COMP 350 Numerical Computing

Ivo Panayotov
Sitao Luan

September 2, 2019

Course Outline

- **Objectives:**

  To provide an introduction to numerical techniques for solving basic computational science problems.

  It emphasizes the design, analysis, and computer implementation of accurate and efficient algorithms.

- **Prerequisites:**

  A high-level scientific programming language (C, C++, Java etc),

  Calculus III, Linear algebra

Course Outline

Topics:

- *Computer numbers and arithmetic.*
- *MATLAB.*
- *Solution of systems of equations.*
- *Root finding.*
- *Interpolation and approximation.*
- *Numerical integration.*
- *Numerical solutions to ordinary differential equations.*

Course Outline

Texts:

- **Ward Cheney & David Kincaid:**
  Numerical Mathematics and Computing, 7th ed,
  CENGAGE Learning, 2013.
  This course will cover (tentatively):
  1.1-1.4, 2.1-2.3, 3.1-3.3, 4.1-4.3, 5.1, 5.3, 5.4, 6.1-6.2, 7.1-7.2 and 9.1.
- **Michael Overton:**
  Numerical Computing with IEEE Floating Point Arithmetic, SIAM, 2004.
  Floating Point Representation and the IEEE Standard, 1997, available on the course web site.

Course Outline

References:

- **Uri Ascher and Chen Greif**:
  A First Course in Numerical Methods, SIAM, 2011.
- **Walter Gander, Martin J. Gander, and Felix Kwok**:
  Scientific Computing - An Introduction using Maple and MATLAB, Springer, 2014.
- **Cleve Moler:**
  Numerical Computing with MATLAB, SIAM, 2004, available on the author's web site.

Lecture notes:

- Available from McGill's myCourses.

Course Outline

Evaluation:

- 6 assignments, available from myCourses, 20%.
- Midterm, Tuesday, Oct. 9 (tentatively), 20%.
- Final, 60%.

Both exams will be **closed book tests**.

There will be a **supplemental exam**, 100%.

There will be no option of doing additional work to upgrade the mark.

Course Outline

Policy on Grading:

- No late homeworks will be accepted without an acceptable excuse.
- Any regrade requests must be requested within 10 working days of the day in which the item is made available to you; after 10 working days have elapsed, regrade requests will not be accepted.

Course Outline

Right to submit in English or French written work:

In accord with McGill University's Charter of Students' Rights, students in this course have the right to submit in English or in French any written work that is to be graded.

Academic Integrity:

McGill University values academic integrity. Therefore, all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Conduct and Disciplinary Procedures.

Course Outline

Contact Information & Office Hours:

- **Instructor Part 1:** Ivo Panayotov
  Email: ivo.panayotov@mcgill.ca
  Office Hours: Tuesday, Thursday, McConnell Eng 204, 19:00–20:00
- **Instructor Part 2:** Sitao Luan
  Email: sitao.luan@mail.mcgill.ca
  Office Hours: Tuesday, Thursday, McConnell Eng 204, 19:00–20:00
- **Teaching Assistants:**
  (For a question about the grade of a specific assignment, please contact only the TA who marks it.)

Course Outline

Jianhao Cao: jianhao.cao@mail.mcgill.ca;
Office Hours: Thursday 9:30am to 10:30pm
Zhilong Chen: zhilong.chen@mail.mcgill.ca;
Office Hours: Thursday 3:00pm to 4:00pm
Runzhou Fan: runzhou.fan@mcgill.ca;
Office Hours: Monday 3:00pm to 4:00pm
Mingde Zhao: mingde.zhao@mail.mcgill.ca;
Office Hours: Wednesday 3:30pm to 4:30pm
Jenny Long: xiong.long@mail.mcgill.ca;
Office Hours: Friday 11:00am to 12:00pm
Office: Trottier Building 3090

Introduction

Numerical computing means

computing with numbers

Introduction – History

- It is almost as old as civilization itself.
- **Modern numerical computing** began with **Isaac Newton** in the 17th century; his invention of calculus was driven by its use in solving **numerical** problems.
- Until the 20th century, calculation was primarily done with **pencil and paper** in the west and the **abacus** in the east.
- In the first half of the 20th century the **slide rule** made multiplication easy, but gave only **3 digits of accuracy**. **Mechanical calculators** were more costly and cumbersome, but accurate.
- The invention of **electronic computer** brought a new era for numerical computing.

Introduction – History

- It is almost as old as civilization itself.
- **Modern numerical computing** began with **Isaac Newton** in the 17th century; his invention of calculus was driven by its use in solving **numerical** problems.
- Until the 20th century, calculation was primarily done with **pencil and paper** in the west and the **abacus** in the east.
- In the first half of the 20th century the **slide rule** made multiplication easy, but gave only **3 digits of accuracy**. **Mechanical calculators** were more costly and cumbersome, but accurate.
- The invention of **electronic computer** brought a new era for numerical computing.

Introduction – History

- It is almost as old as civilization itself.
- **Modern numerical computing** began with **Isaac Newton** in the 17th century; his invention of calculus was driven by its use in solving **numerical** problems.
- Until the 20th century, calculation was primarily done with **pencil and paper** in the west and the **abacus** in the east.
- In the first half of the 20th century the **slide rule** made multiplication easy, but gave only **3 digits of accuracy**. **Mechanical calculators** were more costly and cumbersome, but accurate.
- The invention of **electronic computer** brought a new era for numerical computing.

Introduction – History

- It is almost as old as civilization itself.
- **Modern numerical computing** began with **Isaac Newton** in the 17th century; his invention of calculus was driven by its use in solving **numerical** problems.
- Until the 20th century, calculation was primarily done with **pencil and paper** in the west and the **abacus** in the east.
- In the first half of the 20th century the **slide rule** made multiplication easy, but gave only **3 digits of accuracy**. **Mechanical calculators** were more costly and cumbersome, but accurate.
- The invention of **electronic computer** brought a new era for numerical computing.

Introduction – History

- It is almost as old as civilization itself.
- **Modern numerical computing** began with **Isaac Newton** in the 17th century; his invention of calculus was driven by its use in solving **numerical** problems.
- Until the 20th century, calculation was primarily done with **pencil and paper** in the west and the **abacus** in the east.
- In the first half of the 20th century the **slide rule** made multiplication easy, but gave only **3 digits of accuracy**. **Mechanical calculators** were more costly and cumbersome, but accurate.
- The invention of **electronic computer** brought a new era for numerical computing.

Introduction – Computers

- **Computers** were first invented in the 1940's and 1950's for solving **hard scientific and engineering problems** which required a great deal of **numerical computing**.
- During the 1950's, the primary usage of computers was for **numerical computing in scientific applications**.
- In the 1960's, computers became widely used by large businesses, for **processing all kinds of information**.
- Computers became far more widespread, to medium-sized businesses in the 1970's, and to many millions of small businesses and individuals during the **PC revolution** of the 1980's and 1990's.
  The main interest is **processing of information**:
  text, image, sound.

Introduction – Computers

- **Computers** were first invented in the 1940's and 1950's for solving **hard scientific and engineering problems** which required a great deal of **numerical computing**.
- During the 1950's, the primary usage of computers was for **numerical computing in scientific applications**.
- In the 1960's, computers became widely used by large businesses, for **processing all kinds of information**.
- Computers became far more widespread, to medium-sized businesses in the 1970's, and to many millions of small businesses and individuals during the **PC revolution** of the 1980's and 1990's.
  The main interest is **processing of information**:
  text, image, sound.

Introduction – Computers

- **Computers** were first invented in the 1940's and 1950's for solving **hard scientific and engineering problems** which required a great deal of **numerical computing**.
- During the 1950's, the primary usage of computers was for **numerical computing in scientific applications**.
- In the 1960's, computers became widely used by large businesses, for **processing all kinds of information**.
- Computers became far more widespread, to medium-sized businesses in the 1970's, and to many millions of small businesses and individuals during the **PC revolution** of the 1980's and 1990's.
  The main interest is **processing of information**:
  text, image, sound.

Introduction – Computers

- **Computers** were first invented in the 1940's and 1950's for solving **hard scientific and engineering problems** which required a great deal of **numerical computing**.
- During the 1950's, the primary usage of computers was for **numerical computing in scientific applications**.
- In the 1960's, computers became widely used by large businesses, for **processing all kinds of information**.
- Computers became far more widespread, to medium-sized businesses in the 1970's, and to many millions of small businesses and individuals during the **PC revolution** of the 1980's and 1990's.

  The main interest is **processing of information**:
  text, image, sound.

Introduction – Computers

- **Computers** were first invented in the 1940's and 1950's for solving **hard scientific and engineering problems** which required a great deal of **numerical computing**.
- During the 1950's, the primary usage of computers was for **numerical computing in scientific applications**.
- In the 1960's, computers became widely used by large businesses, for **processing all kinds of information**.
- Computers became far more widespread, to medium-sized businesses in the 1970's, and to many millions of small businesses and individuals during the **PC revolution** of the 1980's and 1990's.
  The main interest is **processing of information**:
  text, image, sound.

Introduction – Uses of numerical computing

In **most scientific disciplines**, computing with **numbers** remains by far the most important use of computers. e.g.

- **Physicists:** solve complicated eqns, from modeling the expansion of the universe, to the microstructure of the atom
- **Chemists and biologists:** determine the molecular structure of proteins.
- **Medical researchers:** design new medical techniques.
- **Atmospheric scientists:** process huge quantities of data and solve appropriate equations to predict the weather.
- **Aeronautical engineers:** designing better airplanes. Boeing 757, 767 & 737-800 were designed relying more heavily on **computer modeling** than on wind tunnel tests.

Introduction – Uses of numerical computing

In **most scientific disciplines**, computing with **numbers** remains by far the most important use of computers. e.g.

- **Physicists:** solve complicated eqns, from modeling the expansion of the universe, to the microstructure of the atom
- **Chemists and biologists:** determine the molecular structure of proteins.
- **Medical researchers:** design new medical techniques.
- **Atmospheric scientists:** process huge quantities of data and solve appropriate equations to predict the weather.
- **Aeronautical engineers:** designing better airplanes. Boeing 757, 767 & 737-800 were designed relying more heavily on **computer modeling** than on wind tunnel tests.

Introduction – Uses of numerical computing

In **most scientific disciplines**, computing with **numbers** remains by far the most important use of computers. e.g.

- **Physicists:** solve complicated eqns, from modeling the expansion of the universe, to the microstructure of the atom
- **Chemists and biologists:** determine the molecular structure of proteins.
- **Medical researchers:** design new medical techniques.
- **Atmospheric scientists:** process huge quantities of data and solve appropriate equations to predict the weather.
- **Aeronautical engineers:** designing better airplanes. Boeing 757, 767 & 737-800 were designed relying more heavily on **computer modeling** than on wind tunnel tests.

Introduction – Uses of numerical computing

In **most scientific disciplines**, computing with **numbers** remains by far the most important use of computers. e.g.

- **Physicists:** solve complicated eqns, from modeling the expansion of the universe, to the microstructure of the atom
- **Chemists and biologists:** determine the molecular structure of proteins.
- **Medical researchers:** design new medical techniques.
- **Atmospheric scientists:** process huge quantities of data and solve appropriate equations to predict the weather.
- **Aeronautical engineers:** designing better airplanes. Boeing 757, 767 & 737-800 were designed relying more heavily on **computer modeling** than on wind tunnel tests.

Introduction – Uses of numerical computing

In **most scientific disciplines**, computing with **numbers** remains by far the most important use of computers. e.g.

- **Physicists:** solve complicated eqns, from modeling the expansion of the universe, to the microstructure of the atom
- **Chemists and biologists:** determine the molecular structure of proteins.
- **Medical researchers:** design new medical techniques.
- **Atmospheric scientists:** process huge quantities of data and solve appropriate equations to predict the weather.
- **Aeronautical engineers:** designing better airplanes. Boeing 757, 767 & 737-800 were designed relying more heavily on **computer modeling** than on wind tunnel tests.

Introduction – Comp. Sci. & Numer. Comp.

- **NC** is important in computer science:
  Bioinformatics, data mining and information retrieval, animation, graphics, image processing, machine learning, modeling and simulation, patten recognition, robotics and computer vision, etc.
- **NC** is an important part of that technology which any computer scientist should understand at some level.

Introduction – Comp. Sci. & Numer. Comp.

- **NC** is important in computer science:
  Bioinformatics, data mining and information retrieval, animation, graphics, image processing, machine learning, modeling and simulation, patten recognition, robotics and computer vision, etc.
- **NC** is an important part of that technology which any computer scientist should understand at some level.

- **Ariane 5 Explosion (1996)**:

  On June 4, 1996, the Ariane 5 rocket launched by the European Space Agency exploded forty seconds after lift-off from Kourou, French Guiana.
  The rocket and its cargo valued at $500 million.

  Cause: failed conversion of a 64 bit floating point number to a 16 bit signed integer.

- **The Patriot Missile Failure (1991)**:

  On February 25, 1991, during the Gulf War, an American Patriot Missile battery in Dhahran, Saudi Arabia, failed to track and intercept an incoming Iraqi Scud missile. The Scud struck an American Army barracks, killing 28 soldiers and injuring around 100 other people.

  Cause: an inaccurate calculation of the time due to computer arithmetic errors.

- **Vancouver Stock Exchange Index (1983)**:

  January 1982: Index established at 1000.
  November 1983: Index was 520.
  But exchange seemed to be doing well.

  Cause: Index rounded down to three digits at each recomputation, e.g., $678.35 \rightarrow 678$.
  Errors always in same direction. Thousands of small errors add up to a large error.

  A correct recalculation gave a value of 1098.892.

Solving a Quadratic Equation

The quadratic equation $ax^2 + bx + c = 0$ has two solutions:

$$x_{12} = (-b \pm \sqrt{b^2 - 4ac})/(2a)$$

Use the formula to solve $x^2 - 10,1000x + 1 = 0$.
True solutions

$$x_1 = 9999.999899999998, \quad x_2 = 0.000100000001000$$

In 8-digit arithmetic, the formula gives

$$x_1 = 10,000.0, \text{ very good}, \quad x_2 = 0, \text{ completely wrong}$$

Better: compute x_1 from the formula and x_2 from

$$x_1 x_2 = c/a$$

This gives an accurate x_2.

Introduction – Programming and Code Advice

- Be careful and be correct
- Use pseudocode
- Check and double check
- Use test case
- Modularize code
- Include warning messages
- Use meaningful variable names
- Include comments
- Use appropriate data structure
- Use built-in functions and program libraries
- Do not over-optimize