Introduction to Agent-Based Modeling

Practical Guide to Building NetLogo Models

A Practical Guide to Building Models in Netlogo

Demonstration of Setting Up a Model

- Create Buttons for **setup**, **step** and **go**
- Create Methods for **setup** and **go**
- Create Turtles (crt # [ ]) in **setup**, Move Forward (fd #) in **go**
- Too Many Turtles - Clear Turtles (ca) in **setup**
- Let Turtles Wiggle (rt) and Trace their Trajectory (pd)
- Too Few Turtles - Add population Slider
- Change Shape, Size, Color, Heading, etc. at Creation
- Add Breeds, Populations, and Monitors
- Add Eating Behavior with Color and Size Change
- Add Poisonous Butterfly
- What Else Would Be Cool?

Homework #2

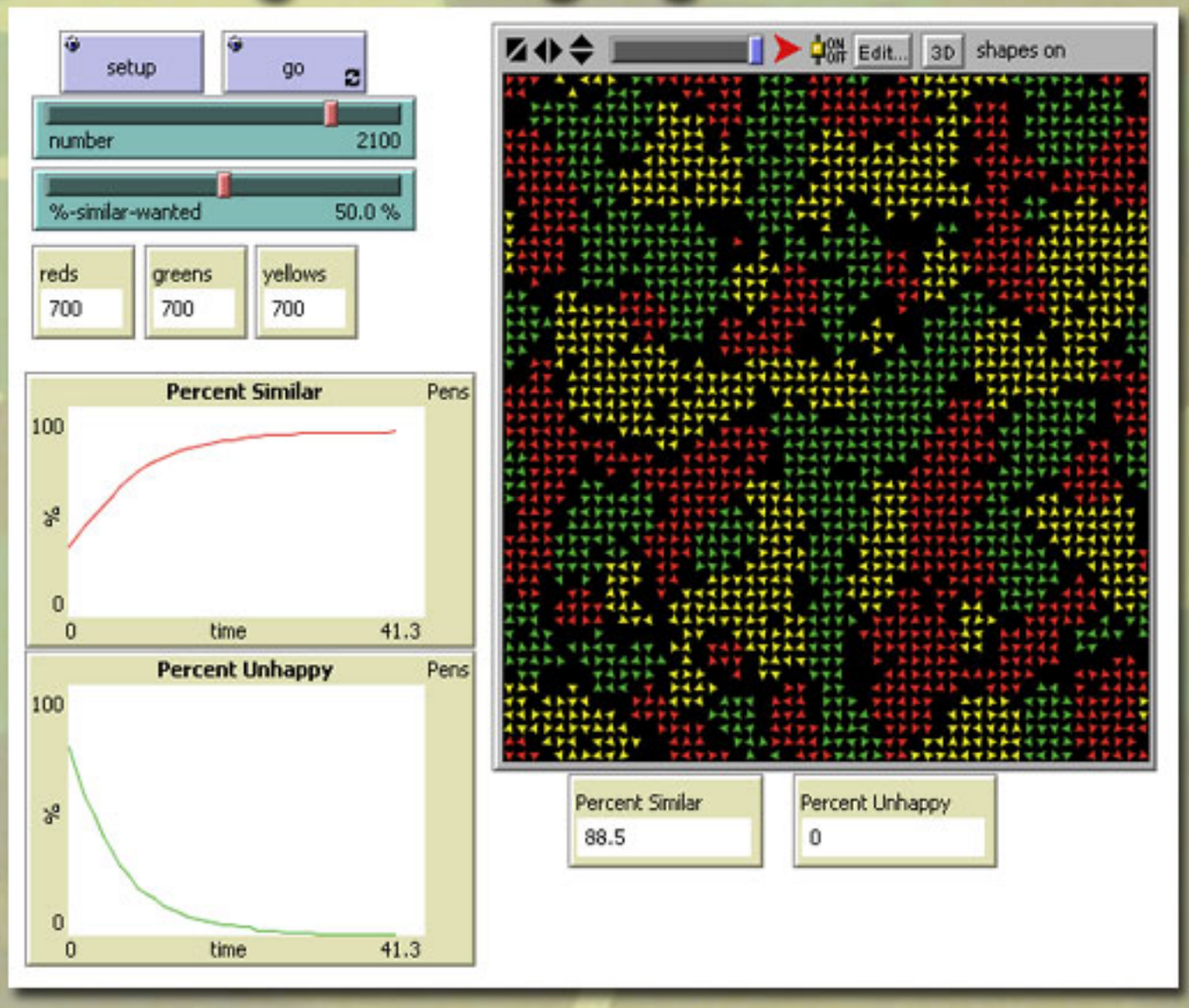
Altering the Segregation Model

- Open the *Segregation* model from the Model Library.
- Add another kind (color) of agent with **equal numbers** in the easiest way imaginable.
- What is (potentially) wrong with that technique?
- To fix it, add a **monitor** to check the number of each color of agent, set number to 2100 and fix the ratios.
- Change the percent-wanted to 50% and find colors that make the display look like camouflage
- The best camo wins a PRIZE

INTRODUCTION TO AGENT-BASED MODELING

Homework #2

Altering the Segregation Model



Beat This!!!

send screenshot to aaronbramson@gmail.com

NetLogo

Homework #2

Further Extensions to the Segregation Model

1) Add more sliders for %-similar-wanted for each color so they can have different tolerance levels.

2) Change the agents' shape so that it's easier to see the distribution of agents.
   Go to the tools menu and select the Shapes Editor and then choose the shape that makes sense to you.

3) Add sliders for the number of each type of agent so you can have an uneven distribution of agent types.
   Run some experiments on different mixes and note any qualitative changes that result.

4) Add a mutation rate so that unhappy agents have a percentage of changing their type rather than moving.
   Note that changing thier type may not make them happy depending on the percentage required.
   Does this speed up or slow down the time to universal happiness?

5) Change the mutation rule into a learning rule so that agents copy the type of turtle that has the highest percentage of happiness among its neighbors.
   That is, find which type of agent among the neighboring agents has the highest percentage of happy agents and copy that type.

6) Instead of just the Moore neighborhood, add a radius slider and have agents calculate happiness based on all the agents within that radius.
   Does this make it easier are harder to be happy?
   What effects does this have on the distribution of agents?

for more details see the EITM presentation: tinyurl.com/kwqnwx