Justus-Liebig-Universität Gießen

Zentrum für Philosophie und

Grundlagen der Wissenschaft

Schriftliche Ausarbeitung des Referats über

„Computer, Paradoxa und die Grundlagen der Mathematik“

von Gregory J. Chaitin

American Scientist, 90 (2002)

Referent: Diego Rybski

Vortragsdatum: 14.1.2004

Seminar: Posthumanismus. Die Transformation des Menschen durch die Technik

Prof. Dr. B. Kanitscheider

WiSe 2003/2004

Computer, Paradoxa und die Grundlagen der Mathematik – einige große Denker des 20. Jahrhunderts haben gezeigt, daß selbst in der strengen Welt der Mathematik Unvollständigkeit und Zufälligkeit weit verbreitet sind.

„Computers, Paradoxes and the Foundations of Mathematics – Some great thinkers of the 20th century have shown that even in the austere world of mathematics, incompleteness and randomness are rife“

Inhaltsverzeichnis

[1] Russell, Betrand, Earl, * Trelleck bei Monmouth 18.5.1872, † Plas Penrhyn bei Penrhyndeudraeth (Wales) 2.2.1970.
[2] Hilbert, David, * Königsberg 23.1.1862, † Göttingen 14.2.1943.
[3] Gödel, Kurt, * Brünn 28.4.1906, † Princeton (N.J.) 14.1.1978.
[4] Turing, Alan Mathison, * Paddington (London) 23.6.1912 , † Wilmslow, Cheshire 7.6.1954 („self-administered potassium cyanide while in a moment of mental imbalance“).

1. Über den Autor

Gregory J. Chaitin ist als Mathematiker an dem IBM Watson Forschungszentrum in Yorktown Heights bei New York tätig. Außerdem ist er Gastprofessor an der Universität von Buenos Aires und an der Universität von Auckland. In den letzten 35 Jahren war er maßgeblich an der Entwicklung der Algorithmischen Informationstheorie[5] beteiligt, mit der er sich bereits als Jugendlicher beschäftigte. Sein letzter Vorstoß gelang mit der Transformation der Algorithmischen Informationstheorie, so daß Vorhersagen über die Größe echter Computerprogramme gemacht werden können. Chaitin hat acht Bücher zum Thema veröffentlicht.

2. Einleitung

Der betrachtete Artikel[6] ist 2002 in der Zeitschrift „American Scientist“ (Ausgabe 90) erschienen. Es handelt sich um den Auszug eines Vortrags, der 1999 an der Universität von Massachusetts in Lowell gehalten wurde. In voller Länge ist der Vortrag in „Conversations with a Mathematician“ (Springer) zu finden.

Chaitin beginnt mit der Behauptung, daß der Computer zwar ein praktisches Ding und in der heutigen Gesellschaft unentbehrlich ist, aber dazu erfunden worden sei, um die philosophische Frage über die Grundlagen der Mathematik („foundations of mathematics“[7] (S. 164)) zu klären. Und zwar habe es mit David Hilbert begonnen, der zu Beginn des 20. Jahrhunderts die komplette Formalisierung der mathematischen Argumentation („mathematical reasoning“[8] (S. 164)) vorschlug. Es zeigte sich allerdings, daß dies nicht möglich ist und in diesem Zusammenhang stellte sich die Idee als Irrweg heraus. Andererseits war es ein Erfolg, denn Formalismen stellen zwar keine Errungenschaft für mathematische Argumentation oder Deduktion dar, wohl aber für Programmierung, Berechnung und Computerarbeit.

3.1 Russells logische Paradoxa

Betrand Russell nimmt eine Schlüsselposition ein, da er einige unangenehme Paradoxa in der Logik an sich entdeckte. Es sind Fälle bei denen das logische Denken korrekt ist, aber zu einem Widerspruch führt. Russell habe einen wesentlichen Beitrag geleistet, die Idee zu verbreiten, daß diese Widersprüche eine ernsthafte Krise darstellen und irgendwie gelöst werden müßten.

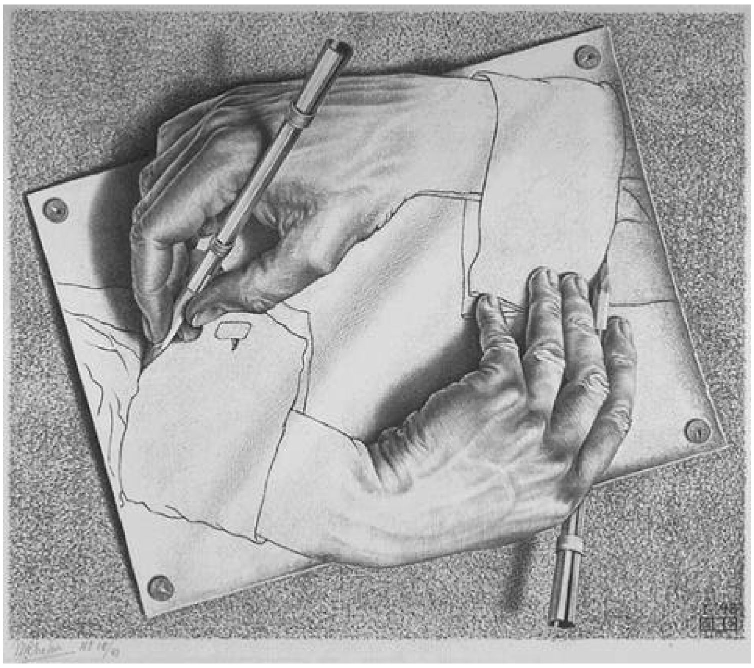
---

[5] „algorithmic information theory” (AIT)

[6] Link: http://www.umcs.maine.edu/~chaitin/amsci.pdf

[7] foundation: Fundament, Grundlage, Basis

Chaitin stellt drei Paradoxa vor:



In Chaitins Artikel wird M.C. Eschers Grafik „Zeichnen“ von 1948 gezeigt (Abb. 1). Die sich selbst zeichnenden Hände stellten eine visuelle Analogie zu Russells Paradoxon dar. „Die folgende Aussage ist wahr. Die vorhergehende Aussage ist falsch.“ Jede Aussage für sich ist widerspruchsfrei, in Kombination sind sie aber paradox, ganz wie die Hände Eschers Bildes.

- Das Russell Paradoxon: Man betrachte die Menge aller Mengen, die nicht Teilmengen ihrer selbst sind. Man frage dann: „Ist diese Teilmenge in sich selbst enthalten?“ Wenn es eine Teilmenge seiner selbst ist, dann darf sie es nicht sein, und umgekehrt.

  Anschaulicher wird es beim Barbier von Sevilla. Der Barbier rasiert alle Männer, (und nur die) die sich nicht selbst rasieren. Das klingt vernünftig bis man fragt, „Rasiert sich der Barbier selbst?“ Er rasiert sich selbst, aber nur wenn er sich nicht selbst rasiert. Russell betont, daß man bei der Betrachtung des mathematischen Konzepts von Mengen nicht einfach über ein logisches Problem hinwegsehen kann.
- Das Epimenides Paradoxon oder Paradoxon des Lügners war bereits den alten Griechen bekannt und ist der Vorgänger des Russell Paradoxons, welches die mengentheoretische Antwort repräsentiert.

  „Diese Aussage ist falsch!“ Wenn sie falsch ist, muß sie folglich richtig sein, aber wenn sie wahr ist, muß sie falsch sein.
- Die Version in zwei Sätzen lautet: „Die folgende Aussage ist wahr. Die vorhergehende Aussage ist falsch.“ Jede einzelne Aussage ist in Ordnung, aber die Kombination macht keinen Sinn.

Chaitin merkt an, daß man möglicherweise verleitet ist diese Paradoxa als Scherzfragen oder sinnlose Wortspiele zu betrachten, daß aber einige der größten Köpfe des 20. Jahrhunderts sie sehr ernst genommen haben.

Problematisch ist die Begriffsbildung, denn zum Beispiel wird von dem Mengenbegriff nicht gesagt, ob er Selbstreferenzen zuläßt oder nicht? Überhaupt werden Selbstreferenzen nicht

[8] reasoning: logisches Denken, Gedankengang, Argumentation, Beweisführung, Schlußfolgerung

diskutiert, obwohl sie bei allen Paradoxa, also auch den noch folgenden, auftreten. Deshalb bleibt die Klassifizierung unklar, also ob das Paradoxe auf den Selbstbezug zurückzuführen ist.

3.2 Hilberts Rettungsplan

Eine der Reaktionen auf diese Krise der Logik sei Hilberts Versuch gewesen in Formalismen zu flüchten. Wenn man in Schwierigkeiten mit vernünftig klingender logischer Argumentation kommt, führt Chaitin ironisch aus, dann könne man versuchen symbolische Logik zu benutzen, um eine künstliche Sprache zu erschaffen, bei der man die Regeln sehr sorgfältig aufstellen muß, so daß keine Widersprüche plötzlich auftauchen. Schließlich sei die Alltagssprache mehrdeutig und man wisse nie, auf was sich ein Pronomen bezieht.

Hilberts Idee war es, eine perfekte künstliche Sprache der Beweisführung, also der Mathematik, der Deduktion zu erschaffen. Deshalb habe Hilbert die Rolle der axiomatischen Methode unterstrichen, wobei man von einem Satz grundsätzlicher Postulate (Axiome) und wohl definierter Regeln der Deduktion ausgeht, um gültige Postulate herzuleiten. Dabei sollte vor allem bei den Regeln absolute Klarheit herrschen, also die Verwendung von Definitionen, elementaren Konzepten, die Grammatik und die Sprache. Auf diese Weise solle jeder nachvollziehen können, wie Mathematik gemacht wird, was zwar nicht der Entwicklung neuer Mathematik dient, aber philosophisch signifikant sei.

In der Tradition von Leibniz, Boole, Frege und Peano sei Hilbert der strikten formalen Mathematik gefolgt. Es stellte sich jedoch heraus, daß es nicht möglich ist, die gesamte Mathematik bis ins letzte Detail zu formalisieren. Hilbert habe zwar geirrt, aber in einer fruchtbaren Weise, denn durch seine Fragen entstand eine komplett neue Disziplin namens Metamathematik. Sie ist ein introspektiver Bereich der Mathematik, bei dem untersucht wird, was von der Mathematik erreicht werden kann und was nicht.

Das grundlegende Konzept ist es, daß man die Bedeutung (der mathematischen Sätze) von der Spache abtrennen kann, sobald man die Mathematik in eine solche Hilbertsche Sprache eingebettet hat. Dann genügt es, den strengen komplett formalen axiomatischen Regeln zu folgen, so daß die Bedeutung hinter der Sprache außer Acht gelassen werden kann. Ein formal axiomatisches System wird als vollständig erachtet, wenn von einer Aussage, wie 0=1, gezeigt werden kann, daß die Aussage entweder wahr oder falsch ist. Hilbert sei es vorgeschwebt, so exakte Regeln aufzustellen, daß man jeden Beweis einem

unvoreingenommenen Schiedsrichter geben kann, und der, einer mechanischen Prozedur[9] gleichend, feststellen kann, ob der Beweis den Regeln folgt oder nicht.
Hilberts Idee sei es nicht gewesen, auf diese Weise Mathematik zu betreiben, sondern die Stärke bzw. Leistungsfähigkeit der Mathematik zu erforschen.
Hilberts Programm wird noch heute verfolgt. Dabei ist die Axiomatisierung - bei der schematisch operiert wird und die Elemente gleichgültig behandelt werden - umstritten. Der Hauptstrom unterscheidet heutzutage zwischen faktischer Bedeutung und formaler Bedeutung, denn die Semantik kann nicht aus dem Nichts entstehen.

3.3 Gödels Unvollständigkeit

Hilbert habe fest daran geglaubt, das Kunststück zu vollbringen. Um so erschütternder sei es gewesen, als Kurt Gödel 1931 zeigte, daß Hilberts Rettungsplan niemals realisiert werden könnte, nicht einmal im Ansatz. De facto ist es nicht möglich ein formal axiomatisches System für die gesamte Mathematik ausfindig zu machen, bei dem unmittelbar ersichtlich wäre, ob etwas wahr oder falsch ist.
Gödel fand heraus, daß ein solches System immer entweder inkonsistent oder unvollständig ist. Nimmt man an, es enthalte nur wahre Aussagen, dann wird es nicht alle Wahrheiten wiedergeben, und umgekehrt. Nimmt man an, die Axiome und Regeln der Deduktion lassen es nicht zu, falsche Theoreme zu beweisen, dann wird es wahre Theoreme geben, die nicht gezeigt werden können.
Gödels Beweis lasse sich mit dem Paradoxon „Diese Aussage ist falsch!“ veranschaulichen, das dann laute „Diese Aussage ist nicht zu beweisen!“ Gelingt es die Aussage in elementarer Zahlentheorie oder Arithmetik aufzustellen (was schwer ist, da es eine mathematische Aussage ist, die sich selbst beschreibt), dann hat man ein Problem, denn wenn sie tatsächlich nicht zu beweisen ist, wie sie selber sagt, dann ist es wahr und die Mathematik ist unvollständig (würde ein Beweis gelingen, dann wäre die Aussage falsch).
Wirft man einen Blick auf Gödels Originalveröffentlichung, dann findet man Ähnlichkeiten zur Programmiersprache LISP, was daran liegt, daß Gödel viele Funktionen rekursiv definiert hat. Obwohl es 1931 noch keine Computer oder Programmiersprachen gab, könne man also im Kern von Gödels Arbeit eine solche sehen.
Damals hinterließ Gödels Schlußfolgerung, also daß ein formales axiomatisches System immer entweder inkonsistent oder unvollständig ist, für viele Leute einen absolut

[9] „mechanical procedure“ (S.166)

vernichtenden Eindruck: Die gesamte traditionelle Philosophie der Mathematik war am Boden zerstört.
Eine Einführung in den Beweis Gödels geben E. Nagel, J. R. Newman in „Der Gödelsche Beweis“[10].

3.4 Turings Maschine

Der nächste Schritt vorwärts wurde fünf Jahre später gemacht, als Alan Turing die Unberechenbarkeit entdeckte. Während Hilbert nur von einer „mechanischen Prozedur“ (siehe oben) spricht, handelt es sich bei Turing explizit um eine Maschine, die seitdem als Turingmaschine bekannt wurde. Auch Turing benutzt in seiner Originalveröffentlichung[11] eine Programmiersprache, jedoch sei es eine viel niedrigere als bei Gödel. Es ist viel mehr eine Maschinensprache, die heute niemand zum Programmieren verwenden wollte. Trotz der Einfachheit von Turings hypothetischer Maschine und der Kompliziertheit der Maschinensprache, ist dieses System sehr flexibel.
Turing behauptet, daß eine solche Maschine jede Rechnung durchführen kann, die ein Mensch auch bewerkstelligen kann. Turing habe danach gefragt, was die Maschine kann und was nicht. Dabei stößt er sofort auf das Halteproblem. Läßt man ein Zeitlimit zu, dann gibt es kein Problem. Will man wissen, ob ein Programm nach einem Jahr fertig ist oder nicht, dann läßt man es einfach laufen. Turing hat gezeigt, daß ein Problem auftritt, wenn man kein Zeitlimit vorgibt, wenn man versucht es herzuleiten, ob ein Programm stoppen wird, ohne es auszuprobieren.
Bei Turings Argumentation geht man von einem Programm aus, das prüft ob ein anderes Programm enden wird. Ein zweites Programm wendet diesen Terminationsprüfer[12] an, um ein beliebiges weiteres zu prüfen. Falls das untersuchte Programm endet, soll das zweite Programm, welches den Terminationsprüfer anwendet, in eine Endlosschleife gelangen. Was passiert nun, wenn das Programm auf eine Kopie von sich selbst angewendet wird? Wenn es selbst endet, dann landet es in einer Endlosschleife, was wiederum bedeutet, daß es nicht endet. Wenn es nicht stoppt, dann wird der Terminationsprüfer das ermitteln und es gelangt nicht in die Endlosschleife, weshalb es endet. Dieses Paradoxon veranlaßte Turing zu dem Schluß, daß sich ein universeller Terminationsprüfer nicht umsetzen läßt. Diese Argumentation ist eine Parallele zu Gödels Beweis, enthält jedoch wieder das Moment der Rückkopplung.

---

[10] Oldenbourg 2003
[11] Link: http://www.abelard.org/turpap2/tp2-ie.asp
[12] „termination tester“ (S. 168)

Chaitin unterstreicht, daß Turing daraus auch gleich ein Korollar abgeleitet hat: Wenn es keine Möglichkeit gibt, mit einer Berechnung vorherzusagen, ob ein Programm halten wird, dann kann es auch keinen Weg geben, es durch logische Argumentation in Erfahrung zu bringen. Turings Erfolg sei es also gewesen zu zeigen, daß es kein formales axiomatisches System ermöglichen kann, abzuleiten ob ein Programm stoppen wird. Somit ist jedes axiomatische System unvollständig.

3.5 Chaitins Zufälligkeit in der Mathematik

In den späten 50er Jahren stieß Chaitin als junger Wissenschaftler auf das Thema. Er habe sich auch etwas mit den Grundlagen der Physik beschäftigt und gelernt, daß in der Physik viele Dinge zufällig, intrinsisch unvorhersagbar, ablaufen. Da habe er sich gefragt, ob es in der reinen Mathematik nicht auch solchen Zufall gibt, und er vermutete daß das der wahre Grund für die Unvollständigkeit sei.

Als Beispiel nennt Chaitin die Primzahlen. Einzelne treten sehr unvorhersagbar auf, aber es gibt auch statistische Muster. In der sogenannten Primzahltheorie kann die gesamte Verteilung ziemlich genau beschrieben werden. Sieht man aber genauer hin, dann stellt man fest, daß die einzelnen Zahlen doch sehr zufällig auftreten. Während diese Zufälligkeit beim genauen Hinsehen in der Physik weitgehend akzeptiert ist, überrascht es doch so etwas auch in der Mathematik zu finden[13]. Diesen Vergleich Chaitins muß man kritisch betrachten, da die Fälle vollkommen unterschiedlichen Situationen entstammen.

Also dachte Chaitin darüber nach, ob diese inherente Zufälligkeit in der Mathematik der Grund für die beschriebene Unvollständigkeit sein könnte. In den 60ern kamen er und A.N. Kolmogorov unabhängig voneinander auf einige neue Ideen und gründeten die Algorithmische Informationstheorie, in der es um die Messung von Komplexität von Berechnungen geht[14]. Nachdem Turing mit der Idee eines perfekten Computers - der keine Fehler macht, so viel Zeit und Platz hat wie er braucht - aufkam, war es ein logischer Schritt, die Rechenzeit als Maß für die Komplexität anzusehen. Bereits 1950 habe von Neumann die Wichtigkeit der Zeit bei Rechenkomplexität hervorgehoben.

Chaitin hatte die Idee, darüber hinaus die für eine bestimmte Aufgabe benötigte Größe eines Computerprogramms, die Menge an Information, die man einem Programm gibt, zu betrachten. Dies sei interessant, da die Programmgrößenkomplexität mit dem Konzept der Entropie aus der Physik verknüpft ist. Der Autor erwähnt kurz Ludwig Boltzmann,

[13] In Abb. 3 werden der radioaktiven Zerfall (Anzahl der Atome gegen die Zeit) und ein integratives Histogramm des Auftretens von Primzahlen skizziert. Die glatten Kurven weisen bei der Vergrößerung mit der dargestellten Lupe Stufen auf.

statistische Mechanik und Thermodynamik. Die Entropie ist ein Maß für Unordnung, Chaos, Zufälligkeit eines physikalischen Systems. Beim Kristall ist sie niedrig und beim Gas hoch. Außerdem ist die Entropie mit der philosophischen Frage verknüpft, warum die Zeit in eine Richtung läuft. Nach dem zweiten Hauptsatz der Thermodynamik nimmt die Entropie im Laufe der Zeit zu. Um ein Gas zu beschreiben, etwa die Positionen aller Atome, wäre ein großes Programm nötig. Wegen der gleichmäßigen Struktur eines Kristalls, tut es hier ein kleines Programm. Folglich stehen Programmgrößenkomplexität und Entropie in Beziehung zueinander.

1960 habe Ray Solomonoff vorgeschlagen, daß die Programmgrößenkomplexität auch mit der Wissenschaftstheorie[15] verknüpft ist. In Analogie[16] dazu, daß eine einfache Theorie die bessere ist, ist auch ein kurzes Programm das bessere, denn schließlich sei eine Theorie auch nur ein Computerprogramm zur Vorhersage von Beobachtungen. Somit stellt ein knappes Computerprogramm die bessere Theorie dar. Hiermit zeigt Chaitin eine sehr instrumentalistische Sichtweise, die vom wissenschaftstheoretischen Standpunkt zweifelhaft ist. Um Chaitins Gedankengang folgen zu können, soll sie mal hingenommen werden. Wenn das kürzeste Programm zur Wiedergabe gewisser experimenteller Daten so groß ist wie die Daten selbst, dann ist die Theorie unzureichend und die Daten sind unbeschreiblich und zufällig. Auf die Frage, ob in diesem Fall die fehlende Kompressibilität auf die Theorie bzw. das Programm oder auf die Daten zurückzuführen ist, geht der Autor später ein. Eine Theorie sei nur dann gut, wenn sie die Daten auf viel kleinere theoretische Annahmen und Regeln der Deduktion zurückführe.

Chaitins Idee war es also Zufälligkeit als etwas aufzufassen, das sich nicht komprimieren läßt, und sie auch so zu definieren. Beschäftigt man sich mit der Programmgrößen- oder Informationskomplexität anstelle der Laufzeitkomplexität, dann stößt man auf die Frage, wie man sicher sein kann, daß man das kleinst mögliche Programm hat. Man könne es nicht wissen und diese Aufgabe entziehe sich der mathematischen Beweisführung.

---

[14] „computational complexity“ (S. 169)
[15] „philosophy of the scientific method“ (S. 170)
[16] Chaitin benutzt hier nicht das Wort Analogie.

Er zeigt dies nicht, sondern beschränkt sich darauf, ein aktuelles Forschungsergebnis wiederzugeben: Hat man *n* Bits an Axiomen, dann läßt es sich nicht zeigen, daß ein Programm das kleinst mögliche ist, falls es selbst länger als *n* Bits ist. Im Allgemeinen könne man also nicht die Programmgrößenkomplexität von etwas ausrechnen, denn dazu ist es nötig, die Größe des knappsten Programms, das es berechnet, zu kennen. Chaitin begründet seine Behauptungen mit folgender Argumentation: Mathematiker benutzen in der Regel kurz und knappe Axiome, da sie andernfalls niemand glauben würde. Eigentlich gebe es aber eine riesige Menge mathematischer Wahrheiten - eine unendliche Menge an Information - aber jeder Satz von Axiomen beschreibe immer nur eine kleine endliche Menge davon. Genau deshalb sei Gödels Unvollkommenheit vielmehr natürlich und unvermeidlich als mysteriös oder kompliziert. Somit schließt sich also der Kreis mit Chaitins anfänglicher Behauptung (vgl. Einleitung), daß der Computer erfunden worden sei, um der Frage über die Grundlagen der Mathematik nachzugehen.

Chaitin geht in dem Artikel nicht auf seine Zahl Ω ein, die wie folgt definiert ist[17]:

$$\Omega \equiv \sum_{p\,halts} 2^{-|p|}$$

Sie gibt die Wahrscheinlichkeit dafür an, daß für einen Satz an Befehlen, eine universelle Turingmaschine halten wird, wobei $|p|$ die Größe in Bit des Programms p ist. Die Ziffern in Ω seien zufällig und könnten nicht im Voraus berechnet werden. Nichtsdestoweniger ist es bekannt, daß Ω eine irrationale Zahl ist. Panu Raatikainen wirft in seiner Buchbesprechung „Exploring Randomness and The Unknowable“[18] einen sehr kritischen Blick auf diese beiden Bücher Chaitins und die Behauptungen über Ω. Es sei gezeigt worden, daß Chaitins philosophische Interpretationen seiner Arbeiten unbegründet und falsch seien, weshalb Raatikainen bezweifelt, daß Chaitins Ansatz die wahre Quelle der Unvollständigkeit erklären könnte. Zwar seien Chaitins Ergebnisse nicht uninteressant, jedoch sei ihre Relevanz für die Grundlagen der Mathematik vollkommen übertrieben (S. 995).

4. Ausblick

Chaitin werde oft nach einem Beispiel gefragt, bei dem sich ein bestimmtes Ergebnis der Algorithmischen Informationstheorie den Möglichkeiten der mathematischen Argumentation entzieht. Jahrelang habe er Fermats[19] letztes Theorem[20] genannt. Andrew Wiles präsentierte

[17] vgl. http://mathworld.wolfram.com/

[18] Notices of the AMS, 48 (2001).

[19] Fermat, Pierre de, * Beaumont-de-Lomagne (Dép. Tarn-et-Garonne) 17.(?)8.1601, † Castres bei Toulouse 12.1.1665.

1993 einen Beweis, der zwar einen Fehler enthielt, dessen Richtigkeit aber heute unumstritten ist. Algorithmische Informationstheorie zeigt, daß es eine Menge Dinge gibt, die man nicht beweisen kann, aber bei Einzelfällen mathematischer Fragen könne sie keine Antwort geben. Somit befindet sich Chaitin in einer unerfreulichen Situation, denn als vernünftige Theorie müßte die Algorithmische Informationstheorie auch Vorhersagen machen.

Chaitin stellt sich selbst die Frage, wie es in der Mathematik trotz der Unvollständigkeit so viel Fortschritt geben kann. Vor lauter Unvollständigkeiten könnte man glauben, daß es keine weiteren Errungenschaften der Mathematik geben könnte. Glücklicherweise scheine dies nicht der Fall zu sein. Abschließend fordert er die jungen Mathematiker der nächsten Generation auf, zu zeigen warum es nicht so ist.

[20] Im Jahre 1653 stellte Fermat die Behauptung auf, daß es für die Gleichung $x^n+y^n=z^n$ keine ganzzahlige Lösung mit $n>2$ gäbe. Dazu notierte er: „Ich habe hierfür einen wahrhaft wunderbaren Beweis, doch ist der Rand hier zu schmal, um ihn zu fassen.“ Paul Friedrich Wolfskehl, ein Bankier aus Darmstadt, setzte im Jahre 1908 100.000 Goldmark für denjenigen aus, der als erster einen Beweis in einer Fachzeitschrift veröffentlicht, wobei Einsendeschluß der 23.9.2007 war. Erst am 27.6.1997 wurde der Wolfskehl-Preis in Höhe von 80.000DM an Andrew Wiles überreicht (Quellen: http://www.rzuser.uni-heidelberg.de/~ci3/fermat.pdf, http://de.wikipedia.org/wiki/Fermats_Letztes_Theorem).