Department of Computer Engineering

MoonAI

Prey-Predator Simulation for Evolutionary Algorithm Training

Team Members

Emir Irkılata
Caner Aras
Oğuzhan Özkaya

Supervisor

Ozan Zorlu

Web Site

Motivation

Manually designed training scenarios and real world data do not scale for artifical intelligence training.

Objective

MoonAI aims to develop a simulation environment to research evolutionary algorithms without real-world data required.

Approach

Simulation

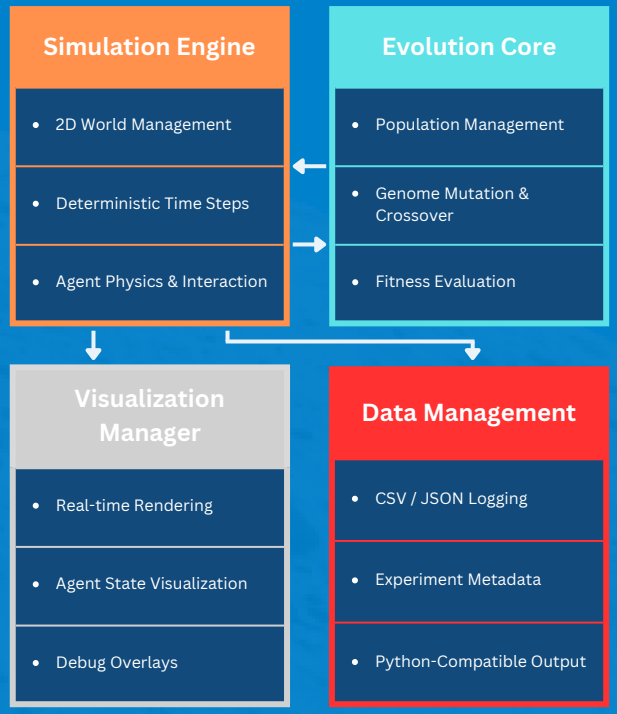
Simulation with evolutionary algorithm controlled predator and prey agents.

Output

All of the simulation and genetic data of the evolutionary agents recorded for analysis.

SYSTEM ARCHITECTURE

Modular, GPU-accelerated evolutionary simulation platform



Why NEAT?

NEAT evolves both neural network weights and topologies, enabling complex behaviors to emerge without manual architecture design.

Genome → Neural Network → Agent Behavior → Fitness → Evolution

This architecture decouples simulation logic from evolutionary strategies, enabling controlled experime ntation and reproducible research.

Engineering Implementation

Technical feasibility and computational strategy

Hardware / Software Mapping

The system uses a heterogeneous computing architecture to maximize performance.

CPU
- Control logic
- Physics simulation
- User interface

GPU (CUDA)
- Neural inference
- Fitness evaluation
- Parallel computation

Sequence of Operation

1. Initialization Load configuration and initialize GPU
2. Main Loop Neural inference → Physics update → Rendering
3. Evolution Phase Selection and mutation

Technologies

Python · CUDA · SFML · C++17

The architecture ensures real-time performance by delegating intensive workloads to the GPU.