Proving Church's Thesis
(Abstract)

Yuri Gurevich

Microsoft Research

The talk reflects recent joint work with Nachum Dershowitz [4].

In 1936, Church suggested that the recursive functions, which had been defined by Gödel earlier that decade, adequately capture the intuitive notion of a computable ("effectively calculable") numerical function[1] [2]. Independently Turing argued that, for strings-to-strings functions, the same goal is achieved by his machines [11].

The modern form of Church's thesis is due to Church's student Kleene. It asserts that every computable numerical partial function is partial recursive. (Originally Church spoke of total functions.)

Kleene thought that the thesis as unprovable: "Since our original notion of effective calculability... is a somewhat vague intuitive one, the thesis cannot be proved" [7]. But he presented evidence in favor of the thesis. By far the strongest argument was Turing's analysis [11] of "the sorts of operations which a human computer could perform, working according to preassigned instructions" [7]. The argument convinced Gödel who thought the idea "that this really is the correct definition of mechanical computability was established beyond any doubt by Turing" [5].

Moreover, Gödel has been reported to have thought "that it might be possible ... to state a set of axioms which would embody the generally accepted properties of [effective calculability], and to do something on that basis" [3]. As explained by Shoenfield [10]:

> It may seem that it is impossible to give a proof of Church's Thesis. However, this is not necessarily the case.... In other words, we can write down some axioms about computable functions which most people would agree are evidently true. It might be possible to prove Church's Thesis from such axioms.... However, despite strenuous efforts, no one has succeeded in doing this (although some interesting partial results have been obtained).

We will demonstrate that, under certain very natural hypotheses regarding algorithmic activity, called the "Sequential Postulates" [6], Church's Thesis is in fact provable. In brief, the postulates say the following.

I. **Sequential Time.** *An algorithm determines a sequence of "computational" states for each valid input.*

[1] For brevity we use the term numerical function to mean a function from natural numbers to natural numbers.

V. Diekert, M. Volkov, and A. Voronkov (Eds.): CSR 2007, LNCS 4649, pp. 1–3, 2007.

II. **Abstract State.** *The states of a computational sequence can be arbitrary (first-order) structures.*

III. **Bounded Exploration.** *The transitions from state to state in the sequence are governed by a finite description.*

For precise formulation of the three postulates see the article [6]. With Bounded Exploration, an algorithm computes in "steps of limited complexity", as demanded by Kolmogorov [8]. This postulate thereby answers Kolmogorov's implicit question: What does it mean to bound the complexity of each individual step?

The postulates are justified in the article [6]. On this ground, a (sequential) algorithm is defined there as any object satisfying the three postulates. One may worry that this definition is too liberal. To this end, the article proves that (sequential) abstract state machines, introduced earlier by the author, satisfy the three postulates, and that, for every algorithm, there is an abstract state machine that emulates the algorithm.

To focus on numerical algorithms, we add the following postulate:

IV. Only basic arithmetic operations are available initially.

Algorithms satisfying postulate IV will be called numerical.

We will show that Church's Thesis provably follows from these four postulates.

Theorem 1. *Any numerical partial function is computed by a numerical algorithm if and only if it is partial recursive.*

Thus, to the extent that one might entertain the notion that there exist non-recursive effective functions, one must reject one or more of these postulates. In a similar way, we can prove Turing's thesis from postulates I–III and a postulate.

V. Only basic string operations are available initially.

Theorem 1 generalizes to the case when oracles are present. If only oracle functions are available initially, postulates I–III suffice. No additional postulates are needed.

Our goal in this work has been to remedy the situation described thus by Montague [9]: "Discussion of Church's thesis has suffered for lack of a precise general framework within which it could be conducted." We show how the Sequential ASM Postulates provide just such a framework. As we mentioned, Gödel surmised that Church's Thesis may follow from appropriate axioms of computability. But, as far as we can ascertain, no complete axiomatization has previously been presented in the literature. In fact, the challenge of proving Church's Thesis is first in Shore's list of "pie-in-the-sky problems" for the twenty-first century [1].

References

1. Buss, S.R., Kechris, A.A., Pillay, A., Shore, R.A.: Prospects for mathematical logic in the twenty-first century. Bulletin of Symbolic Logic 7(2), 169–196 (2001)
2. Church, A.: An unsolvable problem of elementary number theory. American Journal of Mathematics 58, 345–363 (1936)
3. Davis, M.: Why Gödel didn't have Church's thesis. Information and Control 54, 3–24 (1982)
4. Dershowitz, N., Gurevich, Y.: A natural axiomatization of Church's thesis (to appear)
5. Gödel, K.: Kurt Gödel: Collected Works. In: Feferman, S. (ed.) Unpublished essays and lectures, vol. III, p. 168. Oxford University Press, Oxford (1995)
6. Gurevich, Y.: Sequential abstract state machines capture sequential algorithms. ACM Transactions on Computational Logic 1, 77–111 (2000)
7. Kleene, S.C.: Introduction to Metamathematics, North Holland, Amsterdam (1952)
8. Kolmogorov, A. N.: O ponyatii algoritma (On the concept of algorithm), Uspekhi Matematicheskikh Nauk, vol. 8(4), pp. 175–176 (1953) (in Russian). An English translation is In: Uspensky, V.A., Semenov, A.L.: Algorithms: Main Ideas and Applications, pp. 18–19. Kluwer (1993)
9. Montague, R.: Towards a general theory of computability. Synthese 12(4), 429–438 (1960)
10. Shoenfield, J.R.: Recursion Theory. Lecture Notes in Logic, vol. 1. Springer, New York (1991)
11. Turing, A.M.: On computable numbers, with an application to the Entscheidungsproblem. In: Proceedings of the London Mathematical Society, vol. 42, pp. 230–265, 1936–37. Corrections in vol. 43, pp. 544–546 (1937)