

## Project 4 – Pine Beetles

### Modelling the spreading of Infection

This model is loosely based on the idea of the spread of pine beetles in a forest. The aim is to understand the factors that allow the coexistence of an infectious agent (beetles in this case) with its host (trees), and to study the evolution of virulence of an infectious agent.

Use the program **beetles1.nlogo**.

Each patch on the lattice may be either

- (i) white = vacant
- (ii) green = healthy tree
- (iii) red = infected tree

On setup, the program starts with 90% trees, 0.1% infected sites, distributed at random.

Trees grow at rate  $r$ . New trees grow in vacancies next to existing trees. In one time step  $\delta t$ , a white patch becomes green with probability  $\delta t \cdot r \cdot n_{green}/8$

Note that  $n_{green}$  is the number of green sites next to the white site. The density of trees on the neighbouring patches is therefore  $n_{green}/8$ . The growth rate is fixed as  $r = 1$  in the program, which sets the time scale for the other rates.

Trees die at rate  $v$  if they are not infected. In one time step, a green site becomes white with probability  $\delta t \cdot v$ . Infected trees die at rate  $v + w$ , *i.e.*  $w$  is the increase in the tree death rate caused by the beetles. In the program supplied,  $v$  is set to 0.1 and  $w$  is set to 1. That means infected trees die much faster than non-infected ones. Additionally, infected trees are not allowed to reproduce.

Beetles spread to neighbouring trees at rate  $k$ . In one time step, each red site has a probability  $\delta t \cdot k$  of infecting a neighbouring site. If this occurs, one random neighbour is selected. If the neighbour is green, then it turns red. The infection only spreads to green neighbours.

**Q1 (10 marks)** – Keep the variables  $v = 0.1$  and  $\delta t = 0.05$  fixed. Run the program with different values of  $k$  and  $w$  to get a feel for what happens. Find values of  $k$  and  $w$  for which each of the following scenarios occurs.

- The beetles all die quickly and the forest remains at high density
- The beetles coexist with the forest. The average beetle density is non-zero and the average tree density is significantly reduced from its natural value without the beetles.
- The beetles almost kill the whole forest and then die because there are too few remaining trees to infect. After the beetles die, a few remaining trees spread and the forest recovers.
- The beetles kill the whole forest. Then the beetles die because there are no more trees.

For each of the above cases, export the plots of the populations of trees and beetles as a function of time and make good graphs of these in Excel, making clear which values of  $k$  and  $w$  were used. Comment on whether the same outcome always happens if you run the program several times with the same parameters. Are there some choices of parameters for which more than one of the above outcomes can occur?

### Measuring mean densities of trees and beetles

Add the following additional features to your program in order to measure the mean densities of green and red sites. Let the number of red and green sites at any one time be  $N_{red}$  and  $N_{green}$ . The densities are  $N_{red}/A$  and  $N_{green}/A$  (where  $A$  is the number of lattice sites). We want to estimate the mean densities, averaged over a long time:

$$\rho_{red} = \langle N_{red}/A \rangle \text{ and } \rho_{green} = \langle N_{green}/A \rangle$$

You can do this by defining variables `sumred` and `sumgreen` in the program. There is some minimum number of ticks `nmin` that you have to wait until the system is in a steady state, and some maximum number of ticks `nmax` at which you can stop the program. When `ticks > nmin` add the current densities to the variables `sumred` and `sumgreen`. When `ticks = nmax`, divide the sums by  $(nmax - nmin)$  to get the time average. Get the program to print the averages into the command centre and then stop.

Suggested values whilst debugging your program are `nmin = 1000` and `nmax = 2000`. These correspond to times 50 and 100 when  $\delta t = 0.05$ . But was the system really in a steady state for the time from `nmin` to `nmax`? Do you need to run the program for longer? Probably `nmin = 4000` and `nmax = 8000` will give you a better time average when you are ready to save your results.

### **Q2 (10 marks)** –

Set the initial fraction of red sites to be zero, and run the program with trees alone. What is the steady state density of trees in absence of beetles? Once you know this, you know that whenever the beetles die out, the density of trees will converge to this same value. For the remainder of this question, change the initial fractions of red and green sites to be 0.2 each. This is useful, because

it helps to reach a steady state with coexisting beetles and trees more easily than the case where we begin with lots of trees and only a few beetles (as we did in Q1).

Keep the variables  $\nu = 0.1$ ,  $\delta t = 0.05$ , and  $w = 1.0$  fixed. Investigate what happens as you change the infection rate  $k$ . Plot the mean density of trees and beetles as a function of  $k$  in the range 0 to 20. Estimate the minimum and maximum values of  $k$  for which the beetles survive, and the value of  $k$  for which the beetle density is highest.

**Q3 (10 marks)** – Using the same program, keep the variables  $\nu = 0.1$ ,  $\delta t = 0.05$ , and  $k = 10.0$  fixed. Investigate what happens as you change  $w$  (the additional death rate of infected trees). Plot the mean density of trees and beetles as a function of  $w$  in the range 0 to 3. Show that there are four regimes corresponding to A-D in Q1. Estimate the minimum and maximum values of  $w$  for which the beetles survive, and the value of  $w$  for which the beetle density is highest. Write a few sentences to explain why too-high  $w$  and too-low  $w$  are both bad for beetles.

### Making Beetles into Agents

So far, we found that beetles have the highest population at intermediate values of  $w$  and  $k$ , and they die out if these parameters are too high or too low. This suggests that moderate values of  $w$  and  $k$  are better, and that the beetles might tend to evolve towards moderate values over time. In order to study the evolution of the beetles, it is useful to re-write the program so that beetles are agents with their own properties, rather than simply patches that are red. In this section, we will write a new version of the program in which beetles are treated as agents but the properties of the beetles are fixed. After that we will allow the properties of the beetles to evolve. Save a new version of your program called **beetles-agent.nlogo** and make the following changes in it.

Use the 'breeds' feature to define a breed of agents called beetles. Add the following line at the beginning of the code before the setup routine.

```
breed [beetles beetle]
```

Read the Netlogo dictionary/manual to see how breeds work.

In the setup routine, you can add the following line to make the beetles look like bugs.

```
set-default-shape beetles "bug"
```

Also in the setup routine, you need to put a beetle on each site that is red.

```
if rr < prinit [  
  sprout-beetles 1 [set color black]  
  set pcolor red  
]
```

Now you have black beetles on red trees! Nice! If you can't see the beetles, you can edit the lattice to make the patch size look bigger.

Now we need to edit the 'to go' routine. The first 'ask patches' loop deals with birth and death of trees. This does not change much, except that when an infected tree dies, we have to make the beetle on this tree die as well. This is just one extra line:

```
if pcolor = red [
  if random-float 1.0 < dt * (v + w) [
    set pcolor white
    ask beetles-here [die]
  ]
]
```

In the **beetles1** program, the loop dealing with the spread of the infection begins

```
ask patches with [pcolor = red] [
```

In the **beetles-agent** program, we need to delete this loop and replace it with an ask beetles loop. The following loop deals with spread of the beetles when the beetles are agents. Make sure you understand this loop, and why it is equivalent to the ask patches loop in the original program.

```
ask beetles [
  if random-float 1.0 < dt * k [
    hatch-beetles 1 [
      move-to one-of neighbors
      if pcolor = red or pcolor = white [die]
      set pcolor red
    ]
  ]
]
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Note that an existing beetle gives birth to a new one with probability  $dt * k$  on the same patch as the parent. The new beetle moves to a neighbouring patch. It dies if there is already another beetle there, or if the site is empty. If the site is green, the beetle infects this site, so pcolor becomes red for this site.

**Q4 (10 marks)** – Include the Netlogo code for your **beetles-agent** program to show that you have added all these features correctly. When you run it, it will look the same as before, except that there will be a black beetle on each of the red trees. The program should still give the same results as before. Measure the mean densities for several of the same parameter values as in Q3, and show that they are the same (apart from a small amount of statistical noise because the random number sequence will be different).

## Allowing Beetles to Evolve

The reason to make beetles into agents was that now we can give beetles their own inherited properties that can evolve. Save another version of the program called **beetles-evolve.nlogo** and make these changes. Beetles have two properties: the infection rate  $k$ , and the effect  $w$  that they have on the death of the trees.

beetles-own [k w]

In this model, the trees don't evolve, so we don't necessarily need to give variables to the patches. However, it will be useful to give the patches a variable  $pw$  that is equal to the  $w$  of the beetle that infects them.

patches-own [pw]

As  $k$  and  $w$  are now properties of the beetles, they are no longer global variables. So you can delete the sliders for  $w$  and  $k$ . Also, we are not making much use of the slider for  $v$ , so you might as well delete that and make a global variable  $v = 0.1$  in the code.

In the setup routine, where the beetles are sprouted, you need to set the initial values – let's start with  $w = 1$  and  $k = 5$ . Then you need to set  $pw = 1$  for the red patches where you just sprouted the beetles.

In the go routine, you need to change the death rate of the red patches to be  $v + pw$ . Remember that  $w$  is a property of beetles, and  $pw$  is a property of patches.

When a beetle moves to a previously uninfected tree, the patch turns red, and the  $pw$  value of this patch has to be set to the  $w$  of the beetle that just moved there. A patch cannot be infected by two beetles at once. If a beetle moves to a patch that is already infected, it dies, and the  $pw$  value of the patch remains equal to the  $w$  of the beetle that is already there.

At the point where you hatch beetles, you need to add mutation. Define a global variable  $pmut$  to be the probability that a mutation occurs in the offspring. We'll start with  $pmut = 0.01$ . Add the following line in the hatch-beetles loop before the beetle moves to a neighbouring patch.

if random-float 1.0 < pmut [set k random-float 20.0]

This means that with a probability 1%, the offspring has a new random value of  $k$  in the range 0 to 20. The default when using hatch is that the offspring is the same as the parent. Therefore, with probability 99%, the offspring will inherit the same  $k$  value as the parent. If you want  $w$  to evolve as well, you can add a similar line that sets  $w$  to a random value. A good range for  $w$  is 0 to 3. It is best to write the  $w$  line separate from the  $k$  line so that mutations in the two quantities occur independently.

Add plots to show the mean values  $\langle w \rangle$  and  $\langle k \rangle$  of the turtles as a function of time. You can get rid of the time averages of the densities now, because these will change as time goes on. Also

get rid of the line that stops the program when  $\text{ticks} = \text{nmax}$ . You will need to leave the evolutionary program running for a long time to see what happens.

### Q5 (10 marks)

Begin with  $k = 5$ , and  $w = 1$ . Allow only mutations in  $k$ . Run the program for a long time and describe what happens. Does the same thing always happen? Export the data for  $\langle k \rangle$  as a function of time and show this for several runs.

Now begin with  $k = 10$  and  $w = 1$ . Allow only mutations in  $w$ . Show the graph of  $\langle w \rangle$  as a function of time for several runs. Explain what happens.

What happens if both  $k$  and  $w$  can evolve at the same time?

Do these results follow our initial expectation that the beetles would evolve to maximize their population?

### Q6 (10 marks)

You will probably find in Q5 that the beetles always evolve towards the unstable region of the parameter space; hence they die out. There seems to be something missing in our model. One problem is that we have not accounted for trade-offs in parameters. An evolutionary trade-off is a situation when there is a dependency between parameters – one parameter cannot keep getting better without another one getting worse.

Let's add a trade-off between  $k$  and  $w$ . The main property of the beetles that can change is the size of the infestation on an infected tree. Let's say  $P$  is the beetle population on a tree when the beetles have taken over. We would expect the rate of spread to neighbouring trees,  $k$ , to be proportional to  $P$ . We would also expect the effect on the tree,  $w$ , to be proportional to  $P$ . We don't really want to keep track of the number of beetles on every tree, so we will simply introduce a trade-off by saying  $w$  is proportional to  $k$ .

For this question, let's assume that  $w = k/5$ . Start with some initial value of  $k$ , and set  $w = k/5$  for all beetles. Allow mutation in  $k$  only. This time, when  $k$  changes,  $w$  has to change at the same time so that it remains equal to  $k/5$ . Run the program and measure  $\langle k \rangle$  as a function of time. Start from several different initial values of  $k$ . Explain what happens. You will probably find that the system is now stable under evolution when the trade-off is added.

### Q7 (10 marks)

Beetles are pretty nasty things in this model! Once a tree is infected, it never recovers, it never reproduces, and it dies much faster than without the beetles. I hope none of us ever catches a disease as bad as this!

Let's change the model to allow the beetles to evolve to be less harmful to the tree. Suppose that infected trees can reproduce at some rate  $r$  that is less than uninfected trees. At the moment we have  $r = 1$  hardwired in for green trees and  $r = 0$  (no reproduction) for red trees. Now you can give the beetles a variable  $r$  that evolves, and the patches a variable  $pr$  that is determined by the  $r$  of the beetle that infects the patch. This is the same way that we dealt with  $w$

and  $p_w$  before. Green trees always have  $r = 1$ , but now red trees have an  $r$  that can evolve in the range 0 to 1. It can't be more than 1 because we don't want the infected trees to reproduce faster than the uninfected ones. Assume that the infected parent (red) gives birth to an uninfected offspring (green). The offspring can only grow in a vacant site, in the same way as usual.

Start with  $k = 5$ ,  $w = 2$  and  $r = 0$ . Allow all three properties to evolve at the same time with  $p_{mut} = 1\%$ . Mutations occur independently in each property. Don't bother with any trade-offs between parameters. Show the graph of the mean parameters as a function of time and explain what happens. If the system is stable, estimate the mean density of trees and beetles. You should see that beetles evolve to a much higher population than before, but their effect on the trees is much less.

*Optional question for those who are interested (no marks).*

*Suppose the trees can recover from the infection. There is some rate, call it  $b$ , at which the beetles on an infected tree die, and the red tree goes back to being green. Assume that  $b$  is a property of the beetle, like  $w$  and  $r$ .*

*How would you put this in the model? What would you expect to happen? Should beetles evolve to have a high  $b$  or a low  $b$ ? Try it and see!*

#### Q8 (10 marks)

The evolution of virulence of infectious diseases is an important topic in biology. 'Virulence' means how harmful is the infection to the host. In different cases, virulence could mean how likely is the host to die, how long is the host ill, or what is the effect on the reproduction of the host. There is a general expectation that if a disease has recently moved to a new host, it can be quite virulent, whereas if a disease has co-evolved with a host for a long time, it tends to become less harmful. There are lots of factors that influence this.

Look up published scientific papers that mention 'evolution of virulence'. It could be diseases of humans or any other type of host organism, and the infection could be caused by microorganisms, or larger parasites, such as insects. Choose one good example where some experimental data is available that tells us something about how the virulence of this disease/parasite evolves. For this example, write about half a page to describe the species involved, how the transmission spreads, and what was found about the evolution of virulence. Cite the paper properly in your report.

**Total marks 80 for this sheet.**