**SYSC 5104: Methodologies for Discrete-Event Modeling and Simulation**

**Instructor: Prof. Gabriel A. Wainer**

**Using DEVS and CELL-DEVS for Modelling of**

**Information Impact on Individuals in Social Network**

**ASSIGNMENT 2**

**Pankti Vyas**

**Student ID : 101080924**

[**panktivyas@cmail.carleton.ca**](mailto:panktivyas@cmail.carleton.ca)

**November 24th, 2017**

* **CONCEPTUAL MODEL** :

This paper depicts the modelling and simulation of the impact of influence on individuals in social networks using Cell DEVS formalism using CD++. The intention of this study is to propogate information within a group of individuals.

The conceptual model figure is shown below :

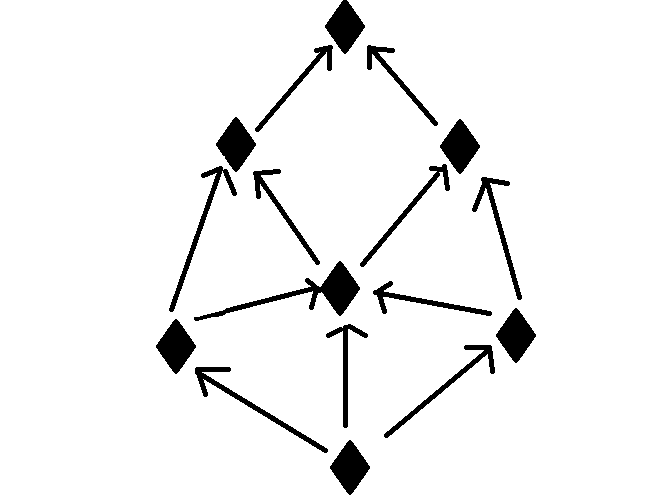


Fig 1. Conceptual Model social network individual impact.

In this model, the message is transmitted to an individual from surrounding individuals. Its upon the individual whether to receive the message or not.

If the person is receiving message, it will make sure the content of the message is proper or not. It will agree upon the data in the after checking its content.

The individual will judge its content depepding upon the opinion of surrounding cells.

* **CELL DEVS** :

Cell DEVS is extension of DEVS formalism, with allowing the implementation of cellular models with specific timing delays. Each cell is defined as atomic model which is integrated to a coupled model depicting a cell space.This model is 2D square grid of 20x20 mesh. A neighborhood is defined for each cell on which the rules will be applied accordingly and the cell will be impacted when the rules are follwed. The cell values taken are 0,1,2. Each cell may represent 3 states : receive, not receive and agree.

There are two rules in this model:

1. If the cells neighborhood is greater than the cell’s initial value, it means that neighborhood will impact the cell to recieve the message. If the cell(individual) receives the message the state changes to 1.
2. Secondly, after receiving the message, the individual should upon agreeing that message received has correct information. For this, the cells neighborhood is given the condition such that if opinion of neighbors are influencing the indiividual, it will agree upon the message content and change its state to 2.
3. If any of both doesn’t happen, the cell will be in its initial state.

The resulting simulation shows that some individuals will recieve and agree with the message and the content respectively and some may not.

**Modifications done in sample paper** : Comparing with sample paper, its given that the individual will receive the message depepnding upon the message content. I have added one more state to this which is when the individual receives a message depending upon the given transition function’s first rule, it will check the opinion of surrounding neighborhood. If particular neighbors less than 8 and 3 or 4 or 2 neighbors are agreeing to the received content, the individual will hence agree that the received message is right. So the cell state will change from 1 to 2.

* **FORMAL SPECIFICATIONS** :

**Cell-DEVS Definition :**

CD = < X, Y, I, S, θ, N, d, δint, δext, τ, λ, D >

Where,

X = 20

Y = 20

S = { 0, 1,2 }

N = neighborhood = { (-1,0), (0,-1), (0,0), (0,1), (1,0) }

d = 100 ms

τ: N🡪S: Individualimpact\_tau

rule : 1 100 { (0,0) = 0 and trueCount = 1 and trueCount < 2}

rule : 2 100 { (0,0) = 1 and trueCount = 2 or trueCount = 3 or trueCount = 4 and trueCount < 8 }

rule : {(0,0)} 0 { t }

δint: internal transition function which is defined by CD++ automatically

δext: external transition function which is defined by CD++ automatically

λ: output function which is defined by CD++ automatically

**CD++ Implementation :**

[top]

components : Individualimpact

[Individualimpact]

type : cell

width : 20

height : 20

delay : transport

defaultDelayTime : 100

border : wrapped

neighbors : Individualimpact(-1,0)

neighbors : Individualimpact(0,-1) Individualimpact(0,0) Individualimpact(0,1)

neighbors : Individualimpact(1,0)

initialvalue : 0

initialrowvalue : 0 00000000000000000000

initialrowvalue : 1 01000000000000000000

initialrowvalue : 2 00000000000000000000

initialrowvalue : 3 00000000000000000000

initialrowvalue : 4 00000000000000000000

initialrowvalue : 5 00000000000000000000

initialrowvalue : 6 00000000000000000000

initialrowvalue : 7 00000000000000000000

initialrowvalue : 8 00000000000000000000

initialrowvalue : 9 00000000000000000000

initialrowvalue : 10 00000000000000000000

initialrowvalue : 11 00000000000000000000

initialrowvalue : 12 00000000000000000000

initialrowvalue : 13 00000000001000000000

initialrowvalue : 14 00000000000000000000

initialrowvalue : 15 00000000000000000000

initialrowvalue : 16 00000000000000000000

initialrowvalue : 17 00000000000000000000

initialrowvalue : 18 00000000000000000010

initialrowvalue : 19 00000000000000000000

localtransition : Individualimpact\_tau

[Individualimpact\_tau]

rule : 1 100 { (0,0) = 0 and trueCount = 1 and trueCount < 2}

rule : 2 100 { (0,0) = 1 and trueCount = 2 or trueCount = 3 or trueCount = 4 and trueCount < 8 }

rule : {(0,0)} 0 { t }

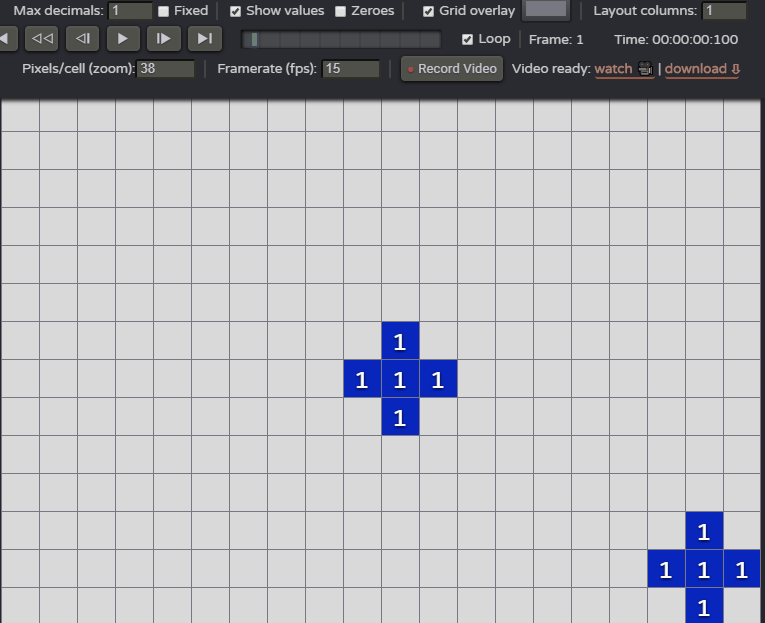
**Explanation of Local Computing function:**

In this function, the condition of how the cell DEVs work is given. How the condition is fullfiled for each cell resulting into state change. The first rule is if the intial value of cell (0,0) is 0, and the truecount is 1 or less than 2 the state of that cell changes to 1 after a delay of 100. This means if one neighbor has received the message the particular cell will also recieve the message. The second rule is if the initial value is 1 and the truecount is one of (2/3/4) but less than 8, it will change the state to 2. This means if 2 or 3 or 4 neighbors are agreeing that the message content is right, the cell will agree that the message content is right and will approve. This rules apply everytime a cell state is changed and when it goes to other cells. If any of these rules does not follow will result into the initial state.

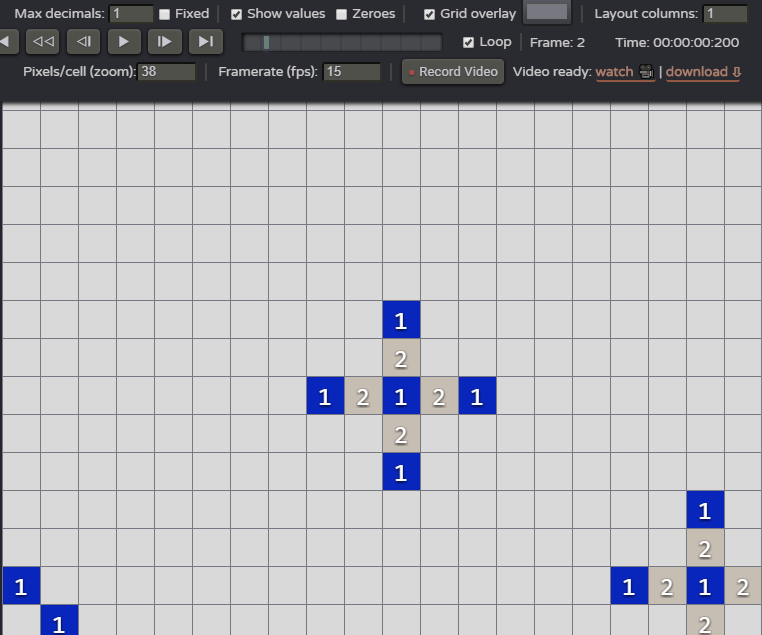
* **Simulation Results:**

1. **Webviewer Results:**

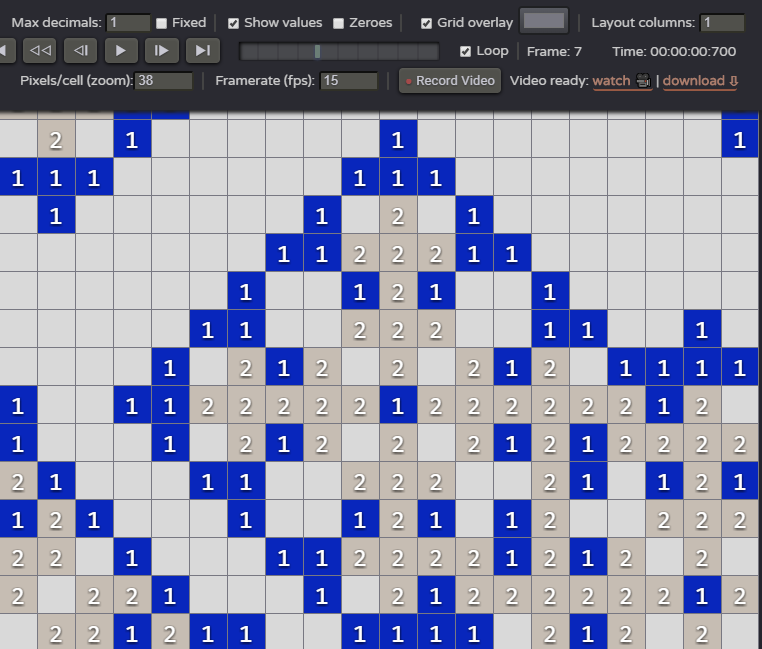
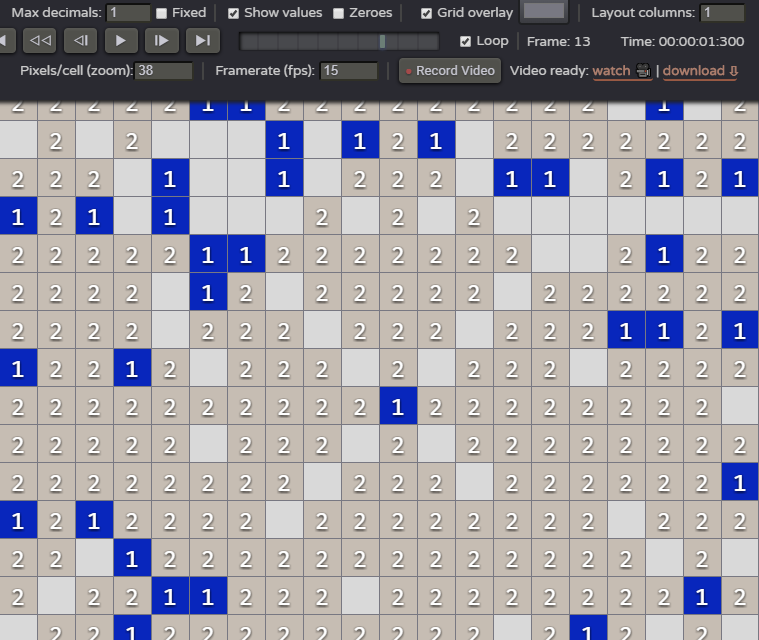
Time : 00:00:00:100

****

Time : 00:00:00:200



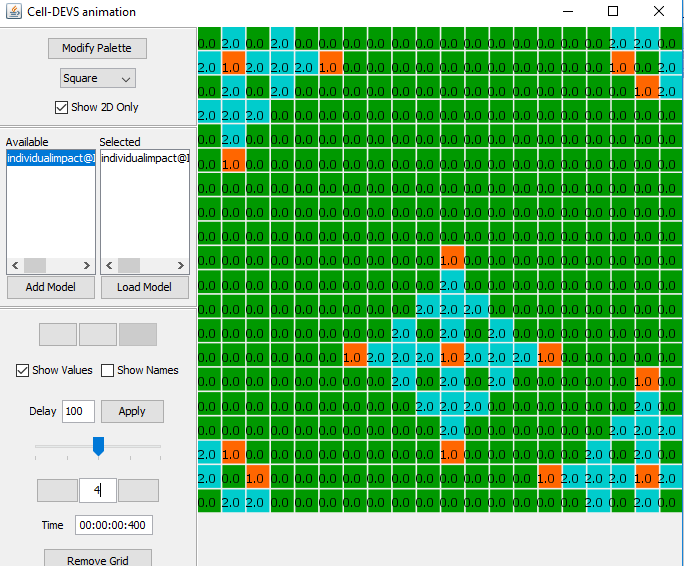
Time : 00:00:00:700 Time : 00:00:01:300

1. **CD++ Modeler Viewer:**

Time : 00:00:00:200 Time : 00:00:01:000



Now I will show the result when input values are changed :

initialrowvalue : 0 00000000000000000000

initialrowvalue : 1 01000000000000000000

initialrowvalue : 2 00000000000000000000

initialrowvalue : 3 00000000000000000000

initialrowvalue : 4 00000000000000000000

initialrowvalue : 5 00000000000000000000

initialrowvalue : 6 00000000000000000000

initialrowvalue : 7 00000000000000000000

initialrowvalue : 8 00000000000000000000

initialrowvalue : 9 00000000000000000000

initialrowvalue : 10 00000000000000000000

initialrowvalue : 11 00000000000000000000

initialrowvalue : 12 00000000000000000000

initialrowvalue : 13 00000000001000000000

initialrowvalue : 14 00000000000000000000

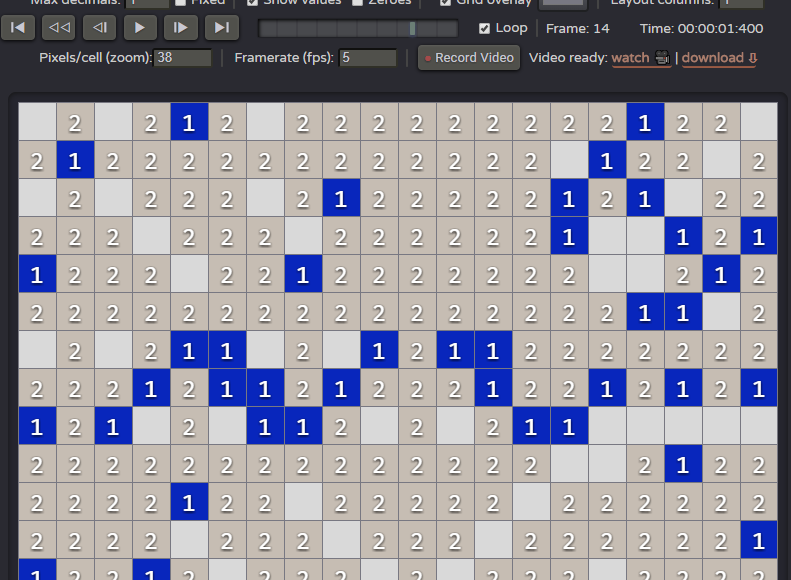
initialrowvalue : 15 00000000000000000000

initialrowvalue : 16 00000000000000000000

initialrowvalue : 17 00000000000000000000

initialrowvalue : 18 00000000000000000010

initialrowvalue : 19 00000000000000000000



Time : 00:00:01:400

**References :**

1. Bakshy, E., Hofman, J. M., MasonW. A. and Watts, D. J. (2011), “Everyone’s an

influencer: quantifying influence on twitter”, In Proceedings of the fourth ACM

international conference on Web search and data mining, pp. 65–74.

2. CELL-DEVS Models, [accessed April 2014], available via:

www.sce.carleton.ca/faculty/wainer/wbgraf/doku.php?id=model\_samples:start

3. Diekmann,O. and Heesterbeek,J.A.P. (2000), Mathematical Epidemiology of Infectious

Diseases, Wiley ser. mathematical and computational biology, England.

4. Ding, W., Pei, L., Li, H., Xi, N., Wang, Y. (2009) “The Effects of Time Delay of

Internet on Characteristics of Human Behaviors”, Proceedings of the 2009 IEEE

International Conference on Networking, Sensing and Control, Okayama, Japan.

5. Dorsey, D. W. and Coovert, M. D. (2003) “Mathematical Modeling of Decision

making: A Soft and Fuzzy Approach to Capturing Hard Decisions”, Human Factors,

Vol. 45, No. 1, pp. 117–135.

6. Faucher, C., Zacharewicz, G., Hamri, E.M., Frydman, C. (2012), “PSYOPS and

CIMIC operations: from concepts to G-DEVS models”, SpringSim (TMS), 42.

7. Fei, L., Ming, Y. and Guobing S. (2007), “Verification of Human Decision Models

in Military Simulations”, Proceedings of the IEEE 2007, The First Asia International

Conference on Modeling & Simulation, pp. 363 – 368.

8. Kubota, N., and Nishida, K. (2006) “Prediction of Human Behavior Patterns

based on Spiking Neurons”, The 15th IEEE International Symposium on Robot

and Human Interactive Communication (RO-MAN06).

9. Seck, M., Frydman, C.S., Giambiasi, N., (2004), “Using DEVS for Modeling and

Simulation of Human Behaviour”, AIS 2004, pp. 692-698.

10. Shang, H. and Wainer, A.G. (2005), “A Model of Virus Spreading Using Cell-

DEVS”, The International Conference on Computational Science 2, volume 3515

of Lecture Notes in Computer Science, Springer, page 373-377.

11. Sun, Z. (2007) “Multi-Agent Based Modeling: Methods and Techniques for Investigating

Human Behaviors”, Proceedings of IEEE International Conference on

Mechatronics and Automation.

12. Wainer, G. (2002), “CD++: a toolkit to develop DEVS models”, Software - Practice

and Experience, pp. 1261-1306.

13. Wainer, G. (2009), “Discrete-Event Modeling and Simulation: a Practitioner’s

approach”. CRC Press.

14. Zacharewicz, G., Frydman, C., Giambiasi, N. (2008), “G-DEVS/HLA Environment

for Distributed Simulations of Workflows”, 84(5), pp. 197–213

15. Zeigler, B.P., Praehofer H., Kim T.G. (2000) “Theory of Modelling and Simulation”.

Editions Academic Press, San Diego.