WP Artificial Intelligence and Software Agents

LaserTag Group Competition: Assignment

05. April 2022

LaserTag: Game Description

LaserTag is a multi-agent game simulation in which four agent teams, each made up of three agents, compete against each other. Team members need to coordinate with each other and be careful not to become overpowered by agents from other teams. To play the game well, agent behavior needs to be designed intelligently such that each agent interacts with its environment, its team members, and its opponents in meaningful ways to work towards the common goal of the team.

Your Assignment

1. **Set up the project:** LaserTag is developed and written in MARS C#. Please check out the project's GitLab repository and set up the project in Rider. The directory **LaserTagBox** contains the game.
2. **Review the documentation:** The LaserTag documentation (PDF) can be found in the **Documentation** directory of the repository. Please use it as a reference guide to familiarize yourself with the model's concepts and mechanisms and while designing your AI.
3. **Study the code:** The source code includes the setup for the four agent teams that will compete in one simulation. Review and experiment with the code to get a feeling for what the possibilities are and how agents behave when calling different methods.
4. **Write your AI:**
   - Create a class that inherits from `PlayerMind`. Your agents' mind (i.e., its AI) will be written inside this class. From here, it can access the `PlayerBody`, which represents the agents' physical form. Call methods defined in `PlayerBody` to shape the agents' behaviors and routines.
     - `ChangeStance2(Stance)`
     - `ExploreBarriers1() : List<Position>`
     - `ExploreDitches1() : List<Position>`
     - `ExploreEnemies1() : List<EnemySnapshot>`
     - `ExploreHills1() : List<Position>`
     - `ExploreTeam() : List<IPlayerBody>`

- `GetDistance(Position) : int`
- `GoTo(Position) : bool`
- `HasBeeline1(Position) : bool`
- `Reload3()`
- `Tag5(Position)`

The numbers at the end of some of the methods indiciate the number of `actionPoints` required to execute the method. Please see the documentation for further details.

- Your code must meet the following requirements:
  a) follow all rules listed in the section "Rules" of the Documentation
  b) no simulation-external information may be loaded into the simulation at runtime (your agent must have an empty constructor)
  c) no loops that are known not to terminate within a reasonable time (example: `while(true)`)
  d) The use of `PropertyDescription` tags (for loading external information into your agents at runtime) is not allowed.

5. **Submit your AI:** The deadline for submitting your code for the competition is 23:59 CET on 18 April 2022. Only your class inheriting from `PlayerMind` (and its associated classes, if any) will be considered. Please submit your code via EMIL.

Goal of the Game

The objective of the game is to gain the highest score of all teams. Every successful tag provides game points.