Nonlinear Physics:
Modeling Chaos and Complexity
Spring 2010
Syllabus

Instructor: Prof. Jim Crutchfield (chaos@cse.ucdavis.edu; `http://cse.ucdavis.edu/~chaos`)
Assistant: Benny Brown (brown@cse.ucdavis.edu)
Time: 2:10 - 3:30 PM TuTh
Locations: 185 Physics (Tu) and 2118 Mathematical Sciences (Th)
WWW: `http://cse.ucdavis.edu/~chaos/courses/nlp/`

Parallel Theme I: Forms of Randomness, Order, and Intrinsic Instability

1. Qualitative Dynamics
2. Continuous-time ODEs and discrete-time maps
3. Bifurcations
4. Stability, Instability, and Chaos
5. Quantifying (In)Stability

Parallel Theme II: Tools for Exploring Chaos and Complexity

1. Modeling methods
2. Graphics
3. Simulation
4. Interaction
5. Programming

Prerequisites:

- Interest in modeling some dynamical phenomenon
- Vector calculus
- Linear algebra
- Lower division Math, Physics, or CS courses
- Programming: C/C++, Java, or Python (We will use Python.)
- Laptop with Python version 2.6 running

Readings:

- *NDAC* textbook: *Nonlinear Dynamics and Chaos*, Strogatz, 2001 Printing (Important!)
- *Python* textbook: *Learning Python*, Lutz & Ascher, Fourth Edition, 2009
- Course Lecture Notes (available via course website).

Contents

First Lecture (30 March, Tuesday): Overview

Readings:

- *Chaos*, JP Crutchfield, JD Farmer, NH Packard, RS Shaw, Scientific American **255** (1986) 46–57.
- *Odds*, Stanislaw Lem, New Yorker **54** (1978) 38–54.

Topics:

1. Introduction and motivations
2. Three parts: Dynamics, Bifurcations, Chaos
3. Survey interests, background, and abilities
4. Course logistics
5. Homeworks and projects
6. Software and program development

Homework Week 0: Everyday unpredictability; see website. Write-up due one week from Thursday, but be prepared to discuss at next Tuesday meeting.

Programming Lab A (1 April, Thursday): Python and Its Environment

Reading: *Python* Part I (Chapters 1-3) and Part II (Chapter 4, pp. 75-78, and Chapter 5).

Topics:

1. Modeling: Simulation, interaction, and graphics programming
2. Python language (Ch. 1)
3. Python and scientific computing packages installed and running (Ch. 2)
4. Developing and running Python using iPython (Ch. 3)
5. Python Data Types (Chapters 4 and 5)
6. Python as a calculator (WWW)

1 Qualitative Dynamics

Dynamics Lecture 2 (6 April, Tuesday): The Big Picture

Reading: *NDAC*, Chapters 1 and 2.

Topics:

1. Discuss *Chaos* and *Odds* readings and homework
2. Pendulum demo
3. Qualitative Dynamics: A geometric view of behavior
4. State space
5. Flows
6. Attractors
7. Basins
8. Submanifolds
9. Concrete, but simple example: One-dimensional flows

Programming Lab B (8 April, Thursday): Python, the Language

Reading: *Python* Part II (Chapters 4 and 7-9) and Part III (Chapters 11-13).

Topics:

1. Sequence Objects
2. Lists (Chapter 4, pp. 86-89, and Chapter 8, pp. 197-206)
3. Tuples (Chapter 4, pp. 96-97, and Chapter 9, pp. 225-228)
4. Loops (Chapter 13)
5. Expressions and conditionals (Chapters 11 and 12)
6. Text files (Chapter 4, pp. 229-238)
7. String operations (Chapter 7)

Homework: Collect Week 0's, assign Week 1's.

Dynamics Lecture 3 (13 April): Example Dynamical Systems

Reading: *NDAC*, Sections 6.0-6.7, 7.0-7.3, and 9.0-9.4.

Topics: Continuous-time ODEs

1. 2D Flows: Fixed points (Sec. 6.0-6.4)
2. 2D Flows: Limit cycles (Sec. 7.0-7.3)
3. 3D Flows: Chaos in Lorenz (Sec. 9.0-9.4)
4. Simulation demo
5. From continuous to discrete time (Sec. 9.4)
   (a) Poincaré Maps and Sections
   (b) Lorenz ODE to cusp map
   (c) Rössler ODE to logistic map (pp. 376–379)
   (d) Discrete-time maps

Programming Lab C (17 April): Arrays, Dictionaries, Functions, and Modularity

Reading: *Python* Part III (Chapters 4, 8, and 15), Part IV (Chapters 16-18), and Part V (Chapters 21-22) & course website (WWW).

Topics:

1. Dictionaries (Chapter 4, pp. 90-95, and Chapter 8, pp. 207-232)
2. Arrays (WWW)
3. Functions (Chapters 16-18)
4. Modules (Chapters 21-22)
5. Command line control (WWW)
6. Scripting (WWW)
7. Documenting code (Chapter 15)

Homework: Collect Week 1's, assign Week 2's.

2 Bifurcations

Dynamics Lecture 4 (20 April): The Big, Big Picture (Bifurcations & Catastrophes)

Reading: *NDAC*, Chapter 3.

Topics:

1. Qualitative Dynamics: Space of all dynamical systems
2. Example: Bifurcations of one-dimensional flows
   (a) Saddle Node
   (b) Transcritical
   (c) Pitchfork
3. Catastrophes: Fixed point to fixed point bifurcation
4. Example: Cusp Catastrophe
5. Catastrophe theory classification of fixed point bifurcations

Programming Lab B (22 April): Statistics, Linear Algebra, and Plotting

Reading: WWW.

Topics:

1. Statistics (WWW)
   (a) Fourier Transforms
   (b) Functions on a grid

   (c) Random numbers

2. Linear Algebra (WWW)
   (a) Vectors and matrices
   (b) Eigensystems
   (c) Root finding
3. Plotting (WWW)

Homework: Collect Week 2's, assign this week's (3).

Dynamics Lecture 5 (27 April): The Big, Big Picture (Bifurcations II)

Reading: *NDAC*, Chapter 8 and Sec. 10.0-10.4.

Topics:

1. Logistic map
2. Fixed point to limit cycle
3. Phenomenon and calculation
4. Limit cycle to limit cycle
5. Phenomenon and calculation
6. Routes to chaos: Period-doubling cascade
7. Phenomenon and calculation
8. Band-merging
9. Periodic windows and intermittency
10. Simulation demo
11. Bifurcations in ODEs:
    (a) Hopf bifurcation
    (b) Limit cycle to torus
    (c) Torus to chaos
    (d) Chaos to chaos

Programming Lab E (29 April): Plotting and One-Dimensional Dynamics

Reading: WWW.

Topics:

1. Plotting
2. Saving results
3. One-dimensional dynamics

Homework: Collect Week 3's, assign this week's (4).

Project: Pick project. Write up project proposal.

3 Visualizing and Quantifying Unpredictability

Dynamics Lecture 6 (4 May): Mechanisms of Chaos

Reading: *NDAC*, Sec. 12.0-12.3, 9.3, and 10.5.

Topics:

1. Chaotic mechanisms: Stretch and fold
2. Baker's map
3. Cat map (and stretch demo)
4. Henon map: stretch-fold and self-similarity
5. Rössler attractor branched manifold

Programming Lab F (6 May): Objects, Classes, and Error Handling

Reading: *Python* Part II (Chapters 4 and 6), Part VI (Chapters 25-31), and Part VII, Chapters 32-35)

Topics:

1. Data Types (Chapter 4)
2. Designing data types (Chapter 6)
3. Object-oriented programming (Chapters 25 and 30)
4. Architecture of simulation tools
5. Classes (Chapters 25-27)
6. Class attributes (Chapter 28)
7. Specializing Classes (Chapters 28 and 29)
8. Expending Classes (Chapter 31)
9. Error handling (Chapters 32-35)

Homework: Collect Week 4's, assign this week's (5).

Project: Project should be chosen and designed.

Dynamics Lecture 7 (11 May): Quantifying Chaos

Reading: *NDAC*, Sec. 12.0-12.3, 9.3, and 10.5.

Topics:

1. Dot spreading: Rössler and Lorenz ODEs
2. Lyapunov characteristic exponents (LCEs)
3. Time to unpredictability
4. Dissipation rate
5. Attractor LCE classification
6. Chaos defined

Programming Lab G (13 May): Numerical Integration and Visualization

Reading: *WWW*.

Topics:

1. Visualizing two-dimensional maps
2. Numerically integrating ODEs
    (a) Euler Integrator
    (b) Runge-Kutta Integrator
3. Three-dimensional visualization

Homework: Collect Week 5's, assign this week's (6).

Dynamics Lecture 8 (18 May): Analyzing Chaotic Maps & Routes to Chaos

Reading: *NDAC*, Chapter 10.

Topics:

1. Shift Map
2. LCEs for Maps
3. Tent Map
4. Logistic Map
5. LCE view of period-doubling route to chaos
6. Period-doubling self-similarity
7. Renormalization group analysis of scaling

Programming Lab H (20 May): Quantifying Chaos

Reading: *WWW*.

Topics:

1. Lyapunov Characteristic Exponents
2. For 1D Maps
3. For 2D Maps
4. For 3D Flows
5. 3D visualization

Homework: Collect Week 6's, assign this week's (7).

Dynamics Lecture 9 (25 May): From Determinism to Stochasticity—Probability Theory of Dynamical Systems

Reading: *Lecture Notes.*

Topics:

1. Probability theory review
2. Dynamical evolution of distributions
3. Invariant measures
4. Examples

Programming Lab I (27 May): Graphical User Interfaces

Topics:

1. TkInter
2. Widgets
3. WxPython

Homework: Collect Week 7's.

Dynamics Lecture 10 (1 June): Immersive visualization

Tour of KeckCAVES sensory immersive environment: `keckcaves.org`.

Programming Lab J (3 June): Spatially Extended Dynamical Systems

Topics:

1. Cellular automata in 1D and 2D
2. Lattice dynamical systems in 1D and 2D

4 Finish Projects and Present

1. Projects presented in class (fewer lectures)?
2. Projects presented in a one-day "workshop"?

Note: Project write-ups due at the end of the last week of classes, which is Friday 4 June.