

ARTIFICIAL INTELLIGENCE

[An assignment of INT404 based on a game named Smart – Box]

A Team Work By:

**Garima Kumar Singh**

(A08) – 11811331

And

**Raghav Nagpal**

(A09) – 11811227

K18JE

Assigned By:

**Dr. Rahul**

Professor

Lovely Professional University

Phagwara, Punjab-144411

# CONTENTS

|  |  |
| --- | --- |
| **TITLE** | **PAGE NO.** |
|  |  |
| Acknowledgement | 02 |
| Introduction | 03 |
| Literature Review | 04 |
| Proposed Methodology | 05 to 06 |
| Algorithm Used | 07 |
| Code and Its Explanation | 08 to 10 |
| Result | 11 |
| Conclusion | 12 |
| References | 13 |
|  |  |

# ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

We are highly indebted to **Dr. Rahul Sir** for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We would like to express our gratitude towards our parents & member of Lovely Professional University for their kind co-operation and encouragement which helped us in completion of this project.

We would like to express our special gratitude and thanks to college persons for giving us such attention and time.

Our thanks and appreciations also goes to our friends in developing the project and people who have willingly helped us out with their abilities.

# INTRODUCTION

**A. Genetic Algorithm**

Genetic algorithm is a metaheuristic inspired by Charles Darwin’s theory of natural selection. It belongs to a larger class of evolutionary algorithms. The fittest individuals are selected for reproduction in order to produce offspring of the next generation. The process of natural selection starts with the selection of the fittest individuals from a population which then produce offspring which inherit the characteristics of the parents and will be added to the next generation. If parents have better fitness, their offspring will be better than them and have a better chance of survival. This process of selection based on fitness keeps on iterating and at the end, a generation with the most adapt or fittest individuals will be found. A similar notion is applicable to search algorithms where we consider a set of solutions and try to find out the best ones out of them.

**B. Benefits of Genetic Algorithm**

Genetic Algorithms are mostly used in situations where one can rely on generating high-quality suboptimal solutions to optimization and search problems.

**C. Pygame**

Originally written by Pete Shinners, Pygame is a cross-platform project of several Python modules designed for building video games. It consists of computer graphics and sound libraries which are made compatible to run with the Python programming language

# LITERATURE REVIEW

The term “Artificial Intelligence” include within its scope a wide range of technological processes, making it tricky to understand and hence create policy for. This literature synthesis attempts to provide a broad overview of the key technologies that compose the umbrella term referred to as AI and the key common factors/issues to its different disciplines. As is evident from this literature synthesis, the field of AI offers tremendous promises as solutions and optimisation for a variety of problem statements we face. However, equally importantly, AI also throws up key normative and practical questions of ethics and governance that will play a central role with increased adoption of these technologies. While the some of the tensions between efficiencies promised by AI, and the criticisms pointed to by those advocating greater caution in its adoption may appear irreconcilable, it is important to delve into these points of conflict, so that we are able to rethink some the existing legal and regulatory paradigms, and create new ones if required.

Research on artificial intelligence in the last two decades has greatly improved performance of both manufacturing and service systems. Currently, there is a dire need for an article that presents a holistic literature survey of worldwide, theoretical frameworks and practical experiences in the field of artificial intelligence. This paper reports the state-of-the-art on artificial intelligence in an integrated, concise, and elegantly distilled manner to show the experiences in the field. In particular, this paper provides a broad review of recent developments within the field of artificial intelligence (AI) and its applications. The work is targeted at new entrants to the artificial intelligence field.

# PROPOSED METHODOLOGY

**A. Game description**

Learning Boxes is a game wherein “boxes” (represented as squares) are generated and placed at a specified starting location. A specified “goal” location is placed somewhere on the map of available spaces. The desired outcome is for the boxes to learn the path towards the goal, while getting around obstacles that block the path. If the boxes hit the edges of the map, or run into obstacles, they will be considered dead and stop moving.

**B. Genetic Algorithm for the problem?**

Genetic Algorithms work well for this type of application, where we don’t necessarily know the best path to take, but we have a way to score each individual’s “fitness” to determine how close it is to the goal, and how quickly it made it there.

The fit individuals are chosen to be parents in the next generation, where the parent parameters are randomly selected and mutated in the children. Assuming our fitness scoring algorithm and mutation algorithms are good, this allows us to improve our AI as we iterate through more generations, without ever having to actually know the best path to take ahead of time.

**C. Parameter descriptions**

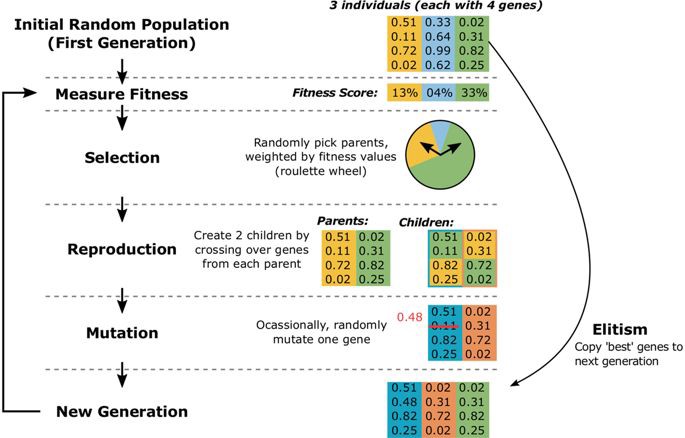
We assign each individual box a position, velocity and acceleration vector, and then keep track of if the box died, whether or not it reached the goal, how many steps it took in the round, its final distance from the goal, and what its calculated fitness score is.

In case when goal is reached: 1/16 + (10000/ (No. of steps) ^2)

In case when goal is reached: (1/ (Distance to Goal) ^2)

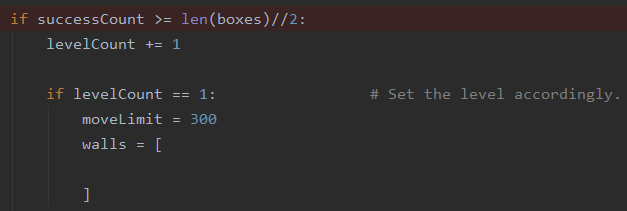
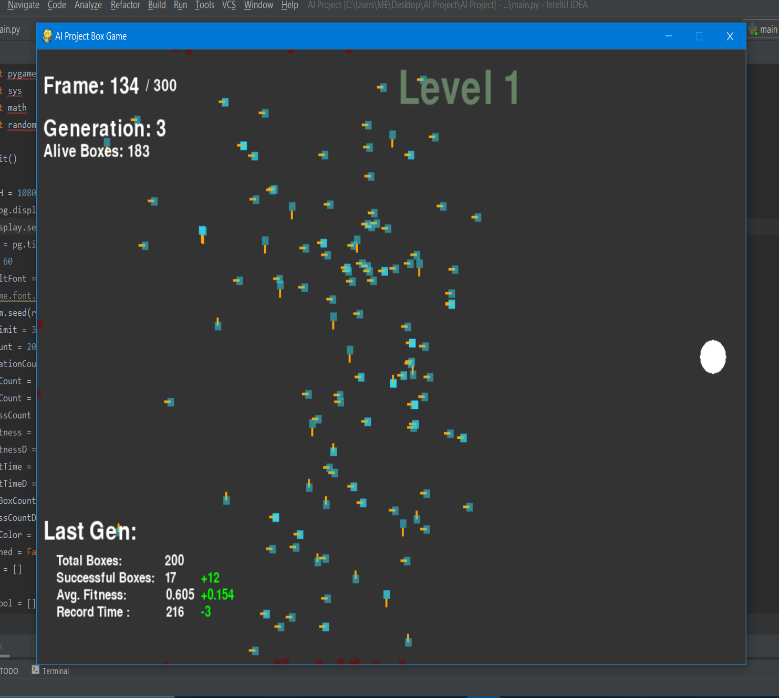
Each box has its own unique “DNA,” which is a list of vectors for the box to follow. These vectors are essentially applying forces to the individual boxes, moving them through space. Initially each box DNA is completely random, and then we score each individual box after the round using the fitness algorithm discussed above. We then select the best rockets to be parents to the next generation, where we take traits from the parents and randomly select parent traits to vary for each child. We then iterate through the process with the children and choose the fit children to be parents to the next generation. This process is repeated until we find an optimized model where we are no longer improving from generation to generation.

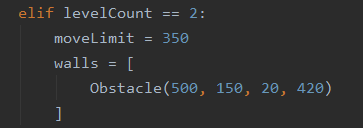
# ALGORITHM USED IN SOLUTION



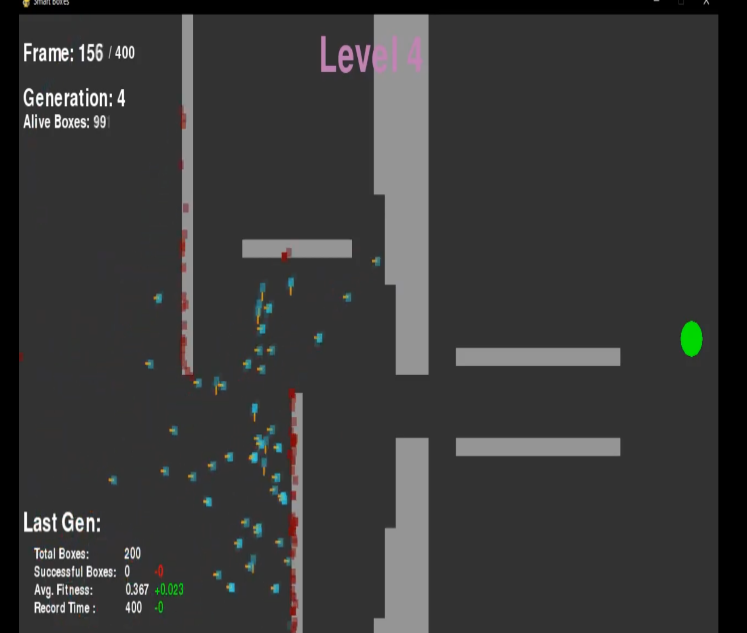
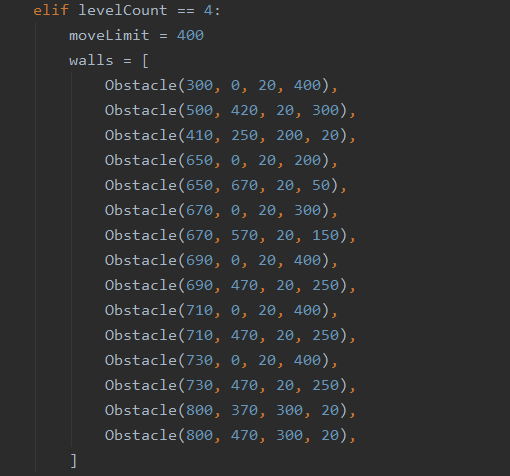
# CODE AND EXPLANATION

Link: <https://vimeo.com/user112578625/review/406836791/505d3790a3>





# 



# 

# RESULT



# CONCLUSION

Our genetic algorithm was very successful in finding an efficient path to the goal, and even with more complex obstacles in the way we were still able to get an AI that completed the course efficiently. By adjusting our fitness function, as well as our method of parent selection and trait randomization in the child generation, we found a model that fit our needs and was also efficient in how many generations it took to find an acceptable AI.

But I and Garima is still working on this project & are looking forward to add additional cool features in this Game!

Hopefully after learning Machine Learning in these vacations we will be able to make some cool additions in this project.

# REFERENCES

**Geek for Geeks**

<https://www.geeksforgeeks.org/genetic-algorithms/>

**Genetic Algorithm tutorial: Code Bullet, “How AIs learn part 2**

**Coded example,”**

YouTube Available: <https://www.youtube.com/watch?v=BOZfhUcNiqk>

**PyGame**

<https://pythonprogramming.net/pygame-python-3-part-1-intro/>