

[7]. AI IN SPORTS GAMES

Basically, in the case of most sports games, we are dealing with large-scale cheating. Take car racing games, for instance. For the needs of the AI, from the geometry of the game map, only and only the polygons belonging to the track of a computer-controlled opponent should travel on and get distinguished. Two curves are then marked on that track: the first represents the optimal driving track, the second – the track used when overtaking opponents. The whole track gets split into appropriately small sectors and, having taken parameters of the surface into account, each element of the split track gets its length calculated. Those fragments are then used to build a graph describing the track, and to obtain characteristics of the road in the vehicle's closest vicinity. In effect, the computer knows it should slow down because it's approaching the curve, or knows that it's approaching an intersection and can, e.g., take a shortcut. Two important attributes of Artificial Intelligence systems in such games is being able to analyse the terrain in order to detect obstacles lying on the road, and strict co-operation with the physics module. The physics module can provide information that the car is skidding, having received which the Artificial Intelligence system should react appropriately and try to get the vehicle's traction back under control.

[8]. THE MOST POPULAR AI ALGORITHMS IN COMPUTER GAMES

In the following part of the article, I would like to discuss the two most popular algorithms used in programming computer games. Possessing knowledge about them, one can successfully design a simple artificial intelligence system fulfilling the needs of simple FPS or RTS games. The first of the two is the A-Star algorithm, used in performing fast searches for the optimal path connecting two points on the map (graph) of a game. The other is the finite state machine, useful, e.g., in preparing behaviour scenarios for computer-controlled opponents, typically delegating its low-level tasks to a path-finding module.

[9]. THE A* ALGORITHM

The problem of finding a way from point A to point B on a map is a key problem in almost any computer game (possibly not counting certain sports games and some other types of games which can be counted using the digits of one hand). At the same time, algorithms from this group belong to the lower level of the games' AI, serving as a base for constructing more complicated and more intelligent types of behaviour, such as strategic planning, moving in formations or groups, and many others. This issue has already been thoroughly evaluated in the world of computer games, with one algorithm – A* – having become a present-day standard.
The following pseudocode describes the algorithm:

```plaintext
function A*(start, goal)
  closedset := the empty set  // The set of nodes already evaluated.
  openset := {start}  // The set of tentative nodes to be evaluated, initially containing the start node
  came_from := the empty map  // The map of navigated nodes.
  g_score[start] := 0  // Cost from start along best known path.
  // Estimated total cost from start to goal through y.
  f_score[start] := g_score[start] + heuristic_cost_estimate(start, goal)

  while openset is not empty
    current := the node in openset having the lowest f_score[] value
    if current = goal
      return reconstruct_path(came_from, goal)
    remove current from openset
    add current to closedset
    for each neighbor in neighbor_nodes(current)
      if neighbor in closedset
        continue
      tentative_g_score := g_score[current] + dist_between(current, neighbor)
      if neighbor not in openset or tentative_g_score < g_score[neighbor]
        came_from[neighbor] := current
        g_score[neighbor] := tentative_g_score
        f_score[neighbor] := g_score[neighbor] + heuristic_cost_estimate(neighbor, goal)
        if neighbor not in openset
          add neighbor to openset
      return failure

function reconstruct_path(came_from, current)
  total_path := [current]
  while current in came_from:
    current := came_from[current]
    total_path.append(current)
  return total_path
```

The above algorithm helps us in searching very accurately and is the most used artificial intelligence algorithm.
[10]. FINITE STATE MACHINES

Finite state machines are one of the least complicated, while at the same time, one of the most effective and most frequently used methods of programming artificial intelligence. For each object in a computer game, it is possible to discern a number of states it is in during its life. For example: a knight can be arming himself, patrolling, attacking, or resting after a battle; a peasant can be gathering wood, building a house, or defending himself against attacks. Depending on their states, in-game objects respond in different ways to (the finite set of) external stimuli or, should there be none, perform different activities. The finite state machine method lets us easily divide the implementation of each game object's behaviour into smaller fragments, which are easier to debug and extend. Each state possesses code responsible for the initialisation and de-initialisation of the object in that state (also often referred to as the state transition code), code executed in the game's each frame (e.g., to fulfil the needs of artificial intelligence functions, or to set an appropriate frame of animation), and code for processing and interpreting messages coming from the environment.

The following pseudocode is the most accurate and structured and maintainable.

Finite State Machines can be colloquially defined as:

1. A set of *states* that the agent can be in
2. Connected by *transitions* that are triggered by a change in the world
3. Normally represented as a directed graph, with the edges labeled with the transition event
4. Ubiquitous in computer game AI.
[11]. FAMOUS GAMES USING AI

- Far Cry 2 (2008)

A first-person shooter where the player fights off numerous mercenaries and assassinates faction leaders. The AI is behaviour based and uses action selection, essential if an AI is to multitask or react to a situation. The AI can react in an unpredictable fashion in many situations. The enemies respond to sounds and visual distractions such as fire or nearby explosions and can be subject to investigate the hazard; the player can utilize these distractions to his own advantage.
There are also social interfaces with an AI but however not in the form of direct conversation but more reactionary, if the player gets too close or even nudges an AI, the player is subject to getting shoved off or sworn at and by extent getting aimed at. Other social interfaces between AI exist when in combat, or neutral situations, if an enemy AI is injured on the ground, he will shout out for help, release emotional distress, etc.[citation needed]


A first-person shooter where the player helps contain supernatural phenomenon and armies of cloned soldiers. The AI uses a planner to generate context-sensitive behaviours, the first time in a mainstream game. This technology used as a reference for many studios still today.

The enemies are capable of using the environment very cleverly, finding cover behind tables, tipping bookshelves, opening doors, crashing through windows, and so on. Squad tactics are used to great effect. The enemies perform flanking manoeuvres, use suppression fire, etc.
Creatures (1996)

‘Creatures’ is an artificial life program where the user "hatches" small furry animals and teaches them how to behave. These "Norns" can talk, feed themselves, and protect themselves against vicious creatures. It's the first popular application of machine learning into an interactive simulation. Neural networks are used by the creatures to learn what to do. The game is regarded as a breakthrough in artificial life research, which aims to model the behaviour of creatures interacting with their environment.

[12]. LATEST AI GAMES

• MIDDLE EARTH: SHADOW OF MORDOR

Shadow of Mordor will go down as one of the most important games of recent years for starting the trend of giving non-story NPCs individual personalities, memories of past encounters and in many cases, a survival instinct! All things lacking in most combat games where NPCs are nothing more than short-lived target dummies.
• PLANETARY ANNihilation

From the AI perspective, many new challenges had to be solved by the team, in particular inter-planetary resource allocation, an AI that can reason strategically on a sphere, and scaling up to huge number of units. On top of this, the implementation focuses on building a skilled AI using a neural network and a form of reinforcement learning. This makes it interesting to play against for learning purposes and challenging while leveling up!

• ALIEN: ISOLATION

What makes Alien: Isolation particularly interesting from an AI perspective is the Xenomorph — which interacts with the player over the course of the entire experience. Sustaining such a prolonged interaction is rare for modern games, and the entire experience relied on its artificial intelligence.
[13]. CONCLUSION

As Artificial Intelligence is one of the most efficient and effective technologies in problem solving and pattern recognition, it is one of the most upcoming technologies in the IT world. The culmination of Artificial Intelligence and game playing techniques and algorithms will enhance and better the gaming environment. The rampant progress of AI technology makes nearly every game very life like and spontaneous. Due to the importance the gaming environment receives, ensuring and realizing the most accurate artificial intelligence algorithm is my ultimate objective in this paper.

REFERENCES

[1]. Ashwin Ram, Santiago Ontanon, and Manish Mehta, Cognitive Computing Lab (CCL) College of Computing, Georgia Institute of Technology Atlanta, Georgia, USA {ashwin, santi, mehtama1}@cc.gatech.edu.


[3]. Georgios N. Yannakakis, Member, IEEE, and Julian Togelius, Member, IEEE, ‘A Panorama of Artificial and Computational Intelligence in Games’.

[4]. Simon M. Lucas, Michael Mateas, Mike Preuss, Pieter Spronck, and Julian Togelius, University of Essex, GB, sml@essex.ac.uk University of California – Santa Cruz, US, michaelm@cs.ucsc.edu, TU Dortmund, DE, mike.preuss@tu-dortmund.de, Tilburg University, NL, p.spronck@uvt.nl, TU Dortmund, DE, julian@togelius.com, ‘Artificial and Computational Intelligence in Games’

[5]. James Wexler University of Rochester Rochester, NY 14627 jw005i@mail.rochester.edu, ’Artificial Intelligence in Games: A look at the smarts behind Lionhead Studio’s “Black and White” and where it can and will go in the future’.


[7]. http://www.aijunkie.com/books/toc_pgaibe.html


[10]. Jeremy A. Glasser, University of Nebraska, jglasser@cse.unl.edu, Leen-Kiat Soh University of Nebraska, lsoh2@unl.edu, AI in Computer Games: From the Player ‘s Goal to AI’s Role.


Artificial Intelligence in Gaming and Game Design


[18]. http://www.bbc.co.uk/newsbeat/30879456

[19]. Michael Thielscher School of Computer Science and Engineering The University of New South Wales, mit@cse.unsw.edu.au, ‘General Game Playing in AI Research and Education’.


[24]. Jonathan Schaeffer, H. Jaap van den Herik, Department of Computing Science, University of Alberta, Edmonton, AB, Canada T6G 2E8 Department of Computer Science, Universiteit Maastricht, PO Box 616, 6200 MD Maastricht, Netherlands, ‘Games, computers, and artificial intelligence’.

[25]. Michael Mateas The Georgia Institute of Technology School of Literature, Communication & Culture and College of Computing 686 Cherry Street Atlanta, GA 30332-0165 USA +1 404 894 2739 michaelm@cc.gatech.edu,’Expressive AI: Games and Artificial Intelligence’.

[26]. Professor Charles Rich Computer Science Department rich@wpi.edu, ‘Basic Game AI-Technical Game Development II’.

[27]. Bernhard Beckert, universität koblenz-landau, ‘Introduction to Artificial Intelligence Game Playing’.