

数学レスキュー

—いかにして数学的才能なしに証明を書くか—

新井 紀子

1 証明とはなんだろう

一般の科学では、「...になる」と「...である」は地続きでつながっています。例えば、ばねの伸びと荷重が正比例するというフックの法則は経験則です。しばしば「フックの法則に従う」と表現されるように「そのようになっている」に過ぎません。百回、千回、一万回と実験を重ねても常に「そのようになっている」のであれば、「そうである」と看做して、科学は前に進みます。しかし、数学ではこの二つを峻別します。証明という特殊な手続きを経た命題だけを定理と呼び、そのとき初めて「...である」という助動詞を与えるのです。

証明という表現様式は今から 2500 年ほど前の古代ギリシャで誕生しました。ただ、証明は古代ギリシャから連綿と受け継がれてきたわけではなく、中世で一度実質的には失われた後、近代科学革命の中で再発見されたのです。さらに 20 世紀初頭にヒルベルトら形式主義者は、与えられた公理系からいくつかの推論ルールだけを使って命題を導出するゲームとして数学を再定義しました。このことは当然ながら、「与えられた命題を機械的に証明する」という自動証明の種を胚胎することになるわけです。自動証明のアイデアを要約すると次のようになります。

1. 数学では分野を超えて論理記号が共有されている。また、その論理記号の用法には共通のルールがある。これを推論規則と呼ぶ。
2. 数学の各分野にはそれらを表現するための語彙がある。また、それぞれの語彙の用法がある。
3. 数学の各分野には分野固有の前提がある。2 と 3 を併せて公理とよぶ。
4. 日本語等の自然言語で表現された命題は、1 と 2 のルールに則った形式的な命題に機械的に翻訳することが可能である（はず）。
5. 形式的な命題は、その論理構造に従って機械的に分析すれば証明できる（はず）。

たとえば、 $A \wedge B$ という命題を証明するには、 A と B それぞれを証明する以外ありません。つまり、 $A \wedge B$ という論理式の構造が、どのように証明すべきかを指し示してくれるはずだ、というのが自動証明の基本的なアイデアなのです。このアイデアの出発点となったのが、Gentzen による一階述語論理の形式化であるシーケント計算 LK の**カット除去定理**です [1]。一階述語論理の正しい命題（モデルに依らずに常に正しい命題）はその命題の部分式に関する論理計算のみから証明できる——そのことをカット除去定理は保証してくれます。つまり、証明しようと思っている命題とかけ離れた秀逸な補題を思いつかなくても、その論理構造に従って愚直に分解しさえすれば、（少なくとも論理的に正しい命題に関しては）自動的に証明できるということです。これは数学的才能の有無に悩んでいる多くの人にとって朗報でしょう。

2 誰もが書ける証明のコツ

自動的に証明を生成するなどということが本当に可能なものかどうか、実際の問題で試してみることにしましょう。

問題 2.1 $f(x) = x^2$ がいたるところ連続であることを示しなさい。

大学生にとって鬼門であるイプシロン - デルタ論法を用いる問題のひとつです。ここには、 x^2 や $=$ といった記号のほか「いたるところ」や「連続」という語彙が登場します。これらの定義に従って、機械的に翻訳して得られるのが次の論理式です。

$$\forall x \forall \epsilon > 0 \exists \delta > 0 \forall y (|y - x| < \delta \rightarrow |y^2 - x^2| < \epsilon) \quad (1)$$

$\rightarrow$ は「ならば」を表わす論理記号です。 x, y や ϵ, δ は「対象領域」である実数の上を自由に動き回る変数で、 $\forall x P(x)$ は「任意の（実数） x について、命題 $P(x)$ が成り立つ」、 $\exists x Q(x)$ は「 $Q(x)$ を満たすような（実数） x が存在する」と読みます。ですから、論理式 (1) を直訳すると次のようになります。

「任意の実数 x と任意の正の実数 ϵ に対して、次のような条件を満たすような正の実数 δ が存在する。その条件とは、任意の $|y - x| < \delta$ であるような実数 y について、 $|y^2 - x^2| < \epsilon$ となることである。」

大変読みにくく、しかも長い文ですね。これでは誤読が起こる上、なかなかイメージがわかりません。そこで、数学では適宜数学用語を導入して量を圧縮するとともに、誤読が起こらないように書き換えるのです。その結果が「 $f(x) = x^2$ はいたるところ連続である」という問題文なのです。

自動証明のアイデア 4 によれば「論理構造に従って機械的に分析すれば形式的な証明が得られる（はず）」です。伝統的な自動証明の方法論では、まず「一番外側の論理記号」に着目します。論理式 (1) の一番外側の論理記号は $\forall x$ です。つまり、この命題は「任意の

x について... が成り立つ」ということを主張している全称命題なのだから、全称記号に関する推論ルールを逆適用することで後ろ向きに証明すればよい、と考えるのです。

全称命題 $\forall a P(a)$ を証明するにはどうすればよいでしょう。数学の分野を超えて通用する方法は唯一つ。それは、任意の a について $P(a)$ が成り立つことを示すことです。馬鹿馬鹿しいほど当たり前に聞こえるかもしれませんが、論理は「当たり前」だけで構成されています。そこに個性を発揮する必要はありません。

ただし、全称命題を証明するために独自の公理を用意している数学の分野もあります。自然数論における数学的帰納法がそうです。

$$(A(0) \wedge \forall n (A(n) \rightarrow A(n+1))) \rightarrow \forall n A(n)$$

ここで、 n は自然数全体を動く変数です。よく見ると最後の項が全称命題 $\forall n A(n)$ になっています。つまり、全称命題 $\forall n A(n)$ に至る筋道として、「任意の a について $A(a)$ が成り立つ」を証明する以外に、「 $A(0)$ が成り立ち、かつ、任意の n について、 $A(n)$ を仮定すると $A(n+1)$ が成り立つ」を証明する、があり得るというのが、自然数論の特徴であり本質なのです。

今、対象としている実数には、数学的帰納法のようなタイプの公理はありませんから、全称命題を証明する筋道はひとつです。ですから、証明の冒頭には迷わずに次の一文を書けばよい。

「 x を任意の実数とする。」

これでひとつ論理記号を外すことができました。(1) の式は次のように簡略化されます。

$$\forall \epsilon > 0 \exists \delta > 0 \forall y (|y - x| < \delta \rightarrow |y^2 - x^2| < \epsilon) \quad (2)$$

次に現れる論理記号も $\forall$ ですから、次の行も同様に処理します。

「 ϵ を任意の正の実数とする。」

こうして、またひとつ論理記号を外すことができ、式 (3) が得られます。

$$\exists \delta > 0 \forall y (|y - x| < \delta \rightarrow |y^2 - x^2| < \epsilon) \quad (3)$$

次に現れる論理記号は $\exists$ です。 $\exists$ の処理は、 $\forall$ のように易しくはありません。なぜなら、論理式 (4) を満たすような $\delta > 0$ を見つけなければならないからです。

$$\forall y (|y - x| < \delta \rightarrow |y^2 - x^2| < \epsilon) \quad (4)$$

ここに探索の必要が生じます。命題 $\exists a P(a)$ が与えられたとき、 $P(t)$ が真であるような t を $\exists a P(a)$ の **witness (証拠)** と呼びます。定数と変数を基に、関数を再帰的に適用して構成されるものを**項 (term)** と呼びますが、項の集合の上を巧く探索して $P(t)$ が真であるようなものが見つかれば当然のことながら $\exists a P(a)$ は真だということになるでしょう¹。

¹witnessを見つけることで $\exists a P(a)$ を示すことを「構成的な証明」と呼びます。すべての非構成的な証明を構成的なものに置き換えることができるとは限りませんが、多くの場合そのように期待できます。

どう探索するかは後回しにして、首尾よく δ の witness である $t(x, \epsilon)$ を発見できたと仮定して、次に進んでみます。まずは発見できた witness を宣言しておきましょう。

「 δ を $t(x, \epsilon)$ とおく。」

次に現れる論理記号は $\forall y$ ですから、次の段には「 y を任意の実数とする」と書くことになるはずです。ここまでくれば、量子子 $\forall, \exists$ はなくなり、シンプルな式が残ります²。

$$(|y - x| < \delta \rightarrow |y^2 - x^2| < \epsilon)$$

後は、条件 $|y - x| < \delta$ と $|y^2 - x^2| < \epsilon$ という2つの不等式から y を消去した上で、 δ について解けば witness が求まると期待できます。実際、関数記号として四則演算のみを含む多項式に関する理論では y は必ず消去でき、witness $t(x, \epsilon)$ が存在するかどうかを少なくとも自動判定することができます [2][3]³。

ここでは機械に頼らずに、ただしなるべく機械的に、この方程式を解いてみましょう。 $|y| \leq |x| + |y - x| < |x| + \delta$ であることから、

$$\begin{aligned} |y^2 - x^2| &= |(y + x)(y - x)| \\ &\leq (|y| + |x|)|y - x| \\ &< (2|x| + \delta)\delta \\ &\leq \epsilon \end{aligned}$$

よって、

$$\delta^2 + 2|x|\delta - \epsilon \leq 0$$

これを解くと、

$$-|x| - \sqrt{x^2 + \epsilon} \leq \delta \leq -|x| + \sqrt{x^2 + \epsilon}$$

$\delta > 0$ という条件と併せることで、求めるべき δ の witness (のひとつ) である $-|x| + \sqrt{x^2 + \epsilon}$ を得ることができます。以上を整理すると次のような証明になるでしょう。

(証明)

x を任意の実数、 ϵ を任意の正の実数とする。ここで、 δ を $-|x| + \sqrt{x^2 + \epsilon}$ とおく。 $|y - x| < \delta$ であるような任意の実数 y について $|y| \leq |x| + |y - x| < |x| + \delta$ であるから、以下の不等式が成り立つ。

$$|y^2 - x^2| = |(y + x)(y - x)|$$

²ただし、絶対値記号の中には、場合分けが「圧縮」されています。本来ならば、場合分けを「解凍」してひとつずつチェックする必要があります。

³その他の初等関数に拡張した場合、一般にどうなるかは未解決問題です。

$$\begin{aligned}
&\leq (|y| + |x|)|y - x| \\
&< (2|x| + \delta)\delta \\
&\leq \epsilon
\end{aligned}$$

$|y^2 - x^2| < \epsilon$ となり, $f(x) = x^2$ がいたるところ連続であることが示された. $\square$

$f(x) = x^2$ の連続性の証明では, どう工夫しても δ の witness から x を取り除くことができません. ということは, 論理式 (1) によく似ている次の論理式は正しくないことがわかります.

$$\forall \epsilon > 0 \exists \delta > 0 \forall x \forall y (|y - x| < \delta \rightarrow |y^2 - x^2| < \epsilon) \quad (5)$$

論理式 (5) は「 $f(x) = x^2$ は実数上一様連続である」を表わします. つまり, $f(x) = x^2$ は連続だが一様連続ではない関数なのです.

3 証明を書くことの理論的困難さ

17 世紀の数学者ライプニッツは「式が代わりに考えてくれる」という名言を残しました. 未解決問題をいかに解くか以上に, 数学の言語 (記号・記法・文法) をいかに整えるかが数学の発展を左右する, ということをライプニッツは強く意識していたのです. 今では想像しがたいことですが, 等号 (=) が発明されたのは 16 世紀も半ばになってからのことで, それまで式は長い文章で書かれていました. 等号が導入されることによって, 私たちは式の意味を考えずに機械的に変形し答えを求めることができるようになりました. 数学史上では, 微積分学の始祖をニュートンに譲ることが多いライプニッツですが, 彼が導入した $\int$ や $\frac{dy}{dx}$ といった記号によって微積分を理解できる人口は爆発的に増えました. 現在, 秀才天才でなくとも微積分を学ぶことができるのは, ライプニッツの機械論的な世界観に依るところが大きいと言えるでしょう.

同様に, 自然言語で書かれた命題は, 形式化されることによって, ある程度機械的に証明することが可能になりました. ゲーデルの不完全性定理や, 素数定理などの極めて高度な定理が (自動ではないものの) 証明支援機を用いて次々に証明されています [4][5]. まさに「論理式が代わりに証明してくれる」のです.

ただ, 残念なことに存在命題の witness の探索は, 一般には手におえない問題であることが理論的にわかっています. 一階述語論理の決定不能性は, witness の探索範囲を限定することができないことを示しています [6]. 存在命題の証明における探索をいかに巧く枝刈りするか——そこに数学のアートがあると言えるのかもしれません.

参考文献

- [1] G. Gentzen, “Untersuchungen über das logische Schließen. I”, *Mathematische Zeitschrift*, Volume 39, Issue 2, 1934, pp.176-210.
- [2] A. Tarski, “A Decision Method for Elementary Algebra and Geometry”, *Quantifier Elimination and Cylindrical Algebraic Decomposition Texts and Monographs in Symbolic Computation* 1998, pp.24-84.
- [3] G. Collins, “Quantifier elimination for real closed fields by cylindrical algebraic decomposition”, *Automata Theory and Formal Languages 2nd GI Conference Kaiserslautern*, *Lecture Notes in Computer Science*, Volume 33, 1975, pp.134-183.
- [4] N. Shankar, *Metamathematics, Machines, and Gödel’s proof*, Cambridge University Press, 1997. ISBN: 9780521585330.
- [5] J. Avigad, K. Donnelly, D. Gray, P. Raff, “A formally verified proof of the prime number theorem”, *ACM Transactions on Computational Logic*, Volume 9, Issue 1, 2007, pp.1-23.
- [6] A. Church, “A note on the Entscheidungsproblem,” *Journal of Symbolic Logic*, Volume 1, 1936, pp.40-41.