

Anwendung von MPO auf MPS

(b) Alternative Idee nach Krutzsche und Cirac

Wir verschieben in Zustand MPS $| \psi_B \rangle$ an das Produkt
 $W | \psi_A \rangle$ anzufügen!

Im Prinzip, ist das Ziel $\| | \psi_B \rangle - W | \psi_A \rangle \|^2$ zu minimieren

Also: $(\langle \psi_B | - \langle \psi_A | W^\dagger) (| \psi_B \rangle - W | \psi_A \rangle)$

$$= \langle \psi_B | \psi_B \rangle + \langle \psi_A | W^\dagger W | \psi_A \rangle - \langle \psi_A | W^\dagger | \psi_B \rangle - \langle \psi_B | W | \psi_A \rangle$$

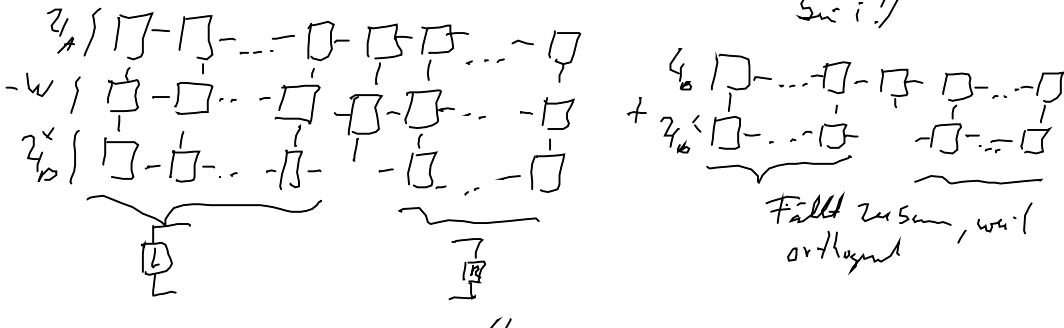
Wir wollen $| \psi_B \rangle$ bestm. approximieren. Wir spalten nach ψ_B^x ,
 nur das — relevant

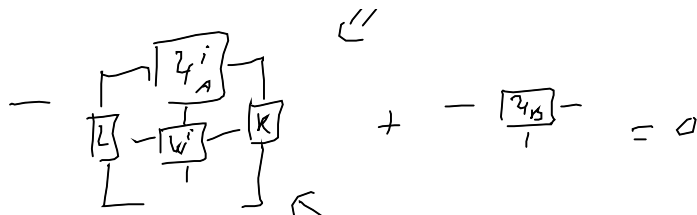
Diagramm

$$\frac{\partial}{\partial \psi_B^x} \left(\begin{array}{c} \psi_A | \square - \square - \square - \dots - \square \\ -W | \square - \square - \square - \dots - \square \\ \psi_B^x | \square - \square - \square - \dots - \square \end{array} + \begin{array}{c} \psi_B^x | \square - \square - \square - \dots - \square \\ \psi_B^x | \square - \square - \square - \dots - \square \end{array} \right) = 0$$

$$\frac{\partial}{\partial \psi_B^{x_i}} \left(- \sum_{\substack{h_1 \dots h_p \\ a_1 \dots a_{p-1} \\ b_1 \dots b_{p-1} \\ n_1 \dots n_p \\ w_1 \dots w_{p-1}}} \psi_A^{h_1} \psi_A^{h_2} \psi_A^{h_3} \dots \psi_A^{h_p} W_{w_1}^{h_1 h_1'} W_{w_2}^{h_2 h_2'} \dots W_{w_p}^{h_p h_p'} \right. \\ \left. + \sum_{\substack{h_1 \dots h_p \\ b_1 \dots b_{p-1} \\ b_1 \dots b_{p-1}}} \psi_B^{h_1} \psi_B^{h_2} \dots \psi_B^{h_p} \psi_B^{h_p} \psi_B^{h_{p-1}} \psi_B^{h_{p-1}'} \right) = 0$$

Manchmal wir es lieber unter Diagrammform (Sei das orthogonale Links-Zahn
 Sei i!)

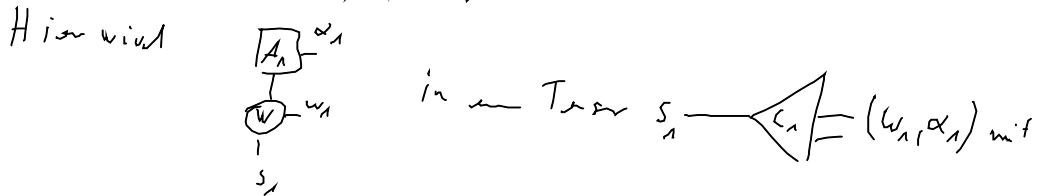




Diese Tensor beschreibt und beschreibt durch $-Z_B$

- Bemerkung:
- (i) Dieser Algorithmus wird wie DMRG, wird durch sweepen unter Verlegen des Orthogonalitätszentrums berechnet (jeweils die obige Seite).
 - (ii) Vorteil dieses Verfahrens skaliert $Nm^3k + Nm^2k^2$ also sehr viel besser.
 - (iii) In der Praxis sind Variablen mit Zwiets, verzerrt.
 - (iv) Konvergenz hängt von Anfangszustand für Z_B ab, falls der Anfangszustand schlecht ist, teilweise kein Konvergenz.

c) Welche Methode ZipUp Algorithmus



ein fester Index zugeht, ein SVP führt auf

$$S_1 = \frac{\alpha_1}{\beta} \frac{\alpha_1}{\beta} \frac{\alpha_1}{\beta} = (w_1, \alpha_1)$$

Das Verfahren wird die fortgesetzt

$$\beta_2 = \frac{\alpha_2}{\beta} \frac{\alpha_2}{\beta} \frac{\alpha_2}{\beta} = (\beta_2, S_1) = (w_2, \alpha_2)$$

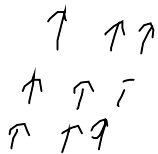
SVD fakt of B_z s_z

$$P_1 - \begin{matrix} \boxed{U} \\ \downarrow \\ s_z \end{matrix} - \begin{matrix} \boxed{V} \\ \downarrow \\ s_z \end{matrix} \left\langle \begin{matrix} \boxed{L} \\ \downarrow \\ s_z \end{matrix} \right\rangle = (u_2 \ u_2), \text{ usw.}$$

d) Auch ein komplettes Orthogonality über die Dichtematrix (siehe Vorlesung zur kanonischen Form) kann benutzt werden, um das genaue Anzählen. Diese Funktion auf alle Variablen ist aber bisher nur in ihrer Code realisiert.

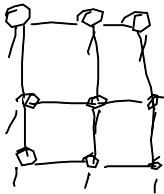
VIII.24 Ausblick auf weitere Tensorstrukturen (insbesondere 2D PEPS)

Bisher nur 1D Spinketten, aber was ist mit 2D Gittern.



, wie kann man diese darstellen?

Natürliche Erwartung 2D PEPS (Projected Entangled Pair States), siehe gutes Review Roman Orús, Annals of Physics, 349, 117-158 (2019)



1) Sieht einfach aus, aber können wir damit rechnen?

Ja, aber meist wird das Problem MPS+ (unzureichende Kopplung).

2) PEPS liess wie MPS direkt in Hilbertraum (Erhöhung der Linkdimension)

3) PEPS keine polynomiale abstellende Kantenstruktur, MPS ortsfest immer exponentiell und kann damit beschränkt (Relevant für Gitter)

4) Die exakte Bedeutung des Skalarproduktes skaliert linear in N für MPS ist aber $\#P$ -Hard (skaliert mit $\exp(N)$) für PEPS!

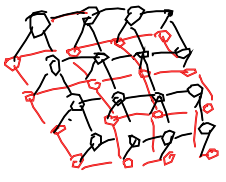
Was ist im Problem!

- (5) Kein exakte Konvergenzform für period. PEPS wie für MPS
(aber es ist links und rechts in Schmidt...)
Es ox. ab. Quasi 2D Zerlegen...

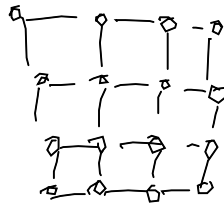
Erwartungswerte für PEPS

Bei MPS normalerweise $O(N_p D^3)$

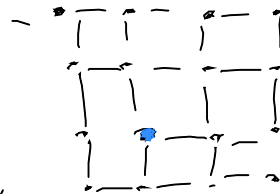
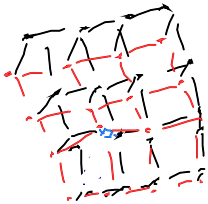
Was ist das Problem bei PEPS?



- 1) Es geht, wie man \rightarrow kontrahiert \rightarrow skaliert
immer exponentiell in Länge



Das funktioniert natürlich auch für Erwartungswerte



← Alternativen (TM Methode (Corner Trench Matrix))

Darüber ex. DMRG Algorithm

