

# プロセス理論としての量子計算

川 辺 治 之

## 1. はじめに

量子計算のモデルとして、ヒルベルト空間とそれに対する（有界）線形作用素が広く用いられている。それに対応して、量子アルゴリズムは量子回路モデルによって実装されることになる。しかし、古典的な計算機においては回路レベルのプログラミングから抽象度の高いプログラミング・パラダイムへと発展したように、量子計算においても高水準なプログラミング言語が求められる。そのような高水準プログラミング言語の枠組みとして、図式に基づくプロセス理論を取り上げる。図式に基づくプロセス理論は、プロセスを合成することによってより複雑なプロセスを構成し、さまざまな対象についての「計算」を記述することができる。本稿で紹介する図式は、ペンローズが用いた図式<sup>[5]</sup>を拡張したもので、最初にクック<sup>[1][2]</sup>およびそれとは独立にカウフマン<sup>[4]</sup>によって導入された。本稿では、このプロセス理論によってヒルベルト空間と等価な量子計算モデルを構成し、その一例として量子テレポーテーションが必然的に生じる古典的データの受け渡しも含めて図式だけで表現できることを、クックとキッシンジャーによる教科書<sup>[3]</sup>に沿って解説する。なお、本文中の図式については、同書から引用した。本稿で割愛したいいくつかの命題の証明やそのほかの具体例についても同書を参照されたい。

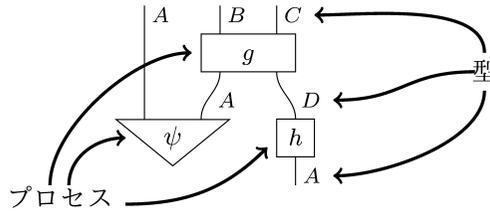
本稿の構成は次のとおりである。2章では、図式に基づくプロセス理論の概要と図式に対する基本的な演算を示す。3章では多くのプロセス理論を表現することのできる回路図式とストリング図式を、4章では線形写像のプロセス理論を定義する。5章では、線形写像のプロセス理論を二重化することから量子写像のプロセス理論を構成し、量子写像の集まりとして量子プロセスを定義する。6章では、量子測定が非決定的量子プロセスとみなせることが分かる。そして、7章では、図式によって量子系だけでなく古典系も含めた表現が可能となり、その一例として量子テレポーテーションを図式によって表現する。

## 2. プロセス理論

それぞれのプロセスには、その入力となる系（入力系）と出力となる系（出力系）がある。プロセスの入力系や出力系は複数あってもよいし、まったくなくてもよい。それぞれの系は型をもち、一つのプロセスに同じ型の入力系や出力系が複数あってもよい。

このようなプロセスを箱で表現し、そのプロセスの入力系はその箱の下辺につなぎこむ結線、出力系はその箱の上辺につなぎこむ結線によって表す。プロセス  $f$  のある出力系の型がプロセス  $g$  のある入力系の型と一致する場合にかぎり、その出力系に対応する前者の箱の上辺と、その入力系に対応する後者の箱の下辺を結線でつなぎ合わせることができ

る。(プロセスが複数の入力系をもつ場合には、それぞれの入力系が箱の下辺に順に並んでいると考えるので、結線をつなぎこむ位置には意味がある。また、あるプロセスの入力系がほかのプロセスの出力系とつながっていない場合にも、プロセスが入力をもつことを示すために、箱の下辺からほかの箱にはつながらないように結線を下方に延ばす。出力系についても同様である。) このようにして、箱(プロセス)を結線でつなぎ合わせた図を**図式**と呼ぶ。



複数のプロセスを区別する必要がある場合には、図式中のプロセスを表す箱の内部にプロセスの名前を付与する。また、結線の型を識別する必要がある場合には、結線の脇にその型の名前を付与する。

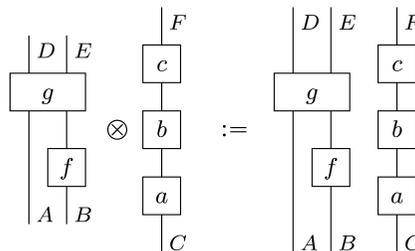
プロセス理論は次のものから構成される。

- 結線によって表現される型の集まり  $T$
- 箱によって表現されるプロセスの集まり  $P$ 、ただし、 $P$ に含まれるプロセスのそれぞれの入力系と出力系の型は  $T$ の中から選ばれる。
- 「プロセスをつなぎ合わせる」方法、すなわち  $P$ に含まれるプロセスからなる図式全体を一つのプロセスとして解釈する方法

図式に対する基本演算として、次にあげる並列合成と逐次合成がある。

### 2.1 並列合成

同時に遂行する二つのプロセスを組み合わせる一つのプロセスにする**並列合成演算**は、横並びに配置された二つの図式から構成される。並列合成演算は、記号「 $\otimes$ 」を用いて表記する。



並列合成演算は結合的である。

$$\left( \begin{array}{c} | \\ \boxed{f} \\ | \end{array} \otimes \begin{array}{c} | \\ \boxed{g} \\ | \end{array} \right) \otimes \begin{array}{c} | \\ \boxed{h} \\ | \end{array} = \begin{array}{c} | \\ \boxed{f} \\ | \end{array} \begin{array}{c} | \\ \boxed{g} \\ | \end{array} \begin{array}{c} | \\ \boxed{h} \\ | \end{array} = \begin{array}{c} | \\ \boxed{f} \\ | \end{array} \otimes \left( \begin{array}{c} | \\ \boxed{g} \\ | \end{array} \otimes \begin{array}{c} | \\ \boxed{h} \\ | \end{array} \right)$$

そして、空図式を単位元とする。(空図式は、入力も出力もなく、また「何もしない」プロセスで、破線で囲まれた矩形を用いて表す.)

$$\begin{array}{c} | \\ \boxed{f} \\ | \end{array} \otimes \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \end{array} \otimes \begin{array}{c} | \\ \boxed{f} \\ | \end{array} = \begin{array}{c} | \\ \boxed{f} \\ | \end{array}$$

また、型  $A$  と  $B$  に対して、**結合型**と呼ばれる新たな型  $A \otimes B$  を構成できる.

$$A \otimes B \quad := \quad \begin{array}{c} | \\ \boxed{A} \\ | \end{array} \begin{array}{c} | \\ \boxed{B} \\ | \end{array}$$

したがって、箱につなぎこまれた複数の結線を、それらの型の結合型による1本の結線として扱うことができる。そして、「入力なし」、「出力なし」、あるいはその両方を表現するために、特別な「空」型を記号  $I$  によって表記する.

$$\begin{array}{c} I \\ | \\ \boxed{f} \\ | \\ A \otimes B \otimes C \end{array} = \begin{array}{c} \boxed{f} \\ | \\ \boxed{A} \quad \boxed{B} \quad \boxed{C} \end{array} \quad \begin{array}{c} A \otimes B \\ | \\ \boxed{g} \\ | \\ I \end{array} = \begin{array}{c} \boxed{A} \quad \boxed{B} \\ | \\ \boxed{g} \end{array} \quad \begin{array}{c} I \\ | \\ \boxed{s} \\ | \\ I \end{array} = \boxed{s}$$

## 2.2 逐次合成

一つのプロセスの結果を別のプロセスに渡す**逐次合成**演算は、ある図式の出力を別の図式の入力に結びつけることで構成される。逐次合成は記号「 $\circ$ 」を用いて表記し、それらの図式をその並び順に結びつける.

$$\left( \begin{array}{c} | \\ \boxed{g} \\ | \\ \boxed{f} \\ | \\ C \quad D \end{array} \right) \begin{array}{c} E \\ | \\ \boxed{a} \\ | \\ A \end{array} \begin{array}{c} F \\ | \\ \boxed{b} \\ | \\ B \end{array} \circ \left( \begin{array}{c} | \\ \boxed{a} \\ | \\ A \end{array} \begin{array}{c} | \\ \boxed{b} \\ | \\ B \end{array} \right) = \begin{array}{c} | \\ \boxed{g} \\ | \\ \boxed{f} \\ | \\ \boxed{a} \quad \boxed{b} \\ | \\ A \quad B \end{array}$$

逐次合成の単位元は、入力に対して「何もしない」箱であり、これは単なる結線とみなすことができる.

$$\begin{array}{c} | \\ \boxed{B} \\ | \end{array} \circ \begin{array}{c} | \\ \boxed{f} \\ | \\ A \end{array} = \begin{array}{c} | \\ \boxed{f} \\ | \\ A \end{array} \circ \begin{array}{c} | \\ \boxed{A} \\ | \end{array} = \begin{array}{c} | \\ \boxed{f} \\ | \\ A \end{array}$$

### 2.3 プロセス理論の例

たとえば、任意の集合を定義域および値域とする関数の集まりはプロセス理論 **functions** を構成する。 **functions** では、それぞれの関数がプロセスであり、その定義域が入力系、値域が出力系になる。また、 **functions** のプロセスの逐次合成は、通常関数の合成である。

集合  $A$  と集合  $B$  の間の関係（これは直積  $A \times B$  の部分集合である）を  $A$  を入力系とし  $B$  を出力系とするプロセスと考えると、任意の集合の間の関係の集まりはプロセス理論 **relations** を構成する。 **relations** のプロセスの逐次合成は、関係の結合演算 (join) である。

関数は関係の特別な場合とみなすことができ、 **functions** の逐次合成・並列合成はそれぞれ **relations** の逐次合成・並列合成の特別な場合とみなすことができるので、 **functions** は **relations** の部分理論である。

### 2.4 特別なプロセス

入力のないプロセスを**状態**と呼ぶ。状態は、入力がないことを強調するために、次のような逆三角形で表現する。



系  $A$  はいくつもの異なる状態をとることができるが、このプロセスはこれらの状態の中から選ばれた特定の状態  $\psi$  にある  $A$  型の系を出力する。入力となる結線がないということは、この系  $A$  がどこからきたのか知らず（あるいは気にせず）、ただそれが存在し、特定の状態になっていることだけを意味する。

一方、出力のないプロセスを**効果**と呼び、次のように表現する。



効果は、ある系を使ったあとに何も残さない、あるいは、単にあとに何が残るかは気にしないという意味で、状態と双対になる概念である。

状態と効果を逐次結合した**数**は、入力も出力もないプロセスである。

$$\begin{array}{c} \triangle \\ \pi \\ | \\ \hline \end{array} \circ \begin{array}{c} | \\ \hline \triangle \\ \psi \end{array} = \begin{array}{c} \triangle \\ \pi \\ | \\ \hline \triangle \\ \psi \end{array}$$

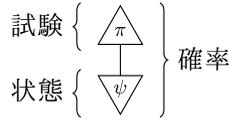
数は、ちょうど状態と効果の三角形を組合せた次のような正方形によって表現する。



あるいは、単に次のように表現することもある。

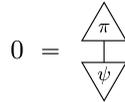
$\lambda$

状態が効果と出会うと数が出現するという事実を，系が特定の状態にあるときにその効果が生じる確率を与えると解釈する．あるいは，状態  $\psi$  に対して効果  $\pi$  の試験を行ったときに，答えとして YES が得られる確率と解釈する．

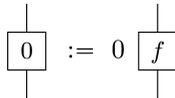


これを一般化ボルン則と呼ぶ．

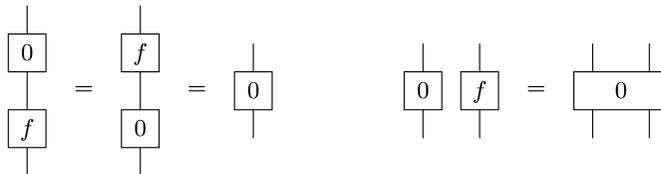
そうすると，状態  $\psi$  に対して効果  $\pi$  が生じることが起こりえないことを意味する



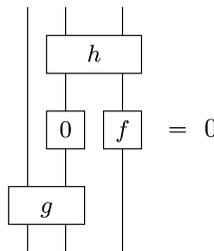
は，不可能を表すべきである．したがって，



は任意のプロセス  $f$  に対して不可能となるべきである．そこで，考えるすべての入力と出力の型に対してゼロプロセスがあり，次の合成則に従うものとする．



このようにして，すべての入出力型に対してゼロプロセスを考えることができるので，それを単に「0」と書き，その入出力線は省略する．



すなわち，図式において，数としての0だけでなく，数0とほかのプロセスを並列合成あるいは逐次合成したプロセスも単に0と表記する．0は空図式ではないことに注意せよ．

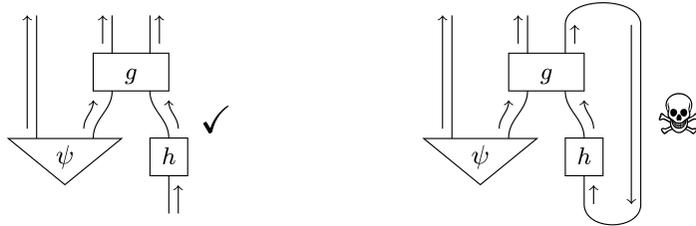
### 3. 回路図式とストリング図式

状態，効果，数を含めたプロセスを，並列結合と逐次結合によって組み合わせた図式に対して，結合のしかたについての制約を変えることで回路図式とストリング図式という図

式のクラスを導入する。また、箱の縦方向および横方向の裏返しによって、図式言語を拡張する。

### 3.1 $\cup$ と $\cap$

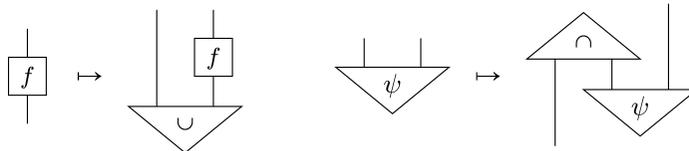
それぞれの結線はそれを出力とする箱から入力とする箱に向かって向き付けられていると考えると、回路図式は次の左側の図式のように有向閉路をもたない図式である。(右側の図式には有向閉路があるので回路図式ではない。)



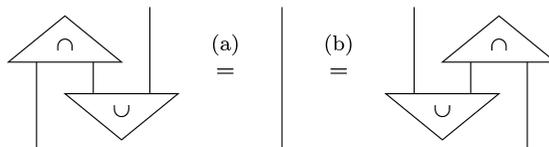
二部状態は、二つの系を出力とする状態である。プロセスと二部状態を相互に変換するために、それぞれの系  $A$  に対して次のような特定の状態  $\cup$  (カップ) と効果  $\cap$  (キャップ) を固定する。



プロセスと二部状態の対応づけ：



は、 $\cup$  と  $\cap$  が次の関係にあるとき、互いの逆操作になる。



この対応づけをプロセス・状態双対性と呼ぶ。プロセス理論 relations ではプロセス・状態双対性が成り立つが、プロセス理論 functions では成り立たない。(functions の任意の二部状態は  $\otimes$  分離可能、すなわち、二つの状態の並列合成として表すことができるので、プロセス・状態双対性が成り立つとしたら、すべてのプロセスは  $\circ$  分離可能、すなわち出力は入力とは無関係に決まることになるからである。)

等式(a), (b)はあまり直感的ではないので、この特別な状態と効果をそれぞれ  $\cup$  状および  $\cap$  状の結線として描くことにする。



そうすると，等式 (a)，(b) は次のようになる。



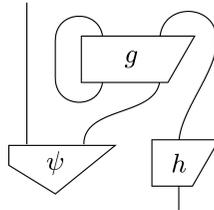
また，つながり方を変えずに箱を動かす操作によって変形させた図式どうしは等しくなつてほしいので，次のような等式も成り立つと考える。



これら 4 個の等式を引き伸ばし等式と呼ぶ。

### 3.2 スtring 図式

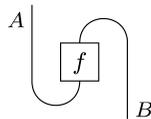
String 図式は，回路図式において有向閉路を禁止せず，さらに入力系どうしや出力系どうしをつなぐことも許した図式である。



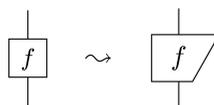
String 図式は，回路図式に特別なプロセスとして引き伸ばし等式を満たす  $\cup$  と  $\cap$  を追加したものとみることが出来る。

### 3.3 転置

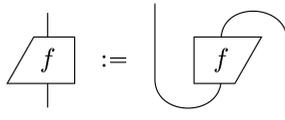
プロセス  $f$  の転置  $f^T$  とは，次のようなプロセスである。



互いに転置の関係にあるプロセスがひと目で分かるように，プロセスの箱の形を少し変える。



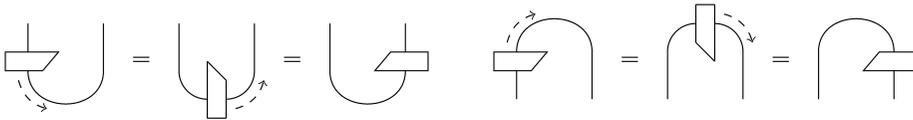
そして， $180^\circ$  回転させた箱に  $f$  と名前をつけたもので  $f$  の転置を表す。



すると、任意のプロセス  $f$  に対して、次の等式が成り立つ。



これは、 $\cup$  状や  $\cap$  状の結線に沿って箱を滑らせて移動させているかのように見える。

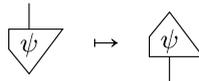


また、状態と効果も、回転させたことを明確にするために、次のように角を切り落とす。



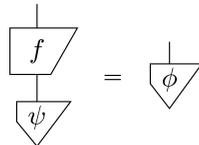
### 3.4 随伴

効果は、ある性質について状態を試験しているものと解釈できる。図式言語を拡張して、系が状態  $\psi$  にあるかを試験する効果を次のような縦方向の裏返しによって表現する。

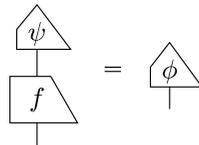


この効果を状態  $\psi$  の**随伴**と呼ぶ。

この縦方向の裏返し操作は、すべてのプロセスに自然に拡張される。 $f$  が次のように状態  $\psi$  を状態  $\phi$  に変換するならば、

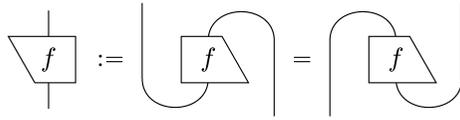


$f$  の**随伴**  $f^\dagger$  は、 $\psi$  の随伴を  $\phi$  の随伴に変換するプロセスである。

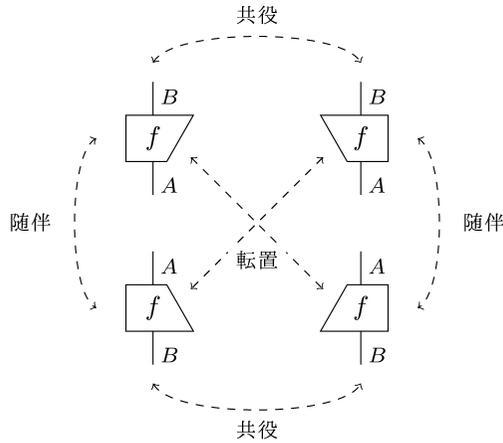


### 3.5 共役

プロセスの**共役**とは、随伴の転置（これは転置の随伴に等しい）である。



ここまでに示した箱に対する操作をすべてを合わせると、箱は次のような4種類の形式で現れる。



### 4. 線形写像のプロセス理論

この章では、量子論のプロセス理論を構成するための出発点として、(有限次元) ヒルベルト空間を系とし線形写像をプロセスとするプロセス理論 **linear maps** を構成する。

型  $A$  の**正規直交基底**とは、次の条件を満たす  $A$  の状態の集合  $B$  である。

$$\forall |i\rangle, |j\rangle \in B: \begin{array}{c} \triangle \\ |j\rangle \\ \triangle \\ |i\rangle \end{array} = \delta_i^j$$

かつ

$$\left( \forall |i\rangle \in B: \begin{array}{c} \triangle \\ f \\ \triangle \\ |i\rangle \end{array} = \begin{array}{c} \triangle \\ g \\ \triangle \\ |i\rangle \end{array} \right) \Rightarrow \begin{array}{c} \triangle \\ f \\ \triangle \end{array} = \begin{array}{c} \triangle \\ g \\ \triangle \end{array}$$

型  $A$  の**次元**  $\dim(A)$  とは、 $A$  の基底の大きさの最小値である。

プロセス理論が**和を許容する**とは、同じ入出力型をもつプロセスの任意の集合  $\{f_i\}_i$  に対して、それらと同じ入出力型をもつプロセス



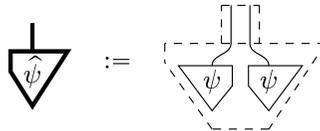
$$\tilde{H} := \left\{ \begin{array}{c} \downarrow H \\ \psi \end{array} \mid \psi \text{は型 } H \text{ の状態} \right\}$$

5. 量子写像のプロセス理論

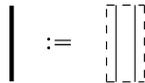
この章では、線形写像のプロセス理論 linear maps から、まず図式を二重化することによって純粋量子写像のプロセス理論 pure quantum maps を構成し、それに廃棄プロセスを追加することで量子写像のプロセス理論 quantum maps を構成する。また、非決定性を表現するために、量子写像の集まりによって量子プロセスを定義する。

5.1 プロセス理論の二重化

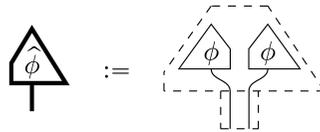
ヒルベルト空間  $A$  の任意の状態  $\psi$  に対して、 $\psi$  を二重化した新たな状態  $\hat{\psi}$  を定義する。



そして、 $\hat{\psi}$  は新たな型  $\hat{A}$  を出力するものとする。 $\hat{A}$  の実体は、 $A$  の複製が2個あるものにはすぎない。

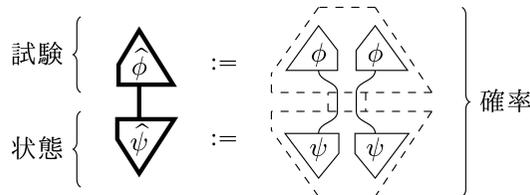


同様にして、この新たな型  $\hat{A}$  に対して次のような効果を定義する。



二重化された状態と効果をそれぞれ純粋量子状態と純粋量子効果と呼ぶ。

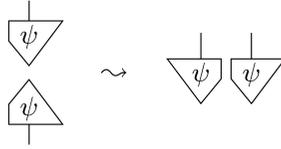
この新たな状態と効果を合わせると、量子論のボルン則は次のように一般化ボルン則の特別な場合として定義される。



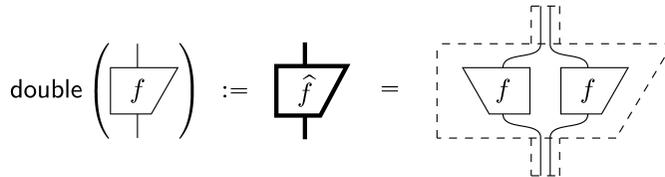
二重化は、量子論における純粋状態ベクトル  $|\psi\rangle$  からそれに付随した次のような密度作用素への移行に対応する。

$$\tilde{\psi} := |\psi\rangle\langle\psi|$$

これは、二重化された状態と同じ情報を持ち、効果  $\psi$  を転置させて状態  $\psi$  の共役にしたものと見ることもできる。



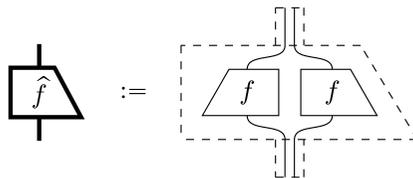
状態と効果だけでなく、任意の線形写像  $f$  も二重化できる。



この二重化によって得られる写像を**純粋量子写像**と呼ぶ。

純粋量子写像のプロセス理論 **pure quantum maps** は、**linear maps** のすべてのヒルベルト空間  $A$  に対する  $\hat{A}$  を型としてもち、**linear maps** のすべてのプロセス  $f$  に対する二重化された線形写像  $\hat{f}$  をプロセスとしてもつ。**pure quantum maps** のプロセスは、実際にはまったく新しいものではなく、特別な種類の線形写像にすぎない。言い換えると、**pure quantum maps** は **linear maps** の部分理論である。とくに、**linear maps** がストリング図式を許容するという事実は、**pure quantum maps** に引き継がれる。

二重化された引き伸ばし等式は成り立ち、また二重化によって転置、逐次合成、並列合成は保たれ、二重化される前の理論から随伴も引き継ぐ。



これらを合わせると、二重化はストリング図式を保つ。すなわち、ストリング図式の等式は二重化しても成り立つ。

## 5.2 系の廃棄

系の廃棄は、系を無視する、または破壊するという意味かもしれないし、二度と見る事ができないように宇宙空間に送り込むことかもしれない。廃棄は、図式から状態を取り除くこと以外何もすべきではない。言い換えると、廃棄は確実に成功する試験であるが、その状態についてそれ以外に分かることはない。とくに、廃棄プロセスは廃棄しようとする系の状態に依存してはならない。

そこで、**廃棄**は次のような効果と定義する。

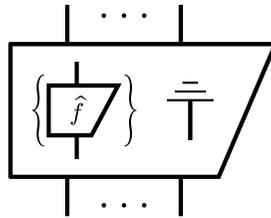
$$\overline{\top} := \text{[Diagram: A top wire with a bar over it, connected to a U-shaped wire structure inside a dashed box.]}$$

このとき、正規化された任意の純粋量子状態  $\hat{\psi}$  に対して、次の等式が成り立つ。

$$\overline{\top} \text{ [Diagram: A top wire with a bar over it, connected to a downward-pointing triangle containing } \hat{\psi} \text{]} = \text{[Diagram: A dashed rectangular box.]}$$

### 5.3 量子写像

量子写像のプロセス理論 **quantum maps** は、二重化されたヒルベルト空間  $\hat{A}$  を型とし、純粋量子写像と廃棄から作られるすべての図式をプロセスとする。



プロセス理論 **quantum maps** もストリング図式を許容する。なぜなら、**quantum maps** は **pure quantum maps** から  $\cup$  と  $\cap$  を引き継ぎ、純粋量子写像の随伴は純粋量子写像であり、廃棄の随伴は  $\cup$  と廃棄の合成であるので、すべての量子写像は随伴をもつからである。

任意の量子写像  $\Phi$  に対して、次の等式を満たす純粋量子写像  $\hat{f}$  が存在する。

$$\Phi = \overline{\top} \text{ [Diagram: A trapezoidal box with } \hat{f} \text{ inside and } \overline{\top} \text{ on top]} := \text{[Diagram: A dashed box containing two } f \text{ boxes connected to } \overline{\top} \text{ and wires]} = \text{[Diagram: A dashed box containing two } f \text{ boxes connected to wires]}$$

これは、任意の量子写像が二重化されたプロセスの出力の一部を無視して生じることを示している。この純粋量子写像  $\hat{f}$  を量子写像  $\Phi$  の**純粋化**とよぶ。

量子写像  $\Phi$  は次の関係が成り立つならば、**因果的**という。

$$\overline{\top} \text{ [Diagram: A trapezoidal box with } \Phi \text{ inside and } \overline{\top} \text{ on top]} = \overline{\top}$$

これは、「プロセスの出力が廃棄されるならば、そのプロセスは起きなかったものとみな



### 6.1 正規直交基底による測定

正規直交基底による破壊的測定は、任意の正規直交基底

$$\left\{ \begin{array}{c} \downarrow \\ i \\ \nabla \end{array} \right\}_i$$

に対して次の形式をした量子プロセスであり、添字  $i$  は測定結果と呼ばれる。

$$\left( \begin{array}{c} \uparrow \\ i \\ \uparrow \end{array} \right)^i$$

正規直交基底による破壊的測定が量子プロセスであるのは、(二重化されていない) 正規直交基底に対して

$$\left| \begin{array}{c} \downarrow \\ j \\ \nabla \\ \uparrow \\ i \\ \nabla \end{array} \right. = \sum_i \delta_i^j \left( \begin{array}{c} \downarrow \\ i \\ \nabla \\ \uparrow \\ i \\ \nabla \end{array} \right) = \sum_i \left( \begin{array}{c} \downarrow \\ i \\ \nabla \\ \uparrow \\ i \\ \nabla \end{array} \right)$$

が成り立ち、プロセス・状態双対性を用いると、二重化された正規直交基底の総和は廃棄と等しくなる、すなわち因果的だからである。

二重化された正規直交基底の効果のかわりに、効果と状態の対を考えると非破壊的測定になる。

$$\left( \begin{array}{c} \downarrow \\ i \\ \nabla \\ \uparrow \\ i \\ \nabla \end{array} \right)^i$$

この場合も次の等式が成り立つ、すなわち因果的なので、量子プロセスである。

$$\sum_i \left( \begin{array}{c} \overline{\downarrow} \\ i \\ \nabla \\ \uparrow \\ i \\ \nabla \end{array} \right) = \sum_i \left( \begin{array}{c} \uparrow \\ i \\ \uparrow \end{array} \right) = \overline{\uparrow}$$

測定の結果は結果状態と呼ばれる。

正規直交基底による非破壊的測定が与えられたとき、結果の系を廃棄するだけで対応する破壊的測定が得られる。



## 7. 量子系と古典系の統合

5章で構成した純粋量子写像のプロセス理論 **pure quantum maps** に対して、量子写像の集まりによって（量子測定を含めた）非決定的量子プロセスを表現し、単一結線を復活させて古典的データを表現することで古典・量子写像のプロセス理論 **quantum-classical maps** を構成する。そして、それに因果性の制約を加えると量子論のプロセス理論 **quantum processes** になる。この **quantum processes** の図式によって、典型的な量子的振る舞いである量子テレポーテーションを図示する。

### 7.1 古典的データの表現

量子系はすでに太い二重化された結線によって表現されているので、新種の系を導入するために単独の結線を復活させる。これが古典系を表現するのにぴったり当てはまることが判明する。

$$\frac{\text{古典系}}{\text{量子系}} = \frac{\text{細い単独の結線}}{\text{太い二重の結線}}$$

固定した正規直交基底の状態によって古典的な値を表現することによって、古典的なデータを単独の結線に変換する。そうすると、これは、 $D$ 次元の古典系上の正規直交基底に対応して、古典的なデータが  $D$ 通りの値をとりうることを意味する。たとえば、1ビットは、2次元の系の上の正規直交基底に対応する。

$$\text{「ビット」} := \left\{ \begin{array}{c} \downarrow \\ \triangle \\ 0 \end{array}, \begin{array}{c} \downarrow \\ \triangle \\ 1 \end{array} \right\}$$

この正規直交基底の状態を次のように解釈する。

$$\begin{array}{c} \downarrow \\ \triangle \\ i \end{array} := \text{「古典的な値 } i \text{ をもたらす」}$$

これに対応する効果は次のようになる。

$$\begin{array}{c} \triangle \\ i \end{array} := \text{「古典的な値 } i \text{ を試験する」}$$

正規直交性は、次のように完全につじつまが合う。

$$\begin{array}{c} \triangle \\ j \\ \downarrow \\ \triangle \\ i \end{array} = \delta_i^j$$

正規直交基底による測定の量子プロセスを古典的結線を用いた新しい表現に変換する。

$$\left( \begin{array}{c} \triangleup \\ \uparrow \\ i \end{array} \right)^i \sim \sum_i \begin{array}{c} \downarrow \\ i \\ \triangleup \\ i \\ \uparrow \end{array}$$

この線形写像は重要なので、専用の記法を用意する。

$$\begin{array}{c} \circ \\ | \end{array} := \sum_i \begin{array}{c} \downarrow \\ i \\ \triangleup \\ i \\ \uparrow \end{array}$$

この写像を以降では**測定**と呼ぶ。測定を任意の量子状態に適用すると次のようになる。

$$\begin{array}{c} \circ \\ \triangleup \\ \rho \end{array} = \sum_i \begin{array}{c} \downarrow \\ i \\ \triangleup \\ i \\ \triangleup \\ \rho \end{array} = \sum_i P(i | \rho) \begin{array}{c} \downarrow \\ i \end{array}$$

ただし、 $P(i|\rho)$ は、状態 $\rho$ を正規直交基底で測定したときに結果 $i$ が得られる確率である。したがって、この線形写像は、次のように量子状態をそれがとりうる測定結果すべてにわたる確率分布へと移す。

$$\begin{array}{c} \downarrow \\ \rho \end{array} \mapsto \sum_i P(i | \rho) \begin{array}{c} \downarrow \\ i \end{array}$$

この測定の随伴を考えることもできる。

$$\begin{array}{c} \circ \\ | \end{array} := \sum_i \begin{array}{c} \downarrow \\ i \\ \triangleup \\ i \\ \uparrow \end{array}$$

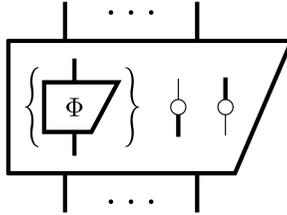
この線形写像を任意の確率分布に適用すると、確率分布を量子状態に符号化したものが得られる。

$$\begin{array}{c} \circ \\ \triangleup \\ p \end{array} = \sum_i \begin{array}{c} \downarrow \\ i \\ \triangleup \\ i \\ \triangleup \\ p \end{array} = \sum_i p^i \begin{array}{c} \downarrow \\ i \end{array}$$

この線形写像を**符号化**と呼ぶ。符号化と測定を用いると、古典的なデータと量子状態を相互に変換することができる。

### 7.2 古典・量子写像 (cq 写像)

古典・量子写像 (cq 写像) は、量子写像, 符号化, 測定を合成して得られる線形写像である.



これに対応するプロセス理論を **classical-quantum maps** (略して **cq-maps**) と呼ぶ. **cq-maps** はストリング図式を許容する.

また, 因果性の概念を cq 写像にまで拡張する. そのために, 古典系において廃棄に対応する次のような削除を導入する.

$$\text{circle with vertical line} := \sum_i \text{triangle with } i \text{ and vertical line}$$

実際には, 削除は符号化と廃棄の合成であり, すでに **cq-maps** に含まれている.

$$\text{double horizontal line over circle with vertical line} = \sum_i \text{triangle with } i \text{ and vertical line over circle with vertical line} = \sum_i \text{triangle with } i \text{ over vertical line} = \text{circle with vertical line}$$

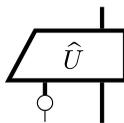
そして, 次の等式が成り立つとき, cq 写像は因果的であるという.

$$\text{circle with vertical line over double horizontal line over } \Phi \text{ box} = \text{circle with vertical line} \text{ over double horizontal line}$$

量子論のプロセス理論 **quantum processes** は, **classical-quantum maps** のプロセスに因果的であるという制限をつけた部分理論である.

### 7.3 制御付きユニタリ変換

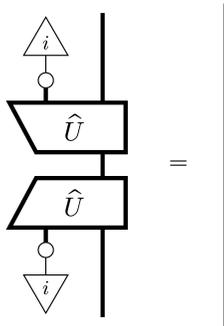
次節で図式を用いて量子テレポーテーションを表現するために, 制御付き等長変換を次のような cq 写像として表現する.



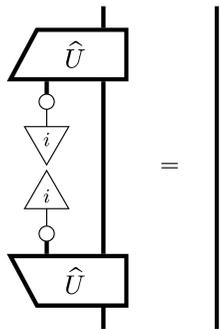
この量子プロセスの分枝である個々の等長変換は, 次のようにして復元できる.



したがって、制御付き等長変換は、すべての  $i$  に対して次の等式が成り立つような cq 写像である。

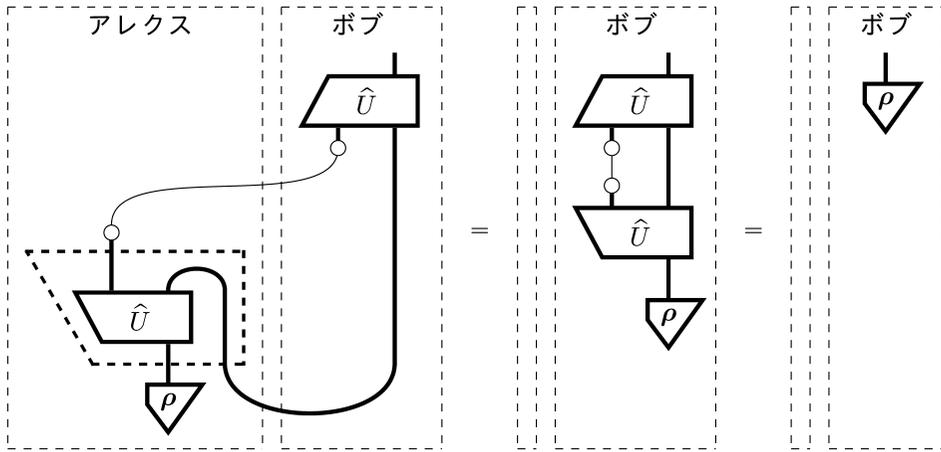


また、制御付きユニタリ変換は、加えてすべての  $i$  に対して次の等式が成り立つ。



#### 7.4 図式を用いた量子アルゴリズムの例

ここでは quantum processes の図式として表現することで、量子テレポーテーションによって量子状態  $\rho$  がアレクス（送信者）からボブ（受信者）にどのように送られるかを示す。



当初、 $\cup$ の一端と量子状態 $\rho$ を保持しているアレクスは、それらに対して量子写像 $\hat{U}$ を遂行し、その出力を測定して得られた古典的データをボブに送る。 $\cup$ の另一端を保持しているボブは、アレクスから受け取った古典的データを使って制御付きユニタリ変換を遂行する。ここで、正規直交基底による測定は、ユニタリ変換のあとに測定プロセスを続けたものとして表すことができることから、破線で囲んだプロセスは制御付きユニタリ変換になる。制御付きユニタリ変換とその随伴が相殺することから、ボブは状態 $\rho$ そのものを受け取ることができるのである。

### 8. おわりに

このように図式に基づくプロセス理論は、量子系だけでなく古典系も含めた量子計算を表現することができる。そして、量子アルゴリズムを図式を用いて記述する利点として次のことが挙げられる。

- 図式で示すことにより直感的な理解の助けとなる。
- 具体的な系やプロセスを明示せずに計算を表現できるために、一般化された計算を提示することができる。
- 量子系だけでなく古典的データも含んだ量子計算を図式として扱うことができ、図式だけで計算全体を表すことができる。
- 量子測定も一種のプロセスとして扱われるので、統一した枠組みで計算が記述できる。

今後は、このような図式言語で記述された量子アルゴリズムから、図式言語にはない制御構造などを付与することによって実際の量子計算機で実行できるプログラミング言語へと変換する研究につなげたい。

- 参考文献** [1] Coecke, B. *The logic of entanglement. An invitation*. Tech. rept. RR-03-12. Department of Computer Science, Oxford University, 2003.
- [2] Coecke, B. The Logic of Entanglement. van Breugel, F., Kashefi, E., Palamidesi, C., and Rutten, J. (eds), *Horizons of the Mind. A Tribute to Prakash Panangaden: Essays Dedicated to Prakash Panangaden on the Occasion of His 60th Birthday*. LNCS No.8464. Springer, 2014, pp 250-267.
- [3] Coecke, B. and Kissinger, A. *Picturing Quantum Processes*. Cambridge Univ. Press, 2017. (邦訳は『図式 (ダイアグラム) による量子論 (仮題)』として森北出版より 2020 年 11 月発刊予定.)
- [4] Kauffman, L. H. Teleportation topology. *Optics and Spectroscopy* 99, 2005, pp 227-232.
- [5] Penrose, R. Applications of negative dimensional tensors. In Welsh, D. (ed), *Combinatorial Mathematics and its Applications*. Academic Press, 1971, pp 221-244.

**執筆者紹介** 川 辺 治 之 (Haruyuki Kawabe)

1985 年日本ユニシス(株)入社. Lisp マシン・UNIX 等オープン系基盤ソフトウェアの開発・保守, ASP 事業の企画・運営, 半構造化データベースの研究開発などに従事. 2014 年より総合技術研究所に所属. 現在の研究開発テーマは量子ソフトウェア開発.

