

Project 2, Variational Monte Carlo studies of electronic systems. Deadline May 31

Computational Physics II FYS4411/FYS9411

Department of Physics, University of Oslo, Norway

Spring semester 2017

Introduction

The aim of this project is to use the coupled cluster method to evaluate the ground state energy of quantum dots with $N = 2$, $N = 6$, $N = 12$ and $N = 20$ electrons. These are so-called closed shell systems.

Theoretical background and description of the physical system

We consider a system of electrons confined in a pure two-dimensional isotropic harmonic oscillator potential, with an idealized total Hamiltonian given by

$$\hat{H} = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 + \frac{1}{2} \omega^2 r_i^2 \right) + \sum_{i < j} \frac{1}{r_{ij}}, \quad (1)$$

where natural units ($\hbar = c = e = m_e = 1$) are used and all energies are in so-called atomic units a.u. We will study systems of many electrons N as functions of the oscillator frequency ω using the above Hamiltonian. The Hamiltonian includes a standard harmonic oscillator part

$$\hat{H}_0 = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 + \frac{1}{2} \omega^2 r_i^2 \right),$$

and the repulsive interaction between two electrons given by

$$\hat{H}_1 = \sum_{i < j} \frac{1}{r_{ij}},$$

with the distance between electrons given by $r_{ij} = |\mathbf{r}_1 - \mathbf{r}_2|$. We define the modulus of the positions of the electrons (for a given electron i) as $r_i = \sqrt{r_{ix}^2 + r_{iy}^2}$.

© 1999-2017, "Computational Physics II

FYS4411/FYS9411": "<http://www.uio.no/studier/emner/matnat/fys/FYS4411/index-eng.html>". Released under CC Attribution-NonCommercial 4.0

license

Project 2 a): In exercises a-d we will deal only with a system of two electrons in a quantum dot with a frequency of $\hbar\omega = 1$. The reason for this is that we have exact closed form expressions for the ground state energy from Taut's work for selected values of ω , see M. Taut, Phys. Rev. A **48**, 3561 (1993). The energy is given by 3 a.u. (atomic units) when the interaction between the electrons is included. If only the harmonic oscillator part of the Hamiltonian, the so-called unperturbed part,

$$\hat{H}_0 = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 + \frac{1}{2} \omega^2 r_i^2 \right),$$

the energy is 2 a.u.

Project 2 b):

Project 2 d):

Project 2 e): The previous exercises have prepared you for extending your calculational machinery to other systems. Here we will focus on quantum dots with $N = 6$, $N = 12$ and $N = 20$ electrons.

Project 2 g): The last exercise is a performance analysis of your code(s) for the case of $N = 6$ electrons. Make a performance analysis by timing your serial code with and without vectorization. Perform several runs with the same number of Monte Carlo cycles and compute an average timing analysis with and without vectorization. Comment your results. Use at least 10^6 Monte Carlo samples.

Compare thereafter your serial code(s) with the speedup you get by parallelizing your code, running either OpenMP or MPI or both. Do you get a near 100% speedup with the parallel version? Comment again your results and perform timing benchmarks several times in order to extract an average performance time.

Literature.

- M. Taut, Phys. Rev. A **48**, 3561 - 3566 (1993).
- B. L. Hammond, W. A. Lester and P. J. Reynolds, *Monte Carlo methods in Ab Initio Quantum Chemistry*, World Scientific, Singapore, 1994, chapters 2-5 and appendix B.
- B.H. Bransden and C.J. Joachain, *Physics of Atoms and molecules*, Longman, 1986. Chapters 6, 7 and 9.
- A. K. Rajagopal and J. C. Kimball, see Phys. Rev. B **15**, 2819 (1977).
- M. L. Pedersen, G. Hagen, M. Hjorth-Jensen, S. Kvaal, and F. Pederiva, Phys. Rev. B **84**, 115302 (2011)

Introduction to numerical projects

Here follows a brief recipe and recommendation on how to write a report for each project.

- Give a short description of the nature of the problem and the eventual numerical methods you have used.
- Describe the algorithm you have used and/or developed. Here you may find it convenient to use pseudocoding. In many cases you can describe the algorithm in the program itself.
- Include the source code of your program. Comment your program properly.
- If possible, try to find analytic solutions, or known limits in order to test your program when developing the code.
- Include your results either in figure form or in a table. Remember to label your results. All tables and figures should have relevant captions and labels on the axes.
- Try to evaluate the reliability and numerical stability/precision of your results. If possible, include a qualitative and/or quantitative discussion of the numerical stability, eventual loss of precision etc.
- Try to give an interpretation of your results in your answers to the problems.
- Critique: if possible include your comments and reflections about the exercise, whether you felt you learnt something, ideas for improvements and other thoughts you've made when solving the exercise. We wish to keep this course at the interactive level and your comments can help us improve it.
- Try to establish a practice where you log your work at the computerlab. You may find such a logbook very handy at later stages in your work, especially when you don't properly remember what a previous test version of your program did. Here you could also record the time spent on solving the exercise, various algorithms you may have tested or other topics which you feel worthy of mentioning.

Format for electronic delivery of report and programs

The preferred format for the report is a PDF file. You can also use DOC or postscript formats or as an ipython notebook file. As programming language we prefer that you choose between C/C++, Fortran2008 or Python. The following prescription should be followed when preparing the report:

- Use Devilry to hand in your projects, log in at <http://devilry.ifi.uio.no> with your normal UiO username and password.

- Upload **only** the report file! For the source code file(s) you have developed please provide us with your link to your github domain. The report file should include all of your discussions and a list of the codes you have developed. The full version of the codes should be in your github repository.
- In your github repository, please include a folder which contains selected results. These can be in the form of output from your code for a selected set of runs and input parameters.
- Still in your github make a folder where you place your codes.
- In this and all later projects, you should include tests (for example unit tests) of your code(s).
- Comments from us on your projects, approval or not, corrections to be made etc can be found under your Devilry domain and are only visible to you and the teachers of the course.

Finally, we encourage you to work two and two together. Optimal working groups consist of 2-3 students. You can then hand in a common report.