Matrix Product States
Notebook

David Aceituno

Contents

HOW-TO

- (**Recommended**) Open on StackEdit, an online open-source markdown+tex editor.
- Install a MathJax renderer for your browser to read directly on the GitHub webpage. For instance, Github with MathJax.
- Open in Typora editor.
- Open this file on your pc/mac with the Atom editor with the `markdown-preview-enhanced` -plugin installed. (Optional: Select Katex renderer in settings).
- Open this file on your pc/mac with the ReText editor. To get inline equations showing nicely, do the following (copy-paste to terminal):
  - `sudo apt install python3-pyqt5.qtwebkit`
  - `echo "mathjax" >> ~/.config/markdown-extensions.txt`.
  - Finally enable webkit inside ReText: `Edit -> Use WebKit Renderer`.

Product states

Matrix Product States

There are two alternative ways to prepare MPS. The first is by using successive Schmidt decompositions, which as it turns out also produces MPS in *canonical form*. The second is a preparation from maximally entangled pairs, the so called valence bonds states.

Canonical form (from Jens paper)

A general quantum state $|\psi\rangle$ on a chain with N sites can be written in MPS form:

$$|\psi\rangle = \sum_{\sigma_1 \dots \sigma_N} A^{\sigma_1} \dots A^{\sigma_N} |\sigma_1 \dots \sigma_N\rangle, \tag{1}$$

where A^{σ_i} is a $r_{i-1} \times r_i$ matrix, r_i being the rank of the Schmidt decomposition at site i. Note that at the boundary we have $r_0 = r_N = 1$, which means that A^{σ_1} and A^N are vectors, and therefore the matrix product returns a scalar coefficient.

We can rewrite the matrices A^{σ_i} as a product of $r_{i-1} \times r_i$ complex matrices Γ^{σ_i} and positive, real, square diagonal matrices Λ^i

$$|\psi\rangle = \sum_{\sigma_1 \dots \sigma_N} \Gamma^{\sigma_1} \Lambda^1 \Gamma^{\sigma_2} \Lambda^2 \dots \Lambda^{N-1} \Gamma^{\sigma_N} |\sigma_1 \dots \sigma_N\rangle, \tag{2}$$

which diagrammatically looks like

This form allows for many possible representations of the same wave function, giving us the opportunity to define a "canonical form" of the MPS.

We can define a set of χ_n wave functions $|\alpha\rangle^n_{L/R}$ to the left/right of a bond, such that

$$|\psi\rangle = \sum_{\alpha=1}^{\chi} \Lambda^n_\alpha |\alpha\rangle^n_L \otimes |\alpha\rangle^n_R, \quad |\alpha\rangle \in H_{L/R} \tag{3}$$

$$\tag{4}$$

The MPS representation $\{\Gamma^{\sigma_1} \Lambda^1 \Gamma^{\sigma_2} \Lambda^2 \dots \Lambda^{N-1} \Gamma^{\sigma_N}\}$ is canonical if for every bond, the set of wave functions $|\alpha\rangle^n_{L/R}$ along with Λ^i form a Schmidt decomposition of ψ. In other words we must have $\langle\hat{\alpha}|\alpha\rangle^n_L = \delta_{\hat{\alpha}\alpha}$, $\langle\hat{\alpha}|\alpha\rangle^n_R = \delta_{\hat{\alpha}\alpha}$ and $\sum {\Lambda^i_\alpha}^2 = 1$ on every bond. Equivalently, $A^{\sigma_i\dagger} A^{\sigma_i} = \mathbf{1}$.

Schmidt decomposition according to Vidal

Summary from the following papers

Vidal, G. (2003). Efficient Classical Simulation of Slightly Entangled Quantum Computations. Physical Review Letters, 91(14), 147902.

Vidal, G. (2004). Efficient simulation of one-dimensional quantum many-body systems. Physical Review Letters, 93(4), 40502–1.

The local decomposition of the state $|\psi\rangle \in H_2^{\otimes n}$ in terms of n tensors $\{\Gamma^{\sigma_l}\}_{l=1}^{n}$ and $\{\lambda^l\}_{l=1}^{n-1}$ is denoted

$$|\psi\rangle \leftrightarrow \Gamma^{\sigma_1}\lambda^1\Gamma^{\sigma_2}\lambda^2...\lambda^{n-1}\Gamma^{\sigma_n} \tag{5}$$

Here, tensor Γ^{σ_l} has at most three indices $\Gamma^{\sigma_1}_{\alpha\alpha'}$, where $\alpha, \alpha' = 0, ..., \chi$ and $\sigma_l = 0, 1$, whereas λ^l is a vector whose components $\lambda^l_{\alpha'}$ store the Schmidt coefficients of the splitting $[1...l] : [(l+1)...n]$. More explicitly we have

$$c_{\sigma_1...\sigma_n} = \sum_{\alpha_1...\alpha_{n-1}} \Gamma^{\sigma_1}_{\alpha_1}\lambda^1_{\alpha_1}\Gamma^{\sigma_2}_{\alpha_1\alpha_2}\lambda^1_{\alpha_2}...\Gamma^{\sigma_n}_{\alpha_{n-1}} \tag{6}$$

so that 2^n coefficients in $c_{\sigma_1...\sigma_n}$ are expressed in terms of about $(2\chi^2+\chi)n$ parameters, a number that grows linearly in n for a fixed value of χ.

Procedure

This is essentially a concatenation of $n-1$ Schmidt decompositions, and depends on how the qubits have been ordered from 1 to n. We first compute the Schmidt decomposition according to the bipartite splitting of $|\psi\rangle$ into qubit 1 and the $n-1$ remaining qubits.

$$|\psi\rangle = \sum_{\alpha_1} \lambda^1_{\alpha_1}|\Phi^1_{\alpha_1}\rangle|\Phi^{2...n}_{\alpha_1}\rangle = \sum_{\sigma_1,\alpha_1} \Gamma^{\sigma_1}_{\alpha_1}\lambda^1_{\alpha_1}|\sigma_1\rangle|\Phi^{2...n}_{\alpha_1}\rangle \tag{7}$$

We then proceed according to the following three steps:

1. Expand each vector $|\Phi^{2...n}_{\alpha_1}\rangle$ in a local basis for qubit 2, $|\Phi^{2...n}_{\alpha_1}\rangle = \sum_{\sigma_2}|\sigma_2\rangle|\tau^{3...n}_{\alpha_1,\sigma_2}\rangle$
2. Write each vector $|\tau^{3...n}_{\alpha_1,\sigma_2}\rangle$ in terms of at most χ Schmidt vectors $\{|\Phi^{3...n}_{\alpha_2}\rangle\}^{\chi}_{\alpha_2}$, i.e. eigenvectors of $\rho^{3...n}$ and the corresponding Schmidt coefficients $\lambda^2_{\alpha_2}$: $|\tau^{3...n}_{\alpha_1,\sigma_2}\rangle = \sum_{\alpha_2}\Gamma^{\sigma_2}_{\alpha_1\alpha_2}\lambda^2_{\alpha_2}|\Phi^{3...n}_{\alpha_2}\rangle$
3. Substitute the equations in (1) and (2) into the first splitting above, i.e.:

$$|\Phi^{2...n}_{\alpha_1}\rangle = \sum_{\sigma_2}|\sigma_2\rangle\sum_{\alpha_2}\Gamma^{\sigma_2}_{\alpha_1\alpha_2}\lambda^2_{\alpha_2}|\Phi^{3...n}_{\alpha_2}\rangle \tag{8}$$

$$= \sum_{\sigma_2,\alpha_2}\Gamma^{\sigma_2}_{\alpha_1\alpha_2}\lambda^2_{\alpha_2}|\sigma_2\rangle|\Phi^{3...n}_{\alpha_2}\rangle \tag{9}$$

followed by

$$|\psi\rangle = \sum_{\sigma_1,\alpha_1}\Gamma^{\sigma_1}_{\alpha_1}\lambda^1_{\alpha_1}|\sigma_1\rangle\sum_{\sigma_2}|\sigma_2\rangle\sum_{\alpha_2}\Gamma^{\sigma_2}_{\alpha_1\alpha_2}\lambda^2_{\alpha_2}|\Phi^{3...n}_{\alpha_2}\rangle \tag{10}$$

$$= \sum_{\sigma_1\sigma_2,\alpha_1\alpha_2}\Gamma^{\sigma_1}_{\alpha_1}\lambda^1_{\alpha_1}\Gamma^{\sigma_2}_{\alpha_1\alpha_2}\lambda^2_{\alpha_2}|\sigma_1\rangle|\sigma_2\rangle|\Phi^{3...n}_{\alpha_2}\rangle \tag{11}$$

Repeating steps 1 to 3 for the Schmidt vectors $|\Phi^{4...n}_{\alpha_3}\rangle, |\Phi^{5...n}_{\alpha_4}\rangle...$ gives us the state $|\psi\rangle$ in terms of tensors Γ^{σ_l} and λ^l.

Remarks on implementation in code

Basically, to begin a decomposition we do a mode-1 unfolding to write the tensor in matrix form $c_{\sigma_1\dots\sigma_N} \to c_{\sigma_1,\sigma_2\dots\sigma_N}$. See below for details on unfolding. The resulting matrix undergoes an SVD decomposition to get $USV^\dagger$, where we then set $US = \Gamma\lambda$.

Steps 1. and 2. above then amount to taking $V^\dagger$ and **doing the mode-1 unfolding backwards**, giving something like $V^\dagger_{\sigma_1\dots\sigma_N}$, and then refolding into $V^\dagger_{\sigma_1\sigma_2,\sigma_3\dots\sigma_N}$.

The reason for taking the extra step back and forth is to avoid having to keep track on which reshaping algorithm is being used; any reshaping order will do!

Tensor decomposition (matrix unfolding)

There are more details on higher-order tensor decomposition (matrix unfolding) here: De Lathauwer, L., De Moor, B., & Vandewalle, J. (2000). A Multilinear Singular Value Decomposition. - Society for Industrial and Applied Mathematics. Journal on Matrix Analysis and Applications

And also here:

Bengua, J. a., Phien, H. N., Tuan, H. D., & Do, M. N. (2015). Matrix Product State for Feature Extraction of Higher-Order Tensors, (1944), 10.

NOTE: The method above works best for decompositions from the right. We want neighboring qubits to pick the "largest" subsections of the flattened matrix! To go from the left, define instead $J_k = \prod_{m=1,m\neq n}^{N-k} I_m$. Notice the product limits! Let's call this the "Left-method", for decompositions from the left, and the other one the "Right-method". But this doesn't really matter much if you undo flattening between each step.

For qubits this is simply $j = 1 + \sum_{k=1,k\neq n} \sigma_k 2^{k-1}$, which allows for a representation in binary numbers!

For a general "bipartite" flattening of a tensor down to a matrix, $c_{\sigma_1\dots\sigma_N} \to c_{[\sigma_1\dots\sigma_n],[\sigma_{n+1}\dots\sigma_N]} = c_{ij}$, we can find the indices as

$$i = \sum_{k=1}^{n} \sigma_k J_k \text{ with } J_k = \prod_{m=1}^{n-k} I_m \tag{12}$$

$$j = \sum_{k=n+1}^{N} \sigma_k J_k \text{ with } J_k = \prod_{m=n+1}^{N-k} I_m \tag{13}$$

which for qubits simplifies to

$$i = 1 + \sum_{k=1}^{n} \sigma_k 2^{k-1} \tag{14}$$

$$j = 1 + \sum_{k=n+1}^{N} \sigma_k 2^{k-n-1} \tag{15}$$

The inverse process, i.e. going $c_{ij} \to c_{\sigma_1\dots\sigma_N}$, is very simple for qubits: Just write i and j in binary form!

Otherwise, for general local dimension I_k we have

$$i = \sigma_n + \sigma_{n-1} I_n + \sigma_{n-2} I_n I_{n-1} + ... + \sigma_1 I_n I_{n-1} I_{n-2} ... I_2 \tag{16}$$

$$j = \sigma_N + \sigma_{N-1} I_N + \sigma_{N-2} I_N I_{N-1} + ... + \sigma_{n+1} I_N I_{N-1} I_{N-2} ... I_{n+2} \tag{17}$$

```
//Let i and j be indices to tensor element c_ij. N is the total number of spins, and n  the last spin i
int C = j-1;
sigma[N] = mod(C,dim[N])
for(k = N; k > 0; k--){
    if(k == n){
        C = (i-1);
    }
    C = (C-sigma[k+1])/dim[k+1];
    sigma[k] = mod(C,dim[k]);
}
```

Local updates

Updating the state $|\psi\rangle$ after a unitary operation U acts on qubit l does only involve transforming Γ^{σ_l}. The computational cost is $\mathcal{O}(\chi^2)$ basic operations.

When a unitary operation V, like a two-qubit gate, is applied to qubits l and $l+1$ only $\Gamma^{\sigma_l}, \lambda^l$ and $\Gamma^{\sigma_{l+1}}$ need to be updated. The computational cost is $\mathcal{O}(\chi^3)$ basic operations.

Orthogonality Center

From page 5 and 6 of Wall, M. L., & Carr, L. D. (2012). Out of equilibrium dynamics with Matrix Product States, *125015*, 35.

In general an MPS has *gauge freedom*, meaning that the tensors A are not uniquely defined. For Open Boundary Conditions (OBS) on can specify the state uniquely (up to possible degeneracies in the Schmidt decomposition) by choosing a site k, called the *orthogonality center* of the MPS, and requiring that all sites i to the left and right of k, satisfy the left

$$\sum_i A^{i\dagger} A^i = I \tag{18}$$

and right

$$\sum_i A^i A^{i\dagger} = I \tag{19}$$

gauge conditions, respectively.

Example 1: Four Qubits in the GHZ-state

This example follows the steps in the book (p. 156) Quantum Many-Body Physics of Ultracold Molecules in Optical Lattices - Models and Simulation Methods

---

The Greenberger-Horne-Zeilinger (GHZ) state is defined as

$$|\mathrm{GHZ}\rangle = |00...0\rangle + |11...1\rangle \tag{20}$$

This represents a realization of Schrödinger's cat paradox, in which a quantum system exists in two very different macroscopic states simultaneously. With 4 qubits, we should get the following MPS in normalized left- or right-canonical form:

$$A^{[j]i} = \begin{pmatrix} \delta_{i,0}/\sqrt{2} & 0 \\ 0 & \delta_{i,1}/\sqrt{2} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}_{i=0} \text{ or } \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}_{i=1} \tag{21}$$

where j is the location on the chain, and i is the value of the qubit. In the Vidal canonical form (VCF), the normalized GHZ state takes the form

$$\lambda^{[j]} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \tag{22}$$

$$\Gamma^{[j]i} = \begin{pmatrix} \delta_{i,0}/\sqrt{2} & 0 \\ 0 & \delta_{i,1}/\sqrt{2} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}_{i=0} \text{ or } = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}_{i=1} \tag{23}$$

for $2 \leq j \leq L-1$ together with the boundaries

$$\lambda^{[1]} = \lambda^{[L+1]} = 1 \tag{24}$$

$$\Gamma^{[1]i} = (\delta_{i,0}\delta_{i,1}) = (1,0)_{i=0} \text{ or } (0,1)_{i=1} \tag{25}$$

$$\Gamma^{[L]i} = \begin{pmatrix} \delta_{i,0} \\ \delta_{i,1} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}_{i=0} \text{ or } \begin{pmatrix} 0 \\ 1 \end{pmatrix}_{i=1} \tag{26}$$

Example 1 (Not thorough): Three Qubits, following Schollwoeck

---

This example tries to follow the steps in Schollwoeck, U. (2010). The density-matrix renormalization group in the age of matrix product states. Annals of Physics, 326(1), 96192.

Step 1:

In general, a 3 qubit state can be written as:

$$|\psi\rangle = \sum_{\sigma_1\sigma_2\sigma_3} c_{\sigma_1\sigma_2\sigma_3}|\sigma_1\sigma_2\sigma_3\rangle, \tag{27}$$

where each $\sigma_i \in \{0,1\}$ and the coefficients $c_{\sigma_1\sigma_2\sigma_3}$ are 2^3 complex numbers. These numbers can be visualized as being on the corners of a (hyper)cube, or simply a long list of numbers corresponding to the 8 possible states.

Consider the following state with 3 qubits: $|\psi\rangle = \frac{1}{\sqrt{2}}(|010\rangle + |101\rangle)$. The cube would look as in the figure below.

or simply,

$$\begin{array}{c|l} 0 & |000\rangle \\ 0 & |001\rangle \\ 2^{-1/2} & |010\rangle \\ 0 & |011\rangle \\ 0 & |100\rangle \\ 2^{-1/2} & |101\rangle \\ 0 & |110\rangle \\ 0 & |111\rangle \end{array} \tag{28}$$

The first step in the decomposition is to define a $d \times d^{L-1} = 2 \times 2^2$ matrix that flattens the tensor:

$$\Psi_{\sigma_1,(\sigma_2\sigma_3)} = \begin{array}{r|cccc} & \boldsymbol{\sigma_1 = 0, \sigma_3 = 0} & \boldsymbol{\sigma_1 = 0, \sigma_3 = 1} & \boldsymbol{\sigma_1 = 1, \sigma_3 = 0} & \boldsymbol{\sigma_1 = 1, \sigma_3 = 1} \\ \boldsymbol{\sigma_2 = 0} & 0 & 0 & 0 & 2^{-1/2} \\ \boldsymbol{\sigma_2 = 1} & 2^{-1/2} & 0 & 0 & 0 \end{array} = \begin{pmatrix} 0 & 0 & 0 & 2^{-1/2} \\ 2^{-1/2} & 0 & 0 & 0 \end{pmatrix}$$

Note how the cube has been sliced and concatenated. Basically, the matrix is composed of two (2×2) matrices side by side, one for each value of σ_1.

Now we perform the single value decomposition on $\Psi_{\sigma_1,(\sigma_2\sigma_3)} = USV^\dagger$:

$$\begin{aligned} \begin{pmatrix} 0 & 0 & 0 & 2^{-1/2} \\ 2^{-1/2} & 0 & 0 & 0 \end{pmatrix} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 2^{-1/2} & 0 \\ 0 & 2^{-1/2} \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix} \\ &= \sum_{a_1}^{r_1} U_{\sigma_1,a_1} S_{a_1,a_1} V^\dagger_{a_1,\sigma_2\sigma_3} \\ &\equiv \sum_{a_1}^{r_1} U_{\sigma_1,a_1} c_{a_1,\sigma_2\sigma_3} \end{aligned}$$

where $r_1 \leq d = 2$ is the rank of the decomposition, i.e., the number of nonzero items in S, and $a_1 \in \{0,1\}$. In the last equality S and $V^\dagger$ have been multiplied. It can then be reshaped into a matrix of dimension $(r_1 d \times d) = (4 \times 2)$, called $\Psi_{(a_1\sigma_2),(\sigma_3)}$. This is NOT done by stacking the (2×2) matrices. Instead, note how the σ_2 index selects the upper/lower row, which then become matrices. Pythons numpy.reshape() does this. **Here the label $a_1$is the index being summed over (by matrix multiplication), and σ_2, σ_3 serve to select appropriate matrices.**

$$c_{a_1,\sigma_2\sigma_3} = \begin{pmatrix} 0 & 0 & 0 & 2^{-1/2} \\ 2^{-1/2} & 0 & 0 & 0 \end{pmatrix} \to \begin{pmatrix} \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 2^{-1/2} \end{pmatrix}\right)_{\sigma_3} \\ \left(\begin{pmatrix} 2^{-1/2} \\ 0 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 0 \end{pmatrix}\right)_{\sigma_3} \end{pmatrix}_{\sigma_2} = \Psi_{(a_1\sigma_2),(\sigma_3)}$$

U is now sliced into $d = 2$ row vectors A^{σ_1}, which we interpret as (1×2) matrices, i.e., $A^{\sigma_1}_{a_1} = U_{\sigma_1,a_1} \to \begin{pmatrix}\begin{pmatrix}1 & 0\end{pmatrix}\\ \begin{pmatrix}0 & 1\end{pmatrix}\end{pmatrix}_{\sigma_1}$, where σ_1 labels each row vector.

By now we have achieved the following:

$$c_{\sigma_1\sigma_2\sigma_3} = \sum_{a_1}^{r_1} A^{\sigma_1}_{a_1}\Psi_{(a_1\sigma_2),(\sigma 3)} = \begin{pmatrix}\begin{pmatrix}1 & 0\end{pmatrix}\\ \begin{pmatrix}0 & 1\end{pmatrix}\end{pmatrix}_{\sigma_1} \begin{pmatrix}\left(\binom{0}{0} \quad \binom{0}{2^{-1/2}}\right)_{\sigma_3}\\ \left(\binom{2^{-1/2}}{0} \quad \binom{0}{0}\right)_{\sigma_3}\end{pmatrix}_{\sigma_2}$$

where the labels $\sigma_1, \sigma_2, \sigma_3$ serve to index the inner elements.

So for instance if $|\sigma_1\sigma_2\sigma_3\rangle = |101\rangle$ we get $c_{101} = (0,1)\binom{0}{2^{-1/2}} = 2^{-1/2}$ as expected (check this!).

Step 2:

Next, we apply the SVD decomposition once more

$$\begin{aligned}\Psi_{(a_1\sigma_2),(\sigma_3)} = USV^\dagger &= \begin{pmatrix}0 & 0\\ 0 & 1\\ -1 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}2^{-1/2} & 0\\ 0 & 2^{-1/2}\end{pmatrix}\begin{pmatrix}-1 & 0\\ 0 & 1\end{pmatrix}\\ &= \sum_{a_2}^{r_2} U_{(a_1\sigma_2),a_2} S_{a_2,a_2}(V^\dagger)_{a_2,(\sigma_3)}\\ &= \sum_{a_2}^{r_2} A^{\sigma_2}_{a_1,a_2}\Psi_{a_2\sigma_3}\end{aligned}$$

where U is replaced by a set of d matrices A^{σ_2} of dimension $r_1 \times r_2 = (2 \times 2)$ with entries $A^{\sigma_2}_{a_1,a_2} = U_{(a_1\sigma_2),a_2}$. As before, $SV^\dagger$ has been reshaped into a matrix Ψ of dimension $r_2 d \times d^{L-3} = (4 \times 1)$.

Explicitly, this reads:

$$\sum_{a_2}^{r_2} A^{\sigma_2}_{a_1,a_2}\Psi_{a_2\sigma_3} = \begin{pmatrix}\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}\\ \begin{pmatrix}-1 & 0\\ 0 & 0\end{pmatrix}\end{pmatrix}_{\sigma_2} \begin{pmatrix}\binom{-2^{-1/2}}{0}\\ \binom{0}{2^{-1/2}}\end{pmatrix}_{\sigma_3}$$

So far we have achieved the following:

$$c_{\sigma_1\sigma_2\sigma_3} = \sum_{a_1,a_2}^{r_1,r_2} A^{\sigma_1}_{a_1}A^{\sigma_2}_{a_1,a_2}\Psi_{a_2\sigma_3}$$

So for instance $c_{101} = A^{\sigma_1=1}A^{\sigma_2=0}\Psi_{a_2\sigma_3=1} = (0,1)\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}\binom{0}{2^{-1/2}} = 2^{-1/2}$, as expected.

Step 3: We perform the SVD decomposition for the last time:

$$\Psi_{(a_1\sigma_2),(\sigma_3)} = USV^\dagger = \begin{pmatrix}-2^{-1/2}\\ 0\\ 0\\ 2^{-1/2}\end{pmatrix} \times 1 \times 1$$

where as before we split U into a collection of d vectors with elements $A^{\sigma_3}_{a_2} = U_{(a_2\sigma_3)}$.

Following the previous prescription, we set $A^{\sigma_3} = \begin{pmatrix} \left(\begin{smallmatrix} -2^{-1/2} \\ 0 \end{smallmatrix}\right) \\ \left(\begin{smallmatrix} 0 \\ 2^{-1/2} \end{smallmatrix}\right) \end{pmatrix}_{\sigma_3}$, and we are done.

We now have

$$|\psi\rangle = \sum_{\sigma_1\sigma_2\sigma_3} c_{\sigma_1\sigma_2\sigma_3}|\sigma_1\sigma_2\sigma_3\rangle = \sum_{a_1,a_2,a_3}^{r_1,r_2,r_3} A^{\sigma_1}_{a_1} A^{\sigma_2}_{a_1,a_2} A^{\sigma_3}_{a_3}|\sigma_1\sigma_2\sigma_3\rangle$$

where

$$\begin{aligned} A^{\sigma_1}_{\alpha_1} &= \{(1,0),(0,1)\} \\ A^{\sigma_2}_{\alpha_1\alpha_2} &= \{\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix}\} \\ A^{\sigma_3}_{\alpha_2} &= \{\begin{pmatrix} -2^{-1/2} \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2^{-1/2} \end{pmatrix}\} \end{aligned}$$

Remark on Normalization It would seem that it is simpler to do the decomposition for unnormalized states using ones everywhere, and then normalize by $1/\sqrt{2}$ perhaps?

Example 2: Four Qubits, following Vidal

Let $|\psi\rangle = \frac{1}{\sqrt{3}}(|1110\rangle + |0011\rangle + |1010\rangle)$

We do a bipartite splitting on the first qubit,

$$|\psi\rangle = \sum_{\alpha_1} \lambda^1_{\alpha_1} |\Phi^1_{\alpha_1}\rangle |\Phi^{\sigma_2\sigma_3\sigma_4}_{\alpha_1}\rangle = \sum_{\sigma_1,\alpha_1} \Gamma^{\sigma_1}_{\alpha_1} \lambda^1_{\alpha_1} |\sigma_1\rangle |\Phi^{\sigma_2\sigma_3\sigma_4}_{\alpha_1}\rangle$$

where $\Gamma^{\sigma_1}_{\alpha_1}$ comes from the SVD decomposition and $\lambda^1_{\alpha_1}$ are the corresponding singular values, or Schmidt coefficients.

Preliminaries: Mode-1 unfolding (Left method)

To actually perform the SVD decomposition initially, the tensor $c_{\sigma_1\ldots\sigma_4}$ needs to be flattened, or matricized.

We begin by doing a *mode-1* matrix unfolding (or flattening), where we get a matrix $\Psi_{i_1,j} = \Psi_{\sigma_1,(\sigma_2\ldots\sigma_4)} \in \mathbb{R}^{2\times(2\cdot2\cdot2)}$. Note that i_k and j denotes the tensor indices, i.e. $i_k \in 1,2,\ldots,I_k$, and $j \in 1,2,\ldots,(\prod_{m\neq k}^{n} I_m)$. An I_k is simply the dimensions of a qubit at k, i.e. 2.

The tensor $c_{\sigma_1\ldots\sigma_4}$ has nonzero elements c_{1100}, c_{0011} and c_{1010} which, in a mode-1 unfolding, maps to matrix coordinates

Using the "Left-method" (see earlier in this section), the tensor $c_{\sigma_1\ldots\sigma_4}$ maps to matrix coordinates

- $c_{1110} \to \Psi_{i_1=2,j=7}$. Here $j = 1 + \sum_{k\neq1}^{4}(i_k - 1)J_k$ with $J_k = \prod_{m\neq1}^{N-k} I_m$. We can use qubit values directly, by instead rewriting $j = 1 + \sum_{k\neq1}^{4} \sigma_k J_k$. This gives $j = 1 + \sigma_2 J_2 + \sigma_3 J_3 + \sigma_4 J_4 = 1 + 1*4 + 1*2 + 0*1 = 7$.
- $c_{0011} \to \Psi_{i_1=1,j=4}$ Using the same method as above we get $j = 1 + 0*4 + 1*2 + 1*1 = 4$.
- $c_{1010} \to \Psi_{i_1=2,j=3}$. And again, $j = 1 + 0*4 + 1*2 + 0*1 = 3$.

Reinserting the normalization factor we get

$$\Psi_{\sigma_1,(\sigma_2\sigma_3\sigma_4)} = \frac{1}{\sqrt{3}} \begin{pmatrix} 0 & 0 & 0 & 1_{0011} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1_{1010} & 0 & 0 & 0 & 1_{1110} & 0 \end{pmatrix}$$

Using the "Right-method" (see earlier in this section), the tensor $c_{\sigma_1\ldots\sigma_4}$ maps to matrix coordinates

- $c_{1110} \to \Psi_{i_1=2,j=5}$. Here $j = 1+\sum_{k\neq i_1}^{4}(i_k-1)J_k$ with $J_k = \prod_{m\neq i_1}^{k-1} I_m$. We can use qubit values directly, by instead rewriting $j = 1+\sum_{k\neq i_1}^{k-1}\sigma_k J_k$. This gives $j = 1+\sigma_2 J_2+\sigma_3 J_3+\sigma_4 J_4 = 1+1*2+1*2+0*4 = 5$.
- $c_{0011} \to \Psi_{i_1=1,j=7}$ Using the same method as above we get $j = 1+0*2+1*2+1*4 = 7$.
- $c_{1010} \to \Psi_{i_1=2,j=3}$. And again, $j = 1+0*2+1*2+0*4 = 3$.

Reinserting the normalization factor we get

$$\Psi_{\sigma_1,(\sigma_2\sigma_3\sigma_4)} = \frac{1}{\sqrt{3}}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1_{0011} & 0 \\ 0 & 0 & 1_{1010} & 0 & 1_{1110} & 0 & 0 & 0 \end{pmatrix}$$

which goes into the first SVD iteration.

First SVD

The SVD decomposition of $\Psi_{\sigma_1,(\sigma_2\sigma_3\sigma_4)}$ yields:

$$U = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$
$$S = \frac{1}{\sqrt{3}}\begin{pmatrix} \sqrt{2} & 0 \\ 0 & 1 \end{pmatrix}$$
$$V^\dagger = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \sqrt{2} & 0 & 0 & 0 & 0 \end{pmatrix}$$

where we can identify $\Gamma^{\sigma_1}_{\alpha_1} = U$ (here σ_1 labels the row-vectors in U), and $\lambda^1_{\alpha_1} = \{\sqrt{\frac{2}{3}}, \frac{1}{\sqrt{3}}\}$. We arrive at

$$|\psi\rangle = \sum_{\alpha_1}\lambda^1_{\alpha_1}|\Phi^1_{\alpha_1}\rangle|\Phi^{\sigma_2\sigma_3\sigma_4}_{\alpha_1}\rangle = \sum_{\sigma_1,\alpha_1}\Gamma^{\sigma_1}_{\alpha_1}\lambda^1_{\alpha_1}|\sigma_1\rangle|\Phi^{\sigma_2\sigma_3\sigma_4}_{\alpha_1}\rangle,$$

and we identify $|\Phi^{\sigma_2\sigma_3\sigma_4}_{\alpha_1}\rangle$ as $V^\dagger$.

Note that the SVD decomposition of $\Psi_{\sigma_1,(\sigma_2\sigma_3\sigma_4)}$ with the “right method” yields:

$$U = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$
$$S = \frac{1}{\sqrt{3}}\begin{pmatrix} \sqrt{2} & 0 \\ 0 & 1 \end{pmatrix}$$
$$V^\dagger = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sqrt{2} & 0 \end{pmatrix}$$

Now we apply step 1. in Vidal's paper: *“expand each Schmidt vector $|\Phi^{\sigma_2\ldots\sigma_3}_{\alpha_1}\rangle$ in a local basis for qubit 2”: $|\Phi^{\sigma_2\ldots\sigma_3}_{\alpha_1}\rangle = \sum_{\sigma_2}|\sigma_2\rangle|\tau^{\sigma_3\sigma_4}_{\alpha_1,\sigma_2}\rangle$“.*

This simply means to take $V^\dagger$ back to tensor form $V^\dagger_{\sigma_1\ldots\sigma_4}$ and then flatten it down to a matrix $V^\dagger_{[\sigma_1\sigma_2],[\sigma_3\sigma_4]}$. For nonzero elements this means:

- $V^\dagger_{1,3} \to V^\dagger$

To continue, we slice $|\Phi^{\sigma_2\sigma_3\sigma_4}_{\alpha_1}\rangle$ for each possible value of σ_2, i.e., two (2×4) matrices, with α_1 labeling the rows. In Vidal's notation

$$|\tau^{\sigma_3\sigma_4}_{\alpha_1,\sigma_2=0}\rangle = \frac{1}{\sqrt{3}}\begin{pmatrix} 0 & 0 & 1_{1010} & 0 \\ 0 & 0 & 0 & 1_{0011} \end{pmatrix}$$
$$|\tau^{\sigma_3\sigma_4}_{\alpha_1,\sigma_2=1}\rangle = \frac{1}{\sqrt{3}}\begin{pmatrix} 0 & 0 & 1_{1110} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

We finish this step by stacking these matrices into a (4 × 4) matrix

$$|\tau^{\sigma_3\sigma_4}_{\alpha_1,\sigma_2}\rangle = \frac{1}{\sqrt{3}}\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

which goes into the next step.

Second SVD

The SVD decomposition of $|\tau^{\sigma_3\sigma_4}_{\alpha_1,\sigma_2}\rangle$ yields:

$$U = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & \sqrt{2} & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & \sqrt{2} & 0 \end{pmatrix}$$

$$S = \frac{1}{\sqrt{3}}\begin{pmatrix} \sqrt{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$V^\dagger = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Like the previous step, we identify $\Gamma^{\sigma_2}_{\alpha_1,\alpha_2} = U$, where σ_2 labels upper/lower (2×4) submatrices, and $\lambda^{\sigma_2}_{\alpha_2} = \{\sqrt{\frac{2}{3}}, \frac{1}{\sqrt{3}}, 0, 0\}$.

Furthermore, we set $|\Phi^{\sigma_3\sigma_4}_{\alpha_2}\rangle = SV^\dagger$ and split again for each possible value of σ_3:

$$|\tau^{\sigma_4}_{\alpha_2,\sigma_3=0}\rangle = \frac{1}{\sqrt{3}}\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$|\tau^{\sigma_4}_{\alpha_2,\sigma_3=1}\rangle = \frac{1}{\sqrt{3}}\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

In stacked matrix form this becomes

$$|\tau^{\sigma_4}_{\alpha_2,\sigma_3}\rangle = \frac{1}{\sqrt{3}}\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

which in turn feeds into the next iteration

Third SVD

The SVD decomposition of $|\tau^{\sigma_4}_{\alpha_2,\sigma_3}\rangle$ yields:

$$U = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & \sqrt{2} & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & \sqrt{2} & 0 \end{pmatrix}$$

$$S = \frac{1}{\sqrt{3}}\begin{pmatrix} \sqrt{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$V^\dagger = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Example with PBC

Virtual systems: Valence bonds or Maximally entangled pairs

> The following is a summary of this presentation and
>
> Wahl, T. B. (2015). Tensor network states for the description of quantum many-body systems. arXiv Preprint, (August), 156.
>
> Pérez-García, D., Verstraete, F., Wolf, M. M., & Cirac, J. I. (2007). Matrix Product State Representation. Quantum Inf. Comp., 7, 401.
>
> Saberi, H. (2008). Matrix-product states for strongly correlated systems and quantum information processing. Dissertation, 141.

On a spin-S chain of length N, replace each d-dimensional system $|\sigma_i\rangle$ at site i, with two *virtual systems* $|l_i, r_i\rangle$ of dimension $D_i = 1 + 2S'$, called the bond dimension. These are sometimes called *auxiliary* systems. In general D_i can be different for each site but it it needs to be larger than the bond dimension. Note that l and r stand for left and right.

Let every pair of virtual systems be maximally entangled with the respective neighboring system. This means that $r_i = l_{i+1} = \alpha_i$. The states are written in the form:

$$|I_{i,i+1}\rangle = \sum_{\alpha_i=1}^{D} |r_i = \alpha_i, l_{i+1} = \alpha_i\rangle$$

And for the whole chain:

$$|I\rangle = \sum_{\alpha_1,\alpha_2,\ldots,\alpha_{N-1}} |\alpha_0, \alpha_1\rangle|\alpha_1, \alpha_2\rangle\ldots|\alpha_{N-1}, \alpha_N\rangle$$

where each sum denotes an *entangled bond*. **Note that without PBC the leftmost and rightmost sites have only one virtual particle**.

Then, apply a map

$$\mathcal{A}^i = \sum_{\sigma_i}\sum_{l_i,r_i} A^{\sigma_i}_{l_i,r_i}|\sigma_i\rangle\langle l_i, r_i|$$

to each of the N sites. We then obtain the MPS

$$|\psi\rangle = (\bigotimes_{i}^{N} \mathcal{A}^i)(\bigotimes_{i}^{N-1} |I_{i,i+1}\rangle) = \sum_{\sigma_1 ... \sigma_N} A^{\sigma_1} ... A^{\sigma_N} |\sigma_1 ... \sigma_N\rangle$$

Example using valence bonds:

As before, let $|\psi\rangle = \frac{1}{\sqrt{2}}(|010\rangle + |101\rangle)$ be the state of 3 qubits, $d = 2$.

Let each site be replaced by pairs of dimension $D = 2$ in a maximally entangled state, i.e. rewrite it in the (unnormalized) form:

$$\begin{aligned}
|I_{1,2}\rangle &= \sum_{\alpha_1} |r_1 = \alpha_1, l_2 = \alpha_1\rangle = |0,0\rangle_{r_1,l_2} + |1,1\rangle_{r_1,l_2} \\
|I_{2,3}\rangle &= \sum_{\alpha_2} |r_2 = \alpha_2, l_3 = \alpha_2\rangle = |0,0\rangle_{r_2,l_3} + |1,1\rangle_{r_2,l_3}
\end{aligned}$$

where the subscripts r_i, l_i are there to remind us that this is the right and left virtual particles corresponding to site i. Now apply the map $\mathcal{A}$ on each site. :

$$\begin{aligned}
\mathcal{A}^1 |I_{1,2}\rangle &= \sum_{\sigma_1} \sum_{r_1} A^{\sigma_1}_{r_1} |\sigma_1\rangle\langle r_1| (\sum_{\alpha_1} |\alpha_1, \alpha_1\rangle) \\
&= \sum_{\sigma_1} A^{\sigma_1}_0 |\sigma_1\rangle_{r_1} \langle 0|0,0\rangle_{r_1,l_2} + A^{\sigma_1}_1 |\sigma_1\rangle_{r_1} \langle 1|1,1\rangle_{r_1,l_2} \\
&= \sum_{\sigma_1} A^{\sigma_1}_0 |\sigma_1\rangle |0\rangle_{l_2} + A^{\sigma_1}_1 |\sigma_1\rangle |1\rangle_{l_2}
\end{aligned}$$

Similarly,

$$\begin{aligned}
\mathcal{A}^2 |I_{1,2}\rangle &= \sum_{\sigma_2} \sum_{l_2,r_2} A^{\sigma_2}_{l_2,r_2} |\sigma_2\rangle\langle l_2, r_2| (\sum_{\alpha_1} |\alpha_1, \alpha_1\rangle) \\
&= \sum_{\sigma_2} (A^{\sigma_2}_{0,0} |\sigma_2\rangle\langle 0,0| + A^{\sigma_2}_{0,1} |\sigma_2\rangle\langle 0,1| + A^{\sigma_2}_{1,0} |\sigma_2\rangle\langle 1,0| + A^{\sigma_2}_{1,1} |\sigma_2\rangle\langle 1,1|)_{l_2,r_2} (|0,0\rangle_{r_1,l_2} + |1,1\rangle_{r_1,l_2}) \\
&= \sum_{\sigma_2} A^{\sigma_2}_{0,0} |\sigma_2\rangle_{r_2} \langle 0|0\rangle_{r_1} + A^{\sigma_2}_{0,1} |\sigma_2\rangle_{r_2} \langle 0|1\rangle_{r_1} + A^{\sigma_2}_{1,0} |\sigma_2\rangle_{r_2} \langle 1|0\rangle_{r_1} + A^{\sigma_2}_{1,1} |\sigma_2\rangle_{r_2} \langle 1|1\rangle_{r_1} \\
\mathcal{A}^2 |I_{2,3}\rangle &= \sum_{\sigma_2} \sum_{l_2,r_2} A^{\sigma_2}_{l_2,r_2} |\sigma_2\rangle\langle l_2, r_2| (\sum_{\alpha_2} |\alpha_2, \alpha_2\rangle) \\
&= \sum_{\sigma_2} A^{\sigma_2}_{0,0} |\sigma_2\rangle_{l_2} \langle 0|0\rangle_{l_3} + A^{\sigma_2}_{0,1} |\sigma_2\rangle_{l_2} \langle 0|1\rangle_{l_3} + A^{\sigma_2}_{1,0} |\sigma_2\rangle_{l_2} \langle 1|0\rangle_{l_3} + A^{\sigma_2}_{1,1} |\sigma_2\rangle_{l_2} \langle 1|1\rangle_{l_3} \\
\mathcal{A}^3 |I_{2,3}\rangle &= \sum_{\sigma_3} \sum_{l_3} A^{\sigma_3}_{l_3} |\sigma_3\rangle\langle l_3| (\sum_{\alpha_2} |\alpha_2, \alpha_2\rangle) \\
&= \sum_{\sigma_3} A^{\sigma_3}_0 |\sigma_3\rangle_{l_3} \langle 0|0,0\rangle_{r_2,l_3} + A^{\sigma_3}_1 |\sigma_3\rangle_{l_3} \langle 1|1,1\rangle_{r_2,l_3} \\
&= \sum_{\sigma_3} A^{\sigma_3}_0 |\sigma_3\rangle |0\rangle_{r_2} + A^{\sigma_3}_1 |\sigma_3\rangle |1\rangle_{r_2}
\end{aligned}$$

All cross-terms such as ${}_{r_i}\langle n|m\rangle_{r_i} ={}_{l_i} \langle n|m\rangle_{l_i} = \delta_{nm}$. All other combinations are equal to zero, i.e. $\mathcal{A}^1|I_{2,3}\rangle = \mathcal{A}^3|I_{1,2}\rangle = 0$.

Now we multiply

$$(\mathcal{A}^1 \otimes \mathcal{A}^2)(|I_{1,2}\rangle \otimes |I_{2,3}\rangle) = \sum_{\sigma_1,\sigma_2} A^{\sigma_1}_0 A^{\sigma_2}_{0,0} |0\rangle_{l_3} + A^{\sigma_1}_0 A^{\sigma_2}_{0,1} |1\rangle_{l_3}$$

$$|\psi\rangle = (\bigotimes_i^3 \mathcal{A}^i)(\bigotimes_i^2 |I_{i,i+1}\rangle) =$$

Comment: Hmmm. This seems overly impractical... perhaps I've misunderstood something

Bond Dimension

Each $A^{\sigma_i}_{a_i,a_{i+1}}$ above is an $(r_i \times r_{i+1})$ matrix, where r_i is the rank of the Schmidt decomposition at each site i. The *bond dimension* of an MPS is defined by

$$D \equiv \max_i r_i$$

Graphical representation

The following is a summary of the review paper by

Eisert, J. (2013). Entanglement and tensor network states.

We represent mathematical objects in the following way:

Scalar Vector Dual Vector Matrix

Trace Partial Trace Scalar Product

Let j_i be a particle with spin-1/2 at position i on a chain with n particles. Then the tensor $c_{j_1,j_2,\dots,j_n}$ is the collection of complex numbers that tells us in what linear combination a state is in, in terms of its basis vectors:

$$|\psi\rangle = \sum_{j_1,j_2\dots j_n} c_{j_1,j_2,\dots,j_n} |j_1, j_2, \dots, j_n\rangle$$

The Schmidt decomposition allows us to rewrite this tensor in terms of matrices. If we use periodic boundary conditions this looks like

$$|\psi\rangle = \sum_{a_1,a_2\dots,a_n}^{r_1,r_2,\dots,r_n} \mathrm{Tr}(A^{j_1}_{a_1,a_2} A^{j_2}_{a_2,a_3} \dots A^{j_n}_{a_n,a_1}) |j_1, j_2, \dots, j_n\rangle$$

where the trace takes care of the periodic boundary.

Remarks on Implementation

The unsupported Eigen::Tensor

Pro	Con
Fast vectorized operations	Very strict about correct dimensions
Easy reshaping and contracting.	Need to flatten before SVD

Pro	Con
A wrapper with map functions enables matrix representations.	Lacking documentation

The ITensor library

Pro	Con
“Intelligent” (minimal number of operations in contractions)	Black box
Can perform SVD on tensors directly	Counterintuitive contractions
	Documentation is unclear on syntax.