

# Práctica 6: Sistema Multiagente

Juan Pablo Rosas Baldazo

19 de septiembre de 2017

## 1. Tarea

Modificar el experimento para poder ejecutarlo en forma paralela. Se utilizó la librería *parallel* y se ejecutó en una laptop Lenovo Procesador: Core i5, Memoria RAM: 8Gb, Núcleos: 4.

Para poder realizar la tarea se creo una función de contagio que sería la parte que se ejecutará de forma paralela.

La función se encarga de cambiar el estado de un agente de *susceptible* a *contagiado* en base a probabilidades preestablecidas y de que tan lejos se encuentra un agente susceptible de un agente infectado y también cambiara el estado de *contagiado* a *recuperado* en base a una probabilidad de recuperación (*pr*).

Se modificó también el criterio de parada para que se detuviera cuando ya no quedaran agentes infectados, es decir que todos tengan estado de *recuperado* o *susceptible*.

---

**Algoritmo 1** Contagio de agentes

---

**Require:**

```
1: agentes: Matriz con la posición y el estado de cada agente.
2: r: Umbral de infección.
3: n: Número de agentes.
4: pr: Probabilidad de recuperación.
5: procedure CONTAGIA(i)
6:   a1  $\leftarrow$  agentes[i,]
7:   Estado  $\leftarrow$  'S'
8:   if a1$estado = 'S' then                                 $\triangleright$  Si el agente es susceptible
9:     for j in 1 : n do
10:      a2  $\leftarrow$  agentes[j,]
11:      if a2$estado = 'I' then                                 $\triangleright$  Buscamos algún agente infectado
12:        dx  $\leftarrow$  a1$x - a2$x
13:        dy  $\leftarrow$  a1$y - a2$y
14:        d  $\leftarrow$   $\sqrt{dx^2 + dy^2}$ 
15:        if d < r then                                         $\triangleright$  Si el infectado esta cerca
16:          p  $\leftarrow$   $\frac{r-d}{r}$                                  $\triangleright$  Obtenemos la probabilidad de contagio
17:          if Número aleatorio < p then
18:            Estado  $\leftarrow$  'I'                                 $\triangleright$  Si el agente se contagia
19:            break
20:          end if
21:        end if
22:      end if
23:    end for
24:  else if (a1$estado = 'I') then                                 $\triangleright$  Si el agente esta infectado
25:    if (Número aleatorio < pr) then
26:      Estado  $\leftarrow$  'R'
27:    else Estado  $\leftarrow$  'I'
28:    end if
29:  else if (a1$estado = 'R') then
30:    Estado  $\leftarrow$  'R'                                 $\triangleright$  Si el agente esta recuperado
31:  end if
32:  return Estado
33: end procedure
```

---

### 1.1. Resultados

Los tiempos de computo son muy diferentes, mientras que en el código sin paralelismo obtenemos tiempos menores a 25 segundos, en el código con paralelismo obtenemos tiempos mas grandes superando los 200 segundos.

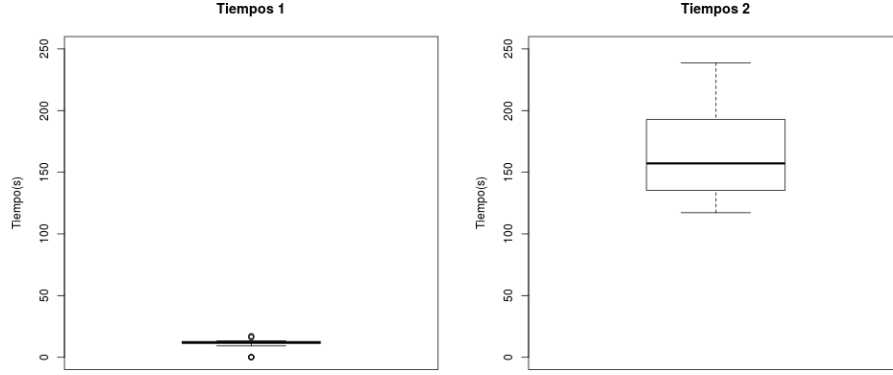


Figura 1: Comparativa de tiempos de computo, el primer plot corresponde al código original sin paralelizar y el segundo al código ya paralelizado

## 2. Reto 1

El primer reto consiste en que al momento de iniciar el experimento existan algunos agentes que fueron vacunados y no es posible contagiarse.

### 2.1. Descripción del experimento

Se modifico la parte del código de la creación de los agentes agregando una parte para que aparecieran como recuperados desde un inicio y ver como se comportaba la evolución de la infección.

Se usaron las siguientes probabilidades de que se crearan agentes vacunados desde un inicio  $pv = 0,05, 0,1, 0,15, 0,2, 0,25, 0,3, 0,35$ .

---

**Algoritmo 2** Iniciar agentes vacunados.

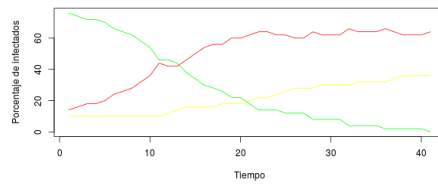
---

**Require:**

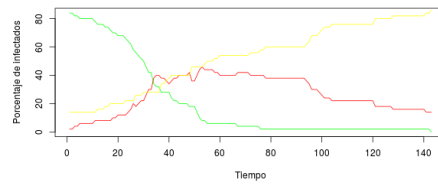
- 1:  $pv$ : Probabilidad de que sea vacunado.
  - 2: **for**  $i$  in  $1 : n$  **do**
  - 3:    $e \leftarrow 'S'$
  - 4:   **if** Número aleatorio  $< p_i$  **then**
  - 5:      $e \leftarrow 'I'$
  - 6:   **else if** Número aleatorio  $< pv$  **then**
  - 7:      $p \leftarrow 'R'$
  - 8:   **end if**
  - 9:   agentes  $\leftarrow$  asigna el estado inicial del agente además de la posición y dirección
  - 10: **end for**
-

## 2.2. Resultados

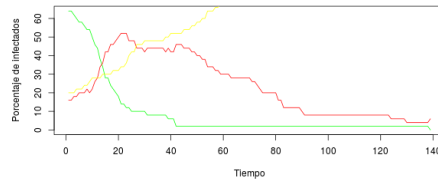
Como se puede observar en las figuras conforme aumentamos la probabilidad de que aparezcan agentes vacunados, el valor máximo de la epidemia no se ve muy afectado es decir no hay gran variación, sin embargo la propagación se vuelve mas lenta esto debido a que la cantidad de agentes susceptibles es menor.



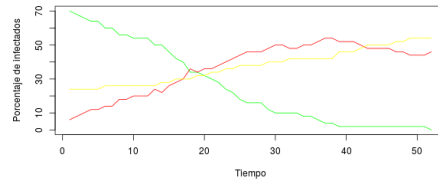
(a) 5 % de probabilidad de nacer vacunado



(b) 15 % de probabilidad de nacer vacunado



(c) 25 % de probabilidad de nacer vacunado



(d) 35 % de probabilidad de nacer vacunado

Figura 2: Evolución de la epidemia al variar la probabilidad de aparición de agentes de vacunados que aparecen desde el inicio.