Exercise Hints and Solutions for Chapter 20

Agent-based and Individual-Based Modeling: *A Practical Introduction, 2nd Edition*

Exercise 1

From Figure 20.3, it appears that the parameter ranges where all three criteria are most likely to be met are around `survival-prob` = 0.98, while the best range of `scout-prob` is not clear. So it makes sense to do another BehaviorSpace experiment with `survival-prob` close to 0.98 (e.g., 0.97 to 0.99, in steps of 0.001), while keeping a broad range in `scout-prob`. An example Excel analysis of such an experiment is in the instructor material file `Ex20-1_WoodHoopoes_Calibration.xlsx`. There is a region around `survival-prob` = 0.978 and `scout-prob` = 0.05 to 0.1 where all three criteria are met at least sometimes.

Exercise 2

Ideally, students working on this rather open-ended exercise will review Section 20.4 for guidance on modifying the calibration process, identify 1-2 changes to the process, implement them, and determine whether they succeed in making the model reproduce the three calibration criteria more robustly. Here are four ideas that students are likely to try.

Approach 1: Using replicates

The Woodhoopoe model is fairly stochastic, and students are likely to have noticed that certain parameter combinations can meet the calibration criteria in some model runs and not others. One way to use replication is to evaluate, for each parameter combination, the percentage of replicates in which the calibration criteria were met.

Approach 2: Add a calibration parameter

So far we have been calibrating with just two of the model's parameters, but it seems likely that survival of scouting forays also fits the description of good calibration parameters in Section 20.4.1: we are unlikely to have an accurate estimate of it, and it might have strong effects. We can move this parameter to a slider on the interface, add it to the BehaviorSpace experiment, and see if it helps meet calibration criteria. A NetLogo file implementing this approach is provided with the instructor materials: `Ex20-2_WoodHoopoes_ScoutSurvivalCalib_2ndEd.nlogo`. The analysis (provided in `Ex20-2_WoodHoopoes_ScoutSurvivalCalib.xlsb`) indicates, however, that scouting survival also has little effect on calibration compared to `survival-prob`.

Approach 3: Broadening the calibration criteria

Section 20.4 implies that it is not always best to calibrate using more parameters or to modify a model just to achieve calibration to data that are themselves also uncertain. Therefore, one approach is to make the calibration criteria a little broader. Students should find that making the ranges of mean abundance, variation in abundance, and alpha vacancy broader by even 10% or so makes it clear that the model is best calibrated when survival-prob is around 0.978 and `scout-prob` is relatively low (<0.3).

Copyright 2011-2018 Steven F. Railsback and Volker Grimm

Approach 4: Modifying the model

Section 20.4.6 suggests reconsidering a model's theory for agent behavior when calibration is challenging. In this case, students who have conducted Exercise 2 of Chapter 19 should have developed an alternative theory for how woodhoopoe decide whether to scout for new territories. They could repeat the calibration exercise using a version of the model that uses alternative theory.

Exercise 3

Students should produce a table very similar to Table 20.1:

Measure of model fit	Parameter set		
	1	2	3
Difference in mean	13*	49	41
Difference in standard deviation	58	3*	75
Maximum error	316	250*	404
Mean squared error	24400*	28800	30800
Number of years with results within 100 units of observed	6*	4	5

*Best value.

Parameter set 1 is best for three of the five measures, including mean squared error (probably the most meaningful measure); set 2 is best for two measures. Parameter set 1 is probably best.

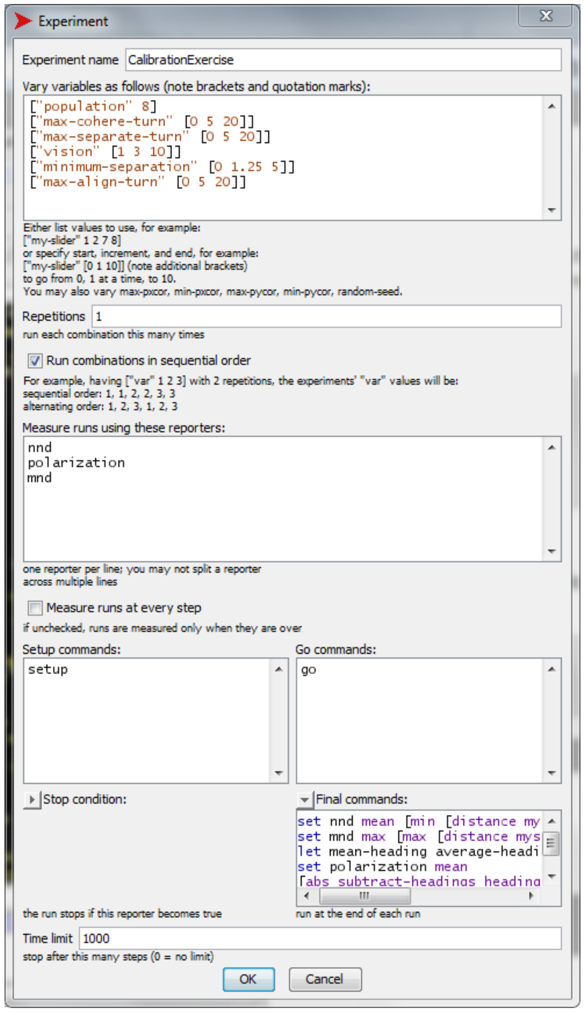
Students should be aware that there are only three significant figures in the observed data (and few types of field data have even that much precision), so they should not compare model results at more than three significant figures of precision.

(The "Observed" resource abundance in Table 20.3 actually came from a simulation with $r = 1.8$, $K = 3150$, and $N_0 = 2500$; with +/-250 units of random noise. These "true" parameter values happen to also produce a mean square error of 24400.)

Copyright 2011-2018 Steven F. Railsback and Volker Grimm

Exercise 4

After modifying the Flocking model as described in the exercise, students can conduct a BehaviorSpace experiment like this:



The output variables are three new global variables that must be updated before the model stops. They could be updated within the code, but it is easy to add them to the BehaviorSpace experiment as "Final commands".

```
set nnd mean [min [distance myself] of other turtles] of turtles
set mnd max [max [distance myself] of other turtles] of turtles
let mean-heading average-heading
set polarization mean [abs subtract-headings heading mean-heading] of turtles
```

This code uses a procedure average-heading to calculate the average heading of all the turtles. However, the Flocking model no longer has this procedure so it must be created by modifying the `average-flockmate-heading` procedure. Simply copy this procedure and modify it to this:

```
to-report average-heading  ;; Now an Observer procedure to report
  ;; mean heading of all turtles
  let x-component sum [dx] of turtles
  let y-component sum [dy] of turtles
```


Copyright 2011-2018 Steven F. Railsback and Volker Grimm

```
  ifelse x-component = 0 and y-component = 0 ; Some defensive programming
    [ report 0 ]
    [ report atan x-component y-component ]
end
```

An example version of the modified model is provided as `Ch20-Ex4_Flocking_2ndEd.nlogo`, and an example analysis of the calibration experiment is at `Ch20-Ex4_Flocking_Analysis.xlsx`. We found all the calibration criteria to be met under most combinations of parameters in the ranges:

- vision: 7-10
- minimum-separation: 1.25-2.5
- max-align-turn: 10-20
- max-cohere-turn: 5-15
- max-separate-turn: 5-20.

Exercise 5

This exercise implies that the model should be run with just one hunter (whereas the code in Chapter 2 includes two hunters); it is OK to use two, though the results will be different. Students need to create sliders for the three parameters and change the code to use the global variables from the sliders and to stop when 50 mushrooms have been found. Then a big BehaviorSpace experiment can report the number of ticks until the 50^{th} mushroom is found. We ran an experiment covering wide ranges of all three parameters:

```
["switch-time" [0 5 100]]
["broad-search-angle" [1 10 181]]
["local-search-angle" [1 15 361]]
```

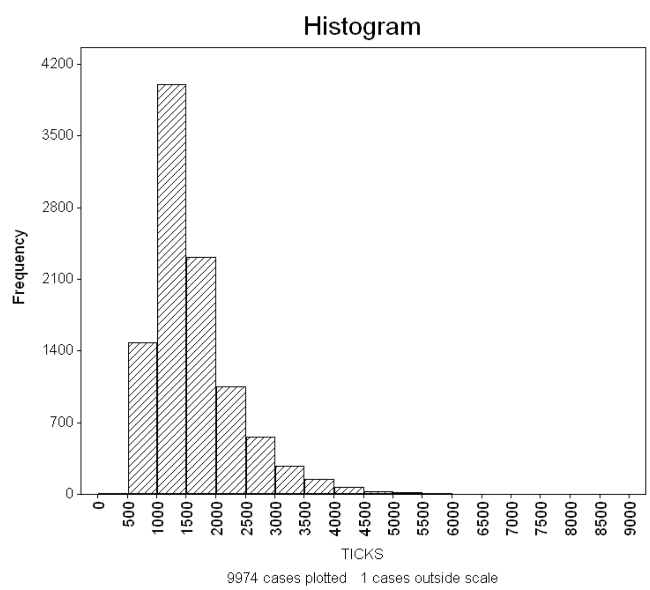
(We avoided search angles of zero because it's remotely possible for them to cause a hunter to search forever without finding another mushroom.)

The problem is interpreting the output systematically to find good parameter combinations. The model turns out to be highly stochastic, with random variation in results that is high compared to the effects of parameter values. One approach is to sort the results in descending order of hunting time, and look for patterns in the parameter values that produced the shortest times (illustrated in the following table). Just looking at the sorted results turns out not to be very useful, as each parameter has values over a wide range even within the model runs producing lowest hunting times:

Copyright 2011-2018 Steven F. Railsback and Volker Grimm

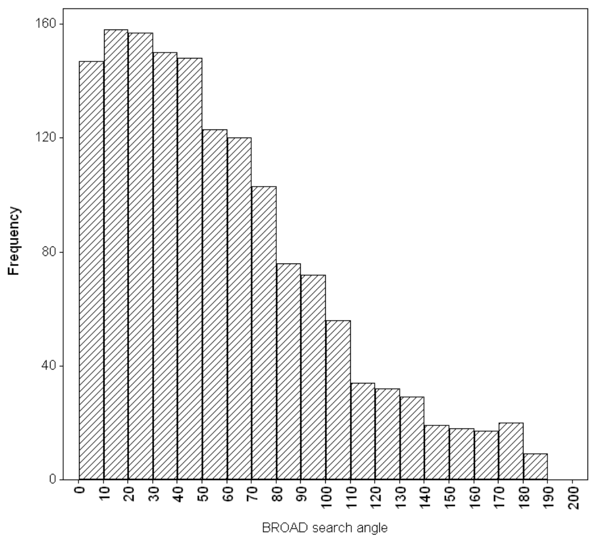
switch-time	broad-search-angle	local-search-angle	ticks to find 50 mushrooms
40	21	136	320
20	41	106	382
80	91	106	391
45	91	121	425
65	41	121	426
90	21	346	445
45	71	166	450
15	1	91	470
15	71	151	471
45	131	106	473
45	41	181	482

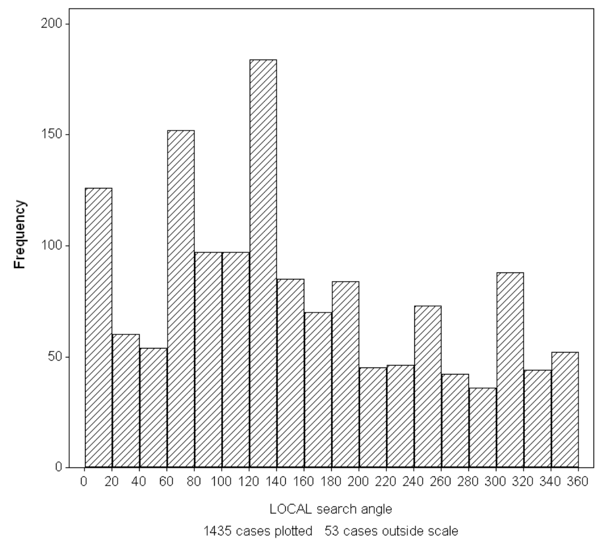
However, it is useful look at the model runs producing the best hunting times statistically. First we can examine the range of model results by histogramming the hunting times over all model runs in the BehaviorSpace experiment. The histogram (below) shows that hunters sometimes took over 9000 ticks to find 50 mushrooms, but most often needed fewer than 1500 ticks.

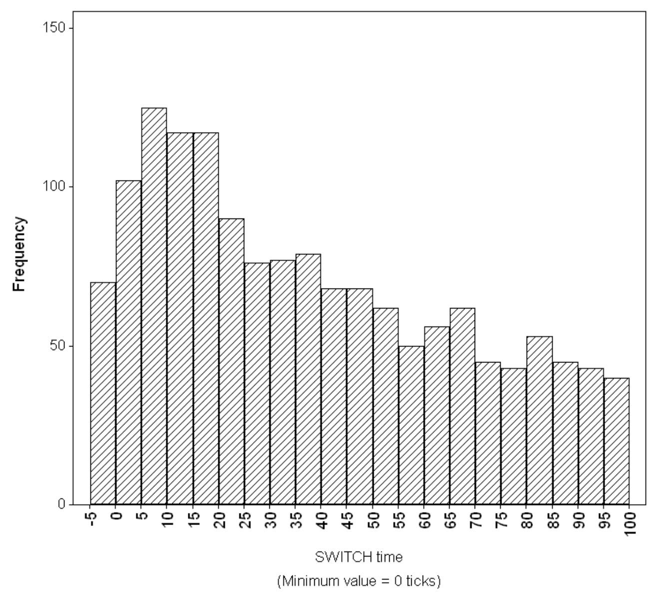


We can define good model runs as those finishing in fewer than 1000 ticks, and look at the parameter values producing those runs. Again, histograms are useful. The following three histograms show the distribution of parameter values in the runs finishing in less than 1000 ticks.

Copyright 2011-2018 Steven F. Railsback and Volker Grimm





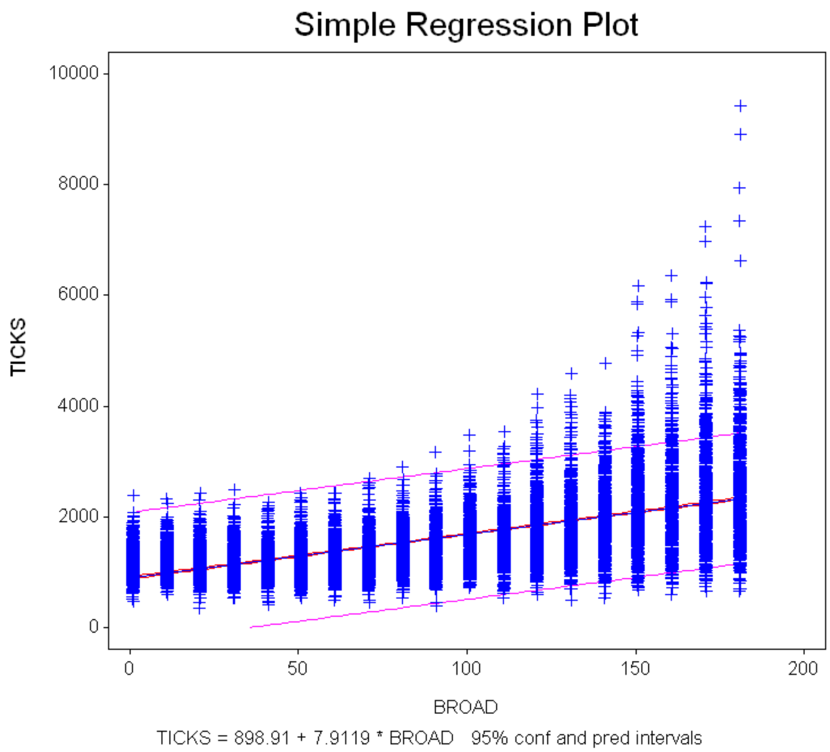


These show that, in model runs producing best results, the broad search angle was most often less than about 80° (a rather wide range), the local search angle varied widely, and the time at which the hunter switches back to broad search also occurred over a broad range but values between 1 and 25 ticks were relatively more common. (Statistically minded students could also look for cross-correlations within the parameter combinations producing good results; we found high switching times to be associated with high broad-search angles and small local-search angles.)

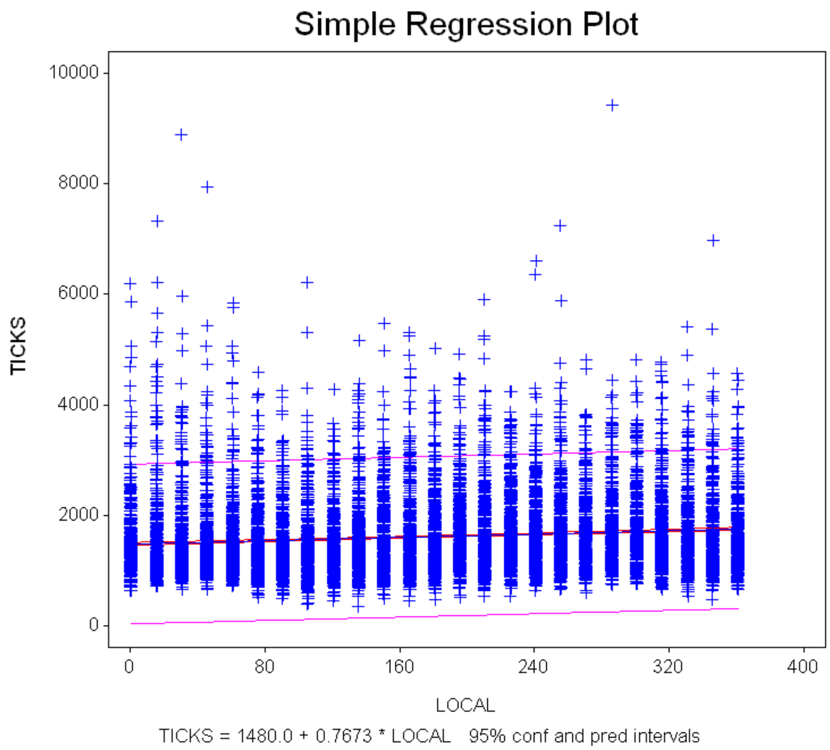
A second approach is to look for statistical relations between parameter values and model results, over all model runs. Linear regression plots are a good place to start, although the above plots indicate that effects of the parameters are unlikely linear at low values. We produced the following regression relations, which indicate that the broad search angle is the only parameter with a strong effect. (Statistics such as P values may indicate that all parameters are “significant” but that is largely an artifact of using so many “observations”—model runs—in the analysis.)

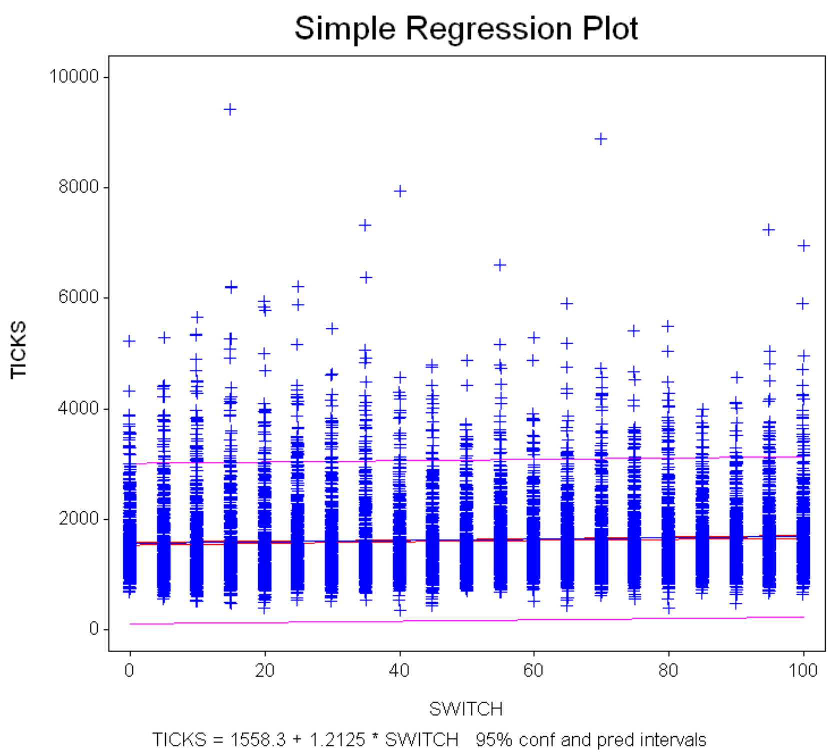
Copyright 2011-2018 Steven F. Railsback and Volker Grimm

In the following plots, the Y axis is the number of ticks at which the 50th mushroom was found, so good parameter values produce low Y values. They indicate that best results occur when the broad search angle is below about 60°; and the local search angle is intermediate (perhaps 90-180°) or high (near 360°) but its effect is small. It is impossible to draw any conclusion about switch time from its plot.



Copyright 2011-2018 Steven F. Railsback and Volker Grimm





Copyright 2011-2018 Steven F. Railsback and Volker Grimm