Introduction to Fluid Dynamics

Remote MWF: MW asynchronous, F 10:20-11:10am Zoom

Contact info :
Benjamin Brown
Laboratory for Atmospheric and Space Physics (LASP)
Department of Astrophysical and Planetary Sciences (APS)
email: bpbrown@colorado.edu
office hours: Slack channel and Zoom

course web presence: Canvas, RC JupyterHub,
Zoom (https://cuboulder.zoom.us/j/94833881576) and Slack (astr5400fluids.slack.com).

Why fluids?

Fluid dynamics underlies many interesting astrophysical problems. Stars sing in hydrodynamic waves, and by listening to their songs we learn about their interior properties. Planets form in accretion disks, where swirling flows concentrate or rarify the raw materials of world formation. The jets of AGN arise from magnetohydrodynamic interactions between plasma flows and magnetic fields, and evolve following basic hydrodynamic processes. Though fluids are everywhere, our expertise in them is not strong. When astronomers say “rotation” or “magnetic fields” do something magical, what they generally mean is “fluid interactions with rotation and/or magnetic fields” do something simple. Here we aim to transform magic into reason.

This course will provide a foundation for your other graduate work, but more importantly, will provide you with a rich toolkit for tackling hard research problems in the future. We will study fluid dynamics in both nonlinear situations, where we will need to perform numerical simulations, and in linear systems where tools of linear algebra will see practical application. We will learn how the fluid equations are partial differential equations (PDEs) and review practical solution strategies, computationally and analytically.

I have three major goals in this course. First, I want you to emerge with a broad set of tools for tackling problems which you will encounter in your future research. Many research problems require a diversity of approaches, and we’ll build fluency with these different approaches here. Second, I want you to feel prepared for further learning in fluid dynamics. This course, by nature, must lightly touch many subjects. Coming out of it however, you will have the fluency to dive deeper into subjects relevant to your research, and familiarity with ways to search for publications in applied math and physics. Third, I want you to emerge from this class with confidence in your computational skills. We will learn and use python, latex, version control for code, and unix-based systems in the course of our work, as these form the foundations of modern research environments.

Content

- INVISCID FLUIDS *derivation and examples of the Euler equations*
  - Continuum hypothesis
  - Eulerian and Lagrangian formulations of fluid flows Inviscid Euler equations
  - Streamlines, streamfunctions
  - Examples of inviscid flows
  - Inviscid flows and numerical solutions
- VISCOUS FLOWS *derivation of the Navier-Stokes equations including the energy equation*
  - Relationships between stress and strain, the stress tensor
  - Navier-Stokes equations: continuity, momentum, energy Reynolds number
  - Transformation to non-inertial frames (e.g. rotating frames)
  - Bernoulli's equations
- VORTICITY *definition and significance of vorticity in fluid flows*
  - Vortex dynamics
  - Kelvin's circulation theorem
  - Taylor-Proudman theorem
  - Potential vorticity
- GRAVITY WAVES *concepts required to deal with waves*
  - Linearization
  - Phase and group velocities, concept of the dispersion relation
  - Gravity waves
  - Deep and shallow water waves
- COMPRESSIBLE FLOWS *compressible fluid dynamics including shocks and sound waves*
  - Thermodynamics of compressible flow 1D flow examples
  - Sound waves
  - Shock waves and jump conditions
  - Weak and strong shocks
  - Sedov solution
- INSTABILITIES *when waves grow*
  - Linearization of the fluid equations and growth rate of perturbations
  - Kelvin-Helmholtz instability
  - Rayleigh-Taylor instability
  - Rayleigh-Benard convection
- BOUNDARY LAYERS *especially important for stars, planets and rockets*
  - Concepts of boundary layers
  - Self-similar solutions for viscous flows
  - Jets
  - Boundary layer separation

Course structure

Course: This course will be taught in a fully remote context. We will use a mixture of asynchronous (two days a week) and synchronous instruction (one day a week, Fridays). We will also do some hands-on experiments during the class, using common materials from homes and kitchens. We'll discuss this in advance and come up with individual alternatives if this is a barrier.

- Slack workspace: astr5400fluids.slack.com

Asynchronous Class: The asynchronous material is a crucial part of your learning in the course. Asynchronous instruction will likely include a mixture of recorded short lectures and guided interactive exercises using Jupyter notebooks. We'll adapt this as we proceed through the semester and see what is working better or worse. You will be expected to (asynchronously) interact with me on each of our asynchronous days (MW) via our shared Slack workspace and our Canvas page.

Synchronous Class: Our synchronous classes will be conducted via Zoom on Fridays. These days will likely be less-lecture, more hands-on work on projects and experiments and guided discussion. We will have readings from the textbook, and possibly online material, that I expect you to have studied before class. These will be communicated to you, and I will expect you to please come to class with questions. We will all learn more through interaction and discussion. I won't know all of the answers. When I don't, I'll find out and bring them to you. If you need to miss class, please discuss this with me in advance so we can make accommodations.

Homework: Early in the semester, there will be homework almost every week, handed out and collected on Wednesdays. Most homeworks will involve some computational aspect, using resources at CU's Research Computing. Please collaborate and work together, but please write up your work (and write your code) individually. Homework will be written up in Jupyter notebooks and will be handed in via our shared RC Jupyterhub workspace.

Final Exam: there will be a final exam. It will be a timed, asynchronous, and will be in April.

Final project: each of you will do a final project studying an interesting fluids problem. These can be computational, experimental, or observational (of fluids in our everyday world). You will give a short, AAS-style talk (5 min presentation, 3 min questions) about your approach and your findings. These will be done in the format of current online meetings to give you practice in these formats in a safe space. This will occur during Finals week. This is in addition to the final exam (above).

Note taking: This class does not have a single topic, and our textbook does not capture everything we will cover. Instead, we are covering a variety of subjects, and this can make it difficult to figure out what exactly we covered when it's time to study for your comprehensive exam.

To help remove this concern, we as a class are going to generate a set of notes from the lectures. Each student will take notes on one or more classes and type those notes up a presentable fashion in whatever composer you prefer (Latex, word, Jupyter, etc.). When they are in sharable form, you will e-mail them to me as both a PDF and also your original document and then I will upload them on our shared Canvas site. Notes are due within two days of class (e.g., if you take notes Monday, please get them to me Weds evening), with an additional day allowed if you take notes Friday (due Tues evening), and notes will be taken for our asynchronous days as well as our synchronous meetings.

Your notes do not need to be works of literature nor art. I will however return your notes to you for revision if they are not clear enough for your fellow students to follow. If there is a minor clarity or typo error, I will revise the original document myself directly, but if the confusion is more substantial, they will come back to you for revision.

The exact number of classes you need to cover will depend on how many of us there are, but I'm estimating it at 1-2 classes each, and we will proceed through the roster alphabetically. Overall, note-taking is worth about 1 homework of credit and I expect about that level of effort (~ 6 hours total).

Grading: about 50% of your grade will comes from assignments and note-taking, 20% from the final exam, and 20% from the final project. The last 10% is from participation, especially on asynchronous days.

week 1	Static solutions to fluid equations
weeks 2–3	Thermodynamics and energy equations
week 4	Vorticity & Kelvin circulation theorm
week 5	Bernoulli & non-viscous flows
weeks 6-7	Viscous flows
week 8	Non-dimensionalization & instability
weeks 9-10	Linear waves
week 11	Non-constant coefficient atmospheres
week 12	Rotating fluid dynamics
week 13	Geophysical and Astrophysical Fluid Dynamics

Table 1: Approximate flow of the course; subject to change as we progress depending on where we have understanding or need more instruction.

Technical Computing

You will need access to a computer (preferably a unix/linux or Mac), and during some of our synchronous days we will be doing hands-on exercises using your machines. In this course, we will do all of our work in Python. Why Python? Traditionally, many astronomers, especially here in Colorado, have used IDL. IDL itself is a closed source platform, and it is difficult to execute parallel processing in IDL. There is a growing trend away from IDL and towards Python. Python is a free, open source, high-level interactive interpreted computing and scripting environment. Python is used for much more than scientific computing. One can easily wrap existing C, C++, or Fortran codes. Within the astronomical community, it is used for everything from telescope observing scripts to quick interactive data visualization, to sophisticated and complex analysis pipelines with hundreds of thousands of lines of code. In this course you will use and learn `python3` (version 3); there are subtle array promotion differences between `python2` and `python3`, and portions of our numerical work will leverage the Dedalus pseudospectral framework, which itself uses python3. To reduce the difficulty of deploying python on different laptops, we will use shared resources on Research Computing's JupyterHub.

Books to learn from

Fluids

- *The Physics of Fluids and Plasmas*, Choudhuri, 1998
  Required course textbook; a good, broad introduction to fluid dynamics with an astrophysical slant to the material. Think more helioseismology, supernova shock waves, and convection, less pipe flow turbulence and wings (though some examples of those too). Covers basic plasma physics and magnetohydrodynamics.
- *Hydrodynamic and Hydromagnetic Stability*, Chandrasekhar, 1961 **Required course textbook;** covers instabilities (like convection) in astrophysically important systems. Critical text to have in your library.

- *Fluid Mechanics* Landau & Lifshitz 1959 (1966 for 3^{rd} ed)
  Volume 6 (of 9) in Landau & Lifshitz's sweeping "Course of Theoretical Physics". Everything is here, if you can understand it.
- *Physical Fluid Dynamics* Tritton 1988
  Beautiful book on incompressible flow with a focus on geophysical fluid dynamics. Highly recommended by one of the best fluid dynamicists I know.
- *An Introduction to Astrophysical Fluid Dynamics* Thompson 2006
  Compact book with strong focus on astrophysical fluid dynamics; close second runner for main course textbook. Great material on stellar oscillations (waves).

Math

- *A First Course in Numerical Methods*, Ascher & Greif, 2011
A broad introduction to linear algebra, numerical techniques, differential equations, etc. Good reference for understanding packaged library routines (e.g., QR factorization, Krylov methods, etc.) and the code examples, though in Matlab, are helpful. Freely available as a PDF from a campus IP address (http://epubs.siam.org/doi/book/10.1137/9780898719987).

Numerical techniques and coding

- O'reilly, "Linux in a Nutshell," or "Unix in a Nutshell," is a comprehensive linux/unix command, shell, and text editor reference.
- Press et al, "Numerical Recipes 3rd Edition." In addition to offering specific numerical algorithms, this book contains excellent introductory text on statistics and data analysis techniques. Beware the restrictive licensing clauses in this book.

Python

- Basic Python tutorial: http://docs.python.org/tutorial/
- Scientific computing using Python: Scipy and Numpy: http://scipy.org/
- Using Python for Interactive Data Analysis in Astronomy: http://stsdas.stsci.edu/perry/pydatatut.pdf
- Interactive computing with Python using Jupyter: https://jupyter.org
- Matlab-like interactive plotting with Python: http://matplotlib.sourceforge.net/
- UC Berkeley Astronomy Department's Python Boot Camp: https://sites.google.com/site/pythonbootcamp/
- O'Reilly's "Learning Python," 3rd edition: http://oreilly.com/python/
- Resources for installing a base python system (with python, numpy, scipy, etc.):
  - Dedalus stack: http://dedalus-project.readthedocs.io/en/latest/installation.html
  - Anaconda: https://www.continuum.io/downloads

Latex

- *A (Not So) Short Introduction to LaTeX 2$_\varepsilon$*, Oetiker
https://www.ctan.org/tex-archive/info/lshort/english/ or
https://tobi.oetiker.ch/lshort/
Good go-to reference for using Latex. Freely available as PDF.