

Komprimierung von Matrix Product Zustand

Idee wir haben ein Tensor χ in Form $\sum_{a_1, \dots, a_n} \chi_{a_1, \dots, a_n} |a_1\rangle \dots |a_n\rangle$

$$\chi_{n_1 \dots n_N} = \sum_{a_1, \dots, a_N} \underbrace{A_{a_1}^{n_1} A_{a_2}^{n_2} \dots A_{a_N}^{n_N}}_{\text{System A}} \underbrace{\Lambda_{a_{N-1} a_N}^{n_N}}_{\text{System B}}$$

Dunkelraum \downarrow

Im Prinzip können wir dies als Startpunkt zu Schmidt Dekomposition verwenden!

$$|\chi\rangle = \sum_{k=1}^D s_k |a_k\rangle_A |a_k\rangle_B$$

- 1) Behalten die erste D Spalte von A^{n_1}
- 2) die erste D Reihe von χ^{n_N}
- 3) und die erste D Reihe und Spalten von Λ^1 .

Bemerkung: Diese Prozedur setzt die orthogonierung von A und B voraus.

Um die gatte zu komprimieren müssen wir die orthogonale! faktoren weglassen.

Z.B. Verschieben von links nach rechts

- 1) Trunkieren
- 2) Orthogonalitätszerlegung
- 3) bis Ende

Illustration des Verfahrens

$$\chi_{n_1 \dots n_N} = \sum_{a_1, \dots, a_{N-1}} A_{a_1}^{n_1} \dots A_{a_{N-1}}^{n_{N-1}} A_{a_{N-1}}^{n_N}$$

$\boxed{A} - \boxed{A} - \dots - \boxed{A}$
 \downarrow rechts herum und

$$Z_{n_1 \dots n_N} = \sum_{\substack{a_1 \dots a_{N-2} \\ b_{N-1} b_N}} A_{a_1}^{n_1} A_{a_1 a_2}^{n_2} \dots A_{a_{N-2} a_{N-1}}^{n_{N-1}} U_{c_{N-1} b_N} S_{b_N} B_{b_N}^{n_N}$$

// erzwungen durch die Schmidt Basis

$$|a_{N-1}\rangle_A = \sum_{\substack{a_1 \dots a_{N-2} \\ a_{N-1}}} (A_{a_1}^{n_1} \dots A_{a_{N-2} a_{N-1}}^{n_{N-1}} U_{a_{N-2} a_{N-1}}) |n_1 \dots n_{N-1}\rangle$$

$$\Rightarrow |Z^{(N-1)}\rangle = \sum_{a_{N-1}} S_{a_{N-1}} |a_{N-1}\rangle_A |c_{N-1}\rangle_B$$

• Jetzt muss die Matrix zu $\tilde{U} \tilde{S} \tilde{B}^{n_N}$ zerlegt mit D Singularwert

• Der nächste A^{N-1} und \tilde{U}, \tilde{S} bilden M^{N-1}

$$\begin{matrix} \boxed{A} & - & \boxed{A} & - & \dots & - & \boxed{A} & - & \boxed{M} & - & \boxed{B} \\ | & & | & & & & | & & | & & | \end{matrix}$$

mit $M_{ij}^{n_N} = M_{i,j}(n_N) = \sum_k U_{ik} S_{kj} B_k(n_N) = \sum_k U_{ik} S_{kj} B_{n_N}^k$

Also

$$Z_{n_1 \dots n_N} = \sum_{b_N} A_{a_1}^{n_1} A_{a_1 a_2}^{n_2} \dots A_{a_{N-2} a_{N-1}}^{n_{N-1}} (A_{a_{N-2} a_{N-1}}^{n_{N-1}} U_{a_{N-2} a_{N-1}} S_{b_N} B_{b_N}^{n_N})$$

zerlegt

$$\sum_{\substack{a_1 \dots a_{N-2} \\ b_N}} A_{a_1}^{n_1} A_{a_1 a_2}^{n_2} \dots A_{a_{N-2} a_{N-1}}^{n_{N-1}} (\tilde{A}_{a_{N-2} a_{N-1}}^{n_{N-1}} \tilde{U}_{a_{N-2} a_{N-1}} S_{b_N} \tilde{S}_{b_N})$$

usw

$$\begin{matrix} \boxed{A} & - & \boxed{A} & - & \dots & - & \boxed{A} & - & \boxed{M} & - & \boxed{B} \\ | & & | & & & & | & & | & & | \end{matrix} \leftarrow \begin{matrix} \boxed{A} & - & \boxed{A} & - & \dots & - & \boxed{A} & - & \boxed{U} & - & \boxed{S} & - & \boxed{B} \\ | & & | & & & & | & & | & & | & & | \end{matrix}$$

M

$$= \sum_{\substack{a_1 \dots a_{N-2} \\ b_N}} A_{a_1}^{n_1} A_{a_1 a_2}^{n_2} \dots A_{a_{N-2} a_{N-1}}^{n_{N-1}} M_{a_{N-2} b_N} B_{b_N}^{n_N}$$

$$\begin{matrix} \boxed{A} & - & \boxed{A} & - & \dots & - & \boxed{A} & - & \boxed{U} & - & \boxed{S} & - & \boxed{B} & - & \boxed{B} \\ | & & | & & & & | & & | & & | & & | & & | \end{matrix}$$

- Bemerkung: 1) Die M abhängige Ergebnis der Trunkung ist
 \Rightarrow Trunkung hängt von mir ab, da wir
 kein die Trunkung unbalanciert sind!
- 2) Insbes. bei stark Trunkung in Probleme
- 3) Numerische Effizienz sehr langsam, teure
 Lösungen falls D' Index der meisten Link
 von Trunkung ist sehr viel größer als D
 Alternativen ...

Iterative Kompression von MPS

Die Kompression mit SVD an der letzten Abschnitt
 ist relativ aufwendig. Eine effiziente Methode startet
 mit Ansatz $|\tilde{\chi}\rangle$, die die Original $|\chi\rangle$
 beschreiben soll.

Mit $\| |\chi\rangle - |\tilde{\chi}\rangle \|_2^2$ kann man z.B. den Fehler abschätzen
 (Auch leicht berechnen)

Also wollen wir $\langle \chi | \chi \rangle - \langle \tilde{\chi} | \chi \rangle - \langle \chi | \tilde{\chi} \rangle + \langle \tilde{\chi} | \tilde{\chi} \rangle$
 bzgl. $|\tilde{\chi}\rangle$ minimieren.

$$\text{Sei } |\chi\rangle \leq \begin{matrix} \boxed{m} & - & \boxed{m} & \boxed{m} & \dots & - & \boxed{m} \\ | & & | & & & & | \end{matrix}$$

$$|\tilde{\chi}\rangle \leq \begin{matrix} \boxed{m} & - & \boxed{m} & - & \dots & - & \boxed{m} \\ | & & | & & & & | \end{matrix}$$

das ist stark mit lineares Optimierungsproblem.

1) Startpunkt für $|\tilde{\chi}\rangle$ im Optimierungsproblem kann z.B.
 ein mit wieder Link in SVD Verfahren erzeugte Approximation.

2) Wir sweepen von links nach rechts und rechts nach links
 durch die \tilde{m} und variieren \tilde{m} . Site i zu $i+1$
 (wobei \tilde{m}_j ($i \neq j$) fest gehalten werden)

Dabei wird \tilde{m}_i so variiert, dass der Fehler minimal wird.

$\Rightarrow \hat{M}$ bestimmt sich nur $-\langle \psi | \hat{V} \rangle + \langle \hat{V} | \psi \rangle$

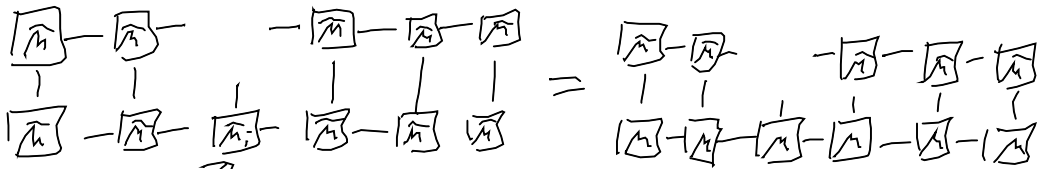
$$\frac{\partial}{\partial \hat{M}_{a_{i-1} a_i}^{n_i, x}} (\langle \psi | \hat{V} \rangle - \langle \hat{V} | \psi \rangle)$$

$$= \sum_{n_1 \dots n_N} (\hat{M}^{n_1 n_2} \dots \hat{M}^{n_{i-1} n_i})_{1, a_{i-1}} (\hat{M}^{n_i n_{i+1}} \dots \hat{M}^{n_N n_1})_{a_i, N}$$

$$- \sum_{n_1 \dots n_N} (\hat{M}^{n_N n_1} \dots \hat{M}^{n_{i-1} n_i})_{1, a_{i-1}} (\hat{M}^{n_i n_{i+1}} \dots \hat{M}^{n_N n_1})_{a_i, N}$$

$$\hat{M}^{n_1 n_2} \dots \hat{M}^{n_i n_{i+1}} \dots \hat{M}^{n_N n_1} = 0$$

Diagrammatisch



Diese Matrix wollen wir bestimmen.

$$\sum_{a_{i-1} \dots a_i} \underbrace{\hat{O}_{a_{i-1} a_i a_{i-1}' a_i'}}_{\text{Matrix}} \underbrace{\hat{M}_{a_{i-1}' a_i}}_{\text{Vektor}} = \underbrace{0_{a_{i-1} a_i}}_{\text{Vektor}}$$

$$\underline{M} \cdot \underline{v} = \underline{b} \Rightarrow \text{lineares Gleichungssystem}$$

Nach der Lsg. des Gleichungssystems können wir Lösgen.

D.h.

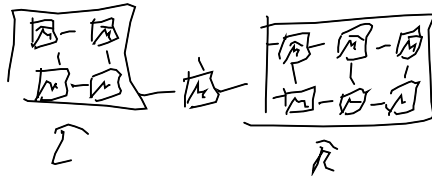
$$\hat{O} = \underbrace{\begin{pmatrix} \hat{M} & -\hat{M} & \hat{M} \\ \hat{M} & -\hat{M} & \hat{M} \end{pmatrix}}_{\sum_{a_{i-1} a_i}} \underbrace{\begin{pmatrix} \hat{M} & -\hat{M} & \hat{M} \\ -\hat{M} & \hat{M} & -\hat{M} \end{pmatrix}}_{\hat{R}_{a_i a_i'}} \equiv \hat{D}_{a_{i-1} a_i a_{i-1}' a_i'}$$

z.B.

$$\Rightarrow \sum_{a_{i-1}, a_i} = \left(\sum_{h_{i-1}} \tilde{M}^{h_{i-1}} \left(\dots \left(\sum_{h_i} \tilde{M}^{h_i} \tilde{M}^{h_i} \right) \dots \right) \tilde{M}^{h_i} \right)_{a_{i-1}, a_i}$$

$$\Rightarrow \sum_{a_{i-1}, a_i} \delta_{a_{i-1}, a_i} \tilde{M}^{a_i} = \sum_{a_{i-1}} \sum_{a_i} \left(\sum_{a_i} \tilde{R}_{a_i} \tilde{M}_{a_i}^{a_i} \right)$$

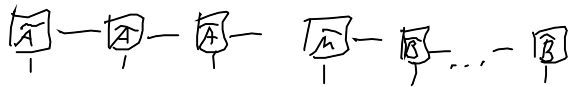
Das sieht man mit



$$\sum_{a_{i-1}, a_i} \sum_{a_i} \tilde{L}_{a_{i-1}, a_i} \tilde{M}_{a_i}^{a_i} \tilde{R}_{a_i}$$

3.) Vereinfacht durch Kammern Form!

Falls $|\mathcal{Q}\rangle$ in der gewöhnlichen Form ist



$$\text{dann } \Sigma \cong \begin{array}{c} \text{A} - \text{A} - \dots - \text{A} \\ | \quad | \quad \dots \quad | \\ \text{A} - \text{B} - \dots - \text{B} \end{array} = \left[\begin{array}{c} \text{B} \\ \text{B} \end{array} \right]$$

$$\text{oder } \sum_{a_{i-1}, a_i} = \delta_{a_{i-1}, a_i}$$

$$\text{sonst } \tilde{R} \cong \begin{array}{c} - \text{B} - \text{B} - \dots - \text{B} \\ | \quad | \quad \dots \quad | \\ - \text{B} - \text{B} - \dots - \text{B} \end{array} \quad \text{mit } \tilde{R}_{a_i, a_i} = \delta_{a_i, a_i} \left[\begin{array}{c} \text{B} \\ \text{B} \end{array} \right]$$

$$\text{wegen der Menge } \Rightarrow \underline{M} = \underline{1_M} \Rightarrow \underline{V} = \underline{1}$$

Spitzensystem ist trivial.

Das wollen wir immer her, also verstehen wir
 fünf Jahre Schritt der Orthogonalitätsrate

So sehr wir es rechts und links und vice versa.

Konvergenz wird mit $\| |\vec{r}_k\rangle - |\vec{r}_{k-1}\rangle \|^2$ überprüft.

was in der kanonischen Form $\| |\vec{r}_k\rangle - |\vec{r}_{k-1}\rangle \|^2 = 1 - \sum_i |\langle \vec{r}_k | \vec{r}_{k-1} \rangle|^2$

Achtung: Diese AA in Algorithmen können stark block
 in nicht globaler Min \Rightarrow Alternative?

Wir nutzen stattdessen ein Zwei-Site Ansatz

Das ist folgendes zuzunehmen

$$-\frac{1}{\sqrt{2}} - \frac{1}{\sqrt{2}} = \begin{array}{c} \square \quad \square \\ | \quad | \\ \square - \square \quad \square - \square \end{array} \quad - \begin{array}{c} \square - \square \quad \square - \square \\ | \quad | \\ \square - \square \quad \square - \square \end{array}$$

Das passiert das nicht so häufig

4) Wichtiges Punkt bei einem Variationsansatz ist das
 initiale Wert, ist dann nicht gut, dann die Iterati-
 onen sehr langsam konvergieren - es gibt Startwerte.