

TeNeS

Tensor Network Solver for Quantum Lattice Systems

<https://www.pasums.issp.u-tokyo.ac.jp/tenes>

<https://github.com/issp-center-dev/TeNeS>

YM and et al., Comput. Phys. Commun. **279**, 108437 (2022)

YM and et al., Comput. Phys. Commun. **315**, 109692 (2025)

質問・要望は
GitHub のissue か
メール (tenes-dev@issp.u-tokyo.ac.jp)へ！

東大物性研 本山裕一

2026-02-02 @ 26回HPC-Phys勉強会

Outline

- ・ 自己紹介
- ・ 物性研ソフトウェア開発高度化プロジェクト (PASUMS) について
- ・ 二次元量子格子模型の計算手法としてのテンソルネットワーク法
- ・ 二次元量子格子模型のテンソルネットワークソルバー TeNeS
- ・ まとめ

自己紹介

- ・ 名前：本山裕一
- ・ 専門は**計算物理（特に統計物理と量子物理）** および**ソフトウェア開発**
- ・ 2014/03 東大院工学系研究科物理工学専攻博士課程修了
 - ・ 量子スピン系の局所ベリー位相を経路積分モンテカルロ法で計算するアルゴリズムの開発
- ・ 2014/04 - 2022/03 東大物性研・特任研究員
 - ・ 連続空間の経路積分モンテカルロ法
 - ・ 虚時間グリーン関数の安定な解析接続手法の開発
 - ・ ソフトウェア開発・高度化プロジェクト（2018-）
- ・ 2022/04 - 現在 東大物性研・技術専門職員
 - ・ **ソフトウェア開発・高度化プロジェクト**

物性研ソフトウェア開発・高度化プロジェクト PASUMS

- ・ **東大物性研**は全国共同利用・共同研究拠点であり、日本全国の物性研究者に実験装置や**スパコン**を全国共同利用課題という形で提供している
 - ・ 実際の研究では、スパコンというハードウェアだけではなく、**スパコンで動くソフトウェア**も必要
 - ・ アルゴリズムの高度化や並列化対応など、実装がどんどん難しくなっていく
 - ・ 個人あるいは研究グループが独立したソフトウェアを持つよりも、更に広い**研究コミュニティ全体**としてソフトウェアを開発・利活用したほうが効率的
 - ・ 研究者が持っているコードを使いやすく整備して、公開する支援をしよう！
- ・ 2015年度に**物性研ソフトウェア開発・高度化プロジェクト** (PASUMS: Project for Advancement of Software Usability in Materials Science) がスタート

物性研ソフトウェア開発・高度化プロジェクト PASUMS

- ・ 全国の物性物理学の研究者から課題ソフトウェアを公募として募集（年1回）
 - ・ 必要な申請資格は物性研スパコンの共同利用課題と同じ
- ・ 物性研スパコンの運営委員会による書類審査およびヒアリング審査
 - ・ 運営委員会の半数は物性研所外の研究者
- ・ 年度ごとに2件程度を課題として採択し、高度化支援（次ページ）
- ・ 成果物は、コミュニティに資するものとして、**オープンソースソフトウェア**としてリリースする
 - ・ 主にGPLv3などのコピーレフトライセンス
 - ・ そのままコミュニティにとって自由なものであり続けてほしいという願い

PASUMSの主な支援内容

- ・ **一般利用者向けのユーザビリティ向上（広く使われるようにする）**
 - ・ **マニュアル・チュートリアル**の整備
 - ・ 入出力の整備、一般化
 - ・ よく使われるモデルやパラメータを使いやすくする
 - ・ 有用な手法・機能の実装
 - ・ **広報・普及活動**
 - ・ ソフトウェア論文執筆
 - ・ **ソフトウェア講習会の開催**
 - ・ 物性研スパコンへのプリンストール
- ・ **開発者向けのユーザビリティ向上（開発が持続するようにする）**
 - ・ リファクタリング
 - ・ 開発者向けマニュアル（API リファレンスなど）
 - ・ テストの導入および自動化

PASUMSで支援したソフト

- ・ 量子格子模型

- ・ H-Wave
 - ・ 平均場近似・RPA
- ・ HΦ
 - ・ 厳密対角化
- ・ mVMC
 - ・ 変分MC
- ・ DSQSS
 - ・ 経路積分MC
- ・ **TeNeS**
 - ・ **テンソルネットワーク**
- ・ DCore
 - ・ 動的平均場理論
- ・ ChiQ
 - ・ Bethe-Salpeter

- ・ 第一原理計算

- ・ abICS
 - ・ 不規則系の第一原理モンテカルロ計算
- ・ OpenMX
 - ・ 局在基底DFTソルバー
- ・ RESPACK
 - ・ 有効模型導出ソフトウェア
- ・ ESM-RISM
 - ・ 古典溶液理論に基づく第一原理固液界面ソルバー

- ・ データ科学・数理ライブラリ

- ・ ODAT-SE (旧2DMAT)
 - ・ 最適化およびベイズ推定
- ・ PHYSBO
 - ・ ベイズ最適化
- ・ Kw
 - ・ Shifted-Krylov 部分空間法
- ・ moller
 - ・ 並列計算機向け網羅計算ツール

二次元量子格子模型の計算手法としての
テンソルネットワーク法
(TeNeSに実装されているアルゴリズム)

量子多体問題 in 物性物理

- ・ 多体ハミルトニアン \mathcal{H} の解析がメイン
 - ・ 特に、固有値問題 $\mathcal{H} |\Psi_n\rangle = E_n |\Psi_n\rangle$
 - ・ ヒルベルト空間の次元が粒子数に対して指数関数的に増えるので、固有値問題を直接解くためのコストも指数関数的に増える
- ・ 物性物理で取り扱う温度・エネルギースケールは低温領域
 - ・ **基底状態 $|\Psi_0\rangle$ や低エネルギー励起状態 $|\Psi_1\rangle, |\Psi_2\rangle, \dots$ さえ求まれば良い**
 - ・ → 変分法が有用

変分法による基底状態探索

- 量子力学における変分原理

- 任意の状態 $|\Psi\rangle$ に対するハミルトニアン \mathcal{H} の期待値 $E[\Psi]$ は、基底エネルギー E_0 と同じか、それよりも大きくなる

$$E[\Psi] := \frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} \geq E_0$$

- $E[\Psi] = E_0$ となるとき、 $|\Psi\rangle = |\Psi_0\rangle$

- 変分法

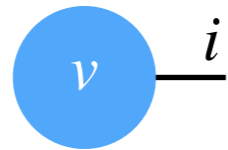
- 基底状態を探索する空間を制限し、その中でエネルギーを最小化する
 - 試行関数（変分波動関数）の形を仮定することで空間を制限する
 - 変分波動関数の作り方やパラメータの数で、手法の効率性や得られる精度をコントロール可能

- 変分波動関数の表現にテンソルネットワーク (TN) を用いるのが今日の話**

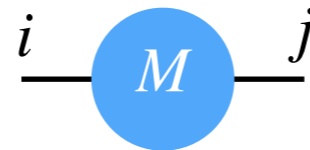
テンソルのダイアグラム表記

- ・（「テンソル」という言葉が意味するのは「多次元配列」程度の内容です）
- ・ テンソルを丸や四角など適当なオブジェクトで図示
 - ・ テンソルの添字はオブジェクトから生えている線（脚）で表現
 - ・ つまり、**n階のテンソルにはn本の脚が生えている**

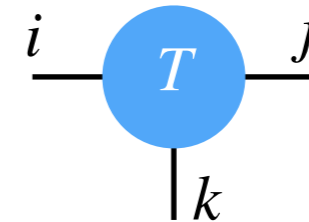
ベクトル v_i



行列 M_{ij}



テンソル T_{ijk}



- ・ **縮約（テンソルの積）は脚を接続することで表現する**

$$C_{ij} = \sum_k A_{ik} B_{kj} \rightarrow \begin{array}{c} i \\ \text{---} \end{array} \text{---} \text{---} \begin{array}{c} j \\ \text{---} \end{array} = \begin{array}{c} i \\ \text{---} \end{array} \text{---} \begin{array}{c} k \\ \text{---} \end{array} \text{---} \begin{array}{c} j \\ \text{---} \end{array}$$

量子多体状態のTN表現

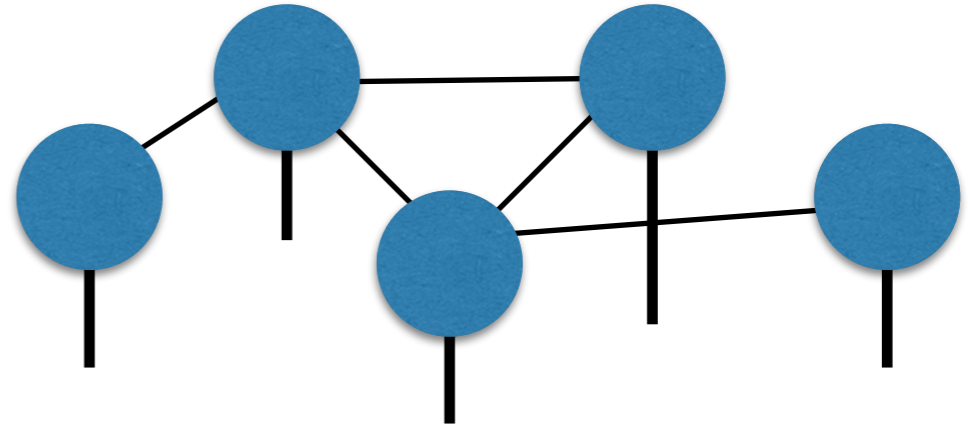
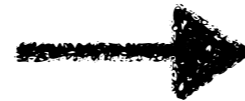
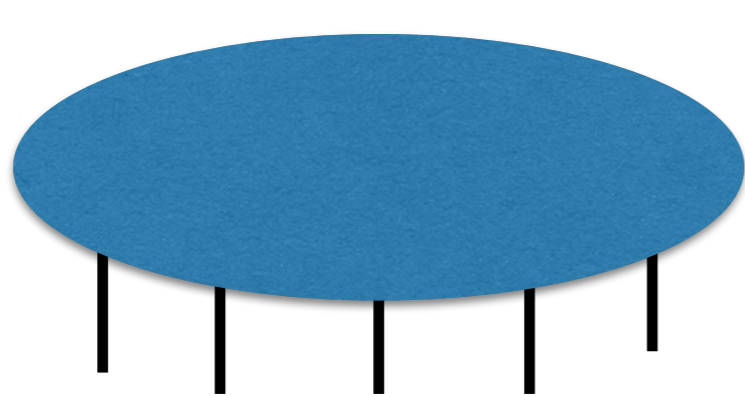
任意の波動関数 $|\Psi\rangle$ は適当な完全直交基底を用いて展開できる

- 例えば格子上に並んだ N 個の $S=1/2$ スピンからなる系では、それぞれのスピンの z 成分を対角化する基底 $|\sigma_i\rangle$ の直積状態 $|\sigma_1\sigma_2\dots\sigma_N\rangle$ を用いて

$$|\Psi\rangle = \sum_{\sigma_1, \sigma_2, \dots, \sigma_N} \Psi_{\sigma_1\sigma_2\dots\sigma_N} |\sigma_1\sigma_2\dots\sigma_N\rangle \text{ と厳密に展開できる}$$

展開係数 $\Psi_{\sigma_1\sigma_2\dots\sigma_N}$ は N -rank テンソル (要素数が $O(e^N)$)

- この巨大なテンソルを $O(N)$ 個の小さなテンソルに分割する
- 元々の脚を **physical bond**, テンソル同士をつなげるための脚を **virtual bond** と呼ぶ



どうつなげるのがよい？

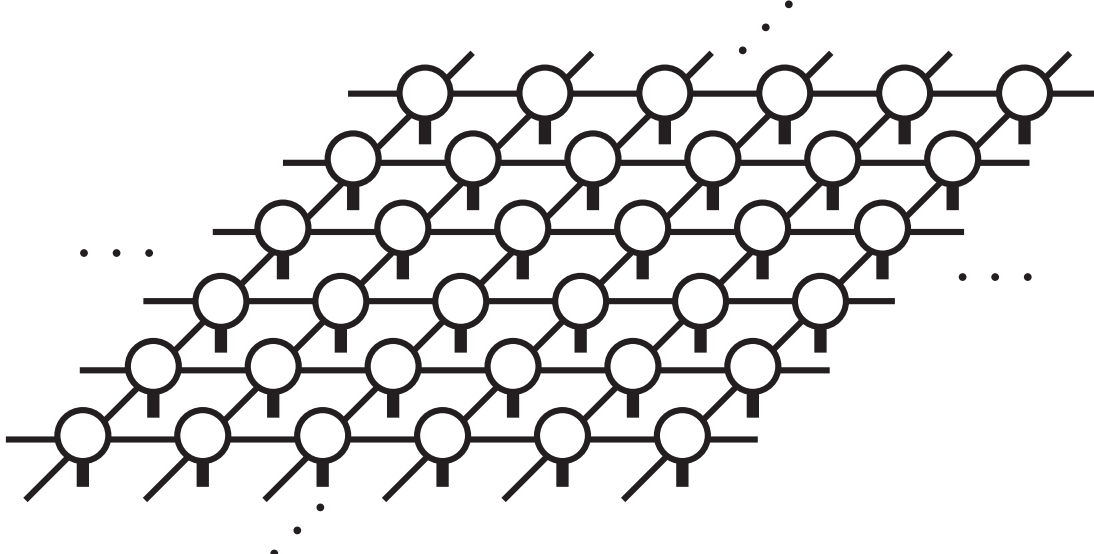
エンタングルメントエントロピーと面積則

- ・ 全系を2つの部分系 A, B に分ける
- ・ 縮約密度行列
 - ・ 密度行列 $\rho = |\Psi\rangle\langle\Psi|$ について、Bの自由度を縮約したものの $\rho_A = \text{Tr}_B |\Psi\rangle\langle\Psi|$
- ・ エンタングルメントエントロピー (EE)
 - ・ ρ_A の von Neumann エントロピー $S_A = -\text{Tr} \rho_A \log \rho_A$
 - ・ AとBがどのくらいエンタングルしているかの指標
- ・ 一般には S_A は A の体積に比例する
- ・ 一方で基底状態や低エネルギー励起状態では、Aの表面積(Bとの界面)に比例する (面積則)
 - ・ **最低限、面積則を満たすようなTNをつくれればよい**

テンソル積状態

- ・ **テンソル積状態** (Tensor Product State, **TPS**)
 - ・ あるいは Projected Entangled-Pair State (PEPS)
- ・ テンソル間の相関はvirtual bondを通して伝わる
 - ・ テンソルネットワークが表現できるエンタングルメントエントロピーの上限は、A,Bの境界にあるvirtual bondの本数に比例する
- ・ たとえば格子模型の格子と同じ形でテンソルを並べ、近接サイト同士をvirtual bondでつなげる

- ・ **面積則をみたます！**

- ・ virtual bondの次元（自由度）を $\Psi^{\text{iTPS}} =$  **ボンド次元** D と呼ぶ

- ・ 波動関数の表現精度をコントロールするパラメータ

$$T_{ijkl}[s] = \begin{array}{c} \text{\scriptsize } i \\ \diagup \quad \diagdown \\ \text{\scriptsize } j \\ \text{\scriptsize } l \quad \text{\scriptsize } k \\ \text{\scriptsize } s \end{array}$$

iTPSに対する物理量計算

- 与えられたiTPSに対する物理量（演算子）の期待値を計算したい

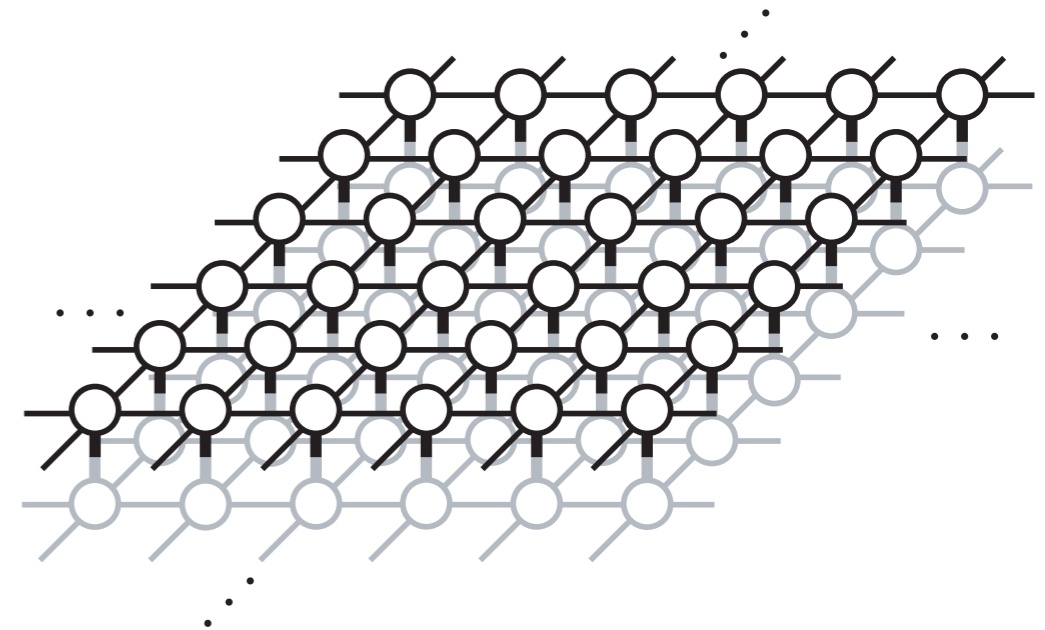
$$\langle \hat{O} \rangle = \frac{\langle \Psi | \hat{O} | \Psi \rangle}{\langle \Psi | \Psi \rangle}$$

- 分母分子ともに、ブラとケットの内積計算が必要
 - ダイアグラムとしては、対応するphysical bond を繋げれば良い
 - ブラとケットの二層あるので **double layered network** とよばれる
- 内積というスカラー値を得るためには、すべてのテンソルを縮約しなければならない

- 厳密な計算は困難（不可能）！
- 近似的な縮約手法が提案されてきた

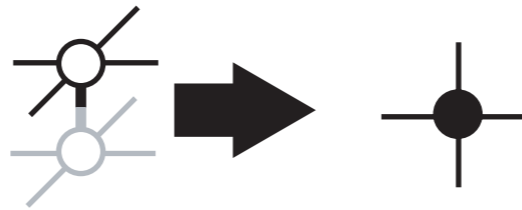
- テンソル繰り込み群
- 境界行列積法(boundary MPS)
- 角転送行列繰り込み群法 (CTMRG)**
- 平均場環境**

$$\langle \Psi | \Psi \rangle =$$



角転送行列繰り込み群法 CTMRG

- まずは一つ一つのテンソルについて、physical bondの縮約をとる



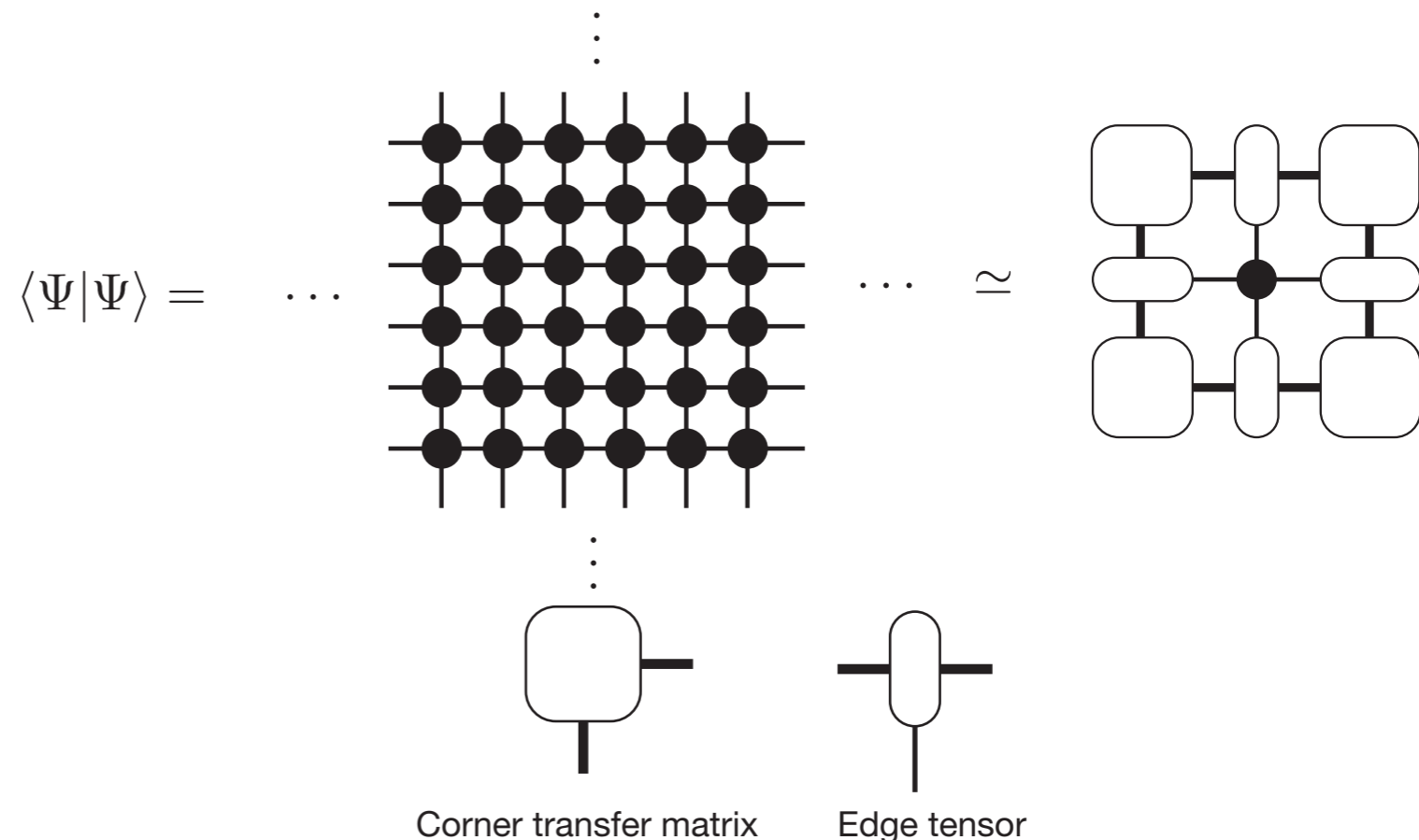
- あるテンソル（あるいはユニットセル）に着目し、その周りにある、無限に広がったテンソルをまとめて**角転送行列**と**エッジテンソル**とで近似する

- これらのボンド次元を χ と書く

- 一般に $\chi \propto D^2$ が必要

- 角転送行列やエッジテンソルが得られれば縮約が取れる

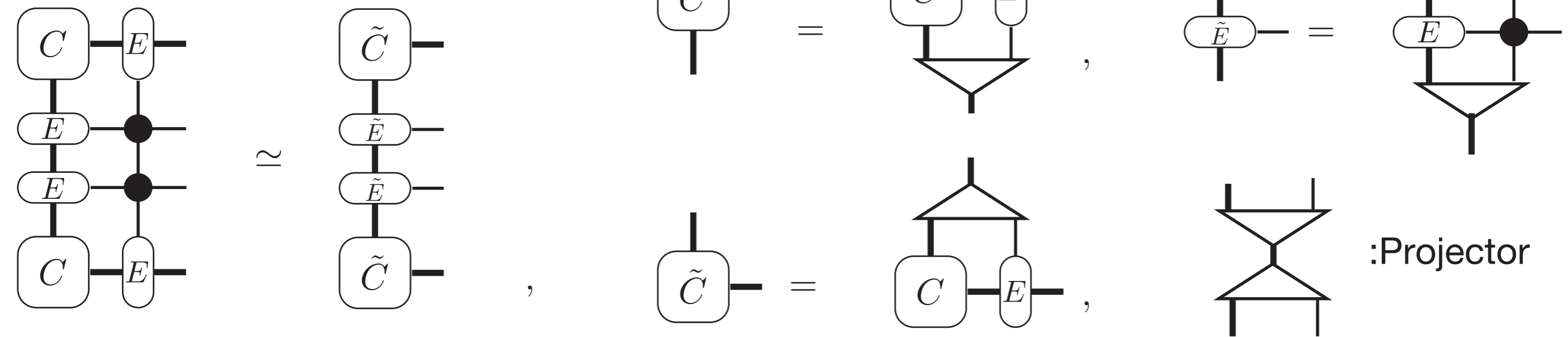
- これらのテンソルをどうやって計算するか？→CTMRG



角転送行列繰り込み群法 CTMRG

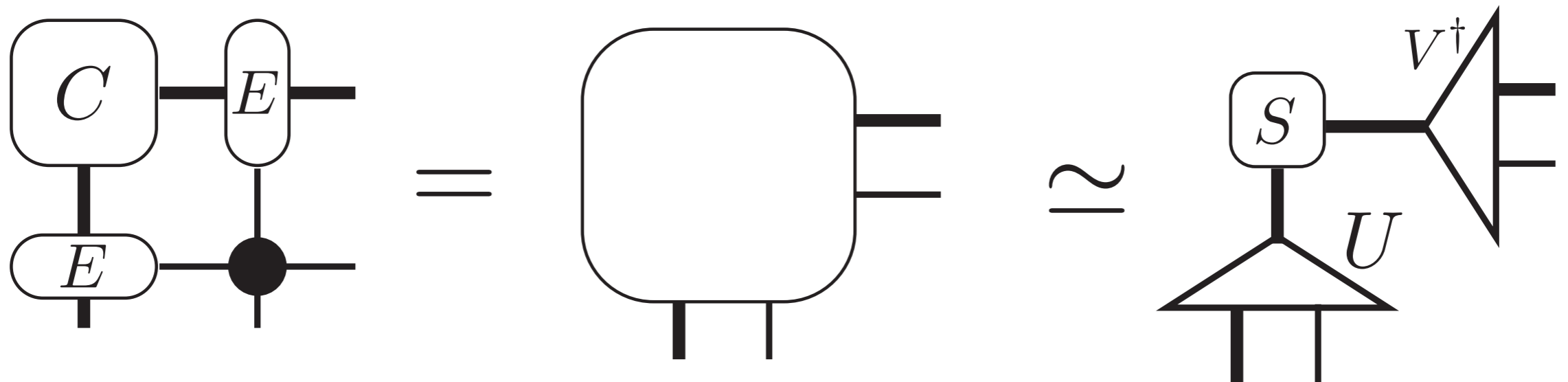
- 角転送行列 C とエッジテンソル E に、隣のテンソルを吸収（縮約）していく
 - （正方格子の場合）上下左右の4方向に縮約していく
 - 収束するまで繰り返す
- 単純にくっつけるとボンド次元が $\chi \rightarrow \chi D^2$ が増えてしまう
 - $\chi D^2 \rightarrow \chi$ にする（近似）ために**プロジェクター**（下図の三角形）を導入する

Left move in CTMRG



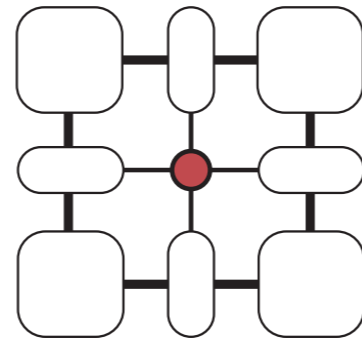
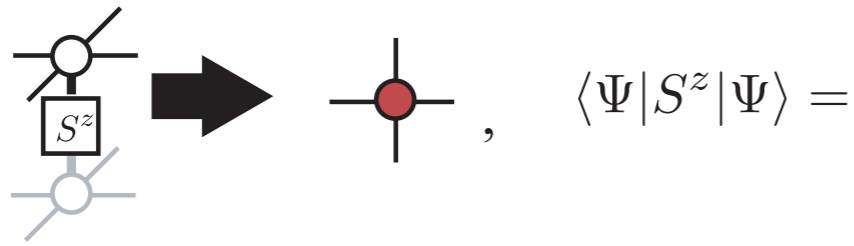
角転送行列繰り込み群法 CTMRG

- ・ プロジェクターの作り方もいくつか方法がある
 - ・ テンソルの一部を縮約して**特異値分解SVD**を行い、特異値の大きいものから χ 個残し、特異ベクトルをプロジェクターにするのが基本戦略
- ・ SVDの計算コストが $O(D^6\chi^3)$ ($\chi \propto D^2$ とすると、 $O(D^{12})$)
 - ・ **CTMRGで一番重い部分**
 - ・ 乱択SVDを採用すると $O(D^{10})$ まで落とせる
 - ・ 最近では $O(D^9)$ のアルゴリズムも提案されている (TeNeS未実装)

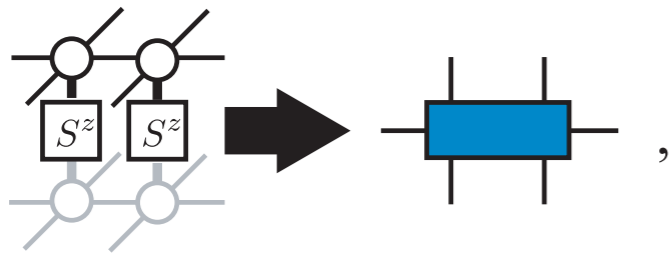
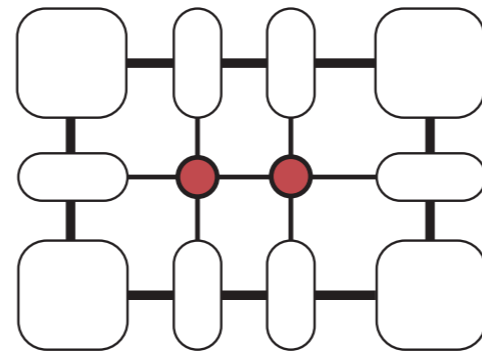


物理量計算

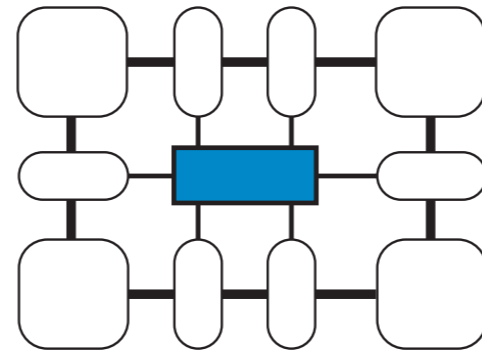
- 角転送行列がわかれば、局所的な物理量が計算できる
- 長距離になるとダイアグラムが大きくなり、計算コストも大きくなる



$$\langle \Psi | S_i^z S_{i+1}^z | \Psi \rangle =$$



=



iTPSの最適化

- ・ 変分最適化法
 - ・ エネルギー期待値の微分を計算し、最小値を目指す
 - ・ 自動微分ライブラリの登場により、最近主流になりつつある
- ・ **虚時間発展法**
 - ・ 適当な初期状態 $|\psi\rangle$ に虚時間発展演算子をかけることで基底状態を得る
$$|\Psi\rangle \propto \lim_{\beta \rightarrow \infty} e^{-\beta \mathcal{H}} |\psi\rangle$$
 - ・ 基底状態計算以外にも計算ルーチンを流用できる
 - ・ 実時間発展演算子に変えることで**実時間発展計算**ができる
 - ・ 密度行列の虚時間発展として**有限温度計算**ができる
 - ・ TeNeSはこちらを採用

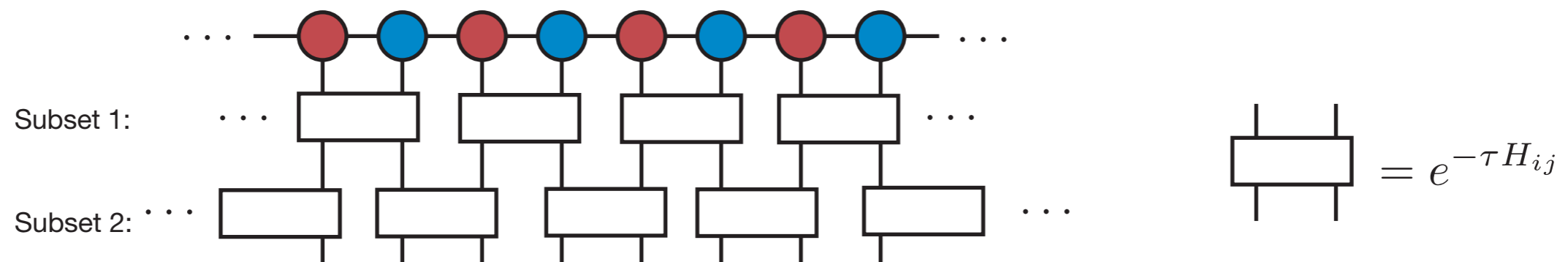
(虚) 時間発展

虚時間発展演算子も巨大なテンソルなのでそのままでは計算できない

多くの系ではハミルトニアンが局所ハミルトニアンで $\mathcal{H} = \sum_{\langle ij \rangle} \mathcal{H}_{ij}$ とかけ

るので、Suzuki-Trotter分解 $e^{-\tau \mathcal{H}} = \prod_{\langle ij \rangle} e^{-\tau \mathcal{H}_{ij}} + O(\tau^2)$ を用いて

$$|\Psi\rangle \propto \lim_{\beta \rightarrow \infty} e^{-\beta \mathcal{H}} |\psi\rangle = \propto \lim_{M \rightarrow \infty} [e^{-\tau \mathcal{H}}]^M |\psi\rangle = \propto \lim_{M \rightarrow \infty} \left[\prod_{\langle ij \rangle} e^{-\tau \mathcal{H}_{ij}} \right]^M |\psi\rangle + O(\tau)$$

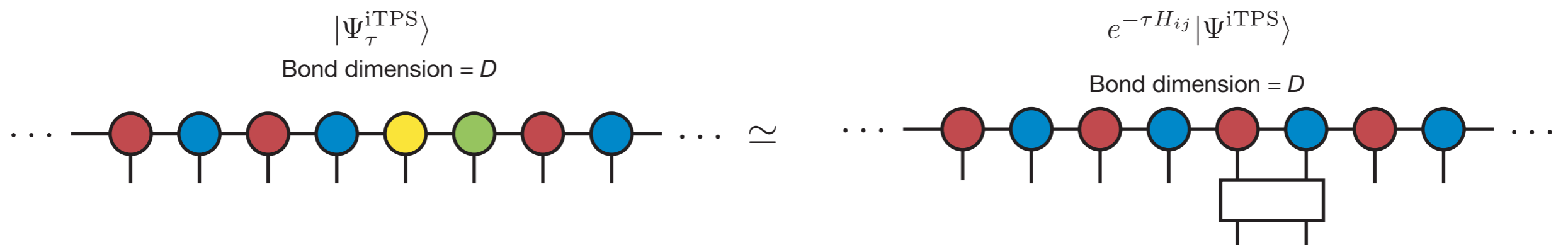


Full update

- ・ 実際には一度にひとつずつ $e^{-\tau \mathcal{H}_{ij}}$ をかけていく
- ・ 演算のたびにボンド次元が増えていくので、毎回 D まで打ち切る
- ・ 打ち切りの前後でベクトルの差の二乗ノルムが最小となるように打ち切る

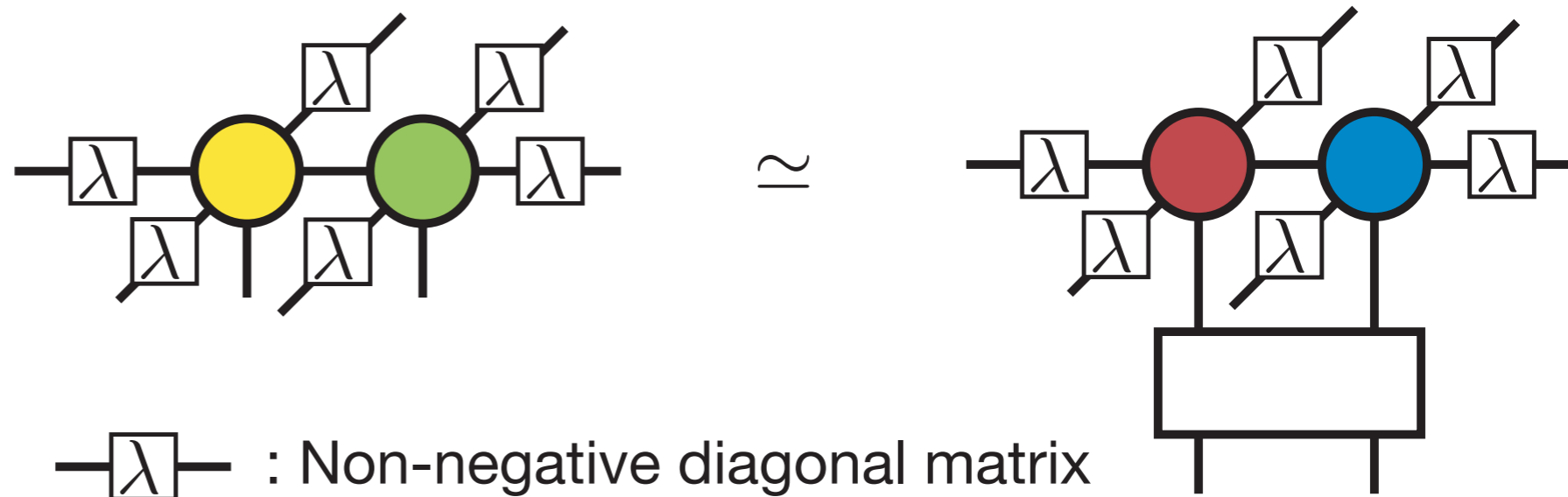
$$\min \|\Psi_\tau^{\text{iTPS}}\rangle - e^{-\tau \mathcal{H}_{ij}} |\Psi^{\text{iTPS}}\rangle\|_2^2$$

- ・ 二乗ノルムは **CTMRG** を使って計算できる
- ・ **時間発展の計算量はCTMRGの計算量と同じ**



Simple update

- Full updateは精度が良いが、いかにせん計算コストも重い
 - 乱択SVDを使っても $O(D^{10})$
- 更新したいテンソルの周りの環境を対角行列 λ まで近似 (**平均場環境**)
- λ ごと全部縮約してSVDし、特異値 σ を上からD個だけ残す
 - 特異値行列がこのボンドの新たな平均場になる ($\lambda_{\text{new}} = \sqrt{\sigma}$)
- **とても軽い!** ($O(D^5)$)
- **初期状態依存性が強い**



Full update と Simple update

- Full update
 - 計算精度が良い
 - 計算コストが重い $O(D^{10})$
- Simple update
 - 計算コストが軽い $O(D^5)$
 - 計算精度は悪い
 - 初期状態依存性がFull updateより強い
 - FUよりは軽いので、複数の初期状態で計算を行い、エネルギーを比較するのが良い
 - 途中で得られる平均場環境 λ を用いると物理量計算も軽量に行える
 - 変分原理を破るので、直接エネルギーの大きさを比較できない

有限温度計算

有限温度 $\beta = 1/k_B T$ での密度行列 $\rho(\beta) = e^{-\beta \mathcal{H}}$ を用いると、有限温度での演算子

\hat{O} の期待値は

$$\langle \hat{O} \rangle_\beta = \frac{\text{Tr} [\rho(\beta) \hat{O}]}{\text{Tr} \rho(\beta)}$$

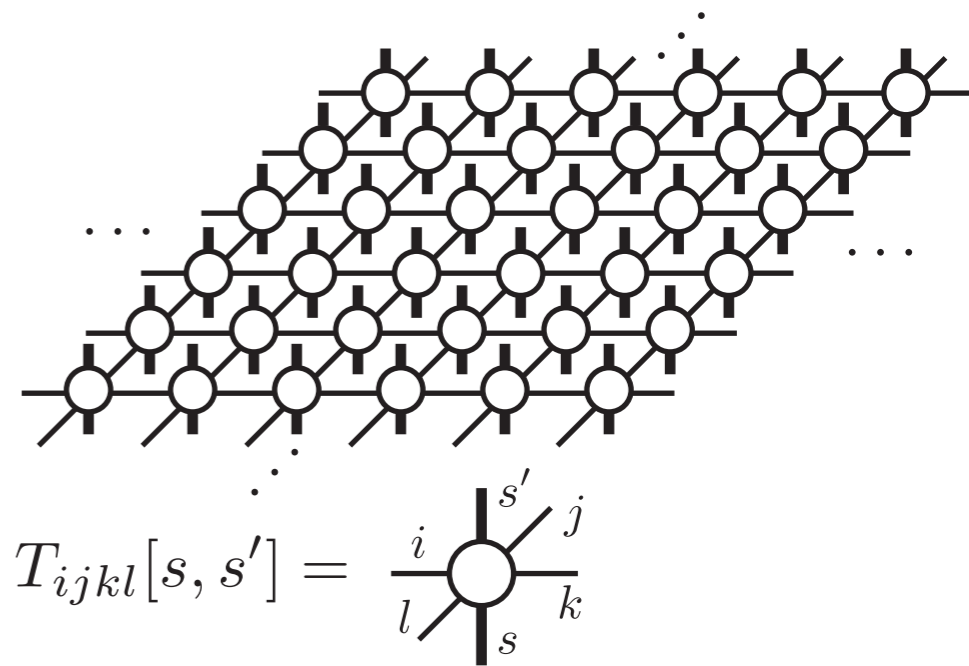
波動関数と同様に、密度行列も完全系で展開できて

$$\rho(\beta) = \sum_{\{\sigma_i\}, \{\sigma'_i\}} \rho_{\sigma_1 \sigma_2 \dots \sigma_N}^{\sigma'_1 \sigma'_2 \dots \sigma'_N} |\sigma'_1 \sigma'_2 \dots \sigma'_N\rangle \langle \sigma_1 \sigma_2 \dots \sigma_N|$$

係数テンソルをTNで近似表現できる

$$\rho^{\text{iTPO}} =$$

iTPO (infinite Tensor Product Operator)



有限温度計算

- Tr_ρ に出てくるトレースは、テンソルに2本生えているphysical bondをつなげる操作に相当する
- 各々のテンソルでトレースを先にとると、iTPSのノルム計算に出てくるテンソルネットワークと同じ形になる
- **iTPSと同様にCTMRGを用いた縮約ができる**
- iTPSと異なり、single layered構造なので、ボンド次元は D のまま、CTMのボンド次元 χ も $\chi \propto D$ とできる
- 計算量が $O(D^6)$ となるので、より大きな D の値まで計算可能



有限温度計算

- $\beta = 0$, すなわち高温極限での密度行列は単位行列 $\rho(0) = 1$ なので、任意の温度の密度行列は初期状態 $\rho(0)$ からの虚時間発展としてかける

$$\rho(\beta) = e^{-\beta\mathcal{H}} = e^{-\frac{\beta}{2}\mathcal{H}} e^{-\frac{\beta}{2}\mathcal{H}} = e^{-\frac{\beta}{2}\mathcal{H}} \rho(0) e^{-\frac{\beta}{2}\mathcal{H}}$$

- **虚時間発展はiTPOと同様**

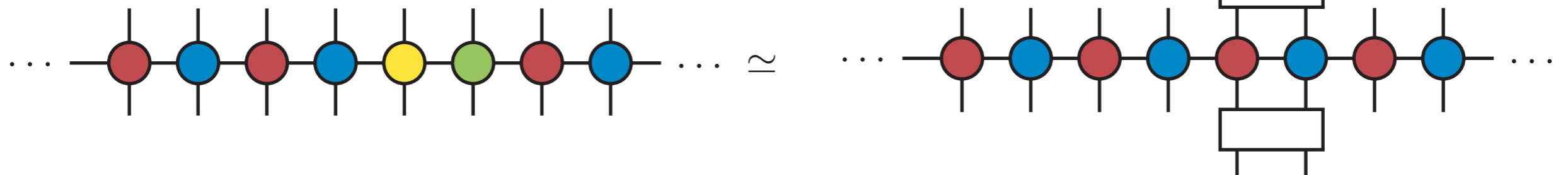
- 2本あるphysical bondに同時に虚時間発展させる
- 1本に束ねてしまえばiTPOの時間発展と同じ
- もちろんFull update, simple updateもそのまま

$$\rho_{\tau}^{\text{iTPO}}$$

Bond dimension = D

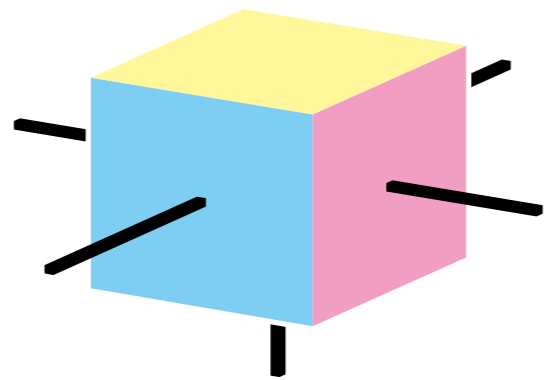
$$e^{-\frac{\tau}{2}\mathcal{H}_{ij}} \rho^{\text{iTPO}} e^{-\frac{\tau}{2}\mathcal{H}_{ij}}$$

Bond dimension = D



二次元量子格子模型

テンソルネットワークソルバー



T e N e S

Tensor Network Solver for Quantum Lattice Systems

TeNeS 開発チーム

- 大久保 毅 (東大院理)
- 森田 悟史 (慶應大)
- 本山 裕一 (東大物性研)
- 吉見 一慶 (東大物性研)
- 青山 龍美 (東大物性研)
- 加藤 岳生 (東大物性研)
- 川島 直輝 (東大物性研)

青字は高度化チーム

TeNeSは
物性研ソフトウェア開発・高度化プロジェクト
2019年度・2023年度課題
として支援を受けた

TeNeS の目標

- ・ 量子格子模型におけるテンソルネットワーク法研究への参入障壁を下げる
 - ・ TN 法のアルゴリズム・プログラム開発
 - ・ アルゴリズムは（ダイアグラム表記のおかげで）割とわかりやすい
 - ・ テンソル演算のライブラリはそれぞれの言語で多数存在
 - ・ 実際に1から組み上げるのは結構難しい（つらい）
 - ・ テンソルの添字の順番など、システムティックに構築しないとすぐに混乱する
 - ・ 複数のテンソルを縮約する場合、順番によって計算量オーダーが変わる
 - ・ 計算結果をどう評価するか？
 - ・ ボンド次元など、ハイパーパラメータが結構多い
 - ・ **TeNeS をベンチマークとして用いる！**
 - ・ 計算結果の比較
 - ・ ソースコードを読む・ TeNeS をベースに新アルゴリズムを入れる
 - ・ GNU GPL v3 であることに留意

TeNeS の目標

- ・ 量子格子模型におけるテンソルネットワーク法研究への参入障壁を下げる
 - ・ 別の分野の研究者
 - ・ 実験家
 - ・ 実験データのフィッティングなど
 - ・ TN 以外の手法開発者
 - ・ TN 法との比較のために使う
- ・ 使いやすさや汎用性を重視
 - ・ 特定の模型や格子に強く依存した最適化はしない
 - ・ 常に正方格子iTPS として扱う
- ・ スパコンなどの大規模並列計算機でも動く (分散メモリ並列によるテンソル演算)
 - ・ 大きなボンド次元でも計算できる
- ・ **「とりあえずTN 計算を試してみる」ができるようにする**

TeNeS の特徴

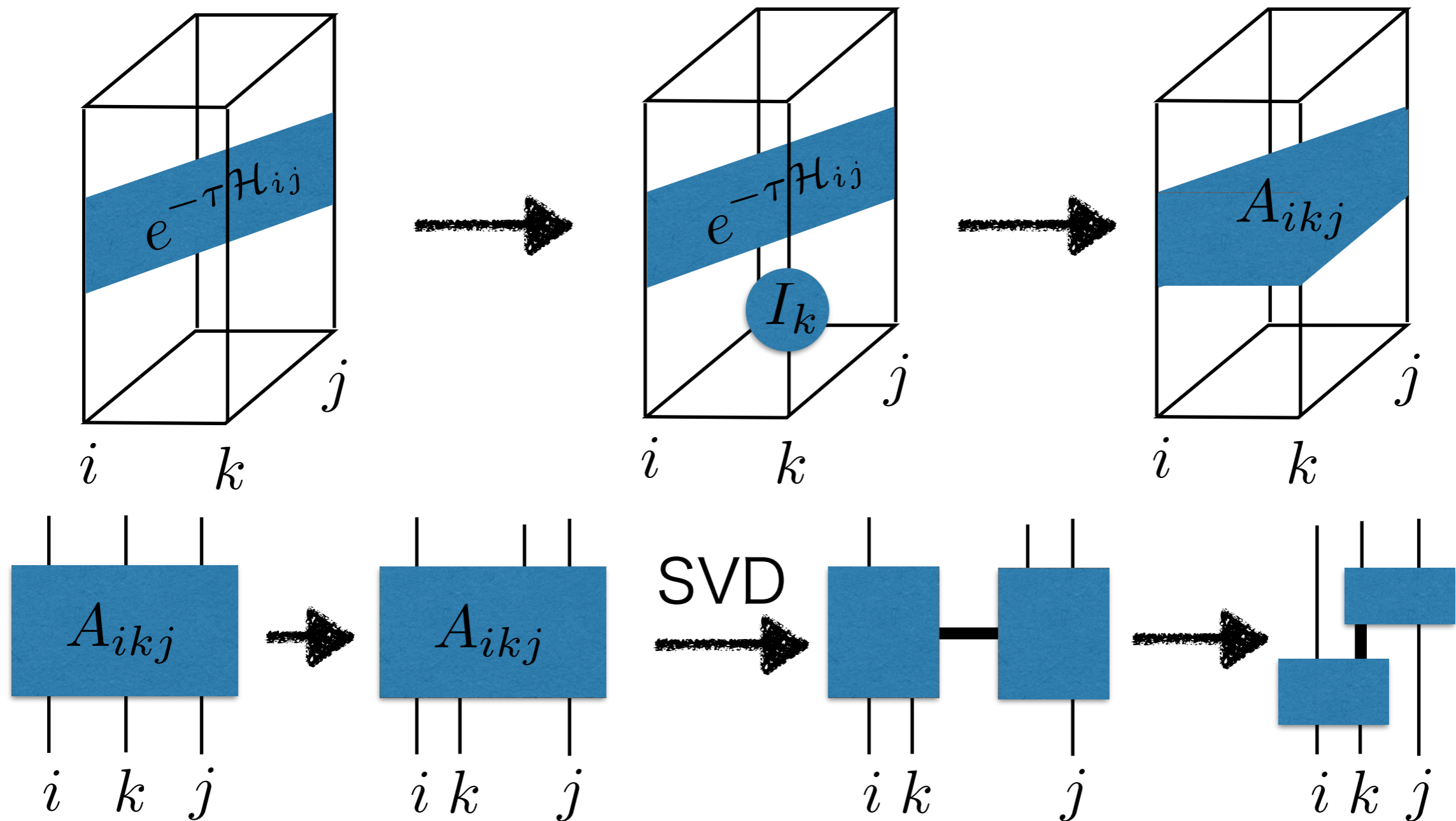
- ・ 二次元量子格子模型の計算を行うソフトウェア
- ・ 正方格子上のテンソルネットワーク (TN) で波動関数と密度行列を表現
- ・ 周期構造を仮定することで無限系を計算する (iTPS/iTPO)
 - ・ 無限系の縮約には **CTMRG** または **平均場環境** を利用する
- ・ **基底状態計算** $|\Psi\rangle \propto \lim_{\beta \rightarrow \infty} e^{-\beta \mathcal{H}} |\phi\rangle = \lim_{n \rightarrow \infty} (e^{-\tau \mathcal{H}})^n |\phi\rangle$
 - ・ 波動関数に虚時間発展演算子をかけていくことで基底状態を得る
 - ・ 最後に物理量測定
- ・ **実時間発展** $|\Psi(T)\rangle \propto e^{-iT\mathcal{H}} |\Psi(0)\rangle = \left(e^{-i(T/n)\mathcal{H}} \right)^n |\Psi(0)\rangle$
 - ・ 波動関数に実時間発展演算子をかけながら物理量を測定
- ・ **有限温度計算** $\rho(\beta) = e^{-(\beta/2)\mathcal{H}} \mathbb{1} e^{-(\beta/2)\mathcal{H}} = \left[e^{-(\tau/2)\mathcal{H}} \right]^n \mathbb{1} \left[e^{-(\beta/2)\mathcal{H}} \right]^n$
 - ・ 密度行列に虚時間発展演算子をかけていくことで有限温度密度行列を得る
- ・ 時間発展演算子をかける操作には **simple update (SU)** と **full update (FU)** を利用

TeNeS の特徴

- ・ ハミルトニアン
 - ・ 正方格子上の任意の短距離2サイト相互作用を計算可能
 - ・ 現状はx, y 方向にそれぞれ3サイト先まで
 - ・ 正方格子の4x4 に収まる範囲内
 - ・ 相互作用の形を工夫することで**正方格子以外も計算可能**
 - ・ 例：正方格子の斜め方向に入れて三角格子
 - ・ フェルミオン系は対象外（スピン系・ボソン系のみ）
- ・ 物理量測定
 - ・ 正方格子4x4 に収まる範囲内の演算子
 - ・ x, y 軸方向に平行な向き of 2点相関関数
 - ・ 相関長
- ・ テンソル演算には **mptensor** (<https://github.com/smorita/mptensor>) を利用
 - ・ MPI 実行するだけで ScaLAPACK を利用した分散メモリ並列がなされる

TeNeS の特徴

- TeNeS のITE は簡単のために正方格子最近接ボンドのみ扱う
- 長距離ITE テンソルは正方格子最近接 ITE テンソルの積に変換しておく



TeNeS の特徴

- よく使われそうな模型・格子は定義済み
- 模型や格子のパラメータを与えるだけで良い
- 定義済み模型
 - 量子スピン模型（スピンの大きさ S は任意）

$$\mathcal{H} = \sum_{i < j} \left[\left(\sum_{\alpha}^{x,y,z} J_{ij}^{\alpha} S_i^{\alpha} S_j^{\alpha} \right) + B_{ij} \left(\vec{S}_i \cdot \vec{S}_j \right)^2 \right] - \sum_i \left[\sum_{\alpha}^{x,y,z} h^{\alpha} S_i^{\alpha} + D (S_i^z)^2 \right]$$

- Bose-Hubbard 模型（サイトの粒子数カットオフ付き）

$$\mathcal{H} = \sum_{i < j} \left[-t_{ij} \left(b_i^{\dagger} b_j + \text{h.c.} \right) + V_{ij} n_i n_j \right] + \sum_i \left[U \frac{n_i (n_i - 1)}{2} - \mu n_i \right]$$

- もちろん自分で模型（局所ハミルトニアン）を定義することも可能

TeNeS の特徴

- ・ よく使われそうな模型・格子は定義済み
 - ・ 模型や格子のパラメータを与えるだけで良い
 - ・ 定義済み格子
 - ・ 格子回転異方性や最近接・2次近接・3次近接相互作用を導入可能
 - ・ 正方格子
 - ・ 三角格子
 - ・ 蜂の巣格子
 - ・ カゴメ格子
- ・ もちろん自分で格子を定義することも可能

TeNeS の入力ファイル

- 定義済みの模型・格子を使う場合には、簡単なテキストファイルひとつで計算可能

- 右は正方格子上の横磁場イジング模型の基底状態計算に必要な入力ファイル

- 2x2 ユニットセル

- SU 1000 steps

- $D=2, \chi = 10$

```
[parameter]
[parameter.general]
is_real = true # Limit tensor elements in real (not complex)

[parameter.simple_update]
num_step = 1000 # Number of steps in simple update
tau = 0.01 # Imaginary time slice

[parameter.full_update]
num_step = 0 # Number of steps in full update
tau = 0.01 # Imaginary time slice

[parameter.ctm]
meanfield_env = false # Use meanfield environment to contract iTNS
iteration_max = 10 # Maximum number of iterations in CTMRG
dimension = 10 # Bond dimension of corner transfer matrix

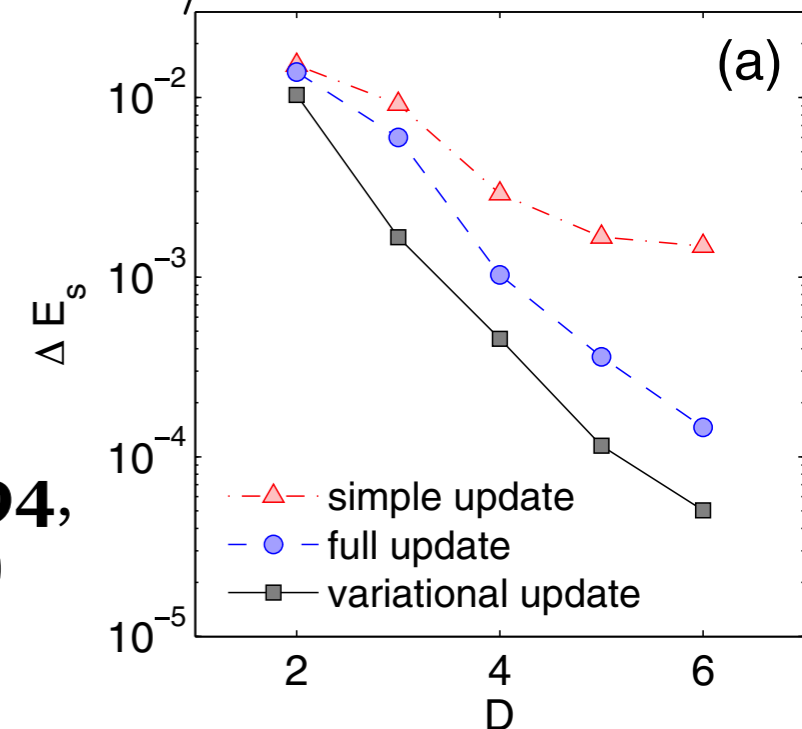
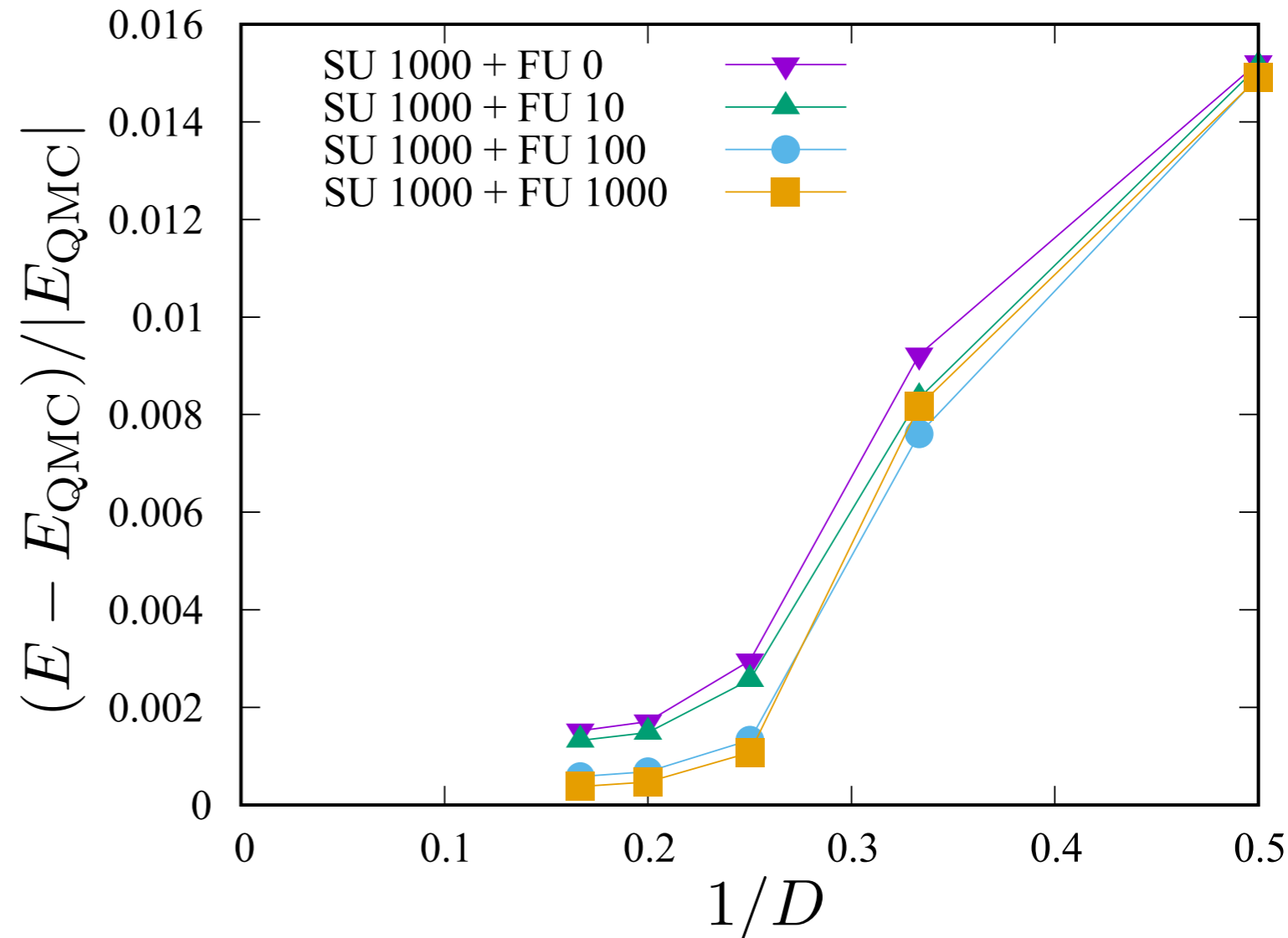
[lattice]
type = "square lattice" # Type of lattice
L = 2 # X length of unit cell
W = 2 # Y length of unit cell
virtual_dim = 2 # Bond dimension of bulk tensors
initial = "ferro" # Initial condition

[model]
type = "spin" # Type of model
Jz = -1.0 # Jz SzSz
Jx = 0.0 # Jx SxSx
Jy = 0.0 # Jy SySy
hx = 0.0 # hx Sx
```

例：正方格子S=1/2反強磁性ハイゼンベルグ模型

$$\mathcal{H} = J \sum_{\langle ij \rangle} S_i \cdot S_j$$

- ・ 正方格子S=1/2反強磁性ハイゼンベルグ模型の基底エネルギー
- ・ 横軸はボンド次元の逆数
- ・ 縦軸はQMC による見積もりからの相対誤差
- ・ A.W. Sandvik, AIP Conf. Proc. **1297**, 135 (2010)
- ・ ユニットセルの大きさは2×2
- ・ $\chi = D^2$
- ・ 24CPU の計算機で数時間程度

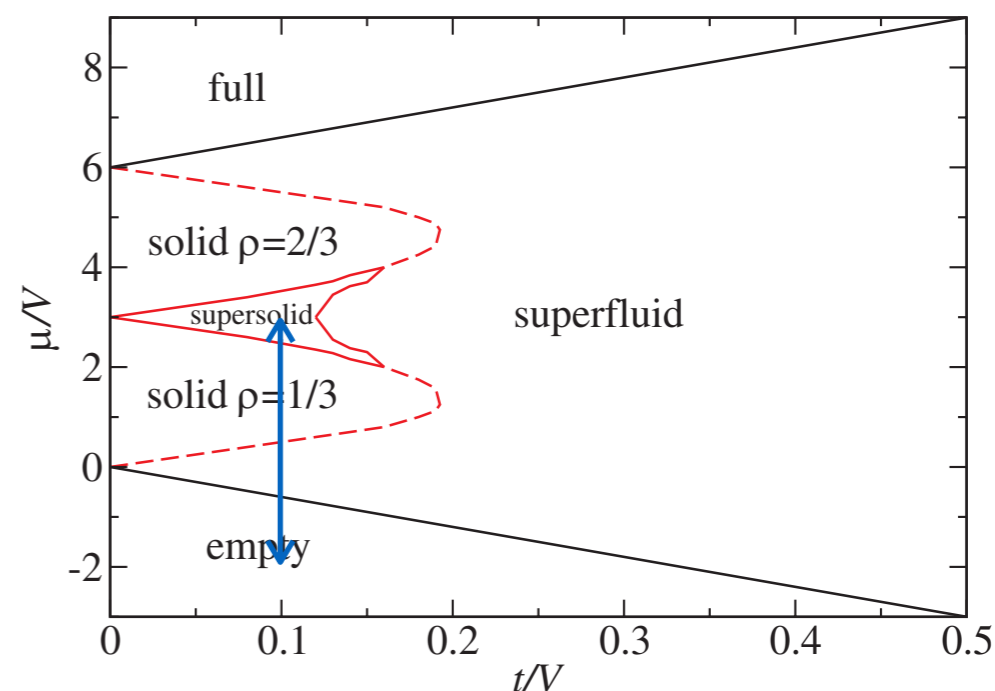
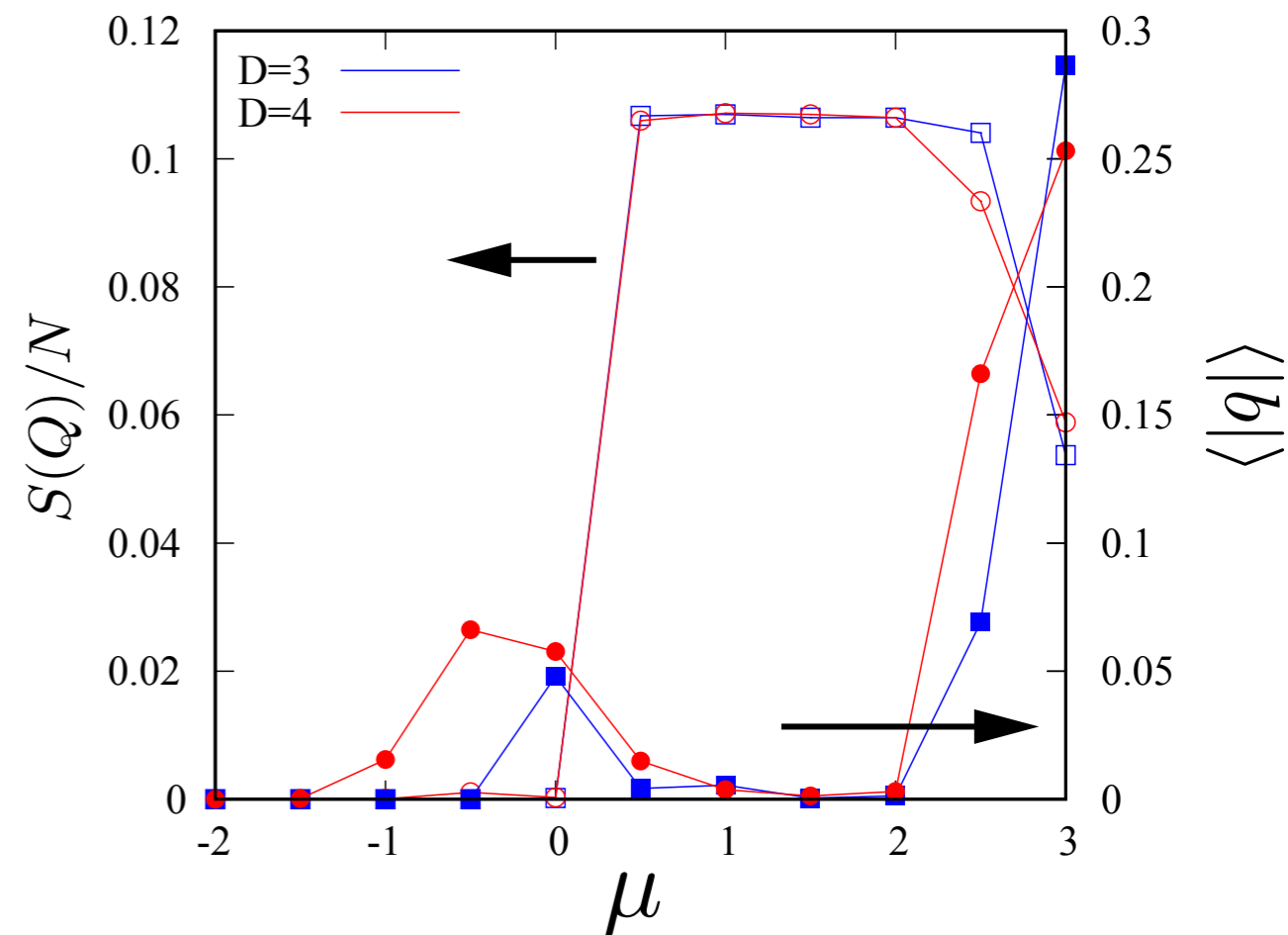


P. Corboz, PRB **94**, 035133 (2016)

例：三角格子ハードコアボースハバード模型

$$\mathcal{H} = -t \sum_{\langle ij \rangle} [b_i^\dagger b_j + b_j^\dagger b_i + V n_i n_j] - \mu \sum_i n_i$$

- 1/3 相と 2/3 相の間に超固体相がある
- 超流動性と固体性が同時に存在
- $t=0.1, V=1.0$
- $D=3$ および $D=4$
- $\chi = D^2$
- ユニットセルは 3×3
- SU 2000 step, FU なし
- 構造因子 $S(Q)$ (左軸) と超流動オーダー $|b|$ (右軸)
- Q は $\sqrt{3} \times \sqrt{3}$ オーダーの波数
- 先行研究(QMC) と矛盾しない結果

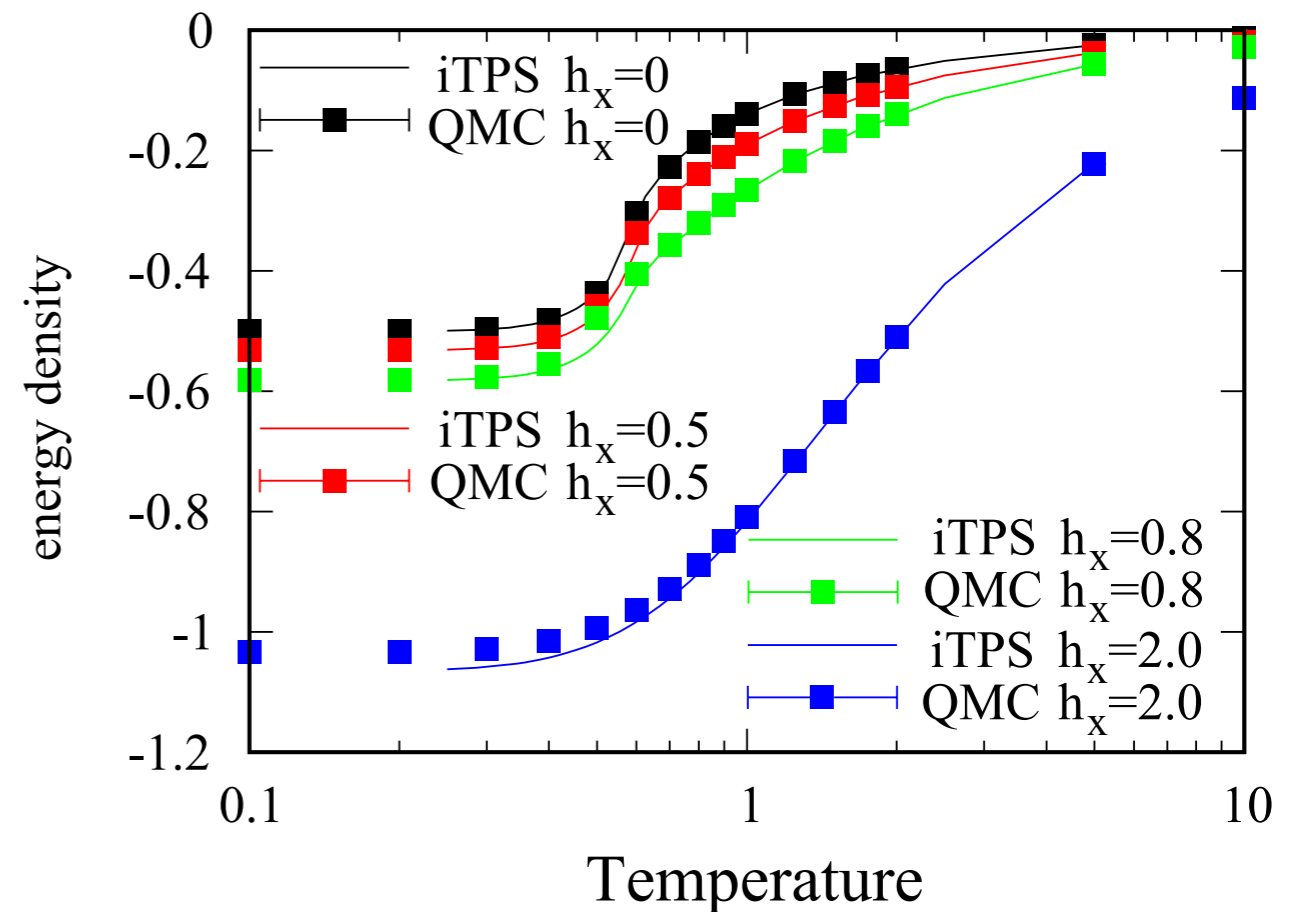


Wessel and Troyer, PRL **95**, 127205 (2005)

例：横磁場イジングモデルの有限温度計算

$$\mathcal{H} = J_z \sum_{\langle ij \rangle} S_i^z S_j^z - h_x \sum_i S_i^x$$

- $J_z = -1$ (強磁性)
- $h_x = 0, 0.5, 0.8, 2.0$
- $T=0$ での量子相転移点は $h_x \sim 1.5$
- $D=3, \chi=10$
- ユニットセルは 2×2
- Simple Updateによる虚時間発展
- 経路積分MCと矛盾しない結果
 - 同じく高度化ソフトウェアであるDSQSSを利用している



まとめ (1/2)

- ・ 2次元量子格子模型のテンソルネットワークソルバー**TeNeS**の紹介をした
- ・ TeNeSに実装されている計算手法を紹介した
 - ・ iTPS(iTPO)は量子多体系の基底状態（密度行列）を効率的にかつ精度よく表現できる
 - ・ 近似精度をボンド次元Dで調整可能
 - ・ 並進対称性を利用すると無限系の状態を有限個のテンソルで表現できる
 - ・ CTMRGを用いることで無限系の環境も効率よく近似可能
 - ・ 虚時間発展法でiTPSの最適化を行える
 - ・ 実時間発展やiTPOの計算にも応用可能
- ・ TeNeSは**幅広い人が・幅広い問題に対して・簡単にTN計算**ができることを目指す
 - ・ 分散メモリ並列による大規模並列計算を実装している
 - ・ **分散メモリ並列で行列演算ができる自動微分ライブラリがないのが困りごと**（今後の課題）
 - ・ 扱える格子・模型の範囲を広く取っているので、特定の問題に特化したアルゴリズム・チューニングは採用されていない
 - ・ 正方格子以外、特に三角格子は精度があまりでない（今後の課題）

まとめ (2/2)

- TeNeSについて
 - website: <https://www.pasums.issp.u-tokyo.ac.jp/tenes>
 - GitHub: <https://github.com/issp-center-dev/TeNeS>
 - ソフトウェア論文
 - YM and et al., Comput. Phys. Commun. **279**, 108437 (2022)
 - YM and et al., Comput. Phys. Commun. **315**, 109692 (2025)
- PASUMS (物性研ソフトウェア開発・高度化プロジェクト) について
 - 国内の計算物性物理研究者のソフトウェア開発を支援するプロジェクト
 - <https://www.pasums.issp.u-tokyo.ac.jp>
 - プロジェクト紹介論文
 - K. Yoshimi, YM, T. Aoyama, M. Kawamura, & N. Kawashima, Science and Technology of Advanced Materials: Methods **5**(1), 2564055 (2025).
 - 最近LLMを用いたソフトウェア開発についても調査・活用しています
 - それぞれの分野におけるソフトウェア開発のありかたというのも議論できたら是非