

# Programming (ERIM)

## 4. Exercise

Deadline for submission: 2013-11-25 23:59 CET

### Introduction: forest fire simulation

The Portuguese government is worried about the path of destruction caused yearly by forest fires in the Alentejo region (<http://en.wikipedia.org/wiki/Alentejo>). Global warming appears to be taking its toll and long dry summer periods render the forests of Alentejo extremely vulnerable to spontaneous combustion and to random people who just like to torch a beautiful piece of nature. In order to be able to effectively fight the fires, the government is obliged by law to declare a state of emergency when 10% of the region is currently burning. The Portuguese government would like you to give them a reliable estimation of the number of days they typically have until they have to declare a state of emergency.

In order to tackle this problem, we first simplify the problem by making several assumptions. First, we assume the region to be completely rectangular (which is a reasonable assumption taking into account the shape of Alentejo). Second, we model the outbreaks of forest fires as a stochastic process and assume the time between two outbreaks to be exponentially distributed. The location of an outbreak is drawn from a uniform distribution defined within the rectangular area. If the location of an outbreak turns out to be already burning, the new outbreak will simply be ignored in our model. Third, we assume that the forest fires grow in a circular way with a steady pace. Fourth, we assume that the individual fires move around in a random motion due to wind and differing amounts of burning material available.

As our assumptions render the number of days until declaration of a state of emergency a stochastic variable, simulation of the outbreaks and growth of forest fires over time is a useful technique for addressing the request made by the Portuguese government. As different simulations may yield different outcomes due to the randomness in our model, running multiple simulations provides a more reliable approximation of the number of days the Portuguese government typically has until they have to declare a state of emergency.

In this exercise, you will be simulating forest fires in accordance with the model and assumptions presented above. The main procedure for the simulation is detailed as pseudocode in Algorithm 1.

### Exercise

1. Write a function that simulates the outbreak, growth and motion of a series of forest fires over time until a state of emergency is declared.
  - (a) The inputs of this function should be:
    - The width in kilometers of the considered area.
    - The length in kilometers of the considered area.
    - The percentage of area required to be currently burning for declaring a state of emergency.

---

**Algorithm 1:** Pseudocode for the forest fire simulation.

---

```
initialize day
initialize map
determine starting day for next fire
while burned area(map)  $\leq$  threshold do
    move fires
    spread fires
    if today is the starting day for next fire then
        determine coordinates of center
        if coordinates not burned yet then
            start fire
        end
        determine starting day for next fire
    end
    update burned area
    visualize forest
    proceed to next day
end
```

---

- The speed (in km/d) of increase of the individual fire radius.
  - The speed (in km/d) with which the fires move around.
  - The mean number of days until the outbreak of a new forest fire (exponentially distributed).
- (b) This function should produce as output the number of days after which a state of emergency is declared.
- (c) The contents of this function should reflect the pseudocode presented above. The function should make use of two other functions, i.e., one function for checking whether pairs of coordinates have been burned and another function for visualizing the simulation. We will implement these functions in the next steps.

**Hint:** Consider keeping track of all forest fires by storing them in a matrix with columns representing the coordinates of their centers and the radii.

**Hint:** For moving the fires, you can sample the direction (in degrees) and transform the move speed as in [http://en.wikipedia.org/wiki/Cartesian\\_coordinate\\_system#Rotation](http://en.wikipedia.org/wiki/Cartesian_coordinate_system#Rotation).

2. Create a function that checks whether sets of coordinates have been burned by any of the current fires.
- (a) Depending on how the forest fires are stored in your application, this function should at least take as input:
- A matrix with all sets of coordinates to be checked.
  - The matrix with information on all current fires, including the coordinates of their centers.
- (b) The output of this function should be a vector, signalling for each set of coordinates whether this point has been burned (*TRUE*) or not (*FALSE*).
- (c) This function should first compute the Euclidean distances between the requested locations and all current forest fires. Then, if the distance of a location to any fire is smaller than or equal to the fire's radius, the location can be considered to have been burned.

**Hint:** given an  $n \times p$  matrix  $\mathbf{A}$  with  $n$  coordinates in  $p$ -dimensional space and an  $m \times p$  matrix  $\mathbf{B}$  with  $m$   $p$ -dimensional coordinates, the  $n \times m$  matrix of their Euclidean distances  $\mathbf{D}$  can be computed as

$$\mathbf{D} = \sqrt{\mathbf{x}\mathbf{1}_m + \mathbf{1}_n\mathbf{y}' - 2\mathbf{A}\mathbf{B}'}, \quad (1)$$

with  $\mathbf{x}$  an  $n \times 1$  vector where the elements represent the sum of squared coordinates of  $\mathbf{A}$ ,  $\mathbf{y}$  an  $m \times 1$  vector with the sum of squared coordinates of  $\mathbf{B}$ ,  $\mathbf{1}_n$  an  $n \times 1$  vector of ones, and  $\mathbf{1}_m$  a  $1 \times m$  vector of ones.

**Hint:** you may find the function `any` useful.

- (d) Use this function in the main simulation in order to check whether a new location of a forest fire has already been burned as well as to estimate the burned percentage of the complete considered area.

**Hint:** only check for every squared kilometer whether it has been burned or not. You may want to use the MATLAB function `meshgrid` or R function `expand.grid` to generate all coordinate pairs at this level of granularity.

3. Create a function that visualizes the forest fires over time.

- (a) The visualization may look like the snapshot presented in Figure 1.

**Hint:** you can plot circles in MATLAB by using the `rectangle` function (as logical as having to press Start to shut down Windows). In R plotting circles is the easiest through function `draw.circle` from the `plotrix` library (`install.packages('plotrix');` `library(plotrix)`).

**Hint:** prevent your animation from slowing down by clearing the figure before plotting an iteration. You can use the MATLAB command `cla` for this.

**Hint:** in order to be able to actually see your animation in MATLAB, you may need to briefly `pause` the animation after plotting an iteration.

4. Write a script that runs 50 simulations of forest fires in an area of 130 by 240 kilometers. Suppose that the time in between the outbreak of two subsequent fires is on average 8 days (exponentially distributed), fires typically spread with a speed of 0.4 kilometers per day, and they move 0.85 kilometers per day. How many days does it take on average until the government has to declare a state of emergency after 10% of Alentejo is burning? Plot a histogram with the results of your simulation. You may want to change your simulation function by adding a parameter with which you can toggle the visualization on and off.

**Inferno day 89 – 10.1% of Alentejo is burning**

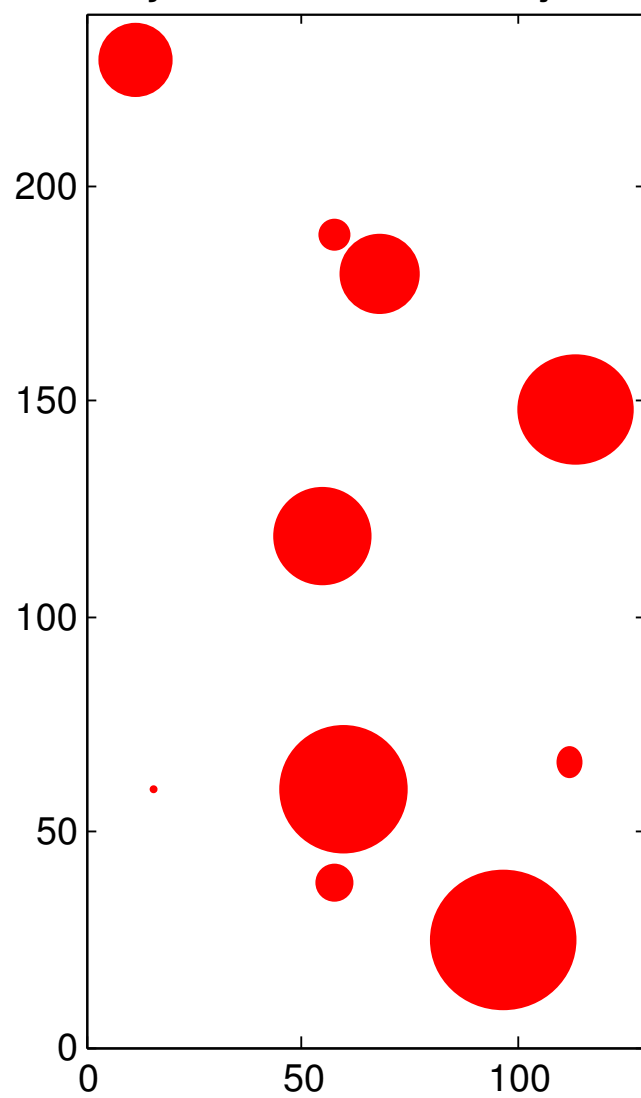


Figure 1: Example visualization of the forest fire simulation.