NST Part III ETP Project

A Quantum Field Theory Simulator

Candidate Number: 8259X

Supervisor: Dr CG Lester

13th May 2013

Abstract

A Quantum Field Theory simulator and a codebase of utilities were developed for use as an educational tool. The simulator models a single real, massive, self-interacting scalar field in a discretised one-dimensional space. The Leapfrog integration method was used to model time development. The variable limits of the simulator were determined through investigating the conservation of energy and probability. The simulations conserved these quantities to better than one part in 10^{10} for an optimised set of system variables. The Power Iteration method was used to determine the lowest eigenvalues of the calculated Hamiltonian matrix and their corresponding eigenstates. These were used to partially validate the simulations by verifying the equivalence of a second-order self-interacting term in the system Lagrangian with a change in mass. This equivalence was established to better than one part in 10^{9} for the eigenstates with the second and third lowest eigenvalues.

Contents

References are denoted by a superscript number, appendices by a superscript letter.

Word count, excluding headings, tables and captions: 4926.

For simulator code, visit https://github.com/charlie-bridge/QFT

For codebase, visit https://github.com/carlandreaslindstrom/QFT-Simulations

---

1 – Introduction

Quantum Field Theory (QFT) forms the theoretical framework of the Standard Model and many Grand Unified Theories. It is fundamental to modern physics. Consequently, it is important that it is taught to new generations of scientists effectively and reliably. A simple to use, distributable QFT Simulator which is appropriate for students does not currently exist. The motivation behind this project was to develop such a simulator and test its functionality. In addition, it was hoped that this development would pose many interesting physical questions. The Falstad Maths, Physics and Engineering Applets[1] provided inspiration for the general simulator operation, as did Dr Chris Lester's version of a QFT Simulator[2].

The simulator considers a single real, massive scalar field. Currently the only field in the standard model of this type is that which gives rise to the Higgs boson[3 4]. Furthermore, the only interactions modelled are self-interactions as only one field is considered. Despite these simplifications the simulator is still capable of demonstrating effects unique to QFT. By comparing the simulator's results with expected phenomena there is also the potential to analyse the numerical methods it utilises.

In Section 2 we discuss the assumptions made by the simulator and in Section 3 we outline the theory it requires. Section 4 explores the details of how this theory was implemented and any code-related decisions made. In Section 5 we look at the testing performed to set the variable limits for the simulator and to verify its functionality.

2 – Assumptions

2.1 Discretisation

In order to compute a field's development in time rather than computing scattering cross-sections it is necessary to limit the computation in some way. The discretisation of space into N points leads to a finite number of momentum modes[A] of the system, which are eigenstates of the free Hamiltonian. Once this basis has been established a further truncation is required as the number of particles in the system is not limited (scalar fields are bosonic[5]). How this truncation is implemented depends on the way in which the basis states are labelled (see Section 3.5). The discretisation of time is required for an integration method to be implemented using the Schrödinger equation.

2.2 One-dimensional ring

To simplify the simulator, it was decided that the system should consist of a single spatial dimension. This also enables larger numbers of states to be considered in a real-time simulation, potentially improving its accuracy. To avoid being forced to consider boundary conditions it was decided that the system should form a loop (see Figure 1).

2.3 Units

We will use natural units ($\hbar = c = 1$) to simplify the equations required by the simulator. From the dimensionality of the Lagrangian we can establish the dimensions of the field. The action is dimensionless in natural units and so the Lagrangian has units of energy and the Lagrangian density units of energy squared. By examining the form of the Lagrangian density shown later in Section 3.1 we find that the field must be dimensionless. This allows us to define the dimension scale of the mass (equivalent to that of inverse time and inverse space) without affecting the dynamics of the field. We arbitrarily set this scale to giga-electronvolts (GeV) as this is a scale familiar to particle physicists.

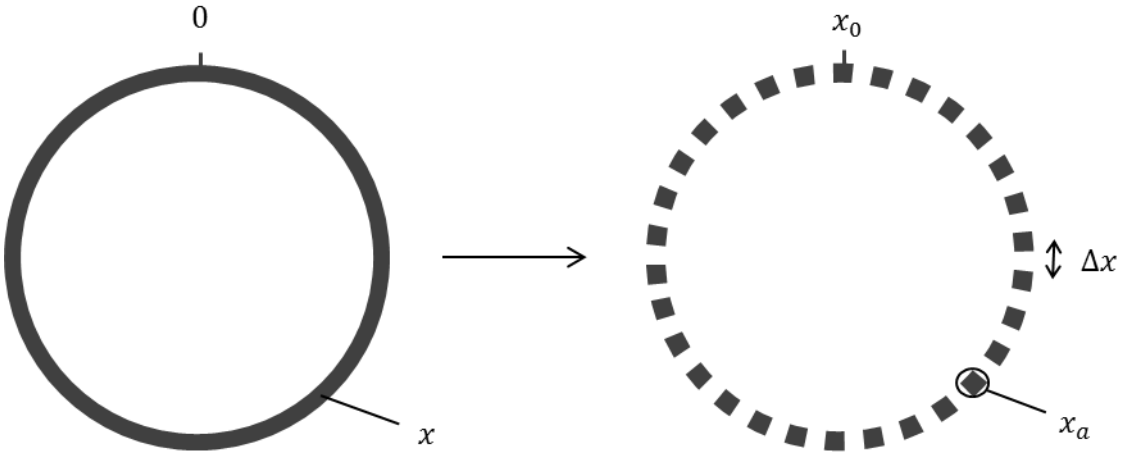


Figure 1: A two-dimensional visualisation of a discretised one-dimensional loop

3 – Theory

3.1 Lagrangian

The development of the equations used to code the simulator closely follows that of normal QFT[6], with the major differences arising as a result of the discretisation of space. We begin with the Lagrangian (L) for a real scalar field (ϕ) with mass m in one spatial dimension (x) and no interactions:

$$\mathrm{L} = \int \mathcal{L}\, dx = \frac{1}{2}\int \left(\dot{\phi}^2 - (\nabla\phi)^2 - m^2\phi^2\right)dx \tag{1}$$

Where $\dot{}$ and ∇ denote differentiation with respect to time and x respectively, and $\mathcal{L}$ is the Lagrangian density. Introducing the concept of discrete spatial points rather than a continuum has a direct effect on our concepts of fields, integration and differentiation (Equations 2, 3 and later, 16). We use these to rewrite the Lagrangian so that it is appropriate for our system:

$$\int dx \rightarrow \sum_n \Delta x \tag{2}$$

$$\phi(x) \rightarrow \phi_x \tag{3}$$

$$\mathrm{L} = \sum_n \mathcal{L}_n \Delta x = \frac{1}{2}\Delta x \sum_a \left(\dot{\phi}_{x_a}{}^2 - \left(\nabla\phi_{x_a}\right)^2 - m^2{\phi_{x_a}}^2\right) \tag{4}$$

Here $\mathcal{L}_n$ is the equivalent of the Lagrangian density for the discretised system and Δx is the distance between the spatial points (Figure 1). The subscript on the fields replaces the label x used in the continuous case. The single field with infinitely many degrees of freedom has been replaced by N fields, each with one degree of freedom.

The fields' conjugate momenta (π_n, Equation 5) enable us to write the Hamiltonian (H) for this system[6], which we will call the Free Hamiltonian due to the absence of interactions (Equation 7).

$$\pi_n = \frac{\partial \mathrm{L}}{\partial \dot{\phi}_n} = \dot{\phi}_n \Delta x \tag{5}$$

$$\mathcal{H}_n = \pi_n \dot{\phi}_n - \mathcal{L}_n \tag{6}$$

$$\mathrm{H} = \sum_n \mathcal{H}_n \Delta x = \frac{1}{2}\Delta x \sum_a \left({\pi_{x_a}}^2 + \left(\nabla \phi_{x_a}\right)^2 + m^2 {\phi_{x_a}}^2 \right) \tag{7}$$

Here $\mathcal{H}_n$ is the equivalent of the Hamiltonian density.

Following normal QFT we undergo canonical quantisation[5]. The commutation relations required (Equation 10) are similar to the usual ones (Equation 9). The definition of the Dirac delta function must change in order to preserve its properties (Equation 8).

$$\delta(x - y) \to \frac{1}{\Delta x}\delta_{xy} \tag{8}$$

$$\left[\hat{\phi}(x), \hat{\pi}(y)\right] = i\delta(x - y) \tag{9}$$

$$\left[\hat{\phi}_{x_a}, \hat{\pi}_{x_b}\right] = \frac{i}{\Delta x}\delta_{x_a x_b} = \frac{i}{\Delta x}\delta_{ab} \tag{10}$$

Still following the methods used in standard QFT, we would like to use a Fourier expansion of the fields and their conjugates in terms of some new operators $\hat{a}_p$ and ${\hat{a}_p}^\dagger$ in the hope that they can later be identified as creation and annihilation operators[6]. If we assume that the normal commutation relations for these operators transform as those shown above (Equations 11 and 12), we find that Equation 10 is recovered if we use the expansions shown in Equations 13 and 14[B].

$$\left[\hat{a}_p, \hat{a}_q^\dagger\right] = (2\pi)\left(2\omega_p\right)\delta(p - q) \tag{11}$$

$$\left[\hat{a}_{p_a}, \hat{a}_{p_b}^\dagger\right] = \frac{(2\pi)\left(2\omega_{p_a}\right)}{\Delta p}\delta_{p_a p_b} = 2L\omega_{p_a}\delta_{p_a p_b} = 2L\omega_{p_a}\delta_{ab} \tag{12}$$

$$\hat{\phi}_{x_a} = \frac{1}{L}\sum_n \frac{1}{2\omega_{p_n}}\left(\hat{a}_{p_n} e^{-ip_n x_a} + \hat{a}_{p_n}^\dagger e^{ip_n x_a}\right) \tag{13}$$

$$\hat{\pi}_{x_a} = -\frac{i}{L}\sum_n \frac{1}{2}\left(\hat{a}_{p_n} e^{-ip_n x_a} - \hat{a}_{p_n}^\dagger e^{ip_n x_a}\right) \tag{14}$$

3.2 Energy-momentum relation

In order to determine if the roles of $\hat{a}_p$ and $\hat{a}_p^\dagger$ are as expected we need to formulate the energy-momentum relation for the system, which differs from the standard relation. The Euler-Lagrange equations enable us to write:

$$\ddot{\phi}_{x_a} - \nabla^2 \phi_{x_a} + m^2 {\phi_{x_a}}^2 = 0 \tag{15}$$

We now assume that the discrete spatial points are very close together (that Δx is small) so the derivative of the field can be approximated in two equivalent ways:

$$\nabla\phi_{x_a} \simeq \frac{\phi_{x_{a+1}} - \phi_{x_a}}{\Delta x} \simeq \frac{\phi_{x_a} - \phi_{x_{a-1}}}{\Delta x} \tag{16}$$

This enables us to rewrite Equation 15 as[C]:

$$\ddot{\phi}_{x_a} - \frac{1}{\Delta x^2}\left(\phi_{x_{a+1}} - 2\phi_{x_a} + \phi_{x_{a-1}}\right) + m^2{\phi_{x_a}}^2 = 0 \tag{17}$$

Trying a travelling wave solution (Equation 18) leads to the energy-momentum relation (Equation 19)[C]:

$$\phi_{x_a} \propto e^{i\left(p_b x_a + \omega_{p_b} t\right)} \tag{18}$$

$${\omega_{p_b}}^2 = m^2 + \frac{4}{\Delta x^2}\left(\sin\left(\frac{b\pi}{N}\right)\right)^2 \tag{19}$$

This tends to the standard relation in the limit of $\Delta x \to 0$ or, equivalently, $N \to \infty$[C].

3.3 Fock state basis

We can now write the Hamiltonian in a much more succinct form; substituting Equations 13, 14 and 16 into Equation 7 we obtain[D]:

$$\mathrm{H} = \sum_n \left(\frac{\hat{a}^\dagger_{p_n}\hat{a}_{p_n}}{2L} + \frac{\omega_{p_n}}{2}\right) \to \sum_n \frac{\hat{a}^\dagger_{p_n}\hat{a}_{p_n}}{2L} \tag{20}$$

In a process mirroring renormalisation, we subtract the large, constant "zero-point" energy from the Hamiltonian. The commutation relations between H and $\hat{a}_p$ and between H and $\hat{a}^\dagger_p$ confirm that they can be considered as creation and annihilation operators:

$$\left[\mathrm{H}, \hat{a}_{p_m}\right] = -\omega_{p_m}\hat{a}_{p_m} \tag{21}$$

$$\left[\mathrm{H}, \hat{a}^\dagger_{p_m}\right] = \omega_{p_m}\hat{a}^\dagger_{p_m} \tag{22}$$

Now all that remains is to determine how to label the eigenstates of the Free Hamiltonian, which form a Fock space[7]. With our interpretation of $\hat{a}_p$ and $\hat{a}^\dagger_p$ we can write a general state as:

$$|p\rangle \propto \frac{\left(\hat{a}^\dagger_{p_0}\right)^{n_0}\left(\hat{a}^\dagger_{p_1}\right)^{n_1} \dots \left(\hat{a}^\dagger_{p_{N-1}}\right)^{n_{N-1}}}{\sqrt{(n_0!)(n_1!)\dots(n_{N-1}!)}}|0\rangle \tag{23}$$

These states are orthogonal[E], and we can choose the normalisation constant so that they are orthonormal. The resulting states are shown in Equation 24 and have eigenvalues under the Free Hamiltonian given by Equation 25[F]:

$$|p\rangle = \frac{\left(\hat{a}^{\dagger}_{p_0}\right)^{n_0}\left(\hat{a}^{\dagger}_{p_1}\right)^{n_1}\dots\left(\hat{a}^{\dagger}_{p_{N-1}}\right)^{n_{N-1}}}{\sqrt{\left(2\omega_{p_0}\right)^{n_0}\left(2\omega_{p_1}\right)^{n_1}\dots\left(2\omega_{p_{N-1}}\right)^{n_{N-1}}}\sqrt{L^{n_0}L^{n_1}\dots L^{n_{N-1}}}\sqrt{(n_0!)(n_1!)\dots(n_{N-1}!)}}|0\rangle \tag{24}$$

$$E_p = \sum_a n_a \omega_{p_a} \tag{25}$$

This completes the necessary theory required by the simulator for non-interacting systems. Before we investigate interacting systems it is useful to compute the effect of the creation and annihilation operators on general states[G] (Equations 26 and 27). Here $|\alpha \pm p_a\rangle$ denotes the original state with one more/less particle in mode a.

$$\hat{a}_{p_a}|\alpha\rangle = \sqrt{2n_a L\omega_{p_b}}|\alpha - p_a\rangle \tag{26}$$

$$\hat{a}^{\dagger}_{p_a}|\alpha\rangle = \sqrt{2(n_a+1)L\omega_{p_b}}|\alpha + p_a\rangle \tag{27}$$

3.4 Interactions

The dynamics of a system with a Lagrangian that leads to the Free Hamiltonian are unremarkable[5]. It is possible to include additional terms in the Lagrangian of the form shown in Equation 28. Equation 29 shows these terms after discretisation.

$$\mathrm{L}_I = -\int \frac{\lambda_y}{y!}\phi^y dx \tag{28}$$

$$\mathrm{L}_I = -\frac{\lambda_y \Delta x}{y!}\sum_a {\phi_{x_a}}^y \tag{29}$$

These do not depend on the time derivative of the field and so do not affect the conjugate momenta. We can use our earlier expansion (Equation 13) to rewrite this interaction term. The change in sign occurs as we are now considering the change in the Hamiltonian:

$$\mathrm{H}_I = \frac{\lambda_y \Delta x}{(2L)^y y!}\sum_{a,q,r\dots}\frac{1}{\omega_{p_q}\omega_{p_r}\dots}\left(\hat{a}_{p_q}e^{-ip_q x_a} + \hat{a}^{\dagger}_{p_q}e^{ip_q x_a}\right)\left(\hat{a}_{p_r}e^{-ip_r x_a} + \hat{a}^{\dagger}_{p_r}e^{ip_r x_a}\right)\dots \tag{30}$$

Performing the sum over the spatial labels we find that momentum is conserved by the interaction terms[D]. For $y = 3$, for example, we can write our Interaction Hamiltonian as:

$$\mathrm{H}_I = \frac{\lambda_3}{48L^2}\sum_{a,q,r}\frac{1}{\omega_{p_q}\omega_{p_r}\omega_{p_{-q-r}}}\left(\hat{a}_{p_q} + \hat{a}^{\dagger}_{p_q}\right)\left(\hat{a}_{p_r} + \hat{a}^{\dagger}_{p_r}\right)\left(\hat{a}_{p_{-q-r}} + \hat{a}^{\dagger}_{p_{-q-r}}\right) \tag{31}$$

Multiple interaction terms can be added to the Lagrangian, each with unique values of y and λ_y. Their effects on the Full Hamiltonian are additive.

3.5 Time development

The time dependence of states is determined by the Schrödinger equation. If we label a general state as a linear superposition of our basis states $|s\rangle$ with time-dependent coefficients $c_s(t)$ this gives:

$$\dot{c}_s(t) = \sum_p c_t(t)\langle s|\mathrm{H}|p\rangle \tag{32}$$

This is the point at which the simulator's operation departs from the usual QFT process. We now have a basis for evaluating the Full Hamiltonian in matrix form and an equation describing the time dependence of a general state. We can, working in the Schrödinger picture, evaluate the time development of that general state using an integration method. The only remaining requirement is a method for truncating the number of eigenstates of the Free Hamiltonian that we will use as our basis.

3.6 Basis truncation

A member of our Free Hamiltonian eigenstate basis can be represented by a list of numbers- $[n_0, n_{-1}, n_1, n_{-2}, n_2 \dots]$ - which correspond to the number of particles in each momentum mode that the state contains. Here, the subscript number refers to the size of the mode's momentum and the sign refers to the direction of that momentum[A]. A simple way of arranging the states so that a truncation method emerges would be (the left column indicates this ordering with a number):

$$\begin{array}{ccc} |0\rangle & \rightarrow & [0,0,0,0\dots] \\ |1\rangle & \rightarrow & [1,0,0,0\dots] \\ |2\rangle & \rightarrow & [0,1,0,0\dots] \\ |N\rangle & \rightarrow & [0,0,0,0\dots 1] \\ |N+1\rangle & \rightarrow & [2,0,0,0\dots] \\ |N+2\rangle & \rightarrow & [1,1,0,0\dots] \\ |N+3\rangle & \rightarrow & [1,0,1,0\dots] \\ |2N+1\rangle & \rightarrow & [0,2,0,0\dots] \end{array} \tag{33}$$

This corresponds to ordering the states primarily by number of particles and arbitrarily within each band of particle number. Truncation at a certain number of particles arises, on which the number of states considered is highly dependent[H]. This method prioritises lower numbers of particles, a valid reason assuming their masses greatly exceed their momenta.

An alternative method for labelling the states is:

$$\begin{array}{ccc} |0\rangle & \rightarrow & [0,0,0,0\dots] \\ |1\rangle & \rightarrow & [1,0,0,0\dots] \\ |2\rangle & \rightarrow & [0,1,0,0\dots] \\ |3\rangle & \rightarrow & [2,0,0,0\dots] \\ |4\rangle & \rightarrow & [1,1,0,0\dots] \\ |5\rangle & \rightarrow & [0,0,1,0\dots] \\ |6\rangle & \rightarrow & [3,0,0,0\dots] \\ |7\rangle & \rightarrow & [2,1,0,0\dots] \end{array} \tag{34}$$

The ordering here favours a combination of low momentum and low particle number, and so may have greater general applicability. If lower mass particles are considered then it is necessary to consider many of them, as the energy cost associated with their production is low. For low masses, then, ordering the states according to particle number is definitely not appropriate.

Equations 26 and 27 show us the effect of creation and annihilation operators on a general state. Combined with Equation 30 this gives an estimate of the size of terms in the Interaction Hamiltonian matrix. As we will consider initial states of low energy, such as zero particles or several low-momentum particles, the state ordering method should ideally prioritise the Fock states based upon the size of their interaction with these low energy-states. Considering the suppression of the interaction terms according to energy in Equation 30, this approximately corresponds to an ordering in terms of energy. It is suggested that this matches physically intuitive expectations.

With the aim of investigating a broad variety of systems, this second labelling method was adopted.

3.7 Relativistic effects

The normalisation of the basis states includes factors of energy (ω_{p_b}) which are required as a result of the Lorentz invariant commutation relations (Equation 12). To validate the necessity of considering relativistic effects we consider the maximum momentum of a single particle in our system. We find that the criterion on the system variables to satisfy a non-relativistic approximation is $m\Delta x \gg 2\pi$[I]. We can see that, for fixed mass, there is a conflict between remaining non-relativistic and improving the validity of our expansion of the spatial derivative. To avoid this conflict, the Lorentz invariant commutation relation is used and its effects can be seen in the sections above.

3.8 Finding eigenstates

The Power Iteration method can be used to find the eigenvector of a matrix corresponding to the largest eigenvalue[8]. This method is suitable for use within our framework as it only requires repeated matrix application. This algorithm can also be used to find the eigenvector (v_0) corresponding to the *smallest* eigenvalue (Equations 35 and 36). The value of τ is constrained by Equation 37, where α_{max} is the largest eigenvalue of the Full Hamiltonian[J].

$$x_{n+1} = \frac{(\tau I - \mathrm{H})\mathrm{x}_n}{\|(\tau I - \mathrm{H})\mathrm{x}_n\|} \tag{35}$$

$$v_0 = \lim_{n\to\infty} x_n \tag{36}$$

$$\tau > \alpha_{max} \tag{37}$$

We use the fact that all the eigenvectors of a matrix are also eigenvectors of the identity matrix with eigenvalue 1. The constraint ensures the method converges as fast as possible for our system. We calculate α_{max} using the Power Iteration method on the Full Hamiltonian alone.

Once the ground state has been found the eigenvector for the next lowest eigenvalue can also be found by projecting out the ground state component of the vector in the algorithm at each step. This whole process can be repeated to find more and more eigenvectors. The algorithm is less efficient for degenerate eigenstates[9] (for example, one particle with some momentum either "forwards" or

"backwards"). The initial vectors can be chosen to aid convergence if approximations for the eigenvectors are known, but should be otherwise be randomised.

4 – Implementation

4.1 Programming language

Java was chosen as the language for the simulator and its framework owing to the ease of distributing Java web applets. Java is suited to collaborative efforts, the relevant details of which are described below.

4.2 Shared codebase

This was project was undertaken in parallel with a fellow Part III student. While the physically relevant simulation is handled separately, a shared codebase was developed to facilitate later distribution of both programs. The codebase consists of a graphical user interface and plotting framework as well as general mathematical tools that would be needed by any simulator. It contains interfaces that ensure both students' realisations of the simulator can function within the framework[Q]. After initial independent exploration of how a simulator might function the details of this shared codebase were established together. We decided to base this framework on the other student's implementation.

4.3 Code structure

The simulator operates according to a simple general process:

1. Calculate the system Hamiltonian
2. Use the Schrödinger equation and an integration method to calculate the system state after a small time step
3. Plot the system state in momentum and position space
4. Repeat steps 2 and 3

The time taken to perform the first step is heavily dependent on a number of factors (see section 4.5), while the later steps scale straightforwardly. As the Fourier transform in step 3 can be slow for large N, and is not essential in every iteration, the simulation can be altered to repeat it only after a certain number of time steps.

4.4 Fock state labelling

Members of our basis are represented by a FockState class[Q]. To facilitate the Full Hamiltonian calculation it is useful to have a method for converting the unique list of numbers associated with a Fock state into a single number which we shall call its index. For this process to be useful it should be reversible, and the numbers generated should preferably be ordered as described in Section 3.5. Fortunately, it is possible to recursively implement Cantor's Pairing Function[10] to develop an $O(N)$ method for calculating this index, as well as a method for the reverse process[K]. It was hoped that this state ordering would correspond more closely to an increasing energy order than the alternative method described. The validity of this assumption is explored in Section 5.1.

4.5 Hamiltonian calculation

The Full Hamiltonian is calculated in several parts. The Free Hamiltonian is sufficiently simple for it to not require a class (it is a diagonal matrix). The Interaction Hamiltonians, the number of which is

specified by the number of non-zero coupling strengths, are instances of the InteractionHamiltonian class. This class utilises the orthogonality of the Fock states to reduce the number of calculations it performs. For a given row of the Interaction Hamiltonian matrix (specified by a Fock state) there is only one Fock state or no Fock states for which each term in Equation 30 yields a number in a column. The potential column (represented by a Fock state) is calculated by inversely applying the creation and annihilation operators to the row Fock state. The calculated state may be rendered invalid by the application of an annihilation operator to the vacuum state. The methods for converting Fock states to indices and vice versa are essential for this calculation process.

The number of terms in Equation 30 depends heavily on the power of the interaction, y. However, the code was written in a sufficiently abstract manner such that an arbitrary power can be used. The maximum number of values calculated per row can be deduced from this expansion as $2^y N^{y-1} S$, where S is the number of basis states considered.

While this might appear to show that our matrix is not necessarily sparse, once momentum conservation is taken into account the number of non-zero values in the matrix is greatly reduced. That Fock states only interact with Fock states of equivalent momentum is a heavy restriction.

4.6 Implementing the basis truncation

Using the labelling method described in Section 4.4, the number of Fock states considered can simply be set to any value desired. To try to remain consistent with state selection the size of the "triangle" of states^K produced by this method is selected instead of a specific number of states.

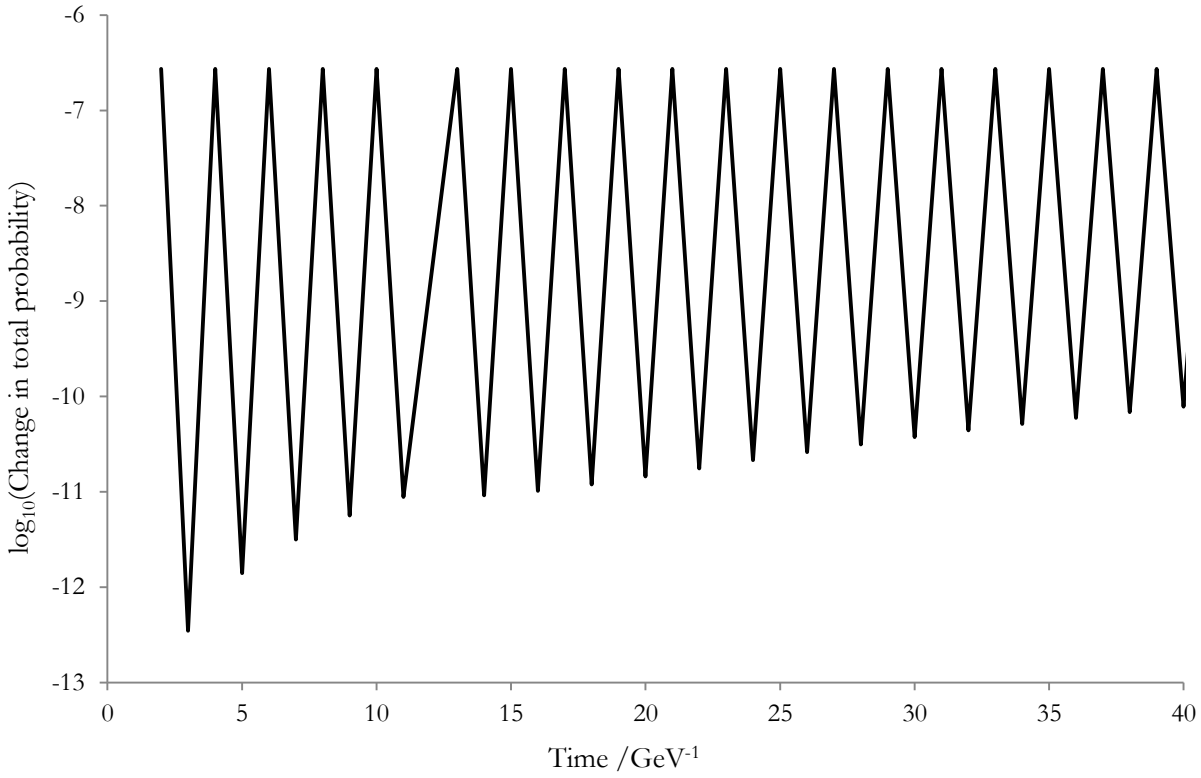
4.7 Integration

Multiple integrators were considered for use in the simulator. After initial testing, the First Order Euler and Semi-Implicit Euler methods[11] were consistently outperformed by the Leapfrog method[11]. This method appears to behave symplectically[11] (Graph 1). It is known that the Leapfrog method is symplectic when used in conjunction with Hamiltonian systems that conserve the two-form $dp \wedge dq$, where p and q are the momentum and position coordinates of the system[11]. This implies that the calculated Hamiltonian meets this same criterion, despite the fact that position and momentum coordinates have been demoted to labels. The details of this are beyond the scope of this report, the only relevant quality being the apparent symplectic behaviour.

4.8 Plotting

To display the dynamical behaviour of the system the simulator plots the coefficients of the zero, one and two particle components. There is only one possible zero particle component, the modulus of which determines the height of a bar and the phase of which determines the bar's colour. The N one particle components are plotted in a similar way, corresponding to the N momentum modes. The two particle states are shown in a plot with axes for the mode of each particle. This plot is necessarily symmetrical about the line of equivalent modes as the particles are indistinguishable. In lieu of bar height the squares representing two particle components have a brightness determined by component modulus.

To supplement the momentum-space plotting of the general state, Fourier transforms of the data described above show the probabilities associated with measuring zero, one or two particles at positions in the system.



Graph 1 – the change in total probability over time for a system with $[N, m, \Delta x, \Delta t, \lambda_2, \lambda_3] = [32, 1GeV, 1GeV^{-1}, 10^{-4}GeV^{-1}, 1GeV^2, 1GeV^2]$

5 – Testing the simulator

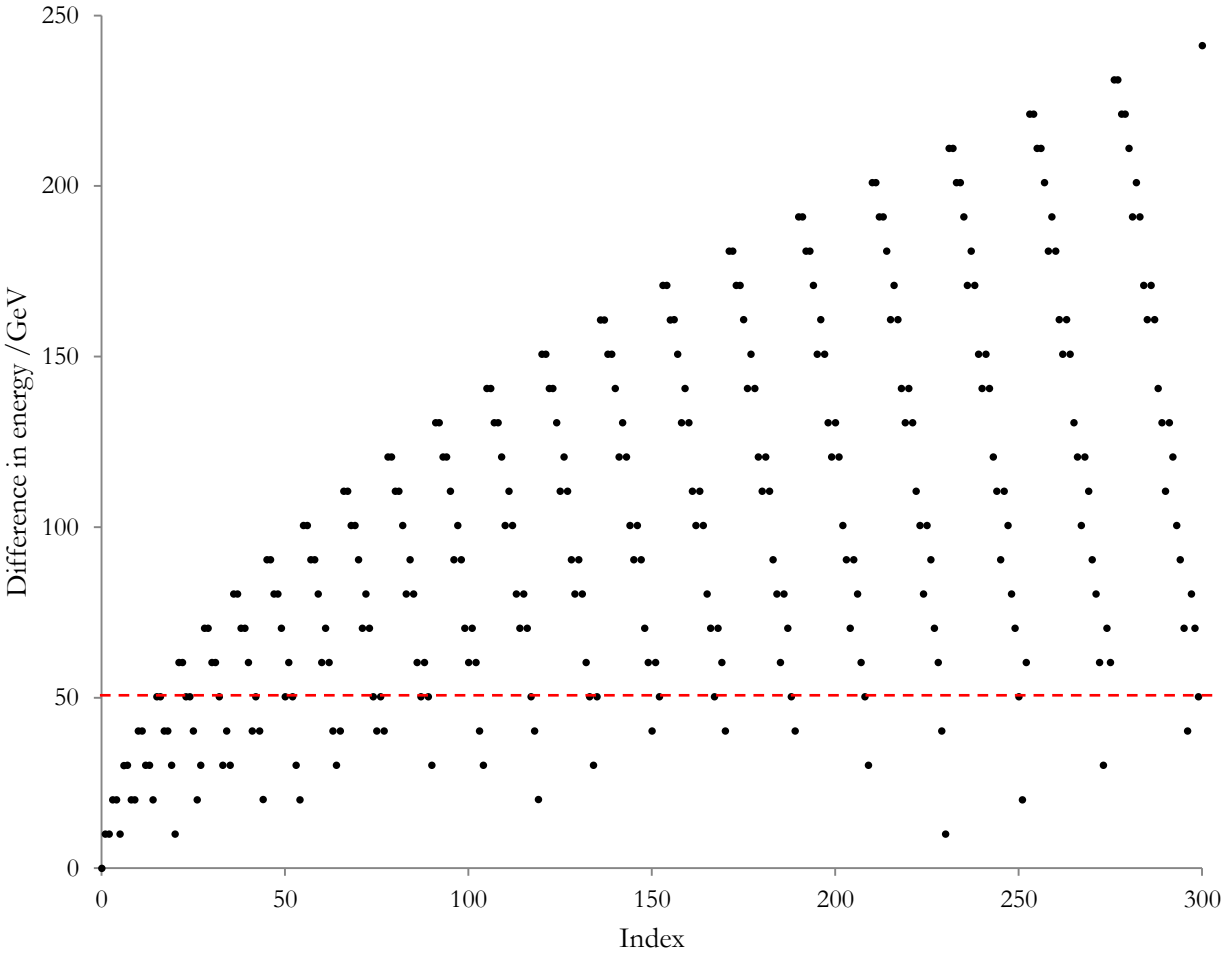
5.1 Basis state labelling tests

To investigate the validity of the ordering method the difference between the energies of the state basis and the ground state energy were plotted against index number for a typical, interacting system ($[N, \Delta x, \lambda_2, \lambda_3] = [32, 1GeV^{-1}, 1GeV^2, 1GeV^2]$). We effectively analyse the relative contributions of particle number (through the mass) and momentum (through Δx) to the basis state energy. We keep Δx fixed to simplify the process.

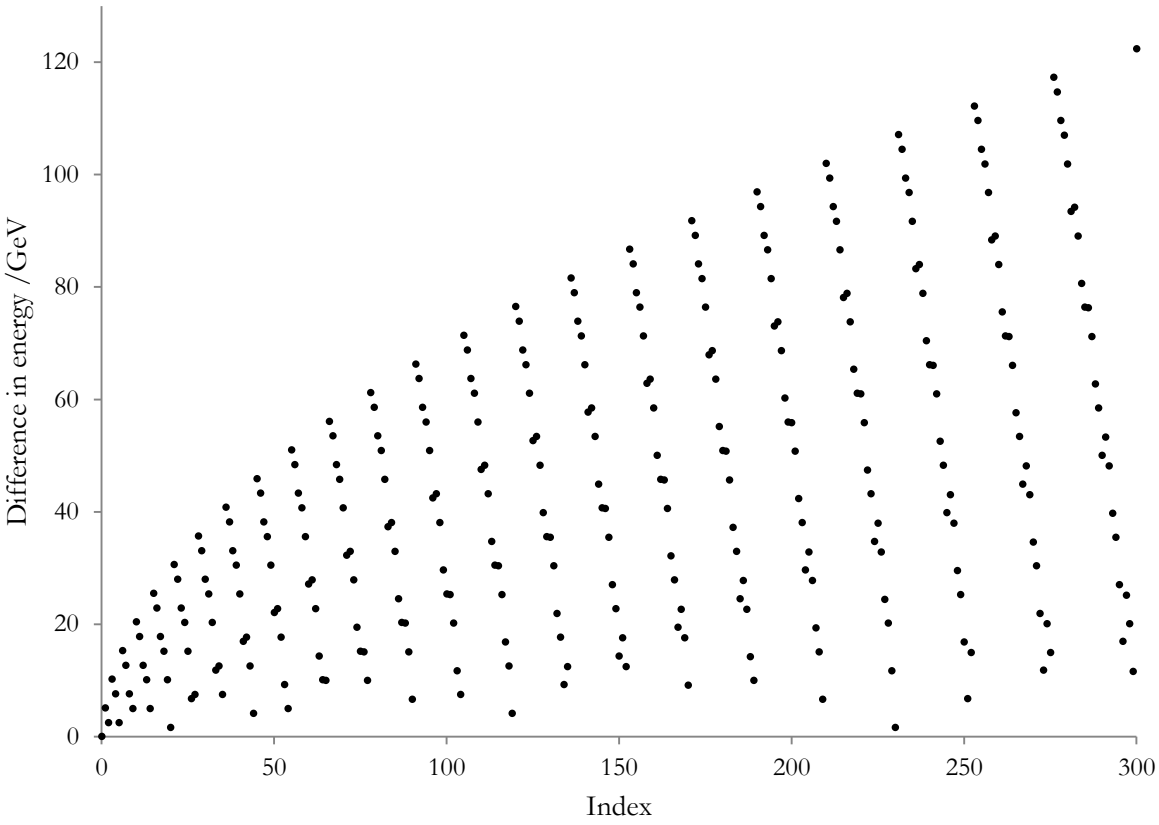
The effect of high mass can be seen in Graph 2. We expect the particle number to dominate the energy and indeed there appear to be bands whose energy roughly corresponds to an integer number of masses. With a mass equal to Δx there is little change in the energy spectrum as the momentum contributing term in Equation 19 is inversely proportional to N for low momentum modes. When the mass is lower than Δx we see a change in the spectrum (Graph 3). There are now no clear bands of particle number: the effects of momentum and mass are interwoven. An even lower mass (Graph 4) shows rough bands according to total momentum.

It appears that the cutoff used (a vertical line on the graphs) does *not* correspond to the ideal energy cutoff (a horizontal line) as the points clearly do not lie on a straight line. Despite this, the method's apparent consistency of energy ordering across a broad range of mass to Δx ratios is encouraging, as these encompass different energy dominance regimes. Importantly, the region of "ignored" basis states with low energies appears to grow sparser with increasing numbers of states considered across the same ratio range, visible in all the graphs. For a large enough number of states considered, there does arise an ever increasing energy for which no states exist with lower energy. For example, in Graph 2, the index

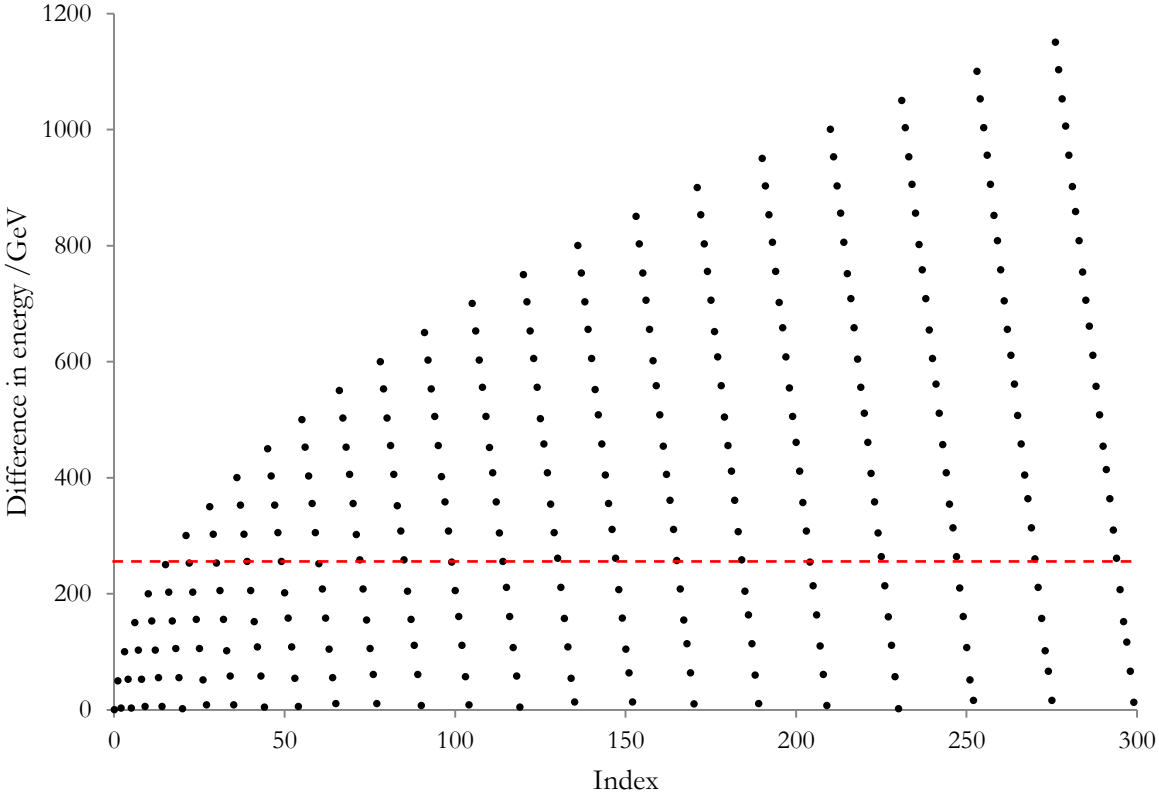
of the last single particle state (which would ensure no states below 20GeV in energy were omitted) is well above the maximum value for an integer type (~2 billion). In Graph 4, the first minimum of this type occurs when an integer number of masses exceeds the energy of one particle with the lowest allowed momentum. Here this corresponds to 386 masses, or a state with an index of 74,691. In both cases the maximum energy considered far exceeds the minimum. We find that there will always be states of relatively low energy that are not considered, but that it is not feasible in a runnable program to attempt to raise the minimum energy bar.



Graph 2 – the energy spectrum of basis states for $[m, N, \Delta x, \lambda_2, \lambda_3] = [10GeV, 32, 1GeV^{-1}, 1GeV^2, 1GeV^2]$. The dotted red line shows a typical band of states with the same particle number. This line, at 50GeV, shows that band of five 10GeV mass particles



Graph 3 – the energy spectrum of basis states for $[m, N, \Delta x, \lambda_2, \lambda_3] = [0.1GeV, 32, 1GeV^{-1}, 1GeV^2, 1GeV^2]$. There are no clear horizontal bands of states for this particle mass.



Graph 4 – the energy spectrum of basis states for $[m, N, \Delta x, \lambda_2, \lambda_3] = [0.01GeV, 32, 1GeV^{-1}, 1GeV^2, 1GeV^2]$. The states now lie in bands according to total momentum (such as shown by the red dotted line).

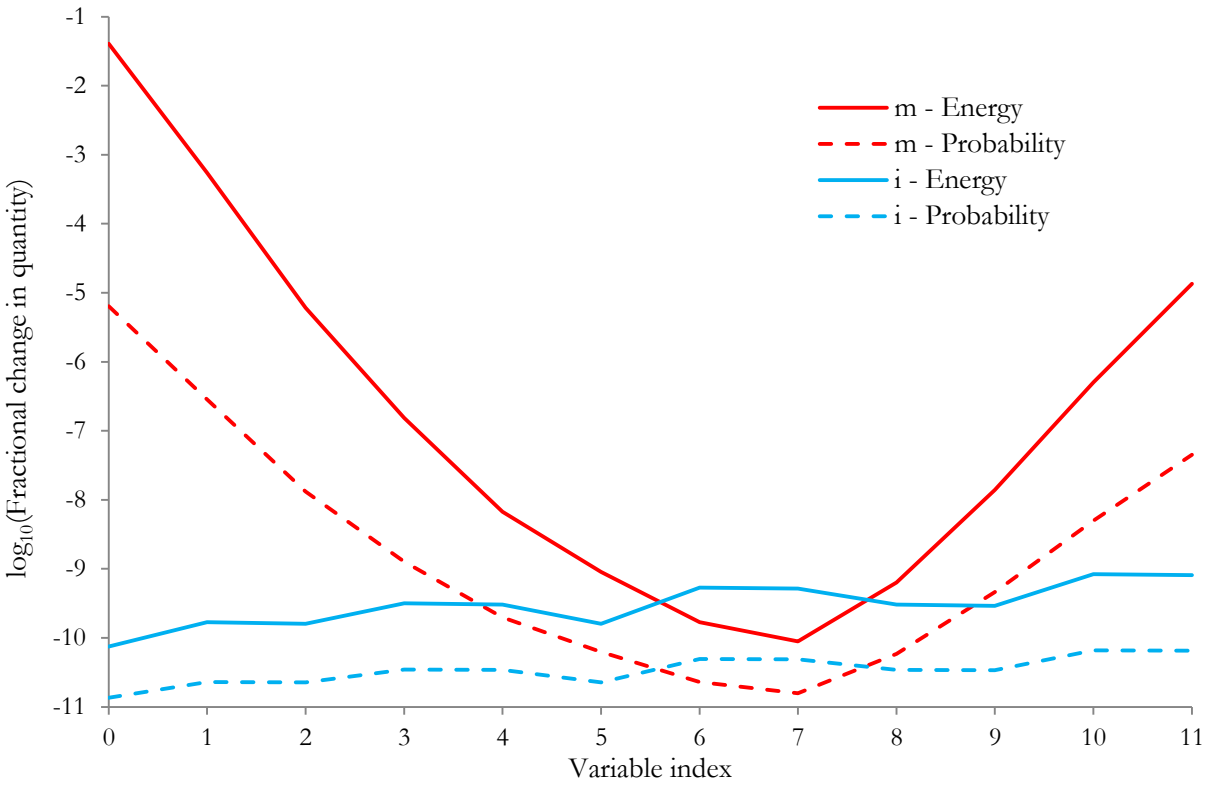
5.2 Conservation of energy and probability

It can be proven directly from the Schrödinger equation that we expect probability and energy to be conserved[LM]. While necessary, these do not prove the simulator is functioning correctly. However, we can use these expectations to test the simulators numerical limits and determine variable boundaries for further investigations.

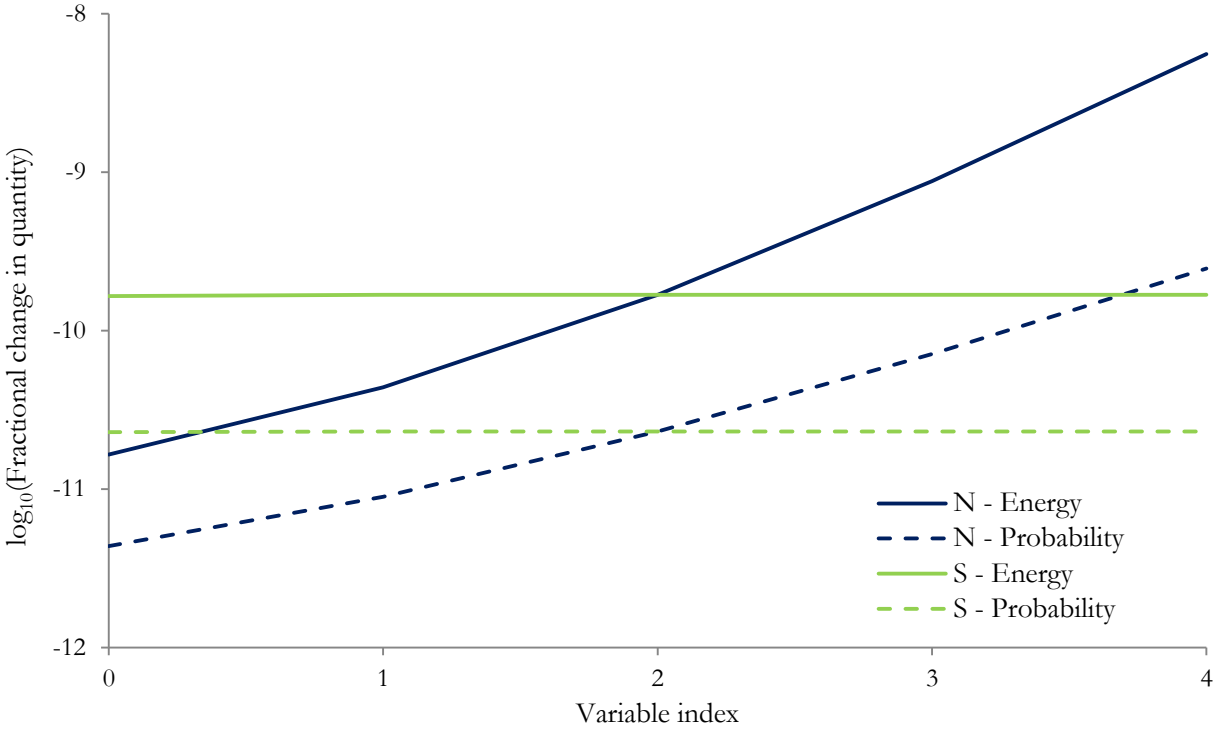
The effects of m, N, Δt, S, and initial state index (i) on energy and probability conservation for a typical interacting system were investigated. With the set of variables: $[\Delta t, \Delta x, \lambda_2, \lambda_3] = [10^{-5}GeV, 1GeV^{-1}, 1GeV^2, 1GeV^2]$, the effects of m, N, S and i on conservation of energy and probability were calculated by fixing the remaining variables to default values (see Graphs 5 and 6) and running the simulation for 1000 timesteps. The sets of values chosen for the variables can be found in the table below with the default values in bold (chosen to represent a typical simulator setup, but which we shall see is nearly optimal).

	Variable indices											
Variable	0	1	2	3	4	5	6	7	8	9	10	11
$m\ /GeV$	0.001	0.003	0.01	0.03	0.1	0.3	**1**	3	10	30	100	300
i	0	**1**	2	3	4	5	6	7	8	9	10	11
N	8	16	**32**	64	128							
S	16	46	106	**529**	1036							

Table 1 – the sets of values for the variables in Graphs 5, 6, 7 and 8.
The default values used to give Graphs 5 and 6 are in bold.



Graph 5 – the average fractional change in energy and total probability over 1000 timesteps for varying mass and initial state. The values corresponding to variable index can be found in Table 1.

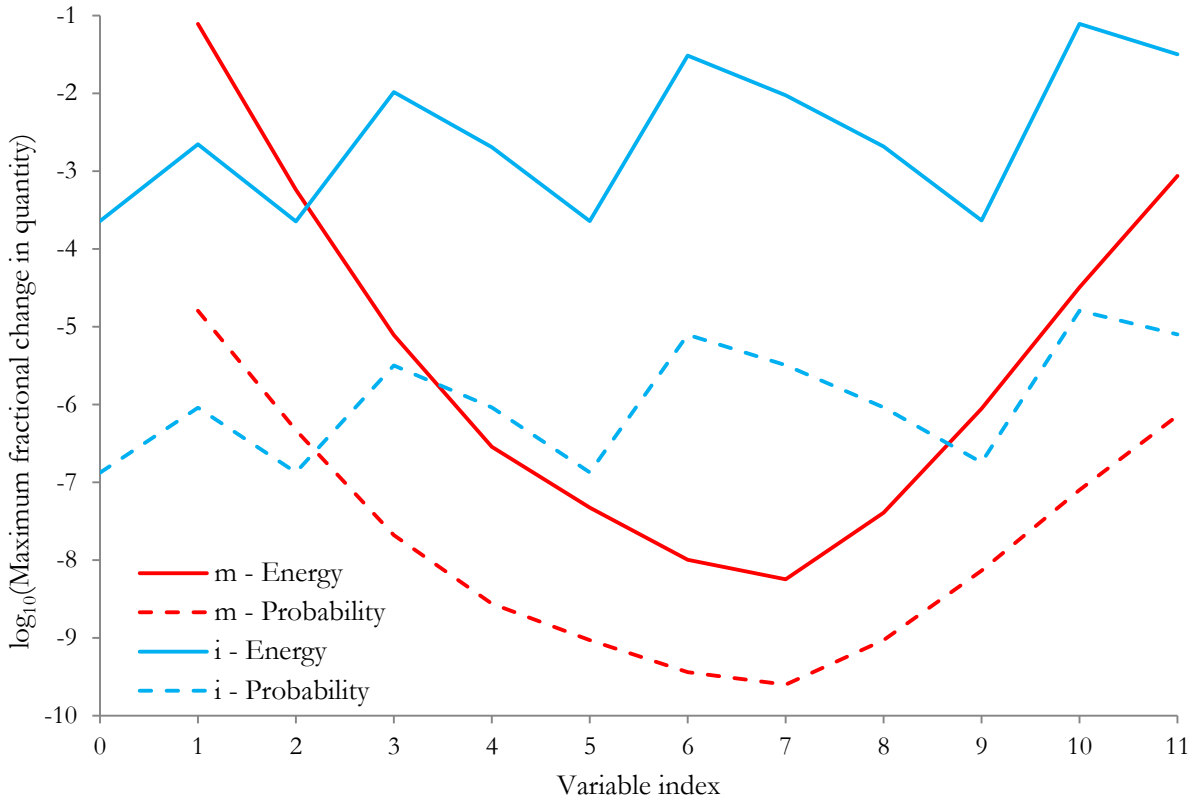


Graph 6 – the average fractional change in energy and total probability over 1000 timesteps for varying numbers of spatial points and states considered. The values corresponding to variable index can be found in Table 1.

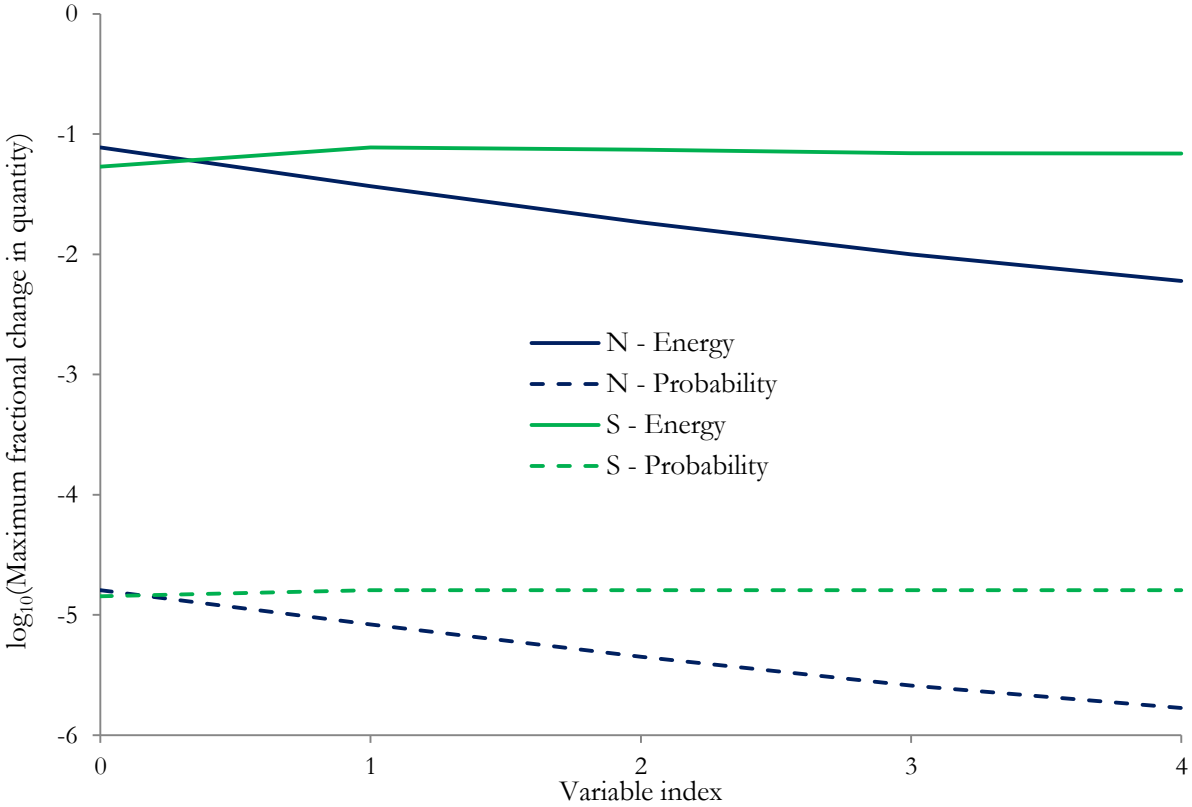
Graph 5 shows a strong dependence of the simulator's precision on mass, with the highest precision for a mass of the same size as Δx. As mentioned in Section 3.5, small masses increase interaction terms. Increasing these terms (they can be of order 10^3 for $m = 10^{-3} GeV$) while those in the Free Hamiltonian are small means the Full Hamiltonian matrix is more ill-conditioned[8], and ill-conditioned matrices perform less well in numerical methods[12]. There is little difference in simulator precision for varying initial state, although it should be noted that the other variables are nearly optimal. Graph 6 shows, unexpectedly, a slow decrease in precision for increasing N. This effect is not severe, and its cause is not known. Graph 6 also shows that the total number of states considered has no effect on system precision, although, yet again, the system is nearly optimal.

We can also plot the maximum variation in fractional conservation for specific variables with the freedom to choose the others from Table 1, as shown in Graphs 7 and 8 below. The data for $m = 10^{-3} GeV$ has been omitted, as it was found to cause the simulator to break (produce infinities). It is possible to produce reasonable simulations for masses of this size, but a lower Δt is required.

Graph 7 shows that the dominant factor affecting simulator precision is the mass. For a mass of the same order of magnitude as Δx we can achieve a maximum fractional energy error of one part in 10^8 for *any* choice of the other variables from Table 1. As previously implied, $O(1)$ values for the masses produce the best conditioned matrices, and so conserve these quantities most precisely. The zigzag effect of i in Graph 7 implies that the precision depends on the number of particles, as each spike corresponds to the first time the number of particles considered increases by one. The mass corresponding to all these points is the lowest in Table 1, an unexpected effect as the labelling system was intended to be more suitable for lower masses (see Section 3.5). A possible explanation could lie in the generally lower precision for these masses, visible in both plots and discussed earlier.



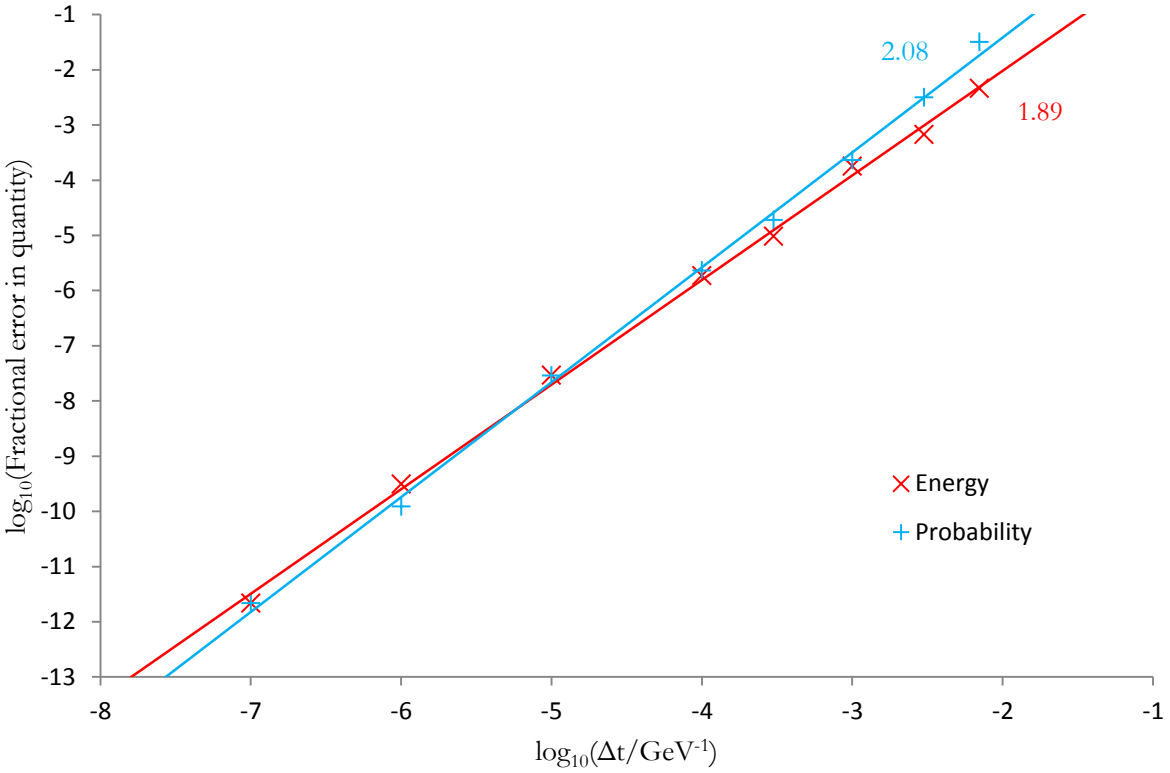
Graph 7 – the maximum average fractional change in energy and total probability over 1000 timesteps for varying mass and initial state. The values corresponding to variable index can be found in Table 1.



Graph 8 – the maximum average fractional change in energy and total probability over 1000 timesteps for varying numbers of spatial points and states considered. The values corresponding to variable index can be found in Table 1.

Graph 8 shows a slow increase in precision with increasing N, in contrast to Graph 6. As with Graph 6, this effect is not dominant and its cause is unknown. The effect of S on precision is once again minimal. For all of these plots, the variation in probability follows identical trends to that of energy, but it is conserved to a greater precision. It is suggested that this discrepancy may be a result of the increased norm of the Hamiltonian matrix for high or low masses[13]. The precise cause of this error difference is not known.

The value of Δt was fixed because it effects the operation of the integrator. The value of Δx was fixed because the scalings of Δx and m are not independent, and changing both is redundant (see Section 2.3). Using the data just obtained to choose the optimal set of $[m, N, S, i] = [1\ GeV, 128, 1036, 0]$ we can investigate the effect of changing Δt on the best expected simulation (Graph 9). We see that this precision is very close to linearly proportional to timestep size, with a gradient of ~2. This matches the expectation that the Leapfrog method is second order[11], but more importantly partially validates the simulator's Hamiltonian calculation. If the Hamiltonian matrix was incorrectly calculated in such a way that any assumed properties of the Schrödinger equation no longer applied then the integration method used might not correspond to the Leapfrog method, and so could not necessarily be expected to be second order.



Graph 9 – the average fractional change in energy and total probability over 10000 timesteps for varying timestep size for a system with $[\Delta x, \lambda_2, \lambda_3, m, N, S, i] = [1GeV^{-1}, 1GeV^2, 1GeV^2, 1\ GeV, 128, 1036, 0]$. The gradients of the lines drawn are shown, corresponding to the lines' colours.

5.3 Changing the mass

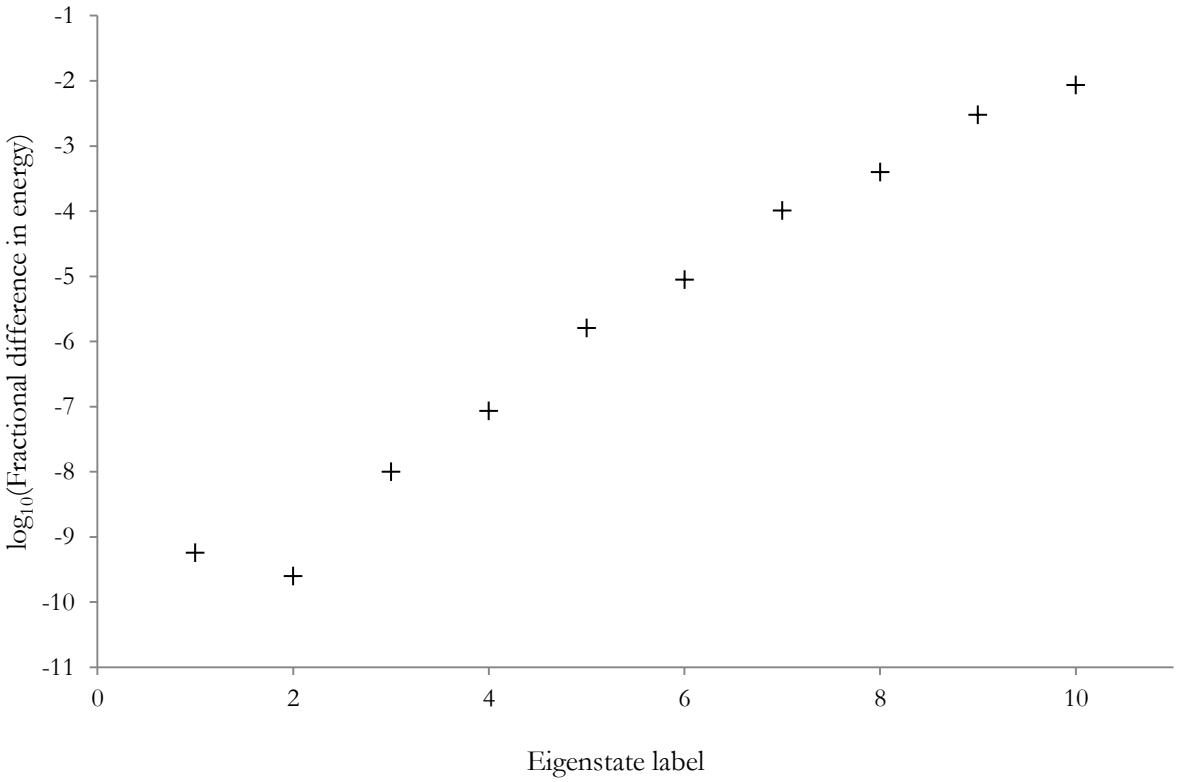
Adding a ϕ^2 interaction term of the form shown in Equation 38 is equivalent to a change in the particles mass as in Equation 39.

$$\mathrm{L}_I = -\Delta x \sum_a \frac{\lambda_2}{2!} {\phi_{x_a}}^2 \tag{38}$$

$$m \to m_{eff} = \sqrt{m^2 + \lambda_2} \tag{39}$$

$$\Delta E_s^{frac} = \frac{\left\| s m_{eff} - E_s^{sim} \right\|}{s m_{eff}} \tag{40}$$

Ignoring the non-trivial shift in energies the simulator will produce as a result of the difference in definitions of ω_p, $\hat{a}_p$ and $\hat{a}_p^\dagger$, the energy difference between the first two eigenstates produced by the simulator should be equal to m_{eff}. By setting a small value of Δx (forcing the momenta to be large) we can hope to see a spectrum of eigenstate energies differing only by integer (s) numbers of m_{eff}. Graph 10 shows that the simulator produces an energy spectrum (E_s^{sim}) for a ϕ^2 system that is close to that predicted, with a fractional difference of better than one part in 10^9 for the second and third eigenvalues found. It should be noted that the eigenstate calculation algorithm becomes decreasingly convergent as the relevant eigenvalue grows larger. While not a direct proof of the validity of the simulations for higher order interactions, we note that the Interaction Hamiltonians are all calculated using the same algorithm.



Graph 10 – the values of ΔE_s^{frac} (see Equation 40) for a system with $[N, m, \Delta x, \lambda_2] = [32, 10^{-2}GeV^{-1}, 10^{-2}GeV^{-1}, 10^{-1}GeV^2]$

We would expect any eigenstates produced by the simulator to have constant magnitudes in momentum space as time progresses; they should simply phase rotate. However, this stasis does not last long for interacting systems. The state typically changes substantially within a few thousand timesteps. This can be explained by noting the similarity between our integration method and the Power Iteration method. By considering the error in the found state in the eigenvector basis, we can see that its components that correspond to higher eigenvalues are "favoured" more than those in lower ones. This non-linear increase in error shows that the eigenstates decompose quickly in the simulator as a result of the same effect that is exploited by the method that found them[O].

5.4 Negative energies

Including a ϕ^3 interaction, with no ϕ^2 term, and calculating the system's ground state shows it has a negative energy. This might seem counter-intuitive – how can there be a lower energy than that of the vacuum? Even though we must acknowledge that our subtraction of the zero-point energy from Equation 20 has already limited our interpretation of energy to one of differences only, we also find that this zero-point energy is unaffected by the ϕ^3 interaction. We can, however, show that we expect the ground state energy of ϕ^3 interacting systems to be lower than that of non-interacting systems, which we define as zero.

The ϕ^3 interaction produces no values between states differing by an even number of particles, so the upper-leftmost element of our Full Hamiltonian matrix remains zero. The ground state of the Free Hamiltonian then still has an energy of zero, but it is not the ground state of the Full Hamiltonian because the matrix has changed[P]. This leaves us with no option but to expect the actual ground state to have a lower energy, which must then be negative. The phase differences between the components of the new ground state enable this to occur. Figure 2 shows the simulator's momentum plots immediately after a typical negative energy ground state has been found. The state still has zero momentum, and the phase difference between the zero, one and two-particle components is shown by their difference in colour. The one and two particle plots have adaptive scales. Comparison of the bar heights between, for example, the zero particle and one particle plots must take this into account.

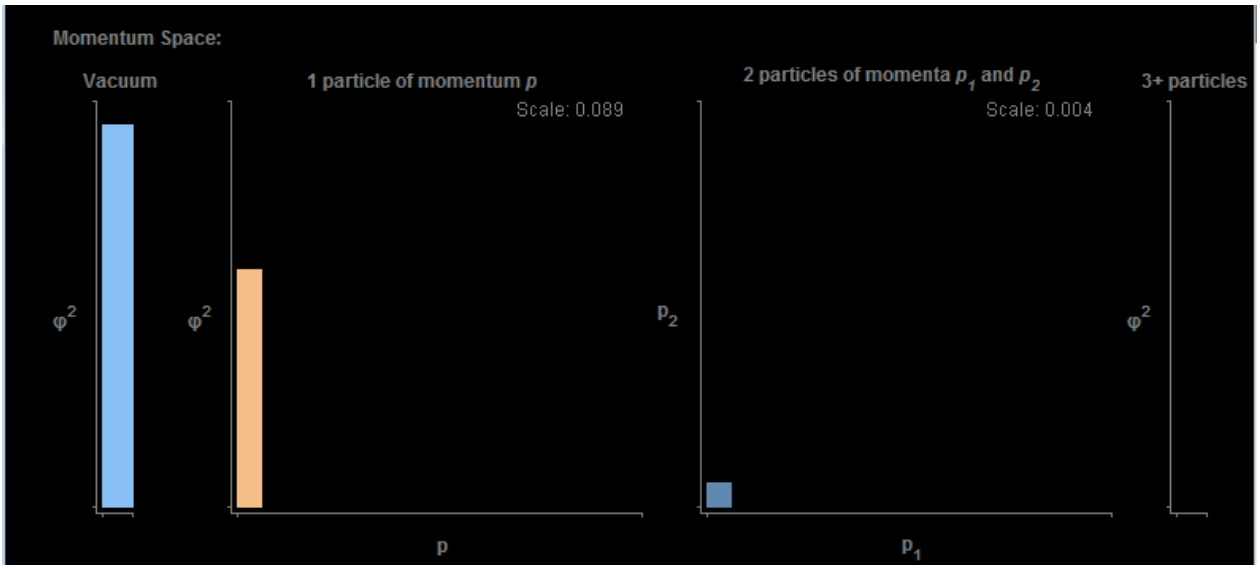


Figure 2: Part of the simulator's graphical output showing the momentum plots immediately after a negative energy ground state has been found. The system variables used to produce this plot were: $[N, m, \Delta x, \Delta t, \lambda_2, \lambda_3] = [16, 3GeV, 0.1GeV^{-1}, 0GeV^2, 5.1GeV^2]$. The calculated ground state energy was -0.17GeV.

7 – Conclusions

1. A Quantum Field Theory simulator was developed for use as a distributable Java web applet, runnable in real-time. A codebase was also produced, with support for this and future simulators including a Graphical User Interface and other utilities.
2. The simulator was tested to determine the effects of the system variables on the precision to which it conserved energy and probability. These quantities were conserved to better than one part in 10^5 for a mass range of 10^{-2} to 10^2 GeV for an otherwise optimised system.
3. The simulator produced eigenvalues for an interacting ϕ^2 system that matched expected values to better than one part in 10^9 for the second and third found.
4. An artificial continuous symmetry of the system Lagrangian that would have implied the existence of a Noether current was not determined. Systems with more than one field provide a greater opportunity for such constructions, which could then be used to test a simulator's accuracy. The codebase is sufficiently abstract for a future implementation to achieve this.

Acknowledgements

I would like to thank my supervisor, Dr Chris Lester, for his support in my somewhat unusual project progression. I would also like to thank my fellow Part III student, undertaking a similar project, for his immense cooperation in establishing the modular, distributable codebase.

References

1. http://www.falstad.com/mathphysics.html
2. https://bitbucket.org/kesterlester/qftvisualiser
3. Weinberg S., 1976, "Mass of the Higgs Boson" *Physics Review Letters, Volume 36, Number 6, Pages 294-296*
4. ATLAS Collaboration, 2012, "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC" *Physics Letters B, Volume 716, Issue 1, Pages 1-29*
5. Gripaios B., 2013, *Gauge Field Theory.* Course handout: Department of Physics, University of Cambridge (Relevant sections: 5.2, 5.7)
6. Tong D., 2006, *Quantum Field Theory.* Course handout: Department of Applied Mathematics and Theoretical Physics, University of Cambridge (Relevant sections: 1.1.1, 1.4, 2.2)
7. Kaku M., 1993, *Quantum Field Theory – A Modern Introduction,* Oxford (Relevant section: 3.2)
8. Hartree R., 1958, *Numerical Analysis,* Oxford (Relevant sections: 8.12, 8.71)
9. Institute of Mathematics and its Applications, Birmingham, ed. Walsh J., *Numerical Analysis: An Introduction,* Academic Press (Relevant section: 3.23)
10. Lisi M., 2007, "Some Remarks on the Cantor Pairing Function" *Le Matematiche, Vol. LXII, Fasc. I, Pages 55-65*
11. Hairer, Ernst, Lubich, Christian, Wanner, Gerhard, 2006, *Geometrical Numerical Integration,* Springer
12. Cheney E.W., Kinkaid D., 2007, *Numerical Mathematics and Computing,* Sixth Edition, Cengage Learning (Relevant section: 8.2)
13. Fröber C.-E., 1969, *Introduction to Numerical Analysis,* Second Edition, Addison-Wesley (Relevant section: 3.6)