

# Artificial Intelligence Techniques in Game Contents

**Sang-Su Ko**

Department of Software Engineering  
Sejong University, Seoul, Korea

**Song-Hwa Chae**

Korea IT Industry Promotion Agency  
Ajou University, Seoul, Korea

**Byung-Woo Nam, Won-Il Kim\***

Department of Digital Contents  
Sejong University, Seoul, Korea

## ABSTRACT

*Nowadays, many people enjoy playing games in computer. In this kind of game, people often meet NPC (Non Player Character). It is the virtual character in simplified form of real player and exists in most of current computer games. Various NPCs add the reality and atmosphere of the game as well as help players. There are several techniques to embody NPC, but developers generally use AI technique. This paper discusses some artificial intelligence techniques used in game contents. Especially this paper focuses on the AI techniques used in computer games in terms of the two main approaches, symbolic approach and sub-symbolic approach*

**Keywords:** AI, techniques, computer games, Non Player Character (NPC), symbolic approach, sub-symbolic approach

## 1. INTRODUCTION

In the game, Non-Player Character (NPC) appears frequently. They sometimes help players by the method of guiding path or giving important information, but sometimes they are attack players. Game developers make these kinds of intelligent NPC by using various techniques. These most techniques are AI techniques. There are many kinds in AI techniques such as Finite State Machine, Path Finding, Decision making, Neural Networks, and Genetic Algorithms. In this paper, we discuss some artificial intelligence techniques used in game contents. Especially this paper focuses on the AI techniques used in computer games in terms of the two main approaches, symbolic approach and sub-symbolic approach.

## 2. NON-PLAYER CHARACTER

Non-player Character is the character that players do not play. If NPC is employed appropriately in the game, it (the game) has more reality and interest. Artificial intelligence algorithm and techniques help NPC display the action that

looks smart.

The decision cycle of those NPCs constantly executes three steps such as perceive, think and act. In the perceive step, the NPC accepts information about the environment. In the next think step, the NPC evaluates perceived information and plan according actions. Finally, in the act executes the planned actions [2].

This far too simple approach may look unsuitable for creating an interesting game, however that is not so. In the actual video games, realistic NPCs are unnecessary, since it requires too much time to make them real people.

If NPCs are as sophisticated and smart as the humans who play the game, players are not able to win in the game and the game become boring. The AI of NPCs should never be too good. Instead it should allow and help the player to win the game in interesting and challenging ways.

## 3. AI TECHNIQUES IN GAME CONTENTS

The game progressed all the time over the past 20 years and

---

\* Corresponding author. E-mail: wikim@sejong.ac.kr  
Manuscript received Sep. 08, 2006 ; accepted Sep. 21, 2006

---

This work was sponsored and funded by Korea Game Development & Promotion Institute as Korean government project. (Ministry of Culture and Tourism)

the AI techniques also progressed within those games. It is proved and some of the successful techniques are hardly changing with time and when AI in their games needs to be implemented, those techniques are almost always the first selection of the developers. However over the past decade, more and more new AI ideas and methods for games have developed into the game development process [3].

### 3.1 Symbolic approach

The techniques that can generally be looked for in computer games are Symbolic approach techniques. They can be easily implemented in the game. Symbolic approach is technique that based on meaningful state and divide by meaning. In symbolic approach, there are finite state machine, fuzzy state machine, decision trees, knowledge based techniques, agents, and annotated environments.

**Finite State Machines:** Finite State Machines that are type of AI are most generally used in games. In an FSM the action of the NPC is arranged in logical states – one state per possible behavior - only with one state active at once. A state is an active or inactive Boolean value - on or off. When it is necessary to change the present behavior into different behavior, the FSM will change between the states. For example, transition from the stance to look out for to the attack on the most intimate partner. It is comparatively simple to program a very stable FSM that may not be very sophisticated but that will get the job done. The main faults of FSM can have very complicated them, and maintaining can become difficult. On the other hand prediction of the behavior that originates in too simple FSM can be attained easily. In order to conquer this problem sometimes hierarchical FSMs are used. These are FSMs where each state can itself be an FSM. In games, recent examples in which FSMs are used are the game-bots in the *Quake* of first-person shooters. Here each NPC has a number of states that define the character's current behavior [3].

**Fuzzy logic:** Fuzzy state machines using fuzzy logic are a permutation of FSMs using Boolean logic. Consequently, states in FuSMs are not limited to being on or off but they can have an intermediate value. It is more complicated to make the construction of FuSMs rather than it makes the construction of an FSM, but active states of FuSMs greatly reduces the predictability of the resulting behavior. It also reduces the complexity of the state machine. For example, a wider range of different behaviors can be encoded with fewer states. FuSMs are a comparatively new game AI technique that can be used in almost all of the zones in which FSMs are used. There are *The Sims* and *Civilization: Call to Power* in games recently using FuSMs.

**Decision trees:** Decision trees are based on machine learning and machine intelligence techniques. When players want to predict future actions or classify conditions, decision trees are most preferred method in the game AI techniques because this method is very reliable and robust. The AI of creature in the *Black & White* contains decision trees method.

**Knowledge based techniques:** Knowledge based techniques are uncommonly used on the main part of engine when it comes to games AI, but they are often used as sub-systems of games AI in strategy games. This would contain terrain analysis techniques such as influence mapping. It allows a strategic AI in a war game to assess the current situation, to identify choke points for ambushes or to position its troops on the virtual battlefield. Moreover search strategies are often used for path finding for NPCs in a wide range of games like individual units in strategy games [7].

**Agents:** Agent is the system that perceives environment and it's affecter through responds to the environment. Recently agent algorithms are used in computer games. Intelligent agents are decision-making entities that are usually composed of the other AI methods. For example, an agent could integrate machine learning techniques with FSMs to be able to analyze the player's behavior, so it can anticipate the player's next move and make appropriate decisions and plans. The computer opponent AI in real-time strategy games has often an agent program.

**Annotated environments:** Many games now use annotated environments to simplify the simulation of intelligent behavior. In short, this method holds information about individual NPC instead of holding all NPCs' informations. It helps NPC to be extensible in the game and to have less complexity. Annotated environment is used in game *The Sims*.

### 3.2 Sub-symbolic approach

Sub-symbolic approach based on neurophysiology is a technique that forms one meaning with a set of many implicit symbols. The recent games use frequently sub-symbolic approach for making smart NPCs. Neural network, evolutionary techniques, and artificial life technique is based on the sub-symbolic approach.

**Neural network:** Neural networks are the complicated nonlinear functions that make one or more input variables connected with one output. Neural network is nonlinear functions use interconnected nodes. It allows the network to learn and improve itself [1]. Using a neural network can enable games to adapt to the way that the player plays by updating itself while playing a game. This technique can be used in the simulation games. For instance, NPC's gesture recognizes in *Black & White* and learning ability in *Creatures series* [1]. However they can be also used successfully in adventure games or action games like *Heavy Gear* in which the robots controlled by the player use neural networks to improve its skills in line with the player's performance.

**Evolutionary techniques:** Evolutionary technique method uses selection method and mutation method until finding optimal state. This evolutionary method is least using in the game because it takes long time to find the optimal state so it is not suitable for the real-time games [4]. However evolutionary technique method finds solution through error, so it is stable and adaptable.

**Artificial life:** Artificial Life is the discipline of trying to emulate natural life in an artificial environment such as a computer. Artificial Life is more of a collected name for a number of different demonstrations rather than a scientific discipline. Recent game that has made use of Artificial Life is *Black & White* [2].

### 3.3 Other AI techniques

One of recent trend in games is to make them extensible by allowing users to modify them to their needs. Some games even tender software interfaces, allowing parts of the games to be reprogrammed. A main area in which games can be modified in this way is the game AI.

**Parameter tweaking:** Parameter tweaking is one kind of extensible AI- this method gives authority to user to modify the game. In parameter tweaking method, we can change NPC behaviour through changing internal parameter of AI games.

**Plug-in interfaces:** Some games like *Quake* include plug-in interfaces. It allows to can change the AI of NPCs in the game. Some games even have complex software development kits to simplify the modification of the game behaviour [5].

**Scripting:** Many new games include complex scripting system that permit the game AI to be extended or modified. Many games have built-in dedicated scripting languages, like *Quake* which includes a scripting language called *QuakeC* or *Unreal* that has a scripting system called UnrealScript. Other games use existing scripting systems that have been modified according to the game's requirements. Such one example is the scripting language Lua which has been used in many games, including the game *MDK2* from Bioware who also used scripting in their role-playing games *Baldur's Gate* and *Neverwinter Nights* [6].

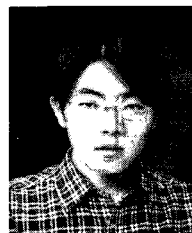
## 4. CONCLUSION

In this paper, we discussed some techniques of game artificial intelligence, and classified AI techniques into symbolic and sub-symbolic. The complexity of AI characters in computer games is continually improving, and AI techniques also progressed within those games.

## REFERENCES

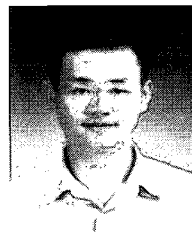
- [1] Steve Rabin. et al, "The Dark Art of Neural Networks", In M. Steve Rabin (ED.), AI Game programming Wisdom 2, Charles River Media, 2003, pp. 640-652
- [2] van Lent, M. et al, "Intelligent Agents in Computer Games", Proceedings of the National Conference on Artificial Intelligence, 1999, pp. 929-930.
- [3] Dybsand, E. "A Generic Fuzzy State Machine in C++." In M. Deoura (Ed.), Game Programming Gems 2, Charles River Media, 2001, pp. 337-341.

- [4] Anderson, E.F, "Off-Line Evolution of Behaviour for Autonomous Agents in Real-Time Computer Games", Parallel Problem Solving from Nature - PPSN VII LNCS VOL. 2439, 2002, pp. 689-699.
- [5] Laird, J. E, "It Knows What You Are Going To Do: Adding Anticipation to a Quakebot", Proceedings of the Agents-2001 International Conference on Autonomous Agents, 2001, pp. 00-02
- [6] Ierusalimschy, R. et al, "Lua-an extensible Extension Language", In Software: Practice & Experience, Vol. 26, 1996, pp. 635-652
- [7] Tozour, P, "Influence Mapping", In M. Deloura (Ed.), Game Programming Gems 2, Charles River Media, 2001, pp. 371-383



**Sang-Su Ko**

He studies Software engineering at Sejong University, Korea. His main research interests include Artificial intelligence and Embedded system.



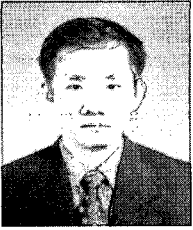
**Byung-Woo Nam**

He studies Digital Contents at Sejong University, Korea. His main research interests include Artificial intelligence and Data mining



**Song-Hwa Chae**

She received a B.E. in Information and Computer Engineering from Ajou University, Korea in 1997 and a M.E. in Computer Engineering from Ajou University, Korea in 1999. She received Ph.D. in Information Communication from Ajou University, Korea in 2006. From 1999 to 2001, she was researcher of Korea Information Security Agency to study PKI system. Since 2006, she has been with Korea IT Industry Promotion Agency, Seoul, Korea. Her research interests include digital content, digital convergence, wireless communication system, and secure information systems.



**Won-II Kim**

He received the B.E in Metal Engineering from Hanyang University, Seoul, Korea in 1982. He worked for Korean Air from 1981 to 1985 as System designer and programmer. He received the B.S., M.S in Computer Science from Southern Illinois

University, U.S.A. in 1988, 1990 respectively. He received Ph.D. in Computer and Information Science from Syracuse University, U.S.A. in 2000. From 2000 to 2001, he worked for Bhasha INC, U.S.A. as technical and research staff. He was with Ajou University, Suwon, Korea from 2002 to 2003. Since 2003, he has been with Department of Digital Content, College of Electronics and Information, Sejong University, Seoul, Korea. His main research interests include Artificial Intelligence, Multimedia Contents and Computer Security.