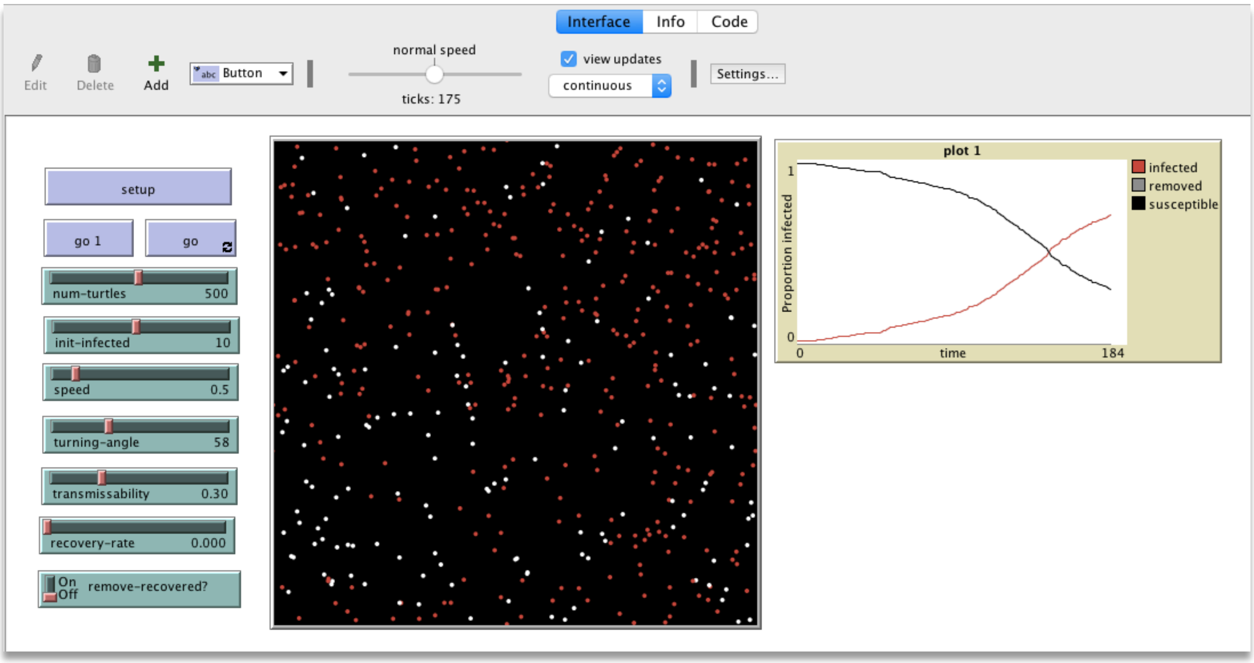
Graduate Workshop on Agent-Based Modeling: Coding Challenge

We will build an epidemiological model in NetLogo in which agents move around in space and infect one another. With the simulation you will be able to vary (1) how the infection starts, (2) how agents move through space and encounter one another, (3) how effectively the infection spreads, (4) how quickly infected agents recover, and (5) whether recovery conveys immunity. You will be able to plot the proportion of agents who are susceptible, infected, and immune. In the end, your Interface tab should look something like this:



num-turtles 500
init-infected 10
speed 0.5
turning-angle 58
transmissability 0.30
recovery-rate 0.000
On Off remove-recovered?

Designing our model

Before you start coding, look through the following model description. This will help you break up the coding of the model into small, discrete tasks.

<u>Agent and their world</u>

The world is a toroidal square grid with dimensions 101 x 101 patches.

Agents are turtles that move through this space. They move constantly, and their movement is not influenced by other agents. Their color indicates their infection state: white = susceptible, red = infected, gray = removed. As their move through space, susceptible agents can become infected if they get too close to infected agents. Infected agents can recover. Depending on the disease, recovered agents may become susceptible again, or they may be "removed," meaning that they cannot become re-infected.

<u>Initialization</u>

First, all variables and agentsets are cleared.

Next, a user-defined number of turtles are created and placed in random locations on the grid. Their statuses as "infected" and "removed" should be set to false, and their color set to white.

Next, the disease is seeded among some number of randomly selected turtles. These turtles should have their color set to red and their status as "infected" set to true.
Finally, we reset the tick counter.

At each tick

- If every turtle is infected (global spread) or if no turtles are infected (the disease is eradicated), stop the simulation.
- For each susceptible agent:
    - If not removed, if there are any infected agents close by, become infected with a probability equal to the transmission rate.
    - [NOTE: avoid having agents able to infect other agents on the same tick on which they become newly infected]
- For each infected agent
    - With a probability equal to the recovery rate, recover.
    - If *remove-recovered?* is true, set recovered to "removed," otherwise set to "susceptible." Update turtles-own boolean variables and agents' colors accordingly.
- For each agent
    - Move:
        - First, turn left and right a random number of degrees as determined by the *turning-angle* slider.
        - Then move forward a distance equal to *speed.*

Plotting

Create a plot with 3 lines as a function of time: the proportion of agents who are (1) susceptible, (2) infected, and (3) removed.

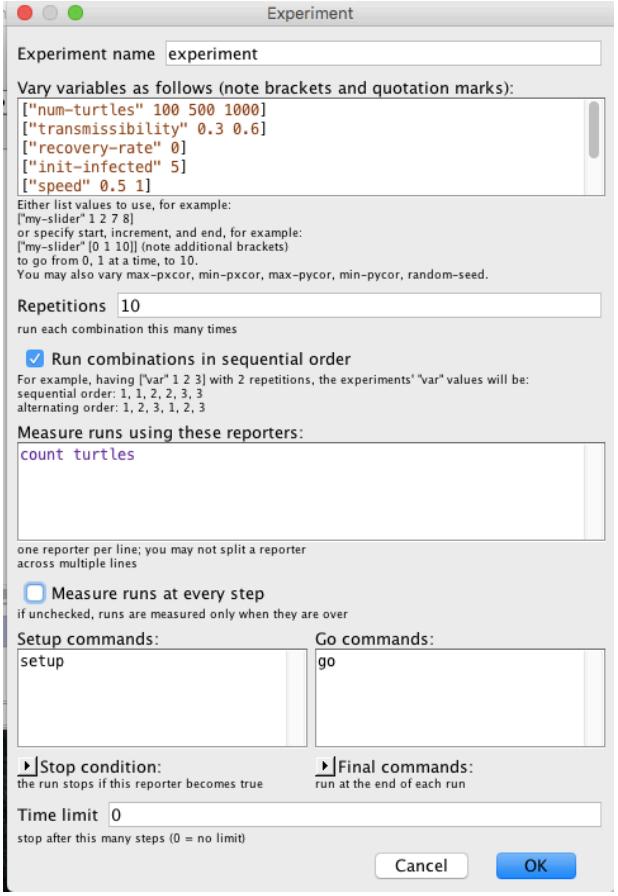
CHALLENGES

1. ANALYZING BATCH RUNS

We want to systematically study this model. This involves running many simulations and sweeping across parameter values. Setup a BehaviorSpace experiment to explore how diffusion rates respond to population density and mobility. For all simulations, set *recovery-rate* to zero (so this is a pure SI diffusion model). Fix *init-infected* at 5. Run 10 simulations for each parameter combination, dealing with the following values:

- *num-turtles* = {100, 500, 1000}
- *speed* = {0.5, 1}
- *turning-angle* = {0, 45, 180}
- *transmissibility* = {0.3, 0.6}

Note how the number of simulations gets very large very quickly, every for this sort of initial exploration! 10 x 3 x 2 x 3 x 2 = 360 simulations. Make sure you only collect data at the end of each run, and that each run stops automatically when all agents are infected. Note that NetLogo will automatically record the number of ticks, which is the variable interest. You will want “Table output.” Analyze the resulting data using analytical tools of your choice. What do you find? In other words, how do the four factors influence the speed of diffusion? Do they interact?



Experiment name: experiment

Vary variables as follows (note brackets and quotation marks):

```
["num-turtles" 100 500 1000]
["transmissibility" 0.3 0.6]
["recovery-rate" 0]
["init-infected" 5]
["speed" 0.5 1]
```

Either list values to use, for example:
["my-slider" 1 2 7 8]
or specify start, increment, and end, for example:
["my-slider" [0 1 10]] (note additional brackets)
to go from 0, 1 at a time, to 10.
You may also vary max-pxcor, min-pxcor, max-pycor, min-pycor, random-seed.

Repetitions: 10
run each combination this many times

☑ Run combinations in sequential order
For example, having ["var" 1 2 3] with 2 repetitions, the experiments' "var" values will be:
sequential order: 1, 1, 2, 2, 3, 3
alternating order: 1, 2, 3, 1, 2, 3

Measure runs using these reporters:

```
count turtles
```

one reporter per line; you may not split a reporter
across multiple lines

☐ Measure runs at every step
if unchecked, runs are measured only when they are over

Setup commands:

```
setup
```

Go commands:

```
go
```

Stop condition:
the run stops if this reporter becomes true

Final commands:
run at the end of each run

Time limit: 0
stop after this many steps (0 = no limit)

2. DETERMINING CONTACT RATE

In mathematical formulations of SIS and SIR models, one can derive a value known as the *basic reproduction number*, R_0. For SIS models, this is the "contact rate" times the ratio of the transmissibility and recovery rates (see any resource on compartmental epidemiology models). While the latter two numbers may be considered intrinsic properties of the disease, the contact rate results from demographics and behavior – it is the average number of interactions per individual, per unit time. In the agent-based model we have built, three parameters influence contact rate: *num-turtles* (the population density), *speed,* and *turning-angle* (the rate at which agents move through the population and contact other agents. Experiment with the simulation to see if you change the equilibrium infection rate without altering the transmissibility or recovery rates. What do you find? Think about how you would might test this systematically using batch runs.

3. COMPLEX CONTAGION

In 2007, sociologists Damon Centola and Michael Macy introduced a model of "complex contagion." This is based on the premise that the adoption of some beliefs, behaviors, or products may require influence from multiple sources. As such, the dynamics of diffusion may be different from class SI models in which exposure to a single "infected" individual is sufficient for spread. Save this model as a new file called *ComplexDisease.nlogo*.

(a) Edit our diffusion model so that an agent can keep track of how many infected agents it has encountered, using a turtles-own parameter called *infection-counter*. Create a slider called *infection-count-threshold.* An agent becomes infected when and only when *infection-counter* ≥ *infection-count-threshold.* How does this appear to change the model dynamics in the SI, SIS, and SIR models?

(b) Individuals may discount experiences they had in the distnat past relative to recent experiences. Create a new slider parameter called *memory-decay-rate* (bounded in [0,1]), that the agent subtracts from its value for *infection-counter.* How does this appear to change the model dynamics in the SI, SIS, and SIR models?

(c) Time permitting, run parameter sweeps in BehaviorSpace to test at least one hypothesis you have about how these new parameters affect the spread of infection.