

# NISQ (Noisy Intermediate-Scale Quantum technology)

## マシンを用いた量子機械学習

藤井 啓祐

大阪大学基礎工学研究科 教授

大阪大学先導的学際研究機構

量子情報量子生命部門 副センター長



大阪大学  
OSAKA UNIVERSITY



大阪大学先導的学際研究機構

量子情報・量子生命研究部門

# 自己紹介



藤井啓祐

大阪大学大学院基礎工学研究科システム創成専攻 教授

- ・ 2011年3月京都大学大学院工学研究科 博士（工学）
- ・ 大阪大学大学院基礎工学研究科 特別研究員（2011-2013）
- ・ 京都大学白眉センター特定助教（2013-2016）
- ・ 東京大学光量子科学研究センター助教（2016-2017）
- ・ 京都大学大学院理学研究科物理学・宇宙物理学専攻、特定准教授（2017-2019）

を経て2019年4月から大阪大学大学院基礎工学研究科 教授

# 自己紹介



藤井啓祐

大阪大学大学院基礎工学研究科システム創成専攻 教授

- ・ 2011年3月京都大学大学院工学研究科 博士（工学）
  - ・ 大阪大学大学院基礎工学研究科 特別研究員（2011-2013）
  - ・ 京都大学白眉センター特定助教（2013-2016）
  - ・ 東京大学光量子科学研究センター助教（2016-2017）
  - ・ 京都大学大学院理学研究科物理学・宇宙物理学専攻、特定准教授（2017-2019）
- を経て2019年4月から大阪大学大学院基礎工学研究科 教授

- ・ 基礎的に面白いこと(量子力学)を工学的に応用する(コンピュータ).
- ・ 次の時代を作るような誰もやっていない新しいことをやる.

# 自己紹介



藤井啓祐

大阪大学大学院基礎工学研究科システム創成専攻 教授

- ・ 2011年3月京都大学大学院工学研究科 博士（工学）
  - ・ 大阪大学大学院基礎工学研究科 特別研究員（2011-2013）
  - ・ 京都大学白眉センター特定助教（2013-2016）
  - ・ 東京大学光量子科学研究センター助教（2016-2017）
  - ・ 京都大学大学院理学研究科物理学・宇宙物理学専攻、特定准教授（2017-2019）
- を経て2019年4月から大阪大学大学院基礎工学研究科 教授

- ・ 基礎的に面白いこと(量子力学)を工学的に応用する(コンピュータ).
- ・ 次の時代を作るような誰もやっていない新しいことをやる.

この分野の教科書など書いてます:



# 自己紹介



藤井啓祐

大阪大学大学院基礎工学研究科システム創成専攻 教授

- ・ 2011年3月京都大学大学院工学研究科 博士（工学）
  - ・ 大阪大学大学院基礎工学研究科 特別研究員（2011-2013）
  - ・ 京都大学白眉センター特定助教（2013-2016）
  - ・ 東京大学光量子科学研究センター助教（2016-2017）
  - ・ 京都大学大学院理学研究科物理学・宇宙物理学専攻、特定准教授（2017-2019）
- を経て2019年4月から大阪大学大学院基礎工学研究科 教授

- ・ 基礎的に面白いこと(量子力学)を工学的に応用する(コンピュータ)。
- ・ 次の時代を作るような誰もやっていない新しいことをやる。

この分野の教科書など書いてます:



授業をYoutubeで配信

第2回：量子力学の復習

[講義ノート][演習課題]

きちんと量子力学を復習したい人は、[量子物理学I・II](#)を参照ください。

# 自己紹介



藤井啓祐

大阪大学大学院基礎工学研究科システム創成専攻 教授

- ・ 2011年3月京都大学大学院工学研究科 博士（工学）
  - ・ 大阪大学大学院基礎工学研究科 特別研究員（2011-2013）
  - ・ 京都大学白眉センター特定助教（2013-2016）
  - ・ 東京大学光量子科学研究センター助教（2016-2017）
  - ・ 京都大学大学院理学研究科物理学・宇宙物理学専攻、特定准教授（2017-2019）
- を経て2019年4月から大阪大学大学院基礎工学研究科 教授

- ・ 基礎的に面白いこと(量子力学)を工学的に応用する(コンピュータ)。
- ・ 次の時代を作るような誰もやっていない新しいことをやる。

この分野の教科書など書いてます:

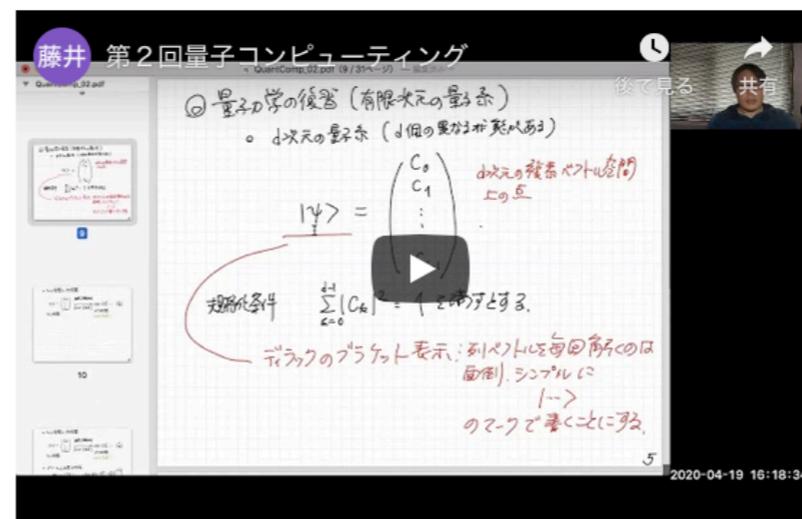


授業をYoutubeで配信

第2回：量子力学の復習

[講義ノート][演習課題]

きっちり量子力学を復習したい人は、量子物理学・IIを参照ください。



Twitterやってます:



# 目次

- ・はじめに
  - ・量子計算ミニマム
  - ・量子コンピュータの現状
- ・NISQマシンを用いた量子機械学習
  - ・量子レザバー計算
  - ・量子回路学習

# 目次

- ・はじめに
  - ・量子計算ミニマム
  - ・量子コンピュータの現状
- ・NISQマシンを用いた量子機械学習
  - ・量子レザバー計算
  - ・量子回路学習

中嶋浩平@東京大学



御手洗光祐



根来誠

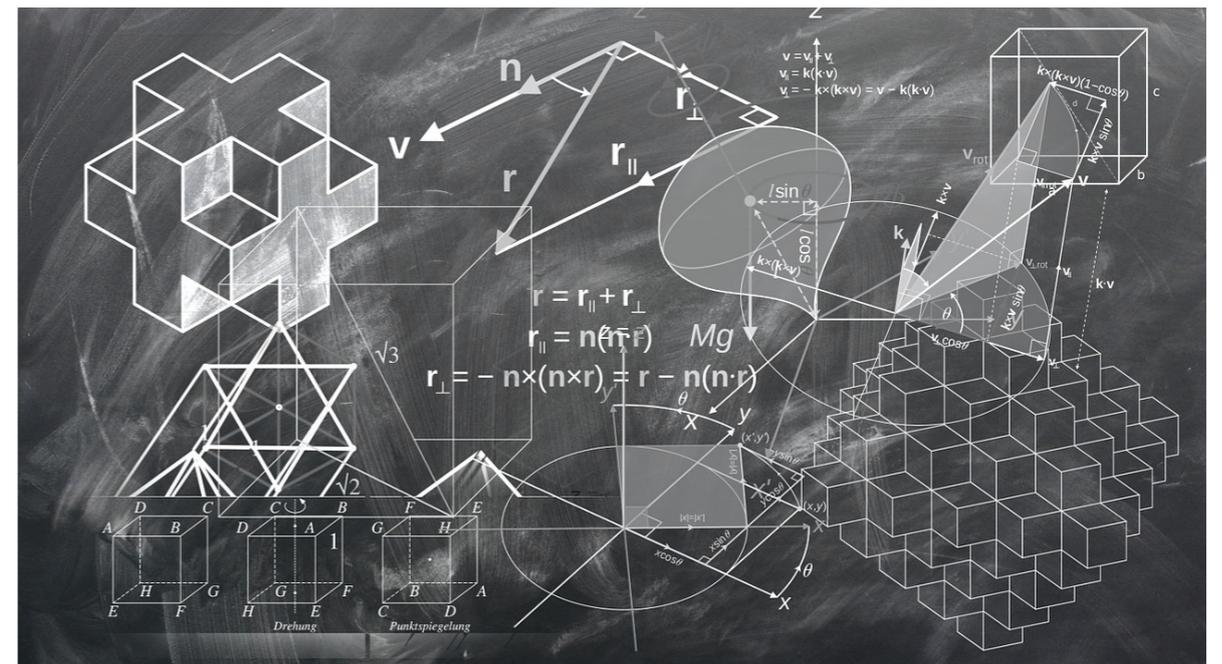


北川勝浩 @大阪大学



その他たくさん共同研究者

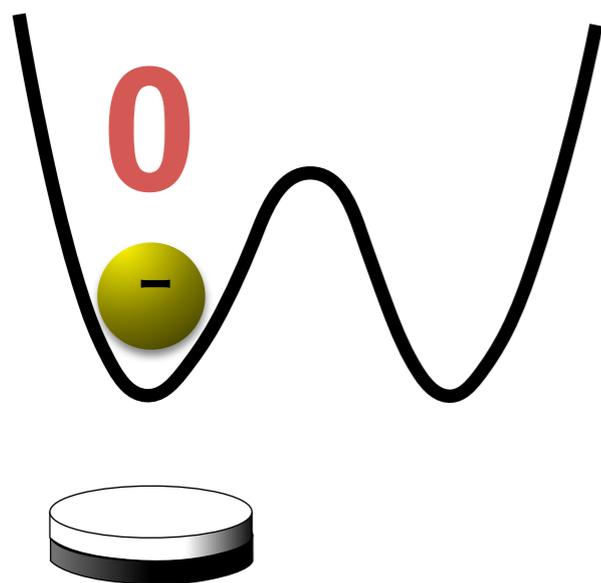
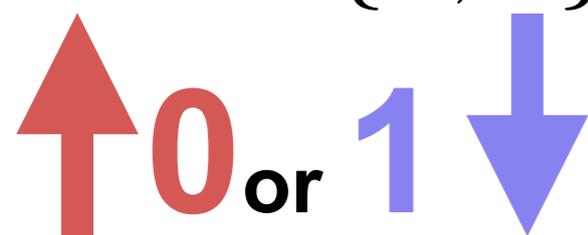
# 量子計算ミニマム



# 量子ビット

古典ビット:

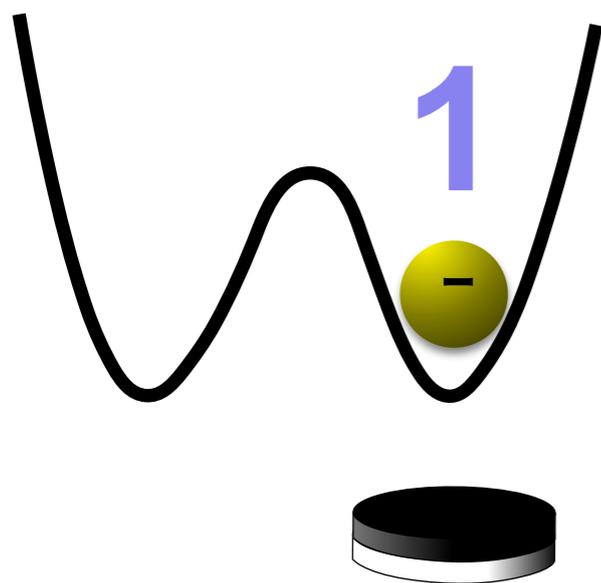
$$x \in \{0, 1\}$$



# 量子ビット

古典ビット:

$$x \in \{0, 1\}$$



# 量子ビット

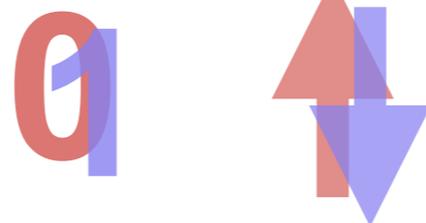
古典ビット:

$$x \in \{0, 1\}$$



量子ビット:

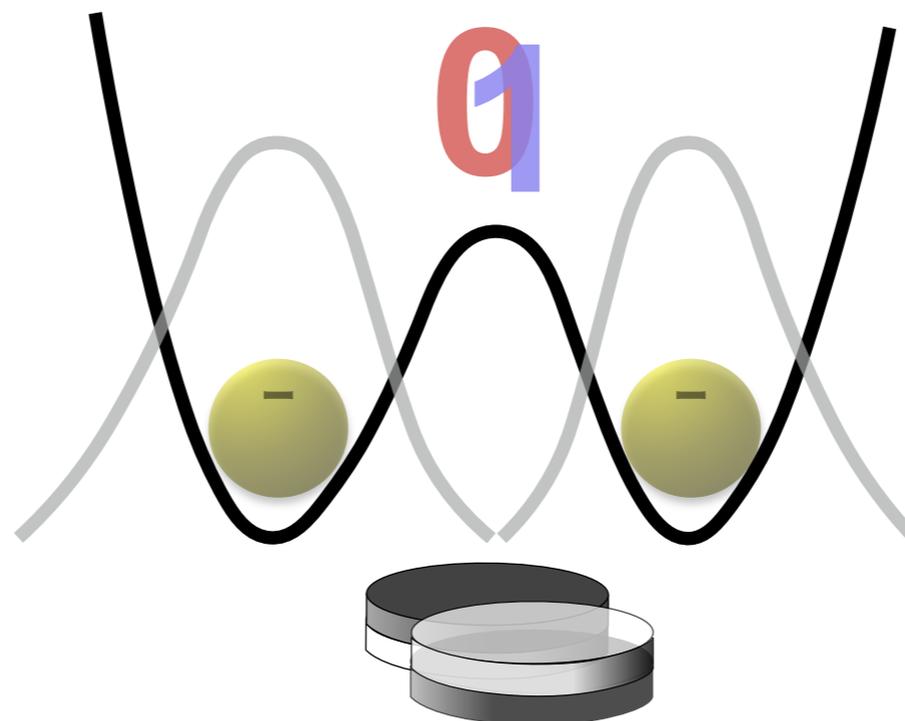
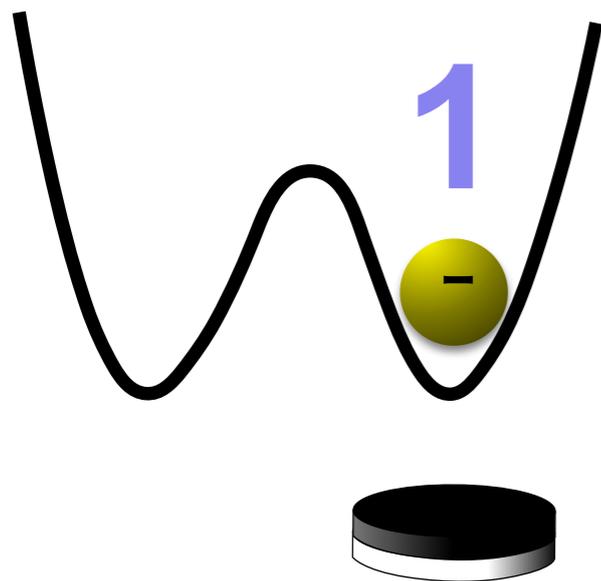
$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$



複素ベクトル空間

0と1の重ね合わせ状態

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$



# 量子ビット

古典ビット:

$$x \in \{0, 1\}$$



量子ビット:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

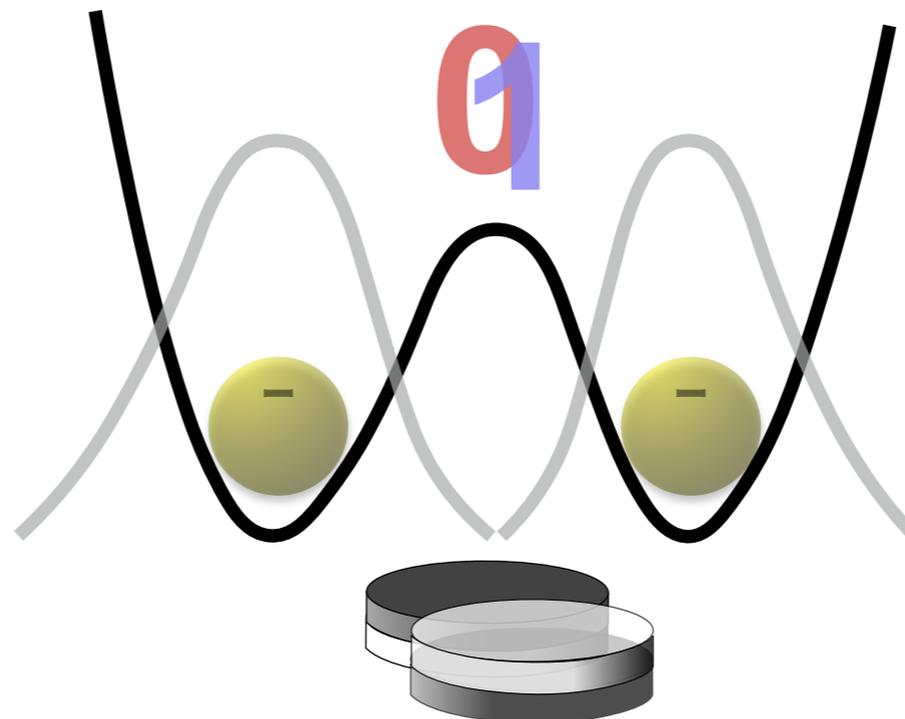
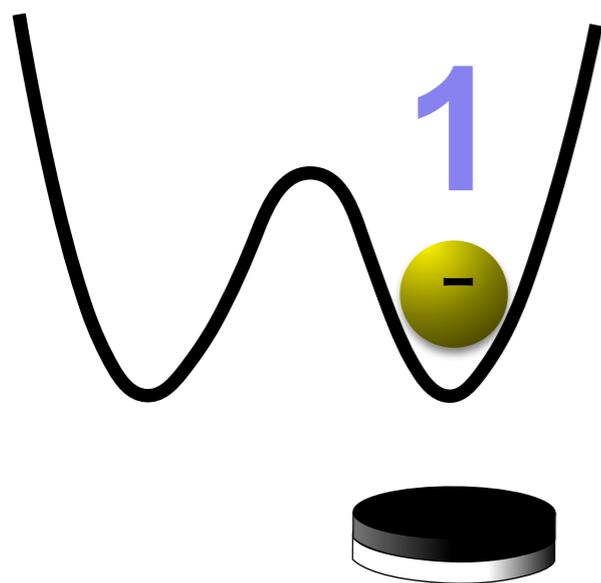
0



複素ベクトル空間

0と1の重ね合わせ状態

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$



0か1かはまだ決定していない

# 量子ビット

古典ビット:

$$x \in \{0, 1\}$$



量子ビット:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

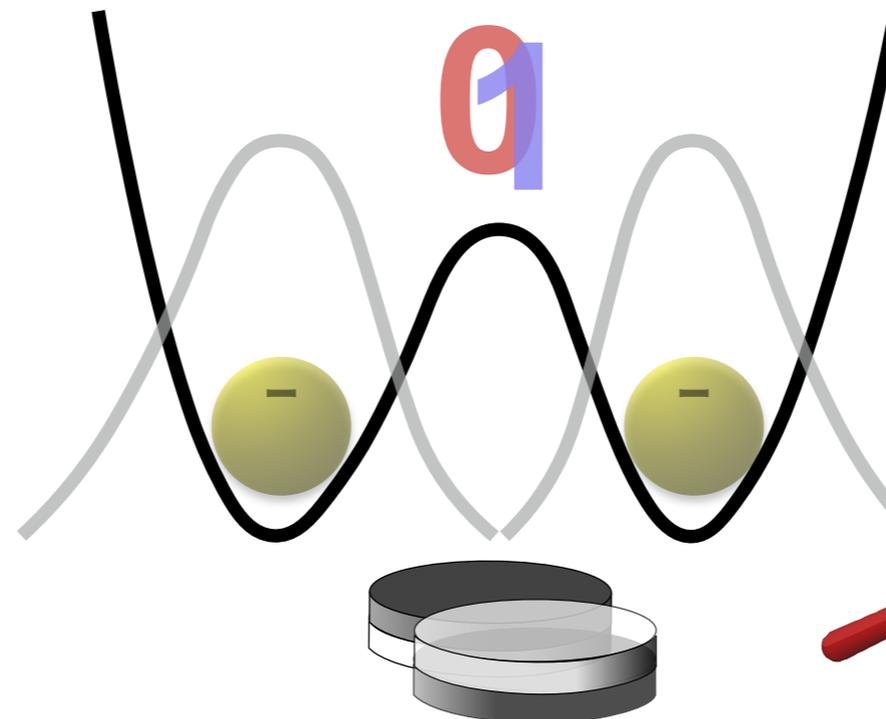
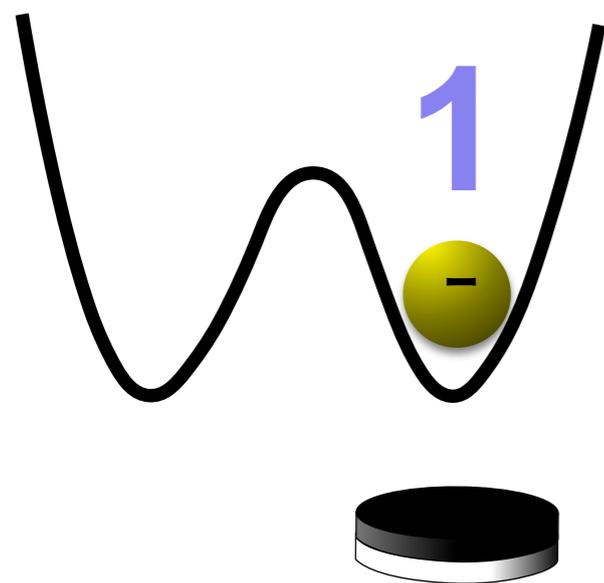
0



複素ベクトル空間

0と1の重ね合わせ状態

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$



測定 0 or 1 ?

$$p_0 = |\alpha|^2$$

$$p_1 = |\beta|^2$$

0か1かはまだ決定していない

絶対値の2乗が確率



# さまざまな量子ビット

## 超伝導量子ビット

Nakamura *et al.*, Nature **398**, 786 (1999)

量子化した電気回路

中村・蔡先生

## イオントラップ

*D. Winland*

<http://patapsco.nist.gov/ImageGallery/details.cfm?imageId=576>

原子(イオン)の内部状態&振動状態

## キャビティQED

原子の内部状態&光

*S. Haroche*

[http://wikipedia.org/wiki/Serge\\_Haroche](http://wikipedia.org/wiki/Serge_Haroche)

## 光量子ビット

J. Carolan *et al.*, Science **349**, 711 (2015)

単一光子(空間モード, 偏光モード)

## 量子ドット

T.D. Ladd *et al.*, Nature **464**, 45 (2010)

半導体量子井戸に捕獲した電子

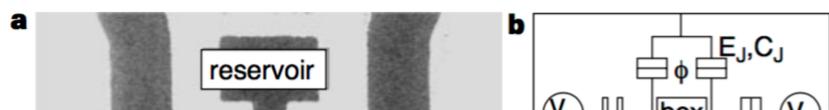
## トポロジカル量子ビット

*S. M. Albrecht et al.*, Nature **531**, 206 (2016)

マヨラナゼロモード  
(半導体ナノワイヤ+超伝導)

# さまざまな量子ビット

## 超伝導量子ビット



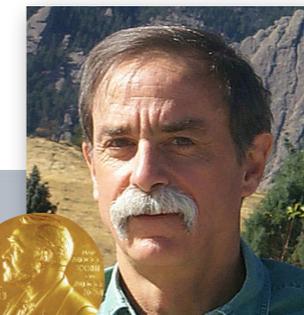
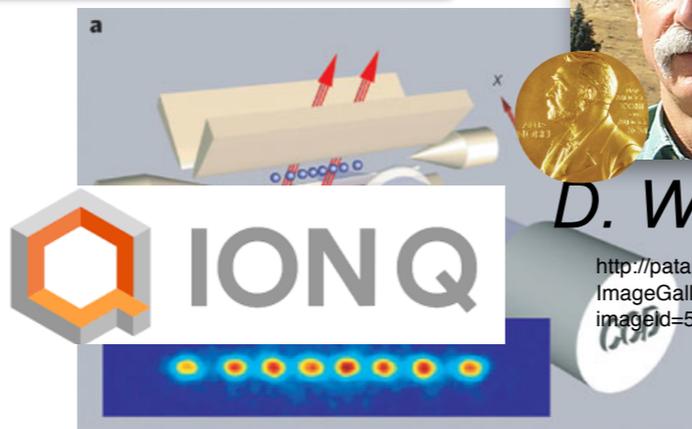
Yonakani et al., Nature 398, 786 (1999)

量子化した電気回路



中村・蔡先生

## イオントラップ

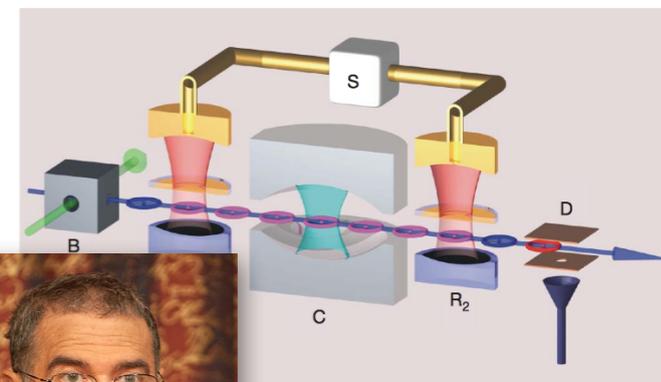


D. Winland

<http://patapsco.nist.gov/ImageGallery/details.cfm?imageId=576>

原子(イオン)の内部状態 & 振動状態

## キャビティQED

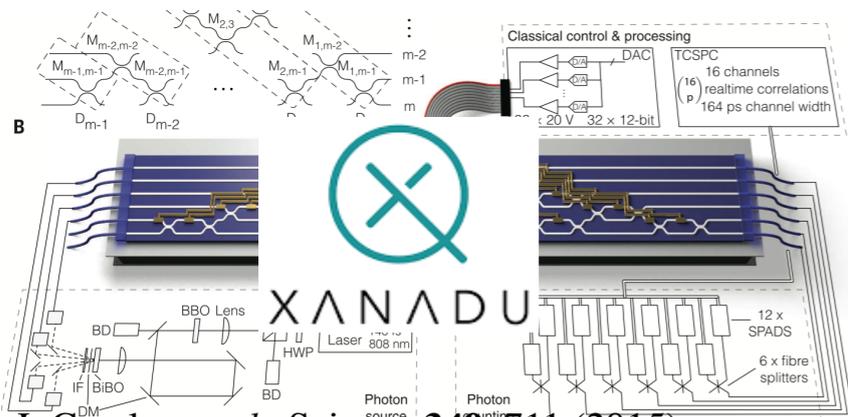


[http://wikipedia.org/wiki/Serge\\_Haroche](http://wikipedia.org/wiki/Serge_Haroche)

S. Haroche

原子の内部状態 & 光

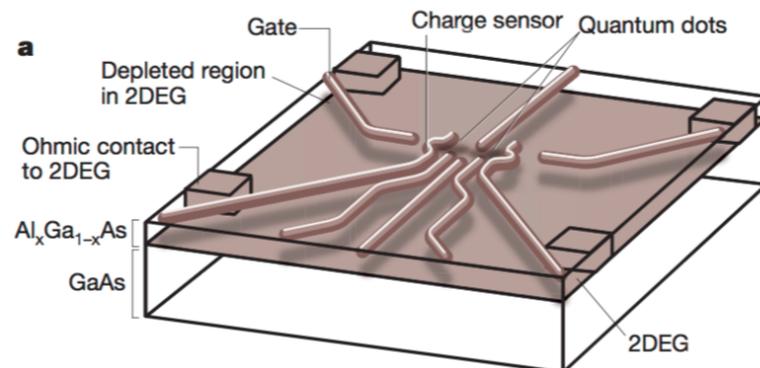
## 光量子ビット



J. Carolan et al., Science 349, 711 (2015)

単一光子(空間モード, 偏光モード)

## 量子ドット



T.D. Ladd et al., Nature 464, 45 (2010)

半導体量子井戸に捕獲した電子

## トポロジカル量子ビット

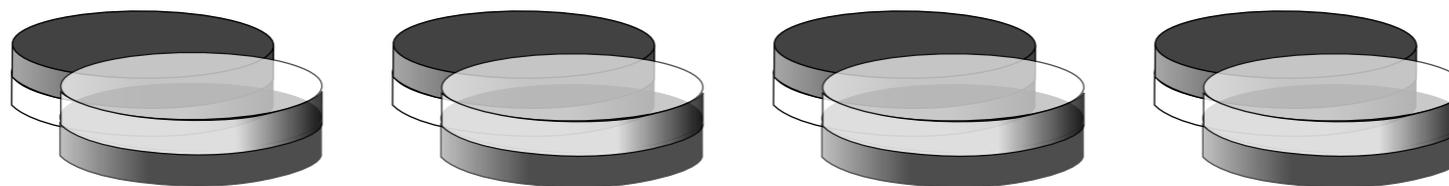


S. M. Albrecht et al., Nature 531, 206 (2016)

マヨラナゼロモード  
(半導体ナノワイヤ+超伝導)

# 量子コンピュータの仕組み

量子ビットの数が増えたら…



**N量子ビット →  $2^N$ パターンの重ね合わせ**

$$|\psi\rangle = \begin{pmatrix} c_{00\dots0} \\ c_{00\dots1} \\ \vdots \\ c_{11\dots1} \end{pmatrix}$$

**次元 (倍精度のメモリ)**

$$2^{10} = 1024$$

$$2^{20} = 10^6$$

$$2^{30} = 10^9 \text{ (16Gbyte)}$$

$$2^{40} = 10^{12} \text{ (16Tbyte)}$$

$$2^{50} = 10^{15} \text{ (16Pbyte)}$$

- 指数的に多くのパターンの**重ね合わせ**

# 量子コンピュータの仕組み

量子ビットの数が増えたら…

ユニタリ行列

$$\begin{pmatrix} c'_{00\dots 0} \\ c'_{00\dots 1} \\ \vdots \\ c'_{11\dots 1} \end{pmatrix} = \begin{pmatrix} & & & \\ & \mathbf{U} & & \\ & & & \\ & & & \end{pmatrix} \begin{pmatrix} c_{00\dots 0} \\ c_{00\dots 1} \\ \vdots \\ c_{11\dots 1} \end{pmatrix}$$

# 量子コンピュータの仕組み

量子ビットの数が増えたら…

ボルン則=絶対値2乗

$$\begin{array}{l}
 p_{00\dots0} \\
 p_{00\dots1} \\
 \vdots \\
 p_{11\dots1}
 \end{array}
 \leftarrow
 \begin{pmatrix}
 c'_{00\dots0} \\
 c'_{00\dots1} \\
 \vdots \\
 c'_{11\dots1}
 \end{pmatrix}
 =
 \begin{pmatrix}
 \text{ユニタリ行列} \\
 \mathbf{U}
 \end{pmatrix}
 \begin{pmatrix}
 c_{00\dots0} \\
 c_{00\dots1} \\
 \vdots \\
 c_{11\dots1}
 \end{pmatrix}$$

確率

直接複素ベクトルにはアドレスできない

# 量子コンピュータの仕組み

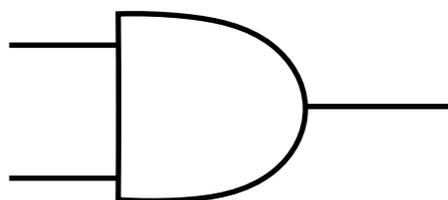
量子ビットの数が増えたら…

ボルン則=絶対値2乗

ユニタリ行列

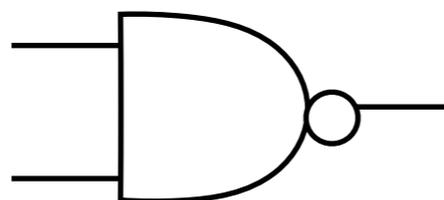
$$\begin{array}{l} \text{確率} \\ p_{00\dots0} \\ p_{00\dots1} \\ \vdots \end{array} \leftarrow \begin{pmatrix} c'_{00\dots0} \\ c'_{00\dots1} \\ \vdots \end{pmatrix} = \begin{pmatrix} U \end{pmatrix} \begin{pmatrix} c_{00\dots0} \\ c_{00\dots1} \\ \vdots \end{pmatrix}$$

古典コンピュータの基本ゲート



**AND演算**

In1	In2	Out
0	0	0
0	1	0
1	0	0
1	1	1



**NAND演算**

In1	In2	Out
0	0	1
0	1	1
1	0	1
1	1	0

古典の万能素子

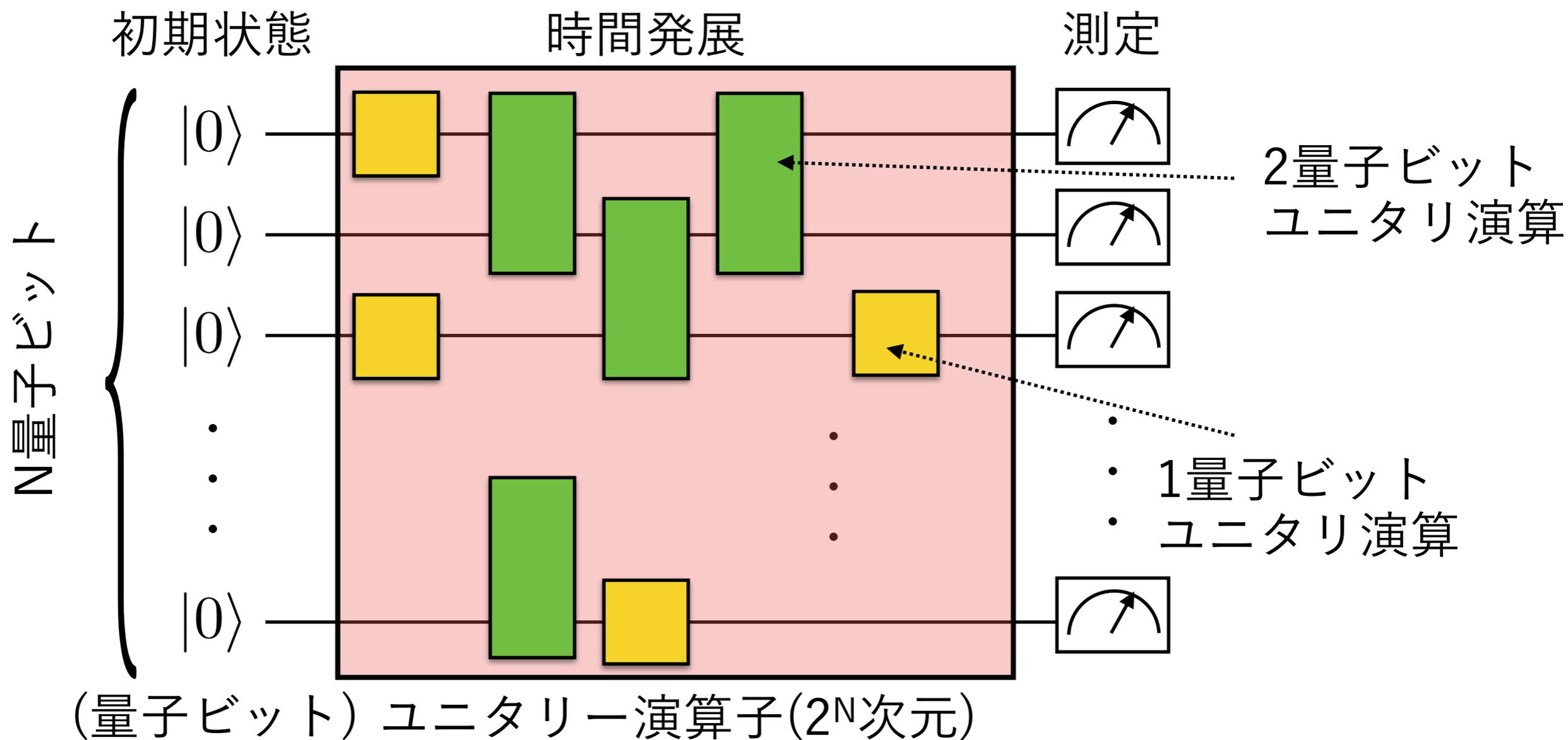
=任意の論理回路

(ブール関数) を構成できる.

# 量子コンピュータの仕組み

量子ビットの数が増えたら…

$$\begin{array}{l} \text{確率} \\ p_{00\dots0} \\ p_{00\dots1} \\ \vdots \\ p_{11\dots1} \end{array} \leftarrow \begin{pmatrix} c'_{00\dots0} \\ c'_{00\dots1} \\ \vdots \\ c'_{11\dots1} \end{pmatrix} = \begin{pmatrix} \text{ユニタリ行列} \\ U \end{pmatrix} \begin{pmatrix} c_{00\dots0} \\ c_{00\dots1} \\ \vdots \\ c_{11\dots1} \end{pmatrix}$$



# 量子コンピュータの仕組み

量子ビットの数が増えたら...

ユニタリ行列

$$p_{00\dots 0} \quad (c'_{00\dots 0}) \quad (c_{00\dots 0})$$

## 量子コンピュータの基本ゲート

$$\text{---} \boxed{Rz(\theta)} \text{---} \quad e^{-i(\theta/2)Z} = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}$$

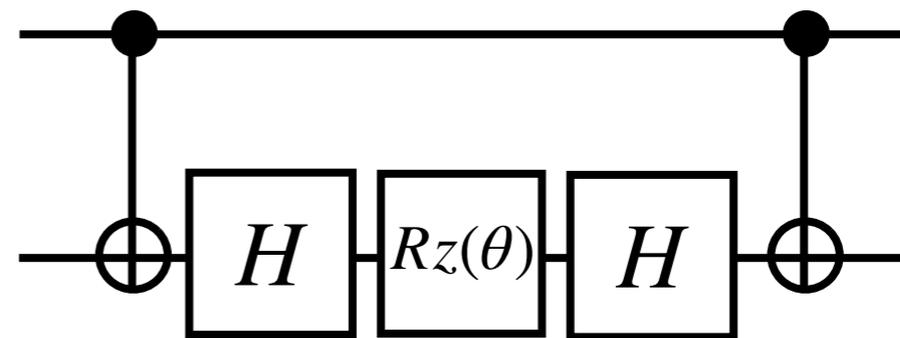
$$\text{---} \boxed{H} \text{---} \quad \frac{1}{\sqrt{2}}(X + Z) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

N量子ビット

$$\begin{array}{c} \text{---} \bullet \text{---} \\ | \\ \text{---} \oplus \text{---} \end{array} \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$|0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X$$

例  $e^{-i\theta X \otimes X}$



量子ビット  
ユニタリ演算

ビット  
演算

(量子ビット) ユニタリー演算子(2<sup>N</sup>次元)

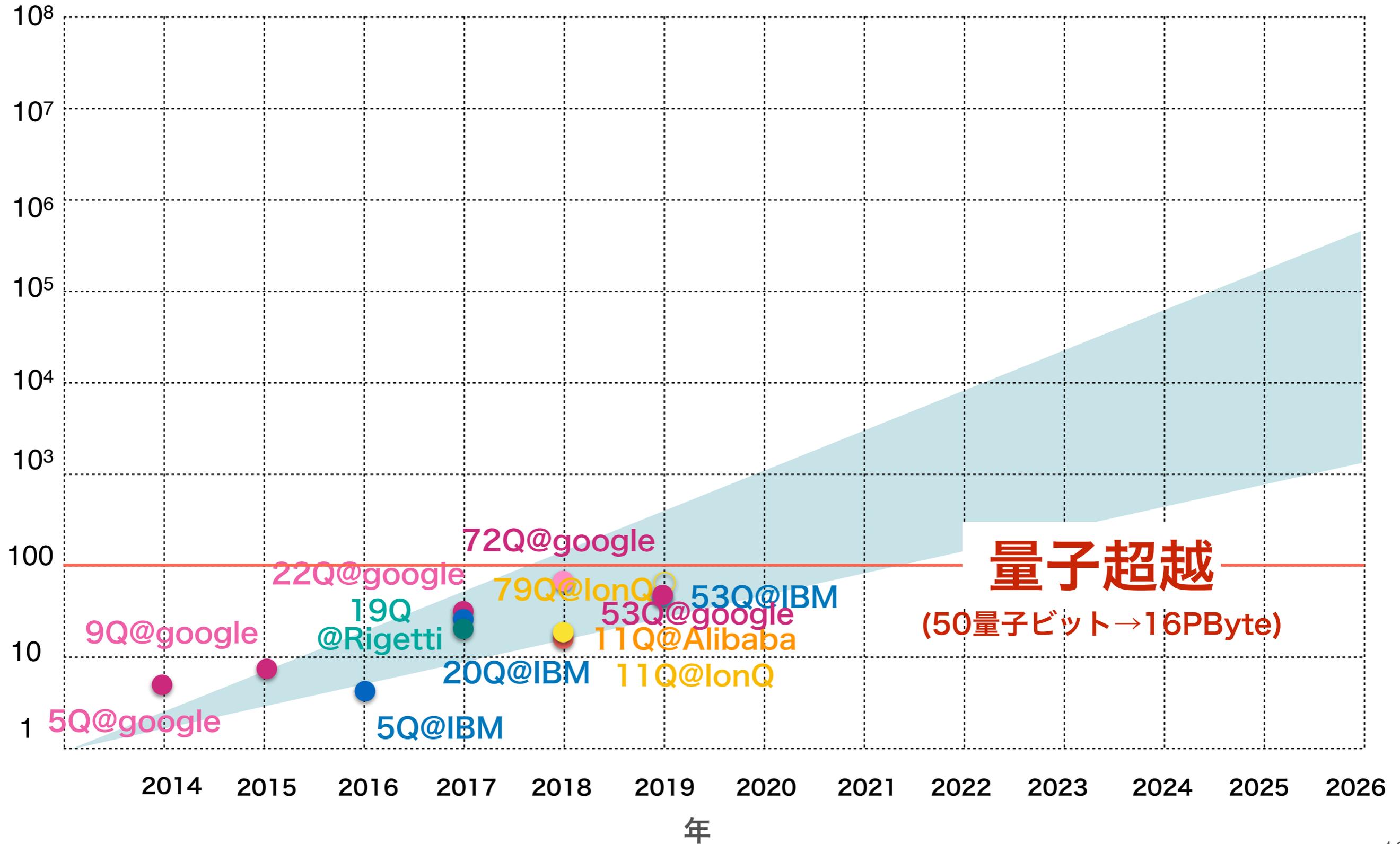


# 量子コンピュータ の現状



# “量子版”ムーアの法則？

量子ビット数



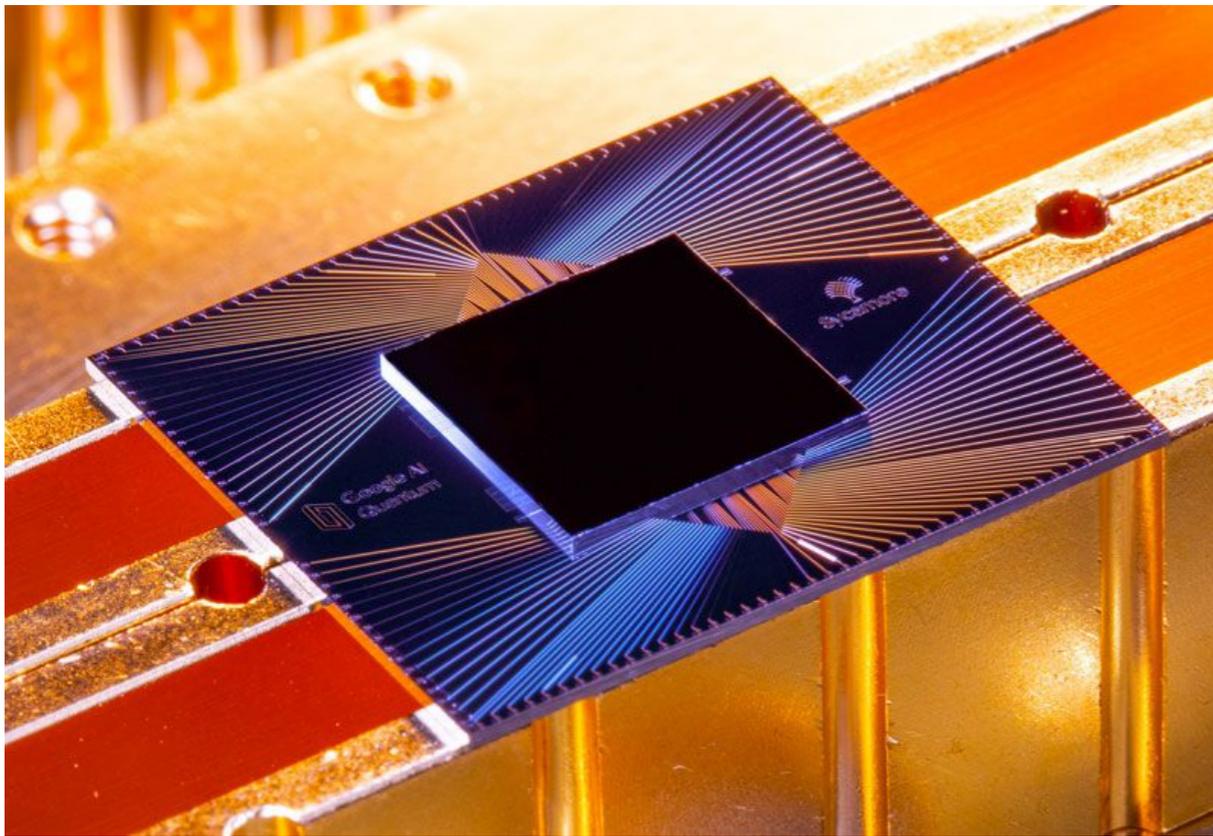
# Googleによる量子超越

NEWS · 23 OCTOBER 2019

## Hello quantum world! Google publishes landmark quantum supremacy claim

The company says that its quantum computer is the first to perform a calculation that would be practically impossible for a classical machine.

Elizabeth Gibney



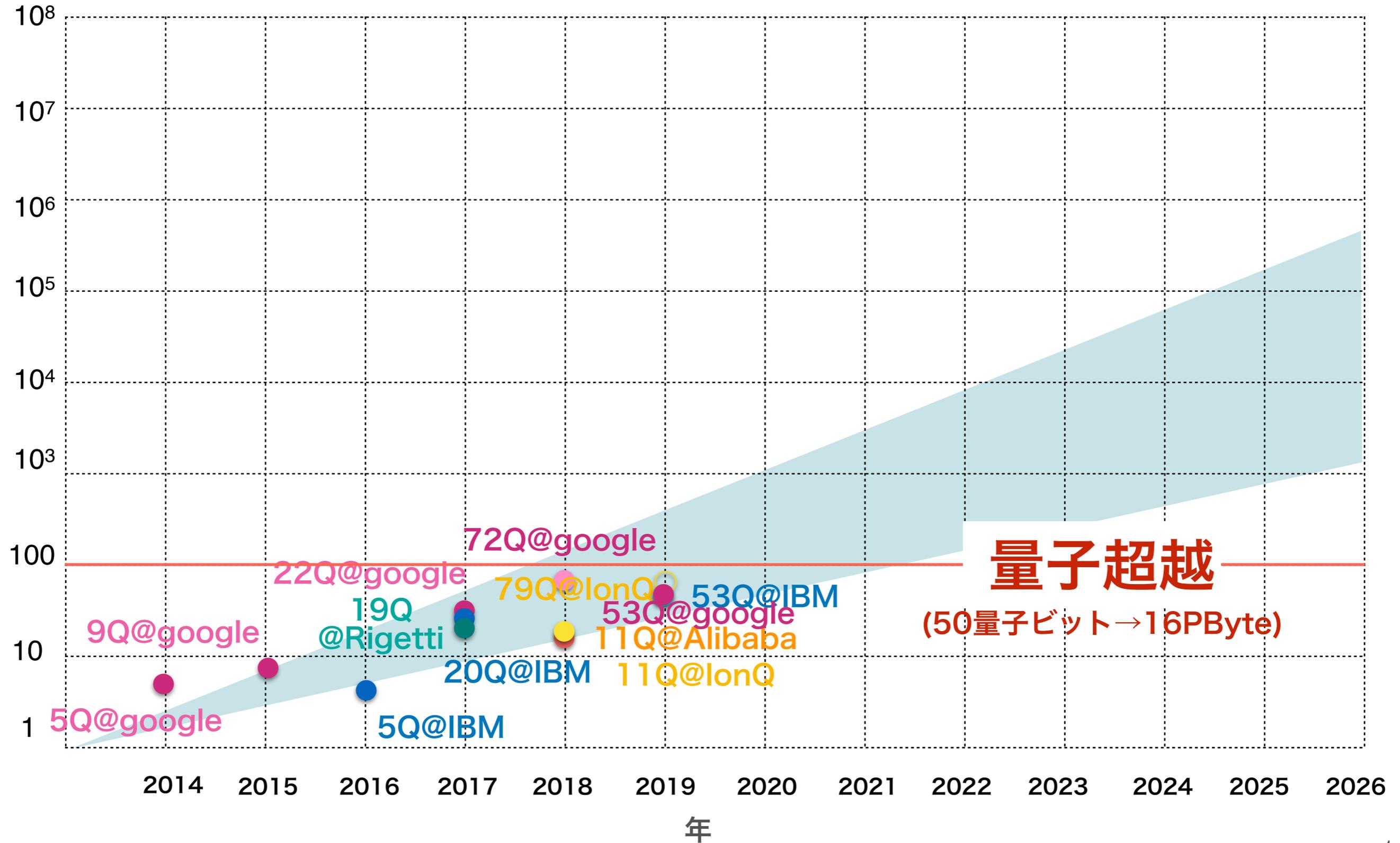
<https://www.nature.com/articles/d41586-019-03213-z>

Google team, Nature **574** 505 (2019)

- プログラム可能な**53量子ビット**を動作させランダム量子回路を実行
- スパコンを用いてその性能を間接的に検証
- 実機**200秒**に対して、スパコンでは**1万年** (**2.5日** by IBMによる反論)
- 53個の量子ビット (**1つの冷凍機**) v.s **4608ノード**(15MW)
- ランダムなビット列をサンプル  
→役にたつ問題ではないが、**量子計算が実際に速いという最初の実証**

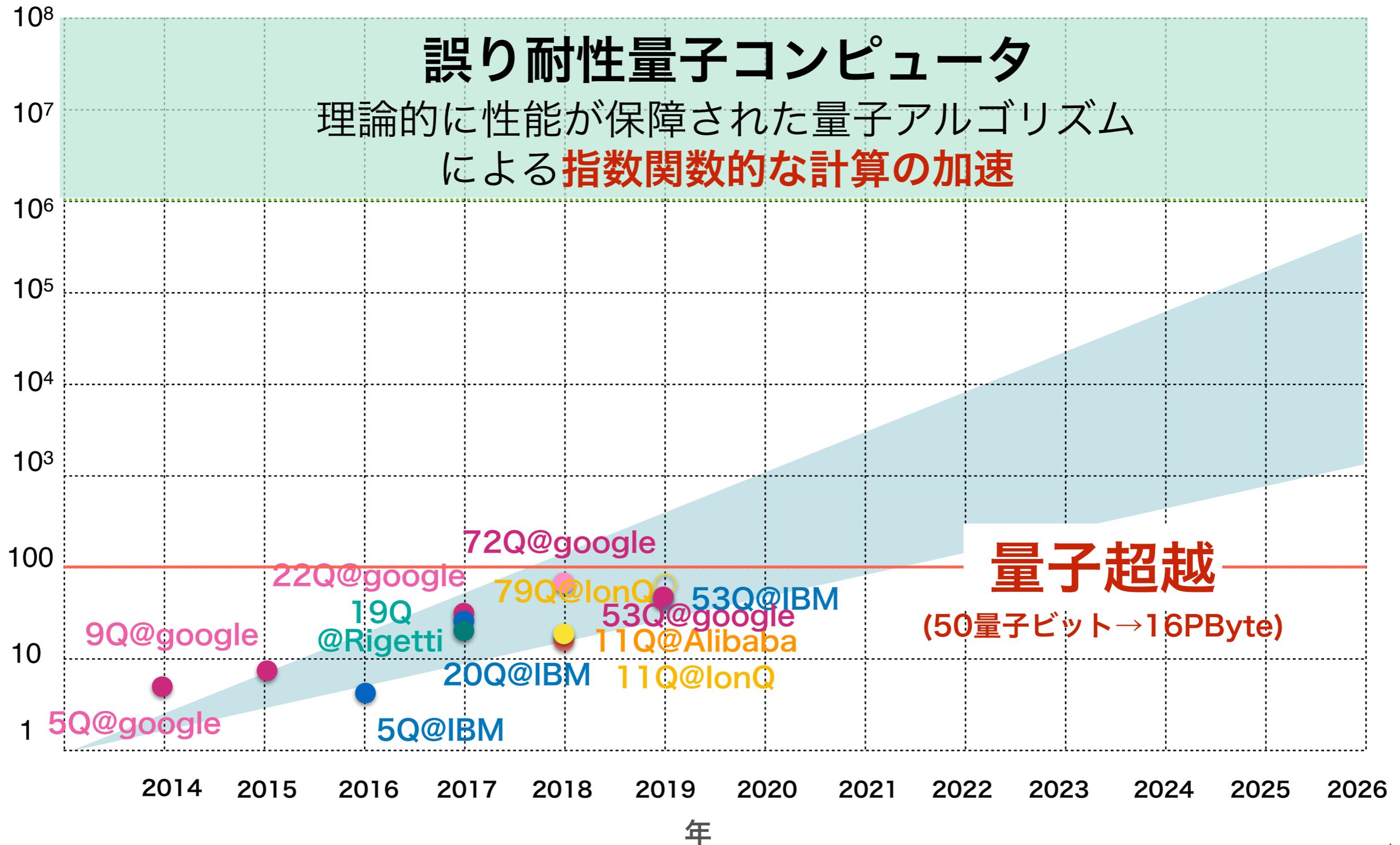
# “量子版”ムーアの法則？

量子ビット数



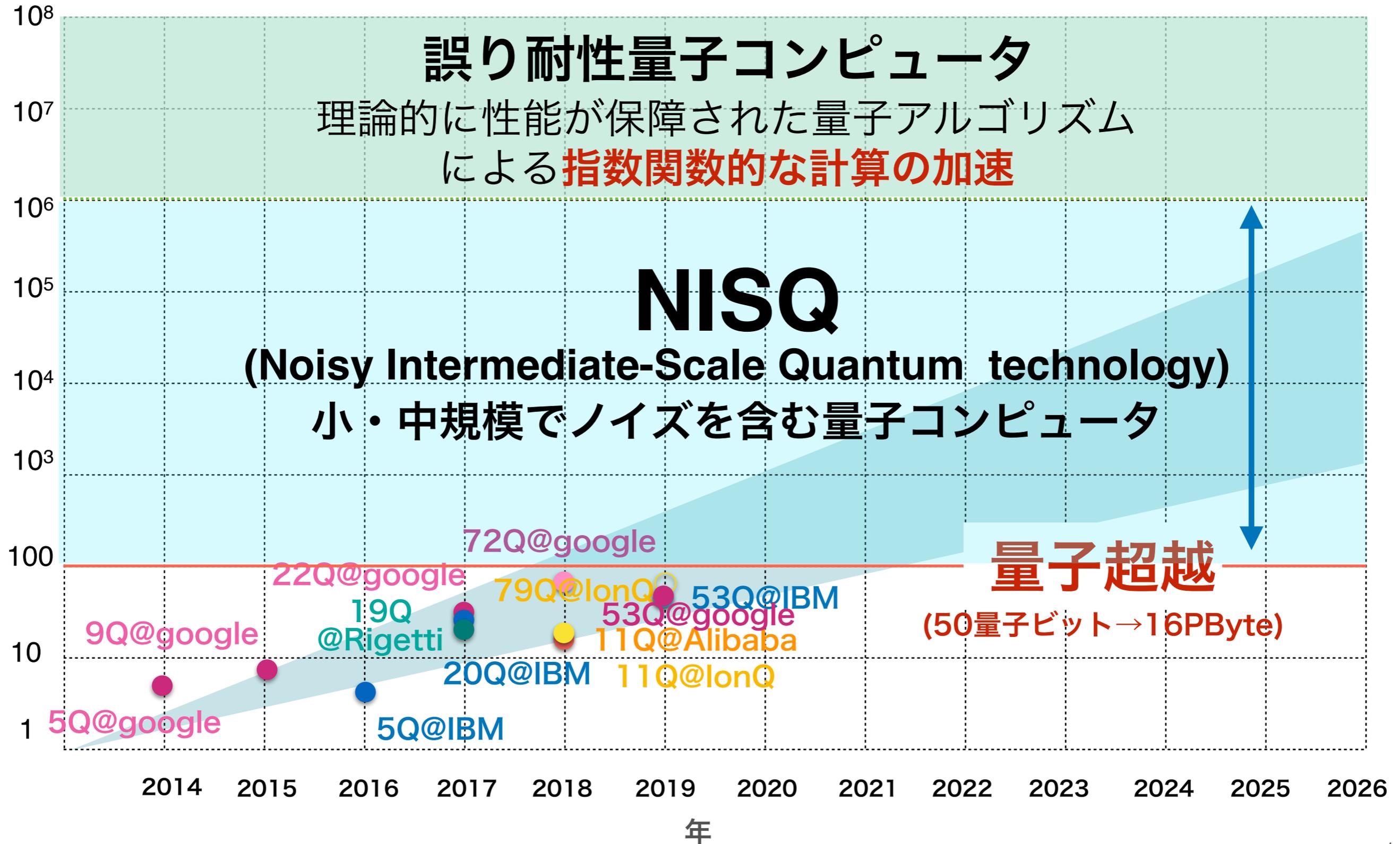
# “量子版”ムーアの法則？

量子ビット数



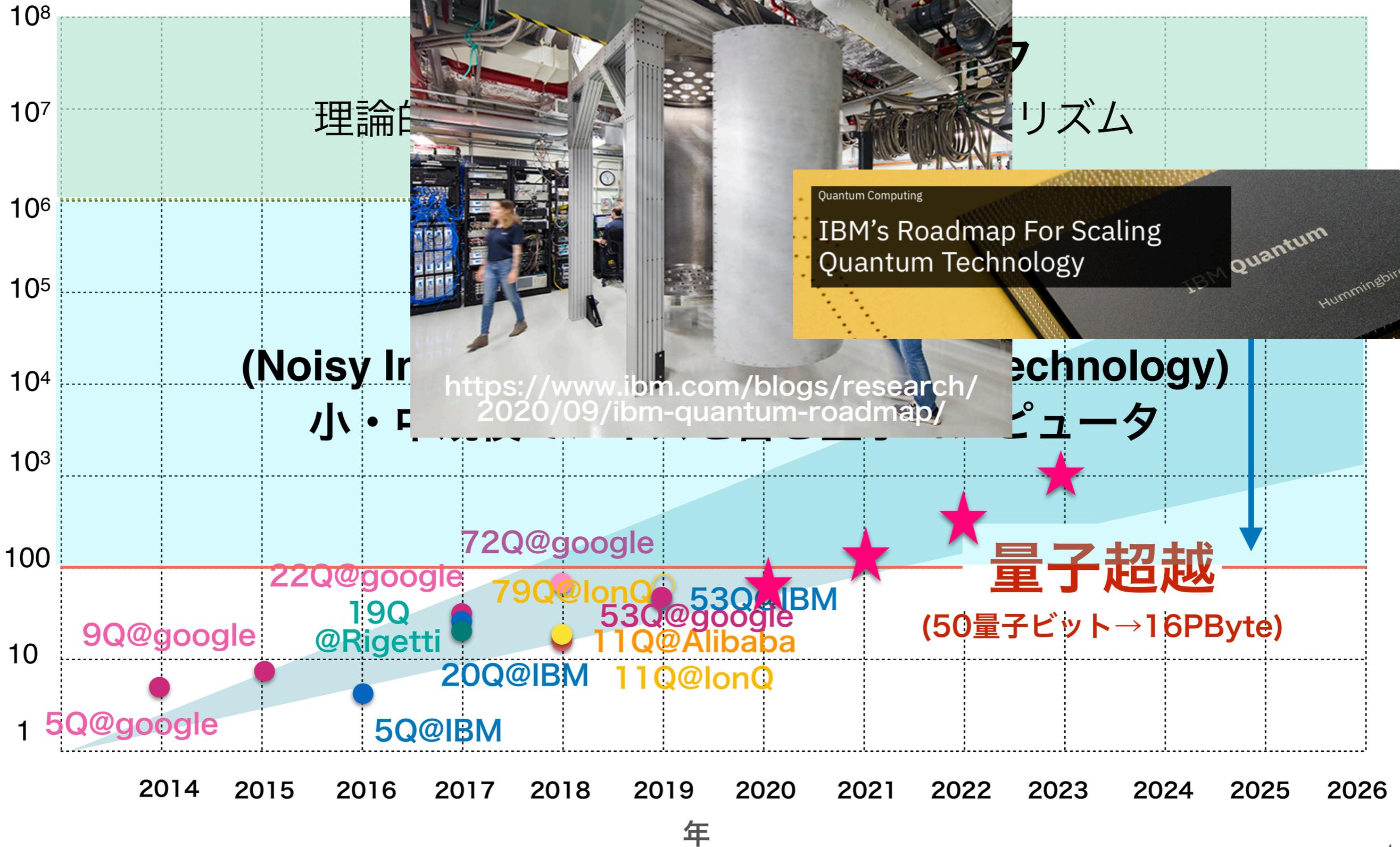
# “量子版”ムーアの法則？

量子ビット数

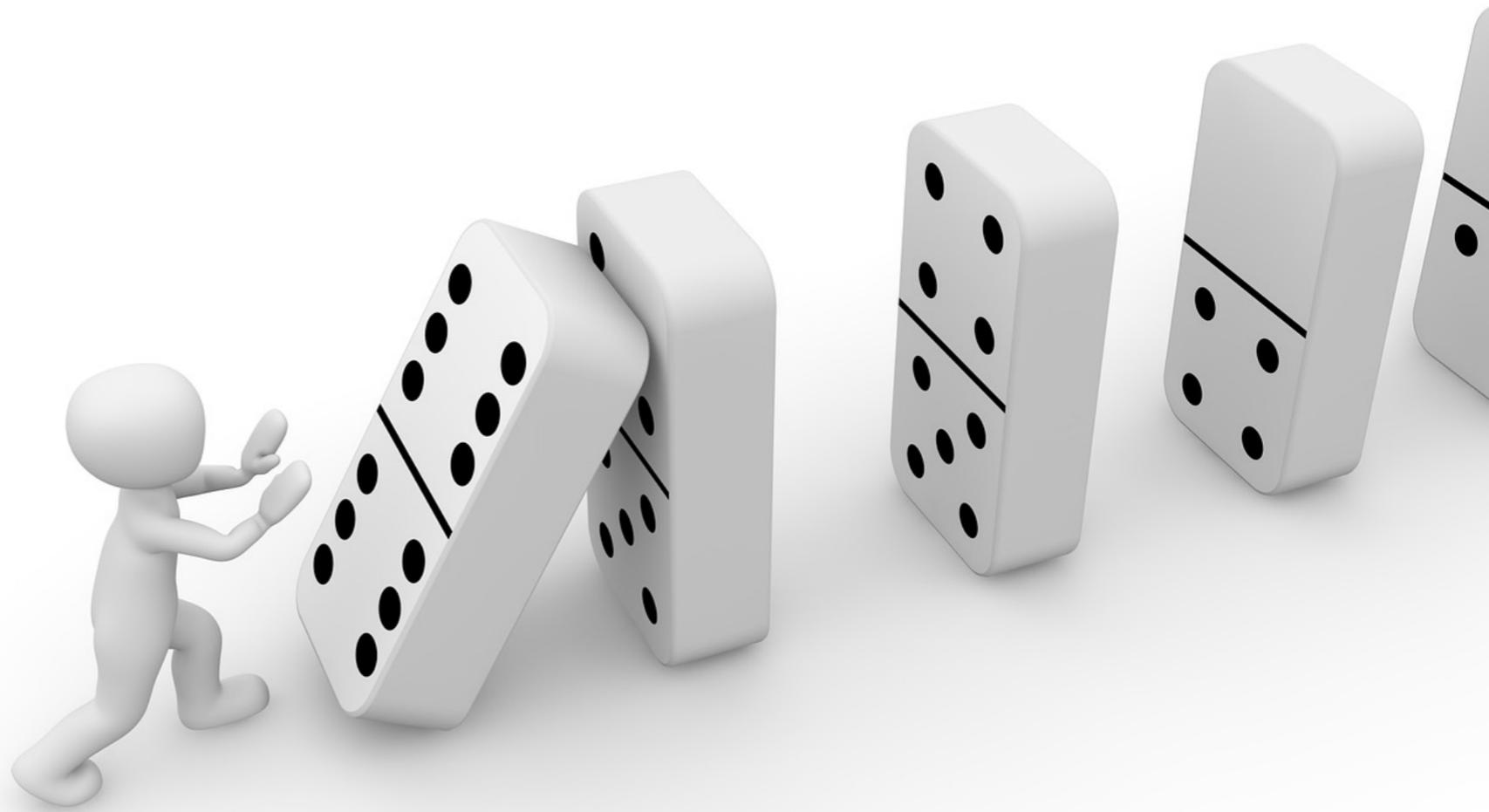


# “量子版”ムーアの法則？

量子ビット数



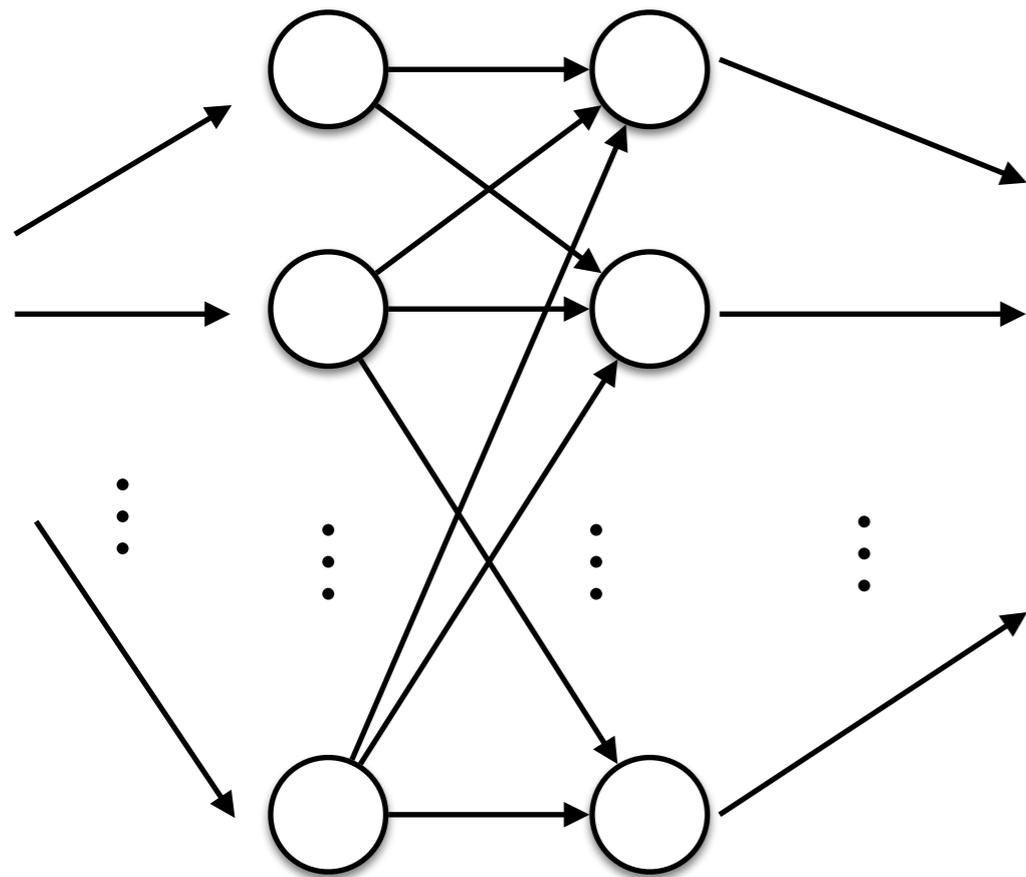
# 量子レザバー計算



# 機械学習

## ニューラルネットワーク(NN)

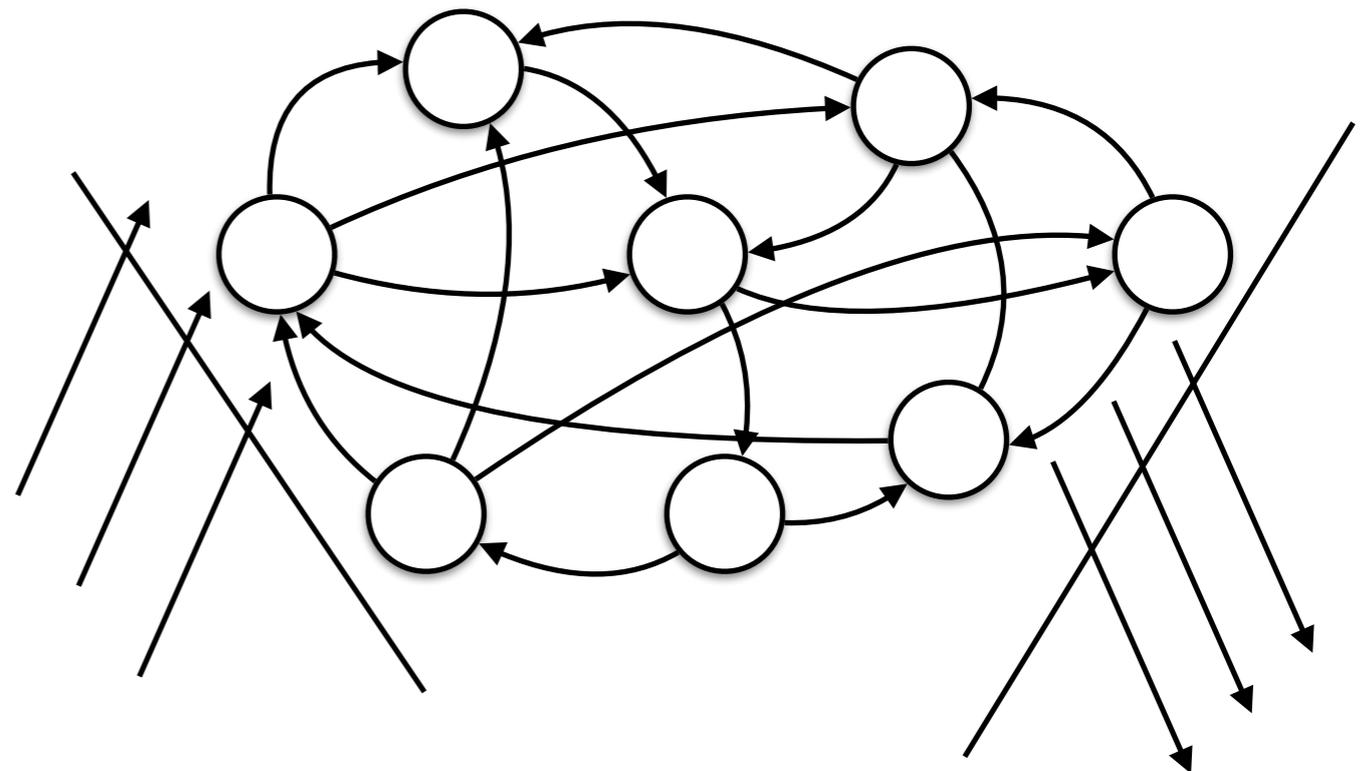
### フィードフォワードNN



履歴が残らない

例) deep learning

### リカレントNN



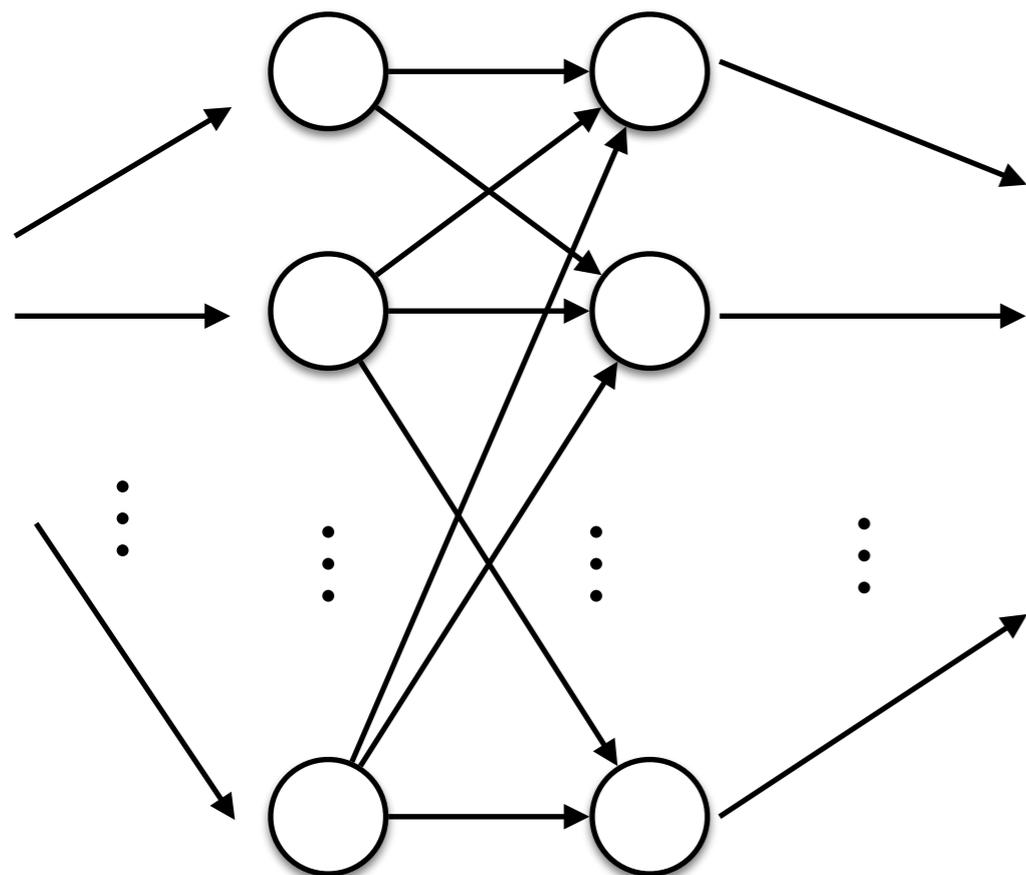
入力情報がネットワークに残る=履歴

例) long short term memory  
echo state network

# 機械学習

## ニューラルネットワーク(NN)

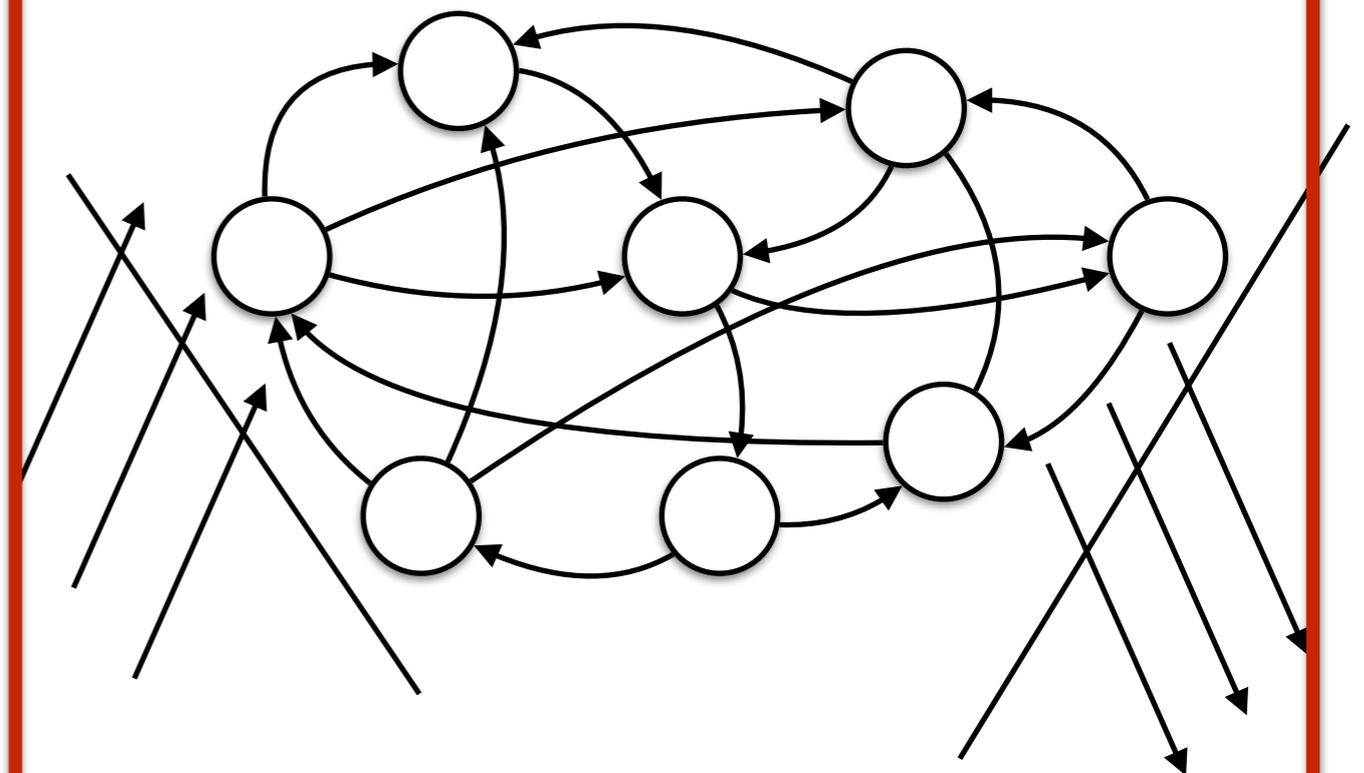
### フィードフォワードNN



履歴が残らない

例) deep learning

### リカレントNN



入力情報がネットワークに残る=履歴

例) long short term memory  
echo state network

# 時系列データ処理

(real time computing)

入力:  $\mathbf{u}(n)$        $n = 1, \dots, T$

出力 (教師データ) :  $\mathbf{y}_{\text{target}}(n)$

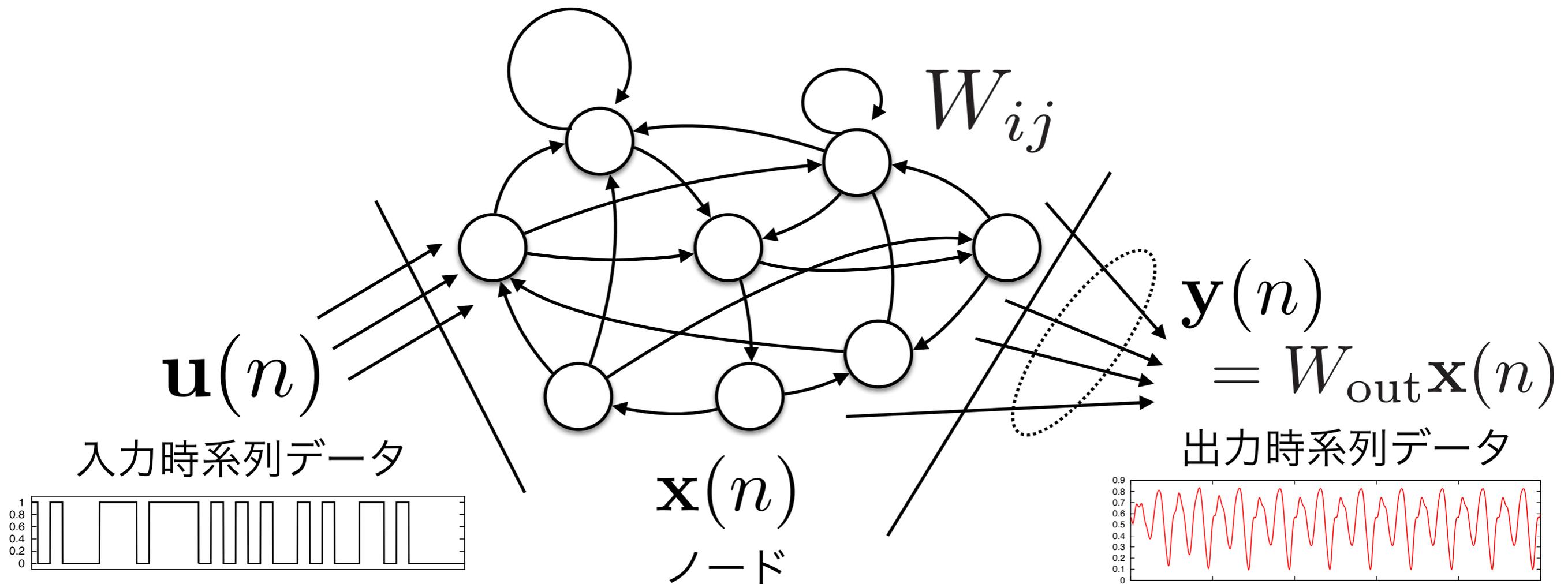
問題 : 誤差  $E(\mathbf{y}, \mathbf{y}_{\text{target}})$  を最小にするような

$$\mathbf{y}(n) = f(\mathbf{u}(n), \mathbf{u}(n-1), \dots, \mathbf{u}(1))$$

非線形関数(フィルター),  $f(\dots)$ を見つける.

例) ノイズ除去, 音声・動画認識, 株価予測, ロボット制御 *etc*

# Recurrent Neural Network



ノードの時間発展

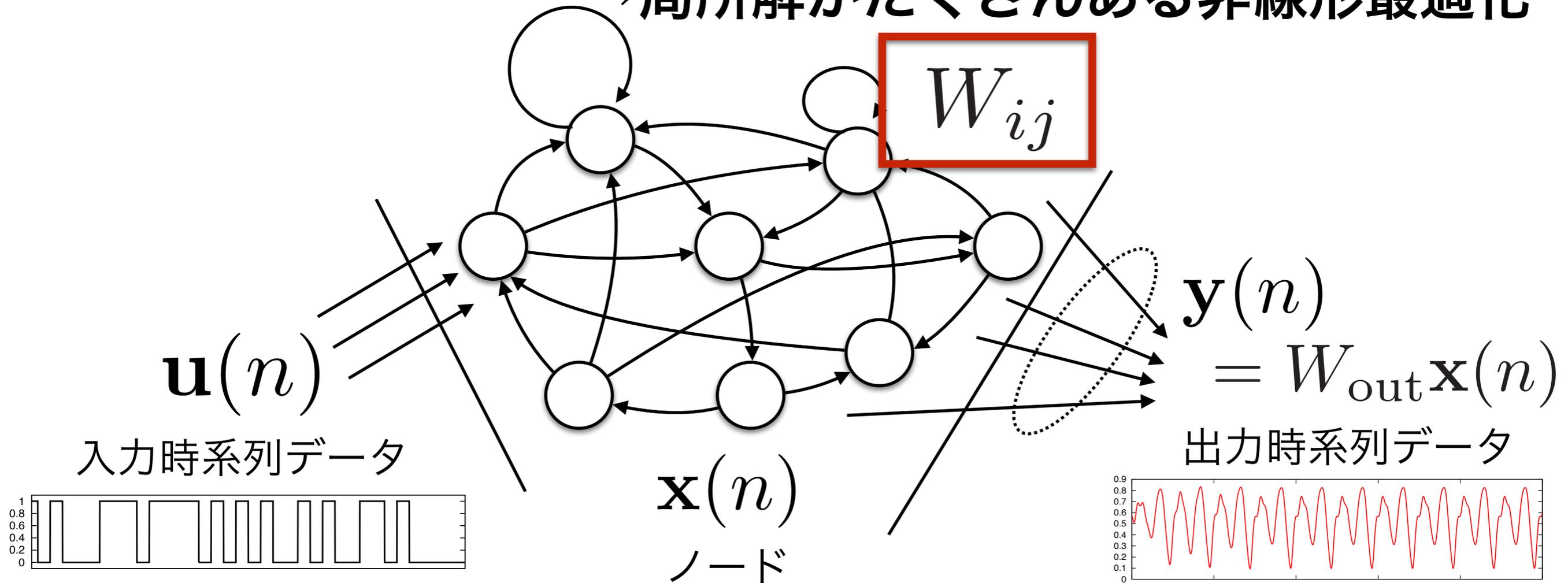
$$\mathbf{x}(n) = f(\mathbf{W}_{in} \mathbf{u}(n) + \mathbf{W} \mathbf{x}(n - 1))$$

シグモイド関数 (非線形性)

# Recurrent Neural Network

ノード間の重み $W$ を最適化する (誤差逆伝搬法)

→ 局所解がたくさんある非線形最適化



入力時系列データ

$\mathbf{x}(n)$   
ノード

出力時系列データ

ノードの時間発展

$$\mathbf{x}(n) = f(\mathbf{W}_{in} \mathbf{u}(n) + \mathbf{W} \mathbf{x}(n - 1))$$

シグモイド関数 (非線形性)

# レザバコンピューティング

(*Echo state network, Liquid state machine*)



H. Jaeger et al.,  
Science, **304**, 78  
(2004).

W. Maass et al.,  
Neural Comput **14**,  
2531 (2002).



<http://www.igi.tugraz.at/maass/>

## reservoir computing

D. Verstraeten et al., Neural Netw. **20**, 391 (2007).

## Echo state network

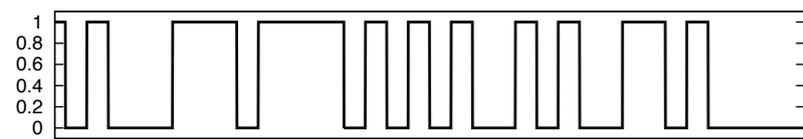
では、重みは乱数

→ ノード数を莫大に増やすことができる

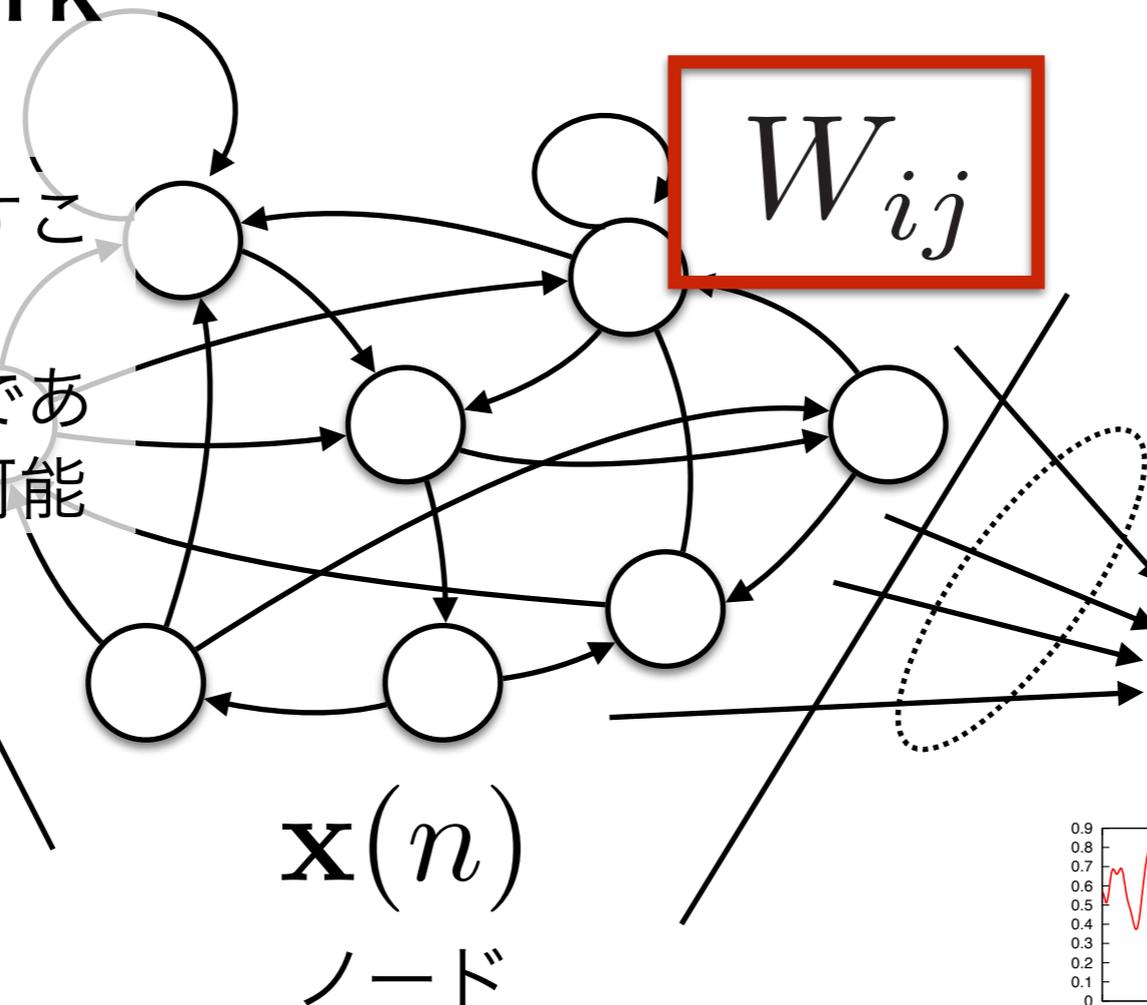
→ 大自由度非線形力学系であればレザバーとして利用可能

$\mathbf{u}(n)$

入力時系列データ

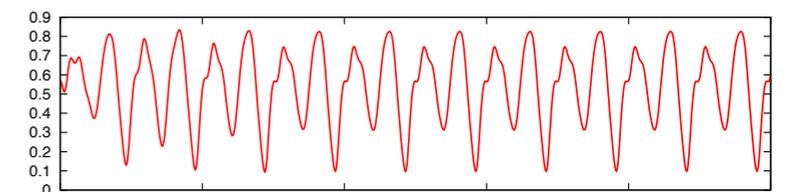


ノードの時間発展



$$\mathbf{y}(n) = \mathbf{W}_{\text{out}} \mathbf{x}(n)$$

出力時系列データ



# レザバコンピューティング

(Echo state network, Liquid state machine)



H. Jaeger et al.,  
Science, **304**, 78  
(2004).

W. Maass et al.,  
Neural Comput **14**,  
2531 (2002).



<http://www.igi.tugraz.at/maass/>

## reservoir computing

D. Verstraeten et al., Neural Netw. **20**, 391 (2007).

## Echo state network

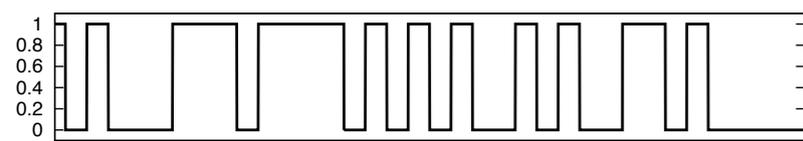
では、重みは乱数

→ ノード数を莫大に増やすことができる

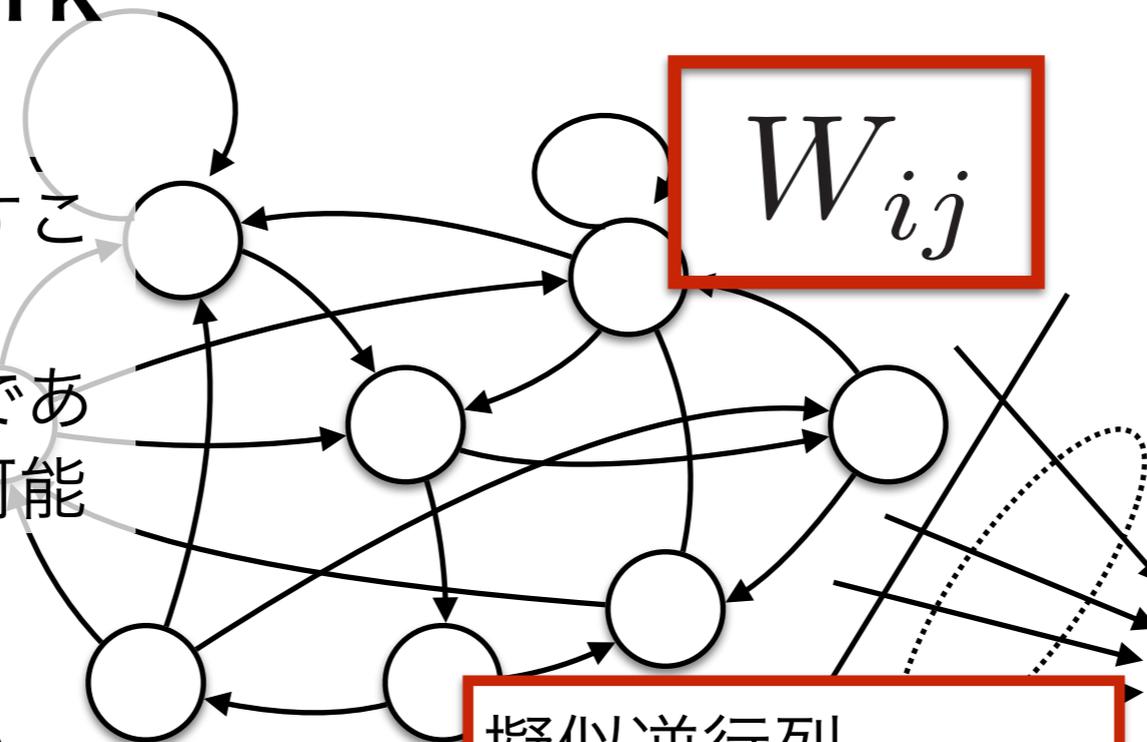
→ 大自由度非線形力学系であればレザバとして利用可能

$\mathbf{u}(n)$

入力時系列データ



ノードの時間発展



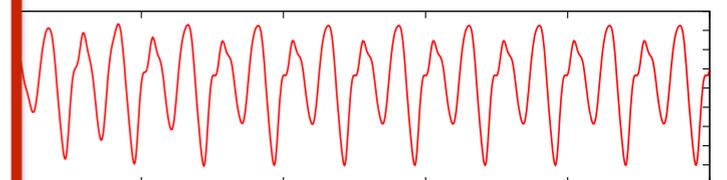
擬似逆行列  
(Moore-Penrose)

$W_{out}$

教師データとの  
誤差が最小

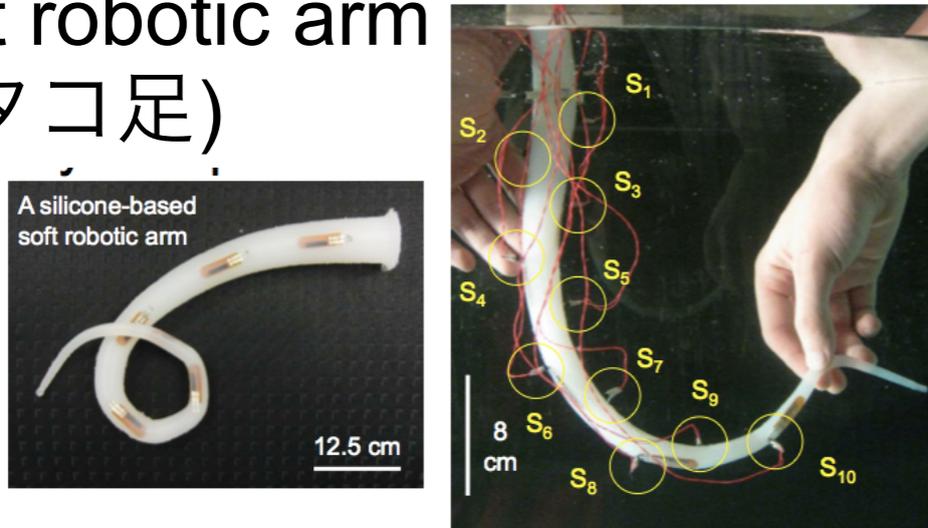
$$\mathbf{y}(n) = W_{out} \mathbf{x}(n)$$

出力時系列データ



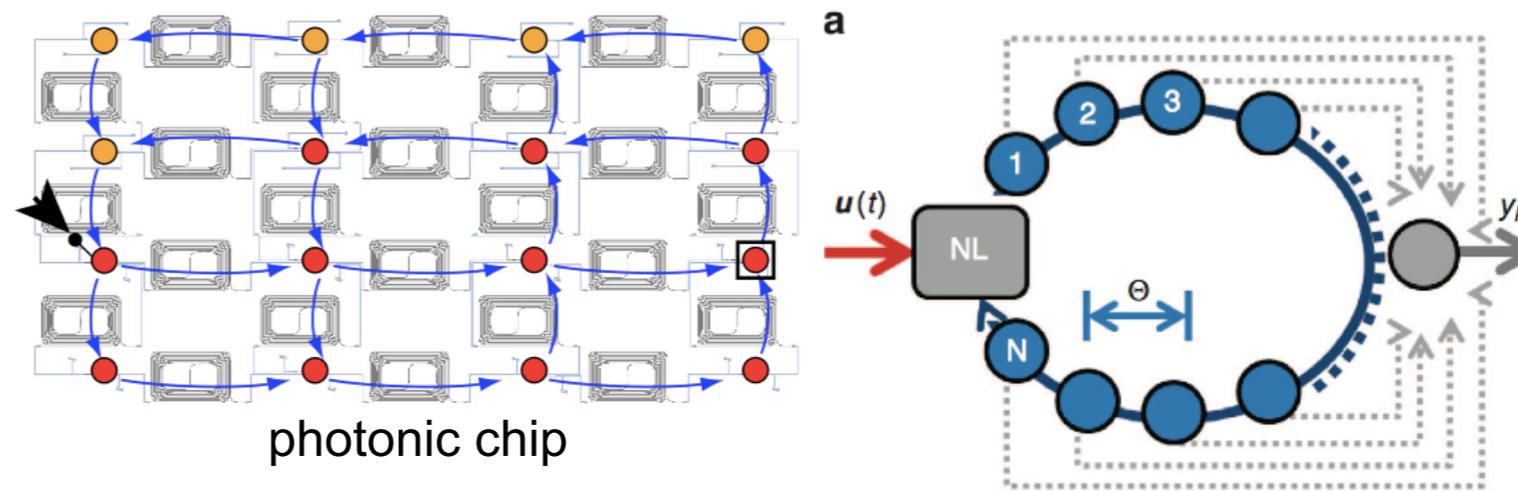
# フィジカルレザバコンピューティング physical reservoir computing

- Soft robotic arm (タコ足)



K. Nakajima et al., *Front. Comput. Neurosci.* **7**, 1 (2013).  
 K. Nakajima et al., *J. R. Soc. Interface* **11**: 20140437, (2014).  
 K. Nakajima et al., *Sci. Rep.* **5**, 10487 (2015).

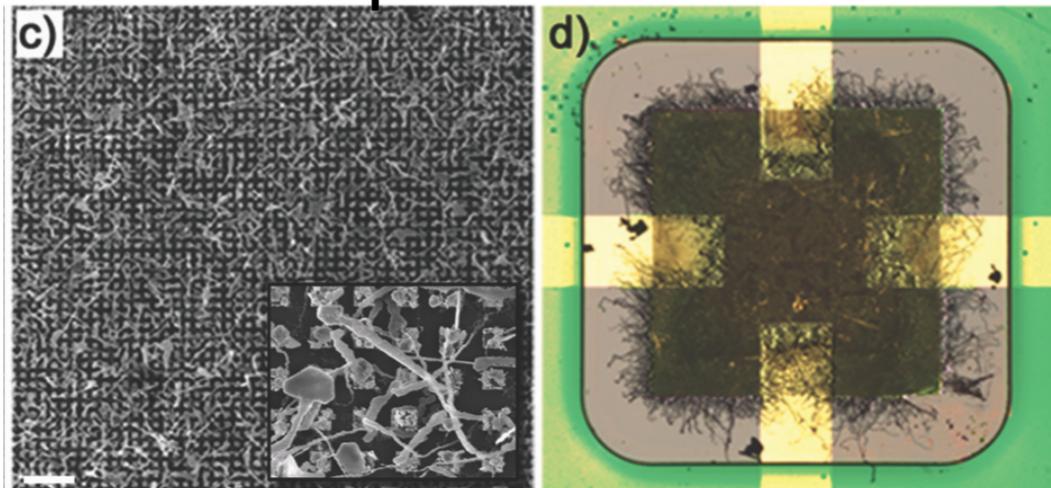
- Photonic reservoir computing



H. J. Caulfield, et al., *Nature Photon.* **4**, 261-263 (2010).  
 L. Appeltant et al. *Nature Commun.* **2**, 468 (2011).  
 Y. Paquot et al., *Sci. Rep.* **2**, 287 (2012).  
 D. Woods, et al. *Nature Phys.* **8**, 257 (2012).  
 R. Martinenghi et al., *PRL* **108**, 244101 (2012).  
 K. Vandoorne et al., *Nature Commun.* **5**, 3541 (2014).  
 and so on....

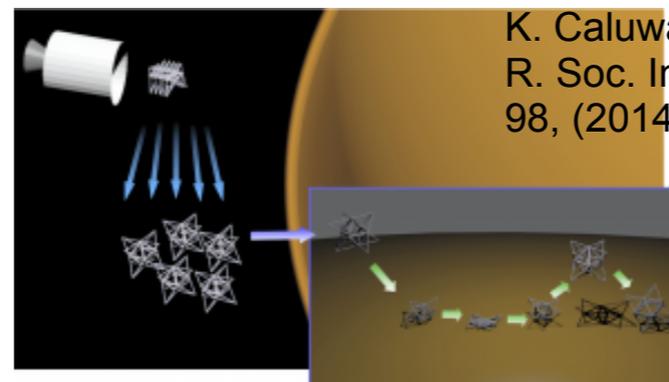
<https://en.wikipedia.org/wiki/Tensegrity>

- Neuromorphic atomic switch network

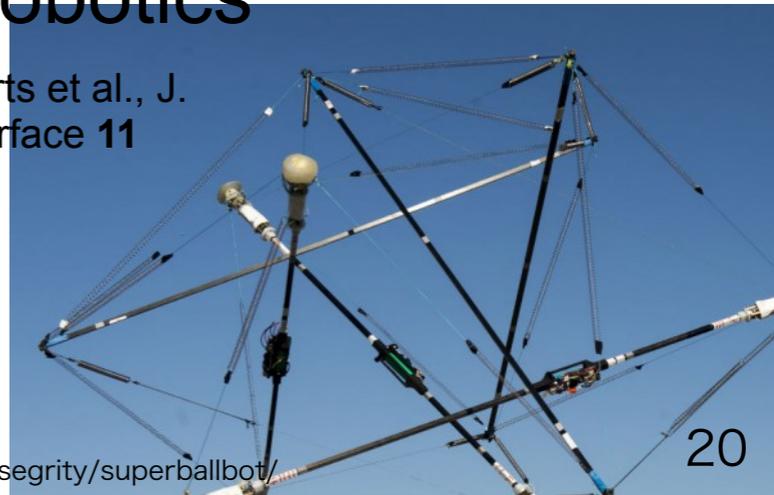


A.Z. Stieg, *Adv. Mater.* **24**, 286 (2012)  
 H.O. Sillin et al., *Nanotechnology* **24**, 384004 (2013).  
 E.C. Demis et al., *Nanotechnology* **26**, 204003 (2015).

- NASA's tensegrity robotics



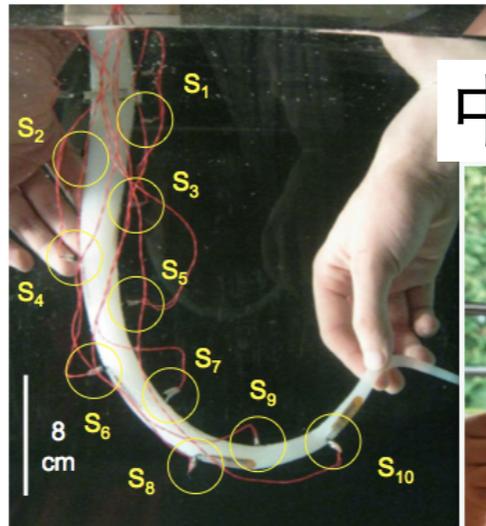
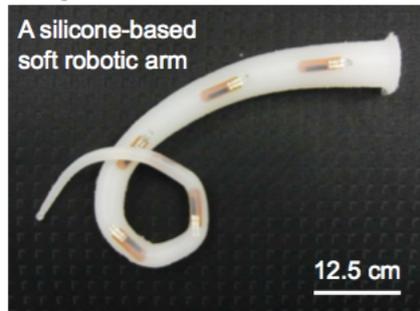
K. Caluwaerts et al., *J. R. Soc. Interface* **11**, 98, (2014).



<http://ti.arc.nasa.gov/tech/asr/intelligent-robotics/tensegrity/superballbot/>

# フィジカルレザバコンピューティング physical reservoir computing

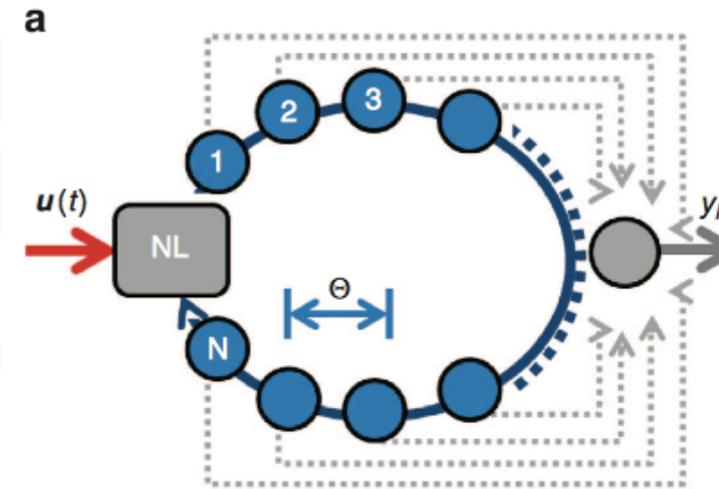
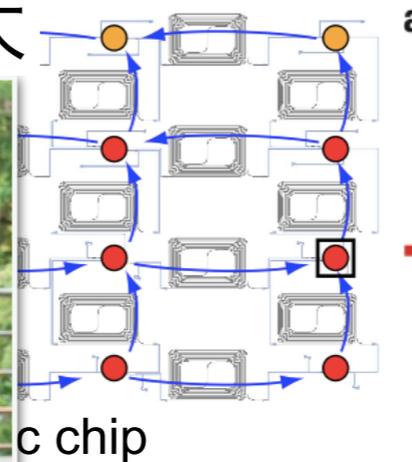
## Soft robotic arm (タコ足)



中嶋さん@東大



## Photonic reservoir computing

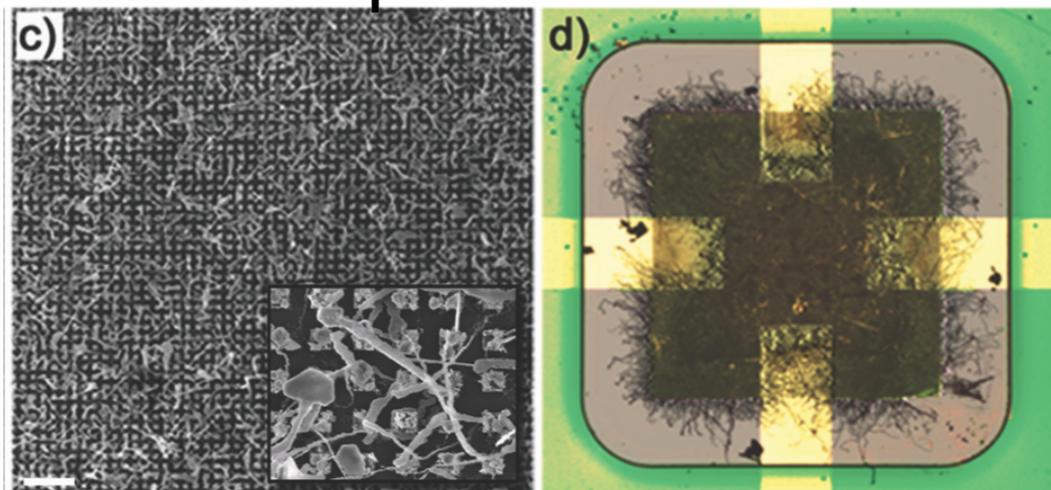


K. Nakajima et al., *Front. Comput. Neurosci.* **7**, 1 (2013).  
 K. Nakajima et al., *J. R. Soc. Interface* **11**: 20140437, (2014).  
 K. Nakajima et al., *Sci. Rep.* **5**, 10487 (2015).

Y. Paquot et al., *Nature Photon.* **4**, 261-263 (2010).  
 et al. *Nature Commun.* **2**, 468 (2011).  
 Y. Paquot et al., *Sci. Rep.* **2**, 287 (2012).  
 D. Woods, et al. *Nature Phys.* **8**, 257 (2012).  
 R. Martinenghi et al., *PRL* **108**, 244101 (2012).  
 K. Vandoorne et al., *Nature Commun.* **5**, 3541 (2014).  
 and so on....

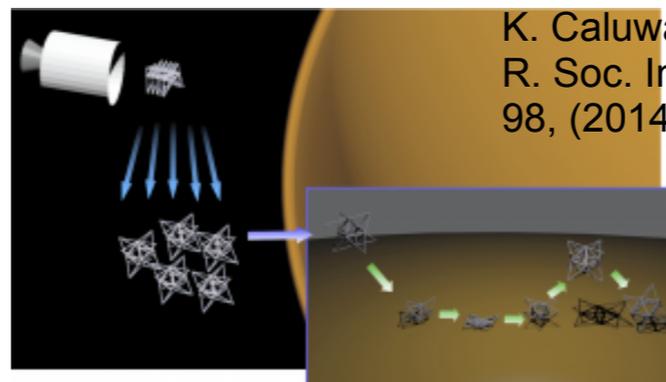
<https://en.wikipedia.org/wiki/Tensegrity>

## Neuromorphic atomic switch network

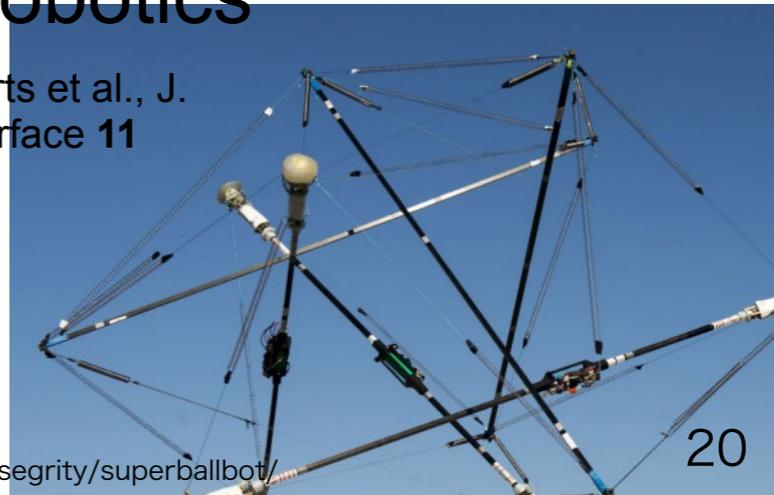


A.Z. Stieg, *Adv. Mater.* **24**, 286 (2012)  
 H.O. Sillin et al., *Nanotechnology* **24**, 384004 (2013).  
 E.C. Demis et al., *Nanotechnology* **26**, 204003 (2015).

## NASA's tensegrity robotics



K. Caluwaerts et al., *J. R. Soc. Interface* **11**, 140198 (2014).

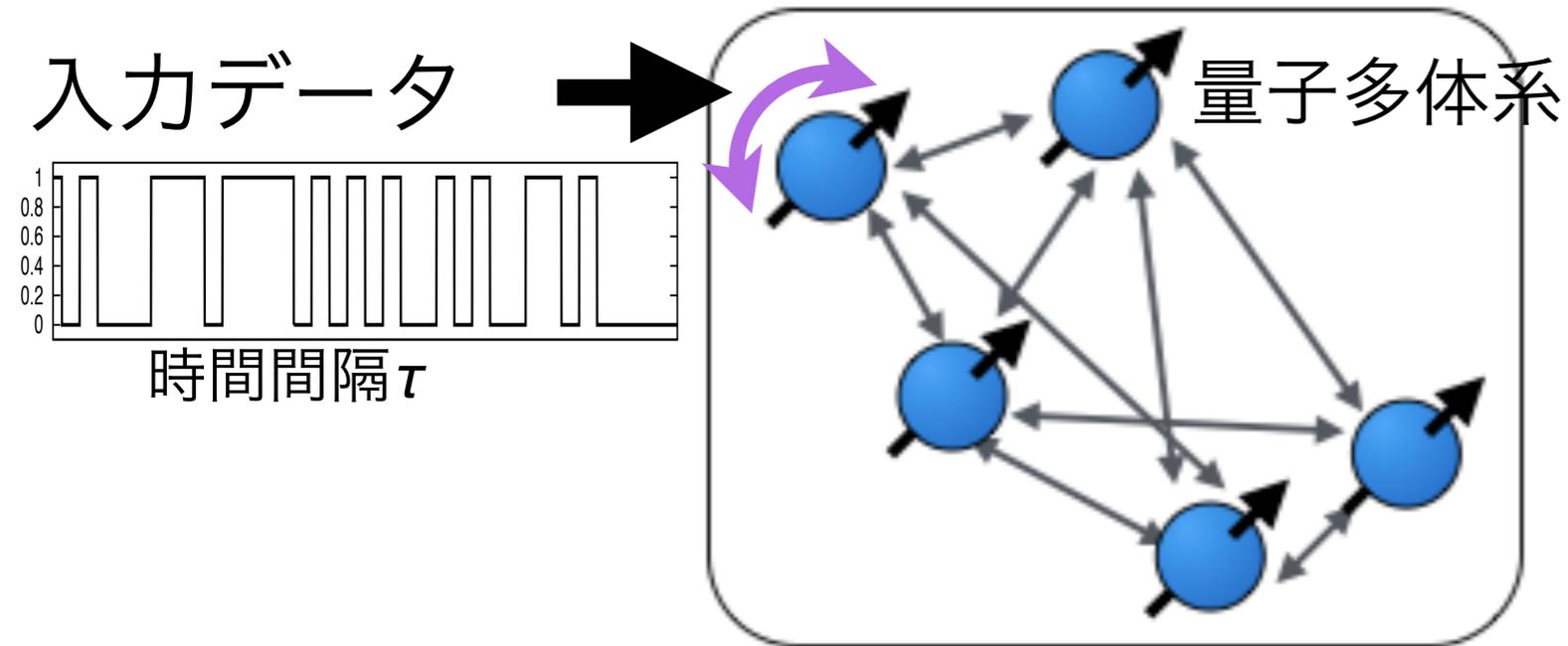


<http://ti.arc.nasa.gov/tech/asr/intelligent-robotics/tensegrity/superballbot/>

# 量子リザバコンピュテーティング

## quantum reservoir computing

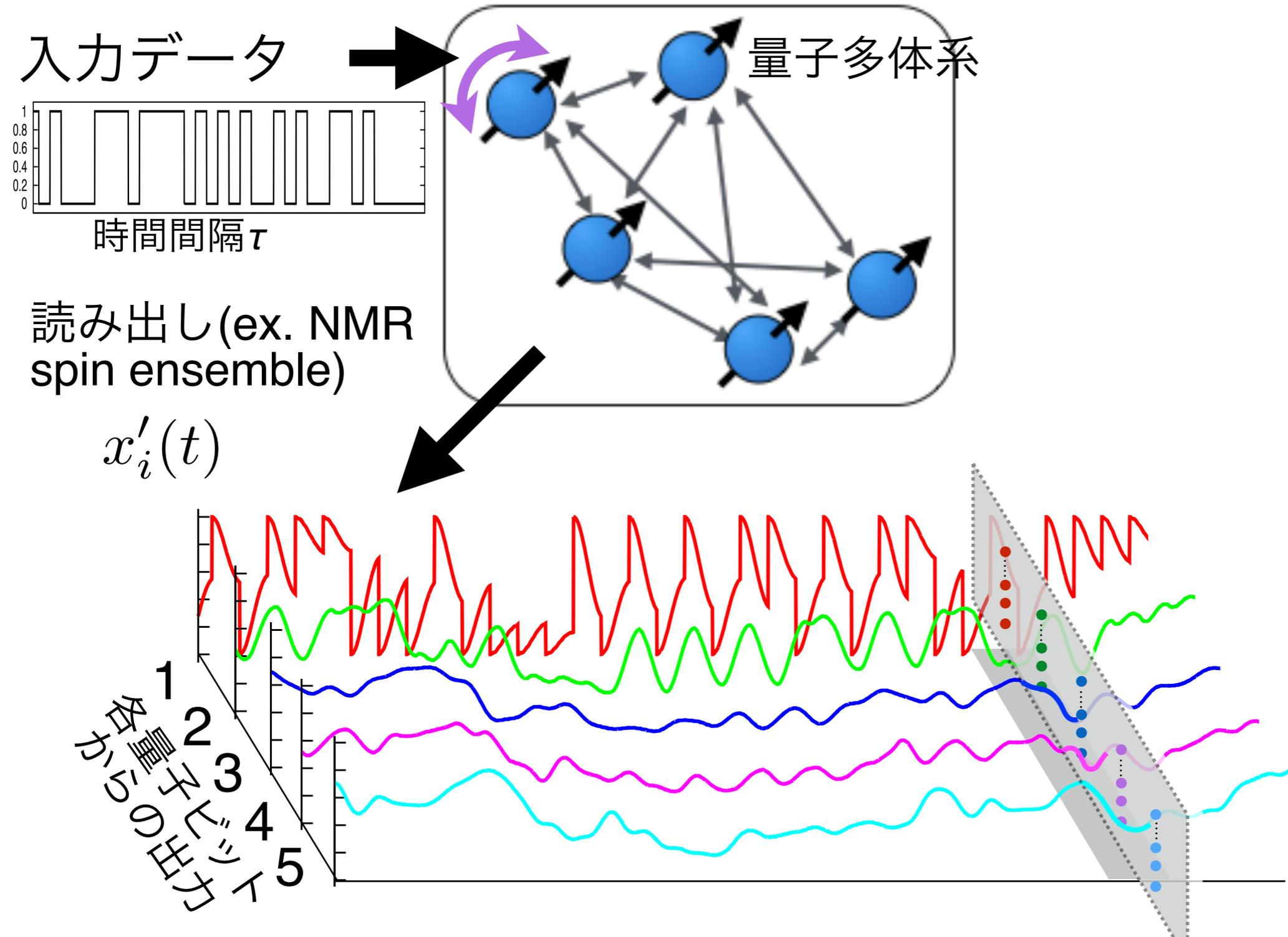
*“Harnessing Disordered-Ensemble Quantum Dynamics for Machine Learning”*,  
K. Fujii, K. Nakajima, Phys. Rev. Applied 8, 024030 (2017).



# 量子リザーバコンピューティング

## quantum reservoir computing

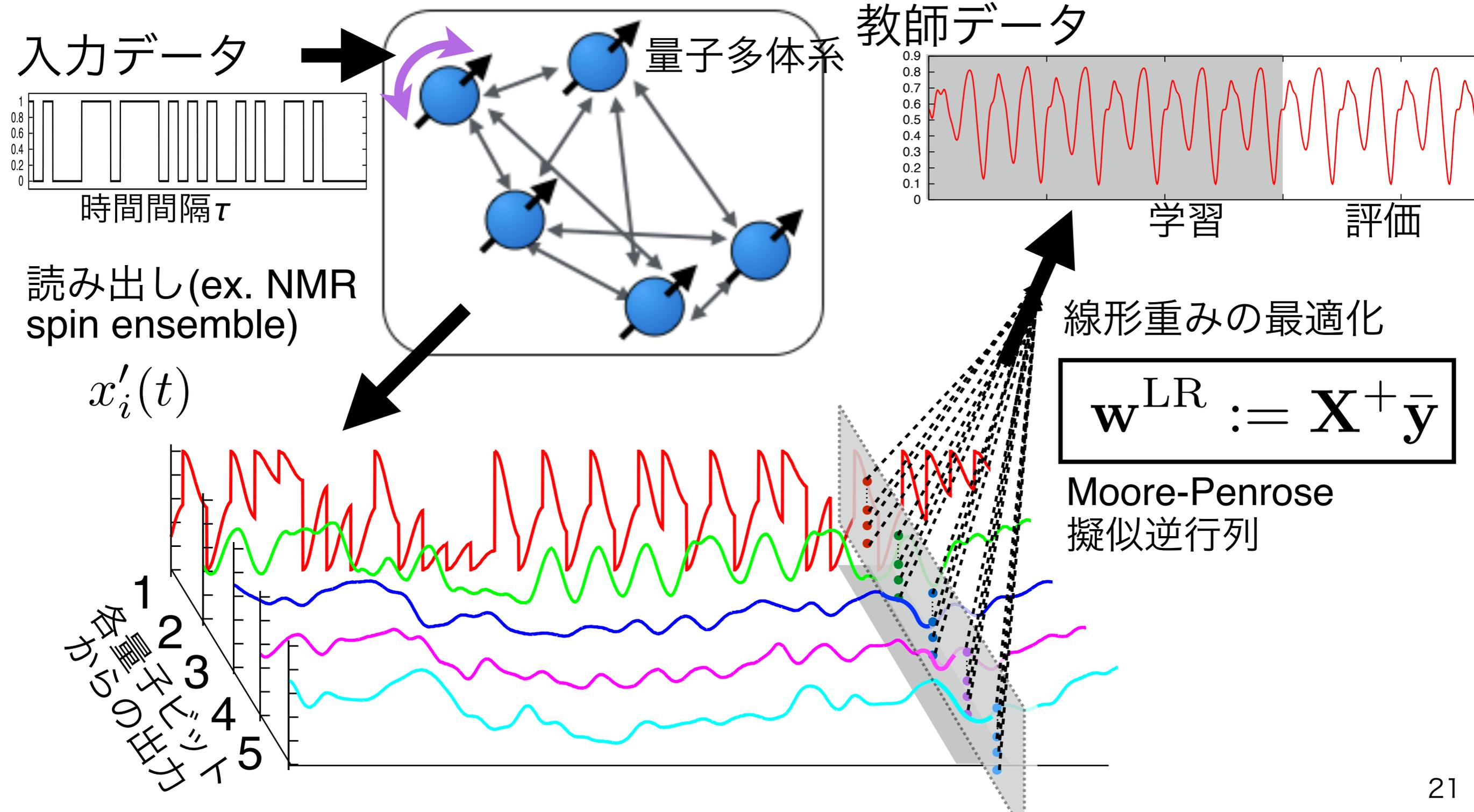
“*Harnessing Disordered-Ensemble Quantum Dynamics for Machine Learning*”,  
K. Fujii, K. Nakajima, Phys. Rev. Applied 8, 024030 (2017).



# 量子リザバコンピュテーティング

## quantum reservoir computing

“*Harnessing Disordered-Ensemble Quantum Dynamics for Machine Learning*”,  
 K. Fujii, K. Nakajima, Phys. Rev. Applied 8, 024030 (2017).



# 量子リザバコンピュテーティング

## quantum reservoir computing

### QR

### RNN (echo state network)

状態  
(ノード)

$$\mathbf{x} \quad (x_i = \text{Tr}[P_i \rho])$$

粒子数に対して指数的

$\mathbf{x}$

入力

$$W_{in}^{(u_k)}$$

写像が入力値に依存 → 非線形性  
(テンソル積構造)

$$+ W_{in} u_k$$

状態に加える

時間発展

$W$

(ユニタリー性 = at the edge of chaos)

$W$

最大特異値で調整

$\tanh$  etc

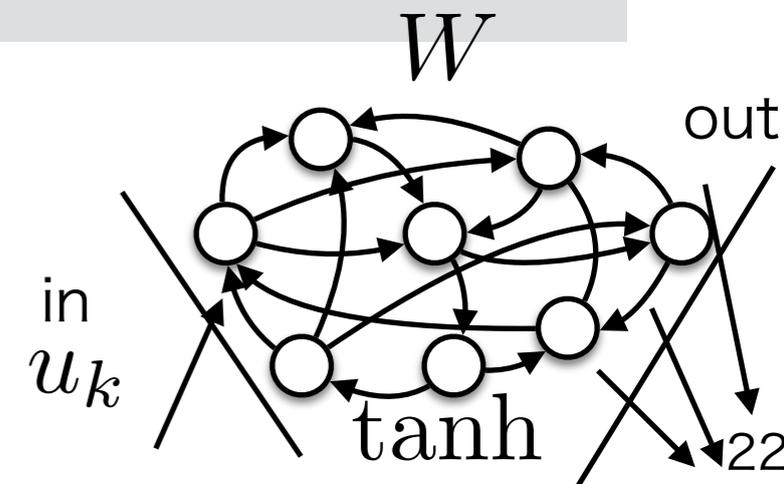
非線形性

出力

$$\bar{x}_i = \langle Z_i \rangle$$

各粒子の観測のアンサンブル期待値  
→ 多項式個

$\mathbf{x}$



# 量子リザバコンピュテーティング

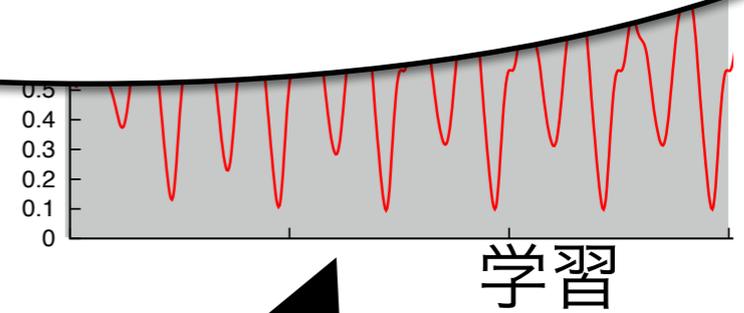
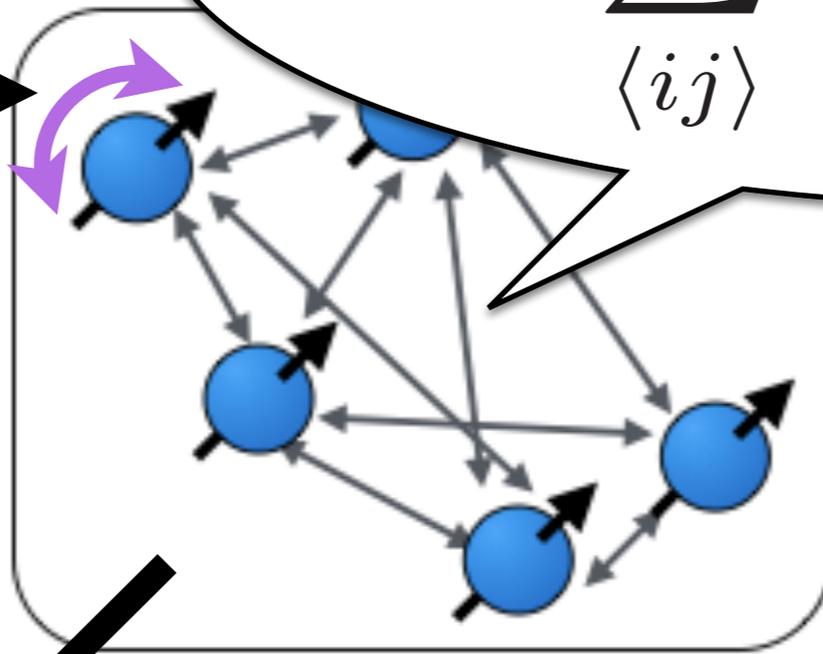
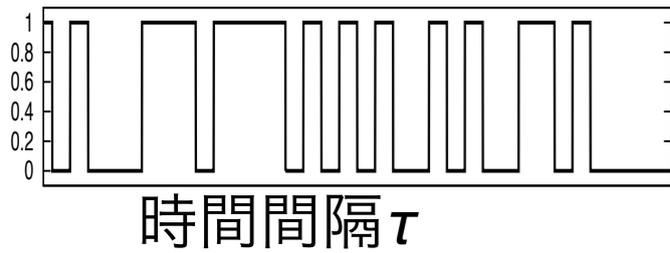
quantum

“Harnessing Disorder”  
K. Fujii, K. Nakajima

ランダム結合, 横磁場イジング

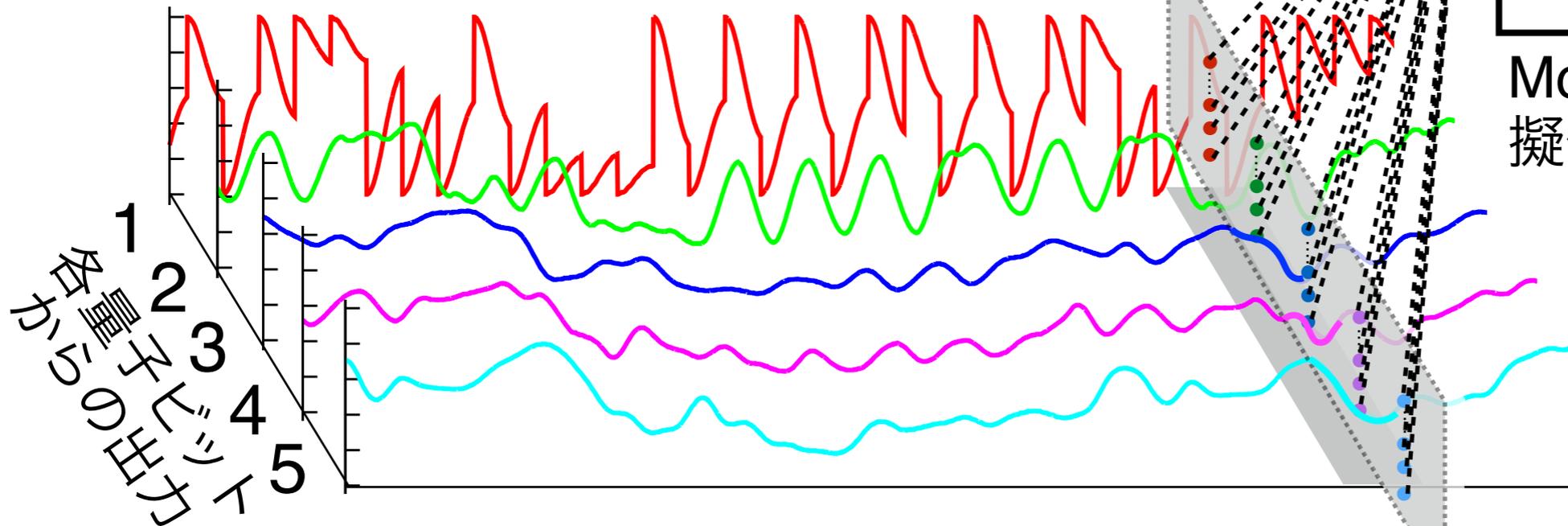
$$H = \sum_{\langle ij \rangle} J_{ij} Z_i Z_j + \sum_i h_i X_i$$

入力データ



読み出し (ex. NMR spin ensemble)

$x'_i(t)$



線形重みの最適化

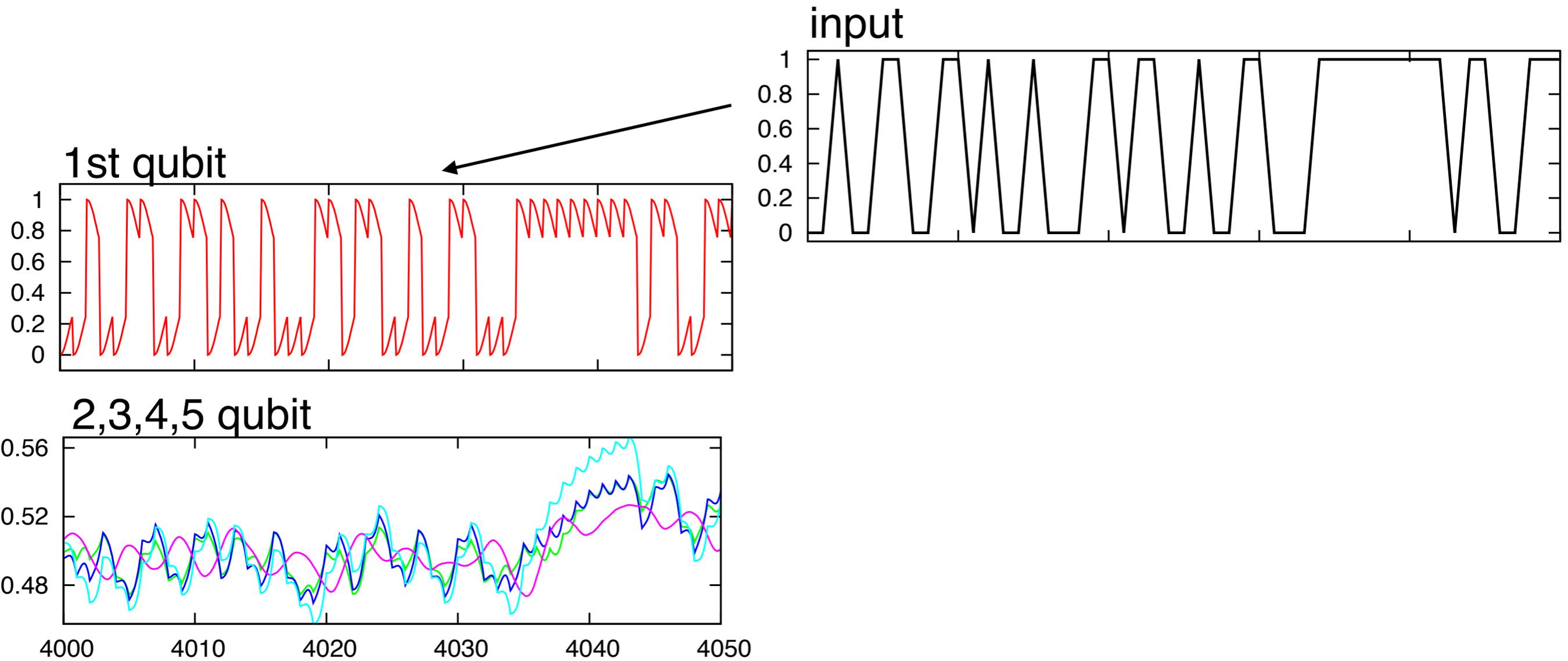
$$\mathbf{w}^{\text{LR}} := \mathbf{X}^+ \bar{\mathbf{y}}$$

Moore-Penrose  
擬似逆行列



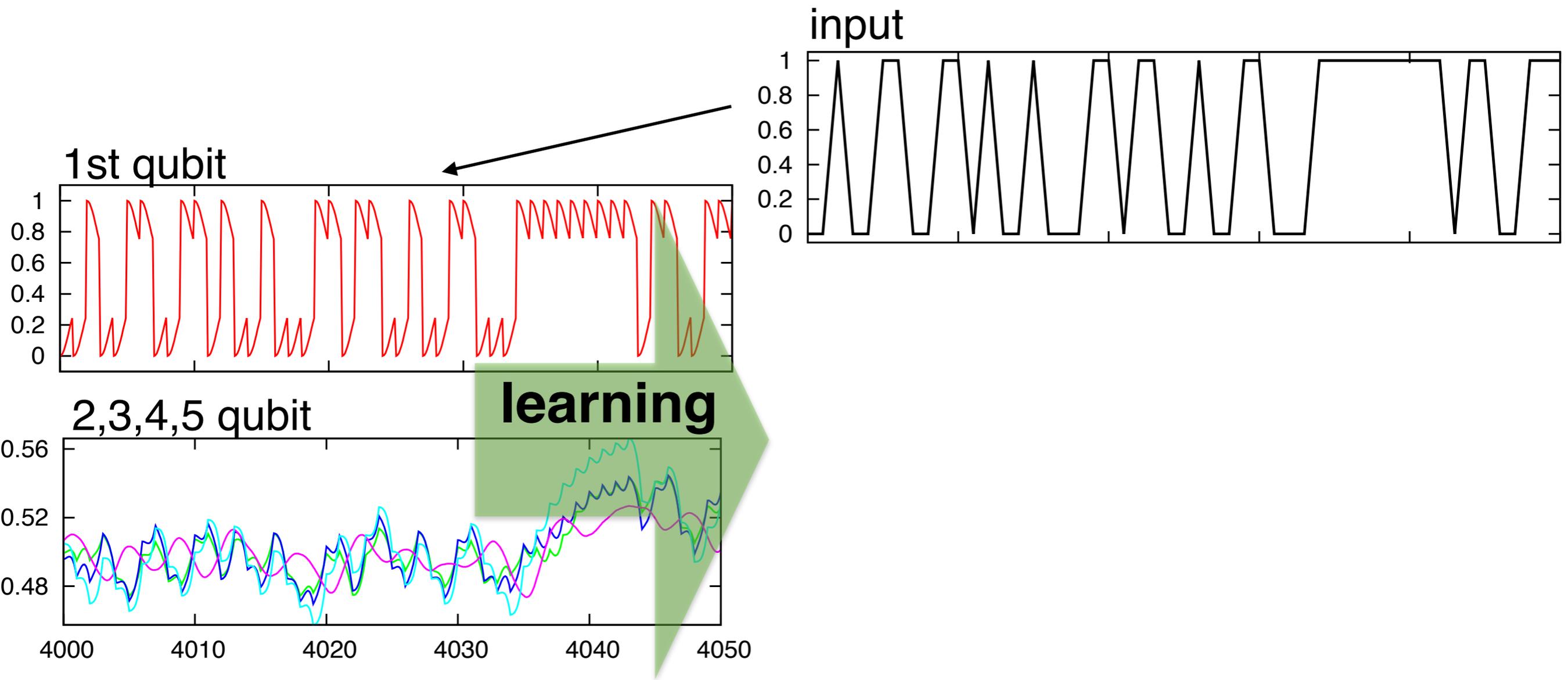
# 短期記憶

## short term memory



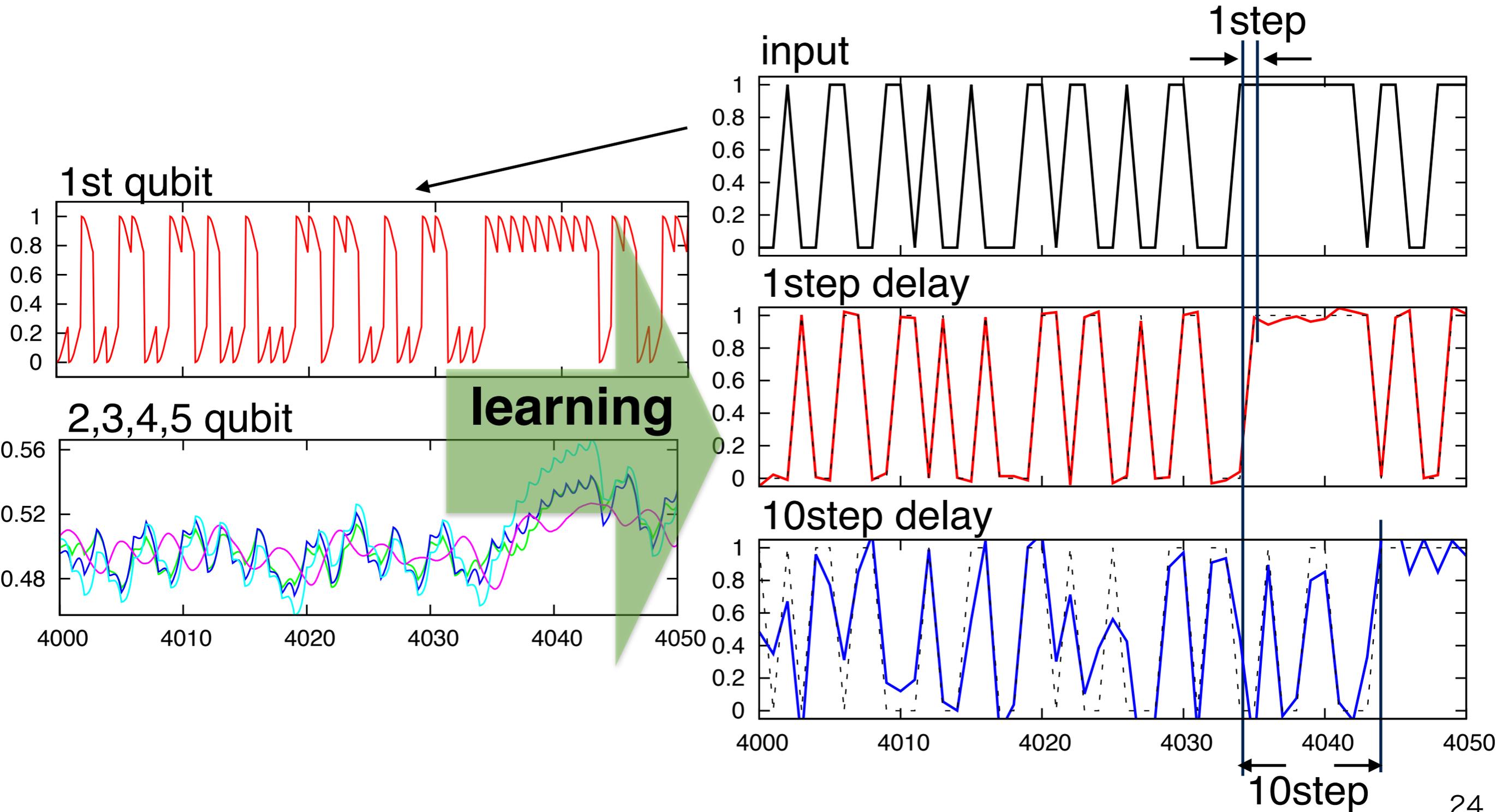
# 短期記憶

## short term memory



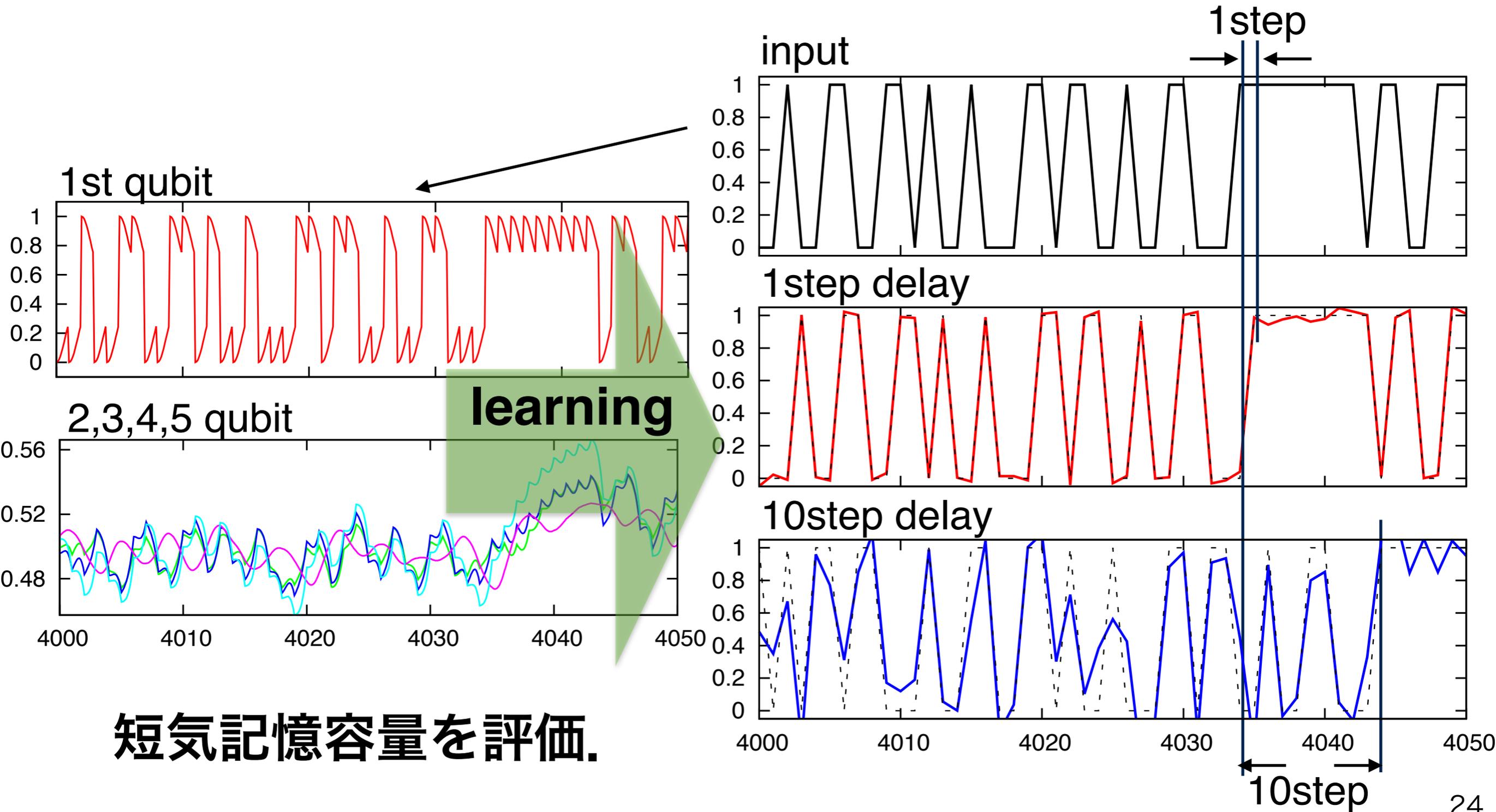
# 短期記憶

## short term memory



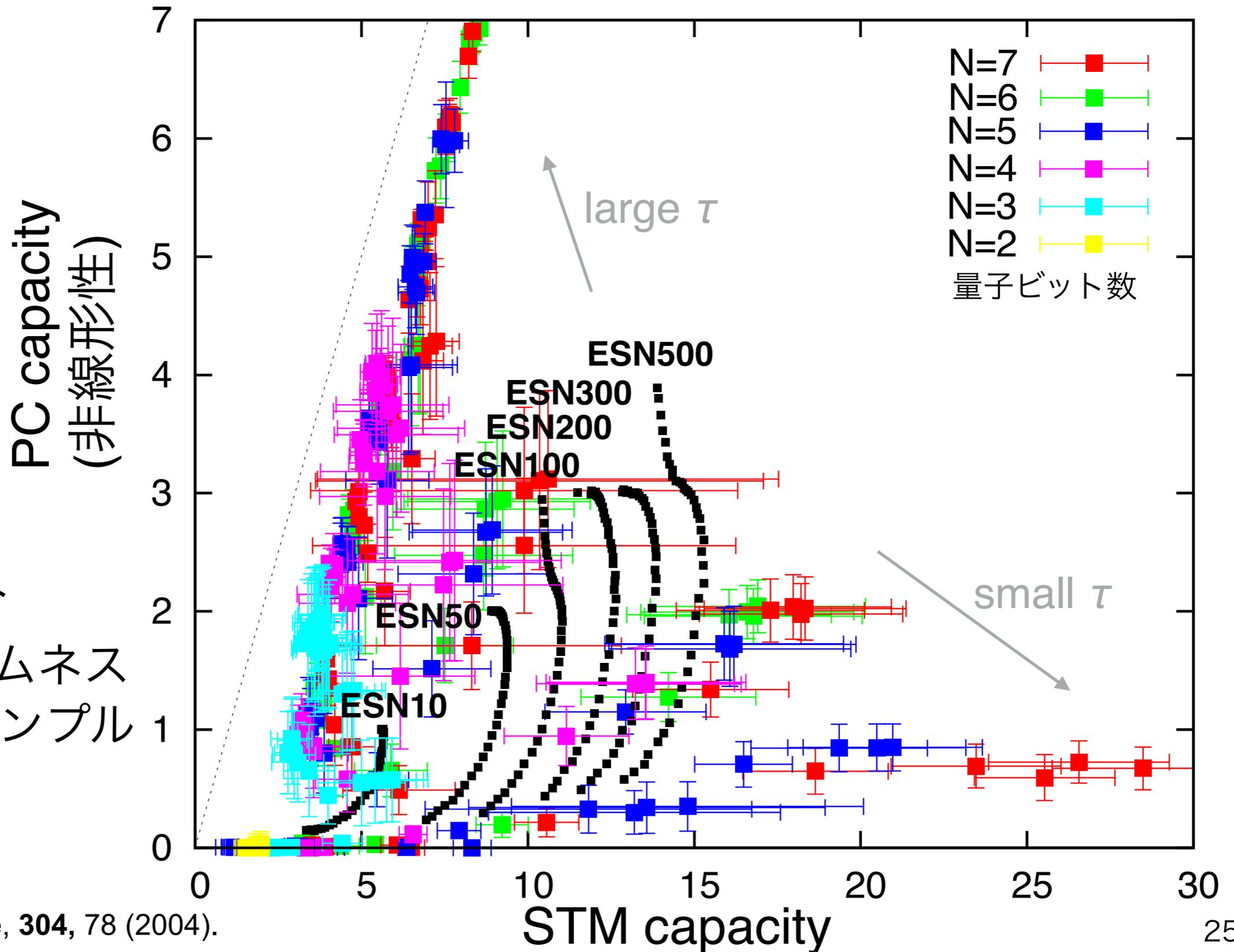
# 短期記憶

## short term memory



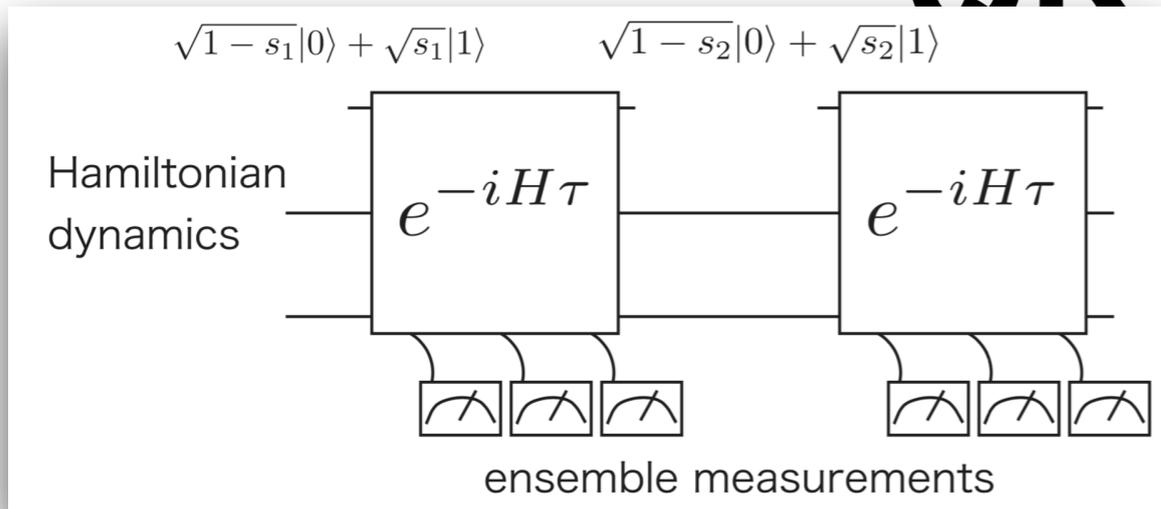
短気記憶容量を評価。

# QR性能評価



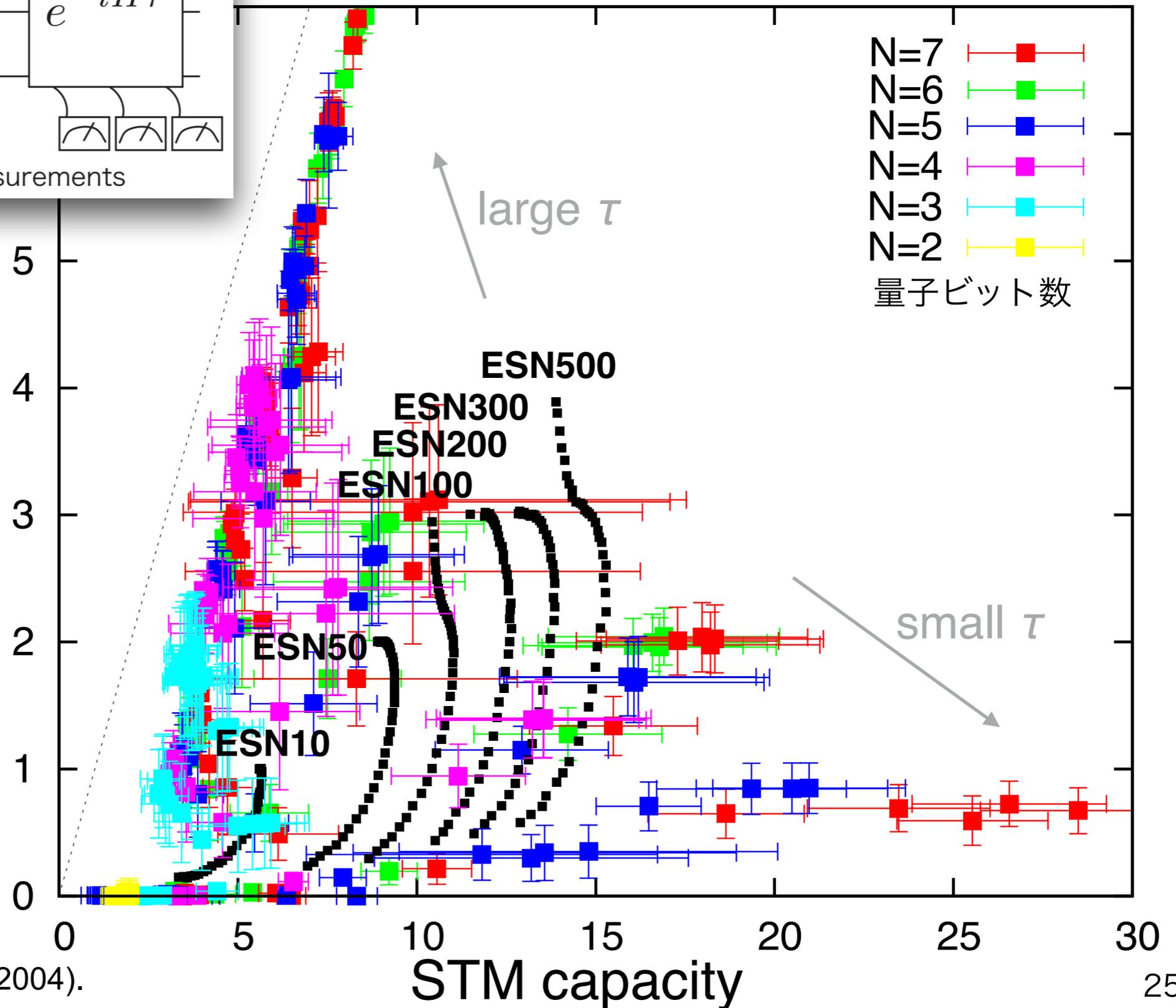
- 2-7量子ビット
- 結合のランダムネスに対して20サンプル
- 時間間隔 $\tau$
- ESNと比較

# QR性能評価



PC capacity  
(非線形性)

- 2-7量子ビット
- 結合のランダムネスに対して20サンプル
- 時間間隔  $\tau$
- ESNと比較



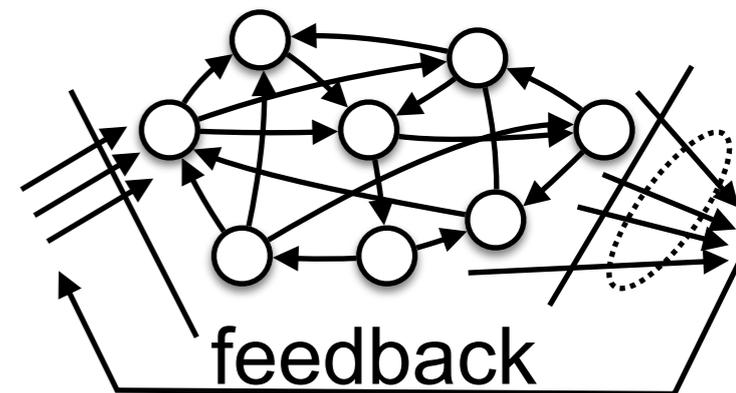
# カオス時系列データ予測

Mackey-glass time series

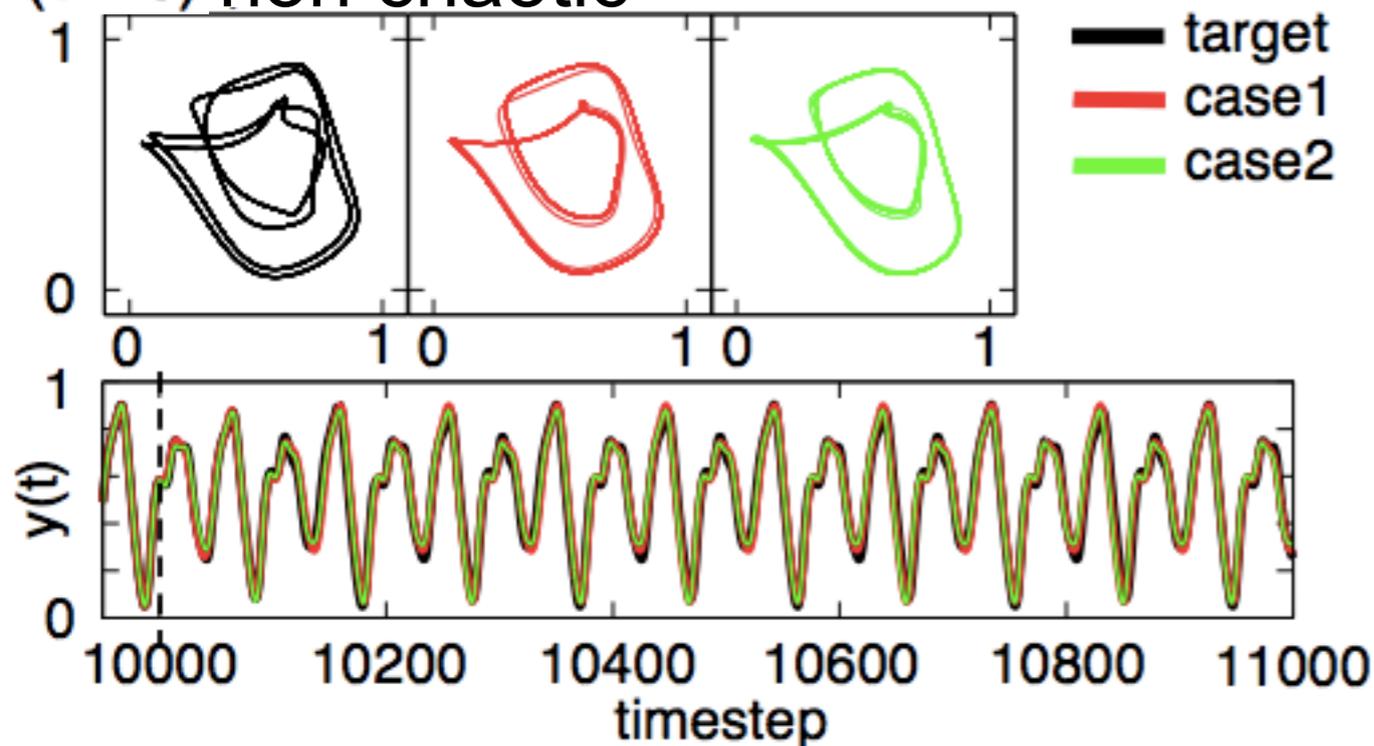
$$\frac{d}{dt}z(t) = \beta \frac{z(t-\tau)}{1+z(t-\tau)^n} - \gamma z(t)$$

$$\beta = 0.2, \gamma = 0.1, n = 10$$

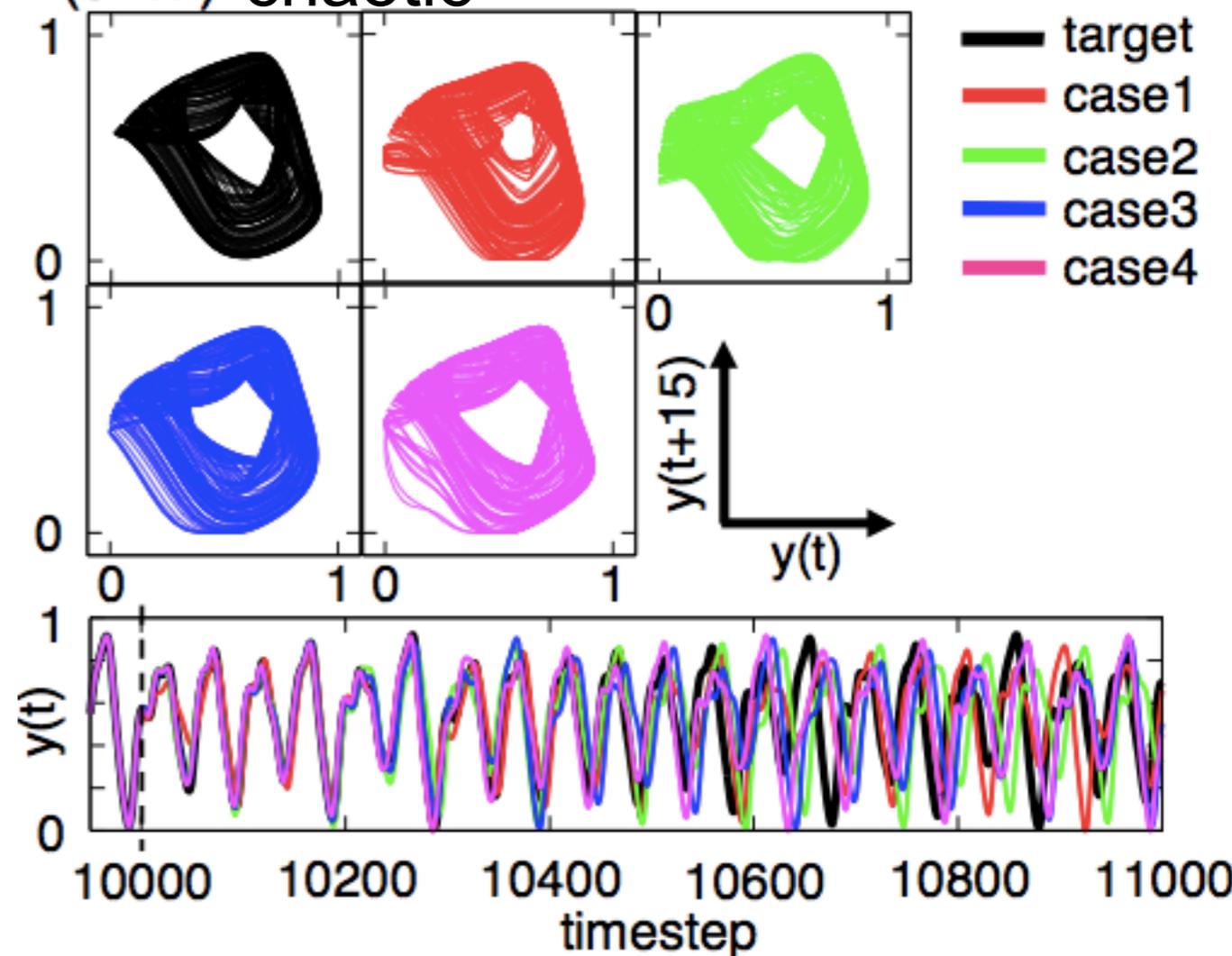
10000ステップ学習



( $\tau=16$ ) non chaotic



( $\tau=17$ ) chaotic



出力を次の入力としてfeedback  
し力学系を埋め込む。

# 詳しくは

## Quantum Native Dojo

### 目次

第0章 そもそも量子コンピュータとは？

第1章 量子情報の基礎

第2章 量子アルゴリズム入門

第3章 量子アルゴリズムの実行環境

第4章 量子ダイナミクスシミュレーション

### 第5章 変分量子回路に基づくアルゴリズム

5-1. Variational Quantum Eigensolver (VQE) アルゴリズム

5-2. Quantum Circuit learning

コラム：Quantum Circuit Learning を用いた分類

### コラム：量子リザーバコンピューティング

問題設定

素朴なRNNによる時系列回帰

より高度なRNN

リザーバコンピューティング

量子リザーバコンピューティング

量子リザーバコンピューティングの性能について

5-3. Quantum Approximate Optimization Algorithm (QAOA):量子近似最適化アルゴリズム

ホーム » 第5章 変分量子回路に基づくアルゴリズム »

コラム：量子リザーバコンピューティング

[Edit on GitHub](#) [Open in Colab](#)

5-2節ではQuantum circuit learningという量子機械学習手法について学んだ。このコラムでは、NISQデバイスを用いた機械学習手法である量子リザーバコンピューティングについて紹介する。

## コラム：量子リザーバコンピューティング

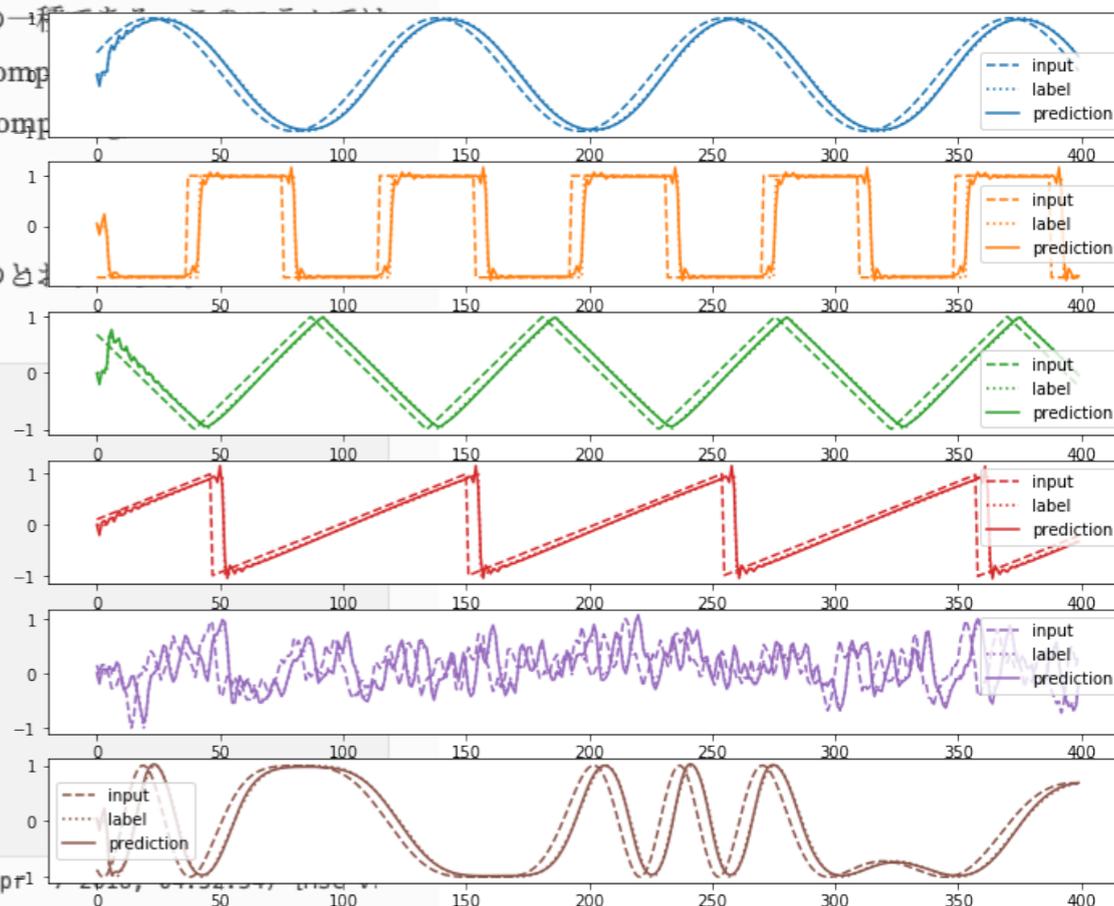
リカレントニューラルネットワーク(Recurrent neural network, RNN)とは主として時系列データの学習に用いられるニューラルネットワークモデルの一種である。RNNの一種であるリザーバコンピューティング(Reservoir computing)あるいは量子リザーバコンピューティング(Quantum reservoir computing)がある。

必要なライブラリの動作を確認しているバージョンは下記のとおり

```
[1]: import sys
import numpy as np
import scipy
import matplotlib
import matplotlib.pyplot as plt
import tensorflow as tf
import tqdm
import time

print("python: %s"%sys.version)
print("numpy: %s"%np.version.version)
print("scipy: %s"%scipy.version.version)
print("matplotlib: %s"%matplotlib.__version__)
print("tensorflow: %s"%tf.VERSION)
print("tqdm: %s"%tqdm.__version__.__version__)
```

```
python: 3.5.5 |Anaconda 4.2.0 (64-bit)| (default, Apr 1
numpy: 1.14.3
scipy: 1.2.0
matplotlib: 2.1.2
```



# その後の発展

提案した当初は、**量子系そのものを用いて機械学習をする利点**はあまり明確ではなかった。

# その後の発展

提案した当初は、**量子系そのものを用いて機械学習をする利点**はあまり明確ではなかった。

npj | Quantum Information

www.nature.com/npjqi

ARTICLE OPEN

## Quantum reservoir processing

Sanjib Ghosh<sup>1</sup>, Andrzej Opala<sup>2</sup>, Michał Matuszewski<sup>2</sup>, Tomasz Paterek<sup>1,3</sup> and Timothy C. H. Liew<sup>1,3</sup>

The concurrent rise of artificial intelligence and quantum information poses an opportunity for creating interdisciplinary technologies like quantum neural networks. Quantum reservoir processing, introduced here, is a platform for quantum information processing developed on the principle of reservoir computing that is a form of an artificial neural network. A quantum reservoir processor can perform qualitative tasks like recognizing quantum states that are entangled as well as quantitative tasks like estimating a nonlinear function of an input quantum state (e.g., entropy, purity, or logarithmic negativity). In this way, experimental schemes that require measurements of multiple observables can be simplified to measurement of one observable on a trained quantum reservoir processor.

*npj Quantum Information* (2019)5:35; <https://doi.org/10.1038/s41534-019-0149-8>

- Entanglement classification(detection)
- Logarithmic negativity
- von Neumann entropy
- Renyi entropy

Ghosh, Sanjib, et al. "Quantum reservoir processing." *npj Quantum Information* 5.1 (2019): 1-6.

# その後の発展

提案した当初は、**量子系そのものを用いて機械学習をする利点**はあまり明確ではなかった。

npj | Quantum Information

www.nature.com/npjqi

ARTICLE OPEN

## Quantum reservoir processing

Sanjib Ghosh<sup>1</sup>, Andrzej Opala<sup>2</sup>, Michał Matuszewski<sup>2</sup>, Tomasz Paterek<sup>1,3</sup> and Timothy C. H. Liew<sup>1,3</sup>

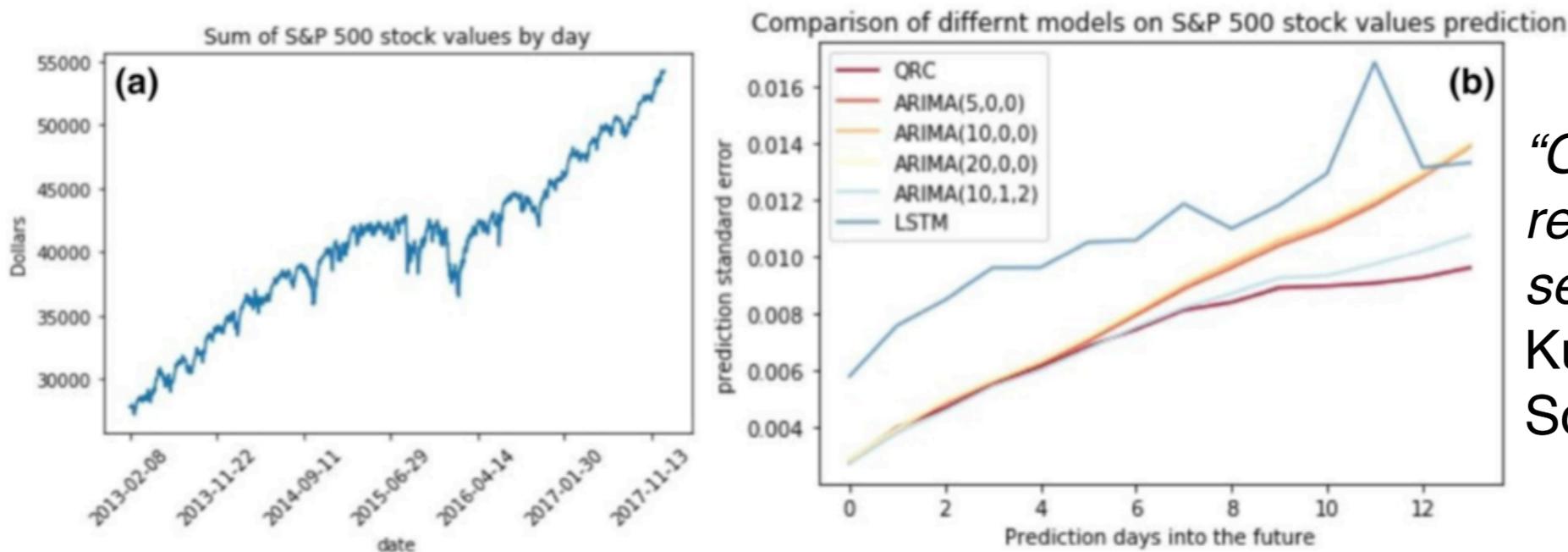
The concurrent rise of artificial intelligence and quantum information poses an opportunity for creating interdisciplinary technologies like quantum neural networks. Quantum reservoir processing, introduced here, is a platform for quantum information processing developed on the principle of reservoir computing that is a form of an artificial neural network. A quantum reservoir processor can perform qualitative tasks like recognizing quantum states that are entangled as well as quantitative tasks like estimating a nonlinear function of an input quantum state (e.g., entropy, purity, or logarithmic negativity). In this way, experimental schemes that require measurements of multiple observables can be simplified to measurement of one observable on a trained quantum reservoir processor.

npj Quantum Information (2019)5:35; <https://doi.org/10.1038/s41534-019-0149-8>

- Entanglement classification(detection)
- Logarithmic negativity
- von Neumann entropy
- Renyi entropy

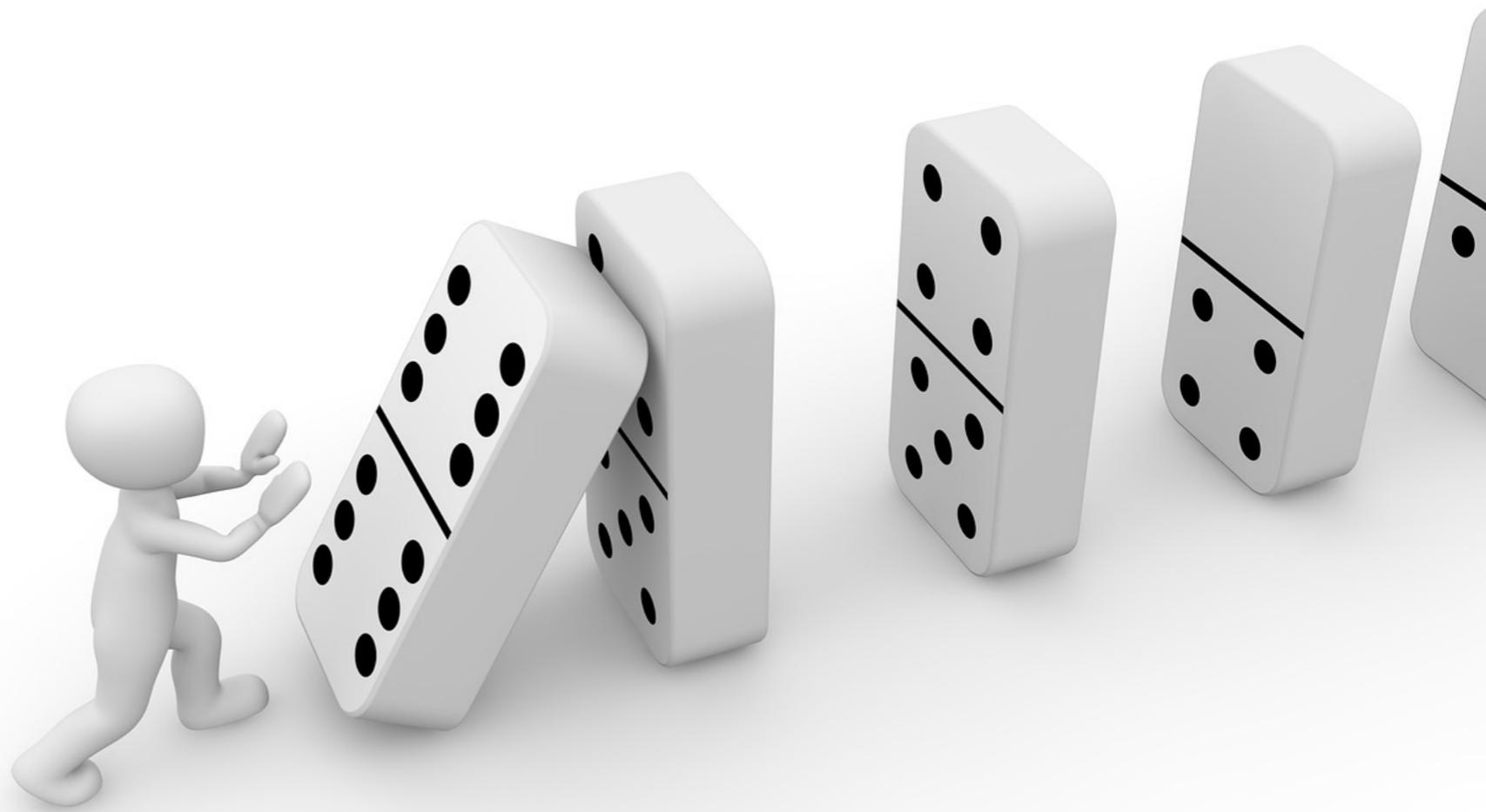
Ghosh, Sanjib, et al. "Quantum reservoir processing." npj Quantum Information 5.1 (2019): 1-6.

## OTOCを用いた量子レザバーの特徴づけと時系列データ予測



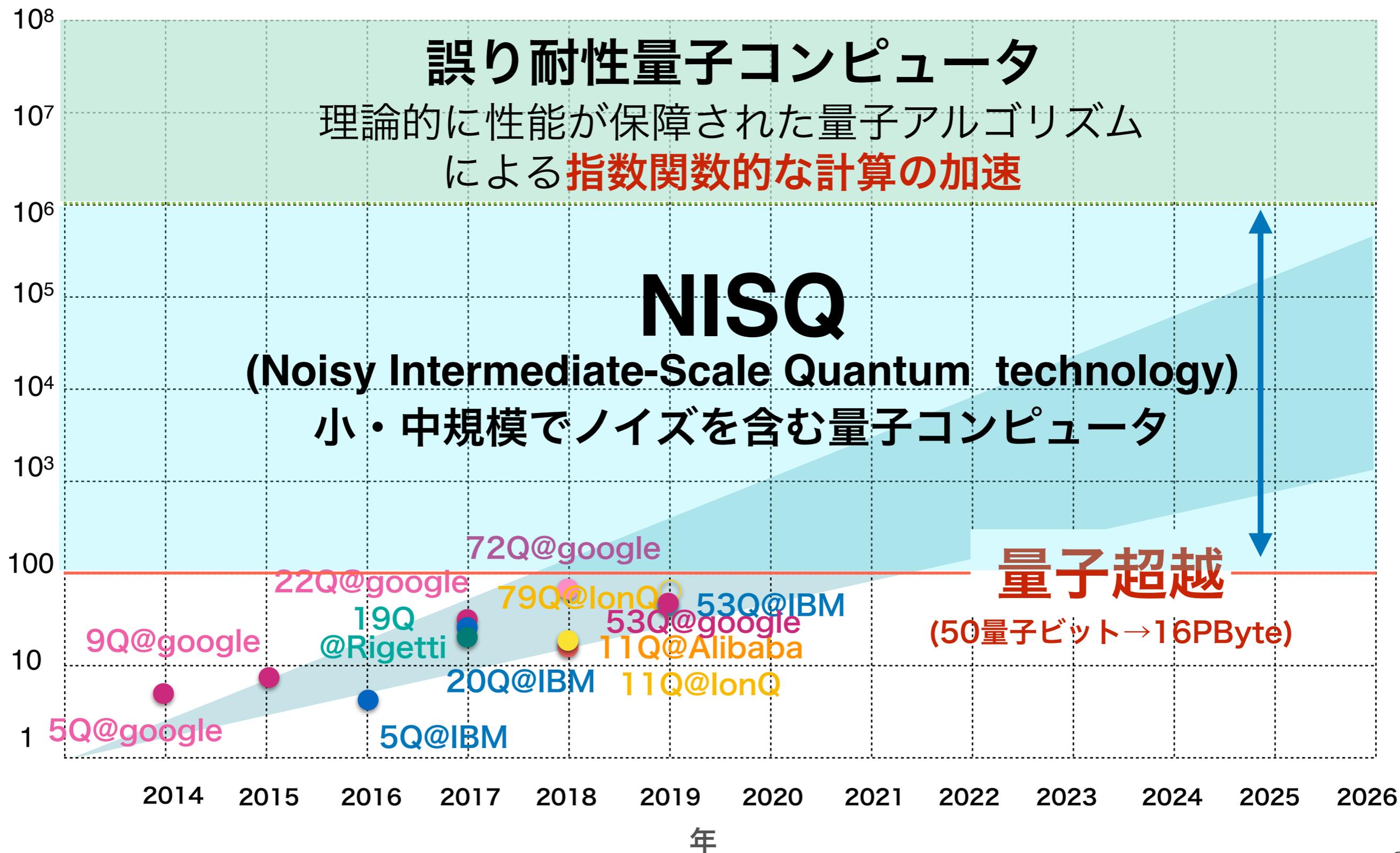
*“Optimizing a quantum reservoir computer for time series prediction”, A. Kutvonen, K. Fujii, T. Sagawa, Sci. Rep. 10, 14687 (2020).*

# 量子回路學習



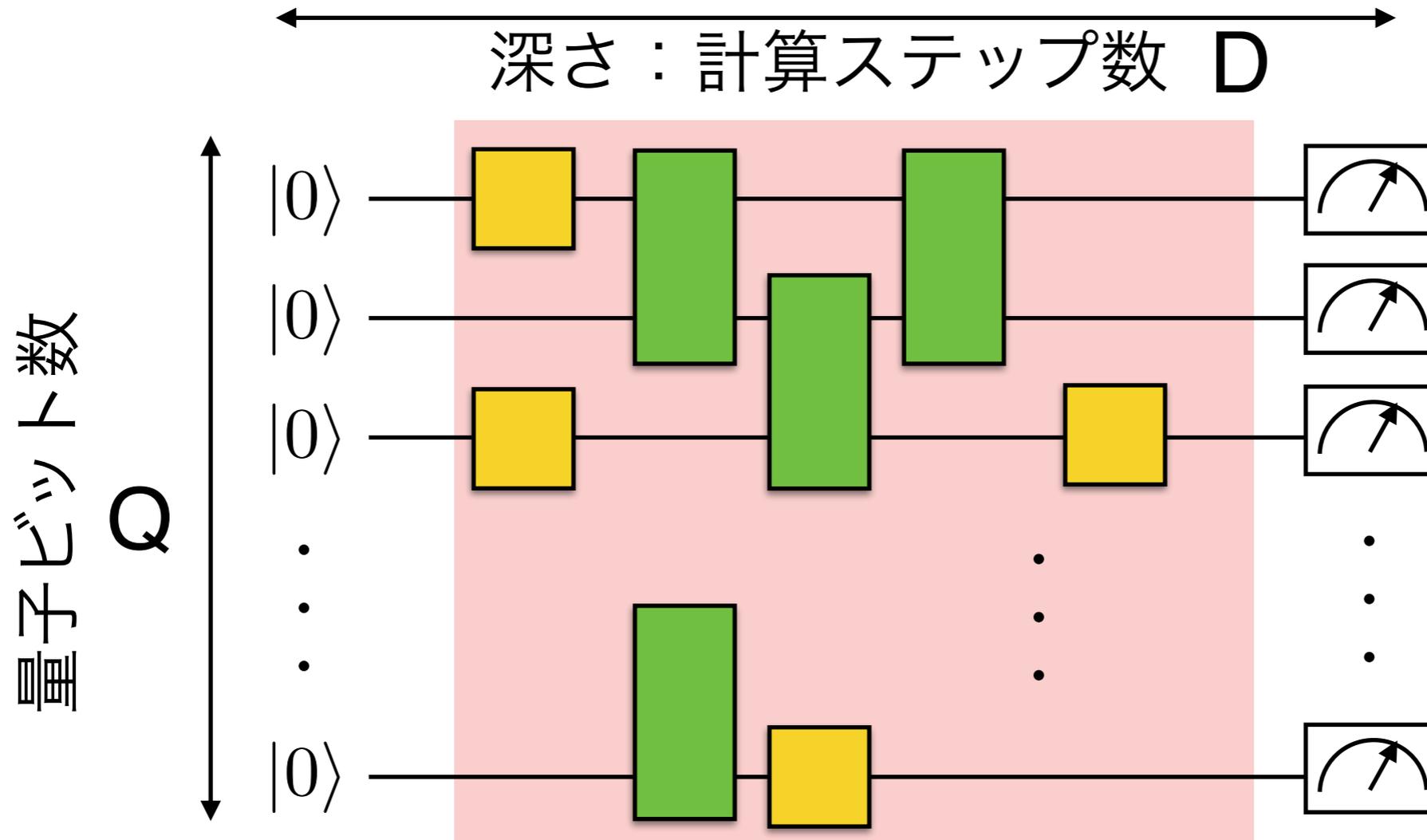
# “量子版”ムーアの法則？

量子ビット数



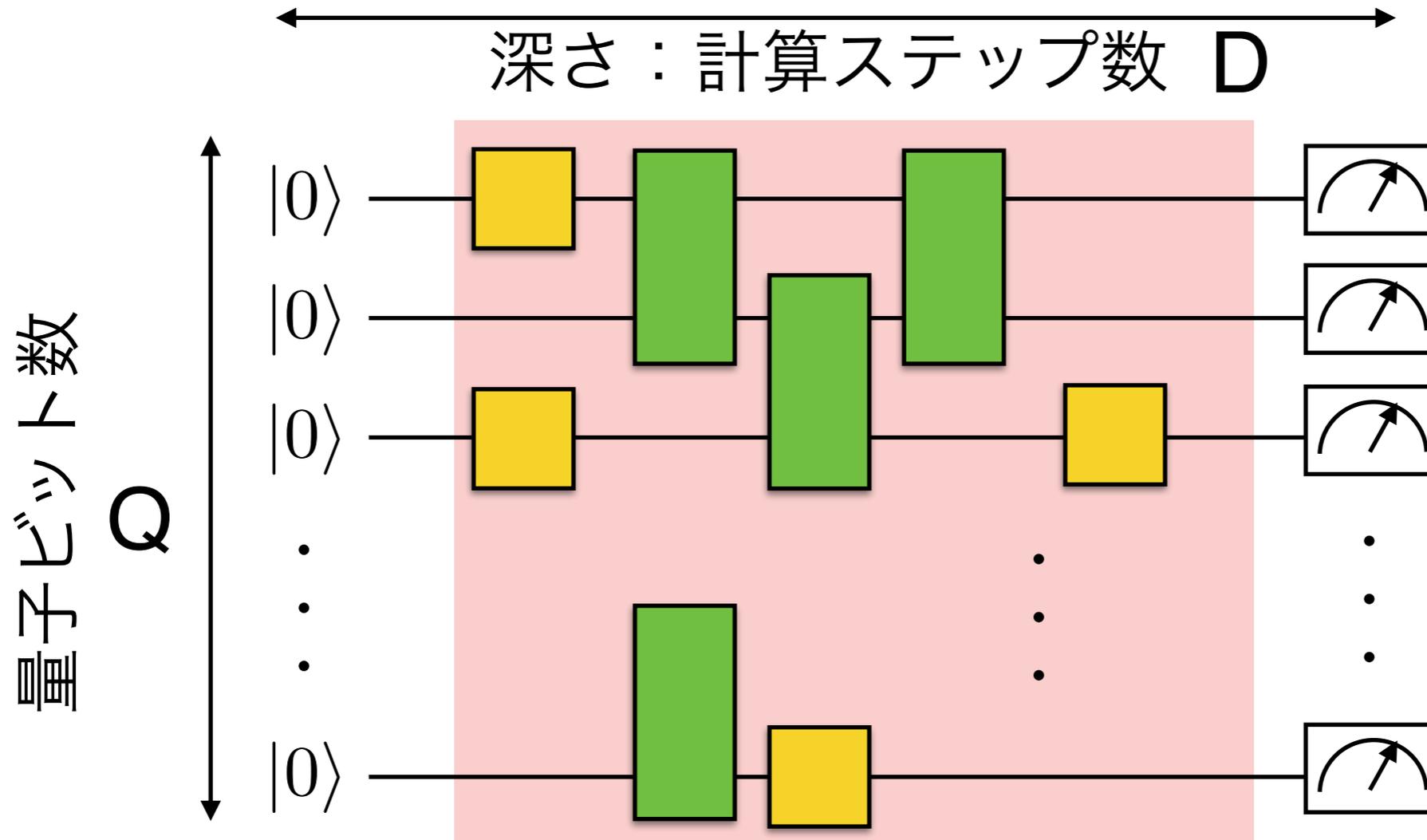
# NISQ = Noise Issue is Severe on current Quantum computer (ノイズのとの戦い)

量子ビット数だけでなく質(忠実度)も重要



# NISQ = Noise Issue is Severe on current Quantum computer (ノイズのとの戦い)

量子ビット数だけでなく質(忠実度)も重要

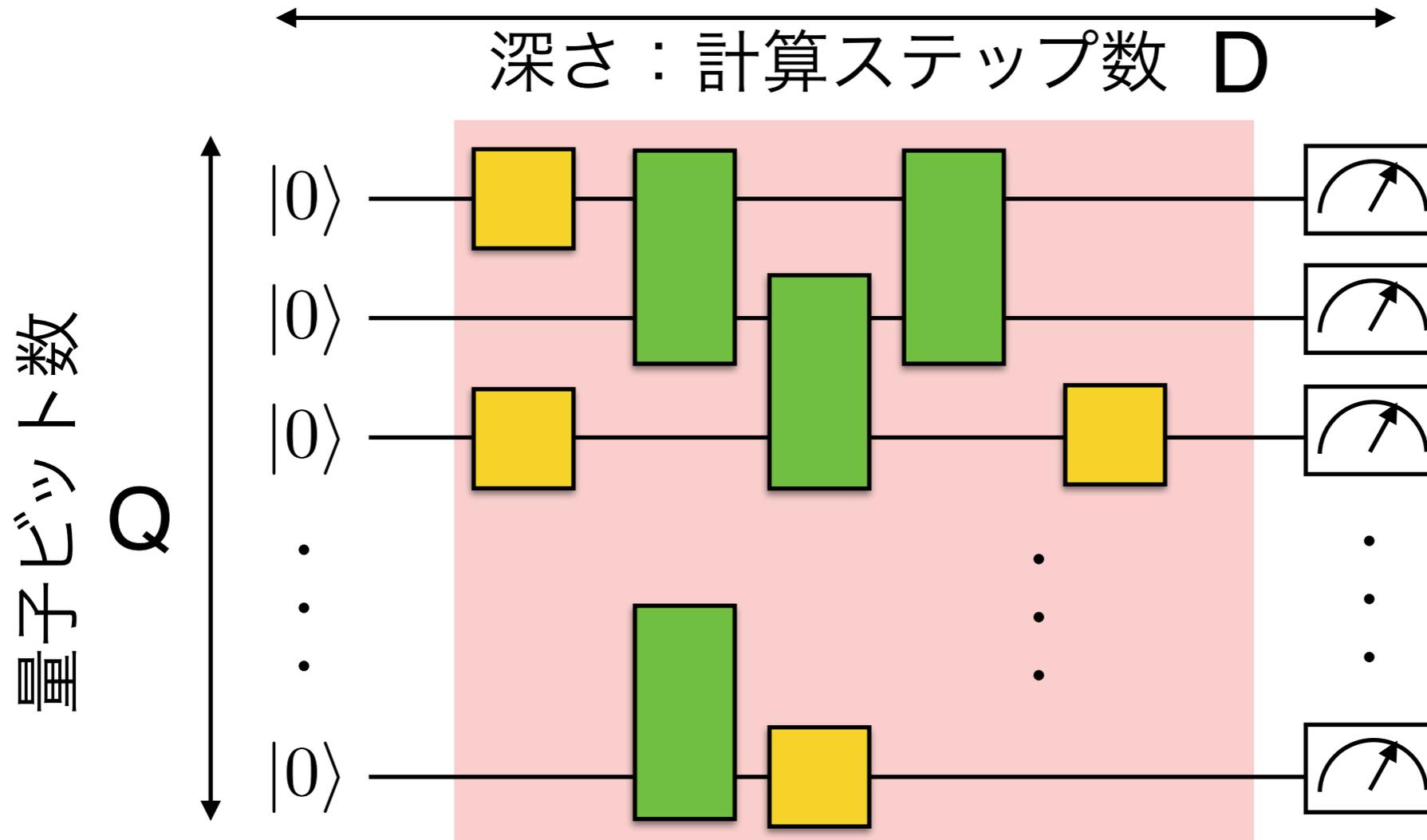


(忠実度)<sup>(QD)</sup> = 全体の忠実度

$$(0.999)^{(50\text{qubit} \cdot 20\text{depth})} = 0.368$$

# NISQ = Noise Issue is Severe on current Quantum computer (ノイズのとの戦い)

量子ビット数だけでなく質(忠実度)も重要



(忠実度)<sup>(QD)</sup> = 全体の忠実度

$$(0.999)^{(50\text{qubit} \cdot 20\text{depth})} = 0.368$$

アナログエラー  
を抑え込む必要あり  
(究極的には誤り訂正が必要)

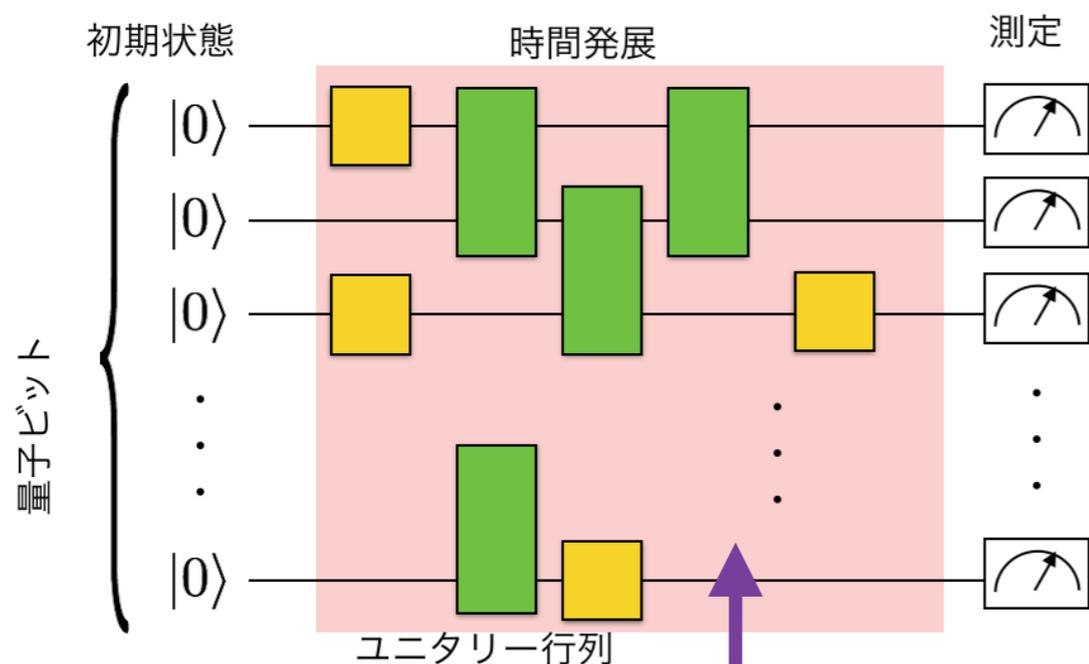
# 量子古典ハイブリッド変分アルゴリズム

量子コンピュータ

古典コンピュータ

量子にしかできないタスク

古典でもできるタスク



期待値



パラメータの更新

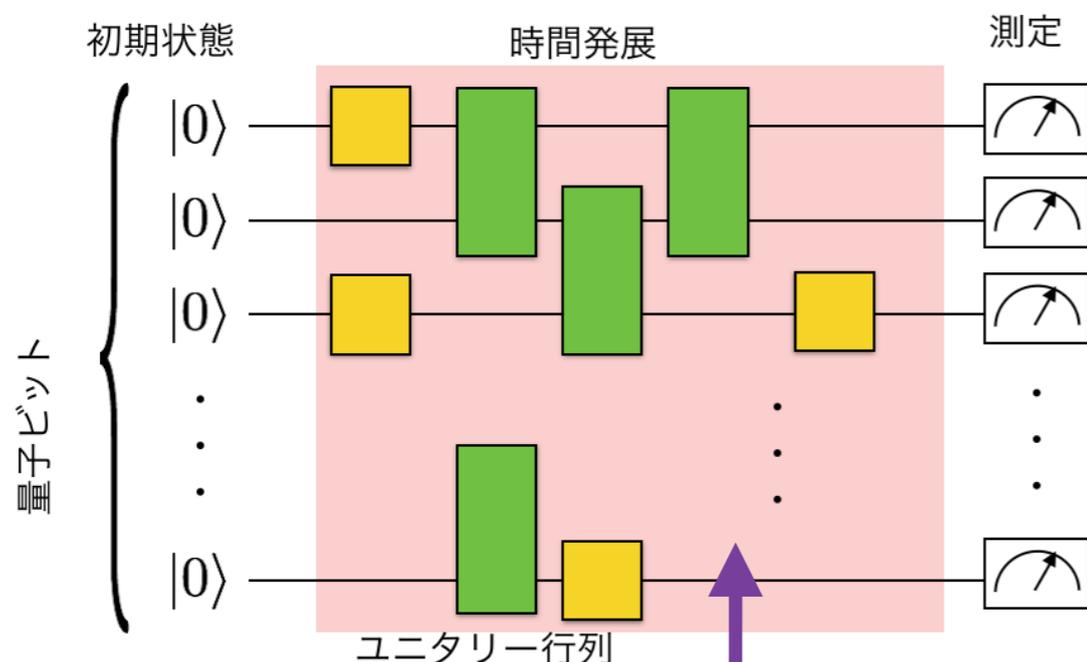
# 量子古典ハイブリッド変分アルゴリズム

量子コンピュータ

古典コンピュータ

量子にしかできないタスク

古典でもできるタスク



期待値



パラメータの更新

$$|\psi(\vec{\phi})\rangle = U(\vec{\phi})|0^n\rangle$$

$$\min \langle \psi(\vec{\phi}) | H | \psi(\vec{\phi}) \rangle$$

パラメータ付き量子回路で生成した量子状態

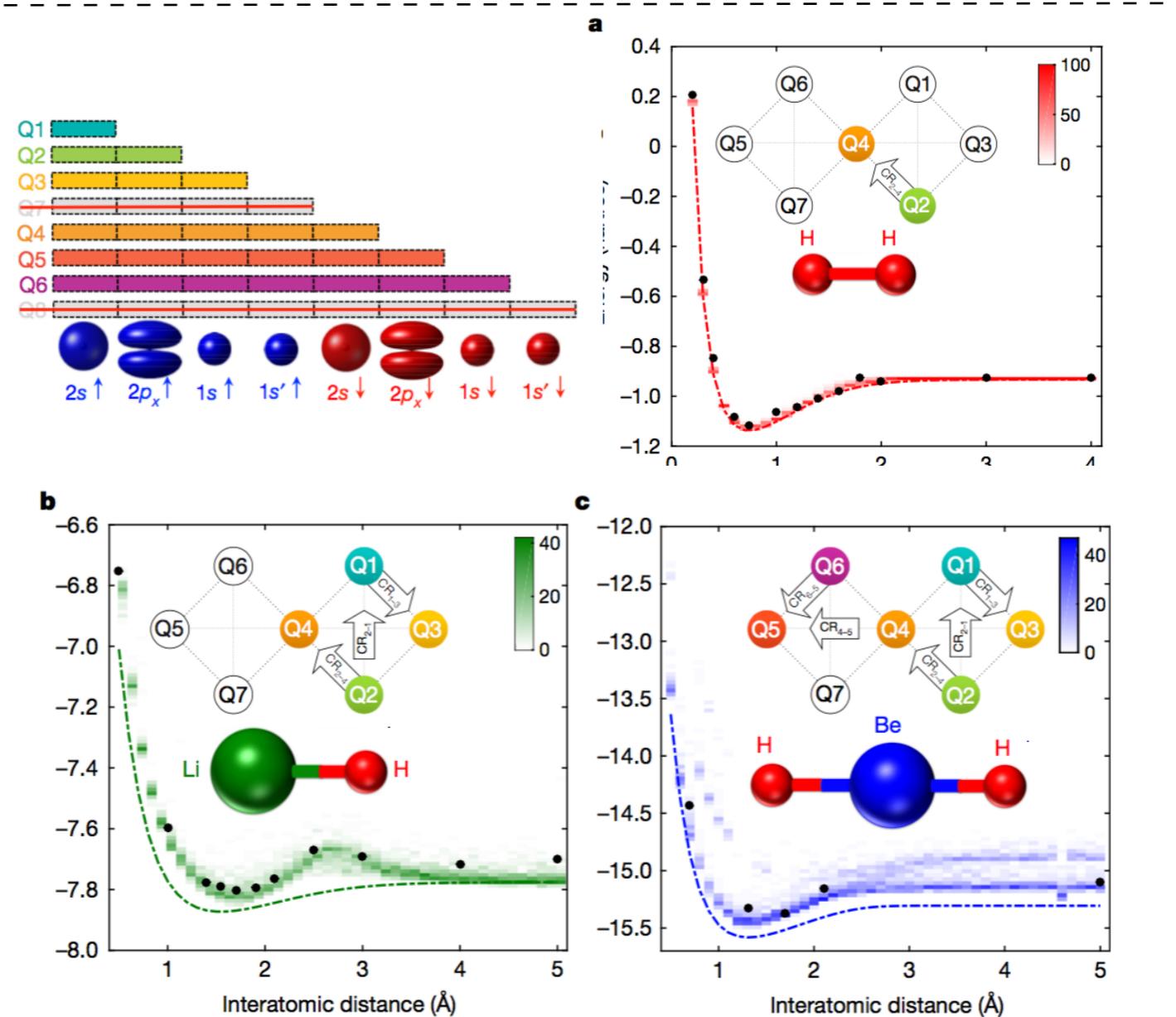
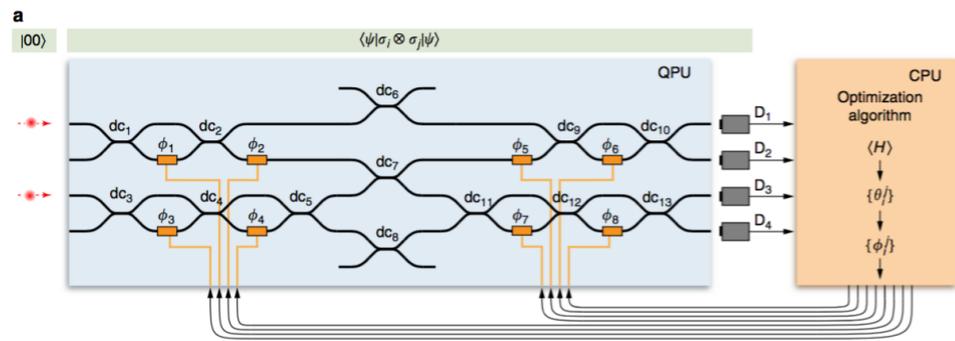
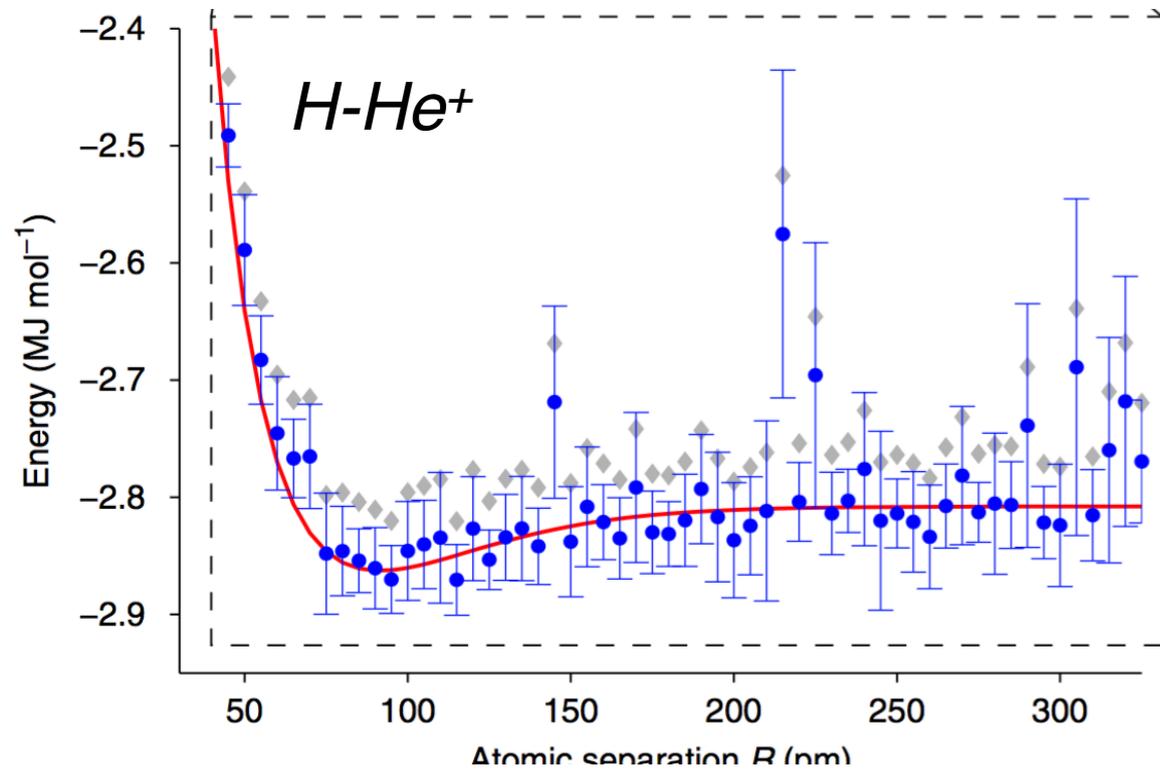
期待値が最小になるようにパラメータを更新

量子状態そのものを変分試行関数として用いることができる

# VQE (Variational quantum eigensolver)

“A variational eigenvalue solver on a photonic quantum processor”

Peruzzo, McClean *et al*, Nature Communication 5:4213 (2014)

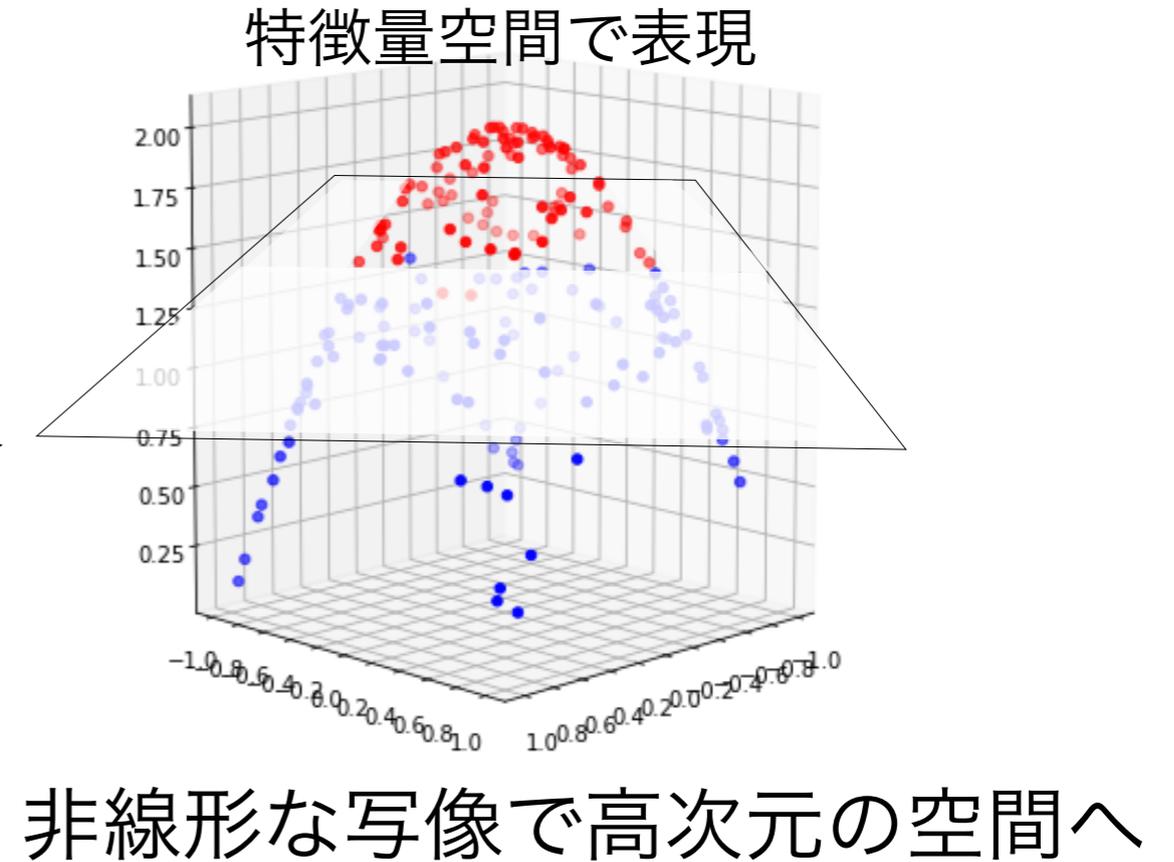
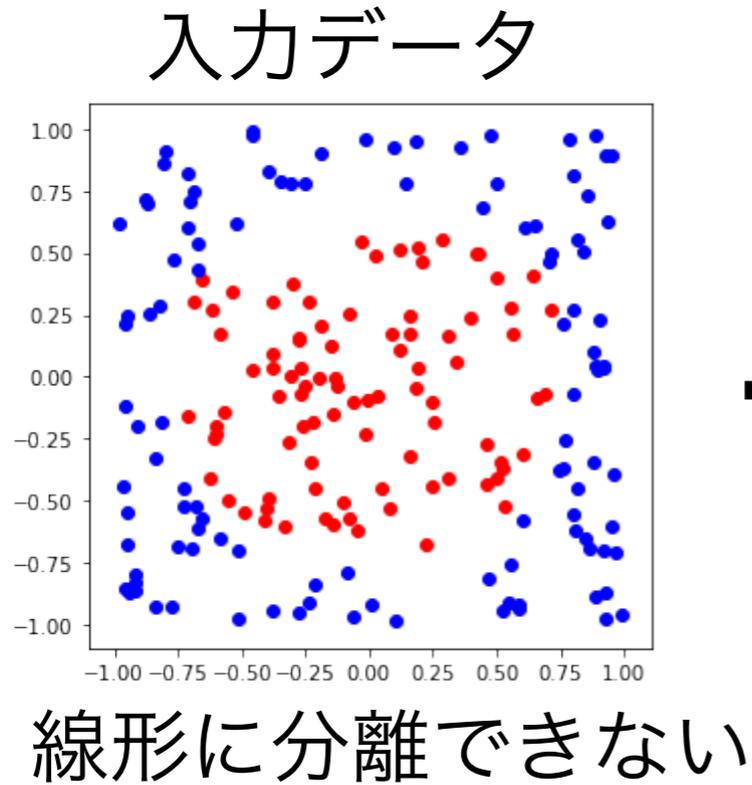


“Hardware-efficient Quantum Optimizer for Small Molecules and Quantum Magnets”

Kandala, Mezzacapo *et al*, Nature 549 242 (2017)

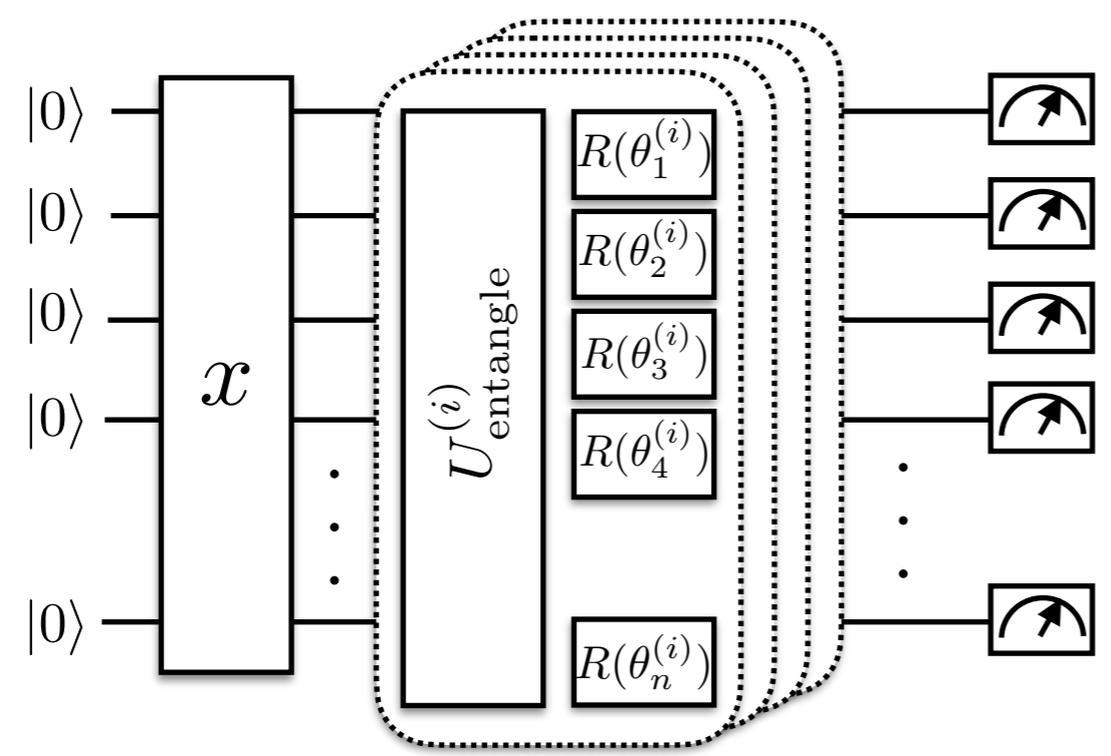
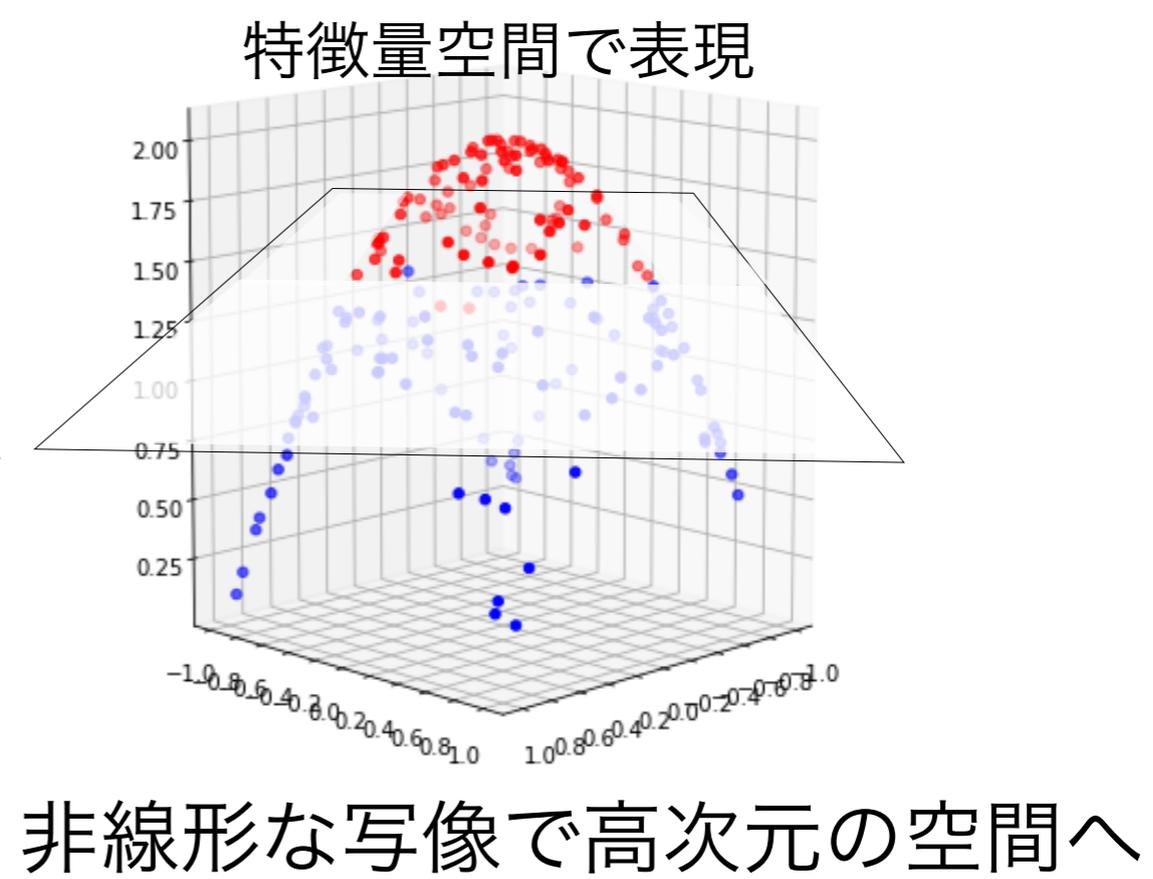
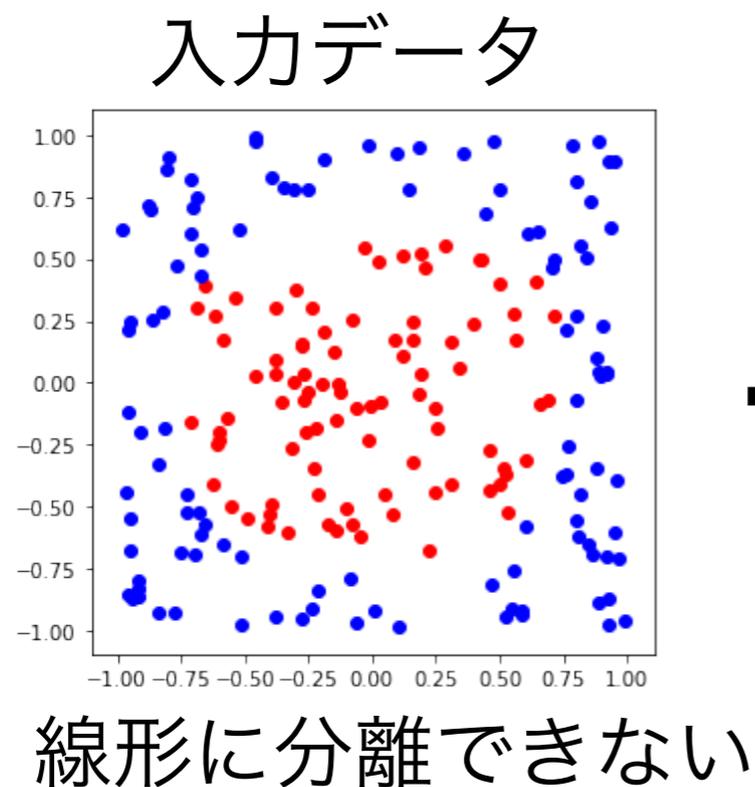
# 量子コンピュータと機械学習

高次元の特徴量空間上で線形分離 (カーネル法、サポートベクトルマシン)



# 量子コンピュータと機械学習

高次元の特徴量空間上で線形分離 (カーネル法、サポートベクトルマシン)



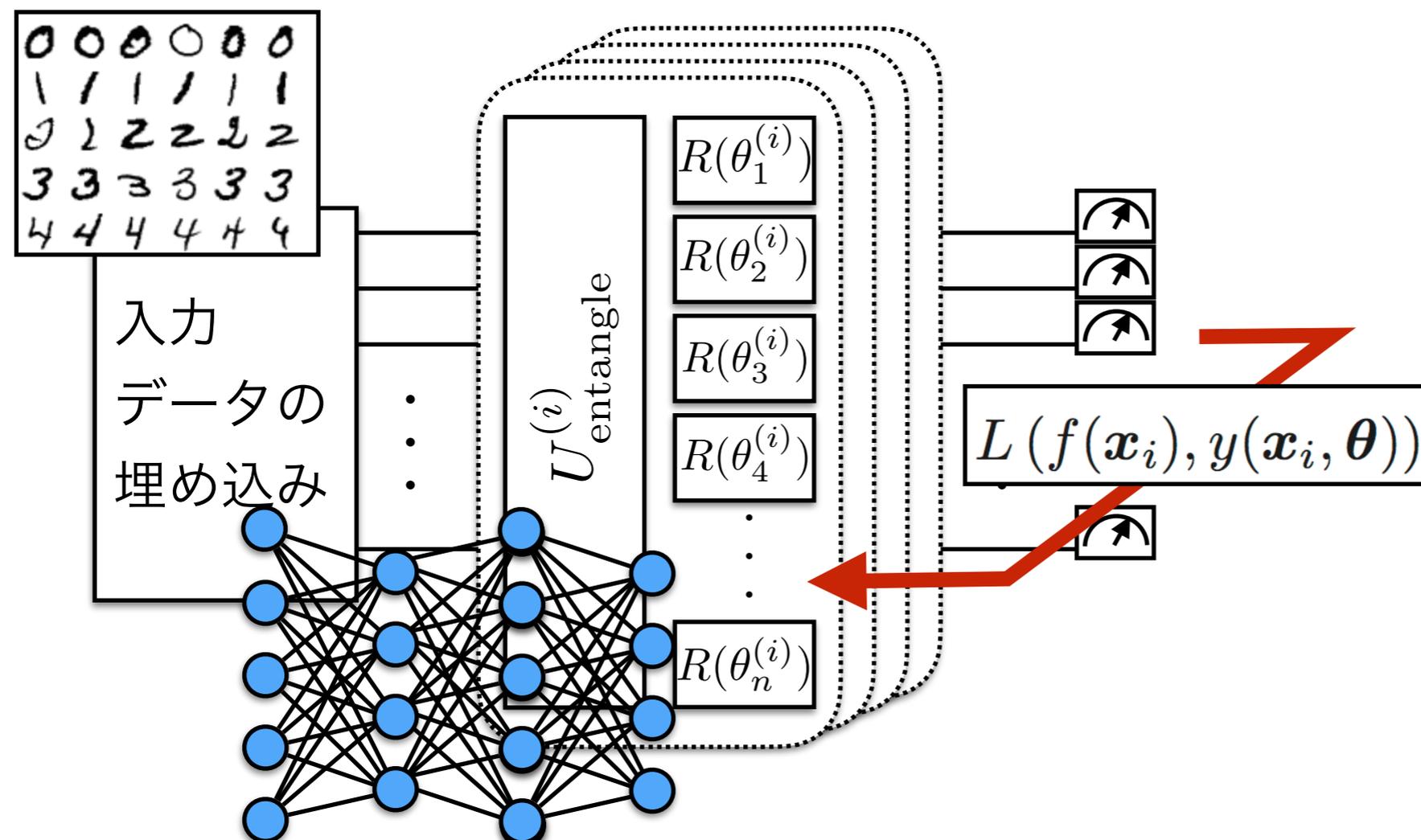
$$|\Psi(x)\rangle = \begin{pmatrix} \psi_0(x) \\ \psi_1(x) \\ \vdots \\ \psi_{2^n-1}(x) \end{pmatrix}$$

指数関数的に大きな空間上の点

# 量子回路学習

“Quantum Circuit Learning”,  
K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii  
*Phys. Rev. A* 98, 032309 (2018)

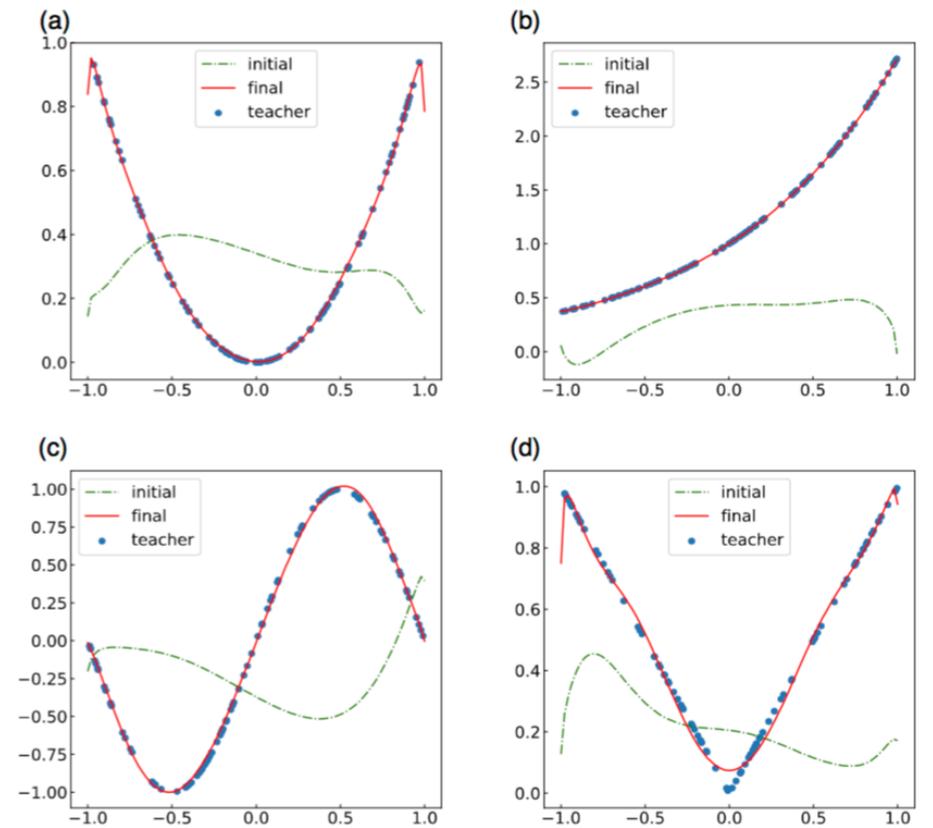
- 変分量子回路を用いた教師あり学習
- テンソル積構造による非線型性とユニタリ変換
- 量子回路の解析微分による勾配の計算



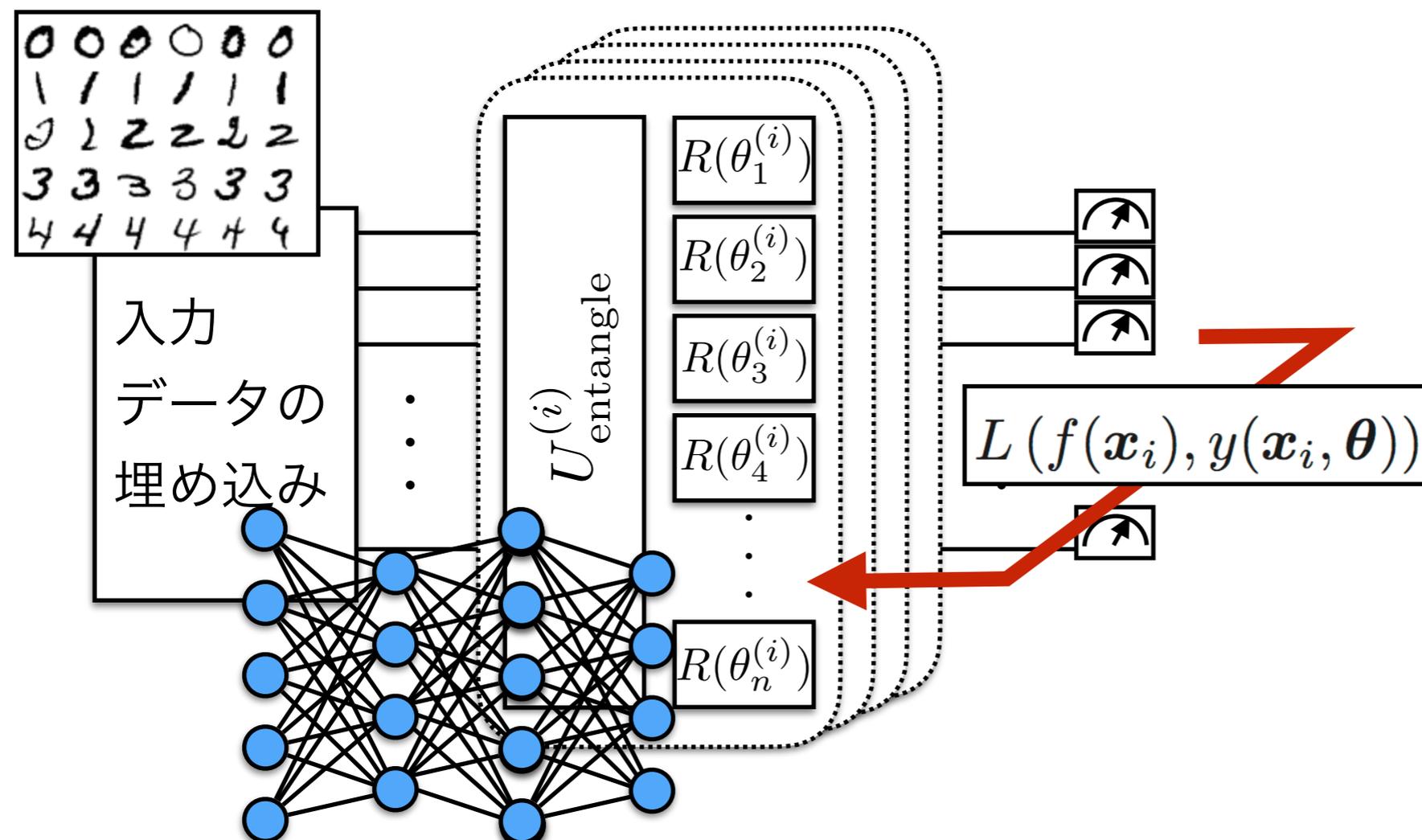
# 量子回路学習

“Quantum Circuit Learning”,  
 K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii  
 Phys. Rev. A 98, 032309 (2018)

- 変分量子回路を用いた教師あり学習
- テンソル積構造による非線型性とユニタリ変換
- 量子回路の解析微分による勾配の計算



非線形関数の学習 (汎化)

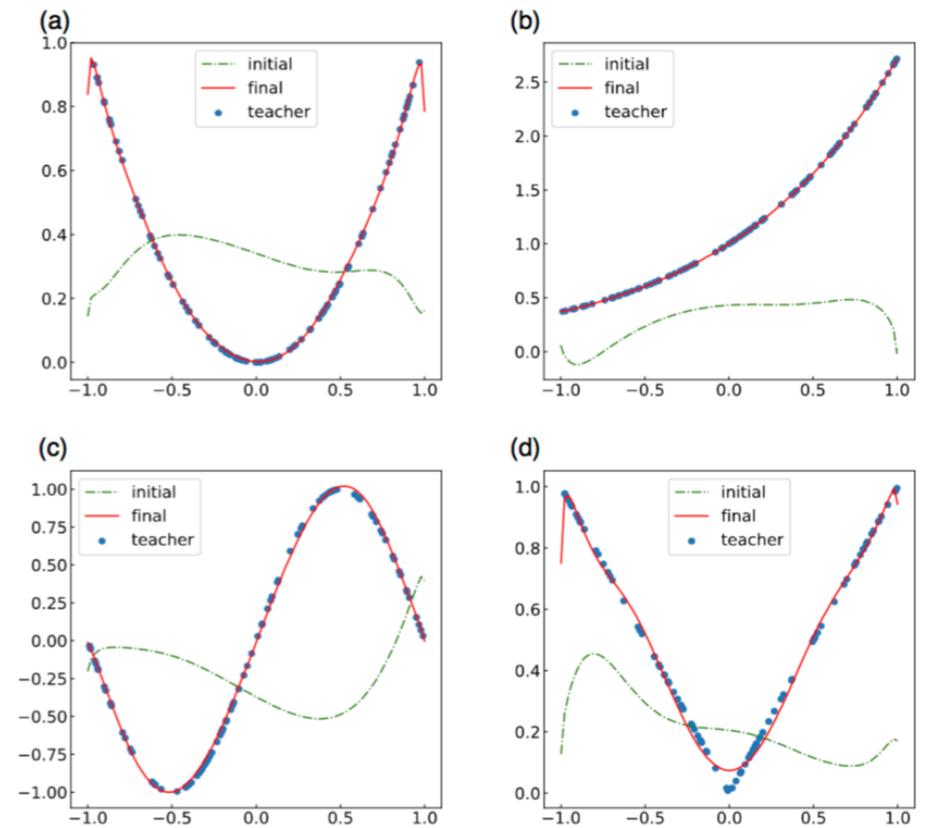


$$L(f(x_i), y(x_i, \theta))$$

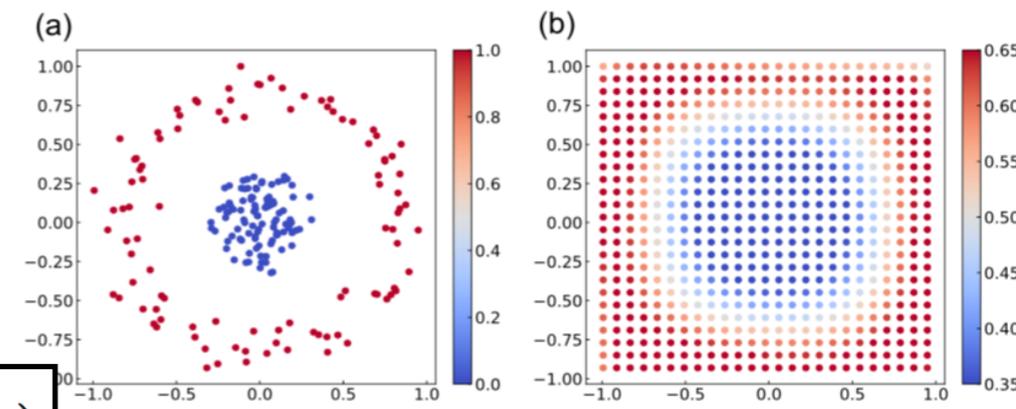
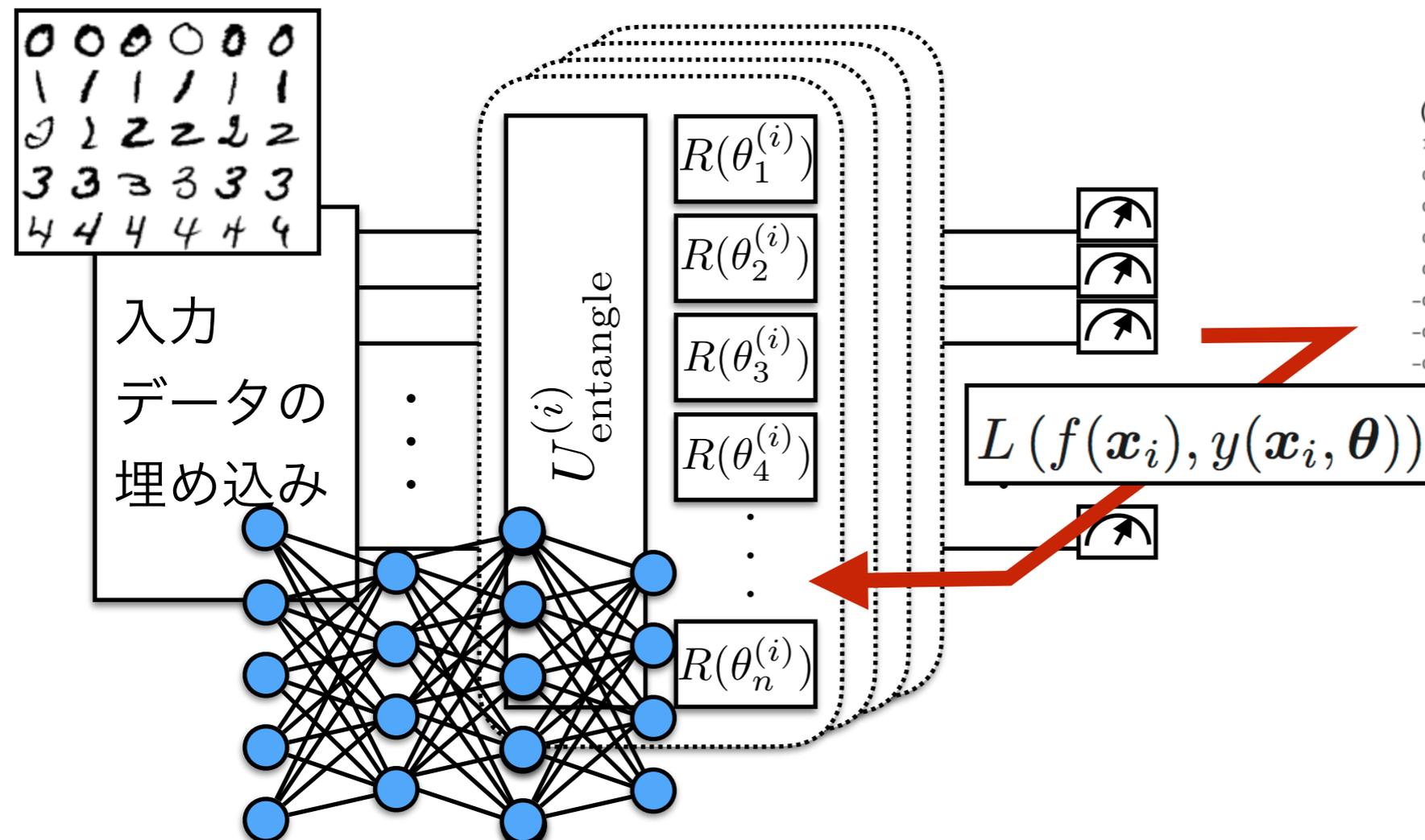
# 量子回路学習

“Quantum Circuit Learning”,  
 K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii  
*Phys. Rev. A* 98, 032309 (2018)

- 変分量子回路を用いた教師あり学習
- テンソル積構造による非線型性とユニタリ変換
- 量子回路の解析微分による勾配の計算



非線形関数の学習 (汎化)

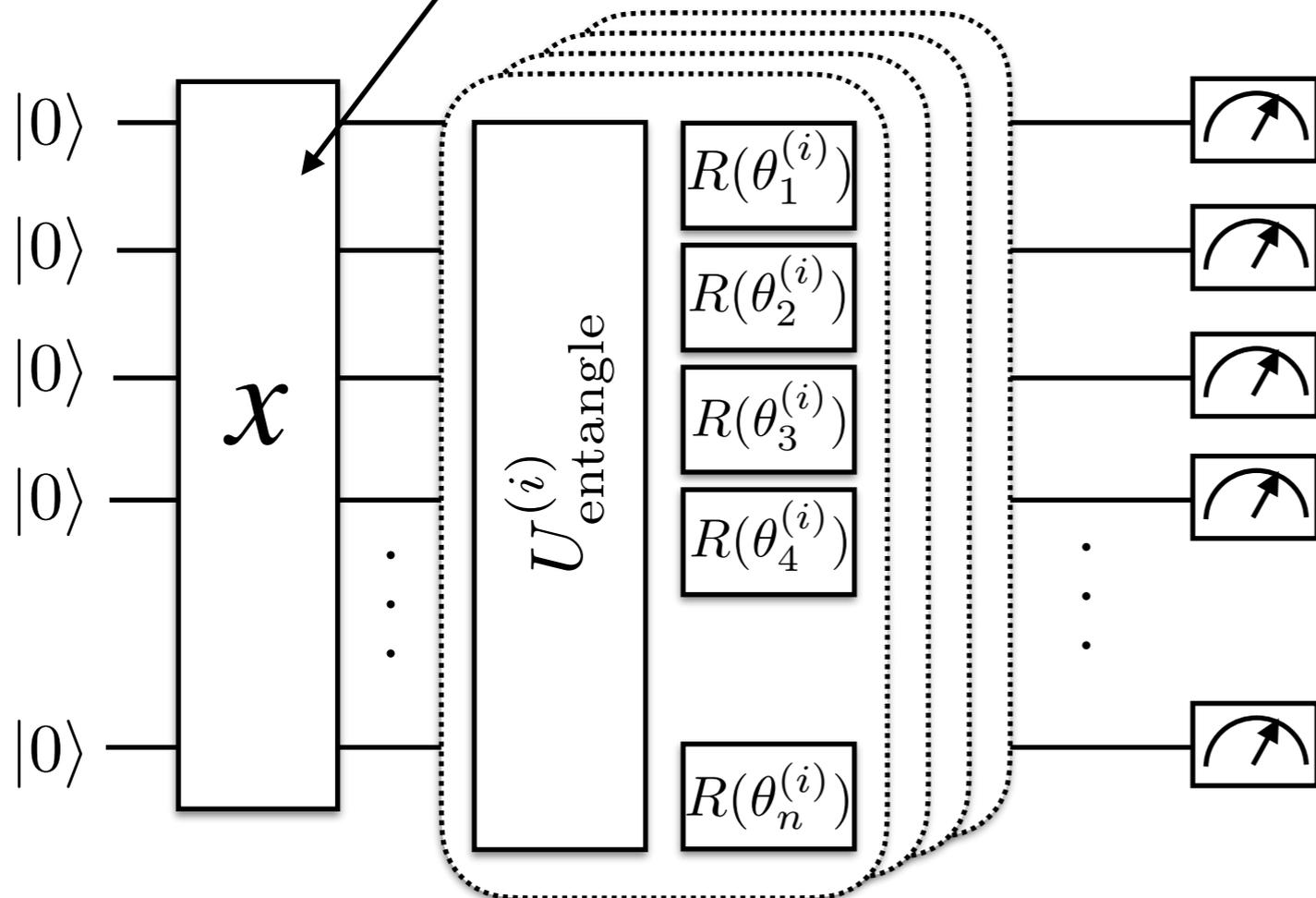


2値分類問題の学習

# 量子回路学習

- 量子系の時間発展は線形→テンソル積構造による非線形性

$$\begin{pmatrix} \psi_1^0(x_1) \\ \psi_1^1(x_1) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_k^0(x_k) \\ \psi_k^1(x_k) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_n^0(x_n) \\ \psi_n^1(x_n) \end{pmatrix} = \sum_{i_1 \dots i_n} \prod_k \left[ \psi_k^{i_k}(x_k) \right] |i_1 \dots i_n\rangle \equiv |\Psi(\vec{x})\rangle$$



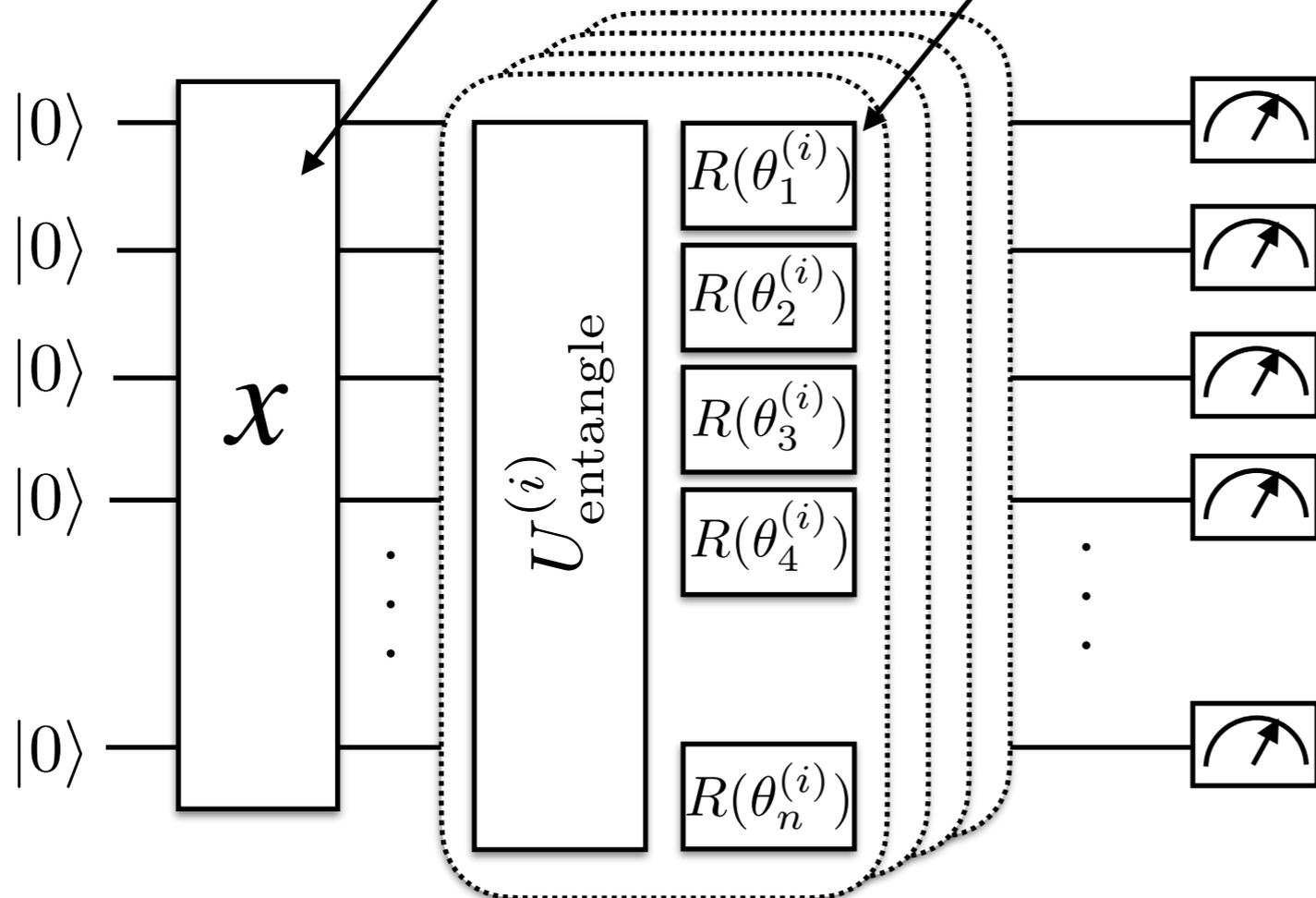
# 量子回路学習

- 量子系の時間発展は線形→テンソル積構造による非線形性

$$\begin{pmatrix} \psi_1^0(x_1) \\ \psi_1^1(x_1) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_k^0(x_k) \\ \psi_k^1(x_k) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_n^0(x_n) \\ \psi_n^1(x_n) \end{pmatrix} = \sum_{i_1 \dots i_n} \prod_k [\psi_k^{i_k}(x_k)] |i_1 \dots i_n\rangle \equiv |\Psi(\vec{x})\rangle$$

ユニタリー時間発展

$$W(\vec{\theta}) |\Psi(\vec{x})\rangle$$



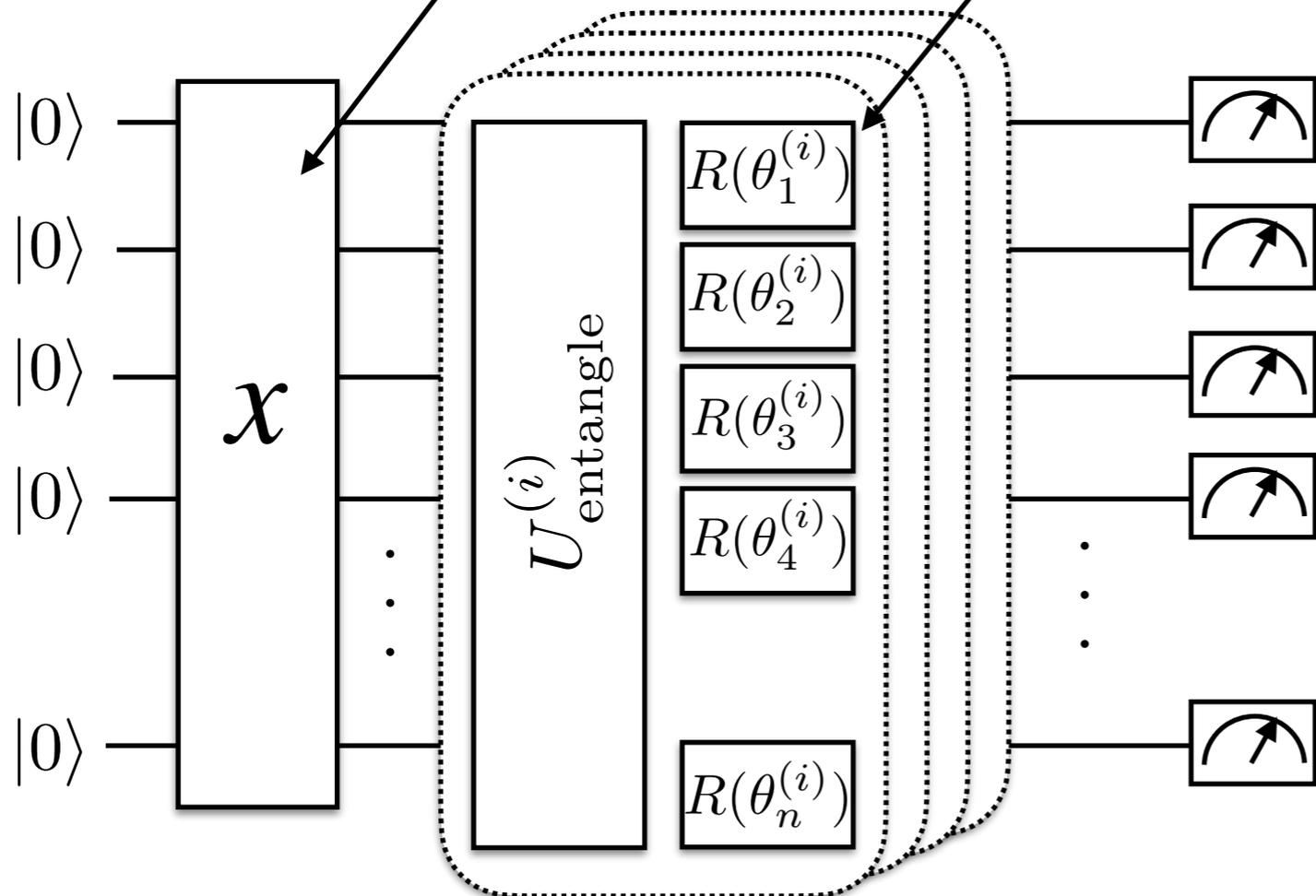
# 量子回路学習

- 量子系の時間発展は線形→テンソル積構造による非線形性

$$\begin{pmatrix} \psi_1^0(x_1) \\ \psi_1^1(x_1) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_k^0(x_k) \\ \psi_k^1(x_k) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_n^0(x_n) \\ \psi_n^1(x_n) \end{pmatrix} = \sum_{i_1 \dots i_n} \prod_k [\psi_k^{i_k}(x_k)] |i_1 \dots i_n\rangle \equiv |\Psi(\vec{x})\rangle$$

ユニタリー時間発展

$$W(\vec{\theta}) |\Psi(\vec{x})\rangle$$



測定

$$y(x, \theta) =$$

$$\langle \Psi(\vec{x}) | W^\dagger(\vec{\theta}) A W(\vec{\theta}) | \Psi(\vec{x}) \rangle$$

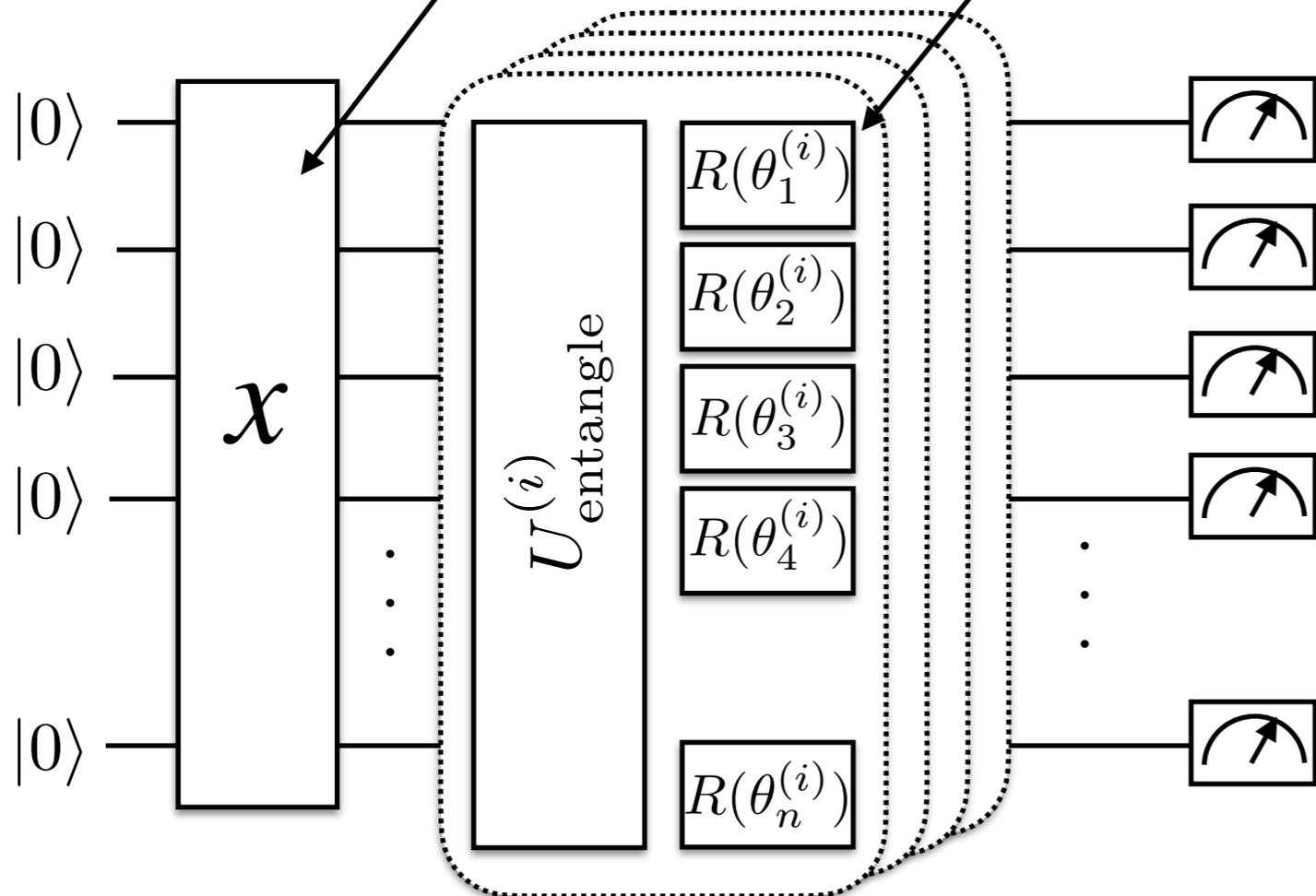
# 量子回路学習

- 量子系の時間発展は線形→テンソル積構造による非線形性

$$\begin{pmatrix} \psi_1^0(x_1) \\ \psi_1^1(x_1) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_k^0(x_k) \\ \psi_k^1(x_k) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_n^0(x_n) \\ \psi_n^1(x_n) \end{pmatrix} = \sum_{i_1 \dots i_n} \prod_k [\psi_k^{i_k}(x_k)] |i_1 \dots i_n\rangle \equiv |\Psi(\vec{x})\rangle$$

ユニタリー時間発展

$$W(\vec{\theta}) |\Psi(\vec{x})\rangle$$



測定

$$y(x, \theta) =$$

$$\langle \Psi(\vec{x}) | W^\dagger(\vec{\theta}) A W(\vec{\theta}) | \Psi(\vec{x}) \rangle$$

損失関数

$$L = (y_{\text{teacher}}^{(i)} - y(x, \vec{\theta}))^2$$

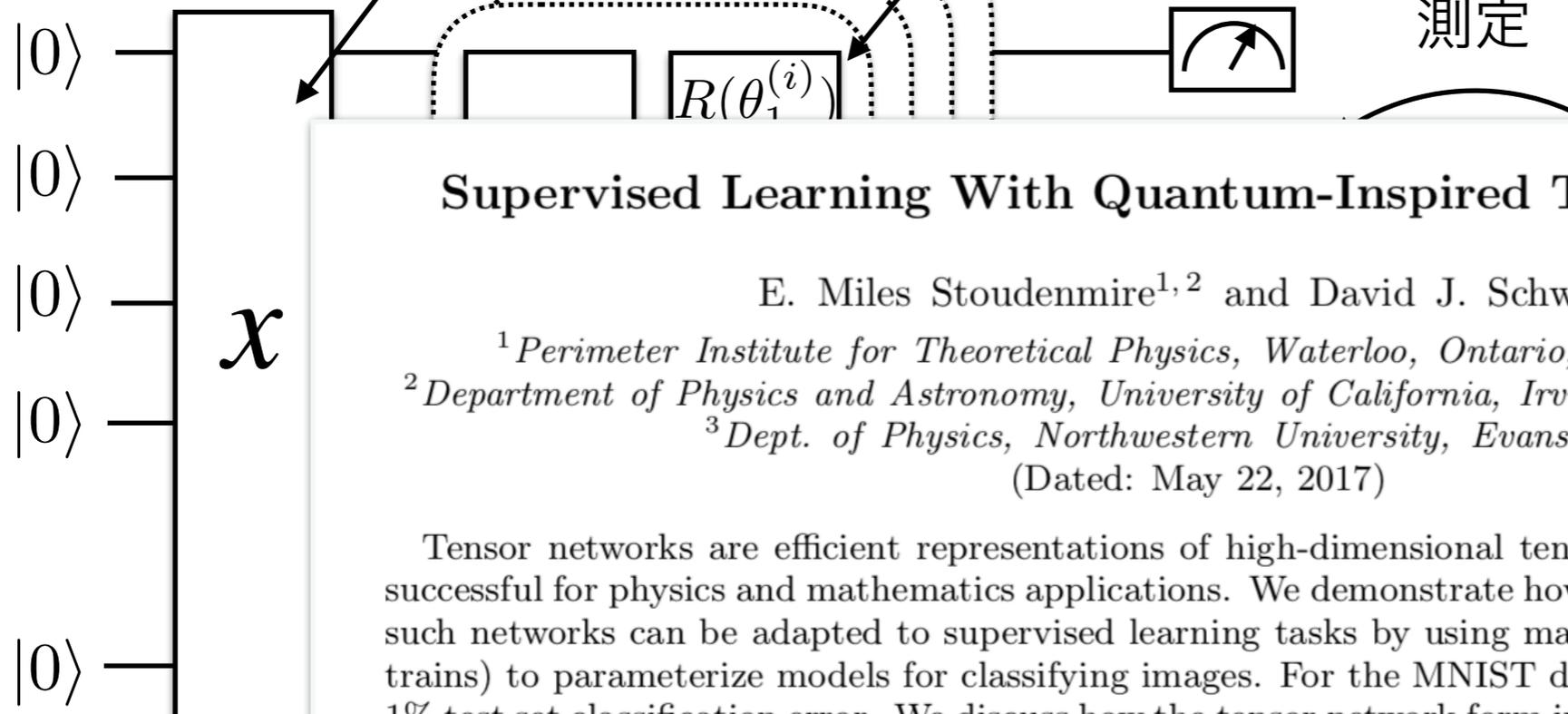
# 量子回路学習

- 量子系の時間発展は線形→テンソル積構造による非線形性

$$\begin{pmatrix} \psi_1^0(x_1) \\ \psi_1^1(x_1) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_k^0(x_k) \\ \psi_k^1(x_k) \end{pmatrix} \otimes \dots \otimes \begin{pmatrix} \psi_n^0(x_n) \\ \psi_n^1(x_n) \end{pmatrix} = \sum_{i_1 \dots i_n} \prod_k [\psi_k^{i_k}(x_k)] |i_1 \dots i_n\rangle \equiv |\Psi(\vec{x})\rangle$$

ユニタリー時間発展

$$W(\vec{\theta}) |\Psi(\vec{x})\rangle$$



## Supervised Learning With Quantum-Inspired Tensor Networks

E. Miles Stoudenmire<sup>1,2</sup> and David J. Schwab<sup>3</sup>

<sup>1</sup>Perimeter Institute for Theoretical Physics, Waterloo, Ontario, N2L 2Y5, Canada

<sup>2</sup>Department of Physics and Astronomy, University of California, Irvine, CA 92697-4575 USA

<sup>3</sup>Dept. of Physics, Northwestern University, Evanston, IL

(Dated: May 22, 2017)

Tensor networks are efficient representations of high-dimensional tensors which have been very successful for physics and mathematics applications. We demonstrate how algorithms for optimizing such networks can be adapted to supervised learning tasks by using matrix product states (tensor trains) to parameterize models for classifying images. For the MNIST data set we obtain less than 1% test set classification error. We discuss how the tensor network form imparts additional structure to the learned model and suggest a possible generative interpretation.

Advances in Neural Information Processing Systems 29, 4799 (2016)

$$|\Psi(\vec{x})\rangle$$

$$|\Psi(\vec{x}, \vec{\theta})\rangle^2$$

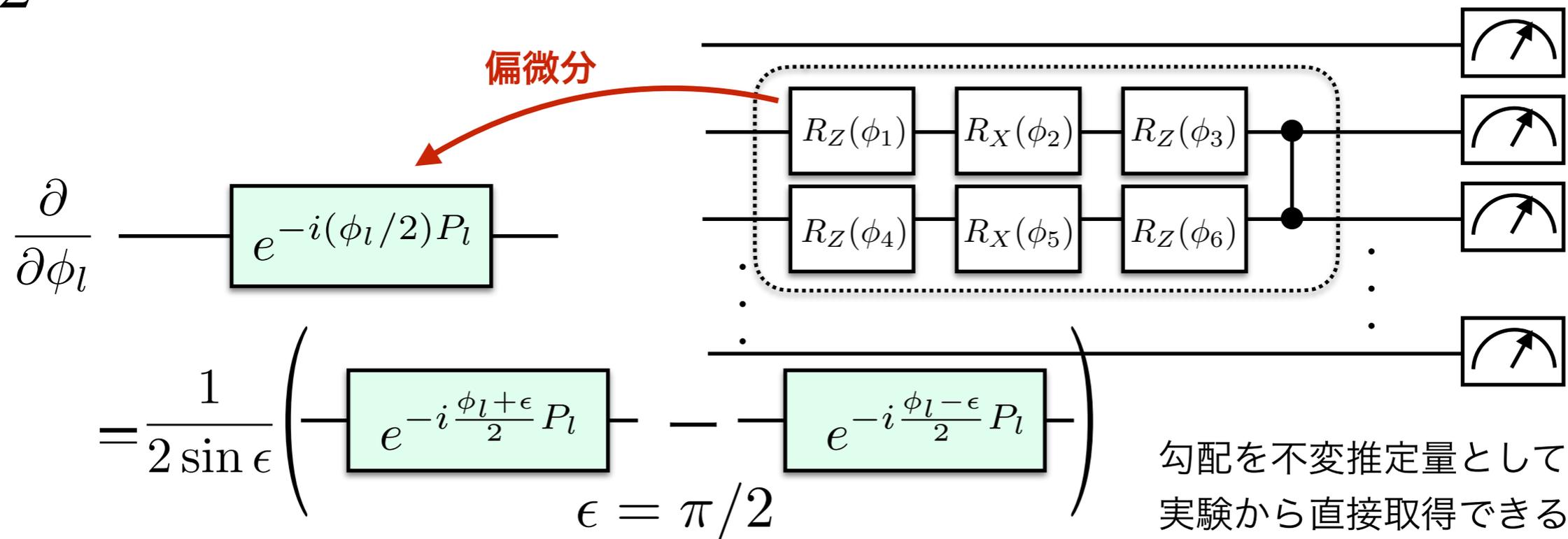
# 量子回路の解析的微分：parameter shift rule

$$\begin{aligned} & \frac{\partial}{\partial \phi} \langle \psi | e^{-iG\phi} H e^{iG\phi} | \psi \rangle \\ &= \frac{1}{2} \left[ \langle \psi | e^{-iG(\phi+\pi/4)} H e^{iG(\phi+\pi/4)} | \psi \rangle - \langle \psi | e^{-iG(\phi-\pi/4)} H e^{iG(\phi-\pi/4)} | \psi \rangle \right] \end{aligned}$$

# 量子回路の解析的微分：parameter shift rule

$$\frac{\partial}{\partial \phi} \langle \psi | e^{-iG\phi} H e^{iG\phi} | \psi \rangle$$

$$= \frac{1}{2} \left[ \langle \psi | e^{-iG(\phi+\pi/4)} H e^{iG(\phi+\pi/4)} | \psi \rangle - \langle \psi | e^{-iG(\phi-\pi/4)} H e^{iG(\phi-\pi/4)} | \psi \rangle \right]$$

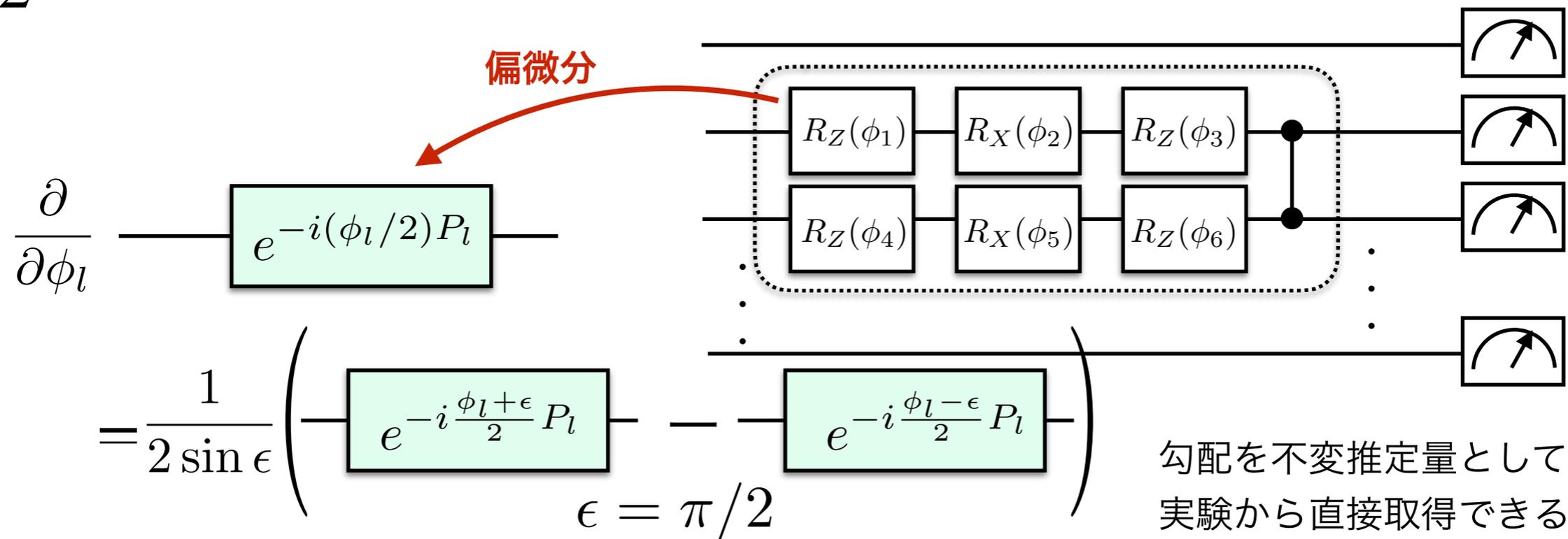


微小量ではなく任意の $\epsilon$ で成立

# 量子回路の解析的微分：parameter shift rule

$$\frac{\partial}{\partial \phi} \langle \psi | e^{-iG\phi} H e^{iG\phi} | \psi \rangle$$

$$= \frac{1}{2} \left[ \langle \psi | e^{-iG(\phi+\pi/4)} H e^{iG(\phi+\pi/4)} | \psi \rangle - \langle \psi | e^{-iG(\phi-\pi/4)} H e^{iG(\phi-\pi/4)} | \psi \rangle \right]$$



微小量ではなく任意の $\epsilon$ で成立

→現在、変分量子アルゴリズムの標準的なパラメータ更新法に。

TensorFlow Quantum  
by Google

PenneyLane  
by Xanadu



TensorFlow Quantum

XANADU  
PENNEY  
LANE

# 量子回路の解析的微分の一般論

$$\frac{\partial}{\partial \phi} \langle \psi | e^{-iG\phi} H e^{iG\phi} | \psi \rangle \quad G^2 = I \text{ (Pauli演算子など)}$$

$$= \frac{1}{2} \left[ \langle \psi | e^{-iG(\phi+\pi/4)} H e^{iG(\phi+\pi/4)} | \psi \rangle - \langle \psi | e^{-iG(\phi-\pi/4)} H e^{iG(\phi-\pi/4)} | \psi \rangle \right]$$

Banchi, Leonardo, and Gavin E. Crooks. "Measuring Analytic Gradients of General Quantum Evolution with the Stochastic Parameter Shift Rule.", *arXiv:2005.10299*

Crooks, Gavin E. "Gradients of parameterized quantum gates using the parameter-shift rule and gate decomposition." *arXiv:1905.13311* (2019).

$$C(x) = \text{Tr}(\hat{A} e^{\mathcal{Z}}[\hat{\rho}])$$

$$\mathcal{Z}[\hat{\rho}] := [i(\hat{H} + x\hat{V}), \hat{\rho}]$$

$$\frac{\partial e^{\mathcal{Z}}}{\partial x} = \int_0^1 ds e^{s\mathcal{Z}} \frac{\partial \mathcal{Z}}{\partial x} e^{(1-s)\mathcal{Z}}$$

$$\partial_x C(x) = \int_0^1 [C_+(x, s) - C_-(x, s)] ds$$

$$C_{\pm}(x, s) = \langle \phi | U_{\pm}(x, s)^{\dagger} \hat{A} U_{\pm}(x, s) | \phi \rangle$$

$$U_{\pm}(x, s) = e^{is(\hat{H}+x\hat{V})} e^{\pm i\frac{\pi}{4}\hat{V}} e^{i(1-s)(\hat{H}+x\hat{V})},$$

→ sを一様ランダムにサンプル

# 量子回路学習の実験

LETTER

nature  
International journal of science

https://doi.org/10.1038/s41586-019-0980-2

## Supervised learning with quantum-enhanced feature spaces

Vojtěch Havlíček<sup>1,2</sup>, Antonio D. Córcoles<sup>1\*</sup>, Kristan Temme<sup>1\*</sup>, Aram W. Harrow<sup>3</sup>, Abhinav Kandala<sup>1</sup>, Jerry M. Chow<sup>1</sup> & Jay M. Gambetta<sup>1</sup>

Machine learning and quantum computing are two technologies that each have the potential to alter how computation is performed to address previously untenable problems. Kernel methods for machine learning are ubiquitous in pattern recognition, with support vector machines (SVMs) being the best known method for classification problems. However, there are limitations to the successful solution to such classification problems when the feature space becomes large, and the kernel functions become computationally expensive to estimate. A core element in the computational speed-ups enabled by quantum algorithms is the exploitation of an exponentially large quantum state space through controllable entanglement and interference. Here we propose and experimentally implement two quantum algorithms on a superconducting processor. A key component in both methods is the use of the quantum state space as feature space. The use of a quantum-enhanced feature space that is only efficiently accessible on a quantum computer provides a possible path to quantum advantage. The algorithms solve a problem of supervised learning: the construction of a classifier. One method, the quantum variational classifier, uses a variational quantum circuit<sup>1,2</sup> to classify the data in a way similar to the method of conventional SVMs. The other method, a quantum kernel estimator, estimates the kernel function on the quantum computer and optimizes a classical SVM. The two methods provide tools for exploring the applications of noisy intermediate-scale quantum computers<sup>3</sup> to machine learning.

The intersection between machine learning and quantum computing has attracted considerable attention in recent years<sup>4-6</sup>. This has led to a number of recently proposed quantum algorithms<sup>1,2,7-9</sup>. Here we present two quantum algorithms that have the potential to run on near-term quantum devices. A suitable class of algorithms for such noisy devices employs short-depth circuits, because they are amenable to error-mitigation techniques that reduce the effect of decoherence<sup>10,11</sup>. There are convincing arguments to indicate that even very simple circuits are hard to simulate classically<sup>12,13</sup>. The algorithm we propose takes on the original problem of supervised learning: the construction of a classifier. For this problem, we are given data from a training set  $T$  and a test set  $S$  of a subset  $\Omega \subset \mathbb{R}^d$ . Both are assumed to be labelled by a map  $m: T \cup S \rightarrow \{+1, -1\}$  unknown to the algorithm. The training algorithm receives only the labels of the training data  $T$ . The goal is to infer an approximate map on the test set  $\tilde{m}: S \rightarrow \{+1, -1\}$  such that it agrees with high probability with the true map  $m(s) = \tilde{m}(s)$  on the test data  $s \in S$ . For this task to be meaningful it is assumed that there is a correlation between the labels given for training and the true map. A classical approach to this problem uses so-called support vector machines (SVMs)<sup>14</sup>. A quantum version of this approach has already been proposed in ref. <sup>15</sup>, where an exponential improvement can be achieved if data is provided in a coherent superposition. However, when data is provided in the conventional way, that is, from a classical computer, then the methods of ref. <sup>15</sup> do not yield this speed-up.

Here we propose two binary classifiers that process data that is provided classically and use the quantum state space as feature

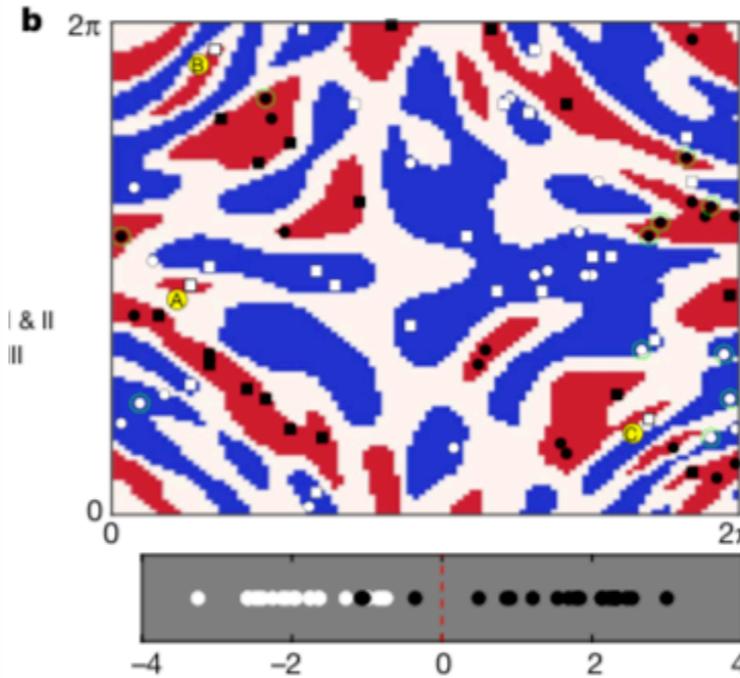
space. The data is mapped non-linearly to a quantum state  $\Phi: \mathbf{x} \in \Omega \rightarrow |\Phi(\mathbf{x})\rangle\langle\Phi(\mathbf{x})|$ ; see Fig. 1a. In the first approach we use a variational circuit as given in refs <sup>1,2,16,17</sup> followed by a binary measurement. Any binary measurement that classifies the data based on the probability of observing one outcome over the other implements a separating hyperplane in state space. Like an SVM, this approach constructs a linear decision function in feature space. The second approach builds on this observation and constructs the hyperplane using a classical SVM, only using the quantum computer to estimate the kernel function. This second approach inherits the performance guarantees from the classical SVM. We implement both classifiers on a superconducting quantum processor with five coupled superconducting transmons, only two of which are used in this work, as shown in Fig. 2a. In the experiment, we want to separate the question of whether the classifier can be implemented in hardware from the problem of choosing a suitable feature map for a practical dataset. The data that are classified here are chosen so that they can be classified with 100% success to verify the method. We experimentally demonstrate that this success ratio is achieved.

Training and classification with conventional SVMs is efficient when inner products between feature vectors can be evaluated efficiently<sup>14,18,19</sup>. Classifiers based on quantum circuits, such as the one presented in Fig. 2c, cannot provide a quantum advantage over a conventional SVM if the feature vector kernel  $K(\mathbf{x}, \mathbf{z}) = |\langle\Phi(\mathbf{x})|\Phi(\mathbf{z})\rangle|^2$  can be computed efficiently on a classical computer. For example, a classifier that uses a feature map that generates only product states can be evaluated in time  $\mathcal{O}(n)$  for  $n$  qubits. To obtain an advantage over classical approaches we need to implement a map based on circuits that are hard to simulate classically. Since quantum computers are not expected to be classically simulable, there exists a long list of (universal) circuit families we can choose from. Here we use a circuit that works well in our experiments and is not too deep. We define a feature map on  $n$ -qubits generated by the unitary  $U_{\Phi(\mathbf{x})} = U_{\Phi(\mathbf{x})} H^{\otimes n} U_{\Phi(\mathbf{x})} H^{\otimes n}$ , where  $H$  denotes the conventional Hadamard gate and

$$U_{\Phi(\mathbf{x})} = \exp\left(i \sum_{S \subset [n]} \phi_S(\mathbf{x}) \prod_{i \in S} Z_i\right)$$

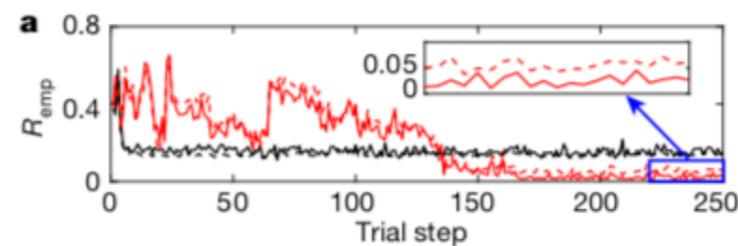
is a diagonal gate in the Pauli Z-basis; see Fig. 1b. This circuit acts on the initial state  $|0\rangle^n$ . We use the coefficients  $\phi_S(\mathbf{x}) \in \mathbb{R}$  to encode the data  $\mathbf{x} \in \Omega$ . In general any diagonal unitary  $U_{\Phi(\mathbf{x})}$  can be used if it can be implemented efficiently. This is, for instance, the case when only weight  $|S| \leq 2$  interactions are considered. The exact evaluation of the inner product between two states generated from a similar circuit with only a single diagonal layer  $U_{\Phi(\mathbf{x})}$  is #P-hard<sup>20</sup>. Nonetheless, in the experimentally relevant context of additive error approximation, simulation of a single-layer preparation circuit can be achieved efficiently classically by uniform sampling<sup>21</sup>. We conjecture that the additive error approximation of inner products generated from circuits with two Hadamard layers and diagonal gates is hard classically; see Supplementary Information for a discussion.

## 教師データ (hardware-efficient dataset)

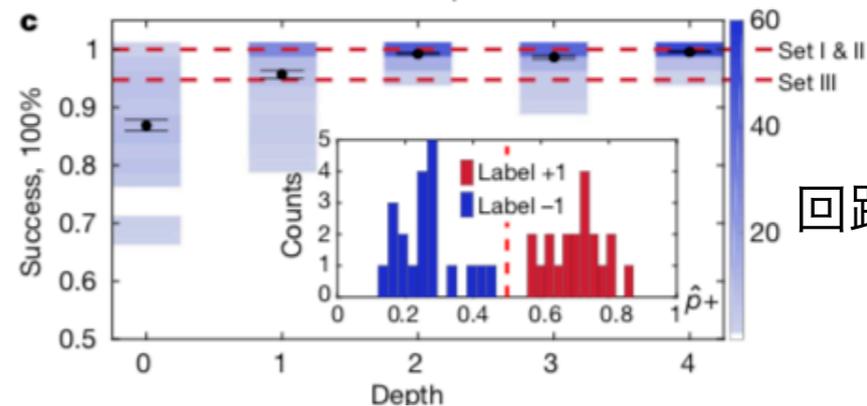


量子コンピュータ  
実機から生成した  
データセット

学習結果



損失関数



回路深さと成功確率

<sup>1</sup>IBM T. J. Watson Research Center, Yorktown Heights, NY, USA. <sup>2</sup>Department of Computer Science, University of Oxford, Wolfson Building, Parks Road, Oxford, UK. <sup>3</sup>Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, MA, USA. \*e-mail: adcorcol@us.ibm.com; kptemme@ibm.com

# NISQを用いた量子機械学習

*"Quantum **autoencoders** for efficient compression of quantum data."*

Romero, Jonathan, Jonathan P. Olson, and Alan Aspuru-Guzik.  
Quantum Science and Technology 2.4 (2017): 045001.

*"Variational quantum generators: **Generative adversarial** quantum machine learning for continuous distributions."*

Romero, Jonathan, and Alan Aspuru-Guzik.  
arXiv preprint arXiv:1901.00848 (2019).

*"Variational quantum circuits for **deep reinforcement learning**."*

Chen, Samuel Yen-Chi, et al. "arXiv preprint arXiv:1907.00397 (2019).

*"Quantum **kitchen sinks**: An algorithm for machine learning on near-term quantum computers."*

Wilson, C. M., et al. arXiv preprint arXiv:1806.08321 (2018).

*"Experimental quantum **kernel machine learning** with nuclear spins in a solid."*

Kusumoto, Takeru, et al.  
arXiv preprint arXiv:1911.12021 (2019).

*"**Transfer learning** in hybrid classical-quantum neural networks."*

Mari, Andrea, et al. "arXiv preprint arXiv:1912.08278 (2019).

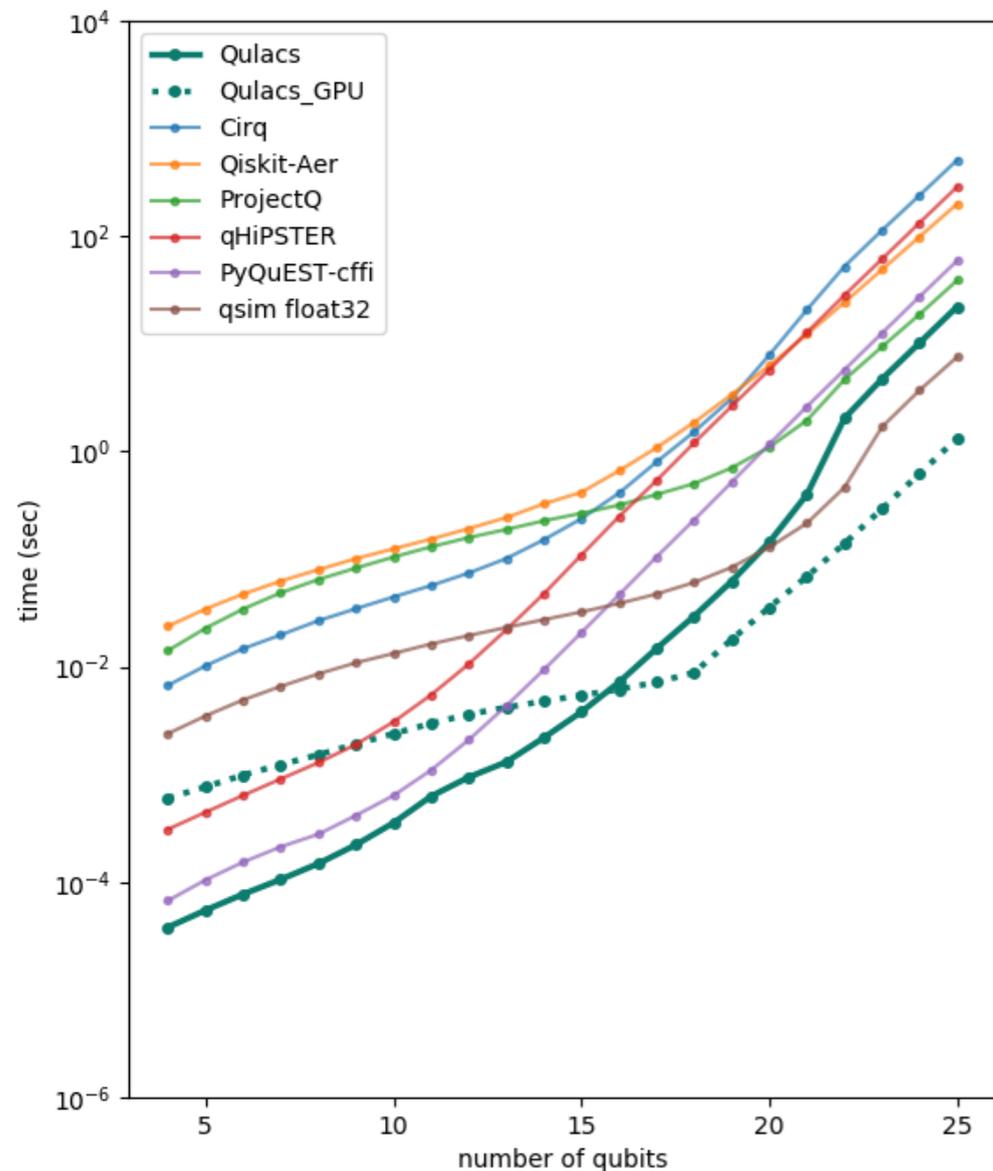
**Review** *"Parameterized quantum circuits as machine learning models."*

Benedetti, Marcello, et al.  
Quantum Science and Technology 4.4 (2019): 043001.

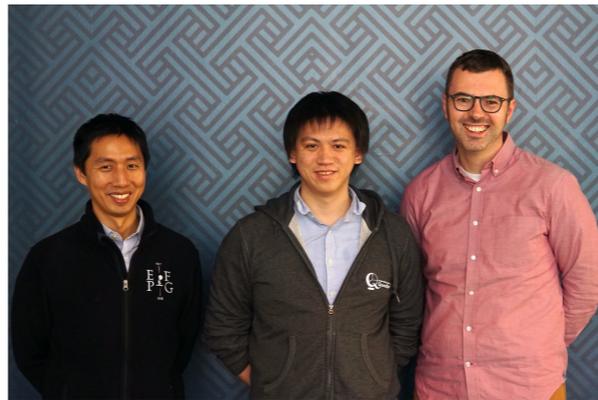
# 開発ツール：Qulacs

## 世界最速pythonベースのオープンソースシミュレータ

- google cirqのバックエンドとしても利用可能  
→cirq-qulacs ×50~500倍
- ZAPATA社が提供する orchestra (tequila)のバックエンドとしても利用されている
- PanneyLaneのバックエンドとしても利用可能  
→panneylane-qulacs
- Yao(julia)やQiboなどシミュレータのベンチマークに



<https://medium.com/より>



[https://medium.com/@yan\\_95915/development-of-fast-cpu-and-gpu-based-quantum-computer-simulator-for-cirq-eaee70fabeb2](https://medium.com/@yan_95915/development-of-fast-cpu-and-gpu-based-quantum-computer-simulator-for-cirq-eaee70fabeb2)

$$H = -\chi(\omega)X(t) + \dots$$

$$G = e^{-\frac{i}{\hbar} \int_0^t H dt}$$

$$L = \langle H \rangle_{U\omega} + \dots$$

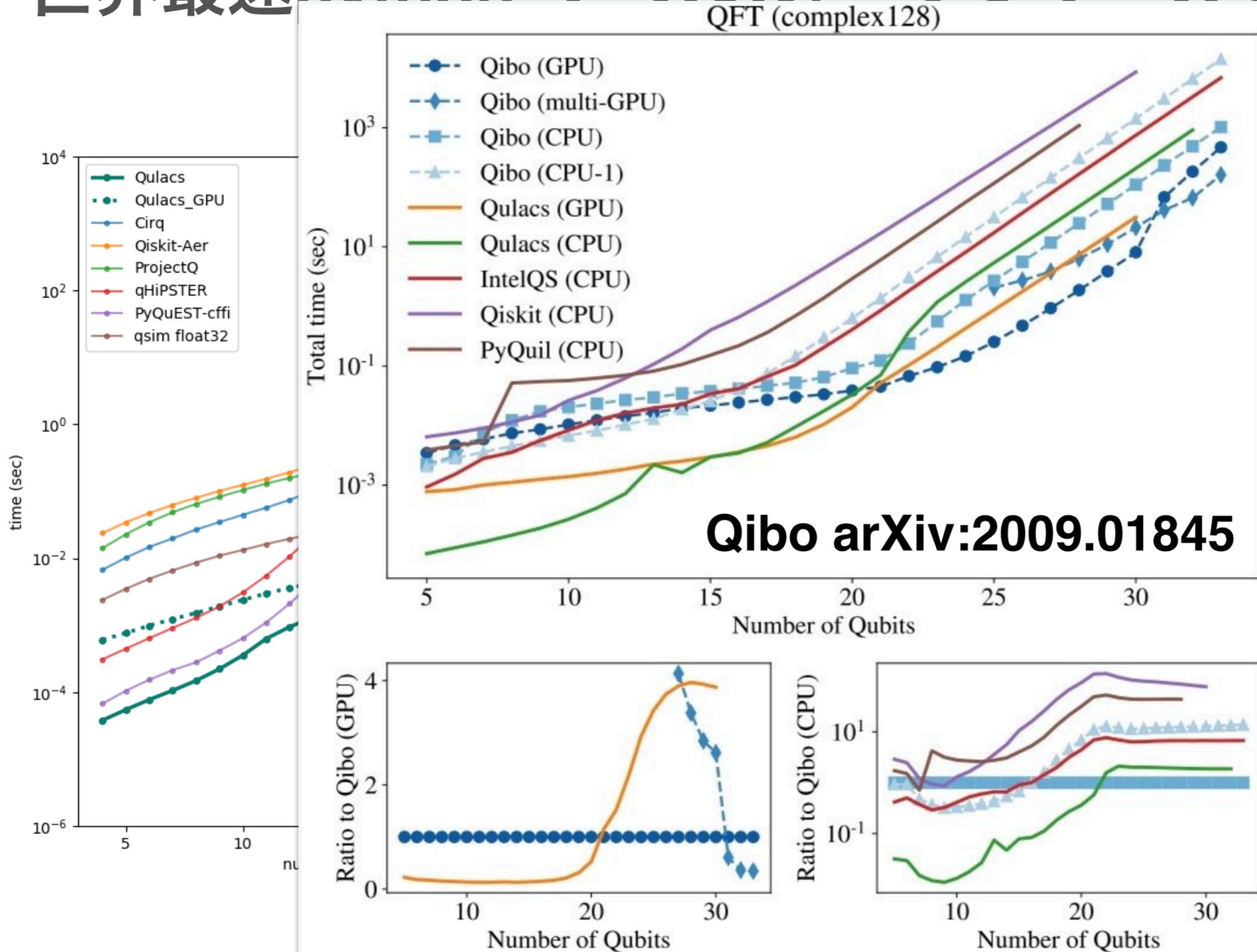
```

a = tq.Variable("a")
U = tq.gates.Ry(angle=(-a**2).apply(tq.numpy.exp)*pi, target=0)
U += tq.gates.X(target=1, control=0)
H = tq.QubitHamiltonian.from_string("-1.0*X(0)X(1)+0.5Z(0)+Y(1)")
E = tq.ExpectationValue(H=H, U=U)
dE = tq.grad(E, "a")
objective = E + (-dE**2).apply(tq.numpy.exp)
result = tq.minimize(method="phoenics", objective=objective)
  
```

<https://github.com/aspuru-guzik-group/tequila>

# 開発ツール：Qulacs

世界最速pythonベースのオープンソースシミュレータ

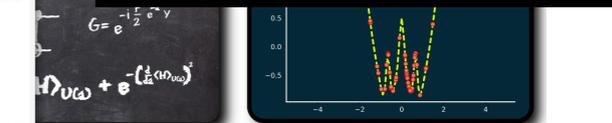


でも利用可能

a (tequila)のバツ

でも利用可能

タのベンチマークに



```

variable("a")
tes.Ry(angle=(-a**2).apply(tq.numpy.exp)*pi, target=0)
ates.X(target=1, control=0)
bitHamiltonian.from_string("-1.0*X(0)X(1)+0.5Z(0)+Y(1)")
pectationValue(H=H, U=U)
rad(E, "a")
= E + (-dE**2).apply(tq.numpy.exp)
tq.minimize(method="phoetics", objective=objective)

```

<https://github.com/aspuru-guzik-group/tequila>

# まとめと今後の課題

- ・ 2014年以降、量子系の制御技術が飛躍的に向上してきた  
→ '14 5Q, '15 9Q, '17 22Q, '19 53Q ...
- ・ このような複雑に制御された量子ダイナミクスが従来コンピュータでシミュレーションが難しい領域までできている  
→ **究極的に複雑な人工物を物理する**
- ・ NISQマシンを用いた量子機械学習：**量子レザバー計算、量子回路学習**
- ・ NISQマシンを用いて、未知の結果を得ることはできていない  
→ **このような複雑な物理系を有意義に活用する方法**
- ・ 物理と機械学習の距離、量子情報と機械学習、量子情報と物理の距離がそれぞれ近くなっている  
→ **なにかありそう**