School of Physics

Modelling & Visualisation in Physics
Checkpoint 2

SIR Model Simulation

Richard Hanes
February 2005

Lecturer: Dr. Graeme Ackland

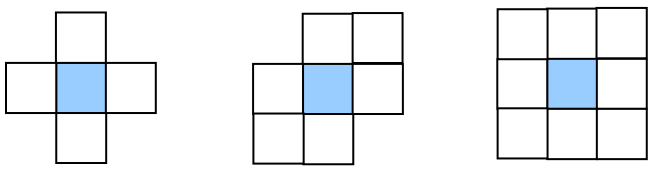
Introduction

The SIR model is a cellular automata simulation of the spread of disease through a population or the spread of a fire in a forest. Each element in the system being studied can be in one of three states, healthy, infected or immune. Each element in the system is updated every time step according to this set of rules:

1. If a healthy element has any neighbours which are infected, it becomes infected with probability p, otherwise it stays healthy.
2. An infected element becomes immune.
3. If an immune element has no infected neighbours, it becomes healthy with probability q, otherwise it stays immune.

A forth state is also used which is an element with permanent immunity; this is not affected by any of the update rules.

The nearest neighbours used when processing the rules can be four, six or eight corresponding to the schematic below.



4 neighbours 6 neighbours 8 neighbours

Each of the neighbours has the same affect on the target; the six neighbours configuration corresponds to a hexagonal lattice.

Two methods for processing these rules have been implemented, simultaneous update and stochastic update.

Simultaneous update involves creating a copy of the system once it has been initialised, looking at the state of the first system and updating the elements on the second system. On the next iteration the simulator looks at the second system and updates the first. This procedure ensures there is no preferred direction to the spread of infection.

Stochastic update picks a random element every time step and processes the rules for that element. Note that there is no guarantee that each element will be processed the same number of times; these two methods produce different results.

Program Usage

The program is executed from it resident directory after it has been compiled, with the command '`java ca`'. Once it has initialised the user is presented with a screen similar to figure 1.

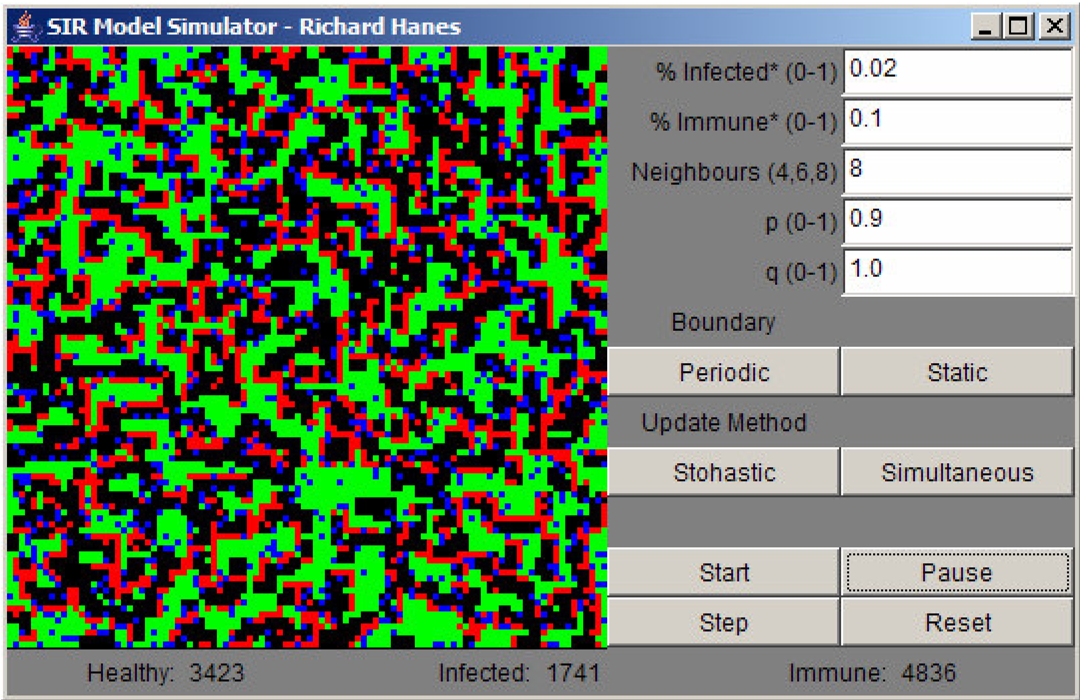


Figure 1: A screenshot of the SIR Model Simulator

The display on the left shows the current state of the simulation. Healthy elements are green, infected red, immune black and permanently immune blue.
The control panel on the right contains field where the desired simulation variables can be changed. The '% Infected*' and '% Immune*' boxes are only re-read when the simulation is reset. All the other boxes can be changed while the simulation is running by entering the new value and pressing the enter key on the keyboard.

Infected	Proportion of infected elements (0.0 – 1.0)
Immune	Proportion of permanently immune elements (0.0 – 1.0)
Neighbours	Number of nearest neighbours used to process rules (4, 6, 8)
p	Probability of infection spreading (0.0 – 1.0)
q	Probability of losing immunity (0.0 – 1.0)

The Step button is only active when the simulation is paused and allows the user to view individual steps of the simulation.

The info bar at the bottom displays the current number of Healthy, Infected and Immune (permanent and otherwise) elements in the system. It is updated once every 10,000 element iterations (i.e. once every sweep through the system).

Results

All results were taken on a system of size 100 by 100 with no periodic boundary conditions.

This size was chosen as it gives a good balance of computational speed with a low probability of absorbing states where there should not be.

Several experiments were run with periodic boundary conditions (PBC) and produced results very similar to static boundary conditions. As PBCs are computationally expensive they were not used for these experiments. Physically it seems more realistic to have a static boundary as we are modelling the spread of disease in a finite population.

In calculating the ensemble average population values, each system was allowed to come to equilibrium before measuring to obtain more accurate results. The amount of equilibration time was found to be 1,000,000 individual element updates for the 100 by 100 system.

Experiment 1: What level of immunity is required to stop an epidemic?

This experiment measured the average population infected (after equilibration) as a function of p, the probability of infection. Runs were performed for simultaneous and stochastic update on all three lattice types.

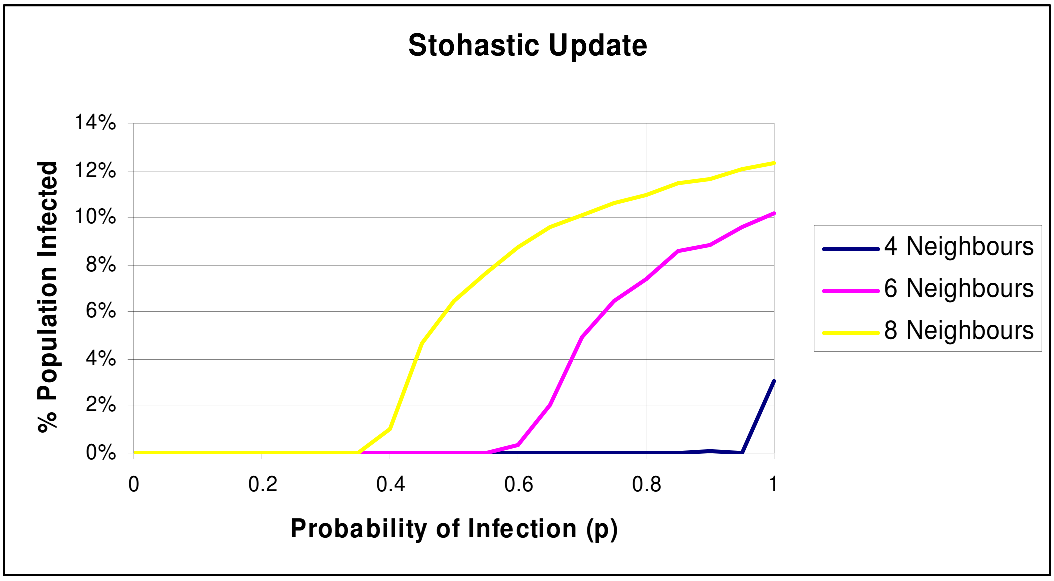


Chart 1: Percentage infected as a function of p on a system updated stochastically

Discussion

Chart 1 show that the percentage of the population infected has a lower cut off limit, below which the infection is wiped out. The cut off values give the level of immunity required to stop an epidemic in each case (1- p):

Number of Neighbours	Level of Immunity Required
4	15% Population
6	50% Population
8	65% Population

The level of immunity required is much higher for 8 neighbours as one infected element can in theory infect eight others, with 4 neighbours it could only infect four.

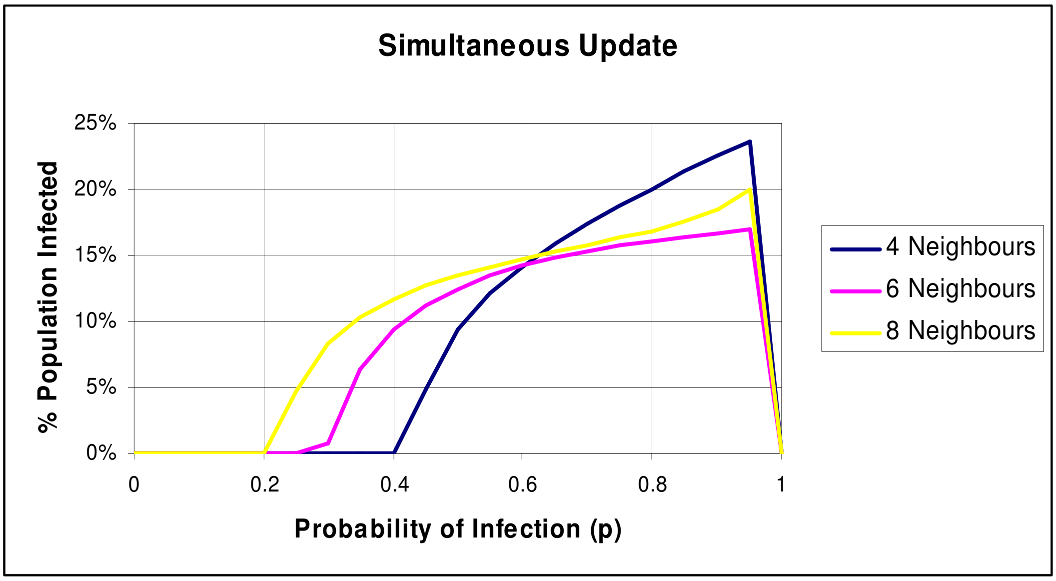


Chart 2: Percentage infected as a function of p on a system updated simultaneously

Discussion

The required level of immunity is given in the table below:

Number of Neighbours	Level of Immunity Required
4	60% Population
6	75% Population
8	80% Population

Chart 2 show a much higher immunity required than the stochastic system. Simultaneous updating allows each infected element to attempt to infect its neighbours. Stochastic updating picks elements randomly so infected elements may not get the chance to infect their neighbours, reducing the required immunity.

When the probability of infection is 1.0, the infection completely dies out on all three lattices. This is because it consumes all healthy elements as it sweeps through the system surrounding itself with immune elements and dying out (fig 2). If p is reduced slightly, not all healthy elements are infected in the initial sweep and the infection survives.

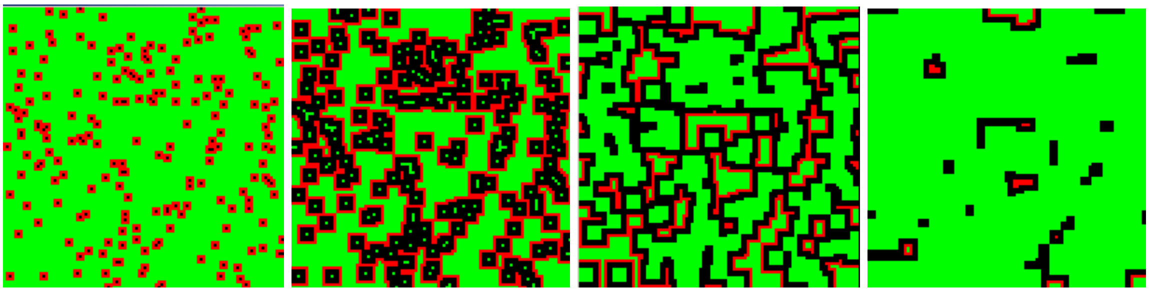


Figure 2: Four screenshots from the simulator showing an infection wipe itself out

Experiment 2: What fraction of permanent immunity is required to stop the epidemic?

These experiments were performed by varying the number of permanently immune elements and recording the average proportion of the population infected. both p and q were held constant. The simulation was run using 8 neighbours and using both the simultaneous and stochastic update methods.

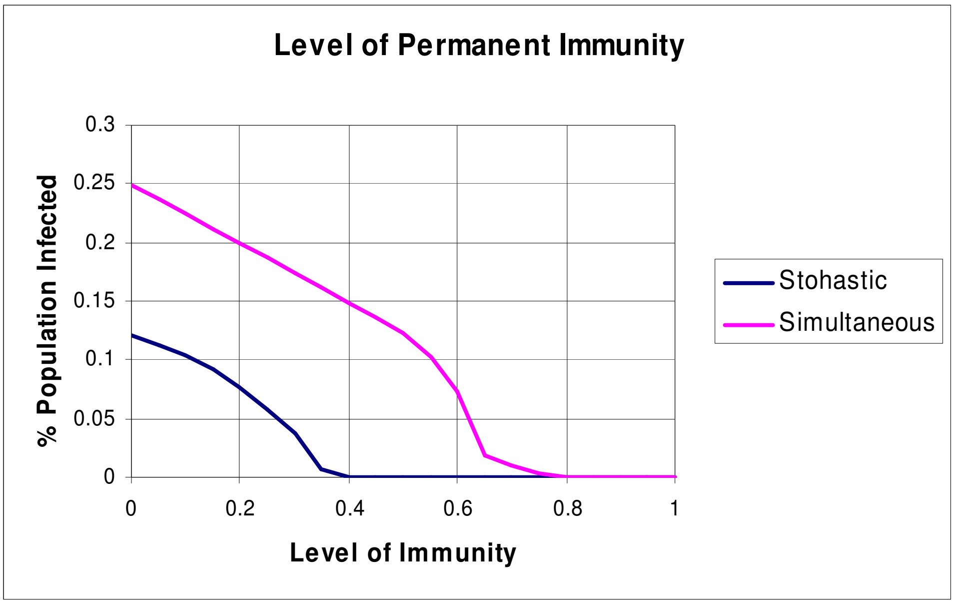


Chart 3: Level of % population infected as a function of permanent immunity

Discussion:

As the level of permanent immunity increases, the % infected decreases, but it does so much more for the stochastic updates than the simultaneous. The spread of infection using simultaneous updates acts like a wave (fig 3) which diffracts around any permanently immune elements, so it is affect much less than the random choice method.

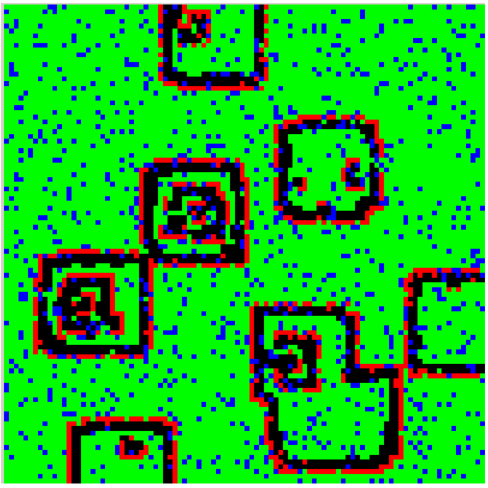


Figure 3: Screenshot showing the wave fronts of infection spreading in a simultaneously updated system.

Experiment 3: Plot the "Phase Diagram" for the System

These experiments were performed using a non graphical version of the simulator to improve the speed. There are four diagrams, two for the stochastic system (one showing % infected, one the % immune) and two for the simultaneous system. The level of permanent immunity was set to zero for these experiments and p and q were varied in steps of 0.05.

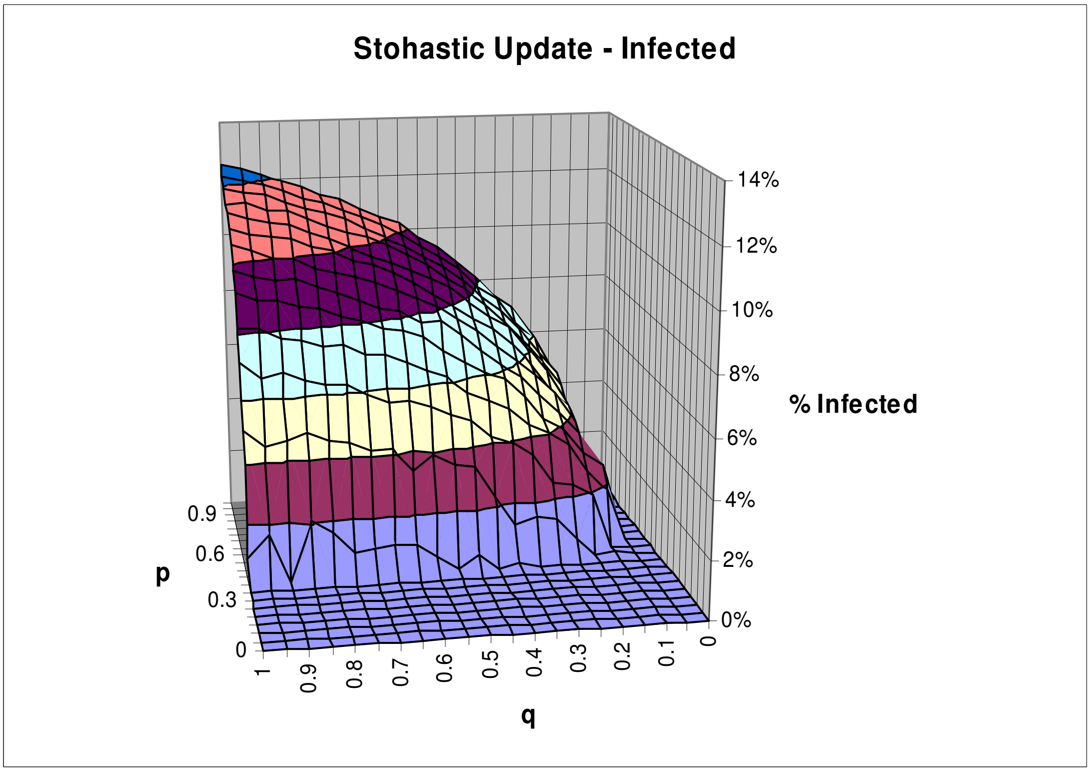


Chart 4: Phase diagram of the Stochastic System showing % infected

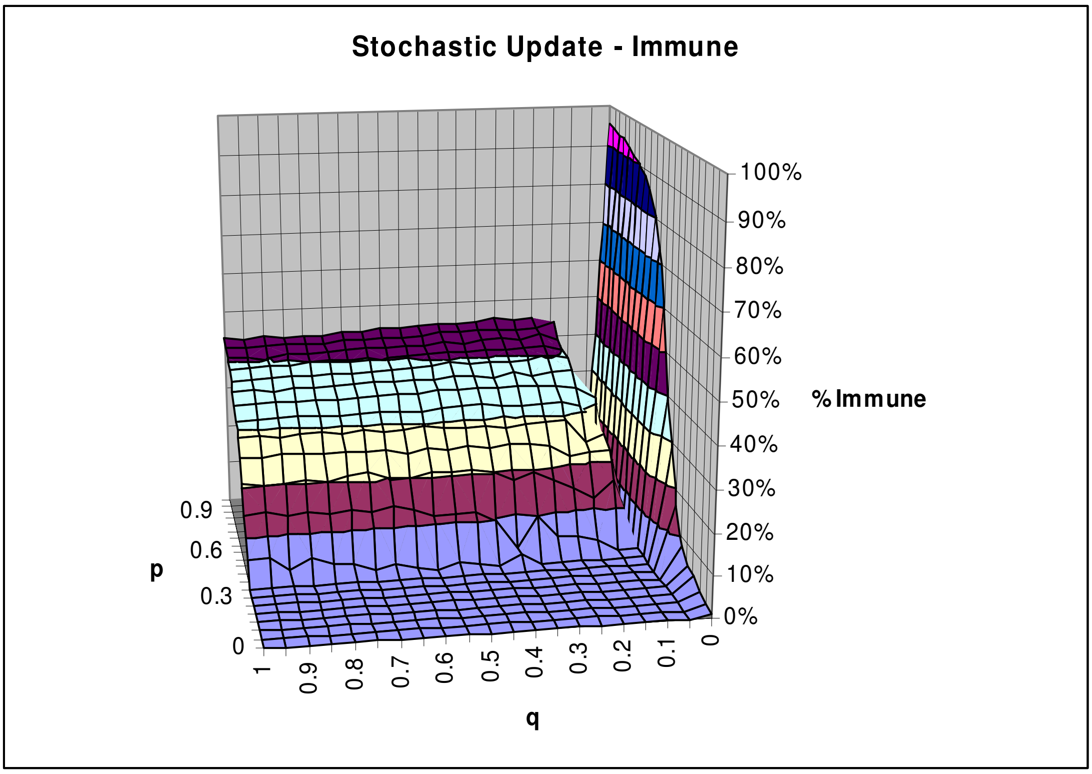


Chart 5: Phase Diagram of the stochastic system shown the % immune

Discussion

The infected phase diagram (Chart 4) shows that the maximum infection is at the maximum p and q. It decreases more rapidly with p and cuts off at p = 0.4, as discussed above.

The Immune phase diagram (Chart 5) shows the % of immune elements hit a maximum at p = 1.0 and q = 0.0. This corresponds to the infection spreading and all the elements becoming permanently immune. At q = 0.05, there is a sharp drop, the infection is wiped out and after enough time all the elements lose their immunity

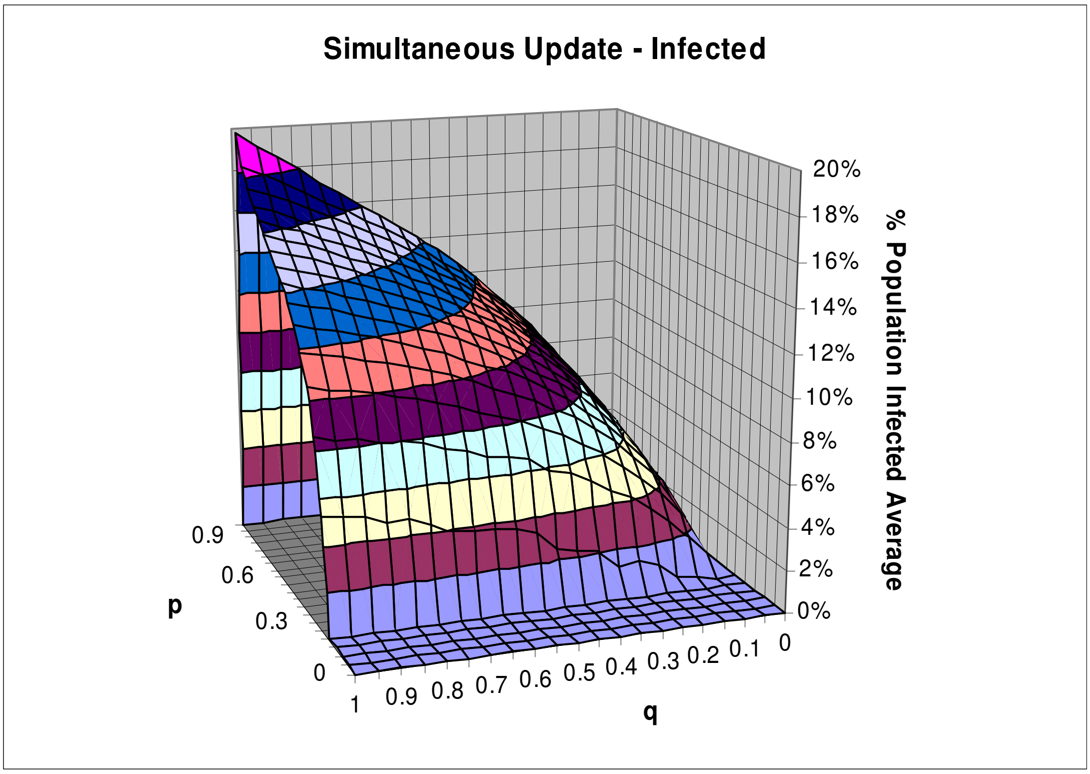


Chart 6: Phase Diagram of the simultaneous system shown the % infected

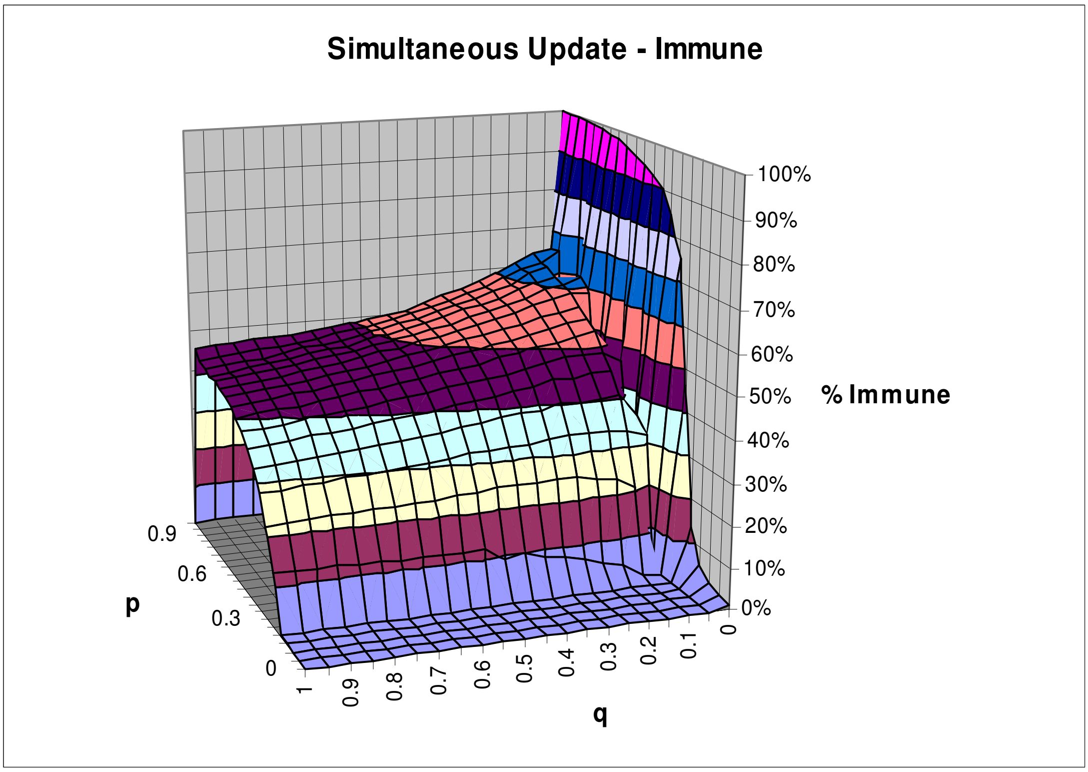


Chart 7: Phase Diagram of the simultaneous system shown the % immune

Discussion

The simultaneous phase diagrams are similar to the stochastic, with the cut-off shifted to p = 0.2. Note that the % infected and immune drops to zero at p = 1.0, as discussed above.

Conclusion

The results from these experiments show that, in the simplest case, the required level of immunity to stop an epidemic is somewhere between 80% and 65% of the total population. The actual value would be much more complicated to calculate. This does raise the question, which update method is more relevant to the physical world, stochastic or simultaneous.

Stochastic updates reflect the probabilistic nature of the universe, everything in nature follows (as far as we know) the laws of Quantum Mechanics. Simultaneous updates reflect the macroscopic order in the universe; on a large scale everything seems deterministic. I could not decide which was better, so implemented both.