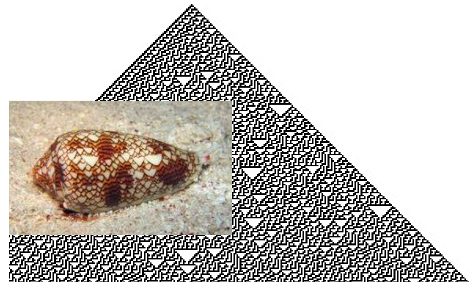
Modeling with Cellular Automata:

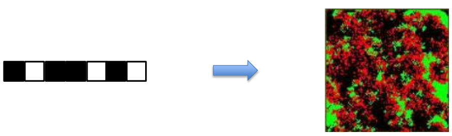
2D CA models and mean-field

Rule 30 - A Conus textile shell

current pattern	111	110	101	100	011	010	001	000
new state for center cell	0	0	0	1	1	1	1	0



From 1D to 2D models



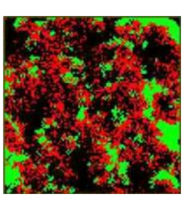
- We will also deviate a little from strictly CA
  - Conservation of 'particles'
  - Probabilistic rules
- (…Or are we?)

Occam's razor

- Newton: "We are to admit no more causes of natural things than such as are both true and sufficient to explain their appearances."
- If two *valid* models are equally predictive, prefer the simplest one

Today, a dichotomy

2D discrete simulation



Mean-field approximation

$$\frac{dx}{dt} = \alpha x - \beta xy$$
$$\frac{dy}{dt} = \delta xy - \gamma y$$

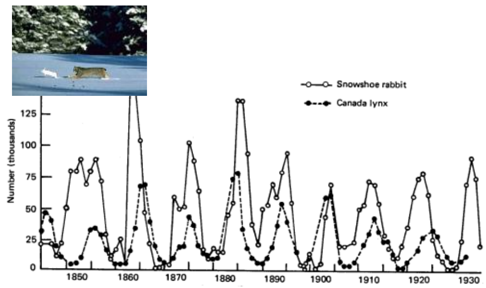
Today, a dichotomy

2D discrete simulation
- Simple micro rules lead to complex macro dynamics
- Spatially extended
- Parallel computation, or even dedicated hardware
- Can find micro-dynamics that explain observations

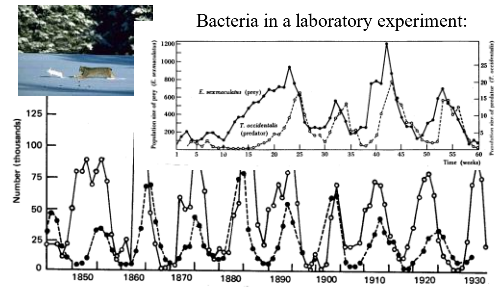
Mean-field approximation
- (If possible!)
- No simulation needed; solution calculated directly
- Good fit to 2D simulation → spatial effect is minimal*
- But says nothing about underlying micro-dynamics

* = so then why bother computing a spatial simulation…

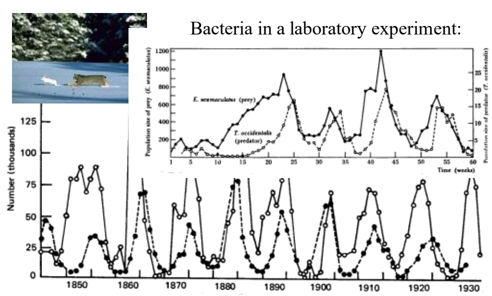
Predator-Prey model



Predator-Prey model



Predator-Prey model



Research question: what could cause (shifted) oscillation

Lattice model predator-prey model

- Rule 1: a prey has a probability *dh* of being captured and eaten by a predator in the neighbourhood of the prey
- Rule 2: no predator in the neighbourhood: prey probability *bh* of giving birth to prey in an empty site of this neighbourhood
- Rule 3: After having eating prey, predator has probability *bp* of giving birth to predator at the site site which was use to be prey
- Rule 4: Predator has a probability *dp* of dying
- Rule 5: Predator move to catch prey; prey move to evade predator

- Note: rules 1-4 simultaneous update; rule 5 sequential update

Predator-Prey model: 2D grid

- https://www.youtube.com/watch?v=FCTCRR5fNgU

Species abundance in simulation

- https://www.youtube.com/watch?v=sGKiTL_Es9w

Awesome! We found a model that explains the observed oscillation!

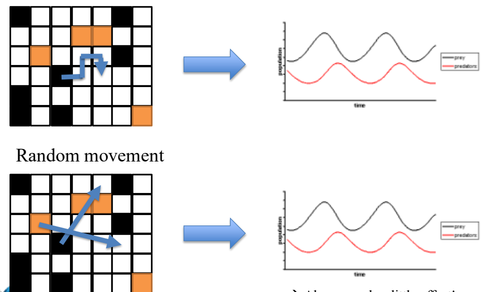
Mean-field approximation

- But does space have a fundamental effect?
  – (For explaining the oscillatory behavior)

Mean-field approximation

Local movement

Random movement



→ Aha, space has little effect!

Mean-field approximation

- Spatial positioning has little effect (in this model)
- **Why not get rid of it?**
- Then all predator/prey agents become *indistinguishable*
  – Their only difference was their x,y coordinates
- **So let's forget about individual agents too**
- Only model *number* of prey x(t) and *number* of predators y(t)

Coupled ODEs

Reproduction rate of sheep

Rate of sheep being eaten

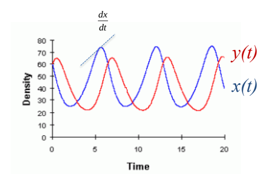
$$\frac{dx}{dt} = \alpha x - \beta xy$$

$$\frac{dy}{dt} = \delta xy - \gamma y$$

Reproduction rate of wolves

Death rate of wolves

Find $x(t)$ and $y(t)$



Solving a simple ODE

$$\frac{dx}{dt} = \alpha x$$

- Know it or look it up
- Guess and try
- **Integration**
- Numerical solution
- …

http://www.animations.physics.unsw.edu.au/jw/DifferentialEquations.htm

Solving a simple ODE: integration

$$\frac{dx}{dt} = \alpha x$$

$\int \frac{dx}{x} = \int \alpha\, dt$ and integration gives

$ln\, x = \alpha t + C$

$ln\, x_0 = \alpha * 0 + C$ so

$C = ln\, x_0$ so the solution is

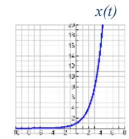
$ln\, x = \alpha t + ln\, x_0$ which gives

$ln\, x - ln\, x_0 = \alpha t$

$ln\left(\frac{x}{x_0}\right) = \alpha t$

$\frac{x}{x_0} = e^{\alpha t}$ or

$x = x_0\, e^{\alpha t}$



$x_0 = x(0) = 1$

http://www.animations.physics.unsw.edu.au/jw/DifferentialEquations.htm

Now let's solve the coupled ODE

$$\frac{dx}{dt} = \alpha x - \beta xy$$
$$\frac{dy}{dt} = \delta xy - \gamma y$$

http://math.stackexchange.com/questions/1367652/exact-solution-to-lotka-volterra-equations

Now let's solve the coupled ODE

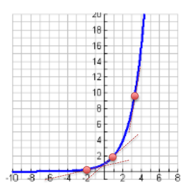
- We can't

$$\frac{dx}{dt} = \alpha x - \beta xy$$
$$\frac{dy}{dt} = \delta xy - \gamma y$$

http://math.stackexchange.com/questions/1367652/exact-solution-to-lotka-volterra-equations

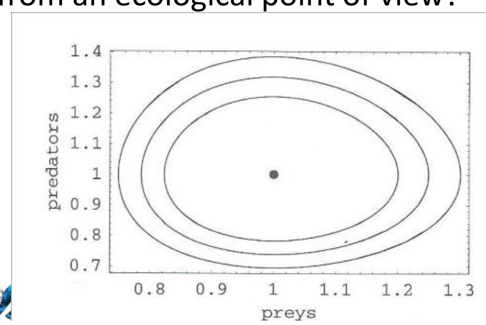
Now let's solve the coupled ODE

- We can't (analytically)
- So we let a computer do it numerically



http://math.stackexchange.com/questions/1367652/exact-solution-to-lotka-volterra-equations

How to build up a model: Lotka-Volterra model, is this model plausible from an ecological point of view?

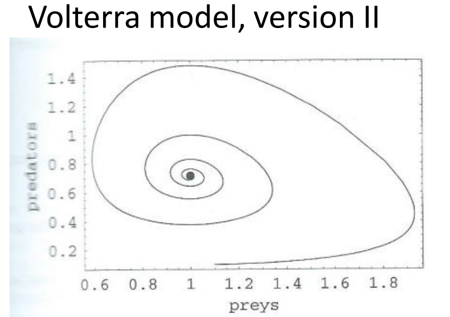


How to build up a model: Lotka-Volterra model, version II

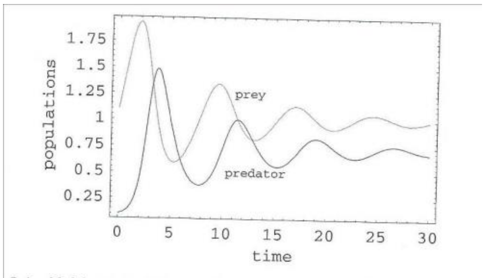
$$\frac{dx}{dt} = \alpha x(1 - \frac{x}{K}) - \beta xy$$
$$\frac{dy}{dt} = -\delta y + \gamma xy$$

- x herbivores, y predators, and five parameters: alpha birth rate prey, beta searching efficiency predator, efficiency food->predators, K carrying capacity

How to build up a model: Lotka-Volterra model, version II



How to build up a model: Lotka-Volterra model, version II



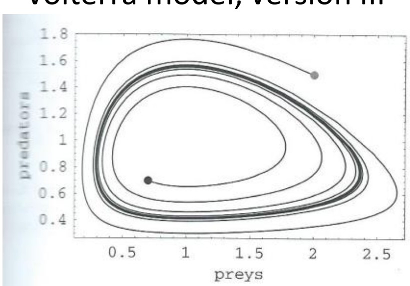
How to build up a model: Lotka-Volterra model, version III

$$\frac{dx}{dt} = \alpha x\left(1 - \frac{x}{K}\right) - \frac{a_H xy}{b + x}$$

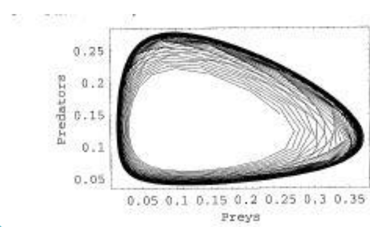
$$\frac{dy}{dt} = \frac{a_P xy}{b + x} - \gamma y$$

$\frac{a\ xy}{b + x}$ is a saturation term

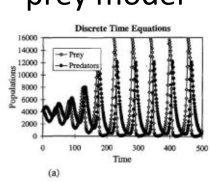
How to build up a model: Lotka-Volterra model, version III

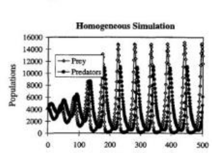


Lattice models, predator-prey model stable limit cycle

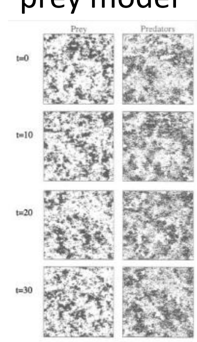


Lattice models, predator-prey model

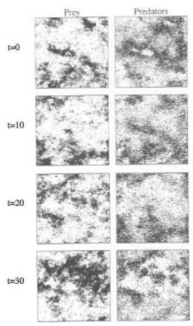




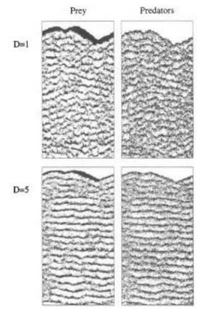
Lattice models, predator-prey model



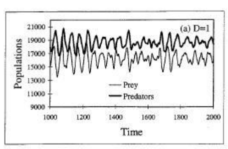
Lattice models, predator-prey model, high mobility

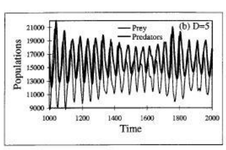


Lattice models, predator-prey model



Lattice models, predator-prey model, fluctuations low vs high mobility

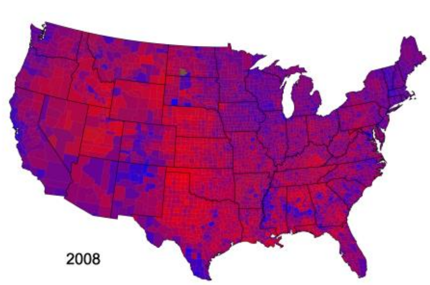




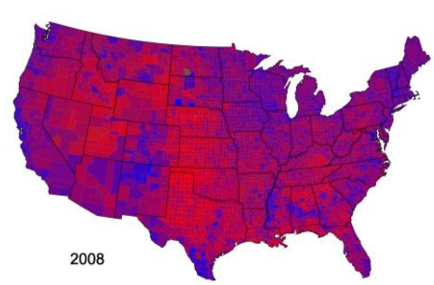
Models of Social Systems

VOTING MODEL

Vote dynamics



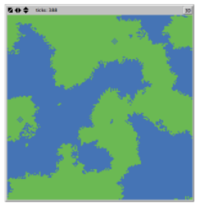
Vote dynamics



Research question: what could cause the spatial clustering?

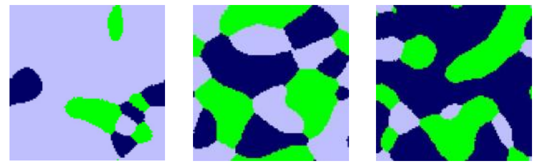
Voting Model

- State:
  - Opinion, political party, 0,1,…,k
  - Initialise as random
- Rule:
  - Majority wins -> your opinion is influenced only by your neighbours.
  - Some variants
    - if it's a tie change or not
    - close calls awarded to loser
- Try NetLogo – Voting
- Called 'majority rule'



Voting: stability

- 3 state, with next nearest neighbour – rapid transition in behaviour with small change in initial conditions



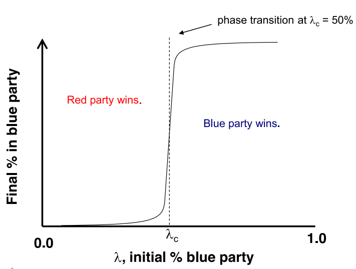
Initial state 36% light blue. (Light blue dominates.)

Initial state 34% light blue. (All colors balanced.)

Initial state 32% light blue. (Other colors dominate.)

Voting: stability

- Abrupt phase transition as we vary initial distribution



Social Temperature

- A measure of how volatile a crowd is.
  - Some people are angry and quick to act.
  - Some people are laid-back and slow to act.
- Assign everyone a number.
  - 0.0 → laid back, ∞ → easily angry
- The average of everyone's number is the social temperature, T.
  - A high social temperature means people are more likely to change their minds.
    - Fickle voters.
    - Reasoned choices go out the window – more random.
  - A low social temperature means people are less likely to change their minds.
    - Phlegmatic voters.
    - Reasoned choices go out the window – just plain stubborn.
  - An in-between social temperature means people may or may not change their minds.
    - Reasoned choices. All options are weighed and balanced, based on their neighbor's opinions.

Probabilistic Voter Model

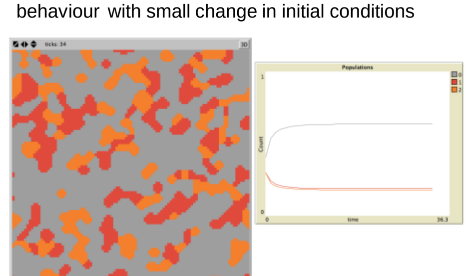
- Can do a better job with temperature if switch to a probabilistic model.
  - Probabilistic choice. Choose each opinion with some probability.
- Suppose have N neighbors.
  - n_1 have opinion "1" and n_2 have opinion "2". $N=n_1+n_2$
  - Probability of choosing opinion "1" is
  - $P_1 = n_1 / N$.
- Probability of choosing opinion "2" is
  - $P_2 = n_2 / N$.
- Note that the probability of a person (cell) changing their mind is

$$P_D(i) = \begin{cases} P_1, if\, s_i = 2 \\ P_2, if\, s_i = 1 \end{cases}$$

where s_i is the ith cell's value.

Voting: stability

- 3 state, with next nearest neighbour – rapid transition in behaviour with small change in initial conditions



Mean-field approximation?

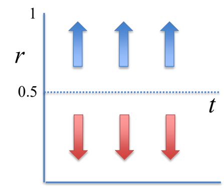
$$\Pr(1)=r$$
$$\Pr(0)=1-r$$
$$\Pr(\{0,1\}\to 1)=r^3(1-r)+r^4$$
$$\Pr(\{0,1\}\to 0)=r^1(1-r)^3+(1-r)^4$$

$$r\to r+\frac{dr_1}{dt},$$ ← r evolves over time

$$\frac{dr}{dt}=\Pr(0)\Pr(\{0,1\}\to 1)-\Pr(1)\Pr(\{0,1\}\to 0).$$

Mean-field approximation?

$$\Pr(1)=r$$
$$\Pr(0)=1-r$$
$$\Pr(\{0,1\}\to 1)=r^3(1-r)+r^4$$
$$\Pr(\{0,1\}\to 0)=r^1(1-r)^3+(1-r)^4$$



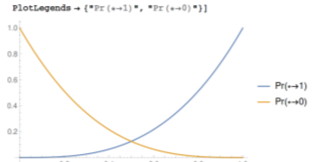
So r goes to 0 or 1 **only**

$$r\to r+\frac{dr_1}{dt},$$ ← r evolves over time

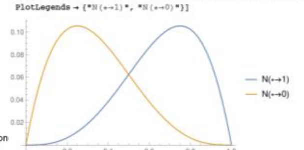
$$\frac{dr}{dt}=\underbrace{\Pr(0)\Pr(\{0,1\}\to 1)}_{\text{Fraction that changes } 0\to 1}-\underbrace{\Pr(1)\Pr(\{0,1\}\to 0)}_{\text{Fraction that changes } 1\to 0}.$$

Mean-field approximation

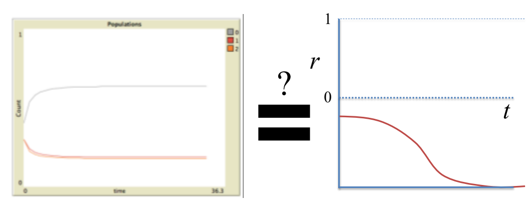
```
Plot[{r^3 (1-r)^1+r^4, r^1 (1-r)^3+(1-r)^4}, {r, 0, 1},
 PlotLegends → {"Pr(←→1)", "Pr(←→0)"}]
```




```
Plot[{(1-r) (r^3 (1-r)^1+r^4), r*(r^1 (1-r)^3+(1-r)^4)}, {r, 0, 1},
 PlotLegends → {"N(←→1)", "N(←→0)"}]
```



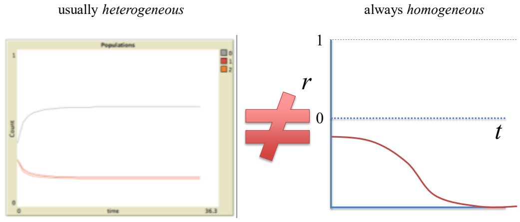
Mean-field approximation?



Mean-field approximation?

Steady state is usually *heterogeneous*

Steady state is always *homogeneous*



Mean-field does not work! (for **this** model)

Mean-field approximation

- (...But what if we consider a stochastic voter model?)

Other 2D models

Lattice Gas

- https://www.youtube.com/watch?v=Dq5Tvsu5-p0

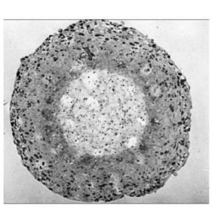
Lattice Gas: zoom-in

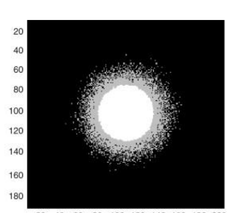
- https://www.youtube.com/watch?v=HluQpDFOceg

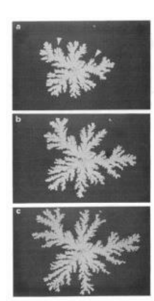
Lattice Gas: crowd evacuation

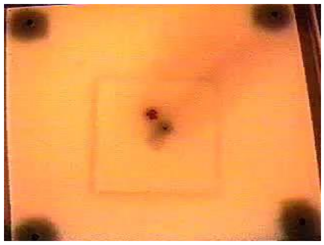
- https://www.youtube.com/watch?v=qeoJotgEUxk

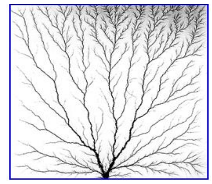
Lattice Gas: Cancer



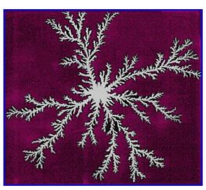








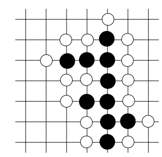
Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.



Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

Construction of the object

1. Determine growth candidates
   black circles are part of the object
   open circles are growth candidates
2. Determine growth probabilities



$$p_g((i,j)\in\circ \rightarrow (i,j)\in\bullet) = \frac{(c_{i,j})^{\eta}}{\sum_{(i,j)\in\circ}(c_{i,j})^{\eta}}$$

   η is a free parameter, usually $0 \le \eta \le 2$, for classical DLA, $\eta = 1$.

3. Grow the object
   for each grow candidate, draw a random number between 0 and 1, and if this number is smaller then p_g, add the candidate to the object.

Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

Mathematical model (including space), diffusion equation

$$\frac{\partial c}{\partial t} = D\nabla^2 c$$

Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

DLA

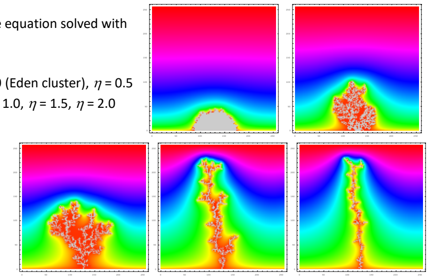
- https://www.youtube.com/watch?v=rkmw3C3JGvo

Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

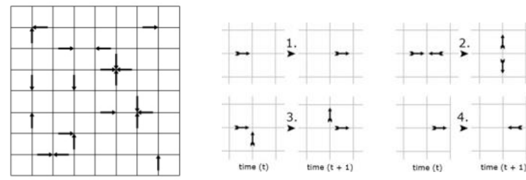
Some results

256^2 grid, Laplace equation solved with SOR.

Top lane : $\eta = 0.0$ (Eden cluster), $\eta = 0.5$
bottom lane : $\eta = 1.0$, $\eta = 1.5$, $\eta = 2.0$

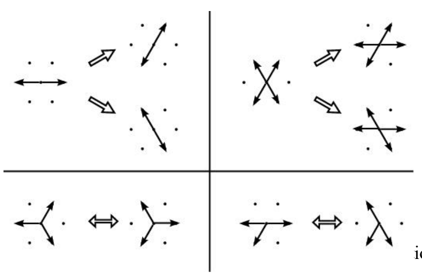


Lattice Gas: implementation



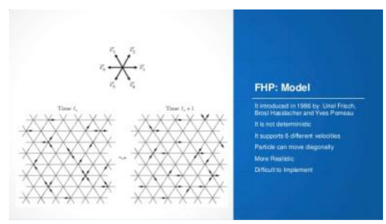
Each cell has four binary variables: one for each direction

Lattice Gas: implementation

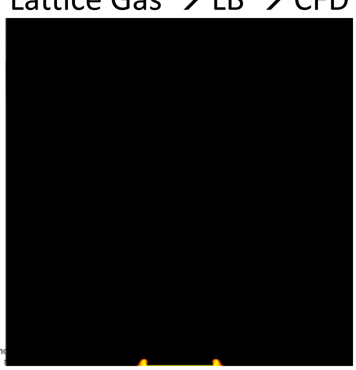


ion

Lattice Gas: implementation



Lattice Gas → LB → CFD

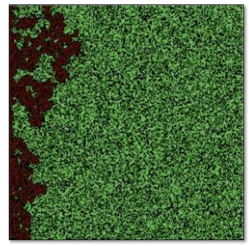


FOREST FIRE

Forest Fires

Important problem for simulation:
- How will a fire spread?
- How best to tackle fire – targeted felling, targeted fire fighting
- Early warning systems
- Is a forest at risk of complete destruction?

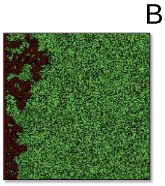
Forest Fires



- CA Model:
  - Cells have state:
    - Burnt,
    - Burning (intensity),
    - Vegetation (Burnable)
  - Fire spreads
    - A cell with fire will spread to neighboring vegetation (probability?)

Forest Fires

Basic Rules:



1. Ignite

burning cells: ignite all moore neighbourhood

2. Fade

burning cell: degrade intensity by α

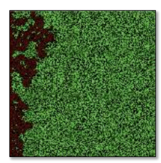
3. Die out

burning cell: if intensity = 0 set state to burnt

Stochastic models:
Rodolfo Maduro Almeida and Elbert E N Macau 2011 *J. Phys.: Conf. Ser.* **285** 012038 doi:10.1088/1742-6596/285/1/012038 (http://iopscience.iop.org/1742-6596/285/1/012038/pdf/1742-6596_285_1_012038.pdf)

Forest Fires

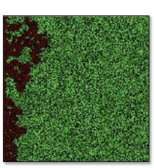
What can it tell us?



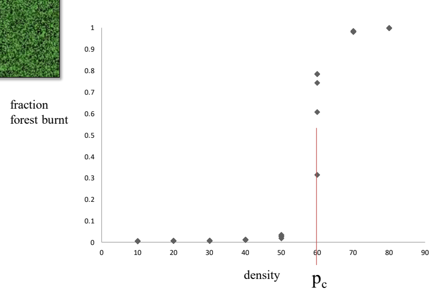
Given a forest density – what do we expect the final amount of burnt trees to be?

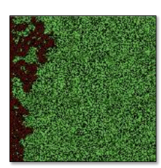
Given a forest density – How long will the fire take to burn out (or consume forest)?

Note that you may see density described as a probability p, which is the probability of seeing a tree or not in the initial configuration

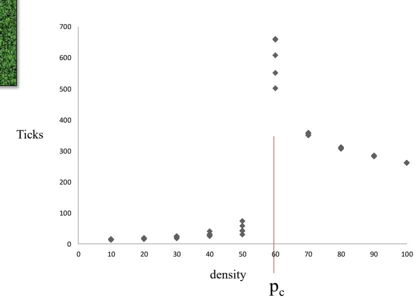


NetLogo - try





NetLogo - try



Forest Ecology

- Many ecosystems, particularly prairie, savanna, chaparral and conifer forests, have evolved with fire as a necessary contributor to habitat vitality and renewal.

- http://www.nps.gov/yell/parkmgmt/fireecology.htm

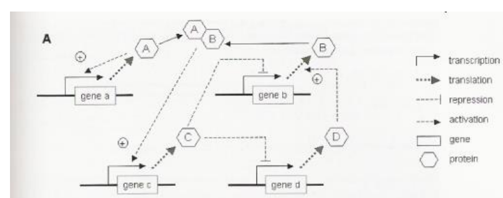
Forest Fires v2

- **http://ccl.northwestern.edu/netlogo/models/community/Fire%20Ecology**
- The model calculates the number of lightning strikes occurring on dry days, based on the length of the fire season, the number of days it rains, the time it takes for the landscape to dry after a rain, and the number of lightning strikes per season.
- Each lightning strike occurring on a dry day then has a certain probability of igniting a fire, which is based on the age of the patch of forest struck, and the flamability (Fm) of the forest type. Ignited fires spread from patch to patch based on the same parameters.
- As in a real forest, a patch can't burn for a certain amount of time after it's burned already (set by variable Lag). In addition, forests become more flammable as they age, modeled by the Mature_Age variable; the longer it takes to reach Mature_Age, the slower the flammability of the forest increases.
- Finally, the model includes an approximation of (human) fire suppression effort. The higher the value of the Suppression variable, the more burning pixels will be targeted for suppression (whether suppression is successful is random). As a result of this design, even a small amount of suppression will likely be effective as long as there are only a few burning pixels. If a fire chances to get rather large, however, the suppression effort quickly becomes ineffective.

Forest fire

- https://www.youtube.com/watch?v=bUd4d8BDIzI
- Three cell types
  - Empty
  - Vegetation
  - Burning
- If 1 or more neighbor cells is burning, then cell will burn.
- Burning cells turn empty.
- Empty cells (eventually) regrow vegetation
- Many model improvements possible to make it more realistic
  - Wind direction, types of vegetation, …
  - http://ccl.northwestern.edu/netlogo/models/community/Fire%20Ecology

Modelling genetic networks, example



Gene networks, Boolean networks

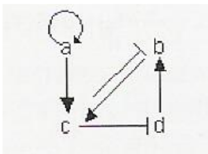
- In the Boolean network approach the expression level of each gene is assigned to a binary variable: a gene is considered to be either on (1) or off (0)
- The states of the genes are updated simultaneously in discrete time steps.
- New state can depend on previous state of the same gene or other genes

GRN, Boolean networks II

- N genes, N nodes of the network
- k interactions of a certain gene, k inputs node
- Every node 2 states, a network of N genes can be in 2^N different states
- State time t is N-dimensional vector
- State time t+1 depends on inputs, use Boolean rules, for k inputs, number of possible Boolean rules

$$2^{2^k}$$

GRN, Boolean networks III, example



GRN, Boolean rules

$$a(t+1) = a(t)$$
$$b(t+1) = (not(c(t))and(d(t))$$
$$c(t+1) = (a(t))and(b(t))$$
$$d(t+1) = (not(c(t))$$

GRN, Boolean networks V, example, successive states

0000 → 0001	1000 → 1001
0001 → 0101	1001 → 1101
0010 → 0000	1010 → 1000
0011 → 0000	1011 → 1000
0100 → 0001	1100 → 1011
0101 → 0101	1101 → 1111
0110 → 0000	1110 → 1010
0111 → 0000	1111 → 1010

Boolean networks VI, trajectories and attractors

- Number of states is finite, number of possible transitions is finite
- Each trajectory leads either to a steady state or a state cycle. These states are called attractors. Transient states do not belong to an attractor. All states that lead to the same attractor constitute the basin of attraction

engineering (inferring networks)

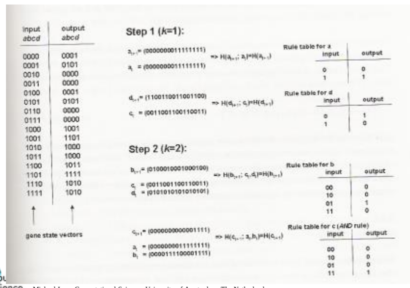
general not possible

- If we have time-dependent data and knockoout experiments, we might succeed. In time-dependent data: a strong expression of a transcription factor at time t will lead to activation or repression of gene expression of its targets at time t+1

engineering (inferring networks),

- Consider a pair of consecutive time-dependent conditions (time t and t+1) on n gene probes
- Binarize the expression values and define a set of rules that allows the computation of binarized expression levels at time t+1 from those of time t

inferring networks), REVEAL algorithm (Liang et al.,1999)



input abcd	output abcd
0000	0001
0001	0101
0010	0000
0011	0000
0100	0001
0101	0101
0110	0000
0111	0000
1000	1001
1001	1101
1010	1000
1011	1000
1100	1011
1101	1111
1110	1010
1111	1010

gene state vectors

Step 1 (k=1):

a_{t+1} = (0000000011111111)
a_t = (0000000011111111)
=> $H(a_{t+1}; a_t) = H(a_{t+1})$

Rule table for a

input	output
0	0
1	1

d_{t+1} = (1100110011001100)
c_t = (0011001100110011)
=> $H(d_{t+1}; c_t) = H(d_{t+1})$

Rule table for d

input	output
0	1
1	0

Step 2 (k=2):

b_{t+1} = (0100010001000100)
c_t = (0011001100110011)
d_t = (0101010101010101)
=> $H(b_{t+1}; c_t, d_t) = H(b_{t+1})$

Rule table for b

input	output
00	0
10	0
01	1
11	0

c_{t+1} = (0000000000001111)
a_t = (0000000011111111)
b_t = (0000111100001111)
=> $H(c_{t+1}; a_t, b_t) = H(c_{t+1})$

Rule table for c (*AND* rule)

input	output
00	0
10	0
01	0
11	1

inferring networks), REVEAL algorithm (Liang et al.,1999)

- 1. Identification of perfect input-output state pairs of connectivity k=1
- 2. Determination of the rules for the identified pairs at k=1
- 3. Identification of perfect input-output state pairs of connectivity k=2
- 4. Determination of the rules for the identified pairs at k=2
- 5. Identification of perfect input-output state pairs of connectivity k=p
- 6. Determination of the rules for the identified pairs at k=p
- 7. Stop if all genes have been assigned to rule, otherwise increment p and go to 5.

Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

inferring networks), REVEAL algorithm (Liang

- Binarization of the data yielded 54 different state transitions, 21 are correct transitions
- Pre-processing state transitions after frequency of occurrence yielded a confidence level for each transition
- If only transitions highest confidence selected 6 out of 7 are correct.

Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

inferring networks), REVEAL algorithm (Liang et

State transition	*t*	*t + 1*	*Confidence*	*Validity*
1	0000	0001	1	True
2	1101	1111	0.5	True
3	1111	1110	0.625	False
4	1110	1010	0.5	True
5	1010	1000	0.3	True
6	1000	1001	1	True
7	1001	1101	0.5	True

Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

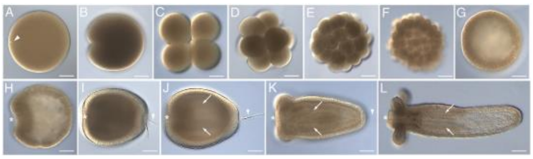
inferring networks), REVEAL algorithm + simulated results from ODE model , conclusions

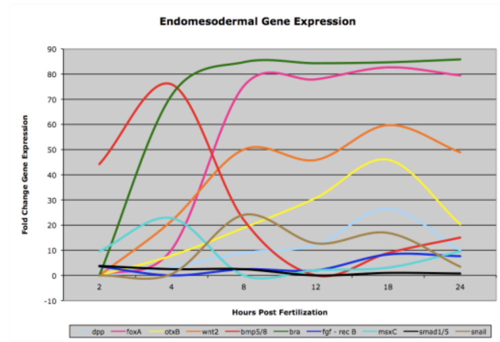
- States are incomplete. In practice most of the state transitions are missing after binarization. Example only 7 of 15
- Availability of many time points is crucial. You need to filter correct states from false states, many state transitions are required (in example 100 time points)
- Time points not too close to each other. The selection of time points determines the granularity of the set of state transitions. Tradeoff between the detection of as many state transistions as possible and avoiding false-positive

Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

Intermezzo: Example gene expression data in Nematostella vectensis obtained with qPCR

Nematostella vectensis
developmental stages

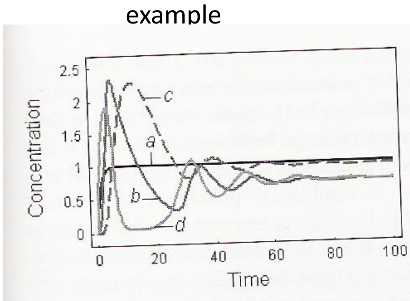




fold changes (normalized to actin and gadph and referenced to 12hpf)

Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.

Modelling genetic networks with ODEs, example



Computational Science Michael Lees: Computational Science, University of Amsterdam, The Netherlands.