A Survey on Application of Artificial

Intelligence for the Non Player Character in

Computer Games

REDWAN ISLAM, Computer Science and Engineering Discipline.

Khulna University

Artificial intelligence for non-player character (NPC) has become a key technology for computer games. Use of AI in modern computer games make NPCs behavior more human like and natural. In this paper we present a survey on artificial intelligence for NPC in computer games. We will discuss about existing AI techniques and algorithms used for NPC and make a comparison between the AI algorithms to improve NPC movement. And lastly talk about the future research areas on this sector. There is a lack of research for improving NPC behavior in computer games. We believe this paper will attract more researchers. So that, in future the game AI becomes more real and human like.

Additional Key Words and Phrases: Artificial Intelligence, NPC, Computer Games, Game AI algorithms

## Introduction

With day by day development of technology of computer hardware and software, the computer games became a form of expression as rich in content as books and movies. The players demand a realistic experience in computer games, so the developers are paying more attention to improve AI in computer games.

The roles in computer games is divided into two parts: player character and non-player character. Non-player character is out of player control. There comes the role of AI. Games that use sophisticated AI techniques are more enjoyable to play.

Most of the research is on how to make this NPCs behave more human like.

AI researchers experiment with video games with a number of reasons

* To improve existing techniques
* To create new types of AI within the confines of game
* To experiment with new AI techniques

The goal of our survey is to investigate AI technique used for NPC in games, and pointing out possibilities of future research. We answer the research questions listed in Table I and make the following contributions:

* Classifying and discussing the existing studies on AI for NPC in games
* An analysis on the existing AI methods
* Identifying future research possibilities in the area.

Our survey can be helpful to the researchers because it will help them know the existing techniques used for AI for NPC in games, find techniques that are yet to be found

## NPC AND AI

A non-player character (NPC) in a game is any character that is not controlled by a player. [1] In video games, this usually means a character controlled by the computer via predetermined or responsive behavior, but not necessarily true artificial intelligence.

AI is a science that simulate thinking processes and acting. It has two forms: strong AI and weak AI. [1] The former argues that people can produce the intelligent machine accorded with working mechanism of human brain. This kind of machine is considered have sensations and self-consciousness. The latter argues that people can’t produce the intelligent machine. It looks like an intelligent machine, but it does not have intelligent and self-consciousness. At the present time, AI in computer games is clearly the latter.

## Types of npc

There are two types of NPCs.

* Reactionary: It is based on player movement. For example, if the player is in 20 meter range of the enemy. The enemy decides to attack.
* Spontaneous: The NPC does its prescribed actions not getting affected by any player movement.

## TABLE 1: RESEARCH QUESTIONS

|  |  |
| --- | --- |
| Q1 | Can we categorize the existing studies on AI for NPC in games? |
| Q2 | What are the existing methods? |
| Q3 | What are the possible future research areas? |

## SURVEY PROCEDURE

We have performed a literature survey on AI for NPC. We have searched ieeexplore.ieee.org (IEEE Xplore Digital Library) for relevant papers. The keywords that we searched are: NPC, AI in computer games. The numbers of papers that we obtained from that website for different keywords are listed in Table II. When searching papers, we used advanced search options and selected appropriate fields of research to narrow down our search results. We found a large amount of papers published on those keywords.

We have also found several research papers on games on Mario and Pac man [2; 3]. We have read those papers abstracts. The main target of those papers are to create an AI bot which can play those games like human. They use the same techniques we found out on our survey topics. We have only focused on methods used for NPC to interact with real time players. So, in this survey we haven’t included those papers.

First we checked the title of the papers to decide whether it is relevant or not. If not we read the abstract. We have read the conclusion of every paper and note down the methods. We also took a look at some experimental results. Finally we selected 97 papers that can answer the research questions in TABLE 1.

In the following sections we present our survey.

Fig 1: Number of publications on AI for NPC in computer games

## existing ai methods for npc

The AI in games is usually described the NPC how to resolve problems, how humans think and act. We have found 34 AI methods in order to make NPC more realistic. In this survey we will talk briefly about the most common AI methods for NPC in computer games. In the late part we will shortly discuss about all the existing methods. The most frequently used methods are:

* Monte Carlo Tree Search
* Fuzzy Logic and Gaussian Distribution
* A Star Path Finding Algorithm
* Reinforcement Learning
* Neural Networks and Genetic Algorithm

## monte carlo tree search

In the last few years, several Monte-Carlo based techniques emerged in the field of computer games. They have already been applied successfully to many games, including POKER (Billings et al. 2002) and SCRABBLE (Sheppard 2002). Monte-Carlo Tree Search (MCTS), a Monte-Carlo based technique that was first established in 2006, is implemented in top-rated GO programs. These programs defeated for the first time professional GO players on the 9×9 board.

Monte Carlo Tree Search is an algorithm used when playing a so-called perfect information game. In short, perfect information games are games in which, at any point in time, each player has perfect information about all event actions that have previously taken place. For this reason, researchers use MCT to create intelligent adaptive game opponent [4]

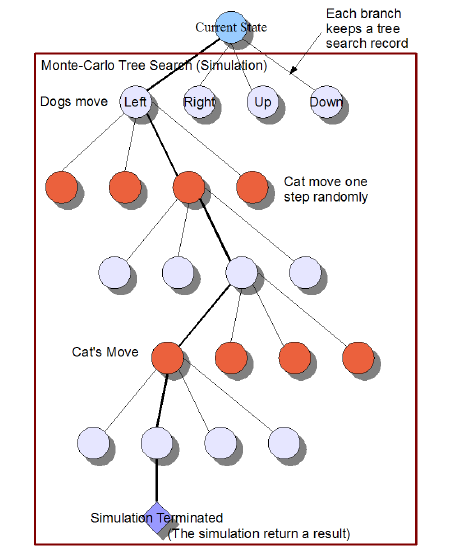


Fig 2: The Monte-Carlo Tree Search in Dead-End [4]

Jiajian Yang, Yuan Gao, Suoju He, Xiao Liu, Yiwen Fu, Yang Chen, Donglin Ji in their paper used MCT to make a simulation of a dog catching a cat by calculating the highest winning rate possible [4].

For this reason, MCT is better than other algorithms such as BFS, DFS.

## Fuzzy Logic and Gaussian Distribution

Fuzzy logic is a form of [many-valued logic](https://en.wikipedia.org/wiki/Many-valued_logic) in which the [truth values](https://en.wikipedia.org/wiki/Truth_value) of variables may be any real number between 0 and 1, considered to be "fuzzy".

Many researches had been done on fuzzy logic in order to maintain dynamic NPC behavior. Researchers implement fuzzy logic in games like first person shooter, role playing game, puzzle platforming game.

Peter Leong, Miao Chunyan exhibit has developed a goal oriented and non-deterministic behavior by using fuzzy logics for a role playing game [5].

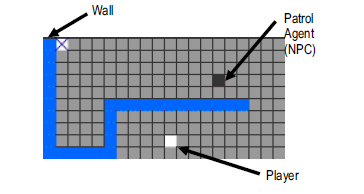


Fig 3: Game scenario [5]

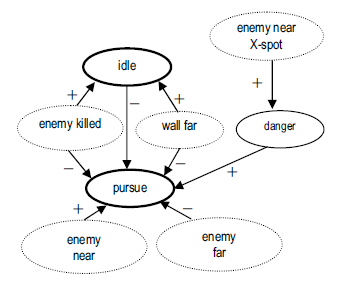


Fig 4: An example of the agent knowledge model based on the case study [5]

The goal net in figure 4 contains the following

elements:

g1: Protect the “X”-spot goal. This is the overarching

goal of the NPC agent.

g2: Patrol along wall goal.

g3: Approach suspicious player goal

g4: Pursue enemy (identified threat) goal

g5: Kill enemy goal.

t1: Move along the wall. Distance from wall

should not be large.

t2: Finished patrol for one checkpoint.

t3: Move towards object if suspicious.

t4: Pursue object if it moves away.

t5: Attempt to kill enemy if close enough.

t6: Finished threat elimination.

We have seen that fuzzy logic is implemented on a single NPC movement. What if we have to implement it in a war game. War games show the battle of NPC troops, one of the AI research challenges in war games is to develop coordination

techniques.

Muhammad Aminul Akbar and Mochamad Hariadi [6] have published a paper on multi behavior NPC coordination. They have shown that, in offensive team strategy, overall actions from coordinated NPC troops become more offensive than uncoordinated NPC. In defensive team strategy, overall actions from coordinated NPC troops become more defensive than uncoordinated NPC.

## A\* Pathfinding Algorithm

The A Star Pathfinding Algorithm is already proved as most effective. It tries to find the lowest cost path that will be the shortest and the only path [1].

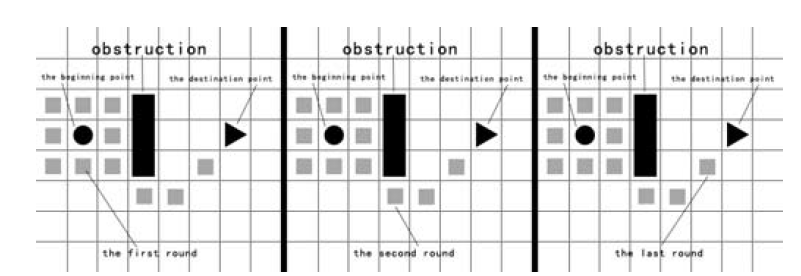


Figure 5: Find the lowest cost path [1]

For instance, we divide the map into some points and then find the path. The first finding searches for the nearest 8 points to the beginning point. The passed road plus the linear distance as the starting condition after each round. Every point has a value. Each round start with the lowest value until finding the destination. As shown in figure 5.

If we consider racing games, the most common artificial intelligence in a racing game is waypoint navigation by carefully placing waypoints (nodes) in the game environment to move the game-controlled characters between each point. This is a very time consuming and CPU intensive problem

Dr. Jung-Ying Wang used A\* path finding algorithm to solve this problem. [1]

## reinforcement learning

In reinforcement learning the NPC learns from the environment to adapt new gameplay. Reinforcement learning is mainly used in turn based board games.

In this section we will be talking about using reinforcement learning in more complex games such as first person shooter(FPS ). A challenge in such environments is that the time that elapses between deciding to take an action and receiving a reward based on its outcome can be longer than the interval

between successive decisions [7].

In 2016 Frank G. Glavin and Michael G. Madden propose a new method for RL updates and reward calculations. Basically they made a NPC which can play fps game in a 3D environment and gather experience from gameplay & rewarding system to be better at shooting.

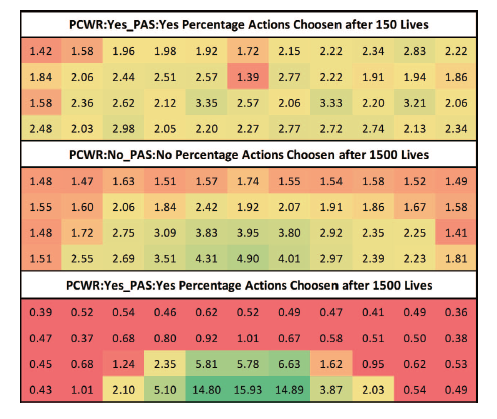


Fig. 6. Percentage of overall shooting actions selected after 150 lives and

after 1500 lives [7].

These results show clear evidence that the RL shooting architectures are capable of improving a bot’s shooting technique over time as it learns the correct actions to take through in game trial and error.

## previous research

General background information on the use of AI in virtual agents can be found in Yannakakis and Hallam [8]. A wide variety of computational intelligence techniques have been successfully applied to FPS games such as Case-Base Reasoning (CBR) [9], Decision Trees [10], Genetic Algorithms [11] and Neural Networks [12].

## artificial neural network and genetic algorithm

An artificial neural network is a system that tries in various degrees to emulate a human brain in order to perform tasks that other computer systems are usually not fit to handle. Artificial neural networks are used in many different areas due to their ability to learn and adapt to many different tasks and make complex predictions.

Artificial Neural Networks(ANNs) have been in wide use since at least the 1980s for, among other things, complex modeling and various recognition, prediction and filtering tasks.

We will address the earlier gaming implementations, discussing implementations for classic board games and continue into modern computer games.

## Board Games

Several implementations of ANNs to serve as trainable AI for games were made popular, including ANNs as AI for Othello, Tic Tac Toe [13], five in a row [14], and checkers [15], which will be discussed in this section.

**ANNs in Othello**

A popular implementation of ANNs for AI is for the game of Othello, also known as Reversi**.** One implementation [16] was done by David E. Moriarty and Risto Miikkulainen in 1995. Moriarty and Miikkulainen created and trained an ANN capable of discovering complex strategies and playing on par with an experienced human.

**MODERN COMPUTER GAMES**

**Using ANNs in racing games**

ANNs have also been used to control cars in racing games. H. Tang, C. H. Tan, K. C. Tan and A. Tay compared the possibilities of creating AI for racing games by using neural networks or Behavior Based Artificial Intelligence (BBAI) [17].

## Other games

Researchers have been experimenting on Nintendo Entertainment System (NES) games such as Super Mario Bros. The idea is to create a AI bot that can play Mario human like.

In 2016 Patrikk D. Sørensen, Jeppeh M. Olsen and Sebastian Risi published a paper on “Breeding a Diversity of Super Mario Behaviors” [18]

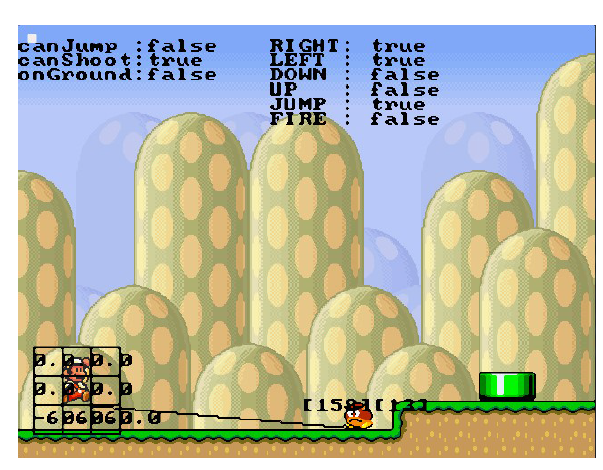


Fig 7: Mario Representation. The controlling ANN receives a 3 \_3 grid as input together with information about the distance and angles to enemies, and conditional domain variables canJump and onGround. The ANN outputs (shown at the top-right corner) determine the action that Marioperforms each tick.[18]

## Genetic Algorithm

Genetic algorithm is a method simulation the biological evolution. The NPC can make a choice in a series of procedures, algorithms and parameters. These methods are based on those of natural evolution, and are powerful in finding the global or near global optimal solution of optimization problems [1].

## other used methods in ai for npc

|  |  |
| --- | --- |
| **Methods** | **Description** |
| Spoken communication with computer game characters | This method explores the characteristics of a new language for communication with game characters. This new language is called game pidgin language |
| Gaussian Distribution | The Gaussian distribution is a continuous function which approximates the exact binomial distribution of events |
| Emotional Model | The NPC changes its facial expression according to its emotion during the interaction with the playable character |
| Chatterbots Conversation | Chatterbots are computer programs that simulate intelligent conversation. |
| Computational Based Behavior | The realistic yet efficient NPCs can be  developed by using a methodology based on well understood  behavior-based techniques. |
| Finite State Machine | A finite-state machine is a model used to represent and control execution flow. It is perfect for implementing AI in games |
| Evolutionary Algorithm | Evolutionary algorithm is the broad term given to the group of optimization and search algorithms that are based on evolution and natural selection, including genetic algorithms, evolutionary computation and evolutionary strategies*.* |
| UCT Algorithm | UCT implemented opponent can achieve good adaptability which contributes well to the entertainment of game as it dynamically adjusts to different gamers’ needs. |
| Real Time Team Behavior | It allows NPCs to react from game environment, learn from their experience and communicate with their opponent. |
| DDA | Dynamic Difficulty Adjustment (DDA)of Game AI  aims at creating a satisfactory game experience by dynamically  adjusting intelligence of game opponents. It can provide a level of challenge that is tailored to the player’s personal ability. |
| Dynamic Team-mate Adaptation | Game systems that perform dynamic team-mate  adaption and even less on developing team-mate NPCs (Non  Player Characters) that adaptively support players in the face of opponents that adaptively increase the difficulty for the player. |
| Petri Nets and feedback  control | Petri nets are graphical and mathematical tool for modeling, analyzing, and designing discrete event applicable to many systems. They can be applied to game design too, especially to design serous game |
| Smart Terrain Method | Smart Terrain has the potential to improve the  realism of games and the behavior of the NPC in games such as  first person shooter games where gameplay is fast paced. The  potential issue would be if NPCs are dynamic enough to  recognize changes in the virtual environment or if the virtual  environment provides the required information to assist the  NPCs with their objectives or tasks. |
| Emotional Behavior Trees | If we wish to have a natural  and interesting behavior, it is important that the characters  behave in an emotional way. It could be claimed that it  is possible to incorporate emotions into behavior trees by  merely using emotions in the conditions. However, doing so  may create large cumbersome behavior trees that are difficult  to manage. For each behavior, a specific set of conditions  would have to be placed on emotional states. These conditions  would most likely take the form of checking the emotional  values against a fixed threshold, which would disable a subtle  emotional effect on decision making. Using this approach  would most likely lead to a large behavior tree with numerous  nested conditions, making it difficult to construct and manage. |
| HTN Planning  Approach Based on Visual Perception | firstly, a visual perception system for  non-player characters (NPCs) along with a short-term memory  (working memory) are implemented so that NPCs will have access  to only limited information and have to build their plans and  make their decisions based on what they can perceive from their surrounding environment. Secondly, an HTN planning approach  is implemented based on SHOP for the domain of our serious  game |
| Machine Learning | There are two types of machine learning used in games. One is learning at design-time, where the results of learning are applied before publishing the game; the other is learning at runtime, fitted to a particular player or game session |
| Coupled Empowerment Maximisation | It is used to fit the NPC behavior with procedural content generator game environment |
| ID3 | ID3 decision-making system can be divided into behavioral tree  part and decision tree part, and the decision making process of  the behavior tree part is the same as the traditional behavior  tree. Decision making process of decision tree part can be  regarded as the process of information analysis and processing  of background data: In the process of interaction between the  NPC and the player, it is inevitable to have a lot of changes in  the parameters of the attributes, which are the information  generated by the game system environment. Have the choice of  the collection of this information, we can use the ID3  algorithm based on information gain to dig out the best value  of the property parameters [19] |
| Belief Desire Intension | The belief–desire–intention software model is a software model developed for programming intelligent agents |
| Swarm Intelligence and Ant colony | The Ant Colony Optimization and Particle Swarm Optimization are two Swarm Intelligence algorithms often utilized for optimization. Swarm Intelligence relies on agents that possess fragmented knowledge, a concept not often utilized in games |
| Morris Water Maze | Virtual versions of the Morris Water Maze (VMWM) have been used to investigate spatial learning in non-human subjects in games. |

## COMPARISON BETWEEN A\* and HA\* ALGORITHM

Path finding algorithms finds the shortest path between two points and it is vastly used in any kind of games. If the maps are usually not very big, traditional algorithms such as Dijkstra algorithm and A Star algorithm work well [20]. But if we want to find the shortest path of two points in a big city or even a country these algorithms may result in bad performance due to lack of memory and CPU resources [20]. The A\* path finding algorithm uses a huge amount of memory to find the path between start and end nodes but with HA\* it takes less time than A\* path finding algorithm achieving a tradeoff result between real-time and performance [20].

In 2014 International Conference on Digital Home Haifeng Wang, Jiawei Zhou, Guifeng Zheng, Yun Liang has published a paper proposing that HA\* is better than A\*

## 

Fig 8: The time cost for different algorithm to find the

shortest path [20]

## Future Research Possibilities

* By Fig 8 we know that HA\* is better and more efficient then A\* pathfinding algorithm but no research had been done on implementing HA\* algorithm in computer games to benefit the NPCs. This demands a future research on this topic.
* In the future, AI development in video games will most likely not focus on making more powerful NPCs in order to more efficiently defeat human players. Instead, development will focus on how to generate a better and more unique user experience. As [Virtual Reality](https://en.wikipedia.org/wiki/Virtual_reality) (VR, which provides an immersive viewing experience by means of a display) and [Augmented Reality](https://en.wikipedia.org/wiki/Augmented_reality) (AR, which combines a human’s physical view of the world with virtual elements) technologies continue to expand, the boundary between the virtual and real world is beginning to blur. Last year’s [Pokémon Go](https://en.wikipedia.org/wiki/Pok%C3%A9mon_Go), the most famous AR game, demonstrated the compelling power of combining the real world with the video game world for the first time. In the future, VR- and AR-based open-world video games may provide players with a “real world” experience

## Conclusion

In this paper we present our survey on the existing research and techniques on AI for NPC in computer games. We categorize the studies on the basis of their research. We also identify the existing AI methods and compare among some methods. We also believe that we have found the answers to the questions asked in Table 1. We hope that our survey will help AI researchers to know all existing possible AI methods for NPC in games.

## References

[1] Jiang Jie, Kuang Yang, Shen Haihui "The Application of AI for the Non Player Character in Computer Games" 2011 International Conference on Computational and Information Sciences

[2] Julian Togelius, Sergey Karakovskiy, Jan Koutnik and Jurgen Schmidhuber(2009) : “Super Mario Evulation” Proceedings on IEEE Symposium on CIG

[3] Xiao Liu, Yao Li, Suoju He, Yiwen Fu, Jiajian Yang, Donglin Ji, Yang Chen "To Create Intelligent Adaptive Game Opponent by Using Monte-Carlo for the Game of Pac-Man" 2009 Fifth International Conference on Natural Computation

[4] Jiajian Yang, Yuan Gao, Suoju He, Xiao Liu, Yiwen Fu, Yang Chen, Donglin Ji "To Create Intelligent Adaptive Game Opponent by Using" 2009 Fifth International Conference on Natural Computation

[5] Peter Leong, Miao Chunyan "Fuzzy Cognitive Agents in Shared Virtual Worlds "Proceedings of the 2005 International Conference on Cyberworlds (CW’05)

[6] Muhammad Aminul Akbar, Wida Praponco, Mochamad Hariadi and Supeno Mardi S.N "Multi Behavior NPC Coordination Using Fuzzy Coordinator And Gaussian Distribution" 2015 International Seminar on Intelligent Technology and Its Applications

[7] Frank G. Glavin and Michael G. Madden "Learning to Shoot in First Person Shooter Games by Stabilizing Actions and Clustering Rewards forReinforcement Learning" IEEE CIG 2015

[8] G. N. Yannakakis and J. Hallam, “Towards optimizing entertainment in computer games,” Applied Artificial Intelligence, vol. 21, no. 10, pp.933–971, 2007.

[9] B. Auslander, S. Lee-Urban, C. Hogg, and H. Munoz-Avila, “Recognizing the enemy: Combining reinforcement learning with strategy selection using case-based reasoning,” in Advances in Case-Based Reasoning. Springer, 2008, pp. 59–73.

[10] A. J. F. Leiva and J. L. Barrag´an, “Decision tree-based algorithms for implementing bot ai in ut2004,” in Foundations on Natural and Artificial Computation. Springer, 2011, pp. 383–392.

[11] A. Esparcia-Alc´azar, A. Mart´ınez-Garc´ıa, A. Mora, J. Merelo, P. Garc´ıa- S´anchez et al., “Controlling bots in a first person shooter game using genetic algorithms,” in Evolutionary Computation (CEC), 2010 IEEE Congress on. IEEE, 2010, pp. 1–8.

[12] S. Petrakis and A. Tefas, “Neural networks training for weapon selection in first-person shooter games,” in Artificial Neural Networks–ICANN 2010. Springer, 2010, pp. 417–422.

[13] Fogel, D.B. “Using Evolutionary Programming to Create Neural Networks that are Capable of Playing Tic-Tac-Toe”. IEEE International Conference on Neural Networks (ICNN), Vol. 2, 28 March – 1 April 1993, p. 875 – 880.

[14] Freisleben, B. “A Neural Network that Learns to Play Five-in-a-row”. Second New Zealand International Two-Stream Conference on Artificial Neural Networks and Expert Systems, 1995, 20-23 November 1995, p. 87-90.

[15] Chellapilla K., and Fogel, D.B, "Evolving Neural Networks to Play Checkers Without Relying on Expert Knowledge”. IEEE Transactions on Neural Networks, Vol. 10, No. 6, November 1999

[16] Moriarty, David E, Miikkulainen, Risto. “Discovering Complex Othello Strategies Through Evolutionary Neural Networks”, Connection Science, Volume 7, 1995, p. 195-209.

[17] Tang, H., Tan, C. H., Tan K. C., and Tay, A. "Neural Network versus Behavior Based Approach in Simulated Car Racing Game". IEEE Workshop on Evolving and Self-Developing Intelligent Systems, 2009. ESDIS '09, March 30 2009-April 2 2009, p. 58 – 65.

[18] Patrikk D. Sørensen, Jeppeh M. Olsen and Sebastian Risi "“Breeding a Diversity of Super Mario Behaviors”

[19] Yang Li , Dai-Wen Xu "A Game AI based on ID3 Algorithm" 2016 2nd International Conference on Contemporary Computing and Informatics (ic3i)

[20] Haifeng Wang, Jiawei Zhou, Guifeng Zheng, Yun Liang2 "HAS:Hierarchical A-Star algorithm for big map navigation in special areas" 2014 International Conference on Digital Home