UW–Madison Math/CS 714
Methods of Computational Mathematics I

General information

Credits:	3
Course description:	Development of finite difference methods for hyperbolic, parabolic and elliptic partial differential equations. Analysis of accuracy and stability of difference schemes. Direct and iterative methods for solving linear systems. Introduction to finite volume methods. Applications from science and engineering.
Requisites:	Graduate/professional standing or member of the Pre-Masters Mathematics (Visiting International) Program *Students are strongly encouraged to have programming skills (e.g. CS 200) and some undergraduate numerical analysis (e.g. MATH/CS 514 or CS 412), analysis and differential equations (e.g. MATH 322 and MATH 521) and linear algebra (e.g. MATH 341).*
Meeting time and location:	Tue/Thu, 11am–12:15pm B135 Van Vleck Hall
Instructional modality:	In-person
Instructor contact info:	Dr. Yue Sun, yue.sun@wisc.edu
Office hours:	Wed 12:45pm–1:45pm (725 Van Vleck Hall) Thu 12:45pm–1:45pm (725 Van Vleck Hall)
Grader:	Ann Bigelow, ann.bigelow@wisc.edu
Websites:	https://web.math.wisc.edu/math714/

Course overview

Math/CS 714 and Math/CS 715 are concerned with the development and the analysis of numerical methods for solving elliptic, parabolic, and hyperbolic partial differential equations (PDEs). In particular, Math/CS 714 will focus on a class of numerical schemes known as finite difference methods (although finite volume and spectral methods will also be covered). For each type of PDE we will develop numerical schemes from physical and mathematical intuition, analyze these schemes, consider improvements, and discuss issues related to the implementation of these methods into computer code.

An important aspect of accurately and efficiently solving PDEs via numerical methods is the ability to solve large linear systems. Therefore, in addition to the basic numerical methods, this course will include discussions of simple iteration methods for solving large linear systems. Homework assignments will involve problems on theoretical (analyzing methods) and practical (implementing methods into computer code) aspects of numerical analysis.

Learning outcomes

- Analyze the theoretical properties of a numerical algorithm, such as its stability, consistency, and convergence.
- Implement a numerical algorithm in computer code.
- Illustrate the properties of a numerical algorithm (e.g., stability, consistency, convergence) using a computer code.
- Analyze algorithms for a variety of classes of equations, such as ordinary differential equations (ODEs), and partial differential equations (PDEs) of elliptic, parabolic, and hyperbolic types.
- Analyze a variety of classes of algorithms, such as finite difference methods, finite volume methods, and spectral methods.
- Evaluate the advantages and disadvantages of numerical algorithms, in terms of properties such as order-of-accuracy and computational cost.

Course website and online platforms

All course material will be posted on the main course website, and will also be synchronized with the Canvas site. The online messaging platform Piazza will be used for class annoucements and questions about the homework, projects, and general logistics. The Piazza site is linked to from the Canvas site. Code examples for lectures and optional activities will be distributed from a Git repository on GitHub.

Textbooks and additional references

The majority of the course will follow the required textbook *Finite Difference Methods for Ordinary and Partial Differential Equations* by Randall J. LeVeque. The final month of the course will use the textbook *Finite Volume Methods for Hyperbolic Problems* by Randall J. LeVeque, but all required material from this book will be presented in lecture, and it will not be necessary to obtain a copy. The last topic of the course will use the textbook *Spectral Methods in MATLAB* by Lloyd N. Trefethen, which is available for free online. It is not necessary to purchase a copy of this book, as all required material will be presented in lecture. The following references may also be useful:

- E. Süli and D. Mayers. *An Introduction to Numerical Analysis.* Cambridge University Press, 2003.
- M. T. Heath. *Scientific Computing: An Introductory Survey.* McGraw-Hill, 2001.

- W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery. *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press, 2007.
- J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis.* Springer, 2002.
- E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems.* Springer, 1993.

The material in this course uses numerical linear algebra, which are covered in detail in these two books:

- L. N. Trefethen and D. Bau. *Numerical Linear Algebra.* SIAM, 1997.
- J. Demmel. *Applied Numerical Linear Algebra.* SIAM, 1997.

The previous instructor, Professor Chris H. Rycroft (who is on parental course relief), has previously taught a year-long sequence in numerical analysis at Harvard University at the graduate level. These course websites are now hosted at UW–Madison:

- Harvard Applied Math 205: `https://people.math.wisc.edu/~chr/am205/`
- Harvard Applied Math 225: `https://people.math.wisc.edu/~chr/am225/`

These courses have extensive notes available. In particular, *Applied Math 205* has a complete sequence of videos on YouTube. Units 0, 1, & 2 are useful background. Math/CS 714 will cover many of the topics in Unit 3.

Homework

The course will have five homework assignments due every 2–3 weeks, with deadlines on Fridays at 5pm Central Time. Homework should be uploaded via Canvas. Extension requests on homework can be sent to Yue Sun, and will be evaluated on a case-by-case basis.

The course will be graded by Ann Bigelow. Small issues or clarifications with grading can be directed to the graders. In most cases, homework regrades will not be considered. All five homework assignments will count toward the final grade.

Students should carefully read the UW–Madison academic integrity policy included in later in this document. Collaboration on homework assignments and discussion with other students is encouraged, but each student must write solutions in the student's own words, and not copy them from anyone else. In addition, you must cite any books, articles, websites, lectures, *etc.* that have helped you with your work using appropriate citation practices. Similarly, you must list the names of students with whom you have collaborated. Use of generative AI tools (such as ChatGPT, Copilot, etc.) is not allowed for the writeup and code.

A note about writing

If you read any textbook or scientific article, you will see that there is a standard way of writing mathematics, in which equations form part of full sentences and occur naturally with the flow of the text. While completely optional, you may wish to practice this in writing your assignments. In general, the course homework assignments and notes posted on Canvas will adhere to this standard.

One way to write mathematics like this is to make use of the free software package LaTeX, which can produce very high quality scientific documents, and is used extensively by mathematicians, physicists, and engineers. LaTeX can be installed on all major computing platforms. There is an excellent guide, The Not So Short Introduction to LaTeX2$_\epsilon$, which can be used to get started. The homework assignments and solutions are all written in LaTeX.

Optional activities

Since the students in Math 714 have a wide range of backgrounds, the course will feature several optional activities. These will consist of 2 h workshops, covering either useful background material, or topics that go beyond the standard course material. A tentative schedule is shown below.

Date	Topic
Sep 10	Introduction to C++
Sep 17	ODE integration methods
Oct 1	The multigrid method
Oct 15	Multi-threaded programming
Nov 19	Methods for fluid simulations

All activities will be held on Wednesdays from 2:00pm to 4:00pm CT in 901 Van Vleck Hall. Activities will be recorded, but in-person participation is strongly encouraged—these sessions provide a useful venue for meeting classmates.

The optional activities will be worth 5% of the grade. Students can attend any number, although one will be required for full credit. Students will be expected to write a short report. The writeups will be graded on participation only, although the teaching staff will reject a submission if incorrect or insufficient. There is no extension to the submission deadlines.

There will be three tries to submit an activity writeup. The deadline for the first submission will be set to 5pm CT twelve days after the activity. If the deadline is missed, this will count as an incomplete first submission. Second and third submissions can submitted up until the final project deadline date, although students are encouraged to submit subsequent tries promptly.

If a student completes $n > 1$ activities, a bonus credit of $2(n-1)\%$ will be awarded, and the final project grade percentage will be reduced by $2(n-1)\%$.

Final project

The course will involve a final project, to be completed in groups of two or three students. Projects with $n \geq 4$ students will be allowed only with permission of the instructor and a statement about how the work will be divided.

- Each group will propose a project topic drawn from an application area of interest. The project should make use of concepts covered in the course.
- The project should be roughly equivalent in scope to a section of a research article.
- You will be required to write software to solve your problem, to submit a report that includes a mathematical discussion of your methodology in relation to the theory covered in the course, and to present your final project in the style of a conference talk (10 minutes presentation plus 5 minutes of Q&A).
- Projects will be assessed based on a written report (20%), the quality and correctness of software (12%), and the final presentation (8%). Code should be well-documented and should be organized so that figures submitted in the report can be easily reproduced.

The project will also involve a proposal, which must be completed by 5pm CT on November 7. This involves scheduling a 30 min meeting with Yue Sun to discuss your paper idea. Nothing written is required for this meeting, but you may find it helpful to bring documents to facilitate the discussion.

Grading

The final score will be based on homework assignments (55%), optional activities (5%), and the final project (40%). The scores will not be curved. The boundaries for letter grades are not fixed, and will be set by the instructor at the end of the course. You are strongly encouraged to attend the course lectures, but attendance/participation is not part of the course grade.

Course schedule

The final column contains notes about how the homework deadlines align with the course material. Each homework assignment will be due at 5pm CT the day after the corresponding lecture (usually on Fridays). It also contains the optional activities dates (usually on Wednesdays) and final project proposal meetings period. Final project presentation dates are tentative and subject to change.

Lecture	Date	Topics	Homework	OA/FP
1	Sep 4	Introduction; finite-difference approximations		
2	Sep 9	BVPs: boundary value problems, Dirichlet BCs, error, stability		
3	Sep 11	BVPs: consistency, convergence, Neumann BCs, nonlinear BVPs		OA1
4	Sep 16	Elliptic PDEs: 2D finite-difference stencils		
5	Sep 18	Elliptic PDEs: accuracy, stability		OA2
6	Sep 23	Iterative methods: Jacobi, Gauss–Seidel, descent methods		
7	Sep 25	Iterative methods: conjugate gradient, preconditioning	HW1	
8	Sep 30	Initial value problems: introduction, multistep methods		
9	Oct 2	Initial value problems: convergence, zero-stability		OA3
10	Oct 7	Initial value problems: absolute stability, stiffness		
11	Oct 9	Parabolic PDEs: explicit and implicit methods, method of lines	HW2	
12	Oct 14	Parabolic PDEs: stability, convergence, multidimensional problems		
13	Oct 16	Hyperbolic PDEs: advection equation, characteristics, upwinding		OA4
14	Oct 21	Hyperbolic PDEs: von Neumann stability analysis, CFL condition		
15	Oct 23	Hyperbolic PDEs: second-order methods	HW3	
16	Oct 28	FVM : finite volume methods, flux, numerical conservation		
17	Oct 30	FVM: Riemann problem, Godunov and Lax–Wendroff method		FPPM
18	Nov 4	FVM: high resolution methods, limiters		
19	Nov 6	FVM: nonlinear equations, shocks, ENO methods		FPPM
20	Nov 11	Mixed equations: fractional step methods		
21	Nov 13	Mixed equations: incompressible Navier–Stokes	HW4	
22	Nov 18	Spectral methods: introduction		
23	Nov 20	Spectral methods: smoothness, spectral accuracy		OA5
–	Nov 25	*No lecture*		
–	Nov 27	*Thanksgiving*		
–	Dec 2	*No lecture*		
24	Dec 4	Spectral methods: Chebyshev spectral methods	HW5	
–	Dec 9	*Final project presentations*		
–	Dec 10	*Final project presentations*		
–	Dec 11	*Final project presentations*		

Since the course will also feature the optional activity sessions, three lectures at the end of course will be omitted, although this may be adjusted depending on the pace of the course. This will also provide students more time to focus on the final project.

Academic Integrity Policy

By virtue of enrollment, each student agrees to uphold the high academic standards of the University of Wisconsin–Madison; academic misconduct is behavior that negatively impacts the integrity of the institution. Cheating, fabrication, plagiarism, unauthorized collaboration, and helping others commit these previously listed acts are examples of misconduct which may result in disciplinary action. Examples of disciplinary sanctions include, but are not limited to, failure on the assignment/course, written reprimand, disciplinary probation, suspension, or expulsion.

AI statement

While generative artificial intelligence (AI) is powerful and rapidly evolving, it is also prone to producing incorrect code, fabricated references, and misleading summaries. To become critical and responsible users of modern tools, students are expected to learn independently—understanding not just what works, but why it works—so that you can adapt and apply these tools effectively and accurately. Responsible use of such tools requires, **first and foremost**, a strong foundation in the subject matter.

The use of generative AI tools and applications (including, but not limited to ChatGPT, Copilot, Claude, etc.) in this course is governed by the following policies:

- **Prohibited uses:** All submitted assignments for this course must be a student's own work. The use of generative AI to draft or produce writeups, solutions, proofs, or code is prohibited. Such use constitutes a violation of UW–Madison's academic misconduct policy, specifically *UWS 14.03(1)b (b) Uses unauthorized materials or fabricated data in any academic exercise.*
- **Permitted uses:** Limited and strategic use of AI is allowed in the following cases:
  - **Copyediting:** AI may be used to check grammar or to assist with formatting.
  - **Plotting:** AI may be used to assist with plotting commands that are not demonstrated in class. However, students must verify and be able to explain any submitted plotting code.
  - **Literature review:** AI may be used as a "*search engine*" to help identify potential references. You must independently verify all findings with cited traditional and verifiable sources. AI-generated summaries cannot substitute for reading the original papers, as they are often skewed by the prompt and may provide incorrect information.
- **Disclosure requirement:** If AI is used in any of the permitted ways, students must include a footnote in the writeup stating how AI was used, and clearly note the usage in comments within your code. Students should be prepared to provide the exact prompt(s) used if requested by the teaching staff. Failure to disclose AI use will be treated as academic misconduct.

Diversity & inclusion statement

Diversity is a source of strength, creativity, and innovation for UW–Madison. We value the contributions of each person and respect the profound ways their identity, culture, background, experience, status, abilities, and opinion enrich the university community. We commit ourselves to the pursuit of excellence in teaching, research, outreach, and diversity as inextricably linked goals. The University of Wisconsin–Madison fulfills its public mission by creating a welcoming and inclusive community for people from every background—people who as students, faculty, and staff serve Wisconsin and the world.

Accommodations for students with disabilities

The University of Wisconsin–Madison supports the right of all enrolled students to a full and equal educational opportunity. The Americans with Disabilities Act (ADA), Wisconsin State Statute (36.12), and UW–Madison policy (UW-855) require the university to provide reasonable accommodations to students with disabilities to access and participate in its academic programs and educational services. Faculty and students share responsibility in the accommodation process. Students are expected to inform faculty of their need for instructional accommodations during the beginning of the semester, or as soon as possible after being approved for accommodations. Faculty will work either directly with the student or in coordination with the McBurney Center to provide reasonable instructional and course-related accommodations. Disability information, including instructional accommodations as part of a student's educational record, is confidential and protected under FERPA. (See: McBurney Disability Resource Center).

Additional UW–Madison academic policies and statements

- Academic calendar and religious observances
- Course evaluations and digital course evaluations
- Mental health and well-being statements
- Privacy of student records and the use of audio recorded lectures statement
- Students' rules, rights and responsibilities
- Teaching and learning data transparency statement
- Campus resources for academic success

(Version 2, last edit: September 1, 2025)