Recap

Terminology recap

- Variable or state
- Differential equation
- Initial condition
- Trajectory
- Parameter
- Steady state/fixed point/equilibrium

Terminology recap

- Phase space/state space
- Vectorfield
- Fixed point (stable/unstable, focus/node)
- Nullcline
- Saddles, separatrix
- Bistability

Terminology recap

- Order of ODE solver
- Step size
- Variable/adaptable stepsize solvers
- Eulers method
- Heun’s method
- Runge Kutta method

Terminology recap

- Steady state
- Jacobian
- Eigenvalues of Jacobian
- Limit cycle
- Bifurcation

Comments on the coursework

- Please use the right terminology – if you don’t I will have a hard time interpreting what you mean
- Please don’t write a novel – Keep it short but to the point
- How: analytical derivation, what you simulated to test for the stability of fixed points, etc.
- Write down what tools you used to plot the phase space, nullclines, vectorfield etc.
- For the last question “Provide an explanation for the change in dynamics (i.e. changes in stability or location or number of fixed points) either analytically or numerically.” Either argue it with linear stability analysis (eigenvalues of Jacobian) or describe the nullclines, vectorfield and trajectories in phase space

Dynamical Systems

Yujiang.wang@ncl.ac.uk

Lecture 5 of 5

Overview

- Noise and stochasticity
- How to relate experimental data to models
- Exercise

- Dealing with higher dimensional systems
- Other ways of modelling: rule based models, discrete state models
- Best practice in modelling

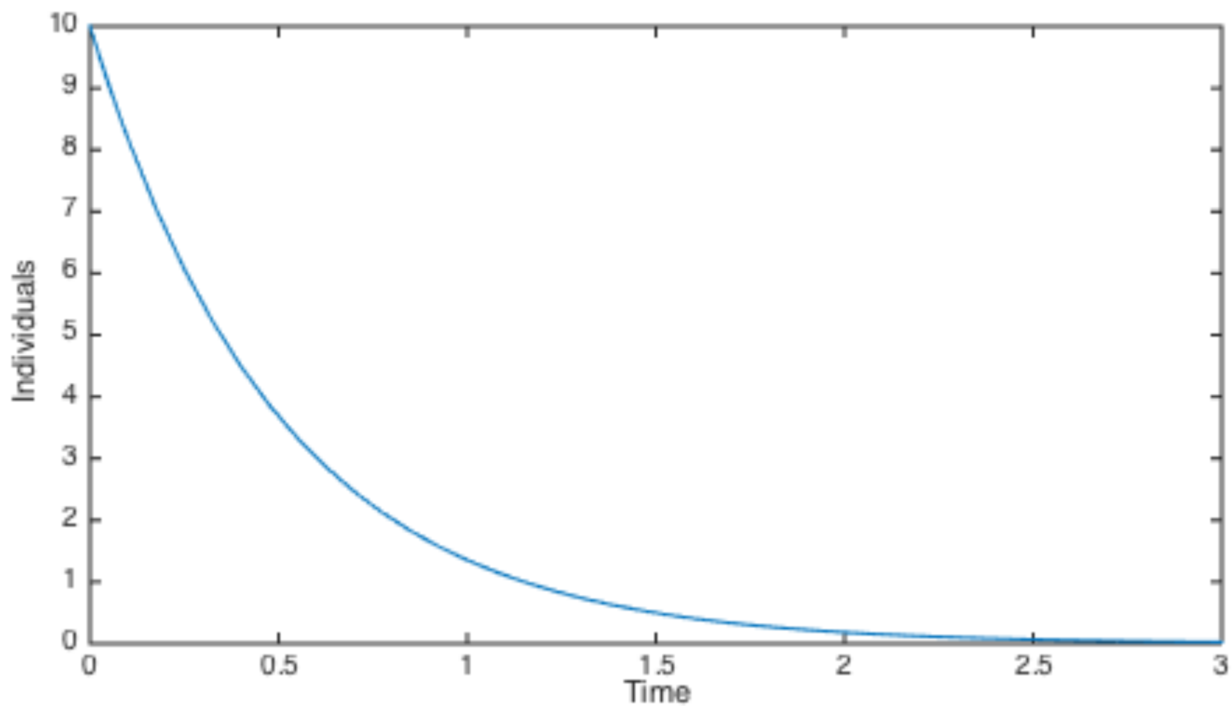
Overview

- Noise and stochasticity
- How to relate experimental data to models
- Exercise

- Dealing with higher dimensional systems
- Other ways of modelling: rule based models, discrete state models
- Best practice in modelling

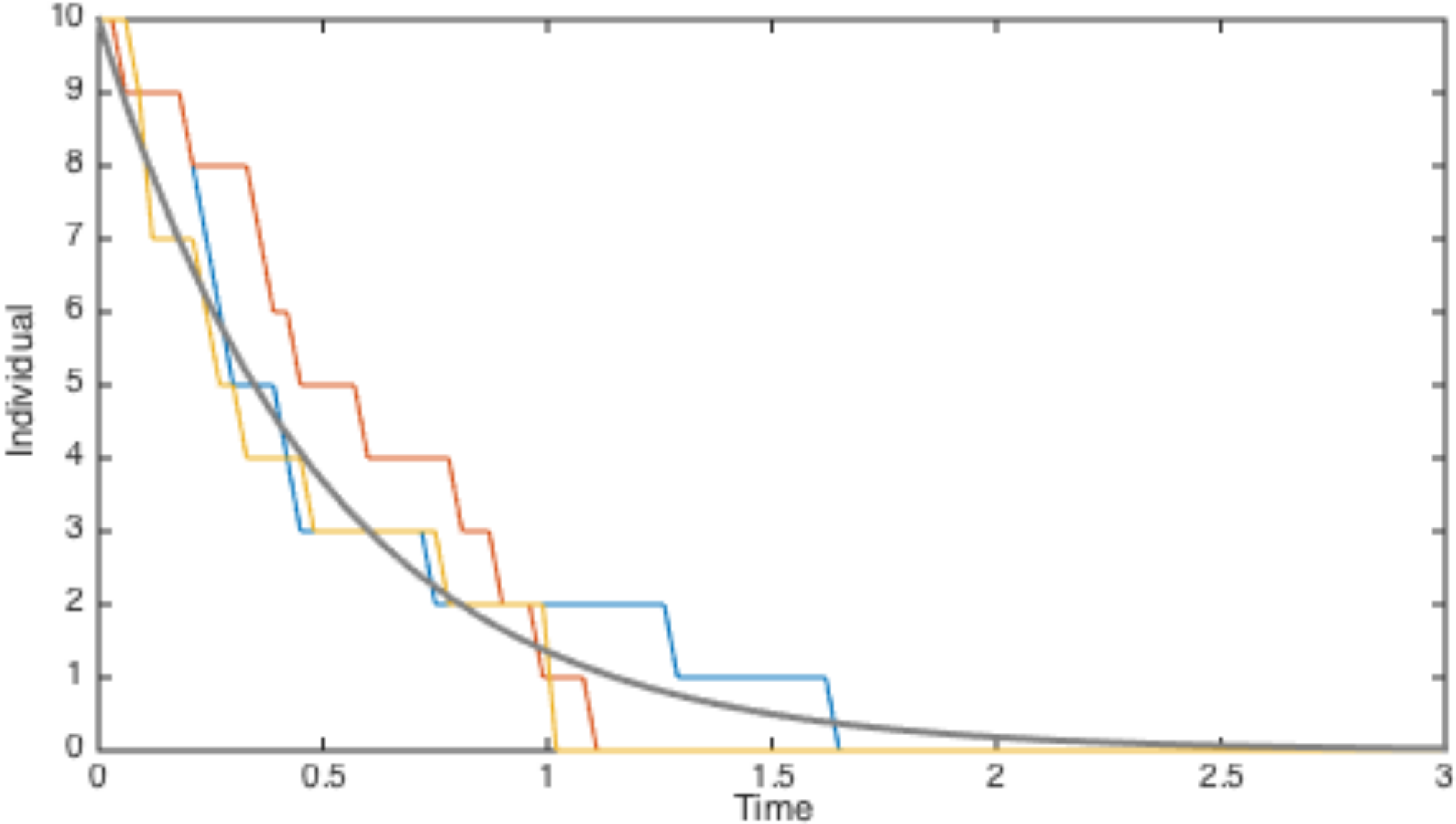
Low numbers

- Simulated population decay: dx/dt=-2*x
- What if we only have 10 individuals to start off with?



Dealing with low numbers

- Ode -> stochastic simulation
- Population x decays at a rate of -2
- Simulate: remove an individual from the population at t+h. h depends on the decay rate -2.
- Gillespie algorithm (use two random variables to determine next time step and next event)



10
9
8
7
6
5
4
3
2
1
0
Individual
0
0.5
1
1.5
2
2.5
3
Time

Stochasticity

- Gillespie algorithm crucial for simulating stochastic processes.
- The RHS of the ode essentially become the propensities of the Gillespie algorithm.
- Algorithm becomes very slow at higher number of individuals/particles/entities -> approaches the deterministic limit
- Warning: Stochastic solution not always converging towards the ode solution at high copy numbers!

Is it simply noise?

- Stochastic simulations are NOT just the ode-simulation with some noise.
- The noisy nature is generated intrinsically in the process, not simply added to the deterministic solution.

But sometimes…

we just want to add a bit of noise.

Stochastic differential equations (SDEs):

dX = (-2*X)*dt + dW

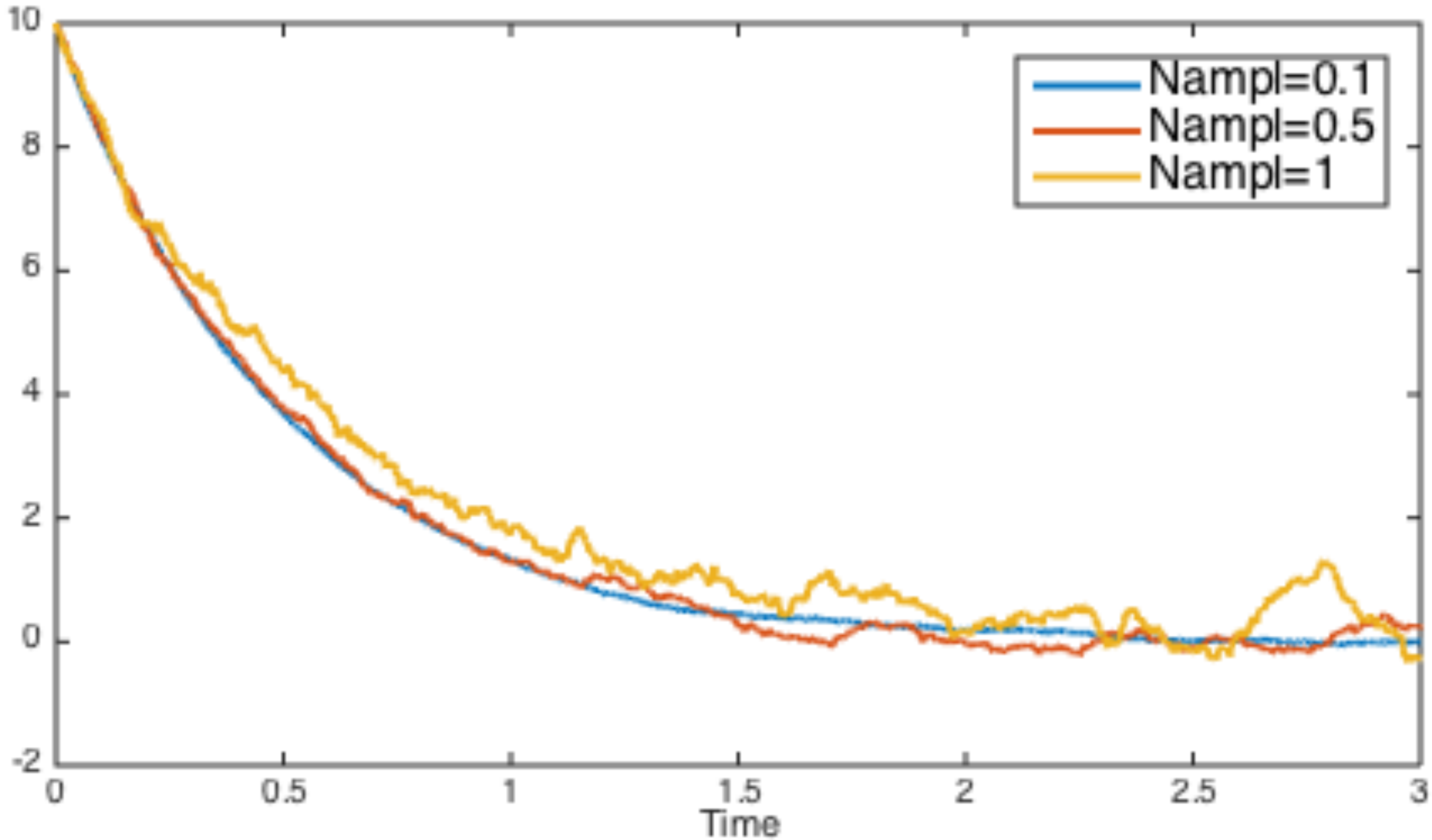
2. Brownian Motion. A scalar *standard Brownian motion*, or *standard Wiener process*, over $[0,T]$ is a random variable $W(t)$ that depends continuously on $t \in [0,T]$ and satisfies the following three conditions.

1. $W(0) = 0$ (with probability 1).
2. For $0 \le s < t \le T$ the random variable given by the increment $W(t) - W(s)$ is normally distributed with mean zero and variance $t - s$; equivalently, $W(t) - W(s) \sim \sqrt{t-s}\,N(0,1)$, where $N(0,1)$ denotes a normally distributed random variable with zero mean and unit variance.
3. For $0 \le s < t < u < v \le T$ the increments $W(t) - W(s)$ and $W(v) - W(u)$ are independent.

Solving equations with added noise

- Euler-Maruyama solver:
- X(t+dt)=X(t)+f(X(t))dt + sqrt(dt)*N(0,1)*Nampl
- dt is the step size
- N(0,1) is a random number drawn from a zero mean and 1 std normal distribution
- Nampl is the noise amplitude
- To check your simulation, run it with different stepsizes and make sure that you choose a stepsize that does not affect your results!

Added noise: dt=0.001

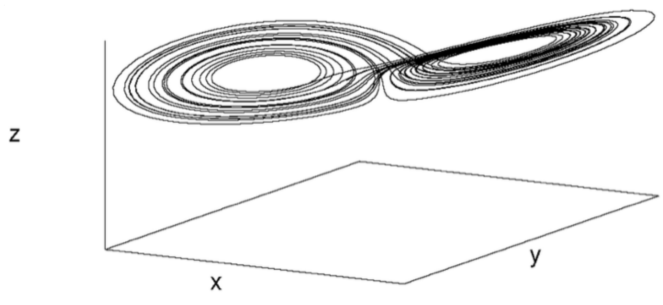


Overview

- Noise and stochasticity
- How to relate experimental data to models
- Exercise

- Dealing with higher dimensional systems
- Other ways of modelling: rule based models, discrete state models
- Best practice in modelling

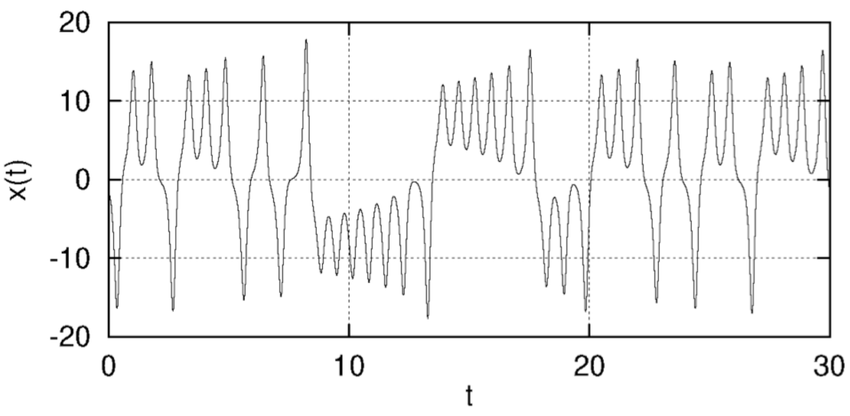
Other than data analysis methods: Takens Theorem

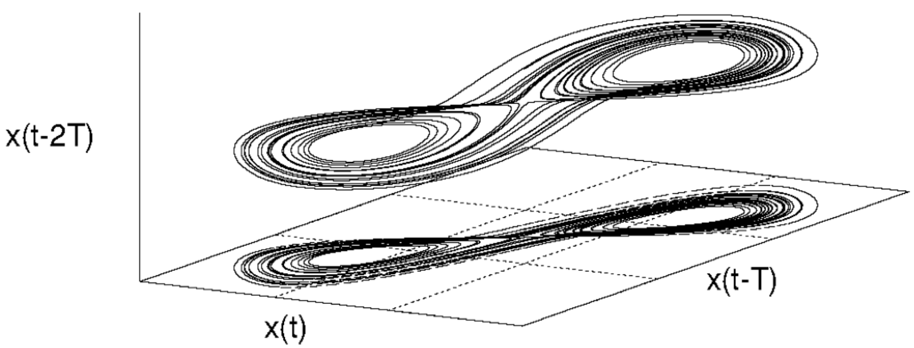


Lorenz attractor

What happens if we don't know the differential equations of a dynamical system?

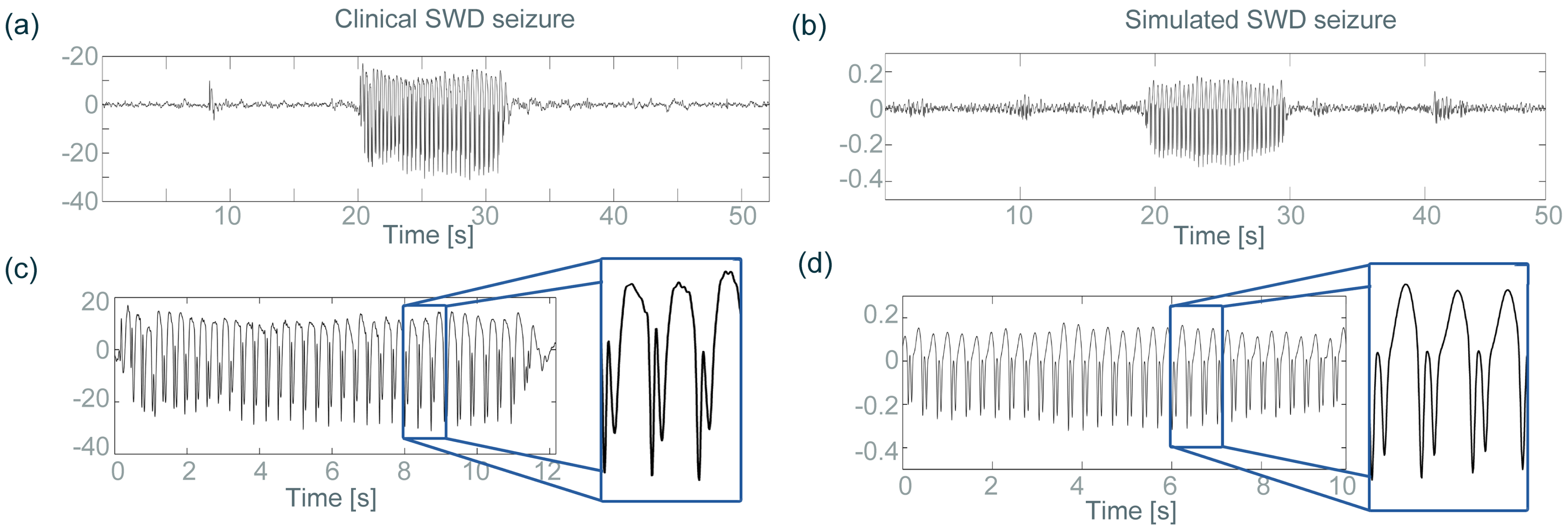
Takens Embedding Theorem proved that the time-delayed versions $[y(t), y(t-\tau), y(t-2\tau), \ldots, y(t-2n\tau)]$ of one generic signal would suffice to embed the n-dimensional manifold.



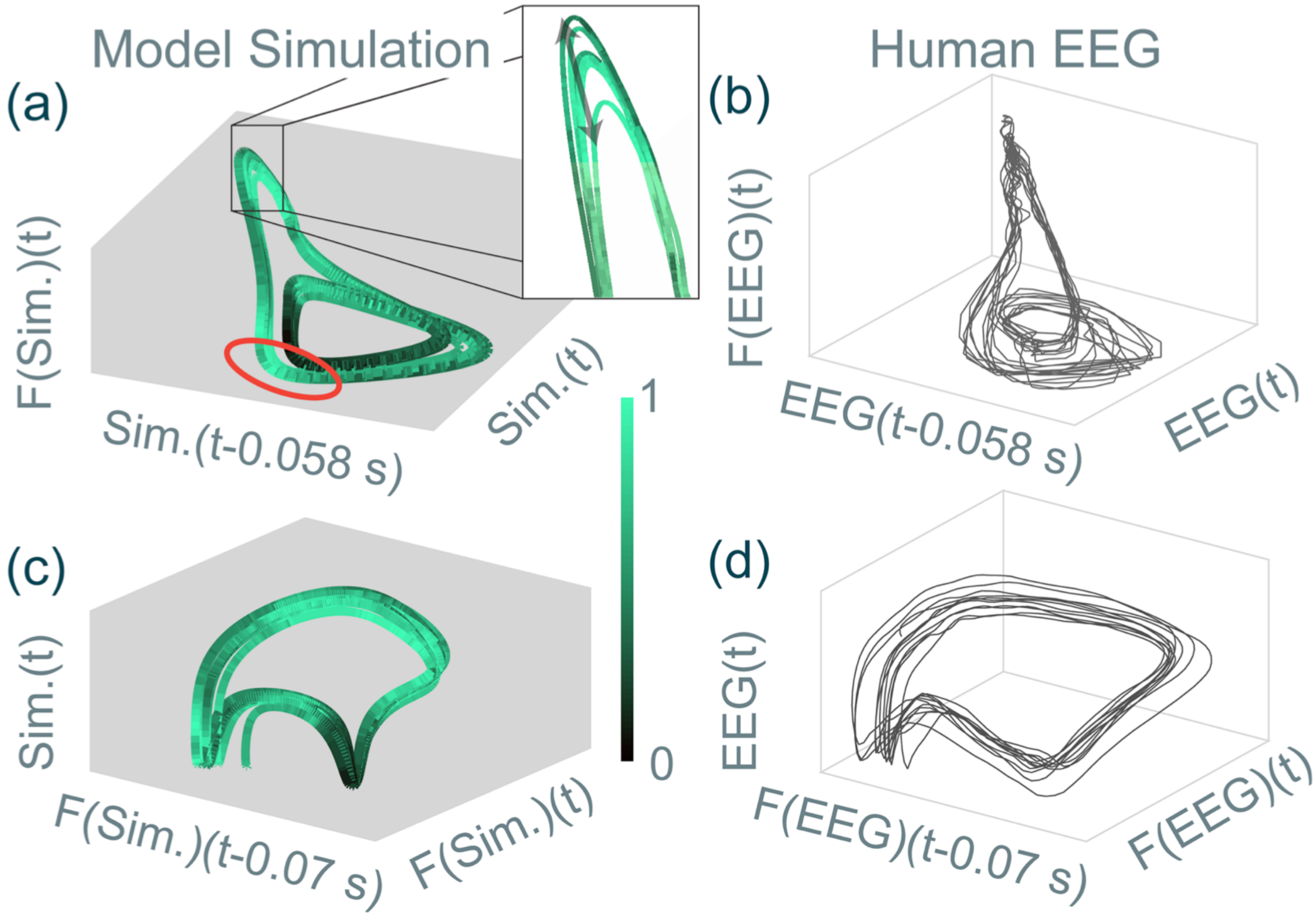


http://www.scholarpedia.org/article/Attractor_reconstruction

Applied to Spike-Wave seizures



Taylor et al. PLOS ONE, 2014 https://doi.org/10.1371/journal.pone.0114316



The simulated seizure (left) and the clinical seizure (right) are reconstructed. The same reconstruction parameters (delay time and filter frequency cut-off) have been used for both simulated EEGs (Sim.) and clinical EEGs (EEG). F(...) indicates low-pass filtering of the simulated EEG, as explained in the Methods section. Time delays used are indicated in seconds on the axis label. **(a, b)** Reconstructed attractor, in this case corresponding to the PY,IN,TC phase space view (c.f. Fig. 5 (a) rotated). **(c, d)** Reconstructed attractor corresponding to the TC,RE,PY phase space view (c.f. Fig. 5

Overview

- Noise and stochasticity
- How to relate experimental data to models
- Exercise: only deals with the lecture up until this point

- Dealing with higher dimensional systems
- Other ways of modelling: rule based models, discrete state models
- Best practice in modelling

Exercise 5

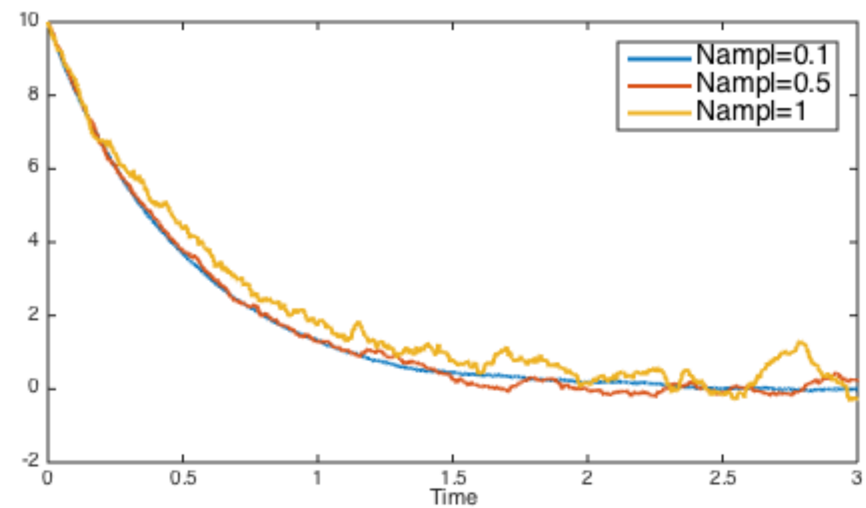
```
function dxdt = simpleode(t,x)

    dxdt=-2*x;

end



tspan=[0 3];
dt=.001;
initX=10;
Nampl=5;
```




```
T=tspan(1):dt:tspan(2);
X=zeros(size(T));
X(1)=initX;
for n=2:length(T)
    X(n)=X(n-1) + simpleode(0,X(n-1))*dt +…
         sqrt(dt)*randn(1)*Nampl;
end


plot(T,X)
```

Overview

- Noise and stochasticity
- How to relate experimental data to models
- Exercise

- Dealing with higher dimensional systems
- Other ways of modelling: rule based models, discrete state models
- Best practice in modelling

Higher dimensional systems

- 3D: fairly straight-forward to visualise and analyse (nullcline planes, vector fields)
- 4D+ : No good general recipes
- 4D+ : fixed points, stability, continuation

Systems built of components

- Analyse 2-3D components individually
- Analyse possible inputs from one component to the other and treat the input as a bifurcation parameters
- Beware of emergent behaviour though! The whole can be more than the sum of the parts.

Confidence in parameters

- IF you can have a high confidence in all the parameter values of a complex system, you will be able to limit the possible behaviour of the system and focus on the dynamics you are interested in.
- Parameters can be derived from experiments.
- Very rare that you will know them all though!

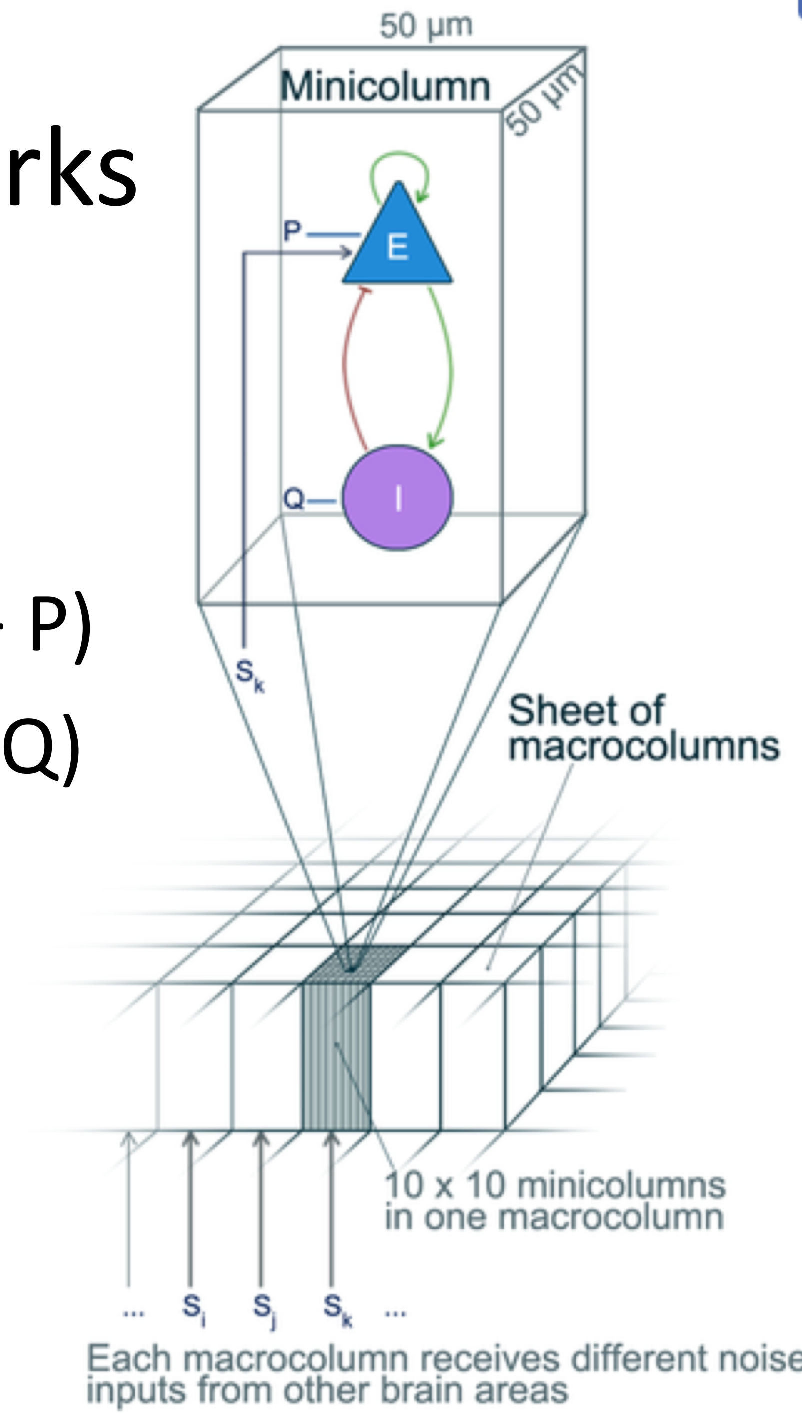
Systems using the same building blocks many times

- Assume the same parameters for the same building blocks.
- Analyse parameters in the building blocks as one ensemble parameter
- Again: beware of emergent behaviour!

Dynamics on networks

$$dE_n/dt = -E_n + S(A*E - B*I + P)$$

$$dI_n/dt = -I_n + S(C*E - D*I + Q)$$



You don’t need to know everything!

Your research question should dictate what type of analysis you will need to perform.

Overview

- Noise and stochasticity
- How to relate experimental data to models
- Exercise

- Dealing with higher dimensional systems
- Other ways of modelling: rule based models, discrete state models
- Best practice in modelling

Discrete time, discrete state

- Computationally often cheaper
- Often used for rule-based systems

- No direct mapping to an ode possible in the general case, but in special cases possible.

Rule-based models

- Rather than using an ODE to describe the evolution of a system over time:
- Use update rules to describe how a system evolves.
- Examples: The game of life, cellular automata

$$x_i(t+1) = \begin{cases} x_i(t) + \delta_1 bA\mathbf{x}(t) - \delta_2 d + \delta_3 p & x_i(t) \text{ is non-refractory}, \\ x_i(t) - d & x_i(t) \text{ is refractory}, \\ 0 \text{ and unit becomes non-refractory} & x_i(t) \leq 0, \\ 1 \text{ and unit becomes refractory} & x_i(t) \geq 1. \end{cases} \tag{4}$$

Rule-based models

- Knowledge of dynamic mechanisms can be incorporated without “parameter fiddling”.
- Good for describing systems, where update rules are more intuitive than continuous dynamics described by derivatives.
- Agent-based models as a special case: describing flock/group/swarm behaviour.
- Hybrids possible: Beware of funny behaviour at the hybridisation boundaries!
- Analytical/theoretical analysis tools less established, although some principles stay the same.
- Two models describing the same system, one rule-based, one ODE do not necessarily give the same results.

Overview

- Noise and stochasticity
- How to relate experimental data to models
- Exercise

- Dealing with higher dimensional systems
- Other ways of modelling: rule based models, discrete state models
- Best practice in modelling

Best practice in modelling

- Opinions diverge on this topic
- Some general views from my side
- Ultimately, “a model” is not a well-defined entity. For some people it is just a concept in their head. For others it is an equation system. For yet other people it is an animal.

Our views:

- At least one model output/variables should reflect the process you are trying to describe, not hidden states/variables. (Validation becomes difficult.)
- Validate your model – not with the assumptions you have put in.
- Predictions are desired to make the model useful, especially in a validated model.
- Be sure that you and others can reproduce your simulations exactly if required (seed random number generators, save parameters, initial conditions).
- A model should be as simple as possible, but not simpler. (If a mechanistic insight is sought.)
- Do you really need a model/simulation to prove your point?
- "All models are wrong, but some are useful."

Conflicts partly caused by different modelling philosophies:

Human Brain Project

- €1.2 billion project funding…

But it proved controversial from the start. Many researchers refused to join on the grounds that it was far too premature to attempt a simulation of the entire human brain in a computer. Now some claim the project is taking the wrong approach, wastes money and risks a backlash against neuroscience if it fails to deliver.

In an open letter to the European commission on Monday, more than 130 leaders of scientific groups around the world, including researchers at Oxford, Cambridge, Edinburgh and UCL, warn they will boycott the project and urge others to join them unless major changes are made to the initiative.

Finally…

Thank you for listening & participating!

Feedback is welcome:

Yujiang.Wang@ncl.ac.uk