基専般

解説

可逆コンピューティング

—ビリヤードボールでコンピュータが作れるか?—

■ 森田 憲一　広島大学大学院工学研究院情報部門

可逆コンピューティングとは

可逆的な計算システムというのは，そのシステムのどの状態も，直前の時刻にとり得る状態を高々1つしか持たないようなシステムのことである．つまりシステム全体の状態の時間的遷移が**図-1**のようになるものをいう．一見すると些細でつまらない性質のように見えるこの制約が物理的可逆性（時間の逆方向にも同じ法則が成り立つという性質）と密接にかかわっており，また計算におけるエネルギー消費の問題を考えるときに重要な役割を果たすのだ，ということをLandauer[3)] が指摘して以後，この計算のパラダイムが注目されるようになった．

Landauerは，計算機において不要な情報を消去するなど，元の状態に戻れないという意味で非可逆な演算を実行すると必ず熱の発生を伴うことを指摘した．これはランダウアの原理（Landauer's principle）として知られている．具体的には，1ビットの情報が消去されると，$kT \ln 2$ のエネルギーが熱として計算機の外部に放出されるということである（k はボルツマン定数，T は絶対温度）．一方，可逆的な演算の場合にはこういった不可避的な熱の発生をなくせる可能性があり，理想的な状況下ではエネルギー消費が0であるような可逆計算機につながる．けれども，現在の計算機で用いられている電子素子は，演算が可逆か非可逆かに関係なくエネルギーを贅沢に消費するようなやり方で構成されているので，$kT \ln 2$ という量はまったく無視できるほど小さく，今のところ，この分を節約しても焼け石に水である．

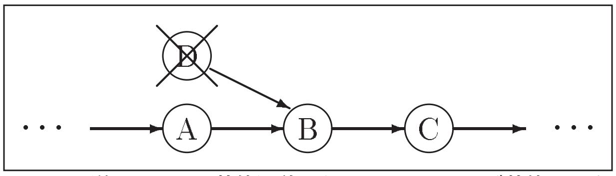

図-1　可逆システムの状態遷移．もしこのシステムが状態Aから Bに遷移するならば，Aとは異なる状態DでBに遷移するものがあってはならない．

現在の電子素子では，論理値の0と1が多数の電子の平均的振舞いによって表現・区別されており，0あるいは1を記憶しているというような1つの巨視的状態に対応する微視的状態が莫大な数になる．しかしながら，近年における計算機素子の急速な微細化からかんがみて，将来的には論理的状態を少数個の微視的状態で実現できる（あるいはそうせざるを得ない）状況になると予想される．そのようになると，論理的な可逆性がエネルギー消費を大幅に減らすための重要な鍵になってくる．またそもそも量子力学が支配するような微視的世界では物理法則が可逆であるため，消費エネルギーの問題だけでなく，そのような自然の法則が将来の計算機にどのような制約を与えるのか，また逆にこの性質をどのように利用すれば計算機をうまく構成して微細化できるのかが大変重要な課題となってくる．

本稿では，可逆的な計算機構が可逆的な論理素子によって，さらにはそれが可逆的な物理的システムによってどのように構成できるのかという問題を，可逆チューリング機械，可逆論理素子，ビリヤードボールモデルなどの理論的計算モデルに基づいて述べる．実用的な可逆計算機を実現するまでの道のりはまだ長いが，ここでは，可逆コンピューティング

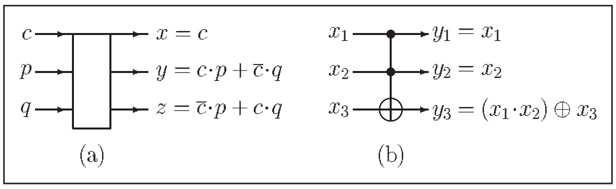

図 -2　(a) フレドキンゲートと (b) トフォリゲート．$\oplus$ は排他的論理和を表す．

の世界にどのような新しい設計のアイディアがあり得るのか，その可能性をこれらの計算モデルを用いた例によって説明する．

可逆論理素子とは

可逆論理素子というのは，その演算・動作を表現する関数が単射である演算素子である．たとえば，論理ゲートの NOT はこれを表す関数が単射なので，可逆的なゲートである．出力が 1 なら入力は 0，出力が 0 なら入力は 1 に，それぞれ一意に決まるからである．一方 AND ゲートは，それを表す関数が単射ではなく，非可逆である．なぜなら，出力が 0 になるような入力は (0,0), (0,1), (1,0) の 3 通りあるからである．可逆論理ゲートは，入出力線数を制限しなければ無限に多くの種類が存在するが，その中で特に有名なものはフレドキンゲート（**図 -2 (a)**）とトフォリゲート（**図 -2 (b)**）である．これらはいずれも 3 入力 3 出力のゲートであり，出力を知ればそのときの入力を一意に決定できることが容易に確かめられる．さらにこれらは「万能」であることが知られている．AND, OR, NOT からなる非可逆な論理回路を，これらの可逆論理ゲートだけからなる回路中に埋め込めるし [2)]，また任意の可逆順序機械や可逆チューリング機械（後述）をこれらゲートと記憶素子だけで構成することもできる（たとえば文献 6）参照）．

可逆論理ゲートとそれによる可逆論理回路の研究はこのように多くなされてきたのだが，ここでは敢えて可逆論理ゲートとはタイプの異なる素子を紹介し，それについて述べる．それは「記憶つき可逆論理素子」である．このような素子にも無限に多くの種類があるが，ここではロータリー素子と呼ぶ特定の素子を考える [4),6)]．これは 1 ビットの記憶を持つ，つまり 2 状態の素子であり，各 4 本ずつの入力線と出力線を持っている．この素子に対しては直観的にとても理解しやすい解釈がある．この素子は概念的には**図 -3** のような，内部に回転可能な棒を持つ機構と見なされ，2 つの状態は棒の方向によって区別される．その動作もいたって簡単で，棒と平行に粒子のような信号が与えられると粒子は直進して出てゆき，もとの状態を保持する（**図 -4 (a)**）が，棒に垂直に粒子が与えられると粒子は右折し，状態が変化する（**図 -4 (b)**）．このとき，時刻 t+1 における素子の状態と出力を知れば，時刻 t の状態と入力が唯一に定まることに注意しよう．この素子はこの意味で可逆である．なお，素子の 2 つ以上の入力線に同時に粒子が入力されることはないと仮定している．可逆性を保ったままで，2 つ以上の入力線に粒子が入力された場合の動作を定義することも可能なのだけれども，ここでは未定義とする．これはそのほうが定義が簡単な上，可逆チューリング機械などの可逆コンピュータを構成するには当面はこの機能だけで十分だからである．論理ゲートの場合には各入力線が 0 と 1 の値を独立にとることができるが，この種の記憶つき論理素子の場合はその点がまったく異なっているので注意が必要である．

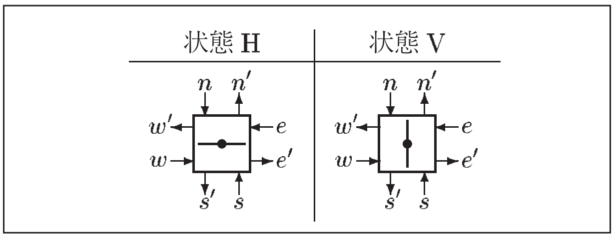

図 -3　ロータリー素子の 2 つの状態を表す概念図

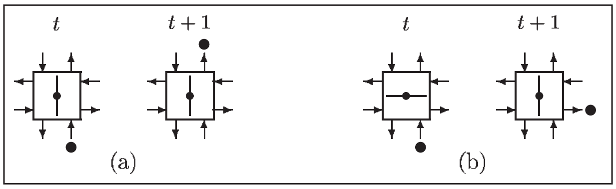

図 -4　ロータリー素子の動作．(a) 粒子が棒と平行に入力された場合と，(b) 棒と垂直の場合．図中の粒子●は時刻 t においては入力，t+1 においては出力と解釈される．

記憶つき可逆論理素子は，可逆的な順序機械と見

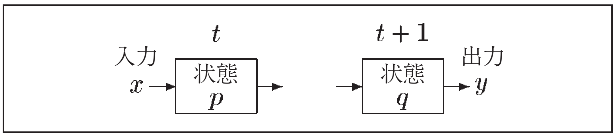

図 -5　順序機械．時刻 t の状態 p と入力 x によって時刻 t+1 の状態 q と出力 y が決まる．

現在の状態	入力 n	入力 e	入力 s	入力 w
状態 H:	w'	w'	e'	e'
状態 V:	s'	n'	n'	s'

表 -1　ロータリー素子の動作を示す表．現在の状態と入力から次の時刻の状態と出力が定められる．

なすことができる．順序機械 (sequential machine) というのは，出力を持つような有限オートマトンにほかならず，**図 -5** のように，現在の状態と入力記号から次の時刻の状態と出力が決定されるものである．ロータリー素子は，2 つの状態，4 種類の入力記号，4 種類の出力記号を持つ順序機械であり，その動作は**表 -1** のように定められる．たとえば，現在の時刻 t の状態が H で入力記号が n ならば，次の時刻 t+1 の状態が V で出力記号が w' となる．そして可逆順序機械というのは，時刻 t+1 の状態と出力から，時刻 t の状態と入力が一意に決まるものである．表 -1 を見ると，次の時刻の状態と出力の組合せで同一のものが表中にないので，そうであることが確かめられる．ここではこのような素子を組み合わせて可逆論理回路を，さらには可逆コンピュータを作ろうとしているのだが，その前に，可逆論理素子の物理的実現の可能性について考えてみよう．

ビリヤードボールで可逆論理素子を実現する

可逆的な計算システムは，それの各々の計算状況が直前の計算状況を高々 1 つしか持たない，という簡単な制約を課したものだが，それにもかかわらず可逆的な物理システムと密接な関係を持っている．次に述べるビリヤードボールモデル (BBM) は，この関係をとても分かりやすく示してくれる．BBM は弾性衝突をする理想的なボールと反射板によって計算を遂行する可逆的な物理モデルであり，

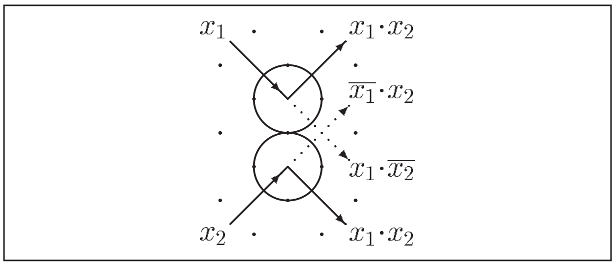

図 -6　可逆的な物理モデルであるビリヤードボールモデルによる相互作用ゲートの実現 [2)]

Fredkin と Toffoli[2)] によって提案された．**図 -6** は，相互作用ゲートと呼ばれる 2 入力 4 出力の可逆論理ゲートの BBM における実現法である．これは単に 2 つのボールが衝突するだけであるが，これによって AND や NOT を実現することもできる．彼らはさらに，これによってフレドキンゲートが実現できることも示した．ところで，ロータリー素子はフレドキンゲートと信号の遅延（つまりは記憶素子）によって実現できる（文献 6）参照）ので，まずフレドキンゲートを BBM の中で実現し，それを組み合わせてロータリー素子を BBM 中に実現することが可能ではある．しかしロータリー素子をゲートと記憶素子に分けて構成しようとすると面倒なことになる．というのは，一般にゲートを BBM の中で実現しようとした場合，複数のボールを定められた位置に完全に同じタイミングで到着させなければならず，遅延時間の調整が非常に厄介になるからである．

しかし，ロータリー素子を，フレドキンゲートを介さず，直接的に BBM 中に実現する簡単な方法がある．それが**図 -7** である．図中の小さな長方形はボールをはね返すための反射板である．ロータリー素子は 2 つの状態 H と V を持つが，それらは図 -7 (a) の H または V の位置に静止した（薄い色の）ボールを置くことによって区別される．図 -7 (b) は，状態が V で入力が s の場合を示している．このときの入力は，運動する濃い色のボールを s と書かれた入力ポートから任意の速度で入れてやればよい．この場合，濃い色のボールは，V の位置で静止している薄い色のボールとも，反射板とも接触することはなく 0 番の位置から 1 番の方向に向かって直進

し，出力ポート n' から出てゆく．これにより図 -4 (a) の動作がシミュレートされる．図 -7 (c) は，状態が H で入力が s の場合である．このときも，運動する濃い色のボールを s と書かれたポートに与える．そうすると，そのボールは 1 番の位置で薄い色のボールと衝突し，それにより後者のボールも動き出す．その後それぞれのボールは 2 番から 5 番までの位置を経由して，再び 6 番の位置で衝突することになる．そうすると薄い色のボールはその位置（つまり状態 V に対応する位置）で停止し，一方，濃い色のボールは 7 番の方向に向かって進み，e' のポートから出てゆく．薄い色のボールが 6 番の位置で停止するのは，その過程が 1 番での衝突の過程を時間反転した過程と同じだからである．以上により図 -4 (b) の動作が正しくシミュレートされる．上記以外の状態と入力の組合せについても同様である．以上のようなやり方で実現してやれば，2 つのボールのタイミングを考慮する必要があるのは，それらが衝突してから再衝突するまでの間だけに限定されるのでやりやすい．

さて，ビリヤードボールで可逆コンピュータが本当に作れるのだろうか？　BBM は明らかに思考実験上のモデルである．このような機械的な仕組みを正しく働かせるには，ボールや反射板の形状・位置などに無限の精度が要求される上に，摩擦のない理想的な環境が必要なので実際に作るのは不可能である．しかしながら，たとえばロータリー素子を BBM で実現したときの静止したボールで表されていた状態は，必ずしも運動可能なボールである必要はなく，外部入力によって状態が可逆的に変化するような物理的状態（たとえば何らかの量子的状態）であればよく，他の物理的実現法の可能性も示唆してくれる．

可逆論理回路を構成する

可逆論理素子を用いて可逆コンピュータを構成する話に戻ろう．そのためには，可逆論理素子を多数接続することにより回路を構成する必要がある．可

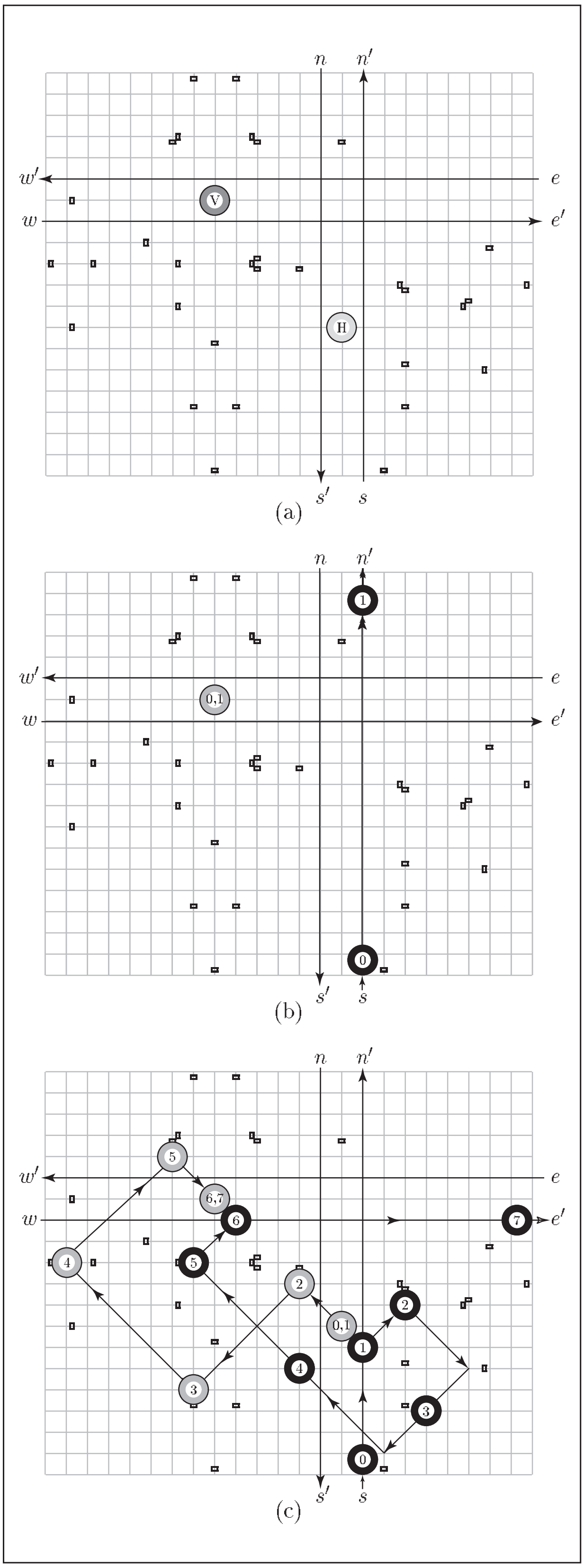

図 -7　ビリヤードボールモデルによるロータリー素子の直接的な実現法 [4), 6)]．小さな長方形は反射板である．(a) は素子の状態 H と V を表すための静止ボールを置く位置を示す．(b) は状態が V で入力が s の場合の動作（図 -4 (a) に対応）を，(c) は状態が H で入力が s の場合の動作（図 -4 (b) に対応）を表す．

現在の状態	入力 a_1	入力 a_2
q_1	q_2b_1	q_3b_2
q_2	q_2b_2	q_1b_1
q_3	q_1b_2	q_3b_1

表 -2　可逆順序機械 M_1 の動作を定める表

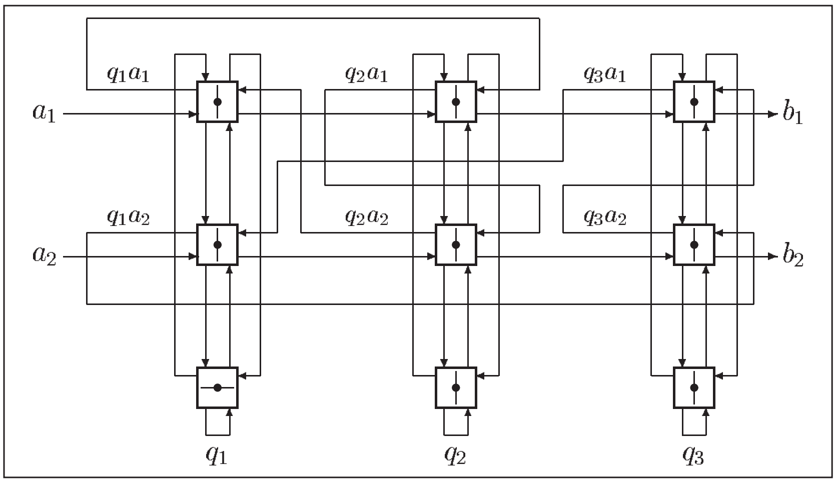

図 -8　可逆順序機械 M_1 を実現するロータリー素子による回路．この図では M_1 の状態は q_1 である．

逆論理回路では素子の接続の際に次の制約を満たさないといけない．それは，素子の各出力線は他の素子またはその素子自身の高々１つの入力線にしか接続できない，つまり出力の分岐は許されないという制約である．また，普通の非可逆な論理回路の場合と同様，2 つ以上の出力線を併合して１つの入力に接続することも許されない．そして，どこにも接続されていない入力線と出力線のいくつかが回路全体の入力と出力として指定される．

ここでは記憶つき可逆論理素子，特にロータリー素子を用いて回路を構成することを考える．従来の（非可逆な）論理回路の設計理論では，記憶を持つ論理回路は，論理ゲートで構成される組合せ論理回路の部分と，純粋の記憶素子からなる部分とに分離して設計がなされ，特に前者の簡単化に力が注がれていた．可逆論理回路の設計でも，これまではこのような従来法が踏襲されることが多かった．実際，記憶つき可逆論理素子自体も可逆論理ゲートと記憶素子から構成できるので，こんな素子を考えても意味がないのではないかと思いがちである．もちろんそれを現在の電子回路技術あるいはその延長線上の方法で実現しようとするならば，そう言えるかもしれない．しかしながら，可逆論理素子に関する新しいアイディアや新しい実現法を得ようとしたときには，従来の考え方は必ずしもよい結果をもたらさない．記憶つき可逆論理素子が有用と考えられる第１の理由は，組合せ論理回路と記憶素子に分離して設計するような従来法にはない新しい考え方の可能性を提示してくれることである．適切な素子を仮定すると，可逆順序機械や可逆チューリング機械の構成法がとても考えやすくなり，簡潔に構成できる．第2の理由は，BBM のところで述べたように，その素子自体を組合せ論理回路と記憶素子に分けて物理的に実現するのではなく，可逆的な物理現象で直接実現する方法の可能性が示唆されることである．

ここで，任意の可逆順序機械がロータリー素子だけからなる可逆論理回路として構成できることを説明しよう．例として，3 つの状態を持ち，入力記号数と出力記号数が共に 2 である可逆順序機械 M_1 で，**表 -2** に定められるものを考える．M_1 をロータリー素子によって実現すると**図 -8** のようになる．この回路は次のようにして構成できる．M_1 の 3 つの状態 q_1, q_2, q_3 の各々に，3 つのロータリー素子からなる「列」を対応させて配置し，これら 3 行 3 列からなる合計 9 個の素子を図のように接続する．このとき，もし M_1 が状態 q_j にあるならば，第 j 列の一番下のロータリー素子の状態を H に設定し，それ以外はすべて状態 V にする．図 -8 は M_1 が状態 q_1 にあることを表している．また，9 個の素子のうちで第 1 行目にあるものは入力記号 a_1 と出力記号 b_1 に，第 2 行目にあるものは入力記号 a_2 と出力記号 b_2 に，それぞれ対応付けられる．図 -8 で，たとえば入力ポート a_1 に粒子が与えられたとする．そうすると第 3 行第 1 列の素子は状態 V に変化し，粒子が“q_1 a_1”と書かれた線上に現れることが確認できる．一般に，状態が q_j で入力が a_i の場合，それらの交点，つまり第 i 行第 j 列の素子から出ている“q_j a_i”と

いう線上に粒子が現れる．表 -2 を見ると，現在の状態が q_1 で入力が a_1 なら，次の状態は q_2 で出力は b_1 となっているので，“q_1 a_1” と書かれた線は q_2 の列と b_1 の行の交点にある素子に接続される．これにより，第 3 行第 2 列の素子が状態 H に設定され，粒子は最終的に出力ポート b_1 から出てゆく．このような方法でロータリー素子からなる回路によって可逆順序機械がシミュレートできる．

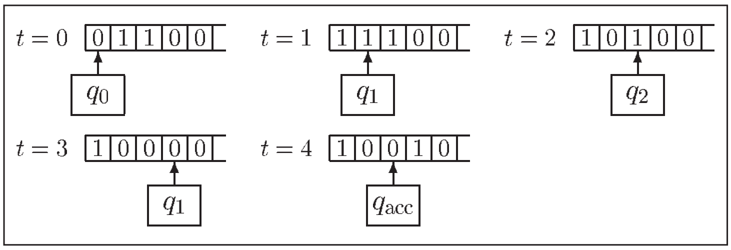

図 -9　可逆チューリング機械 T_{parity} に入力記号列 “11” を与えたときの計算過程

可逆論理素子で可逆チューリング機械を作る

チューリング機械 (Turing machine) は，計算機の理論モデルの中で最も古典的なものである．これは，ます目に区切られた無限長のテープと，ます目上の記号を読み書きするためのヘッドを持つ有限制御部から構成される．可逆チューリング機械というのは，それのどの計算状況（機械全体の状態）も直前の時刻の計算状況を高々 1 つしか持たない，言い換えれば時間の逆方向に決定的なチューリング機械のことである（厳密な定義は文献 1)，4)，6) 参照)．そして Bennett[1] は，どんな非可逆なチューリング機械でも可逆チューリング機械によってシミュレートできること，したがって可逆性の制約を課しても計算能力が下がらないことを示した．チューリング機械の動作は $[q, s, s', d, q']$ の形式の「5 項組」の集合によって規定される．この 5 項組は，有限制御部の状態が q でテープ記号 s を読んだなら，記号を s' に書き換え，$d \in \{L, R\}$（ただし L と R は左と右）の方向にヘッドを動かし，状態を q' に遷移させる，ということを表す．ここで可逆チューリング機械の例 T_{parity} を考える．これの動作は次の 5 項組集合によって定められる：$\{[\,q_0, 0, 1, R, q_1\,], [\,q_1, 0, 1, L, q_{\mathrm{acc}}\,], [\,q_1, 1, 0, R, q_2\,], [\,q_2, 0, 1, \mathrm{L}, q_{\mathrm{rej}}\,], [\,q_2, 1, 0, R, q_1\,]\}$．$T_{\mathrm{parity}}$ は，記号 1 が n 個並んだ列つまり非負整数 n の単進表現が与えられたときに n の奇偶を調べ，偶数のときには受理状態 q_{acc} で，奇数のときには拒否状態 q_{rej} で停止する．ただし，記号列を左から右方向に読んでゆきながら，記号 1 を 0 に，0 を 1 に書き換える．**図 -9** は，記号列 “11” が与えられたときの計算過程である．

可逆チューリング機械 T_{parity} をロータリー素子からなる回路で実現する方法を詳しく説明すると長くなるので，ここには述べないが，最終的に得られる回路は**図 -10** のようなものであり，かなり単純に構成できることが見てとれる．可逆チューリング機械の有限制御部とテープの各ます目は可逆順序機械として定式化できるので，基本的には前述の方法でそれを実現するロータリー素子の回路を構成できるのだが，図 -10 では回路をより簡単にする工夫がなされている[6]．図中の Begin のポートに粒子を入力すると計算を開始し，計算が終わると Accept または Reject から粒子が出力される．この回路による計算の全過程は文献 5) で見ることができる．

さて，図 -10 の回路中の各ロータリー素子を図 -7 の方法で BBM 中に実現したとすると，反射板は多数必要になるが，各素子ごとに静止したボールを 1 個，適切な位置に置くだけでよい．計算を開始させるには，図 -10 の Begin ポートに対応する位置から，運動するボールを任意の速度で任意のタイミングで入れてやればよい．計算が終了すると，Accept または Reject のポートから運動するボールが出てくる．もし BBM の空間が理想的なものであるならば，つまりボール同士やボールと反射板の衝突が弾性的であり，しかも摩擦がなければ，計算過程においてエネルギーが消費・散逸することはない．

可逆コンピューティングの展望

可逆コンピューティングの研究は 1960 年頃[3] から始まったと考えられ，割合長い歴史を持つが，今

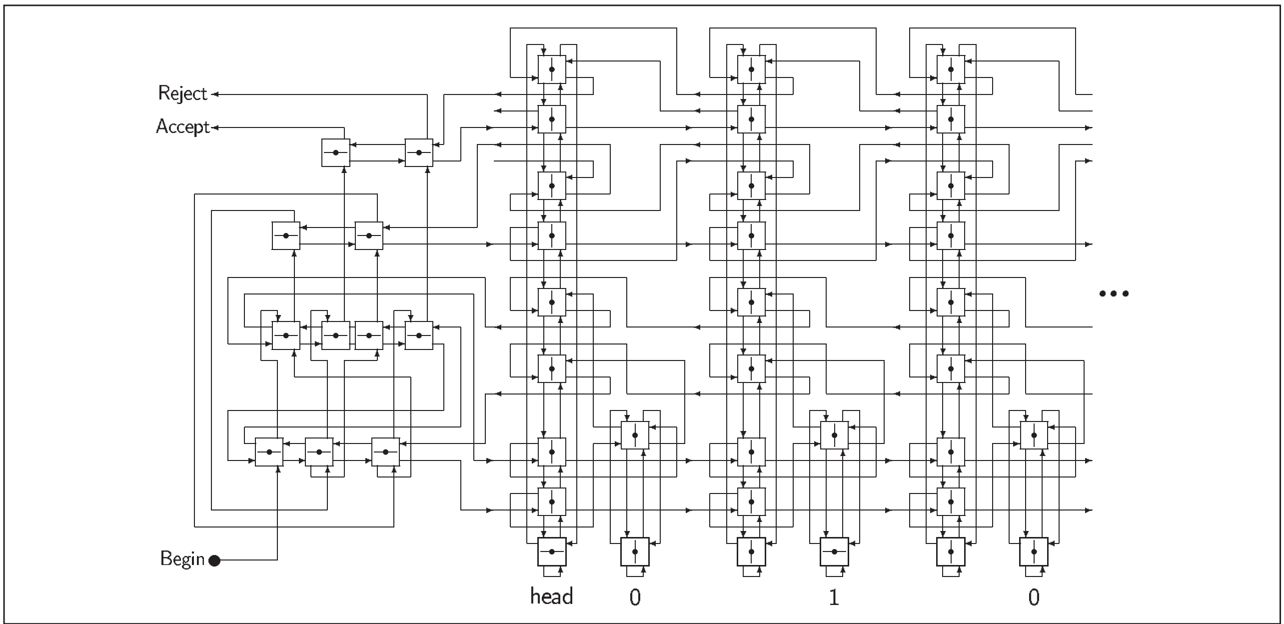

図-10　ロータリー素子により構成された可逆チューリング機械 T_{parity} 5)．この場合の入力記号列は “1” である．

なお発展途上にある．可逆コンピューティングは，それと近い関係にある量子コンピューティングが 1990 年代後半から脚光を浴びたために，その陰に隠れていた感もあるが，特に理論面では地道に研究が続けられていた．2009 年からは Reversible Computation (RC) の国際ワークショップが毎年開催されており，世界各地で活発な研究が行われていることがうかがえる．RC 2012 は 7 月にコペンハーゲンで開催される．

現在の計算機は，AND, OR, NOT といった演算を基礎に置いているが，これらは，遠く古代ギリシャのストア派やメガラ派の時代から知られている概念である．これらは人の思考過程の分析から出てきたもので，人にとって考えやすいという利点はあるが，自然界の法則を直接的に反映しているとは言えない．もちろん，たとえばロータリー素子などの論理的な働きを，AND, OR, NOT など従来型の論理関数を用いて記述することは可能であり，その意味で可逆コンピューティングはそれらとまったく無関係ではあり得ない．しかし，可逆コンピューティングはそれらが考え方の基礎になっているのではなく，物理的可逆性という自然界の法則を反映した演算・操作に基礎を置いているという点でスタンスがまったく異なっている．したがって，従来型の論理演算や計算モデルにとらわれずに新しいアイディアを考え出すことが必要になってくる．実際，この研究分野は，従来型の計算システムとは大きく異なった思考法へと導いてくれるので，将来の計算システムを考えるための新たな視野を開き，知的好奇心を呼び起こしてくれる領域である．

参考文献

1) Bennett, C. H. : Logical Reversibility of Computation, IBM J. Res. Dev., Vol.17, pp.525-532 (1973).
2) Fredkin, E. and Toffoli, T. : Conservative Logic, Int. J. Theoret. Phys., Vol.21, pp.219-253 (1982).
3) Landauer, R. : Irreversibility and Heat Generation in the Computing Process, IBM J. Res. Dev., Vol.5, pp.183-191 (1961).
4) Morita, K. : Reversible Computing and Cellular Automata — A Survey, Theoret. Comput. Sci., Vol.395, No.1, pp.101-131 (2008).
5) Morita, K. : Constructing a Reversible Turing Machine by a Rotary Element, a Reversible Logic Element with Memory, Hiroshima University Institutional Repository, http://ir.lib.hiroshima-u.ac.jp/00029224 (2010).
6) 森田憲一：可逆計算（ナチュラルコンピューティング・シリーズ第 5 巻），近代科学社，東京 (2012).

(2012 年 2 月 1 日受付)

森田憲一（正会員） morita@iec.hiroshima-u.ac.jp
1973 年大阪大学大学院基礎工学研究科修士課程修了．同大基礎工学部助手，山形大学助教授，同教授を経て，1993 年広島大学教授．可逆コンピューティング，セルオートマトン等の研究に従事．工学博士．