Selected topics in Computational Physics
SS 17

A. Scrinzi

This seminar is devoted to topics that are closely related to ongoing research at the chair. Apart from contributions from within the research groups we offer a set of topics for student presentations.

1 Seminar topics

I offer to supervise the following student seminars:

- Non-linear wave equations: Hartree, Hartree-Fock, DFT, and Gross-Pitaevski
- A tRecX implementation of non-linear wave dynamics
- Tensor and low-rank approximations: the H-matrix technique
- Efficient integrators for non-linear systems.

The first topic is largely theoretical, all other topics include computational practice in varying extent. The exact part of coding and computation can be adjusted, depending on your primary interest.

If you are interested in any of the above, send an e-mail to armin.scrinzi@lmu.de before the next seminar on Thursday, Nov 11, 2016.

2 Brief descriptions and materials

2.1 Non-linear wave equations

There is a family of closely related non-linear, mean-field type of equations that play an important role in physics: the Hartree ansatz extensively used for chemical reaction dynamics, the Hartree-Fock ansatz that forms the basis of all computatial chemistry and plays an important role in solid state physics, Density Functional Theory (DFT), that is the starting point for all modern computations material science, and Gross-Pitaevski equation that describes the ground state and basic dynamics of Bose-Einstein condensates.

In the seminar you will report these methodes (the exact selection and amount of detail remains to be decided) and compare them, in particular with respect to their computational requirements and efficiency.

2.2 A tRecX implementation of non-linear wave dynamics

Picking one of the non-linear equations above (good candidate: Gross-Pitaevski), you implement it the context of an exising package. The tRecX package (trecx.physik.uni-muenchen.de) is a modern C++ structure for solving partial differential equations, such as the time-dependent Schrödinger equation.

You will learn to handle a realistic compuational task and report actual solutions for problems like, say, a Bose-Einstein condensate on a microchip.

2.3 The H-matrix technique

Matrices can often be represented efficiently by structures of tensor product form. For example, a large $MN \times MN$ matrix $\widehat{A}$, could be represented as short sum of tensor products of $M \times M$ matrices $\widehat{B}_i$ with $N \times N$ matrices $\widehat{C}_j$

$$\widehat{A} = \sum_{i=1}^{I} \widehat{B}_i \otimes \widehat{C}_i. \tag{1}$$

Among others, one sees that the information seemingly present in $\widehat{A}$, namely $(MN)^2$ different numbers, may be representable by $I \times M^2 \times N^2$ numbers, which, depending on the size of I, are many fewer numbers than the original $(MN)^2$. Detection of such a condensed representation teaches use about the deeper structure of the problems. On a practical side, it will not come as a suprise that the condensed form of the matrix on the right hand side allow much more efficient computataions than the full matrix $\widehat{A}$.

Such situation arises naturally in quantum mechanics. Similar ideas underly the now popular "matrix product states" used in correlation physics. They are also used for the efficient representation of complicated many-body potentials.

The H-matrix technique draws on such ideas. You will report it and demonstrate performance compared to matrix techniques, using available software.

Alternatively, closer to Computational practice, you implement the method yourself, where the C++ tRecX provides a useful environment, but implemention in an environment like Matlab or Python will be feasible, if you are skilled with those.

2.3.1 Literature

- W. Hackbusch, A sparse matrix arithmetic based on H-matrices. Part I: Introduction to H-matrices, Computing (1999), 62:89108
- M. Bebendorf, Hierarchical matrices: A means to efficiently solve elliptic boundary value problems, Springer (2008)
- W. Hackbusch, Hierarchische Matrizen. Algorithmen und Analysis, Springer (2009)

2.4 Time-integrators

The tRecX framework contains a general frame for solvers for ordinary differential equations of the form

$$\frac{d}{dt}\vec{y} = f(\vec{y}, t). \tag{2}$$

You may have encountered the Runge-Kutta methods. In the non-linear wave equation mentioned above, a rather complicated, non-linear rhs. $f(\vec{h}, t)$ arises, whose structure, however, favors certain so-called "split-step" time-integrators.

We will suggest the implementation and use of two types of integrates and compare with the performance of the existing Runge-Kutta method.