量子計算の古典検証

三菱電機 情報技術総合研究所
情報セキュリティ技術部
JST ACT-X
水谷明博

2022/8/2 耐量子計算機暗号と量子情報の数理

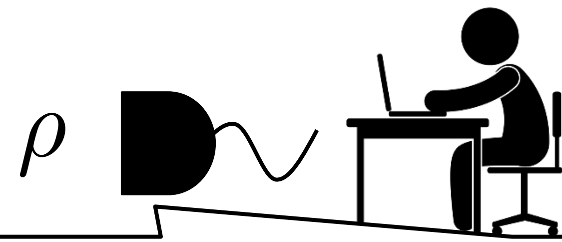
量子計算の古典検証

Mahadev, FOCS 2018
論文：[1804.01082v2] Classical Verification of Quantum Computations (arxiv.org)
動画：Classical Verification of Quantum Computations - YouTube

Morimae-Fitzsimonsプロトコルの詳細 (竹内さんのスライド)

[T. Morimae, arXiv:2003.10712]

- サーバが送った答えがYESだった場合



1. の確率で、ペア(*i,j*)選ぶ。
2. *1/2*の確率で

$$Z_i \otimes Z_j \quad \text{or} \quad X_i \otimes X_j$$

を選ぶ。
3. 前者の時は、*i*
後者の時は、*i*
4. 測定結果の足

となれば受理する。

X,Z測定をProverに委託できればOK (Measurementプロトコル)

Measurementプロトコル

測定基底$h \in \{Z,X\}^n$

古典Verifier (V)

古典メッセージ

量子多項式時間 (QPT)Prover (P)

受理
or
拒否

$\boldsymbol{m}$ (Vが得るX,Z測定結果)
$\boldsymbol{D_{P,h}}$(mの確率分布)

n Qubit状態

Pr[VがPを受理]=1 ⇒$\exists\sigma, \forall h,$ $\boldsymbol{D_{P,h} \approx D_{\sigma,h}}$

$\sigma \longrightarrow$ h

$$m \leftarrow \begin{cases} D_{P,h} \\ D_{\sigma,h} \end{cases} \longrightarrow \boxed{\text{QPT}} \longrightarrow$$

0を出力する確率は
neglしかずれない

LWE仮定：LWE問題がQPTで解けない

量子計算の古典検証プロトコル

任意のBQPのインスタンスxは2Localハミルトニアン問題に帰着

$H_x = d_1 IZZI + d_2 \boldsymbol{IIXX} + d_3 IIII$のエネルギーが高いか低いかを判定

① VはランダムにH_xの項を選ぶ → $\boldsymbol{IIXX}$ ($h = XX$)

② Vは量子状態σを基底hで測定し($m = m_1 m_2$)を出力

③ $(-1)^{m_1 \oplus m_2} = -\mathrm{sign}(d_2)$ならPを受理

ポストホック検証プロトコル

量子計算の古典検証プロトコル

任意のBQPのインスタンスxは2Localハミルトニアン問題に帰着
$H_x = d_1 IZZI + d_2 \boldsymbol{IIXX} + d_3 IIII$のエネルギーが高いか低いかを判定

① VはランダムにH_xの項を選ぶ → $\boldsymbol{IIXX}$ ($h = XX$)

② Measurementプロトコルを実行し、Vは($m = m_1 m_2$)を出力

③ $(-1)^{m_1 \oplus m_2} = -\mathrm{sign}(d_2)$ならPを受理

ある量子状態σをhで測定した分布と区別つかない

<u>Pの答えが正しい場合</u>：

Measurementプロトコルのσをhistory stateとすれば
$\Pr[\text{受理}] \geq 1 - \mathrm{negl}$

<u>Pの答えが正しくない場合</u>：

任意のσに対してH_xのエネルギーが高いので
$\Pr[\text{受理}] \leq \frac{1}{2} + \mathrm{negl}$　($\frac{1}{2}$はmを得るラウンドを選ばない確率)

Measurementプロトコルのポイント

Pr[VがPを受理]=1 ⇒$\exists\boldsymbol{\sigma}, \forall \boldsymbol{h}, \boldsymbol{D_{P,h}} \approx \boldsymbol{D_{\sigma,h}}$

$\forall \boldsymbol{h}, \exists\boldsymbol{\sigma}, \boldsymbol{D_{P,h}} \approx \boldsymbol{D_{\sigma,h}}$
だと何が問題？

<u>ハミルトニアンのエネルギーが測れないので古典検証できない</u>

$H_x = \sum_i H_x^i$のエネルギー$\mathrm{Tr}[H_x\sigma]$を測るため、ランダムなH_x^iに対応する基底hでMeasurementプロトコルを実行していた。$\mathrm{Tr}[H_x\sigma] = \sum_i \mathrm{Tr}[\boldsymbol{H_x^i}\sigma]$

基底hに依存.
σはh無依存がマスト

Measurementプロトコル

測定基底$h \in \{Z, X\}^n$

古典Verifier (V)

古典メッセージ

量子多項式時間 (QPT)Prover (P)

受理 or 拒否

$\boldsymbol{m}$ (Vが得るX,Z測定結果)
$\boldsymbol{D_{P,h}}$(mの分布)

Part 1

Pr[VがPを受理]=1 ⇒$\exists \sigma, \forall h, \boldsymbol{D_{P,h}} \approx \boldsymbol{D_{\sigma,h}}$

$\sigma \longrightarrow$ h

$$m \leftarrow \begin{cases} D_{P,h} \\ D_{\sigma,h} \end{cases} \longrightarrow \boxed{\text{QPT}} \longrightarrow$$

0を出力する確率はneglしかずれない

Part 2

LWE仮定：LWE問題がQPTで解けない

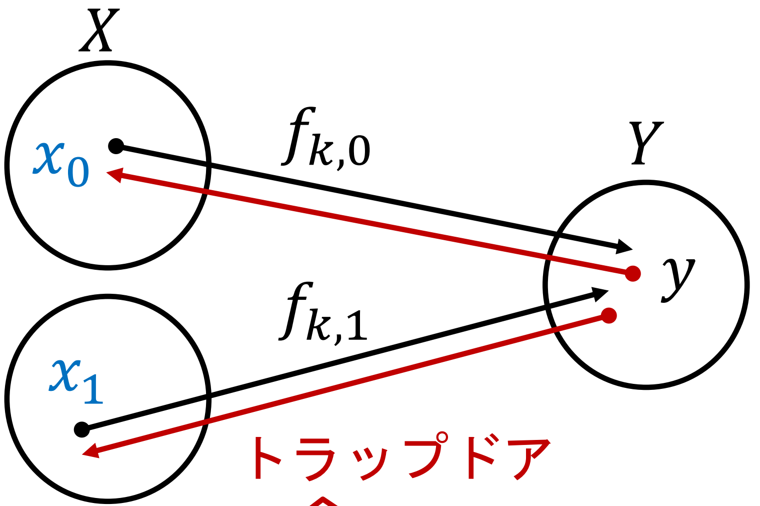
Trapdoor claw-free 関数族

LWE仮定から関数族FとGが構成可能

関数族 $F = \{f_{k,0}, f_{k,1}\}_k$

1. **2 to 1**:
   $f: X \to Y$
   像yの原像(x_0, x_1)が存在して
   $y = f_{k,0}(x_0) = f_{k,1}(x_1)$

2. **Claw-free性:**
   QPT Pは像yから2つの原像(x_0, x_1)を求めることが難しい

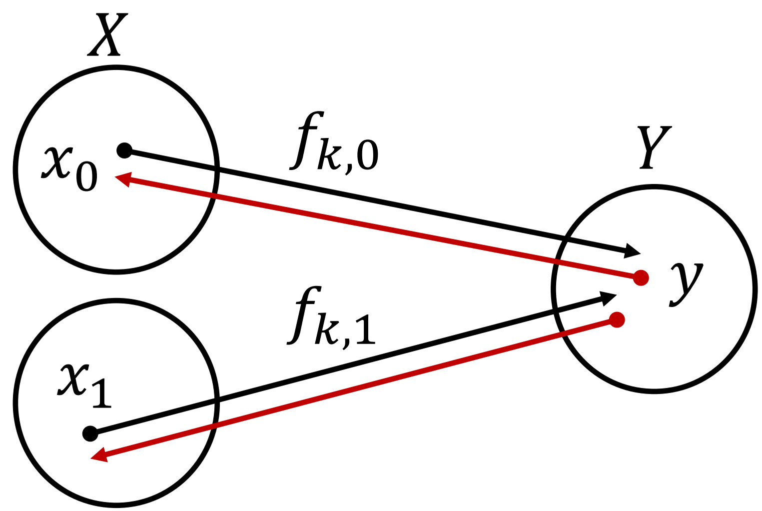


$x_b = INV_F(t_k, y, b)$
$b \in \{0,1\}$

Trapdoor claw-free 関数族

LWE仮定から関数族FとGが構成可能

関数族 $F = \{f_{k,0}, f_{k,1}\}_k$



3. Adaptive hardcoreビット性1:

QPT Pは公開鍵kから
$(y, b, x_b, r, d \neq 0)$
を出力することが難しい.
$y = f_{k,b}(x_b), r = d \cdot (x_0 \oplus x_1) \in \{0,1\}$

4. Adaptive hardcoreビット性2:

QPT Pは公開鍵kから$d \neq 0$に対して$r = d \cdot (x_0 \oplus x_1) \in \{0,1\}$
を求めることが難しい

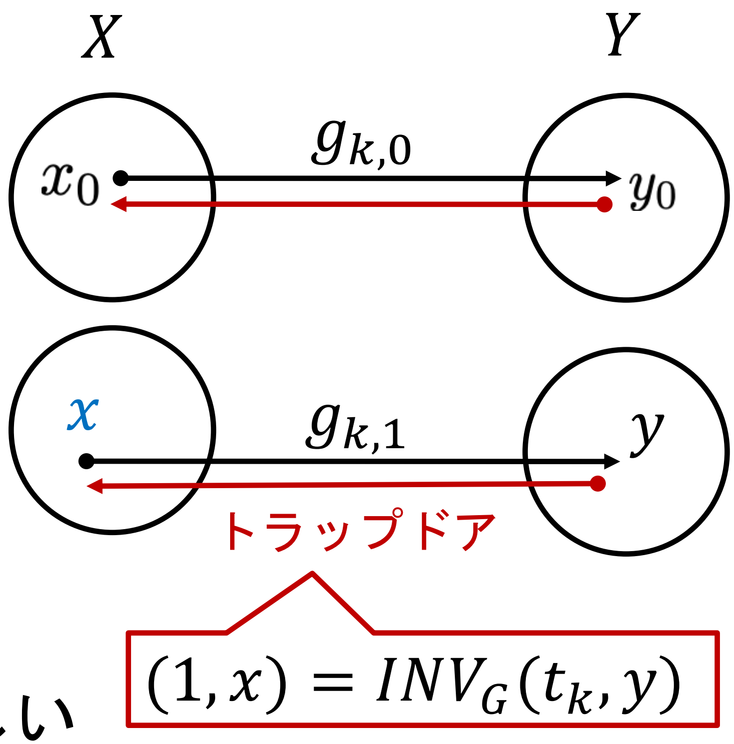
Trapdoor claw-free 関数族

LWE仮定から関数族FとGが構成可能

関数族 $G = \{g_{k,0}, g_{k,1}\}_k$

1. **1 to 1**:
   $g: X \to Y$
   像yの原像xが一意に
   存在して$y = g_{k,b}(x)$

2. **Injective invariance**:
   QPT Pは公開鍵kから
   FかGかを識別することは難しい

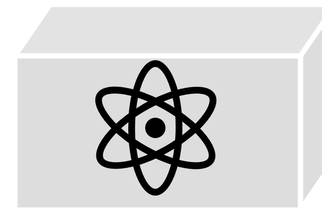


Measurementプロトコル (with Honest P)

$h = Z \rightarrow k$とトラップドアt_kをGから生成
$h = X \rightarrow k$とトラップドアt_kをFから生成

公開鍵k →

$|\varphi\rangle = \sum_{b=0}^{1} \alpha_b |b\rangle$



← ビット列y

$$\sum_{b,x} \alpha_b |b\rangle |x\rangle \underline{|g_{k,b}(x)\rangle}$$

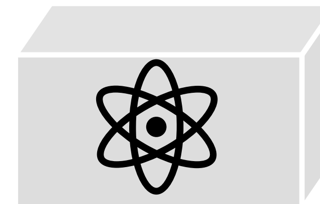
Z測定

(計算基底の測定)

Measurementプロトコル (with Honest P)

$h = Z \rightarrow k$とトラップドアt_kをGから生成
$\boldsymbol{h = X} \rightarrow k$とトラップドア$t_k$を$\boldsymbol{F}$から生成

$|\varphi\rangle = \sum_{b=0}^{1} \alpha_b |b\rangle$



公開鍵k →

← ビット列y

$$\sum_{b,x} \alpha_b |b\rangle |x\rangle \underline{|\boldsymbol{f}_{k,b}(x)\rangle}$$

Z測定

(計算基底の測定)

Measurementプロトコル (with Honest P)

$h = Z \rightarrow k$とトラップドアt_kをGから生成
$\boldsymbol{h} = \boldsymbol{X} \rightarrow k$とトラップドア$t_k$を$\boldsymbol{F}$から生成

$|\varphi\rangle = \sum_{b=0}^{1} \alpha_b |b\rangle$

公開鍵k →

← ビット列y

$\sum_{b,x} \alpha_b |b\rangle |x\rangle \underbrace{|f_{k,b}(x)\rangle}_{\text{Z測定}}$

選択 {**テスト**,アダマール}

Z基底で測定して →

$\sum_{b=0}^{1} \alpha_b \underbrace{|b\rangle |x_{b,y}\rangle}_{\text{Z測定}}$

← Z基底結果(b, x)

(b, x)がyの正しい原像であれば受理

Measurementプロトコル (with Honest P)

$\boldsymbol{h = Z}$ $\rightarrow k$とトラップドアt_kを$\boldsymbol{G}$から生成
$h = X \rightarrow k$とトラップドアt_kをFから生成

$|\varphi\rangle = \sum_{b=0}^{1} \alpha_b |b\rangle$

公開鍵k →

← ビット列y

$\sum_{b,x} \alpha_b |b\rangle |x\rangle \underline{|\boldsymbol{g}_{k,b}(x)\rangle}$

Z測定

選択 {**テスト**,アダマール}

Z基底で測定して →

$\underline{|b\rangle |x_{b,y}\rangle}$

Z測定

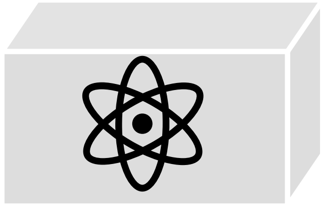
← Z基底結果(b, x)

(b, x)がyの正しい原像であれば受理

Measurementプロトコル (with Honest P)

$h=Z\rightarrow k$とトラップドアt_kをGから生成
$h=X\rightarrow k$とトラップドアt_kをFから生成

$|\varphi\rangle = \sum_{b=0}^{1} \alpha_b |b\rangle$



公開鍵k →

← ビット列$\boldsymbol{y}$

$\sum_{b,x} \alpha_b |b\rangle|x\rangle\underline{|\boldsymbol{g_{k,b}(x)}\rangle}$

Z測定

選択 {アダマール}

X基底で測定して →

$\underline{|b\rangle|x_{b,y}\rangle}$

X測定

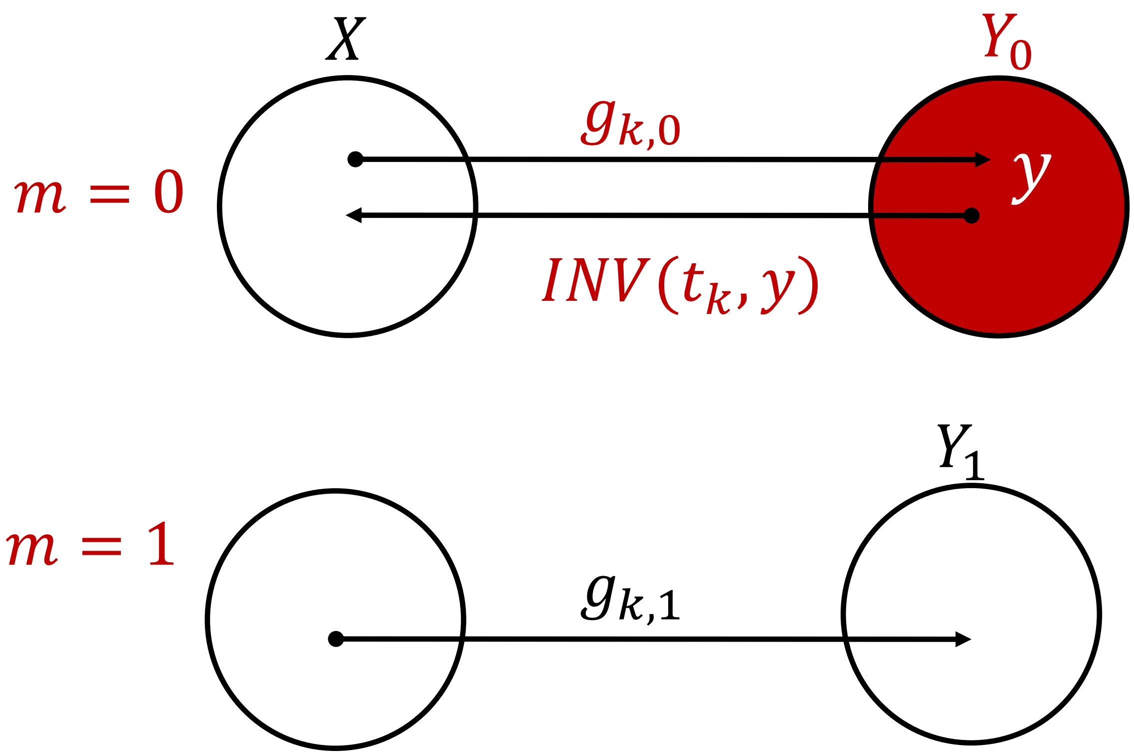
← X測定結果(b,d)

(b,d)を無視
$(\boldsymbol{m},x) = INV_G(t_k,\boldsymbol{y})$
mをZ測定結果として出力

$D_{P,Z}(m=0) = \Pr[y\in Y_0] = |\alpha_0|^2$
$= D_{|\varphi\rangle,Z}(m=0)$

Trapdoor claw-free 関数族

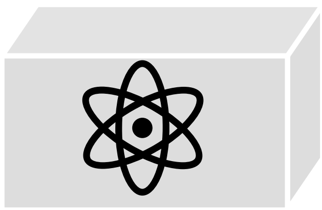
関数族 $G = \{g_{k,0}, g_{k,1}\}_k$



Measurementプロトコル (with Honest P)

$h = Z \rightarrow k$とトラップドアt_kをGから生成
$\boldsymbol{h} = \boldsymbol{X} \rightarrow k$とトラップドア$t_k$を$\boldsymbol{F}$から生成

$|\varphi\rangle = \sum_{b=0}^{1} \alpha_b |b\rangle$



公開鍵k →

← ビット列y

$$\sum_{b,x} \alpha_b |b\rangle |x\rangle \underline{|\boldsymbol{f}_{k,b}(x)\rangle}$$

Z測定

選択 {**アダマール**}

X基底で測定して →

$$\sum_{b=0}^{1} \alpha_b |b\rangle \underline{|x_{b,y}\rangle}$$

X測定(d)

← X測定結果(b,d)

$$\langle b| \boxed{X^{d\cdot(x_0 \oplus x_1)} \boldsymbol{H} |\boldsymbol{\varphi}\rangle}$$

$$\boldsymbol{m} = b \oplus d \cdot (x_0 \oplus x_1)$$

$$\begin{cases} x_0 = INV_F(t_k, 0, y) \\ x_1 = INV_F(t_k, 1, y) \end{cases}$$

をX測定結果として出力

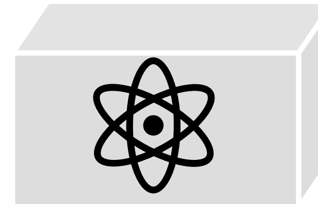
$$\boldsymbol{D}_{\boldsymbol{P},\boldsymbol{X}} = \Pr[b \oplus d \cdot (x_0 \oplus x_1)] = \boldsymbol{D}_{|\boldsymbol{\varphi}\rangle,\boldsymbol{X}}$$

Measurementプロトコル

測定基底$h = Z \text{ or } X$

古典V

古典メッセージ



QPT P

{テスト or アダマール}

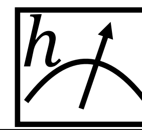
受理
or
拒否

m (hの測定結果)
$D_{P,h}$(mの分布)

Part 1

一般のPに対して

Pr[VがPを受理]=1 ⇒$\exists\sigma, \forall h, D_{P,h} \approx D_{\sigma,h}$

$\sigma \longrightarrow$ 

$$m \leftarrow \begin{cases} D_{P,h} \\ D_{\sigma,h} \end{cases} \longrightarrow \boxed{\text{QPT}} \longrightarrow$$

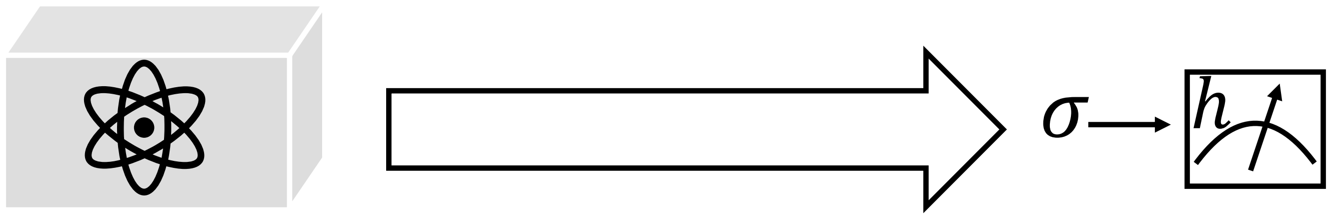
0を出力する確率は
neglしかずれない

Part 2

LWE仮定：LWE問題がQPTで解けない

Part 2の概略

一般のP



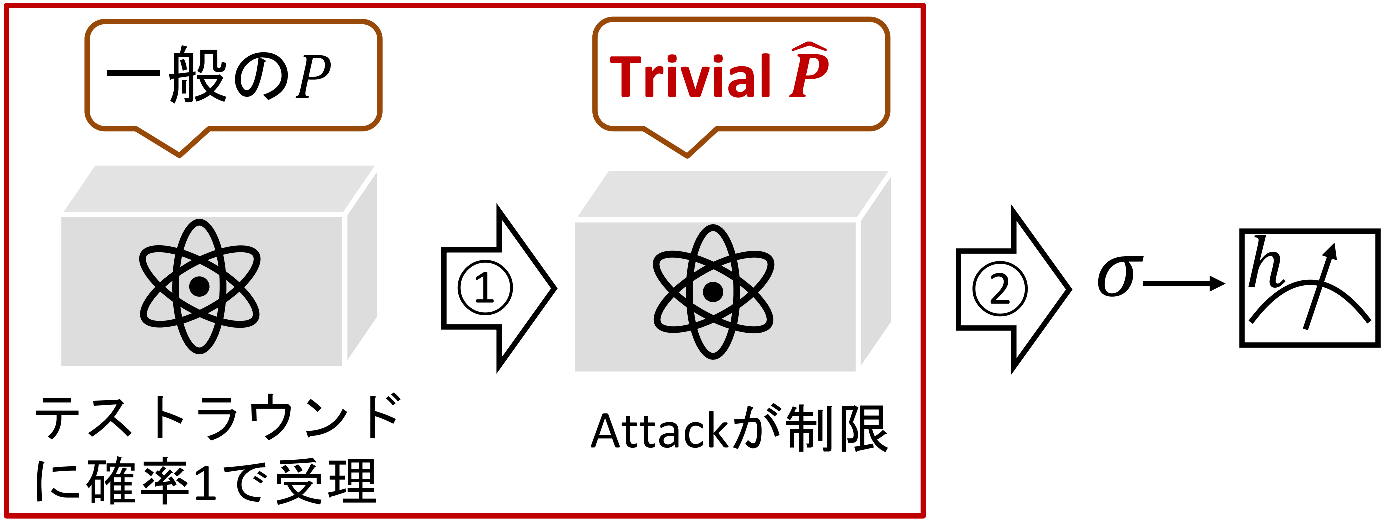
テストラウンド
に確率1で受理

$$\boldsymbol{D_{P,h}} \approx \boldsymbol{D_{\sigma,h}}$$

アダマールラウンドでVが
測定結果mを出力する分布

状態σを基底$h \in \{Z, X\}$
で測定した分布

Part 2の概略



$$\boldsymbol{D_{P,h}} \approx \boldsymbol{D_{\hat{P},h}} \approx D_{\sigma,h}$$

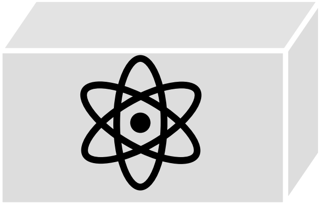
① 一般のPに対してTrivial $\hat{P}$が存在して$D_{P,h} \approx D_{\hat{P},h}$

② Trivial $\hat{P}$に対してある量子状態σが存在して$D_{\hat{P},h} \approx D_{\sigma,h}$

一般のPのふるまい(ステップ①)

$h = Z \to k$とトラップドアt_kをGから生成
$h = X \to k$とトラップドアt_kをFから生成

公開鍵k



$U_0(|e\rangle \otimes |k\rangle)$

QPTで作れる任意のユニタリ

ビット列$\boldsymbol{y}$ (関数の像)

選択{テスト}

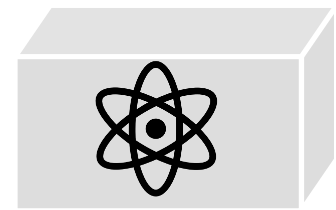
Z基底で測定して

Z基底結果$(\boldsymbol{b}, \boldsymbol{x})$ ($\boldsymbol{y}$の原像)

一般のPのふるまい(ステップ①)

$h=Z\rightarrow k$とトラップドアt_kをGから生成
$h=X\rightarrow k$とトラップドアt_kをFから生成

公開鍵k →



$|b\rangle_1, |x\rangle_2, |y\rangle_3$を持ち$(b,x,y)$はZ測定結果とみなせる

$U_0(|e\rangle\otimes|k\rangle)$

← ビット列**y (関数の像)**

$|y\rangle_3$をZ測定

選択{**テスト**}

テストラウンドのユニタリ

Z基底で測定して →

U_Tをかけて
$|b\rangle_1, |x\rangle_2$をZ測定

← Z基底結果**(b, x) (yの原像)**

一般のPのふるまい (ステップ①)

$h=Z\rightarrow k$とトラップドアt_kをGから生成
$h=X\rightarrow k$とトラップドアt_kをFから生成

$P\Rightarrow$ 2つのユニタリ(U_0, U)
で特徴づけられる

公開鍵k

$\boldsymbol{U_0}(|e\rangle\otimes|k\rangle)$

ビット列y

$|y\rangle_3$をZ測定

選択{**アダマール**}

アダマールラウンドのユニタリ

X基底で測定して

$\boldsymbol{U_H}$をかけて
$|b\rangle_1, |x\rangle_2$をX測定

X基底結果(b, d)

一般のPのふるまい (ステップ①)

$h = Z \rightarrow k$とトラップドアt_kをGから生成
$h = X \rightarrow k$とトラップドアt_kをFから生成

公開鍵k

$U_0(|e\rangle \otimes |k\rangle)$

ビット列y

$|y\rangle_3$をZ測定

選択 {テスト}

テストラウンドのユニタリ

Z基底で測定して

U_Tをかけて
$|b\rangle_1, |x\rangle_2$をZ測定

Z基底結果(b, x)

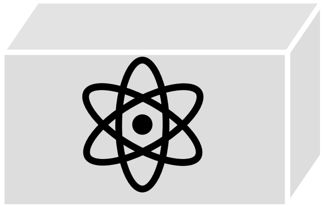
一般のPのふるまい(ステップ①)

$h=Z\rightarrow k$とトラップドアt_kをGから生成
$h=X\rightarrow k$とトラップドアt_kをFから生成

$P\Rightarrow$ 2つのユニタリ(U_0,U)
で特徴づけられる

$P(U_0,U)$

公開鍵k



$\boldsymbol{U}_0(|e\rangle\otimes|k\rangle)$

ビット列y

$|y\rangle_3$をZ測定

選択{**アダマール**}

$\boldsymbol{U_H U_T}^{\dagger}$
$/\!/$

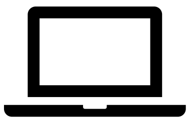
X基底で測定して

$\boldsymbol{U}$をかけて
$|b\rangle_1,|x\rangle_2$をX測定

X基底結果(b,d)

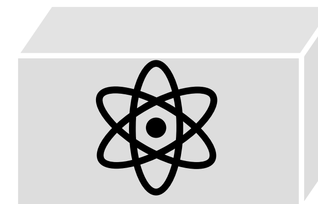
Trivial $\hat{P}$の定義

$h = Z \rightarrow k$とトラップドアt_kをGから生成
$h = X \rightarrow k$とトラップドアt_kをFから生成



$P \Rightarrow$ 2つのユニタリ(U_0, U)で特徴づけられる

$P(U_0, U)$



公開鍵k →

$U_0(|e\rangle \otimes |k\rangle)$

← ビット列y

$|y\rangle_3$をZ測定

選択{アダマール}

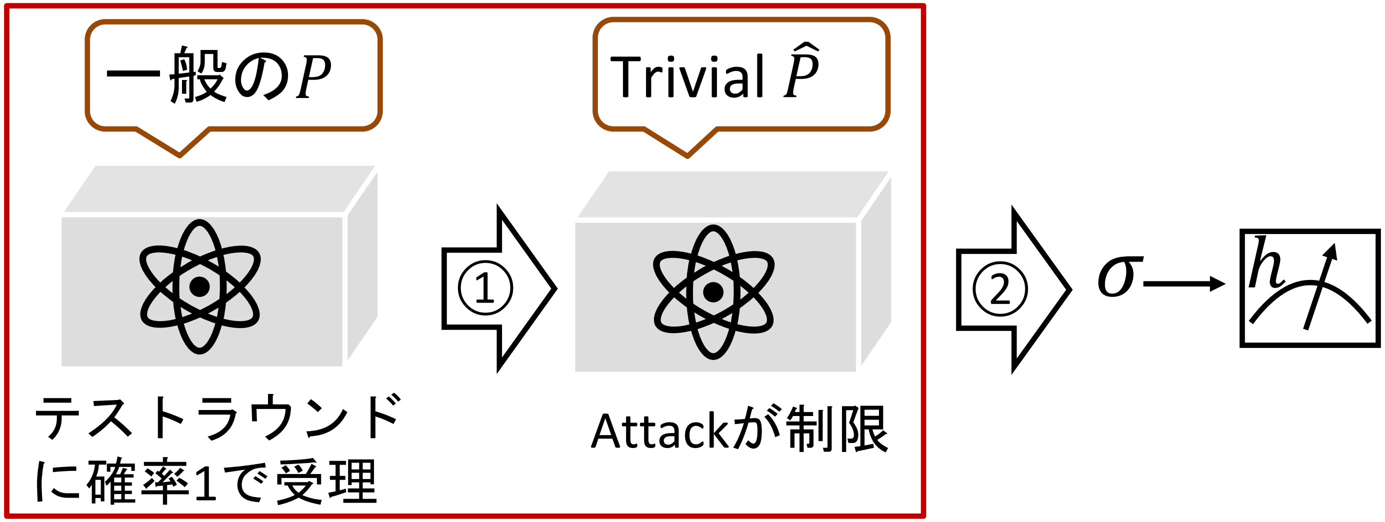
$|b\rangle_1$への作用がZ対角

X基底で測定して →

$\boldsymbol{U}$をかけて
$|b\rangle_1, |x\rangle_2$をX測定

← X基底結果(b, d)

Part 2の概略



$$\boldsymbol{D_{P,h}} \approx \boldsymbol{D_{\hat{P},h}} \approx D_{\sigma,h}$$

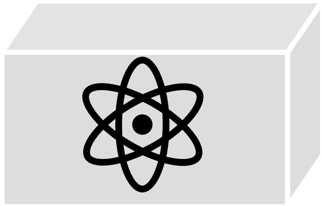
$h \in \{Z, X\}$

$D_{P,Z} = D_{\hat{P},Z}$の証明 (ステップ①)

$\forall \boldsymbol{U}$で$D_{P,Z}$は等しい ⇒ $\boldsymbol{D_{P,Z} = D_{\hat{P},Z}}$

$h = Z \rightarrow k$とトラップドア$\boldsymbol{t_k}$をGから生成

公開鍵k



$U_0(|e\rangle \otimes |k\rangle)$

ビット列$\boldsymbol{y}$

$|y\rangle_3$をZ測定

選択{アダマール}

mへの影響なし

X基底で測定して

$\boldsymbol{U}$をかけて

$|b\rangle_1, |x\rangle_2$をX測定

$(\boldsymbol{m}, x) = INV_G(\boldsymbol{t_k}, \boldsymbol{y})$

mをZ測定結果として出力

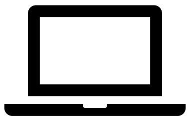
X基底結果(b, d)

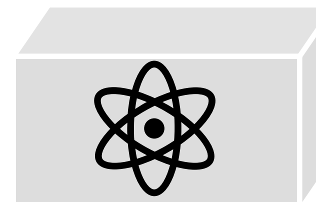
$D_{P,X} \approx D_{\hat{P},X}$の証明 (ステップ①)

$P(U_0, U)$と$P(U_0, Z_1 U Z_1)$でmの分布が識別不可

$\Rightarrow \; D_{P,X} \approx D_{\hat{P},X}$ (∵Twirling Lemma)

一般の P





公開鍵k →

$U_0(|e\rangle \otimes |k\rangle)$

← ビット列y

$|y\rangle_3$をZ測定

$$\boldsymbol{m} = b \oplus d \cdot (x_0 \oplus x_1)$$

$$\begin{cases} x_0 = INV_F(t_k, 0, y) \\ x_1 = INV_F(t_k, 1, y) \end{cases}$$

をX測定結果として出力

X基底で測定して →

U or $Z_1 U Z_1$

$|b\rangle_1$と$|x\rangle_2$をX測定

← X基底結果(b, d)

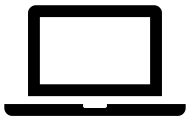
MITSUBISHI ELECTRIC
Changes for the Better

$D_{P,X} \approx D_{\hat{P},X}$の証明 (ステップ①)

$P(U_0, U)$と$P(U_0, Z_1 U Z_1)$でmの分布が識別不可

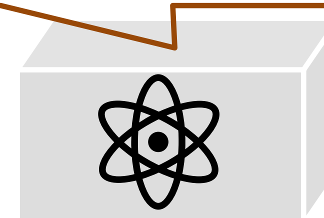
$\Rightarrow$ $D_{P,X} \approx D_{\hat{P},X}$ (∵Twirling Lemma)

テストラウンドに必ず受理



公開鍵k →

← ビット列y



$$\sum_{b=0,1} |b\rangle_1 \left|x_{b,y}\right\rangle_2 \left|\psi_{b,x_{b,y}}\right\rangle$$

yの正しい原像

$\boldsymbol{m} = b \oplus d \cdot (x_0 \oplus x_1)$

$$\begin{cases} x_0 = INV_F(t_k, 0, y) \\ x_1 = INV_F(t_k, 1, y) \end{cases}$$

をX測定結果として出力

X基底で測定して →

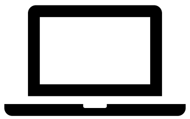
← X基底結果(b, d)

U or $Z_1 U Z_1$

$|b\rangle_1$と$|x\rangle_2$をX測定

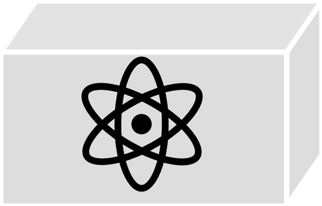
$D_{P,X} \approx D_{\hat{P},X}$の証明 (ステップ①)

$P(U_0, U)$と$P(U_0, Z_1 U Z_1)$でmの分布が識別可
⇒ **対角項 or 非対角項が生成する分布が識別可**
⇒ **LWE仮定に反する**



公開鍵k

ビット列y



$$\sum_{b=0,1} |b\rangle_1 |x_{b,y}\rangle_2 \, |\psi_{b,x_{b,y}}\rangle$$

$$=: |\omega_y\rangle\langle\omega_y|$$

$$=: \boldsymbol{\rho_y^D} + \boldsymbol{\rho_y^C}$$

$\boldsymbol{m} = b \oplus \ d \cdot (x_0 \oplus x_1)$

$$\begin{cases} x_0 = INV_F(t_k, 0, y) \\ x_1 = INV_F(t_k, 1, y) \end{cases}$$

をX測定結果として出力

X基底で測定して

X基底結果(b, d)

$\boldsymbol{U}$ **or** $\boldsymbol{Z_1 U Z_1}$
$|b\rangle_1$と$|x\rangle_2$をX測定

対角項分布が識別可 → AHB性1を破る

一般の P

公開鍵k →

← ビット列y

$$\rho_y^D = \sum_b |b, x_{b,y}, \psi_{b,x_{b,y}}\rangle\langle \text{"} |$$

$|\omega_y\rangle$をZ測定 (b, x)

$$\boldsymbol{m} = b \oplus \ d \cdot (x_0 \oplus x_1)$$

$$\begin{cases} x_0 = INV_F(t_k, 0, y) \\ x_1 = INV_F(t_k, 1, y) \end{cases}$$

をX測定結果として出力

X基底で測定して →

← X基底結果(b, d)

X測定結果のbit flip

$\boldsymbol{U}$ **or** $\boldsymbol{Z_1 U Z_1}$ 影響なし

$|b\rangle_1$と$|x\rangle_2$をX測定

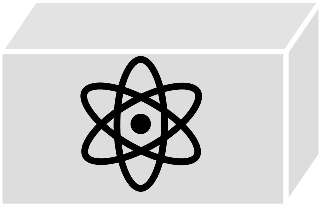
$$U: \sum_{m=0,1} P_U\,[m]|m\rangle\langle m| \quad \xleftrightarrow[\text{の関係}]{\text{bit flip}} \quad Z_1UZ_1: \sum_{m=0,1} P_{Z_1UZ_1}\,[m]|m\rangle\langle m|$$

対角項分布が識別可 → AHB性1を破る

公開鍵k

ビット列y

一般の P



$|\omega_y\rangle$をZ測定 (b,x)

$$\rho_y^D = \sum_b |b, x_{b,y}, \psi_{b,x_{b,y}}\rangle\langle " |$$

$$\boldsymbol{m} = b \oplus \boldsymbol{d} \cdot (\boldsymbol{x_0} \oplus \boldsymbol{x_1})$$

$$\begin{cases} x_0 = INV_F(t_k, 0, y) \\ x_1 = INV_F(t_k, 1, y) \end{cases}$$

をX測定結果として出力

X基底で測定して

X基底結果(b,d)

U or $Z_1 U Z_1$

$|b\rangle_1$と$|x\rangle_2$をX測定

$\sum_m P_U\,[m]|m\rangle\langle m|$と$\sum_m P_U\,[m \oplus 1]|m\rangle\langle m|$が識別可

⇒ $\boldsymbol{d} \cdot (\boldsymbol{x_0} \oplus \boldsymbol{x_1})$が得られる

Trapdoor claw-free 関数族

LWE仮定から関数族FとGが構成可能

関数族 $F = \{f_{k,0}, f_{k,1}\}_k$

3. Adaptive hardcoreビット性1:

QPT Pは公開鍵kから

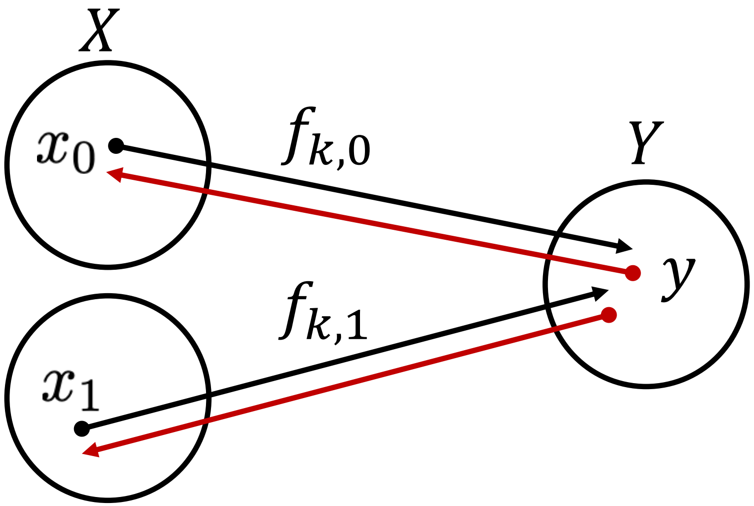
$(y, b, x_b, r, d \neq 0)$

を出力することが難しい.

$y = f_{k,b}(x_b), r = d \cdot (x_0 \oplus x_1)$

Z測定結果

X測定結果

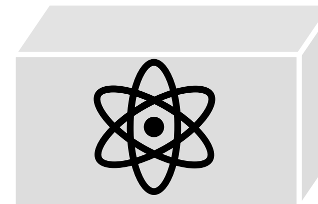


非対角項分布が識別可 → AHB性2を破る

公開鍵k →

← ビット列y

一般の P



$$\rho_y^C = \sum_{b=0,1} |b, x_{b,y}\rangle\langle b\oplus 1, x_{b\oplus 1,y}| \ldots$$

$$\boldsymbol{m} = b \oplus d \cdot (x_0 \oplus x_1)$$

$$\begin{cases} x_0 = INV_F(t_k, 0, y) \\ x_1 = INV_F(t_k, 1, y) \end{cases}$$

をX測定結果として出力

X基底で測定して →

← X基底結果(b, d)

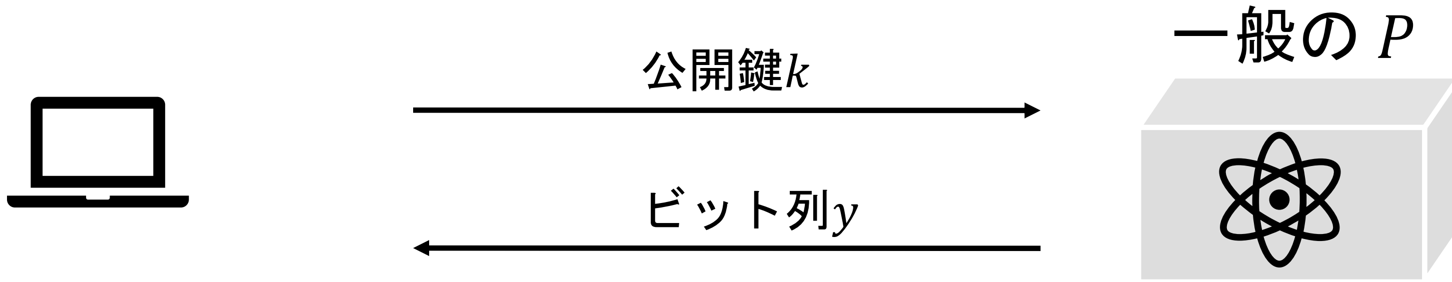
$\boldsymbol{U}$ or $\boldsymbol{Z_1 U Z_1}$

$|b\rangle_1$と$|x\rangle_2$をX測定

$$U: \sum_{m=0,1} P_U[m]|m\rangle\langle m|$$

$$Z_1UZ_1: \sum_{m=0,1} P_{Z_1UZ_1}[m]|m\rangle\langle m|$$

非対角項分布が識別可 → AHB性2を破る



非対角分布が識別可⇒ $\{|\omega_y\rangle, Z_1|\omega_y\rangle\}$が識別可

A

効率的に作れる$Z_2^d|\omega_y\rangle$

$$= Z_1^{d\cdot(x_0\oplus x_1)}|\omega_y\rangle$$

A

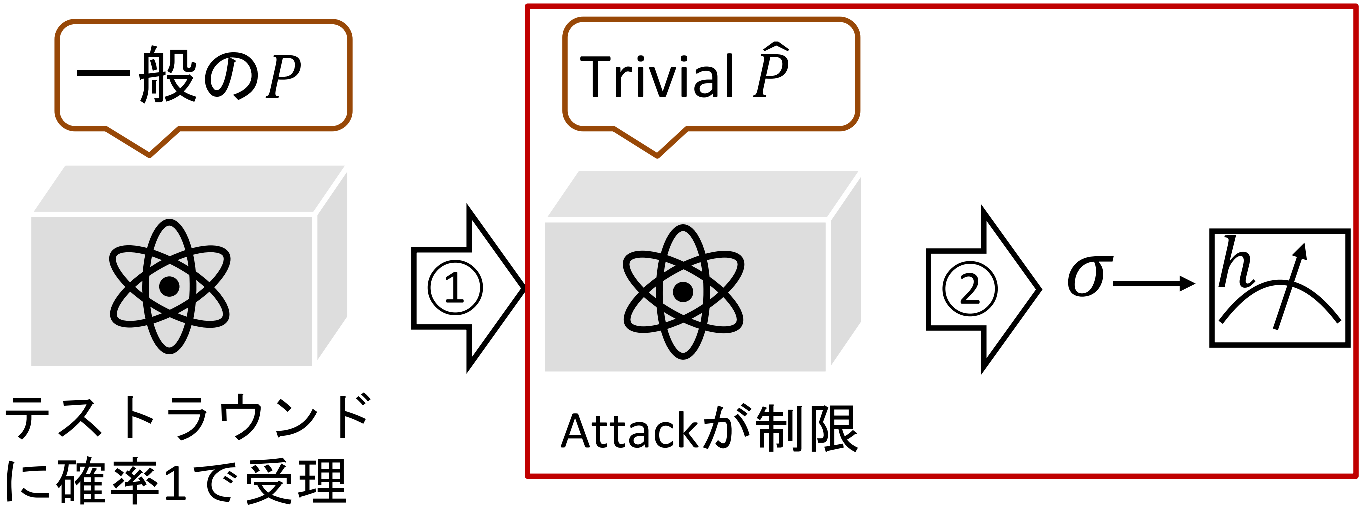
A'

A'が選んだdに対する

$d\cdot(x_0\oplus x_1)$が求まる

[＝AHB性2を破る]

Part 2の概略



$$D_{P,h} \approx \boldsymbol{D_{\hat{P},h}} \approx \boldsymbol{D_{\sigma,h}}$$

①一般のPに対してTrivial $\hat{P}$が存在して$D_{P,h} \approx D_{\hat{P},h}$

②Trivial $\hat{P}$に対してある量子状態σが存在して$D_{\hat{P},h} \approx D_{\sigma,h}$

Trivial $\hat{P}, D_{\hat{P},h} \approx D_{\sigma,h}$の証明 (ステップ②)

$\hat{P}(U_0, U)$に対応する状態σの構成 (h無依存)

1. 関数族Fから$($公開鍵k, トラップドア$t_k)$を生成
2. ユニタリU_0を作用、$|y\rangle_3$をZ測定、Uを作用
3. 原像レジスタ$|x\rangle_2$をX測定し、dを得る
4. トラップドアを用いて$\boldsymbol{Z^{d\cdot(x_0 \oplus x_1)}}$を$|b\rangle_1$に作用
5. 量子ビット$|b\rangle_1$以外を捨てる

$h = Z$なら不要

$h = Z$なら不要

$h = Z$なら不要

⇒ $\boldsymbol{D_{\sigma,X} = D_{\hat{P},X}}$ [証明終]

∵ 4. はVのbit flip $[\oplus\, d \cdot (x_0 \oplus x_1)]$をシミュレート

$\boldsymbol{D_{\sigma,Z} \approx D_{\hat{P},Z}}$を2つの状態を介して示す

$\boldsymbol{D_{\sigma,Z}} = D_{\sigma_Z^{(1)},Z} \approx D_{\sigma_Z^{(2)},Z} = \boldsymbol{D_{\hat{P},Z}}$

$\boldsymbol{\hat{P}(U_0, U)}$に対応する状態$\boldsymbol{\sigma_Z^{(1)}}$の構成

1. 関数族$\boldsymbol{F}$から(公開鍵k, トラップドアt_k)を生成. t_kを捨てる
2. ユニタリU_0を作用、$|y\rangle_3$をZ測定、Uを作用
3. 量子ビット$|b\rangle_1$以外を捨てる

$\boldsymbol{D_{\sigma,Z} \approx D_{\hat{P},Z}}$を2つの状態を介して示す

$$\boldsymbol{D_{\sigma,Z} = D_{\sigma_Z^{(1)},Z}} \approx D_{\sigma_Z^{(2)},Z} = D_{\hat{P},Z}$$

Trivial $\hat{P}, D_{\hat{P},h} \approx D_{\sigma,h}$の証明 (ステップ②)

$\hat{\boldsymbol{P}}(\boldsymbol{U_0}, \boldsymbol{U})$に対応する状態$\boldsymbol{\sigma_Z^{(2)}}$の構成

1. 関数族$\boldsymbol{G}$から$\left(\text{公開鍵}k,\text{トラップドア}t_k\right)$を生成. t_kを捨てる
2. ユニタリU_0を作用、$|y\rangle_3$をZ測定、Uを作用
3. 量子ビット$|b\rangle_1$以外を捨てる

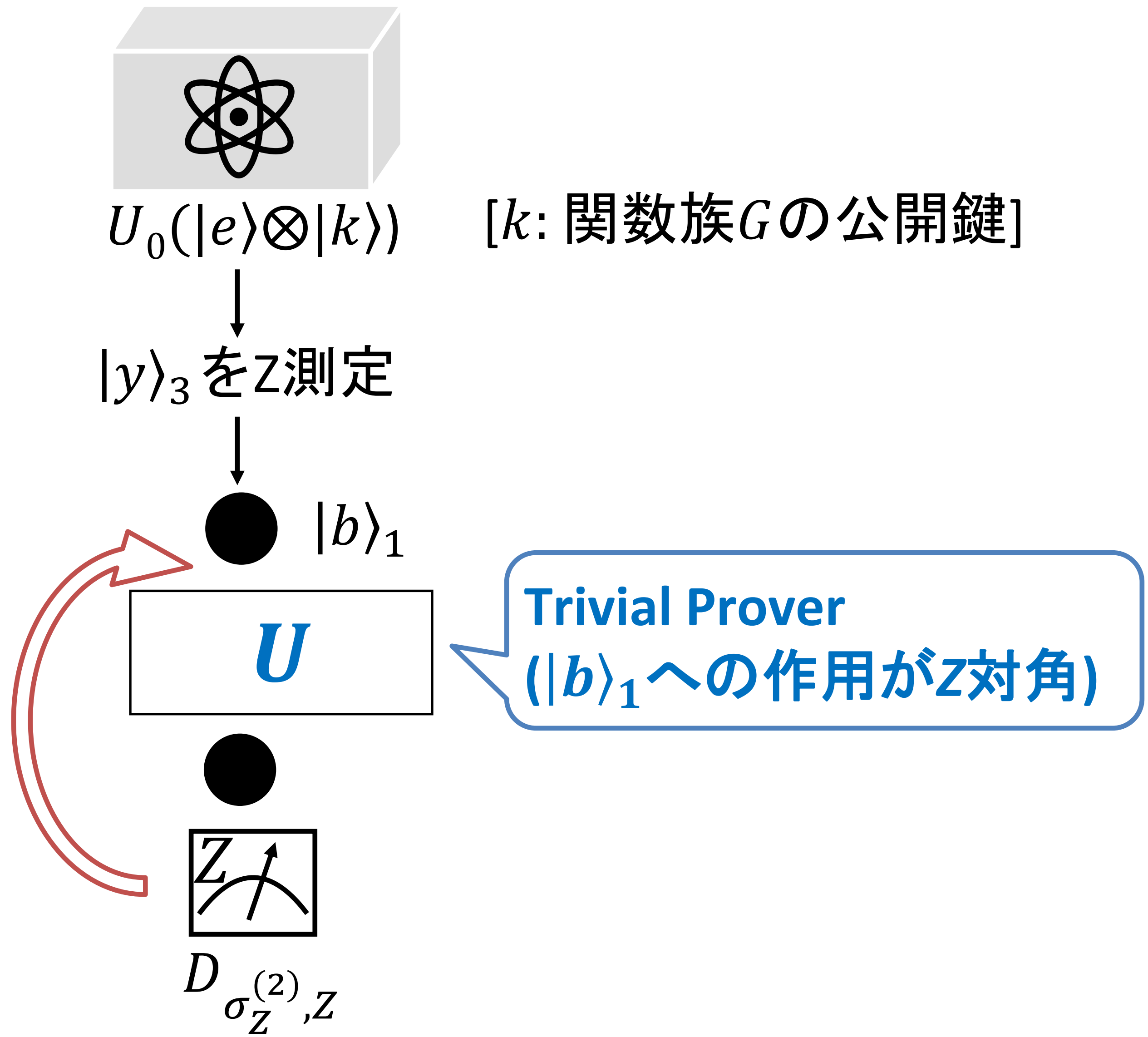
$\boldsymbol{D_{\sigma,Z} \approx D_{\hat{P},Z}}$を2つの状態を介して示す

$$D_{\sigma,Z} = \boldsymbol{D_{\sigma_Z^{(1)},Z}} \approx \boldsymbol{D_{\sigma_Z^{(2)},Z}} = D_{\hat{P},Z}$$

(対偶証明)
k (F or G)入力のQPT操作でFor Gが識別可
⇒TCF関数のInjective invarianceに反する

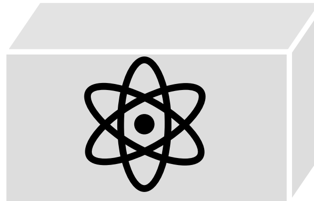
Trivial $\hat{P}, D_{\hat{P},h} \approx D_{\sigma,h}$の証明 (ステップ②)

$\sigma_Z^{(2)}$の構成



Trivial $\hat{P}, D_{\hat{P},h} \approx D_{\sigma,h}$の証明 (ステップ②)

$\boldsymbol{\sigma_Z^{(2)}}$の構成



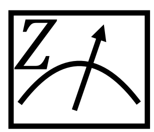
$U_0(|e\rangle \otimes |k\rangle)$　　[k: 関数族Gの公開鍵]

↓

$|y\rangle_3$をZ測定

↓

● $|b\rangle_1$



$\boldsymbol{D_{\sigma_Z^{(2)},Z}}$

$y \in Y_0$か$y \in Y_1$
かを正しく出力

↓

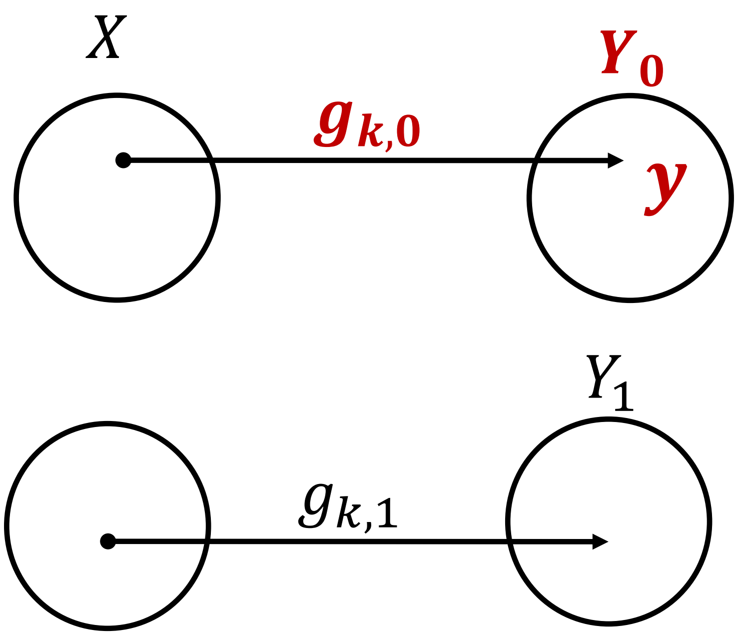
$INV_G(t_k, y)$
と同じ結果

↓

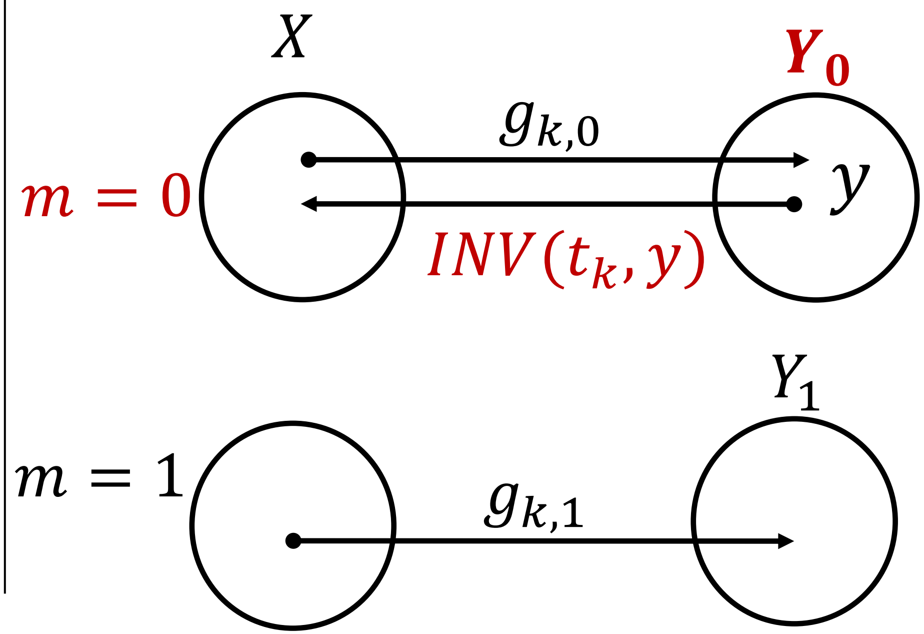
Vの出力mと同じ
$\boldsymbol{D_{\hat{P},Z}}$

関数族 $G = \{g_{k,0}, g_{k,1}\}_k$

$\sigma_Z^{(2)}$ でのZ測定：yがg_0 or g_1を正しく知る
(テストラウンド確率1で受理するPのため)

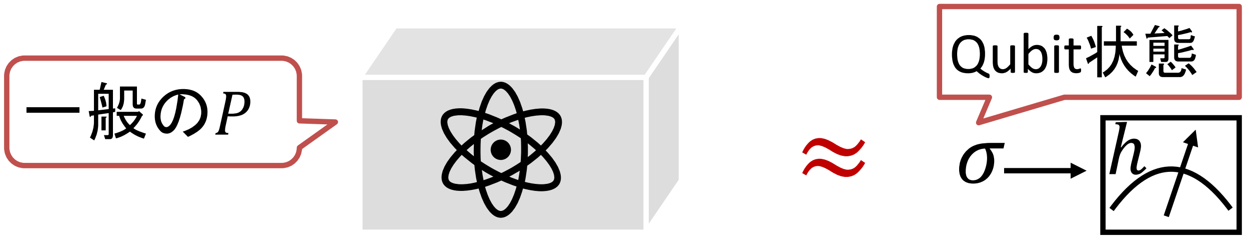


Vの出力m: t_kを使いg_0 or g_1を知る



Mahadev [FOCS 2018]のまとめ

Measurementプロトコル



テストラウンド
に確率1で受理

$$\exists \sigma, \forall h \in \{Z, X\}^n, \quad \boldsymbol{D_{P,h}} \approx \boldsymbol{D_{\sigma,h}}$$

量子計算の古典検証
(Measurement + ポストホックプロトコル)

量子計算機の答えが正しい場合：

$\Pr[\text{VがPを受理}] \geq 1 - \mathrm{negl}$

量子計算機の答えが正しくない場合：

$\Pr[\text{VがPを受理}] \leq \frac{1}{2} + \mathrm{negl}$

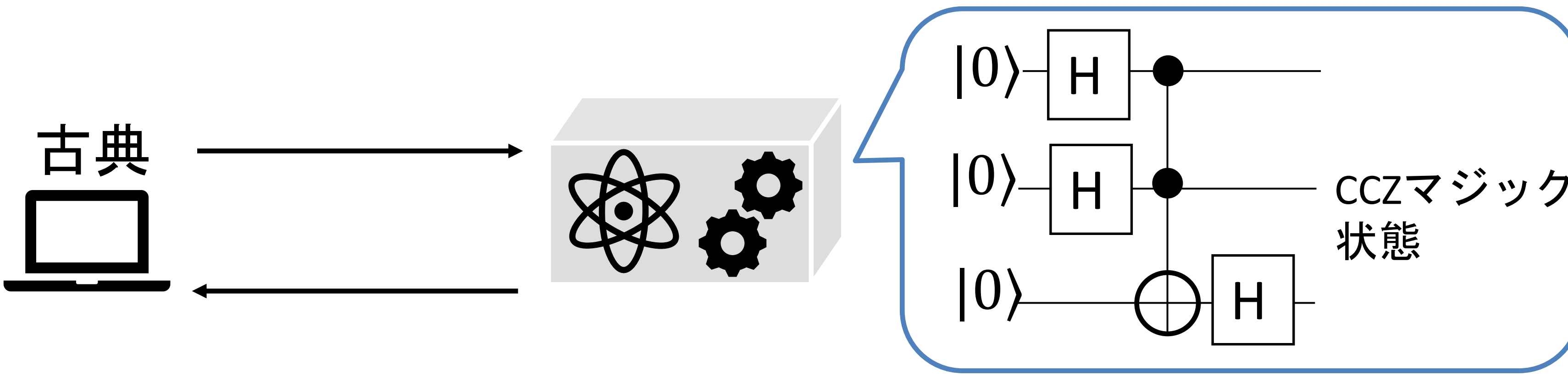
量子計算機の動作検証

三菱電機とNTTの共同研究

Mizutani, Takeuchi, Hiromasa, Aikawa, Tani,
Physical Review A (Letter) **106**, L010601 (2022).

CCZマジック状態の生成・測定機能の古典検証

耐故障性量子計算機の実現に重要な機能



どれだけ複雑な量子ダイナミクスを古典系で検証できるか？