Joint ICTP–DEMOCRITOS
Advanced School on Quantum Monte Carlo Methods in Physics and Chemistry
ICTP, Trieste, Italy, 21 January–2 February 2008

Lab on Phaseless Auxiliary-Field Quantum Monte Carlo: trapped atomic Bose Gas

Wirawan Purwanto and Shiwei Zhang

January 29, 2008

Abstract

We describe `bh1d`, a pedagogical program for learning the basics of the phaseless auxiliary-field quantum Monte Carlo method. This program calculates the ground-state properties of a one-dimensional boson gas, either confined in a harmonic trap or under a periodic boundary condition. This handout is for the lab session at the "Advanced School on Quantum Monte Carlo Methods in Physics and Chemistry", 21 Jan.–2 Feb. 2008.

1 Boson Gas on a Lattice

Consider a one-dimensional lattice containing L points, as shown in Fig. 1. There are N identical spinless bosons living on this lattice, described by the Hamiltonian:

$$\hat{H} \equiv \underbrace{-t \sum_{\langle i,j \rangle} (c_i^\dagger c_j + c_j^\dagger c_i)}_{\hat{T}} + \underbrace{\tfrac{1}{2}\kappa \sum_i |\mathbf{r}_i - \mathbf{r}_0|^2 \hat{n}_i}_{\hat{V}_{\text{trap}}} + \underbrace{\tfrac{1}{2} U \sum_i \hat{n}_i \hat{n}_i}_{\hat{U}}\,, \tag{1}$$

$$= -t \sum_{i=1}^{L-1} (c_i^\dagger c_{i+1} + c_{i+1}^\dagger c_i) - \delta_{\text{bc}} t (c_L^\dagger c_1 + c_1^\dagger c_L) + \tfrac{1}{2}\kappa \sum_i |\mathbf{r}_i - \mathbf{r}_0|^2 \hat{n}_i + \tfrac{1}{2} U \sum_{i=1}^{L} \hat{n}_i \hat{n}_i\,, \tag{2}$$

where $\hat{n}_i \equiv c_i^\dagger c_i$ is the density operator, $\langle i, j \rangle$ denotes two nearest-neighbor sites, and $|\mathbf{r}_i - \mathbf{r}_0|$ is the distance of site i to the trap center. The interparticle interaction is an on-site attractive interaction of strength $|U|$.

```
    o----o----o----o . . . o----o- - *
i = 1    2    3    4      L-1   L    1 again (if PBC on)
```

Figure 1: Illustration of the one-dimensional lattice used in this problem.

The boundary condition is determined by δ_{bc}: 0 for open (i.e. the wave function is zero at the boundaries), or 1 for periodic boundary condition (PBC). When the trap potential is nonzero, we typically choose $\delta_{\text{bc}} = 0$.

2 Overview of the `bh1d` AFQMC Program

A summary of the boson AFQMC method is given in the appendix. A more detailed description is given in the references (see also lecture notes). The simple C++ program, "`bh1d`" (Bose-Hubbard in 1-D), is a pedagogical tool to learn this method and the related phaseless AFQMC method for fermion systems. In brief, an AFQMC calculation using the `bh1d` program contains the following steps:

1. At the beginning of the calculation, a population of random walkers are initialized to $|\Psi_{\rm T}\rangle$.

2. The first phase is the *equilibration* phase consisting of `NEqSteps` random-walk steps.

3. The second phase is the *growth* phase to obtain an improved estimate of the trial energy $E_{\rm T}$. This phase can be repeated `NAdjustCount` times, each taking `NAdjustSteps` random-walk steps.

4. Then comes the *measurement* phase, which contains `NBlocks` blocks of `NBlkSteps` random walk steps each. In each block, we carry out (`NBlkSteps`/`itvMeasure`) measurements, and average their results. This average is termed a *block measurement*.

5. Finally, the statistics is computed by averaging the block measurements. This result is printed in the output file as the `BLK_AVG` and `ERROR` columns (see Sec. 3.1).

The `bh1d` code does not have importance sampling. The ground-state wave function is given by

$$|\Phi_0\rangle \doteq \sum_{\rm i} w_{\rm i} |\phi_{\rm i}\rangle \,. \tag{3}$$

We use the *mixed-estimate* to compute the expectation of ground-state observables:

$$\langle A\rangle_{\rm mixed} = \frac{\langle\Psi_{\rm T}|\hat{A}|\Phi_0\rangle}{\langle\Psi_{\rm T}|\Phi_0\rangle} \doteq \frac{\sum_{\rm i} w_{\rm i}\langle\Psi_{\rm T}|\hat{A}|\phi_{\rm i}\rangle}{\sum_{\rm i} w_{\rm i}\langle\Psi_{\rm T}|\phi_{\rm i}\rangle} \tag{4}$$

This estimate is only exact if $\hat{A}$ commutes with the Hamiltonian. An alternative estimator is the *brute-force estimate*, obtained by replacing the trial wave function $\langle\Psi_{\rm T}|$ above by the Monte Carlo samples of the ground-state wave function, $\sum_{\rm j}\langle\phi_{\rm j}|$. This is exact, but is often noisy. We will explore its behavior in the exercises.

2.1 Obtaining the Code

An automated install procedure is provided. Execute the file

```
/afs/ictp/public/s/shiwei/bh1d-setup.sh
```

This will create a subdirectory `bh1d/` in your home directory, containing the source code, makefile, documentation, and examples. The setup file will also compile the QMC executable code (`bh1d.exe`) and the exact-diagonalization code (`bh1d-exact.exe`).

2.2 `bh1d` Input File Syntax

The `bh1d` input file is line-oriented, containing one command per line. The first word is a keyword, followed by arguments (see the sample input files). All arguments must be in the same line as the keyword. Any text after a '`#`' character is a comment. A more detailed description of the input file syntax is given in the appendix.

To run the QMC program with an input file (say, `qmc.in`):

```
$ ./bh1d.exe qmc.in
```

The output will be written to the file `qmc.out` and also echoed to the standard output (i.e., the terminal). Several other files containing diagnostics output will also appear, all prepended by "`qmc.`".

2.3 Exact-Diagonalization Input File Syntax

The input file for the exact-diagonalization program is primitive, consisting of only *one* line with five numbers: the number of sites, the number of particles, t, U, and κ parameters, in that order. Here is an example:

```
13 3 2.67612 -1.53843 0.350333
# 3 particles on 13 sites
# This is a sample exact-diagonalization input file
```

Comments can only appear after the second line, prefixed with the '#' character.

To run the exact-diagonalization program with an input file (say, `exact.in`), type

```
$ ./bh1d-exact.exe exact.in
```

The output is *always* written to the standard output; to redirect it to a file,

```
$ ./bh1d-exact.exe exact.in > exact.out
```

Beware not to set L and N too large ($\mathcal{O}(10)$). The computing time and memory grow exponentially with them.

3 Exercises

3.1 Running the AFQMC and exact diagonalization programs

The first assignment is to run a "demo" with no input file. The aim is to get a feel for the code, understand the output file, and compare the AFQMC results with those from exact-diagonalization.
Let us start with the AFQMC run (the exact version number of your code may be different):

```
$ ./bh1d.exe
Warning: sample session is now about to run. Its result will not be recorded
into a log file.
Bosonic 1-D Hubbard Model Toy Program  version 0.13.1

SAMPLE PROGRAM SESSION
(use --help switch to learn how to run your own sessions)

Settings:

Nsites = 4, Nparticles = 2
t      = 1, U = 0, dt = 0.025, ET = -4
kappa  = 1e-200
psi_T  = (0.587785 0.951057 0.951057 0.587785)
NWalkers = 200 (limited from 100 to 400)
Weight = 0.2 ... 5
NEqSteps = 1000, NAdjustCount = 1, NAdjustSteps = 1000, NBlocks = 25, NBlkSteps = 300
itvPopCtl = 5, itvMeasure = 10, NWeightHistory = 6
NREqSteps = 500

Population control algorithm: simple-combing
runtime-verbosity = 0
ET_auto = 0, brute_force = 0
Normalized trial WF:
 0.371748 0.601501 0.601501 0.371748
```

...and so on. The final result is printed at the end of the output:

```
[-] GROUND-STATE OBSERVABLES
    Quantities                   BLK_AVG             GRAND                VAR     ERROR
 -  overlap         =                 1       63510.91423                  1
 -  g.s.energy      =      -3.236067977      -3.236067977      -3.236067977   1.4e-08
 -  <T>             =      -3.236067977      -3.236067977      -3.236067977   1.4e-08
 -  <Vtrap>         =  8.027864045e-201  8.027864045e-201  8.027864045e-201         0
 -  <U>             =                 0                 0                 0         0
 -  Density-profile:
     0 (     0 )           0.2763932023      0.2763932023      0.2763932023   7.7e-10
     1 (     1 )           0.7236067977      0.7236067977      0.7236067977   5.4e-09
     2 (     2 )           0.7236067977      0.7236067977      0.7236067977   3.8e-09
     3 (     3 )           0.2763932023      0.2763932023      0.2763932023   1.6e-09
```

The numbers are organized into four columns: `BLK_AVG` is the average of all the block measurements, and its corresponding error estimate is given in the `ERROR` column. The second column gives a *grand average* from a slightly different averaging method. A disagreement beyond one or two error bars between `BLK_AVG` and `GRAND` indicates there is likely a problem with the QMC run (except for the `overlap`). `VAR` gives the variational estimate, i.e., the expectation value of the observables with respect to $\Psi_{\rm T}$. There can be two more result columns if the `brute-force` option is enabled: they are the brute-force averages and the statistical errors.

There are five quantities measured: the total energy, kinetic energy $\langle\hat{T}\rangle$, trap energy $\langle\hat{V}_{\rm trap}\rangle$, interaction energy $\langle\hat{U}\rangle$, and density profile, $\langle\hat{n}_i\rangle$ for each site i [ignore the (repeating) site index in parentheses].

Note that this demo run is deterministic. Why?

Now let us do the calculation with the exact-diagonalization code. The input file is also optionally supplied as the first argument to the program `bh1d-exact.exe`. Remember, its format is different from that of the QMC code and they should not be interchanged. Use exact diagonalization to verify your QMC result.

3.2 Comparing AFQMC and exact diagonalization results

Now let us turn on interaction. We will use a 5-site, 2-boson system ($t = 1, U = -0.5, \kappa = 0.3$) to verify that the AFQMC code produces the correct result. The input files are in the subdirectory `examples/smr1929/exer02.5s2p/`:

- `5s2p-trap.exact.in` for the exact diagonalization.
- `5s2p-trap.eqlb.in` for the AFQMC code.

Run the QMC and observe how the energy behaves as a function of imaginary time. The output of the measurement phase looks like:

```
#MEASURING: 1000 blks of 20 stps each
#        tau NWlk          TotalWeight              Block_E              Grand_E
#          0  200       200.0000000000      -3.594855257190      -3.594855257190
         0.1  200       395.3096831156      -3.627098053427      -3.627098053427
         0.2  200       393.3917176540      -3.661203625001      -3.644109370219
         0.3  200       392.6648319479      -3.692703137296      -3.660261062095
         0.4  200       392.5707321653      -3.709587035175      -3.672563926694
         0.5  199       384.2445828298      -3.718559084337      -3.681589336101
...
```

The `Block_E` column contains the *block measurement* The `Grand_E` contains the running average of the measurements. The first line (at imaginary time $\tau = 0$) contains the trial energy ($E_{\rm T}$), which in this case is equal to the variational energy of the trial wave function.

1. Plot the `Block_E` energy vs. `tau` using your favorite plotting program, and explain its behavior. How much imaginary time is needed to equilibrate the QMC? This interval, $\beta_{\rm eq}$, corresponds to the input `NEqSteps * dt` in `bh1d`. Now discard the "equilibration" part of the data, and do statistical analysis on the "measurement" phase: determine the minimum length of each measurement block (`NBlkSteps`) that is necessary to obtain uncorrelated results (and a reliable estimate of the statistical error).

2. Now modify the previous input file using the correct `NEqSteps` and `NBlkSteps` for a "standard" QMC run (i.e. equilibration $\rightarrow$ growth $\rightarrow$ measurement). Compare the energies (kinetic, trap, interaction, and total energies) with the exact result. Explain any agreements and disagreements.

3.3 Controlling the basic run parameters

Let us now run the code with a different set of input parameters, given in subdirectory `exer03.13s3p`:

1. Run the QMC code with the input file provided but for at least 3 different values of $\Delta\tau$ (e.g., 0.01, 0.04, 0.08), and examine the convergence of computed quantities (e.g., the total energy) as a function of $\Delta\tau$. Note that the block lengths (`NEqSteps`, `NAdjustSteps`, `NBlkSteps`, `itvMeasure`, and optionally `itvPopCtl`), should be adjusted to obtain comparable statistics.

2. Run the QMC code with the input file provided but for several values of `Nwalkers` (e.g., 10, 20, 40, 80, 160, 320) and examine how the *population bias* (systematic error vs. the population size). Note that with fewer walkers, you will need to run more blocks to get comparable statistical accuracy.

3. For observables other than the energy, better results can be obtained using brute-force estimate (use "`brute-force 1`" in the input file). Make sure your population size is big enough (e.g. 500). What happens to the measured energies if the population size is too small? Compare the `Block_E` and `Brute_E` columns in the output file (they are the mixed and brute-force estimators, respectively). What is the difference between them?

4. There are four possible trial wave functions in the input file. Can you find out experimentally which one is the best? How does the trial wave function affects the *statistical accuracy* of your QMC results? How about the *systematic accuracy*?

3.4 Study of density profiles

Do a larger system and study the effect of interactions, with the trap turned on. For example, a reasonable set of parameters is $L = 21$, $N = 10$; $t = 1$, $U = -0.2$, and $\kappa = 0.025$. You should turn on the brute-force estimate. Study the system's behavior as the interaction strength is changed, for example, $U = 0, -0.01, -0.5, -1.0$. Plot the density profiles and compare them. Also, examine how the energies change. Do the results make sense?

Now change the number of particles while keeping the other parameters the same. What happens to the density? Explain. (Note that the time step $\Delta\tau$ needs to be reduced at large $|U|$ and/or large N. Why?)

3.5 The physics of correlation

Let us study the system of two interacting bosons in a box, $N = 2$ with $L = 2$ sites. We will label the two sites $i = 1, 2$. We set $t = 1$ and change U: $= 0, -1, -2, -3, -4$ (where $\kappa = 0$ by setting `kappa` parameter to `1e-200`). You should turn on the so-called brute-force estimate.

1. As U increases, what should the density profile look like?

2. What should be the 2-body correlation function $\langle \hat{n}_1 \hat{n}_i \rangle$ do (where $i = 1, 2$)? Note that $\langle \hat{n}_1 \hat{n}_1 \rangle$ can be derived from the potential energy, which the code calculates. The code does not calculate $\langle \hat{n}_1 \hat{n}_2 \rangle$, but devise a way to derive it based on results of your run. Do your results make sense?

3.6 Additional Exercises

Optional exercises which are a bit more challenging:

1. *Changing the trap potential.* How do we do a problem with a quartic trap potential x^4 instead of the quadratic one? We will need to change the source files and recompile. Look at the file `bh1d_engine.cpp`, modify the subroutine `BosonAFMC::SetupK()`, and then recompile the code:

   ```
   $ make qmc      or      $ make qmc optimize=1
   ```

   This would generate `bh1d.exe`, the QMC code.

   Run this code and compare your findings with previous results. Modify the exact diagonalization code correspondingly and make comparisons with QMC.

2. *Modify the code to do repulsive interactions* ($U > 0$). Note the appearance of complex fields. (Hint: the first modification is to change the `Scalar` datatype into complex number; this is accomplished by adding `-D__opt_complex_scalar__` in the compilation command in the makefile.) Repeat the "equilibration" exercise (see Sec. 3.2) and study the behavior of the total energy as a function of projection time.

3. *Put in importance sampling.* See the hybrid method described in the references. The phase problem seen above can be dealt with once the importance sampling is put in, using the procedure discussed in the lectures.

APPENDICES

A Auxiliary-Field Quantum Monte Carlo for Bosons

The phaseless auxiliary-field quantum Monte Carlo method that you learned in the class is applicable to both boson and fermion systems. In particular, we still employ the ground-state projector $e^{-\Delta\tau(\hat{H}-E_{\rm T})}$ to obtain the ground-state wave function from a trial wave function $|\Psi_{\rm T}\rangle$. We approximate $e^{-\Delta\tau\hat{H}}$ with the Trotter decomposition $e^{-\frac{1}{2}\Delta\tau\hat{K}}e^{-\Delta\tau\hat{U}}e^{-\frac{1}{2}\Delta\tau\hat{K}}$, and use the Hubbard-Stratonovich transformation to evaluate $e^{-\Delta\tau\hat{U}}$.

Summarizing Ref. [1], the key features in the actual calculation of an N-particle boson system are:

1. Many-body wave functions are expressed in terms of permanents instead of determinants.

2. For each permanent (call it $|\Phi\rangle$), all the N bosons can occupy the same spatial orbital (no Pauli principle):

$$|\Phi\rangle = \hat{\phi}^\dagger\hat{\phi}^\dagger\ldots\hat{\phi}^\dagger|0\rangle\,, \tag{5}$$

   where $\hat{\phi}^\dagger$ is a boson creation operator for the one-particle orbital ϕ. As a result, the matrix representation is much simpler than the fermion case—a permanent can be represented by a single column vector

$$\boldsymbol{\phi} \equiv \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_N \end{pmatrix}. \tag{6}$$

3. The necessary overlaps can be computed easily:

$$\langle\Psi|\Phi\rangle = N!(\boldsymbol{\psi}^\dagger\cdot\boldsymbol{\phi})^N\,, \tag{7a}$$

$$\langle\Psi|c_i^\dagger c_j|\Phi\rangle = N!\,N(\psi_i^*\phi_j)(\boldsymbol{\psi}^\dagger\cdot\boldsymbol{\phi})^{N-1}\,, \tag{7b}$$

$$\langle\Psi|c_i^\dagger c_j^\dagger c_k c_l|\Phi\rangle = N!\,N(N-1)\psi_i^*\psi_j^*\phi_k\phi_l(\boldsymbol{\psi}^\dagger\cdot\boldsymbol{\phi})^{N-2}\,. \tag{7c}$$

B BH1D Input File Syntax

This section describes the input file syntax of the `bh1d` code. The input file is line-oriented, with at most one command per line. The first word is a keyword, followed by zero or more arguments. All arguments must be present in the same line as the keyword (i.e., no line continuation).

In the specification below, arguments enclosed by angle brackets ⟨ ⟩ are mandatory, and those in square brackets [ ] are optional. For commands with optional arguments, at least one (or more) value must be present. In either case, we can always skip an argument and not change its current value. To do this, put an asterisk (`*`) in the place of the argument. The asterisk can also be used to "comment out" the argument if it is the first character of the argument, and not separated by a space. For example, in the following command, the *Nsites* parameter is commented out (therefore disregarded):

```
system-size *15 3
```

Here is the list of valid commands accepted by `BH1D`'s input reader:

- `end`
  Terminates the script reading and ends the program.

- `Nsides` ⟨*Nsides*⟩
  `Nsites` ⟨*Nsides*⟩
  Determines the number of sites per linear dimension of the lattice. In 1-D Nsides=Nsites.

- `Nparticles` ⟨*Nparticles*⟩
  Determines the number of particles in lattice.

- `system-size` [*Nsites*] [*Nparticles*]
  Determine the size of the system at once. This is a shortcut for the `Nsites` and `Nparticles` commands.

- `NWalkers` ⟨*NWalkers*⟩
  Determines the initial number of walkers in simulation, which is also the average number of walkers we want to maintain throughout the simulation.

- `t` ⟨t⟩
  The kinetic matrix element in the Hubbard Hamiltonian—see Eq. (1). By default this is set to 1, and typically there is no need to change this quantity.

- `U` ⟨U⟩
  The potential matrix element in the Hubbard Hamiltonian. It must be negative for this program.

- `dt` ⟨$\Delta\tau$⟩
  The discrete timestep.

- `ET` ⟨E_{T}⟩
  The initial trial energy. A special keyword `auto` (or `trial_wf`) can be used in place of E_{T} parameter above, so that the trial energy is automatically computed based on the given trial wave function.

- `NEqSteps` ⟨*NEqSteps*⟩
  The number of steps taken to equilibrate at the beginning of the simulation.

- `NAdjustSteps` ⟨*NAdjustSteps*⟩
  The number of steps taken in each trial-energy adjustment period.

- `NAdjustCount` ⟨*NAdjustCount*⟩
  The number of times the trial-energy adjustment is performed.

- `NBlocks` ⟨*NBlocks*⟩
  The number of blocks used in the measurement phase.

- `NBlkSteps` ⟨*NBlkSteps*⟩
  The number of random-walk steps in each measurement block.

- `itvMeasure` ⟨*itvMeasure*⟩
  The interval between two adjacent observable measurement. To measure at every step, set this quantity to 1.

- `itvPopCtl` ⟨*itvPopCtl*⟩
  The interval between two adjacent population control.

- `psi_T` ⟨$\psi_1 \quad \psi_2 \quad \ldots \quad \psi_{\mathrm{Nsites}}$⟩
  `psi_T noninteracting`
  The trial wave function Ψ_{T} (a permanent). In the first format, the trial wave function is input as an array of numbers $\psi_1 \ldots \psi_{\mathrm{Nsites}}$. The numbers must be listed horizontally on the same line, and there must be exactly `Nsites` numbers altogether. In the second format, a special keyword is used to denote that request automatic calculation of the trial wave function from the noninteracting (one-body) Hamiltonian. Note that in either case the trial wave function will be automatically normalized at the beginning of the QMC calculation.

- `run`
  Executes the calculation with current parameters.

- `brute-force` ⟨*integer*⟩
  A switch to turn on the bruteforce measurement. A nonzero value turns it on, while zero value will turn it off. By default this option is not enabled.

- `popctl-scaling`
  Uses the *scaling* method for the population control if the number of walkers lies outside the allowed range.

- `popctl-simple-combing`
  Uses the *simple combing* method for the population control if the number of walkers lies outside the allowed range. This is the default setting.

Comment lines can be inserted in the input file by prepending them with the pound '`#`' character. Anything from the pound character through the end of the line will be discarded. Comments can be put anywhere in the input file, whether in a separate line or at the end of a command. These sample lines show the possible ways of putting comment texts:

```
# This is a sample QMC input file
NParticles 50 # We use fifty particles
```

A copy of the input file is kept in the output file with all comments stripped.

The QMC program can perform more than one calculations, executed sequentially, through a single input file. Nonetheless it is recommended to specify only one calculation per input file.

C Supporting Files

Here is a list of the support files contained in this package.

- Directory `examples/smr1929/` contains test input files to play with in this lab session.

- Directory `testcase/` contains a few test input files to check the (future) program against regressions. Two sample inputs are supplied in this directory: `test-01.in` and `T1D-13s3p.Ux1.0.in`. The former basically just exercises the "demo" run, but records the output to a file. The latter is a "real" sample input: 3 particles on 13 sites, with a nonvanishing trap potential.

- File `doc/COMPILING.txt`: the advanced instruction on compiling the programs. Usually you won't need this unless you are porting the code to platforms other than Linux.

- `doc/lab_AFQMC.pdf`: the electronic version of this handout.

D Contact Address

You are welcome to contact the authors if you have questions regarding this package. Our email addresses are `wxpurw@wm.edu` and `shiwei@physics.wm.edu`.

References

[1] Wirawan Purwanto and Shiwei Zhang, "Quantum Monte Carlo method for the ground state of many-boson systems", Phys. Rev. **E70**, 056702 (2004), preprint version: `http://arxiv.org/abs/physics/0403146`.

[2] Wirawan Purwanto and Shiwei Zhang, "Correlation effects in the ground state of trapped atomic Bose gases", Phys. Rev. **B72**, 053610 (2005), preprint version: `http://arxiv.org/abs/cond-mat/0506219`.