WP Artificial Intelligence and Software Agents

LaserTag Group Competition

June 15, 2020

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 be careful not to be overpowered by agents from other teams. To play the game well, agent behavior needs to be designed intelligently such that each agent interact 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 communication:** Please join the channel # lasertag in MARS Explorers on Slack. All announcements, communication, and technical discussions about LaserTag and the competition will occur via this channel.
   - There is a pinned message in the channel. Please have one of your group members register your group for the competition by responding to the message with the names of your group members by 7 June 2020. This will help us organize the competition.
2. **Set up the project:** LaserTag is written in the MARS DSL. Please check out the project's GitLab repository and set up the project in Eclipse.
3. **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.
4. **Study the code:** The source code includes the setup for the four agent teams that will compete in one simulation. (The four agent types `Red`, `Yellow`, `Green`, and `Blue` are identical, so working in one of them is enough). Consider experimenting with the code to get a feeling for how your agents behave when calling different methods.
   - Note: if the IDE lags due to long build and compilation times, commenting out agents that you don't work in (except for their attributes and passive methods) and/or deactivating the option "Build Automatically" under the Project header of the menu bar might help.
5. **Write your AI:**
   - Write code for one `tick` method (see the TODO in the source code). The agents' behavior should be defined in the way you think will lead to your agent team's victory.
   - Your code must meet the following requirements:

a) only calls to methods labeled with "USER METHOD" in a comment above the method signature (please see the documentation PDF for more details):
   - `exploreEnv(string)`
   - `exploreTeam() : Agent[]`
   - `exploreEnemies() : Tuple<Agent[], Agent[], Agent[]>`
   - `goTo(real, real): bool`
   - `changeStance(string)`
   - `tag(Agent)`
   - `tag(real, real)`
   - `reload()`
   - `hasBeeLine(real, real)`

b) no use of MARS DSL keywords such as `distance`, `explore`, `kill`, `move`, `nearest`, `pos at`, and `spawn`

c) agent attributes may be queried, but may **not** be set. The only attribute whose value may be updated during the `tick` method is `stage`

d) no use of setter methods pertaining to the `battleground`layer. `DimensionX`, `DimensionY`, and getter methods pertaining to the `battleground` layer may be used

e) no loops that are known not to terminate after a reasonable time (example: `while(true)`)

6. **Submit your AI:** The deadline for submitting your code for the competition is 20:00 CET on 22 June 2020.
7. **Attend the competition:** As announced in the course calendar, the competition will begin 9:00 CET on 24 June 2020. The code your group wrote for your agent team's `tick` method will be inserted into one of the four agent types. (Any attributes and methods you defined will not be considered). In a tournament-style setup, simulations with four teams at a time will be run. The three highest-scoring teams will advance to the next round.

Goal of the Game

Please see the section titled "Objective" in the LaserTag documentation for a description of the goal of the game. A separate document describing the tournament setup and rules will follow soon.

Questions and Feedback

Please submit any questions, feedback, feature requests, and bug reports to the Slack channel mentioned above. We (Daniel Osterholz, Xavier R Adams-Stewart, and Nima Ahmady-Moghaddam) will do our best to address any issues as quickly as possible.

Happy coding and good luck! :)