## Answers to Quick Review Question

From the text's website, download your computational tool's *10_2QRQ.pdf* file for answers to the system-dependent questions.

2. $3.6 = (1 - 8 \times 0.1)(5) + 0.1(2 + 3 + 4 + 0 + 6 + 1 + 3 + 7)$
3. a. $5 \times 5$
   b. 0, 0, 0, 0, 0
   c. 2, 2, 3, 4, 4
   d. 7, 1, 3, 7, 1
5. a. $95 \times 60$
   b. (32, 25)
   c. (34, 24)

## References


Cunningham, Steve. 2007. *Computer Graphics: Programming in OpenGL for Visual Communication*. Upper Saddle River, NJ: Prentice-Hall.

NASA Spinoff. 1988. "Spinoff from Mooncraft Technology." NASA.

NASA Spinoff. 2011. "Fire-Resistant Reinforcement Makes Steel Structures Sturdier." http://spinoff.nasa.gov/Spinoff2006/ps_3.html (accessed June 15 2012)

Podroužek, Jan, and Břetislav Teplý. 2008. "Modelling of Chloride Transport in Concrete by Cellular Automata." *Engineering Mechanics*, (15)3: 213–222.

Wikipedia Contributors, "Diffusion," *Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/wiki/Diffusion (accessed June 13, 2012)

Wikipedia Contributors, "Heat Transfer," *Wikipedia, The Free Encyclopedia*, http://en.wikipedia.org/wiki/Heat_transfer (accessed June 13, 2012)

# MODULE 10.3

## Spreading of Fire

*Prerequisite: Module 10.2, "Diffusion: Overcoming Differences."*

### Downloads

For several computational tools, the text's website has available for download a *Fire* file containing the simulation this module develops and a *10_3QRQ.pdf* file containing system-dependent Quick Review Questions and answers.

### Introduction

Human beings, with some justification, have considerable fear of fire. History is replete with disastrous losses of life and property from it. Nevertheless, fires in areas like the western United States are natural and, ecologists tell us, beneficial to the plant communities there. Periodic fires help to clear the forest floor of debris and promote the growth of sturdy, fire-resistant trees. Unfortunately, expanding human populations have intruded on previously uninhabited areas, establishing their own communities in "fire-prone" zones. Furthermore, human activities, such as fire suppression, livestock grazing, and logging, have increased the possibility of hotter and more destructive fires (NPS 2012).

During the fall of 2003, residents of Southern California faced a series of firestorms driven by powerful Santa Ana winds. After 3 days, the fires had destroyed more than 400,000 acres and 900 homes and had killed 15 people. Hundreds of firefighters battled a chain of fires that extended from Ventura County, north of Los Angeles, east into San Bernadino County and south to Tijuana, Mexico. A haze of toxins draped over the area like a pall (Wilson et al. 2003).

The Malibu region above Los Angeles is dominated by the Santa Monica Mountains and canyons that run from north to south. Much of the natural vegetation is dry **chaparral**, consisting of many small, oily, woody plants that are extremely flammable. This vegetation naturally would burn every 15 to 45 years, clearing out old and dead plant materials and returning nutrients to the soil. With the prevailing dry conditions, an illegal campfire can set off a ferocious blaze that may stop only after traveling many miles to the Pacific Ocean (SBCCBS 2010; Los Angeles Times 2010).

Fighting fires in Southern California or anywhere else is a very risky job, where loss of life is a real possibility. Proper training is essential. In the United States the National Fire Academy, established in 1974, presents courses and programs that are intended "to enhance the ability of fire and emergency services and allied professionals to deal more effectively with fire and related emergencies." The Academy has partnered with private contractors and the U.S. Forest Service to develop a 3D land fire-fighting training simulator. This simulator exposes trainees to a convincing fire-propagation model, where instructors can vary fuel types, environmental conditions, and topography. Responding to these variables, trainees may call for appropriate resources and construct fire lines. Instructors may continue to alter the parameters, changing fire behavior. Students can review the results of their decisions, where they can learn from their mistakes in the safety of a computer laboratory (DAS 2012).

This module develops a two-dimensional computer simulation for the spread of fire. The techniques can be extended to numerous other scientific examples involving contagion, such as the propagation of infectious diseases and distribution of pollution.

## Problem

Our problem is to simulate the spread of fire from an initial landscape of empty ground, nonburning trees, and trees that are on fire. Moreover, the area can suffer from lightning strikes, which may or may not start additional fires.

## Initializing the System

For our cellular automaton simulation of the spread of fire, a cell of an $n \times n$ grid can contain a value of 0, 1, or 2 indicating an empty cell, a cell with a nonburning tree, or a cell with a burning tree, respectively. Table 10.3.1 lists these values and meanings, along with associated constants, ***EMPTY***, ***TREE***, and ***BURNING***, which have values of **0**, **1**, and **2**, respectively. We initialize these constants at the beginning and employ the descriptive names throughout the program. Thus, the code is easier to understand and to change.

**Table 10.3.1**
Cell Values with Associated Constants and their Meanings

| *Value* | *Constant* | *Meaning* |
|---|---|---|
| 0 | *EMPTY* | The cell is empty ground containing no tree. |
| 1 | *TREE* | The cell contains a tree that is not burning. |
| 2 | *BURNING* | The cell contains a tree that is burning. |

To initialize this discrete stochastic system, we employ the following two probabilities:

***probTree***: The probability that a tree (burning or not burning) initially occupies a site. Thus, *probTree* is the initial tree density measured as a percentage.
***probBurning***: If a site has a tree, the probability that the tree is initially burning or that the grid site is *BURNING*. Thus, *probBurning* is the fraction of the trees that are burning when the simulation begins.

Using the preceding probabilities and cell values, we employ the following logic in a function, ***initForest***, to return an initialized grid for the forest. In the pseudocode, two slashes, //, indicate that the rest of the line is a comment.

***initForest(n, probTree, probBurning)***

Function to return an $n \times n$ grid of values—*EMPTY* (no tree), *TREE* (nonburning tree), or *BURNING* (burning tree)—where *probTree* is the probability of a tree and *probBurning* is the probability that the tree is burning

***Pre:*** $n$ is the size (number of rows or columns) of the square grid and is positive.
*probTree* is the probability that a site is initially occupied by tree.
*probBurning* is the probability that a tree is burning initially.
***Post:*** A grid as described earlier was returned.
***Algorithm:***

```
for every cell in an n × n grid, forest, do the following:
   if a random number is less than probTree            // tree at site
      if another random number is less than probBuring // tree is burning
         assign BURNING to the cell
      else                                             // tree is not burning
         assign TREE to the cell
   else  // no tree at site
      assign EMPTY to the cell
return forest
```

## Quick Review Question 1

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that implements *initForest*.

## Updating Rules

At every simulation iteration, we apply a function ***spread*** to each cell site to determine its value—*EMPTY*, *TREE*, or *BURNING*—at the next time step. The cell's value at the next instant depends on the values of the cells in its von Neumann neighborhood, as in Figure 10.2.3a—the cell's current value (***site***) and the values of its neighbors to the north (*N*), east (*E*), south (*S*), and west (*W*). For this simulation, the state of a diagonal cell to the northeast, southeast, southwest, or northwest does not have an impact on a site's value at the next iteration. Thus, we include five parameters—*site*, *N*, *E*, *S*, and *W*—for *spread*. (Shortly, we will see that *spread* should have two additional parameters.) In a call to this function, each neighborhood argument is one of three values: *EMPTY*, indicating an empty cell with no tree, *TREE* for a nonburning tree, or *BURNING* for a burning tree in that location.

Updating rules apply to different situations: If a site is empty (cell value *EMPTY*), it remains empty at the next time step. If a tree grows at a site (cell value *TREE*), at the next instant the tree may or may not catch fire (value *BURNING* or *TREE*, respectively) due to fire at a neighboring site or to a lightning strike. A burning tree (cell value *BURNING*) always burns down, leaving an empty site (value *EMPTY*) for the next time step. We consider each situation separately.

## Quick Review Question 2

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that develops *spread*'s rule for the situation where a site does not contain a tree at this or any time step.

When a tree is burning, the first argument, which is the site's value, is *BURNING*. Regardless of its neighbors' situations, the tree burns down, so that at the next iteration of the simulation the site's value becomes *EMPTY*. Thus, the relevant rule for the *spread* function has a first argument of *BURNING*; each of the other four arguments are immaterial; and the function returns value of *EMPTY*.

## Quick Review Question 3

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that develops *spread*'s rule for the situation where a site contains a burning tree.

To develop this dynamic, discrete stochastic system, we employ the following additional probabilities, which we include as parameters for *spread*:

***probImmune***: The probability of immunity from catching fire. Thus, if a site contains a tree (site value of *TREE*) and fire threatens the tree, *probImmune* is the probability that the tree will not catch fire at the next time step.

***probLightning***: The probability of lightning hitting a site.

When a tree is at a location (site value of *TREE*), at the next iteration the tree might be burning due to one of two causes, a burning tree at a neighboring site or a lightning strike at the site itself. Even if one of these situations occurs, the tree at the site might not catch fire. Separate rules apply to the two causes for fire.

For the first situation involving a neighboring burning tree, we employ the following logic:

```
if (site is TREE) and (N, E, S, or W is BURNING)
    if a random number between 0.0 and 1.0 is less than probImmune
        return TREE
    else
        return BURNING
```

Thus, even if a tree has the potential to burn because of a neighboring burning tree, it may not. Because of conditions such as wet weather, such a tree has a probability of *probImmune* of not burning.

### Quick Review Question 4

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that develops *spread*'s rule for the situation where a site contains a nonburning tree that may catch fire because a neighboring site contains a burning tree.

A tree might also catch fire because of a lightning strike. The probability that the tree is struck by lightning is *probLightning*. However, with a probability of *probImmune*, the tree will not burn even if hit by lightning. In contrast, the probability that the tree is not immune to fire is (1 – *probImmune*). For example, if the probability of immunity (*probImmune*) is 0.4 = 40%, then a (1 – 0.4) = 0.6 = 60% chance exists for the tree not to be immune from burning. For the tree to catch fire due to lightning, it must be hit and not be immune. Thus, lightning causes a tree to catch fire with the probability that is the product *probLightning* * (1 – *probImmune*). For example, if a 0.2 = 20% chance exists for a lightning strike at the site of a tree, the tree burns with a probability of (0.2)(0.6) = 0.12 = 12%. Two things must happen: Lightning must strike, and the tree must not be immune from burning.

### Quick Review Question 5

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that completes *spread*'s rule for the situation where a site contains a nonburning tree that may be hit by lightning and burn.

## Periodic Boundary Conditions

For this simulation, we apply the function *spread* to every grid point, using periodic boundary conditions. Thus, to apply *spread* we extend the boundaries by one cell, as in Figure 10.2.7. The next two quick review questions extend the grid first to the north and south and then to the east and west.

### Quick Review Question 6

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that extends a grid, as in Figure 10.3.1, by attaching

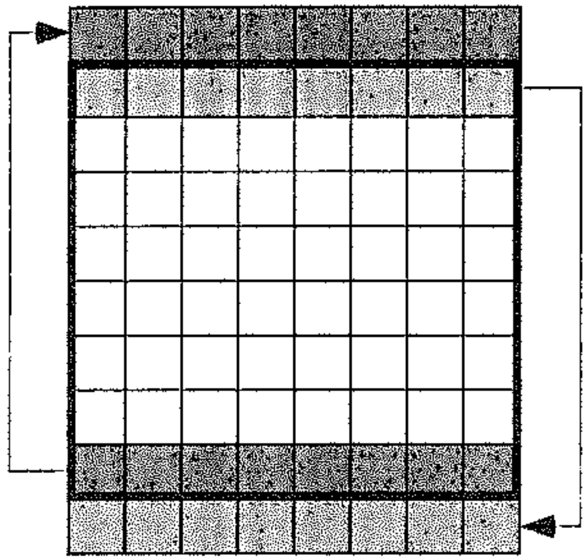

**Figure 10.3.1** Grid (in bold square) extended by having a new first row that is a copy of the last row on the original grid and having a new last row that is a copy of the first row on the original grid

the last row to the beginning and the first row to the end of the original grid to form a new grid, *matNS*.

## Quick Review Question 7

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that extends a lattice as in Figure 10.3.2.

To consolidate these tasks, we define a function, ***periodicLat***, using periodic boundary conditions to extend the square lattice by one cell in each direction. Pseudocode for the function follows.

***periodicLat(lat)***

Function to accept a grid and to return a grid extended one cell in each direction with periodic boundary conditions

***Pre:*** *lat* is a grid.
***Post:*** A grid extended one cell in each direction with periodic boundary conditions was returned.
***Algorithm:***
*latNS* ← concatenation of last row of *lat*, *lat*, and first row of *lat*
return concatenation of last column of *latNS*, *latNS*, and first column of *latNS*

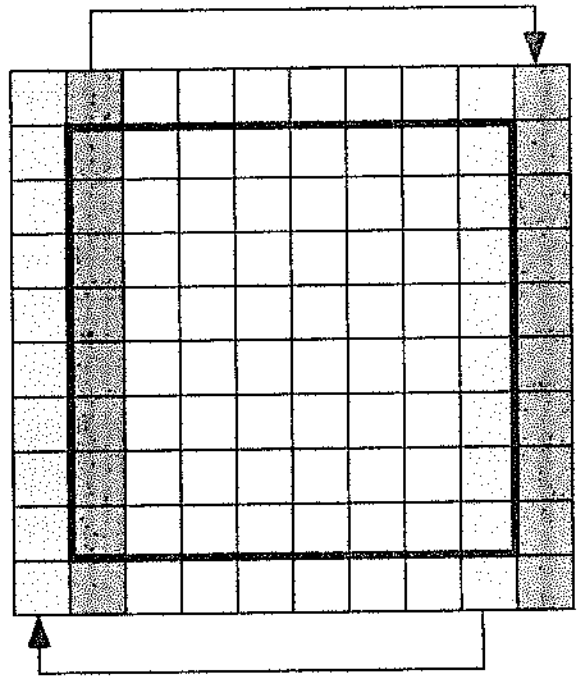

**Figure 10.3.2** Grid from Figure 10.3.1 expanded by having a new first column that is a copy of the last column and a new last column that is a copy of the first column

## Applying a Function to Each Grid Point

After extending the grid by one cell in each direction using periodic boundary conditions, we apply the function *spread* to each internal cell and then remove the boundary cells. Similar to *applyDiffuseExtended* in Module 10.2, a function ***applyExtended*** takes an extended square lattice (*latExt*) and two probabilities (*probLightning* and *probImmune*) that *spread* requires and returns the internal lattice with *spread* applied to each site.

### Quick Review Question 8

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that develops the function *applyExtended*.

## Simulation Program

To drive the simulation of spreading fire, we define a function *fire* with parameters *n*, the grid size, or number of grid rows or columns; *probTree*; *probBurning*; *probLightning*, the probability of lightning hitting a site; *probImmune*, the probability of immunity from catching fire; and *t*, the number of time steps. As with *diffusionSim* from Module 10.2. the function *fire* returns a list of the initial lattice and the next *t* lattices in the simulation. The functions *spread* and *fire* need the probabilities of lightning and immunity. Pseudocode for *fire* is as follows.

***fire(n, probTree, probBurning, probLightning, probImmune, t)***

Function to return a list of grids in a simulation of the spread of fire in a forest, where a cell value of *EMPTY* indicates the cell is empty; *TREE*, the cell contains a nonburning tree; and *BURNING*, a burning tree

***Pre:***

*n* is the size (number of rows or columns) of the square grid and is positive.
*probTree* is the probability that a site is initially occupied by tree.
*probBurning* is the probability that a tree is burning initially.
*probLightning* is the probability of lightning hitting a site.
*probImmune* is the probability of a tree being immune from catching fire.
*t* is the number of time steps
*spread* is the function for the updating rules at each grid point.

***Post:***

A list of the initial grid and the grid at each time step was returned.

***Algorithm:***

*forest* ← *initForest(n, probTree, probBurning)*
*grids* ← list containing *forest*
do the following *t* times:
  *forestExtended* ← *periodicLat(forest)*
  *forest* ← *applyExtended(forestExtended, probLightning, probImmune)*
  *grids* ← the list with *forest* appended onto the end of *grids*
return *grids*

## Quick Review Question 9

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that implements the loop in the *fire* function.

## Display Simulation

For each lattice in the list returned by *fire*, we generate a graphic for a rectangular grid, with yellow representing an empty site; green, a tree; and burnt orange, a burning tree. The function ***showGraphs*** with parameter *graphList* containing the list of lattices from the simulation produces these figures. We animate the sequence of graphics to view the changing forest scene.

### Quick Review Question 10

From the text's website, download your computational tool's *10_3QRQ.pdf* file for this system-dependent question that develops the function *showGraphs*, which produces an animation with a graphic corresponding to each simulation lattice in a list (*graphList*).

Figure 10.3.3 displays several frames of a fire sequence in which empty cells are white; burning cells are in color; and cells with nonburning trees are gray. Clearly, different initial random number generator seeds result in different sequences. This simulation employs the parameters $n = 50$, $probTree = 0.8$, $probBurning = 0.0005$, $probLightning = 0.00001$, $probImmune = 0.25$, and $t = 50$. The initial graphic displays one fire toward the bottom of the grid. At time step $t = 2$, a lightning strike starts a fire at an isolated location toward the top of the grid. Subsequent frames show both fires spreading to neighboring cells. Grids for times starting at $t = 14$ reveal the influence of periodic boundary conditions as the fire at the bottom spreads to the top of the grid, and vice versa.

## Exercises

*On the text's website,* Fire *files for several computational tools contain the code for the simulation of the module. Complete the following exercises using your computational tool.*

*For Exercises 1–3, write update rules for* spread, *where neighbor refers to a location in the von Neumann neighborhood other than the site itself. Revise grid values as necessary.*

1. A tree takes two time steps to burn completely.
2. A tree catches on fire from neighboring trees with a probability proportional to the number of neighbors on fire.
3. A tree grows instantaneously in a previously empty cell with a probability of *probGrow*.
4. Describe changes to the code to include diagonal elements as neighbors as well.
5. Write the code to assign the values to the northeast, southeast, southwest, and northwest to variables *NE*, *SE*, *SW*, and *NW*, respectively, of a site in the lattice *latExt*.
6. Suppose a lattice *g* has values for a forest grid, where a cell can be empty (value $EMPTY = 0$), a tree with the value (1 through 4) indicating the level of maturity from young to old, or a burning tree with the value indicating the intensity of the fire (5 for less intense or 6 for intense). Write code to show a graphic representing *g*, with yellow for an empty cell, a different level of green from pale to full green representing the age of a tree, light red for a less-intense fire and full red for an intense fire. Use constants, such as *EMPTY*, for the cell values.

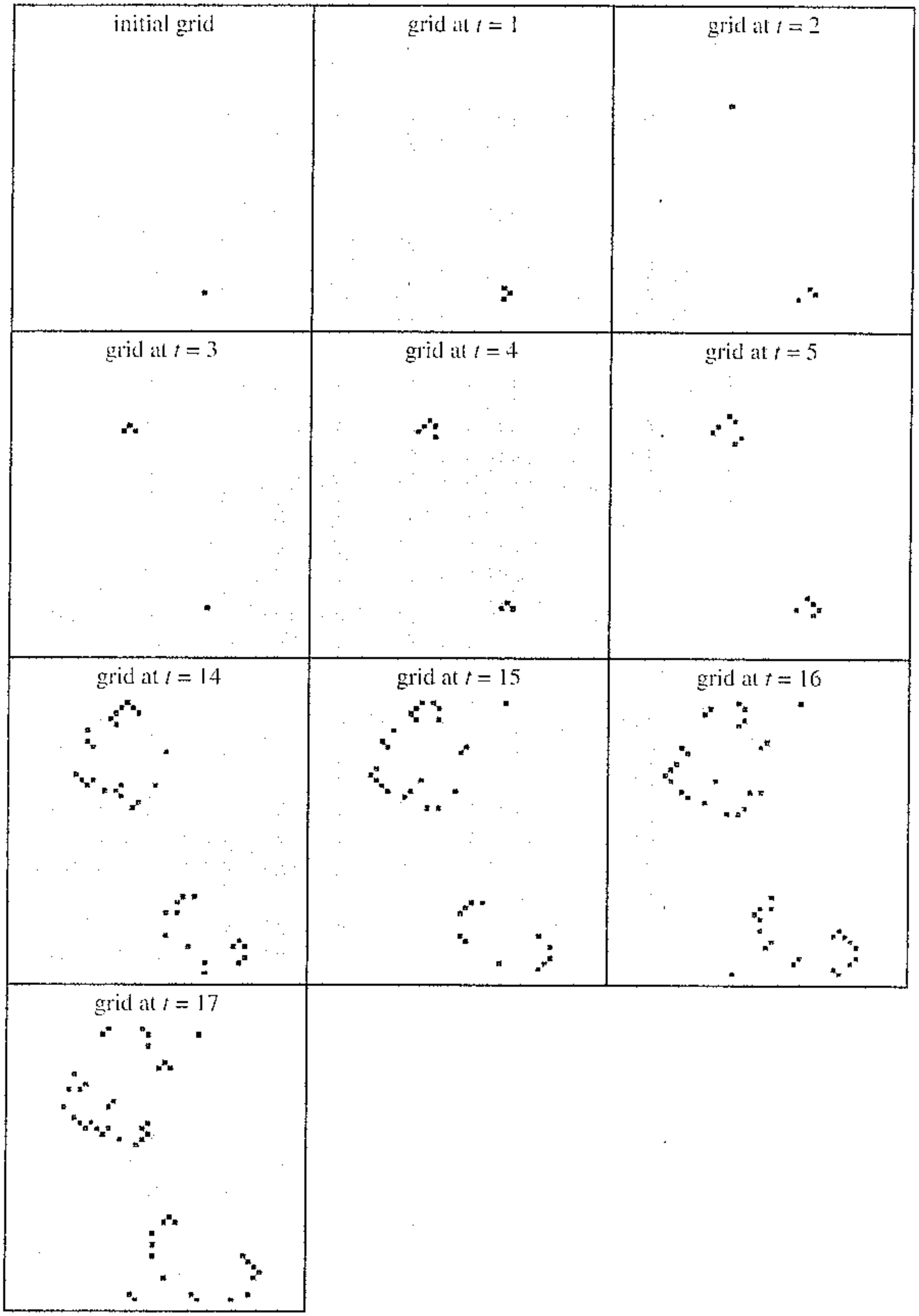


**Figure 10.3.3** Several frames in an animation of the spreading of fire

## Projects

*On the text's website,* Fire *files for several computational tools contain the code for the simulation of the module. Complete the following projects using your computational tool.*

*For additional projects, see Module 14.3, "Foraging—Finding a Way to Eat"; Module 14.4, "Pit Vipers—Hot Bodies, Dead Meat"; Module 14.5, "Mushroom Fairy Rings—Growing in Circles"; Module 14.6, "Spread of Disease—Sharing Bad News"; Module 14.7, "HIV—The Enemy Within"; Module 14.8, "Predator-Prey—'Catch Me If You Can'"; Module 14.9, "Clouds—Bringing It All Together"; Module 14.11, "Spaced Out—Native Plants Lose to Exotic Invasives"; and Module 14.12, "Re-Solving the Problems with Cellular Automaton Simulations."*

**1.** Run the simulation for fire several times for each of the following situations and discuss the results.

**a.** *probBurning* is almost 0; *changeLightning* = *changeImmune* = 0

**b.** *probBurning* is 0; *changeImmune* is 0

**c.** *probBurning* is 0; *changeLightning* is 0

**d.** Devise another situation to consider.

*In each of Projects 2–8, revise the fire simulation to incorporate the change indicated in the exercise or boundary condition. Discuss the results.*

**2.** Exercise 1 **3.** Exercise 2 **4.** Exercise 3

**5.** Exercise 4 **6.** Exercise 6

**7.** Absorbing boundary conditions **8.** Reflecting boundary conditions

**9.** Develop a fire simulation in which every cell in a 17 × 17 grid has a tree and only the middle cell's tree is on fire initially. Do not consider the possibility of lightning or tree growth. The simulation should have a parameter for *burnProbability*, which is the probability that a tree adjacent to a burning tree catches fire. The function should return the percent of the forest burned. The program should run eight experiments with *burnProbability* = 10%, 20%, 30%, . . .. and 90% and should conduct each experiment 10 times. Also, have the code determine the average percent burned for each probability. Plot the data and fit a curve to the data. Discuss the results (Shodor Educational Foundation, "Fire").

**10. a.** Develop a fire simulation that considers wind direction and speed. Have an accompanying animation. Do not consider the possibility of lightning. The simulation should have parameters for the probability (*probTree*) of a grid site being occupied by a tree initially, the probability of immunity from catching fire, the fire direction (value *N*, *E*, *S*, or *W*), wind level (value *NONE* = 0, *LOW* = 1, or *HIGH* = 2), coordinates of a cell that is on fire, and the number of cells along one side of the square forest. The function should return the percent of the forest burned (Shodor Educational Foundation, "Better Fire").

**b.** With a wind level of *LOW* (1) and a fixed *probTree*, vary wind direction and through animations observe the affects on the forest burn. Discuss the results.

c. Develop a program to run three experiments with wind levels of $NONE = 0$, $LOW = 1$, and $HIGH = 2$. Have fixed wind direction and *probTree*. The program should conduct each experiment 10 times. Also, have the code determine the average percent burned for each level. Discuss the results.

d. Develop a program to run eight experiments with no wind and *probTree* = 10%, 20%, 30%, . . ., 90%. The program should conduct each experiment 10 times. Also, have the code determine the average percent burned for each probability. Plot the data and fit a curve to the data. Discuss the results.

11. Develop a fire simulation in which a tree once ignited or hit by lightning in one time step takes five additional time steps to burn. The fire can spread from the burning tree to a neighboring tree with different probabilities only on the second, third, and fourth time steps after catching fire. Assume a tree's fire is hottest the third time step after ignition.

12. Develop a fire simulation with accompanying animation in which a section of the forest is damper and, hence, harder to burn. Discuss the results.

## Answers to Quick Review Question

From the text's website, download your computational tool's *10_3QRQ.pdf* file for answers to these system-dependent questions.

## References

DAS (Dynamic Animation Systems, Inc.) 2012. U.S. Forest Service. http://www.d-a-s.com/node/36 (accessed December 26, 2012)

Dossel, B., and F. Schwabl. 1994. "Formation of Space-Time Structure in a Forest-Fire Model." *Physica Abstracts*, 204: 212–229.

Gaylord, Richard J., and Kazume Nishidate. 1996. "Contagion in Excitable Media." *Modeling Nature: Cellular Automata Simulations with Mathematica*. New York: TELOS/Springer-Verlag, pp. 155–171.

Los Angeles Times. 2010. "Two Men Who Started Malibu Corral Canyon Fire Sentenced to a Year in Jail." *LA Now*, September 9. http://latimesblogs.latimes.com/lanow/2010/09/two-men-who-started-malibu-corral-canyon-fire-sentenced-to-a-year-in-jail.html (accessed December 22, 2012)

NPS (National Park Service). 2012. "Fire Ecology." Yellowstone National Park. http://www.nps.gov/yell/parkmgmt/fireecology.htm (accessed December 22, 2012)

SBCCBS (Santa Barbara City College Biological Sciences). 2010. "Fire in the Chaparral." Biology 100 Concepts of Biology: Introduction to the Chaparral. http://www.biosbcc.net/b100plant/htm/fire.htm (accessed December 22, 2012)

Shodor Education Foundation. "Fire!!" The Shodor Education Foundation, Inc., 1997–2003. http://www.shodor.org/interactivate/activities/Fire/ (accessed December 22, 2012)

———. "A Better Fire!!" The Shodor Education Foundation, Inc., 1997–2003. http://www.shodor.org/interactivate/activities/ABetterFire/ (accessed December 22, 2012)

Wilson, Tracy, Stuart Pfeifer, and Mitchell Landsberg. 2003. "California Fires Threaten 30,000 More Homes." *Pittsburg Post-Gazette*, October 28. http://www.post-gazette.com/stories/news/us/california-fires-threaten-30000-more-homes-520702/ (accessed December 22, 2012)