

Software Engineering Project

Mourjo Sen, Muhammed Raheel

M1 International 2013-2014

University of Nice

This is a readme file to introduce the SE project.

Energy of Creatures

The Bouncing creature and the Smart creature have a new attribute called “energy”. The color of a creature is proportional to the energy of the creature. A creature is born with high energy (red) and its energy slowly fades away and so its color becomes bluish with time, unless it passes over an energy point. When a creature’s energy is zero, its color becomes blue. *See figure 1.*



Figure 1: The color of a creature denotes its energy - red is high and blue is low energy

Energy Points

The environment consists of a fixed number of randomly selected points called “energy points”. Every energy point has an influence region (denoted in black) – when a creature passes over this region, its energy increases. There is another region (outside the influence region) denoted by a grey circle called the “event horizon”. When a smart creature or a bouncing creature is inside the event horizon, the creature can feel the presence of the energy point and thus can reorient itself to pass over the influence region. *See figure 2.* Creatures can sense the energy point even if it is not in its field of vision. It is like the force of gravity that whether we are looking or not, we can feel it.

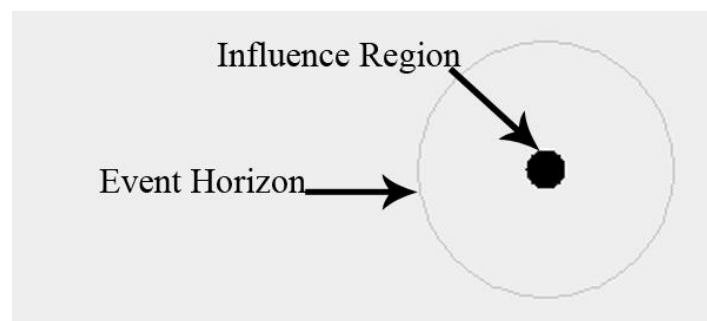


Figure 2: The influence region (black) and the event horizon (grey circle)

The Mad Creature

The mad creature is a creature whose actions and behavior depends on the plugin used to run it. The Mad Creature is called so because it has a lot of random, unpredictable behaviors. The design pattern used is the strategy pattern. We have a strategy object (IBehavior Interface) in the MadCreature class that chooses the concrete BehaviorPlugin loaded at run time. The mad creature does not use the energy parameter – its energy is always full. Hence its color **does not** denote its energy. There are six plugins available:

1. No special functionality: This plugin is the same as a stupid creature. It has no random or “mad” behavior. It does not interact with energy points or walls.
2. Color change: This plugin makes the mad creature change color randomly. It also does not interact with energy points or walls.
3. Avoid walls: This plugin makes the mad creature avoid walls. When it goes near a wall, it turns around. It does not interact with energy points.
4. Random speed and direction: This plugin makes the mad creature change direction and speed randomly with a probability of 1/50 for change in each speed and direction. It also does not interact with energy points or walls.
5. Avoid energy points: This plugin makes the mad creature avoid energy points. Whenever the creature senses an energy point, it turns around to run away. It also changes direction with a probability of 1/50 to avoid loops in avoiding energy points. It does not detect walls.
6. All features: This plugin makes the mad creature incorporate all the above characteristics 2-5. It randomly changes color, avoids walls, randomly changes speed and direction and avoids energy points. It is because of these characteristics that it is called a “mad” creature. *Since the mad creature has erratic behavior, its functioning maybe better understood if the number of creatures is reduced to 5 or less.*

Plugins

Plugins are available only for the Mad Creature. The available mad creature plugins are stored in the directory named “MadCreaturePlugins”. It should be noted that the Eclipse/JAR project does not contain the source code (.java file) of the BehaviorPlugin class. Source files of each version of the BehaviorPlugin class are present with the corresponding .class file in the subdirectories of the MadCreaturePlugins directory. The names of the subdirectory will indicate which of the 6 plugins it is.

To run the MadCreature plugins in Eclipse, the BehaviorPlugin.class file has to be copied to the **myplugins/repository/creatures** directory of the eclipse project and then the “Reload Plugins” button has to be pressed to load the new plugin. By default, the AllFeatures plugin is loaded in the **myplugins/repository/creatures** directory.

The jar file is contained in a directory named “JAR Project”. Inside the directory, there is a **myplugins/repository/creatures** directory. The plugin (BehaviorPlugin.class) must be placed in this directory.

Changes in the Smart Creature

The SmartCreature's functionality has not changed much except that when it senses an energy point, it changes its direction towards it. When the SmartCreature is in a group, however, the creature decides to change its direction towards the energy point if its energy is too low, otherwise it prefers to stay in the group. Also, in a large group, a creature loses less energy.

Tests

There are three new test classes:

1. EnergyPointTest: This class tests the behavior of SmartCreature and BouncingCreature near an energy point.
2. MadCreatureTest: This class tests the avoiding wall feature of the MadCreature.
3. MadEnergyPointTest: This class tests if the MadCreature avoids energy points correctly.

Please note that the MadCreature's tests are done with the AllFeatures plugin. Also, there is 3/50 chance that the MadCreatureTest and MadEnergyPointTest will fail. That is because of the random erratic change in direction and speed of the MadCreature.

The other new classes like EnergyPointVisualizer and ParameterWrapper are used to obtain the required functionalities and there is no explicit computation done by them that can be tested. The success of EnergyPointTest, MadCreatureTest and MadEnergyPointTest denote that they are working also.

Notes

The src folder, test folder etc. are in the **Eclipse Project/SEProject** directory.

Creatures (Smart and Bouncing) are now initialized with RED (full energy). As for the MadCreature, the constructor of the MadCreature randomly chooses a color for itself. So the ColorStrategy was not required in the project.

Resizing of the running application window works fine but when maximized, there are a few problems with the visualization. We tried our best to solve this problem but could not. The simulator works perfectly when its size is changed by dragging with the mouse.