

11.2 Agents of Interaction: Steering a Dangerous Course

NetLogo Quick Review Questions

*Introduction to Computational Science:
Modeling and Simulation for the Sciences, 2nd Edition*
Angela B. Shiflet and George W. Shiflet
Wofford College
© 2014 by Princeton University Press

Agent-Based Modeling

- Quick Review Question 1** Indicate to which each of the following applies, cellular automaton (CA) simulations, agent-based (AB) simulations, or both:
- a. Autonomous, decision-making entity has a state and behaviors
 - b. Grid cell has state and transition rules specify next state
 - c. Relationship with neighbors determines next state
 - d. Can use grid
 - e. For each time step, iteration is over each grid cell
 - f. For each time step, iteration is over each autonomous, decision-making entity
 - g. Local interactions can cause global change

Compose all the following answers in NetLogo:

Model Environment

- Quick Review Question 2** Write the *WHEN-CREATING-NEW-AGENT* function for initialization of a *Farm* agent.

Agents and Their States

- Quick Review Question 3** Write the *WHEN-CREATING-NEW-AGENT* function for initialization of a *Cattle* agent and the associated *countSIRM* function.

Agent Behaviors

- Quick Review Question 4** Write cattle *WHILE-RUNNING* function, which is the cattle scheduler function.

- Quick Review Question 5** Write cattle *sir* function.

Quick Review Question 6 Write cattle *sirSlaughterhouse* function.

Quick Review Question 7 Write cattle *inFarm* function.

Quick Review Question 8 Write cattle *farm2Sale* function.

Quick Review Question 9 Write cattle *inSaleBarn1* function.

Quick Review Question 10 Write cattle *moveInSalebarn* function.

Quick Review Question 11 Write cattle *inStocker* function.

Quick Review Question 12 Write cattle *inFeedlot* function.

Model Refinement

Quick Review Question 13 Write the *SimulationDriver*'s *WHILE-RUNNING* function.

Quick Review Question 14 Write cattle *initI* function.

Answers to Quick Review Questions

1. a. AB b. CA c. both d. both
 e. CA f. AB g. both

2.

↳ **WHEN-CREATING-NEW-AGENT ()**

Procedure to initialize a Farm agent possibly to have a Cattle agent on top

Number of rules: 1

If

SEE ([] , []), and
 %-CHANCE (@INIT_CATTLE_FRACTION*100)

Then

NEW ([] , [])

3.

▾ WHEN-CREATING-NEW-AGENT ()


Procedure to initialize new calf with a random weight between 60 and 100 pounds, to establish the days sick to be 0 for an infected calf, and to establish various category counters

Number of rules: 2

If

SEE ( , )


Then

SET (weight , to , 60+random(41.0)), and
SET (daysSick , to , 0), and
MAKE ( , countSIRM)

If

SEE-A ( , Cattle)

Then

SET (weight , to , 60+random(41.0)), and
MAKE ( , countSIRM)

▾ ON (countSIRM)

Procedure to update numS, numI, numR, numM, and numCattle after addition of a new beef cow

Number of rules: 4

If

SEE ( , )

Then

SET (@numS , to , @numS + 1), and
SET (@numCattle , to , @numCattle + 1)

If

SEE ( , )

Then

SET (@numI , to , @numI + 1), and
SET (@cumulativeI , to , @cumulativeI + 1), and
SET (@numCattle , to , @numCattle + 1)

If

SEE ( , )

Then

SET (@numR , to , @numR + 1), and
SET (@numCattle , to , @numCattle + 1)

If

SEE ( , )

Then

SET (@numM , to , @numM + 1), and
SET (@numCattle , to , @numCattle + 1)

4.

▷WHILE-RUNNING ()



Cattle Driver of Simulation

Number of rules: 7

If

STACKED-A (immediately above , a , Farm), and
 IS (weight , < , 600)



Then

MAKE ( , sir), and
 MAKE ( , inFarm)

If

STACKED-A (immediately above , a , Farm)



Then

MAKE ( , sir), and
 MAKE ( , farm2Sale)

If

IS (weight , < , 900), and
 STACKED-A (somewhere above , a , Salebarn)



Then

MAKE ( , sir), and
 MAKE ( , inSaleBarn1)

If

STACKED-A (immediately above , a , Stocker)



Then

MAKE ( , sir), and
 MAKE ( , inStocker)

If

IS (weight , >= , 900), and
 STACKED-A (somewhere above , a , Salebarn)



Then

MAKE ( , sir), and
 MAKE ( , inSaleBarn2)

If

STACKED-A (somewhere above , a , Feedlot)





Then

MAKE ( , sir), and
 MAKE ( , inFeedlot)

If

STACKED-A (somewhere above , a , Slaughterhouse)

Then

MOVE (), and
 MAKE ( , sirSlaughterhouse), and
 CHANGE ( , )

5.

ON (sir)

Procedure to advance an infected beef cow's illness, possibly to recovery, and determine if a susceptible cattle agent becomes sick

Number of rules: 3

If

```
SEE ([ ] , [ ]), and
IS (daysSick , > , @INFECTIOUS_PERIOD)
```

Then

```
CHANGE ([ ] , [ ]), and
SET (daysSick , to , 0), and
SET (@numI , to , @numI - 1), and
SET (@numR , to , @numR + 1)
```

If

```
SEE ([ ] , [ ])
```

Then

```
SET (daysSick , to , daysSick + 0.25)
```

If

```
SEE ([ ] , [ ]), and
NEXT-TO (>= , 1 , [ ]), and
%-CHANCE (@INFECTION_PROBABILITY * 100)
```

Then

```
CHANGE ([ ] , [ ]), and
SET (@numS , to , @numS - 1), and
SET (@numI , to , @numI + 1), and
SET (@cumulativeI , to , @cumulativeI + 1), and
SET (daysSick , to , 0)
```

6.

ON (sirSlaughterhouse)

Procedure to adjust appropriate system variables when a beef cow is slaughtered

Number of rules: 4

If

```
SEE ([ ] , [ ])
```

Then

```
SET (@numS , to , @numS - 1)
```

If

```
SEE ([ ] , [ ])
```

Then

```
SET (@numI , to , @numI - 1)
```

If

```
SEE ([ ] , [ ])
```

Then

```
SET (@numR , to , @numR - 1)
```

If

```
SEE ([ ] , [ ])
```

Then

```
SET (@numM , to , @numM - 1)
```

7.

ON (inFarm)

Cattle agent's behavior on a Farm tile

Number of rules: 1

```

If
  no condition
Then
  MOVE-RANDOM-ON ( ), and
  SET (weight , to , weight + 0.5 + random(0.25))

```

8.

ON (farm2Sale)

Cattle agent's behavior in moving from Farm tiles to SaleBarn tiles

Number of rules: 6

```

If
  SEE ( , )
Then
  MOVE ( )

If
  SEE ( , )
Then
  MOVE ( )

If
  SEE ( , )
Then
  MOVE ( )

If
  SEE ( , )
Then
  MOVE ( )

If
  SEE ( , )
Then
  MOVE ( )

If
  SEE ( , )
Then
  MOVE ( ), and
  SET (timeInSale , to , 1 + random(5))

```

9.

¬ON (inSaleBarn1)

*Cattle agent's behavior when in sale barn for the first time**Number of rules: 2*

```

If
  IS (timeInSale , > , 8)
Then
  MOVE (◀)

If
  no condition
Then
  SET (timeInSale , to , timeInSale + 1), and
  MAKE (◻ , moveInSaleBarn)

```

10.

¬ON (moveInSaleBarn)

*Procedure for a cattle agent's random movement in a sale barn**Number of rules: 5*

```

If
  NEXT-TO (> , 0 , ◼)
Then
  MOVE-RANDOM-ON (◼)

If
  NEXT-TO (> , 0 , ◼)
Then
  MOVE-RANDOM-ON (◼)

If
  NEXT-TO (> , 0 , )
Then
  MOVE-RANDOM-ON ( )

If
  NEXT-TO (> , 0 , ◼◼)
Then
  MOVE-RANDOM-ON (◼◼)

If
  NEXT-TO (> , 0 , ◼◼◼)
Then
  MOVE-RANDOM-ON (◼◼◼)

```



11.

▼ON (inStocker)


Procedure for a cattle agent's behavior in stocker

Number of rules: 3



If

IS (weight , >= , 900), and
SEE ( , )


Then

MOVE ()

If

IS (weight , >= , 900), and
SEE ( , )


Then

MOVE (), and
SET (time2InSale , to , 1 + random(5))

If

no condition

Then

MOVE-RANDOM-ON (), and
SET (weight , to , weight + 0.4 + random(0.2))


12.




▼ON (inFeedlot)




Procedure for a cattle agent's behavior in feedlot




Number of rules: 5

```

If
  IS (weight , >= , 1300)
Then
  MOVE (  )

If
  SEE (  ,  )
Then
  MOVE (  ), and
  SET (weight , to , weight + 0.7 + random(0.1))

If
  SEE (  ,  )
Then
  MOVE (  ), and
  SET (weight , to , weight + 0.5 + random(0.5))

If
  SEE (  ,  )
Then
  MOVE (  ), and
  SET (weight , to , weight + 0.5 + random(0.5))

If
  no condition
Then
  SET (weight , to , weight + 0.5 + random(0.5))

```

13.

▼WHILE-RUNNING ()

*Put text here to explain what this method does!**Number of rules: 5*

```

If
  IS (@phase , = , 0)
Then
  BROADCAST (Farm , randomCattle), and
  SET (@phase , to , 1)

If
  IS (@phase , = , 1), and
  IS (@numI , = , 0)
Then
  BROADCAST (Cattle , initI)

If
  IS (@phase , = , 1)
Then
  SET (@phase , to , 2)

If
  IS (@phase , = , 2), and
  IS (@numI , = , 0)
Then
  STOP-SIMULATION ()

If
  IS (@phase , = , 2)
Then
  BROADCAST (Cattle , cattleBehave)

```

14.

▼ON (initI)

*Method to change Cattle agent to InfectedCattle agent with probability 1/numCattle**Number of rules: 1*

```

If
  SEE ([ ] , [ ]), and
  IS (@numS , = , @numCattle), and
  %-CHANCE (100/@numCattle)
Then
  CHANGE ([ ] , [ ]), and
  SET (@numS , to , @numS - 1), and
  SET (@numI , to , 1), and
  SET (@cumulativeI , to , 1)

```