PHY407: Computational Physics

Instructor:

Nicolas Grisouard

Teaching Assistants:

Alex Cabaj, Vince Pascuzzi, Haruki Hirasawa (marker)

Lecture:

Mondays: noon-1 MP134

Lab:

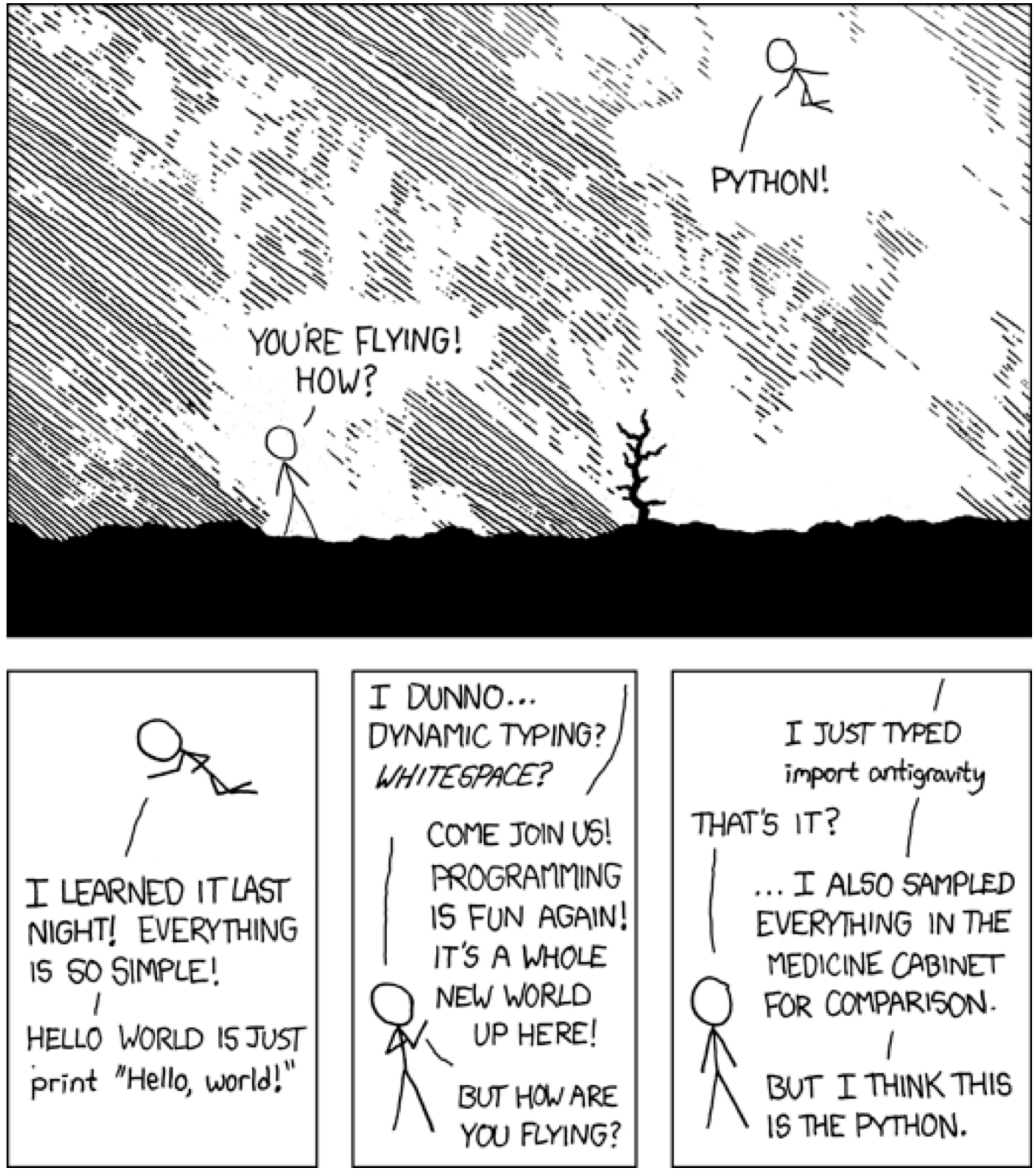
Wednesdays: 9-12 (tutorial 9-10, sometimes)

MP257 (& MP238)

Office Hour:

Thursdays: 1-3 MP257

(room booked noon-3)



xkcd.com/353

Syllabus

- On Quercus.

- There is a lot... Five documents:
  - The syllabus.
  - “Software”: instructions and recommendations about which tools to use,
  - “Assignment policy”: How to hand in your lab reports. Don’t get caught by surprise on Friday!
  - “Guidelines on evaluation of lab assignments”: what the marker will be looking for.
  - “Project”: guidelines about your end-of-term project.

Course Organization

- This course is an introduction to scientific computing in physics
- Lab-style course. Each week:
    - Lecture (1 hr/week): discussing theory, questions on text, pre-lab, and providing good computing practice tips
    - Expected to do text readings and pre-lecture quizzes before the lecture time
    - 3 hour lab. Sometimes, 30-50 min tutorial during the lab. Goal of labs: working through problems and adding instructions as necessary.
- **Work with a partner!** It can be a different partner each lab.
    - Please make sure to clearly attribute who did which work.
- Pre-lecture quizzes, required reading and Labs will be available on Fridays of previous week.
- Due dates: Pre-lectures: Mondays by 12:00pm. Labs: Fridays by 5:00pm

Course Organization

- **Work with a partner.**

Course Organization

- **DO YOUR LABS WITH A PARTNER!**

- **Half the work --- same marks!!**

Partners/Handing in Your Labs

- Please read the grading rubric for the labs.
- This year we are using the Quercus *group* function to hand in assignments.
- You need to sign up for a Quercus group. Let us know if you this is working for you.
- Remember your two options for assignments:
  1. No lab partner: Sign up for separate groups.
  2. Lab partner, hand in assignment together: You both sign up for one group. Label both lab partners on the assignment and assign credit (more info in the lab grading rubric).

Course Topics

- Range of computational techniques of interest to physicists:
  - Function analysis (derivatives and integrals)
  - Solving linear systems, eigenvalue problems, nonlinear systems
  - Fourier transforms
  - Solving ODEs
  - Solving PDEs
  - Monte Carlo techniques

- You will learn these methods through examples related to a range of physics topics. Examples covered in the past have included
  - Solving Schrödinger's equation (time independent and time dependent)
  - Shallow water (wavelength >> depth) equations for tsunami propagation
  - Ising model for magnetization
  - Cluster growth models
  - Phase transitions
  - Molecular dynamics
  - …

Course Expectations

- Prerequisite: PHY224H/250H
- This is a 3rd/4th year physics course. Background in 2nd/3rd year physics is assumed.
    - If you haven't taken several 2nd or 3rd year physics courses, please see me.
- Additional CSC courses helpful, but not assumed.
- This is not a course about software engineering or requiring deep computing expertise. But some Python background is expected and if you don't have that background you must be willing to learn it quickly or you will have trouble keeping up.
- Specifically, you need to be comfortable with the material in Chapters 2 & 3 of the text and the computational physics (replaces compwiki) tutorials before the second lab.
- There are a lot of labs, and we'll hit the ground running!

Course Structure & Evaluation

- 11 labs: (weeks 1-11). Introduced to lab in Monday lectures. Work on during Wednesday 3 hour lab time (plus before/after if necessary). Tutorial time is scheduled during labs to help you along. Hand in on Fridays.
  First lab is this week and due Friday!

- 10 pre-lecture online ‘quizzes’: (weeks 2-11). Answering a few short questions about the pre-lecture reading. Will be done online through Quercus. These are meant to ensure you’ve done the reading and have some preliminary exposure to the material in the lab.

- 1 final project (week 12 and afterwards). Will be due at the end of the course. This will give you some freedom to explore a computational problem in more depth in a topic of your choice (we’ll give some options).

- Grading Scheme:
  - Pre-lectures: (each of equal weight), 10% total
  - Labs: (each of equal weight), 66% total
  - Final Project: 24%

Course Resources

- Course webpage: via Quercus
  - all pre-lecture quizzes, labs, recommended readings, announcements, links etc. will be found here
  - you will also submit all labs through Quercus

- Text: Computational Physics (revised and expanded) by Mark Newman
  - should be available at bookstore, also available at amazon
  - first few chapters (covering first 2 weeks of course) along with some python programs you'll need are available at:

    http://www-personal.umich.edu/~mejn/cp/

- Computational physics site:
  - Replaces the old compwiki website. If you see something, say something.
  - introductory tutorials on python for physics
  - support email for issues related to downloading/installing python distribution

    https://computation.physics.utoronto.ca

Lecture notes

- I will sometimes use Jupyter notebooks to code during lectures.

- How to access them:
  1. On Quercus, a few days ahead of time, and
  2. On GitHub:

     https://github.com/PHY407-UofT/lectures

     The instructions on the first page of the GitHub repository will allow you to access them from even a smartphone, with the notebooks running on Compute Canada's servers.

A typical PHY407 week

- Monday, 11:55 am: submit your pre-lecture quiz on Quercus. **The deadline will be mercilessly enforced by Quercus!**
- Monday, noon: 1-hour lecture (MP134)
- Wednesday, 9-noon:
    - 1-hour tutorial (MP257) and 2-hour lab (MP257 and P238)
      OR
    - 3-hour lab (MP257 and 238)
- Thursday, 1-3: office hours in MP257 (room booked noon-3).
- Friday, 4:55 pm: submit your lab solution on Quercus.
- Friday, 4:56: take a break. Breathe in. Breathe out.
- Friday, 5:01 pm: the next lab and pre-lecture quiz are assigned. Start working on it. **You want to have advanced far enough on Wednesday to ask plenty of questions during Wednesday's lab and Thursday's office hour!**

How to Submit Assignments

- Labs and pre-labs are posted using an “Assignment” function on Quercus.
- You need to sign up to a Lab Group on Quercus. This will allow two partners to submit one assignment on Quercus, following the same “Assignment” link. Do not email us your solutions!
- First time we are trying Quercus: expect initial hiccups! Apologies in advance.
- We are looking for **C**^{3} solutions: **C**omplete, **C**lear, and **C**oncise
- Carefully read through the “Assignment Policy” (found on Quercus with the Syllabus) before submitting anything.
- Please see course materials for policy on academic integrity. Please make sure that if you are partnering with someone you clearly state who did which work, and that you share the work fairly.
- **See “Assignment Policy” sheet for more details, including grading expectations and policy on academic offences.**

How to Submit Assignments

- "Normally, students will be required to submit their course essays to Turnitin.com for a review of textual similarity and detection of possible plagiarism. In doing so, students will allow their essays to be included as source documents in the Turnitin.com reference database, where they will be used solely for the purpose of detecting plagiarism. The terms that apply to the University's use of the Turnitin.com service are described on the Turnitin.com web site. Note though that we never tested this feature with Quercus, nor, in this course, and may need to abandon this experiment early on."

- Turnitin does not make decisions. We do.

- You can opt-out. Let us know by Wednesday at noon.

- We are experimenting with it, and we may give up anyway.

How to Submit Assignments

- Do not use python 2!
- Uploading your lab report using .pdf and .py files: please test your code on a departmental machine!
- For your end-of-year project: if you intend to use more than **numpy**, **scipy, pylab** and/or **matplotlib**, ask us!

  We are not against it, but we need to make sure the marker has the right packages when he tests your code in December.

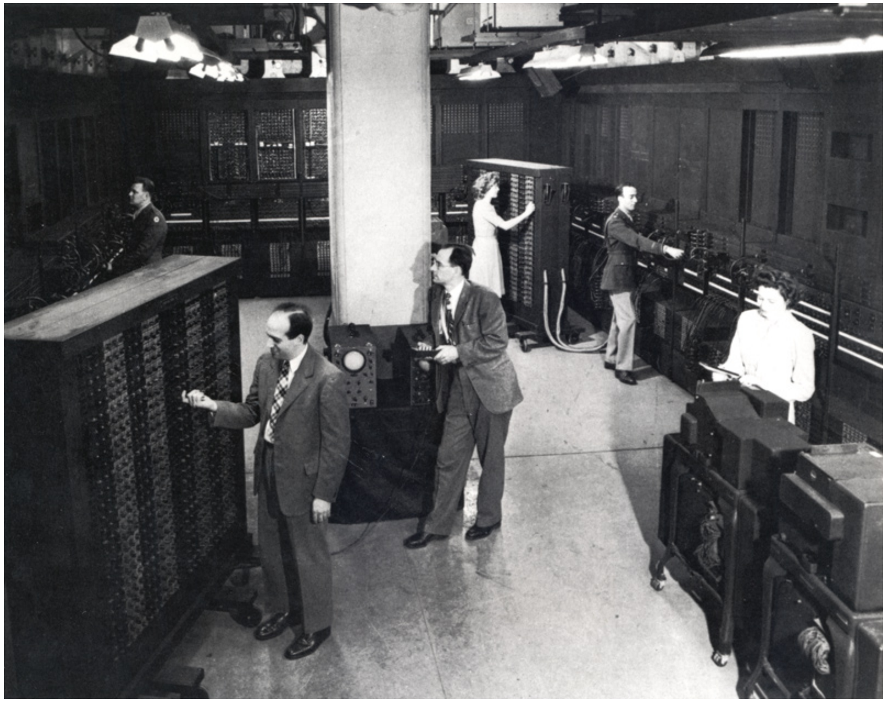
Lab coming on Wednesday

- Tutorial will start ~ 9:10 am in MP257 (in practice we start about 9:20).
- After tutorial we split into two rooms, MP238 and MP257.
- Download Lab01 (it is already online actually).
- Use lab computers or update your python distribution to Python 3 using instructions on https://computation.physics.utoronto.ca
- Or use your syzygy account (yes, you have one): https://utoronto.syzygy.ca

Digital computing has transformed our world and the way we do Physics

ENIAC (1946):
30 tonnes
100 kHz, 400 byte memory
18,000 vacuum tubes

http://www.computerhistory.org/revolution/birth-of-the-computer/4/78/319

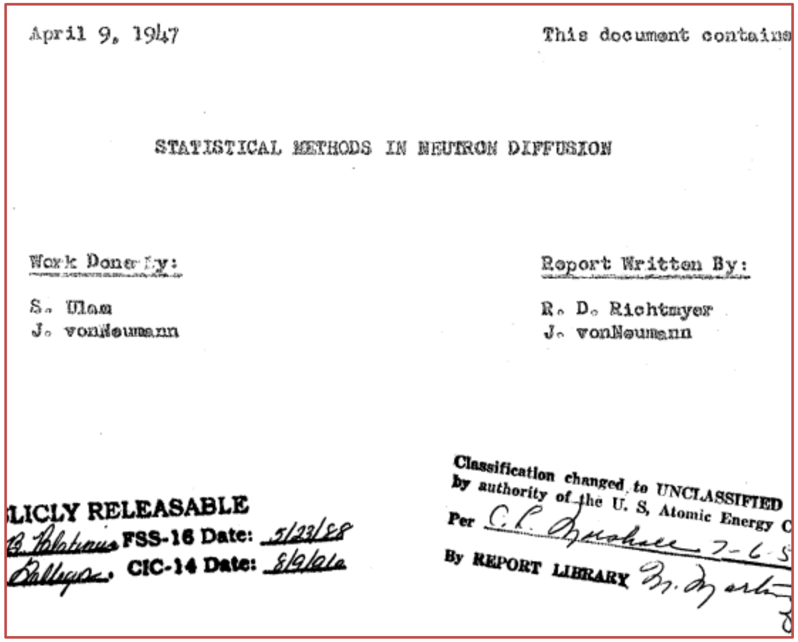


Raspberry Pi 3 B+ (2018):
45 g ~ 10^{-6} ENIACs
Quad Core 1.4 GHz ~ 10^{4} ENIACs each core
1GB ~ 10^{6} ENIACs

https://commons.wikimedia.org/wiki/File:Raspberry-Pi-2-Bare-BR.jpg
By Evan-Amos [Public domain], from Wikimedia Commons

Computational physics applications are more advanced …

1947: Monte Carlo statistical physics calculation (Ulam, von Neumann et al.) on ENIAC: computer cards, 100 degrees of freedom



April 9, 1947

This document contains

STATISTICAL METHODS IN NEUTRON DIFFUSION

Work Done By:

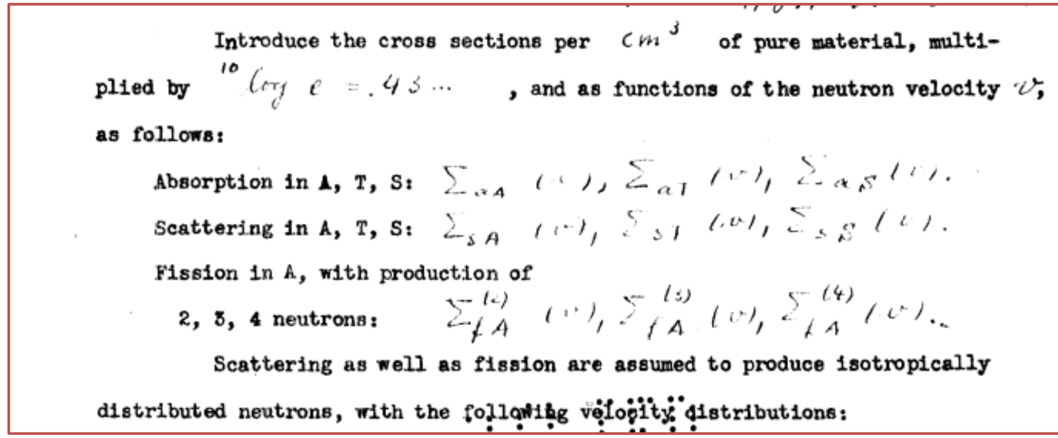
S. Ulam
J. vonNeumann

Report Written By:

R. D. Richtmyer
J. vonNeumann

LICLY RELEASABLE
FSS-16 Date: 5/23/88
CIC-14 Date: 8/9/96

Classification changed to UNCLASSIFIED
by authority of the U. S. Atomic Energy C
Per 7-6-5
By REPORT LIBRARY



Introduce the cross sections per cm^3 of pure material, multiplied by $^{10}\log e = .43\ldots$, and as functions of the neutron velocity v, as follows:

Absorption in A, T, S: $\Sigma_{aA}(v), \Sigma_{aT}(v), \Sigma_{aS}(v)$.

Scattering in A, T, S: $\Sigma_{sA}(v), \Sigma_{sT}(v), \Sigma_{sS}(v)$.

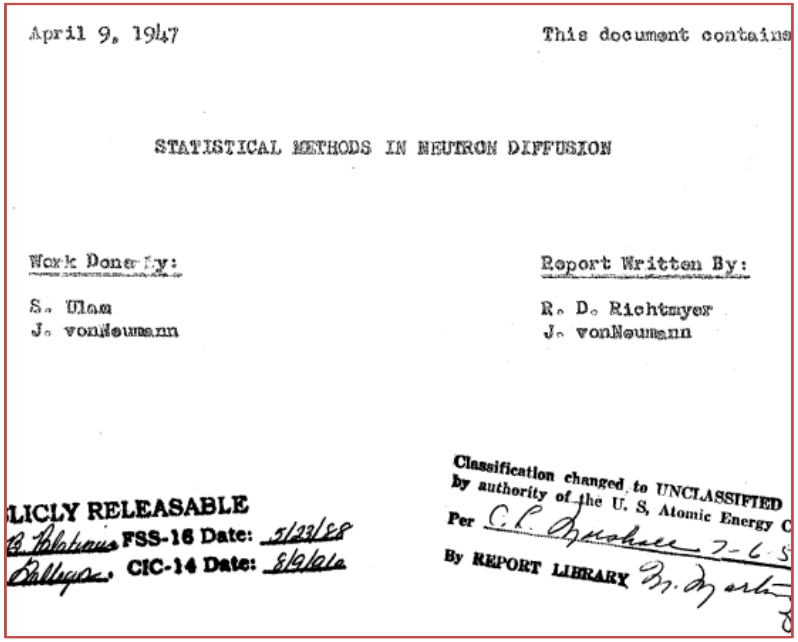
Fission in A, with production of

2, 3, 4 neutrons: $\Sigma_{fA}^{(2)}(v), \Sigma_{fA}^{(3)}(v), \Sigma_{fA}^{(4)}(v)$..

Scattering as well as fission are assumed to produce isotropically distributed neutrons, with the following velocity distributions:

http://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-00551-MS

1947: Monte Carlo statistical physics calculation (Ulam, vonNeumann et al.) on ENIAC: computer cards, 100 degrees of freedom



April 9, 1947

This document contains

STATISTICAL METHODS IN NEUTRON DIFFUSION

Work Done By:

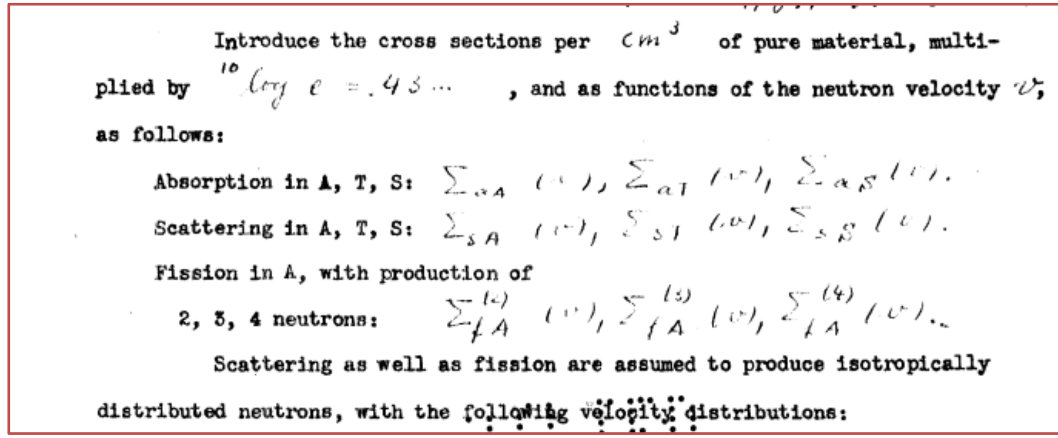
S. Ulam
J. vonNeumann

Report Written By:

R. D. Richtmyer
J. vonNeumann

LICLY RELEASABLE
FSS-16 Date: 5/23/88
CIC-14 Date: 8/9/96

Classification changed to UNCLASSIFIED
by authority of the U. S. Atomic Energy C
Per 7-6-5
By REPORT LIBRARY



Introduce the cross sections per cm^3 of pure material, multiplied by $^{10}\log e = .43\ldots$, and as functions of the neutron velocity v, as follows:

Absorption in A, T, S: $\Sigma_{aA}(v), \Sigma_{aT}(v), \Sigma_{aS}(v)$.

Scattering in A, T, S: $\Sigma_{sA}(v), \Sigma_{sT}(v), \Sigma_{sS}(v)$.

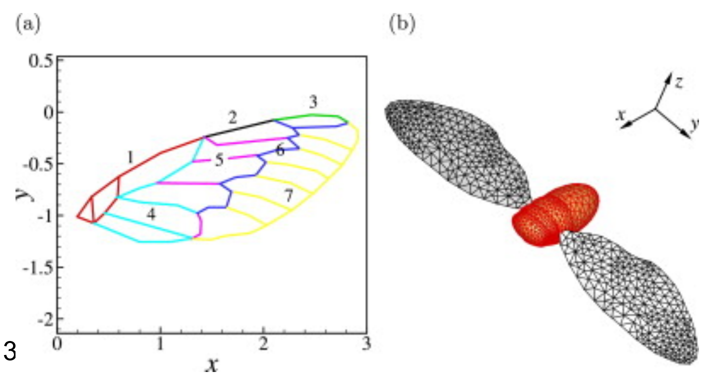
Fission in A, with production of 2, 3, 4 neutrons: $\Sigma_{fA}^{(2)}(v), \Sigma_{fA}^{(3)}(v), \Sigma_{fA}^{(4)}(v)$.

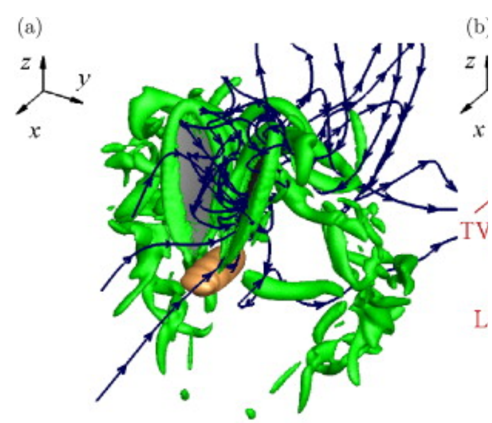
Scattering as well as fission are assumed to produce isotropically distributed neutrons, with the following velocity distributions:

http://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-00551-MS

2014: 3D simulation of fluid-solid interactions for biological physics. 10^6 grid boxes, reasonably powerful workstation.

Journal of Computational Physics, Volume 258, 2014, 451–469





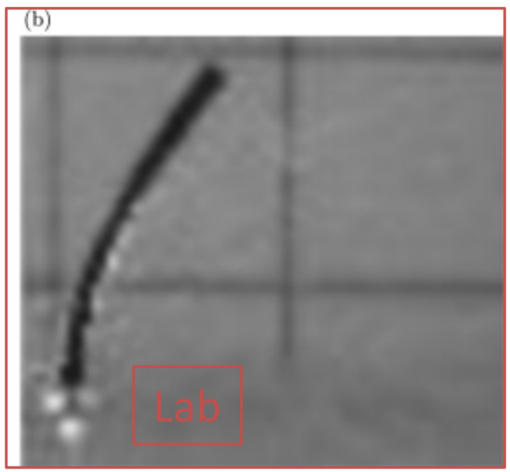
http://dx.doi.org/10.1016/j.jcp.2013047

Computational physics applications are more advanced …

… but the approach and style haven’t changed a lot.

Computational physics applications are more advanced …

Data from lab and field obs



(Fluid flow past a plate.)

… but the approach and style haven't changed a lot.

Computational physics applications are more advanced …

Data from lab and field obs

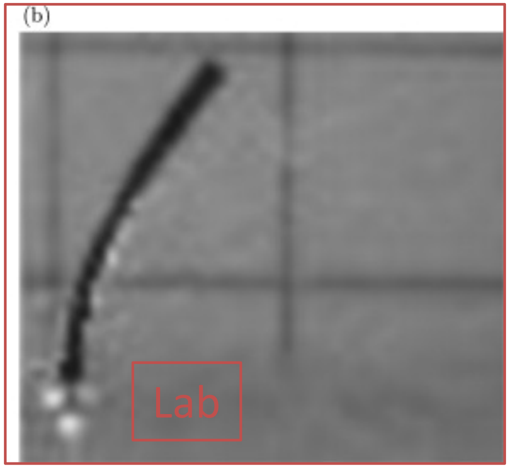
Mathematical model
(Empirical/dynamical)

$$\rho_s \frac{d^2 u_i}{dt^2} + \eta_d \frac{du_i}{dt} = \frac{\partial \sigma_{ij}}{\partial x_j} + \rho_s b_i,$$

Numerical method
Algorithm
Code/Program

$$\left[K_T + \frac{\gamma}{\beta \Delta t} C + \frac{1}{\beta \Delta t^2} M\right] \left\{\Delta u^{n+1,s+1}\right\} = \left\{P^{n+1}\right\} - \left\{F^{n+1,s}\right\} - [M] \left\{\ddot{u}_n + \frac{1}{\beta \Delta t^2} \left(u^{n+1,s} - \tilde{u}\right)\right\} - [C] \left\{\tilde{v} + \frac{\gamma}{\beta \Delta t} \left(u^{n+1,s} - \tilde{u}\right)\right\},$$

Computation
Simulation
Analysis



… but the approach and style haven't changed a lot.

Computational physics applications are more advanced …

Data from lab and field obs

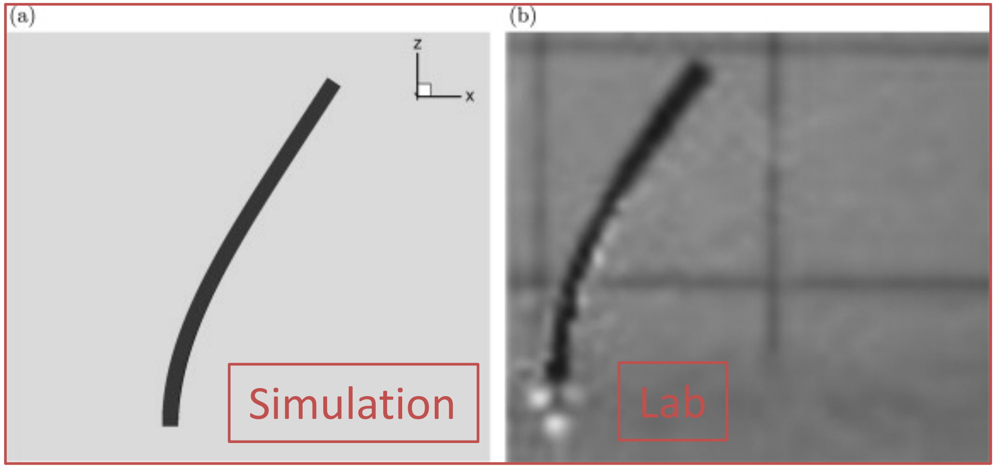
Mathematical model
(Empirical/dynamical)

$$\rho_s \frac{d^2 u_i}{dt^2} + \eta_d \frac{du_i}{dt} = \frac{\partial \sigma_{ij}}{\partial x_j} + \rho_s b_i,$$

Numerical method
Algorithm
Code/Program

$$\left[K_T + \frac{\gamma}{\beta \Delta t} C + \frac{1}{\beta \Delta t^2} M\right]\{\Delta u^{n+1,s+1}\} = \{P^{n+1}\} - \{F^{n+1,s}\} - [M]\left\{\ddot{u}_n + \frac{1}{\beta \Delta t^2}(u^{n+1,s} - \tilde{u})\right\} - [C]\left\{\tilde{v} + \frac{\gamma}{\beta \Delta t}(u^{n+1,s} - \tilde{u})\right\},$$

Computation
Simulation
Analysis



Hey, it looks pretty good!
My simulation checks out!

… but the approach and style haven't changed a lot.

Computational physics applications are more advanced …

Data from lab and field obs

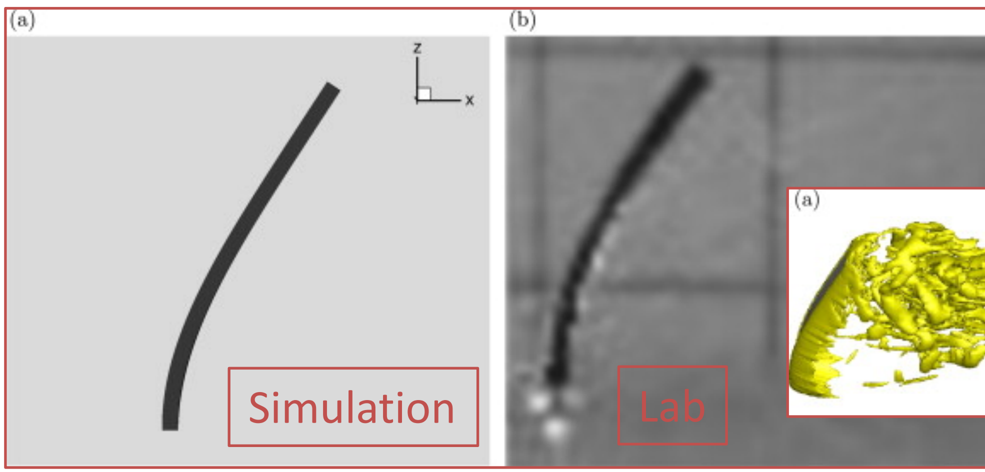
Mathematical model
(Empirical/dynamical)

$$\rho_s \frac{d^2 u_i}{dt^2} + \eta_d \frac{du_i}{dt} = \frac{\partial \sigma_{ij}}{\partial x_j} + \rho_s b_i,$$

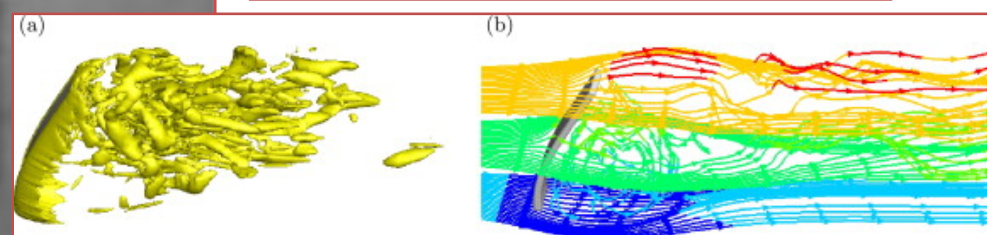
Numerical method
Algorithm
Code/Program

$$\left[K_T + \frac{\gamma}{\beta \Delta t} C + \frac{1}{\beta \Delta t^2} M\right] \{\Delta u^{n+1,s+1}\} = \{P^{n+1}\} - \{F^{n+1,s}\} - [M]\left\{\ddot{u}_n + \frac{1}{\beta \Delta t^2}(u^{n+1,s} - \tilde{u})\right\} - [C]\left\{\tilde{v} + \frac{\gamma}{\beta \Delta t}(u^{n+1,s} - \tilde{u})\right\},$$

Computation
Simulation
Analysis



In-depth analysis of simulation



… but the approach and style haven't changed a lot.

Computational Physics style

- Experimental and theoretical – a third branch of physics.
- Exploratory
- Iterative.
- A bit informal and freewheeling.

Summary

- Advances in hardware and software has made computational physics far more accessible and a standard part of the working physicists toolkit.
- Hands-on, practical, experimental, iterative, collaborative (**work with a partner!**).
- Purpose of this course: develop working physicist’s toolkit for numerical methods.
- Download and start working on the labs, and see you Wednesday.