

Numerical Methods and Software

Introduction

Uwe Naumann



Informatik 12: Software and Tools for Computational Engineering (STCE)

RWTH Aachen University

◆□▶ ◆□▶ ◆三▶ ◆三▶ ● つくぐ



2

Cover Story

Motivation

Introduction



3

Cover Story

Motivation

Introduction

Cover Story Modern Family





© my wife and me

Cover Story Gender Equality





©dex1.info: Dinge, die Männer an Frauen unattraktiv finden

Cover Story Indeed ...





Verblüffender Effekt Wer Bier trinkt, bricht seltener das Studium ab

Wissenschaftler haben einen Zusammenhang zwischen dem Genuss von Alkohol und einem erfolgreichen Studienabschluss gefunden. Doch die Promille sind gar nicht entscheidend.



Männer mit Bierkasten

© Spiegel Online



7

Cover Story

Motivation

Introduction





Motivation Numerical Simulation





Numerical Methods and Software, info@stce.rwth-aachen.de

70 80



accuracy

efficiency

sustainability

... of numerical simulations.

This course provides basic understanding of selected numerical methods including corresponding software support.



Note: We expect the derivative (as well as it finite difference approximation) to be equal to one.

Remark: My code examples are written in simple C++. If you do not "speak" C++, then you will pick some of it up during this course as a nice side effect :-)



```
#include "Eigen/Dense"
1
    #include <cassert>
2
    #include <iostream>
3
4
    int main(int c, char* v[]) {
5
      assert(c==2);
6
      int n=std::stoi(v[1]);
7
      using T = float;
8
      // using T=double;
9
      typedef Eigen::Matrix<T,Eigen::Dynamic,Eigen::Dynamic> MT;
10
      typedef Eigen::Matrix<T,Eigen::Dynamic,1>VT;
11
      MT A=MT::Random(n,n), B=MT::Random(n,n);
12
      VT C=VT::Random(n);
13
      MT D=A*B*C:
14
      // MT D=A*(B*C);
15
      std::cout.precision(7);
16
      // std::cout.precision(15);
17
      std::cout << D << std::endl:
18
      return 0:
19
20
```





Use your / aquire software engineering skills!



Cover Story

Motivation

Introduction

Software and Totis for Computational Ingineering

Reasons for not just using pen and paper ...

- mathematically unknown solution needs to be approximated [iteratively]
- mathematically known solution needs to be evaluated repeatedly over a short time period

Ability of computers to perform a very large number of elemental algorithmic steps within a short time is exploited.

Efficient, robust, flexible and sustainable numerical solutions require good knowledge of computer infrastructure (programming language, compiler, libraries, hardware and run time support)

Software and Totis tor Computational Inginteering

- introduction to numerical computing for undergraduate CS students
- essential prerequistes for further specialization in diverse subareas of CS (machine learning, speech recognition, computer graphics and vision, high-performance scientific computing ...)
- focus on numerical algorithms and software based on essential analysis and linear algebra (no formal proofs).
- covering
 - ▶ 1D case \Rightarrow visualization
 - *n*D case illustrated by lower-D examples \Rightarrow visualization

Introduction Contents

Software and Tools for Computational Engineering

Story in 1D ...

- essential calculus and linear algebra
- error analysis and computer arithmetic
- numerical / algorithmic differentiation
- ▶ root finding (linearization → Newton's method)
- unconstrained nonlinear convex optimization (steepest descent, Newton's method)
- linear regression (normal equation, Givens rotation, Householder reflection)
- nonlinear regression (linearization + linear regression)

 \dots and generalization to nD case.



A-Level Math

Analysis

Linear Algebra

Programming

Software and Tools for Computational Engineering

Research, development and teaching @ STCE is driven by

- algorithmic differentiation
- numerical methods
- combinatorial scientific computing
- program analysis and transformation
- high-performance computing
- simulation software engineering

See www.stce.rwth-aachen.de for more.

Introduction

Admin

Software and Tools tor Computational Engineering

- main lecture (one per week throughout the semester) covers general story including
 - theory
 - algorithms
 - sample prototype implementation
- software lecture /lab (one per week during second half of the semester; t.b.a.) covers numerical software including
 - Eigen
 - GNU Scientific Library
 - NAG Library
- tutorial (one per week throughout the semester starting in week three) supports main lecture by discussing both pen&paper as well as programming exercises
- learning material
 - videos, slides, code, reference solutions for tutorial exercises, pointers into literature



- ... to complement material presented during lectures/tutorials:
 - ► Heath: Scientific Computing, Second Edition, McGraw-Hill, 1997.
 - Dahmen, Reusken: Numerik f
 ür Ingenieure und Naturwissenschaftler, 1. Auflage, Springer, 2008.
 - Knorrenschildt: Numerische Mathematik, Carl Hanser Verlag, 6. Auflage, 2017.



Cover Story

Motivation

Introduction



- Numerical simulation is an essential element of mankind's toolbox for understanding the world.
- Computer scientists should to able to use numerical software to support, for example, machine learning, computer graphics, high-performance scientific computing.
- Computer scientists should drive the development of numerical software to ensure sustainability ⇒ in-depth knowledge is required.
- You should attend this course ... ;-)