

Symmetric Infinite Projected Entangled-Pair State Study of Quantum Lattice Models

Simulation of Quantum Lattice Models with Nearest and Next-Nearest Neighbor Interactions

Changkai Zhang



München 2021

Symmetric Infinite Projected Entangled-Pair State Study of Quantum Lattice Models

Simulation of Quantum Lattice Models
with Nearest and Next-Nearest
Neighbor Interactions

Changkai Zhang

Dissertation
an der Fakultät für Physik
der Ludwig-Maximilians-Universität
München

vorgelegt von
Changkai Zhang
aus China

München, den 2021

Erstgutachter: Prof. Jan von Delft

Datum der Abgabe: 12.03.2021

Abstract

Various fascinating phenomena and phases of matter that emerge from the strongly correlated many-body systems have received much interest in condensed matter physics. However, direct treatment of a many-body system is close to impossible due to the exponentially large number of degrees of freedom. Tensor network techniques provide a compelling framework to circumvent the complexity problem. In this thesis, we employ the infinite Projected Entangled-Pair State (iPEPS) tensor network to simulate 2-dimensional quantum models defined on a square lattice with nearest-neighbor and next-nearest neighbor interactions. Compared with other popular numerical methods such as the Density Matrix Renormalization Group (DMRG) and Quantum Monte Carlo (QMC) method, iPEPS is especially competitive in its faithful representation of the entanglement area law and is free from the sign problem. Therefore, it is particularly suitable for studying fermionic models such as the Hubbard model where numerous captivating phenomena including high- T_c superconductivity may emerge.

A unique feature of our iPEPS algorithms is the capability to exploit symmetries. The QSpace tensor library is utilized to automatically keep track of the U(1) or SU(2) symmetry of the tensors. This dramatically reduces the numerical costs and allows quantum states which conserve different types of symmetries can be studied.

The quantum lattice models studied in this thesis include the Heisenberg model, the free-fermion model and the Hubbard model. We validate our iPEPS implementation using well-understood models (e.g. nearest-neighbor Heisenberg model) and an exactly solvable free-fermion model. Specifically, we investigate the U(1) and SU(2) symmetric ground state properties of next-nearest neighbor Hubbard model with next-nearest neighbor hopping amplitude $t_2 = -0.25$ at 1/8 hole doping. Remarkably, we find that the SU(2) symmetric ground state has a lower energy than the U(1) symmetric ground state with striped charge and spin orders found in previous iPEPS calculations.

Acknowledgements

I am sincerely grateful to my advisor Jan von Delft for providing me the chance to carry out this interesting project and leading me to the world of condensed matter physics. In many aspects, he has been an ideal academic advisor in turning fancy and risky ideas into well-defined practical tasks and offering guidance whenever available, despite that we are physically separated by a distance of more than 9000 km.

I am also very grateful to Jheng-Wei Li, who has provided numerous technical help and discussions which in many cases greatly advanced my progress. Also thanks to Andreas Weichselbaum, whose QSpace tensor library is key to the major findings in this thesis.

I greatly appreciate the support of my families and friends for enriching my daily life and offering help in many administrative affairs.

During the course of this project, the world was suffering from a fatal pandemic. Therefore, I want to thank everybody who contributed to the fight against the disease and who worked hard to ensure the normal order of society.

Contents

1	Introduction and Motivation	1
1.1	Effective Quantum Lattice Models	2
1.1.1	Heisenberg Model	2
1.1.2	Hubbard Model	3
1.2	Common Numerical Methods	3
1.3	Goals and Objectives	4
2	Tensor Network Techniques	5
2.1	Graphical Notation	5
2.2	Matrix Product States	6
2.3	Fermionic Statistics	10
2.4	Symmetry Tracking	12
2.5	Projected Entangled-Pair States	14
2.6	Corner Transfer Matrix	18
2.7	Ground State Search	21
2.7.1	Simple Update	23
2.7.2	Full Update	26
2.8	Observable Measurement	28
3	Simulation Results	29
3.1	General Strategies	29
3.2	Heisenberg Model	30
3.2.1	Nearest-Neighbor Heisenberg Model	31
3.2.2	Next-nearest Neighbor Heisenberg Model	32
3.3	Free-Fermion Model	33
3.4	Hubbard Model	35
3.4.1	Nearest-Neighbor Hubbard Model	35
3.4.2	Next-Nearest Neighbor Hubbard Model	38
4	Conclusions	43
References		48

Chapter 1

Introduction and Motivation

Various fascinating phenomena and phases of matter may emerge from the interplay of strong quantum correlations induced by many-body interactions. The high energy theory of most quantum many-body systems is quantum electrodynamics. However, direct application of quantum electrodynamics would encounter severe complexity problems. Simulating even a tiny piece of material is already away beyond the capability of computation we currently have. Therefore, aiming at an exact solution by brute-force *ab initio* method is pragmatically impossible.

Such issues triggered the emergence of condensed matter physics, which attempted to address quantum many-body physics via effective theories. The fundamental idea is that collective degrees of freedom could emerge when there exists a vast amount of interacting particles, and the entire system may be effectively described by emergent quasi-particles and the interaction between them. Famous examples include Bloch's nearly-free electron models and Landau's Fermi liquid theory. Nevertheless, most effective theories are highly dependent on the perturbative method, which works only when the interaction is weak enough.

The past few decades have witnessed the discovery of numerous peculiar phenomena which can not be explained by conventional effective theories. Examples include Mott insulators [1], high- T_c superconductivity [2][3], frustrated quantum magnets [4], the fractional quantum Hall effect [5] and many more. The common feature of these intriguing phenomena is that they typically exist in specifically designed materials where the interaction between particles is strongly enhanced and becomes comparable or even larger than the kinetic energy. Strong interaction leads to strong quantum correlation which sources these unconventional effects.

The appearance of strong correlation calls for alternative non-perturbative methods which can take the full many-body wavefunction into account. This poses new challenges to both theoretical and numerical analysis. Fortunately, recent researches on tensor networks, especially infinite Projected Entangled-Pair State (iPEPS) [6–8], show their highly competitive ability to resolve quantum many-body problems. And this is the method used in this thesis.

Projected Entangled-Pair State (PEPS) [9] is a type of tensor network state ansatz specifically designed to simulate the quantum state of 2-dimensional quantum lattice

models. Utilizing translational invariance, infinite PEPS (iPEPS) [10] can represent an infinitely extended quantum state through a finite number of tensors. Compared with other popular numerical methods, iPEPS has the advantage of faithful representation of entanglement properties, and circumvents the sign problem. Hence, it is particularly competitive in simulating fermionic models, which incubate many strong correlation phenomena. In the next two sections, we discuss important effective models studied in this thesis and major benefits of iPEPS in these contexts.

1.1 Effective Quantum Lattice Models

The physics inside strongly correlated systems includes complicated interaction between electrons and nucleons. Thus, additional approximations are needed to bring down the complexity of computation. A common strategy is to consider effective Hamiltonians defined on a lattice. The effective Hamiltonian includes considerably fewer degrees of freedom and should capture essential (low energy) properties of the original high energy theories. Consequently, by studying effective Hamiltonian can we still learn a lot about the physics behind.

Numerous different lattice geometries occur in nature. However, many notable strongly correlated phenomena, e.g. high- T_c superconductivity in cuprates, live on square lattices. Therefore, this thesis will focus on studying quantum lattice models defined on a square lattice.

1.1.1 Heisenberg Model

The Heisenberg model [11] describes the interaction between spins in a quantum lattice system where each lattice site hosts exactly one spin. It is extremely successful in simulating the magnetic properties of various strongly correlated systems. The Hamiltonian of Heisenberg model with nearest-neighbor interaction reads

$$\mathcal{H} = J \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j, \quad (1.1)$$

where J is the coupling constant and \mathbf{S}_i the spin operator of the electron at site i . For positive J , the spins in the ground state tends to align anti-ferromagnetically. The J_1 - J_2 Heisenberg model introduces next-nearest neighbor interaction, i.e.

$$\mathcal{H} = J_1 \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j + J_2 \sum_{\langle\langle i,j \rangle\rangle} \mathbf{S}_i \cdot \mathbf{S}_j. \quad (1.2)$$

For positive J_2 , the additional term favours an anti-ferromagnetic order in next-nearest neighbors which competes with the first term and causes frustration. Therefore, depending on the ratio of J_1 and J_2 , the ground state of J_1 - J_2 Heisenberg

model exhibits various different phases, and rich physics may emerge in the transition region between the phases. The exact nature of the ground state in this region is still under active research.

1.1.2 Hubbard Model

The Hubbard model [12] is believed to be the effective model which captures essential physics behind high- T_c superconductors. Hubbard models typically consist of a kinetic term which describes the *hopping* of electrons from one site to another, and an on-site interaction term. The Hamiltonian of a Hubbard model with nearest-neighbor hopping can be written as

$$\mathcal{H} = -t \sum_{\langle i,j \rangle, \sigma} (c_{i\sigma}^\dagger c_{j\sigma} + c_{j\sigma}^\dagger c_{i\sigma}) + U \sum_i n_{i\uparrow} n_{i\downarrow}, \quad (1.3)$$

where $c_{i\sigma}$ and $c_{i\sigma}^\dagger$ are annihilation and creation operator of electron with spin $\sigma \in \{\uparrow, \downarrow\}$ at site i , and $n_{i\sigma}$ the number of electrons operator with spin $\sigma \in \{\uparrow, \downarrow\}$ at site i ; t is the hopping amplitude and U mimics the Coulomb repulsion of electrons at the same site. For small U , the ground state shows a metallic behavior at half-filling; for sufficiently large U , the large on-site interaction would suppress the electron mobility at low temperatures, resulting in a Mott insulating state [13]. Superconducting d-wave order and stripe states are observed at finite hole doping.

The Hubbard model can also accommodate next-nearest neighbor hopping with a different hopping amplitude, i.e.

$$\begin{aligned} \mathcal{H} = & -t_1 \sum_{\langle\langle i,j \rangle\rangle, \sigma} (c_{i\sigma}^\dagger c_{j\sigma} + c_{j\sigma}^\dagger c_{i\sigma}) \\ & - t_2 \sum_{\langle\langle i,j \rangle\rangle, \sigma} (c_{i\sigma}^\dagger c_{j\sigma} + c_{j\sigma}^\dagger c_{i\sigma}) + U \sum_i n_{i\uparrow} n_{i\downarrow}. \end{aligned} \quad (1.4)$$

Solving the Hubbard model is analytically not yet possible and numerically challenging, as the ground state may exhibit abundant combinations of charge and spin orders depending on the relative value of t_1 , t_2 , U and doping, and different candidates of the ground state are energetically very close. Therefore, accurate numerical simulation is required for determination of the authentic ground state.

1.2 Common Numerical Methods

The most straight-forward way of obtaining the ground state wavefunction of strongly correlated systems is the *exact diagonalization* method [14]. However, the dimen-

sion of the full Hilbert space grows exponentially with the system size. Thus, the computational expenses required quickly go beyond our current availability.

Another important and powerful numerical method for dealing with strongly correlated systems is the *quantum Monte Carlo* method [15–17]. The common idea of quantum Monte Carlo methods is to utilize stochastic simulation, e.g. by sampling the partition function. Quantum Monte Carlo algorithms are usually highly efficient, so it is feasible to study some very large systems with hundreds of sites at a polynomial cost. However, quantum Monte Carlo techniques suffer from the well-known *sign problem* [18], which poses great barriers to the simulation of fermionic systems and many frustrated spin systems.

One of the strategies to circumvent the aforementioned limitation is offered by *tensor network* techniques. Tensor network (TN) methods encode the many-body wavefunction in the form of interconnected tensors. Each lattice site is equipped with a tensor with one open physical index representing local state space, and several virtual indices connected with neighboring sites representing entanglement.

Early examples of tensor network methods are *numerical renormalization group (NRG)* [19–21] for quantum impurity model, and *density matrix renormalization group (DMRG)* [22][23] for treating 1-dimensional systems in the form later developed as *matrix product state (MPS)*. For 2-dimensional models, one can also use a 1-dimensional chain to traverse through the entire lattice. However, such a strategy would not be able to comply with the area law of entanglement entropy.

The *Projected Entangled-Pair State (PEPS)* ansatz [9][6][7] is a 2-dimensional generalization of MPS ansatz. PEPS simulation is numerically more demanding than DMRG. Nonetheless, PEPS possesses a much better encoding of entanglement, and is thus physically more meaningful. Other superiority of PEPS includes immunization of sign problem, non-perturbative nature and the capability to treat large-size (or even infinite-size) systems. These features make PEPS method a compelling choice in simulating 2-dimensional fermionic systems.

1.3 Goals and Objectives

In this thesis, we will develop computer programs for iPEPS methods to study Heisenberg model and Hubbard model with nearest-neighbor and next-nearest neighbor interactions. The nearest-neighbor models and an exactly solvable free-fermion model will be utilized for sanity checks. For next-nearest neighbor Heisenberg model, we will exemplify different phases with representative values of J_2/J_1 . For next-nearest neighbor Hubbard model, we will investigate the U(1) and SU(2) symmetric ground state properties near 1/8 doping where various spin and charge orders may appear and the true ground state is still under controversy.

Chapter 2

Tensor Network Techniques

Tensor network techniques include a series of numerical methods for many-body physics which encode the quantum state into a network of tensors. The major tensor network method employed in this thesis is the infinite Projected Entangled-Pair State (iPEPS) method. Section 2.1 introduces a graphical notation for tensors which dramatically simplifies working with them. Section 2.2 motivates the usage of tensor networks in 1-dimensional lattice models by introducing Matrix Product State (MPS) methods. Section 2.3 and section 2.4 illustrate how fermionic statistics is implemented and how one keeps track of the symmetries in a tensor network respectively. Section 2.5 introduces the PEPS method tackling 2-dimensional models, followed by section 2.6 which elaborates how one implements iPEPS via Corner Transfer Matrix (CTM) scheme. Then, section 2.7 presents the simple update scheme (for models with both nearest-neighbor and next-nearest neighbor interactions) and the full update scheme (for models with nearest-neighbor interaction only), the core algorithms to obtain the ground state of given models. Finally, section 2.8 demonstrates how observables can be measured in the iPEPS formalism.

2.1 Graphical Notation

This section explains the graphical notation of tensors and their contraction operation. Tensor network methods typically involve complicated manipulations of high-rank tensors. Presenting all these tensor operations in terms of traditional mathematical formulas would lead to lengthy expressions with numerous tensor indices. Moreover, the significance of many 2-dimensional tensor expressions would be obscure when writing them out line by line. Therefore, analogous to the Feynman diagrams which have been widely used in perturbation theories, diagrammatic expressions for tensor networks are invoked to overcome the inconvenience of formulas.

Tensor network diagrams usually consist of circles (or rectangles, etc.) and lines. Each circle represents a tensor and the lines that connect with the circle represent the indices of the tensor. Figure 2.1 shows three examples of diagrammatic expressions

of rank-1 tensor (vector), rank-2 tensor (matrix) and rank-3 tensor. The azimuth of lines is usually irrelevant, while the order of lines can be crucial in fermionic cases.

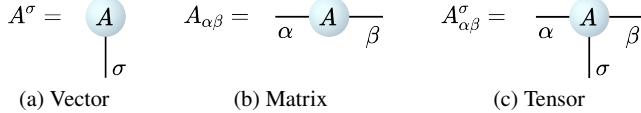


Fig. 2.1 Diagrammatic expression of vector, matrix and rank-3 tensor. Each circle represents a tensor. Each external line (usually called leg or bond) represents an index of the tensor.

The contraction of tensors is diagrammatically represented by joining lines corresponding to the same index. As shown in Figure 2.2, the internal line corresponding to index β indicates a contraction. Conventionally, open lines are called legs and connected lines are called bonds.

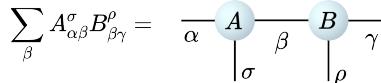


Fig. 2.2 Diagrammatic expression of a contraction operation on index β . Contraction is represented by a connected line. Indices are labelled for clarity, but can be omitted in practice.

A tensor network may involve numerous contraction operations. And the order of performing the contractions does not influence the results obtained. However, it does affect the computational complexity of the algorithm. The computational complexity can be directly read off from the tensor diagrams. For instance, the computation of the contraction in Figure 2.2 requires $O(\mathcal{D}[\alpha]\mathcal{D}[\sigma]\mathcal{D}[\beta]\mathcal{D}[\rho]\mathcal{D}[\gamma])$ operations, where $\mathcal{D}[i]$ is the dimension of index i . In general, the computational complexity is the product of the dimension of all the lines appearing in the diagram. However, depending on the topology of the diagram, the actual numerical cost may depend on the order of in which contractions are computed.

2.2 Matrix Product States

Before heading towards 2-dimensional tensor networks, let us start first with the 1-dimensional case. There is no intention for a detailed and self-contained discussion of 1-dimensional models and algorithms here. However, many fundamental concepts that are constantly encountered in various tensor network formalisms become much more intuitive in the simple 1-dimensional situation. Therefore, it would be beneficial to elaborate on these basic components and then generalize to higher dimensions.

Consider a quantum lattice model defined on a 1-dimensional chain of length N . Each lattice site hosts a local state space of dimension d . Then, any quantum state can be expanded using the basis of Fock space $|\sigma_1\rangle|\sigma_2\rangle\cdots|\sigma_N\rangle$ as

$$|\psi\rangle = \sum_{\sigma_1, \sigma_2, \dots, \sigma_N} \Psi^{\sigma_1 \sigma_2 \cdots \sigma_N} |\sigma_1\rangle|\sigma_2\rangle\cdots|\sigma_N\rangle. \quad (2.1)$$

The coefficient $\Psi^{\sigma_1 \sigma_2 \cdots \sigma_N}$ can be regarded as a rank- N tensor. And the tensor network techniques are exactly the methods used to construct and analyze this tensor. The idea of Matrix Product State (MPS) [23–25] is to decompose this large tensor into the product of a set of rank-3 tensors as

$$\Psi^{\sigma_1 \sigma_2 \cdots \sigma_N} = \sum_{\alpha_1, \alpha_2, \dots, \alpha_N} A_{\alpha_1}^{\sigma_1} A_{\alpha_1 \alpha_2}^{\sigma_2} A_{\alpha_2 \alpha_3}^{\sigma_3} \cdots A_{\alpha_{N-2} \alpha_{N-1}}^{\sigma_{N-1}} A_{\alpha_{N-1}}^{\sigma_N} \quad (2.2)$$

Such a long mathematical formula is not convenient and informative enough. Therefore, the diagrammatic expression introduced in the last section is invoked. In terms of diagrams, the above formula can be rewritten as shown in Figure 2.3 [25].

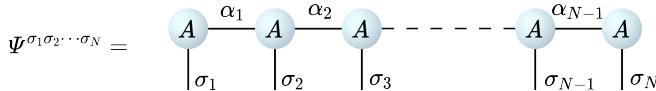


Fig. 2.3 Matrix Product State (MPS) representation of many-body wavefunction. Each component of the tensor Ψ is acquired by a series of matrix products.

The motivation of the MPS representation is straight-forward. The first tensor $A_{\alpha_1}^{\sigma_1}$ is simply a unitary transform that changes the basis of the local space associated to the first site. The second tensor $A_{\alpha_1 \alpha_2}^{\sigma_2}$ takes the local state of the first site (depicted by index α_1) and the local state of the second site (*physical index* σ_2), and combines them into a two-body state labelled by *virtual bond index* α_2 . Such a procedure is repeated until all the N sites are included.

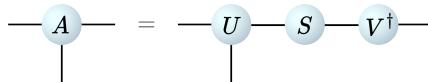


Fig. 2.4 Singular Value Decomposition (SVD) of tensor A , where $U^\dagger U = \mathbb{1}$, $V^\dagger V = \mathbb{1}$ and S is an non-negative diagonal matrix. Truncation may be performed via keeping only the largest singular values.

This construction generates an MPS form of the wavefunction accurately without any approximation. And it is apparent that the dimension of bond α_i is d^i , which still grows exponentially. Thus, extra measures are needed to reduce the bond dimension.

The technique that we are going to adopt is the Singular Value Decomposition (SVD) that has already been widely used in the field of data compression. The

singular values procured from the SVD act as a perfect measure of the *contribution* of each spectrum in reconstructing original information. Figure 2.4 shows the SVD of the tensor A . Next, one truncates matrix S by keeping only the largest D singular values. Then, use the truncated version of U as the new version of A and contract

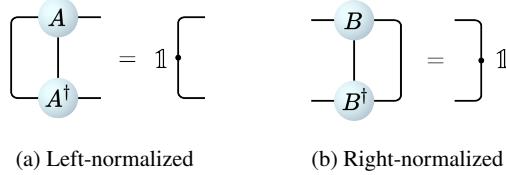


Fig. 2.5 Left-normalized and right-normalized tensor. $\mathbb{1}$ labels the identity matrix. Both can be generated through SVD.

S and V^\dagger into the followed up tensor. The SVD can subsequently be performed from left to right (right to left) and produce left-normalized (right-normalized) tensors (defined as Figure 2.5) with bond dimensions no larger than D .

The Hermitian conjugate of the tensor Ψ (considered as a linear operator) can be immediately obtained from the MPS form. Diagrammatically, one simply needs to perform a vertical flip of the diagram in Figure 2.3 [25].

$$\Psi^{\dagger \sigma_1 \sigma_2 \cdots \sigma_N} = \begin{array}{c} | \sigma_1 \\ | \sigma_2 \\ | \sigma_3 \\ \vdots \\ | \sigma_{N-1} \\ | \sigma_N \end{array} \begin{array}{ccccccc} A^\dagger & - & A^\dagger & - & A^\dagger & - & A^\dagger \end{array} \begin{array}{c} A^\dagger \\ A^\dagger \\ A^\dagger \\ \vdots \\ A^\dagger \\ A^\dagger \end{array}$$

Fig. 2.6 MPS representation of the Hermitian conjugate of tensor Ψ . Diagrammatically, this is simply the vertical flip of Figure 2.3.

The computation of observables can also be made explicit by using the MPS representation of ket and bra. For example, if we have local observables O_i and O_j located at site i and j respectively, the correlation $\langle \psi | O_i O_j | \psi \rangle$ can be computed using the diagram shown in Figure 2.7.

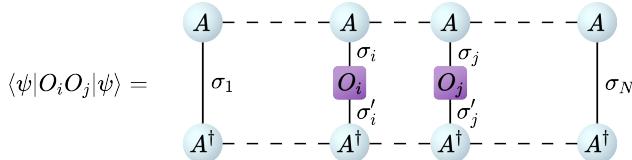


Fig. 2.7 Observable measurement using MPS representation. O_i and O_j are two local observables located at site i and j respectively.

The singular values computed from the SVD of tensor A have vital physical significance. To understand this, we need the Lambda-Gamma form of the MPS as shown in Figure 2.8. In Lambda-Gamma form [26][27], a singular value matrix

Λ is inserted between any two rank-3 Γ tensors. By absorbing Λ matrix into a neighboring Γ tensor, e.g. defining $A_i \equiv \Lambda_{i-1}\Gamma_i$ (left-normalized) or $B_i \equiv \Gamma_i\Lambda_i$ (right-normalized), one retrieves the original form of the MPS.

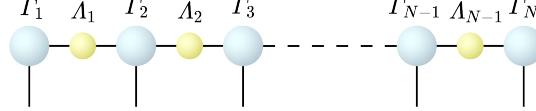


Fig. 2.8 Lambda-Gamma form of MPS. Singular value matrices Λ_i explicitly enters the tensor network. Each pair of Λ and Γ constitute a tensor.

Let us now investigate the physical significance of the singular values. As shown in Figure 2.9, an arbitrary Λ_i divides the entire tensor network into left and right parts. For specific α (or β), the left part Ξ_L (or the right part Ξ_R) defines a quantum state $|\alpha\rangle_L$ (or $|\beta\rangle_R$) for the left (or right) subsystem.

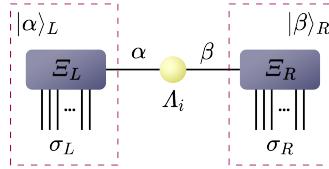


Fig. 2.9 Any Λ_i divides the MPS into two parts. Both parts define a quantum state of the corresponding subsystem. Thus, the singular values become Schmidt coefficients encoding the entanglement.

Notice that the singular value matrix Λ_i is diagonal. Consequently, the total many-body state can be written in the form [28]

$$|\psi\rangle = \sum_{\alpha} \Lambda_i^{\alpha\alpha} |\alpha\rangle_L |\alpha\rangle_R. \quad (2.3)$$

Thus, we can identify the singular values as the *Schmidt coefficients* which encode the entanglement between the left and right subsystems.

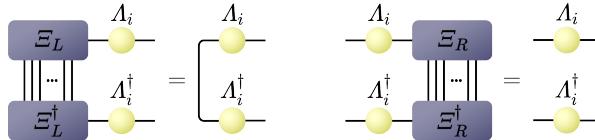


Fig. 2.10 All tensors in Ξ_L (Ξ_R) are left-normalized (right-normalized). Hence, one can replace Ξ_L (Ξ_R) by the neighboring singular value matrix.

Another crucial application of the Λ matrices emerges from the information carried by the singular values. It is clear that the Γ and Λ tensors in Ξ_L can pair up

in a way to make all the tensors left-normalized. This gives $\Xi_L^\dagger \Xi_L = \mathbb{1}$. Similarly, there is $\Xi_R \Xi_R^\dagger = \mathbb{1}$. Therefore, the singular value matrix Λ already carries enough information about Ξ_L or Ξ_R to enable observables to be computed [25], as shown in Figure 2.10.

The above property is especially proficient when studying infinite-size tensor networks. The singular value matrix Λ can effectively replace Ξ regardless of how many tensors Ξ includes. Thus, one can simply treat Λ as some *environment* instead of using a possibly infinite-size Ξ . Such a technique is the foundation of the simple update algorithm for the ground state search discussed in the later section.

2.3 Fermionic Statistics

In the previous discussions, we implicitly suggested that the tensor network abides by bosonic statistics. However, much attention and efforts focus on the strongly correlated system that includes fermions. Fermionic lattice models such as Hubbard model are particularly challenging partly because the most effective competitor, quantum Monte Carlo method, suffers from sign problems. Therefore, much expectation is placed upon tensor network techniques.

Parity Conservation

Most fermionic systems preserve the parity of the number of fermions, i.e. whether the system has an odd or even number of fermions. This \mathbb{Z}_2 parity symmetry poses restrictions to the tensors in the tensor network [6][29]. Since the tensors simply represent quantum states, they should also preserve parity symmetry p , i.e.

$$A_{\alpha\beta}^\sigma = 0, \quad \text{if} \quad p(\alpha)p(\beta)p(\sigma) = -1. \quad (2.4)$$

where $p(\alpha) \in \{-1, 1\}$ labels the parity of the state corresponding to index α ; $p = -1$ indicates an odd number of fermions and $p = 1$ an even number of fermions. One can use the symmetry tracking technique introduced in section 2.4 to record the propagation of parities during computation.

For convenience, we require all the tensors entering the algorithm to comply with the above parity rule. Some operators, such as fermion creation c^\dagger and annihilation c operator, do not naturally support the parity conservation. Extra settings required in this case will be discussed later in this section.

Fermionic Swap Gate

The parity symmetry alone does not capture the fermionic statistics. However, the parity of the quantum state itself is crucial information for the determination of fermionic signs. Quantum states with an odd number of fermions can be created with an odd number of creation operators. Therefore, switching the order of two indices

with odd parity generates a minus sign. This procedure can be implemented by the so-called *fermionic swap gate* [6][29] defined as Figure 2.11. Each line crossing in the fermionic tensor network diagram should thus be replaced by the swap gate.

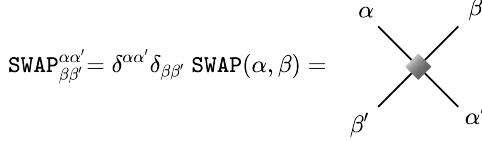


Fig. 2.11 Definition and diagrammatic expression of the fermionic swap gate, where $\text{SWAP}(\alpha, \beta) = -1$ if $p(\alpha) = p(\beta) = -1$ and $\text{SWAP}(\alpha, \beta) = 1$ otherwise. A swap gate is needed for every line crossing in the tensor network.

As shown in Figure 2.11, a swap gate accounts for the fermionic statistics by adding an extra minus sign when the crossed lines both carry odd fermionic parities. For complicated tensor network diagrams, especially 2-dimensional diagrams, there may be numerous unavoidable line crossings. However, adding swap gates does not alter the apparent computational complexity, as each application of the swap gate can be absorbed into one single tensor. Consequently, one only needs to add the swap gate at appropriate spots in the algorithm.

Fermionic Operator

It has been stated earlier that we restrict the tensors entering fermionic tensor network algorithms to be parity-preserving. This is easily satisfied by tensors which describe quantum states. However, some vital operators such as the creation and annihilation operators violate the parity rule. Therefore, we use a trick that adds an auxiliary leg which takes only one value and odd parity [6][29], as shown in Figure 2.12.

$$(c^\dagger)_{\sigma, \delta}^{\sigma'} = \begin{array}{c} | \\ \text{green circle} \\ | \\ \sigma' \end{array} - \delta \quad (c)_{\sigma, \delta}^{\sigma'} = \begin{array}{c} | \\ \text{green circle} \\ | \\ \sigma' \end{array} - \delta$$

Fig. 2.12 Tensors representing parity changing fermionic operators need an auxiliary leg, with odd parity by definition, in order to maintain the parity conservation requirement.

The creation c^\dagger and annihilation c operator change the parity by adding or removing one fermion. Thus, the parity of σ and σ' are opposite to each other. By attaching one extra index δ with odd parity, one retrieves the parity conservation

$$p(\sigma)p(\sigma')p(\delta) = -p(\delta) = 1. \quad (2.5)$$

We conclude this section by mentioning another possible subtlety lying in the expansion of two-site operators with respect to some two-site basis. The Hermitian

conjugate of basis $|\sigma_i\sigma_j\rangle$ is $\langle\sigma_j\sigma_i|$ (notice the order of σ_i and σ_j has been switched), where $\sigma_i \in \{0, 1\}$ labels the occupation number of site i and similarly for σ_j . Hence, one is advised to pay extra attention to the minus sign that may occur in some components of two-site operators.

2.4 Symmetry Tracking

Symmetry is one of the most essential concepts in physics. In quantum physics, symmetry-generated quantum numbers are the only indicators for distinct quantum states. Moreover, quantum phase transitions are often accompanied by a switch of symmetries.

In the context of tensor networks, tracing the symmetries within the network may have profound effects. The existence of symmetry implies that the tensors may be highly sparse, and one can make use of symmetries for compression which could lead to huge numerical advances. Besides, by turning on different symmetries, one can study possible symmetry-breaking in specific models of interest.

In this thesis, the work of symmetry tracking is handed over to the QSpace library [30]. Here, we only discuss the basic mechanism of symmetry book-keeping. For more technical details, it is advised to consult the introductory paper [29] or documentation of QSpace library [30].

Symmetry and Group

Strongly correlated systems usually incorporate abundant different symmetries. Mathematically, symmetries can be handled using group theory. In the discussion of fermionic tensor networks 2.3, we have exploited the parity \mathbb{Z}_2 symmetry. U(1) symmetry is another commonly encountered symmetry which accounts for e.g. the conservation of charges. Many spinful models preserve SU(2) symmetry which conserves the total spin of the system. Albeit more complex symmetries are also possible in some strongly correlated systems, they are not used in this thesis nonetheless.

A direct consequence of symmetry is the commutation of the Hamiltonian \mathcal{H} and the symmetry group generators T_i , i.e. $[\mathcal{H}, T_i] = 0$. This indicates that the Hamiltonian can be written in a block-diagonal form containing non-zero elements only within the blocks (named *symmetry sectors*). Therefore, one attains great computational benefits by getting rid of unnecessary zero elements. The more intricate the system is, the more evident the numerical benefits will be.

Abelian Symmetry

Abelian symmetry is a simple symmetry where any two elements of the symmetry group commute with each other. Quantum states $|ql\rangle$ which preserve Abelian symmetry can be identified using two numbers q and l , where q designates the symmetry quantum number and l further resolves different states with the same

quantum number. Take the U(1) electric charge symmetry in a two-electron system for example. In this case, q labels the total charge of the system. There are three possibilities: $q = 0, -1, -2$. The first and third states are generated uniquely by $|0\rangle|0\rangle$ and $| -1 \rangle| - 1 \rangle$ respectively. While the second state may be either $| -1 \rangle|0\rangle$ or $|0\rangle| - 1 \rangle$. Therefore, the additional l is needed to distinguish these two states.

The building blocks of tensor network description of many-body states are the tensors which combine some states to give new ones. The rank-3 tensor in the MPS formalism is a direct example. When turning on the Abelian symmetry, this tensor can be rewritten as

$$|q''n\rangle = \sum_{ql,q'm} [A_{q'q''}^q]_{mn}^l |ql\rangle|q'm\rangle, \quad (2.6)$$

where quantum numbers enter as additional labels. Numerically, this means more book-keeping efforts. However, quantum numbers must obey selection rules in the symmetric case, e.g. $q'' = q + q'$ in U(1) symmetry. This leads to considerable numerical speed-up and reduced memory costs.

Non-Abelian Symmetry: SU(2)

In this thesis, we will extensively exploit non-Abelian SU(2) symmetry. One needs two quantum numbers q and q_z to designate a distinctive SU(2) representation. However, an additional label l is also needed to select a unique quantum state. Take the SU(2) spin symmetry in a three-spin system for example. There are two states with $S = 1/2, S_z = 1/2$, i.e. $|\uparrow\rangle(|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle)/\sqrt{2}$ and $|\downarrow\rangle|\uparrow\uparrow\rangle$. By Wigner-Eckardt theorem, the tensor has the form

$$|q''m, q''_z\rangle = \sum_{qlq_z, qm, q'_z} [A_{q'q''}^q]_{mn}^l \cdot C_{q'_z q''_z}^{q_z} |qlq_z\rangle|q'mq'_z\rangle, \quad (2.7)$$

where $C_{q'_z q''_z}^{q_z} = \langle qq'; q_z q'_z | q''; q''_z \rangle$ [30] is the Clebsch-Gordan coefficients which encode the selection rules. Analogously, the operators can also be expanded using Wigner-Eckardt theorem as [30]

$$\langle q'l'; q'_z | \hat{O}_{q_z}^{\tilde{q}} | ql; q_z \rangle = (O_{qq'}^{[\tilde{q}]})_{ll'}^{[1]} \cdot C_{q_z q'_z}^{[\tilde{q}_z]} \quad (2.8)$$

Notice that for a specific symmetry group, the Clebsch-Gordan coefficients are definite and only need to be computed once. This allows further compression of non-zero data blocks and reduction of numerical costs, despite an increase of book-keeping efforts.

Diagrammatic Feature

The tensors which preserve certain symmetries have an extra diagrammatic feature. As depicted in Figure 2.13, each leg of the tensor is accompanied by an arrow, and the flow of quantum numbers from *incoming* legs (bra) and *outgoing* ones (ket) should

be conserved in the sense of Eqs. (2.6), (2.7) and (2.8). There is no objective choice of the convention of arrows. Nevertheless, one has to maintain the same convention throughout the computation.

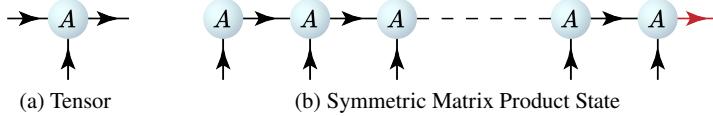


Fig. 2.13 Diagrammatic expression of the tensor and symmetric Matrix Product State. Arrows are equipped to indicate the flow of quantum numbers. An additional leg is attached to tackle possible multiplet states.

As the many-body state is constructed via a symmetric tensor network, it must preserve some global quantum numbers of the corresponding symmetry group. For finite-size tensor networks, one can always target various states with different global quantum numbers by attaching an additional leg to any tensors. However, this is not possible for infinite-size tensor networks. Therefore, infinite tensor networks are always constrained to the singlet sector.

2.5 Projected Entangled-Pair States

The Projected Entangled-Pair State (PEPS) [9] tensor network is a generalization of the Matrix Product State (MPS) ansatz to higher dimensions. In this thesis, we only study PEPS in two dimensions where many strongly correlated phenomena emerge. Like its 1-dimensional counterpart, PEPS consists of a network of tensors. However, as now each site is accompanied by four neighboring sites, the tensors need four virtual bonds, as shown in Figure 2.14.

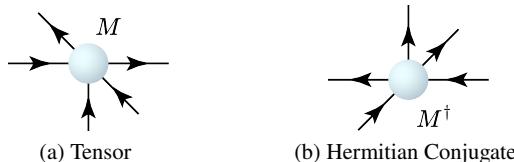


Fig. 2.14 The 2-dimensional tensor which constitutes the PEPS tensor network and the Hermitian conjugate of the tensor. The direction of arrows are flipped after complex conjugation.

We adopt the convention that the top and right legs have outgoing arrows, while the bottom and left legs have incoming arrows. Physical legs, aligned vertically here,

have incoming arrows. The Hermitian conjugate is obtained by turning the tensor diagram upside down and flipping the direction of arrows of all the legs.

Finite-size PEPS [31] consisting of a finite number of tensors simulates the quantum state on a finite-size lattice. Figure 2.15 shows a PEPS describing a 3×3 cluster. Boundary tensors may have fewer legs due to a lack of neighboring sites.

In two dimensions, one faces the subtlety of arranging the physical legs of PEPS. Figure 2.15 shows one possible arrangement, but other conventions are in general allowed. However, different conventions may lead to different placements of swap gates, which enforces one to adopt the same convention throughout the computation.

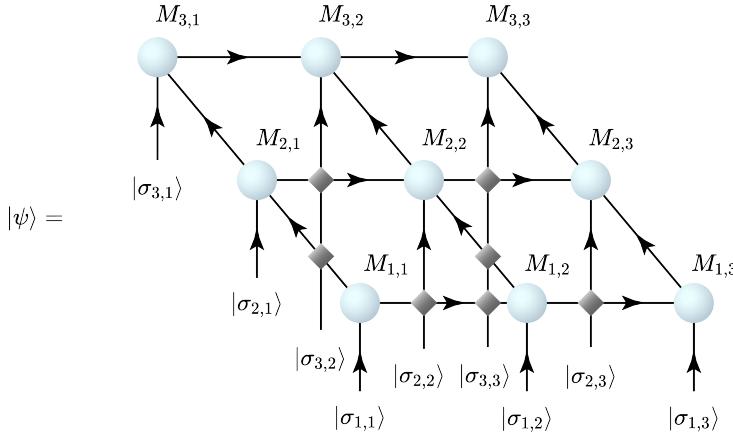


Fig. 2.15 Finite-size PEPS living on a 3×3 cluster. The arrangement order of the local indices is in general irrelevant, as long as the same convention is referred throughout the computation.

The construction that every neighboring pair of local sites has a connecting virtual bond has profound significance. Compared with competing algorithms such as Density Matrix Renormalization Group (DMRG) which traverses the 2-dimensional lattice via a 1-dimensional tensor network [32], PEPS is able to maintain the entanglement properties required by a 2-dimensional system. Concretely, for a subregion of size $L \times L$, the entanglement entropy of PEPS tensor network is bounded by [28]

$$S = O(L \log D), \quad (2.9)$$

where D is the bond dimension of the PEPS tensors. This complies with the area law of entanglement entropy [28] which is satisfied by most strongly correlated systems. However, issues might still occur near quantum criticality where extra logarithmic correction term like $O(L \log L)$ can emerge.

The bra $\langle \psi |$ is required either in computing the norm $\langle \psi | \psi \rangle$ or the expectation value of some observable $\langle \psi | O | \psi \rangle$. Diagrammatically, $\langle \psi |$ is represented by a diagram which is simply the horizontal mirror image of Figure 2.15. However, this

procedure leads to awkward technical implementations due to the complexity in the arrangement of physical sites and the consequent irregular placement of swap gates.

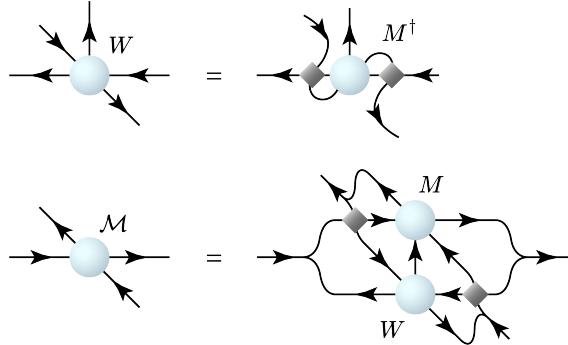


Fig. 2.16 Definition of conjugate tensor and double-layer composite tensor.
Double-layer composite tensors will be represented by calligraphy letters.

An alternative way [6][29] is depicted in Figure 2.16. We first construct the conjugate tensor W by distorting two legs of M^\dagger , such that all the four virtual bonds of W lie in parallel with the tensor M . Next, contract the physical index of corresponding M and W , and reshape the parallel legs of M and W to get the double-layer composite tensor labeled by calligraphy letter \mathcal{M} .

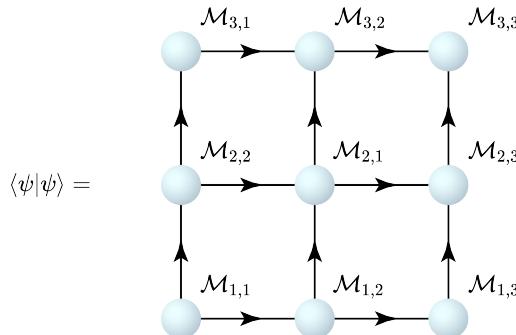


Fig. 2.17 The norm of the many-body wavefunction is now represented by a tensor network consisting of only the composite tensors without any explicit fermionic swap gates.

Using composite tensors \mathcal{M} , the norm of the many-body wavefunction $\langle\psi|\psi\rangle$ can be rephrased as Figure 2.17. The procedure in Figure 2.16 is universal and include all the fermionic swap gates into the composite tensor. Thus, one can iterate the same procedure for all sites to produce a tensor network in Figure 2.17 with no explicit swap gates.

Many strongly correlated systems lie on an infinite-size lattice with translational symmetry. Correspondingly, one can exploit the translational invariance to define the infinite PEPS (iPEPS) [10][6] tensor network.

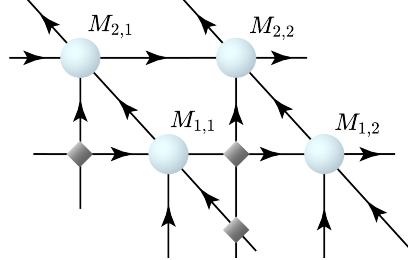


Fig. 2.18 The iPEPS tensor network with a 2×2 supercell, i.e. the entire infinite tensor network is created by infinite copies of the supercell.

The iPEPS tensor network is created by an infinite number of translated copies of some *supercell* [8]. The supercell serves as a “unit cell” of the corresponding infinite tensor network. Figure 2.18 shows a 2×2 supercell of an iPEPS. The ground state of some strongly correlated systems may spontaneously break the translational symmetries. In this case, one may demand a larger size of the supercell.

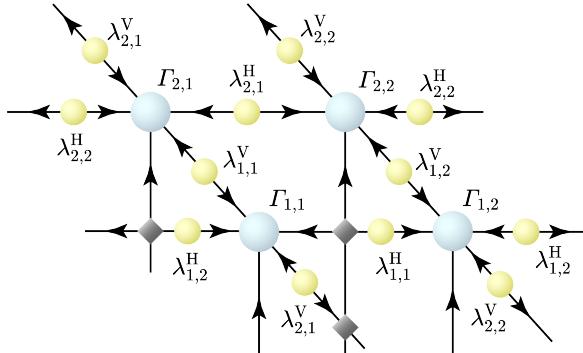


Fig. 2.19 The Lambda-Gamma form of the iPEPS tensor network. A singular value matrix λ is inserted into each virtual bond which encodes the entanglement between the two sites.

Similar to the MPS, one can define the Lambda-Gamma form of the iPEPS by introducing a singular value matrix for each virtual bond. These singular value matrices encode the entanglement between the two sides of the matrix. As seen in section 2.2, the singular value matrix can be regarded as some *environment* which carries the information of some surrounding region. This is the foundation of the simple update algorithm for ground state search and will be discussed further in

section 2.7.1. However, unlike the MPS, 2-dimensional iPEPS does not possess a canonical form [25] (e.g. left-normalized or right-normalized tensor network). Hence, alternative methods are required to obtain accurate results.

2.6 Corner Transfer Matrix

The iPEPS tensor network includes an infinite number of tensors which is pragmatically impossible to process. Thus, one has to find a way to represent some infinite subregion by a finite number of tensors. In one dimension, the singular values do the work perfectly. Nonetheless, the same strategy fails in two dimensions owing to the absence of a well-defined canonical form.

The Corner Transfer Matrix (CTM) scheme [6][33–35] is a powerful method to tackle the problem. The CTM scheme employs four corner matrices \mathcal{C} and four transfer matrices (tensors, rigorously speaking) \mathcal{T} for each composite tensor \mathcal{M} . These eight tensors represent the infinitely extended tensor network surrounding the composite tensor.

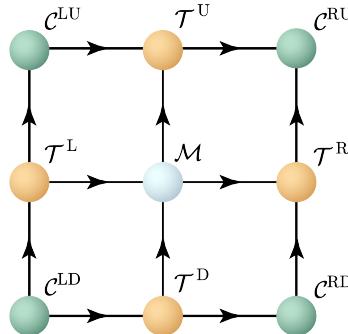


Fig. 2.20 The Corner Transfer Matrix scheme generates four corner matrices and four transfer matrices which represent the infinitely extended tensor network surrounding the composite tensor.

In order to accurately simulate the infinite-size environment, one typically needs an environmental bond (the bonds connecting corner and transfer matrix) dimension χ that is much larger than the bond dimension of the composite tensor D^2 . This is diagrammatically indicated by the different thicknesses of lines. Besides, as each composite tensor demands a set of corner and transfer matrices, the total number of tensors required is dependent on the size of the supercell. For instance, an iPEPS with 2×2 supercell needs 16 corner matrices and 16 transfer matrices in total.

The CTM scheme produces the corner and transfer matrices by a series of coarse-graining steps in all directions. As shown in Figure 2.21, the coarse-graining moves consist of two steps: first, contract the existing transfer matrix with the nearest

composite tensor, which gives a new transfer matrix with a larger environment bond dimension χD^2 ; then perform renormalization described below to reduce the environment bond dimension back to χ . Notice that the transfer matrix generated belongs to a different site, e.g. $\mathcal{T}_{j,i+1}^L$ instead of $\mathcal{T}_{j,i}^L$ in Figure 2.21. This procedure is iterated until the spectrums of transfer matrices reach convergence.

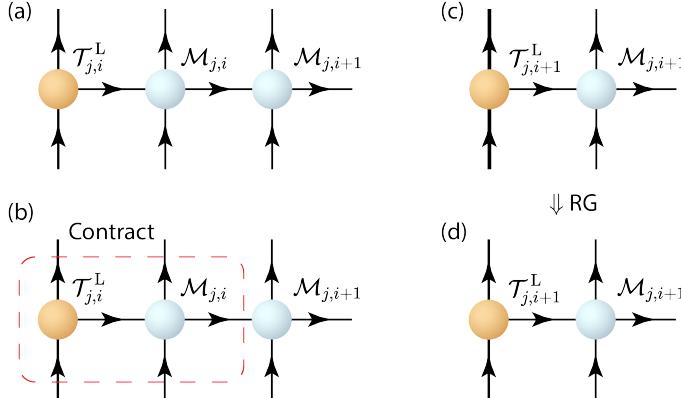


Fig. 2.21 The renormalization of transfer matrix. (a) The initial setup; (b) Contract the transfer matrix with the nearest composite tensor to get the new transfer matrix; (c) The new transfer matrix has a larger environment bond dimension χD^2 ; (d) Perform renormalization to reduce the environment bond dimension back to χ .

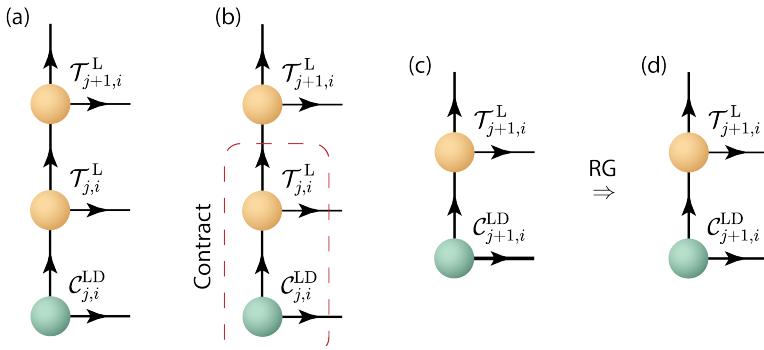


Fig. 2.22 The renormalization of corner matrix. (a) The initial setup; (b) Contract the corner matrix with the nearest transfer matrix to get the new corner matrix; (c) The new corner matrix has a larger environment bond dimension χ^2 ; (d) Perform renormalization to reduce the environment bond dimension back to χ .

The coarse-graining of corner matrices can be carried out in a similar manner. As shown in Figure 2.22, the corner matrix is contracted with the nearest transfer matrix, before renormalization is conducted to reduce the environment bond dimension.

However, there is one extra complication: there are two different directions for coarse-graining. Both directions are required to obtain convergence.

Before the renormalization procedure, an initialization is needed first as a starting point. In principle, one could use any random tensors as initialization. Nonetheless, this usually costs much more renormalization steps for convergence. A much better strategy is demonstrated in Figure 2.23 which makes use of the tensor and its conjugate.

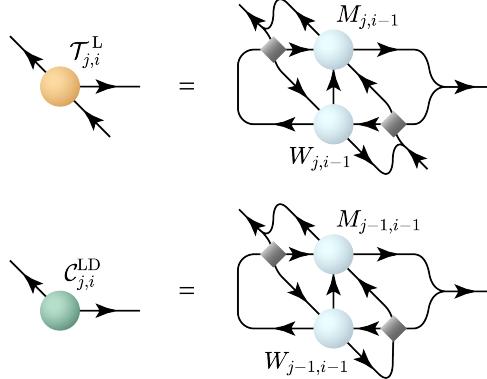


Fig. 2.23 The corner and transfer matrices can be initialized using the tensor and its conjugate. This strategy leads to a faster convergence property than random initialization.

We are still left with one final ingredient — the CTM Renormalization Group scheme. In each coarse-graining step, one needs an RG to truncate the increasing environment bond dimension. Next, we elaborate on how to achieve a relatively optimal and numerically stable truncation.

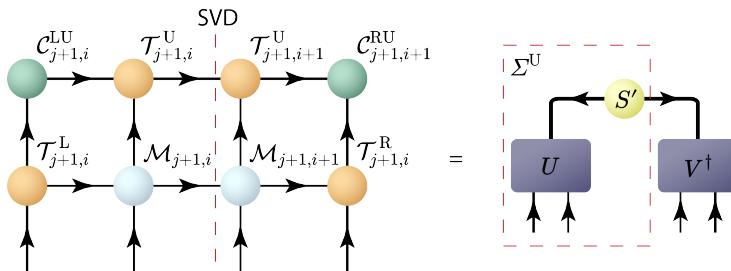


Fig. 2.24 Effective tensor network representing upper half iPEPS network. SVD is applied for the purpose of numerical stability.

First of all, we need an effective description of the upper (and lower) half of the iPEPS tensor network, as shown in Figure 2.24. An early CTMRG [6] simply employed the left-hand side of Figure 2.24, while [35] later on proposed an SVD in

the middle to improve numerical stability by removing tiny singular values. Then, contract the tensor U and singular value matrix S' obtained from the SVD to get tensor Σ^U . Analogously, one can get Σ^D from the lower half plane.

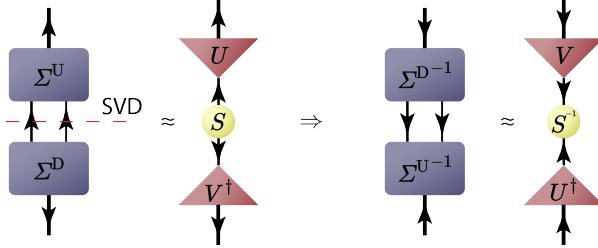


Fig. 2.25 Apply SVD on $\Sigma^U \Sigma^D$ and keep only the largest χ singular values for truncation. The inverse $\Sigma^{D-1} \Sigma^{U-1}$ is prepared for the next step.

Next, construct matrix $\Sigma^U \Sigma^D$ and apply an SVD. The renormalization or truncation is achieved by keeping only the largest χ singular values. To the right of Figure 2.25, we also show the inverse of matrix $\Sigma^U \Sigma^D$, which is required for the following construction of projectors.

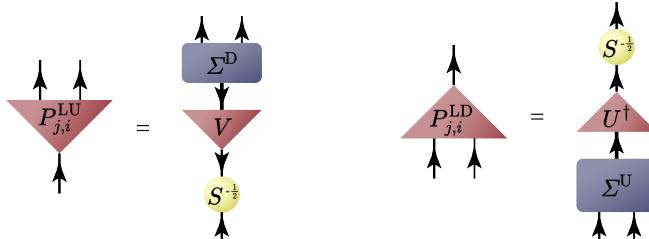


Fig. 2.26 Construction of projectors for CTMRG scheme. Two different projectors are built each time dealing with legs pointing in opposite directions.

The final step builds two projectors, as shown in Figure 2.26. One can easily verify that $P^{LU} P^{LD} \equiv \mathbb{1}$. Projector $P_{j,i}^{LU}$ acts on the downward leg of $T_{j+1,i+1}^L$ or $C_{j+1,i+1}^{LU}$. Projector $P_{j,i}^{LD}$ acts on the upward leg of $T_{j,i+1}^L$ or $C_{j,i+1}^{LU}$. The procedure needs to be repeated for all i and j to generate a complete set of projectors. Projectors are contracted with tensors within the red box in Figure 2.21 and Figure 2.22 to achieve renormalization [35][29].

2.7 Ground State Search

In this section, we discuss how to search for the ground state represented by an iPEPS under a given interacting Hamiltonian. Conventional perturbative methods hold solely for weak interactions. Here, we employ imaginary time evolution [26], which

is non-perturbative, to tackle strongly correlated systems. Both nearest-neighbor and next-nearest neighbor interaction terms are allowed.

The imaginary time evolution method is based on the fact that an evolution to $\beta \rightarrow \infty$ projects out all excited states. Concretely, if one applies the time evolution gate $e^{-\beta \mathcal{H}}$ to any randomly generated state $|\psi\rangle$, there is

$$e^{-\beta \mathcal{H}} |\psi\rangle = e^{-\beta E_g} |GS\rangle + e^{-\beta E_1} |1\rangle + e^{-\beta E_2} |2\rangle + \dots, \quad (2.10)$$

where $|GS\rangle, |1\rangle, |2\rangle$ are ground state, first, second excited state respectively, and E_g, E_1, E_2 their corresponding energy. By definition, $E_g < E_1 < E_2 < \dots$. Therefore, when $\beta \rightarrow \infty$, all the excited states acquire a negligibly small coefficient compared with the ground state, which leaves only the ground state on the right-hand side.

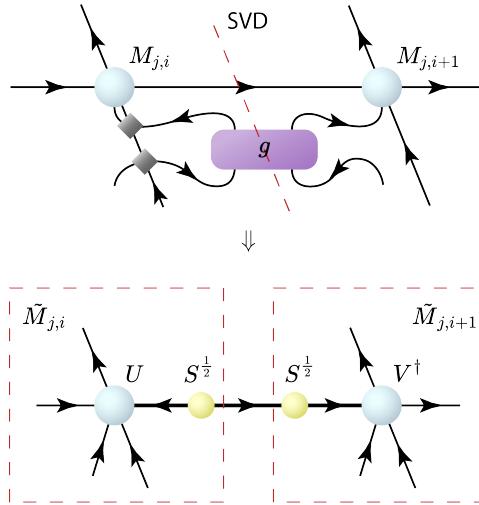


Fig. 2.27 Apply nearest-neighbor Suzuki-Trotter gate to time evolve the state. Perform SVD to separate the two tensors. The new tensors acquire larger bond dimensions, which are subjected to truncations later on.

Direct application of the imaginary time evolution gate is numerically implausible as it is a tensor with infeasibly high ranks. A common strategy is to utilize Suzuki-Trotter decomposition

$$e^{-\beta \mathcal{H}} = \prod_{(x,y)} e^{-\beta \mathcal{H}_{x,y}} + O(\beta^2), \quad (2.11)$$

where $\mathcal{H}_{x,y}$ is the local two-site interaction term of site x and y . Suzuki-Trotter decomposition turns the high-rank imaginary time evolution gate into a sequence of two-site local Trotter gates $e^{-\beta \mathcal{H}_{x,y}}$. The application of a Trotter gate is much easier. Figure 2.27 exemplifies in the nearest-neighbor case. Next, an SVD is carried out

and updated tensors are obtained. Notice that the updated tensors acquire a larger bond dimension (D^3d in this case). Therefore, additional truncation schemes are needed to avoid exponential growth of the bond dimension.

Suzuki-Trotter decomposition comes with a systematic error (named Trotter error) of order $O(\beta^2)$. Hence, to achieve high accuracy, one typically needs to use a small time step β and apply the gate numerous iterations to go to the temporal boundary.

2.7.1 Simple Update

Simple update [36] is a numerically economical but relatively inaccurate update scheme. As argued in section 2.2, the singular values can be regarded as the environment of some finite region of the iPEPS. This is the core idea of simple update.

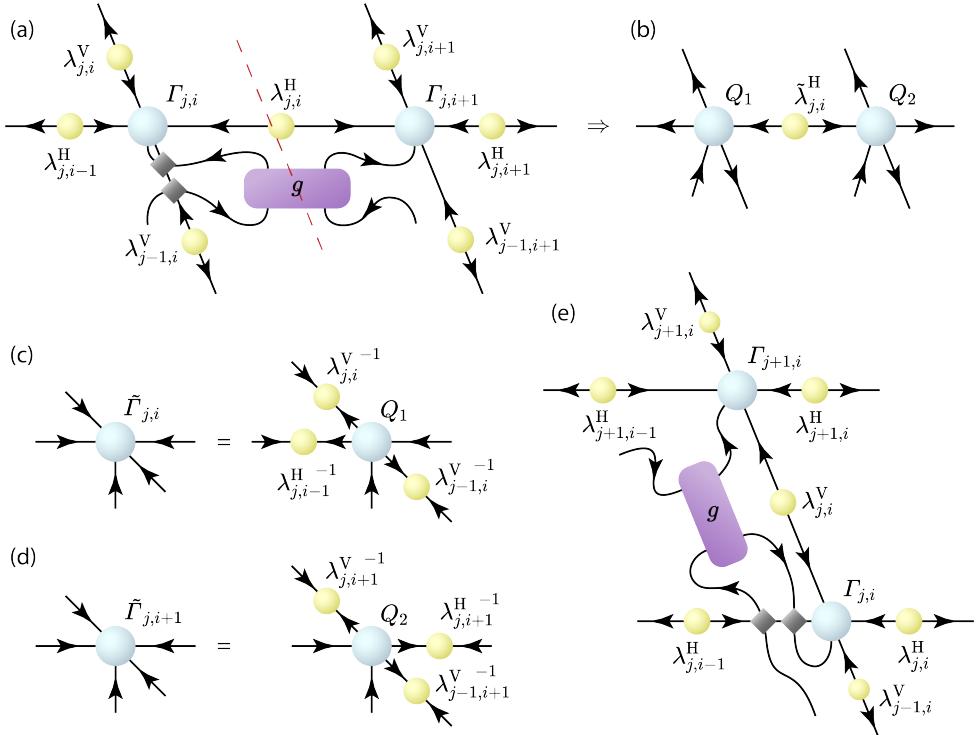


Fig. 2.28 Nearest neighbor simple update. (a) Application of Trotter gate g to a horizontal bond. (b) An SVD gives an updated singular value matrix and two tensors. (c)(d) Unmount the environmental singular value matrices. (e) Simple update for vertical bond.

As shown in Figure 2.28, the Trotter gate is applied to some horizontal bond with surrounding singular value matrices contracted as the environment. Next, an SVD

is performed to give two tensors Q_1 and Q_2 and an updated singular value matrix $\tilde{\lambda}_{j,i}^H$. Then, detach the environment by contracting the inverse singular value matrices from Q_1 and Q_2 to retrieve the updated tensors $\tilde{I}_{j,i}$ and $\tilde{I}_{j,i+1}$. The simple update of vertically aligned bonds can be conducted in an analogous way.

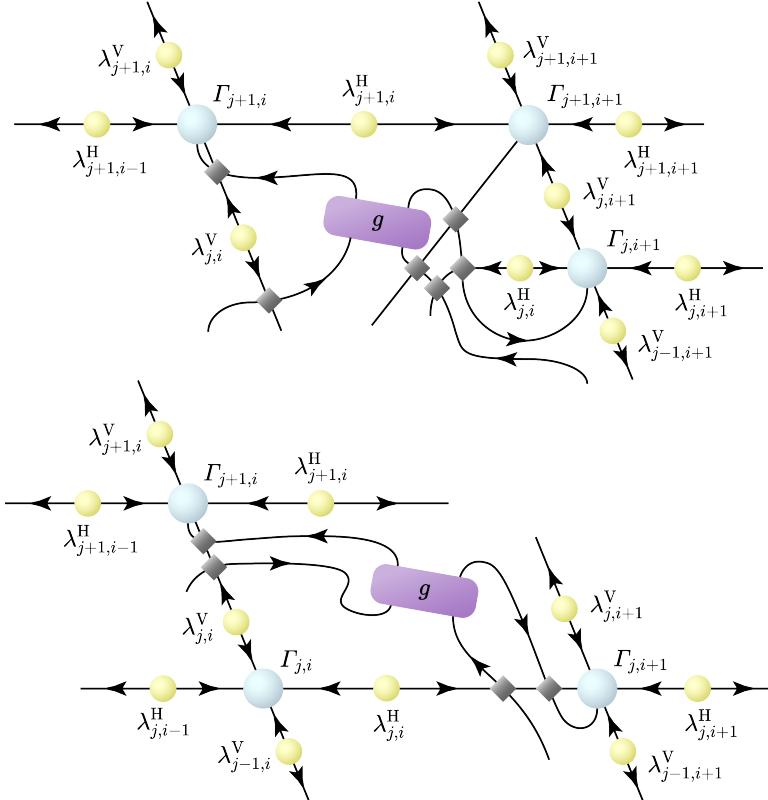


Fig. 2.29 Simple update diagrams for next-nearest neighbor terms connecting site $(j, i+1)$ and site $(j+1, i)$ (anti-diagonal terms). A third site must be passed through and two diagrams passing different assisting site are used for a symmetric update.

The simple updates of next-nearest neighbor terms [7] are much more complicated. There are in general two types of next-nearest neighbor terms — anti-diagonal terms (connecting site (j, i) and site $(j+1, i+1)$) and diagonal terms (connecting site $(j, i+1)$ and site $(j+1, i)$). As there are no bonds directly connect next-nearest neighbor sites, one has to pass through a third assisting site. For each type of next-nearest neighbor terms, there are two possible choices of the assisting site. Therefore, there are in total four diagrams for the next-nearest neighbor simple update.

Figure 2.29 shows the two diagrams for updating anti-diagonal bonds. After applying the Trotter gate, two SVDs are performed to separate the large tensor into three tensors and two singular value matrices [7]. Notice that when both diagrams are included in one single update step, the Trotter gate has to be squared-rooted, i.e. $g = \exp\{-\beta\mathcal{H}/2\}$.

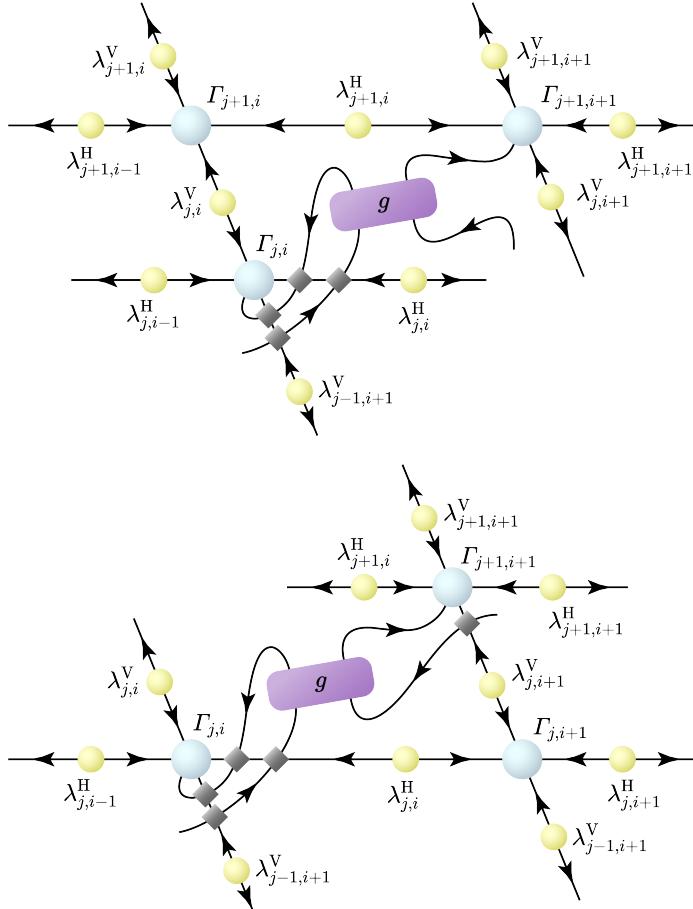


Fig. 2.30 Simple update diagrams for next-nearest neighbor terms connecting site (j, i) and site $(j + 1, i + 1)$ (diagonal terms). A third site must be passed through and two diagrams passing different assisting site are used for a symmetric update.

Figure 2.30 shows the two diagrams for updating diagonal bonds, which is completely analogous to the anti-diagonal case. The fact that two diagrams are computed in each next-nearest neighbor simple update step dramatically increases the numeri-

cal costs. Moreover, utilizing singular values as the environment is also less optimal in the next-nearest neighbor case, which produces a slightly less accurate result compared with the nearest-neighbor update.

2.7.2 Full Update

Compared with simple update, (fast) full update scheme [10] takes the corner and transfer matrices generated by CTMRG as the environment and are therefore more accurate while numerically more expensive. Only nearest-neighbor full update is considered here.

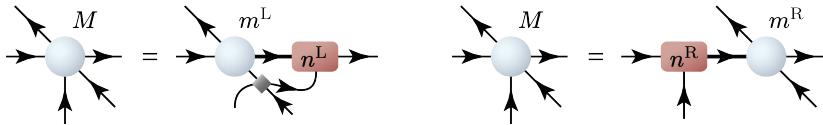


Fig. 2.31 Separation of physical leg for horizontal update. The physical leg and the bond to be updated are separated from the tensor using SVD.

The goal of the full update is to find an optimal truncation from \tilde{M} to M after applying the Trotter gate. This is achieved by finding M to minimize $\|\tilde{M} - M\|$, where $|\tilde{M}\rangle$ and $|M\rangle$ are states before and after the truncation. However, manipula-

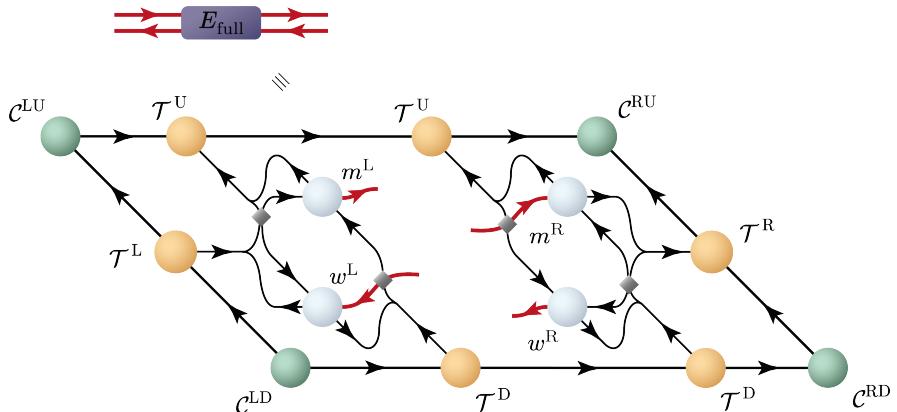


Fig. 2.32 Construction of horizontal full environment E_{full} for the rank-3 tensors n^L and n^R . Vertical environment can be constructed analogously.

tions directly on tensor M would be inconvenient. A more practical technique is to separate M using SVD into an isometry m^L (or m^R) and a rank-3 tensor n^L and n^R containing the physical leg and the bond to be updated, as shown in Figure 2.31.

After the separation, one can construct the environment of the rank-3 tensors as shown in Figure 2.32. Notice that tensor w^L and w^R are the conjugate of isometries m^L and m^R respectively subject to a twist of legs analogous to Figure 2.16.

In terms of n^L and n^R , the optimal truncation turns into minimization of the cost function (where \tilde{n} and n denotes tensors before and after the truncation)

$$d(\tilde{n}^L, \tilde{n}^R, n^L, n^R) = \| |\tilde{n}^L, \tilde{n}^R\rangle - |n^L, n^R\rangle \| . \quad (2.12)$$

The norm is quadratic in n^L and n^R , and thus can be optimized using the alternating least-square algorithm. Concretely, one first optimizes n^L with fixed n^R by asking the partial derivative to be zero, as shown in Eq. (2.13) and Figure 2.33. Then, optim-

$$\frac{\partial}{\partial u^L} d(\tilde{n}^L, \tilde{n}^R, n^L, n^R) = 0 \Rightarrow n^L X = Y \quad (2.13)$$

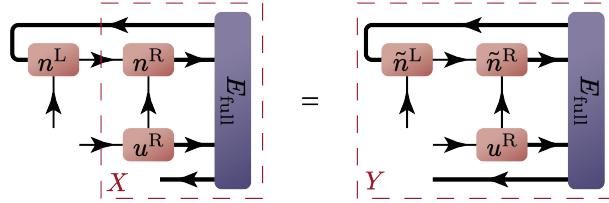


Fig. 2.33 Diagrams for optimization of tensor n^L with n^R fixed, where u^L and u^R is the Hermitian conjugate of n^L and n^R respectively.

ize n^R in a similar manner, as shown in Eq. (2.14) and Figure 2.34. This two procedures are iterated until the fidelity $f = |d_{i+1} - d_i|/d_0$ [37] where d_i is the value of cost function at the i th iteration drops below a desired value.

$$\frac{\partial}{\partial u^R} d(\tilde{n}^L, \tilde{n}^R, n^L, n^R) = 0 \Rightarrow X n^R = Y \quad (2.14)$$

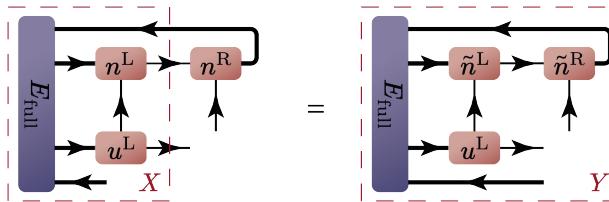


Fig. 2.34 Diagrams for optimization of tensor n^R with n^L fixed, where u^L and u^R is the Hermitian conjugate of n^L and n^R respectively.

The full update requires computing the environment tensors through CTMRG in every update step, which unavoidably consumes enormous computational resources. Fast full update runs only one CTMRG step after each update considering the change of tensors is small after an update step, and hence is more numerically efficient.

2.8 Observable Measurement

Measurement of observables can be achieved after obtaining the ground state and the corresponding corner and transfer matrices. Figure 2.35 and Figure 2.36 exemplifies the measurement of nearest-neighbor and next-nearest neighbor two-site observables.

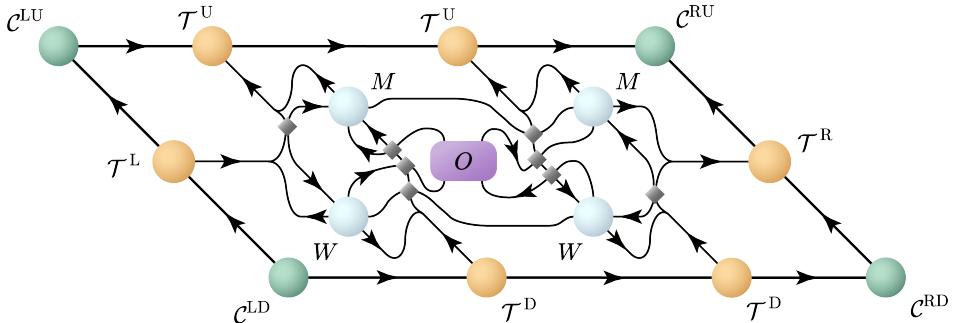


Fig. 2.35 Measurement of nearest-neighbor horizontal two-site observables. Two-site observables concerning vertically aligned sites can be measured in an analogous manner.

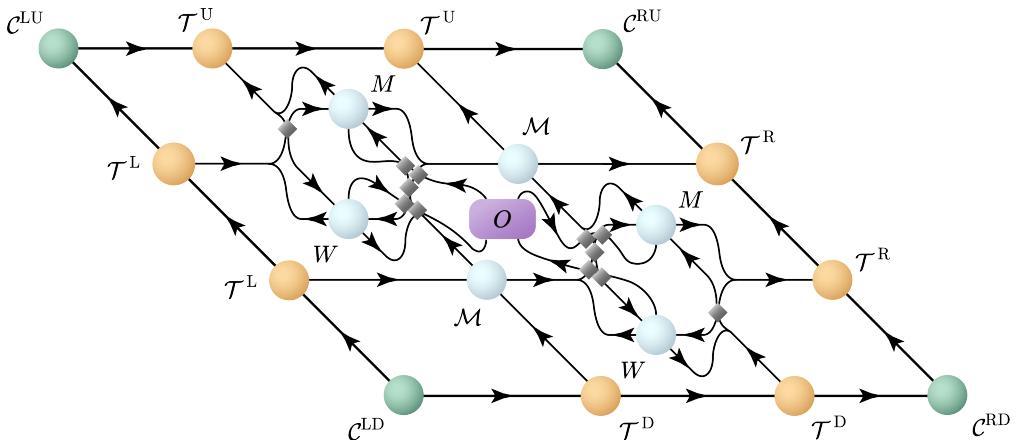


Fig. 2.36 Measurement of next-nearest neighbor anti-diagonal two-site observable. Two-site observables concerning diagonally aligned sites can be measured in an analogous manner.

Usually, the physical legs of the related sites need to cross other lines multiple times, which results in numerous swap gates in the fermionic case. These swap gates need extra attention in the implementation.

Chapter 3

Simulation Results

The quantum lattice models studied in this thesis include the Heisenberg model, a free-fermion model and the Hubbard model, all with nearest-neighbor and next-nearest neighbor interaction. This chapter presents the results of the simulations. Section 3.1 discusses the general strategies for finding the ground state and computing observables. Section 3.2 presents the simulation results for the nearest-neighbor and next-nearest neighbor Heisenberg model. They serve as sanity checks of the iPEPS program by comparing our results with published ones. Section 3.3 presents simulations of an exactly solvable free-fermion model, which verifies the implementation of the fermionic iPEPS method. Section 3.4 presents simulations of the nearest-neighbor and next-nearest neighbor Hubbard model. The nearest-neighbor Hubbard model is used to confirm the implementation of spinful fermions. New discoveries from next-nearest neighbor Hubbard model provide new clues on the spin and charge orders of the ground state near 1/8 hole doping. Analyses show that an SU(2) symmetric uniform state can be lower in energy than the U(1) symmetric stripe states.

3.1 General Strategies

This section describes technical details about the initial setup and parameter choices that are used for the computation of all models. Unless specifically pointed out, the parameter settings in this chapter follow the description of this section.

Initialization. All computations start with random initialization. In symmetric tensor networks, it is often beneficial for the initial tensors to have several different combinations of quantum numbers. A useful practice is to construct an isometry first and assign a random number in the range $[0, 1]$ to each data entry.

Simple Update. The Trotter error is of order $O(\tau^2)$. Therefore, high accuracy requires a small (imaginary) time step. However, a large number of iterations may be needed to reach convergence if one begins with a tiny time step. A useful strategy is to

start with a larger time step and gradually reduce it as the iteration goes. Concretely, a fixed, large initial time step is used first, as the random initial state is usually quite far away from the ground state. In practice, $\tau = 0.1$ is found to be a very good choice as an initial time step for all models studied. This initial time step is kept fixed for usually dozens of iterations until the change of the average spectrum $\Delta\bar{\Lambda}$ drops below a threshold, e.g. 10^{-4} . Then, one changes the time step depending on the value of $\Delta\bar{\Lambda}$. A useful strategy is to reduce the time step by half when $\Delta\bar{\Lambda}$ drops below $K\tau^2$ where K is some constant, because the Trotter error is of order τ^2 . Empirically, $K = 1$ is a good choice in most cases. In special cases when the size of the unit cell is large, one may need a smaller value of K . Simple update stops when τ drops below a threshold value of e.g. 10^{-5} .

CTMRG. The environment bond dimension χ is set based on the iPEPS bond dimension D (or the number of multiplets kept D^*) and the symmetry. In practice, $\chi = 10D$ for U(1) symmetric iPEPS and $\chi = 5D^*$ for SU(2) symmetric iPEPS are normally enough for a good measurement of observables. The average spectrum of the corner and transfer matrices are monitored and the renormalization stops after its change drops below a threshold of e.g. 10^{-6} .

Full Update. The convergence properties of full update can be dramatically improved by utilizing the ground state obtained by simple update as initialization. As the ground state from simple update is already close to the exact ground state, the initial time step of full update can be smaller, e.g. $\tau = 0.0125$. Similarly, full update stops when $\Delta\bar{\Lambda}$ drops below a threshold value of e.g. 10^{-5} .

Bond Dimension. Starting directly with a large bond dimension is in most cases not an optimal strategy, as the optimization algorithms suffer from the local minimum problem. Therefore, a common practice is to start with $D = 2$ (or $D^* = 2$) and increment the bond dimension by 1 after convergence.

Example. (i) Randomly initialize the tensors. Set $D = 2$ or $D^* = 2$. (ii) Run simple update with $\tau = 0.1$ until $\Delta\bar{\Lambda} < 10^{-4}$. (iii) Run simple update with initial $\tau = 0.1$ and reduce τ by half when $\Delta\bar{\Lambda} < \tau^2$. Stop when $\tau < 10^{-5}$. (iv) Run CTMRG. (v) Run fast full update with initial $\tau = 0.0125$ and reduce τ by half when $\Delta\bar{\Lambda} < \tau^2$. Stop when $\tau < 10^{-5}$. (vi) Run CTMRG. (vii) Measure observables. (viii) Increment D or D^* by 1 and go to (ii). (ix) Finish computation when reaching desired accuracy or bond dimensions.

3.2 Heisenberg Model

The Heisenberg model is one of the most widely-studied quantum lattice models. Particularly, nearest-neighbor Heisenberg model has received accurate quantum Monte Carlo simulations, and the result can be used to testify our algorithms.

In this section, we focus on spin-1/2 Heisenberg model. Section 3.2.1 presents the benchmark of several iPEPS algorithms using nearest-neighbor Heisenberg model by comparing with Monte Carlo simulations. Section 3.2.2 investigates typical phases

of next-nearest neighbor Heisenberg model when next-nearest neighbor interaction and magnetic frustrations come into play, and verifies the capability of simple update to detect different phases.

3.2.1 Nearest-Neighbor Heisenberg Model

Nearest-neighbor Heisenberg model is the simplest type of Heisenberg model with only nearest-neighbor interaction terms. The Hamiltonian used in this section is

$$\mathcal{H} = \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j, \quad (3.1)$$

where we set $J = 1$ in Eq. (1.1) for simplicity. Figure 3.1 gives the deviation of the computed ground state energy to the reference value $E_g = -0.669437$ [38][39].

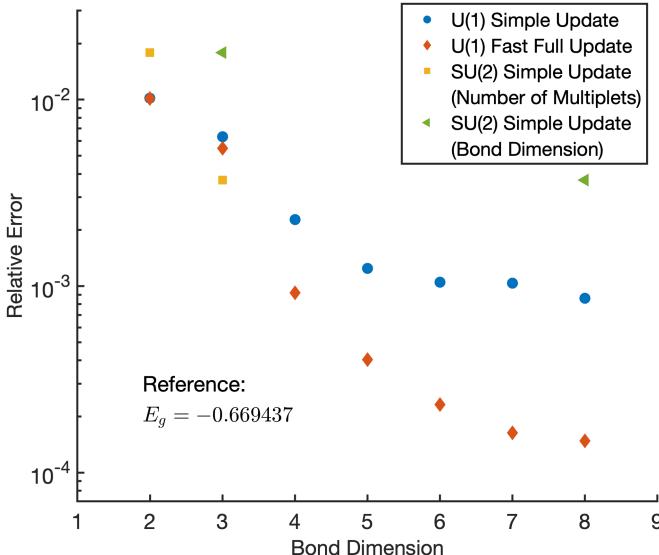


Fig. 3.1 Ground state energy of nearest-neighbor Heisenberg model obtained using different symmetries and update schemes. Symbols show the deviation to the reference value $E_g = -0.669437$ [38][39].

It can be seen from the figure that the simple update with U(1) spin symmetry produces a relative error of order 10^{-3} , while full update provides better accuracy of order 10^{-4} . However, as the full update consumes much more computational resources while it leads to solely a tiny improvement of accuracy, this will be the only full update simulation in this thesis.

Also, the ground state energy from the simulation with SU(2) spin symmetry is higher than that with only U(1) symmetry, as shown in Figure 3.1. Therefore, the results support an SU(2) symmetry breaking in the ground state of nearest-neighbor Heisenberg model, consistent with the fact that its ground state is known to be an Néel anti-ferromagnet.

3.2.2 Next-nearest Neighbor Heisenberg Model

The next-nearest neighbor Heisenberg model (also named J_1 - J_2 Heisenberg model) introduces next-nearest neighbor interactions. The Hamiltonian used here is

$$\mathcal{H} = \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j + J_2 \sum_{\langle\langle i,j \rangle\rangle} \mathbf{S}_i \cdot \mathbf{S}_j, \quad (3.2)$$

where again, we set $J_1 = 1$ in Eq. (1.2) for simplicity. J_1 - J_2 Heisenberg model incorporates several different phases as the first term favours an anti-ferromagnetic order in nearest neighbors and the second term favours an anti-ferromagnetic order in next-nearest neighbors. Thus, different values of J_2 lead to different orders. [40]

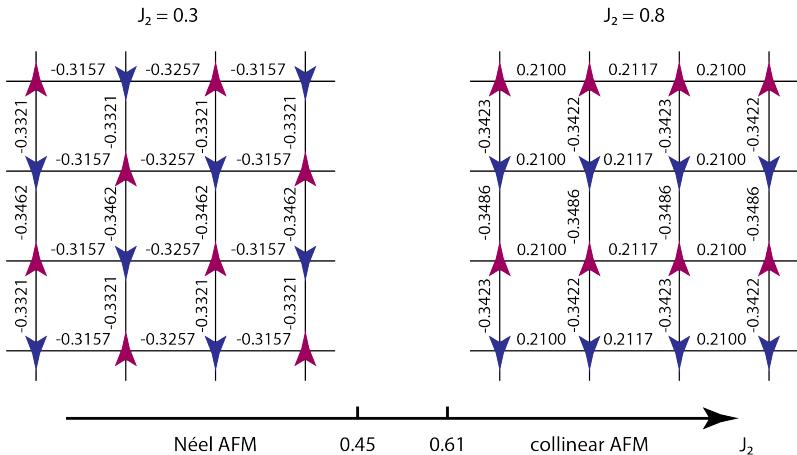


Fig. 3.2 Néel anti-ferromagnetic phase and collinear anti-ferromagnetic phase at small and large J_2 , simulated with U(1) symmetry and $D = 6$. The direction of arrows labels the direction of the spin. Numbers near the bonds give the corresponding bond energy. Phase boundaries are taken from [41].

Figure 3.2 shows the two phases at small and large J_2 simulated with U(1) symmetry and $D = 6$. For small J_2 , the first term in Eq. (3.2) dominates and the ground state still shows an Néel anti-ferromagnetic order, similar to the nearest

neighbor Heisenberg model. For large J_2 , the second term in Eq. (3.2) dominates and the ground state turns into a collinear anti-ferromagnetic phase, where anti-ferromagnetic order appears in next-nearest neighbors.

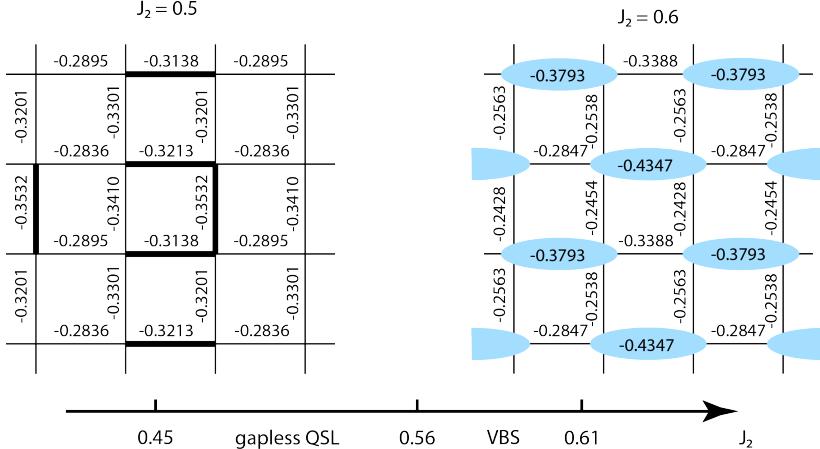


Fig. 3.3 Quantum spin liquid phase and valence bond solid phase near the transition region, simulated with U(1) symmetry and $D = 6$. Thick lines and blue ovals indicate larger bindings and valence bonds. Numbers near the bonds give the corresponding bond energy. Phase boundaries are taken from [41].

Figure 3.3 shows the two phases at transition region simulated with U(1) symmetry and $D = 6$. A smaller J_2 gives a quantum spin liquid phase and a larger J_2 produces a valence bond solid phase. The two phases are still directly visible even though they only occupy a small region in the entire phase space.

The simulation of next-nearest neighbor Heisenberg model verifies that although simple update is less accurate than full update, it is still capable to identify different phases of the ground state. Therefore, in cases where the phases and orders are more significant than the accuracy, simple update is already enough to reveal crucial information.

3.3 Free-Fermion Model

The free-fermion model is an analytically solvable model with only the hopping and pairing of a single channel of spinless fermions. The fact that it is exactly solvable makes the free-fermion model a perfect choice for benchmarking our fermionic iPEPS algorithms. The simplest choice is to include only the hopping terms. However, this results in a highly critical system whose ground state possesses a logarithmic correction to the entanglement entropy and hence violates the area law. Such violation

can not be captured by the CTMRG procedure in the iPEPS algorithm. Therefore, a pairing term is necessary to make the system less critical.

In this section, we directly aim at the next-nearest neighbor free-fermion model. The Hamiltonian studied can be written as

$$\begin{aligned} \mathcal{H} = & - \sum_{\langle i,j \rangle} [c_i^\dagger c_j + c_j^\dagger c_i + \gamma(c_i^\dagger c_j^\dagger + c_j c_i)] \\ & - t_2 \sum_{\langle\langle i,j \rangle\rangle} [c_i^\dagger c_j + c_j^\dagger c_i + \gamma(c_i^\dagger c_j^\dagger + c_j c_i)] + \mu \sum_i c_i^\dagger c_i, \end{aligned} \quad (3.3)$$

where γ is the pairing amplitude and μ the chemical potential. Here, we set $\gamma = 1$ in all our computations. Analytic diagonalization can be achieved by Bogoliubov transformation as described in [42].

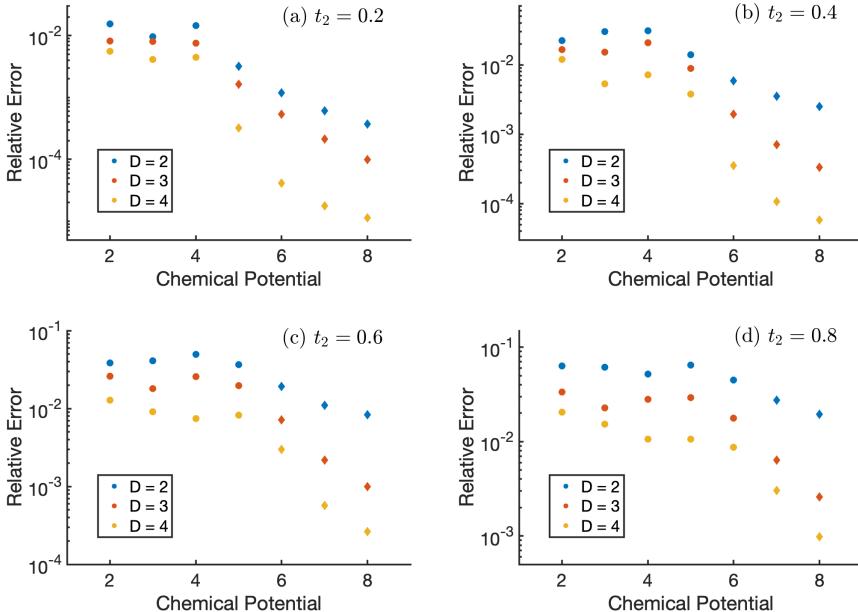


Fig. 3.4 Relative errors of the ground state energy of the free-fermion model compared with the exact solutions. Circle and diamond symbols label critical and gapped phases respectively.

The errors of the ground state energy with different chemical potential and bond dimensions relative to the exact diagonalization are shown in Figure 3.4. In most cases, errors at $D = 4$ drops below 10^{-2} , confirming the validity of the simulation. Besides, the errors grow as t_2 increases. This is the expected artifact of the inaccuracy of next-nearest neighbor simple update.

Theoretically, free-fermion model incorporates a critical and a gapped phase. At $t_2 = 0$, the phase boundary lies exactly at $\mu = 4$, while at positive t_2 , the boundary will move towards a larger μ . The two phases are labelled in Figure 3.4 using circle and diamond symbols. It can be seen that the accuracy of simulation in the gapped phase is much higher than that in the critical phase. This indicates a slight violation of the area law of entanglement entropy in the critical phase.

3.4 Hubbard Model

Hubbard model is one of the most famous effective models for strongly correlated systems, which is considered to capture key ingredients of high- T_c superconductors. The nearest-neighbor Hubbard model has been shown to possess a Mott insulating phase at half-filling and is believed to generate a strange metallic phase and superconductivity at finite hole doping.

In this section, we focus on the Hubbard model with a finite next-nearest neighbor hopping amplitude. The nearest-neighbor Hubbard model will be studied first in section 3.4.1 and compared with existing results as a benchmark of the algorithm. Then, section 3.4.2 presents new data for next-nearest neighbor Hubbard model where controversy turned up recently [8][43].

3.4.1 Nearest-Neighbor Hubbard Model

The nearest-neighbor Hubbard model includes only a nearest-neighbor hopping term and an on-site interaction term. An additional chemical potential term may be invoked to control the doping. The Hamiltonian used in this section is

$$\mathcal{H} = - \sum_{\langle i,j \rangle, \sigma} (c_{i\sigma}^\dagger c_{j\sigma} + c_{j\sigma}^\dagger c_{i\sigma}) + U \sum_i n_{i\uparrow} n_{i\downarrow} + \mu \sum_i n_i, \quad (3.4)$$

where we have set $t = 1$ in Eq. (1.3), μ is the chemical potential and n_i the particle number operator. The nearest-neighbor Hubbard model at half filling has been studied by a variety of numerical methods, including Auxiliary Field Quantum Monte-Carlo (AFQMC) method, Density Matrix Embedding Theory (DMET) and Density Matrix Renormalization Group (DMRG) [44]. Therefore, we can benchmark our algorithms by comparing with existing results.

Figure 3.5, 3.6, 3.7 and 3.8 show the ground state energy, the double occupancy and the local moment of the nearest-neighbor Hubbard model simulated on a 2×2 supercell with $U = 0$, $U = 4$, $U = 8$ and $U = 12$ respectively. The chemical potential μ is set to $-U/2$ to ensure an explicit particle-hole symmetry and hence a half filling. The $U = 0$ case is equivalent to an exactly solvable two-channel free-fermion model, and thus can be cross-checked with the free-fermion simulations.

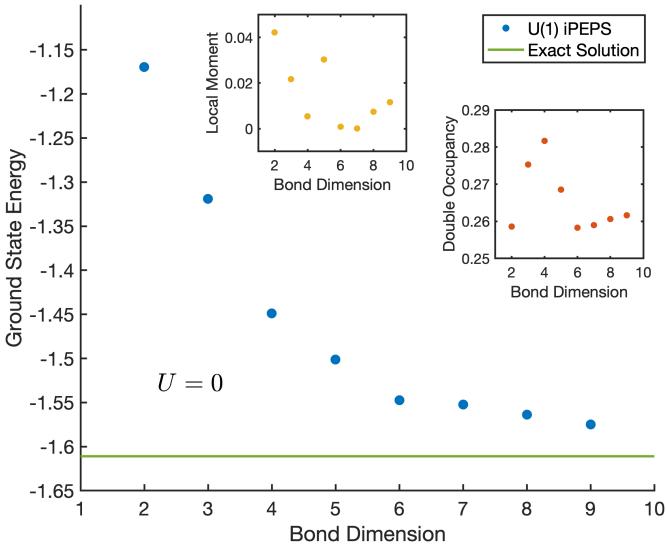


Fig. 3.5 Ground state energy, double occupancy and local moment of nearest-neighbor Hubbard model with $U = 0$ obtained using $U(1)$ symmetry. Reference value $E_g = -0.16211$ comes from the exact solution [45].

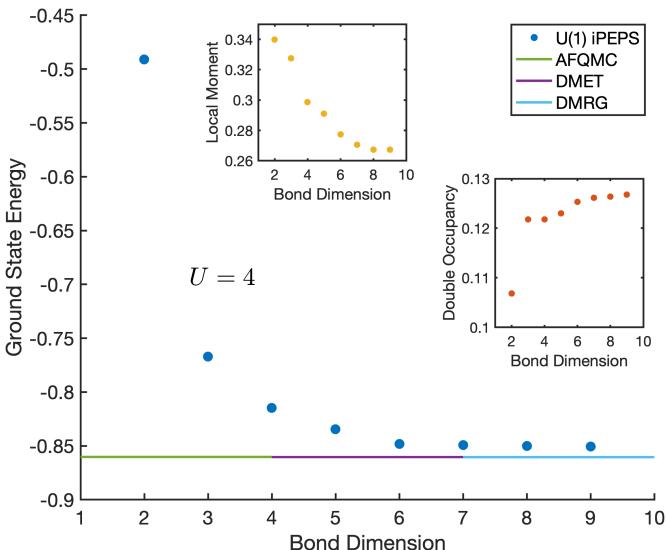


Fig. 3.6 Ground state energy, double occupancy and local moment of nearest-neighbor Hubbard model with $U = 4$ obtained using $U(1)$ symmetry. Reference values come from AFQMC, DMET and DMRG [44].

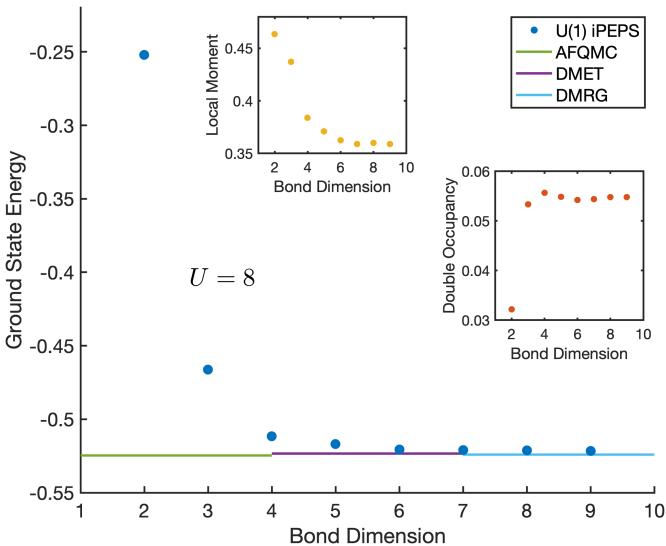


Fig. 3.7 Ground state energy, double occupancy and local moment of nearest-neighbor Hubbard model with $U = 8$ obtained using $U(1)$ symmetry. Reference values come from AFQMC, DMET and DMRG [44].

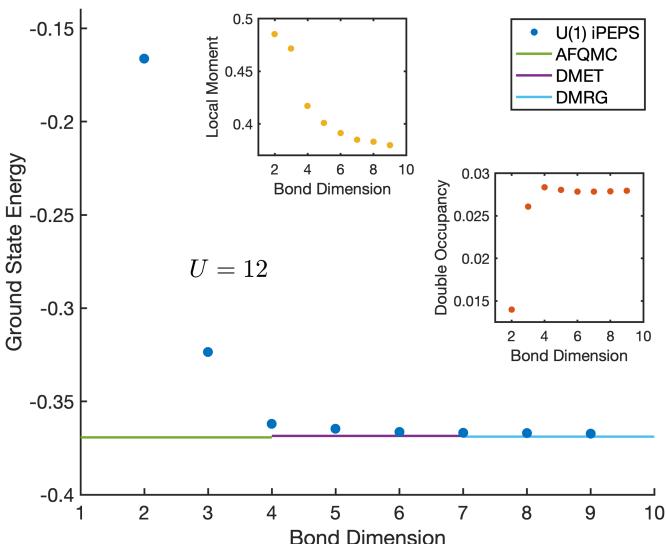


Fig. 3.8 Ground state energy, double occupancy and local moment of nearest-neighbor Hubbard model with $U = 12$ obtained using $U(1)$ symmetry. Reference values come from AFQMC, DMET and DMRG [44].

It can be seen that the double occupancy decreases and the local anti-ferromagnetic moment is enhanced as U increases. This is the expected behaviour of a rising on-site interaction. Also, the accuracy of the simulation is better with a larger value of U . This indicates a less critical ground state compared with free fermions.

3.4.2 Next-Nearest Neighbor Hubbard Model

The next-nearest neighbor Hubbard model introduces next-nearest neighbor hopping terms which lead to magnetic frustration. Thus, the ground state may incorporate a variety of different orders in charge and spin. The Hamiltonian of the next-nearest neighbor Hubbard model can be written as

$$\begin{aligned} \mathcal{H} = & - \sum_{\langle i,j \rangle, \sigma} (c_{i\sigma}^\dagger c_{j\sigma} + c_{j\sigma}^\dagger c_{i\sigma}) + \mu \sum_i n_i \\ & - t_2 \sum_{\langle\langle i,j \rangle\rangle, \sigma} (c_{i\sigma}^\dagger c_{j\sigma} + c_{j\sigma}^\dagger c_{i\sigma}) + U \sum_i n_{i\uparrow} n_{i\downarrow}, \end{aligned} \quad (3.5)$$

where again $t_1 = 1$ and a chemical potential term is added to control the doping.

A consensus has been reached that the ground state of the nearest-neighbor Hubbard model at 1/8 hole doping is a stripe state with a period of 8 sites in the charge order [46]. However, the period 4 stripe typically observed in experiments [47–49] was found to have a higher energy. Therefore, it is natural to suspect that the next-nearest neighbor interaction contributes in reality.

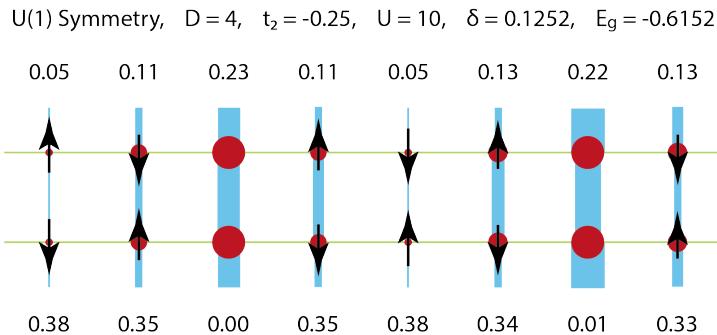


Fig. 3.9 Period 4 stripe obtained using U(1) iPEPS. Sizes of the red disk and lengths of arrows scale with the hole density and the local moment, with average values indicated in the top and bottom rows. Widths of the bond grow with the singlet pairing amplitude with two colors labelling opposite signs.

A detailed scan of the phase space using U(1) iPEPS shows that the model prefers a stripe state with period 4 in charge order and period 8 in spin order within the range $0.16 < -t_2 < 0.42$ [8]. However, another research using DMRG on a width

4 cylinder and $t_2 = -0.25$ generates a ground state with period 4 charge density wave only [43]. Therefore, we intend to investigate these two contradictory results by studying the charge and spin orders of the ground state at $t_2 = -0.25$ using U(1) and SU(2) symmetric iPEPS methods.

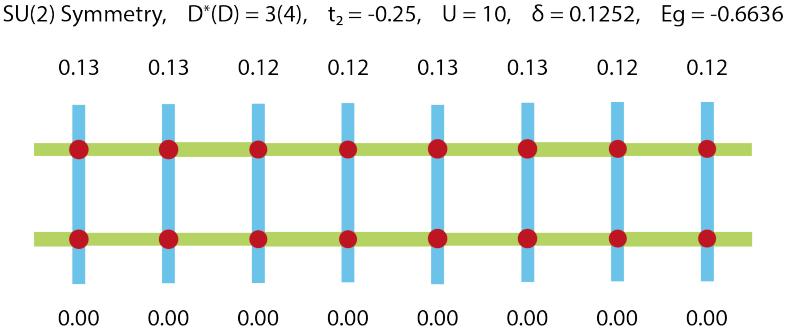


Fig. 3.10 Uniform state obtained using SU(2) iPEPS. Sizes of the red disk and lengths of arrows scale with the hole density and the local moment, with average values indicated in the top and bottom rows. Widths of the bond are proportional to the SU(2) symmetric singlet pairing amplitude with two colors labelling opposite signs.

Figure 3.9 shows a stripe state at $t_2 = -0.25$ and 1/8 doping obtained using U(1) iPEPS at a fixed bond dimension $D = 4$ with an 8×2 supercell. This ground state features a charge density wave of period 4 and an anti-ferromagnetically ordered spin density wave of period 8. Singlet pairing amplitudes $\Delta_s = (c_{i\uparrow}c_{j\downarrow} - c_{j\uparrow}c_{i\downarrow})/\sqrt{2}$ are enhanced and the local magnetic moments are suppressed near large hole densities. These characteristics generally agree with the previous findings [8].

Figure 3.10 shows the ground state also at $t_2 = -0.25$ and 1/8 doping obtained using SU(2) iPEPS at a fixed bond dimension $D = 4$ ($D^* = 3$) with a 4×2 supercell. Two supercells are included in the figure for a better comparison. This ground state is uniform with neither charge density wave nor spin density wave. A d-wave superconducting order is clearly observed by looking at the SU(2) symmetric singlet pairing amplitudes. The absence of a charge order is in contrast to the DMRG simulation [43] which found a stable charge density wave in the ground state.

An intriguing discovery is that the SU(2) symmetric uniform state in Figure 3.10 has a much lower energy $E_g = -0.6636$ than the U(1) symmetric stripe state ($E_g = -0.6152$). Therefore, the SU(2) symmetric uniform state becomes a highly competitive ground state at $t_2 = -0.25$ and 1/8 doping. Although the extrapolation with respect to bond dimensions is not possible due to the limited computation time available in this project, we want to emphasize that the value $E_g = -0.6636$ is already close to the U(1) iPEPS extrapolation $E_g \sim -0.68$ found in [8].

In iPEPS simulations, doping is tuned utilizing chemical potential. Therefore, it is in general not possible to accurately fix the doping on a target value. Hence, to obtain an accurate estimate of the ground state energy at 1/8 doping, five states are

computed with a hole doping near 1/8 and the ground state energy at exactly 1/8 doping can be extracted by a linear fitting. The U(1) iPEPS with 8×2 supercell suggests a ground state energy of $E_g = -0.6140$ and the SU(2) iPEPS with 4×2 supercell gives $E_g = -0.6650$, as indicated in Figure 3.11. Also, as the SU(2) symmetric iPEPS produces a uniform state which should be irrelevant to the size of the supercell, simulations with a 2×2 supercell are carried out as a cross verification.

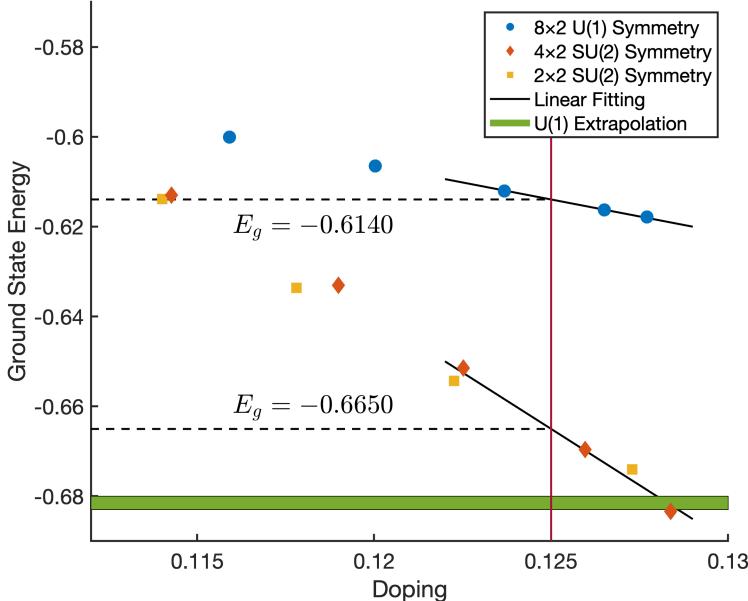


Fig. 3.11 Comparison of U(1) and SU(2) symmetric ground state energy. Green shaded region labels the extrapolation of U(1) symmetric iPEPS [8]. The SU(2) symmetric uniform states are lower in energies than the U(1) symmetric period 4 stripes, and are already close to the extrapolated lower bound in the U(1) case. Linear fitting gives the ground state energies at exactly 1/8 doping. SU(2) simulations with 4×2 and 2×2 supercells produce similar outcomes, confirming the uniformity of the SU(2) symmetric ground state.

Figure 3.12 shows various observables measured in U(1) and SU(2) iPEPS simulations. The double occupancy of the SU(2) symmetric uniform states are much larger than that of the U(1) stripes. The SU(2) symmetry suppresses the local moments and thus no magnetization is observed in the SU(2) symmetric states, while U(1) symmetric stripes host an anti-ferromagnetic order and hence a finite magnetization. Also, the average singlet pairings in SU(2) symmetric ground states are much stronger than in the U(1) cases. This suggests a possible superconducting ground state near 1/8 doping.

In Figure 3.12(d), we investigate the contribution of nearest-neighbor energies and next-nearest neighbor energies in U(1) and SU(2) symmetric ground states. It can

be seen that the average next-nearest neighbor energies (around -0.01) are heavily suppressed and the nearest-neighbor terms dominate in the U(1) symmetric stripes. This is within expectation as the anti-ferromagnetic orders in stripes compete against the next-nearest neighbor hopping. By contrast, the average next-nearest neighbor energies (around -0.1) in SU(2) symmetric uniform state are much lower as no spin orders exist under the SU(2) symmetry.

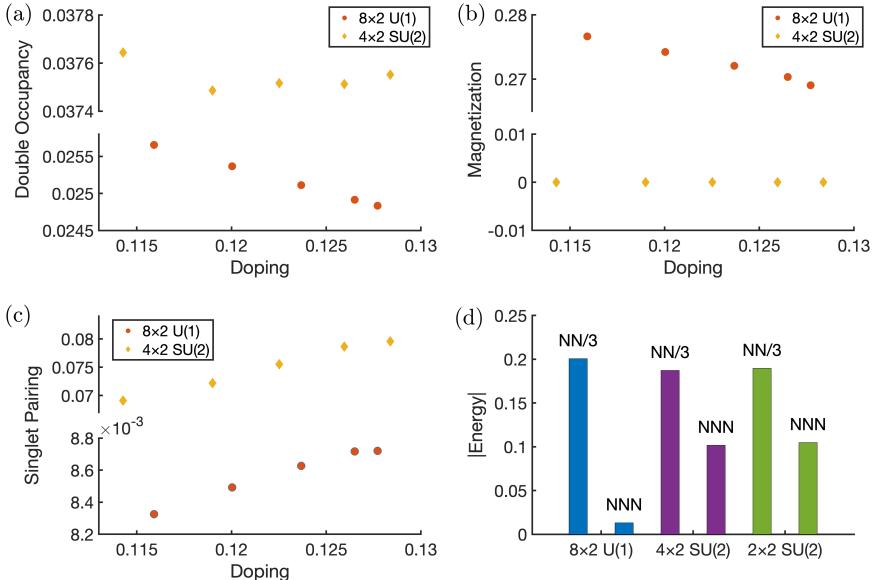


Fig. 3.12 Various observables of U(1) and SU(2) symmetric ground state: (a) double occupancy vs doping; (b) avarage magnetization vs doping; (c) avrage singlet pairing amplitude vs doping; (d) contribution of nearest-neighbor (NN) energies and next-nearest neighbor (NNN) energies. The nearest-neighbor energies are divided by 3 for a better visual presentation.

The discovery of a uniform d-wave superconducting state is basically within expectation. Numerous previous investigations on nearest-neighbor Hubbard model [44][46] or t - J model [50][35][51] (viewed as a descendant of the Hubbard model) have already probed a uniform superconducting state whose energy is merely slightly higher than the stripes [52]. Thus, it is not surprising that this uniform state survives an additional next-nearest neighbor hopping term. However, the next-nearest neighbor interactions lead to magnetic frustrations in the stripes and hence dramatically raise the energy. This provides a possibility for the SU(2) symmetric uniform state to be the ultimate ground state, as it incorporates no spin orders and is therefore free from any magnetic frustrations.

In summary, simulations using SU(2) symmetric iPEPS algorithms present a highly competitive ground state with SU(2) symmetry for the next-nearest neighbor Hubbard model at $t_2 = -0.25$ and $1/8$ hole doping. The energy of this SU(2) sym-

metric ground state at a rather small bond dimension $D=4$ is already approaching the lower bound of the U(1) symmetric counterparts. The newly discovered ground state is uniform without any charge and spin orders. The lack of anti-ferromagnetic orders allows much lower next-nearest neighbor energies and thus produces a lower energy overall.

Chapter 4

Conclusions

In this thesis, we employ the symmetric iPEPS methods to simulate several 2-dimensional quantum lattice models with nearest-neighbor and next-nearest neighbor interactions. The PEPS tensor network makes use of a network of rank-5 tensors with one physical index representing the local state and four indices connecting neighboring tensors which encodes the entanglement. Infinite PEPS utilizes translational invariance and is able to tackle infinite-size systems using a supercell of a finite number of tensors. The CTMRG scheme encodes an infinitely-extended environment using a finite number of corner and transfer matrices. Fermionic statistics are captured by considering parity preserving tensors and adding swap gates at line crossings. Symmetries of tensors are automatically tracked using the QSpace tensor library. The ground state can be obtained through algorithms based on the imaginary time evolution and truncation schemes such as the simple and full update counteract the exponential growth of the bond dimension. Observables can be measured after the ground state and corresponding corner and transfer matrices have been computed.

Simulations of the nearest-neighbor Heisenberg model benchmarks the simple update scheme and the fast full update scheme. The simple update scheme produces a ground state with a relative error of order 10^{-3} while the fast full update scheme achieves an accuracy of order 10^{-4} . Also, U(1) and SU(2) symmetric simulations confirm the ground state to be an Néel anti-ferromagnet.

Simulations of next-nearest neighbor Heisenberg model confirm the capability of the simple update scheme to capture distinct physical phases. The computations with $J_2 = 0.3$, $J_2 = 0.5$, $J_2 = 0.6$ and $J_2 = 0.8$ generate ground states being a Néel anti-ferromagnet, a quantum spin liquid, a valence bond solid and a collinear anti-ferromagnet respectively.

Simulations of an exactly solvable free-fermion model and the nearest-neighbor Hubbard model confirm the validity of the implementation of fermionic iPEPS. The accuracy is satisfactory even when the ground state of the system is critical and slightly violates area law.

Simulations of next-nearest neighbor Hubbard model with next-nearest neighbor hopping amplitude $t_2 = -0.25$ at 1/8 hole doping produce a U(1) symmetric ground state with a period 4 in the charge order and a period 8 in the spin order, as

well as an SU(2) symmetric uniform ground state with a d-wave superconducting order but no charge or spin orders. This is in contrast to the previous DMRG results. Remarkably, the SU(2) symmetric uniform state has a much lower energy than the U(1) symmetric stripe at fixed bond dimension $D=4$. However, current simulations are insufficient to draw solid conclusions as the extrapolation with respect to bond dimensions is not yet available. The low energy of the SU(2) symmetric uniform state presumably comes from the avoidance of anti-ferromagnetic orders, which allows much lower next-nearest neighbor energies.

References

- [1] M. Imada, A. Fujimori, and Y. Tokura, *Metal-insulator transitions*, Rev. Mod. Phys., **70**, 1039 (1998).
- [2] P. W. Anderson, *The Theory of Superconductivity in the High-Tc Cuprate Superconductors*, Princeton University Press (1997).
- [3] E. Dagotto, *Correlated electrons in high-temperature superconductors*, Rev. Mod. Phys., **66**, 763 (1994).
- [4] A. P. Ramirez, *Strongly Geometrically Frustrated Magnets*, Annu. Rev. Mater. Sci., **24**, 453 (1994).
- [5] H. L. Stormer, *Nobel Lecture: The fractional quantum Hall effect*, Rev. Mod. Phys., **71**, 875 (1999).
- [6] P. Corboz, R. Orús, B. Bauer, and G. Vidal, *Simulation of strongly correlated fermions in two spatial dimensions with fermionic projected entangled-pair states*, Phys. Rev. B, **81**, 165104 (2010).
- [7] P. Corboz, J. Jordan, and G. Vidal, *Simulation of fermionic lattice models in two dimensions with projected entangled-pair states: Next-nearest neighbor Hamiltonians*, Phys. Rev. B, **82**, 245119 (2010).
- [8] B. Ponsioen, S. S. Chung, and P. Corboz, *Period 4 stripe in the extended two-dimensional Hubbard model*, Phys. Rev. B, **100**, 195141 (2019).
- [9] F. Verstraete and J. I. Cirac, *Renormalization algorithms for Quantum-Many Body Systems in two and higher dimensions* (2004), arXiv:cond-mat/0407066.
- [10] J. Jordan, R. Orús, G. Vidal, F. Verstraete, and J. I. Cirac, *Classical Simulation of Infinite-Size Quantum Lattice Systems in Two Spatial Dimensions*, Phys. Rev. Lett., **101**, 250602 (2008).
- [11] W. Heisenberg, *Zur theorie des ferromagnetismus*, Z. Phys., **49**, 619 (1928).
- [12] J. Hubbard, *Electron correlations in narrow energy bands V. A perturbation expansion about the atomic limit*, Proc. R. Soc. London A, **296**, 82 (1967).

- [13] D. J. Scalapino, *The 2D Hubbard Model and the High Tc Cuprate Problem*, J. Supercond. Nov. Magn., **19**, 195 (2006).
- [14] A. Weiße and H. Fehske, *Exact Diagonalization Techniques, Computational Many-Particle Physics*, Springer Berlin Heidelberg, 529–544 (2008).
- [15] G. Sugiyama and S. Koonin, *Auxiliary field Monte-Carlo for quantum many-body ground states*, Ann. Phys., **168**, 1 (1986).
- [16] E. Gull, A. J. Millis, A. I. Lichtenstein, A. N. Rubtsov, M. Troyer, and P. Werner, *Continuous-time Monte Carlo methods for quantum impurity models*, Rev. Mod. Phys., **83**, 349 (2011).
- [17] W. M. C. Foulkes, L. Mitas, R. J. Needs, and G. Rajagopal, *Quantum Monte Carlo simulations of solids*, Rev. Mod. Phys., **73**, 33 (2001).
- [18] M. Troyer and U.-J. Wiese, *Computational Complexity and Fundamental Limitations to Fermionic Quantum Monte Carlo Simulations*, Phys. Rev. Lett., **94**, 170201 (2005).
- [19] K. G. Wilson, *The renormalization group: Critical phenomena and the Kondo problem*, Rev. Mod. Phys., **47**, 773 (1975).
- [20] R. Bulla, T. A. Costi, and T. Pruschke, *Numerical renormalization group method for quantum impurity systems*, Rev. Mod. Phys., **80**, 395 (2008).
- [21] A. Weichselbaum, *Tensor networks and the numerical renormalization group*, Phys. Rev. B, **86**, 245124 (2012).
- [22] S. R. White, *Density matrix formulation for quantum renormalization groups*, Phys. Rev. Lett., **69**, 2863 (1992).
- [23] U. Schollwöck, *The density-matrix renormalization group in the age of matrix product states*, Ann. Phys., **326**, 96 (2011).
- [24] F. Verstraete, V. Murg, and J. Cirac, *Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems*, Adv. Phys., **57**, 143 (2008).
- [25] B. Bruognolo, *Tensor network techniques for strongly correlated systems*, Ph.D. thesis, Ludwig-Maximilian-Universität München (2017).
- [26] G. Vidal, *Classical Simulation of Infinite-Size Quantum Lattice Systems in One Spatial Dimension*, Phys. Rev. Lett., **98**, 070201 (2007).
- [27] G. Vidal, *Efficient Classical Simulation of Slightly Entangled Quantum Computations*, Phys. Rev. Lett., **91**, 147902 (2003).
- [28] J. Eisert, *Entanglement and tensor network states*, Model. Simul., **3** (2013), arXiv:1308.3318.
- [29] B. Bruognolo, J.-W. Li, J. v. Delft, and A. Weichselbaum, *A beginner’s guide to non-abelian iPEPS for correlated fermions*, arXiv (2020), arXiv:2006.08289.

- [30] A. Weichselbaum, *Non-abelian symmetries in tensor networks: A quantum symmetry space approach*, Ann. Phys., **327**, 2972 (2012).
- [31] M. Lubasch, J. I. Cirac, and M.-C. Bañuls, *Algorithms for finite projected entangled pair states*, Phys. Rev. B, **90**, 064425 (2014).
- [32] E. Stoudenmire and S. R. White, *Studying Two-Dimensional Systems with the Density Matrix Renormalization Group*, Annu. Rev. Condens. Matter Phys., **3**, 111 (2012).
- [33] T. Nishino and K. Okunishi, *Corner Transfer Matrix Renormalization Group Method*, J. Phys. Soc. Japan, **65**, 891 (1996).
- [34] R. Orús and G. Vidal, *Simulation of two-dimensional quantum systems on an infinite lattice revisited: Corner transfer matrix for tensor contraction*, Phys. Rev. B, **80**, 094403 (2009).
- [35] P. Corboz, T. M. Rice, and M. Troyer, *Competing States in the t - J Model: Uniform d -Wave State versus Stripe State*, Phys. Rev. Lett., **113**, 046402 (2014).
- [36] H. C. Jiang, Z. Y. Weng, and T. Xiang, *Accurate Determination of Tensor Network State of Quantum Lattice Models in Two Dimensions*, Phys. Rev. Lett., **101**, 090603 (2008).
- [37] H. N. Phien, J. A. Bengua, H. D. Tuan, P. Corboz, and R. Orús, *Infinite projected entangled pair states algorithm improved: Fast full update and gauge fixing*, Phys. Rev. B, **92**, 035142 (2015).
- [38] A. W. Sandvik, *Finite-size scaling of the ground-state parameters of the two-dimensional Heisenberg model*, Phys. Rev. B, **56**, 11678 (1997).
- [39] C. Hubig, *Abelian and non-abelian symmetries in infinite projected entangled pair states*, SciPost Physics, **5**, 047 (2018).
- [40] R. Haghshenas and D. N. Sheng, *$U(1)$ -symmetric infinite projected entangled-pair states study of the spin-1/2 square J_1 - J_2 Heisenberg model*, Phys. Rev. B, **97**, 174408 (2018).
- [41] W.-Y. Liu, S.-S. Gong, Y.-B. Li, D. Poilblanc, W.-Q. Chen, and Z.-C. Gu, *Gapless quantum spin liquid and global phase diagram of the spin-1/2 J_1 - J_2 square antiferromagnetic Heisenberg model*, arXiv (2020).
- [42] W. Li, L. Ding, R. Yu, T. Roscilde, and S. Haas, *Scaling behavior of entanglement in two- and three-dimensional free-fermion systems*, Phys. Rev. B, **74**, 073103 (2006).
- [43] H.-C. Jiang and T. P. Devereaux, *Superconductivity in the doped Hubbard model and its interplay with next-nearest hopping t'* , Science, **365**, 1424 (2019).
- [44] J. P. F. LeBlanc, A. E. Antipov, F. Becca, I. W. Bulik, G. K.-L. Chan, C.-M. Chung, Y. Deng, M. Ferrero, T. M. Henderson, C. A. Jiménez-Hoyos, E. Kozik, X.-W. Liu, A. J. Millis, N. V. Prokof'ev, M. Qin, G. E. Scuseria, H. Shi, B. Svistunov, L. F. Tocchio, I. S. Tupitsyn, S. R. White, S. Zhang,

- B.-X. Zheng, Z. Zhu, and E. Gull, *Solutions of the Two-Dimensional Hubbard Model: Benchmarks and Results from a Wide Range of Numerical Algorithms*, Phys. Rev. X, **5**, 041041 (2015).
- [45] P. Corboz, *Improved energy extrapolation with infinite projected entangled-pair states applied to the two-dimensional Hubbard model*, Phys. Rev. B, **93**, 045116 (2016).
- [46] B.-X. Zheng, C.-M. Chung, P. Corboz, G. Ehlers, M.-P. Qin, R. M. Noack, H. Shi, S. R. White, S. Zhang, and G. K.-L. Chan, *Stripe order in the underdoped region of the two-dimensional Hubbard model*, Science, **358**, 1155 (2017).
- [47] J. M. Tranquada, B. J. Sternlieb, J. D. Axe, Y. Nakamura, and S. Uchida, *Evidence for stripe correlations of spins and holes in copper oxide superconductors*, Nature, **375**, 561 (1995).
- [48] A. Mesaros, K. Fujita, S. D. Edkins, M. H. Hamidian, H. Eisaki, S.-i. Uchida, J. C. S. Davis, M. J. Lawler, and E.-A. Kim, *Commensurate $4a_0$ -period charge density modulations throughout the $\text{Bi}_2\text{Sr}_2\text{CaCu}_2\text{O}_{8+x}$ pseudogap regime*, Proc. Natl. Acad. Sci., **113**, 12661 (2016).
- [49] J. M. Tranquada, J. D. Axe, N. Ichikawa, A. R. Moodenbaugh, Y. Nakamura, and S. Uchida, *Coexistence of, and Competition between, Superconductivity and Charge-Stripe Order in $\text{La}_{1.6-x}\text{Nd}_{0.4}\text{Sr}_x\text{CuO}_4$* , Phys. Rev. Lett., **78**, 338 (1996).
- [50] P. Corboz, S. R. White, G. Vidal, and M. Troyer, *Stripes in the two-dimensional t - J model with infinite projected entangled-pair states*, Phys. Rev. B, **84**, 041108 (2011).
- [51] J.-W. Li, B. Bruognolo, A. Weichselbaum, and J. von Delft, *Study of spin symmetry in the doped t - J model using infinite projected entangled pair states*, Phys. Rev. B, **103**, 075127 (2021).
- [52] N. J. Robinson, P. D. Johnson, T. M. Rice, and A. M. Tsvelik, *Anomalies in the pseudogap phase of the cuprates: competing ground states and the role of umklapp scattering*, Rep. Prog. Phys., **82**, 126501 (2019).