PLTW COMPUTER SCIENCE

Activity 4.1.4

Varying Parameters

Introduction

(c)

One of the most important qualities of a good simulation is its ability to reproduce data obtained by other means. For example, in a climate change simulation we'd want to make sure that it is able to reproduce values for average world temperature and CO_2 content of the atmosphere at historical benchmarks. Without being able to reproduce results, little can be said about any predictions based on the model for the future. **Verifying** a model is an important process.

Exploring a model by modifying its parameters as we did in the previous activity has its benefits. Adjusting the parameters by hand and rerunning the model is one way to explore a model. Doing this by hand makes the exploration rather slow. Like anything, if it was worth doing repetitively, you might be surprised at the power of automating the task. Actions scale up with little effort. What do you learn if you explore a million combinations of the simulation parameters?

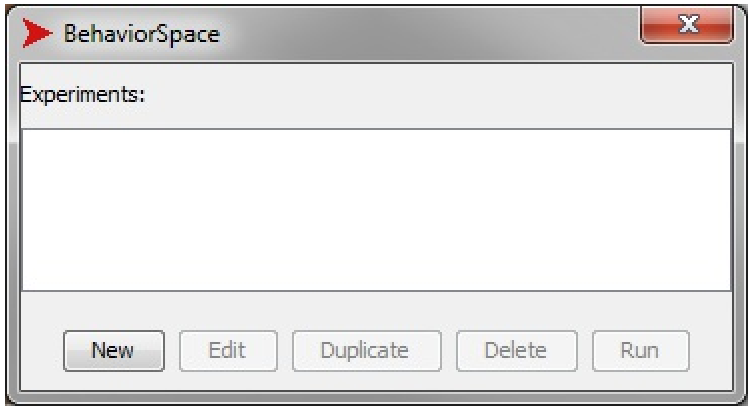
Materials

- Paper and a writing utensil
- Computers with NetLogo installed

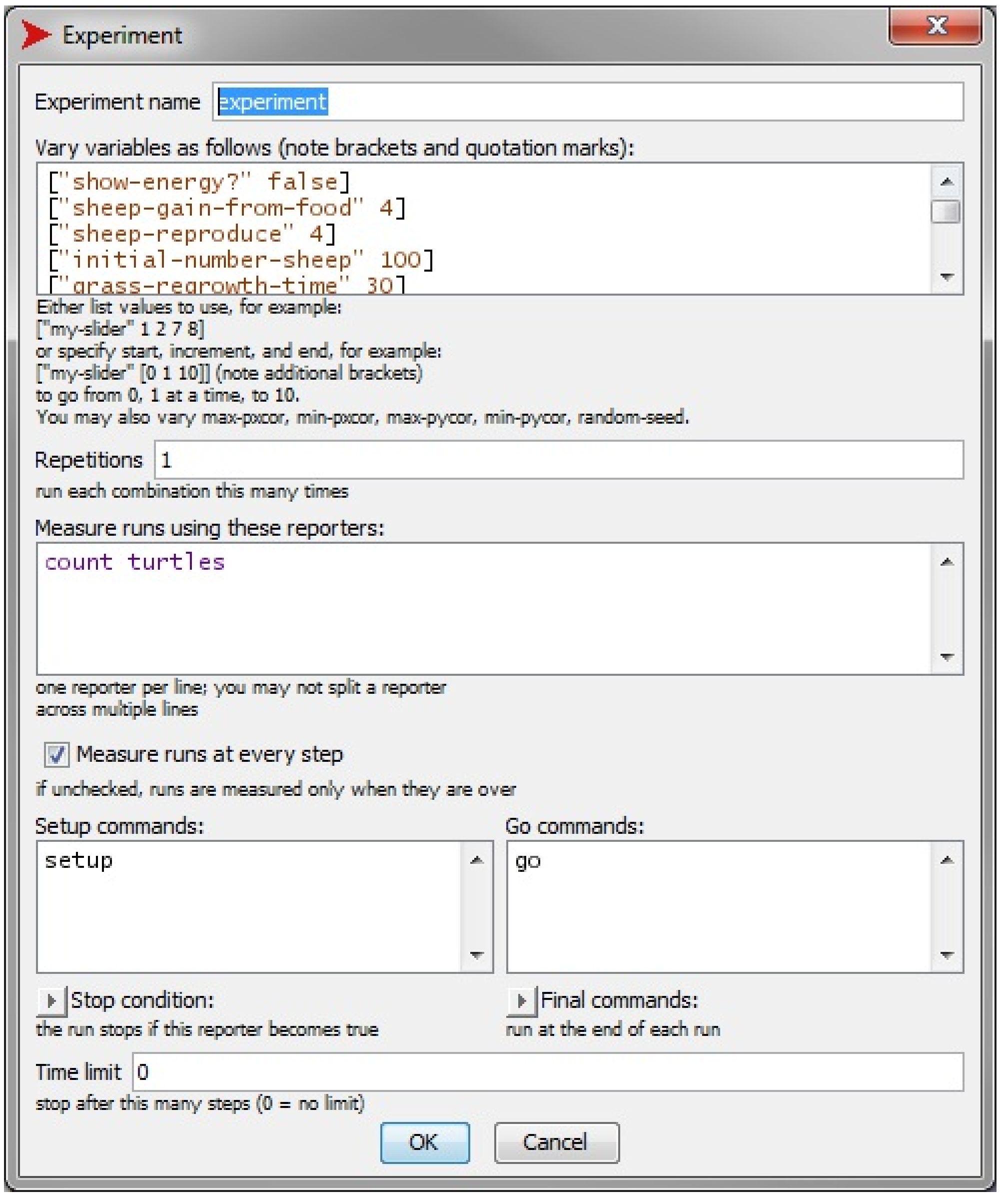
Procedure

Part I: BehaviorSpace

1. Form pairs as directed by your teacher. Meet or greet each other to practice professional skills.
2. BehaviorSpace is a tool built into NetLogo that allows you to run a simulation many times in an automated fashion and gather data from each run. You will use it to examine the reliability of the conclusions that you made about the Wolf and Sheep Predation model in the previous activity. Open the Wolf Sheep Predation model and set the value of `grass?` to "On".
3. Review your notes from the last activity and discuss with your partner or other groups as needed: what predictions did you make about this model with its default parameter values?
4. From the menu bar, select **Tools > BehaviorSpace.** You will see the following window.



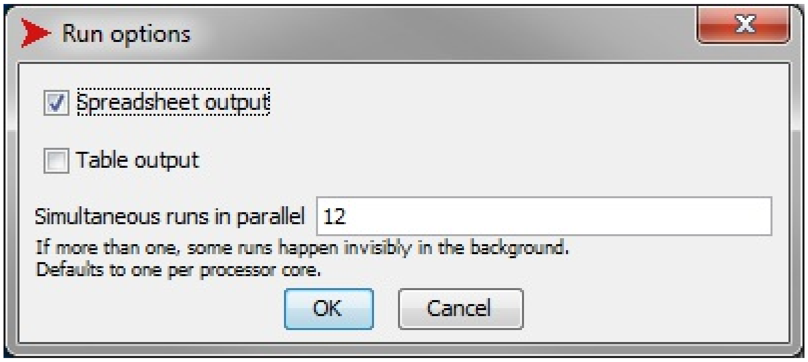
5. Click **New**. You will see a window like the one shown below.



6. This window will allow you to run a Monte Carlo style experiment in which you can obtain data about anything in your simulation. The default values of the simulation are already shown in NetLogo code in the “Vary variables as follows:” field. Modify the rest of the fields as follows:
   - In the “Vary variables...” field, make sure `grass?` is set to “true”.
   - Set “repetitions” to `50`.

    - Set “Measure runs using these reporters:” to `count wolves`.
    - Un-check the “Measure runs at every step” check-box.
    - Set “Time limit” to `5000`.

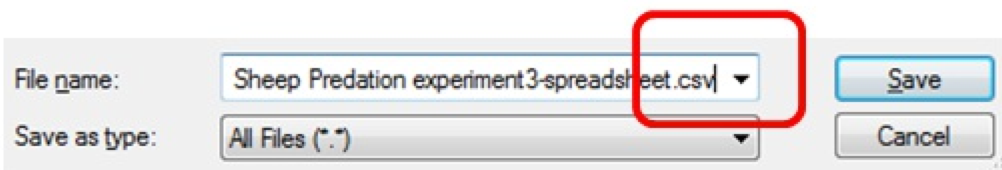
7. Click **Ok**. Then click **Run** in the BehaviorSpace window. You will see a window like the one below:



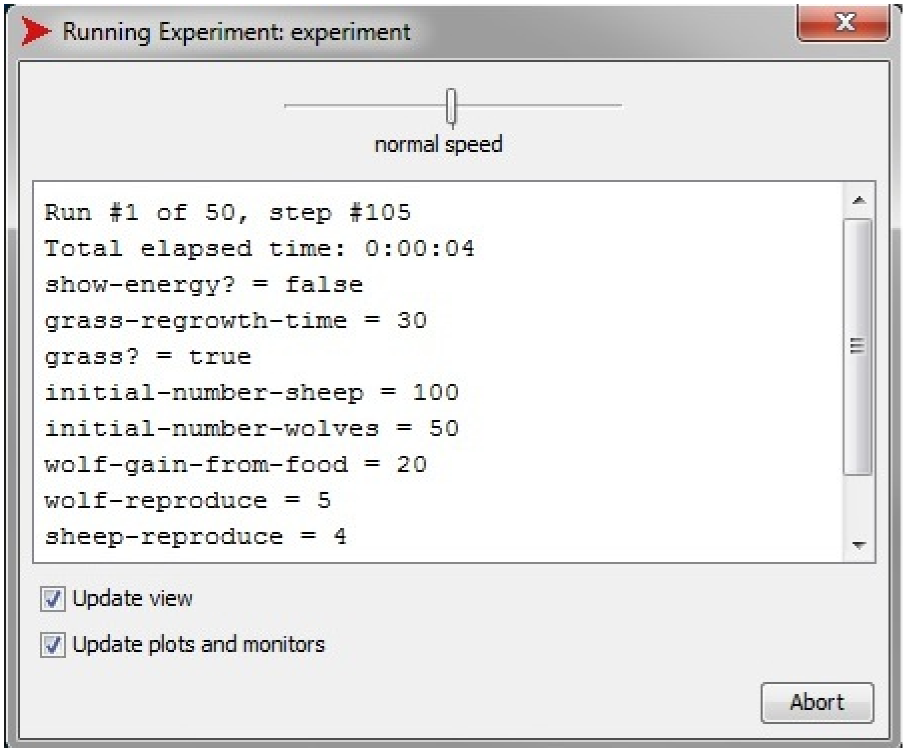
8. Make sure that the “Spreadsheet output” box is checked. This window will also tell you how many runs of the simulation your computer will be able to perform at the same time. In the example above, the computer will perform 12 simultaneous runs. This number is directly related to the number of cores on your processor. Typical home PCs in 2014 have 4 core processors.

   BehaviorSpace is an example of parallelization of a problem. Why do you think it is easy for your computer to accomplish this task in parallel?

9. Click the **Ok** button and choose a place to save the results of your experiment. Confirm that your file has a `.csv` extension so that you can open it in Excel later. Click **Save**.



10. You should see your simulation begin to run in the viewport. You will also see a window like the one shown below. Slide the slider in the experiment window all the way to the “faster” side, and un-tick the “Update view” and “Update plots and monitors” check boxes.



Why might these steps speed up the rate at which your simulations are able to run? Discuss with your partner and then share with the class as directed by your instructor.

11. When your experiment is done running, open the CSV file that it created. You should see something that looks like this:

	A	B	C	D	E	F	
1	BehaviorSpace results (NetLogo 5.0.5)						
2	Wolf Sheep Predation.nlogo						
3	experiment3						
4	04/22/2014 10:35:04:279 -0400						
5	min-pxcoı	max-pxco	min-pycoı	max-pycor			
6	-25	25	-25	25			
7	[run numb	1	2	3	4	5	
8	show-ene	FALSE	FALSE	FALSE	FALSE	FALSE	F
9	sheep-gai	4	4	4	4	4	
10	sheep-rep	4	4	4	4	4	
11	initial-nur	100	100	100	100	100	
12	grass-regr	30	30	30	30	30	
13	grass?	TRUE	TRUE	TRUE	TRUE	TRUE	T
14	wolf-repr	5	5	5	5	5	
15	wolf-gain-	20	20	20	20	20	
16	initial-nur	50	50	50	50	50	
17	[steps]	5000	5000	5000	5000	5000	
18							
19	[initial & f	count wol	count wol	count wol	count wol	count wol	cou
20		73	58	97	88	72	
21							

The results in this file show the values of the parameters in every run. The parameters were the same in every case here, though you can vary them in later experiments. The bottom row of the results shows the number (count) of wolves at the end of each run.

- – In cell A20, type `=AVERAGE(B20:CW20)` to cause that cell to calculate the average number of wolves left at the ends of the simulation runs.
- – In cell A21, type `=MIN(B20:CW20)`.
- – In cell A22, type `=MAX(B20:CW20)`.
- – In cell A23, type `=STDEV(B20:CW20)`.

Answer the following questions and discuss with the class as directed by your instructor.

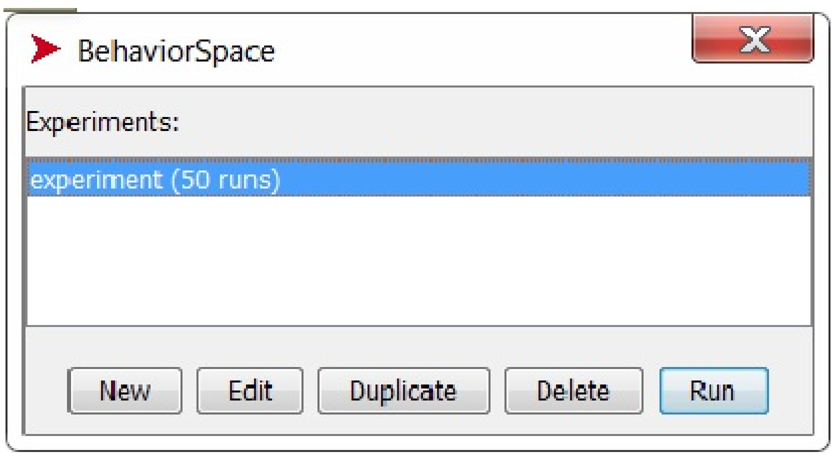
- What was the average number of wolves left at the end of your experiment?
- What were the minimum and maximum values of wolves left at the end of your experiment?
- What was the standard deviation from the mean of the number of wolves left at the end of simulations in your experiment?
- Did the wolves die out in any of your runs? How do you know?
- How many ticks did each of your runs execute before ending?
- Does this mean that with these parameters the wolves will never die out?
- What do your results suggest about the number of sheep left at the end of each of your simulation runs? Talk with your partner about how you might modify BehaviorSpace to collect data on your theories about the sheep population.

Part II: Parameterization and Calibration

12. Parameterization is the process by which you determine what the parameters of your simulation should be in order to best align with the situations that you are trying to model. If you were going to create your own simulation in a field of your choice, what parameters would you choose? You want your parameters to be granular enough to provide useful results, but not so granular that the simulation is difficult to create or that it includes unnecessary information.

    Calibration is the process of using data to configure parameter values for your simulation that will provide historically accurate results. Once a simulation has been properly calibrated, its results can better be used to predict future outcomes. Imagine that the wolf sheep predation model was actually meant to model the coexistence of these two species somewhere in the real world. If you know that neither of your two species are extinct yet and that both have coexisted for a long time, then you would want to first calibrate your model so that these species achieve an equilibrium. With the value of `grass?` set to “Off”, the model does not achieve equilibrium. With the value of `grass?` set to “On”, the model achieves equilibrium, so no further calibration is needed for this model for our purposes.

13. Suppose you want to determine the effects of wolves reproducing more quickly on your model. You can use BehaviorSpace to find out at what point wolves will reproduce quickly enough to upset the equilibrium. Edit your BehaviorSpace experiment as follows.
    - From the menu bar select **Tools > BehaviorSpace**. You will see the following window.



    - Select `experiment(50 runs)` and click edit.
    - Change the line in the “Vary variables by” filed that reads:

      `["wolf-reproduce" 5]` to read
      `["wolf-reproduce" [1 1 15]]`
      (note the second set of brackets)

    - Change repetitions to 5.
    - Click Ok, Run, Ok, and then when prompted, save your data file with a new filename.
    - Speed up the simulation rate by turning off the processor-dependent graphics effects: uncheck the options "Update view" and "Update plots and monitors" in the "Running Experiment" window.

    In the first version of this statement, the number `5` means to run the simulation using 5 as the value for `"wolf-reproduce"`. In the second version the `[1 2 25]` means vary the variable `"wolf-reproduce"` from 1 to 25 in steps of size 2. Run the experiment. Using the spreadsheet, at what value of `wolf-reproduce` do you start to see the wolf population (`wolf count`) dying off?

14. To take a step toward refining your result from the previous step, modify the Behavior Space experiment again.
    - Vary `"wolf-reproduce"` in a range that includes values ranging from a minimum of 5 less than your value from Step 12 up to a new maximum that you think will result in 100% wolf extinction. Use a step value of 1.

      `["wolf-reproduce" [` min `1` max`]]`

    - Increase the Repetitions to 10.

    What is the first value of `"wolf-reproduce"` at which you see the wolves die out this time?

15. Refine Step 14 as necessary to determine a real value for the point at which the wolf reproduction rate becomes too quick to maintain equilibrium. Continue until your teacher instructs you to stop, and then discuss your methods with your class as directed by your teacher.

Part III: Challenges

16. Set the following initial conditions for `wolf-reproduce` and `wolf-gain-from-food` the wolf sheep predation model:

    This default configuration of the simulation with only these two values modified represents a **transient** state, one in which the populations will appear to be at equilibrium for a while, but eventually the balance will tip and at least one species will die off.

    Your goal is calibrate the simulation so that it reaches true equilibrium. The challenge is to make the adjustments you think will have the most impact *without changing the given two values*. Describe your method for reaching and proving equilibrium along with the necessary values for each parameter.

17. Open the Moths simulation by selecting **File > Models Library** and then **Sample Models > Biology > Moths.**
    - View the following video on youtube **https://www.youtube.com/watch?v=GhR9E8ztatg**.
    - Attempt to calibrate your simulation so that the behavior of your moths in your simulation matches the behavior of real moths as closely as possible.
    - Record and describe your method for reaching and proving that your simulation matches

the biological behavior observed along with the necessary values for each parameter.

Conclusion

1. Describe why you think proper parameterization is important in all simulations.
2. Describe why you think proper calibration is important in all simulations.