INTERNATIONAL JOURNAL OF RESEARCH –
GRANTHAALAYAH
A knowledge Repository

RACSIT - 17

DEVELOPMENT OF AUTONOMOUS GAME AGENT WITH LEARNING AND REACTIVE BEHAVIORS

Deepa.T.P. [*1]
[*1] Department of Computer Science and Engineering, SET, Jain University, Bangalore, India

DOI: https://doi.org/10.29121/granthaalayah.v5.i4RACSIT.2017.3360

Abstract

The main goal of this paper is to develop software agent which is autonomous with reactive behavior and learning abilities. One of the applications of such agents are in gaming. Gaming characters are expected to work in unpredictable environment with decision making capabilities like weapon selection for different targets, wall following. This can be achieved artificial intelligence (AI) techniques and methods. In this paper, agent is designed to exhibit capabilities like - Moving Abilities, Steering behavior and obstacle avoidance, Synthesis of movement enhancing movement, weapon selection, adapting defense strategies, strategic Decision making.

Keywords: Agent; Autonomous; Navigation; Gaming; Weapon Selection.

Cite This Article: Deepa.T.P.. (2017). “DEVELOPMENT OF AUTONOMOUS GAME AGENT WITH LEARNING AND REACTIVE BEHAVIORS.” *International Journal of Research - Granthaalayah,* 5(4) RACSIT, 91-96.

1. Introduction

There is a significant change and development in the gaming industry over years. This has demanded for the character appearing in the gaming world is intelligent and close to human in the real world. This helps game programmers to design more realistic games. There is a need for game characters with cognitive behaviors like how they interact with the game terrain, other characters in a game, how they take decision in various situations of game for example in shooter game, whether to face the enemy or choose weapon and attack. There is a need for incorporating some of the natural behavior of human in synthetic game characters [13]. Especially in the games like first person shooter, player should learn from the environment and experience based on which gaming decisions should be taken. In such games, players should have abilities to learn from the old game and apply decisions in the new game without much prior details [14]. Overall an effort is made to design non-player characters (NPC) in game to behave like a human and get adopted to different gaming terrain. First person shooter (FPS) game is characterized by extremely realistic environment compared to other games like puzzles. Also, it provides flexible platform to change terrain, maps, items, obstacles etc. in the environment [15]. Most of the experiments on FPS will be conducted using Quake 3 arena. Though FPS is older generation game, but there exists few challenges in weapon choices, flexibility in the size of player base. This also allows accessing control settings of environment through command-line. Quake 3 has relatively short loading time. Through, quake 3, one can experiment on different maps in a game [16].

2. Related Works

In [1], Authors describes about the handling behaviour of robots in dynamic environment. Authors describes tools and techniques for development of robots. Also, knowledge and learning behaviour of agents, behaviour based and hybrid agent architecture was discussed. Author gave an insight about software perception of robot and future trends on it. In [3], Author describes about the challenges to face when the character-based agent for gaming has to be designed. Development of virtual autonomous characters that are empathetic. They are implemented using pattern recognition techniques and planning abilities. In [5], first author et al, Describes about key ideas and algorithms for implementing reinforcement learning. They also discussed case study problems and solutions using reinforcement learning. In [6], Authors describes about how to enhance the movement primitives of agents. They proposed a framework for modular motor control based on dynamic movement primitives. They proposed a method to convert smooth kinematic control policies to motor commands. In [7], First author et al describes about how agent perform perception, control behaviour is the changing environment. They demonstrated a virtual marine environment similar to that of real time. Their algorithms emulated complex group behaviours which are found in many aquatic environments.

3. Proposed System

In this work, agents are working in gaming environment. Autonomous agent in this work is an Intelligent player with behaviors like moving abilities, steering behavior, object picking and avoidance, navigation, weapon selection, wall following. Different artificial intelligence techniques like neural network, genetic algorithms, and decision trees were used to incorporate these behaviors in agent. Player has learning behavior which was implemented using batch and incremental learning. Different techniques are used for different behavior of the agent as follows-

A. Moving Abilities

3D reactive environment was considered. To design navigation which involves movement with purpose was challenging. Proposed method used rule-based systems and fuzzy logic to enhance the accuracy of results. Fuzzy logic dealt with tricky contraptions in an environment. Fuzzy variable was current position an agent. With known variables, applying fuzzy rules could achieve the task. These fuzzy rules kept track of position in sequence. Knowledge of position and directions were represented implicitly using rule based system with specific rules were implemented for make sure there is no side effects of execution order. Finally fuzzy logic interpreter with rule based system was created.

B. Steering Behavior and Obstacle Avoidance

An acceleration applied by game agent is called as steering force. This represents locomotion of an agent. This is nothing but position, velocity and locomotion which were handled by animation. Sometimes, environment imposed frictions, so agent stops by eliminating velocity. Through steering force, velocity of an agent was adjusted. This feature was implemented along with obstacle avoidance. Obstacles were detected based on their shape like circular. Other obstacles were detected by approximating with circular bounding shapes. This helps agent to steer around the obstacle. The agent could also avoid collision by implicitly scanning free space which was found during obstacle detection. Also free space was identified by tracing lines through empty spaces.

C. Weapon Selection

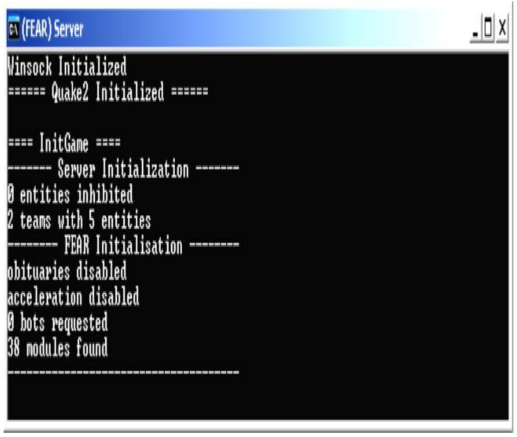
There are various weapons available for an agent in a game. These weapons are characterized based on their projectile speed and reload times. Though, some of the weapons performed similar tasks but they behaved in a different way. Sometimes weapons should be selected based on the environment. Agent was selecting weapon based on the fitness value. This fitness value was determined by mapping features and criteria into single value. Agent would compare fitness values and decide which weapon to select. Here agent was interacting with game engine to gather information and perform changes in selecting weapons. These interfaces were designed using query to an inventory due to spatial restriction. Agent was able to determine health and weapons available by querying inventory. Based on the feedback, call back and events from the environment it was indirectly found how much damage caused to an enemy agent. To decide most appropriate weapon to the current situation, classification trees was used. To estimate fitness of a particular weapon and estimate different properties of weapons and to predict the value for each features of weapon, regression trees were used.

D. Aiming

Aiming at the target was implemented using neural networks. Two perceptron’s were used to increase the realistic aiming behavior similar to human players. One Perceptron learns to turn smoothly by approximating model of aiming errors. This feature was trained offline using batch training algorithm. Another perceptron was increasing effectiveness of aiming by approximating solution that compensated aiming error.

E. Developing Adapting Defense Strategies Using Genetic Algorithms

The combination of genetic algorithm and rule-based systems were used to make agent learn simple behaviors like rocket jumping and dodging rockets. Genetic algorithm selects candidate variables based on the value of fitness function. That is it will select fittest value and discards weakest. Two fitness functions were used. One for rocket jumping and other for dodging the fire. For example, for fitness value assigned based on the upward movement. To prevent agent jumping high with larger fitness value while running upstairs. Value is considered for upward movement when agent is not touching floor. For dodging fire, it depends on the incoming fire, distance of the agent to the point of explosion is measured. If rocket hits the agent then distance and fitness was near. If agent scampers far away from the collision point, then fitness was high. As rockets explodes quickly, no limit was imposed on the fitness. But, when agent was away from the collision point and standing skill, fitness was high. To prevent this case, agent was monitoring the difference in the distance which is done by measuring average velocity away from the point of collision.



```
Winsock Initialized
====== Quake2 Initialized ======

==== InitGame ====
------- Server Initialization -------
0 entities inhibited
2 teams with 5 entities
-------- FEAR Initialisation --------
obituaries disabled
acceleration disabled
0 bots requested
38 modules found
-------------------------------------
```

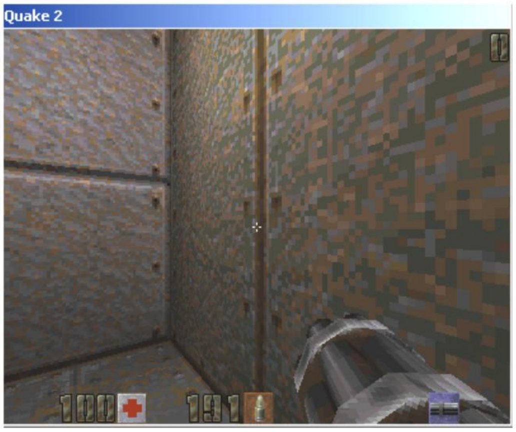
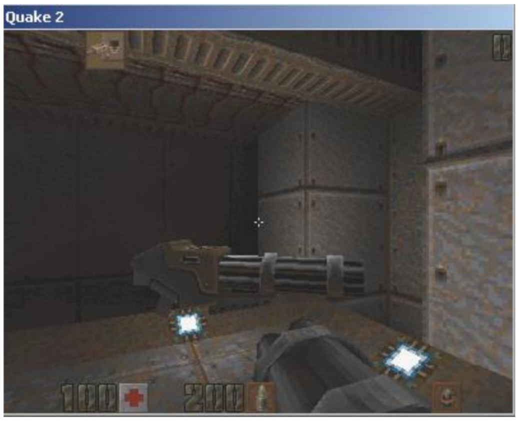


Figure 1: Quake server- game environment

Figure 2: Agent wall following



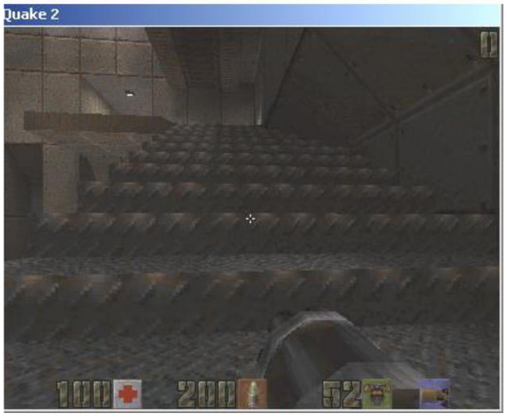
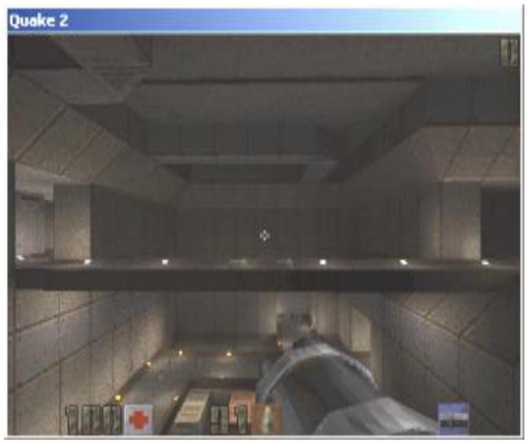


Figure 3: Agent selecting the weapon

Figure 4: Agent- climbing the stairs



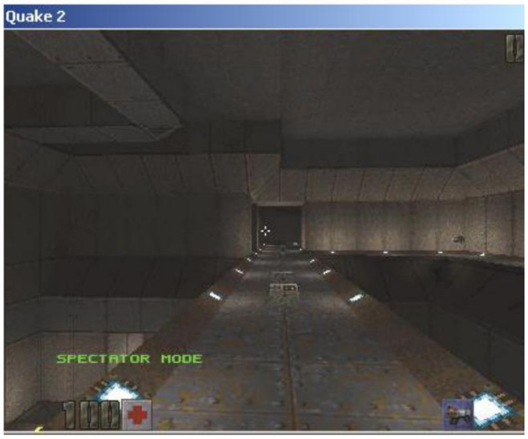


Figure 5: Agent- at the dead end

Figure 6: Agent- environment scan

Table 1: Test cases and observations on agent's behavior

Test Conditions	Invalid output	Valid Output
Target Selection	Selects different and Unwanted target	Proper target is selected and tracking of target is done.
Weapon Decision	Makes wrong decision and selects wrong weapon	With the information from the terrain about the target and required weapon, a proper weapon is selected.
Target distance	Agent might shoot before the Target or beyond the target.	Agent calculates the exact distance of the target relative to its position and fires at the exact target.
Weapon selection	User has to train the agent regarding which weapon has to be selected for the particular situation	Agent learns from its experience about the situation and automatically selects the needed weapon.

4. Conclusion

Autonomous agent research represents a new approach to the study of intelligence. This is successful in the area of mobile robots and software agents. The problem is splitted into manageable parts and implemented each of the solutions separately. The behaviors that are created contribute toward the intelligence of the agent and provide additional realism. A rule-base system gives smooth output with a manageable number of rules. The target-selection behavior provided a great example of applying a neural network. Using fuzzy expert systems deductive reasoning, determining new facts from a set of statements in a human-like fashion were implemented.

5. Future Work

An agent with the combination of deliberative and reactive behaviour can be developed which consists of several subsystems like subsystems that develop plans and make decisions using symbolic reasoning (deliberative component) and Reactive subsystems that are able to react quickly to events without complex reasoning (reactive component).

References

[1] Arkin, R. C. (1998). Behavior-Based Robotics. Intelligent Robotics and Autonomous Agents, MIT Press, ISBN: 9780262011655

[2] Cass, S. (2002). Mind games [computer game AI], IEEE Spectrum ,Volume: 39, Issue: 12, Dec 2002, pages 40–44.

[3] Isla, D. and Blumberg.B, New challenges for character-based AI for games. In Proceedings of the AAAI Spring Symposium on AI and Interactive Entertainment, 2002

[4] Laird, J. E. and v. Lent, M. (2000). Interactive Computer Games: Human-Level AI's Killer Application. In Proc. AAAI, pages 1171–1178.

[5] Sutton, R. S. and Barto, A. G. (1998). Introduction to Reinforcement Learning. MIT Press, 2nd ed.

[6] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert, "Learning Movement Primitives," in International Symposium on Robotics Research (ISRR2003), Springer Tracts in Advanced Robotics. Ciena, Italy: Springer, 2004.

[7] Terzopoulus, D., Tu, X., and Grzeszczuk, R. (1994). Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world. In Artificial Life, pages 327–351.

[8] Tomlinson, B. and Blumberg, B. (2002). Alphawolf: Social learning, emotion and development in autonomous virtual agents. First GSFC/JPL Workshop on Radical Agent Concepts.

[9] D. Coco. Creating Intelligent Creatures. Computer Graphics World, July, 1997.

[10] R. E. Fikes and N. Nilsson. STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. Artificial Intelligence 5(2): 189–208, 1971.

[11] J. Funge, X. Tu, and D. Terzopoulos. Cognitive Modeling: Knowledge, Reasoning and Planning for Intelligent Characters. In Proceedings of the International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH’99), 29–38, 1999.

[12] T. Gard. Building Character. Gamasutra, http://www.gamasutra.com/features/20000720/gard01.htm, June, 2000

[13] F. Dignum,1 J.Westra,1 W. A. van Doesburg,2 and M. Harbers1, 2, Games and Agents: Designing Intelligent Gameplay, Hindawi Publishing Corporation International Journal of Computer Games Technology, Volume 2009, Article ID 837095, 18 pages, doi:10.1155/2009/837095.

[14] Firas Safadi, Raphael Fonteneau, and Damien Ernst, Artificial Intelligence in Video Games: Towards a Unified Framework, Hindawi Publishing Corporation International Journal of Computer Games Technology Volume 2015, Article ID 271296, 30 pages, http://dx.doi.org/10.1155/2015/271296

[15] Dongkyu Choi, Tolga Konik, Negin Nejati, Chunki Park, and Pat Langley, A Believable Agent for First-Person Shooter Games, page:71-73, American Association for Artificial Intelligence (www.aaai.org),2007.

[16] Kajal T. Claypool, Mark Claypool, On frame rate and player performance in first person shooter games, Multimedia Systems, Springer, DOI 10.1007/s00530-007-0081-1

*Corresponding author.
E-mail address: deepa.sce@gmail.com