From Animations To Science

EES 4760/5760

Agent-Based and Individual-Based Computational Modeling

Jonathan Gilligan

Class #6: Monday, September 8 2025

Download files

Download files for Butterfly model

Download the following files from the Download page on the class web site:

- A single ZIP file with all the files in it:
  https://ees4760.jgilligan.org/models/class_06/class_06.zip
  - Unzip after downloading.
- Or individual files listed on Download page
  https://ees4760.jgilligan.org/downloads/butterfly_science/
- Start NetLogo and load `butterfly_class_06a.nlogo`

Projects

Individual Semester Project

- Semester Project:
  - Pick a model that you want to work with
    - Recommendations: use an open-source repository
      - (CoMSES OpenABM)
      - NetLogo “model library”
    - Or a model from the *Modeling Social Behavior* textbook
    - Or another place
  - Next Monday (Sept. 15) we’ll discuss choosing a project model in class.
  - Friday, Sept. 19:
    - One-page description of model and thoughts for extending it
  - Wed., Oct. 15: Analysis of model ODD and code.
    - Short write-up of how model works and output from running it
  - Wed., Oct. 29: ODD for extending model
  - Wed., Nov. 12: Draft of extended model
  - Mon.–Wed. Dec. 1–3: Presentations on experiments with extended models
  - Fri. Dec. 5: Write-up of research project (around 10 pages)
- Detailed Assignments for Individual & Team projects are on the “Projects” page of the course web site.

Team Project

- Team Project:
  - Each team (2–3 students) will code a model from an ODD in the textbook:
    - The *Business Investor Model* (Ch. 10) studies the *adaptive behavior* design concept
    - The *Telemarketer Model* (Ch. 13) studies the *interaction* design concept.
  - By Friday Sept. 12, complete the survey on Brightspace to indicate your first choice model.
    - **Optionally:** If you and 1–2 classmates want to work together as a team say who is on your team.
  - Use model to do exercises from book
  - Make presentation about what you learned (Wed. Oct. 8).
  - Short written report Fri. Oct. 17.
- Detailed Assignments for Individual & Team projects are on the “Projects” page of the course web site.

Emergence

Emergence

- A tricky concept.
- Joshua Epstein in *Growing Artificial Societies:* “stable macroscopic patterns arising from the local interaction of agents.”
- Epstein ten years later: “I have always been uncomfortable with the vagueness and occasional mysticism surrounding this word.”
- Epstein now prefers to talk about “*Generative* Social Science” instead of “*emergence*”

Example of Emergence: flocks of starlings

- Thousands of individuals
  - unique and different
  - interact locally
  - show adaptive behavior

Behavioral Ecology
doi:10.1093/beheco/arq149

Self-organized aerial displays of thousands of starlings: a model

H. Hildenbrandt,[a] C. Carere,[b,c] and C.K. Hemelrijk[a]
[a]Theoretical biology; Behavioural Ecology and Self-organisation, Centre for Ecological and Evolutionary Studies, University of Groningen, PO Box 14, 9750 AA, Haren, The Netherlands, [b]CNR–INFM, Dipartimento di Fisica, Universita' di Roma La Sapienza, P.le A. Moro 2, 00185 Roma, Italy, and [c]Dipartimento di Ecologia e Sviluppo Economico Sostenibile Università degli Studi della Tuscia, Viterbo, Italy

Through combining theoretical models and empirical data, complexity science has increased our understanding of social behavior of animals, in particular of social insects, primates, and fish. What are missing are studies of collective behavior of huge swarms of birds. Recently detailed empirical data have been collected of the swarming maneuvers of large flocks of thousands of starlings (*Sturnus vulgaris*) at their communal sleeping site (roost). Their flocking maneuvers are of dazzling

H. Hildenbrandt, C. Carere, & C.K. Hemelrijk, *Behavioral Ecology*, **21**, 1349. doi: 10.1093/beheco/arq149

Starling murmuration

By Liberty Smith & Sophie Windsor Clive, Islands and Rivers, https://vimeo.com/31158841

Flock of thousands of starlings

H. Hildenbrandt, C. Carere, & C.K. Hemelrijk, *Behavioral Ecology*, **21**, 1349. doi: 10.1093/beheco/arq149

Simulated flock of thousands of starlings

H. Hildenbrandt, C. Carere, & C.K. Hemelrijk, *Behavioral Ecology*, **21**, 1349. doi: 10.1093/beheco/arq149

Simulated flock of thousands of starlings

H. Hildenbrandt, C. Carere, & C.K. Hemelrijk, *Behavioral Ecology*, **21**, 1349. doi: 10.1093/beheco/arq149

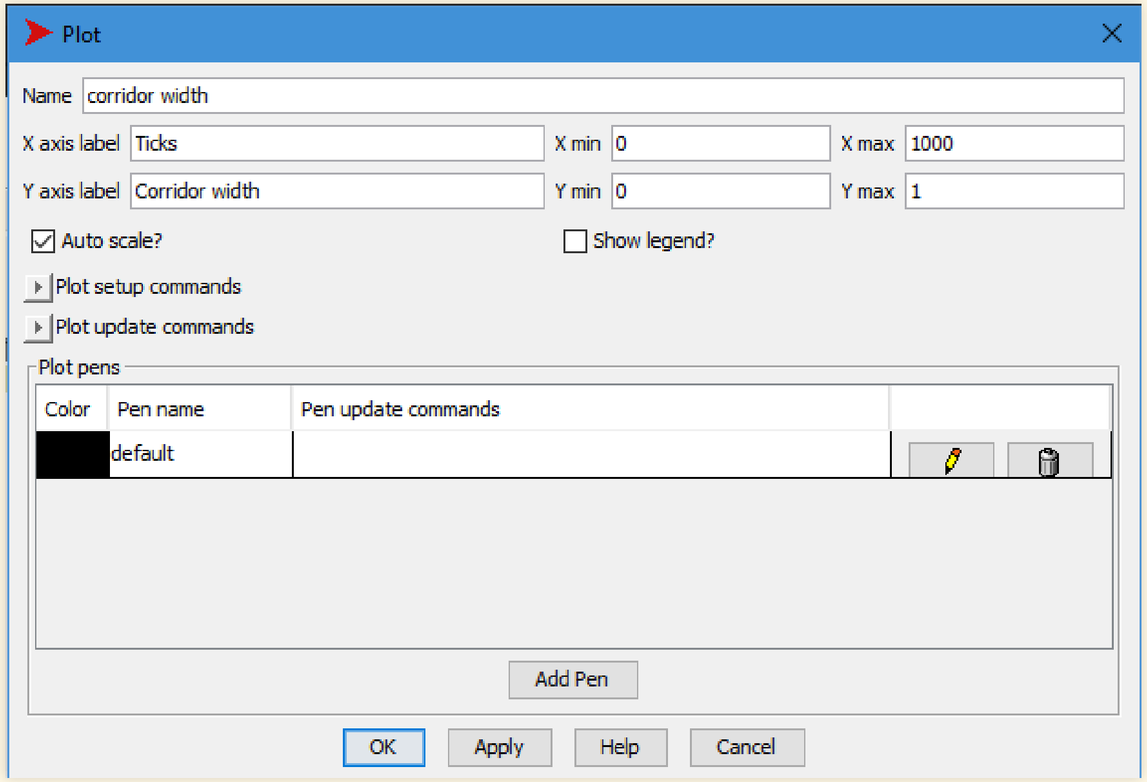
More Experiments with the Butterfly Model

More Experiments with the Butterfly Model

- You should have downloaded:
  - Various versions of NetLogo butterfly models.
  - A digital elevation map of real hills https://ees4760.jgilligan.org/models/class_06/ElevationData.txt
- Start NetLogo and load `butterfly_class_06a.nlogo`

Plot Corridor Width

- On the interface tab, add a plot



- On the code tab, add a line to `go` to plot the corridor width

```
plot corridor-width
```

Enhance Interface

- Add a button to export the plot to a file:

```
export-plot "Corridor-width" (word "corridor-output-for-q-" (precision q 2) ".csv")
```

  - `precision q 2` rounds `q` off to two decimal places.
  - `word` combines several different things into a single text string: "`corridor-output-for-q-0.40.csv`".
    - Parentheses tell NetLogo which things `word` should apply to:

```
(word a b c d ... q)
```

      will combine the values of variables `a`, `b`, `c`, `d`, ..., `q` into a single text string.

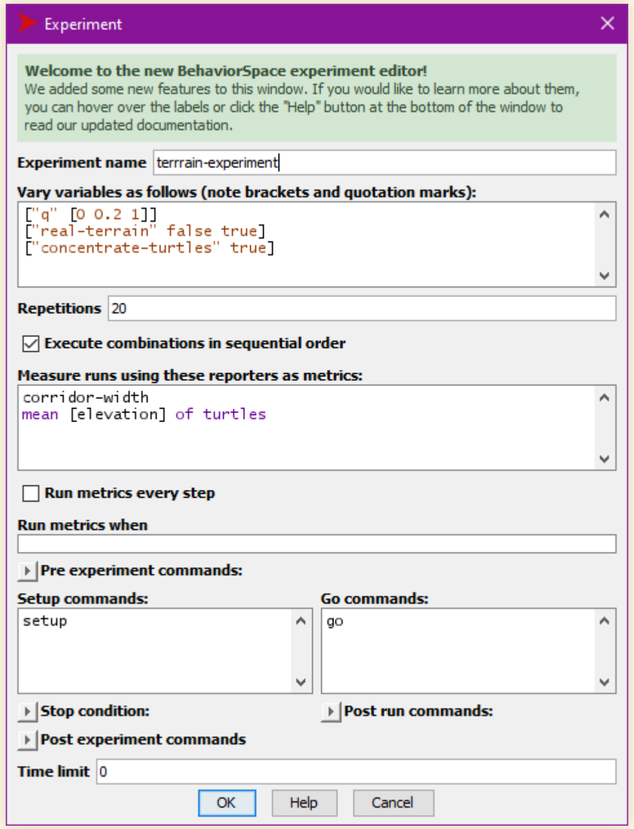
- Add a button to increment *q* by 0.1
- Add a switch `concentrate-turtles` and edit `to setup` to change the code for `crt 500` to include this:

```
ifelse concentrate-turtles
[
  setxy x0 + random 10 - 5 y0 + random 10 - 5
]
[
  setxy random-pxcor random-pycor
]
```

BehaviorSpace

- If your model is having problems, compare it to `butterfly_class_06b.nlogo`
- Open BehaviorSpace and create an experiment
  - Call it `experiment`
  - Vary `real-terrain` between `false` and `true`
  - Vary `q` from 0 to 1 in steps of 0.2
  - set `concentrate-turtles` to `true`
  - Run 20 repetitions for each value of `q`.
  - Measure `corridor-width` and `mean-elevation` at the last tick only
  - Set time limit to 0 to let model run until it stops
- Run BehaviorSpace experiment
  - Save “table” output
  - Speed things up by unchecking “Update view” and “Update plots and monitors”
- Open the analyzeBehaviorspace app at https://ees4760.jgilligan.org/analyze_behaviorspace and use it to compare the relationship between corridor width and `q` for each terrain
  - If it takes too long to run, there are results in the folder “`behaviorspace-results`”

BehaviorSpace



Practice

Practice

- Work together with a partner
- Add a button to erase the tracks of the turtles (Exercise 5.2)
- Using the realistic terrain, play with q and see what values do best at helping butterflies find mates near hilltops.

Testing

Leaving trails

Turtles leave trails with the pen (because you told them `pen-down` when you created them in `to-setup`). The pens are the same color as the turtles, so it's hard to tell the difference between turtles and trails.

It would be nice to color patches they visited yellow so you could erase the pen trails and see the red turtles contrasting with the yellow trails of visited patches.

- Open "`butterfly_class_06b.nlogo`"
- Add code to color patches yellow when a turtle visits them:
  - At the end of `to move`, add

```
set pcolor yellow
```

- Add a button to the interface to erase the pen trails:
  - Give the button the command `clear-drawing` and the display name "erase trails"

Testing Models

- Using monitors
- Testing for consistency
  - Open `butterfly_class_06c.nlogo`
  - In `to go`, after the turtles move add:

```
if (count patches with [visited?]) != (count patches with [pcolor =
yellow])
[ print "# visited patches does not match # yellow patches." ]
```

NetLogo output

- Four commands to output to *Command Center*:
  - `show` Indicates which turtle or patch used it. Ends the line after the command.
  - `print` Ends the line after the command.
  - `type` Does not end the line.
  - `write` Does not end the line. Text strings are quoted.
- Similar commands output to the *Output* area
  - `output-show`
  - `output-print`
  - `output-type`
  - `output-write`
- See the "**Output** section of the **Programming Guide** in the *NetLogo User Manual*.

More testing

- In `to move` replace the `ifelse` block with this:

```
ifelse random-float 1 < q
[ ; move uphill
  let current-elevation elevation
  move-to max-one-of neighbors [elevation]
  if elevation < current-elevation
  [ show "Turtle is moving downhill." ]
]
[ move-to one-of neighbors ] ; move randomly
```

- In `to go` replace

```
ask turtles with [not finished?] [move]
```

  with

```
ask turtles [move]
```

- Set `q` to 1 on the slider
- Turtles should always move uphill. Why do they sometimes move downhill?