SIR Model
Cellular Automata*

Monit Sharma†
Indian Institute of Science Education And Research Mohali

(Term Paper)
(Dated: February 7, 2021)

A stochastic Cellular Automata (CA) model is used for the simulation of the **S**usceptible-**I**nfected-**R**emovals population over space and time. Two initial grid configurations are used to compare and contrast the system spatio-temporal dynamics: random, and centre. The simulations show that random configuration infect more of the population, but quickly dissipating through space. The centre case slowly propagates through space and infects less of the people, while the patchy configuration shows to be a middle case between random and centre.

I. BASICS AND ASSUMPTIONS

Susceptible-infective-removals (SIR) is the classic mathematical model for studying infectious diseases in epidemiology. To model and simulate epidemics of contagious disease, we use cellular automata. The basic assumptions of SIR and naive CA limit their applicability to real-world characteristics. A global stochastic cellular automaton (GSCA) is proposed, incorporating geographic and demographic-based interactions. The interaction between the cells is a function of population density and the Euclidean distance and has been extended to include geographic constraints. The progression of diseases using traditional CA and classic SIR is analyzed. Similar behaviour to the SIR model is exhibited by GSCA, using the geographic information systems (GIS) gravity model for interactions. The GSCA model addresses the limitations of the SIR and naive CA models of a homogeneous population with uniform mixing. The GSCA model is oriented to the heterogeneous population and can incorporate interactions based on geography, demography, environment and migration patterns. The progression of diseases can be modelled at higher fidelity levels using the Global Cellular Automata.

The basic assumptions in the model are:

1. After some time infected people will be cured and will get immune to the disease
2. Birth rate and Death rate is zero, as the disease doesn't last long as much as to change the population, so the total population remains constant.

Susceptible(S) have no immunity from the disease. Infected(I) have the condition and can spread it to others. Recovered (R) have recovered from the illness and are immune to further infection. β is the infection rate which tells how the infection spread through infected people to susceptible. α is recover rate which means how people recover from illness. For better understanding we take

$$s = \frac{S}{N}, i = \frac{I}{N}, r = \frac{R}{N}$$

II. INTRODUCTION

Landscape epidemiology studies the disease patterns that arise from abiotic and biotic conditions. In this paper, the objective is to simulate the dynamics of an infectious disease propagation on various landscapes using a stochastic cellular automaton(CA) susceptible- infected-removed (SIR) model.

III. BACKGROUND

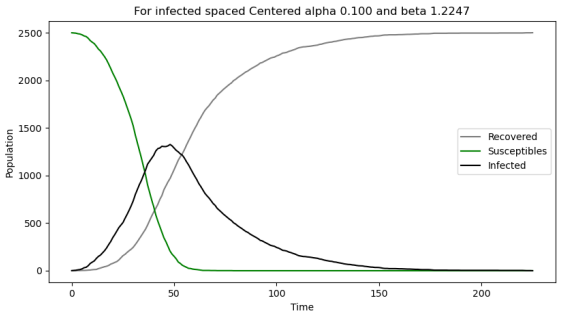
Cellular Automaton applied to grid-based modelling is a method to model disease propagation over space and time. CA models provide rules that are biologically motivated and easily programmable. In this approach, a grid array of cells represents a landscape. Each cell contains an embedded mini-model composed of state variables describing its condition, a means of communicating with surrounding cells (neighbourhood), and rules dictating the cell's response to its state and communications from its neighbours through a series of time-steps. The imposition of relatively simple rules can generate problematic emergent behaviours as the landscape evolves through time.

IV. DYNAMIC SYSTEM

We can extract the Cellular Automaton rules from the classical SIR models' ideas based on differential equations. The set of ordinary differential equations corresponding to the CA model are:

$$\frac{dS}{dt} = -\frac{\beta SI}{N}$$

*
† MS17135Physics Major, IISER Mohali.; ms17135@iisermohali.ac.in



$$\frac{dI}{dt} = \frac{\beta SI}{N} - \alpha I$$

$$\frac{dR}{dt} = \alpha I$$

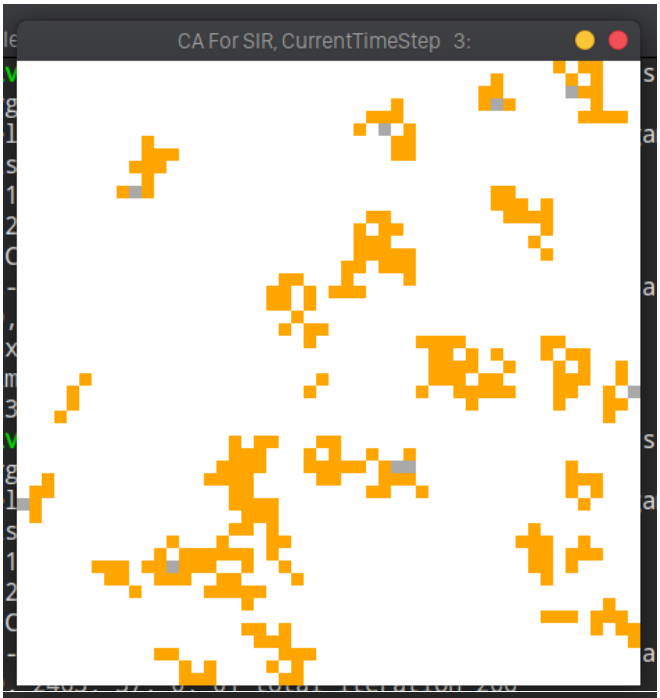
furthermore, $\frac{dI}{dt} = I\alpha(\frac{\beta S}{\alpha N} - 1)$, so the term R_o came into practice. $R_o = \frac{\beta S}{\alpha}$ defines the shape of the curve S and I.

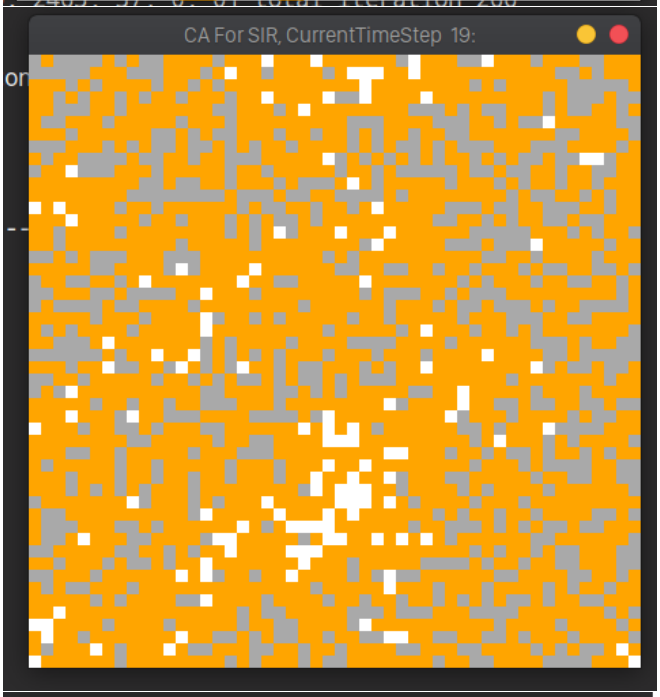
The system is then divided into three groups, where each cell represents an individual that can be in one of three states: S, when the individual is susceptible to infection by neighbours; I, when the individual is infected and can transmit the disease for neighbouring susceptible cells; and R, when the individual is recovered. Below Figure shows the ODE system plotted against time.

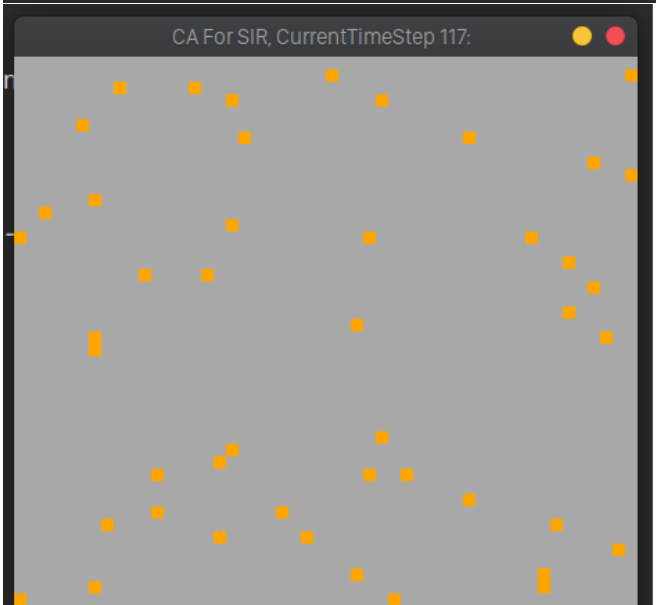
For simple simulation we choose $\beta = 1.2247$ and $\alpha = 0.1$ and N = 2500 and S = 2499 and I = 1. So $R_o = 1$ so here Shape of I curve is first it will increase then attain certain maxima and then decrease.

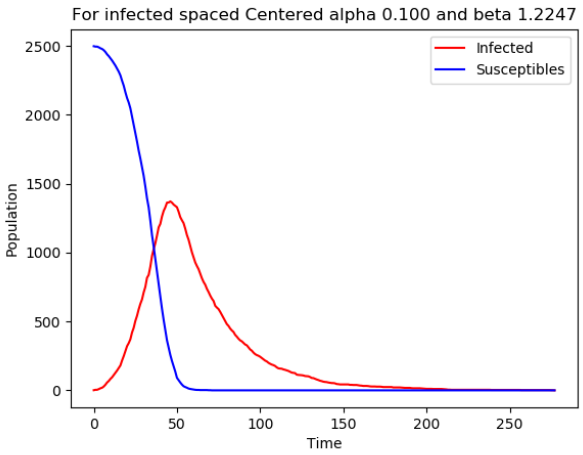
V. METHOD

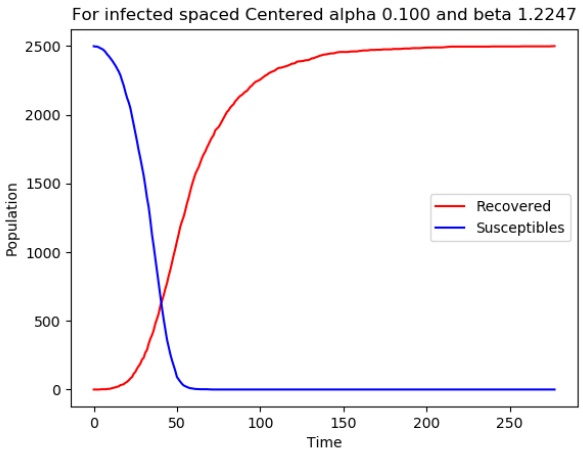
Scientific python was used to visualize the CA model depiction of the spatial disease propagation.For Python Code click here.The disease will propagate through the landscape based on a set of probabilities of state transitions. At each time step, there is a probability of an **S** cell becoming infected according to their set probabilities. Likewise, each **I** cell can become recovered based on probability P_c or parameter from the previous ODE model. The spatial and temporal dynamics were examined with simulations of various initial landscape population configurations, i.e. random, center, and patchy. The grid size used was 450x450 with 100% cell are susceptible, and the remaining one cell is infected. In first Center simulation we set the probability to S become infected when its surrounded with infected is according to uniform random variable $\in$ (0, 1) if generated random sample is less than β then we set **S** to **I**. White pixel states susceptible, Orange pixel for Infected and Dark grey pixel for Recovered species. Below Three figures for infected spaced randomly, First snap the initial stage, second figure is when infected reached maximum and last the figure is when almost no infected remaining.

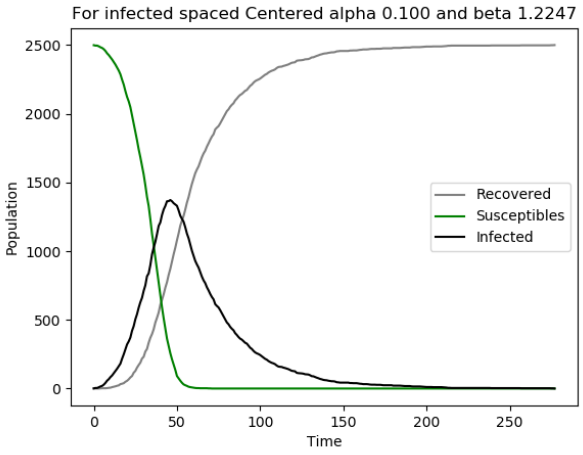












VI. RESULT

Each initial case (random, centre, and patchy) were initiated and simulated over 300-time steps and aver- aged over ten runs. Below Figures show an example snapshot of each case after ten-time steps. Here, it is clear that the

```
''' Print the current generation '''
def printGenerationUniverse(universeList, currentTimeStep, cellCountX, cellCountY, normalCharacter,
                            susceptibleCharacter, infectedCharacter, recoveredCharacter):
    logging.info("TimeStep %3i:  " %currentTimeStep)
    rowLabel = "  "
    for l in range(cellCountX):
        rowLabel += str(l) +" "
    logging.info(rowLabel)
    for currentRow in range(cellCountY):
        logging.info ("%s %s" %(currentRow, universeList[currentRow].replace('0',normalCharacter+" ")
                .replace('1', susceptibleCharacter + " ")
                .replace('2', infectedCharacter + " ").replace('3', recoveredCharacter + " "))
               )
    return
```

```
''' This method calculates the new state of the cell based on Van Neumann neighborhood '''
def getNewState2D(currentRowNeighbours, upperRowNeighbours, lowerRowNeighbours,beta,gamma):
    newState = '1'

    leftCharacter = currentRowNeighbours[0]
    selfCharacter = currentRowNeighbours[1]
    rightCharacter = currentRowNeighbours[2]

    upperLeftCharacter = upperRowNeighbours[0]
    upperCenterCharacter = upperRowNeighbours[1]
    upperRightCharacter = upperRowNeighbours[2]

    lowerLeftCharacter = lowerRowNeighbours[0]
    lowerCenterCharacter = lowerRowNeighbours[1]
    lowerRightCharacter = lowerRowNeighbours[2]

    newState = selfCharacter

    if selfCharacter == '1': # If Normal and there is an Infected close, be Susceptible
        if leftCharacter == '2' or rightCharacter == '2' or\
        upperLeftCharacter == '2' or\
        upperRightCharacter == '2' or\
        upperCenterCharacter == '2'or\
        lowerLeftCharacter == '2' or\
        lowerRightCharacter == '2' or lowerCenterCharacter == '2':
            #betaChance =np.random.uniform(beta-(beta/2),beta+(beta/2))
            #betaChance=np.random.uniform()
            betaChance= (2 - np.random.uniform()) # UNIFORM
            if betaChance > 0 and betaChance < beta:
                newState = '2'
    elif selfCharacter == '2': # if Infected, calculate the probability to be Recovered 'to recover'
        gammaChance = (1 - np.random.normal(0.5, 1.0)) # NORMAL
        #gammaChance = (1 - np.random.uniform()) # UNIFORM
        #gammaChance=np.random.uniform()
        #betaChance = (2 - (np.random.poisson(2) % 10) * 0.1) # POISSON
        if gammaChance < gamma and gammaChance > 0:
            newState = '3'
    return newState
```

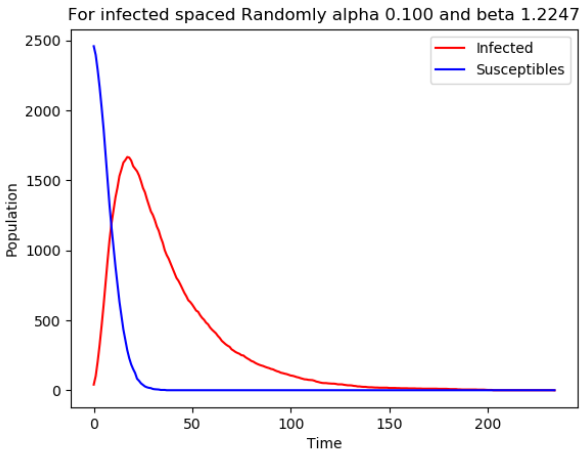
random chance infects the population more quickly than the centre or patchy patients. The table shows the averaged maximum infected populations and time steps with their corresponding standard deviations. In combination with below

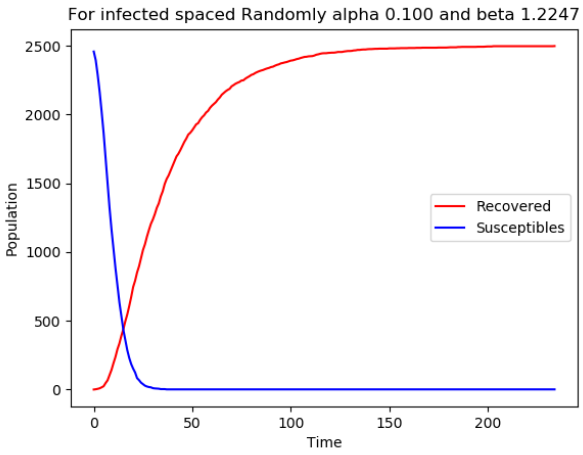
Figures show the the example runs S, I, and R populations plotted against time; we see that the centre case takes the longest time to propagate through the population while infecting the least amount of individuals. The random chance infects most individuals, but relatively quickly. fits somewhere in between the random and centre cases.

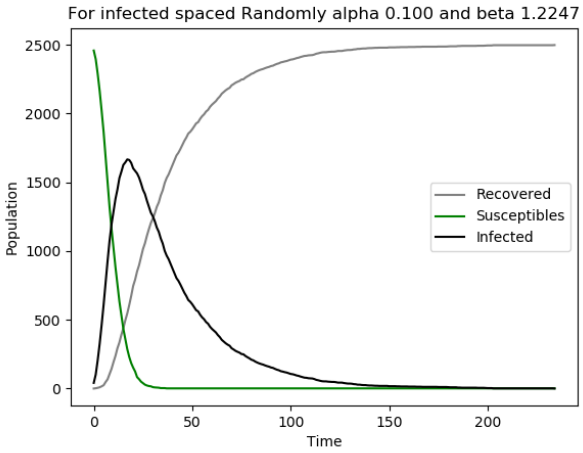
From figures we can show that in random case infection spreads more quickly compare to centered case this is because of R_o.In random case our $R_o = \frac{\beta S_o}{\alpha}$

changed. In random case initial average 32-35 people infected initial stage where in spaced centered case 1 people infected. So R_o is less compare to second case since we know lesser the R_o more the epidemic spread (More no of people infected/infection spread rapidly).

This model can be extended by including environmental layers, thus incorporating actual landscape barriers into the rules. It can also be improved to consider long-range interactions between CA cells that would incorporate population dynamics.





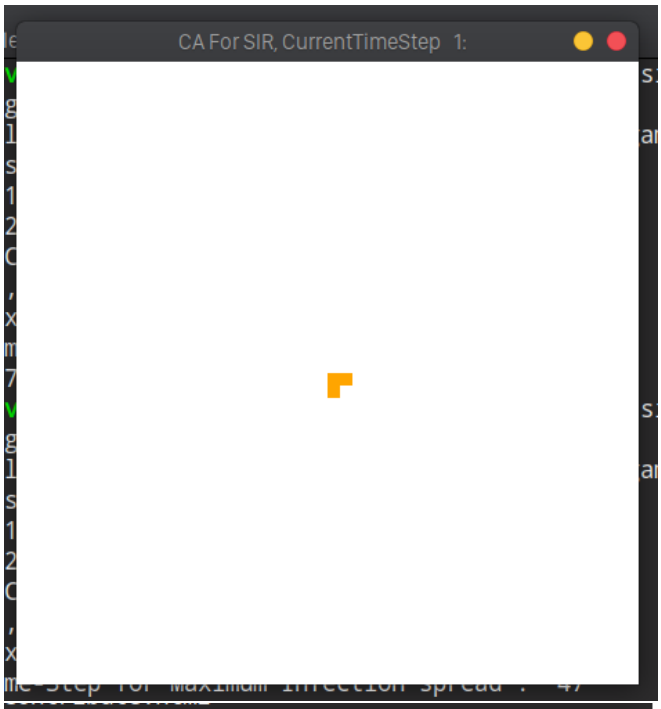


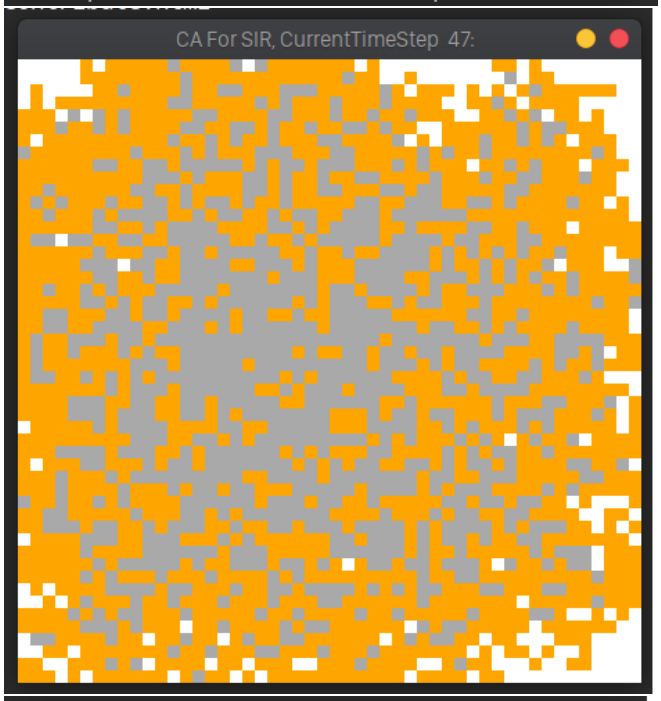
VII. CONCLUSION

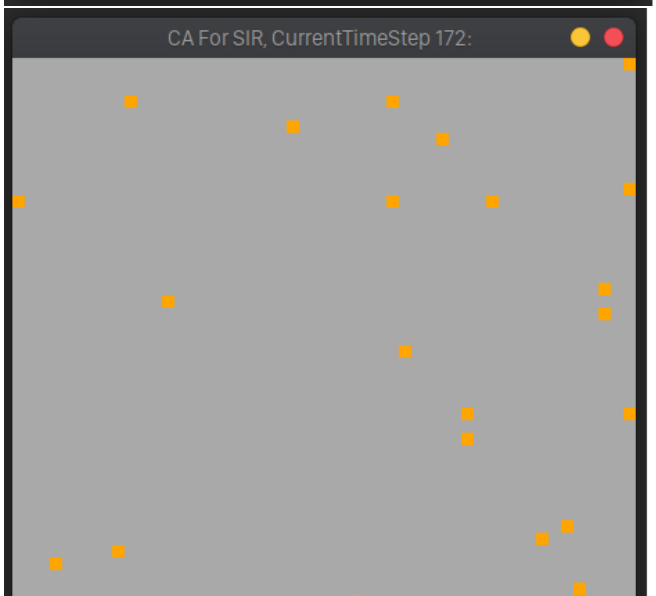
In this work, a theoretical model to simulate the spreading of a plague is introduced. It is supported by the utilization of two-dimensional cellular automata endowed with an appropriate local transition function. The whole population is classified into three categories: susceptible, infected and recovered individuals. Consequently, the proposed model is often considered as a SIR-type model. Its main features are the following: the entire amount of population within the cellular space is constant. Nevertheless, it cannot be uniformly distributed between the cells. The local transition function is straightforward, and several epidemiological and environmental parameters are involved. The vaccination effect is considered. This model's main characteristic is the definition of each cell's state as a three-up let formed by a suitable discretization portion of its population which is susceptible, infected and recovered at whenever step, alongside the definition of the local transition function involving these parameters. The simulations obtained using artificially chosen parameters seem to accept as accurate with an actual epidemic's expected behaviour. The worked model can be used as a basis for developing another algorithm to simulate real epidemics—consequently, further work aimed at testing its performance against real data. Obviously, in real simulations, one has to take care of the scale, and an appropriate size of the cells must be used to obtain an efficient simulation. The proposed model do have some shortcomings, e.g., the small-world effect which is crucial to model SARS, Foot and mouth disease and Aviation Flu, and seasonality effects which are crucial to model measles.
You can find the source code for the simulation here.

VIII. REFERENCES

- G.Schneckenreither,N.Popper,G.Zauner, F.Breitenecker
  Modelling SIR-type epidemics by ODEs, PDEs, difference equations and cellular automata
- Damien Regnault, Nicolas Schabanel, Éric Thierry
  On the Analysis of "Simple" 2D Stochastic Cellular Automata
- M.A.Fuentes, M.N.Kuperman
  Cellular automata and epidemiological models with spatial dependence
- Hoya White, A. Martíndel Rey, G. Rodríguez Sánchez
  Modeling epidemics using cellular automata


CA For SIR, CurrentTimeStep 1:


CA For SIR, CurrentTimeStep 47:


CA For SIR, CurrentTimeStep 172: