Lecture TF 502 on

Numerical Analysis

Introduction & Organisation

Contents

- Objectives
- Organisation
- Exercises
- Software
- Literature
- Grading

Objectives

This course is the basis for all scientific computing based research as needed in, e.g.,

- Process Control and Optimization, Robotics
- Image Processing, Computer Vision
- Wireless Networks; Distributed Optimization
- Computational Methods in Physics and Biology

We will learn how about algorithms that allow us solve a great variety of numerical computation and optimization problems arising from these and other application areas.

Objectives

By the end of this lecture, you should be able to

- solve least-squares regression and fitting problems
- solve linear and nonlinear equations using a computer
- formulate and solve simple optimization problems
- implement simple algorithms in at least one programming language
- solve a variety of numerical computation and optimization problems from application areas.

Objectives

By the end of this lecture, you should be able to

- solve least-squares regression and fitting problems
- solve linear and nonlinear equations using a computer
- formulate and solve simple optimization problems
- implement simple algorithms in at least one programming language
- solve a variety of numerical computation and optimization problems from application areas.

Objectives

By the end of this lecture, you should be able to

- solve least-squares regression and fitting problems
- solve linear and nonlinear equations using a computer
- formulate and solve simple optimization problems
- implement simple algorithms in at least one programming language
- solve a variety of numerical computation and optimization problems from application areas.

Objectives

By the end of this lecture, you should be able to

- solve least-squares regression and fitting problems
- solve linear and nonlinear equations using a computer
- formulate and solve simple optimization problems
- implement simple algorithms in at least one programming language
- solve a variety of numerical computation and optimization problems from application areas.

Objectives

By the end of this lecture, you should be able to

- solve least-squares regression and fitting problems
- solve linear and nonlinear equations using a computer
- formulate and solve simple optimization problems
- implement simple algorithms in at least one programming language
- solve a variety of numerical computation and optimization problems from application areas.

Contents

- Objectives
- Organisation
- Exercises
- Software
- Literature
- Grading

Organisation

Mondays 10:15-11:45, Wednesdays 10:15-11:45

Exceptions will be announced in the lecture.

All information about the lecture, news, announcements, exercises, tutorials, program code examples, animations, slides, etc. at:

`http://sist.shanghaitech.edu.cn/faculty/boris/`

(click on “Courses”)

Organisation

Mondays 10:15-11:45, Wednesdays 10:15-11:45

Exceptions will be announced in the lecture.

All information about the lecture, news, announcements, exercises, tutorials, program code examples, animations, slides, etc. at:

`http://sist.shanghaitech.edu.cn/faculty/boris/`

(click on “Courses”)

Contents

- Objectives
- Organisation
- Exercises
- Software
- Literature
- Grading

Exercises

- we will handout several exercise sheets (=homework) during the semester;
- solutions have to be handed in before the deadline mentioned on the exercise sheets (usually one week, but for some programming projects you'll have more time).
- submit via e-mail to `zhayl@shanghaitech.edu.cn` (as pdf file, subject of the e-mail "YOUR_NAME_EXERCISE_XX"), or bring a printed copy to the lecture.
- **VERY IMPORTANT: you can discuss the homework with others, but you'll have to write it down yourself.**

Exercises

- we will handout several exercise sheets (=homework) during the semester;
- solutions have to be handed in before the deadline mentioned on the exercise sheets (usually one week, but for some programming projects you'll have more time).
- submit via e-mail to `zhayl@shanghaitech.edu.cn`
  (as pdf file, subject of the e-mail "YOUR_NAME_EXERCISE_XX"),
  or bring a printed copy to the lecture.
- **VERY IMPORTANT: you can discuss the homework with others, but you'll have to write it down yourself.**

Exercises

- we will handout several exercise sheets (=homework) during the semester;
- solutions have to be handed in before the deadline mentioned on the exercise sheets (usually one week, but for some programming projects you'll have more time).
- submit via e-mail to `zhayl@shanghaitech.edu.cn` (as pdf file, subject of the e-mail "YOUR_NAME_EXERCISE_XX"), or bring a printed copy to the lecture.
- **VERY IMPORTANT: you can discuss the homework with others, but you'll have to write it down yourself.**

Contents

- Objectives
- Organisation
- Exercises
- Software
- Literature
- Grading

Software

This course will be entirely based on open-source software.

- In the lectures we will use the programming language "JULIA".
- JULIA is currently developed and used in courses at MIT.
- Used in many courses, e.g., in Stanford—and here.

- JULIA is recommended for exercises and we will provide tutorials,
- but in the exercises it's allowed to use Matlab or C++ as an alternative.

Software

This course will be entirely based on open-source software.

- In the lectures we will use the programming language “JULIA”.
- JULIA is currently developed and used in courses at MIT.
- Used in many courses, e.g., in Stanford—and here.

- JULIA is recommended for exercises and we will provide tutorials,
- but in the exercises it’s allowed to use Matlab or C++ as an alternative.

Software

This course will be entirely based on open-source software.

- In the lectures we will use the programming language “JULIA”.
- JULIA is currently developed and used in courses at MIT.
- Used in many courses, e.g., in Stanford—and here.
- JULIA is recommended for exercises and we will provide tutorials,
- but in the exercises it’s allowed to use Matlab or C++ as an alternative.

Software

This course will be entirely based on open-source software.

- In the lectures we will use the programming language “JULIA”.
- JULIA is currently developed and used in courses at MIT.
- Used in many courses, e.g., in Stanford—and here.

- JULIA is recommended for exercises and we will provide tutorials,
- but in the exercises it’s allowed to use Matlab or C++ as an alternative.

Contents

- Objectives
- Organisation
- Exercises
- Software
- Literature
- Grading

Literature

- Search for a book in a style and language that you like reading!
- The course is loosely based on:

1. R.L. Burden and J.D. Faires.

   Numerical Analysis. BrooksCole, Cengage Learning.
2. J. Stoer and R. Burlisch.

   Introduction to numerical analysis, 3. Edition, Springer 2002.

- More material (slides, scripts, JULIA code examples, etc.) will be provided on the webpage.

Literature

- Search for a book in a style and language that you like reading!
- The course is loosely based on:

1. R.L. Burden and J.D. Faires.

   Numerical Analysis. BrooksCole, Cengage Learning.
2. J. Stoer and R. Burlisch.

   Introduction to numerical analysis, 3. Edition, Springer 2002.

- More material (slides, scripts, JULIA code examples, etc.) will be provided on the webpage.

Literature

- Search for a book in a style and language that you like reading!
- The course is loosely based on:

1. R.L. Burden and J.D. Faires.

   Numerical Analysis. BrooksCole, Cengage Learning.
2. J. Stoer and R. Burlisch.

   Introduction to numerical analysis, 3. Edition, Springer 2002.

- More material (slides, scripts, JULIA code examples, etc.) will be provided on the webpage.

Contents

- Objectives
- Organisation
- Exercises
- Software
- Literature
- Grading

Grading

- Homework: 20%
- Mid-Term Exam: 20%
- Project Presentation: 20%
- Project Report: 20%
- Final Exam: 20%