Infinite Time-Evolving Block Decimation (iTEBD)

[Vidal2007], [Schollwöck2011, Sec. 10.4]

Goal: ground state search for infinite system while exploiting translational invariance.

We will use Vidal's $\Gamma\Lambda$ notation [see MPS-V], but everything can be translated into AB notation.

Basic idea: 'imaginary time evolution':

$$\lim_{\beta\to\infty} e^{-\beta\hat{H}}|\psi\rangle \propto |g\rangle \tag{1}$$

Reason: high-energy states die out quickly (if ground state is gapped):

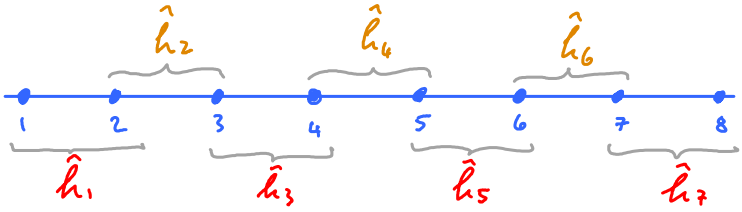
$$e^{-\beta\hat{H}} = \sum_\alpha e^{-\beta\hat{H}}|\alpha\rangle\langle\alpha| \xrightarrow{\beta\to\infty} \underbrace{e^{-\beta E_g}|g\rangle\langle g|}_{\text{projector onto ground state}} \tag{2}$$

1. Trotter decomposition of time evolution operator [Schollwöck2011, Sec. 7.1.1]

General: write Hamiltonian as

$$\hat{H} = \sum_\ell \hat{h}_\ell = \hat{H}_o + \hat{H}_e \tag{3}$$

($\hat{h}_\ell$ connects sites ℓ and $\ell+1$; $\hat{H}_o$: odd, $\hat{H}_e$: even)

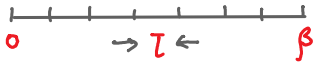


(4)

Then all odd terms mutually commute, and all even terms mutually commute:

$$[\hat{h}_\ell, \hat{h}_{\ell'}] = 0 \quad \text{if } \ell,\ell' \text{ are both odd or both even} \tag{5}$$

Divide time interval into L slices: $\beta = \tau N_\beta$



(6)

$$e^{-\beta\hat{H}} \overset{\text{Trotter decomposition}}{=} \left[e^{-\tau\hat{H}}\right]^{N_\beta} = \left[e^{-\tau(\hat{H}_o+\hat{H}_e)}\right]^{N_\beta} \tag{7}$$

$$\xrightarrow{N_\beta\to\infty,\ \tau\to 0} \simeq \left[e^{-\tau\hat{H}_o}\, e^{-\tau\hat{H}_e} + \mathcal{O}(\tau^2)\right]^{N_\beta} \quad \text{'first order Trotter approx.'} \tag{8}$$

$$\text{or} \quad \simeq \left[e^{-\frac{\tau}{2}\hat{H}_o}\, e^{-\tau\hat{H}_e}\, e^{-\frac{\tau}{2}\hat{H}_o} + \mathcal{O}(\tau^3)\right]^{N_\beta} \quad \text{'second order Trotter approx.'} \tag{9}$$

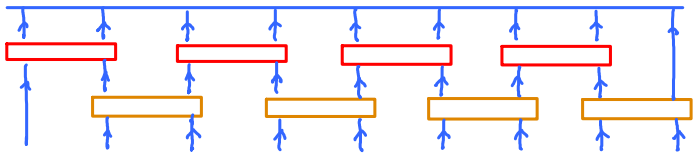
Exploiting (5), odd and even exponents can both be expanded separately without further approximation:

$$e^{-\tau\hat{H}_o} = e^{-\tau\hat{h}_1}\, e^{-\tau\hat{h}_3} \cdots e^{-\tau\hat{h}_{N-1}} \equiv U_{[1]}U_{[3]}\cdots U_{[N-1]} \tag{10a}$$

$$e^{-\tau\hat{H}_e} = e^{-\tau\hat{h}_2}\, e^{-\tau\hat{h}_4} \cdots e^{-\tau\hat{h}_N} \equiv U_{[2]}U_{[4]}\cdots U_{[N]} \tag{10b}$$

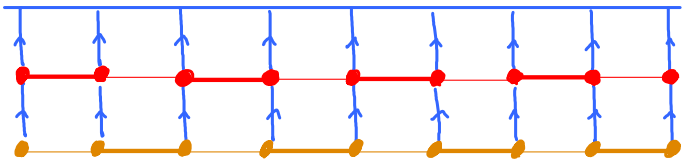
So, when applying $e^{-\beta \hat{H}}$ to $|\psi\rangle$, we can successively apply all odd terms, then truncate, then all even ones, then truncate, etc.

$e^{-\tau \hat{H}_e}\, e^{-\tau \hat{H}_o}\, |\psi\rangle =$



(11)

in MPO notation: =



(12)

since $\hat{H}_o$ factorizes, even bonds have dimension $D_{w,e} = 1$

since $\hat{H}_e$ factorizes, odd bonds have dimension $D_{w,o} = 1$

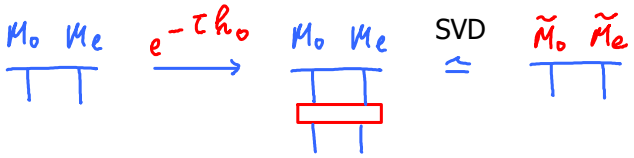
All of this can be done for finite chain of length N. But a simplification occurs for $N \to \infty$

Then we can exploit translational invariance:

Adopt a two-site unit cell (no left- or right-normalization implied).

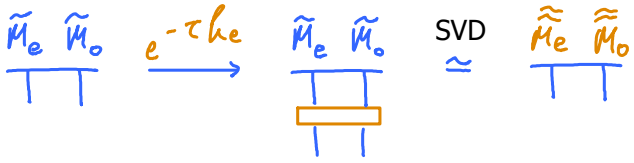
Step 1: time-evolve 'odd bond':

(first site odd, second site even)



Step 2: time-evolve (updated!) even bond:

(first site even, second site odd)



Iterate until convergence! (To discuss details, we will use $\Gamma\Lambda$ notation.)

iETBD is a 'power method': the projector to the ground state is constructed as an increasing number of powers of $e^{-\tau \hat{H}_e}\, e^{-\tau \hat{H}_o}$.

This is to be contrasted to DMRG ground state search, which is a variational method.

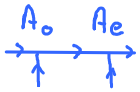
Main advantage of iTEBD: costs not proportional to system size, hence comparatively cheap.

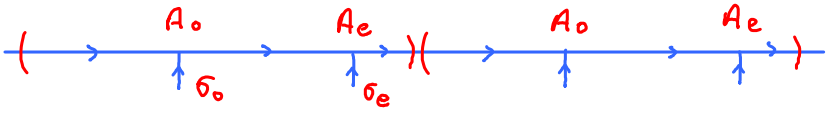
Main disadvantage: loss of orthogonality due to projection, without explicit reorthogonalization.

to be explained below

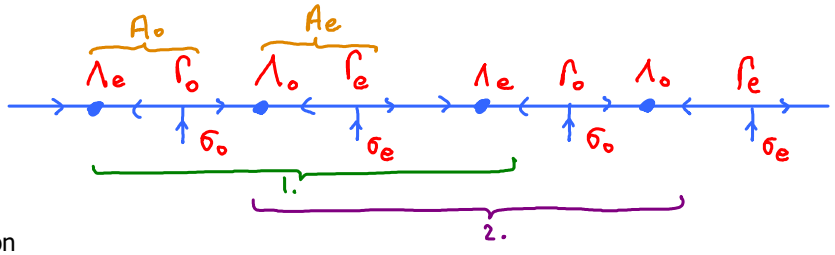
2. iTEBD: Explicit formulation

[Vidal2007], [Schollwöck2011, Sec. 10.4]

Use two-cite unit cell,  , repeated periodically,

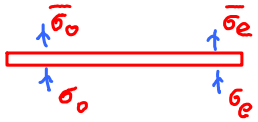
$|\psi\rangle$ =  (1)

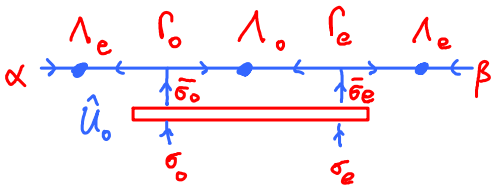
and express it in $\Gamma\Lambda$ notation: $A_o = \Lambda_e \Gamma_o$, $A_e = \Lambda_o \Gamma_e$

$|\psi\rangle$ =  (2)

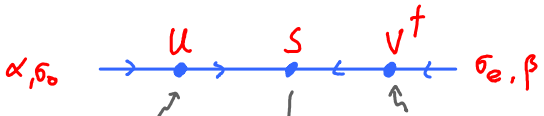
(to avoid cluttering, σ indices on Γ, A are not displayed, but implicitly understood)

Each iTEBD iteration involves two steps:

Step 1: Time-evolve odd bond using $\hat{U}_o \equiv e^{-\tau \hat{h}_o}$ =  (3)

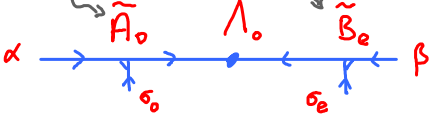


contract, reshape, = SVD

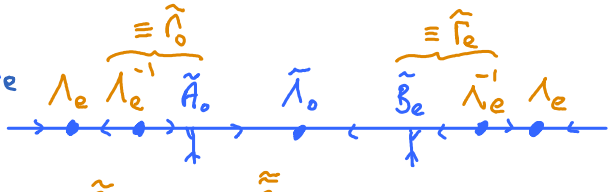
 (4a)

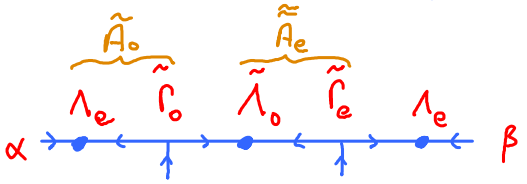
left-normalized, right-normalized

truncate, normalize = reshape

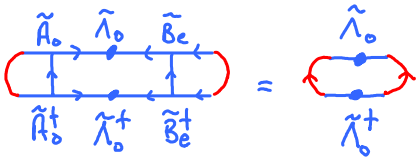
 (4b)

define $\tilde{\Gamma}_o \equiv \Lambda_e^{-1} \tilde{A}_o$, $\tilde{\Gamma}_e \equiv \tilde{B}_e \Lambda_e^{-1}$

reinstate Λ_e =  (5)

≡  (6)

$\hat{U}_o$ is projector (not unitary operation), hence reduces norm. Thus, $\tilde{\Lambda}_o$ is normalized to unity by hand:

$\tilde{\Lambda}_o \equiv \frac{S_{trunc}}{[\mathrm{Tr}(S_{trunc}^\dagger S_{trunc})]^{1/2}}$, then  = $\mathrm{Tr}\, \tilde{\Lambda}_o^\dagger \tilde{\Lambda}_o = 1$. (7)

(6) completes update of odd bond. The updated MPS now has the form $|\tilde{\psi}\rangle = \sum_{\vec{\sigma}} \tilde{A}_o \tilde{\tilde{A}}_e \tilde{A}_o \tilde{\tilde{A}}_e \ldots$ (8)

Updated bond energy :

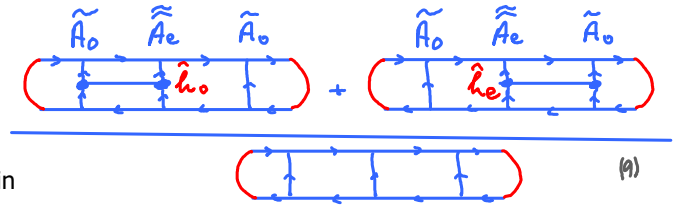
(8)

Updated bond energy :

$$\bar{h}_{bond} = \tfrac{1}{2}(\bar{h}_o + \bar{h}_e) \simeq$$

consider only two sites

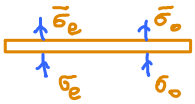
ignore tensors describing rest of chain



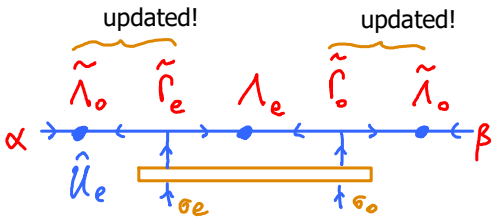
(9)

Updating odd bond lowers $\bar{h}_o$, slightly raises $\bar{h}_e$ ('odd bond much happier, even bond slightly unhappier').

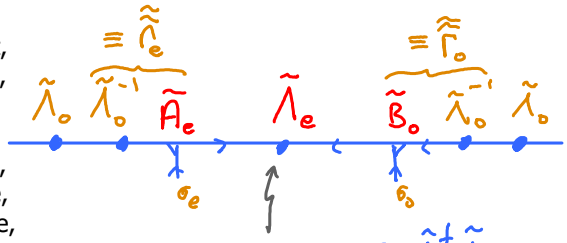
Step 2: Time-evolve even bond, using $\hat{U}_e \equiv e^{-\tau \hat{h}_e} =$



(10)



contract, reshape, SVD, $\simeq$ reshape, truncate, normalize, reinstate



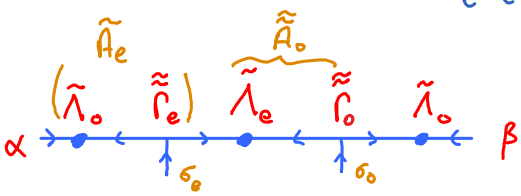
(11)

with normalization $\mathrm{Tr}\, \tilde{\Lambda}_e^\dagger \tilde{\Lambda}_e = 1$ (12)

define $\tilde{\tilde{\Gamma}}_e \equiv \tilde{\Lambda}_e^{-1} \tilde{A}_e$

$\tilde{\tilde{\Gamma}}_o \equiv \tilde{B}_o \tilde{\Lambda}_o^{-1}$

$=$



(13)

(12) completes update of even bond. Updated MPS now has the form $|\tilde{\tilde{\psi}}\rangle = \prod_{\vec{\sigma}} |\vec{\sigma}\rangle \tilde{A}_e \tilde{\tilde{A}}_o \tilde{A}_e \tilde{\tilde{A}}_o$ (14)

Compute updated bond energy using (8), with $o \leftrightarrow e$.

Updating even bond lowers $\bar{h}_e$, slightly raises $\bar{h}_o$ ('even bond much happier, odd bond slightly unhappier').

Now iterate (apply $\hat{U}_o$, then $\hat{U}_e$, etc.) until convergence is reached (monitor ground state energy…)

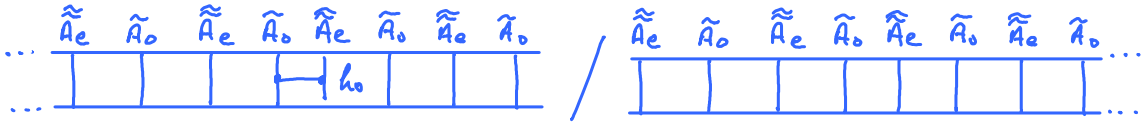
Remarks:

1. In principle, computation of Λ_o^{-1}, $\tilde{\Lambda}_e^{-1}$ can become unstable, because singular values can be very small. Thus: truncate by discarding smallest singular values $\lesssim 10^{-8}$, only then invert.

2. Note that $\tilde{A}_o$, $\tilde{A}_e$ is left-normalized, but $\tilde{\tilde{A}}_e \overset{(6,5)}{=} \tilde{\Lambda}_o \tilde{B}_e \tilde{\Lambda}_e^{-1}$ is not! 'Loss of orthogonality'.

$\tilde{\tilde{A}}_o \overset{(13,11)}{=} \tilde{\Lambda}_e \tilde{B}_o \tilde{\Lambda}_o^{-1}$

This causes problems when computing expectation values. For example, odd bond energy, given by

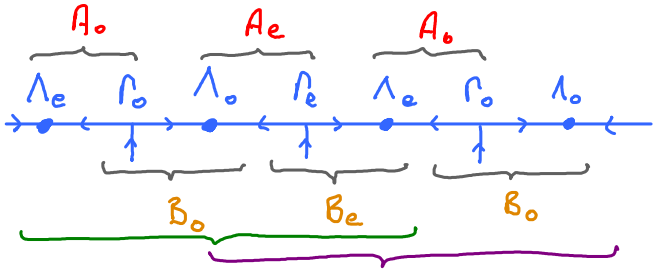


/

does not reduce to (9), because zippers can not be closed from left and right. Hence (9) involves an approximation, namely ignoring the rest of the chain.

3. iETBD: Hastings' method [Hastings2009, Sec. II.A], [Schollwöck2011, Sec. 7.3.2]

Inverting singular-value matrix (i.e. dividing by small singular values) can be avoided as follows:

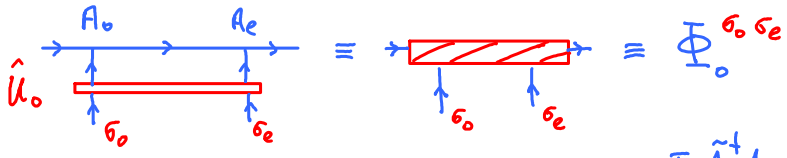
Write $|\psi\rangle =$  (13)

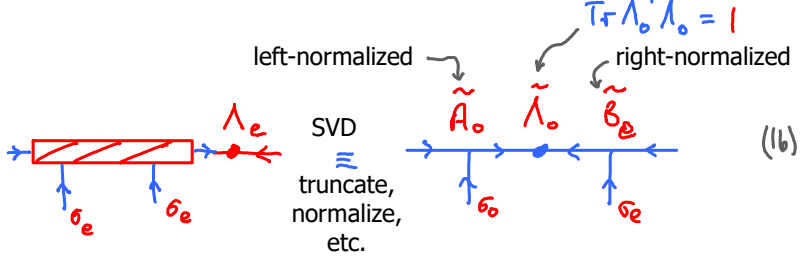
with $A_o = \Lambda_e \Gamma_o \quad A_e = \Lambda_o \Gamma_e \quad B_o = \Gamma_o \Lambda_o, \quad B_e = \Gamma_e \Lambda_e$ (14)

Step 1: Time-evolve odd bond $A_o \Lambda_o B_e$ to define $\tilde{A}_o \tilde{\Lambda}_o \tilde{B}_e$ via SVD, and $\tilde{\tilde{A}}_e$ via contraction.

Step 2: Time-evolve even bond $\tilde{\tilde{A}}_e \Lambda_e B_o$ to define $\tilde{A}_e \tilde{\Lambda}_e \tilde{B}_o$ via SVD, and $\tilde{\tilde{A}}_o$ via contraction.

Then rename $\tilde{\tilde{A}}_o \to A_o, \quad \tilde{A}_e \to A_e, \quad \tilde{\Lambda}_{o,e} \to \Lambda_{o,e}, \quad \tilde{B}_{o,e} = B_{o,e}$, and iterate.

Step 1 (odd-even): Compute  (15)

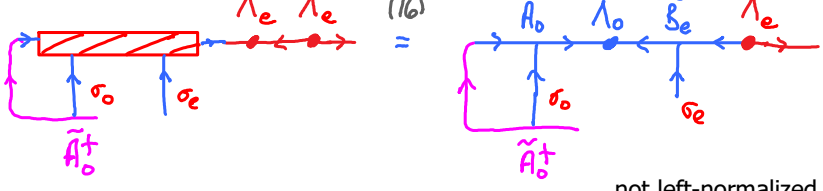
Do SVD on $\Phi_o^{\sigma_o \sigma_e} \Lambda_e =$ 

(16)

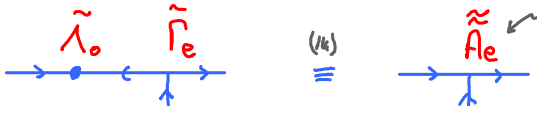
SVD yields updated tensors $\tilde{A}_o, \tilde{\Lambda}_o, \tilde{B}_e$. Note that the outgoing leg of $\tilde{A}_o$ involves a truncation, governed by $\tilde{\Lambda}_o$. Since this is also the incoming leg of A_e , we need an updated $\tilde{\tilde{A}}_e$, involving a truncation governed by $\tilde{\Lambda}_o$ on its incoming leg. This is achieved by left-contraction with $\tilde{A}_o^\dagger$:

$$\tilde{\tilde{A}}_e^{\sigma_e} \equiv \tilde{A}_{o\sigma_o}^\dagger \Phi_o^{\sigma_o \sigma_e}$$ [double-tilde denotes: not left-normalized, see (19)] (17)

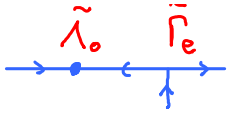
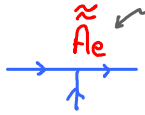
(Note: no inversion of singular matrix required!) Justification for this definition:

$\tilde{A}_{o\sigma_o}^\dagger \Phi_o^{\sigma_o \sigma_e} \overset{(15)}{=}$ 

(18)

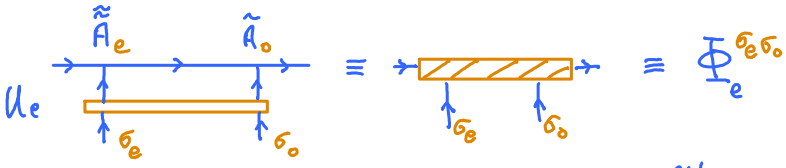
$\tilde{A}_{o\sigma_o}^\dagger \tilde{A}_o^{\sigma_o} = \mathbb{1} =$ 

(19)

$\tilde{A}^{T}_{o\sigma_o}\,\tilde{A}^{\sigma_o}_{o} = \mathbb{1}$ $=$  (14) $\equiv$  (19)

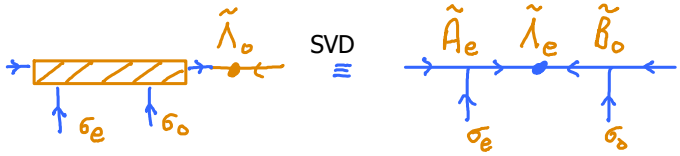
where we associated $\tilde{\Gamma}_e \equiv \tilde{B}_e \Lambda_e^{-1}$ and $\tilde{\tilde{A}}_e \equiv \tilde{\Lambda}_o \tilde{\Gamma}_e$ by analogy to (14). (20)

This concludes step 1. We now have updated tensors $\tilde{A}_o, \tilde{\Lambda}_o, \tilde{B}_e, \tilde{\tilde{A}}_e$, but not updated $\tilde{\Lambda}_e, \tilde{B}_o$

Step 2 (even-odd): Compute  (21)

$\mathrm{tr}\,\tilde{\Lambda}_e^{\dagger}\tilde{\Lambda}_e = 1$

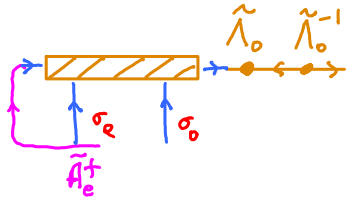
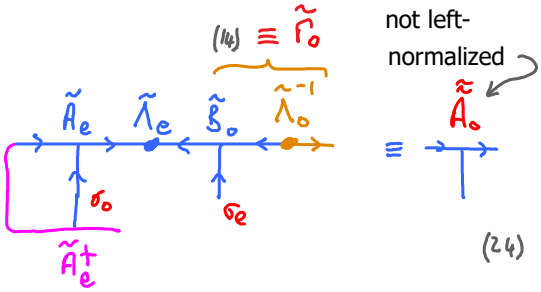
left-normalized right-normalized

Do SVD on $\Phi_e^{\sigma_e\sigma_o}\,\tilde{\Lambda}_o =$  (22)

The SVD yields updated tensors $\tilde{A}_e, \tilde{\Lambda}_e, \tilde{B}_o$, and $\tilde{A}_e$ has a $\tilde{\Lambda}_e$ truncation on its outgoing leg, i.e. incoming leg of $\tilde{A}_o$. Define an updated $\tilde{\tilde{A}}_o$ with matching $\tilde{\Lambda}_e$ truncation on incoming leg, by

$$\tilde{\tilde{A}}^{\sigma_o}_{o} \equiv \tilde{A}^{\dagger}_{e\sigma_e}\,\Phi_e^{\sigma_e\sigma_o} \qquad \text{[not left-normalized, see (24)]} \qquad (23)$$

Justification:

$\tilde{A}^{\dagger}_{e\sigma_e}\,\Phi_e^{\sigma_e\sigma_o}$ (21) $=$  (22) $=$  (24)

where we associated $\tilde{\Gamma}_o \equiv \tilde{B}_o \tilde{\Lambda}_o^{-1}$ and $\tilde{\tilde{A}}_o \equiv \tilde{\Lambda}_e \tilde{\Gamma}_o$ by analogy to (14). (25)

This concludes step 2.

Now iterate (apply $\hat{U}_o$, then $\hat{U}_e$, etc.) until convergence of bond energy is reached.

Compute bond energy using (iTEBD2.9) for step 1, or its $o \hookrightarrow e$ version for step 2.

Concluding remarks:

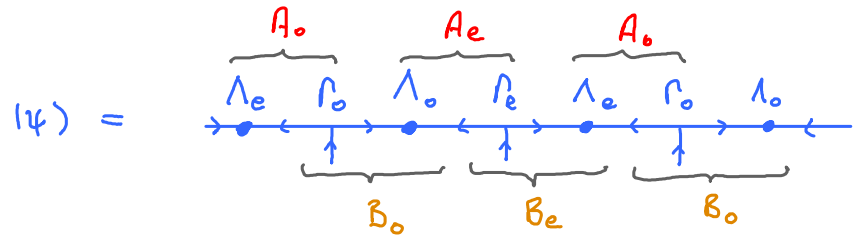
Main advantage of iTEPD: costs not proportional to system size, hence comparatively cheap.

Main disadvantage: loss of orthogonality due to projection, without explicit reorthogonalization.

4. Orthonormalization

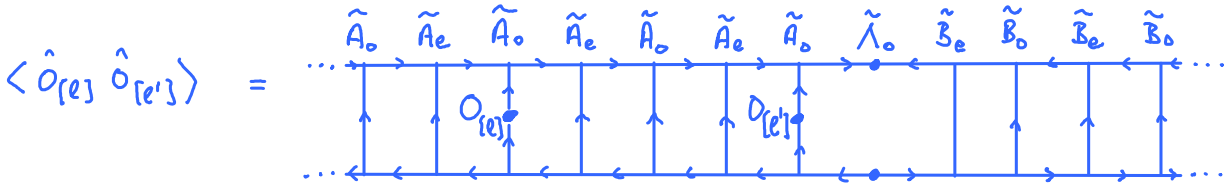
Correlators via transfer matrix [Schollwöck2011, Sec. 10.5.1]

An infinite, translationally invariant MPS with two-site unit cell, expressed in the form

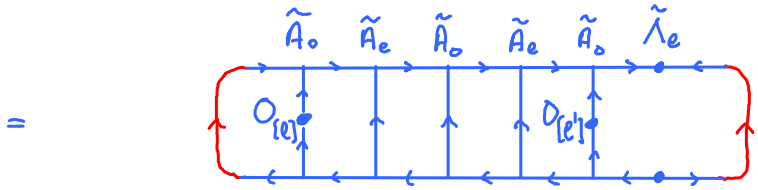


is called 'canonical' if $A_{o,e}$ are left-normalized and $B_{o,e}$ are right-normalized.

Correlators can then be computed using transfer matrix methods:



close zippers



Problem: iTEBD (including Hastings' version) yields infinite MPS that are not in canonical form, due to loss of orthogonality. It is possible to restore orthogonality (albeit at the cost of inverting singular value matrices).

[Orus2008], [Schollwöck2011, Sec. 10.5]