Course Manual PRA1004: Scientific Computing

Period 5

Contents

1 Introduction

Mathematical models form the core of many scientific studies across all academic disciplines. Especially in exact sciences, it is difficult to overstate the role of such models. However, the complexity of such models directly implies the need of computers in order to estimate their parameters and/or use them for prediction. Scientific computation deals with problems that frequently arise when implementing mathematical models on a computer, paying particular attention to issues that arise by making the transition from mathematics to the finite representations used by computers.

This skills provides an introduction to programming in high-level scientific programming languages like Matlab, Mathematica and Simulink. It presents a broad overview of numerical methods for solving selected problems in scientific computing, including linear and nonlinear equations, interpolation, differential equations, integration, optimization, random numbers and simulation.

2 General Information

2.1 Course Coordinator

Name:	Frans Oliehoek
Department:	DKE (RAI group)
Location:	SSK 39, room 2.001
Tel.:	+31 43 3883485
Email:	frans.oliehoek@maastrichtuniversity.nl
WWW:	http://people.csail.mit.edu/fao/

2.2 Schedule and Location

Course days will start with college. The rest of the day will be a lab course. A detailed schedule is listed in Section 4.

2.3 Text Book & Course Materials

The course will make use of a number of sources, such as on-line tutorials and handouts with assignments and slides. In addition, it makes use of the following text book:

- *Scientific Computing with MATLAB and Octave.*
  Series: Texts in Computational Science and Engineering, Vol. 2
  Alfio Quarteroni, Fausto Saleri & Paola Gervasio
  3rd edition.

We will refer to the text book as QSG.

Further Recommendations

For students without any programming experience that would like to have a book containing the Matlab basics, the following book is recommended:

- *Introduction to MATLAB.* Delores M. Etter. 2nd ed.

  (Note that most of the information in this book is also available from the help functionality within Matlab. Still, the book gives a nice an clear overview and includes a number of nice examples.)

Many of the method of scientific computing build upon linear algebra. While QSG probably covers all the linear algebra needed for this course, I highly recommend everybody to own a text book on linear algebra. The linear algebra course offered by the Maastricht Science Program uses the following book:

- *Linear Algebra and Its Applications.* David C. Lay. 4th ed.

For students interested in further exploring numerical methods and numerical analysis, I highly recommend the following book for further reading:

- *Numerical Methods. An introduction to Scientific Computing Using MATLAB.* Peter Linz, Richard L.C. Wang.

3 Objectives

This course serves two goals:

1. To make students familiar with the concepts of programming and the get them accustomed with high-level languages like Matlab en Mathematica.
2. To provide an overview of some of the issues and problems that arise in scientific computation, such as (non-)linear systems, numerical and symbolic integration, differential equations and simulation.

After this course the student should be able to analyze scientific problems in a structured way leading to the design and implementation of an appropriate solution in an adequate scientific programming language (such as matlab, mathematica, simulink).

4 Schedule

The proposed schedule is as follows (and subject to changes):

Week	Topic
1	Introduction Matlab & programming. Non-linear equations (Newton's method).
2	Floating point numbers. Matrix computations and linear systems, linear regression.
3	Clustering, compression and reconstruction (k-means clustering, PCA).
4	Introduction Mathematica & symbolic integration. Numerical integration methods.
5	Deterministic simulation. Introduction Simulink.
6	Stochastic simulation.

5 Assessment

Assessment will take place via grading of lab reports. Each lab session requires the students to write a short report to accompany their code. The report and code should be handed in 2 working days after each session and will be graded. Also, in the final lab session there will be a written test. The final mark will be computed as the mean of these grades.