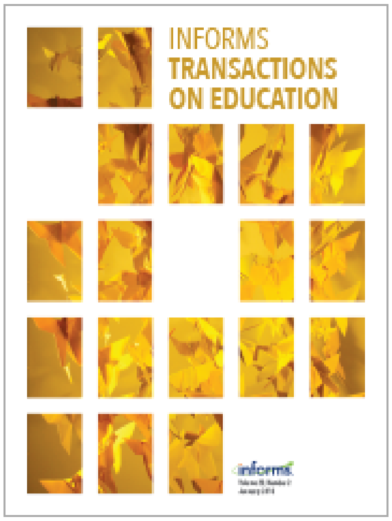


INFORMS Transactions on Education

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Simulating the Spread of a Virus in a Computer Network

Jihad Estrada, Zvi Drezner,

Simulating the Spread of a Virus in a Computer Network

Jihad Estrada And Zvi Drezner
California State University - Fullerton
College of Business andEconomics
Fullerton, CA 92834
jaeestrada@gmail.com
zdrezner@fullerton.edu

Abstract

In this paper we propose a simulation problem that can be used as an exercise in a simulation class using an excel add-on simulation package. Suppose that N objects (computers, people, animals, plants) are interacting in pairs. Every time unit (for example, a second for a computer, a day or a week for people), each object randomly selects another object and contacts it. Suppose that one object is infected with a virus. How long will it take until all objects are infected? How many objects are infected after 5 time units? This process is a simplified model for the spread of AIDS among humans, computer viruses among computers, or any communicable disease among humans, animals, or plants.

1. Introduction

Most of us, during our lifetime, have caught a cold or the flu, and most likely have caught it from the spread of a virus from being in contact with another person with the virus. Most of us have, or know someone who has, contracted a computer virus. Both of these types of viruses spread similarly, with the infected contacting a new host and duplicating itself onto it (Symantec, 1999). A different view of the spread process can be created using a simulation.

Consider N objects with full communication between each pair of objects. At every time unit, an object randomly selects an object and communicates with it. Suppose that one object has a virus. How long will it take, on the average, until the virus infects all the objects? How many objects have the virus after t time units?

There are two cases for modeling such a problem: a synchronized system and a non-synchronized system. (i) A synchronized system means that every object communicates with the randomly selected object at exactly the same time. (ii) In the asynchronous case, communications are accomplished at different times but we still assume that every object communicates every time unit.

In Barak and Drezner (1986) it is shown that in a synchronized system it takes about $1.693\log_2 N$ time units until all objects get the virus. In Amar et al. (2006) the question about how many objects are infected following a given number of time units is solved both for the synchronized and non-synchronized cases.

It is possible that not all objects are responsive. In this case, an object selects an object from all N objects but the target object will not receive it if it is non-responsive. For the synchronized case it was shown in Barak and Drezner (1986) that it takes approximately $[1.693+1.414(1-n/N)]\log_2 N$ time units until all responsive objects are infected when n out of N objects are responsive. All these approximations assume a large N and represent asymptotic behavior when the number of objects tends to infinity.

In this paper we propose a simulation model for the synchronized system to be used in a simulation class using any Excel add-on simulation package (see Appendix). The construction of the Excel spreadsheet that calculates the required values (without using VBA) is a very challenging problem even for an advanced class. Once the spreadsheet is constructed, the simulation itself is quite straightforward.

 © INFORMS ISSN: 1532-0545

2. The Simulation Problem

Consider 50 objects which communicate with one another. One of the objects is infected by a virus. Every time unit an object randomly selects an object and communicates with it. If the object is infected, the target object becomes infected with the virus. How long will it take until all objects are infected? How many objects are infected, on the average, after 5 time units?

In Barak and Drezner (1986) the probability that k objects are infected after j time units is explicitly calculated. Let $P_j(k)$ be the probability that k objects are infected after j time units. Then,

$$P_{j+1}(k) = \sum_{m=0}^{[k/2]} \binom{N-k+m}{m} R(m,k-m) P_j(k-m)$$

where

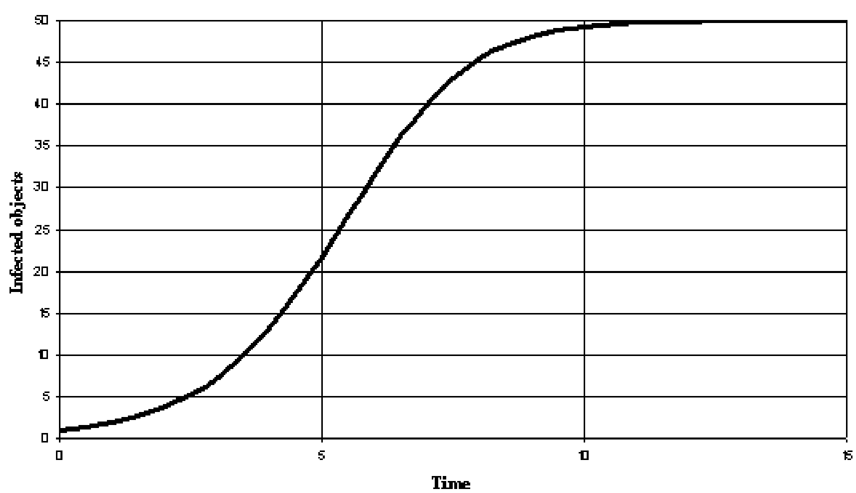
$$R(m,i) = \sum_{l=m}^{i} \binom{i}{l} \left(\frac{m}{i}\right)^{l} \left(\frac{i}{N}\right)^{i} Q(m,l)$$

and where

$$Q(m,l) = \sum_{i=1}^{l-m+1} \binom{l}{i} \left(\frac{1}{m}\right)^{i} \left(1-\frac{1}{m}\right)^{l-i} Q(m-1,l-i)$$

and $Q(1,l) = 1$.

Figure 1: Expected number of infected objects



For the case that not all objects are active the reader is referred to Barak and Drezner (1986). These formulas can be easily programmed by VBA. See Excel file[(1)] which includes three sheets named P R and Q. You must first execute the program in Q to obtain *Q(m,l)* , then the program in R to obtain *R(m,i)* , and then the program in P to obtain the probabilities. The expected number of infected objects is calculated in column B of the spreadsheet P and is graphed in Figure 1. Note that the expected number of infected objects after 5 time units is 21.69533402.

This model is a simplified representation of a spread of an AIDS virus, or any other communicative disease between humans, animals, or plants. A spread of a computer virus is also quite similar. In reality there may be many more objects, the selection of a target object may not be random, and each time unit, more than one communication may occur. Our model illustrates the rapid spread of an infectious process. It is conceptually not more difficult to code a simulation with many more objects, and more than one communication each time unit. However, such an exercise is unnecessarily more cumbersome and will not serve as a constructive class exercise.

Creating a spreadsheet to calculate these two values in preparation for the simulation is very challenging.

(1) http://ite.pubs.informs.org/Vol6No2/DreznerEstrada/Calculation.xls

First, one should make sure the proper statistical packages are present. If using Microsoft Excel, go to "Tools" then "Add-Ins" and make sure the "Analysis ToolPak" and "Analysis ToolPak - VBA" are checked off and added in. One needs to know the RANDBETWEEN command for generating a random integer number in a range (this should be easy; after all it is a simulation class). It is important to note that the command RANDBETWEEN is available only if the data analysis feature is added in. If it is not, the RANDBETWEEN(1,50) command should be replaced by INT(RAND()*50)+1. Our proposed spreadsheet (See Excel file [2] for our proposed solution) utilizes less commonly known Excel commands: VLOOKUP and ISERROR. VLOOKUP searches for a value in the first column of a table and returns the value in a different column of the table in the same row where the searched value is found. If the value is not found in the first column of the table, the VLOOKUP command returns an error code. The ISERROR command is usually used in conjunction with an IF command. It returns the logical value "true" if the value in the ISERROR command is an error value and returns the logical value "false" if the value is not an error. Once these two commands are explained to the students, the good students in the class should be able to construct the spreadsheet.

2.1. A Variation on the Theme

Now suppose that the last 10 objects are not active (or have some sort of anti virus immunization). That means that an object can select them as a target but they do not get infected. We ask the same two questions: How long will it take until all active objects are infected? How many active objects are infected, on the average, after 5 time units?

2.2. Explanation of the Construction of the Spreadsheet

The attached spreadsheet [3] has two sheets. The first one is programmed for the case that all 50 objects are active. A "0" in the cell means that the object is "clean" and a positive number means that it is infected. The number in the cell of an infected object is the target object. At time 0 the first object is "infected". Every time unit, an infected object sends the virus to a randomly selected object. Conditional formatting is used to indicate "clean" objects with a yellow background and "infected" objects with a blue background. Note that clicking the F9 key recalculates the spreadsheet and the colors of the objects automatically change.

In column C the first object (cell C3) selects an object to send the virus to by the command =RANDBETWEEN(1,50) and all other cells have "0" meaning no virus. Recall that Data Analysis must be installed for the command RANDBETWEEN to be activated.

In Column D we enter (in cell D3 and dragged down):

=IF(C3>0,RANDBETWEEN(1,50),IF($C$3=B3,RANDBETWEEN(1,50),0))

which selects an object either when the object was infected before (is object #1) or it is the destination selected by the first object (in cell $C3$).

In all the cells from Column E and higher we entered the command (entered in E3 and dragged down and to all columns from E to the right):

=IF(ISERROR(IF(D3 > 0, RANDBETWEEN(1,50), IF(VLOOKUP(B3,$D$3:$D$52,1,FALSE)=0, RANDBETWEEN(1,50),0))) = TRUE, 0 ,RANDBETWEEN(1,50))

This command is quite complex. It basically checks whether the present object was selected by an infected object or was infected before. If this is the case a target object is selected. Otherwise the value for this object remains "0".

In row 53 we count how many objects are infected and in row 54 we enter 0 if the count is less than 50 and 1 if it is. In cell C56 we count how many zeroes are in row 54 which gives the first time unit at which all objects are infected. We also highlighted cell H53 which contains the number of infected objects after 5 time units.

The second spreadsheet assumes that objects 41-50 are not active and thus they simply have zeros for all columns (but can be selected by other objects).

(2) http://ite.pubs.informs.org/Vol6No2/DreznerEstrada/Simulation.xls
(3) http://ite.pubs.informs.org/Vol6No2/DreznerEstrada/Simulation.xls

2.3. Results

By the analysis in Barak and Drezner (1986) we can estimate the answers to the first question "how long will it take until all objects are infected". When all objects are active, *N*=50, which yields $1.693\log_2 50=9.6$. Note that this estimate is based on using a large *N* which may not hold for *N*=50. When only *n*=40 objects are active (80%), the estimate for the time when all active objects are infected is: $[1.693+0.2x1.414]\log_2 50=11.2$ time units.

When the simulation was repeated 10,000 times, we got the following results (the mean standard error is given in parentheses):

Outcome	Result
All objects are active	
All objects infected	9.70 time units (.01)
Objects infected after 5 time units	21.69 nodes (.03)
40 objects are active	
All active objects infected	11.25 time units (.02)
Active objects infected after 5 time units	13.85 nodes (.04)

The results for the time it takes for all objects to be infected are amazingly close to the results of the theoretical analysis (Barak and Drezner, 1986). The theoretical results are 9.6 and 11.2 while the simulation results are 9.70 ± 0.01 and 11.25 ± 0.02, respectively. The calculated expected number of infected objects after 5 time units when all objects are active is 21.6953 while the simulation result is 21.69 ± 0.03. These results demonstrate how fast a virus is spreading in a population. This is true for real diseases transmitted from one human (or animal or plant) to another and demonstrates the extreme danger of the spread of viruses either among computers through the Internet of for infectious diseases in a population.

References

Amar L., A. Barak, I. Peer, and Z. Drezner, (2006), "Gossip Algorithms for Maintaining a Distributed Bulletin Board with Guaranteed Age Properties," working paper, Department of Computer Science, The Hebrew University, Jerusalem. Submitted for publication.

Barak A. and Z. Drezner, (1986), "An Asynchronous Algorithm for Scattering Information in a Multicomputer System," *Journal of Parallel and Distributed Computing* , Vol. 3, pp. 344-351.

Symantec. (1999), "What is a Virus?", Accessed (1/1/06), http://service1.symantec.com/SUPPORT/nav.nsf/aab56492973adccd8825694500552355/024c927836400f528825675100593 eb2?OpenDocument&src=sec_web_nam

Appendix

Appendix Add-ons Simulation Software

In this appendix we list in alphabetical order some of the commonly used simulation software packages that are Excel Add-ons.

1. @Risk [(4)]
2. Crystal Ball [(5)]
3. SIMULACION [(6)]
4. XLSIM [(7)]

(4) http://www.hallogram.com/@risk/
(5) http://www.decisioneering.com/
(6) http://www.cema.edu.ar/~jvarela/index_eng.htm
(7) http://analycorp.com/software.htm

