united nations
educational, scientific
and cultural
organization

international atomic
energy agency

the
abdus salam
international centre for theoretical physics

ICTP 40th Anniversary

SMR 1595 - 18

Joint DEMOCRITOS - ICTP School on

CONTINUUM QUANTUM MONTE CARLO METHODS

12 - 23 January 2004

COMPUTER LABORATORY SESSION

LAB ON AUXILIARY-FIELD QUANTUM MONTE CARLO

Wirawan PURWANTO, Wissam AL-SAIDI, Malliga SUEWATTANA and Shiwei ZHANG
College of William and Mary,
Dept. of Physics
Williamsburge, VA, USA

These are preliminary lecture notes, intended only for distribution to participants.

Joint DEMOCRITOS-ICTP School on Continuum Quantum Monte Carlo Methods
ICTP Centre, Trieste, Italy, 12–23 January 2004

Lab on Auxiliary-field Quantum Monte Carlo:
Bosonic Alkali atoms in a 1-D trap

Wirawan Purwanto, Wissam Al-Saidi, Malliga Suewattana, and Shiwei Zhang

January 19, 2004

Abstract

We describe BH1D, a pedagogical program to learn the auxiliary-field quantum Monte Carlo method. This program calculates the ground-state properties of a boson gas confined in one-dimensional trap. This handout was specifically written as a roadmap to the lab session at the "Joint DEMOCRITOS-ICTP School on Continuum Quantum Monte Carlo Methods", 12–23 January 2004.

1 Definition of the Problem

Consider a one-dimensional lattice containing L points, as shown in Fig. 1. On these lattice sites we put N identical spinless bosons. This system is described by the following Hamiltonian:

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

$$\begin{aligned} &= -t \sum_{i=1}^{L-1} (c_i^\dagger c_{i+1} + c_{i+1}^\dagger c_i) - \delta_{\kappa,0} t (c_L^\dagger c_1 + c_1^\dagger c_L) + \tfrac{1}{2}\kappa \sum_i |\mathbf{r}_i - \mathbf{r}_0|^2 \hat{n}_i \\ &+ \tfrac{1}{2} U \sum_{i=1}^{L} \hat{n}_i \hat{n}_i - \tfrac{1}{2} U \sum_{i=1}^{L} \hat{n}_i \,. \end{aligned} \tag{2}$$

where $\hat{n}_i \equiv c_i^\dagger c_i$ as usual, $\sum_{\langle i,j \rangle}$ denotes the sum over nearest-neighboring pair of sites, and $|\mathbf{r}_i - \mathbf{r}_0|$ is the (radial) distance of the site point i to the trap center. The factor $\frac{1}{2}$ in the interaction term is needed to compensate the double-counting. The question is: *What are the ground-state properties of this system*? We will use auxiliary field quantum Monte Carlo to answer this question. Let's now get into the program package that solves this problem.


```
    o----o----o----o . . . o----o- - *
i = 1    2    3    4      L-1   L    1 again
                                    (only if
                                   p.b.c. is on)
```

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

1.1 Method

We use the ground-state projector $e^{-\Delta\tau(\hat{H}-E_T)}$ to compute the ground-state wave function, where $e^{-\Delta\tau\hat{H}}$ is approximated by the Trotter breakup $e^{-\frac{1}{2}\Delta\tau\hat{K}} e^{-\Delta\tau\hat{U}} e^{-\frac{1}{2}\Delta\tau\hat{K}}$, and $e^{-\Delta\tau\hat{U}}$ is evaluated using the Hubbard-Stratonovich transformation. The theoretical ground of this calculation is available as a *draft* from the authors (WP or SZ).

2 Overview of the BH1D Program

2.1 Installing the code

An automated install procedure has been provided. Execute the file[1]

```
/afs/ictp/public/w/wpurwant/Boson-Hubbard-setup
```

This will create a subdirectory `bh1d/` in your home directory, which contains the source code, the make-file, and subdirectory `Exec/` which have links to the precompiled executables. The setup file will also compile the QMC executable code (`bh1d.exe`) and the exact-diagonalization code (`bh1d-exact.exe`).

If the compilation runs okay, the executable `bh1d.exe` and `bh1d-exact.exe` will appear in `bh1d/` directory. If this is not the case, please use in the mean time the executables in the directory `bh1d/Exec/`.

2.2 Computation Process

The QMC calculation runs as follows: the random walkers are initialized to $|\Psi_T\rangle$ at the beginning of the simulation. The first phase is the *equilibration* phase consisting of `NEqSteps` random-walk steps. This is followed by the *growth* phase to obtain the best estimate of the trial energy E_T. Then comes the *measurement* phase, which contains `NBlocks` blocks of `NBlkSteps` Monte Carlo steps each. At each block we carry out (`NBlkSteps/itvMeasure`) measurements, and average their results. This average is termed the *block measurement*. The actual statistics is computed at the end of the measurement phase by averaging the block measurements. This result is printed in the output file in the `BLK_AVG` and `ERROR` columns, as will be explained in Sec. 3.1.

We use the *mixed-estimators* to compute the ground-state observables:

$$\langle A \rangle_{\text{mixed}} = \frac{\langle \Psi_T | \hat{A} | \Phi_0 \rangle}{\langle \Psi_T | \Phi_0 \rangle} \doteq \frac{\sum_i w_i \langle \Psi_T | \hat{A} | \phi_i \rangle}{\sum_i w_i \langle \Psi_T | \phi_i \rangle}$$

[1] The complete lab package is located in a compressed tar file: `/afs/ictp/public/w/wpurwant/bh1d.tar.bz2`

An exact—at least in principle—estimate is the *brute-force* estimate, obtained by replacing the trial wave function $\langle\Psi_T|$ above by the Monte Carlo samples of the ground-state wave function, $\sum_j\langle\phi_j|$. We will explore its behavior in the exercises.

2.3 BH1D Input File Syntax

The impatient can go right away to the next section, and go back to this section as necessary. The input file is relatively straightforward. It is line-oriented, one command per line. The first word is a keyword (see the sample input file for a few examples), followed by zero or more arguments. It should be easy for user to change and experiment with. All arguments must be present in the same line as the keyword. Those marked with 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 returns to the operating system (shell).

- `Nsides` ⟨*Nsides*⟩ or `Nsites` ⟨*Nsides*⟩
  Determines the number of sites per linear dimension of the Hubbard lattice. In 1-D this is equal to the total number of sites.

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

- `system-size` [*Nsites*] [*Nparticles*]
  Determine the size of the Hubbard 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 word `auto` 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` ⟨ψ_1 ψ_2 ... ψ_{Nsites}⟩
  The trial wave function Ψ_T (an permanent), which is input as an array of numbers $\psi_1 \ldots \psi_{\text{Nsites}}$. The numbers must be listed horizontally on the same line, and there must be exactly `Nsites` numbers altogether. Note that this wave function will be automatically normalized at the beginning of the QMC calculation.

- `run`
  Executes the calculation with current parameters.

- `brute-force` ⟨*0—1*⟩
  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 prefixing 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 result file with all comments stripped.

The QMC program can perform more than one calculations (executed sequentially) through a single input file. Nonetheless it is best to write input files that do only one calculation per file. Each output file will have only one set of result.

2.4 Exact-Diagonalization Input File Syntax

The input file of 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 of a valid input file:

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

Comments, if any, must come on the second line onward. Prefix them with the character '#', as before. Try running the 1-D exact-diagonalization program with this input file. It should give a ground-state energy of -17.50903835. Beware not to set too large `Nsites` and `Nparticles`. The computing time and memory grow exponentially with the system sizes.

3 Exercises

3.1 Running the AFQMC and exact diagonalization programs

The first assignment is to run a "demo" run which needs no input file. The aim is to get a feeling of the code, to understand the output file, and to compare the AFQMC results with those from an exact-diagonalization calculation.

First let us start with the AFQMC run. It would go like this:

```
$ ./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 0.12.5

SAMPLE PROGRAM SESSION (use --help switch 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
```

...and so on. The results are not recorded into a file, but all that we care about the result printed at the end of the output, like this:

```
[-] 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: the first column (`BLK_AVG`) is the average of the block measurements, and its corresponding error estimate is given in the `ERROR` column. The second columns gives a *grand average*. This is a slightly different method of doing average. `BLK_AVG` and `GRAND` should agree within one or two errorbars (except for the 'verlap"). Otherwise, there may be a problem with the QMC run. The third column gives the expectation value of the observables with respect to the trial wave function Ψ_{T}. There can be two more result columns if the `brute-force` option is enabled: they are the brute-force averages and their standard errors. The averaging procedure is similar to the `BLK_AVG`.

There are five quantities measured: the total energy, kinetic energy $\langle\hat{T}\rangle$, trap energy $\langle\hat{V}_{\mathrm{trap}}\rangle$, interaction energy $\langle\hat{U}\rangle$, and density profile, $\langle\hat{n}_i\rangle$ for each site i. For 1D, the number in parentheses repeats the first number to the left in the density profile which is the site index (they are zero-based).

Question: The demo run is deterministic. Why is this?

Now let us redo 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, the format of the input file is very different from that of the QMC code and they should not be interchanged. Run the program without parameters:

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

Unlike the QMC program, the output is printed only to stdout (so you may want to redirect the output to a file), e.g.

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

3.2 Controlling the basic run parameters (T1D_3.2.in)

Let us now run the code with different input parameters as supplied with the input file. To run the input file, supply the input filename as the first argument to the program, e.g.

```
$ ./bh1d.exe examples/smr1595/T1D_4.2.in
```

For small systems (13 sites or fewer), you can compare the QMC results against the exact-diagonalization answer. Here are the exercises:

- 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 $\Delta\tau$. You must also adjust the block lengths (`NEqSteps`, `NAdjustSteps`, `NBlkSteps`, `itvMeasure`, and [optionally] `itvPopCtl`) so that the total simulation time τ is constant.
- 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 systematic error depends on the population size.
- Configure the run to see the convergence of the total energy as a function of projection time $\tau = n\ \Delta\tau$ for one set of parameters. Plot E vs τ. *Hint:* use `NEqSteps` = 0 and `NAdjustSteps` = 0.

3.3 The physics of correlations (T1D_3.3.in)

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

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

3.4 Study of density profiles (T1D_3.4.in)

Do a larger system and study the effect of interactions, with the trap turned on. For example, $L = 21$, $N = 10$, and $t = 1$. You should turn on the so-called brute-force estimator. Study the system's behavior as the interaction is increased (for example $U = -1, -0.5, 0$). Plot the density profiles and compare. Also, examine how the energies change. Do the results make sense?

3.5 Optional Exercises

- Changing the trapping potential: In this assignment, the user will change the source files, recompile, and run. The aim is to use a quartic potential x^4 instead of a quadratic one x^2 as already implemented in the code. This can be done by looking at the file `boson-hubbard1d.cpp` and changing the subroutine `BosonAFMC::SetupK()`.

  Now one has to compile the code. There is nothing to configure in order to (re)compile the code. It should simply run, like this:


```
$ make qmc
```

or if you want the optimized code (which runs quite faster), use

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

instead. 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.

- Put in importance sampling.
- Modify the code to do repulsive interactions ($U > 0$). Note the appearance of complex fields. Repeat the exercise in bullet 3 in Sec. 3.2 and study the behavior of the total energy as a function of projection time. The phase problem can be dealt with once the importance sampling is put in, using the procedure discussed in this morning's lecture.

4 Supporting Files

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

- Directory `examples/smr1595/` 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. Please disregard this subdirectory, as your relevant input files are supplied in the subdirectory `examples/smr1595/`.
- 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-smr1595.tex`: the source manuscript of this handout.

5 Contact Address

You are welcome to contact the authors if you have questions regarding this package. Email addresses `wirawan@camelot.physics.wm.edu` and `shiwei@physics.wm.edu`.