



**ADVANCED ARTIFICIAL INTELLIGENCE
CMP9132M
ASSESSMENT 2**

RAKESH KUMAR ELUMALAI (25954263)

Abstract:

Arcade games, as a genre, are truly hybrids that combine a variety of gaming styles. Short levels with easy and intuitive controls are common in arcade games, which swiftly rise in difficulty. Game players are effectively renting the game for the duration of their game avatar's life. To make this business model successful, the game's difficulty must be high enough to cause players to reach a game-over state while yet being entertaining or addicting enough to keep them playing. In most parts of the world, the arcade is plainly gone, as what was once great technology is now available on your phone. Arcade games are also referred to as PC or console games if they have the same characteristics as true arcade games, such as Controls that are intuitive and straightforward to use, as well as basic physics. It has short levels that gradually become more harder as the game goes. A focus on gameplay rather than on content or story.

Introduction:

Arcade video games were introduced in the 1970s, with Pong being the first commercially successful game. Electronic or computerized circuitry takes input from the player and displays it on an electronic display device, such as a monitor or television set. The gamer's performance in playing a game is usually defined unambiguously by the game's score or outcome, which is specified by an explicit set of rules. From single-player puzzles to two-player board games to massively multiplayer video games, games are extremely versatile and vary tremendously in complexity. Techniques developed expressly for game play can frequently be easily adapted to other domains, considerably expanding the range of applications for such techniques.

In this task we are going to make tallon to move in a way that it avoids meanies and pits to collect maximum number of rewards with the help of Markov decision process.

Similar Game:**PAC MAN**

Pac-Man was first released in May 1980 and has since become one of the most iconic games in the gaming industry, selling over 100,000 units in its first year. The game was originally known as 'Puck Man' in Japan before being renamed for the foreign market. While virtual reality gaming software has just lately gained popularity thanks to devices like the Oculus Rift, Pac-Man VR was first launched in 1996. Although the game didn't have a significant impact on the industry at the time, it does show how the iconic franchise has strived to modernize gaming throughout its growth. Pac-Man was still available to play despite the evolution of video game devices. Pac-Man 256 was developed and released by BANDAI NAMCO in August 2015, and it was compatible with the PlayStation 4 and Xbox One, as well as PCs and Macs.



Fig: PAC-MAN (Arcade Game)

While many of the genre's titles, such as Pac-Man, have made the transition from famous arcade machines to current platforms, their stylistic effect on broader

audiences has risen. Aside from console expansions, numerous online casinos have included vintage arcade features into their latest releases. Along with games like Arcade Bomb, casino party's famed melon madness blends free spins with conventional arcade graphics to offer prospective players a throwback progressive jackpot title. Furthermore, because the platform is compatible with both Android and iOS devices, the arcade genre has never been more accessible.

Game Design:

In this game the tallons are made to move from the initial point to the point where it collects the maximum number of bonus points by avoiding obstacles and the pit on its way. In this Markov decision process method is used so that the talon moves towards more bonus points.



Fig: The Game

The utility is computed for each and every value, and this is used in MDP to identify the optimum decision. For the tallons choice process, this formula is utilised to calculate the optimal utility value.

Value iteration:

Value iteration is a technique for determining the best MDP and policy values.

```
mdptoolbox.util.check(P2, R2)
vi2 = mdptoolbox.mdp.ValueIteration(P2, R2, 0.99)
```

The value iteration starts with an arbitrary value and continues until it achieves an ideal value, at which point the tallon decision is made properly.

Policy Iteration:

Policy iteration is the process of estimating and evaluating policies.

```
pi2 = mdptoolbox.mdp.PolicyIteration(P2, R2, 0.9)
```

Every time a new policy is generated in the improvement section, and every time a new policy is created in the policy assessment section, an initial policy is created that is used in the improvement. If no further progress is made, the iteration will end.

Q-Learning:

Q-Learning is a method of active reinforcement learning that does not require the usage of a model.

```
r12 = mdptoolbox.mdp.QLearning(P2, R2, 0.9)
```

The tallon moves in four different directions which is decided based on the movement of the monsters and also depends on the number of pits involved in the environment.

Markov Decision Process(MDP):

Here we use different Markov decision making process which implicit the process in easy way. In this case, the action A influences the probability of the process advancing to a new state Sa. MDP's solution or policy is expressed as. However, we'd need the best policy possible.

$$\pi^* = \arg \max_{\pi} U^{\pi}(s)$$

After that, the policy's utility should be calculated. The predicted utility value is used to compare policies.

$$U^\pi(s) = E \left[\sum_{t=0}^{\infty} \gamma^t R(S_t) \right]$$

At time t , the agent appears at state S_t .

Bellman equation:

If the state space and action space are both continuous, the optimum criterion can be found by solving the Bellman equation.

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s'|s, a) U(s')$$

Partial Observability:

A partially observable Markov decision process (POMDP) is a Markov decision process that is more general (MDP). A POMDP is a model of an agent decision process in which the system dynamics are thought to be defined by an MDP, but the agent is unable to perceive the underlying state directly. Instead, it must keep track of the underlying MDP and a sensor model (the probability distribution of distinct observations given the underlying state). POMDP's policy is a mapping from the observations (or belief states) to the actions, unlike MDP's policy function, which maps the underlying states to the actions. The POMDP framework can be used to simulate a wide range of real-world sequential decision processes.

Utility values and policies are found in way that the tallon could move towards bonus points by one by one step ahead.

Fully observability:

In fully observability the system is fully visible and the obstacles can be avoided easily and it makes tallons to collect bonus points easily.

Reinforcement learning :

Reinforcement Learning (RL) is a teaching method in which a student learns to behave in an interactive environment by performing actions and receiving rewards for them.

This method is used in the MDP to find the cumulative rewards. This is also used in different fields like machine learning and artificial intelligence. In reinforcement learning, the Markov Decision Process (MDP) is a mathematical framework for describing an environment.

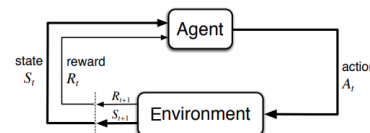


Fig: Reinforcement Learning cycle

Alternate Methods for Decision making:

By calculating expected value,

$$E(X) = \sum_k k \Pr(X = k)$$

With the help of the expected value we can obtain utilities and optimum value for decision making.

But in real world we obviously don't know the utility value hence we have to calculate expected utility value.

$$E(u(a)) = \sum_{s' \in S_a} u(s') \cdot \Pr(s_a = s')$$

And also there are some more alternative methods for decision making.

Conclusion:

In this task we have used different types of Markov Decision process for decision making. We have built a program to move tallons towards reward point avoiding obstacles and pit. The learning outcomes are about how effective the MDP method is to find the solution and decision making.

References:

1. DiMarzio, J.F., 2012. What Is an Arcade Game?. In *Android Arcade Game App* (pp. 7-11). Apress, Berkeley, CA.
2. Rohlfshagen, P., Liu, J., Perez-Liebana, D. and Lucas, S.M., 2017. Pac-man conquers academia: Two decades of research using a classic arcade game. *IEEE Transactions on Games*, 10(3), pp.233-256.
3. Newman, J., 2016. Mazes, monsters and multicursality. Mastering Pac-Man 1980–2016. *Cogent Arts & Humanities*, 3(1), p.1190439.
4. Bowman, R.F., 1982. A "Pac-Man" theory of motivation: Tactical implications for classroom instruction. *Educational technology*, 22(9), pp.14-16.
5. Sigaud, O. and Buffet, O. eds., 2013. *Markov decision processes in artificial intelligence*. John Wiley & Sons.
6. Koller, D. and Parr, R., 2013. Policy iteration for factored MDPs. *arXiv preprint arXiv:1301.3869*.
7. Dai, P. and Goldsmith, J., 2007, January. Topological Value Iteration Algorithm for Markov Decision Processes. In *IJCAI*(pp. 1860-1865).
8. Carmona, R., Laurière, M. and Tan, Z., 2019. Model-free mean-field reinforcement learning: mean-field MDP and mean-field Q-learning. *arXiv preprint arXiv:1910.12802*.
9. Rachelson, E., Garcia, F. and Fabiani, P., 2008. Extending the Bellman equation for MDPs to continuous actions and continuous time in the discounted case.
10. Geibel, P., 2006, September. Reinforcement learning for MDPs with constraints. In *European Conference on Machine Learning* (pp. 646-653). Springer, Berlin, Heidelberg.