

Looking at the Harmonic Oscillator with Lattice Field theoretical methods

DESY Summer Student Programme, 2008

Alexandra Martín Sánchez

(Universidad de Salamanca, Spain)

Roman Welsing

(Fachbereich Physik, Freie Universität Berlin, Germany

École Normale Supérieure de Lyon, France)

Zeuthen (Berlin), September 11, 2008

Abstract

In this report we study the path integral as both a theoretical and numerical approach to Quantum Mechanics. We derive the formulation of the path integral on a discrete time lattice and we illustrate this with explicit calculations of the propagator for both the free particle and the Harmonic Oscillator. Furthermore, we use the discretised form of the path integral to perform numerical calculations of several observables for the Harmonic Oscillator using the Hybrid Monte-Carlo algorithm (HMC). We perform a detailed study of the autocorrelations, after which we find clear agreement between our numerical calculations and the exact results.

Contents

1	Introduction	3
2	Theoretical foundations	3
2.1	Path integrals in the continuum	3
2.2	Path integrals on the lattice	4
2.3	Euclidean path integral	4
3	Analytic examples	5
3.1	The Free Particle	5
3.2	The Harmonic Oscillator	6
4	Numerical methods	8
4.1	HMC algorithm	8
4.2	Simulation	10
5	Results	11
5.1	Autocorrelation effects	11
5.2	$\langle x^2 \rangle$ results	15
5.3	$\langle x(0)x(t) \rangle$ and ΔE	17
6	Conclusions	19
7	Acknowledgments	19
A	Calculations for the Free Particle path integral	21
B	Classical action for the Harmonic Oscillator	22
C	Calculations for the action on the lattice under Fourier transformations for the Harmonic Oscillator	22

1 Introduction

Since Richard Feynman as a young graduate student at Princeton University in 1948 first discovered the path integral method to calculate propagators¹ in Quantum Mechanics, it has turned out to be a very powerful tool for numerical calculations, in QED and QCD especially. Furthermore the path integral formalism has the advantage that it makes the symmetries of the considered problem manifest, as those symmetries are described by an invariance of the Lagrangian and the path integral formalism allows to bypass the construction of the Hamiltonian.²

Therefore in this report we will proceed as follows. First of all we will try to motivate the canonical form of the path integral. Then we will introduce the discretized form of the path integral in order to calculate the propagator for the Harmonic Oscillator using the method of Fourier transforms. In a second part we will present the Hybrid Monte-Carlo algorithm (HMC), a numerical method we have used to solve the path integral. We will show the results we have obtained for some observables, $\langle x^2 \rangle$ and the ground state energy, by performing our numerical simulations.

2 Theoretical foundations

2.1 Path integrals in the continuum

In Quantum Mechanics physical entities have an observable probability to be found in a given state at a given time. These states can be described by wavefunctions $\Psi(x, t)$ obeying the Schrödinger equation

$$H|\Psi(t)\rangle = i\hbar \frac{\partial}{\partial t} |\Psi(t)\rangle. \quad (1)$$

The solutions of this time-dependent Schrödinger equation can be factorized into a stationary part $|\Psi(0)\rangle$ fulfilling the stationary Schrödinger equation $H|\Psi\rangle = E|\Psi\rangle$, whereas the time evolution is described by the time evolution operator $U(t) = e^{-iHt/\hbar}$, with $|\Psi(t)\rangle = U(t)|\Psi(0)\rangle$.

The quantum mechanical amplitude for a particle to propagate from a position x_a at time t_a to a position x_b at time $t_b = t_a + T$ can be described in exactly the same manner, namely by considering the propagator:

$$K(x_b, x_a, T) = \langle x_b | e^{-iHT/\hbar} | x_a \rangle \quad (2)$$

The probability P for the particle to propagate from x_a to x_b is the absolute square of the amplitude $P(b, a) = |K(x_b, x_a, T)|^2$. As observed by Feynman, the amplitude itself is just the sum of the contributions of all the possibilities to propagate from x_a to x_b

$$K(x_b, x_a, T) = \sum_{\text{paths}} A[x(t)] \quad (3)$$

where the sum over paths includes all paths that start at x_a and end at x_b a time T later and we assume A , according to the superposition principle, to be a phase factor $A[x(t)] = e^{i\Phi[x(t)]}$.

In the classical limit only the classical path will contribute. This classical path satisfies the Principle of Minimum Action:

$$\frac{\delta}{\delta x(t)} (S[x(t)])|_{x=x_c} = 0. \quad (4)$$

¹Quantities describing the probability of propagation from one space-time point to another.

²See [3] p.306 f.

On the other hand we expect by the method of stationary phase:

$$\frac{\delta}{\delta x(t)}(\Phi[x(t)])|_{x=x_{cl}} = 0 \quad (5)$$

Therefore it is reasonable to assume that each contribution of a path has a phase depending only on the action S , which in the limit $\hbar \rightarrow 0$ gives the classical path according to the stationary phase approximation. Finally we get

$$\Phi[x(t)] = e^{\frac{i}{\hbar}S[x(t)]} \quad (6)$$

and the propagator is written as follows

$$\langle x_b | e^{-iHT/\hbar} | x_a \rangle = \int \mathcal{D}x(t) e^{\frac{i}{\hbar}S[x(t)]} \quad (7)$$

where $\int \mathcal{D}(t) = \sum_{\text{paths}}$. One can show that this propagator solves the Schrödinger equation.

2.2 Path integrals on the lattice

In order to perform numerical calculations with the path integral, we must provide a mathematical rigorous definition. We do so by introducing a time-lattice with spacing $a = \frac{T}{N}$, where N is the number of lattice points:

$$K(x_b, x_a, T) = \langle x_b | e^{-iHT/\hbar} | x_a \rangle = \langle x_b | (e^{-iHa/\hbar})^N | x_a \rangle \quad (8)$$

Now we insert a complete set of intermediate states between each of these factors, $\mathbf{1} = \int dx_k |x_k\rangle \langle x_k|$ and obtain:

$$K(x_b, x_a, T) = \langle x_b | e^{-iHT/\hbar} | x_a \rangle = \left(\prod_{i=1}^{N-1} \int_{-\infty}^{+\infty} dx_i \right) \prod_{i=0}^{N-1} \langle x_{i+1} | e^{-iHa/\hbar} | x_i \rangle \quad (9)$$

where we have defined $x_0 = x_a$ and $x_N = x_b$. The Hamiltonian H is a function of momentum and position, $H = H(x, p)$, and in general we cannot commute x and p in order to get a resulting action which is symmetric in nearest neighbour pairs x_i and x_{i+1} . But if H does not contain products of x and p we can do so up to first order in a . One way to show this is the Baker-Campbell-Hausdorff formula. But it is easier to see by developing in orders of a and then splitting the parts of H , which depend only on x and p :

$$\langle x_{i+1} | e^{-iaH/\hbar} | x_i \rangle \approx \langle x_{i+1} | 1 - iaH/\hbar + \mathcal{O}(a^2) | x_i \rangle \quad (10)$$

We thus find after insertion of a complete set of momentum eigenstates:

$$\langle x_{i+1} | H(q, p) | x_i \rangle = \langle x_{i+1} | \frac{H_q(x_{i+1})}{2} + H_p(p) + \frac{H_q(x_i)}{2} | x_i \rangle = \int \frac{dp_i}{2\pi\hbar} \frac{1}{2} (H(x_{i+1}, p_i) + H(x_i, p_i)) e^{i \frac{p_i(x_{i+1}-x_i)}{\hbar}} \quad (11)$$

And going back to exponential form we get the resulting expression:

$$K(x_b, x_a, T) = \left(\prod_i \int dx_i \int \frac{dp_i}{2\pi\hbar} \right) e^{\frac{i}{\hbar}a \left(\sum_i \frac{p_i(x_{i+1}-x_i)}{a} - \frac{1}{2} (H(x_{i+1}, p_i) + H(x_i, p_i)) \right)} \quad (12)$$

where we see that $\frac{x_{i+1}-x_i}{a}$ is just the discretized time derivative of the position, \dot{x} , and therefore we can identify the Legendre transform of the Hamiltonian $L(x, \dot{x}) = p\dot{x} - H(x, p)$. Overall we get a discretised action $S = \int L dt = a \sum_i L(\frac{x_{i+1}+x_i}{2}, \frac{x_{i+1}-x_i}{a})$.

2.3 Euclidean path integral

As one may have noticed from the representation of the kernel, there is a very close connection between classical statistical mechanics and quantum theory, which is revealed by Feynman's path integral approach.

This connection opens up the possibility to use numerical methods that are well known from statistical mechanics, the Metropolis method, for example. In this subsection we will therefore try to establish this connection by performing a Wick rotation in the complex plane, going from a complex Minkowski to an Euclidean path integral.

When we compare the kernel for propagation, $K(x_b, x_a, T) = \langle x_b | e^{-iHT/\hbar} | x_a \rangle$, which we have introduced in Eq. (2), to the partition function of statistical mechanics

$$Z = \int_{-\infty}^{\infty} dx \langle x | e^{-\beta H} | x \rangle = \text{Tr}[e^{-\beta H}] \quad (13)$$

we immediately see that the partition function is just an integral over the kernel, namely $Z = \int_{-\infty}^{\infty} dx K(x, x, T)$ for an imaginary time $T = -i\beta\hbar$. This clearly motivates the definition of an Euclidean path integral

$$Z_{fi} = \langle x_f | e^{-HT/\hbar} | x_i \rangle = \int \mathcal{D}x e^{-S_E[x]/\hbar} \quad (14)$$

where we have Wick rotated by $\frac{\pi}{2}$ in the complex plane, from real to imaginary time $\tau = it$, leading us to an Euclidean action

$$S_E[x] = \int_0^T d\tau \left[\frac{1}{2} m \left(\frac{dx}{d\tau} \right)^2 + V(x) \right] \quad (15)$$

with the boundary conditions $x(0) = x_i$ and $x(T) = x_f$.

On the lattice this action may again be approximatively discretised to read:

$$S_{lat}[x] = \sum_{i=0}^N a \left[\frac{1}{2} m \frac{(x_{i+1} - x_i)^2}{a^2} + \frac{V(x_{i+1}) + V(x_i)}{2} \right] \quad (16)$$

From the preceding definition of the partition function, which is $Z = \int dx_i dx_f \delta(x_i - x_f) Z_{fi}$, one may immediately verify that $e^{-S[x]/\hbar}$ in the case of the Euclidean path integral is just the Boltzmann factor, where \hbar is the temperature. That means, as the reader may recall from statistical mechanics, that we can write the expectation value of any observable in terms of a path integral over these Boltzmann factors:

$$\langle O \rangle = \frac{\int \mathcal{D}x O(x) e^{-S[x]/\hbar}}{\int \mathcal{D}x e^{-S[x]/\hbar}} \quad (17)$$

3 Analytic examples

3.1 The Free Particle

Following the discussion in Sec. 2.2, we now consider the simplest system, the free particle. The Lagrangian will be of the form $L(x, \dot{x}) = m \frac{\dot{x}^2}{2}$ and therefore the kernel can be written as

$$K(x_b, x_a, T) = \left(\frac{1}{A} \prod_i \int \frac{dx_i}{A} \right) e^{i \frac{m}{2\hbar} \sum_i \left(\frac{x_{i+1} - x_i}{a} \right)^2} \quad (18)$$

with the normalization constant $A = \left(\frac{2\pi i \hbar a}{m} \right)^{1/2}$. This result can be obtained by completing the square for the Hamiltonian $H = \frac{p^2}{2m}$ and integrating the momenta³.

³See [1].

Position integrals may be evaluated as well by completing the square starting with the first integral

$$\int_{-\infty}^{+\infty} \left(\frac{2\pi i \hbar a}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar a} [(x_2-x_1)^2 + (x_1-x_0)^2]} dx_1 = \left(\frac{2\pi i \hbar (2a)}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar (2a)} (x_2-x_0)^2} \quad (19)$$

The derivation is given in App. [A]. We finally get the kernel:

$$K(x_b, x_a, T) = K(x_N, x_0, T) = \left(\frac{2\pi i \hbar T}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar T} (x_0-x_N)^2} \quad (20)$$

3.2 The Harmonic Oscillator

As the Harmonic Oscillator is most probably the physical model with the widest range of applications it is tempting to try to solve it's path integral. The Lagrangian for the Harmonic Oscillator is given to be of the form:

$$L(x, \dot{x}) = \frac{m}{2} \dot{x}^2 - \frac{m\omega^2 x^2}{2} \quad (21)$$

Therefore we consider the kernel:

$$K(x_b, x_a, T) = \int_a^b e^{\frac{i}{\hbar} \int_{t_a}^{t_b} [\frac{m}{2} (\dot{x}^2 - \omega^2 x^2) dt]} \mathcal{D}x(t) \quad (22)$$

In order to solve this path integral we can, as we already know the classical trajectory, split the path $x(t) = x_{cl} + y(t)$ into the classical part and a deviation $y(t)$ from the classical path which must vanish at the boundaries, i.e. $y(0) = y(T) = 0$.

Now if we try to calculate the action functional $S[x(t)]$ in terms of parts, $S[x_{cl}(t), y(t)]$, we find that it is separable, i.e. $S[x_{cl}(t), y(t)] = S_{cl} + S[y(t)]$, because mixed terms, after integration of the Lagrangian considered here, vanish for the boundary conditions.

Accordingly we can write our propagator as

$$K(x_b, x_a, T) = F(T) e^{\frac{i}{\hbar} S_{cl}} \quad (23)$$

where S_{cl} is the classical action and the factor

$$F(T) = \int_a^b e^{\frac{i}{\hbar} \int_0^T [\frac{m}{2} (\dot{y}^2 - \omega^2 y^2) dt]} \mathcal{D}y(t) \quad (24)$$

only depends on the relative time difference T .

We derive the classical action for the Harmonic Oscillator in App [B]. The result for that is:

$$S_{cl} = \int_{t_a}^{t_b} dt \frac{m}{2} [\dot{x}^2 - \omega^2 x^2] = \frac{m}{2} \frac{\omega}{\sin \omega T} [(x_a^2 + x_b^2) \cos \omega T - 2x_a x_b] \quad (25)$$

This gives us the following (intermediate) form of the propagator:

$$K(x_b, x_a, T) = F(T) e^{\frac{im\omega}{2\hbar \sin \omega T} [(x_a^2 + x_b^2) \cos \omega T - 2x_a x_b]} \quad (26)$$

But in order to determine the factor $F(T)$ we still need to evaluate the path integral for the fluctuations $y(t)$. We do so by again introducing a time lattice $y(t_n) = y_n$ for the action $S[y(t)]$ of the Harmonic Oscillator according to the method described in Sec. 2.2. This procedure leads to an action

$$\frac{i}{\hbar} S[y(t)] = \sum_{n=0}^N \left(\frac{im(y_{n+1} - y_n)^2}{2a\hbar} - \frac{iam\omega^2 (y_{n+1}^2 + y_n^2)}{4\hbar} \right) \quad (27)$$

and to the following path integral

$$F(T) = \left(\frac{m}{2\pi\hbar ia}\right)^{N/2} \left(\prod_{i=1}^{N-1} \int_{-\infty}^{+\infty} dy_i\right) \exp\left(\sum_{n=0}^N \frac{im(y_{n+1} - y_n)^2}{2a\hbar} - \frac{iam\omega^2(y_{n+1}^2 + y_n^2)}{4\hbar}\right) \quad (28)$$

which we are going to solve by performing the Fourier transformation

$$y_n = \frac{1}{\sqrt{N}} \sum_{m=1}^{N-1} \sin\left(\frac{\pi nm}{N}\right) A_m \quad (29)$$

according to the boundary conditions given above.

We show the calculations in App. [C]. We get:

$$\begin{aligned} \sum_{n=0}^{N-1} y_n^2 &= \sum_n \frac{1}{2} A_n^2 \\ \sum_{n=0}^{N-1} y_{n+1}^2 &= \sum_n \frac{1}{2} A_n^2 (1 + C(n)/N) \\ \sum_{n=1}^{N-1} (y_{n+1} - y_n)^2 &= \sum_n A_n^2 2 \sin^2\left(\frac{\pi n}{2N}\right) \end{aligned} \quad (30)$$

This finally gives us the Fourier transform of the action

$$\frac{i}{\hbar} S[y] = \sum_{n=1}^N \left[\frac{im}{2a\hbar} A_n^2 2 \sin^2\left(\frac{\pi n}{2N}\right) - \frac{iam\omega^2}{4\hbar} A_n^2 \right] \quad (31)$$

and accordingly after transformation of variables the path integral

$$F(T) = J \int \mathcal{D}A \cdot e^{\sum_{n=1}^N \left[\frac{im}{2a\hbar} A_n^2 2 \sin^2\left(\frac{\pi n}{2N}\right) - \frac{iam\omega^2}{4\hbar} A_n^2 \right]} \quad (32)$$

where J is the Jacobian $J = \left| \frac{\partial y_n}{\partial A_m} \right| = \left| \frac{1}{\sqrt{N}} \sin\left(\frac{\pi nm}{N}\right) \right| = \sqrt{2}^{1-N}$ of the transformation of variables. Thus, with the appropriate normalization A , we have:

$$F(T) = \frac{J}{A} \prod_{n=1}^{N-1} \int \frac{dA_n}{A} e^{\sum_{n=1}^N \left[\frac{im}{2a\hbar} A_n^2 2 \sin^2\left(\frac{\pi n}{2N}\right) - \frac{iam\omega^2}{4\hbar} A_n^2 \right]} \quad (33)$$

which is just a Gaussian integral that we can solve easily with $A = \sqrt{\frac{2\pi\hbar ia}{m}}$ to be equal to:

$$F(T) = J \left(\sqrt{\frac{m}{2\pi i\hbar a}}\right)^N \left(\sqrt{\frac{2\pi\hbar ia}{m}}\right)^{-\frac{N-1}{2}} \prod_{n=1}^{N-1} \left[2 \sin^2\left(\frac{\pi n}{2N}\right) - \frac{ia^2 m \omega^2}{2} \right]^{-\frac{1}{2}} \quad (34)$$

Now, in the next step, we will factorize the sin and approximate it in the limit of $N \rightarrow \infty$, i.e. $\frac{a^2 \omega^2}{4 \sin^2\left(\frac{\pi n}{2N}\right)} = \left(\frac{\omega T}{\pi n}\right)^2$, which results in:

$$F(T) = J \sqrt{\frac{m}{2\pi i\hbar a}} \prod_{n=1}^{N-1} \left[2 \sin^2\left(\frac{\pi n}{2N}\right) \left(1 - \left(\frac{\omega T}{\pi n}\right)^2 \right) \right]^{-\frac{1}{2}} \quad (35)$$

If we use $\prod_{n=1}^{\infty} \left(1 - \frac{x^2}{\pi^2 n^2}\right) = \frac{\sin x}{x}$ then, we get:

$$F(T) = J \sqrt{\frac{m}{2\pi i\hbar a}} \sqrt{\frac{\omega T}{\sin \omega T}} \left[\prod_{n=1}^{N-1} \frac{1}{2} \left(2 \sin\left(\frac{\pi n}{2N}\right) \right)^2 \right]^{-\frac{1}{2}} \quad (36)$$

which by rearranging the infinite product with $\prod_{n=1}^{N-1} 4 \sin^2\left(\frac{\pi n}{2N}\right) = \frac{1}{2} \prod_{n=1}^{2N-1} 2 \sin\left(\frac{\pi n}{2N}\right)$ gives us

$$F(T) = J \sqrt{\frac{m}{2\pi i \hbar a}} \sqrt{\frac{\omega T}{\sin \omega T}} \sqrt{2}^{N-1} \left[\frac{1}{2} \prod_{n=1}^{2N-1} \left(2 \sin\left(\frac{\pi n}{2N}\right) \right) \right]^{-\frac{1}{2}} \quad (37)$$

where we will now use $\prod_{n=1}^{N-1} 2 \sin\left(\frac{n\pi}{N}\right) = N$ and $Na = T$ to get:

$$F(T) = J \sqrt{\frac{m\omega}{2\pi i \hbar \sin \omega T}} \sqrt{N} \sqrt{2}^{N-1} \left(\frac{2N}{2}\right)^{-\frac{1}{2}} \quad (38)$$

So with the Jacobian given above we get the final result

$$F(T) = \sqrt{\frac{m\omega}{2\pi i \hbar \sin \omega T}} \quad (39)$$

which we can combine with the result for the classical action to get the kernel of the Harmonic Oscillator:

$$K(x_b, x_a, T) = \sqrt{\frac{m\omega}{2\pi i \hbar \sin \omega T}} e^{\frac{i m \omega}{2 \hbar \sin \omega T} [(x_a^2 + x_b^2) \cos \omega T - 2x_a x_b]} \quad (40)$$

4 Numerical methods

4.1 HMC algorithm

If we want to calculate the expectation value of an observable numerically, one standard way is the Monte-Carlo method. The basic idea is to draw random numbers distributed according to our Boltzmann factor $e^{-S[x]/\hbar}$ with a probability distribution:

$$P(x_k) \mathcal{D}x = \frac{e^{-S[x]/\hbar} \mathcal{D}x}{\int \mathcal{D}x e^{-S[x]/\hbar}} \quad (41)$$

Once we have generated sequences of random numbers $(x_k)_{k \in N}$ fulfilling this probability distribution, the expectation value of our observable O , turns out to be as simple as a sum:

$$\langle O \rangle = \frac{1}{N} \sum_{k=1}^N O(x_k) \quad (42)$$

For $N = \infty$ this would provide the exact value for $\langle O \rangle$. Of course, in practise N is always finite and we only obtain some estimate of $\langle O \rangle$ which is correct within a certain error. The computation of a reliable error is one of the most important problems for the numerical simulations and we will discuss it a lot below.

Now in Statistical Mechanics there are well known methods to achieve such probability distributions. One of these methods is called the Hybrid Monte-Carlo algorithm (HMC).

After this short motivation let us consider how we can obtain the HMC algorithm, at least in a heuristic way. We start a stochastic differential equation, the Langevin equation (a famous description of Brownian motion),

$$\dot{\pi} = -\gamma\pi + \eta(t) \quad (43)$$

with π being the momentum of a particle driven by a random force η with the Gaussian noise properties $\langle \eta(t) \rangle = 0$ and $\langle \eta(t)\eta(t') \rangle = \delta(t-t')$ and γ is a friction coefficient.

The probability to find a particle at time t with a momentum π is given by the solution of the Fokker-Planck equation, which is an equivalent form of the Langevin equation for the probability:

$$\frac{\partial}{\partial t}P(\pi, t) = \gamma \frac{\partial}{\partial \pi}(\pi P) + \frac{1}{2} \frac{\partial^2 P}{\partial \pi^2} \quad (44)$$

Moreover if we consider Brownian motion in an external field $F = -\frac{\partial}{\partial x}V(x)$ we are led to the following equations:

$$m\dot{x} = \pi \quad (45)$$

$$\dot{\pi} = -\gamma\pi + F + \eta(t) \quad (46)$$

This system of stochastic differential equations is again equivalent to a generalized Fokker-Planck equation for the probabilities, the Kramer's equation:

$$\frac{\partial}{\partial t}P(x, \pi, t) + \pi \frac{\partial P}{\partial x} + F(x) \frac{\partial P}{\partial \pi} = \gamma \frac{\partial}{\partial \pi}(\pi P) + \frac{1}{2} \frac{\partial^2 P}{\partial \pi^2} \quad (47)$$

The solution of the Kramer's equation in the limit of large time scales leads to the desired exponential probability distribution:

$$\lim_{t \rightarrow \infty} P(x, \pi, t) \propto e^{-\frac{\gamma}{2}\pi^2} e^{-V(x)} \quad (48)$$

Thus we obtain for large times a stationary solution and, moreover, this solution is even unique.

Let us compare this now to the expectation value of our operator

$$\langle O \rangle = \frac{\int \mathcal{D}x O(x) e^{-S[x]/\hbar}}{\int \mathcal{D}x e^{-S[x]/\hbar}} = \frac{\int \mathcal{D}x \int \mathcal{D}\pi O(x) e^{-\frac{1}{2}\pi^2 - S[x]/\hbar}}{\int \mathcal{D}x \int \mathcal{D}\pi e^{-\frac{1}{2}\pi^2 - S[x]/\hbar}} \quad (49)$$

where we have just added a Gaussian integral factor of one, set $\gamma = 1$ and let the action play the role of the potential in Eq. (46).

Thus we may consider a 1-dimensional Hamiltonian:

$$H = \frac{1}{2}\pi^2 + S[x(t)] \quad (50)$$

This Hamiltonian depends on virtual momenta $\pi(\tau)$, which are Gaussian distributed as functions of a virtual Monte-Carlo time τ :

$$\langle \pi(\tau) \rangle = 0 \quad (51)$$

$$\langle \pi(\tau)\pi(\tau') \rangle = \delta(\tau - \tau') \quad (52)$$

Let us now evolve the system in this virtual Monte-Carlo time according to the set of stochastic differential equations

$$\dot{x} = \pi \quad (53)$$

$$\dot{\pi} = -\frac{\partial}{\partial x}S \quad (54)$$

which we will refer as Hamilton's equations. Then, in analogy to the case of the Kramer's equation, in this virtual Monte-Carlo time our probability distribution converges to the desired form, namely the Boltzmann factor $e^{-S[x]/\hbar}$. Thus our ingredients for the algorithm are the solutions of Hamilton's equations and to generate π as Gaussian distributed random numbers. After this, we are sure that we obtain the desired probability distribution for the final positions and momenta.

Numerically we may solve Hamilton's equations in a discretized form, using the so called leapfrog scheme (where we use alternating time steps of for positions and momenta), which is time-invariant and cancels some of the discretization errors in $\delta\tau$, which denotes the non-zero step size in the numerical integration. In a first step we create the starting values for the momenta at $\tau = 0$, which are a Gaussian distributed

random number for $\pi(t)[0]$ and a flat random distribution for $x(t)[0]$. The second step is the integration of Hamilton's equations via the leapfrog scheme

$$\pi(t) \left[\frac{\delta\tau}{2} \right] = \pi(t)[0] - \frac{\delta\tau}{2} \left[\frac{\partial}{\partial x} S \right]_{x[0]} \quad (55)$$

$$x(t) [\delta\tau] = x(t)[0] + \pi(t) \left[\frac{\delta\tau}{2} \right] \delta\tau \quad (56)$$

$$\pi(t) \left[3 \frac{\delta\tau}{2} \right] = \pi(t) \left[\frac{\delta\tau}{2} \right] - \delta\tau \left[\frac{\partial}{\partial x} S \right]_{x[\delta\tau]} \quad (57)$$

etc. Moreover, in a third step, we can compensate for the discretization error $\delta\tau$ by inserting a Metropolis like accept/reject step, i.e. we accept the new position configuration $[x_{end}, \pi_{end}]$ only with an acceptance probability:

$$P_{accept} = \min(1, e^{H(x_{in}, \pi_{in}) - H(x_{end}, \pi_{end})}) \quad (58)$$

This acceptance probability is realized by creating a random number $r \in (0, 1]$: if $e^{\Delta H} < r$ then we accept, otherwise we do not. We create the next random values, by starting with the preceding random value, which we put into Hamilton's equations and so on, until we have created the desired number of the random distributed $[x_k, \pi_k]$.

4.2 Simulation

The above sketched algorithm has been implemented and used for all our simulations. Now we are going to give a brief explanation of the input parameters, which are important to understand how we actually run the program and what we will do afterwards in our numerical analysis. The input parameters for the simulations are:

Physical parameters:

m : mass of the Harmonic Oscillator.

ω : frequency of the Harmonic Oscillator.

\hbar : reduced Planck constant. We will set this to 1 for simplicity.

Lattice parameters:

a : lattice spacing.

N : number of lattice points.

Note: $aN = T$, where T is the physical time of the problem.

Simulation and algorithm parameters:

seed: changing this number selects different sequences of the random numbers that are generated.

Nsteps: number of steps that the algorithm makes to move from one complete trajectory (involving all lattice points) to a new one on which, in principle, measurements can be performed.

$\delta\tau$: size of the steps that the algorithm makes to increment the values of the position x and the momentum π , to move from one trajectory to the next one.

Note: We will fix $\delta\tau Nsteps = 1$. These two parameters affect the simulation about accepting or rejecting trajectories. Setting $\delta\tau$ smaller and *Nsteps* bigger increases the acceptance rate, but the simulation takes longer to run.

Ntraj: number of trajectories to be generated and considered.

Nskip: number of trajectories to be skipped. This means that the algorithm will generate *Nskip* trajectories before using one for measurements. In practise this means that only every *Nskip* trajectory is written to disk.

Note: The total number of trajectories that the program generates is $Ntraj \times Nskip$, but only *Ntraj* are used to compute the output quantities. The number of output trajectories will always be *Ntraj*.

Let us explain what the algorithm does, including now the parameters. Each trajectory may also be called a configuration, in analogy with field theoretical models. In each configuration we have N numbers,

which are the positions of the lattice points. So we start at one initial configuration which has $x(t) = 0$ at all times t on the lattice. Then we generate N random numbers for the momentum so as we can move to another configuration. At this point we solve the Hamiltonian equations, where the potential in the hamiltonian is our action for the Harmonic Oscillator, as shown in Eq. (50). We do this numerically, making $Nsteps$ of length $\delta\tau$. In the new configuration, we compute the action. If the action is decreased compared to the action for the previous configuration, we keep the new configuration; otherwise we reject it and move back to the previous configuration (we would like to achieve minimum action, as stated in the Principle of Minimum Action). We go on in this way, generating configurations up to $Ntraj$ number of them. Finally, we compute the expectation value of our observable as shown in Eq. (42) i.e. averaging its value over all configurations.

5 Results

In the first set of simulations we performed, we made ourselves familiar with the simulation program, the algorithm parameters and the output of the quantities of interest. We ran the simulation for different values of the parameters, i.e. the lattice spacing a and the number of discrete time points N . We set $\hbar = 1$ always, and choose the lattice spacing a and the number of lattice points N so as to have the total time extent $T = 10$ (remember that $aN = T$). Additionally, we always want the acceptance rate of the trajectories to be bigger than 90%, in order to get good results. A large acceptance rate can be achieved by making the step size $d\tau$ small enough while keeping the trajectory length $d\tau Nsteps = 1$ where $Nsteps$ is the number of steps in the leapfrog integration. We observed that making the lattice spacing smaller while keeping $d\tau$ fixed leads to a significant decrease of the acceptance rate. Therefore, when decreasing the values of the lattice spacing, also the values for $d\tau$ must be decreased and the number of steps $Nsteps$ increased.

5.1 Autocorrelation effects

After having tuned the values of $d\tau$ for each value of the lattice spacing we used, we can now performed some simulations and look at the physical results. To do that, the simulation program provides as an output $x_{\tau_{MC}}(t)$, i.e. we get a value of x for each trajectory, denoted by τ_{MC} , and for each lattice point, denoted by t (which is actually the time in lattice units, as $aN = T$). In this way, we can store all trajectories and calculate several different observables after the actual simulation.

We first concentrated on the values of $x^2(\tau_{MC})$ measured on each trajectory τ_{MC} . Producing many trajectories, we can determine the expectation value of $\langle x^2 \rangle$, which is the quantity we are going to study:

$$\langle x^2 \rangle = \frac{1}{Ntraj} \sum_{Ntraj} x^2(\tau_{MC}) \quad (59)$$

In the same way, the error of $\langle x^2 \rangle$ can be calculated by the standard error formula

$$\sigma_{x^2} = \sqrt{\frac{\langle (x^2 - \langle x^2 \rangle)^2 \rangle}{Ntraj}} \quad (60)$$

After some previous trials, we decided to fix the following parameters for the rest of our work: the physical parameters $m = 1.2$ and $\omega = 0.5$; the number of trajectories $Ntraj = 100000$, which we consider big enough so as to obtain good statistics. Another important parameter for the simulations is the number $Nskip$ of skipped trajectories between measurements. At this point, we are as well setting $Nskip = 1$, i.e. every trajectory generated in the Monte-Carlo process is used for measurements.

Some examples of the outcomes of the runs performed in the above described setup are plotted in Fig. 1, which shows the Monte-Carlo time history of $\langle x^2 \rangle$ for two values of the lattice spacing. Averaging $x^2(\tau_{MC})$

over these Monte-Carlo histories, we can obtain its expectation value according to Eq. (59) and its error from Eq. (60). At this point, we want to compare the values of $\langle x^2 \rangle$ with the exact values at each of the different lattice spacings we have simulated. We can calculate these exact values by the formula in the Creutz and Freedman paper [2]

$$\langle x^2 \rangle_{theory} = \frac{\hbar}{2m\omega\sqrt{1 + \frac{a^2\omega^2}{4}}} \left(\frac{1 + R^N}{1 - R^N} \right) \quad (61)$$

$$R = 1 + \frac{a^2\omega^2}{4} - a\omega\sqrt{1 + \frac{a^2\omega^2}{4}}$$

which gives the explicit dependence on the lattice a and the number of lattice points N .

Our first analysis led us to some funny results. We show them in Tab. 1. As we can see, we do not obtain a good agreement between the exact theoretical value and the outcome from the simulation. In particular, we find that the numerical values are inconsistent within the errors, which are very small.

a	$\langle x^2 \rangle_{theory}$	$\langle x^2 \rangle$	$error$
0.8	0.828609	0.823	0.0018
0.5	0.838264	0.837	0.0017
0.2	0.843609	0.856	0.0017
0.1	0.844382	0.850	0.0017
0.08	0.844474	0.840	0.0017
0.05	0.844575	0.831	0.0017
0.02	0.844629	0.824	0.0016
0.01	0.844637	0.841	0.0016
0.008	0.844638	0.846	0.0017
0.005	0.844639	0.869	0.0020
0.002	0.844639	0.763	0.0014
0.001	0.844639	0.830	0.0015

Table 1: Comparison of $\langle x^2 \rangle$ with naive errors.

We had to find a way to explain this discrepancy. If we look again at our time histories in Fig. 1 we can appreciate that for $a = 0.005$ clear bunches of data appear which do not differ much. This means that the trajectories are strongly correlated, and this led us to wrong results, because it reduces the amount of independent data. If we want to get right results for our observable we need more statistics, i.e. to run more trajectories in the simulation. Moreover, our formula for the error computation in Eq. (60) assumes that the data are uncorrelated. Given the strong correlation between the trajectories we observe, this formula can therefore not be applied and the error bars given in Tab. 1 are underestimated. We see this so-called autocorrelation between trajectories as well for $a = 0.05$, where the bunches are narrower but they still exist.

The way to fix this is to determine the autocorrelation time τ_{int} . This parameter gives us the number of trajectories which are correlated, i.e. the width of the bunches. Once we have determined τ_{int} , we have two ways to proceed to solve the problem described above. The first way is to compute the error by taking the autocorrelation time into account. The second way is to set the value of skipped trajectories $N_{skip} > \tau_{int}$ in which case we will have uncorrelated measurements and the naive error formula can be applied. We will discuss this later in more detail.

An important observation is that the autocorrelation time is dependent on the lattice spacing as can already be seen in Fig. 1 at least on a qualitative level. Theoretically we expect that $\tau_{int} \cong 1/a$. This result comes from the form of the discretized action for the Harmonic Oscillator that we are studying. In Eq. (31) in Sec. 3.2 we have the form of the action on the lattice. The autocorrelation time is inversely proportional to the action kernel in momentum space. Therefore the longest autocorrelation time comes from the slowest mode which is the one at momentum zero. Hence, the autocorrelation time is given by $\tau_{int} = \frac{4\hbar}{m\omega^2 a} \frac{1}{A_n^2}$ which leads to the $1/a$ law.

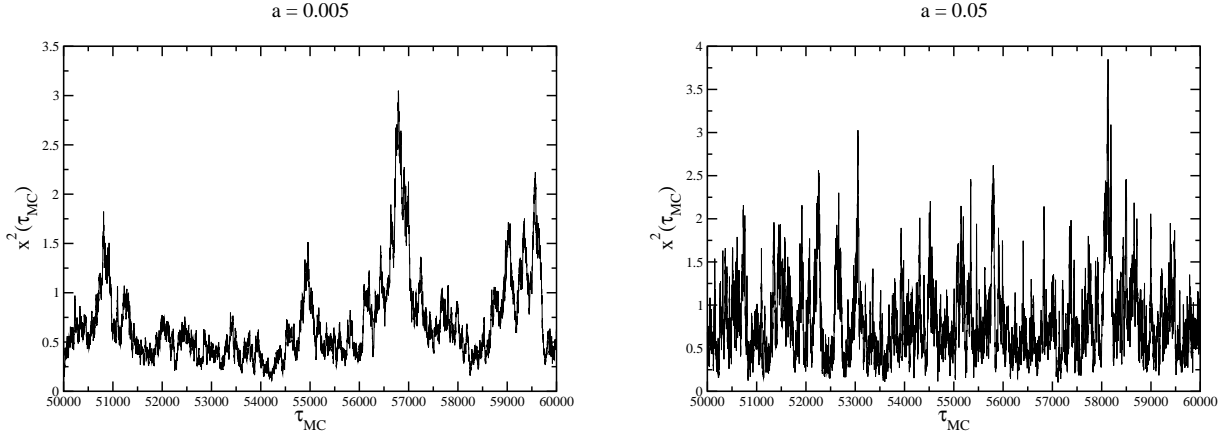


Figure 1: Time histories for $a=0.005$ and 0.05 . We can observe autocorrelations between trajectories, higher for $a = 0.005$ than for $a = 0.05$.

To determine the autocorrelation time or “integrated autocorrelation time” we use the autocorrelation function, which shows an exponential decay with a rate that is given by τ_{int} :

$$\Gamma_{x^2}(d\tau_{MC}) = \langle (x^2(\tau_{MC}) - \langle x^2 \rangle)(x^2(\tau_{MC} + d\tau_{MC}) - \langle x^2 \rangle) \rangle \cong e^{-d\tau_{MC}/\tau_{exp}} \quad (62)$$

The autocorrelation time can be now obtained by just fitting this autocorrelation function to an exponential. Instead of this one may also compute some effective autocorrelation time τ_{eff} given by $-1/\ln\left(\frac{\Gamma_{x^2}(\tau_{MC}+1)}{\Gamma_{x^2}(\tau_{MC})}\right) = \tau_{eff}$, which gives a plateau for some region in $d\tau_{MC}$. A very simple way to estimate the error is then to sum the biggest and the smallest value within the plateau and divide it by two. To be more accurate, we choose however the whole plateau region and fit in this region the exponential and the plateau. All of this is shown in Fig. 2.

Making use of this strategy, we obtain the autocorrelation time for every value of the lattice spacing we have performed simulations for. This is shown in Tab. 2.

a	τ_{int}	$error$
0.8	3.9	0.6
0.5	6.6	1.3
0.2	21	10
0.1	26	4
0.08	36	6
0.05	51	10
0.02	130	20
0.01	210	20
0.008	350	40
0.005	690	110
0.002	900	200
0.001	1600	400

Table 2: Autocorrelation time for the lattice spacing we performed.

If we again look at the last plot in Fig. 2, we can see that the scaling $\tau_{int} \cong 1/a$ is well reproduced. For big values of the lattice spacing, we see that the straight line is completely consistent with the data. For small lattice spacings we see some deviation. However, we are assuming that the variation in the plateau region corresponds to just one standard deviation σ for the uncertainty (the error), but we could also take

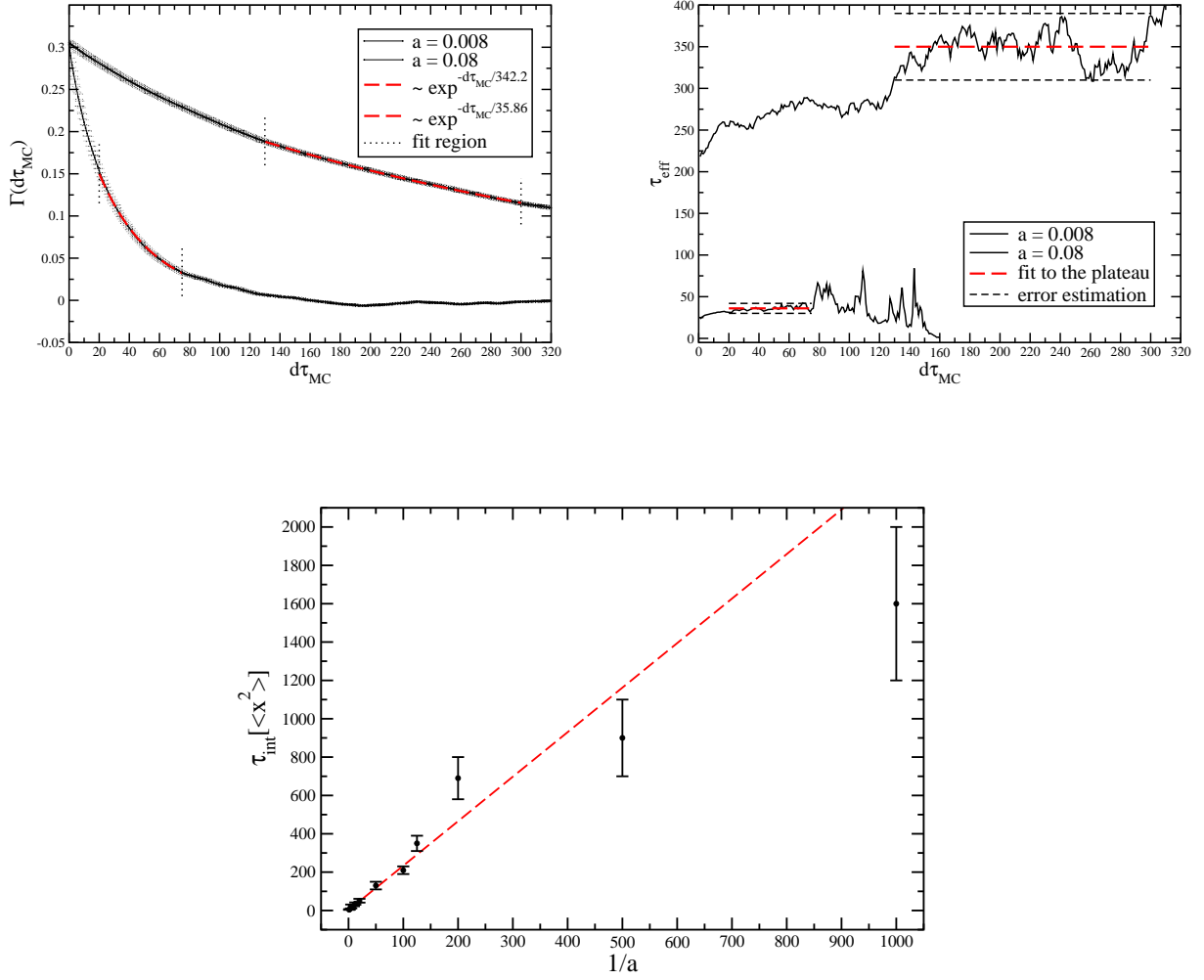


Figure 2: Calculation of the autocorrelation time and the error. While the exponential fit gives us a precise value for the autocorrelation time, the plateau method serves as a useful estimate of the error. We also show the scaling of the autocorrelation time with the lattice spacing. As can be seen from the graph, it follows the expected $1/a$ law.

2σ or even 3σ , as statistically reasonable. Within these 2 or 3 standard deviation bands, all of our results for τ_{int} would fall on the line (actually 2σ would be enough). Hence we can assume that our error study is correct and that the autocorrelation time is determined with acceptable accuracy.

5.2 $\langle x^2 \rangle$ results

Now that the autocorrelation time is known, we can fix the discrepancies for $\langle x^2 \rangle$ in two different ways. First we use this autocorrelation time to recalculate the error from the naive estimation. When the data is correlated, we have to include a correction in Eq. (60) that depends on the autocorrelation time, as follows:

$$\sigma_{x^2} = \sqrt{\frac{\langle (x^2 - \langle x^2 \rangle)^2 \rangle}{Ntraj}} \sqrt{\frac{\sum_{\tau_{MC}} \Gamma_{x^2}(d\tau_{MC})}{\Gamma_{x^2}(0)}} = \sqrt{\frac{\langle (x^2 - \langle x^2 \rangle)^2 \rangle}{Ntraj}} \sqrt{2\tau_{int}[x^2]} \quad (63)$$

If we do this, we get the results in Tab. 3. As we can see, now the agreement with the exact value from the formula in Creutz and Freedman paper is much more satisfactory.

a	$\langle x^2 \rangle_{theory}$	$\langle x^2 \rangle$	$error$
0.8	0.828609	0.823	0.0050
0.5	0.838264	0.837	0.0063
0.2	0.843609	0.856	0.011
0.1	0.844382	0.850	0.013
0.08	0.844474	0.840	0.015
0.05	0.844575	0.831	0.017
0.02	0.844629	0.824	0.026
0.01	0.844637	0.841	0.033
0.008	0.844638	0.846	0.046
0.005	0.844639	0.869	0.073
0.002	0.844639	0.763	0.058
0.001	0.844639	0.830	0.087

Table 3: Comparison of $\langle x^2 \rangle$ with corrected errors.

Second, now that we know the autocorrelation time we can run the simulation again, but keeping the $Nskip$ parameter bigger than the autocorrelation time that we have calculated for each value of the lattice spacing. Thus, for $Nskip > \tau_{int}$ the relevant trajectories on which measurements are performed will be taken only every $Nskip$ times, hence leading to uncorrelated trajectories and correspondingly independent data. That increases the accuracy of our result enormously, and we can use the naive error calculation without any correction, as the autocorrelation time now will be 1, as we explicitly checked again). Of course, the total number of trajectories increases from $Ntraj$ in the previous case to $Nskip \times Ntraj$ in this case and hence the simulation time is much longer, especially for small lattice spacings. We show the new results for $\langle x^2 \rangle$ in Tab. 4. The results from the simulations are now in total agreement with the theoretical expectation.

We can also compare the time histories for $Nskip = 1$, as we set in our first running, and for $Nskip > \tau_{int}$, which we have now. We plot that for the value of the lattice spacing $a = 0.005$ in Fig. 3. The difference between the two plots is visibly clear. As we see, when we skip a number of trajectories bigger than the autocorrelation time the bunches of correlated trajectories do not appear anymore, and the data is fluctuate without any sign of autocorrelation.

To sum up and demonstrate the substantial improvement, we plot various results for $\langle x^2 \rangle$ vs. a^2 in Fig. 4. The first and second plots show the data for $Nskip = 1$. The first one has naive error bars. As we see, the data is very far away from the exact value of the formula in Creutz and Freedman paper and even inconsistent amongst themselves. As described, we have traced this finding back to the fact that in the first naive analysis the error bars were underestimated because the autocorrelation time had not been taken into

a	$\langle x^2 \rangle_{exact}$	$\langle x^2 \rangle$	$error$
0.8	0.828609	0.8317	0.0018
0.5	0.838264	0.8364	0.0017
0.2	0.843609	0.8475	0.0017
0.1	0.844382	0.8428	0.0017
0.08	0.844474	0.8438	0.0017
0.05	0.844575	0.8484	0.0017
0.02	0.844629	0.8421	0.0017
0.01	0.844637	0.8454	0.0017

Table 4: Comparison of $\langle x^2 \rangle$ setting $Nskip > \tau_{int}$. The number of data sets is smaller because the running takes now much longer, as we are increasing enormously the number of trajectories that the simulation generates.

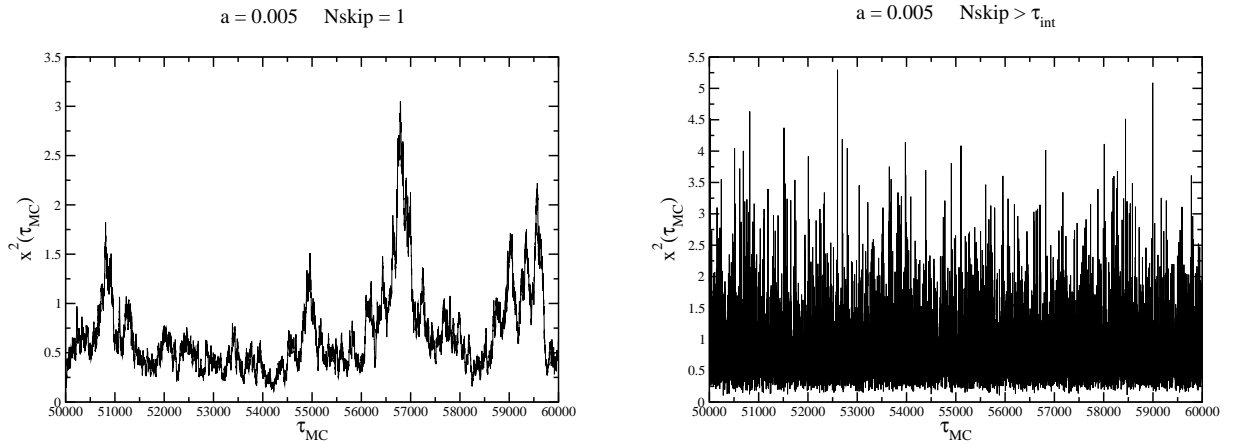


Figure 3: Time histories before and after setting $Nskip$ bigger than the autocorrelation time. All other parameters are held.

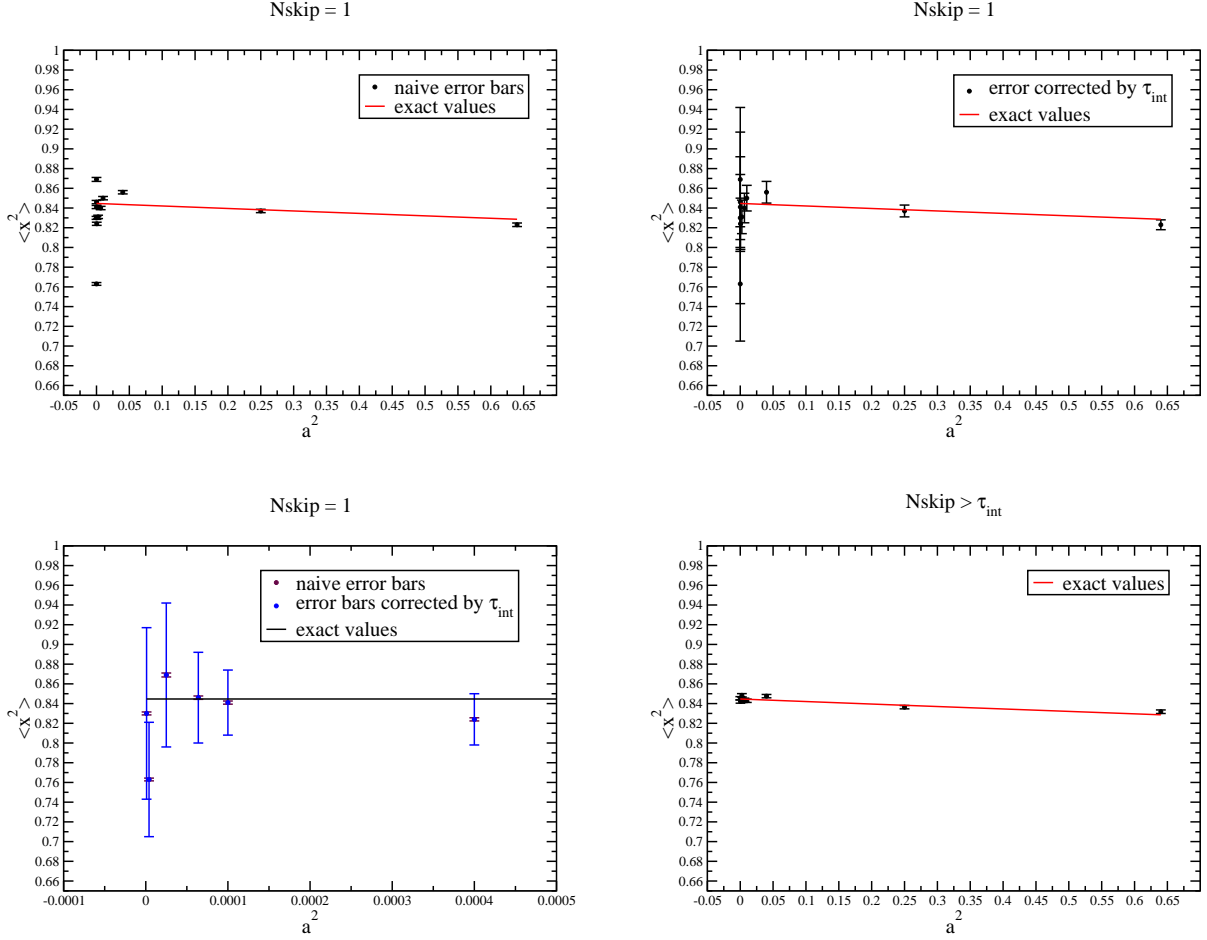


Figure 4: Scaling of $\langle x^2 \rangle$. The two plots at the top show $\langle x^2 \rangle$ vs. a^2 for $Nskip = 1$; the first one has naive error bars, which are not satisfactory, and the second one has the errors corrected by the autocorrelation time. The last plot shows the results when the simulation is run for $Nskip$ bigger than the autocorrelation time in each case.

account properly. The second plot has error bars corrected by the autocorrelation time. The effect is that the errors are substantially increased and within these larger error bars a much better agreement with the exact formula is obtained. The third plot is a zoom of these two graphs for the smallest lattice spacings, so that we can differentiate the various data points. The fourth plot shows the results when $Nskip > \tau_{int}$, i.e. when the data is not correlated. The values lie perfectly consistent on the line for the exact formula even with the very small naive error bars. Clearly this is the best strategy to get the right results although the simulations are rather long and demanding.

5.3 $\langle x(0)x(t) \rangle$ and ΔE

In this section, we are going to study another observable, namely the energy of the system. We run the simulation taking advantage of the previous analysis, so we keep $Nskip > \tau_{int}$. This way we do not have to worry about the data being correlated, as the autocorrelation time will be 1, as we have checked.

We want to construct the correlator, which obeys an exponential decay law in time with a rate that is

given by the energy⁴:

$$\langle x(0)x(t) \rangle \cong e^{-\frac{a\Delta E t}{\hbar}} \quad (64)$$

In order to obtain the energy ΔE we will fit the numerically computed correlator to the exponential function⁵. Note that we have to take into account the periodic boundary conditions for the correlator, which is symmetric about the middle of the lattice. So we must modify Eq. (64) to include contributions of two exponentials, as follows:

$$\begin{aligned} \langle x(0)x(t) \rangle &= A \left(e^{-\frac{a\Delta E t}{\hbar}} + e^{-\frac{a\Delta E(N-t)}{\hbar}} \right) = A e^{-\frac{a\Delta E N}{2\hbar}} \left(e^{-\frac{a\Delta E t}{\hbar}} e^{\frac{a\Delta E N}{2\hbar}} + e^{\frac{a\Delta E t}{\hbar}} e^{-\frac{a\Delta E N}{2\hbar}} \right) = \\ &= B \left(e^{-\frac{a\Delta E}{\hbar} \left(t - \frac{N}{2}\right)} + e^{\frac{a\Delta E}{\hbar} \left(t - \frac{N}{2}\right)} \right) = 2B \cosh \left[\frac{a\Delta E}{\hbar} \left(t - \frac{N}{2}\right) \right] \end{aligned} \quad (65)$$

where A and B are constants that do not depend on t . Thus, what we do is to use a cosh function instead of the exponential to obtain a value for the energy ΔE . To get error bars, we proceed as in the previous Sec. 5.1, and we invert the cosh⁶ at each time t in order to search for a plateau of the effective energy ΔE_{eff} . We set a fixed fitting region in physical time, which we choose to be from $at = 2$ to $at = T/2$ (remember that $T = 10$). We skip the first points on the lattice because these points have contributions from higher energy levels than the smallest gap. We also only take into account half of the lattice points because, as we said before, the correlator is symmetric about the middle of the lattice. The value for the energy can then also be obtained by a constant fit to the plateau values for $2 < at < T/2$. We estimate the error in this region, summing the biggest and the smallest values, but this time we will divide this by four instead of two, in order not to overestimate it⁷. We illustrate this in Fig. 5. The results that we get proceeding in this way are given in Tab. 5.

a	ΔE	$error$
0.8	0.496	0.004
0.5	0.501	0.006
0.2	0.497	0.009
0.1	0.496	0.016
0.08	0.499	0.020
0.05	0.508	0.025
0.02	0.50	0.07
0.01	0.50	0.08

Table 5: Results for the energy ΔE as obtained from the correlator analysis.

Note that from Quantum Mechanics $\Delta E = \hbar\omega$. As we set $\hbar = 1$, we should get $\Delta E = \omega$. But on the lattice, we have to introduce a correction dependent on the lattice spacing, as follows:

$$\omega_{theory} = \omega \sqrt{1 + \frac{a^2\omega^2}{4}} \approx \omega \left(1 + \frac{a^2\omega^2}{8} \right) \quad (66)$$

This is the exact formula for the energy in our case. In Fig. 5 we also plot our results for ΔE as a function of a^2 along with this exact formula⁸. We can see again that the agreement is very good: the line for the

⁴For a more rigorous way to proceed, we average not only over trajectories but also over time on the lattice, so what we really calculate is $\langle x(0)x(t) \rangle = \frac{1}{N \text{traj}} \sum_{\text{Ntraj}} \frac{1}{N} \sum_{s=0}^N x(s)x(s+t)$. Note that N is the number of lattice points.

⁵ $\Delta E = E_1 - E_0$, where E_0 is the energy of the ground state and E_1 the energy of the first excited satate.

⁶ $x(t) \cong \cosh \left[\frac{a\Delta E}{\hbar} \left(t - \frac{N}{2}\right) \right]$

$R(t) = \frac{x(t+1)+x(t-1)}{x(t)} = \frac{2 \cosh \left[\frac{a\Delta E}{\hbar} \left(t - \frac{N}{2}\right) \right] \cosh \left[\frac{a\Delta E}{\hbar} \right]}{\cosh \left[\frac{a\Delta E}{\hbar} \left(t - \frac{N}{2}\right) \right]} = \cosh \left[\frac{a\Delta E}{\hbar} \right]$

$\frac{a\Delta E}{\hbar} = \cosh^{-1} \left(\frac{R(t)}{2} \right) = \ln \left(\frac{R(t)}{2} + \sqrt{\frac{(R(t))^2}{4} - 1} \right)$

⁷Although this procedure is a little bit arbitrary, it provides a rather conservative error and gives an upper bound for the real error, which would need much more involved methods.

⁸For the exact formula, we should see some deviation from a straight line for big lattice spacing, but our range is not big enough to show this effect.

exact formula lies within the error bars for almost all the values and that deviations are not very big. Let us note here again that we have set $\omega = 0.5$ in the simulation, and that is what we get.

To finish we would like to compare the correlators for different lattice spacings. In Fig. 6 we show this in two different ways, which we find very interesting to compare. The first plot shows the correlator for three different values of a as a function of the time in lattice units. We can see that the smaller the lattice spacing, the longer the time extent: the correlator stretches in time for small lattice spacings and shrinks for big ones, but the maximum and the minimum values are the same. The second plot shows the correlator, but now as a function of the time in physical units (remember that we have set $T = 10$ as mentioned at the beginning of this report), for the same three values of a . As we see, the correlator tends to approach the same behaviour, and this is the right thing to have, as it depends on the energy, which is a physical quantity. Here we can also see clearly the different numbers of lattice points that we have when we change from one value of the lattice spacing to another, as we have kept T fixed.

6 Conclusions

In this paper we have described our work on Lattice Field Theory. We focus on a simple system, the quantum mechanical Harmonic Oscillator. This allows us to perform both, some analytical calculations and to get some numerical results by means of the Hybrid Monte-Carlo algorithm, so that we can show how the lattice methods work.

We started with a study of Quantum Mechanics and path integrals using the lattice point of view. We demonstrated with some simple examples how the path integral works and introduced a time lattice to discretise the Harmonic Oscillator. We can see already from these examples that the lattice is a nice way to approach difficult calculations in Quantum Mechanics that other methods, such as perturbative ones, cannot. We perform an explicit analytical calculation for the kernel of the lattice Harmonic Oscillator, on which we focus also further.

As a second part, we use the Hybrid Monte-Carlo algorithm to obtain some results for the Harmonic Oscillator on the lattice by means of numerical simulations. We explain the algorithm and our strategy for the simulations. We study two observables, which are the expectation value of the square of the position $\langle x^2 \rangle$ and the energy difference between the ground state and the first excited state $\Delta E = E_1 - E_0$. The agreement of our final results with the exact formulas, which we know for this simple case, is very good.

We emphasize the importance to take the autocorrelation time that appears in the simulations into account to achieve a correct error analysis. It is demonstrated that accounting for the autocorrelation time can lead to long simulations to reach sufficient statistics for precise results. Although for our example of the Harmonic Oscillator the simulations are very fast, this can constitute a problem in more complicated situations such as lattice QCD. Therefore, the development of improved algorithms that significantly reduce the autocorrelation time is of high importance.

We conclude that Lattice Theory is a nice method to do calculations. This kind of approach can be extended and better algorithms can be used to study more difficult systems and theories, such as QCD, where we do not know the exact values.

7 Acknowledgments

We wish to thank Dru Renner for his warm reception and support as well as for all the time he has spent on explanations and Jenifer González López for her nice help. We would also like to thank Karl Jansen for his

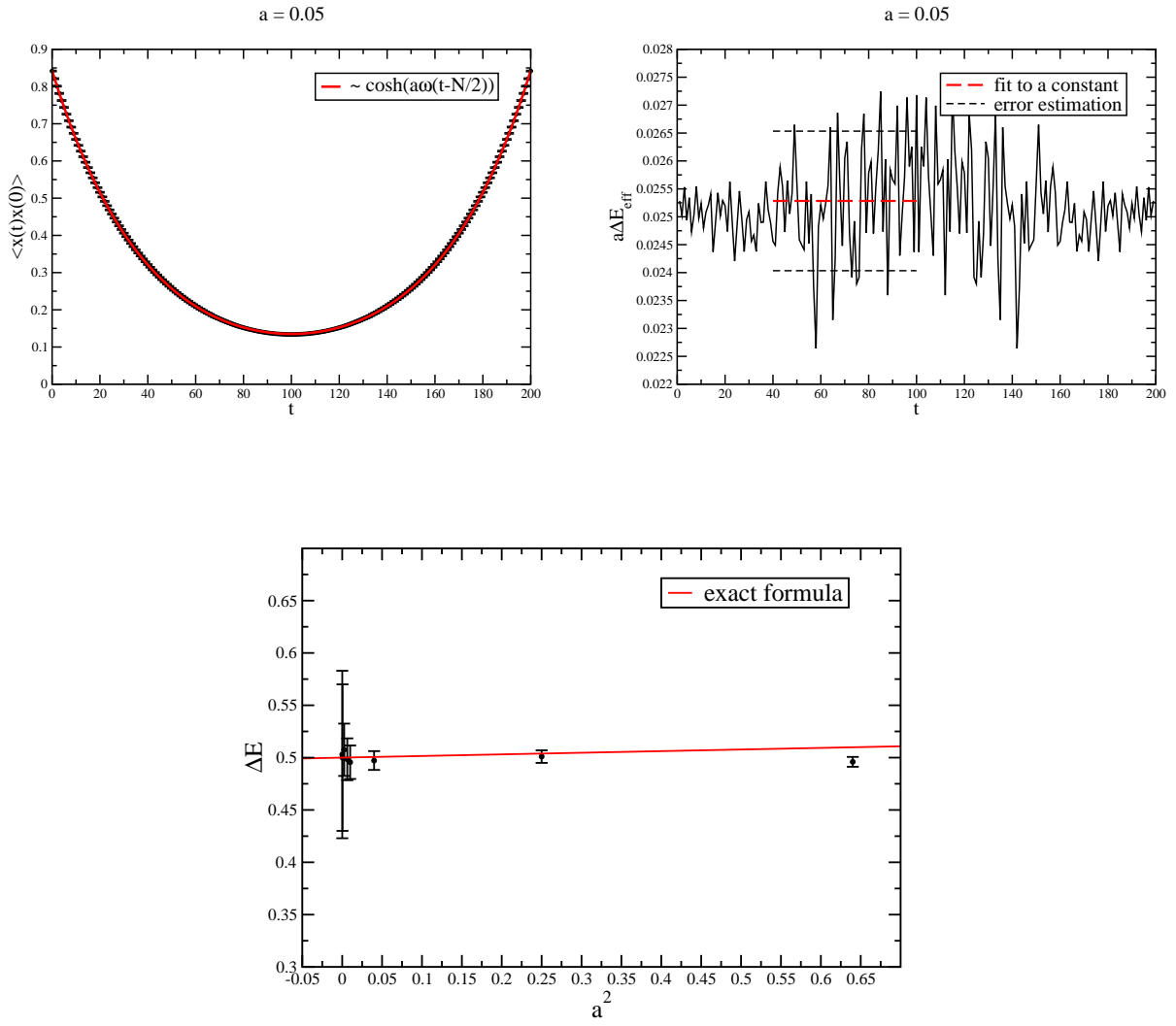


Figure 5: Correlator analysis. We show in successive order an example of the correlator and plateau analysis and the lattice spacing scaling for the energy ΔE .

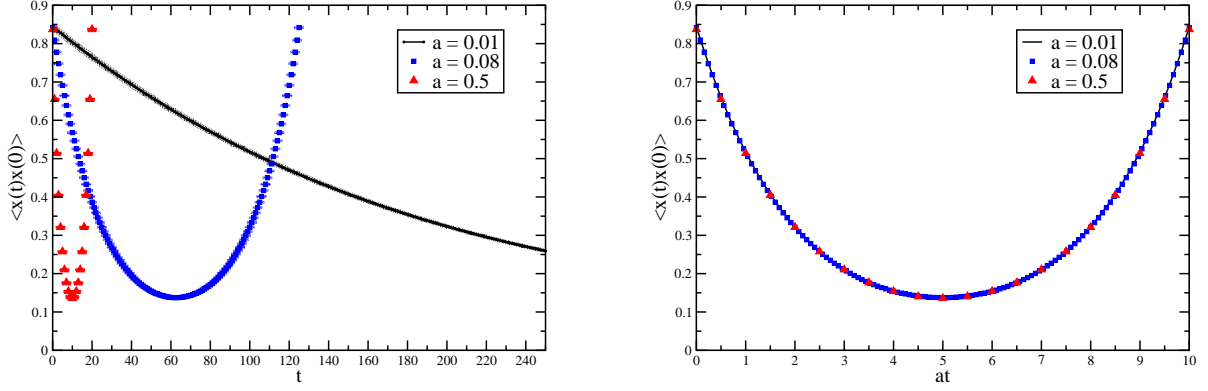


Figure 6: Comparison of the correlator for different lattice spacing. The plot on the left shows time in lattice units (points on the lattice), while the one on the right shows time in physical units ($T = 10$). For clarity, the error bars are shown only in the first plot.

patience and guidance throughout the project in his supervising task. Finally we would like to thank all the people at the summer student program and DESY staff to make possible this great experience at Zeuthen.

A Calculations for the Free Particle path integral

We want to evaluate the position integrals

$$\int_{-\infty}^{+\infty} \left(\frac{2\pi i \hbar a}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar a} [(x_2 - x_1)^2 + (x_1 - x_0)^2]} dx_1 = \left(\frac{2\pi i \hbar (2a)}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar (2a)} (x_2 - x_0)^2} \quad (67)$$

We proceed as follows:

$$e^{-\frac{m}{2i\hbar a} [(x_2 - x_1)^2 + (x_1 - x_0)^2]} dx_1 = e^{-\frac{m}{2i\hbar a} [2x_1^2 - 2x_1(x_2 + x_0) + x_2^2 + x_0^2]} dx_1 = e^{-\frac{m}{i\hbar a} [x^2 - \frac{(x_2 + x_0)^2}{4} + \frac{x_2^2 + x_0^2}{2}]} dx \quad (68)$$

The result for the integration over x_2 that we obtain by completing the square is similar:

$$\int_{-\infty}^{+\infty} \left(\frac{\sqrt{8}\pi i \hbar a}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{4i\hbar a} [2(x_3 - x_2)^2 + (x_2 - x_0)^2]} dx_2 = \left(\frac{2\pi i \hbar (3a)}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar (3a)} (x_3 - x_0)^2} \quad (69)$$

We can generalize this to an induction step for the $(n-1)$ th integration:

$$\begin{aligned} & \int_{-\infty}^{+\infty} \left(\frac{2\sqrt{n-1}\pi i \hbar a}{m} \right)^{-1} e^{\frac{m}{2i\hbar a(n-1)} [nx_{n-1} - 2x_{n-1}(x_0 + (n-1)x_n) + x_0^2 + (n-1)x_n^2]} dx_{n-1} = \\ & = \int_{-\infty}^{+\infty} \left(\frac{2\sqrt{n-1}\pi i \hbar a}{m} \right)^{-1} e^{\frac{nm}{2i\hbar a(n-1)} [(x_{n-1} - \frac{x_0 + (n-1)x_n}{n})^2] - \frac{m}{2i\hbar a(n-1)} [\frac{(x_0 + (n-1)x_n)^2}{n} - x_0^2 - (n-1)x_n^2]} dx_{n-1} = \\ & = \left(\frac{2\pi i \hbar n a}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar a(n-1)n} [-x_0^2 - (n-1)^2 x_n^2 - 2(n-1)x_0 x_n + n x_0^2 + n(n-1)x_n^2]} = \\ & = \left(\frac{2\pi i \hbar n a}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar a(n-1)n} [(n-1)x_0^2 + (n^2 - n - n^2 + 2n - 1)x_n^2 - 2x_0 x_n (n-1)]} = \left(\frac{2\pi i \hbar n a}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar a n} (x_0 - x_n)^2} \quad (70) \end{aligned}$$

So that finally the kernel (because of $Na = T$) after $(N - 1)$ integrations will be:

$$K(x_b, x_a, T) = K(x_N, x_0, T) = \left(\frac{2\pi i \hbar T}{m} \right)^{-\frac{1}{2}} e^{\frac{m}{2i\hbar T} (x_0 - x_N)^2} \quad (71)$$

B Classical action for the Harmonic Oscillator

We consider the classical action

$$S_{cl} = \int_{t_a}^{t_b} dt \frac{m}{2} [\dot{x}^2 - \omega^2 x^2] \quad (72)$$

Here $x(t)$ is the classical trajectory with the boundary conditions $x(0) = x_a$ and $x(T) = x_b$. As $x(t)$ is the classical path, it must be a solution of the classical equation of motion. Thus $x(t)$ is of the form $x(t) = A \cos \omega t + B \sin \omega t$ and with respect to the boundary conditions will be:

$$x(t) = x_a \cos \omega t + \frac{x_b}{\sin \omega T} \sin \omega t - x_a \frac{\cos \omega T}{\sin \omega T} \sin \omega t \quad (73)$$

For the time derivative we find respectively:

$$\dot{x}(t) = -\omega x_a \sin \omega t + \frac{\omega x_b}{\sin \omega T} \cos \omega t - x_a \omega \frac{\cos \omega T}{\sin \omega T} \cos \omega t \quad (74)$$

By squaring and putting the result into the appropriate form, we get:

$$\begin{aligned} \dot{x}(t)^2 - \omega^2 x^2 = & -\omega^2 x_a^2 (\sin^2 - \cos^2 \omega t) + \frac{(x_b - x_a \cos \omega T)^2}{\sin^2 \omega T} \omega^2 (\cos^2 \omega t - \sin^2 \omega t) - \\ & - 4\omega^2 x_a \frac{x_b - x_a \cos \omega T}{\sin \omega T} \cos \omega T \sin \omega t \end{aligned} \quad (75)$$

After integration by parts we see:

$$\int_0^T dt \sin^2 \omega t = - \left[\frac{\sin \omega t \cos \omega t}{\omega} \right]_{t=0}^{t=T} + \int_0^T dt \cos^2 \omega t \quad (76)$$

As well as:

$$\int_0^T dt \sin \omega t \cos \omega t = \left[\frac{\sin^2 \omega t}{\omega} \right]_{t=0}^{t=T} - \int_0^T dt \cos \omega t \sin \omega t = \frac{1}{2\omega} \sin^2 \omega T \quad (77)$$

So in performing the integral in time we find the action

$$S_{cl} = \frac{m}{2} \omega \left[\frac{(x_b - x_a \cos \omega T)^2}{\sin \omega T} \cos \omega T - x_a^2 \sin \omega T \cos \omega T - 2x_a (x_b - x_a \cos \omega T) \sin \omega T \right] \quad (78)$$

which we can simplify to get the final result for the classical action of the Harmonic Oscillator:

$$S_{cl} = \frac{m}{2} \frac{\omega}{\sin \omega T} [(x_a^2 + x_b^2) \cos \omega T - 2x_a x_b] \quad (79)$$

C Calculations for the action on the lattice under Fourier transformations for the Harmonic Oscillator

In order to perform the calculation of the sums we insert the exponentials, i.e. $\sin\left(\frac{\pi n m}{N}\right) = \frac{e^{i\pi n m/N} - e^{-i\pi n m/N}}{2}$. That way, for terms of the form $\sum_{n=1}^{N-1} y_n^2$ we obtain the Fourier transforms:

$$\sum_{n=1}^{N-1} y_n^2 = \sum_{n,m,l} -\frac{1}{4N} (e^{i\pi n m/N} - e^{-i\pi n m/N}) A_m (e^{i\pi n l/N} - e^{-i\pi n l/N}) A_l = \quad (80)$$

$$= \sum_{n,m,l} -\frac{1}{4N} A_m A_l (e^{i\pi n(m+l)/N} + e^{-i\pi n(m+l)/N} - e^{i\pi n(m-l)/N} - e^{-i\pi n(m-l)/N}) =$$

Now we change the summation index $n \rightarrow -n$ in the second sum:

$$= \sum_{m,l} -\frac{1}{4N} A_m A_l \left[\sum_{n=1}^{N-1} (e^{i\pi n(m+l)/N} - e^{i\pi n(l-m)/N}) + \sum_{n=-1}^{1-N} (e^{i\pi n(m+l)/N} - e^{i\pi n(l-m)/N}) \right] =$$

If we now shift the second sum by $2N$ we get

$$\begin{aligned} &= \sum_{m,l} -\frac{1}{4N} A_m A_l \left[\sum_{n=1}^{N-1} (e^{i\pi n(m+l)/N} - e^{i\pi n(l-m)/N}) + \sum_{n=2N-1}^{N+1} (e^{i\pi n(m+l)/N} - e^{i\pi n(l-m)/N}) \right] = \\ &= \sum_{m,l} -\frac{1}{4N} A_m A_l \left[\sum_{n=1}^{2N-1} (e^{i2\pi n(m+l)/2N} - e^{i2\pi n(l-m)/2N}) \right] = \sum_{m,l} -\frac{1}{4N} A_m A_l [-2N \delta_{lm}] \end{aligned}$$

where we have used $\sum_{n=1}^{N-1} e^{i2\pi n(m-l)/N} = N \delta_{ml}$ to finally obtain the result:

$$\sum_{n=0}^{N-1} y_n^2 = \sum_n \frac{1}{2} A_n^2 \quad (81)$$

We follow the same procedure for the sum $\sum_{n=1}^{N-1} y_{n+1}^2$:

$$\begin{aligned} \sum_{n=1}^{N-1} y_{n+1}^2 &= \sum_{n,m,l} -\frac{1}{4N} A_m A_l (e^{i\pi(n+1)(m+l)/N} + e^{-i\pi(n+1)(m+l)/N} - \\ &\quad - e^{i\pi(n+1)(m-l)/N} - e^{-i\pi(n+1)(m-l)/N}) = \end{aligned} \quad (82)$$

After we change the summation index $n+1 \rightarrow -n$ in the second sum, we shift it by $2N$ to get:

$$= \sum_{m,l} -\frac{1}{4N} A_m A_l \left[\sum_{n=2}^N (e^{i\pi n(m+l)/N} - e^{i\pi n(l-m)/N}) + \sum_{n=2N-2}^N (e^{i\pi n(m+l)/N} - e^{i\pi n(l-m)/N}) \right] =$$

Now we contract into one sum:

$$\begin{aligned} &= \sum_{m,l} -\frac{1}{4N} A_m A_l \left[\sum_{n=1}^{2N-1} (e^{i2\pi n(m+l)/2N} - e^{i2\pi n(l-m)/2N}) - \right. \\ &\quad \left. - (e^{i2\pi(m+l)/2N} - e^{i2\pi(l-m)/2N}) - (e^{i2\pi(2N-1)(m+l)/2N} - e^{i2\pi(2N-1)(l-m)/2N}) \right] \end{aligned}$$

where we have subtracted the extra summands we have added to the sum, in order to obtain (again with $\sum_{n=1}^{N-1} e^{i2\pi n(m-l)/N} = N \delta_{ml}$)

$$\sum_{n=0}^{N-1} y_{n+1}^2 = \sum_n \frac{1}{2} A_n^2 (1 + C(n)/N) \quad (83)$$

which in the desired limit $N \rightarrow \infty$ reads the same as the sum discussed above.

Finally we have to consider the mixed sum $\sum_{n=1}^{N-1} (y_{n+1} - y_n)^2$. We will execute this calculation in all steps, but one may skip the minor details. First of all we insert the Fourier transforms:

$$\begin{aligned} \sum_{n=1}^{N-1} (y_{n+1} - y_n)^2 &= \sum_{n,m,l} -\frac{1}{4N} (e^{i\pi(n+1)m/N} - e^{-i\pi(n+1)m/N} - e^{i\pi nm/N} + e^{-i\pi nm/N}) A_m \times \\ &\quad \times (e^{i\pi(n+1)l/N} - e^{-i\pi(n+1)l/N} - e^{i\pi nl/N} + e^{-i\pi nl/N}) A_l = \end{aligned} \quad (84)$$

Then we factorize and multiply out to get:

$$\begin{aligned}
&= \sum_{n,m,l} -\frac{1}{4N} A_m A_l [e^{i\pi n m/N} (e^{i\pi m/N} - 1) - e^{-i\pi n m/N} (e^{-i\pi m/N} - 1)] \times \\
&\quad \times [e^{i\pi n l/N} (e^{i\pi l/N} - 1) - e^{-i\pi n l/N} (e^{-i\pi l/N} - 1)] = \\
&= \sum_{n,m,l} -\frac{1}{4N} A_m A_l [e^{i\pi n(l+m)/N} (e^{i\pi l/N} - 1)(e^{i\pi m/N} - 1) + e^{-i\pi n(l+m)/N} (e^{-i\pi l/N} - 1)(e^{-i\pi m/N} - 1) - \\
&\quad - e^{i\pi n(m-l)/N} (e^{-i\pi l/N} - 1)(e^{i\pi m/N} - 1) - e^{-i\pi n(m-l)/N} (e^{i\pi l/N} - 1)(e^{-i\pi m/N} - 1)] =
\end{aligned}$$

This we can as well write as:

$$\begin{aligned}
&= \sum_{m,l} -\frac{1}{4N} A_m A_l \left[\sum_{n=1}^{N-1} e^{i\pi(n+1)(l+m)/N} (1 - e^{-i\pi l/N})(1 - e^{-i\pi m/N}) + \sum_{n=1}^{N-1} e^{-i\pi n(l+m)/N} (e^{-i\pi l/N} - 1) \times \right. \\
&\quad \times (e^{-i\pi m/N} - 1) - \sum_{n=1}^{N-1} e^{i\pi(n+1)(m-l)/N} (1 - e^{i\pi l/N})(1 - e^{-i\pi m/N}) - \\
&\quad \left. - \sum_{n=1}^{N-1} e^{-i\pi n(m-l)/N} (e^{i\pi l/N} - 1)(e^{-i\pi m/N} - 1) \right] =
\end{aligned}$$

Now we can make the appropriate changes of summation indices, i.e. $n+1 \rightarrow n$ and $-n \rightarrow n$

$$\begin{aligned}
&= \sum_{m,l} -\frac{1}{4N} A_m A_l \left[\left(\sum_{n=2}^N + \sum_{n=-1}^{1-N} \right) e^{i\pi n(l+m)/N} (e^{-i\pi l/N} - 1)(e^{-i\pi m/N} - 1) - \right. \\
&\quad \left. - \left(\sum_{n=2}^N + \sum_{n=-1}^{1-N} \right) e^{i\pi n(m-l)/N} (e^{i\pi l/N} - 1)(e^{-i\pi m/N} - 1) \right] =
\end{aligned}$$

and again shift the sum by $2N$, that means here we have $(\sum_{n=2}^N + \sum_{n=-1}^{1-N}) = (\sum_{n=2}^N + \sum_{n=2N-1}^{N+1})$ to get (where we again subtract the extra terms for $n=1$, which we have added to the sum):

$$\begin{aligned}
&= \sum_{m,l} -\frac{1}{4N} A_m A_l \left[\sum_{n=1}^{2N-1} (e^{i\pi n(l+m)/N} (e^{-i\pi l/N} - 1)(e^{-i\pi m/N} - 1) - e^{i\pi n(m-l)/N} (e^{i\pi l/N} - 1)(e^{-i\pi m/N} - 1)) - \right. \\
&\quad \left. - e^{i\pi(l+m)/N} (e^{-i\pi l/N} - 1)(e^{-i\pi m/N} - 1) + e^{i\pi(m-l)/N} (e^{i\pi l/N} - 1)(e^{-i\pi m/N} - 1) \right]
\end{aligned}$$

At this point we can again explicitly calculate the sum over n to get the δ -functions and neglect extra terms of the order $1/N$. Thus we overall find the result for the mixed sum, which is:

$$\begin{aligned}
\sum_{n=1}^{N-1} (y_{n+1} - y_n)^2 &= \sum_n \frac{A_n^2}{2} (e^{-i\pi n/N} - 1)(e^{+i\pi n/N} - 1) - \sum_{m,l} \frac{A_m A_l}{4N} C(m,l) \\
\sum_{n=1}^{N-1} (y_{n+1} - y_n)^2 &\rightarrow \sum_n \frac{A_n^2}{2} (2 - (e^{-i\pi n/N} + e^{+i\pi n/N})) = \sum_n A_n^2 2 \sin^2 \left(\frac{\pi m}{2N} \right) \tag{85}
\end{aligned}$$

References

- [1] R.P. Feynman and A.R. Hibbs, *Quantum Mechanics and Path Integrals*, McGraw-Hill, New York (1965)
- [2] M. Creutz and B. Freedman, *Annals of Physics* **132**, p. 427, (1981)
- [3] M.E. Peskin and D.V. Schröder, *Quantum Field Theory*, Perseus, Cambridge, Massachusetts (1995)