# Numerical Investigation of Exotic Phases in Quantum Lattice Models

Niall Moran
B. Eng.

NUI MAYNOOTH
Ollscoil na hÉireann Má Nuad

Thesis presented for the degree of

Doctor of Philosophy

to the

Department of Mathematical Physics
Faculty of Science
National University of Ireland, Maynooth

Research supervisor

Dr. Jiri Vala

October 2010

**Abstract**

In this thesis we present details of the design, development and application of a large scale exact diagonalisation code named DoQO (Diagonalisation of Quantum Observables). Among the features of this code are its ability to exploit physical symmetries and the fact that it has been designed to run in parallel to take advantage of modern high performance computing resources. The primary motivation for developing this code has been the investigation of exotic phases in quantum lattice models, and in particular of topological phases. A significant portion of the thesis concerns the investigation of supersymmetric lattice models, which involves significant use of the developed DoQO code. These are a relatively new (2003) family of models consisting of spinless fermions hopping on a lattice with the interactions tuned to a point where the spectrum exhibits supersymmetry. These models are extremely rich in the physics that they exhibit. Among the phases believed to exist in these models are critical, super-frustrated and super-topological phases. DoQO was also employed to investigate finite size effects in the Kitaev honeycomb lattice model. This is a spin model which exhibits both abelian and non abelian topological phases.

# Contents

# Chapter 1

# Introduction

The picture of physics that I find most striking is the analogy made by Richard Feynman [1], in which he describes a physicist as being similar to an individual attempting to discover the rules of a complicated board game along the lines of chess, by observing only small portions of the board at sporadic intervals. Over time specific rules governing how certain pieces can be moved are deduced. Looking more closely these specific rules may be found to be special cases of a more general rule, which can replace these. However knowing the allowable moves of each piece does not automatically make one an expert at playing the game. Even if the set of allowable moves of each individual piece is small, the number of ways these moves can be combined is astronomically large and thus attempting to systematically evaluate the optimal move to make at each stage is highly non-trivial. To become a good player one learns to recognise patterns in the game and over time develop strategies for how best to approach them.

Quantum mechanics and quantum field theory are the most accurate frameworks we have for describing the behaviour of physical systems where the effects of gravity are unimportant. Similar to the situation with the game, using these frameworks and taking into account all degrees of freedom to determine the behaviour of a physical system is impossible for any but the simplest of systems. As a result to study a particular physical system or physical phenomena we look for simplified models which capture the

essence of the physical system, or exhibit similar properties to the physical phenomenon under study.

This thesis is concerned with the exploration of exotic phases of strongly correlated quantum lattice models, primarily via exact numerical methods. The use of a lattice can be seen as a means of approximating a continuum or in other cases is justified as approximating the crystal lattice structure found in solids. The fact that these models are strongly correlated means that perturbative approaches as well as attempts to make further approximations that simplify the models are in general ineffective. An exotic phase in this context is a phase with novel or interesting properties.

## 1.1 Topological phases of matter

Of particular interest to us are so-called topological phases. These phases are effectively described by a Topological Quantum Field Theory (TQFT). Topological phases posses long range order but unlike other ordered phases, this order cannot be described by local order parameters. Instead this order is described by properties dependent on the topology of the system manifold, among which is the ground state degeneracy. An example of this is found in the toric code model [2]. For this two dimensional model there is a single ground state when the system is realised on a plane and a four fold degenerate ground state when wrapped on a torus. Other characteristics of systems exhibiting topological phases are topological entanglement entropy, a robust spectral gap and critical edge modes described by conformal field theory (CFT). As well as being fascinating in their own right, topological phases are the essential ingredient for topological quantum computation.

Two dimensional quantum lattice models provide a setting for the theoretical investigation of the microscopic mechanisms leading to topological phases and also a possible setting for their experimental realisation in atomic and molecular systems [3, 4]. The simplest and most well known lattice model exhibiting topological order is the toric code model [2]. This is an exactly solvable model on a square lattice exhibiting an abelian topological phase. Another important model is the Kitaev honeycomb lattice model [5]. This

is a model of interacting spin half particles on a honeycomb lattice that exhibits both abelian and non-abelian topological phases. The method of constructing models with string net ground states introduced in [6] allows the realisation of lattice models with arbitrary topological phases. However these models can contain unphysical interaction terms involving large numbers of sites. Loop gas models which are realisable on lattices can also exhibit topological phases [7, 8, 9, 10]. There is also indication that topological order may exist in supersymmetric lattice models and in particular on the square octagon lattice [11].

Some aspects of these models are discussed in this thesis. This consists of work on supersymmetric lattice models(chapter 3) and on the Kitaev honeycomb lattice model[12] (chapter 4) where finite size effects are investigated on the torus.

## 1.2 Supersymmetry

Supersymmetry (SUSY) is a powerful theory which emerged first in the field of particle physics. It allows for a solution of the mass hierarchy problem and facilitates the unification of non-gravitational forces at high energies. The core idea of SUSY is that for each particle there is a super-partner with the same energy but with spin whose value differs by a half. For each fermionic particle there is a bosonic super-partner with equal energy. To date no direct experimental evidence has been found to prove the existence of supersymmetry. This is attributed to supersymmetry being broken at low energies and the super-partners being too massive to have been found in experiments carried out to date. There is hope that the Large Hadron Collider could change this and provide experimental evidence of supersymmetry. Despite the lack of experimental confirmation the theory of SUSY has grown and spread to other fields of physics including condensed matter physics.

We investigate a family of models called SUSY lattice models[13], relevant to condensed matter physics which feature fermions on a lattice where the interactions are tuned to a point where the states exhibit SUSY properties. For the SUSY chain calculations of the finite size gap scaling, dispersion

relation and entanglement entropy confirm that this model can be described by a superconformal field theory and that the results from DoQO are correct. For the staggered SUSY chain we present results of finite size gap scaling calculations at different staggerings. We then focus on the staggering limits and show exact expressions for the ground states and entanglement entropy in these states for each case. These are supported with numerical calculations where appropriate. For the square octagon SUSY model the one and two point functions are calculated, the effect of adding defects are investigated and the numerically calculated ground state is compared to a proposed trial Projected Product State(PPS) wavefunction.

## 1.3 Numerical tools

Solving a quantum mechanical system means finding the eigenvalues and eigenstates of a complete set of commuting observables. For a small set of quantum many-body systems exact analytical expressions can be found for these quantities. In other cases analytical insights can provide only qualitative descriptions of the behaviour of the system. Numerical tools play a key role by providing unbiased data for a given model which are used to form and verify analytical descriptions and hypotheses. However these numerical calculations can prove extremely challenging. Here we provide an overview of the most common techniques for treating these systems.

### 1.3.1 Exact diagonalisation

The most straightforward numerical approach is to numerically diagonalise the quantum observables. Using this method all quantities can be calculated for a given system. However the dimensions of the matrices involved grow exponentially with the the number of particles and as a result this method is only tractable for relatively small systems. Despite this limitation this technique remains extremely relevant due to the exact unambiguous results it provides. Finite size scaling studies can also give a good indication of the macroscopic model behaviour. This is the central numerical method used

throughout this thesis.

It is possible to calculate many quantities relevant to research into topological phases with exact diagonalisation. These include the ground state degeneracy, values of excitation gaps, spectral flows, dispersion relations, expectation values of additional operators and entanglement properties.

We developed a code named DoQO (Diagonalisation of Quantum Observables). This is a large scale exact diagonalisation code capable of diagonalising observables of arbitrary systems of spin half or spinless fermionic particles, with many-particle interactions, on arbitrary lattices or graphs, using large high performance computing resources and employing symmetries where applicable.

### 1.3.2 Quantum Monte Carlo

Monte Carlo methods are a family of general methods which can be used for calculating properties of physical systems [14]. These methods consider only a portion of the complete phase space, obtained by sampling with the appropriate probability distribution. Quantum Monte Carlo methods extend these methods to treat quantum systems and can provide very accurate results for large numbers of particles. However for frustrated and/or fermionic systems these methods are hampered by the 'sign problem' which severely impacts their efficiency. The sign problem describes the appearance of negative values appearing as probabilities when certain quantum systems are mapped to classical systems. It has been shown that a general solution to the sign problem is an NP-hard (nondeterministic polynomial-time hard) problem [15]. The models discussed in this thesis are afflicted with sign problems and thus Quantum Monte Carlo methods are not used.

### 1.3.3 Approximative techniques

Approximative methods exist which attempt to truncate the full Hilbert space to a relevant subspace using physically motivated assumptions. A network of tensors is used to represent the states in this truncated Hilbert space. The dimension of these tensors $D$ is proportional to the amount of

entanglement that can be accurately represented. For gapped systems with local interactions the entanglement entropy is thought to follow the 'area law' which says that the entanglement entropy is proportional to the area of the boundary [16].

The much celebrated Density Matrix Renormalisation Group (DMRG) [17, 18] and equivalent Matrix Product State (MPS) [19] methods are very effective at treating gapped 1D quantum systems. This results from the fact that the boundary of 1D systems is constant. This means that the low lying states of gapped 1D systems can be accurately represented by a tensor network with constant $D$.

For gapped 2D systems $D$ must be increased as the boundary increases to accurately represent the low lying states. In addition performing tensor contraction operations exactly for 2D networks quickly becomes intractable, with the result that further approximations are required. For these reasons the use of approximative methods for treating 2D systems remains mainly a research topic. Tensor Product States (TPS) [20] and Projected Entangled Pair States (PEPS) [21] are generalisations of MPS to 2D systems. Tree Tensor Networks (TTN) [16] and the Multi-scale Entanglement Renormalisation Ansatz (MERA)[22] attempt to represent states with long range entanglement more efficiently through the use of hierarchical tensor networks.

## 1.4 Thesis outline

Chapter 2 describes the developed exact diagonalisation code named Diagonalisation of Quantum Observables (DoQO) [23]. In chapter 3 SUSY lattice models are discussed. After introducing the SUSY lattice models, results of calculations for the SUSY chain, staggered SUSY chain and square octagon SUSY models are presented. In chapter 4 details of investigations into the finite size effects in the Kitaev honeycomb lattice model are presented. Finally chapter 5 discusses conclusions and future work.

# Chapter 2

# Diagonalisation of Quantum Observables

In this chapter we discuss the exact diagonalisation method and present details of a large scale exact diagonalisation code that we have developed. This code has been named Diagonalisation of Quantum Observables (DoQO). It is capable of constructing and diagonalising observables for spin half and spinless fermionic particles on both regular lattices and general graphs. It produces numerically exact data about the low energy part of the operator spectrum and provides access to the full eigenfunctions, making it possible to calculate useful physical quantities. Exact diagonalisation techniques, like those implemented in DoQO, are essential for getting important physical insights and also for understanding the limits of approximative techniques which in principle allow larger systems to be studied. DoQO can exploit physical symmetries which reduce the relevant basis set size and the related memory requirements, while providing additional information about each eigenstate. In addition, DoQO has been designed to work in parallel to take advantage of modern High Performance Computing (HPC) resources and has been benchmarked on various HPC platforms. The memory, load and indexing issues are taken care of automatically using a number of custom built and generalizable algorithms.

In the rest of this chapter we introduce essential concepts relevant to

quantum observables and symmetries and then discuss the associated implementation issues. In section 2.2 we provide an overview of the software together with usage information. In section 2.3 the central components of the software are explained in more detail. In section 2.4 the scaling and performance of the code for a benchmark system on a number of different HPC platforms is investigated. In the last chapter of the thesis, section 5 we briefly outline additional features that could improve and extend the functionality of this code. This includes extensions related to handling different particle types as well as exploiting additional symmetries.

## 2.1 Quantum observables

The main objective of DoQO is to provide exact low energy spectral data of quantum Hamiltonians, specifically ground state and low lying eigenstates and eigenvalues. This data can be used to formulate and verify relevant analytical models and to benchmark approximative techniques.

Quantum observables are self-adjoint operators which represent observable physical quantities. Their eigenvalues are real numbers and together with the related eigenstates correspond to the possible measurement outcomes. A prominent example of an observable is the Hamiltonian. When the Hamiltonian is time independent this observable represents the total energy of the system and generates quantum dynamics through the Schrödinger equation. Its eigenvalues are the energy levels and the eigenstates are the corresponding stationary states. In the context of condensed matter and statistical physics we are generally interested in the ground state and/or low lying part of the energy spectrum.

DoQO is designed to work with systems of spin half or spinless fermionic particles on regular lattices or general graphs. It is possible to describe interactions involving arbitrary numbers of particles. Each spin half particle is a two-level quantum system whose states are vectors in a two-dimensional Hilbert space $\mathcal{H}^2$. Likewise each site of a spinless fermionic system can be either empty or occupied.

The Hilbert space of a system of $n$ spin half particles is the n-fold tensor

product $\bigotimes_1^n \mathcal{H}^2$, so its dimension is an exponential of the number of spin half particles. DoQO tackles the system size limitation, which derives from the exponential scaling of the basis set size, by several means. First, symmetries can be used to reduce the basis set without compromising the quantitative accuracy of the computed data. Second, DoQO exploits the sparsity of the matrix representation to save memory. Lastly DoQO constructs the observables such that the diagonalisation procedure can be performed efficiently in parallel on massively parallel distributed memory architectures.

### 2.1.1 Employing symmetries

The use of symmetries reduces the computational resources required to diagonalise an observable while simultaneously providing additional physical information about each eigenstate. A symmetry operator (SO) is an operator which commutes with the Hamiltonian. By working in the eigenbasis of a SO and reordering the basis elements appropriately, the matrix representation of the observable becomes block diagonal. Each block corresponds to a particular eigenspace of the SO and can be diagonalised separately. The states resulting from the diagonalisation of a given block are labelled by a quantum number that is the eigenvalue of the SO associated with that block.

DoQO is capable of exploiting the symmetries which conserve parity (section 2.1.1.2), filling (section 2.1.1.1) and momentum (section 2.1.1.3). Where multiple symmetries are compatible they can be exploited simultaneously. Here we define these symmetries more precisely and show how DoQO determines and indexes the relevant eigenvectors of each SO.

#### 2.1.1.1 Filling

The SO's that conserve filling are defined as $\frac{1}{2}\sum_i(\sigma_i^z+1)$ for spin half systems and $\sum_i n_i$ for spinless fermionic systems with $i$ running over the subset of sites for which filling is conserved. $\sigma^z$ is one of the Pauli matrices defined in section 2.B.1 and $n_i$ is the spinless fermionic number operator defined as $n_i = c_i^\dagger c_i$ where $c_i^\dagger$ and $c_i$ are the spinless fermionic creation and annihilation operators respectively. Filling is related via a constant to magnetisation

(defined as $\sum_i \sigma_i^z$) for spin half systems and for spinless fermionic systems is simply the number of fermions. These SO's are diagonal in the standard basis that DoQO uses. For spin half systems the standard basis refers to the basis in which $\sigma^z$ operators are diagonal and for fermionic systems, the number occupancy basis. As a result of the SO's being diagonal in the standard basis no change of basis elements is required to exploit this symmetry. If filling is conserved over a subset of sites this implies that parity (section 2.1.1.2) is also conserved over that subset. In this respect conservation of filling is stronger than conservation of parity. If filling is conserved over $M$ subsets of sites with each subset containing $s_i$ sites there are $\prod_i^M (s_i + 1)$ blocks. The dimension of the block with filling $f_i$ in each subset is $\prod_i \binom{s_i}{f_i}$.

#### 2.1.1.2 Parity

The SO that conserves parity is defined as $\prod_i \sigma_i^z$ for spin half systems and $\prod_i (2n_i - 1)$ for spinless fermionic systems. This symmetry says that the parity of the magnetisation for spin half systems or filling (section 2.1.1.1) for spinless fermionic systems is conserved. Here the products are over the subset of sites for which parity is conserved. Like the SO's that conserve filling these operators are diagonal in the standard basis. DoQO can also exploit situations where parity is conserved over subsets of spins or sites. This results in a further reduction of the basis set size for each block and thus the computational resources required. For example, if there are $M$ subsets of sites over which parity is conserved then there are $2^M$ blocks each with dimension $2^{N-M}$ where $N$ is the total number of sites.

#### 2.1.1.3 Momentum

DoQO is capable of exploiting translational symmetry in two dimensions, which conserves the momentum in each direction. The SO's in this case are the translation operators $T = e^{i\hat{x}\hat{k}}$ and the quantum numbers $e^{ik}$, where $k$ is the momentum with $0 \leq k \leq 2\pi$. As these operators are not diagonal in the standard basis a change of basis elements is required to exploit these symmetries. In DoQO the method of representatives is used to work in the

eigenbasis of the SO's, and calculate the matrix elements of each block. This method is discussed in [24, 25].

From each set of configurations which can be related via translation a representative configuration $|\,\psi_r\rangle$ is selected. This is conventionally chosen to be the configuration with the smallest numerical label in the set. An eigenstate of the translation operators with a given momentum can be determined from each representative configuration. These eigenstates have the form of a weighted sum of states obtained by translating the representative configuration. The weight of each state is a function of the momentum and of the number of translations needed to cycle back to the representative configuration. For a square lattice with $L \times L$ sites and translation operators $T_1$ and $T_2$, an eigenstate with momentum $k_i$ in the direction of $T_i$ is given by:

$$|\,\psi_{k_1,k_2,r}\rangle = \frac{1}{\mathcal{N}} \sum_{x_1=0}^{L-1} \sum_{x_2=0}^{L-1} e^{i(x_1 k_1 + x_2 k_2)} |\,\psi_r^{x_1,x_2}\rangle$$

where $\mathcal{N}$ is a normalisation factor ensuring $\langle\psi_{k_1,k_2,r}|\psi_{k_1,k_2,r}\rangle = 1$ and $|\,\psi_r^{x_1,x_2}\rangle = T_1^{x_1} T_2^{x_2} |\,\psi_r\rangle$. This is in essence the discrete Fourier transform.

DoQO exploits the fact that the matrix elements for a given block can be calculated by working with the representative configurations alone. This can be clearly seen by expanding the expression for each matrix element in a given block $\langle\psi_{k_1,k_2,r}\,|H|\,\psi_{k_1,k_2,r'}\rangle$.

#### 2.1.1.4 Labelling

In order to implement the exploitation of symmetries, the basis elements of each block must also be labelled appropriately. To do this DoQO uses perfect mapping functions for the symmetries which conserve parity and filling. Sorted distributed arrays are used to label the representative configurations when exploiting translational symmetries. The implementation details of the so called perfect mapping functions used are explained in section 2.3.2 and the use of distributed sorted arrays is discussed in sections 2.3.3.1 and 2.3.3.2.

## 2.2 Software structure and usage

DoQO is written in C++ and makes extensive use of the PETSc [26, 27, 28] and SLEPc[29] libraries which are required for DoQO to work. The XML file format is used for some of the input files due to its extensibility. An open source library named TinyXML[30] is used to parse the XML files and is included with the DoQO code.

### 2.2.1 Compiling DoQO

A makefile is supplied with the DoQO source which can be used to build DoQO. For this to work the PETSc and SLEPc libraries must first be present and the relevant environment variables set. A README file is also supplied which contains detailed instructions about compiling PETSc, SLEPc and DoQO.

### 2.2.2 Basic usage

DoQO can be run as a single process or as multiple intercommunicating processes. To run a single process of DoQO a command as in code block 1 is used from the directory containing the DoQO executable. The input filename is passed to DoQO via the '-input' switch.

**Code block 1** Command used to launch a single process of DoQO.

```
./doqo -input sample_input.xml
```

To run multiple intercommunicating processes of DoQO the MPI launcher application is used. This is named 'mpiexec' or 'mpirun' depending on the MPI implementation in use. An example of the command used to launch 16 DoQO processes is given in code block 2.

**Code block 2** Command used to launch DoQO with sixteen processes using mpiexec.

```
mpiexec -np 16 ./doqo -input sample_input.xml
```

The input file which is passed to the DoQO executable is an XML file which contains all the parameters that control how DoQO runs. An example of a simple input file is given in code block 3.

**Code block 3** Sample input file for DoQO containing required parameters. These are the model file which describes the observable, the task list file which specifies the coefficient values for each task and the output prefix which is the prefix used for the output files.

```
<?xml version="1.0" encoding="UTF-8"?>
<SIMULATION>
    <PARAMETERS>
        <MODEL_FILE>ising_chain_L_8.ham</MODEL_FILE>
        <TASK_LIST>ising_chain_tasks</TASK_LIST>
        <OUTPUT_PREFIX>ising_chain_L_8</OUTPUT_PREFIX>
    </PARAMETERS>
</SIMULATION>
```

The file specified by the MODEL_FILE parameter describes the observable that is being diagonalised. An example of such a file for a system of spin $\frac{1}{2}$ particles is shown in code block 4. This example describes the Hamiltonian for the transverse field Ising model on a ring with eight spins.

$$H = J \sum_{i=1}^{8} \sigma_i^x \sigma_{i+1}^x + h \sum_{i=1}^{8} \sigma_i^z$$

with $\sigma_{8+1}^x = \sigma_1^x$. The format of this file is:

- The first line specifies the number of sites.
- The second line contains the label 'PARAMETERS'.
- The following lines up to the line containing the label 'TERMS' contain the parameter names (one per line) of the coefficients used in the definition of the observable.
- Next is a line containing the label 'TERMS'.

- The rest of the lines specify the terms that make up the observable. The format for these lines is:
  - A comma separated list followed by an asterisk where each entry is composed of a site index and an X,Y or Z which signify a $\sigma^x, \sigma^y$ or $\sigma^z$ operator acting on the site specified by the site index.
  - A comma separated list of the parameters that are multiplied together to get the coefficient for each term.

There is an implicit identity for each site not explicitly mentioned. In this way '* J' would describe the identity operator over all sites multiplied by the coefficient $J$. For spinless fermionic systems a similar format is used except

**Code block 4** Input specification for Hamiltonian of Ising chain in a transverse magnetic on an 8 site ring.

```
SITES 8
PARAMETERS
J
h
TERMS
1 X,2 X * J
2 X,3 X * J
3 X,4 X * J
4 X,5 X * J
5 X,6 X * J
6 X,7 X * J
7 X,8 X * J
8 X,1 X * J
1 Z * h
2 Z * h
3 Z * h
4 Z * h
5 Z * h
6 Z * h
7 Z * h
8 Z * h
```

instead of using X,Y and Z to represent $\sigma^x, \sigma^y$ and $\sigma^z$ we use C and A to

represent the creation and annihilation operators $c^\dagger$ and $c$ respectively. Refer to appendix 2.B for further details on constructing operators.

The file specified by the TASK_LIST parameter contains a list of the parameter values to be used for each task. This file has one line per task, where each line consists of a comma separated list of parameter assignments. These parameters are used to calculate the coefficients for each term of the observable. Any parameters which are not specified are set to zero. By specifying appropriate parameter values in the task file, scans of the parameter space can be performed with a single call to DoQO. An example of a task file with seven tasks is given in code block 5.

**Code block 5** Example of task file specifying seven tasks for the Ising chain described in code block 4.

```
J = -1.0
J = -1.0, h = 0.5
J = -1.0, h = 1.0
J = -1.0, h = 1.5
J = -1.0, h = 2.0
J = -1.0, h = 2.5
J = -1.0, h = 3.0
```

DoQO produces XML output files. A general output file is created which lists the specific output files that contain the results of each diagonalisation. The name of the general output file is composed of the output prefix as specified in the input file with the suffix '.output.xml' appended. Details about the code version and environment are also provided in this file. In addition for each task and symmetry block a separate output file is created containing the results of that diagonalisation. The information written to these output files includes the eigenvalues converged, the error estimates as well as additional information. This includes the basis set size, the time taken and the number of iterations of the solver method used.

### 2.2.3 Exploiting symmetries

To enable the use of symmetries in DoQO an element describing the symmetries to be exploited is added to the XML input file. The element is enclosed within tags labelled 'SYMMETRIES'. For each symmetry that one wishes to exploit a child element describing that symmetry is added inside the initial element. These child elements are enclosed between tags labelled 'PARITY', 'FILLING' and 'MOMENTUM' corresponding to the symmetries that conserve parity, filling and momentum respectively. Code block 6 contains an extract of an input that has a symmetries element with a child element specifying a symmetry that conserves parity.

Additional details for each symmetry are given in a metadata file specified by the 'file' attribute for each symmetry element. For the symmetries that conserve parity and filling this metadata file specifies the subsets of sites on which the parity or filling is conserved. For translational symmetries this file specifies the lattice geometry and translation vectors. The format for these files is explained in more detail later in this section.

DoQO will by default diagonalise an observable in each eigenspace of the supplied symmetry operators one after another. An optional child element can be added within each symmetry element that allows one to restrict the calculation to specific eigenspaces. This element is enclosed by tags labelled 'RELEVANT_SECTORS' and has an attribute called 'number' that specifies how many of the eigenspaces are to be used. Each eigenspace to be used is then specified by a child element of this element labelled by 'SECTOR' that encloses an integer that uniquely labels an eigenspace of the symmetry operator. The details of how each eigenspace is labelled for each of the symmetries that DoQO exploits is described later in this section. Code block 6 shows an example of an input file that specifies that DoQO exploit the conservation of parity and that it is restricted to the eigenspaces of the parity operators labeled one and three.

For the symmetry that conserves parity the metadata file identifies the subsets of sites for which parity is conserved. This file contains one line for each subset of sites. Each line consists of a bit string with the number of

**Code block 6** Input file specifying that the symmetry that conserve parity should be used.

```
<SIMULATION>
    <PARAMETERS>
       ...
    <SYMMETRIES>
        <PARITY file="subsets.txt">
            <RELEVANT_SECTORS number="2">
                <SECTOR>1</SECTOR>
                <SECTOR>3</SECTOR>
            </RELEVANT_SECTORS>
        </PARITY>
    </SYMMETRIES>
    </PARAMETERS>
</SIMULATION>
```

bits matching the number of sites in the system. Each site corresponds to a bit in this string with the site indices increasing from right to left. For sites included in the subset, there is a '1' in the position that corresponds to that site and a '0' otherwise. In code block 7 an example of such a file for a system with eight sites is shown. Here two subsets of sites are defined, one which spans sites with indices 1 to 4 and one that spans sites with indices 5 to 8. The SO's that conserve parity over these subsets of sites for spin $\frac{1}{2}$ systems are $P_0 = \prod_{i=1}^{4} \sigma_i^z$ and $P_1 = \prod_{i=5}^{8} \sigma_i^z$. The four eigenspaces with eigenvalues $(p_0, p_1) = (1,1), (1,-1), (-1,1), (-1,-1)$ are labelled with integers from zero to three respectively. If, as specified in code block 6, only eigenspaces labelled one and three were used, this would select the eigenspaces $(p_0, p_1) = (1,-1), (-1,-1)$.

When using the symmetry that conserves filling the metadata file specifies the subsets of sites over which filling is conserved. The format of this file is the same as that used for the symmetry that conserves parity. In this case the SO's are $C_i = \frac{1}{2} \sum_{j \in N_i} (\sigma_j^z + 1)$ where $N_i$ is the set of sites in the $i^{th}$ subset. There are $\prod_i (s_i + 1)$ eigenspaces where $s_i$ is the number of sites in the $i^{th}$ subset. These eigenspaces correspond to the possible fillings that each

subset can have. If $f_i$ are the fillings for each subset then each eigenspace is labelled by $\sum_i (f_i \prod_{j=0}^{j<i} (s_j + 1))$.

**Code block 7** File specifying two subsets of sites, 1 to 4 and 5 to 8.

```
00001111
11110000
```

The metadata file supplied when describing symmetries that conserve momentum provides details of the lattice geometry as well as the translation vectors for which the system is invariant. The information contained in this file consists of two translation vectors in lattice units in each direction, the normalised version of these vectors and the dimensions of the lattice in each direction. Code block 8 shows an example of what this file looks like for a four by four square lattice with periodic boundary conditions. The number of eigenspaces of the translation operators is then $n_0 n_1$ where $n_0$ and $n_1$ are the numbers of translations possible in each direction. The corresponding eigenvalues of the translation operators for these eigenspaces are $e^{\frac{2\pi i x_0}{n_0}}$ and $e^{\frac{2\pi i x_1}{n_1}}$ where $x_0$ and $x_1$ are integers with $0 \leq x_0 < n_0$ and $0 \leq x_1 < n_1$. DoQO uses numerical labels given by $x_0 + x_1 n_0$ to refer to each of these eigenspaces.

**Code block 8** Example of the metadata file that would be supplied to exploit translational invariance for a four by four square lattice. The LATTICE_VECTOR vectors specify the translation vectors in terms of lattice sites. The NORM_VECTOR vectors correspond to the same vectors except normalised to one and the LATTICE_DIMENSIONS specifies the size of the lattice in each direction.

```
LATTICE_VECTOR1 =  1,0
LATTICE_VECTOR2 =  0,1
NORM_VECTOR1 =  1,0
NORM_VECTOR2 =  0,1
LATTICE_DIMENSIONS =  4,4
```

### 2.2.4 Nearest neighbour exclusion

DoQO has the capability to work in a basis in which all configurations obey a nearest neighbour exclusion condition. This condition only allows configurations in which no two neighbouring sites are occupied. The supersymmetric lattice models discussed in [13] enforce this condition. Being able to work in this restricted basis significantly reduces the computational resource requirements for treating these systems. To enable this feature in DoQO a parameter named 'NN_EXCLUSION' is added to the input file. Code block 9 shows an example of how this parameter is used. An XML file is also provided which provides information about which sites are adjacent. An example of such a file for a four site ring is shown in code block 10. A recursive algorithm that finds these configurations more efficiently has also been implemented and is described in section 3.5.1. The use of this algorithm is enabled by setting the recursive attribute to true.

**Code block 9** Parameter enabling nearest neighbour exclusion using adjacency file file.adj.

```
<SIMULATION>
    <PARAMETERS>
         ....
        <NN_EXCLUSION adjacency_file="file.adj"
        recursive="true">true</NN_EXCLUSION>
         ....
    </PARAMETERS>
</SIMULATION>
```

### 2.2.5 Additional parameters

DoQO supports additional parameters that can be specified in the input file. What follows is a list of the parameters with a short description of each (required parameters are marked with a *).

**MODEL_FILE***: File describing the observable. Example of spin $\frac{1}{2}$ operator in code block 4.

**Code block 10** File providing adjacency information for a four site ring.

```
<?xml version="1.0" encoding="UTF-8"?>
<EDGES number="4">
<EDGE from="1" to="2" ></EDGE>
<EDGE from="2" to="3" ></EDGE>
<EDGE from="3" to="4" ></EDGE>
<EDGE from="4" to="1" ></EDGE>
</EDGES>
```

**TASK_LIST***: The file containing the values of parameters for each task that is to be run. Example in code block 5.

**OUTPUT_PREFIX***: Prefix for output files.

**MODEL_TYPE:** The type of observable that is being used. Currently supported options are SPIN_HALF and FERMIONIC. The default is SPIN_HALF.

**VERBOSITY:** Specifies the level of verbosity to use. For minimal output while the code is running choose a small value and for more detailed output choose a higher value for the verbosity. The range is from zero to fifteen with the default set to one.

**EIGENVALUES:** Number of eigenvalues to retrieve for each diagonalisation. Default is two.

**NN_EXCLUSION:** Specifies that neighbouring sites cannot be occupied. Adjacency information required see section 2.2.4 for more details.

**SAVE_STATES:** Save states to disk for further analysis. Format attribute can be set to ascii, binary or matlab to output vectors in human readable, binary (default) or matlab compatible format.

**SAVE_MATRIX:** Saves each matrix to a file on disk. Again format attribute can be set to ascii, binary or matlab to output vectors in human readable, binary (default) or matlab compatible format.

**BENCHMARK:** Set initial vector for solver to all ones to get constant number of iterations for benchmarking purposes.

**SOLVER_TYPE:** Sets the type of solver to use. Any SLEPc eigenproblem solver type can be used. To select a particular solver to use one specifies the label for that solver. These are 'arnoldi' for Arnoldi, 'lanczos' for Lanczos, 'krylovschur' for Krylov-Schur and 'arpack' for Arpack. The default is to use Krylov-Schur.

**MAX_ITERATIONS:** The maximum number of iterations of the solver method to perform before giving up. Default is 500.

**USE_DISK:** Use shared disk in basis list construction when using symmetries. Details at end of section 2.3.3.1.

**USE_BST:** Use binary sorting tree instead of linked list in exchange of basis indices. For further details see the last paragraph of section 2.3.3.2.

**SOLVER_TOLERANCE:** Tolerance for converged eigenvalues. Default is $10^{-13}$.

**DEGENERACY_TOLERANCE:** Tolerance for degeneracy analysis. Default is $10^{-10}$.

**PHASE_TOLERANCE:** Tolerance for comparing phases when exploiting translational invariance. Default is $10^{-10}$.

As well as the parameters in the input file, it is possible to pass command line arguments to change the behaviour of PETSc and SLEPc. A complete list can be found in the PETSc and SLEPc documentation. Some useful arguments are:

–**eps_monitor:** Option to print detail of convergence after iteration.

–**eps_monitor_draw:** Option to show plot monitoring convergence (requires X11 graphical environment).

–**eps_plot_eigs:** Option to plot approximations of converged eigenvalues (requires X11 graphical environment).

## 2.3 Individual software components

In this section we provide technical details of some components of the DoQO code. It is not essential to know these details to use DoQO. However they provide a deeper understanding of how DoQO works and could shed light on possible performance issues. They are also helpful when attempting to extend the functionality of DoQO.

### 2.3.1 Parallelism

DoQO can take advantage of modern massively parallel distributed memory machines. On these machines multiple DoQO processes are run simultane-

ously on interconnected processing nodes. The key data structures, including the matrix representation of the observable as well as the vectors used during diagonalisation are partitioned among all the processes.

DoQO makes use of MPI as well as the PETSc [26, 27, 28] and SLEPc [29] libraries. The MPI (Message Passing Interface) manages groups of processes as well as allowing communication between the individual processes. PETSc (Portable, Extensible Toolkit for Scientific Computation) is built on top of MPI and provides distributed vectors and matrix data structures as well as functionality for performing basic operations on these efficiently in parallel. One such operation which is of particular relevance to iterative diagonalisation methods is the sparse matrix vector multiplication routine. SLEPc (Scalable Library for Eigenvalue Problem Computations) is a library that leverages the functionality of PETSc and implements a variety of iterative eigensolver methods making use of the underlying PETSc data structures and operations. MPI, PETSc and SLEPc have the added advantage of allowing DoQO to be very portable.

In particular, DoQO has been designed to run on large distributed memory machines, where the amount of memory and the number of cores per node is small, relative to the total amount of memory and number of cores. The IBM Blue Gene machines are an extreme example of this where each node contains only 1-2GB of memory and 2-4 processing cores. In contrast to distributed memory machines there are Shared Memory MultiProcessor (SMP) machines. These possess multiple processors that share a single memory address space. As well as these there is a growing trend of hybrid machines which are distributed memory machines with so-called 'fat' nodes where each node is a small SMP machine. In the case of SMP and hybrid machines it is often possible to fit the vectors entirely within the memory of a single node. This can simplify the implementation of exact diagonalisation techniques and in particular the implementation of matrix free methods (discussed in appendix 2.C). Most other exact diagonalisation codes use this approach. While DoQO still works on these platforms it is on large truely distributed memory machines, with basis sets of tens or hundreds of millions that DoQO really stands out from the majority of other diagonalisation codes.

### 2.3.2 Perfect mapping functions

Perfect mapping functions are used to index subsets of basis elements such that the indices are consecutive integers. DoQO employs such functions when exploiting the symmetries which conserve filling and parity. Here we demonstrate how such functions work in the case of symmetries that conserve filling. It is easily extended for symmetries which conserve parity.

To index all the possible configurations of $c$ particles on $n$ sites a bijective function $L$ is used which maps the set of bit strings with length $n$, containing $c$ ones and $(n-c)$ zeros, to the set of natural numbers less than $\binom{n}{c}$. For example if we have four sites with filling two the function operates as:

$$\begin{aligned} L(0011) &= 0 \\ L(0101) &= 1 \\ L(0110) &= 2 \\ L(1001) &= 3 \\ L(1010) &= 4 \\ L(1100) &= 6 \end{aligned}$$

Given a bit string $b$ of length $n$, with filling $c$ and bits labelled from left to right $b_i, 1 \leq i \leq n$, we can use the index:

$$L(b) = \sum_{i=1}^{n} b_i \binom{n-i}{c - (\sum_{j=1}^{i-1} b_j)}$$

for this bit string, where we have used the recursive relation:

$$\binom{n}{c} = \binom{n-1}{c} + \binom{n-1}{c-1}, \forall c > 0$$

The inverse function which given an index returns the corresponding bit string can also be easily implemented. For $n$ sites and filling $c$ the algorithm is:

1. Set $i = 1$ and set $l$ to the value of the index.

2. if $l < \binom{n-i}{c}$ then $b_i = 0$.

3. If $l \geq \binom{n-i}{c}$ then $b_i = 1$, $l \to l - \binom{n-i}{c}$ and $c \to c - 1$.

4. if $i < n$ increment $i$ and repeat from step two.

These functions can also be used for parity symmetries. In this case we map the set of bit strings with length $n$ containing either an even or odd number of '1's to the set of natural numbers less than $\sum_{c \in P} \binom{n}{c}$ where P is the set of natural numbers less than $n$ with the desired parity.

### 2.3.3 Distributed basis arrays

Where no deterministic method of mapping a basis element to its index (like those described in section 2.3.2) is known a sorted distributed array of basis elements can be used for this purpose. The sections that follow discuss how these arrays are constructed and utilised.

#### 2.3.3.1 Construction of the sorted distributed basis array

When exploiting translational invariance and/or using a model with a nearest neighbour exclusion condition DoQO uses a distributed sorted array to index the basis elements. The valid basis elements (VBE) are stored in this array and indexed by their position within the array. In the context of exploiting translational invariance the VBEs are the representative configurations discussed in section 2.1.1. For models with a nearest neighbour exclusion condition the VBEs are those configurations in which no two adjacent sites are occupied. Two methods for populating these arrays in parallel have been implemented in DoQO. Here we discuss the difficulties involved in populating this array in parallel and how these are overcome.

In the absence of a deterministic method for finding and indexing the VBEs DoQO iterates over the full list of possible basis elements (PBEs) to find the set of all VBEs. Iterating over this list of PBEs in parallel and efficiently populating a sorted distributed array with the VBEs is a non trivial task. This is because the VBEs are not in general uniformly distributed throughout this list.

The first and default method can be broken into three steps:

1. Partitioning the list of PBEs equally among all processes, then iterating over each of these partitions on each process and counting the number of VBEs in each partition. These individual counts are then communicated to all processes using a collective MPI gather operation.

2. These counts are then used to repartition the list of PBEs so that the VBEs are more evenly distributed among the partitions. Each process then iterates over the new partitions and counts the VBEs in each partition. The counts from the first iteration are used where possible to make this process more efficient. The counts are again communicated among all the processes.

3. The counts of VBEs from the different processes are summed to determine the total number of VBEs. From this the size of the portion of the final sorted distributed array on each process is calculated and the space for this array is allocated. Using the counts from each process it is also possible to determine the indices in the final sorted distributed array of the first and last VBEs in each partition. Each process then uses this information and starts iterating from the appropriate position in the list of PBEs to populate its local portion of the final sorted distributed array.

Figure 2.1 illustrates how this process works using four processes with possible full basis set of 256 elements of which 76 are valid basis elements.

The second method makes use of a shared filesystem to create the sorted distributed array. This method performs better in situations where the VBEs are very sparsely distributed over the set of PBEs. This is the case for models with a nearest neighbour exclusion condition. Here the set of PBEs is partitioned as in the first method. Each process then iterates over these partitions but this time writes each VBE it finds to a file on the shared filesystem. Once this has completed each process communicates how many elements it found within its own partition to the other processes. Each process then allocates space for its portion of the array and populates this

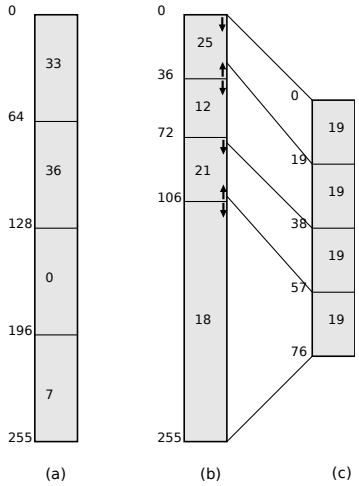



Figure 2.1: Figures showing the construction of the sorted distributed array of basis elements using four processes with 76 elements out of a possible 256 elements. (a) Shows the initial partitioning of possible basis elements amongst the four processes with the indices along the left and the number of elements found within each partition shown within the partition itself. (b) Shows the repartitioned possible basis set and as can be seen the basis elements are more evenly distributed among the partitions. The arrows indicate the places in the possible basis set from which the iterations begin to populate the portions of the final distributed array. At the partition points the correct global indices in the array are known so it is possible to begin from these points on each process simultaneously. (c) Shows the sorted distributed array as it is stored across the memory of the four processes. The lines to (b) show to which range of the possible basis the valid basis elements belong.

by reading from the appropriate positions in the files on the shared filesystem. To enable this method in DoQO the USE_DISK parameter is set to true in the input file as mentioned in section 2.2. This method requires that a shared filesystem is available and its performance will depend on the performance of this filesystem.

#### 2.3.3.2 Using the sorted distributed array

In DoQO the matrix representation of a quantum observable is stored in a distributed sparse matrix data structure provided by PETSc. Each process

has ownership over a range of rows and stores all the non zero elements for those rows. When using a sorted distributed array to index basis elements, the row and column indices of each matrix element $\langle \psi_a | O | \psi_b \rangle$ correspond to the global indices of the basis elements $| \psi_a \rangle$ and $| \psi_b \rangle$ in this array. For each process the indices of the basis elements for each local row are easily obtained from the local portion of the sorted distributed array. However to determine the column indices it is necessary to communicate with the process that has that basis element in its portion of the sorted distributed array. The fact that the distributed array is sorted makes it possible to identify the process on which a particular basis element is stored.

In DoQO the interprocess communication required to exchange indices is done in an organised fashion using only peer to peer communication. In this way synchronisation issues from collective communication operations are avoided and the method is scalable. Each process works out all the basis elements it will need indices for. It then requests the indices for these from the relevant processes. Once these have been retrieved each process can populate its own portion of the matrix. The amount of additional memory required by each process to store these basis elements and indices is in the worst case equal to the amount required to store the non zero elements of the final sparse matrix. This memory is freed up once the matrix has been created and is available during the diagonalisation.

A data structure is required on each process to accumulate the basis elements corresponding to the columns for which indices are required. This data structure stores the basis elements so they are sorted to avoid duplicate entries and to facilitate the communication. Two different data structures can be used for this purpose in DoQO. The default is the doubly linked list implementation from the C++ standard template library and the alternative is the AVL tree implementation from the BOOST intrusive package. The AVL tree is more efficient and should be used where possible. However not all C++ compilers support the BOOST intrusive package so in these cases the doubly linked list can be used.

When using the doubly linked list it takes $O(n^2)$ to insert $n$ items into a sorted list. This results from the fact that it takes $O(n)$ to find the correct

position in the list to insert an item. The maximum length of list possible is the size of the partition of basis elements stored on each process. This is the basis size divided by the number of processes $N/M$. For basis sets of $O(10^9)$ even using $O(10^3)$ processes the individual lists on each process will be $O(10^6)$. The number of operations required to fill each of these lists is on the order of trillions or $O(10^{12})$. This very quickly becomes the main performance bottleneck.

An AVL tree is a self balancing binary search tree which takes $O(nlog(n))$ to insert $n$ items. This is a significant improvement over the $O(n^2)$ it takes to insert $n$ items into the doubly linked list. This feature is enabled by setting the USE_BST parameter to true in the DoQO input file. DoQO must be compiled with BOOST for this to work. See README for further details in relation to compiling DoQO with BOOST.

## 2.4 Performance

A performance study of DoQO was undertaken to determine the most relevant factors influencing the performance of DoQO. A benchmark system was chosen and results were obtained for a variety of lattice sizes and numbers of processors on different machines.

The benchmark system chosen was the Kitaev honeycomb lattice model [5] (see chapter 4) with parameters $J_x = -0.1, J_y = -0.45, J_z = -0.45$. These parameters were chosen to avoid any special points of the model's phase diagram. Two eigenvalues were calculated using the Krylov-Schur algorithm for systems of 16, 20, 24 and 28 spins and to keep the number of iterations constant for each system size the initial vector was set to all ones (set BENCHMARK parameter to true in the input file).

The machines used were an Opteron based gigabit ethernet cluster, an IBM Blue Gene/P and a Xeon based SGI ICE cluster with ConnectX Infiniband interconnect. Plots of the strong scaling are shown in figures 2.2, 2.3 and 2.4. The speedup in each case is calculated as $s = \frac{mt_m}{t_n}$ where $n$ is the number of processors for which the scaling is being calculated, $m$ is the minimum number of processors on which the given system can be treated on

and $t_i$ is the time taken on $i$ processors.

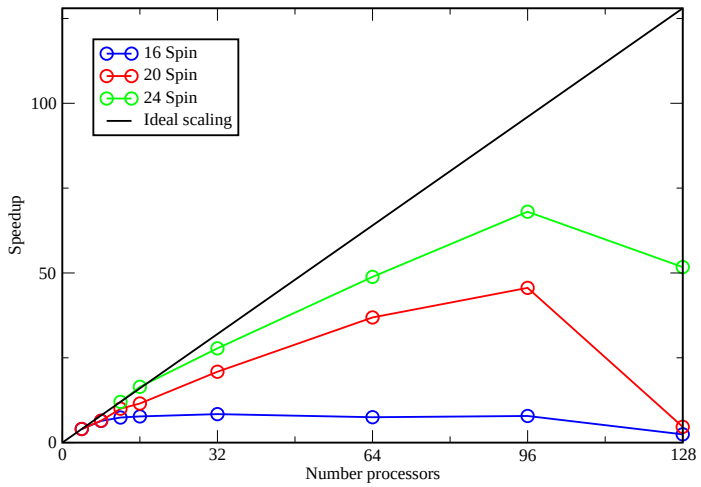



Figure 2.2: Scaling behaviour of benchmark on Opteron basis gigabit ethernet cluster using up to 128 cores.

On the Opteron based gigabit ethernet cluster (figure 2.2) a speedup is observed for the 20 and 24 spin systems up to 96 processors. This machine consists of nodes with two AMD Opteron 250 2.4GHz processors and 4GB or memory connected by a core routed gigabyte ethernet switch.

On the Blue Gene/P system (figure 2.3) impressive scaling is observed all the way up to 2048 cores for 24 and 28 spin systems. The Blue Gene/P is made up nodes consisting of four PowerPC 450 cores running at 850 MHz and 2GB of memory. These are connected with multiple networks including a high speed low latency 3D toroidal network used for peer to peer communication. Peaks can be seen in the plot at 512 and 1024 cores and troughs in-between. It is thought that these are due to the fact that there are 512 cores on each mid-plane and thus a more optimal mapping of the cores occurs when using multiples of 512 cores [31].

On the SGI ICE machine (figure 2.4) good scaling is observed for the 28 spin system up to 2048 cores. For the 24 spin system the speedup drops off

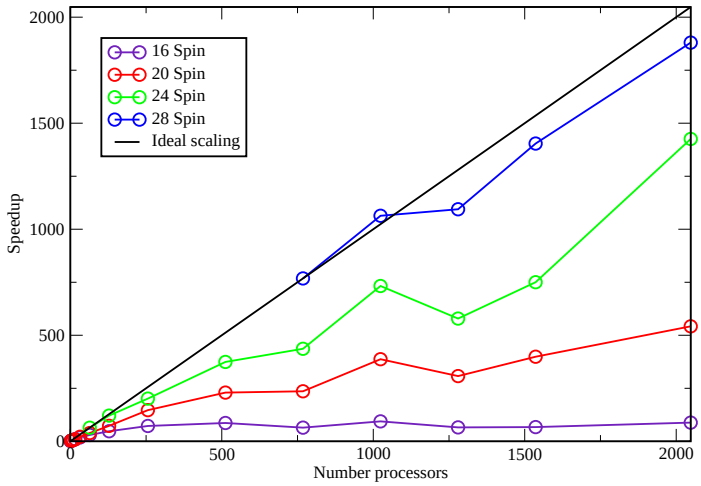



Figure 2.3: Scaling behaviour of benchmark on Blue Gene/P using up to 2048 cores.

towards 2048 cores. The nodes of this machine each consist of two quad core Intel Xeon E5462 processors running at 2.8GHz and 12GB of memory. These are connected via a ConnectX Infiniband interconnect.

### 2.4.1 Performance conclusions

From the performance study undertaken we conclude that the principle factors that influence the performance of DoQO are the performance of the communications network and the available memory bandwidth.

The matrix vector multiplication operation which is the central operation in DoQO involves significant interprocess communication. The gigabit ethernet interconnect has substantially higher latency and lower bandwidth than the interconnects found on the Blue Gene/P and the SGI ICE machines. This explains the poor scaling performance past 64 processors which was observed. The multiple special purpose networks used on the Blue Gene/P for interprocess communication accounts for the superior scaling behaviour observed.

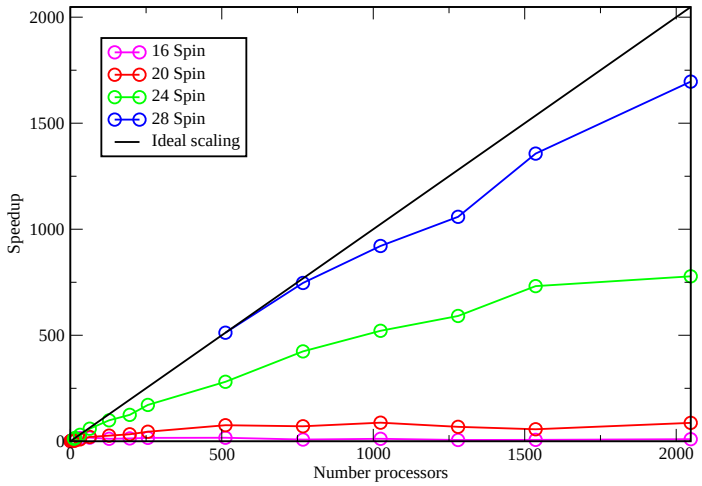



Figure 2.4: Scaling behaviour of benchmark on SGI ICE system using up to 2048 cores.

With sparse matrix vector multiplication the memory bandwidth is central to ensuring that the processing cores are fully utilised. Even though each core on the SGI ICE is significantly faster than those found on the Blue Gene/P the time taken for the calculations on the same numbers of cores is comparable. This perhaps indicates that the Xeon cores are not being fully utilised due to insufficient memory bandwidth. Evidence of this is observed by running DoQO on two and four cores of a Xeon quad core processor. There is almost no perceptible gain in speed while using four cores over two cores.

## 2.A Appendix: Matrix memory requirements

Here we discuss how to work out an upper bound on the amount of memory required to store the non zero elements of a matrix operator. Each term of an operator can contribute at most one non-zero entry per row.

Matrices are stored using Compressed Sparse Row (CSR) format. To

store the matrix each non zero entry requires an integer for the column index, a double for storing the value (two doubles in the case of complex values) and for each row an integer is required to point to the starting position in the values and columns arrays for that row. For systems with thirty two particles and above, eight byte integers are required to accommodate the column and row indices but for smaller systems four byte integers are sufficient.

If $n$ is the number of spins in the system and $t$ is the number of terms then the upper bound on the amount of memory required to store all the non zero elements using double and complex arithmetic and four and eight byte integers are listed below (where D indicates double arithmetic and C complex arithmetic and the four and eight specify the number of bytes used for storing integers in each case).

$$M_{D4} = 2^n(12t + 4)$$
$$M_{C8} = 2^n(16t + 8)$$
$$M_{D4} = 2^n(20t + 4)$$
$$M_{C8} = 2^n(24t + 8)$$

As well as requiring memory to store the non zero elements, additional memory is required by PETSc to store metadata. This data enables efficient matrix vector multiplication operations in parallel. The amount of memory required for this depends on the number of processes and the structure of the matrix. Typically it is on the order of an additional $20\% - 30\%$.

## 2.B Appendix: Operator Construction

The matrices representing the operators are generally extremely sparse. These matrices are stored in sparse matrix format where only the non zero elements and their indices are stored. This data is distributed uniformly across the available processes.

### 2.B.1 Spin half systems

This section provides explicit demonstrations of how quantum operators are built up for the benefit of readers that are not familiar with these concepts. The matrix for a spin $\frac{1}{2}$ operator for a finite quantum system is made up of a sum of terms. Each term can be a single site term or an interaction term acting on multiple sites. Terms are written in terms of the Pauli matrices and the identity matrix. The Pauli matrices used for spin half particles are:

$$\sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \sigma^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

The term $\sigma_1^x \sigma_2^y$ is an interaction term between the spin at position one and the spin at position two. The matrix corresponding to this term results from taking the tensor product of the matrices for the individual operators.

$$\sigma_1^x \sigma_2^y := \sigma_1^x \otimes \sigma_2^y = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & i & 0 \\ 0 & -i & 0 & 0 \\ i & 0 & 0 & 0 \end{pmatrix}$$

When constructing the matrix for a term implicit identity matrices are used for the sites not mentioned. For a three spin system the term $\sigma_1^x \sigma_2^y$ becomes $\sigma_1^x \otimes \sigma_2^y \otimes I_3$.

The column index and value of the non zero entry corresponding to a given term on a given row can be calculated easily. The process is as follows;

1. Set the column index to the given row index. Set the non zero value to one.

2. For each $\sigma^x$ and $\sigma^y$ operator in the term flip the appropriate bit in the binary representation of the column index. The bit to flip is the one in the position corresponding to the site index of the $\sigma^x$ or $\sigma^y$ operator in question. A bitwise exclusive or operator can be used to do this efficiently.

3. For each $\sigma^z$ operator in the term check if there is a 1 in the binary representation of the row index in the position on which the $\sigma^z$ acts. If there is multiply the value by minus one.

4. For each $\sigma^y$ operator in the term check if there is a 1 in the binary representation of the column index in the position on which the $\sigma^y$ acts. If there is multiply the value by $-i$ and if not then multiply the value by $i$.

Using this process one can determine the values and column indices of all the non zero values for a given row. Each process then loops over its local chunk of rows and sets the non zero values for those rows without any communication with the other processes.

## 2.C Appendix: Matrix free methods

Matrix free methods refer to diagonalisation methods which do not store matrix elements, but calculate them as they are needed. Significant savings in memory result from using these methods which can enable calculations for larger systems to be performed. These methods were investigated in the context of DoQO, however they are not currently used due to complications encountered while attempting to achieve acceptable scalability, whilst maintaining the desired generality to deal with arbitrary spin half and spinless fermionic systems, whilst at the same time exploiting physical symmetries. It is hoped that future versions of DoQO will use these matrix free methods. Here we discuss some of the issues encountered.

At the core of iterative exact diagonalisation techniques is the matrix vector multiplication operation. For matrix free methods this operation is performed by calculating the matrix elements as they are needed rather than storing them in memory. While this is straightforward to implement for serial architectures the intensive interprocess communication involved in parallel implementations causes complications when using large numbers of processes.

In a parallel implementation each vector is divided into sub vectors of dimension $n = \frac{N}{p}$ where $N$ is the dimension of the vector and $p$ is the number

of processes. Each sub vector is stored in the memory of a different process with processes labelled $0, 1, 2, ..., p-1$. We label the sub vector of vector $x$ stored by process $i$ with $x_i$ and the elements of this sub vector are labeled $(x_i)_j$ with $0 \leq j < n$. While for matrix free methods the matrices are not stored we will use a similar notation for matrices. In this case there are $p^2$ sub matrices each of dimension $n \times n$. The sub matrix $A_{ij}$ with $0 \leq i, j < p$ of matrix $A$ contains elements from $A$ with row indices in the range $[in, (i+1)n)$ and column indices in the range $[jn, (j+1)n)$. Elements of the sub matrix $A_{ij}$ are labelled $(A_{ij})_{kl}$ with $0 \leq k, l < n$.

Using this notation for the matrix vector multiplication $Ax = y$ the elements of the sub vector $y_i$ are given by:

$$(y_i)_j = \sum_{k=0}^{p-1} (\sum_{l=0}^{n-1} (A_{i,k})_{j,l} (x_k)_l$$

This expression shows that the elements of each sub vector of $y$ can depend on all the sub vectors of the $x$ vector [1]. To perform this operation in parallel there are two approaches which can be taken which are:

1. Each process $i$ multiplies the values from the sub vector $x_i$ by the appropriate matrix elements from the sub matrices $A_{j,i}, 0 \leq j < p$ and sends the products to the appropriate processes to add to the values of the sub vectors $y_j, 0 \leq j < p$.

2. Each process $i$ retrieves the values of the sub vector $x_j, 0 \leq j < p$ from the process $j$, multiplies these by the appropriate matrix elements from sub matrices $A_{i,j}, 0 \leq j < p$ and adds the products to the values of the sub vector $y_i$.

These approaches are roughly equivalent and suffer many of the same issues. The first approach has advantages resulting from the fact that it is cheaper to send a specific group of values to another process than it is to retrieve a specific group of values, as this requires an additional request for the values. In

[1] Dependent on structure of the matrix A. $y_i$ depends on $x_j$ if the sub matrix $A_{i,j}$ contains non zero elements.

light of this we examine further how the first approach may be implemented and the issues encountered.

The most straightforward means of implementing this approach is to use the PETSc function MatSetValues to add the products to the sub vectors of the $y$ vector on the relevant processes. When this function is called it buffers the indices and values that are to be added and then when the VecAssemblyBegin and VecAssemblyEnd functions are called the values are sent to the relevant processes. The process used to send the values consists of a collective all to all communication operation which communicates from all processes to all processes how many values are to be sent from each process to each process. A series of peer to peer exchanges are then used to send the actual values. This technique does not scale well to large numbers of processes. This is due mainly to synchronisation issues which mean that many processes are left idle while waiting for other processes. There is freedom to choose how often the assembly functions can be called. By allowing some of the values to be buffered and calling the assembly functions less the efficiency can be improved somewhat. However if all the values were buffered then almost as much memory is needed as is required to store the sparse matrix entries and no significant advantage is gained by using matrix free methods.

To overcome the synchronisation issues described in the above paragraph the use of one sided communication operations which are part of the MPI-2 standard were investigated. However it was found that in practise due to the implementation details these operations do not offer any advantage. The use of the Global Arrays package [32] was also investigated and shows promise. However this package was not available on the machines being used and a lot of work would be required to use this package along with the PETSc and SLEPc libraries.

Another approach aimed at overcoming the synchronisation issues described above was devised. This approach involves only the use of peer to peer communication. In this approach each process calculates the values that it is to send to one other process. Process 0 calculates the values to send to process 1, process 1 to 2 and so on up to process $p-1$ to process 0. These values are then sent using an MPI peer to peer communication operation.

Each process then calculates the values to send to the next process and sends them. This would be process 0 calculates the values for process 2, process 1 for process 3 and so on. This continues through $p-1$ iterations. This method is much more scalable as all communication is peer to peer. However it is necessary to calculate all the non zero elements of a sub matrix $A_{i,j}$ of A at each step. When using the method described in section 2.B for calculating the non zero matrix elements it is natural to calculate all the non zero elements for a given column. Thus to calculate the non zero elements of a sub matrix $A_{i,j}$ one calculates the row indices of each of the non zero elements from the columns with indices in the range $[jn, (j+1)n)$. Unless the row indices lie within the range $[in, (i+1)n)$ they are discarded. The amount of row indices that must be calculated and subsequently discarded scales with the number of processes being used and as a result becomes a bottleneck. Figure 2.5 shows results of benchmarks which show that for a system with basis size of $2^{24}$ this technique scales reasonably well up to approximately 100 processes but after this no speedup is observed. It is possible that improvements could be achieved by optimising further the code that calculates and evaluates the row indices or performing pre processing steps to allow the non zero elements of a given sub matrix to be calculated more directly.

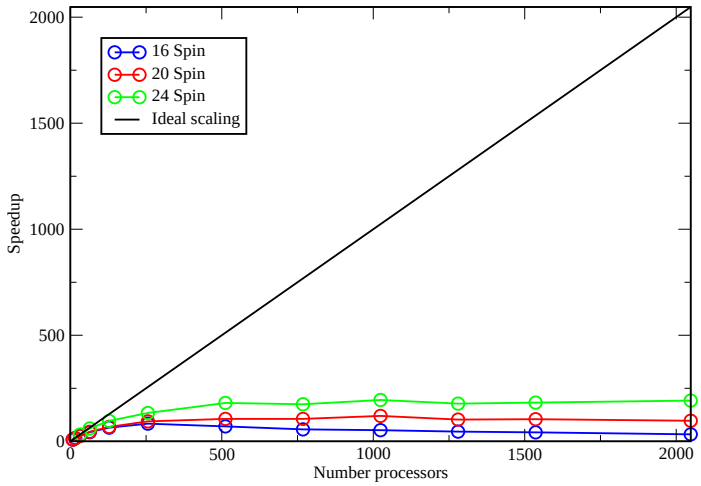



Figure 2.5: Scaling behaviour of benchmark on Blue GeneP up to 2048 cores with matrix free method.

When exploiting symmetries for which there are no perfect mapping functions 2.3.2 to map basis elements to their indices a distributed array 2.3.3 is used. This further complicates the implementation of matrix free methods as extra communication is required to determine the indices of matrix elements. This can be performed in a pre processing step as is done for the matrix construction described in section 2.3.3.2 however this introduces significant memory overheads which significantly reduce the advantage of using matrix free methods in the first place.

# Chapter 3

# Supersymmetric lattice models

The supersymmetric (SUSY) lattice models discussed here were first introduced by Fendley et al. [13] and can be thought of as a generalisation of supersymmetry from single particles to multi-particle states on a lattice. For these models it is the overall state of the system that is fermionic or bosonic, corresponding to states containing odd or even numbers of fermions respectively. These models are extremely rich in the physics they exhibit. Among the phases believed to exist in these models are critical, 'super-frustrated' and 'super-topological' phases.

In this chapter we introduce the SUSY lattice models, review some of what is known to date and present results of numerical calculations that have been performed for these models. In section 3.1 the Hamiltonian for the SUSY lattice models is defined and explained along with properties of the spectrum and useful commutation relations. This section also describes briefly the use of the Witten index, cohomology techniques as well as transfer matrix methods. In section 3.2 the SUSY chain is explored. This critical model has been studied extensively [13, 33, 11, 34] and here we show numerical results which support this work and allows us to verify the correctness of our numerical tools for treating these models. The staggered SUSY chain [35] is discussed in section 3.3. For this model we focus in particular on the staggering limits. In section 3.4 we explore the square octagon lattice. For this lattice we look at the effect of adding defects to the plaquettes and compare

the properties of the numerically calculated ground states to those of trial wavefunctions named Projected Product States (PPS) wavefunctions. Finally in section 3.5 we review issues encountered when performing numerical calculations for these models.

## 3.1 Definition and introduction

The Hamiltonian for the supersymmetric lattice models we study here is defined as the anti-commutator of the nilpotent supercharge operators $Q$ and $Q^\dagger$.

$$H = \{Q^\dagger, Q\} \tag{3.1}$$

The $Q^\dagger$ ($Q$) operator is defined as the sum of spinless fermionic creation (annihilation) operators times the projector $P_{\langle i \rangle}$ at each site.

$$Q^\dagger = \sum_i c_i P_{<i>}$$

$$Q = \sum_i c_i^\dagger P_{<i>}$$

The projector $P_{\langle i \rangle} = 1$ if all sites neighbouring site $i$ are vacant and zero otherwise. These projectors ensure that no two adjacent sites can be occupied which is known as nearest neighbour exclusion.

$$P_{\langle i \rangle} = \prod_{j \ next \ to \ i} P_j = \prod_{j \ next \ to \ i} (1 - c_j^\dagger c_j) \tag{3.2}$$

We now show explicitly how the Hamiltonian can be written as kinetic and potential parts by expanding the general form of the Hamiltonian (equation 3.1) in terms of the fermionic creation and annihilation operators as well as

the projectors $P_{\langle i \rangle}$.

$$\begin{aligned}
H =& Q^\dagger Q + QQ^\dagger \\
=& \sum_i \sum_k [c_i^\dagger P_{<i>} c_k P_{\langle k \rangle} + c_i P_{\langle i \rangle} c_k^\dagger P_{\langle k \rangle}] \\
=& \sum_i P_{\langle i \rangle} [c_i^\dagger c_i + c_i c_i^\dagger] + \sum_i \sum_{k \ not \ next \ to \ i} P_{\langle i \rangle} [\cancel{c_i^\dagger c_k + c_i c_k^\dagger}] P_{\langle k \rangle} \\
&+ \sum_i \sum_{k \ next \ to \ i} P_{\langle i \rangle} [c_i^\dagger c_k + \cancel{c_i c_k^\dagger}] P_{\langle k \rangle} \\
=& \sum_i P_{\langle i \rangle} + \sum_i \sum_{k \ next \ to \ i} P_{\langle i \rangle} c_i^\dagger c_k P_{\langle k \rangle}
\end{aligned}$$

The simplifications where $c_i^\dagger c_i + c_i c_i^\dagger = 1$ and $c_i^\dagger c_k + c_i c_k^\dagger = 0$ result from the canonical fermionic anti-commutation relation $\{c_i, c_k^\dagger\} = \delta_{i,k}$. The fact that $P_{\langle i \rangle} c_i c_k^\dagger P_{\langle k \rangle} = 0$ when site $k$ is next to site $i$ can be recognised by noticing that the projector $P_{\langle i \rangle}$ will include $(1 - c_k^\dagger c_k)$ in the product when site $k$ is next to site $i$ so expanding we get:

$$P_{\langle i \rangle} c_i c_k^\dagger P_{\langle k \rangle} = [\prod_{j \ next \ to \ i, j \neq k} (1 - c_j^\dagger c_j)](1 - c_k^\dagger c_k) c_i c_k^\dagger P_{\langle k \rangle}$$

Then taking out the terms acting on site $k$ we get:

$$\begin{aligned}
&(1 - c_k^\dagger c_k) c_i c_k^\dagger = c_i c_k^\dagger - c_i c_k^\dagger c_k c_k^\dagger = c_i c_k^\dagger - c_i c_k^\dagger (1 - c_k^\dagger c_k) \\
&= c_i c_k^\dagger - c_i c_k^\dagger + 0 = 0
\end{aligned}$$

The outcome of this expansion is that we can write the Hamiltonian as the sum of kinetic and potential terms.

$$H = H_{kin} + H_{pot}$$

The kinetic part is written as:

$$H_{kin} = \sum_i \sum_{k \ next \ to \ i} P_{\langle i \rangle} c_i^\dagger c_k P_{\langle k \rangle} \tag{3.3}$$

and contains hopping terms between neighbouring sites where the projection operators ensure that the hopping is permitted by the nearest neighbour exclusion condition. The potential part is written as:

$$H_{pot} = \sum_i P_{\langle i \rangle} \tag{3.4}$$

and assigns energy to each configuration. More fermions are favoured energetically but there is an energy penalty for placing fermions too close together.

The spectrum is positive definite with ground state energy equal to zero[1].

$$\begin{aligned} \langle \psi | H | \psi \rangle &= \langle \psi | Q^+ Q^- + Q^- Q^+ | \psi \rangle \\ &= |Q^- | \psi \rangle|^2 + |Q^+ | \psi \rangle|^2 \end{aligned}$$

### 3.1.1 Commutation relations

We now discuss some important commutation relations.

$$[H, F] = 0 \tag{3.5}$$

Here $F = \sum_i n_i$ with $n_i = c_i^\dagger c_i$ is the total number of fermions. This relation tells us that the total number of fermions is conserved. Why this is so can be seen by noting that the Hamiltonian is made up of hopping terms and number operators only, neither of which changes the number of fermions in a given configuration. This can be more clearly seen when the Hamiltonian in written in terms of the spinless fermionic creation and annihilation operators (equations 3.3 and 3.4).

$$[H, Q^{(\dagger)}] = 0 \tag{3.6}$$

The Hamiltonian also commutes with the supercharge operators $Q^{(\dagger)}$ which results from the fact that these operators are nilpotent ($(Q^{(\dagger)})^2 = 0$) and the Hamiltonian is the anti-commutator of these operators. This commutation relation indicates that states with finite energy form doublets which for the

[1]Ground state energy can be non zero when the supersymmetry is broken [33]

appropriate chooses of basis can be written as $\{|\,\psi\rangle, Q|\,\psi\rangle\}$ or $\{|\,\psi\rangle, Q^\dagger|\,\psi\rangle\}$. If $|\,\psi\rangle$ is an eigenstate of the Hamiltonian then the following holds.

$$\begin{aligned} H|\,\psi\rangle &= E|\,\psi\rangle \\ Q(H|\,\psi\rangle) = Q(E|\,\psi\rangle) &= H(Q|\,\psi\rangle) = E(Q|\,\psi\rangle) \end{aligned}$$

The same can be done for $Q^\dagger$.

$$[F, Q] = -Q$$

$$[F, Q^\dagger] = Q^\dagger$$

The supercharge operators do not commute with $F$. It is straightforward to derive these commutation relations.

$$\begin{aligned} [F, Q] = FQ - QF &= \sum_i n_i \sum_j c_j P_{\langle j\rangle} - \sum_j c_j P_{\langle j\rangle} \sum_i n_i \\ &= \sum_{i\neq j} \cancel{[n_i c_j P_{\langle j\rangle} - c_j P_{\langle j\rangle} n_i]} + \sum_i \cancel{[n_i c_i P_{\langle i\rangle}} - c_i P_{\langle i\rangle} n_i] \\ &= -\sum_i c_i c_i^\dagger c_i P_{\langle i\rangle} = -\sum_i [c_i - \cancel{c_i^\dagger c_i c_i}] P_{\langle i\rangle} = -Q \end{aligned}$$

The cancelations are a direct result of the fermionic anti-commutation relations. The same can be done for the commutator $[F, Q^\dagger]$. As a result, applying one of the supercharge operators to a state changes the number of fermions in that state:

$$\begin{aligned} H|\,\psi\rangle &= f|\,\psi\rangle \\ QF|\,\psi\rangle = Q(f|\,\psi\rangle) &= (FQ + Q)|\,\psi\rangle = f(Q|\,\psi\rangle) \\ F(Q|\,\psi\rangle) &= (f-1)(Q|\,\psi\rangle) \end{aligned}$$

The same can be done for $Q^\dagger$.

### 3.1.2 Witten Index

The Witten index [36] is defined as $W = tr[(-1)^F e^{-\beta H}]$ where the trace is over the entire Hilbert space, $\beta = \frac{1}{kT}$ is the inverse temperature and $F = \sum_i n_i$ is the total number of fermions. The absolute value of this gives a lower bound on the number of zero energy ground states for a given SUSY lattice model. States with finite energy always have a super-partner with the same energy. This means that all the finite energy states cancel and what is left is a trace over the singlet ground states. For certain lattice geometries not all ground states occur at the same filling resulting in cancelations between ground states. This is why the Witten index only provides a lower bound on the number of ground states. In [33] results of numerical calculations of the Witten index for different lattice geometries are presented.

### 3.1.3 Cohomology

Cohomology can be used to provide accurate information about the ground state degeneracy of SUSY lattice models [37]. The ground states of these models are singlet states that are annihilated by $Q$ and $Q^\dagger$. The kernel of an operator consists of all states which are annihilated by the operator, so all ground states are in $ker(Q)$. Singlet states also cannot be the result of applying $Q$ or $Q^\dagger$ to any state. The image of an operator contains all states which can result from the action of that operator on a state. Thus singlet states are not part of the $Im(Q)$. The ground states are all states in the kernel of $Q$ but not in the image. This is the definition for the cohomology of $Q$ which is $H_Q = kerQ/ImQ$. Thus the dimension of the ground state space of these models is the same as the dimension of the cohomology class of $Q$. In many lattices the dimension of the cohomology class can be calculated analytically [37].

### 3.1.4 Transfer matrices

Transfer matrix methods can be used to calculate the Witten index, the partition function and also the dimension of the Hilbert space at each filling

for SUSY lattice models [37, 11, 33, 38, 39]. For the closed 1D chain with L sites the partition function is given by:

$$Z(z) = tr(\mathcal{T}^L) = tr\left(\begin{pmatrix} 1 & 1 \\ z & 0 \end{pmatrix}^L\right)$$

The basis elements on which the transfer matrix acts correspond to the two possible configurations on a site which are empty and occupied respectively. The matrix elements of the transfer matrix $\mathcal{T}$ are given by $\mathcal{T}_{ij} = (1-\delta_{2,n_i+n_j})z^{n_i}$ where $n_1 = 0$ and $n_2 = 1$ are the number of fermions in basis configuration one and two respectively. The top left entry corresponds to adding an empty site to an empty site, the bottom left to adding an occupied site to an empty site, the top right an empty site to an occupied site and the bottom right an occupied site to an occupied site. Adding an occupied site to an occupied site is not permitted by the nearest neighbour exclusion condition and thus the bottom right matrix element is zero.

Setting the fugacity $z = -1$ results in the Witten index and setting $z = 1$ gives the dimension of the full Hilbert space. If the calculation is performed symbolically the resulting polynomial in $z$ tells the size of the Hilbert space at each filling. Here the coefficient of the $z^n$ term is the size of the Hilbert space at filling $n$. For example for the closed six site chain this works out to be:

$$Z(z) = 1 + 6z + 9z^2 + 2z^3$$

which tells the size of the Hilbert space at fillings $0, 1, 2$ and $3$ are $1, 6, 9$ and $2$ respectively. Being able to calculate the dimension of the Hilbert space is useful when trying to work out whether or not calculations are tractable. Table 3.12 in appendix 3.D gives the dimension of Hilbert spaces for SUSY models on various lattices calculated using transfer matrices.

For the open 1D chain where the end sites are not connected we take the transfer matrix to the power of $L - 1$, multiply by a vector $c = (1, z)$ and

then sum the entries of the resulting vector.

$$Z(z) = \sum_{i,j=1}^{2} (\mathcal{T}^{L-1})_{ij} c_j$$

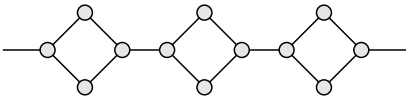


Figure 3.1: Square octagon chain.

Transfer matrices can also be used for more complicated lattices. For the square octagon chain, as shown in figure 3.1, the transfer matrix for adding a plaquette to the chain is:

$$\begin{aligned} \mathcal{T} = ABA &= \begin{pmatrix} 1 & 1 \\ z & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ z^2 + 2z & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ z & 0 \end{pmatrix} \\ &= \begin{pmatrix} 1 + 3z + z^2 & 1 + 2z + z^2 \\ z + z^2 & z \end{pmatrix} \end{aligned} \tag{3.7}$$

The matrix $A$ is the same as the transfer matrix for the 1D chain and adds a single site to the chain, whereas the $B$ matrix gives the weights for adding the two sites which are at the top and bottom of each plaquette. Multiplying them together gives the transfer matrix for adding a complete plaquette to the chain. The partition function in this case for a closed chain with $N$ squares is:

$$Z(z) = tr(\mathcal{T}^N)$$

This method can also be used for 2D lattices in which case each transfer matrix becomes a row to row transfer matrix. Here the weight of the entries corresponds to the weight of adding a particular row configuration to another row configuration. Calculations of the Witten index for various 2D lattices using this technique were carried out in [33].

For the square octagon lattice on a cylinder with two columns and $N$ rows in the direction that wraps around the cylinder the partition function

is given by:

$$Z(z) = tr((ABA)^N)$$

where $A$ is the transfer matrix for adding two sites and $B$ is the transfer matrix for adding four sites. These matrices are:

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ z & z & 0 & 0 \\ z & 0 & z & 0 \\ z^2 & 0 & 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2z+z^2 & 0 & 2z+z^2 & 0 \\ 2z+z^2 & 2z+z^2 & 0 & 0 \\ 3z^2+2z^3 & 0 & 0 & 0 \end{pmatrix}$$

## 3.2 SUSY chain

The 1D SUSY chain is the simplest SUSY lattice model [13] and has been studied extensively. Here we review some of the properties of this model and show some numerical results which serve to support the existing work and to confirm the accuracy of our numerical tools for treating these models.

On a chain each site $i$ has two neighbouring sites $i-1$ and $i+1$. As a result the projection operators $P_{\langle i \rangle}$ are defined[2] as $P_{\langle i \rangle} = P_{i-1}P_{i+1} = (1 - n_{i-1})(1-n_{i+1})$. Putting this into the expressions for the kinetic and potential parts shown in equations 3.3 and 3.4 the Hamiltonian is written as:

$$\begin{aligned} H &= H_{kin} + H_{pot} \\ H_{kin} &= \sum_{i=1}^{L} [P_{i-1}(c_i^\dagger c_{i+1} + c_{i+1}^\dagger c_i) P_{i+2}] \\ H_{pot} &= \sum_{i=1}^{L} P_{i-1}P_{i+1} = \sum_{i=1}^{L} (1 - n_{i-1})(1 - n_{i+1}) \\ &= \sum_{i=1}^{L} (1 - n_{i+1} - n_{i-1} + n_{i-1}n_{i+1}) = N - 2F + \sum_i n_{i-1}n_{i+1} \end{aligned}$$

[2]For a closed chain with $n$ sites site $n+1$ is site 1 and site $-1$ is site $n$. For the open chain the sites at the ends only have one neighbouring site.

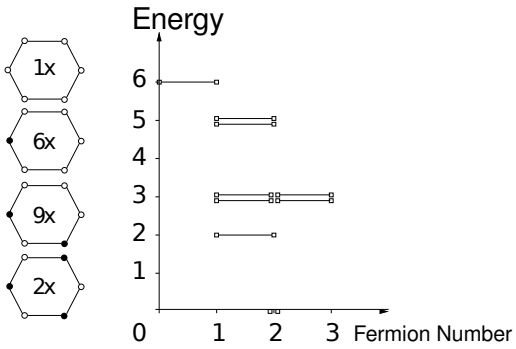



Figure 3.2: Plot of energy spectrum of six site closed SUSY chain against fermion number. Super-partners are indicated by connected points and illustrations on the left show the number of configurations at each filling. Figure taken from [40].

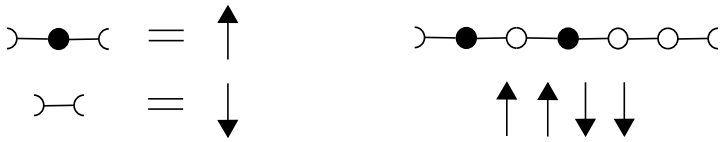
If there are one or more zero energy singlet ground states we say that SUSY is unbroken [36], otherwise we say the SUSY is broken. For the 1D SUSY chain the number of zero energy ground states has been computed using the Witten index and cohomology [13] (details [11]). For a closed SUSY chain there are two zero energy ground states when the chain length $L \mod 3 = 0$ and one zero energy ground state otherwise. For the open SUSY chain there is a single zero energy ground state when $L \mod 3 = 0, 2$. For an open chain with $L \mod 3 = 1$ the SUSY is broken and there are no zero energy singlet ground states. Figure 3.2 shows the spectrum of the closed six site SUSY chain. The doublet finite energy states and two singlet ground states are clearly visible here.

In the thermodynamic limit the closed 1D SUSY chain is critical (see appendix 3.C) and is described by the simplest $\mathcal{N} = (2, 2)$ superconformal field theory with $c = 1$ [11]. The $\mathcal{N} = (2, 2)$ means that both the holomorphic as well as the anti-holomorphic fields satisfy an $\mathcal{N} = 2$ superconformal algebra. The open 1D SUSY chain is also critical in the thermodynamic limit but is described by an $\mathcal{N} = 2$ superconformal field theory.

It is possible to map the SUSY chain to the Heisenberg XXZ chain with anisotropy parameter $\Delta = -\frac{1}{2}$ [41, 13, 33] (up to an overall constant and with twisted boundary conditions). Figure 3.3 shows the mapping between sites of the SUSY chain and the spins of the XXZ model. SUSY chains at

different fillings correspond to XXZ spin chains of different lengths. The XXZ chain has Hamiltonian:

$$H_{XXZ} = \frac{1}{2}\sum_{i}^{L}[\sigma_i^x\sigma_{i+1}^x + \sigma_i^y\sigma_{i+1}^y - \Delta\sigma_i^z\sigma_{i+1}^z]$$



(a) (b)

Figure 3.3: Mapping of sites to spins. (a) Each occupied site along with its links to the neighbouring unoccupied sites are mapped to the up spin and each link between unoccupied sites is mapped to a down spin. (b) An example showing the mapping from a six site chain to spins.

### 3.2.1 Computations

Finite size scaling (FSS) calculations of the energy gap were performed for closed SUSY chains with lengths $L = 3k$ where $3 \leq k \leq 13$. The energy gap is defined as the difference in energy between the ground state and first excited state. Figure 3.4 shows the energies of the first and second excited states plotted against the inverse chain length. Also included in the plot is the expected scaling of the energy gap obtained from superconformal field theory (SCFT) [11]. This is given by the expression $E = \frac{2\pi E_{SCFT} v_F}{L}$ with a correction on the order of the inverse square length, $E_{SCFT} = \frac{2}{3}$ and $v_F = \frac{9\sqrt{3}}{4}$. It is also known that the first excited state will be doubly degenerate in the infinite limit. From the plot it is clear that the difference in energy between the first and second excited states goes to zero and that as the system size is increased the energy gap converges to that predicted by SCFT and goes to zero in the limit of infinite system size. The fact that the numerically calculated energy values show good agreement with the analytical description confirms that DoQO is working correctly for these models. Additionally the fact that the ground state energy is identically zero and that all finite energy states have

super-partners is further proof that the numerically calculated spectral values are correct.

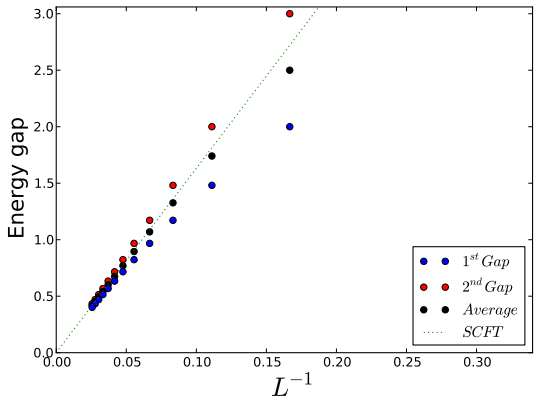



Figure 3.4: First and second excitation energies and the average for the closed SUSY chain plotted with inverse chain length. Expected behaviour from SCFT also included.

By exploiting the translational invariance of the closed SUSY chain the momentum of each state in the spectrum can be determined. Figure 3.5 shows the energy (rescaled) of the low lying states of the closed SUSY chain with $L = 45$ and filling $f = 15$ plotted against momentum. The 'tower of states' structure expected of conformally critical systems is evident here.

From the analysis given in [11] we know that for the closed SUSY chain with length $3k, k \in \mathbb{Z}$ , the highest weight states with the lowest energies are $V_{0,\pm\frac{1}{2}}|\,0\rangle$ and $V_{0,\pm\frac{3}{2}}|\,0\rangle$. These states have conformal dimensions $h_{L,R} = \frac{1}{24}$ and $h_{L,R} = \frac{3}{8}$ respectively (see appendix 3.C for information on conformal dimensions). The energy is given by the expression $E = h_L + h_R - \frac{c}{12}$ so the energies of the states $V_{0,\pm\frac{1}{2}}|\,0\rangle$ are 0 and the energies of the states $V_{0,\pm\frac{3}{2}}|\,0\rangle$ are $\frac{2}{3}$. To relate these energies to the numerically calculated energy spectrum we rescale the numerically calculated energy spectrum such that the first gap is $\frac{2}{3}$ [42].

These highest weight states have descendants which result from acting on these states with the generators of the Virasoro algebra $L_{L,-n}$ and $L_{R,-m}$ with $m, n > 0$. If states are labeled by their conformal dimensions

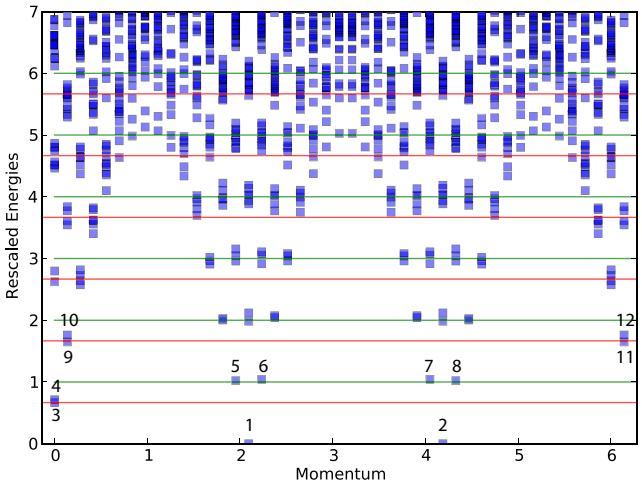



Figure 3.5: Dispersion plot for closed 1D chain with $L = 45$ sites at filling $f = 15$. The energies have been rescaled so that the first excitation gap is $\frac{2}{3}$. Fields corresponding to labels are listed in table 3.1. As a guide green (red) lines are shown at energies $n, n \in \mathbb{Z}$ ($\frac{2}{3} + n, n \in \mathbb{Z}$) corresponding to fields $V_{0,\pm\frac{1}{2}}$ ($V_{0,\pm\frac{3}{2}}$) and descendants.

$|\, h_L, h_R\rangle$ then the action of the generators of the Virasoro algebra are given by: $L_{L,-n}|\, h_L, h_R\rangle = |\, h_L + n, h_R\rangle$ and $L_{R,-m}|\, h_L, h_R\rangle = |\, h_L, h_R + m\rangle$. The effect on the energy of $L_{L,-n}$ and $L_{R,-m}$ is $+n$ and $+m$ respectively. The change in momentum in terms of the conformal dimensions is given by $h_L - h_R$ in units of $\frac{2\pi}{L}$ so the change in momentum under the action of $L_{L,-n}$ and $L_{R,-m}$ is $\frac{2n\pi}{L}$ and $\frac{-2m\pi}{L}$ respectively. In the plot (figure 3.5) highest weight states and some of their descendants have been labelled and the corresponding fields are listed in table 3.1. There is good agreement between the numerically calculated values and the CFT predictions for the low lying values which as expected gets worse for higher energy states.

### 3.2.2 Entanglement

The entanglement entropy (EE) is a measure of how much entanglement exists between two parts of a system. The EE between two parts of a system

| Highest weight | | Descendants | |
|---|---|---|---|
| Label | Fields | Label | Fields |
| 1 | $V_{0,-\frac{1}{2}}$ | 5 | $L_{R,-1}V_{0,-\frac{1}{2}}$ |
| | | 6 | $L_{L,-1}V_{0,-\frac{1}{2}}$ |
| 2 | $V_{0,\frac{1}{2}}$ | 7 | $L_{R,-1}V_{0,\frac{1}{2}}$ |
| | | 8 | $L_{L,-1}V_{0,\frac{1}{2}}$ |
| 3 | $V_{0,-\frac{3}{2}}$ | 9 | $L_{R,-1}V_{0,-\frac{3}{2}}$ |
| | | 11 | $L_{L,-1}V_{0,-\frac{3}{2}}$ |
| 4 | $V_{0,\frac{3}{2}}$ | 10 | $L_{R,-1}V_{0,\frac{3}{2}}$ |
| | | 12 | $L_{L,-1}V_{0,\frac{3}{2}}$ |

Table 3.1: Fields corresponding to the states labeled in figure 3.5.

labelled subsystems $A$ and $B$ in the state $|\,\psi\rangle$ is defined as the von Neumann entropy of the reduced density matrix of subsystem $A$ defined as $\rho_A = Tr_B(\rho)$ or equivalently of subsystem $B$ defined as $\rho_B = Tr_A(\rho)$.

$$S_{VN} = -Tr(\rho_A \log(\rho_A)) = -Tr(\rho_B \log(\rho_B))$$

The EE is zero when there is no entanglement and has a maximum value of $log(D)$, where $D = min(dim\mathcal{H}_A, dim\mathcal{H}_B)$ when the two subsystems are maximally entangled.

Another equivalent method which can be used to calculate the entanglement entropy [43] involves getting the Schmidt decomposition of the state $|\,\psi\rangle$ which gives:

$$|\,\psi\rangle = \sum_i e^{-\frac{1}{2}\xi_i} |\,\psi_A^i\rangle \otimes |\,\psi_B^i\rangle$$

where $|\,\psi_A^i\rangle \in \mathcal{H}_A$ and $|\,\psi_B^i\rangle \in \mathcal{H}_B$ are orthonormal states spanning these spaces. $\mathcal{H}_A$ and $\mathcal{H}_B$ are the Hilbert spaces of the states of subsystems A and B respectively with the full Hilbert space given by $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$. The Schmidt decomposition of the state $|\,\psi\rangle$ is equivalent to getting the singular value decomposition of the weight matrix $W_{\alpha,\beta}$. The state $|\,\psi\rangle$ expressed in

terms of the weight matrix is:

$$|\,\psi\rangle = \sum_{\alpha,\beta} W_{\alpha,\beta} |\,\psi_A^\alpha\rangle \otimes |\,\psi_B^\beta\rangle$$

where $\alpha$ labels the configurations on subsystem $A$ and $\beta$ labels the configurations on subsystem $B$. The singular values of this matrix are $e^{-\frac{1}{2}\xi_i}$. The EE in terms of $\xi_i$ is given by:

$$S_{VN} = \sum_i \xi_i e^{-\xi_i}$$

The area law states that the EE is proportional to the length of the boundary. For 1D systems the boundary consists of the end points only and therefore is constant. For 1D gapped systems the area law holds and the EE saturates to a constant when the length of the subset becomes greater than the correlation length. For critical 1D systems however the EE is related to the logarithm of the length of the subset and continues to grow as the subset size increases. For a large but finite critical system of length $L$ the EE represented by $S_{VN}$ of a subsystem $A$ of length $l_A$ is given by the expression:

$$S_{VN} = \frac{c}{3} \log(\frac{L}{\pi} \sin(\frac{l_A \pi}{L})) + d \tag{3.8}$$

for closed chains and:

$$S_{VN} = \frac{c}{6} \log(\frac{L}{\pi} \sin(\frac{l_A \pi}{L})) + d \tag{3.9}$$

for open chains [44, 45, 46]. Here $c$ is the central charge of the CFT describing the critical point and $d$ is a non-universal constant (different for open and closed chains).

We now look at the EE for the 1D SUSY chain. For the closed case we look at the EE for chains with lengths that are not multiples of three. In these cases there is a single ground state which makes calculating the EE more straightforward. Figure 3.6 shows the EE at different subsystem sizes and different length chains in the ground state. Fits to the data with

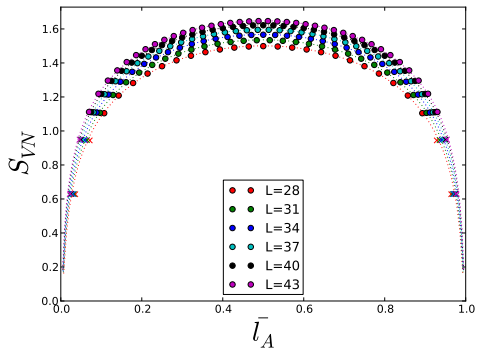



Figure 3.6: Entanglement entropy in the ground state of the closed SUSY chain. $\bar{l_A} = \frac{l_A}{L}$ where $l_A$ is the length of subsystem $A$. The function used for the fits is shown in equation 3.8. The data points with the 'x' symbols were not considered for the fits.

equation 3.8 are also shown with the dashed lines and the parameters of the fits are shown in table 3.2. To obtain good fits to the data, the EE for subsystems with $l_A < 3$ and $l_A > L - 3$ were excluded. It is known that in the thermodynamic limit the closed SUSY chain is described by a CFT with $c = 1$. The values obtained for $c$ from the fits are in good agreement with the expected value and get better with increasing system size.

Calculations of the EE were also performed for the open chain. The results of these calculations are shown in figure 3.7. One thing that is immediately apparent here is that the values of the EE here are substantially lower than for the closed case. This is explained by the fact that for the open system a subsystem here has only one boundary instead of two for closed systems. The other significant difference in this case is that there are oscillations appearing in the EE resulting from the way the system is partitioned. This can be clearly seen in figure 3.7a. Fits were performed for various system sizes using subsystem lengths that are multiples of three. These fits are shown along with the data in figure 3.7b. The parameters of the fits are also shown in table 3.2. The values for $c$ obtained from the fits are again close to the expected $c = 1$ and get closer with increasing system size.

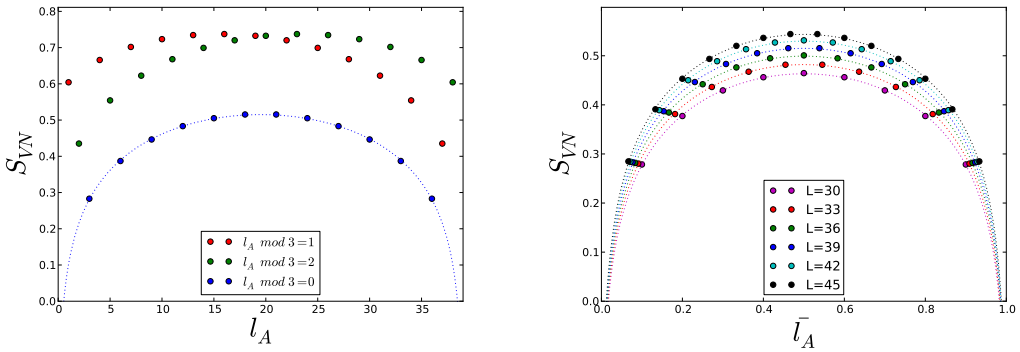
| Closed chain | | | Open chain | | |
|---|---|---|---|---|---|
| Length | c | d | Length | c | d |
| 28 | 1.0582 | 0.7294 | 30 | 0.94918 | 0.1064 |
| 31 | 1.0521 | 0.7333 | 33 | 0.96175 | 0.1051 |
| 34 | 1.0473 | 0.7366 | 36 | 0.97260 | 0.1040 |
| 37 | 1.0433 | 0.7394 | 39 | 0.98207 | 0.1029 |
| 40 | 1.0400 | 0.7418 | 42 | 0.99040 | 0.1019 |
| 43 | 1.0371 | 0.7438 | 45 | 0.99780 | 0.1009 |

Table 3.2: Values of the fitting parameters obtained when fitting equations 3.8 and 3.9 to the data shown in figures 3.6 and 3.7 respectively. In each case the error of the least squares fit is on the order of $10^{-6}$.

## 3.3 Staggered SUSY chain

The staggered chain [35, 47] is a variant on the regular chain where every third site is staggered. For this model we show the results of gap scaling calculations which serve to establish the nature of the phase for both open and closed chains for each value of staggering parameter. Exact expressions for the ground state(s) in each of the staggering limits are then presented. Particular attention is paid to the infinite staggering limit of the open chain for which we introduce the 'kink' picture. We then show expressions for the entanglement entropy and entanglement spectrum in each case. These are derived from the expressions of the ground state(s) and are verified with numerical calculations where appropriate. The introduction of staggering does not affect the ground state degeneracy or the filling at which they are found. The supercharge operators are defined as $Q(a)^{(\dagger)} = Q_1^{(\dagger)} + aQ_2^{(\dagger)}$ where $a$ is the staggering parameter. $Q_1^{(\dagger)}$ acts on the sites represented by circles and $Q_2^{(\dagger)}$ acts on the sites represented by squares in figure 3.8. The supercharge operators for the staggered chain of length $L = 3k$ can be written as:

$$Q(a)^{(\dagger)} = \sum_{i=0}^{k-1} (P_{\langle 3i \rangle} c_{3i}^{(\dagger)} + aP_{\langle 3i+1 \rangle} c_{3i+1}^{(\dagger)} + P_{\langle 3i+2 \rangle} c_{3i+2}^{(\dagger)})$$




(a) (b)

Figure 3.7: Entanglement entropy in the ground state of the open SUSY chain for (a) chain length $L = 39$ with all subsystem sizes and (b) various lengths with $33 \leq L \leq 45$ and subsystem sizes $l_A \mod 3 = 0$ with $\bar{l_A} = \frac{l_A}{L}$. The function used for the fits is shown in equation 3.9.

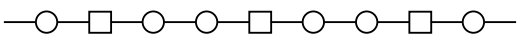


Figure 3.8: Staggered chain.

which results in the Hamiltonian:

$$H = \sum_{\langle i,j \rangle} \alpha_i \alpha_j (c_i^\dagger c_j + c_j^\dagger c_i) + \sum_i \alpha_i^2 P_{<i>}$$

where $\alpha_i = a$ if $i \ mod\ 3 = 1$ and $\alpha_i = 1$ otherwise (sites are labelled from 0 to $3k - 1$). Setting $a = 1$ corresponds to the regular chain which is critical and separates two distinct phases.

### 3.3.1 Low staggering limit

In the low staggering limit the staggering parameter $a \to 0$. In this limit the structure of the ground state is known exactly [48]. For the open chain the single ground state consists of the configuration where all the staggered sites are occupied. This configuration is illustrated for a chain of length $L = 9$ in figure 3.9 and in terms of the fermionic creation operators acting on the

vacuum:

$$|\,\psi_{GS_1}\rangle = \prod_{i=0}^{k-1} c^\dagger_{3i+1}|\,\emptyset\rangle \tag{3.10}$$

Here sites are labelled from 0 to $3k-1$. It is easy to see from the Hamiltonian why this configuration has zero energy. Any hopping terms to or from the staggered sites have coefficient zero in this limit and the potential term on this site contributes zero energy.

Figure 3.9: Configuration on staggered chain with $L = 9$ with all staggered sites occupied.

For the closed chain of length $L = 3k$ the two fold degenerate ground state space is spanned by the ground state of the open chain of length $L = 3k$ and a state made up of a superposition of the $2^k$ possible configurations with $k$ fermions such that no staggered sites are occupied. These configurations for a chain of length $L = 9$ are shown in figure 3.10. This state can be expressed as:

$$|\,\psi_{GS_2}\rangle = \frac{1}{2^{\frac{k}{2}}} \prod_{i=0}^{k-1} (c^\dagger_{3i} - c^\dagger_{3i+2})|\,\emptyset\rangle \tag{3.11}$$

Here sites are labelled from 0 to $3k-1$ and the label $3k$ is site 0. It is not as straightforward to see that this state has zero energy in this limit. While each configuration has finite energy the way the configurations are summed results in a net energy of zero.

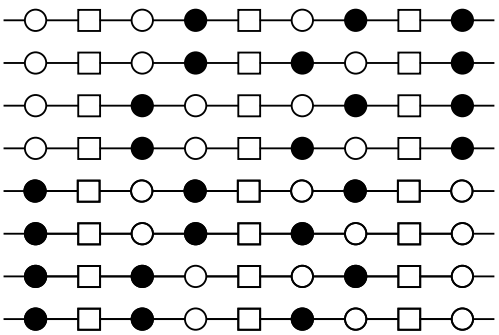
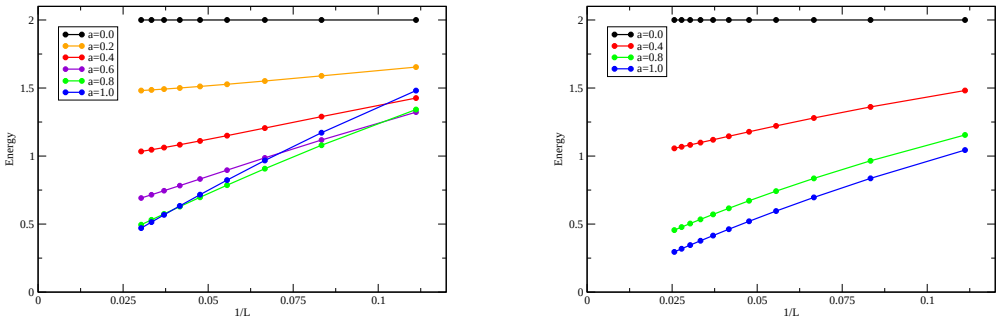


Figure 3.10: All possible configurations on closed staggered chain of length $L = 9$ with filling $F = 3$ such that no staggered sites are occupied.




(a) Closed chain.

(b) Open chain.

Figure 3.11: FSS of the excitation gap for the (a) closed and (b) open staggered SUSY chain as $a \to 0$.

The gap in this limit is 2 with the energy of all excitations being multiples of 2. FSS gap calculations for different values of the staggering parameter $0 \leq a \leq 1$ show that as $a \to 1$ the gap goes to a constant value of 2 for both closed and open chains. The results of these calculations are shown in figure 3.11.

### 3.3.2 Large staggering limit

In the large staggering limit as $a \to \infty$ it becomes extremely energetically expensive for the staggered sites to be occupied or for a fermion to hop to or from one of these sites. In this case the ground state space of the closed chain is spanned by the two possible configurations for which each staggered site has an occupied neighbour. The presence of an occupied site next to each staggered site makes it impossible for any fermions to be created on the staggered sites. For a chain of length $L = 3k, k = 3$ the possible configurations are shown in figure 3.12. For the open chain the ends are no longer connected and as a result both ends may be occupied simultaneously. In this case for a chain of length $L = 3k$ there are an additional $k - 1$ configurations that contribute to the ground state. The additional configurations for a chain of

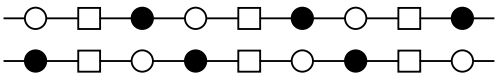


Figure 3.12: The two possible configurations on closed chain where each staggered site has an occupied neighbour.

length $L = 3k, k = 3$ are shown in figure 3.13. The single ground state is an

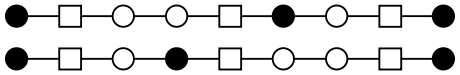
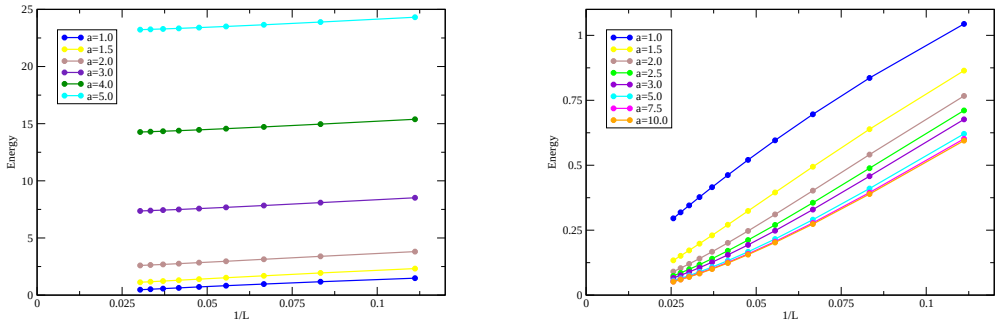


Figure 3.13: Additional configurations possible in large staggering limit of the open chain.

equal superposition over these configurations and has the form:

$$|\,\psi_{GS}\rangle = \frac{1}{\sqrt{k+1}} \sum_{i=1}^{k+1} [\prod_{j=0}^{k-i} c_{3j}^{\dagger}][\prod_{j=k-i+1}^{k-1} c_{3j+2}^{\dagger}]|\,\emptyset\rangle \tag{3.12}$$

where sites are labelled from 0 to $3k-1$. FSS calculations of the energy gap




(a) Closed chain.

(b) Open chain.

Figure 3.14: FSS of the excitation gap for the (a) closed and (b) open staggered SUSY chain as $a \to \infty$.

were performed for both open and closed chains in the large staggering limit. The results of these calculations are plotted in figure 3.14. The scaling of

the gap is drastically different in each case in this limit. For the closed chain as the staggering parameter increases a gap opens up and continues to grow as the staggering parameter is increased (figure 3.14a). This excitation gap can be attributed to configurations where staggered sites are either occupied or not blocked by an occupied neighbouring site. For the open chain (figure 3.14b) the model remains gapless as the staggering parameter is increased. However the gap no longer scales to zero linearly with the inverse system size. In this regime in the thermodynamic limit there is a continuum of states above the ground state. This regime is explored further in the next section.

#### 3.3.2.1 Excited states

In the previous section we showed that the ground state of the open staggered chain of length $L = 3k$ in the large staggering limit $a \to \infty$ is made up of a sum over $k + 1$ configurations. Each of these configurations does not have any fermions on staggered sites, but at the same time each staggered site has a neighbouring site that is occupied. We label these configurations $|\,\psi_1\rangle$ to $|\,\psi_{k+1}\rangle$ as in figure 3.15. One can view each of these configurations as having

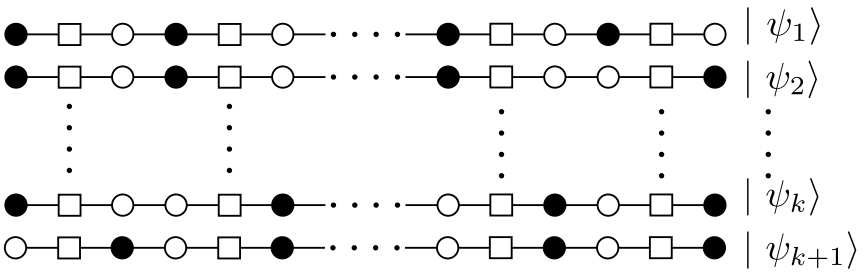



Figure 3.15: All possible configurations for the open staggered chain where no staggered site is occupied and each staggered site has an occupied neighbour.

a 'kink'. This is simply a break in the regular pattern. In $|\,\psi_1\rangle$ the kink is on the far right and it moves to the left until it ends up on the far left in configuration $|\,\psi_{k+1}\rangle$.

Since in these configurations the staggered sites are blocked, it is natural to assume that these configurations are also the dominant configurations

of the first excited states. We now show the derivation of an analytical expression for the energies of these low lying states [49] and compare this to results form numerical calculations. The Hamiltonian can be written in terms of $Q_1^{(\dagger)}$ and $Q_2^{(\dagger)}$ where $Q_1^{(\dagger)}$ acts on the regular sites represented by circles in figure 3.15 and $Q_2^{(\dagger)}$ acts on the staggered sites represented by squares. The Hamiltonian for the staggered chain can then be expanded as:

$$H = a^2(Q_2Q_2^\dagger + Q_2^\dagger Q_2) + a(Q_1Q_2^\dagger + Q_2Q_1^\dagger + Q_1^\dagger Q_2 + Q_2^\dagger Q_1) + Q_1Q_1^\dagger + Q_1^\dagger Q_1$$

First we look at the effect of the term $Q_1Q_1^\dagger$ on $|\,\psi_i\rangle$. Applying $Q_1^\dagger$ on the states $|\,\psi_i\rangle$ results in a sum of states, each with an additional fermion inserted at the appropriate position as allowed by the nearest neighbour exclusion condition. For $|\,\psi_1\rangle$ and $|\,\psi_{k+1}\rangle$ there is only one appropriate position, which is the very last site and the very first site respectively. For each additional configuration applying $Q_1^\dagger$ results in an equal superposition of the configurations that have an additional fermion inserted at each of the two available positions.

Applying $Q_1$ to $Q_1^\dagger|\,\psi_i\rangle$ results in a superposition over $k+1$ states for $|\,\psi_1\rangle$ and $|\,\psi_{k+1}\rangle$ and $2(k+1)$ states for the other states. For $|\,\psi_1\rangle$ and $|\,\psi_{k+1}\rangle$ one of the configurations in the superposition will be the original configuration, $|\,\psi_1\rangle$ or $|\,\psi_{k+1}\rangle$ respectively. For the other states $|\,\psi_i\rangle$ where $1 < i < (k+1)$ two of the configurations will be instances of the original configuration $|\,\psi_i\rangle$. As well as instances of the original configuration $|\,\psi_i\rangle$, there will be one instance of the previous configuration $|\,\psi_{i-1}\rangle$, and one instance of the next configuration in the list $|\,\psi_{i-1}\rangle$. However these will have a negative phase resulting from the fermionic anti-commutation relations. In the case of the first and last configurations $|\,\psi_1\rangle$ and $|\,\psi_{k+1}\rangle$ there will only be a single instance of the next and previous configurations respectively. Again these instances have a negative phase.

In the superposition of configurations resulting from the application of $Q_1Q_1^\dagger$ on $|\,\psi_i\rangle$ as well as the instances of configurations from the subset of states $|\,\psi_i\rangle$, $1 \leq i \leq (k+1)$ there are other configurations for which every staggered site is not blocked by an occupied neighbour. We ignore these

configurations for now. In summary acting on $|\,\psi_i\rangle$ with $Q_1Q_1^\dagger$ and ignoring configurations where the staggered sites are unblocked results in:

$$
\begin{aligned}
&Q_1Q_1^\dagger|\,\psi_1\rangle = |\,\psi_1\rangle - |\,\psi_2\rangle \\
&Q_1Q_1^\dagger|\,\psi_2\rangle = -|\,\psi_1\rangle + 2|\,\psi_2\rangle - |\,\psi_3\rangle \\
&... \\
&Q_1Q_1^\dagger|\,\psi_k\rangle = -|\,\psi_{k-1}\rangle + 2|\,\psi_k\rangle - |\,\psi_{k+1}\rangle \\
&Q_1Q_1^\dagger|\,\psi_{k+1}\rangle = -|\,\psi_k\rangle + |\,\psi_{k+1}\rangle
\end{aligned}
$$

If the states $|\,\psi_i\rangle$ are chosen as a basis $Q_1Q_1^\dagger$ can be written as the matrix:

$$
\begin{pmatrix}
1 & -1 & & & & & & \\
-1 & 2 & -1 & & & & 0 & \\
 & -1 & 2 & -1 & & & & \\
 & & . & . & . & & & \\
 & & & . & . & . & & \\
 & & & & -1 & 2 & -1 & \\
 & & 0 & & & -1 & 2 & -1 \\
 & & & & & & -1 & 1
\end{pmatrix}
$$

This matrix has the form of a second derivative with eigenvalues given by the expression [49]:

$$\lambda_n = 4sin^2(\frac{n\pi}{2k+2}) \tag{3.13}$$

where $0 \leq n < k+1$. This result shows very good agreement with the numerically calculated energies, even for reasonably small ($L = 18$) systems. Figure 3.16 shows the energy of the low lying excitations for comparatively small chains alongside the analytical expression in equation 3.13.

The match between this expression and the numerically calculated values is surprisingly good considering this is just one of the terms of the Hamiltonian and configurations other than $|\,\psi_k\rangle$ are ignored. This is explained by noting that configurations lying outside this subset of configurations contribute to the low lying states and cancel with these terms. The weight with which these configurations contribute is on the order of $\frac{1}{a}$. This can be seen

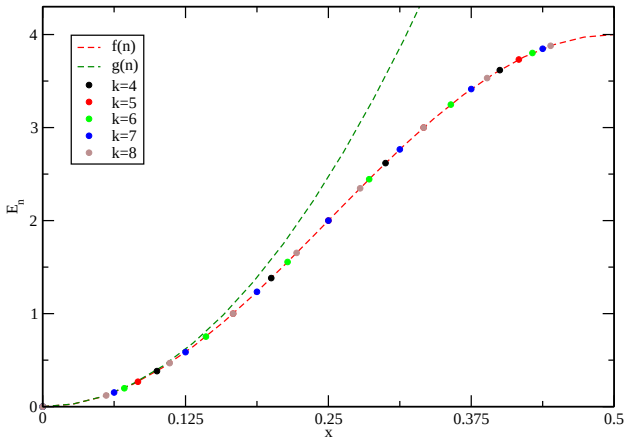



Figure 3.16: Plot of the energies of low lying excitations of open staggered SUSY chains with staggering $a = 100$ and lengths $3k$ with $4 \leq k \leq 8$. $x = \frac{n}{2k+2}$. The equations used for the fits are $f(n) = 4\sin^2(\frac{n\pi}{2k+2})$ and $g(n) = 4(\frac{n\pi}{2k+2})^2$.

by working through a very small example by hand. To verify this further, calculations of the one point functions in the ground state and first excited states at one of the staggered sites were calculated for different values of the staggering parameter. The results of these calculations are shown in figure 3.17. Here it was found that the one point function at the staggered site is inversely related to the square of the staggering parameter $a$. This indicates that configurations with staggered sites occupied do indeed appear with weight on the order of $\frac{1}{a}$ in the ground state and low lying excited states.

Another point worth noting is that in the limit of large system size the energies of the lowest states match those for the particle in a box model.

$$\lambda_n = 4sin^2(\frac{n\pi}{2k+2}) \approx 4(\frac{n\pi}{2k+2})^2$$

A fit for this expression is also shown in figure 3.16.

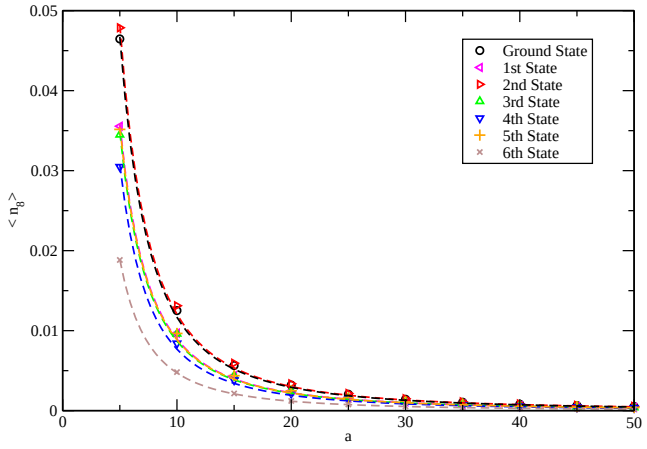



Figure 3.17: The one point function values at site 8 (one of the staggered sites) in the ground state and first excited states of the 18 site open staggered SUSY chain plotted against the staggering parameter $a$. The dashed lines are fits to the data with function $f(a) = \frac{const.}{a^2}$.

### 3.3.3 Entanglement

We now look at the entanglement for the staggered SUSY chain. In both high and low staggering limits there are exact expressions for the ground state which can be used to determine exactly the entanglement entropy (EE) and entanglement spectrum (ES) in these limits.

In the low staggering limit of the open SUSY chain there is no entanglement. Here there is a single ground state described by the expression in equation 3.10. For any choice of partitioning it is possible to write this state as a product of states from each subset. Thus the EE is zero and there is no ES.

For the closed SUSY chain in the low staggering limit there are two ground states. One state is identical to the ground state of the open chain and has no entanglement. The other state (equation 3.11) has entanglement that is constant for each type of partitioning. The first type of partitioning we look at is where the system is partitioned such that each partition is made up of

blocks of three sites, where the central site of each block is a staggered site. In this case the EE between the subsets is a constant $S_{VN} = log(4)$ and the ES consists of a four fold degenerate point: $\xi_i = log(4), 0 \leq i < 4$. As discussed in section 3.2.2 this is the expected behaviour for 1D gapped systems. Why this is so can be seen by noting that a subset of this ground state that consists of $k$ blocks consists of configurations with fillings $k$ and $k+1$. For the configurations with filling $k$ there are three possible configurations on the boundary sites: both vacant, left vacant or right vacant and for configurations with filling $k+1$ both boundary sites are always occupied. The ground state can then be written as the equal superposition of four products states. From these weights the EE and ES are easily calculated. For the other possible choices of partitions the EE is also constant. For the case where both boundaries of a partition are at one of the staggered sites there is no entanglement. For the case where a single boundary is at one of the staggered sites the EE is given by $S_{VN} = log(2)$ and the ES consists of a two fold degenerate point with $\xi_i = log(2), 0 \leq i < 2$.

In the large staggering limit on the closed SUSY chain each ground state can be expressed as a simple product state over any subsets chosen so there is no entanglement. For the SUSY open chain the ground state is shown in equation 3.12 and the situation is more complicated. When an open chain of length $6k$ is divided into two equal subsets it was found that it is possible to analytically derive expressions for the reduced density matrix, the EE and the ES [49]. Here we show how the reduced matrix can be derived from the weight matrix and extend this technique to work with other choices of partitioning.

When an open chain with $3k$ sites is divided into subsets $A$ and $B$ containing $l_A = 3l$ and $l_B = 3(k-l)$ sites respectively the weight matrix $W_{i,j}$ has $m$ rows and $n$ columns where $m = l+1, n = k-l+1$. The elements of this matrix are given by the expression:

$$W_{i,j} = \frac{1}{\sqrt{k+1}}(\delta_{0,i} + \delta_{n-1,j} - \delta_{0,i}\delta_{n-1,j})$$

Multiplying this matrix by its transpose results in an $m \times m$ matrix which

is the reduced density matrix of the subset $A$.

$$\rho_A = WW^T = Tr_B(\rho_{AB})$$

The matrix elements of this matrix are given by:

$$[\rho_A]_{i,j} = \begin{cases} \frac{1}{k+1} & \text{for } i > 0 \text{ or } j > 0 \\ \frac{n}{k+1} & \text{for } i = j = 0 \end{cases}$$

This is a rank two matrix with eigenvalues given by:

$$\lambda_{\pm} = \frac{1}{k+1}(\frac{m+n-1}{2} \pm \frac{\sqrt{(m-n-1)^2+4(m-1)}}{2})$$

and unnormalised eigenvectors of the form:

$$|\, x_{\pm}\rangle = |\, \frac{(n-m+1}{2} \pm \frac{\sqrt{(m-n-1)^2+4(m-1)}}{2}, 1, 1, .....1\rangle \qquad (3.14)$$

When an open SUSY chain with a multiple of six sites is cut in half, $m = n$ and the expression for the eigenvalues becomes:

$$\lambda_{\pm} = \frac{1}{k+1}(\frac{2n-1}{2} \pm \frac{\sqrt{4n-3}}{2})$$

When the chain is divided into subsets with lengths $l_A = 3l - 1$ and $l_B = 3(k-l)+1$ the weight matrix has dimension $m \times n$ with $m = l+1, n = k-l+2$. The elements of this matrix are given by the expression:

$$W_{i,j} = \frac{1}{\sqrt{k+1}}(\delta_{0,i} + \delta_{n-1,j} - 2\delta_{0,i}\delta_{n-1,j})$$

The related reduced density matrix resulting from the multiplication of this weight matrix by its transpose again has dimension $m \times m$ but in this case has elements given by:

$$[\rho_A]_{i,j} = \begin{cases} \frac{1}{k+1} & \text{for } i > 0 \text{ and } j > 0 \\ \frac{n-1}{k+1} & \text{for } i = j = 0 \end{cases}$$

This is a rank two matrix with eigenvalues given by:

$$\lambda_0 = \frac{m-1}{k+1} = \frac{l}{k+1}, \qquad \lambda_1 = \frac{n-1}{k+1} = \frac{k-l+1}{k+1}$$

and unnormalised eigenvectors of the form:

$$|\, x_0\rangle = |\, 1, 0, 0, .....0\rangle, \qquad |\, x_1\rangle = |\, 0, 1, 1, .....1\rangle \tag{3.15}$$

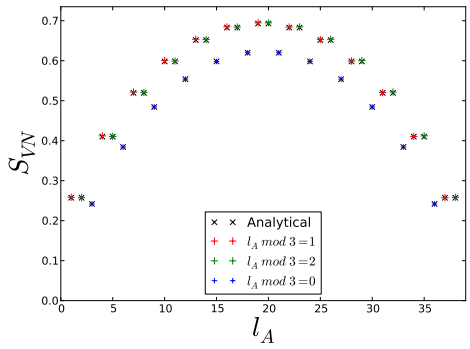
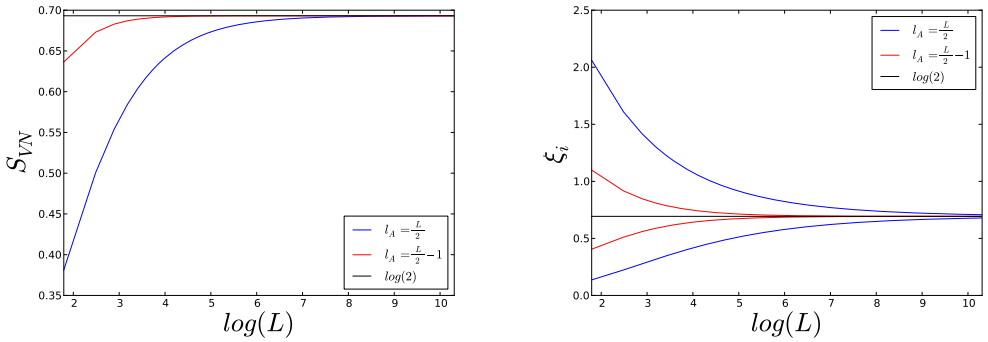



Figure 3.18: Numerically calculated EE data for the ground state of the open $L = 39$ staggered chain with staggering parameter $a = 50$ plotted alongside analytical results for the infinite staggering limit.

The EE in terms of the eigenvalues of the reduced density matrix is $S_{VN} = \sum_i -\lambda_i log(\lambda_i)$ and the values of the ES are $\xi_i = -log(\lambda_i)$. Using the expression for the EE the analytical results were compared to numerical data for an open chain with $L = 39$ and staggering parameter $a = 50$ (figure 3.18). There is almost exact agreement between the analytical and numerical data. The very slight deviation can be attributed to the fact that taking the staggering parameter to be $a = 50$ is not the infinite staggering limit.

In the limit of infinite system size where the system is partitioned into two equally sized subsets it can be seen from the expressions for the eigenvalues that both converge to a half. In this case the EE is $S_{VN} = log(2)$ and the ES consists of two points at $log(2)$. Figure 3.19 shows the convergence of the EE and the ES with chain length $L$ for the cases where the subset length




(a) (b)

Figure 3.19: Analytical (a) EE and (b) ES at different subset lengths plotted against the logarithm of the chain length with $L = 3k, 2 \leq k \leq 10000$

$l_A = \frac{L}{2}$ and $l_A = \frac{L}{2} - 1$.

The two points of the ES in each case can be interpreted in terms of the 'kink' picture, discussed in section 3.3.2.1. In the case where the subset length is not a multiple of three, it is clear from the eigenvectors (equation 3.15) that the two eigenvalues of the reduced density matrix correspond to the cases where the 'kink' is in subset A only or in subset B only. In the case where the chain is split into subsets with lengths that are multiples of three it is possible that as well as the kink being in either subset it can also be split by the partitioning such that half is in each subset. We define the vector $|\, x_A\rangle$ which corresponds to the kink being completely in subset $A$ and the vector $|\, x_{\bar{A}}\rangle$ which corresponds to the kink being either not in $A$ or split by the partitioning. In terms of these vectors the eigenvectors of $\rho_A$ (equation 3.14) can be expressed as: $|\, x_\pm\rangle = \frac{1}{\sqrt{2}}(|\, x_A\rangle \pm |\, x_{\bar{A}}\rangle)$. In the infinite limit where the eigenvalues are degenerate, the superpositions $\frac{1}{\sqrt{2}}(|\, x_+\rangle \pm |\, x_-\rangle)$ are also eigenvectors. Then $\frac{1}{\sqrt{2}}(|\, x_+\rangle + |\, x_-\rangle) = |\, x_A\rangle$ which corresponds to the kink being completely in $A$ and $\frac{1}{\sqrt{2}}(|\, x_+\rangle - |\, x_-\rangle) = |\, x_{\bar{A}}\rangle$ which corresponds to the situation where either the kink is not in $A$ or is split by the partitioning.

## 3.4 Square octagon lattice

In this section we investigate the square octagon (SO) SUSY model[11]. The SO lattice consists of both squares (plaquettes) and octagons. Figure 3.20 shows an illustration of a $3 \times 3$ plaquette SO lattice.

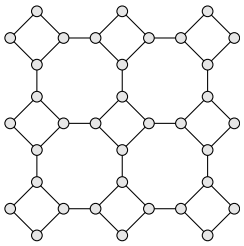


Figure 3.20: 3 x 3 plaquette SO lattice with 36 sites.

It is known that the degeneracy of the ground state of the SO SUSY model on a closed SO lattice with $M \times N$ plaquettes is $2^M + 2^N - 1$ and that on an open SO lattice there is a single ground state [35]. The ground states are also always found at a quarter filling. This makes the SO SUSY models stand out from the majority of other 2D SUSY lattice systems. The majority of 2D SUSY models like the triangular and hexagonal SUSY models have ground state degeneracies that grow substantially faster with the lattice dimensions and exhibit 'super-frustrated' phases [35, 38].

From the cohomology picture of the ground states of the SO SUSY model it is believed that defects on the plaquettes of this model could exhibit non-trivial braiding properties indicating the existence of topological order and a so-called 'super-topological' phase [50]. Questions of whether topological order is present in these models, and if so, what are its characteristics are not trivial to answer. In this thesis we present results of numerical calculations which can be viewed as initial steps towards resolving these questions.

In section 3.4.1 the SO chain is discussed and the results of FSS calculations of the gap for the closed chain are shown. Section 3.4.2 shows results of one point functions (defined in section 3.4.2) in the ground state. In section 3.4.3 defects and their effect on the ground state degeneracy are discussed. Section 3.4.3.1 shows how to gradually introduce defects which can be used

to cause a slight splitting of the ground state degeneracy which is useful for distinguishing between ground states. Results of calculations of the one point functions of systems with defects are also shown in this section. In section 3.4.4 the Projected Product State (PPS) wavefunction [49] is proposed as a trial wavefunction and compared to the numerically calculated ground state to access how well it captures the ground state properties.

### 3.4.1 Square octagon chain

The SO chain is the 1D limit of the 2D SO model and is shown in figure 3.21. The low lying energy spectrum of the $p$ plaquette SO SUSY model with $4p$ sites at $\frac{1}{4}$ filling is identical to that of the staggered SUSY chain (section 3.3) with $3p$ sites at $\frac{1}{3}$ filling with staggering parameter $a = \sqrt{2}$. Why this is true can be seen by noting that the low lying states of the SO chain do not include any configurations in which both the upper and lower sites of a plaquette are simultaneously occupied. Results of FSS calculations of

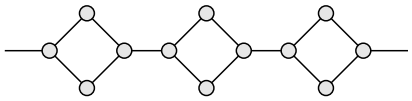
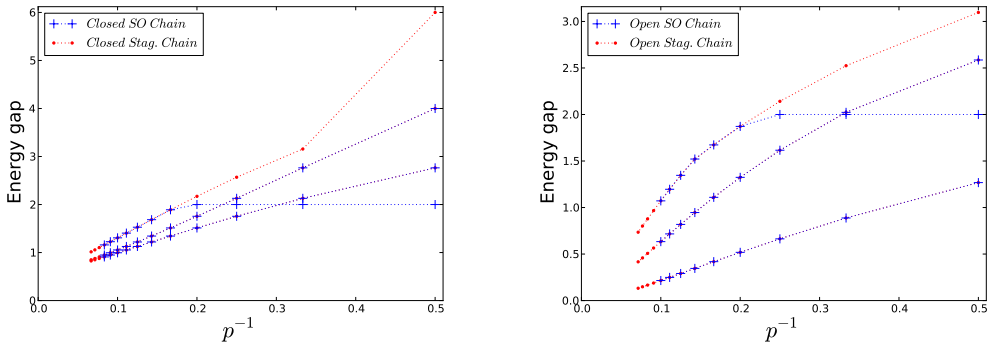


Figure 3.21: Square octagon chain.

the first three excitation gaps for both the closed and open cases of the SO chain and staggered SUSY chain at $a = \sqrt{2}$ are shown in figure 3.22. From these calculations it is clear that the low lying spectrum of both models are identical in each case (for large enough systems).

It is also clear from these plots (figure 3.22) that the closed SO SUSY model is gapped in the thermodynamic limit and that the open SO SUSY model is gapless in this limit. While the open SO SUSY is gapless, this gap does not close linearly with the inverse system length indicating that this model is not conformally critical.




(a) (b)

Figure 3.22: Plots showing equivalence of low lying spectrum of SO chain and staggered chain at staggering $a = \sqrt{2}$ (for sufficiently large chains). FSS of the first three excitation gaps are shown for (a) closed SO SUSY chains with $4p$ sites, $2 \leq p \leq 12$, closed staggered SUSY chains at $a = \sqrt{2}$ with $3p$ sites, $2 \leq p \leq 15$, (b) open SO SUSY chains with $4p$ sites, $2 \leq p \leq 10$ and closed staggered SUSY chains at $a = \sqrt{2}$ with $3p$ sites, $2 \leq p \leq 14$ all at filling $p$ which is $\frac{1}{4}$ filling for the SO chain and $\frac{1}{3}$ filling for the staggered SO chain.

### 3.4.2 Ground state structure

Here we look at the one point functions at each site in the ground state of the open SO SUSY model to get a better idea of the structure of the ground state. For a ground state $|\psi_{GS}\rangle$ the one point function at site $i$ is defined as $\langle\psi_{GS}|n_i|\psi_{GS}\rangle$ where $n_i = c_i^\dagger c_i$ is the number operator. The sum of the one point functions at all sites in the ground state is always equal to the number of plaquettes.

Figure 3.23 shows a plot of the one point functions at each site in the ground state of the $3 \times 3$ plaquette open SO lattice. The numerical values are displayed in table 3.5 in appendix 3.D. In this plot the one point functions at each site are reasonably uniform. The values of the one point functions are however greater than average on the boundary sites. This is explained by the fact that these boundary sites only have two neighbouring sites compared to three for the rest of the sites. This makes it more probable that these sites are occupied, as less configurations with this site occupied are projected out

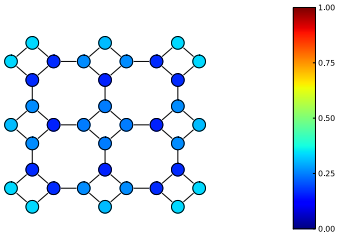



Figure 3.23: One point functions at each site in the ground state of the $3 \times 3$ plaquette open SO SUSY model.

due to the nearest neighbour exclusion. Also of interest is the fact that the values of the one point functions on the non boundary sites of the boundary plaquettes are less than average. This suggests that in the ground state it is unlikely that two sites of a single plaquette are simultaneously occupied. This is confirmed by the values of the two point functions for sites on the same plaquette which are shown in table 3.7 in appendix 3.D. This serves as motivation for using the Projected Product State (PPS) wavefunctions as trial ground states (3.4.4).

### 3.4.3 Defects

Defects in the form of diagonal links can be added to the plaquettes of the SO lattice as shown in figure 3.24 for a single plaquette. It is thought [51] that

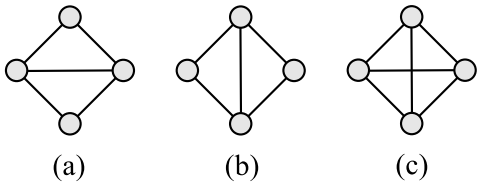



Figure 3.24: A single plaquette with (a) horizontal, (b) vertical and (c) both horizontal and vertical defects.

these defects could have non trivial braiding properties indicating the presence of topological order and a so-called 'super-topological' phase. Adding defects increases the degeneracy of the ground state. This can be clearly

observed by looking at the Witten index (section 3.1.2) which for SO SUSY models provides the exact number of ground states. For a single open plaquette with no defects the Witten index is:

$$W = 1(-1)^0 + 4(-1)^1 + 2(-1)^2 = -1$$

with a single horizontal or vertical defect it is:

$$W = 1(-1)^0 + 4(-1)^1 + 1(-1)^2 = -2$$

and with both horizontal and vertical defects it is:

$$W = 1(-1)^0 + 4(-1)^1 + 0(-1)^2 = -3$$

This shows that adding defects to a single open plaquette increases the ground state degeneracy from one to two in the case of a single defect and from one to three in the case of both horizontal and vertical defects. For larger open systems adding these defects to a single square plaquette increases the ground state degeneracy in the same manner. This has been confirmed numerically where results of exact diagonalisation calculations of open systems have exhibited the expected ground state degeneracies in each case.

#### 3.4.3.1 Gradually introducing defects

Adding defects to the plaquettes of the SO lattice increases the ground state degeneracy as discussed above. When there is a degenerate ground state, the states returned by exact diagonalisation are an arbitrary set of orthonormal states which span this ground state space. By adding the defects gradually we aim to split the degeneracy slightly to get a more natural choice of basis.

To gradually introduce defects weights are introduced on edges between sites, where a weight of zero indicates the absence of a link and a weight of one indicates the presence of a link. By increasing the weight on an edge continuously from zero to one a link is gradually added. The weight on the

edge connecting sites $i$ and $j$ is labelled $e_{ij}$. When an edge weight other than zero or one is used the supersymmetry is broken and the ground state energy is not identically zero. However as $e_{ij} \to 1$ the ground state energy goes to zero and the expected degeneracy is restored.

For $(1-e_{ij})$ small there is a slight splitting in the ground state degeneracy which gives a natural basis for the ground state space. We now look at how these weighting parameters are used in the Hamiltonian and show the effect on the energy spectrum. We then calculate the one point functions for systems where the weight on the defect link(s) approaches one and the ground state degeneracy is split slightly.

Using the weights $e_{ij}$ on each link the projectors $P_{\langle i \rangle}$ are modified to:

$$P_{\langle i \rangle} = \prod_{j \neq i} (1 - e_{ij} c_j^\dagger c_j)$$

Using these modified projectors we expand the Hamiltonian along the lines of what was done in section 3.1.

$$H = Q^\dagger Q + Q Q^\dagger = \sum_i P_{\langle i \rangle} [c_i^\dagger c_i + c_i c_i^\dagger] + \sum_i \sum_{\{k : e_{ik} = 0\}} P_{\langle i \rangle} [\cancel{c_i^\dagger c_k + c_i c_k^\dagger}] P_{\langle k \rangle}$$

$$+ \sum_i \sum_{\{k : e_{ik} > 0\}} P_{\langle i \rangle} [c_i^\dagger c_k + c_i c_k^\dagger] P_{\langle k \rangle}$$

$$= \sum_i P_{\langle i \rangle} + \sum_i \sum_{\{k : e_{ik} = 1\}} P_{\langle i \rangle} [c_i^\dagger c_k + \cancel{c_i c_k^\dagger}] P_{\langle k \rangle} + \sum_i \sum_{\{k : 0 < e_{ik} < 1\}} P_{\langle i \rangle} [c_i^\dagger c_k + c_i c_k^\dagger] P_{\langle k \rangle}$$

Now just looking at the terms $P_{\langle i\rangle}[c_i^\dagger c_k + c_i c_k^\dagger]P_{\langle k\rangle}$ where $0 < e_{ik} < 1$.

$$
\begin{aligned}
&P_{\langle i\rangle}[c_i^\dagger c_k + c_i c_k^\dagger]P_{\langle k\rangle} \\
&= [\prod_{j\neq k}(1 - e_{ij}c_j^\dagger c_j)](1 - e_{ik}c_k^\dagger c_k)(c_i^\dagger c_k + c_i c_k^\dagger)(1 - e_{ik}c_i^\dagger c_i)[\prod_{l\neq i}(1 - e_{kl}c_l^\dagger c_l)] \\
&= [\prod_{j\neq k}(1 - e_{ij}c_j^\dagger c_j)](c_i^\dagger c_k + c_i c_k^\dagger - \cancel{e_{ik}c_i^\dagger c_k c_i^\dagger c_i} - e_{ik}c_i c_k^\dagger c_i^\dagger c_i \\
&- \cancel{e_{ik}c_k^\dagger c_k c_i^\dagger c_k} - e_{ik}c_k^\dagger c_k c_i c_k^\dagger + \cancel{e_{ik}^2 c_k^\dagger c_k c_i^\dagger c_k c_i^\dagger c_i} + e_{ik}^2 c_k^\dagger c_k c_i c_k^\dagger c_i^\dagger c_i)[\prod_{l\neq i}(1 - e_{kl}c_l^\dagger c_l)] \\
&= [\prod_{j\neq k}(1 - e_{ij}c_j^\dagger c_j)](\cancel{c_i^\dagger c_k} - \cancel{c_k^\dagger c_i} + 2e_{ik}c_k^\dagger c_i - e_{ik}^2 c_k^\dagger c_i)[\prod_{l\neq i}(1 - e_{kl}c_l^\dagger c_l)]
\end{aligned}
$$

Thus the full Hamiltonian with the modified projectors is:

$$
\begin{aligned}
H = &\sum_i P_{\langle i\rangle} + \sum_i \sum_{\{k:e_{ik}=1\}} P_{\langle i\rangle}c_i^\dagger c_k P_{\langle k\rangle} \\
&+ \sum_i \sum_{\{k:0<e_{ik}<1\}} [\prod_{j\neq k}(1 - e_{ij}c_j^\dagger c_j)](e_{ik}c_k^\dagger c_i(2 - e_{ik}))[\prod_{l\neq i}(1 - e_{kl}c_l^\dagger c_l)]
\end{aligned}
$$

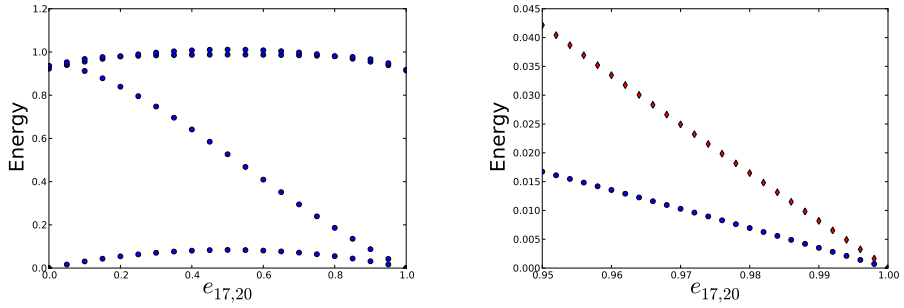
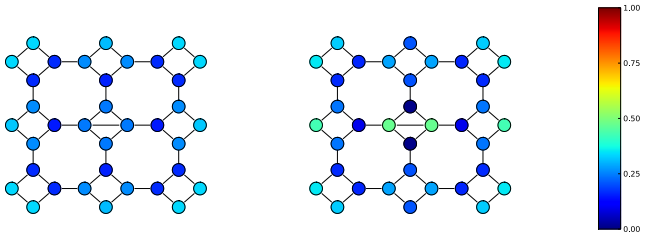



Figure 3.25: Plots of the energy of low lying states of $3\times3$ plaquette open SO lattice against link weight of the horizontal defect on the central plaquette.

The effect of adding a defect with varying weights between zero and one on the low lying energy spectrum is shown in the plots in figure 3.25. From these plots it is clear that as the link weight $e_{i,j} \to 1$ the excitation gap $E_1 - E_0 \to 0$. We now make use of this method to slightly split the degeneracy

and calculate one and two point functions.



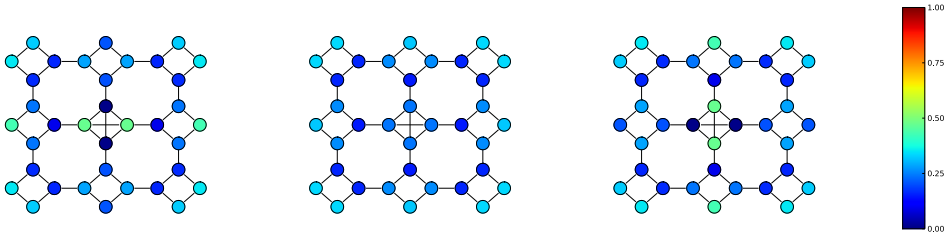

(a) (b)

Figure 3.26: One point functions in the (a) ground state and (b) the first excited state of the $3 \times 3$ plaquette open SO SUSY model with a horizontal defect added with link weight of 0.999 to the central plaquette.

The one point functions in the ground state and first excited state were calculated for the $3 \times 3$ plaquette open SO SUSY model with a horizontal defect added to the central plaquette with weight $e_{17,20} = 0.999$. The results are plotted in figure 3.26 and listed in appendix 3.D. The values of the one point functions in the state with slightly lower energy (fig 3.26a) are almost identical to those in the ground state of the system without any defects (fig 3.23). The overlap between these two states was calculated and found to be 0.9861 which confirms that these states are indeed very similar.

The one point functions for the state with slightly higher energy (fig 3.26b) are substantially different to those in the ground state. The overlap of this state with the ground state of the system with no defects was calculated to be $5 \times 10^{-20}$ which indicates that these states are essentially orthonormal. An interesting feature of the one point functions of this state are the substantially higher values on the sites at either end of the defect link and the sites at the opposite sides of the plaquettes which are horizontally adjacent to the central plaquette. Also of note is the substantially lower values of the one point functions at the sites which neighbour the sites at the end of the defect link. The case where a vertical defect is added to the central plaquette is the same except the lattice is rotated by ninety degrees.

When looking at the case where there is a horizontal and vertical defect on




(a) (b) (c)

Figure 3.27: One point functions in the (a) ground state, (b) first excited state and (c) second excited state of the $3 \times 3$ plaquette open SO lattice with a horizontal defect added with a weight of 0.99999 and a vertical defect added with a weight of 0.99998 to the central square. It is interesting to note that the ground state here is the same as the state with the horizontal defect added and not the state with no defects as might be expected.

the central plaquette we want to split the three fold ground state degeneracy. This is achieved by adding the horizontal and vertical defects with different weights, both of which are close to one. For the calculation of the one point functions for the $3 \times 3$ plaquette open SO SUSY model which are shown in figure 3.27 and listed in appendix 3.D the horizontal defect has been added with a weight of 0.99999 and the vertical defect with a weight of 0.99998.

Here the one point functions in the ground state (fig 3.27a) are almost identical to those of the first excited state for the system with the horizontal defect shown in figure 3.26b (right of figure). The overlap of 0.9998 provides strong confirmation for how similar these states are. The first excited state here (fig 3.27b) is essentially the same as the ground state of the system with no defects and the ground state of the system with a horizontal defect added to the central plaquette with weight of almost one (figs 3.26a and 3.23). The overlaps here are 0.9726 and 0.9863 respectively. The second excited state here is almost identical to the first excited state of the system where a vertical defect is added with weight of approximately one (fig 3.26b rotated by ninety degrees).

### 3.4.4 Projected product state wavefunction

It has been suggested [49] that a good approximation for the ground state of the SUSY model on the SO lattice is given by a projected product state (PPS) wavefunction. To judge how well this PPS wavefunction approximates the actual ground state wavefunction we look at the overlap of this wavefunction with the numerically calculated ground state. We also compare quantities of interest calculated from both wavefunctions. These quantities of interest here include correlation functions, the entanglement entropy and the entanglement spectrum.

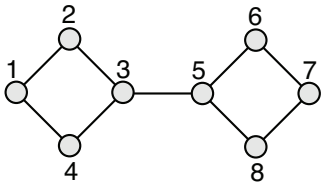



Figure 3.28: Illustration of the open SO chain with two plaquettes.

The PPS wavefunction is constructed by taking the product of the ground state wavefunction of each plaquette as if it were isolated from the rest of the lattice and then projecting onto the subspace of the Hilbert space consisting of configurations which satisfy the nearest neighbour exclusion condition. The resulting wavefunction is then normalised. For the open two plaquette SO chain shown in figure 3.28 the ground state of the plaquette on the left if isolated is given by $|\,\psi\rangle = |\,1\rangle - |\,2\rangle + |\,3\rangle - |\,4\rangle$ where $|\,n\rangle$ is the configuration on that plaquette in which all sites are empty except site $n$. The PPS wavefunction for this lattice is then given by:

$$\begin{aligned}
|\,\psi\rangle_{GS} &= \frac{\hat{P}}{\mathcal{N}}(|\,1\rangle - |\,2\rangle + |\,3\rangle - |\,4\rangle)(|\,5\rangle - |\,6\rangle + |\,7\rangle - |\,8\rangle) \\
&= \frac{1}{\sqrt{15}}(|\,1,5\rangle - |\,1,6\rangle + |\,1,7\rangle - |\,1,8\rangle - |\,2,5\rangle + |\,2,6\rangle - |\,2,7\rangle + |\,2,8\rangle \\
&\quad + \cancel{|\,3,5\rangle} - |\,3,6\rangle + |\,3,7\rangle - |\,3,8\rangle - |\,4,5\rangle + |\,4,6\rangle - |\,4,7\rangle + |\,4,8\rangle)
\end{aligned}$$

Here the crossed out configurations are those that are projected out because

they are disallowed by the nearest neighbour exclusion condition. In the next sections we test how good the PPS wavefunction is at approximating the ground state of the the SO SUSY model.

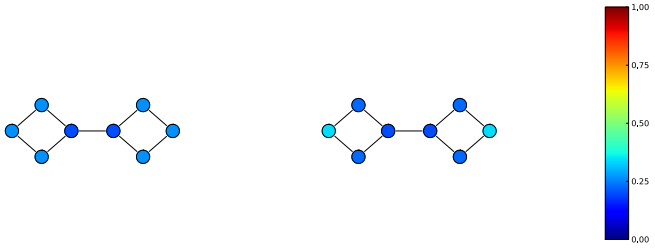
#### 3.4.4.1 Square octagon chain

For the open SO SUSY chain with two plaquettes as shown in figure 3.28 the overlap of the ground state with the PPS wavefunction is very high indicating that the PPS wavefunction is a good approximation to the ground state in this case.

$$|\langle\psi_{GS}|\psi_{prod}\rangle|^2 = 0.9634$$

Figure 3.29 shows the one point functions at each site for both the PPS wavefunction and the real ground state for this case. The values are shown in table 3.9 in appendix 3.D. There is good agreement between the one point functions of each wavefunction with the most significant difference occurring on the sites at each end of the chain. The PPS wavefunction here fails to recognise that it is more probable that a fermion is found here due to the fact that these sites have two neighbours compared to three for the rest of the sites. Two point functions across each plaquette were also calculated for the real ground state to see if any configurations with more than a single fermion per plaquette contribute to the ground state. By construction each of these are zero for the PPS wavefunction. The values of the two point functions horizontally across each plaquette (sites one and four and sites five and eight in figure 3.28) in the ground state are 0.00461 and in the vertical direction are zero. This tells us that configurations with two fermions on opposite horizontal sides of a plaquette contribute to the ground state in a very minor way and configurations with fermions on vertically opposite sites of a plaquette do not contribute at all.

While it is a nice illustrative example, the chain with two plaquettes is too small to infer how well the PPS wavefunction reproduces the features of the ground state of the SO SUSY chain in general. The overlap of the ground state with the PPS wavefunction has been calculated for open chains with up to ten plaquettes and the results plotted in figure 3.30. As the sys-




(a) (b)

Figure 3.29: Plots of the one point functions at each site of the PPS wavefunction (a) and the actual ground state (b) for the two plaquette SO chain.

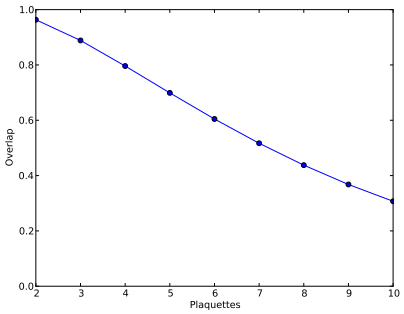
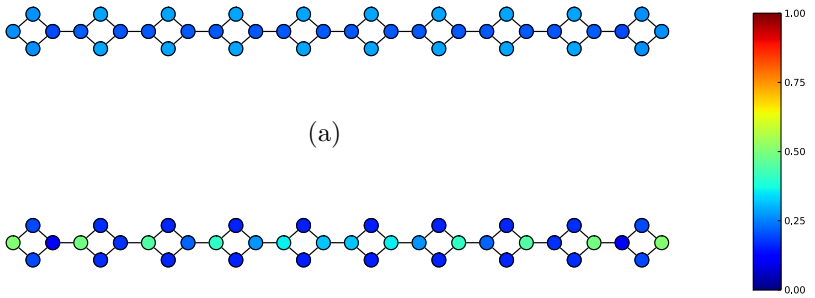



Figure 3.30: Plot of the overlap of the ground state and PPS wavefunction plotted against the number of plaquettes for the open SO chain.

tem size is increased the overlap decreases approximately linearly while the dimension of the Hilbert space grows exponentially fast. We infer that this result indicates that the PPS wavefunction captures some of the ground state structure well. The one point functions at all sites of the PPS wavefunction and the numerically calculated ground state of the ten plaquette open SO SUSY chain were calculated and are shown in figure 3.31. The values are shown in table 3.10 in appendix 3.D. In this case there are dramatic differences between the one point functions of each state. The two point functions across the plaquettes were also calculated for this case. Here it was found that again the value of the two point functions between the top and bottom sites of each plaquette is zero. In this case though the value of the two point




(a)

(b)

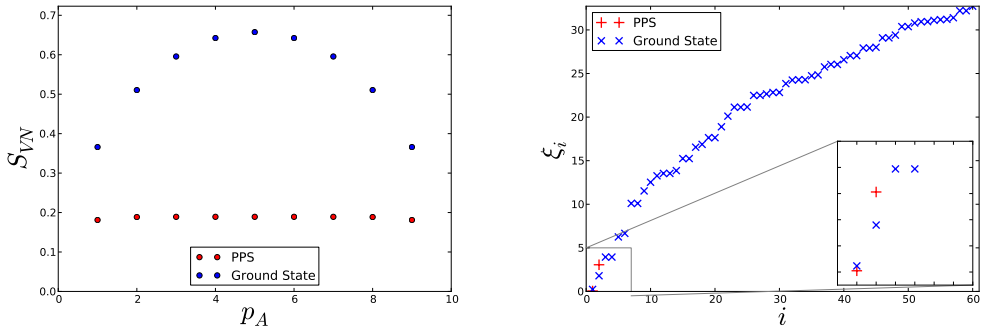
Figure 3.31: One point functions of the (a) PPS wavefunction and (b) numerically calculated ground state of the open SO SUSY chain with ten plaquettes.

functions between the sites to the left and right of each plaquette is greater at approximately 0.025. The specific values of these two point functions are listed in table 3.11 in appendix 3.D.

Calculations of the entanglement entropy (EE) and entanglement spectrum (ES) of the PPS wavefunction and numerically calculated ground state were performed which reveal that the PPS wavefunction fails to capture a large amount of the entanglement present in the ground state of the SO chain. Figure 3.32 shows plots of both the EE and ES. In figure 3.32a we see that the EE is significantly lower for the PPS wavefunction and does not change much with changes in subsystem size. The plot of the ES (figure 3.32b) shows that while there are significantly less points in the ES of the PPS wavefunction, the lowest and most significant point is close to that of the numerically calculated ground state. Product state wavefunctions by construction do not have any entanglement. For the PPS wavefunctions the entanglement is introduced by the projection to the Hilbert space of allowable configurations.

#### 3.4.4.2 Square octagon lattice

We now investigate the PPS wavefunction for the full two dimensional SO SUSY model by first looking at overlaps for various lattice sizes. We then




(a) (b)

Figure 3.32: Entanglement of PPS wavefunction and numerically calculated ground state of the ten plaquette open SO chain. (a) Entanglement entropy plotted against subsystem size where $p$ is the number of plaquettes in subsystem A. (b) Entanglement spectrum when chain is partitioned into two equally sized subsystems.

look at the one point functions of the $3 \times 3$ plaquette open SO SUSY model with and without defects. Finally we look at the entanglement properties for a $2 \times 4$ plaquette system with different choices of partitioning.

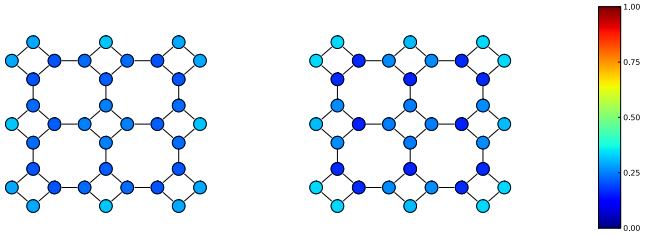
| | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.963 | 0.889 | 0.796 | 0.699 | 0.605 | 0.517 | 0.438 | 0.368 | 0.307 |
| 2 | 0905 | 0.794 | 0.663 | 0.535 | | | | | |
| 3 | 0.794 | 0.651 | | | | | | | |

Table 3.3: Overlaps of PPS wavefunction with numerically calculated ground states for open SO lattices with $m \times n$ plaquettes. The values of $m$ and $n$ are given in the first row and first column of the table.

Table 3.3 shows the overlaps between the PPS wavefunctions and numerically calculated ground states for various sized open SO lattices. It is interesting to note that the overlaps for the two and three row lattices are better than the chain with the same number of plaquettes. For example the overlap for the $3 \times 3$ lattice is significantly better than for $1 \times 9$ lattice.

For the open $3 \times 3$ plaquette open SO model the one point functions for the

PPS wavefunction are shown alongside those from the numerically calculated ground state in figure 3.33, as well being listed in table 3.8 in appendix 3.D. From these it can be seen that there is very good agreement between the one point functions of the real ground state and of the PPS ground state for this lattice. However again we see that the PPS wavefunction does not adequately capture the fact that it is more probable for the boundary sites to be occupied.

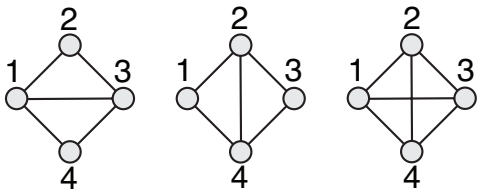



(a) (b)

Figure 3.33: One point functions in the (a) PPS wavefunction and (b) numerically calculated ground state of the $3 \times 3$ plaquette open SO lattice.

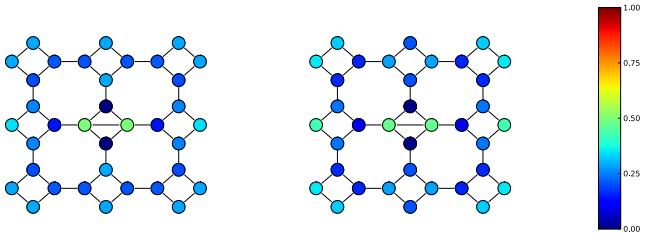
When a single horizontal defect is added to a single plaquette there are two ground states as discussed in section 3.4.3. One of these ground states is given by $|\psi_{GS_1}\rangle = |1\rangle - |2\rangle + |3\rangle - |4\rangle$ and the other by $|\psi_{GS_2}\rangle = |1\rangle - |3\rangle$ where the sites are labelled as in figure 3.34. When a single vertical defect is added again there are two ground states given by $|\psi_{GS_1}\rangle = |1\rangle - |2\rangle + |3\rangle - |4\rangle$ and $|\psi_{GS_2}\rangle = |2\rangle - |4\rangle$. When both horizontal and vertical defects are added at the same time there are three ground states given by $|\psi_{GS_1}\rangle = |1\rangle - |2\rangle + |3\rangle - |4\rangle$, $|\psi_{GS_2}\rangle = |1\rangle - |3\rangle$ and $|\psi_{GS_3}\rangle = |2\rangle - |4\rangle$. To construct the PPS wavefunctions for a SO lattice which has plaquettes featuring defects one simply includes a different single plaquette wavefunction in each PPS wavefunction for the plaquettes with defects.

In figure 3.35a the one point functions for the PPS wavefunction of the open SO lattice with a horizontal defect are shown (and values listed in table 3.8 in appendix 3.D). These values are close to the values of the numerically calculated ground state shown in figure 3.35b with the most significant devi-




(a) (b) (c)

Figure 3.34: A single plaquette with (a) horizontal, (b) vertical and (c) both horizontal and vertical defects.

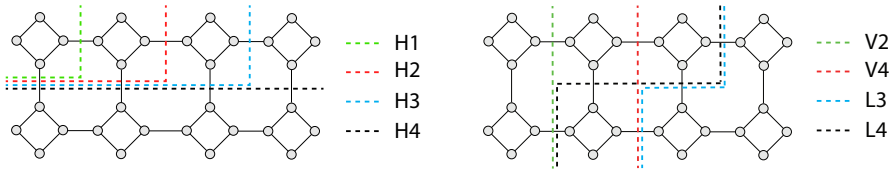



(a) (b)

Figure 3.35: One point functions for (a) the PPS wavefunction and (b) the numerically calculated ground state for the $3 \times 3$ plaquette open SO lattice with horizontal defect on the central plaquette. The numerically calculated ground state shown here is the state with slightly higher energy when the ground state degeneracy is split as described in section 3.4.3.1.

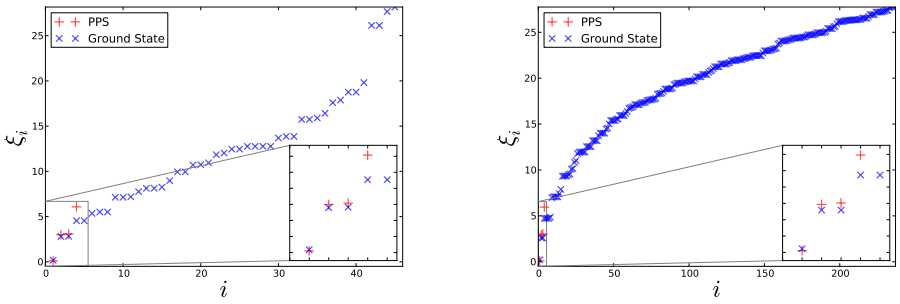
ations occurring on the boundary sites and the sites of the central plaquette. The overlap between these states is 0.634, almost as good as the case without defects. The case of the vertical defect is identical except that it is rotated by ninety degrees.

We investigated eight choices of partitioning for the $2 \times 4$ plaquette open SO system. These can be split into groups which we call horizontal, vertical and 'L' shaped. We describe each type of partitioning in terms of the shape of one of the subsystems. There are four shapes in the horizontal group labelled H1 to H4, corresponding to SO chains with one to four plaquettes. In the vertical group there are two shapes labelled V2 and V4 corresponding to $2 \times 1$ and $2 \times 2$ plaquette SO lattices. In the 'L' shape group there are two shapes labelled L3 and L4 which correspond to three and four plaquettes




(a) (b)

Figure 3.36: Different types of partitioning of the $2 \times 4$ plaquette open SO lattice with labels. (a) Shows choice of partitions H1-H4 and (b) shows V2, V4, L3 and L4.

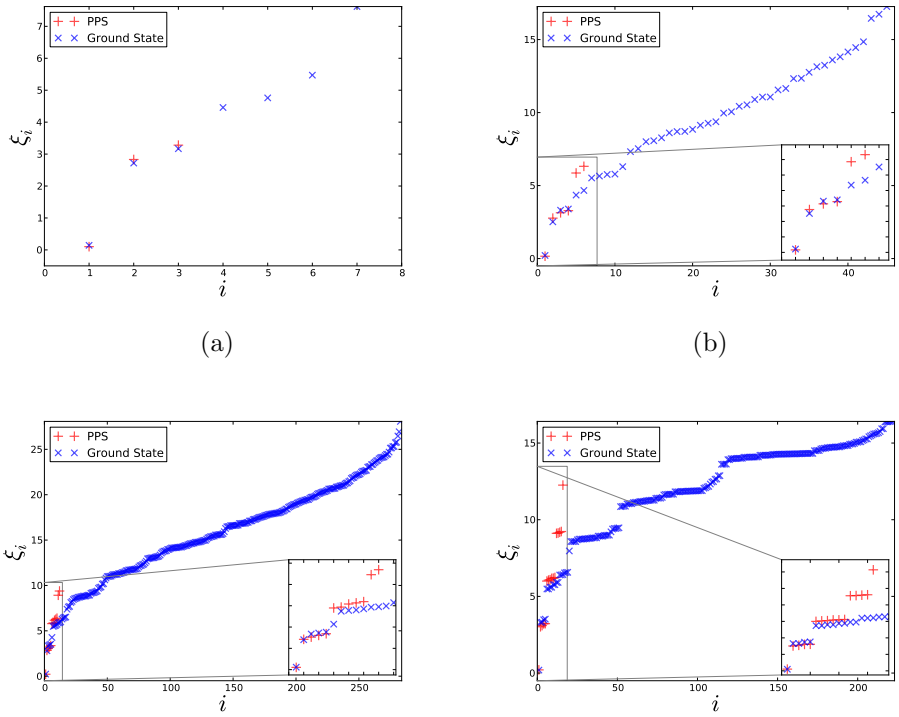



(a) (b)

Figure 3.37: Entanglement spectrum of the PPS and numerically calculated ground state of the $2 \times 4$ plaquette open SO SUSY model with partitioning types (a) V2 and (b) V4.

arranged in the shape of the letter 'L'. Illustrations of these partitions as well as labels are given in figure 3.36.

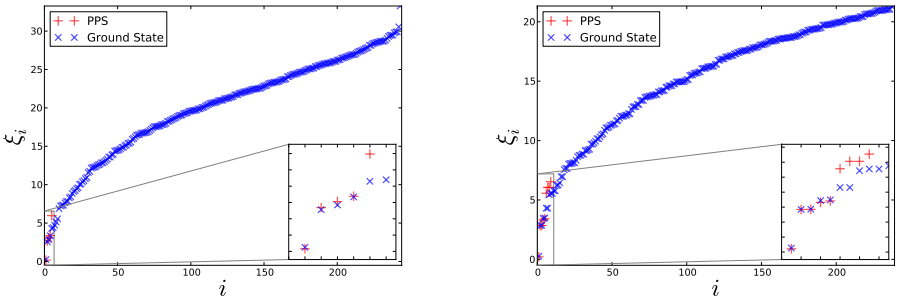
The EE and ES has been calculated for both the PPS wavefunction and numerically calculated ground state for each choice of partitioning. The EE values are shown in table 3.4 and plots of the ES for each are shown in figures 3.37,3.38 and 3.39. These calculations show that while the PPS captures the low end of the ES well, there is significant entanglement in the ground states which is not captured by the PPS wavefunction.




(a) (b)

(c) (d)

Figure 3.38: Entanglement spectrum of the PPS and numerically calculated ground state of the $2 \times 4$ plaquette open SO SUSY model with partitioning types (a) H1, (b) H2, (c) H3 and (d) H4.




(a) (b)

Figure 3.39: Entanglement spectrum of the PPS and numerically calculated ground state of the $2 \times 4$ plaquette open SO SUSY model with partitioning types (a) L3 and (b) L4.

| | V2 | V4 | H1 | H2 | H3 | H4 | L3 | L4 |
|---|---|---|---|---|---|---|---|---|
| GS | 0.6795 | 0.8156 | 0.5554 | 0.8265 | 0.9664 | 0.8763 | 0.9253 | 1.060 |
| PPS | 0.3989 | 0.4176 | 0.3821 | 0.5980 | 0.7975 | 0.7988 | 0.5975 | 0.796 |

Table 3.4: Entanglement entropy of the PPS wavefunction and numerically calculated ground state for the $2 \times 4$ plaquette open SO lattice with different partitioning types.

## 3.5 Numerical issues

A number of challenges were encountered while carrying out exact diagonalisation calculations for these SUSY lattice models. In this section these challenges are discussed along with adaptions made to the DoQO code to overcome these issues.

### 3.5.1 Basis configurations

A recursive method was developed to efficiently generate configurations which satisfy the nearest neighbour exclusion condition for arbitrary graphs. This method offers significant improvement over naive approaches which simply iterate over all possible configurations checking each.

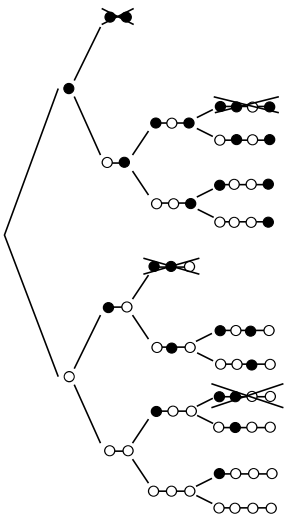


Figure 3.40: Tree representing configurations on a four site chain. Configurations that are crossed out break the nearest neighbour exclusion condition.

All the possible configurations of particles on a graph with $N$ sites can be represented by the leaves of a binary tree of depth $N+1$ where the root configuration contains no sites. The two immediate children of each configuration are related to their parent configuration by (i) the addition of an occupied site or (ii) the addition of an unoccupied site. If a particular configuration breaks the nearest neighbour exclusion condition then so too do all its descendant configurations. Thus a list of all configurations satisfying this condition can be constructed by traversing the tree and skipping any branches where the parent configuration breaks the nearest neighbour exclusion condition. Figure 3.40 shows an illustration of such a tree for a four site chain. Configurations which break the nearest neighbour exclusion condition are crossed out in this illustration. Using this algorithm it is possible to restrict to a given filling by stopping along a certain path once the desired filling has been reached.

This algorithm is easily adapted to run in parallel. In this case a breath first traversal of the tree is performed on each process until there are at least as many leaves with valid configurations as there are processes. These valid configurations are then distributed equally among the processes and each process performs depth first traversals on their configurations. Each process saves the valid configurations it encounters on its traversal to a file on disk similar to what was described in the last paragraph of section 2.3.3.1. However this list of valid configurations is not ordered according to the numeric label for each configuration. The method of building a matrix using a basis list as described in section 2.3.3.2 assumes the list is ordered. To overcome this issue a parallel quick sort algorithm was implemented to sort this distributed array of basis configurations.

## 3.A Appendix: Transforming to spin representation

Using the Jordan-Wigner transform it is possible to write the SUSY Hamiltonian in terms of spin $\frac{1}{2}$ operators. The fermionic creation and annihilation

operators can be represented using the spin half operators as:

$$c_i = (\prod_{m=1}^{i-1} \sigma_m^z)(\frac{\sigma_i^x + i\sigma_i^y}{2}), \qquad c_i^+ = (\prod_{m=1}^{i-1} \sigma_m^z)(\frac{\sigma_i^x - i\sigma_i^y}{2})$$

Using these we can rewrite the projectors $P_i$ as:

$$\begin{aligned} P_i &= 1 - n_i = 1 - c_i^\dagger c_i \\ &= 1 - (\prod_{m=1}^{i-1} \sigma_m^z)(\prod_{m=1}^{i-1} \sigma_m^z)(\frac{\sigma_i^x - i\sigma_i^y}{2})(\frac{\sigma_i^x + i\sigma_i^y}{2}) \\ &= 1 - \frac{1}{4}(\sigma_i^x\sigma_i^x + i\sigma_i^x\sigma_i^y - i\sigma_i^y\sigma_i^x + \sigma_i^y\sigma_i^y) \\ &= 1 - \frac{1}{2}(1 - \sigma_i^z) = \frac{1}{2}(1 + \sigma_i^z) \end{aligned}$$

The hopping terms are written as:

$$\begin{aligned} c_i^\dagger c_{i+1} + c_{i+1}^\dagger c_i &= (\prod_{m=1}^{i-1} \sigma_m^z)(\frac{\sigma_i^x - i\sigma_i^y}{2})(\prod_{m=1}^{i} \sigma_m^z)(\frac{\sigma_{i+1}^x + i\sigma_{i+1}^y}{2}) \\ &+ (\prod_{m=1}^{i} \sigma_m^z)(\prod_{m=1}^{i-1} \sigma_m^z)(\frac{\sigma_{i+1}^x - i\sigma_{i+1}^y}{2})(\frac{\sigma_i^x + i\sigma_i^y}{2}) \\ &= (\frac{\cancel{-i\sigma_i^y\sigma_{i+1}^x} + \sigma_i^y\sigma_{i+1}^y + \sigma_i^x\sigma_{i+1}^x + \cancel{i\sigma_i^x\sigma_{i+1}^y}}{4}) \\ &+ (\frac{\cancel{i\sigma_{i+1}^x\sigma_i^y} + \sigma_{i+1}^x\sigma_i^x + \sigma_{i+1}^y\sigma_i^y - \cancel{i\sigma_{i+1}^y\sigma_i^x}}{4}) \\ &= \frac{\sigma_{i+1}^x\sigma_i^x + \sigma_{i+1}^y\sigma_i^y}{2} \end{aligned}$$

so the full kinetic term is written as:

$$H_{kin} = \sum_i^L \frac{1}{8}(1 + \sigma_{i-1}^z)(\sigma_{i+1}^x\sigma_i^x + \sigma_{i+1}^y\sigma_i^y)(1 + \sigma_{i+2}^z)$$

However when using periodic boundary conditions with the chain this means that the last site and the first site will interact. This interaction will be different due to the convention we adopt when using fermions. For an $n$ site

chain the kinetic terms linking the start and end of the chain are

$$\begin{aligned}
c_n^\dagger c_1 + c_1^\dagger c_n &= (\prod_{m=1}^{n-1} \sigma_m^z)(\frac{\sigma_n^x - i\sigma_n^y}{2})(\frac{\sigma_1^x + i\sigma_1^y}{2}) \\
&+ (\frac{\sigma_1^x - i\sigma_1^y}{2})(\prod_{m=1}^{n-1} \sigma_m^z)(\frac{\sigma_n^x + i\sigma_n^y}{2}) \\
&= (\prod_{m=2}^{n-1} \sigma_m^z)(\frac{\cancel{i\sigma_n^x\sigma_1^y} + \sigma_n^x\sigma_1^x + \sigma_n^y\sigma_1^y - \cancel{i\sigma_n^y\sigma_1^x}}{4} \\
&+ (\prod_{m=2}^{n-1} \sigma_m^z)(\frac{\cancel{-i\sigma_1^y\sigma_n^x} + \sigma_1^y\sigma_n^y + \sigma_1^x\sigma_n^x + \cancel{i\sigma_1^x\sigma_n^y}}{4}) \\
&= (\prod_{m=2}^{n-1} \sigma_m^z)\frac{\sigma_1^x\sigma_n^x + \sigma_1^y\sigma_n^y}{2}
\end{aligned}$$

The extra chain of $\sigma^z$ operators breaks the translational invariance of the model meaning that it is no longer possible to diagonalise in each momentum sector for closed boundary conditions when using the spin $\frac{1}{2}$ representation. The potential term in spin representation is written as

$$H_{pot} = \sum_{i=1}^{L} \frac{1}{4}(1+\sigma_{i-1}^z)(1+\sigma_{i+1}^z) = \sum_{i=1}^{L} \frac{1}{4}(1+\sigma_{i-1}^z+\sigma_{i+1}^z+\sigma_{i-1}^z\sigma_{i+1}^z)$$

## 3.B Appendix: Spectral flow calculations

Spectral flow calculations involve twisting the boundary conditions of a model and can prove very useful for probing the properties of lattice models in particular in the vicinity of critical points[52]. In particular scaling properties of relevant operators of the conformal field theory can be deduced. Critical systems respond much more dramatically than non critical systems to changes in boundary conditions. This can be attributed to the infinite correlation length 'sensing' the change at the boundary. In this section we show how these calculations are performed in the context of the SUSY lattice models. No spectral flow calculations appear in this thesis, however the derivation here is still relevant as it demonstrates the use of representative configura-

tions which are used for implementing general space group symmetries.

When performing spectral flow calculations a parameter $\alpha$ is introduced to the model which is used to specify the boundary conditions. A particle crossing the boundary picks up a phase of $e^{2\pi i\alpha}$ or $e^{-2\pi i\alpha}$ if crossing in the opposite direction. Integer values of $\alpha$ correspond to periodic boundary conditions (Ramond sector) whereas half integer values of $\alpha$ correspond to anti-periodic boundary conditions (Neveu-Schwarz sector). Any terms in the Hamiltonian that cause a particle to hop across the boundary are multiplied by a phase of $e^{2\pi i\alpha}$ or $e^{-2\pi i\alpha}$ depending on the direction. A term in the Hamiltonian written as $c_N^\dagger c_1 + h.c.$ becomes $e^{2\pi i\alpha} c_N^\dagger c_1 + h.c.$.

The model remains translationally invariant and the momentum depends linearly on the twist parameter $\alpha$. The translation operator $T_\alpha^L$ that acts on a chain of length $L$ and translates it by one site has eigenvalues:

$$\lambda = e^{2\pi i \frac{p_0}{L}} e^{2\pi i \alpha \frac{F}{L}}$$

where $p_0$ is an integer with $0 \leq p_0 < L$ and $F$ is the number of fermions in $|\,\psi\rangle$. Note that the translation operator adds a phase of $e^{2\pi i\alpha}$ or $e^{-2\pi i\alpha}$ for each particle it carries over the boundary. It is also important to note that for fermionic systems in general the application of a translation operator to a configuration can result in a phase of $e^{1\pi}$ due to the fermionic anti-commutation relations.

Simultaneous eigenstates of both the Hamiltonian and translation operator for a given representative state (see section 2.1.1.3) $|\,\psi_r\rangle$ and eigenvalue $\lambda$ can be written as:

$$|\,\psi_{\lambda,r}\rangle = \frac{1}{\sqrt{\mathcal{N}_{\lambda,r}}} \sum_{j=0}^{L-1} (\lambda^*)^j (T_\alpha^L)^j |\,\psi_r\rangle$$

where the normalisation factor $\mathcal{N}_{\lambda,r}$ is calculated as:

$$\mathcal{N}_{\lambda,r} = \sum_{j=0}^{L-1} \sum_{j'=0}^{L-1} \lambda^{j-j'} \langle\psi_r\,| (T_\alpha^L)^{j'-j} |\,\psi_r\rangle$$

The sums here can be expanded to give:

$$\begin{aligned}\mathcal{N}_{\lambda,r} &= L\langle\psi_r|\psi_r\rangle + \sum_{j=1}^{L-1}(L-j)\lambda^{-j}\langle\psi_r\left|(T_\alpha^L)^j\right|\psi_r\rangle \\ &+ \sum_{j=1}^{L-1}(L-j)\lambda^{j}\langle\psi_r\left|(T_\alpha^L)^{-j}\right|\psi_r\rangle\end{aligned}$$

Then noting that $(T_\alpha^L)^{-j} = e^{-2\pi i\alpha F}(T_\alpha^L)^{L-j}$ and $\lambda^{-j} =$ and $\lambda^{-j} = e^{-2\pi i\alpha F}\lambda^{L-j}$ we rewrite $\mathcal{N}_{\lambda,r}$ as:

$$\begin{aligned}\mathcal{N}_{\lambda,r} &= L\langle\psi_r|\psi_r\rangle + L\sum_{j=1}^{L-1}e^{-2\pi i\alpha F}\lambda^{L-j}\langle\psi_r\left|(T_\alpha^L)^j\right|\psi_r\rangle \\ &= L\sum_{j=0}^{L-1}\lambda^{-j}\langle\psi_r\left|(T_\alpha^L)^j\right|\psi_r\rangle\end{aligned}$$

The result is that the expression for $\mathcal{N}_{\lambda,r}$ is simplified and only contains a single summation. The same procedure is performed for the matrix elements $\langle\psi_{\lambda,r'}\left|H\right|\psi_{\lambda,r}\rangle$:

$$\begin{aligned}\langle\psi_{\lambda,\lambda,r'}\left|H\right|\psi_{\lambda,r}\rangle &= \frac{1}{\sqrt{\mathcal{N}_{\lambda,r'}\mathcal{N}_{\lambda,r}}}(\sum_{j=0}^{L-1}\langle\psi_{r'}\left|\lambda^j((T_\alpha^L)^\dagger)^j\right)H(\sum_{j=0}^{L-1}(\lambda^*)^j(T_\alpha^L)^j\left|\psi_r\right\rangle) \\ &= \frac{1}{\sqrt{\mathcal{N}_{\lambda,r'}\mathcal{N}_{\lambda,r}}}\sum_{j=0}^{L-1}\sum_{j'=0}^{L-1}\langle\psi_{r'}\left|\lambda^j(\lambda^*)^{j'}((T_\alpha^L)^\dagger)^j(T_\alpha^L)^{j'}H\right|\psi_r\rangle \\ &= \frac{1}{\sqrt{\mathcal{N}_{\lambda,r'}\mathcal{N}_{\lambda,r}}}(L\langle\psi_{r'}\left|H\right|\psi_r\rangle + L\sum_{j=1}^{L-1}e^{-2\pi i\alpha F}\lambda^{L-j}\langle\psi_{r'}\left|(T_\alpha^L)^jH\right|\psi_r\rangle) \\ &= \frac{L}{\sqrt{\mathcal{N}_{\lambda,r'}\mathcal{N}_{\lambda,r}}}(\sum_{j=0}^{L-1}\lambda^{-j}\langle\psi_{r'}\left|(T_\alpha^L)^jH\right|\psi_r\rangle)\end{aligned}$$

The resulting expression is significantly simplified and results in significant reduction of computational resources required to perform spectral flow calculations. This technique of simplification is not unique to spectral flow calculations and translational invariance but can be used for rotation invari-

ance, spin flip invariance, and mirror symmetries.

### 3.B.1 2D Case

If the symmetry operators of multiple symmetries commute with each other then it is possible to exploit these symmetries simultaneously. Here we demonstrate the use of multiple symmetries by deriving expressions for the eigenvectors and matrix elements when performing spectral flow calculations in two dimensions.

With a two dimensional system the boundaries can be twisted independently of each other. We label the twist parameters in each direction $\alpha_0$ and $\alpha_1$ and the translation operators in each direction as $T_{\alpha_0}^{L_0}$ and $T_{\alpha_1}^{L_1}$. The eigenvalues of the symmetry operators are then written as:

$$T_{\alpha_0}^{L_0}|\,\psi\rangle = e^{2\pi i \frac{k_0+\alpha_0 F}{L_0}}|\,\psi\rangle$$
$$T_{\alpha_1}^{L_1}|\,\psi\rangle = e^{2\pi i \frac{k_1+\alpha_1 F}{L_1}}|\,\psi\rangle$$

where $0 \leq k_0 < L_0$ and $0 \leq k_1 < L_1$. The eigenvectors are then written as:

$$|\,\psi_{\vec{k},r}\rangle = \frac{1}{\sqrt{\mathcal{N}_{\vec{k},r}}} \sum_{\vec{x}=(0,0)}^{\vec{L}-1} e^{-2\pi i(\vec{x}.\frac{1}{\vec{L}}(\vec{k}+F\vec{\alpha}))}(T_{\vec{\alpha}}^{\vec{L}})^{\vec{x}}|\,\psi_r\rangle$$

where the normalisation $\mathcal{N}_{\vec{k},r}$ is:

$$\begin{aligned}\mathcal{N}_{\vec{k},r} &= \sum_{\vec{x}=(0,0)}^{\vec{L}-1}\sum_{\vec{x}'=(0,0)}^{\vec{L}-1} e^{2\pi i(\vec{x}.\frac{1}{\vec{L}}(\vec{k}+F\vec{\alpha})-\vec{r'}.\frac{1}{\vec{L}}(\vec{k}+F\vec{\alpha}))}\langle\psi_r\,|(T_{\vec{\alpha}}^{\vec{L}})^{-\vec{x}}(T_{\vec{\alpha}}^{\vec{L}})^{\vec{x}'}|\,\psi_r\rangle \\ &= L_0 L_1 \sum_{\vec{d}=(0,0)}^{\vec{L}-1} e^{2\pi i(-\vec{d})(\frac{1}{\vec{L}}.\vec{k}+\frac{1}{\vec{L}}.F\vec{\alpha})}\langle\psi_r\,|(T_{\vec{\alpha}}^{\vec{L}})^{\vec{d}}|\,\psi_r\rangle\end{aligned}$$

The matrix elements can then be written as:

$$\langle \psi_{\vec{k},r'} | H | \psi_{\vec{k},r} \rangle = \frac{1}{\sqrt{\mathcal{N}_{\vec{k},r'}\mathcal{N}_{\vec{k},r}}} \sum_{\vec{x}=(0,0)}^{\vec{L}-1} \sum_{\vec{x'}=(0,0)}^{\vec{L}-1} e^{2\pi i(\vec{x'}-\vec{x}).(\frac{1}{\vec{L}}.\vec{k}+\frac{1}{\vec{L}}.F\vec{\alpha})} \langle \psi_{r'} | (T_{\vec{\alpha}}^{\vec{L}})^{\vec{x}-\vec{x'}} H | \psi_r \rangle$$

$$= \frac{L_0 L_1}{\sqrt{\mathcal{N}_{\vec{k},r'}\mathcal{N}_{\vec{k},r}}} \sum_{\vec{d}=(0,0)}^{\vec{L}-1} e^{-2\pi i\vec{d}.(\frac{1}{\vec{L}}.\vec{k}+\frac{1}{\vec{L}}.F\vec{\alpha})} \langle \psi_{r'} | (T_{\vec{\alpha}}^{\vec{L}})^{\vec{d}} H | \psi_r \rangle$$

## 3.C Appendix: Criticality and CFT

In this appendix the basic ideas of criticality and conformal field theory (CFT) are described. These are very large areas and the treatment here is by no means complete. A critical point is a point on a phase diagram at which a continuous (also known as second order) phase transition occurs. At such a transition the macroscopic properties of the system change abruptly without a jump in the latent heat, or any discontinuity in the average value of microscopic variables, as is the case for first order phase transitions. A typical example of a critical point is the phase transition between the ferromagnetic and anti-ferromagnetic phases of the two dimensional classical Ising model [39]. Critical points only occur in the thermodynamic limit which is when the system size $L \to \infty$ or equivalently when the lattice spacing $a \to 0$. Another important property of critical points is that the correlation length $\xi$ diverges and there are fluctuations on all length scales.

The states at critical points are not only scale invariant but also conformally invariant. A CFT is a quantum field theory that is conformally invariant. CFT can be used to describe the long distance, low energy properties of critical models [53] and is particularly effective for two dimensional systems where the group of conformal transformations is infinite. For a given CFT characterised by its central charge $c$ there are a number of primary fields which transform simply when a conformal transformation is applied. The way in which a primary fields transform is described by their associated conformal dimensions. More information can be found on CFT in [54] and [55].

## 3.D Appendix: Calculated values

In this appendix we show the values of the numerically calculated one and two point functions that are referred to in this chapter. We also show dimensions of the Hilbert spaces containing the ground states for SUSY chains in table 3.12.

Table 3.5 shows the one point functions in the ground states of the $3 \times 3$ plaquette open SO lattice without defects, with a horizontal defect and with both horizontal and vertical defects on the central plaquette. In the cases with defects the ground state degeneracy was split slightly to choose a suitable basis as described in section 3.4.3.1 and states ordered by increasing energy. Labelling of sites is shown in figure 3.41. Table 3.7 shows the two point functions for this system. Here the two point functions are between the sites specified (as labelled in figure 3.41) and the sites at the opposite side of the plaquette.

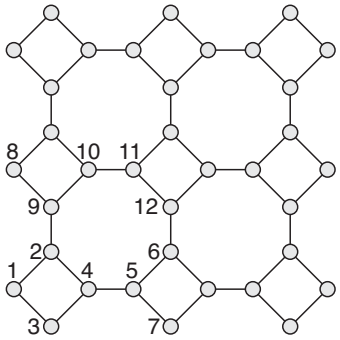



Figure 3.41: Labelling of the sites of the 3 x 3 plaquette SO lattice. The values at the unlabelled sites are the same as those at the symmetrically equivalent labelled sites.

Table 3.9 contains the values of the one point functions for the ground state and PPS wavefunction of the two plaquette open SO chain.

The values of the one point functions for the PPS wavefunctions on the $3 \times 3$ plaquette open SO lattice with and without defects on the central plaquette are shown in table 3.8. Labelling of sites is shown in figure 3.41.

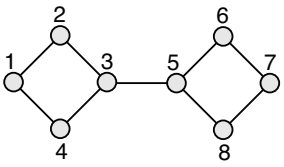



Figure 3.42: Open SO chain with two plaquettes with sites labelled.

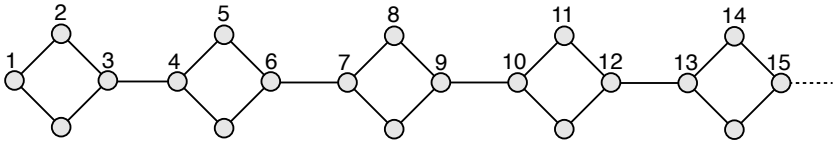



Figure 3.43: Half of ten plaquette open SO chain with sites labelled.

| Site | No defect | H defect | | HV defects | | | # |
|---|---|---|---|---|---|---|---|
| 1 | 0.33921 | 0.33922 | 0.35175 | 0.35175 | 0.33947 | 0.32719 | 4 |
| 2 | 0.16437 | 0.16407 | 0.16762 | 0.16762 | 0.16399 | 0.16036 | 4 |
| 3 | 0.33921 | 0.33947 | 0.32719 | 0.32719 | 0.33947 | 0.35175 | 4 |
| 4 | 0.16437 | 0.16428 | 0.16036 | 0.16036 | 0.16399 | 0.16762 | 4 |
| 5 | 0.26357 | 0.26384 | 0.28891 | 0.28892 | 0.26484 | 0.24077 | 4 |
| 6 | 0.15823 | 0.15884 | 0.20565 | 0.20555 | 0.15019 | 0.09483 | 2 |
| 7 | 0.31066 | 0.30953 | 0.21050 | 0.21059 | 0.32114 | 0.43167 | 2 |
| 8 | 0.31066 | 0.32237 | 0.43147 | 0.43167 | 0.32114 | 0.21059 | 2 |
| 9 | 0.26357 | 0.26458 | 0.24078 | 0.24077 | 0.26484 | 0.28892 | 4 |
| 10 | 0.15823 | 0.14953 | 0.09486 | 0.09483 | 0.15019 | 0.20555 | 2 |
| 11 | 0.24680 | 0.24464 | 0.47347 | 0.47344 | 0.24207 | 0.01070 | 2 |
| 12 | 0.24680 | 0.24418 | 0.01083 | 0.01070 | 0.24207 | 0.47344 | 2 |
| | 9.00000 | 9.00000 | 9.00000 | 9.00000 | 9.00000 | 9.00000 | 36 |

Table 3.5: One point function values for GS of $3 \times 3$ plaquette open SO lattice. Labelling of sites is shown in figure 3.41.

| Plaquette | No defect | H defect | | HV defects | | |
|---|---|---|---|---|---|---|
| 1 | 1.00717 | 1.00704 | 1.00692 | 1.00692 | 1.00692 | 1.00692 |
| 5 | 0.99603 | 0.99605 | 0.99398 | 0.99398 | 1.00101 | 1.00805 |
| 8 | 0.99603 | 1.00105 | 1.00789 | 1.00805 | 1.00101 | 0.99398 |
| 11 | 0.98721 | 0.97763 | 0.96860 | 0.96828 | 0.96827 | 0.96828 |

Table 3.6: Sum of one point function values on each plaquette for the GS of $3 \times 3$ plaquette open SO lattice. Plaquettes are labelled by the label of the site to the left of each plaquette as shown in figure 3.41.

| Site | No defect | H defect | | HV defects | | |
|---|---|---|---|---|---|---|
| 1 | 8.199e-03 | 8.192e-03 | 7.856e-03 | 7.854e-03 | 8.067e-03 | 8.276e-03 |
| 2 | 8.199e-03 | 8.073e-03 | 8.276e-03 | 8.276e-03 | 8.067e-03 | 7.854e-03 |
| 5 | 8.461e-03 | 8.470e-03 | 9.121e-03 | 9.121e-03 | 8.507e-03 | 7.897e-03 |
| 6 | 7.510e-03 | 7.462e-03 | 1.266e-04 | 1.289e-04 | 8.041e-03 | 1.595e-02 |
| 8 | 7.510e-03 | 8.087e-03 | 1.578e-02 | 1.595e-02 | 8.041e-03 | 1.289e-04 |
| 9 | 8.461e-03 | 8.499e-03 | 7.898e-03 | 7.897e-03 | 8.507e-03 | 9.121e-03 |
| 11 | 7.962e-03 | 2.475e-12 | 2.535e-12 | 2.532e-12 | 2.445e-12 | 2.682e-14 |
| 12 | 7.962e-03 | 7.865e-03 | 1.362e-04 | 1.073e-13 | 9.778e-12 | 1.013e-11 |

Table 3.7: Two point functions in ground states of $3 \times 3$ plaquette open SO lattice with and without defects.

| Site | No defect | H defect | V defect | # |
|---|---|---|---|---|
| 0 | 0.292903 | 0.292903 | 0.292903 | 4 |
| 1 | 0.207097 | 0.199851 | 0.214344 | 4 |
| 2 | 0.292903 | 0.292903 | 0.292903 | 4 |
| 3 | 0.207097 | 0.214344 | 0.199851 | 4 |
| 4 | 0.232462 | 0.210721 | 0.254202 | 4 |
| 5 | 0.216810 | 0.289279 | 0.144341 | 2 |
| 6 | 0.318267 | 0.289279 | 0.347254 | 2 |
| 7 | 0.318267 | 0.347254 | 0.289279 | 2 |
| 8 | 0.232462 | 0.254202 | 0.210721 | 4 |
| 9 | 0.216810 | 0.144341 | 0.289279 | 2 |
| 10 | 0.250000 | 0.500000 | 0.000000 | 2 |
| 11 | 0.250000 | 0.000000 | 0.500000 | 2 |
| | 9.000004 | 0.000000 | 9.000002 | 36 |

Table 3.8: One point function values for PPS ground states of $3 \times 3$ plaquette open SO lattice.

| Site | Real GS | PPS |
|---|---|---|
| 1 | 0.341014 | 0.266667 |
| 2 | 0.228111 | 0.266667 |
| 3 | 0.228111 | 0.266667 |
| 4 | 0.202765 | 0.200000 |
| 5 | 0.202765 | 0.200000 |
| 6 | 0.228111 | 0.266667 |
| 7 | 0.228111 | 0.266667 |
| 8 | 0.341014 | 0.266667 |
| | 2.000000 | 2.000002 |

Table 3.9: One point functions of the real ground state and PPS wavefunction of the two plaquette SO chain. Labelling shown in figure 3.42.

| Site | Real GS | Trial GS |
|---|---|---|
| 1 | 0.511142 | 0.267949 |
| 2 | 0.197778 | 0.267949 |
| 3 | 0.112138 | 0.196152 |
| 4 | 0.491453 | 0.215390 |
| 5 | 0.168873 | 0.287187 |
| 6 | 0.172653 | 0.210236 |
| 7 | 0.451036 | 0.211617 |
| 8 | 0.160364 | 0.288568 |
| 9 | 0.223434 | 0.211247 |
| 10 | 0.407159 | 0.211346 |
| 11 | 0.157247 | 0.288667 |
| 12 | 0.270888 | 0.211319 |
| 13 | 0.362310 | 0.211326 |
| 14 | 0.156160 | 0.288675 |
| 15 | 0.316942 | 0.211325 |
| | 10.000000 | 9.999998 |

Table 3.10: One point functions of the real ground state and trial ground state of the ten plaquette SO chain. Sites labelled as in figure 3.43.

| Sites | Real GS |
|---|---|
| 1-3 | 0.021361 |
| 4-6 | 0.026068 |
| 7-9 | 0.027321 |
| 10-12 | 0.027709 |
| 13-15 | 0.027823 |

Table 3.11: Two point functions horizontally across the plaquettes for the real ground state of the ten plaquette open SO chain. Sites labelled as in figure 3.43.

| Blocks | Closed | Open | Closed SO | Open SO |
|---|---|---|---|---|
| 1 | 3 | 3 | 4 | 4 |
| 2 | 9 | 10 | 18 | 19 |
| 3 | 30 | 35 | 94 | 100 |
| 4 | 105 | 126 | 522 | 556 |
| 5 | 378 | 462 | 2994 | 3190 |
| 6 | 1386 | 1716 | 17520 | 18670 |
| 7 | 5148 | 6435 | 103940 | 110776 |
| 8 | 19305 | 24310 | 622866 | 663895 |
| 9 | 72930 | 92378 | 3761338 | 4009414 |
| 10 | 277134 | 352716 | 22852058 | 24360799 |
| 11 | 1058148 | 1352078 | 139522266 | 148741696 |
| 12 | 4056234 | 5200300 | 855317532 | 911876356 |
| 13 | 15600900 | 20058300 | - | - |
| 14 | 60174900 | 77558760 | - | - |
| 15 | 232676280 | 300540195 | - | - |
| 16 | 901620585 | 1166803110 | - | - |

Table 3.12: Dimensions of Hilbert spaces of SUSY chains calculated using transfer matrices (section 3.1.4). For the regular chains each block is made up of three sites and for the SO chain each block is a four site plaquette. The dimension given is for the space where the filling is the same as the number of blocks. This is the filling the ground state is found at in each case.

# Chapter 4

# Kitaev honeycomb lattice model

The Kitaev honeycomb lattice model [5] is a model of interacting spin half particles which is known to exhibit both abelian and non-abelian topological phases. The Hamiltonian for this model is written as:

$$H = -\sum_{\alpha\in\{x,y,z\}}\sum_{i,j} J_\alpha K_{ij}^\alpha$$

where $K_{ij}^\alpha \equiv \sigma_i^\alpha \otimes \sigma_j^\alpha$ denotes the exchange interaction occurring between the sites $i, j$ connected by an $\alpha$ link; see figure 4.1. The plaquette operators are defined as:

$$W_p = K_{1,2}^z K_{2,3}^x K_{3,4}^y K_{4,5}^z K_{5,6}^x K_{6,1}^y$$

where the numbers 1 through 6 label lattice sites on a single hexagonal plaquette p (see figure 4.1), are the closed loop operators around each of the hexagons of the lattice. Since these commute with the Hamiltonian and with each other we may choose energy eigenvectors $|\, n\rangle$ such that $W_p = \langle n\,|W_p|\, n\rangle = \pm 1$. If $W_p = -1$, one says that the state $|\, n\rangle$ carries a vortex at $p$. The thermodynamic system is known to exist in four unique phases; see figure 4.2 and [5]. The three A phases are gapped and are related by permutations of x, y and z directions. The transition to the gapless B phase occurs when $J_\alpha = J_\beta + J_\gamma$.

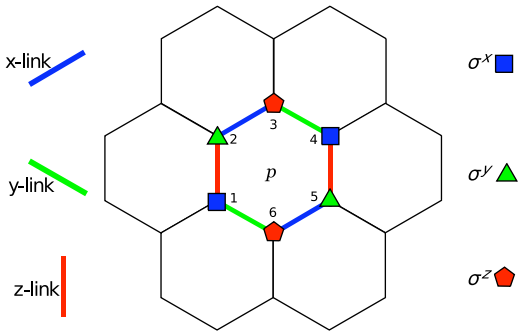



Figure 4.1: The honeycomb lattice and plaquette operator $W_p$.

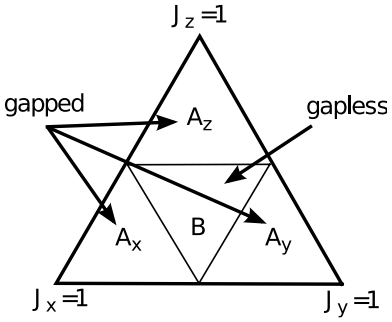



Figure 4.2: Visual representation of the system parameter space in the thermodynamic limit. The A phases are gapped. The B phase contains gapped vortices but gapless fermions [5].

## 4.1 Finite size effects

In the abelian phases the model can be mapped to the toric code [2] using Brillouin-Wigner perturbation theory[5]. This involves a perturbative expansion around the fully dimerized points found at each of the corners of the phase diagram (figure 4.2). However on finite sized tori there are additional terms related to homologically nontrivial loops which split the expected topological degeneracy. Using Brillouin-Wigner perturbation theory expressions for these finite size terms up to the fourth order were discovered. Calculations performed using DoQO numerically verified these finite size terms [12]. Note that this analysis applies only to the A phases of the system on a torus.

The lattice orientation in which we work is illustrated in figure 4.1. In order to specify a toroidal cell, one need only specify two lattice vectors. The length of the vectors gives the periodicity in that direction, i.e. the start and end points of the vector specify the same point on the torus. In general we require a minimum periodicity of two hexagon cells along any direction.

One should be aware that, because of the periodic boundary conditions, certain seemingly different vector pairs can be used to describe the same torus tiling. Note also that a rotation of a particular lattice vector pair by $\frac{2n\pi}{3}$ or reflection about the horizontal or vertical axis has the physical effect of permuting the values of $J_x, J_y$ and $J_z$. Our convention therefore will be to fix one of the lattice vectors to $a\mathbf{i}$, where $a$ is an integer. The other lattice vector may then be fixed to the positive quadrant without loss of generality. However, even with this convention there is still some redundancy in the definition and some caution must be exercised. In figure 4.3 we illustrate some possible tilings and their associated lattice vectors.

The spectrum of any toroidal system depends a great deal on the underlying configuration. In the perturbative analysis of these configurations to follow we will in general see two forms for the non-finite size fourth-order effective Hamiltonian. These Hamiltonians are locally identical (i.e. are of the form $\sum Q_p$) but have different topological degrees of freedom. The fourth-order non-finite size effective Hamiltonians that can be unitarily mapped to the toric code will be denoted as $H_K$ (K for Kitaev) and those which

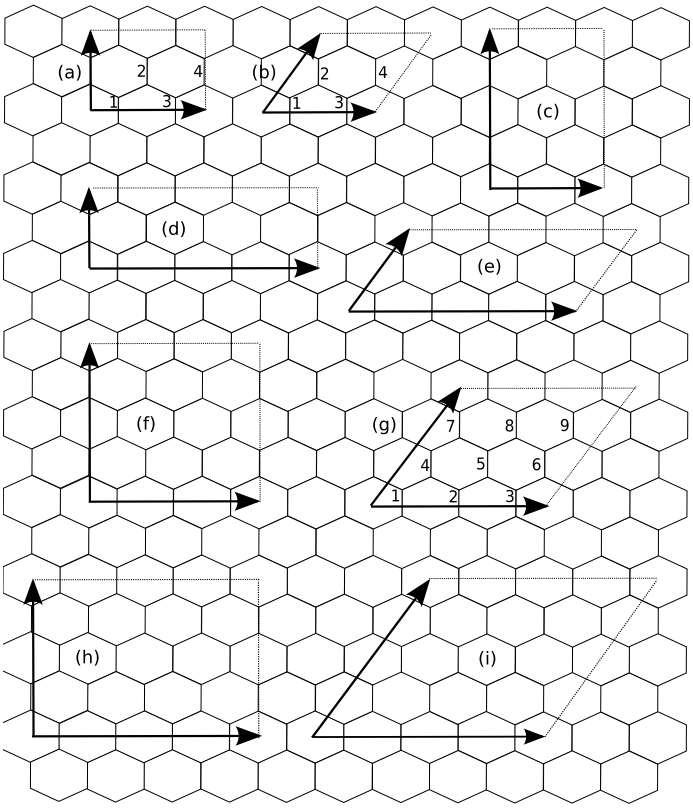



Figure 4.3: Some periodic configurations of the lattice. Each configuration is specified by two lattice vectors. Setting $\boldsymbol{n} = (\boldsymbol{i} + \sqrt{3}\boldsymbol{j})/2$ we see that configurations (a) $(2\boldsymbol{i}, 2\boldsymbol{j})$ and (b)$(2\boldsymbol{i}, 2\boldsymbol{n})$ contain 8 spins. (c) $(2\boldsymbol{i}, 4\boldsymbol{j})$, (d) $(4\boldsymbol{i}, 2\boldsymbol{j})$ and (e)$(4\boldsymbol{i}, 2\boldsymbol{n})$ are three 16-spin configurations. (f) $(3\boldsymbol{i}, 4\boldsymbol{j})$ is a 24-spin system and (g) $(3\boldsymbol{i}, 3\boldsymbol{n})$ is an 18-spin system, the only one depicted with an odd number of plaquettes. (h) $(4\boldsymbol{i}, 4\boldsymbol{j})$ and (i) $(4\boldsymbol{i}, 4\boldsymbol{n})$ are two possible 32-spin systems related to each other by a twist of the boundary conditions. Note that configurations (g) and (i) are the only configurations shown that are symmetric with respect to x, y and z links.

Table 4.1: List of toroidal configurations and type of fourth-order non-finite size Hamiltonian obtained in each A phase.

| N | Configuration | $A_x$ | $A_y$ | $A_z$ |
|---|---|---|---|---|
| 8 | (2i, 2j) | $H_W$ | $H_W$ | $H_K$ |
| 8 | (2i, 2n) | $H_K$ | $H_K$ | $H_K$ |
| 12 | (3i, 2j) | $H_W$ | $H_W$ | $H_K$ |
| 12 | (3i, 2n) | $H_W$ | $H_W$ | $H_K$ |
| 16 | (2i, 4j) | $H_K$ | $H_K$ | $H_K$ |
| 16 | (4i, 2j) | $H_W$ | $H_W$ | $H_K$ |
| 16 | (4i, 2n) | $H_K$ | $H_K$ | $H_K$ |
| 18 | (3i, 3n) | $H_W$ | $H_W$ | $H_W$ |
| 20 | (5i, 2j) | $H_W$ | $H_W$ | $H_K$ |
| 20 | (5i, 2n) | $H_W$ | $H_W$ | $H_K$ |
| 24 | (2i, 6j) | $H_W$ | $H_W$ | $H_K$ |
| 24 | (3i, 4j) | $H_W$ | $H_W$ | $H_K$ |
| 24 | (6i, 2j) | $H_W$ | $H_W$ | $H_K$ |
| 24 | (6i, 2n) | $H_K$ | $H_K$ | $H_K$ |
| 28 | (5i, 2j) | $H_W$ | $H_W$ | $H_K$ |
| 28 | (5i, 2n) | $H_W$ | $H_W$ | $H_K$ |
| 30 | (3i, 5n) | $H_W$ | $H_W$ | $H_W$ |
| 32 | (2i, 8j) | $H_K$ | $H_K$ | $H_K$ |
| 32 | (4i, 4j) | $H_K$ | $H_K$ | $H_K$ |
| 32 | (4i, 4n) | $H_K$ | $H_K$ | $H_K$ |
| 32 | (8i, 2j) | $H_W$ | $H_W$ | $H_K$ |
| 32 | (8i, 2n) | $H_K$ | $H_K$ | $H_K$ |
| 36 | (3i, 6j) | $H_W$ | $H_W$ | $H_K$ |
| 50 | (5i, 5n) | $H_W$ | $H_W$ | $H_W$ |

cannot as $H_W$ (W for Wen [56]). In the table 4.1 we list some of the possible small finite toric configurations and note the form of the non-finite size fourth-order contributions in each of the A phases. We can now review the finite size corrections that enter the perturbative expansion for small toroidal configurations.

### 4.1.1 Second-order finite size corrections

In order to calculate the second-order corrections we see that, in almost all toroidal configurations, the only two term sequences that connect up basis

elements of the dimerized subspace are those like $K^x_{ij}K^x_{ij}$ and $K^y_{ij}K^y_{ij}$. Since these terms connect each basis element to itself they are therefore constant [5]. However, in the $(a\boldsymbol{i}, 2\boldsymbol{j})$ configurations and the eight-spin $(2\boldsymbol{i}, 2\boldsymbol{n})$ configuration, because of the tight confinement, sequences like $K^x_{ij}K^y_{ij}$ and $K^y_{ij}K^x_{ij}$ can connect up different basis elements of the dimer subspace.

For the N = 8 spin $(2\boldsymbol{i}, 2\boldsymbol{n})$ configuration the non-constant second-order effective Hamiltonian is:

$$H^{(2)} = \frac{1}{2|J_z|}[J_x^2(\sigma_1^x\sigma_2^x + \sigma_3^x\sigma_4^x) + J_y^2(\sigma_2^x\sigma_3^x + \sigma_1^x\sigma_4^x)]$$

where the subscripts are shown in figure 4.3(b). For all $(a\boldsymbol{i}, 2\boldsymbol{j})$ configurations the spectral properties of the $A_z$ phase are different from the $A_x$ and $A_y$ phases. In the $A_z$ phase the second-order effective system is governed by a simple Ising spin chain Hamiltonian:

$$H^{(2)} = \frac{1}{2|J_z|}J_xJ_y\sum_{n=1}^{N/2}\sigma_n^y\sigma_{n+1}^y$$

where the subscripts (see for example figure 4.3(a)) are modulo $N/2$.

### 4.1.2 Third-order corrections

To consider the third-order perturbation correction for finite systems we play an identical game except that this time we must consider weighted sums over terms like:

$$\langle a\,|V|\,j\rangle\langle j\,|V|\,k\rangle\langle k\,|V|\,b\rangle$$

This means that we need three term sequences, $K^\alpha_{ij}K^\beta_{lm}K^\gamma_{no}$, that connect up the dimerized basis elements $|\,a\rangle$ and $|\,b\rangle$. Sequences like this occur for example in the $A_x$ and $A_y$ phases of $(a\boldsymbol{i}, 2\boldsymbol{j})$ configurations with $a > 2$, and in all the A phases of the 18-spin $(3\boldsymbol{i}, 3\boldsymbol{n})$ configuration. This 18-spin system is unusual because the unit cell is $3 \times 3$ plaquettes and it cannot be mapped to the toric code in any of its A phases.

We illustrate two of the six third-order finite-size terms in figure 4.4. The

full effective third-order Hamiltonian can be written as:

$$H^{(3)} = \frac{3}{8|J_z|^2}[J_x^3 \sum_{n=0}^{2} \sigma_{3n+1}^x \sigma_{3n+6}^x \sigma_{3n+8}^x - J_y^3 \sum_{n=0}^{2} \sigma_{n+1}^x \sigma_{n+4}^x \sigma_{n+7}^x]$$

where the numbering of the effective spins used (figure 4.3(g)) are modulo 9. Setting $J = J_x = J_y$ gives a spectrum with 3 degenerate energy levels at:

$$E^{(3)} = \begin{bmatrix} +\frac{3|J|^3}{2|J_z|^2} \\ 0 \\ -\frac{3|J|^3}{2|J_z|^2} \end{bmatrix}$$

where the upper and lower splittings are 96 times degenerate and the 0 energy term is 320 times degenerate.

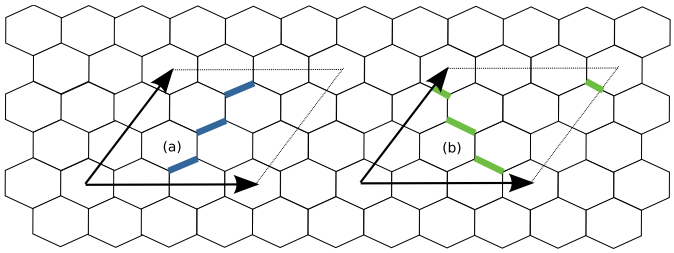



Figure 4.4: Graphical representations of two of the six third-order finite size corrections terms for the 18-spin $(3\boldsymbol{i}, 3\boldsymbol{n})$ configuration.

### 4.1.3 Fourth-order corrections

In this section we examine the additional finite size terms that appear in the fourth-order perturbative expansion. As an example we consider the 16-spin $(2\boldsymbol{i}, 4\boldsymbol{j})$ configuration in the $A_z$ phase. This particular configuration is important in that all the alternative non-constant fourth-order terms are present in one form or another. In figure 4.5 we illustrate some of the ways that different basis elements are connected for the 16-spin $(2\boldsymbol{i}, 4\boldsymbol{j})$ configuration. The plaquette terms $Q_p$ are of type (a). There are also 16 sequences that go around the torus in the vertical direction and 12 that go in the horizon-

tal direction. The overall non-constant fourth-order effective Hamiltonian is therefore quite a complicated entity with a number of different excitation types; see figure 4.5. The full fourth-order effective Hamiltonian for this configuration may be written as:

$$H^{(4)} = -\frac{J_x^2 J_y^2}{16|J_z|^3}\sum_{n=1}^{8}(Q_n + R_n - 5A_n) - \frac{J_x^2 J_y^2}{16|J_z|^3}\sum_{n=1}^{4}(Z_n + 5Y_n)$$
$$- \frac{5}{16|J_z|^3}(J_x^4\sum_{n=1}^{2}X_n + J_y^4\sum_{n=3}^{4}X_n)$$

where the $R_n$s are (horizontal) strings of the form $\sigma^z\sigma^x\sigma^z\sigma^x$, with the $\sigma^x$s

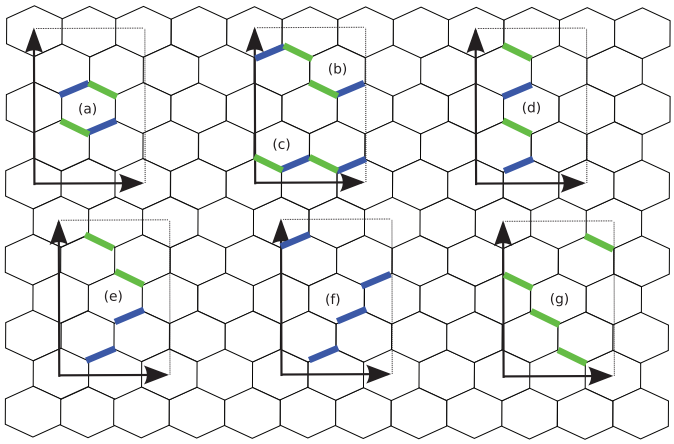



Figure 4.5: Some different four term sequences that non-trivially connect up the dimer basis vectors on the 16-spin $(2\boldsymbol{i}, 4\boldsymbol{j})$ configuration lattice. Type (a) is a plaquette term $Q_n$ and is valid for all non-horizontal configurations. Types (b) and (c) are horizontal string terms $R_n$ and $Z_n$ respectively. Type (d) and (e) are vertical strings $Y_n$ and $A_n$ respectively. Types (f) and (g) are vertical $X_n$ strings.

operating on dimers that are acted on at both ends by a $\sigma^x$ or $\sigma^y$ in the full system. The horizontal Z terms contain four effective $\sigma^z$ terms and the vertical X and Y strings contain four effective $\sigma^x$s and $\sigma^y$s respectively. The eight (vertical) A terms are mixtures of two effective $\sigma^y$ and $\sigma^x$ terms; see figure 4.5.

### 4.1.4 24-Spin $(3\boldsymbol{i}, 4\boldsymbol{j})$ computations

A case study was undertaken for the 24-spin $(3\boldsymbol{i}, 4\boldsymbol{j})$ lattice configuration where calculations were performed using the DoQO code described in chapter 2. As the system size is increased certain finite size terms drop out. Extending the 16-spin $(2\boldsymbol{i}, 4\boldsymbol{j})$ configuration (figure 4.5) to a 24-spin configuration can be done in two different ways. Extending the system vertically to a $(2\boldsymbol{i}, 6\boldsymbol{j})$ configuration means taking the X, Y and A terms from the fourth-order calculation and adding in additional Z and R terms. If we extend the system horizontally, so that we have a $(3\boldsymbol{i}, 4\boldsymbol{j})$ plaquette configuration, all the Z, R and X terms drop out while additional Y and A 'vertical' terms must be added as shown in figure 4.6.

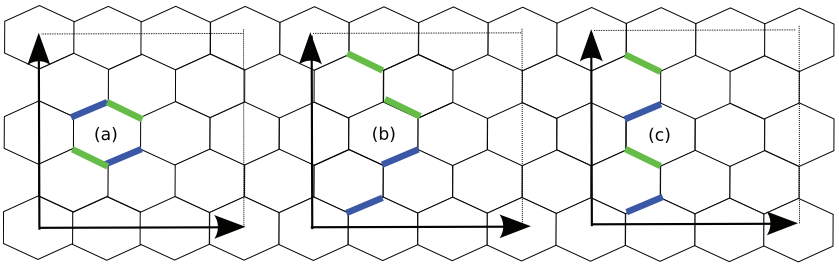



Figure 4.6: Some different four term sequences that non-trivially connect up the dimer basis vectors on the 24-spin $(3\boldsymbol{i}, 4\boldsymbol{j})$ configuration lattice. Type (a) is a plaquette term $Q_n$ and is valid for all non-horizontal configurations. Types (b) and (c) are vertical string terms $A_n$ and $Y_n$ respectively.

In this case, if we set $J = J_x = J_y$, the full effective Hamiltonian can be written as:

$$H_{eff} = cI + J_{eff}(H_K + H_{FS}^{(4)}) + O(J^6)$$

where $H_K$ is the toric code Hamiltonian on the effective lattice, $J_{eff} = J^4/(16|J_z|^3)$ and:

$$H_{FS}^{(4)} = -5(\sum_{n=1}^{6} Y_n - \sum_{n=1}^{1} 2A_n)$$

One way to demonstrate the accuracy of the above calculations is to subtract out the low order finite size contributions from the numerically calculated spectrum. This leaves the toric code contribution plus higher order

corrections. First we define $\sigma(M)$ as the appropriately ordered spectrum of any operator M and then note that:

$$\frac{\sigma(H) - E_0}{J_{eff}} - \sigma(H_{FS}^{(4)}) = \sigma(H_K) + O(J^2)$$

In figure 4.7 we plot the lowest four values of the left hand side of this equation as a function of $J^2$. The splitting of the four-fold degenerate ground state due to the sixth-order finite size effects is clearly demonstrated.

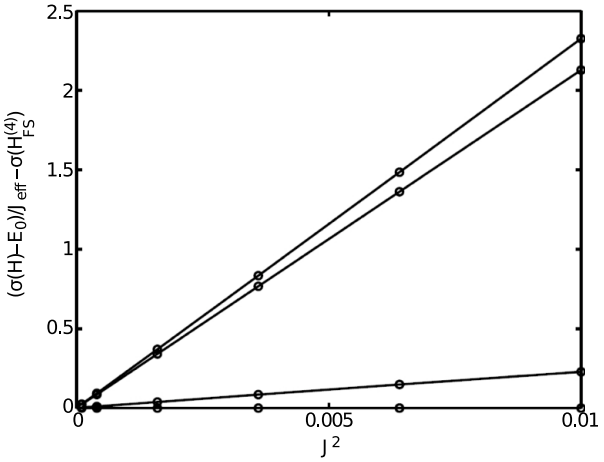



Figure 4.7: $(\sigma(H) - E_0)/J_{eff} - \sigma(H_{FS}^{(4)})$ versus $J^2$. Lifting of the $(3\boldsymbol{i}, 6\boldsymbol{j})$ toroidal honeycomb model ground state degeneracy via sixth-order finite size effects.

We now look at the details of the calculations that are summarised in figure 4.7 which shows the lowest four values of $\frac{(\sigma(H)-E_0)}{J_{eff}} - \sigma(H_{FS}^{(4)})$ plotted against $J^2$.

The calculation of $\frac{(\sigma(H)-E_0)}{J_{eff}}$ involves the diagonalisation of the Hamiltonian operator for the Kitaev honeycomb lattice model on the 24-spin $(3\boldsymbol{i}, 4\boldsymbol{j})$ lattice configuration. The basis size for this calculation is $2^{24}$ (or $16.78 \times 10^6$) and was carried out with DoQO on 2048 cores of a Blue Gene/P supercomputer. To ensure the correct ordering as indicated by the $\sigma$ notation extra terms were added which apply an energy penalty to states which contain vortices (any state for which $W_p = -1$ for any $p$). The operator that was

diagonalised was:

$$H + 50(12I - \sum_p W_p)$$

The energy of the ground state was subtracted from each of the retrieved eigenvalues and these were then blown up by dividing by $J_{eff}$.

The procedure to calculate $\sigma(H_{FS}^{(4)})$ is similar except in this case $H_{FS}^{(4)}$ is defined on the effective lattice which has 12 spins in this case and a basis size of 4096 elements. Again it is necessary to add additional terms to ensure that the states returned do not contain any vortices. To do this we define the $W_p$ operators on the effective lattice. The $W_p$ operators on the effective lattice are written as $Q_p = \sigma_d^z \sigma_r^y \sigma_u^z \sigma_l^y$ where the subscripts denote the relative position of the effective site for each plaquette, down,right,up and left respectively.

The result of this case study is that by subtracting the fourth order finite size terms we were able to reproduce the expected four fold degenerate ground state of the toric code up to the sixth order as show in figure 4.7. These sixth order terms appear as second order in $J$ resulting from the fact that the spectrum was divided by $J_{eff}$, which contains $J^4$.

### 4.1.5 36-Spin $(3\boldsymbol{i}, 6\boldsymbol{j})$ computations

It is useful to ask for what configurations the perturbative expansion to the fourth-order is equivalent to the toric code Hamiltonian. Using the arguments like those above we see that we can rule out all finite size terms at the fourth and lower orders in the $A_z$ phase of the 30-spin $(3\boldsymbol{i}, 5\boldsymbol{n})$ configuration. However, the effective fourth-order Hamiltonian is not unitarily equivalent to the toric code and is of type $H_W$ in all A phases. However, the $A_z$ phase of the 36-spin $(3\boldsymbol{i}, 6\boldsymbol{j})$ (or equivalently $(3\boldsymbol{i}, 6\boldsymbol{n})$) configuration has finite size effects on the sixth order and above only, and the fourth-order perturbative expansion can be unitarily mapped to the toric code Hamiltonian. Figure 4.8 shows a plot of the 36-spin $(3\boldsymbol{i}, 6\boldsymbol{j})$ configuration. Calculations for this lattice configuration have been performed which verify that there are no non-constant finite size effects lower than the sixth order for this configuration.

The basis set for this lattice configuration is $2^{36}$ (or $64 \times 10^9$) which is

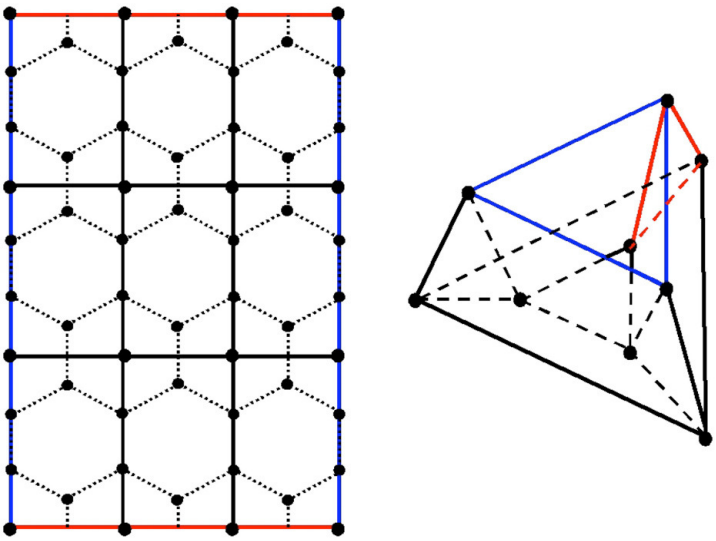


Figure 4.8: The 36-spin $(3\boldsymbol{i}, 6\boldsymbol{j})$ configuration with and its associated toric code lattice.

too large to be tractable on the available computational resources without exploiting symmetries. By exploiting the conservation of parity along each of the rows and the conservation of momentum in the horizontal direction the calculation reduces to 192 calculations, each with a basis set of $\frac{2^{30}}{3}$. It was possible to perform calculations for these using 4096 cores of the Blue Gene/P supercomputer. See section 2.1.1 for details about the symmetries DoQO is capable of employing. The results as expected show a four fold degenerate ground state up to the sixth order and agree with results obtained using the exact solution for the model [12].

# Chapter 5

# Conclusions

In conclusion I will reiterate the most salient points and refer to possible further work.

This thesis centres around the development and application of the large scale exact diagonalisation code named DoQO (Diagonalisation of Quantum Observables) [23]. DoQO is a versatile tool which can perform calculations for a broad range of models on a large range of platforms. The most significant features are its ability to work for arbitrary spin half and spinless fermionic models with many-particle interaction terms, to exploit physical symmetries and to work efficiently in parallel on high performance computing architectures. DoQO can be used effectively on standard workstations to treat systems with basis sets on the order of $10^5$, and on large capability machines it can treat systems with basis sets on the order of $10^9$ and possibly more.

While DoQO has significant capabilities, many improvements and extensions are possible some of which I will now list. The implementation of matrix free methods discussed in section 2.C could make possible larger calculations as a result of the savings in memory. Extensions to treat other particle types would allow calculations to be performed for additional systems. Some possible additional particle types include: particles with spin greater than a half, fermions with spin as well as deformed spins. In addition the ability to mix particle types could prove useful. The ability to exploit additional symme-

tries would also be useful to further reduce the basis set size and provide additional information.

A significant portion of the thesis is devoted to the investigation of SUSY lattice models (chapter 3). This treatment focuses on the staggered SUSY chain (section 3.3) and the square octagon (SO) SUSY model (section 3.4).

In section 3.3 the behaviour of the staggered SUSY chain in the staggering limits is explored in detail. Expressions for the ground states, entanglement entropy and entanglement spectrum are described and numerical verification is provided where appropriate. We are also interested in the behaviour of these models away from the staggering limits. In particular in the vicinity of the critical point at $a = 1$, and when the staggering parameter $a = \sqrt{2}$, because at this staggering the model is related to the SO SUSY chain.

For the SO SUSY model a number of results are shown in section 3.4. Calculations for the SO SUSY chain confirm that the low lying spectrum of this model is identical to that of the staggered SUSY chain at staggering $a = \sqrt{2}$. FSS calculations of the gap indicate that the closed SO chain is gapped and the open SO chain is gapless. In future we would like to establish if there is a gap in the full SO SUSY model but FSS calculations are very challenging here.

Calculations to investigate the ground state structure of the SO SUSY model were also undertaken and the results are discussed in section 3.4. Among these are calculations of the one and two point functions in the ground state(s) of the ten plaquette SO chain and the $3 \times 3$ plaquette SO SUSY model with and without defects. The Projected Product State (PPS) wavefunctions are proposed and their ability to reproduce features of the ground state are investigated. These investigations consist of calculations of overlaps as well as comparisons of one point functions and entanglement properties for each. From these investigations we conclude that the PPS wavefunctions manage to capture some of the ground state structure, as evidenced by the non exponential decay of the overlaps with increasing system size, the qualitative similarities between one point functions and the accurate reproduction of some of the low lying points of the entanglement spectrum. It is clear though from the differences in entanglement properties and one

point functions that there is significant structure that is not captured by the PPS wavefunctions. In addition the non zero values obtained for the two point functions in the ground state indicate that configurations featuring plaquettes with two fermions contribute to the ground state. In future it would be interesting to investigate so-called PPS2 wavefunctions. These supplement the standard PPS wavefunctions with configurations featuring plaquettes with two fermions.

Investigations of finite size effects in the Kitaev honeycomb lattice model were also discussed. Here analytical expressions for finite size terms up to the fourth order for finite toroidal configurations were deduced from perturbation theory. Using DoQO we were able to numerically verify these expressions and extract the finite size effects from the numerical data.

There have been other projects which have benefited from the use of DoQO that have not been mentioned in this thesis. One such project concerned the investigation of two dimensional multipartite valence bond states [57] where calculations performed using DoQO were used to verify results of DMRG calculations.

# Bibliography


[1] R. P. Feynman, The pleasure of finding things out. Basic Books, 2000.

[2] A. Kitaev, "Fault-tolerant quantum computation by anyons," Ann. Phys., vol. 303, pp. 2–30, 2003.

[3] L.-M. Duan, E. Demler, and M. D. Lukin, "Controlling spin exchange interactions of ultracold atoms in optical lattices," Phys. Rev. Lett., vol. 91, p. 090402, Aug 2003.

[4] A. Micheli, G. K. Brennen, and P. Zoller, "A toolbox for lattice-spin models with polar molecules," Nat. Phys., vol. 2, pp. 341–347, 2006.

[5] A. Kitaev, "Anyons in an exactly solved model and beyond," Ann. Phys., vol. 321, pp. 2–111, 2006.

[6] M. A. Levin and X. Wen, "String-net condensation: A physical mechanism for topological phases," Phys. Rev. B, vol. 71, p. 045110, 2005.

[7] M. H. Freedman, "A magnetic model with a possible Chern-Simons phase," Comm. Math. Phys., vol. 234, no. 1, pp. 129–183, 2003.

[8] M. Freedman, C. Nayak, and K. Shtengel, "Extended hubbard model with ring exchange: A route to a non-abelian topological phase," Phys. Rev. Lett., vol. 94, p. 066401, Feb 2005.

[9] M. Freedman, C. Nayak, and K. Shtengel, "Lieb-schultz-mattis theorem for quasitopological systems," Phys. Rev. B, vol. 78, p. 174411, Nov 2008.

[10] P. Fendley, "Topological order from quantum loops and nets," Ann. Phys., vol. 323, pp. 3113–3136, December 2008.

[11] L. Huijse, "A supersymmetric model for lattice fermions," Phd. thesis, 2010.

[12] G. Kells, N. Moran, and J. Vala, "Finite size effects in the Kitaev honeycomb lattice model on a torus," J. Stat. Mech., vol. 2009, p. P03006, March 2009.

[13] P. Fendley, K. Schoutens, and J. de Boer, "Lattice Models with N=2 Supersymmetry," Phys. Rev. Lett., vol. 90, no. 12, pp. 1–4, 2003.

[14] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," J. Chem. Phys., vol. 21, no. 6, pp. 1087–1092, 1953.

[15] M. Troyer and U. J. Wiese, "Computational Complexity and Fundamental Limitations to Fermionic Quantum Monte Carlo Simulations," Phys. Rev. Lett., vol. 94, no. 17, p. 170201, 2005.

[16] L. Tagliacozzo, G. Evenbly, and G. Vidal, "Simulation of two-dimensional quantum systems using a tree tensor network that exploits the entropic area law," Phys. Rev. B, vol. 80, p. 235127, Dec 2009.

[17] S. White, "Density matrix formulation for quantum renormalization groups," Phys. Rev. Lett., vol. 69, no. 19, pp. 2863–2866, 1992.

[18] U. Schollwöck, "The density-matrix renormalization group," Rev. Mod. Phys., vol. 77, no. 1, p. 259, 2005.

[19] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, "Matrix product state representations," Quantum Inf. Comput., vol. 7, p. 401, 2007.

[20] H. C. Jiang, Z. Y. Weng, and T. Xiang, "Accurate determination of tensor network state of quantum lattice models in two dimensions," Phys. Rev. Lett., vol. 101, p. 090603, Aug 2008.

[21] V. Murg, F. Verstraete, and J. I. Cirac, "Variational study of hard-core bosons in a two-dimensional optical lattice using projected entangled pair states," Phys. Rev. A, vol. 75, p. 033605, Mar 2007.

[22] G. Vidal, "Class of quantum many-body states that can be efficiently simulated," Phys. Rev. Lett., vol. 101, p. 110501, Sep 2008.

[23] N. Moran, G. Kells, and J. Vala, "Diagonalisation of quantum observables on regular lattices and general graphs," Comp. Phys. Comm., vol. 182, pp. 1083–1092, 2011.

[24] N.Laflorencie and D. Poilblanc, "Simulations of pure and doped low-dimensional spin-1/2 gapped systems," Lect. Notes Phys., vol. 645, pp. 227–252, 2004.

[25] A. Läuchli, "Introduction to exact diagonalization," in IPAM, Workshop in numerical approaches to quantum many-body systems, 2009.

[26] S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang, "PETSc URL." http://www.mcs.anl.gov/petsc.

[27] S. Balay, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang, "PETSc Users Manual."

[28] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, "Efficient management of parallelism in object oriented numerical software libraries," 1997.

[29] V. Hernandez, J. E. Roman, and V. Vidal, "SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems," ACM Trans. Math. Soft., vol. 31, pp. 351–362, Sept. 2005.

[30] L. Thomason, "TinyXML website." http://www.grinninglizard.com/tinyxml/.

[31] G. Lakner, B. Knudson, and C. Sosa, Blue Gene/P Application Development. http://www.redbooks.ibm.com/: IBM Redbooks, 2008.

[32] J. Nieplocha, B. Palmer, V. Tipparaju, M. Krishnan, H. Trease, and E. Apr, "Advances, Applications and Performance of the Global Arrays Shared Memory Programming Toolkit," IJHPCA, vol. 20, no. 2, pp. 203–231, Summer 2006.

[33] H. V. Eerten, "Supersymmetric Lattice Models," Masters thesis, 2005.

[34] M. Beccaria and G. De Angelis, "Exact Ground State and Finite-Size Scaling in a Supersymmetric Lattice Model," Phys. Rev. Lett., vol. 94, no. 10, pp. 1–4, 2005.

[35] P. Fendley and K. Schoutens, "Exact Results for Strongly Correlated Fermions in 2+1 Dimensions," Phys. Rev. Lett., vol. 95, no. 4, pp. 1–4, 2005.

[36] E. Witten, "Constraints on supersymmetry breaking," Nucl. Phys. B, vol. 202, no. 2, pp. 253 – 316, 1982.

[37] L. Huijse and K. Schoutens, "Supersymmetry, lattice fermions, independence complexes and cohomology theory," Adv. Theor. Math. Phys., vol. 14, pp. 643–694, 2010.

[38] H. van Eerten, "Extensive ground state entropy in supersymmetric lattice models," J. Math. Phys., vol. 46, no. 12, p. 123302, 2005.

[39] R. J. Baxter, "Exactly Solved Models in Statistical Mechanics," 1982.

[40] L. Huijse and K. Schoutens, "Superfrustration of charge degrees of freedom," Eur. Phys. J. B, vol. 64, no. 3, pp. 543–550–550, 2008.

[41] P. Fendley, B. Nienhuis, and K. Schoutens, "Lattice fermion models with supersymmetry," J. Phys. A, vol. 36, p. 12399, 2003.

[42] A. Feiguin, S. Trebst, A. W. W. Ludwig, M. Troyer, A. Kitaev, Z. Wang, and M. H. Freedman, "Interacting anyons in topological quantum liquids: The golden chain," Phys. Rev. Lett., vol. 98, pp. 1–5, April 2006.

[43] H. Li and F. D. M. Haldane, "Entanglement spectrum as a generalization of entanglement entropy: Identification of topological order in non-abelian fractional quantum hall effect states," Phys. Rev. Lett., vol. 101, p. 010504, Jul 2008.

[44] P. Calabrese and J. Cardy, "Entanglement entropy and quantum field theory," J. Stat. Mech, vol. 2004, no. 06, p. P06002, 2004.

[45] P. Calabrese and J. Cardy, "Entanglement entropy and quantum field theory: a non-technical introduction," Int. J. Quant. Inf., vol. 4, p. 429, 2006.

[46] C. Holzhey, F. Larsen, and F. Wilczek, "Geometric and renormalized entropy in conformal field theory," Nucl. Phys. B, vol. 424, no. 3, pp. 443 – 467, 1994.

[47] P. Fendley and C. Hagendorf, "Exact and simple results for the xyz and strongly interacting fermion chains," J. Phys. A: Math. Theor., vol. 43, no. 40, p. 402004, 2010.

[48] P. Fendley and C. Hagendorf, "Ground-state properties of a supersymmetric fermion chain," arXiv:1011.6386v1, 2010.

[49] L. Huijse and K. Schoutens private communication.

[50] L. Huijse and K. Schoutens, "Quantum phases of supersymmetric lattice models," preprint, vol. 0910.2386, 2009.

[51] L. Huijse and K. Schoutens, "Quantum phases of supersymmetric lattice models," no. 1, pp. 1–5, 2009.

[52] L. Huijse, J. Halverson, P. Fendley, and K. Schoutens, "Charge frustration and quantum criticality for strongly correlated fermions," Phys. Rev. Lett., vol. 101, no. 146406, 2008.

[53] K. Schoutens, "The seven faces of cft," unpublished, April 2005.

[54] J. Cardy, "Conformal field theory and statistical mechanics," July 2008. Notes from les Houches Summer school.

[55] P. Ginsparg, "Applied conformal field theory," arXiv:hep-th/9108028v1, 1988.

[56] X.-G. Wen, "Quantum orders in an exact soluble model," Phys. Rev. Lett., vol. 90, p. 016803, Jan 2003.

[57] E. Rico, R. Hübener, S. Montangero, N. Moran, B. Pirvu, J. Vala, and H. Briegel, "Valence Bond States: Link models," Ann. Phys., vol. 324, pp. 1875–1896, September 2008.

# Acknowledgements

I would like to thank many people for their help and support over the course of my Phd, in matters both directly related to and unrelated to my studies.

I would like to sincerely thank my advisor Jiri Vala for his constant support, belief and encouragement over the years. I also wish to extend thanks to Graham whose help was invaluable, particularly in the early stages. I extend my thanks to my collaborators Liza Huijse, Kareljan Schoutens and Enrique Rico. In addition I would like to thank Ahmet for his help and advice. For the smooth running of the department, their approachability and openness I wish to sincerely thank Danny, Monica, Charles and Jonivar.

I extend special thanks to my parents Pat and Rita who have always been there for me and also Niamh, Peter, Laura and Conor. In addition I would like to thank the many people I have had the opportunity to spend time with while at Maynooth. In particular house mates, office mates, lunch time regulars for stimulating lunchtime discussions and those from the surf and juggling clubs for much needed distractions.