Computations in Quantum Tensor Networks

Thomas Huckle

Thomas Schulte-Herbrüggen, Konrad Waldherr

Bonn, 4.8.2011

Overview

(1) Problem setting: Computation of Ground States

Physical model and math. description

(2) Efficient Representation of Vectors as Tensors:

(a) Matrix Product States (MPS) and Tensor Trains (TT):

Description

Computations/Contractions

Normalizations (SVD, DMRG)

(b) Matrix Product Operators (MPO)

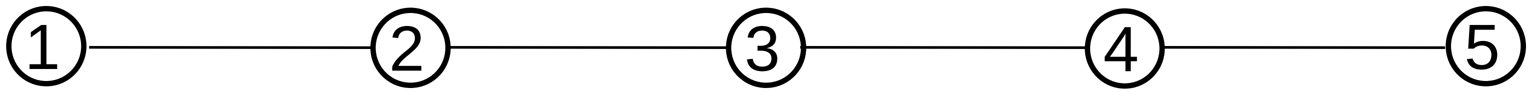
(c) Utilizing symmetries in the vector representation

(d) Krylov methods for eigenvectors in MPS representation

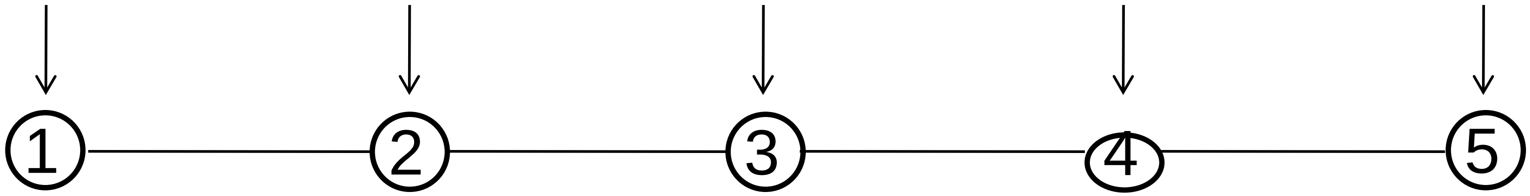
1. Computation of ground states

Physical system with N particles (here 1D spin chain)

Interaction within the system (e.g. nearest-neighbor interaction)



External interaction (e.g. exterior magnetic field)



Goal:
Find ground state – smallest energy level of the system – minimum eigenvalue/vector

Quantum system described by Hamiltonian operator H.

The real eigenvalues of H are the energy levels of stationary states described by the related eigenvectors:

$$H\,|\psi_k> \;=\; E_k\,|\psi_k>$$

Any state is represented by a vector $x \in C^{2^N}$

Hamiltonian matrix $H \in C^{2^N \times 2^N}$

Hamiltonian H may be formulated as a weighted sum of Kronecker products of Pauli matrices (2 x 2 Hermitian, unitary matrices).

Pauli matrices as spin operators:

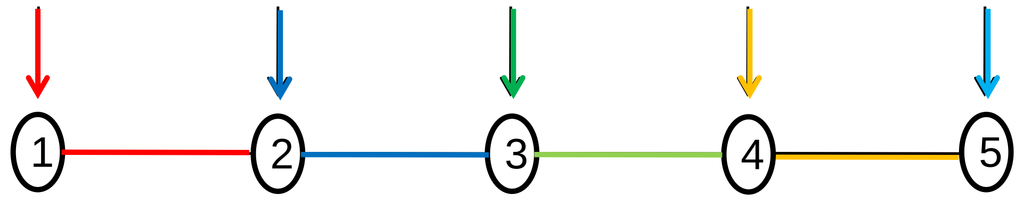
$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

Typical spin vectors x for spin up: $|\uparrow>, \quad |1>, \quad \begin{pmatrix} 1 \\ 0 \end{pmatrix},$

$$|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad |ij\rangle = |i\rangle\,|j\rangle = |i\rangle \otimes |j\rangle$$

Kronecker product <-> Tensor product <-> $\otimes$

Example: Ising-type Hamiltonian



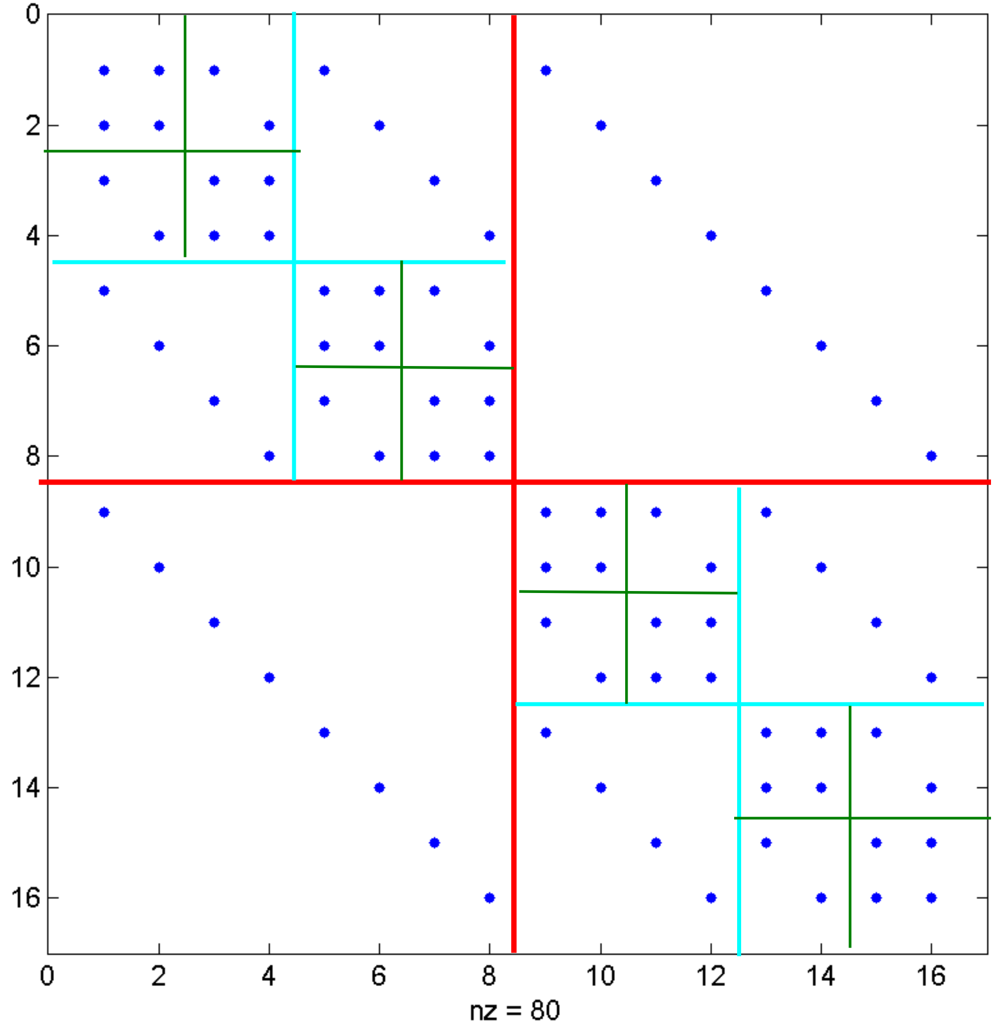
$$
\begin{aligned}
H = {} & \sigma_z \otimes \sigma_z \otimes I \otimes I \otimes I \\
+ {} & I \otimes \sigma_z \otimes \sigma_z \otimes I \otimes I \\
+ {} & I \otimes I \otimes \sigma_z \otimes \sigma_z \otimes I \\
+ {} & I \otimes I \otimes I \otimes \sigma_z \otimes \sigma_z \\
\hline
+ {} & \sigma_x \otimes I \otimes I \otimes I \otimes I \\
+ {} & I \otimes \sigma_x \otimes I \otimes I \otimes I \\
+ {} & I \otimes I \otimes \sigma_x \otimes I \otimes I \\
+ {} & I \otimes I \otimes I \otimes \sigma_x \otimes I \\
+ {} & I \otimes I \otimes I \otimes I \otimes \sigma_x
\end{aligned}
$$

Typical Pattern

Sparsity:
O(n log(n))

Structured:
constant
along
diagonals

1D spin chain, control Hamiltonian



General Hamiltonian

$$\sum_{k=1}^{M} \alpha_k Q_1^{(k)} \otimes Q_2^{(k)} \otimes \cdots \otimes Q_N^{(k)}$$

$$Q_j^{(k)} \in \{I_2, \sigma_x, \sigma_y, \sigma_z\} : Pauli\ \ matrices$$

Problem: For N=50 eigenvector has 2^{50} components!

Solution: Find suitable vectors x with „sparse“ representation that allow

- good approximations of the eigenvector we are looking for
- easy computations Hx, $x^H y$ for numerical eigenvalue computations (Rayleigh Quotient, Vector Iteration,….

In suitable subset solve $\min \frac{x^T H_{eff}\, x}{x^T N_{eff}\, x}$

Typical Hamiltonians I
Open Boundary Conditions OBC

Notation: $P_x = \sigma_x, \quad S_x = \frac{\hbar}{2} P_x, \quad \vec{S} = \begin{pmatrix} S_x \\ S_y \\ S_z \end{pmatrix}, \quad P_{x,i}$ at site i

$$H_x = \sum_{i=1}^{N} I_{2^{i-1}} \otimes (P_x)_i \otimes I_{2^{N-i}} = \sum_{i=1}^{N} P_{x,i}$$

OBC: $$H_{xx} = \sum_{i=1}^{N-1} I_{2^{i-1}} \otimes (P_x)_i \otimes (P_x)_{i+1} \otimes I_{2^{N-i-1}} = \sum_{i=1}^{N-1} P_{x,i} P_{x,i+1}$$

$$H_{yy} = \sum_{i=1}^{N-1} P_{y,i} P_{y,i+1}, \qquad H_{zz} = \sum_{i=1}^{N-1} P_{z,i} P_{z,i+1},$$

$$\sum_{i=1}^{N-2} I_{2^{i-1}} \otimes (P_x)_i \otimes I_2 \otimes (P_x)_{i+2} \otimes I_{2^{N-i-2}} = \sum_{i=1}^{N-2} P_{x,i} P_{x,i+2}$$

Typical Hamiltonians II

Periodic Boundary Conditions PBC: $H'_{xx} = \sum_{i=1}^{N} P_{x,i} P_{x,i+1 \bmod N}$

		Heisenberg model (generalized)
isotropic	Heisenberg-XX	$J_X H_{xx} + J_X H_{yy}$
anisotropic	Heisenberg-XY	$J_X H_{xx} + J_Y H_{yy}$
	Heisenberg-XZ	$J_X H_{xx} + J_Z H_{zz}$
isotropic	Heisenberg-XXX	$J_X H_{xx} + J_X H_{yy} + J_X H_{zz}$
	Heisenberg-XXZ	$J_X H_{xx} + J_X H_{yy} + J_Z H_{zz}$
	Heisenberg-XYZ	$J_X H_{xx} + J_Y H_{yy} + J_Z H_{zz}$

(all of the above, grouped by a brace) $+ \lambda H_x$

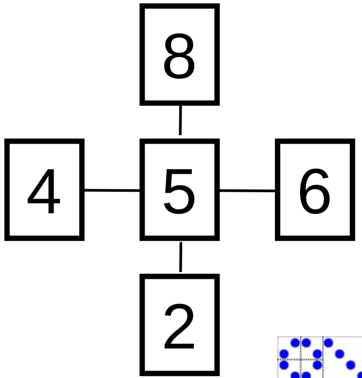
Typical Hamiltonians III
Bilinear biquadratic - AKLT

AKLT model: $$H = \sum_i \vec{S}_i \vec{S}_{i+1} + \frac{1}{3}\left(\vec{S}_i \vec{S}_{i+1}\right)^2$$

Bilinear biquadratic: $$H = \sum_i \left(\cos(\theta)\vec{S}_i \vec{S}_{i+1} + \sin(\theta)\left(\vec{S}_i \vec{S}_{i+1}\right)^2\right)$$

2D: Summation over „neighbors“ with index i,j:

$$H = \sum_{i,j:neighbors} \vec{S}_i \vec{S}_j + \frac{1}{3}\left(\vec{S}_i \vec{S}_j\right)^2$$



2. Sparse and efficient representation/approximation of vector x:

Consider vector x as binary tensor:

$$x = (x_i)_{i=0,\dots,2^N-1} = (x_{i_1 \dots i_N})_{i_j=0,1}$$

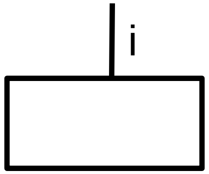
via binary representation of index i.
Reshape!

$$|i\rangle = |i_1 i_2 \dots i_N\rangle = |i_1\rangle |i_2\rangle \dots |i_N\rangle = |i_1\rangle \otimes |i_2\rangle \otimes \dots \otimes |i_N\rangle$$

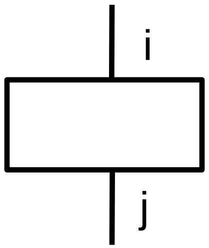
$$x = \sum_i x_i \, |i> = \sum_{i_1,\dots,i_N} x_{i_1 \dots i_N} \, |i_1> \otimes \cdots \otimes |i_N>$$

Graphical Notation

Vector (1 leg): $x_i \leftrightarrow$

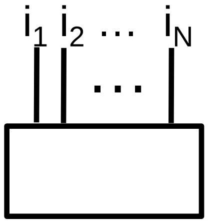


Matrix (2 legs): $(a_{ij})_{i,j} \leftrightarrow$



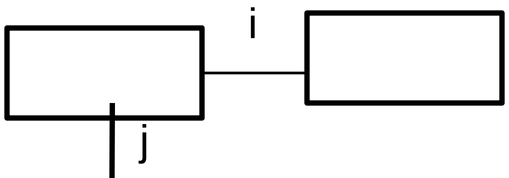
General tensor with N legs

$x_{i_1 \ldots i_N} \leftrightarrow$



Matrix-vector product – contraction over index i:

$(a_{ij})_{i,j} \cdot (x_i)_i = (y_j)_j \leftrightarrow$



First Subset of tensor approx:

Consider all vectors of the form

$$x = \begin{pmatrix} a_1 \\ b_1 \end{pmatrix} \otimes \begin{pmatrix} a_2 \\ b_2 \end{pmatrix} \otimes \ldots \otimes \begin{pmatrix} a_N \\ b_N \end{pmatrix} = x_1 \otimes x_2 \otimes \ldots \otimes x_N , \qquad a_j, b_j \in R, i = 1, \ldots, N$$

Inner product: $y^H x = \prod_{j=1}^{N} y_j^H x_j$ O(N)

Matrix-vector product: $Hx =$

$$= \left(\sum_{k=1}^{M} \alpha_k Q_1^k \otimes \ldots \otimes Q_N^k \right) \cdot \left(x_1 \otimes \ldots \otimes x_N \right)$$

$$= \sum_{k=1}^{M} \alpha_k \left(Q_1^k x_1 \right) \otimes \ldots \otimes \left(Q_N^k x_N \right)$$

Costs: =(NM), but unsatisfactory approximation property.
CP bad approx., Tucker cannot be applied.

(a) Matrix Product States
Tensor Train - Approximations

Quantum Physics: Verstraete, Schollwöck,…,
Mathematics: Tyrtyshnikov, Oseledets,…

Use rank-1 terms like CP but in a linked form that
- reflects the underlying Physics
- allows fast computations

$$x = \sum_{m_2,\dots,m_N}^{D} \begin{pmatrix} a_{1,0,m_2} \\ a_{1,1,m_2} \end{pmatrix} \otimes \begin{pmatrix} a_{2,0,m_2 m_3} \\ a_{2,1,m_2 m_3} \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} a_{N,0,m_N} \\ a_{N,1,m_N} \end{pmatrix} =$$

$$= \sum_{m_2,\dots,m_N}^{D} x_{1,m_2} \otimes x_{2,m_2 m_3} \otimes \dots \otimes x_{N,m_N},$$

Matrix Product States, cont.

$$x = \sum_{m_2,\dots,m_N}^{D} \begin{pmatrix} a_{1,0,m_2} \\ a_{1,1,m_2} \end{pmatrix} \otimes \begin{pmatrix} a_{2,0,m_2 m_3} \\ a_{2,1,m_2 m_3} \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} a_{N,0,m_N} \\ a_{N,1,m_N} \end{pmatrix} =$$

$$= \sum_{m_2,\dots,m_N}^{D} x_{1,m_2} \otimes x_{2,m_2 m_3} \otimes \dots \otimes x_{N,m_N},$$

Summation overlap reflects neighborhood relation in spin chain

$$x_{i_1 i_2 \cdots i_N} = \sum_{m_2,\cdots,m_N} a_{1 i_1, m_2} \cdot a_{2 i_2, m_2 m_3} \cdot \cdots \cdot a_{N i_N, m_N} =$$

$$= a_{1 i_1}^{H} \cdot A_{2 i_2} \cdot \cdots \cdot A_{N-1 i_{N-1}} \cdot a_{N i_N}$$

Matrix Product States, cont.

Periodic boundary conditions (Tensor chains):

$$x = \sum_{m_1,\dots,m_N}^{D} \begin{pmatrix} a_{1,0,m_1 m_2} \\ a_{1,1,m_1 m_2} \end{pmatrix} \otimes \begin{pmatrix} a_{2,0,m_2 m_3} \\ a_{2,1,m_2 m_3} \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} a_{N,0,m_N m_1} \\ a_{N,1,m_N m_1} \end{pmatrix} =$$

$$= \sum_{m_1,\dots,m_N}^{D} x_{1,m_1 m_2} \otimes x_{2,m_2 m_3} \otimes \dots \otimes x_{N,m_N m_1},$$

$$x_{i_1 i_2 \cdots i_N} = \sum_{m_1,\cdots,m_N} a_{1 i_1, m_1 m_2} \cdot a_{2 i_2, m_2 m_3} \cdot \cdots \cdot a_{N i_N, m_N m_1} =$$

$$= trace\left(A_{1 i_1} \cdot A_{2 i_2} \cdot \cdots \cdot A_{N i_N} \right)$$

For exact representation of x one needs larger $A_{j i_j}$!

We are only interested in small matrix sizes D and approximations!

i_0 $\quad i_1 \quad \ldots \quad i_{N-1} \quad i_N$

i=0, i=1:

$$\begin{bmatrix} a_{1,0}^H \\ \hline a_{1,1}^H \end{bmatrix} \cdot \begin{bmatrix} A_{2,0} \\ \hline A_{2,1} \end{bmatrix} \cdot \cdot \cdot \cdot \cdot \begin{bmatrix} A_{N-1,0} \\ \hline A_{N-1,1} \end{bmatrix} \cdot \begin{bmatrix} a_{N,0} & \big| & a_{N,1} \end{bmatrix}$$

$$x_{0,0,\ldots,0}$$

$$x_{1,0,\ldots,0}$$

$$\vdots$$

$$x_{1,1,\ldots,1}$$

$$x = \sum_{m_1,\dots,m_N}^{D} \begin{pmatrix} a_{1,0,m_1m_2} \\ a_{1,1,m_1m_2} \end{pmatrix} \otimes \begin{pmatrix} a_{2,0,m_2m_3} \\ a_{2,1,m_2m_3} \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} a_{N,0,m_Nm_1} \\ a_{N,1,m_Nm_1} \end{pmatrix} =$$

$$= \sum_{m_1,\dots,m_N}^{D} x_{1,m_1m_2} \otimes x_{2,m_2m_3} \otimes \dots \otimes x_{N,m_Nm_1},$$

m2 m3 m1

$$\sum_{m_1 \dots m_N}^{D} \; m_1 \begin{bmatrix} x_{1,11} & \cdots & x_{1,1D} \\ \vdots & & \vdots \\ x_{1,D1} & \cdots & x_{1,DD} \end{bmatrix} \text{'}\otimes\text{'} \begin{bmatrix} x_{2,11} & \cdots & x_{2,1D} \\ \vdots & & \vdots \\ x_{2,D1} & \cdots & x_{2,DD} \end{bmatrix} \text{'}\otimes\text{'} \dots \text{'}\otimes\text{'} \begin{bmatrix} x_{N,11} & \cdots & x_{N,1D} \\ \vdots & & \vdots \\ x_{N,D1} & \cdots & x_{N,DD} \end{bmatrix} m_N$$

$m_1=1$, $m_2=D$ $m_3=1$, and so on.

For notation see Khoromskji/Kazeev: „rank core product“: $\bowtie$

Core Tensors $A_{j,ij}$:

$$trace\begin{bmatrix} \cdots & A_{j0} & \cdots \\ \cdots & A_{j1} & \cdots \end{bmatrix} \qquad \sum_{m_1,\dots,m_N} \cdots \otimes \begin{pmatrix} a_{j,0,m_j m_{j+1}} \\ a_{j,1,m_j m_{j+1}} \end{pmatrix} \otimes \cdots$$

$$\begin{pmatrix} a_{j0,1,1} & & a_{j0,1,D} \\ & \ddots & \\ a_{j0,D,1} & & a_{j0,D,D} \end{pmatrix}$$

$$\begin{pmatrix} a_{j1,1,1} & & a_{j1,1,D} \\ & \ddots & \\ a_{j1,D,1} & & a_{j1,D,D} \end{pmatrix}$$

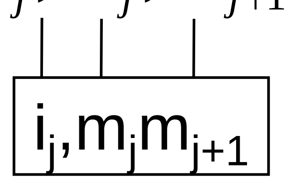
For the computation of $x^H y$ we need the inner product of two MPS vectors:

$$\sum_{i_1\ldots i_N;m_1\ldots m_N;k_1\ldots k_N} \left(\bar{b}_{1i_1,k_1k_2}\cdot \bar{b}_{2i_2,k_2k_3}\cdots\cdots \bar{b}_{pi_p,k_pk_1}\right)\cdot\left(a_{1i_1,m_1m_2}\cdot \boxed{a_{2i_2,m_2m_3}}\cdots\cdots a_{pi_p,m_pm_1}\right)$$

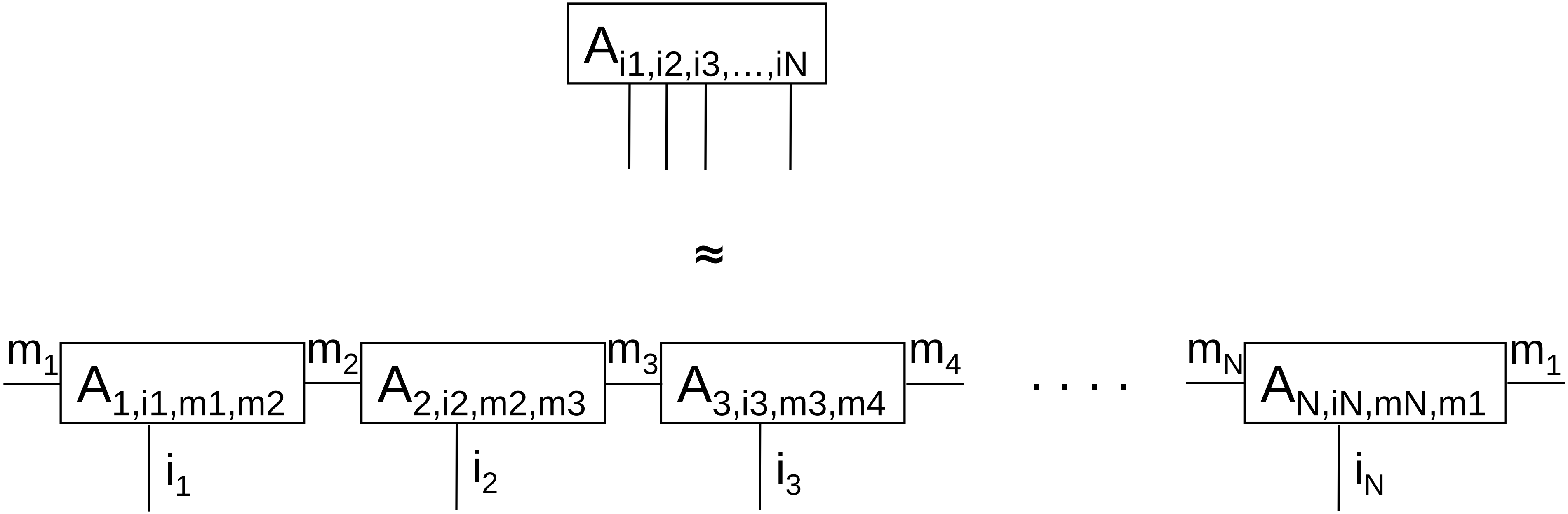
Here, we have to decide about the order of the summation (contractions).

Therefore, we represent the single factors in the above sum as small tensors with three legs (indices) structured by indices that appear in two different tensor factors.

$$a_{j;i_j;m_jm_{j+1}} = a_j(i_j, m_j, m_{j+1}), \qquad i_j = 0,1; \quad m_j, m_{j+1} = 1,\ldots,D;$$



MPS graphical



$$x_{i_1 i_2 \cdots i_N} = \sum_{m_1, \cdots, m_N} A_{1 i_1, m_1 m_2} \cdot A_{2 i_2, m_2 m_3} \cdot \cdots \cdot A_{N i_N, m_N m_1} =$$

$$= trace\left(A_{1 i_1} \cdot A_{2 i_2} \cdot \cdots \cdot A_{N i_N}\right)$$

MPS Normalization

$$trace\left(A_{1i_1} \cdot A_{2i_2} \cdot \cdots \cdot A_{Ni_N}\right)$$

MPS representation is not unique.
Between matrix products we can insert $X_i X_i^{-1}$ without changing the vector components.

Transforming the matrices into unitary matrices via SVD.
Combine the two matrices at position j into rectangular block matrix:

$$\begin{pmatrix} A_{j,0} \\ A_{j,1} \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} A_{j,0} & A_{j,1} \end{pmatrix}$$

Unfolding or matricisation of 3-leg tensor A_j

Replace the two matrices A_{j,i_j} by parts of a unitary matrix:

Compute SVD: $$\begin{pmatrix} A_{j,0} \\ A_{j,1} \end{pmatrix} = \begin{pmatrix} U_{j,0} \\ U_{j,1} \end{pmatrix} \cdot \Lambda_j \cdot V_j$$

Replace A_j matrices by U_j.
Multiply the ΛV part on the right neighboring pair A_{j+1}.

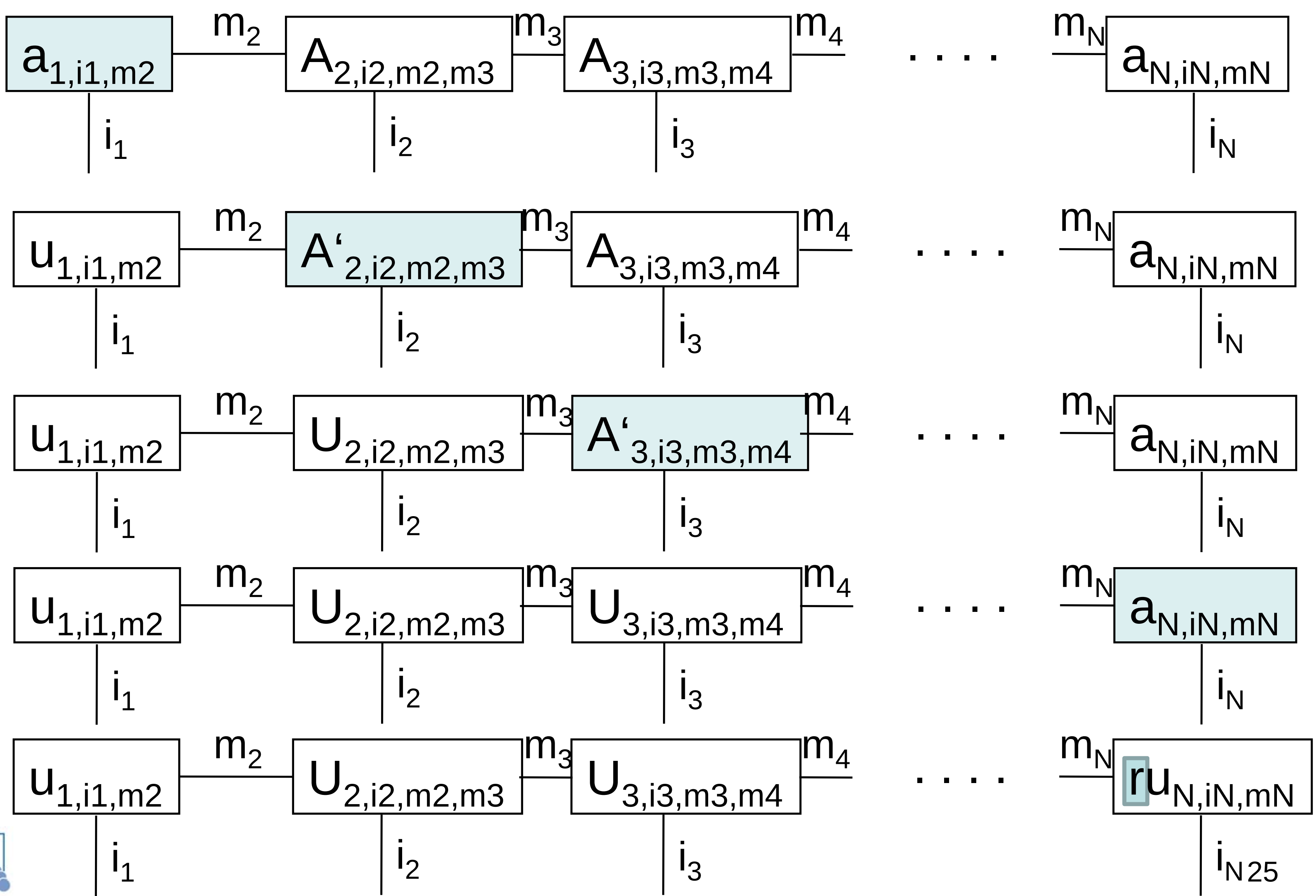
In the same way we can consider the SVD

$$\begin{pmatrix} A_{j,0} & A_{j,1} \end{pmatrix} = V_j \cdot \Lambda_j \cdot \begin{pmatrix} U_{j,0} & U_{j,1} \end{pmatrix}$$

Then we can move the $V \Lambda$ to the left neighbour pair A_{j-1}.

So we can replace all A by U, upto 1 (the last remaining one).

For open boundary conditions this can be used to orthogonalize every matrix pair e.g. from the left upto the last vector pair at the right end.



OBC:
In the open boundary case the factor r in the left most pair can be extracted as factor for the whole MPS-vector and can be ignored.

Advantage in Rayleigh Quotient minimization:
$N_{eff} = I$ and faster convergence.

PBC:
In the periodic case we cannot get rid of the last matrix pair. Therefore, unitary matrices can be achieved upto one matrix pair.

Advantage in Rayleigh Quotient minimization:
Numerical stability in N_{eff} and faster convergence.

The normalization via SVD leads to the conditions

$$\sum_{i_j} A_{ji_j}^H \cdot A_{ji_j} = I \qquad or \qquad \sum_{i_j} A_{ji_j} \cdot A_{ji_j}^H = I$$

Written componentwise:

$$\sum_{m_j, i_j} \overline{A}_{jm_j i_j, k_{j+1}} \cdot A_{jm_j, i_j m_{j+1}} = \delta_{m_{j+1}, k_{j+1}} \qquad or \qquad \sum_{i_j, m_{j+1}} A_{jm_j, i_j m_{j+1}} \cdot \overline{A}_{jk_j, i_j m_{j+1}} = \delta_{m_j, k_j}$$

$A_{j,ij}$ is then called Kraus operator.

In the nonperiodic case the first and last elements are vectors with

$$\sum_{i_1} \overline{a}_{1i_1, k_2} \cdot a_{1, i_1 m_1} = \delta_{m_2, k_2} \qquad or \qquad \sum_{i_p} b_{pm_p, i_p} \cdot \overline{b}_{pk_p, i_p} = \delta_{m_p, k_p}$$

Orthogonalization via DMRG

Combine two neighbouring matrix pairs and apply SVD:

$$trace\left(A_{1i_1} \cdot A_{2i_2} \cdot A_{3i_3} \cdots\cdots A_{Ni_N}\right)$$

$$\begin{pmatrix} A_{1,0} \\ A_{1,1} \end{pmatrix} \cdot \begin{pmatrix} A_{2,0} & A_{2,1} \end{pmatrix} = \begin{pmatrix} A_{1,0}A_{2,0} & A_{1,0}A_{2,1} \\ A_{1,1}A_{2,0} & A_{1,1}A_{2,1} \end{pmatrix} = U\Lambda V =$$

$$= \begin{pmatrix} U_{1,0} \\ U_{1,1} \end{pmatrix} \cdot \begin{pmatrix} \Lambda V_{2,0} & \Lambda V_{2,1} \end{pmatrix} = \begin{pmatrix} U_{1,0}\Lambda \\ U_{1,1}\Lambda \end{pmatrix} \cdot \begin{pmatrix} V_{2,0} & V_{2,1} \end{pmatrix}$$

Start e.g. on the left, and orthogonalize each matrix pair, and then go the right neighbour.

Further Transformation

$$u_{1i_1} \cdot U_{2i_2} \cdot U_{3i_3} \cdot \cdots \cdot u_{Ni_N}$$

with $U_{j0}^H \cdot U_{j0} + U_{j1}^H \cdot U_{j1} = I$

$$\begin{matrix} \cdots \\ \cdots \end{matrix} \left(\frac{U_{j,0}}{U_{j,1}} \right) \left(\frac{U_{j+1,0}}{U_{j+1,1}} \right) \begin{matrix} \cdots \\ \cdots \end{matrix}$$

Use SVD $U_{j,0} = V \Sigma W^H$

$$\begin{matrix} \cdots \\ \cdots \end{matrix} \left(\frac{V \Sigma W^H}{U_{j,1}} \right) \left(\frac{U_{j+1,0}}{U_{j+1,1}} \right) \begin{matrix} \cdots \\ \cdots \end{matrix} \Rightarrow \begin{matrix} \cdots \\ \cdots \end{matrix} \left(\frac{V \Sigma}{U_{j,1} W} \right) \left(\frac{W^H U_{j+1,0}}{W^H U_{j+1,1}} \right) \begin{matrix} \cdots \\ \cdots \end{matrix}$$

$$\begin{matrix} \cdots \\ \cdots \end{matrix} \left(\frac{B_{j,0}}{B_{j,1}} \right) \left(\frac{W^H U_{j+1,0}}{W^H U_{j+1,1}} \right) \begin{matrix} \cdots \\ \cdots \end{matrix}$$

with $\underbrace{B_{j,0}^H B_{j,0}} + \underbrace{B_{j,1}^H B_{j,1}} = I$

nonnegative diagonal

MPS + MPS gives larger MPS:

$$x_{i_1 i_2 \cdots i_N} + y_{i_1 i_2 \cdots i_N} =$$

$$\sum_{m_1,\cdots,m_N} a_{1i_1,m_1m_2} \cdot a_{2i_2,m_2m_3} \cdot \cdots \cdot a_{Ni_N,m_Nm_1} + \sum_{k_1,\cdots,k_N} b_{1i_1,k_1k_2} \cdot b_{2i_2,k_2k_3} \cdot \cdots \cdot b_{Ni_N,k_Nk_1} =$$

$$trace\left(A_{1i_1} \cdot A_{2i_2} \cdot \cdots \cdot A_{Ni_N}\right) + trace\left(B_{1i_1} \cdot B_{2i_2} \cdot \cdots \cdot B_{Ni_N}\right) =$$

$$trace\left(\begin{pmatrix} A_{1i_1} & 0 \\ 0 & B_{1i_1} \end{pmatrix} \cdot \begin{pmatrix} A_{2i_2} & 0 \\ 0 & B_{2i_2} \end{pmatrix} \cdot \cdots \cdot \begin{pmatrix} A_{Ni_N} & 0 \\ 0 & B_{Ni_N} \end{pmatrix}\right)$$

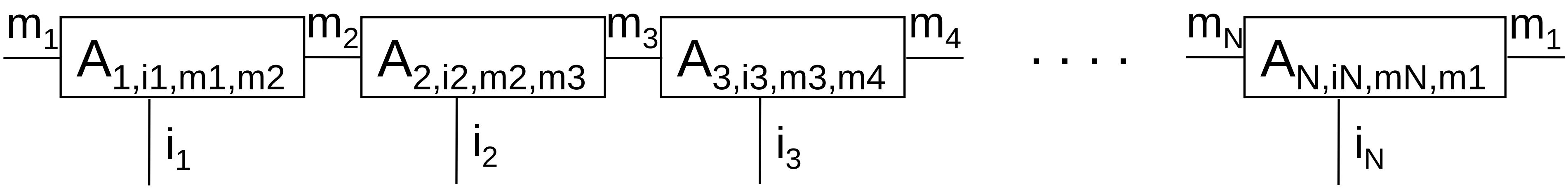
So for original MPS vectors with matrix size D the sum is MPS vector with matrix size 2D.

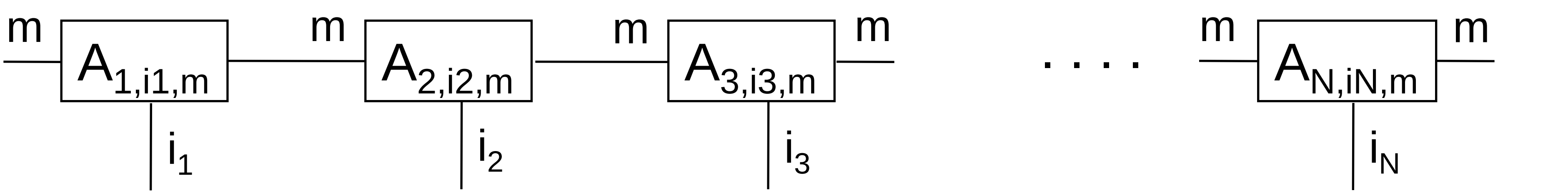
$$MPS_D + MPS_D = MPS_{2D}$$

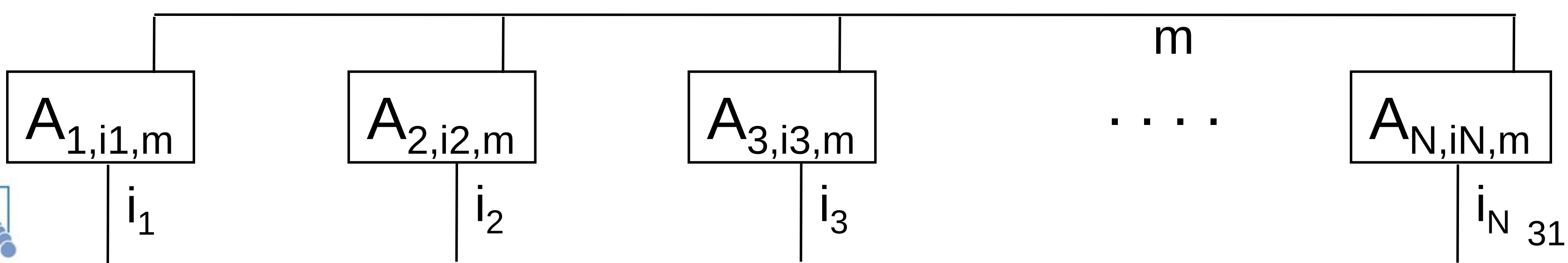
MPS with diagonal matrices = CP

$$\mathrm{trace}\left(\begin{pmatrix} a_{1i_1 1} & 0 & \\ 0 & \ddots & 0 \\ & 0 & a_{1i_1 D} \end{pmatrix} \cdot \begin{pmatrix} a_{2i_2 1} & 0 & \\ 0 & \ddots & 0 \\ & 0 & a_{2i_2 D} \end{pmatrix} \cdot \cdots \cdot \begin{pmatrix} a_{Ni_N 1} & 0 & \\ 0 & \ddots & 0 \\ & 0 & a_{Ni_N D} \end{pmatrix} \right) =$$

$$\sum_{j=1}^{D} a_{1i_1 j} \cdot a_{2i_2 j} \cdot \cdots \cdot a_{Ni_N j}$$







MPS Manifold

The space $\mathcal{M}$ of MPS vectors is no linear subspace, but has certain properties:

The unit vectors are in $\mathcal{M}$ with D=1:

$$\left(e_j\right)_{i_1 \ldots . i_N} = \left(e_{j_1 \ldots j_N}\right)_{i_1 \ldots i_N} = \sum_{m_1,\ldots,m_N=1}^{1} \delta_{i_1 j_1 m_1 m_2} \delta_{i_2 j_2 m_2 m_3} \ldots \delta_{i_N j_N m_N m_1}$$

Sparse vectors with nnz=D are members of $\mathcal{M}$ for matrix size D.

$\mathcal{M}$ is a so called matrix manifold → Tangent space

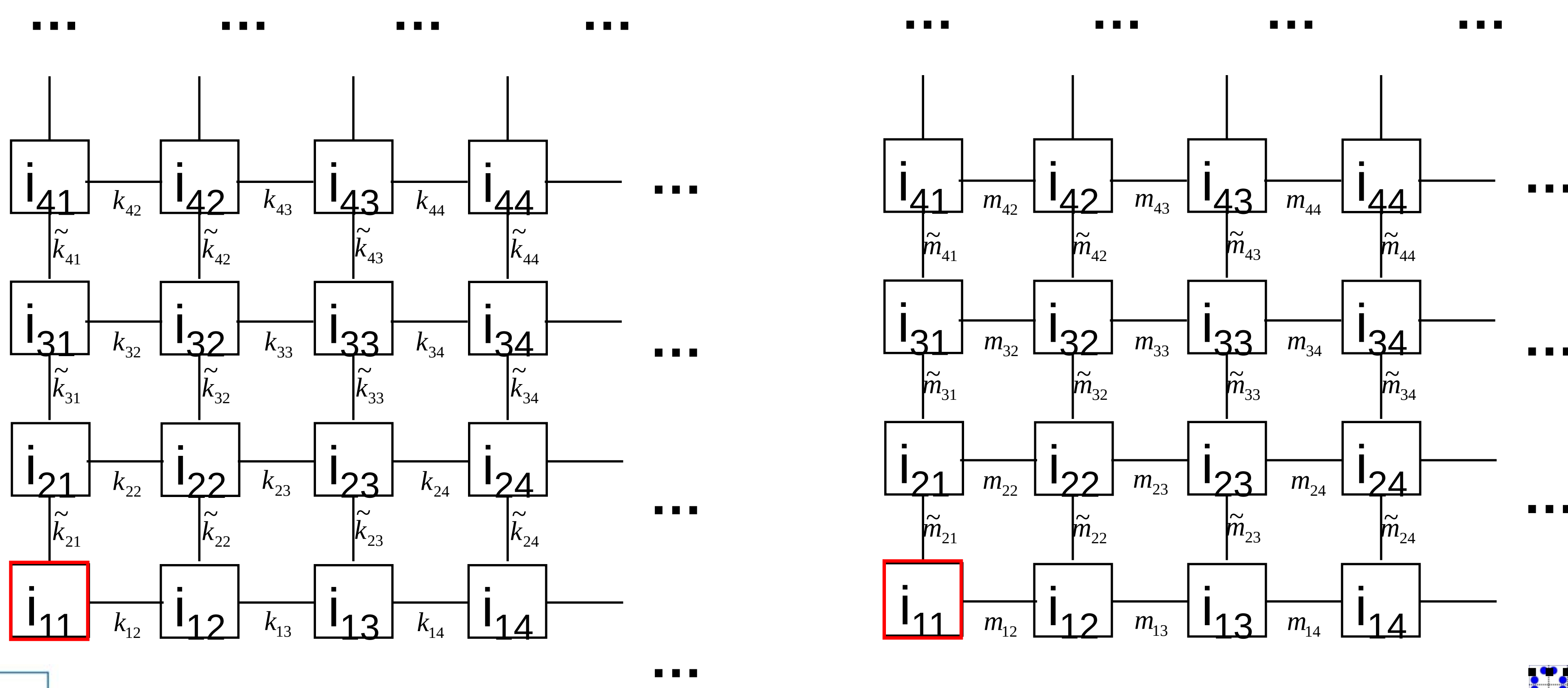
R.Schneider e.a., Absil e.a.

PEPS

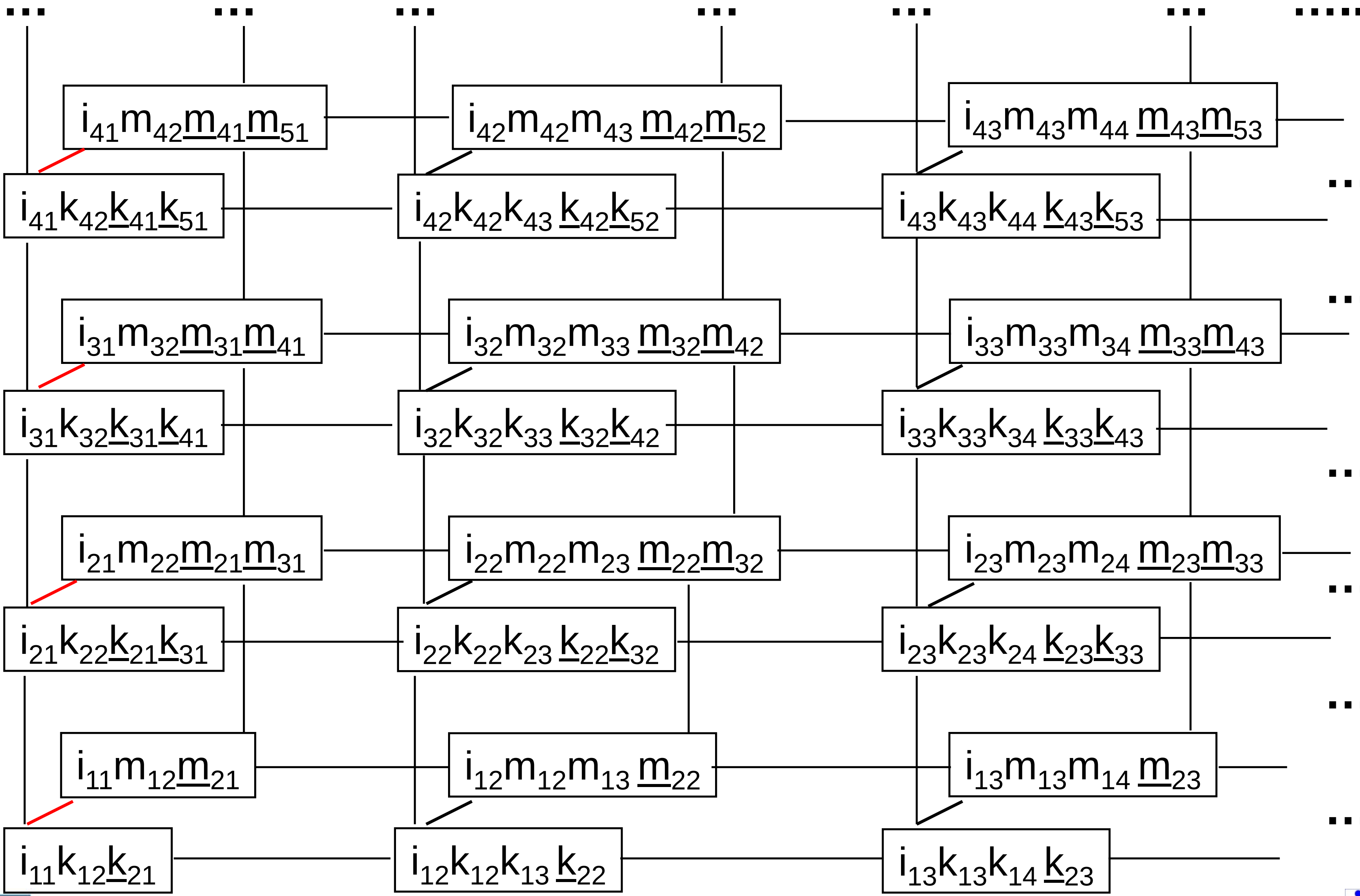
Inner product between two PEPS tensors (k and m):

Contraction beginning from down left with the first left column:

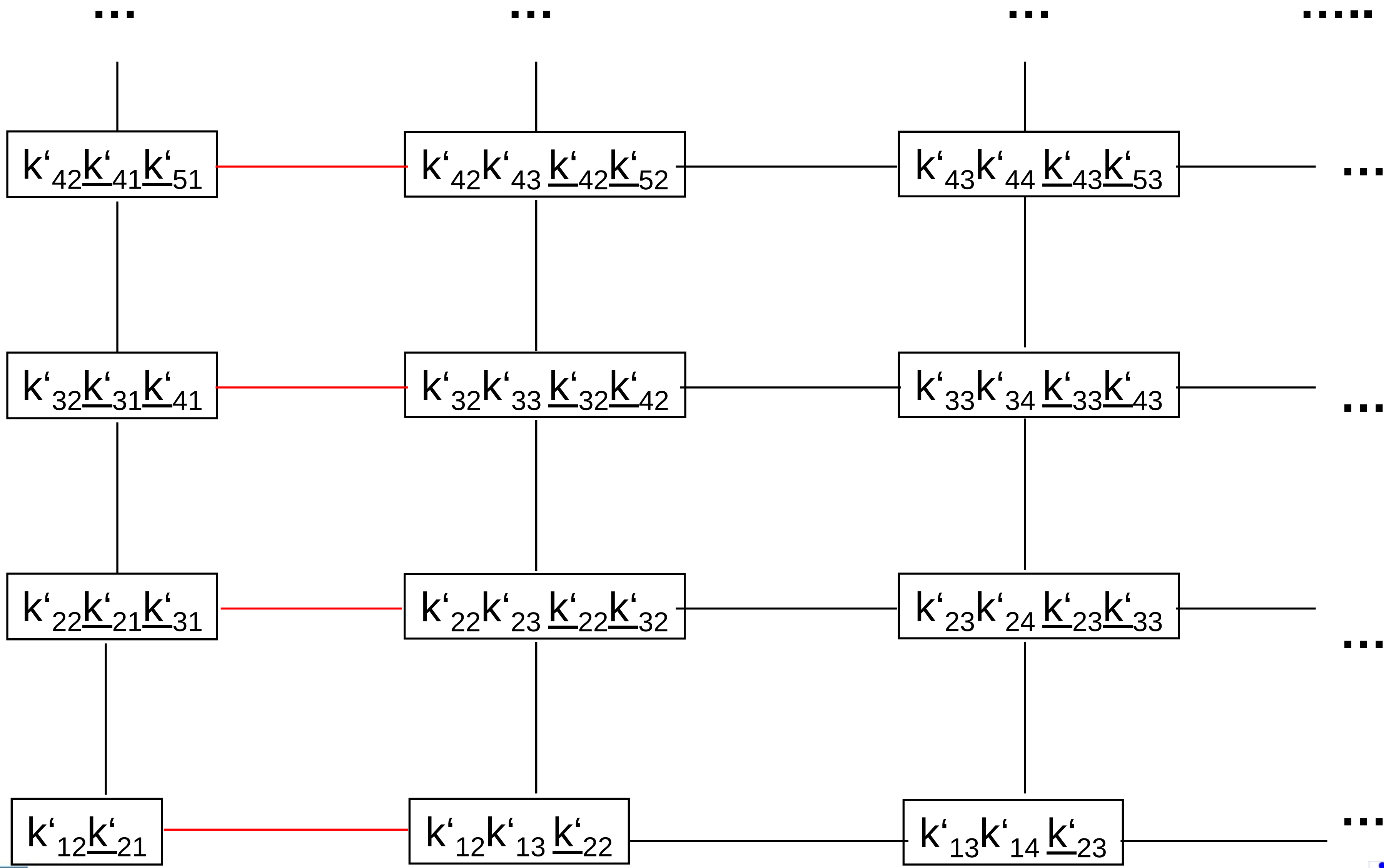
$$X_{i_{11}\cdots i_{NN}} = \sum A_{i_{1,1}m_{1,1}m_{1,2}m_{2,1}m_{2,2}} A_{i_{1,2}m_{1,2}m_{1,3}m_{2,2}m_{2,3}} \cdots$$



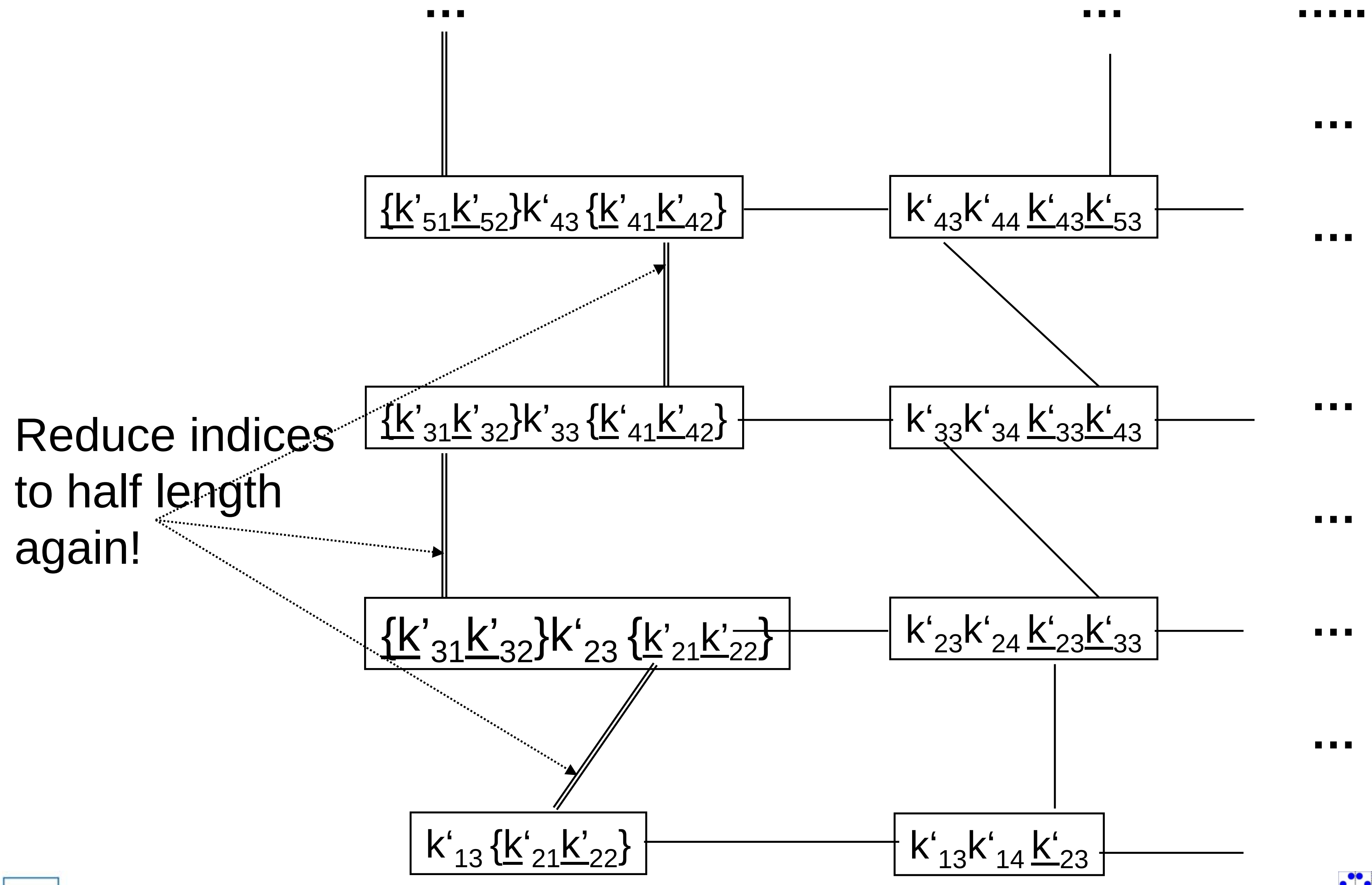
Inner product of two tensors (related to index k, resp. m). First step: Contraction in all i indices.



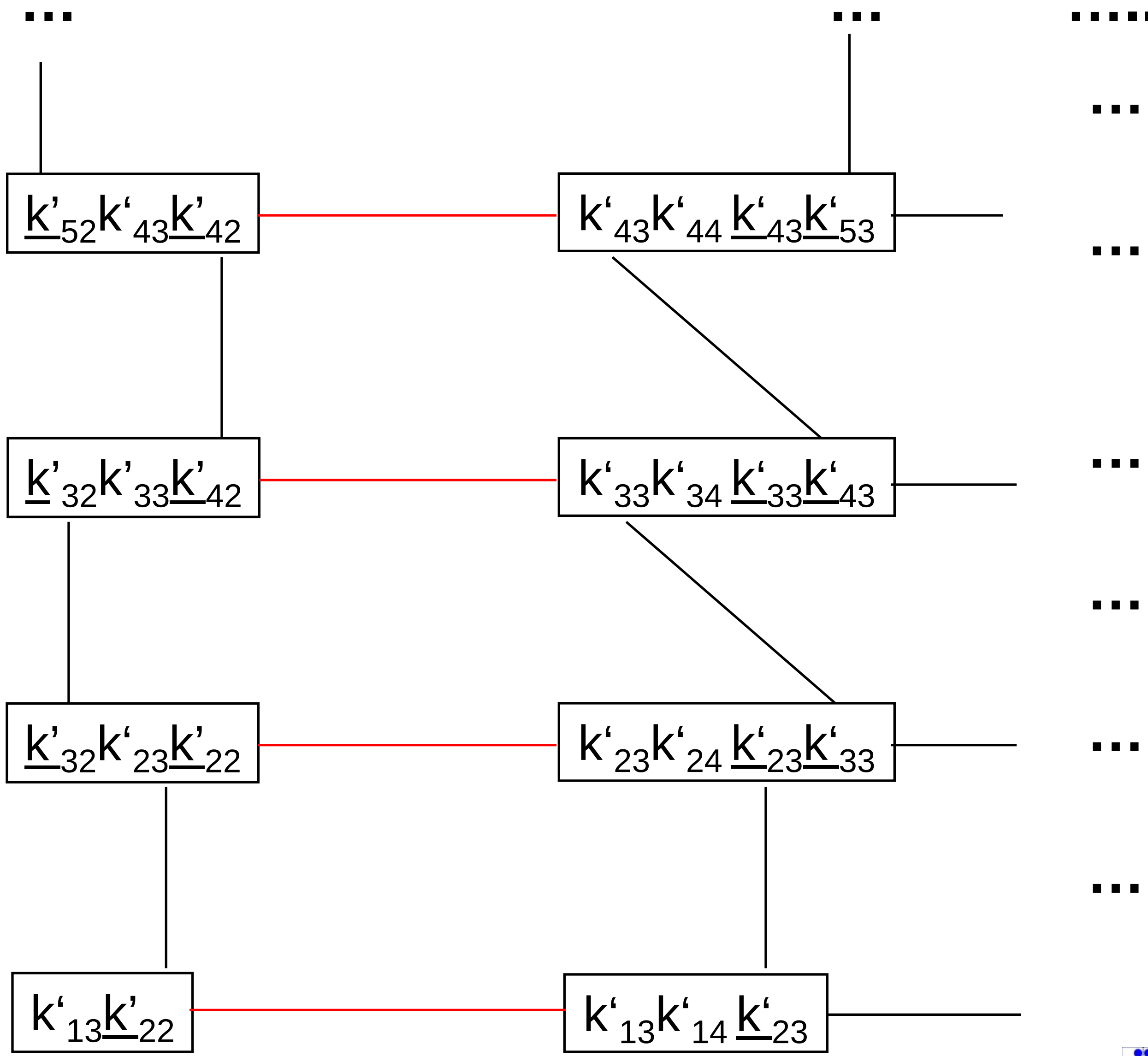
melting k and m to longer k‘ of length r=D².
Contractions pairwise in first and second column:



Melting and reduction to short indices:

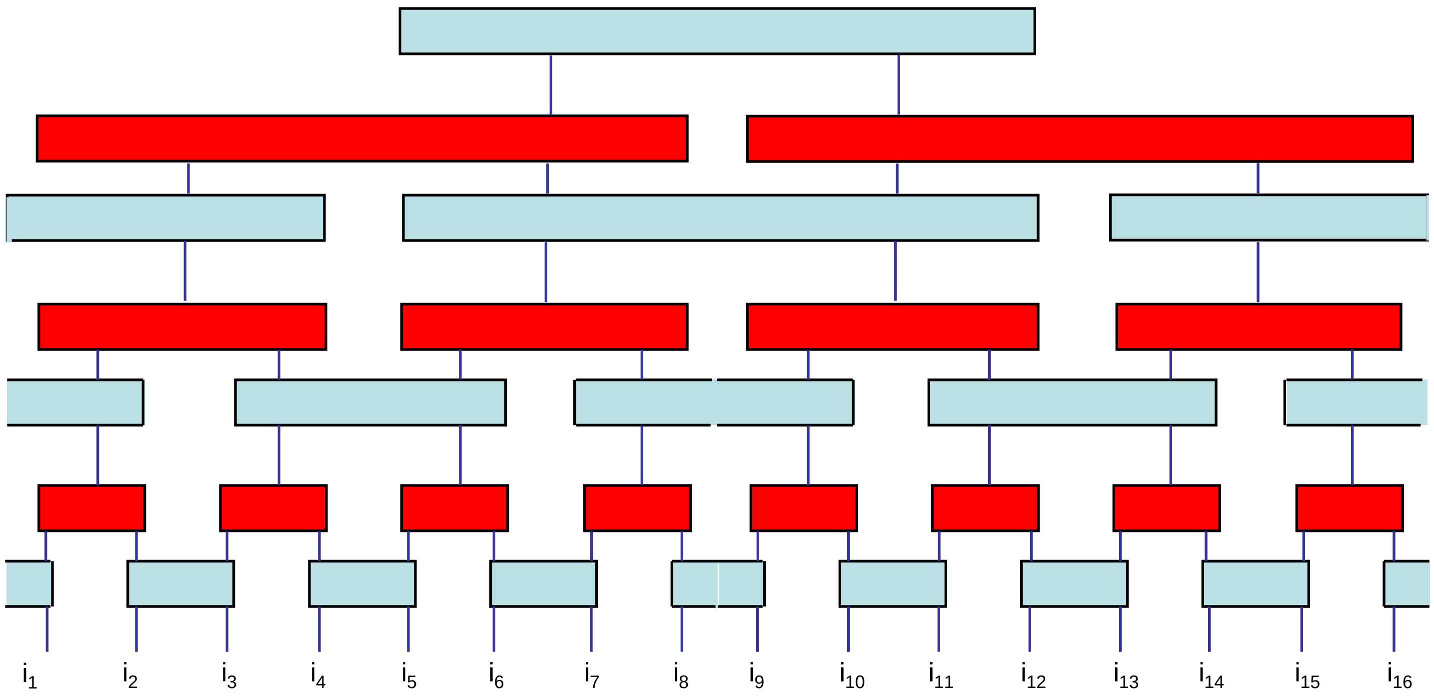


Now, all indices are of short length r again,



Repeat until one column left

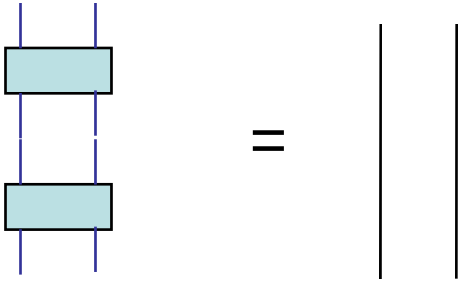
MERA



Layers with unitary tensors and isometries:

$A_{m1m2,k1k2}$ $B_{m1m2,k1}$

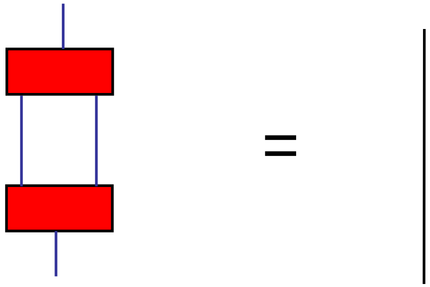
unitary core tensor:



$$\sum_{k_1 k_2} A_{m_1 m_2 k_1 k_2} A'_{k_1 k_2 m'_1 m'_2} = \delta_{m_1 m'_1} \delta_{m_2 m'_2}$$

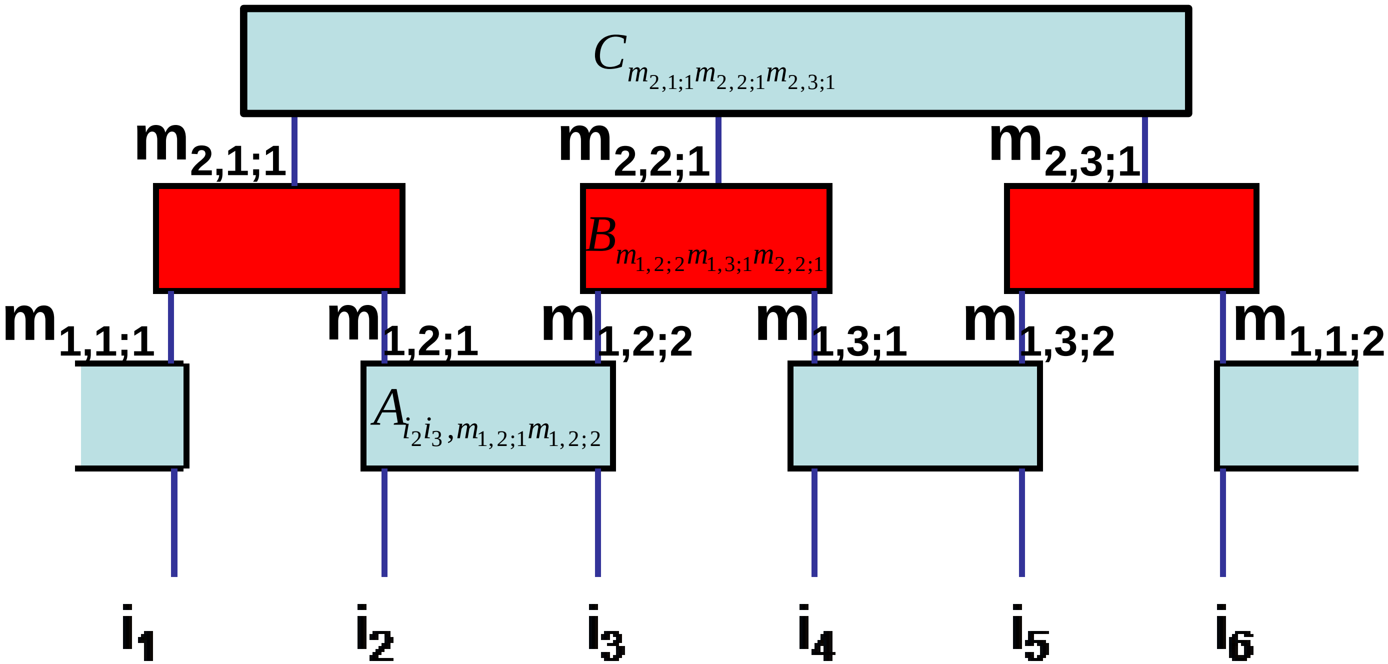
$A_{m1m2,k1k2}$

Isometry:



$$\sum_{m_1 m_2} B'_{k'_1 m_1 m_2} B_{m_1 m_2 k_1} = \delta_{k'_1 k_1}$$

$B_{m1m2,k1}$



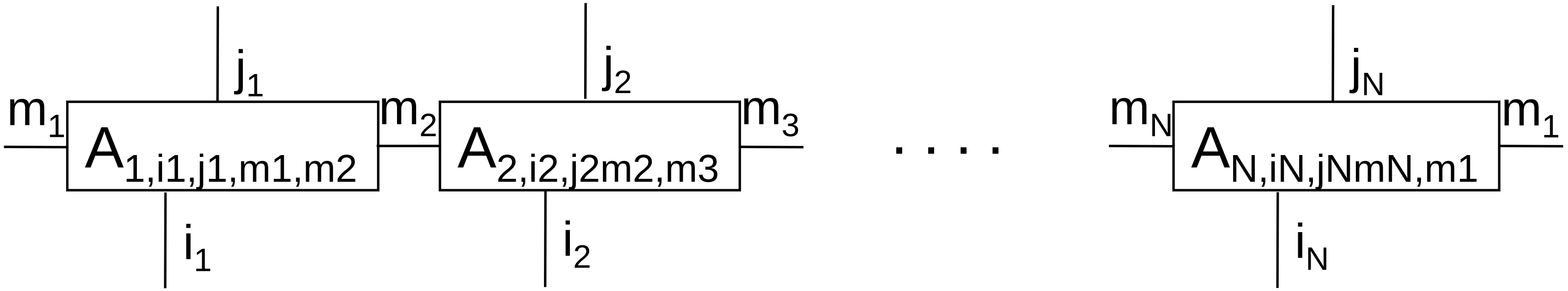
$$x_{i_1i_2i_3i_4i_5i_6} = \sum_{m_{1,1;1},m_{1,2;1},m_{1,2;2},m_{1,3;1},m_{1,3;2},m_{1,1;2}} \sum_{m_{2,1;1},m_{2,2;1},m_{2,3;1}} A_{i_1i_6;m_{1,1;1}m_{1,1;2}} \cdots B_{m_{1,1;1}m_{1,2;1}m_{2,1;1}} \cdots C_{m_{2,1;1}m_{2,2;1}m_{2,3;1}}$$

(b) MPO - Matrix Product Operator

$$O = \sum_{i_1,\dots,i_N} trace\left(A_{1,i_1} \dots A_{N,i_N}\right) \cdot \sigma_{i_1} \otimes \cdots \otimes \sigma_{i_N}$$ with e.g. $$\sigma_0 = I_2, \sigma_1 = \sigma_z$$

or

$$O = \sum_{i_1 j_1 \dots i_N j_N} trace\left(A_{1,i_1 j_1} \dots A_{N,i_N j_N}\right) | i_1 >< j_1 | \dots | i_N >< j_N |$$



Quantum Physics: Verstraete, Schollwöck,..
Mathematics: Oseledets, Khoromskji, R. Schneider,…

MPS: $$x = \sum_{m_1,\dots,m_N}^{D} \begin{pmatrix} a_{1,0,m_1 m_2} \\ a_{1,1,m_1 m_2} \end{pmatrix} \otimes \begin{pmatrix} a_{2,0,m_2 m_3} \\ a_{2,1,m_2 m_3} \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} a_{N,0,m_N m_1} \\ a_{N,1,m_N m_1} \end{pmatrix} =$$

$$= \sum_{m_1,\dots,m_N}^{D} x_{1,m_1 m_2} \otimes x_{2,m_2 m_3} \otimes \dots \otimes x_{N,m_N m_1}$$

MPO: $$O = \sum_{m_1,\dots,m_N}^{D} \begin{pmatrix} a_{1;0,0;m_1 m_2} & a_{1;0,1;m_1 m_2} \\ a_{1;1,0;m_1 m_2} & a_{1;1,1;m_1 m_2} \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} a_{N;0,0;m_N m_1} & a_{N;0,1;m_N m_1} \\ a_{N;1,0;m_N m_1} & a_{N;1,1;m_N m_1} \end{pmatrix} =$$

$$= \sum_{m_1,\dots,m_N}^{D} X_{1,m_1 m_2} \otimes \dots \otimes X_{N,m_N m_1}$$

m$_2$: 1 2 — m$_3$: 1 2 — m$_N$: 1 2 — m$_N$: 1 2

$$\sum_{m_2 \ldots m_N}^{D} \begin{bmatrix} X & I \end{bmatrix} \,'\otimes'\, \begin{bmatrix} I & 0 \\ X & I \end{bmatrix} \,'\otimes'\, \cdots \,'\otimes'\, \begin{bmatrix} I & 0 \\ X & I \end{bmatrix} \,'\otimes'\, \begin{bmatrix} I \\ X \end{bmatrix}$$

$$X \otimes I \otimes \cdots \otimes I \otimes I$$

$$I \otimes X \otimes \cdots \otimes I \otimes I$$

$$I \otimes I \otimes \cdots \otimes X \otimes I$$

$$I \otimes I \otimes \cdots \otimes I \otimes X$$

$$X \otimes I \otimes \cdots \otimes I + I \otimes X \otimes \cdots \otimes I + \cdots + I \otimes I \otimes \cdots \otimes X$$

Further MPOs I

$$[X \mid I_r] \text{'}\otimes\text{'} \left[\begin{array}{c|c} I_r & 0 \\ \hline X & I_r \end{array}\right] \text{'}\otimes\text{'} \cdots \text{'}\otimes\text{'} \left[\begin{array}{c|c} I_r & 0 \\ \hline X & I_r \end{array}\right] \text{'}\otimes\text{'} \left[\begin{array}{c} I_r \\ \hline X \end{array}\right] \quad \Leftrightarrow \quad \left[\begin{array}{c|c} I_r & 0 \\ \hline X & I_r \end{array}\right]$$

$$\left[\begin{array}{c|c} I_r & 0 \\ \hline X & I_r \end{array}\right] \Rightarrow \sum X_i = X \otimes I + I_r \otimes X \otimes I + \cdots$$

$$\left[\begin{array}{c|c|c} I_r & 0 & 0 \\ \hline X & 0 & 0 \\ \hline 0 & Y & I_r \end{array}\right] \Rightarrow \sum X_i Y_{i+1}$$

→ H_{xx}

$$\left[\begin{array}{c|c|c|c} I_r & 0 & 0 & 0 \\ \hline X & 0 & 0 & 0 \\ \hline 0 & Y & 0 & 0 \\ \hline 0 & 0 & Z & I_r \end{array}\right] \Rightarrow \sum X_i Y_{i+1} Z_{i+2}$$

Further MPOs II

$$\begin{bmatrix} I_r & 0 & 0 \\ X & 0 & 0 \\ Y & Z & I_r \end{bmatrix} \Rightarrow \sum Y_i + Z_i X_{i+1}$$

→ Ising

$$\begin{bmatrix} I_r & 0 & 0 & 0 \\ X & 0 & 0 & 0 \\ 0 & I_r & 0 & 0 \\ 0 & Y & Z & I_r \end{bmatrix} \Rightarrow \sum Y_i X_{i+1} + Z_i X_{i+2}$$

$$\begin{bmatrix} I & 0 & 0 & 0 & 0 \\ X & 0 & 0 & 0 & 0 \\ Y & 0 & 0 & 0 & 0 \\ Z & 0 & 0 & 0 & 0 \\ 0 & U & V & W & I \end{bmatrix} \Rightarrow \sum U_i X_{i+1} + V_i Y_{i+1} + W_i Z_{i+1}$$

Core tensor in different forms:

$$\begin{bmatrix} I_r & 0 & 0 \\ X & 0 & 0 \\ Y & Z & I_r \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ X_{11} & X_{12} & 0 & 0 & 0 & 0 \\ X_{21} & X_{22} & 0 & 0 & 0 & 0 \\ Y_{11} & Y_{12} & Z_{11} & Z_{12} & 1 & 0 \\ Y_{21} & Y_{22} & Z_{21} & Z_{22} & 0 & 1 \end{bmatrix}$$

$$A_{00} = \begin{pmatrix} 1 & 0 & 0 \\ X_{11} & 0 & 0 \\ Y_{11} & Z_{11} & 1 \end{pmatrix}, \quad A_{01} = \begin{pmatrix} 0 & 0 & 0 \\ X_{12} & 0 & 0 \\ Y_{12} & Z_{12} & 0 \end{pmatrix}, \quad A_{10}, \quad A_{11} \rightarrow (A_{i,j})$$

MPO with D=3

$$\left[\begin{array}{c|c|c} I_r & 0 & 0 \\ \hline X & 0 & 0 \\ \hline Y & Z & I_r \end{array}\right] \Leftrightarrow \begin{pmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{pmatrix}$$

$$\left[\begin{array}{cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hline X_{11} & X_{12} & 0 & 0 & 0 & 0 \\ X_{21} & X_{22} & 0 & 0 & 0 & 0 \\ \hline Y_{11} & Y_{12} & Z_{11} & Z_{12} & 1 & 0 \\ Y_{21} & Y_{22} & Z_{21} & Z_{22} & 0 & 1 \end{array}\right] \Leftrightarrow \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 0 & 0 & 0 \\ X_{11} & 0 & 0 & X_{12} & 0 & 0 \\ Y_{11} & Z_{11} & 1 & Y_{12} & Z_{12} & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 \\ X_{21} & 0 & 0 & X_{22} & 0 & 0 \\ Y_{21} & Z_{21} & 0 & Y_{22} & Z_{22} & 1 \end{array}\right]$$

Advantage: MPO · MPS

$$MPO_{D_O} \cdot MPS_D =$$

$$= \left(\sum_{m_1,\dots,m_N}^{D_O} \begin{pmatrix} a_{1;0,0;m_1m_2} & a_{1;0,1;m_1m_2} \\ a_{1;1,0;m_1m_2} & a_{1;1,1;m_1m_2} \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} a_{N;0,0;m_Nm_1} & a_{N;0,1;m_Nm_1} \\ a_{N;1,0;m_Nm_1} & a_{N;1,1;m_Nm_1} \end{pmatrix} \right) \cdot MPS =$$

$$= \sum_{m_1,\dots,m_N}^{D_O} X_{1,m_1m_2} \otimes \dots \otimes X_{N,m_Nm_1} \cdot \sum_{k_1,\dots,k_N}^{D} a_{1,k_1k_2} \otimes \dots \otimes a_{N,k_Nk_1} =$$

$$= \sum_{m_1,\dots,m_N}^{D_O} \sum_{k_1,\dots,k_N}^{D} \left(X_{1,m_1m_2} a_{1,k_1k_2} \right) \otimes \dots \otimes \left(X_{N,m_Nm_1} a_{N,k_Nk_1} \right) =$$

$$= \sum_{(m_1,k_1),\dots,(m_N,k_N)}^{D_O,D} \left(X_{1,m_1m_2} a_{1,k_1k_2} \right) \otimes \dots \otimes \left(X_{N,m_Nm_1} a_{N,k_Nk_1} \right) =$$

$$= \sum_{k_1',\dots,k_N'}^{D_O \cdot D} b_{1,k_1'k_2'} \otimes \dots \otimes b_{N,k_N'k_1'} = MPS_{D_O \cdot D}$$

$$MPO_{D_O} + MPO_{D_O} = MPO_{2D_O}$$

(c) Symmetries

The considered Hamiltonians often have special symmetries:

J the anti-identity: $J = \begin{pmatrix} & & 1 \\ & \iddots & \\ 1 & & \end{pmatrix}$ $H = H^T$ and $JHJ = H^T$

symmetric persymmetric

Then the eigenvalues are symmetric: $Jx = \pm x$

Other symmetries: $x = \begin{pmatrix} a \\ a \end{pmatrix}, \quad x = \begin{pmatrix} a \\ -a \end{pmatrix}, \quad Jx = \pm \bar{x},$

Or for general permutations P: $Px = \pm x, \quad Px = \pm \bar{x},$

Question: How to model these symmetries in the MPS ansatz?

Example I

$$x_{i_1 \dots i_N} = tr\left(A_{i_1} \cdot \cdots \cdot A_{i_N}\right)$$ with the same matrix pair $\begin{pmatrix} A_0 \\ A_1 \end{pmatrix}$ at each position

This is related to translation invariant spin periodic spin system or TI MPS.

Leads to strong symmetries, e.g. $x_{001} = x_{010} = x_{100}$

$$tr(A_0 A_0 A_1) = tr(A_0 A_1 A_0) = tr(A_1 A_0 A_0)$$

Main property: trace(AB) = trace(BA) → bit shift symmetry:

$$x_{i_1 i_2 \dots i_N} = x_{i_2 i_3 \dots i_N i_1} = x_{i_3 i_4 \dots i_N i_1 i_2} = \dots = x_{i_N i_1 i_2 \dots i_{N-1}}$$

Similarly blockwise:

$$x_{i_1 i_2 \ldots i_6} = tr(A_{i_1} B_{i_2} C_{i_3} | A_{i_4} B_{i_5} C_{i_6}) \Rightarrow$$

$$x_{100000} = tr(A_1 B_0 C_0 | A_0 B_0 C_0) =$$

$$= tr(A_0 B_0 C_0 | A_1 B_0 C_0) = x_{000100}$$

$$x_{i_1 i_2 i_3 i_4 i_5 i_6} = x_{i_4 i_5 i_6 i_1 i_2 i_3}$$

$$x_{i_1 i_2 \ldots i_6} = tr(A_{i_1} B_{i_2} | A_{i_3} B_{i_4} | A_{i_5} B_{i_6}) \Rightarrow$$

$$x_{100001} = tr(A_1 B_0 | A_0 B_0 | A_0 B_1) =$$

$$= tr(A_0 B_0 | A_0 B_1 | A_1 B_0) = x_{000110} =$$

$$= tr(A_0 B_1 | A_1 B_0 | A_0 B_0) = x_{011000}$$

$$x_{i_1 i_2 i_3 i_4 i_5 i_6} = x_{i_3 i_4 i_5 i_6 i_1 i_2} = x_{i_5 i_6 i_1 i_2 i_3 i_4}$$

Example II

Bitreversal Symmetry:

MPS $x_{i_1 \dots i_N} = tr\left(A_{1,i_1} \cdot \cdots \cdot A_{N,i_N}\right)$ with $A_{j,i_j}^T = A_{N+1-j,i_j}$, $\quad j = 1, \dots, N$

$$
\begin{aligned}
& x_{i_1 i_2 \dots i_{N-1} i_N} = \\
& = tr\left(A_{1,i_1} \cdot \cdots \cdot A_{N,i_N}\right) = tr\left(A_{1,i_1} \cdot \cdots \cdot A_{N,i_N}\right)^T = \\
& = tr\left(A_{N,i_N}^T \cdot \cdots \cdot A_{1,i_1}^T\right) = tr\left(A_{1,i_N} \cdot \cdots \cdot A_{N,i_1}\right) = \\
& = x_{i_N i_{N-1} \dots i_2 i_1}
\end{aligned}
$$

Special case TI MPS: $x_{i_1 \dots i_N} = tr(A_{i_1} \cdots A_{i_N}) \quad with \quad A_i^T = A_i$

Example III

$$x_{i_1 \ldots i_N} = tr\left(A_{1,i_1} \cdot \cdots \cdot A_{N,i_N}\right) \quad \text{with} \quad \begin{pmatrix} A_{1,0} \\ A_{1,1} \end{pmatrix} = \begin{pmatrix} B \\ \pm B \end{pmatrix} \quad \Rightarrow \quad x = \begin{pmatrix} b \\ \pm b \end{pmatrix}$$

Similarly, if the last matrix pair has this property, then

$$\begin{pmatrix} A_{N,0} \\ A_{N,1} \end{pmatrix} = \begin{pmatrix} B \\ \pm B \end{pmatrix} \quad \Rightarrow \quad x = \begin{pmatrix} b_1 \\ \pm b_1 \\ b_2 \\ \pm b_2 \\ \vdots \\ \vdots \end{pmatrix}$$

$$A_{1,0}A_{2,0} = A_{1,1}A_{2,1} \Rightarrow x_{00i_3 \ldots i_N} = x_{11i_3 \ldots i_N}$$

$$A_{1,0}A_{2,0} = A_{1,1}A_{2,1} = A_{1,0}A_{2,1} = A_{1,1}A_{2,0} \Rightarrow$$

$$x_{00i_3 \ldots i_N} = x_{11i_3 \ldots i_N} = x_{01i_3 \ldots i_N} = x_{10i_3 \ldots i_N}$$

Persymmetry – Bit flip Symmetry

$$tr\left[\begin{array}{c|c|c|c|c} A_{1,0} & A_{2,0} & \cdots & A_{N-1,0} & A_{N,0} \\ \hline A_{1,0}U_2 & U_2A_{2,0}U_3 & \cdots & U_{N-1}A_{N-1,0}U_N & U_NA_{N,0} \end{array}\right]$$

with U_j involutions: $U_j^2 = I$

$$x_{111\ldots1} = tr\left(\left(A_{1,0}U_2\right)\left(U_2A_{2,0}\right)\cdots U_NA_{N,0}\right) = x_{000\ldots0}$$

It holds: $A_{j,1} = U_jA_{j.0}U_{j+1},\quad resp.\quad A_{j,0} = U_jA_{j.1}U_{j+1}$

In general:

$$x_{i_1i_2\ldots i_N} = tr\left(A_{1,i_1}A_{2,i_2}\cdots\right) = tr\left(A_{i,\tilde{i}_1}U_2U_2A_{2,\tilde{i}_2}U_3\cdots\right) = x_{\tilde{i}_1\tilde{i}_2\ldots\tilde{i}_N}$$

Bit flip Normal Form

U_j involution → $U_j = S_j^{-1} D_{j,\pm 1} S_j$ (Jordan block $J_U^2=I$)

$$tr\left[\begin{array}{c|c|c|c|c} \frac{A_{1,0}}{A_{1,0}U_2} & \frac{A_{2,0}}{U_2 A_{2,0} U_3} & \frac{\cdots}{\cdots} & \frac{A_{N-1,0}}{U_{N-1}A_{N-1,0}U_N} & \frac{A_{N,0}}{U_N A_{N,0}} \end{array}\right] =$$

$$= tr\left[\begin{array}{c|c|c|c} \frac{A_{1,0}}{A_{1,0}S_2^{-1}D_{2,\pm1}S_2} & \frac{A_{2,0}}{S_2^{-1}D_{2,\pm1}S_2A_{2,0}S_3^{-1}D_{3,\pm1}S_3} & \frac{\cdots\quad\cdots}{\cdots\quad\cdots} & \frac{A_{N,0}}{S_N^{-1}D_{N,\pm1}S_N A_{N,0}} \end{array}\right] =$$

$$= tr\left[\begin{array}{c|c|c|c} \frac{A_{1,0}S_2^{-1}}{A_{1,0}S_2^{-1}D_{2,\pm1}} & \frac{S_2A_{2,0}S_3^{-1}}{D_{2,\pm1}S_2A_{2,0}S_3^{-1}D_{3,\pm1}} & \frac{\cdots\quad\cdots}{\cdots\quad\cdots} & \frac{S_N A_{N,0}}{D_{N,\pm1}S_N A_{N,0}} \end{array}\right] =$$

$$= tr\left[\begin{array}{c|c|c|c} \frac{\tilde{A}_{1,0}}{\tilde{A}_{1,0}D_{2,\pm1}} & \frac{\tilde{A}_{2,0}}{D_{2,\pm1}\tilde{A}_{2,0}D_{3,\pm1}} & \frac{\cdots\quad\cdots}{\cdots\quad\cdots} & \frac{\tilde{A}_{N,0}}{D_{N,\pm1}\tilde{A}_{N,0}} \end{array}\right]$$

Embed all $D_{j,\pm1}$ in anti-identy J

Quasi Uniqueness

Assume that the MPS vector is of the form

$$tr\left[\frac{A_{1,0}}{A_{1,0}U_2}\middle|\frac{A_{2,0}}{V_2A_{2,0}U_3}\middle|\frac{\cdots}{\cdots}\middle|\frac{A_{N-1,0}}{V_{N-1}A_{N-1,0}U_N}\middle|\frac{A_{N,0}}{V_NA_{N,0}}\right]$$

with unitary matrices V_j and U_j.
Assume that it holds $Jx = x$ for all possible choices of $A_{j,0}$.

Then it follows that $U_j^2 = U_j = V_j$ are involutions for all j.

Full Bit Symmetry

$$tr\left[\begin{array}{c|c|c|c|c} A_0 & A_0 & \cdots & A_0 & A_0 \\ \hline JA_0J & JA_0J & \cdots & JA_0J & JA_0J \end{array}\right], \qquad A_0^T = A_0$$

Gives bit reverse/flip/shift symmetry (we can assume sym. invol.)

Without Persymmetry: $A_0^T = A_0 = U^H \Lambda U, \quad A_1^T = A_1, \quad B = U^H A_1 U$

$$tr\left[\begin{array}{c|c|c|c|c} A_0 & A_0 & \cdots & A_0 & A_0 \\ \hline A_1 & A_1 & \cdots & A_1 & A_1 \end{array}\right] =$$

$$= tr\left[\begin{array}{c|c|c|c|c} \Lambda & \Lambda & \cdots & \Lambda & \Lambda \\ \hline UA_1U^H & UA_1U^H & \cdots & UA_1U^H & UA_1U^H \end{array}\right] =$$

$$= tr\left[\begin{array}{c|c|c} \Lambda & \cdots & \Lambda \\ \hline B & \cdots & B \end{array}\right]$$

as possible normal form.

Advantages

Reduction in degree of freedom by using symmetries:

bit shift $2^N \rightarrow 2^N/N$
bit reversal $2^N \rightarrow 2^N/2$
bit flip $2^N \rightarrow 2^N/2$

More compact normal form.

Less storage,
faster convergence, and
higher accuracy
in vector approximation.

(d) Krylov - MPS

Replaced Rayleigh Quotient Minimzation by Krylov Subspace minimization in MPS space.

Generate Orthonormal basis of $K_n(H,x) = span\{x, Hx, ..., H^{n-1}x\}$

Problem:
Hx_{MPS} and pairwise orthogonalization of x_{MPS} gives MPS vectors with larger blocksize D_{new}.

Therefore, we have to apply back projection into MPS_D-space

MPO representation of H reduces the costs for Hx dramatically!

Projected Krylov Subspace Iteration

- avoid orthogonalization,
- use subspaces of fixed size.

$$A_n = \begin{pmatrix} x^H H x & x^H H^2 x & \cdots & x^H H^n x \\ x^H H^2 x & x^H H^3 x & \cdots & x^H H^{n+1} x \\ \vdots & \vdots & \ddots & \vdots \\ x^H H^n x & x^H H^{n+1} x & \cdots & x^H H^{2n-1} x \end{pmatrix} \qquad B_n = \begin{pmatrix} x^H x & x^H H x & \cdots & x^H H^{n-1} x \\ x^H H x & x^H H^2 x & \cdots & x^H H^n x \\ \vdots & \vdots & \ddots & \vdots \\ x^H H^{n-1} x & x^H H^n x & \cdots & x^H H^{2n-2} x \end{pmatrix}$$

Solve $A_n y = \lambda B_n y$

New eigenvector approximation $x_{new} = y_1 x + y_2 H x + \ldots + y_n H^{n-1} x$

Projections

Replace in subspace $y_j = H^j x_{MPS\_D}$ (a $D_o^j \cdot D$ MPS vector)

by projection into MPS_D space:

$$\min \left\| y_{MPS\_D} - y_{MPS\_DO^j D} \right\|_2$$

Solve this minimization approximately by
- SVD compression
- Alternating Least Squares minimization

Subspace Iteration

$$\tilde{K}_n(H,x) = span\{x, P(Hx), P(H \cdot P(Hx))..., P(H \cdot P(H \cdot ... \cdot P(Hx)...))\}$$
$$= span\{x_1, x_2, ..., x_n\}$$

$$A_n = \begin{pmatrix} x_1^H H x_1 & \cdots & x_1^H H x_n \\ \vdots & \ddots & \vdots \\ x_n^H H x_1 & \cdots & x_n^H H x_n \end{pmatrix} \qquad B_n = \begin{pmatrix} x_1^H x_1 & \cdots & x_1^H x_n \\ \vdots & \ddots & \vdots \\ x_n^H x_1 & \cdots & x_n^H x_n \end{pmatrix}$$

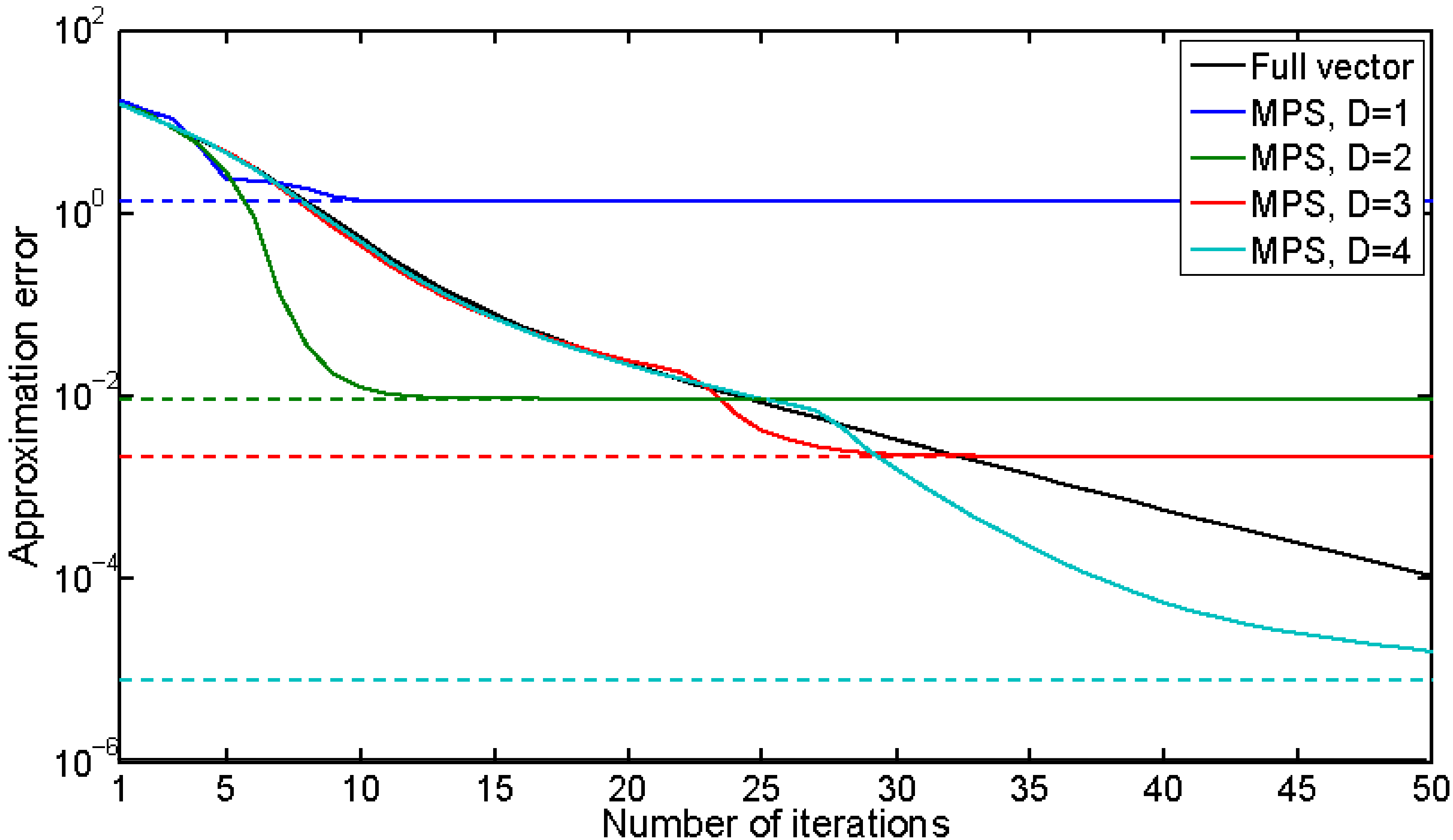
Solve $A_n y = \lambda B_n y$

Gives new eigenvector estimate

$$x_{MPS} = P(y_1 x_1 + ... + y_n x_n)$$

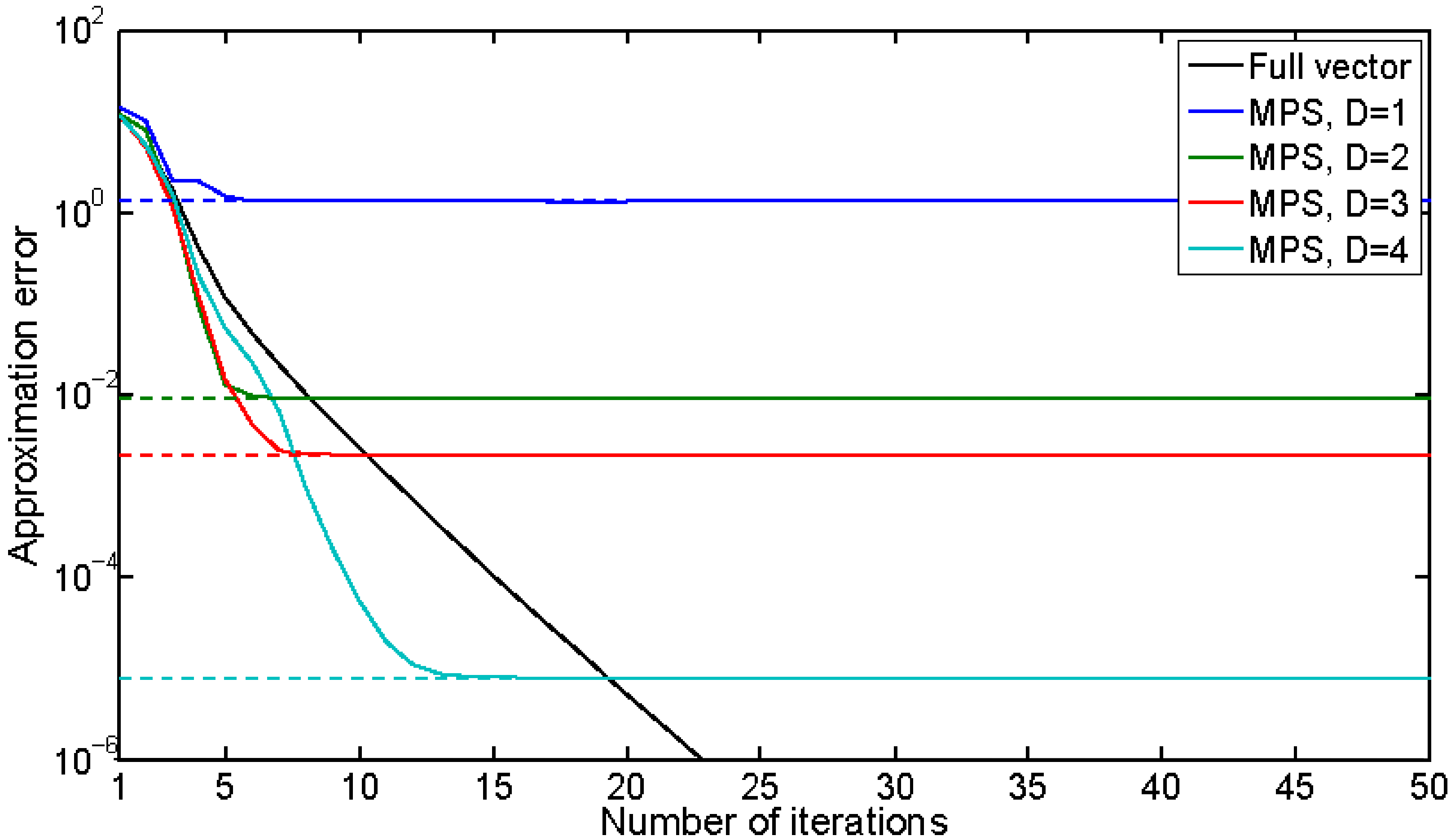
Numerical results

Ising-type Hamiltonian, $N = 10$, $dim = 2$



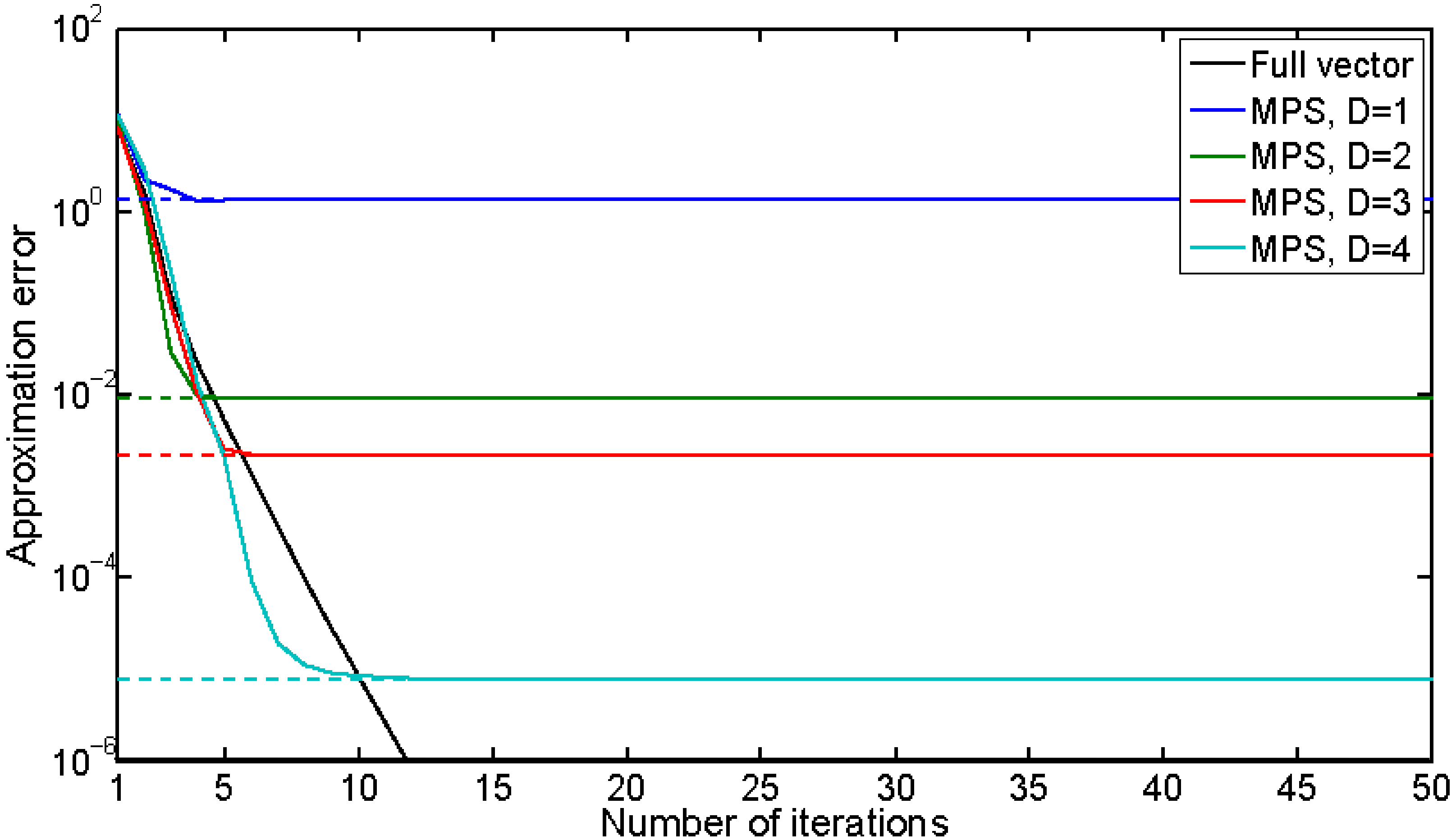
Numerical results

Ising-type Hamiltonian, $N = 10$, $dim = 3$



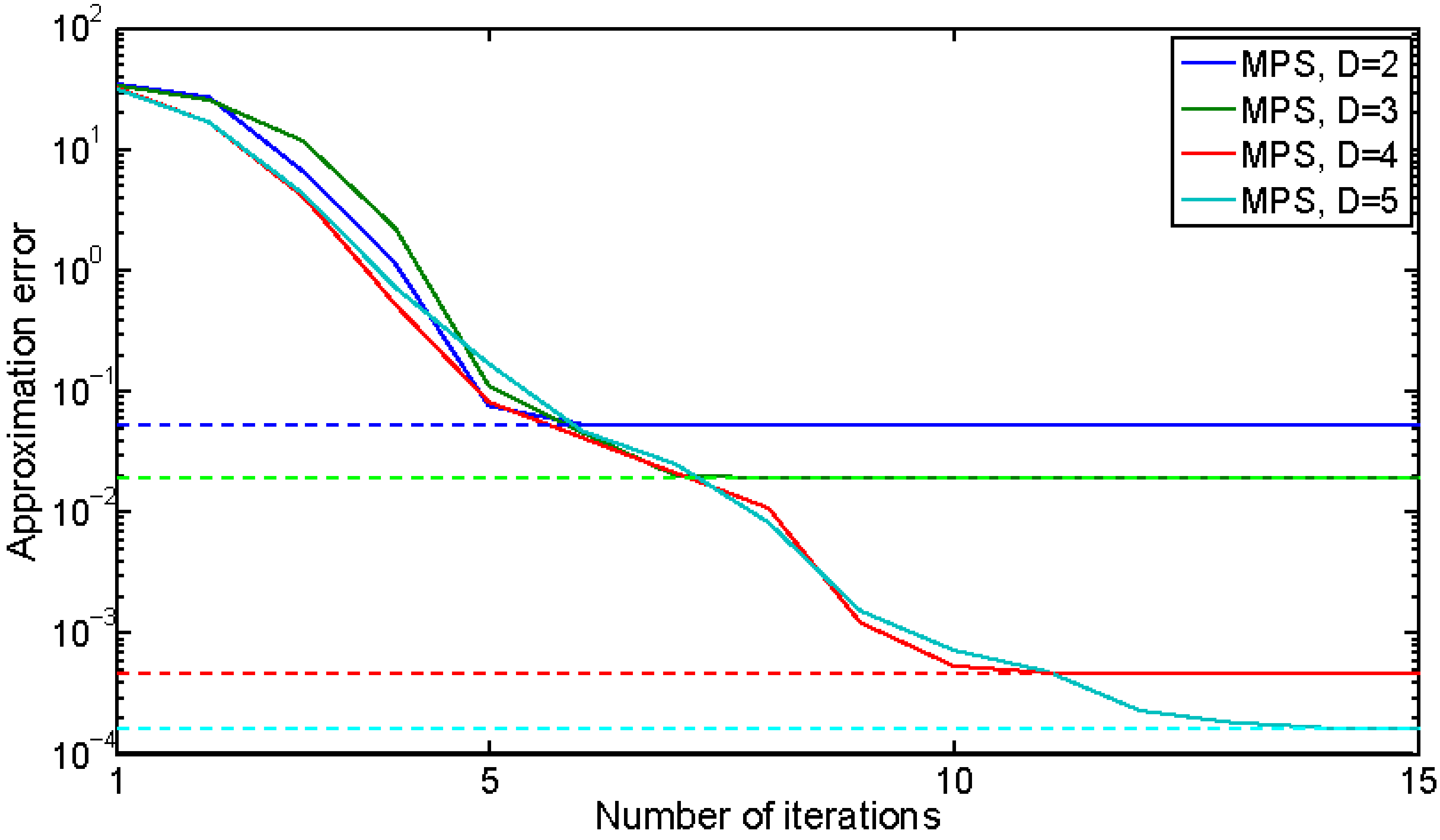
Numerical results

Ising-type Hamiltonian, $N = 10$, $dim = 4$

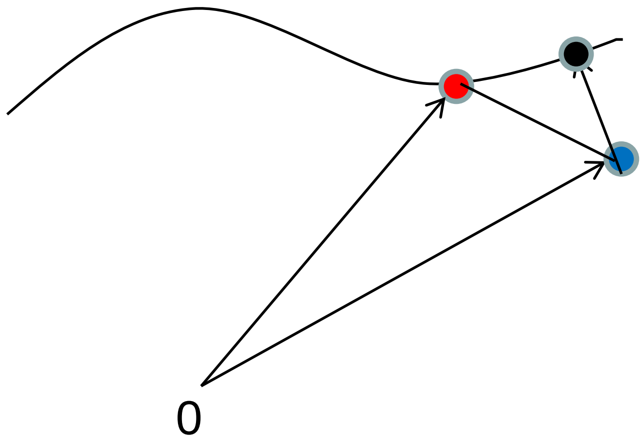


Numerical results

Ising-type Hamiltonian, $N = 25$, $dim = 5$



- Faster convergence by projection because exact eigenvector is very close to MPS manifold
- Use solution for D as start approximation for D+1
- Can compute more eigenvalues/vectors
- Only projection is needed



Modifications:

- Size of subspace,
- D,
- use exact Krylov matrix
- include projected orthogonalization

$$x_1 := x_1 / \|x_1\|;$$

$$rx_2 := Hx_1 - (x_1^H H x_1) x_1; \qquad x_2 = P(x_2);$$

$$\vdots$$

Conclusions:

- MPS-TT allows efficient and high quality approximation of eigenvectors of huge Hamiltonians

- Matrix Product Operators are very useful in connection with MPS vectors

- Symmetries in the vector can be expressed in the MPS ansatz

- Krylov methods can be applied including projections

Thank you