Disaster Simulation: An AnyLogic Agent-Based Approach

Example:
Flood simulation and evacuation
GIS Environment

Ali Asgary
ADERSIM
York University
2017
Lesson 5: Defining Agents’ Behavior 3- Person agent

• Defining population of agents’ behavior using statechart
• Emergency evacuation simulation.
We want people to receive the flood warning issued by the sensor and react to it by moving to evacuation zone if their houses are located in the flood zone. To do this we use statecharts to define different states of the Person agent.

We assume that Person has two main states:
- Normal: when water level is below certain level (no flood).
- Flooding: when water level is above the normal level and flood warning is issued.

As soon as people receive the warning and their homes are located in the flood zone they start preparing and then moving to the pre designated evacuation zone. They will remain there till flood warning is lifted. When flood warning is lifted by the sensor, people who have taken shelter in the evacuation zone return to their homes.
Open Person window
Double click on Person to open its window.
Expand Statechart tools

Click on the Palette tab and click on Statechart tools.
Creating statechart for Person

To create a statechart for Person you need to drag and drop a Statechart Entry Point to your Person window.
Adding states

You can add a state by dragging and dropping a state from the Statechart tools to your Person space.

Make sure that the state is connected to the Statechart Entry Point.
Change the state name to Normal

Click on the state that you just created and in the property window change its name to Normal
Change the state color

Using the state property window change the Fill color to lime.
Transition from Normal state to Flooding state
If Person receives flood warning from the Sensor its state changes from Normal to Flooding.
For this to happen we drag and drop a transition to the Normal state as shown:
Define properties of the transition
Change the Name to: flooding
Change the Triggered by to: Message
Change Message type to: String
Select Fire transition: On particular message
For the Message write: “flood warning”
Add the Flooding state
Drag another state from the Statechart tools and drop it into the Person window and attach it to the flooding transition as shown.

Ali Asgary, ADERSIM, York University, 2017
Change the properties of the Flooding state

Change the name to: Flooding
Change the Fill Color to blue

Make sure that you create a large area for the Flooding state.
Adding Warning, Evacuating and Evacuated states inside the Flood state

Once a Person is in the Flooding state he/she may be at Warning state, Evacuating state or Evacuated state. Therefore we will add these three states inside the Flooding state.
Drag and drop a state inside the Flooding state and change its properties as shown:

Name: Warning

Entry action:
shapeBody.setFillColor(yellow);
Make sure that you extend the Flooding transition to be connected to the warning state.
Add transition from Warning to the next state (Evacuating)

Change the Triggered by to: Timeout

Add the following code for the Timeout: uniform (1, 30).

This means that each Person randomly takes between 1 to 30 minutes to start evacuating.
Drag and drop another state from the statechart pallet to the Person window as shown here:

Change its name to: Evacuating
Change its Fill color to: Red
Add the following lines of codes to the Entry action section:

```javascript
distance = main.river.contains(this.getLatitude(), this.getLongitude(), 200.0);
if (distance == true)
{
    this.moveTo(main.evacuationArea);
    shapeBody.setFillColor(orange);
}
main.evacuating++;
```
In the previous slide we use the code to identify people who are living within 200 meters distance of the river.

```java
distance = main.river.contains(this.getLatitude(), this.getLongitude(), 200.0);
```

Those who are within the 200 meters distance evacuate the area and move to evacuation zone.

```java
if (distance==true)
{
    this.moveTo(main.evacuationArea);
}
```

We also change the color of these persons to orange

```java
shapeBody.setFillColor(orange);
```

We also create a variable in the main calling it evacuating to measure number of people who are evacuating
Drag and drop another transition to the Person window and attach it to the Evacuating state as shown. Change its triggered by to : Agent Arrival
Drag and drop another state to the Person window and attach it to the hanging transition as shown. Change its name to: Evacuated Change its Fill color to: green Add the following codes to the Entry action:

```
shapeBody.setFillColor(green);
main.evacuated++;
```

First line changes the color of the person animation to green. Second line adds this person to the people who have evacuated.
Drag and drop a transition to the Person window and attach it to the Warning state and Normal state as shown.

Change its triggered by to: Message

Change Message type to: String

Select On particular message for Fire transition

Change the Message to: “Normal”

Type the following line of code in the Action section:

`shapeBody.setFillColor(black);`
Drag and drop a transition to the Person window and attach it to the Evacuating state and Normal state as shown.
Change its triggered by to: Message
Change Message type to: String
Select On particular message for Fire transition
Change the Message to: “Normal”

Type the following lines of code in the Action section:
```
thisto.moveTo(Y, X);
shapeBody.setFillColor(black);
main.evacuating--;
```
Drag and drop a transition to the Person window and attach it to the Evacuated state and Normal state as shown.

Change its triggered by to: Message

Change Message type to: String

Select On particular message for Fire transition

Change the Message to: “Normal”

Type the following lines of code in the Action section:

```java
this.moveTo(Y, X);
shapeBody.setFillColor(black);
main.evacuated--;
```
• Drag and drop a variable from the Palette and change its name to: distance.
• Change its type to: boolean
• Set its initial value to: false
• Drag and drop another variable from the Palette and change its name to: X
• Change its type to: double
• Set its initial value to: getLongitude()
• Drag and drop another variable from the Palette and change its name to: Y
• Change its type to: double
• Set its initial value to: getLatitude()
• Double click on the Main and open the main window
• Drag and drop a variable from the Palette and change its name to: evacuating
• Change its type to: int
• Drag and drop another variable from the Palette to Main and change its name to: evacuated
• Change its type to: int
While on the Main. Click on the river line and change its name from gisRoute to river.
• While on the Main. Click on the gisRegion and change its name from gisRegion to: evacuationArea
Run the model
Results

If you do not see any error, you will see that after flood reaches to the Sensor, sensor symbol increases. It also sends the warning to people that we will see them in the next lesson.

People who receive the warning are shown in yellow and then people who are close to the river start evacuating.
Results

- People who are evacuating move to the evacuation area
• This was another way of creating behavior for an agent using statecharts.
• We will create more complicated statecharts in the future.
• We will explain how to extract data and graph from this simulation in lesson 5.
• Save your project.