OXFORD

Martin Davis: An Overview of his Work in Logic, Computer Science, and Philosophy

Liesbeth De Mol,*[iD] Yuri V. Matiyasevich,†[iD] Eugenio G. Omodeo,‡[iD] Alberto Policriti,§[iD] Wilfried Sieg‖[iD] and Elaine J. Weyuker¶[iD]

*UMR 8163 Savoirs, Textes, Langage, CNRS – Université de Lille, 59653, Villeneuve d'Ascq, France

†Laboratory of mathematical logic and discrete mathematics, St Petersburg Department of Steklov Mathematical Institute of the Russian Academy of Sciences, 191023 St Petersburg, Russia

‡Department of Mathematics, Informatics, and Geosciences, University of Trieste, 34127 Trieste, Italy

§Department of Mathematics, Computer Science, and Physics, University of Udine, 33100 Udine, Italy

‖Department of Philosophy, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, U.S.A

¶Department of Computer Science, University of Central Florida, Orlando, Florida 32816, U.S.A

ABSTRACT

In his autobiographical essay written in 1999, 'From logic to computer science and back', Martin David Davis (1928 3 8–2023 1 1) indicated that he viewed himself as a logician *and* a computer scientist. He expanded the essay in 2016 and expressed a new perspective through a changed title, 'My life as a logician'. He points out that logic was the unifying theme underlying his scientific career. Our paper attempts to provide a consistent vision that illuminates Davis's successive contributions leading to his landmark writings on computability, unsolvable problems, automated reasoning, as well as the history and philosophy of computing.

*Orcid.org/0000-0001-8520-2493

†Orcid.org/0000-0001-7046-3746

‡Orcid.org/0000-0003-3917-1942

§Orcid.org/0000-0001-8502-5896

‖Orcid.org/0000-0002-7130-0524. E-mail: sieg@cmu.edu.

¶Orcid.org/0000-0002-1660-199X

> *When I was invited to speak at the weekly colloquium of the Berkeley mathematics department, I took the opportunity to express my view of computability theory as an autonomous branch of mathematics. Tarski, who was in the audience, took strong exception during the discussion after my talk.* Martin Davis [2020]

The current paper attempts to provide a consistent vision that illuminates Davis's successive contributions, leading him from his early *passionate* (his word) interest in the foundations of real analysis, to his landmark writings on computability, unsolvable problems, automated reasoning, and the history and philosophy of computing.

1. PERSONAL AND ACADEMIC LIFE

Martin David Davis was born in 1928 to immigrant Polish Jewish parents in the Bronx, New York City, which was then a largely working class community. A few years before Davis started high school, New York opened a specialized public high school, The Bronx High School of Science, offering a highly enriched free education with admission determined by a student's exam score. Davis graduated in 1944 and then attended the City College of New York, which also provided a free education to New York City students. He majored in mathematics and studied under Emil Leon Post, a man whom Davis greatly admired and who highly influenced Davis's life work. After graduation in 1948, Davis went to Princeton to study for his Ph.D. in mathematical logic. His dissertation advisor was Alonzo Church, another major figure in the early world of computation theory. He earned his degree in 1950.

Davis spent the decade following his doctorate, as he recalls in [Davis, 2016a] (see also [Calvert *et al.*, 2024]), moving among various academic institutions: University of Illinois at Urbana-Champaign, Institute for Advanced Study in Princeton, University of California–Davis, Ohio State University, Hartford Graduate Division of Rensselaer Polytechnic Institute. In 1951 Davis married Virginia Palmer[1] with whom he had two sons: Harold and Nathan. Around 1960, Davis settled in New York City and developed a program in logic at Yeshiva University. From 1965 to his retirement in 1996, he was a faculty member at New York University's Courant Institute of Mathematical Sciences, first as a professor of Mathematics and later of Computer Science. He served as Chair of Computer Science from 1988 to 1990. After retiring from NYU, he lived in Berkeley and continued to work as a "Visiting Scholar" at the university 'where Alfred Tarski had developed a world-class logic group' [Davis, 2016a, p. 31].

---

[1] Their marriage would last 71 years till they both passed away on January 1, 2023.

2. COMPUTABILITY AND UNSOLVABILITY

The main focus, henceforth, will be on aspects of Davis's activity that have moved forward computer science and its foundations. Unavoidably, his contributions to logic as such (for example in [Davis and Fechter, 1991]) and his explorations of the foundations of mathematics and analysis will remain in the shadows, except for a quick account in Sec. 5.

In May of 1950, Davis's dissertation *On the Theory of Recursive Unsolvability* was recommended for acceptance by Princeton's Department of Mathematics.[2] When expressing his gratitude "toward the various persons whose help has meant so much in the preparation of this dissertation", Davis thanked Alonzo Church for reading the manuscript several times and offering "valuable criticisms and suggestions". That standard "Thank you!" of a doctoral student to his advisor is followed by a striking acknowledgement of intellectual debt: "It was Professor Emil L. Post who first introduced me to the theory of recursive functions, and who is largely responsible for my present point of view toward the subject." That point of view was not just Davis's point of view in 1950, but persisted throughout his career.

The thesis — as we will argue next — is a remarkable document.[3] In Part I, Kleene's theory of recursive functions is recast, generalized, and shown to be equivalent to a similar one of Post's. One deep expression of the equivalence is the theorem stating that a set is recursively enumerable if and only if it is normal. That is, any set of natural numbers that is the range of a recursive function can be generated by a normal system and vice versa. Normal systems were introduced by Post and are essentially very simple rewrite systems for strings. In Part II, Kleene's arithmetical hierarchy is extended into the transfinite along the constructive second number class generating the hyperarithmetical hierarchy. Central results were presented in a talk at the 1950 International Congress of Mathematicians; see [Davis, 1952]. They, together with then recent results, are also described in the last sections of [Davis, 1958]. A detailed development of the theory of hyperarithmetical sets and Davis's seminal role is found in Moschovakis's [2016]. Finally, Part III focuses on an "irresistible" problem, namely, Hilbert's Tenth Problem. The thesis exhibits characteristic traits of Davis's work; namely, a beautifully clear mathematical exposition as well as a deep desire to show the broad context for the introduction of mathematical notions and for their systematic development.

The broad context is sketched in the Introduction to the thesis. There, Davis not only traces the development of recursive function theory but also describes its "most spectacular results". These include a number of logically equivalent

---

[2] Davis had just turned twenty-two. Recursive functions are a key formalism within computability theory. Essentially, these are functions that work on natural numbers. They were used as a formal model for effective calculability by Church.

[3] In his *Notices of the AMS* interview [Jackson, 2008, p. 563], Davis asserted: "By the time I graduated [from] City College, I knew I wanted to be a logician. I had written a term paper for an advanced logic course in the Philosophy Department, which in a way was a first draft of part of what was later my dissertation."

results related to the impossibility of solving certain mathematico-logical problems in a finite number of steps. The halting problem for Turing machines is one such instance. Most important to Davis's exposition then was Post's elegant mathematical formulation of Turing machines to offer an alternative proof to the unsolvability of the halting problem which was then used to prove that the word problem for semi-groups is also unsolvable. This then recent research result was presented in Davis's textbook *Computability and Unsolvability* [1958]; a broad swath of unsolvable problems is found in Davis's [1977b]. Not only is Post's result established in the textbook, but Post's way of defining Turing machines, as production systems, provided the basis for the exposition of computation theory. Indeed, central results coming from mathematical logic (incompleteness and undecidability theorems) are established on that very basis. It has to be emphasized most strongly that Davis in 1958 thought of the subject no longer as "recursive function theory" (as he had done in his dissertation) but rather as "computability theory". At the time, *Computability and Unsolvability* was one of the founding texts of an emerging new field, computer science.

This deep connection to theoretical computer science is seen easily from a contemporary perspective; it is directly pointed out in the textbook co-authored with Elaine Weyuker, *Computability, Complexity, and Languages — Fundamentals of Theoretical Computer Science* [Davis and Weyuker, 1983].[4] In the Preface they write:

> Theoretical computer science is the mathematical study of models of computation. As such, it originated in the 1930s, well before the existence of modern computers, in the work of the logicians Church, Gödel, Kleene, Post, and Turing. This early work has had a profound influence on the practical and theoretical developments of computer science. Not only has the Turing-machine model proved basic for theory, but the work of these pioneers presaged many aspects of computational practice that are now commonplace.

For Davis this quite dramatic change of conceptual perspective from recursive function theory to computability theory was based on the fact that Post completely captured Turing's way of analyzing the mechanical calculations of a human being via "production systems" in [Turing, 1937]. Indeed, in [1950], Turing extended Post's techniques to establish the undecidability of the word problem for semigroups with cancellation; then in [Turing, 1954], he used Post's concept of deterministic production systems (and dubbed them "substitution puzzles") for two goals: (1) to discuss the methodological problem of articulating a mathematically precise and adequate concept of computation, and (2) to prove central undecidability results. The deep underlying conceptual unity of Post's and Turing's approaches is palpable in Davis's book. It is explicit in

---

[4] A much enriched second edition, [Davis *et al.*, 1994], was co-authored with Ron Sigal.

a paper, "Conceptual confluence in 1936: Post and Turing" [Davis and Sieg, 2015]. This is standardly discussed under the headings of Church's thesis and Turing's thesis.[5]

In his dissertation Davis had listed "spectacular results" of, what he called then, recursive function theory. He made a real contribution to this field by editing a collection of most of the original papers in which these results had been established. The collection was published as *The Undecidable: Basic Papers on Undecidable Propositions, Unsolvable Problems and Computable Functions* [1965]. To this day, it is a fundamental source book for anyone studying the history, philosophy, and theory of computability. Post has a major place in the volume as it includes not only some of his published papers but also the fascinating and previously unpublished *Absolutely unsolvable problems and relatively undecidable propositions — Account of an anticipation.* That paper contains Post's account of how he anticipated results of Church, Gödel, and Turing in the early 1920s.[6] Of course, in *The Collected Works of Emil L. Post*, all of Post's mathematical papers were published. Davis finished editing the volume in 1993 and remarked in the Preface that many of Post's students were deeply influenced by him and pursued successful careers in mathematics. About himself Davis wrote: "I am the only one whose own research interests have followed in the directions he pioneered, and so I feel it particularly appropriate that I play the role of his editor."

Davis worked on the history of computing not only through editing the two volumes described above, but by original explorations. Since he contributed to laying out a theory of computation, it should not come as a surprise that he also contributed to philosophical aspects of the field. We return to this in Sec. 6. However, there is one particular mathematical challenge that compelled his passionate engagement starting with his thesis, namely, Hilbert's Tenth Problem. The absolutely fascinating developments related to this problem are the topic of the next section.

3. HILBERT'S TENTH PROBLEM

One cannot overestimate the role played by Davis in resolving Hilbert's Tenth Problem. It is one of the twenty-three mathematical problems posed by David

---

[5] Church's thesis states that any effectively calculable function is general recursive (and λ-definable). It is extensionally equivalent to Turing's thesis stating that any computable function is computable by a Turing machine. At the core of the theses is the methodological question of the legitimacy of identifying an informal notion (effective calculability, computability) with a rigorous mathematical concept (general recursiveness, Turing machine computability).

[6] Post submitted this paper to the *American Journal of Mathematics* in 1941. However, the editor-in-chief Hermann Weyl rejected it, saying that the *Journal* was "*no place for historical accounts*" (quoted in [Davis, 1994]). At the same time, Weyl suggested focusing on the central mathematical results; that led to a shortened version that was ultimately published as [Post, 1943]. It contains Post's normal form theorem and a statement of Post's first thesis.

Hilbert [1900; 1901–1902] in 1900 at the Second International Congress of Mathematicians held in Paris.

Among these twenty-three problems the tenth is the only one which can be considered today as belonging to Computer Science (which did not exist in 1900). In this problem Hilbert asked about the solvability of *Diophantine equations*. They are equations of the form

$$P(x_1, \ldots, x_m) = 0, \tag{1}$$

where P is a polynomial with integer coefficients.

The equations are named after the Greek mathematician Diophantus of Alexandria who lived in the third century AD. Polynomial equations had been considered by Greek mathematicians before him but in a geometrical manner. For example, to solve equation

$$x^2 = 2 \tag{2}$$

one had to draw a square with the unit side and take its diagonal.

The novelty introduced by Diophantus was as follows: he was seeking solutions of polynomial equations in (positive) rational numbers. For him equation (2) had no solution.

Hilbert in his Tenth Problem asked about solving Diophantine equations in integers. After Diophantus's time mathematicians found solutions of many Diophantine equations, and established that many others had no solutions. This was done by diverse, sometimes *ad hoc*, methods (a well-known example is Fermat's Last Theorem). Hilbert proposed finding a *universal method* applicable to every Diophantine equation.

In today's terminology we would say that Hilbert proposed finding an algorithm. However, he did not use this word — at that time there was no mathematically rigorous general notion of algorithm. It was developed only in the 1930s by Kurt Gödel, Alan Turing, Emil Post, Alonzo Church, and other logicians. They gave several equivalent definitions via different models of computation such as *Turing machines*, *λ-calculus*, *recursive functions*, *etc.*

Davis's interest in Hilbert's Tenth Problem was inspired by his teacher Emil Post who wrote in his paper [1944] that this problem "begs for an unsolvability proof".

Davis quickly devised a conjecture, first announced in Davis [1950b], which would imply the non-existence of the required algorithm. However, the statement of his conjecture did not contain the word "algorithm" either. Dealing with Diophantine equations, it is more convenient to work with another concept closely related to algorithms.

A set $\mathfrak{M}$ of n-tuples of natural numbers is called *listable* (aka *semidecidable*, *recursively enumerable*) if all its elements (and only those) can be printed by some algorithm (computer program working potentially infinitely long). A set $\mathfrak{M}$ of n-tuples of natural numbers is called *decidable*, if there exists an algorithm which for given arbitrary n-tuples of natural numbers tells us after a finite

number of steps whether it belongs to $\mathfrak{M}$ or not. The existence of undecidable listable sets was a major discovery in computability theory.

A set $\mathfrak{M}$ of n-tuples of natural numbers $\langle a_1, \ldots, a_n \rangle$ is called *Diophantine* if there exists a polynomial $P(a_1, \ldots, a_n, x_1, \ldots, x_m)$ with integer coefficients such that

$$\langle a_1, \ldots, a_n \rangle \in \mathfrak{M} \iff \exists x_1, \ldots, x_m [P(a_1, \ldots, a_n, x_1, \ldots, x_m) = 0]. \quad (3)$$

An equivalence of this form is called a *Diophantine representation* of the set $\mathfrak{M}$.

It is easy to see that every Diophantine set is listable. Martin Davis conjectured that the inverse is also true.

**Martin Davis conjecture.** *Every listable set is Diophantine.*

This conjecture implies the negative solution of Hilbert's Tenth Problem. Indeed, it suffices to take an undeciable listable set for $\mathfrak{M}$ in (3); then no algorithm would decide whether the Diophantine equation on the right-hand side of (3) has a solution in $x_1, \ldots, x_m$ for given values of the parameters $a_1, \ldots, a_n$.

Davis's conjecture is much stronger than what would be sufficient for proving the undecidability of Hilbert's Tenth Problem. Namely, it would be enough to allow P in the definition (3) to be any function which becomes a polynomial in $x_1, \ldots, x_m$ after substituting numerical values for $a_1, \ldots, a_n$. As an example of such a generalization, one could allow the parameters (but not the unknowns) to occur in the exponents. Such a weaker form of Davis's approach to Hilbert's Tenth Problem was suggested by Anatolii Mal'tsev [1968] (see also [Matiyasevich, 1993, Comments to Ch. 5]).

Matiyasevich (one of the coauthors of the current paper) once called Davis's conjecture "bold". Indeed, he had few informal arguments supporting it. On the other hand, the conjecture had, in addition to the undecidability of Hilbert's Tenth Problem, many corollaries, some of which were quite striking. For example, the conjecture implies that the (evidently listable) set of all prime numbers is exactly the set of all non-negative values assumed by a certain polynomial for all possible integer values of its variables.

The theory of computability can be developed either in terms of algorithms or in a dual form via listable sets. The second approach was used by Davis's teacher Emil Post [1943] and later by Grigori S. Tseitin [1964] and Per Martin-Löf [1970].

In 1976 Leonard M. Adleman and Kenneth L. Manders [1976] expressed the relationship between the two concepts by introducing the notion of *non-deterministic Diophantine machines*, NDDMs for short. Such a machine is specified by a parametric Diophantine equation and works as follows: on input $\langle a_1, \ldots, a_n \rangle$ it guesses the numbers $x_1, \ldots, x_m$ and calculates the value of the polynomial; if it vanishes, then the n-tuple $\langle a_1, \ldots, a_n \rangle$ is accepted (is recognized as belonging to the set).
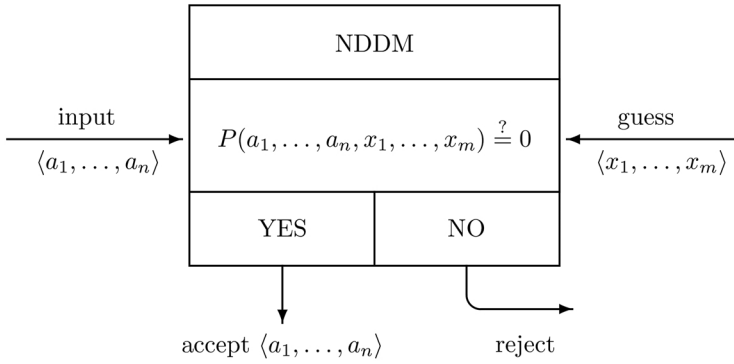
Davis's conjecture can be restated as follows.

Davis's conjecture. *Non-deterministic Diophantine machines are as powerful as, say, Turing machines.*

The rationale behind the introduction of yet another model of computations was as follows. Diophantine machines are essentially non-deterministic, and have explicit separation between the non-deterministic part (guessing the values of $x_1, \ldots, x_m$) and extremely simple deterministic actions (just computing a polynomial). There was a hope that Diophantine machines might be a tool for attacking the notorious Millennium problem $P \stackrel{?}{=} NP$ about deterministic and non-deterministic computations.

Davis provided the initial step towards the proof of his conjecture by first announcing in [Davis, 1950a], proving in his dissertation [Davis, 1950b], and publishing in Davis [1953] the following result:

> **Theorem (Davis).** *For every listable set $\mathfrak{M}$ there exists a polynomial Q with integer coefficients such that*
>
> $$\begin{aligned} &\langle a_1, \ldots, a_n \rangle \in \mathfrak{M} \iff \\ &\exists z \forall y \leq z \exists x_1, \ldots, x_m \big[Q(a_1, \ldots, a_n, x_1, \ldots, x_m, y, z) = 0 \big]. \end{aligned} \quad (4)$$

Representations of this type became known as *Davis normal forms*. They can be considered as an improvement of Kurt Gödel's technique of arithmetization. This technique allows one to represent any listable set by an arithmetical formula containing, possibly many, universal quantifiers. Thus Davis normal form might be viewed as an elimination of all but one universal quantifiers from general arithmetical representations. In fact, the proof in [Davis, 1953][7] is not an

---

[7] A footnote in [Davis, 1953] tells us that this proof was due to an unknown referee of the paper. Presumably, this was Raphael Robinson who soon afterward [Robinson, 1956] quantitatively improved Davis's result by showing that one can always have $m = 4$ in (4).

elimination but a "compression" of arbitrarily many universal quantifiers into a single, bounded one. The original proof given in [Davis, 1950b] and reproduced in [Davis, 1958] was quite different: Martin Davis arithmetized normal systems (introduced by E. Post [1943]) in a very economical way — using just a single universal quantifier.

Davis continued to work towards a proof of his conjecture together with Hilary Putnam. In his recollection [1993a] of this collaboration during the summers 1957–1960 Davis cites a letter from Putnam: "What I remember from that summer is not so much the mathematical details as the sheer *intensity* with which we worked. I have never in my life been so absorbed in a mathematical problem, and I'm sure the same was true of you."

After a number of intermediate results, Davis and Putnam [1958a] "almost proved" the conjecture. The two missing links were as follows.

In the first place, Davis and Putnam were forced to deal with a broader class of *exponential Diophantine representations*. They are similar to (3) but exponentiation is allowed (besides addition and multiplication) for constructing a P.

Second, the proof given by Davis and Putnam was conditional: they assumed that *for every k there exists an arithmetical progression of length k consisting of different prime numbers*.

In 1959 this hypothesis was considered plausible but it was proved only much later, in 2004, by Ben Green and Terence Tao [2008]. Luckily, Davis and Putnam did not have to wait that long for progress. The help came from Julia Robinson. She began to study Diophantine representations at the same time that Davis stated his conjecture. However, the direction of her original research was to show the opposite. Based on a suggestion by her professor Alfred Tarski, Robinson tried to prove that the (evidently listable) set of all powers of 2 is *not* Diophantine. After unsuccessfully attempting to prove this, she switched to searching for Diophantine representations of this set and, more generally, for the set

$$\{\langle a, b, c \rangle | a = b^c\}. \tag{5}$$

Robinson found a sufficient condition (later named JR by Davis) for the existence of such a representation.

Davis and Robinson had met in 1950 at the International Congress of Mathematicians in Cambridge, Massachusetts. Both of them presented their results related to Diophantine representations. Robinson recollects in [Reid, 1996]: "I remember that he [Davis] said he didn't see how my work could help to solve Hilbert's problem, since it was just a series of examples. I said, well, I did what I could." In [1993a] Davis confessed: "It's been said that I told her that I doubted that her approach would get very far, surely one of the more foolish statements I've made in my life."

In 1959 Davis and Putnam sent their new and still unpublished result to Robinson. She was able to get rid of the hypothesis about arithmetical progressions of prime numbers. In retrospect, one can say that Davis, Putnam, and

Robinson were under the magic of Gödel: they were in fact trying to reproduce his technique by Diophantine equations. In [1993a] Davis writes: "Later Yuri Matiyasevich showed that in fact any sufficiently large coprime moduli could be used so that our efforts in connection with prime factors were really unnecessary."

Davis and Putnam withdrew their paper and published the joint paper [1961b] with Robinson. Its main result was as follows.

DPR-theorem. *Every listable set $\mathfrak{M}$ has an exponential Diophantine representation.*

After that, in order to prove Davis's conjecture about *arbitrary* listable sets, it was sufficient to find a Diophantine representation for a *single* special set (5). In turn, for this it was sufficient to fulfill the above-described condition JR (found by Robinson), and Davis tried to do it. Davis's paper [1968] has the intriguing title *One equation to rule them all*. In it, he proved that if a particular equation (meant in the title) has only a trivial solution, then JR is fulfilled. Robinson's attitude to this result can by seen from the following fragment of her letter to Davis [1993a]:

> I have enjoyed studying it, but my faith in JR still hasn't been restored. However, for the first time, I can see how it might be proved. Indeed, maybe your equation works, but it seems to need an infinite amount of good luck!

Unfortunately, several authors later found non-trivial solutions of that equation. However this did not invalidate Davis's idea completely. It would be sufficient to establish that the set of solutions is finite. Moreover, this turned out to have some other interesting consequences.

The first example of a Diophantine equation satisfying JR was constructed by Matiyasevich [1970] (see also [Davis, 1973a]). In February 1970, Davis got from Robinson handwritten notes that had been taken by John McCarthy at a lecture about this example given by Grigory Tseitin in Novosibirsk. Davis [1993a] recollects: "I was able to have the great pleasure of reconstructing the proof. But I was not satisfied until I had produced my own variant of Dr. Matiyasevich's proof and had presented it (on March 10) at a seminar at Rockefeller University at Hao Wang's invitation." This proof was published in [Davis, 1971].

Having been proved, Davis's conjecture is now known under two abbreviations:

DPRM-theorem ($\equiv$ MRDP-theorem). *Every listable set has a Diophantine representation.*

Thus, two notions taking their origin in very different parts of science (Diophantine equations in Number Theory and listable sets in Computer Science) determine the same class of sets of natural numbers. This implies that the concept of computability can be defined in purely mathematical terms.

In [1972] Davis generalized the negative solution of Hilbert's Tenth Problem in the following way: *There is no algorithm for deciding whether the cardinality of the set of solutions of a given Diophantine equation belongs to a particular set unless this set or its complement is empty.*

Davis [1973b] transferred the so-called *speed-up theorem* of Manuel Blum [1967] to Diophantine equations.

Davis made the solution of Hilbert's Tenth Problem available to a broad mathematical readership in his paper [1973a] in *The American Mathematical Monthly*. A full proof can also be found in the book [Matiyasevich, 1993]; Davis was the editor of the English translation and wrote a *Foreword* to it. Recently another book, [2019], with full detailed proof was published by M. Ram Murty and Brandon Fodden.

The DPRM-theorem found numerous applications. Some of them are presented in the joint paper [Davis *et al.*, 1976] written by Davis, Robinson, and Matiyasevich. For many other applications of the DPRM-theorem consult the bibliography in [Matiyasevich, 1993] (which was extended in the Greek translation) and in [Murty and Fodden, 2019].

The DPRM-theorem was not the final point in the investigation of Hilbert's Tenth Problem. On the contrary, the theorem opened a way to study two natural generalizations of the problem:

- Q1 *Is there an algorithm for solving Diophantine equations in the field* $\mathbb{Q}$ *of rational numbers?*
- Q2 *Is there an algorithm for solving Diophantine equations in algebraic integers from a given finite extension of* $\mathbb{Q}$?

Positive solution of Hilbert's Tenth Problem would imply affirmative answers to both questions, but the negative solution has no straightforward implications for these questions.

After numerous partial results about Q2 (for particular fields or classes of fields) by many authors the negative answer was recently announced by Peter Koymans and Carlo Pagano [2024] and by Levent Alpöge, Manjul Bhargava, Wei Ho, and Ari Shnidman [2025].

As for Q1, this question remains open in 2025. Davis [2010a] proposed a fresh approach to this problem by putting forth a new conjecture. It exploits not the mere existence of undecidable listable sets but the existence of so called *simple sets*. The latter had been introduced by Emil Post [1944].

4. AUTOMATED REASONING

We shall be concerned with the problem of testing formulas in conjunctive normal form for consistency; or, dually, with the problem of testing formulas in disjunctive normal form for validity. Martin Davis and Hilary Putnam [1958b]

Davis was also a pioneer in a field that is called today 'computational logic'. He entered this field after a quick exposure to some of the earliest computers. This journey into concrete computer programming began in 1951,

when he joined a group involved in developing programs for an ORDVAC machine, supporting military efforts during the Korean War. He transitioned to the Control Systems Laboratory under Frederick Seitz after leaving his position as a Research Instructor at the University of Illinois at Urbana-Champaign. This move, coupled with a subsequent two-year Office of Naval Research grant at the Institute for Advanced Study (IAS) in Princeton, enabled him to sidestep conscription.

Following a year-long period of intensive programming training, Davis gained sufficient confidence in his computing abilities to secure funding for a project focused on implementing a decision procedure for Presburger's integer arithmetic of addition. The procedure was implemented at the IAS, on a JOHNNIAC machine, during the summer of 1954. "Its great triumph was to prove that the sum of two even numbers is even" [Davis, 1983]. However modest that achievement may seem today, it is often considered as the first computer-generated mathematical proof (*cf.* [Siekmann and Wrightson, 1983]), ushering in the automated-proof era. Briefly, to characterize the difference in outlook between his and another mid-fifties achievement in automating deduction, namely, the Logic Theorist by Allen Newell, Herbert A. Simon (the Nobel laureate), and Cliff Shaw [1983] Davis phrases the respective slogans as "Use mathematical logic" and "Simulate people".

Over the years, Davis made occasional, yet significant, contributions to automated reasoning. He also coined some of the terminology that has become standard in that field.[8]

The report [1958b], written jointly by Martin Davis and Hilary Putnam in 1958, evolved into articles ([Davis and Putnam, 1960], [Chinlund *et al.*, 1964], *etc.*) that furthered its broad influence on later research. That report arose from a project that Davis and Putnam, initially willing to tackle proof automation in an all-encompassing framework (undecidable, albeit semi-decidable[9]), had proposed to the National Security Agency. "When we talked about our proposal — Davis recalls in [2020] — , they made it clear that they had no interest in proof procedures for first order logic, but they were interested in SAT.[10] They warned me that it is a difficult problem and doubted that we could make much headway in one summer. However if we were willing to work only on

---

[8] See, in particular, [Davis, 1963]: in [Davis, 1983; Davis and Fechter, 1991] Davis admitted that there he had termed "the Herbrand Universe" a domain — the collection of all closed terms built over a non-empty set of constants and a set of function symbols — which, he says, should more correctly be called "the Skolem Universe".

[9] Given a formal statement, a decision algorithm should tell whether or not it is provable as a theorem; a semi-decision procedure may, when confronted with a false or unprovable statement, run forever.

[10] Authors' note. SAT (short for 'satisfiability') is the problem of whether there is a way to assign truth values to the variables in a formula of propositional logic that makes the formula evaluate to 'true'. The momentum – and specific algorithmic contribution – which Davis and Putnam gave to satisfiability testers for propositional formulas will emerge from the ongoing. Our discourse will soon shift to a more general notion of satisfiability, which refers to formulas of first-order logic.

SAT, they were prepared to fund our proposal." As an outcome, the afore-cited [Davis and Putnam, 1958b] deliverable approaches propositional calculus from a broad angle. Its first part discusses the advantage of bringing formulas to some standard format; it notes that not all such forms have the same properties and argues that conjunctions of disjunctions of propositional variables and negations of such (CNF formulas, for short) are a convenient format to be treated by satisfiability testing methods. Embedding a CNF-tester in a proof procedure for full quantification theory later enabled the approach by Davis and Putnam to outperform competitors of the time (see [Davis and Putnam, 1960]), particularly the theorem-proving programs developed by Paul C. Gilmore [1960], Hao Wang [1960], and Dag Prawitz *et al.* [1960].

Two CNF-satisfiability testers (see [Davis and Putnam, 1960; Davis *et al.*, 1962]), whose acronyms are DP and DPLL after the names Davis–Putnam–Logemann–Loveland, evolved from the above-mentioned seminal study [Davis and Putnam, 1958b]. Interestingly, the DPLL decision procedure remains at the core of fast Boolean satisfiability solvers decades later (see [Loveland *et al.*, 2016] and [Biere *et al.*, 2021]). Furthermore, the problem of recognizing CNF-satisfiability has become the standard for representing the difficulty of solving by a feasible algorithm a class of hard combinatorial problems — a millennium-prize challenge.[11]

To describe the functioning of DPLL briefly, let us assume that the CNF formula to be analyzed is input as a set $\mathcal{S}_0$ whose members are sets of nonnull integers: positive integers encode propositional variables and negative integers represent their opposites. The set $\mathcal{S}_0$, as well as its members, henceforth named *clauses*, are finite; the members of the clauses are called *literals*. Clauses represent the disjunctions that form the given formula. As extreme cases, the empty set of clauses represents the truth value 'true' whereas any set to which the empty clause belongs represents 'false'.

Simplifying a clause set $\mathcal{S}$ with respect to some number ℓ means removing all clauses to which ℓ belongs from $\mathcal{S}$, while removing its opposite, $-\ell$, from the remaining clauses. Performing such a simplification means assigning the value 'true' to the literal represented by ℓ; thus, all clauses to which ℓ belongs become true, while $-\ell$ can no longer help any other clause to become true.

DPLL starts by putting $\mathcal{S} = \mathcal{S}_0$ and repeatedly simplifies $\mathcal{S}$ until it becomes empty, in which case the initial $\mathcal{S}_0$ is declared satisfiable and the algorithm terminates. Each simplification step extends a truth-value assignment intended to make $\mathcal{S}_0$ true: When simplifying w.r.t. a literal ℓ, it is choosing to make ℓ true and prevents $-\ell$ from being chosen later. Therefore, DPLL does not just establish satisfiability but also produces a model for $\mathcal{S}_0$ — if any.

The case when $\mathcal{S}$ becomes false (because one of its clauses empties) is asymmetric: the algorithm backtracks to a *choice point* — specifically, a situation where the literal ℓ that triggered a simplification was preferred over its

---

[11] See https://www.claymath.org/millennium/p-vs-np/.

unattempted opposite, $-\ell$. At this point, the alternative simplification is triggered. When no choice points remain and $\mathcal{S}$ has not emptied, $\mathcal{S}_0$ is declared unsatisfiable.

Let us go into more detail. DPLL orchestrates the following three rules, giving priority to the first two over the third.[12]

- *Unit rule:* After finding a singleton clause $\{\ell\}$ in $\mathcal{S}$, simplify $\mathcal{S}$ w.r.t. ℓ.
- *Pure-literal rule:* After finding some literal ℓ belonging to a clause of $\mathcal{S}$ such that $-\ell$ does not belong to any of its clauses, simplify $\mathcal{S}$ w.r.t. ℓ.
- *Splitting rule:* Pick a literal ℓ such that both ℓ and $-\ell$ belong to clauses of $\mathcal{S}$; set a choice point here. Tentatively simplify $\mathcal{S}$ with respect to ℓ and continue. If subsequent simplifications do not lead to satisfying $\mathcal{S}$, undo the previous simplification and, instead, simplify $\mathcal{S}$ w.r.t. $-\ell$.

It deserves being mentioned that the original DP algorithm had, in place of the splitting rule, a rule based on the following remark (which somehow anticipates the Resolution principle — see below):

Rule for Eliminating Atomic Formulas. Let the given formula F be put into the form $(A \lor p) \,\&\, (B \lor \bar{p}) \,\&\, R$, where $A, B,$ and R are free of p. (This can be done simply by grouping together the clauses containing p and 'factoring out' occurrences of p to obtain A, grouping the clauses containing $\bar{p}$ and 'factoring out' $\bar{p}$ to obtain B, and grouping the remaining clauses to obtain R.) Then F is inconsistent if and only if $(A \lor B) \,\&\, R$ is inconsistent. [Davis and Putnam, 1960].

It is reported in [Davis *et al.*, 1961a] that after programming DP using this rule, it was decided to replace it by the splitting rule: the two are interchangeable, but the former proved "prohibitive in a computer if one's fast access storage is limited".

With regard to first-order logic, Davis devised a method named *linked-conjunct* to generate a proof of a sentence automatically by seeking a counterexample to its negation. That method was implemented in a LISP-based theorem prover at Bell Labs — see [Davis, 1963; Chinlund *et al.*, 1964]. It borrowed from the above-mentioned Prawitz prover the idea of 'mating' — through a suitable algorithm based on the so-called *unification*[13] — instances

---

[12] Additional heuristics were present in the original implementation.

[13] A unification algorithm's task is to find a substitution of individual variables with terms that makes two given terms or atomic formulas syntactically identical. If no such substitution exists, the algorithm simply fails; otherwise, it produces one of maximum generality. Mechanisms for this purpose were independently devised around 1960 by various authors (including Davis's team at Bell Labs), unaware that Jacques Herbrand had already sketched a unification algorithm in his Ph.D. thesis in 1930. As pointed out in Davis's introductory paper [Davis, 1994] to *The Collected Works of Post*, Post also had a unification algorithm but it was unpublished.

of atomic formulas bearing complementary signs in different conjuncts, *i.e.*, disjunctive clauses. Even though the linked-conjunct method became less popular after the more influential Resolution principle by John Alan Robinson gained prominence, some improved variants of the latter can be best explained from the viewpoint of the former method (*cf.* [Yarmush, 1976; Omodeo, 1982]), as we will now highlight.

Automatic theorem-proving procedures usually presuppose that the axioms of a theory together with the negation of a theorem's claim are converted into an *unsatisfiable* conjunctive form;[14] their main task is to find a *refutation* of the set of (disjunctive) clauses representing this conjunctive form, where no quantifier appears, but every individual variable is tacitly regarded as universally quantified. There are several notions of refutation to choose from. For example, the individual steps that form a refutation may generate clauses or conjunctions thereof; correspondingly, the final step of a refutation leads to the empty clause or to a *truth-functionally* unsatisfiable set of clauses. In the latter case, one may insist — as Davis did, with no loss of generality — that every literal occurs both affirmed and negated within the final step of a refutation (this is the 'linkedness' property referred to in the name of Davis's method).

When the steps of a refutation generate clauses (as is the case with strictly resolution-based methods), each application of an inference rule removes one or more occurrences of literals from a set of unifiable occurrences. The loss of information due to this deletion is one of the causes of the creation of an intractable number of useless clauses during the search for a refutation. This 'combinatorial explosion' turns out to be the main drawback of resolution-based methods for automatic theorem proving.

In the linked-conjunct method, on the other hand, the detection of a pair of complementarily unifiable literals does not lead to literal deletion, but rather to the 'mating' of two literals with one another. Thus, in a more precise description than the one given above, a single step of a refutation manipulates a set of clauses with a superimposed mating relationship holding between pairs of occurrences of literals (not to mention a binding of terms to individual variables). As a support for such manipulations, graph-like structures can conveniently represent the set of clauses. Such graphs, which the linked-conjunct method handles explicitly, serve mainly as a conceptual tool for demonstrating the completeness of sophisticated refinements of Robinson's resolution method.

Over the decades, there has been a noticeable shift in expectations regarding proof-assistant systems. These systems have evolved from stand-alone theorem provers into proof-checking frameworks that range from highly interactive reasoning assistants to simple proof-script verifiers. Davis played a role in catalyzing this shift in perspective.

---

[14] We use the term 'conjunctive form' instead of 'CNF' in the present context to emphasize that we are referring to first-order logic. Here, within formulas, 'atoms' of the form $R(t_0, t_1, \ldots, t_n)$ — where R stands for an $(n+1)$-argument predicate symbol and each t_i represents a term — play a role analogous to the true-false variables of propositional logic.

In [1979], Davis and Jacob T. Schwartz advocated for incorporating *meta-mathematical extensibility* mechanisms into mature proof-verification technology, enabling proof-checkers to undergo substantial enhancements without compromising their reliability.

Around 1980, Davis spent time at John McCarthy's Stanford Artificial Intelligence Laboratory. In the lab, he also worked with the FOL ('First Order Logic') proof checker developed by Richard Weyhrauch. Referring to that experience, he said that while he enjoyed the quality of interaction with the automated assistant, he also felt irritated by the need to make many painstaking tiny steps to justify inferences that were quite obvious. As a result, by reusing the LISP source code for the linked-conjunct theorem prover, a Stanford undergraduate successfully implemented an 'obvious' facility as an add-on to FOL, see Davis [1981]. In addition, [Davis, 1980] was also written during his time in what Davis found to be a very stimulating environment.

5. NONSTANDARD ANALYSIS

Perhaps indeed, enthusiasm for nonstandard methods is not unrelated to the well-known pleasures of the illicit. Martin Davis [1977a]

Davis recalled in [2016a]:

> As an undergraduate I had tried briefly to rehabilitate Leibniz's use of infinitesimal quantities as a foundation for calculus [. . .] It was therefore with great excitement and pleasure that I heard Abraham Robinson's address before the Association for Symbolic Logic [...] in which he provided an elegant solution to this problem using techniques that he dubbed *nonstandard analysis*.

After Davis and Reuben Hersh wrote, on this subject, the popular article [1972] for *Scientific American* magazine, Davis published the monograph [1977a] which appeared in print at the same time as two other major books on infinitesimal calculus (*cf.* [Blass, 1978]).

In the introduction of [Davis, 1977a], he stated:

> The subject can only be claimed to be of importance insofar as it leads to simpler, more accessible expositions, or (more important) to mathematical discoveries. As to the first, the reader must be the judge. The best evidence for the second is the Bernstein–Robinson theory [. . .], which settled a question that had remained open for many years.

The book in fact culminates in the proofs, by nonstandard methods, of important theorems about Hilbert spaces, including the Bernstein–Robinson theorem.

The way in which Davis conceived the initial part of this monograph offers a brilliant example of information hiding as a guiding principle for the design of

widely applicable constructions and methods of proof. As emphasized in [Cantone *et al.*, 2016], the concept of withholding details, as stated here, is treated in another 1977 writing: the previously cited *Metamathematical extensibility for theorem provers and proof-checkers* [Davis and Schwartz, 1979], which was more related to proof technology than to the foundations of analysis.

The first chapter of [Davis, 1977a] concentrates, in fact, on how to enlarge a standard 'universe' (this word has a specific technical sense) into a nonstandard one. While taking stock at the end of this chapter, Davis remarks that much of the machinery developed up to that point will play no role in the rest of the book; then, in recapitulating which key issues the reader should bear in mind, he stresses three main tools of nonstandard analysis: *transfer principle*, *concurrence*, and *internality*. Concerning the elaborate construction in which he had focused on for over forty pages — the so-called *ultra-power* construction — , he concludes: "The reader who remembers these key points will do well in what follows. In particular, it is now quite all right to entirely forget how the nonstandard universe was defined and to banish ultrafilters from our consciousness".

6.   HISTORY AND PHILOSOPHY OF COMPUTING

Davis not only lived through a large part of the early history of computing and computer science, having studied with Post and Church, he also cared deeply about preserving that history and making that history accessible to students with different levels of mathematical maturity. The volume *The Undecidable*, mentioned above, contained two unpublished papers, namely, Gödel's Princeton Lectures "On undecidable propositions of formal mathematical systems" and Post's "Account of an anticipation", as well as fifteen other published, pioneering papers in computability theory by Alonzo Church, Stephen C. Kleene, J. Barkley Rosser, and Alan M. Turing, together with other papers by Gödel and Post. Each of the seventeen texts was accompanied by a short introductory note, contextualizing it in the landscape of early computability theory. These introductory notes succeed in illuminating this intellectual landscape for a broad audience. As mentioned above, Davis later edited Post's *Collected Works* for which he wrote an extended biographical introduction — a testimony to Davis's admiration and respect for his teacher.

Davis's contributions to the history of computing were not restricted to "just" editing important documents but included writing scholarly papers on topics that reflected his deep interest in logical foundations and the emergence of computer science. The resulting sequence of papers includes, among others, "Why Gödel didn't have Church's thesis" [Davis, 1982]. At the core of this study are basic questions related to the early history of the so-called Church–Turing thesis.[15] As Davis explains, it was prompted by Kleene's work [1981] on the origins of recursive functions and the more philosophical work [1980] about

---

[15] We say "so-called" since, historically, there is no such thing as the Church–Turing thesis. Cp. Note 5. In his [1952], Kleene had devoted two sections for the separate insightful discussion of Church's, respectively Turing's, thesis.

the consequences of the "Church–Turing thesis" for issues related to mechanism and mentalism. Using historical documents and personal letters, Davis explored why Gödel was hesitant to support a thesis similar to Church's, while Church was "willing to go ahead".[16] Moreover, Davis is careful to distinguish between Church's, Gödel's, Kleene's, Post's, and Turing's theses and provides a much needed historically and conceptually diversified view on the so-called "Church–Turing thesis". By reconstructing the formulation of these different theses, Davis also provides an account of how Post anticipated the results of Church, Gödel, and Turing in the early 1920s.[17] This paper remains a classic for anyone working on the early history of the "Church–Turing thesis".

While Davis worked on the history of "pure" computability theory, he also wrote extensively on the significance of logic for the emergence of computer science and the modern computer. The earliest publication on those topics is his paper "Mathematical logic and the origin of modern computers" [1991]. Perhaps best-known is his book *The Universal Computer: The Road from Leibniz to Turing* [2000]. A new version was published in 2012, the Turing centenary. There, Davis inscribes the history of the modern computer in a much longer history of logic and mathematics going back to Leibniz's notions of a *characteristica universalis* and a *calculus ratiocinator*. He develops from there a narrative of the history of logic and mathematics that leads to the work of Gödel and Turing and shows how logic and computation are "*two sides of the same coin*" [Davis, 2000, p. 200]. He then argues that this historical path from Leibniz to Turing laid the foundation for the modern computer. The topic of the influence of logic on computing, in particular the significance of Turing's work in this context, has been a difficult and indeed controversial one amongst computer scientists and historians alike; Davis's 2012 edition too got caught up in that debate.[18]

But his work should also be seen in its proper context: at the time Davis started to write about these topics, it was one of the few historical works available on the topic. Davis's work here was heavily influenced by his own experience as a logician who became involved with programming early on, and whose work on automatic theorem proving was later seen as fundamental in

---

[16] Sieg [1997], having access to additional documents, showed that Church was initially also quite hesitant.

[17] The reader might wish to compare this paper with Gandy's "The confluence of ideas in 1936" [1988]. Davis and Sieg were alluding to the title of that paper when emphasizing in their [2015] the remarkable conceptual confluence of Post's and Turing's work.

[18] The significance of Turing's 1936 paper for the actual design and construction of the first computers is debatable; there is mostly consensus among historians that Turing is, at best, one of many who contributed to that development. As was argued in [De Mol *et al.*, 2018], the introduction of the Turing-machine concept as a model for modern computers only came in the early 1950s. — The close ties between logic and computer science were loosened by emphasizing other important contributions in [Mahoney, 1988]. Davis's depiction of those ties was criticized, for example, in [Daylight, 2014].

computer science. This work provides an exemplar of the perspective of a computer scientist who was trained as a logician and discovered that his work could be recast as belonging to a subject that did not exist at the time of his training.

We already argued, in Sec. 2, that Davis was strongly convinced of the correctness of the Turing–Post analysis of computability. In fact, Davis was explicit that he was a mechanist and so convinced that human brains cannot surpass incompleteness [Jackson, 2008; Davis, 2016b]. This methodological and philosophical stance made him an outspoken critic of "hypercomputability" as formulated, among others, by Copeland [2000] and Siegelmann [1999]. They considered versions of the Church–Turing thesis as false, because they constructed some "physically realizable" models that can "compute" more than Turing machines can. Davis criticized these works on multiple occasions through conceptual and historical arguments but avoided dogmatism.[19] Amongst others, he argued that such a "result" can be obtained only when finiteness conditions are violated that are essential in Turing's analysis. Being clear about this restriction, Davis is open to the possibility of an expanded, different notion of "computability". In one of his pieces on hypercomputation he writes: "Can we really utterly exclude the possibility of some new development leading to a physical realization of an uncomputable quantity? Of course not. [. . .]" [2004, p. 206]. When questions such as "Is the mind a computer?" or "Is human insight algorithmic?" were raised, Davis took a similarly open position. For instance, Roger Penrose argued in two books [Penrose, 1989; 1994] that Gödel's incompleteness theorem implies that the human mind cannot be a computer. Davis rejected Penrose's claims with clear technical arguments. He did this in two short papers, [Davis, 1990; 1993b]. In the first, he concludes: "There is certainly room for disagreement about whether the processes by which mathematical (or physical) theories are developed and accepted are algorithmic. But Gödel's theorem has nothing decisive to contribute to the discussion." These papers reveal Davis as a writer who could not stand what he considered to be false or badly argued claims but also as a thinker who is careful when speaking about questions that cannot be solved by proof alone. In such contexts, he often referred to an empiricist or inductive position. That position is elaborated in what is perhaps Davis's most explicitly philosophical paper. There it becomes clear how Davis's anti-dogmatism is intertwined with this more empirical view on mathematical practice, for instance, when he writes [2016b]:

> We explore simple austere worlds that differ from the one we inhabit both by their stark simplicity and by their openness to the infinite. It is simply an empirical fact that we are able to obtain apparently reliable and objective information about such worlds. And, because of this, any illusion that this knowledge is certain must be abandoned. [. . .] If presented with a proof that Peano Arithmetic is inconsistent or even that some huge

---

[19] See for instance [Davis, 2004; 2006]

natural number is not the sum of four squares, I would be very very skeptical. But I will not say that I know that such a proof must be wrong.

Davis's anti-dogmatic empiricism is also what drove his more "unconventional" position with respect to the P vs NP problem. He repeatedly stated that, from his view, there is no *convincing* heuristic evidence that $P \neq NP$.[20]

The *The Universal Computer* treats the history of computability theory no longer narrowly as being equivalent to that of recursion theory. On the contrary, the subject is connected to penetrating questions and claims of Leibniz, to foundational problems in the nineteenth and early twentieth century mathematics, as well as to contemporary issues pertaining to computer technology, cognitive psychology, and philosophy of mind. Davis's goal was not only to provide a clear, non-technical exposition of the development of computability theory and its central results, but also to describe their broad impact on contemporary thought. More importantly, perhaps, Davis wanted to bring out how logic, deeply embedded in philosophy, articulated foundational problems and addressed them from a genuinely revolutionary perspective. At the end of the book's Introduction, Davis wrote: "Nowadays, as computer technology advances with such breathtaking rapidity, as we admire the truly remarkable accomplishments of the engineers, it is all too easy to overlook the logicians whose ideas made it all possible. This book tells their story." For us it is clear that Davis became through his work an integral part of "their story" — deeply rooted in the intersection of philosophy, mathematics, and computer science.

REFERENCES

Adleman, L., and K. Manders [1976]: 'Diophantine complexity', in *17th Annual IEEE Symposium on Foundations of Computer Science*, pp. 81–88. New York: IEEE. DOI: https://doi.org/10.1109/SFCS.1976.13.

Alpöge, L., M. Bhargava, W. Ho, and A. Shnidman [2025]: 'Rank stability in quadratic extensions and Hilbert's tenth problem for the ring of integers of a number field'. DOI: https://doi.org/10.48550/arXiv.2501.18774.

A., Biere, M. Heule, H. van Maaren, and T. Walsh, eds [2021]: *Handbook of Satisfiability*. Frontiers in Artificial Intelligence and Applications; 336. 2nd ed. Amsterdam: IOS Press. DOI: https://doi.org/10.3233/FAIA336.

Blass, A. [1978]: Review of Martin Davis, *Applied Nonstandard Analysis*, K.D. Stroyan and W.A.J. Luxemburg, *Introduction to the Theory of Infinitesimals*, and H. Jerome Keisler, *Foundations of Infinitesimal Calculus, Bull. AMS* **84**, 34–41. Available at https://projecteuclid.org/euclid.bams/1183540371. Accessed June 2025.

Blum, M. [1967]: 'A machine-independent theory of the complexity of recursive functions', *Journal ACM* **14**, 322–336.
DOI: https://doi.org/10.1145/321386.321395.

Calvert, W., V. Harizanov, E.G. Omodeo, A. Policriti, and A. Shlapentokh [2024]: 'In memory of Martin Davis', *Notices AMS* **71**. (A preprint with an extensive

---

[20] See for instance [Jackson, 2008].

bibliography is available at https://arxiv.org/pdf/2401.10154.)
DOI: https://doi.org/10.1090/noti2982.

Cantone, D., E.G. Omodeo, and A. Policriti [2016]: 'Banishing ultrafilters from our consciousness', in [Omodeo and Policriti, 2016], pp. 255–283.
DOI: https://doi.org/10.1007/978-3-319-41842-1_10.

Chinlund, T.J., M. Davis, P.G. Hinman, and M.D. McIlroy [1964]: 'Theorem-proving by matching', Technical report, Murray Hill, N.J.: Bell Telephone Laboratories, Inc.

Copeland, B.J. [2000]: 'Narrow versus wide mechanism: Including a re-examination of Turing's views on the mind-machine issue', *Journal of Philosophy* **97**, 5–32. Available at https://www.jstor.org/stable/2678472. Accessed June 2025.

Davis, M. [1950a]: 'Arithmetical problems and recursively enumerable predicates' (abstract), *J. Symbolic Logic* **15**, 77–78.

———— [1950b]: *On the Theory of Recursive Unsolvability.* PhD thesis, Princeton University.

———— [1952]: 'Relatively recursive functions and the extended Kleene hierarchy' (abstract), in *Proceedings of the International Congress of Mathematicians, (Harvard University, Cambridge, MA, August 30–September 6, 1950)*, Vol. 1, p. 723. Providence, R.I.: AMS.

———— [1953]: 'Arithmetical problems and recursively enumerable predicates', *J. Symbolic Logic* **18**, 33–41. Russian transl. in [Matematika, 1964], pp. 15–22.

———— [1958]: *Computability and Unsolvability.* New York: McGraw-Hill. Reprinted with an additional appendix, Dover, 1983. Italian translation in [Davis, 1975]. Japanese translation in [Davis, 1966].

———— [1963]: 'Eliminating the irrelevant from mechanical proofs', *Proc. Symp. Appl. Math.* **15**, 15–30. (Reprinted in [Siekmann and Wrightson, 1983], pp. 315–330; Russian transl. in [Kib, 1970], pp. 160–179.

————, ed. [1965]: *The Undecidable: Basic Papers on Undecidable Propositions, Unsolvable Problems and Computable Functions.Hewlett*, N.Y.: Raven Press.

———— [1966]: *Theory of Computation.* Iwanami, Tokio: Modern Science Selection. Japanese translation of [Davis, 1958]. Shigeru Watanabe, trans.

———— [1968]: 'One equation to rule them all', *Transactions of the New York Academy of Sciences* (2) **30**, 766–773.

———— [1971]: 'An explicit Diophantine definition of the exponential function', *Comm. Pure Appl. Math.* **24**, 137–145.

———— [1972]: 'On the number of solutions of Diophantine equations', *Proc. AMS* **35**, 552–554.

———— [1973a]: 'Hilbert's tenth problem is unsolvable', *Amer. Math. Monthly* **80**, 233–269. Reprinted with corrections as Appendix 2 of the Dover edition of *Computability and Unsolvability* [Davis, 1958], pp. 199–235 (1983 ed.).

———— [1973b]: 'Speed-up theorems and Diophantine equations', in R. Rustin, ed., *Courant Computer Science Symposium 7: Computational Complexity*, pp. 87–95. New York: Algorithmics Press.

———— [1975]: *Computabilità e Insolubilità Introduzione alla teoria della computabilità e alla teoria delle funzioni ricorsive.* Rome: Collana di Epistemologia diretta da Evandro Agazzi. Edizioni Abete. Italian translation of [Davis, 1958]; foreword by Mariano Bianca, ed.

———— [1977a]: *Applied Nonstandard Analysis.* John Wiley & Sons, Inc. Reprinted with corrections, Dover, 2005. Russian translation. Moscow: Mir, 1980. Japanese translation, 1977.

_____ [1977b]: 'Unsolvable problems', in J. Barwise, ed., *Handbook of Mathematical Logic*, pp. 567–594. Amsterdam: North-Holland.

_____ [1980]: 'The mathematics of non-monotonic reasoning', *Artif. Intell.* **13**, 73–80.

_____ [1981]: 'Obvious logical inferences', in *Proceedings of the 7th International Joint Conference on Artificial Intelligence*. Vol. **1**, pp. 530–531. San Francisco: Morgan Kaufmann Publishers Inc.

_____ [1982]: 'Why Gödel didn't have Church's thesis', *Information and Control* **54**, 3–24.

_____ [1983]: 'The prehistory and early history of Automated Deduction', in [Siekmann and Wrightson, 1983], pp. 1–28.

_____ [1990]: 'Is mathematical insight algorithmic?', *Behavioral and Brain Sciences* **13**, 659–660. DOI: https://doi.org/10.1017/S0140525X00031915.

_____ [1991]: 'Mathematical logic and the origin of modern computers', in R. Herken, ed., *The Universal Turing Machine: A Half-Century Survey*, pp. 149–175. Oxford University Press. DOI: https://doi.org/10.1093/oso/9780198537748.003.0005.

_____ [1993a]: Foreword to the English translation of [Matiyasevich, 1993], pp. xiii–xvii. Cambridge, Mass.: MIT Press. Reproduced with small modifications under the title 'The collaboration in the United States', in [Reid, 1996], pp. 91–97, and in its Russian translation [Rid, 2023], pp. 109–115.

_____ [1993b]: 'How subtle is Gödel's theorem? More on Roger Penrose', *Behavioral and Brain Sciences* **16**, 611–612.
DOI: https://doi.org/10.1017/S0140525X00031915.

_____ [1994]: 'Emil L. Post: His life and work', in M. Davis, ed., *Solvability, Provability, Definability: The Collected Works of Emil L. Post*, pp. xi–xxviii. Birkhäuser.

_____ [1999]: 'From logic to computer science and back', in C.S. Calude, ed., *People and Ideas in Theoretical Computer Science: Discrete Mathematics and Theoretical Computer Science*, pp. 53–85. Springer.

_____ [2000]: *The Universal Computer: The Road from Leibniz to Turing*. First edition, New York: Norton. Turing Centenary Edition, CRC Press, 2012. Third edition, 2018, DOI: https://doi.org/10.1201/9781315144726.

_____ [2004]: 'The myth of hypercomputation', in C. Teuscher, ed., *Alan Turing: Life and Legacy of a Great Thinker*, pp. 195–211. Springer.
DOI: https://doi.org/10.1007/978-3-662-05642-4_8.

_____ [2006]: 'Why there is no such discipline as hypercomputation', *Appl. Math. Comput.* **178**, 4–7. (Special issue on Hypercomputation, F.A. Doria and J.F. Costa, eds). DOI: https://doi.org/10.1016/j.amc.2005.09.066.

_____ [2010a]: 'Representation theorems for r.e. sets and a conjecture related to Poonen's large subring of $\mathbb{Q}$'. *Zapiski Nauchnykh Seminarov Peterburgskogo Otdeleniya Matematicheskogo Instituta im. V.A. Steklova RAN (POMI)* **377**, 50–54. Reproduced as [Davis, 2010b].

_____ [2010b]: 'Representation theorems for recursively enumerable sets and a conjecture related to Poonen's large subring of $\mathbb{Q}$', *J. Math. Sci. (N.Y.)* **171**, 728–730. Reproduction of [Davis, 2010a].

_____ [2016a]: 'My life as a logician', in [Omodeo and Policriti, 2016], pp. 1–33. (Revised and expanded version of [Davis, 1999]).
DOI: https://doi.org/10.1007/978-3-319-41842-1_1.

_____ [2016b]: 'Pragmatic Platonism', in [Omodeo and Policriti, 2016], pp. 349–356. (Originally published at: https://foundationaladventures.files.wordpress.com/2012/01/platonic.pdf.) DOI: https://doi.org/10.1007/978-3-319-41842-1_14.

_____ [2020]: 'Seventy years of computer science', in A. Blass, P. Cégielski, N. Dershowitz, M. Droste, and B. Finkbeiner, eds, *Fields of Logic and Computation III — Essays Dedicated to Yuri Gurevich on the Occasion of His 80th Birthday*, pp. 105–117. Lecture Notes in Computer Science; 12180. Springer.
DOI: https://doi.org/10.1007/978-3-030-48006-6_8.

Davis, M., and R. Fechter [1991]: 'A free variable version of the first-order predicate calculus', *J. Logic Comput.* **1**, 431–451.

Davis, M., and R. Hersh [1972]: 'Nonstandard analysis', *Scientific American* **226**, 78–86.

Davis, M., and H. Putnam [1958a]: 'Reductions of Hilbert's tenth problem', *J. Symbolic Logic* **23**, 183–187. Russian transl. in [Matematika, 1964], pp. 49–53.

_____ [1958b]: 'Feasible computational methods in the propositional calculus', Technical report. Troy, N.Y.: Rensselaer Polytechnic Institute, Research Division. Reprinted in [Omodeo and Policriti, 2016], pp. 373–408. Available at https://link.springer.com/content/pdf/bbm:978-3-319-41842-1/1. Accessed July, 2025.

_____ [1959]: 'A computational proof procedure; Axioms for number theory; Research on Hilbert's Tenth Problem', Technical Report AFOSR TR59-124. U.S. Air Force. Reprinted in [Omodeo and Policriti, 2016], pp. 411–430. Available at https://link.springer.com/content/pdf/bbm:978-3-319-41842-1/1. Accessed July, 2025.

_____ [1960]: 'A computing procedure for quantification theory', *Journal ACM* **7**, 201–215. Preprinted as [Davis and Putnam, 1959], Part I; reprinted in [Siekmann and Wrightson, 1983], pp. 125–139.

Davis, M., and J.T. Schwartz [1979]: 'Metamathematical extensibility for theorem verifiers and proof-checkers', *Comput. Math. Appl.* **5**, 217–230.

Davis, M., and W. Sieg [2015]: 'Conceptual confluence in 1936: Post and Turing', in G. Sommaruga and T. Strahm, eds, *Turing's Revolution: The Impact of His Ideas about Computability*, pp. 3–27. Birkhäuser.
DOI: https://doi.org/10.1007/978-3-319-22156-4_1.

Davis, M., and E.J. Weyuker [1983]: *Computability, Complexity, and Languages. Fundamentals of Theoretical Computer Science. Computer science and applied mathematics.* Academic Press.

Davis, M., G. Logemann, and D.W. Loveland [1961a]: 'A machine program for theorem-proving', Technical Report AFOSR 819, IMM-NYU288. New York: Institute of Mathematical Sciences, New York University.

_____ [1962]: 'A machine program for theorem-proving', *Commun. ACM* **5**, 394–397. (Reprinted in [Siekmann and Wrightson, 1983], pp. 267–270.)
DOI: https://doi.org/10.1145/368273.368557.

Davis, M., H. Putnam, and J. Robinson [1961b]: 'The decision problem for exponential Diophantine equations', *Ann. of Math.* (2) **74**, 425–436. (Reprinted in [Robinson, 1996], pp. 77–88. Russian transl. in [Matematika, 1964], pp. 69–79.)
DOI: https://doi.org/10.2307/1970289.

Davis, M., Y. Matijasevič, and J. Robinson [1976]: 'Hilbert's tenth problem. Diophantine equations: positive aspects of a negative solution', in F.E. Browder, ed., *Mathematical Developments Arising From Hilbert Problems*, pp. 323–354, 357–358, 355–356, 359–378 (should be read in this order) Proc. Sympos. Pure Math.; 28. Providence, R.I.: AMS. Reprinted in [Robinson, 1996], pp. 269 ff.

Davis, M., R. Sigal, and E.J. Weyuker [1994]: *Computability, Complexity, and Languages. Fundamentals of Theoretical Computer Science.* Computer Science and Scientific Computing. 2nd ed. Academic Press.

Daylight, E.G. [2014]: 'A Turing tale', *Commun. ACM* **57**, No. 10, 36–38.
    DOI: https://doi.org/10.1145/2629499.

De Mol, L., B. Maarten, and E.G. Daylight [2018]: 'Less is more in the fifties: Encoun-
    ters between logical minimalism and computer design during the 1950s', *IEEE
    Ann. of the Hist. of Comput.* **40**, 19–45.
    DOI: https://doi.org/10.1109/MAHC.2018.012171265.

Gandy, R. [1988]: 'The confluence of ideas in 1936', in R. Herken, ed., *The Universal
    Turing Machine: A Half-Century Survey*, pp. 55–111. Oxford University Press.
    DOI: https://doi.org/10.1093/oso/9780198537748.003.0003.

Gilmore, P.C. [1960]: 'A proof method for quantification theory: Its justification and
    realization', *IBM Journal of Research and Development* **4**, 28–35.
    DOI: https://doi.org/10.1147/rd.41.0028.

Green, B., and T. Tao [2008]: 'The primes contain arbitrarily long arithmetic progres-
    sions', *Ann. of Math.* (2) **167**, 481–547.
    DOI: https://doi.org/10.4007/annals.2008.167.481.

Hilbert, D. [1900]: 'Mathematische Probleme. Vortrag, gehalten auf dem interna-
    tionalen Mathematiker-Kongreß zu Paris 1900', *Nachrichten von der Königliche
    Gesellschaft der Wissenschaften zu Göttingen*, 253–297. (Translated as [Hilbert,
    1901–1902]).

———— [1901–1902]: 'Mathematical Problems. Lecture delivered before the Interna-
    tional Congress of Mathematicians at Paris in 1900', *Bull. AMS* **8**, 437–479.

Jackson, A. [2008]: 'Interview with Martin Davis', *Notices AMS* **55**, 560–571.
    DOI: https://doi.org/10.1090/noti457.

Kib [1970]: *Kiberneticheskiy sbornik* (N.S.) **7**. Moscow: Mir.

Kleene, S.C. [1952]: *Introduction to Metamathematics*. Groningen: Noordhoff N.V.

———— [1981]: 'Origins of recursive function theory', *IEEE Ann. Hist. Comput.* **3**,
    52–67. DOI: https://doi.org/10.1109/MAHC.1981.10004.

Koymans, P., and C. Pagano [2024]: 'Hilbert's tenth problem via additive combina-
    torics'. DOI: https://doi.org/10.48550/arXiv.2412.01768.

Loveland, D., S. Ashish, and B. Selman [2016]: 'DPLL: The core of modern satisfia-
    bility solvers', in [Omodeo and Policriti, 2016], pp. 315–335.
    DOI: https://doi.org/10.1007/978-3-319-41842-1_12.

Mahoney, M.S. [1988]: 'The history of computing in the history of technology', *Ann.
    Hist. Comput.* **10**, 113–125.

Mal'tsev, A. [1968]: 'O nekotorykh pogranichnykh voprosakh algebry i logiki', in *Trudy
    Mezhdunarodnogo Kongressa Matematikov*, pp. 217–231 Moscow: Mir.

Martin-Löf, P. [1970]: *Notes on Constructive Mathematics*. Stockholm: Almqvist &
    Wikseil.

Matematika [1964]: *Matematika* **8**. Moscow: Mir.

Matiyasevich, Y.V. [1970]: 'Diofantovost' perechislimykh mnozhestv', *Doklady
    Akademii Nauk SSSR* **191**, 279–282. (Russian. English translation by A.
    Doohovskoy as Ju.V. Matijasevič, 'Enumerable sets are Diophantine', *Soviet Math.
    Doklady* **11**, 354–358; Correction, *ibid.* **11**, vi (for 1970, pub. 1971). (Translation
    reprinted in [Sacks, 2003], pp. 269–273).

———— [1993]: *Desyataya Problema Gilberta*. Moscow: Nauka. English translation:
    Hilbert's Tenth Problem. Cambridge, Mass.: MIT Press. French translation: *Le
    dixième Problème de Hilbert: Son indécidabilité*, Paris: Masson. Greek translation:
    To dekato provlima tou Hilbert, Athens: Euryalos, 2022. https://logic.pdmi.ras.ru/
    ~yumat/H10Pbook/. Accessed July 2025.

Moschovakis, Y.N. [2016]: 'Hyperarithmetical sets', in [Omodeo and Policriti, 2016], pp. 107–149. DOI: https://doi.org/10.1007/978-3-319-41842-1_5.

Murty, M.R., and B. Fodden [2019]: *Hilbert's Tenth Problem. An Introduction to Logic, Number Theory, and Computability.* Stud. Math. Libr.; 88 Providence, R.I.: AMS. DOI: https://doi.org/10.1090/stml/088.

Omodeo, E.G. [1982]: 'The linked conjunct method for automatic deduction and related search techniques', *Comput. Math. Appl.* **8**, 185–203.

Omodeo, E.G., and A. Policriti, eds. *Martin Davis on Computability, Computational Logic, and Mathematical Foundations.* Outstanding Contributions to Logic; 10. Springer. DOI: https://doi.org/10.1007/978-3-319-41842-1.

Penrose, R. [1989]: *The Emperor's New Mind: Concerning Computers, Minds and the Laws of Physics.* Oxford University Press.

———— [1994]: *Shadows of the Mind: A Search for the Missing Science of Consciousness.* Oxford University Press.

Post, E.L. [1943]: 'Formal reductions of the general combinatorial decision problem', *Amer. J. Math.* **65**, 197–215.

———— [1944]: 'Recursively enumerable sets of positive integers and their decision problems', *Bull. AMS* **50**, 284–316.

Prawitz, D., H. Prawitz, and N. Voghera [1960]: 'A mechanical proof procedure and its realization in an electronic computer', *Journal ACM* **7**, 102–128.

Reid, C. [1996]: *Julia: A Life in Mathematics.* Washington, D.C.: Mathematical Association of America. With contributions by L. Gaal, M. Davis, and Yu. Matijasevich. Russian translation: [Rid, 2023].

Rid, K. [2023]: *Dzhuliya. Zhizn v matematike.* St Petersburg: 'Obrazovatelnye proekty' ('Educational Projects'). Translation of [Reid, 1996].

Robinson, J. [1996]: *The Collected Works of Julia Robinson.* Vol. 6. With an introduction by Constance Reid. Edited and with a foreword by Solomon Feferman. Providence, R.I.: AMS.

Robinson, R.M. [1956]: 'Arithmetical representation of recursively enumerable sets', *J. Symbolic Logic* **21**, 162–186. DOI: https://doi.org/10.2307/2269027.

G.E., Sacks, ed. [2003]: *Mathematical Logic in the 20th Century.* Singapore: Singapore University Press. River Edge, N.J.: World Scientific Publishing Co.

Sieg, W. [1997]: Step by recursive step: Church's analysis of effective calculability', *Bull. Symbolic Logic* **3**, 154–180. DOI: https://doi.org/10.2307/421012.

Siegelmann, H.T. [1999]: *Neural networks and analog computation: Beyond the Turing limit.* Progress in theoretical computer science. Birkhäuser.

Siekmann, J., and G. Wrightson, eds. [1983]: *Automation of Reasoning 1: Classical Papers on Computational Logic 1957–1966.* Springer.

Tseitin, G.S. [1964]: 'A method of presenting the theory of algorithms and enumerable sets' (in Russian). *Trudy Mat. Inst. Steklov* **72**, 69–98. English translation in *Five Papers on Logic and Foundations by G.S. Ceitin, N.N. Vorob'ev, A.V. Idel'son, I.D. Zaslavskii, and N.A. Sanin*, pp. 1–39. AMS Translations, Series II; 99. Providence, R.I.: AMS, 1972. DOI: https://doi.org/10.1090/trans2/099.

Turing, A.M. [1937]: 'On computable numbers, with an application to the *Entscheidungsproblem*', *Proc. London Math. Soc.* (2) **42**, 230–265. DOI: https://doi.org/10.1112/plms/s2-42.1.230. Correction, *ibid.* **43** (1938), 544–546. DOI: https://doi.org/10.1112/plms/s2-43.6.544.

———— [1950]: 'The word problem in semi-groups with cancellation', *Ann. of Math.* (2) **52**, 491–505. DOI: https://doi.org/10.2307/1969481.

_____ [1954]: 'Solvable and unsolvable problems', *Science News* (Penguin Press) **31**, 7–23. Reprint with introduction
DOI: https://doi.org/10.1093/oso/9780198250791.003.0024.

Wang, H. [1960]: 'Toward mechanical mathematics', *IBM Journal of Research and Development* **4**, 2–22. DOI: https://doi.org/10.1147/rd.41.0002.

Webb, J.C. [1980]: *Mechanism, Mentalism and Metamathematics.* Dordecht: Reidel.

Yarmush, D.L. [1976]: 'The linked conjunct and other algorithms for mechanical theorem-proving'. Technical Report IMM 412. New York: Courant Institute of Mathematical Sciences, New York University.