Société Générale

Program Report

11/18/17

Introduction of the problem

A collection of particles is contained in a linear chamber. They all have the same speed, but some are headed toward the right and others are headed toward the left. These particles can pass through each other without disturbing the motion of the particles, so all the particles will leave the chamber relatively quickly.

Key points for problem solving

Firstly, I try to define the main parameters of this problem, in order to know the best way to implement the solution. Thus, the two first parameters are the ones given by the user:

1) The first one is the speed of particles, which will be the number of positions each particle moves in one time unit.

2) The second one is the Initial configuration of the linear chamber which give us several information:

The chamber **length**: number of possible positions for particles

The Different locations:

".": Empty location

"R": Rightward particle

"L": Leftward particle

3) At each unit of time, we want a string showing occupied and non-occupied locations, so we want every **chamber state** during the processing. So I introduced the variable **timeUnit** which will be incremented for every new processing, and so every new chamber state. And finally the **chamberState** which is a List of String with current chamber configuration.

4) All those explanations, to introduce the key parameter of the total distance travelled by a particle which corresponds to total particle movement since the beginning of the simulation in term of number of locations.

For example, if the speed is 3, and we are at the 4th time unit:

The total distance travelled by a particle will be: **totalDistance** = 3*4 = 12 locations

Explanation of the process

In order to complete each new state of the chamber, I browse throw the Initial configuration of the chamber, regarding the current **totalDistance**, and taking into account several specific cases. And while the chamber isn't empty, I continue the process to reach the final state with an empty chamber.

First and above, the easiest case is when the speed exceeded the chamber's length because automatically, the chamber will be empty at the next time unit.

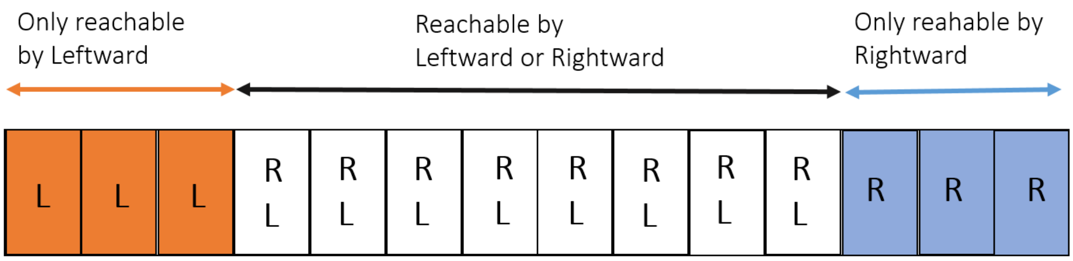
```
#If speed >= chamber_lenght, the next state of the chamber will be the final state
if (totalDistance>=lenght):
    chamberState = finalChamberState
    affiche_to_str(chamberState)
```

Then we have to process the several situations separately.

To explain it, we can introduce: **<u>the chamber configuration after the 3rd time unit with initial speed = 1.</u>**

- The total distance travelled by a particle (**totalDistance**) will be 3, which immediately allow us to know that some positions are no longer reachable by every particles.
- **A)** In fact regarding this **totalDistance**, the first three positions starting from the left, are only reachable by Leftward Particle.
- **B)** In addition following the same reasoning, the three ending position are only reachable by rightward particle.
- **C)** And the other positions are reachable by either rightward or leftward.

<u>Possible positions of particles after 3 time units with speedvalue of 1 location</u>



A)

```
#Critical case, Position only reachable by Leftward particles
elif(i<totalDistance):
    if(init[i+totalDistance]=="L"):
        chamberState[i] = "X"
    else:
        chamberState[i]="."
```

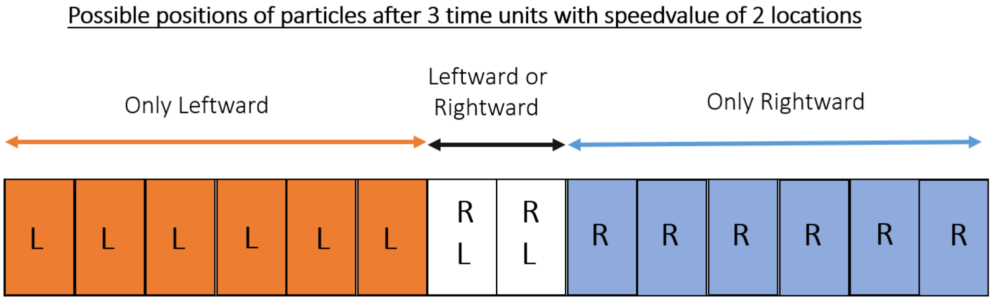
B)

```
#Critical case, Position only reachable by Rightward particles
elif(i>=lenght-totalDistance):
    if(init[i-totalDistance]=="R"):
        chamberState[i]="X"
    else:
        chamberState[i]="."
```

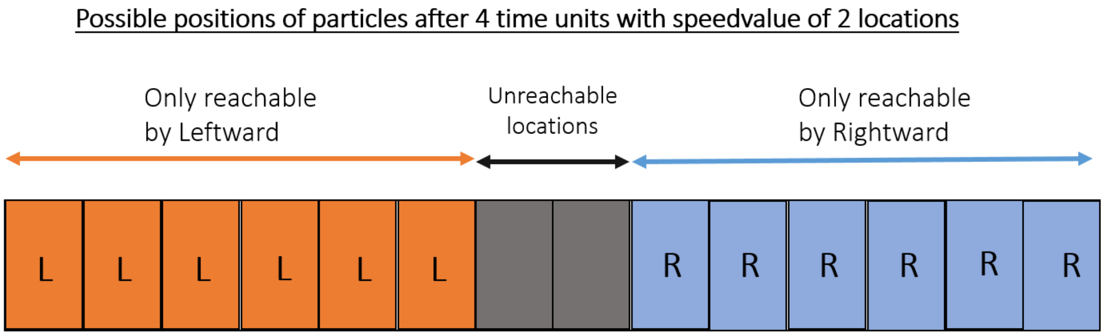
C)

```
#Casual case, Positions possibly reachable by Rightward or Leftward particles
else:
    if(init[i+totalDistance]=="L" or init[i-totalDistance]=="R"):
        chamberState[i]="X"
    else:
        chamberState[i]="."
```

To complete, if we take a speed value of 2, and we want to define the chamber configuration after the 3rd time unit (**totalDistance=6**), we will reach the following conclusions.



Finally, the most critical situation that we must take into account, is the case when unreachable positions appeared. This happened when the total movement of a particle exceeded a half of chamber length following next example.




```
#Most Critical case, Positions no longer reachable by a particule, regarding Distance T
if (i>=lenght-totalDistance and i <totalDistance):
    chamberState[i]="."
```

Thus, the division of these distinct cases allows me to reduce time processing and the number of variables.