CPSC 565: Emergent Computing, Winter 2020

Assignment 1: Wasp Nest Construction (NetLogo)

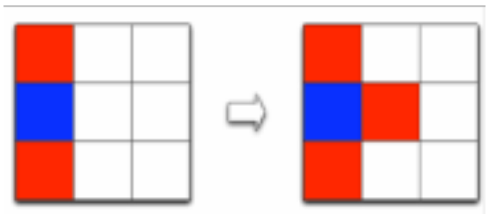
Due Date: February 7, 2020

Wasps build complex, yet well structured, nests for their colonies. Each wasp species builds a structurally distinct nest using similar building materials. With no master builder, the task of nest building is distributed over all the involved wasps. How does each wasp know what to do in order to succeed in building a structurally correct and functional nest? One of the theories to explain this emergent phenomenon uses the idea of qualitative stigmergy where the wasps use visual queues (stigma) in their environment to decide on their actions. In this assignment we implement a model [1] of wasp nest construction based on qualitative stigmergy.

This assignment must be implemented using NetLogo, and completed individually. We simplify the standard model to use autonomous non-communicating agents (representing wasps) moving around in 2D simulation space and interacting with a 2D rectangular grid. The grid represents possible building sites that the wasps can use in order to put down their building materials.

Each wasp agent should randomly move around the simulation environment. When a wasp encounters an empty building site (represented in NetLogo as a patch), it should make a decision whether or not to build in that site and what type of material to put down. This decision is made qualitatively by observing the wasps' immediate environment (in a 2D rectangular grid this equates to the 8 neighbouring sites of the site the wasp is considering). Your simulation must support at least 2 different building materials (i.e. each patch should have at least 3 states: 0=empty, 1=material A, 2=material B, etc.). Each state should be represented by a different patch color in the simulation to distinguish it from others. Initially, the wasps and building materials are randomly distributed in the grid. Wasps should not pick up existing material placements and have an unlimited supply of materials.

The wasps use a set of rules to decide their actions. Each rule in the rule-set is composed of the neighbourhood state (list of the states of the 8 neighbourhood patches) and the material to be placed on the currently occupied site. For example, the image below shows the rule: “(0 0 0 0 0 1 2 1) - > 1” - which means that if the wasp matches the specific rule pattern to its neighbourhood then it will place a building material of the given type into its current site. Here material 1 is red and material 2 is blue. The list of neighbours begins with index 0 referring to the patch directly north of the current site and continues clockwise.



One rule is not enough to produce any useful result - hence, a rule set consisting of multiple rules should be used. Only the first successful match in a rule set should be executed.

Implementation Requirements:

This assignment must be implemented using NetLogo 6 (http://ccl.northwestern.edu/netlogo). NetLogo contains a large assortment of sample programs (via the Model Library) and excellent online documentation (at http://ccl.northwestern.edu/netlogo/docs). Working with lists and reporters is an important part of this assignment.

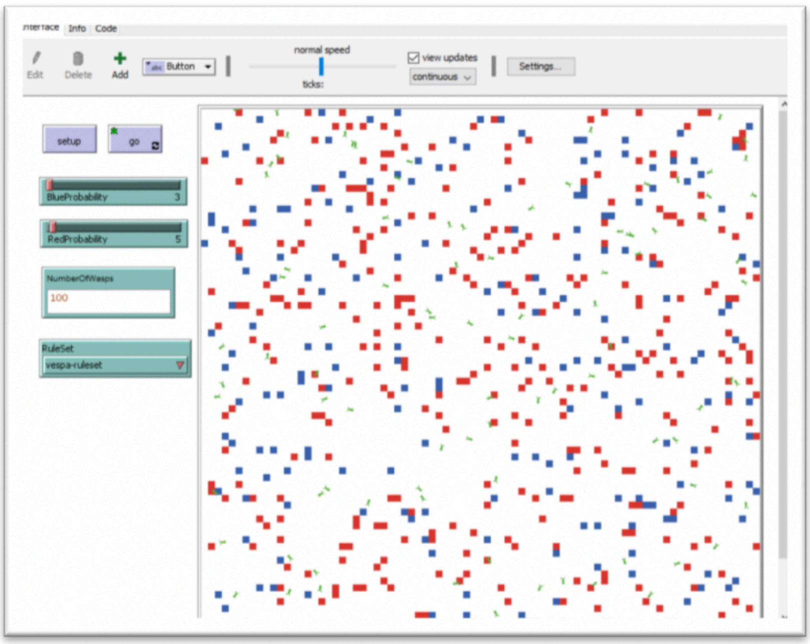
Several models contain useful information you will need to complete your assignment. The Biology/Termites model gives a nice introduction to agent control and agent-patch cooperation. The Computer Science/Cellular Automata models provide examples of patch queries.

You should come up with your own rule sets to produce interesting structures. Some example rule sets (inspired by nature) will be provided via the D2L website and will be used in order to evaluate your simulation.

You are required to implement the 3 rule-sets provided on D2L and come up with at least 2 other interesting rule-sets. All rule-sets must be implemented symmetrically, i.e. each rule is actually four rules, one for each direction. To create the three other rules you may use the code provided with the required rule-sets or make your own. Make sure whatever code you use is working ….

User Interface:

Your UI should consist of UI components such that your project can be easily run from the main menu. It should at least include a setup button, a go button, the initial density of the building materials, and the number of wasps. In addition, while the rule sets can be hard coded in the code, you should provide a drop down rule selection button to easily switch rule sets.



User interface example (available on D2L as a separate file)

Documentation:

All project documentation should be provided via the Information tab of your NetLogo project. Please provide documentation of your modeling choices and information on how to run your simulation. You should describe all the rule sets that you created and what types of structures they generated.

Submission:

Submit your NetLogo program (with .nlogo extension) using the D2L dropbox for assignment 1.

Grading:

Your grade will be a letter grade based on the following rubric.

A+	All requirements for an A plus either: More than two materials supported and used User defined rule-sets are unusually interesting Code is extremely NetLogo-y and efficient
A	All rule-sets are implemented symmetrically Two user defined rule-sets are implemented UI is fully implemented as required Wasps behave properly Good documentation and analysis
A-	One or two minor errors in implementation, or Documentation or analysis not quite adequate
B+	One or two minor errors in implementation, and documentation or analysis not quite adequate, or A significant implementation error but no other problems
B	A significant implementation error as well as one or two minor issues and docs or analysis lacking
B-	A significant implementation error, one or two minor issues and docs or analysis lacking
C+	A little better than a C
C	Multiple errors of note, documentation or analysis completely inadequate or missing, but an effort was made

C-	A little worse than a C
D	Little or no effort made, solution produces some results
F	Submission missing or non-functional

[1] Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). Swarm Intelligence: From Natural to Artificial Systems. Chapter 6. Oxford University Press, New York.