A Simple Cellular Automaton Model for Influenza A Viral Infections

Catherine Beauchemin,
Department of Physics,
University of Alberta

January 7, 2004
Overview

I will go over the following:

• The project
• The rules of the cellular automaton
• The calibration of the model
• Results
• Future projects
The Project

Project:
- Build a simple CA model which can reproduce experimental results of an uncomplicated viral infection with Influenza A.

Motivations:
- Current cellular automaton (CA) models are too complex to reveal immune mechanisms and not well calibrated enough to perform as immune simulators;
- Differential equation models of viral infection are typically spatially homogeneous and few include delays.

Why a CA?
- A CA offers a natural description of the physical system. There is good correspondance between the CA’s parameters and measurable quantities.

Ultimate objective:
- Explore whether spatial inhomogeneities and age classes affect the development and outcome of a viral infection.
The CA Model: The Epithelial Cells

An epithelial cell can be in any of five states: healthy, infected, expressing, infectious, or dead. Transitions between epithelial cell states occur as follows:

- Epithelial cells of all states become dead when they are older than \( \text{CELL\_LIFESPAN} \).
- A healthy epithelial cell becomes infected with probability \( \frac{\text{INFECT\_RATE}}{(8 \text{ nearest neighbours})} \) for each infectious nearest neighbour.
- An infected cell becomes expressing, i.e. begins expressing the viral peptide, after having been infected for \( \text{EXPRESS\_DELAY} \).
- An expressing cell becomes infectious after having been infected for \( \text{INFECT\_DELAY} > \text{EXPRESS\_DELAY} \).
- Infected, expressing, and infectious cells become dead after having been infected for \( \text{INFECT\_LIFESPAN} \).
The CA Model: The Epithelial Cells

Continued...

- Expressing and infectious cells become dead when “recognized” by an immune cell. See below for the meaning of “recognition.”

- A dead cell is revived at a rate $\text{DIVISION\_TIME}^{-1} \times \# \text{healthy}/\# \text{dead}$. When revived, a dead cell becomes a healthy cell or an infected cell with probability $\text{INFECT\_RATE}/(8 \text{ nearest neighbours})$ for each infectious nearest neighbour.

A simulation is initialized with each epithelial cell being assigned a random age between 0 and $\text{CELL\_LIFESPAN}$ inclusively. All epithelial cells start in the healthy state with the exception of a fraction $\text{INFECT\_INIT}$ of the total number of epithelial cells which, chosen at random, are set to the infected state.
The CA Model: The Immune Cells

An immune cell can be in any of two states: virgin or mature. A virgin cell is an immune cell that has no specificity. A mature cell is an immune cell that has either already encountered an infected cell or has been recruited by another mature immune cell.

- Immune cells move randomly on the CA lattice at a speed of one lattice site per time step.
- An immune cell is removed if it is older than IMM_LIFESPAN.
- An encounter between an immune cell and an expressing or infectious epithelial cell requires the immune cell to be in the same site as the expressing or infectious cell.
- A virgin immune cell becomes a mature immune cell if the lattice site it is occupying is in the expressing or infectious states.
The CA Model: The Immune Cells

Continued ...

- A mature immune cell occupying an expressing or infectious lattice site “recognizes” the epithelial cell and causes it to become dead.
- Each recognition event causes RECRUITMENT mature immune cells to be added at random sites on the CA lattice after a delay of RECRUIT_DELAY.
- Virgin immune cells are added at random lattice sites as needed to maintain a minimum density of BASE_IMM_CELL virgin immune cells.

A simulation is initialized with a density of BASE_IMM_CELL virgin immune cells at random locations on the CA lattice, each with a random age.
The CA Model: The Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grid_width</td>
<td>440</td>
<td>Width of the grid (cells)</td>
</tr>
<tr>
<td>grid_height</td>
<td>280</td>
<td>Height of the grid (cells)</td>
</tr>
<tr>
<td>FLOW_RATE*</td>
<td>6 t.s. · h⁻¹</td>
<td>Speed of immune cells (time step/hour)</td>
</tr>
<tr>
<td>CELL_LIFESPAN</td>
<td>380 h</td>
<td>Lifespan of a healthy epithelial cell</td>
</tr>
<tr>
<td>DIVISION_TIME*</td>
<td>12 h</td>
<td>Duration of epithelial cell division</td>
</tr>
<tr>
<td>INFECT_INIT*</td>
<td>0.01</td>
<td>Fraction of cells initially infected</td>
</tr>
<tr>
<td>INFECT_LIFESPAN</td>
<td>24 h</td>
<td>Lifespan of an infected epithelial cell</td>
</tr>
<tr>
<td>EXPRESS_DELAY</td>
<td>4 h</td>
<td>Delay from infected to viral expression</td>
</tr>
<tr>
<td>INFECT_DELAY</td>
<td>6 h</td>
<td>Delay from infected to infectious</td>
</tr>
<tr>
<td>BASE_IMM_CELL</td>
<td>1.5 × 10⁻⁴</td>
<td>Initial density of immune cells</td>
</tr>
<tr>
<td>IMM_LIFESPAN</td>
<td>168 h</td>
<td>Lifespan of an immune cell</td>
</tr>
<tr>
<td>RECRUIT_DELAY</td>
<td>7 h</td>
<td>Delay between the recruitment call and the addition of immune cells</td>
</tr>
</tbody>
</table>
The CA Model: The Free Parameters

All but the 2 parameters below were extracted from


<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECRUITMENT</td>
<td>0.25</td>
<td>Number of immune cells recruited when one recognizes the virus</td>
</tr>
<tr>
<td>INFECT_RATE</td>
<td>2 h(^{-1})</td>
<td>Rate of infection of neighbours</td>
</tr>
</tbody>
</table>
Characteristics to fit

1. The infection should peak on day 2 (48 h).

2. Over the course of the infection, the fraction of epithelial cells that are dead should be as follows:
   
   (a) 10% on day 1 (24 h);
   
   (b) 40% on day 2 (48 h);
   
   (c) 10% on day 5 (120 h).

3. From Bocharov et al. 1994, virus concentration should decline to inoculation level on day 6 (144 h). From another source, experimental data recovered from 8 volunteers indicated that virus shedding persisted for $5 \pm 2$ d (72–168 h).

4. The number of immune cells should peak anywhere between day 2 (macrophages’ peak) and day 7 (cytotoxic T cells’ and B cells’ peak) (48 h–168 h).

5. At their peak, the number of B cells, helper T cells, and cytotoxic T cells should be 100-fold greater than their normal concentration, while that of plasma cells should be $10^4$-fold greater.
Results

![Graph showing % total cells over time (h)]
Varying FLOW RATE

Varying from 2 to 20 time steps per hour
Varying DIVISION TIME

Varying from 7 h to 24 h

% total cells

0 24 48 72 96 120 144 168 192 216 240 264 288

time (h)

0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

% total cells
Varying **INFECT_INIT**

Varying from 0.001 to 0.1
Varying RECRUITMENT

Varying from 0.05 to 1.25 cells added per positive recognition
Varying INFECT RATE

Varying from 1 to 4 healthy neighbour per infectious cell
Future Projects

Here, we have introduced a CA model for an uncomplicated viral infection. Once parameterized for the particular case of influenza A, our CA model is sophisticated enough to reproduce the basic dynamical features of a typical uncomplicated viral infection with influenza A.

One expects a 7 state variable and 12 parameter model to be able to match 7 dynamical features, however, all but 5 parameters of this model are sufficiently-well bound by physiological data they cannot be used to tune its behaviour. Remarkably, our model’s agreement with the experimental dynamical characteristics compares well to that of the 13 variable and 60 parameter model presented in Bocharov et al. 1994.

In future work, we will make use of the CA model introduced here to investigate various theoretical aspects of viral infections, for example the extent to which such things as spatial inhomogeneities and age classes can affect the evolution and outcome of a viral infection.
What is MASyV

MASyV stands for Multi-Agent System Visualization. It is a powerful general purpose user interface accompanied by a message passing library. It enables one to write cellular automaton simulations programs and visualize them with very little additional time and effort. The simulation (client) and the user interface (server) interact via UNIX domain sockets using the message passing library. Simulation runs can be saved as movies through the use of transcode.

MASyV requires:

- GTK+ 2.0 or higher
- an OpenGL compatible library
- as well as GtkGlExt, an OpenGL widget for GTK+

MASyV comes with a few client modules which stand as good examples of how to write a client module and how to use the message passing library.

MASyV is distributed under the GNU General Public License and is hosted on SourceForge (http://masyv.sourceforge.net).
Visualizing the Simulations

An example simulation...
Where to find information and links:

MASyV on Sourceforge and movie files from ma.immune
  
  http://masyv.sourceforge.net

The article on Influenza A used to parametrize the model


A good introduction to immunology for physicists