Applications of Tensor Network Algorithms in Quantum Many-Body Physics

A Dissertation Presented

by

Colin G. West

to

The Graduate School

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Physics

Stony Brook University

May 2016

Stony Brook University

The Graduate School

Colin G. West

We, the dissertation committee for the above candidate for the Doctor of Philosophy degree, hereby recommend acceptance of this dissertation.

Tzu-Chieh Wei – Dissertation Advisor
Assistant Professor, Department of Physics and Astronomy

Lukasz Fidkowski – Chairperson of Defense
Assistant Professor, Department of Physics and Astronomy

Dominik Schneble
Associate Professor, Department of Physics and Astronomy

Sriram Ganeshan
Research Assistant Professor
Simons Center for Geometry and Physics

This dissertation is accepted by the Graduate School.

Charles Taber
Dean of the Graduate School

Abstract of the Dissertation

Applications of Tensor Network Algorithms in Quantum Many-Body Physics

by

Colin G. West

Doctor of Philosophy

in

Physics

Stony Brook University

2016

The classical simulation of many-body quantum systems is an essential tool in understanding many fundamental aspects of condensed matter physics. But a major obstacle arises from the number of degrees of freedom involved in describing such systems, which is exponential in the system size. Recently, however, a class of numerical techniques based on structures called "tensor networks" has emerged, which allows many "typical" quantum states (such as the ground states of gapped, local Hamiltonians) to be represented much more efficiently.

In this work we extend and apply these techniques to consider several central topics in quantum many-body physics. After reviewing the relevant background material from the field of tensor networks and tensor network states, we demonstrate a method for computing high order moments and cumulants of operators with respect to such states, including the so-called "Binder cumulant," a powerful tool for detecting phase transitions. Next, we employ tensor network algorithms to characterize the ground state phase diagram a quantum spin model, including both symmetry-breaking phases and symmetry protected topological order (SPT), and find a significant variety of phases and phase transitions. Finally, we consider the entanglement properties of quantum states exhibiting many-body localization, using a combination of exact diagonalization and tensor network techniques.

To my wife and parents

Contents

List of Figures **ix**

Acknowledgements **xxiv**

1 Introduction **1**

2 Tensor Network States **5**
2.1 Tensors and Tensor Networks 5
2.1.1 Graphical Tensor Notation 6
2.2 Tensor Network States: Structure and Motivation 10
2.2.1 Truncated Singular Value Decompositions 11
2.2.2 Matrix Product States 13
2.2.3 Matrix Product Operators 21
2.2.4 PEPS 24

3 Ground State Preparation Algorithms **26**
3.1 DMRG 26
3.1.1 Basic Algorithm 27
3.1.2 Additional Applications and Extensions 37
3.2 TEBD and iTEBD 38
3.2.1 Basic Algorithm 43
3.2.2 Two-Site iTEBD 50
3.2.3 L-Site iTEBD 52
3.2.4 Two Dimensional Systems 57
3.2.5 Comments on Convergence Schemes 60

4 Moments and Cumulants with Tensor Network States **69**
4.1 Moments, Cumulants, and The Binder Cumulant 70
4.1.1 Binder's Cumulant 71
4.2 Expectation Values for Product Operators 72
4.2.1 Finite-Length Chains 72

4.2.2 Infinite-Length Chains 73
4.3 Evaluating Higher-Order Moments and Cumulants 78
4.3.1 Moment and Cumulant-Generating Functions 78
4.3.2 Evaluating Generating Functions on a Finite Matrix Product State 80
4.3.3 Evaluating Generating Functions on an Infinite Matrix Product State 86
4.3.4 Higher-Dimensional States 89
4.4 Examples 90
4.4.1 Spin-1/2 Transverse Ising Model 90
4.4.2 Spin-1 Transverse Ising Model 99
4.4.3 Spin-1 Ising Model in Crystal Field 100
4.4.4 Spin-1/2 Ising Model on a 2D Lattice 104
4.5 Summary 109

5 Detection of Gapped Phases of a 1D Spin Chain with Onsite and Spatial Symmetries 111
5.1 Introduction 111
5.2 Overview of Main Results 113
5.2.1 The Hamiltonian 113
5.2.2 Summary of Numerical Results 114
5.3 Review of Classification System 118
5.3.1 Symmetry Breaking 120
5.3.2 Onsite Symmetry 121
5.3.3 Parity 122
5.3.4 Parity and Onsite Symmetry 124
5.4 Using the Parameters to Understand the Phases of the A_4 Hamiltonian 125
5.4.1 Details of the Phase Diagram 125
5.5 Numerical Methods for Obtaining the Phase Diagram 128
5.5.1 Ground State Preparation 129
5.5.2 Symmetry Detection and Extraction of Order Parameters 130
5.5.3 Obtaining the SPT Labels $\{\omega, \beta(P), \gamma(g)\}$ 138
5.6 Summary and Future Directions 141

6 Many-Body Localization and Entanglement 143
6.1 Many-Body Localization: an Overview 144
6.1.1 Thermalization in Closed Quantum Systems 145
6.1.2 Localization 147
6.1.3 Indicators of Localization 148
6.2 MBL States with Matrix Product Algorithms 150

6.2.1 SIMPS 150
6.2.2 Comments on Other Algorithms 158
6.3 Entanglement Results 159
6.3.1 Random-Field Heisenberg Model 159
6.3.2 Concurrence 160
6.3.3 Negativity 161
6.3.4 Relationship Between Entanglement Measures 169
6.4 Summary and Outlook 172

7 Conclusions 175

Bibliography 177

A Extrapolation with the BST Algorithm 200

B Proof of Projector Property for Unital Quantum Channels 203

C Cholesky Decomposition of a Positive Semidefinite Matrix 205

List of Figures

2.1 (Color Online). Examples of the graphical notation for tensor operations. (a) Examples of tensor. A solid shape represents a tensor, with extended lined representing indices. Here evidently A is a rank-three tensor, and B is rank-one, i.e. it is a vector. (b) Adjacent tensors with imply a tensor product. This diagram represents the equation $M = M_1 \otimes M_2$. (c) Tensor contraction is represented by connecting indices. The equation $C^{\alpha,\beta} = A^{\alpha,\beta,\gamma} B^{\gamma}$ is represented here. (d) By changing our labelling convention, the indices of a tensor can be "grouped" or ungrouped. In graphical notation, this can be represented by literally grouping the lines. Pictured here is the vectorization of a matrix: $V_{\alpha,\beta} \to V_{(\alpha\beta)}$ 8

2.2 (Color Online). Additional examples of graphical tensor notation. (a) Clearly, since $A\mathbb{1}B = AB$, then in our graphical notation we can simply represent the identity matrix with a line. (b) Tracing over a matrix is shown by connecting its two indices, contracting it with itself. (c) Tracing in this sense can happen over any pair of indices originating from the same tensor. Pictured here is the equation $X_\beta = Z^{\alpha}_{\alpha,\beta}$. This is related to the notion of a "partial trace," which we shall also discuss in Chapter 6. 9

2.3 (Color Online). The coefficients for a one-dimensional spin system of length L could be represented by an enormous tensor $c_{s_1 \dots s_L}$. However, we may be able to iteratively reduce the overall size of the tensor structure needed to specify the state by means of a truncated singular value decomposition. First, we conceptually group all of the indices except that for the first site, leaving a rectangular matrix. This matrix can then be decomposed and simplified by truncated SVD. The matrices Λ and $V^\dagger_{S_{2\dots L}}$ collectively serve as a large coefficient tensor for all of the state but the first site. This tensor can in turn be further decomposed by the same process. 14

2.4 (Color Online). The structures of common tensor network systems. (a) A matrix product state (MPS) for a system of length L with open boundary conditions, as in Eq. 2.8. The physical indices which relate to the physical degrees of freedom are explicitly labelled. The indices connecting the $A_{[i]}$ tensors to one another are called "virtual" indices. (b) A matrix product *operator*, as in Eq. 2.19. The left and right boundary vectors are explicitly indicated to clarify the structure, although they could have been subsumed by the definitions of $M_{[1]}$ and $M_{[L]}$. (c) A two dimensional tensor network state or "PEPS", for the particular case of a square lattice. 17

2.5 (Color Online). The transfer matrix of a translationally-invariant matrix product state, demonstrated in graphical tensor notation. In (a), the construction of the transfer matrix is shown as a contraction of two MPS matrices, with the virtual indices grouped to form a single matrix. In (b), the relationship between the transfer matrix and the norm square of the state is shown. Finally, in (c) we show graphically the behavior of a matrix product state in canonical form: such a state has a transfer matrix whose dominant eigenvector is a vectorized version of the identity matrix. In particular, this demonstrates the "right" version of canonical form; the condition for left canonical form appears the same, but with a horizontal reflection. 22

3.1 (Color online) (a) The matrix H_{eff} with respect to some site j of a system with open boundary conditions, illustrated in graphical notation to show its internal structure, as described in Eq. 3.6. The incoming and outgoing indices α and β are composite indices which group two virtual indices and a physical index of M_j, where the M tensors specify the total Hamiltonian H as an MPO. (b) Similar structure showing the effective norm operator N_{eff} with respect to site j. See Eq. 3.7 31

3.2 (Color online) With open boundary conditions, the effective norm tensor N_{eff} at site j can be substantially simplified if certain conditions are enforced. (a) First, we must place the state into a left canonical form for all tensors $A_{[i]}$left of site j, i.e. $i < j$. (b) Next, we must similarly place all tensors $A_{[i]}$ for $i > j$ in a *right* canonical form. (c) As a result, the tensor N_eff can be "collapsed in" by repeated applications of properties (a) and (b) (recall that the initial structure has only a null index of dimension zero at the boundary). 33

3.3 (Color online) The structure of the effective operators (a) H_{eff} and (b) N_{eff} for the case of a system with periodic boundary conditions; compare to Fig. 3.1. Note also that these structure can be though of as a product of transfer matrices; one such transfer matrix is highlighted here in green. 35

3.4 (Color online) Graphical notation demonstrating the application of $e^{-\delta\tau H}$ for $H = \sum H_{odd} + H_{even}$ to an MPS. The evolution operators is decomposed into three "layers" by a second order Suzuki-Trotter decomposition (Eq. 3.23). Within each layer, terms are mutually commuting and can be applied efficiently to the MPS through the "master step" of the TEBD algorithm (Fig. 3.6). This sequence of layers can then be repeated many times until the desired amount of imaginary time evolution is achieved. 41

3.5 (Color online) (a) Graphical notation demonstrating the procedure for decomposing a typical matrix product state into the "Γ, Λ" form of Ref. [81]. The tensor $A_{[1]}$ is decomposed by SVD, with the resulting unitary matrix $V_{[1]}^{\dagger}$ "pushed" to the right to form a new tensor $A_{[2]}$, as highlighted in green. The new tensor $A_{[2]}$ is then similarly decomposed and the process iterates. (b) A complete MPS represented in Γ, Λ form. Note that the entanglement entropy across any bond j can be quickly computed by considering the entries of the diagonal matrix $\Lambda_{[j]}$. 43

3.6 (Color online) Graphical notation outlining the “master step” of the TEBD algorithm for the case of a two-body Hamiltonian decomposed into two-body operators by Suzuki-Trotter. In a single step, one such two-body operator U is applied to sites j and $j+1$. (a) First, a tensor Θ_0 is constructed from the MPS tensors immediately surrounding the sites in question: $\Lambda_{[j-1]}$ $\Gamma_{[j]}$, $\Lambda_{[j]}$, $\Gamma_{[j+1]}$, and $\Lambda_{[j+1]}$. The tensors $\Lambda_{[j-1]}$ and $\Lambda_{[j+1]}$ are included to provide information about how this particular two-body subsystem relates to the “environment” formed by the rest of the chain. (b) Next, this tensor, which possesses both physical and virtual indices, is updated with the operator U to give a tensor Θ. (c) By grouping the virtual and physical indices as shown we can re-interpret Θ as a matrix. This matrix then undergoes a singular value decomposition (SVD), with the smallest singular values truncated away to prevent unwanted increase in the bond dimension of the MPS. (d) For the tensors resulting from the SVD, we “ungroup” the virtual and physical indices and again interpret them as a set of tensors $\tilde{\Gamma}_{[j]}$, $\tilde{\Lambda}_{[j]}$, and $\tilde{\Gamma}_{[j+1]}$. (e) Finally, we update our MPS with a new set of tensors which “undo” the inclusion of the environment tensors in (a), giving $\Gamma_{[j \text{ new}]} = \Lambda^{-1}_{[j-1]}\tilde{\Gamma}_{[j]}$ and $\Gamma_{[j+1 \text{ new}]} = \tilde{\Gamma}_{[j]}\Lambda^{-1}_{[j+1]}$. The updated matrix $\Lambda_{[j]}$ is simply given by $\tilde{\Lambda}_{[j]}$ (not pictured). . . 46

3.7 (Color online). A graphical representation of a translationally invariant infinite state presented using a two-site unit cell of tensors, $\{\Gamma_{[A]}, \Lambda_{[A]}, \Gamma_{[B]}, \Lambda_{[B]}\}$. These tensors may be chosen in a two-site representation because the state possesses only two-site translation invariance. But even if the state posesses translational invariance on the one-site level, this format is necessary for the iTEBD algorithm described below. 51

3.8 (Color online) Graphical notation for an example of the “master step” of the iTEBD algorithm. Compare to the case of a finite TEBD algorithm in Fig. 3.6. The key distinction is simply that, because of the translational invariance of the state, the environment tensors included in (a) are both given by $\Lambda_{[B]}$. Note that a full iTEBD algorithm will alternate between steps acting first on odd sites and steps acting first on even sites. The latter case corresponds to simply exchanging the labels A and B everywhere in this diagram. 53

3.9 Graphical notation for an example of the "master step" of the iTEBD algorithm generalized to an arbitrary unit cell of length L (compare to the case of a finite TEBD algorithm in Fig. 3.6 and the special two-site case in Fig. 3.8). The principle difference from the two-site case occurs here in step (c). The tensor Θ will possess L physical indices, and we perform our SVD by grouping only the first index to the left, and the remaining indices to the right. In steps (d) and (e), the result of the SVD gives updated tensors for $\Gamma_{[1]}$ and $\Lambda_{[1]}$, and also gives us an updated version of the tensor Θ which now has only $L-1$ physical indices. The steps (c) through (e) are repeated with this new tensor Θ until only two physical indices remain, at which point we decompose Θ into the remaining tensors by the normal two-step process. Note that, as in the two-site case, the full implementation of this algorithm involves applying operators which act on the sites of the unit cell in different sequences, which all differ by circular permutation. Illustrated here the algorithm acts first with an operator on site 1. The remaining steps look the same, but with the labels circularly permuted so that $1 \to 2, 2 \to 3 \ldots L \to 1$. 55

3.10 (Color online) An arbitrary portion of two dimensional tensor network state (PEPS) on a square lattice, decomposed into Γ, Λ form for a TEBD-type algorithm. The tensors which will be updated during a TEBD step are highlighted in green. By contrast to the one-dimensional case, note the following. First, note that the unit cell for such a system will likely have many more free virtual indices, with the exact number depending on the coordination number of the lattice. This, along with the difficulties involved in contracting two-dimensional tensor networks, places sharp restrictions on the bond dimensions which can be used. Unlike the finite cases, where bond dimensions typically range into the hundreds, many PEPS systems must be computed with a bond dimension of only two or three. Additionally, there are now two types of Λ associated with each site, one in the horizontal direction $\Lambda^{[h]}$, and one in the vertical $\Lambda^{[v]}$ (we have used the convention that the Λ's are labelled by the site below them and to the left). Next, note that at a minimum, six matrices Λ must be included to provide information about the "environment" of this cell; in fact, as commented in the text, even this is not sufficient to give complete information in two dimensions. Finally, observe that in addition to an alternating sequence of operators which will be applied to a cell like this (acting first on the odd site, then acting first on the even site, etc) we will also need to consider a *different* unit cell in the horizontal direction, and perform updates to that as well in order to achieve a complete update of the whole system. 59

3.11 A test of the energy convergence behavior during the iTBED algorithm, as applied to the spin-1 Heisenberg model with a transverse field. After each step, the difference in energy compared to the previous step is plotted. The segments separated by sharp transitions indicate the different stages of the algorithm, with the size of the imaginary time step, $\delta\tau$, decreased for each subsequent stage. Because the changes in the energy can become so small during the course of any individual stage, one might erroneously believe that convergence had been achieved, but moving to a smaller time step reveals that this is not so. 61

3.12 (Color Online) A test of the convergence behavior of E, S, and $\langle S_x \rangle$ during the iTBED algorithm, as applied to the spin-1 Heisenberg model with a transverse field. Every 100 steps, the previous 300 steps are sampled and the standard deviation is computed, and then divided by the mean. The resulting coefficient of variation is plotted in the figure above. The curve for the energy is consistently much lower, indicating that this quantity converges much more quickly than the others. 62

3.13 A test comparing the convergence behavior of the energy (left axis) and $\langle \Delta H^2 \rangle$ (right axis) during the iTBED algorithm as applied to the spin-1 Heisenberg model with a transverse field. The underlying data are the same as Fig. 3.12. Though the energy has converged to the level of one part in 10^5 after approximately 1000 steps, the energy cumulant takes much longer to reach a similar level of precision. In part, this is because the cumulant requires evolution with smaller time steps, in order to avoid plateaus and fully converge. 64

3.14 (Color Online) The transverse Ising model on a chain of L = 30 is studied by a basic TEBD algorithm. The initial imaginary time step is $\delta\tau = 0.1$. After every 500 steps of evolution, the step size is decreased either by a factor of 3 (triangles) or a factor of 10 (circles). In the early stages (steps 500-1500) the low-factor scheme is outperformed, because it spends more time stuck in local plateaus. In the later stages, however, the large-factor scheme falls behind. The evolution between steps 1000 and 1500 is truncated too early, before it had locally converged, and the subsequent time steps are too small to cause a meaningful change in the state. 65

3.15 (Color Online) The transverse Ising model on a chain of L = 30 is studied by a basic TEBD algorithm. In contrast with the convergence scheme used in Fig. 3.14, here we use the coefficient of variation from the 30 most recent points to determine when to move on to a smaller step size. The result is that unnecessary plateaus are largely avoided, and no time step is terminated before it has finished producing a useful change in the state. As a result, the energy cumulant $\langle \Delta H^2 \rangle$ is always quite similar regardless of the step size. The final energy computed for the ground state differs by less than one part in 10^7. 67

4.1 Graphical notation demonstrating the structures of matrix product states. In this notation, a shape represents a tensor, and a line represents an index. Connected lines between shapes represent contracted indices between tensors. (a) A finite spin chain state $|\psi\rangle$ represented as a matrix product state. The state is specified by the set of rank-three tensors $\{A_j\}$, with the physical degrees of freedom s_j left open. (b) The expectation value of a product operator $Q = \otimes_j Q_j$ with respect to $|\psi\rangle$. Each Q_j acts locally on only one site. The total expectation value can be thought of as a trace over a product of transfer matrices T_j, defined in Eq. (2.15). An example of an individual transfer matrix, T_1 is highlighted. 74

4.2 (a) An infinite spin chain state $|\psi_{inf}\rangle$, possessing translation invariance with respect to a unit cell of length $\ell = 2$, represented as a matrix product state. (b) A product operator $Q = \otimes_j Q_j$ which possesses the same translation symmetry as $|\psi\rangle$; i.e. $Q_j = Q_{j+\ell}$ (c) To compute the quantity of interest, we first construct T_ℓ, the transfer matrix containing an entire unit cell of $|\psi\rangle$ and Q, and extract its dominant eigenvalue λ_1. (d) To normalize the result, we will also need $\tilde{T}_\ell$ (a transfer matrix which contains only the identity operator) and it's dominant eigenvalue $\tilde{\lambda}_1$. The desired quantity $\lim_{L\to\infty} \frac{1}{L} \log\langle Q\rangle$ is given by $\log(\lambda_1/\tilde{\lambda}_1)^{1/\ell}$. . . 77

4.3 (a) Graphical representation of the moment-generating function $F(a) = \langle e^{aO}\rangle$ for an operator $M = \sum_j O_j$. Since each term in M acts at only one site, the moment-generating operator possesses the same structure, even though the moments M^n are fundamentally non-local. (b) The moment-generating function for an operator which is the sum of two-body terms and which possesses the form $H = H_{odd} + H_{even}$, such as the transverse Ising Hamiltonian defined in Eq. (4.37). The operator is approximated by the second-order Suzuki-Trotter formula in Eq. (4.42), which produces three "layers" of operations. Each layer is a sum of two-body terms. 84

4.4 The moment-generating operator e^{aM} for an operator of the form $M = \sum_j \sum_k O_{jk}$, applied to a two-dimensional state on a square lattice. As in the one-dimensional case, the locality of each term in M ensures the locality of the terms in e^{aM}, and hence, the moment-generating operator can still be evaluated all at once, by applying the appropriate onsite operator at each lattice site. 91

4.5 (Color online) A Binder cumulant study of the transverse Ising model. The cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e. L = 10 and L = 15. These crossing values can then be seen to approach the known value of the critical field, $B_c = 1$ (inset). The BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.001(1)$. 93

4.6 (Color online) Per-site value of the second cumulant of the longitudinal magnetization, $\frac{1}{L}\langle \Delta M_x^2 \rangle = \frac{1}{L}(\langle M_x^2 \rangle - \langle M_x \rangle^2)$, for the transverse Ising model. The cumulant is plotted for various finite system sizes, plotted against a range of applied fields. As the system length increases, the behavior tends towards the infinite limit (inset). In the limit, the cumulant diverges at the critical point. 95

4.7 (Color online) Second cumulant of the transverse magnetization, $\langle \Delta M_z^2 \rangle = \langle M_z^2 \rangle - \langle M_z \rangle^2$, for the transverse Ising model (computed per site). The cumulant is plotted for various finite system sizes, plotted against a range of applied fields. As the system length increases, the behavior tends towards the infinite limit (inset) where the derivative of the cumulant shows a discontinuity at the critical point. This behavior is in excellent agreement with the analytical result for the thermodynamic limit, that $\langle \Delta M_z^2 \rangle / L = 1$ for $B < 1$ and $1/B^2$ for $B > 1$. . . . 96

4.8 (Color online) The Binder cumulants for the transverse Ising model, plotted for a variety of system sizes as a function of $L^{1/\nu}(B - B_c)$ for the known values $\nu = 1$ and $B_c = 1$. As expected, for these values the curves are seen to collapse to a functional form essentially independent of the length scale. This property can be used to estimate the values of the critical point and the critical exponent by treating them as fit parameters and optimizing the collapse. 97

4.9 (Color online) The Binder cumulants for a finite sub-block of the infinite transverse Ising model, plotted for a variety of bond dimensions χ. For each bond dimension, the apparent correlation length of the ground state, ξ, is used to set the size of the block. The scaling behavior of these blocks as χ increases gives an alternative method for identifying the critical point, as well as allowing an estimate of the critical Binder cumulant ($B_c = 0.998$ and $U_4^* = 0.57$ for this data). 98

4.10 (Color online) The energy of the spin-1/2 transverse Ising model on a chain of length 10 is calculated using approximate ground states generated by the TEBD algorithm ($\chi = 20$). The numerical data (points) are plotted alongside the exact solution (line). Error bars are calculated from $\epsilon = \sqrt{\langle \Delta H^2 \rangle}$, with $\langle \Delta H^2 \rangle$ the second cumulant of the energies. (a) After 10 steps, the energies are still noisy and the error bars are quite large. (b) After 100 steps, the error bars have clearly decreased, and are largest for the points with the largest discrepancies from the exact solution. (c) By 1000 steps, most error bars are within the size of the data points, and the approximate energies are very close to the known analytical result. 99

4.11 (Color online) A Binder cumulant study of the spin-1 transverse Ising model. As above, the cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e. L = 10 and L = 15, and the BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.327(1)$. 101

4.12 Per-site value of the second cumulant of the longitudinal magnetization, $\frac{1}{L}\langle \Delta M_x^2 \rangle = \frac{1}{L}(\langle M_x^2 \rangle - \langle M_x \rangle^2)$, computed for the Spin-1 Ising model. The cumulant is calculated for an infinite system directly. 102

4.13 (Color online) The Binder cumulants for the spin-1 transverse Ising model, plotted for a variety of system sizes as a function of $L^{1/\nu}(B - B_c)$ for the known values $\nu = 1$ and $B_c = 1.326$. The length-independence of the curves allows this technique to be used as a means to estimate both ν and B_c 103

4.14 (Color online) A Binder cumulant study of the spin-1 transverse Ising model with crystal field. As above, the cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e. L = 10 and L = 15, and the BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.999(1)$. 104

4.15 Per-site value of the second cumulant of the longitudinal magnetization, $\frac{1}{L}\langle\Delta M_x^2\rangle = \frac{1}{L}(\langle M_x^2\rangle - \langle M_x\rangle^2)$, computed for the spin-1 Ising model with a transverse crystal field. The cumulant is calculated for an infinite system. 105

4.16 (Color online) The Binder cumulants for the spin-1 Ising model with crystal field, plotted for a variety of system sizes as a function of $L^{1/\nu}(B - B_c)$ for the known values $\nu = 1$ and $B_c = 2$. As expected, for these values the curves are seen to collapse to a functional form essentially independent of the length scale. 106

4.17 (Color online) A rough Binder cumulant study of the spin-1/2 transverse Ising model on a square lattice, using a local-update numerical algorithm with bond dimension $\chi = 2$. As in the one-dimensional case, the cumulants are computed for different system sizes across a range of values for the transverse field B. The largest crossing point, between $L = 10$ and $L = 12$, is at $B = 3.11(1)$. Extrapolating to the infinite limit gives $B_c = 3.3(2)$, though this cannot be done with high reliability on such a limited dataset (see text). 108

5.1 (Color Online). The phase diagram for a two-parameter Hamiltonian constructed to have an A_4 onsite symmetry group, as well as parity symmetry and one-site translation invariance. The symmetries of the Hamiltonian break down into five different residual symmetry groups in the ground states. These break down further when classified according to the relevant topological parameters, yielding eight distinct phases overall. The diversity of phases from the comparatively simple Hamiltonian shows the necessity of carefully accounting for all possible symmetries and topological parameters when attempting to characterize the phase of a ground state. For a description of the phases **A-H**, see discussions in the main text. 115

5.2 (Color Online). Free energy derivatives along the line $\mu = 2$ in the phase diagram above show the nature of the phase transitions. The continuous first derivative (blue) contrasts with divergence in the second derivative (red), showing a second-order transition. All three regions are topologically nontrivial SPT phases. Data shown here were computed with a bond dimension of 30, and the behavior has been seen to be stable as the bond dimension increases. 117

5.3 (Color Online). Entanglement entropy versus the log of the correlation length for states very close to the transition point. The slope is directly proportional to the central charge of the associated CFT, via Eq. 5.5. Data is generated by computing ground states at the point $\mu = 2, \lambda = 0.865$, and increasing the "bond dimension" of the numerical scheme to allow us to find states closer to the critical point where the correlation length diverges. The behavior shown here is representative of that seen elsewhere along the lines $\lambda = \pm 0.865$. Away from these lines, the entanglement entropy saturates at a finite value of ξ. The best-fit line has a slope of 0.225(1), which corresponds to a central charge of 1.35(1). 119

5.4 (Color Online). The notion of the transfer matrix can be generalized to include (a) onsite operation $U = \bigotimes_j u_j$, or (b) a parity operation $\mathcal{P}_\omega$. Generalizations to other symmetries are possible, but outside the scope of this work as they are not present in our model. 131

5.5 (Color online).The projective representation V of a symmetry can be obtained from a state's generalized transfer matrix because the dominant eigenvector of said matrix will be the vectorization of, V^{-1}, so long as the original state is in canonical form. This relation is demonstrated graphically for the case of an onsite symmetry, but easily generalizes to the parity case. 133

6.1 (Color online) Graphical notation showing the structures of the SIMPS algorithm, during the optimization step j of Eq. 6.16, where we are seeking a new state $|\phi_{n+1}\rangle = \tilde{H}|\phi_n\rangle$. (a) The matrix O_{eff}, which represents the optimization of the term $\langle\phi_{n+1}|O^{\dagger}O|\phi_{n+1}\rangle$ with respect to the tensor at site j. The operator O has been represented by an MPO with tensors M. Note that the virtual and physical indices are grouped in the same manner as in the matrix H_{eff} in traditional DMRG. (b) The vector representing the term $\langle\phi_n|O|\phi_{n+1}\rangle$ during the update of site j. Note the distinctions between the ket tensors $A^{(n)}_{[j]}$, which represent the state $|\phi_n\rangle$ from the previous iteration, and the tensors $A^{(n+1)}_{[j]}$ which represent the state being updated, $|\phi_{n+1}\rangle$. 155

6.2 (Color online) The total nearest-neighbor concurrence $\mathcal{C}^{tot}_{NN}$(triangles, left axis) and the normalized participation ratios (asterisks; right axis) for the random-field spin-1/2 Heisenberg model, shown for three system sizes. Data are collected using exact diagonalization; 1000 disordered configurations are computed, and for each we compute 50 eigenstates from the middle of the spectrum, where the behavior most resembles the thermodynamic limit. A sharp decrease in the NPR is known to show the transition to the localized phase (estimated for this model to occur around $W_c = 3.7$ [220]), and the concurrence is seen to increase significantly just as the NPR decreases. Note that the NPR values here have been normalized by the volume of the Hilbert spaces so that we can compare different system sizes. The crossing point in the concurrence curves is not considered to precisely identify the transition point, as it does not persist in any finite-size scaling relation. 162

6.3 (Color online) The total nearest-neighbor negativity $\mathcal{N}^{tot}_{NN}$(inverted triangles, left axis) and the normalized participation ratios (asterisks; right axis) for the random-field spin-1/2 Heisenberg model, shown for three system sizes. Data collection is identical to the concurrence results in Fig. 6.2. As with the concurrence, a sharp decrease in the NPR near the transition also coincides with a rapid increase in the negativity. As before, the NPR values here have been normalized by the volume of the Hilbert spaces, and the crossing point in the negativity curves is not likely of physical significance, as in the case of concurrence. . . 165

6.4 (Color online) L = 10 data for concurrence (triangles) and negativity (inverted triangles) are presented together, along with numerical fits from a polynomial of degree 9. The fit is in good agreement with the qualitative behavior of the data, particularly in the transition region. 166

6.5 (Color online) Total nearest-neighbor concurrence data is shown for various system sizes, in relation to the scaling function in Eq. 6.31. We use the values of $a \approx 0.5$, $b \approx 0.6$, and $W_c = 3.7$ which were identified in Ref. [65]. Clear evidence of scaling can be seen for these values, but not for other choices. The regions where the data "overshoot" one another. Data collapse is particularly clear when $L > 10$, since these systems have smaller finite-size effects. 167

6.6 (Color online) Total nearest-neighbor negativity data is shown for various system sizes in relation to the scaling function in Eq. 6.31. Comparing the result here to Fig. 6.5, we see that both the concurrence and the negativity show very similar scaling behavior, suggesting that both can be used to detect the localized to delocalized transition point. 168

6.7 (Color online) Using Data from SIMPS, we compute the concurrence and negativity between sites i and j in a disordered Heisenberg chain of length 20 in the localized regime ($W = 6$). Both quantities decay exponentially with the distance $d = |i-j|$ between the sites; note that here we have plotted the log of each (concurrence in black, negativity in red). Fit lines are included as a guide to the eye and to demonstrate that the rate of decay is very similar for both negativity and concurrence, suggesting that both quantities capture the same characteristic entanglement length. The slope of the concurrence fit line is $m_{\mathcal{C}} = -1.08$; for negativity it is $m_{\mathcal{N}} = -1.12$. This behavior is representative of other system lengths as well, although the slopes appear to have some weak dependence on the strength of disorder. Note that we show here only distances out to $d = L/2$; beyond this point the entanglements are so close to zero that they are very sensitive to noise and finite size effects, as discussed in Ref. [65]. 170

6.8 (Color online) Negativity and concurrence are compared for different disordered samples in an $L = 12$ system (data from exact diagonalization). Black lines indicate the analytic bounds on the relationship between concurrence and negativity for the two-qubit case; the negativity is never larger than the concurrence, and always larger than $\sqrt{(1-\mathcal{C})^2+\mathcal{C}^2}-(1-\mathcal{C})$. Qualitatively, we see that in subsystems from delocalized states (blue circles), both quantities are relatively small, and the lower bound is more likely to be saturated. In the case of strong disorder (red triangles) the full range of possible values for each is explored, with the upper bound more likely to be saturated. The intermediate case near the localized/delocalized transition shows behavior much more akin to the fully localized case, though this may be the result of finite size effects which tends to increase localized behavior in small systems. 171

6.9 (Color online) The ordering of concurrence and negativity is considered by comparing equivalent two-qubit subystems from pairs of different states in the disordered ensemble. The difference between concurrences is plotted versus the difference in negativities. Thus, data points in quadrants I and III represent "well-ordered" pairs, in which the state with the larger concurrence also has the larger negativity. The presence of data points in quadrants II and IV shows also that "ill-ordered" subsystems are possible, though not common. Data plotted here are from a system of $L = 20$ and $W = 6$, though the pattern is typical for states, at least in the localized regime. 173

Acknowledgements

If you wish to make an apple pie from scratch, you must first invent the universe.

— Carl Sagan

All of us arrive at our current places in life as the result of many factors beyond our control. In my case, my desire and ability to complete this degree depended upon the support and guidance of many pivotal people and organizations. I am thankful to everyone in my life who has played some role in helping me reach this point. I wish very much that it were possible to make an exhaustive list of such individuals, but I have been so fortunate in my friends and family over the years that these pages could scarcely contain the list of all those who have impacted me positively. Indeed, it is likely there are many to whom I owe thanks without even knowing it. But at a minimum, I wish to acknowledge and thank individuals whose influence has been most on my mind during the past months as I prepared this dissertation.

Professionally, of course, I am indebted most of all to my advisor, Tzu-Chieh Wei, for guiding me through the process of developing strong research projects and supporting my development as a scientist. I am further grateful to the other graduate students and postdocs with whom I collaborated and discussed this work, notably Ching-Yu Huang, for always helping me understand the finer points of numerical methods; Abhishodh Prakash, for identifying excellent research problems and for keeping my numerical explorations grounded in theoretical rigor; and Artur Garcia-Saez, for both providing essential training in the art of computer programming and for providing invaluable insights and encouragement during my first forays into graduate research. I'm grateful also to the National Science Foundation for financial support, as well as to the Alan Alda Center for Communicating Science, which has taught me that good communication skills are not only essential to interfacing with the public, but also to participating effectively in today's increasingly interdisciplinary research community.

Personally, I am grateful for a long, unbroken chain of mentors and supporters who guided me to this point in my life. First and foremost, I thank my parents, Paul and Lisa, who expertly nurtured my curiosity and thirst for knowledge from my very first days, and who have both served as extraordinary role models to me in all aspects of life. I'm thankful for all of my teachers, but especially for Shiela Ferguson, for encouraging my earliest "academic research" interests; for Joel Peterson and the WAVE program, for showing me the thrill of hands-on, collaborative science projects; and for Paul DeMaret, for developing my self-confidence and reminding me that in an era of high specialization, there is still much value in a strong foundation of general knowledge.

Last but not least, I am incredibly indebted to my wife, Alison Mae West, who has sacrificed so many things and selflessly given so much to ensure that I could complete this work. It is no exaggeration to say that she has been my biggest source of sanity and support the past five years. One could not ask for a greater partner with whom to navigate the different phases and transitions of life, and I am honored and humbled that my life's story has become entangled with hers.

Chapter 1

Introduction

There is, perhaps, a common misconception that physicists simply want to know what makes up the universe, by finding the most fundamental "building blocks." To be sure, this desire to find the smallest, indivisible constituents of matter is certainly *one* of driving forces behind advancements in physics. It has been from the days when ancient Greeks contemplated the uncuttable "$\alpha\tau\mu o\nu$" to the days of the true atomic age of the mid-twentieth century [1]; it was partly this desire which brought us the quark model and the revolution of high-energy particle physics, and it continues to this day as a motivation for the study of string theory [2]. But while identifying the building blocks of the universe is a necessary part of understanding its physics, this alone is not sufficient.

An equally or perhaps even more important goal of physics is to understand how these building blocks interact and interrelate. It is only from this, an understanding of how particles in the universe affect one another, that we might hope to achieve a complete understanding of their behavior, and from this that we might hope to harness their behavior to produce new and better technologies.

To this end, the field of quantum many-body physics has alway sought to explain the incredible complexity which can arise when particles affect one another at a quantum scale. The unique set of relationships that can exist between particles in quantum physics is in some ways its most remarkable property, and perhaps ultimately its most useful. In particular, the uniquely quantum notion of *entanglement* [3, 4] has not only played a vital role in our conceptual understanding of the universe [5, 6], but it may also hold the key to tremendously powerful new technologies in fields like efficient communication [7], cryptography [8], and computing itself [9, 10].

But the power of interaction and entanglement comes at a price. When every particle in a system can potentially be entangled with every other particle, it can take an enormous amount of information to describe their state unless the number of particles involved is very small. In an age where progress in science increasingly relies upon computer simulations and computer-aided data processing, we might hope that our PCs and laptops could come to our aid. But for a spin-1/2 system, a description of the state of a mere 30 particles will require so many numbers to specify that it can scarcely be held in the memory of an everyday hard drive. And this is only what is required to describe the state itself; if one wishes to work with the Hamiltonian operator that governs the behavior of the system, it will require nearly a trillion such hard drives to store it. As Laughlin put it [11], "No computer existing, or that will ever exist, can break this barrier because it is a catastrophe of dimension."

There may however be a loophole in this catastrophe. Nature, it appears, is fundamentally local, and the "typical" states that we might most easily expect to encounter or construct in a laboratory (namely, low-energy eigenstates of gapped, local Hamiltonians) tend not to be quite as hopelessly entangled as one might expect of a more generic quantum state. Instead, these states obey a so-called "area law" [12] for the entanglement of their subsystems, which limits the amount of entanglement that has to be considered, even when the system becomes large in size [13].

In recent years, a new set of numerical methods has emerged for representing quantum systems and simulating them with computers [14–16]. These techniques describe the quantum states with a "tensor network" structure, which naturally embodies the entanglement area law property and can dramatically simplify the amount of information required to describe such states.

Initially, these methods were introduced in the context of one-dimensional systems, where they are referred to as "matrix product states" (MPS) [17–20]. Soon after, the methods rose to even greater prominence when it was realized that an important computational technique for studying quantum systems, the celebrated "Density Matrix Renormalization Group" (DMRG) method [21, 22] could be reformulated (and even expanded) entirely in terms of tensor network methods [23–26]. Additional algorithms based on MPS soon followed, such as the"Time Evolving Block Decimation" (TEBD) technique [27, 28], which is important both for calculating the ground state wavefunction of a given Hamiltonian, and for simulating the time evolution of such a state. Since then, tensor network methods have also been readily applied to gapless systems by means of the "Multiscale Entanglement Renormalization Anzatz" (MERA) [29, 30], as well as generalized to higher-dimensional systems, where they are termed "tensor network states," or "projected-entangled pair states" (PEPS)[31–37].

In this dissertation we will be examining closely the class of tensor network algorithms, considering the best methods for putting them into practice, and exploring new problems to which they can be applied. We begin with a review of the current state of such methods. In Chapter 2, we first establish some of the theoretical and mathematical details behind tensor networks, and give an explanation as to why they are so efficient at representing certain quantum states. In Chapter 3, we then focus specifically on two classes of algorithms that we have implemented numerically and used to generate the data in the remaining sections of this work. Specifically, we describe our implementation of DMRG for finite systems, and then discuss the application of TEBD to both the finite and infinite states. In the latter case, we also describe some generalizations, and perform a study of how to set some of the tunable numerical parameters to optimize the algorithm's efficiency.

In the remaining chapters, we then employ these MPS techniques to study three different types of phenomena that have become focal points of study in quantum many-body physics: quantum phase transitions, symmetry-protected topological phases, and many-body localization.

Our work in Chapter 4 begins with the "traditional" view of quantum phases and phase transitions [38–40]: we consider phases characterized by different instances of spontaneous symmetry breaking, and ask about the order parameters which distinguish them. Matrix product states have can be used to efficiently compute various observables in an effort to detect these transitions and calculate the location of the critical points, a central problem in many-body physics [19, 41–44]. In our work, we look beyond the expectation value of an operator and ask instead about the higher-order moments and cumulants. We introduce a method that allows these cumulants to be evaluated efficiently using tensor networks. Then, we show how this allows the so-called "Binder Cumulant" [45], a technique from the world of quantum Monte Carlo, to be imported into the world of matrix product states. By applying the Binder Cumulant and other moment/cumulant techniques to a variety of different spin systems, we show how the locations of the quantum phase transitions can be computed to relatively high precision at low computational cost.

In Chapter 5, we expand our understanding of phase transitions in quantum systems and consider in particular the phenomenon of symmetry-protected topological order [46–48], a particular type of quantum phase classification which can emerge even when comparing states which share a symmetry group. The possibility of phases labelled by spontaneous symmetry breaking, symmetry-protected topological order, or both, gives rise to an enormous diversity of phases which can exist in one dimensional systems. A complete set of classification parameters sufficient to describe all the possible phases has been established [49–53], and it has also been shown that many of these parameters can be computed directly from the MPS representation of a state [52]. We give the necessary methods for extracting *all* such parameters, and demonstrate them on a model Hamiltonian which we have constructed to possess an onsite A_4 symmetry, in addition to invariance under lattice translations and inversions. We give a phase diagram for this Hamiltonian and identify eight different quantum phases within it, including an interesting example of two nontrivial SPT phases with a second-order phase transition between them.

Finally, in Chapter 6 we turn from the question of static phase transitions in quantum ground states to the phenomenon of "many body localization," [54–58] where in fact a different type of *dynamical* quantum phase transition can exist in the excited states of a disordered Hamiltonian. Depending on the nature of the disorder (and the dimensionality of the system), the excited states may fail to thermalize, even if the system contains interactions. The critical strength of disorder beyond which this occurs then serves as a kind of phase boundary between the localized and delocalized regimes. Naturally, we discuss how MPS can be used to study systems of this type [59–62], and also consider two measures of local entanglement, the concurrence [63] and negativity [64]) as possible indicators for locating this phase boundary (the former of which was also recently considered in Ref. [65]). We show that indeed both measures show distinctive behavior in the localized case as compared to a delocalized, thermal system. Moreover, we give numerical evidence that the two measures, though analytically distinct [66] appear quite similar in terms of their ability to detect many-body localization. This may be important, since the negativity is a more flexible measure and can be applied to study a wider variety of systems.

To conclude, our results and findings are summarized in Chapter 7. We discuss briefly some ongoing projects that build further on this work, and identify other interesting questions which remain open as directions for future study.

Chapter 2

Tensor Network States

In this chapter, we begin by presenting some basic concepts and clarifying some notation and terminology with respect to tensors and tensor networks as used in this work (Sec. 2.1). We shall then briefly introduce the motivation for tensor network states in Sec. 2.2, and review some of their features as relevant to this work with a particular focus on matrix product states. The formalism of these states will form the backbone for all the techniques and algorithms applied in subsequent chapters.

2.1 Tensors and Tensor Networks

To begin, we state some very basic principles about tensors which will be used extensively, in order to fully clarify our notation and define terminology. In the context of this work, a rank-n "tensor" is simply an n-dimensional array. We do not distinguish between, say, tensors and pseudotensors, and the use of the term implies nothing about transformation laws, as might necessary when viewing a tensor as a basis-independent multilinear map. We will use both upper and lower indices to label tensors without implying or distinguishing the notions of covariance and contravariance. We may explicitly make efforts to distinguish types of index in other ways, particularly in the case of a simple matrix, in which case we will call the indices "incoming" an "outgoing" when necessary. We do, however, make use of the Einstein summation convention in which any repeated indices are implicitly summed over, except where an explicit sum is necessary for clarity.

Many times, the index of a matrix or tensor will describe the elements of some composite space; in these situations, we have a choice between separately associating an index with each subspace, or assigning a larger index to the whole space. For example consider a space $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$. Vectors which live in this space can be written in terms of basis elements $|\tau = \{1,2,3,4\}\rangle$, or in terms of elements $|\alpha = \{1,2\}\rangle \otimes |\beta = \{1,2\}\rangle$. These labelings are in general quite artificial (unless a certain symmetry is being imposed), and can be directly equated through a standard mapping, e.g. $\tau = 1 \to (\alpha = 1, \beta = 1)$ etc. Accordingly we call the index τ a "composite" index, and instead of using the name "τ," as in the vector X_τ, we may write $X_{(\alpha\beta)}$ to emphasize the composite nature of the index.

Sometimes, it will be helpful to explicitly consider the constituent indices within such a composite index. For example, an operation which acts with operation Z in subspace A and trivially in the B subspace could be written as $M = Z_A \otimes \mathbb{1}_B$, and applied to the vector X as $M_{(\alpha\beta)}^{(\alpha'\beta')} X_{(\alpha\beta)}$. But alternatively, we could consider "ungrouping" the composite index of X and consider $X_{\alpha,\beta}$. From this point of view, the operation above can be expressed as $(Z_A)_\alpha X_{\alpha,\beta}$.

The reverse process is is also often useful, in which we "group" the incoming and outgoing indices of a matrix $V_{\alpha,\beta}$ into a single composite index $(\alpha\beta)$, so that the matrix can now be viewed as a vector. We call this operation the "vectorization" of V. Note that grouping or ungrouping of a set of indices does not mathematically change the tensor; the same numbers are stored, but simply indexed with slightly different labels. The choice of grouping is purely to aid in our understanding or to allow for more compact notation.

2.1.1 Graphical Tensor Notation

In this work, we will make use of the so-called "graphical notation" for tensors. This notation, often attributed to Penrose (see for example Ref. [67]) has been used widely to simplify the expression of complex tensor systems, with applications far outside the field of tensor network algorithms. Even here, it includes many useful subtleties; for a general overview we refer the reader to the discussion in Ref. [68]. In our context, however, the following points are by far the most relevant:

- *A tensor is represented by a solid shape.* The nature of the shape is not mathematically relevant and more akin to a "variable name"– i.e., we may choose to represent all tensors which serve a particular function as circles, just as we might give them a similar set of names such as $\{A_n\}$. But the shape does not say anything directly about the mathematical structure of the tensor.

- *A line extending from a tensor is an index of the tensor.* Thus, a rank-n tensor should have n lines extending from it. Where appropriate and necessary (i.e., to distinguish bras and kets), the incoming and outgoing indices of a matrix will be represented by lines on opposite sides of the tensor, either left/right or top/bottom, depending on the context. Other than this, the "shape" of the lines also does not carry mathematical significance; where necessary, lines in our diagrams may be curved for visual convenience.

- *two tensors next two each other but not connected by a line are combined by the tensor product.* This is a visual analogy to placing two variables next to one another algebraically to imply scalar multiplication

- *A connected line between two tensors represents a contraction.* This is the visual equivalent of the Einstein summation convention; when two index lines are connected, they are shown to be "the same," just as under the Einstein convention, in which two indices of the same name are assumed to be "contracted," i.e.

$$C^{\alpha,\beta} = A^{\alpha,\beta,\gamma} B^{\gamma} \tag{2.1}$$

- *Composite indices can be graphically "grouped" and "ungrouped".* When necessary for clarity, we will explicitly show grouping in our diagrams, either by bending indices together, labelling them with a bracket, or both.

For examples of these conventions in graphical notation, see Fig. 2.1.

Note that in this notation that the identity matrix $\mathbb{1}_{i,j} = \delta_{i,j}$ can simply represented by a line. Similarly, tracing over a pair of indices, i.e. constructing $X_\beta = Z^{\alpha}_{\alpha,\beta}$ is represented graphically by simply connecting the two indices with a line (Fig. 2.2).

Naturally, the phrase "tensor networks" contemplates that we will be contracting many tensors together, potentially in elaborate ways. In these cases, the order in which the contractions are performed can be very important. Although finding the optimal sequence of contractions is generally a hard problem subject to its own study (see for example [69]), at a minimum one can often identify suboptimal choices by considering the size of the intermediate tensors. Clearly, the memory impact of a tensor is measured by the product of the dimensions of its indices. Hence, when choosing the order in which to contract the tensors in a network, one should "look ahead" to foresee the largest total size of the open indices which occurs during the intermediate steps. Keeping this size small can dramatically change the performance of an algorithm based on tensor contractions. For more comments on implementing these contractions in practice, we also refer readers to the Matlab-based tensor contraction

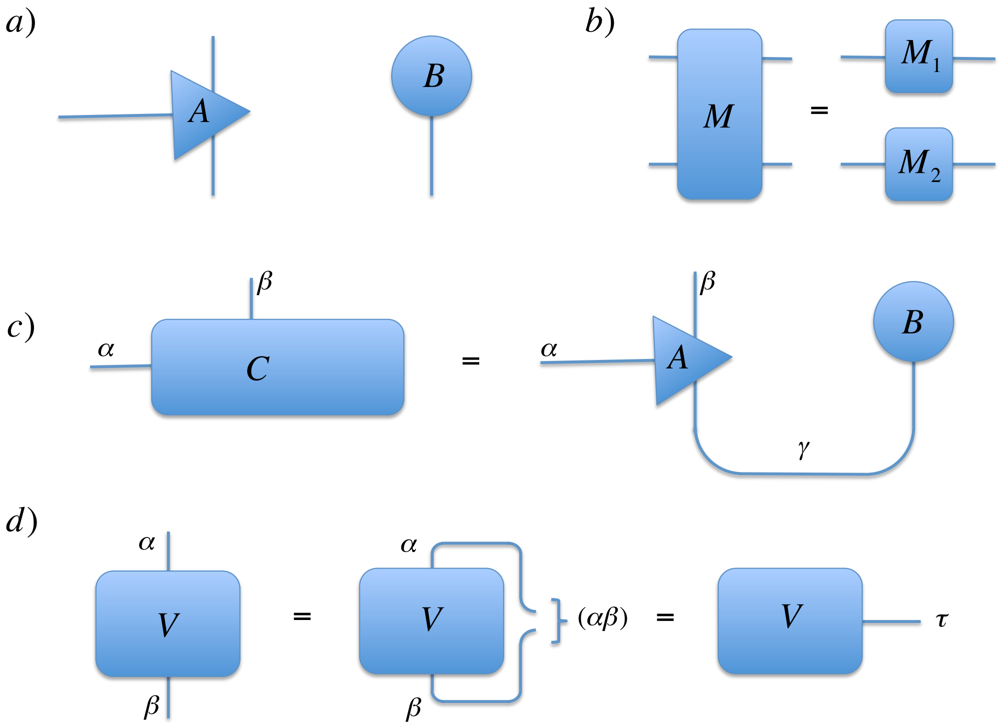

Figure 2.1: (Color Online). Examples of the graphical notation for tensor operations. (a) Examples of tensor. A solid shape represents a tensor, with extended lined representing indices. Here evidently A is a rank-three tensor, and B is rank-one, i.e. it is a vector. (b) Adjacent tensors with imply a tensor product. This diagram represents the equation $M = M_1 \otimes M_2$. (c) Tensor contraction is represented by connecting indices. The equation $C^{\alpha,\beta} = A^{\alpha,\beta,\gamma} B^{\gamma}$ is represented here. (d) By changing our labelling convention, the indices of a tensor can be "grouped" or ungrouped. In graphical notation, this can be represented by literally grouping the lines. Pictured here is the vectorization of a matrix: $V_{\alpha,\beta} \rightarrow V_{(\alpha\beta)}$

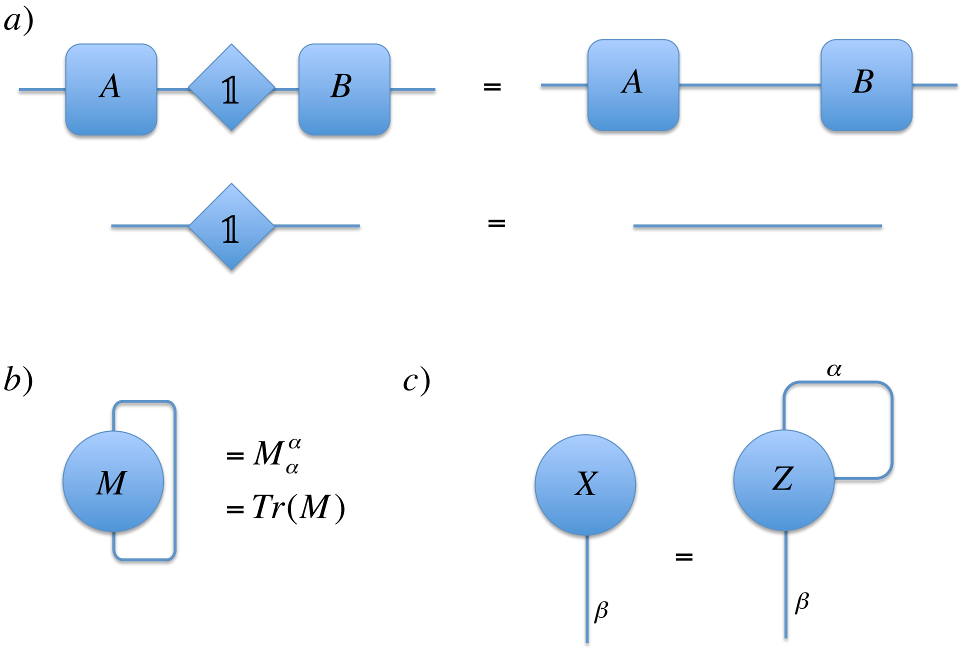

Figure 2.2: (Color Online). Additional examples of graphical tensor notation. (a) Clearly, since $A\mathbb{1}B = AB$, then in our graphical notation we can simply represent the identity matrix with a line. (b) Tracing over a matrix is shown by connecting its two indices, contracting it with itself. (c) Tracing in this sense can happen over any pair of indices originating from the same tensor. Pictured here is the equation $X_{\beta} = Z^{\alpha}_{\alpha,\beta}$. This is related to the notion of a "partial trace," which we shall also discuss in Chapter 6.

subroutine “NCON,” [70] which can perform sequential tensor contractions in a robust manner, and also provides warnings in many situations where the specified sequence is inefficient.

2.2 Tensor Network States: Structure and Motivation

We now consider the purposes and properties of matrix product states, with a particular focus on those properties relevant to our subsequent work. For a fully comprehensive review and additional technical details, see Refs. [14, 15, 71]. For a more pedagogically-motivated introduction to tensor networks states and their applications, see Refs. [16, 72].

Before the arrival of tensor network states, numerical simulation of quantum many-body systems was faced with a seemingly insurmountable barrier: the so-called “curse of dimensionality” with respect to the Hilbert space. Consider for example a simple spin system consisting of N different spin-s particles. Each particle’s spin degrees of freedom live in a Hilbert space local dimension $d = 2s + 1$; hence, to capture the spin degrees of freedom for the entire system (to say nothing of any other necessary quantum numbers) we will require a space of dimension d^N. Thanks to the remarkable power of the exponential function, this scaling becomes physically unworkable almost impossibly quickly.

More tangibly, consider the state vector for a set of L particles on a one-dimensional lattice, and expand the vector in some orthonormal basis. For compactness, the expansion coefficients can be collected up into a tensor $c_{s_1 \ldots s_L}$, with L indices, each of size d–one for each of the local Hilbert spaces (we shall call these the “physical indices,” for reasons which will become clear later). In other words, we have

$$|\psi\rangle = \sum_{s_1=1}^{d} \ldots \sum_{s_L=1}^{d} c_{s_1 \ldots s_L} |s_1 \ldots s_L\rangle \tag{2.2}$$

The problem with representing such a state in a computer is the size of the coefficient tensor $c_{s_1 \ldots s_L}$: this is the structure which must store d^L numbers to represent an arbitrary state. But one may ask if there is some way we can condense this tensor to reduce the amount of information which must be stored. After all, “data compression” is a familiar method for dealing with the need to store large amounts of numerical data. And indeed, there is a loose sense in which the answer is “yes.” At least for certain classes of quantum states, the massive tensor $c_{s_1 \ldots s_L}$ can be restructured and simplified through a controlled approximation scheme, so that nearly the same coefficients can be represented by far fewer numbers.

2.2.1 Truncated Singular Value Decompositions

To understand how this restructuring can be done, consider the Schmidt decomposition of a vector [73, 74]. In particular, imagine a (pure) state $|\psi\rangle$ living in a Hilbert space $\mathcal{H}$ which is a tensor product of two subspaces, $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$. The Schmidt decomposition states that, for some choice of orthonormal bases for these subspaces, $\{|i\rangle_A\}$ and $\{|j\rangle_B\}$, we can decompose $|\psi\rangle$ as

$$|\psi\rangle = \sum_i p_i |i\rangle_A |i\rangle_B. \tag{2.3}$$

The coefficients p_i are called the "Schmidt Coefficients," and for a normalized state $|\psi\rangle$, they clearly must satisfy $\sum_i p_i^2 = 1$. The number of nonzero coefficients required to form this decomposition is called the "Schmidt rank" of the system, and can be taken as a rough indicator of the entanglement between the subsystems $\mathcal{H}_A$ and $\mathcal{H}_B$. A product state, which by definition can be written as $|\psi_{prod}\rangle = |\psi_A\rangle|\psi_B\rangle$ is a state with Schmidt rank 1.

The Schmidt coefficients p_i are deeply connected to the singular value decomposition (SVD), a standard linear algebra technique generalizing the eigenvalue decomposition which likely requires no introduction (though we wish to call the reader's attention to Ref. [75], which gives an intriguing history of the technique, and to Ref. [76], the original statement of the fully general matrix SVD by Eckart and Young.) The SVD decomposes an $m \times n$ matrix M into the form $M = U\Lambda V^\dagger$, with U an $m \times m$ unitary matrix, V an $n \times n$ unitary matrix (n.b. the historically motivated convention of the Hermitian conjugate), and Λ an $m \times n$ rectangular diagonal matrix with non-negative entries λ_i. These entries of Λ are the titular "singular values" of the original matrix M.

What is special about Eq. 2.3 is the fact that there is only a single index which must be summed over; the same coefficient can be applied to the basis states in each subspace when chosen correctly, with no "cross terms" necessary in the expansion. Naturally, for a more generic choice of orthonormal bases $|\tilde{i}\rangle_A$ and $|\tilde{j}\rangle_B$, our state $|\psi\rangle$ could have been more trivially expanded in the two subspaces as

$$|\psi\rangle = \sum_i \sum_j c_{ij} |\tilde{i}\rangle_A |\tilde{j}\rangle_B, \tag{2.4}$$

But here, the expansion coefficients depend on two indices, and as such can be viewed as a matrix c_{ij}. To recover the Schmidt decomposition, we must perform an SVD on this matrix, $c_{ij} = U\Lambda V^{\dagger}$. U and $V^{\dagger}$ can then be absorbed into the choice of bases, so that the "special" Schmidt bases of Eq. 2.3 are given by

$$|i\rangle_A = \sum_{i'} U_{i,i'}|i'\rangle_A |i\rangle_B = \sum_{i'} V^{\dagger}_{i,i'}|i'\rangle_B \tag{2.5}$$

Thus, the singular value decomposition gives us both the correct choice of Schmidt bases, and also the Schmidt coefficients themselves, as evidently $p_i = \lambda_i$, the singular values of $c_{i,j}$ (henceforth we shall simply refer to them with the notation λ_i).

This refines our understanding of the decomposition allows us to make two observations which will be of great significance. First, we can now see the connection to a much more nuanced indicator of entanglement: instead of simply counting the number of nonzero coefficients, we can ask about the relative sizes of such coefficients. After all, a state with two Schmidt coefficients essentially equal in magnitude is much further from a product state than a state whose two Schmidt coefficients are given by $\lambda_1 = \epsilon$ and $\lambda_2 = \sqrt{1-\epsilon^2}$ for some ϵ which is nonzero but vanishingly small. To give proper consideration to the relative size these coefficients, we want a measure of entanglement which grows as the Schmidt coefficients become more mutually comparable in size. This notion recalls precisely the idea of entropy, and in fact, if we consider the density matrix $\rho = |\psi\rangle\langle\psi|$ and the von Neumann entropy $S(\rho) = tr(\rho \log \rho)$, we can see (by substituting the Schmidt decomposition of $|\psi\rangle$) that the entropy is given directly from the singular values λ_i, with $S = -\sum_i \lambda_i^2 \log \lambda_i^2$. This is the so-called "entanglement entropy" between the subsystems $\mathcal{H}_A$ and $\mathcal{H}_B$.

This notion also allows us to imagine a way in which the degrees of freedom required to represent the state ψ could be reduced, furthering our overall goal of being able to represent a many-body system in a manner not exponentially large relative to the system size. If we return to the case contemplated above, where a state's Schmidt coefficients are given by $\lambda_1 = \epsilon$ and $\lambda_2 = \sqrt{1-\epsilon^2}$, we can see informally that such state is, to a good approximation, really just a product state. If we chose to neglect the smallest coefficient, it seems likely that little would change about our understanding of the state, at least up to order ϵ. This idea was explored even in the early work on the singular value decomposition, which considered the extent to which a matrix changes when one performs an SVD, and "truncates" the n smallest singular values, i.e., these values are set to zero, and the corresponding rows/columns of U and $V^{\dagger}$ are therefore ignored [76, 77]. The resulting "approximation" of the original matrix, which has smaller rank, has been shown to be "ideal," in the sense that it maximizes the fidelity with the original matrix relative to all other matrices of the same dimensionality [15, 77, 78]. For our purposes, if we recall that the "matrix" in question can be the matrix of coefficients representing a bipartition of the state, this means that we can use the SVD as an optimal tool to eliminate degrees of freedom without changing the state too significantly.

Let us return our focus now to the original goal, of taking the coefficient tensor $c_{s_1 \ldots s_L}$ from Eq. 2.2 and simplifying its structure, so that the associated state can be represented more compactly. This tensor is just a larger generalization of the matrix $c_{i,j}$, which we simplified through the truncated SVD above. Hence, at least for the class of states whose Schmidt decompositions produce many small, truncatable values, the tensor of coefficients can be dramatically simplified by successively applying the truncated SVD operation to split $c_{s_1 \ldots s_L}$ into smaller and smaller pieces. For example, we can start by considering separately only the first index s_1, and grouping the remaining indices into some composite index $S_{2 \ldots L}$. Viewed in this fashion, the coefficient tensor is simply a matrix (admittedly, one with highly unbalanced dimensions) and we can break it apart by SVD. This gives

$$\begin{aligned} c_{s_1,(s_2 \ldots s_L)} &= U_{s_1} \Lambda V^{\dagger}_{S_{2 \ldots L}} && (2.6) \\ &= A_{s_1} \tilde{c}_{s_2 \ldots s_L} && (2.7) \end{aligned}$$

where in the second line, we have simply defined $A_{s_1} = U_{s_1}$ and $\tilde{c}_{s_2 \ldots s_L} = \Lambda V^{\dagger}_{S_{2 \ldots L}}$, in the latter case also ungrouping the index $s_2 \ldots s_L$. And if, in particular, we *truncate* the SVD in Eq. 2.6 by discarding any small singular values, then we will have reduced the total amount of information without substantially changing the coefficients represented.

The process above, also demonstrated graphically in Fig. 2.3 can be iteratively applied to the "remaining" portion of the coefficient tensor, $\tilde{c}_{s_2 \ldots s_L}$, continuing to split off pieces of the tensor for each of the physical indices (i.e., for each particle in the system). If we are fortunate enough to have many small, truncatable singular values along the way, we can potentially realize a dramatic reduction in the size of the tensor needed to specify the state coefficients. In the end, instead of representing the coefficients with a massive, single tensor, we have instead employed a "network" of smaller tensors, all contracted together, which collectively specify the state. A state expressible in this manner is thus generally called a "tensor network state."

2.2.2 Matrix Product States

Thus, we have seen a possible method by which *some* quantum states could be represented with a much smaller amount of information, by discarding

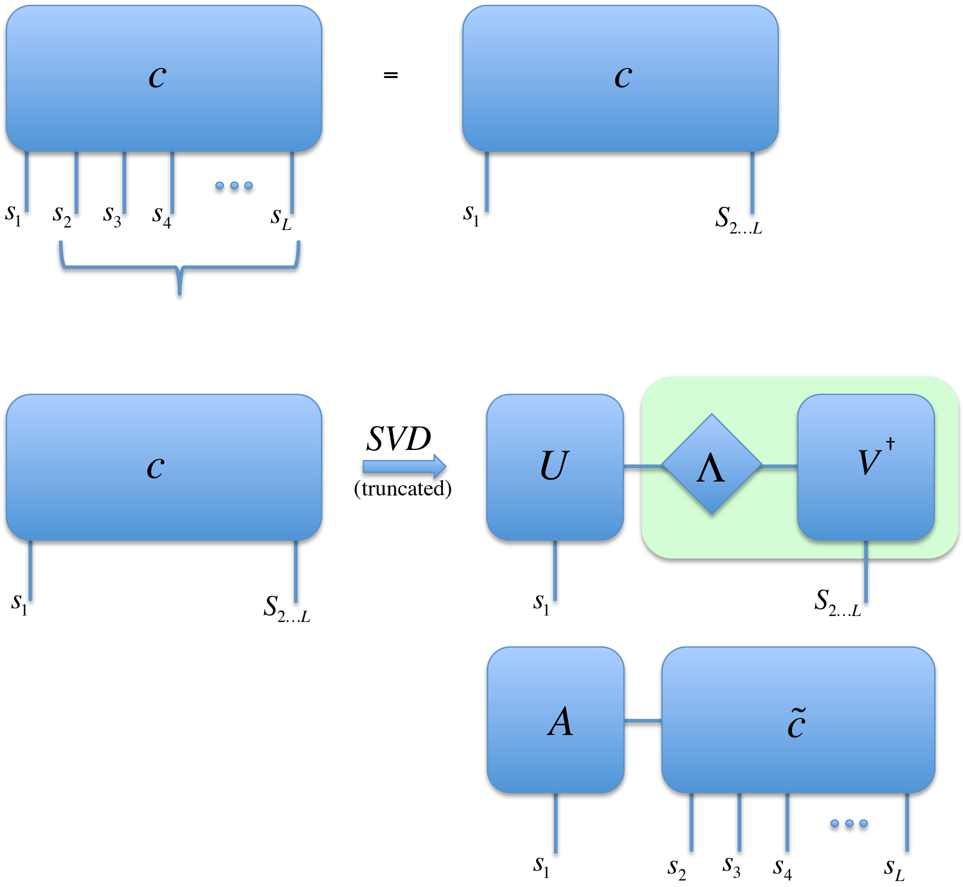

Figure 2.3: (Color Online). The coefficients for a one-dimensional spin system of length L could be represented by an enormous tensor $c_{s_1 \ldots s_L}$. However, we may be able to iteratively reduce the overall size of the tensor structure needed to specify the state by means of a truncated singular value decomposition. First, we conceptually group all of the indices except that for the first site, leaving a rectangular matrix. This matrix can then be decomposed and simplified by truncated SVD. The matrices Λ and $V^{\dagger}_{S_{2\ldots L}}$ collectively serve as a large coefficient tensor for all of the state but the first site. This tensor can in turn be further decomposed by the same process.

information about the Schmidt coefficients when those coefficients are very small or zero. But one might still ask how useful this trick is. What hope do we have that a physically *interesting* state could have its coefficients efficiently represented in this way? Fortunately, if we consider what is happening during the truncated SVD process, we can see reason to suspect that many states of interest might admit this kind of representation. At each step, we were looking at two neighboring particles and discarding some of the information about the entanglement (measured by entanglement entropy) between them. Hence, and "slightly entangled" states, without long-range correlations reaching across the system, are promising candidates.

Thankfully, these properties can all be found a familiar and useful family of states: the ground states (and low energy excitations) of gapped Hamiltonians with purely local interactions. The entanglement and correlation properties of such states give rise to exponentially decaying Schmidt coefficients along bifurcations of the systems[13, 15, 27]. In particular, such states obey an "area law" for entanglement scaling: if we choose some subsystem within the state and ask about its entanglement with the rest of the system, we will find that it depends upon the size of the *boundary* of the subsystem, not the *volume* of the subsystem [12, 79]. Thus, for such cases, we should always be able to find small values in the tails of the singular value distribution which can be safely truncated without dramatically altering the state, allowing us to perform the tensor decomposition described above. As a result, the ground states of gapped, local Hamiltonians can be efficiently represented by one dimensional tensor network systems [13, 15, 16, 80]. But such states are also among the most physically relevant, as both laboratory experiments and natural phenomena often arise from local interactions, and much of the typical physics occurs in the low-lying energy levels.

Unsurprisingly, then, this is precisely the context in which tensor network methods were first successfully applied to the representation of quantum states[17–20]. For one-dimensional systems, these representations are called "matrix product states," [14, 15] or simply "MPS."

In a chain of length L with open boundary conditions, an MPS has the form

$$|\psi\rangle = \sum_{s} A^{s_1}_{[1]} A^{s_2}_{[2]} \dots A^{s_L}_{[L]} |s_1 s_2 \dots s_L\rangle. \tag{2.8}$$

This structure, which is perhaps made clearer in the graphical notation of Fig. 2.4, is essentially what results if one takes a generic coefficient tensor of Eq. 2.2 and decomposes it site-by-site through a sequence of truncated SVDs, as described above. We have changed notation slightly compared to Eq. 2.6.

The indices of the "A" tensors are now being written as superscripts, but recall that in our context this is not meant to convey anything physical; rather, it is done to make room for the bracketed labels"$[j]$." These do not label an index, but rather are part of the "name" of a tensor, typically indicating the lattice site with which it is associated. As such, $A^{s_j}_{[j]}$ and $A^{s_k}_{[k]}$ represent in principle two different tensors. We could perhaps have called them A^{s_j} and B^{s_k}, but we prefer the flexibility and precision of the bracketed notation, as there will be times in subsequent discussions when the labelled index may not match the bracketed label, resulting in a tensor such as $A^{s_j}_{[j+1]}$.

We have said that the $A_{[j]}$ tensors all possess three indices. Of these, one index has special significance. The labelled indices s_1, etc., correspond to the indices of the original coefficient tensor, and are also the indices which must match the labels in the basis vectors $|s_1 s_2 \dots s_L\rangle$. We call these the "physical indices" of the tensors.

The other two indices of any given A tensor are often suppressed in the MPS notation, because for any fixed value of s_j the two remaining virtual indices simply represent a matrix. For this reason, the labels of the remaining indices are often suppressed, because the contractions of these indices occur in the same manner as ordinary matrix multiplication. Nevertheless, although they are rarely labelled explicitly, the two "non-physical" indices, typically termed the "virtual" indices (sometimes also "bond" or "auxilliary indices") are extremely important. The dimension of these indices, called the "bond dimension" and often represented as "χ," controls the amount of entanglement and the lengths of correlation which can be encoded by the MPS. For example, if our MPS is constructed through the truncated SVD process described above, then χ is a value of our choosing: it represents the number of singular values which we chose to leave in place after the truncation. χ can thus also be seen as a numerical parameter, controlling the representative power of the MPS. Smaller values of χ are of course less numerically expensive, but larger values can allow the MPS to more accurately represent the features of the state, particularly in systems with longer correlation lengths. When necessary, the value of χ can also vary from matrix to matrix, although in such cases we may still refer to the largest of these values as the "bond dimension" of the state.

Note now that the MPS structure can be more general and flexible than simply that given in Eq. 2.8. For example, A small adjustment will also allow us to consider systems with periodic boundary conditions, by simply tracing over the product of $A_{[j]}$ tensors so that the outgoing virtual index of the final tensor $A_{[L]}$ connects to the incoming virtual index of $A_{[1]}$, i.e.

$$|\psi\rangle = \sum_s Tr\left(A^{s_1}_{[1]} A^{s_2}_{[2]} \dots A^{s_L}_{[L]}\right) |s_1 s_2 \dots s_L\rangle. \tag{2.9}$$

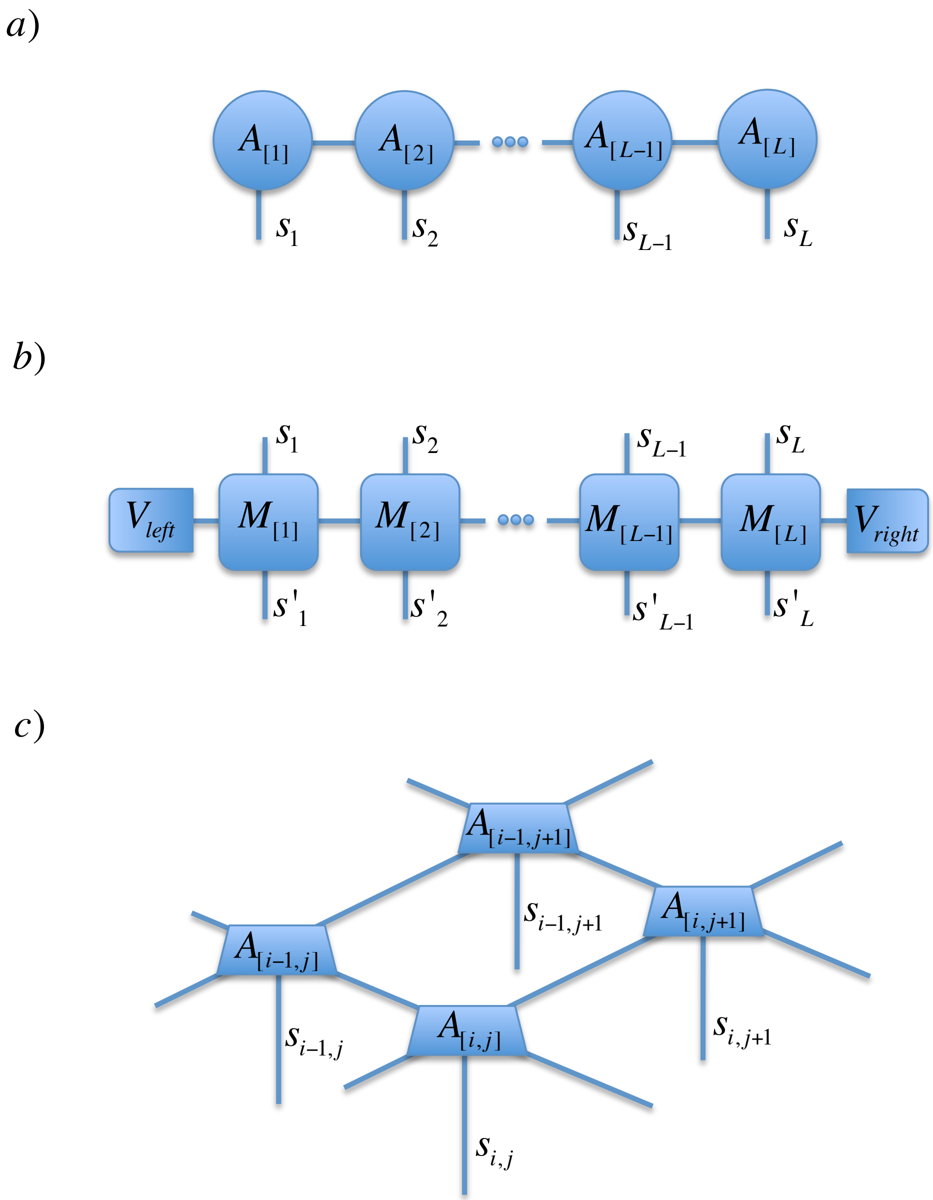

Figure 2.4: (Color Online). The structures of common tensor network systems. (a) A matrix product state (MPS) for a system of length L with open boundary conditions, as in Eq. 2.8. The physical indices which relate to the physical degrees of freedom are explicitly labelled. The indices connecting the $A_{[i]}$ tensors to one another are called "virtual" indices. (b) A matrix product *operator*, as in Eq. 2.19. The left and right boundary vectors are explicitly indicated to clarify the structure, although they could have been subsumed by the definitions of $M_{[1]}$ and $M_{[L]}$. (c) A two dimensional tensor network state or "PEPS", for the particular case of a square lattice.

Perhaps more importantly, the MPS structure also extends naturally to certain infinite systems (where they are sometimes called "iMPS"). This can be done for systems with some form of translation invariance, so that the properties of the system are completely captured by describing only a representative unit cell. For example, consider an infinite, one-dimensional periodic system possessing translation invariance with respect to a unit cell of length ℓ. Represented as a matrix product, the state is of the form given by Eq. (2.8), but with the further restriction that not all tensors $A^{j}_{[i]}$ are distinct, and instead repeat every ℓ sites. A state with only a one-site unit cell (full translation invariance) can therefore be specified by only a single tensor A, i.e.

$$|\psi\rangle = \sum_{s} Tr\left(A^{s_1} A^{s_2} \ldots\right) |s_1 s_2 \ldots\rangle. \tag{2.10}$$

Similarly, a state with two-site translation invariance ($\ell = 2$) specified by two tensors, A_1, A_2 and has the form

$$|\psi\rangle = \sum_{s} Tr(A^{s_1}_{[1]} A^{s_2}_{[2]} A^{s_3}_{[1]} A^{s_4}_{[2]} ...) |s_1 s_2 s_3 s_4 \ldots\rangle. \tag{2.11}$$

If we carefully take into account the infinite nature of these states, their iMPS representations allow us to study properties of the states directly in the thermodynamic limit, rather than through any kind of finite-size extrapolation. Note however that finding an MPS representation for an infinite system still relies on having limitations on the entanglement. In particular, for an iMPS with finite bond dimension to capture an infinite state, we must still have exponentially decaying correlations, i.e. correlations which are not themselves also infinite [25]. A ground-state preparation algorithm for infinite states will be discussed subsequently in Sec. 3.2.2, and we will present more methods for evaluating properties of infinite states in Chapter 4 and 5.

A final note about the usage of MPS representations in practice: in the foregoing, we have motivated the MPS structure by imagining the decomposition of a general coefficient tensor, and indeed, in principle it is possible to construct an MPS representation of any quantum state with known coefficients in a manner similar to this (for complete details, see Ref. [81]). However, this situation is often not the one which arises in practice. After all, our desire is to find representations for quantum systems with so many particles that their coefficient tensors become unworkably large; in such cases, of course we will not already be in possession of a coefficient tensor which we can decompose. Instead, we shall more typically be developing numerical methods in which we consider *only* states with an MPS form, and then numerically search for particular state of interest from among this class of MPS states. Examples and descriptions of such methods will be presented in Sec. 3.1 and Sec. 3.2.

Gauge Freedom and Canonical Form

Before presenting the details of the algorithms, we wish to identify a few additional properties of MPS states and their generalizations. First and foremost, it is important to observe that some gauge freedom is allowed in the representation of MPS states. For example, consider a periodic state represented as in Eq. 2.9 by a set of tensors $\{A_{[1]}, A_{[2]}, \ldots A_{[L]}\}$. If for some appropriately-sized matrix X and some phase factor $e^{i\phi}$ we transform each MPS tensor according to

$$A^{s_j}_{[j]} \to e^{i\phi} X A^{s_j}_{[j]} X^{-1} \tag{2.12}$$

then by substitution of this transformation into Eq. 2.9 we can see that all the X's will cancel and the state itself will only change by an overall factor $e^{iL\phi}$. Since the global phase is not observable, this transformation of the tensors essentially leaves the state invariant. Indeed, this isn't even the most general transformation that could be made in this case without changing the state; one can easily imagine a site-dependent transformation $A^{s_j}_{[i]} \to e^{i\phi} X_{[j]} A^{s_j}_{[i]} X^{-1}_{[j+1]}$ which accomplishes the same thing. Clearly, a single physical state can be represented by many different sets of MPS tensors.

This freedom allows us to make some choices about the structure of the representation which will prove useful in subsequent calculations. In particular, we can choose our MPS to be represented in the so-called "canonical form" [15], in which each of the state tensors satisfy the property,

$$A^{j}_{\alpha,\beta} (A^{*})^{j}_{\alpha',\beta'} \delta^{\beta,\beta'} = \delta_{\alpha,\alpha'} \tag{2.13}$$

Note that in this case, we have explicitly labelled both the physical index and the virtual indices of the tensors for clarity, with Greek indices for the virtual space. Strictly speaking, the canonical form of Ref. [15] also implies the related condition that

$$(\Lambda_{[i-1]})_{\alpha,\alpha'} (A_{[i]})^{j}_{\alpha,\beta} (A^{*}_{[i]})^{j}_{\alpha',\beta'} = (\Lambda_{[i]})_{\beta,\beta'} \tag{2.14}$$

for some set of positive, diagonal matrices $\{\Lambda_{[i]}\}$, which are related directly to the Λ's of Eq. 2.6. However in this work, when we refer to employing the canonical form, we will be using the property in Eq. 2.13 unless otherwise stated.

Generally, the properties of canonical forms will turn out to be quite useful in our applications of MPS states, and hence it is desirable to work with MPS states which have these properties. Happily, when an MPS is constructed from a coefficient tensor through successive SVD's as described above, that state is already in canonical form [15, 82]. Essentially, this occurs directly from the unitarity of the A^{s_j} which arise from the SVD in Eq. 2.6; for greater detail see Appendix B in Ref. [82].

To achieve the canonical form in other cases, analytical procedures have been given in Refs. [15, 83] for transforming a general MPS without changing the physical state. Furthermore, the numerical application of sequential Schmidt decomposition to all pairs of neighboring sites can be used to approximate the canonical form up to machine precision [84, 85]. This latter method, which we employ for most of our numerical algorithms, is discussed below in Sec. 3.2.1.

Transfer Matrices

The canonical condition for the MPS tensors in Eq. 2.13 can also be thought of in terms of the "transfer matrix" for those tensors (see Fig. 2.5). This object, a common construction used in MPS formalism to compute things like expectation values, is given by:

$$T^{(\alpha\alpha')}_{(\beta\beta')} \equiv A^j_{\alpha,\beta}(A^*)^j_{\alpha',\beta'}. \tag{2.15}$$

Now consider the dominant eigenvector of T, which will be some vector $X_{(\beta,\beta')}$. Because the outgoing indices of T are a composite of smaller indices (β,β'), any eigenvector of this matrix can also be thought of as a (smaller) matrix in its own right, by interpreting $X_{(\beta,\beta')}$ as $X^{\beta}_{\beta'}$, so that $X_{(\beta,\beta')}$ is a vectorization of the matrix $X^{\beta}_{\beta'}$. Now, the condition for canonical form can be rephrased as the requirement that the dominant eigenvector of the state's transfer matrix is a vectorization of the identity matrix, i.e.

$$(T)^{(\alpha\alpha')}_{(\beta\beta')}\delta^{(\beta\beta')} = \delta^{(\alpha\alpha')}. \tag{2.16}$$

For a graphical depiction, see Fig. 2.5.

Of course, there is nothing special about the fact that we have considered the right eigenvector. From the transfer matrix point of view, we now see that we could have also defined a comparable choice of canonical form in which the tensors satisfy instead

$$\delta^{(\alpha\alpha')}(T)^{(\alpha\alpha')}_{(\beta\beta')} = \delta^{(\beta\beta')}. \tag{2.17}$$

We call these conditions the "right" and "left" canonical forms of an MPS. In general when we refer subsequently to states in "canonical form," we mean the right canonical form, although as we will see in Sec. 3.1, there will be situations where it is advantageous to use a combination of both right and left.

Let us now look more closely at the transfer matrix. Because it represents a contraction of the physical indices of the tensors A^j, the transfer matrix can be thought of as containing the overlap of the state with itself at a single site. In other words, in an L-site periodic state with one-site translation invariance, the norm square of the state is given by taking a product of L transfer matrices (one for each site) and then tracing over them.

$$\langle\psi|\psi\rangle = Tr(T^N) \tag{2.18}$$

This fact in turn produces a relationship between the eigenvalues λ_j of the transfer matrix, and the norm of the state. Consider for example an infinite-length, translation-invariant state with unique largest eigenvalue λ_1, whose norm is given by $\lim_{N\to\infty} Tr(T^N) = \sum_j \lambda_j^N \approx \lambda_1^N$. This state is normalized if $|\lambda_1| = 1$. Hence in this case, computing the largest eigenvalue of the transfer matrix gives us a convenient way to ensure normalization.

2.2.3 Matrix Product Operators

As difficult as it is to represent an arbitrary quantum state because of large the Hilbert space becomes relative to the number of particles, recall that representing a generic *operator* on such space becomes even more prohibitive. One might naturally wonder if the same techniques used to represent states as MPS could be used to condense the representation of the associated operators. And one would find that indeed they can.

The ansatz in this case is called a "matrix product operator" (MPO) [26, 32, 86, 87]. In a manner quite similar to Eq. 2.19, the MPO representation of some operator O consists of taking the expansion of the operator in some orthonomal basis, and expressing the coefficients with a product of tensors. The only difference is that the tensor associated with each site will now have to carry *two* physical indices (corresponding to the incoming and outgoing indices of O), so that in total the tensors are rank four. We write this representation as

$$O = \sum_{s,s'} \left(\langle V_{\text{left}}| M_{[1]}^{s_1,s'_1} M_{[2]}^{s_2 s'_2} \dots M_{[L]}^{s_L s'_L} |V_{\text{right}}\rangle \right) |s_1 s_2 \dots s_L\rangle\langle s'_1 s'_2 \dots s'_L| \tag{2.19}$$

Note that in addition to the set of tensors, for the open boundary case we must specify also two boundary vectors $\langle V_{\text{left}}|$ and $|V_{\text{right}}\rangle$. These vector set

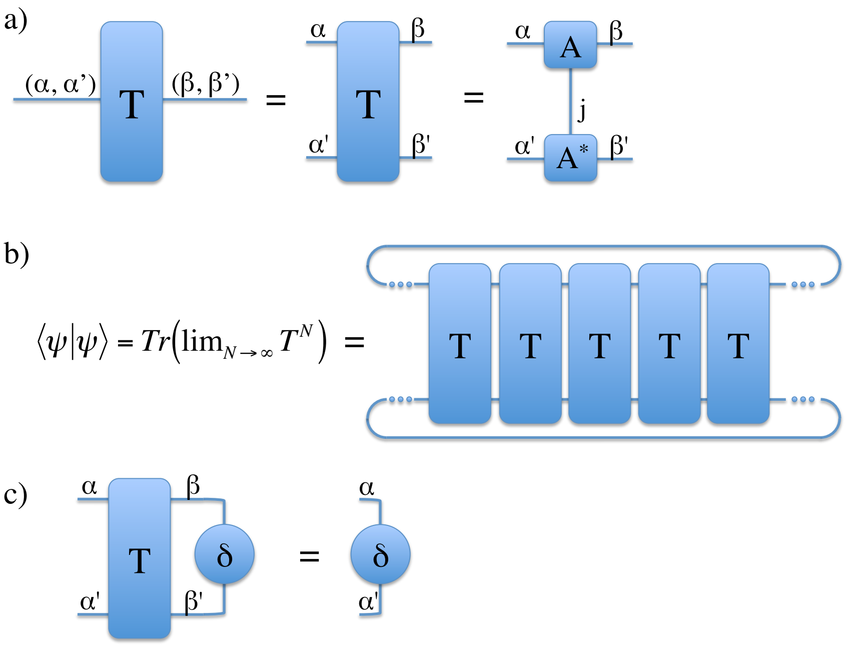

Figure 2.5: (Color Online). The transfer matrix of a translationally-invariant matrix product state, demonstrated in graphical tensor notation. In (a), the construction of the transfer matrix is shown as a contraction of two MPS matrices, with the virtual indices grouped to form a single matrix. In (b), the relationship between the transfer matrix and the norm square of the state is shown. Finally, in (c) we show graphically the behavior of a matrix product state in canonical form: such a state has a transfer matrix whose dominant eigenvector is a vectorized version of the identity matrix. In particular, this demonstrates the "right" version of canonical form; the condition for left canonical form appears the same, but with a horizontal reflection.

the boundary conditions of the system and ensure that the product in Eq. 2.19 gives a scalar result. We have included them here for clarity, but elsewhere in this work we shall absorb them into the definitions of $M_{[1]}$ and $M_{[L]}$. The MPO description can also be generalized to cases with periodic boundary conditions (where a trace replaces the boundary vectors), or to infinite systems (by means of a representative unit cell).

Like an MPS, an MPO can be described by the bond dimension, i.e. the dimension of the virtual indices of its tensors. And like an MPS, there is no guarantee that a generic operator can be expressed efficiently in this language, without resorting to an exponentially large bond dimension. But perhaps unsurprisingly, operators which can be expressed as a sum of local terms–such as Hamiltonians whose ground states admit efficient MPS representations–are themselves well-suited to representation as an MPO. For example, consider a Hamiltonian composed of some two-body terms X_iY_{i+1} and some onsite operations Z_i with coefficients $\{J_i\}$ and $\{h_i\}$, i.e.

$$H = \sum_i J_i X_i Y_{i+1} + h_i Z_i. \tag{2.20}$$

It is convenient to represent the rank-four tensors of Eq. 2.19 as matrix-valued matrices. The indices of the "outer" matrix are the virtual indices of the tensors, whereas the indices of the "inner" matrices (the elements of the outer matrices) are the the physical indices of the tensor. Under this convention, The Hamiltonian in Eq. 2.20 is given by the tensors.

$$M_{[i]} = \begin{pmatrix} \mathbb{1} & 0 & 0 \\ Y & 0 & 0 \\ h_i Z & J_i X & \mathbb{1} \end{pmatrix}, \tag{2.21}$$

coupled with the boundary vectors

$$\langle V_{\text{left}}| = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}, \tag{2.22}$$

and

$$|V_{\text{right}}\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \tag{2.23}$$

Note also that the representation achieved here is *exact*, with no approximate truncations. Additional two-body terms in the Hamiltonian can be accommodated by inserting additional rows and columns, keeping the identity matrices as the first and last diagonal elements, and always keeping the single-site term in the lower left corner. Terms with three-body operations can be incorporated by including additional nonzero terms beyond the first column, although this can be done while maintaining a lower-triangular structure [32]. In general, this construction can represent an operator with up to n different types of interaction terms in an MPO with a bond dimension of most $n+2$.

The MPO and MPS structures are also well-suited to calculating expectation values. Just as one would compute $\langle\psi|O|\psi\rangle$ by contracting the indices of bras and kets onto the incoming and outgoing indices of the operator O, so too can we compute this quantity by contracting all the physical indices of the tensors representing $\langle\psi|$ and $|\psi\rangle$ around the MPO tensors representing O, as shown in Fig. 2.4.

2.2.4 PEPS

Finally, we conclude by considering the obvious question: can the matrix product structure be generalized to systems with more than one spatial dimension? The answer is that indeed they can, although not all of the properties and advantages of the representation are present to the same degree. The MPS structure took the form of a rank three tensor associated with each site along a chain, with the virtual indices of each tensor all contracted together. We can naturally imagine the same structure generalized to an arbitrary graph, with a tensor at each site of rank $z+1$, where z is the coordination number of the site. An example for the case of a square lattice is demonstrated in Fig. 2.4. As with an MPS, this network could represent a state with the expansion coefficients for each basis vector given by fixing the physical indices and contracting the virtual degrees of freedom. Since this contraction involved a product of matrices in the case of an MPS, and now involves contracting a more general tensor network, it would seem natural to call this representation a "tensor network state." But while this term can be found in the literature, due to historical reasons [31] and notational inertia it is more commonly called a "projected-entangled pair state" (PEPS) [14, 16, 31, 32, 34, 35]. In reluctant deference to this emerging consensus, we shall use that term here as well.

For the most part, PEPS (and iPEPS, their generalization to infinite systems with translation invariance [33]) share the important qualities of the MPS structure. For example, since the entanglement of any particular site with its neighbors is limited by the bond dimension of the PEPS, it can be shown that PEPS states also satisfy an area law with respect to the entanglement of subsystems with their environment [88, 89]. In turn it can be shown that the ground states and low-lying excitations of gapped two-dimensional Hamiltonians can be efficiently represented by PEPS [33, 80, 88, 89]. As with the MPS case, this family of states contains many systems of particular practical or theoretical significance, including states with special significance to the study of quantum computing, such as the Cluster state [90, 91], the two-dimensional generalizations of the AKLT state [92–94], or the toric code [95]. In fact in some ways the power of PEPS extends beyond what was possible with an MPS; whereas MPS with finite bond dimension could only represent states with exponentially decaying correlations, a finite-bond PEPS can represent a state whose correlations correlations decay only *polynomially* with distance [88], meaning that we can also represent states from gapless Hamiltonians.

But as one might expect, the increased complexity of two-dimensional systems also comes at a much greater numerical cost. Evaluating any properties of a PEPS requires contracting the underlying two-dimensional tensor network, and in general exact calculation has been shown to be exponentially costly in the size of the system (in particular, it is a #P-hard problem [96]). The reason for this may be intuitively apparent: earlier we remarked that when contracting a network of tensors, the optimal sequence of contractions is the one which minimizes the number of free indices which appear during the intermediate steps. For arbitrary two-dimensional graphs (picture, for example, a square lattice), this number of free indices will inevitably be of order $\sqrt{N}$, where N is the total number of tensors [72]. As a result it is generally necessary to keep the bond dimensions of the tensors very small; values of $\chi = 2, 3$ or 4 are typical. But even in these cases, the process can be expensive. Consequently, PEPS methods generally rely on a variety of numerical techniques to *approximate* these contractions. These methods typically involve the use of truncated SVD's to remove low-weight information about the contraction; for more details and particular examples, see Refs. [33, 97–101].

Chapter 3

Ground State Preparation Algorithms

As discussed in Chapter 2, much of the power of tensor networks and matrix product states lies in their ability to efficiently represent the ground states (and low-lying excitations) of gapped, local Hamiltonians. Typically, to find an MPS representation for such a state, we start by first *assuming* an MPS structure, and then, within the class of states described by this structure, we search for our desired state. When the state is found, the MPS structure can allow us to study its properties with great efficiency and explore a variety of important topics in quantum many-body physics.

In this chapter we present a detailed look at two classes of algorithms which do precisely this, searching the space of MPS states for the ground states of a given Hamiltonian. The first of these, the so-called "Density Matrix Renormalization Group" method (DMRG), is described in Sec 3.1). The second method, based upon "Time Evolving Block-Decimation" (TEBD) appears in Sec. 3.2). Both families of methods were used extensively to generate the data studied and presented in subsequent chapters, and we give considerable detail about both the theory behind these methods and about our implementations in practice.

3.1 DMRG

Ironically, the Density Matrix Renormalization Group (DMRG) algorithm, perhaps the most widely-known and often-employed tensor network algorithm for finding ground states, was in fact in development *before* tensor network states emerged as a fully-recognized area of study in its own right. Originally, the technique was introduced by White [21, 22], drawing directly (as the name implies) upon Wilson's renormalization-group(RG) method [102, 103], and applied to perform real-space RG calculations for spin chains, and quickly generalized further. Then, during the course of an intervening decade, it was gradually discovered that the states which were well-described by DMRG methods corresponded to those which could be represented as an MPS [20, 23], and that key steps and structures of the DMRG process (as applied to both finite and infinite systems) were deeply connected to the language of tensor networks [23, 24]. Ultimately, it was realized that the entirety of the DMRG technique could be understood as a variational optimization problem for tensor network states [24, 25, 104].

Even then, this striking equivalence was regarded to some extent as a fact of more conceptual than practical significance, since the original purpose of DMRG (to compute ground states for finite spin chains with open boundary conditions) is not noticeably better served by the MPS reformulation [26]. Beginning in the early 21st century, however [31, 86, 105], a wide variety of additional applications and generalizations for the DMRG technique began to be identified which would not have been possibly in the original formulation (notably, even the problem of periodic spin chains, discussed below, is addressed more naturally in the language of MPS). For a more complete review of the nature and history of these techniques, see [25, 26].

3.1.1 Basic Algorithm

We present here a brief overview of the method behind DMRG, as viewed in the framework of a variational problem for matrix product states, for one-dimensional finite systems of length L. This is the context for which it has been principally employed in this dissertation (and which forms the basis for an additional algorithm presented in Chapter 6). We emphasize of course that extensions to higher-dimensional systems exist [31].

Defining the Optimization Problem

Any variational method for finding a (normalized) ground state $|\psi_{gs}\rangle$ of some Hamiltonian H essentially contemplates the inequality

$$\frac{\langle\phi|H|\phi\rangle}{\langle\phi|\phi\rangle} \geq \langle\psi_{gs}|H|\psi_{gs}\rangle \;\forall\; |\phi\rangle, \tag{3.1}$$

which is true almost by definition, with equality only when $|\phi\rangle = |\psi_{gs}\rangle$ up to overall phases and normalization. Any $|\phi\rangle$ which is different from $|\psi_{gs}\rangle$ is expressible as a linear combination of $|\psi_{gs}\rangle$ and other normalized energy eigenstates $|\psi_i\rangle$, *i.e.*

$$|\phi\rangle = c_0|\psi_{gs}\rangle + \sum_i c_i|\psi_i\rangle \tag{3.2}$$

for some $c_i = \langle\psi_i|\phi\rangle$. Consequently, the energy expectation value for such state is

$$\langle\phi|H|\phi\rangle = |c_0|^2\langle\psi_{gs}|H|\psi_{gs}\rangle + \sum_{i=1} |c_i|^2\langle\psi_i|H|\psi_i\rangle. \tag{3.3}$$

Now, because by definition the matrix elements on the right are all bounded below by $\langle\psi_{gs}|H|\psi_{gs}\rangle$, we can write

$$\begin{aligned}\langle\phi|H|\phi\rangle &\geq \sum_{i=0} |c_i|^2\langle\psi_{gs}|H|\psi_{gs}\rangle \\ &\geq \langle\phi|\phi\rangle\langle\psi_{gs}|H|\psi_{gs}\rangle.\end{aligned}$$

and the desired inequality follows directly. From here, it is obvious that the search for the ground state can be cast as an optimization problem by simply seeking the state $|\phi\rangle$ which saturates the bound, which is to say

$$|\psi_{gs}\rangle = \min_{|\phi\rangle} \frac{\langle\phi|H|\phi\rangle}{\langle\phi|\phi\rangle}. \tag{3.4}$$

In the foregoing, we have very intentionally avoided a presumption that the general state $|\phi\rangle$ is normalized. In practice, as one varies numerically the parameters which specify a state, it may well not be a trivial to do so in a manner which preserves the norm while exploring the full domain of possible states. But the norm of the state is an essential part of the desired inequality. Hence, in practice we take the normalization into account as an additional constraint on the optimization problem. A standard approach [14] is to include this constraint by means of a Lagrange multiplier and minimize $\langle\phi|H|\phi\rangle$ subject to $\langle\phi|\phi\rangle = 1$ by minimizing the function

$$\mathcal{L} = \langle\phi|H|\phi\rangle - \lambda\langle\phi|\phi\rangle - \lambda. \tag{3.5}$$

Note that the final term in Eq.3.5 is a constant and can safely be ignored without affecting the minimum energy or the minimizing state.

Optimization with Matrix Product States

If we have reason to believe that our target ground state $|\psi_{gs}\rangle$ can be expressed efficiently as a matrix product state (for example, if we are seeking the ground state of a gapped, local Hamiltonian), then the minimization described above can be restricted to the domain of such states, with the matrices $\{A_{[j]}^{s_j}\}$ of the MPS serving as the variational parameters. Recall that the bracketed labels $[j]$ are part of the "name" of the tensor (not an additional tensor index). As discussed in [14], when viewed this way, the problem amounts to an optimization with respect to a standard multiquadratic cost function. As such, it can be solved by the method of alternating least-squares (ALS), wherein one seeks to optimize over a single parameter while keeping the others fixed, and then iteratively "sweeps" back and forth over the other parameters until a solution converges. In the language of matrix product states, this means moving back and forth across the sites in the spin chain and choosing at each site a new tensor which minimizes the overall energy, treating the other tensors as fixed for the time being. Note that while in principle this method is not guaranteed to globally converge [106] (and one can construct special cases where it does not [107]), in practice the convergence properties of the DMRG algorithm have been extensively studied [108, 109] and judged to be very strong, particularly given the properties of "typical" ground states which matrix product states are naturally designed to represent [13].

At each step of the ALS optimization, we are no longer considering the full cost function of Eq. 3.5. Rather, in step j we are optimizing only the tensor $A_{[j]}^{s_j}$. If we momentarily regard this tensor as a "state" $|A_{[j]}\rangle$ (i.e., we group all the indices and treat them as a combined, "vectorized" index), then the rest of the state can be grouped with the original Hamiltonian H to form an "effective Hamiltonian" H_{eff} with respect to this "state" which is to be optimized. More concretely, let $|A_{[1\ldots j]}\rangle_{v_j}$ be the state represented by the tensors, $\{A_{[1]}\ldots A_{[j]}\}$, with the subscript v_j indicating that the uncontracted virtual index at site j is left open. The effective Hamiltonian can then be described as

$$\begin{aligned} &H_{eff} = \\ &\langle A_{[1\ldots j-1]}|_{v'_{j-1}} \otimes \langle A_{[j+1\ldots L]}|_{v'_{j+1}} H |A_{[1\ldots j-1]}\rangle_{v_{j-1}} \otimes |A_{[j+1\ldots L]}\rangle_{v_{j+1}} \end{aligned} \tag{3.6}$$

Note that the open virtual indices are *not* contracted with one another, and as such are separately labelled with and without primes. The open virtual indices and the remaining physical degrees of freedom from H are grouped to form the indices α and β of $H_{eff}^{\alpha\beta}$. In practice, to achieve this desired overlap, we cast H as an MPO with tensors $\{M_{[i]}^{s'_i s_i}\}_{i=1\ldots L}$, and contract it with the tensors for the MPS at all sites except site j. This structure is substantially clarified by graphical notation as shown in Fig. 3.1, and Fig. 3.3 for the case of periodic boundary conditions.

A similar approach allows us to define N_{eff}, an effective norm operator, with respect to the tensor we are optimizing, i.e.

$$N_{eff} = \\ \langle A_{[1\ldots j-1]}|_{v_{j-1}} \otimes \langle A_{[j+1\ldots L]}|_{v_{j+1}} |A_{[1\ldots j-1]}\rangle_{v'_{j-1}} \otimes |A_{[j+1\ldots L]}\rangle_{v'_{j+1}} \tag{3.7}$$

This structure can be dramatically simplified in the case of open boundary conditions (see Sec. 3.1.1), but for the sake of completeness and clarity we discuss it first in its most general form.

In this language, the cost function in Eq. 3.5 becomes

$$\mathcal{L} = \langle A_{[j]}|H_{eff}|A_{[j]}\rangle - \lambda\langle A_{[j]}|N_{eff}|A_{[j]}\rangle. \tag{3.8}$$

When we consider variation with respect to the tensor, represented here as $\langle A_{[j]}|$, and seek to make this derivative vanish, the optimization simply amounts to solving

$$H_{eff}|A_{[j]}\rangle - \lambda N_{eff}|A_{[j]}\rangle = 0 \tag{3.9}$$

which is simply a generalized Hermitian eigenvalue problem (GHEP), a well-studied class of linear problems which can be numerically solved by any of a number of suitable methods[110] to give us both an eigenvalue λ and a solution vector $|A_{[j]}\rangle$. This vector gives us the new tensor for site j, by simply re-interpreting its vectorized index as the three original tensor indices. We update the MPS with the new tensor, and observe that, by acting this bra on Eq. 3.9, we have

$$\langle A_{[j]}|H_{eff}|A_{[j]}\rangle - \lambda\langle A_{[j]}|N_{eff}|A_{[j]}\rangle = 0 \tag{3.10}$$

Which, by definition of H_{eff} and N_{eff}, implies that

$$\langle A_{[1\ldots L]}|H|A_{[1\ldots L]}\rangle - \lambda\langle A_{[1\ldots L]}||A_{[1\ldots L]}\rangle = 0 \tag{3.11}$$

Or in other words, since $|A_{[1\ldots L]}\rangle$ is just the full state ϕ,

$$\lambda = \frac{\langle\phi|H|\phi\rangle}{\langle\phi|\phi\rangle}. \tag{3.12}$$

Hence, one can see that the eigenvalue is in fact equal to the energy of the new state with respect to the original Hamiltonian. Choosing the vector associated with the smallest such eigenvalue will thus assure that we have locally minimized the energy. We can then move to the next site, and repeat the optimization calculation with respect to the new MPS. Proceeding in this way, the algorithm is in fact guaranteed to be what is known as a “greedy” algorithm: it monotonically pursues the solution by choosing the locally optimal update

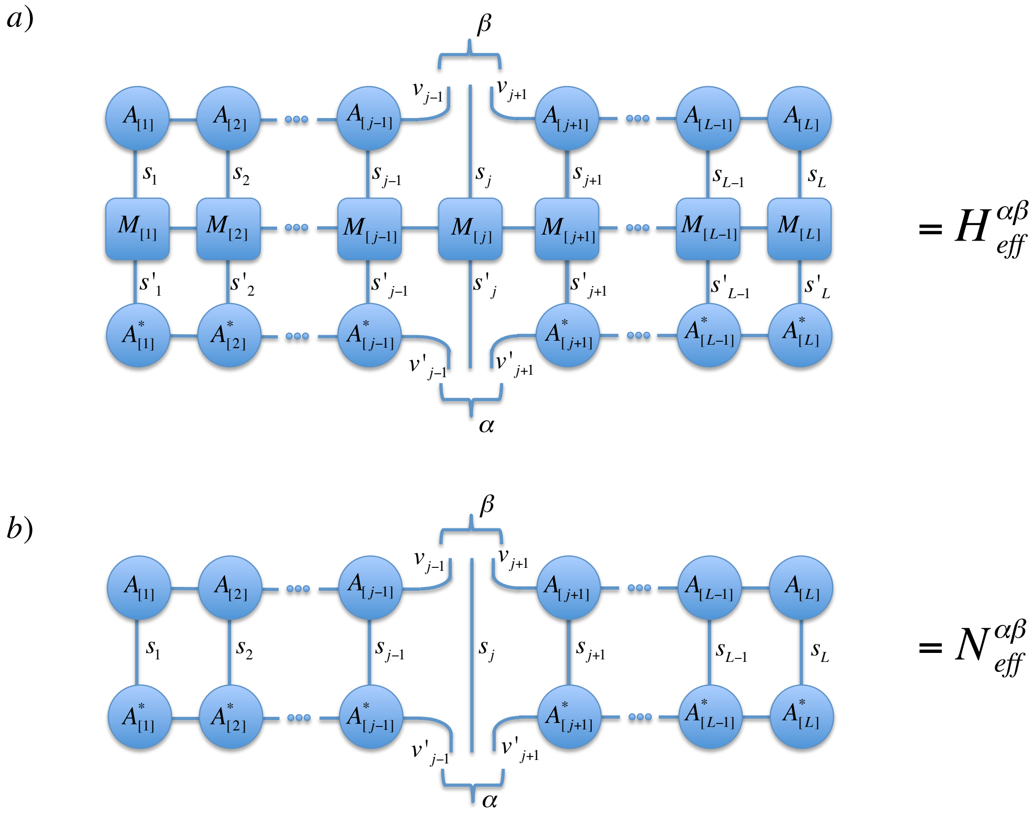

Figure 3.1: (Color online) (a) The matrix H_{eff} with respect to some site j of a system with open boundary conditions, illustrated in graphical notation to show its internal structure, as described in Eq. 3.6. The incoming and outgoing indices α and β are composite indices which group two virtual indices and a physical index of M_j, where the M tensors specify the total Hamiltonian H as an MPO. (b) Similar structure showing the effective norm operator N_{eff} with respect to site j. See Eq. 3.7

at each step.

Broadly, then, the steps of the numerical algorithm are as follows

1. We set a maximum bond dimension χ for our matrix product states and initialize the algorithm with a "random" state $|\phi_1\rangle$; in practice, this is simply an MPS whose L tensors $\{A^{s_1}_{[i]} \dots A^{s_L}_{[L]}\}$ have been chosen with random elements.

2. Beginning at site 1, we fix the tensors at all remaining sites 2 through L and construct H_{eff} and N_{eff} in accordance with Eq. 3.6 and Eq. 3.7.

3. Solve the GHEP of Eq. 3.9 for the smallest eigenvalue λ and interpret the resulting eigenvector as the new tensor $\tilde{A}^{s_1}_{[1]}$.

4. By replacing the old tensor $A^{s_1}_{[1]}$ with $\tilde{A}^{s_1}_{[1]}$, we arrive at a new state $|\phi_2\rangle$.

5. Shift our focus to site two, and repeat the above. These steps are iteratively "swept" along the sites until we reach site L; at this point, a sound numerical practice is to reverse directions and sweep back towards site one. This ensures a sort of symmetry in the overall optimization, so that no portion of the state excessively leads or lags the minimization. During the forward sweep, the optimization of sites at the end of the chain benefit from the additional information provide by optimizing the sites at the beginning. When sweeping backwards, the reverse is true. Note that for this reason, it is good practice to sweep "back and forth" with respect to an arbitrary starting point even for the case of a periodic system, where one could in principle sweep "around" the chain nonstop.

6. The entire process can be repeated until one's desired convergence criteria are met. Further discussion of convergence criteria can be found below in Sec. 3.1.1; comments in Sec. 3.2.5 may also be applicable.

Numerical Considerations

When implementing this algorithm in a programming language, a number of important practical considerations emerge to aid in the efficiency of the algorithm. Perhaps the most significant issue arises in the case where one considers a system with open boundary conditions. Recall the canonical form (discussed in Sec. 2.2.2) into which an MPS can be placed. Suppose we have reached step j of a DMRG sweep, i.e. that we are preparing to optimize the tensor $A^{s_j}_{[j]}$. As demonstrated graphically in Fig. 3.2, if the tensors $A^{s_i}_{[i]}$ for $i > j$ satisfy the "right" canonical form condition (Eq. 2.16), then the entire structure of N_{eff} to the right of site j is simply equal to the identity matrix

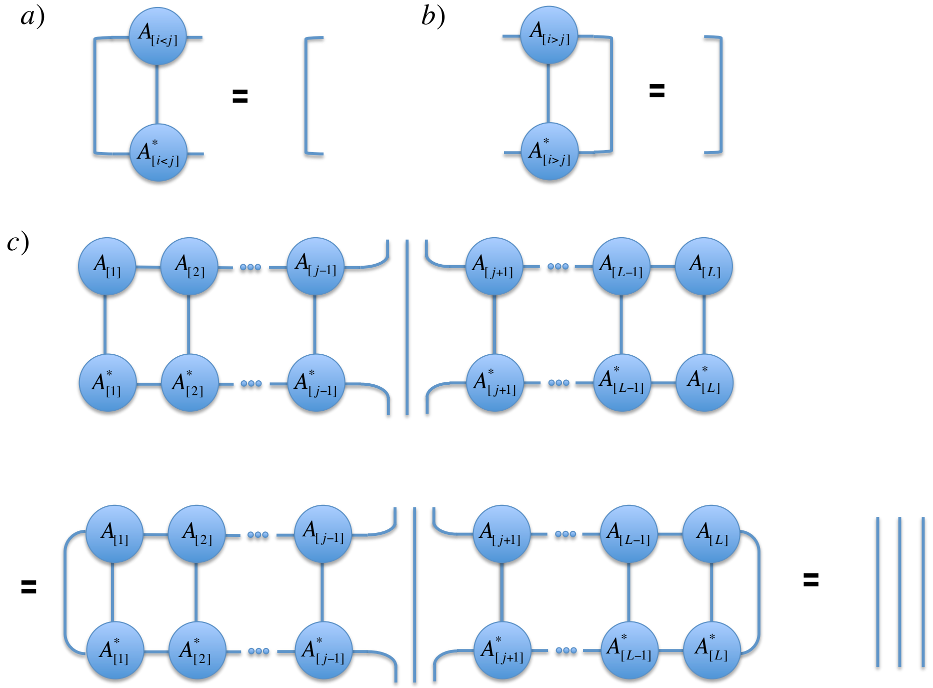

Figure 3.2: (Color online) With open boundary conditions, the effective norm tensor N_{eff} at site j can be substantially simplified if certain conditions are enforced. (a) First, we must place the state into a left canonical form for all tensors $A_{[i]}$left of site j, i.e. $i < j$. (b) Next, we must similarly place all tensors $A_{[i]}$ for $i > j$ in a *right* canonical form. (c) As a result, the tensor N_eff can be "collapsed in" by repeated applications of properties (a) and (b) (recall that the initial structure has only a null index of dimension zero at the boundary).

and can be neglected. In the same manner, if the tensors $A^{s_i}_{[i]}$ for $i < j$ are in "left" canonical form (Eq. 2.17), the same can be said about N_{eff} to the left of site j. As such, although there is a small extra computational cost involved, it us ultimately quite profitable to explicitly enforce these canonical forms throughout the DMRG sweeps, since we can then avoid constructing the entire N_{eff} tensor at each step. The result is that one can avoid a large number of repeated tensor contractions, far outweighing the cost of the canonical form.

Indeed, it is relatively simple to enforce the canonical conditions in this case, as we are dealing with the state only one tensor at a time. Let us suppose we initialize our system with a state which is already in canonical form (either by construction, by application of the techniques in Refs. [15, 83],or through the identity gate TEBD method of Sec. 3.2.1). At each step, once one obtains the optimized tensor $A^{s_j}_{[j]}$, it can be reshaped and decomposed by SVD, keeping only the unitary parts which already satisfy canonical form, as in Eq. 2.6. Then, the remaining pieces of the decomposition can simply be neglected, since at step j they would have been "pushed" onto the tensor $A^{s_{j+1}}_{[j+1]}$. But this is the tensor which will be replaced by optimization in the very next step, so there is no need to change it now. This keeps the state in right canonical form; when left canonical form is desired, we simply group the indices differently before the SVD. Note, however, that we can now see why this optimization is only available for the open boundary case: since the tensors to the left and right of the current site have to be placed in canonical form in different directions, in a periodic chain these successive transformations will eventually come into conflict.

Unfortunately, while this technique allows one to avoid the costly construction of the large N_{eff} matrix in the open boundary case, there still remains the even-more-expensive process of constructing H_{eff} at each site. And for the periodic case, construction of *both* H_{eff} and N_{eff} is unavoidable. Nevertheless, the computational cost of this process can be minimized by using a basic form of "dynamical programming," i.e. by allowing the algorithm to store the result of certain tensor contractions when one can anticipate that they will be reused, minimizing the number of redundant steps. In this context, the potential redundancies become apparent when one considers that H_{eff} and N_{eff} can be viewed as products of (potentially generalized) transfer matrices (see Fig 3.3). For example, at step j, the "left side" of N_{eff} is given by

$$N^{(j)}_{left} = \prod_{j'}^{j-1} T_{j'}. \tag{3.13}$$

where

$$T_j = \sum_{s_j} A^{s_j}_{[j]} A^{*s_j}_{[j]}. \tag{3.14}$$

In the subsequent step $j+1$, we will be constructing

$$N^{(j+1)}_{left} = \prod_{j'}^{j} T_{j'} = N^{(j)}_{left} \cdot T_j, \tag{3.15}$$

which simply builds upon the result from the previous site. Analogous statements can be made about H_{eff}, with the transfer matrices $T_{j'}$ generalized to include the matrices from the Hamiltonian MPO. Storage of these transfer ma-

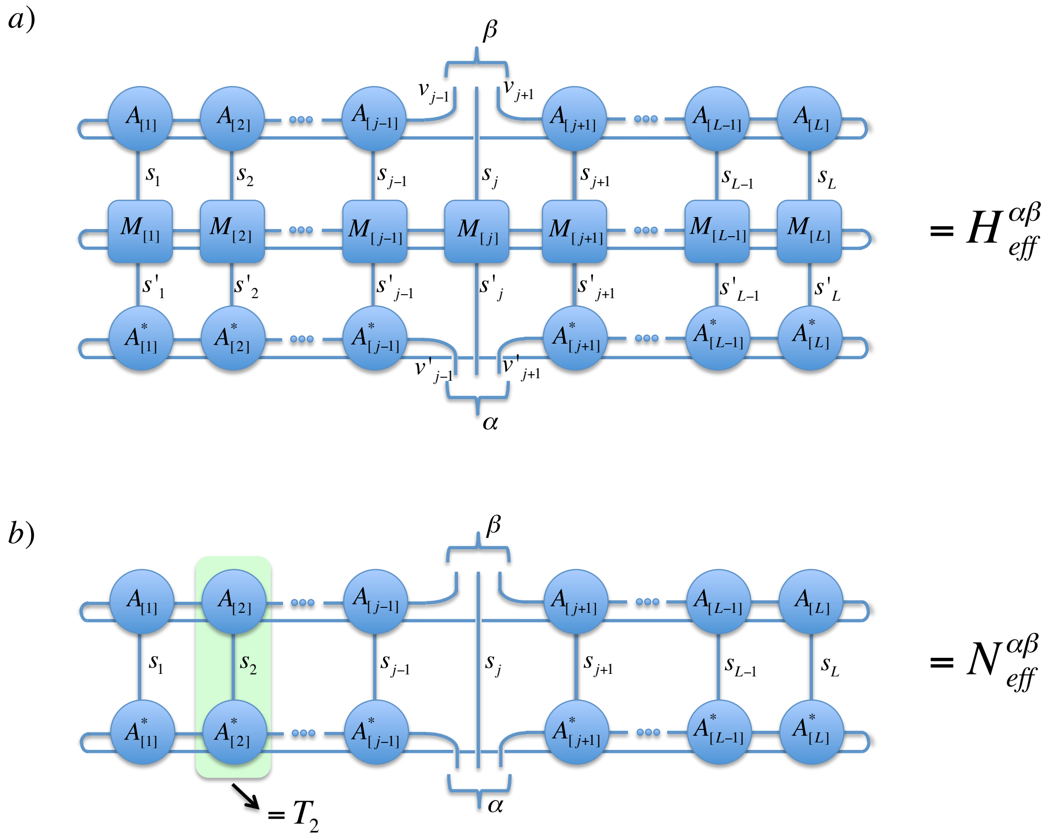

Figure 3.3: (Color online) The structure of the effective operators (a) H_{eff} and (b) N_{eff} for the case of a system with periodic boundary conditions; compare to Fig. 3.1. Note also that these structure can be though of as a product of transfer matrices; one such transfer matrix is highlighted here in green.

trix products can therefore substantially reduce computational effort; naively constructing the left side of these structures at each site requires j transfer matrix constructions at step j, for a total of $L(L+1)/2$ operations in a system of length L. If, however, we save the previous structures during the sweep, we need only construct one additional transfer matrix at each site, for a total cost which is only linear rather than quadratic in L. Furthermore, since the relevant left structure will always be given by the previous left structure times a newly updated transfer matrix, the number of transfer matrix multiplications can also be made linear in L throughout the sweep.

The situation is similar with respect to the structure on the right. Here, since the relevant tensors are not changing during the update sweep, the requisite transfer matrices and products of transfer matrices can all be computed in advance and simply stored for use a the appropriate step, and the costs remain similarly linear. Note of course when the algorithm is sweeping "back" to the left, the same statements apply, but to the right and left sides, respectively. This storage system also provided additional motivation for using a "back-and-forth" sweep even in the case of periodic systems.

The remaining consideration is the same one which arises in nearly every numerical method: the choice of a halting condition. Here, we have vaguely stated that we wish our algorithm to sweep back and forth optimizing the tensors until the ground state is "converged," but we have specifically not defined that notion as there are many criteria one might choose to set to signal convergence. This choice may depend strongly on the parameters of the problem at hand, but one common choice is to monitor the energy of our wavefunction $|\phi\rangle$. This quantity is available for "free," since it emerges as λ, the eigenvalue of the GHEP which must be computed anyway. In principle, when the algorithm approaches a ground state, the values of λ should be asymptotically approaching some value from above, and hence the difference between values of λ after successive steps should get increasingly small. One can monitor this change in λ between steps j and $j+1$ either by looking directly at $|\lambda_j - \lambda_{j+1}|$, or by some more sophisticated method such as by considering the standard deviation of a some set of the most recent values of λ. This method is cheapest and highly advisable for cases where the energy itself is a quantity of primary interest. Its principle drawbacks are the possibility that convergence can be falsely triggered if the algorithm settles into a point of metastability, or that some superposition of low-lying states is being reached with energy close to the ground state. These can be guarded against in part by looking at the variance of the energy, $\langle(\Delta H)^2\rangle = \langle H^2\rangle - \langle H\rangle^2$, which should become vanishingly small for a true eigenstate. More comments on use of this quantity for convergence are discussed in the context of the iTEBD algorithm in Sec. 3.2.5; we provide an efficient method to compute this quantity in Chapter 4.

An alternative point of view on convergence criteria would be to look at how the state $|\phi\rangle$ itself is changing, perhaps by computing the overlap $\langle\phi_j|\phi_{j+1}\rangle$ between steps j and $j+1$ and seeking to let the magnitude of this quantity converge towards 1. Such overlap is more costly and does not resolve the possibility of false metastability, but can be a stronger convergence criterion if one wishes to ensure that all observables of the state are as strongly converged as the energy. As a compromise, one might also consider only the change in the particular *tensors* as the states are updated, but must be careful to take possible gauge freedoms into consideration.

3.1.2 Additional Applications and Extensions

First and foremost, it is important to remark that there is an important generalization of DMRG to the case of an infinite, translationally invariant system (represented by the unit cell in an iMPS). This technique is typically termed iDMRG [32, 111]. Although iDMRG is certainly widely employed, our implementation of the algorithm was not used for the work presented here, and hence we refer the reader to the original references for more information; helpful and approachable overviews of the basic technique can also be found in Refs. [44, 112].

In our discussion of DMRG, we have also focused specifically on finding ground state wavefunctions, for two principle reasons. First, these ground states are guaranteed to admit efficient MPS representation in a manner which excited states are not. But secondly, and more importantly, the variational principle as stated above is premised upon properties of the ground state, i.e. that by definition it has the lowest energy of any state in the energy eigenbasis with which we expanded our trial wavefunction.

Nevertheless, if one *knew* the ground state $|\psi_{gs}\rangle$, one could identify the first excited state by solving the same optimization problem articulated in Sec. 3.1.1, but minimizing only over the sector of states $|\tilde{\phi}\rangle$ which satisfy $\langle\tilde{\phi}|\psi_{gs}\rangle = 0$. It is by this process that RG methods and in particular DMRG[14, 26, 87] have traditionally been brought to bear on low-lying excitations even beyond the first excited state. Indeed, if one knows the first n excitations $\{|\psi_{gs}\rangle, |\psi_1\rangle, \cdots, |\psi_n\rangle\}$, then one can in principle variationally seek $|\psi_{n+1}\rangle$ by solving the optimization problem from among wavefunctions mutually orthogonal to the known states.

One can envision the incorporation of this constraint by means of the transformation $H_{eff} \rightarrow P^T H_{eff} P$ and $N_{eff} \rightarrow P^T N_{eff} P$, where P is a projector onto the appropriate orthogonal subspace. In fact, in the language of MPS, such projectors can be efficiently computed and applied, even for multiple states [87]. Calculation then proceeds as before, subject only to the need for the resulting wavefunction be projected back. Naturally, however, this projector is itself going to be an approximation, as the true ground state wavefunction is unlikely to be known *exactly* in the cases of interest. Typical practice is to first compute the approximate ground state $|\phi_0\rangle$ as described above, and then repeat the method for states orthogonal to this estimation. The process can continue to be iterated, with the proviso that error will increasingly accumulate: the wavefunctions themselves are approximations, and they are computed subject to increasingly-approximate orthogonality constraint. Note however that in some cases, if symmetry properties of the low-lying states are known (and differ from the ground state), then enforcing these symmetries on the MPS throughout the DMRG algorithm can achieve the same effect [87, 113, 114].

3.2 TEBD and iTEBD

The second major family of algorithms are based upon a technique referred to as Time Evolving Block Decimation (TEBD)[27, 81, 115]. As the name suggests, they were originally introduced as a method to simulate unitary time evolution of a system under a known Hamiltonian, although from their inception it has been recognized that they could also be used for ground state preparation be means of evolution in *imaginary* time. Whereas ordinary unitary time evolution of a state $|\phi_0\rangle = |\phi(t=0)\rangle$ proceeds according to

$$|\phi(t)\rangle = e^{-itH}|\phi_0\rangle, \tag{3.16}$$

by taking an "imaginary time" parameter $\tau = it$ we can also consider the state

$$|\phi(\tau)\rangle = \frac{e^{-\tau H}|\phi_0\rangle}{||e^{-\tau H}|\phi_0\rangle||}, \tag{3.17}$$

where here we have explicitly included a normalization since the operator $e^{-\tau H}$ is no longer unitary. Let H be a gapped Hamiltonian and, as before, consider a random initial starting state $|\phi_0\rangle$ expanded in $\{|\psi_i\rangle\}$ the energy eigenbasis of H, so that $|\phi_0\rangle = \sum_i c_i|\psi_i\rangle$. After application of $e^{-\tau H}$, we have up to normalization

$$e^{-\tau H}|\phi_0\rangle = \sum_i c_i e^{-E_i\tau}|\psi_i\rangle. \tag{3.18}$$

It is clear that, as $\tau \to \infty$, every term in the right hand side will be exponentially suppressed relative to the ground state $|\psi_{gs}\rangle$, whose energy is the smallest and whose coefficient $c_0 e^{-E_0 \tau}$ is hence the largest for large τ. This assumes, of course, that the initial state $|\phi_0\rangle$ has nonzero overlap $\delta = |\langle\phi_0|\psi_{gs}\rangle|$ with the ground state, which in practice will be true for an initial state which is suitably random. In particular, when the Hamiltonian has a gap of size Δ, the suppression of the excited states in Eq. 3.18 can be shown to be [27]

$$|\langle\phi_\tau|\psi_{gs}\rangle > 1 - \mathcal{O}\left(\frac{e^{-2\Delta\tau}}{\delta^2}\right). \tag{3.19}$$

Indeed, while the above equation is explicitly applicable only for systems of finite length (where nonzero δ has clear meaning) and for gapped Hamiltonians, in practice it has been consistently shown that the principle of imaginary time evolution extends well beyond these limitations, as will be discussed below. The result in general is a powerful method for pursuing the ground state: in a certain sense, the pursuit of the ground state state by the imaginary time evolution of Eq. 3.17 is simply an alternative way of attempting to solve the optimization problem of Eq. 3.4, with the state updated globally according to the imaginary time evolution operator rather than locally according to the DMRG method described above.

There are two essential elements to TEBD algorithms for matrix product states: the truncated SVD, discussed above in Sec. 2.2.1, and the Suzuki-Trotter decomposition, which we shall discuss below, before synthesizing both techniques into a complete algorithm.

Suzuki-Trotter Approximations

The need for the Suzuki-Trotter decomposition arises from the fact that, although we can often efficiently apply the operator H to a matrix product state when H is bounded and local by means of a compact MPO, there is unlikely to be a similarly efficient MPO representation of the operator $e^{-\tau H}$. The possibility of a compact description of H comes from the fact that each of its terms acts only to a finite number of local sites, but the exponentiation generally mixes the action at each site into a global operation. Imagine, however, the special case of a (perhaps boring) Hamiltonian $H = \sum_i h_i$, whose terms each act only at a single site and hence mutually commute, $[h_i, h_j] = 0$. In this case, $e^{-\tau H} = \prod_i e^{-\tau h_i}$ separates into a product of pieces which (a) act only on a single site at a time, and (b) mutually commute. Such pieces could be applied efficiently to an MPS in a parallel manner, as we will demonstrate in greater detail in Sec. 4.2.

Of course this compelling property does not exist for the more typical and interesting class of Hamiltonians we generally consider, which might contain e.g. two-body terms, so that $H = \sum_i h_{i,i+1}$. These terms do not by nature have the pleasant property of mutual commutation. But they *can* be partitioned into two classes of terms: those which act first on an odd-numbered site (such as $h_{1,2}$, $h_{3,4}$, etc.) and those who act first on an even-numbered site (such as $h_{2,3}, h_{4,5}$). As such, H can be written as

$$H = H_{odd} + H_{even}, \tag{3.20}$$

with

$$H_{odd} = \sum_{i_{odd}} h_{i,i+1} \tag{3.21}$$

and

$$H_{even} = \sum_{i_{even}} h_{i,i+1}. \tag{3.22}$$

The terms within $e^{-\tau H_{even}}$ (and $e^{-\tau H_{odd}}$) now individually possess the desirable property of mutual commutation, and can be applied while considering only local information about the state. Each individual exponential can therefore be applied very efficiently (see also further details in Chapter 4). The only problem that remains is to find an efficient description of $e^{-\tau H}$ in terms of $e^{-\tau H_{odd}}$ and $e^{-\tau H_{even}}$.

This, happily, is exactly the class of problem contemplated (and solved) by Trotter [116] and subsequently brought to bear on many-body physics problems by Suzuki [117, 118]. To second order in small $\delta\tau$, the approximation which now bears their name has the form

$$e^{-\delta\tau(H_{odd}+H_{even})} \approx e^{\frac{-\delta\tau}{2}H_{odd}} e^{-\delta\tau H_{even}} e^{\frac{-\delta\tau}{2}H_{odd}}. \tag{3.23}$$

Consequently, one can now imagine applying the entire operator $e^{-\delta\tau H}$ as a sequence of just three steps: $e^{\frac{\delta\tau}{2}H_{\text{even}}}$, $e^{-\delta\tau H_{\text{odd}}}$, and $e^{\frac{\delta\tau}{2}H_{even}}$, with many operations performed (nearly) in parallel at each step. These steps, are sometimes called "layers" because of how they appear in graphical notation; see for example Fig. 3.4. In any particular "layer," the parameter $\delta\tau$ will be very small (it must be, for Eq. 3.23 to hold), but by repeating this sequence of layers many times, one can achieve imaginary time evolution (or indeed, a real time evolution) out to large values of total accumulated time. The precise choice of how many steps to take and corresponding choice of step size $\delta\tau$ becomes in practice a matter of careful numerical tuning and choice of convergence criteria. It will be discussed in greater detail in Sec. 3.2.5.

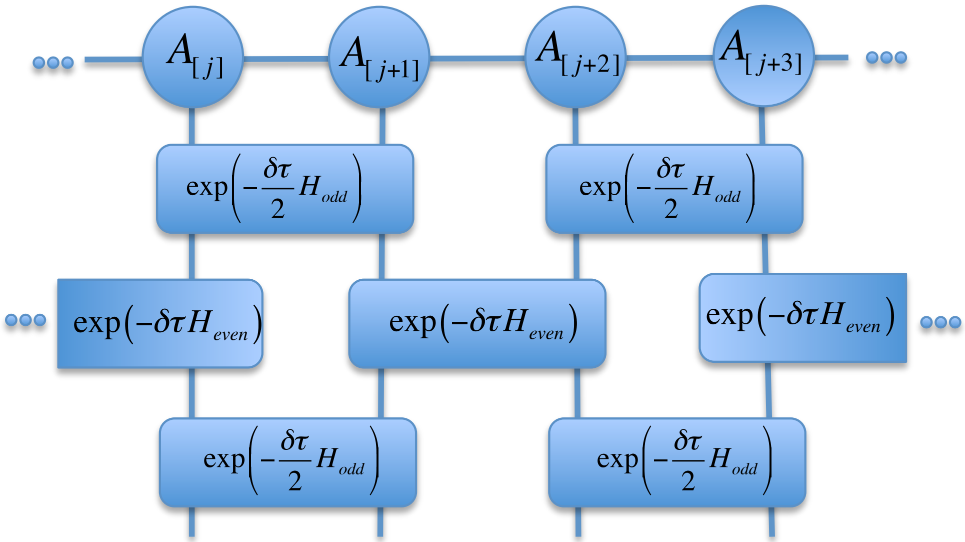
$A_{[j]}$
$A_{[j+1]}$
$A_{[j+2]}$
$A_{[j+3]}$
$\exp\left(-\frac{\delta\tau}{2}H_{odd}\right)$
$\exp\left(-\frac{\delta\tau}{2}H_{odd}\right)$
$\exp(-\delta\tau H_{even})$
$\exp(-\delta\tau H_{even})$
$\exp(-\delta\tau H_{even})$
$\exp\left(-\frac{\delta\tau}{2}H_{odd}\right)$
$\exp\left(-\frac{\delta\tau}{2}H_{odd}\right)$

Figure 3.4: (Color online) Graphical notation demonstrating the application of $e^{-\delta\tau H}$ for $H = \sum H_{odd} + H_{even}$ to an MPS. The evolution operators is decomposed into three "layers" by a second order Suzuki-Trotter decomposition (Eq. 3.23). Within each layer, terms are mutually commuting and can be applied efficiently to the MPS through the "master step" of the TEBD algorithm (Fig. 3.6). This sequence of layers can then be repeated many times until the desired amount of imaginary time evolution is achieved.

Note that while the second-order approximation in Eq. 3.23 is typically good enough for many applications, higher-order versions exist which minimize the error relative to the time step. When using TEBD or iTEBD to compute a ground state, which will inevitably be reached as a result of the exponential suppression of excited states, this dependence of the so-called "Trotter error" on the time step may often be of little consequence. But when using these techniques to perform *real time* evolution, one may wish to study the wave function at many intermediate points throughout the evolution, not simply at the end, and to do so with the largest acceptable timestep to improve speed and efficiency (the same is also true for comparable algorithms which consider finite temperature states; see [119, 120]). Thankfully, higher-order approximations have been widely studied[121, 122] and can be generated to the desired order with relative ease[123, 124]. One particularly common and useful higher-order expansion[123] is the fourth-order formula of the form

$$e^{-\delta\tau H_{even}+\delta\tau H_{even}} \approx U(\alpha_1)U(\alpha_1)U(\alpha_2)U(\alpha_1)U(\alpha_1) \tag{3.24}$$

where the component exponentials $U(\alpha)$ are each themselves second-order decompositions; i.e. they are defined as

$$U(\alpha) = e^{\frac{\alpha\tau}{2}H_{even}}e^{\alpha\tau H_{odd}}e^{\frac{\alpha\tau}{2}H_{even}} \tag{3.25}$$

and the constants α_1, α_2 are

$$\begin{aligned}\alpha_1 &= \frac{1}{4-4^{1/3}}\\ \alpha_2 &= 1-4\alpha_1\end{aligned}$$

As can be seen, these higher-order approximations require a greater number of exponentials, which introduces a tradeoff: each exponential on its own can be applied to the MPS in parallel, but the different exponentials must be applied sequentially; hence, the more exponentials in the approximating product, the slower the cost at any given step. On the other hand, since $\delta\tau$ can be taken to be larger, fewer steps may be necessary. Striking the right balance depends entirely on the needs of the problem at hand. The fourth order approximation given above also has the nice property that its terms are in a certain sense "symmetric," in that a sequence of U's is first applied and then applied in reverse. This symmetry can have beneficial properties for numerical stability [125].

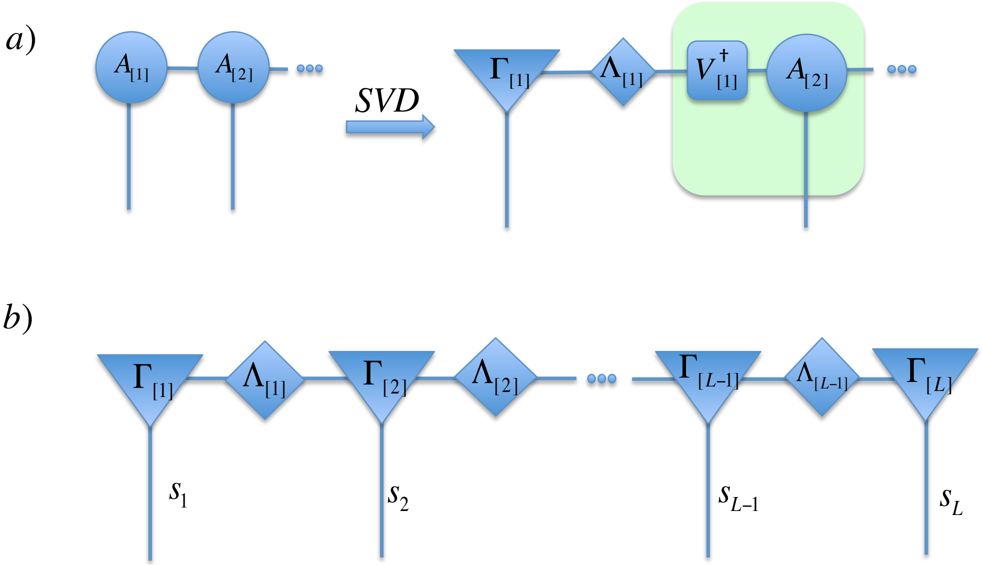

Figure 3.5: (Color online) (a) Graphical notation demonstrating the procedure for decomposing a typical matrix product state into the "Γ, Λ" form of Ref. [81]. The tensor $A_{[1]}$ is decomposed by SVD, with the resulting unitary matrix $V_{[1]}^{\dagger}$ "pushed" to the right to form a new tensor $A_{[2]}$, as highlighted in green. The new tensor $A_{[2]}$ is then similarly decomposed and the process iterates. (b) A complete MPS represented in Γ, Λ form. Note that the entanglement entropy across any bond j can be quickly computed by considering the entries of the diagonal matrix $\Lambda_{[j]}$.

3.2.1 Basic Algorithm

The TEBD algorithm essentially combines this Suzuki-Trotter technique with the truncated SVD. To evolve an arbitrary initial state towards the ground state, we efficiently apply a small imaginary time evolution operators $e^{-\delta\tau H}$ by means of Suzuki-Trotter approximation. For a Hamiltonian with interacting terms, this operation potentially increases the entanglement between neighboring sites, thereby increasing the Schmidt rank of any given bifurcation. This in turn would undesirably require an increasingly large bond dimension for our MPS representation, so to combat this, we consider the various spatial bipartitions of the state as the evolution operators are being applied and systematically perform truncated SVD operations so that the bond dimension can stay constant (or at least, grow only in a controlled manner which we can proscribe).

We consider first the application to a finite system of length L. As with DMRG, we initialize with a random starting state represented as an MPS. This time, however, it is beneficial to choose our state in a particular format. Because we will be repeatedly performing SVD decompositions between neighboring sites, one may as well begin with a state which is thusly decomposed. If a particular starting state is desired, one can decompose it via successive SVD's by the same process which one would employ to obtain the MPS representation in the first place [81]. This time, we wish to keep separate the unitary portion of the decomposition which contains the physical index and the diagonal matrix containing the Schmidt coefficients (the remaining unitary piece from site j can be "pushed" onto the tensor for site $j+1$; see Fig. 3.5). In other words, after appropriately grouping the physical and virtual indices, we take

$$A^{j_1} = \Gamma^{j_1}\Lambda_1 V_1^\dagger \tag{3.26}$$

and then group $V_1^\dagger$ onto A^{j_2} as $A^{j_2} \to V_1^\dagger A^{j_2}$. From here, we iteratively repeat these two operations until the entire state is represented by two families of tensors, $\{\Gamma^{j_1}_{[1]} \dots \Gamma^{j_L}_{[L]}\}$ and $\{\Lambda_{[1]} \dots \Lambda_{[L-1]}\}$. The reason for this grouping in the context of the algorithm will become subsequently more clear clear, although one can already foresee one advantage: the Schmidt coefficients (and by extension the entanglement entropy) across any choice of bipartition between sites i and $i+1$ are immediately accessible from the coefficients of $\Lambda_{[i]}$. The MPS representation for the state itself now takes the form:

$$|\psi\rangle = \sum_{s_1 \dots s_L} \Gamma^{s_1}_{[1]}\Lambda_{[1]}\Gamma^{s_2}_{[2]}\Lambda_{[2]} \dots \Gamma^{s_{L-1}}_{[L-1]}\Lambda_{[L-1]}\Gamma^{s_L}_{[L]}|s_1 s_2 \dots s_L\rangle \tag{3.27}$$

which is demonstrated graphically in Fig. 3.5. Here as elsewhere, we typically suppress labelling of the virtual indices and will explicitly label the physical spin index at site i as "s_i". When necessary for clarity, we will also include the bracketed label $[i]$ as part of the tensor name to distinguish different types of tensor, but recall that this label does not represent any kind of tensor index. Hence $\Gamma^{s_1}_{[1]}$ is a the rank-three tensor associated with site one; it has two virtual indices (not explicitly labelled) and one physical index, s_1. By contrast, $\Lambda_{[1]}$ is the diagonal matrix associated with site one; it has two implicit virtual indices and no physical index. Since each tensor is in fact a matrix for fixed value of its physical index, notation such as $\Gamma^{s_1}_{[1]}\Lambda_{[1]}$ is used to imply ordinary matrix multiplication in the virtual indices, i.e. that for a fixed value of s_1, the corresponding virtual indices between $\Gamma^{s_1}_{[1]}$ and $\Lambda_{[1]}$ are contracted.

Note that, if indeed we are merely choosing a random initial state, it is not necessary to first choose random tensors $A_{[i]}$ and subsequently perform this decomposition; rather, one can choose simultaneously a random set of tensors $\Gamma_{[i]}$ and random matrices Λ, subject to the constraint that $\sum_i \lambda_i^2 = 1$.

Recall that our goal is to apply the operator $e^{-\delta\tau H}$, which we will do by means of the Suzuki-Trotter approximation discussed above. Hence at each stage, we are applying not the global operator $e^{-\delta\tau H}$, but some local piece of the Suzuki-Trotter decomposition, such as perhaps $e^{-\delta\tau h_1 h_2}$, which acts only on the first two sites. Therefore, the central piece of the algorithm, which we shall subsequently call the "master step", consists of applying such a local operator to neighboring sites and then recovering an efficient MPS representation of the resulting state. Here first we consider the very common case where the Hamiltonian consists of 2-local terms, so that the master step involves applying two-site operator to a pair of neighboring sites (in principle, longer-range interactions could be applied by means of swap gates [27]). We present the method first, followed by a more detailed examination of the principle behind it.

Let us suppose we wish to apply a two-site operation U to sites j and $j+1$. We will now outline the procedure, which is also demonstrated graphically in Fig. 3.6.

We start by collecting up the tensors surrounding sites j and $j+1$, namely $\Lambda_{[j-1]}$, $\Gamma_{[j]}$, $\Lambda_{[j]}$, $\Gamma_{[j+1]}$, and $\Lambda_{[j+1]}$. If the outer tensors $\Lambda_{[j-1]}$ or $\Lambda_{[j+1]}$ fail to exist because we have reached the boundary of the system, they can be set to the identity. By contracting their virtual indices in sequence, these tensors group together to form a new tensor $\Theta_0^{s_1 s_2}$ with two physical indices open, i.e.

$$\Theta_0^{s_j s_{j+1}} = \Lambda_{[j-1]}\Gamma_{[j]}^{s_i}\Lambda_{[j]}\Gamma_{[j+1]}^{s_{j+1}}\Lambda_{j+1]}. \tag{3.28}$$

The evolution operator U can now be applied to the physical indices of Θ_0 to form a new tensor Θ,

$$\Theta = \sum_{s'_j s'_{j+1}} U^{s_j s_{j+1}}_{s'_j s'_{j+1}} \Theta_0^{s'_j s'_{j+1}} \tag{3.29}$$

which contains now information about the original state at sites i and $i+1$ under the action of the evolution operator. Care must be taken here with one's conventions to ensure consistency with respect to bras and kets; we have assumed our original MPS represented a ket $|\psi\rangle$, so we take the indices $s'_j s'_{j+1}$ of U to refer to its outgoing indices.

Note in fact (again see Fig. 3.6 that the resulting tensor Θ has two virtual indices (not labelled above) in addition to its physical indices: the leftmost virtual index of $\Lambda_{[j-1]}$ and the rightmost virtual index of $\Lambda_{[j+1]}$. Let us now temporarily give these virtual indices explicit labels, calling them v_{j-1} and v_{j+1} respectively. Then, grouping the indices v_{j-1} with s_j and v_{j+1} with s_{j+1},

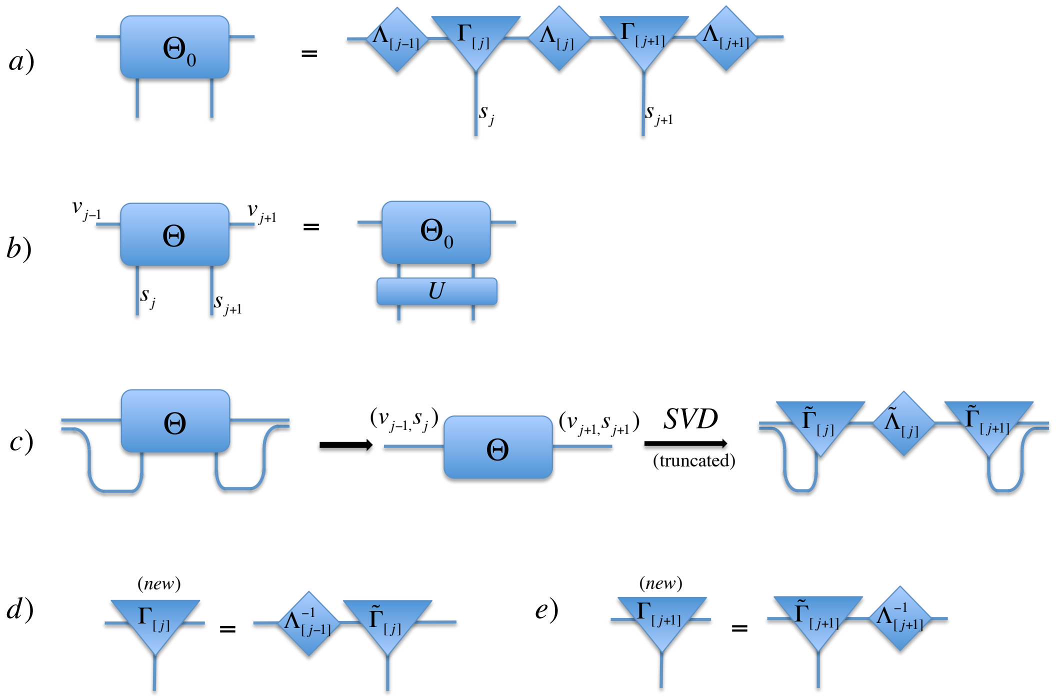

Figure 3.6: (Color online) Graphical notation outlining the "master step" of the TEBD algorithm for the case of a two-body Hamiltonian decomposed into two-body operators by Suzuki-Trotter. In a single step, one such two-body operator U is applied to sites j and $j+1$. (a) First, a tensor Θ_0 is constructed from the MPS tensors immediately surrounding the sites in question: $\Lambda_{[j-1]}$ $\Gamma_{[j]}$, $\Lambda_{[j]}$, $\Gamma_{[j+1]}$, and $\Lambda_{[j+1]}$. The tensors $\Lambda_{[j-1]}$ and $\Lambda_{[j+1]}$ are included to provide information about how this particular two-body subsystem relates to the "environment" formed by the rest of the chain. (b) Next, this tensor, which possesses both physical and virtual indices, is updated with the operator U to give a tensor Θ. (c) By grouping the virtual and physical indices as shown we can re-interpret Θ as a matrix. This matrix then undergoes a singular value decomposition (SVD), with the smallest singular values truncated away to prevent unwanted increase in the bond dimension of the MPS. (d) For the tensors resulting from the SVD, we "ungroup" the virtual and physical indices and again interpret them as a set of tensors $\tilde{\Gamma}_{[j]}$, $\tilde{\Lambda}_{[j]}$, and $\tilde{\Gamma}_{[j+1]}$. (e) Finally, we update our MPS with a new set of tensors which "undo" the inclusion of the environment tensors in (a), giving $\Gamma_{[j\text{ new}]} = \Lambda^{-1}_{[j-1]}\tilde{\Gamma}_{[j]}$ and $\Gamma_{[j+1\text{ new}]} = \tilde{\Gamma}_{[j]}\Lambda^{-1}_{[j+1]}$. The updated matrix $\Lambda_{[j]}$ is simply given by $\tilde{\Lambda}_{[j]}$ (not pictured).

we can interpret Θ as a matrix, and perform a singular value decomposition, yielding new tensors

$$\Theta^{(v_{j-1}s_j)}_{(v_{j+1}s_{j+1})} = \tilde{\Gamma}_{[j]}\tilde{\Lambda}_{[j]}\tilde{\Gamma}_{[j+1]}. \tag{3.30}$$

If the virtual indices of the initial MPS had dimension χ and the physical indices had dimension d, then the matrices in Eq. 3.30 are all of dimension $\chi d \times \chi d$ (we assume here equal bond dimensions throughout the chain for clarity, but this is not necessary and indeed may be undesirable in practice since the ends of the chain will likely be more weakly entangled. The generalization to site-dependent bond dimensions is straightforward). Our intention is to revert to interpreting $\tilde{\Gamma}_{[j]}$ and $\tilde{\Gamma}_{[j+1]}$ as rank-three tensors with a physical index, by "ungrouping" it from the composite matrix indices.

Notice, however, that this will leave a noticeably increased bond dimension between the two tensors. The matrix $\tilde{\Lambda}_{[j]}$ has grown in size by a factor of d, which is to be expected, given that the application of U will in general have created some entanglement between the sites. But this increase in bond dimension would not be sustainable given that we intend to apply many copies of U in order to evolve out to large imaginary time. Thus, it is here we employ the second tool in the TEBD toolbox, truncating the SVD by removing the smallest elements of $\Lambda_{[j]}$. It is sensible also at this point to manually enforce $\sum_i \lambda_i^2 = 1$; this helps to ensure that the state remains numerically well-conditioned.

The number of elements to remove during this truncation can depend somewhat on the needs of the problem at hand; a common choice is to remove all but the χ largest singular values, so that in fact the bond dimension does not increase at all. Alternatively, sometimes a small, controlled increase (such as allowing the bond dimension to increase by one or two every time a gate is applied) may be desirable if the number of gates to be applied is small. A check can also be performed at this stage by computing the total of all the singular values which have been truncated away; this number should remain small if the TEBD algorithm is going to yield a good approximate ground state. We note that this is also an interesting point of comparison between TEBD and DMRG. In DMRG, the effect of the finite bond dimension was somewhat obscured; we minimized the energy over all states which could be represented by a given bond dimension, knowing implicitly that the result would be an imperfect approximation if the bond dimension was too small, but there was no natural way to immediately quantify this error, or to optimize the bond dimension beyond blind trial and error. In TEBD, however, the impact of our chosen bond dimension can be directly quantified, and one can exercise considerably more control over this behavior. For example, one might allow the bond dimension of the system to increase slightly at a given step if the alternative is an unacceptably large truncation error.

At this point we can also see why we have used the Γ, Λ representation of the MPS, and why we have included the matrices $\Lambda_{[j-1]}$ and $\Lambda_{[j+1]}$ in the construction of Θ_0 (Eq. 3.28). These matrices are sometimes called "environment tensors," because they contain information about the entanglement between the two sites we are currently updating and their relationship to rest of the chain (their "environment"). Including this information allows us to do our truncation in an optimal way, because the relative weights of the singular values in $\Lambda_{[j-1]}$ and $\Lambda_{[j+1]}$ will be included when we do the SVD in Eq. 3.30. In this way our truncation tends to discard information which is neither significant in terms of the entanglement between sites j and $j+1$, nor significant with respect to the larger system. Note, since we are considering a finite system, the fully optimal way to incorporate this environment information would be to include *all* of the tensors from the surrounding portions of the state, providing complete information. But including the entire state as environment is quite numerically costly, and in practice, the information contained in the Λ matrix is typically quite sufficient.

We now have new, truncated matrices $\tilde{\Gamma}_{[j]}$, $\tilde{\Lambda_{[j]}}$, and $\tilde{\Gamma}_{[j+1]}$, which collectively reflect the action of U. All that remains is to interpret in the context of the original MPS; that is, we want to recover tensors $\Gamma_{[j \text{ new}]}$, etc., which can take the place of $\Gamma_{[j]}$ in our updated MPS. The matrix $\Lambda_{[j \text{ new}]}$ will simply be the matrix $\tilde{\Lambda}_{[j]}$; it is already in the correct, updated form. For the Γ tensors, two adjustments need to be made. First, the composite matrix indices must be carefully "ungrouped" so that we can treat the tensor as a rank-three object with two virtual indices and a physical index. For example, the truncated matrix $\tilde{\Gamma}_{[j]}$ has dimension $\chi d \times \chi$; the larger incoming index is still a composite of the χ-dimensional virtual index v_j and the d- dimensional physical index s_j. A similar statement is true for the outgoing index of $\Gamma_{[j+1]}$. After thusly reshaping our interpretation of the index groupings, it only remains to adjust for the fact that we explicitly included the matrices $\Lambda_{[j-1]}$ and $\Lambda_{[j+1]}$. These matrices were necessary to take into account information about the environment, but they themselves are not being updated during this current step. To undo their inclusion, we thusly take

$$\Gamma_{[j \text{ new}]} = \Lambda^{-1}_{[j-1]} \tilde{\Gamma}_{[j]} \tag{3.31}$$

and

$$\Gamma_{[j+1 \text{ new}]} = \tilde{\Gamma}_{[j]} \Lambda^{-1}_{[j+1]} \tag{3.32}$$

Note numerically that these inverses are trivial to calculate because the matrices are diagonal, although in practice, it will be desirable to use a cutoff value below which tiny numerical values are set to zero, and to use only a pseudoinverse, computed over the nonzero values of the matrix, which could otherwise be singular. Replacing the tensors $\Gamma_{[j]}$, $\Lambda_{[j]}$ and $\Gamma_{[j+1]}$ with $\Gamma_{[j \text{ new}]}$, $\Lambda_{[j \text{ new}]}$ and $\Gamma_{[j+1 \text{ new}]}$ completes the "master step" of the algorithm.

Having updated these tensors, we proceed to the next pair of sites in the current layer of evolution (for the two-body operator case considered here, this will be an update to the sites $j+2$ and $j+3$). Updates continue in this fashion until all the evolution operators in the layer have been applied across the chain, and layers are applied repeatedly until the desired amount of time evolution or imaginary time evolution has been achieved. In summary, the general outline of the algorithm is as follows:

1. Decompose the imaginary time evolution operator $e^{-\delta\tau H}$ according to one's desired choice of Suzuki-Trotter approximation; the result will be n "layers" which need to be applied sequentially.

2. Initialize with a random state $|\psi_0\rangle$ in Γ, Λ form

3. For each of the n layers of the evolution operator, sweep over neighboring pairs of sites in the chain. At each pair, the relevant evolution operator is applied to the pair in accordance with the "master step", Eqs. 3.28-3.31 and Fig. 3.6. This will update the two tensors $\Gamma_{[j]}$ and $\Gamma_{[j+1]}$, as well as the matrix $\Lambda_{[j]}$ which lives between them. Once all pairs have been updated, we proceed to the next layer of the evolution operator

4. When all layers have been applied, the new MPS now represents $e^{-\delta\tau H}|\psi_0\rangle$. Step 3 is iterated on the new state, until the desired amount of imaginary time evolution has been performed.

TEBD and Canonical Form

Finally, we remark that, in addition to the explicit analytical methods referenced above [15, 83], it has been shown that successive Schmidt decompositions of the state during an iTEBD algorithm are themselves equivalent to enforcing canonical form, so long as the operators being applied to the state are unitary. Of course, when one computes a ground state using imaginary time evolution, the operators which are used, of the form $e^{-\delta\tau H}$ (see Eq. 5.24), are not in general unitary. But for $\delta\tau$ very small, they will be quite close. Since a typical iTEBD algorithm ends with a sequence of very small time step evolutions, the resulting states are also typically "close" to canonical form [85]. To take this to its logical extension, it is a good practice to terminate every

TEBD algorithm with e.g. 100 steps of evolution in which we apply only the identity gate (which corresponds to the exact $\delta\tau = 0$ limit), Of course, this identity gate evolution is both explicitly unitary and incapable of changing the underlying state. In this way, one can ensure that the states computed via iTEBD algorithm are exactly in canonical form [84] (up to numerical precision) without the need to introduce additional numerical methods.

3.2.2 Two-Site iTEBD

Naively, one might think that extending this finite-system technique to the thermodynamic limit would involve greater difficulty, or at least (as is the case with iDMRG) some subtle re-imagining of the technique. But when our system exhibits some form of translational invariance, we can exploit this symmetry to extend the TEBD algorithm to the infinite case in a very natural way. Indeed, the process becomes potentially simpler and more numerically stable without the need to carefully consider the system boundaries. The infinite TEBD or "iTEBD" algorithm [28] (whose name was curiously coined just after the start of Apple's iProduct craze) has become perhaps even more popular than its finite older brother, and may be more accurate at determining ground states than other proposals for studying infinite systems [32]. Indeed, in subsequent chapters, we have generally employed the DMRG algorithm to study finite systems and the iTEBD algorithm for systems in the thermodynamic limit.

Recall from the earlier discussion of translationally-invariant matrix product states that we can represent an MPS simply by the tensors which form a "unit cell." A one-site translationally invariant state can therefore be represented with a single tensor A, with the understanding that the same tensor is associated with every site in the chain, $A_{[j]} = A \ \forall \ j$. When this symmetry exists (both in the state and in the parent Hamiltonian) we can therefore consider even the behavior of infinite systems, as represented by their unit cell.

Two-Site Unit Cell

Since it represents the simplest case and also one of the most common for practical problems, let us consider first the case of a Hamiltonian which is a sum of two-body interactions. In such a case, it will turn out to be necessary for the iTEBD algorithm that we work with a two-site representation of our state. This necessity arises even when we have reason to believe that the ground state we wish to calculate will have a one-site translation invariance, a point which will turn out to have particular significance to our work in Chapter 5. For now, however, we simply note the source of this requirement: the Suzuki-Trotter decomposition of this Hamiltonian will contain two different types of terms:

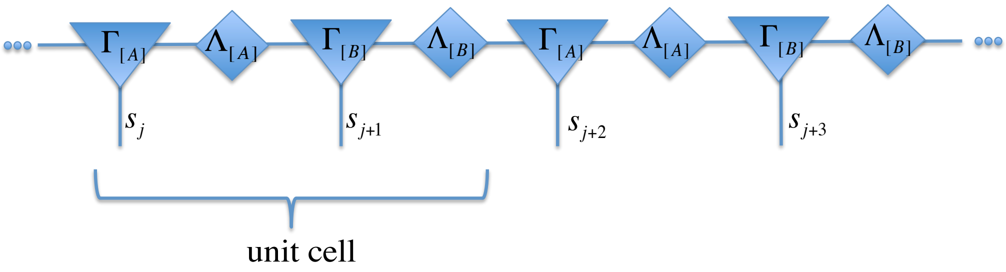

Figure 3.7: (Color online). A graphical representation of a translationally invariant infinite state presented using a two-site unit cell of tensors, $\{\Gamma_{[A]}, \Lambda_{[A]}, \Gamma_{[B]}, \Lambda_{[B]}\}$. These tensors may be chosen in a two-site representation because the state possesses only two-site translation invariance. But even if the state posesses translational invariance on the one-site level, this format is necessary for the iTEBD algorithm described below.

those acting on odd sites first, and those acting on even sites first. In order to consider the difference in the action of these operators on our state, we must be able to distinguish between "odd sites" and "even sites," which is not possible unless there are at least two sites in our MPS representation.

Consequently, we imagine our initial state to be represented completely by the set of tensors $\{\Gamma_{[A]}, \Lambda_{[A]}, \Gamma_{[B]}, \Lambda_{[B]}\}$, where A and B distinguish the odd and even sites, and we have employed the same Γ, Λ decomposition of the tensors described above. The structure of the state (and the subsequent steps) are depicted in Fig. 3.7.

The assumed structure and translation invariance of the Hamiltonian means that it decomposes as

$$H = \sum_i h_{[i,i+1]}^{s_i s_{i+1}} \tag{3.33}$$

where, $h^{s_i s_{i+1}}$ is some operator which acts on two spins at once, and we have again used the bracketed subscript notation to show the site where the operator is applied. Unlike the notation "$h_i^{s_i s_{i+1}}$", which might imply a collection of different operators which are index by i, the salient point here is that terms like $h_{[i,i+1]}$ and $h_{[j,j+1]}$ differ only in the sites at which they are applied.

Thus, all evolution operators in the Suzuki-Trotter decomposition have the form $U_A(\alpha) = e^{-\alpha\delta\tau h}$ for some two-body operator h and some coefficients α

(which depend upon the choice of Suzuki-Trotter and may differ depending on the "layer" of the decomposition). But while their form is identical, what differs is the nature of the tensors they act upon. We have assumed also an MPS represented by two types of tensor, "A" type at odd sites, "B" type at even sites (see Fig. 3.7). The entire Hamiltonian can therefore be understood as a sum of identical terms applied to sites in the order "AB", and a similar sum of terms applied to sites in the order "BA."

It follows that all the desired imaginary time evolution can be performed using only two different instances of the master step described above. Either, we apply $U(\alpha)$ to the sequence of tensors $\{\Lambda_{[B]}, \Gamma_{[A]}, \Lambda_{[A]}, \Gamma_{[B]}, \Lambda_{[B]}\}$, or we apply it to the tensors $\{\Lambda_{[A]}, \Gamma_{[B]}, \Lambda_{[B]}, \Gamma_{[A]}, \Lambda_{[A]}\}$. This is depicted in Fig. 3.8; it is perhaps instructive to compare this to the general master step in Fig. 3.6 above. In any given step, either the matrix $\Lambda_{[A]}$ is being updated and truncated, with two matrices $\Lambda_{[B]}$ serving as the environment tensors, or vice verse.

Hence, the iTEBD algorithm, alternating between only two types of steps which each follow the same master pattern, truly takes advantage of the parallelization offered within the layers of a Suzuki-Trotter decomposition. The result is a robust and computationally efficient algorithm which allows the infinite, thermodynamic limit of a system to be studied without the need for finite size scaling.

3.2.3 L-Site iTEBD

By far the most common applications of iTEBD are to Hamiltonians with two-body terms, but the underlying principle is not so limited. One can certainly imagine cases where an algorithm based around a two-site unit cell would be insufficient, such as when studying a system like the cluster state [91], whose parent Hamiltonian contains three-body terms, or when studying systems whose ground state translation symmetry may be spontaneously broken beyond the two site level. A more subtle motivation for such an algorithm will arise for us in Chapter 5, and hence we wish to give a sensible, general version of the algorithm which can pursue the ground state of an infinite system with an arbitrary, L-site unit cell.

In the case of the system with a two-site unit cell, we showed above that each two-body evolution operator is essentially identical, and what changes is merely whether they are applied to MPS tensors which are in the order "AB" or "BA." Naturally, a similar statement is true for the case of an L-site unit cell; now, we will have L different types of tensor, and still only a single type of L-body evolution operator. The different layers of the Suzuki-Trotter decomposition will correspond to applying this to L different cyclic permutations of the order of the tensors: $(1, 2, \ldots L)$, $(2, 3, \ldots L, 1)$, $(3, 4, \ldots, L, 1, 2)$, etc. Note

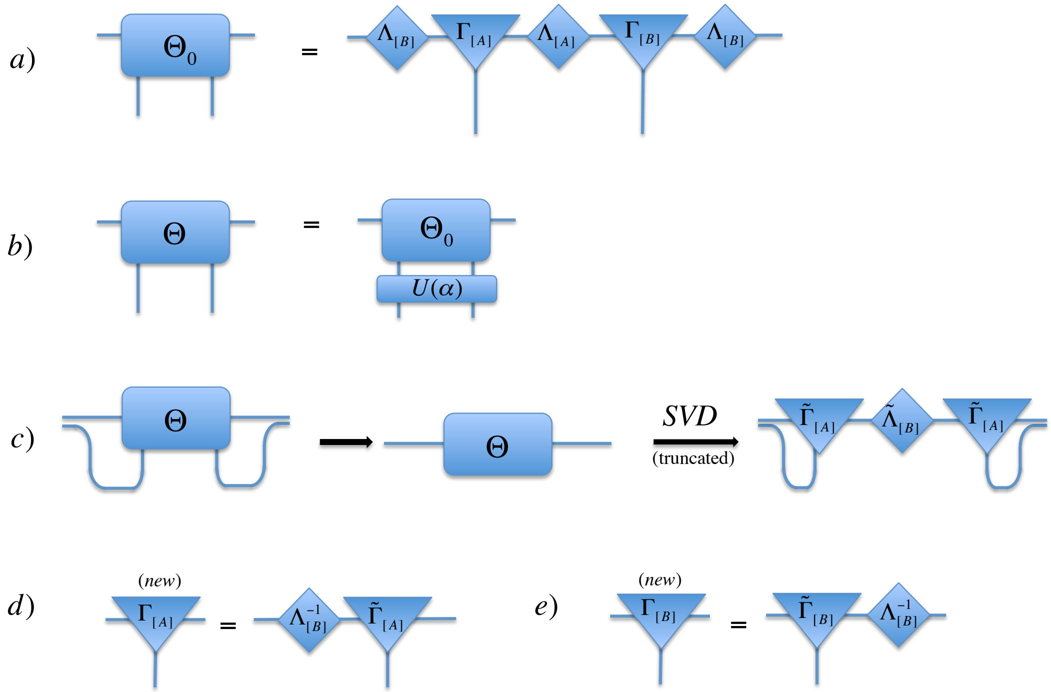

Figure 3.8: (Color online) Graphical notation for an example of the "master step" of the iTEBD algorithm. Compare to the case of a finite TEBD algorithm in Fig. 3.6. The key distinction is simply that, because of the translational invariance of the state, the environment tensors included in (a) are both given by $\Lambda_{[B]}$. Note that a full iTEBD algorithm will alternate between steps acting first on odd sites and steps acting first on even sites. The latter case corresponds to simply exchanging the labels A and B everywhere in this diagram.

that the two-term Suzuki-Trotter decomposition contemplated above easily extends for the L-site case by simply "nesting" a sequence of $L-1$ two-term decompositions.

Hence, just like the two-site case, we need define only one "master step," which we will subsequently apply to $L-1$ different cyclic permutations of the tensors. For clarity, we outline the steps here with respect to the tensors in their "natural" order $(1, 2, \ldots L)$, as the generalization from there is quite simple.

As before, we begin by grouping the tensors around the sites which are to be acted upon, namely

$$\Theta_0^{s_1 s_2 \ldots s_L} = \Lambda_{[L]}\Gamma_{[1]}^{s_1}\Lambda_{[1]} \ldots \Gamma_{[L]}^{s_L}\Lambda_{[L]}. \tag{3.34}$$

The evolution operator is then applied to this tensor, which has L open indices corresponding to the L physical degrees of freedom. This gives

$$\Theta = \sum_{s'_1 \ldots s'_L} U^{s_1 \ldots s_L}_{s'_1 \ldots s'_L} \Theta_0^{s'_1 \ldots s'_L}. \tag{3.35}$$

Still following the previous cases, we now set out to recover updated tensors Γ and Λ by singular value decomposition. In this case, all of the tensors in the cell save for $\Lambda_{[L]}$ will be updated, as all have been potentially entangled with their neighbors. But because there are now so many tensors, a single SVD will not do the trick. Instead, we iteratively apply the following steps (see Fig. 3.9).

First, we interpret Θ as a matrix by grouping the left virtual index v_1 and the first physical index s_1 as the incoming index, and the right virtual index v_L with the remaining physical indices as the outgoing index. In this perspective, the dimensions of the matrix are $\chi d \times \chi d^{L-1}$. Performing an SVD of this matrix and making our desired truncation gives

$$\Theta^{(v_1 s_1)}_{(v_L s_2 \ldots s_L)} \to \tilde{\Gamma}_{[1]}\tilde{\Lambda}_{[1]}\tilde{\Theta}_{[2\ldots L]}. \tag{3.36}$$

The truncated matrix $\Lambda_{[1]}$ will be the updated matrix for the first site $\Lambda_{[1 \text{ new}]}$. And as before, we ungroup the physical and virtual indices from the incoming index of $\tilde{\Gamma}_{[1]}$, and apply the inverse of matrix we used to represent the environment to recover our updated tensor, i.e.

$$\Gamma_{[1 \text{ new}]} = \Lambda^{-1}_{[L]}\tilde{\Gamma}_{[1]}. \tag{3.37}$$

Of course, at this point we are *not* in a position to recover the updated form of any other tensors; all we have left to work with is the matrix $\tilde{\Theta}_{[2\ldots L]}$, which represents the remainder of the state's unit cell. It is tempting to simply

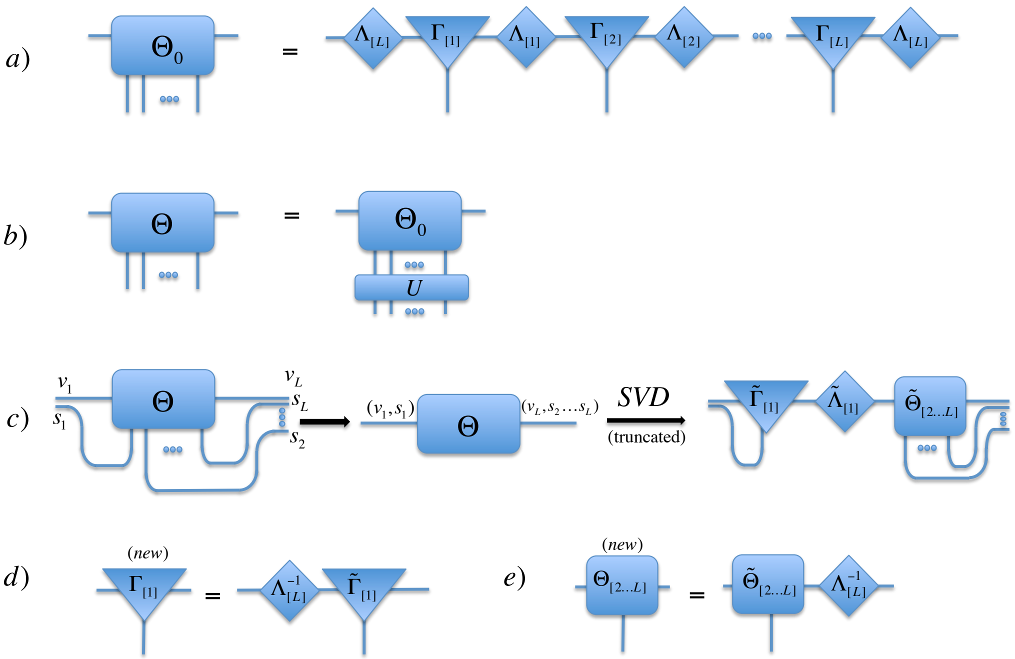

Figure 3.9: Graphical notation for an example of the "master step" of the iTEBD algorithm generalized to an arbitrary unit cell of length L (compare to the case of a finite TEBD algorithm in Fig. 3.6 and the special two-site case in Fig. 3.8). The principle difference from the two-site case occurs here in step (c). The tensor Θ will possess L physical indices, and we perform our SVD by grouping only the first index to the left, and the remaining indices to the right. In steps (d) and (e), the result of the SVD gives updated tensors for $\Gamma_{[1]}$ and $\Lambda_{[1]}$, and also gives us an updated version of the tensor Θ which now has only $L-1$ physical indices. The steps (c) through (e) are repeated with this new tensor Θ until only two physical indices remain, at which point we decompose Θ into the remaining tensors by the normal two-step process. Note that, as in the two-site case, the full implementation of this algorithm involves applying operators which act on the sites of the unit cell in different sequences, which all differ by circular permutation. Illustrated here the algorithm acts first with an operator on site 1. The remaining steps look the same, but with the labels circularly permuted so that $1 \to 2, 2 \to 3 \ldots L \to 1$.

perform another SVD on this matrix, separating out the second physical index s_2 and repeating the steps from Eq. 3.36 onward. And indeed, doing so does produce a workable version of the algorithm. But it is not optimal. After all, the tensor $\tilde{\Theta}_{[2\ldots L]}$ no longer contains the most up-to-date information about its surrounding environment. This information is now contained in $\Lambda_{[1 \text{ new}]}$, and as such, performing a truncation base upon its SVD does not optimally reduce the MPS rank until we take it into account. Therefore, we first make the transformation

$$\tilde{\Theta}_{[2\ldots L\text{new}]} = \Lambda_{[1 \text{ new}]}\tilde{\Theta}_{[2\ldots L]} \tag{3.38}$$

and *then* proceed to further decompose this matrix, separating off one more physical index at a time under Eq. 3.36. This process iterates a total of $L-1$ times to reach the final SVD, at which point the remaining decomposition is simply identical to the standard two-site iTEBD decomposition, giving

$$\Theta^{(v_L s_{L-1})}_{(v_R s_L)} \to \tilde{\Gamma}_{[L-1]}\tilde{\Lambda}_{[L-1]}\tilde{\Gamma}_{[L]}. \tag{3.39}$$

To complete the process, simply take

$$\Gamma_{[L-1 \text{ new}]} = \Lambda^{-1}_{[L-2]}\tilde{\Gamma}_{[L-1]} \tag{3.40}$$

and

$$\Gamma_{[L \text{ new}]} = \tilde{\Gamma}_{[L]}\Lambda[L]^{-1}, \tag{3.41}$$

so that we have performed a complete update of all tensors except $\Lambda_{[L]}$. At this point, we circularly shift our tensors one spot to the left, so that they are ordered $(2, 3, \ldots L, 1)$, and repeat the process.

This L-site generalization of the iTEBD algorithm has proven for us to be surprisingly stable in most cases (we have tested up to six sites) but it must be used with care. The larger the unit cell, the more opportunity for accumulated error from the truncations, or from any numerical noise which is introduced. This is particularly relevant for real-time evolution; when used for imaginary time evolution these potential instabilities will likely be overcome by the power of the exponential suppression of everything but the ground state, though the convergence may be erratic and must be checked carefully. In both cases, however, there is potentially an even more important concern: the choice of unit cell must be commensurate with the symmetry properties of the anticipated ground state, or the algorithm will not be able to converge a physically meaningful result. For example, imagine a system whose ground state displays an alternating magnetization, perhaps polarized "up" in some basis for odd-numbered sites and "down" for even sites. Attempting to compute such a ground state with a three-site iTEBD cell will be impossible. Furstration-like effects will appear as the algorithm circularly permutes the sites, with the three types of tensors constantly shifting back and forth between being treated as representing odd and even sites. Note that such a system *could*, however, be studied in principle by an algorithm with a six-site unit cell. Though this would be unlikely to be of use in practice, it illustrates the point that different size unit cells must be used and compared when one does not know the translational symmetry properties of the ground state.

3.2.4 Two Dimensional Systems

Finally, we note that the steps of the TEBD and iTEBD algorithms outlined above can also be applied to systems in two dimensions and higher, with some important modifications. In two dimensions, for example, one can consider systems on many different forms of lattice. These and other complications introduce substantial additional considerations, and a full treatment of their complexities is beyond the scope of this work; we refer the reader instead to Refs. [33, 84, 126–129] (note that in the literature, the term "iPEPS" is used both to refer to an infinite two dimensional tensor network state, and to the particular class of TEBD-like algorithms which use them to compute the ground state). For completeness, however, and because we will be employing a basic version of the TEBD algorithm in Chapter 4, we will briefly comment on the basic changes that need to be made for the case of a two-dimensional system on a square lattice. First and foremost, the Suzuki-Trotter approximation to the evolution operator will require more layers. For example, in the one dimensional case we partitioned the terms of the Hamiltonian into two classes, those acting first on an odd site, and those acting first on an even site. A similar approach can be used for a two dimensional system, but *four* classes will be needed to allow for mutually commuting terms: two classes for the odd/even sites in the horizontal direction, and two similar classes for the vertical direction. In other words, we first divide the Hamiltonian into two sets of terms

$$H_{2D} = \sum_{(i,j)} h_{(i,j)(i+1,j)} + \sum_{(i,j)} h_{(i,j)(i,j+1)} \qquad (3.42)$$

where we have used i and j to label the horizontal and vertical location of the sites, respectively. Each set of terms can then be further broken down as:

$$\begin{aligned}H_{2D} = \sum_{(i\ \text{odd},j)} h_{(i,j)(i+1,j)} + \sum_{(i\ \text{even},j)} h_{(i,j)(i+1,j)} \\ + \sum_{(i,j\ \text{odd})} h_{(i,j)(i,j+1)} + \sum_{(i,j\ \text{even})} h_{(i,j)(i,j+1)}.\end{aligned} \tag{3.43}$$

We then then perform a Suzuki-Trotter decomposition for all for sets of terms (to do this at the simplest level, the Suzuki-Trotter approximation outlined above for two classes of terms can be recursively "nested"). As with the case for one-dimensional systems, we are left with layers of evolution operators to apply to the state.

Our PEPS representation for the two dimensional system also looks similar to the Γ, Λ decomposition used in one dimensional TEBD, though with additional Λ matrices arising from the fact that we must consider bifurcations in both the vertical and horizontal directions to separate the classes of commuting terms. Since the evolution operators still act only pairwise, at first glance the "master step" for updating tensors may appear unchanged. However, recall the importance of using the neighboring Λ tensors to form an "environment" which allows us to more efficiently and accurately perform the truncation in the one dimensional case. It is here that the algorithm becomes substantially more complicated in the two dimensional case. Naively, not only do we wish to include the two environment tensors which appear along the same axis as the current operator (vertical or horizontal), but there are four additional tensors along the orthogonal axis which should be considered (see Fig. 3.10). Worse, these six tensors alone turn out to be insufficient to obtain optimal information about the environment, as they will not all be fully updated at the same time, and the two dimensional environment depends on more than merely the boundary with the states being updated [84].

The increased complexity of considering the environment has caused the world of 2D TEBD algorithms to break into essentially two approaches. In the first case, broader environmental effects can simply be ignored, using only local information to inform the truncations when the evolution operators are applied. These approaches (see [126, 127]) are typically termed "simple update" methods, and the name is fitting– although they have had some success for finite states and relatively simple systems, they have generally been shown to have stability issues or trouble faithfully reproducing ground states of more complex Hamiltonians [129]. The other approach, particularly in the case of infinite systems is to use the entire surrounding state as the environment, but

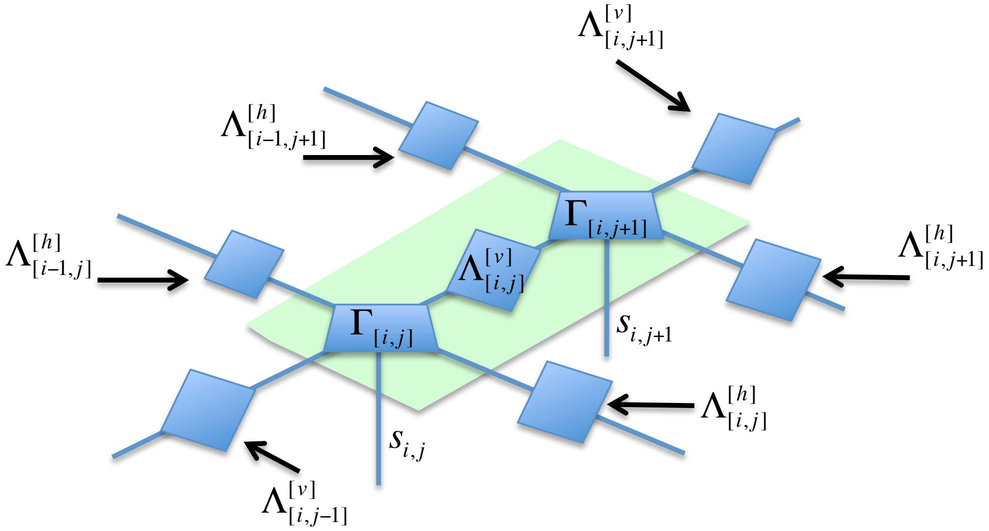
$\Lambda^{[v]}_{[i,j+1]}$
$\Lambda^{[h]}_{[i-1,j+1]}$
$\Lambda^{[h]}_{[i-1,j]}$
$\Gamma_{[i,j+1]}$
$\Lambda^{[v]}_{[i,j]}$
$\Lambda^{[h]}_{[i,j+1]}$
$\Gamma_{[i,j]}$
$s_{i,j+1}$
$\Lambda^{[h]}_{[i,j]}$
$s_{i,j}$
$\Lambda^{[v]}_{[i,j-1]}$

Figure 3.10: (Color online) An arbitrary portion of two dimensional tensor network state (PEPS) on a square lattice, decomposed into Γ, Λ form for a TEBD-type algorithm. The tensors which will be updated during a TEBD step are highlighted in green. By contrast to the one-dimensional case, note the following. First, note that the unit cell for such a system will likely have many more free virtual indices, with the exact number depending on the coordination number of the lattice. This, along with the difficulties involved in contracting two-dimensional tensor networks, places sharp restrictions on the bond dimensions which can be used. Unlike the finite cases, where bond dimensions typically range into the hundreds, many PEPS systems must be computed with a bond dimension of only two or three. Additionally, there are now two types of Λ associated with each site, one in the horizontal direction $\Lambda^{[h]}$, and one in the vertical $\Lambda^{[v]}$ (we have used the convention that the Λ's are labelled by the site below them and to the left). Next, note that at a minimum, six matrices Λ must be included to provide information about the "environment" of this cell; in fact, as commented in the text, even this is not sufficient to give complete information in two dimensions. Finally, observe that in addition to an alternating sequence of operators which will be applied to a cell like this (acting first on the odd site, then acting first on the even site, etc) we will also need to consider a *different* unit cell in the horizontal direction, and perform updates to that as well in order to achieve a complete update of the whole system.

compute the environment tensors through some sort of approximating contraction scheme to reduce the costs. These methods, called “full update” methods[84, 129], appear more promising, but can involve costly calculations and are sometimes limited to systems with particular symmetry or other properties. A possible blended approach termed “cluster update” has also been proposed [128].

In Chapter 4 we shall be considering a simple two-dimensional Ising model on a square lattice and performing finite-size scaling as a proof-of-principle for a more general technique. For these purposes, it will be sufficient to employ a simple update scheme specialized to the case of square lattices.

3.2.5 Comments on Convergence Schemes

When finding a ground state in an algorithm such as TEBD or iTEBD (and indeed also DMRG and iDMRG), the function of each iteration is to rotate the current state vector closer to an eigenstate with a lower energy. It is therefore quite natural and typical to define the halting condition of such scheme by reference to the energy of the current state. In the simplest instance, for example, one might set a convergence threshold of ϵ and then, after each step i of the algorithm, compute the resulting change in the state's energy, $\Delta E = E_i - E_{i+1}$, with the understanding that the iteration will halt if ever $\Delta E \leq \epsilon$. A more sophisticated version of the same idea would be to compute the standard deviation for the N most recent steps instead, to make the process less susceptible to small numerical fluctuations.

Depending upon the quantities of interest, however, and the desired levels of precision, this type of convergence framework may not be optimal. For one thing, as discussed above, one must be careful of instance of “metastability” in the energy values, which may produce periods of time during the algorithm where the energy is changing very slowly and yet is still far from the true ground state value. This, in part, is the reason why a typical iTEBD convergence scheme consists of a set of sequential iterations, each time with a smaller imaginary time step $\delta\tau$ [28], since it is quite common to see the energy values experience a greater change after the step size is reduced (see Fig. 3.11). Even with decreasing step sizes, however, it is possible for the changes in the energy to stall into metastability.

In such a case, it can be helpful to consider the variance of the energy, $\langle \Delta H^2 \rangle = \langle H^2 \rangle - \langle H \rangle^2$. This quantity measured the quantum fluctuation of the energy in the state, and hence in a metastable region, where the current numerical state is not a true eigenstate of the Hamiltonian, we expect a large value. When it is small, we can rule out the possibility that the current energy is far from that of an eigenstate, and the exponential suppression of excited

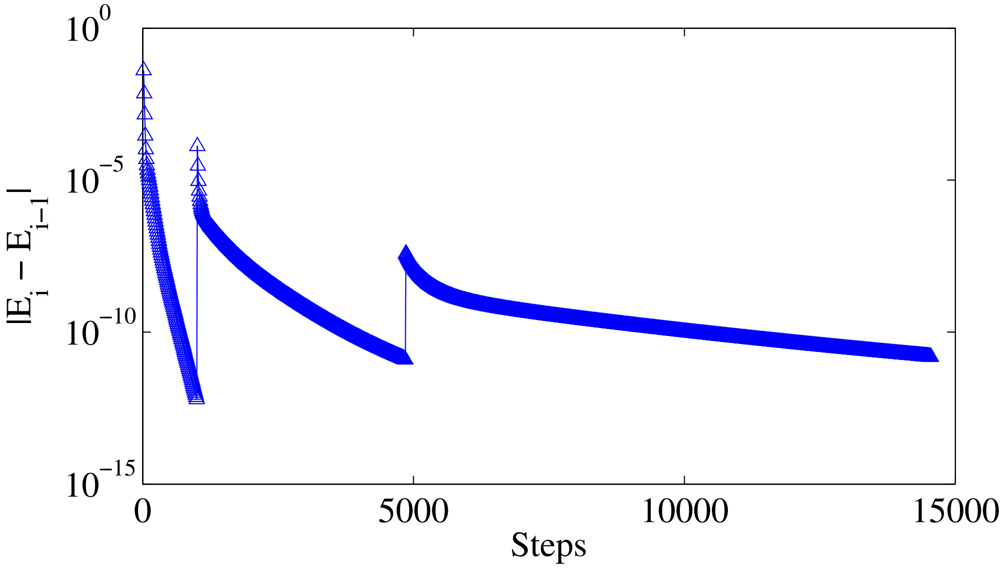

Figure 3.11: A test of the energy convergence behavior during the iTBED algorithm, as applied to the spin-1 Heisenberg model with a transverse field. After each step, the difference in energy compared to the previous step is plotted. The segments separated by sharp transitions indicate the different stages of the algorithm, with the size of the imaginary time step, $\delta\tau$, decreased for each subsequent stage. Because the changes in the energy can become so small during the course of any individual stage, one might erroneously believe that convergence had been achieved, but moving to a smaller time step reveals that this is not so.

states by the iTEBD algorithm ensures that if we are in *an* eigenstate, that eigenstate should be the ground state. A method to efficiently compute $\langle \Delta H^2 \rangle$ at little additional computational cost will be given in Chapter 4

It is also important to consider, however, that not all quantities will converge at the same rate. In Fig. 3.12, for example, we show the results of a study of the spin-1 Heisenberg model with a relatively large transverse field. We consider the observables E and $\langle S_x \rangle$, as well as the entanglement entropy S (note that the state must be put in canonical form before the exact entanglement entropy can be directly calculated from the tensors). After every 100 steps of evolution, we compute for each quantity the coefficient of variation (the standard deviation divided by the mean, $\sigma(X)/\bar{X}$) over a sample of the previous 300 steps, to determine how much the quantities are currently being "changed" by the algorithm. Since these coefficients of variation are dimensionless, different quantities can be compared directly in terms of their

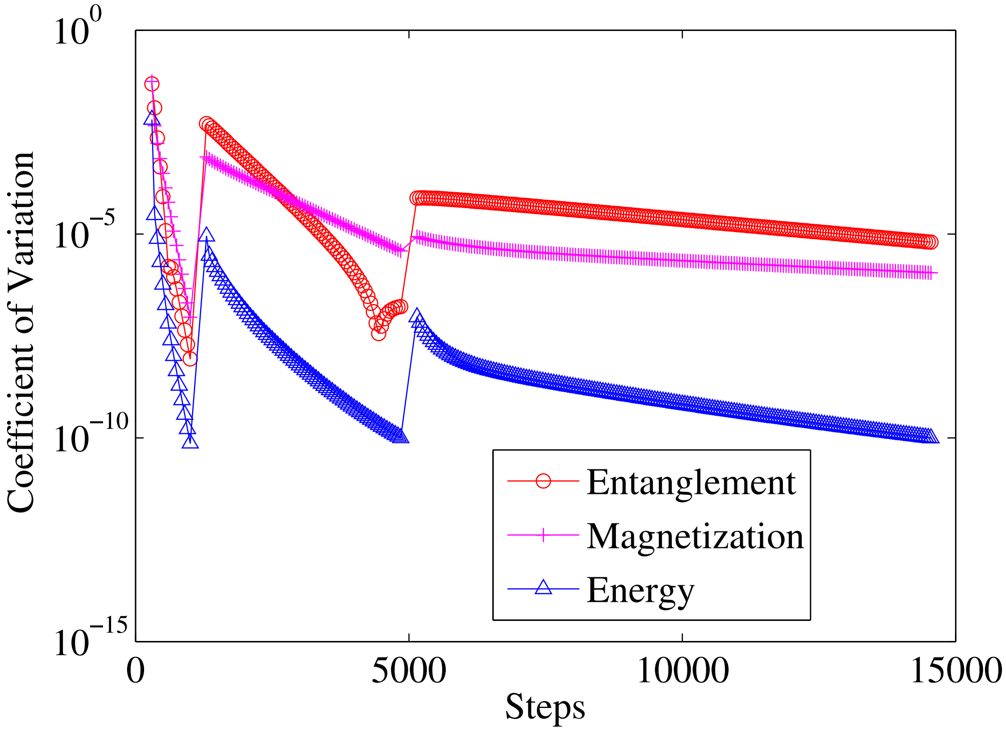

Figure 3.12: (Color Online) A test of the convergence behavior of E, S, and $\langle S_x \rangle$ during the iTBED algorithm, as applied to the spin-1 Heisenberg model with a transverse field. Every 100 steps, the previous 300 steps are sampled and the standard deviation is computed, and then divided by the mean. The resulting coefficient of variation is plotted in the figure above. The curve for the energy is consistently much lower, indicating that this quantity converges much more quickly than the others.

convergence. As can be seen in the figure, the number of steps sufficient to achieve a stable value of the energy out to a high precision is not at all sufficient to achieve the same accuracy in the other quantities. A study which focused on the magnetization of ground states, for example, might be better off using the stability of the magnetizations themselves, rather than the energies, to certify convergence.

Unfortunately, it does not seem to be the the case that any one particular quantity is universally more difficult to converge than the others; instead, the relative convergence rates tend to depend on the Hamiltonian being studied. Therefore, when precision is required in more than one observable, One option is to to perform a small convergence study first, to determine which quantity will produce the strictest convergence across the board. We do note, however, that since algorithms like iTEBD operate directly on the energy of the state, the energy is likely to be among the *fastest* quantities to converge, making it a poor choice for a convergence criterion if other quantities are desired as well. Interestingly, however, in our experience the same is not true for the second cumulant of the energy, $\langle \Delta H^2 \rangle$, which is often (though likely not always) slower to converge than the energy (see for example Fig. 3.13) and which frequently seems not to converge until other observables like the magnetizations have converged as well. Since this quantity can also be efficiently computed using an MPO or a cumulant generating function, it may be a good convergence criterion for a variety of models.

Finally, having chosen which quantity or quantities to monitor for convergence, it remains to decide which sequence of imaginary time steps should be used. Here, too, the optimal choice may depend on the model being considered, and perhaps also the bond dimension of the MPS representing it. In general, however, the precise set of steps sizes chosen is likely less important than it may initially appear. To see this, consider the possible consequences of a poor choice of convergence scheme. Either the evolution may "waste time" with too large a time step if it winds up in a plateau, or, if the time steps become too small too quickly, it may fail to change the state enough to ever approach the true ground state. Both of these problems are demonstrated in Fig. 3.14, in which we compare two different convergence schemes. Both begin with an initial step size of $\delta\tau = 0.1$, and in each case, after a fixed number of steps, the step size is decreased by a constant reduction factor. At early stages, it seems to be a problem to use a small reduction factor, since the evolution quickly get stuck in a plateau at each stage. In the later stages, however, where the time steps are small, we see how a large reduction factor can cause problems as well, since the step size quickly becomes too small to produce any meaningful change.

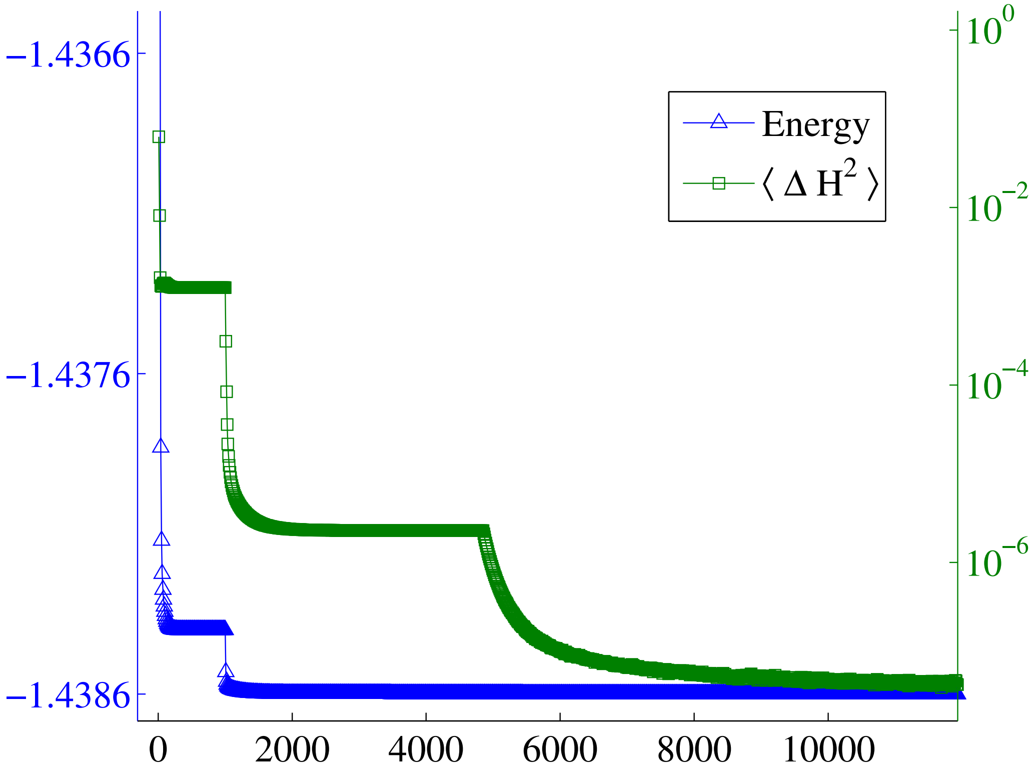

Figure 3.13: A test comparing the convergence behavior of the energy (left axis) and $\langle \Delta H^2 \rangle$ (right axis) during the iTBED algorithm as applied to the spin-1 Heisenberg model with a transverse field. The underlying data are the same as Fig. 3.12. Though the energy has converged to the level of one part in 10^5 after approximately 1000 steps, the energy cumulant takes much longer to reach a similar level of precision. In part, this is because the cumulant requires evolution with smaller time steps, in order to avoid plateaus and fully converge.

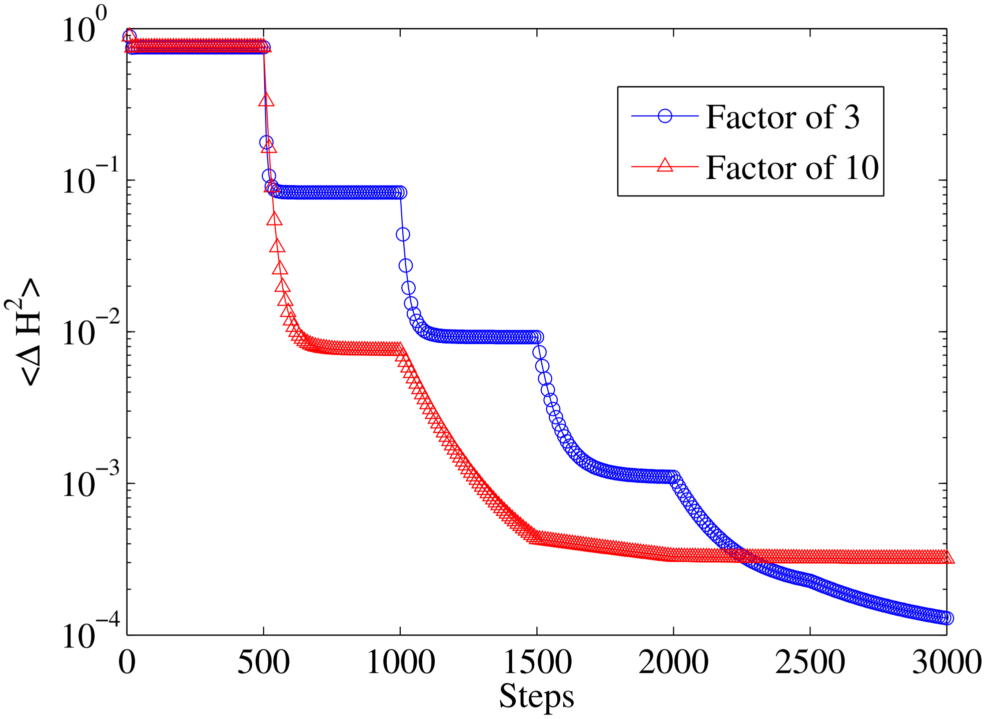

Figure 3.14: (Color Online) The transverse Ising model on a chain of L = 30 is studied by a basic TEBD algorithm. The initial imaginary time step is $\delta\tau = 0.1$. After every 500 steps of evolution, the step size is decreased either by a factor of 3 (triangles) or a factor of 10 (circles). In the early stages (steps 500-1500) the low-factor scheme is outperformed, because it spends more time stuck in local plateaus. In the later stages, however, the large-factor scheme falls behind. The evolution between steps 1000 and 1500 is truncated too early, before it had locally converged, and the subsequent time steps are too small to cause a meaningful change in the state.

Both of these problems, however, can be viewed not as the result of improper step sizing, but rather as the result of the fact that we used a fixed number of steps at each stage. If we employ a convergence check of the kind described above (for example, computing the coefficient of variation of the 30 most recent points) and move on to the next size time step whenever the data have locally converged, the differences in the two algorithms largely disappear (see Fig. 3.15). Time is no longer wasted in plateaus, since a new step size is triggered whenever the data have stopped changing significantly. On the other hand, the small time steps are never employed prematurely because the algorithm does not move on to a smaller step until the current step has outlived its usefulness. Hence, by requiring local convergence before changing the size of the time step, the differences between different sets of step sizes can be largely "smoothed out." Although there may be small advantages to a particular scheme in the context of a specific Hamiltonian, in general any set of step sizes that contains some intermediate-sized points can be successful (in practice, this likely means reduction factors between roughly 2 and 10).

Finally, we note that of course, in any convergence scheme, it must be remembered that the calculations involved in a convergence check can themselves contribute substantially to an algorithm's run-time, and one must take care to balance this against the desire to run in an optimal number of steps. In practice, the most important feature of an algorithm is likely the amount of real-world time taken to reach the desired level of precision, and the sophistication and frequency of the convergence checks can be relaxed if they are at odds with this goal.

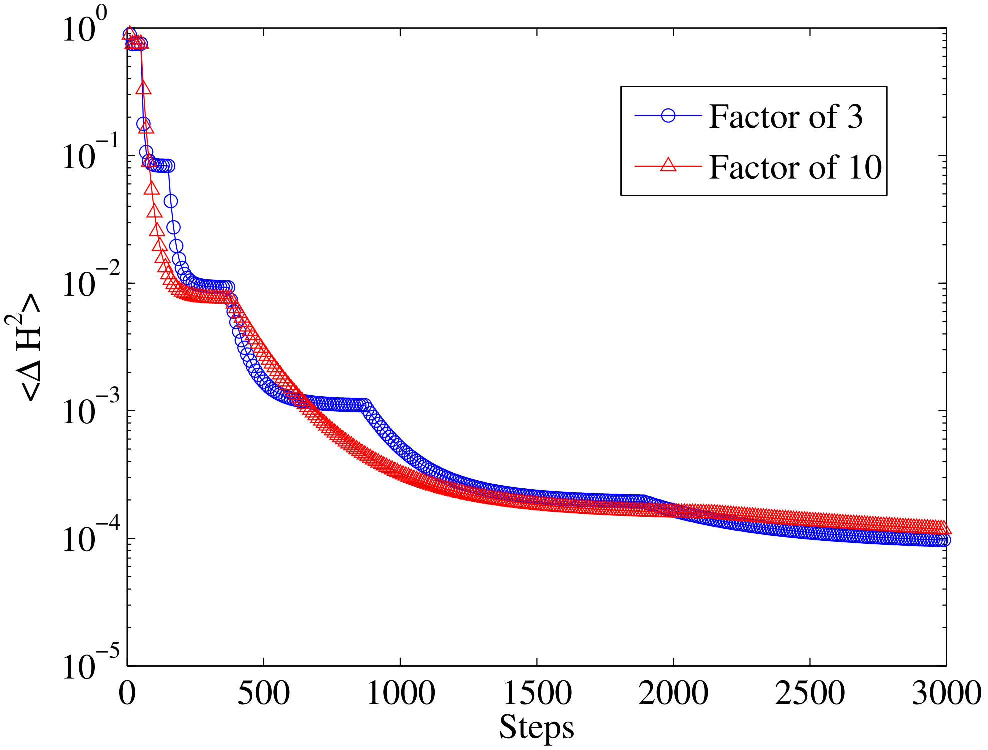

Figure 3.15: (Color Online) The transverse Ising model on a chain of L = 30 is studied by a basic TEBD algorithm. In contrast with the convergence scheme used in Fig. 3.14, here we use the coefficient of variation from the 30 most recent points to determine when to move on to a smaller step size. The result is that unnecessary plateaus are largely avoided, and no time step is terminated before it has finished producing a useful change in the state. As a result, the energy cumulant $\langle \Delta H^2 \rangle$ is always quite similar regardless of the step size. The final energy computed for the ground state differs by less than one part in 10^7.

A principle goal of tensor network algorithms is to obtain precise approximations to ground-state wave functions, which can then be used for many purposes. One such purpose of central importance to many-body physics is the detection of phase transitions [130], a goal with which tensor network methods have already proven enormously useful [19, 41–44]. Here, we present an additional method by which tensor network can contribute to the study of phase transitions, both by locating the critical transition points and by helping to identify critical exponents.

Chapter 4

Moments and Cumulants with Tensor Network States

In the symmetry-breaking paradigm for phase transitions, one first looks for an operator M whose expectation value $\langle M \rangle$ can serve as the order parameter, i.e. a quantity whose behavior changes sharply across a critical point. When this order parameter is represented by a local operator, it can be computed efficiently on a tensor network state [14]. But while an expectation value is the most straightforward piece of information associated with an operator and a state, there is considerably more information available which one may want to compute. For instance, one may wish to study the higher moments of the operator, $\mu_n = \langle M^n \rangle$. A related set of quantities called "cumulants," typically labelled κ_n, is also frequently of interest. An obvious example is the variance of the operator $\langle \Delta M^2 \rangle$, which is simply the second cumulant $\kappa_2 = \mu_2 - \mu_1^2$. Even more important to the search for phase transitions is the so-called "Binder cumulant," first introduced by Kurt Binder in 1981 in a study of the classical Ising Model [45]. In many settings, such as thermal or disordered systems, it is considered to be one of the most accurate and reliable means of detecting a critical point [131–133], and it has since been applied to a wide variety of models [134–142].

Computing these higher order moments and cumulants, however, is less straightforward. Direct calculation quickly becomes impractical for large n, since the number of terms to evaluate can be exponential in n. In a classical system with a Hamiltonian H_0, one might define $H(\lambda) = H_0 + \lambda M$, and relate the higher moments of M to the derivatives of an associated partition function, using $\langle M^n \rangle = (\beta \frac{\partial}{\partial \lambda})^n Tr(e^{-\beta H(\lambda)})$. In quantum systems, however, this equation only holds when $[H_0, M] = 0$, which is not true for a wide variety of physically interesting cases. Because of these barriers to direct calculation, usage of techniques such as the powerful Binder cumulant has in the past been generally confined to studies based on quantum Monte Carlo [134].

Such a powerful technique should not be limited in this way. The question thus naturally arises whether these quantities can be efficiently and systematically evaluated using the elegant structure of tensor network states. In this chapter, we demonstrate that the answer is yes. The feasibility of using matrix product states for computing the variance of Hamiltonians has been previously pointed out in the context of DMRG [26, 143], and is demonstrated by I. McCulloch in [32] and [87], via the technique of so-called "matrix product operators" (MPO) [14]. We propose an alternative technique, which gives a simple and efficient method to evaluate all general moments and cumulants for tensor network states, based on moment-generating and cumulant-generating functions. We demonstrate the calculation of moments and cumulants for finite one-dimensional states, and show that the method can also be used for per-site cumulants in the case of an infinite system. We also show how the techniques naturally generalize to finite systems in higher dimensions. These methods have a variety of useful applications which we demonstrate at length, including the use of the Binder and other cumulants to detect critical points to relatively high precision at a low numerical cost (calculation of the Binder cumulants by MPO methods has also been recently considered in [144] and [145]). We also apply the second cumulant of the energy to examine the convergence of numerical methods based on imaginary time evolution.

This chapter is organized as follows: In Sec. 4.1 we review moments and cumulants, in particular presenting the Binder cumulant and some of its applications. In Sec. 4.3, we demonstrate how to use these expectation values to efficiently compute the moments and cumulants for a general class of operators on an MPS. Sec. 4.4 contains examples of the method as applied to three different spin-chain models (the transverse Ising model, the spin-1 Ising model, and the spin-1 Ising model in a crystal field), as well as a demonstration of the method as applied to a two dimensional system (the transverse Ising model on a square lattice). Our results are summarized in Sec. 4.5.

4.1 Moments, Cumulants, and The Binder Cumulant

A state $|\psi\rangle$ and an operator M collectively imply a probability distribution: the probability density function of ψ in M-space. The expectation value $\langle M\rangle$ specifies the central value of the distribution, while the complete set of "Moments" defines the entirety of the shape [146]. The n^{th} moment of the distribution is defined to be $\mu_n = \langle M^n\rangle$; the first moment μ_1 is the expectation value $\langle M \rangle$ itself.

The cumulants of the distribution, κ_n, form an alternative but equivalent way of specifying its shape. These cumulants contain, in total, the same information as the moments; a complete set of either moments or cumulants completely specifies the distribution. Indeed, the n^{th} cumulant can always be expressed as a polynomial combination of the first n moments, and vice versa [147]. For example, as we have noted above, the second cumulant of the distribution, is the distribution's variance, defined by

$$\kappa_2 = \mu_2 - \mu_1^2. \tag{4.1}$$

The third cumulant κ_3 gives the distribution's skewness, and is related to the first three moments by

$$\kappa_3 = \mu_3 - 3\mu_2\mu_1 + 2\mu_1^3. \tag{4.2}$$

Similarly, the fourth cumulant κ_4 is related to the kurtosis, and is given by

$$\kappa_4 = \mu_4 - 4\mu_3\mu_1 - 3\mu_2^2 + 12\mu_2\mu_1^2 - 6\mu_1^4. \tag{4.3}$$

Although moments and cumulants are properly defined with respect to a distribution and hence depend on both M and $|\psi\rangle$, when $|\psi\rangle$ is general or clear from context we shall refer to μ_n (κ_n) as the "n^{th} moment (cumulant) of M".

4.1.1 Binder's Cumulant

The aforementioned Binder Cumulant is a particularly useful quantity in the study of critical points and phase transitions. For some system with some known order parameter M, for example a total x-magnetization $\sum_j \sigma_j^x$ or a staggered magnetization $\sum_j (-1)^j \sigma_j^x$, Binder's cumulant represents a modified version of that parameter's 4th cumulant. Though some slight variations exist in the definition, generally, it is given by

$$U_4 = 1 - \frac{\langle M^4 \rangle}{3\langle M^2 \rangle^2}. \tag{4.4}$$

The utility of the Binder cumulant arises from the special features of its length dependance. The behavior of the Binder cumulant at a critical point depends only weakly on the size of the system, and elsewhere, its behavior with respect to the system size differs depending upon the phase. For example, below the critical point in a symmetry-breaking magnetic phase the cumulant will increase with the length of the system, but above the critical point, with symmetry unbroken, it decreases instead. The result is that, when curves of the Binder Cumulant vs temperature are plotted for various lengths, the critical point is indicated by a simultaneous crossing. Typically, because the behavior at the critical point is already approximately universal, only a set of relatively small system sizes need be considered, eliminating the need for complicated extrapolations of very large systems to the thermodynamic limit.

The Binder cumulant also gives access to the critical exponent of the correlation length, by means of traditional finite size scaling techniques in which one seeks to "collapse" the data. [45, 148]. Up to some small finite size corrections (which become increasingly suppressed as the system size increases), the cumulants show a standard functional form, [148]

$$U_4(L, B) = \tilde{U}\left(L^{1/\nu}(B - B_c)\right), \tag{4.5}$$

where ν is the usual critical exponent. A plot of U_4 vs $L^{1/\nu}(B - B_c)$ should therefore appear essentially independent of L, since all of the length-dependance has been absorbed into the independent variable of the plot. Hence, B_c and ν can be treated as free parameters, and varied until this length-independence is optimized; for example, one could seek to minimize the total absolute square distance between the curves for a variety of lengths L.

4.2 Expectation Values for Product Operators

4.2.1 Finite-Length Chains

To demonstrate how to efficiently compute quantities such as the Binder cumulant for a system represented by an MPS [149], recall the nature of the MPS representation itself:

$$|\psi\rangle = \sum_s Tr(A^{s_1}_{[1]} A^{s_2}_{[2]} ... A^{s_L}_{[L]}) |s_1 s_2 ... s_L\rangle. \tag{4.6}$$

Let us now consider how to calculate the expectation value of a general family of product operators with respect to a matrix product state, as given in Eq. 4.6. Consider first the case of a simple operator which is given by a tensor product of on-site operations. For such an operator, of the form

$$Q = \bigotimes_j Q_j, \tag{4.7}$$

the expectation value $\langle Q \rangle$ is given by

$$\langle\psi|Q|\psi\rangle = \sum_{s_j,s'_j}\left[Tr\left(\prod_{j=1}^{L} A_{[j]}^{s_j}\otimes A_{[j]}^{*s'_j}\right)\prod_{j=1}^{L}\langle s'_j|Q_j|s_j\rangle\right], \tag{4.8}$$

or equivalently,

$$= Tr\left[\prod_{j=1}^{L}\sum_{s_j,s'_j}\left(A_{[j]}^{s_j}\otimes A_{[j]}^{*s'_j}\right)\langle s'_j|Q_j|s_j\rangle\right]. \tag{4.9}$$

From this expression, it is clear that, up to normalization, the expectation value is simply a trace over a set of L generalized transfer matrices (which extend the definition of the ordinary transfer matrix from Eq. 2.15); i.e.

$$\langle Q\rangle = \frac{1}{\langle\psi|\psi\rangle}Tr\left(\prod_{j=1}^{L}T_j\right), \tag{4.10}$$

where the T's are defined as

$$T_j \equiv \sum_{s_j,s'_j}\left(A_{[j]}^{s_j}\otimes A_{[j]}^{*s'_j}\right)\langle s'_j|Q_j|s_j\rangle. \tag{4.11}$$

This procedure is also demonstrated in graphical notation in Fig. 4.1. The norm of the state can be fixed in a similar fashion, by evaluating a transfer matrix for the special case where $Q_j = \mathbb{1}$.

Tensor products of few-body operators can be handled in a similar fashion by grouping the relevant sites. More general operators are simply evaluated by decomposing them into a sum of tensor products. Considerably more detail on the general process of taking expectation values can be found in the now-extensive body of literature on matrix product states, [14, 16, 83]. For our purposes, however, it will be sufficient to be able to evaluate operators of the simple form in Eq. (4.7).

4.2.2 Infinite-Length Chains

Recall from Chapter 3 that matrix products can also be used to represent infinite systems, provided that those systems possess sufficient translation invariance. For a state with a unit cell of length ℓ, we require only ℓ types of tensors to express it. For example, a state with two-site translation invariance ($\ell = 2$) is specified by only two tensors, A_1, A_2, and has the form

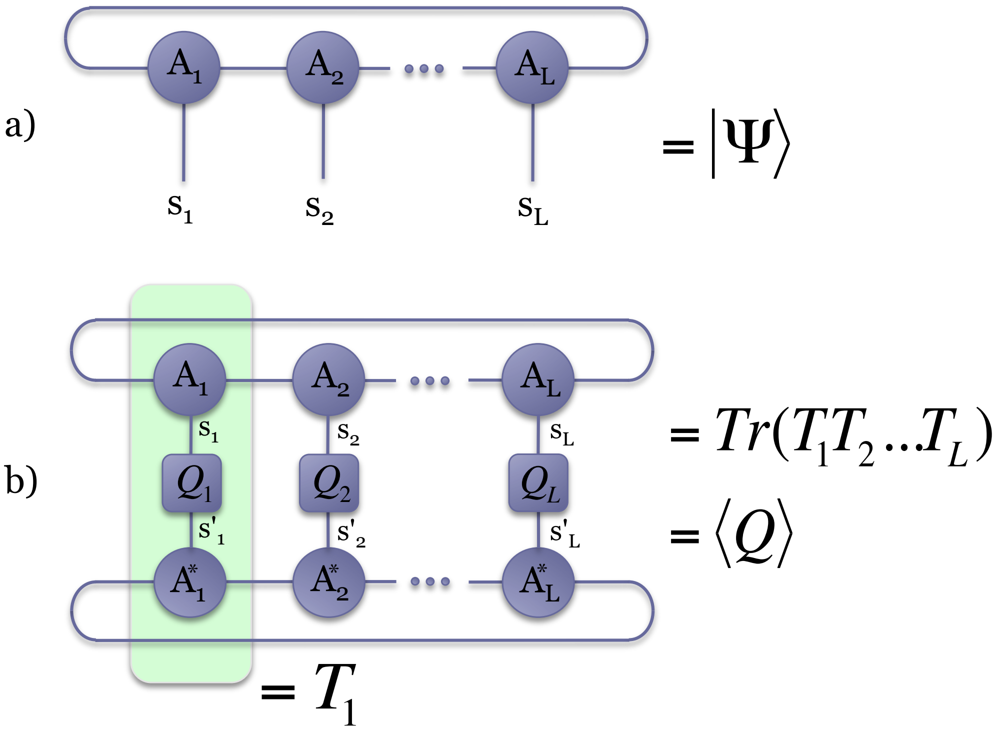

Figure 4.1: Graphical notation demonstrating the structures of matrix product states. In this notation, a shape represents a tensor, and a line represents an index. Connected lines between shapes represent contracted indices between tensors. (a) A finite spin chain state $|\psi\rangle$ represented as a matrix product state. The state is specified by the set of rank-three tensors $\{A_j\}$, with the physical degrees of freedom s_j left open. (b) The expectation value of a product operator $Q = \otimes_j Q_j$ with respect to $|\psi\rangle$. Each Q_j acts locally on only one site. The total expectation value can be thought of as a trace over a product of transfer matrices T_j, defined in Eq. (2.15). An example of an individual transfer matrix, T_1 is highlighted.

$$|\psi\rangle = \sum_s Tr(A^{s_1}_{[1]}A^{s_2}_{[2]}A^{s_3}_{[1]}A^{s_4}_{[2]}...)|s_1s_2s_3s_4...\rangle. \tag{4.12}$$

Of course, because the sums in Eq. 2.11 run over an infinite number of sites, in general there is no way to specify or compute the coefficients. Certain expectation values, on the other hand, may still be expressed as the limit of an infinite product of transfer matrices. It is quite common, for example, to consider the expectation value of an operator with the same translational invariance as the state in question, by looking at the per-site behavior. For our purposes, we will again be concerned with product operators of the form given in equation 4.7. However, we will now restrict ourselves further by imposing translation invariance on Q. For an infinite system with a unit cell of length ℓ, we shall consider only Q with $Q_j = Q_{j+\ell}$.

In this situation, one can still sensibly define the expectation value as

$$\langle Q\rangle = \frac{1}{\langle\psi|\psi\rangle} Tr\left(\prod_{j=1}^{\infty} T_\ell\right). \tag{4.13}$$

Here, the transfer matrix T_ℓ is now “enlarged” to represent an entire unit cell of the chain

$$T_\ell \equiv \prod_{j=1}^{l} T_j. \tag{4.14}$$

In order to approach the infinite case, we shall first examine the case of a finite but very long chain of length L, so that the product in Eq. (4.13) is limited to L/ℓ terms, i.e.

$$\langle Q\rangle_L = \frac{1}{\langle\psi|\psi\rangle} Tr\left(\prod_{j=1}^{L/\ell} T_\ell\right). \tag{4.15}$$

For L sufficiently large, the product can then be approximated by considering an eigenvalue decomposition of the transfer matrix $T_\ell = U\Lambda U^{-1}$ and inserting it into Eq. (4.15) (note that, by construction, the transfer matrix as defined by Eqs. (2.15) and (4.14) is Hermitian and hence diagonalizable). By applying the cyclic property of the trace operation, all U matrices can be made to cancel, leaving us only

$$\langle Q\rangle_L = \frac{1}{\langle\psi|\psi\rangle} Tr\left(\Lambda^{L/\ell}\right), \tag{4.16}$$

or,

$$\langle Q\rangle_L = \frac{1}{\langle\psi|\psi\rangle}\sum_{j=1}^{\chi^2}\lambda_j^{L/\ell}, \tag{4.17}$$

where λ_j are the diagonal elements of Λ, i.e. the eigenvalues of the matrix . At this point, it can be observed that in the infinite limit, only the largest eigenvalue λ_1 will contribute to the sum. In other words, we have simply

$$\langle Q\rangle_L = \frac{1}{\langle\psi|\psi\rangle}\left(\lambda_1\right)^{L/\ell}. \tag{4.18}$$

To fix the norm, we consider a particular transfer matrix $\tilde{T}_\ell$, defined as usual by Eq. (2.15) for the special case where $Q = \bigotimes_j \mathbb{1}_j$. Then, we calculate $\tilde{\lambda}_1$, the largest eigenvalue of $\tilde{T}_\ell$, which satisfies

$$\langle\psi|\psi\rangle = \left(\tilde{\lambda}_1\right)^{L/\ell}. \tag{4.19}$$

Substituting, we have

$$\langle Q\rangle_L = \left(\frac{\lambda_1}{\tilde{\lambda}_1}\right)^{L/\ell}. \tag{4.20}$$

We then gain access to the per-site behavior by means of a logarithm, which gives

$$\frac{1}{L}\log\langle Q\rangle_L = \log\left(\frac{\lambda_1}{\tilde{\lambda}_1}^{1/\ell}\right). \tag{4.21}$$

Eq. (4.21), however, does not depend on having a finite L. Thus, even for our infinite system, we can consider the limit

$$\lim_{L\to\infty}\frac{1}{L}\log\langle Q\rangle = \log\left(\frac{\lambda_1}{\tilde{\lambda}_1}^{1/\ell}\right). \tag{4.22}$$

Hence, with these procedures (illustrated graphically in Fig. 4.2), we can extract information about product operators in their infinite limit even though their expectation values generally diverge. As we will show in Chapter 4, this information will also allow us to compute the cumulants of operators with translation invariance even in the infinite case.

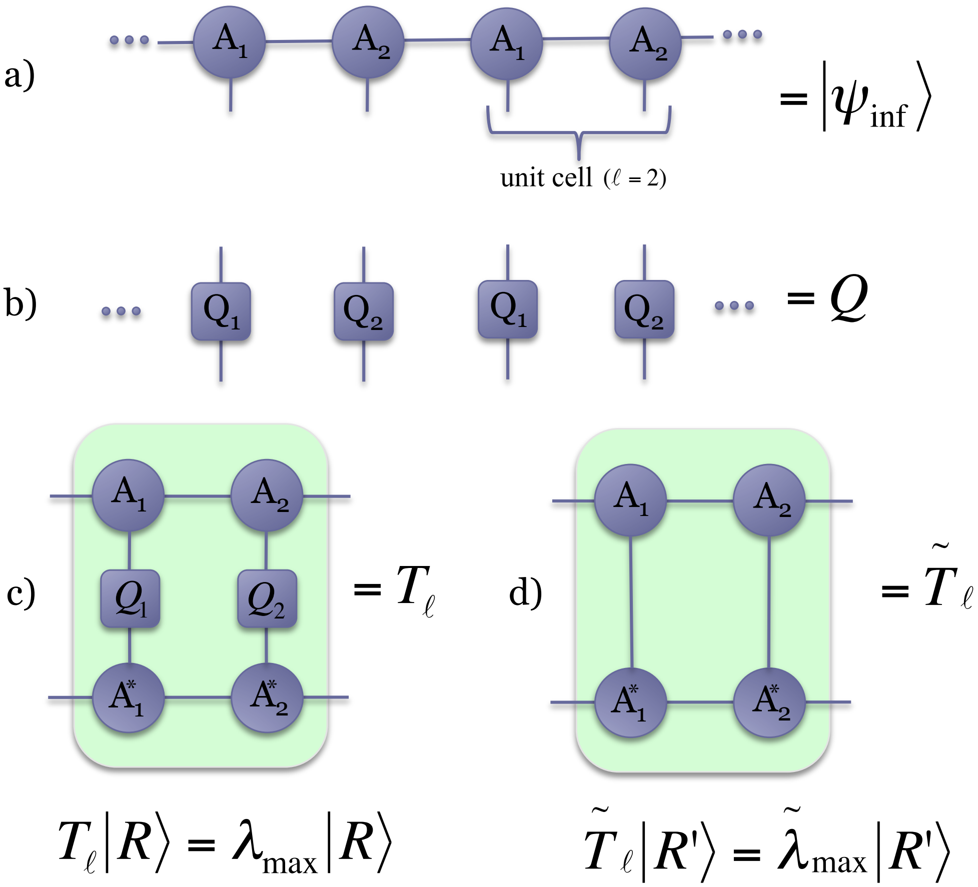

Figure 4.2: (a) An infinite spin chain state $|\psi_{inf}\rangle$, possessing translation invariance with respect to a unit cell of length $\ell = 2$, represented as a matrix product state. (b) A product operator $Q = \otimes_j Q_j$ which possesses the same translation symmetry as $|\psi\rangle$; i.e. $Q_j = Q_{j+\ell}$ (c) To compute the quantity of interest, we first construct T_ℓ, the transfer matrix containing an entire unit cell of $|\psi\rangle$ and Q, and extract its dominant eigenvalue λ_1. (d) To normalize the result, we will also need $\tilde{T}_\ell$ (a transfer matrix which contains only the identity operator) and it's dominant eigenvalue $\tilde{\lambda}_1$. The desired quantity $\lim_{L\to\infty} \frac{1}{L} \log\langle Q\rangle$ is given by $\log(\lambda_1/\tilde{\lambda}_1)^{1/\ell}$.

4.3 Evaluating Higher-Order Moments and Cumulants

4.3.1 Moment and Cumulant-Generating Functions

For a given operator M and a state $|\psi\rangle$, there is an associated function F which contains all of the non-local information about the higher moments $\langle M^n\rangle$, and yet, as we shall subsequently demonstrate, can still be efficiently evaluated within the framework of a tensor product state. In particular, this function is given by $F(a) \equiv \langle e^{aM}\rangle$.

In probability theory, $F(a)$ is termed the "moment generating function" of the probability distribution. It is so named because the information about every moment of the distribution is not only contained, but readily accessible from this single function. This can be made explicit by considering a Taylor-expansion of e^{aM} about $a = 0$ and then computing the expectation value in $F(a)$ term-by-term

$$F(a) = 1 + a\langle M\rangle + \frac{a^2}{2}\langle M^2\rangle + ... \qquad (4.23)$$

From the result, it is clear that every (non-vanishing) moment $\langle M^n\rangle$ will appear in the expansion. Furthermore, these moments can be directly accessed by computing

$$F^{(n)}(a) = \mu_n + \mathcal{O}(a), \qquad (4.24)$$

where $F^{(n)}(a)$ is as usual the n^{th} derivative of F.

The moment-generating function F is closely related to the so-called "characteristic function" of the distribution, $G(a) \equiv \langle e^{iaM}\rangle$ [150]. For typical states with well-behaved wave functions, these functions will be essentially interchangeable (up to a factor of i). Hence in this work, both will be used, sometimes in combination, depending on the particular moment or operator being computed. It should be noted, however, that for some "pathological" wave functions, such as those specifying a Lorentzian probability distribution, the function $F(a)$ may fail to exist. The characteristic function $G(a)$, however, being the expectation value of a bounded operator, does not suffer from this complication in any situation [151].

While Eq. (4.24) gives a result for the desired moment which is only accurate up to first order in the parameter a, the precision can be improved by instead computing appropriate combinations of the functions $F(a)$, $F(-a)$, $G(a)$, and $G(-a)$. For example, when seeking to compute an even-ordered moment; i.e. a moment of the form $\langle M^{2n}\rangle$, we construct

$$\tilde{F}(a) = F(a) + F(-a) = 2 + a^2\langle M^2\rangle + \frac{a^4}{12}\langle M^4\rangle + ... \tag{4.25}$$

And hence obtain the desired moments from the relation

$$\tilde{F}^{(n)}(a) \propto \langle M^{2n}\rangle + \mathcal{O}(a^2). \tag{4.26}$$

Odd-ordered moments can of course be found to higher precision from $F(a) - F(-a)$. Even greater precision can also be obtained by including the characteristic functions. The combination $F(a) + F(-a) - G(a) - G(-a)$, for instance, determines $\langle M^2\rangle$ up to $\mathcal{O}(a^4)$.

We employ a similar technique to extract the cumulants of the distribution. This is done by means of the "cumulant generating function," defined as

$$l_F(a) \equiv \log F(a). \tag{4.27}$$

To see the utility of this function, observe that

$$l_F(a) \approx \log(1 + a\langle M\rangle + \frac{a^2}{2}\langle M^2\rangle + ...). \tag{4.28}$$

For small enough values of a, one can see from the expansion $\log(1+x) \approx x - \frac{1}{2}x^2 + ...$ that

$$l_F(a) = a\langle M\rangle + \frac{a^2}{2}\langle M^2\rangle - \frac{a^2}{2}\langle M\rangle^2 + \mathcal{O}(a^3). \tag{4.29}$$

Grouping these terms by the powers of a, we find that in fact

$$l_F(a) = a\kappa_1 + \frac{a^2}{2}\kappa_2 + ... \tag{4.30}$$

In other words, the derivatives of the function $l_F(a)$ give us direct access to the cumulants in the same manner as the moments in Eq. (4.24)

$$l_F^{(n)}(a) = \kappa_n + \mathcal{O}(a). \tag{4.31}$$

Of course, as with the moments, appropriate combinations of $l_F(a)$, $l_F(-a)$ and the associated complex functions can be used to suppress the higher order terms and improve the accuracy.

4.3.2 Evaluating Generating Functions on a Finite Matrix Product State

Evaluating all of these moments and cumulants thus boils down to evaluating the expectation values of operators like e^{aM}. It remains to be shown that these operators, which we term "moment generating operators," can be applied in an efficient manner. Fortunately, the exponential structure of the operator guarantees that this is indeed the case.

We will consider these operators in two cases, depending on the nature of the operator M. The first, special case is the large class of operators where M can be written as $\sum_j O_j$, where j runs over all the sites in the system and for some arbitrary set of on-site operators $\{O_j\}$. Most usefully, this set of operators contains the traditional magnetization operators such as $M_x = \sum_j \sigma_j^x$, as well as staggered magnetizations, crystal field magnetizations, etc. Subsequently, we will examine the more general case where the terms within M act on more than one site at a time.

Sums of Single-Body Operators

In this case, since the operators O_j all act at separate sites, the combined operator M is in fact simply a Kronecker sum, $M = \bigoplus_j O_j$. From this it follows that we can write [152]

$$e^{aM} = \bigotimes_j e^{aO_j}. \tag{4.32}$$

In other words, the moment-generating operator can be decomposed into a set of operators $\{e^{aO_j}\}$, each acting only at a single site. The moment generating function, in turn, is just the expectation value of all these operators applied simultaneously.

Since our moment-generating operator can be written in this tensor product format, we can compute its expectation value directly through Eqs. (4.10) and (2.15), with $Q_j = e^{aO_j}$ (see Fig. 4.3). The calculation of our moment-generating function $F(a)$ has therefore been reduced to the calculation of L transfer matrices and a single trace over their product. In practice, to calculate a higher moment like $\langle M^2 \rangle$, we then need to repeat this procedure and compute F for slightly different values of a, so that it is possible to evaluate the necessary derivative numerically. This can be done through any of the wide variety of standard methods; in this work we have used primarily the classic divided difference formulas [153]. In general, the more values of a at which $F(a)$ is computed, the higher the accuracy of the derivative. However, since the initial a is already chosen to be quite small, in practice it is often the case that only a very small number of points needs to be computed (to check the behavior and accuracy, the procedure can always be repeated with a smaller value of a).

Let us examine now the performance of the method for the case of $\langle M^2 \rangle$. A second derivative is necessary, which can be computed to second order in a from three values of a, centered at $a = 0$ [153]. Noting that when $a = 0$ we have trivially $F(a) = 1$, it follows that we only have to compute two expectation values, each involving the construction of (at most) L transfer matrices, which are then multiplied together and traced over. By comparison, to compute $\langle M^2 \rangle$ directly, as the sum of all correlators $\langle O_j O_k \rangle$, requires the construction of the same number of transfer matrices (to cover the special case of the correlator where $j = k$), but these matrices must be multiplied and traced over up to L^2 separate times. Since some of these products of transfer matrices in these calculations will appear more than once, the actual computational cost can be reduced somewhat through use of a suitably "dynamically programmed" algorithm, where previously calculated products are saved and recycled [14]. Even in this case, however, far more than two solitary products would be required. Furthermore, as the order of the desired moment μ_n increases, the advantage of the moment-generating function method becomes increasingly pronounced, as the numerical derivative will require only approximately n expectation values, instead of L^n.

Simply put, the fact that the exponential nature of the moment generating operator turns long Kronecker sums into simple Kronecker products makes it ideally suited for use with a matrix product state. In all cases, only a small number of expectation values must be computed in order to allow the calculation of a numerical derivative, with each expectation value containing the operator e^{aO_j} at every site j. Moreover, application of these local operators does not increase the bond dimension of the state.

Such moments can in principle also be computed by means of an MPO [32, 87]. As a straightforward demonstration of this, consider for example the MPO given by [154]

$$\hat{C}_j = \begin{pmatrix} \mathbb{1}_j & 0 \\ O_j & \mathbb{1}_j \end{pmatrix}, \tag{4.33}$$

coupled with the boundary conditions

$$\langle \phi_L | = \begin{pmatrix} 0 & 1 \end{pmatrix}, \tag{4.34}$$

and

$$|\phi_R\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{4.35}$$

To evaluate the moments, one defines the total MPO W to be

$$\begin{aligned} W &\equiv \langle \phi_L | \prod_{j=1}^{L} \hat{C}_j | \phi_R \rangle \\ &= \sum_{j=1}^{L} O_j. \end{aligned}$$

so that

$$\mu_n = \langle \psi | W^n | \psi \rangle. \tag{4.36}$$

In this naive implementation, each application of W to $|\psi\rangle$ increases the bond dimension of the state by a factor of two, and thus, to calculate the n^{th} moment in this way requires a bond dimension exponential in n. This increase can be overcome, if necessary, by means of a standard truncation approximation, as done in the TEBD algorithm. Alternatively, a more sophisticated MPO can be constructed [155] which represents the operator M^n, but with a bond dimension of just $n+1$, resulting in a procedure which still scales linearly.

Sums of Many-Body Operators

We now examine the case of calculating the higher-order moments of a more general set of operators $M = \sum_j O_j O_{j+1} ... O_{j+k}$. In other words, we consider operators which are a sum of terms acting on at most k sites at a time. So long as k is finite, it remains possible to evaluate the moment generating functions with a single expectation value. This can be done by appealing to the same TEBD technique [27, 81, 115] discussed in Chapter 3.

To begin, we partition the terms of the operator into classes of mutually commuting operators. This will require at most k classes. For concreteness, consider as an operator the two-body transverse Ising Hamiltonian

$$H = -\sum_j \sigma_j^x \sigma_{j+1}^x + B\sigma_j^z. \tag{4.37}$$

For simplicity, let us write H in a manner which makes it explicitly a sum of two-body operators

$$H = -\sum_{j} \sigma_j^x \sigma_{j+1}^x + \frac{B}{2}\left(\sigma_j^z + \sigma_{j+1}^z\right). \tag{4.38}$$

Then, the terms can be partitioned between the even and odd pairs of sites, and H can be written as

$$H = H_{even} + H_{odd}, \tag{4.39}$$

with

$$H_{even} = -\sum_{jeven} \sigma_j^x \sigma_{j+1}^x + \frac{B}{2}\left(\sigma_j^z + \sigma_{j+1}^z\right) \tag{4.40}$$

and

$$H_{odd} = -\sum_{jodd} \sigma_j^x \sigma_{j+1}^x + \frac{B}{2}\left(\sigma_j^z + \sigma_{j+1}^z\right). \tag{4.41}$$

The moment generating function therefore has the form $F(a) = e^{aH_{even}+aH_{odd}}$, and thus, as in the case of the iTEBD time evolution operator, it admits a Suzuki-Trotter approximation [118]. To second order in a, this approximation has the form

$$e^{aH_{even}+aH_{odd}} \approx e^{\frac{a}{2}H_{even}} e^{aH_{odd}} e^{\frac{a}{2}H_{even}}. \tag{4.42}$$

Higher order versions of the approximation have also been well-documented and can be easily substituted where greater precision is required [123].

Because H_{even} and H_{odd} were explicitly constructed to be sums of mutually commuting terms, each exponential in the right hand side of Eq. (4.42) is now in the same Kronecker sum form as we had in the case of on-site operators, and each can therefore be equivalently expressed as a single tensor product of operations acting on the entire state at once, in the manner of Eq. (4.32). This is graphically depicted in Fig. 4.3. Hence, by applying the three exponentials from Eq. (4.42) in sequence, we can easily calculate the expectation value that represents the moment-generating function.

From this point, evaluating the expectation value is no different than the case of on-site operators. The bulk of the numerical costs are therefore essentially the same for both on-site and many-body operator, with only one difference: applying these layers of exponential operators will increase the bond dimension of the system, and the size of these bonds may need to be "truncated" by some approximation scheme to keep the system numerically tractable. This, however, is a common practice in the field of MPS algorithms, and is easily done by means of a Schmidt decomposition (see for example [28]).

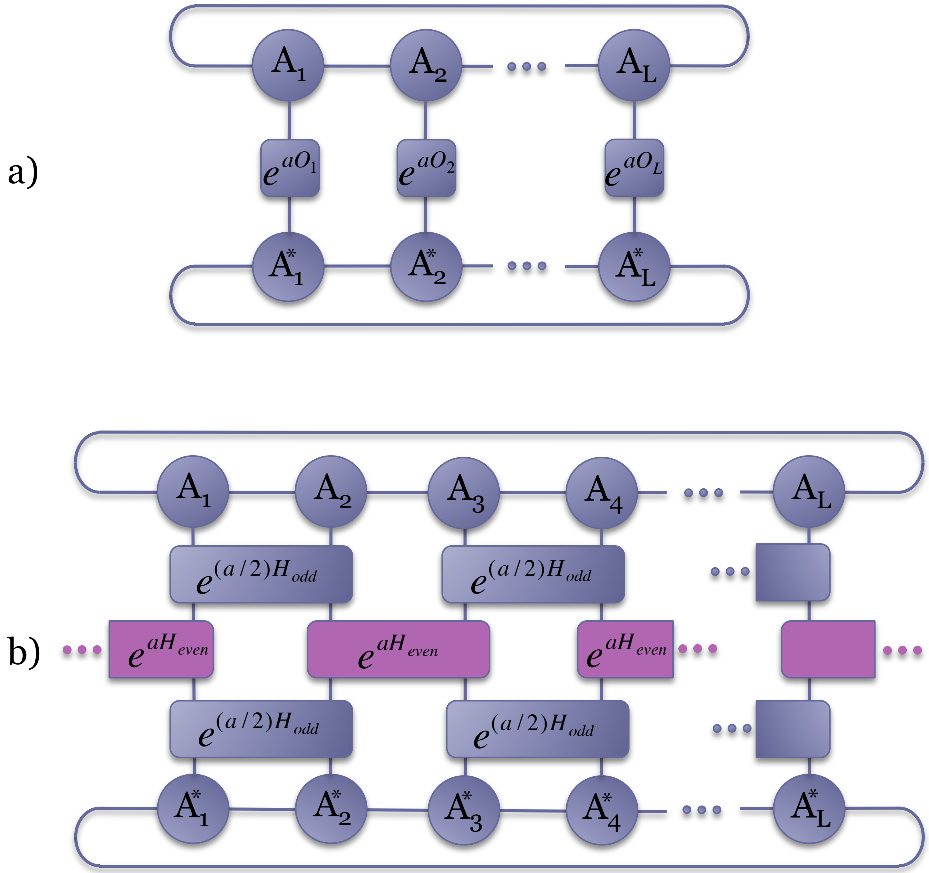

Figure 4.3: (a) Graphical representation of the moment-generating function $F(a) = \langle e^{aO} \rangle$ for an operator $M = \sum_j O_j$. Since each term in M acts at only one site, the moment-generating operator possesses the same structure, even though the moments M^n are fundamentally non-local. (b) The moment-generating function for an operator which is the sum of two-body terms and which possesses the form $H = H_{odd} + H_{even}$, such as the transverse Ising Hamiltonian defined in Eq. (4.37). The operator is approximated by the second-order Suzuki-Trotter formula in Eq. (4.42), which produces three "layers" of operations. Each layer is a sum of two-body terms.

While our example considered an operator with $k = 2$, that is, two-body interactions which could be partitioned into two internally commuting classes, the technique easily generalizes to larger interactions. The Suzuki-Trotter approximations, for example, can be iteratively applied to an operator e^{A+B+C} by first approximating $e^{A+(B+C)}$ in terms of e^A and e^{B+C}, and then approximating e^{B+C}.

The ability to calculate moments and cumulants for an operator with many-body terms has a particularly useful application in the world of numerical state estimation. A common goal of tensor network algorithms is the calculation of an approximate numerical ground state, from a given Hamiltonian H. These algorithms are typically iterative in nature, gradually refining the approximation as the energy E tends towards E_0. It is therefore often important to have a means of actively checking this convergence during the course of the algorithm. The variance (second cumulant) of the Hamiltonian, $\langle \Delta H^2 \rangle = \langle H^2 \rangle - \langle H \rangle^2$ is well-suited to this task [14]. For $\epsilon = \sqrt{\langle \Delta H^2 \rangle}$, there will be an exact eigenvalue E_{ex} within ϵ of the approximate energy E. In other words

$$|E - E_{ex}| \leq \epsilon. \tag{4.43}$$

In the case of finite systems, the quantity ϵ can be directly computed from the methods above, and can therefore be directly used as an error bound on the calculated energy of the system. For the infinite case, there is a small subtlety: one does not compute the total energy of the system, since this is infinite, but rather the energy per site, E/L. Consequently, a proper "error bound" on the measurement would not be ϵ, but $\epsilon/L = \sqrt{\langle \Delta H^2 \rangle}/L$. This quantity, unfortunately, is inaccessible using the techniques above, since the cumulant generating function on the infinite system gives only $\langle \Delta H^2 \rangle/L$. However, one would still expect the latter number to be monotonically related to the true error. Alternatively, as we will discuss in the next section, the related quantity $\sqrt{\langle \Delta H_I^2 \rangle}/\ell$ can be evaluated on a finite interval of length ℓ. Hence in either case, we can derive something which gives a sufficient criterion for halting a converging algorithm like TEBD or iTEBD. Although ϵ (in the finite case) or $\langle \Delta H^2 \rangle$ (for the infinite) are not guaranteed to be small as soon as the energy has converged, if we iterate our algorithms until they *are* very small, we can be assured that the approximate ground state energy is very close to the true value E_0.

In [14], a dynamically programmed algorithm was given for computing $\langle \Delta H^2 \rangle$ in the context of a finite matrix product state. The method presented here performs at least as efficiently as that case, with the advantage that it can also be applied to infinite (or indeed, higher-dimensional) systems. In [32], the same quantity was presented and evaluated by means of an MPO. As discussed above, the MPO technique can in principle be used as an alternative method to compute other moments and cumulants as well, at the cost of allowing the bond dimension to increase. The MPO techniques will also have an advantage in the case of an operator with non-local characteristics or other properties that make Suzuki-Trotter decomposition impractical.

For the case of the energy cumulant, our method is also particularly well-suited for use with the TEBD/iTEBD algorithms. We have previously remarked that for a Hamiltonian H, the calculation of the associated moment-generating operator e^{aH} is essentially identical to an imaginary time-evolution operator with a time step of $\delta t = a$. Each iteration of this algorithm therefore amounts to calculating $e^{aH}|\psi\rangle$, from which the moment-generating function $F(a) = \langle e^{aH}\rangle$ can easily be computed. If one also computes $F(-a) = \langle e^{-aH}\rangle$, the error bound can therefore be computed very efficiently up to order a^2 in accordance with Eqs. (4.25) and (4.26).

Performing this check at regular intervals throughout the evolution offers a halting condition to certify convergence of the energy. In some respects, this convergence criterion is superior to the typical methods, which often signal a halt when δE, the change in the approximate energy between two successive iteration steps, drops below some minimum value. Such a method can occasionally give a false sense of convergence when the algorithm "stalls out" and begins evolving only very slowly, despite remaining some distance from the ground state. The variance of the energy provides information not about the convergence of the algorithm but of the energy itself, by identifying when the system is very close to an exact eigenstate (however, in some cases care must be taken that the nearby eigenstate is in fact the ground state, and not some excitation). More details about convergence schemes can be found in Appendix B.

4.3.3 Evaluating Generating Functions on an Infinite Matrix Product State

At first glance, it may seem that moment-generating techniques discussed above cannot be applied to infinite systems, since the value of a quantity like $M = \sum_j^\infty O_j$ is clearly diverging, and only related limits like

$$\langle M\rangle = \lim_{L\to\infty} \frac{1}{L}\sum_j^\infty \langle O_j\rangle \tag{4.44}$$

are well-defined. In this situation, however, while the moment-generating function F defined above may diverge, one can still define and calculate the related quantity

$$F_\infty = \lim_{L\to\infty} \langle e^{aM}\rangle^{1/L}. \tag{4.45}$$

As discussed in Sec. 4.2.2, quantities of this form can in fact be computed quite naturally. Using equation 4.20 (see Fig. 4.2), clearly we have

$$F_\infty = \lim_{L\to\infty} \langle e^{aM}\rangle^{1/L} = \frac{\lambda_1}{\tilde{\lambda}_1}. \tag{4.46}$$

In other words, the desired quantity is simply the largest eigenvalue of the transfer matrix associated with the state and operator in question. Hence by defining

$$l_{F_\infty}(a) = \log F_\infty(a), \tag{4.47}$$

we find that we have access to the per-site limits of the cumulants even in the infinite case. In the same manner as with finite systems, they are given by the derivatives of l_{F_∞} with respect to a

$$l^{(n)}_{F_\infty}(a) = \lim_{L\to\infty} \frac{1}{L}\kappa_n + \mathcal{O}(a). \tag{4.48}$$

We shall comment briefly on some practical considerations that are important when evaluating l_{F_∞} for a real matrix product state. First, algorithms for generating the states which are based on the iTEBD principle are likely to require the use of a two-site unit cell, even if the state is expected to possess only one-site translation invariance, as a result of the two-body nature of most parent Hamiltonians. In this case, of course, a two-site transfer matrix is required ($\ell = 2$), and we must take a square root of its largest eigenvalue in order to recover the correct per-site limit.

Additionally, we observe that for the second cumulant in particular, it can be particularly desirable to calculate using the characteristic function $G = \langle e^{iaM}\rangle$ instead of F. This is because one can then take advantage of the fact that

$$\begin{aligned}\lim_{L\to\infty} \frac{1}{L}\kappa_2 &= l_{G_\infty}(a) + l_{G_\infty}(-a) + \mathcal{O}(a^2)\\ &= \log G_\infty(a) + \log G_\infty(-a)\end{aligned}$$

Then, combining the two log terms and observing that $G_\infty(-a) = G_\infty(a)^*$, we have

$$\lim_{L\to\infty} \frac{1}{L}\kappa_2 = \log\left(|G_\infty(a)|^2\right) + \mathcal{O}(a^2). \tag{4.49}$$

In other words, we can calculate the per-site second cumulant up to second order in a by evaluating $G_\infty(a)$ only once, and without directly computing any numerical derivative.

Note that, although this method gives access to the per-site limit of cumulants on an infinite system out to arbitrary order, the typical definition of the Binder cumulant, as a rational function of moments, cannot be computed in this way without encountering a divergence. When the Binder cumulant is being studied for it's finite-size scaling properties, this difficulty is often immaterial. In other case, however (such as when the limiting value of the Binder Cumulant itself is the quantity of interest), we refer the reader to a recent and inventive MPO technique which can be used to access the cumulant directly; see [156, 157].

It should be noted also that within a system, the moment and cumulant methods can also be applied to any finite subset of spins, for example, $S_I \equiv \sum_{i\in I} S_i^z$, where I is any set of locations. In the infinite case, it can be applied to any subset of spins in a finite interval. For example, one can apply gates $e^{iaS_1^z}...e^{iaS_L^z}$ to only a segment of L sites in the infinite system. The expectation value of this, which we denote by $f_L(a)$, can be straightforwardly calculated by the transfer matrix method, in a manner analogous to the case of a finite state:

$$f_L(a) = \langle v_l | T_L^{S^z} | v_r \rangle / \tilde{\lambda}_1^L, \tag{4.50}$$

where as before $\tilde{\lambda}_1$ is the largest eigenvalue of the transfer matrix containing only the identity, T_L, and $T_L^{S_z}$ is the generalized transfer matrix containing the operator S_z at each site.

Of particular interest in this case, for example, are the moments given by

$$\left\langle \left(\sum_{i=1}^{L} S_i^z \right)^2 \right\rangle_L = -f^{(2)}(0), \tag{4.51}$$

and

$$\left\langle \left(\sum_{i=1}^{L} S_i^z \right)^4 \right\rangle_L = f^{(4)}(0), \tag{4.52}$$

from which the L-dependent Binder cumulant $U_4(L)$ can be computed as

$$U_4(L) = 1 - f^{(4)}(a=0)/[3f^{(2)}(0)]^2. \tag{4.53}$$

This is the Binder cumulant for a segment of L spins in an infinite chain.

Although our example discussed an operator which was a sum of single-body terms, the technique is not so limited. In a similar fashion, one can compute the moments and cumulants of some finite subset of the Hamiltonian, $H_I = \sum_{i \in I} h_i$. In this case, one would split the chain into two disjoint segments, I and its complement. Then, in the same manner by which a single step of the iTEBD algorithm was used to apply required operator e^{iaH} in the infinite case, so too can the finite TEBD algorithm be used to apply e^{iaH_I} to the finite interval. After this, the interval is recombined with the infinite remainder of the state, and standard finite and infinite MPS techniques can be used to evaluate the requisite expectation value.

4.3.4 Higher-Dimensional States

Although in this work we shall be applying these techniques to one-dimensional systems, there is nothing about the procedures above that cannot be immediately generalized to finite-sized higher dimensional states. Consider for example a total magnetization-type operator on a 2 dimensional, $L \times L$ square lattice, given by

$$M = \sum_j \sum_k O_{jk}, \tag{4.54}$$

where O_{jk} represents a specific operator acting locally on site (j, k) of the lattice. Of course by representing both j and k by some composite index J (now running from 1 to L^2) we can immediately see that M is no different than the magnetization-like operators we considered in the one-dimensional case

$$M = \sum_J O_J, \tag{4.55}$$

and hence that our earlier analysis goes through: the moment generating operator e^{aM} can still be written as

$$e^{aM} = \bigotimes_J e^{aO_J}. \tag{4.56}$$

This object is still an operator that acts only locally and whose expectation value can be evaluated all at once (graphically depicted in Fig. 4.4).

If we consider instead an operator H which contains many-body terms (but for whom each term acts nontrivially only on a finite number of sites), one can play the same tricks as in one dimension, first partitioning the terms into mutually commuting sets of terms $H_1, H_2 + H_3 ...$, then expressing the moment-generating operator as

$$e^{aH} = e^{a(H_1+H_2+H_3\ldots)}, \tag{4.57}$$

and finally applying some form of Suzuki-Trotter approximation as described above to express the operator as a product of exponentials, each of which can be applied to the state all at once. For nearest-neighbor interaction terms on a square lattice, the procedure is essentially identical to the one dimensional case, except that one must use four classes instead of two: two for interactions in the horizontal direction, and two for the vertical. For operators with more complicated terms (such as a sum of "plaquettes,") the number of partitions may be larger, but in general the costs do not grow rapidly despite the increase in system dimension.

Hence, in either case, the associated moment-generating operators can be disentangled into tensor products or sequences of tensor products, even in higher dimensions. Simply put, the essential "power" of the moment-generating function method is the fact that the moment-generating operator e^{aM} of a local operator M is itself a local operator, and this fact is unchanged regardless of the dimensionality of the system.

Once the moment-generating operator has been expressed as a tensor product and applied to the state, it still remains to numerically contract the tensor network. It is at this stage where things become more difficult than in the one-dimensional case, since computing the expectation value of *any* operator on a higher-dimensional tensor network state can be quite hard. Recall from Chapter 2 that exact calculation has been shown to be exponentially costly in L (in particular, it is a #P-hard problem [96]). Nevertheless, a wide variety of numerical techniques have been developed to approximate these contractions efficiently with minimal errors, the details of which are outside the scope of this project (See for example refs. [33, 97–101]). We do caution, however, that in our experience, the higher the order of the moment, the higher the sensitivity of the result to the errors introduced by approximate contraction.

4.4 Examples

4.4.1 Spin-1/2 Transverse Ising Model

As has become almost customary, we begin by demonstrating our technique in the context of the widely-studied transverse Ising model; a chain of spin-1/2 particles governed by the Hamiltonian

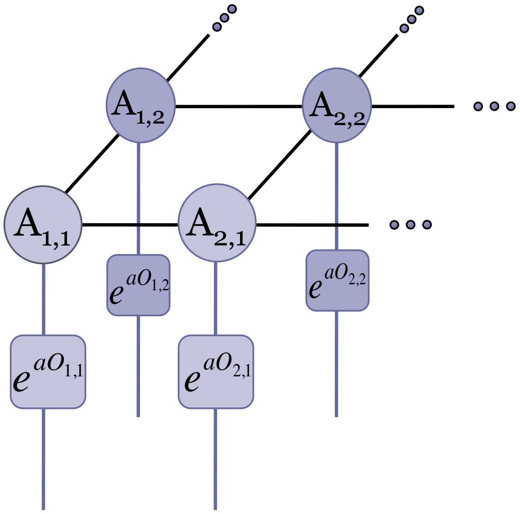

Figure 4.4: The moment-generating operator e^{aM} for an operator of the form $M = \sum_j \sum_k O_{jk}$, applied to a two-dimensional state on a square lattice. As in the one-dimensional case, the locality of each term in M ensures the locality of the terms in e^{aM}, and hence, the moment-generating operator can still be evaluated all at once, by applying the appropriate onsite operator at each lattice site.

$$H = -\sum_{j}^{L} \sigma_j^x \sigma_{j+1}^x + B\sigma_j^z. \tag{4.58}$$

This model is a useful proving ground as it has been extensively studied and admits a well-known analytical solution [158, 159], as well as possessing a straightforward order parameter of $M_x = \sum_j \sigma_j^x$, the total magnetization in the x-direction. We can therefore test our techniques by using them to study the phase transition known to occur exactly at $B_c = 1$. To apply the Binder cumulant technique, we first use a numerical method to find the ground state by solving the generalized eigenvalue problem [14]. We consider system lengths between $L = 10$ and $L = 45$ in steps of five, using a bond dimension of $\chi = 10$ (we verify that increasing the bond dimension does not change the results of the methods up to our working precision). For each system length, ground states of the Hamiltonian in Eq. (4.58) are computed as we sweep over a range of values for the field coefficient B. Then for each value of L and B, we compute the Binder cumulant using the methods described above, by first computing μ_2 and μ_4 by means of the moment-generating function.

As shown in Fig. 4.5, the crossings of the Binder cumulants at various lengths are already clustered very close to the transition point, even though the lengths of the states are relatively short compared to the thermodynamic limit. But the location of the critical point can be computed to even greater accuracy by considering the pattern of successive crossings. These crossings show a clear trend towards a limiting value as the system sizes increase. This limiting value can be estimated by means of the BST Algorithm [160], which has been found to be a very powerful tool for estimating the infinite limit of a series of data based on finite size corrections which obey a power-law, even from a relatively small number of data points [161]. From this extrapolation, we estimate a critical point of $Bc = 1.001(1)$. Here and elsewhere, the reported uncertainty in our extrapolation represent an estimate of the typical uncertainties in the location of the crossing points, propagated through the BST Algorithm. Much more detail about the BST technique can be found in Appendix A.

The critical point of the Ising model can be probed directly through the higher-order cumulants of the order parameter, as well. Through the techniques above, these can be calculated from the finite systems at various system sizes. Alternatively, we can calculate a ground state for the infinite system through the iTEBD algorithm (in this case using a bond dimension of $\chi = 20$) and then calculating the second cumulant directly. Both procedures are showcased in Fig. 4.6. The behavior in the infinite case can be seen to agree with the limiting trend of the finite systems as the length is increased. The cumulant can be seen to become singular near the critical point, and can also be

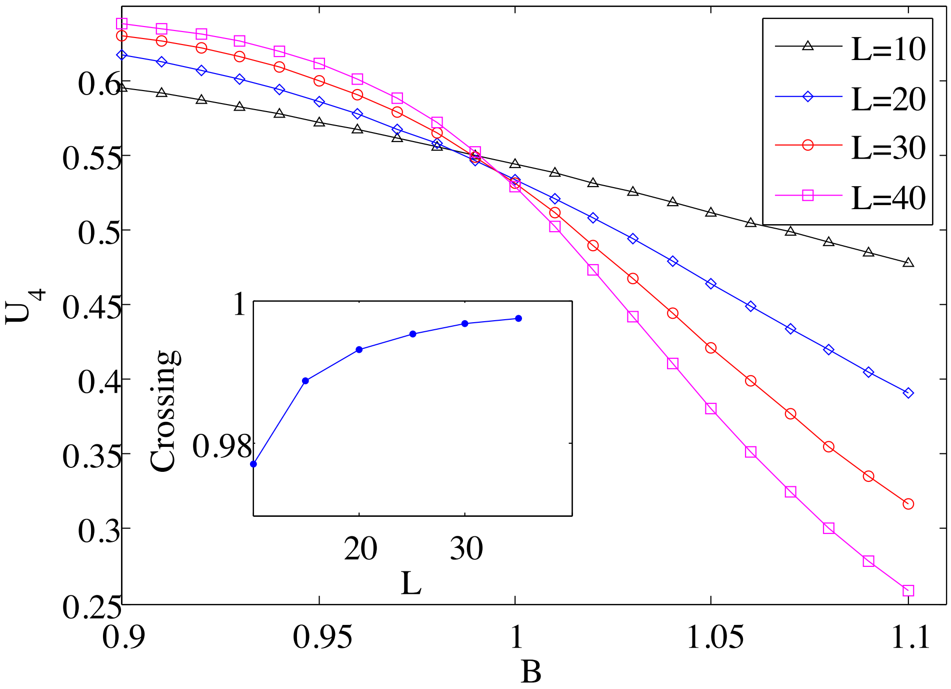

Figure 4.5: (Color online) A Binder cumulant study of the transverse Ising model. The cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e. L = 10 and L = 15. These crossing values can then be seen to approach the known value of the critical field, $B_c = 1$ (inset). The BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.001(1)$.

used to detect the transition. Using this method, we estimate $B_c = 1.00(1)$

This technique can also be applied to the magnetization in the transverse direction, $M_z = \sum_j \sigma_j^z$. In this case, it is the derivative of the cumulant which becomes singular in the thermodynamic limit to signify the critical point. Again, the results from the finite chains can be seen trending towards the infinite limit. In this case, the expected behavior in the thermodynamic limit can also be calculated analytically in this case, by using the known form of the Ising model wavefunctions. The cumulant as a function of the applied field B is found to be

$$\langle \Delta M_z^2 \rangle / L = \begin{cases} 1 & : B \leq 1 \\ 1/B^2 & : B > 1 \end{cases} . \tag{4.59}$$

This numerical data (see figure 4.7) are in agreement with this expression to a very high degree of accuracy.

The critical exponent of the correlation length of the model can also be studied by means of the Binder cumulant. Once the critical point has been estimated, the curves of U_4 can then be plotted against $L^{1/\nu}(B - B_c)$ for various values of ν. At the true critical exponent, the data should "collapse" to a single functional form independent of L, as seen in Fig. 4.8.

Additionally, we can use this model to consider the scaling of the Binder cumulant with the correlation length, by employing the procedure outlined above to compute moments and cumulants on a finite sub-block of an infinite state. From this, the so-called "critical Binder cumulant" U_4^* (the limiting value of the cumulant at the critical point) can be extracted. As suggested by McCulloch [162], we can evaluate $U_4(L)$ close to the critical point with the choice that $L \approx 10\xi$ for a number of bond dimensions χ. Choosing a sub-block of this size should allow us to see the desired scaling behavior while suppressing the finite-χ corrections to the cumulant. This can then computed over a range of transverse field values B, and the resulting Binder cumulant vs. B curves for different bond dimensions should cross approximately near the critical point and at approximately the value of U_4^* [162].

A simple demonstration of this, using bond dimensions from $\chi = 2$ to $\chi = 8$, can be seen in figure 4.9. Despite the fact that the bond dimensions are so small, this alternative scaling already gives a crossing of $B_c \approx 0.998$. The value of the cumulant at the crossing point (the so-called "critical Binder cumulant") is $U_4^* \approx 0.57$.

Finally, we can use this model on a finite lattice to demonstrate the utility of the energy variance $\langle \Delta H^2 \rangle$ in assessing numerical convergence, as described above. Starting from a random state with $L = 10$, we apply the TEBD algorithm with the transverse Ising Hamiltonian and evolve towards the ground

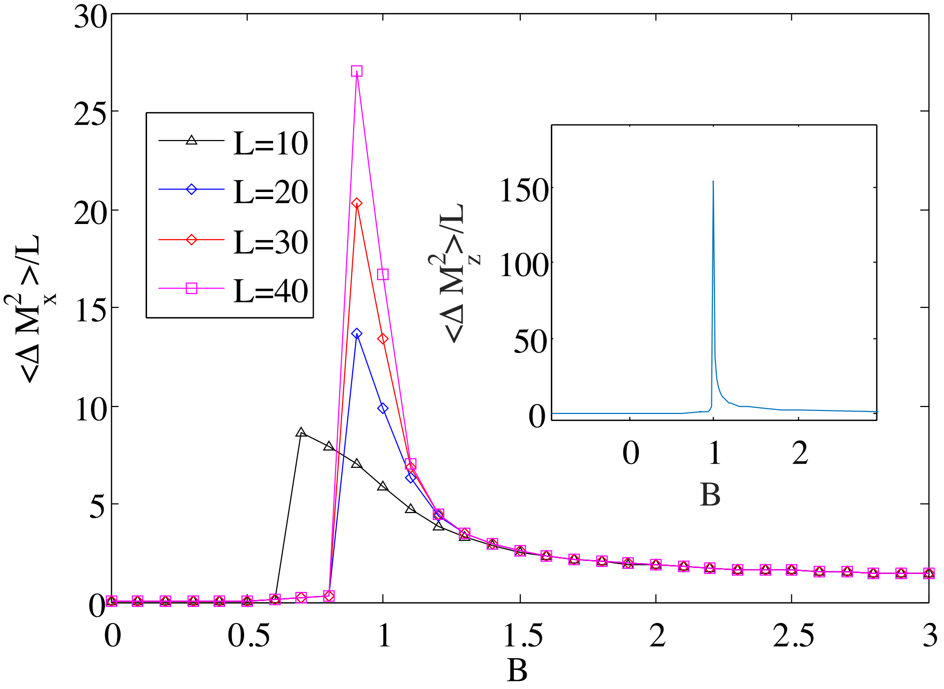

Figure 4.6: (Color online) Per-site value of the second cumulant of the longitudinal magnetization, $\frac{1}{L}\langle \Delta M_x^2 \rangle = \frac{1}{L}(\langle M_x^2 \rangle - \langle M_x \rangle^2)$, for the transverse Ising model. The cumulant is plotted for various finite system sizes, plotted against a range of applied fields. As the system length increases, the behavior tends towards the infinite limit (inset). In the limit, the cumulant diverges at the critical point.

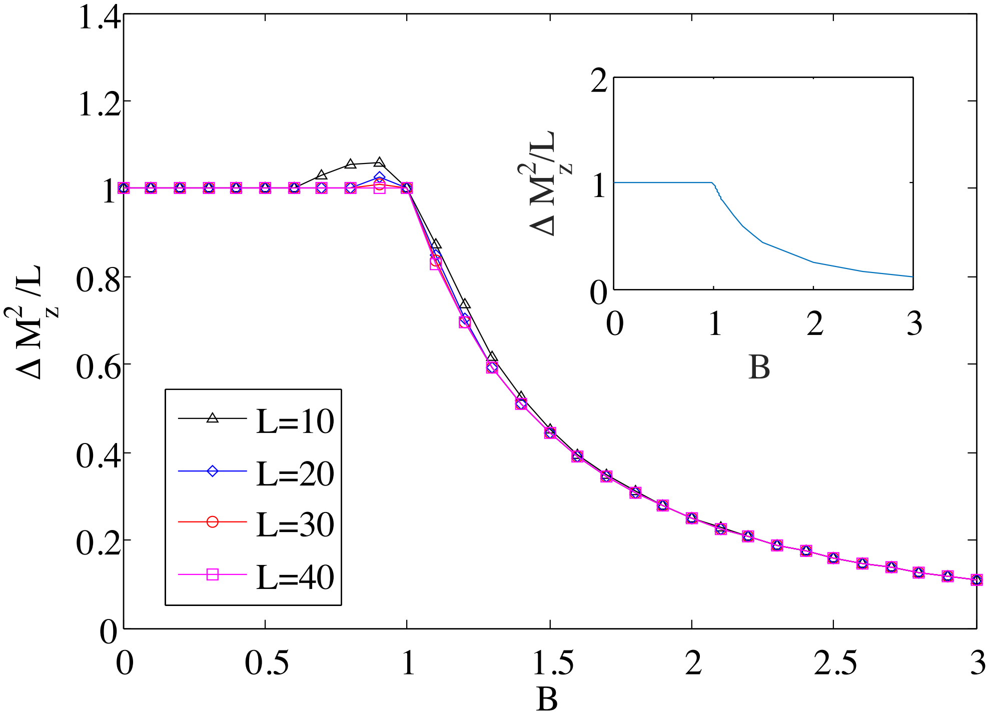

Figure 4.7: (Color online) Second cumulant of the transverse magnetization, $\langle \Delta M_z^2 \rangle = \langle M_z^2 \rangle - \langle M_z \rangle^2$, for the transverse Ising model (computed per site). The cumulant is plotted for various finite system sizes, plotted against a range of applied fields. As the system length increases, the behavior tends towards the infinite limit (inset) where the derivative of the cumulant shows a discontinuity at the critical point. This behavior is in excellent agreement with the analytical result for the thermodynamic limit, that $\langle \Delta M_z^2 \rangle / L = 1$ for $B < 1$ and $1/B^2$ for $B > 1$.

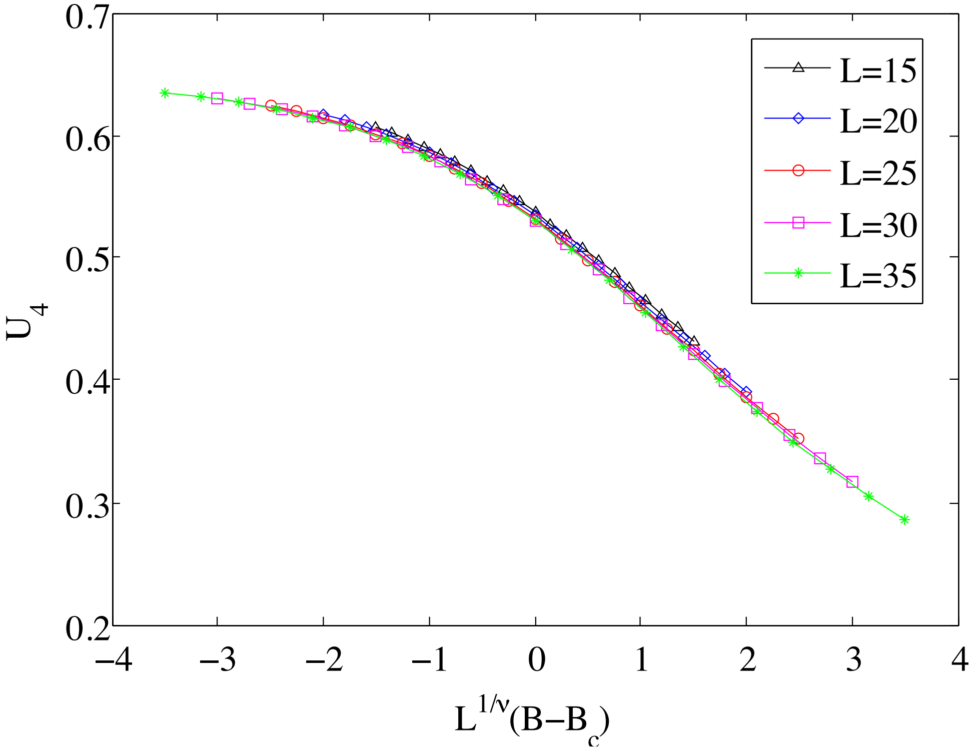

Figure 4.8: (Color online) The Binder cumulants for the transverse Ising model, plotted for a variety of system sizes as a function of $L^{1/\nu}(B - B_c)$ for the known values $\nu = 1$ and $B_c = 1$. As expected, for these values the curves are seen to collapse to a functional form essentially independent of the length scale. This property can be used to estimate the values of the critical point and the critical exponent by treating them as fit parameters and optimizing the collapse.

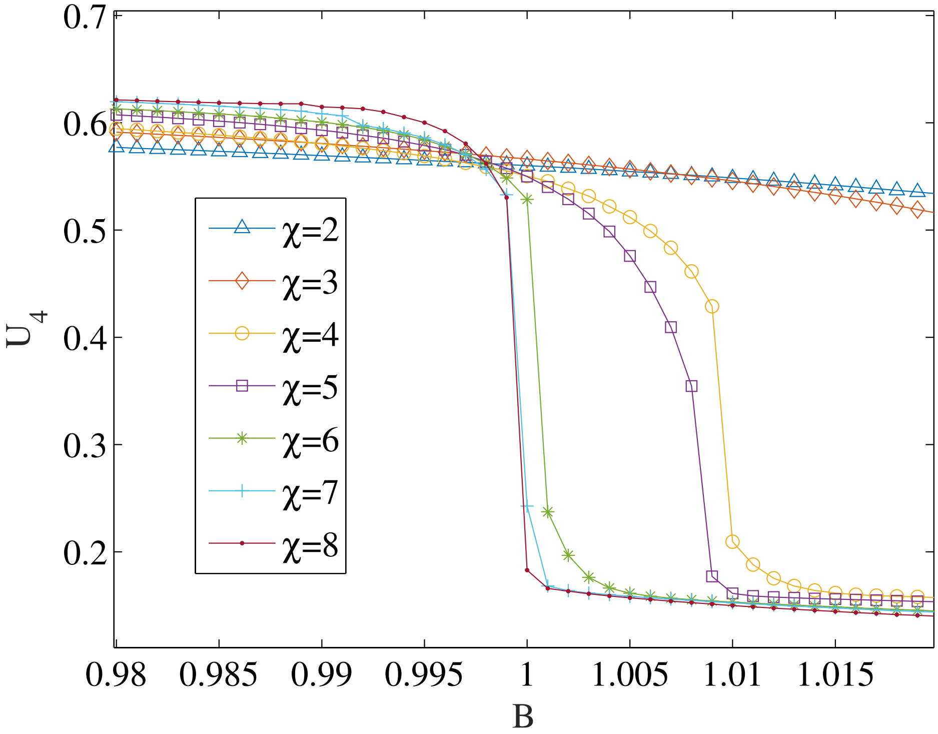

Figure 4.9: (Color online) The Binder cumulants for a finite sub-block of the infinite transverse Ising model, plotted for a variety of bond dimensions χ. For each bond dimension, the apparent correlation length of the ground state, ξ, is used to set the size of the block. The scaling behavior of these blocks as χ increases gives an alternative method for identifying the critical point, as well as allowing an estimate of the critical Binder cumulant ($B_c = 0.998$ and $U_4^* = 0.57$ for this data).

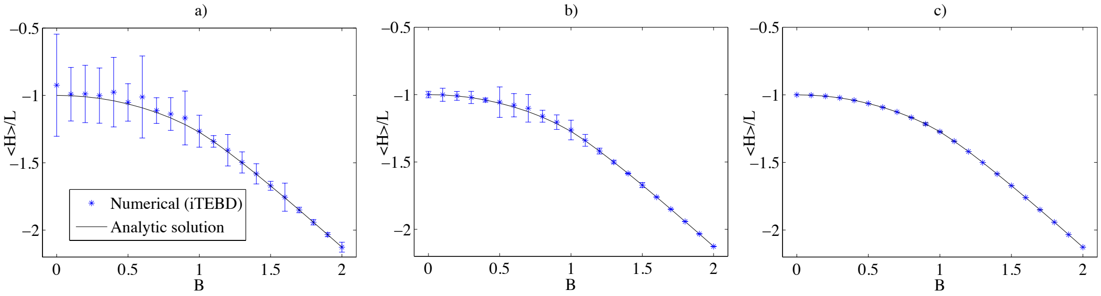

Figure 4.10: (Color online) The energy of the spin-1/2 transverse Ising model on a chain of length 10 is calculated using approximate ground states generated by the TEBD algorithm ($\chi = 20$). The numerical data (points) are plotted alongside the exact solution (line). Error bars are calculated from $\epsilon = \sqrt{\langle \Delta H^2 \rangle}$, with $\langle \Delta H^2 \rangle$ the second cumulant of the energies. (a) After 10 steps, the energies are still noisy and the error bars are quite large. (b) After 100 steps, the error bars have clearly decreased, and are largest for the points with the largest discrepancies from the exact solution. (c) By 1000 steps, most error bars are within the size of the data points, and the approximate energies are very close to the known analytical result.

state, over a range of field strengths B. As shown in Fig. 4.10, the results are initially somewhat noisy when compared to the analytical E vs. B curve, which is reflected by the large error bars computed from $\langle \Delta H^2 \rangle$. However, as the algorithm continues, these error bars shrink and eventually become essentially zero, signaling the complete convergence of the energies.

4.4.2 Spin-1 Transverse Ising Model

We consider next the spin-1 generalization of the Ising model, with Hamiltonian

$$H = -\sum_{j}^{L} S_j^x S_{j+1}^x + B S_j^z. \tag{4.60}$$

Here, we have simply replaced the spin-1/2 Pauli matrices from Eq. (4.58) with their spin-1 counterparts. This model is of interest because unlike the spin-1/2 case, it has no exact analytic solution. Nevertheless, in the thermodynamic limit the magnetization is qualitatively similar to the spin-1/2 case. Notably, it still displays a quantum phase transition at a critical value of the transverse field, which has been studied by various numerical methods[163, 164].

The accepted value for this critical field is $B_c = 1.326$ [163].

We study this transition point with the same techniques as before, calculating the Binder cumulants from the second and fourth moments of the x-magnetization for various system sizes (Fig. 4.11). The ground states are calculated using the same finite MPS technique and with a bond dimension of $\chi = 20$. The successive crossings are then compared and a limiting value extrapolated using BST. In this case, we compute an estimate of the transition at $B_c = 1.327(1)$.

Once again, it is also constructive to consider the second cumulant on its own. Numerical calculations of the magnetization for this model invariably show some finite size effects around the transition, producing a finite "tail" near the transition point, which makes an exact determination difficult using the order parameter alone [165]. But the transition appears much more sharply as a singularity when we consider the second cumulant, as in Fig. 4.12 (higher cumulants such as κ_4 can also be used for this purpose). From this quantity, we obtain $B_c = 1.324(2)$, an estimation which is to within less than 0.2%.

As before, can also study the critical exponent ν, known for this model to be the same as the spin-1/2 case, $\nu = 1$. As a proof of principle, the "data collapse" for the known values of ν and B_c are shown in Fig. 4.13. As expected, for these values the curves are seen to collapse to a functional form essentially independent of the length scale. This property can be used to estimate the values of the critical point and the critical exponent by treating them as fit parameters and optimizing the collapse.

4.4.3 Spin-1 Ising Model in Crystal Field

For another application, we consider also a variation on the spin-1 Ising model, where the usual transverse field has been replaced by a quadratic, crystal field term, to give the following Hamiltonian

$$H = -\sum_{j}^{L} S_j^x S_{j+1}^x + B(S_j^z)^2. \tag{4.61}$$

This variation of the spin-1 model admits a mapping to the spin-1/2 case (c.f. [166]), from which the critical point of $B_c = 2$ can be analytically obtained. To compare our method, we perform the same numerical calculations as above: first generating ground states at various finite lengths using MPS methods with a bond dimension of $\chi = 10$, and then computing the Binder cumulants. From the Binder curves (see Fig 4.14) we once again perform the BST extrapolation of the successive crossings to arrive at an estimate of $B_c = 1.999(1)$.

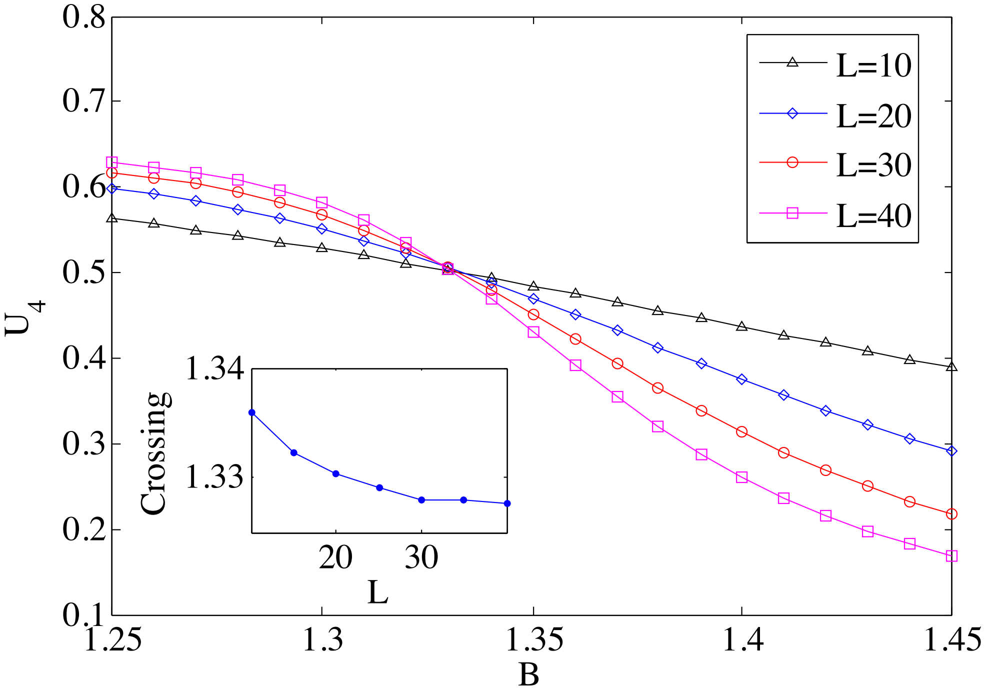

Figure 4.11: (Color online) A Binder cumulant study of the spin-1 transverse Ising model. As above, the cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e. L = 10 and L = 15, and the BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.327(1)$.

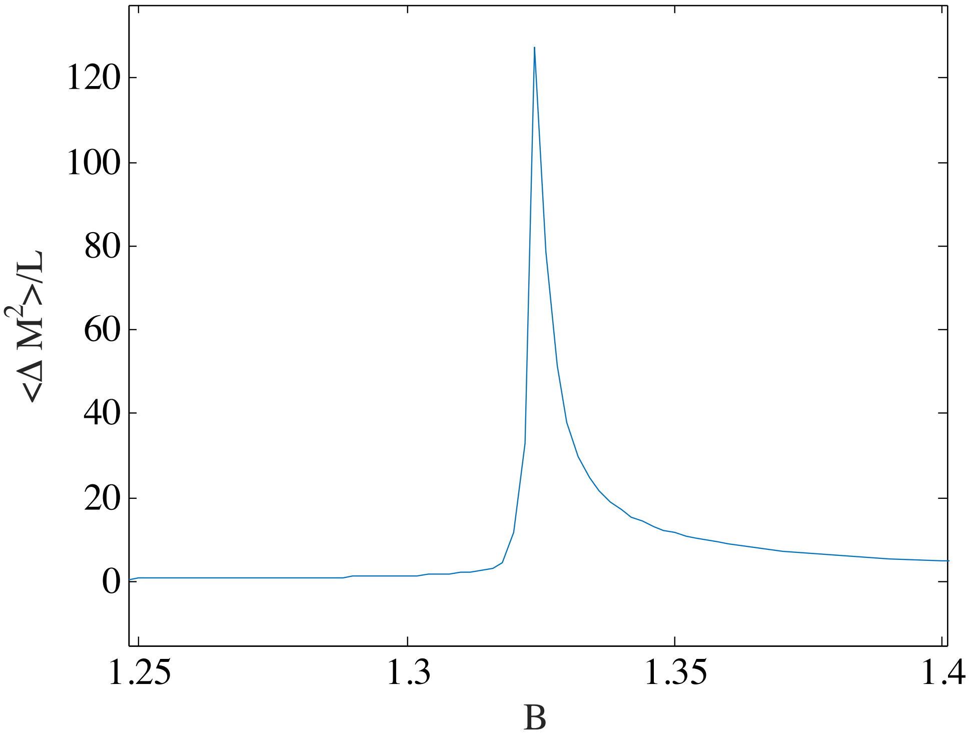

Figure 4.12: Per-site value of the second cumulant of the longitudinal magnetization, $\frac{1}{L}\langle \Delta M_x^2 \rangle = \frac{1}{L}(\langle M_x^2 \rangle - \langle M_x \rangle^2)$, computed for the Spin-1 Ising model. The cumulant is calculated for an infinite system directly.

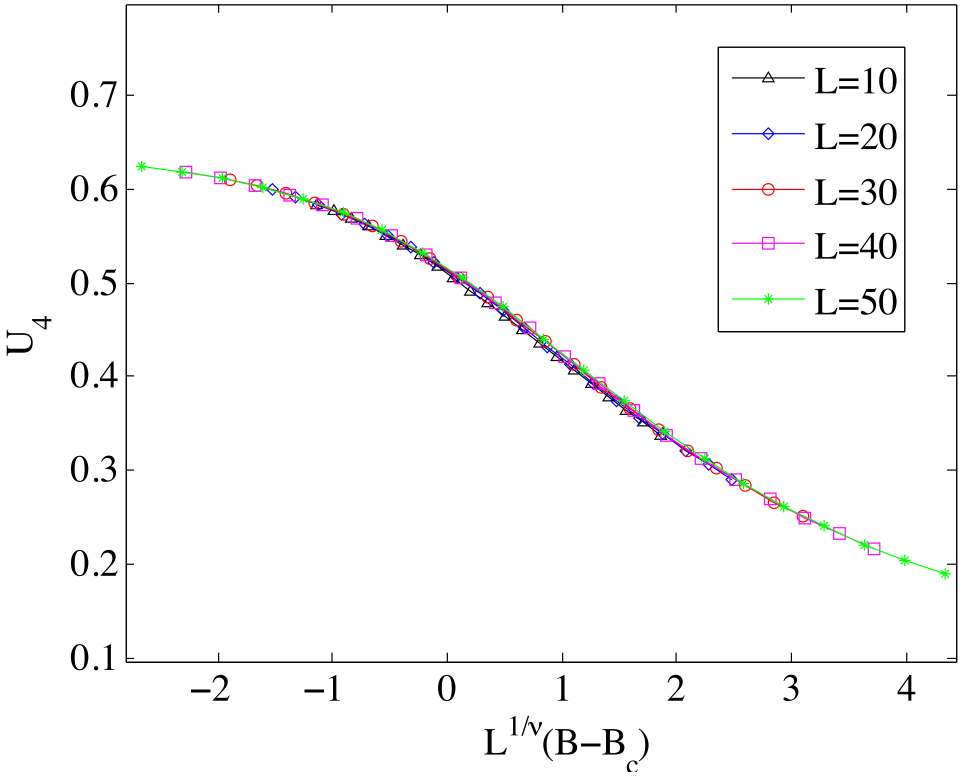

Figure 4.13: (Color online) The Binder cumulants for the spin-1 transverse Ising model, plotted for a variety of system sizes as a function of $L^{1/\nu}(B - B_c)$ for the known values $\nu = 1$ and $B_c = 1.326$. The length-independence of the curves allows this technique to be used as a means to estimate both ν and B_c

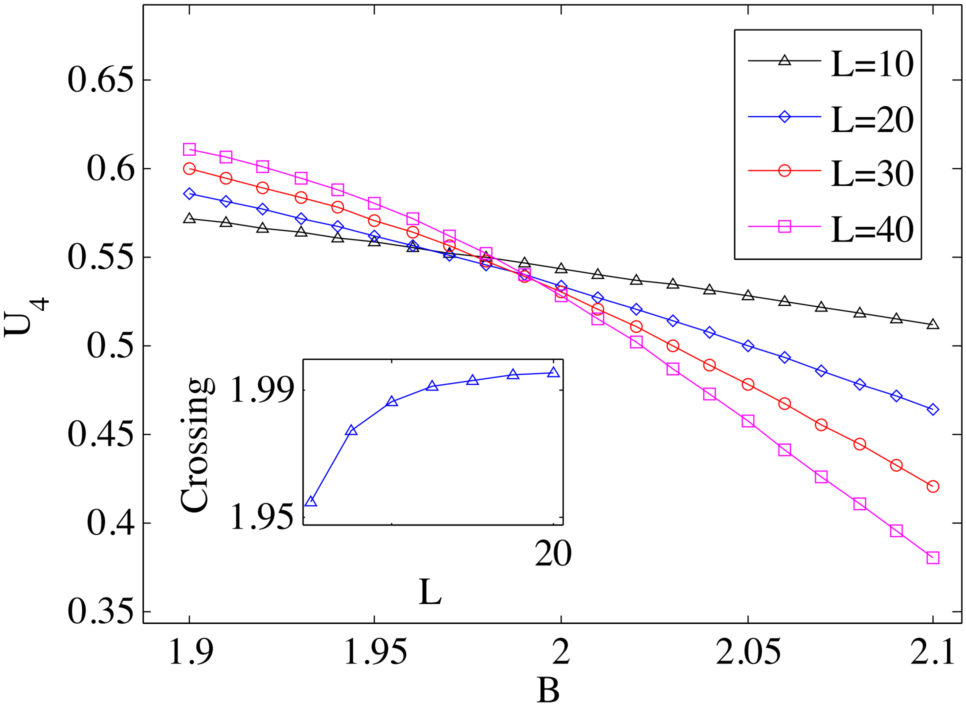

Figure 4.14: (Color online) A Binder cumulant study of the spin-1 transverse Ising model with crystal field. As above, the cumulants are computed for different system sizes across a range of values for the transverse field B (some intermediate system sizes have been suppressed for clarity of the figure). Crossing points are interpolated for successive pairs of curves, i.e. L = 10 and L = 15, and the BST algorithm is used to extrapolate these values to the infinite limit, which gives $B_c = 1.999(1)$.

Direct examination of the second cumulant in the infinite system is also still a viable method for estimating the transition point. In this case, the location of the maximum gives $B_c = 1.996(1)$ (see Fig. 4.15). As before, it also remains possible to study the critical exponent ν and critical field value simultaneously by seeking to collapse the data to its universal behavior as a function of $L^{1/\nu}(B - B_c)$ (Fig. 4.16).

4.4.4 Spin-1/2 Ising Model on a 2D Lattice

Finally, as a proof-of-principle, we briefly demonstrate the application of these methods to a two-dimensional system: the spin-1/2 Ising model on a 2D

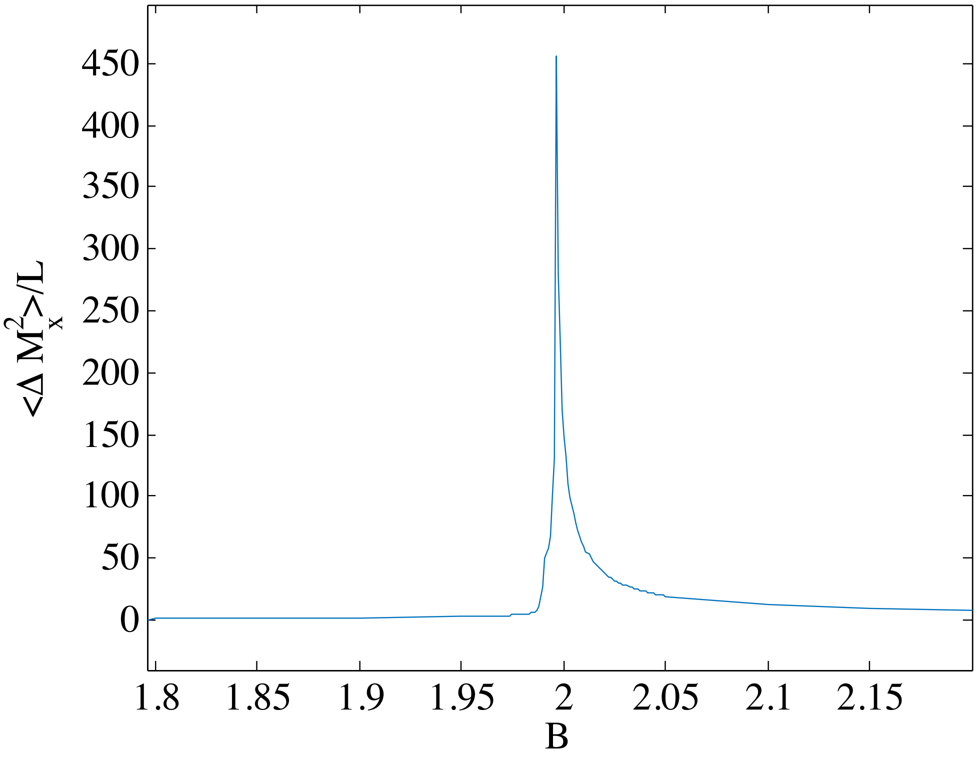

Figure 4.15: Per-site value of the second cumulant of the longitudinal magnetization, $\frac{1}{L}\langle \Delta M_x^2 \rangle = \frac{1}{L}(\langle M_x^2 \rangle - \langle M_x \rangle^2)$, computed for the spin-1 Ising model with a transverse crystal field. The cumulant is calculated for an infinite system.

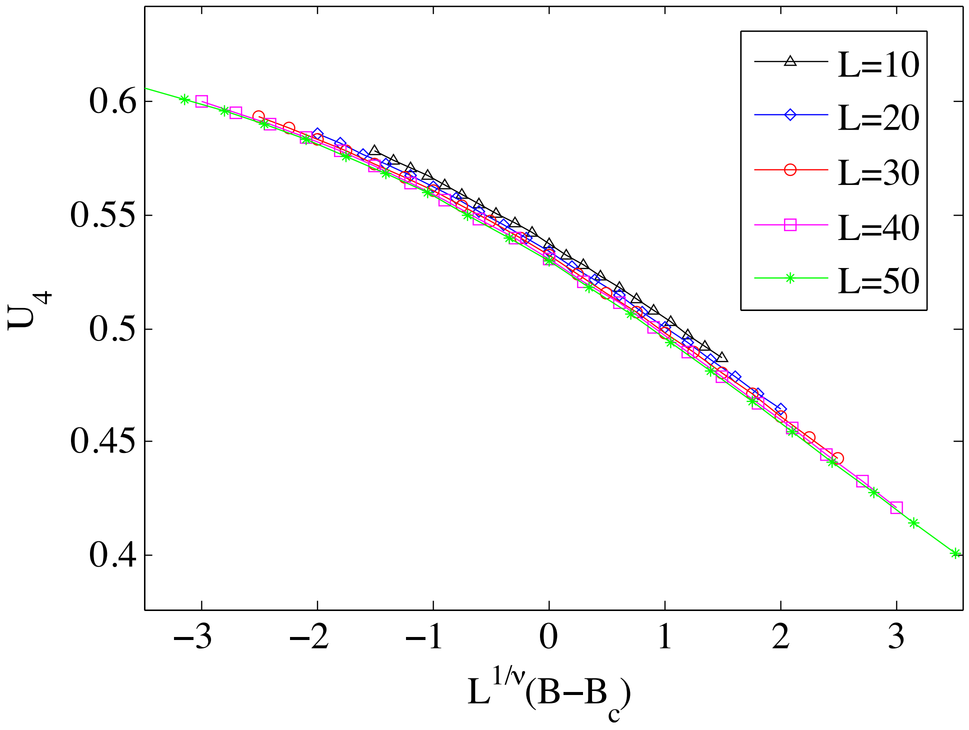

Figure 4.16: (Color online) The Binder cumulants for the spin-1 Ising model with crystal field, plotted for a variety of system sizes as a function of $L^{1/\nu}(B-B_c)$ for the known values $\nu = 1$ and $B_c = 2$. As expected, for these values the curves are seen to collapse to a functional form essentially independent of the length scale.

square lattice. This system is described by the Hamiltonian

$$H = -\sum_{j,k} \sigma^x_{j,k}(\sigma^x_{j+1,k} + \sigma^x_{j,k+1}) + BS^z_{j,k}, \tag{4.62}$$

where the subscripts are understood to terminate at the boundary of the system. As discussed above, the study of two dimensional systems with tensor network states is considerably more involved than the study of one-dimensional systems. More elaborate methods must be undertaken to numerically approximate the ground states, and elaborate approximation schemes must be used in order to calculate expectation values, which are otherwise prohibitively costly in time and memory. A detailed and high-precision study of the critical point of this model is therefore beyond the scope of this project (see instead [33]). Nevertheless, we include the following rough estimation in order to demonstrate how easily the moment-generating function method can be generalized to higher dimensional states, as well as to underscore the utility of Binder cumulant techniques even for data calculated relatively cheaply.

To this end, we generate approximate ground states for the model, using a simple method of local updates (a 2-D generalization of TEBD) and the smallest nontrivial bond dimension, $\chi = 2$. To check the behavior of our system, we consider the order parameter

$$M = \sum_{j,k} \sigma^x_{j,k}. \tag{4.63}$$

Then, as in the one-dimensional case, we compute the Binder cumulant of the order parameter across a range of applied fields, for systems of size $L \times L$ up to $L = 12$, and observe the crossings (Fig. 4.17). The largest crossing we are able to compute, between $L = 10$ and $L = 12$, occurs at $B = 3.11(1)$, which is already reasonably accurate compared to the accepted value of $B_c = 3.044$, as calculated by quantum Monte Carlo [167]. A BST Extrapolation of the data gives $B_c = 3.3(2)$, a crude estimation but with relatively large error bars, owing largely to the fact that only three crossing values have been used in the extrapolation. We note also that, unlike the case of one-dimensional systems, the Binder crossings for two-dimensional systems are not necessarily converging monotonically and hence may not necessarily admit an easy extrapolation. Instead, greater precision could likely be obtained through the use of more sophisticated two-dimensional methods (or additional computational resources) to study the crossings for slightly larger system sizes.

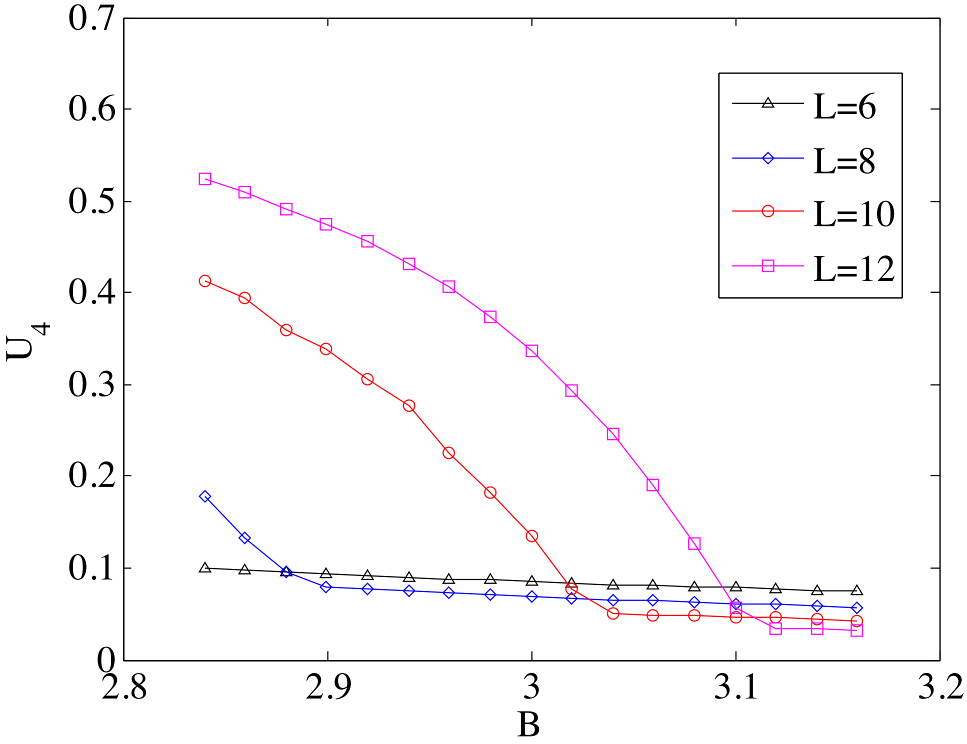

Figure 4.17: (Color online) A rough Binder cumulant study of the spin-1/2 transverse Ising model on a square lattice, using a local-update numerical algorithm with bond dimension $\chi = 2$. As in the one-dimensional case, the cumulants are computed for different system sizes across a range of values for the transverse field B. The largest crossing point, between $L = 10$ and $L = 12$, is at $B = 3.11(1)$. Extrapolating to the infinite limit gives $B_c = 3.3(2)$, though this cannot be done with high reliability on such a limited dataset (see text).

4.5 Summary

In this chapter we have presented a method for efficiently calculating the higher order moments and cumulants of general operators on systems represented by tensor network states. For finite systems, this capability has a variety of applications in the search for phase transitions in quantum systems. Chief among these is the calculation of the celebrated "Binder cumulant," which provides a powerful tool for not only detecting phase transitions, but determining their location to a high degree of accuracy using only relatively small finite systems to probe the infinite limit. The finite size scaling of the Binder cumulant also provides an estimate the critical exponent of the correlation length. Although the second cumulants of Hamiltonians have been considered in the context of matrix product states, to our knowledge, critical point detection techniques based on the Binder cumulant (or cumulants in general) have not generally been put to use in studies based on tensor networks, despite being widely applied to classical systems and quantum Monte Carlo studies. It is our hope that the methods presented in this chapter will allow them to be embraced by the tensor network community as well.

In the case of infinite systems, we also present a method to calculate the per-site limits of the cumulants efficiently as well. The higher cumulants of an order parameter often show sharp behavior at the critical points, which in many cases allows for easier detect than the changes in the order parameter itself. In particular, we show how singularities in the second cumulant can produce a relatively precise (computationally cheap) estimation of the location of the transition. All the techniques (finite and infinite) are demonstrated in the context of the transverse Ising model, the spin-1 transverse Ising model, the Ising model in a crystal field, and could easily be applied to other models. We also demonstrate a useful application of the second cumulant of the energy. This quantity, which we calculate for both finite and infinite systems, provides a useful sufficient condition to determine when a numerical ground-state estimation algorithm has converged. As we demonstrate in the context of the Ising model, it can identify convergence up to a very high level of precision.

Finally, we present a proof-of-principle demonstration of the methods as applied to the transverse Ising model on a square lattice. Our result demonstrates that the methods for computing moments and cumulants are easily generalized to states in two dimensions or higher. Precise calculation of the moments and cumulants of a such a system may be more difficult, since state preparation and the process of computing expectation values are themselves much more complicated in higher dimensions. However, the central idea of our method on its own remains just as straightforward as in one dimension.

During the preparation of this work, it was brought to our attention that the technique based on matrix product operators (described above) has also been suggested as an alternative method for evaluating cumulants and moments, particularly in the context of the second cumulant of the energy and its use as a convergence check. We believe both methods have complimentary strengths and weaknesses, depending on the context in which they are applied.

Chapter 5

Detection of Gapped Phases of a 1D Spin Chain with Onsite and Spatial Symmetries

In this chapter, we turn our attention to the second area unique quantum many-body phenomena: the classification of phases of matter beyond the model of spontaneous symmetry breaking. To that end, we investigate the phase diagram of a quantum spin-1 chain whose Hamiltonian is invariant under a global onsite A_4, translation and lattice inversion symmetries. We detect different gapped phases characterized by not only symmetry breaking, but also so-called "symmetry protected topological order," using a well-established set of matrix product state order parameters. We observe a rich variety of phases of matter characterized by a combination of symmetry breaking and symmetry fractionalization and also the interplay between the onsite and spatial symmetries. Examples of continuous phase transitions directly between topologically nontrivial SPT phases are also observed.

5.1 Introduction

In the previous chapter, we applied tensor network techniques to compute cumulants of operators, motivated in part by the desire to detect phase transitions and measure their associated order parameters and critical exponents. The view of phase transitions presented there followed the approach developed by Landau and Ginzburg [38–40, 168], which for many years was the prevailing understanding of the concept of a "phase" or "phase transition." In this picture, one considers the possible ground-state phases of a Hamiltonian by considering the global symmetry group G. Ground states *may* share the symmetry of the Hamiltonian, but may instead spontaneously break some or all of the symmetry. The symmetry subgroup H which may remain (including perhaps the trivial subgroup in the case of total symmetry breaking) characterizes the nature of the ground state phase.

While many of these concepts still hold, we now know that the spontaneous symmetry-breaking (SSB) picture, powerful though it may be, is incomplete [169]. The first evidence of this was provided by the discovery of the quantum Hall effect and its subsequent study[170–172]. For example, the fractional quantum Hall states were found clearly to behave as existing in distinct phases, despite the fact that they all share the same set of symmetries [172]. Similar systems with this property subsequently emerged such as chiral spin states [173, 174], string nets [175] and quantum double models [95]. Such states are referred to as "topological phases," in that the information required to characterize such a phase cannot be extracted from a local order parameter, but instead, in an abstract sense, is encoded in the global topology of the state.

To understand these new phases, it is beneficial to consider two alternate definitions of what it means for two ground states to be in "different phases." As a concrete example, consider two states $|\psi_1\rangle$ and $|\psi_2\rangle$ with associated parent Hamiltonians H_1 and H_2. In the traditional perspective on quantum phase transitions, the ground states are said to be in the same phase if it is impossible to construct a parameterized Hamiltonian $H(\lambda)$ that connects H_1 and H_1 without becoming gapless at some point in between [130]. But it has been shown to be equivalent to say that the states are in the same phase if they can be deformed into one another through the action of local unitary operators alone [176, 177].

In this definition the behavior of topological phases is much more obvious; local unitary transformations cannot change global entanglement properties, so states with different global structure can certainly be in different phases, regardless of whether they share a symmetry group. It has also been shown that in gapped, one dimensional spin systems, no topological phases can occur, since *any* one dimensional state can be deformed to a product state by local unitary operations [46, 49–53], and therefore any two states can be deformed to one another. But suppose this deformation is not possible without breaking the symmetry of the state along the way. This may occur even when the two states share the same symmetry group! In such case, we can still consider there to be a phase boundary between the states, but one which must be "protected" by the symmetry. Since these phase distinctions also cannot be detected by local order parameters, they are also reminiscent of the true topological phases found in higher-dimensional systems. In light of these properties, such phases are called *symmetry protected topological* (SPT) phases [46–48]. The "Haldane Phase" in spin-1 chains [92, 178, 179] is a famous example of a nontrivial SPT phase.

The classification of phases in one dimensional systems, both SSB and SPT has been extensively studied, and the formalism of matrix product states can play a key role in understanding the different phases which are possible [49–53]. In this chapter, we explore the MPS techniques which can be used to numerically characterize the phases of of a Hamiltonian, building off of the methods introduced in Ref. [52]. In particular we construct and study the phase diagram of a two parameter Hamiltonian in a spin-1 chain which is invariant under a global onsite (internal) A_4 symmetry, lattice translation and lattice inversion (parity). Through suitable order parameters, we detect both the different SSB and SPT phases and label them using the classification framework of Ref. [50]. A total of eight distinct phases are identified within the parameter space we consider. In particular, we find among these a direct, continuous transition between two topologically nontrivial A_4-symmetric SPT phases, distinguished by the 1D representations of the symmetries, as explained below. Along the way, we also present important numerical techniques which are useful for applying the SPT classification scheme in practice, including a method to restore one-site translation invariance to an MPS computed by the iTEBD algorithm (recall that this algorithm necessarily requires at least a two-site MPS representation; see Sec. 3.2.2).

This chapter is organized as follows. In section 5.2, we described the A_4 spin-chain Hamiltonian studied here and present its phase diagram which contains the main results of this work. In section 5.3, we review the classification of 1D gapped-spin chains and list parameters which can be used to completely classify phases. In section 5.4, we describe the full details of the phase diagram of the A_4 model, and also enumerate the several possible phases that can in principle exist given the symmetry group of the parent Hamiltonian. Section 5.5 presents, in detail, the numerical MPS techniques by which the states and parameters were obtained, and section 5.6 gives a summary of our results.

5.2 Overview of Main Results

5.2.1 The Hamiltonian

We will now describe an A_4 and inversion symmetric Hamiltonian whose phase diagram we study in detail. The Hamiltonian we present here is a modified version of the one used in Ref [180] where it was found that the there was an extended region where the ground state is exactly the AKLT state and hence useful for single qubit quantum information processing [181]. Here, we slightly modify the Hamiltonian to retain the essential features only and study the phase diagram.

The total Hamiltonian consists of three parts. The first is the Hamiltonian for the spin-1 Heisenberg antiferromagnet which is invariant under the spin-1 representation of $SO(3)$:

$$H_{Heis} = \sum_i \boldsymbol{S}_i \cdot \boldsymbol{S}_{i+1}, \tag{5.1}$$

$$\tag{5.2}$$

where $\boldsymbol{S}_i \cdot \boldsymbol{S}_{i+1} \equiv S_i^x S_{i+1}^x + S_i^y S_{i+1}^y + S_i^z S_{i+1}^z$. We add two other combinations, H_q and H_c to the Heisenberg Hamiltonian which breaks the $SO(3)$ symmetry to A_4, the alternating group of degree four and the group of even permutations on four elements (equivalently, the rotation group of a tetrahedron). These terms are defined as:

$$H_q = \sum_i (S_i^x S_{i+1}^x)^2 + (S_i^y S_{i+1}^y)^2 + (S_i^z S_{i+1}^z)^2,$$

and

$$\begin{aligned} H_c = \sum_i [&(S^x S^y)_i S_{i+1}^z + (S^z S^x)_i S_{i+1}^y + (S^y S^z)_i S_{i+1}^x \\ &+ (S^y S^x)_i S_{i+1}^z + (S^x S^z)_i S_{i+1}^y + (S^z S^y)_i S_{i+1}^x \\ &+ S_i^x (S^y S^z)_{i+1} + S_i^z (S^x S^y)_{i+1} + S_i^y (S^z S^x)_{i+1} \\ &+ S_i^x (S^z S^y)_{i+1} + S_i^z (S^y S^x)_{i+1} + S_i^y (S^x S^z)_{i+1}]. \end{aligned} \tag{5.3}$$

For details on how the perturbations are constructed, see Refs. [180, 182]. The operators in H_c are symmetrized so that the Hamiltonian is invariant under inversion as well as lattice translation. With this we have a two-parameter Hamiltonian invariant under an onsite A_4 symmetry along with translation invariance and inversion.

$$H(\lambda, \mu) = H_{Heis} + \lambda H_c + \mu H_q. \tag{5.4}$$

5.2.2 Summary of Numerical Results

We employ the iTEBD algorithm[28] to numerically analyze the ground states across a range of parameters $\mu = [-3, 4]$, $\lambda = [-2, 2]$ and find a wide

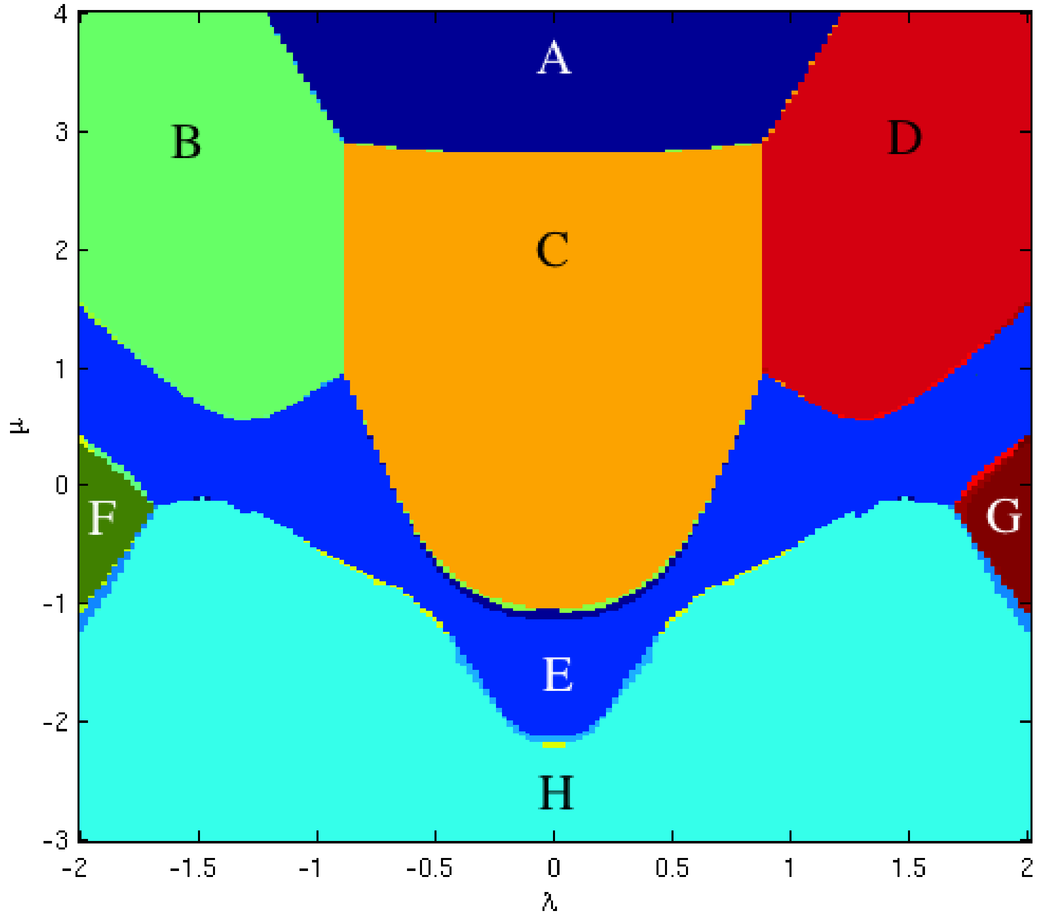

Figure 5.1: (Color Online). The phase diagram for a two-parameter Hamiltonian constructed to have an A_4 onsite symmetry group, as well as parity symmetry and one-site translation invariance. The symmetries of the Hamiltonian break down into five different residual symmetry groups in the ground states. These break down further when classified according to the relevant topological parameters, yielding eight distinct phases overall. The diversity of phases from the comparatively simple Hamiltonian shows the necessity of carefully accounting for all possible symmetries and topological parameters when attempting to characterize the phase of a ground state. For a description of the phases **A-H**, see discussions in the main text.

variety of phases. In the parameter space analyzed, a total of eight distinct regions can be identified (labeled with letters **A-H** in Fig. 5.1). These regions are distinguished both by the symmetries of the ground states, and also by the classification parameters of Ref [50].

From the symmetry group G of the parent Hamiltonian, which contains A_4, spatial inversion, and translation symmetries, only the inversion and translation symmetries remain in the ground states of region **A**. Regions **B**, **C**, and **D**, by contrast, all respect the full set of symmetries of the parent Hamiltonians but are differentiated by one of the SPT parameters: namely, the overall complex phase produced under A_4 transformations. These complex phases are different 1D irreducible representations (irreps) of A_4 and correspond to distinct SPT phases protected by translation and onsite symmetries. In phase **E**, the ground state breaks the symmetry to onsite $\mathbb{Z}_2$ and parity. The translation symmetry in this region is broken down from single-site translation invariance to two-site. This broken, two-site translation symmetry is also present in regions **F** and **G**, but here the remaining symmetries of A_4 and parity are completely preserved. Like regions **B**, **C**, and **D**, regions **F** and **G** have the same symmetry but are distinguished from one another only by the values of their SPT parameters. Finally, in region **H**, the residual symmetry group has an internal $\mathbb{Z}_2 \times \mathbb{Z}_2$ symmetry and parity along with an one-site translation invariance.

Among these eight phases, five correspond to instances of SSB and the remaining three correspond to SPT phases without symmetry breaking. The complete set of such parameters classifying these phases will be described in section 5.3, and the particular values which distinguish them from one another are presented in section 5.4.

Because the phases **B**, **C**, and **D** are not distinguished by any symmetry-breaking criteria (and because none of them are topologically trivial), the boundary lines between them are of particular interest as examples of non-trivial SPT to non-trivial SPT phase transitions. Such transitions are considered uncommon and have recently attracted particular interest[183–186], as compared to the more typical case of a transition between SPT and symmetry breaking phases, or trivial to non-trivial SPT phase transitions. Our analysis, however, shows that this model contains direct nontrivial SPT to SPT transitions, and that the transition is second-order in nature. By directly calculating the ground-state energy and its derivatives, we see sharp divergences in the second derivative, but a continuous first derivative across the boundary between these phases. Representative behavior is shown in Fig. 5.2.

The numerical methods employed here also allows us to probe the central charge of the conformal field theory (CFT) associated with the continuous

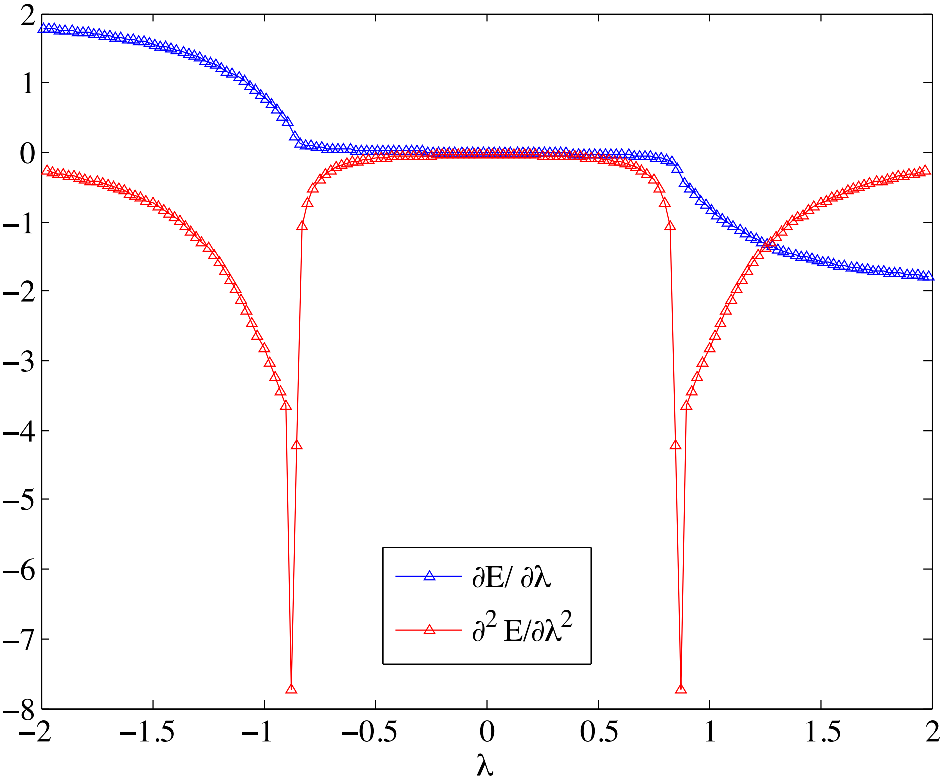

Figure 5.2: (Color Online). Free energy derivatives along the line $\mu = 2$ in the phase diagram above show the nature of the phase transitions. The continuous first derivative (blue) contrasts with divergence in the second derivative (red), showing a second-order transition. All three regions are topologically nontrivial SPT phases. Data shown here were computed with a bond dimension of 30, and the behavior has been seen to be stable as the bond dimension increases.

phase transitions. As one approaches the transition, the correlation length begins to diverge. The central charge of the CFT appears in an important scaling relation between this diverging correlation length and the mid-bond entanglement entropy [187, 188]. In particular, it has been shown that up to an additive constant,

$$S = \frac{c}{6} \log \xi \tag{5.5}$$

where c is the central charge, and ξ is the correlation length measured in units of lattice spacing. S is the entanglement entropy, given by performing a Schmidt decomposition between sites and computing the entropy of the resulting Schmidt coefficients λ_i,

$$S = -\sum_i \lambda_i \log \lambda_i. \tag{5.6}$$

The MPS algorithms employed here to determine the ground state are not well-suited to computing ground states at the actual critical points. This is because the numerical accuracy of these algorithms are controlled by a tunable numerical parameter, the so called "bond dimension." The closer we approach the critical point, the bigger this parameter needs be chosen for the ground states to be computed faithfully. By gradually increasing the bond dimension near the critical point, we obtain states with increasingly large correlation length, allowing us to fit the scaling relation of Eq. 5.5. We can also use this data to estimate the location of the transition, because away from the critical point, the scaling relation will not hold, and S will saturate for large enough ξ (or in practice, for large enough bond dimension). We find the critical lines to be located at $\lambda = \pm 0.865(2)$; fits at multiple points along these lines suggest a central charge of $c = 1.35(1)$, as shown for example in Fig. 5.3.

5.3 Review of Classification System

We now review the classification of 1D gapped phases of spin chains following [50]. Given the group of global symmetries G, the classification gives us a set of labels whose values distinguishe all possible phases of matter that can exist. We will systematically list these labels for various types of symmetries. It is the value of these labels that we extract numerically to determine the phase diagram presented in Sec 5.2. In keeping with the theme of this dissertation, the classification system is cast here entirely in the language of MPS, which allows us to bring to bear all the numerical MPS techniques which have been previously discussed.

Two important features of the MPS representation bear relevance to the

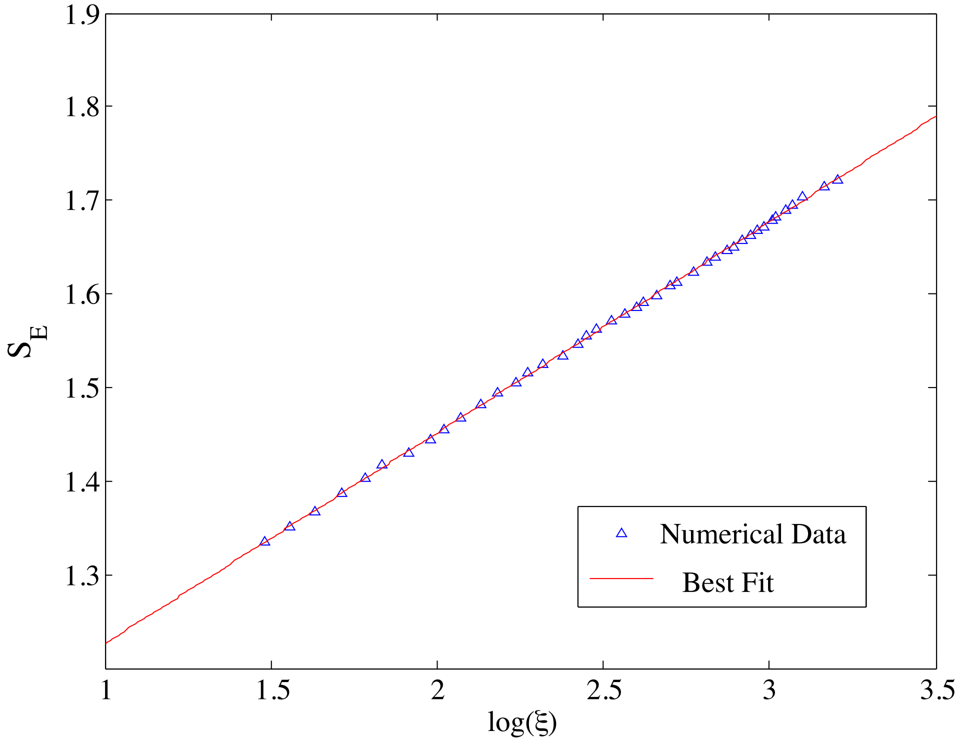

Figure 5.3: (Color Online). Entanglement entropy versus the log of the correlation length for states very close to the transition point. The slope is directly proportional to the central charge of the associated CFT, via Eq. 5.5. Data is generated by computing ground states at the point $\mu = 2, \lambda = 0.865$, and increasing the "bond dimension" of the numerical scheme to allow us to find states closer to the critical point where the correlation length diverges. The behavior shown here is representative of that seen elsewhere along the lines $\lambda = \pm 0.865$. Away from these lines, the entanglement entropy saturates at a finite value of ξ. The best-fit line has a slope of 0.225(1), which corresponds to a central charge of 1.35(1).

numerical methods employed in this paper. The first is the bond dimension of the representation. As discussed in Chapter 3, if the wavefunction is the ground-state of a gapped Hamiltonian and hence has a finite correlation length, then it can be efficiently written as an MPS wavefunction whose bond dimension approaches a constant value that is independent of the size of the chain [12, 81, 189]. However, as one approaches a critical point, where the correlation length diverges, an increasingly large bond dimension is required to faithfully capture the ground states. As a result, even though the ground states at criticality therefore cannot be accurately represented by an MPS, one can employ the scaling results discussed above and in Fig. 5.3, where increasingly large correlation lengths are probed by gradually increasing the bond dimension. Note that while we have typically used the variable "χ" to refer to the bond dimension of an MPS, in this Chapter we shall use the name "D," to avoid confusion with the SPT parameter χ to be introduced below.

The second property to recall is that when a state possesses translation invariance, the MPS matrices themselves may be chosen to respect the same symmetry. A state invariant under one-site translations, for example, can be represented in the form above with the same MPS tensor at each site, $A_{[j]}^{s_j} = A^{s_j}$. This, in turn, allows a state with translation invariance of any length to be represented by d matrices where d is the dimension of the local Hilbert space. In general, a state with K-site translation invariance requires Kd MPS matrices to represent it. There are times, however, when a larger number of matrices may be required, for example, when a state is being studied by a numerical algorithm which is not itself invariant on an N-site level. This, indeed, will frequently be the case for our states, as discussed in detail in sections 5.5.1 and 5.5.2.

5.3.1 Symmetry Breaking

Before considering the possibility of "exotic" SPT phases, we begin by classifying phases as we would under the old Landau-Ginzburg picture. We ask whether or not the global symmetry of the Hamiltonian, G, is spontaneously broken in the ground state to some subgroup $H \subset G$ (and note that it is possible that the whole symmetry is preserved, so that $H = G$). This "residual" symmetry group H is sufficient to distinguish some phases, as states with different residual symmetries are not in the same phase. However, the distinctions drawn on the basis of residual symmetries are not sufficient to fully classify the phase diagram, as there may be different SPT phases which share the same residual symmetry group H. We shall discuss these now, following the arguments presented in Ref [50].

5.3.2 Onsite Symmetry

The additional parameters required to classify SPT phases with the same residual symmetry H depend upon H itself. In particular, when the elements of H act upon the physical degrees of freedom in a matrix product state, the equivalent action in the virtual ("bond") space forms a representation of H. The different equivalence classes of these representations in virtual space are the basis for the SPT classification schemes [47, 48, 50–53]. To better understand this, we first consider the case of a symmetry group G_{int} which act directly onsite. For clarity of the examples, we shall also assume one-site translation invariance.

Let u_g be a unitary representation of an element $g \in G_{int}$, and let $U(g) = u_1(g) \otimes \cdots \otimes u_N(g)$ be an operator on the entire spin chain acting $u(g)$ at each site. If $U(g)$ is still a symmetry of the ground state $|\psi\rangle$ (and hence part of the residual symmetry group), then $U(g)|\psi\rangle$ must leave the state invariant, except for a possible overall phase factor,

$$U(g)|\psi\rangle = \chi(g)^N|\psi\rangle. \tag{5.7}$$

We have written the overall phase as $\chi(g)^N$ so that we can compare this overall factor to the action of $u(g)$ on an individual site. As stated, we wish to consider $|\psi\rangle$ as an MPS and compare the action of the symmetry on the physical degrees of freedom to the action in the virtual space. Whatever change is introduced in the virtual space must still leave the state able to satisfy Eq. 5.7, Given that our MPS has translational invariance, We consider only a singly type of tensor from the MPS, A^j. The gauge freedoms of the MPS representation imply that in order to preserve symmetry, we must have [49–52]

$$u(g)_{ij}A^j = \chi(g)V^{-1}(g)A^iV(g). \tag{5.8}$$

i.e., the A^j can undergo a similarity transformation by some matrix V, and each site can pick up a phase factor $\chi(g)$.

Because u is a group representation, the usual group properties must hold, and we must have $u(g_1)u(g_2) = u(g_1g_2)$, etc. But enforcing these properties in Eq. 5.8 also constrains $\chi(g)$ and $V(g)$; for example, if we consider acting $u(g_1)$ and $u(g_2)$ on A^j in sequence, a pair of phases and a set of "nested" similarity transformations will appear, giving us

$$u(g_1)_{ik}u(g_2)_{kj}A^j = \chi(g_1)\chi(g_2)V^{-1}(g_1)V^{-1}(g_2)A^iV(g_2)V(g_1). \tag{5.9}$$

But for this to hold at the same time as

$$u(g_1g_2)_{ij}A^j = \chi(g_1g_2)V^{-1}(g_1g_2)A^iV(g_1g_2), \tag{5.10}$$

which is simply Eq. 5.8 for the case $g = g_1g_2$, then clearly we must have $\chi(g_1)\chi(g_2) = \chi(g_1g_2)$. Similar arguments can verify the remaining group properties; in fact $\chi(g)$ itself is a (one-dimensional) representation of G_{int}.

The same can be seen to be true of $V(g)$, with one exception: clearly we could change $V(g)$ by some phase factor $\omega(g)$ without affecting the validity of Eq. 5.8, as the factor will be cancelled by $V(g)^{-1}$. Hence while the $V(g)$'s also form a representation, they form only a *projective* representation, i.e. a representation up to an overall complex phase, so that

$$V(g_1)V(g_2) = \omega(g_1, g_2)V(g_1g_2). \tag{5.11}$$

If we further proceed to enforce associativity on the phases $\omega(g_1, g_2)$, we see that the relative choices of the ω's fall only a handful of possible classes. These classes are precisely those given by the elements of the second cohomology group of G_{int} over $U(1)$ phases, $H^2(G_{int}, U(1))$. In fact, the classes of the phase factors are distinct, even between states which share a symmetry group, so that for each element of $H^2(G_{int}, U(1))$ we have a different SPT phase [50, 190]. The phase corresponding to the identity element corresponds to the case where there are *no* relative phases among the $V(g)$, so that the representation they form is linear rather than projective. This case is called the "trivial phase"; other phases are called "topologically nontrivial."

Since we have translation invariance in addition to onsite symmetry G_{int}, then the different choices of 1D irreducible representations (irreps) $\chi(g)$ that can appear in Eq (5.8) also correspond to different SPT phases. Note that *without* translation invariance, however, these phase factors cannot be considered as labels for different phases; sites could always be grouped together in such a case until the total phase factor for a group of sites is $\chi(g) = 1$. However, for the case of $G = G_{int} \times \mathcal{L}_{\mathcal{T}}$ (where $\mathcal{L}_{\mathcal{T}}$ is the group of lattice translations), then the different possible SPT phases are labelled by two quantities, ω and χ.

5.3.3 Parity

Without Onsite Symmetry

The general term parity refers to both a spatial inversion of the lattice and a possible onsite operation. We shall refer in particular to parity as $\mathcal{P}_w$, which is generated by the inversion $\mathcal{I}$, as well as the onsite unitary operation w at each site. Note the restriction that, since $\mathcal{P}^2 = e$, w must be some representation of $\mathbb{Z}_2$.

The action of $\mathcal{I}$ deserves slightly more specificity; it is defined as

$$\mathcal{I}:\sum_{i_1\dots i_N} c_{i_1\dots i_N}|i_1\dots i_N\rangle \to \sum_{i_1\dots i_N} c_{i_1\dots i_N}|i_N i_{N-1}\dots i_1\rangle$$
$$= \sum_{i_1\dots i_N} c_{i_N\dots i_1}|i_1\dots i_N\rangle. \quad (5.12)$$

In other words, the ordering of the sites is reversed, or equivalently, the labels on the expansion coefficients are exchanged, so that

$$\mathcal{I}: c_{i_1\dots i_N} \to c_{i_N\dots i_1}, \quad (5.13)$$

.

In order to produce the transformation in Eq. 5.13, $\mathcal{I}$ must act on the MPS tensors as

$$\mathcal{I}: Tr[A^{i_1}_{[1]}A^{i_2}_{[2]}\dots A^{i_N}_{[N]}] \to Tr[A^{i_N}_{[1]}A^{i_{N-1}}_{[2]}\dots A^{i_1}_{[N]}] \quad (5.14)$$

Where we have once again made explicit use of bracketed subscript labels to distinguish the tensors. In this case, the label can be interpreted as indicating the site with which the tensor was *originally* associated, before the application of inversion. The subscripts on the physical indices continue as always to represent the site with which the tensor is *currently* associated.

Equivalently, since we can insert an overall transpose inside the argument of the trace, we can view this as

$$\mathcal{I}: Tr[A^{i_1}_{[1]}A^{i_2}_{[2]}\dots A^{i_N}_{[N]}] \to Tr[(A^{i_1}_{[N]})^T(A^{i_2}_{[N-1]})^T\dots(A^{i_N}_{[1]})^T]. \quad (5.15)$$

In terms of the individual tensors at some particular site M, inversion gives

$$\mathcal{I}: A^i_{[M]} \to (A^i_{[N-M+1]})^T, \quad (5.16)$$

and the full action of parity is

$$\mathcal{P}: A^i_{[M]} \to w_{ij}(A^j_{[N-M+1]})^T \quad (5.17)$$

As we have done in the case of pure onsite symmetry, we will seek to reveal the nature of the possible SPT phases by comparing the effect of this symmetry on the state itself and in the virtual space of the MPS. If $G_{\mathcal{P}}$ is both a symmetry of the Hamiltonian and a part of the ground state residual symmetry group, then in terms of the wavefunction itself we must once again have invariance up to an overall phase, which in this case we term $\alpha(P)$.

$$\mathcal{P}|\psi\rangle = \alpha(P)^N|\psi\rangle. \tag{5.18}$$

As before, the action in virtual space must take the form of similarity transformation and a complex phase, i.e.

$$w_{ij}(A^j)^T = \alpha(P)N^{-1}A^iN. \tag{5.19}$$

But recall again that by definition, $\mathcal{P}$ squares to the identity. It follows from considering two subsequent application of Eq. 5.19 that we must have $\alpha(P) = \pm 1$. Like $\chi(g)$ in the onsite symmetry case, these one dimensional representations label distinct phases; in this case, they correspond to the cases of even and odd parity [50].

Similarly, enforcing that $\mathcal{P}$ squares to the identity also requires that $N^{-1}N^T = \beta(P)\mathbb{1}$, with $\beta(P)$ an ambiguous phase factor which can take on the values ± 1 (in this manner, $\beta(P)$ is somewhat analogous to the relative phase factors ω which arose between the $V(g)$ in the onsite case). Collectively, the two labels $\{\alpha(P), \beta(P)\}$ label the 4 distinct SPT phases protected by $G_{\mathcal{P}}$ [50].

5.3.4 Parity and Onsite Symmetry

Finally, we consider the case where the residual symmetry group contains *both* an onsite symmetry group G_{int} (see Sec. 5.3.2) as well as parity. In such a case, we will clearly need all four of the previously identified SPT parameters ($\chi, \omega, \alpha(P)$ and $\beta(P)$) to distinguish possible phases. But in fact, one additional parameter may arise in certain cases. The actions of parity and an onsite symmetry will often commute with one another, i.e.

$$U(g)\mathcal{P}|\psi\rangle = \mathcal{P}U(g)|\psi\rangle, \tag{5.20}$$

But when this happens, we can consider the actions of the combined symmetries in terms of their effect in the virtual MPS space. To do this, we first, insert Eq. 5.8 into Eq. 5.19, and then conversely Eq. 5.19 into Eq. 5.8, and compare the results, which should be equal since we are presuming that the actions commute. By requiring this, one finds the following constraint between the matrics $V(g)$ and N:

$$N^{-1}V(g)N = \gamma_P(g)V^*(g). \tag{5.21}$$

In this case, it turns out once again that the phase factor $\gamma_P(g)$ is a one-dimensional irrep of G_{int} [50].

Note that there is a certain redundancy in this parameter $\gamma_P(g)$. We can rephase $V(g) \mapsto \theta(g)V(g)$ (so long as $\theta(g)$ is also a representation of G_{int}) with the result that $\gamma_P(g) \mapsto \gamma_P(g)/\theta^2(g)$ in Eq. 5.21. But this rephasing cannot have changed anything at the physical level, and therefore, the 1D irreps γ_P and γ_P/θ^2 are equivalent labels (i.e., they label the same phase of matter) for all the 1D irreps θ of G_{int}. Hence, to determine the relevance of γ_P for a given G_{int}, we must ask whether the squares of the 1D irreps of G_{int} are distinguishible from the 1D irreps themselves. When they are not (as is the case for several groups relevant to this work, such as $\mathbb{Z}_3, and A_4$) then there is effectively only one distinct value for γ_P, and it is not needed in order to characterize the SPT phase.

In sum, the different SPT phases of matter protected by $G = G_{int} \times G_{\mathcal{P}}$ are labeled by the set of parameters $\{\omega, \chi(g), \alpha(P), \beta(P), \gamma_P(g)\}$ [50]. Because our Hamiltonian is not invariant under time reversal, we do not review the classification of SPT phases protected by time-reversal invariance and combinations with other symmetries here; see instead Refs. [50, 52]. We emphasize however that the numerical characterization techniques that we we will subsequently present can be easily extended to the time-reversal case.

5.4 Using the Parameters to Understand the Phases of the A_4 Hamiltonian

5.4.1 Details of the Phase Diagram

Armed with the family of parameters described in the last section, $\{\omega, \chi, \alpha, \beta, \gamma_P\}$, we now describe in detail the different phases of the Hamiltonian of Eq (5.4) seen in Fig. 5.1. The internal symmetry is A_4 which is a group of order 12 and can be enumerated by two generators with the presentation

$$\langle a, x | a^3 = x^2 = (ax)^3 = e \rangle. \tag{5.22}$$

The 3D representation of these generators are

$$a = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \; x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \tag{5.23}$$

This can also be visualized as the rotational symmetry group of the tetrahedron.

First we briefly outline the steps followed:

1. For every point in parameter space $\mu = [-3, 4]$, $\lambda = [-2, 2]$ of the Hamiltonian of Eq (5.4), we use the iTEBD algorithm [28] to compute the ground state.

2. We determine the residual symmetry $H \subset G$ of the full symmetry group $G = A_4 \times G_{\mathcal{P}}$ that leaves the ground state invariant. This includes checking the level of translation invariance, which may be broken down from one-site to two-site or beyond.

3. We determine the labels (subset of $\{\omega, \chi(g)\alpha(P), \beta(P), \gamma_P(g)\}$) that characterizes the fractionalization of residual symmetry and measure their values using the appropriate MPS order parameters.

Several of these steps involve important numerical considerations. Full details of our implementation of these steps can be found in Sec 5.5.

We find that there are eight different phases in total. These phases, labeled "**A**" through "**H**" as indicated to match the phase diagram in Fig. 5.1, are characterized as follows:

1. Phase **A:** Parity and one-site translation only i.e. $H = G_{\mathcal{P}}$ (all internal symmetries are broken). This region is therefore classified by the values of $\{\alpha(P), \beta(P)\}$ and is found to have values

   - $\{\alpha(P) = -1, \beta(P) = -1\}$

2. Phases **B, C, and D:** No unbroken symmetries. The ground state in these three regions are invariant under the full symmetry group $G = A_4 \times G_{\mathcal{P}}$. The relevant labels are $\{\omega, \chi(g), \alpha(P), \beta(P)\}$ (Since all three 1D irreps of A_4 are equivalent under the relation $\gamma_P \sim \gamma_P/\chi^2$, $\gamma_P(g)$ is a trivial parameter). The MPS matrices in all three regions transform projectively i.e. these are non-trivial SPT phases with $\omega = -1$ where $H^2(A_4, U(1)) \cong \mathbb{Z}_2 \cong \{1, -1\}$. Also, $\alpha(P) = -1, \beta(P) = -1$ for all three phases. However, they can be distinguished by the values of χ, i.e. observing that the 1D irrep produced under the A_4 symmetry transformation (Eq (5.7)) in the three regions corresponds to the three different 1D irreps of A_4 . The values of the set of parameters which characterize the regions are as follows.

   - Phase **B**: $\{\omega = -1, \chi : \{a = e^{\frac{i2\pi}{3}}, x = 1\}, \alpha = -1, \beta = -1\}$
   - Phase **C**: $\{\omega = -1, \chi : \{a = 1, x = 1\}, \alpha = -1, \beta = -1\}$
   - Phase **D**: $\{\omega = -1, \chi : \{a = e^{-\frac{i2\pi}{3}}, x = 1\}, \alpha = -1, \beta = -1\}$

3. Phase **E:** Parity, $\mathbb{Z}_2$ and two-site translation. This region possess a hybrid parity $G_{\mathcal{P}}$, generated not by inversion alone but rather the combination of inversion and the order 2 element axa^2 of A_4. Additionally, there is an unbroken onsite $\mathbb{Z}_2$ actions with elements $\{e, x\}$. The relevant labels are $\{\chi(g), \alpha(P), \beta(P), \gamma_P(g)\}$ with values

   - $\{\chi : \{e = 1, x = 1\}, \alpha = 1, \beta = 1, \gamma_P = \{e = 1, x = 1\}\}$

4. Phases **F and G:** These regions possess the same parity and onsite A_4 symmetry as phases **B**, **C**, and **D**, but have translation invariance which is broken down to the two-site level. They are also distinct from the above phases because the MPS matrices transform under a linear representaion of A_4, and have a trivial representation of parity at the two site level. The relevant labels are parameters are $\{\omega, \chi(g), \alpha(P), \beta(P)\}$ with values

   - Phase **F**: $\{\omega = +1, \chi : \{a = e^{-\frac{i2\pi}{3}}, x = 1\}, \alpha = +1, \beta = +1\}$
   - Phase **G**: $\{\omega = +1, \chi : \{a = e^{+\frac{i2\pi}{3}}, x = 1\}, \alpha = +1, \beta = +1\}$

5. Phase **H:** In this final region, the onsite symmetry is broken down to a $\mathbb{Z}_2 \times \mathbb{Z}_2$ subgroup with elements $\{e, x, a^2xa, axa^2\}$. Parity and translation symmetry are both fully retained. It is therefore the only region in our sample phase diagram which requires all five labels $\{\omega, \chi, \alpha, \beta, \gamma_P\}$ to characterize. The values here are

   - $\{\omega = +1, \chi = \{1, -1, 1, -1\}, \alpha = +1, \beta = +1, \gamma_P = \{1, 1, 1, 1\}\}$

Note here that for compactness, the set of values given χ and γ refer to the four elements $\{e, x, a^2xa, axa^2\}$, respectively.

The diversity of phases seen in this phase diagram show the importance of carefully checking for both conventional symmetry-breaking phases and SPT phases. The phases present here also underscore the importance of considering the different possible instances of parity and translation invariance which can occur, since in addition to traditional one-site translation invariance and inversion, one might find e.g. translation breaking without inversion breaking (phases **F** and **G**), or inversion which only exists when hybridized with an onsite symmetry (phase **E**). Such phases go well beyond the 26 phases which could be identified solely by the 26 symmetry-breaking subgroups of the global symmetry group $A_4 \times \mathcal{P}$. Indeed, even restricting ourselves to cases which preserve translation invariance, there are hundreds of possible phases for this model which can be enumerated under the classification scheme reviewed in Sec. 5.3. A detailed review of the forms these phases could take is provided by our co-author in Ref. [182]. the subsequent section, we show the wide variety of phases which could potentially exist given the symmetries of this parent Hamiltonian.

5.5 Numerical Methods for Obtaining the Phase Diagram

For gapped 1D spin chains, the authors of Ref. [46, 47, 52, 191] describe ways of numerically determining the SPT parameter described above, and distinguishing different SPT orders. We build on the technique developed in Ref. [52] where the authors obtain the SPT labels using the representations of symmetry at the virtual level. The numerical characterization of the phase diagram of a general parametrized Hamiltonian $H(\lambda, \mu, \ldots)$ proceeds according to the following steps:

1. Identify the group of symmetries of the Hamiltonian, G of the Hamiltonian.

2. For each point in parameter space $\{\lambda, \mu, \ldots\}$, obtain the ground state $|\psi(\lambda, \mu, \ldots)\rangle$ of the Hamiltonian $H(\lambda, \mu, \ldots)$ numerically as a MPS.

3. For each point in parameter space $\{\lambda, \mu, \ldots\}$, identify the subgroup of symmetries $H \subset G$ that leaves the ground state $|\psi(\lambda, \mu, \ldots)\rangle$ invariant. In our case, this means checking each of the 24 elements of $G = A_4 \times \mathcal{P}$. We also must explicitly check the translation invariance.

4. Obtain the relevant virtual representations for the elements of H, i.e. $\chi, V, \alpha(P)$, and N.

5. From the representations and their commutation relations, obtain all other labels that completely characterize the phase.

In general, this requires calculating the full family $\{\chi, \omega, \alpha(P), \beta(P), \gamma(P)\}$ for each point in parameter space. However, in some cases, the elements of H are such that not all such parameters are necessary or even well-defined. For example, if the subgroup H does not contain the parity operator, then $\alpha(P), \beta(P)$ and $\gamma(P)$ do not exist. Similarly, if $H = \mathbb{Z}_3$ or $\mathbb{Z}_2$, there is only one possible value of ω, and hence we do not need it to distinguish the phase.

5.5.1 Ground State Preparation

Having constructed our Hamiltonian with an explicit symmetry group $G = A_4 \times \mathcal{P}$, the next step is to obtain the ground states. For this, we use the numerical "iTEBD" algorithm [27, 28, 83] to compute the ground states over a range of parameters, $\lambda \in [-2, 2]$ and $\mu \in [-3, 4]$ (this range is simply chosen based on our results to include a large but not necessarily comprehensive sample of different SPT phases). The algorithm computes the ground state of a Hamiltonian H through the imaginary time evolution of an arbitrary initial state $|\psi\rangle$, since $|\psi\rangle$ can be expanded in the energy eigenbasis of Hamiltonian as $|\psi\rangle = \sum_i c_i|E_i\rangle$ and hence $e^{-\tau H}|\psi\rangle$ will suppress all such components except for the ground state $|E_0\rangle$ in the large-τ limit. Except where otherwise noted, data in this chapter were prepared with a random initial state represented as an MPS with bond dimension $\chi = 24$, and evolved according to a fixed sequence of timesteps which were chosen to be sufficient to converge the energy to the level of 10^{-8} at the most numerically "difficult" states. Within each phase, a random set of points have also been recomputed using states with a series of larger bond dimensions ($\chi = 36, 42$, and 60) and a longer sequence of imaginary timesteps, in order to verify that the observed characteristics are not likely to be artifacts of the numerical parameters.

Recalling the details of the iTEBD algorithm from Sec. 3.2.2, there is one salient point which must be remarked upon. For a Hamiltonian H with two-body interactions, the algorithm relies on a decomposition of the Hamiltonian into two sets of terms, those acting first on an even site (H_A) and those acting first on an odd site (H_B), so that $H = H_A + H_B$. As such the imaginary time evolution operator can be approximated by the Suzuki-Trotter decomposition [118, 123], which, to second order, gives

$$e^{-\tau H} \approx (e^{-\delta\tau H_A/2} e^{-\delta\tau H_B} e^{-\delta\tau H_A/2})^N, \tag{5.24}$$

with $\delta\tau = \tau/N$. The total operator can then be applied as a sequence of smaller operators, acting either on an even site first, or an odd site first. This distinction, then, requires the state to be represented with at least two tensors, A_A^j and A_B^j, even if the the resulting state is expected to possess a one-site translation invariance (which would generally allow it to be represented by only a single tensor A^j. This fact will have relevance in later sections, when the translational invariance of the MPS is explicitly discussed).

We will also make explicitly use of the canonical form conditions for our MPS; namely, that the dominant eigenvector of the state's transfer matrix is a vectorization of the identity matrix, i.e.

$$(T)^{(\alpha\alpha')}_{(\beta\beta')}\delta^{(\beta\beta')} = \delta^{(\alpha\alpha')} \tag{5.25}$$

A general iMPS computed via iTEBD will not necessarily be in exactly canonical form. However, because this form is ultimately so useful in the process of extracting SPT parameters below, it is worthwhile to enforce it for the ground state representations at the time of their calculation. As described in Sec. 3.2.2, we have generally achieved this here by means of identity-gate iTEBD evolution at the end of any ground-state calculation.

5.5.2 Symmetry Detection and Extraction of Order Parameters

States with One-Site Translation Invariant Representations

The general numerical scheme for extracting the topological order parameters from a numerical MPS was presented in [52], where it was principally used to study the order parameters $\omega, \beta(P)$, and $\beta(T)$, a parameter for time-reversal symmetry. Here we emphasize that it can be used to extract other parameters like the 1D representation χ as well. We consider the situation first for onsite symmetries and assume that the infinite state possesses one-site translation invariance and is represented by a single tensors A^j. The generalization to other symmetries and to different levels of translation invariance will be considered subsequently.

To check for symmetry and ultimately access the topological parameters, we recall the notion of a "generalized" transfer matrix,

$$T_u \equiv A^j u_{j,j'} (A^*)^{j'}, \tag{5.26}$$

which extends the definition of the ordinary transfer matrix (Eq. 2.15) to include the action of some onsite operator u between the physical indices. In the same manner that the original transfer matrix T represents the contribution of one site to the overlap $\langle\psi|\psi\rangle$, in this case the generalized transfer matrix T_u represents the contribution of one site to the expectation value $\langle\psi|U|\psi\rangle$, where $U = \bigotimes_j u_j$ represents the application of u to every site on the chain (see Fig. 5.4). And similarly, just as an iMPS is not normalized unless T has largest eigenvalue 1, so too is such state only symmetric under U if T_u has largest eigenvalue with unit modulus.

To study the SPT classification of a state, we thus begin by determining the symmetry. To check if the state is symmetric under the application of U, then we first construct T_u and compute the dominant eigenvector X and the associated eigenvalue λ_1. Note that, when the dimensions of T_u is large, it

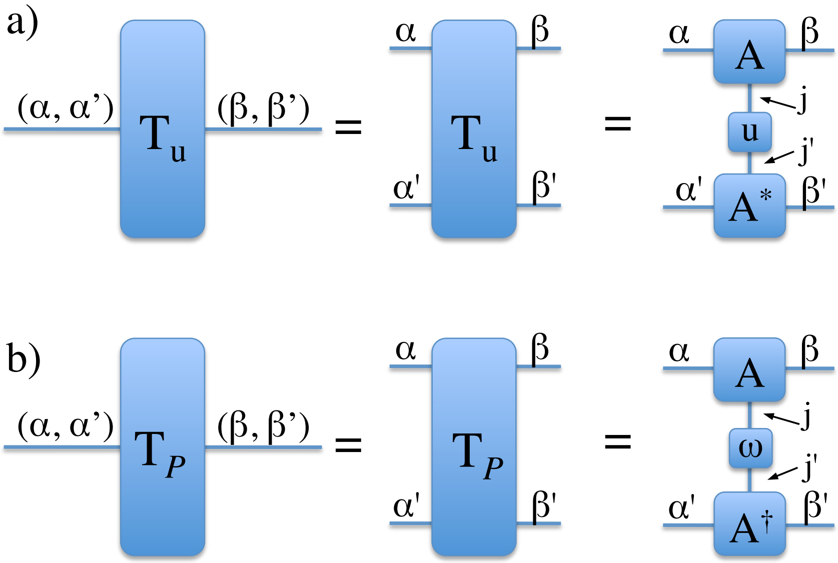

Figure 5.4: (Color Online). The notion of the transfer matrix can be generalized to include (a) onsite operation $U = \bigotimes_j u_j$, or (b) a parity operation $\mathcal{P}_\omega$. Generalizations to other symmetries are possible, but outside the scope of this work as they are not present in our model.

is numerically far easier to use some iterative procedure such as a power or Lanczos algorithm [192–194] to extract this, since only the largest eigenvalue is required and not the entire spectrum. If $|\lambda_1| < 1$, the state is not symmetric under U because $\langle\psi|U|\psi\rangle = \lim_{N\to\infty} T(u)^N$ will vanish. If, however, the unique largest eigenvalue gives $|\lambda_1| = 1$, then we can proceed with the analysis.

Consider now a normalized iMPS in canonical form, which is invariant under a set of symmetries $u(g)$ at each site for g in some symmetry group $H \in G$. Per Eq. 5.8 above, this invariance implies the existence of a set of matrices $V(g)$, which are generally projective representations, and $\chi(g)$, a one-dimensional representation. As shown in [52], one can extract both the projective and a 1-dimensional representation parameters directly from the dominant eigenvector and eigenvalue of the generalized transfer matrix. In particular, if X is the dominant eigenvector (or more precisely, if $X^{\beta}_{\beta'}$ is a matrix and it's vectorization $X_{(\beta\beta')}$ is the dominant eigenvector), then $V = X^{-1}$. The one-dimensional rep $\chi(g)$ is simply equal to the dominant eigenvalue itself. In other words,

$$(T_u)^{(\alpha\alpha')}_{(\beta\beta')}(V^{-1})^{(\beta\beta')} = \chi\cdot(V^{-1})^{(\alpha\alpha')} \tag{5.27}$$

To see this, consider the left hand side of the equation (In many ways, this line of argument is clarified when represented by graphical notation; see also Fig. 5.5.2). Combining the definition of the generalized transfer matrix, Eq. 5.26, with the symmetry fractionalization condition in Eq. 5.8, we have

$$\begin{aligned}(T_u)^{(\alpha\alpha')}_{(\beta\beta')} &= \chi\cdot(V^{-1})^{\alpha\rho}A^{j}_{\rho\sigma}V^{\sigma\beta}(A^{\dagger})^{j}_{\alpha'\beta'}\\ &= \chi\cdot(V^{-1})^{\alpha\rho}T^{(\rho\alpha')}_{(\sigma\beta')}V^{\sigma\beta}.\end{aligned}$$

When this is inserted in the left hand side of Eq. 5.27, the resulting cancellation of V and V^{-1} gives us

$$(T_u)^{(\alpha\alpha')}_{(\beta\beta')}(V^{-1})_{(\beta\beta')} = \chi\cdot(V^{-1})^{\alpha\rho}T^{(\rho\alpha')}_{(\sigma\beta')}\delta^{\sigma\beta'}. \tag{5.28}$$

Then, relabeling the dummy indices ρ and σ into α and β, we can appeal to the canonical form condition of Eq. 2.13 to see that

$$(T_u)^{(\alpha\alpha')}_{(\beta\beta')}(V^{-1})^{(\beta\beta')} = \chi\cdot(V^{-1})^{\alpha\alpha'}, \tag{5.29}$$

which proves that V^{-1} (vectorized) is an eigenvector with eigenvalue χ. Furthermore, because the state is normalized and because we required as a condition for symmetry that$|\chi| = 1$, this proves that V^{-1} is the dominant eigenvector, up to an overall phase factor in V. Hence, any procedure to numerically

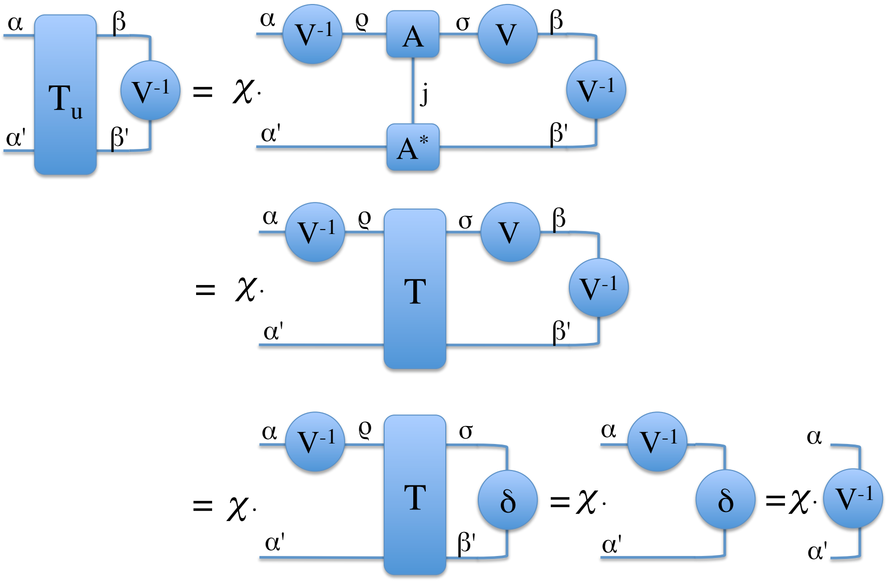

Figure 5.5: (Color online).The projective representation V of a symmetry can be obtained from a state's generalized transfer matrix because the dominant eigenvector of said matrix will be the vectorization of, V^{-1}, so long as the original state is in canonical form. This relation is demonstrated graphically for the case of an onsite symmetry, but easily generalizes to the parity case.

extract the dominant eigenvector and largest eigenvalue from the generalized transfer matrix is sufficient to extract both the 1D representation χ and the projective representation required to compute the projective parameters ω as defined above.

In the foregoing, we have considered only onsite symmetries applied globally to every site on the state[52]. To include other types of symmetries, one simply generalizes further the notion of the already-generalized transfer matrix. For example, the parity symmetry can be studied by means of the matrix

$$T_{\mathcal{P}} \equiv A^j w_{j,j'} (A^\dagger)^{j'}. \tag{5.30}$$

In comparison to Eq. 5.26, we have simply inserted the action of the inversion operator $\mathcal{I}$ by performing a transpose on the virtual indices of the second MPS tensor (seen graphically in panel (b) of Fig. 5.4). In this way, the resulting generalized transfer matrix still represents a one-site portion of the overlap $\langle\psi|\mathcal{P}|\psi\rangle$. By the same arguments as above, and by analogy between Eq. 5.8 and Eq. 5.19, one can see that the quantities $\alpha(P)$ and N can be extracted from the dominant eigenvector and eigenvalue as before, with the latter used to compute the parity parameter $\beta(P)$ as described above.

States Without One-Site Translationally Invariant Representations

Thus far, we have also assumed a state with one-site translation invariance. However, even when the ground state being studied *does* possess a one-site translational symmetry, the tensors in the MPS *representation* of this state may not, because the gauge freedom of an MPS is not itself constrained to be translationally invariant. For example, consider a set of translationally-invariant tensors $\{A^1, A^2, A^3 \dots\}$ and the gauge transformation

$$A^j \to \begin{cases} XA^jZ^{-1}, & j \text{ even} \\ \\ ZA^jX^{-1} & j \text{ odd} \end{cases} \tag{5.31}$$

for any appropriately-dimensioned matrices X and Z. Such a gauge transformation results in an MPS representation of the state whose tensors at even and odd sites may look dramatically different. But both sets of tensors (before and after the transformation) collectively represent the same, translationally invariant state. Cases like this are of particular interest here because, as noted above, the iTEBD method (like other MPS ground-state preparation algorithms) necessarily results in an MPS representation with different tensors at even and odd sites, regardless of the translational symmetry of the physical state.

This feature does not affect our numerical calculation of the SPT order parameters ω and $\beta(P)$, which are obtained as eigenvectors of the generalized transfer matrices, but has important significance for the one-dimensional parameters $\chi, \alpha(P)$, and $\gamma(P)$. Consider, for example, a state which is represented by k sets of tensors $\{A^{j_1} ... A^{j_k}\}$, either because the underlying state has only a k-site symmetry, or perhaps simply because our particular numerical representation requires it. The symmetry condition of Eq. 5.8 must still hold on a k-site level; that is, we will have

$$u(g)^I_J A^J = \chi(g)^k V^{-1}(g) A^J V(g), \tag{5.32}$$

where $A^J = A^{j_1}A^{j_2}\cdots A^{j_k}$ is now a tensor representing the entire block of spins which are the unit cell of the translation invariance, and the composite indices I and J are equal to $(i_1 i_2 \cdots i_k)$ and $(j_1 j_2 \cdots j_k)$. Clearly if we now define a k-site generalized transfer matrix,

$$T_u^{(k)} \equiv A^J u_{I,J'} (A^*)^{J'} \tag{5.33}$$

then the arguments from the preceding section show that V^{-1} can still be found as the dominant eigenvector of $T_u^{(k)}$.

The largest eigenvalue, on the other hand, is now equal not to χ, but to χ^k. In the typical case of an iTEBD state, where $k = 2$, this is problematic because for many common symmetry groups, the values of $\chi(g)$ will be ± 1, so a numerical calculation which gives only χ^2 will be unable to distinguish between the different phases. More generally of course, a k-site representation will always leave us initially unable to distinguish the cases where χ is a k^{th} root of unity.

Of course, if the underlying state has a one-site translation invariance (despite being represented by tensors with only a two-site invariance), one expects that by use of some suitable gauge transformations it should be possible to transform the representation itself back into a translationally-invariant form. Here, we show how this can be done in practice. Suppose we have a translationally-invariant state with, say, a two-site representation $\{A^j, B^{j+1}\}$ and an even number of total spins, such that the state in question is given by either

$$|\psi\rangle = \sum_{j_1\cdots} Tr[A^{s_1}B^{s_2}A^{s_3}\ldots B^{s_N}]|s_1 s_2 \ldots s_N\rangle \tag{5.34}$$

or

$$|\psi\rangle = \sum_{j_1\cdots} Tr[B^{s_1}A^{s_2}B^{s_3}\ldots A^{s_N}]|s_1 s_2 \ldots s_N\rangle \tag{5.35}$$

To recover a one-site representation, we first construct a new tensor of the form:

$$\tilde{A}^{s_j} = \begin{pmatrix} \mathbf{0} & B^{s_j} \\ A^{s_j} & \mathbf{0} \end{pmatrix}. \tag{5.36}$$

This new tensor in fact describes the same wavefunction $|\psi\rangle$. This can be seen by considering the product:

$$\prod_j \tilde{A}^{s_j} = \begin{pmatrix} A^{s_1}B^{s_2}A^{s_3}B^{s_4}\cdots & \mathbf{0} \\ \mathbf{0} & B^{s_1}A^{s_2}B^{s_3}A^{s_4}\cdots \end{pmatrix}. \tag{5.37}$$

If we take the $\tilde{A}^{s_j}$ to be the tensor specifying a new MPS and compute the coefficients, we will have

$$\begin{aligned}|\tilde{\psi}\rangle &= \sum_{s_1\cdots} Tr[\tilde{A}^{s_1}\dots\tilde{A}^{s_N}]|s_1\dots s_N\rangle \\ &= \sum_{s_1\cdots} Tr\left(\prod_j \tilde{A}_{[j]}\right)|s_1\dots s_N\rangle\end{aligned} \tag{5.38}$$

and thus, upon substituting Eq. 5.37, we find

$$\begin{aligned}|\tilde{\psi}\rangle &= \sum_{s_1\cdots} Tr[A^{s_1}B^{s_2}\cdots]|s_1\dots s_N\rangle \\ &\qquad + \sum_{s_1\cdots} Tr[B^{s_1}A^{s_2}\cdots]|s_1\dots s_N\rangle \\ &= 2|\psi\rangle\end{aligned} \tag{5.39}$$

In other words, the state described by the tensor $\tilde{A}^{s_j}$ is essentially identical to the state specified by the original tensors $\{A^{s_j}, B^{s_{j+1}}\}$. The only difference is that the correct product of tensors needed to give the coefficients of the state in Eq. 5.34 will always appear twice, differing only by an irrelevant one-site translation (because the underlying state has a one-site translation invariance to begin with, these two copies of the state are still equivalent).

Because the new tensor $\tilde{A}^{s_j}$ now contains two degenerate descriptions of the same state, it can be placed in a block diagonal form by appealing to the procedure given in Ref. [15] for block-diagonalizing an MPS representation (see also Appendix C in [195]). The resulting blocks will each independently represent the state, but with one-site translation invariance.

The procedure, briefly outlined, is as follows: first, one must ensure that the tensor $\tilde{A}^{s_j}$ is itself in the canonical form, in the sense that it satisfies Eq. 2.13. To do this, construct the transfer matrix for $\tilde{A}^{s_j}$ and compute the dominant eigenvector. This may result in a degenerate manifold of eigenvectors, but by properties of the transfer matrix, at least one of these will be the vectorization of some positive matrix X [196]. Since this X is invertible, we can then take $\tilde{A}^{s_j} \to X^{-1/2}\tilde{A}^{s_j}X^{1/2}$. By construction this new definition of $\tilde{A}^{s_j}$ will satisfy the canonical form.

From this, we once again construct a transfer matrix and compute its dominant eigenvector(s). At least one corresponds to a matrix Z which is not proportional to the identity matrix (up to numerical precision). Furthermore, since the vectorization of $Z^\dagger$ is also an eigenvector of the transfer matrix in canonical form, we can take $Z \to (Z + Z^\dagger)/2$ so that Z is Hermitian (unless $(Z + Z^\dagger)/2$ is itself proportional to the identity, in which case one can always choose instead $Z \to i(Z - Z^\dagger)/2$.) Finally, we compute the largest magnitude eigenvalue z_1 of this new matrix Z, so that we can construct a matrix $W = \mathbb{1} - (1/z_1)Z$ to be a matrix which is manifestly not full rank. Let P be a projector onto the support of W, and $P^\perp$ the projector onto its complement. We can now decompose $\tilde{A}^{s_j}$ around theses spaces, as

$$\tilde{A}^{s_j} = P\tilde{A}^{s_j}P + P^\perp \tilde{A}^{s_j}P^\perp + P\tilde{A}^{s_j}P^\perp + P^\perp \tilde{A}^{s_j}P. \tag{5.40}$$

The reason for the construction of the matrix W from a fixed point now becomes clear, as it has been shown that for such matrix W and its associated projector P, we have $\tilde{A}^{s_j}P = P\tilde{A}^{s_j}P$ [15] (for proof, see Appendix B. Consequently, the final term in Eq. 5.40, which represents one of two off-diagonal blocks in $\tilde{A}^{s_j}$, vanishes identically. This, in turn, ensures that the remaining off-diagonal block cannot mix with either of the diagonal blocks in any product $\tilde{A}^{s_j}\tilde{A^{s_{j+1}}}\cdots$. It therefore does not participate in the calculation of the coefficients of the corresponding states, and can be ignored.

The remaining terms, $P\tilde{A}^{s_j}P$ and $P^\perp \tilde{A}^{s_j}P^\perp$, represent the relevant blocks along the diagonal of the tensor. We remark that in principle, one may need to carry out the above procedure iteratively for each such block ($P\tilde{A}^{s_j}P$ and $P^\perp \tilde{A}^{s_j}P^\perp$) to see if further block reduction is possible. But in practice, for the two-site iTEBD ansatz, a single iteration should suffice. Then, by construction of $\tilde{A}^{s_j}$, each will be an equivalent representation of the same state, and each can represent the state with only a one-site translation invariance. In other words, if we simply treat $P\tilde{A}^{s_j}P$ as the tensor representing the state, we can use all the procedures in the preceding section to directly compute the entire family of SPT parameters.

An alternative method for extracting the one-dimensional parameters when their values are k^{th} roots of unity would be to compute the ground state with a version of the iTEBD algorithm designed to act on an n-site unit cell, where n does not divide k. In this case, the dominant eigenvalue of the generalized transfer matrix will be χ^n, from which χ can now be calculated without ambiguity. Such generalized iTEBD algorithms have been employed successfully (see for example [197]), but may be less numerically stable, and cannot be used for a general state unless one is sure that n is commensurate with the underlying translation invariance of the state. Nevertheless, both methods are possible in practice, and we have used both to cross-check one another in the results presented in this paper.

States with Broken Translation Invariance

Finally, it may also be the case that a state lacks a one-site translationally invariant representation precisely because the ground state is not one-site translationally invariant. When this occurs, one can still compute topological order parameters for onsite symmetries, but only once they and the associated symmetries have been suitably redefined to be consistent with the translational invariance. In other words, if the state has a k-site translation invariance and is represented by the k tensors $\{A^{j_1}A^{j_2}\dots A^{j_k}\}$, one combines the tensors in the same manner contemplated above, forming a new tensor $A^J = A^{j_1}A^{j_2}\cdots A^{j_k}$ with an enlarged physical index which is given by the composite index $J = (j_1 j_2 \dots j_k)$. We then also re-interpret the onsite symmetry operation to be $u^I_J = u^{i_1}_{j_1} \otimes u^{i_2}_{j_2} \otimes \dots u^{i_k}_{j_k}$ under the same convention. Once again, with the tensors merged so they continue to represent an individual "unit cell" of the state, then the relation of Eq. 5.8 will still hold, and we can compute the projective representations of the symmetry from the dominant eigenvalue of the transfer matrix. Unlike the situation described above, however, where the dominant eigenvalue did not give the one-dimensional representation χ (but rather χ^k), in this case the eigenvalue for the merged cell still gives an order parameter. Indeed, there is no longer a physical meaning to the k^{th} root of the eigenvalue, because one-site translation is no longer a symmetry.

For such states, it is also essential to carefully verify the level of any residual translation symmetry. As discussed above, the traditional iTEBD algorithm assumes a two-site invariant representation of the state; hence, if this algorithm produces a state which appears to have translation symmetry which is broken on the one-site level but present at a two-site level, it cannot be assumed that two-site translation is a symmetry of the true ground state; such symmetry may instead have been forced by the algorithm. In this work, whenever one-site translation symmetry is broken, we recompute the ground state using a version of iTEBD with a larger (say, four-site) unit cell. If the two-site translation invariance is still present after such a test, it can then be safely assumed to be a genuine property of the true ground state, and not a property forced by the numerical ansatz. In general, an algorithm with an k-site ansatz cannot by itself confirm translation invariance at the k-site level.

5.5.3 Obtaining the SPT Labels $\{\omega, \beta(P), \gamma(g)\}$

It is clear how the one-dimensional representations χ and $\alpha(P)$ can be used by themselves to label a phase, since each is a single number. Now, however we must discuss how to extract similar numerical labels from the projective representations and other matrices obtained above (e.g. V, N, etc). The authors of [52] show that the formula

$$\omega = \frac{1}{D} Tr \left[V(x_1)V(x_2)V^\dagger(x_1)V^\dagger(x_2) \right] = \pm 1 \tag{5.41}$$

provides a satisfying method to compute a single number $\omega \in H^2(G_{int}, U(1))$ with which to classify the projective representations V for the case in which the residual symmetry group is $H = \mathbb{Z}_2 \times \mathbb{Z}_2$ (with $x_1, x_2 \in \mathbb{Z}_2 \times \mathbb{Z}_2$). In this construction, the combination of $V(x)$ and $V(x)^\dagger$ for each of the elements x_1, X_2 ensures that the result is insensitive to any gauge transformation in the MPS, which could change the projective representations by $V \mapsto XVX^{-1}$ and/or $V \mapsto e^{i\theta}V$. When the virtual representations are trivial, the product $V(x_1)V(x_2)V^\dagger(x_1)V^\dagger(x_2)$ will give the identity matrix, and hence Eq. 5.41 will give $\omega = +1$. For a nontrivial representation, the product picks up an overall minus sign and the result is instead $\omega = -1$. Thus these two cases, consistent with $\omega \in H^2(G_{int}, U(1)) = \mathbb{Z}_2$, clearly distinguish the two types of phase resulting from either linear or projective symmetry representations in virtual space.

We wish to define formulas similar to Eq. 5.41 for the remaining possible subgroups which can appear in our model. Thus in addition to $\mathbb{Z}_2 \times \mathbb{Z}_2$, we must consider $\mathbb{Z}_2$, $\mathbb{Z}_3$, and the full symmetry group A_4. Pleasantly, the first two cases both give rise to trivial second group cohomologies, $H^2(G, U(1)) =$ trivial group. In these cases, there are no projective representations, only linear, and we need not construct a parameter.

This leaves only the case of A_4 to be handled. The formula which we use to define a parameter in this case is

$$\omega = \frac{1}{D} Tr \left[(V(a)V(x)V^\dagger(a)V^\dagger(x))^2 \right] = \pm 1. \tag{5.42}$$

Like Eq. 5.41, this formula was constructed by considering products of elements which will give either the identity matrix $\mathbb{1}$, or $-\mathbb{1}$, depending on whether we have linear or projective representations, respectively. A similar approach could generally be used to derive formulas for the parameters for the case of different symmetry groups in other models.

A somewhat parallel construction, also introduced in Ref. [52], can be used for the inversion parameter $\beta(P)$; namely

$$\beta(P) = \frac{1}{D} Tr \left(N N^* \right). \tag{5.43}$$

Furthermore, from Eq (5.21) we can see that the parameter $\gamma(g)$ which results from the commutation of onsite and parity can be obtained as

$$\gamma(g) = \frac{1}{D} Tr \left[N^{-1} V(g) N V^T(g) \right] \tag{5.44}$$

Here, however, an important technical point arises. Although eq (5.44) has a similar form to the equations used to compute ω and β, it differs in an important respect. Recall that, as calculated above, the matrices V and N are obtained only up to arbitrary overall phase factors. These phases are irrelevant to the calculation of ω and β, as both V and V^* appear equally in the equations which define them. In Eq. 5.44, however, the matrix V^T will fail in general to cancel the phase contributed by V.

Since the $V(g)$ can carry a different phase for each g, we must find a way to self-consistently fix the phase factors of each. In principle, this can always be done by appealing to the properties of projective representations. The extracted matrices V should satisfy a set of relationships

$$V(g_1)V(g_2) = \omega(g_1, g_2) V(g_1 g_2), \tag{5.45}$$

with the phases $\omega(i, j)$ forming the "factor system" of the representation. Since the matrices which we numerically extract by the above procedure do not automatically satisfy this relationship, let us label them $\tilde{V}$, with $\tilde{V}(g) = \theta_g V(g)$ for some phase factor θ_g. From this, one can conclude that the numerical matrices obey a similar relation:

$$\tilde{V}(g_1)\tilde{V}(g_2) = \frac{\theta_{g_1 g_2}}{\theta_{g_1}\theta_{g_2}} \omega(g_1, g_2) \tilde{V}(g_1 g_2). \tag{5.46}$$

By analogy to Eq. 5.45, let us define

$$\tilde{\omega}(g_1, g_2) = \frac{\theta_{g_1 g_2}}{\theta_{g_1}\theta_{g_2}} \omega(g_1, g_2). \tag{5.47}$$

Note that these phases $\tilde{\omega}(g_1, g_2)$ can be computed numerically from

$$(1/D) Tr[\tilde{V}(g_1)\tilde{V}(g_2)\tilde{V}(g_1 g_2)^{-1}]. \tag{5.48}$$

Furthermore, since parity is assumed to be a symmetry of the state in question (if it is not, then the concept of a γ parameter is undefined and the phase factors θ are irrelevant), then we must have $\omega(g_1 g_2)^2 = 1$ [50]. Inverting Eq. 5.47 and applying this condition tells us that

$$\theta_{g_1}^2 \theta_{g_2}^2 \tilde{\omega}(g_1, g_2)^2 = \theta_{g_1 g_2}^2. \tag{5.49}$$

Since the $\tilde{\omega}$ are known, this set of equations, which run over all the group elements g, are sufficient to solve for the phases θ. In fact, when V is unitary, it is clear from the definition of γ in Eq. 5.44 that only θ^2, and not θ itself, is needed to correct for the spurious phase factors, which further simplifies the system of equations which must be solved.

In practice, another convenient way to fix these phase factors is by interpreting the projective representations of the group, $\tilde{V}$ as linear representations of the covering group (or at least, one of the covering groups). For example, in the case of $\mathbb{Z}_2 \times \mathbb{Z}_2$, the quaternion group $\mathbb{Q}_8$ is a covering group. Hence the elements of the projective representation of $\mathbb{Z}_2 \times \mathbb{Z}_2$, $V(g)$ can have their overall phases fixed so that they obey the structure of this group; in particular, for the representation of the identity element we must have $V(e)^2 = \mathbb{1}$, and for all others, $V(g)^2 = -\mathbb{1}$.

5.6 Summary and Future Directions

In this chapter, we have presented a numerical study of a one-dimensional spin-1 Hamiltonian symmetric under one-site translation, lattice inversion, and an on-site A_4 symmetry group. Ground states are generated using the iTEBD algorithm, and then tested to determine the residual symmetry group at each point in parameter space. We also describe in detail how the SPT classification parameters of Ref. [50] can be numerically evaluated. These methods are built upon the approach of Ref. [52], and we extend them here by demonstrating how to extract the one-dimensional representations from the two-site MPS ansatz required by iTEBD, as well as how to remove spurious phase effects in the evaluation of the SPT parameter $\gamma(P)$. Over a range of values of the Hamiltonian's two parameters, we identify eight distinct gapped phases, distinguished by a combination of symmetry breaking and SPT parameters.

Among these phases, we also identify three contiguous, topologically nontrivial phases (regions **B**, **C** and **D**). By studying the phase transition between these regions, we provide numerical evidence that the transitions are second-order in nature with associated CFTs whose central charges are both given by $c \approx 1.35$. This result is consistent with the recent work of Ref. [185], in which the authors similarly examined continuous phase transitions between SPT phases and observed that the central charge of the CFT at the transition should have $c \geq 1$.

We note that the phase transitions study here differ from the specific examples in Ref. [185] in an important respect. Our phases possess a one-site translation invariance in addition to the onsite A_4 symmetry in the ground state. As a result, we can define order parameters from the one-dimensional representations in addition to the projective representations, and in fact, our phases **B**, **C** and **D** are distinguished not by the projective representations of the onsite symmetry (which is nontrivial and equivalent in all three phases), but instead *only* by the one-dimensional representations.

As a further area of study, we wish to compare our results to the work of Ref. [183], which conjectures that there can exist no continuous phase transitions from one non-trivial SPT phase to another when the internal symmetry is discrete at all length scales. Our model appears to contain counterexamples, and we do not believe these are simply the result of some fine-tuning, as the same behavior appears across a finite range of the Hamiltonian parameter μ. However, it remains to be studied whether perhaps the discrete symmetries we consider become somehow enhanced to continuous symmetries at the phase transitions. Further study of the properties of SPT-to-SPT phase transitions is clearly indicated, particularly in the case of models with translation invariance and of nontrivial to nontrivial SPT phase boundaries.

Chapter 6

Many-Body Localization and Entanglement

The standard treatment of a system of particles with a finite temperature [198, 199] is to assume that the system is in contact with an external reservoir, and that, given sufficient time, the system will "thermalize," reaching an equilibrium where its properties are described by only a few numbers, such as particle number and the temperature itself. Indeed, even the notion of a "temperature of the system" is ill-defined except for a system which is at equilibrium. But while this approximate description has played a powerful role in the development of quantum statistical mechanics, the notion of a system completely isolated from the outside environment is also of great importance. Such systems can serve as theoretical models to give us unique insight into quantum dynamics, and are also increasingly being approximated in real-world experiments, where for example strongly isolated optical lattices and other condensed matter systems serve as our models for rudimentary quantum computers.

Logically, one must then consider the behavior of *closed* quantum systems, i.e. those which are not taken to be in contact with an external reservoir. In such systems, it is still quite possible to observe a thermalization process, with the system acting as "its own reservoir" for all subsystems, and reaching a homogenized state after a sufficient period of time. But this behavior, while perhaps common, is not the only possibility. Instead, under the right conditions, which we shall discuss below, a closed system can remain "localized," with subsystems that do not "talk" to one another in the conventional thermal sense, as famously observed by Anderson [200] . Clear perturbative [201] and numerical evidence [54] strongly suggests that this is true even for systems with interactions, where the phenomenon is called "Many-Body Localization" (MBL). The field of MBL has recently attracted substantial interest in the condensed matter community, as the properties of MBL systems have begun to be better understood (see for example Refs. [54–58]) and even tested in practice [202, 203].

As more and more MBL systems are studied, it has become particularly important to develop reliable indicators which can serve as "order parameters" to distinguish between instances of localized and more conventional thermal ("ergodic") behavior. In fact, in systems where localization may appear as a function of some tunable parameter (such as the strength of some disordered field), the transformation from localized to thermal is now regarded as a proper quantum phase transition (though it must be understood as a dynamic transition in the spectrum, not a transition of ground states) [58]. One such possibly MBL indicator is the "concurrence" of the system, an entanglement measure introduced in in the context of spin-1/2 systems in Ref. [63] and studied quite recently for MBL systems in [65]. The quantity has many desirably qualities as an MBL parameter, which we shall discuss below, but it is also strongly limited. While generalizations to larger system types exist [204–206], a convenient closed-form expression for concurrence exists only for pairs of spin-1/2 particles [204, 207] . Naturally, it would be desirable to study localization behavior in systems with higher spin, and to look at entanglement properties in subsystems of arbitrary size. Fortunately, a related entanglement measure, the "Negativity" [64], is not subject to the same restrictions. It is natural to ask whether the negativity can also be used as an indicator of MBL behavior, and this chapter, we provide numerical evidence that the answer is likely "yes."

This chapter is organized as follows: In section 6.1, we present a more detailed explanation of the nature of many-body localization. In section 6.2, we consider some very recent proposals for algorithms which can generate MPS representations of MBL states[59–62] , and present our implementation of the so called "SIMPS" algorithm [59]. Section 6.3 presents our results on the behavior of concurrence and negativity in MBL systems, using a combination of exact diagonalization and data from SIMPS. We provide a summary and discussion of future projects in 6.4.

6.1 Many-Body Localization: an Overview

We first briefly review the current understanding of MBL systems and identify some some relevant properties which have been used to certify the presence or absence of localization. For further information on the subject, we refer readers to the comprehensive review in Ref. [55].

6.1.1 Thermalization in Closed Quantum Systems

Perhaps the best way to understand localized behavior in a system is to consider its opposite. We first discuss the “expected” ergodic behavior of a closed quantum system, with the intent of clarifying by contrast what conditions are *violated* by localization.

Consider a quantum system S which is “closed” (in the sense that we do not assume contact with any outside reservoir) and a particular finite subsystem A, with the extent of S taken to the thermodynamic limit. In particular, A may be imagined as a particular “pocket” of space within S, but this need not be the case. For example, instead of considering this spatial pocket, ¢hange perspective to momentum space. Any finite set of degrees of freedom is equally acceptable. Define also the complementary subsystem $E = S \setminus A$.

We can imagine what this system would look like *if it were* in contact with a reservoir at temperature T. The natural state would be described by a thermal density matrix

$$\rho_{th}(T) = e^{-H/k_b T}/Z \tag{6.1}$$

for Z the standard partition function. It follows that we could also construct the state of our subsystem A (when hypothetically thermalized by a reservoir) by simply tracing over E, i.e.

$$\rho_{th}^{(A)}(T) = tr_E \left(e^{-H/k_b T}/Z\right) \tag{6.2}$$

We emphasize again that Eq. 6.1 is *not* the state we expect to see our closed system in for large time t; we can see that this cannot be the case because our system must have evolved by a unitary, i.e. information-preserving process

$$\rho(t) = e^{-itH} \rho(t = 0) e^{itH}. \tag{6.3}$$

Hence, it cannot simply have arrived in a Boltzmann distribution ρ_{th}wherein all information about the initial state $\rho(t = 0)$ has been lost. So the only interpretation of the density matrix in Eq. 6.1 is the state which the system *would* be in if it were in contact with a reservoir at temperature T.

But while ρ_{th} cannot be the correct description of the full system by unitarity, there is no such restriction preventing Eq. 6.4 from holding for our subsystem. Even if the full, isolated system is not in a thermal distribution, we might expect

$$\rho^{(A)}(t) = tr_E \left(\rho_{th}(T)\right). \tag{6.4}$$

for large values of t. After all, from the perspective of A, the remainder of S, i.e., the subsystem E, appears as an infinite reservoir in the thermodynamic limit.

Thus, we can form a sensible definition of what “thermal equilibrium” can mean in the context of a closed system: if Eq. 6.4 holds for *all* choices of subsystem A, then we can say that as a whole, the closed system thermalizes to temperature T [208, 209]. Note that, as argued above, such a system must still contain all of the information about its initial configuration. This information is simply no longer locally observable, as it has decohered across the entire state [55].

The Eigenstate Thermalization Hypothesis

An important consequence of this description of thermalization in closed systems is the so-called “Eigenstate Thermalization Hypothesis,” (ETH)[208–212] which we can understand as follows. Consider some thermalizing system governed by a Hamiltonian H, and a particular energy eigenstate $|\alpha\rangle$ with $H|\alpha\rangle = E_\alpha|\alpha\rangle$. Though likely not possible in practice, suppose we initialize an isolated system to be a pure state in precisely this eigenstate, i.e.

$$\rho(t=0) = |\alpha\rangle\langle\alpha| \tag{6.5}$$

Now, we allow time to run. But because we are precisely in an eigenstate, the time evolution of Eq. 6.3 is trivial,

$$\begin{aligned}\rho(t) &= e^{-itH}|\alpha\rangle\langle\alpha|e^{itH} = |\alpha\rangle\langle\alpha| \\ &= \rho(t=0).\end{aligned}$$

In other words, systems prepared as eigenstates persist as eigenstates for all t. Therefore, the only way for the system to be thermal in the sense of Eq. 6.4 is for it to have been thermal *all along*! And because we have not used here any properties of the initial state besides its status as an eigenstate, it must be the case that *if* a system is eventually going to thermalize, its eigenstates must *already* be thermalized, and remain so at all times. Initial states which are *not* thermal can only appear as special combinations of eigenstates carefully constructed so that their thermal components destructively interfere; thermalization results because this carefully balanced superposition will eventually decohere under the unitary evolution.

6.1.2 Localization

The alternative to this kind of thermal behavior is MBL (we set aside for now the fine-tuned case of an integrable system, which is localized by its many explicitly-conserved quantities). We now briefly discuss the behavior and properties of such systems. In general, we will be discussing localization arising in systems with strong disorder, though it is important to note that MBL behavior can also arise in quasiperiodic systems where the period of the potential is incommensurate with the underlying lattice (see e.g. Ref. [213, 214]). Except where noted, all comments which follow can apply equally to such cases. Indeed, for any quasiperiodic potential, a related random model can be constructed in which the strength of the potential at site i is randomly drawn from the distribution $[-W_i, W_i]$, with the distribution widths W_i themselves varying quasiperiodically. Such systems should have comparable behavior. For this and other reasons, it is suspected that there are no qualitative differences in the behavior of MBL arising from disorder versus MBL arising from quasiperiodicity [215], although this remains an open question.

In an MBL system, the ETH as stated above is explicitly *false*; states exist (both eigenstates and general linear combinations of them) for which the whole system does *not* act as a reservoir for all of the subsystems. The behavior of such states seems to defy intuition: in a system with interactions, which one might reasonably presume would allow every subsystem to "talk" to every other to the extent that all initial information becomes distributed across the state. But instead, such systems display a kind of "memory" of their initial conditions, with some (though likely not all) local observables yielding the same expectation values out to large timescales.

From this behavior arises the notion of an "l-bit" (for "localized bit") [216, 217]. Consider a system of noninteracting particles; such a system would naturally be described by a set of conserved charges in some basis. For example, in a trivial, non-interacting spin chain $H = \sum_i \sigma_i^z$, eigenstates can all be described by the set of local spins $\{\sigma_i^z\}$, which all commute with the Hamiltonian and hence provide good quantum numbers for a basis of simultaneous eigenstates. But if this behavior exists for isolated spins ("bits") which are *explicitly* not interacting, then the same behavior should hold for localized systems as well, so that a new family of conserved charges (the l-bits) emerges. In particular, it has been shown[216] that these l-bits are simply "dressed" versions of the pre-existing bits: they are given by products (and sums of products) of the original bits at nearby sites. Contributions from sites which are far away are exponentially suppressed, giving rise to a new set of quantum numbers for the system. This l-bit behavior will form the basis for several of the MPS algorithms for finding MBL states discussed in section 6.2. The l-bit construction also helps contextualize the “localized” behavior of fully integrable systems: such systems can be seen as fine-tuned, exact instances where the l-bits emerge by construction.

6.1.3 Indicators of Localization

Since the rapid reemergence of interest in MBL systems within the past decade, a wide variety of quantities have been studied as possible indicators to detect MBL behavior. We will briefly review now a sampling of these indicators, emphasizing that considerably more information can be found about each elsewhere in the MBL literature, and that the list is hardly exhaustive.

- **Behavior of local observables:** If the ETH holds for a state, then one expects local observables to differ very little between eigenstates which are close together in energy [208, 209]. Hence, a simple test for whether the ETH holds for a system (i.e. whether the state is ergodic or MBL) is to evaluate some local observable (such as the operator σ_j^z at a specific, arbitrary site j) for many eigenstates within a very narrow range of energies. In a thermal system, the results should be tightly bunched and nearly independent of energy; in MBL, they will be much more broadly distributed. Note however, that one must be careful with the choice of the energy window to be considered if there is any concern that not all eigenstates may be localized. Further, it is subject to concerns about one’s choice of observable; such observable must be chosen to be suitably “typical” in order to be a valid indicator [218]. See also the interesting conjecture in Ref. [219].

- **Behavior of neighboring eigenstates:** A related prediction from the ETH is that neighboring eigenstates should be locally very similar. In [220] it was proposed that Kullback-Liebler information [221] could be used to measure the distance between such wavefunctions. In particular, different values of the Kullback-Liebler divergence will be expected in the thermal and MBL regions. Conversely, in Ref. [59], it was observed that a small local perturbation to an eigenstate should produce an eigenstate with very similar energy if the system is thermal, but may produce an eigenstate with dramatically different energy in an MBL system. Because the MPS structure makes such local perturbations relatively simple to implement (by perturbing only a single tensor of a state), this heuristic is well-suited to studies using MPS algorithms.

- **Level Statistics:** Another particularly common and powerful way to distinguish thermal and MBL systems is to consider the average ratio of consecutive level spacings[222]. Let Δ_n be the difference between energy levels, $E_n - E_{n-1}$, and let $\tilde{r}_n$ be the ratio Δ_n/Δ_{n+1}. Further define $r_n = \min\{\tilde{r}_n, \tilde{r}_n^{-1}\}$, so that $r_n \leq 1$. The distribution of these ratios across a representative sampling of the spectrum is well known from random matrix theory to be quantitatively different in a localized case (where the distribution is Poisson) versus a thermal case (where the distribution is the Gaussian orthogonal ensemble). In particular, these two distributions have markedly different means, so we expect the average (across states and disorder configurations) to be given by $\langle r_n \rangle = 2\log 2$ in localized systems and $\langle r_n \rangle \approx 0.53$ otherwise [223].

- **Normalized Participation Ratio:** The normalized participation ratio (NPR) is somewhat unique in that it can be computed and even given meaning for a single state in the spectrum (though it is still typically used by averaging over disordered configurations). Consider a general state $|\psi\rangle$ expanded in some configuration basis $|i\rangle$ so that $|\psi\rangle = \sum_i c_i |i\rangle$.The NPR for this state and basis is defined as

$$P(\psi) = \frac{\left(\sum_i |c_i|^2\right)^2}{\sum_i |c_i|^4}. \tag{6.6}$$

Observe that for a normalized vector, this is simply $1/\sum_i p_i^2$, where p_i is the probability for $|\psi\rangle$ to be in the configuration $|i\rangle$. Hence the NPR ranges from $P(\psi) = 1$ when the system is completely localized to a single state, to $P(\psi) = N$ when the state is uniformly distributed across all N possible configurations with probability $p_i = 1/N$ for each. Note that for instances where the volume of the relevant Hilbert space is not fixed, one may wish to normalize the NPR by including this factor before comparing the NPR of different states [213]. It is interesting historically to note that this method of measuring localization appears to have its origins in the chemistry of atomic vibrations, Ref. [224]. A related quantity, the participation entropy [225] has also recently been employed as an indicator of MBL; see for example Refs. [220, 226].

- **Entanglement Scaling Law:** For a thermalizing system, a direct consequence of the ETH is that the entanglement entropy between any choice of subsystem and environment must be equal to the equilibrium thermal entropy of the subsystem, which in turn will scale with the volume of the subsystem [208–212]. Hence we have also a volume scaling law for for the entanglement entropy in such systems. In MBL systems, however, this equality breaks down along with the ETH. In fact, in localized systems almost all eigenstates obey instead an area law with respect to the boundary of the subsystem [227–229]. A scaling study of the entanglement in a typical subsystem can hence also be used as a strong diagnostic of MBL behavior, though because it requires finite-size scaling to the point of entanglement saturation it can be costly to compute. Other related properties of entanglement have also been used as indicators, such as the standard deviation of entanglement entropy across a disordered average [230], or the growth with time of entanglement in an initially unentangled state [231, 232].

When considering any of these indicators, it should be remembered as a note of caution that finite-size effects can strongly impact any determination of a localized-delocalized transition. All existing evidence points to the idea that finite size effects tend to increase localization, meaning that the critical value of the disorder will appear smaller than it may actually be in the thermodynamic limit [233].

6.2 MBL States with Matrix Product Algorithms

Of the properties of localized states described above, the last item in particular has special significance in our context. One-dimensional states whose entanglement obeys an area law are precisely those which can be efficiently represented as an MPS with finite bond dimension [17]. Indeed under a set of quite generic assumptions about the Hamiltonian such as nondegeneracy of energy levels, it has been rigorously shown that all eigenstates (up to possibly mobility edge [234]) of a localizing Hamiltonian can be represented by an MPS whose bond dimension is only polynomial in the system size [229].

Thus, it is natural to ask how to obtain MPS representations of such systems. Several proposals have recently been advanced; we present here our implementation of the "Shift and Invert MPS" ("SIMPS") algorithm of Yu, Pekker, and Clark [59]. Additional algorithms will be briefly compared and discussed below.

6.2.1 SIMPS

Broadly, the goal of the SIMPS algorithm (discussed here only for open-boundary systems) is as follows: assume we have a Hamiltonian represented by an MPO, and we wish to search for an MBL state with energy $\tilde{\lambda}$ close to a target value λ. Construct a shifted operator $O = H - \lambda$, and observe that the state $|\psi_{\tilde{\lambda}}\rangle$ with energy closest to our target can be found as the dominant eigenvector of the inverted Hamiltonian

$$\tilde{H} = (H-\lambda)^{-1} = O^{-1}. \tag{6.7}$$

In principle, it could also be found by seeking the smallest magnitude eigenvalue of $(H-\lambda)^2$, but the construction in Eq. 6.7 has a notable advantage. Suppose that we wish to discriminate between the state $|\psi_{\tilde{\lambda}}\rangle$ with energy λ and a nearby state $|\psi_{\tilde{\lambda}+\epsilon}\rangle$ with energy $\lambda+\epsilon$. As the size of the system we consider increases, the size of this ϵ naturally becomes quite small, and if we take $H \to (H-\lambda)^2$, the new separation ϵ^2 becomes smaller still. But in the SIMPS formulation with $\tilde{H} = (H-\lambda)^{-1}$, the new separation ϵ^{-1} will become quite large, helping us to isolate what might otherwise be a difficult state to resolve.

It remains now to numerically find the state $|\psi_{\tilde{\lambda}}\rangle$. Since this is not the *ground* state of $\tilde{H}$, we cannot simply employ traditional MPS algorithms. Instead, we shall employ the power method for identifying dominant eigenvectors. This method is straightforward and well-known, but we briefly review the relevant attributes here in order to comment on convergence properties of the SIMPS algorithm. In the power method, we take a random initial state $|\phi_0\rangle$ and apply $\tilde{H}$ iteratively, normalizing at each step to preserve stability, so that

$$|\phi_{n+1}\rangle = \frac{\tilde{H}|\phi_n\rangle}{||\tilde{H}|\phi_n\rangle||} \tag{6.8}$$

After enough iterations, all that remains will be the dominant eigenvector which we had targeted, $|\psi_{\tilde{\lambda}}\rangle$. To see this, let $\{|i\rangle\}$ be the eigenbasis of the new Hamiltonian $\tilde{H}$, with the (somewhat unorthodox) convention that the eigenvalues $\tilde{E}_i$ are ordered from largest to smallest, $\{\tilde{E}_1 > \tilde{E}_2 > \tilde{E}_3 \dots\}$. As we have with other MPS algorithms, we assume that in general this random choice will have nonzero overlap with our target state, i.e.

$$|\phi_0\rangle = c_1|\psi_{\tilde{\lambda}}\rangle + \sum_{i=2}^{R-1} c_i|i\rangle \tag{6.9}$$

where R is the rank of $\tilde{H}$ and $\{c_i\}$ are the appropriate expansion coefficients. As we iteratively apply $\tilde{H}$ to $|\phi_n\rangle$ we have

$$|\phi_{n+1}\rangle \propto \tilde{H}^n|\phi_0\rangle = c_1\tilde{E}_1^n\left(|\psi_{\tilde{\lambda}}\rangle + \sum_{i=2}^{R}\frac{c_i}{c_1}\left(\frac{\tilde{E}_i}{\tilde{E}_1}\right)^n|j\rangle\right). \tag{6.10}$$

Clearly the coefficients in the sum on the right can be bounded above in terms of E_2, the first subdominant eigenvalue, since $\left(\frac{\tilde{E}_i}{\tilde{E}_1}\right)^n \leq \left(\frac{\tilde{E}_2}{\tilde{E}_1}\right)^n \ \forall \ i$. Hence in the limit of large n, we have

$$\lim_{n\to\infty} |\phi_n\rangle = |\psi_{\tilde{\lambda}}\rangle \tag{6.11}$$

with the orthogonal terms decaying as $|\tilde{E}_2/\tilde{E}_1|^n$. Hence the convergence of the power method is controlled by the ratio of the two largest eigenvalues in our shifted-and-inverted Hamiltonian, or, to put it more naturally in terms of the eigenvalues $\{E_i\}$ of the original Hamiltonian,

$$\frac{\tilde{E}_2}{\tilde{E}_1} = \frac{|\tilde{E}_1 - \lambda|}{|\tilde{E}_2 - \lambda|}. \tag{6.12}$$

In other words, what matters is the ratio of the distance from lambda to the nearest eigenvalue, and the distance to the next-nearest eigenvalue. Generally speaking, this is good news, as the Hamiltonian's spectrum is likely densely populated near (as we expect for large systems) and a randomly chosen λ should be much closer to one particular state than to others (which will be amplified after inversion). The only problems occur when by very poor fortune our target λ falls almost exactly between true energy levels. These two equidistant eigenvalues will yield near degeneracies in $\tilde{H}$ and an inefficient convergence ratio close to unity. This problem can be handled by searching for states with a range of λ's using a step size which is sufficiently small compared to the expected level spacings in H, so that even if a few choices of λ do not converge, others in the range will converge strongly, and the true set of nearest states can be analyzed by comparing the state reached for different values of λ. Alternatively, the choice of λ can also be actively fine-tuned during the algorithm, introducing a small perturbation in the target energy if the current the convergence rate seems poor. A precise way to do this by fitting the convergence profile to extract the convergence ratio is elaborated upon in [59].

Hence, if we can find a way to efficiently apply $\tilde{H}$ to an MPS representation of our initial state, we will have an algorithm that converges to the desired result. If $\tilde{H}$ admits a compact MPO representation, we can proceed through traditional DMRG methods, but in general this will not be the case; the MPO construction techniques we have outlined previously (see also Refs.[32, 156]) explicitly rely on the sum-of-local-terms structure in an operator. Taking the inverse of H generally destroys this.

We do, however, have the operator $O = \tilde{H}^{-1} = H - \lambda$, which *can* still be efficiently represented as an MPO. With this the authors of Ref. [59] reimagine the optimization problem of traditional DMRG to perform the desired update.

We seek a state $|\phi_n\rangle$ such that (up to normalization), $|\phi_{n+1}\rangle = \tilde{H}|\phi_n\rangle$. In terms of O, this state will satisfy

$$O|\phi_{n+1}\rangle - |\phi_n\rangle = 0 \tag{6.13}$$

so we can build an appropriate cost function with an absolute minimum by taking an absolute square. The desired state $|\phi_{n+1}\rangle$ will be the state which minimizes

$$\mathcal{L} = ||O|\phi_{n+1}\rangle - |\phi_n\rangle|| \tag{6.14}$$

Expanding Eq. 6.14, we have

$$\mathcal{L} = \langle\phi_{n+1}|O^\dagger O|\phi_{n+1}\rangle - \langle\phi_n|O|\phi_{n+1}\rangle - \langle\phi_{n+1}|O^\dagger|\phi_n\rangle + \langle\phi_n|\phi_n\rangle \tag{6.15}$$

As we did in the case of ground-state DMRG, we will seek to solve this optimization problem through an alternating least-square approach, sweeping back and forth and optimizing one tensor of our state at a time in search of global convergence. Recall our notation from Chapter 3 that $|A_{[j]}\rangle$ is a vectorization of the tensor to be updated at site j, and that more generally, $|A_{[1\ldots j]}\rangle_{v_j}$ is the state represented by the tensors $\{A_{[1]}\ldots A_{[j]}\}$, with the subscript v_j indicating an uncontracted virtual index at site j. We will be trying to minimize Eq. 6.15 by considering variation with respect to $|A_{[j]}\rangle$. Let us refer to the tensors representing $|\phi_n\rangle$ as $A^{(n)}$. Keeping only the terms which contribute to the variation, this means solving

$$\frac{\delta}{\delta|A^{(n+1)}_{[j]}\rangle}\left(\langle\phi_{n+1}|O^\dagger O|\phi_{n+1}\rangle - \langle\phi_n|O|\phi_{n+1}\rangle\right) \tag{6.16}$$

The dependence of the terms here on $|A^{(n+1)}_{[j]}\rangle$ is not explicit, but by considering the states here in their MPS representation, it becomes clear. By analogy to the procedure in traditional DMRG, Eq. 6.16 can be written simply as

$$O_{eff}|A_{[j]}\rangle = |B_{eff}\rangle \tag{6.17}$$

where O_{eff} is an effective tensor including the action of $O^\dagger O$ and the remaining state tensors around the current site to be updated,

$$O_{eff} = \langle A^{(n+1)}_{[1\ldots j-1]}|_{v'_{j-1}}\langle A^{(n+1)}_{[j+1\ldots L]}|_{v'_{j-1}} O^\dagger O|A^{(n+1)}_{[1\ldots j-1]}\rangle_{v_{j-1}}|A^{(n+1)}_{[j+1\ldots L]}\rangle_{v_{j+1}} \tag{6.18}$$

and B_{eff} is a similar "effective vector" for the structure on the right hand side of Eq. 6.17 without the tensor $A^{(n+1)}_{[j]}$, i.e.

$$|B_{eff}\rangle = \langle A^{(n+1)}_{[1\ldots j-1]}|_{v'_{j-1}}\langle A^{(n+1)}_{[j+1\ldots L]}|_{v'_{j-1}} O|A^{(n)}_{[1\ldots L]}\rangle. \tag{6.19}$$

As was the case in DMRG, these structures may be much more easily understood in graphical notation, as shown in Fig. 6.1.

Hence to converge to our desired new state, at each site of the chain we are solving a general linear system as shown in Eq. 6.17. This forms the bottleneck of the algorithm, as solving such systems can be costly when the dimensions involved are large. If we use MPS representations of maximal bond dimension χ and study a system with local physical dimension d, the tensor O_{eff} can be as large as $\chi^2 d \times \chi^2 d$. Fortunately, by construction this matrix is also symmetric and positive-semidefinite. It thus admits a Cholesky decomposition [235] which we have used in our implementation to optimize the run-time of this step. In particular, we can always find upper triangular matrices U such that

$$O_{eff} = U^T U \tag{6.20}$$

And hence we can find our desired vector by first solving

$$U^T|\tilde{A_{[j]}}\rangle = |B_{eff}\rangle \tag{6.21}$$

and then solving

$$U|A_{[j]}\rangle = |\tilde{A_{[j]}}\rangle \tag{6.22}$$

Note that we have avoided writing explicitly the solution for $|A_{[j]}\rangle$ in terms of matrices like U^{-1}; the most efficient numerical solution will likely not involve computing a full inverse. Although this procedure requires us to solve two linear systems instead of one, the triangular matrices involved in this equation are already factorized and are much easier for a numerical linear solver to manipulate. Their usage can result in a substantial speedup of this step by approximately a factor of two [235, 236].

Note also that in some numerical packages, the Cholesky decomposition is only supported for fully positive definite (not positive semidefinite) matrices, because otherwise the size of the matrix may change. However, if one carefully handles the resulting rows of zeros, the decomposition is perfectly well-defined in the more general semidefinite case [235], and one can always implement by hand a numerically stable alternative (see Appendix C).

Once we have obtained the new tensor in the form of $|A_{[j]}\rangle$, we update the state, move to the next site, and repeat. As in the case of traditional DMRG,

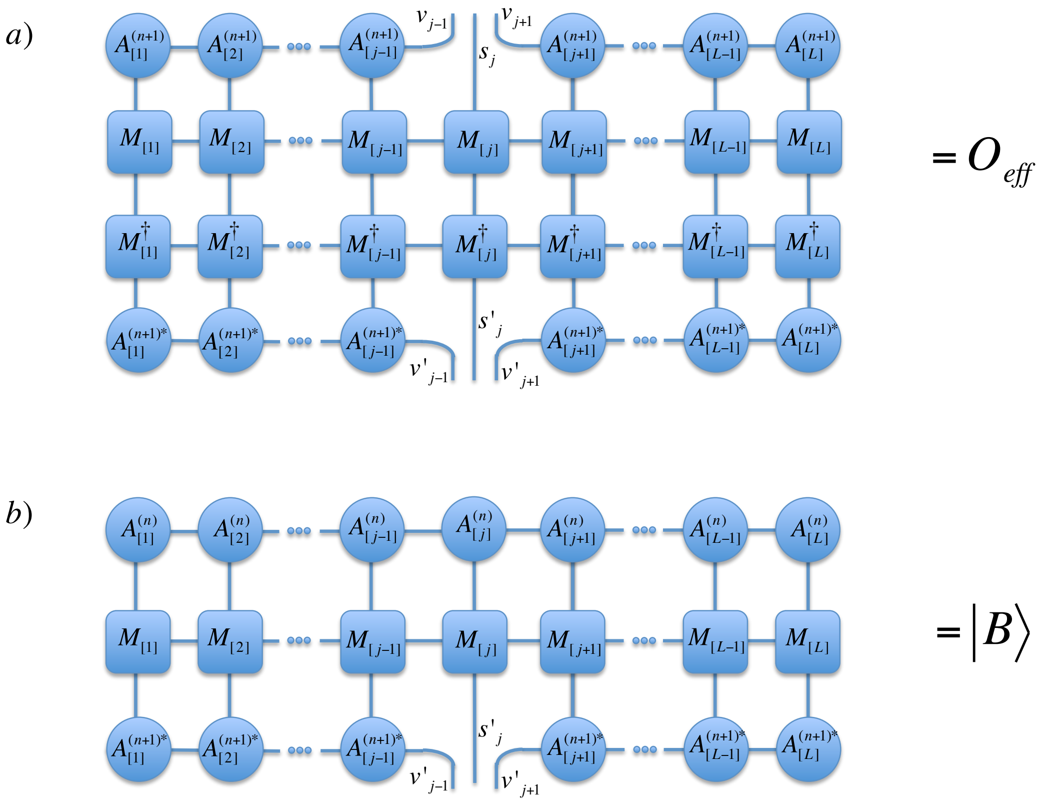

Figure 6.1: (Color online) Graphical notation showing the structures of the SIMPS algorithm, during the optimization step j of Eq. 6.16, where we are seeking a new state $|\phi_{n+1}\rangle = \tilde{H}|\phi_n\rangle$. (a) The matrix O_{eff}, which represents the optimization of the term $\langle\phi_{n+1}|O^{\dagger}O|\phi_{n+1}\rangle$ with respect to the tensor at site j. The operator O has been represented by an MPO with tensors M. Note that the virtual and physical indices are grouped in the same manner as in the matrix H_{eff} in traditional DMRG. (b) The vector representing the term $\langle\phi_n|O|\phi_{n+1}\rangle$ during the update of site j. Note the distinctions between the ket tensors $A^{(n)}_{[j]}$, which represent the state $|\phi_n\rangle$ from the previous iteration, and the tensors $A^{(n+1)}_{[j]}$ which represent the state being updated, $|\phi_{n+1}\rangle$.

the best convergence appears to come from sweeping "back and forth" along the chain until convergence is reached. The complete algorithm can be thought of as a pair of nested loops, outlined as follows:

- Initialize a random initial state $|\phi_0\rangle$
- Loop 1: Iteratively apply $\tilde{H}$ to get $|\phi_{n+1}\rangle = \tilde{H}|\phi_n\rangle/||\tilde{H}|\phi_n\rangle||$ until we converge to the target state
    - Loop 2: For each application of $\tilde{H}$, sweep back and forth across the state, optimizing tensors as described above until the resulting state converges

Numerical Considerations and Convergence

The numerical tolerance and convergence in these two loops is controlled in our implementation by five parameters, with two parameters for the "inner" loop and three for the outer.

In the inner loop (where we are sweeping across the state to find $|\phi_{n+1}\rangle = \tilde{H}|\phi_n\rangle$), we set our convergence based upon the quantity $\delta_1 \equiv |\langle\phi_n|O|\phi_{n+1}\rangle|$, which clearly should converge to zero when $|\phi_{n+1}\rangle$ is in the desired state because $O = \tilde{H}^{-1}$. As observed in [59], this quantity can be computed essentially for free, because it is equivalent to the overlap $\langle A_{[j]}|B_{eff}\rangle$. Hence, we set some tolerance ϵ_1 and consider the sweeping process to have converged when $1 - \delta_1 < \epsilon_1$.

In practice we have observed that in the early stages of the algorithm, this sweeping back and forth across the state often "stalls" in the sense that δ_1 may asymptotically approach some limit which is strictly less than 1, and hence that the convergence criterion with respect to δ_1 may never be reached. This is likely due to finite bond-dimension effects: although we are searching for a final state $|\phi_{n\to\infty}\rangle$ which we believe can be efficiently represented by an MPS because of its localized properties, there is no clear reason why the intermediate states $|\phi_n\rangle$ must be similarly expressible. Hence the convergence may stall when the current MPS does not represent the desired state $\tilde{H}|\phi_n\rangle$, but rather only the best approximation availabile at the given bond dimension. To handle this, we consider after each sweep the quantity $\delta_2 \equiv \delta_1^{(k)} - \delta_1^{(k-1)}$ which simply measures the change in our convergence quantity between successive sweeps $k-1$ and k. We set an accompanying tolerance ϵ_2 and exit the loop, assuming the process to have stalled, if δ_2 drops below this threshold. Typically, this kind of early exit is tolerable, and may actually improve the speed of the algorithm by avoiding fruitless sweeping steps. The resulting vector $|\phi_{n+1}\rangle$ will be imperfect, but will still have greater overlap with the target state than $|\phi_n\rangle$, and additional applications of $\tilde{H}$ in subsequent steps will continue to rotate us towards the desired state. In practice, we have seen that as long as the quantity α is relatively close to 1 (e.g. perhaps $\delta_1 > 0.8$) the state will continue converging towards the desired target, at which point the MPS representation will become more accurate and the "stalling" problem will vanish.

Having described how one might establish convergence in the inner loop, we turn out attention to the outer loop, and consider the question of how to decide how many times $\tilde{H}$ must be applied before the target state has been reached to a good approximation. We use two additional quantities to measure this. The first is simply $\delta_3 = |E_n - E_{n+1}|/E_n$, the relative change in the energy of the state (with respect to the *original* Hamiltonian) between steps n and $n+1$. We also examine the variance of the energy $\delta_4 = \langle \Delta H^2 \rangle$ (using methods such as those proposed in Chapter 4); this can be an important convergence check for any algorithm and particularly in this context, as we wish to be sure we are studying proper localized eigenstates(rather than simply a superposition of such). Since the energy fluctuation should vanish in a true eigenstate, we can also set a convergence threshold for this quantity, and reject any states whose variance is too large.

Unfortunately, even with these checks in place, in practice we have occasionally seen the algorithm converge towards what appears to be a superposition of eigenstates (as judged by comparison to an exact diagonalization result). This is not entirely uncommon, since two such neighboring states will both produce large eigenvalues in the shifted-and-inverted spectrum, and although theoretically in the large-n limit only the dominant eigenvector will remain, it may take many steps before this is the case, and components from the neighboring state will persist longer than any other. But a superposition of these states may still show very little change in the energy values after successive steps of the algorithm, and even a relatively small energy variance. To avoid being fooled into accepting such a superposition, we use one additional convergence check: the overlap between states at subsequent steps of the algorithm, $\delta_5 = \langle \phi_n | \phi_{n+1} \rangle$. When two states are close to the target energy, this quantity will tend to decrease at first, but then may begin to increase temporarily until the algorithm fully settles in to one state. In practice this is a very strong convergence check.

Note that naturally, one might wonder if the performance of the algorithm can be improved by simply squaring the MPO (i.e., taking $\tilde{H} \to \tilde{H}^2$) so that at each step of the outer loop we apply more than one inverse. Our tests with this method show that in practice, the required increase in the bond dimension of the MPO slows the construction of the structures O_{eff} in a manner which overcomes the potential advantage. The sweeping procedure also experiences more instability and has greater difficulty converging to a result.

6.2.2 Comments on Other Algorithms

The MPS algorithms which have emerged thus far for studying excited MBL states fall roughly into two categories. Algorithms such as SIMPS (and less directly, the algorithm given in Ref. [61]) have a tunable parameter which allows particular energy levels to be targeted. But these algorithms do not explicitly seek out localized states; instead, they will attempt to converge *any* eigenstate near the targeted energy level, with the understanding that, because only localized states will admit an efficient MPS representation, it is only these states which will be produced when the algorithm successfully converges.

On the other hand, algorithms like DMRG-X [60] and Es-DMRG [59] explicitly pursue states with a localized structure. Recall that the typical DMRG algorithm (see Sec. 3.1) involves sweeping over the sites of an MPS, and at each site, an updated tensor is chosen from the eigenstates of an effective Hamiltonian H_{eff}. For ground state DMRG, we choose the tensor which minimizes energy of the state as much as possible. In this way, we "greedily" pursue the global ground state. In these modified MBL algorithms, the optimization step is changed so that instead of choosing a new tensor which minimizes the energy, we choose a new tensor which in some way makes the state look as much as possible like a localized state. In DMRG-X, this is done by choosing an initial state which is a product state which resembles the l-bit structure of a localized excitation. Then at each state of the DMRG update, instead of minimizing the energy we choose from the spectrum of H_{eff} the tensor which *minimizes the change of the current state*–i.e., the tensor such that the new, updated state will have as much overlap as possibly with the previous state. A similar approach is used in Es-DMRG, choosing the updated tensor to minimize the *change* in the energy of the state. In both cases, the overall goal is to start with a trial wavefunction that looks like a localized state, and then optimize the tensors to find the eigenstate of the Hamiltonian which is in some sense "closest" to the starting point. In our experience implementing these algorithms, individual steps of Es-DMRG can be faster because one can find the desired tensor directly. If the current energy is E_0, the smallest magnitude eigenvalue of $(H_{eff} - E_0)^2$ (which can be found efficiently by Lancszos methods [194]) will give the desired update. In DMRG-X, one must find many eigenstates of H_{eff} and try each to determine which gives maximum overlap with the current state. But while individual steps are slower, tests described in Ref. [60] tend to show that DMRG-X requires far *fewer* steps to converge, so that in practice it is often the faster of the two. Our preliminary tests with the algorithm agree with this conclusion.

Note that this second family of algorithms has properties exactly reversed compared to algorithms like SIMPS: by their nature, they aggressively pursue only states which look localized. Yet as a result, it is difficult if not impossible to target a particular energy: instead, the energy of the resulting state is controlled by the properties of the initial product state which was fed into the algorithm. Fine-tuning this state may allow one some rough control over the energy of the result, but in general one must simply explore a wide range of possible initial states in order to broadly sample the MBL states in the Hamiltonian. These traits form the principle trade-offs between available MBL MPS algorithms (though the "En-DMRG" approach of Ref. [62] seeks to somewhat blend these possibilities). Note also that all algorithms here (but particularly SIMPS and DMRG-X) seem capable of finding resonances in the spectrum, an area of particular interest in the study of MBL states [55].

6.3 Entanglement Results

6.3.1 Random-Field Heisenberg Model

Perhaps the most widely-studied example of MBL behavior is the spin-1/2 Heisenberg model with a disordered random field. This model is given by

$$H_W = \sum_i \sigma_i^x \sigma_{i+1}^x + \sigma_i^y \sigma_{i+1}^y + \sigma_i^z \sigma_{i+1}^z + h_i \sigma_i^z. \tag{6.23}$$

The important feature of this model is in the field-strength coefficients $\{h_i\}$, which vary from site to site and at each site are drawn randomly from a uniform distribution within the interval $[-W, W]$. The model parameter W therefore quantifies the strength of the disorder. A considerable body of work (e.g. Refs. [54, 220, 230, 237, 238]) using both the MBL indicators discussed above and other methods has shown that this model appears to have a transition from ergodic to localized somewhere between $W = 3$ and $W = 4$, with most evidence indicating that it occurs close to $W = 3.7$ [220]. Some of the variation in the reported locations of this transition comes from the fact that the tails of the spectrum are likely to localize before the states in the middle [55, 220], so that work which looks only at states in the center may report a larger value of the critical disorder strength W_c than work which averages over the entire spectrum. In our work below, we will be looking for localization among states in the middle of the spectrum; if one is to refer to "the" critical disorder value of the model, it is most sensible to define this as the value which localizes the entire spectrum.

6.3.2 Concurrence

We now turn to the question of concurrence and negativity as indicators of localization, as well as the relationship between these two measures in the context of an MBL system. The random-field Heisenberg model is also the model studied in Ref. [65]. In this work, the authors detect evidence of the transition consistent with previous work by looking at the behavior of the *concurrence*, a measure of entanglement introduced in Ref. [63]. We first briefly review in relevant part the definition and properties of this measure.

Concurrence often arises in the context of entanglement of *mixed* systems; in this context, we say that a (mixed) state is unentangled if it can be expressed as a mixture of unentangled pure states; otherwise it is an entangled state. For these mixed states, several different general measures of entanglement have been proposed and their merits debated at length (a topic well beyond the scope of this work; see instead Refs. [73, 207, 239]). Concurrence appears as part of an important formula for one such measure (the entanglement of formation [207]) but has emerged as an important entanglement measure in its own right.

Originally, concurrence was defined for two-qubit systems (e.g. two spin-1/2 systems or their equivalent), and although generalizations to higher dimensional systems have been proposed [204–206], this remains the only setting in which a convenient and closed-form expression has been obtained which would allow for direct calculation. In this context, the concurrence is given by comparing the two-qubit system with a spin-flipped version of itself. If the system were a pure state in the computational (σ^z) basis, we could effectuate the spin-flip with a time-reversal operation, i.e. with complex conjugation and the Pauli operator σ^y on each spin. Hence if we have instead a mixed state described by a density matrix ρ, its spin-flipped counterpart is given by

$$\tilde{\rho} = (\sigma^y \otimes \sigma^y)\rho^*(\sigma^y \otimes \sigma^y) \tag{6.24}$$

As shown in [204], we can now construct a sensible measure of entanglement by taking the matrix $\rho\tilde{\rho}$ and finding its eigenvalues λ_i. Let these eigenvalues be in descending order so that $\lambda_1 \geq \lambda 2 \geq \ldots$. In this case, the concurrence is given by

$$\mathcal{C} = \max(0, \sqrt{\lambda_1} - \sqrt{\lambda_2} - \sqrt{\lambda_3} - \sqrt{\lambda_4}) \tag{6.25}$$

Note that the eigenvalues of $\rho\tilde{\rho}$ are guaranteed to be nonnegative so that Eq. 6.25 is always well-defined.

As a measure of entanglement in mixed states, concurrence is potentially well-suited for use in the context of MBL systems, which are defined in terms of the properties of their (potentially mixed) subsystems. In particular, concurrence obeys monogamy of entanglement [240], so that when two spins are highly entangled with one another, they are less able to be entangled with the remainder of the system. Intuitively, this suggests that the concurrence between locally neighboring spins should be closely related to localization within the system; if subsystems are to be mutually connected in a manner that allows their environment to act as a reservoir, they cannot be too strongly entangled as pairs of particles.

This observation motivated our interest in the "total nearest neighbor concurrence" $\mathcal{C}_{NN}^{tot}$ of a spin system as an indicator of localization: we consider all pairs of nearest-neighbors in a system, calculate the concurrence of each, and sum the results. The resulting quantity should be small in a delocalizes system, since few if any pairs of spins have large amounts of entanglement with each other, preferring instead to entangle with the broader environment. Conversely, the presence of even a few highly entangled pairs of neighboring spins, as in a localized state, can make the quantity grow rapidly. And indeed this behavior is exactly what we find for the random Heisenberg model (in which we average the quantity over many different disordered realizations). As shown in Fig. 6.2, the total concurrence is small for weak values of the disorder W, and grows rapidly in the region near the known value of the transition to MBL. For comparison, we also consider the NPR of the state (normalized by the volume of the Hilbert space), and find that the increase in total concurrence coincides with the expected decrease of NPR seen in localized systems. It was during the early stages of this investigation into the concurrence that we became aware of the work in Ref. [65], which explores the behavior of concurrence in localized systems to a greater depth (including similarly demonstrating that the increase in concurrence coincides with the onset of localization by comparing to the average level spacing ratio $\langle r_n \rangle$).

6.3.3 Negativity

Concurrence is a powerful measure of entanglement for two-qubit systems. But as previously remarked, it also possesses substantial limitations which prevent it from addressing the full range of questions which arise in MBL systems. But an alternate measure of entanglement in mixed systems, the *negativity* can be computed directly for subsystems of arbitrary dimension (not simply qubits) and arbitrary system sizes sizes (not simply two spins). It has also recently been shown that tensor network techniques can be used to extract negativity even for large and difficult systems [241], although within the scope of this work it will suffice to perform the straightforward calculation.

Negativity [64] arose in the wake of the Peres-Horodecki [242, 243] criterion,

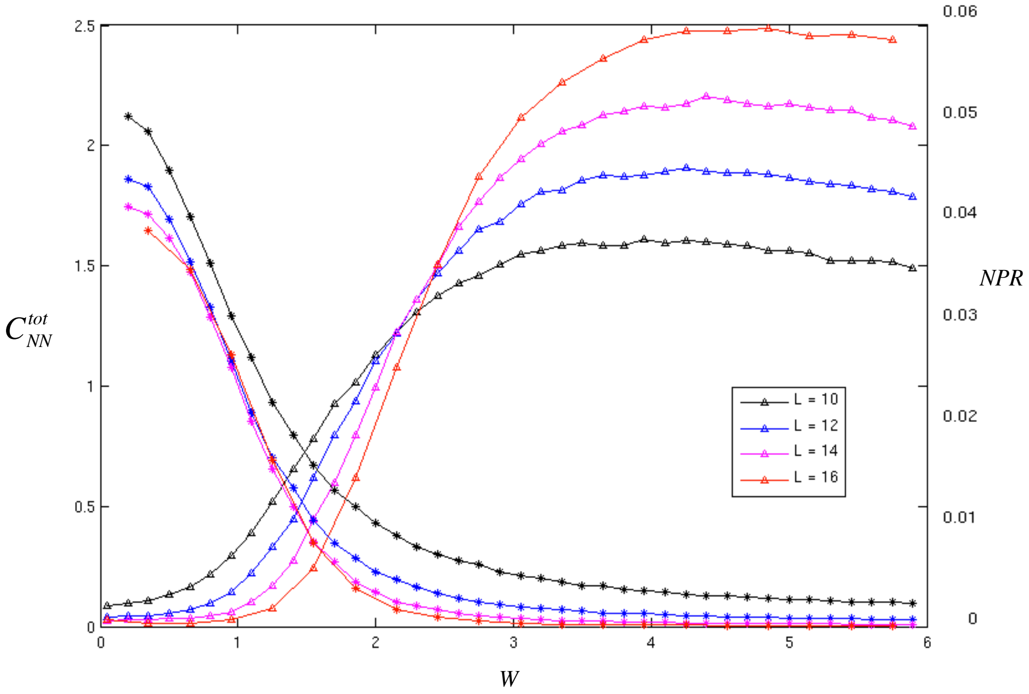

Figure 6.2: (Color online) The total nearest-neighbor concurrence $\mathcal{C}_{NN}^{tot}$(triangles, left axis) and the normalized participation ratios (asterisks; right axis) for the random-field spin-1/2 Heisenberg model, shown for three system sizes. Data are collected using exact diagonalization; 1000 disordered configurations are computed, and for each we compute 50 eigenstates from the middle of the spectrum, where the behavior most resembles the thermodynamic limit. A sharp decrease in the NPR is known to show the transition to the localized phase (estimated for this model to occur around $W_c = 3.7$ [220]), and the concurrence is seen to increase significantly just as the NPR decreases. Note that the NPR values here have been normalized by the volume of the Hilbert spaces so that we can compare different system sizes. The crossing point in the concurrence curves is not considered to precisely identify the transition point, as it does not persist in any finite-size scaling relation.

a necessary condition for a density matrix to represent a separable (unentangled) quantum state. To understand this criteron, as well as the associated concept of negativity, we require the concept of a "partial transpose," [73] in which a matrix is transposed only with respect to a particular subsystem. For example let (α, β) and (γ, δ) be the composite indices of some matrix M, so that the matrix can be written as

$$M = \sum_{\alpha,\beta,\gamma,\delta} m^{\alpha\beta}_{\gamma\delta} |\alpha\beta\rangle\langle\gamma\delta|. \tag{6.26}$$

A full transpose of this matrix is simply given by swapping incoming and outgoing indices, i.e.

$$M^T = \sum_{\alpha,\beta,\gamma,\delta} m^{(\alpha\beta)}_{(\gamma\delta)} |\gamma\delta\rangle\langle\alpha\beta|. \tag{6.27}$$

But if, the indices are truly composite, i.e. $|\alpha\beta\rangle = |\alpha\rangle \otimes |\beta\rangle$ and $|\gamma\delta\rangle = |\gamma\rangle \otimes |\delta\rangle$, then we can "ungroup" the composite indices, and could consider the matrix (now in a certain sense a rank-4 tensor) as comprising two subspaces,

$$M = \sum_{\alpha,\beta,\gamma,\delta} m^{\alpha\beta}_{\gamma\delta} |\alpha\rangle\langle\gamma| \otimes |\beta\rangle\langle\delta|. \tag{6.28}$$

In this perspective we can speak of a "partial transpose" with respect to only *one* of these two subspaces. For example, with respect to the $|\beta\rangle\langle\delta|$ subspace, the partial transpose gives

$$M^{T_{\beta\delta}} = \sum_{\alpha,\beta,\gamma,\delta} m^{\alpha\beta}_{\gamma\delta} |\alpha\rangle\langle\gamma| \otimes |\delta\rangle\langle\beta|. \tag{6.29}$$

The Peres-Horodecki condition states that, if ρ is a density matrix on a composite space $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$, then ρ is separable only if the partial transpose ρ^{T_A} has non-negative eigenvalues (we choose subspace A w.l.o.g. because $\rho^{T_B} = (\rho^{T_A})^T$ and the full transpose operation does not change the eigenvalues). In our case, where we are particularly interested in measuring entanglement, it is the contrapositive of this statement which holds greater significance: if ρ^{T_A} *does* have negative eigenvalues, ρ must have represented an entangled state (with respect to the systems $\mathcal{H}_A$ and $\mathcal{H}_B$.

Thus, we define the negativity as a measure of entanglement to represent the extent to which this criterion is violated: it is the sum of the magnitudes of all negative eigenvalues which appear after partial transposition. Because a proper density matrix has unit trace, the size of this sum is inherently normalized and can be compared between systems. In practice, this recipe for the negativity can be compactly written as

$$\mathcal{N}(\rho, A) = \sum_i \frac{|\lambda_i| - \lambda_i}{2} \tag{6.30}$$

where as always the elements $\{\lambda_i\}$ represent the eigenvalues of ρ^{T_A}.

Since the negativity shares many properties with the concurrence, and also displays the monogamy of entanglement [244], it is natural to ask whether it also serves as an indicator of MBL. By analogue to how we approached the concurrence, we consider the total nearest-neighbor negativity of a state. For each pair of nearest neighbors, we construct the (reduced) density matrix for these two spins, and then partially transpose with respect to one of the spins to compute the negativity. The sum of these negativities for all neighboring pairs give us our candidate quantity $\mathcal{N}_{NN}^{tot}$. We apply this criteria to the same system as before, and find very similar behavior. As shown in Fig. 6.3, we once again find that our measure of entanglement remains small until the transition, at which point it begins to grow rapidly. As was the case with the concurrence, the total negativity for points in the localized phase grows with the system size.

in Ref. [65], a scaling relationship was found empirically between the second derivative of $\mathcal{C}_{NN}^{avg}$ and the length of the system, with the scaling collapse used to extract a specific estimation of the localized-delocalized transition. Note here that we are using now the per-site average of the concurrence rather than the total concurrence. In particular, it was shown that there is a universal scaling function Φ for which data collapse can be observed from, namely

$$\frac{1}{L^a} \frac{d^2 \mathcal{C}_{NN}^{avg}}{dW^2} = \Phi\left(L^b (W - W_c)\right), \tag{6.31}$$

with parameters $a \approx 0.5$, $b \approx 0.6$, and $W_c = 3.7$. The authors argue that this scaling of the second concurrence derivative can be used as another method to identify the location of the localized-delocalized transition.

Naturally, we wish to see if the same holds true for $\mathcal{N}_{NN}^{avg}$. Since taking derivatives amplifies noise, our data sets are not yet sufficiently smooth to allow us to repeat this analysis directly. However, as preliminary evidence that the behavior of the negativity is indeed comparable, we can perform a numerical fit of our data, and consider instead the derivatives of the fit. Fortunately, the data in the relevant domain an be fit very well by a polynomial of degree 9, as shown in Fig. 6.4. To avoid overfitting, we have compared the result for various subsets of our data and found it to be relatively stable.

As shown in Fig. 6.5, if we use the same scaling parameters as above we indeed see evidence of universal scaling. The data collapse is particularly

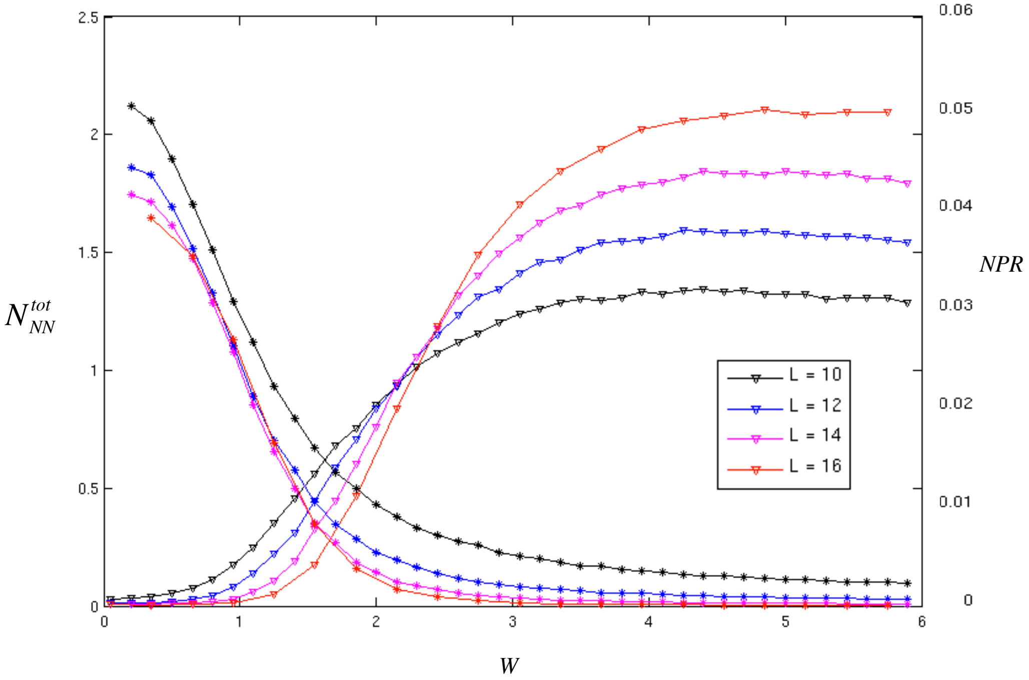

Figure 6.3: (Color online) The total nearest-neighbor negativity $\mathcal{N}_{NN}^{tot}$(inverted triangles, left axis) and the normalized participation ratios (asterisks; right axis) for the random-field spin-1/2 Heisenberg model, shown for three system sizes. Data collection is identical to the concurrence results in Fig. 6.2. As with the concurrence, a sharp decrease in the NPR near the transition also coincides with a rapid increase in the negativity. As before, the NPR values here have been normalized by the volume of the Hilbert spaces, and the crossing point in the negativity curves is not likely of physical significance, as in the case of concurrence.

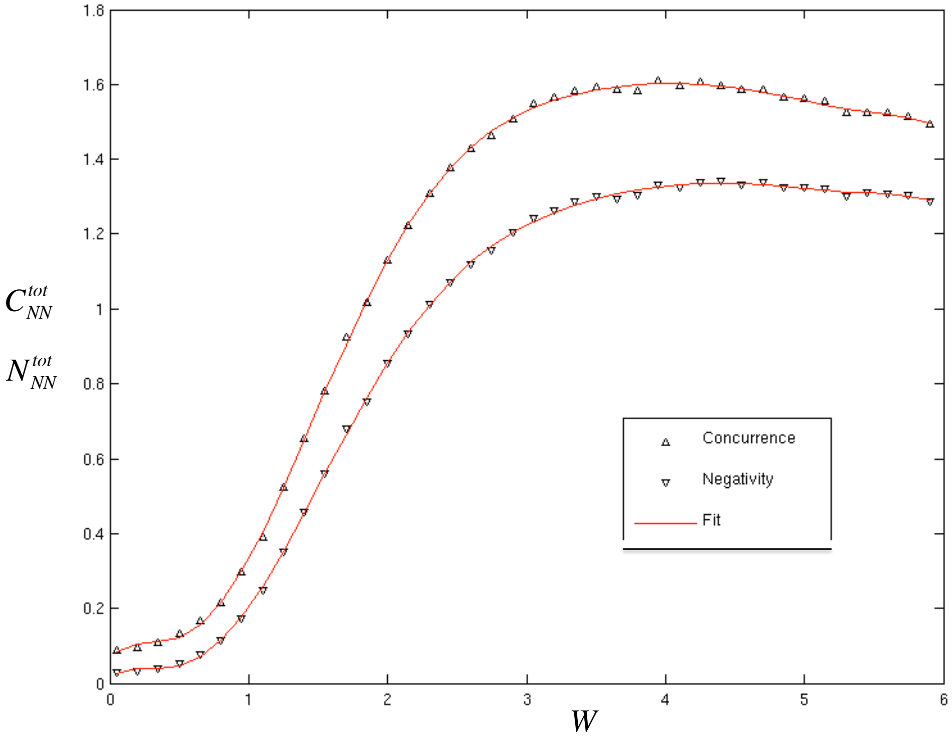

Figure 6.4: (Color online) L = 10 data for concurrence (triangles) and negativity (inverted triangles) are presented together, along with numerical fits from a polynomial of degree 9. The fit is in good agreement with the qualitative behavior of the data, particularly in the transition region.

evident for systems with $L > 10$, likely because the smallest system is the most subject to finite-size effects. Since we do not see any evidence of the scaling relationship except at these particular values of a, b, and W_c, we argue that our data also support the conclusion that this scaling can be used to locate the localized-delocalized transition. Furthermore, because we find virtually identical behavior in the second derivative of the negativity (Fig. 6.6), it seems that either measure of entanglement may be used for this purpose. Verifying this result directly–i.e., without using a fit to generate the derivatives–should be possible once we have compiled additional disordered samples to increase the statistics of our states. This will be an important part of our future work in this area.

It was also demonstrated In Ref. [65] that the concurrence between two arbitrary spins (not necessarily nearest neighbors) displays distinctive behavior

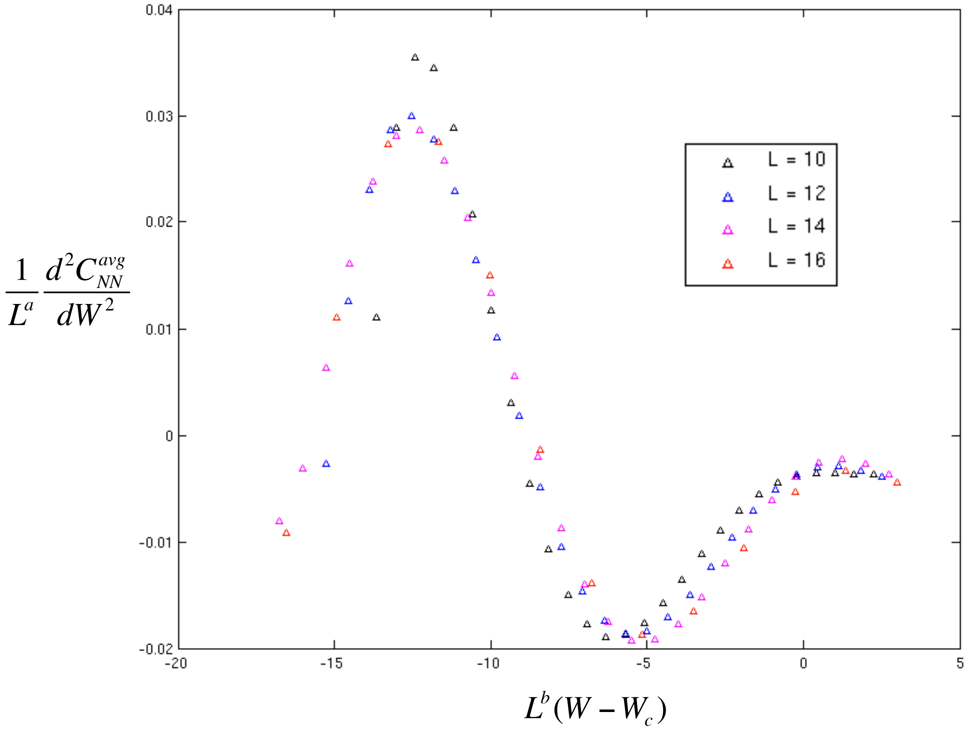

Figure 6.5: (Color online) Total nearest-neighbor concurrence data is shown for various system sizes, in relation to the scaling function in Eq. 6.31. We use the values of $a \approx 0.5$, $b \approx 0.6$, and $W_c = 3.7$ which were identified in Ref. [65]. Clear evidence of scaling can be seen for these values, but not for other choices. The regions where the data "overshoot" one another. Data collapse is particularly clear when $L > 10$, since these systems have smaller finite-size effects.

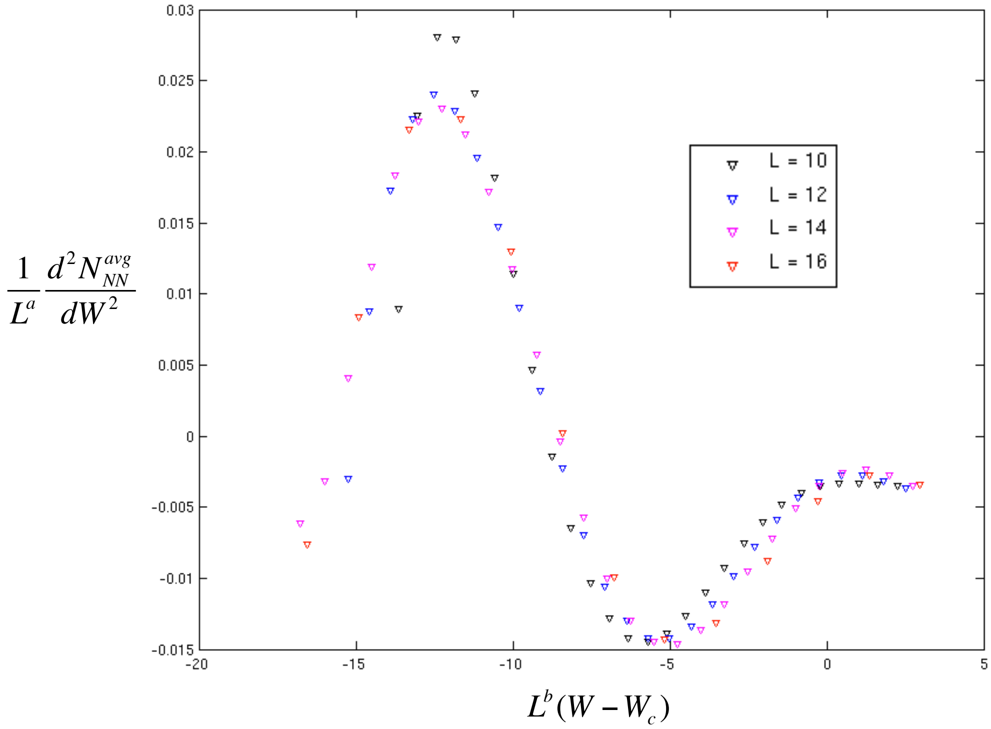

Figure 6.6: (Color online) Total nearest-neighbor negativity data is shown for various system sizes in relation to the scaling function in Eq. 6.31. Comparing the result here to Fig. 6.5, we see that both the concurrence and the negativity show very similar scaling behavior, suggesting that both can be used to detect the localized to delocalized transition point.

in the localized phase: for two spins at sites i and j, the concurrence $\mathcal{C}(\rho_{ij})$ decays exponentially with the distance $|i-j|$ between them. This behavior appears only in the localized phase; in delocalized phase, the concurrence between two spins is negligible and essentially independent of the distance between them. Hence, the observation of such exponential behavior is itself a diagnostic criterion of MBL. Furthermore, the specific nature of the exponential decay can be used to study the nature of the localization. It is shown in [65] that one can extract a characteristic entanglement length, ξ_E, from the decay constant of the exponential, i.e.

$$\mathcal{C}(\rho_{i,i+d}) \propto \mathcal{C}(\rho_{i,i+1})e^{-d/\xi_E}. \tag{6.32}$$

For the systems we have considered, it appears that the negativity shares not only a similar exponential form, but the same entanglement length. In Fig. 6.7, for example, we show (in semilog scale) the concurrence and entanglement (averaged over disorder) for a system of length $L=20$, with states computed from SIMPS. Up to the level of the statistical noise resulting from the disordered average, the lines have very comparable slope. This is representative of similar behavior has been seen for shorter systems as well. This further suggests that the negativity can also be used for the same purposes as the concurrence.

6.3.4 Relationship Between Entanglement Measures

Finally, we briefly make two comments on the relationship between the negativity and total concurrence. For the case of two-qubit systems, strict analytical bounds exist for the relationship between these two measures. While both quantities cover the interval $[0,1]$, the negativity is always bounded above by the concurrence, and bounded below by [245]

$$\mathcal{N} > \sqrt{(1-\mathcal{C})^2+\mathcal{C}^2}-(1-\mathcal{C}). \tag{6.33}$$

From these relations, it is clear why the negativity also displays exponential decay with distance in the localized phase, since it is bounded above by an exponentially decaying quantity, although it is not immediately obvious that the decay constants should be the same. The existence of the lower bound in this case also helps to motivate the similarity in the qualitative behavior across the transition. We note (see Fig. 6.8) that states in the delocalized regime seem more likely to saturate the lower bound, whereas localized states are more likely to saturate the upper bound where the two quantities are equal.

The other important fact about the relationship between concurrence and negativity is that the two measures, while closely related, do not share the same ordering of entanglement. In other words, consider a pair of two-qubit

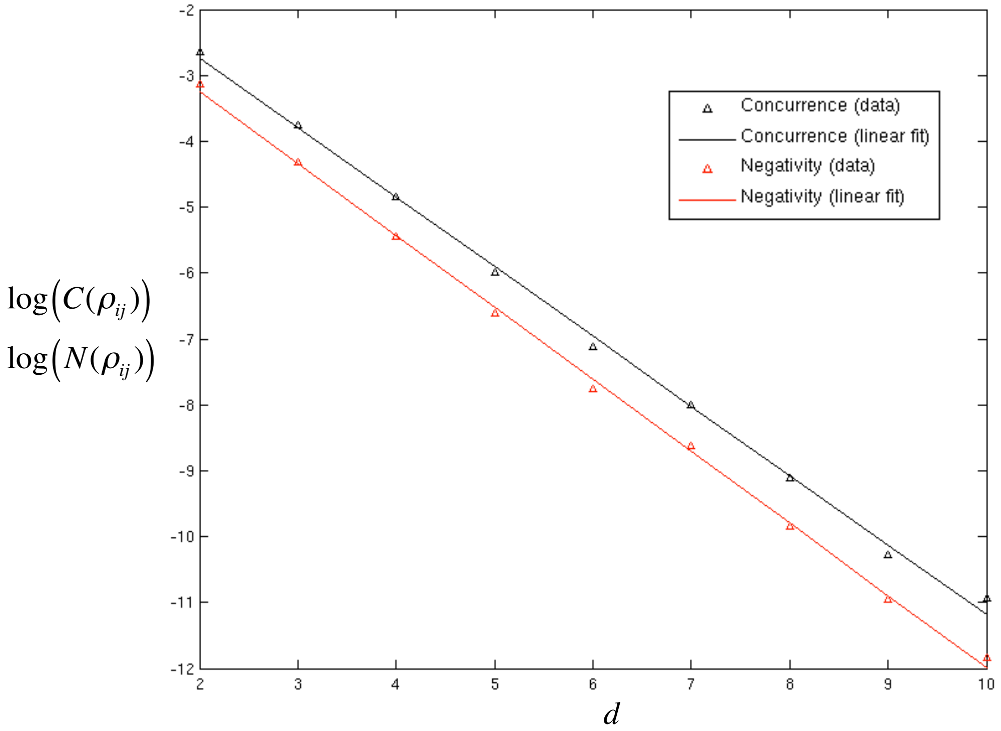

Figure 6.7: (Color online) Using Data from SIMPS, we compute the concurrence and negativity between sites i and j in a disordered Heisenberg chain of length 20 in the localized regime ($W = 6$). Both quantities decay exponentially with the distance $d = |i - j|$ between the sites; note that here we have plotted the log of each (concurrence in black, negativity in red). Fit lines are included as a guide to the eye and to demonstrate that the rate of decay is very similar for both negativity and concurrence, suggesting that both quantities capture the same characteristic entanglement length. The slope of the concurrence fit line is $m_{\mathcal{C}} = -1.08$; for negativity it is $m_{\mathcal{N}} = -1.12$. This behavior is representative of other system lengths as well, although the slopes appear to have some weak dependence on the strength of disorder. Note that we show here only distances out to $d = L/2$; beyond this point the entanglements are so close to zero that they are very sensitive to noise and finite size effects, as discussed in Ref. [65].

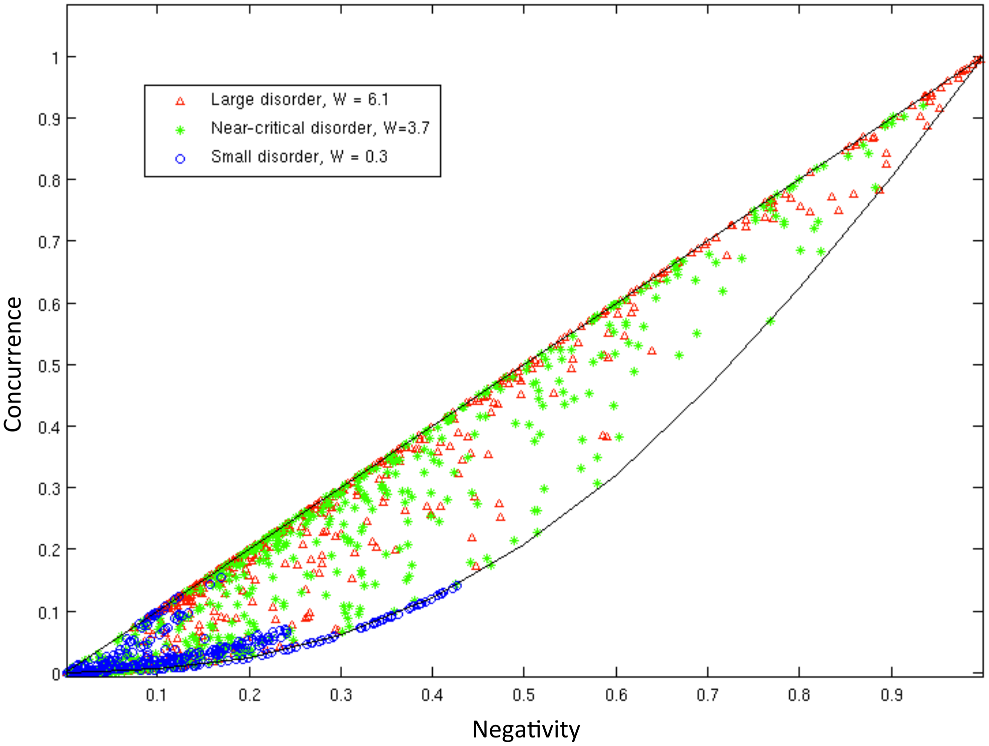

Figure 6.8: (Color online) Negativity and concurrence are compared for different disordered samples in an $L = 12$ system (data from exact diagonalization). Black lines indicate the analytic bounds on the relationship between concurrence and negativity for the two-quibit case; the negativity is never larger than the concurrence, and always larger than $\sqrt{(1-\mathcal{C})^2+\mathcal{C}^2}-(1-\mathcal{C})$. Qualitatively, we see that in subsystems from delocalized states (blue circles), both quantities are relatively small, and the lower bound is more likely to be saturated. In the case of strong disorder (red triangles) the full range of possible values for each is explored, with the upper bound more likely to be saturated. The intermediate case near the localized/delocalized transition shows behavior much more akin to the fully localized case, though this may be the result of finite size effects which tends to increase localized behavior in small systems.

states ρ_1 and ρ_2; if we observe that $\mathcal{C}(\rho_1) > \mathcal{C}(\rho_2)$, it does *not* necessarily imply that $\mathcal{N}(\rho_1) > \mathcal{N}(\rho_2)$ [246, 247]. We have found that, among the two-qubit subsystems of states in the disorderd Heisenberg model, such "ill-ordering" are uncommon but hardly rare (see Fig. 6.9). The fraction of the subsystems showing this property seems relatively constant with disorder strength; this area is left as a direction for future study.

6.4 Summary and Outlook

In this chapter, we have reviewed the phenomenon of many-body localization, in which closed quantum states states fail to thermalize (in the sense of the ETH) despite the presence of interactions in the system. Systems which display this localization display a variety of notable, diagnostic properties. Among these is the fact that MBL states feature an area-law for the growth of their entanglement entropy with respect to the size of a subsystem. As a result, these states admit efficient representations as matrix product states, and we have reviewed the very recent literature proposing algorithms to find these representations. In particular, we present our implementation of the "SIMPS" algorithm, including commentary on its various convergence criteria.

We also consider the role that local entanglement measures can play in identifying MBL systems by looking at the behavior of the total nearest-neighbor concurrence $\mathcal{C}_{NN}^{tot}$ and total nearest-neighbor negativity $\mathcal{N}_{NN}^{tot}$ in the context of the random-field spin-1/2 Heisenberg model. The former was also recently examined in Ref. [65], in which it was shown that $\mathcal{C}_{NN}^{tot}$ is small for the delocalized regime, and increases rapidly around the transition until the localized phase is reached. The behavior there was shown to coincide with the behavior of the average level spacing ratios; here we have also compared it to the normalized participation ratio. Furthermore, by making a similar comparison for $\mathcal{N}_{NN}^{tot}$, we provide evidence that the negativity can be used as a comparable indicator of localization. This could be quite useful, as the negativity is calculable for a wider variety of systems and subsystems, and hence could be applied to study localization on a wider array of contexts.

Several major directions remain for future work in this area. First and foremost, it would be desirable to continue gathering disordered samples to increase the statistics for the data presented here, so that the behavior of the derivatives could be analyzed directly. With approximately 10 to 100 times more data, the curves would likely be smooth enough to make this comparison [248] without using a fit.

Additionally, it would be desirable to test both the concurrence and the negativity for additional models displaying MBL, to ensure that their diag-

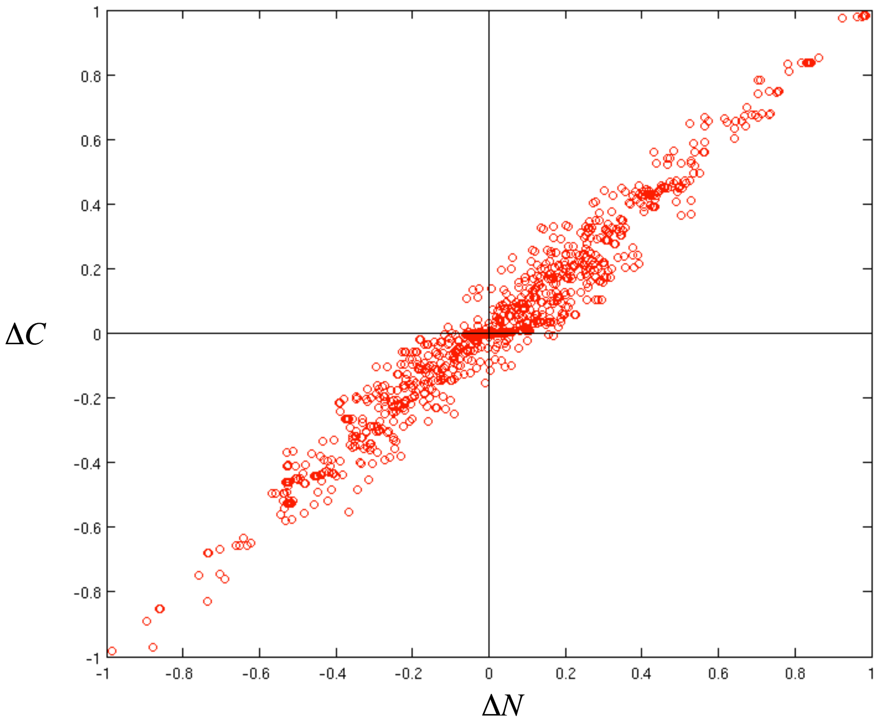

Figure 6.9: (Color online) The ordering of concurrence and negativity is considered by comparing equivalent two-qubit subsystems from pairs of different states in the disordered ensemble. The difference between concurrences is plotted versus the difference in negativities. Thus, data points in quadrants I and III represent "well-ordered" pairs, in which the state with the larger concurrence also has the larger negativity. The presence of data points in quadrants II and IV shows also that "ill-ordered" subsystems are possible, though not common. Data plotted here are from a system of $L = 20$ and $W = 6$, though the pattern is typical for states, at least in the localized regime.

nostic power is not model-dependant. A natural place to start would be the quasiperiodic Heisenberg model– furthermore, since quasiperiodic models do not require averaging over large numbers of disorder samples, it may be easier for such a model to achieve smooth datasets and examine scaling properties of the derivatives.

Next, having established an argument that the negativity is comparable to the concurrence in localized systems, it would be desirable to take advantage of the greater generality and apply the negativity to a disordered (or quasiperiodic) spin-1 system. Such systems seem not to have been widely studied numerically in the context of MBL, likely in part because exact diagonalization for higher spin systems is much more restricted by memory limitations and chains beyond length 10 can be extremely difficult to compute; with the MPS algorithms above, however, such systems should still be possible to probe, at least from the localized side. A disordered version of the spin-1 Heisenberg model may be a natural place to start. Finally, it would be interesting to perform further study of the ordering of concurrence and negativity in the disordered regimes. Data for these and other projects are being pursued for future work.

Chapter 7

Conclusions

In this work, we have considered the use of tensor network structures in quantum many-body physics. These structures can take the place of a general coefficient tensor in specifying the expansion of a quantum state, and when they do, they can greatly reduce the amount of information required to specify the state. For states with the right set of properties, such as an entanglement area law, these tensor network representations of the states are both efficient and faithful. Consequently they are well-suited for use in numerical algorithms which calculate and study quantum states; we have presented details of several such algorithms and discussed sound numerical practices to optimize their performance in the early sections of this dissertation.In the remaining chapters, we have presented applications of these tensor network techniques in the context of three major areas of interest in many-body quantum physics: locating phase transitions, characterizing phases beyond the symmetry breaking paradigm, and detecting many-body localization.

We have showed that the Binder cumulant from quantum Monte Carlo can also be applied profitably in the language of tensor networks and matrix product states, and demonstrated for various models that it can compute both the location of the phase transitions and the values of certain critical exponents to a high degree of accuracy. Although most of our techniques were discussed in the context of one-dimensional states, we also shown that they can be applied to two-dimensional systems. Our proof-of-concept application to the Ising model on a square lattice was limited in its accuracy by the use of a simple "local update" code to compute the states, and by the need to perform approximations in the lattice contractions while computing the cumulants. But with newly available PEPS algorithms and contraction techniques, a more careful study of this model with the Binder cumulant might yield a very precise measurement of the critical point. The use of these methods to study transitions in disordered systems would also be an interesting direction for future work.

Our study of symmetry protected topological phases presented a two-parameter Hamiltonian of our construction containing an A_4 onsite symmetry as well as symmetry under lattice translation and inversion. For this model, we performed a detailed "scan" of the parameter space and created a numerical phase diagram showing eight different types of quantum phase, ranging from traditional symmetry-breaking phases to SPTs distinguished only by their one-dimensional representation parameters. Along the way we presented important numerical techniques which allow the necessary order parameters to be computed directly (and without ambiguity) from an MPS representation of the ground states, including a method to recover single-site translation invariance in states generated using iTEBD. Within our phase diagram we have also identified two instances of continuous phase transitions between topologically nontrivial SPT phases. As the nature of SPT to SPT transitions has recently attracted increased interest, it would be desirably to study these transitions further, perhaps to identify the nature of the CFT associated with the transition, or to determine if the symmetry of the system remains discrete even at the critical point.

Finally, we have looked at the uniquely phenomenon of many-body localization. Using a combination of MPS techniques and exact diagonalization, we have studied the concurrence and the negativity as indicators of localization. In particular, we take the total amount of entanglement between nearest-neighbor pairs of sites (totaled over all such pairs) and find that both measures of entanglement show promise as methods for identifying localization. Furthermore, because they appear to show similar behavior, it is possible that the negativity could be employed to study localization in systems where the concurrence cannot be easily computed, such as in systems of higher spin. Previously, most studies of many-body localization effects have been confined to spin-1/2 systems, in part because higher spin systems are prohibitively difficult to compute numerically with exact diagonalization. But since this difficulty is substantially removed by the use of MPS methods, the evaluation of such systems with negativity is a very natural future project. Hence, the future outlook of this works ends almost back where this dissertation began: with the tantalizing potential of tensor network states to open up previously intractable problems.

Bibliography

[1] Werner Heisenberg. *Physics and philosophy: The revolution in modern science*. Harper & Brothers, 1958.

[2] Gerard 't Hooft. *In search of the ultimate building blocks*. Cambridge University Press, 1997.

[3] Erwin Schrödinger. Die gegenwärtige situation in der quantenmechanik. *Naturwissenschaften*, 23(49):823–828, 1935.

[4] Albert Einstein, Boris Podolsky, and Nathan Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical review*, 47(10):777, 1935.

[5] JS Bell. On the Einstein-Podolsky-Rosen paradox. *Physics*, 1:195–200, 1964.

[6] Lynden K Shalm, Evan Meyer-Scott, Bradley G Christensen, Peter Bierhorst, Michael A Wayne, Martin J Stevens, Thomas Gerrits, Scott Glancy, Deny R Hamel, Michael S Allman, et al. Strong loophole-free test of local realism. *Physical review letters*, 115(25):250402, 2015.

[7] Charles H Bennett and Stephen J Wiesner. Communication via one- and two-particle operators on Einstein-Podolsky-Rosen states. *Physical review letters*, 69(20):2881, 1992.

[8] Artur K Ekert. Quantum cryptography based on bell's theorem. *Physical review letters*, 67(6):661, 1991.

[9] Richard P Feynman. Simulating physics with computers. *International journal of theoretical physics*, 21(6):467–488, 1982.

[10] Robert Raussendorf and Hans J. Briegel. A one-way quantum computer. *Phys. Rev. Lett.*, 86:5188–5191, May 2001. doi: 10.1103/PhysRevLett.86.5188. URL http://link.aps.org/doi/10.1103/PhysRevLett.86.5188.

[11] Robert B Laughlin and David Pines. The theory of everything. *Proceedings of the National Academy of Sciences*, 97(1):28–31, 2000.

[12] Matthew B Hastings. An area law for one-dimensional quantum systems. *Journal of Statistical Mechanics: Theory and Experiment*, 2007 (08):P08024, 2007.

[13] Frank Verstraete and J Ignacio Cirac. Matrix product states represent ground states faithfully. *Physical Review B*, 73(9):094423, 2006.

[14] Frank Verstraete, Valentin Murg, and J Ignacio Cirac. Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems. *Advances in Physics*, 57(2): 143–224, 2008.

[15] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac. Matrix product state representations. *Quantum Info. Comput.*, 7(5):401–430, July 2007. ISSN 1533-7146. URL http://dl.acm.org/citation.cfm?id=2011832.2011833.

[16] Román Orús. A practical introduction to tensor networks: Matrix product states and projected entangled pair states. *Annals of Physics*, 349: 117–158, 2014.

[17] M. Fannes, B. Nachtergaele, and R. F. Werner. Finitely correlated states on quantum spin chains. *Communications in Mathematical Physics*, 144(3):443–490, 1992. URL http://projecteuclid.org/euclid.cmp/1104249404.

[18] A Klümper, A Schadschneider, and J Zittartz. Equivalence and solution of anisotropic spin-1 models and generalized tj fermion models in one dimension. *Journal of Physics A: Mathematical and General*, 24(16): L955, 1991.

[19] A Klümper, A Schadschneider, and J Zittartz. Matrix product ground states for one-dimensional spin-1 quantum antiferromagnets. *EPL (Europhysics Letters)*, 24(4):293, 1993.

[20] Stefan Rommer and Stellan Östlund. Class of ansatz wave functions for one-dimensional spin systems and their relation to the density matrix renormalization group. *Physical review b*, 55(4):2164, 1997.

[21] Steven R. White. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.*, 69:2863–2866, Nov 1992. doi:

10.1103/PhysRevLett.69.2863. URL http://link.aps.org/doi/10.1103/PhysRevLett.69.2863.

[22] Steven R White. Density-matrix algorithms for quantum renormalization groups. *Physical Review B*, 48(14):10345, 1993.

[23] Stellan Östlund and Stefan Rommer. Thermodynamic limit of density matrix renormalization. *Phys. Rev. Lett.*, 75:3537–3540, Nov 1995. doi: 10.1103/PhysRevLett.75.3537. URL http://link.aps.org/doi/10.1103/PhysRevLett.75.3537.

[24] T Nishino, T Hikihara, K Okunishi, and Y Hieida. Density matrix renormalization group: Introduction from a variational point of view. *International Journal of Modern Physics B*, 13(01):1–24, 1999.

[25] U. Schollwöck. The density-matrix renormalization group. *Rev. Mod. Phys.*, 77:259–315, Apr 2005. doi: 10.1103/RevModPhys.77.259. URL http://link.aps.org/doi/10.1103/RevModPhys.77.259.

[26] U. Schollwöck. The density-matrix renormalization group in the age of matrix product states. *Annals of Physics*, 326(1):96–192, 2011.

[27] Guifré Vidal. Efficient simulation of one-dimensional quantum many-body systems. *Phys. Rev. Lett.*, 93:040502, Jul 2004. doi: 10.1103/PhysRevLett.93.040502. URL http://link.aps.org/doi/10.1103/PhysRevLett.93.040502.

[28] G. Vidal. Classical simulation of infinite-size quantum lattice systems in one spatial dimension. *Phys. Rev. Lett.*, 98:070201, Feb 2007. doi: 10.1103/PhysRevLett.98.070201. URL http://link.aps.org/doi/10.1103/PhysRevLett.98.070201.

[29] Guifre Vidal. Entanglement renormalization. *Physical review letters*, 99 (22):220405, 2007.

[30] G. Evenbly and G. Vidal. Class of highly entangled many-body states that can be efficiently simulated. *Phys. Rev. Lett.*, 112:240502, Jun 2014. doi: 10.1103/PhysRevLett.112.240502. URL http://link.aps.org/doi/10.1103/PhysRevLett.112.240502.

[31] Frank Verstraete and J Ignacio Cirac. Renormalization algorithms for quantum-many body systems in two and higher dimensions. *arXiv preprint cond-mat/0407066*, 2004.

[32] Ian P McCulloch. Infinite size density matrix renormalization group, revisited. *arXiv preprint arXiv:0804.2509*, 2008.

[33] Jacob Jordan, Roman Orús, Guifre Vidal, Frank Verstraete, and J Ignacio Cirac. Classical simulation of infinite-size quantum lattice systems in two spatial dimensions. *Physical review letters*, 101(25):250602, 2008.

[34] David Perez-Garcia, Frank Verstraete, Michael M Wolf, and J Ignacio Cirac. Peps as unique ground states of local hamiltonians. *Quantum Information & Computation*, 8(6):650–663, 2008.

[35] Norbert Schuch, Ignacio Cirac, and David Pérez-García. Peps as ground states: Degeneracy and topology. *Annals of Physics*, 325(10):2153–2192, 2010.

[36] Luca Tagliacozzo, G Evenbly, and G Vidal. Simulation of two-dimensional quantum systems using a tree tensor network that exploits the entropic area law. *Physical Review B*, 80(23):235127, 2009.

[37] Y-Y Shi, L-M Duan, and Guifre Vidal. Classical simulation of quantum many-body systems with a tree tensor network. *Physical Review A*, 74 (2):022320, 2006.

[38] Lev Davidovich Landau. On the theory of phase transitions. i. *Zh. Eksp. Teor. Fiz.*, 11:19, 1937.

[39] Lev Davidovich Landau. On the theory of phase transitions. ii. *Zh. Eksp. Teor. Fiz.*, 11:627, 1937.

[40] Lev Davidovich Landau and Evgenii Mikhailovich Lifshitz. *Course of theoretical physics: Statistical Physics*, volume 5. Elsevier, 2013.

[41] Marco Cozzini, Radu Ionicioiu, and Paolo Zanardi. Quantum fidelity and quantum phase transitions in matrix product states. *Physical Review B*, 76(10):104420, 2007.

[42] M Asoudeh, V Karimipour, and A Sadrolashrafi. Quantum phase transitions and matrix product states in spin ladders. *Physical Review B*, 75 (22):224427, 2007.

[43] F Heidrich-Meisner, IP McCulloch, and AK Kolezhuk. Phase diagram of an anisotropic frustrated ferromagnetic spin-1 2 chain in a magnetic field: A density matrix renormalization group study. *Physical Review B*, 80(14):144417, 2009.

[44] Jonas A Kjäll, Michael P Zaletel, Roger SK Mong, Jens H Bardarson, and Frank Pollmann. Phase diagram of the anisotropic spin-2 xxz model: Infinite-system density matrix renormalization group study. *Physical Review B*, 87(23):235106, 2013.

[45] Kurt Binder. Finite size scaling analysis of ising model block distribution functions. *Zeitschrift für Physik B Condensed Matter*, 43(2):119–140, 1981.

[46] Zheng-Cheng Gu and Xiao-Gang Wen. Tensor-entanglement-filtering renormalization approach and symmetry-protected topological order. *Phys. Rev. B*, 80:155131, Oct 2009. doi: 10.1103/PhysRevB.80.155131. URL http://link.aps.org/doi/10.1103/PhysRevB.80.155131.

[47] Frank Pollmann, Ari M. Turner, Erez Berg, and Masaki Oshikawa. Entanglement spectrum of a topological phase in one dimension. *Phys. Rev. B*, 81:064439, Feb 2010. doi: 10.1103/PhysRevB.81.064439. URL http://link.aps.org/doi/10.1103/PhysRevB.81.064439.

[48] Frank Pollmann, Erez Berg, Ari M Turner, and Masaki Oshikawa. Symmetry protection of topological phases in one-dimensional quantum spin systems. *Phys. Rev. B*, 85(7):075125, 2012.

[49] Xie Chen, Zheng-Cheng Gu, and Xiao-Gang Wen. Classification of gapped symmetric phases in one-dimensional spin systems. *Phys. Rev. B*, 83:035107, Jan 2011. doi: 10.1103/PhysRevB.83.035107. URL http://link.aps.org/doi/10.1103/PhysRevB.83.035107.

[50] Xie Chen, Zheng-Cheng Gu, and Xiao-Gang Wen. Complete classification of one-dimensional gapped quantum phases in interacting spin systems. *Phys. Rev. B*, 84:235128, Dec 2011. doi: 10.1103/PhysRevB.84.235128. URL http://link.aps.org/doi/10.1103/PhysRevB.84.235128.

[51] Lukasz Fidkowski and Alexei Kitaev. Topological phases of fermions in one dimension. *Phys. Rev. B*, 83:075103, Feb 2011. doi: 10.1103/PhysRevB.83.075103. URL http://link.aps.org/doi/10.1103/PhysRevB.83.075103.

[52] Frank Pollmann and Ari M. Turner. Detection of symmetry-protected topological phases in one dimension. *Phys. Rev. B*, 86:125441, Sep 2012. doi: 10.1103/PhysRevB.86.125441. URL http://link.aps.org/doi/10.1103/PhysRevB.86.125441.

[53] Norbert Schuch, David Pérez-García, and Ignacio Cirac. Classifying quantum phases using matrix product states and projected entangled pair states. *Phys. Rev. B*, 84:165139, Oct 2011. doi: 10.1103/PhysRevB.84.165139. URL http://link.aps.org/doi/10.1103/PhysRevB.84.165139.

[54] Arijeet Pal and David A Huse. Many-body localization phase transition. *Physical review b*, 82(17):174411, 2010.

[55] Rahul Nandkishore and David A Huse. Many body localization and thermalization in quantum statistical mechanics. *arXiv preprint arXiv:1404.0686*, 2014.

[56] IV Gornyi, AD Mirlin, and DG Polyakov. Interacting electrons in disordered wires: Anderson localization and low-t transport. *Physical review letters*, 95(20):206603, 2005.

[57] Vadim Oganesyan, Arijeet Pal, and David A Huse. Energy transport in disordered classical spin chains. *Physical Review B*, 80(11):115104, 2009.

[58] IL Aleiner, BL Altshuler, and GV Shlyapnikov. A finite-temperature phase transition for disordered weakly interacting bosons in one dimension. *Nature Physics*, 6(11):900–904, 2010.

[59] Xiongjie Yu, David Pekker, and Bryan K Clark. Finding matrix product state representations of highly-excited eigenstates of many-body localized hamiltonians. *arXiv preprint arXiv:1509.01244*, 2015.

[60] Vedika Khemani, Frank Pollmann, and SL Sondhi. Obtaining highly-excited eigenstates of many-body localized hamiltonians by the density matrix renormalization group. *arXiv preprint arXiv:1509.00483*, 2015.

[61] DM Kennes and C Karrasch. Entanglement scaling of excited states in large one-dimensional many-body localized systems. *arXiv preprint arXiv:1511.02205*, 2015.

[62] SP Lim and DN Sheng. Nature of many-body localization and transitions by density matrix renormaliztion group and exact diagonalization studies. *arXiv preprint arXiv:1510.08145*, 2015.

[63] Scott Hill and William K Wootters. Entanglement of a pair of quantum bits. *Physical review letters*, 78(26):5022, 1997.

[64] G. Vidal and R. F. Werner. Computable measure of entanglement. *Phys. Rev. A*, 65:032314, Feb 2002. doi: 10.1103/PhysRevA.65.032314. URL http://link.aps.org/doi/10.1103/PhysRevA.65.032314.

[65] Soumya Bera and Arul Lakshminarayan. Local entanglement structure across a many-body localization transition. *arXiv preprint arXiv:1512.04705*, 2015.

[66] Frank Verstraete, Koenraad Audenaert, Jeroen Dehaene, and Bart De Moor. ami. *Journal of Physics A: Mathematical and General*, 34 (47):10327, 2001.

[67] Roger Penrose. Applications of negative dimensional tensors. *Combinatorial mathematics and its applications*, 221244, 1971.

[68] Christopher J Wood, Jacob D Biamonte, and David G Cory. Tensor networks and graphical calculus for open quantum systems. *Quantum Information & Computation*, 15(9&10):759–811, 2015.

[69] Robert NC Pfeifer, Jutho Haegeman, and Frank Verstraete. Faster identification of optimal contraction sequences for tensor networks. *Phys. Rev. E*, 90:033315, Sep 2014. doi: 10.1103/PhysRevE.90.033315. URL http://link.aps.org/doi/10.1103/PhysRevE.90.033315.

[70] Robert NC Pfeifer, Glen Evenbly, Sukhwinder Singh, and Guifre Vidal. Ncon: A tensor network contractor for matlab. *arXiv preprint arXiv:1402.0939*, 2014.

[71] J Ignacio Cirac and Frank Verstraete. Renormalization and tensor product states in spin chains and lattices. *Journal of Physics A: Mathematical and Theoretical*, 42(50):504004, 2009.

[72] Jacob C Bridgeman and Christopher T Chubb. Hand-waving and interpretive dance: An introductory course on tensor networks. *arXiv preprint arXiv:1603.03039*, 2016.

[73] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[74] Anirban Pathak. *Elements of quantum computation and quantum communication*. Taylor & Francis, 2013.

[75] Gilbert W Stewart. On the early history of the singular value decomposition. *SIAM review*, 35(4):551–566, 1993.

[76] Carl Eckart and Gale Young. A principal axis transformation for non-hermitian matrices. *Bulletin of the American Mathematical Society*, 45 (2):118–121, 1939.

[77] Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[78] Leon Mirsky. Symmetric gauge functions and unitarily invariant norms. *The quarterly journal of mathematics*, 11(1):50–59, 1960.

[79] Jens Eisert, Marcus Cramer, and Martin B Plenio. Colloquium: Area laws for the entanglement entropy. *Reviews of Modern Physics*, 82(1): 277, 2010.

[80] Matthew B Hastings. Solving gapped hamiltonians locally. *Physical Review B*, 73(8):085115, 2006.

[81] Guifré Vidal. Efficient classical simulation of slightly entangled quantum computations. *Phys. Rev. Lett.*, 91:147902, Oct 2003. doi: 10.1103/PhysRevLett.91.147902. URL http://link.aps.org/doi/10.1103/PhysRevLett.91.147902.

[82] Maria Eckholt Perotti. *Matrix Product formalisms*. PhD thesis, Technische Universität München München, 2005.

[83] R. Orús and G. Vidal. Infinite time-evolving block decimation algorithm beyond unitary evolution. *Phys. Rev. B*, 78:155117, Oct 2008. doi: 10.1103/PhysRevB.78.155117. URL http://link.aps.org/doi/10.1103/PhysRevB.78.155117.

[84] Ho N. Phien, Ian P. McCulloch, and Guifré Vidal. Fast convergence of imaginary time evolution tensor network algorithms by recycling the environment. *Phys. Rev. B*, 91:115137, Mar 2015. doi: 10.1103/PhysRevB.91.115137. URL http://link.aps.org/doi/10.1103/PhysRevB.91.115137.

[85] Johannes Schachenmayer. Dynamics and long-range interactions in 1d quantum systems. Master's thesis, Technischen Universität, Muünchen, November 2008.

[86] Frank Verstraete, Juan J Garcia-Ripoll, and Juan Ignacio Cirac. Matrix product density operators: simulation of finite-temperature and dissipative systems. *Physical review letters*, 93(20):207204, 2004.

[87] Ian P McCulloch. From density-matrix renormalization group to matrix product states. *Journal of Statistical Mechanics: Theory and Experiment*, 2007(10):P10014, 2007.

[88] Frank Verstraete, Michael M Wolf, David Perez-Garcia, and J Ignacio Cirac. Criticality, the area law, and the computational power of projected entangled pair states. *Physical review letters*, 96(22):220601, 2006.

[89] Michael M Wolf, Frank Verstraete, Matthew B Hastings, and J Ignacio Cirac. Area laws in quantum systems: mutual information and correlations. *Physical review letters*, 100(7):070502, 2008.

[90] Robert Raussendorf, Daniel E. Browne, and Hans J. Briegel. Measurement-based quantum computation on cluster states. *Phys. Rev. A*, 68:022312, Aug 2003. doi: 10.1103/PhysRevA.68.022312. URL http://link.aps.org/doi/10.1103/PhysRevA.68.022312.

[91] Michael A Nielsen. Cluster-state quantum computation. *Reports on Mathematical Physics*, 57(1):147–161, 2006.

[92] Ian Affleck, Tom Kennedy, Elliott H. Lieb, and Hal Tasaki. Rigorous results on valence-bond ground states in antiferromagnets. *Phys. Rev. Lett.*, 59:799–802, Aug 1987. doi: 10.1103/PhysRevLett.59.799. URL http://link.aps.org/doi/10.1103/PhysRevLett.59.799.

[93] Tzu-Chieh Wei, Ian Affleck, and Robert Raussendorf. Affleck-kennedy-lieb-tasaki state on a honeycomb lattice is a universal quantum computational resource. *Physical review letters*, 106(7):070501, 2011.

[94] Tzu-Chieh Wei, Ian Affleck, and Robert Raussendorf. Two-dimensional affleck-kennedy-lieb-tasaki state on the honeycomb lattice is a universal resource for quantum computation. *Phys. Rev. A*, 86:032328, Sep 2012. doi: 10.1103/PhysRevA.86.032328. URL http://link.aps.org/doi/10.1103/PhysRevA.86.032328.

[95] A.Yu. Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2 – 30, 2003. ISSN 0003-4916. doi: http://dx.doi.org/10.1016/S0003-4916(02)00018-0. URL http://www.sciencedirect.com/science/article/pii/S0003491602000180.

[96] Norbert Schuch, Michael M Wolf, Frank Verstraete, and J Ignacio Cirac. Computational complexity of projected entangled pair states. *Physical review letters*, 98(14):140506, 2007.

[97] Michael Levin and Cody P Nave. Tensor renormalization group approach to two-dimensional classical lattice models. *Physical Review Letters*, 99 (12):120601, 2007.

[98] Román Orús and Guifré Vidal. Simulation of two-dimensional quantum systems on an infinite lattice revisited: Corner transfer matrix for tensor contraction. *Physical Review B*, 80(9):094403, 2009.

[99] Artur García-Sáez and José I Latorre. Renormalization group contraction of tensor networks in three dimensions. *Physical Review B*, 87(8): 085130, 2013.

[100] ZY Xie, Jing Chen, MP Qin, JW Zhu, LP Yang, and Tao Xiang. Coarse-graining renormalization by higher-order singular value decomposition. *Physical Review B*, 86(4):045139, 2012.

[101] Tomotoshi Nishino and Kouichi Okunishi. Corner transfer matrix renormalization group method. *Journal of the Physical Society of Japan*, 65 (4):891–894, 1996.

[102] Kenneth G Wilson and John Kogut. The renormalization group and the ϵ expansion. *Physics Reports*, 12(2):75–199, 1974.

[103] Kenneth G Wilson. The renormalization group: Critical phenomena and the kondo problem. *Reviews of Modern Physics*, 47(4):773, 1975.

[104] Jorge Dukelsky, Miguel A Martín-Delgado, Tomotoshi Nishino, and Germán Sierra. Equivalence of the variational matrix product method and the density matrix renormalization group applied to spin chains. *EPL (Europhysics Letters)*, 43(4):457, 1998.

[105] Frank Verstraete, Diego Porras, and J Ignacio Cirac. Density matrix renormalization group and periodic boundary conditions: a quantum information perspective. *Physical review letters*, 93(22):227205, 2004.

[106] Stephen Boyd and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

[107] J Eisert. Computational difficulty of global variations in the density matrix renormalization group. *Physical review letters*, 97(26):260501, 2006.

[108] Örs Legeza and Gábor Fáth. Accuracy of the density-matrix renormalization-group method. *Physical Review B*, 53(21):14349, 1996.

[109] Ivan Oseledets. Dmrg approach to fast linear algebra in the tt-format. *Computational Methods in Applied Mathematics Comput. Methods Appl. Math.*, 11(3):382–393, 2011.

[110] Zhaojun Bai, James Demmel, Jack Dongarra, Axel Ruhe, and Henk van der Vorst. *Templates for the solution of algebraic eigenvalue problems: a practical guide*, volume 11. Siam, 2000.

[111] Gregory M. Crosswhite, A. C. Doherty, and Guifré Vidal. Applying matrix product operators to model systems with long-range interactions. *Phys. Rev. B*, 78:035116, Jul 2008. doi: 10.1103/PhysRevB.78.035116. URL http://link.aps.org/doi/10.1103/PhysRevB.78.035116.

[112] Robert NC Pfeifer. Symmetry-protected local minima in infinite dmrg. *Physical Review B*, 92(20):205127, 2015.

[113] D. Pérez-García, M. M. Wolf, M. Sanz, F. Verstraete, and J. I. Cirac. String order and symmetries in quantum spin lattices. *Phys. Rev. Lett.*, 100:167202, Apr 2008. doi: 10.1103/PhysRevLett.100.167202. URL http://link.aps.org/doi/10.1103/PhysRevLett.100.167202.

[114] Sara Bergkvist, Ian P McCulloch, and Anders Rosengren. Spinful bosons in an optical lattice. *Physical Review A*, 74(5):053419, 2006.

[115] Juan José García-Ripoll. Time evolution of matrix product states. *New Journal of Physics*, 8(12):305, 2006.

[116] Hale F Trotter. On the product of semi-groups of operators. *Proceedings of the American Mathematical Society*, 10(4):545–551, 1959.

[117] Masuo Suzuki. Generalized Trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems. *Communications in Mathematical Physics*, 51 (2):183–190, 1976.

[118] Masuo Suzuki. General theory of higher-order decomposition of exponential operators and symplectic integrators. *Physics Letters A*, 165(5): 387–395, 1992.

[119] Michael Zwolak and Guifré Vidal. Mixed-state dynamics in one-dimensional quantum lattice systems: a time-dependent superoperator renormalization algorithm. *Physical review letters*, 93(20):207205, 2004.

[120] Frank Verstraete, Juan J Garcia-Ripoll, and Juan Ignacio Cirac. Matrix product density operators: simulation of finite-temperature and dissipative systems. *Physical review letters*, 93(20):207204, 2004.

[121] W Janke and T Sauer. Properties of higher-order Trotter formulas. *Physics Letters A*, 165(3):199–205, 1992.

[122] RM Fye. New results on Trotter-like approximations. *Physical Review B*, 33(9):6271, 1986.

[123] IP Omelyan, IM Mryglod, and Reinhard Folk. Optimized forest–ruth- and suzuki-like algorithms for integration of motion in many-body systems. *Computer Physics Communications*, 146(2):188–202, 2002.

[124] Naomichi Hatano and Masuo Suzuki. Finding exponential product formulas of higher orders. In *Quantum annealing and other optimization methods*, pages 37–68. Springer, 2005.

[125] Adolfo Avella and Ferdinando Mancini. *Strongly correlated systems: numerical methods*, volume 176. Springer Science & Business Media, 2013.

[126] HC Jiang, ZY Weng, and T Xiang. Accurate determination of tensor network state of quantum lattice models in two dimensions. *Physical review letters*, 101(9):090603, 2008.

[127] Wei Li, Jan von Delft, and Tao Xiang. Efficient simulation of infinite tree tensor network states on the bethe lattice. *Physical Review B*, 86 (19):195137, 2012.

[128] Ling Wang and Frank Verstraete. Cluster update for tensor network states. *arXiv preprint arXiv:1110.4362*, 2011.

[129] Ho N. Phien, Johann A. Bengua, Hoang D. Tuan, Philippe Corboz, and Román Orús. Infinite projected entangled pair states algorithm improved: Fast full update and gauge fixing. *Phys. Rev. B*, 92:035142, Jul 2015. doi: 10.1103/PhysRevB.92.035142. URL http://link.aps.org/doi/10.1103/PhysRevB.92.035142.

[130] Subir Sachdev. Quantum phase transitions. *Physics world*, 12(4):33, 1999.

[131] H Hong, H Park, and LH Tang. Anomalous Binder cumulant and lack of self-averageness in systems with quenched disorder. *J. Korean Phys. Soc.*, 49(cond-mat/0611509):L1885, 2006.

[132] RN Bhatt and AP Young. Search for a transition in the three-dimensional j ising spin-glass. *Physical review letters*, 54(9):924, 1985.

[133] N Kawashima and AP Young. Phase transition in the three-dimensional±j ising spin glass. *Physical Review B*, 53(2):R484, 1996.

[134] Kurt Kremer and Kurt Binder. Monte carlo simulation of lattice models for macromolecules. *Computer Physics Reports*, 7(6):259–310, 1988.

[135] W Selke. Critical Binder cumulant of two-dimensional ising models. *The European Physical Journal B-Condensed Matter and Complex Systems*, 51(2):223–228, 2006.

[136] W Selke. The critical Binder cumulant for isotropic ising models on square and triangular lattices. *Journal of Statistical Mechanics: Theory and Experiment*, 2007(04):P04008, 2007.

[137] W Selke and LN Shchur. Critical Binder cumulant in a two-dimensional anisotropic ising model with competing interactions. *Physical Review E*, 80(4):042104, 2009.

[138] Anders W. Sandvik. Evidence for deconfined quantum criticality in a two-dimensional heisenberg model with four-spin interactions. *Phys. Rev. Lett.*, 98:227202, Jun 2007. doi: 10.1103/PhysRevLett.98.227202. URL http://link.aps.org/doi/10.1103/PhysRevLett.98.227202.

[139] H. P. Ying, H. J. Luo, L. Schülke, and B. Zheng. Dynamic monte carlo study of the two-dimensional quantum xy model. *Modern Physics Letters B*, 12(29n30):1237–1243, 1998. doi: 10.1142/S0217984998001463. URL http://www.worldscientific.com/doi/abs/10.1142/S0217984998001463.

[140] Jae-Kwon Kim and Adrian Patrascioiu. Critical behavior of the specific heat in the two dimensional site diluted ising system. *Phys. Rev. Lett.*, 72:2785–2788, Apr 1994. doi: 10.1103/PhysRevLett.72.2785. URL http://link.aps.org/doi/10.1103/PhysRevLett.72.2785.

[141] H. G. Ballesteros, A. Cruz, L. A. Fernández, V. Martín-Mayor, J. Pech, J. J. Ruiz-Lorenzo, A. Tarancón, P. Téllez, C. L. Ullod, and C. Ungil. Critical behavior of the three-dimensional ising spin glass. *Phys. Rev. B*, 62:14237–14245, Dec 2000. doi: 10.1103/PhysRevB.62.14237. URL http://link.aps.org/doi/10.1103/PhysRevB.62.14237.

[142] Guillaume Grégoire and Hugues Chaté. Onset of collective and cohesive motion. *Physical review letters*, 92(2):025702, 2004.

[143] A Weichselbaum, F Verstraete, U Schollwöck, J Ignacio Cirac, and Jan von Delft. Variational matrix-product-state approach to quantum impurity models. *Physical Review B*, 80(16):165117, 2009.

[144] Fei Zhan, Jacopo Sabbatini, Matthew J Davis, and Ian P McCulloch. Miscible-immiscible quantum phase transition in coupled two-component Bose-Einstein condensates in one-dimensional optical lattices. *Physical Review A*, 90(2):023630, 2014.

[145] SN Saadatmand, BJ Powell, and IP McCulloch. Phase diagram of the spin-1/2 triangular j_1-j_2 heisenberg model on a 3-leg cylinder. *arXiv preprint arXiv:1503.07570*, 2015.

[146] Maurice G Kendall and Alan Stuart. Moments and cumulants. *The advanced theory of statistics*, 1:57–96, 1977.

[147] Peter J Smith. A recursive formulation of the old problem of obtaining moments from cumulants and vice versa. *The American Statistician*, 49 (2):217–218, 1995.

[148] Hangmo Yi. Quantum phase transition of the transverse-field quantum ising model on scale-free networks. *Phys. Rev. E*, 91:012146, Jan 2015. doi: 10.1103/PhysRevE.91.012146. URL http://link.aps.org/doi/10.1103/PhysRevE.91.012146.

[149] Colin G West, Artur Garcia-Saez, and Tzu-Chieh Wei. Efficient evaluation of high-order moments and cumulants in tensor network states. *Physical Review B*, 92(11):115103, 2015.

[150] Eugene Lukacs. *Characteristic functions*, volume 5. Griffin London, 1970.

[151] Walter Rudin. *Real and complex analysis*. Tata McGraw-Hill Education, 1987.

[152] H Neudecker. A note on kronecker matrix products and matrix equation systems. *SIAM Journal on Applied Mathematics*, 17(3):603–606, 1969.

[153] Bengt Fornberg. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of computation*, 51(184):699–706, 1988.

[154] F. Pollmann. private communication.

[155] Volckmar Nebendahl. private communication.

[156] L Michel and IP McCulloch. Schur forms of matrix product operators in the infinite limit. *arXiv preprint arXiv:1008.4667*, 2010.

[157] Ho N Phien, Guifré Vidal, and Ian P McCulloch. Infinite boundary conditions for matrix product state calculations. *Physical Review B*, 86 (24):245107, 2012.

[158] Pierre Pfeuty. The one-dimensional ising model with a transverse field. *ANNALS of Physics*, 57(1):79–90, 1970.

[159] Amit Dutta, Uma Divakaran, Diptiman Sen, Bikas K Chakrabarti, Thomas F Rosenbaum, and Gabriel Aeppli. Transverse field spin models: From statistical physics to quantum information. *ChemInform*, 42 (18), 2011.

[160] Roland Bulirsch and Josef Stoer. Numerical treatment of ordinary differential equations by extrapolation methods. *Numerische Mathematik*, 8(1):1–13, 1966.

[161] Malte Henkel and Gunter Schutz. Finite-lattice extrapolation algorithms. *Journal of Physics A: Mathematical and General*, 21(11):2617, 1988.

[162] Ian P McCulloch. private communication.

[163] Malte Henkel. *Conformal invariance and critical phenomena*. Springer Science & Business Media, 1999.

[164] T Kaneyoshi, M Jaščur, and IP Fittipaldi. Transverse ising model with arbitrary spin. *Physical Review B*, 48(1):250, 1993.

[165] Chen Liu, Ling Wang, Anders W. Sandvik, Yu-Cheng Su, and Ying-Jer Kao. Symmetry breaking and criticality in tensor-product states. *Phys. Rev. B*, 82:060410, Aug 2010. doi: 10.1103/PhysRevB.82.060410. URL http://link.aps.org/doi/10.1103/PhysRevB.82.060410.

[166] J. Oitmaa and A. M. A. von Brasch. Spin-1 ising model in a transverse crystal field. *Phys. Rev. B*, 67:172402, May 2003. doi: 10.1103/PhysRevB.67.172402. URL http://link.aps.org/doi/10.1103/PhysRevB.67.172402.

[167] Henk WJ Blöte and Youjin Deng. Cluster monte carlo simulation of the transverse ising model. *Physical Review E*, 66(6):066110, 2002.

[168] Lev Davidovich Landau and VL Ginzburg. On the theory of superconductivity. *Zh. Eksp. Teor. Fiz.*, 20:1064, 1950.

[169] Wen Xiao Gang. *Quantum field theory of many-body systems: from the origin of sound to an origin of light and electrons*. Oxford University Press, Oxford, 2007. URL https://cds.cern.ch/record/803748.

[170] D. C. Tsui, H. L. Stormer, and A. C. Gossard. Two-dimensional magnetotransport in the extreme quantum limit. *Phys. Rev. Lett.*, 48:1559–1562, May 1982. doi: 10.1103/PhysRevLett.48.1559. URL http://link.aps.org/doi/10.1103/PhysRevLett.48.1559.

[171] R. B. Laughlin. Quantized hall conductivity in two dimensions. *Phys. Rev. B*, 23:5632–5633, May 1981. doi: 10.1103/PhysRevB.23.5632. URL http://link.aps.org/doi/10.1103/PhysRevB.23.5632.

[172] R. B. Laughlin. Anomalous quantum hall effect: An incompressible quantum fluid with fractionally charged excitations. *Phys. Rev. Lett.*, 50: 1395–1398, May 1983. doi: 10.1103/PhysRevLett.50.1395. URL http://link.aps.org/doi/10.1103/PhysRevLett.50.1395.

[173] V. Kalmeyer and R. B. Laughlin. Equivalence of the resonating-valence-bond and fractional quantum hall states. *Phys. Rev. Lett.*, 59:2095–2098, Nov 1987. doi: 10.1103/PhysRevLett.59.2095. URL http://link.aps.org/doi/10.1103/PhysRevLett.59.2095.

[174] Xiao-Gang Wen, Frank Wilczek, and Anthony Zee. Chiral spin states and superconductivity. *Physical Review B*, 39(16):11413, 1989.

[175] Michael A. Levin and Xiao-Gang Wen. String-net condensation: A physical mechanism for topological phases. *Phys. Rev. B*, 71:045110, Jan 2005. doi: 10.1103/PhysRevB.71.045110. URL http://link.aps.org/doi/10.1103/PhysRevB.71.045110.

[176] Xie Chen, Zheng-Cheng Gu, and Xiao-Gang Wen. Local unitary transformation, long-range quantum entanglement, wave function renormalization, and topological order. *Physical review b*, 82(15):155138, 2010.

[177] Frank Verstraete, J Ignacio Cirac, JI Latorre, E Rico, and Michael M Wolf. Renormalization-group transformations on quantum states. *Physical review letters*, 94(14):140601, 2005.

[178] F. D. M. Haldane. Nonlinear field theory of large-spin heisenberg antiferromagnets: Semiclassically quantized solitons of the one-dimensional

easy-axis néel state. *Phys. Rev. Lett.*, 50:1153–1156, Apr 1983. doi: 10.1103/PhysRevLett.50.1153. URL http://link.aps.org/doi/10.1103/PhysRevLett.50.1153.

[179] F.D.M. Haldane. Continuum dynamics of the 1-d heisenberg antiferromagnet: Identification with the o(3) nonlinear sigma model. *Physics Letters A*, 93(9):464 – 468, 1983. ISSN 0375-9601. doi: http://dx.doi.org/10.1016/0375-9601(83)90631-X. URL http://www.sciencedirect.com/science/article/pii/037596018390631X.

[180] Abhishodh Prakash and Tzu-Chieh Wei. Ground states of one-dimensional symmetry-protected topological phases and their utility as resource states for quantum computation. *Phys. Rev. A*, 92:022310, Aug 2015. doi: 10.1103/PhysRevA.92.022310. URL http://link.aps.org/doi/10.1103/PhysRevA.92.022310.

[181] Gavin K. Brennen and Akimasa Miyake. Measurement-based quantum computer in the gapped ground state of a two-body hamiltonian. *Phys. Rev. Lett.*, 101:010502, Jul 2008. doi: 10.1103/PhysRevLett.101.010502. URL http://link.aps.org/doi/10.1103/PhysRevLett.101.010502.

[182] Abhishodh Prakash, Colin G West, and Tzu-Chieh Wei. Detection of gapped phases of a 1d spin chain with onsite and spatial symmetries. *arXiv preprint arXiv:1604.00037*, 2016.

[183] Ji-Yao Chen and Zheng-Xin Liu. Symmetry protected topological phases in spin-1 ladders and their phase transitions. *Annals of Physics*, 362:551–567, 2015.

[184] Lokman Tsui, Hong-Chen Jiang, Yuan-Ming Lu, and Dung-Hai Lee. Quantum phase transitions between a class of symmetry protected topological states. *Nuclear Physics B*, 896:330–359, 2015.

[185] Lokman Tsui, Fa Wang, and Dung-Hai Lee. Topological versus landau-like phase transitions. *arXiv preprint arXiv:1511.07460*, 2015.

[186] Hiroshi Ueda and Shigeki Onoda. Symmetry-protected topological phases and transition in a frustrated spin-1/2 xxz chain. *arXiv preprint arXiv:1406.5382*, 2014.

[187] Pasquale Calabrese and John Cardy. Entanglement entropy and quantum field theory. *Journal of Statistical Mechanics: Theory and Experiment*, 2004(06):P06002, 2004.

[188] Frank Pollmann, Subroto Mukerjee, Ari M. Turner, and Joel E. Moore. Theory of finite-entanglement scaling at one-dimensional quantum critical points. *Phys. Rev. Lett.*, 102:255701, Jun 2009. doi: 10.1103/PhysRevLett.102.255701. URL http://link.aps.org/doi/10.1103/PhysRevLett.102.255701.

[189] Itai Arad, Zeph Landau, and Umesh Vazirani. Improved one-dimensional area law for frustration-free systems. *Phys. Rev. B*, 85:195145, May 2012. doi: 10.1103/PhysRevB.85.195145. URL http://link.aps.org/doi/10.1103/PhysRevB.85.195145.

[190] Xiao-Gang Wen. Quantum orders and symmetric spin liquids. *Physical Review B*, 65(16):165113, 2002.

[191] Sukhwinder Singh. Identifying quantum phases from the injectivity of symmetric matrix product states. *Phys. Rev. B*, 91:115145, Mar 2015. doi: 10.1103/PhysRevB.91.115145. URL http://link.aps.org/doi/10.1103/PhysRevB.91.115145.

[192] Cornelius Lanczos. *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators.* United States Governm. Press Office Los Angeles, CA, 1950.

[193] Daniela Calvetti, L Reichel, and Danny Chris Sorensen. An implicitly restarted lanczos method for large symmetric eigenvalue problems. *Electronic Transactions on Numerical Analysis*, 2(1):21, 1994.

[194] Danny C Sorensen. *Implicitly restarted Arnoldi/Lanczos methods for large scale eigenvalue calculations.* Springer, 1997.

[195] Xie Chen, Zheng-Xin Liu, and Xiao-Gang Wen. Two-dimensional symmetry-protected topological orders and their protected gapless edge excitations. *Phys. Rev. B*, 84:235141, Dec 2011. doi: 10.1103/PhysRevB.84.235141. URL http://link.aps.org/doi/10.1103/PhysRevB.84.235141.

[196] David E Evans and Raphael Høegh-Krohn. Spectral properties of positive maps on c*-algebras. *Journal of the London Mathematical Society*, 2(2):345–355, 1978.

[197] Hong-Lei Wang, Yan-Wei Dai, Bing-Quan Hu, and Huan-Qiang Zhou. Bifurcation in ground-state fidelity for a one-dimensional spin model with competing two-spin and three-spin interactions. *Physics Letters A*, 375(45):4045–4048, 2011.

[198] Linda E Reichl and Ilya Prigogine. *A modern course in statistical physics*, volume 71. University of Texas press Austin, 1980.

[199] Mehran Kardar. *Statistical physics of particles*. Cambridge University Press, 2007.

[200] Philip W Anderson. Absence of diffusion in certain random lattices. *Physical review*, 109(5):1492, 1958.

[201] DM Basko, LI Aleiner, and BL Altshuler. On the problem of many-body localization. *Problems of Condensed Matter Physics*, pages 50–70, 2006.

[202] MS Skolnick, JM Rorison, KJ Nash, DJ Mowbray, PR Tapster, SJ Bass, and AD Pitt. Observation of a many-body edge singularity in quantum-well luminescence spectra. *Physical review letters*, 58(20):2130, 1987.

[203] Michael Schreiber, Sean S Hodgman, Pranjal Bordia, Henrik P Lüschen, Mark H Fischer, Ronen Vosk, Ehud Altman, Ulrich Schneider, and Immanuel Bloch. Observation of many-body localization of interacting fermions in a quasirandom optical lattice. *Science*, 349(6250):842–845, 2015.

[204] William K Wootters. Entanglement of formation of an arbitrary state of two qubits. *Physical Review Letters*, 80(10):2245, 1998.

[205] Armin Uhlmann. Fidelity and concurrence of conjugated states. *Physical Review A*, 62(3):032307, 2000.

[206] Pranaw Rungta, V. Bužek, Carlton M. Caves, M. Hillery, and G. J. Milburn. Universal state inversion and concurrence in arbitrary dimensions. *Phys. Rev. A*, 64:042315, Sep 2001. doi: 10.1103/PhysRevA.64.042315. URL http://link.aps.org/doi/10.1103/PhysRevA.64.042315.

[207] William K Wootters. Entanglement of formation and concurrence. *Quantum Information & Computation*, 1(1):27–44, 2001.

[208] JM Deutsch. Quantum statistical mechanics in a closed system. *Physical Review A*, 43(4):2046, 1991.

[209] Mark Srednicki. Chaos and quantum thermalization. *Physical Review E*, 50(2):888, 1994.

[210] Mark Srednicki. The approach to thermal equilibrium in quantized chaotic systems. *Journal of Physics A: Mathematical and General*, 32 (7):1163, 1999.

[211] Marcos Rigol, Vanja Dunjko, and Maxim Olshanii. Thermalization and its mechanism for generic isolated quantum systems. *Nature*, 452(7189): 854–858, 2008.

[212] Hal Tasaki. From quantum dynamics to the canonical distribution: general picture and a rigorous example. *Physical review letters*, 80(7):1373, 1998.

[213] Shankar Iyer, Vadim Oganesyan, Gil Refael, and David A. Huse. Many-body localization in a quasiperiodic system. *Phys. Rev. B*, 87:134202, Apr 2013. doi: 10.1103/PhysRevB.87.134202. URL http://link.aps.org/doi/10.1103/PhysRevB.87.134202.

[214] Xiaopeng Li, Sriram Ganeshan, JH Pixley, and S Das Sarma. Many-body localization and quantum nonergodicity in a model with a single-particle mobility edge. *Physical Review Letters*, 115(18):186601, 2015.

[215] D. Huse. private communication.

[216] David A Huse, Rahul Nandkishore, and Vadim Oganesyan. Phenomenology of fully many-body-localized systems. *Physical Review B*, 90(17): 174202, 2014.

[217] Anushya Chandran, Isaac H Kim, Guifre Vidal, and Dmitry A Abanin. Constructing local integrals of motion in the many-body localized phase. *Physical Review B*, 91(8):085425, 2015.

[218] Pavan Hosur and Xiao-Liang Qi. Characterizing eigenstate thermalization via measures in the fock space of operators. *arXiv preprint arXiv:1507.04003*, 2015.

[219] James R Garrison and Tarun Grover. Does a single eigenstate encode the full hamiltonian? *arXiv preprint arXiv:1503.00729*, 2015.

[220] David J Luitz, Nicolas Laflorencie, and Fabien Alet. Many-body localization edge in the random-field heisenberg chain. *Physical Review B (Condensed Matter)*, 91:081103, 2015.

[221] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[222] Vadim Oganesyan and David A. Huse. Localization of interacting fermions at high temperature. *Phys. Rev. B*, 75:155111, Apr 2007. doi: 10.1103/PhysRevB.75.155111. URL http://link.aps.org/doi/10.1103/PhysRevB.75.155111.

[223] Y. Y. Atas, E. Bogomolny, O. Giraud, and G. Roux. Distribution of the ratio of consecutive level spacings in random matrix ensembles. *Phys. Rev. Lett.*, 110:084101, Feb 2013. doi: 10.1103/PhysRevLett.110.084101. URL http://link.aps.org/doi/10.1103/PhysRevLett.110.084101.

[224] RJ Bell and P Dean. Atomic vibrations in vitreous silica. *Discussions of the Faraday society*, 50:55–61, 1970.

[225] RJ Bell. The dynamics of disordered lattices. *Reports on Progress in Physics*, 35(3):1315, 1972.

[226] David J. Luitz, Fabien Alet, and Nicolas Laflorencie. Universal behavior beyond multifractality in quantum many-body systems. *Phys. Rev. Lett.*, 112:057203, Feb 2014. doi: 10.1103/PhysRevLett.112.057203. URL http://link.aps.org/doi/10.1103/PhysRevLett.112.057203.

[227] Brian Swingle. A simple model of many-body localization. *arXiv preprint arXiv:1307.0507*, 2013.

[228] Bela Bauer and Chetan Nayak. Area laws in a many-body localized state and its implications for topological order. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(09):P09005, 2013.

[229] M Friesdorf, AH Werner, W Brown, VB Scholz, and J Eisert. Many-body localization implies that eigenvectors are matrix-product states. *Physical review letters*, 114(17):170505, 2015.

[230] Jonas A Kjäll, Jens H Bardarson, and Frank Pollmann. Many-body localization in a disordered quantum ising chain. *Physical review letters*, 113(10):107204, 2014.

[231] Jens H Bardarson, Frank Pollmann, and Joel E Moore. Unbounded growth of entanglement in models of many-body localization. *Physical review letters*, 109(1):017202, 2012.

[232] Arun Nanduri, Hyungwon Kim, and David A. Huse. Entanglement spreading in a many-body localized system. *Phys. Rev. B*, 90:064201, Aug 2014. doi: 10.1103/PhysRevB.90.064201. URL http://link.aps.org/doi/10.1103/PhysRevB.90.064201.

[233] Trithep Devakul and Rajiv R. P. Singh. Early breakdown of area-law entanglement at the many-body delocalization transition. *Phys. Rev. Lett.*, 115:187201, Oct 2015. doi: 10.1103/PhysRevLett.115.187201. URL http://link.aps.org/doi/10.1103/PhysRevLett.115.187201.

[234] Eman Hamza, Robert Sims, and Günter Stolz. Dynamical localization in disordered quantum spin systems. *Communications in Mathematical Physics*, 315(1):215–239, 2012.

[235] James E Gentle. *Matrix algebra: theory, computations, and applications in statistics*. Springer Science & Business Media, 2007.

[236] James Hardy Wilkinson, James Hardy Wilkinson, and James Hardy Wilkinson. *The algebraic eigenvalue problem*, volume 87. Clarendon Press Oxford, 1965.

[237] Andrea De Luca and Antonello Scardicchio. Ergodicity breaking in a model showing many-body localization. *EPL (Europhysics Letters)*, 101 (3):37003, 2013.

[238] Soumya Bera, Henning Schomerus, Fabian Heidrich-Meisner, and Jens H. Bardarson. Many-body localization characterized from a one-particle perspective. *Phys. Rev. Lett.*, 115:046603, Jul 2015. doi: 10.1103/PhysRevLett.115.046603. URL http://link.aps.org/doi/10.1103/PhysRevLett.115.046603.

[239] Luigi Amico, Rosario Fazio, Andreas Osterloh, and Vlatko Vedral. Entanglement in many-body systems. *Reviews of Modern Physics*, 80(2): 517, 2008.

[240] Valerie Coffman, Joydip Kundu, and William K. Wootters. Distributed entanglement. *Phys. Rev. A*, 61:052306, Apr 2000. doi: 10.1103/PhysRevA.61.052306. URL http://link.aps.org/doi/10.1103/PhysRevA.61.052306.

[241] Pasquale Calabrese, Luca Tagliacozzo, and Erik Tonni. Entanglement negativity in the critical ising chain. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(05):P05002, 2013.

[242] Asher Peres. Separability criterion for density matrices. *Physical Review Letters*, 77(8):1413, 1996.

[243] Michał Horodecki, Paweł Horodecki, and Ryszard Horodecki. Separability of mixed states: necessary and sufficient conditions. *Physics Letters A*, 223(1):1–8, 1996.

[244] Tobias J. Osborne and Frank Verstraete. General monogamy inequality for bipartite qubit entanglement. *Phys. Rev. Lett.*, 96:220503, Jun 2006. doi: 10.1103/PhysRevLett.96.220503. URL http://link.aps.org/doi/10.1103/PhysRevLett.96.220503.

[245] Frank Verstraete, Koenraad Audenaert, Jeroen Dehaene, and Bart De Moor. A comparison of the entanglement measures negativity and concurrence. *Journal of Physics A: Mathematical and General*, 34(47): 10327, 2001.

[246] Adam Miranowicz and Andrzej Grudka. Ordering two-qubit states with concurrence and negativity. *Physical Review A*, 70(3):032326, 2004.

[247] Adam Miranowicz and Andrzej Grudka. A comparative study of relative entropy of entanglement, concurrence and negativity. *Journal of Optics B: Quantum and Semiclassical Optics*, 6(12):542, 2004.

[248] S. Bera. private communication.

[249] Thomas Jörg and Helmut G Katzgraber. Universality and universal finite-size scaling functions in four-dimensional ising spin glasses. *Physical Review B*, 77(21):214426, 2008.

[250] H Chamati and DM Dantchev. Renormalization group treatment of the scaling properties of finite systems with subleading long-range interaction. *The European Physical Journal B-Condensed Matter and Complex Systems*, 26(1):89–99, 2002.

[251] Nelson A Alves, JR Drugowich De Felicio, and Ulrich HE Hansmann. Partition function zeros and leading-order scaling correction of the 3d ising model from multicanonical simulations. *Journal of Physics A: Mathematical and General*, 33(42):7489, 2000.

[252] John A Holbrook, David W Kribs, and Raymond Laflamme. Noiseless subsystems and the structure of the commutant in quantum error correction. *Quantum Information Processing*, 2(5):381–419, 2003.

[253] Nicholas J Higham et al. Analysis of the cholesky decomposition of a semi-definite matrix. In *in Reliable Numerical Computation*, 1990.

Appendix A

Extrapolation with the BST Algorithm

We now briefly overview the Bulirsch-Stoer extrapolation scheme, commonly referred to as the "BST" Algorithm (the meaning of the "T" in this acronym has evidently been lost to time). This method was introduced in [160] in the context of differential equations, but has been widely adopted as an extrapolation scheme whenever one seeks to project a sequence of data with unknown functional form to its infinite limit. In particular it has become a useful tool for finite-size scaling techniques, and was studied extensively in the context of lattice models in [161].

The BST Algorithm assumes that we are attempting to extrapolate to a limiting value for an infinite system, which is subject to power-law corrections when approximated by a finite-size. For example suppose we have a sequence of critical field values which are approximants to the true value of the critical field in the infinite system: $\{B(L_1), B(L_2)...B(L_N)\}$, which approach $B_\infty \equiv B(L \to \infty)$. The BST Algorithm applies when, for each estimate $B(L)$, $B_\infty - B(L) = P(L)$ for some fixed (but unknown) polynomial P. This pattern of power-law corrections has generally been found to be true in the case of Binder Cumulants [249, 250].

The technique works by taking the initial sequence $\{\alpha_1^{(0)}, \alpha_2^{(0)}, ...\alpha_N^{(0)}\}$ and using it to construct a new sequence, $\{\alpha_1^{(1)}, \alpha_2^{(1)}, ...\alpha_{k-1}^{(1)}\}$ whose convergence towards the infinite limit has been accelerated, so that $\alpha_{N-1}^{(1)}$ is in fact a better estimate than $\alpha_N^{(0)}$. For clarity, note that we are using parenthetical superscripts to label the sequence, and subscripts to enumerate the terms within a sequence.

The terms in this new sequence are defined as follows

$$\alpha_k^{(j+1)} \equiv \alpha_{k+1}^{(j)} + \frac{\alpha_{k+1}^{(j)} - \alpha_k^{(j)}}{\left(\frac{L_{k+1}}{L_k}\right)^{\omega}\left(1 - \frac{\alpha_{k+1}^{(j)} - \alpha_k^{(j)}}{\alpha_{k+1}^{(j)} - \alpha_{k+1}^{(j-1)}}\right) - 1}. \tag{A.1}$$

Note that, since the denominator of equation A.1 makes reference to the sequence $\alpha^{(j-1)}$, it is necessary to define the sequence $\{\alpha^{(-1)}\}$ to handle the initial step of the algorithm in which $j = 0$. To that end, one simply takes $\alpha_k^{(-1)} = 0$ for all k.

This procedure can then be repeated, taking the sequences $\alpha^{(0)}$ and $\alpha^{(1)}$ as the inputs to generate $\alpha^{(2)}$, and so on. This iteration can be done at most $N - 1$ times, at which point the resulting sequence $\alpha^{(N-1)}$ contains only one term. This term is the BST algorithm's best estimation of the infinite limit of the original sequence.

The parameter "ω", which appears as the exponent on the length scales, is a free parameter in the algorithm. The value of ω which gives the best convergence will depend on the form of the power law corrections in the original sequence, which is generally unknown. Hence, in practice, a range of parameters must be considered, selecting the one which best optimizes the convergence. To this end, note that a rough estimate of the "precision" of the sequences can be made by

$$\Delta_k^{(j+1)} = 2|\alpha_k^{(j)} - \alpha_{k+1}^{(j)}|. \tag{A.2}$$

This value should be decreasing with each iteration if the procedure is correctly accelerating the convergence of each new sequence. The final value of this estimator, $\Delta_{final} = \Delta_1^{(N-1)}$ gives a convenient way to fix the free parameter ω: we simply repeat the algorithm while varying ω, and choose the one which minimizes Δ_{final}. In practice, it has often been observed [251] that the dependence of the estimations on ω is rather weak, with large ranges of values giving comparable results. In other words, it is often more important simply to avoid a "bad" value of ω than to try to find its absolute "best" value. In our work, we have used as our procedure a sweep over the range $\omega \in (0, 2]$, testing with steps of size 0.1. We also require that our extrapolation be "stable" under small variations in ω.

We note that the value of Δ_{final} cannot be used as a complete measure of the error in a final estimation. It is a measure of the internal consistency and the precision of the acceleration in the BST algorithm, but cannot contain any information about whether the algorithm has captured the "true" functional form of the finite-size corrections. Additionally, it does not reflect the propagation of errors on the data which are being extrapolated. A very small and stable value of Δ_{final} indicates that the algorithm is extrapolating the data to the best of it's capability given its assumptions and the finite number of input points. It does not necessarily indicate that the result is extremely precise.

In this paper, to estimate the error in a BST extrapolation, we start with the uncertainties of the input points, and essentially determine the propagated uncertainty empirically. In our case, the input points are the crossings of Binder cumulant curves. The crossings are computed by linearly extrapolating between data points, so we generously assume an uncertainty of one half the step size between points. The error is then estimated by considering a "worst-case scenario" in which the first few crossing points are perturbed downward by this amount, and the later points perturbed upwards. We run these perturbed points through the BST algorithm and observe the effect on the resulting extrapolation. The size of this effect is taken to be a rough upper bound on the total uncertainty.

Appendix B

Proof of Projector Property for Unital Quantum Channels

In Sec. 5.5.2, we considered a one-site translationally invariant MPS represented by the tensor A^j, a constructed a matrix W from the dominant eigenvector of this state's transfer matrix. We further constructed an associated projector P which projected onto the support of W, and subsequently used the statement that

$$A^j P = P A^j P \tag{B.1}$$

This statement was crucial to demonstrating how to place a general MPS tensor in block diagonal form. It is also employed in Refs. [15, 195], but we wish to provide here a more detailed proof than the motivation which is given for it in those works.

In Ref. [15]) it is observed that that the dominant eigenvector of the transfer matrix is equivalent to the fixed point of the quantum channel

$$\varepsilon(X) = \sum_j A^j X (A^j)^\dagger \tag{B.2}$$

In this perspective, the tensors A^j are serving as the so-called "Kraus Operators" of the channel. Further, the fact that the tensors represent an MPS in canonical form is equivalent to saying that the channel is "unital," i.e. that it preserves the identity matrix [15, 252]. For a unital channel, it is known that the fixed point W commutes with all the Kraus operators, i.e.

$$[W, A^j] = 0. \tag{B.3}$$

See, for example, Ref. [252], Theorem III.4.

From here, we can prove directly our desired result that $A^j P = PA^j P$. Recall that P is a projector onto the support of W. As such we have trivially that $W = PW$. Consequently,

$$[PW, A^j] = 0. \tag{B.4}$$

Expanding this commutator, we have

$$P[W, A^j] + [P, A^j]W = 0. \tag{B.5}$$

Here, the first term vanishes according to Eq. B.4, so we have also

$$[P, A^j]W = 0, \tag{B.6}$$

from which it trivially follows also that

$$[P, A^j]WW^{-1} = 0 \tag{B.7}$$

Naturally, WW^{-1} is equal to the identity, but crucially, this is only strictly true within the support space of W. Thus, it is more proper to say that $WW^{-1} = P$.Therefore, from Eq. B.7, we have

$$[P, A^j]P = 0 \tag{B.8}$$

or equivalently, writing out the commutator in full,

$$(PA^j - A^j P)P = 0. \tag{B.9}$$

However, since orthogonal projectors are idempotent with $P^2 = P$, this proves the original claim, because now we have

$$PA^j P = A^j PP = A^j P. \tag{B.10}$$

Appendix C

Cholesky Decomposition of a Positive Semidefinite Matrix

In this appendix we briefly describe an efficient algorithm for computing the uppder-triangular Cholesky decomposition of a matrix M as $M = U^T U$. The formal requirements for a Cholesky decomposition to exist are simply that the matrix be symmetric and positive semidefinite; however, it is only when the matrix is fully positive *definite* that the decomposition is unique [253]. As a result of this, and perhaps also an ambiguity about how to handle rows of zeros which appear in the positive semidefinite case, several commercially available linear algebra packages contain Cholesky decomposition functions which will only return a result for a positive definite input. We outline here, based in part on details provided in Ref[235], a simple alternative implementation which is flexible enough to handle a semidefinite matrix. When we refer to elements being "equal to zero" here we imply that they are equal to zero up to machine precision. In practice it may be necessary to utilize a small cutoff to determine which values are vanishing.

Let M be an $n \times n$ matrix with elements $m_{i,j}$ which is symmetric and positive semidefinite, i.e. $m_{i,j} = m_{j,i}$ and $m_{i,i} \geq 0$. Crucially from this second property the square root of a diagonal element of M will always be real, although it may be zero. We will define a new matrix U in terms of its elements $u_{i,j}$, with $u_{i,j} = 0 \; \forall \; i > j$ so that the matrix is upper-triangular. The algorithm proceeds as follows:

1. Since most of the elements will ultimately be zero, initialize U with $u_{i,j} = 0 \; \forall \; i, j$.

2. Begin with element $m_{1,1}$. If this element is equal to zero, then all elements in the first row and column must also be equal to zero. To see this, consider the submatrix

$$\tilde{M} = \begin{pmatrix} 0 & m_{1,2} \\ m_{2,1} & m_{2,2} \end{pmatrix} \tag{C.1}$$

and note that, since $m_{1,2} = m_{2,1}$, then the determinant of the submatrix is $\det \tilde{M} = 0 - (m_{1,2})^2$. If we assume $m_{1,2} \neq 0$, then the determinant of the submatrix is negative. But this contradicts the assumption that M itself is positive-definite, since we could construct acted upon only by this sub-block which would yield a negative matrix element with M. Hence we must have $m_{1,2} = m_{2,1} = 0$. This argument applies iteratively out to $m_{1,n}$ and $m_{n,1}$, so that the entire row and column must vanish whenever the diagonal element does.

Thus, in step one, we skip (i.e, leave as zero) the entire first row and column of the matrix U if we find $m_{1,1} = 0$. This continues iteratively until we find $m_{i_0,i_0} \neq 0$ for some i_0.

3. Now, for the first nonzero diagonal element m_{i_0,i_0}, we set

$$u_{i_0,i_0} = \sqrt{m_{i_0,i_0}} \tag{C.2}$$

and then fill in the remainder of the row with

$$u_{i_0,j} = \frac{m_{i_0,j}}{u_{i_0,i_0}}. \tag{C.3}$$

4. For the remaining rows i from $i_0 + 1$ to n, we first construct the new diagonal element from the preceding rows,

$$u_{i,i} = \sqrt{m_{i,i} - \sum_{k=1}^{i-1} u_{k,i}^2} \tag{C.4}$$

If this value is equal to zero, we can save time by setting the remaining elements of the row to zero as well, $u_{i_0,j} = 0$ for $j > i$. Note that this situation only arises in the positive semidefinite case.

If $u_{i,i}$ is nonzero, we fill in the remainder of the row with

$$u_{i,j} = \frac{m_{i,i} - \sum_{k=1}^{i-1} u_{k,i} u_{k,j}}{u_{i,i}} \tag{C.5}$$

for $j > i$.

In this fashion, the desired matrix U is constructed after n steps.