第2回量子ソフトウェア産学協働ゼミ

テンソルネットワークを用いた量子回路シミュレーション

2022.9.27

理学系研究科 量子ソフトウェア寄付講座

秋山 進一郎

今日の講義の内容

1) 量子回路の復習

- 典型的な量子ゲート
- 今日の演習で扱う量子アルゴリズム

2) テンソルネットワークシミュレーション

- 量子回路のTN表示
- 確率振幅，縮約密度行列の計算
- 計算コスト
- 縮約計算の順序の工夫
- 量子ゲートの行列積分解

今日の講義の内容

1) 量子回路の復習

- 典型的な量子ゲート
- 今日の演習で扱う量子アルゴリズム

2) テンソルネットワークシミュレーション

- 量子回路のTN表示
- 確率振幅，縮約密度行列の計算
- 計算コスト
- 縮約計算の順序の工夫
- 量子ゲートの行列積分解

量子回路とは？

量子ビットに演算する量子ゲートの回路図

情報理論　量子ゲートによる量子計算モデル

物理学　**時間発展する量子ビットの多体系**

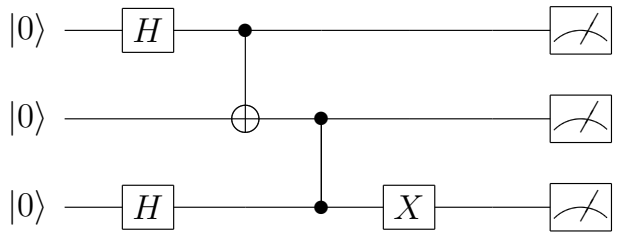
量子回路における計算の流れ

1) 量子ビットの初期状態を準備

2) 順番に量子ゲート（ユニタリ行列）を演算

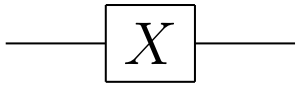
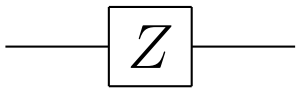
3) 測定を行なって計算結果を得る

Ex. 量子回路の例



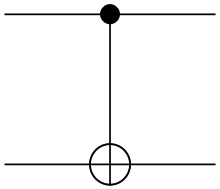
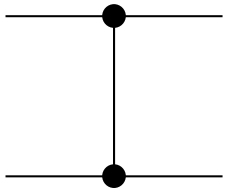
典型的な1量子ビットゲート

計算基底のベクトル表示　$|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}, \quad |1\rangle = \begin{bmatrix}0\\1\end{bmatrix}$

	回路上の表記	行列表示	量子ビットへの作用
Xゲート		$X = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$	$X\|0\rangle = \|1\rangle$ $X\|1\rangle = \|0\rangle$
Zゲート		$Z = \begin{bmatrix}1 & 0\\0 & -1\end{bmatrix}$	$Z\|0\rangle = \|0\rangle$ $Z\|1\rangle = -\|1\rangle$
Hゲート		$H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$	$H\|0\rangle = \frac{\|0\rangle + \|1\rangle}{\sqrt{2}}$ $H\|1\rangle = \frac{\|0\rangle - \|1\rangle}{\sqrt{2}}$

典型的な2量子ビットゲート

計算基底のベクトル表示 $|00\rangle = \begin{bmatrix}1\\0\\0\\0\end{bmatrix}$, $|01\rangle = \begin{bmatrix}0\\1\\0\\0\end{bmatrix}$, $|10\rangle = \begin{bmatrix}0\\0\\1\\0\end{bmatrix}$, $|11\rangle = \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$

	回路上の表記	行列表示	量子ビットへの作用
CXゲート		$CX = \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}$	$CX\|00\rangle = \|00\rangle$ $CX\|01\rangle = \|01\rangle$ $CX\|10\rangle = \|11\rangle$ $CX\|11\rangle = \|10\rangle$
CZゲート		$CZ = \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&-1\end{bmatrix}$	$CZ\|00\rangle = \|00\rangle$ $CZ\|01\rangle = \|01\rangle$ $CZ\|10\rangle = \|10\rangle$ $CZ\|11\rangle = -\|11\rangle$

量子回路に基づく量子アルゴリズムのポイント

- 多数の基底の重ね合わせ状態を初期状態として用意
  → Hゲートなどを用いる

- 量子並列性の活用
  → 量子回路に重ね合わせ状態を入力
  終状態は対応する出力の重ね合わせ

- 目的に合わせて量子ゲートを順次演算して量子状態を操作する
  → 欲しい答えを高確率で実現するようにする

今日の演習で扱う量子アルゴリズム 1/3

1) Grover アルゴリズム　Grover, STOC '96(1996)212-219

非構造化データ(規則性がなく検索しづらいデータ)を探索する

N個のアイテムリスト中，1個の当たりデータ(☆)を探す

1	2	3	…	☆	…	N

古典計算　$O(N)$のステップ数で☆に辿りつける

量子計算　**Groverの方法により，$O(\sqrt{N})$のステップ数で☆に辿りつける**

今日の演習で扱う量子アルゴリズム 2/3

2) Quantum bit string comparator

Oliveira-Ramos, Quantum Computers and Computing (2007) 7

2つの量子状態を比較し，それらが同一か異なっているかを特定する

n個の量子ビットからなる2つの状態 $|a\rangle$ と $|b\rangle$ に対し，$m+2$個の補助量子ビットを用意．次のようにユニタリ発展させる：

$$U|a\rangle|b\rangle|0^{\otimes m}\rangle|0\rangle|0\rangle = |a\rangle|b\rangle|\psi\rangle|xy\rangle$$

このとき，以下を実現するようなユニタリ変換Uを構築できる

$$a = b \Rightarrow x = y = 0$$

$$a > b \Rightarrow x = 1, y = 0$$

$$a < b \Rightarrow x = 0, y = 1$$

→ 2つのビット長 $a(= a_1 a_2 \cdots a_n)$ と $b = (b_1 b_2 \cdots b_n)$ の大小関係を判定

今日の演習で扱う量子アルゴリズム 3/3

2) Quantum bit string comparator

Oliveira-Ramos, Quantum Computers and Computing (2007) 7

2つの量子状態を比較し，それらが同一か異なっているかを特定する

n個の量子ビットからなる2つの状態 $|a\rangle$ と $|b\rangle$ に対し，$m+2$個の補助量子ビットを用意．次のようにユニタリ発展させる：

$$U|a\rangle|b\rangle|0^{\otimes m}\rangle|0\rangle|0\rangle = |a\rangle|b\rangle|\psi\rangle|xy\rangle$$

Ex. $|a\rangle = |11\rangle$, $|b\rangle = \alpha|01\rangle + \beta|11\rangle$ の場合

確率 $|\alpha|^2$ で $a > b$ $\Rightarrow$ $x = 1,\ y = 0$

確率 $|\beta|^2$ で $a = b$ $\Rightarrow$ $x = y = 0$

$\rightarrow |xy\rangle = \alpha|10\rangle + \beta|00\rangle$

量子アルゴリズムの展望：実社会への応用

現在の量子コンピュータ（NISQ）で実社会の問題を高速に解ける量子アルゴリズムは確立されていない

現状では？ → Ex. 量子機械学習

Neural Netを変分量子回路（パラメタ付量子回路）で置き換える

ライブラリの開発も進んでいる

Cf. TensorFlow Quantum , PENNY LANE etc.

将来的には？ → Ex. 交通系，金融系における最適化問題への応用

ベストな古典アルゴリズムとの比較が必要

→ 量子アルゴリズムの優位性の検証

Cf. ランダム量子回路の古典サンプリング Liu+, SC '21(2021)12

量子計算で200秒 by Google
古典計算で**304秒**（当初の見積もりだと1万年）

今日の講義の内容

1) 量子回路の復習

- 典型的な量子ゲート
- 今日の演習で扱う量子アルゴリズム

2) テンソルネットワークシミュレーション

- 量子回路のTN表示
- 確率振幅，縮約密度行列の計算
- 計算コスト
- 縮約計算の順序の工夫
- 量子ゲートの行列積分解

今日の演習で扱う量子回路のシミュレーション手法

Schrödingerシミュレーション

n量子ビットからなる状態を2^nサイズのベクトルとして表現

n量子ビットゲートを$2^n \times 2^n$サイズの行列として表現

Straightforwardなシミュレーション手法

量子ビット数nに関して指数関数的なコストがかかる

テンソルネットワークシミュレーション

量子回路をテンソルネットワークで表現

縮約計算の順序の工夫，行列積分解の活用

Cf. Feynmanシミュレーション

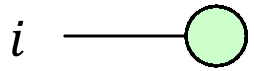
2021年度のパイロット講義（第2回）を参照

テンソルネットワークのダイヤグラム記法 1/2

スカラー → 0脚テンソル

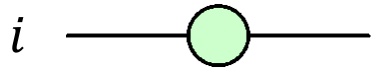
c

ベクトル → 1脚テンソル



v_i

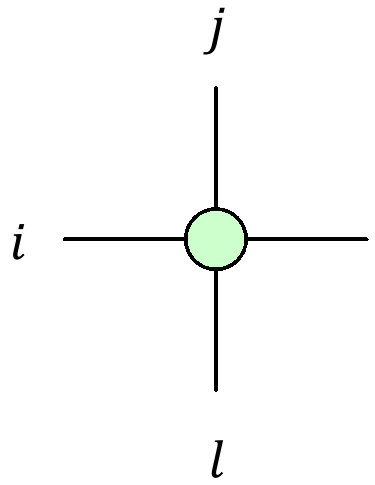
行列 → 2脚テンソル



A_{ij}

r階テンソル → r脚テンソル

Ex. 4階テンソル → 4脚テンソル

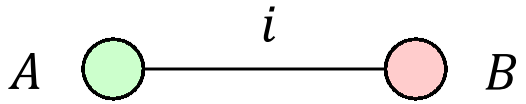


T_{ijkl}

テンソルネットワークのダイヤグラム記法 2/2

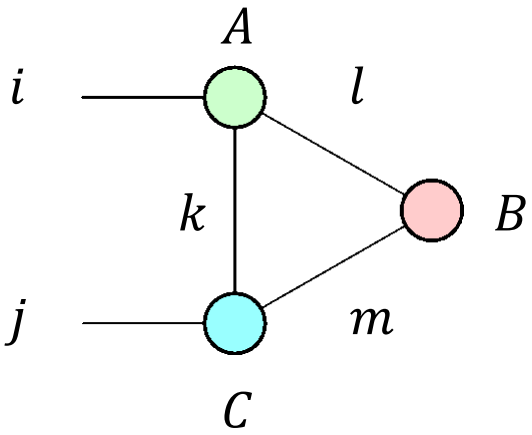
内線は縮約される脚，外線は縮約されない脚

Ex. 1



$\sum_i A_i B_i$

Ex. 2



$\sum_{klm} A_{ikl} B_{lm} C_{jkm}$

n量子ビットからなる量子回路のTN表示

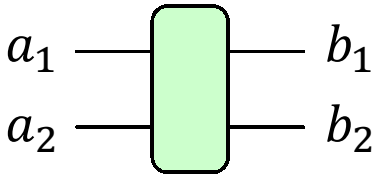
初期状態のTN表示

直積状態 $|0\rangle \otimes |0\rangle \otimes \cdots \otimes |0\rangle \rightarrow n$個の1脚テンソル

量子ゲートのTN表示

1量子ビットゲート（2×2行列） → 2脚テンソル

2量子ビットゲート（$2^2 \times 2^2$行列） → 4脚テンソル



r量子ビットゲート（$2^r \times 2^r$行列） → $2r$脚テンソル

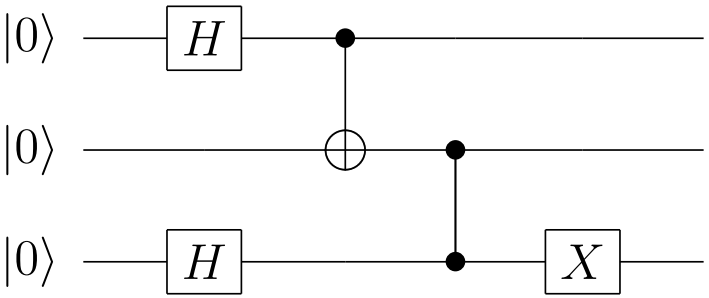
ボンド次元（それぞれの脚のサイズ）は2

r量子ビットゲートのTN表示は2^{2r}個の成分を記述する

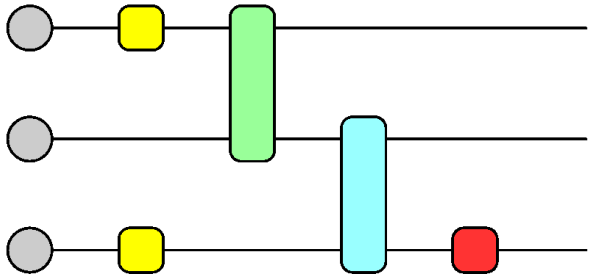
テンソルネットワークシミュレーションの概要

量子回路をTN表示して縮約を実行する

量子回路



TN表示

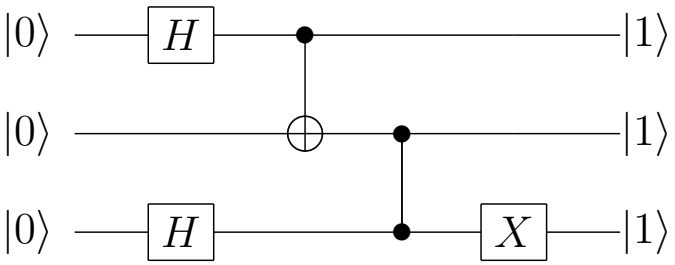


時間発展と同じ順番で縮約すればSchrödingerシミュレーションと等価

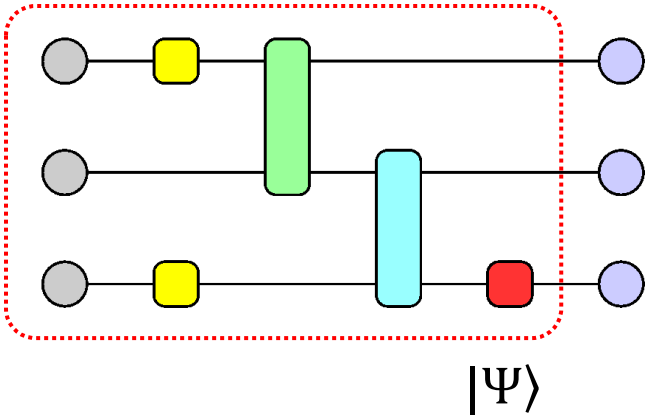
テンソルネットワークによる確率振幅の計算

求めたい確率振幅に対応するTN表示を作って縮約を実行する

量子回路（振幅⟨111|Ψ⟩）



振幅⟨111|Ψ⟩ のTN表示



TNシミュレーションのご利益は？

→ **縮約をとる順序の工夫，行列積分解の活用で計算コストを下げ得る**

テンソルネットワークによる縮約密度行列の計算 1/2

特定の量子ビットに着目し，そこで実現する状態の確率を求めたい

例えば，次で定義されるような周辺分布 $P(s_1)$ を求めたい

$$P(s_1, s_2, \cdots, s_n) = |\langle s_1 s_2 \cdots s_n|\Psi\rangle|^2 / \langle\Psi|\Psi\rangle$$

$$P(s_1) = \sum_{s_2, \cdots, s_n} P(s_1, s_2, \cdots, s_n)$$

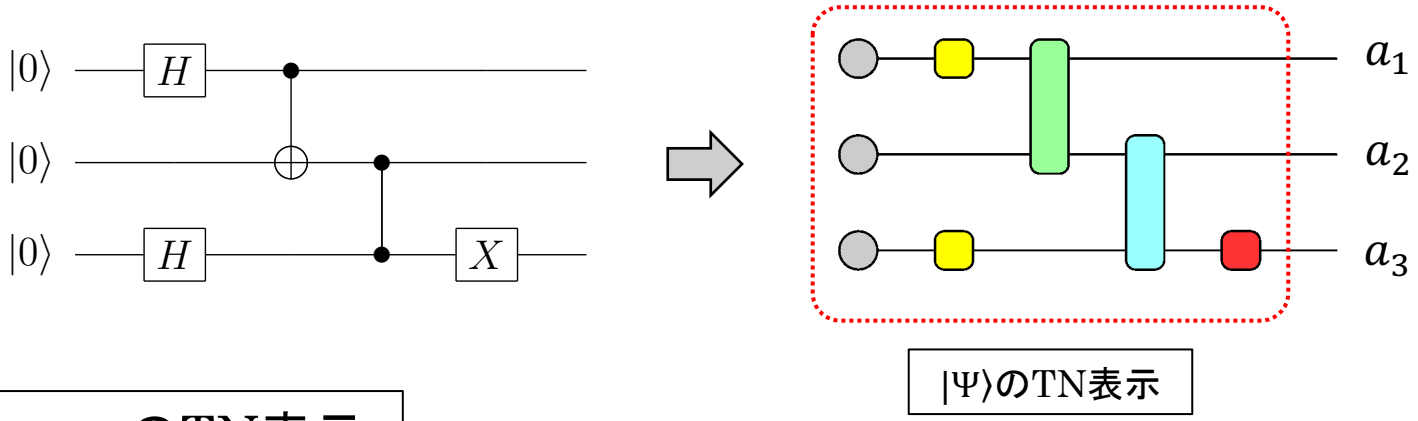
$P(s_1)$を求めるには，以下の縮約密度行列 $\rho_{s_1 s_1'}$ の対角成分を見ればよい

$$\rho_{s_1 s_1'} = \sum_{s_2, \cdots, s_n} \langle s_1 s_2 \cdots s_n|\Psi\rangle\langle\Psi|s_1' s_2 \cdots s_n\rangle / \langle\Psi|\Psi\rangle$$

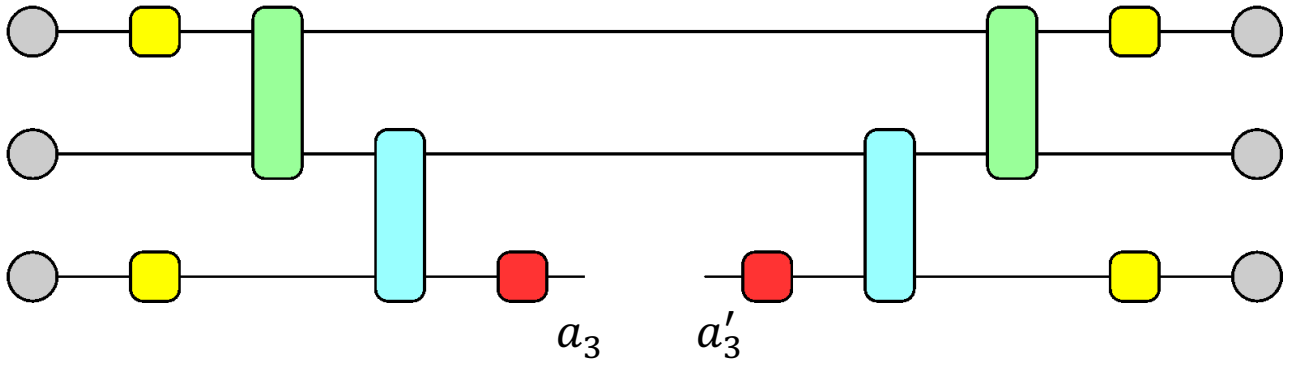
$|\Psi\rangle$のTN表示から $\rho_{s_1 s_1'}$ のTN表示を作ることができる

テンソルネットワークによる縮約密度行列の計算 2/2

Ex. 下図で a_3 に着目する場合 i.e. $P(a_3)$ を求めたい場合



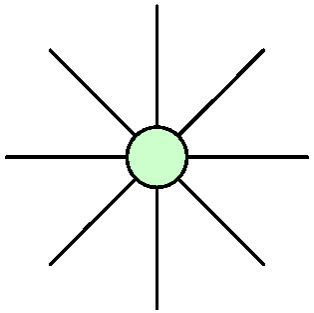
$\rho_{a_3 a_3'}$のTN表示



テンソルの保持にかかるメモリコスト

ボンド次元χのr脚テンソルを保持するために必要なメモリコストは$O(\chi^r)$

Ex. ボンド次元χの8脚テンソルのメモリコスト



$T_{ijklmnop} \in \mathbb{C}$

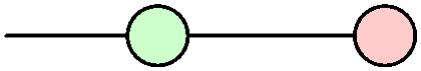
ボンド次元	メモリ
$\chi = 2$	4KB
$\chi = 5$	6MB
$\chi = 10$	1.6GB
$\chi = 15$	41GB
$\chi = 20$	410GB

※倍精度複素数の場合

縮約にかかる計算コスト

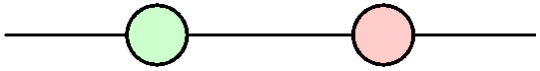
素朴には内線と外線の本数を数えることで計算コストが見積もれる

行列・ベクトル積



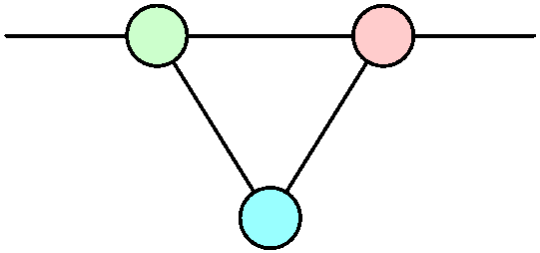
$O(\chi^2)$

行列・行列積



$O(\chi^3)$

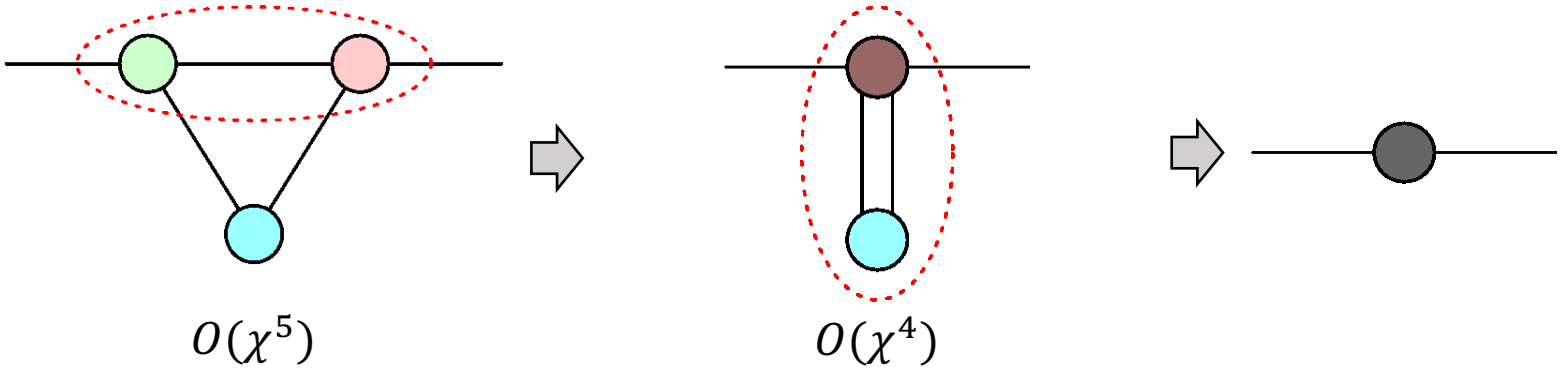
次のような縮約の計算コストはどうか？



縮約計算の順序を工夫する 1/2

縮約の計算コスト＆メモリコストは縮約の順序（contraction path）に依存

Path 1



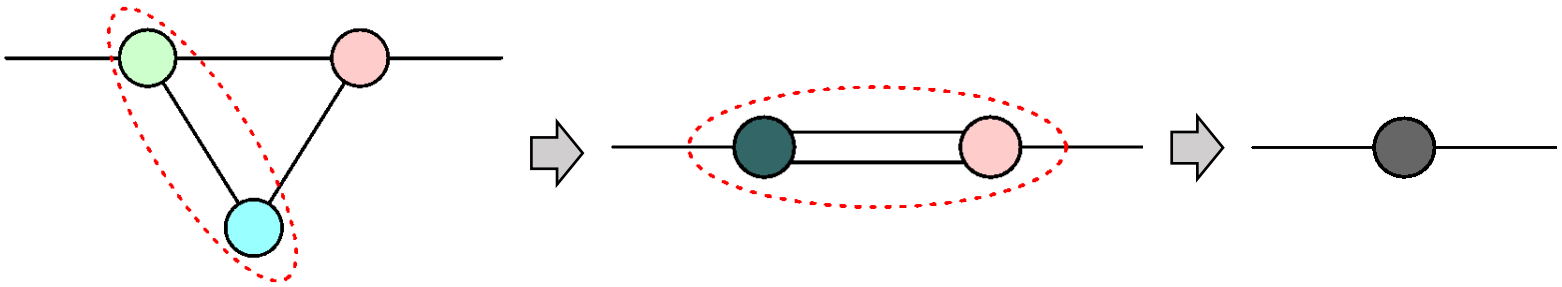
$O(\chi^5)$ $O(\chi^4)$

Path 1の場合，縮約の計算コストは$O(\chi^5)$，メモリコストは$O(\chi^4)$

縮約計算の順序を工夫する 2/2

縮約の計算コスト＆メモリコストは縮約の順序（contraction path）に依存

Path 2



$O(\chi^4)$ $O(\chi^4)$

Path 2の場合，縮約の計算コストは$O(\chi^4)$，メモリコストは$O(\chi^3)$

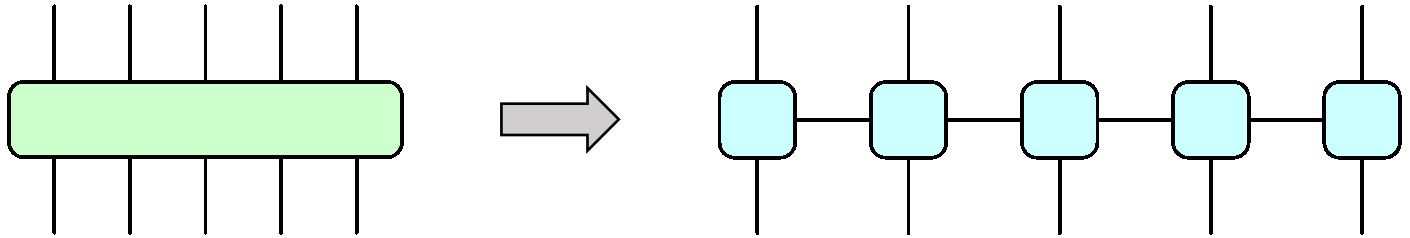
Path 1の場合，縮約の計算コストは$O(\chi^5)$，メモリコストは$O(\chi^4)$

行列積分解を活用する

多脚テンソルをより階数の小さなテンソルの縮約形で表現する

場合によっては計算コスト，メモリコストを大幅に削減可能

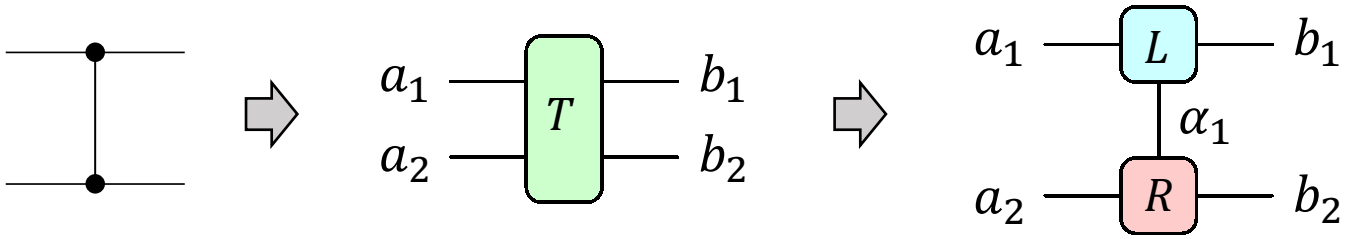
Cf. Matrix Product State (MPS), Tensor Train Decomposition



※行列積分解のより詳しい説明は第1回産学協働ゼミの講義資料を参照

Ex. CZゲート（2量子ビット）の行列積分解

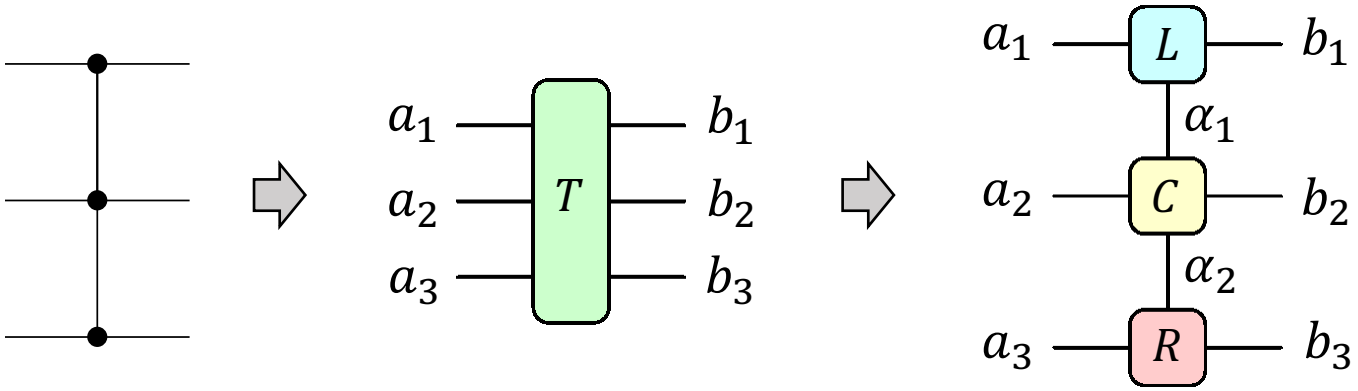
CZゲートは$\chi = 2$の行列積演算子に**厳密に**書き換え可能



$$T^{a_1 a_2 b_1 b_2} = \sum_{\alpha_1=0,1} L_{\alpha_1}^{a_1 b_1} R_{\alpha_1}^{a_2 b_2}$$

Ex. CZゲート（3量子ビット）の行列積分解

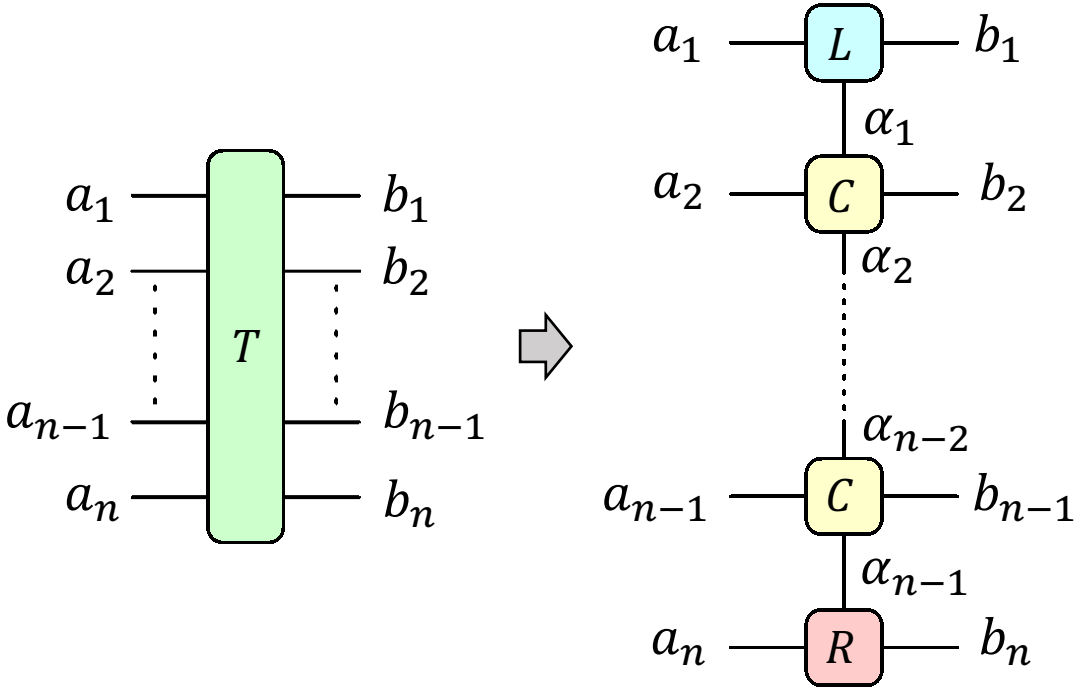
CZゲートは$\chi = 2$の行列積演算子に**厳密に**書き換え可能



$$T^{a_1 a_2 a_3 b_1 b_2 b_3} = \sum_{\alpha_1=0,1} \sum_{\alpha_2=0,1} L^{a_1 b_1}_{\alpha_1} C^{a_2 b_2}_{\alpha_1 \alpha_2} R^{a_3 b_3}_{\alpha_2}$$

Ex. CZゲート（n量子ビット）の行列積分解

CZゲートは$\chi = 2$の行列積演算子に**厳密に**書き換え可能



$$T^{a_1 a_2 \cdots a_{n-1} a_n b_1 b_2 \cdots b_{n-1} b_n} =$$

$$\sum_{\alpha_1=0,1} \sum_{\alpha_2=0,1} \cdots \sum_{\alpha_{n-2}=0,1} \sum_{\alpha_{n-1}=0,1} L^{a_1 b_1}_{\alpha_1} C^{a_2 b_2}_{\alpha_1 \alpha_2} \cdots C^{a_{n-1} b_{n-1}}_{\alpha_{n-2} \alpha_{n-1}} R^{a_n b_n}_{\alpha_{n-1}}$$

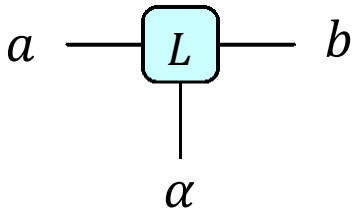
保持すべき成分数が2^{2n}個から$(n-1)2^4$個に削減

行列積分解の具体的な表示とその解釈 1/3

L_α^{ab}の成分表示

$$L^{00} = \begin{bmatrix}1\\0\end{bmatrix}, \quad L^{11} = \begin{bmatrix}0\\1\end{bmatrix}$$

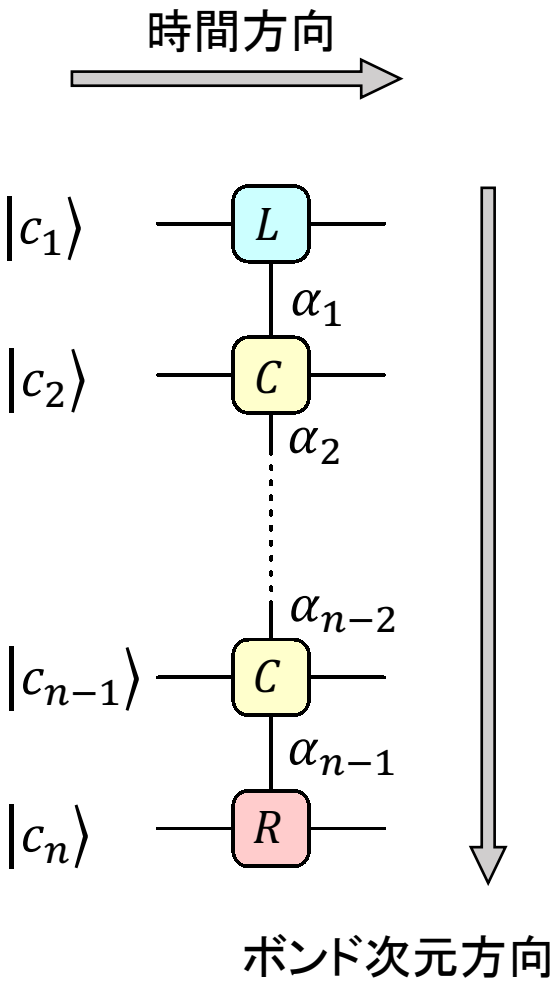
$$L^{01} = L^{10} = \begin{bmatrix}0\\0\end{bmatrix}$$



L_α^{ab}は仮想的な量子ビットとみなせる

物理的な量子ビット$|c_1\rangle$が0ならば$|0\rangle$，1ならば$|1\rangle$

$$L^{00} = \begin{bmatrix}1\\0\end{bmatrix} = |0\rangle, \quad L^{11} = \begin{bmatrix}0\\1\end{bmatrix} = |1\rangle$$

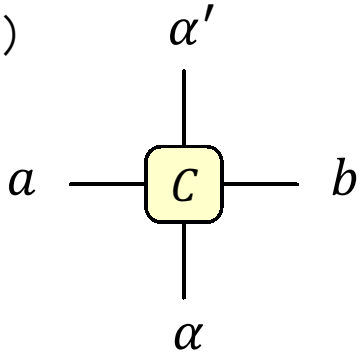


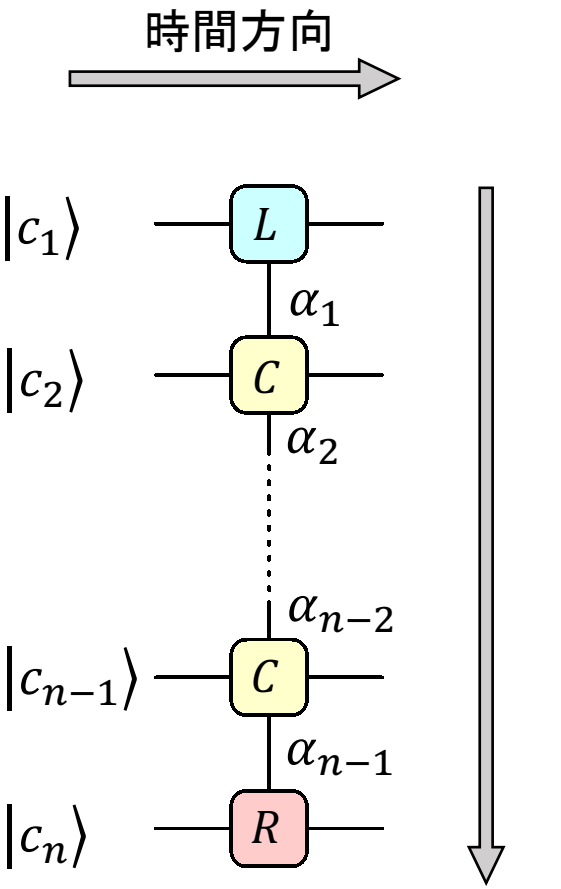
行列積分解の具体的な表示とその解釈 2/3

$C^{ab}_{\alpha\alpha'}$の成分表示 （行がα，列がα'）

$$C^{00} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, \quad C^{11} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$C^{01} = C^{10} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$





$C^{ab}_{\alpha\alpha'}$はボンド次元方向の転送行列

仮想的な量子ビット$L^{ab}_{\alpha'}$が初期状態

$$C^{00}|0\rangle = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle$$

$$C^{00}|1\rangle = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = |0\rangle$$

C^{00}が作用すると$|0\rangle$

$$C^{11}|0\rangle = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle$$

$$C^{11}|1\rangle = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle$$

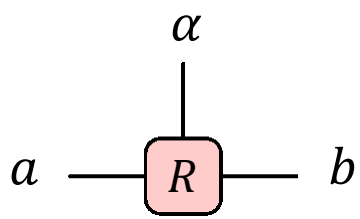
C^{11}は何もしない

行列積分解の具体的な表示とその解釈 3/3

R_α^{ab}の成分表示

$$R^{00} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad R^{11} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

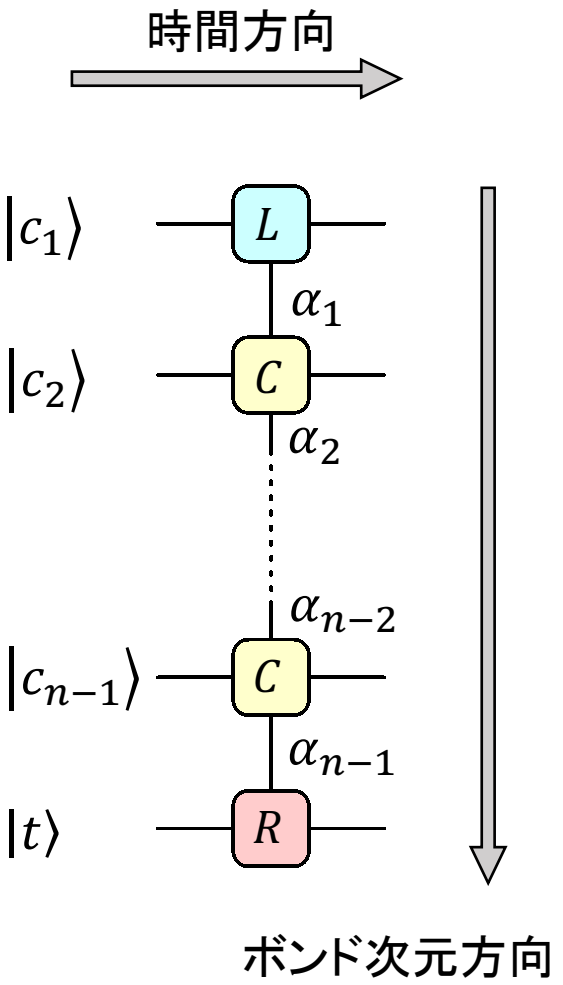
$$R^{01} = R^{10} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$



R_α^{ab}は物理的な量子ビット$|t\rangle$に作用する演算子

仮想的な量子ビットαを制御ビットとするCZゲート

$$R_{\alpha=0} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R_{\alpha=1} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$



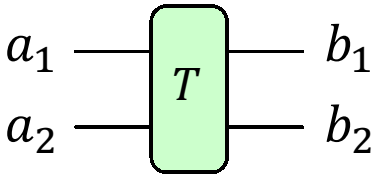
行列積分解を求める時の注意

1) ゲージ自由度により，TN表示は一意ではない

$$AB = AMM^{-1}B = (AM)(M^{-1}B) = A'B'$$

2) テンソルをどう行列表示するかによってその行列のランクは変わる

Ex. $n = 2$のCZゲート



$$\begin{matrix} & b_1b_2 & & & \\ \end{matrix}$$

$$\begin{matrix} & \\ a_1 & \\ a_2 & \\ \end{matrix}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}\begin{matrix} \\ a_1a_2 \end{matrix}$$

ランク4

$$\begin{matrix} & a_2b_2 & & & \\ \end{matrix}$$

$$\begin{matrix} & \\ a_1 & \\ b_1 & \\ \end{matrix}\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix}\begin{matrix} \\ a_1b_1 \end{matrix}$$

ランク2

Q. CXゲートの行列積分解は？（他のcontrolledゲートは？）

今日の講義のまとめ

1) 量子回路の復習

- 典型的な量子ゲート
- 今日の演習で扱う量子アルゴリズム → Grover & QBSC

Quantum bit string comparator

2) テンソルネットワークシミュレーション

- 量子回路のTN表示
- 確率振幅，縮約密度行列の計算
- 計算コスト
- 縮約計算の順序の工夫
- 量子ゲートの行列積分解 → **CZゲートの行列積分解**