

2022-07-14 第1回量子ソフトウェアシンポジウム  
「量子技術と量子ソフトウェアの未来」



# 量子コンピューティング × テンソルネットワーク

東京大学大学院理学系研究科

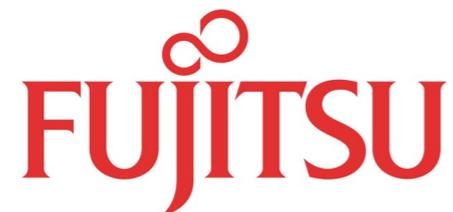
藤堂眞治 <[wistaria@phys.s.u-tokyo.ac.jp](mailto:wistaria@phys.s.u-tokyo.ac.jp)>

東京大学大学院理学系研究科

# 「量子ソフトウェア」 寄付講座

---

- <https://qsw.phys.s.u-tokyo.ac.jp/>
- 2021年6月設置
- 目標
  - 量子コンピュータとテンソルネットワーク・サンプリングなどの情報圧縮・情報抽出手法の組み合わせによる新しい量子機械学習手法や量子アプリケーションの開発、大規模シミュレーションによる量子コンピュータの背後に潜む物理の理解、最先端知見の獲得を通じ、社会実装における課題の解決、および、量子ネイティブ育成と研究成果の社会への還元を目指す
- 教員
  - 藤堂眞治 (教授)、大久保 毅 (特任准教授)、秋山進一郎 (特任助教)



# 「量子ソフトウェア」寄付講座

## 東京大学

大学院理学系研究科  
(知の物理学研究センターと連携)

### 「量子ソフトウェア」寄付講座

- 産業界の動向の把握
- 本学の量子関連研究領域の教育強化
- 協賛・協力企業との更なる産学連携の可能性

量子ネイティブ育成  
(専門講座等)

運営への協賛

## 協賛企業・協力企業

SCSK株式会社  
株式会社NTTデータ  
株式会社電通国際情報サービス  
日鉄ソリューションズ株式会社  
株式会社三井住友フィナンシャルグループ・株式会社日本総合研究所  
日本電気株式会社  
BIPROGY株式会社  
富士通株式会社  
blueqat株式会社  
凸版印刷株式会社

- 量子コンピュータ人材育成への貢献
- 自社のR&D、ビジネス応用に向けた人材育成やスキル向上
- 東京大学および協賛・協力企業間の共創、連携の強化

## 受講生 (学部生・大学院生 / 社会人)

- 量子機械学習や量子アプリケーションの最先端知見・スキルの獲得と社会実装における課題の解決
- 量子ネイティブの育成と研究成果の社会への還元

# 寄付講座における人材育成活動

---

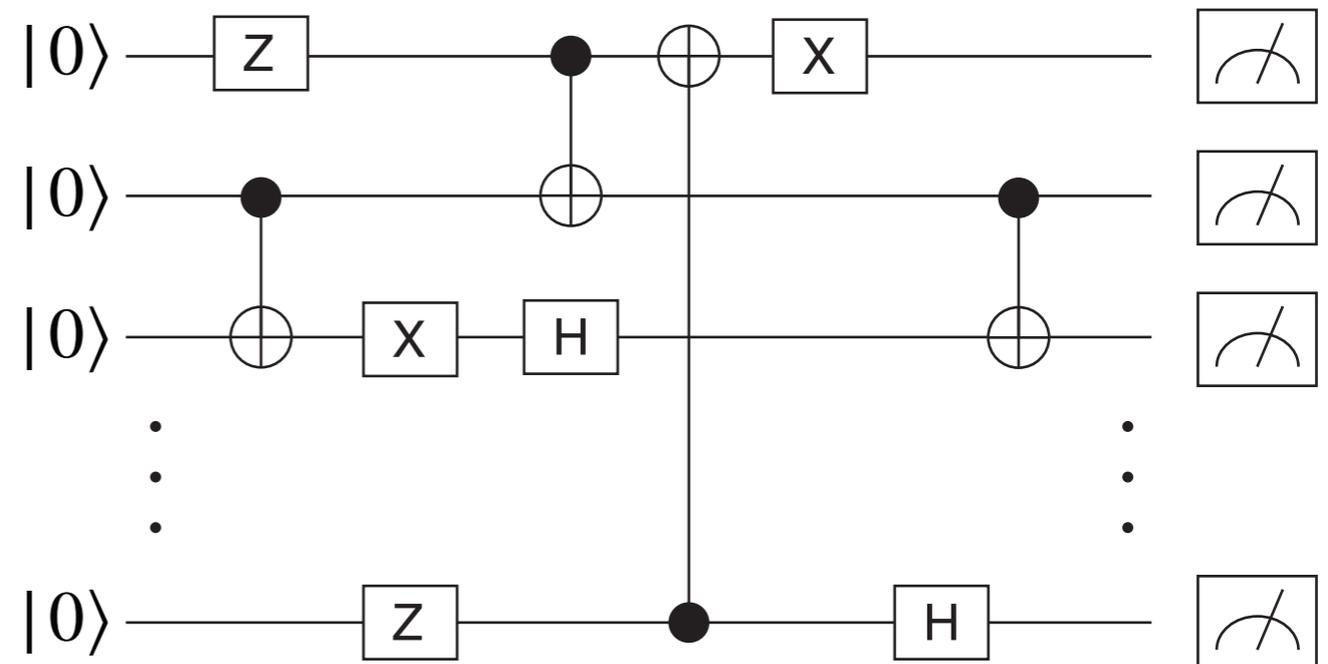
- 量子ソフトウェアワークショップ・シンポジウム
  - 年2回開催
- 大学院講義「計算科学・量子計算のための情報圧縮」
  - 2021年度パイロット講義実施、2022年度 秋学期開講
- 社会人向け講座 (次回: 2022年9月上旬開催予定)
  - 量子力学のための数学、量子コンピュータのための量子力学、テンソルネットワークとサンプリング等を対象に、セミナー形式で実施
- 産学協働ゼミ (次回: 2022年9月下旬開催予定)
  - 関係者の垣根を超えた情報共有、ディスカッションする機会
  - 協賛・協力企業の業務課題を持ち寄り、量子ソフトウェアの開発訓練による応用スキル習得

# 産学一体の活動

---

- 予算もスタッフも限られている
  - フルスペックの研究・教育ではなく得意な領域に特化する
  - テンソルネットワークとサンプリングをキーワードに
  - 量子コンピュータだけにこだわらず「量子インスパイアード古典アルゴリズム」「古典インスパイアード量子アルゴリズム」の研究を推進
- 量子デバイスメーカー～量子コンピュータ「ユーザ」まで様々な立場の協賛・協力企業が参画している利点を活かす
  - 全ての活動に協賛・協力企業が参加
  - 活動の企画、ワークショップ・シンポジウムの講演者、パイロット講義の受講、社会人向け講座・産学協働ゼミの講師、等
- 本寄付講座の趣旨にご賛同・ご支援いただける企業を募集しています！

# 量子コンピュータと量子回路

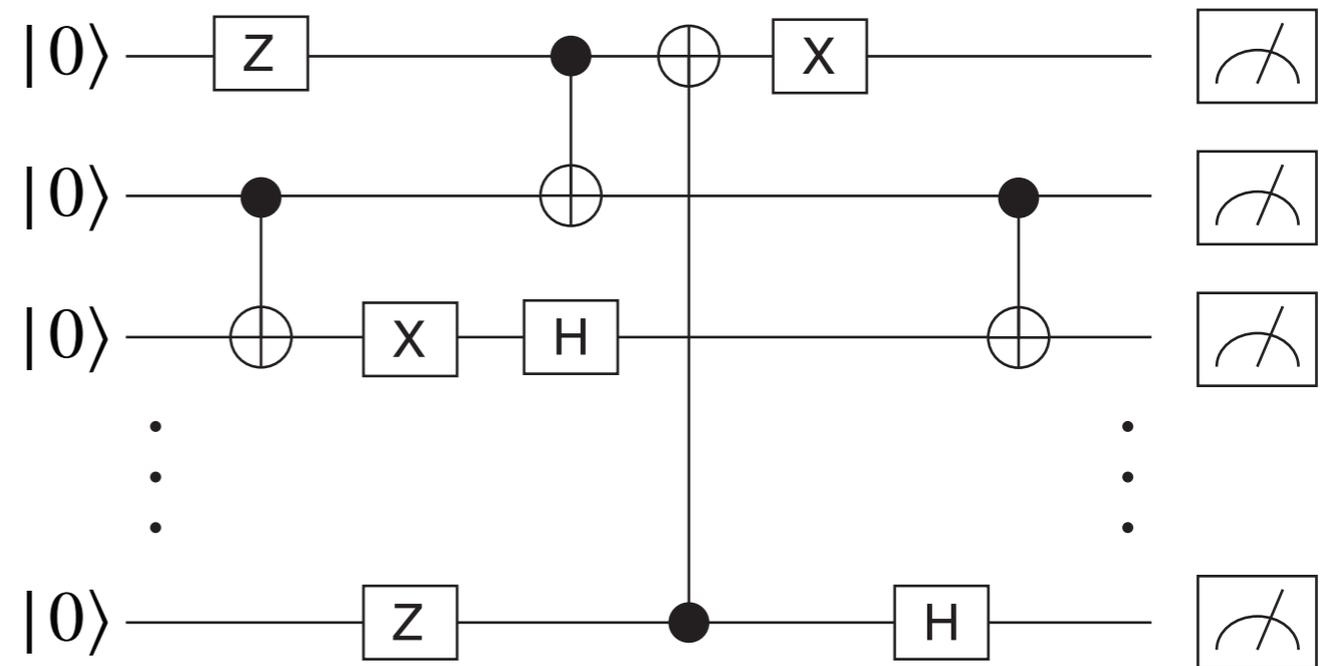
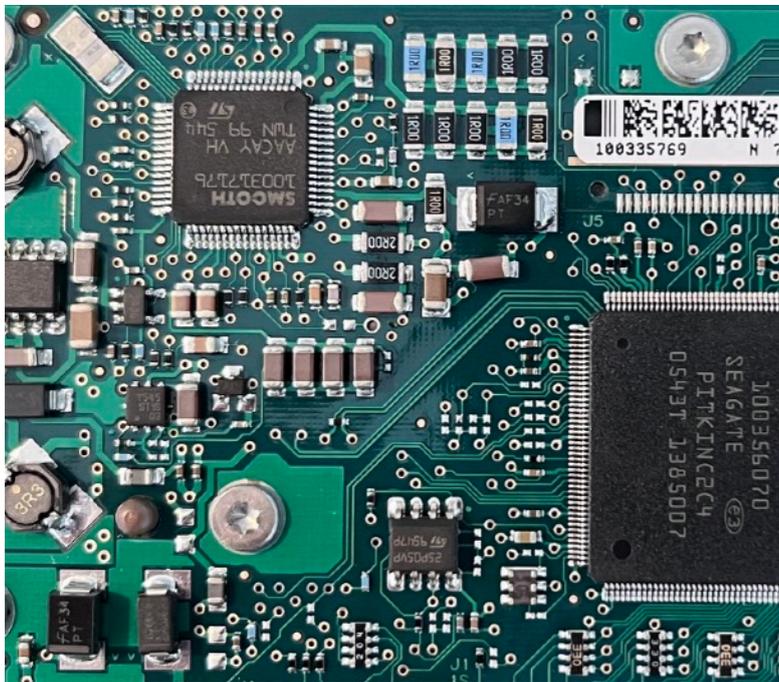


- 量子コンピュータ

- @東京大学 浅野キャンパス ハードウェア・テストセンター

- 量子コンピュータの中には量子回路？

# 電子回路と量子回路



- 電子回路

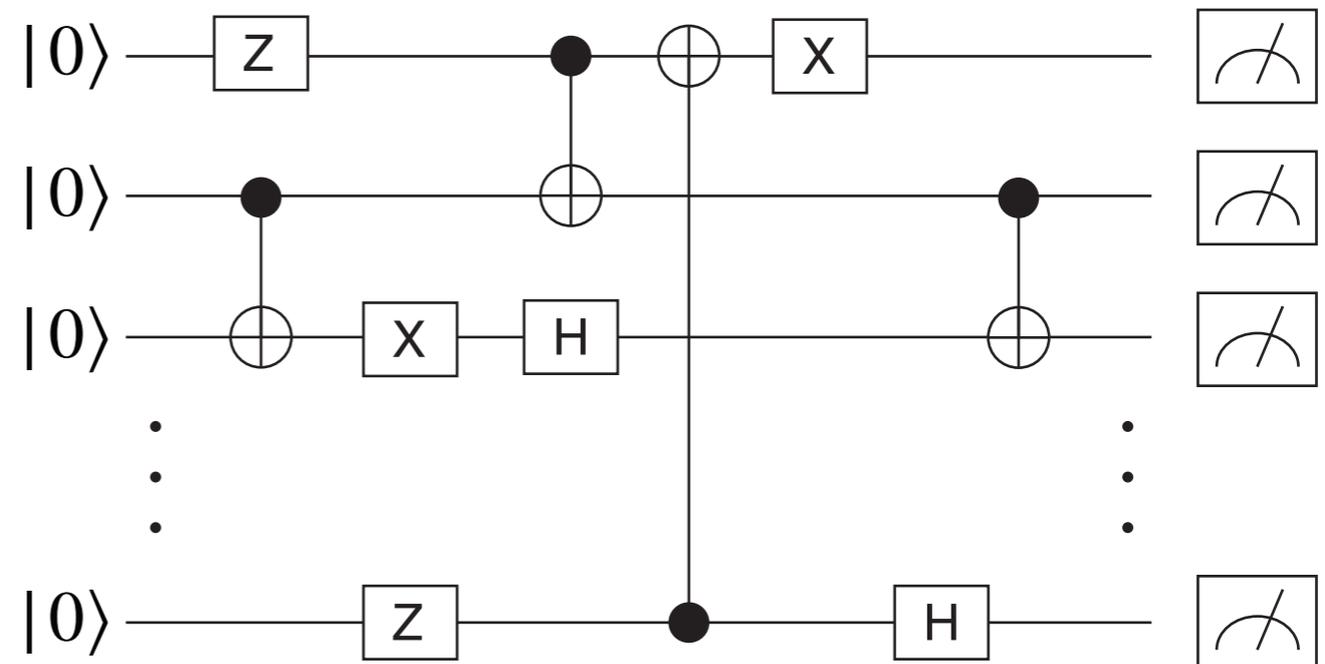
- 電圧や流れる電流が時間的に変化して「計算」を行う

- 量子回路：横軸は時間

- 左から右へ進むにつれて「計算」が進む

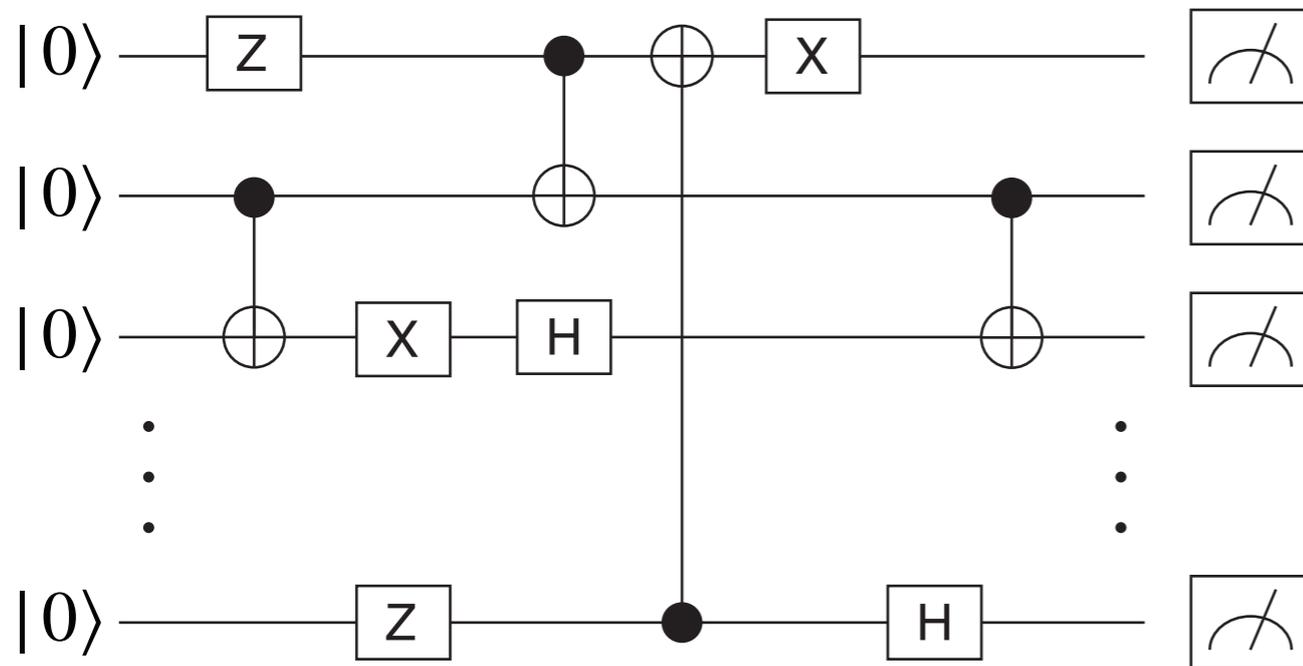
# プログラムと量子回路

```
x = 1;  
y = 2;  
z = 2*x + y*y;  
...
```



- プログラム
  - 上から下へ進むにつれて「計算」が進む
- 量子回路：横軸は時間
  - 左から右へ進むにつれて「計算」が進む

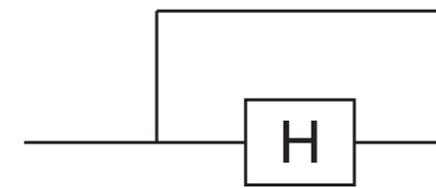
# 健全な量子回路



• 量子回路：横軸は時間

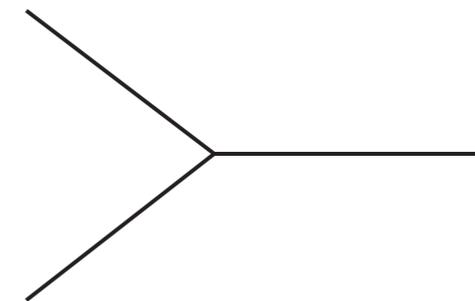
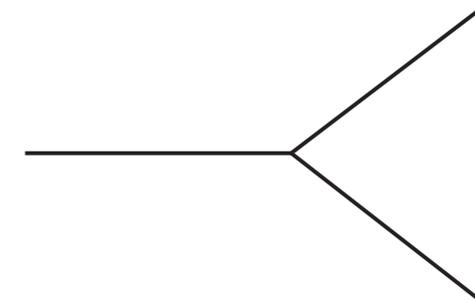
- 左端：入力（初期状態）  $|000\dots 0\rangle$
- 左から右へゲートを通過するにつれて状態が変化
- 右端：出力（観測）

• ループは禁止

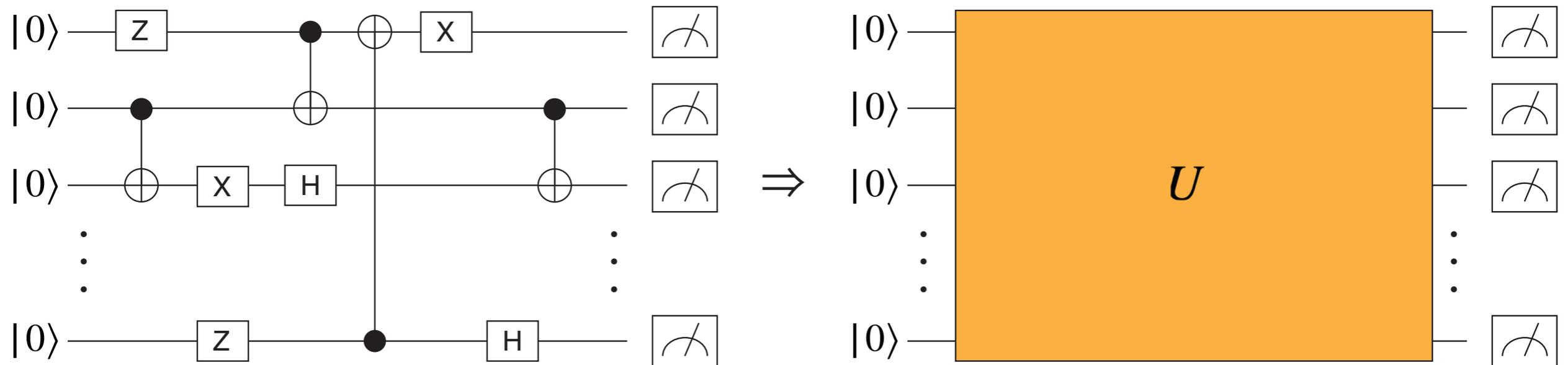


• 確率の保存

• 量子ビットの数は保存



# 量子回路の標準形



- 量子回路：時間は左から右に
  - 量子ゲートをひとまとめにして  $U$  と書く
- 数式：時間は右から左に

$$U = U_{23}^{\text{CNOT}} H_n X_1 U_{1n}^{\text{CNOT}} \dots X_3 U_{23}^{\text{CNOT}} Z_1$$

# 典型的な量子ゲート

---

• Xゲート (別名: NOTゲート) 

• 状態  $|0\rangle$  を入力すると  $|1\rangle$  が出力される  $|0\rangle$    $\Rightarrow |1\rangle$

• 状態  $|1\rangle$  を入力すると  $|0\rangle$  が出力される  $|1\rangle$    $\Rightarrow |0\rangle$

• ベクトル・行列表示

• 状態  $|0\rangle$  を  $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ 、状態  $|1\rangle$  を  $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$  と書くと  $X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

•  $X|0\rangle = |1\rangle$ 、 $X|1\rangle = |0\rangle$  (時間は右から左に)

• 重ね合わせの状態  $\alpha|0\rangle + \beta|1\rangle$  を入力すると

•  $X(\alpha|0\rangle + \beta|1\rangle) = \alpha X|0\rangle + \beta X|1\rangle = \alpha|1\rangle + \beta|0\rangle$

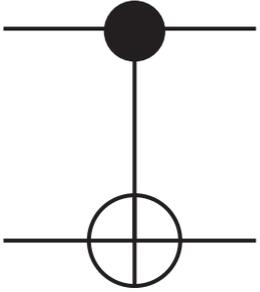
• それぞれが反転したものの重ね合わせ状態が出力される

# 典型的な量子ゲート

---

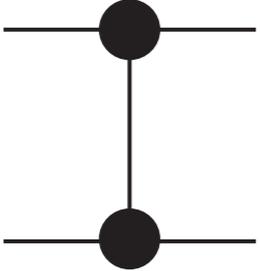
- CXゲート (別名: CNOT、制御NOT)

- 1番目の量子ビットが  $|1\rangle$  のときだけ2番目のビットに  $X$  を作用する


$$U^{\text{CNOT}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

- CZゲート (別名: 制御Z)

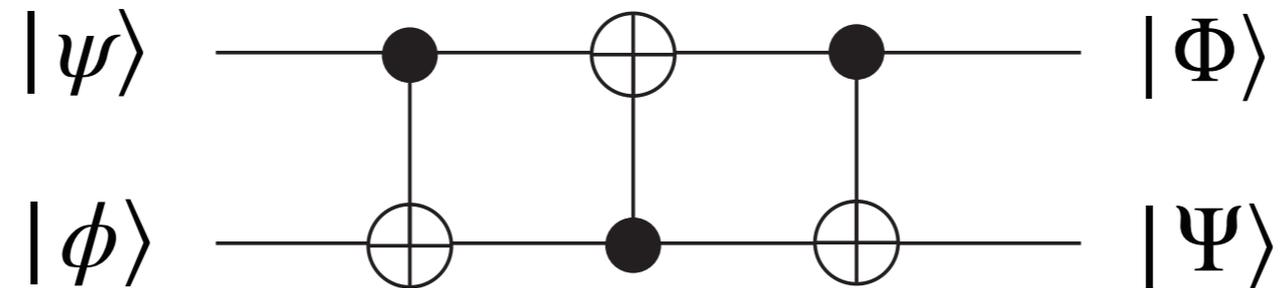
- 1番目と2番目の量子ビットがともに  $|1\rangle$  のとき、状態の符号を反転する


$$U^{\text{CZ}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

# 簡単な量子ゲート

---

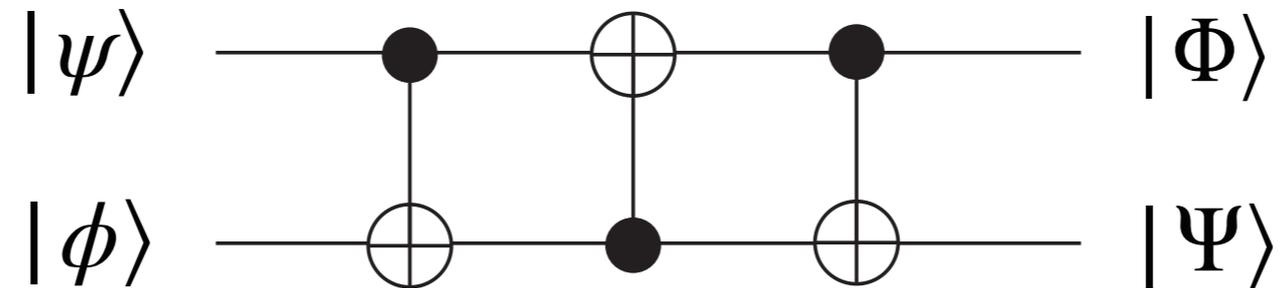
## • SWAPゲート



- $|00\rangle$  状態を入力すると  $|00\rangle$  が出力される
  - $|00\rangle \Rightarrow |00\rangle \Rightarrow |00\rangle \Rightarrow |00\rangle$
- $|01\rangle, |10\rangle, |11\rangle$  状態を入力してみると
  - $|01\rangle \Rightarrow |01\rangle \Rightarrow |11\rangle \Rightarrow |10\rangle$
  - $|10\rangle \Rightarrow |11\rangle \Rightarrow |01\rangle \Rightarrow |01\rangle$
  - $|11\rangle \Rightarrow |10\rangle \Rightarrow |10\rangle \Rightarrow |11\rangle$

# 簡単な量子ゲート

- SWAPゲート



- **重ね合わせ状態**  $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ ,  $|\phi\rangle = \gamma|0\rangle + \delta|1\rangle$  を代入すると

- $|\psi\phi\rangle = (\alpha|0\rangle + \beta|1\rangle)(\gamma|0\rangle + \delta|1\rangle)$   
 $= \alpha\gamma|00\rangle + \alpha\delta|01\rangle + \beta\gamma|10\rangle + \beta\delta|11\rangle$

- $\Rightarrow \alpha\gamma|00\rangle + \alpha\delta|10\rangle + \beta\gamma|01\rangle + \beta\delta|11\rangle$   
 $= \gamma(\alpha|00\rangle + \beta|01\rangle) + \delta(\alpha|10\rangle + \beta|11\rangle)$   
 $= (\gamma|0\rangle + \delta|1\rangle)(\alpha|0\rangle + \beta|1\rangle) = |\phi\psi\rangle$

- **任意の状態**が入れ替え可能

# COPYゲート

- 量子状態の入れ替えはできた
  - 量子状態のコピーはできるか？
- 古典論理回路では OR ゲート（あるいはXORゲート）を使えば可能



- 量子回路でも CX ゲートを使えばできるか？



# COPYゲート

---

- しかし、1番目の量子ビットに重ね合わせ状態を入れてみると

$$\begin{aligned} |\psi 0\rangle &= (\alpha|0\rangle + \beta|1\rangle)|0\rangle = \alpha|00\rangle + \beta|10\rangle \\ &\Rightarrow \alpha|00\rangle + \beta|11\rangle \end{aligned}$$

- これは、欲しかった出力とは違う

$$\begin{aligned} |\psi\psi\rangle &= (\alpha|0\rangle + \beta|1\rangle)(\alpha|0\rangle + \beta|1\rangle) \\ &= \alpha^2|00\rangle + \alpha\beta|01\rangle + \beta\alpha|10\rangle + \beta^2|11\rangle \end{aligned}$$

- No Cloning 定理 「任意の状態をコピーする量子回路を作ることはできない」
  - 量子アルゴリズム設計の難しさ
  - しかし全ての古典アルゴリズムを含んでいる

# 典型的な量子ゲート

---

• Hゲート (別名：アダマールゲート) 

• 状態  $|0\rangle$  を入力すると  $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$  が出力される

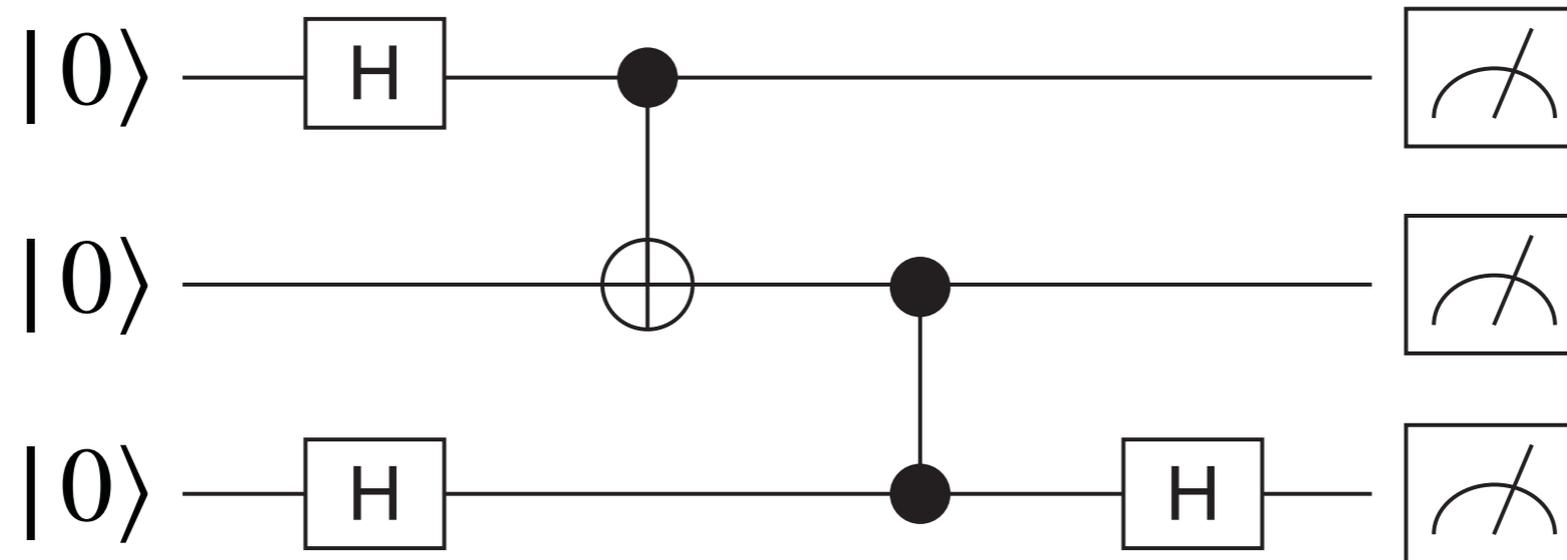
• 状態  $|1\rangle$  を入力すると  $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$  が出力される

• 行列表示

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

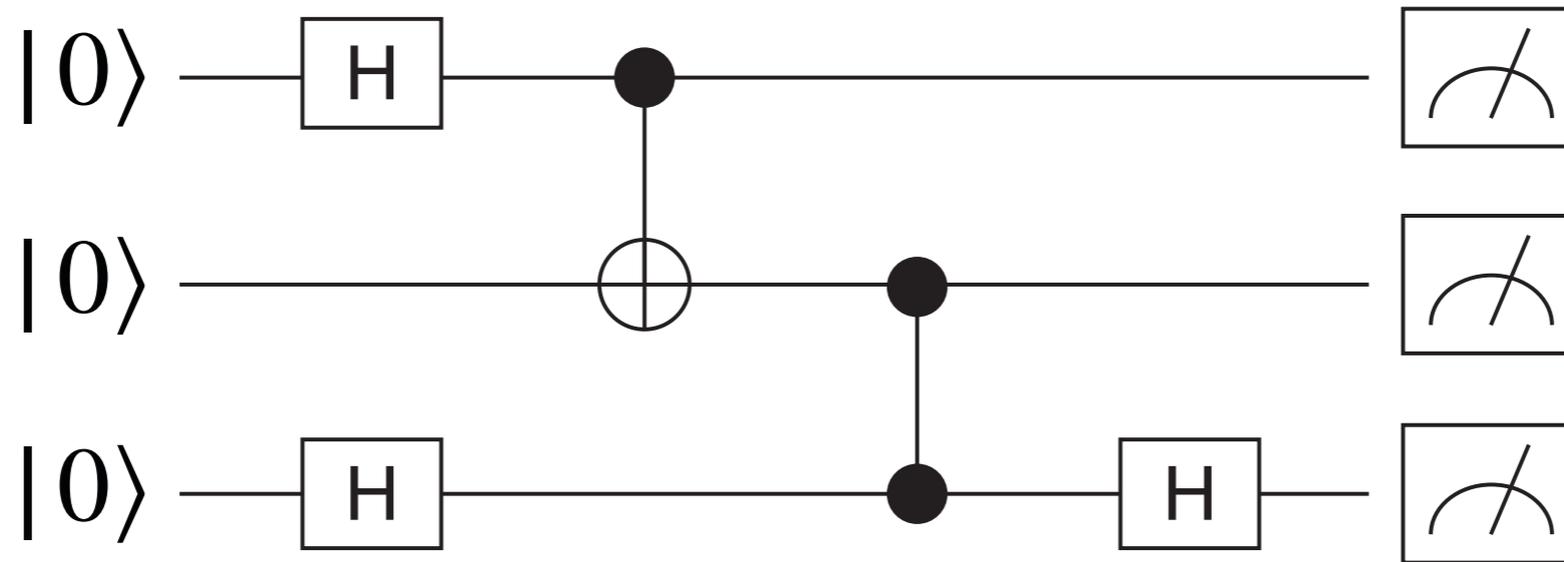
• 「重ね合わせ状態」を生成できる

# 状態の分岐と干渉

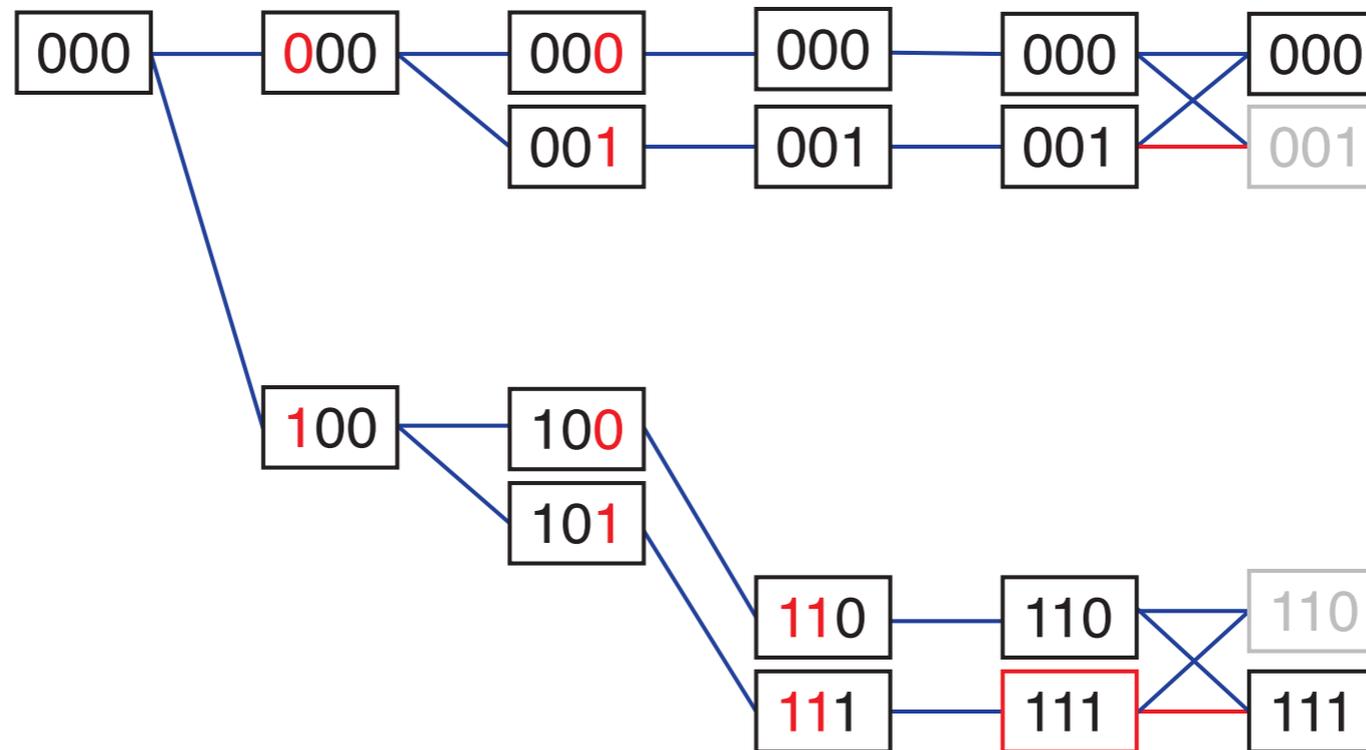


$$\begin{aligned}
 \bullet \quad |000\rangle &\Rightarrow \frac{1}{2}(|0\rangle + |1\rangle)|0\rangle(|0\rangle + |1\rangle) = \frac{1}{2}(|000\rangle + |001\rangle + |100\rangle + |101\rangle) \\
 &\Rightarrow \frac{1}{2}(|000\rangle + |001\rangle + |110\rangle + |111\rangle) \Rightarrow \frac{1}{2}(|000\rangle + |001\rangle + |110\rangle - |111\rangle) \\
 &\Rightarrow \frac{1}{2\sqrt{2}}(|00\rangle + |11\rangle)(|0\rangle + |1\rangle) + (|00\rangle - |11\rangle)(|0\rangle - |1\rangle) \\
 &= \frac{1}{2\sqrt{2}}(|000\rangle + |001\rangle + |110\rangle + |111\rangle + |000\rangle - |001\rangle - |110\rangle + |111\rangle) \\
 &= \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)
 \end{aligned}$$

# 状態の分岐と干渉



## • 状態遷移図



• 干渉により状態が消滅 ⇒ サンプリングにおける「負符号問題」

# 量子回路による量子演算

---

- 並列性

- 重ね合わせ状態を入力すると、対応する出力も重ね合わせとなる

- 分岐

- Hゲートを作用させると状態が分岐する

- 干渉

- 各状態の重ね合わせ係数は複素数であり、互いに打ち消しあって消滅することがある

- 収縮

- 測定に用いた基底のいずれかの状態に収縮(崩壊)する

# テンソルとテンソルネットワーク

- テンソルとは

- ベクトルや行列を一般化した概念

- ベクトル：添字が1つ  $\Rightarrow$  1脚テンソル

- 行列：添字が2つ  $\Rightarrow$  2脚テンソル

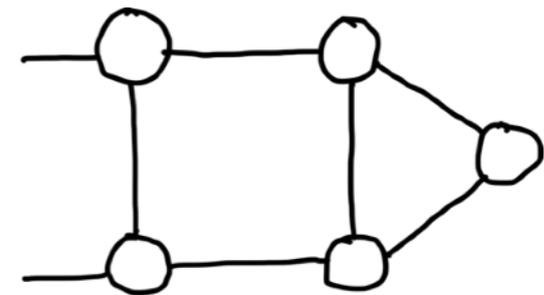
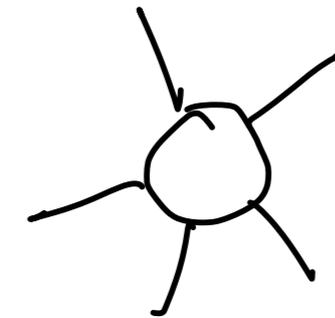
- 一般化： $n$  本足への一般化  $\Rightarrow n$  脚テンソル

- それぞれの足の次元：ボンド次元 ( $\chi$  と書く)

- テンソルネットワーク (テンソルダイヤグラム)

- 行列・行列の積の一般化

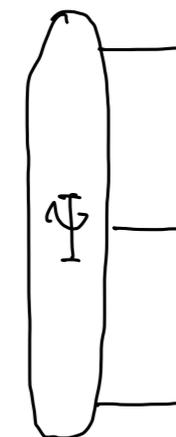
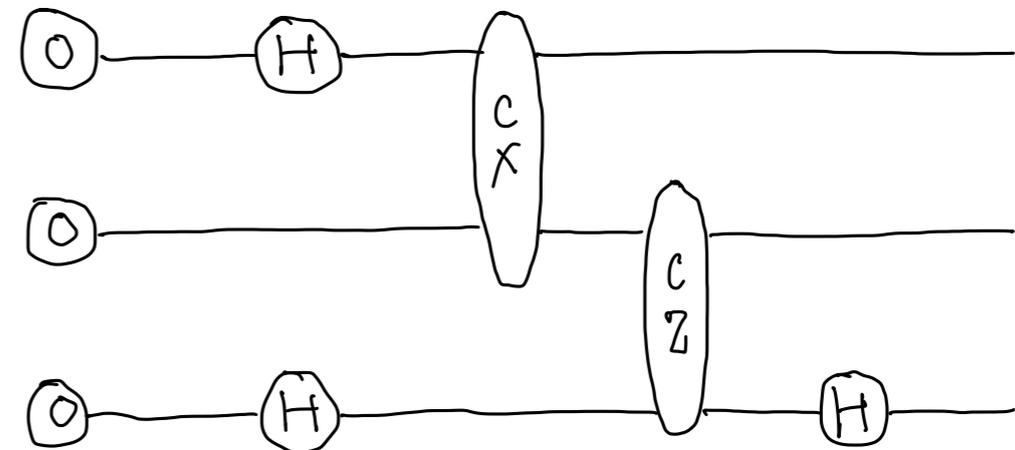
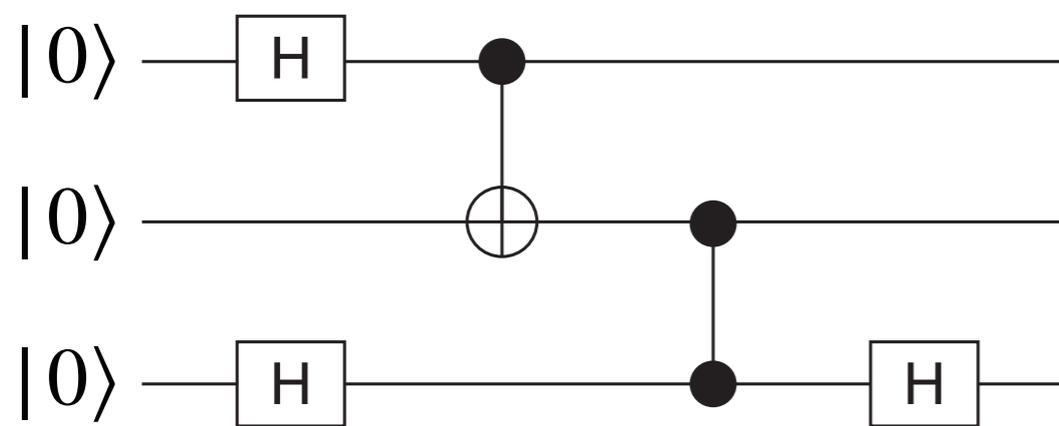
- 線をつないだ足について和を取る (縮約)



$$A = TR \quad \Leftrightarrow \quad A_{ij} = \sum_k T_{ik} R_{kj} \quad \Leftrightarrow \quad \text{---} \circ \text{---} = \text{---} \circ \text{---} \circ \text{---}$$

# 量子回路とテンソルネットワーク

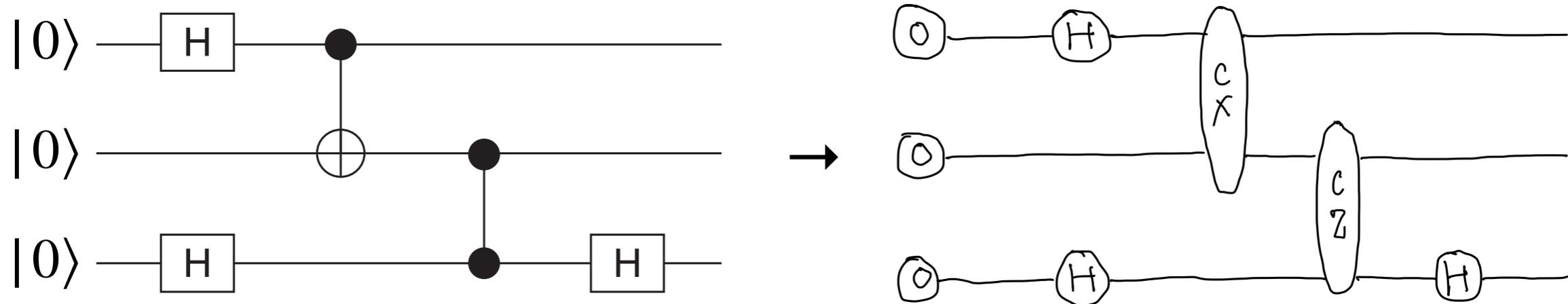
- 量子回路  $\Rightarrow$  テンソルネットワーク



- 1ビットゲート  $\rightarrow$  2脚テンソル
- 2ビットゲート  $\rightarrow$  4脚テンソル
- 初期状態  $\rightarrow$  1脚テンソル(ベクトル)の組
- 出力量子状態  $\rightarrow$  テンソルネットワークを縮約することで得られる

# 量子回路とテンソルネットワーク

- 量子回路  $\Rightarrow$  テンソルネットワーク



- 量子回路では必ず左から右に時間が進む
  - テンソルネットワークの縮約はどのような順番で計算しても結果は同じ
- テンソルネットワークの古典コンピュータでの計算は非常にコストが高い
  - 縮約順序を工夫することで計算量を劇的に減らせる可能性
  - 縮約に「近似」の導入が必要 (低ランク近似、テンソルくりこみ群)
  - 新しい近似の方向性 (サンプリング?)

# 量子コンピュータの古典シミュレーション

---

- 量子ソフトウェア・アルゴリズムの**精度・性能評価**
  - 古典/量子ハイブリッドアルゴリズム、量子機械学習、量子誤り訂正
- **古典計算の限界**を知る
  - 量子超越の本当の境界はどこか？
- **量子技術**の古典コンピュータ上での利用
  - 量子アルゴリズムを古典コンピュータで近似
- 量子回路は典型的な**量子多体系**
  - 量子回路のシミュレーション技術 → 物理やその他の分野への展開
  - 超伝導・量子液体・素粒子・原子核
  - 複雑系・時系列データ・機械学習
- **量子コンピューティングの本質**を知る
  - 別の形に書き直してはじめて理解できる

# テンソルネットワークから量子回路へ

- テンソルネットワークはさまざまな場面で現れる
  - 量子多体系・複雑系シミュレーション
    - 素粒子物理～原子核～物性物理学～流体力学
  - 機械学習・データ科学
- テンソルネットワークを量子回路に「埋め込む」ことができるならば指数関数的に加速
  - ⇒ 量子シミュレーション・量子機械学習
- 「埋め込み」への問題点
  - 時間の流れる方向・ループ禁止
  - 確率保存・量子ビット数保存
- サンプリング・観測との組み合わせ

