Infinite size density matrix renormalization group, revisited

I.P. McCulloch

Institut für Theoretische Physik C, RWTH Aachen University, D-52056 Aachen, Germany and School of Physical Sciences, The University of Queensland, Brisbane, Queensland, 4072, Australia

I revisit the infinite-size variant of the Density Matrix Renormalization Group (iDMRG) algorithm for obtaining a fixed-point translationally invariant matrix product wavefunction in the context of one-dimensional quantum systems. A crucial ingredient of this algorithm is an efficient transformation for obtaining the matrix elements of the wavefunction as the lattice size is increased, and I introduce here a versatile transformation that is demonstrated to be much more effective than previous versions. The resulting algorithm has a surprisingly close relationship to Vidal's Time Evolving Block Decimation for infinite systems, but allows much faster convergence. Access to a translationally invariant matrix product state allows the calculation of correlation functions based on the transfer matrix, which directly gives the spectrum of all correlation lengths. I also show some advantages of the Matrix Product Operator (MPO) technique for constructing expectation values of higher moments, such as the exact variance $\langle (H-E)^2 \rangle$.

PACS numbers:

I. INTRODUCTION

For many years, the density-matrix renormalizationgroup algorithm^{1,2} (DMRG) has been used in an increasing diversity of applications in fields such as condensed matter physics, condensed matter physics, quantum chemistry, nuclear physics and quantum information science. The DMRG algorithm is used to obtain a variational approximation for groundstates and low-lying excited states, with later variants being developed for finitetemperature and dynamical properties. The basic algorithm comes in two varieties. The most commonly used algorithm is the so-called *finite-size* algorithm, whereby a matrix-product state (MPS) wavefunction on a finitesize lattice (defined through a distinct set of matrices $A_n^{s_n}$, for each lattice site $n=1,2,\ldots,L$, where $|s_n\rangle$ is a d-dimensional basis for the local Hilbert space at site n) is iteratively improved until convergence. The infinite-size algorithm, on the other hand, grows the lattice by one or more sites each iteration and produces, at the fixed point, a wavefunction that is translationally invariant (possibly with a non-trivial unit cell). This approach is useful for probing quantities in the thermodynamic limit, without the influence of boundary effects, and such a translationally invariant state can readily be used as an initial state to study real-time evolution of a homogeneous quench as well as local (translation symmetry breaking) perturbations. Recently, interest in MPS algorithms that obtain a thermodynamic fixed point has been rekindled with the invention of a variant of Vidal's Time-Evolving Block Decimation (TEBD) algorithm¹¹ that applies directly to an infinite size wavefunction to give iTEBD¹², the resulting wavefunction being translationally invariant by construction (although with the restriction that the unit cell must be at least two sites in size). The iTEBD utilizes imaginary time evolution via a Trotter-Suzuki decomposition⁶, amounting to using the power method³³ to obtain the groundstate eigenvector of an approximation of the exponential of the Hamiltonian.

However the power method is not an efficient eigensolver, and the use of the Trotter-Suzuki decomposition requires careful scaling of the time-step to zero in order to obtain a properly converged state. Hence, all else being equal, a DMRG approach where an efficient local eigensolver is used to find a variationally optimal state ought to be much more robust and efficient.

In the past, the main use for the infinite size DMRG algorithm however has not been to achieve a fixed point, but rather to grow an initial wavefunction of a particular size as an initialization step for the finite-size algorithm¹. As an algorithm for obtaining a translationally invariant fixed point, infinite-size DMRG (hereafter referred to as iDMRG) has been virtually abandoned. There are several reasons for this: Firstly, finite-size DMRG (fDMRG) readily allows the application finite-size scaling 30 with respect to the lattice size, whereas for iDMRG the only available parameter is the number of states kept m. In fact, this is not a problem because the scaling relations in terms of the number of states have already been investigated by several authors^{27,29,31}, although having received insufficient attention this approach is not yet widely utilized. In fact, having only a single scaling parameter is a marked advantage of iDMRG over finite size approaches where either the number of states kept must be sufficiently large that the truncation effects are negligible, which may be impractical, or m must be considered as an additional parameter and scaling performed with respect to both parameters, which in the past has not always been done correctly (see Ref. 27 for the correct approach).

A second disadvantage of iDMRG over fDMRG is a technical limitation of the algorithm, namely the lack of a good initial guess vector for the local eigensolver at each DMRG iteration. In contrast, the development of the wavefunction acceleration procedure in fDMRG¹⁴, to transform the basis for the superblock wavefunction from one iteration to be the initial guess vector for the next iteration, was a breakthrough in the development of

finite-size DMRG that improves the efficiency of the algorithm by orders of magnitude. This transformation is essentially implicit in the matrix product formulation of DMRG^{4,8,13}. From this point of view, DMRG is an algorithm for locally updating the matrix elements of an MPS one (or a few) lattice sites at a time, and the wavefunction transformation results from the orthonormalization the basis which is necessary to ensure numerical stability. However, this transformation only applies to finite-size MPS, and the apparent lack of a similar transformation for iDMRG has been the focus of several studies in the last decade or more.

The problem that needs to be solved to obtain a wavefunction transformation for iDMRG is to find a procedure for predicting the sign of the new basis elements when the lattice size is increased. This sign structure has two components, being the intrinsic nodal-structure of the wavefunction, as well as a phase ambiguity of the eigenvectors of the density matrix (or singular value decomposition). There have been several works that attempt to obtain a transformation algorithm, either using Marshall's sign rule¹⁷ to predict the sign structure of the wavefunction 18, or directly comparing the sign structure from different steps^{19,20}. These approaches have had mixed success, in that they rely on a 1-1 matching of density matrix eigenvalues (equivalently, singular values arising from the Schmidt coefficients) from successive iterations and attempting to match signs of the eigenvectors (say, by choosing that the first component of the vector will always be positive). Where applicable, the Marshall's sign rule approach is very effective 18, but the need to know the nodal structure of the wavefunction a priori is a severe limitation, eg. this approach breaks down completely for frustrated systems. The approach of Qin and Lou does lead to a wavefunction that is somewhat better than a random state, and in this sense this approach is useful in reducing the number of iterations required by the local eigensolver. However, repairing eigenvector signs in this manner is not robust and this procedure has not been demonstrated to be effective in obtaining a fixed point wavefunction; that is, in the studies utilizing this approach 19,20 the algorithm does not converge to a point where the transformation reproduces the fixed point wavefunction with full fidelity.

It may be a surprise, therefore, that a solution to most of these problems exists in the literature and has done for quite some time, pre-dating even the wavefunction transformation in fDMRG, under the name of Product Wave-Function Renormalization Group (PWFRG)^{23,24,25,26}. In this approach, a recurrence relation is set up to obtain the translation operator that shifts the wavefunction by one site (or unit cell, which may be an arbitrary number of lattice sites). Using this operator the initial wavefunction for the next iteration can be obtained. However, while this procedure is effective in obtaining a translationally invariant fixed point, it still relies on matching singular values from successive iterations and the rate of convergence away from the fixed point is quite poor. This

is because the wavefunction transformation does not have full fidelity until the translation operator itself has converged, however the translation operator obtained from the recurrence relation converges is quite slowly. A main result of this paper is the introduction of a new transformation that reaches the same fixed point as PWFRG, but converges much faster when away from the fixed point, ie. when the lattice size is small. In fact, the convergence is effective enough that this approach is useful even for exact diagonalization of small systems. Also in this paper, I investigate some approaches to calculating correlation functions using the transfer operator. This approach has been known for quite some time^{8,9,29}, but has not been widely utilized. Thirdly, the more recent development of the Matrix Product Operator (MPO) formulation¹³, for obtaining the exact matrix product representation of the Hamiltonian and observables is demonstrated to have significant advantages and I present algorithms for obtaining expectation values in the infinite-size limit directly from the MPO representation.

The layout of this paper is as follows: In section II, I describe the basic matrix product approach and notation, for both wavefunctions and operators. In section III I describe the iDMRG algorithm itself and some discussion of the convergence properties. Section IV treats the calculation of observables for many cases including local quantities, correlation functions including the spectrum of correlation lengths, groundstate fidelities, and translationally-invariant operators such as the energy and the variance. Finally, section V contains a summary and some concluding remarks.

II. MATRIX PRODUCT STATES

We denote an MPS on an L-site lattice by the form,

$$\operatorname{Tr} \sum_{\{s_i\}} A^{s_1} A^{s_2} \cdots A^{s_L} \quad |s_1\rangle \otimes |s_2\rangle \otimes \cdots \otimes |s_L\rangle . \quad (1)$$

The local index s_i represents an element of the d-dimensional local Hilbert space at site i, and the A^{s_i} are $m \times m$ matrices, m being the dimension of the matrix basis (this quantity is also often called D, and sometimes χ). As a short-hand notation, it is convenient to omit the basis states $|s_i\rangle$ and the summation from Eq. (1) and speak of an MPS as being defined purely as a sequence of matrices. This notation is quite consistent if one regards the local index s_i as being a ket vector. Thus, in shorthand notation, Eq. (1) is equivalent to simply $A^{s_1}A^{s_2}\cdots A^{s_L}$.

In practice, a MPS state with no particular constraints on the form of the A-matrices is numerically difficult to handle and it is usual to impose some orthonormalization constraints (see eg. Ref. 13 for details),

$$\sum_{s} A^{s\dagger} A^s = 1 , \qquad (2)$$

for the left-orthonormalization constraint, and conversely

$$\sum_{s} B^s B^{s\dagger} = 1 , \qquad (3)$$

for the right-orthonormality constraint. In this paper, I use always A^s to denote a set of matrices satisfying the left-orthonormalization constraint Eq. (2), and B^s to denote a set of matrices satisfying the right constraint Eq. (3). The infinite-size DMRG algorithm is based around adding sites to the center of the chain. To this end, it is useful to orthonormalize the sites of the MPS from the edges, working in towards the center, to give a wavefunction of the form,

$$A^{s_1} \dots A^{s_{n-1}} A^{s_n} \Lambda B^{s_{n+1}} B^{s_{n+2}} \dots B^{s_L}$$
 (4)

Here Λ is a matrix that is left over after all of the sites in the MPS have been orthonormalized, which I refer to as the *center matrix*, and this matrix is also equivalent to what is called the *superblock wavefunction* in DMRG terminology.² In Ref. 13, I used C for this matrix, here I use Λ to stand for the matrix of singular values (λ_i) as start towards harmonizing the various notations associated with DMRG and MPS approaches.

These orthonormalization conditions correspond to a basis in which the reduced density matrix for the left (right) half of the system is diagonal, and are imposed for example by performing a singular value decomposition treating A^s as a single $dm \times m$ matrix (or B^s as a $m \times dm$ matrix),

$$A = \begin{bmatrix} A^1 \\ A^2 \\ \vdots \\ A^d \end{bmatrix}$$

$$B = \begin{bmatrix} B^1 B^2 \dots B^d \end{bmatrix}$$
(5)

and then performing the singular value decomposition $A = U\Lambda V^{\dagger}$, where Λ is a $m \times m$ diagonal matrix such that Λ^2 is the reduced density matrix, and U,V are row-unitary matrices. Upon rewriting U back as a set of $m \times m$ matrices analogously to Eq. (5), one obtains U^s as satisfying the orthogonality constraint Eq. (2), with a remainder term ΛV^{\dagger} appearing on the right (which we can, for example, incorporate into the neighboring A-matrix). The procedure for obtaining the right-orthonormalization constraint is similar, now ending up with a remainder matrix appearing on the left. In this fashion, the boundary of the left/right orthonormalized states can be moved to an arbitrary position in the lattice, a procedure I refer to as a rotation of the center matrix to the left or right.

The infinite-size DMRG produces is a wavefunction that grows by adding sites in the center, which, for the example case of a 2-site unit cell, gives a wavefunction at the $n^{\rm tm}$ iteration that is 2n sites long,

$$A_1^{s_1'} \dots A_{n-1}^{s_{n-1}'} A_n^{s_n'} \Lambda_n B_n^{s_n} B_{n-1}^{s_{n-1}} \dots B_1^{s_1} . \tag{6}$$

Sites are added until the A^s and B^s matrices in the center of the lattice converge to a fixed point. At this fixed point, the MPS will be translationally invariant, possibly with a non-trivial unit cell, such that the A^s, B^s matrices are independent of position. In TEBD the associated MPS is always of this translationally invariant form, but in iDMRG we will see that the translational invariance arises from the wavefunction transformation converging to the identity operator at the fixed point.

The MPS form Eq. (1) is related to the canonical form used by Vidal¹¹,

$$\dots \Gamma^{s_{n-1}} \Lambda \Gamma^{s_n} \Lambda \Gamma^{s_{n+1}} \Lambda \Gamma^{s_{n+2}} \dots , \tag{7}$$

by $A^s = \Lambda \Gamma^s$ and $B^s = \Gamma^s \Lambda^s$. For the canonical form, the orthogonality constraints take the form

$$\sum_{s} \Gamma^{s\dagger} \rho_{R} \Gamma^{s} = 1$$

$$\sum_{s} \Gamma^{s} \rho_{L} \Gamma^{s\dagger} = 1,$$
(8)

where $\rho_L = \Lambda \Lambda^\dagger$ and $\rho_R = \Lambda^\dagger \Lambda$ are the reduced density matrices for the left and right semi-infinite strips respectively. This shows that the Γ^s and Λ matrices for a canonical MPS are not independent. In terms of the A^s, B^s matrices, the corresponding fixed point relations are, ¹⁰

$$\sum_{s} A^{s} \rho_{L} A^{s\dagger} = \rho_{L}$$

$$\sum_{s} B^{s\dagger} \rho_{R} B^{s} = \rho_{R} . \tag{9}$$

States that have a non-zero quantum number can be readily obtained in this scheme, as long as a repeated unit cell can be properly defined. For example, for finite doping or finite magnetization with a density p/qfor coprime integers p, q, we use a unit cell of q sites and insert a shift in the quantum numbers of -p per unit cell. The effect of this shift in the wavefunction (equivalent to an auxiliary particle of quantum number -p) is to force the physical system to have an average quantum number of +p throughout the unit cell, while the entire MPS (physical system plus the quantum number shift) has an average quantum number of zero and thereby allowing a repeated unit cell. This shift can be implemented in a variety of equivalent ways, for example by introducing explicitly an additional matrix Q into the MPS unit cell that has matrix elements $\langle j' | Q | j \rangle = \delta_{j',j-p}$, or by shifting the quantum numbers in the local basis of one site in the unit cell by -p. Note that this procedure does not enforce that there will be exactly p particles per unit cell, which would be an undesirable restriction, but arbitrarily large fluctuations are allowed within the constraint that the average quantum number on the *infinite* system is p/q.

A. Matrix product operators

Ref. 13 introduces a representation for the Hamiltonian and other associated operators as an exact matrix

product operator (MPO). These operators come in several varieties, the main form that we consider here is operators that are characterized by an MPO representation that can be put in a lower (upper) triangular form, and therefore represent operators that are polynomial combinations of local operators, hence the expectation value of such an operator is a polynomial function of the lattice size, and in typical cases will be linear, for example giving the energy per site. A second form of MPO arises from products of local operators, such as occurs in the exponential of a triangular operator. This form has an expectation value that is an exponential function λ^L of the lattice size, hence I call this class exponential MPO's. Crosswhite and Bacon²¹ have also considered MPO's from the point of view of a correspondence with weighted finite-state automata.

Firstly, we briefly examine some properties of triangular MPO's. The notation we use here is that an MPO is a matrix of local operators of dimension $M \times M$. This form, after becoming familiar with the notation, is a very convenient way of representing an MPO, see Ref. 13 for full details. For simplicity we take here the unit cell to be a single site, so that the infinite MPO is represented by the same matrix repeated on every site. For cases where the unit cell of the Hamiltonian is non-trivial, for example an N-leg ladder, the associated MPO has the same periodicity which leads to the technicality of treating MPO matrices that may be non-square (trapezoidal, in the case of triangular MPO's), but this presents no serious difficulty and all of the procedures described here carry over to this case straightforwardly. Starting from the most general lower-triangular 2×2 MPO.

$$W = \begin{pmatrix} C & 0 \\ B & A \end{pmatrix} , \tag{10}$$

this represents the infinite sum of all terms of the form

$$A \otimes \cdots \otimes A \otimes B \otimes C \otimes \cdots \otimes C . \tag{11}$$

Note that Ref. 13 used M to denote the MPO matrix, whereas here I use W to avoid confusion with M, the dimension of the W matrix. By appropriate choices of local operators A, B, C, this form can be used for many variants of one-particle operators, including bosonic and fermionic particle operators at arbitrary lattice momenta, or the sum of local interactions. Moving to 3×3 matrices, we can generalize this form to a bond operator, or by including a term on the diagonal,

$$W = \begin{pmatrix} E & 0 & 0 \\ D & C & 0 \\ 0 & B & A \end{pmatrix} , \tag{12}$$

which represents the sum of all string-like terms of the form

$$A \otimes \cdots \otimes A \otimes B \otimes C \otimes \cdots \otimes C \otimes D \otimes E \otimes \cdots \otimes E$$
. (13)

Thus long-range string operators can be represented just as easily as short-range terms. Taking the string operator C to be proportional to the identity operator shows

that non-local but exponentially decaying interactions can be used in DMRG with negligible loss of efficiency over purely local interactions, a fact that is probably well known to DMRG practitioners although I am not aware of any studies that make use of this.

For a concrete example of a triangular MPO, combining a bond term and a local term a simple but non-trivial Hamiltonian operator is the Ising model in a transverse field (ITF),

$$H = \sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z + \lambda \sum_i \sigma_i^x , \qquad (14)$$

which has the lower-triangular MPO form as 3×3 matrices.

$$W = \begin{pmatrix} 1 & 0 & 0 \\ \sigma^z & 0 & 0 \\ \lambda \sigma^x & \sigma^z & 1 \end{pmatrix} . \tag{15}$$

The use of MPO's gives a convenient way to construct a software code that has no explicit dependence on the particular Hamiltonian. The main advantage, however, is that the MPO form allows arithmetic operations, in particular sums and products of operators are represented simply by taking the matrix direct sum and direct product respectively of the MPO representation, as described in Ref. 13. If both operators have a lower-triangular MPO representation, then the sum or product is also a lower-triangular MPO. This means that it is very easy to obtain operators that would take considerable effort to construct using the ad-hoc methods traditionally used in DMRG and MPS approaches. One such example is the variance, $(H-E)^2$, which replaces the discarded weight (the truncation error, in DMRG terminology) as the convergence measure of choice as it gives a measure that is both independent of the details of the algorithm and a more reliable indicator of convergence.

Not all useful operators have the form of a triangular MPO. The triangular MPO's represent sums of either short-range or string terms, whereas other kinds of operators of interest comprise instead products of terms. An example of this is the MPO representation of the Trotter-Suzuki approximation of an exponential, for example the first order decomposition of the exponential of a Hamiltonian operator with only nearestneighbor terms into terms acting on even and odd bonds, $\exp[-\beta H] \simeq \exp[-\beta H_{\text{even}}] \exp[-\beta H_{\text{odd}}]$. The MPO representation of $\exp[-\beta H_{\text{even}}]$ has a 2-site unit cell with alternating $1 \times M$ and $M \times 1$ -dimensional W-matrices representing the product of all of the exponentials for the even bonds, and similarly $\exp[-\beta H_{\rm odd}]$ alternates $M \times 1$ and $1 \times M$ matrices. The product of the two operators therefore is an $M \times M$ MPO that will not be lower-triangular. This operator falls into the class of exponential MPO's, with an expectation value of the form λ^L for L sites, where λ is the eigenvalue corresponding to the fixed point E^{α} eigenmatrices (see below).

The MPO formulation gives a natural form for obtaining the actual matrix elements of an operator in the form

required to calculate an observable using an MPS wavefunction. This is done by obtaining matrices E^{α} and F^{β} that represent the operator matrix elements on the left and right halves of the system respectively. Later we will see how to obtain these matrices for infinite states directly, however let us first consider how to construct these matrices for a finite, but arbitrary size system, which precisely what is needed for the iDMRG algorithm. Let us take the wavefunction of size 2n sites, given by Eq. (6). The initial E^{α} , F^{β} matrices for the boundaries are taken to be

$$E_0^{\alpha} = \delta_{\alpha,M} I_{\text{left}} ,$$

$$F_0^{\beta} = \delta_{\beta,1} I_{\text{right}} .$$
(16)

where $I_{\rm left}$ and $I_{\rm right}$ are the identity operators acting on the left- and right-hand edges of the wavefunction. For an open boundary condition state, which is always what we use in iDMRG, these operators will simply be 1×1 identity matrices. The E^{α} , F^{β} for the remainder of the chain are then defined recursively,

$$E_{n}^{\alpha'} = \sum_{s',s,\alpha} W_{\alpha'\alpha}^{s's} A^{s'_{n}\dagger} E_{n-1}^{\alpha} A^{s_{n}} F_{n}^{\alpha'} = \sum_{s',s,\alpha} W_{\alpha'\alpha}^{s's} B^{s'_{n}} F_{n-1}^{\alpha} B^{s_{n}\dagger}$$
(17)

Once we have obtained the matrices E_n^{α} , F_n^{β} for the center of the chain, we can perform many operations. For example, the action of the operator on the center matrix in its projected Hilbert space is

$$W(\Lambda_n) = \sum_{\alpha} E_n^{\alpha} \Lambda_n F_n^{\beta \dagger} , \qquad (18)$$

and the expectation value is given by the matrix inner product,

$$\langle \Lambda_n | W | \Lambda_n \rangle = \operatorname{Tr} E_n^{\alpha} \Lambda_n F_n^{\beta \dagger} \Lambda^{\dagger} .$$
 (19)

Note that the action on the center matrix Eq. (18) occurs in the projected Hilbert space of the $m \times m$ dimensional basis that is the support of Λ_n , and Eq. (18) in no way represents the action of the operator on the entire state (but that action is easy to obtain, see for example Ref. 13). It would be a serious error to expect, for example, that the action of the product of two operators W_1W_2 on a wavefunction can be obtained by applying separately the E, F matrices for W_2 and then the E, Fmatrices for W_1 . That action would correspond instead to the operator W_1PW_2 , where P is the projector onto the $m \times m$ -dimensional Hilbert space. Nevertheless, for a properly constructed set of E, F matrices, the expectation value Eq. (19) is exact, because it results from the inner product of the action of the operator with the original state, for which the fact that this operation is being performed within a projected Hilbert space does not matter.

Having seen how to obtain matrix elements in the projected Hilbert space of a Λ matrix in the MPS, the final ingredient before presenting the iDMRG algorithm is to

see how the MPS itself is iteratively updated. For this task, we need the action of an operator on not just a Λ matrix, but on one (or more) of the A^s , B^s themselves. This is most easily demonstrated for the action of an MPO on a single site and incorporating the center matrix. Thus, taking the wavefunction of Eq. (6) and leaving all matrices fixed except $A_n^{s'_n}$ and Λ_n , we obtain the action of an MPO on these matrices,

$$W(A_n^{s_n'}\Lambda_n) = \sum_{\alpha,\beta,\sigma} E_{n-1}^{\alpha} A_n^{\sigma} \Lambda_n F_n^{\beta\dagger} W_{\alpha\beta}^{\sigma s_n'}, \qquad (20)$$

where we have used the explicit matrix elements of the W matrix defining the MPO $W_{\alpha\beta}^{\sigma s'_n}$, with σ, s'_n indices running over the d-dimensional local basis and α, β indices running over the M-dimensional matrix basis of the MPO. For modifying two or more sites at a time, analogous formulas apply with one W matrix appearing per site.

III. IDMRG

In this section, I describe the iDMRG algorithm and discuss some convergence and stability issues. The algorithm itself is presented in figure 1. For simplicity I have used a unit cell of two sites, however the unit cell can be any size, including a single site. If the system is parity-symmetric then step 3 can be avoided with the B^s matrices taken to be the transpose of A^s (if good quantum numbers are used, the quantum numbers in the matrix basis must be flipped as well, otherwise one ends up with a conjugate tensor rather than a normal tensor¹³), being careful to use an eigenvalue decomposition rather than a singular value decomposition to ensure symmetry of the left and right basis vectors. For unit cells larger than two sites, the only modification is steps 2 and 3 rotate the center matrix through the remainder of the unit cell, optimizing the matrices comprising the unit cell in the process. Thus, this transformation gives a recurrence relation whereby a wavefunction on an n-site lattice and a wavefunction on a larger n + k-site lattice for some arbitrary k > 0, are used to obtain an approximate wavefunction for an n + 2k-site lattice. Note that steps 2 and 3 act independently on $|\Psi_n\rangle$ and could be performed in

The key component of this algorithm is step 4, where the wavefunction center matrix $\Lambda_{n+1}^{\text{trial}}$ is obtained from those of the previous two iterations,

$$\Lambda_{n+1}^{\text{trial}} = \Lambda_n^R \Lambda_{n-1}^{-1} \Lambda_n^L . \tag{21}$$

Note that Λ_n^L and Λ_n^R are center matrices of the same wavefunction $|\Psi_n\rangle$, but located at different partitions, at the left and right boundaries of the unit cell respectively. Eq. (21) is a straightforward computation that, unlike the PWFRG transformation²³ and sign-rule based approaches, can be applied directly without any kind of

- 1. (initialization) Obtain the wavefunction for a two-site lattice $|\Psi_0\rangle = A_0^{s_0'} \Lambda_0 B_0^{s_0}$ and a four-site lattice $|\Psi_1\rangle = A_0^{s_0'} A_1^{s_1'} \Lambda_1 B_1^{s_1} B_0^{s_0}$, and set n=1 for the start of iterations.
- 2. Rotate the center matrix of $|\Psi_n\rangle$ one step to the left, to obtain $\dots \Lambda_n^L B_{n+1}^{s'_{n+1}} B_n^{s_n} \dots$ 3. Rotate the center matrix of $|\Psi_n\rangle$ one step to the
- 3. Rotate the center matrix of $|\Psi_n\rangle$ one step to the right, to obtain ... $A_n^{s_n'} A_{n+1}^{s_{n+1}} \Lambda_n^R \dots$
- 4. (new wavefunction) The trial wavefunction for a lattice with size increased by two sites is given by $|\Psi_{n+1}^{\text{trial}}\rangle = \dots A_{n-1}^{s'_{n+1}} \Lambda_n^R \Lambda_n^{-1} \Lambda_n^L B_{n-1}^{s_{n+1}} \dots$
- $|\Psi_{n+1}^{\text{trial}}\rangle = \dots A_{n+1}^{s'_{n+1}} \Lambda_{n}^{R} \Lambda_{n-1}^{-1} \Lambda_{n}^{L} B_{n+1}^{s_{n+1}} \dots$ 5. (eigensolver) Use $|\Psi_{n+1}^{\text{trial}}\rangle$ as the initial guess vector for an eigensolver, updating the matrices $A_{n+1}^{s'_{n+1}} \Lambda_{n+1} B_{n+1}^{s_{n+1}}$ to obtain the final $|\Psi_{n+1}\rangle$.
 6. (truncation) Truncate the dimension of Λ_{n+1} as
- 6. (truncation) Truncate the dimension of Λ_{n+1} as desired, using eg. a singular value decomposition or eigenvalue decomposition, and normalize.
- 7. (stopping criteria) If Λ_{n+1} is sufficiently close to $\Lambda_n^{L,R}$, then the fixed point has been reached. Otherwise increase $n \leftarrow n+1$ and return to step 2.

FIG. 1: The iDMRG algorithm, including the wavefunction transformation.

matching of singular values from different basis sets. This is because the basis for the right hand side of Λ_n^R is the same as the right hand side of Λ_{n-1} , and similarly the left hand side of Λ_n^L matches the left hand side of Λ_{n-1} . Thus there is not even any requirement here to diagonalize the Λ matrices, they need not even be square. For simplicity of calculating the inverse Λ_{n-1}^{-1} however, one can choose a basis in which Λ_{n-1} is diagonal, which is normally obtained as part of the truncation step anyway.

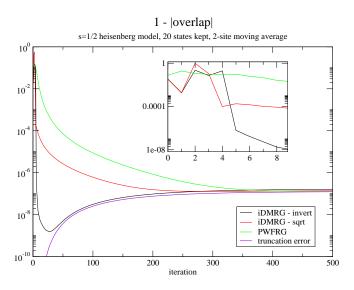


FIG. 2: Fidelity $1-\langle \Psi_n^{\rm trial}|\Psi_n\rangle$ of the wavefunction transformation for iDMRG and PWFRG. For comparison, the truncation error is also included, which sets a bound for the maximum possible fidelity. To remove small period two oscillations caused by the two-site unit cell, a two-site moving average was used for all quantities.

Figure 2 shows the effectiveness of the iDMRG wavefunction transformation compared with PWFRG for a simple example. This figure shows the fidelity of the initial wavefunction with respect to the variational minimum, $1 - \langle \Psi_n^{\text{trial}} | \Psi_n \rangle$ for the spin-1/2 Heisenberg model with a 2-site unit cell and m=20 states kept in an SU(2)-symmetric basis 13,16 . The correlation length for this basis size is ~ 48.5 lattice sites. The exact groundstate of this Hamiltonian is critical, however for a finite basis size the correlation length of a matrix product wavefunction will always be finite.^{7,8} The signature of critical behavior is a scaling law for the correlation length as the basis size is increased, of the form $\xi = m^{\kappa}$, where the exponent $\kappa \simeq 1.3$ for the Heisenberg model. 29,31 Also included in the figure is a variant of Eq. (21) that avoids the matrix inverse by taking instead the square root of the singular values of $\Lambda_n^R \Lambda_n^L$ in the basis where Λ_{n-1} is diagonal, but this transformation is much less effective for small systems. As expected, the PWFRG transformation converges only for system sizes much greater than the correlation length due to the slow convergence of the translation operator.

An interesting feature of Fig. 2 is the excellent performance of the iDMRG transformation for systems much smaller than the correlation length where the transformation captures the effects of the open boundary conditions extremely well, even for systems small enough to be exactly diagonalizable. But for an exact diagonalization, periodic boundary conditions are more natural. So, in Fig. 3, I compare the effectiveness of the transformation also for periodic boundary conditions, increasing the number of states kept to the limit of the numerical accuracy of the wavefunction transformation. Using just 10 Lanczos iterations at each lattice size, we obtain an energy for a 30-site system with periodic boundary conditions of E = -13.321963058(9), accurate to 10 significant figures, showing that this transformation could be useful for accelerating exact diagonalization studies.

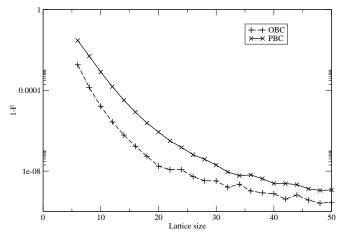


FIG. 3: Fidelity $1 - \langle \Psi_n^{\rm trial} | \Psi_n \rangle$ of the wavefunction transformation for open and periodic boundary conditions.

Clearly this algorithm has a similar function to

iTEBD in the sense that both algorithms result in a translationally-invariant variational approximation to the groundstate wavefunction. But in fact the relationship between iDMRG and iTEBD is much deeper. If we replace step 5 by an alternative,

5. (evolution) Obtain
$$|\Psi_{n+1}\rangle$$
 by applying a single bond evolution operator to sites s'_{n+1} and s_{n+1} of $|\Psi_{n+1}^{\text{trial}}\rangle$, (22)

then the iTEBD algorithm is recovered *exactly*, where the translationally invariant wavefunction at the n^{th} step is given, in notation corresponding with that of Vidal¹², as,

$$\dots \Lambda_B \Gamma_A^{s_1} \Lambda_A \Gamma_B^{s_2} \Lambda_B \Gamma_A^{s_3} \Lambda_A \Gamma_B^{s_4} \Lambda_B \dots , \qquad (23)$$

where $\Gamma_A^s = \Lambda_{n-1}^{-1} A_n^s$, $\Lambda_A = \Lambda_n$, $\Gamma_B^s = B_n^s \Lambda_{n-1}^{-1}$, and $\Lambda_B = \Lambda_{n-1}$. This is remarkable, because the iTEBD algorithm was originally conceived as evolving, simultaneously, all bonds on an infinite-size translationally invariant wavefunction, but it is formally equivalent to a lattice-growth scheme where two sites are added to the lattice at each iteration and the evolution is done performed via a single bond operator at the center of the lattice. This latter interpretation has advantages, in particular the basis states in the iDMRG point of view are always orthogonal, which provides a framework from which the convergence and stability properties are amenable to analysis.

A. The orthogonality fidelity

A measure of how close the wavefunction is to the fixed point is obtained by comparing the wavefunctions at successive iterations. At the fixed point, $\Lambda_n^R \Lambda_{n-1}^{-1}$ will be a diagonal matrix that represents the phases of the eigenvectors chosen in the truncation procedure (if there are degenerate eigenvalues then the corresponding block can be an arbitrary unitary matrix) that can be incorporated into the neighboring matrices $A_{n+1}^{s'_{n+1}}$ so that the wavefunction transformation is the identity operation. Similarly, the diagonal matrix of phases of $\Lambda_{n-1}^{-1}\Lambda_n^L$ can be incorporated into $B_{n+1}^{s_{n+1}}$. A measure for how similar the wavefunctions are can easily be obtained by calculating the fidelity between these wavefunctions at successive iterations, which I call the *orthogonality fidelity*,

$$F_{\text{ortho}} = \langle \Lambda_n | \Lambda_{n-1} \rangle ,$$
 (24)

which is best obtained via the reduced density matrices,

$$F_{\text{ortho}} = \text{Tr}\sqrt{\sqrt{\rho_n^R} \rho_{n-1} \sqrt{\rho_n^R}}, \qquad (25)$$

with the density matrices given by $\rho_n^R = \Lambda_n^{R\dagger} \Lambda_n^R$ and $\rho_{n-1} = \Lambda_{n-1}^{\dagger} \Lambda_{n-1}$. This reduces to the sum of the singular values of $\Lambda_n^R \Lambda_{n-1}^{\dagger}$, so is quite easy and efficient to calculate. From the TEBD point of view, this measure

shows how close the basis states are to being orthogonal. If there is any discarded weight in the truncation procedure of Fig. 1 step 5, this has a corresponding effect on $F_{\rm ortho}$. That is, in iDMRG the wavefunction only reaches an exactly orthogonal fixed point if it is arranged that the truncation error is zero, or in iTEBD the requirement for exact orthogonality is that the time-step goes to zero.

This points to an undesirable side-effect of a non-zero truncation of the wavefunction. While having a finite truncation is generally highly beneficial to convergence compared with, eg. single-site algorithms with no density-matrix mixing that are highly susceptible to convergence problems, near the fixed point of iDMRG a finite truncation error is a hindrance. To investigate the effects of truncation errors on the fixed point, I have compared iTEBD for various time-steps as well as two-site and single-site iDMRG algorithms with respect to how close the resulting state is to optimal, for a given number of states. Much of this section applies as well for finite-size calculations, although probably with less extreme differences.

Compared with standard two-site (i)DMRG, the (i)TEBD algorithm with a careful reduction in time-step towards zero, produces a better wavefunction for a given number of states. Surprisingly, this difference is quite substantial, in all cases I looked at amounting to around 30% improvement in the number of states needed for a given accuracy. In most applications of DMRG this is of no consequence, and it is not an efficient use of computer resources to try to converge to the best possible wavefunction for a given number of states. Instead of performing many many iterations to get 'optimal' convergence, often a better variational state is achieved with a less-well converged wavefunction by a small increase in the basis size. Nevertheless in some cases the number of states is all-important. In particular, to calculate an optimal state for a given m in a scaling analysis, an error bar of 30% is unacceptable. In addition, for applications such as time evolution, where the number of states kept grows substantially (often exponentially) during the calculation, a 30% reduction in the number of states required for the initial state may lead to a substantial improvement in the efficiency of the evolution. In a time-evolution calculation, obtaining the initial state is usually much faster than the time evolution itself, so additional time spent on improving the initial state is worthwhile.

Takasaki, Hikihara and Nishino²⁸ have proposed a combination of two-site and single-site algorithms, using the single-site algorithm in the last few sweeps of the calculation. However, this still doesn't produce the best possible result because the best states to keep do not correspond exactly with the largest m eigenstates of the density matrix. To be more precise, we can interpret the two-site algorithm as equivalent to a single-site algorithm with an environment that is perturbed by additional degrees of freedom corresponding to the second site. This perturbed environment means that the state isn't exactly translationally invariant due to the appear-

ance of a shallow bound-state at the central two sites, as noted by Dukelsky et al.³ But this perturbation to the wavefunction is such that the largest m density matrix states do not give the optimal states to keep at the point of crossing over to the single-site algorithm. With no good quantum numbers this is not such a serious problem and I would expect that the algorithm converges to a nearly-optimal state regardless (although possibly quite slowly). However practically all problems of interest utilize good quantum numbers, and in that case the choice of the number of states in each symmetry sector, which is fixed and does not change during the course of the single-site algorithm, will not be optimal. The solution is instead to evolve the wavefunction slowly, such that as the fixed point is approached the change in wavefunction at each iteration smoothly goes to zero. This does not happen in two-site DMRG because at the 'fixed point' the change in wavefunction per iteration is precisely the truncation error, which is usually small but non-negligible. A slow evolution of the wavefunction corresponds to imaginary time-evolution with a small time-step, which could be done either with TEBD via the Trotter-Suzuki decomposition, or directly with DMRG by replacing the eigensolver by a multiplication with $1 - \delta \tau H$. A similar effect could be obtained simply by performing just a single iteration of the eigensolver (Lanczos or similar) in the final sweeps of standard two-site DMRG. An alternative approach that should achieve the same effect is White's single-site density matrix mixing approach⁵, and smoothly sending the density matrix perturbation to zero in the final stages of the calculation.

B. Explicit orthogonalization

The previous section has argued that, regardless of which algorithm is used to obtain the wavefunction, a finite truncation of the wavefunction leads a loss of orthogonality, characterized by an orthogonality fidelity $1-F_{\rm ortho}$ which will be of the same magnitude as the discarded weight of the truncation. This is not ideal, since even a small non-orthogonality affects the computation of expectation values, and the iterative algorithms for expectation values described in Sec. IV below are simpler and perform better when the wavefunction is exactly orthogonalized. Fortunately, this presents no problem as a procedure for explicitly orthogonalizing a translationally invariant MPS has been developed by Orús and Vidal 32 .

For concreteness let us take a unit cell of 2 sites, so the wavefunction at some iteration n is given by the matrices $A_n^{s'_n} \Lambda_n B_n^{s_n}$ together with the center matrix at the previous iteration, Λ_{n-1} . Rotating the center matrix to the right gives the unit cell in the form of left-orthogonalized matrices,

$$A_n^{s_n'} A_{n+1}^{s_{n+1}'} \Lambda_n^R \Lambda_{n-1}^{-1} ,$$
 (26)

and the infinite wavefunction comprises this set of matrices repeated. ie, with $P=\Lambda_n^R\Lambda_{n-1}^{-1}$ the wavefunction

is

$$A_n^{s_1} A_{n+1}^{s_2} P A_n^{s_3} A_{n+1}^{s_4} P A_n^{s_5} A_{n+1}^{s_6} P \dots$$
(27)

If the wavefunction is exactly orthogonal, then P will be the identity matrix (up to a unitary matrix that could be incorporated into one of the A-matrices), and the orthogonality fidelity will be $1 - F_{\text{ortho}} = 0$. But if $P \neq 1$, the MPS is not orthogonal because no matter whether we attach P to the A-matrix on the left or the right, the resulting A-matrix will not satisfy the orthogonality constraint Eq. (2). A simple-minded way of making the MPS orthogonal would be to simply rotate the Pmatrices to the right iteratively, until a fixed point of the updated A-matrices is reached. 45 This is clearly not efficient, however. Instead, the procedure of Orús and Vidal³² is to find a basis such that the dominant eigenmatrix of the transfer operator is the identity matrix⁸. The transfer operator in the left-orthogonal basis T_L is simply the identity operator of the unit cell, given by,

$$T_L(E) = \sum_{s_1, s_2} P^{\dagger} A_{n+1}^{s_2 \dagger} A_n^{s_1 \dagger} E A_n^{s_1} A_{n+1}^{s_2} P . \qquad (28)$$

For an orthogonal state, Eq. (2) implies $T_L(1) = 1$ is a fixed point, as we would expect for a representation of the identity operator. For a non-orthogonal state we have instead $T_L(1) = P^{\dagger}P$. So, in order to transform to a basis in which P = 1, we obtain, with an efficient eigensolver such as the Arnoldi method³³, the dominant eigenvector V_L of T_L , which we decompose into

$$V_L = X^{\dagger} X \ . \tag{29}$$

Note that V_L is always Hermitian and non-negative. The factorization Eq. (29) could be done by a Cholesky facorization, however this is unstable if V_L is close to singular³³; factorizing via a singular value or eigenvalue decomposition is more robust. We now perform a similarity transformation on the unit cell,

$$A_n^{s_1}A_{n+1}^{s_2}P\to XA_n^{s_1}A_{n+1}^{s_2}PX^{-1}\;, \tag{30}$$

and in this new basis, $T'_L(1) = 1$ as required. Note that to obtain the new Λ matrices, an additional step is needed. This could be done by solving the fixed point relation Eq. (9) but here we follow the outline of Ref. 32 and solve the corresponding right-orthogonal transfer matrix for the right-orthogonalized unit cell,

$$T_R(E) = \sum_{s_1, s_2} Q B_n^{s_1} B_{n+1}^{s_2} E B_{n+1}^{s_2 \dagger} B_n^{s_1 \dagger} Q^{\dagger} , \qquad (31)$$

where $Q = \Lambda_{n-1}^{-1} \Lambda_n^L$. Similarly to the left eigenvector, we now obtain the eigenvector V_R of T_R , and decompose into

$$V_L = YY^{\dagger} \,, \tag{32}$$

from which we see that the complete set of transformations is,

$$\begin{array}{cccc} A_{n}^{s_{1}} & \rightarrow & XA_{n}^{s_{1}} \\ B_{n+1}^{s_{2}} & \rightarrow & YB_{n+1}^{s_{2}} \\ \Lambda_{n-1} & \rightarrow & X\Lambda_{n-1}Y \\ \Lambda_{n}^{R} & \rightarrow & \Lambda_{n}^{R}Y \\ \Lambda_{n}^{L} & \rightarrow & X\Lambda_{n}^{L} \end{array} \tag{33}$$

This choice of tensors has the advantage over that used in Ref. 32 that the inverses of the X, Y matrices are not actually required, only the inverse of $X\Lambda_{n-1}Y$ which determines the singular values in the orthogonal basis. There is one other difference here to the procedure described by Orus and Vidal, concerning the treatment of unit cells of N > 1 sites. Orus and Vidal suggested handling these unit cells with a coarse-graining procedure that combines a unit cell of N cites into a single site with an increased local dimension. This is inefficient, as it corresponds to evaluating the summations in the eigenvalue equation Eq. (28) in a non-optimal way, using $O(d^N)$ matrix multiplies rather than O(Nd) multiplies if the summations are performed sequentially. Aside from this difference, and the slightly different choice of tensors, the procedure described here is equivalent to that of Orús and Vidal. Note that this procedure orthogonalizes the unit cell, it does not however guarantee that the individual sites comprising a multi-site unit cell separately satisfy Eq. (2). The step of orthogonalizing each site should be done for the sake of numerical robustness, and is readily achieved through the usual orthogonalization procedure, described in Sec. II.

The orthogonalization procedure is quite efficient, and for robustness I always perform this orthogonalization step before calculating the final expectation values.

IV. EXPECTATION VALUES

In this section, we consider expectation values of MPO's, firstly for the case of an operator that acts only on a finite range of the lattice, then for translationally invariant triangular MPO's.

A. Finite-range operators

Any finite-range operator of width N can be expressed as a position-dependent MPO with matrices $W_1 \dots W_N$ with an open-boundary condition representation. That is, the left-hand MPO matrix is a $1 \times M$ row vector, and the right-hand MPO is a $M' \times 1$ column vector. Starting from the right-hand side, the initial operator represents the identity operator on an infinite lattice, which will be trivial if the state is orthonormalized. Thus define $F_0^1 = 1$ to be this identity operator, then starting from the right hand side,

$$F_{n+1}^{\alpha'} = \sum_{\alpha, s', s} W_{(n+1)\alpha'\alpha}^{s's} B^{s'} F_n^{\alpha} B^{s\dagger} . \qquad (34)$$

The final iteration results in a single matrix F_N^1 that represents the operator matrix elements. The expectation value is then given by $\operatorname{Tr} F_N^1 \rho$ as usual, where ρ is the density matrix for the semi-infinite strip, being the fixed point of $\rho = \sum_s B^{s\dagger} \rho B^s$.

B. Triangular operators

To evaluate an expectation value of a lower-triangular MPO, we need to determine the matrix elements as a function of the wavefunction A-matrix elements. For an MPO of dimension $M \times M$, we have M different matrices E^{α} , $\alpha = 1, 2, \ldots, M$, that represent the components of the MPO on the remainder of the lattice. For concreteness, we firstly assume that the operator is of the form of a sum of short-range terms. This means that the diagonal of the MPO is zero, except for the top-left and bottom-right entries which are equal to the identity operator. This gives the fixed-point recursion

$$E^{\alpha'} = \sum_{\alpha, s', s} W_{\alpha'\alpha}^{s's} A^{s'\dagger} E^{\alpha} A^s - E_0 \delta_{\alpha', 1} , \qquad (35)$$

where E_0 is the expectation value per site. This is straightforward to evaluate term by term. Starting from the last entry at $\alpha = M$, since the bottom-right entry is the local identity operator we have,

$$E^M = \sum_s A^{s\dagger} E^M A^s , \qquad (36)$$

which is just the fixed point of the orthogonalization constraint, so for left-normalized matrices $E^M=1$ identically. The remaining matrices E^α for $1<\alpha< M$ are simple to construct in descending order because these matrices are functions only of the previous matrices $E^{\alpha'}$ for $\alpha'>\alpha$. The exception is the final matrix E^1 , which again has an identity operator on the diagonal, giving the linear equations

$$E^{1} - \sum_{s} A^{s'\dagger} E^{1} A^{s} + E_{0} = \sum_{\alpha > 1, s', s} W_{1\alpha}^{s's} A^{s'\dagger} E^{\alpha} A^{s} , \quad (37)$$

where the expectation value E_0 is given by the trace of the right hand side. If only the expectation value is required, these linear equations do not need to be actually solved. The full solution is required only if the matrix E^1 needed, for example to restart an iDMRG calculation from a given translationally invariant wavefunction.

The procedure when there are additional matrix elements on the diagonal of the MPO follows similarly.

C. Transfer matrix and symmetry breaking

The spectrum of the transfer matrix Eq. (28) is very useful as this determines the possible correlation lengths in the system.⁹ The transfer matrix has always at least

one eigenvalue equal to 1, as follows from the normalization constraint Eq. (2). The next leading eigenvalue determines the longest correlation length in the system. Normally this eigenvalue will be strictly less than 1. Even in the presence of long-range order, a non-zero order parameter will lead to a spontaneously broken symmetry at the infinite-size fixed point, as long as the symmetries that are imposed on the MPS representation allow this. In an infinite size system, a broken symmetry implies multiple groundstates which are exactly degenerate. Such spontaneous symmetry breaking typically does not occur in finite-sized systems because there are usually matrix elements connecting the groundstates that lift the degeneracy, and such matrix elements are exponentially small in the lattice size $\sim \exp[-L]$. Note that translation symmetry is an exception to this, since a finite system inherently breaks translation symmetry from the beginning.

Spontaneous symmetry breaking of discrete symmetries is realized in an infinite MPS with matrix elements connecting degenerate groundstates going to exactly zero at the fixed point. Hence we expect in a numerical simulation that exactly one of the degenerate states will be chosen non-deterministically. (If any of the groundstates has a lower entanglement then this state would tend to be preferentially selected by the variational principle. as this state would have a lower energy for a particular basis size.) However, if good quantum numbers are used then the obtained groundstate will always have this symmetry even if it would otherwise be spontaneously broken. For example, In the low-field ordered phase of the ITF model Eq. (14), there are two degenerate thermodynamic groundstates, namely the two fully polarized states $|\downarrow\downarrow\ldots\rangle$ and $|\uparrow\uparrow\ldots\rangle$. In the absence of any mechanism enforcing Z_2 symmetry, iTEBD or iDMRG will spontaneously select one of these groundstates, selected eg. by numerical noise from finite-precision arithmetic that leads to one groundstate having slightly higher weight than the other, which then quickly dominates.⁴⁶ However if a Z_2 quantum number is used, then the obtained groundstate will be an exact eigenstate of Z_2 , namely $|\downarrow\downarrow...\rangle \pm |\uparrow\uparrow...\rangle$, which is a GHZ state⁴⁰ with long-range order leading to an additional eigenvalue of unity in the transfer matrix spectrum. Thus, the transfer matrix is very useful for diagnosing the presence of longrange order, without a priori knowledge of the specific order parameter. Similar considerations also apply for the transfer matrix extended to string order parameters,³⁸ where the spontaneous symmetry breaking is instead associated with edge excitations that do not affect the bulk wavefunction 39 .

Figure 4 shows the transfer matrix spectrum of the ITF model in the vicinity of the critical point $\lambda=1$, for some different basis sizes. As can be seen, the ITF model, which is remarkable for having a very large correlation length even with a very small basis size, has relatively few eigenvalues that contribute to long range correlations. For example, considering only the eigenvalues cor-

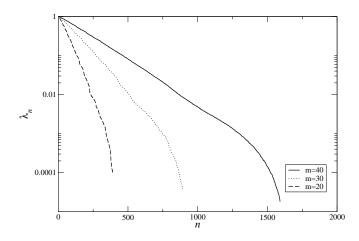


FIG. 4: The spectrum of the transfer matrix for the ITF model near the critical point. The next-leading eigenvalue determines the correlation length, here $\xi = 578.7, 1236.6, 2539.2$ for m = 20, 30, 40 respectively.

responding to correlation lengths of greater than 10 sites gives only around 30 relevant eigenvalues, even for the case of m = 40 where the correlation length is a few thousand sites. Thus the long-distance behavior of correlation functions can be determined by looking at just a few eigenvalues. In the case of a critical system, the powerlaw behavior of a correlation function is obtained approximately, as the sum of a set of exponentials⁹. Hence the eigenvalues of the transfer matrix give a good guide as to the range of validity where an approximate powerlaw can be expected, ie. to a distance \sim the maximum correlation length in the system. A complication of using the transfer matrix directly to expand a correlation function as a set of exponentials is that the transfer matrix of Eq. (28) is not a normal matrix and hence the eigenvectors are not orthogonal. But this presents no essential difficulty, as long as care is taken to project out the large number of components with short correlation lengths prior to determining the amplitudes (expansion coefficients) of each mode, eg. by calculating the operator E-matrix for a correlation at some distance larger than the smallest correlation length eigenvalue obtained from the transfer operator.

D. Fidelity

The overlap of two MPS is a useful operation that is used to obtain the groundstate fidelity $F(g_1, g_2) = |\langle g_1|g_2\rangle|$ between groundstates of different interaction parameters g_1 and g_2 . The fidelity has attracted much recent attention as a tool for probing quantum phase transitions.⁴² For an MPS, this quantity is always of the form d^L , for an L-site lattice with 0 < d < 1. Thus, the fidelity is measured per site in much the same way as the expectation value of an exponential MPO (indeed, the fidelity is equivalent to the expectation value of the

identity operator which can be seen as the simplest possible exponential MPO). The *fidelity susceptibility*⁴¹ is a measure of the curvature of the fidelity at a particular point, and hence shows how quickly the wavefunction is changing with the interaction parameter. The fidelity susceptibility per site is given by,

$$\chi_F(g) = \lim_{\delta g \to 0, L \to \infty} \frac{-2 \ln F(g, g + \delta g)}{L \, \delta g^2} \,. \tag{38}$$

For finite MPS, the fidelity has been proposed as a measure since at least the work of Andersson, Boman and Östlund.²⁹ For an infinite MPS, the fidelity (and hence also the fidelity susceptibility) is straightforward to calculate via an eigenvalue equation. Similar to the transfer operator (which could be seen as calculating the norm $\langle \Psi | \Psi \rangle$), the fidelity between two states $| \Psi 1 \rangle$ and $| \Psi 2 \rangle$ is given by the largest eigenvalue of an overlap operator, here given for a two-site unit cell example (compare Eq. (28), assuming the states are orthogonalized),

$$T_{12}(E) = \sum_{s_1, s_2} A_2^{s_2\dagger}(1) A_1^{s_1\dagger}(1) E A_1^{s_1}(2) A_2^{s_2}(2) , \quad (39)$$

where $A_n^{s_n}(1)$ and $A_n^{s_n}(2)$ are the matrices representing $|\Psi 1\rangle$ and $|\Psi 2\rangle$ respectively. The fidelity per unit cell is then given by the eigenvalue λ of $T_{12}(E) = \lambda E$, which can be any complex number of magnitude ≤ 1 .

In a finite MPS, the fidelity is sometimes complicated by boundary effects that are irrelevant to the bulk properties of the wavefunction. For a concrete example, the bilinear-biquadratic S = 1 chain⁴³ has a variety of groundstate phases including a gapped resonating valence bond (RVB) phase (including the AKLT point⁴⁴ that is exactly solved with a non-Abelian MPS with dimension m=1), a gapped dimer phase, and a critical trimerized phase. The RVB phase is characterized by effective spin-1/2 edge excitations that are best treated by applying real spin-1/2 states at the boundaries, which means that the bulk wavefunction in the infinite size limit has quantum numbers that are all half-integer. This presents no problems for MPS algorithms in the infinite limit, however in other phases such as the gapped dimer phase it is best not to supply spin-1/2 states at the boundaries, as these frustrate the formation of dimer pairs and also introduces spurious effects into the fidelity. On the other hand, to calculate the fidelity for a finite size system the Hilbert space of the wavefunctions must coincide exactly, which causes severe difficulties in calculating the fidelity across the RVB-dimer phase transition. These problems do not arise in the infinite MPS case, as the boundary effects are not present and having different spin representations occurring in the bond basis of the wavefunctions is no problem for the fidelity calculation.

V. CONCLUSIONS

In this paper I have examined the infinite-size DMRG algorithm from the perspective of the matrix prod-

uct approach, which gives directly a translationally invariant fixed point previously studied by Rommer and Östlund^{8,9}. The core development in this algorithm is the wavefunction transformation, which improves upon that used in PWFRG²³, and is useful even for small finite systems. Sec. IV has shown how to calculate expectation values in the thermodynamic limit of operators directly from the matrix product form of an observable, which gives access to higher moments such as the energy variance. The spectrum of the transfer matrix gives detailed information on the correlation length and ordered phases, and expanding a correlation function as a sum of exponentials corresponding to the modes of the transfer matrix gives an alternative approach to fitting a critical exponent. The groundstate fidelity between different groundstates is readily calculated, and this gives a precise way to determine the fidelity that is free of complications due to boundary effects.

Infinite-size DMRG represents an efficient method for finding translationally invariant states that could replace many of the current uses for finite-size DMRG. However one situation in which the current algorithms are not a sufficient replacement is the calculation of excited states, for example energy gaps, low-lying excited states and spectral functions. Despite some suggestions to the contrary²², I believe these calculations should be readily achievable. Indeed, the basic approach of representing a particle excitation as a superposition of local perturbations with an MPS has already been discussed by Rommer and Östlund⁹, leading to an ansatz for an excitation, for example the application of a particle creation c_k^{\dagger} at some momentum k to an infinite MPS. Building upon this approach I see no reason why there should not exist efficient infinite-size counterparts for the well-established correction vector³⁶ and dynamical DMRG³⁷ algorithms for frequency-space dynamical correlations of finite-size systems. Note that calculating spectral functions via real-time evolution of a local particle excitation¹⁵ is already achievable, by using a section of a translationally invariant state as the input for a finite-size time evolution.

The iDMRG algorithm presents an interesting dichotomy of solving simultaneously for both the wavefunction A-matrix elements and the Hamiltonian E-matrix elements. The present scheme uses an efficient Lanczos eigensolver to obtain the wavefunction, while the effect of growing the lattice one site at a time amounts to using the less efficient power method to obtain the Hamiltonian matrix elements, in the sense that exactly one iteration of the E-matrix multiply is performed per step. Therefore, this algorithm will converge in a minimum number of iterations that is at least as large as the correlation length. On the other hand, Sec. IVB shows how to obtain the Hamiltonian matrix elements of a translationally invariant state much more efficiently using a linear solver. This suggests an accelerated algorithm that efficiently solves simultaneously the fixed point wavefunction and the associated fixed point Hamiltonian Eq. (35), which could in

principle converge in a number of iterations much smaller than the correlation length at the cost of the algorithm being non-linear.

Algorithms for finding the groundstate of a onedimensional Hamiltonian form a component of some algorithms for higher dimensional systems, most notably the iPEPS algorithm³⁴ for 2D tensor product states³⁵. The higher efficiency of direct eigensolvers should serve a similar function as the 1D case, although the analogue of a triangular MPO for the Hamiltonian construction is clearly more complicated in 2D.²¹

Acknowledgments

Many thanks to Ulrich Schollwöck, Thomas Barthel, Guifre Vidal and Luca Tagliacozzo for stimulating conversations, and even more thanks to Ulrich and Luca for providing valuable comments on the manuscript.

- ¹ S. R. White, Phys. Rev. Lett. **69**, 2863 (1992); Phys. Rev. B **48**, 345 (1993).
- ² U. Schollwöck, Rev. Mod. Phys. **77**, 259 (2005).
- ³ J. Dukelsky, M. A. Martín-Delgado, T. Nishino and G. Sierra, Europhys. Lett. **43**, 457 (1998); I. P. McCulloch and M. Gulácsi, Europhys. Lett. **61**, 138 (2003).
- ⁴ F. Verstraete, D. Porras, and J. I. Cirac, Phys. Rev. Lett. 93, 227205 (2004).
- ⁵ S. R. White, Phys. Rev. B **72**, 180403(R) (2005).
- ⁶ H. F. Trotter, Am. Math. Soc. Transl. **10**, 545 (1959); M. Suzuki, Phys. Rev. B **31**, 2957 (1985).
- M. Fannes, B. Nachtergaele, and R. F. Werner, Commun. Math. Phys. 144, 443 (1992).
- ⁸ S. Östlund and S. Rommer, Phys. Rev. Lett. **75**, 3537 (1995).
- S. Rommer and S. Östlund, Phys. Rev. B 55, 2164 (1997).
 75, 3537 (1995).
- D. Peres-Gercia, F. Verstraete, M.M. Wold and J.I. Cirac, quant-ph/0608197.
- ¹¹ G. Vidal, Phys. Rev. Lett. **91**, 147902 (2003).
- ¹² G. Vidal, Phys. Rev. Lett. **98**, 070201 (2007).
- ¹³ I. P. McCulloch, J. Stat. Mech. P10014 (2007).
- ¹⁴ S. R. White, Phys. Rev. Lett. **77**, 3633 (1996).
- ¹⁵ S. R. White and A. E. Feiguin, Phys. Rev. Lett. **93**, 076401 (2004).
- ¹⁶ I. P. McCulloch and M. Gulácsi, Europhys. Lett. 57, 852 (2002).
- W. Marshall, Proc. R. Soc. London, Ser. A 232, 48 (1955).
- ¹⁸ U. Schollwöck, Phys. Rev. B **58**, 8194 (1998); **59**, 3917(E) (1999).
- ¹⁹ S.-J. Qin and J.-Z. Lou, Commun. Theor. Phys. **36**, 635 (2001).
- ²⁰ L. Sun, J. Zhang, S. Qin and Y. Lei, Phys. Rev. B 65, 132412 (2002).
- $^{21}\,$ G. M. Crosswhite and D. Bacon, preprint arXiv:0708.1221 $^{22}\,$ P. Pippan, S. R. White and H. G. Evertz, preprint
- ²² P. Pippan, S. R. White and H. G. Evertz, preprin arXiv:0801.1947
- ²³ T. Nishino and K. Okunishi, J. Phys. Soc. Jpn. **64**, 4084 (1995).
- Y. Hieida, K. Okunishi and Y. Akutsu, Phys. Lett. A 233, 464 (1997).
- ²⁵ K. Okunishi, Y. Hieida and Y. Akutsu, Phys. Rev. B **59**, 6806 (1999).
- ²⁶ K. Ueda, T. Nishino, K. Okunishi, Y. Hieida, R. Derian and A. Gendiar, J. Phys. Soc. Jpn. **75**, 014003 (2006).
- ²⁷ T. Nishino, K. Okunishi and M. Kikuchi, Phys. Lett. A 213, 69 (1996).
- ²⁸ H. Takasaki, T. Hikihara, and T. Nishino, J. Phys. Soc.

- Jpn. 68, 1537 (1999).
- ²⁹ M. Andersson, M. Boman and S. Östlund, Phys. Rev. B 59, 10494 (1999).
- ³⁰ For example, J. Cardy, Scaling and Renormalization in Statistical Physics, Cambridge University Press, Cambridge (1997).
- ³¹ L. Tagliacozzo, Thiago. R. de Oliveira, S. Iblisdir and J. I. Latorre, preprint arXiv:0712.1976
- ³² R. Orus and G. Vidal, preprint arXiv:0711.3960
- ³³ For example, G. H. Golub and C. F. van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore and London (1996).
- ³⁴ J. Jordan, R. Orús, G. Vidal, F. Verstraete and J. I. Cirac, preprint arXiv:cond-mat/0703788
- ³⁵ For example, N. Maeshima, J. Phys. Soc. Jpn. **73**, 60 (2004).
- ³⁶ T. D. Kühner and S. R. White, Phys. Rev. B **60**, 335 (1999).
- ³⁷ E. Jeckelmann, Phys. Rev. B **66**, 045114 (2002).
- ³⁸ D. Perez-Garcia, M.M. Wolf, M. Sanz, F. Verstraete and J.I. Cirac, preprint arXiv:0802.0447
- ³⁹ M. den Nijs and K. Rommelse, Phys. Rev. B, **40**, 4709 (1989); T. Kennedy and H. Tasaki, Phys. Rev. B **45**, 304 (1992).
- ⁴⁰ D. M. Greenberger, M. A. Horne and A. Zeilinger, Bell's theorem, Quantum Theory, and Conceptions of the Universe, Kluwer Academics, Dordrecht, The Netherlands (1989).
- ⁴¹ W. L. You, Y. W. Li, and S. J. Gu, Phys. Rev. E 76, 022101 (2007).
- For example, P. Zanardi and N. Paunković, Phys. Rev. E 74, 031123 (2006); H.-Q. Zhou and J. P. Barjaktarevic, preprint cond-mat/0701608.
- ⁴³ For example, A. Läuchli, G. Schmid and S. Trebst, Phys. Rev. B **74**, 144426 (2006).
- ⁴⁴ I. Affleck, T. Kennedy, E. Lieb and H. Tasaki, Phys. Rev. Lett. **59**, 799 (1987).
- ⁴⁵ This procedure is equivalent to running iterations of iDMRG/iTEBD with no update to the wavefunction (ie. omitting step 5 of Fig. 1).
- ⁴⁶ This is assuming that the number of states kept is not so big that multiple degenerate groundstates can be represented with high fidelity. In that situation the stability of the algorithm is in question, eg. it is possible that a small numerical noise could result in multiple copies of the same groundstate, thus leading to a catastrophically nonorthogonal basis.