Quantum Mechanics on the Lattice

Gary F. Sinclair
University of St Andrews
St Andrews, Scotland
Email: gfs1@st-andrews.ac.uk

This report presents some basic theory of Feynman's path integral formulation and the lattice approach to quantum mechanics. The theory is illustrated by a Markov Chain Monte Carlo simulation of the Simple Harmonic Oscillator using the Metropolis Algorithm.

1 Introduction

1.1 The Kernel or Green's Function

A central problem in Quantum Mechanics is the calculation of the time evolution of a wavefunction given its initial distribution. In the Schrödinger Picture of Quantum Mechanics this is simply done by solving the well known Schrödinger Equation. A very convenient method of solution involves the application of Green's functions, which describes how an arbitrary initial wavefunction will evolve. The Green's function describes how the region of influence of each point of the initial wavefunction spreads to affect other points at a later time. The final wavefunction is therefore just the integral of the Kernel with each point of the initial wavefunction.

$$\psi(x,t)=\int K(x_f,x_i,t)\psi(x_i,0)dx_i \qquad (1)$$

The Green's Function can be shown to be a solution of the Schrödinger equation, subject to the initial condition

$$K(x_f,x_i,0)=\delta(x_f-x_i) \qquad (2)$$

Solving this equation using the standard methods of differential equations can be a difficult and time consuming process. Instead it is beneficial to investigate a quicker and more physically insightful method to find the Green's Function: that of Feynman Path Integrals.

1.2 Feynman Path Integrals

The Feynman Path Integral approach to Quantum Mechanics is based on the idea that a particle can explore an infinite number of paths through space between two chosen fixed points. The actual probability of a particle arriving at the destination point is dictated by the interference of all of the paths with each other that satisfy the fixed point boundary conditions. Let us introduce Dirac notation for convenience, and denote the transition probability between two position eigenstates (in other words, the Kernel $K(x,x',t)$) as,

$$\langle x_f|e^{-H(t_f-t_i)/\hbar}|x_i\rangle=\int Dx(t)e^{-S(x)/\hbar}. \qquad (3)$$

where

$$S(x)=\int_{t1}^{t2}dtL(x,\dot{x})=\int_{t1}^{t2}dt\frac{m\dot{x}(t)^2}{2}+V(x(t)) \qquad (4)$$

The right hand side of eq.(3) denotes an integral over all paths in space between our two fixed points. We have chosen to introduce an imaginary time scale to transform the oscillatory exponential to a real decaying function of the action, S(x). This important trick makes the visualisation of the problem much simpler and numerical simulations possible.

To facilitate the elucidation of the important physical properties of the wavefunction evolution, let us consider a particle performing a periodic motion such that:

$$x_i-x_f=x \qquad (5)$$
$$t_f-t_i=T \qquad (6)$$

On expanding the right hand side in terms of an energy eigenfunction basis, the kernel can be written as:

$$\langle x|e^{-HT/\hbar}|x\rangle = \sum_n \langle x|E_n\rangle e^{-E_nT/\hbar}\langle E_n|x\rangle \quad (7)$$

By using this expansion in terms of energy eigenvectors, we decompose our initial position eigenstates (Dirac delta functions) into components that have a particularly simple time evolution (only a phase change). From this equation it is clear to see that only the component of the initial state in the eigensubspace of the ground state projector will survive as the evolution time T becomes large.

$$\lim_{T\to\infty} \sum_n \langle x|E_n\rangle e^{-E_nT/\hbar}\langle E_n|x\rangle \to e^{-E_0T/\hbar}|\langle x|E_0\rangle|^2 \quad (8)$$

Thus, as T becomes large, the wavefunction will evolve into and remain in the ground state. Our problem of finding the ground state of the system therefore can be solved if we can find a way to calculate the kernel using path integrals. For this to be done numerically, we must discretise the problem using a lattice approximation of the path.

1.3 The Lattice Approximation

The fundamental change in the lattice approximation is the transition from a time continuum to a set of discrete time lattice points. At each point t_n in time we have an associated point in space x_n. Together the set of points $\{x_n\}$ describe the configuration of the system, or the path of the particle. It is important to highlight that the time lattice points must lie within the finite range [0, T], but the space lattice points can describe any path in space and therefore lie in the range $(-\infty, \infty)$. Using this discretisation scheme, we rewrite the action over a path as:

$$S_{lat}[x] = \sum_{j=1}^{N} a\left(\frac{m(x_{j+1}-x_j)^2}{2a} + V(x_j)\right) \quad (9)$$

and therefore the kernel can be written as:

$$K(x_f, x_i, t)_{lat} = \int_{x_1=x_i}^{x_N=x_f} e^{-S_{lat}[x]/\hbar} \prod_{j=1}^{N} \frac{dx_j}{A} \quad (10)$$

where A can be considered a normalisation factor, and the integral is taken over all N of the x_n co-ordinates. During analytical calculations, the time step a, can be taken to zero for the continuum limit.

We now have a suitable discretised integral for the kernel over all paths in space and must turn to the practicalities of evaluating such a sum. It is important to note that most random paths will produce vanishingly small contributions to the integral, and therefore a simple random Monte Carlo evaluation of the integral would be impossible. Rather than randomly selecting path points, we must use the method of importance sampling.

1.4 Importance Sampling and Statistical Mechanics

The key to selecting points for integral evaluation comes from the similarity between the imaginary time formulation of our path integral and the partition function of the microcanonical ensemble. One can see from eq.(10) that the kernel is of the same form as the 'Zustandssumme':

$$Z_{fi} = \langle x_f|e^{-HT/\hbar}|x_i\rangle \quad (11)$$

Thus, the likelihood that a given path occurs will obey Boltzmann statistics. Here we have $\hbar$ playing a similar role to the temperature T. The similarity is striking: as $\hbar$ tends to zero we expect classical physics to emerge, so that only the path of least action contributes to the kernel. In statistical physics the equivalent statement is that statistical fluctuations are removed as the temperature tends to zero. In both cases fluctuation in path or observable are suppressed as the corresponding quantity tends to zero. Therefore to make most efficient use of our simulation it is desirous to select path points in accordance with the Boltzmann distribution - this is the basis of importance sampling.

1.5 Markov Chains and the Metropolis Algorithm

Before continuing to a description of the Metropolis Algorithm for selecting points, a brief review of Markov Chains must be given. A Markov Chain can be represented as a $N \times N$ matrix W_{ij} for which each element describes the transition probability from state i to state j.

The matrix W_{ij} must therefore satisfy the conditions:

$$\sum_j W_{ij} = 1 \tag{12}$$

$$W_{ij} \geq 0 \tag{13}$$

In the continuum limit we redefine W for our Markov step as the probability of a transition of a point from x to x', such that:

$$\int dx' W(x, x') = 1 \tag{14}$$

$$W(x, x') \geq 0 \tag{15}$$

As we would expect, transition probabilities can be composed so that:

$$W^2(x, x'') = \int dx' W(x'', x') W(x', x) \tag{16}$$

It can be shown that repeated application of Markov steps will lead an ensemble into a unique equilibrium state.

$$\lim_{n\to\infty} W^n(x', x) = P(x') \tag{17}$$

Applying a further Markov step, will of course leave the equilibrium ensemble unchanged. Thus W(x', x) acts as an identity on the distribution.

$$P(x') = \int dx W(x', x) P(x) dx \tag{18}$$

Our aim is to apply many Markov steps (a Markov Chain) so that our ensemble of paths reach a unique equilibrium state - the ground state. We must now find a function W(x, x') that satisfies the above conditions (14,15,18). A popular choice of substitution is called the detailed balance condition:

$$\frac{W(x, x')}{W(x', x)} = \frac{P(x')}{P(x)} \tag{19}$$

This still does not define W(x, x') uniquely, but we are closer. We can simplify matters by restricting ourselves to Markov processes where only a single path point is changed at a time. Assuming that we expect the probability P(x) to obey Boltzmann Statistics the detailed balance condition becomes

$$\frac{W(x, x')}{W(x', x)} = \frac{e^{-S[x_j']}}{e^{-S[x_j]}} \tag{20}$$

The most intuitive method to achieve this would be to use the so called "Heat Bath" algorithm, where the transition probability is chosen such that,

$$W_s(x_j, x_j') \propto e^{-S(x_j')} \tag{21}$$

However, this method is quite tricky to implement in practice, so instead an alternative algorithm shall be used: The Metropolis Algorithm. The algorithm is best described in words; first a new point is chosen randomly within some region surrounding the existing co-ordinate. If the substitution of this point reduces the action of the path, then the change is implemented. However, if substitution causes an increase in action, then a random number, r in the interval [0,1] is chosen. If $e^{-\Delta S} > r$ then the change is made. It can be shown that this process to satisfies the detailed balance condition given above. Although this process is less efficient per step than the Heat Bath algorithm, in practice it is often faster due to the computational simplicity and speed of each step.

2 Numerical Simulation

2.1 Harmonic Oscillator

To illustrate the numerical techniques described above, a programme was written to simulate the ground state of the simple harmonic oscillator, which is given by:

$$\psi(x) = \left(\frac{m\omega}{\pi\hbar}\right)^{1/4} e^{-m\omega x^2/2\hbar} \tag{22}$$

A slightly simplified version of the *main()* function, written in c++ is given below to give an overview of its structure:

```
main()
{
Paths newpaths(Options);
newpaths.Initialise();
newpaths.Thermalise();
newpaths.Metropolise();
newpaths.Distributions(true, false);
}
```

Firstly a class of object describing the paths is created, *newpaths* . This is then initialised to either a 'hot' start - a random distribution, or a 'cold' start - a straight line. This distribution is then thermalised, that is about 500 Monte Carlo Iterations (that is a Markov Step is done on each lattice point) are done on it to produce a reasonable starting path. Once a properly thermalised path is achieved further MCI's are done on the path, so that every five MCI's a new path is generated and stored. The iterations between path stores is done to decorrelate adjacent paths. Finally, the stored paths are analysed to produce position and energy density plots. An example of a position probability plot generated is given below, Fig.(1).

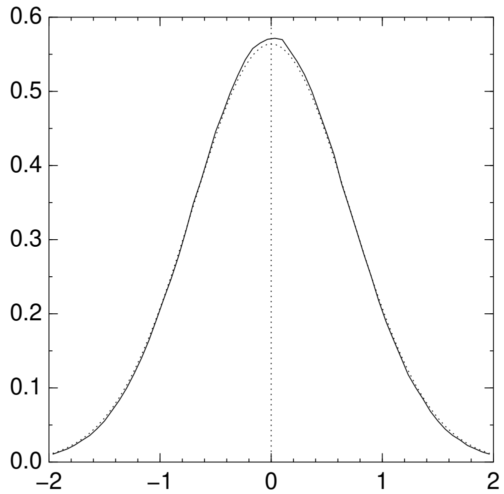

Fig. 1: *Position probability distribution of ground state for $\omega = 1$, m = 1*

The graph given above was calculated by generating 10'000 paths, each of 200 points over a simulated time interval of 100 seconds. The solid line shows the numerically generated ground state distribution, whereas the dashed line is the theoretical prediction. It can be seen that the agreement is good, even with relatively few paths. The position probability distribution is calculated by reading each point of each path into the analysis function which then counts the number of points in each data bin. These values are then normalised by dividing by the area under the distribution graph.

Similarly the same process can be done for more unusual potentials such as the anharmonic oscillator - a graph of which can be seen below in Fig(2). This was created using the same path parameters as for the harmonic oscillator above.

To maintain a good 'hit rate', that is the percentage of random point selections that are im-

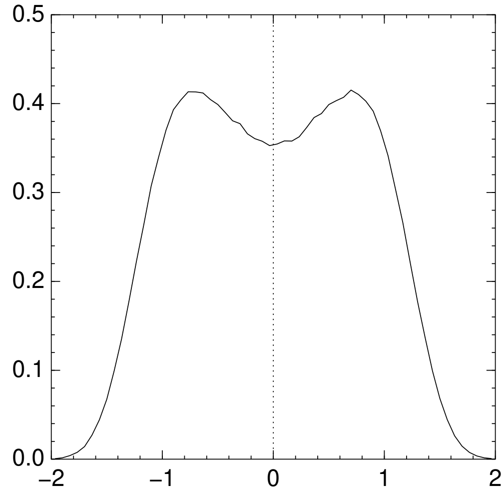

Fig. 2: *Position distribution for anharmnic oscillator $\omega^2 = -4$, m = 1*

plimented, the region about which new points are selected from was dynamically modified. A hit rate of 40 % was chosen as a good rate and the window adjusted to maintain this.

Graphs of the potential energy distribution are also shown below. These were produced by taking numerical integrals of the positon probability density multiplied by the potential at that point. As can be seen, the area under the graph is equal to 1/4, as would be expected from theory for our given parameters; that is half of the total energy given by $E_n = \omega(n = 1/2)$ ($\hbar$ is set to unity for convenience).

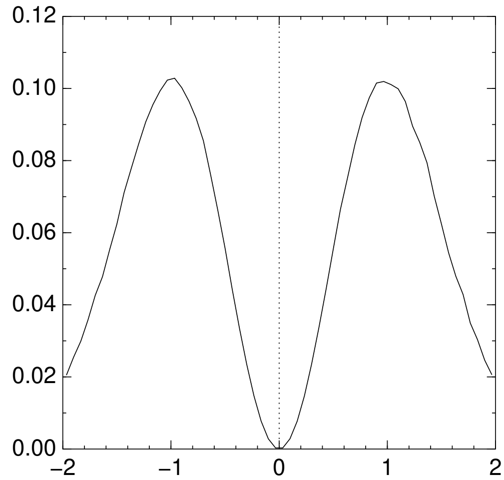

Fig. 3: *Energy distribution for harmnic oscillator $\omega^2 = 1$, m = 1*

The anharmonic oscillator potential distribution shows a more interesting behaviour than the simple harmonic. It can be seen that only a small region at the edges of the wavepacket are actually energetically bound (positive potential energy), whereas the majority of the interior is repulsed by the negative harmonic term (note $\omega^2 = -4$).

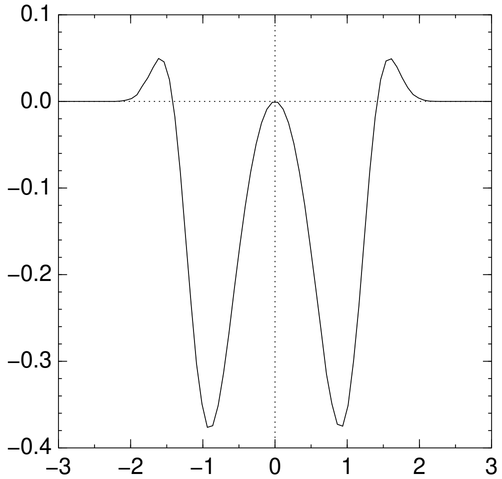

Fig. 4: *Energy distribution for anharmnic oscillator* $\omega^2 = -4$, *m = 1*

3 Acknowledgments

I would like to thank Dr Karl Jansen for his help and supervision during this summer project, and likewise all the other members of the NIC group who helped with the c++ programming. Furthermore my thanks go to Karlheinz Hiller for his supervision of the summer student programme.

References

[1] M. Creutz and B. Freedman, *A Statistical Approach to Quantum Mechanics*, Annals of Physics 132, 427-462, (1981)

[2] R.P. Feynman and A.R Hibbs, *Quantum Mechanics and Path Integrals*, (McGraw Hill).

[3] G.P Lepage, *Lattice QCD for Novices*, Newman Laboratory of Nuclear Studies, Cornell University, Ithaca