

## 論理学の歴史とコンピュータ

高橋正子 (国際基督教大学)\*

現在のコンピュータの原型と見られる万能コンピュータが 20 世紀半ばに誕生してから半世紀余りが過ぎた。その間に、機械の性能や、利用形態、応用分野の広がりなどの面で著しい進展が見られる一方で、コンピュータの基本設計そのものには大きな変化はないという。本稿では、コンピュータより半世紀余り前に芽を出し、やがて急速に花開いた論理学<sup>1</sup>が、現代のコンピュータの誕生にどのような影響を与えたかについて、既存の資料<sup>2</sup>をもとに考察を行う。

1 節では、万能コンピュータが誕生するまでの 1 世紀余りの間に自動計算機械作成のためにどのような試みが行われたかを概観する。2 節では、万能コンピュータとは何か、それまでに作られた道具や機械と比べて万能コンピュータは根本的にどこが違うのかなどの問題を現代の視点で整理する。次の 3 節では、万能コンピュータに対する Babbage と Turing の試みを比較し、二人がそれぞれ生きた 19 世紀と 20 世紀の間に何が基本的に変化したのかについて論じる。続く 4 節では、von Neumann が Turing の成果を現実の (万能) コンピュータの誕生に繋ぐ上で実際にどんな役割を演じたのかについて考察する。最後の 5 節では、コンピュータの歴史に対する今後の期待などを述べる。

### 1. コンピュータ小史 (Babbage から万能コンピュータの誕生まで)

この節では、次節以降の準備として、万能コンピュータの誕生に関連する主な出来事を簡単な注と共に列挙する。

- 1.1 ケンブリッジ大学の数学の教授だった C. Babbage (1791 - 1871) は 1822 年に英国政府の予算を得て、数表作成の自動化を目指す歯車式計算機 difference engine の開発を始めた。しかし、試作機の作成を終え実用機の完成を目指して

\*Masako Takahashi, Department of Information Science, International Christian University, Tokyo 181-8585 Japan. E-mail: mth@icu.ac.jp

<sup>1</sup>記号論理学, 数理論理学, 数学基礎論などの名でよばれる数学的な論理学をここでは簡単のため単に論理学とよぶ

<sup>2</sup>コンピュータの歴史とその主な登場人物に関する一般的な事項については [1, 5, 7, 10, 11, 13, 15, 17, 21] を本文中のあちこちで参考にした。それ以外の事項についてはその都度引用文献を示す。

いた 1834 年頃、より高性能の新しい計算機の着想を得たためこの計画を断念した。Analytical engine とよばれるその新しい機械は、計画によると「あらゆる種類の数値計算」を行うためのもので、ある種の「プログラム」を穿孔カードから読みながら、そこに指定された演算を、歯車で表現されたデータに対して次々に適用していく手動の計算機である。そして、そのプログラムの中に条件分岐や繰り返し計算の機能も含み得るとされた [16]。しかし、彼の多年の努力にも拘らず、この構想も実現には至らなかった。

1.2 それから約 1 世紀後の 1944 年、ハーバード大学と IBM が共同でリレー式計算機 Harvard-IBM Mark I を完成した。これは Babbage の analytical engine の構想の縮小版を 20 世紀前半の電磁気のテクノロジーを用いて実現したもので、(条件分岐を含まない) 基本演算の列を紙テープから読み込み、そこに示された演算をメモリー中のデータに順次適用する自動計算機械である。なお、これ以前に作られたリレー式計算機として、2 進法を採用したベルリン工科大学の Z1 (1938 年) や Z3 (1941 年) がある。

1.3 話が前後するが、1936 年に英国の若い数学者 A. Turing (1912-1954) は以下の内容の論文 [25] を発表した。

- (1) コンピュータの概念を今日「チューリング機械」とよばれる形で数学的に定式化した上で、
- (2) 次に述べる意味で万能なチューリング機械が存在することを示した：任意のチューリング機械  $M$  の記述とそのための入力データを万能チューリング機械  $U$  に与えると、 $U$  は、それらを必要に応じて参照したり書き換えたりしながら、本来  $M$  が行う筈の仕事を  $M$  に代わって行う。
- (3) 更に、この万能チューリング機械を使って、その当時 D.Hilbert が提起していた論理学上の決定問題を否定的に解いた。

ここで (1) の「コンピュータ」として現在我々の身の回りにあるコンピュータを考えてもよいが、1930 年代にこの言葉は計算する人を意味した。要するに人間や機械が行う「計算」とよぶに相応しい一連の行為に共通でしかも本質的な部分を体系化して作られたのがチューリング機械である。実際、彼の定式化は、計算ということばで我々が理解する内容を的確に表現したものであるとして、現実のコンピュータが登場する前もそれ以後も、この種の問題に関心をもつ人々の間で広く支持されている。

(2) の「チューリング機械の記述」は計算の内容を記号列で表記したものであり、今日のことばで言うとプログラムに相当する。(1) と (2) をこのことばを使って言い替えると次のようになる：

計算の手順は記号列の形で（プログラムとして）正確に記述することができ、そのようなプログラムをデータと共に記憶し必要に応じて随時参照したり書き換えたりできるという状況のもとで、そのプログラムに書かれたとおりのことを実行する万能コンピュータがある。

彼は、このように一台ですべてのアルゴリズム（計算手順）を実行することのできる万能チューリング機械の存在を単に証明しただけでなく、その具体的な記述を論文の中で与えている。

- 1.4 第二次世界大戦中に、軍事上の特殊目的のための電子計算機がいくつか開発された。その代表的なものとして、ペンシルバニア大学と米国陸軍が主に弾道計算用に開発し、1946年に完成した ENIAC (Electronic Numerical Integrator and Calculator) がある。

この機械はその名が示すとおり、アナログ計算機（積分機）の仕組みを膨大な数の真空管を使ってデジタル計算機の形で再現した 10 進法による自動計算機である。そのため、プログラムの一部はハードウェアに組み込まれ、それ以外の（データによって変化する）部分は配電盤上の結線やスイッチで指示する方式が採用された。この機械の演算速度は電子化によりそれまでのリレー式機械に比べて千倍程度向上したが、デジタル計算機としては無駄の多い構造であること、プログラムの組み替えに手作業を要すること、プログラムの再利用が困難なことなどの欠点があった。

ENIAC の他に、戦時中に軍事目的のために作られ実際に活躍した電子式計算機として、英国がドイツの暗号解読のために開発した COLOSSUS (1943 年) および Mark-II COLOSSUS (1944 年) がある。また、アイオワ州立大学で、連立一次方程式を解くための 2 進法による電子式計算機 ABC (Atanasoff-Berry Computer) の開発が 1942 年まで進められたが、戦争のため未完に終わった。

- 1.5 ENIAC の開発グループは設計が終わると、機械の製作と平行してこの機械の問題点について議論を重ね、次の目標として、新しく開発した大容量の（遅延線による）メモリに入力データと共にプログラムも記憶して計算を行う汎用電子計算機 EDVAC (Electronic Discrete Variable Computer) の構想をまとめつつあった。

その過程で、この開発グループに 1944 年から加わっていたプリンストン高等研究所の J. von Neumann (1903-1957) が、1945 年にそれまでの議論の結果を彼の視点で「EDVAC レポート（初稿）」[2] としてまとめた。彼の単独名で書かれたその未完の草稿は、電子計算機の開発に関心をもつ人々の間に「プログラム内蔵方式」とか「ノイマン型」コンピュータの名で広く知られるようになり、その後のコンピュータを方向付ける結果となった。

1.6 ENIACが完成した1946年にこのグループは意見の食い違いのため解散したが、ケンブリッジ大学のグループが1949年にEDVACレポートに基づく汎用電子計算機EDSAC (Electronic Delay Stored Automatic Calculator) を完成した。

一方、戦時中に暗号解読やレーダー技術の開発を行っていた英国の研究者達のグループが、戦後マンチェスター大学で新しく(陰極管による)大容量のメモリを開発し、EDSACとは独立に、「プログラム内臓方式」(2.4参照)による汎用電子計算機の実験機 Manchester Baby Mark-I と実用機 Manchester Mark-I をそれぞれ1948年と1949年に完成した。

これら3台の機械は次節に述べるように、いずれも万能コンピュータとよぶに相応しい機械である。

## 2. 万能コンピュータとそうでないもの

この節では、前節(1.3, 1.6)に登場した万能コンピュータについて、そもそも万能コンピュータとは何か、それまでに作られた道具や機械と比べて万能コンピュータは根本的にどこが違うのか、現代のコンピュータは果たして万能か、などの問題について整理する。

2.1 簡単にいうと、万能コンピュータとは「万能チューリング機械と同等の計算能力をもつ機械またはその仕組み」をさす<sup>3</sup>。ただし、この説明は万能チューリング機械の定義を述べなければ無意味だが、それは適当な教科書[12]に任せることにして、代わりにそれが行う仕事をアルゴリズム(計算手順)の形で述べると次の(1)~(3)のようになる。

- (1) [入力] 任意のプログラム  $P$  とそれに対する入力データをメモリに記憶する。
- (2) [初期設定]  $P$  の実行に必要なデータ領域の初期設定を行う。特に、プログラム  $P$  で最初に実行する命令の位置をプログラム・カウンタ<sup>4</sup>にセットする。

<sup>3</sup>Turing が論文を書いた当時は、彼が構成し我々が現在「万能チューリング機械」とよんでいるものが、上で述べた意味の万能コンピュータとして知られる唯一のものだった。そして彼はそれを「万能コンピュータ」とよんだ。今日ではそれと同等の能力をもつものが沢山知られているが、我々は M. Davis [7] に従い、それらをすべて万能コンピュータとよぶ。なお蛇足ながら、万能コンピュータは宇宙の森羅万象を扱うコンピュータではなく、計算できることのすべてを一台でやり遂げるに過ぎないことに注意して欲しい。(実際、計算では解けない問題があることを Turing の論文 [25] は具体的に示している。)

<sup>4</sup>プログラムの実行中に、次に実行すべき命令(の位置、またはアドレス)を記憶するための変数をこのようによぶ。

- (3) [命令の実行とプログラム・カウンタの値の更新] プログラム・カウンタの示す命令  $S$  が終了命令ならこのプログラムの処理を終る。そうでなければ  $S$  を実行し、次いで、プログラム  $P$  の中で  $S$  の次に実行すべき命令の位置をプログラム・カウンタにセットして (3) に戻る。

2.2 ここで肝心なのは、プログラム  $P$  としてあらゆるアルゴリズム（をこのコンピュータ用の書式で記述したもの）を万能コンピュータは処理できることで、その性質をもたないものは仮令上の (1)~(3) の流れに従って作業を行うものでも万能コンピュータではない。ところで、そうするとプログラム  $P$  として想定されるものは無限にあるから、従って  $P$  の長さはどれほどにも長くなり得る。そのため万能コンピュータのメモリ（記憶装置）は有限ではあり得ない。しかし、現実のコンピュータが無限に多くのメモリをもつことはないから、万能コンピュータの実現という場合には、「必要に応じてメモリをどれだけでも拡張できるという理想化された状況のもとで万能チューリング機械と同等の働きをするコンピュータ」<sup>5</sup> を万能コンピュータという。

2.3 上で述べた万能コンピュータの記述は Turing の論文に記されているものと比べると見かけ上大きく異なる。その理由は、チューリング機械の基本設計や命令体系が今日のコンピュータのそれとは異なるためである。特に、チューリング機械のメモリ（彼はこれをテープとよんだ）にはアドレスがなく、従ってプログラム・カウンタもない。そのため、チューリング機械の場合は、はじめに読み込んだプログラムやデータを途中で参照するだけでなく何度も書き換えることによって上の (1)~(3) に相当する作業を行っている。

2.4 万能コンピュータの「万能性」とは、すべてのアルゴリズムが実行できることだと述べたが、その性質を保証するのが、プログラムと入力データをまずメモリに読み込み実行時に必要な箇所を随時参照するという意味での「プログラム内蔵方式」である<sup>6</sup>。その際、万能チューリング機械の場合のように、記憶したプログラム（やその他のデータ）を実行時に書き換えてもよいが、そのことは本質的ではない。実際、ある種の基本的な機能が備わっているコンピュータであれば、実行時にプログラムを書き換える必要はない。具体的に言うと、次の三つの機能をもつコンピュータの場合、プログラムを実行時に書き換えることなしに万能コンピュータのアルゴリズムを実装できることが知られている<sup>7</sup>：

<sup>5</sup>このコンピュータはあくまでも有限の現実的な機械であり、無限に大きい仮想機械を考えている訳ではない。一方、チューリング機械は無限のメモリをもつ仮想機械である。

<sup>6</sup>プログラム内蔵方式 (stored-program) ということばは二つの異なる意味で使われることがあるので注意を要する。その一つはここで述べた意味で、もう一つは単にプログラムを実行前にメモリに読みこむという意味である。本稿では、混乱を避けるため、ここで述べた意味のときはカッコをつけて「プログラム内蔵方式」と書くことにする。

<sup>7</sup>これらの機能をもつコンピュータを Random Access Machine（略して RAM）という。RAM の概念は、1950年代末に複数の研究者により独立に見出され、1961年にそれが万能コンピュータの条件を満たすことが文献 [14, 18] に発表された。詳しくは、例えば [3, 23] を参照せよ。

(1) メモリにアドレスの機能がある。(2) メモリの指定された位置に整数 0 を書き込んだり 1 を足したりする機能がある。(3) メモリの指定された位置に記憶されている値に従って条件分岐を行う機能がある。

ところで、1.6 で述べた 1940 年代末に完成した 3 台のコンピュータは、いずれも上の機能をもっているから、その 3 台とも万能コンピュータである。

2.5 一方、プログラムをメモリに読み込まずに、端から順に 1 文字ずつ読みながら処理していくコンピュータを考えると、上で述べた意味での「プログラム内臓方式」に比べて計算能力に決定的な違いがある。実際、データを読みながら（途中で後戻りしたりせずに）その順に処理を行うコンピュータの数学的なモデルは「有限オートマトン」とよばれ、その計算能力はチューリング機械に比べて著しく小さいことが知られている<sup>8</sup>。

2.6 現代のコンピュータは、半世紀前にその原型が誕生して以来、例えばインターネットやマルチメディアなどの新しい応用分野を次々に開拓しつつあるが、こうしたコンピュータの多様な活躍ぶりは、「あらゆるアルゴリズムを実行できる」という万能コンピュータの万能性の現れと見ることができよう。そう考えると、コンピュータの応用分野が新しく開拓されると、その都度新しい入出力機器やアルゴリズムを開発する必要はあっても、コンピュータの基本設計を変える必要がないのは当然だと言える。この点が、特定の目的のために作られる従来の道具や機械と比べて現代の (万能) コンピュータが決定的に異なる点である。

### 3. Babbage の時代に不足していたもの

3.1 Babbage は analytical engine (1.1 参照) を開発しながら、その構想についてヨーロッパ各地で講演を行った。その中の一つであるイタリアでの講演をまとめた文献 [16] が analytical engine に関する比較的詳しい解説とされるが、その中にこの機械で何ができるかについて次のような説明がある：すべての数値計算は四則演算に還元することができ、四則演算はこの機械の基本演算として装備されるから、この機械であらゆる種類の数値計算を行うことができる。Analytical engine に関する記述にはこうした大胆な（しかし必ずしも意味が明確でない）文章と、具体的な数値計算の例について計算のステップを詳しく解説したものが多く見受けられるが、残念ながら機械の全体の輪郭が掴み難い。その一つの理由は、この機械ではプログラムをどのように書くのかについての記述が見当たらないことである。（例えば [13] にもそのことが指摘されている。）

<sup>8</sup>例えば [12] を見よ。

3.2 ところで、ある分野の計算（またはその他の人間の行動）の手順を機械に伝えて代行させるとなると、ふだん人間がやっていることを曖昧さのない形で詳細に書き下し、それを機械が認識できるような形に記号化して与える必要がある。そしてそのためには、機械化しようとする事柄について人間が行っていることのすべてを正確に表現することのできる一種の人工言語をまず設計する必要がある。ただし、もし機械に実行させようとする仕事が（例えば自動販売機のように）ごく限られた種類のものであれば、予め機械に必要な手順を組み込んでおき、外側に押しボタン等を用意すれば済む<sup>9</sup>。しかし、もっと多様な種類の計算を機械に代行させようとする場合はそのような方法では済まない。実際、Babbage は analytical engine で初めて計算手順の指定が実行時に必要なことに気が付き、そのために「オペレータ・カード」や「オペランド・カード」とよばれる穿孔カードを使ってそれらを読み込むアイデアをジャカル織機<sup>10</sup> から得たという。しかし、上で指摘したように、それらのカードでどのように（条件判定を含む）計算手順を指定するのかを彼が明確にしなかったということは、彼がその問題の重要性を認識していなかったか、あるいは納得のいく結論が得られないままに終わったことを示唆するように思われる。

3.3 しかし考えてみると、Babbage に限らず当時の人々に、「ふだん人間が半ば無意識に行っていることを、完全に曖昧さのない形で書き下し、それを機械が認識できるように記号化して表現する」ことを期待することは果たして妥当だろうか。今でこそ、コンピュータは人間のように融通が利かないことや、従ってコンピュータに何か仕事をさせるにはプログラミング言語という一種の人工言語を使って逐一その内容を指示しなければならないことが常識になっているものの、当時、それに類するものはまだ全く姿を見せていなかったのではないだろうか。

3.4 歴史上これに近い考え方が初めて登場したのは、近代科学としての（数学的な）論理学が誕生した 19 世紀後半から 20 世紀はじめにかけてではないかと思われる。というのは、論理学はそもそも数理科学的なものの考え方の原理やその性質を数学的な厳密さをもって研究しようとする学問であるから、数理科学の諸分野でふだん人々が伝統的に何気なく使っているいろいろな表現を例えば「論理式」や「証明」（の形と意味）として定式化し、それらの間に見られる規則性（その多くはやはり直観的に人々が理解しているもの）を明示的かつ網羅的に体系としてまとめる作業が必要だった筈である。しかし、皮肉なことに Babbage が analytical engine の実現のために奮闘していた 19 世紀中頃という時代は、

<sup>9</sup>実際、Babbage が最初に手掛けた difference engine の場合は、計算の内容が、階差法に基づき多項式（で近似される関数）の数表を作成するというごく限られたものだったため、機械の中に組み込まれた計算機構に外から必要なパラメータを入力データと共に与えればよかった。また、後の ENIAC の場合も、プログラムが機械に組み込まれた固定部と後からスイッチなどで指定する可変部から成るという意味で、同様の考え方に基いて限られた種類の数値計算を行う機械と見ることができる。

<sup>10</sup>19 世紀初頭にフランスで発明され当時広く使われていた折込模様入りの織物を織る機械。

(数学的な) 論理学がまさにこれから生まれようとする時代だった<sup>11</sup>.

- 3.5 それに対して, Babbage より約 120 年後に生まれた Turing の時代には状況が一転していた. 彼が万能コンピュータに関する研究を行った経緯は概略次のとおりである.

1934 年 6 月にケンブリッジ大学数学科を卒業した後, 大学に残って研究を続けることを希望した彼は, 翌年 3 月に大学のフェローの資格を得て一人前の研究者としての道を歩きはじめた. またそれと同じ頃 (1935 年春学期), 彼は M. H. A. Newman による「数学基礎論」の講義を聴き, その数年前に発表されたばかりの K. Gödel の不完全性定理とその証明を学んだ. 更に彼はその講義で, 論理体系が満たすべき条件として Hilbert が提起した「完全性」, 「無矛盾性」, 「決定可能性」の三つの条件のうち, はじめの二つは Gödel の結果により (ある厳密な意味で) 達成不可能であることが示されたが最後の条件は未解決のままであることを知った. そして彼は述語論理の体系が決定不能であることの証明を考え始めた.

- 3.6 そのため彼はまず, 「計算」という直観的な概念をチューリング機械という新しい体系を用いて的確に定式化し, その上で, Gödel によるゲーデル数化や G. W. Cantor による対角線論法などの既知の手法と彼自身の新しい証明のアイデアを用いて, 万能チューリング機械の構成と Hilbert の決定問題に対する否定的な解を得ることに 1936 年 4 月に成功した<sup>12</sup>. その際, 彼自身が見出した新しい証明のアイデアのうち特に重要なのは, 2.1 に述べた万能コンピュータのアルゴリズムの考え方で, これは後に (1945 年以降に) 「プログラム内蔵方式」として, また (1950 年代以降に) コンパイラの基本原則として知られる考え方に繋がるものである. 彼は, 現実の自動計算機がまだ一台も作られていなかった 1936 年という早い時期にそのアイデアに到達し, それを使って世界で最初の万能コンピュータである万能チューリング機械を構築した.

- 3.7 Turing の場合にはこのように, (1) 直観的な概念を定式化および記号化するという考え方が論理学の新しい伝統の中で既に始まっていたこと, また, (2) Gödel の不完全性定理のような深い結果がその上に築かれ, 論理学にそれまで

<sup>11</sup> 古代ギリシャに始まる広義の (哲学や弁論の術などを含む) 論理学の歴史を振り返ると, アリストテレスの推論体系 *sylogism* が紀元前 4 世紀に形成され, それがすべての正しい推論を網羅するものとして 2000 年以上に亘って本質的な変化のないまま継承された後, 1850 年前後になってようやく数学的視点に基づく論理学の芽生えが G. Boole, C. S. Peirce, らによって研究され始めた. そして 1879 年に, 本格的な述語論理の体系が G. Frege によりはじめて構築された. ただしその Frege の体系も, 表記法 (つまり, 上で述べた人工言語の形式の部分) の問題点などのために, 20 世紀初頭に B. Russell が有名なパラドックスを指摘するまで誰も注意を向けなかったといわれる [4, 8].

<sup>12</sup> なお, その論文の初稿完成とほぼ同時期に, プリンストン大学の A. Church により  $\lambda$  計算に関する関連研究が発表されたことを Newman から知らされた Turing は, 直ちにチューリング機械と  $\lambda$  計算が同等の能力をもつことの証明を論文に附録として書き加えた. 更に彼は翌 1937 年に, チューリング機械,  $\lambda$  計算, 帰納的関数など, その時点で知られていた計算モデルの間の関係をより詳しく研究した [26]. その結果, 当時の論理学者達の間でこれらを計算の妥当な定式化と見なすことについての実質的な合意が得られるようになった [8, 9].

とは異なる新たな展望が開かれると同時に、そのために必要な新しい証明の手法が開発されつつあったこと、そして (3) 彼自身が是非解明したいと思う魅力的な論理学上の問題が目の前にあったこと、などの好条件があった。

Babbage と Turing の時代の間、こうした論理学の誕生と目覚ましい発展があったことに注目する視点は、現代のコンピュータと論理学の関係を正当に評価する上で重要であると思う。

#### 4. 万能コンピュータの実現に対する von Neumann の貢献

本節では、一般の書物でよく見かける「コンピュータの生みの親は von Neumann である」という説の妥当性について検討する。

4.1 これまでに 2 節で、von Neumann の書いた「EDVAC レポート」の中心的なアイデアである「プログラム内蔵方式」と万能コンピュータの関係について、前者は後者を実現するための方法であることを述べ、3 節では、後者が Turing の論文に由来するものであることを述べた。時間的にみると Turing の論文は EDVAC レポートより 9 年前に発表されたから、万能コンピュータのアイデアの発見 (あるいは発明?) 者は明らかに Turing である。しかし、学問の世界でよくあるように、von Neumann が Turing の論文とは独立に「プログラム内蔵方式」のアイデアを考えたということも可能性としてはあり得る。あるいはそうでないとしても、von Neumann がコンピュータの生みの親とよばれるに相応しい何か別の理由があるかも知れない。もしそうだとすると、それはどんな役割だろうか。

4.2 上の疑問に答えるため、以下にまず関連する事柄を列挙する。それらを見比べると互いに矛盾しているような印象を受けるものもあるが、時間差などを考慮すると必ずしもそうではなく、全体を矛盾なく説明する一貫したストーリーが見えてくるように思われる。

- (1) Von Neumann は、論理学、量子力学、連続幾何学、リー群論、ゲーム理論、コンピュータの開発と応用などの広範な分野で活躍した著名な数学者である。論理学の分野では、Russell のパラドックスを回避する目的で、「集合」より大きい「クラス」の概念を導入した公理的集合論 (NBG 集合論とよばれるものの前身) を構築・研究した博士論文 [24] が有名である。しかし彼は 1930 年、Gödel の不完全性定理によりそれまで Hilbert 学派の仲間達と共に目指していた目標が打ち砕かれたことを知って論理学を離れた。1933 年に彼はプリンストン高等研究所に最年少の教授として着任し、その後次第に応用数学、特にコンピュータを利用したいいわゆるビッグ・

サイエンスの方向に舵を切り換え、研究所内外のグループを率いて精力的に活動した。(Von Neumann の生涯、業績、思想などについては [1, 10, 15, 21] を見よ.)

- (2) Turing は 1936 年、チューリング機械の論文 [25] を書き終えたのち、その年の夏から 2 年間プリンストン大学の博士課程に留学し Church のもとで  $\lambda$  計算、群論、相対的計算可能性などの研究を行った。帰国後は、暗号解読機 (COLOSSUS およびその前身) の論理設計に従事した後、国立物理学研究所やマンチェスター大学のコンピュータ・グループに加わり、コンピュータの開発と応用に参加するかたわら、初期の人工知能論や数理生物学などの研究を行った。(Turing の生涯、業績、思想などについては [7, 11, 17] を見よ.)
- (3) Turing が渡米する前年の 1935 年、von Neumann はケンブリッジ大学を訪れ、概周期関数に関する講義を行った。一方、予て von Neumann の著書「量子力学の数学的基礎」[27] を勉強していた Turing はそこに記されていた著者の概周期関数に関する結果を拡張した論文を数週間前に書き、学会誌に投稿していた [11].
- (4) Turing がプリンストンで 1937 年夏に奨学金を申請した際、彼のために von Neumann が書いた推薦書が残されている。その中で von Neumann は、自分が興味をもった Turing の仕事として、純粋数学の論文 2 篇<sup>13</sup> を挙げているが、既に出版済みだったチューリング機械の論文 [25] には触れていない [7, 11].
- (5) 1938 年春、帰国を数か月後に控えた Turing に von Neumann はプリンストン高等研究所における助手のポストを申し出た。しかし Turing はそれを断わり、予定どおりケンブリッジ大学に戻った [7, 10, 11].
- (6) 1939 年に von Neumann は高等研究所の同僚 S. Ulam に何度も Turing の論文 [25] の素晴らしさについて語った [10, 11].
- (7) Von Neumann は、1930 年代末に弾道計算と関わりをもつようになってからコンピュータに本格的に関心を寄せるようになり、1943 年前半の英国滞在が彼のコンピュータへの興味を更に駆り立てた [1, 10, 21].
- (8) 1943 年頃 von Neumann は、ロスアラモス科学研究所の物理学者 S. Frankel に、コンピュータの基本的なアイデアは他の誰でもなく Turing に負うと強調して語り、Turing の論文を読むことを強く勧めた [11, 20].
- (9) 1944 年 1 月に ENIAC グループのメンバーにより大容量のメモリが開発された。そして、このメモリを使ってプログラムとデータを記憶して計算を行う ENIAC の後継機の計画が話し合われた [13, 17].

<sup>13</sup>その一つは (3) で述べた Turing の最初の論文で、もう一つは彼がプリンストンに来てから von Neumann の薦めで取り組んだ群論に関する論文である。

- (10) Von Neumann は, ENIAC グループの数学系のメンバーである H. H. Goldstine の誘いで 1944 年 9 月から ENIAC グループに加わり, 翌年 6 月に EDVAC レポートを書いた. 彼自身はこのレポートを公開する意志はなかったという [10].
- (11) 1946 年に von Neumann は高等研究所の彼のコンピュータ・プロジェクトのメンバー達に, 計算理論に関する教育の一環として Turing の論文 [25] を読ませた [1, 11].
- (12) 1946 年 von Neumann は N. Wiener に宛てた手紙の中で, 彼の EDVAC レポートの内容は, Turing の万能コンピュータと神経網に対する McCulloch-Pitts モデルという二つの偉大な研究成果を合体させたものであるとのコメントを述べている [11].
- (13) ENIAC グループおよびプリンストン高等研究所のコンピュータ・グループで von Neumann と行動を共にした Goldstine は, von Neumann の死後, 彼のコンピュータに関する業績を中心にコンピュータの歴史を著した [10] が, その中で彼は Turing の論文 [25] に言及する際, チューリング機械とよく似た方法で計算を定式化した E. L. Post の論文<sup>14</sup> [19] と対にして, 常に Post-Turing work とよんでいる.

4.3 これらの証言を読むと, 二人は 1935 年 (当時 von Neumann は 32 歳, Turing は 23 歳) 数学者として知り合い, 少なくとも Turing の渡米 1 年後の 1937 年夏までは二人の間の話は純粋数学に関するものに限定されていたようである. 特に, 当時出版されたばかりのチューリング機械の論文に対して von Neumann は関心を示さなかった. しかし, その後 von Neumann はコンピュータに興味を持つようになると共にチューリング機械の論文を読み, 高く評価するようになったと思われる.

一方, ENIAC グループでは ENIAC の後継機として, 新たに開発された大容量メモリにプログラムとデータを記憶する案が検討されていたところに von Neumann が加わり, 翌年 6 月に彼は「EDVAC レポート」を書いた. その時点で von Neumann は Turing の論文を理解していたが, 実際のコンピュータでそれが実現可能かどうかについては必ずしも確信を持っていなかったかも知れない. 一方, グループの他のメンバーが Turing の論文の特に万能チューリング機械について理解していた可能性は極めて低いと思われる.

4.4 そのような状況の下で von Neumann は,

(a) 万能コンピュータを実現するためにはどのようなアーキテクチャ<sup>15</sup> であ

<sup>14</sup>Post は Turing とは独立に研究を行い, Turing の論文よりやや遅れてこの論文が発表された. ただし, この論文には万能コンピュータに類するものは全く登場しない.

<sup>15</sup>コンピュータを構成する個々のハードウェアについて語るのではなく, それらが全体の中でどのような論理的役割を担っているかに注目してコンピュータの構造を述べたもの.

ればよいかを考えてみることに彼自身興味をもち、また

- (b) もしそれが上手くいけば、グループのメンバーにそのアーキテクチャを提案して賛同を得たいと考え、

グループ内の討議資料として彼は EDVAC レポートを書いたのではないだろうか。しかし、予想に反して彼のレポートはグループのメンバーの反感を買い、ENIAC グループは解散した。そして、更に予想外の展開として、彼がグループのメンバーに伝えなかったことは、このグループの代わりに世界のコンピュータ関係者に彼の名と共に伝えられる結果となった<sup>16</sup>。

- 4.5 上のように考えると、彼が何故個人名でこのレポートを書いたか理解できるような気がする。つまり、彼はこのレポートをグループを代表して外に発表するために書いたのではなく、万能コンピュータが現実に可能かどうか彼自身で確かめるために書き、その結果が良好と思われた段階でそれをグループのメンバーに自分からの提案として渡したのだとすると、彼が(提案者として)自分の名前を書くことは至極当然である。また、このレポートにはグループでの話し合いで出された様々なアイデアが誰のものか説明なしに書かれているという非難があったというが、それもこのレポートの意図を上のように考えれば不自然なことではない。

なお、そのことに関連して、実はこのレポートには Turing の名前や万能コンピュータについての言及もないという事実がある。そのこと自体は上と同じ理由で説明がつくと思うが、しかし、そのことのために「コンピュータは大天才 von Neumann がごく短期間で発明した」というような子供じみた俗説が広まったとすると、それは将来を担う(大天才でない)子供達にとって不幸なことだと思う。

- 4.6 Von Neumann が現代のコンピュータにどんな貢献をしたかについては、様々なことが語られている<sup>17</sup>が、その中で、物理学者 Frankel による次の証言 [20] が私には最も的を得たものに思われる。

多くの人が von Neumann をコンピュータの父だと主張するが、彼自身は決してそうは思っていなかった。むしろ彼は助産婦とよぶ方が相応しい。彼はよく私や他の人々に向かって、基本的な考え方は Turing に負っているのだと強く言っていた。私の考えでは、von Neumann が行った本質的なことは、Turing によって明らかにされた基本的な考え方と ENIAC グループやその他の人々が行ったコンピュータの開発の仕事の世界中に知らせることだったと思う。

<sup>16</sup>当時のコンピュータ関係者の間でこのレポートが如何に好感をもって迎えられたかを示す証言として例えば [22] がある。

<sup>17</sup>例えば [13] にそれらがまとめて紹介されている。

この証言は、大事なポイントを非常に端的に述べているのに対して、本稿は（決してそのように意図した訳ではないが）この証言に私の推測に基づく肉付きを与えたものと見なすことができるかも知れない。

- 4.7 最後に一言、上のことばに付け加えて、ここでの助産婦役は決して誰にでもできる簡単な仕事ではなかったことを指摘したい。というのは、Turing の論文の本質を正しく読み解き、その背後にあるアルゴリズムを現実のコンピュータ上で実現するには、どういう基本設計や命令体系があればよいかを（そもそも基本設計という考え方がなかった時点でそれを導入して）考えるという仕事は、当時の電子計算機の実情に十分詳しく、しかも論理学に造詣の深い研究者にして初めて可能な仕事だと思われるからである。

## 5. おわりに

はじめに述べたように本稿のねらいは、現代のコンピュータの原型と見られる万能コンピュータが1940年代末に誕生するに当たって（数学的な）論理学がどのような役割を果たしたかについて、既存の資料に基づいて考察することだった。そのため前半の1,2節で準備を行い、後半の3節と4節で試論を展開した。

- 5.1 まず3節では、Turing が万能コンピュータの概念に到達するに当たって論理学からの寄与がどのようなものであったかについて考察し、Babbage の時代との違いを指摘した。なおその際、機械で実行する事柄を詳細に記述するための人工言語の問題に触れたが、その点を重要視し過ぎていると感じられた向きもあろうかと思う。実際、私がこの問題を考え始めた時点でこのことは全く念頭になかった。しかし、Babbage がプログラムの書き方を説明していないことに気づき、不審に思って色々考えながら、ふと（数学的な）論理学の場合に共通な問題があることに気付いて調べるうちに、19世紀半ばに「数学における論理」に注意が向けられ始めてから、Boole, Peirce, Frege, Peano らの仕事を経て Whitehead-Russell の “Principia Mathematica” に至る実に半世紀という歳月をかけて、ようやく論理学のための（人工）言語の土台が一応固まったという意外な事実<sup>18</sup>に遭遇し、この問題の容易ならざること気付かされた次第である。論理学の場合もコンピュータの場合も、一旦そのための人工言語に慣れてしまえば、表現の手段としても思考の道具としても何ら問題なく、あたかも水や空気のように自然に感じられるが、その必要性に気づき、無からそれを実際に作り上げる仕事は、決して過小評価すべきではないだろうと思う。

<sup>18</sup>例えば [4] を見よ。

- 5.2 ところで近年、大学のあり方が様々に問われ、その中で教育や研究においてもとかく即戦力が問題にされる傾向が強まっているが、この3節の内容は、その時流に逆らい、即戦力とは対極にある息の長い理論的な学問のもつ底力について改めて私自身考える良い機会となった。
- 5.3 次いで4節では、Turingの結果を現実の万能コンピュータに繋ぐ重要な橋渡しの役を von Neumann がどのような形で果たしたかについて考察した。これまで私自身、von Neumann が ENIAC グループの議論をまとめるに当たって、何故グループの人々に断りなく彼一人の名前でそれを書いたのか、また、このレポートで Turing について一言も触れていないのは何故かという疑問を感じていた。しかし、今回4節のテーマについて考えているうちに、Turing の仕事を von Neumann が現実的なコンピュータのアーキテクチャというレベルで表現したからこそ、それが多くの人々に伝わり、結果として、不自然なものではなく望ましい構造をもった万能コンピュータが世の中に広まったことを好運に感じた。なおこの件に関連して、英国のマンチェスター大学と国立物理学研究所のコンピュータ開発計画に Turing と彼の論理学上の師である Newman が関わった由であるが、彼らの果たした役割についても時間があれば調べてみたい。
- 5.4 私はこれまで理系の立場からコンピュータ・サイエンスの基礎理論とその関連分野の研究・教育を行ってきたが、最近、教育上の必要からコンピュータの歴史を調べ始めてみて、まだ謎につつまれた部分が多く、自分であれこれ想像を巡らす余地が沢山あり面白い分野だと思うようになった。その面白さの一つは、近年コンピュータを巡る状況が次第に総合科学的な様相を呈しつつあるが、コンピュータの歴史も従来より幅広い視点から見直すことに意味がありそうだと気付いたことが上げられる。その幅広い視点の一つとして、ここに理学的な視点からコンピュータの歴史を考える試みを行った次第である。ただし、私自身は歴史にもコンピュータ技術(特にハードウェア)にも昏く、従って本稿にも多くの間違いや無理解があるのではないかと懸念する。読者諸賢の忌憚のないご批判を頂ければ幸いである。

## 参考文献

- [1] W. Aspray: *John von Neumann and the Origins of Modern Computing*, MIT Press, 1990.
- [2] W. Aspray and A. Burks (eds.): *Papers of John von Neumann on Computing and Computer Science*, MIT Press, pp.17-82 (1987)

- [3] J. Bell and M. Machover: *A Course in Mathematical Logic*, North-Holland, 1977.
- [4] N. Bourbaki: *Éléments d'Histoire des Mathématiques*, Hermann, 1969. 村田全他訳「ブルバキ 数学史」東京図書, 1970.
- [5] M. Campbell-Kelly and W. Aspray: *Computer: A History of the Information Machine*, BasicBooks, 1996.
- [6] M. Davis (ed.): *The Undecidable – Basic Papers on Undecidable Propositions, Unsolvable Problems and Computable Functions*, Raven Press, 1965.
- [7] M. Davis: *The Universal Computer – The Road from Leibniz to Turing*, W. W. Norton & Company, 2000.
- [8] J. W. Dawson: *Logical Dilemmas*, A. K. Peters, 1997.
- [9] K. Gödel: On undecidable propositions of formal mathematical systems, Lecture notes by S. C. Kleene and J. B. Rosser, Inst. for Advanced Study, Princeton. (Reprinted with corrections in [6] pp.39-74)
- [10] H. H. Goldstine: *The Computer – From Pascal to von Neumann*, Princeton University Press, 1972.
- [11] A. Hodges: *Alan Turing: The Enigma*, Walker & Company, 2000.
- [12] J. E. Hopcroft, R. Motwani and J. D. Ullman: *Introduction to Automata Theory, Languages, and Computation*, Second Edition, Addison Wesley, 2001.
- [13] 星野力 「誰がどうやってコンピュータを創ったのか?」 共立出版, 1995.
- [14] J. Lambek: How to program an infinite abacus, *Can. Math. Bull.*, Vol.4, pp.295-302 (1961)
- [15] N. Macrae: *John von Neumann*, Pantheon Books, 1992. 渡辺正 他訳「フォン・ノイマンの生涯」朝日選書 610, 1998.
- [16] L. F. Menabrea and A. A. Lovelace: Sketch of the analytical engine invented by Charles Babbage, <http://www.fourmilab.ch/babbage/sketch.html>
- [17] N. Metropolis et al. (ed.): *A History of Computing in the Twentieth Century*, Academic Press, 1980.
- [18] M. L. Minsky: Recursive unsolvability of Post's problem of "tag" and other topics in the theory of Turing machines, *Ann. Math. Bull.*, Vol.34, pp.437-455 (1961)

- [19] E. L. Post: Finite combinatory processes – formulation 1, *J. Symbolic Logic*, Vol.1, pp.103-105 (1936) (Reprinted in [6] pp.289-291)
- [20] B. Randell: The COLOSSUS, in [17], pp. 47-92 (1980)
- [21] 佐々木力「二十世紀数学思想」みすず書房, 2001.
- [22] 高橋秀俊「電子計算機の誕生」中公新書 273, 1972.
- [23] 高橋正子「計算論 – 計算可能性とラムダ計算」近代科学社, 1991.
- [24] A. H. Taub (ed.): *John von Neumann: Collected Works*, Macmillan, Vol. 1, pp. 339-422 (1960)
- [25] A. Turing: On computable numbers with an application to the Entscheidungsproblem, *Proceedings of the London Mathematical Society*, ser.2, vol. 42 (1936), pp.230-267. Correction: vol.43 (1937), pp.544-546. (Reprinted in [6] pp.116-154)
- [26] A. Turing: Computability and  $\lambda$ -definability *J. Symbolic Logic*, Vol.2, pp.153-163 (1937)
- [27] J. von Neumann: *Mathematische Grundlagen der Quantenmechanik*, Springer, 1932.