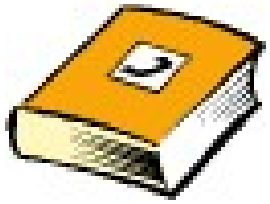
Groverのアルゴリズム
（+少しだけHHL）

グローバーのアルゴリズムに可能なこと

- グローバーのアルゴリズムは，「データベース検索に対する量子アルゴリズム」とも言われる．



電話帳

072-155-2456

メモ

『電話番号はメモってあるんだけど，これって誰の番号だっけ？？』

前のページから順番に探していくととても大変！

阿井上男 057-256-1894

阿川亜美 066-278-2789

. . .

和田信二 072-155-2456

グローバーのアルゴリズムに可能なこと

- 電話番号以外何の情報もない場合，N人の電話番号の載った電話帳なら最悪N回（平均N/2回）電話番号を探索する必要がある.
- グローバーのアルゴリズムを使うと，約 √N回の探索で電話番号に対応する人の名前を（高確率で）見つけられる！

Nと√Nの違いは，Nが大きくなる（つまりデータのサイズがでかくなる）ほど顕著になる

100倍

N	1	100	1万	100万	1億
√N	1	10	100	1000	1万

グローバーのアルゴリズムに可能なこと

N=4の場合，１回の探索により100パーセントで
電話番号に対応する人の名前を見つけられる

電話帳

阿井上男 057-256-1894
阿川亜美 066-278-2789
鈴木太一 088-672-0192
和田信二 072-155-2456

メモ

072-155-2456

普通には「当たり」（＝和田信二）を見つけるために最悪で**4回の探索**が必要.

グローバーのアルゴリズムなら**1回の探索**で「当たり」を見つけることが可能.

グローバーのアルゴリズムを眺める

ーN=4の場合

探索を量子状態的に見直すと

従来の探索

ステップ1　$|A\rangle, |B\rangle, |C\rangle, |D\rangle$　どれかの状態を準備する.

ステップ2　準備した状態に関して「当たり」かどうかを調べる.

当たりを引く確率は1/4．もちろん，量子力学特有の重ねあわせは全く使っていない.

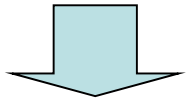
量子力学特有の重ねあわせを使うと

ステップ1　$|A\rangle, |B\rangle, |C\rangle, |D\rangle$ の重ねあわせ状態

$$|p\rangle = \frac{1}{2}|A\rangle + \frac{1}{2}|B\rangle + \frac{1}{2}|C\rangle + \frac{1}{2}|D\rangle = \begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{pmatrix}$$

を準備する.

ステップ2　準備した状態を測定した後,「当たり」かどうかを調べる.



やはり，当たりを引く確率は$(1/2)^2=1/4$．量子力学特有の重ねあわせは全く有効に使われていない.

グローバーのアルゴリズムのポイント

グローバーのアルゴリズムは，４次元ベクトルを変形させることにより，当たりに対応するベクトル（例えば|B>）を最終的に得る作業と考えられる．

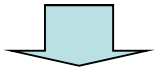
|p>:=最初に準備した状態（均等重ね合わせ）

|B‘>に関する折り返し:＝当たりかどうかをチェック（探索1回）

|p>に関する折り返し：＝自前で実行可能

というような（4次元空間内の）2次元の平面上での折り返し変換を行うことにより・・・

１回くじを引けば，当たりを１００パーセントで得ることが可能！



まさに，このような対応を見つけたことによって
グローバーのアルゴリズムは得られた．

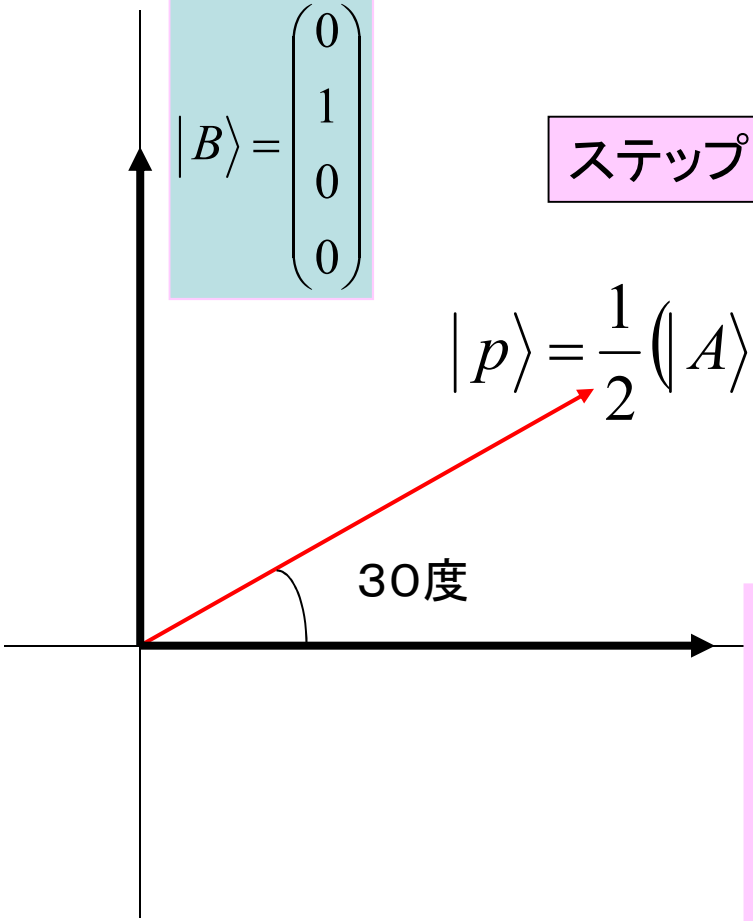
グローバーのアルゴリズムの動き

Bを当たりとすると，このような2次元平面上で量子状態は動く

$$|B\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

ステップ1.　|p>を準備する.

$$|p\rangle = \frac{1}{2}\left(|A\rangle + |B\rangle + |C\rangle + |D\rangle\right) = \frac{1}{2}\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \frac{\sqrt{3}}{2}|B'\rangle + \frac{1}{2}|B\rangle$$

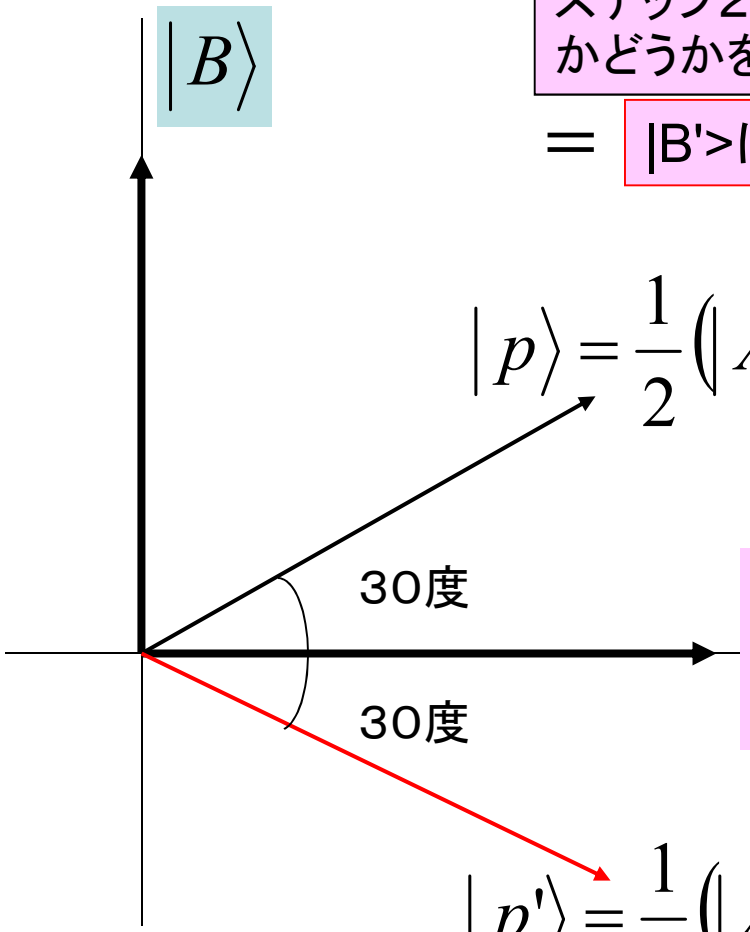


$$|B'\rangle = \frac{1}{\sqrt{3}}\left(|A\rangle + |C\rangle + |D\rangle\right) = \frac{1}{\sqrt{3}}\begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

グローバーのアルゴリズムの動き

ステップ2　準備した状態に関して「当たり」かどうかを量子並列操作により探索する.

＝ |B'>に関する折り返し



$$|p\rangle = \frac{1}{2}\left(|A\rangle + |B\rangle + |C\rangle + |D\rangle\right) = \frac{\sqrt{3}}{2}|B'\rangle + \frac{1}{2}|B\rangle$$

$$|B'\rangle = \frac{1}{\sqrt{3}}\left(|A\rangle + |C\rangle + |D\rangle\right)$$

$$|p'\rangle = \frac{1}{2}\left(|A\rangle - |B\rangle + |C\rangle + |D\rangle\right) = \frac{\sqrt{3}}{2}|B'\rangle - \frac{1}{2}|B\rangle$$

グローバーのアルゴリズムの動き

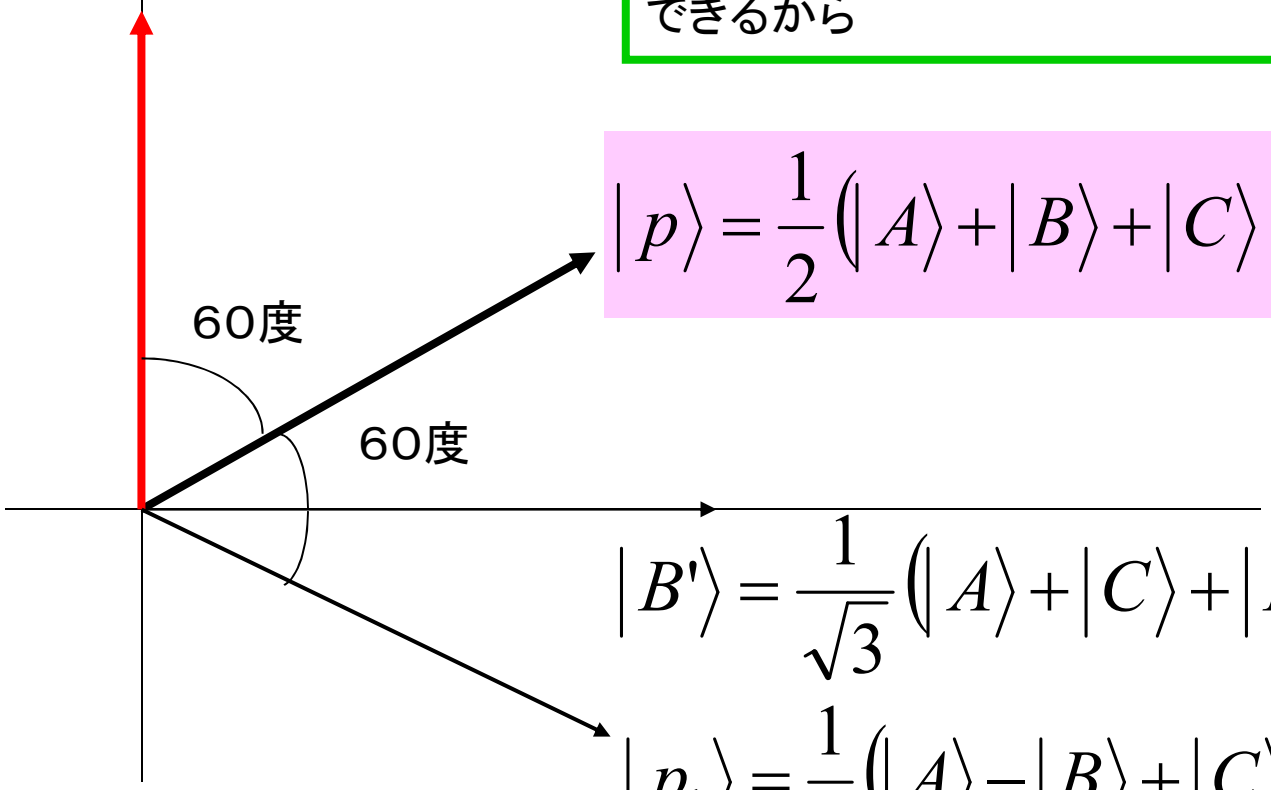
測定すれば
確率1で当たりBを得る

$$|p_2\rangle = |B\rangle$$

ステップ3　|p>に関する折り返し

これは，くじを引くことなく行える操作．理由は，|p>がくじの当たりが何であるかに関係なく準備できるから

$$|p\rangle = \frac{1}{2}\left(|A\rangle + |B\rangle + |C\rangle + |D\rangle\right)$$



$$|B'\rangle = \frac{1}{\sqrt{3}}\left(|A\rangle + |C\rangle + |D\rangle\right)$$

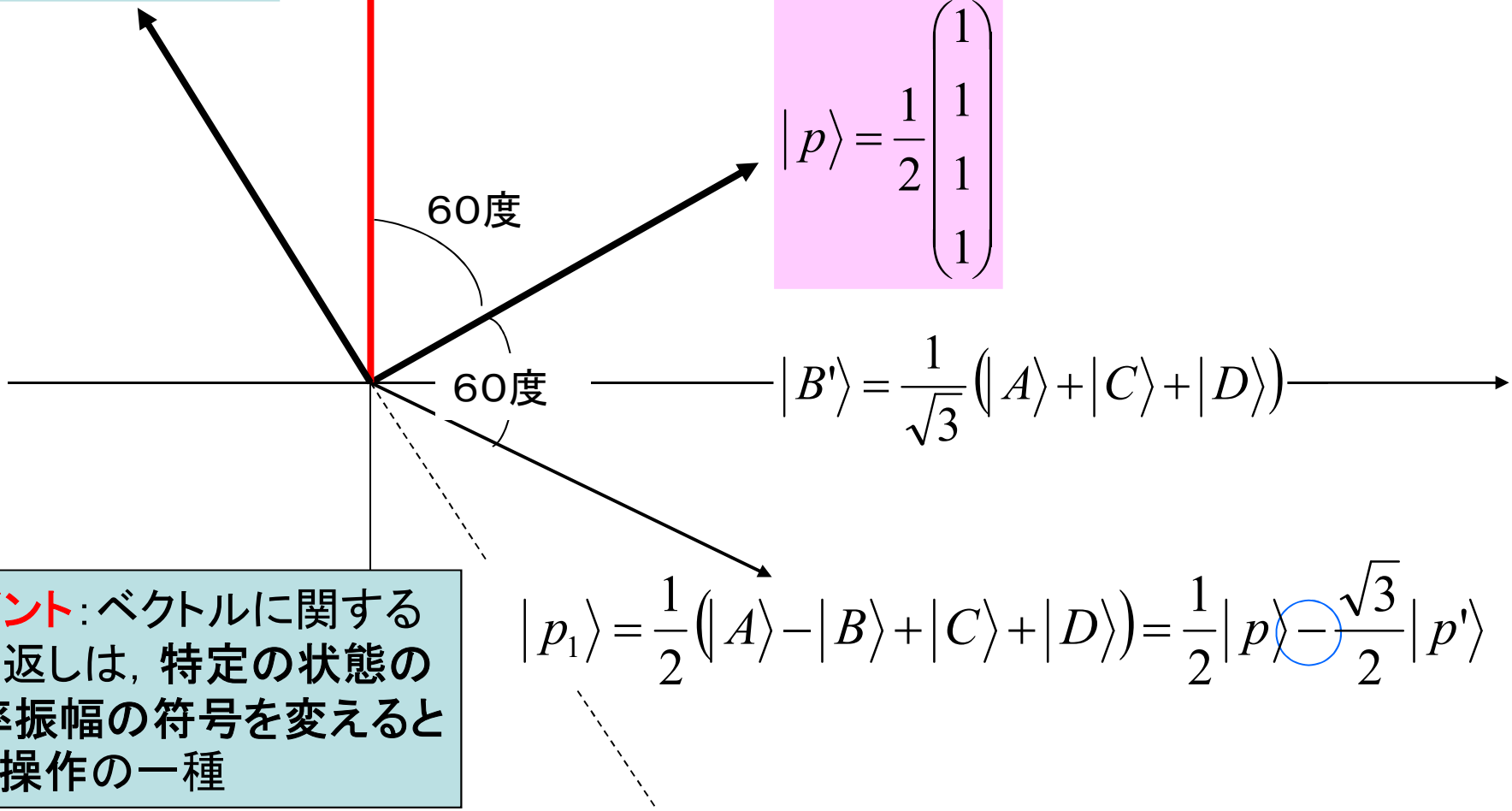
$$|p_1\rangle = \frac{1}{2}\left(|A\rangle - |B\rangle + |C\rangle + |D\rangle\right)$$

ステップ3. ベクトル|p> に関する折り返し

$$|p'\rangle = \frac{1}{2\sqrt{3}}\begin{pmatrix}-1\\3\\-1\\-1\end{pmatrix}$$

$$|p_2\rangle = |B\rangle = \frac{1}{2}|p\rangle + \frac{\sqrt{3}}{2}|p'\rangle$$

$$|p\rangle = \frac{1}{2}\begin{pmatrix}1\\1\\1\\1\end{pmatrix}$$



$$|B'\rangle = \frac{1}{\sqrt{3}}\left(|A\rangle + |C\rangle + |D\rangle\right)$$

$$|p_1\rangle = \frac{1}{2}\left(|A\rangle - |B\rangle + |C\rangle + |D\rangle\right) = \frac{1}{2}|p\rangle - \frac{\sqrt{3}}{2}|p'\rangle$$

ポイント：ベクトルに関する折り返しは，**特定の状態の確率振幅の符号を変えるという操作の一種**

グローバーのアルゴリズムを眺める

ー当たりの候補の数が一般の自然数Nの場合

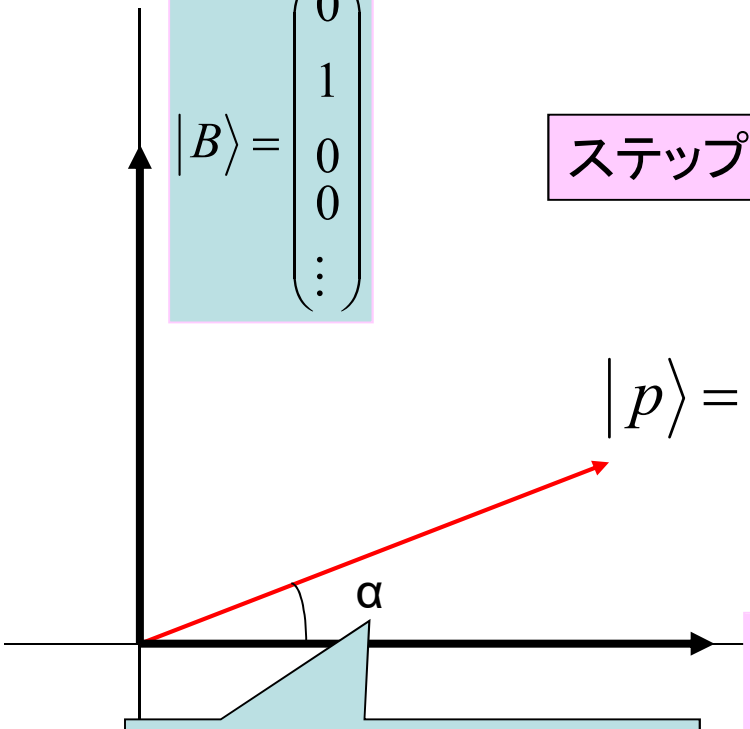
グローバーのアルゴリズムの動き

Bを当たりとすると，このような2次元平面上で量子状態は動く

$$|B\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ \vdots \end{pmatrix}$$

ステップ1．　|p>を準備する.

$$|p\rangle = \frac{1}{\sqrt{N}}\left(|A\rangle + |B\rangle + |C\rangle + |D\rangle + \ldots\right)$$



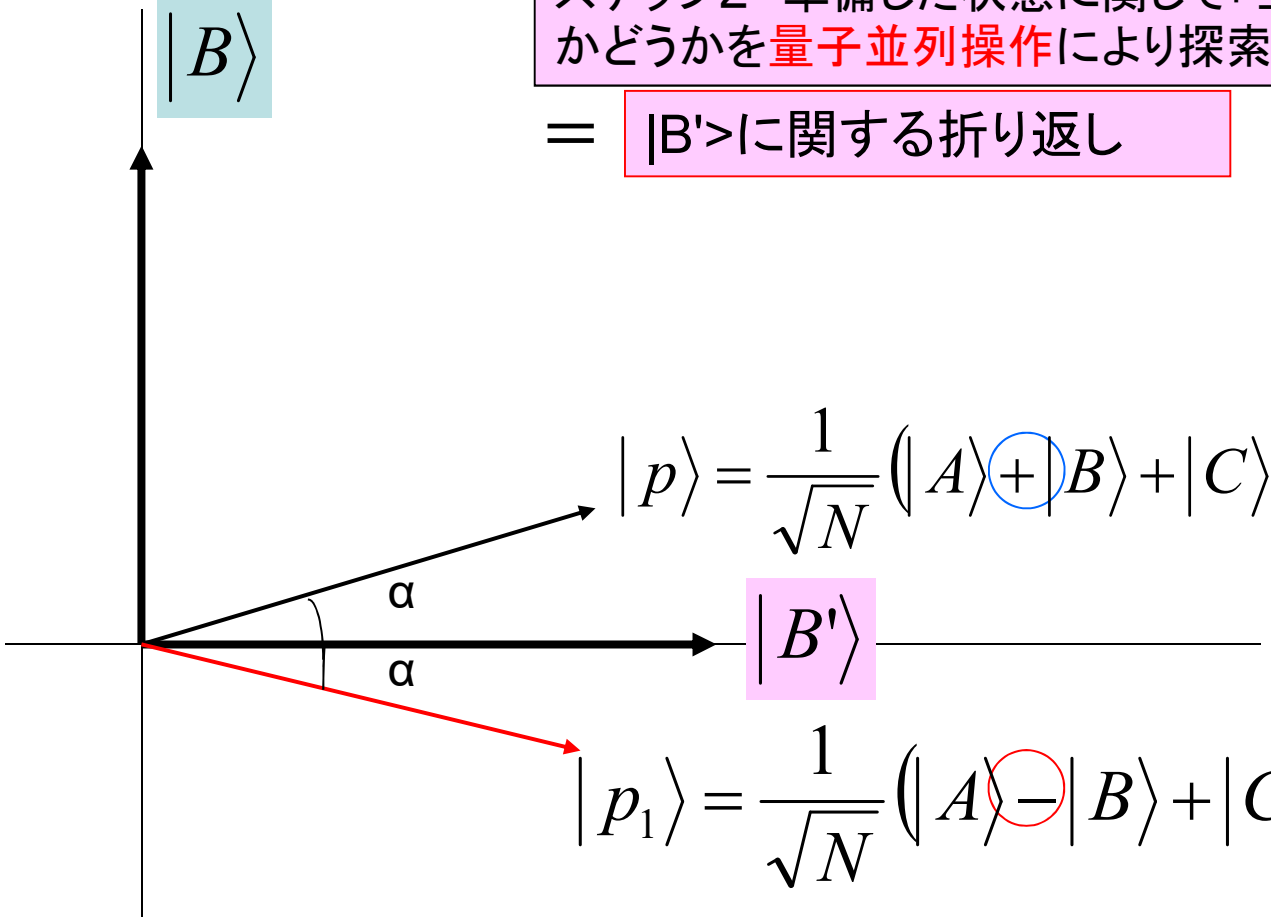
$$|B'\rangle = \frac{1}{\sqrt{N-1}}\left(|A\rangle + |C\rangle + |D\rangle + \ldots\right)$$

N=4のときは|p>と|B'>の
なす角度はたまたま30度
だった.

グローバーのアルゴリズムの動き

ステップ2　準備した状態に関して「当たり」かどうかを量子並列操作により探索する.

＝ |B'>に関する折り返し

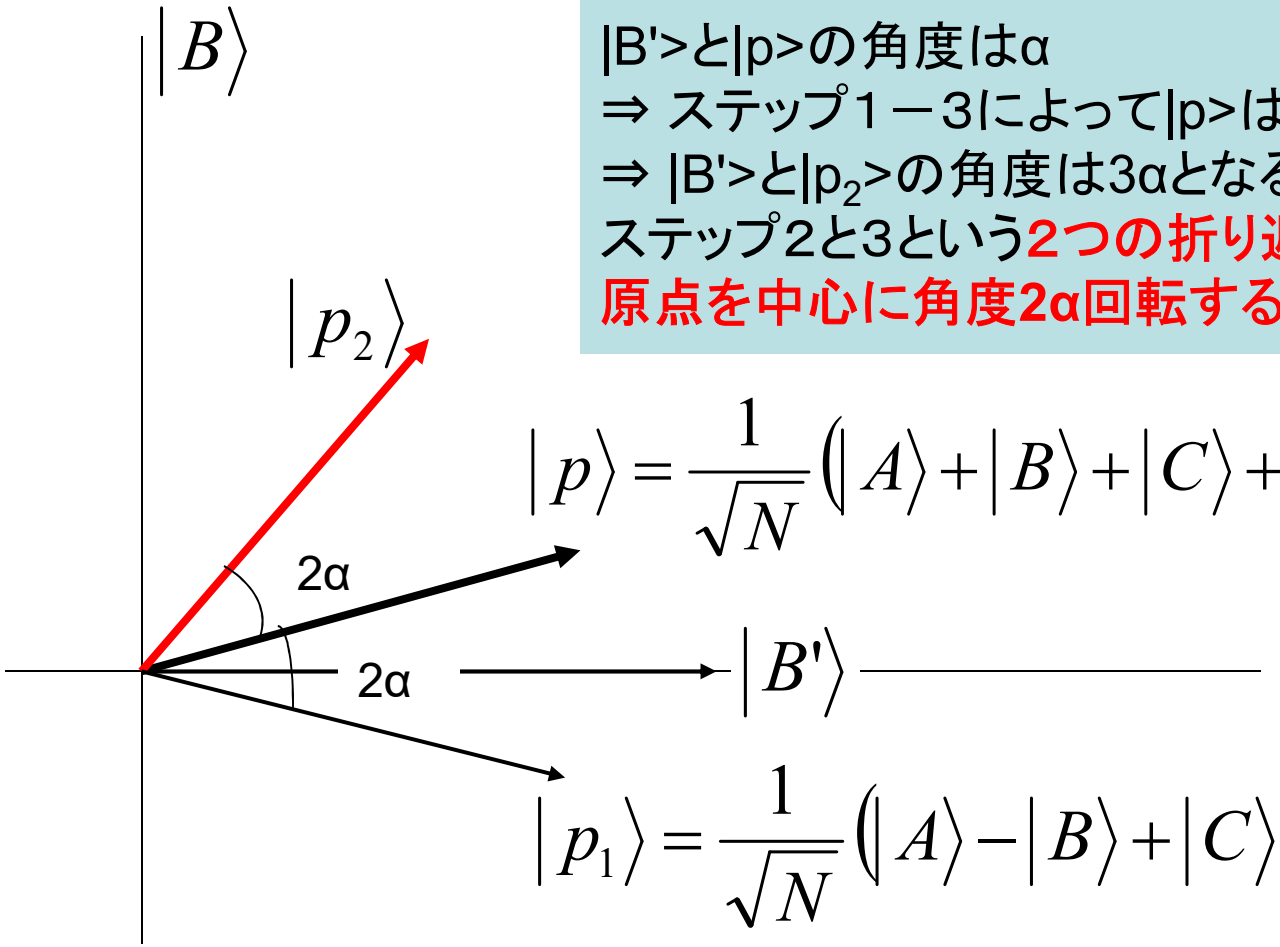


$$|p\rangle = \frac{1}{\sqrt{N}}\left(|A\rangle + |B\rangle + |C\rangle + |D\rangle + \ldots\right)$$

$$|p_1\rangle = \frac{1}{\sqrt{N}}\left(|A\rangle - |B\rangle + |C\rangle + |D\rangle + \ldots\right)$$

グローバーのアルゴリズムの動き

ステップ3　|p>に関する折り返し：データベースへのアクセス不要な操作

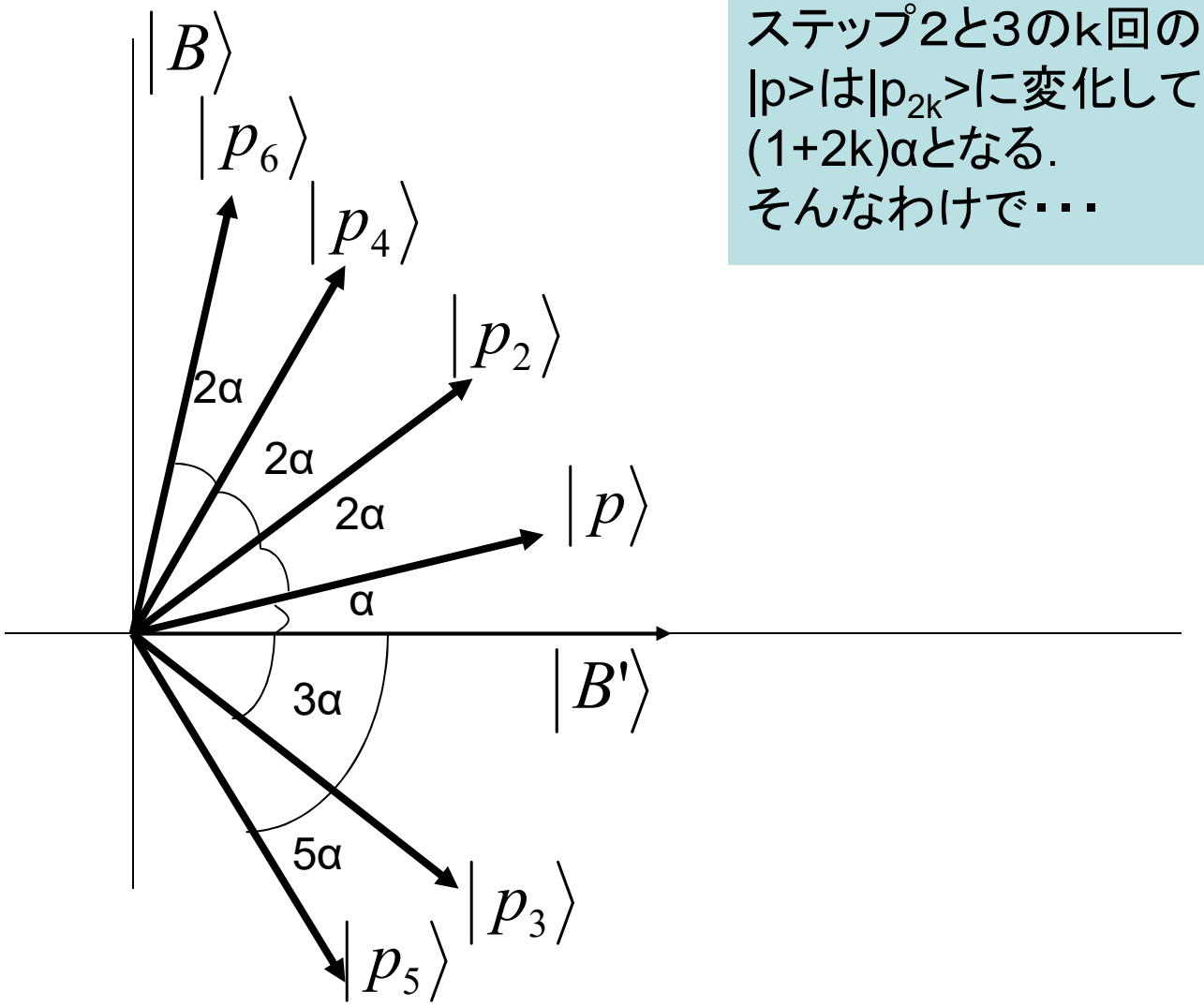


|B'>と|p>の角度はα
⇒ ステップ1－3によって|p>は$|p_2>$に変化
⇒ |B'>と$|p_2>$の角度は3αとなる.
ステップ2と3という**2つの折り返し変換はベクトルを原点を中心に角度2α回転する変換**となっている

$$|p\rangle = \frac{1}{\sqrt{N}}\left(|A\rangle + |B\rangle + |C\rangle + |D\rangle + \ldots\right)$$

$$|p_1\rangle = \frac{1}{\sqrt{N}}\left(|A\rangle - |B\rangle + |C\rangle + |D\rangle + \ldots\right)$$

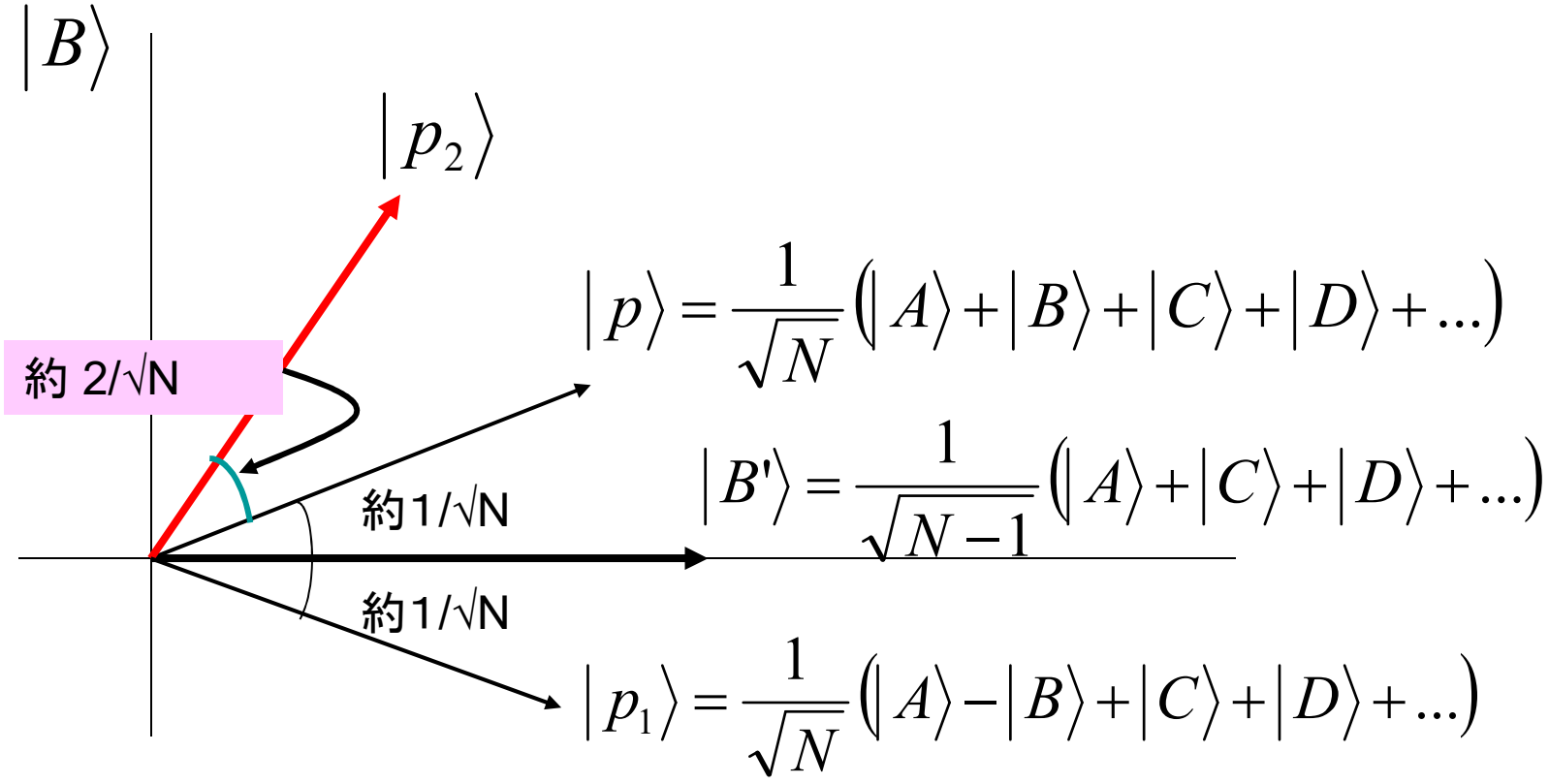
さらにステップ2と3を繰り返せば・・



ステップ2と3のk回の繰り返しにより
|p>は$|p_{2k}\rangle$に変化して|B'>と$|p_{2k}\rangle$の角度は
(1+2k)αとなる.
そんなわけで・・・

候補の数が一般の自然数Nの場合

- N個のうち１個が当たりの場合は，２種類のベクトルに関する折り返し（ステップ２－３）の繰り返し．
- ２回のベクトルに関する折り返しは約 2/√N の回転なので，当たりに達するには，約 (π√N)/4 回データベースにアクセスを行えばOK．



グローバーのアルゴリズムの応用例

グローバーのアルゴリズムは汎用性が高い

- **もし，当たりが幾つかあったらどうなのか？**
  - 大丈夫．N個のうちk個が当たりの場合，約 $\sqrt{\frac{N}{k}}$ 回の探索で当たりを見つけることが可能です．
- **当たりの個数が分からなくても大丈夫？**
  - 大丈夫．量子数え上げと呼ばれるアルゴリズムがあり，当たりの個数さえおおまかに知ることができます．
- **他に応用はあるの？**
  - あります．例えば，困難そうな問題を問題の構造など関係なく速く解くことに利用できるし，暗号解読にも利用できます．またグローバーのアルゴリズムを利用して様々な別のアルゴリズムが提案されています．

グローバーのアルゴリズムを応用した問題

例1： 最小値探索問題

N個のデータ $j \in \{1,2,3,\ldots,N\}$ の入ったデータベースがあり，各データ j は値 f(j) を持つ．最小の値を取るデータ k（つまり，$f(k) \le f(j)$が全ての j に対して成立）を見つけよ．

通常は，全てのデータをチェックする以外，いい方法はないので N回のデータベースへのアクセスが必要．

約√N回のデータベースへのアクセスで最小値を与えるデータを見つける量子アルゴリズムが発見されている　[デュル，ホイヤー 1996].

データ1	18
データ2	35
データ3	16
データ4	19
データ5	18
データ6	33
データ7	14
データ8	19

データベース（N=8の例）

最小値を取るデータは
データ7

グローバーのアルゴリズムを応用した問題

例2：　衝突探索問題

N個のデータ j ∈ {1,2,3,…,N} の入ったデータベースがあり，各データ j は値 f(j) を持つ．但し，fは「2対1」関数である．つまりf(j)=f(k)となる(j,k)のペア（衝突）がちょうどN/2個ある．衝突を1つ見つけよ．

通常は，√N回のデータベースへのアクセスが必要（いわゆる，バースデーパラドックスを利用）．

約$N^{1/3}$回のデータベースへのアクセスで衝突を見つける量子アルゴリズムが発見されている[ブラサード，ホイヤー，タップ 1998]．

データ1	18
データ2	35
データ3	19
データ4	19
データ5	16
データ6	35
データ7	16
データ8	18

衝突探索問題のデータベース（N=8の例）

衝突は(1,8),(2,6),(3,4),(5,7)の4つでどれか1つを見つければよい

Groverのアルゴリズムの汎用性

- GroverのアルゴリズムはShorほど高速ではない一方で，古典の乱択アルゴリズムと組み合わせることが可能であるなど汎用性が高い

振幅増幅(Amplitude Amplification)アルゴリズム

古典の乱択アルゴリズムが計算量T，成功率 p である要素を見つけるなら

（古典の場合，繰り返しにより計算量 T/p，成功率 0.99でその要素を見つけるところを）

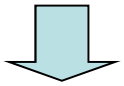
ある量子アルゴリズムが存在して，そのアルゴリズムは計算量 T/√p，成功率0.99で

その要素を見つけることができる

探索問題に対するGroverのアルゴリズムは，振幅増幅アルゴリズムの特殊ケース

fの評価を行う x をランダムに選べば T=1 回の評価で p=1/N の成功率で

f(x)=1なる x が見つかる.



量子アルゴリズムは T/√p=O(√N)回の評価で0.99の成功率でそのような x を得る

3SATへの応用

3SAT

入力　3SAT論理式　$f(x_1,x_2,...,x_n)$

出力　$f(x)=1$となる割り当て$x=(x_1,x_2,...,x_n)$が存在すればYES，存在しなければNO

古典の全探索は2^nの計算量を要するが，

探索問題に対する量子アルゴリズムを使えば $2^{n/2}=(1.414)^n$

しかし，問題の構造を考慮すれば3SATは古典でもっと速く解けることが知られている．例えばSchoningのランダムウォークを用いたアルゴリズムは$(4/3)^n=(1.333)^n$の計算量で3SATを解く．

poly(n)の計算量，成功率$(3/4)^n$で割り当てを見つけるので，
振幅増幅により量子だと計算量 $poly(n)(4/3)^{n/2}=O((1.155)^n)$
で3SATを解く

まとめ

- グローバーのアルゴリズムの可能性
  - N個の候補から「当たり」を見つけるのに約√N回の探索でOK.
  - 特に，N=4なら1回の探索で確実に「当たり」をつかむ.
  - 汎用性が高く，様々な問題への応用を持っている.
- グローバーのアルゴリズムのポイント
  - 最初準備したベクトルを2種類のベクトルに関する折り返しで「当たり」の方へ回転（回転角約2/√N）させていくことができる.

    「折り返し変換2回＝回転」という初等幾何の典型的応用

  - 2種類の折り返しのうち，1種類は探索を（量子並列的に）行うことに対応し，もう1種類は探索不要.

少しだけHHL

Harrow-Hassidim-Lloyd

- 解きたい問題：線形方程式

入力：N次行列A，およびN次単位ベクトル$\boldsymbol{b}$

出力：$A\boldsymbol{x} = \boldsymbol{b}$なる解$\boldsymbol{x}$

- 実際にHHLが出す出力 $|\boldsymbol{x}\rangle = \sum_{i=1}^{N} x_i |i\rangle$の近似ベクトル$|\boldsymbol{x}'\rangle$
- 計算量 $\mathrm{poly}(\log N, \kappa, 1/\epsilon)$
  - $\left| |x\rangle - |x'\rangle \right| \leq \epsilon$
  - κはAの最大固有値/最小固有値の比（condition number）：既知
- AはエルミートとしてもOK：

$$\begin{pmatrix} 0 & A \\ A^{\dagger} & 0 \end{pmatrix} \begin{pmatrix} 0 \\ \boldsymbol{x} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b} \\ 0 \end{pmatrix}$$

＊　以下，最小固有値が$1/\kappa$以上，最大固有値が1以下と仮定
（記述簡単化のためで本質でない）

Harrow-Hassidim-Lloyd

- $A = \sum_{j=1}^{N} \lambda_j |a_j\rangle\langle a_j|$: スペクトル分解
- $|b\rangle = \sum_j \beta_j |a_j\rangle$
- $A^{-1}|b\rangle = \sum_j \beta_j \left(\frac{1}{\lambda_j}\right)|a_j\rangle$得ればHHL完了
- $U = e^{iA}$は効率的に実行可能（Hamiltonian simulation）
- Uの位相推定でλ_j知ることができて

$|a_j\rangle|0\rangle \mapsto |a_j\rangle\left(\frac{1}{\kappa\lambda_j}|0\rangle + \sqrt{1-\frac{1}{(\kappa\lambda_j)^2}}|1\rangle\right)$ 実行可能．このとき

$$\sum_j \beta_j |a_j\rangle\left(\frac{1}{\kappa\lambda_j}|0\rangle + \sqrt{1-\frac{1}{(\kappa\lambda_j)^2}}|1\rangle\right) = \frac{1}{\kappa}\sum_j \beta_j \left(\frac{1}{\lambda_j}\right)|a_j\rangle|0\rangle + |g\rangle|1\rangle$$