**Sir M Visvesvaraya Institute of Technology**

**Bengaluru-562 157**

**Department of Computer Science and Engineering**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Title :** Intelligent Autonomous Game Bots using Deep Q Reinforcement   |  |  | | --- | --- | | Learning. |  | | | | |
| **Abstract** | | | |
| Deep Reinforcement Learning has recently become a really hot area of research, due to the huge amount of breakthroughs in last couple of years. By feeding sufficient data into deep neural networks, it is often possible to learn better representations than handcrafted features. Our goal is to connect a reinforcement learning algorithm to a deep neural network which operates directly on RGB images and efficiently process training data by using stochastic gradient updates. The main idea of DQN is to compress Q-table by learning to recognize in-game objects and their behaviour, in order to predict delayed reward for each action given the state.  The method of learning goal-directed behaviour in environments with sparse feedback is a major challenge for reinforcement learning algorithms. The primary difficulty arises due to insufficient exploration, resulting in an agent being unable to learn robust value functions. Intrinsically motivated agents can explore new behaviour for its own sake rather than to directly solve problems. Such intrinsic behaviours could eventually help the agent solve tasks posed by the environment. We present a parallel-DQN, a framework operating at different temporal scales, with intrinsically motivated deep reinforcement learning. A top-level value function learns a policy over intrinsic goals, and a lower-level function learns a policy over atomic actions to satisfy the given goals. This provides an efficient space for exploration in complicated environments. | | | |
| **Batch No :** | |  | |
| **USN** | **1MV13CS004** | **Name** | **Adarsh Trivedi** |
| **1MV13CS005** | **Aditya Roy Choudhary** |
| **1MV13CS014** | **Amit Asish Bhadra** |
| **1MV13CS047** | **Karan Saxena** |

**Salient Features:**

1. We use OpenAI gym to get the environment (here, a game) details. This avoids the problem of designing the game itself.
2. Abstracting away the environment from the logic, we are able to make our code Environment Agnostic.
3. We also tackle Exploration vs Exploitation by parallelizing the algorithm with multiple agents, and finally settling for the sweet spot in terms of the number of agents required.