Crystallization of ^{4}He in 2D with Variational and Diffusion Monte Carlo

Helium is the only substance in Nature that remains liquid at $T = 0$ K. It crystallizes only under pressure.
To high accuracy, ^{4}He atoms can be considered as point-like particles with zero spin, interacting through a Lennard-Jones (LJ) potential,

$$v(r) = 4\epsilon[(\sigma/r)^{12} - (\sigma/r)^6], \tag{1}$$

with $\epsilon = 10.22$ K and $\sigma = 2.556$ Å (we use energy units of Kelvin and length units of Angstrom).
We will use the code `lj.f` to study ^{4}He in 2D near the crystallization transition at $T = 0$ with the VMC and DMC algorithms.
Our trial function will be of the Jastrow form with two– and three–body correlations,

$$\Psi_L(R) = e^{-\sum_{i<j} u_L(r_{ij})} e^{-\sum_i \sum_{j\neq i} \xi(r_{ij})\mathbf{r}_{ij}\cdot\sum_{k\neq i}\xi(r_{ik})\mathbf{r}_{ik}} \tag{2}$$

to describe the liquid phase, and a "Nosanow" function

$$\Psi_S(R) = e^{-\sum_{i<j} u_S(r_{ij})}\ e^{-\sum_i |\mathbf{r}_i - \mathbf{s}_i|^2/d^2} \tag{3}$$

to describe the solid. In the Nosanow function there are two-body correlations and a gaussian factor localizing the ith particle within a distance $\sim d$ around a preassigned lattice site $\mathbf{s}_i$. This trial function breaks the Bose symmetry because each particle is identified by its lattice site. However since particle exchanges are rare in the solid this is a good approximation.

Files:

- `doc_lj.pdf`: this file
- `setup_lj.f`: a setup fortran code
- `lj.f`: the QMC fortran code[1]
- `lj.h`: a file included by `lj.f` at compile time
- `statfor.f`: fortran code for statistical analysis (see Appendix C)
- `statforw.f`: same as `statfor.f` for weighted averages (see Appendix D)

[1] This is essentially the same code used for the electron gas in the lab and in the first test.

Assignments

For each question, write the answer on a text file and keep a copy of the output files. At the end of the test, compress the folder(s) with your work and send it by email to `saveriomoroni@gmail.com`.

Part A. General questions (10 points out of 30)

1. Why do we use correlated wave functions such as (2) and (3)?
2. Why do we use Monte Carlo to calculate expectation values with wave functions such as (2) and (3)?
3. How do we choose the time step (size of the move) in VMC?
4. Does the average energy depend on the time step in VMC?
5. How do we choose the time step and the trial energy in DMC?
6. Does the average energy depend on the time step in DMC?
7. Why do we need a population of walkers in DMC?
8. Does the average energy depend on the number of walkers in DMC?
9. For a potential which diverges at zero distance, do spin-1/2 fermions have a different cusp condition in the Jastrow factor for parallel or antiparallel spins?
10. Does the cusp condition apply to bosons? If the parallel and antiparallel fermion cusp conditions are different, which one applies to bosons?

Part B. Simulations (20 points out of 30)

Compile all four codes `setup_lj.f, lj.f, statfor.f, statforw.f`. With the `gfortran` compiler, use the options `-w` to suppress warning messages, `-O3` to get a faster executable, `-o filename` to choose the name of the executable, and `-fallow-argument-mismatch` if you get argument mismatch errors.

The setup code prompts you to choose the run id (the name of the simulation), the density ρ, and the phase (Jastrow wave funtion for the liquid or Nosanow wave function for the solid). It produces all needed files, with wave functions already optimized, and a file `runid` which contains the chosen run id (the QMC code `lj.f` will run the simulation specified in the file `runid`).

You have to create the input file `xxx.in`, where `xxx` is your chosen run id. See Appendix B for the keywords of the input file.

1. In 2D, ^{4}He is expected to crystallize at a density ρ of about 0.07Å$^-$2. At this density, run a VMC simulation for the liquid and one for the solid (use a different run id for the two cases). Remember to choose an appropriate time step. In at least one case, use the `statfor.f` code to check for possible autocorrelation. In the output files, the entry `e2` is the average of the square of the local energy. Use `e2` and `elocal` to calculate the variance of the local energy for the liquid and for the solid. Based on these variances, which phase is described more accurately by the wave functions generated by the setup?

2. Change the density, increasing and decreasing in steps of $\sim$0.01Å^{-2}, and run a few more VMC simulations to locate the crystallization density (just a rough estimate: a careful calculation, not requested here, would need a Maxwell double-tangent construction on the equation of state $E(1/\rho)$ to locate the coexistence region between the melting and the freezing densities).

3. For $\rho = 0.07$Å^{-2}, run a DMC simulation for the liquid and one for the solid phase. Remember to generate the initial population with VMC (50 walkers is OK) and to choose an appropriate time step and trial energy. In at least one case, use the `statforw.f` code to check for possible autocorrelation. Would a DMC calculation shift the crystallization to lower or higher density? For each of the two phases, compare the VMC and DMC energies: how does this comparison relate with the variance of the local energy (question B1)?

Appendix A: Keywords for the file `xxx.sy` used in this calculation

ndim `d`

`d` (integer): dimensions of the physical space

type `name number hbs2m file`

define a type of particles

`name` (string): name of this type of particles

`number` (integer): number of particles of this type

`hbs2m` (real): value of $\hbar^2/2m$ for this type of particles

`file` (string): file with initial configuration(s) of the particles

v2 `name_a name_b file` iexp

pair potential between particles of type `name_a` and `name_b`

`name_a` (string): name of a type of particles

`name_b` (string): name of a type of particles

`file` (string): name of the file with the tabulated potential

`iexp` (integer): the short-range part of the tabulated potential is multiplied by a factor $r^{-\mathrm{iexp}}$

u2 `name_a name_b file`

two-body Jastrow factor for particles of type `name_a` and `name_b`

`name_a` (string): name of a type of particles

`name_b` (string): name of a type of particles

`file` (string): name of the file with the tabulated Jastrow factor

vshift `value`

`value` (real): a constant shift in the potential [2]

vtail `value`

`value` (real): a tail correction in the potential energy

pbc `L_x [L_y [...]]`

periodic boundary conditions

`L_x` (real): side of the simulation cell in the x direction

[2] The LJ potential $v(r)$ is shifted to zero at $r = L/2$, and taken to be zero for larger distances; `vshift` puts back the shift in the average potential energy, and `vtail` puts back, approximately, the effect of the potential for $r > L/2$.

`[...]` (real): side of the simulation cell in the other direction(s)

u3 `name_a name_b file`

three-body Jastrow factor for particles of type `name_a` and `name_b`

`name_a` (string): name of a type of particles

`name_b` (string): name of a type of particles

`file` (string): name of the file with the tabulated three-body function

sites `name number file`

define a set of lattice sites

`name` (string): name of the lattice sites

`number` (integer): number of sites

`file` (string): name of the file with the lattice sites' positions

nosanow `name_a name_b file`

nosanow function for particles `name_a` and sites `name_b`

`name_a` (string): name of the particles

`name_b` (string): name of the sites

`file` (string): name of the file with the tabulated nosanow function

gofr

turn on the calculation of the pair distribution function

sofk

turn on the calculation of the $S(\mathbf{k})$ for $\mathbf{k}_n = (0, n\frac{2\pi}{L_y})$

Appendix B: Keywords for the file `xxx.in` used in this calculation

vmc `nblocks` `nstep` `tstep` `[istore]`

define a VMC run

`nblocks` (integer): number of blocks

`nsteps` (integer): number of steps per block

`tstep` (real): time step (size of the move)

`istore` (integer, optional): store a configuration every `istore` step

dmc `nblocks` `nstep` `tstep` `nwalkers` `etrial`

define a DMC run

`nblocks` (integer): number of blocks

`nsteps` (integer): number of steps per block

`tstep` (real): time step

`nwalkers` (integer): number of walkers

`etrial` (real): initial value of the trial energy

Appendix C: Statistical analysis using the code `statfor.f`

compile the code, e.g. `gfortran statfor.f -o statfor.x`

the code reads a stream of numbers from standard input

writes number of data, mean, variance, correlation time, number of independent data and statistical error on standard output

also writes: a 51-bin histogram of data on file 'histo.out', and the autocorrelation of the data on file 'corr.out'

to analyze (say) the block averages of elocal from file `vmc.out`, use the command:

```
grep elocal vmc.out | ./statfor.x
```

Appendix D: Statistical analysis using the code `statforw.f`

compile the code, e.g. `gfortran statforw.f -o statforw.x`

the code reads two streams of numbers (data and weights) from standard input

writes number of effective data in the weighted average, mean, variance, correlation time, number of independent data and statistical error on standard output

also writes: a 51-bin histogram of data on file 'histo.out', and the autocorrelation of the weighted data on file 'corr.out'

to analyze (say) the block averages of elocal from file `dmc.out`, use the command:

```
grep elocal dmc.out | awk '{print $1, $4}' | ./statforw.x
```