



# Numerical Methods in Physics

*Numerische Methoden in der Physik, 515.421.*

**Instructor:** Ass. Prof. Dr. Lilia Boeri  
Room: PH 03 090  
Tel: +43-316-873 8191  
Email Address: [l.boeri@tugraz.at](mailto:l.boeri@tugraz.at)

**Room:** TDK Seminarraum

**Time:** 8:30-10 a.m.

**Exercises:** Computer Room, PH EG 004 F

[http://itp.tugraz.at/LV/boeri/NUM\\_METH/index.html](http://itp.tugraz.at/LV/boeri/NUM_METH/index.html)

# A little bit about myself (1):



Lilia Boeri, Ass. Prof.,  
Institute for Theoretical and Computational Physics,  
TU Graz. (since 1<sup>st</sup> April 2013).

**Before coming here...**

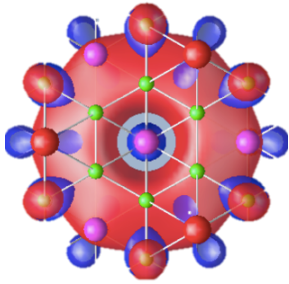


Universita' La Sapienza, Rome, (Italy);  
**Master and PhD in physics.**



Max-Planck Inst. for Solid State Research, Stuttgart (Germany):  
**Member of the scientific Staff and Max-Planck-Group Leader.**

# A little bit about myself (2):



## Scientific Interests:

Computational Superconductivity: Use *ab-initio* methods (numerical first-principles calculations) to understand and predict superconductivity.

If you want to know more:

<http://www2.fkf.mpg.de/boeri>

## AT TU I will be teaching:

Numerical Methods in Physics (WS 2013/2014).

Theoretical Solid State Physics 1. (WS 2013/2014).

Theoretical Solid State Physics 2. (SS 2014/2015).

# And my collaborators (1):

## Exercise Instructors:



**Cristoph Heil**, Dipl.-Ing.

Tel: 0316-873-8189

Email: [cheil@sbox.tugraz.at](mailto:cheil@sbox.tugraz.at)



**Gernot Kapper**, Dipl.-Ing.

Tel: 0316-873-8194

Email: [gernot.kapper@tugraz.at](mailto:gernot.kapper@tugraz.at)

# And my collaborators (2):

## Exercise Tutors:



**Gernot Krabberger:**

gkrabberger@student.tugraz.at



**Markus Krammer:**

markus.krammer@student.tugraz.at



**Peter Wriesnik:**

peter.wriesnik@student.tugraz.at

# The First Lecture:

## Practicalities:

- Structure of the course.
- Script
- Important dates.
- Exercise class (rules and practical matters).
- Point System.

## Introduction

*I will upload a pdf of this lecture on the web page of the course.*

# Structure of the Course:

**Main Lecture:** will illustrate the most important aspects of Numerical Methods in Physics, with practical examples.

➤ **Exam:** Oral Examination, at the end of the semester (Appointment).

**Exercise Class:** In the exercise class you will be required to solve practical physical problems, using the numerical methods presented in the main lecture. **Exercises must be programmed in C or Matlab.** The computers in the Computer room are equipped with the necessary software.

➤ **Exam:** Rules for the exercise classes are more complicated, I will explain them better today and repeat them next week at the exercise classes.

# Important Dates:

**First Lecture:** 1<sup>st</sup> October 2013.

**Last Lecture:** 21<sup>st</sup> January 2014.

**Exams:** The oral exam can be taken starting on 1<sup>st</sup> of February 2014 (by appointment).

**First Exercise Class:** 1<sup>st</sup> October 2013.

**Last Exercise Class (Last Abgabetermin):** 21<sup>st</sup> January 2014.

**Exams:** The last available date for the presentation of the exercises is the **10<sup>th</sup> of March 2014**. We will accept no exercises after this date.



# Exercise Class (1):

**Exercise Class:** In the exercise class you will be required to solve **four** practical physical problems, using the numerical methods presented in the main lecture.

**A few examples from the past years (prof. Sormann):**

- Fast-Fourier Transforms (Gibbs Equation).
- Initial Value Problems (Particle in Magnetic Fields).
- Eigenvalues: (band structure of Lithium).
- Cubic splines and shooting-Numerov (Helium Atom).

*The text of the first exercise (zeroes of real functions) is already available on the web page of the course.*

# Exercise Class (2):

The exercise class will take place in the computer room of the physics department (PH EG 004 F).

You can choose to program the exercises in **C** or **matlab**.

Computers are equipped with compilers and plotting programs. We will provide you help and instructions; two tutors will be present at all times during the exercises, to help you with problems and questions.

**If you don't have a user account on the TU server, please contact your lecturer (Boeri, Heil, Kapper).**

If you would like to ask any questions to the lecturers, you can find us during the exercise hours, or request an appointment by email.

# Exercise Class (3):

The computer room can accommodate 16 people at most. Therefore you will be divided in **groups** ( $\leq 16$  people), depending on the initial of your surname (alphabetical order).

For the moment, we have 63 people registered (4 groups).

<b>1<sup>st</sup> Group:</b>	<b>10:00-11:30</b>	Lilia Boeri
<b>2<sup>nd</sup> Group:</b>	<b>11:30-13:00</b>	Gernot Kapper
<b>3<sup>rd</sup> Group:</b>	<b>13:00-14:30</b>	Gernot Kapper
<b>4<sup>th</sup> Group:</b>	<b>14:30-16:00</b>	Cristoph Heil

For the moment, we have 63 people registered (4 groups).  
If you have not done so already, please register to the exercise class on TUG online today; we may have to organise a fifth group.

# Exercise Class (4):

There are two pre-conditions to pass the exercise class:

- 1) Presence at 10 out of 14 dates of the exercises – The tutors will record your presence, don't forget to sign!
- 2) Successful presentation of **all** four exercises before the 10<sup>th</sup> of March 2014. Penalties apply if the exercises are presented later than the set date (Abgabetermin).

## **Abgabetermine WS 2013/14:**

**1<sup>st</sup> Exercise:** 22.10.2013.

**2<sup>nd</sup> Exercise:** 19.11.2013.

**3<sup>rd</sup> Exercise:** 10.12.2013.

**4<sup>th</sup> Exercise:** 21.1.2014.

# Point System (1):

The final grade for the exercise class will be determined according to the **following scheme:**

- Exercise 1: 6 points max.
- Exercise 2: 12 points max.
- Exercise 3: 12 points max.
- Exercise 4: 12 points max.
- Total: 42 points (max).

This adds up to:

- $\text{Tot} \leq 18$ : Nicht genuegend (5).
- $19 \leq \text{Tot} \leq 24$ : Genuiegend (4).
- $25 \leq \text{Tot} \leq 30$ : Befriedingend (3).
- $31 \leq \text{Tot} \leq 36$ : Gut (2).
- $37 \leq \text{Tot} \leq 42$ : Sehr Gut (1).

# Point System (2):

A successful presentation of each exercise, including plots, is worth 12 (or 6) points\*. The mark can be reduced for the following reasons:

- **Delayed presentation** (after Abgabetermin): **2 points.**
- Poor knowledge of the **physical/numerical** problem: **1 point.**
- Poor knowledge of the **program**: **2 points.**
- Poor **presentation** of the results: **2 points.**

**IMPORTANT:** Please note that, according to a new TU regulation, all students who attend the first two classes of an exercise class and then “drop out” of the course, will fail (5) the exercise class. We will enforce this rule for all students who present the first exercise and “disappear” afterwards!

*\* Penalties are halved for the 1<sup>st</sup> exercise.*

# Script:

➤ For this lecture, I will follow the script of Prof. H. Sormann,

*(Numerische Methoden in der Physik, 515.421).*

You can get a copy (**in German**) at the OH Referat, or at the following web address (pdf):

[http://itp.tugraz.at/LV/boeri/NUM\\_METH/script.html](http://itp.tugraz.at/LV/boeri/NUM_METH/script.html)

➤ I have translated in **English** the parts which are relevant to this lecture (and exercises).

You can get a copy **of the English translation** at the following web address (pdf):

[http://itp.tugraz.at/LV/boeri/NUM\\_METH/script.html](http://itp.tugraz.at/LV/boeri/NUM_METH/script.html)

➤ Another useful reference is: *R.W. Hamming, Numerical Methods for Scientists and Engineers, Dover Books (in English).*



## TOPICS (all):

- Chapter 1: Introduction.
- Chapter 2: Numerical methods for the solution of linear inhomogeneous systems.
- Chapter 3: Interpolation of point sets.
- Chapter 4: Least-Squares Approximation.
- Chapter 5: Numerical solution of transcendental equations.
- Chapter 6: Numerical Integration.
- Chapter 7: Eigenvalues and Eigenvectors of real matrices.
- Chapter 8: Numerical Methods for the solution of ordinary differential equations: initial value problems.
- Chapter 9: Numerical Methods for the solution of ordinary differential equations: marginal value problems.





## TOPICS (**this year**):

- **Chapter 1: Introduction.**
- **Chapter 2: Numerical methods for the solution of linear inhomogeneous systems.**
- Chapter 3: Interpolation of point sets.
- **Chapter 4: Least-Squares Approximation.**
- **Chapter 5: Numerical solution of transcendental equations.**
- Chapter 6: Numerical Integration.
- Chapter 7: Eigenvalues and Eigenvectors of real matrices.
- **Chapter 8: Numerical Methods for the solution of ordinary differential equations: initial value problems.**
- Chapter 9: Numerical Methods for the solution of ordinary differential equations: marginal value problems.



Any Questions?



# The First Lecture:

## Practicalities:

- Structure of the course.
- Important dates.
- Exercise class (rules and practical matters).
- Exams and grades.
- Script.

## Introduction

- **What are numerical methods? (Basic Concepts)**
- **What are the most important applications?**



# Introduction

*What are numerical methods?*



**Why use computers?** Because we deal with problems which are too complex to be solved analytically.




But using computers requires **additional care, because ...**  
**Finite size; design suitable algorithms etc.**

**“Computers are useless, they can only give you answers” ...**  
**(P. Picasso)**

# Basic Concepts:

- Applied Mathematics.
- Algorithms.
- Numbers: Round-off and Truncation Errors.
- Feedback and Stability.
- Finite Differences.

From R.W. Hamming, Numerical Methods for Scientists and Engineers,  
***"An essay on numerical Methods"***.



**Applied Mathematics:** simulate mathematical processes, which simulate *real-world* situations. There is a *specific purpose* behind computing.

**Numerical Methods**  
(Applied)

**vs.**

**Numerical Analysis**  
(Theory)

*The purpose of computing is insight, not numbers.*



The choice of a particular formula, or ***algorithm***, influences not only the computing, but also the way we are to understand the results when they are obtained.



## Algorithms:

- **Definition:** A finite number of precise instructions, which require specific input data and, executed in a given sequence, determine the final solution.

Often, there are several algorithms available for a class of problems (for example, eigenvalue problems, Fourier interpolation, etc...). It is important to be able to judge which of the methods is most suited for a given problem. Since we are not dealing with abstract problems ...

... the choice of a particular formula, or *algorithm*, influences not only the computing, but also the way we are to understand the results when they are obtained.

## Numbers:

Computers use a **finite-digit algebra** to simulate the mathematicians' number system which uses infinitely long representations.

This leads to two main sources of errors in numerical methods:

**Round-off error:** is introduced when a finite number of digits is used to represent a real number.

$$\frac{1}{3} \approx 0.33333333$$

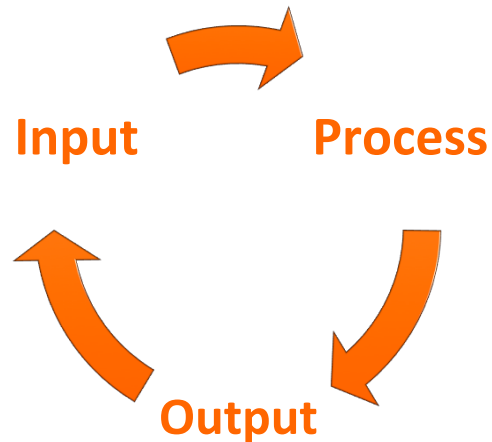
**Truncation error:** Many of the processes of mathematics (differentiation, integration, etc) imply the use of limit, which is an infinite process. A computer has a finite speed and can only perform a finite number of operations in a finite amount of time.

$$\int dx \rightarrow \sum_i$$



## Feedback and Stability:

A central concept in numerical methods is *feedback*: the numbers produced at one stage are fed back into the computer to be processed again and again (**feedback loops**).



**Stability:** Will a small error grow or decay through the successive iterations?  
(*example: shower temperature or beginner driver*)

## Finite differences:

In numerical methods **finite difference** calculus replaces **infinitesimal** calculus.  
(instead of going to the limit, the finite calculus stops at a fixed step size).

Difference calculus:



Differential calculus

Summation calculus:



Integral calculus

Difference equations:



Differential equations

Obviously, these methods encompass some of the most important problems of physics (and engineering).

# Typical Applications:

- Interpolation (polynomial, Fourier...)
- Least square fitting...
- Solving systems of linear equations (linear algebra).
- Finding zeroes of a function.
- Solving systems of differential equations.
- Random number generators.

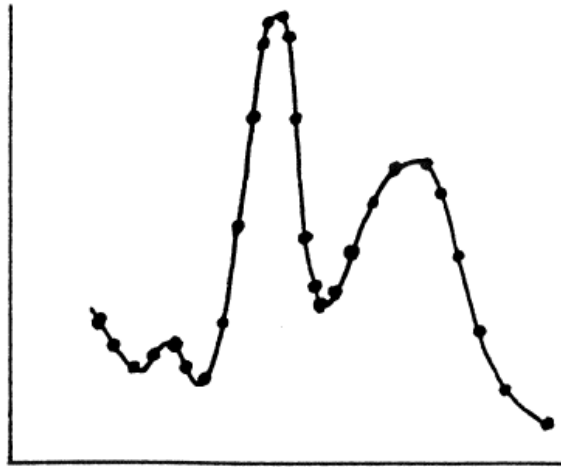


## TOPICS (from prof. Sormann's script) :

- **Chapter 1: Introduction.**
- **Chapter 2: Numerical methods for the solution of linear inhomogeneous systems.**
- Chapter 3: Interpolation of point sets.
- **Chapter 4: Least-Squares Approximation.**
- **Chapter 5: Numerical solution of transcendental equations.**
- Chapter 6: Numerical Integration.
- Chapter 7: Eigenvalues and Eigenvectors of real matrices.
- **Chapter 8: Numerical Methods for the solution of ordinary differential equations: initial value problems.**
- Chapter 9: Numerical Methods for the solution of ordinary differential equations: marginal value problems.

## Interpolation:

- ➔ **The problem:** Given a finite number of data points  $(x_n, y_n)$ , we want to determine a function which passes through them **exactly** (interpolation).



The easiest assumption is an  $n^{\text{th}}$  order **polynomial**:

$$y_n(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_0$$

Such that:

$$y_n(x_i) = y_i$$

More refined polynomial approximation methods use also **derivatives** of the function in the points  $x_i$ .

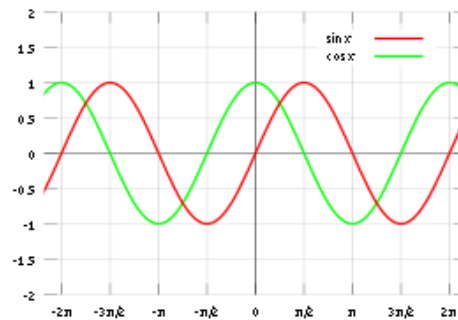
- **Analytic Substitution:** in place of a function we could not handle (or we do not know) we take some samples, exactly match a polynomial to the data, and finally analytically operate this polynomial.
- **Example:** Simpson's formula (assuming we know  $f(1), f(0), f(-1)$ ):

$$P_2(x) = \frac{x(x-1)}{2} y(-1) + (1-x^2) y(0) + \frac{x(x+1)}{2} y(1)$$

Its integral is given by:

$$\int_{-1}^1 y(x) dx = \frac{1}{3} y(-1) + \frac{4}{3} y(0) + \frac{1}{3} y(1)$$

- ➔ **Interpolating functions:**  $n^{\text{th}}$  order polynomials are an easy (natural) choice, but they are not convenient for many physical problems. In particular, they are not good for **periodic** (oscillating) functions.



- ➔ **Fourier (spectral) analysis:**

$$y(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx)$$

$$a_k = \frac{1}{\pi} \int_0^{2\pi} y(x) \cos kx \, dx$$

$$b_k = \frac{1}{\pi} \int_0^{2\pi} y(x) \sin kx \, dx$$

## Zeroes of a function:

- ➔ **The problem:** Finding the zeroes of a function  $F(x)$ , i.e. determining the solutions of the equation:

$$F(x_i) = 0$$

### “Classical” Mathematics (Numerical Analysis):

- Analytical Solution (not always possible).
- Graphical Solution (always possible).

The zeroes of the function can be determined with arbitrary precision.

The situation is very different when we use a computer, because the very nature of the problem changes, when we deal with a **finite resolution**.



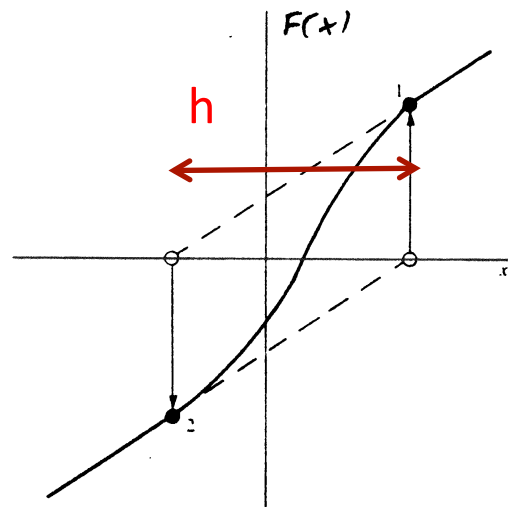


## Numerical solution:

Two main problems:

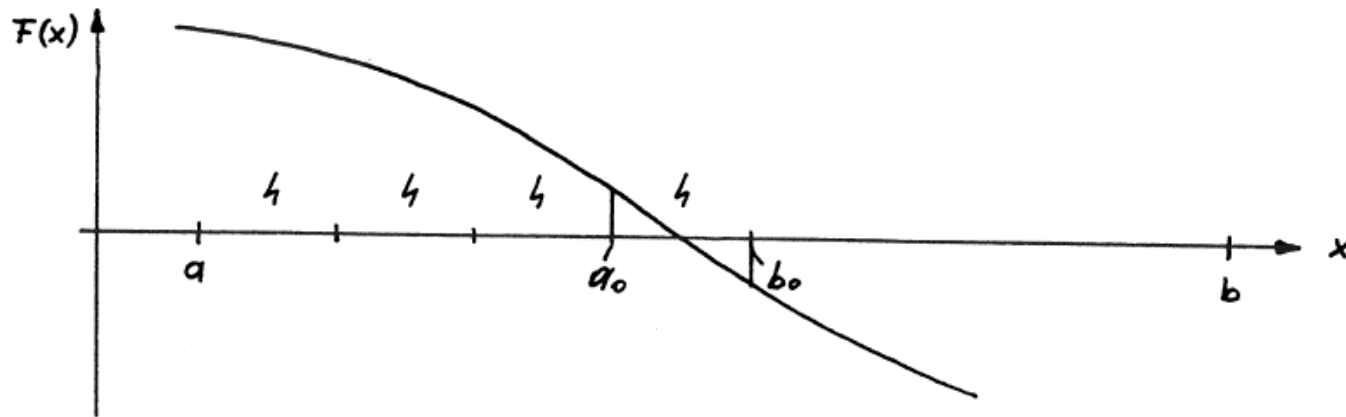
- **Floating point system:** The relative error is hard to keep under control; it is hard to obtain zeroes with considerable absolute accuracy far from the origin.
- **Round-off error:** close to a zero the positive and negative terms of the function cancel almost exactly. The effect of roundoff error is therefore **severe**.
- Furthermore, **round-off** limits the precision with which the zeroes can be determine.

- ➔ **Idea:** Instead of looking for the zeros of a function, we look for an interval where the function **changes sign**...



Once this interval is found, we can further divide it to determine the position of the zeroes with increasing precision (**h**).

- ➔ **Iteration:** The method can be iterated (nested interval method + gross search).



- 1) Gross search: divide the total interval in several sub-intervals of width  $h$ .
- 2) Find the intervals for which  $F(x_i) \cdot F(x_i+h) < 0$ .
- 3) Subdivide these into smaller intervals ( $h/2$ ).
- 4) Iterate up to the required precision.

- **Problem:** With this method, one can only determine odd numbers of zeroes within an interval (the number and character of zeroes is limited by machine precision).

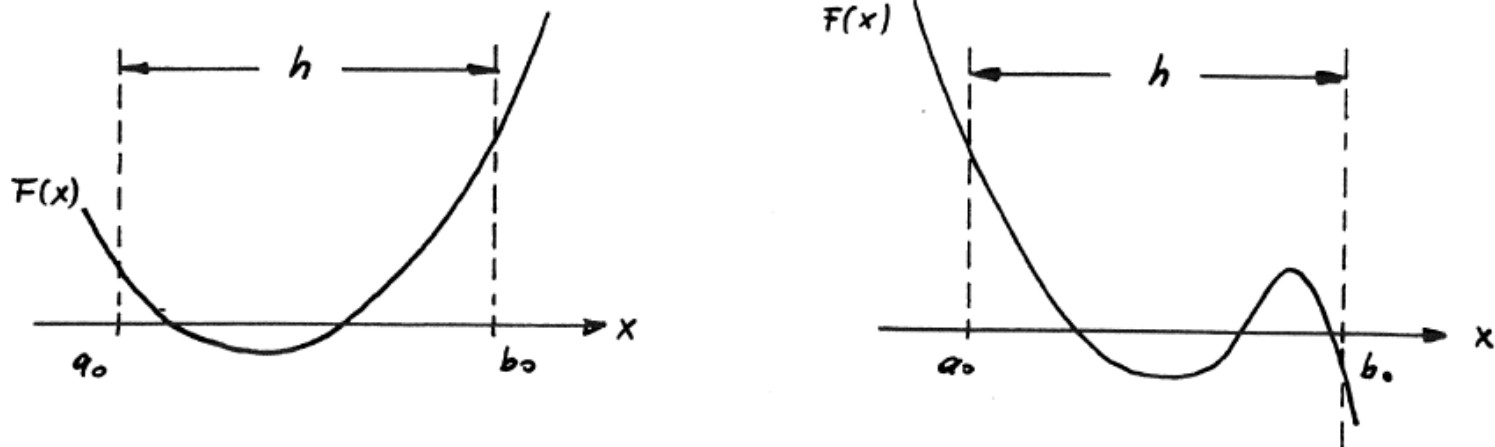


Figure 4.9: Problems connected with the method of nested intervals.



Any Questions?

