Observation

EES 4760/5760

Agent-Based & Individual-Based Computational Modeling

Jonathan Gilligan

Class #10: Tues. February 13 2018

Observation

The **observation** design concept is about what you (the scientist) want to observe about your model in order to learn something relevant to your research question.

Some ways to observe your model include:

- The model view
- Monitors on the interface
- Agent-monitors (`inspect-turtle`, `inspect-patch`, `watch-turtle`, `follow-turtle`)
- Set turtle or patch labels (`set label elevation` or `set plabel count turtles-here`)
- Set turtle or patch color (`set color blue` or `set pcolor green`)
- Set turtle shape or size (`set shape "butterfly"`, `set size 4`)
- Plots and histograms on the interface
- Output to output window, command center, or file (`print`, `show`, `type`, `write`, `file-print`, `output-print`, etc.)
- BehaviorSpace experiments

Observing the Butterfly Model

- Download butterfly model https://ees4760.jgilligan.org/models/class_10/butterfly_class_10.nlogo and elevation file https://ees4760.jgilligan.org/models/class_10/ElevationData.txt, save them in the same folder, and open the butterfly model in NetLogo.
- Go to the code tab and look at `update-display` and `color-patches`

Observations in the View

- Go to `move` and change

```
ifelse random-float 1 < q
[
  move-to max-one-of neighbors [elevation] ; Move uphill
]
[
  move-to one-of neighbors ; Otherwise move randomly
]
```

  to

```
ifelse random-float 1 < q
[
  move-to max-one-of neighbors [elevation] ; Move uphill
  set color red
]
[
  move-to one-of neighbors ; Otherwise move randomly
  set color blue
]
```

Observations in the View

- Edit `update-display`:

```
to update-display
  color-patches
  ifelse show-butterflies?
  [ ask turtles [show-turtle]]
  [ ask turtles [hide-turtle]]
  ifelse show-labels?
  [ ask turtles [set label elevation]]
  [ ask turtles [set label ""]]
end
```

- Set `num-butterflies` to 0 or 5, press `setup`, and run the model
- When you set `label` to a variable:
  - It gets the current value of the variable.
  - It won’t update automatically when the variable changes.
    - So you have to update it by re-setting it:
    - e.g., Update `label` in `update-display`, and call `update-display` every tick

Observations in the View

- Edit `color-patches`:

```
to color-patches
  ifelse patch-coloring = "elevation"
  [
    let max-elevation max [elevation] of patches
    let min-elevation min [elevation] of patches
    ask patches [ set pcolor scale-color green elevation min-elevation max-elevation]
  ]
  [
    let max-turtles max [count turtles-here] of patches
    let min-turtles 0
    ask patches [ set pcolor scale-color cyan (count turtles-here) min-turtles max-turtles]
  ]
end
```

- Turn off `show-labels?` and set `num-butterflies` to 50
- Setup and run the model
- Clear the paths, turn `show-butterflies?` off, and press `update-display`

Adding monitors

- Write a new reporter:

```
to-report fraction-crowded
  let crowd-count sum [count turtles-here] of patches with [count turtles-here >= 4]
  report crowd-count / num-butterflies
end
```

- Add 3 monitors to the interface:
  - **Mean elevation** gets reporter

    ```
    mean [elevation] of turtles
    ```

  - **Mean turtles** gets reporter

    ```
    mean [count turtles-here] of patches with [any? turtles-here]
    ```

  - **Fraction crowded** gets reporter

    ```
    fraction-crowded
    ```

Monitoring agents

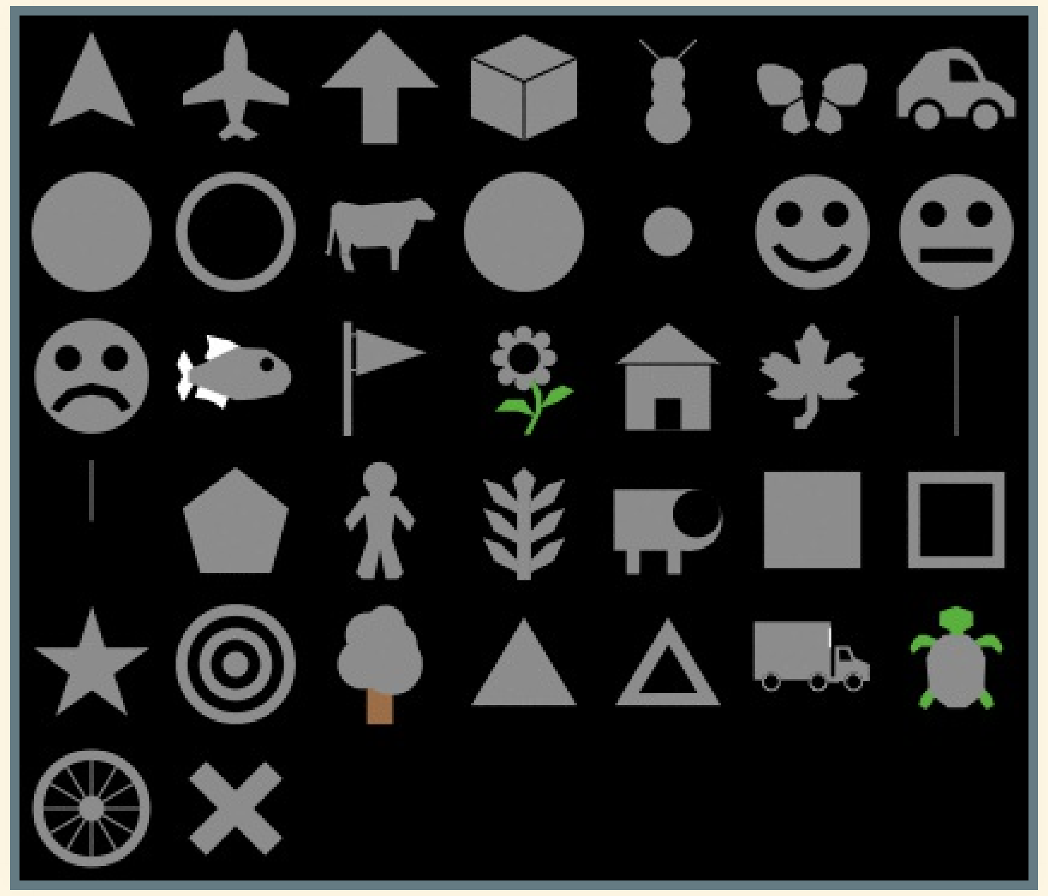
- Set `num-butterflies` to 10 and setup.
- Right click your mouse near a turtle and choose “Inspect Patch”
- In the patch-inspect window, right click on a turtle and choose “Inspect Turtle #”
- In the turtle window, click “Watch me”
- Click “step” a few times or let the model “go” for a while
- Close the monitoring windows
- Right click on the world and choose “Reset perspective”
- Right click on a turtle and choose “Turtle #/Follow turtle #”
- Click “step” several times, or “go”
- Setup the model again
- Type `inspect turtle 5` at the observer line.
- Type `watch turtle 3` at the observer line.

Adding plots and histograms

- Add three plots:
  - **Patch density**:
  - Edit default pen: set mode to “Bar”
  - Set update command to `histogram [count turtles-here] of patches`
    - Set X min = 1, X max = 20, Y min = 0, Y max = 10
    - Set X label to “Turtles per patch”, Y label to “count”
  - **Fraction crowded**:
    - Set plot command to `plot 100 * fraction-crowded`
    - Set Y min = 0, Y max = 100
    - Set X label to “Tick”, Y label to “Percent”
  - **Mean elevation**:
    - Set plot command to `plot mean [elevation] of turtles`
    - Set Y min = 500, Y max = 500
    - Set X label to “Tick”, Y label to “Elevation”
  - When you put plotting commands into your code:
    - You usually don’t want to plot inside an `ask turtles` or `ask patches` command
    - Update plots once per tick (in `to go` or in a submodel called from `to go`)
    - If you have multiple plots, be sure to use `set-current-plot` before plotting.

Setting turtle size and shape

- Turtle shapes:



- See Shapes Editor in the NetLogo manual for details

```
ask turtles [set shape "butterfly"]
```

File output

- First open file (generally, check whether a file exists, and delete it). It’s also a good idea to write a header with the names of the columns.

```
if (file-exists? "my_test_output.csv")
[
  carefully
    [ file-delete "my_test_output.csv" ]
    [ print error-message ]
]
file-open "my_test_output.csv"
file-type "id,"
file-type "tick,"
file-print "elevation"
```

- Then at each tick, you could write data:

```
file-open "my_test_output.csv"
ask turtles [
  file-type word who ","
  file-type word ticks ","
  file-print elevation
  ]
```

Closing files

- When the model stops, you need to close the file.

```
to go
...
if ticks > 500
[
  file-close-all
  stop
]
```

Multiple file output

You can have multiple files open at once. Switch between them using `file-open`.

```
file-open "turtle_output.csv"
ask turtles [
   file-type word who ","
   file-type word ticks ","
   file-print elevation
   ]

file-open "summary_output.csv"
file-type word ticks ","
file-type word (mean [elevation] of turtles) ","
file-print 100 * fraction-crowded
```

Exporting to files

- If you have plots, you can output the data from the plot to a file using `export-plot`

```
export-plot "Fraction crowded" "frac_crowded.csv"
```

- You can export the current state of your entire model (all turtles, patches, their turtles-own, patches-own variables, etc.) using `export-world`
    - You can import the world to restart your model where you left off using `import-world`
- You can also export plots, world, etc. from the “File/Export” menu.

Model with Observations

Model with Observations

You can download the full model with different observations we have been discussing at

https://ees4760.jgilligan.org/models/class_10/butterfly_class_10_observing.nlogo