CDMTCS
Research
Report
Series

Breaking the Turing Barrier

Cristian S. Calude
Michael J. Dinneen
Department of Computer Science
Univeristy of Auckland

CDMTCS-084
May 1998

Centre for Discrete Mathematics and
Theoretical Computer Science

Breaking the Turing Barrier

C.S. Calude and M.J. Dinneen

Is there any limit to discrete computation, and more generally, to scientific knowledge? This is one of the problems studied by the Centre for Discrete Mathematics and Theoretical Computer Science of the University of Auckland.

The story started in 1936. As a result of an original analysis of mental activity, A. Turing introduced (in [8]) the abstract model of a machine—now called the Turing machine—to define the concept of a fixed computational method or algorithm. He also introduced the "universal" Turing machine, a single machine capable of performing any instructions given to it.

Practically all conventional computers (e.g., PC's, Unix workstations, and mainframes) are based on the idea of a computer that stores and executes a program from internal memory (or from an external device). These are known as the J. von Neumann architectures as opposed to machines that have "hard-wired" instructions. They are a realization of Turing's universal machine where the instructions for a basic Turing machine (e.g., a program) are read from an infinite tape and executed. See Figure 1.

For over fifty years, the Turing machine model of computation has defined what it means to "compute" something and essentially all theoretical results in computer science rest on this model.

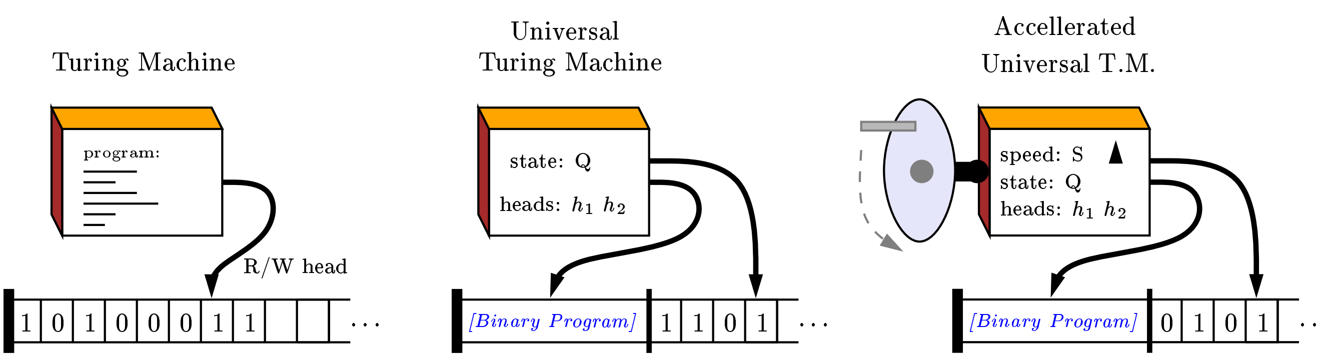

Figure 1: Various types of Turing machines.

In recent years, researchers have looked at natural processes in the physical and biological world as motivation for constructing new models of computation holding out the *hope of breaking the "Turing barrier."* But are there alternatives? The quantum phenomenon of interference has led to one such model, as has the process of folding of DNA strands in a living cell. In addition, refinements to the Turing view of computing have led to "super-Turing" models, that allow one to compute in ways that transcend Turing's original scheme. Breaking Turing's barrier is double important: a) theoretically, as unconventional models are explored with an eye toward understanding the *true* limits of computation, thereby shedding light on the basic questions on the limits to scientific knowledge, b) practically, as a method to speed-up computations beyond classical capabilities.

Here is a possible scenario. Computers, in contrast to Turing machines, are physical devices: whatever they can or cannot do is determined by the laws of physics. In particular, Turing machines make no mention of time! A Turing machine operates with a finite list of primitive operations—read a square of the tape, write a single symbol on a square of the tape, move one square to the right, and so forth—but it makes no mention of the *duration* of each primitive operation (of course, we may assume that each primitive operation requires a fixed duration, but this is not part of the original model). It is only important whether or not the machine halts after a finite *number* of operations. Temporal considerations are not relevant for these mathematical models. Things are different for real computers where time does matter.

Well before Turing's model, H. Weyl, in 1927, has considered a hypothetical machine that is capable of completing an *infinite* sequence of distinct operations within a *finite* time[1] say, by supplying the first result after 1/2 minute, the second after another 1/4 minute, the third 1/8 minute later, etc. In fact, this temporal patterning was described earlier by B. Russell, in a famous lecture given in Boston, 1914. In a discussion of Zeno's paradox of the race-course, Russell said "If half the course takes half a minute, and the next quarter takes a quarter of a minute, and so on, the whole course will take a minute". Is it physically possible to execute an infinite number of operations in a finite amount of time? Science has not offered a definitive answer yet. Russell (1935) has argued that for human beings this scenario is not *logically* impossible, but it may be *biologically* impossible... (he said: "Might not a man's skill increase so fast that he performed each operation in half the time required for its predecessor?"). What about computers? K. Svozil [7] suggests the answer depends upon the underlying physical environment considered for computation. Classically, i.e., when computation is performed within a classical universe and information is measured in bits, infinite sequences of operations cannot be operationally executed in a finite time. However, if one places the computation into a quantum mechanical universe, when classical bits are replaced by coherent superpositions of two orthonormal quantum states, the so-called quantum bits (qubits), infinite sequences of computations can be performed. Stated

[1] See [10]; for some more references on accelerated machines see Copeland [3].

differently, at the level of probability amplitudes, quantum theory allows a Zeno type computation. There is a price to be paid: computation appears to occur entirely random.

Is there a realistic chance to perform quantum computations? Nonlinearity (to support quantum logic and ensure universality) and coherence (for the manipulation of coherent quantum superpositions) are necessary and, in principle, sufficient conditions for computation. Conventional devices under investigation for carrying out these operations include ion traps, high finesse cavities for manipulating light and atoms using quantum electrodynamics, and molecular systems designed to compute using nuclear magnetic resonance. The latter store quantum information in the states of quantum systems such as photons, atoms, or nuclei, and realise quantum logic by semiclassical potentials such as microwave or laser fields. Unconventional ideas for quantum computation include fermionic quantum computers, bosonic computers, and architectures relying on anyons. See more in [1, 2].

Turing thought that a computer was capable in principle of doing anything the human brain can do, and in 1950 he set forth a theory that remains a cornerstone of artificial intelligence. The test proposed by Turing involved a computer communicating with human judges via a teleprinter link: if the computer's responses to questions were indistinguishable from those of a human being, Turing stated, the computer has to be regarded as exhibiting intelligence. The Turing test provoked a lot of discussion. R. Penrose has dedicated three books to this topic: [4, 5, 6]. In his last one, [6], he claims that "appropriate physical action of the brain evokes awareness, but this physical action cannot even be properly simulated computationally". The reference is made to classical computation; unconventional paradigms may change dramatically this view.

References

[1] Calude, C. S., Casti, J., and Dinneen, M. J. *Unconventional Methods of Computation.* Springer, Singapore, 1998.

[2] Calude, C. S., and Casti, J. L. Parallel thinking. *Nature 392* (1998), 549–551.

[3] Copeland, J. Even Turing machines can compute uncomputable functions. In *Unconventional Models of Computation* (Auckland, 1998), C. S. Calude, J. Casti, and M. J. Dinneen, Eds., Springer, Singapore, 1998, pp. 150–164.

[4] Penrose, R. *The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics.* Oxford University Press, Oxford, 1990.

[5] Penrose, R. *Shadows of the Minds, A Search for the Missing Science of Consciousness.* Oxford University Press, Oxford, 1994.

[6] PENROSE, R. *The Large, the Small and the Human Mind.* Cambridge University Press, Cambridge, 1997.

[7] SVOZIL, K. The Church-Turing Thesis as a guiding principle. In *Unconventional Models of Computation* (Auckland, 1998), C. S. Calude, J. Casti, and M. J. Dinneen, Eds., Springer, Singapore, 1998, pp. 371–385.

[8] TURING, A. M. On computable numbers, with an application to the Entscheidungsproblem". *Proceedings of the London Mathematical Society, Series 2, 42* (1936), 230–265 and *43* (1937), 544–546.

[9] TURING, A. M. Computing machinery and intelligence. *Mind 59* (1950), 433–460.

[10] WEYL, H. *Philosophie der Mathematik und Naturwissenschaft.* R. Oldenbourg, Munich, 1927.