From density-matrix renormalization group to matrix product states

To cite this article: Ian P McCulloch *J. Stat. Mech.* (2007) P10014

View the article online for updates and enhancements.

From density-matrix renormalization group to matrix product states

Ian P McCulloch

Institut für Theoretische Physik C, RWTH-Aachen, D-52056 Aachen, Germany
E-mail: ianmcc@physik.rwth-aachen.de

Abstract. In this paper we give an introduction to the numerical density matrix renormalization group (DMRG) algorithm, from the perspective of the more general matrix product state (MPS) formulation. We cover in detail the differences between the original DMRG formulation and the MPS approach, demonstrating the additional flexibility that arises from constructing both the wavefunction and the Hamiltonian in MPS form. We also show how to make use of global symmetries, for both the Abelian and non-Abelian cases.

Contents

1. Introduction

The DMRG algorithm was introduced by White [1], as an algorithm for calculating ground state properties of principally one-dimensional strongly correlated systems in condensed matter physics. The connection between DMRG and matrix product states [2]–[5] (also known as finitely correlated states) was first made by Rommer and Östlund [6], who identified the thermodynamic limit of DMRG with a position-independent matrix product wavefunction. Although DMRG had already proven itself to be useful empirically, this was an important step in rigorously establishing the physical basis of the algorithm due to the concrete and easy-to-manipulate form of matrix product states. Work on the spectra of density matrices [7], later formulated as scaling of the von Neumann entropy [8], has placed the algorithm on a firm footing, showing that the required computational effort (realized via the basis dimension m) is essentially a function of the entanglement of the wavefunction [9], which for one-dimensional ground states scales at worst logarithmically with the system size [10].

Computationally, MPS algorithms came to the fore with the assistance of a quantum information perspective, leading to algorithms for periodic boundary conditions [11] and finite temperature algorithms based on density operators [12]–[14]. At around the same time, methods for the simulation of real-time evolution were developed in DMRG [15, 16], which can also benefit from MPS formulations [17].

The common theme of MPS approaches is to allow algorithms that operate on multiple, distinct wavefunctions at the same time. This is possible in the original formulation of DMRG only by constructing a mixed effective Hilbert space that is weighted appropriately to represent all of the relevant states simultaneously. This is inefficient, as the algorithms typically scale as $O(m^3)$ (or up to $O(m^5)$ for periodic boundary

conditions [11]) in the number of basis states m, so increasing m so as to represent multiple states in the same basis is typically much slower than performing separate operations on each basis. In addition, the mixed basis approach lacks flexibility. While traditional DMRG programs calculate the wavefunction and a few (often predetermined) expectation values or correlation functions, if instead the wavefunction is calculated in the MPS representation of equation (1) it can be saved for later use as an *input* for many purposes. Perhaps the simplest such operation beyond the scope of traditional DMRG is to calculate the *fidelity*, or *overlap* between the wavefunctions obtained from separate calculations. In the MPS formulation, this calculation is straightforward. Nevertheless the determination of the scaling function for the fidelity of finite-size wavefunctions for different interaction strengths, provides a new tool for investigating phase transitions and crossover phenomena [18]–[20]. Indeed, due to the simplicity of the calculation the fidelity is likely in the coming years to be the first choice for quantitatively determining critical points. Similar measures of entanglement, such as the concurrence and single- and two-site entropy [21, 22], are also straightforward to calculate, hence the MPS formalism allows us to apply directly the emerging tools of quantum information to the study of realistic systems in condensed matter physics. An alternative measure, the Loschmidt Echo [23], is important because, unlike many of the quantum information theoretic measures, this is directly accessible in experiments while showing the rich behavior of the simpler measures. The Loschmidt Echo is more time-consuming to measure numerically as it requires a full time evolution simulation rather than a direct measurement: nevertheless it is well within the current state of the art [24].

In this paper, we focus on the case of open boundary condition matrix product states. This does not preclude calculation of periodic systems: however, the entanglement of such periodic states is increased such that in the large L limit (where L is the lattice size), the number of states kept tends to the square of that required for open boundary conditions [25]. Algorithms exist for periodic boundary conditions [11] and infinite systems [26] (not to be confused with the 'infinite-size' DMRG algorithm), and the basic formulas introduced here carry over to these cases, but we do not describe the specific algorithms here. In section 2, we introduce the basic formulation of matrix product states and formulas for the fidelity. Section 3 is devoted to a new approach, whereby we construct the Hamiltonian operator itself as an MPS, with many advantages. We cover some remaining details of the DMRG algorithm in section 4, before discussing in detail the use of Abelian and non-Abelian quantum numbers in section 5. We finish with a few concluding remarks in section 6, including some observations on finite temperature states.

2. Matrix product states

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

$$\text{Tr} \sum_{\{s_i\}} A^{s_1} A^{s_2} \cdots A^{s_L} \quad |s_1\rangle \otimes |s_2\rangle \otimes \cdots \otimes |s_L\rangle. \tag{1}$$

The local index s_i represents an element of a local Hilbert space at site i. The two important cases we cover here are when s_i runs over a d-dimensional local basis for a wavefunction $|s_i\rangle$, in which case we refer to this as a matrix product wavefunction (MPW), or s_i is a d^2-dimensional local basis for all operators acting on a local site, which we refer

to as a matrix product operator (MPO). In this paper, we use MPS for a generic state irrespective of the form of the local space, and use MPW or MPO as necessary when the distinction between wavefunctions and operators is important. Where possible, we suppress the local index s_i and regard the A matrix as a matrix of kets (or local operators, in the case of MPOs). This notation is useful because often operations on MPSs can be cast in the form of an ordinary matrix operation, save for the obvious difference that the matrix elements are not ordinary numbers. In general, the matrix product form can also represent periodic [11] and infinite (non-periodic) states [26], but here we use only the open-boundary form equivalent to the wavefunction obtained by DMRG [6]. To enforce this boundary condition, we require the leftmost matrix A^{s_1} to be $1 \times m$-dimensional, and the rightmost matrix A^{s_L} to be $m \times 1$. Here we have introduced m as the basis size, or dimension, of the *matrix basis* of the A matrices. This quantity is often denoted D, or sometimes χ, in the quantum information literature, but we emphasize it is exactly the same quantity in all cases. In general m is position-dependent, as we do not require the A matrices to be square even away from the boundary. Because of the one-dimensional basis at the boundaries we can regard the MPS wavefunction to be a sequence of operators attached to left and right (or outgoing and incoming) vacuum states. This makes the operator product in equation (1) an ordinary number, so the trace operation can be dropped.

2.1. Orthogonality constraints

In practice, an MPS state with no particular constraints on the form of the A matrices is numerically difficult to handle. We are always free to insert some product of a non-singular $m \times m$ operator X and its inverse X^{-1} in the middle of our MPS; thus we can apply an arbitrary transformation to the matrix basis of an A matrix, as long as we make the corresponding transformation to its neighbor. Using this freedom, we can transform the A matrices into a form where they are orthonormalized, that is, we prefer that they satisfy one of two possible constraints:

$$
\begin{aligned}
\sum_s A^s A^{s\dagger} = \mathbf{1} \qquad &\text{(right-handed)} \\
\sum_s A^{s\dagger} A^s = \mathbf{1} \qquad &\text{(left-handed)}.
\end{aligned}
\tag{2}
$$

States satisfying these conditions are orthonormalized in the sense that if all A matrices to the left of some matrix A^{s_n} are orthogonalized in the left-handed sense, then the basis on the left-hand side of A^{s_n} is orthonormal (i.e. the identity operator in the effective Hilbert space is trivial). Conversely, if all A matrices to the right of A^{s_n} are orthogonalized in the right-hand sense, then the basis on the right-hand side of A^{s_n} is orthogonal. Usually, we want both these conditions to be true simultaneously. Note that it is not, in general, possible for *all* of the A matrices (including A^{s_n} itself) to be in orthonormal form at the same time. There are several ways of transforming an arbitrary MPS into this normalized form. Two ways that we consider here are the singular value decomposition (SVD) and the related reduced density matrix, as used in DMRG [1]. The simplest, and in principle the fastest, is the SVD, well known from linear algebra [27]. For example, for the left-handed orthogonality constraint on A^s_{ij}, where we have re-inserted the matrix indices i, j, we

consider s, i to be a single index of dimension dm, giving an ordinary $dm \times m$-dimensional matrix, and carry out the singular value decomposition,

$$A_{ij}^s = \sum_{kl} U_{(si)k} D_{kl} (V^\dagger)_{lj}, \tag{3}$$

where U is column-orthogonal, $U^\dagger U = 1$, and $V^\dagger$ is row-orthogonal, $V^\dagger V = 1$. D is a non-negative diagonal matrix containing the singular values. This form coincides with the Schmidt decomposition, where D gives the coefficients of the wavefunction in the Schmidt basis [21]. The matrix U therefore satisfies the left-handed orthogonality constraint, so we use this as the updated A matrix, and multiply the A matrix on the right by $DV^\dagger$. This implies that the A matrix on the right is no longer orthonormalized (even if it was originally), but we can apply this procedure iteratively to shift the non-orthogonal A matrix to the boundary—or even beyond it—at which point the 1×1 A matrix coincides with the norm of the wavefunction. An important point here is that we can choose to discard some of the states, typically those that have the smallest singular value. This reduces the matrix dimension m, at the expense of introducing an approximation to our wavefunction, such that the squared norm of the difference of our approximate and exact wavefunctions is equal to the sum of the squares of the discarded singular values. Note, however, that the singular values only correspond to the coefficients of the Schmidt decomposition if all of the remaining A matrices are orthogonalized according to equation (2). If this is not the case, the singular values are not useful for determining which states can be safely discarded.

Alternatively, we can construct the reduced density matrix, obtained by tracing over half of the system. This is achieved by

$$\rho_{ij}^{s's} = \sum_k A_{ik}^{s'*} A_{jk}^s, \tag{4}$$

which is a $dm \times dm$ matrix, with m eigenvalues coinciding with the values on the diagonal of D^2, and the remaining eigenvalues are zero. Again, the eigenvalues are only meaningful if the remaining A matrices are appropriately orthogonalized. The utility of the density matrix approach over the SVD is that we can introduce mixing terms into the density matrix which can have the effect of stabilizing the algorithm and accelerating the convergence, which is further discussed in section 4.

The overlap of two MPS is an operation that appears in many contexts. For wavefunctions this gives the fidelity of the two states, and for operators this is equivalent to the operator inner product $\langle A|B \rangle = \mathrm{Tr}\, A^\dagger B$ which induces the Frobenius norm. Direct expansion of the MPS form yields

$$\langle A|B \rangle = \sum_{\{s_i\}} (\mathrm{Tr}\, A^{s_1*} A^{s_2*} \ldots)(\mathrm{Tr}\, B^{s_1} B^{s_2} \ldots)$$
$$= \mathrm{Tr} \sum_{\{s_i\}} (A^{s_1*} \otimes B^{s_1})(A^{s_2*} \otimes B^{s_2}) \ldots. \tag{5}$$

Due to the open boundary conditions, the direct product $E_1 = A^{s_1*} \otimes B^{s_1}$ reduces to an ordinary $m \times m$ matrix, after which one can construct successive terms recursively, via

$$E_n = \sum_{s_n} A_n^{s_n \dagger} E_{n-1} B_n^{s_n}, \tag{6}$$

with an analogous formula if we wish to start at the right-hand side of the system and iterate towards the left boundary:

$$F_n = \sum_{s_n} A_n^{s_n} F_{n+1} B_n^{s_n\dagger}. \tag{7}$$

For the purposes of numerical stability, it is advisable for the A and B matrices to be orthogonalized in the same pattern, that is, E matrices are associated exclusively with the left-hand orthogonality constraint and F matrices are associated with the right-hand orthogonality constraint. If we iterate all the way to the boundary, the E (or F) matrix ends up as a 1×1 matrix that contains the final value of the fidelity. Alternatively, we can iterate from both ends towards the middle and calculate the fidelity as $\mathrm{Tr}\, EF^\dagger$.

2.2. Local updates

The key notion of the matrix product scheme is that of *local updates*; that is, we modify, typically through some iterative optimization scheme, one (or perhaps a few) A matrix while keeping the remainder fixed. A useful and flexible alternative is the *center matrix formulation*, where, instead of modifying an A matrix directly, we introduce an additional matrix C into the MPS:

$$A^{s_1} A^{s_2} \cdots A^{s_n} C A^{s_{n+1}} \cdots A^{s_L}. \tag{8}$$

This allows us to preserve orthogonality of the matrices at all times; matrices A^{s_i} for $i \leq n$ are normalized always according to the left-handed constraint, and matrices for $i > n$ are normalized according to the right-handed constraint. We directly modify only the matrix C which simplifies the local optimization problem as C is just an ordinary matrix. To introduce the local degrees of freedom, say for the $|s_n\rangle$ states, we *expand* the basis for A^{s_n}. That is, we replace the $m \times m$-dimensional matrices A^{s_n} with $m \times dm$ matrices A'^{s_n}, given by

$$A'^{s_n}_{ij} = \delta_{id+s_n, j}, \tag{9}$$

and introduce the $dm \times m$-dimensional center matrix

$$C_{jk} = A^{s_n}_{ik}, \tag{10}$$

with $j = \mathrm{i}d + s_n$ running over dm states. This does not change the physical wavefunction, as $A'^{s_n} C = A^{s_n}$. Similarly, we can expand the basis for the A matrix on the right side of C, to give the effect of modifying either a single A matrix, or two (or more) at once. In the center matrix formulation, the singular value decomposition required for truncating the basis is simply the ordinary SVD on the matrix $C = UDV^\dagger$, and we multiply (for a right-moving iteration) $A'^{s_n} U$, which preserves the left-handed orthogonality constraint, and $DV^\dagger A^{s_{n+1}}$ which is not orthogonal, but becomes so when we again expand the basis to construct the new C matrix. The density matrix in the center matrix formulation is simply $\rho = CC^\dagger$ or $\rho = C^\dagger C$ for left and right moving iterations respectively. For readers already familiar with DMRG, the center matrix corresponds exactly with the matrix form of the *superblock wavefunction* [29].

2.3. Matrix product arithmetic

The utility of the MPS approach is realized immediately upon attempting manipulations on the wavefunction equation (1). Suppose we have two distinct MPS, defined over the same local Hilbert spaces,

$$
\begin{aligned}
|\Psi_A\rangle &= A^{s_1} A^{s_2} \ldots A^{s_L} |s_1\rangle |s_2\rangle \ldots \\
|\Psi_B\rangle &= B^{s_1} B^{s_2} \ldots B^{s_L} |s_1\rangle |s_2\rangle \ldots .
\end{aligned}
\tag{11}
$$

The superposition $|\psi_C\rangle = |\Psi_A\rangle + |\Psi_B\rangle$ is formed by taking the sum of the matrix products, $A^{s_1} A^{s_2} \ldots + B^{s_1} B^{s_2} \ldots$, which can be factorized into a new MPS, $|\Psi_C\rangle = C^{s_1} C^{s_2} \ldots$, with

$$
C^{s_i} = A^{s_i} \oplus B^{s_i},
\tag{12}
$$

or, with the notation that A, B, C are matrices of kets (or, in the case of MPOs, local operators), this operation is simply $C = A \oplus B$. To preserve the one-dimensional basis at the boundaries, the direct sum is replaced by a concatenation of columns or rows for the left and right boundary, respectively. This procedure results in an MPS of increased dimension, $m_C = m_A + m_B$. Thus, after constructing these matrices we need to re-orthogonalize the state, and then we can, if necessary, truncate the basis size to a given truncation error, which is well defined here and measures the exact difference between the original and truncated wavefunctions. Alternatively, the normalized and truncated MPS $|\Psi_C\rangle$ can be constructed directly by calculating the overlap matrices E between $|\Psi_A\rangle$ and $|\Psi_B\rangle$. From the E matrices introduced in equation (6), we can construct directly the orthogonalized reduced density matrices of $|\Psi_C\rangle$ and truncate the basis as required, in a single step. This approach has better computational scaling than the two-step procedure of first orthogonalizing and then truncating, especially when the number of MPS in the superposition is large. But in general, iterative optimization approaches, where we use a DMRG-like algorithm to optimize the overlap $\langle\Psi_C|(|\Psi_A\rangle + |\Psi_B\rangle)$, have even better performance scaling with large m or more states in the superposition.

3. Operators

A useful generalization of the MPS structure equation (1) is to use it to represent an operator (an MPO) instead of a wavefunction. This has been used before for calculating finite-temperature density matrices [12], but here we instead want to use this structure to represent the Hamiltonian operator itself. All Hamiltonian operators with finite-range interactions have an *exact* MPS representation with a relatively small matrix dimension M. For example, the Ising model in a transverse field has a dimension $M = 3$ and the fermionic Hubbard model has dimension $M = 6$. We use the capital letter to distinguish from the dimension of the wavefunction, m. Similarly, the local dimension of the upper index is denoted here by D, which is usually just equal to d^2, but slightly complicated in the case of non-Abelian quantum numbers (see section 5.2).

We denote an MPO by the form

$$
\sum_{\{s_i, s_i'\}} M^{s_1' s_1} M^{s_2' s_2} \cdots M^{s_L' s_L} \quad |s_1'\rangle\langle s_1| \otimes |s_2'\rangle\langle s_2| \cdots,
\tag{13}
$$

where again we require that the first and last dimensions are $M = 1$, for open boundary conditions. Again, where possible we omit the local indices s', s and regard equation (13) as a sequence of operator-valued matrices.

The orthogonality constraint used previously for the MPS, equation (2), is not appropriate for Hamiltonian operators. When applied to an operator, the usual orthogonality constraints utilize the (Frobenius) operator norm, which scales as the square root of the dimension of the Hilbert space and thus exponentially in the lattice size. Thus with this normalization, components of an MPO Hamiltonian, such as the identity operator or some interaction term, tend to differ in magnitude by some factor that can easily become very large. Arithmetic manipulations on such quantities are a recipe for catastrophic cancellation [27] resulting in loss of precision. Mixing operators with a unitary transformation (for example $O'_1 = O_1 \cos \theta + O_2 \sin \theta, O'_2 = -O_1 \sin \theta + O_2 \cos \theta$) will lead to a disaster if O_1 and O_2 differ by a sufficiently large order of magnitude, $\sim 10^{16}$ for typical double-precision floating point arithmetic. But such rotations are inevitable in the orthogonalization procedure because in general the operator inner product $\langle O_1 | O_2 \rangle = \text{Tr}\, O_1^\dagger O_2$ will not be zero. Instead we completely avoid mixing different rows/columns of the operator M matrices, only collapsing a row or column if it is exactly parallel with another row or column. In this case, the actual norm of each component of the M matrices is irrelevant (but see also the discussion of the single-site algorithm in section 4) as they are never mixed with each other. For physical Hamiltonian operators this remains essentially optimal, with the minimum possible matrix dimension M. The only operators for which this orthogonalization scheme does not produce an optimal representation are operators that have a form analogous to an AKLT [28] state where the local basis states of the $S = 1$ chain are replaced by local operators. The resulting operator contains an exponentially large number of N-body interactions for all $N \to \infty$. We know of no physical Hamiltonians of this form.

Given a Hamiltonian as a sum of finite-range interactions, it is possible to construct the operator M matrices such that they are entirely lower (or upper) triangular matrices: thus in principle we can 'normalize' the matrices via some kind of generalized LU or QR decomposition. In practice we do not need to do this, as the Hamiltonian can easily be constructed in lower-triangular form from the beginning. Imposing, again without loss of generality, that the top-left and bottom-right components of the operator M matrices are equal to the identity operator I, we can construct the sum of 1-site local terms $H = \sum_i X_i$ as a position-independent MPS:

$$M^{s's} = \begin{pmatrix} \delta^{s's} & 0 \\ X^{s's} & \delta^{s's} \end{pmatrix}, \tag{14}$$

or, suppressing the local indices,

$$M = \begin{pmatrix} I & 0 \\ X & I \end{pmatrix}, \tag{15}$$

which we regard as a 2×2 matrix, the elements of which are $d \times d$-dimensional local operators. For nearest-neighbor terms, $H = \sum_i X_i Y_{i+1}$, we have

$$M = \begin{pmatrix} I & 0 & 0 \\ Y & 0 & 0 \\ 0 & X & I \end{pmatrix}, \tag{16}$$

with the obvious generalization to N-body terms. The direct sum and direct product of lower triangular matrices is itself lower triangular, thus this form can be preserved throughout all operations. For open boundary conditions, the left (right) boundary $1 \times M$ (or $M \times 1$) matrices are initialized to $(0, \ldots, 0, I)$ and $(I, 0, \ldots, 0)^{\mathrm{T}}$, respectively.

The principal advantage of formulating the Hamiltonian operator (and indeed, *all* operators needed in the calculation) in this way is that it can be manipulated extremely easily, amounting to a symbolic computation on the operators. This is in contrast to the ad hoc approach used in past DMRG and MPS approaches where the block transformations required for each operator are encoded manually, with limited scope for arithmetic operations. In particular, the sum of operators is achieved via equation (12). Products of operators are achieved by matrix direct product; given MPOs A and B, the product $C = AB$ is given by the matrices

$$C^{s's} = \sum_t A^{s't} \otimes B^{ts}, \tag{17}$$

or simply $C = A \otimes B$, where the ordinary product operation on the operator-valued matrix elements of A and B is understood. An implication of this is that the square of an MPO has a matrix dimension of at most M^2, which, since M is usually rather small, means that it is quite practical to calculate expectation values for higher-order moments, for example of the *variance*

$$\sigma^2 = \langle H^2 \rangle - \langle H \rangle^2 \ = \langle (H - E)^2 \rangle, \tag{18}$$

which has been mentioned previously [31] as it gives a rigorous *lower bound* on the energy (although with no guarantee that this corresponds to the ground state). In practice this lower bound is too wide to be useful in all but the simplest cases, but of more interest is the property that the variance is, to first order, proportional to the squared norm of the difference between the exact and numerical wavefunctions, and therefore also proportional to the truncation error [33] (see section 4). Thus, this quantity gives a quantitative estimate of the goodness of the wavefunction even for situations where the truncation error is not available. For our numerical MPS algorithms the variance takes the role of the precision ϵ in numerical analysis [27] via $\epsilon \sim \sqrt{\sigma^2}$.

Of a similar form to the product of two operators, the action of an operator M on a wavefunction $|A\rangle$ gives a wavefunction $|B\rangle$ with matrix elements,

$$B^{s'} = \sum_s M^{s's} \otimes A^s, \tag{19}$$

which similarly simplifies to $B = M \otimes A$, upon regarding M and A as matrices of operators and kets, respectively. The MPO formulation also gives a natural form for the evaluation of expectation values, similarly to the fidelity of equation (5):

$$\langle A|M|B \rangle = \sum_\alpha \mathrm{Tr}\, E_n^\alpha F_n^{\alpha\dagger} \,, \tag{20}$$

where the E and F matrices now have an additional index α that represents the terms in
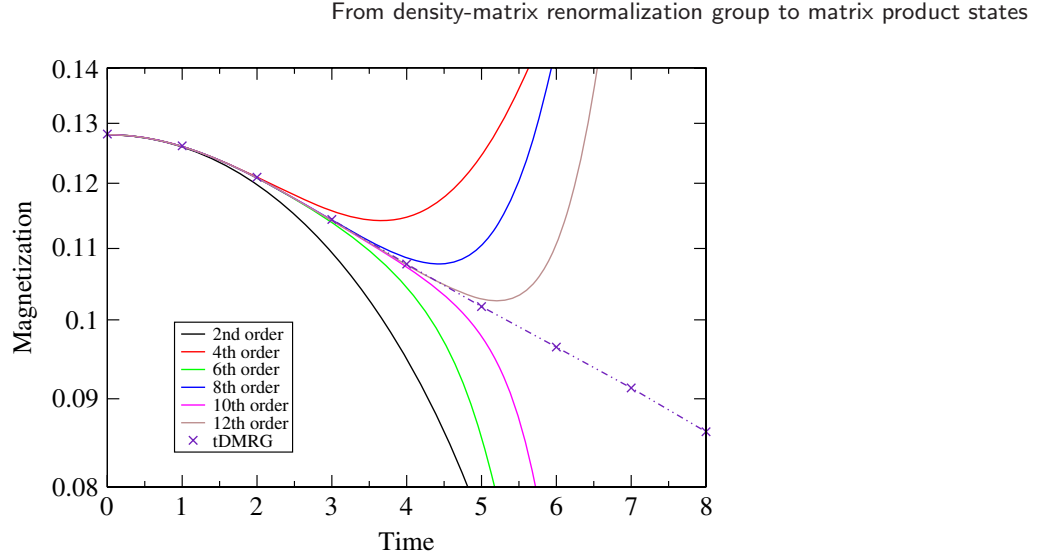
Figure 1. Polynomial expansion for the relaxation of the impurity magnetization in the SIAM, up to order t^{12}. The $\times$ symbols denote the impurity magnetization calculated via adaptive time DMRG, while the dashed line is a guide to the eye. Parameters of the calculation were (in units of bandwidth), $\Gamma = 0.1, U = 0.2, h_0 = 0.1, \epsilon_{d0} = 0.05$, on a log-discretized Wilson chain with $\Lambda = 1.8$. At time $t = 0$, the Hamiltonian was switched to $h_1 = 0$ and $\epsilon_{d1} = -0.1$.

the MPO M matrix, with a recursive definition

$$
\begin{aligned}
E_n^{\alpha'} &= \sum_{s',s,\alpha} M_{\alpha'\alpha}^{s's} A^{s_n'\dagger} E_{n-1}^{\alpha} B^{s_n} \\
F_n^{\alpha'} &= \sum_{s',s,\alpha} M_{\alpha'\alpha}^{s's} A^{s_n'} F_{n+1}^{\alpha} B^{s_n\dagger}
\end{aligned}
\tag{21}
$$

where again we can either iterate all the way to a boundary, at which point the α index collapses down to one-dimensional and the E^{α} or F^{α} are 1×1-dimensional matrices containing the expectation value, or we can iterate from both boundaries and meet at the middle, where our expectation value is given by the scalar product equation (20).

Incidentally, given that the identity operators occur in a fixed location in the operator M matrix (i.e. at the top left and bottom right of the M matrix) this fixes the index α of the reduced Hamiltonian and identity operators for the left and right partitions of the system. That is, in the application of the Hamiltonian E^{α}, F^{α} matrices to the wavefunction we are guaranteed that the $\alpha = 1$ component of $E^{\alpha} \otimes F^{\alpha\dagger}$ corresponds precisely to $H_L \otimes I_R$, and the $\alpha = M$ component corresponds to $I_L \otimes H_R$. Thus, even after an arbitrary series of MPO computations we can still identify exactly which component of the E, F matrices corresponds to the block Hamiltonian. This is useful for eigensolver preconditioning schemes [30], for example to change to a basis where the block Hamiltonian is diagonal.

As an example of the utility of this approach, figure 1 shows the time evolution of the magnetization of the impurity spin in the single impurity Anderson model (SIAM), where the ground state is obtained with a small magnetic field which is then turned off at time $t = 0$. The MPS operator approach readily allows the evaluation of the commutators

required for a small t expansion of the expectation value of an observable in the Heisenberg picture:

$$A(t) = A + \frac{\mathrm{i}t}{\hbar}[H, A] - \frac{t^2}{2!\hbar^2}[H, [H, A]] - \frac{\mathrm{i}t^3}{3!\hbar^3}[H, [H, [H, A]]] + \cdots. \qquad (22)$$

Since the number of terms in the repeated commutator will, in general, increase exponentially the accessible timescales from such an expansion are clearly limited. Nevertheless this is a very fast and accurate way to obtain short-timescale dynamics, and in this example twelfth order is easily enough to determine the T_1 relaxation rate. For this calculation, the terms up to t^8 took a few seconds to calculate, while the t^{10} term took 6 min and the t^{12} term took just over an hour on a single-processor 2 GHz Athlon64. This time was divided between calculating the MPO matrix elements (the dimension of which was just over 2500 at the impurity site) and calculating the expectation value itself.

4. DMRG

We now have all of the ingredients necessary to construct the DMRG algorithm for determining the ground state. Indeed, given the previous formulations, the DMRG itself is rather simple; using the center matrix formulation, we iteratively improve the wavefunction locally by using the center matrix C as input to an eigensolver for the ground state of the Hamiltonian. The details of this calculation are precisely as for DMRG, already covered elsewhere [29].

An important component of DMRG, which has been neglected in some matrix product approaches, is the truncation error. If only a single site is modified at a time, the maximum number of non-zero singular values is bounded by the matrix dimension m: thus the matrix dimension cannot be incrementally increased as the calculation progresses. Some way of avoiding this limitation is practically essential for a robust algorithm. The original DMRG formulation [1] solved this problem by modifying two A matrices simultaneously, equivalent to expanding the matrix dimension for both the left and right A matrices in equation (8) so that the center matrix has dimension $md \times md$. A scheme for single-site algorithms that avoids the limit on the singular values was introduced by White [34], which uses a mixed density matrix constructed by a weighted sum of the usual reduced density matrix and a perturbed density matrix formed by applying the E^α matrices (on a right-moving sweep) or F^α matrices (on a left-moving sweep) of the Hamiltonian:

$$\rho' = \rho + c \sum_\alpha E^\alpha \rho E^{\alpha\dagger}, \qquad (23)$$

where c is some small factor that fixes the magnitude of the fluctuations. This solves nicely the problem of the bound on the number of singular values and introduces fluctuations into the density matrix that give the algorithm good convergence properties, often better than the two-site algorithm. A minor problem is that the scaling of the E^α matrices is not well defined, in that we can scale the E^α matrices by an arbitrary $M \times M$ non-singular matrix $X_{\alpha'\alpha}$, while at the same time scaling the F matrices by X^{-1}. For one- and two-site terms, there is an 'obvious' scaling factor to use, whereby the scaling factors are chosen such that the (Frobenius) operator norm of the E and F matrices are identical, but I do not know how this would apply more generally. An alternative that appears

interesting is to apply the full Hamiltonian to a density operator for the full system, $\rho' = \rho + c\,\mathrm{Tr}_R\,H(\rho \otimes I)H$, constructed from the left (right) reduced density matrix and the right (left) identity operator, followed by a trace over the right (left) partition. In MPS form, this operation is

$$\rho' = \rho + c \sum_{\alpha,\beta} E^\alpha \rho E^{\beta\dagger} G_{\alpha\beta}, \qquad (24)$$

where $G_{\alpha\beta} = \mathrm{Tr}\,F^{\alpha\dagger}F^\beta$ is an $M \times M$ coefficient matrix. However, this scheme often fails; incorporating the $G_{\alpha\beta}$ matrix reduces the fluctuations such that $E^\alpha \rho E^{\beta\dagger}\,G_{\alpha\beta}$ differs little from ρ itself and the algorithm typically fails to reach the ground state. The single-site algorithm [34] corresponds to choosing $G_{\alpha\beta} = \delta_{\alpha\beta}$. A useful compromise appears to be using only the *diagonal* elements, such that $G_{\alpha\beta} = \delta_{\alpha\beta}\,\mathrm{Tr}\,F^{\alpha\dagger}F^\beta$, but this is surely not the last word on this approach. Both the two-site and mixed single-site algorithm inevitably result in a reduction in the norm of the wavefunction by truncating the smallest non-zero eigenvalues of the density matrix. The sum of the discarded eigenvalues, summed over all iterations in one sweep, is equal to the truncation error τ, familiar from DMRG [1] (but note that it is common in the literature to quote an average or maximum truncation error *per site*). This quantity is useful in giving an estimate of the error in the wavefunction if this is, for a properly converged wavefunction, proportional to the norm of the difference between the exact ground state and the numerical approximation. The presence of the truncation error explains why the bare single-site algorithm, despite having slightly better variational wavefunction than the two-site or mixed single-site algorithms [35], converges much slower; the single-site algorithm is a highly constrained optimization within an m-dimensional basis, whereas the two-site and mixed single-site algorithms are selecting the optimal m basis states out of a pool of a much larger set of states, namely the discarded states at each iteration (total $\sim Lm$ states). While the notion of truncation error remains useful in MPS algorithms, for the purposes of error analysis we much prefer the variance equation (18) as being a direct measure of the accuracy of the wavefunction, independent of the details of a particular algorithm [33].

Low-lying excited states can be constructed using this algorithm. This has been done in the past in DMRG by targeting multiple eigenstates in the density matrix, but the MPS formulation allows a substantial improvement. Namely, it is easy to incorporate into the eigensolver a constraint that the obtained wavefunction is orthogonal to an arbitrary set of predetermined MPSs. That is, after constructing the MPS approximation to the ground state, we can, as a separate calculation, determine the first excited state by running the algorithm again with the constraint that our obtained wavefunction is orthogonal to the ground state. This is achieved by constructing the E matrices that project the set of states to orthogonalize against onto the local Hilbert space. These matrices are precisely those used in constructing the fidelity, equation (5), thus given the center matrix of some state C_X, we project this onto the current Hilbert space $C_X' = EC_XF^\dagger$, and as part of the eigensolver, orthogonalize our center matrix against this state. This is a very fast operation, much faster than even a single Hamiltonian-wavefunction multiplication. So it is quite practical to orthogonalize against a rather large number of states, the practical limit is rather on numerical limitations in orthogonalizing the Krylov subspace in the eigensolver. If this is combined with an eigensolver capable of converging to eigenvalues in the middle of the spectrum (say, the lowest eigenvalue larger than some bound E_0), then

we need only a small number of states to orthogonalize against, say half a dozen states immediately above E_0 in energy. In our numerical tests it seems to be rather common to skip eigenvalues, which is why we cannot simply orthogonalize against a single state. With a larger number of states to orthogonalize against, skipping eigenvalues is less of a problem as we are likely to recover the missing eigenstate on a later calculation. Using this approach, quantities such as the level spacing statistics can be determined for system sizes far beyond exact diagonalization [33].

5. Quantum numbers

An important feature of matrix product states is that they can easily be constrained by quantum numbers representing the global symmetries of the Hamiltonian, as long as the symmetry is not broken by the spatial geometry of the MPS. For example, internal rotational symmetries such as $U(1)$ and $SU(2)$ [36] can be maintained exactly, but for a real-space lattice we cannot utilize the momentum in the same way because the representation itself violates this symmetry[1]. To achieve this, we impose a symmetry constraint on the form of the A matrices, so that they are *irreducible tensor operators*. That is, under a symmetry rotation the matrix A^s for each local degree of freedom s transforms according to an irreducible representation $D(j_s)$ of the global symmetry group. This is a very general procedure, that is applicable to essentially all MPS approaches and generalizations thereof.

5.1. Abelian symmetries

For Abelian symmetries, the representations are one-dimensional; therefore the set of quantum numbers labeling the irreducible representations also forms a group, which we can write as

$$D(j) \otimes D(k) = D(j + k), \tag{25}$$

for two representations $D(j)$ and $D(k)$, where $j + k$ denotes the group operation. Thus to incorporate Abelian symmetries into the algorithm we simply attach a quantum number to all of the labels appearing in the MPS, with the constraint that each A matrix transforms irreducibly, so that the only non-zero matrix elements are

$$A^k_{q'q}, \qquad \text{with } q' = q + k, \tag{26}$$

where k, q', q are the quantum numbers attached to the local basis state and left and right matrix basis states, respectively. We have suppressed here indices not associated with a quantum number, a convention which will be followed for the remainder of the paper.

By convention, for our open boundary condition MPS we choose the right-hand vacuum state to have quantum number zero. The symmetry constraint equation (26) then implies that the quantum number at the left-hand vacuum will denote how the state as a whole transforms (the *target state* in DMRG terminology). This is the only real difference between DMRG and MPS algorithms, in that the DMRG convention is to construct both blocks starting from a scalar (quantum number zero) vacuum state, so

---

[1] See also [37] for a real-space approach to constructing momentum eigenstates.

that the superblock wavefunction is a tensor product of two ket states:

$$|\Psi\rangle = \sum_{uv} \psi_{uv}|u\rangle \otimes |v\rangle, \qquad u + v = \text{target}, \tag{27}$$

whereas for the MPS formulation the superblock wavefunction is represented by a scalar operator with a tensor product basis $|u\rangle \otimes \langle v|$ with quantum numbers $u = v$. This means that, in contrast to the usual formulation of DMRG, the target quantum number is not encoded in the superblock but rather in the left vacuum state. A consequence of this is that DMRG is capable of representing simultaneously states with different quantum numbers, but an MPS is not. This is an important detail, for example in the calculation of dynamical correlations, as both the correction vector [38] and the similar DDMRG [39] algorithm require a basis optimized for both the ground-state $|\Psi\rangle$ and the so-called Lanczos vector $A|\Psi\rangle$, where A is some operator that may not be scalar. However, the MPS formulation allows significant optimizations to these algorithms whereby the calculation of the ground state is decoupled from that of the Lanczos vector [31, 32] and the two need never appear in the same basis.

5.2. Non-Abelian symmetries

If the symmetry group is large enough that some elements do not commute with each other, then it is no longer possible to construct a basis that simultaneously diagonalizes all of the generators; hence the approach of the previous section needs some modification. Instead, we label the basis states by quantum numbers that denote the representation, which is no longer simply related to the group elements themselves as the representations are, in general, no longer ordinary numbers, but instead are matrices of dimension >1, and equation (25) no longer applies. For $SU(2)$, we choose to label the representations by the total spin s, being related to the eigenvalue of the spin operator, $\boldsymbol{S}^2 = s(s+1)$. Assuming that all of the required operations can be formulated in terms of manipulations of these representations, we have a formulation that is *manifestly* $SU(2)$ invariant; the rotational invariance is preserved at all steps and at no time in the calculation is it necessary to choose an axis of quantization [36]. This supersedes the earlier approach based on the Clebsch–Gordan coefficients [40]. The non-Abelian formulation is an important optimization, because it increases the performance of the algorithm by an order of magnitude or more [36] compared with the Abelian case, while enabling more accurate and detailed information about the ground-state magnetization. The basic ingredient that enables this rotationally invariant construction is the Wigner–Eckart theorem [41], which we can state as: when written in an angular momentum basis, each matrix element of an irreducible tensor operator is a product of two factors, a purely angular momentum-dependent factor (the 'Clebsch–Gordan coefficient') and a factor that is independent of the projection quantum numbers (the 'reduced matrix element'). We formulate the algorithm in such a way that we store and manipulate only the reduced matrix elements, factorizing out completely the Clebsch–Gordan coefficients. The efficiency improvement resulting from the non-Abelian formulation is that the matrix dimensions m and M now refer to the number of irreducible representations in the basis, which is typically much smaller than the total degree of the representation. For a scalar state, this equivalence is precise: a single representation of degree N in the non-Abelian approach results in N degenerate eigenstates when the symmetry is not used, with a corresponding improvement

in efficiency. We do not give here a full introduction to the theory of quantum angular momentum, rather we present, in the style of a reference, the important formulas required to manipulate MPS wavefunctions and operators. For a comprehensive introduction see, for example, [42, 43].

Using the normalization convention of Biedenharn [42], we define the matrix elements of a tensor operator $\boldsymbol{T}^{[k]}$ transforming as a rank k tensor under $SU(2)$ rotations, as

$$\langle j'm'|T_M^{[k]}|jm\rangle = \langle j'\|\boldsymbol{T}^{[k]}\|j\rangle C_{mMm'}^{j\ k\ j'}, \tag{28}$$

where $C_{\cdots}^{\cdots}$ is the Clebsch–Gordan (CG) coefficient, $j'j$ label the representation of $SU(2)$, and $m = -j, -j+1, \ldots, j$ and $m' = -j', -j'+1, \ldots, j'$ label the projections of the total spin onto the z axis. Using the orthogonality of the Clebsch–Gordan coefficients, this defines the reduced matrix elements,

$$\langle j'\|\boldsymbol{T}^{[k]}\|j\rangle = \sum_{mM} C_{mMm'}^{j\ k\ j'}\langle j'm'|T_M^{[k]}|jm\rangle, \tag{29}$$

where m' is arbitrary. Note that this normalization is *not* the same as that used by Varshalovich *et al* [43], who instead use an additional factor $\sqrt{2j'+1}$ in the reduced matrix elements. This is a tradeoff; some formulas simplify slightly with this normalization (for example, equation (38) below), but the normalization used here has the advantage that the reduced matrix elements of scalar operators (with $k = 0$) coincide with the actual matrix elements as all of the relevant Clebsch–Gordan coefficients are equal to unity. Given the definition of the reduced matrix elements, we formulate the remaining formulas without further reference to the axis of quantization, except as an intermediate step to relate the reduced matrix elements prior to factorizing out the Clebsch–Gordan coefficients.

The coupling of two operators is just as for the coupling of ordinary spins:

$$\left[\boldsymbol{S}^{[k_1]} \times \boldsymbol{T}^{[k_2]}\right]^{[k]}, \tag{30}$$

which denotes the set of operators with components

$$\left[\boldsymbol{S}^{[k_1]} \times \boldsymbol{T}^{[k_2]}\right]_\mu^{[k]} = \sum_{\mu_1\mu_2} C_{\mu_1\mu_2\mu}^{k_1\,k_2\,k} S_{\mu_1}^{[k_1]} T_{\mu_2}^{[k_2]}. \tag{31}$$

Applying the Wigner–Eckart theorem gives, after a few lines of algebra,

$$\langle j'\|[\boldsymbol{S}^{[k_1]} \times \boldsymbol{T}^{[k_2]}]^{[k]}\|j\rangle = (-1)^{j+j'+k} \sum_{j''} \sqrt{(2j''+1)(2k+1)} \left\{\begin{array}{ccc} j' & k_1 & j'' \\ k_2 & j & k \end{array}\right\}$$

$$\times\ \langle j'\|\boldsymbol{S}^{[k_1]}\|j''\rangle\langle j''\|\boldsymbol{T}^{[k_2]}\|j\rangle, \tag{32}$$

where $\{\cdots\}$ denotes the $6j$ coefficient [42, 43].

A special case of the coupling law equation (32) that we will need is when the operators act on different spaces, such that they have a tensor product form

$$\begin{aligned} S_{\mu_1}^{[k_1]} &= T_{\mu_1}^{[k_1]}(1) \otimes I(2), \\ T_{\mu_2}^{[k_2]} &= I(1) \otimes T_{\mu_2}^{[k_2]}(2). \end{aligned} \tag{33}$$

Here $I(i)$ denotes the identity operator and $\boldsymbol{T}^{[k_i]}(i)$ is an irreducible tensor operator with respect to the angular momentum $\boldsymbol{J}(i)$ of part i of a two-part physical system ($i = 1, 2$). The total angular momentum of the system is $\boldsymbol{J} = \boldsymbol{J}(1) + \boldsymbol{J}(2)$. In this case, we write the

coupling as $[\boldsymbol{S}^{[k_1]}{\times}\boldsymbol{T}^{[k_2]}]^{[k]} \equiv [\boldsymbol{T}^{[k_1]}(1){\otimes}\boldsymbol{T}^{[k_2]}(2)]^{[k]}$. Repeated application of the Wigner–Eckart theorem to these tensor operators gives, after some algebra,

$$\langle j'(j_1'j_2'\alpha_1'\alpha_2')\|[\boldsymbol{T}^{[k_1]}(1){\otimes}\boldsymbol{T}^{[k_2]}(2)]^{[k]}\|j(j_1j_2\alpha_1\alpha_2)\rangle$$

$$= \begin{bmatrix} j_1 & j_2 & j \\ k_1 & k_2 & k \\ j_1' & j_2' & j' \end{bmatrix} \langle j_1'(\alpha_1')\|\boldsymbol{T}^{[k_1]}(1)\|j_1(\alpha_1)\rangle\langle j_2'(\alpha_2')\|\boldsymbol{T}^{[k_2]}(2)\|j_2(\alpha_2)\rangle, \tag{34}$$

where

$$\begin{bmatrix} j_1 & j_2 & j \\ j_1 & k_2 & k \\ j_1' & j_2' & j' \end{bmatrix} \equiv [(2j_1'+1)(2j_2'+1)(2j+1)(2k+1)]^{1/2} \begin{Bmatrix} j_1 & j_2 & j \\ k_1 & k_2 & k \\ j_1' & j_2' & j' \end{Bmatrix}, \tag{35}$$

and the term in curly brackets is the Wigner $9j$ coefficient, which can be defined as a summation over $6j$ coefficients [42]:

$$\begin{Bmatrix} j_1 & j_2 & j \\ k_1 & k_2 & k \\ j_1' & j_2' & j' \end{Bmatrix} \equiv (-1)^{j_1+j_2+j+k_1+k_2+k+j_1'+j_2'+j'}\sum_{j''}(-1)^{2j''}(2j''+1)$$

$$\times \begin{Bmatrix} j' & k_1 & j'' \\ k_2 & j & k \end{Bmatrix}\begin{Bmatrix} j' & j_2' & j_1' \\ j_1 & k_1 & j'' \end{Bmatrix}\begin{Bmatrix} j'' & j_1 & j_2' \\ j_2 & k_2 & j \end{Bmatrix}. \tag{36}$$

We can define an operator norm, corresponding to the usual Frobenius norm, such that

$$\|\boldsymbol{X}^{[k]}\|_{\text{frob}}^2 = \text{Tr}\,\boldsymbol{X}^{[k]}\cdot\boldsymbol{X}^{\dagger[k]} = \text{Tr}\,\boldsymbol{X}^{\dagger[k]}\cdot\boldsymbol{X}^{[k]}, \tag{37}$$

which is the scalar product of rank k tensors. After some arithmetic, we see that

$$\|\boldsymbol{X}^{[k]}\|_{\text{frob}}^2 = \sum_{j'j}(2j'+1)|\langle j'\|\boldsymbol{T}^{[k]}\|j\rangle|^2. \tag{38}$$

For the center matrix formalism, we need the transformation

$$\boldsymbol{A}^{[s]}{}_{ij} \rightarrow \sum_k C_{ik}\boldsymbol{A'}^{[s]}{}_{kj}, \tag{39}$$

where k is a $d \times m$-dimensional index that encapsulates both a s' and a j' index[2]: $k \simeq (s', j')$. Requiring $\boldsymbol{A'}^{[s]}{}_{kj}$ to satisfy the right orthogonality constraint, $A'A'^\dagger = 1$, this requires

$$\boldsymbol{A'}^{[s]}{}_{kj} = \delta_{j'j}\delta_{s's} \qquad [\text{with } k \simeq (s', j')], \tag{40}$$

with

$$C_{ik} = \boldsymbol{A}^{[s']}{}_{ij'}. \tag{41}$$

In the other direction, we need

$$\boldsymbol{A}^{[s]}{}_{ij} \rightarrow \sum_k \boldsymbol{A'}^{[s]}{}_{ik}C_{kj}, \tag{42}$$

---

[2] More precisely, k runs over each irreducible representation in the Clebsch–Gordan expansion of $D(s') \otimes D(j')$.

where $k \simeq (s', i')$. Requiring $\boldsymbol{A'}^{[s]}{}_{ik}$ to satisfy the left orthogonality constraint, $A'^\dagger A' = 1$, this requires

$$\boldsymbol{A'}^{[s]}{}_{ik} = \delta_{s's}\delta_{i'i}\sqrt{\frac{2k+1}{2i+1}} \qquad [\text{with } k \simeq (s', i')], \tag{43}$$

and

$$C_{kj} = \boldsymbol{A}^{[s']}{}_{i'j}\sqrt{\frac{2i'+1}{2k+1}}. \tag{44}$$

Matrix product operators contain a large number of indices, so to clarify the symmetry properties we temporarily introduce lower and upper indices to represent covariant and contravariant quantum numbers, respectively. The tensor coupling rules introduced previously follow the usual convention where an upper index is contracted with a lower index. Using this notation, an MPW A matrix has the form $\boldsymbol{A}_{[s]}{}^i{}_j$. The most natural definition for a matrix product operator generalizes this to two upper indices and three lower indices:

$$\boldsymbol{M}_{[k]}{}^{s'}{}_s{}^{j'}{}_j, \tag{45}$$

which transforms as the product of two rank k operators, with matrix elements

$$\langle s'q'; j'm'|M_r^{[k]}|sq; jm\rangle = \langle s'; j'\|\boldsymbol{M}^{[k]}\|s; j\rangle C_{qrq'}^{sks'}C_{mrm'}^{j\;kj'}. \tag{46}$$

Here the s', s indices denote the quantum numbers belonging to the local Hilbert space with z-axis projections of q', q, respectively; the j', j indices denote the quantum numbers of the matrix basis with z-axis projections m', m respectively, and the operator as a whole transforms as rank k with z-axis projection r. Typically an M matrix does not transform irreducibly so the k index must also be summed over as necessary. An important detail, which appears to be the only place in manipulating MPSs where the usual upper/lower convention for tensor contractions does not apply cleanly, is the product of an MPO and an MPW, which requires a contraction of the index s, which has the symmetry of a contraction of two *lower* indices, and then transforming the result index s' from upper to lower. This introduces an additional symmetry-dependent phase factor. For $SU(2)$, this factor is $(-1)^{s+k-s'}$, giving the rule

$$\boldsymbol{B}^{[s']} = (\boldsymbol{M} \otimes \boldsymbol{A})^{[s']} = \sum_s (-1)^{s+k-s'} \boldsymbol{M}^{[k]s's} \otimes \boldsymbol{A}^{[s]}, \tag{47}$$

where the tensor product operation is given by equation (34).

The product of a matrix product operator M with another matrix product operator N is

$$\boldsymbol{X}^{[x]s's} = \sum_t \boldsymbol{M}^{[m]s't} \otimes \boldsymbol{N}^{[n]ts}, \tag{48}$$

which corresponds to the ordinary (contracted) product in the local basis and the tensor product in the matrix basis, and therefore results in the product of a $6j$ and a $9j$ coefficient from equations (32) and (34), respectively.

For the evaluation of matrix elements, we need the tensorial form of the operation

$$F'^{a'}_{i'j'} = \sum_{s',s,i,j,a} M^{s's}_{a'a} A^{*s'}_{i'i} B^s_{j'j} E^a_{ij}. \tag{49}$$

On expanding out the reduced matrix elements, we see immediately that the coupling coefficient is

$$
\boldsymbol{F'}^{[a']}{}_{i'j'} = \sum_{a,i,j,k,s,s'} \begin{bmatrix} j & s & j' \\ a & k & a' \\ i & s' & i' \end{bmatrix} \boldsymbol{M}^{[k]}{}^{s's}_{a'a} \boldsymbol{A}^{[s']*}{}_{i'i} \boldsymbol{B}^{[s]}{}_{j'j} \boldsymbol{F}^{[a]}{}_{ij}. \tag{50}
$$

Conversely, from the left-hand side, we need the tensor form of

$$
E'^a_{ij} = \sum_{s',s,i',j',a'} E^{a'}_{i'j'} M^{s's}_{a'a} A^{*s'}_{i'i} B^s_{j'j}, \tag{51}
$$

which is

$$
\boldsymbol{E'}^{[a]}{}_{ij} = \sum_{a',i',j',k,s',s} \frac{2i'+1}{2i+1} \begin{bmatrix} j & s & j' \\ a & k & a' \\ i & s' & i' \end{bmatrix} \boldsymbol{E}^{[a']}{}_{i'j'} \boldsymbol{M}^{[k]}{}^{s's}_{a'a} \boldsymbol{A}^{[s']*}{}_{i'i} \boldsymbol{B}^{[s]}{}_{j'j}. \tag{52}
$$

On interchanging $A \leftrightarrow E$, $B \leftrightarrow F$, this becomes the equation for the projected action of an MPO on a MPW A matrix that gives, for example, the matrix–vector multiply operation in single-site DMRG without the center matrix formulation. If the center matrix is used, we want instead the operation

$$
C'_{i'i} = \sum_{j'jk} E^k_{i'j'} C_{j'j} F^{k*}_{ij}, \tag{53}
$$

where C and C' transform as scalars, i.e. the quantum numbers are constrained to $i' = i$, $j' = j$. This is essentially a scalar product $E \cdot F$, and the coupling coefficients drop out.

6. Conclusions

In this paper, we have presented an introduction to the MPS formulation of the DMRG algorithm for the calculation of ground and excited states of one-dimensional lattice Hamiltonians. The MPS formulation is extremely flexible, allowing the possibility for algorithms that act on several distinct wavefunctions at once. The simplest such algorithms are for the fidelity and expectation values involving unrelated wavefunctions, $\langle\psi|\phi\rangle$ and $\langle\psi|M|\phi\rangle$, which are difficult to extract from conventional DMRG. This gives access to new tools for the analysis of quantum phase transitions by measuring the scaling function and exponents for the fidelity between ground states as a function of the interaction strength. In addition, the MPS formulation allows optimized versions of algorithms for dynamical correlations [31, 32] and time evolution [17], which remains a fertile area for continued algorithmic improvements.

Finally, we note that, in the simulation of finite temperature states via a density operator or purification [12, 14] in the absence of dissipative terms that mix the particle numbers between the real and auxiliary systems, the symmetries of the system are *doubled*, such that the symmetries of the Hamiltonian are preserved by the real and auxiliary parts independently. For simulations in a canonical ensemble, this leads to a significant efficiency improvement that, as far as we know, has not yet been taken into consideration.

Acknowledgments

References

[1] White S R, 1992 *Phys. Rev. Lett.* **69** 2863
    White S R, 1993 *Phys. Rev.* B **48** 10345
[2] Klümper A, Schadschneider A and Zittartz J, 1991 *J. Phys. A: Math. Gen.* **24** L955
[3] Fannes M, Nachtergaele B and Werner R F, 1992 *Commun. Math. Phys.* **144** 443
[4] Klümper A, Schadschneider A and Zittartz J, 1992 *Z. Phys.* B **87** 281
[5] Derrida B, Evans M R, Hakim V and Pasquier V, 1993 *J. Phys. A: Math. Gen.* **26** 1493
[6] Östlund S and Rommer S, 1995 *Phys. Rev. Lett.* **75** 3537
[7] Chung M C and Peschel I, 2001 *Phys. Rev.* B **64** 064412
[8] Vidal G, Latorre J I, Rico E and Kitaev A, 2003 *Phys. Rev. Lett.* **90** 227902
[9] Schuch N, Wolf M M, Verstraete F and Cirac J I, 2007 *Preprint* 0705.0292
[10] Korepin V E, 2004 *Phys. Rev. Lett.* **92** 096402
[11] Verstraete F, Porras D and Cirac J I, 2004 *Phys. Rev. Lett.* **93** 227205
[12] Verstraete F, García-Ripoll J J and Cirac J I, 2004 *Phys. Rev. Lett.* **93** 207204
[13] Zwolak M and Vidal G, 2004 *Phys. Rev. Lett.* **93** 207205
[14] Feiguin A E and White S R, 2005 *Phys. Rev.* B **72** 220401
[15] Daley A J, Kollath C, Schollwöck U and Vidal G, 2004 *J. Stat. Mech.* P04005
[16] White S R and Feiguin A E, 2004 *Phys. Rev. Lett.* **93** 076401
[17] García-Ripoll J J, 2006 *New J. Phys.* **8** 305
[18] Zanardi P and Paunković N, 2006 *Phys. Rev.* E **74** 031123
[19] Cizzini M, Ionicioiu R and Zanardi P, 2006 *Preprint* cond-mat/0611727
[20] Zhou H-Q and Barjaktarevic J P, 2007 *Preprint* cond-mat/0701608
[21] Nielsen M A and Chuang I L, 2000 *Quantum Computation and Quantum Information* (Cambridge: Cambridge University Press)
[22] Legeza Ö and Sólyom J, 2006 *Phys. Rev. Lett.* **96** 116401
[23] Quan H T, Song Z, Liu X F, Zanardi P and Sun C P, 2006 *Phys. Rev. Lett.* **96** 140604
[24] Freidrich A, 2006 *PhD Thesis* RWTH-Aachen
[25] Legeza Ö, Gebhard F and Rissler J, 2006 *Phys. Rev.* B **74** 195112
[26] Vidal G, 2007 *Phys. Rev. Lett.* **98** 070201
[27] Higham N J, 2002 *Accuracy and Stability of Numerical Algorithms* (Philadelphia: Society for Industrial and Applied Mathematics)
[28] Affleck I, Kennedy T, Lieb E H and Tasaki H, 1987 *Phys. Rev. Lett.* **59** 799
[29] Schollwöck U, 2005 *Rev. Mod. Phys.* **77** 259
[30] McCulloch I P, 2001 *PhD Thesis* Australian National University
[31] Verstraete F, Weichselbaum A, Schollwöck U, Cirac J I and von Delft J, 2005 *Preprint* cond-mat/0504305
[32] Friedrich A, Kolezhuk A K, McCulloch I P and Schollwöck U, 2007 *Phys. Rev.* B **75** 094414
[33] McCulloch I P, 2007 in preparation
[34] White S R, 2005 *Phys. Rev.* B **72** 180403(R)
[35] Dukelsky J, Martín-Delgado M A, Nishino T and Sierra G, 1998 *Europhys. Lett.* **43** 457
[36] McCulloch I P and Gulácsi M, 2002 *Europhys. Lett.* **57** 852
[37] Porras D, Verstraete F and Cirac J I, 2006 *Phys. Rev.* B **73** 014410
[38] Kühner T D and White S R, 1999 *Phys. Rev.* B **60** 335
[39] Jeckelmann E, 2002 *Phys. Rev.* B **66** 045114
[40] McCulloch I P and Gulácsi M, 2001 *Phil. Mag. Lett.* **81** 447
    McCulloch I P and Gulácsi M, 2000 *Aust. J. Phys.* **53** 597
[41] Wigner E P, 1959 *Group Theory and Its Applications to the Quantum Mechanics of Atomic Spectra* (New York: Academic)
[42] Biedenharn L C and Louck J D, 1981 *Angular Momentum in Quantum Physics* (Reading, MA: Addison-Wesley)
[43] Varshalovich D A, Moskalev A N and Khersonskii V K, 1988 *Quantum Theory of Angular Momentum* (Singapore: World Scientific)
[44] Singh S, Zhou H-Q and Vidal G, 2007 *Preprint* cond-mat/0701427