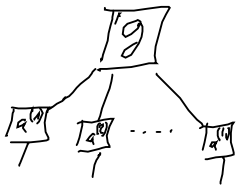
Tucker-Decomposition

Nächste Idee für hoch dimensionale Schnitte:

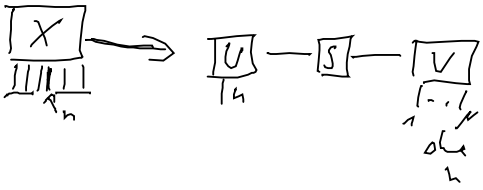
$$T_{n_1 n_2 n_3 \cdots n_k} = \sum_{k_1 \cdots k_k} g_{k_1 \cdots k_k} a^{(1)}_{k_1 n_1} a^{(2)}_{k_2 n_2} \cdots a^{(k)}_{k_k n_k}$$



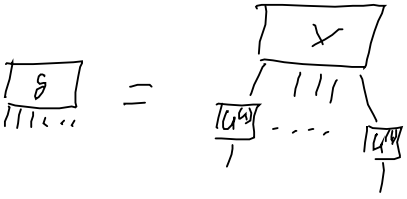
1) Der wesentliche Punkt für die Kompression ist, dass der Core Tensor g geringere Dimensionalität als der ursprüngliche Tensor.
2.) Jeder Tensor kann in diese Form gebracht werden, aber die Dimension von g muss nicht kleiner sein.

Berechnung der Tucker Decomposition mit HOSVD

HOSVD ($X_{k_1 \cdots k_\mu}$)
for $n = 1, \ldots, N$ do
  Fasse alle Indizes k_i ($i \neq n$) zu einem Index zusammen und führe die SVD durch (Nehme nur die wichtigsten Singulär Werte mit)
  Speichere die linken singulär Vektoren $U^{(n)}$
end



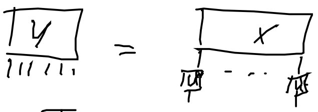
$$g_{n_1 \cdots n_\mu} = \sum_{k_1 \cdots k_\mu} U^{(1)}_{n_1 k_1} \cdots U^{(\mu)}_{n_\mu k_\mu} X_{k_1 \cdots k_\mu}$$



Bemerkung

- Meist ist die Approximation nicht die effektivste, meist wird noch ein ALS Fitting algorithmus
- Oder es wird ein iterativer Verfahren verwendet:

  HOOI ($X_{n_1 \cdots n_\mu}$)
  initialisiere $U^{(n)}$ für $n = 1, \ldots N$ mit HOSVD
  repeat
    for $n = 1, \ldots, N$ do



Führe SVD für Y aus bezgl. n Dimen, wie bei HOSVD
Erneuere $U^{(n)}$

until / und solange wie die Approximation nicht verbessert

und test.

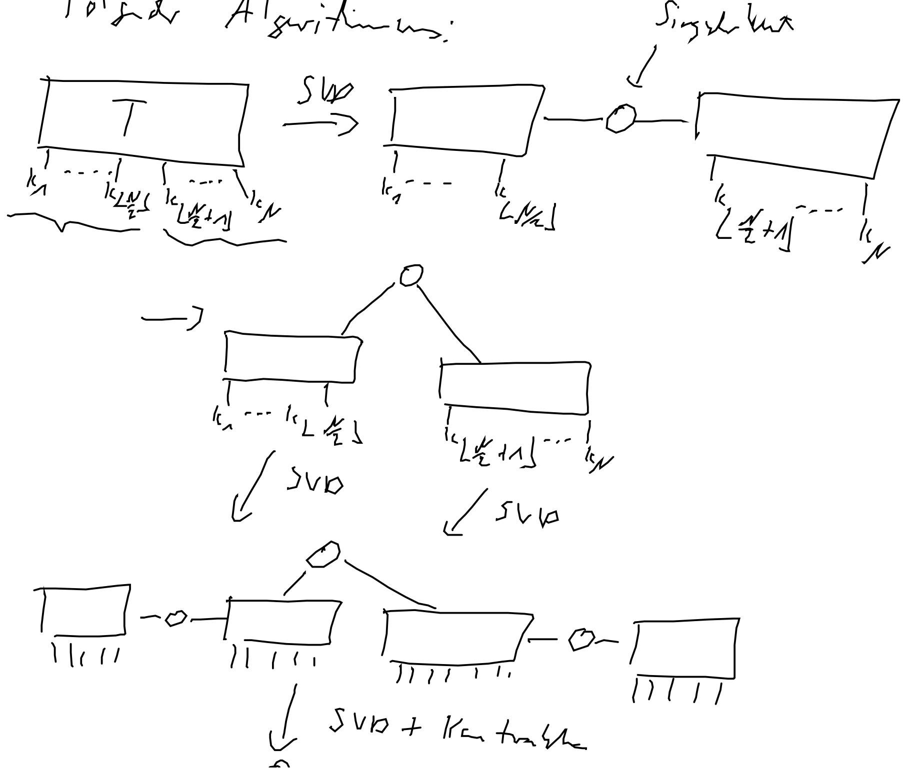
Problem: - Keine Möglichkeit den Tensor leicht zu updaten
immer komplett füllen notwendig (keine Dynamik, keine Eigenwert)

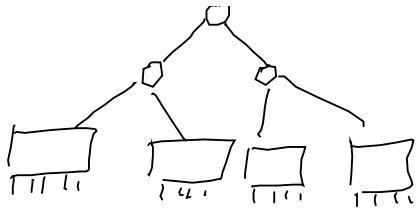
Bemerkung: Es gibt noch weitere Varianten dieser Zerlegungen.

Hierarchical Tucker

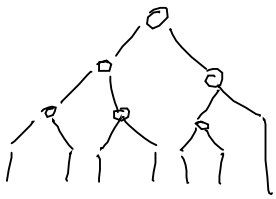
Die Idee hinter Hierarchical Tucker, ist nicht die Decomposition an einem Tensor durchzuführen, sondern hierarchisch verzweigt..

Folgender Algorithmus:





und so weiter

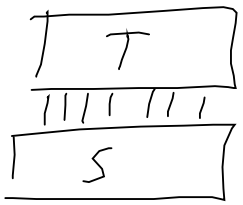


Das hierarchische Tucker Format ist wie ein Binärer Tensor Baum

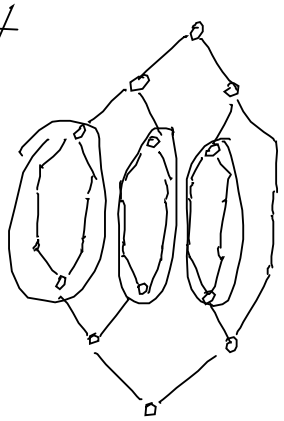
Kann wir mit dem HTucker Format rechnen?

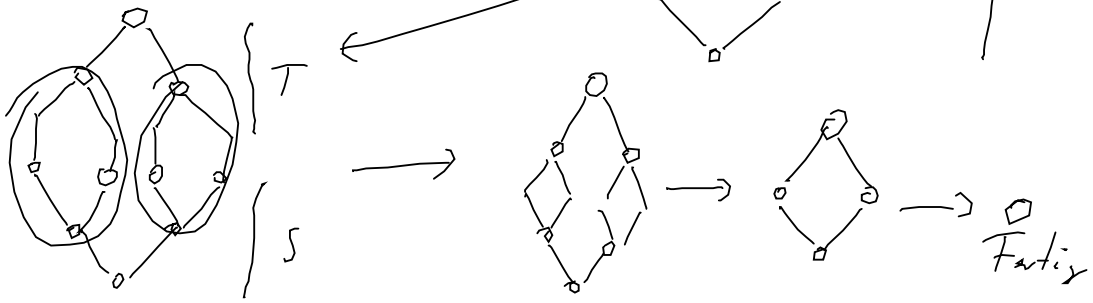
Ja, klar!

1) Norm und Skalarprodukt



ist die Form für den vollen Tensor





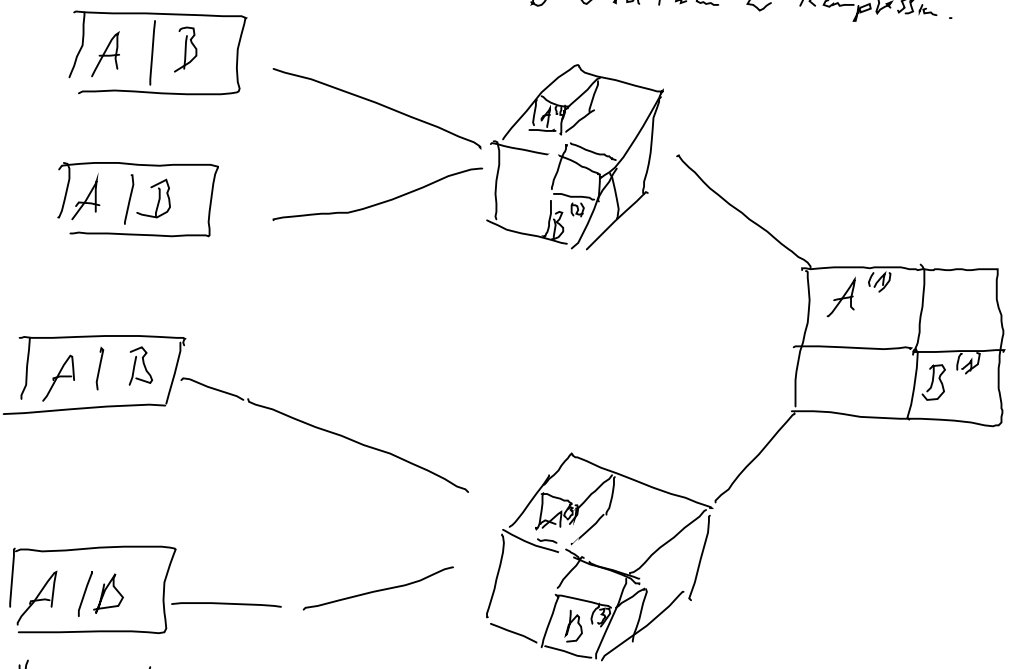
2.) Addition

Sehen wir zunächst von zwei Matrizen aus in SVD

$A = U_A S_A V_A \qquad B = U_B S_B V_B$ dann gilt

$$A + B = [U_A \; U_B] \cdot \begin{pmatrix} S_A & 0 \\ 0 & S_B \end{pmatrix} \cdot [V_A \; V_B]^T$$

$\Rightarrow$ Das sollte dann wieder eine SVD durch für Kompression.



3) Orthogonalisierung

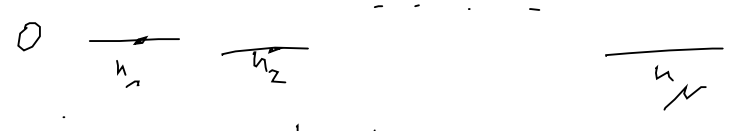
Es gibt für jede Zerlegung Regeln wie die Tensoren orthogonalisiert werden können und hat zwei Einflüsse (i) Trunkierung, (ii) Teilung in Norm und Skalarprodukt schnell berechnet wird.

S. Bestehorn MATHICSE Technical report 4 2012
"htucker - a matlab toolbox for tensors in the hierarchical tucker format".

Hier besteht das Problem die Tensoren zu manipulieren.

VIII.2.1 Matrix Product States und Matrix Product Operators

Quanteninformatik ist ein großes Forschungsgebiet,
Grundbausteine sind meist Qubits (aka Zweiniveausyst.)



in anderen Kontext nennt man das auch Spin-1/2 Ketten oder Bose Hubbard model

Wenn n_i der Zustandsindex des i Qubits ist, die Gesamtwellenfkt $\psi_{n_1 n_2 \dots n_N}$ ein Tensor!

Zur Simulation muss die Schrödingergl. gelöst werden:

$i\hbar \partial_t |\psi\rangle = H|\psi\rangle \Leftrightarrow i\hbar \partial_t \psi_{n_1 \dots n_N} = \sum_{n_1' \dots n_N'} H_{n_1 \dots n_N\, n_1' \dots n_N'} \psi_{n_1' \dots n_N'}$

mit $|\psi\rangle = \sum_{n_1 \dots n_N} \psi_{n_1 \dots n_N} |n_1 \dots n_N\rangle$

$\Rightarrow$ TODO Tensornetzwerk für ψ und für H finden!

$\Rightarrow$ Matrix Product States (Vidal PRL 91 (14) 147902 (2003)
(schöne Zusammenfassung zum Schöllwock
arxiv: 1008.3477v2 (2011)