Density Matrix Renormalization Group (DMRG) Application to Heisenberg Model

Phillip Helms

August 2017

Contents

List of Figures

1 MPS Representation of Quantum States

To describe how an arbitrary quantum state can be represented using a Matrix Product State (MPS), consider a one-dimensional lattice with L sites where each site has a d-dimensional state space $\{\sigma_i\}$. For example, if one considers a spin-1/2 system, $d = 2$ because at each site, the particle can be spin-up or spin-down. In the most general of representations, this quantum state can be expressed as:

$$\left|\psi\right\rangle = \sum_{\sigma_1,\cdots,\sigma_L} \boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} \left|\sigma_1 \cdots \sigma_L\right\rangle \tag{1}$$

where $\left|\psi\right\rangle$ represents the given state, $\left|\sigma_1 \cdots \sigma_L\right\rangle$ represents each possible configuration, and c is an array of weighting coefficients of size d^L. While c contains all information about the state of the system, without in-depth analysis, it provides little information about what is happening at each of the L sites. One advantage which is obtained when the state is represented as an MPS will be a localized expression of c.

1.1 Singular Value Decomposition Review

To make this transformation, extensive use of the singular value decomposition (SVD) is employed. Recall that any $N_A \times N_B$ matrix, M, can be deconstructed as follows

$$\boldsymbol{M} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\dagger \tag{2}$$

where $\boldsymbol{U}$, $\boldsymbol{S}$, and $\boldsymbol{V}$ are each matrices of dimensions $(N_A \times \min(N_A, N_B))$, $(\min(N_A, N_B) \times \min(N_A, N_B))$, and $(\min(N_A, N_B) \times N_B)$, and $\boldsymbol{V}^\dagger$ represents the transpose of matrix $\boldsymbol{V}$. These three resulting matrices have a number of key properties:

1. The columns of $\boldsymbol{U}$ are orthonormal meaning that $\boldsymbol{U}^\dagger\boldsymbol{U} = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix. Also, if $N_A \leq N_B$, then $\boldsymbol{U}\boldsymbol{U}^\dagger = \boldsymbol{I}$.
2. Conversely, the rows of $\boldsymbol{V}^\dagger$ are orthonormal with $\boldsymbol{V}^\dagger\boldsymbol{V} = \boldsymbol{I}$ and if $N_B \leq N_A$ then $\boldsymbol{V}\boldsymbol{V}^\dagger = \boldsymbol{I}$.
3. $\boldsymbol{S}$ is a purely diagonal matrix, with the singular values being in the vector $\boldsymbol{s}_i = \boldsymbol{S}_{ii}$. The number of non-zero singular values is the rank, r, of the matrix.
4. The optimal approximation of a matrix, M, of rank r as a matrix M' of rank r' is obtained by truncating $\boldsymbol{s}'$ and retaining only the first r' largest singular values. The approximate matrix is then calculated as $\boldsymbol{M}' = \boldsymbol{U}\boldsymbol{S}'\boldsymbol{V}^\dagger$.

1.2 Quantum State Decomposition

Now, starting with the arbitrary quantum state described by array $\boldsymbol{c}$, the system can be decomposed into an MPS. Initially, a matrix is defined, now using Einstein notation, as $\boldsymbol{\Psi}_{\sigma_1,(\sigma_2\cdots\sigma_L)} = \boldsymbol{c}_{\sigma_1,\cdots,\sigma_L}$. Performing an SVD of $\boldsymbol{\Psi}$ gives

$$\begin{aligned} \boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} &= \boldsymbol{\Psi}_{\sigma_1,(\sigma_2\cdots\sigma_L)} \\ &= \sum_{a_1}^{d} \boldsymbol{U}_{\sigma_1,a_1} \boldsymbol{s}_{a_1} \boldsymbol{V}^{\dagger}_{a_1,(\sigma_2\cdots\sigma_L)} \\ &= \sum_{a_1}^{d} \boldsymbol{U}_{\sigma_1,a_1} \boldsymbol{c}_{a_1,\sigma_2,\cdots,\sigma_L} \end{aligned} \tag{3}$$

We now extract d vectors from $\boldsymbol{U}$ such that $\boldsymbol{A}^{\sigma_1}_{a_1} = \boldsymbol{U}_{\sigma_1,a_1}$ and define a new matrix $\boldsymbol{\Psi}_{(a_1\sigma_2),(\sigma_3\cdots\sigma_L)} = \boldsymbol{c}_{\sigma_1,\cdots,\sigma_L}$. This results in the expression

$$\boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} = \sum_{a_1}^{d} \boldsymbol{A}^{sigma_1}_{a_1} \boldsymbol{\Psi}_{(a_1\sigma_2),(\sigma_3\cdots\sigma_L)}. \tag{4}$$

Performing the same procedure on the newly produced matrix $\boldsymbol{\Psi}$ gives:

$$\boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} = \sum_{a_1,a_2}^{d} \boldsymbol{A}^{\sigma_1}_{a_1} \boldsymbol{A}^{\sigma_2}_{a_1,a_2} \boldsymbol{\Psi}_{(a_2\sigma_3),(\sigma_4\cdots\sigma_L)} \tag{5}$$

This procedure can be continued until we reach the final site, thus producing an MPS representation of the given state

$$\boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} = \sum_{a_1\cdots a_{L-1}}^{d} \boldsymbol{A}^{\sigma_1}_{a_1} \boldsymbol{A}^{\sigma_2}_{a_1,a_2} \cdots \boldsymbol{A}^{\sigma_{L-1}}_{a_{L-2},a_{L-1}} \boldsymbol{A}^{\sigma_L}_{a_{L-1}} \tag{6}$$

or expressed using matrix multiplication as:

$$\boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} = \boldsymbol{A}^{\sigma_1} \boldsymbol{A}^{\sigma_2} \cdots \boldsymbol{A}^{\sigma_{L-1}} \boldsymbol{A}^{\sigma_L} \tag{7}$$

Also, note that the best approximate representation of the system can be achieved with lower rank matrices if we choose to sum to $r' < d$.

1.2.1 Gauge Degrees of Freedom

While the state was easily decomposed into an MPS, it is important to note that this specific MPS is not unique. To illustrate this, imagine that we insert an arbitrary invertible matrix

$\boldsymbol{X}$ and its inverse $\boldsymbol{X}^{-1}$ into Equation 7 such that

$$\boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} = \boldsymbol{A}^{\sigma_1}\boldsymbol{X}\boldsymbol{X}^{-1}\boldsymbol{A}^{\sigma_2}\cdots\boldsymbol{A}^{\sigma_{L-1}}\boldsymbol{A}^{\sigma_L} \tag{8}$$

We can now define $\boldsymbol{\Lambda}^{\sigma_1} = \boldsymbol{A}^{\sigma_1}\boldsymbol{X}$ and $\boldsymbol{\Lambda}^{\sigma_2} = \boldsymbol{X}^{-1}\boldsymbol{A}^{\sigma_2}$ to create a new, but equivalent, MPS

$$\boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} = \boldsymbol{\Lambda}^{\sigma_1}\boldsymbol{\Lambda}^{\sigma_2}\cdots\boldsymbol{A}^{\sigma_{L-1}}\boldsymbol{A}^{\sigma_L} \tag{9}$$

The inherent ability to express a given state as an infinite number of MPSs is referred to as the gauge degree of freedom and there exists three key MPS representations.

The first primary MPS expression is the one which was generated Equations 3 - 7. This is referred to as the Left-Canonical MPS since it is generated by taking SVDs of the left-most matrices. As a more formal definition of the Left-Canonical MPS, it is defined as the set of $\{A^{\sigma_i}\}$ where for all sites

$$\sum_{\sigma_i}(\boldsymbol{A}^{\sigma_i})^\dagger\boldsymbol{A}^{\sigma_i} = \boldsymbol{I} \tag{10}$$

Alternatively, if the MPS is built by sweeping from the right side towards the left side, it is the Right-Canonical MPS. This is done in a fashion equivalent to what was shown in Equations 3 - 7

$$\begin{aligned}
\boldsymbol{c}_{\sigma_1,\cdots,\sigma_L} =& \boldsymbol{\Psi}_{(\sigma_1\cdots\sigma_{L-1}),\sigma_L} \\
&= \sum_{a_{L-1}}^{d} \boldsymbol{U}_{(\sigma_1\cdots\sigma_{L-1}),a_{L-1}}\boldsymbol{s}_{a_{L-1}}\boldsymbol{V}^\dagger_{a_{L-1},\sigma_L} \\
&= \sum_{a_{L-1}}^{d} \boldsymbol{\Psi}_{(\sigma_1\cdots\sigma_{L-2}),(\sigma_{L-1}a_{L-1})}\boldsymbol{B}^{\sigma_L}_{a_{L-1}} \\
&\vdots \\
&= \sum_{a_1,\cdots,a_{L-1}}^{d} \boldsymbol{B}^{\sigma_1}_{a_1}\boldsymbol{B}^{\sigma_2}_{a_1,a_2}\cdots\boldsymbol{B}^{\sigma_{L-1}}_{a_{L-2},a_{L-1}}\boldsymbol{B}^{\sigma_L}_{a_{L-1}} \\
&= \boldsymbol{B}^{\sigma_1}\boldsymbol{B}^{\sigma_2}\cdots\boldsymbol{B}^{\sigma_{L-1}}\boldsymbol{B}^{\sigma_L}
\end{aligned} \tag{11}$$

Similar to the Left-Canonical MPS, the Right-Canonical arrangement is formally defined as the system where

$$\sum_{\sigma_i}\boldsymbol{B}^{\sigma_i}(\boldsymbol{B}^{\sigma_i})^\dagger = \boldsymbol{I} \tag{12}$$

is satisfied at each site.

The final common representation is referred to as the Mixed-Canonical MPS and involves decomposing towards the center from both sides. To construct this, we perform i left-normalized decompositions to arrive at

$$c_{\sigma_1,\cdots,\sigma_L} = \sum_{a_1,\cdots,a_i}^{d} A_{a_1}^{\sigma_1} \cdots A_{a_{i-1},a_i}^{\sigma_i} s_{a_i} V_{a_i,(\sigma_{i+1}\cdots\sigma_L)}^{\dagger} \tag{13}$$

Now, the matrix V can be decomposed using the Right-Canonical methodology to find

$$c_{\sigma_1,\cdots,\sigma_L} = \sum_{a_1,\cdots,a_L}^{d} A_{a_1}^{\sigma_1} \cdots A_{a_{i-1},a_i}^{\sigma_i} s_{a_i} B_{a_i,a_{i+1}}^{\sigma_{i+1}} \cdots B_{a_{L-1}}^{\sigma_L} \tag{14}$$

which can be represented using matrix multiplication as

$$c_{\sigma_1,\cdots,\sigma_L} = A^{\sigma_1} \cdots A^{\sigma_i} S B^{\sigma_{i+1}} \cdots B^{\sigma_L} \tag{15}$$

1.2.2 Similarity to Schmidt Decomposition

When dealing with the Mixed-Canonical MPS, the system is naturally broken into two main blocks - the left, or A block, and the right, or B block. Rewriting the original representation of an arbitrary quantum state, Equation 1, using the Mixed-Canonical represntation gives:

$$\begin{aligned} |\psi\rangle &= \sum_{\sigma_1,\cdots,\sigma_L} A^{\sigma_1} \cdots A^{\sigma_i} S B^{\sigma_{i+1}} \cdots B^{\sigma_L} |\sigma_1 \cdots \sigma_L\rangle \\ &= \sum_{\sigma_1,\cdots,\sigma_L} (A^{\sigma_1} \cdots A^{\sigma_i} |\sigma_1 \cdots \sigma_i\rangle) S (B^{\sigma_{i+1}} \cdots B^{\sigma_L} |\sigma_{i+1} \cdots \sigma_L\rangle) \\ &= S |a_i\rangle_A |a_i\rangle_B \end{aligned} \tag{16}$$

where $|a_i\rangle_A$ and $|a_i\rangle_B$ are the A and B blocks associated with the partitioning at site i. Because of the properties of the SVD, the optimal approximation to $|\psi\rangle$ can be obtained by retaining the first r' singular values

$$|\psi\rangle = \sum_{a_i}^{r'} s_{a_i} |a_i\rangle_A |a_i\rangle_B \tag{17}$$

This approximation is named the Schmidt decomposition and will be shown to be crucial in the implementation of the DMRG algorithm.

1.3 Hamiltonians as MPOs

Similar to how we sought to express the state of the system in terms of localized matrices, it is also advantageous to express operators as matrices that act on a single site. These

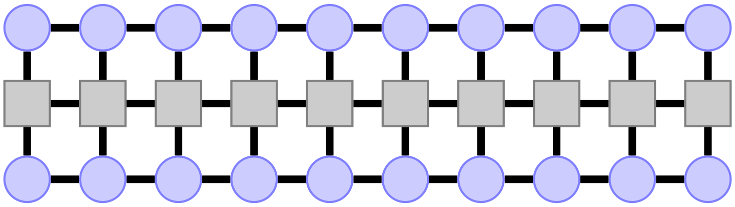

Figure 1: Illustration of the contraction $\langle\psi\big|\hat{\mathcal{O}}\big|\psi\rangle$ where squares represent MPOs and circles represent MPSs.

operators are referred to as Matrix Product Operators (MPOs). With an MPS, a single coefficient can be calculated using matrix multiplication

$$c_{\sigma_1,\cdots,\sigma_L} = \langle\sigma_1\cdots\sigma_L\big|\psi\rangle = \boldsymbol{M}^{\sigma_1}\boldsymbol{M}^{\sigma_2}\cdots\boldsymbol{M}^{\sigma_{L-1}}\boldsymbol{M}^{\sigma_L} \tag{18}$$

Note that here $\boldsymbol{M}^{\sigma_i}$ is used to denote a general MPS representation that is not necessarily Right- or Left-Canonical. We can develop a similar expression for the coefficient of a operator

$$\hat{\boldsymbol{\mathcal{O}}} = \langle\sigma_1\cdots\sigma_L|\hat{\mathcal{O}}\big|\sigma_1'\cdots\sigma_L'\rangle = \boldsymbol{W}^{\sigma_1\sigma_1'}\boldsymbol{W}^{\sigma_2\sigma_2'}\cdots\boldsymbol{W}^{\sigma_{L-1}\sigma_{L-1}'}\boldsymbol{W}^{\sigma_L\sigma_L'} \tag{19}$$

with the only key difference being we now need an ingoing and outgoing physical state. Though outside of the scope of this work, it is possible to show that any operator can be brought into this form. Given a Hamiltonian, it is often a straightforward process to construct the MPOs. As an example, we can show the MPO representation of the Heisenberg Hamiltonian

$$\hat{\mathcal{H}} = -\frac{J}{2}\sum_{i=1}^{N}\left(\hat{S}_i^x\hat{S}_{i+1}^x + \hat{S}_i^y\hat{S}_{i+1}^y + \hat{S}_i^z\hat{S}_{i+1}^z\right) + h\sum_{i=1}^{N}\hat{S}_i^z \tag{20}$$

where $\hat{S}_i^\alpha$ is the associated Pauli spin-1/2 matrix operating on site i. The MPO representation of this is given as

$$\hat{\boldsymbol{W}}^{\sigma_i,\sigma_i'} = \begin{bmatrix} \hat{I} & 0 & 0 & 0 & 0 \\ \hat{S}_i^x & 0 & 0 & 0 & 0 \\ \hat{S}_i^y & 0 & 0 & 0 & 0 \\ \hat{S}_i^z & 0 & 0 & 0 & 0 \\ -h\hat{S}_i^x & -\frac{J}{2}\hat{S}_i^x & -\frac{J}{2}\hat{S}_i^y & -\frac{J}{2}\hat{S}_i^z & \hat{I} \end{bmatrix} \tag{21}$$

with the MPOs on the first and last sites respectively being the bottom row and left-most column of the given matrix. If longer range interactions occur, non-zero terms will appear in the matrix outside the bottom-most row and left-most column. By construction, however, all non-zero elements will be below the diagonal, with the exception of the two identity matrices.

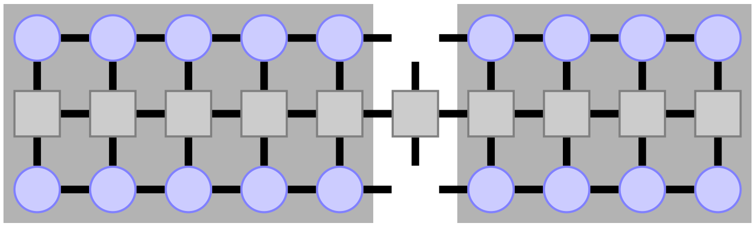

Figure 2: Tensor network diagram representing $\langle a_{i-1}\sigma_i a_i|\hat{\mathcal{H}}\left|a'_{i-1}\sigma'_i a'_i\right\rangle$. The network is naturally broken into the left and right blocks with the MPO of the site of interest separating the two - written respectively as $\boldsymbol{L}$, $\boldsymbol{R}$, and $\boldsymbol{W}$.

It will be shown that another important step of the DMRG algorithm involves applying a Hamiltonian MPO to a Mixed-Canonical MPS, thus we will briefly describe how this is done. Figure 1 shows the tensor network diagram representing this type of operation. The goal of this operation is to calculate the matrix elements associated with the given MPO, which can be shown to be expressed as

$$\begin{aligned}\langle a_{i-1}\sigma_i a_i|\hat{\mathcal{H}}\left|a'_{i-1}\sigma'_i a'_i\right\rangle = \sum_{\{a_i,b_i,a'_i\}} &\left(\sum_{\sigma_1,\sigma'_1}\left(\boldsymbol{A}^{\sigma_1}_{1,a_1}\right)^*\boldsymbol{W}^{\sigma_1,\sigma'_1}_{1,b_1}\boldsymbol{A}^{\sigma'_1}_{1,a_1}\right)\left(\sum_{\sigma_2,\sigma'_2}\left(\boldsymbol{A}^{\sigma_2}_{a_1,a_2}\right)^*\boldsymbol{W}^{\sigma_2,\sigma'_2}_{b_1,b_2}\boldsymbol{A}^{\sigma'_2}_{a_1,a_2}\right)\\ &\cdots\left(\sum_{\sigma_i,\sigma'_i}\left(\boldsymbol{A}^{\sigma_i}_{a_{i-1},a_i}\right)^*\boldsymbol{W}^{\sigma_i,\sigma'_i}_{b_{i-1},b_i}\boldsymbol{A}^{\sigma'_i}_{a_{i-1},a_i}\right)\boldsymbol{W}^{\sigma_i,\sigma'_i}_{b_{i-1},b_i}\\ &\left(\sum_{\sigma_{i+1},\sigma'_{i+1}}\left(\boldsymbol{B}^{\sigma_{i+1}}_{a_i,a_{i+1}}\right)^*\boldsymbol{W}^{\sigma_{i+1},\sigma'_{i+1}}_{b_i,b_{i+1}}\boldsymbol{B}^{\sigma'_{i+1}}_{a_i,a_{i+1}}\right)\\ &\cdots\left(\sum_{\sigma_L,\sigma'_L}\left(\boldsymbol{B}^{\sigma_L}_{a_{L-1},1}\right)^*\boldsymbol{W}^{\sigma_L,\sigma'_L}_{b_{L-1},1}\boldsymbol{B}^{\sigma'_L}_{a'_{L-1},1}\right)\end{aligned} \tag{22}$$

This loses much of the elegance thus far associated with the our MPS/MPO representation of a given state, but using a tensor network diagrammatic representation, shown in Figure 2, we see it simplifies significantly. It also becomes clear that by defining

$$\boldsymbol{L}^{a_{i-1},a'_{i-1}}_{b_{i-1}} = \sum_{\{a_j,b_j,a'_j\}}\left(\sum_{\sigma_1,\sigma'_1}\left(\boldsymbol{A}^{\sigma_1}_{1,a_1}\right)^*\boldsymbol{W}^{\sigma_1,\sigma'_1}_{1,b_1}\boldsymbol{A}^{\sigma'_1}_{1,a'_1}\right)\cdots\left(\sum_{\sigma_i,\sigma'_i}\left(\boldsymbol{A}^{\sigma_i}_{a_{i-1},a_i}\right)^*\boldsymbol{W}^{\sigma_i,\sigma'_i}_{b_{i-1},b_i}\boldsymbol{A}^{\sigma'_i}_{a'_{i-1},a'_i}\right) \tag{23}$$

where $j \in [0, i-1]$, as the left block and

$$\boldsymbol{R}_{b_i}^{a_i,a_i'} = \left(\sum_{\sigma_{i+1},\sigma_{i+1}'} \left(\boldsymbol{B}_{a_i,a_{i+1}}^{\sigma_{i+1}} \right)^* \boldsymbol{W}_{b_i,b_{i+1}}^{\sigma_{i+1},\sigma_{i+1}'} \boldsymbol{B}_{a_i',a_{i+1}'}^{\sigma_{i+1}'} \right) \cdots \left(\sum_{\sigma_L,\sigma_L'} \left(\boldsymbol{B}_{a_{L-1},1}^{\sigma_L} \right)^* \boldsymbol{W}_{b_{L-1},1}^{\sigma_L,\sigma_L'} \boldsymbol{B}_{a_{L-1}',1}^{\sigma_L'} \right) \tag{24}$$

where $j \in [i+1, L]$ as the right block. The equation is thus simplified to

$$\langle a_{i-1} \sigma_i a_i | \hat{\mathcal{H}} \left| a_{i-1}' \sigma_i' a_i' \right\rangle = \sum_{b_{i-1},b_i} \boldsymbol{L}_{b_{i-1}}^{a_{i-1},a_{i-1}'} \boldsymbol{W}_{b_{i-1},b_i}^{\sigma_i,\sigma_i'} \boldsymbol{R}_{b_{i-1}}^{a_{i-1},a_{i-1}'} \tag{25}$$

As the DMRG algorithm is discussed, this expression will be used in each iteration and an emphasis will be placed on the efficient calculation and storage of the left and right blocks.

2 DMRG Ground State Search Algorithm

Now that it has been shown how a quantum state can be represented using the MPS/MPO formalism, we can discuss how the DMRG algorithm is employed to determine a system's ground state. For a given Hamiltonian, $\hat{\mathcal{H}}$, the variational principle dictates that the ground state of the system, $\left|\psi_0\right\rangle$, is the state such that

$$E = \frac{\langle \psi | \hat{\mathcal{H}} | \psi \rangle}{\langle \psi | \psi \rangle} \tag{26}$$

is minimized. Mathematically, this optimization can be done via the introduction of a Lagrange multiplier, λ, and extremization

$$\langle \psi | \hat{\mathcal{H}} | \psi \rangle - \lambda \langle \psi | \psi \rangle \tag{27}$$

such that, following optimization, $\left|\psi\right\rangle$ is the ground state with an associated energy λ. Because the state is represented as an MPS

$$\left|\psi\right\rangle = \sum_{\sigma} M^{\sigma_1} \cdots M^{\sigma_L} \left|\sigma\right\rangle \tag{28}$$

each term in the Lagrangian is highly nonlinear, causing the optimization to become nontrivial.

The DMRG algorithm prescribes an iterative method where the matrix, M^{σ_i}, associated with a single site, i, is optimized while all others are held constant. By systematically varying each matrix, the approximated energy is lowered with each optimization, eventually leading to a converged estimate. The iterative nature of this method simplifies the optimization to straightforward linear algebra, but simultaneously introduces the possibility of convergence to local minima. The remainder of this section provides a description of the optimal DMRG algorithm.

2.1 Initialization

To begin the algorithm, an initial guess for $\left|\psi\right\rangle$ must be specified. Because of the gauge degrees of freedom, there are numerous potential choices. The most useful choice is either a Left- or Right-Canonical representation and for this work, we will choose to begin with a right-normalized MPS. Additionally, we may further simplify our work by simply setting the first element of $\boldsymbol{c}$ equal to one

$$\begin{aligned} \boldsymbol{c}_{\sigma_1\cdots\sigma_L} &= \boldsymbol{\Psi}_{(\sigma_1\cdots\sigma_{L-1}),\sigma_L} \\ &= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{29}$$

then undergoing the right-normalized SVD procedure to arrive at a correctly normalized initial guess.

2.2 Right Sweep

After initializing the Right-Canonical MPS, the iterative optimization sweep is carried out starting at site $i = 1$ and systematically stepping towards $i = L-1$. At each site, the optimization is begun by taking the partial derivative the the Lagrangian with respect to $\boldsymbol{M}^{\sigma_i}_{a_{i-1},a_i}$. From the information in Section 1.3, the Lagrangian can be fully represented as:

$$\begin{aligned} \left\langle\psi\right|\hat{\mathcal{H}}\left|\psi\right\rangle - \lambda\left\langle\psi\middle|\psi\right\rangle = &\sum_{\sigma_i,\sigma_i'}\sum_{a_{i-1}',a_i'}\sum_{a_{i-1},a_i}\sum_{b_{i-1},b_i} \boldsymbol{L}^{a_{i-1},a_{i-1}'}_{b_{i-1}} \boldsymbol{W}^{\sigma_i,\sigma_i'}_{b_{i-1},b_i} \boldsymbol{R}^{a_i,a_i'}_{b_i} \left(\boldsymbol{M}^{\sigma_i}_{a_{i-1},a_i}\right)^* \boldsymbol{M}^{\sigma_i'}_{a_{i-1}',a_i'} \\ &- \lambda \sum_{\sigma_i}\sum_{a_{i-1}',a_i'}\sum_{a_{i-1},a_i} \boldsymbol{\Psi}^A_{a_{i-1},a_{i-1}'} \left(\boldsymbol{M}^{\sigma_i}_{a_{i-1},a_i}\right)^* \boldsymbol{M}^{\sigma_i}_{a_{i-1}',a_i'} \boldsymbol{\Psi}^B_{a_{i-1},a_{i-1}'} \end{aligned} \tag{30}$$

where

$$\boldsymbol{\Psi}^A_{a_{i-1},a_{i-1}'} = \sum_{\sigma_1,\cdots,\sigma_{i-1}} \left(\boldsymbol{M}^{\sigma_{i-1}\dagger}\cdots\boldsymbol{M}^{\sigma_1\dagger}\boldsymbol{M}^{\sigma_1}\cdots\boldsymbol{M}^{\sigma_{i-1}}\right)_{a_{i-1},a_{i-1}'} \tag{31}$$

and

$$\boldsymbol{\Psi}^B_{a_i,a_i'} = \sum_{\sigma_{i+1},\cdots,\sigma_L} \left(\boldsymbol{M}^{\sigma_{i+1}}\cdots\boldsymbol{M}^{\sigma_L}\boldsymbol{M}^{\sigma_L\dagger}\cdots\boldsymbol{M}^{\sigma_{i+1}\dagger}\right)_{a_i',a_i} \tag{32}$$

Because we have assumed that the left block is left-normalized and the right is right-normalized, then the product of $\boldsymbol{\Psi}^A$ and $\boldsymbol{\Psi}^B$ is the identity matrix. Taking the partial derivative then gives:

$$\sum_{\sigma_i'}\sum_{a_{i-1}',a_i'}\sum_{b_{i-1},b_i} \boldsymbol{L}^{a_{i-1},a_{i-1}'}_{b_{i-1}} \boldsymbol{W}^{\sigma_i,\sigma_i'}_{b_{i-1},b_i} \boldsymbol{R}^{a_i,a_i'}_{b_i} \boldsymbol{M}^{\sigma_i'}_{a_{i-1}',a_i'} - \lambda \boldsymbol{M}^{\sigma_i}_{a_{i-1},a_i} = 0 \tag{33}$$

which can be put into the form of a simple eigenvalue problem

$$\boldsymbol{H}\boldsymbol{\nu} = \lambda\boldsymbol{\nu} \tag{34}$$

if we now define

$$\boldsymbol{H}_{(\sigma_i a_{i-1} a_i),(\sigma_i' a_{i-1}' a_i')} = \sum_{b_{i-1},b_i} \boldsymbol{L}^{a_{i-1},a_{i-1}'}_{b_{i-1}} \boldsymbol{W}^{\sigma_i,\sigma_i'}_{b_{i-1},b_i} \boldsymbol{R}^{a_i,a_i'}_{b_i} \tag{35}$$

and

$$\boldsymbol{\nu}_{(\sigma_i a_{i-1} a_i)} = \boldsymbol{M}^{\sigma_i}_{a_{i-1},a_i} \tag{36}$$

These matrices are then fed to an iterative diagonalization routine such as the Lanczos or Jacobi-Davidson algorithms to determine the optimal $\boldsymbol{M}^{\sigma_i}$.

Since we began with a right-normalized MPS and are moving to the right, it is important to left normalized the optimized matrices before stepping forward. To do this, an SVD is performed on $\boldsymbol{M}^{\sigma_i}$ as described in Equation 3

$$\begin{aligned}\boldsymbol{M}^{\sigma_i}_{a_{i-1},a_i} &= \boldsymbol{M}_{(\sigma_i,a_{i-1}),a_i} \\ &= \boldsymbol{U}_{(\sigma_i,a_{i-1}),s_1} \boldsymbol{s}_{s_i} \left(\boldsymbol{V}_{s_k,a_k}\right)^\dagger\end{aligned} \tag{37}$$

The resulting $\boldsymbol{U}$ matrix is reshaped into d left-normalized matrices, $\boldsymbol{A}^{\sigma_i}$:

$$\boldsymbol{U}_{(\sigma_i,a_{i-1}),a_i} = \boldsymbol{A}^{\sigma_i}_{a_{i-1},a_i} \tag{38}$$

while the $\boldsymbol{S}$ and $\boldsymbol{V}$ matrices are multiplied into $\boldsymbol{M}^{\sigma_{i+1}}$

$$\boldsymbol{s}_{s_i} \left(\boldsymbol{V}_{s_i,a_i}\right)^\dagger \boldsymbol{M}^{\sigma_{i+1}}_{a_i,a_{i+1}} = \boldsymbol{M}^{'\sigma_{i+1}}_{s_i,a_{i+1}} \tag{39}$$

The optimization at site i is then complete and we move forward to site $i+1$ and continue until reaching site $L-1$.

2.2.1 Block Calculation and Storage

While Equations 23 and 24 show the simplest representation of the left and right blocks used in Equation 35, it is evident that repeating the numerous summations at each step is inefficient. In this section, improved methodology for the calculation and storage of these blocks is discussed. From Equation 23, it can be seen that the first array at site $i = 1$ is

$$\boldsymbol{L}^{a_1,a_1'}_{b_1} = \sum_{a_0,b_0,a_0',\sigma_1,\sigma_1'} \left(\boldsymbol{A}^{\sigma_1}_{1,a_1}\right)^\dagger \boldsymbol{W}^{\sigma_1,\sigma_1'}_{1,b_1} \boldsymbol{A}^{\sigma_1'}_{1,a_1'} \tag{40}$$

and for $i = 2$, we can rewrite Equation 23 as:

$$\boldsymbol{L}^{a_2,a_2'}_{b_2} = \sum_{a_1,b_1,a_1',\sigma_2,\sigma_2'} \left(\boldsymbol{A}^{\sigma_2}_{a_1,a_2}\right)^\dagger \boldsymbol{W}^{\sigma_2,\sigma_2'}_{b_1,b_2} \boldsymbol{A}^{\sigma_2'}_{a_1',a_2'} \boldsymbol{L}^{a_1,a_1'}_{b_1} \tag{41}$$

Thus we get the general expression for block i to be:

$$\boldsymbol{L}^{a_i,a_i'}_{b_i} = \sum_{a_{i-1},b_{i-1},a_{i-1}',\sigma_i,\sigma_i'} \left(\boldsymbol{A}^{\sigma_i}_{a_{i-1},a_i}\right)^{\dagger} \boldsymbol{W}^{\sigma_i,\sigma_i'}_{b_{i-1},b_i} \boldsymbol{A}^{\sigma_i'}_{a_{i-1}',a_i'} \boldsymbol{L}^{a_{i-1},a_{i-1}'}_{b_{i-1}} \tag{42}$$

which enables us to calculate the left block as we move from site i to $i+1$ using the left block associated with sites $1 \rightarrow i-1$ and the matrices associated with site i. This can be done similarly for the $\boldsymbol{R}$-expressions to give:

$$\begin{aligned} \boldsymbol{R}^{a_{L-1},a_{L-1}'}_{b_{L-1}} &= \sum_{a_{L-1},b_{L-1},a_{L-1}',\sigma_L,\sigma_L'} \left(\boldsymbol{B}^{\sigma_L}_{a_{L-1},1}\right)^{\dagger} \boldsymbol{W}^{\sigma_L,\sigma_L'}_{b_{L-1},1} \boldsymbol{B}^{\sigma_L'}_{a_{L-1}',1} \\ \boldsymbol{R}^{a_{L-2},a_{L-2}'}_{b_{L-2}} &= \sum_{a_{L-2},b_{L-2},a_{L-2}',\sigma_{L-1},\sigma_{L-1}'} \left(\boldsymbol{B}^{\sigma_{L-1}}_{a_{L-2},a_{L-1}}\right)^{\dagger} \boldsymbol{W}^{\sigma_{L-1},\sigma_{L-1}'}_{b_{L-2},b_{L-1}} \boldsymbol{B}^{\sigma_{L-1}'}_{a_{L-2}',a_{L-1}'} \boldsymbol{R}^{a_{L-1},a_{L-1}'}_{b_{L-1}} \\ &\vdots \\ \boldsymbol{R}^{a_i,a_i'}_{b_i} &= \sum_{a_i,b_i,a_i',\sigma_{i+1},\sigma_{i+1}'} \left(\boldsymbol{B}^{\sigma_{i+1}}_{a_i,a_{i+1}}\right)^{\dagger} \boldsymbol{W}^{\sigma_{i+1},\sigma_{i+1}'}_{b_i,b_{i+1}} \boldsymbol{B}^{\sigma_{i+1}'}_{a_i',a_{i+1}'} \boldsymbol{R}^{a_{i+1},a_{i+1}'}_{b_{i+1}} \end{aligned} \tag{43}$$

Thus as part of the initialization, following the calculation of a Right-Canonical MPS, it is useful to calculate and store all $\boldsymbol{R}$-expressions as shown here. Then, as the right sweep propagates, the $\boldsymbol{L}$-expressions are updated and saved after each optimization.

2.3 Left Sweep

The left sweep is carried out in a method conceptually equivalent to what was described in the previous section. It begins at site L and steps iteratively towards site 2. As a summary of the previous section and a concise explanation of its equivalent implementation in the opposite direction, at each step, four basic operations are carried out:

1. Employ an iterative diagonalization method to solve the eigenproblem for $\boldsymbol{M}^{\sigma_i}$.

2. Left-Normalize the resulting matrix $\boldsymbol{M}^{\sigma_i}$ to $\boldsymbol{B}^{\sigma_i}$ by performing an SVD on $\boldsymbol{M}^{\sigma_i}$ and reshaping the resulting $\boldsymbol{V}$ matrix.

3. Multiply the $\boldsymbol{U}$ and $\boldsymbol{S}$ matrices into $\boldsymbol{M}^{\sigma_{i-1}}$ to be used as the starting point for the eigenproblem at the next site to the left.

4. Update the $\boldsymbol{L}$-expression by using the methodology of Section 2.2.1.

2.4 Convergence

The left and right sweeps are continued until the system converges to a steady solution, which is assumed to be the ground state. Because we hope all information contained in the MPS/MPO system is converging, there are multiple potential variables that can be used to test for convergence. The simplest implementation can be achieved by monitoring the change in the estimated energy after each left/right sweep combination. Convergence is achieved as this value approaches zero. Alternatively, a more robust convergence criteria is to determine whether the system has arrived at an eigenstate. As the system approaches an eigenstate, we will see

$$\langle\psi|\hat{\mathcal{H}}^2|\psi\rangle - \left(\langle\psi|\hat{\mathcal{H}}|\psi\rangle\right)^2 \to 0 \tag{44}$$

Because in the current implementation, we use the energy to evaluate convergence, it is informative to briefly write the equation used to calculate the energy

$$E = \frac{\langle\psi|\hat{\mathcal{H}}|\psi\rangle}{\langle\psi|\psi\rangle} \tag{45}$$

where

$$\langle\psi|\hat{\mathcal{H}}|\psi\rangle = \sum_{\sigma_i,\sigma_i'}\sum_{a_{i-1}',a_i'}\sum_{a_{i-1},a_i}\sum_{b_{i-1},b_i} \boldsymbol{L}_{b_{i-1}}^{a_{i-1},a_{i-1}'} \boldsymbol{W}_{b_{i-1},b_i}^{\sigma_i,\sigma_i'} \boldsymbol{R}_{b_i}^{a_i,a_i'} \left(\boldsymbol{M}_{a_{i-1},a_i}^{\sigma_i}\right)^* \boldsymbol{M}_{a_{i-1}',a_i'}^{\sigma_i'} \tag{46}$$

and

$$\langle\psi|\psi\rangle = \sum_{\sigma_i}\sum_{a_{i-1}',a_i'}\sum_{a_{i-1},a_i} \boldsymbol{\Psi}^{A}_{a_{i-1},a_{i-1}'} \left(\boldsymbol{M}_{a_{i-1},a_i}^{\sigma_i}\right)^* \boldsymbol{M}_{a_{i-1}',a_i'}^{\sigma_i} \boldsymbol{\Psi}^{B}_{a_{i-1},a_{i-1}'} \tag{47}$$

where $\boldsymbol{\Psi}^{A}_{a_{i-1},a_{i-1}'}$ and $\boldsymbol{\Psi}^{B}_{a_{i-1},a_{i-1}'}$ are described in equations 31 and 32