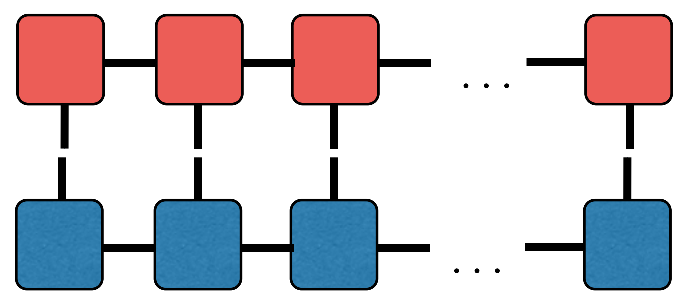
Tensor network techniques in 1D

Jorge Yago Malo

Group meeting Pisa 2022

Tensor network techniques

$$\langle \psi | \phi \rangle =$$



U. Schollwoeck, Annals of Physics 326, 96 (2011)

G. Vidal, Phys. Rev. Lett. **91**, 147902 (2003)
G. Vidal, Phys. Rev. Lett. **93**, 040502 (2004)
A. J. Daley, C. Kollath, U. Schollwöck, and G.Vidal, Journal Stat. Mech.: Th and Exp. P04005 (2004)

S. R. White and A. E. Feiguin, PRL **93**, 076401 (2004)
F. Verstraete, V. Murg, and J. I. Cirac, Advances in Physics 57, 143 (2008).

Tensor networks out there: existing open access libraries

- **TNT**: tensor network theory (https://arxiv.org/pdf/1610.02244.pdf)

- **iTensor**: intelligent tensor (https://itensor.org/)

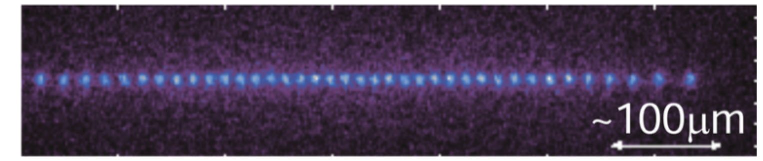
- **ALPS**: algorithms and libraries for physics simulations (http://alps.comp-phys.org/mediawiki/index.php/Main_Page)

- **OSMPS**: Open Source Matrix Product States (https://openmps.sourceforge.io/)

- …

Quantum many-body systems $\longrightarrow$ PROBLEM

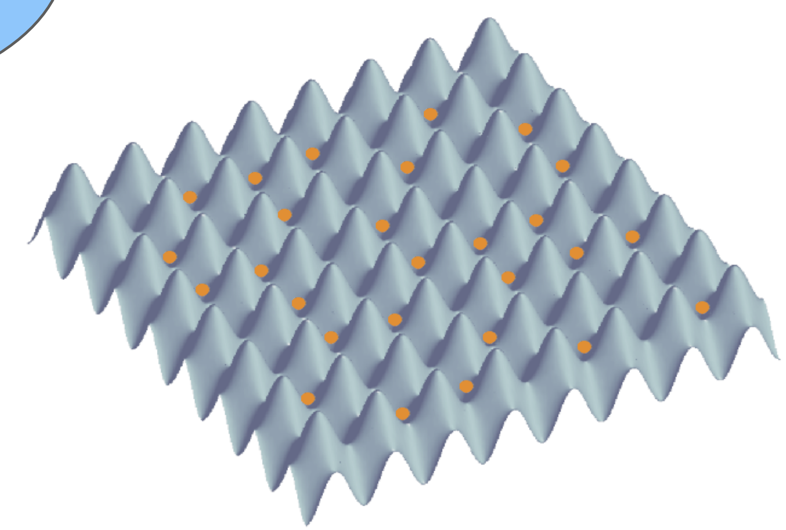
- We cannot model the whole system:



Local dimension d

System size M

$$\mathcal{H} \propto d^M$$



- E.g. :

Two-level spin system $\sigma = +, -$ $\quad (d = 2)$

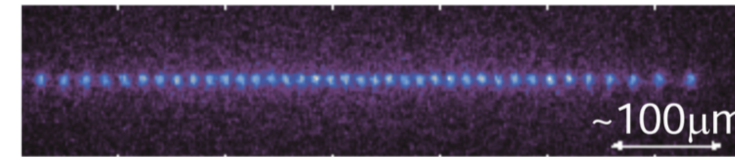
$$M = 3 \qquad dim(\mathcal{H}) = 2^3 = 8$$

$$M = 20 \qquad dim(\mathcal{H}) = 2^{20} = 1048576$$

- Other systems: bosons/fermions in optical lattices, …

Quantum many-body systems $\longrightarrow$ PROBLEM

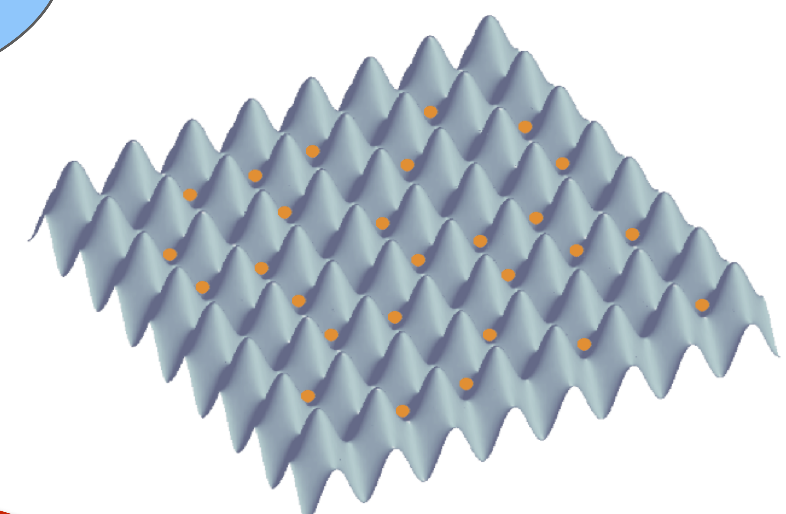
- We cannot model the whole system:



Local dimension d

System size M

$$\mathcal{H} \propto d^M$$



- E.g. :

Two-level spin system $\sigma = +, -$ $(d = 2)$

$M = 3$ $\quad dim(\mathcal{H}) = 2^3 = 8$

$M = 20$ $\quad dim(\mathcal{H}) = 2^{20} = 1048576$

Exponentially large!

- Other systems: bosons/fermions in optical lattices, …

What does Nature have to say?

- We observe that **physical** systems exhibit **low entanglement**.

What does Nature have to say?

- We observe that **physical** systems exhibit **low entanglement**.

GS and first excited states of Hamiltonians
(**1D**, local interactions…)

What does Nature have to say?

- We observe that **physical** systems exhibit **low entanglement**.

GS and first excited states of Hamiltonians
(**1D**, local interactions…)

- Why?

  - High entanglement does not help reducing the energy of the system.

  - Area law: J. Eisert, M. Cramer, and M. B. Plenio, Rev. Mod. Phys. **82**, 277 (2010)

    Entanglement between two subregions scale as the size of their boundary for gapped local Hamiltonians.

What does Nature have to say?

- We observe that **physical** systems exhibit **low entanglement**.

GS and first excited states of Hamiltonians
(**1D**, local interactions…)

- Why?

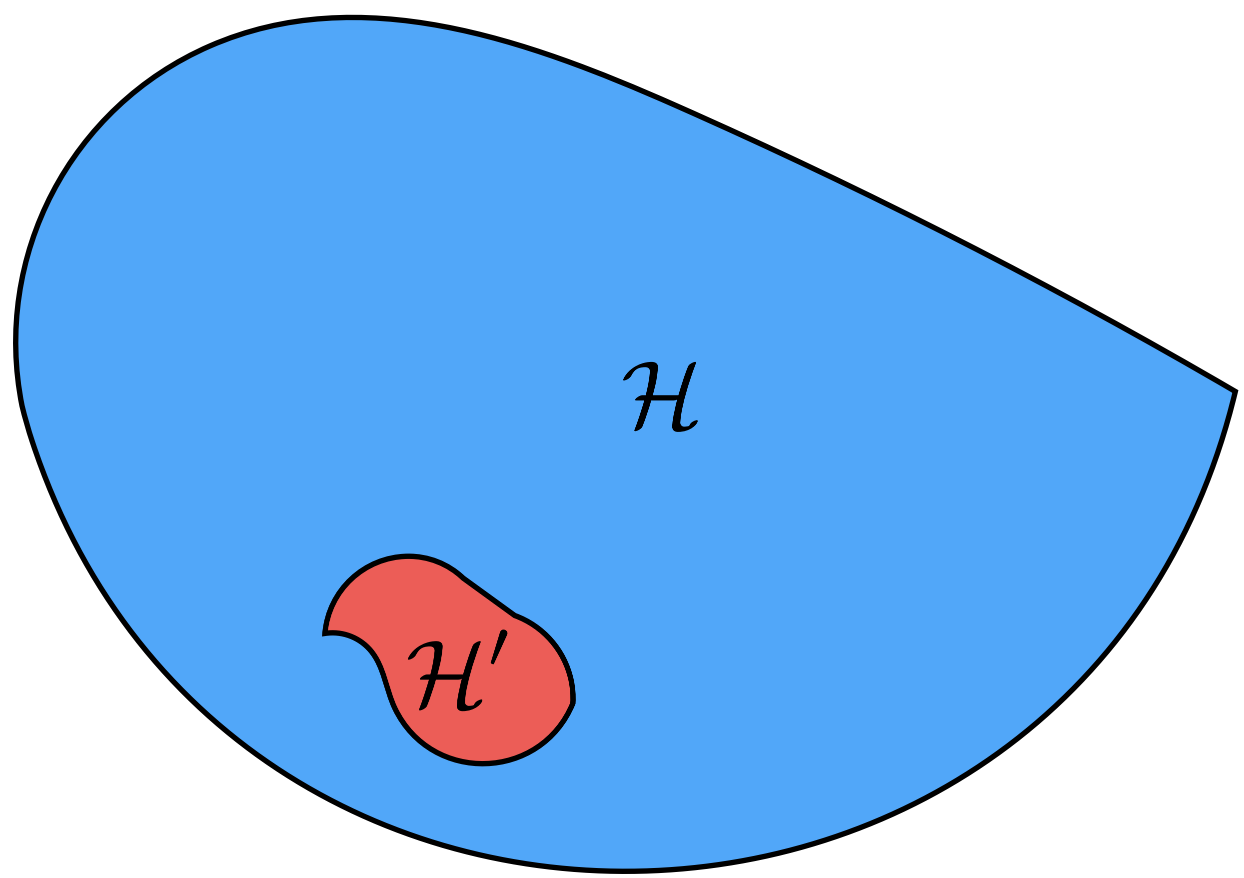
- High entanglement does not help reducing the energy of the system.

- Area law.

- What is then the real picture?

The relevant region of the Hilbert space is much smaller.

We need an **efficient representation**



What does Nature have to say?

- We observe that **physical** systems exhibit **low entanglement**.

GS and first excited states of Hamiltonians
(**1D**, local interactions…)

- Why?

  - High entanglement does not help reducing the energy of the system.

  - Area law.

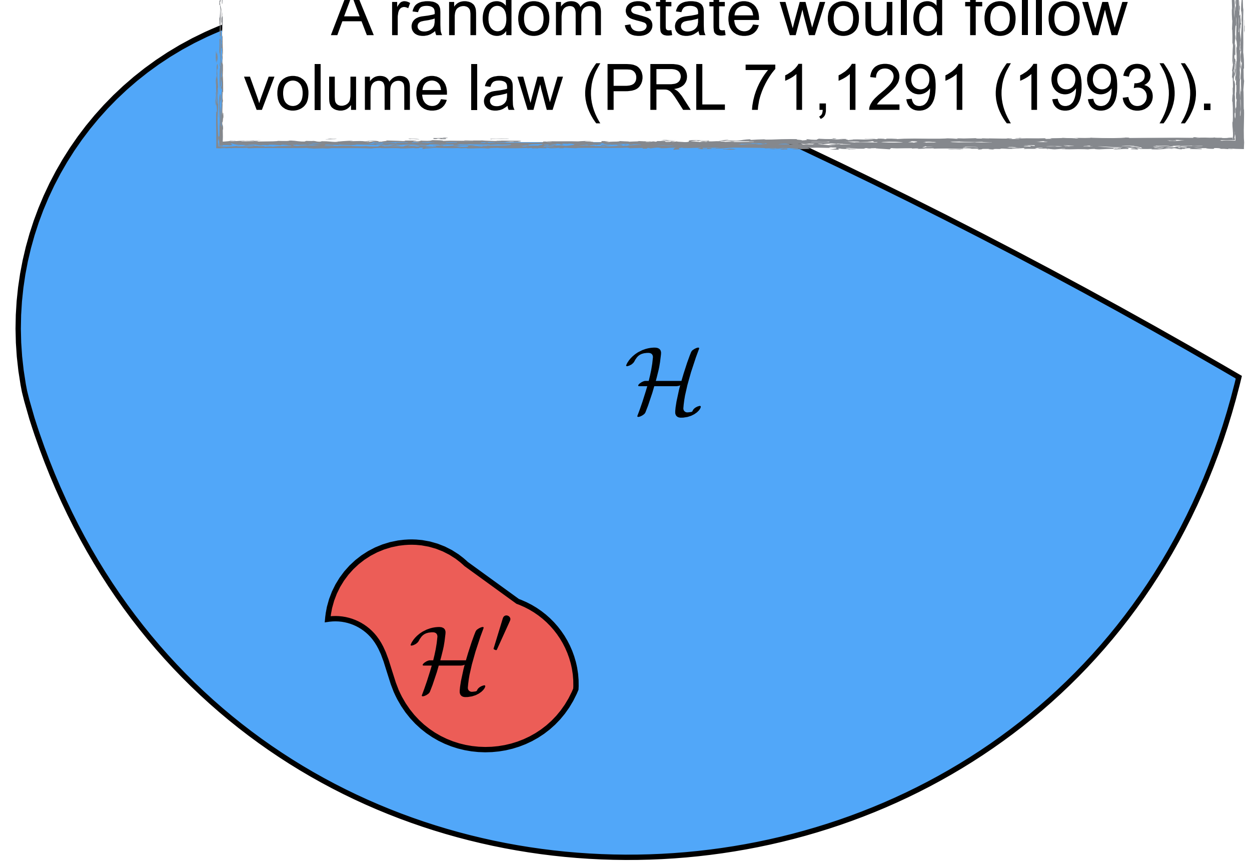
- What is then the real picture?

The relevant region of the
Hilbert space is much smaller.

↓

We need an **efficient representation**

A random state would follow volume law (PRL 71,1291 (1993)).



Key decomposition towards efficient tensor networks:

- I) Schmidt decomposition: from quantum info, allows for the representation of a given **pure** quantum state in the basis of two subsystems of the total system.

$$|\phi\rangle = \sum_{ab} \Phi_{ab} |a\rangle_A |b\rangle_B$$

$|a\rangle_A, |b\rangle_B$: orthonormal bases of subsystems A and B

Φ_{ab} : matrix of coefficients.

Key decomposition towards efficient tensor networks:

- I) Schmidt decomposition: from quantum info, allows for the representation of a given **pure** quantum state in the basis of two subsystems of the total system.

$$|\phi\rangle = \sum_{ab} \Phi_{ab} |a\rangle_A |b\rangle_B$$

$|a\rangle_A, |b\rangle_B$: orthonormal bases of subsystems A and B

Φ_{ab} : matrix of coefficients.

- II) Singular Value Decomposition: allows for the decomposition of an arbitrary matrix

$$M = USV^\dagger$$

U : matrix for by the orthonormal left singular vectors, fulfilling $U^\dagger U = I$

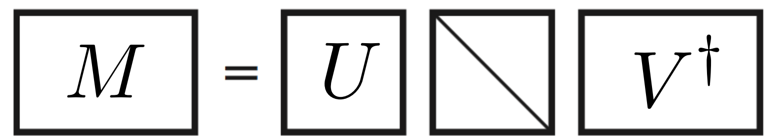
$S = diag(s_1, s_2, ..., s_r, 0, ..., 0)$: matrix of singular values (with **r** non-zero entries).

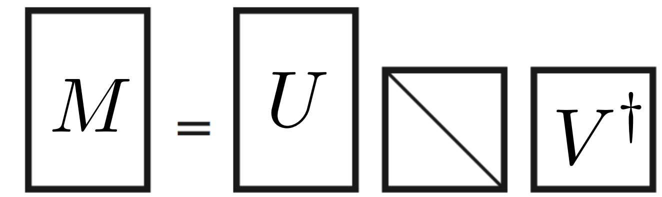
$V^\dagger$: matrix for by the orthonormal right singular vectors, fulfilling $VV^\dagger = I$

Key decomposition towards efficient tensor networks:

- II) Singular Value Decomposition: allows for the decomposition of a generic matrix

$$M = USV^\dagger$$

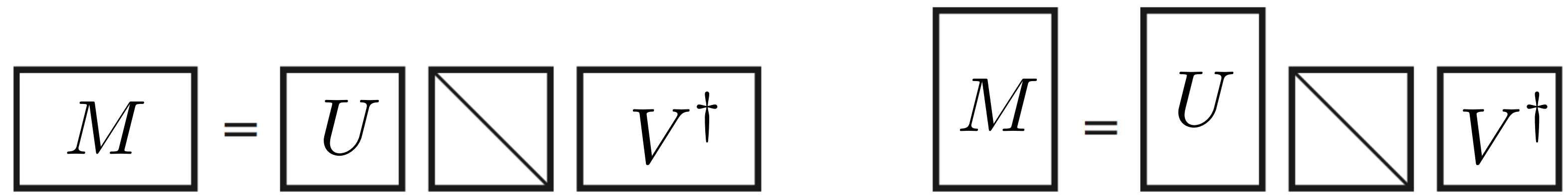




Key decomposition towards efficient tensor networks:

- II) Singular Value Decomposition: allows for the decomposition of a generic matrix

$$M = USV^\dagger$$



If we apply this to the Schmidt decomposition we obtain:

$$|\phi\rangle = \sum_{ab} \Phi_{ab} |a\rangle_A |b\rangle_B = \sum_{ab} \sum_r U_{ar} S_{rr} V^\dagger_{rb} |a\rangle_A |b\rangle_B =$$

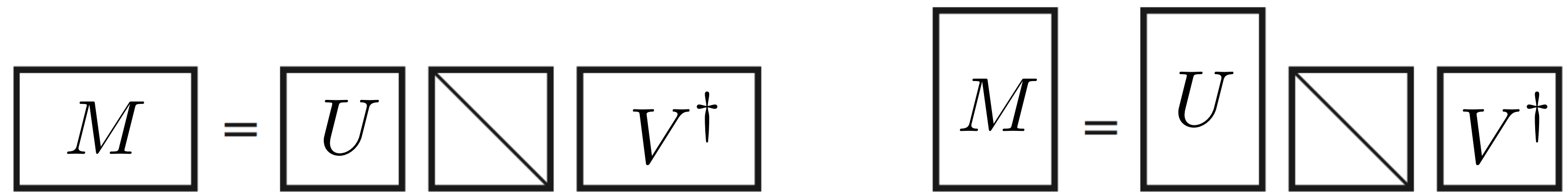
$$= \sum_{ab} \sum_r U_{ar} S_{rr} V^*_{br} |a\rangle_A |b\rangle_B = \sum_r (\sum_a U_{ar} |a\rangle_A) S_{rr} (\sum_b V^*_{br} |b\rangle_B) =$$

$$= \sum_r S_{rr} |a'\rangle_A |b'\rangle_B = \sum_r s_r |a'\rangle_A |b'\rangle_B$$

Key decomposition towards efficient tensor networks:

- II) Singular Value Decomposition: allows for the decomposition of a generic matrix

$$M = USV^\dagger$$



If we apply this to the Schmidt decomposition we obtain:

$$|\phi\rangle = \sum_{ab} \Phi_{ab} |a\rangle_A |b\rangle_B = \sum_{ab} \sum_r U_{ar} S_{rr} V^\dagger_{rb} |a\rangle_A |b\rangle_B =$$

$$= \sum_{ab} \sum_r U_{ar} S_{rr} V^*_{br} |a\rangle_A |b\rangle_B = \sum_r (\sum_a U_{ar} |a\rangle_A) S_{rr} (\sum_b V^*_{br} |b\rangle_B) =$$

$$= \sum_r S_{rr} |a'\rangle_A |b'\rangle_B = \sum_r s_r |a'\rangle_A |b'\rangle_B$$

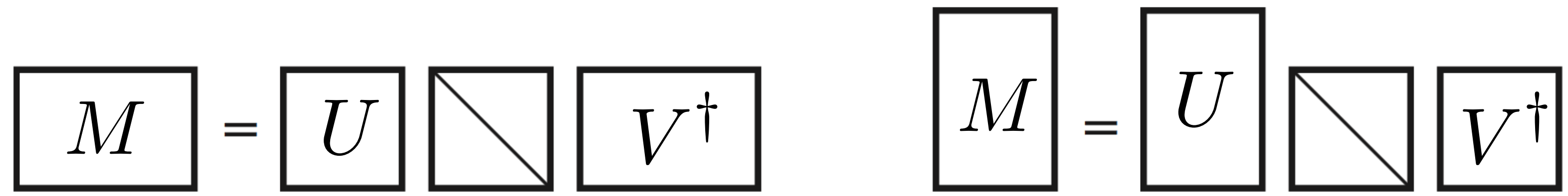
Given the orthonormality of the left and right singular vectors, we can define the bases a' and b' in the subsystems.

We observe **r**, the rank of S, is simply the Schmidt rank and so it represents the entanglement in the bipartition A-B.

Key decomposition towards efficient tensor networks:

- II) Singular Value Decomposition: allows for the decomposition of a generic matrix

$$M = USV^\dagger$$



We observe **r**, the rank of S, is simply the Schmidt rank and so it represents the entanglement in the bipartition A-B.

E.g., r=1, implies a product state.

We can also construct the subsystem reduced density operators from the partial traces as:

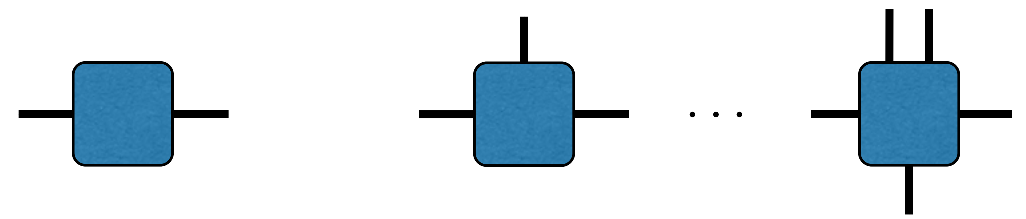
$$\boxed{\rho_A = \sum_r s_r^2 |a'\rangle_A \langle a'|_A, \quad \rho_B = \sum_r s_r^2 |b'\rangle_b \langle b'|_B}$$

We have access to the spectrum of these operators and it is given by the square of the singular values. Moreover, we can measure the entanglement properties:

$$\boxed{S_{vN} = -\mathrm{Tr}(\rho_{A/B} \log \rho_{A/B}) = -\sum_r s_r^2 \log s_r^2}$$

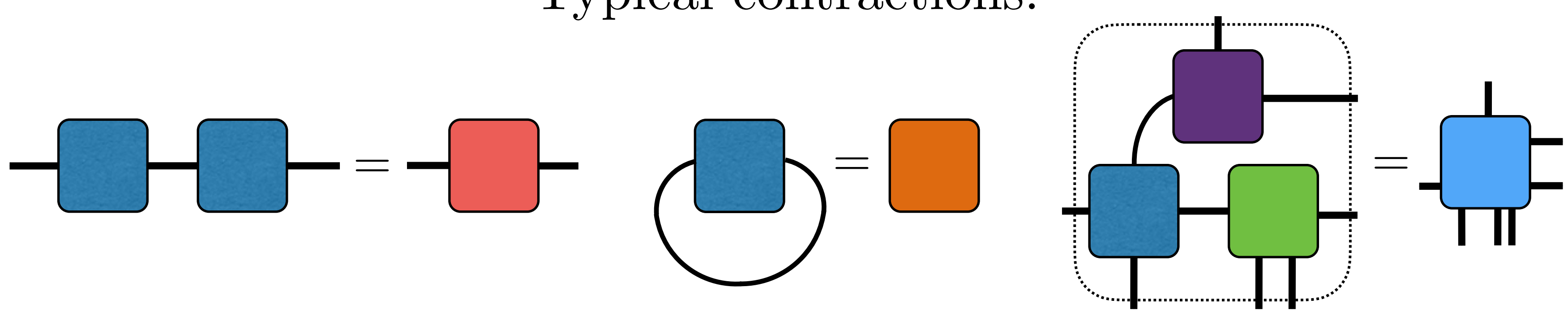
Graphical notation:

Before we start it is important to introduce this notation that simplifies enormously the contractions with tensor networks.



c (Scalar) v^σ (Vector) M_{ab} (Matrix) T^σ_{ab} (3-Tensor) T (N-Tensor)

Typical contractions:



Multiplication

$$M_{ac} = \sum_b A_{ab} B_{bc}$$

Trace

$$t = \sum_a M_{aa}$$

General contraction

Efficient representation: Matrix Product States (MPS)

- Consider a state in the complete Hilbert space

$$|\phi\rangle = \sum_{\sigma_1,\ldots,\sigma_M} c_{\sigma_1,\ldots,\sigma_M} |\sigma_1, \ldots, \sigma_M\rangle \qquad d^M \text{ coefficients}$$

Efficient representation: Matrix Product States (MPS)

- Consider a state in the complete Hilbert space

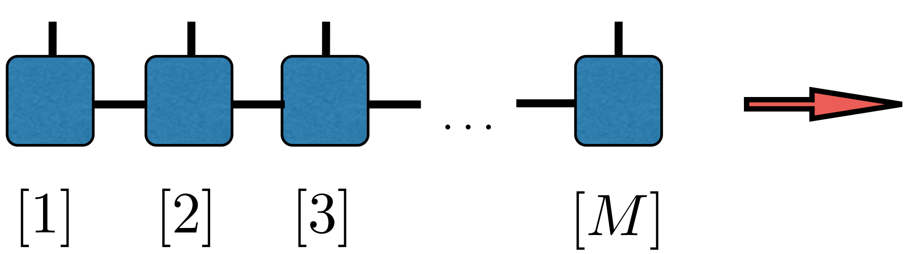
$$|\phi\rangle = \sum_{\sigma_1,\ldots,\sigma_M} c_{\sigma_1,\ldots,\sigma_M} |\sigma_1,\ldots,\sigma_M\rangle \qquad d^M \text{ coefficients}$$

- We can rearrange the coefficients as*

$$c_{\sigma_1,\ldots,\sigma_M} = \sum_{a_1\ldots a_{M-1}} A^{\sigma_1}_{a_1} A^{\sigma_2}_{a_1,a_2} \ldots A^{\sigma_{M-1}}_{a_{M-2},a_{M-1}} A^{\sigma_M}_{a_{M-1}}$$

$$|\phi\rangle = \sum_{\sigma_1,\ldots,\sigma_M} A^{\sigma_1} A^{\sigma_2} \ldots A^{\sigma_{M-1}} A^{\sigma_M} |\sigma_1,\ldots,\sigma_M\rangle$$

$|\phi\rangle =$



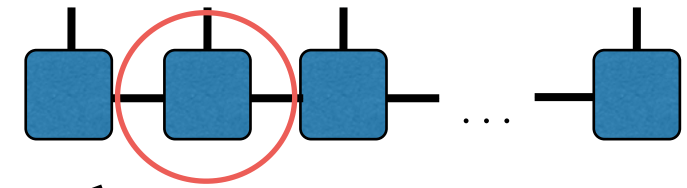
d^M coefficients as $d * M$ matrices

* We will see soon how to systematically perform this step.

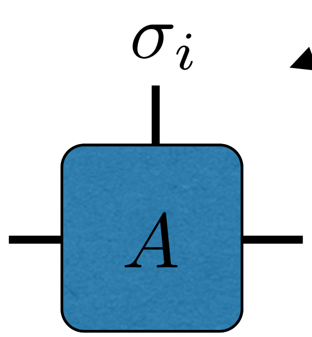
Matrix Product States (MPS): Truncation

- Consider a state

$$|\phi\rangle = \sum_{\sigma_1,\dots,\sigma_M} A^{\sigma_1} A^{\sigma_2} \dots A^{\sigma_{M-1}} A^{\sigma_M} \, |\sigma_1, \dots, \sigma_M\rangle$$



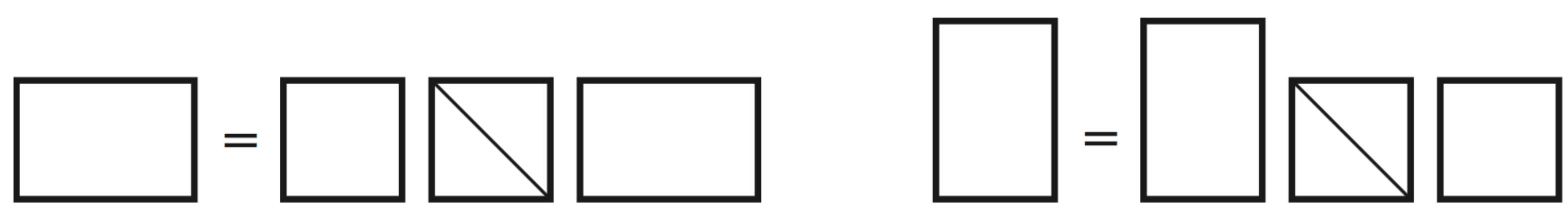
- What is the best possible approximation to this state with a fewer number of parameters?



Singular Value Decomposition

$$\boxed{A^{\sigma_i} = M = USV^\dagger}$$

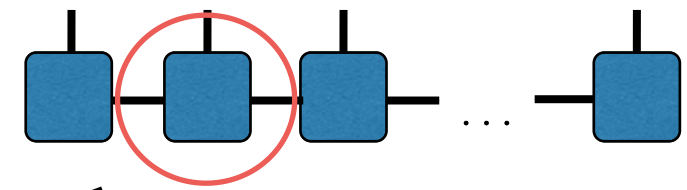
$$S = diag(s_1, s_2, \dots, s_r, 0, \dots, 0)$$



Matrix Product States (MPS): Truncation

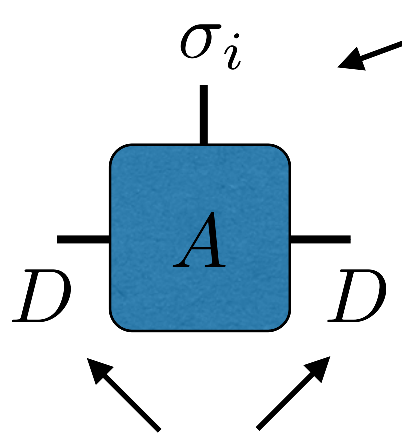
- Consider a state

$$|\phi\rangle = \sum_{\sigma_1,\ldots,\sigma_M} A^{\sigma_1} A^{\sigma_2} \ldots A^{\sigma_{M-1}} A^{\sigma_M} \, |\sigma_1, \ldots, \sigma_M\rangle$$



- What is the best possible approximation to this state with a fewer number of parameters?

Physical Dim.

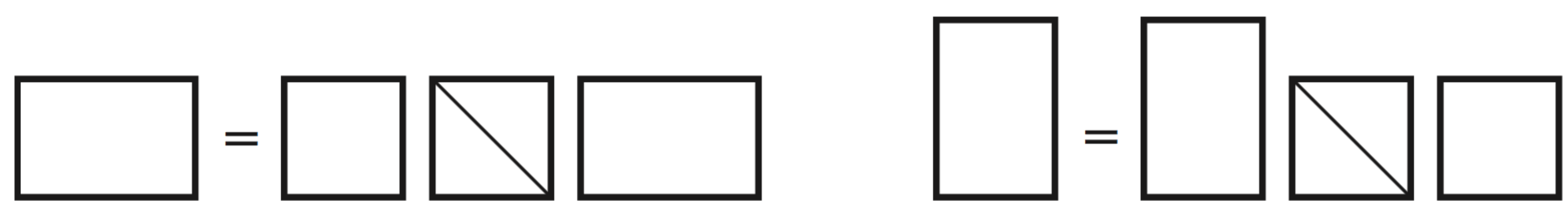


Bond Dimension

Singular Value Decomposition

$$\boxed{A^{\sigma_i} = M = USV^\dagger}$$

$$S = diag(s_1, s_2, \ldots, s_r, 0, \ldots, 0)$$

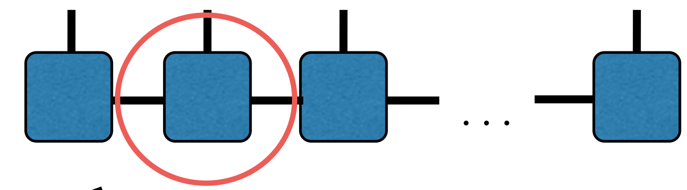


- Better approximation to M with rank D is the one obtained by taking the D greater singular values

Matrix Product States (MPS): Truncation

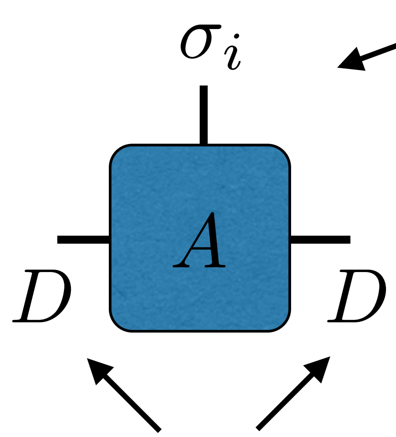
- Consider a state

$$|\phi\rangle = \sum_{\sigma_1,\dots,\sigma_M} A^{\sigma_1} A^{\sigma_2} \dots A^{\sigma_{M-1}} A^{\sigma_M} \, |\sigma_1, \dots, \sigma_M\rangle$$



- What is the best possible approximation to this state with a fewer number of parameters?

Physical Dim.

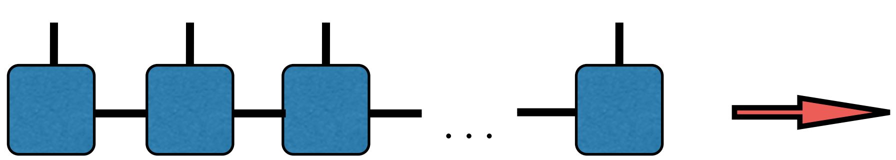


Bond Dimension

Singular Value Decomposition

$$\boxed{A^{\sigma_i} = M = USV^\dagger}$$

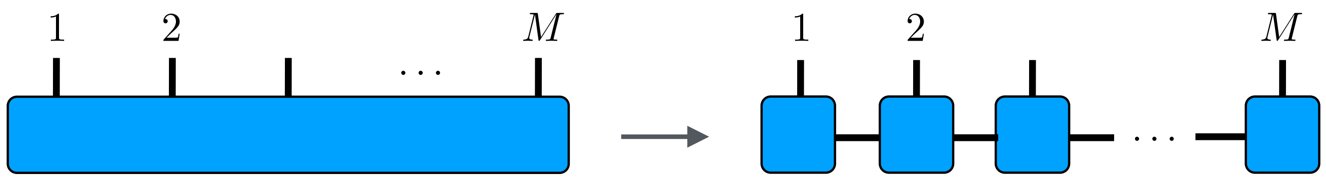
$$S = diag(s_1, s_2, \dots, s_r, 0, \dots, 0)$$



MdD^2 coefficients

$$\boxed{poly(M) \text{ vs } d^M}$$

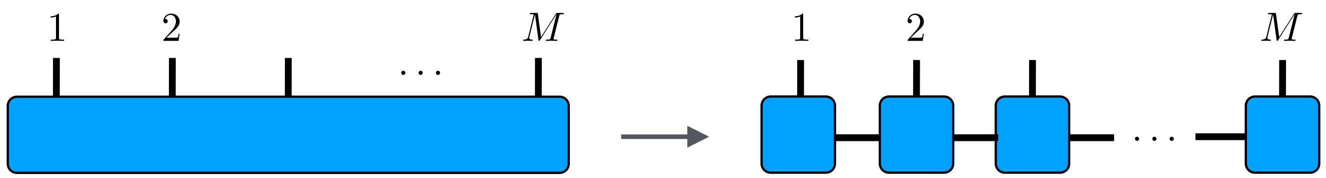
How to systematically represent a state in MPS form:



Let us reshape the multidimensional tensor of coefficients *c* into a matrix with rows corresponding to the first local Hilbert space:

$$c_{\sigma_1,\ldots,\sigma_M} = M_{\sigma_1,(\sigma_2,\ldots,\sigma_M)}$$

How to systematically represent a state in MPS form:

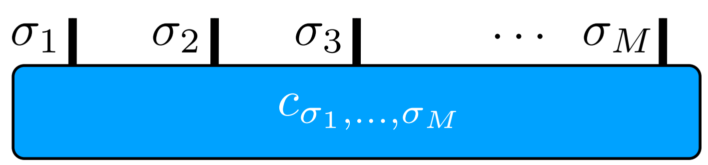


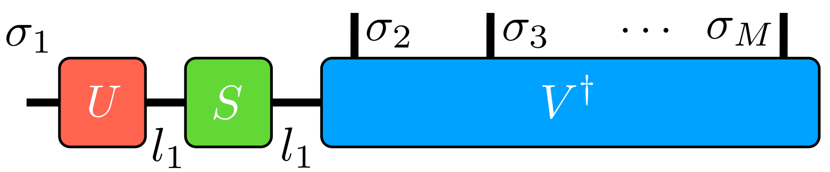
Let us reshape the multidimensional tensor of coefficients *c* into a matrix with rows corresponding to the first local Hilbert space:

$$c_{\sigma_1,\ldots,\sigma_M} = M_{\sigma_1,(\sigma_2,\ldots,\sigma_M)}$$

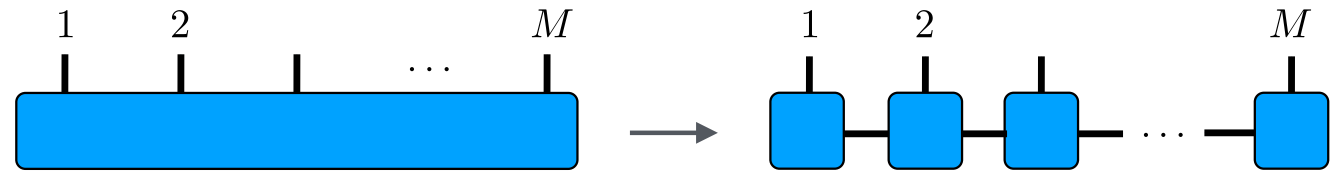
We can perform SVD in this matrix to obtain:

$$M_{\sigma_1,(\sigma_2\ldots,\sigma_M)} = \sum_{l_1}^{r_1} U_{\sigma_1,l_1} S_{l_1,l_1} V^{\dagger}_{l_1,(\sigma_2\ldots,\sigma_M)}$$





How to systematically represent a state in MPS form:

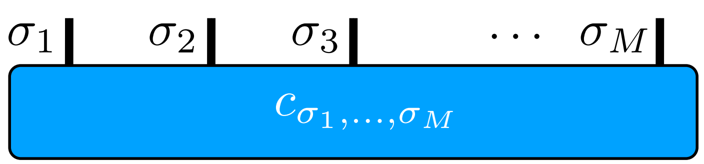


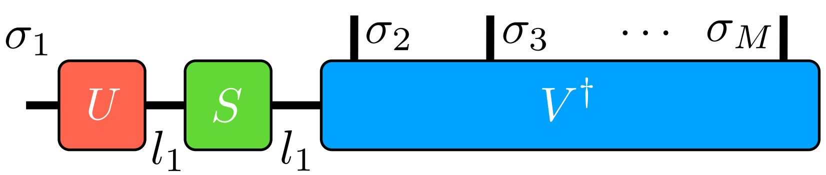
Let us reshape the multidimensional tensor of coefficients *c* into a matrix with rows corresponding to the first local Hilbert space:

$$c_{\sigma_1,\ldots,\sigma_M} = M_{\sigma_1,(\sigma_2,\ldots,\sigma_M)}$$

We can perform SVD in this matrix to obtain:

$$M_{\sigma_1,(\sigma_2\ldots,\sigma_M)} = \sum_{l_1}^{r_1} U_{\sigma_1,l_1} S_{l_1,l_1} V^{\dagger}_{l_1,(\sigma_2\ldots,\sigma_M)}$$



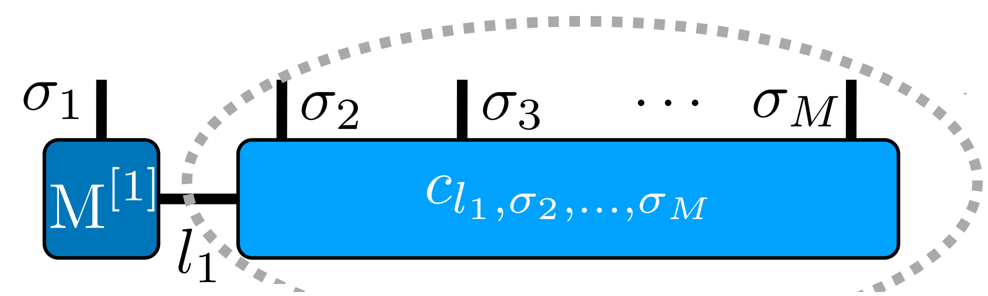


In order to recover MPS form, we multiply *S* and *V* and reshape our matrix *U* into a set of row vectors:

$$U_{\sigma_1,l_1} = M^{[1]\,\sigma_1}_{l_1}$$

And we obtain:

$$c_{\sigma_1,\ldots,\sigma_M} = \sum_{l_1}^{r_1} M^{[1]\,\sigma_1}_{l_1} c_{l_1,(\sigma_2\ldots,\sigma_M)}$$



We repeat!

How to systematically represent a state in MPS form:

And we obtain:

$$c_{\sigma_1,\ldots,\sigma_M} = \sum_{l_1}^{r_1} M^{[1]\,\sigma_1}_{l_1} c_{l_1,(\sigma_2\ldots,\sigma_M)}$$

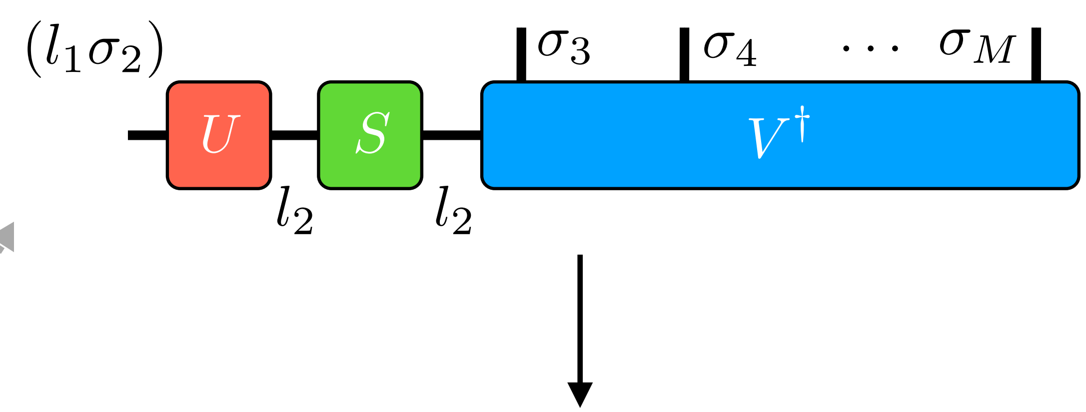
Then, we repeat the protocol incorporating the next physical dimension.

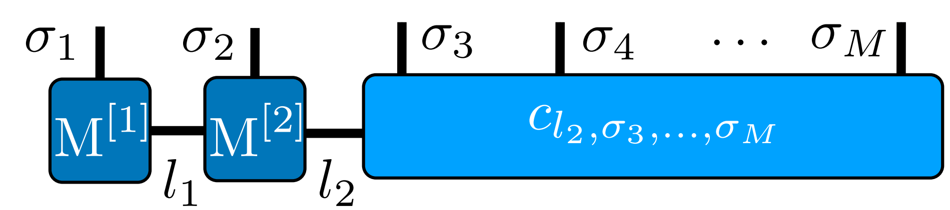
$$c_{l_1,(\sigma_2\ldots,\sigma_M)} = M_{(l_1\sigma_2),(\sigma_3,\ldots,\sigma_M)} = U_{(l_1\sigma_2),l_2} S_{l_2,l_2} V^\dagger_{l_2,(\sigma_3,\ldots,\sigma_M)}$$

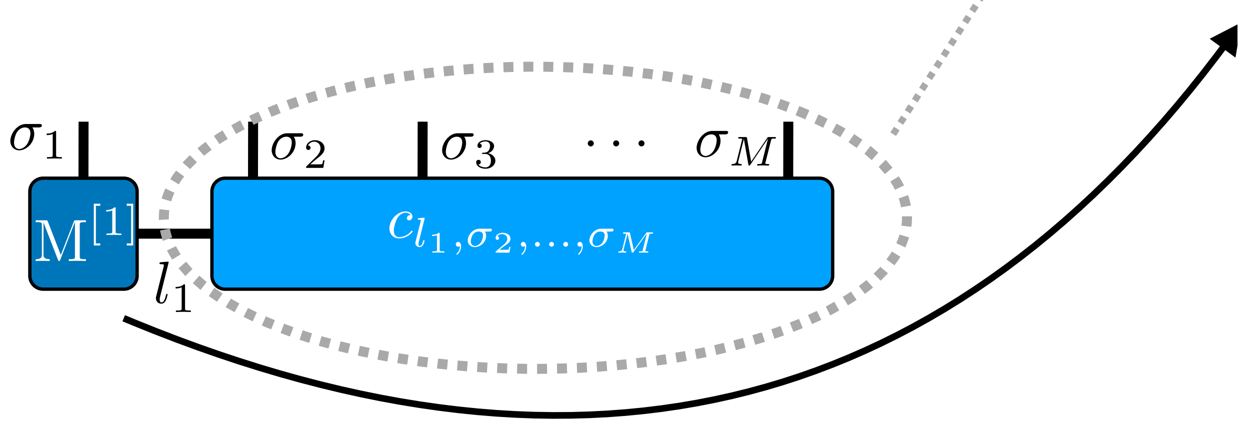
And again, reshape:

$$U_{(l_1\sigma_2),l_2} = M^{[2]\,\sigma_2}_{l_1,l_2}$$

$$S_{l_2,l_2} V^\dagger_{l_2,(\sigma_3,\ldots,\sigma_M)} = c_{l_2,(\sigma_3,\ldots,\sigma_M)}$$



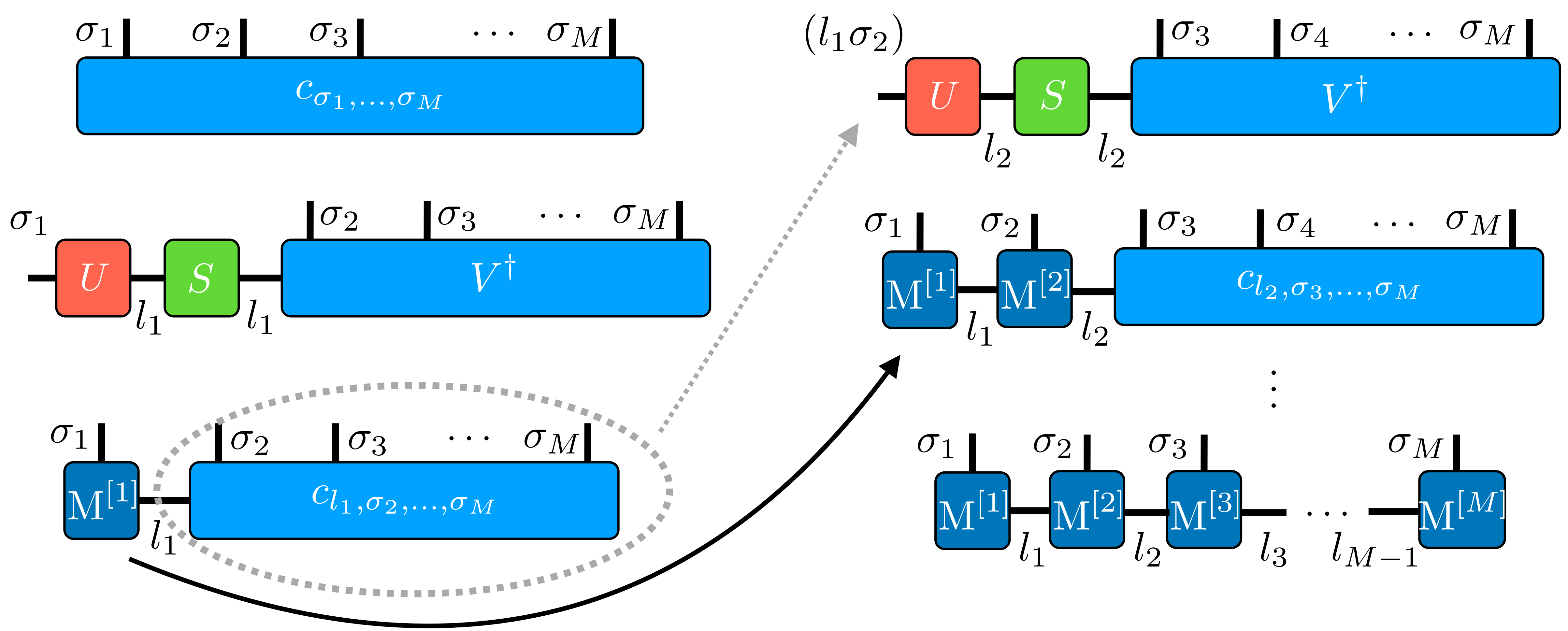




Reshape *U* into *M*, and multiply *S* and *V* to form *c*.

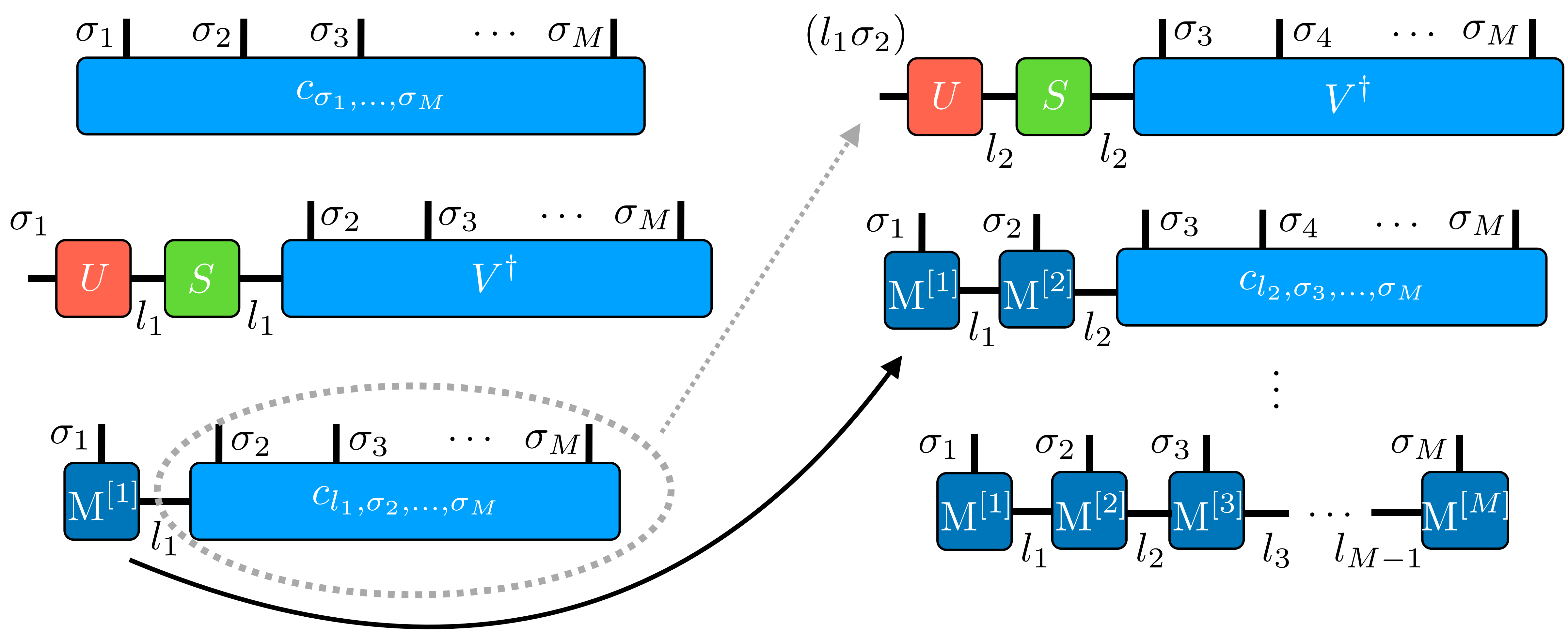
How to systematically represent a state in MPS form:

Repeating for all our sites we construct an MPS.



How to systematically represent a state in MPS form:

Repeating for all our sites we construct an MPS.



Important observations:

- The bond dimensions will start to reduce when we reach the middle of the chain and in the last iteration we will be left with a set of row vectors again.

- This representation is not unique: gauge freedom.

Examples of MPS:

Let us consider a spin 1/2 system with 2 particles:

Examples of MPS:

Let us consider a spin 1/2 system with 2 particles:

- 50/50 superposition product state:

$$|\phi_{\text{Product}}\rangle = \frac{|0\rangle + |1\rangle}{\sqrt(2)} \otimes \frac{|0\rangle + |1\rangle}{\sqrt(2)} \longrightarrow \forall c_{i,j} = 1/\sqrt{2}$$

Examples of MPS:

Let us consider a spin 1/2 system with 2 particles:

- 50/50 superposition product state:

$$|\phi_{\text{Product}}\rangle = \frac{|0\rangle + |1\rangle}{\sqrt(2)} \otimes \frac{|0\rangle + |1\rangle}{\sqrt(2)} \longrightarrow \forall c_{i,j} = 1/\sqrt{2}$$

In two sites matrix will have a max. dimension d*(1,d) and d*(d,1). However, we have a product state. Then, we know that the bond dimension =1.

$$M^{[1]0} = M^{[1]1} = M^{[2]0} = M^{[2]1} = 1/\sqrt{2}$$

Examples of MPS:

Let us consider a spin 1/2 system with 2 particles:

- 50/50 superposition product state:

$$|\phi_{\mathrm{Product}}\rangle = \frac{|0\rangle + |1\rangle}{\sqrt(2)} \otimes \frac{|0\rangle + |1\rangle}{\sqrt(2)} \longrightarrow \forall c_{i,j} = 1/\sqrt{2}$$

In two sites matrix will have a max. dimension d*(1,d) and d*(d,1). However, we have a product state. Then, we know that the bond dimension =1.

$$M^{[1]0} = M^{[1]1} = M^{[2]0} = M^{[2]1} = 1/\sqrt{2}$$

- Bell state:

$$|\phi_{\mathrm{Bell},+}\rangle = (|0_1\rangle |0_2\rangle + |1_1\rangle |1_2\rangle)/2 \longrightarrow c_{0,0} = c_{1,1} = 1/\sqrt{2}, c_{1,0} = c_{0,1} = 0$$

Examples of MPS:

Let us consider a spin 1/2 system with 2 particles:

- 50/50 superposition product state:

$$|\phi_{\text{Product}}\rangle = \frac{|0\rangle + |1\rangle}{\sqrt(2)} \otimes \frac{|0\rangle + |1\rangle}{\sqrt(2)} \longrightarrow \forall c_{i,j} = 1/\sqrt{2}$$

In two sites matrix will have a max. dimension d*(1,d) and d*(d,1). However, we have a product state. Then, we know that the bond dimension =1.

$$M^{[1]0} = M^{[1]1} = M^{[2]0} = M^{[2]1} = 1/\sqrt{2}$$

- Bell state:

$$|\phi_{\text{Bell},+}\rangle = (|0_1\rangle\,|0_2\rangle + |1_1\rangle\,|1_2\rangle)/2 \longrightarrow c_{0,0} = c_{1,1} = 1/\sqrt{2}, c_{1,0} = c_{0,1} = 0$$

$$A^{[1]0} = \begin{pmatrix} 1 & 0 \end{pmatrix},\ A^{[1]1} = \begin{pmatrix} 0 & 1 \end{pmatrix}$$

$$A^{[2]0} = \begin{pmatrix} 1/\sqrt{2} \\ 0 \end{pmatrix},\ = A^{[2]1} = \begin{pmatrix} 0 \\ 1/\sqrt{2} \end{pmatrix}$$

Position of the normalizing constant depends on direction on which we construct (gauge freedom!)

Examples of MPS:

Let us consider a spin 1/2 system with 2 particles:

- 50/50 superposition product state:

$$|\phi_{\text{Product}}\rangle = \frac{|0\rangle + |1\rangle}{\sqrt(2)} \otimes \frac{|0\rangle + |1\rangle}{\sqrt(2)} \longrightarrow \forall c_{i,j} = 1/\sqrt{2}$$

In two sites matrix will have a max. dimension d*(1,d) and d*(d,1). However, we have a product state. Then, we know that the bond dimension =1.

$$M^{[1]0} = M^{[1]1} = M^{[2]0} = M^{[2]1} = 1/\sqrt{2}$$

- Bell state:

$$|\phi_{\text{Bell},+}\rangle = (|0_1\rangle |0_2\rangle + |1_1\rangle |1_2\rangle) / 2 \longrightarrow c_{0,0} = c_{1,1} = 1/\sqrt{2}, c_{1,0} = c_{0,1} = 0$$

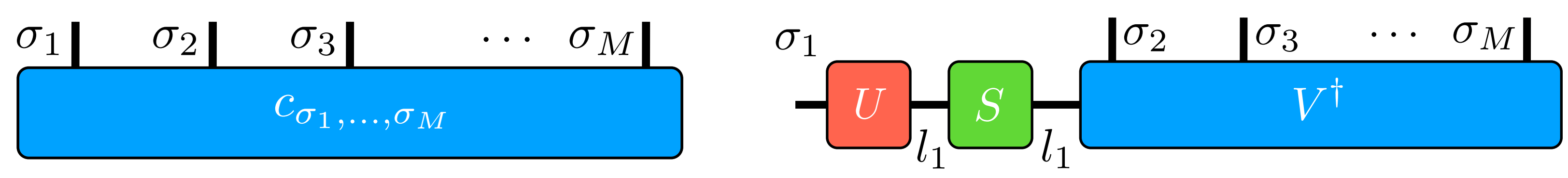
$$A^{[1]0} = \begin{pmatrix} 1 & 0 \end{pmatrix}, \ A^{[1]1} = \begin{pmatrix} 0 & 1 \end{pmatrix}$$

$$A^{[2]0} = \begin{pmatrix} 1/\sqrt{2} \\ 0 \end{pmatrix}, = A^{[2]1} = \begin{pmatrix} 0 \\ 1/\sqrt{2} \end{pmatrix}$$

Position of the normalizing constant depends on direction on which we construct (gauge freedom!)

Homework: show that $c_{\sigma_1,\sigma_2} = \sum_j A^{[1]\sigma_1}_{1j} A^{[2]\sigma_2}_{j1}$.

How does it look when coding the routines?




```
d=size(mps.Gamma{datind},1);
DL=size(mps.Gamma{datind},2);
DR=size(mps.Gamma{datind},3);

dR=size(mps.Gamma{datind+1},1);
DRL=size(mps.Gamma{datind+1},2);
DRR=size(mps.Gamma{datind+1},3);

[Q R]=qr(reshape(mps.Gamma{datind},d*DL,DR),0);
schmidt=svd(R);

mps.Gamma{datind}=reshape(Q,d,DL,DR);

tmp=R*reshape(permute(mps.Gamma{datind+1},[2 1 3]),DRL,dR*DRR);
mps.Gamma{datind+1}=permute(reshape(tmp,DRL,dR,DRR),[2 1 3]);
```

Extract sizes of the tensors.

QR decomposition

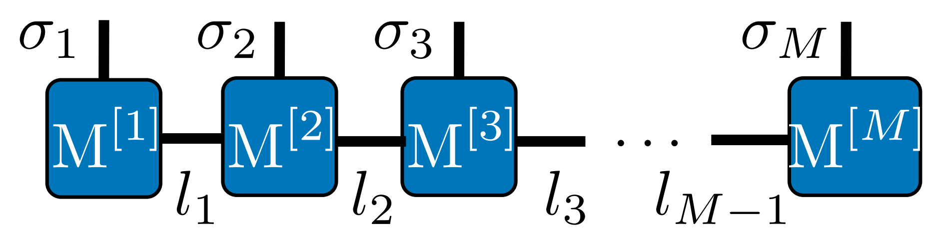
Save Schmidt values

Reshape U/Q into the 3-legged tensor

Incorporate S*V/R to the right side and reshape into the corresponding size.

How to systematically represent a state in MPS form:

- This representation is not unique: gauge freedom.



- The previous state can be represented equally by the following local matrices, with *T* any invertible matrix.

$$M^{[2]} \to M^{[2]} \cdot T, \;\; M^{[3]} \to T^{-1} \cdot M^{[3]}$$

- The form we just saw before, is denoted as **left canonical** since, the individual matrices, coming from the left singular vector, fulfill:

$$\sum_{\sigma_i} \left(M^{[i]\,\sigma_i} \right)^{\dagger} M^{[i]\,\sigma_i} = I$$

- If we start the procedure from the right-hand side, we denote it as **right canonical** since, the individual matrices, coming from the left singular vector, fulfill:

$$\sum_{\sigma_i} M^{[i]\,\sigma_i} \left(M^{[i]\,\sigma_i} \right)^{\dagger} = I$$

- We can also consider, mixtures of this two procedures.

Entanglement and truncation error

- Let us consider now the implications that this state representation ansatz has when we truncate our maximum bond dimension during every SVD.

- We recall, the Schmidt decomposition:

$$|\phi\rangle = \sum_{i,j} \lambda_{ab} |i\rangle_A |j\rangle_B$$

- With this we can define our reduce density operator

$$\rho_{1,\dots,i-1} = \mathrm{Tr}_{i+1,\dots,M}\left(|\phi\rangle\langle\phi|\right) = \sum_{l_i}^{r_i} (\lambda_i)^2 \, |i\rangle_A \langle i|_A$$

- Then, if we fix our bond to a maximum value D_{max} this implies an error:

$$\boxed{\epsilon_i \equiv \sum_{l_i = D_{max}+1}^{r_i} (\lambda_i)^2}$$

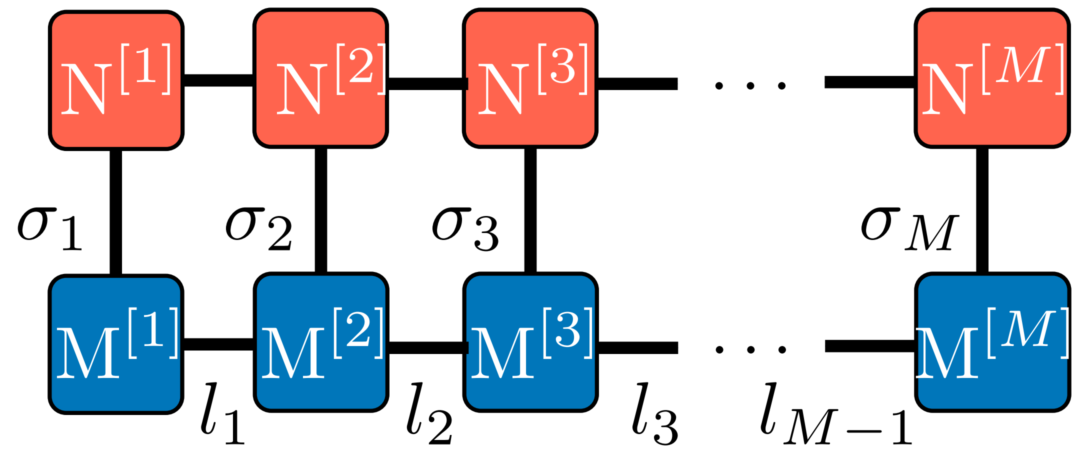
- Also, implying a limit in the entanglement that we can represent:

$$S_{vN} = -\mathrm{Tr}\left(\rho_{1,\dots,i-1} \ln \rho_{1,\dots,i-1}\right) = -\sum_{l_i}^{r_i} (\lambda_i)^2 \ln (\lambda_i)^2 \;\rightarrow\; \boxed{S_{vN}^{max} \leq \ln (D_{max})}$$

Common routines with MPS

- Computing overlaps:

$\langle \psi | \phi \rangle =$

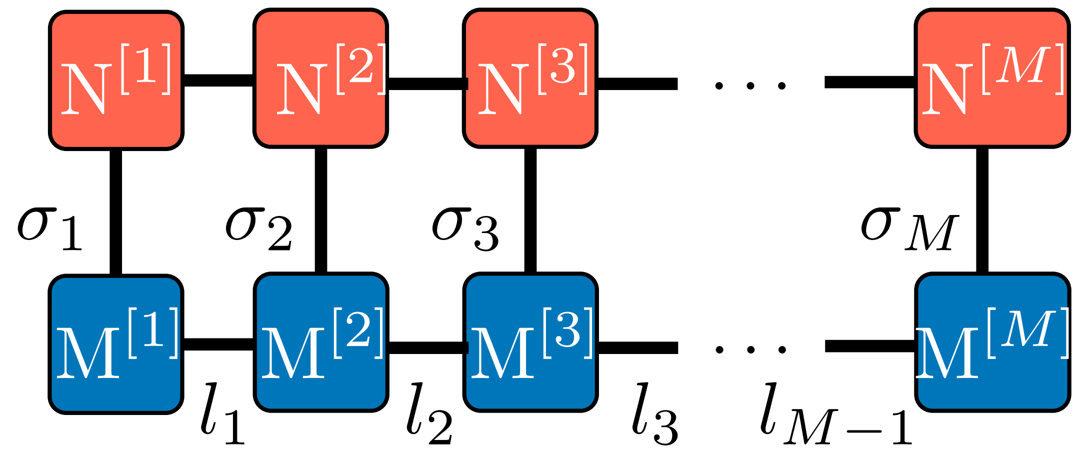


$$\langle \psi | \phi \rangle = \sum_{\sigma_M} \left(N^{[M]\,\sigma_M} \right)^{\dagger} \left(\dots \left(\sum_{\sigma_1} \left(N^{[1]\,\sigma_1} \right)^{\dagger} M^{[1]\,\sigma_1} \right) \dots \right) M^{[M]\,\sigma_M}$$

Common routines with MPS

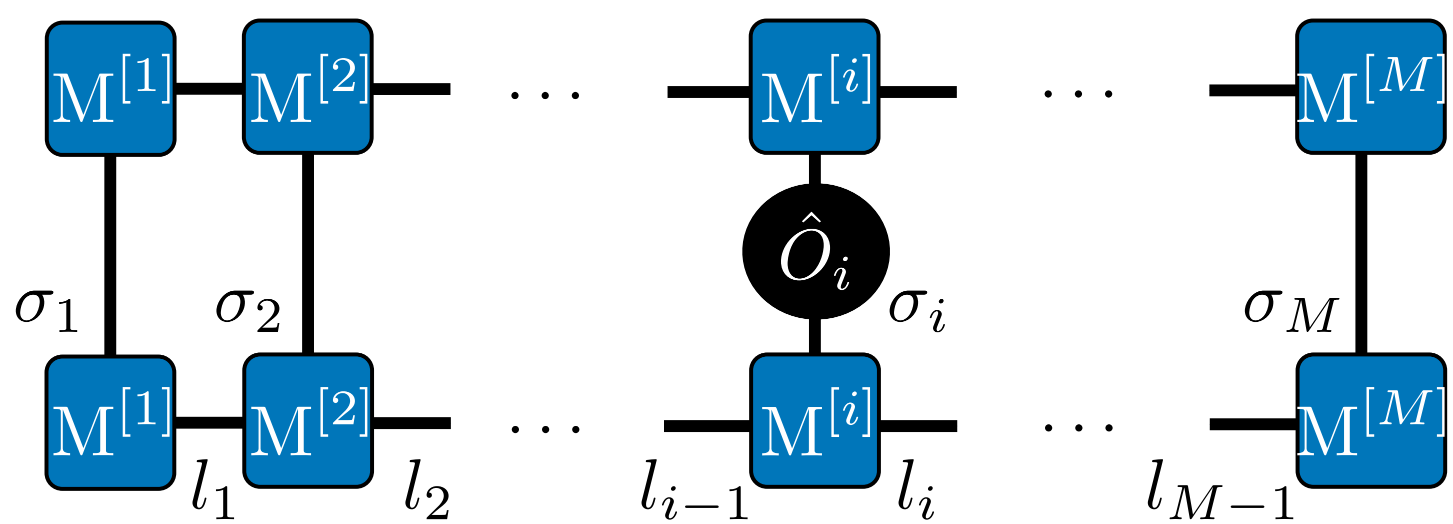
- Computing overlaps:

$\langle \psi | \phi \rangle =$



- Computing local expectation values:

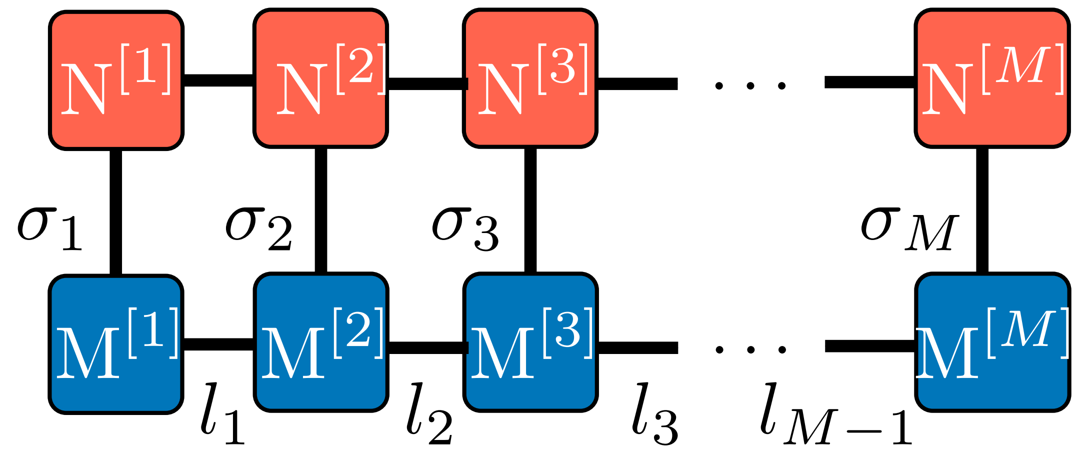
$\langle \phi | \hat{O}_i | \phi \rangle =$



Common routines with MPS

- Computing overlaps:

$\langle \psi | \phi \rangle =$

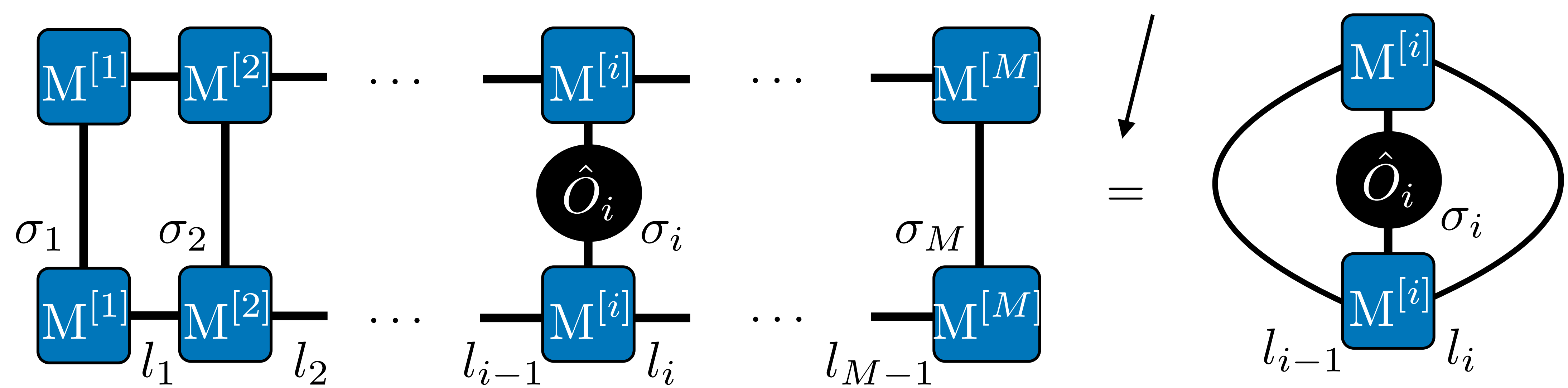


$$\sum_{\sigma_i} \left(M^{[i]\,\sigma_i}\right)^\dagger M^{[i]\,\sigma_i} = I$$
$$\sum_{\sigma_i} M^{[i]\,\sigma_i} \left(M^{[i]\,\sigma_i}\right)^\dagger = I$$

- Computing local expectation values:

Using mixed canonical form up to site i

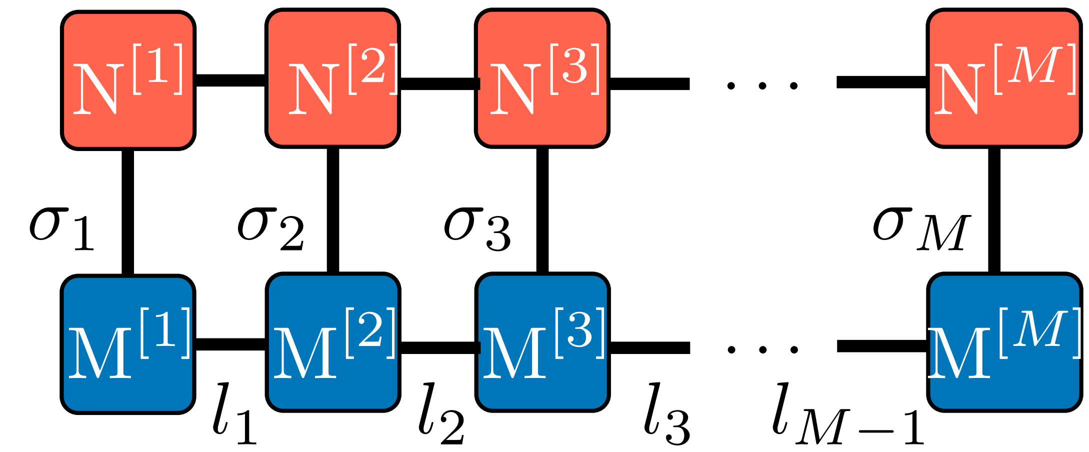
$\langle \phi | \hat{O}_i | \phi \rangle =$



Common routines with MPS

- Computing overlaps:

$\langle \psi | \phi \rangle =$

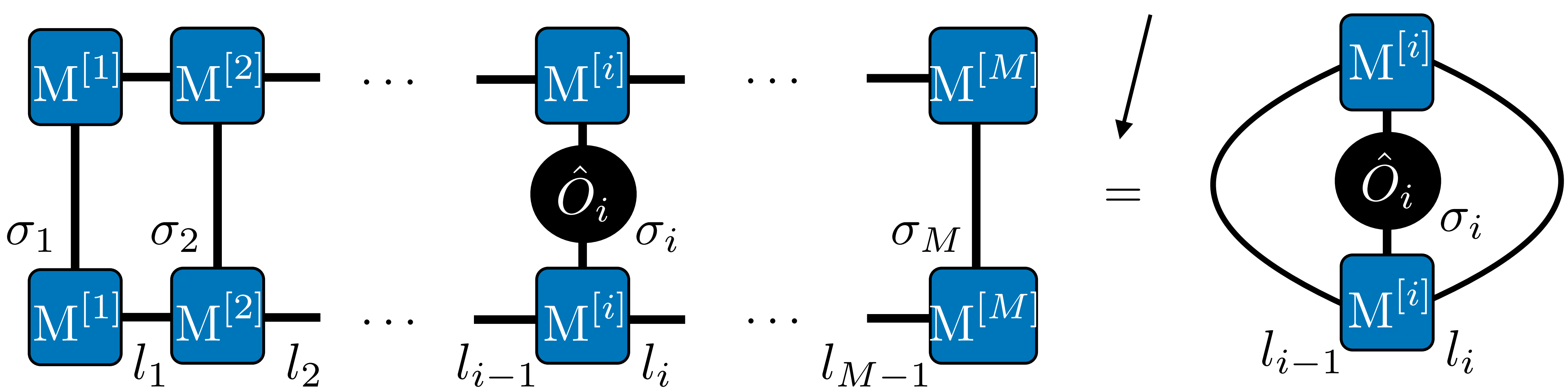


$$\sum_{\sigma_i} \left(M^{[i]\,\sigma_i}\right)^{\dagger} M^{[i]\,\sigma_i} = I$$
$$\sum_{\sigma_i} M^{[i]\,\sigma_i} \left(M^{[i]\,\sigma_i}\right)^{\dagger} = I$$

- Computing local expectation values:

Using mixed canonical form up to site i

$\langle \phi | \hat{O}_i | \phi \rangle =$

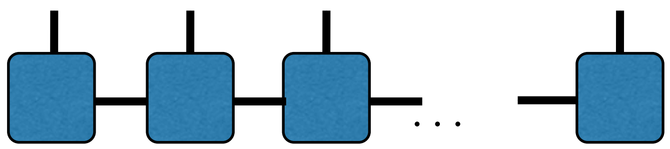


- How do we deal with non local operators? $\Rightarrow$ Matrix Product Operator (MPO)

Matrix Product Operators (MPO)

- We can extract it from the notion of MPS

$$|\phi\rangle = \sum_{\sigma_1,\ldots,\sigma_M} A^{\sigma_1}A^{\sigma_2}\ldots A^{\sigma_{M-1}}A^{\sigma_M}\,|\sigma_1,\ldots,\sigma_M\rangle$$



with coefficients $\langle\vec{\sigma}\,|\phi\rangle = A^{\sigma_1}A^{\sigma_2}\ldots A^{\sigma_{M-1}}A^{\sigma_M}$

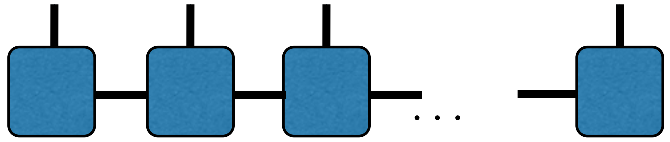
Pirvu, Murg, Cirac, Verstraete, NJP 12 (2010) 025012

Verstraete, García-Ripoll, Cirac, PRL 93, 20 (2004)

Matrix Product Operators (MPO)

- We can extract it from the notion of MPS

$$|\phi\rangle = \sum_{\sigma_1,\ldots,\sigma_M} A^{\sigma_1} A^{\sigma_2} \ldots A^{\sigma_{M-1}} A^{\sigma_M} \,|\sigma_1, \ldots, \sigma_M\rangle$$



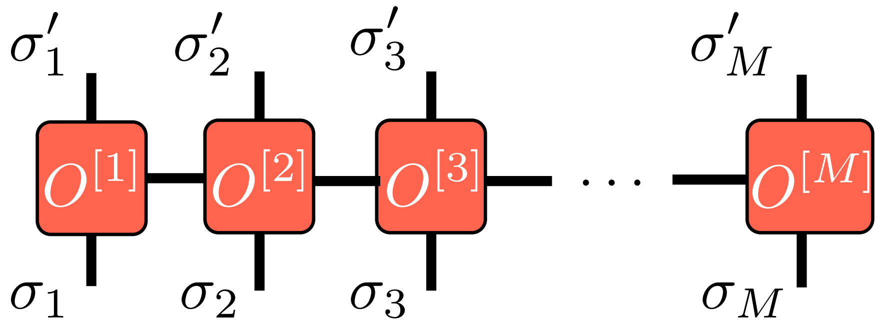
with coefficients $\langle \vec{\sigma}\,|\phi\rangle = A^{\sigma_1} A^{\sigma_2} \ldots A^{\sigma_{M-1}} A^{\sigma_M}$

- An operator will have the coefficients given by

$$\langle \vec{\sigma}|\hat{O}|\vec{\sigma}'\rangle = W^{\sigma_1 \sigma'_1} W^{\sigma_2 \sigma'_2} \ldots W^{\sigma_{M-1} \sigma'_{M-1}} W^{\sigma_M \sigma'_M}$$

$$\downarrow$$

$$\hat{O} = \sum_{\vec{\sigma},\,\vec{\sigma}'} W^{\sigma_1 \sigma'_1} W^{\sigma_2 \sigma'_2} \ldots W^{\sigma_{M-1} \sigma'_{M-1}} W^{\sigma_M \sigma'_M} \,|\vec{\sigma}\rangle\langle\vec{\sigma}'|$$



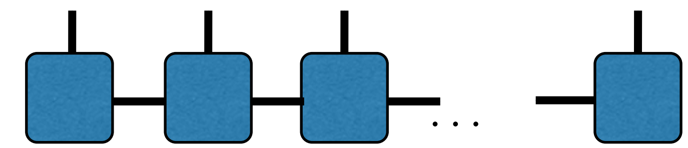
Pirvu, Murg, Cirac, Verstraete, NJP 12 (2010) 025012

Verstraete, García-Ripoll, Cirac, PRL 93, 20 (2004)

Matrix Product Operators (MPO)

- We can extract it from the notion of MPS

$$|\phi\rangle = \sum_{\sigma_1,\ldots,\sigma_M} A^{\sigma_1} A^{\sigma_2} \ldots A^{\sigma_{M-1}} A^{\sigma_M} \, |\sigma_1, \ldots, \sigma_M\rangle$$



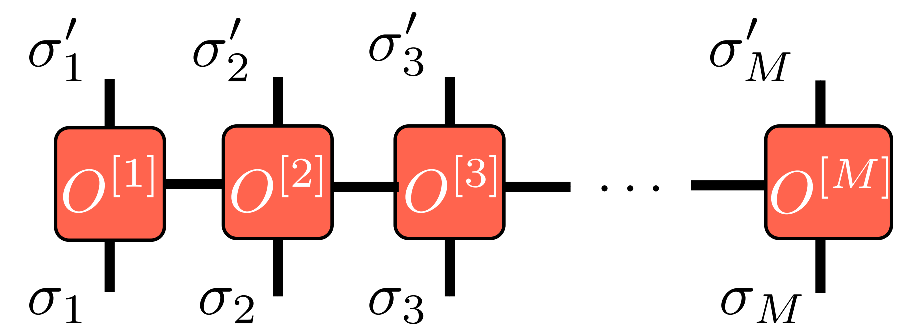
with coefficients $\langle \vec{\sigma}\,|\phi\rangle = A^{\sigma_1} A^{\sigma_2} \ldots A^{\sigma_{M-1}} A^{\sigma_M}$

- An operator will have the coefficients given by

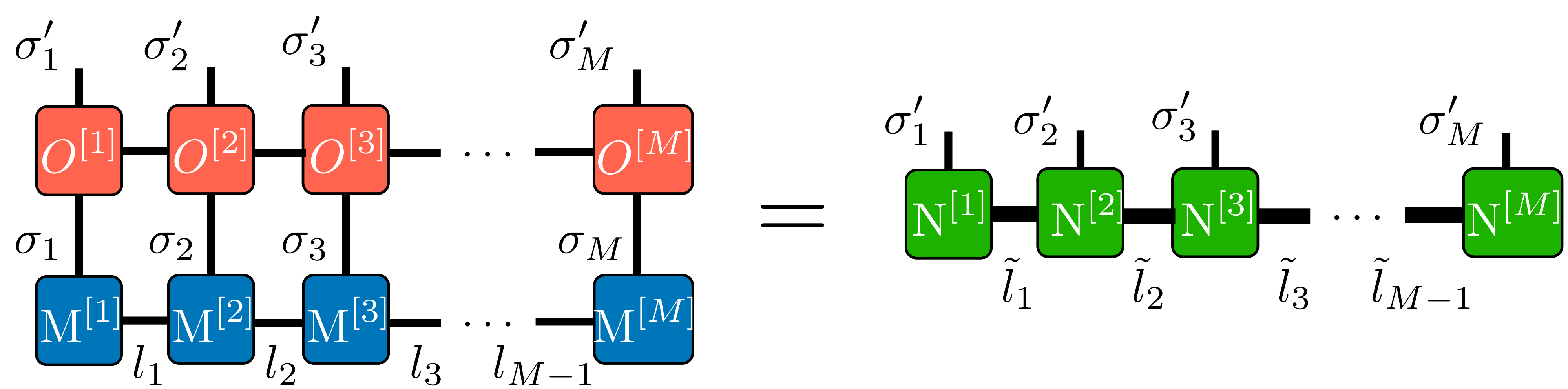
$$\langle \vec{\sigma}|\hat{O}|\vec{\sigma}'\rangle = W^{\sigma_1\sigma_1'} W^{\sigma_2\sigma_2'} \ldots W^{\sigma_{M-1}\sigma_{M-1}'} W^{\sigma_M\sigma_M'}$$

$$\downarrow$$

$$\hat{O} = \sum_{\vec{\sigma},\,\vec{\sigma}'} W^{\sigma_1\sigma_1'} W^{\sigma_2\sigma_2'} \ldots W^{\sigma_{M-1}\sigma_{M-1}'} W^{\sigma_M\sigma_M'} \, |\vec{\sigma}\rangle\langle\vec{\sigma}'|$$



$$\hat{O}|\phi\rangle = |\psi\rangle$$



Pirvu, Murg, Cirac, Verstraete, NJP 12 (2010) 025012

Verstraete, García-Ripoll, Cirac, PRL 93, 20 (2004)

Example of MPOs and how to construct them

PAPER • OPEN ACCESS

Out-of-equilibrium dynamics with matrix product states

M L Wall[3,1] and Lincoln D Carr[1,2]

Published 18 December 2012 • © IOP Publishing and Deutsche Physikalische Gesellschaft

New Journal of Physics, Volume 14, December 2012

As a concrete example, consider the Ising model

$$\hat{H} = -J \sum_{\langle i,j \rangle} \hat{\sigma}_i^z \hat{\sigma}_j^z - h \sum_i \hat{\sigma}_i^x \, . \tag{25}$$

The Hamiltonian consists of two rules. The first is a site rule $\mathcal{R}_{\mathrm{site}}(\hat{\sigma}^x, h, -1)$ which generates the string $-h \sum_i \hat{\sigma}_i^x$. The three matrices which provide this rule are

$$\mathcal{W}_{\mathrm{site}}^{[1]} = \begin{pmatrix} -h\hat{\sigma}^x & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{site}}^{[2 \le j \le L-1]} = \begin{pmatrix} \hat{I} & 0 \\ -h\hat{\sigma}^x & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{site}}^{[L]} = \begin{pmatrix} \hat{I} \\ -h\hat{\sigma}^x \end{pmatrix} . \tag{26}$$

As can be verified,

$$\prod_{j=L-1}^{k} \mathcal{W}_{\mathrm{site}}^{[j]} \mathcal{W}_{\mathrm{site}}^{[L]} = \begin{pmatrix} \hat{I} \dots \hat{I} \\ -h \sum_{i=k}^{L} \hat{\sigma}_i^x \end{pmatrix} , \tag{27}$$

and so this rule produces the desired operator.

Example of MPOs and how to construct them

PAPER • OPEN ACCESS

Out-of-equilibrium dynamics with matrix product states

M L Wall[3,1] and Lincoln D Carr[1,2]

Published 18 December 2012 • © IOP Publishing and Deutsche Physikalische Gesellschaft
New Journal of Physics, Volume 14, December 2012

As a concrete example, consider the Ising model

$$\hat{H} = -J \sum_{\langle i,j \rangle} \hat{\sigma}_i^z \hat{\sigma}_j^z - h \sum_i \hat{\sigma}_i^x \,. \tag{25}$$

Similarly, there is a bond rule $\mathcal{R}_{\mathrm{bond}}(\{\hat{\sigma^z}, \hat{\sigma}^z\}, J, -1)$ given by

$$\mathcal{W}_{\mathrm{bond}}^{[1]} = \begin{pmatrix} 0 & -J\hat{\sigma}^z & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{bond}}^{[2\le j\le L-1]} = \begin{pmatrix} \hat{I} & 0 & 0 \\ \hat{\sigma}^z & 0 & 0 \\ 0 & -J\hat{\sigma}^z & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{bond}}^{[L]} = \begin{pmatrix} \hat{I} \\ \hat{\sigma}^z \\ 0 \end{pmatrix}, \tag{28}$$

which produces $-J\sum_{\langle i,j\rangle} \hat{\sigma}_i^z \hat{\sigma}_j^z$, with $\langle i, j\rangle$ denoting a sum over nearest neighbors i and j. The full Hamiltonian is given by the direct sum of the matrices. Collecting rows of the direct sum which are exactly parallel, we have the MPO representation of the full operator

$$\mathcal{W}_{\mathrm{Ising}}^{[1]} = \begin{pmatrix} -h\hat{\sigma}^x & -J\hat{\sigma}^z & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{Ising}}^{[2\le j\le L-1]} = \begin{pmatrix} \hat{I} & 0 & 0 \\ \hat{\sigma}^z & 0 & 0 \\ -h\hat{\sigma}^x & -J\hat{\sigma}^z & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{Ising}}^{[L]} = \begin{pmatrix} \hat{I} \\ \hat{\sigma}^z \\ -h\hat{\sigma}^x \end{pmatrix}. \tag{29}$$

Example of MPOs and how to construct them

PAPER • OPEN ACCESS

Out-of-equilibrium dynamics with matrix product states

M L Wall[3,1] and Lincoln D Carr[1,2]

Published 18 December 2012 • © IOP Publishing and Deutsche Physikalische Gesellschaft

New Journal of Physics, Volume 14, December 2012

As a concrete example, consider the Ising model

$$\hat{H} = -J \sum_{\langle i,j \rangle} \hat{\sigma}_i^z \hat{\sigma}_j^z - h \sum_i \hat{\sigma}_i^x \,. \tag{25}$$

Similarly, there is a bond rule $\mathcal{R}_{\mathrm{bond}}(\{\hat{\sigma^z}, \hat{\sigma}^z\}, J, -1)$ given by

$$\mathcal{W}_{\mathrm{bond}}^{[1]} = \begin{pmatrix} 0 & -J\hat{\sigma}^z & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{bond}}^{[2\le j\le L-1]} = \begin{pmatrix} \hat{I} & 0 & 0 \\ \hat{\sigma}^z & 0 & 0 \\ 0 & -J\hat{\sigma}^z & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{bond}}^{[L]} = \begin{pmatrix} \hat{I} \\ \hat{\sigma}^z \\ 0 \end{pmatrix}, \tag{28}$$

which produces $-J\sum_{\langle i,j\rangle} \hat{\sigma}_i^z \hat{\sigma}_j^z$, with $\langle i, j\rangle$ denoting a sum over nearest neighbors i and j. The full Hamiltonian is given by the direct sum of the matrices. Collecting rows of the direct sum which are exactly parallel, we have the MPO representation of the full operator

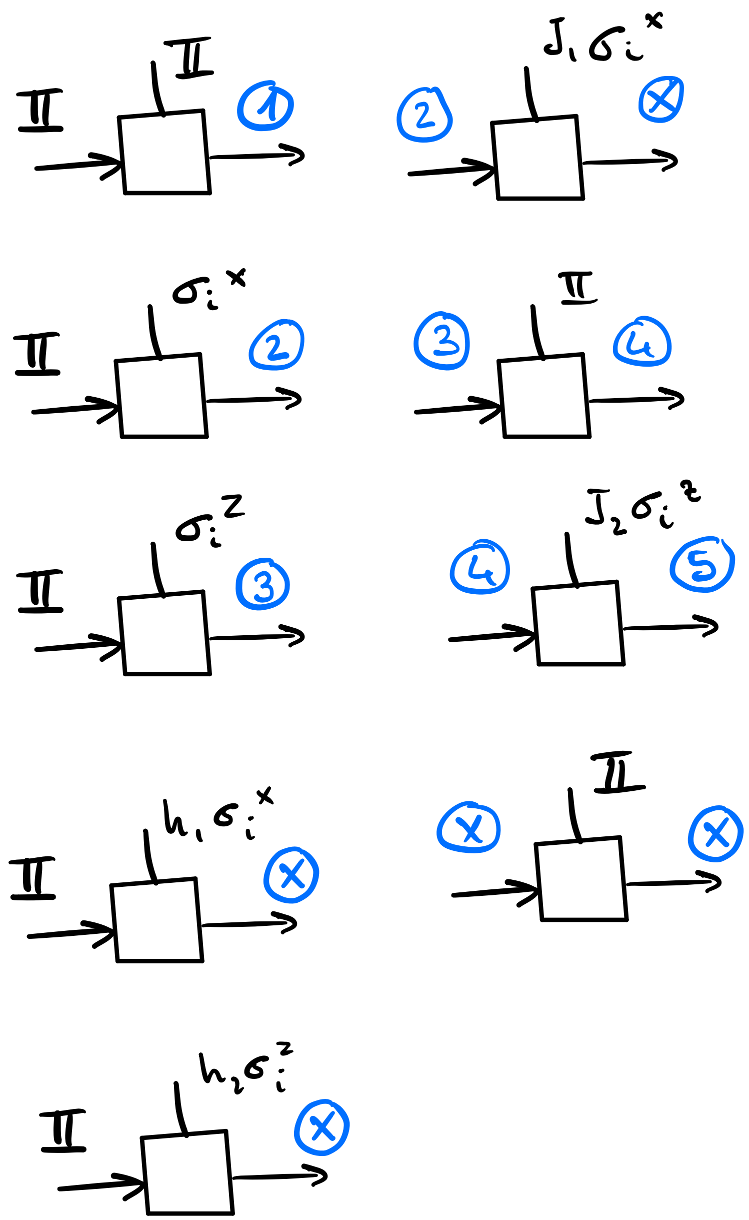
$$\mathcal{W}_{\mathrm{Ising}}^{[1]} = \begin{pmatrix} -h\hat{\sigma}^x & -J\hat{\sigma}^z & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{Ising}}^{[2\le j\le L-1]} = \begin{pmatrix} \hat{I} & 0 & 0 \\ \hat{\sigma}^z & 0 & 0 \\ -h\hat{\sigma}^x & -J\hat{\sigma}^z & \hat{I} \end{pmatrix}, \quad \mathcal{W}_{\mathrm{Ising}}^{[L]} = \begin{pmatrix} \hat{I} \\ \hat{\sigma}^z \\ -h\hat{\sigma}^x \end{pmatrix}. \tag{29}$$

How do we make an MPO for an arbitrary Hamiltonian?

How do we make an MPO for an arbitrary Hamiltonian?

- We can use the automata method, where the individual MPO tensor is a machine that gets operators as inputs and provides outputs:

$$H = \sum_i \left(J_1\, \sigma_i^x \sigma_{i+1}^x - J_2\, \sigma_i^z \sigma_{i+2}^z \right) + \sum_i h_1 \sigma_i^x + h_2 \sigma_i^z$$



$$W_i = \begin{pmatrix} \mathbb{I} & \sigma_i^x & \sigma_i^z & 0 & h_1\sigma_i^x + h_2\sigma_i^z \\ 0 & 0 & 0 & 0 & J_1\sigma_i^x \\ 0 & 0 & 0 & \mathbb{I} & 0 \\ 0 & 0 & 0 & 0 & J_2\sigma_i^z \\ 0 & 0 & 0 & 0 & \mathbb{I} \end{pmatrix}$$

$$W_1 = \begin{pmatrix} \mathbb{I} & \sigma_i^x & \sigma_i^z & 0 & h_1\sigma_i^x + h_2\sigma_i^z \end{pmatrix}$$

$$W_M = \begin{pmatrix} h_1\sigma_i^x + h_2\sigma_i^z \\ J_1\sigma_i^x \\ 0 \\ J_2\sigma_i^z \\ \mathbb{I} \end{pmatrix}$$

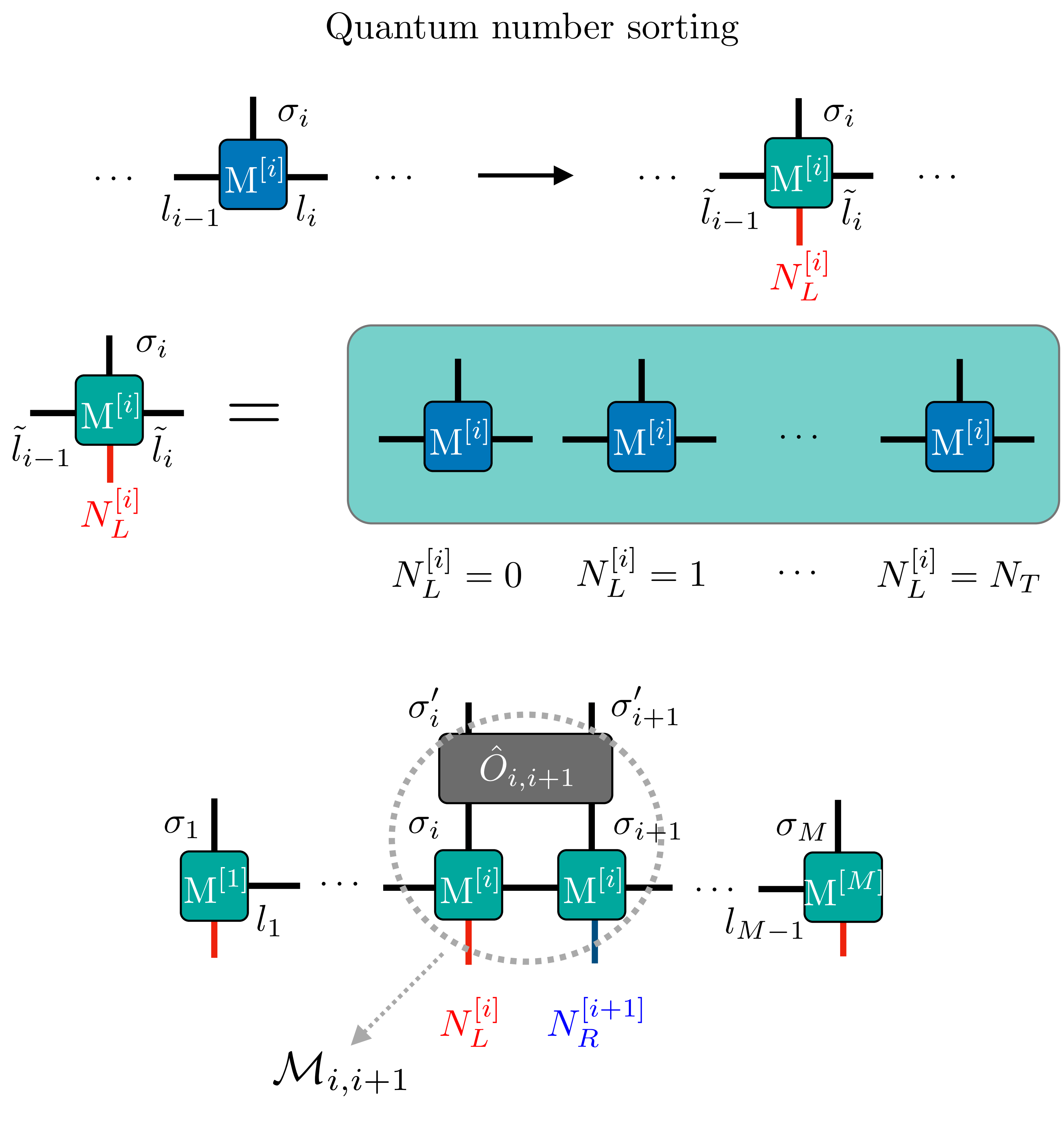
Encoding conservation laws in MPS

- We can incorporate certain symmetries into our MPS.

- E.g., if our Hamiltonian has an abelian symmetry like U(1), i.e. conserves the particle number. We can incorporate an extra leg to account for the quantum number.

- Then, whenever we apply an operator, we can take advantage of this block diagonal structure for speed-up.

- Useful for fermions. Where JW transformation generates highly non-local terms.

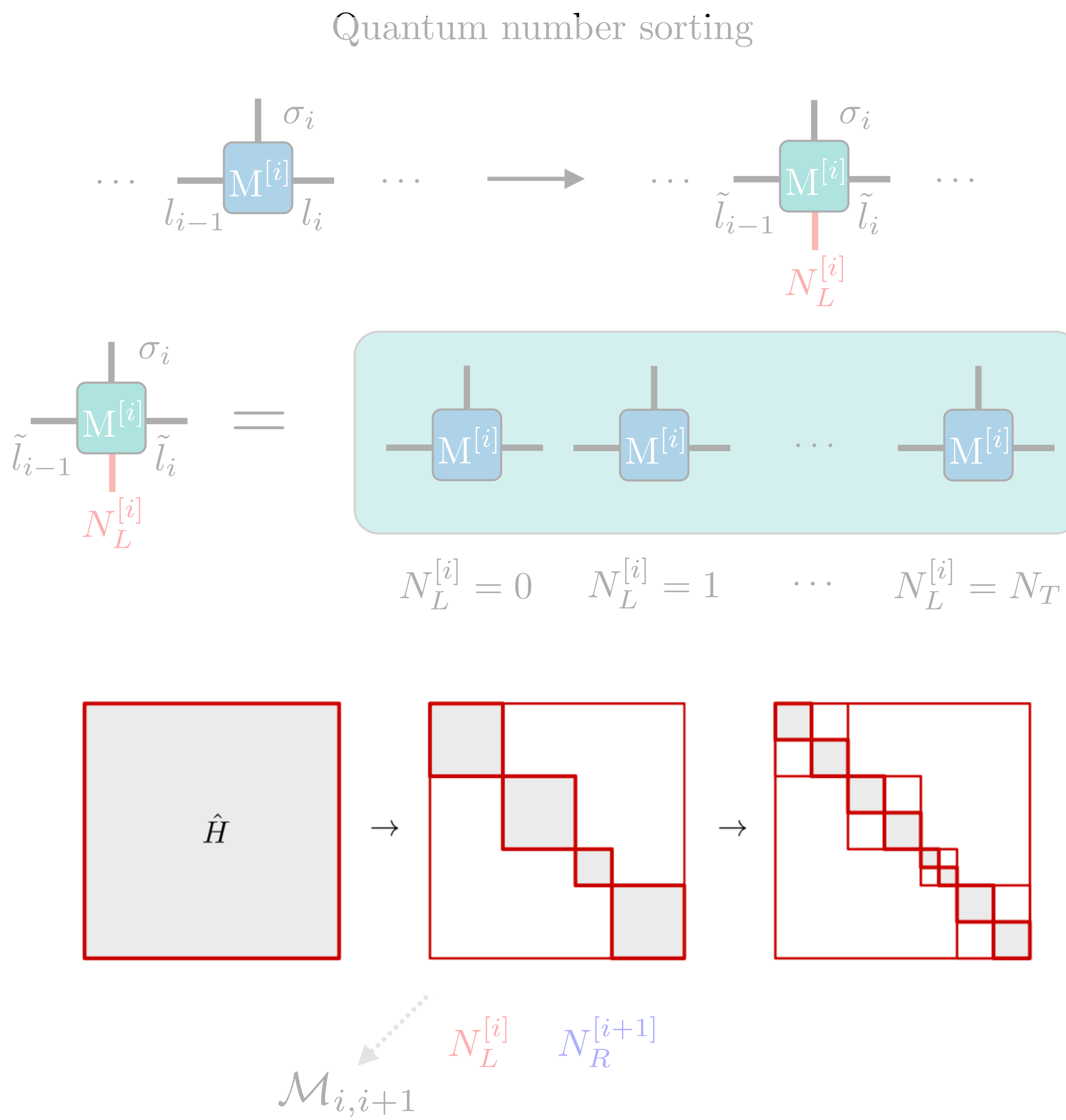


$$\mathcal{M}_{i,i+1} \neq 0 \Rightarrow N_L^{[i]} + N^{[i]} + N^{[i+1]} + N_R^{[i+1]}$$

Block-diagonal matrix

Encoding conservation laws in MPS

- We can incorporate certain symmetries into our MPS.
- E.g., if our Hamiltonian has an abelian symmetry like U(1), i.e. conserves the particle number. We can incorporate an extra leg to account for the quantum number.
- Then, whenever we apply an operator, we can take advantage of this block diagonal structure for speed-up.
- Useful for fermions. Where JW transformation generates highly non-local terms.



$$\mathcal{M}_{i,i+1} \neq 0 \Rightarrow N_L^{[i]} + N^{[i]} + N^{[i+1]} + N_R^{[i+1]}$$

Block-diagonal matrix

Algorithms with MPS/MPO

- Real (imaginary) time evolution $\Rightarrow$ TEBD

Suzuki-Trotter decomposition of the evolution operator:

$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

G. Vidal, Physical Review Letters 91, 147902 (2003)

Algorithms with MPS/MPO

- Real (imaginary) time evolution $\longrightarrow$ TEBD

Suzuki-Trotter decomposition of the evolution operator:

$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

*Other evolution methods can provide higher order approximation to the evolution operator at the cost of the application of more terms to our state: Commutator-Free Magnus Expansions, Krylov Subspace Propagation…

But they will all rely on realizing this $H_{i,i+1}$ in a compatible form with our MPS.

G. Vidal, Physical Review Letters 91, 147902 (2003)

Algorithms with MPS/MPO

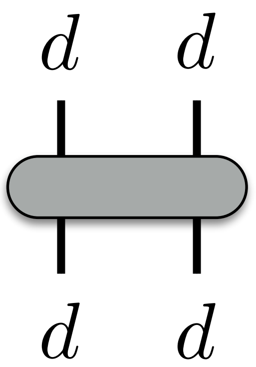
- Real (imaginary) time evolution $\longrightarrow$ TEBD

Suzuki-Trotter decomposition of the evolution operator:

$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

- If the couplings are local, then we can build two-site gates:

$$e^{-iH_{i,i+1}\delta t} =$$



G. Vidal, Physical Review Letters 91, 147902 (2003)

Algorithms with MPS/MPO

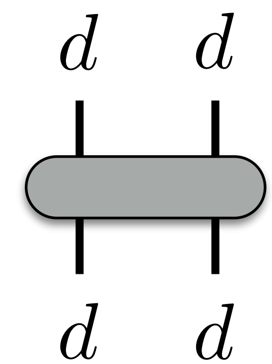
- Real (imaginary) time evolution $\longrightarrow$ TEBD

  Suzuki-Trotter decomposition of the evolution operator:

$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

- If the couplings are local, then we can build two-site gates:

$$e^{-iH_{i,i+1}\delta t} =$$



- How, to build this? We just need to consider the terms that form H:

  E.g.: spin 1/2, we can explicitly build the tensor product of operators and reshape it.

$$J\sigma_i^x \otimes \sigma_{i+1}^x \rightarrow (d^2, d^2) \rightarrow (d, d, d, d)$$

  Taking into account that local terms need to be split into two parts:

$$h\sigma_i^z \rightarrow \frac{h}{2}\sigma_i^z + \frac{h}{2}\sigma_{i+1}^z, \text{ for } 1 < i < M, \; i = 1 \; h\sigma_1^z + \frac{h}{2}\sigma_2^z$$

G. Vidal, Physical Review Letters 91, 147902 (2003)

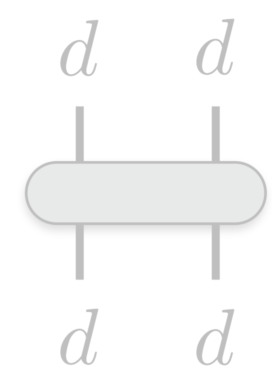
Algorithms with MPS/MPO

- Real (imaginary) time evolution → TEBD

Trotter decomposition of the evolution operator:

```
% the interaction part is the same
Hlocal = J * kron(sx,sx);

% the local part depends on the site position
if      ii==1
    Hlocal = Hlocal + B * ( kron(sz,id)   + kron(id,sz)/2 );
elseif ii==M-1
    Hlocal = Hlocal + B * ( kron(sz,id)/2 + kron(id,sz)   );
else
    Hlocal = Hlocal + B * ( kron(sz,id)/2 + kron(id,sz)/2 );
end
```

- If the coupl
site gates:



- How, to build this? We just need to consider the terms that form H:

E.g.: spin 1 … eshape it.

```
% bring to "gate form" from "kron form"
% (this is an important step, as kron and reshape order states
% differently)
Hlocal=reshape(Hlocal,[d d d d]);
Hlocal=permute(Hlocal,[2 1 4 3]);
Hlocal=reshape(Hlocal,[d*d d*d]);
```

Taking into

$h\sigma_i^z$ … $-\frac{h}{2}\sigma_2^z$

Important to keep track of how every function manages the index orders

G. Vidal, Physical Review Letters 91, 147902 (2003)

Algorithms with MPS/MPO

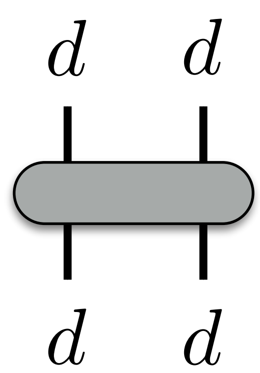
- Real (imaginary) time evolution → TEBD

Suzuki-Trotter decomposition of the evolution operator:

$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

- If the couplings are local, then we can build two-site gates:

$$e^{-iH_{i,i+1}\delta t} =$$



- How, to build this? We just need to consider the terms that form H:

E.g.: in cold atoms where often we cannot use simply tensor products, we can simply every element of the 4-legged tensor.

$$J a^\dagger_{i,\uparrow} a_{i+1,\uparrow} |0,\uparrow\rangle = |\uparrow, 0\rangle \;\rightarrow\; \text{Element: } (\uparrow, 0, 0, \uparrow) = J$$

$$\ldots$$

G. Vidal, Physical Review Letters 91, 147902 (2003)

Algorithms with MPS/MPO

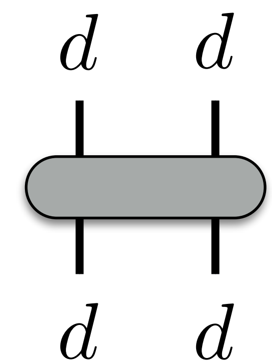
- Real (imaginary) time evolution → TEBD

Suzuki-Trotter decomposition of the evolution operator:

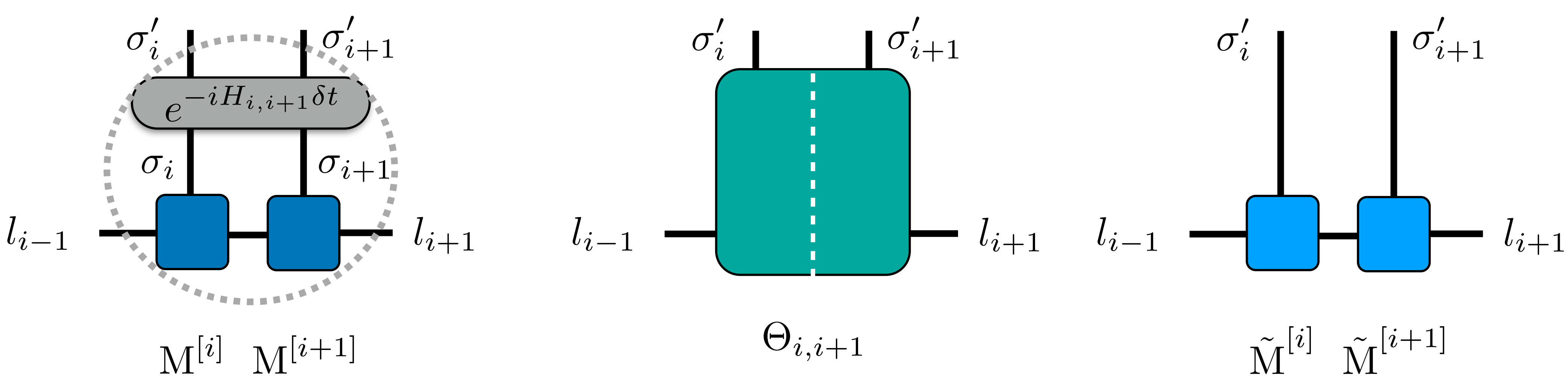
$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

- If the couplings are local, then we can build two-site gates:

$$e^{-iH_{i,i+1}\delta t} =$$



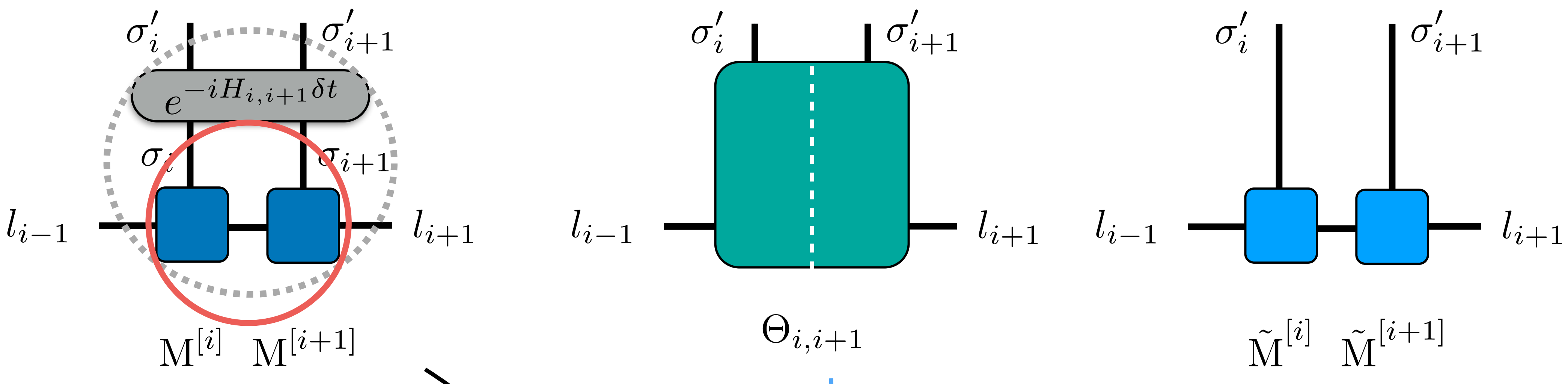
- If we apply it to our MPS:



G. Vidal, Physical Review Letters 91, 147902 (2003)

Algorithms with MPS/MPO

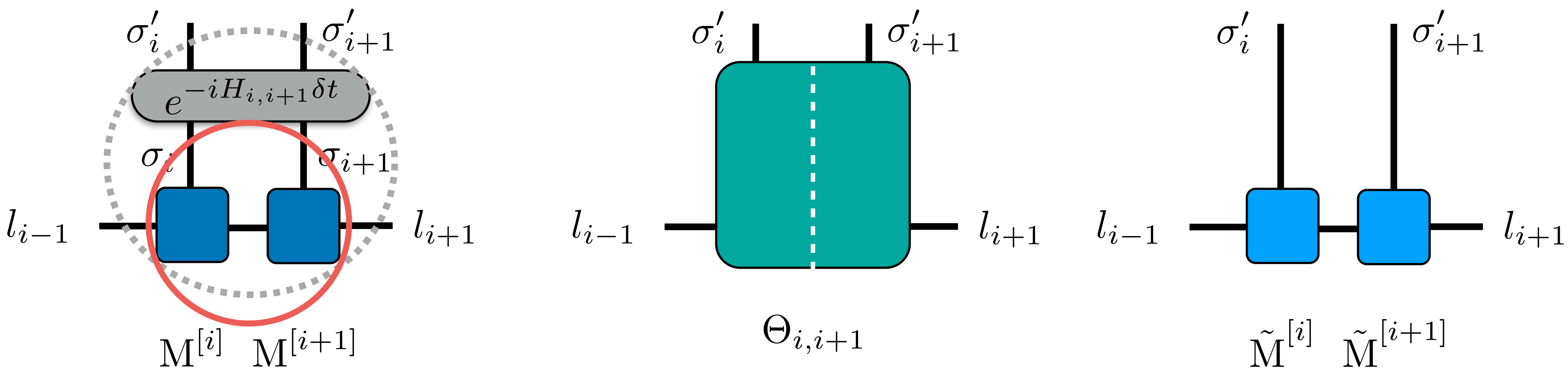
- If we apply it to our MPS:




```
[d0,DL0,DR0]=size(mps.data{pos});
[d1,DL1,DR1]=size(mps.data{pos+1});

LL=reshape(mps.data{pos}, [d0*DL0 DR0]);
RR=permute(mps.data{pos+1}, [2 1 3]);
RR=reshape(RR, [DL1 d1*DR1]);

joined=reshape(LL*RR, [d0 DL0 d1 DR1]);
joined=permute(joined, [1 3 2 4]);
joined=reshape(joined, [d0*d1 DL0*DR1]);

joined=reshape(loc2op*joined, [d0 d1 DL0 DR1]);
joined=permute(joined, [1 3 2 4]);
joined=reshape(joined, [d0*DL0 d1*DR1]);
```

Algorithms with MPS/MPO

- If we apply it to our MPS:



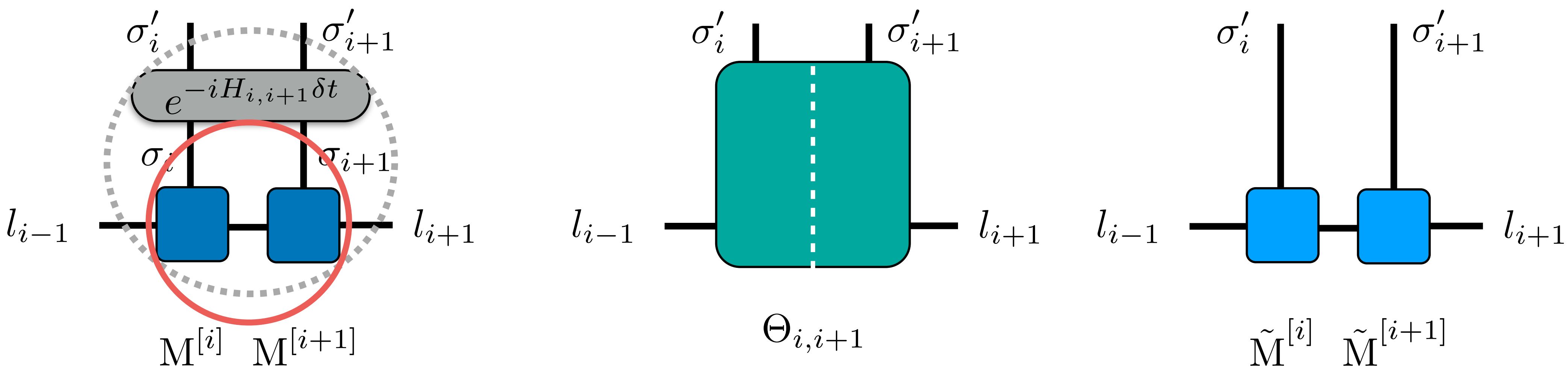

```
[d0,DL0,DR0]=size(mps.data{pos});
[d1,DL1,DR1]=size(mps.data{pos+1});

LL=reshape(mps.data{pos}, [d0*DL0 DR0]);
RR=permute(mps.data{pos+1}, [2 1 3]);
RR=reshape(RR, [DL1 d1*DR1]);

joined=reshape(LL*RR, [d0 DL0 d1 DR1]);
joined=permute(joined, [1 3 2 4]);
joined=reshape(joined, [d0*d1 DL0*DR1]);

joined=reshape(loc2op*joined, [d0 d1 DL0 DR1]);
joined=permute(joined, [1 3 2 4]);
joined=reshape(joined, [d0*DL0 d1*DR1]);
```

SVD + truncation:

```
[L,lam,R]=svdecon(joined);

if is_dyn_trunc
    cDmax=min(length(find(lam>mps.zero_thres*max(lam))),mps.Dmax); % dynamic
else
       cDmax=min(DR0,mps.Dmax); % D can only grow
    cDmax=min(max(DR0,length(find(lam>mps.zero_thres*max(lam)))),mps.Dmax);
end
trunc=sum(lam((cDmax+1):end).^2);
```

Algorithms with MPS/MPO

- If we apply it to our MPS:




```
[d0,DL0,DR0]=size(mps.data{pos});
[d1,DL1,DR1]=size(mps.data{pos+1});

LL=reshape(mps.data{pos}, [d0*DL0 DR0]);
RR=permute(mps.data{pos+1}, [2 1 3]);
RR=reshape(RR, [DL1 d1*DR1]);

joined=reshape(LL*RR, [d0 DL0 d1 DR1]);
joined=permute(joined, [1 3 2 4]);
joined=reshape(joined, [d0*d1 DL0*DR1]);

joined=reshape(loc2op*joined, [d0 d1 DL0 DR1]);
joined=permute(joined, [1 3 2 4]);
joined=reshape(joined, [d0*DL0 d1*DR1]);
```

SVD + truncation:

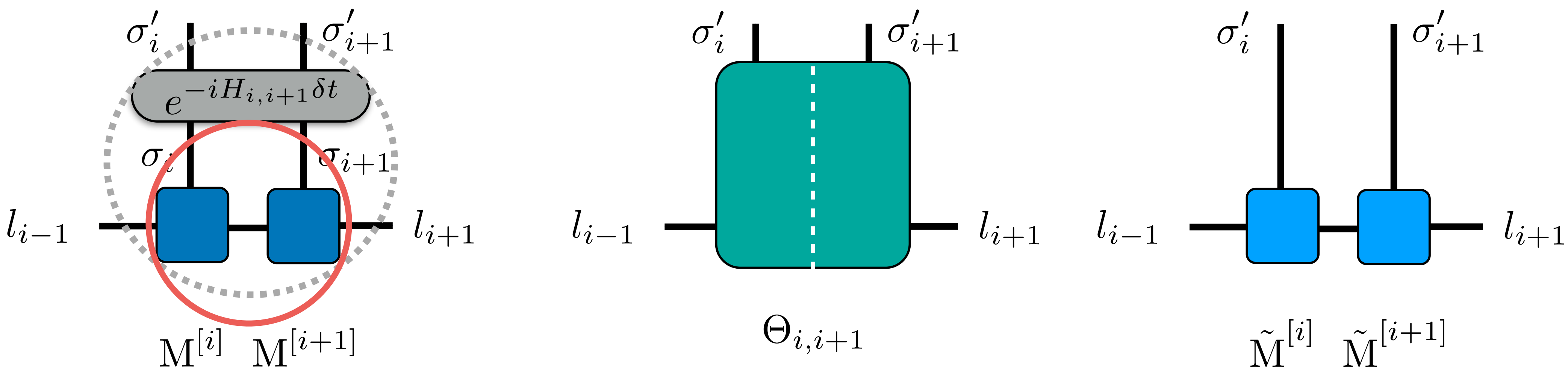
```
[L,lam,R]=svdecon(joined);

if is_dyn_trunc
    cDmax=min(length(find(lam>mps.zero_thres*max(lam))),mps.Dmax); % dynamic
else
      cDmax=min(DR0,mps.Dmax); % D can only grow
    cDmax=min(max(DR0,length(find(lam>mps.zero_thres*max(lam)))),mps.Dmax);
end
trunc=sum(lam((cDmax+1):end).^2);
```

Reshape into MPS:

```
if ~lr
    % right tensor will be in the rigth canonical form
    L=L(:,1:cDmax)*diag(lam(1:cDmax));
    R=R(1:cDmax,:);
else
    % left  tensor will be in the left  canonical form
    L=L(:,1:cDmax);
    R=diag(lam(1:cDmax))*R(1:cDmax,:);
end
mps.data{pos}=reshape(L,[d0 DL0 cDmax]);

R=reshape(R, [cDmax d1 DR1]);
R=permute(R, [2 1 3]);
mps.data{pos+1}=reshape(R,[d1 cDmax DR1]);
```

Algorithms with MPS/MPO

- If we apply it to our MPS:




```
[d0,DL0,DR0]=size(mps.data{pos});
[d1,DL1,DR1]=size(mps.data{pos+1});

LL=reshape(mps.data{pos}, [d0*DL0 DR0]);
RR=permute(mps.data{pos+1}, [2 1 3]);
RR=reshape(RR, [DL1 d1*DR1]);

joined=reshape(LL*RR, [d0 DL0 d1 DR1]);
joined=permute(joined, [1 3 2 4]);
joined=reshape(joined, [d0*d1 DL0*DR1]);

joined=reshape(loc2op*joined, [d0 d1 DL0 DR1]);
joined=permute(joined, [1 3 2 4]);
joined=reshape(joined, [d0*DL0 d1*DR1]);
```

SVD + truncation:

```
[L,lam,R]=svdecon(joined);

if is_dyn_trunc
    cDmax=min(length(find(lam>mps.zero_thres*max(lam))),mps.Dmax); % dynamic
else
      cDmax=min(DR0,mps.Dmax); % D can only grow
    cDmax=min(max(DR0,length(find(lam>mps.zero_thres*max(lam)))),mps.Dmax);
end
trunc=sum(lam((cDmax+1):end).^2);
```

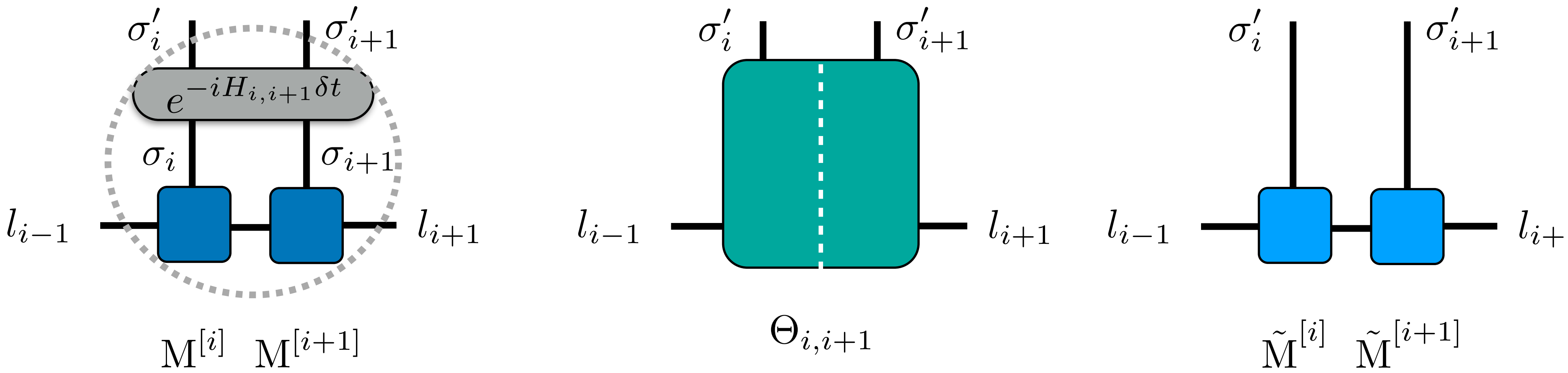
Reshape into MPS:

```
if ~lr
    % right tensor will be in the rigth canonical form
    L=L(:,1:cDmax)*diag(lam(1:cDmax));
    R=R(1:cDmax,:);
else
    % left  tensor will be in the left  canonical form
    L=L(:,1:cDmax);
    R=diag(lam(1:cDmax))*R(1:cDmax,:);
end
mps.data{pos}=reshape(L,[d0 DL0 cDmax]);

R=reshape(R, [cDmax d1 DR1]);
R=permute(R, [2 1 3]);
mps.data{pos+1}=reshape(R,[d1 cDmax DR1]);
```

We need to know index orders!

Algorithms with MPS/MPO

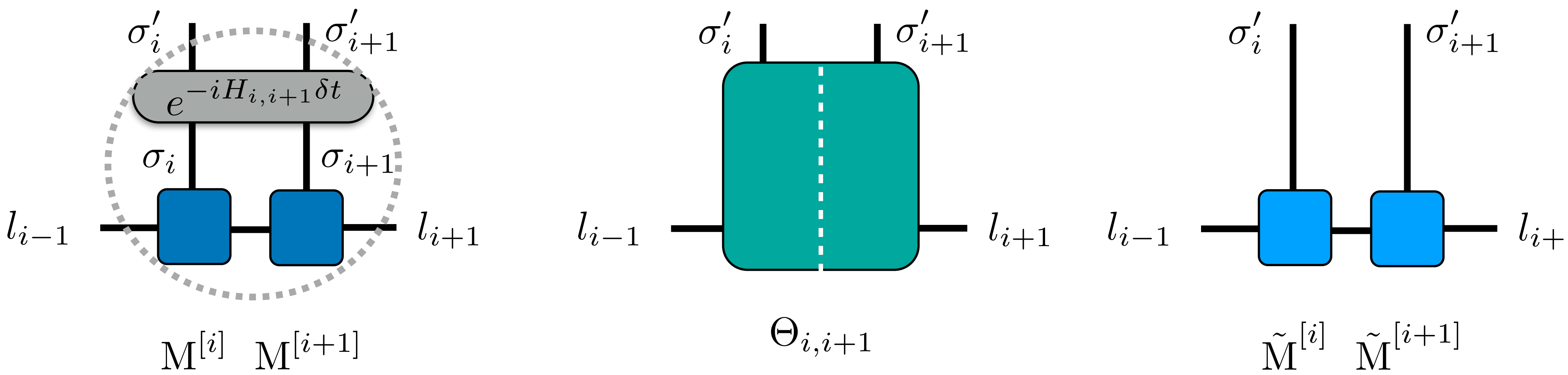
- If we apply it to our MPS:



- When we apply this, we can loose orthonormality of local basis in (i-1) and (i+1) due to truncation error (also if applying a non-unitary operator). Solutions:

Algorithms with MPS/MPO

- If we apply it to our MPS:



- When we apply this, we can loose orthonormality of local basis in (i-1) and (i+1) due to truncation error (also if applying a non-unitary operator). Solutions:

1. small time step.
2. Full orthogonalization sweep.

| Not very efficient

3. Higher order Trotter including a transpose sweep to avoid ever acting on non-orthogonal basis before SVD.

$$\hat{U}_H = \hat{L}\,\hat{R}\,\hat{L}\,\hat{R}'\,\hat{L}\,\hat{L}\,\hat{L}\,\hat{L}\,\hat{R}\,\hat{L}\,\hat{R}\,\hat{R}\,\hat{R}\,\hat{R}\,\hat{L}'\,\hat{R}\,\hat{L}\,\hat{R} + \mathcal{O}(\Delta t^5)$$

$$\hat{R} = \left(\prod_k e^{-i\frac{\Delta t}{12}\hat{H}^{[nn]}_{k,k+1}}\right)^T \qquad \hat{R}' = \left(\prod_k e^{i\frac{\Delta t}{6}\hat{H}^{[nn]}_{k,k+1}}\right)^T$$

$$\hat{L} = \left(\prod_k e^{-i\frac{\Delta t}{12}\hat{H}^{[nn]}_{k,k+1}}\right) \qquad \hat{L}' = \left(\prod_k e^{i\frac{\Delta t}{6}\hat{H}^{[nn]}_{k,k+1}}\right)$$

Algorithms with MPS/MPO

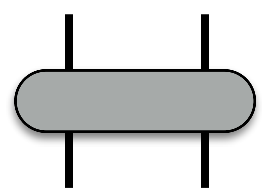
- Real (imaginary) time evolution ⟶ TEBD

Trotter decomposition of the evolution operator:

$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

- The evolution is obtained applying the gates to our state sequentially:

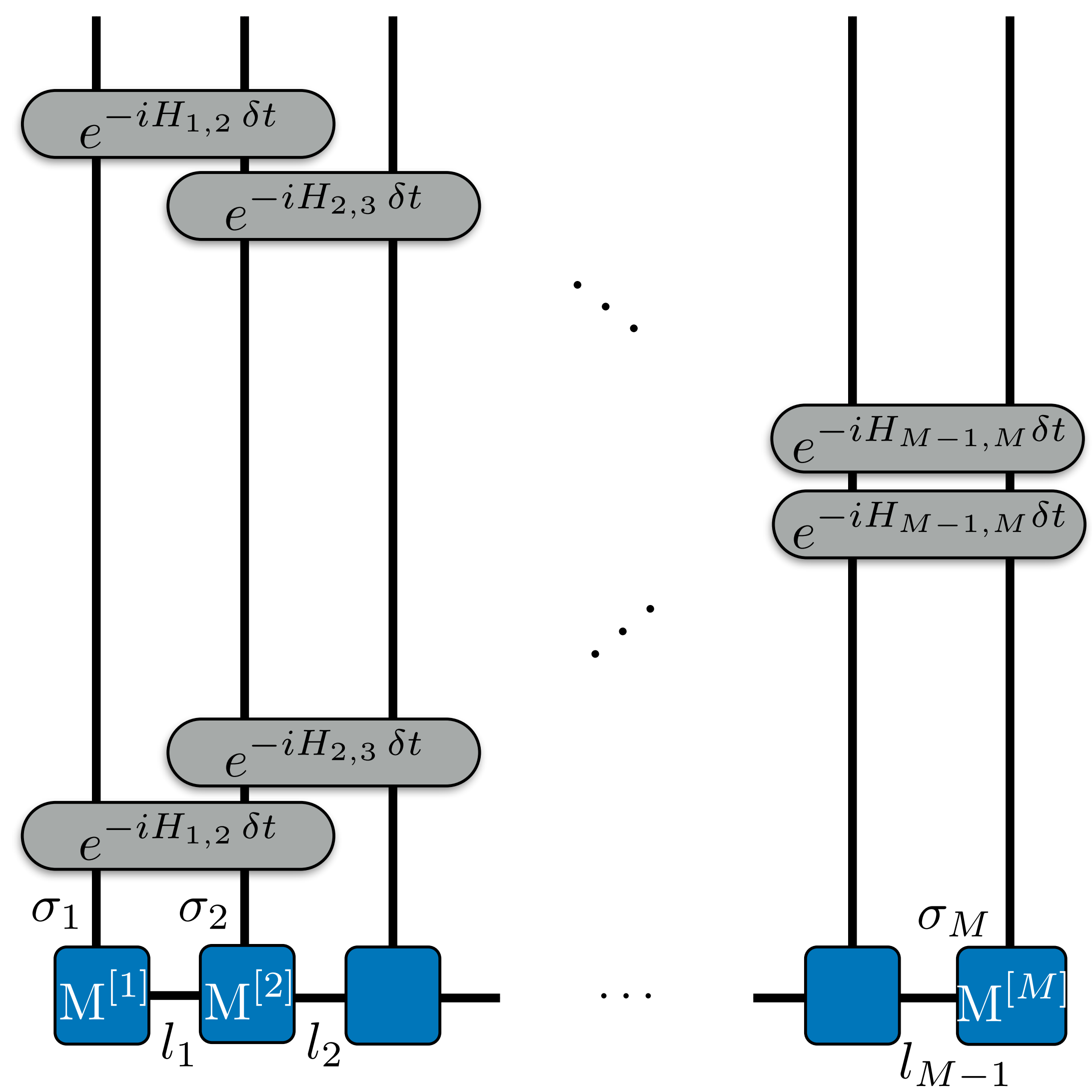
$$e^{-iH_{i,i+1}\delta t} =$$



Inside $H_{i,i+1}$ terms that belong to H_i and H_{i+1} do not commute.

↓

Higher order expansions account for it.



G. Vidal, Physical Review Letters 91, 147902 (2003)

Algorithms with MPS/MPO

- Real (imaginary) time evolution $\longrightarrow$ TEBD

Trotter decomposition of the evolution operator:

$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

- Non local interactions? Build Hamiltonian as MPO

G. Vidal, Physical Review Letters 91, 147902 (2003)

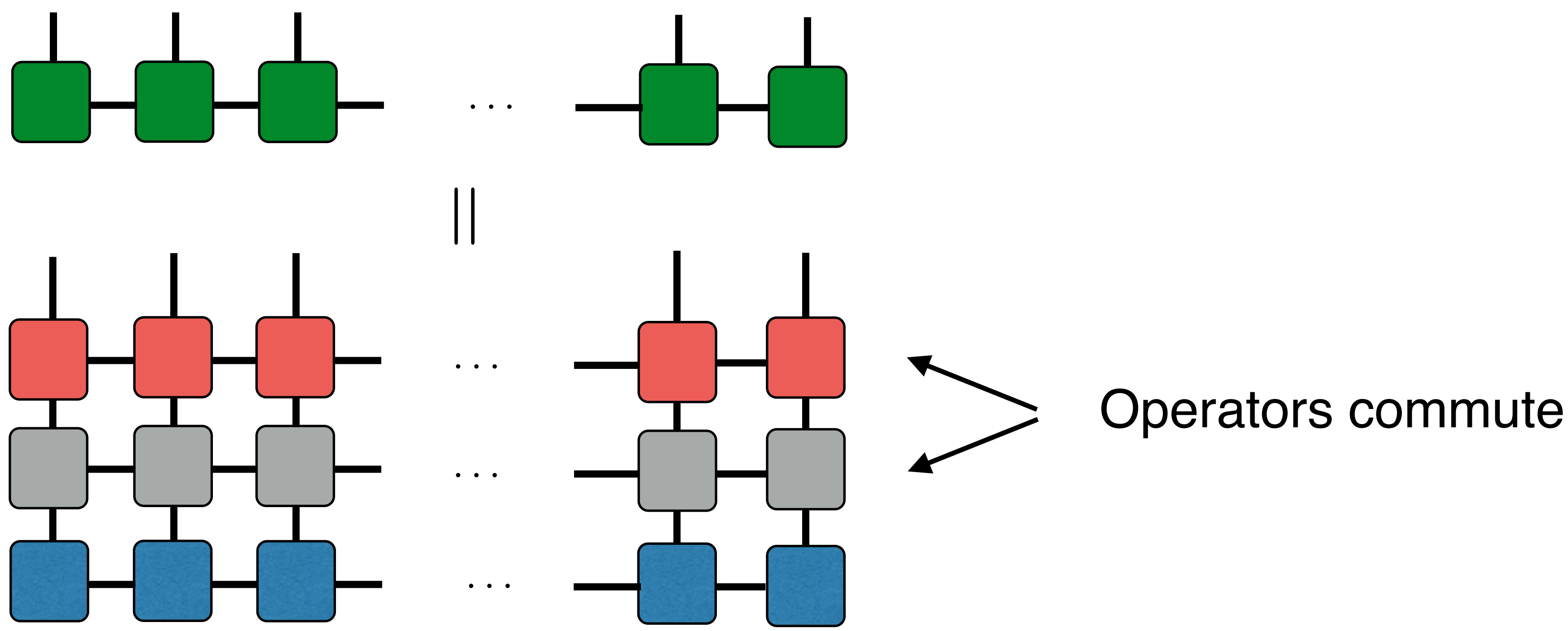
Algorithms with MPS/MPO

- Real (imaginary) time evolution → TEBD

  Trotter decomposition of the evolution operator:

$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

- Non local interactions? Build Hamiltonian as MPO



$$|\phi(t+\delta t)\rangle = e^{-iH_{even}\delta t} e^{-iH_{odd}\delta t} |\phi(t)\rangle$$

G. Vidal, Physical Review Letters 91, 147902 (2003)

Algorithms with MPS/MPO: imaginary time evolution

- How does imaginary time evolution work? What would we use it?

- We obtain the GS of a quantum system by performing time evolution with imaginary time units.

$$|\psi\rangle = \sum_n \alpha_n |\phi_n\rangle$$

$$|\psi(\tau)\rangle = \sum_{n=0} \alpha_n e^{-E_n \tau} |\phi_n\rangle = \sum_{n=0} \alpha_n e^{-E_0 (E_n/E_0)\tau} |\phi_n\rangle =$$

$$= e^{-E_0 \tau} |\phi_0\rangle + \sum_{n=1} \alpha_n e^{-E_0 (E_n/E_0)\tau} |\phi_n\rangle \stackrel{\tau \to \infty}{=} |\phi_0\rangle$$

- Obtaining an exponential decay the slowest decaying term corresponding to the minimum energy, is the one that remains.

- In principle, we require some overlap of our initial state and the ground state. However, numerical error comes in our aid since any numerical noise proportional to the ground state will be exponentially amplified during the procedure.

Algorithms with MPS/MPO

- Real (imaginary) time evolution → TEBD

Trotter decomposition of the evolution operator:

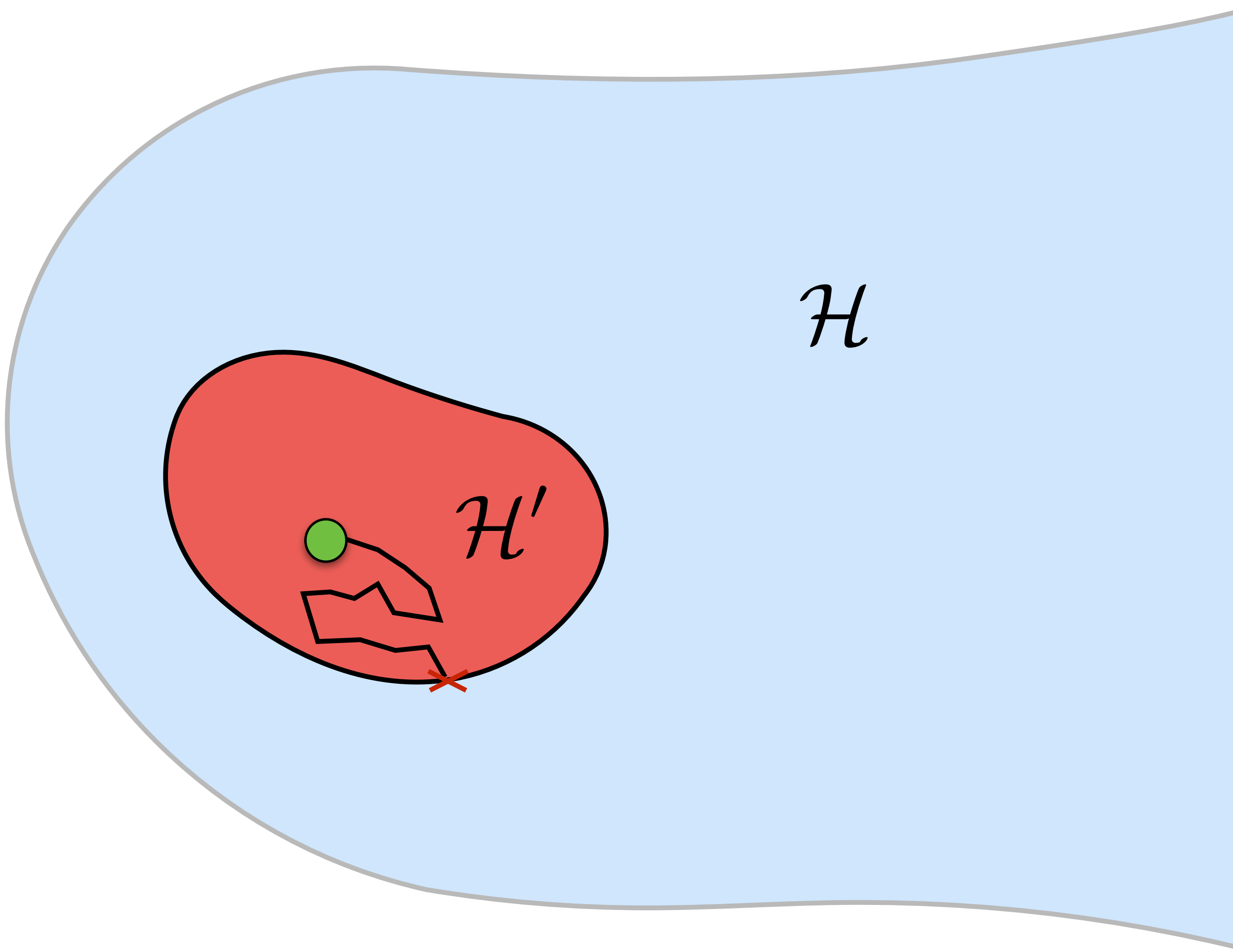
$$\hat{U}_H(\delta t) = e^{-iH\delta t} = \prod_i e^{-iH_{i,i+1}\delta t} + \mathcal{O}(\delta t^2)$$

- What is the limit?

The time we can evolve is limited by how accurate our representation is:

Time evolution usually causes entanglement to grow.

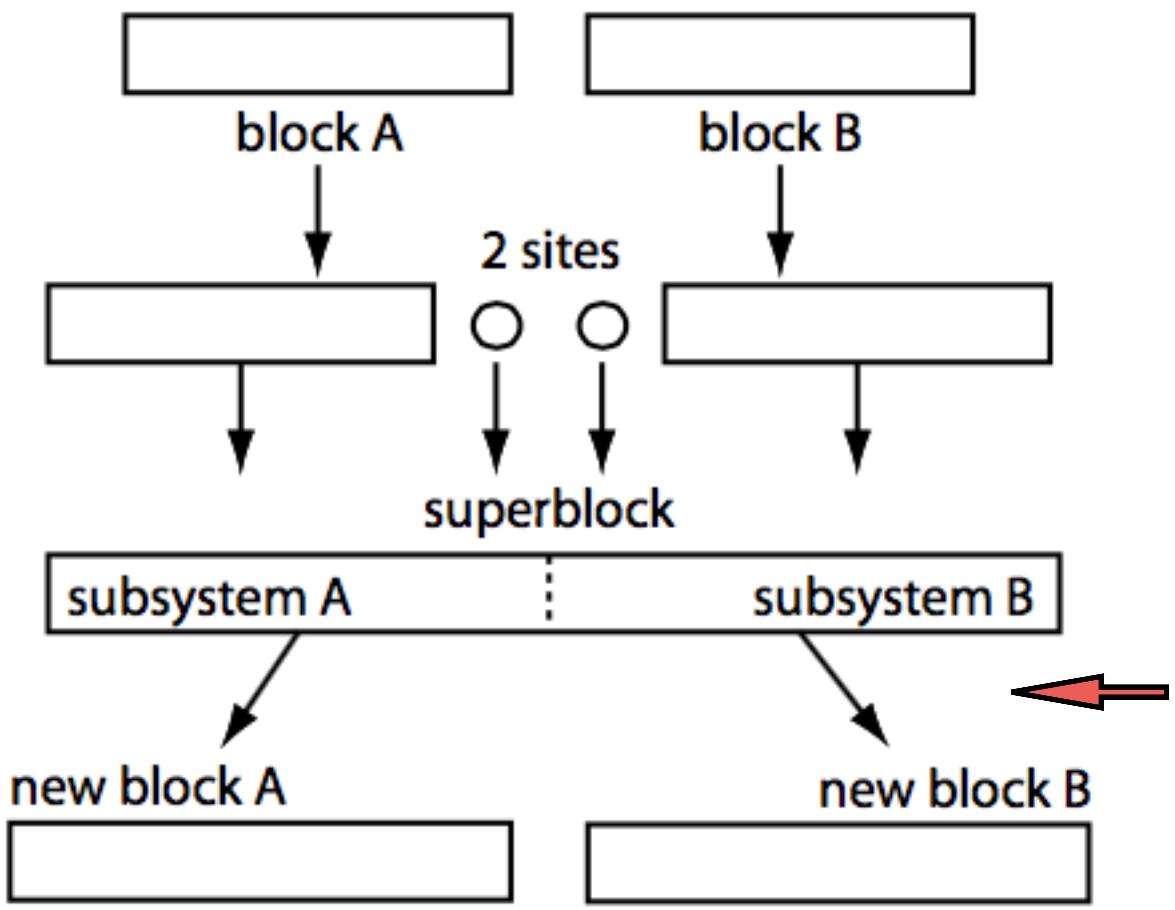
Our representation becomes poor if the projection into $\mathcal{H}'$ is smaller and smaller.



Algorithms with MPS/MPO: DMRG, GS variational state search

- Density matrix renormalization group techniques (DMRG)

Variational method to compute ground states in large systems.



Assume we know the state of size L

We increase each block to $L+1$

We compute the ground state

Truncation

It relays on a truncation procedure identical to the one explained.

- It can be applied to study stationary properties in the context of MPS

S. R. White, PRL 69, 2863 (1992)

S. R. White and A. E. Feiguin, Physical Review Letters 93, 076401 (2004)

Algorithms with MPS/MPO: DMRG, GS variational state search

- How does the DMRG work in the case of MPS, building a variational state search:

  We want to minimize our energy of our system:

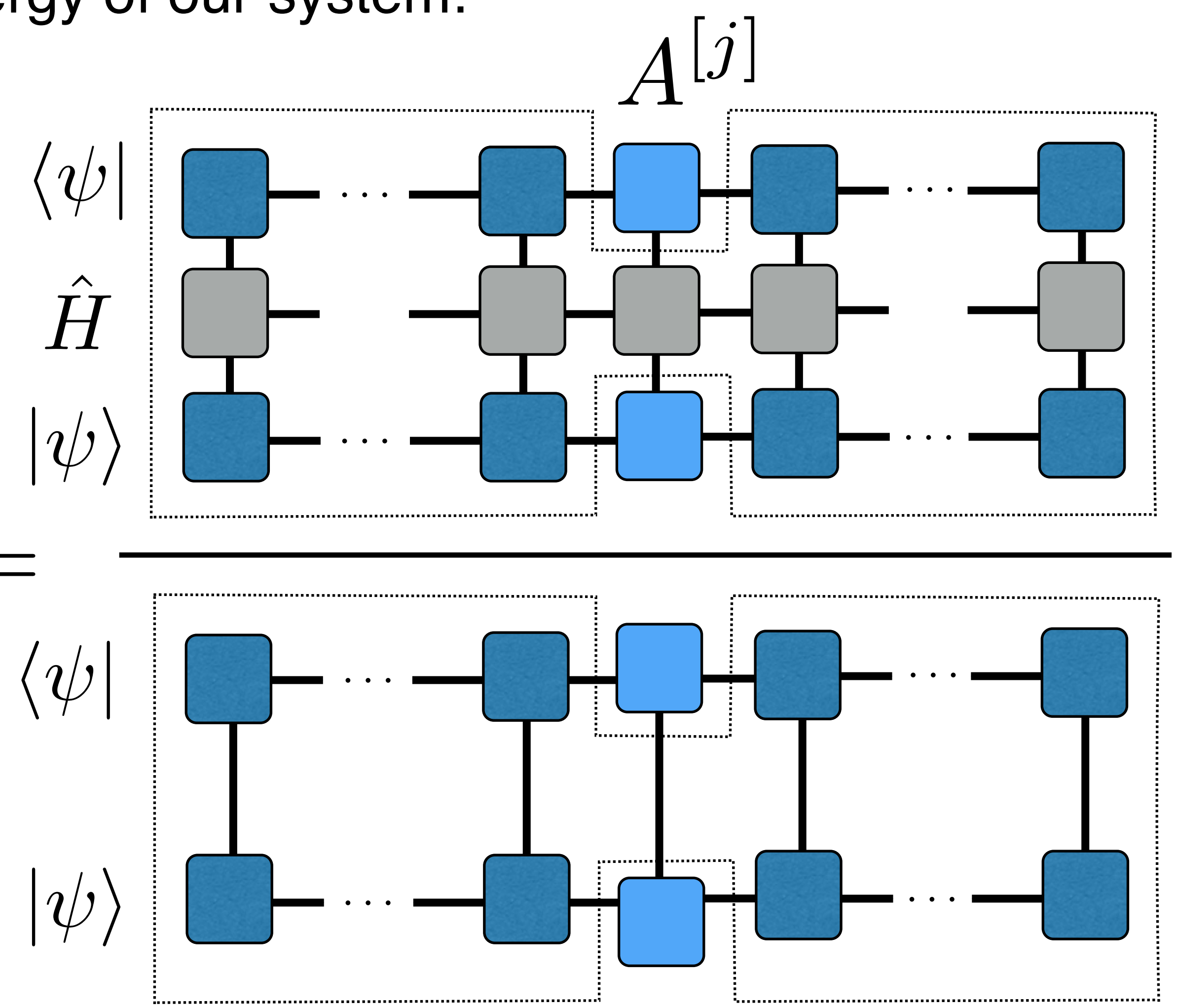
$$\frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle}$$

Algorithms with MPS/MPO: DMRG, GS variational state search

- How does the DMRG work in the case of MPS, building a variational state search:

We want to minimize our energy of our system:

$$\frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle} =$$

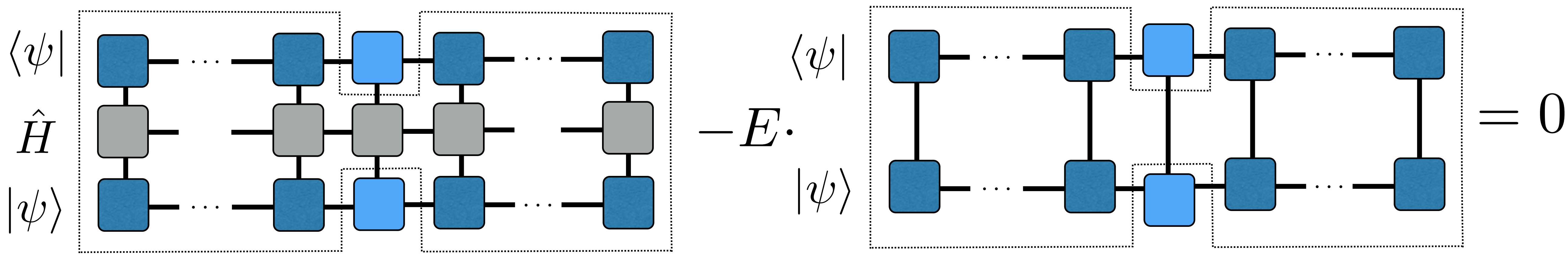


We can rephrase this problem into finding the extremum of the following expression:

$$\underbrace{\frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle}}_{=E} \rightarrow \frac{\partial}{\partial A^{[j]}}\left[\langle\psi|H|\psi\rangle - E\langle\psi|\psi\rangle\right] = 0$$

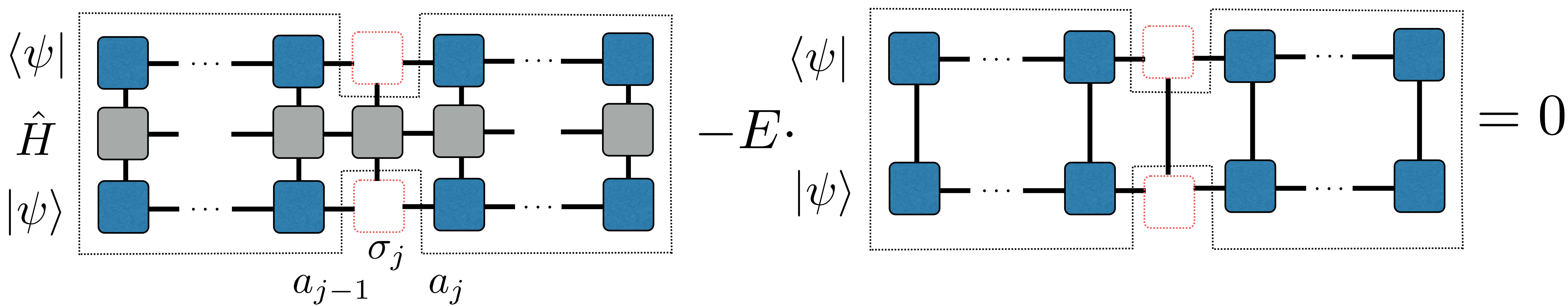
Algorithms with MPS/MPO: DMRG, GS variational state search

$$\underbrace{\frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle}}_{=E} \rightarrow \frac{\partial}{\partial A^{[j]}}\left[\langle\psi|H|\psi\rangle - E\langle\psi|\psi\rangle\right] = 0$$



Algorithms with MPS/MPO: DMRG, GS variational state search

$$\underbrace{\frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle}}_{=E} \rightarrow \frac{\partial}{\partial A^{[j]}}\left[\langle\psi|H|\psi\rangle - E\langle\psi|\psi\rangle\right] = 0$$



We need to find the solution by modifying the local tensor. This is a generalized eigenvalue problem for the effective matrix:

$$\mathcal{H}^{[j]} := \langle a_{j-1}\sigma_j a_j|\,\hat{H}\,|a_{j-1}\sigma_j a_j\rangle$$

Where our state is its ground state with dimension d*D²: $dim(V_j) = (a_{j-1}\sigma_j a_j)$

$$\mathcal{H}^{[j]}V_j - E\mathcal{N}^{[j]}V_j$$

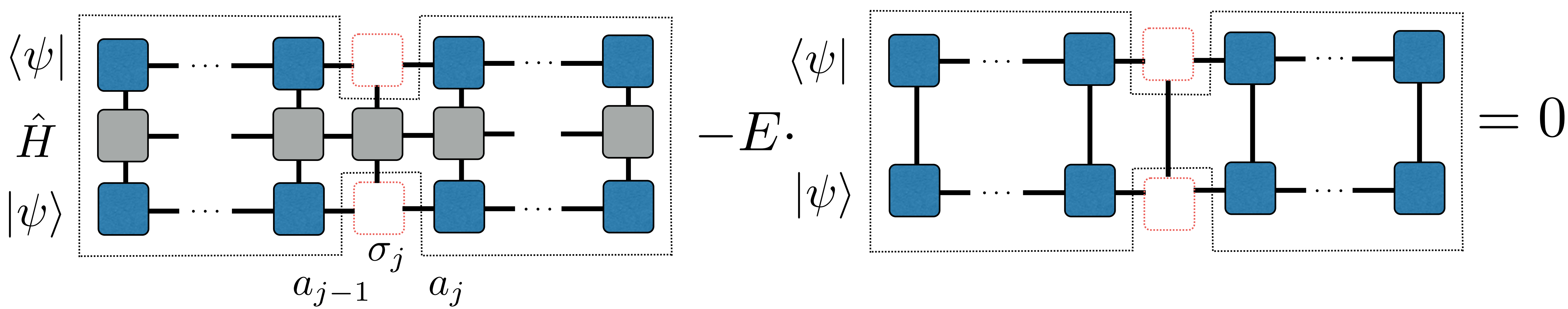
$$\mathcal{H}^{[j]}V_j = \lambda^{(j)}_{min}\mathcal{N}^{[j]}V_j$$

Dim. does not depend on system size!

Scaling linear with number of sites

Algorithms with MPS/MPO: DMRG, GS variational state search

$$\underbrace{\frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle}}_{=E} \to \frac{\partial}{\partial A^{[j]}}\left[\langle\psi|H|\psi\rangle - E\langle\psi|\psi\rangle\right] = 0$$



We need to find the solution by modifying the local tensor. This is a generalized eigenvalue problem for the effective matrix:

$$\mathcal{H}^{[j]} := \langle a_{j-1}\sigma_j a_j|\,\hat{H}\,|a_{j-1}\sigma_j a_j\rangle$$

Where our state is its ground state with dimension d*D²: $\quad dim(V_j) = (a_{j-1}\sigma_j a_j)$

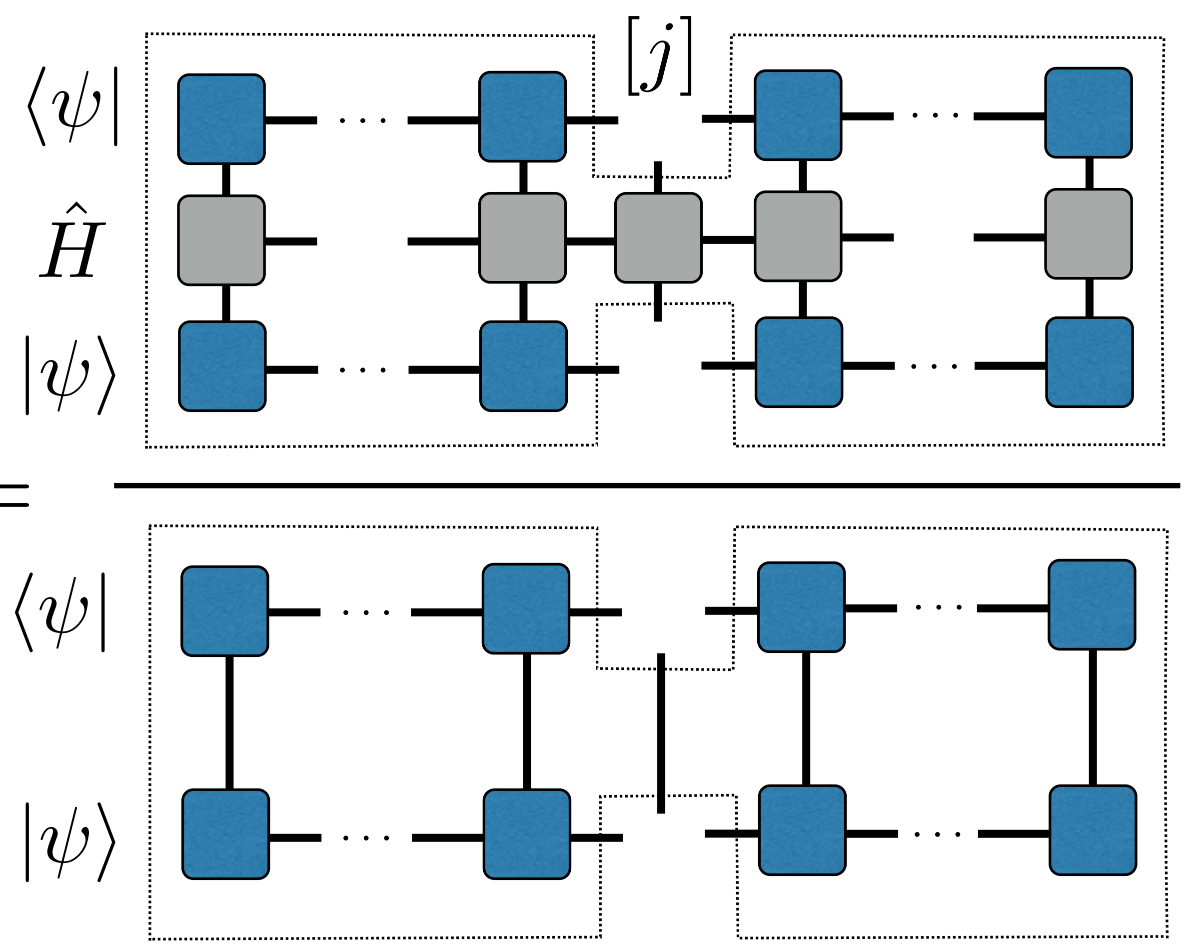
$$\begin{gathered}\mathcal{H}^{[j]}V_j - E\mathcal{N}^{[j]}V_j \\ \mathcal{H}^{[j]}V_j = \lambda_{min}^{(j)}\mathcal{N}^{[j]}V_j\end{gathered} \longrightarrow min\left(\frac{V_j^{\dagger}\mathcal{H}^{[j]}V_j}{V_j^{\dagger}\mathcal{N}^{[j]}V_j}\right)$$

Algorithms with MPS/MPO: DMRG, GS variational state search

- Ground state variational search

We have to solve a generalized eigenvalue problem: $\mathcal{H}^{[j]} := \langle a_{j-1}\sigma_j a_j | \, \hat{H} \, | a_{j-1}\sigma_j a_j \rangle$

$$\frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle} =$$



$$= \frac{\mathcal{H}^{[j]}}{\mathcal{N}^{[j]}}$$

$$\mathcal{H}^{[j]} V_j = \lambda_{min}^{(j)} \mathcal{N}^{[j]} V_j$$

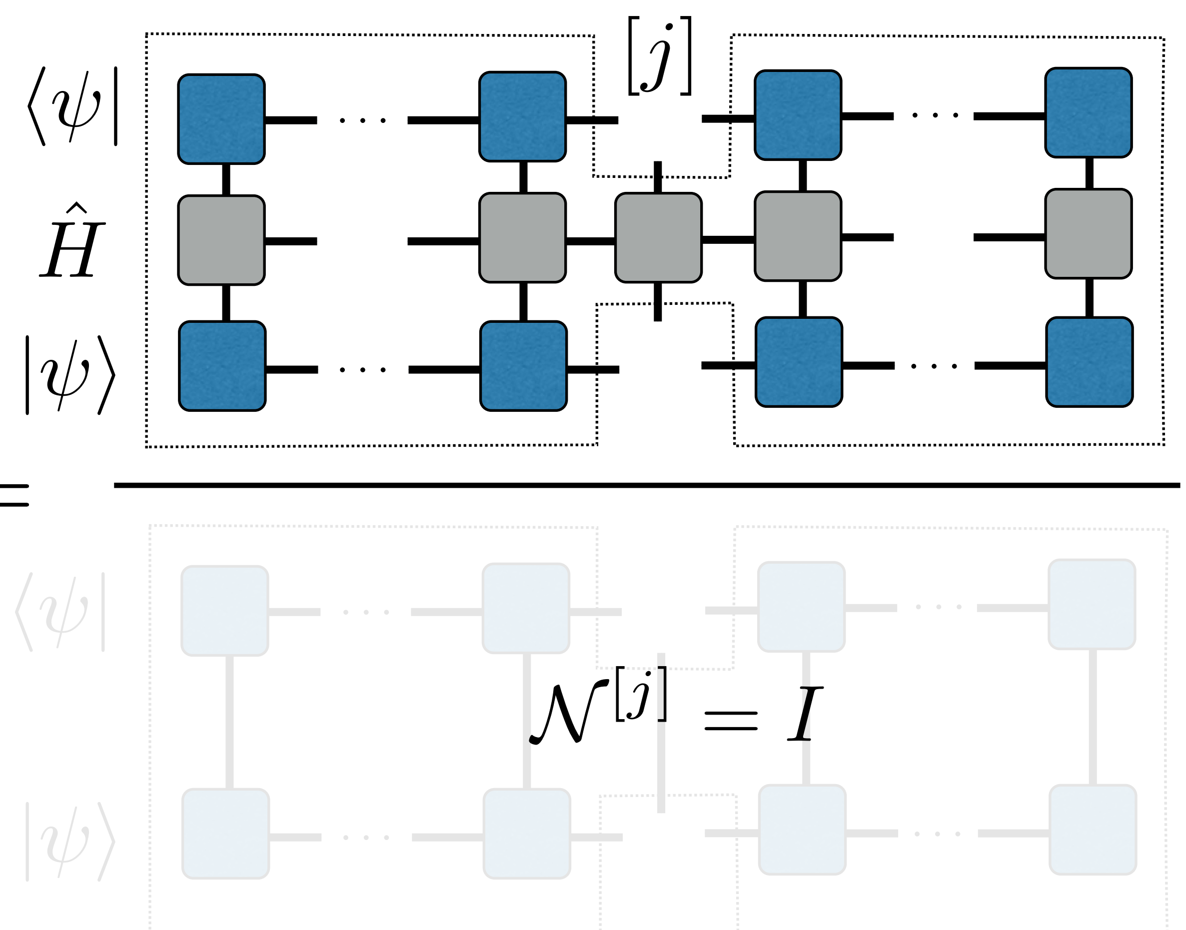
$$dim(V_j) = (a_{j-1}\sigma_j a_j) \longrightarrow min\left(\frac{V_j^{\dagger}\mathcal{H}^{[j]}V_j}{V_j^{\dagger}\mathcal{N}^{[j]}V_j}\right)$$

Algorithms with MPS/MPO: DMRG, GS variational state search

- Ground state variational search

We have to solve a generalized eigenvalue problem: $\mathcal{H}^{[j]} := \langle a_{j-1}\sigma_j a_j | \hat{H} | a_{j-1}\sigma_j a_j \rangle$

$$\frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle} =$$



$$\mathcal{N}^{[j]} = I$$

$$= \frac{\mathcal{H}^{[j]}}{\mathcal{N}^{[j]}}$$

$$\mathcal{H}^{[j]} V_j = \lambda_{min}^{(j)} \mathcal{N}^{[j]} V_j$$

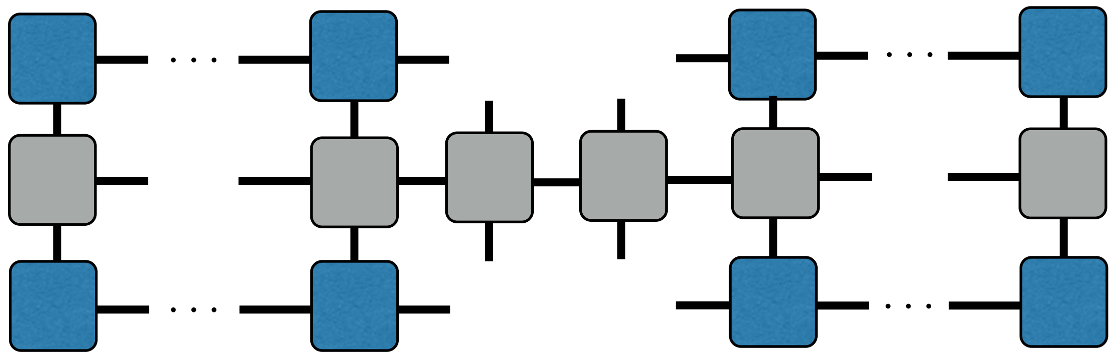
$$dim(V_j) = (a_{j-1}\sigma_j a_j) \longrightarrow min\left(\frac{V_j^\dagger \mathcal{H}^{[j]} V_j}{V_j^\dagger \mathcal{N}^{[j]} V_j}\right)$$

Using the gauge freedom we can ensure $\mathcal{N}^{[j]} = I$

Algorithms with MPS/MPO: DMRG, GS variational state search

Extensions to ground state variational search:

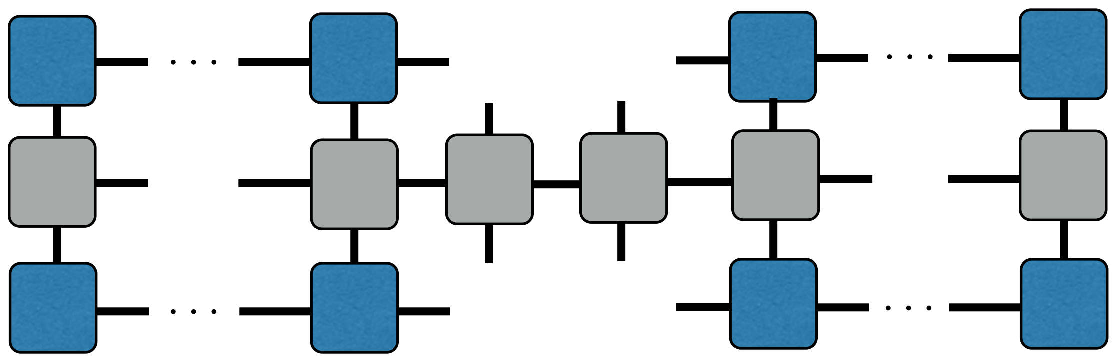
- Multisite search: we can generalize this algorithm to several sites. Our vector GS will have dimension $d^n \cdot D^2$.



Algorithms with MPS/MPO: DMRG, GS variational state search

Extensions to ground state variational search:

- Multisite search: we can generalize this algorithm to several sites. Our vector GS will have dimension $d^n \cdot D^2$.



- Excited state search: we can extend this to find some higher excited state. The simpler way is by ensuring orthogonality to the GS.

$$\mathcal{H}^{[j]} \to (1 - |\psi_0\rangle\langle\psi_0|)\mathcal{H}^{[j]}(1 - |\psi_0\rangle\langle\psi_0|)$$

Alternative ways, including the use of Lagrange multipliers during the minimization process, can also be implemented.

Algorithms with MPS/MPO: TDVP, variational state search for time evolution

- With the same spirit as in DMRG we can use single site update to find the state after a time evolution step. In first order we have:

$$|\psi(t+dt)\rangle = |\psi(t)\rangle - i\delta t\hat{H}|\psi(t)\rangle$$

- In order to optimize our evolution we consider a given region of the Hilbert space where our state lives at time *t* and then project the evolved state into this manifold (tangent space) by finding the state that minimizes:

$$F[|\psi(t)\rangle] = \|(\mathcal{I} - i\delta t\hat{H})|\psi(t)\rangle - |\phi\rangle\|^2$$

- We can do this by defining projectors onto the tangent space that act locally, doing local updates similar to Trotter or DMRG:

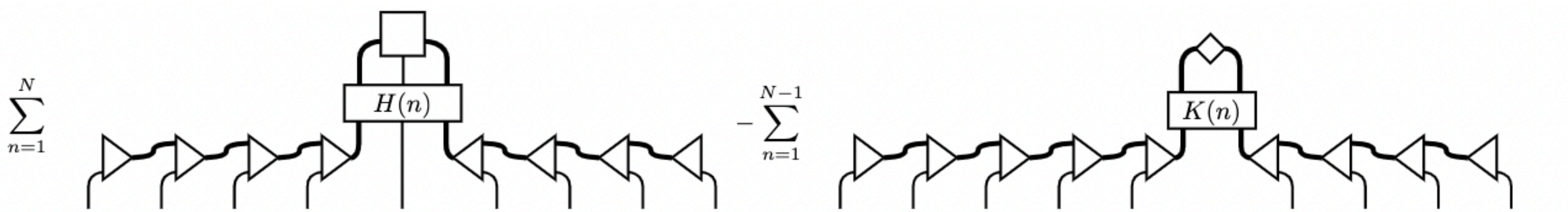


Figure 3. Right-hand side (up to the factor $-i$) of the TDVP equation [Eq. (3)].

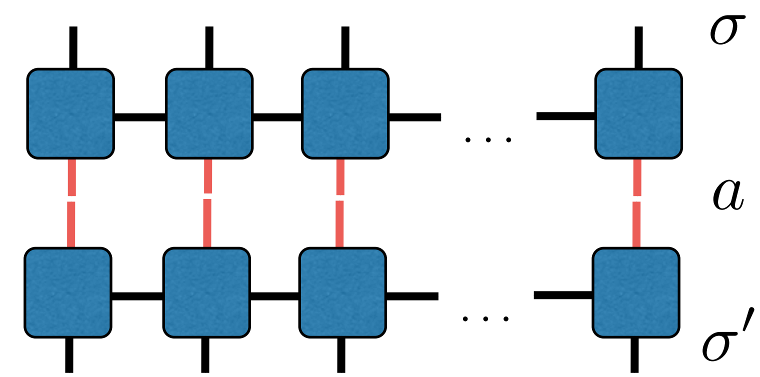
J. Haegeman et al. Physical Review Letters (2011)
J. Haegeman et al. Physical Review B (2016)

Encore (I): MPS in open quantum systems:

- We need an effective way of encoding our density operator:

$$\rho = \sum_{\vec{\sigma},\vec{\sigma}'} M^{\sigma_1\sigma'_1} M^{\sigma_2\sigma'_2} \ldots M^{\sigma_{M-1}\sigma'_{M-1}} M^{\sigma_M\sigma'_M} |\vec{\sigma}\rangle\langle\vec{\sigma}'|$$

$$M^{\sigma_i\sigma'_i} = \sum_{a_i} A^{\sigma_i,a_i} \otimes (A^{\sigma'_i,a_i})^\dagger$$



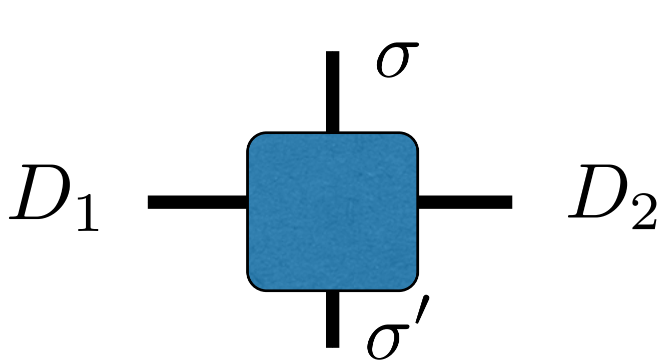
F. Verstraete, J. J. García-Ripoll, J. I. Cirac, PRL 93, 20 (2004)

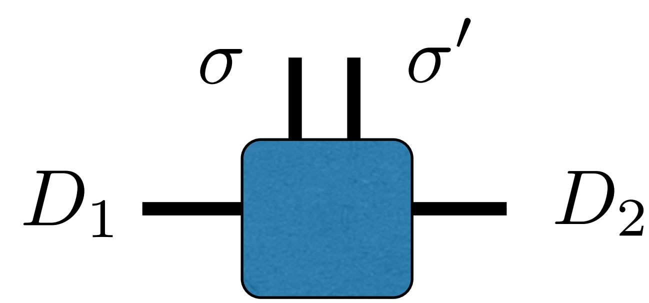
- Can we make use of all the tools we have from GS variational search?

Vectorizing our density operator into an MPS: $|\sigma_i\rangle\langle\sigma'_i| \Rightarrow |\sigma_i\sigma'_i\rangle$

(Choi's isomorphism)

$$\rho = \sum_{\sigma_i,\sigma'_i} \rho_{\sigma_i\sigma'_i} |\sigma_i\rangle\langle\sigma'_i| \qquad |\rho\rangle = \sum_{\sigma_i,\sigma'_i} \rho_{\sigma_i\sigma'_i} |\sigma_i\sigma'_i\rangle$$





J. Cui, J. I. Cirac, and M. C. Bañuls, Phys. Rev. Lett. 114, 220601 (2015)

Encore (II): MPS beyond 1D

- We can extend the notion of MPS to higher dimensions.

  Disclaimers: I) Less efficient (area law becomes worse in higher dimensions, still okay in 2D).

  II) Algorithms cannot be generally optimized but depends on the problem.

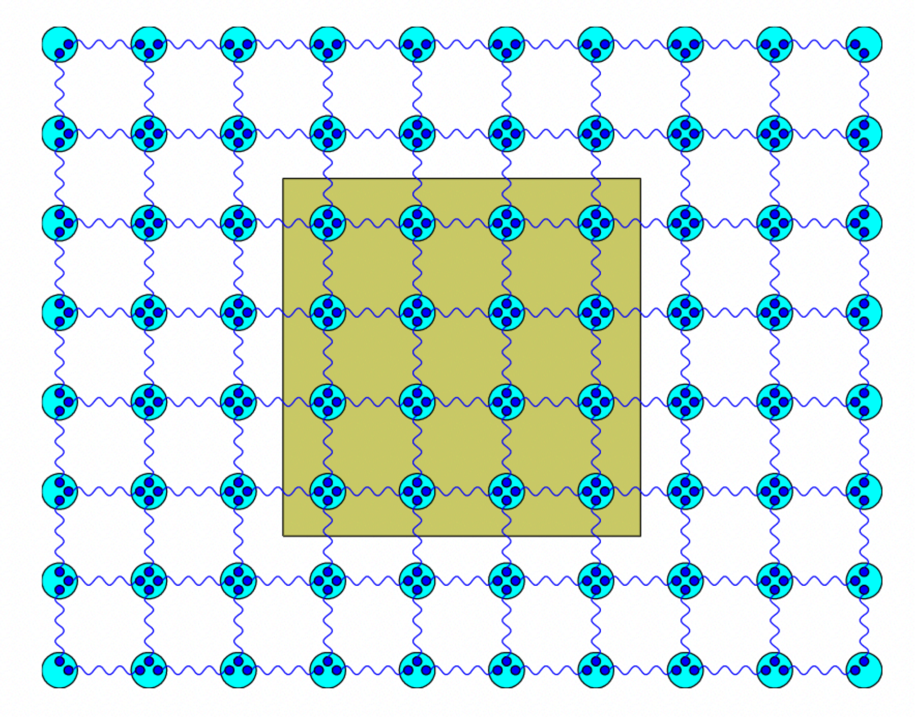
Encore (II): MPS beyond 1D

- We can extend the notion of MPS to higher dimensions.

- PEPS: projected entangled pair states

  Here every local tensor has dimension:

  $$dD^4$$ 1 phys. dimension + 4 bond dimensions

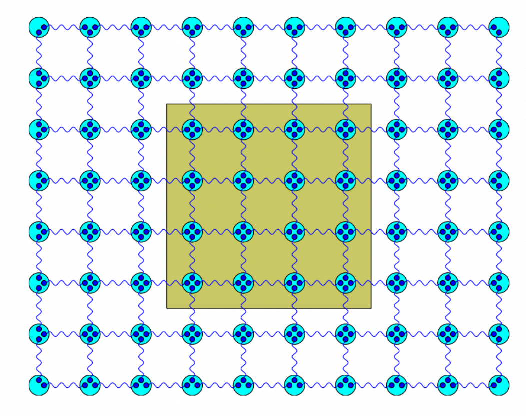


arXiv:cond-mat/0407066
F. Verstraete, M. M. Wolf, D. Perez-Garcia, and J. I. Cirac
Phys. Rev. Lett. **96**, 220601

Encore (II): MPS beyond 1D

- We can extend the notion of MPS to higher dimensions.

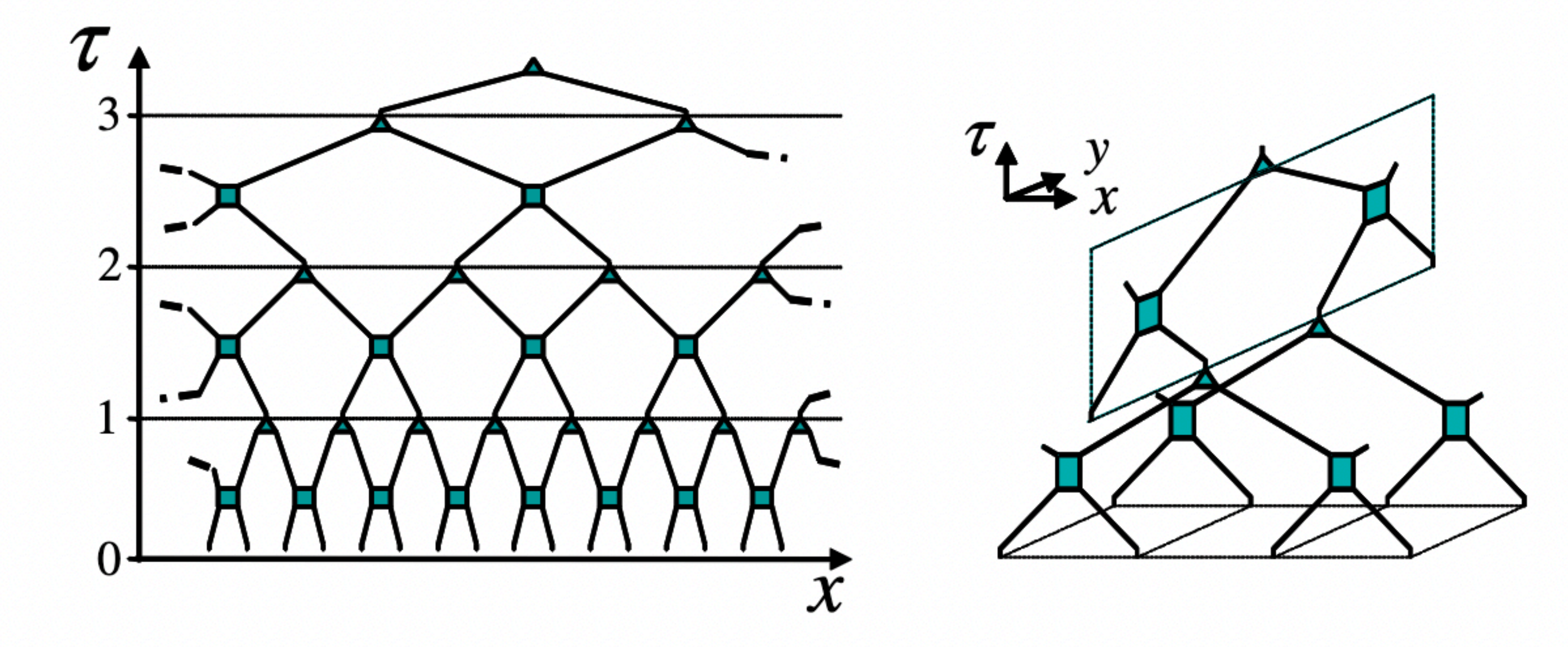
- PEPS: projected entangled pair states



arXiv:cond-mat/0407066
F. Verstraete, M. M. Wolf, D. Perez-Garcia, and J. I. Cirac
Phys. Rev. Lett. **96**, 220601

- Tree tensor networks: they loose the planar structure, no area law.

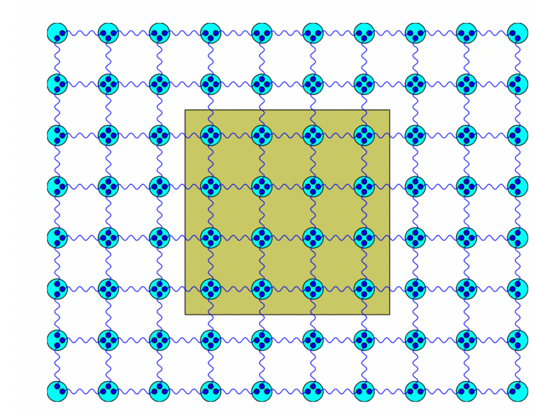
  - E.g.: MERAs, multiscale entanglement renormalization ansatz, (Vidal, PRL 99, 220405 (2007))



Encore (II): MPS beyond 1D

- We can extend the notion of MPS to higher dimensions.

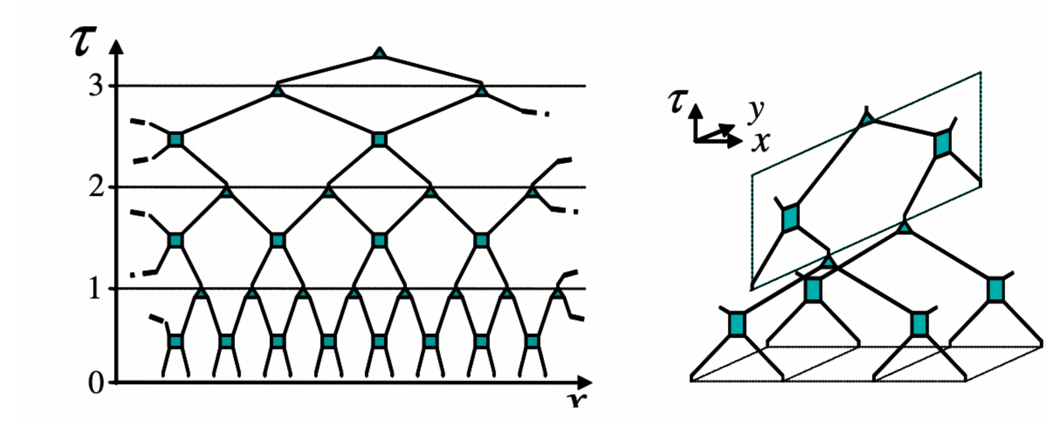
- PEPS: projected entangled pair states



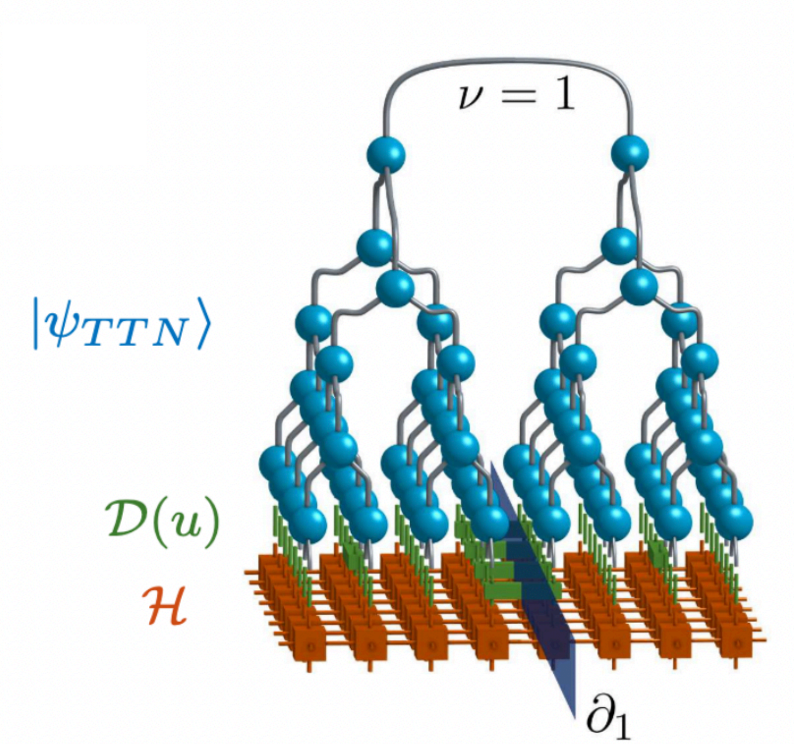
arXiv:cond-mat/0407066
F. Verstraete, M. M. Wolf, D. Perez-Garcia, and J. I. Cirac
Phys. Rev. Lett. **96**, 220601

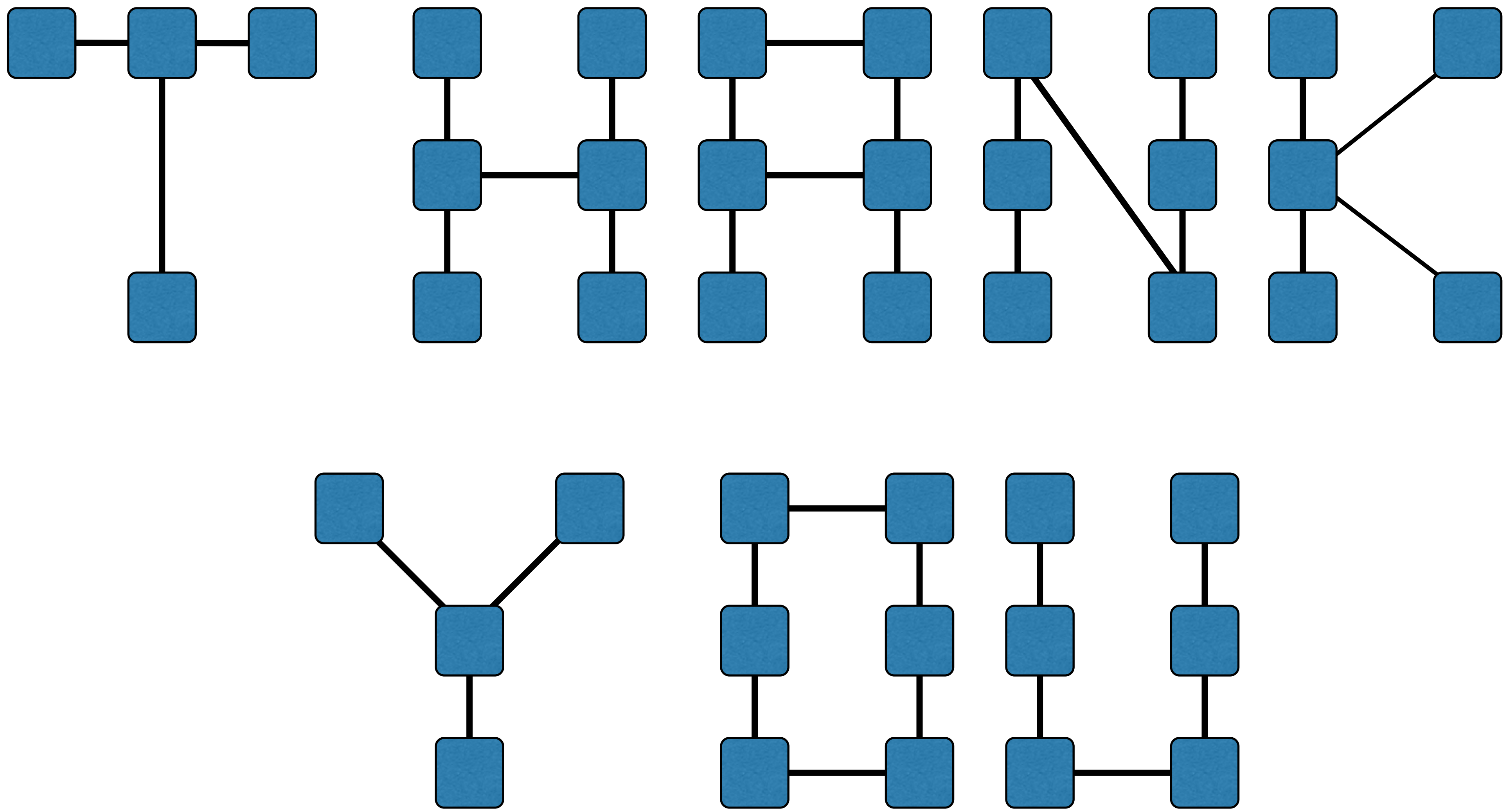
- Tree tensor networks: they loose the planar structure, no area law.

  - E.g.: MERAs, multiscale entanglement renormalization ansatz, (Vidal, PRL 99, 220405 (2007))



  - E.g.2: more recently, Montangero Phys. Rev. Lett. **126**, 170603 (2021)




THANK
YOU

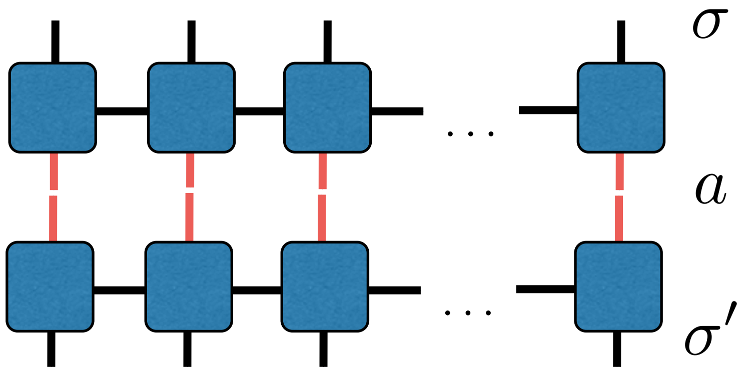
Dissipative systems into MPS language

Dissipative systems into MPS: VARIATIONAL APPROACH

- We need an effective way of encoding our density operator:

$$\rho = \sum_{\vec{\sigma}, \vec{\sigma}'} M^{\sigma_1 \sigma'_1} M^{\sigma_2 \sigma'_2} \ldots M^{\sigma_{M-1} \sigma'_{M-1}} M^{\sigma_M \sigma'_M} |\vec{\sigma}\rangle\langle\vec{\sigma}'|$$

$$M^{\sigma_i \sigma'_i} = \sum_{a_i} A^{\sigma_i, a_i} \otimes (A^{\sigma'_i, a_i})^\dagger$$



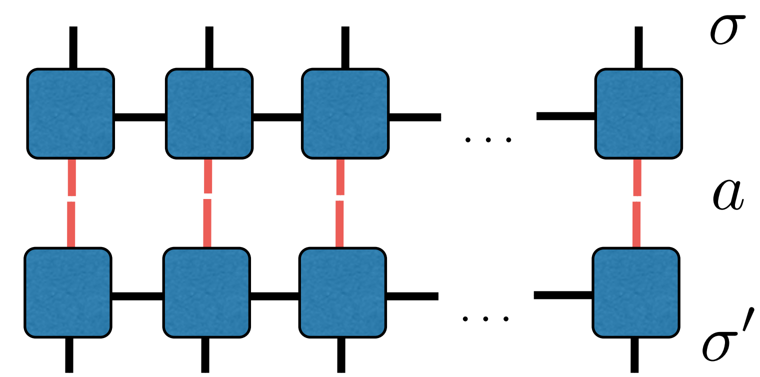
F. Verstraete, J. J. García-Ripoll, J. I. Cirac, PRL 93, 20 (2004)

Dissipative systems into MPS: VARIATIONAL APPROACH

- We need an effective way of encoding our density operator:

$$\rho = \sum_{\vec{\sigma},\vec{\sigma}'} M^{\sigma_1\sigma'_1} M^{\sigma_2\sigma'_2} \ldots M^{\sigma_{M-1}\sigma'_{M-1}} M^{\sigma_M\sigma'_M} |\vec{\sigma}\rangle\langle\vec{\sigma}'|$$

$$M^{\sigma_i\sigma'_i} = \sum_{a_i} A^{\sigma_i,a_i} \otimes (A^{\sigma'_i,a_i})^\dagger$$



F. Verstraete, J. J. García-Ripoll, J. I. Cirac, PRL 93, 20 (2004)

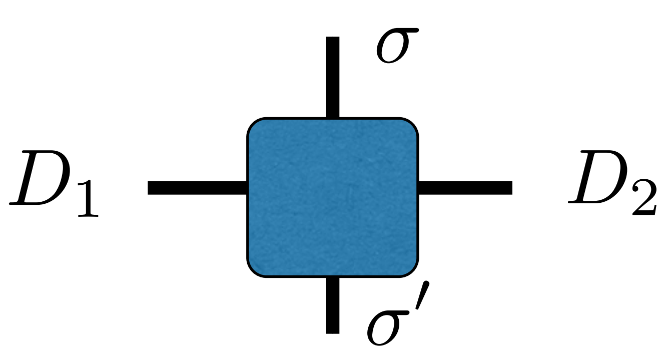
- Can we make use of all the tools we have from GS variational search?

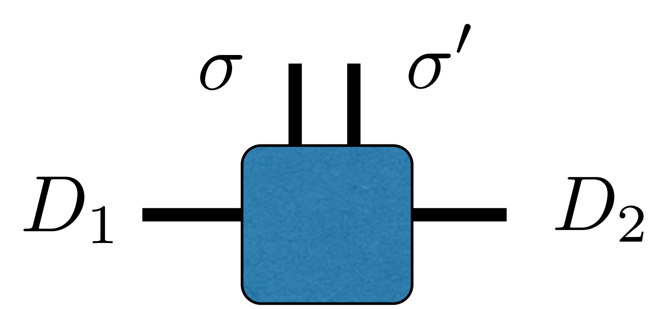
Vectorizing our density operator into an MPS: $|\sigma_i\rangle\langle\sigma'_i| \;\Rightarrow\; |\sigma_i\sigma'_i\rangle$

(Choi's isomorphism)

$$\rho = \sum_{\sigma_i,\sigma'_i} \rho_{\sigma_i\sigma'_i} |\sigma_i\rangle\langle\sigma'_i|$$

$$|\rho\rangle = \sum_{\sigma_i,\sigma'_i} \rho_{\sigma_i\sigma'_i} |\sigma_i\sigma'_i\rangle$$





J. Cui, J. I. Cirac, and M. C. Bañuls, Phys. Rev. Lett. 114, 220601 (2015)

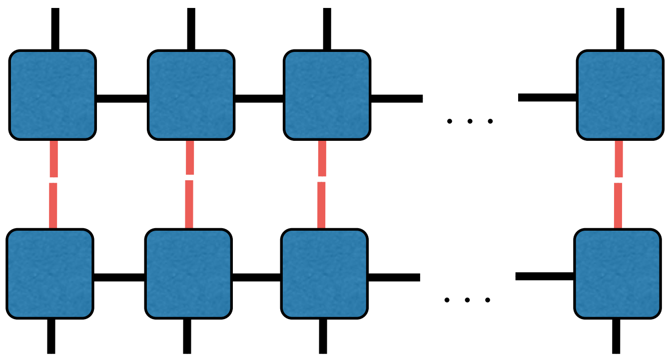
Dissipative systems into MPS: VARIATIONAL APPROACH

- We need an effective way of encoding our density operator:

We presented already the purification scheme:

$$\rho = \sum_{\vec{\sigma}, \vec{\sigma}'} M^{\sigma_1 \sigma'_1} M^{\sigma_2 \sigma'_2} \ldots M^{\sigma_{M-1} \sigma'_{M-1}} M^{\sigma_M \sigma'_M} |\vec{\sigma}\rangle\langle\vec{\sigma}'|$$

$$M^{\sigma_i \sigma'_i} = \sum_{a_i} A^{\sigma_i, a_i} \otimes (A^{\sigma'_i, a_i})^\dagger$$



F. Verstraete, J. J. García-Ripoll, J. I. Cirac, PRL 93, 20 (2004)

- Can we make use of all the tools we have from GS variational search?

Vectorizing our density operator into an MPS: $|\sigma_i\rangle \langle\sigma'_i| \;\Rightarrow |\sigma_i \sigma'_i\rangle$

(Choi's isomorphism)

$$\rho = \sum_{\sigma_i, \sigma'_i} \rho_{\sigma_i \sigma'_i} |\sigma_i\rangle \langle\sigma'_i| \qquad |\rho\rangle = \sum_{\sigma_i, \sigma'_i} \rho_{\sigma_i \sigma'_i} |\sigma_i \sigma'_i\rangle$$

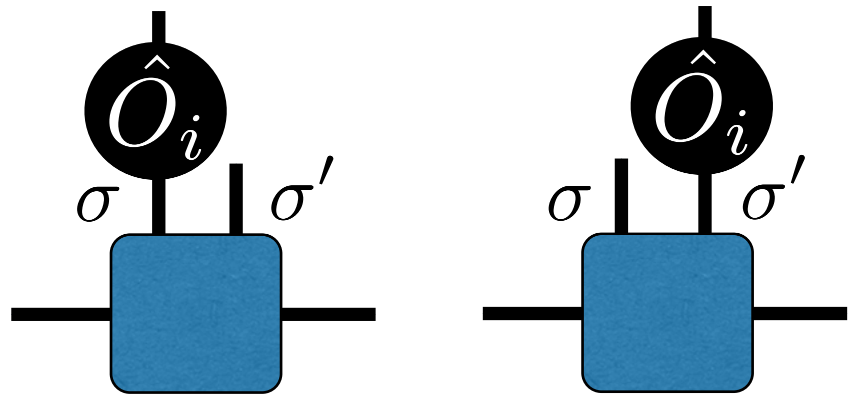
- We build our operator as an MPS living in a physical dimension d^2

J. Cui, J. I. Cirac, and M. C. Bañuls, Phys. Rev. Lett. 114, 220601 (2015)

Dissipative systems into MPS: VARIATIONAL APPROACH

- How does it affect to operators acting on it?

$$|\sigma_i \sigma_i'\rangle = |\sigma_i \otimes \sigma_i'\rangle$$



$$\hat{O}\rho \;\rightarrow\; \left(\hat{O}_\sigma \otimes \hat{I}_{\sigma'}\right)|\rho\rangle$$

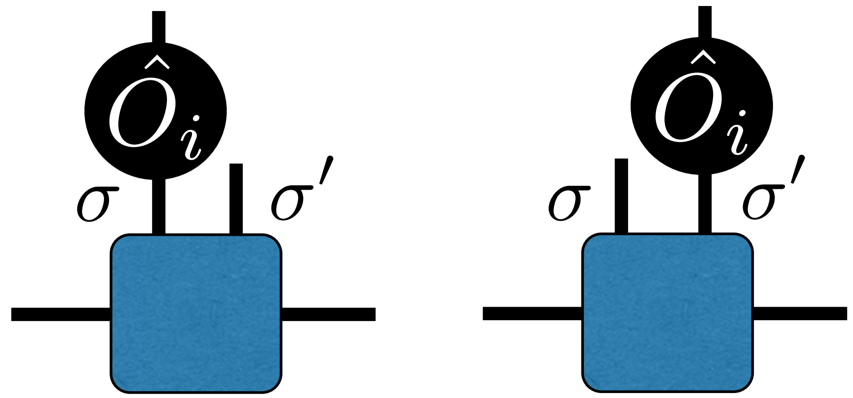
$$\rho\hat{O} \;\rightarrow\; \left(\hat{I}_\sigma \otimes \hat{O}^T_{\sigma'}\right)|\rho\rangle$$

J. Cui, J. I. Cirac, and M. C. Bañuls, Phys. Rev. Lett. 114, 220601 (2015)

Dissipative systems into MPS: VARIATIONAL APPROACH

- How does it affect to operators acting on it?

$$|\sigma_i \sigma_i'\rangle = |\sigma_i \otimes \sigma_i'\rangle$$



$$\hat{O}\rho \;\rightarrow\; \left(\hat{O}_\sigma \otimes \hat{I}_{\sigma'}\right)|\rho\rangle$$

$$\rho\hat{O} \;\rightarrow\; \left(\hat{I}_\sigma \otimes \hat{O}^T_{\sigma'}\right)|\rho\rangle$$

- This can be extended to the master equation:

$$\mathcal{L}\rho = -i\left[\hat{H},\, \rho\right] - \frac{1}{2}\gamma \sum_j (c_j^\dagger c_j \rho + \rho c_j^\dagger c_j - 2 c_j \rho c_j^\dagger)$$

$$\Downarrow$$

$$\mathcal{L} = -i(\hat{H} \otimes \hat{I} - \hat{I} \otimes \hat{H}^T) + \gamma \left[\sum_i c_i \otimes \bar{c}_i - \frac{1}{2}\left(\sum_i c_i^\dagger c_i\right) \otimes \hat{I} - \frac{1}{2}\hat{I} \otimes \left(\sum_i c_i^T \bar{c}_i\right)\right]$$

Representation in the vectorized space

J. Cui, J. I. Cirac, and M. C. Bañuls, Phys. Rev. Lett. 114, 220601 (2015)

Dissipative systems into MPS: VARIATIONAL APPROACH

- Steady state condition for $\mathcal{L}$:

$$\mathcal{L}|\rho_s\rangle = 0 \quad \Rightarrow \quad \mathcal{L}^\dagger\mathcal{L}|\rho_s\rangle = 0 \quad (\mathcal{L}^\dagger\mathcal{L} \geq 0)$$

J. Cui, J. I. Cirac, and M. C. Bañuls, Phys. Rev. Lett. 114, 220601 (2015)

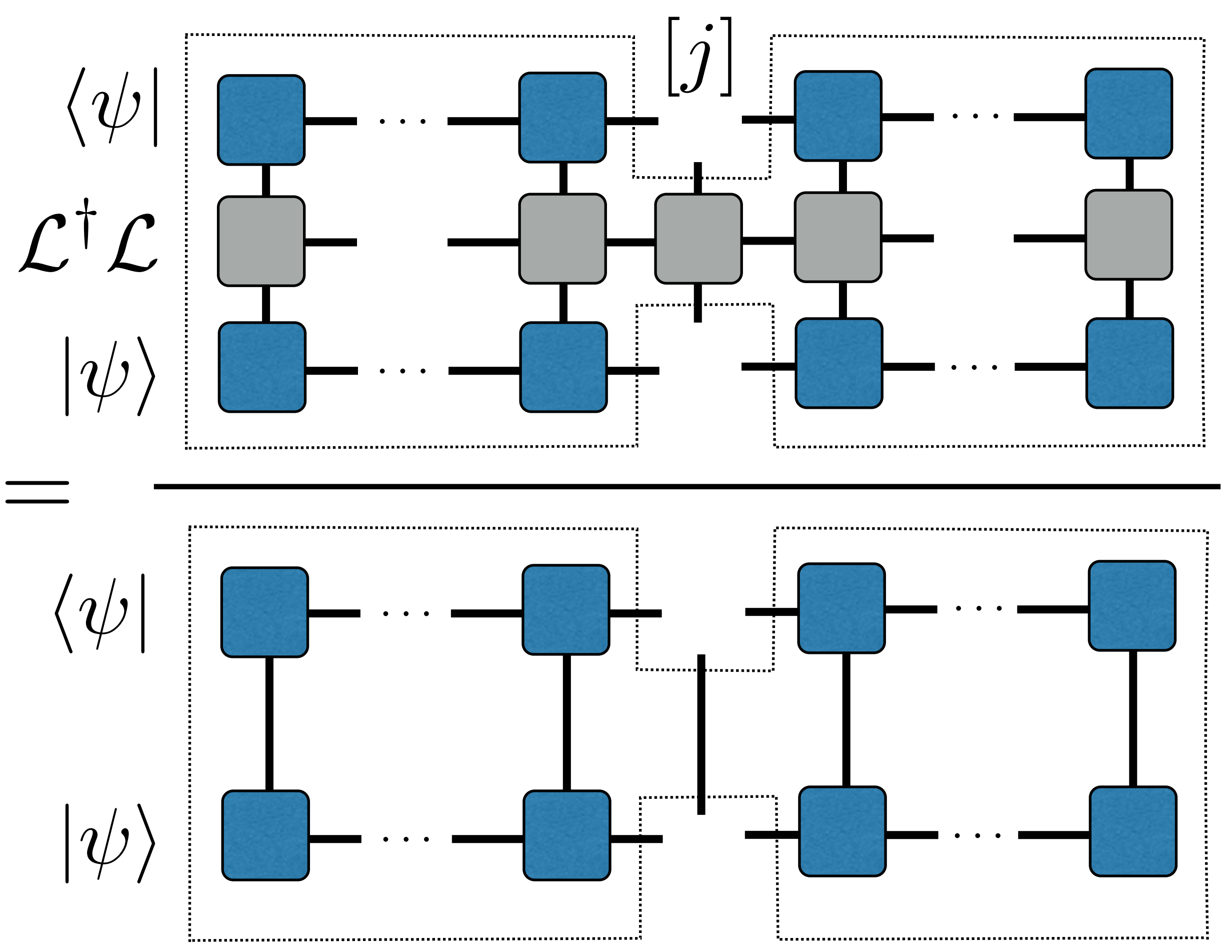
Dissipative systems into MPS: VARIATIONAL APPROACH

- Steady state condition for $\mathcal{L}$:

$$\mathcal{L}|\rho_s\rangle = 0 \quad \Rightarrow \quad \mathcal{L}^\dagger\mathcal{L}|\rho_s\rangle = 0 \qquad (\mathcal{L}^\dagger\mathcal{L} \geq 0)$$

- We can build the MPO representation for $\mathcal{L}^\dagger\mathcal{L}$ and use standard DMRG variational search

$$\frac{\langle\psi|\mathcal{L}^\dagger\mathcal{L}|\psi\rangle}{\langle\psi|\psi\rangle} =$$



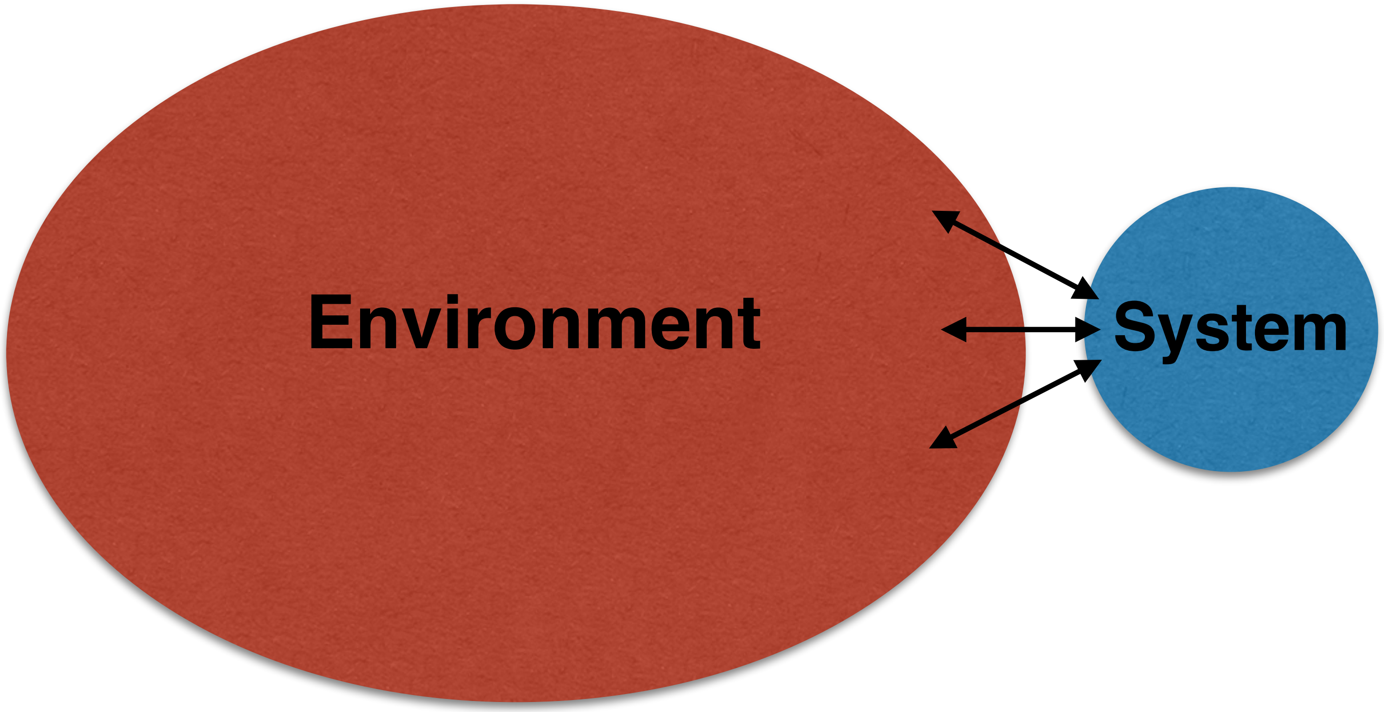
$$= \frac{\mathcal{L}^\dagger\mathcal{L}^{[j]}}{\mathcal{N}^{[j]}}$$

- It is possible to work with $\mathcal{L}$ only (see Mascarenhas et al, PRA 92, 022116 (2015))

J. Cui, J. I. Cirac, and M. C. Bañuls, Phys. Rev. Lett. 114, 220601 (2015)

Extra material

Dissipative Driving in many-body systems



Dissipative driving into many-body states

- It provides an interesting alternative to prepare states in the optical lattices.

  - Hamiltonian engineering:

$$|\psi\rangle(t) = e^{-i\hat{H}t}|\psi\rangle(0) \;\; \rightarrow |GS\rangle \; \langle GS| \;\; (\, t \rightarrow \infty \,)$$

  Relays on tunability of the interactions in the lattice.

vs

  - Reservoir engineering:

$$\dot{\rho} = \mathcal{L}\rho \quad \Rightarrow \quad \rho(t) \rightarrow \rho_{steady} \; (t \rightarrow \infty)$$

  Make use of the dissipation to assist the preparation.

  - Other alternatives: adiabatic state preparation (see last talk), …

Driven dissipative dynamics in Quantum Optics

- Dynamics described by master equation $\frac{d}{dt}\rho = \mathcal{L}\rho$

$$\mathcal{L}\rho = -i\left[\hat{H}_0,\ \rho\right] - \frac{1}{2}\sum_j \gamma_j (c_j^\dagger c_j \rho + \rho c_j^\dagger c_j - 2 c_j \rho c_j^\dagger)$$

A. J. Daley, Advances in Physics 63, 77 (2014)

Driven dissipative dynamics in Quantum Optics

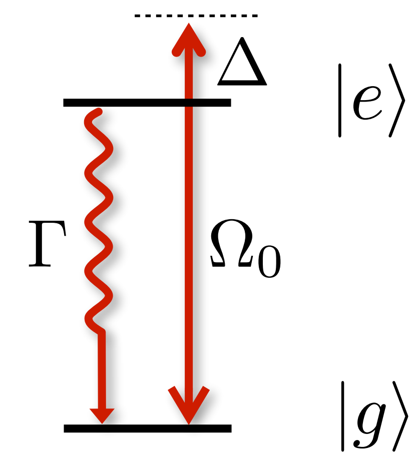
- Dynamics described by master equation $$\frac{d}{dt}\rho = \mathcal{L}\rho$$

$$\mathcal{L}\rho = -i\left[\hat{H}_0,\ \rho\right] - \frac{1}{2}\sum_j \gamma_j (c_j^\dagger c_j \rho + \rho c_j^\dagger c_j - 2 c_j \rho c_j^\dagger)$$

- Key approximations:

  - **Born approx.** : timescales of dynamics induced by S-E are much slower than system or environment ones.
  - **Markov approx.** : evolution does not depend on history of the system.

  (+ **RWA**)



$$\Gamma, \Omega_0, \Delta \ll \omega_{sys}$$

A. J. Daley, Advances in Physics 63, 77 (2014)

Driven dissipative dynamics in Quantum Optics

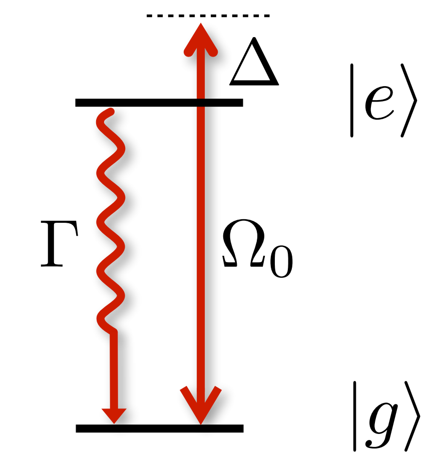
- Dynamics described by master equation $\frac{d}{dt}\rho = \mathcal{L}\rho$

$$\mathcal{L}\rho = -i\left[\hat{H}_0,\ \rho\right] - \frac{1}{2}\sum_j \gamma_j (c_j^\dagger c_j \rho + \rho c_j^\dagger c_j - 2 c_j \rho c_j^\dagger)$$

- Key approximations:

  - **Born approx.** : timescales of dynamics induced by S-E are much slower than system or environment ones.
  - **Markov approx.** : evolution does not depend on history of the system.

  (+ **RWA**)



$$\Gamma, \Omega_0, \Delta \ll \omega_{sys}$$

- A **pure steady state** should be a unique eigenstate of H and an eigenstate of all c operators with zero eigenvalue, i.e. a **dark state**

$$H|\psi_{ss}\rangle = \alpha_{ss}|\psi_{ss}\rangle \qquad \forall c_m, \quad c_m|\psi_{ss}\rangle = 0$$

A. J. Daley, Advances in Physics 63, 77 (2014)

Quantum Trajectories

- Master Equation $$\dot{\rho} = -i[H,\rho] - \frac{1}{2}\sum_m \left[c_m^\dagger c_m \rho + \rho c_m^\dagger c_m - 2c_m \rho c_m^\dagger\right]$$ $$= -i(H_{\text{eff}}\rho - \rho H_{\text{eff}}^\dagger) + \sum_m c_m \rho c_m^\dagger \qquad H_{\text{eff}} = H - \frac{i}{2}\sum_m c_m^\dagger c_m$$

- Single trajectory: Initial state $|\phi(0)\rangle$ sampled from ρ

- Single trajectory: Propagation 1: Calculate evolution under effective Hamiltonian

$$|\phi^{(1)}(t+\delta t)\rangle = (1 - iH_{\text{eff}}\delta t)\,|\phi(t)\rangle$$

$$\begin{aligned}
\langle\phi^{(1)}(t+\delta t)|\phi^{(1)}(t+\delta t)\rangle &= \langle\phi(t)|\,(1+iH_{\text{eff}}\delta t)\,(1-iH_{\text{eff}}\delta t)\,|\phi(t)\rangle \\
&= 1 - \delta p
\end{aligned}$$

$$\begin{aligned}
\delta p &= \delta t\langle\phi(t)|i(H_{\text{eff}} - H_{\text{eff}}^\dagger)|\phi(t)\rangle \\
&= \delta t \sum_m \langle\phi(t)|c_m^\dagger c_m|\phi(t)\rangle = \sum_m \delta p_m
\end{aligned}$$

H. Carmichael, *An Open Systems Approach to Quantum Optics*, Springer, Berlin, 1993.
K. Mølmer, J. Dalibard, Y. Castin, JOSA B **10**, 524 (1993)
R. Dum, A. S. Parkins, P. Zoller, and C. W. Gardiner, PRA **46**, 4382 (1992)

$$\delta p = \delta t \sum_m \langle \phi(t)|c_m^\dagger c_m|\phi(t)\rangle = \sum_m \delta p_m$$

- Propagation 2: $|\phi(t)\rangle$
  - With probability $1 - \delta p$ $\quad |\phi(t+\delta t)\rangle = \dfrac{|\phi^{(1)}(t+\delta t)\rangle}{\sqrt{1-\delta p}}$
  - With probability δp $\quad |\phi(t+\delta t)\rangle = \dfrac{c_m|\phi(t)\rangle}{\sqrt{\delta p_m/\delta t}}$ choose *m* with probability: $\Pi_m = \delta p_m/\delta p$

- Stochastic average: Form density operator $\sigma(t) = |\phi(t)\rangle\langle\phi(t)|$

$$\overline{\sigma(t+\delta t)} = (1-\delta p)\frac{|\phi^{(1)}(t+\delta t)\rangle}{\sqrt{1-\delta p}}\frac{\langle\phi^{(1)}(t+\delta t)|}{\sqrt{1-\delta p}} + \delta p \sum_m \Pi_m \frac{c_m|\phi(t)\rangle}{\sqrt{\delta p_m/\delta t)}}\frac{\langle\phi(t)|c_m^\dagger}{\sqrt{\delta p_m/\delta t)}}$$

$$\begin{aligned}\overline{\sigma(t+\delta t)} &= \sigma(t) - i\delta t(H_{\text{eff}}\sigma(t) - \sigma(t)H_{\text{eff}}^\dagger) \\ &\quad + \delta t \sum_m c_m \sigma(t) c_m^\dagger\end{aligned}$$

- Stochastic average equivalent to master equation $\dot{\rho} = -i(H_{\text{eff}}\rho - \rho H_{\text{eff}}^\dagger) + \sum_m c_m \rho c_m^\dagger$