# FYS-4096 Computational Physics: Exercise 9

Ilkka Kylänpää

Week 12

**General:**

- You should have access to `ex9_help` on teacher's repo.
- There you will find, e.g., files `build_csc_puhti_complex_only.sh`, `problem4_help.py`, and `optim_params.py` that will most likely be useful in this exercise.
- You also need a username and password to access the CSC supercomputing facilities.
- For general future use the QMCPACK website might become useful for QMCPACK and Nexus related details.
- General information on Puhti at CSC might also be of use.
- **Deadline always before noon on Monday the following week!** That is, for these problems before noon on Monday March 29.

**Problem 1: Variational and Diffusion Monte Carlo**

- Consider the code `vmc_dmc_simple.py` and make comments to identify details on the lecture notes.
- Use the VMC code to solve the Hydrogen molecule with internuclear distance of $1.4a_0$ using a trial wave function

$$\Psi_T(\mathbf{r}_1, \mathbf{r}_2) \propto (e^{-ar_{1A}} + e^{-ar_{1B}})(e^{-ar_{2A}} + e^{-ar_{2B}}),$$

  where $A$ and $B$ refer to the nuclei, i.e, protons, and $a > 0$. Use $a = 1.1$ at first.
- When the parameters are set up correctly, then adjust the VMC time step to obtain a reasonable acceptance ratio. The desired acceptance ratio is mentioned in the lecture notes.
- Once the VMC time step is reasonable, make the code print the VMC total energy at the end of the simulation. Remember that the error estimate, which is similar to the one in the Monte Carlo integration of the first exercise set, but without the multiplication by the "volume".
- Also, print out the absolute value of the variance to energy ratio, which is one kind of metric on the "goodness" of the wave function.
- Once these prints are working as expected, consider a couple of different values for $a$, i.e., $a = 1.1, 1.2, 1.3, 1.4, 1.5$. For these plot the total energies with the errorbars, and plot also the variance to energy ratio.
- From the previous $a$ values, choose the one that gives you the minimum energy... most likely also the minimum variance to energy ratio. Use that $a$-value for the next bullet point.
- Now that the wave function is (somewhat) optimized, test also the DMC runs. Notice the difference in the needed computer time between VMC and DMC runs. Notice neither of the routines are optimized, so even this simple case can take some time. Compare the total energies from VMC and DMC.

- What kind of Jastrow factor is used? Does the inclusion of Jastrow factor improve your VMC result? To see this, test the code with and without the Jastrow term using a poor wave function, that is, use $a = 0.5$ and also modify the $b_1 = 50.0$ to, e.g., $b_1 = 0.5$ in the electron-Ion Jastrow. With a good wave function, simple jastrow cannot improve much, but for a poor wave function...
- Make some random thoughts (a few sentences or bullet points) on how to improve the efficiency of VMC and DMC in general in this code, and on how one could perform optimization of wave function parameters with VMC.
- Notice: the exact energy is roughly $-1.1745$ at internuclear distance $1.4a_0$. However, VMC is only as good as the wave function used, but DMC, in principle, should get it exactly (within numerical accuracy).

**Problem 2: QMCPACK installation on Puhti at CSC**

- Open a terminal.
- Write `ssh username@puhti.csc.fi` and press enter (as username you need to use your own csc student username).
- It asks for a password, which you should know already at this point.
- Once logged in, go to your project application folder `cd /projappl/project.../student...`
- Next go to your qmcpack folder and `config` therein.
- Either copy the file `build_csc_puhti_complex_only.sh` into the folder or create a file with the same contents into the folder.
- Then go to the qmcpack folder and run command
  `./config/build_csc_puhti_complex.sh`
- This should start building QMCPACK. Notice that you should be able to execute the file... if you do not have permissions, write, e.g., `chmod +x filename` for proper permission for the file "filename".
- Look into the configure files, so that you know what modules were loaded etc. These modules need to be loaded when you are submitting jobs running this program.
- Add qmcpack binary location to `.bashrc`. Should be somewhere like build_csc_puhti_complex_only/bin/ at the qmcpack folder level.
- Write source .bashrc after which you should have access to qmcpack executable. (For some reason you might need to logout and back in for proper refresh.) **Include your CSC `.bashrc` file as `dot_bashrc.txt` into your git return.**

**Problem 3: Calculation of diamond example using VMC**

- You should find diamond examples files at nexus folder at
  `/projappl/project_.../student.../qmcpack/nexus/examples/qmcpack/rsqmc_misc/diamond/`
- **First set generate only to 1** in order to avoid sending the job with wrong settings. (You can always delete the job from queue also, but this is more convenient.)
- Notice that you might want to use the `machine_configs.py` in the `ex9_help` files. Alternatively, you can use the one from last weeks exercise, but then you do need to modify it to include `pw2qmcpack.x` and `qmcpack` executables. Whichever route you choose, you still do need to include the needed modules into the file!
- Remember that all the runs need to be done at the work directory, i.e., /scratch/project.../student.../:
  - Go to puhti.csc.fi, i.e, `ssh username@puhti.csc.fi`

  - Go to your work directory, i.e, `cd /scratch/project.../student...`
  - Once in your work directory make a folder for this problem, go to that folder, and copy the required files: "cp -r from_folder/* ."
  - For running the Nexus python script remember to load python-env.
  - If you are using rsync (or scp) from your own computer, then you can copy files and folders, e.g., as `rsync -av example_folder username@puhti.csc.fi:location_at_csc/.`

- The QMCPACK energy estimate can be obtained, e.g., by command
  `qmca -e 5 -q ev *scalar.dat`
- Try to find out what `-e` and `-q` stand for. Are there any other options? (Hint: `--help`)
- Look at the structure files at our scf and vmc folders. Are they the same? Should they be the same or different, i.e., what kind of cell has been used in each calculation?
- Compare the energies from VMC and density functional theory. If the VMC energy is roughly 8 times larger in magnitude, then everything is good. Why is this? (The energies should be roughly $-22.52257768$ Ry and $-87.846646 \pm 0.025$ Ha, 1 Ry $= 0.5$ Ha. Hint: input has `tiling = (2,2,2)`)
- Notice that this VMC wave function is definitely not optimal.

**Problem 4: Jastrow optimization and diffusion Monte Carlo for the diamond case**

- Do this also at CSC, and in your work directory.
- Look at the `problem4_help.py` and `optim_params.py` files.
- Use those files to modify your `diamond.py` to include both the wave function optimization and the diffusion Monte Carlo runs into the same workflow.
- Compare the energies of the VMC without Jastrow, VMC with optimized Jastrow, and DMC with the two different time steps. Make a plot of the comparison.
- Make also a plot showing the total energy at different stages in the optimization.

**Returning your exercise**

1. Make new folder "exercise9" to your existing Computational Physics repo.
2. Create a file solvedProblems.txt at the root of your "exercise9" git repo. Inside it, write a comma separated list of problems you have solved, e.g., 1,2,3.
3. Make sure all your source files are under version control and push them to GitLab.
4. Push your commits to GitLab before noon on Monday March 29:
   `git push --all && git push --tags`