Modelling and Simulation of Smog over North West India due to Stubble Burning using CELL DEVS Cellular Automata Model in Lopez Simulator

Ekambir Sidhu (Student ID: 300036992) *Manohar Deep Singh Gill* (Student ID: 101055138)
esidh097@uottawa.ca manohargill@cmail.carleton.ca

Prof. Gabriel A. Wainer
Department of Systems and Computer Engineering
Carleton University, Ottawa, ON
gabriel.wainer@sce.carleton.ca

***Abstract* - In this term paper, the CELL DEVS cellular automata model for the simulation and modeling of smog formation over North West India due to stubble burning of crops has been implemented and proposed. The CELL DEVS formalism approach using the state variables has been employed in Lopez Simulator for the implementation of the proposed models for smog modelling. The term paper clearly illustrates through the various test model implementations and simulations that the smog formation in the North-West India is because of extensive stubble burning by the farmers. The smog has been major problem in north-west part of India in the past few years and the smog formation has peaked in November 2017 leading to the least air quality index. The proposed model also takes into account the effect of south-west monsoon on the crop sowing pattern through the formation of zones in Lopez Simulator along with the use of state variables i.e. the crops are sown first in the south-east and north-east India followed by the crop sowing in the south-west and north-west India because the monsoon arrives earlier in the southern part of India. It is to be noted that the random crop sowing pattern in the simulation follows a normal or gaussian distribution with normalized mean of 0.4 and standard deviation of 0.3 in all models. The various phases of crop growth in farming – crop sowing, reproductive phase and ripening phase followed by crop harvesting have also been taken into account and implemented. The total simulation time for the various test models have been taken into account, compared and analyzed.**

Keywords – Cell DEVS formalism, crop-sowing, Lopez Simulator, monsoon, normal distribution, smog, state variables, , ploughing, reproductive phase, ripening phase, stubble burning

I. Introduction

Smog is a visible air pollution which comprises of nitrogen oxide (NO), sulphur oxide (SO_2), ozone (O_3) and smoke, arising from coal emissions, vehicular emissions, forest and agriculture fires [1]. Smog is a severe problem in New Delhi and the adjoining areas in the National Territory of India. The smog pollution peaked in November 2017 and it has reported that the air quality was worst since 1999 [2]. The 'Great Smog' led to cancellation and delay of public transport, major accidents because of decreased visibility and major health problems such as asthma and bronchitis [2][3]. The major source of smoke which led to the formation of smog in India is stubble burning of crops in the North-West part of India [2].

The air quality can be measured by the amount of PM 2.5 and PM 10 particulates suspended in air. On November 7, 2017, the PM 2.5 levels in Delhi shot up to 999 which was much above the recommended 60 micrograms. At the same time, the PM 10 shot to 999 (the maximum level for the monitors), instead of the recommended limit of 100 [2]. It was because of the increased smog pollution that health emergency was declared in the capital by the Central Government of India in order to cope with the extrusive amount of polluted air [2].

The smog formed in November 2017 over the north-west part of India was so dense and thick that it became visible through the satellite images. The NASA weather forecasting satellites and the NASA International Space Station (ISS) has photographed the satellite images over India which shows the smog in the form of dense white clouds over the north-west India. The smog was so thick that the ground, green vegetation and farms were completely invisible in the satellite images. The Figure 1 below shows the smog over the north-west India in the form of white dense clouds as photographed by NASA on 25th November, 2017 [3].

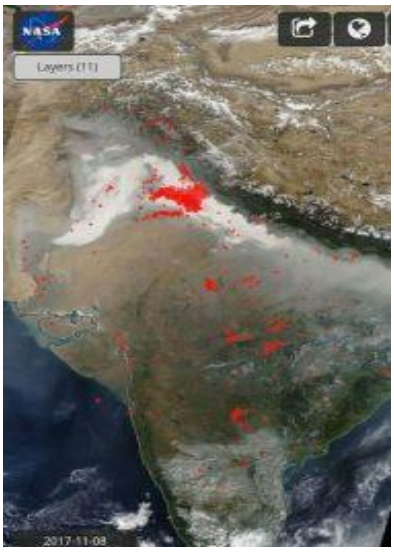


Figure 1 – Satellite image photographed by NASA on 25th November showing the thick dense smog clouds over north-west India [3]

The major reason of smog formation in India in the month of October-November over years is because of the stubble burning of crops by farmers in north-west India (especially in Punjab and Haryana states). After harvesting the crops, the farmers prefer to burn the stubble and chaff instead of ploughing it. The ploughing of left over stubble after crop harvesting is common in south-west, north-east and south-east parts of India and thus, there is no smoke and no smog over the south west and south-east India as shown in Figure 1 [4].

This smoke and fumes arising from the agricultural fires (also known as stable burning) react in the atmosphere with sunlight to form thick clouds of secondary pollutants (nitrogen oxide, sulphur oxide and ozone) as shown by satellites images captured by NASA weather forecasting satellite in figure 2 below [4]. The figure 2 clearly shows the major stubble burning areas in Punjab and Haryana states of India shown by red dots on the map and smog exists over the north-west part of India because of the stubble burning in those parts of India.

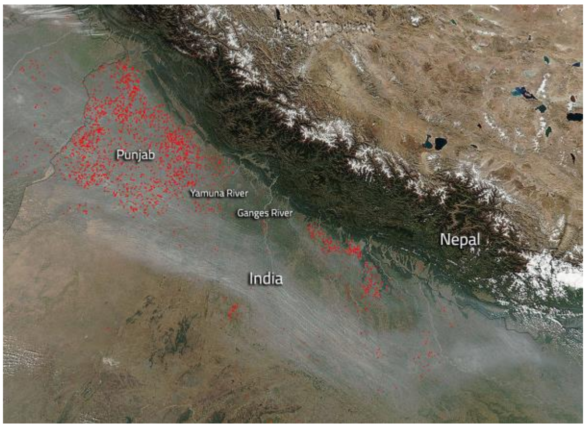


Figure 2 – NASA captured image on 28th December showing the smog over north-west India with the major stubble burning areas and zones shown by red dots [4]

This term paper focus on the simulation and modelling of smog over the India because of stubble burning of crops. The modelling using cellular automata for modelling of non-linear models can effectively analyse the smog formation in the form of satellite imaging through cell approach. This type of modelling has been used in the modelling and simulation of dispersion of pollutants [9]. The dynamic behaviours associated with the formation of smog can be effectively implemented in the CA model and improve our understanding of dynamic random dynamic process tied to the aetiology of smog formation because of stubble burning.

The cellular automata (CA) approach is a discrete element framework which allows the implementation of a phenomena or a process through the series of discrete steps in time or space through the concept of formation of cells. Essentially, the standard permits the new condition of a cell in light of the condition of the neighboring cells at each discrete time instant. Apart, it also allows the modeler to model and design complex element frameworks through the determination of the neighborhood dynamics. When utilized as simulation instrument, the CA have shown to be exceptionally helpful in terms of building and modelling of fake scenarios, chiefly in those areas where different methodologies are not reasonable.

The concept of discretization in time and space in the implementation of model instead of mimicking the continuous equations has been employed in the model based on the approach used in the cellular automata model for air quality simulation by Giogrio Guariso and Vittorio Maniezzo in 1992 [10]. The method of discretization in time and space model is based on linear local updating rules which have the unique features: it can be easily calibrated to be consistent with the results of commonly accepted air pollutant cellular automata models and it can immediately fit different environmental conditions in different areas and thus, helps in exploiting the parallelism inherent to the Cellular Automata approach [8].

II. BACKGROUND

The model simulation has been carried out in Lopez Simulator using Cell DEVS approach and thorough the use of state variables because of the number of certain advantages and unique features the Lopez Simulator offer in comparison to other simulation tools as discussed in detail in this section.

The general approach used to formulate and simulate any problem or process is carried out by using the certain simulation five steps as discussed below:

(i) *Real time process analysis*

The simulation starts with the deep observation of a real-time process which has to be modelled and simulated.

(ii) *Model Definition*

The entities for the real-time system are defined and the abstract representation known as the system model is laid out.

(iii) *Simulation*

The designed model with the help of various entities defined for the model is executed and simulated using a simulator. The simulator is run on a computer system with the detailed instructions written in the form of code in order to implement the model.

(iv) Validation

Finally, the results obtained from the simulation process are mapped with the real-time system and compared for validation.

(v) *Testing*

Thereafter, a series of test execution may be carried out on the simulator for testing the model under various circumstances (e.g. to test the system under the worst-case scenario)

The below Figure 3 shows the sequential five basic steps discussed above essential in the modelling and simulation of any process or problem.

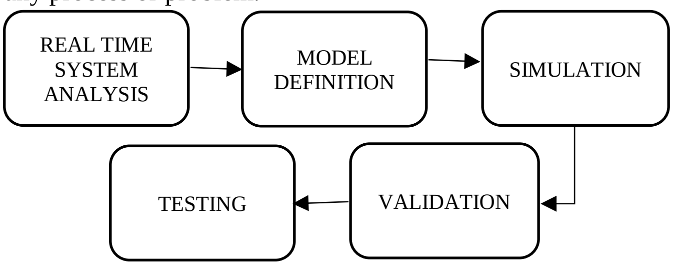


Figure 3 – Steps in the basic simulation process

There are number of simulation techniques and paradigms which can be used to simulate the models. Among these, the **Discrete Event Simulation (DEVS)** formalism provides a framework which allows the construction of hierarchical models in a modular structure, thus allowing for the model reuse and reduces the model development time. In the DEVS modeling approach, a model is considered as a black box with a specific state and a specific duration of that state. When the duration time for the state elapses, an output event occurs and an internal transition takes place leading the model to change its current state to new state. It is to be noted that a change of state can also occur when an external event is received at the input port. The figure 4 below shows the model definition by DEVS formalism approach [5].

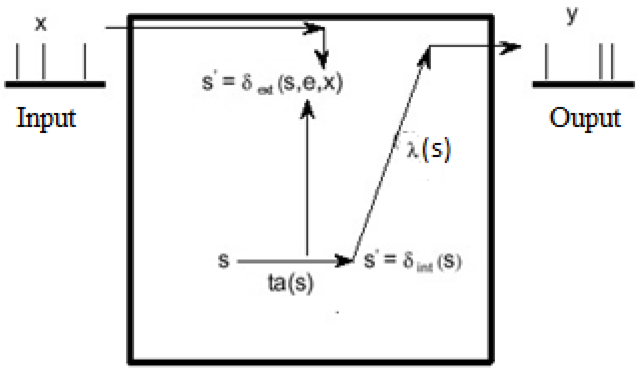


Figure 4 – DEVS Model Representation/Specification [5]

The **Cell-DEVS** formalism is another framework which is based on the DEVS employed for the simulation of cellular models. A cellular model can be defined as a lattice of cells comprising of m rows and n columns (thus, the model has dimensions: m x n), where each cell has a value and a local transition rule that defines how a cell retains a new value based on the current state of the cell and the state of the neighboring cells. The cells in the cellular model are updated synchronously i.e. all cells are updated at the same time. In other words, the Cell-DEVS defines a single cell as a DEVS model and a cellular automaton as a coupled model. The basic advantage of Cell DEVS is that it is more flexible than the other existing synchronous formalism approaches [5].

The formal definition of atomic CELL DEVS model can be given by [5]:

$$CD = < X, Y, I, S, \theta, E, \text{delay}, d, \delta_{int}, \delta_{ext}, \tau, \lambda, D >$$

where;

X = set of external input events;

Y = set of external output events;

I = model's modular interface;
S = set of sequential states for the cell;
θ = set of the cell's state variables;
E = set of states for the input events;
delay = type of delay (can be transport or inertial);
d = transport delay for the cell;
δ_{int} = internal transition function;
δ_{ext} = external transition function;
τ = local computation function;
λ = the output function; and
D = state's duration function.

A formal definition of coupled cell DEVS model can be given by [5]:

CCD = < *Xlist, Ylist, I, X, Y, n,* $\{t_1,...,t_n\}$, *N, C, B, Z, select* >

where;
Xlis = input coupling list;
Ylist = output coupling list;
I=< $\boldsymbol{\eta}$, $\boldsymbol{\mu}^x$, $\boldsymbol{\mu}^y$, $\boldsymbol{p}^x$, $\boldsymbol{p}^y$ > represents the definition of the interface for the modular model whose size is $\eta \in N, \eta < \infty$;
$\boldsymbol{p}^x$ = set of all input ports ($\boldsymbol{\eta}$ neighbor ports + $\boldsymbol{\mu}^x$ external ports)
$\boldsymbol{p}^y$ = set of all output ports
X = set of external input events;
Y = set of external output events;
n = dimensions of the cell space;
$\{t_1,...,t_n\}$ = number of cells in each of the dimensions;
N = neighborhood set;
C = cell space;
B = set of border cells;
Z = translation function; and
select = tie-breaking function for simultaneous events.

There are variety of simulation tools which can be used to model the real-time systems where the choice of simulator depends on the nature and complexity of real time model to be implemented. The various simulation tools which can be used to model the real-time systems are:

(i) CD++ (Eclipse)
(ii) Lopez Simulator

CD++ is a simulation tool for the simulation of DEVS and Cell-DEVS models used to simulate a variety of real time models such as traffic lights, forest fires, and evacuation models. The CD++ allows the ease of implementation and simulation of simple models, but the lack of state variables and the inability to create a number of neighbor ports posed a threat and problem when implementing complex models.

The main limitation of CD++ was that it does not support multiple state variables. Therefore, in order to deal with this problem, the modelers require extra planes in their cell spaces, and create as many new layers in their cellular models as the number of state variables they needed. For example, when one state variable was needed in a planar cell space, the possible solution was to create a three-dimensional cell workspace with two planar layers. The layer 0 (x, y, 0) was used for the initial reference cell value, and the other layer (x, y, 1) was used to store a new value based on the corresponding cells values in the above layer (say layer 0 in this example) [5].

An example of this technique has been shown in Figure 5 below where the cell numbered 'b' value in layer 2 is updated in relation to the corresponding cell number in the layer 1 as shown by the red dotted line.

.

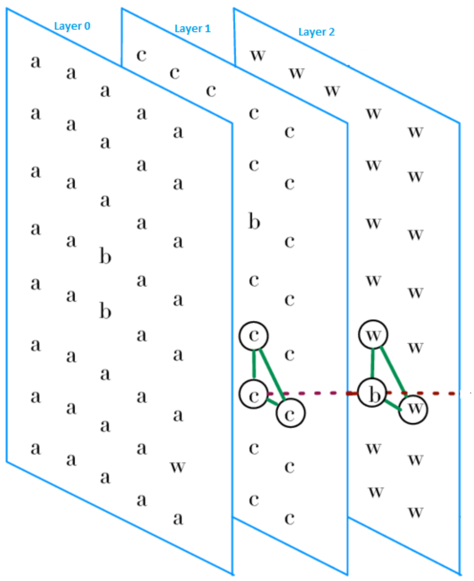


Figure 5 –3Dimensional layered implementation in Cell DEVS

Therefore, there was a need to extend the CD++ to allow the modeler to declare and use of state variables to store values inside the cell, and to declare and use multiple ports to communicate extra values to the neighbor cells. The Lopez Simulator provides the solution to the two-basic limitation of the CD++ CELL DEVS modeler. In Lopez Simulator, the modeller can design the CELL DEVS models by using the state variables or by using the multiple intra cell ports which allows communication to the neighbouring cells in an effective way, thus eliminating the need of multiple layers to be implemented as in case of CD++ Cell DEVS implementation. In the proposed term paper, the models have been designed and simulation has been carried out in Lopez Simulator because of the two basic advantages and features the Lopez Simulator offers as mentioned above. Another important advantage of the Lopez Simulator is that it provides the total simulation time the model takes to run on simulator. In the term paper, the total simulation time of the various proposed models have also been mentioned, compared and analysed [5].

III. MODEL DEFINITION

In this term paper, the complete model simulation has been carried out thought the implementation of four test models in Lopez Simulator, namely:

(i) India Map and Sowing Model
(ii) Crop Phases Model
(iii) Monsoon Model
(iv) Smog Model (final model)

The description of the four implemented models in Lopez Simulator has been mentioned below in Table 1.

Table 1 – Various Implemented Test Models

Model Name	Description
India Map and Sowing Model	The model implements the map of India as viewed in the satellite images and shows the random sowing of crops using normal distribution with normalized mean of 0.4 and standard deviation of 0.3.
Crop Phases Model	The model implements the various phases of crop growth in farming - crop sowing, reproductive phase and ripening phase followed by the crop harvesting. In this model, the sowing of crops follows the normal distribution with normalized mean of 0.4 and standard deviation of 0.3 with threshold of 0.1.
Monsoon Model	The monsoon model implements the effect of south east monsoon on the crop sowing pattern in India i.e. the crops are sown first in south east India followed by crop sowing in south-west, south-west and north-west India in sequential manner. The sowing of crops in the various zones follow normal distribution with mean of 0.4 and standard deviation of 0.3.
Smog Model	This model implements the formation of smog over north-west India because of stubble burning and no formation of smog in other parts of India. The model takes into account the effect of south west monsoon on cop sowing and transitions through the various crop phases followed by harvesting.

The model definitions and the rules for the various models mentioned above have been discussed below in detail.

(i) *India Map and Sowing Model Definition*
In this model, the replica of the India map as seen from the satellite image has been created in Lopez Simulator (using multiple state values) having cell space of 20 x 20 i.e. the cell space has 20 rows and 20 columns. In the model implementation, the state variable named 'value' has been used to update the cell values and the initial values of the state variable 'value' have been defined in the file named indiamapandsowing.stvalues.

The formal definition of this cellular automata model implemented in Lopez Simulator has been mentioned below in Figure 6:

A = < X, Y, S, N, type, d, τ, δint, δext, λ, ta
X = Y = S = {0,1,2,11};
(0: initial state; 1: neighboring country state; 2: crop sowing phase; 11: ocean)
N = { (0,0), (0,-1), (-1,0), (1,0), (0,1)};
type = transport;
d = { 1000 };
δint , δext , λ, and ta are denied using Cell-DEVS specification.

Figure 6 – Formal Definition of India Map and Sowing Model

It is to noted that τ(N) depends on location of the cell within the coupled model cell space. The Table 2 below shows the various transition rules for the implemented model.

Table 2 – Transition rules for India Map and Sowing Model

Location	τ(N)	Description
τ_1 (0,0)	2	{$value} {$value =2} 1000 { $value =0 and normal(0.4,0.3)>0.1 } % update the variable named 'value' to 2 and assign the updated value of variable to the cell if the initial value of

		variable 'value' of cell (0,0) is zero and the normal(0.4,03)>0.1 i..e sowing of crops follows normal distribution
	2	{ $value } { $value := 2; } 1000 { $value =2} % if value=2, assign the value of variable to the cell i.e. assign 2 to the cell
	1	{ $value } { $value := 1; } 1000 { $value =1} %if the value=1, assign the value of variable to the cell i.e. assign 1 to the cell
	11	{ $value } { $value := 11; } 1000 { $value =11} %if the value=11, assign the value of variable to the cell i.e. assign 11 to the cell (ocean)
	0	TRUE

The colour palate for the corresponding India Map and Sowing Model has been shown below in Table 3:

Table 3 – Colour Palate for the India Map and Sowing Model

Cell State	Cell Value	Cell Colour
Initial State	0	
Boundary countries	1	
Crop Sowing	2	
Ocean (water)	11	

(ii) *Crop Phases Model Definition*

In this model, the various phases of crop growth in farming i.e. crop sowing, crop reproduction phase and the ripening phase followed by the harvesting of crops has been implemented in Lopez simulator using state variables. The initial values of the state variable 'state' has been used to store the cell values has been stored in the initial state values file named cropphases.stvalues. The formal definition of the crop phases CA model has been mentioned below in Figure 7:

A = < X, Y, S, N, type, d, τ, δint, δext, λ, ta
X = Y = S = {0,1,2,3,4,5,11};
(0: initial state; 1: neighboring country state; 2: crop sowing phase; 3: reproduction phase; 4: ripening phase; 5: crop harvesting;11: ocean)
N = { (0,0), (0,-1), (-1,0), (1,0), (0,1)};
type = transport;
d = { 1000 };
δint , δext , λ, and ta are denied using Cell-DEVS specification.

Figure 7 – Formal Definition of Crop Phases Model

It is to noted that τ(N) depends on location of the cell within the coupled model cell space. The Table 4 below shows the various transition rules for the implemented crop phases model.

Table 4 – Transition rules for Crop Phases Model

Location	τ(N)	Description
τ_1 (0,0)	2	{$value} {$value =2} 1000 { $value =0 and normal(0.4,0.3)>0.1 } % update the variable named 'value' to 2 and assign the updated value of variable to the cell if the initial value of variable 'value' of cell (0,0) is zero and the normal(0.4,03)>0.1 i..e sowing of crops follows random normal distribution
τ_1 (0,0)	1	{ $value } { $value := 1; } 1000 { $value =1} %if the value=1, assign the value of variable to the cell i.e. assign 1 to the cell (neighbouring countries in map)
τ_1 (0,0)	11	{ $value } { $value := 11; } 1000 { $value =11} %if the value=11, assign the value of variable to the cell i.e. assign 11 to the cell (ocean)
τ_2 (0,0)	3	{$value} {$value =3} 3000 { $value =2 } % transition from crop sowing to reproduction phase
τ_3 (0,0)	4	{$value} {$value =4} 5000 { $value =3 } %transition from reproduction to ripening phase
τ_4 (0,0)	5	{$value} {$value =5} 7000 { $value =4 } %transition from ripening to crop harvesting phase

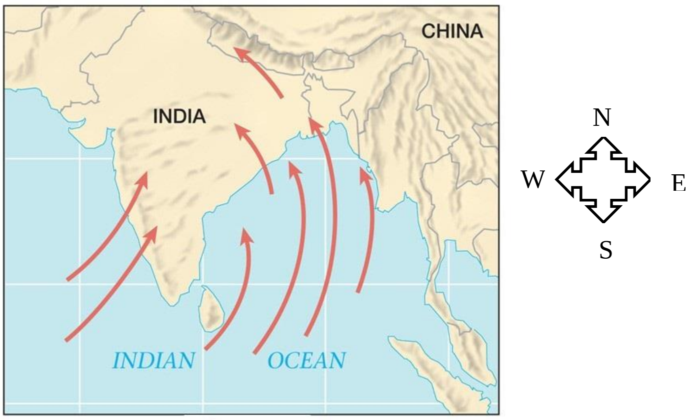
The colour palate for the corresponding crop phases model has been shown below in Table 5:

Table 5 – Colour Palate for the Crop Phases Model

Cell State	Cell Value	Cell Colour
Initial State	0	
Boundary countries	1	
Crop Sowing	2	
Reproductive Phase	3	
Ripening Phase	4	
Crop Harvesting	5	
Ocean (water)	11	

(iii) *Monsoon Model*

The Monsoon Model implements the effect of arrival of south east monsoon on the crop sowing pattern in India. The crop sowing in India is influenced by the arrival of monsoons. The south-east monsoon arrives earlier in the southern eastern part of India in comparison to other part of India. Therefore, the crops are sown first in the south-east India due to early arrival of monsoons followed by the crop sowing in the north-east India, south-west India and north-west India in sequential manner. The below figure 8 below shows the pattern of arrival of south east monsoon in India in summer [8].



Summer

Figure 8 – Arrival of South-East Monsoon in India [8]

The monsoon imitates the effect of south east monsoon arrival on the crop sowing pattern in India. The model has been implemented by dividing the cell space of map of India into four zones namely – south-east zone, south-west zone, north-west zone and north-east zone where each zone cells have the state variables and the state variables of the cells in different zones are updated in such a way that the crops are sown first in south-east zone followed by crop sowing in north east-zone, south-west zone and north-west zone.

The monsoon model employs the cell space of 20 rows and 20 columns (i.e. 20 x 20) where each cell has a state variable named ‘state’. The nature of cell space used in the monsoon model is no warped and the initial values of the cells have has been initialized to zero. The initial values of the state variable ‘value’ is defined in the file named monsoon.stvalues.

The formal definition and specifications of the monsoon cellular automata model implemented in Lopez Simulator has been described in Figure 9. The cell space uses a set of four neighbouring cells as neighbour space and the nature of cell space used in the model is no-warped (or un-warped).

A = < X, Y, S, N, type, d, τ, δint, δext, λ, ta
X = Y = S = {0,1,2,7,8,9,10, 15};
(0: initial state; 1: neighboring country state; 2: crop sowing phase; 7: north-west zone; 8: south-west zone; 9: north-east zone; 10: south-east zone- 15: ocean)
N = { (0,0), (0,-1), (-1,0), (1,0), (0,1)};
type = transport;
d = { 1000 };
δint , δext , λ, and ta are denied using Cell-DEVS specification.

Figure 9 – Formal Definition of Monsoon Model

It is to noted that τ(N) depends on location of the cell within the coupled model cell space. The Table 6 below shows the various transition rules for the implemented monsoon model.

Table 6 – Transition rules for Monsoon Model

Location	**τ(N)**	**Description**
τ_1 (0,0)	2	{$value} {$value =2} 1000 { $value =10 and normal(0.4,0.3)>0.15 } %sowing of crops in south-east India following normal distribution
τ_1 (0,0)	1	{ $value } { $value := 1; } 1000 { $value =1} %if the value=1, assign the value of variable to the cell i.e. assign 1 to the cell (neighbouring countries in map)
τ_1 (0,0)	15	{ $value } { $value := 15; } 3000 { $value =15} %if the value=15, assign the value of variable to the cell i.e. assign 15 to the cell (ocean)
τ_2 (0,0)	2	{$value} {$value =2} 3000 { $value =9 and normal(0.4,0.3)>0.15 } % sowing of crops in north-east India following normal distribution
τ_3 (0,0)	2	{$value} {$value =2} 5000 { $value =8 and normal(0.4,0.3)>0.15 } % sowing of crops in south-west India following normal distribution
τ_4 (0,0)	2	{$value} {$value =2} 6000 { $value =7 and normal(0.4,0.3)>0.15 } % sowing of crops in north-west India following normal distribution

The colour palate for the corresponding monsoon model has been shown below in Table 7:

Table 7 – Colour Palate for the Monsoon Model

Cell State	**Cell Value**	**Cell Colour**
Cells Initial State	0	
Boundary countries	1	
Crop Sowing	2	
North-West Zone initialization	7	
South-West Zone initialization	8	
North-East Zone initialization	9	
South-East Zone initialization	10	
Ocean (water)	15	

(iv) *Smog Model (final model)*

This model incorporates all the implementations and conclusions of the above implemented three models – India map and sowing model, crop phases model and monsoon model and shows the formation of smog over north India due to stubble burning of crops using state values in Lopez Simulator.

The model implements the four zones - south-east zone, south-west zone, north-west zone and north-east zone where each zone cells have the state variable named 'value' and the state variables of the cells in different zones are updated in such a way that the crops are sown first in south-east zone, followed by crop sowing in north east-zone, south-west zone and north-west zone. Followed by the crop sowing, the sown crops transients through the various crop growth phases in farming as implemented in second model named crop phases model.

It is to be noted that since the crops are sown first in south east India, therefore, the crops in south-east India are will be first transitioning from sowing to the ripening phase. When the crops reach the ripening phase in different zones, they are harvesting. After harvesting, the stubble left over after harvesting is burnt in north-west zone whereas, the stubble is ploughed in south-east, south-west and north-east zones. The burning of stubble in north west zone creates smoke and finally leads to the smog formation over north west India.

The formal definition and specifications of the smog cellular automata model implemented in Lopez Simulator has been described in Figure 10. The cell space employed in the model has dimensions of 20 rows and 20 columns (20x20) and uses a set of four neighbouring cells as neighbour space. The nature of cell space used in the model is no-warped (or un-warped).

A = < X, Y, S, N, type, d, τ, δint, δext, λ, ta
X = Y = S = {0,1,2,3,4,5,6,7,8,9,10,11,12,13,15,16,17,18};
(0: initial state; 1: neighboring country state; 2: crop sowing phase; 3: reproductive phase; 4: ripening phase; 5: crop harvesting phase; 6: ploughing; 7: north-west zone; 8: crop sowing in north west zone; 9: crop ripening in north-west zone; 10: crop ripening in north west zone; 11: crop harvesting in north west zone; 12: stubble burning phase; 13: smog formation phase; 15: ocean; 16: south-east zone; 17: north-east zone; 18: south-west zone;)
N = { (0,0), (0,-1), (-1,0), (1,0), (0,1)};
type = transport;
d = { 1000 };
δint , δext , λ, and ta are denied using Cell-DEVS specification.

Figure 10 – Formal Definition of Smog Model

It is to noted that τ(N) depends on location of the cell within the coupled model cell space. The Table 8 below shows the various transition rules for the implemented smog model.

Table 8 – Transition rules for Smog Model

Location	**τ(N)**	**Description**
τ_1 (0,0)	1	{ \$value } { \$value := 1; } 1000 { \$value =1} %if the value=1, assign the value of variable to the cell i.e. assign 1 to the cell (neighbouring countries in map)
τ_1 (0,0)	15	{ \$value } { \$value := 15; } 3000 { \$value =15} %if the value=15, assign the value of variable to the cell i.e. assign 15 to the cell (ocean)
τ_2 (0,0)	2	{\$value} {\$value =2} 2000 { \$value =16 and normal(0.4,0.3)>0.15 } %sowing of crops in south-east India following normal distribution
τ_2 (0,0)	2	{\$value} {\$value =2} 2000 { \$value =17 and normal(0.4,0.3)>0.15 } %sowing of crops in north-east India following normal distribution
τ_3 (0,0)	2	{\$value} {\$value =2} 2500 { \$value =18 and normal(0.4,0.3)>0.15 } %sowing of crops in south-west India following normal distribution

τ_3 (0,0)	8	{$value} {$value =8} 2500 { $value =7 and normal(0.4,0.3)>0.15 } %sowing of crops in north-west India following normal distribution
τ_4 (0,0)	3	{$value} {$value =3} 3000 { $value =2 } %crop reproductive phase in north-east zone, south-west zone and south-east zone
τ_4 (0,0)	9	{$value} {$value =9} 3000 { $value =8 } %crop reproductive phase in north-west zone
τ_5 (0,0)	4	{$value} {$value =4} 4000 { $value =3 } %crop ripening in north-east zone, south-west zone and south-east zone
τ_5 (0,0)	10	{$value} {$value =10} 4000 { $value =9 } %crop ripening in north-west zone
τ_6 (0,0)	5	{$value} {$value =5} 5000 { $value =4 } %crop harvesting phase in north-east zone, south-west zone and south-east zone
τ_6 (0,0)	11	{$value} {$value =11} 5000 { $value =10 } %crop harvesting phase in north-west zone
τ_7 (0,0)	6	{$value} {$value =6} 6000 { $value =5 } %ploughing of stubble in north-east zone, south-west zone and south-east zone
τ_7 (0,0)	12	{$value} {$value =12} 6000 { $value =11 } %stubble burning in north-west zone
τ_8 (0,0)	13	{$value} {$value =13} 7000 { $value =12 } %smog formation in north-west zone

The colour palate for the corresponding smog model has been shown below in Table 9:

Table 9 – Colour Palate for the Smog Model

Cell State	**Cell Value**	**Cell Colour**
Initial State	0	
Boundary countries	1	
Crop Sowing in north east, south west and south east zone	2	
Reproductive phase of crops in in north east, south west and south east zone	3	
Ripening Phase of Crops in north east, south west and south east zone	4	
Crop Harvesting in in north east, south west and south east zone	5	
Stubble Ploughing in north east, south west and south east zone	6	
North-west zone initialization	7	
Crop Sowing in North west zone	8	
Reproductive Phase of crops in North west zone	9	
Ripening Phase of crops in North west zone	10	
Crop Harvesting in North west zone	11	
Stubble Burning in North west zone	12	
Smog formation in North west zone	13	
Ocean (water)	15	
South-east zone initialization	16	
North-west zone initialization	17	
South-west zone initialization	18	

IV. MODEL IMPLEMENTATION

The models defined above have been implemented using Lopez Simulator Client downloaded as RISE_Client_vs1_server from the course website. The basic files required to run the four models have been below in Table 10. The necessary files required to run any model in Lopez simulator using state variables are: .ma file, .xml file and .stvalues file. The .stvlues file is optional but we have initialized the cell values in all the models.

Table 10 – Necessary model files required to run model in Lopez Simulator

S. No.	Model Description	Files required to run the respective model
1	India Map and Sowing Model	(i) indiamapandsowing.ma (ii) indiamapandsowing.xml (iii) indiamapandsowing.stvalues
2	Crop Phases Model	(i) cropphases.ma (ii) cropphases.xml (iii) cropphases.stvalues
3	Monsoon Model	(i) monsoon.ma (ii) monsoon.xml (iii) monsoon.stvalues
4	Smog Model	(i) smog.ma (ii) smog.xml (iii) smog.stvalues

It is to be noted that the .ma file and .stvalues file are zipped together in a clean folder with the same name as the model and the .zip file containing the .ma file and .stvalues file should be placed in the same folder containing the RESTful_CDppTest.jar file. Make sure that the internet connection is established and running. The models have been run with the help of Lopez Simulator using DOS Prompt using the following steps mentioned below:

STEP I - Open the DOS PROMPT and go the folder containing the file RESTful_CDppTest.jar inside folder RISE_Client_vs1_server using cd command in DOS Prompt as shown in Figure 11.

For example, use command:
>>cd Users\Ekambir\Downloads\RISE_Client_vs1_server

STEP 2 - Use the below command in DOS Prompt to create workspace on server and provide the model xml file.

java -jar RESTful_CDppTest.jar PutXMLFile test test test/lopez/ekambir monsoon.xml

It is to be noted that in the example, the monsoon model has been implemented. Here, ekambir is the name of workspace and monsoon.xml is the xml file for model monsoon containing the details of monsoon model to be simulated.

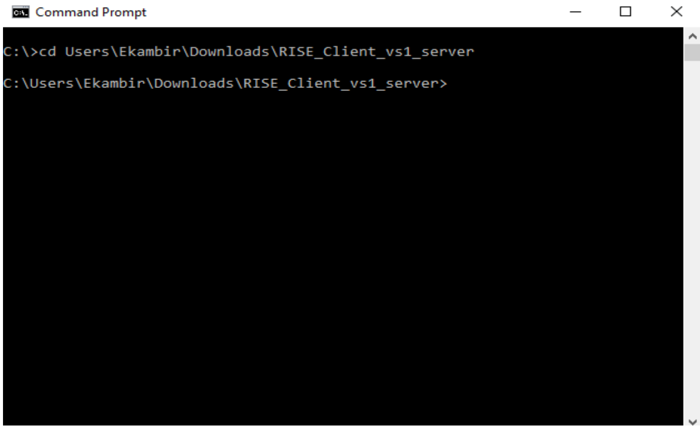


Figure 11 – Implementation of CD command to reach the folder

The figure 12 below shows the creation of workspace on server using the command discussed in Step II above.

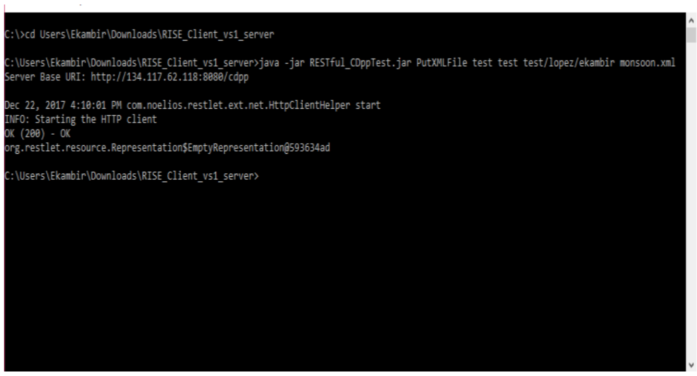


Figure 12 – Creating workspace and loading xml file

STEP 3: Use the below command in DOS Prompt to provide the .ma file and other files (e.g. .val or .stvalues files) zipped together in Figure 13.

java -jar RESTful_CDppTest.jar PostZipFile test test test/lopez/ekambir?zdir=monsoon monsoon.zip

[NOTE: Here monsoon is the zip file name and we are using the previously created workspace 'ekambir']

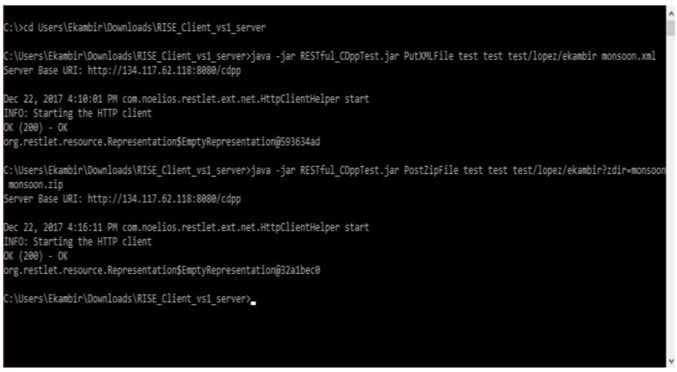


Figure 13 – Providing access to .zip file on server for simulation

STEP 4: Simulate the model by using the command below in DOS Prompt as shown in Figure 14:

```
java -jar RESTful_CDppTest.jar PutFramework test test test/lopez/ekambir/simulation
```

[NOTE: The workspace is ekambir in the above instruction used for simulation]

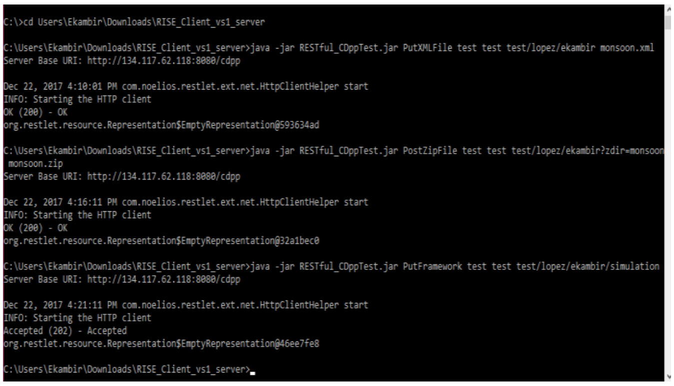


Figure 14 – showing the simulation of the model done correctly

STEP 5: Download the simulated results online form the below link:

```
http://vs1.sce.carleton.ca:8080/cdpp/sim/workspaces/test/lopez/ekambir
```

The simulation results and the Model Debugging /Statistical Logs for the simulated model can be downloaded online from the link mentioned above. The output of simulation is available in monsoon.log01. The same steps as mentioned above for monsoon model has been carried out for the other models for the simulation of the respective models.

V. RESULTS

The results of the various test models defined above has been discussed below:

(i) *India Map and Sowing Model – Simulation Results*

The results of the India Map and Sowing model have been shown in Figure 16(a) and Figure 16(b) below. The figure 16(a) shows the cells initialized to values ‘1’ and ‘11’ at time t=1:40 and the cells which are not initialized are initialized by random sowing of crops (state ‘1’) at t=3:20 seconds in Figure 16(b).

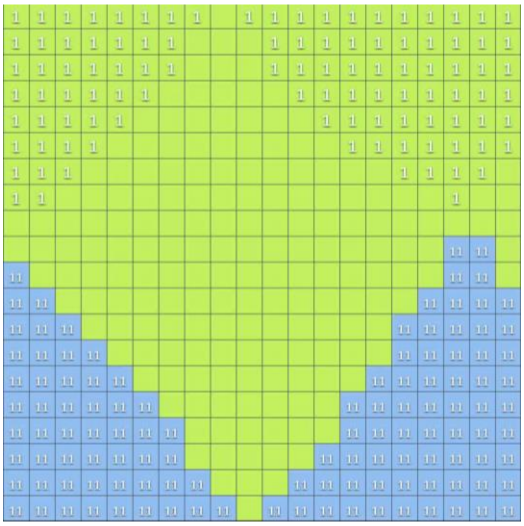


Figure 16(a) India map and sowing model output showing the cell initializations at t=1:40

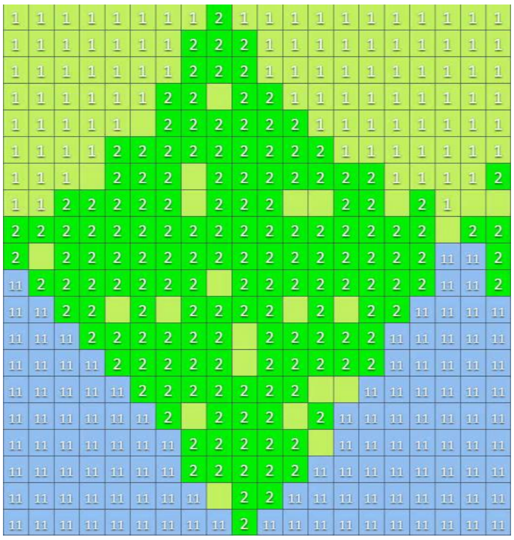


Figure 16 (b) India map and sowing model output showing the crop sowing following normal distribution at t=2:30

The cell state ‘1’ in the above Figure 16(a) and Figure 16(b) indicates the neighbouring countries of India (Pakistan on left side and China on right side). The cell state ‘11’ indicates the

Indian Ocean at the bottom part of India and the cell state ‘2’ illustrates the sowing of crops which follows the normal distribution. The cells in the above Figures 16 (a) and Figure 16(b) which do not have values are initialized to cell state ‘0’.

(ii) *Crop Phases Model – Simulation Results*

The simulation results of crop phases model have been shown in Figure 17(a) - Figure 17(d) at various discrete time instants.

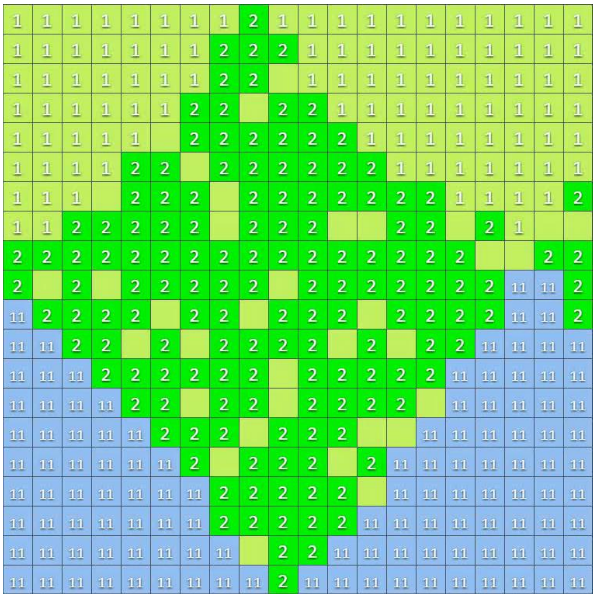


Figure 17(a) - Simulation of Crop Phases Model at t=1:40 showing the sowing of crops (cell state ‘2’) following normal distribution

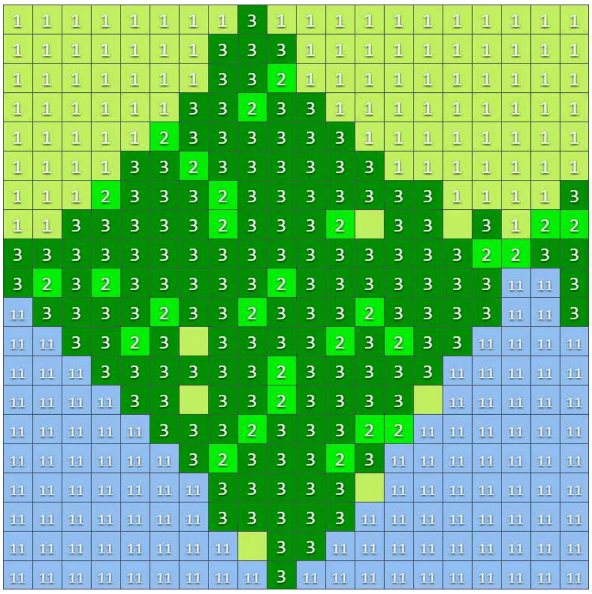


Figure 17(b) - Simulation of Crop Phases Model at t=6:40 showing the crops transition from sowing (cell state ‘2’) to reproductive phase (cell state ‘3’)

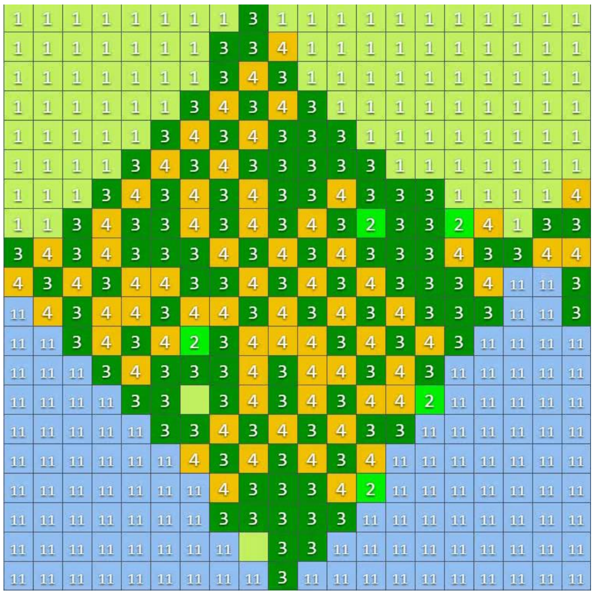


Figure 17(c) Simulation of Crop Phases Model at t=11:40 showing the crops transition from reproductive phase (cell state ‘3’) to ripening phase (cell state ‘4’)

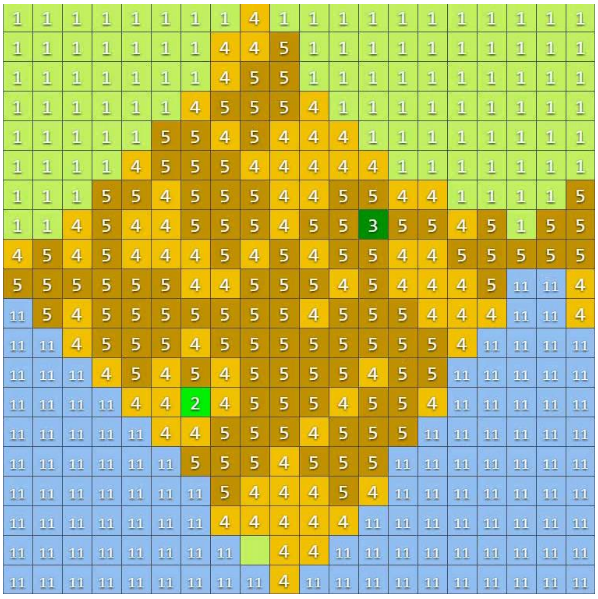


Figure 17(d) Simulation of Crop Phases Model at t=20-:00 showing the crops transition from ripening phase (cell state ‘4’) to ripening phase (cell state ‘5’)

The simulation results in Figure 17(a) – Figure 17(d) show the various crop growth phases the crop transitions through in India before it is harvested. The crops in the model are sown randomly following normal distribution (state ‘2’) which finally transition to reproductive phase (state ‘3’) followed by transition to ripening phase (state ‘4’) and finally the ripened crops are harvested (state ‘5’).

(iii) *Monsoon Model – Simulation Results*

The Monsoon Model simulation results are shown in Figure 18(a) to Figure 18(d)

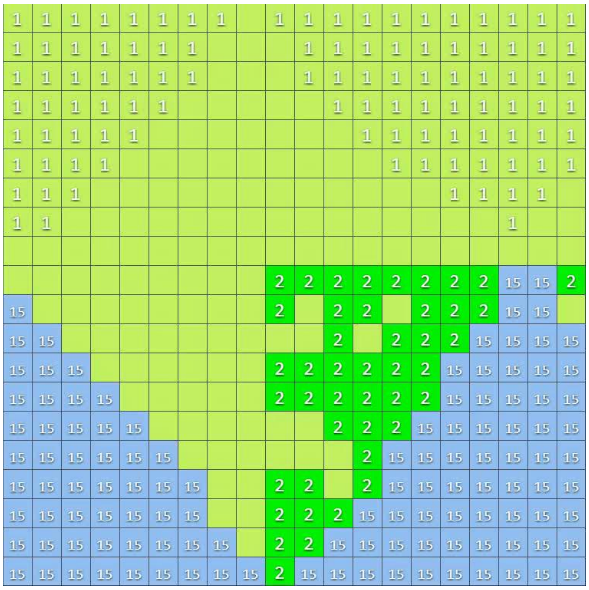


Figure 18(a) - Simulation of Monsoon Model at t=1:40 showing the crops sowing (cell state ‘2’) in the south-east zone of India due to pre-arrival of south-east monsoon in south-east India in comparison to rest other parts of India

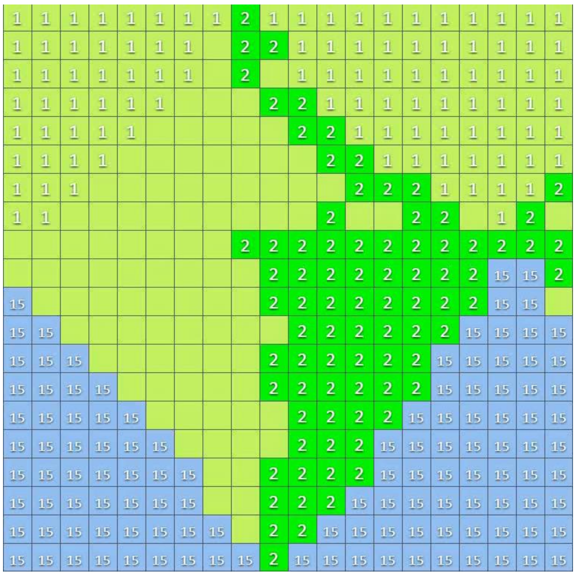


Figure 18(a) - Simulation of Monsoon Model at t=5:00 showing the crops sowing in south-east zone and north-east zone of India (cell state ‘2’) following normal distribution

The Figure 18(a) to Figure 18(d) shows that the crops are first sown in south east--India under the influence of early arrival of south-east monsoons, followed by crop sowing in north-east India, followed by crop sowing in south-west India and finally crops are sown in the north-west India at last. The sowing of crops in four different zones follow random normal distribution with normalized mean of 0.4 and standard deviation of 0.3. The technique which has been employed to sow the crops in the different regions of India at different time instants in sequential manner is the formation of zones in Lopez Simulator and the four different zones have been initialized in form of initial state values in .stvalues file.

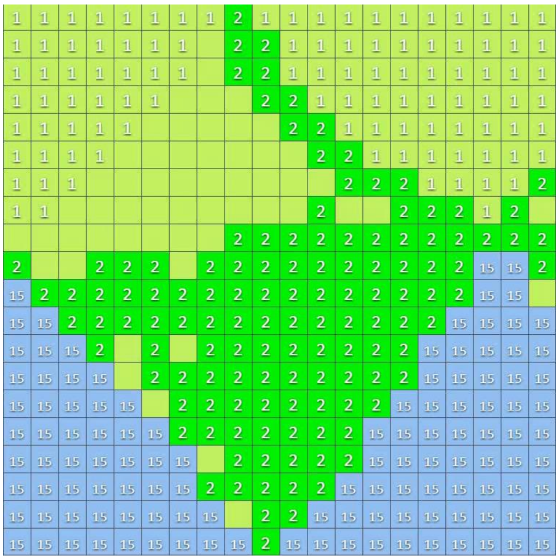


Figure 18(c) - Simulation of Monsoon Model at t=8:20 showing the sowing of crops in south-east, north-east and south-west zones of India following normal distribution

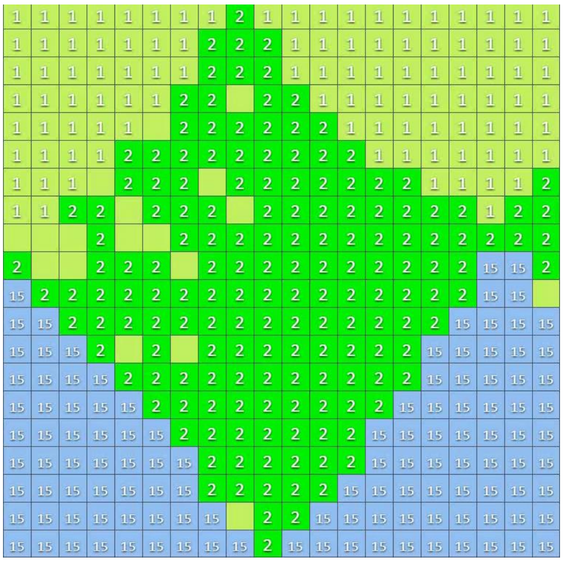


Figure 18(d)- Simulation of Monsoon Model at t=10:00 showing the sowing of crops in south-east, north-east, south-west and north-west zones of India following normal distribution

(iv) *Smog Model – Simulation Results*

The Figure 19(a)-Figure 19(g) shows the output and simulation results of smog model.

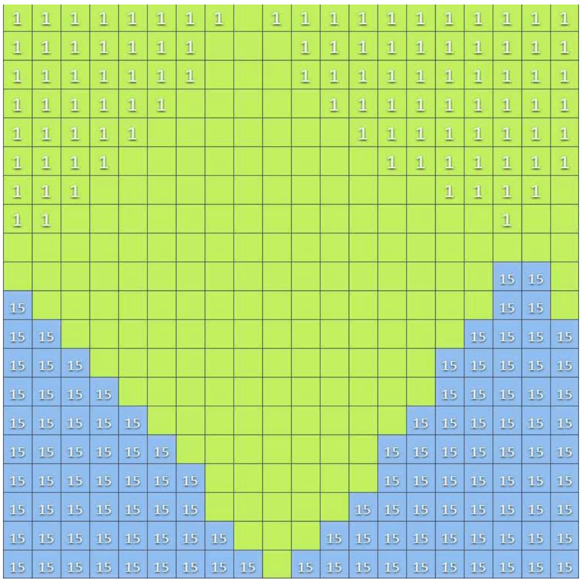


Figure 19(a) – Simulation output of Smog Model at t=1:40 showing the cell initializations representing the Pakistan and China neighbouring counties (cell state'1') and Indian ocean (cell state '15') at the bottom in blue colour

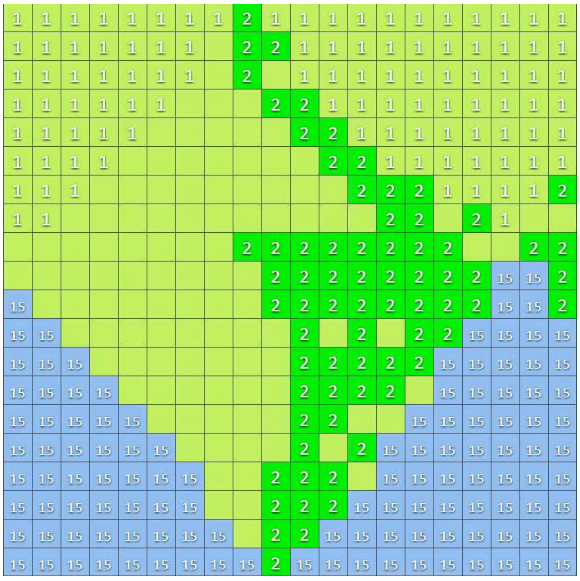


Figure 19(b) – Simulation output of Smog Model at t=3:20 illustrating the crop sowing in south-east and north-east part of India under the influence of arrival of south east India (cell state'2') following normal distribution

The above Figures 19(a) and Figure 19(b) shows the cell initializations and the crop sowing under the influence of south-west monsoon i.e. the crops are sown first in eastern part of India followed by western part of India. This has been implemented by the dividing the cell space into four zones – south east zone, south-west zone, north-east zone and north-west zone. The crops are first sown in south-east and north-east India followed by the crop sowing in the south-west and north-west India as shown in Figure 19(c) below.

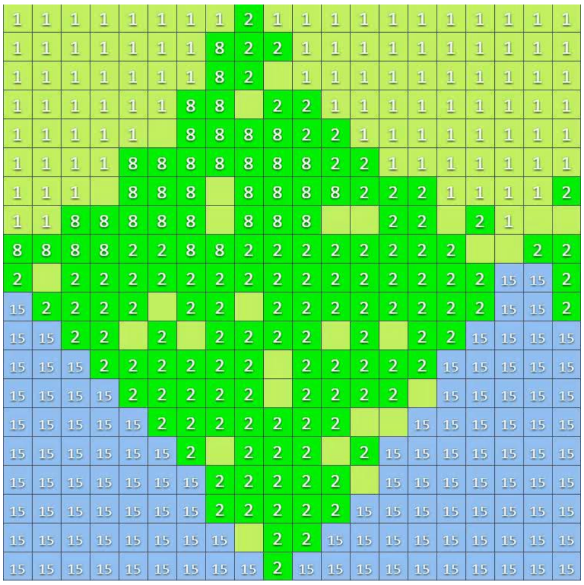


Figure 19(c) – Simulation output of Smog Model at t=4:10 showing thr crop sowing in the western part of India (cell state '2' and cell state '8') following the crop sowing in eastern part of India

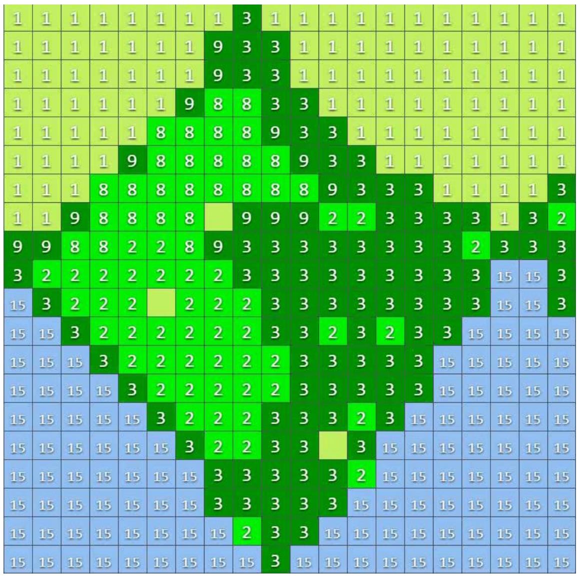


Figure 19(d) - Simulation output of Smog Model at t=8:20 showing the reproductive phase of crops in the south-east zone and north-east zone of India due the fact that the crops sown earlier will reach reproductive phase earlier than the crops sown later in the other parts of India

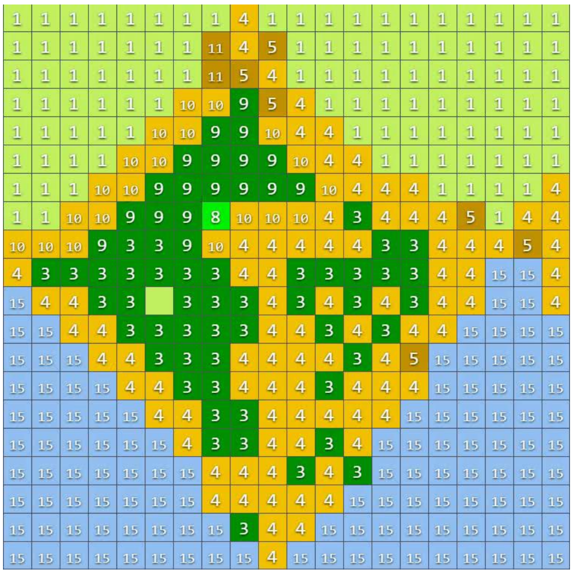


Figure 19(d) - Simulation output of Smog Model at t=13:20 showing the crops transition from reproductive phase (cell state '3' and cell state '9') to ripening phase (cell state '4' and cell state '10') in the different zones

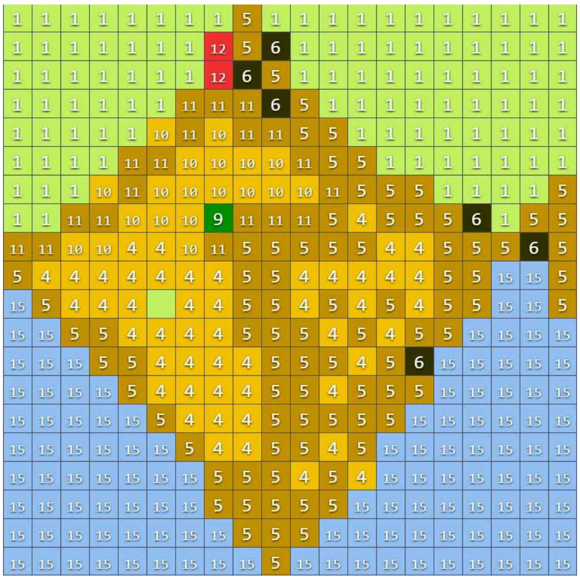


Figure 19(e) - Simulation output of Smog Model at t=16:40 showing the transition of ripened crops (cell state '4' and cell state '10') to crop harvesting (cell state '5' and cell state '11') in different zones

It can be observed from the Figure 19(a) - Figure 19(g) that the different state variable values and cell states are used for the north-west zone in comparison to rest of the zones in order the differentiate it from the other zones in the process of smog formation over moth-west India. Apart, it can be observed from the figures that the crops which are sown first, transitions through the respective crop phases in sequential manner earlier than the crops sown later. The Figure 19(f) shows the burning of stubble (in red colour) by the farmers in north-west India which leads to the formation of smog over north-west India shown by grey colour in Figure 19(g).

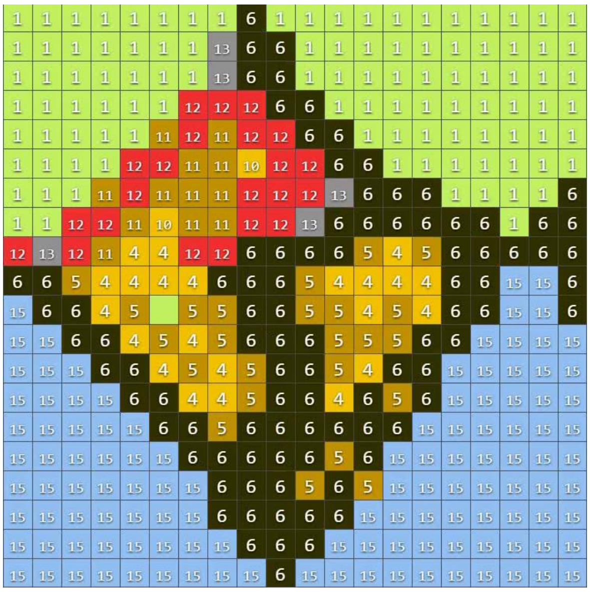


Figure 19 (f) - Simulation output of Smog Model at t=20:50 showing the crop harvested in south-west zone, north-east zone and south-east zone (cell state'5') are ploughed (cell state '6') whereas after crops harvested in north-west India (cell state '11'), the stubble is burnt (cell state '12')

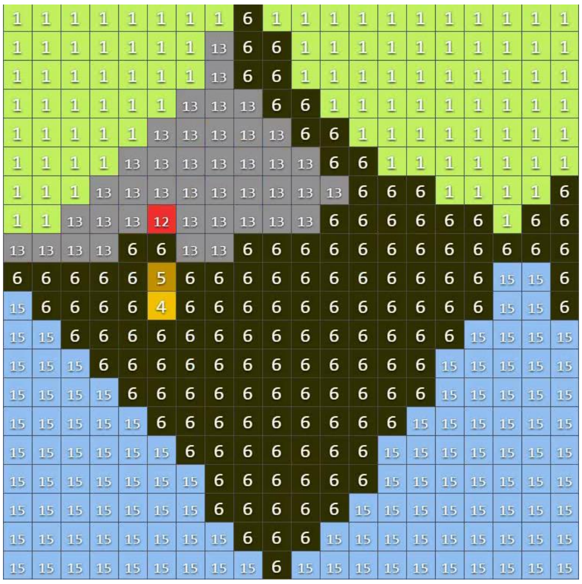


Figure 19 (g) - Simulation output of Smog Model at t=30:50 showing the formation of smog over the north-west zone of India (cell state '13') due to stubble burning whereas the sky is clear and there is no smog formation in north-east zone, south-east zone and south-west zone.

The simulation time recorded for the three random simulations for each model have been recorded and tabulated below in Table 11.

Table 11 – Total Simulation Time required for simulating the various models using state variables

S. No.	Model Description	1st Run Duration (t1)	2nd Run Duration (t2)	3rd Run Duration (t3)	Average Duration (tavg)
1	India Map and Sowing Model	0.120 seconds	0.132 seconds	0.131 seconds	0.383 seconds
3	Monsoon Model	2.283 seconds	2.280 seconds	2.284 seconds	3.423 seconds
2	Crop Phases Model	4.459 seconds	4.614 seconds	4.588 seconds	6.605 seconds
4	Smog Model	4.582 seconds	4.814 seconds	4.591 seconds	6.993 seconds

The average simulation time, t_{avg} in the above table has been computed as the mean of the simulation time taken by three simulations for the respective model i.e.

$$tavg = \frac{1}{3}\sum_{i=1}^{3} ti = \frac{(t1 + t2 + t3)}{3} \quad (1)$$

where;

t_{avg} = average simulation time

t1 = simulation time during first run

t2 = simulation time during second run

t3 = simulation time during third run

The Figure 20 shows the comparison of the simulation times tabulated in Table 11 above in the form of bar-graphs.

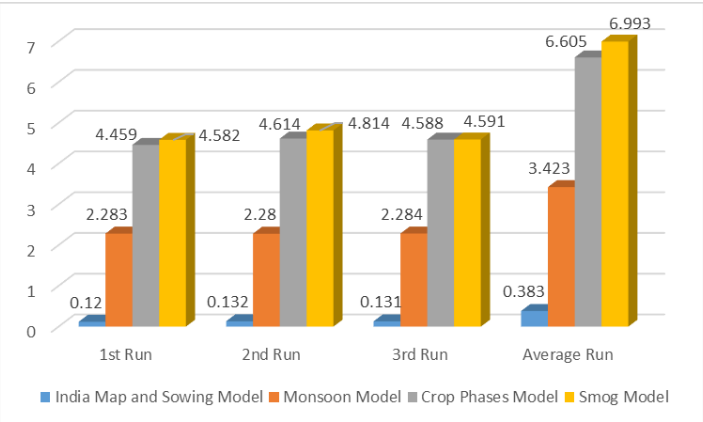
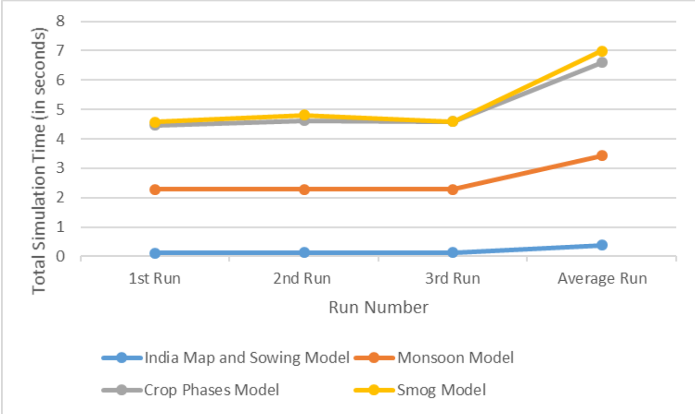


Figure 20 – Comparison of simulation time for the various models using state variables implemented in Lopez Simulator

It can be concluded from the figure that as the complexity of model increases, the simulation time also increases. For example, the smog model employing maximum number of state variable values and computations/cell assignments requires the maximum simulation time. The below figure 21 plots the average simulation time required for the various models



implemented in Lopez Simulator.

Figure 21 – Simulation time plots for the various models implemented and simulated in Lopez Simulator

VI. CONCLUSION

In this term paper, the simulation and modelling of smog formation over the north-west India has been carried out through the implementations of various test models using state variables in Lopez Simulator and the total simulation time of the implemented models have been compared and analysed. The four test models namely, India map and Sowing Model, Monsoon Model, Crop Phases Model and Smog Model have been implemented using state variables in Lopez Simulator.

The India Map and Sowing Model implements the basic map of India as seen through the satellite and implements the random sowing of crops following normal distribution with mean of 0.4 and standard deviation of 0.3. The crop phases model implements the basic phases of crop growth in farming i.e. crop sowing, reproductive phase, ripening phase followed by crop harvesting following gaussian distribution. The monsoon model implements the crop sowing by dividing the India into four zones – south-east, north-east, south-west and north-west monsoon under the influence of south-east monsoon. The crops are first grown in the south-east zone due to the early arrival of monsoon in south-east zone followed by crop sowing in north-east, south-west and north-west zones in sequential manner. The final smog model implements the smog formation over north-west India due to stubble burning of crops. The smog model divides the India into four zones where each cell in the respective zone has its state variable and the sowing of crops in the zones has been carried out considering the effect of monsoon. Further, it has been taken care of the fact that the crops in the regions which are sown earlier will transition through crop growth phases earlier and will be harvested than the crops in the regions which are sown later. After the harvesting of crops, the stubble in north-west zone is burnt by the farmer whereas the stubble in other three zones will be ploughed shown by the change of cell state variable values and cell value assignments in Lopez Simulator. The stubble burning of crops finally leads to formation of smog over the north-west India whereas there will be no smog formation over the rest of the zones in the model.

Further, the total simulation times for three random simulations of the implemented models in Lopez Simulator has been noted and compared. It has been observed that higher is the model complexity in terms of more state variable value assignments, the model simulation time increases. It has been concluded among all the implemented models that the smog model has the maximum simulation time.

Acknowledgment

We would like to express our sincere gratitude and thanks towards our Prof. Gabriel A Wainer, Department of Systems and Computer Engineering, Carleton University, Ottawa (CA) for providing an invaluable opportunity to work on this project as a part of the course – Methodologies for discrete event modelling and simulation (SYSC5104) and his timely guidance throughout the course during term Fall 2017.

References

[1] https://en.wikipedia.org/wiki/Smog
[2] https://en.wikipedia.org/wiki/Smog_in_Delhi
[3] https://visibleearth.nasa.gov/view.php?id=75053
[4] http://www.financialexpress.com/photos/business-gallery/926389/delhi-pollution-nasa-photos-show-smoke-clouds-heading-from-punjab-haryana-to-delhi-see-horrific-visuals/
[5]http://www.sce.carleton.ca/courses/sysc5104/lib/exe/fetch.php?media=cd_lopez.zip
[6] http://www.cbc.ca/radio/thecurrent/the-current-for-november-13-2017-1.4397572/new-delhi-s-toxic-smog-poses-serious-health-threat-warns-doctor-1.4397897
[7] https://visibleearth.nasa.gov/view.php?id=64765
[8] https://wildcardweather.com/2014/09/07/monsoons-and-the-north-american-monsoon/
[9] M. Marín, V. Rauch, A. Rojas-Molina, C.S López-Cajún, A. Herrera, V.M Castaño, *"Cellular Automata simulation of dispersion of pollutants",* Computational Material Science, ELSEVIER, vol. 18, Issue 12, pp. 132-140, August 2000.
[10] Giorgio Guariso, Vittorio Maniezzo, "*Air quality through cellular automata model"*, Environmental Modeling and Software, ELSEVIER, vol. 7, Issue 3, pp. 131-141, 1992.