Lattice Monte-Carlo simulation of the quantum mechanical path integral

Instructions

Joshua Berin, Marc Wagner
berlin@th.physik.uni-frankfurt.de, mwagner@th.physik.uni-frankfurt.de

October 2, 2014

1 Introduction

- Study the introduction *MC_introduction.pdf* to the quantum mechanical path integral and to Monte-Carlo simulations.
- Study the provided C code `MC_QM.C`.

2 Simulation of the harmonic oscillator

2.1 First run

Check whether the following parameters in the code are defined as follows:

- `m = 0.5`
- `omega = 0.75`
- `numSweepsOut = 10`
- `numSweepsThermalization = 1000`
- `numSweeps = 5000`
- `N = 64`
- `delta_t = 1.0`
- `delta_q = 3.0`
- `PathNumber = 0`
- `qMaxStart = 5.0`

If not, define then accordingly. Compile and run the code, i.e. perform a first simulation.

2.2 The action S and thermalization

Plot $\langle S \rangle$ (the action averaged over `numSweepsOut`) for a hot and a cold start for `delta_q = {0.003, 0.03, 0.3, 3.0, 30.0, 300.0}`.

- Are hot or cold starts preferable?
- Which is the best choice for `delta_q`?
- After approximately how many steps did the simulation thermalize in each of these twelve cases?

Crudely optimize `delta_q` for the parameters defined in **??**. A heuristic criterion is that around 50% of all Metropolis updates are accepted.

Now change `omega` to `omega = 0.125`. Is the previously optimized `delta_q` still optimal? If not, optimize again.

2.3 Investigation of paths generated during the simulation

Plot several arbitrary paths after thermalization taken from simulations at `omega = {0.75, 0.125, 5.0}`. Explain and interpret your observation in particular the qualitative differences.

Plot the first ten paths of a simulation to study thermalization in more detail. To this end perform a simulation with a cold start and `delta_q = 5.0`.

2.4 Computation of correlation functions and extraction of energy differences

Return to `omega = 0.75` and the corresponding optimized `delta_q`.

- Use the code to compute the correlation function $C(t) = \langle x(t)x(0) \rangle$.
  - Study the details of the implementation. In particular explain how $\langle x(t)x(0) \rangle$ is implemented.
- Fit $A \exp(-\Delta E t)$ to the numerical results for $C(t)$ (fitting parameters $A, \Delta E$) to extract the energy difference of the first excited state and the ground state.
  - Why is it dangerous to include small t in the fit in general?
  - Why is it not a problem here?
  - Why is it dangerous to include large t in the fit?
  - Specify and use a reasonable fit window to extract ΔE and a corresponding statistical error.
  - Compare with your analytical expectation.

- Perform another two simulations with `numSweepsOut = 100`, `numSweeps = 50000` and `numSweepsOut = 400`, `numSweeps = 200000` (such that $C(t)$ its evaluated again on 500 paths).
  - Extract ΔE again for both cases.
  - Compare the three results for ΔE, in particular the associated statistical errors. Interpret your observation.
- Vary `numSweeps` (keeping `numSweepsOut` fixed) at least by a factor of 10 and investigate the behavior of the error of the correlation function. How does the error behave as a function of `numSweeps`?

3 Simulation of the anharmonic oscillator

Now that you are an expert in lattice MC simulation, study another problem, which cannot be solved analytically. Change the potential $V(x)$ in `void S_E()` and `double DeltaS_E(int index, double q_new)` to $V(x) = \frac{m\omega^2}{2}x^2 + \lambda x^4$.

Of course the results you will obtain for ΔE do not only depend on ω but also on λ.

- Compute ΔE for `m = 0.5, omega = 0.75` and `lambda = { 0.001, 0.002, 0.004, 0.006, 0.01, 0.02, 0.1 }`.
- Use standard textbook perturbation theory to check your code modification and results for small λ.