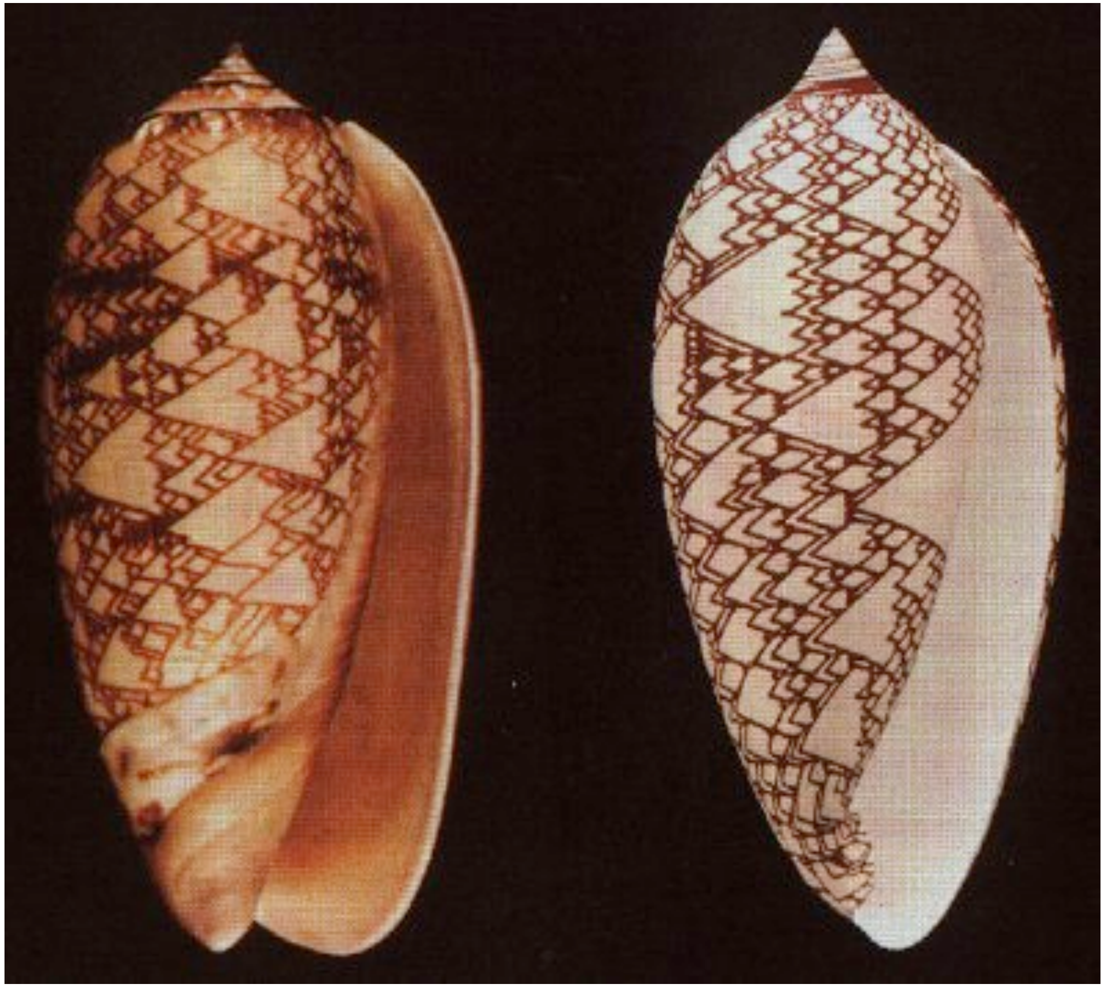
Modelling Complex Systems

David J. T. Sumpter
Qi Ma

What is a complex system?

What is a complex system?



Meinhardt, H. 1995 *Algorithmic Beauty of Sea Shells*

What is a complex system?

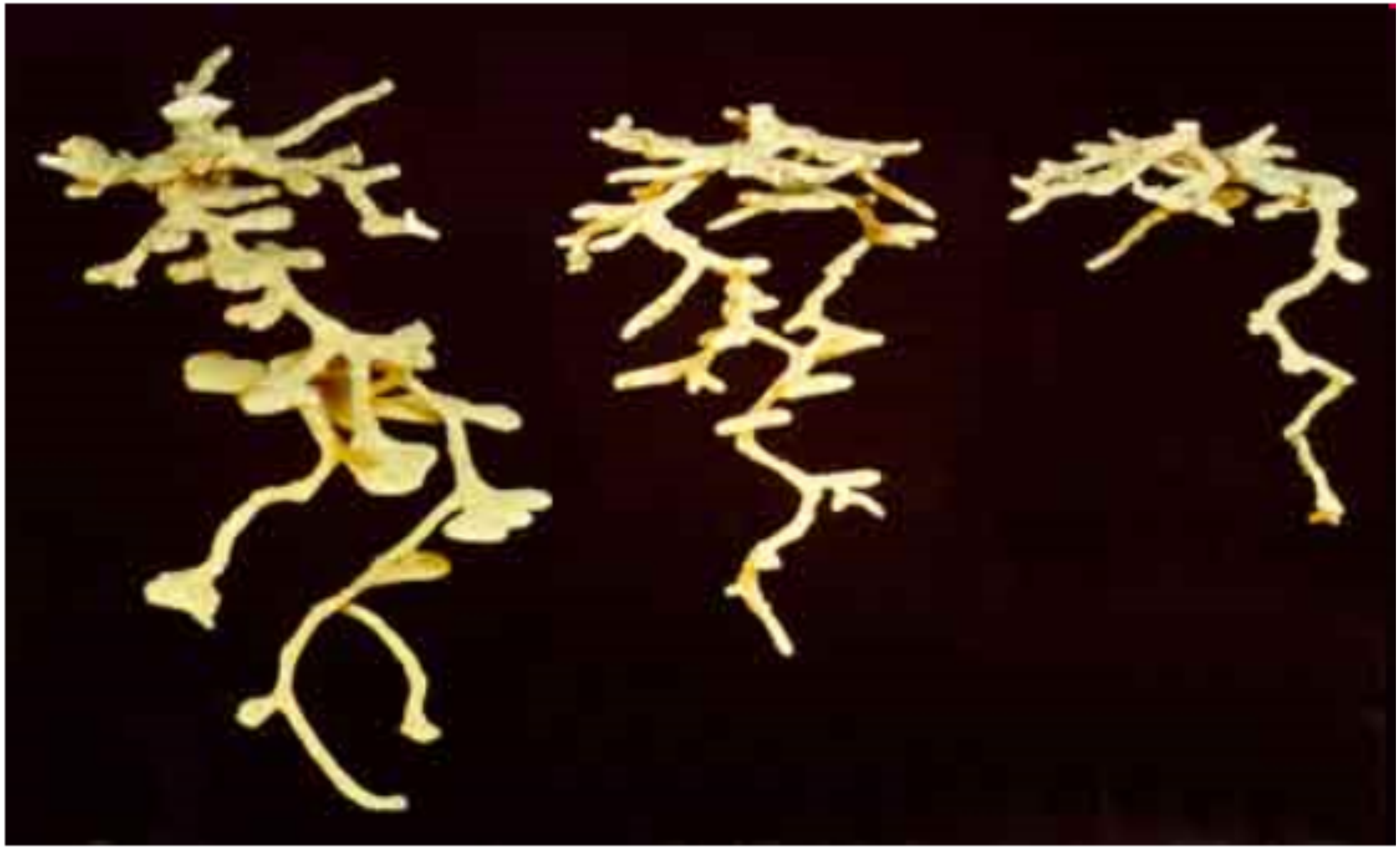
What is a complex system?

STARFLAG project: http://angel.elte.hu/starling/index.html

What is a complex system?

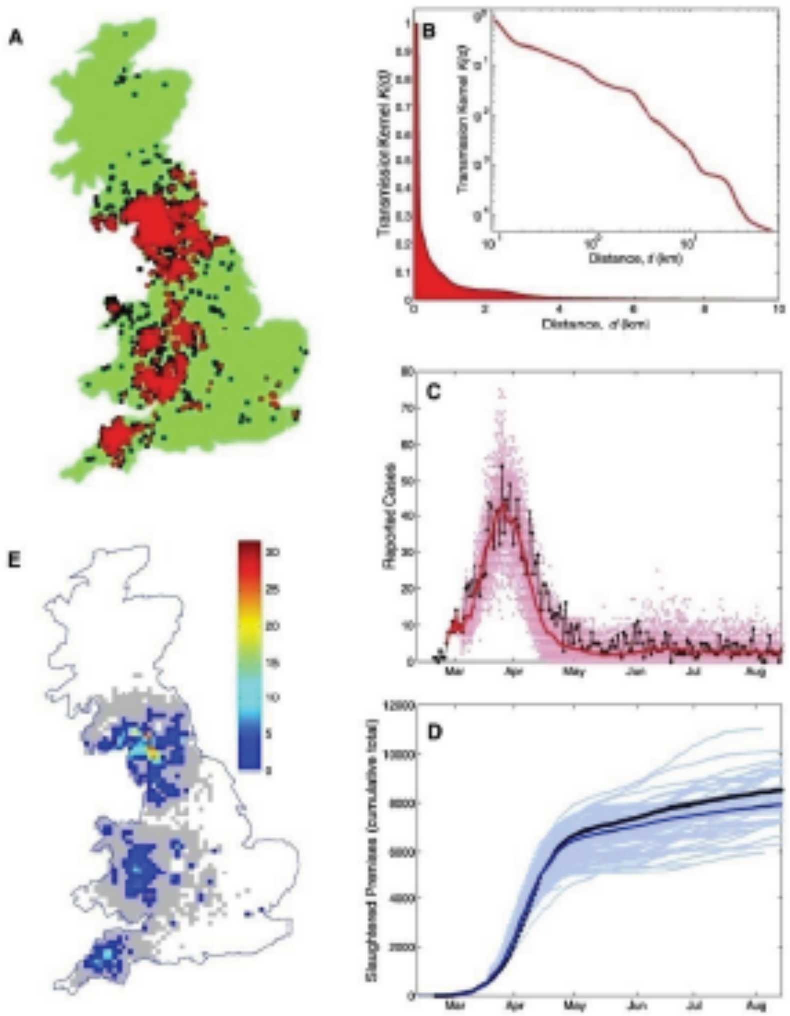
STARFLAG project: http://angel.elte.hu/starling/index.html

What is a complex system?



Tschinkel, *Journal of Insect Science,* 2004

What is a complex system?



Keeling et al., *Science 2001*

What is mathematical modelling?

A way of travelling securely from **A** to **B**.

A: Assumptions about the world.

B: Consequences of those assumptions

Mathematics **is** rigorous thinking.

Richard Feynman

Why do we do mathematical modelling?

1, Explain data as simply as possible.

2, Link together levels of explanation.

3, To provide detailed descriptions.

4, To predict future outcomes.

1, Explaining data simply

Provide one or two simple rules from which everything else is derived.

This is qualitative modelling, but necessarily some comparison to data.

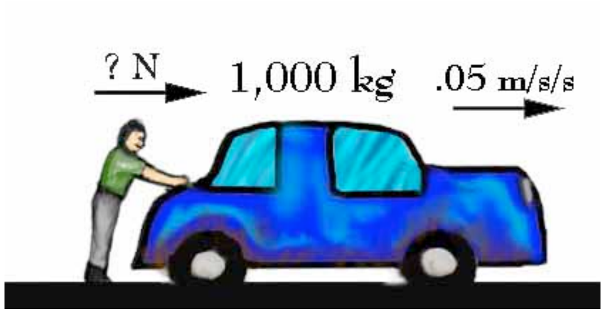
Explanation ratio: Explained/Assumptions

Dawkins: http://richarddawkins.net/articles/2236

1, Explaining data simply

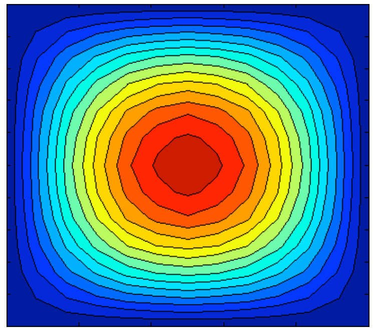
Newton's second law

$$\mathbf{F} = \frac{\mathrm{d}}{\mathrm{d}t}(m\mathbf{v})$$



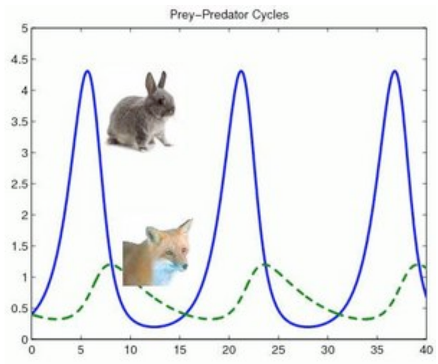
Heat equation

$$\frac{\partial u}{\partial t} - \alpha\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) = 0$$



Predator-prey models.

$$\dot{x} = xf(x,y)$$
$$\dot{y} = yg(x,y)$$



2, Linking levels of explanation

Large aggregates cannot be understood by simple extrapolation from the behaviour of a few particles.

Need mathematical models to integrate our understanding from one level to the next.

Explanation ratio may be lower, but more accurate.

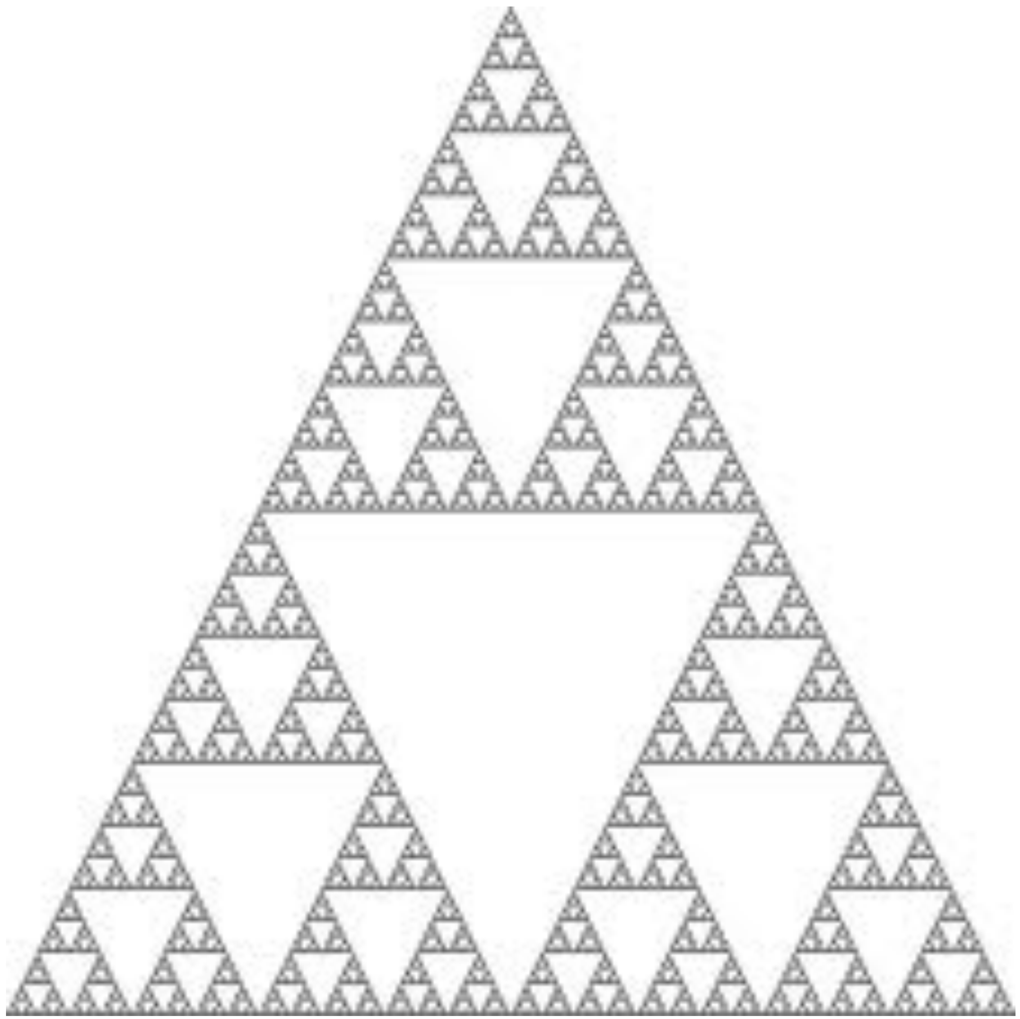
according to the idea: The elementary entities of science X obey the laws of science Y.

X	Y
solid state or many-body physics	elementary particle physics
chemistry	many-body physics
molecular biology	chemistry
cell biology	molecular biology
•	•
•	•
•	•
psychology	physiology
social sciences	psychology

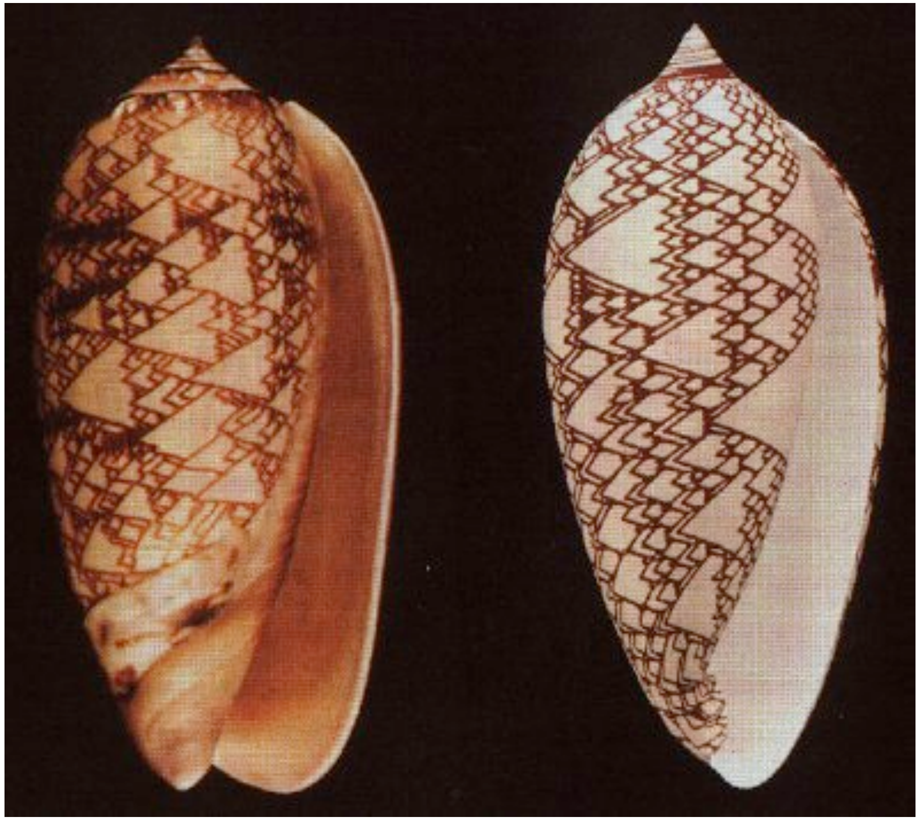
But this hierarchy does not imply that science X is "just applied Y." At each stage entirely new laws, concepts, and generalizations are necessary, requiring inspiration and creativity to just as great a degree as in the previous one. Psychology is not applied biology, nor is biology applied chemistry.

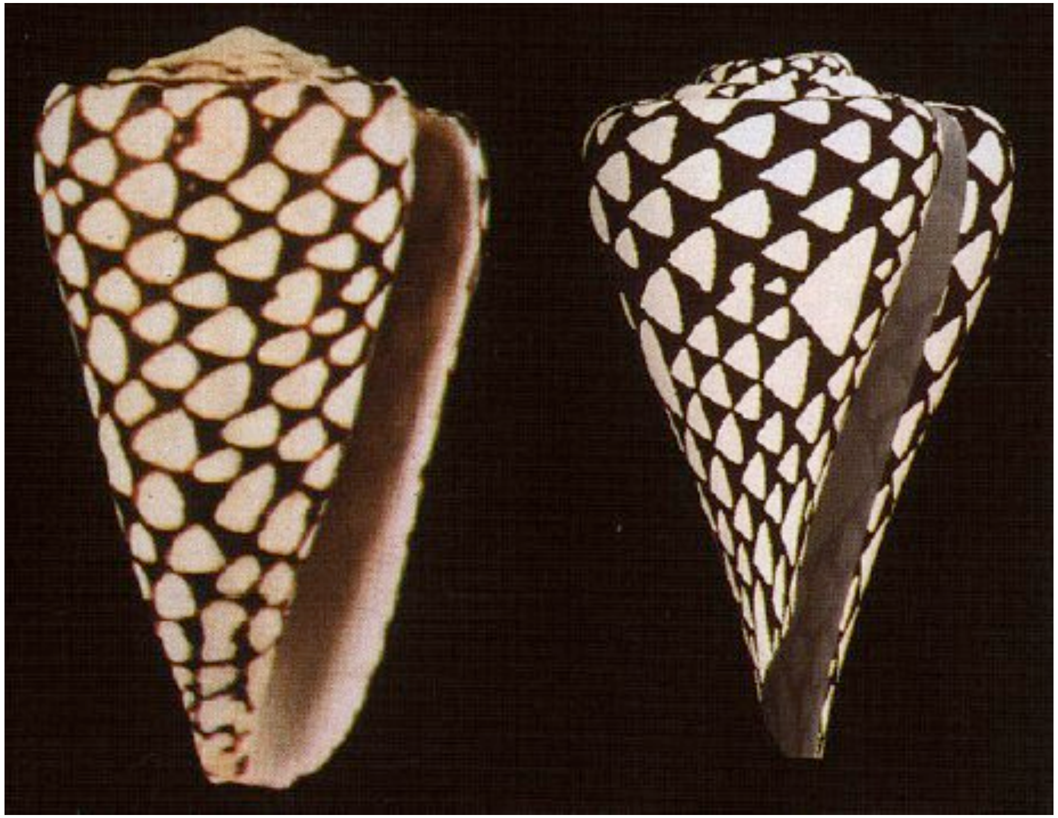
P. W. Anderson (1972) Science

2, Linking levels of explanation
Cellular Automata



2, Linking levels of explanation
Cellular Automata





Meinhardt, H. 1995 *Algorithmic Beauty of Sea Shells*
Cellmorphs: http://aimfeld.ch/cellmorphs/cellmorphs.html

3, Detailed descriptions

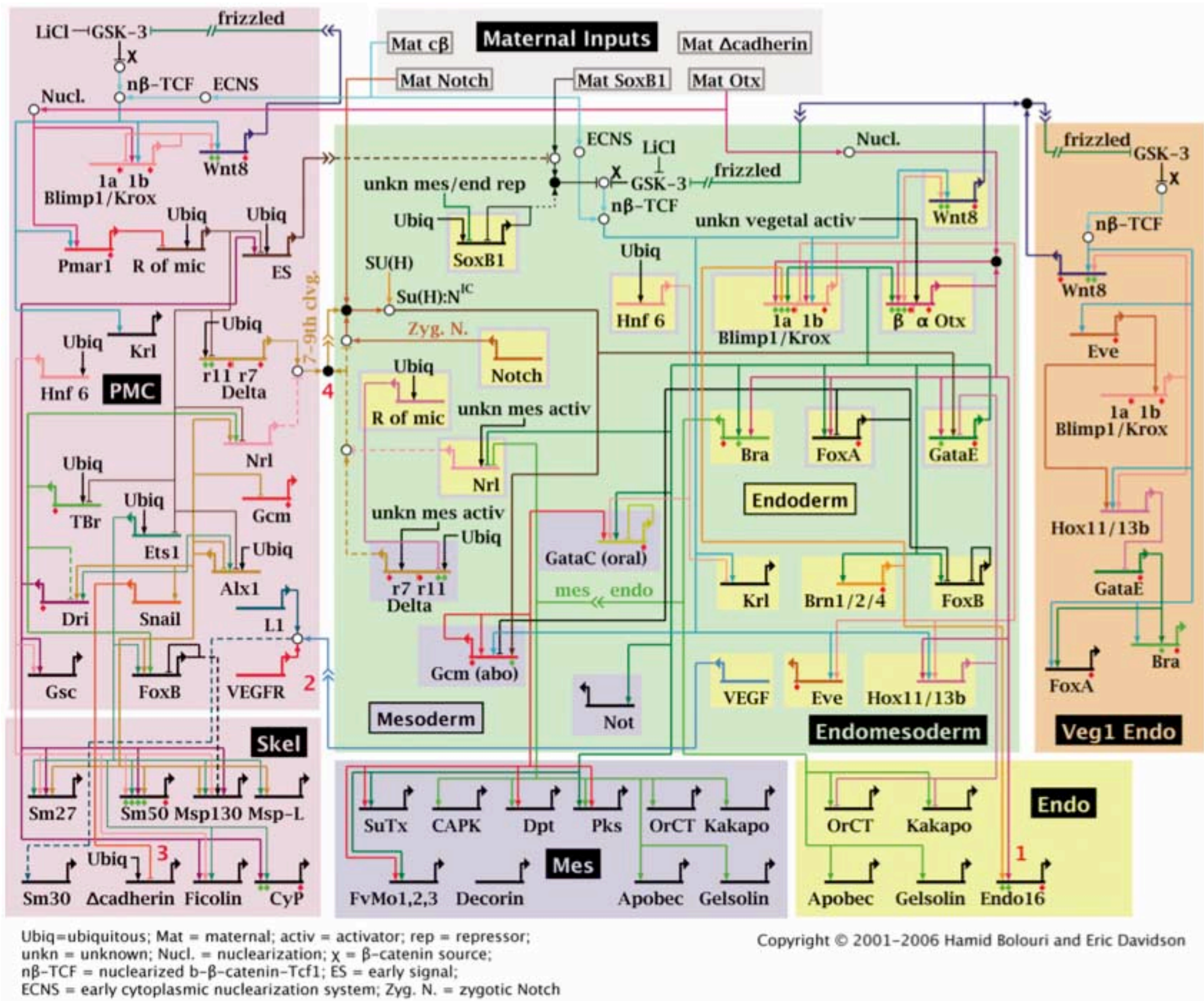
Put everything we know down in one place.

Quantitative modelling.

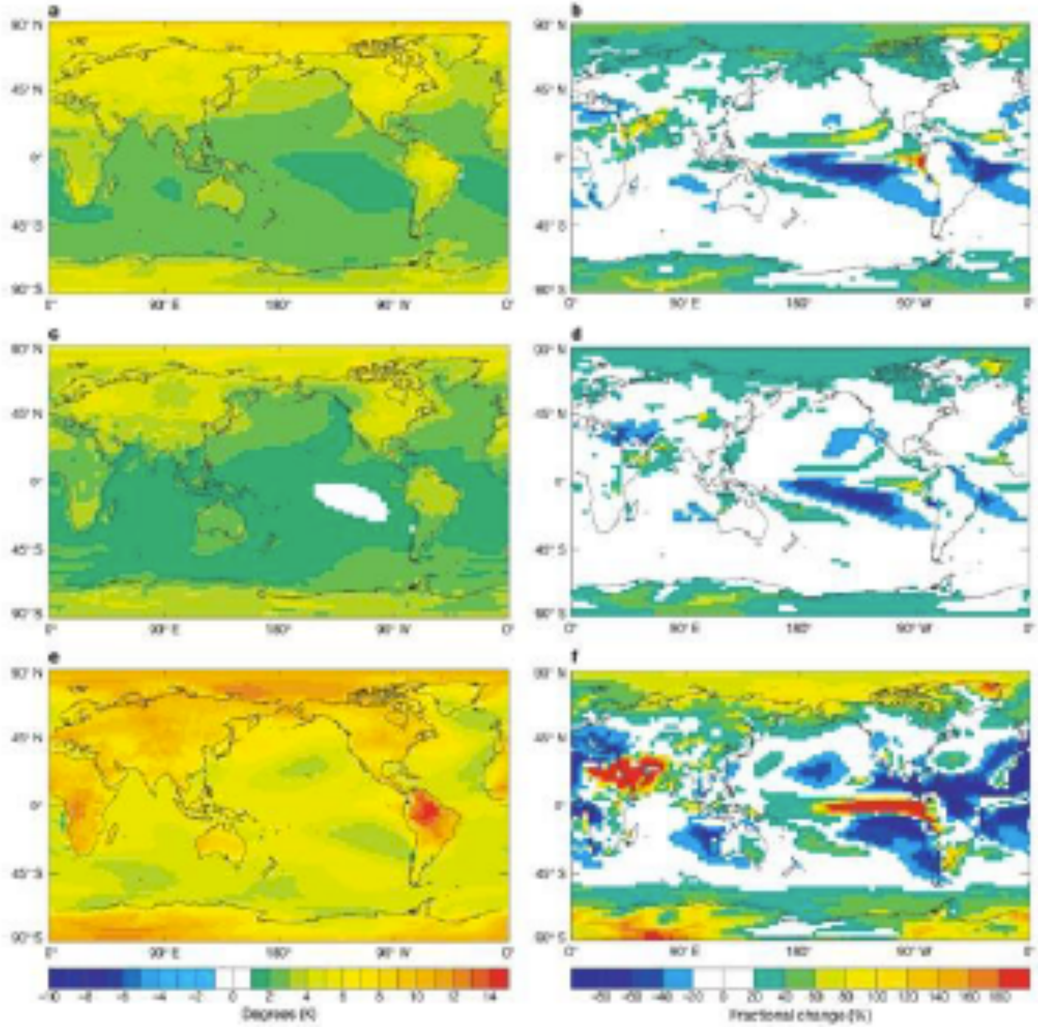
Test that this knowledge is self-consistent.

Find out if we really do understand how the system works.

3, Detailed descriptions



4, Predicting the future



Stainforth et al., *Nature 2005*

Why do we do mathematical modelling?

Decreasing level of abstraction

Increasing level of description

1, Explain data as simply as possible.

2, Link together levels of explanation.

3, To provide detailed descriptions.

4, To predict future outcomes.

Why do we do mathematical modelling?

Qualitative comparison between systems

1, Explain data as simply as possible.

2, Link together levels of explanation.

Quantitative description of particular system

3, To provide detailed descriptions.

4, To predict future outcomes.

Why do we do mathematical modelling?

Fun!

1, Explain data as simply as possible.

2, Link together levels of explanation.

Hard work

3, To provide detailed descriptions.

4, To predict future outcomes.

Why do we do mathematical modelling?

Fun!

1, Explain data as simply as possible.

2, Link together levels of explanation.

Hard work

3, To provide detailed descriptions.

4, To predict future outcomes.

Course structure

- Four two hour lectures.
- Eight/nine two hour labs. These are a central part of the learning experience.
- Two exercise sheets due May 9th and June 9th.
- Programming in Matlab preferred.
- This is not a programming course but we will help with programming.

Two exercise sheets

- Each exercise sheet has three 'projects', each consisting of a series of questions.
- After each question is a number of points associated with the question.
- The total points over all the questions is 100.
- To pass the course (grade 3) you must correctly answer questions amounting to at least 50 points. In order to get grade 4 you must get 75 points. In order to get a grade 5 you must correctly answer some of the questions labelled *grade 5 work* and have answered at least 75 points.

Projects

1. **Particles in boxes**. State-based simulations; stochastic simulations; mean-field approximations.
2. **Fads and fashions.** Co-operative phenomena; tipping points; bifurcation diagrams.
3. **Population dynamics**. Randomness and chaos; Lyapunov exponent and Entropy.
4. **Cellular automata.** Complex patterns from simple rules; box counting dimension
5. **Forest fires**. power laws; self-organised criticality.
6. **Self-propelled particle models.** Aggregation in space; phase transitions; flocking birds.

’Classical’ models

- Ordinary differential equation models.
- Stochastic differential equations.
- Partial differential equations.
- Markov chain models

Usually a whole course or number of courses will be dedicated to looking at these models.

These types of models are also essential in modelling complex systems.

’Complex systems’ models

To model complex systems we do not take any particular ’classical’ model as our starting point.

We are going to use techniques from the above approaches combined with computer simulations in order to better understand complex systems.

The basic approach we are going to take is that of writing down an algorithm which describes how our complex system works and use this algorithm to better understand the system.

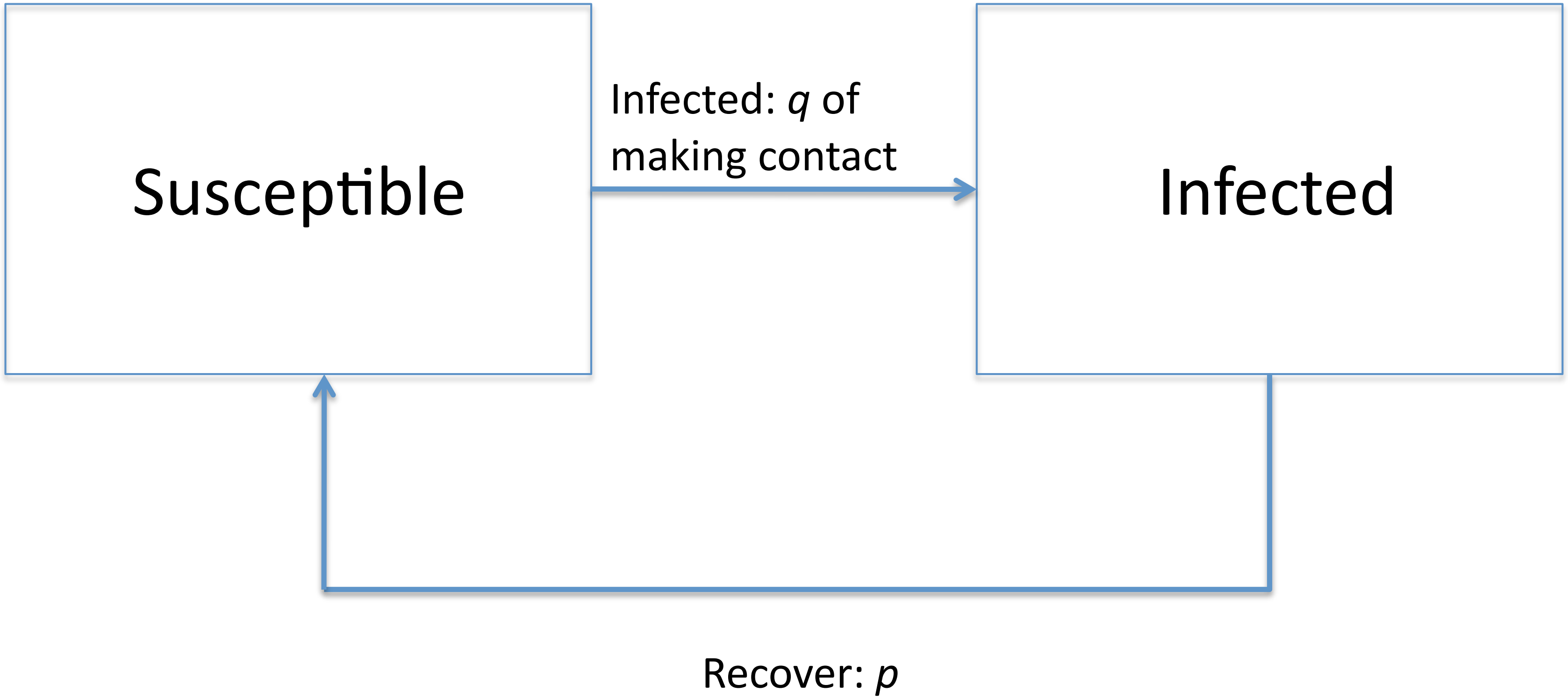
This algorithm becomes a computer simulation which we run to see how the system behaves. But the algorithm can also be related to more 'traditional' mathematical models in order that we can better analyse the system.

A simple example

Lets assume we want to model the spread of an epidemic disease.

- Each individual can have one of two states 'susceptible' or 'infected'.
- When infected we say that each individual has a probability *p* per day of recovering.
- They also have a probability *q* of making contact with another randomly chosen individual.
- When contacted the other individual will catch the disease, if they do not already have it.

States of the model



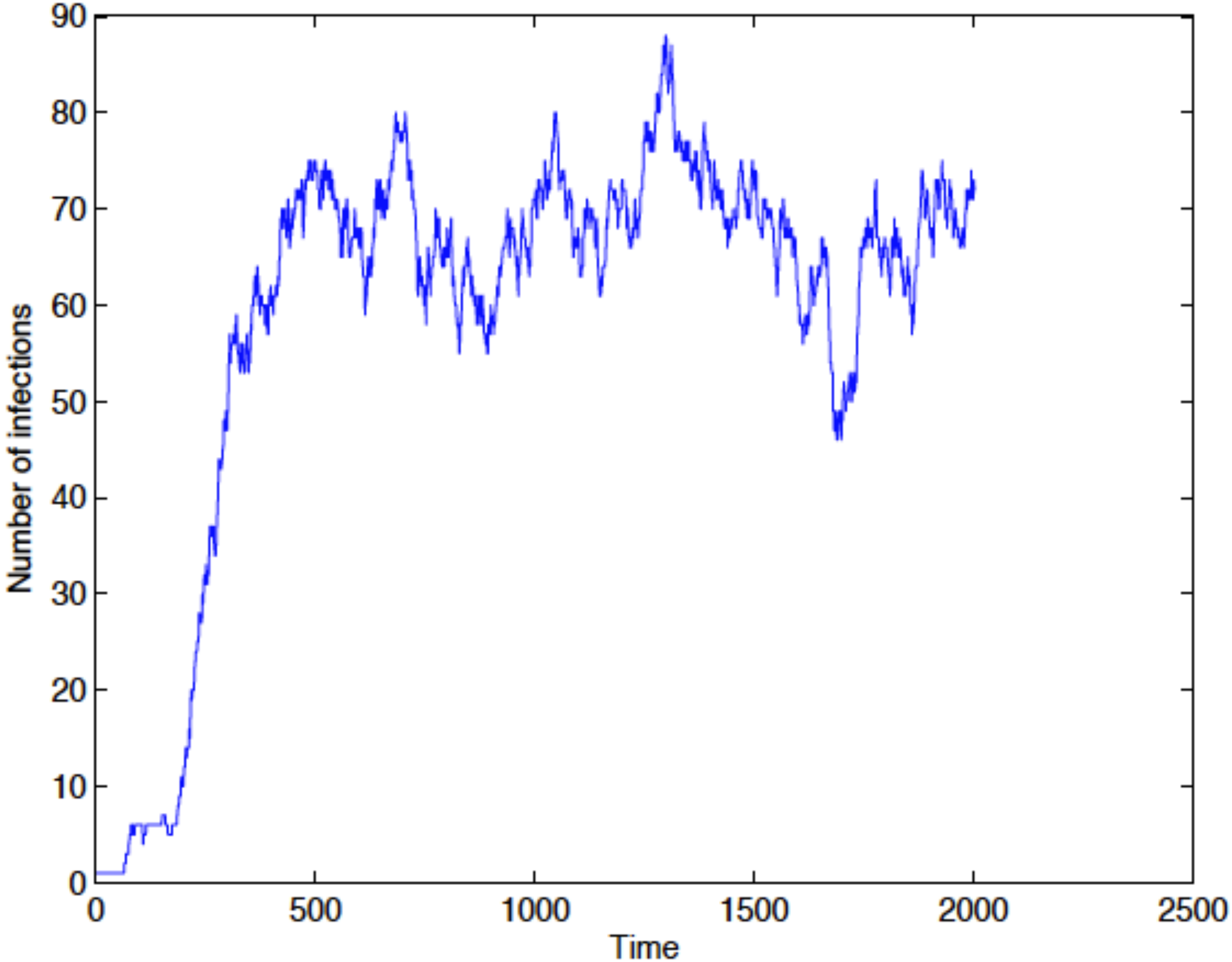
Assumptions

- Discrete time steps (e.g. Days).
- Can contact and recover within the same time step.
- All people equally likely to contact each other.
- No seasonality, environmental effects, differences between individuals or anything else!

Matlab code

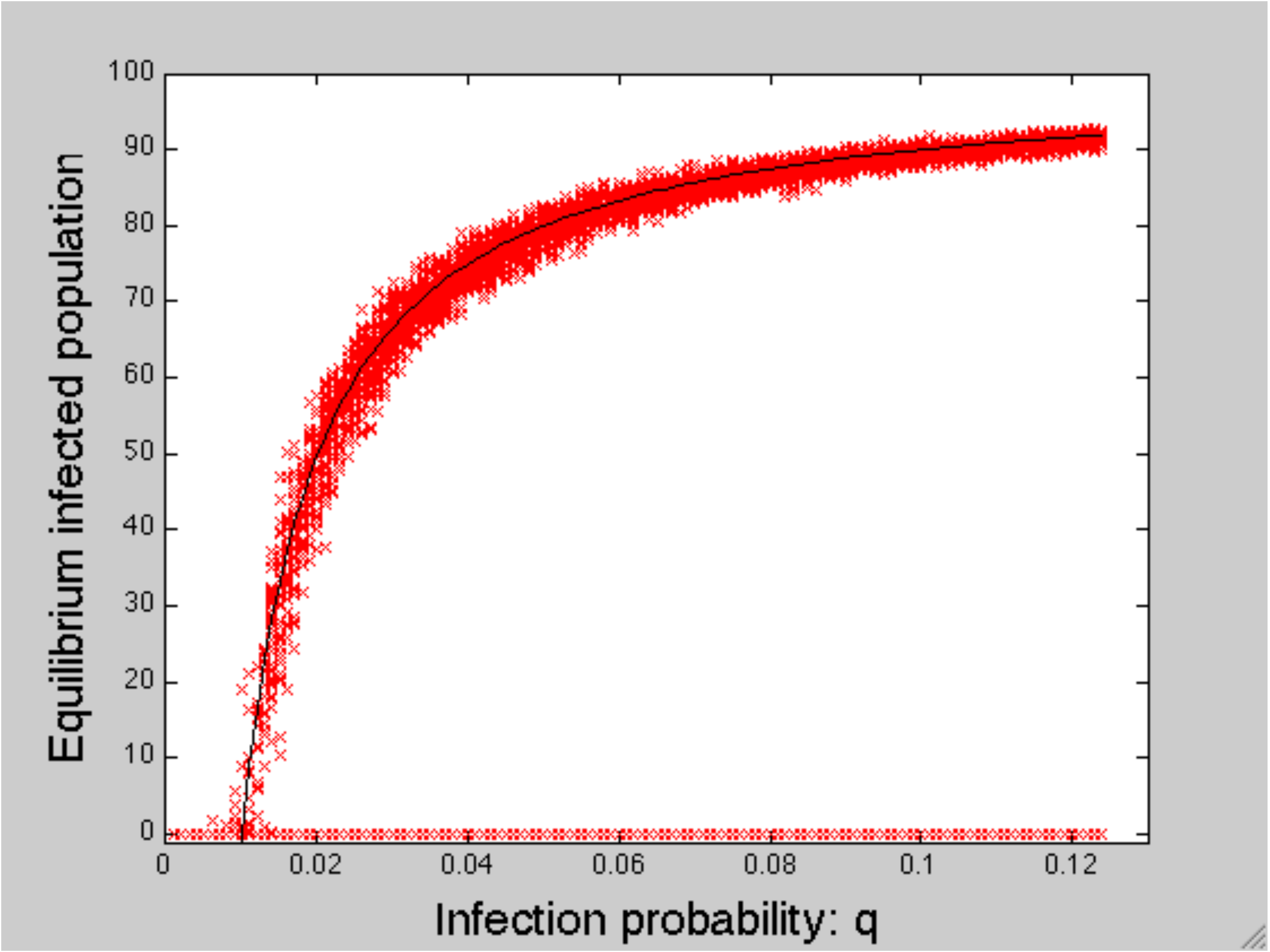
```
function x = runepidemic(p,q,N,T)
    x=zeros(T,1); %Number of individuals infected initially.
    x(1)=1;

    %For all time steps
    for t=1:T
        %Find the number recovering per time step using binomial distribution
        recover=binornd(x(t),p);
        %Find the number infecting others using binomial distribution
        contacted=binornd(x(t),q);

        %Now find number of contacted indivduals who are not infected already.
        %Here we have to be careful to take in to account of the fact that
        %if the same individual is contacted twice, they are only infected
        %once. We first make a random list of all the individuals contacted
        contacts=ceil(rand(contacted,1)*N);
        %Now find the unique set of contacted individuals
        contacts_unique=unique(contacts);
        %Then finally consider only those contacts which are not infected
        %already
        contacted_s=sum(contacts_unique>x(t));
        %Update infected population
        x(t+1)=x(t)-recover+contacted_s;
    end
```

Simulation outcome



Mean-field model

Bifurcation diagram



Bifurcation diagram

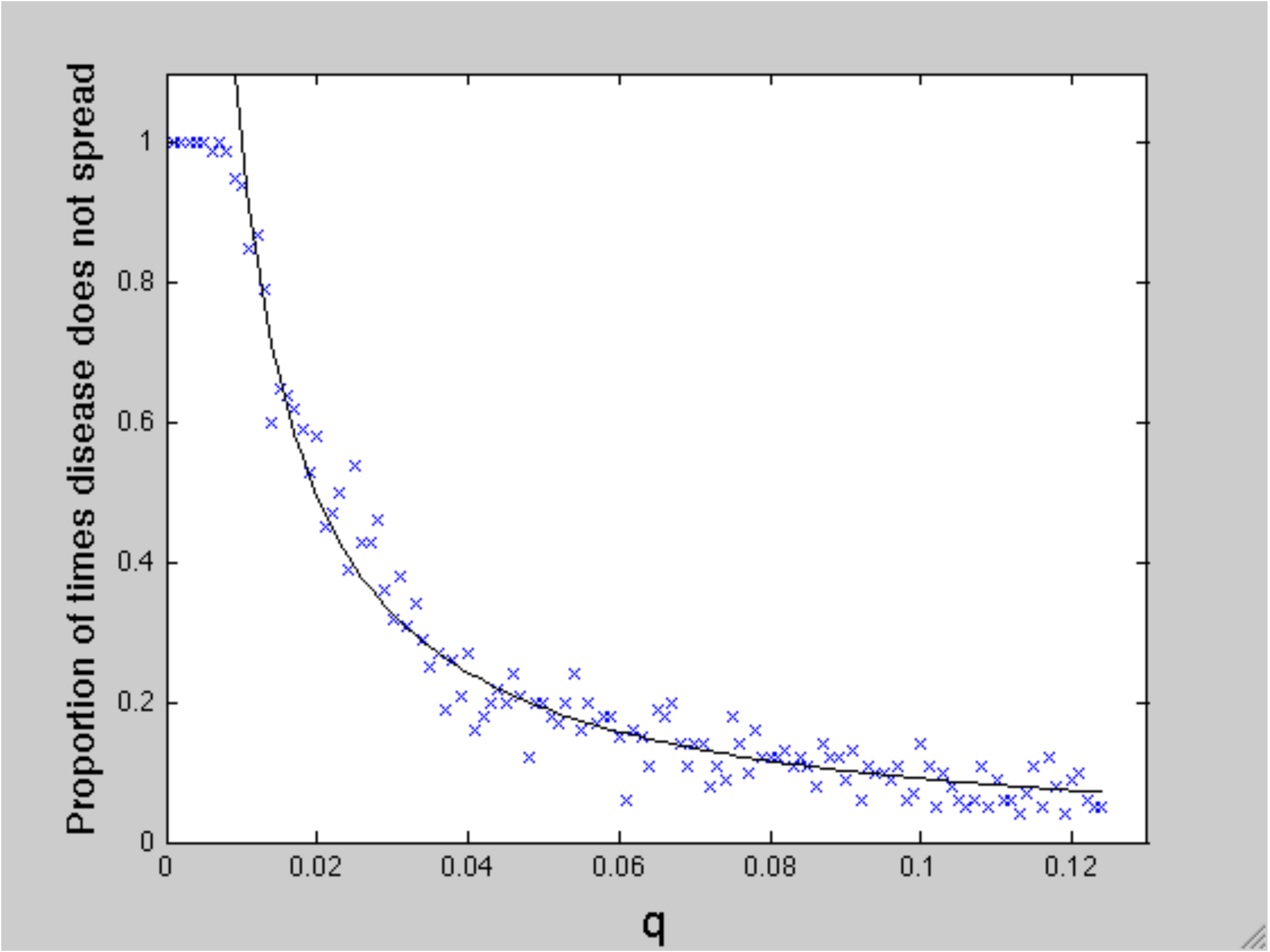
How the simulation behaves at equilibrium as we change a parameter.

Often number of inviduals or a transition rate is changed.

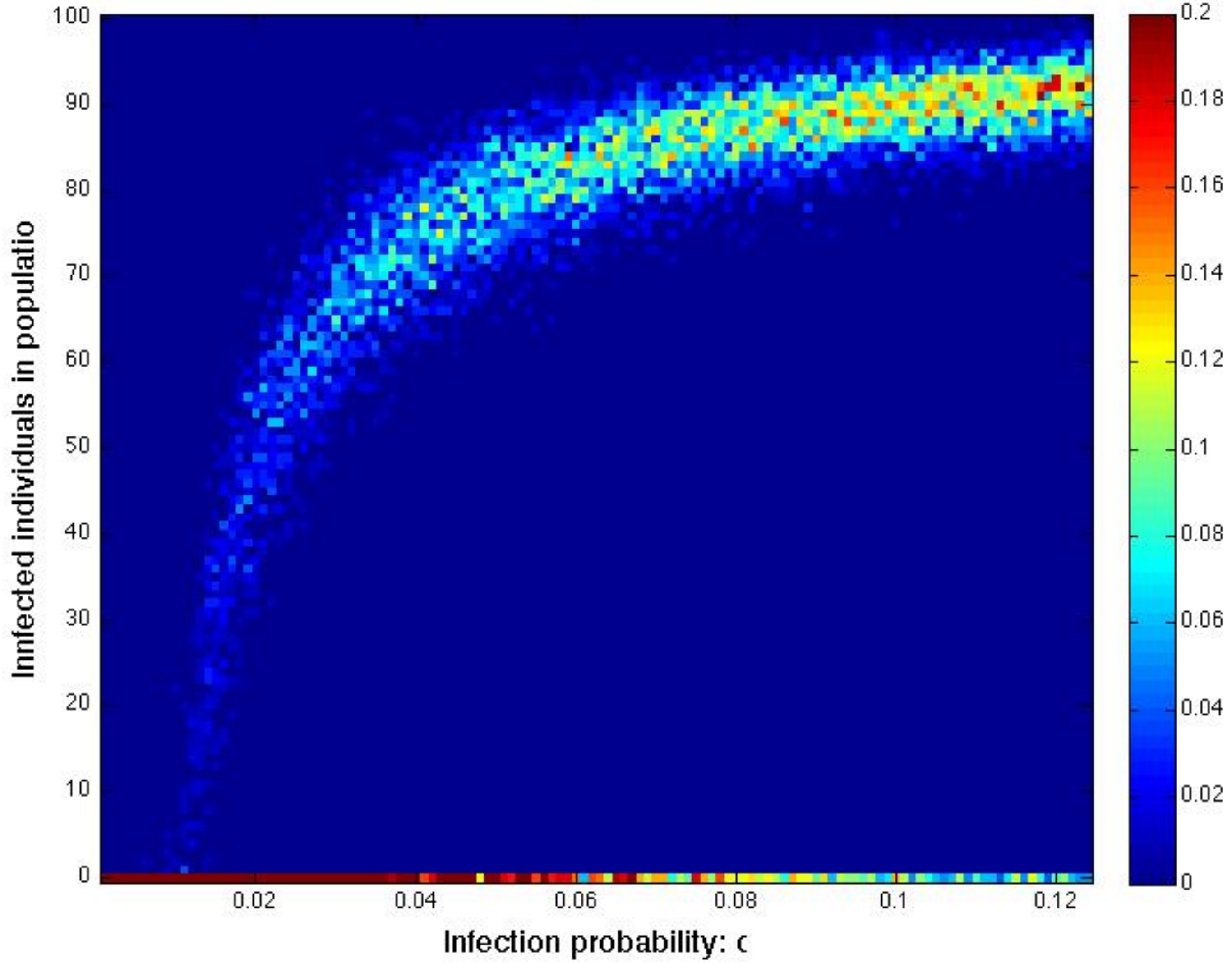
Can summarise total model behaviour in one figure.

Branching process model

Branching process model



'Heat map' of final distribution



Code

The code to generate the figures on the previous slides can be found at

www2.math.uu.se/~david/MCS/

runepidemic(*p*,*q*,*N*,*T*) takes paramters *p* and *q*, the population size *N* and the *T* time steps to run the simulation and returns number of infected.

simepidemic produces the above figures.

Why state-based model?

This is meant to be a course about complex models but the current example has been rather simple. Indeed, we have been able to turn it in to a differential equation model and a Markov chain model, both of which give a good insight in to the models behaviour. Why then would we ever use an individual-based or state-based model if we can use simple models like this?

The reason is that we may now wish to add more complex aspects to our model.

- We might want to make individuals have different characteristics which make them more or less likely to go to catch the disease.
- We might construct a social network or spatial arrangement for their interactions, whereby certain individuals are more likely to contact particular other people.
- We might want the individuals' chance of contracting a disease depend on their past record of infection. Individuals won't contact others if they have been infected recently.

All of these aspects would lead to complications which would make differential equation models or Markov chain models difficult to analyse.

But don’t throw away what you have learnt.

So if all complexities make the simple models unwieldy, why not limit our work to state-based and individual-based models? Why not just build in all the details we want in to the simulation and run them?

There are two important answers to this question

Approximation By keeping a set of simplified models which give an approximate description of a simulation we can understand why certain outcomes occur. If we see for example that a more complex model reaches a certain equilibrium we can use a mean-field approximation to understand why this particular equilibrium is reached.

Measures Many of the analytic tools from standard mathematical models can be applied to understanding simulation models. For example, the Lyapunov exponent in dynamical systems, concept of entropy from stochastic processes, the fractal dimension from measure theory are all useful tools for characterising models of complex systems. Concepts such as bifurcation diagrams where we measure a simulations response to a consistent change of one variable are key to understanding the behaviour of simulations.

It is these two aspects of approximation and measure which I will frequently return to when we try to understand complex systems.

Without an ability to approximate and measure a simulation model it is impossible to quantify its behaviour and understand its predictions.

Mathematics and Simulation

Are not different things!