

Implementing Artificial Intelligent Ghosts in MS. Pac-Man Game

Tariq Ahmed Khan ¹ & S.M Saifuz Zaman ²

^{1,2} Department of Computer Science and Engineering, Daffodil International University

Email : ¹ partho222@gmail.com , ² saifuzzaman1789@gmail.com

Abstract – In this paper A* search & AI is introduced for controlling the ghosts in MS Pacman game . A* search is used to find the optimal path. AI is used to produce the illusion of intelligence among the behavior of NPC's (Non player character). Here each ghost is given same AI method for movement.

Keywords – Artificial intelligence; AI; A* search; games

I. INTRODUCTION

AI has been utilized to play and rivals human on chess, board games, Othello etc. A big number of studies have been done on video game AI nowadays. AI system are requisite to operate adaptively and instantly according to environment in video games like Ms. Pac-Man. The rule of these game is very simple. Let the player control the Pac Man character to lead the way through the maze, walk off the four opponent ghosts that are trying to understand and catch Pac Man. The player earns points by collecting pellets that are dispersed in the maze avoiding eaten by the ghosts. Four special power pellets become Pac Man to the predator and gain even more points for a small amount of time. When all pellets have been eaten by Pac Man, the game will begin again on a new maze with new map. The game terminates when Pac Man loses three lives and a life is lost by clashing with an enemy ghost.

This paper offering a developed controller for the Ms. Pac Man video game founded with neural networking. The main achievement of our designed Pac Man game is the AI based controller. In with non-deterministic nature of a ghost, it becomes more aggressive and it is much more difficult by the player to track out the nature of each ghost. All these ghosts behave in a approximate disorder way. As more maps are finished, the game

becomes much harder. Not only do the ghosts speed up, but also they get “intellectual”. All possible directions are developed which means the proposed controller program has to response to the running game status in actual duration. The next section of this paper gives an overview of related work in the area of Pac Man controller research.

Section III explains the piece of information about implementation of Ms. Pac Man game, while section IV explains experimental setup of the proposed system. Section V analyses the results of the experiments that have been taken on hand during this project. In section VI, what were the challenges explains and the final section then give an outline of presented work and aspect to possible following projects.

II. LITERATURE REVIEW

A. Previous experiments and related works

There have been respective previous efforts to make the Pac Man game more attractive and thrilling. Koza [1] was among the first to look into develop a controller for a custom and feature reduced version of Pac Man game. In his work Koza defined a set of 15 functions: two conditionals (e.g. Distance-To-Pellets) and 13 primitives for controlling the Pac Man (e.g. Advance-To-Pellets). In this version, all ghosts follow the same nature and his variant of the game turned out to much easier than arcade version. More recently, Gallagher and Ryan [2] developed a Pac Man player based on finite-state-machine (FSM) additionally rule set. In their work they evolved the parameters of the rule set (85 in total), where the particular rules were hand specified. However it was great a simulation of this game and it makes the game more attractive. De Bonet and Stouffer [3] explained a way to simple mazes and there has been some work on learning routes for Pac Man also, but this

approach is not workable for Ms. Pac Man. Many researches have applied their AI techniques to teach the computer how to play Pac Man. The Monte-Carlo tree search (MCTS) [4] approach has grown in popularity over the last different years and has been applied on Pac Man more than one. Tong and Sung [5] used MCTS to give Ms. Pac-Man the capability to avoid the ghosts in order to enrich the final score.

B. Scope of artificial intelligence in games

Artificial intelligence (AI) in computer games basically works for game-playing characters' (non-player character or NPC) behavior & decision making. Now-a-days most of the computer game industries implement AI in their game. Such games consolidate rich and complex situations with expertly created, stable, physical science based simulation. They are constant and extremely dynamic, empowering quick and canny choices. Computer games are additionally frequently multi operators, making collaboration, rivalry, and NPC demonstrating key components to achievement. In commercial games, like as action games, role-playing games, and strategy games, the behavior of the NPC is typically executed as a variety of straightforward rule-based frameworks. With a couple of special cases, machine-learning strategies are scarcely ever connected to condition of-the-workmanship computer games. Machine-learning procedures may empower the NPC's with the ability to enhance their execution by learning from mistakes and successes, to naturally adjust to the strengths and weaknesses of a player, or to learn from their opponents by mirroring their strategies.[6]

III. IMPLEMENTATION

The classes that have been worked on to implement Artificial Intelligence are Map, Ghost, AI Manager, and Path & Path Finder. We use java to develop our game & a map file to calculate the place on screen for pellet, block, ghost & legend. We use A* search algorithm to find path to the present position of "pacman" in this game.

If $\mathbf{p} = (p_1, p_2)$ and $\mathbf{q} = (q_1, q_2)$ then the distance Euclidean Distance Calculation: [9]

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}$$

Manhattan Distance Calculation: [10]

$$d_1(p, q) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^n |p_i - q_i|$$

IV. DESIGN & DEVELOPMENT

In this research work, the following methods have been used to find out the path that ghost should follow.

Path Finder: Find a path from the starting location provided (sx, sy) to the target location (tx, ty) avoiding blockages and attempting to honor costs provided by the tile map. It uses the A* search algorithm to find the path

Process: Run all logic required for AI operation fear, ghost release, path updates for *Ghost act()* functions are called here. It randomizes the x & y coordinate for ghost entity then uses pathfinder to find the path.

Get Movement Cost: It will calculate the cost for making a movement by the ghost

A* search (Cost Calculation):

- The past path-cost function, which is the known distance from the starting node to the current node x (usually denoted s(x)).
- A future path-cost function, which is an admissible "heuristic estimate" of the distance from x to the goal node (usually denoted t(x)). [7]

The AI Manager control all the four ghost to wander in a random & shortest direction to pursue the pacman.

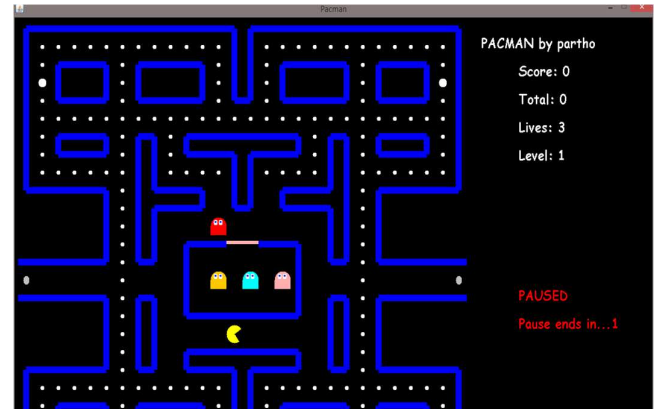


Fig 1: partial simulation of MS pacman game

The partial representation of the Ms. Pac-Man game is as follows:

There are four power pills at four different corners of the game maze when the Pac-Man eats these power pills Pac-Man score increases and Ghosts enter into edible state. The white dots are the nodes where the Pac-Man scores a score of ten for every node it passes. Junction is place where two or more paths are available or there is an intersection of two or more paths.

The game was designed and developed by IntelliJ IDE with JAVA which is described in section III.

V. RESULT & ANALYSIS

In this section we will discuss about the outcome of the research mean how the ghosts are working. Each ghost will search through all the neighbors of the current node evaluating them as next step after a specific time.

VI. CHALLENGES

Problems that we have faced while implementing AI are the following:

- AI sensitivity for changes in games' logic & game play
- Determining proper input & algorithm
- Implementing theory in real time

VII. CONCLUSIONS

This paper expected a methodology to find optimized, minimum cost, conceivable next area for the current area of the Pac Man. The simulation of the game is about like the original game in terms of features. In any case, this version is more troublesome than the original one. In this period of research, we have only focused on execute absolutely pure artificial intelligence on the ghosts which has been effectively done and tested.

Ms. Pac-Man is an energizing and challenging game to create controllers for ghosts. There are a lot of exploration opportunities and much efforts stays to be carried out here. This paper investigated the utilization of artificial intelligence for controlling the Pac-Man, however there are numerous other conceivable methodologies to be implemented to make the ghosts more intelligent. It would additionally be interesting to investigate how genetic programming techniques could be acclimated in the game.

ACKNOWLEDGEMENT

Author would like to give thanks to **Mr Md. Mahmudul Hasan** (*Senior Lecturer of Daffodil International University, CSE*) for his support in this research work.

REFERENCES

- [1] Koza J. Genetic programming: on the programming of computers by means of natural selection. Cambridge, MA: The MIT Press; 1992.
- [2] M. Gallagher and A. Ryan. When will a genetic algorithm outperform hill climbing? In Proceedings of IEEE Congress on Evolutionary Computation, pages 2462 – 2469. 2003.
- [3] J. S. D. Bonet and C. P. Stauffer. Learning to play Pac-Man using incremental reinforcement learning. 1999. [Online]
[URL]
<http://www.ai.mit.edu/people/stauffer/Projects/PacMan>
(Accessed: 08 November 2014)
- [4] Monte Carlo Tree Search for Collaboration Control of Ghosts in Ms. Pac-Man
- [5] A Monte-Carlo approach for ghost avoidance in the Ms. Pac-Man game - Tong, Sung – 2010
- [6] International Journal of Computer Games Technology
Volume 2009 (2009), Article ID 251652 [Online]
[URL]
<http://www.hindawi.com/journals/ijcgt/2009/251652/>
(Accessed: 08 November 2014)
- [7] A*search Algorithm [Online]
[URL]
http://en.wikipedia.org/wiki/A*_search_algorithm
(Accessed: 08 November 2014)
- [8] Artificial Intelligence [Online]
[URL]
http://en.wikipedia.org/wiki/Artificial_intelligence_%28video_games%29
(Accessed: 10 November 2014)
- [9] Euclidian distance calculation [Online]
[URL]
http://en.wikipedia.org/wiki/Euclidean_distance
(Accessed: 10 November 2014)
- [10] Manhattan distance calculation [Online]
[URL]
http://en.wikipedia.org/wiki/Taxicab_geometry
(Accessed: 10 November 2014)