Alan Turing: Mathematical Mechanist[1]

Anthony F. Beavers
The University of Evansville

I live just off of Bell Road outside of Newburgh, Indiana, a small town of 3,000 people. A mile down the street Bell Road intersects with Telephone Road not as a modern reminder of a technology belonging to bygone days, but as testimony that this technology, now more than a century and a quarter old, is still with us. In an age that prides itself on its digital devices and in which the computer now equals the telephone as a medium of communication, it is easy to forget the debt we owe to an era that industrialized the flow of information, that the light bulb, to pick a singular example, which is useful for upgrading visual information we might otherwise overlook, nonetheless remains the most prevalent of all modern day information technologies. Edison's light bulb, of course, belongs to a different order of informational devices than the computer, but not so the telephone, not entirely anyway.

Alan Turing, best known for his work on the Theory of Computation (1937), the Turing Machine (also 1937) and the Turing Test (1950), is often credited with being the father of computer science and the father of artificial intelligence. Less well-known to the casual reader but equally important is his work in computer engineering. The following lecture on the Automatic Computing Engine, or ACE, shows Turing in this different light, as a mechanist concerned with getting the greatest computational power from minimal hardware resources. Yet Turing's work on mechanisms is often eclipsed by his thoughts on computability and his other theoretical interests. This is unfortunate for several reasons, one being that it obscures our picture of the historical trajectory of information technology, a second that it emphasizes a false dichotomy between "hardware" and "software" to which Turing himself did not ascribe but which has, nonetheless, confused researchers who study the nature of mind and intelligence for generations. We are only today starting to understand the error of our ways. *Our* ways … not Turing's, as the following essay makes clear.

The second issue follows from the first, from understanding Turing independently of his mechanistic tendencies and his connections to the *industrial* information revolution of the 19th Century. Thus it is fitting to rescue the popular under-

---

[1] Penultimate draft of an introductory piece for Turing's 1947 Lecture to the London Mathematical Society. In *Alan Turing: His Work and Impact,* edited by S. Barry Cooper and Jan van Leeuwen (Elsevier, 2011), forthcoming.

standing of Turing from the two essays for which he is most notably known (1937 & 1950) in order to shed some light on his genuine place in history and, at the same time, to examine some of the implications that should have been clear to the philosophers and psychologists that immediately followed him. He was, in more than one way, ahead of his time, though oddly, as we shall see, because he was thoroughly connected to his past.

"It is … quite difficult to think about the code entirely in abstracto without any kind of circuit," (???) Turing writes in this lecture, suggesting that in a working machine there is no "code," just hardware, and that really what "computer code" does is to configure circuitry within computing machinery to perform a particular information processing task. For ease of communication, I will call the belief that software constitutes a separate level of machine processing, that "code" *can* be understood in abstraction from circuitry, the "software seduction." This position is most famously advanced by David Marr (1982). Though it was originally intended to assist him in the analysis of vision, it set the stage for several theories in cognitive science and the philosophy of mind. Marr's "Tri-Level Hypothesis" decomposed an information processing system into three levels for the sake of analysis, the computational, algorithmic and implementational levels. The first level concerns the input/output specifications of the system, while the algorithmic level specifies the processes and representations whereby inputs are transformed into their appropriate outputs. The implementational level concerns how the algorithmic level is instantiated in something physical.

For Marr, and several in the tradition that follows him, the real work of information processing is best understood by examining algorithms while the implementation is largely incidental. In other words, the "code" here *is* predominantly understood in abstraction from the circuits, or, in modern terms, "software" is deemed more important for understanding information processing than "hardware." In turn, this tri-partite division led historically to a "computer analogy" for understanding human cognition in which mind is taken to be to software as brain (or body) is to hardware. As a consequence, to understand mental function, we need only consider problem-solving and other cognitive tasks on the algorithmic level. The details of the neural substrate belong to another science.

Traditional cognitive scientists still ascribe to this distinction, which is deeply rooted in the separation between cognitive psychology and neuroscience that has become part of the DNA of the modern academy. Of course, as traditions tend to fall to the past and new conceptions take their place, this paradigm too is on its way out. Embodied and situated cognition has made considerable headway in producing explanations that are both more fruitful and biologically plausible that those rooted in the aforementioned "software seduction." We are coming to understand that mind cannot be understood in abstraction from brain, body and environment just as code cannot be understood in abstraction from circuitry. Regarding Turing then and, in particular, the essay to follow, these new explanations of cognition cannot be said to represent a turn away from the Turing paradigm, but rather a turn toward it, once we understand Turing's own mechanical tendencies. If mind is to brain as software is to hardware, then a true rendition of the "computer analogy" would suggest that in a real working brain, mind is just more brain, more circuitry. We see why this is so in the subsequent essay, which again connects Turing to his past and helps us understand his true contribution to the history of information technology, a topic which I now address.

Though one common view is that the "information age" begins with the birth of digital technologies and thus owes its debt to Turing (Floridi 2008), when we explore the history of informational mechanisms, we see that this is only partially correct. It is quite true that the recent explosion in informational devices has increased dramatically since *Time Magazine* named the computer "person of the year" in 1982, but it belongs nonetheless to a trajectory that was firmly set in motion by the end of the 19th Century with what we might call the "multimedia revolution" (Beavers 2012). Such a view does not minimize Turing's contribution, but helps to illustrate precisely where it lies. It is obviously not insignificant.

By "multimedia revolution," I mean to invoke here an era in which information was decoupled from the exigencies of transportation technology and made to move on its own. Prior, if information traveled from point A to B, it was because someone carried it there; but just as the industrial revolution issued in a range of new mechanisms for everything from agriculture to textile manufacturing, it did the same for information. The casual reader, to be sure, is mostly aware of this fact, but its significance might not readily be clear. Some inventions and their approximate dates might add some clarity.

Though putting starting dates on such historical transformation is risky business, it is perhaps not too much of a falsification to date the beginning of the industrialization of information with the telegraph in 1836 and the Daguerreotype in 1839, which introduced practical photography. The telegraphic printer and the stock ticker followed soon after in 1856 and 1863, respectively. The years between 1876 and 1881 were perhaps the most immediately transformative with the telephone in 1877, the phonograph in 1878, and the light bulb and the photophone in 1880. The latter, Bell's favorite invention, could mediate telephone communications by modulating wave forms on a beam of light. Wireless telegraphy and the wax cylinder, which made Edison's phonograph practical, both emerged in 1881 with the wax cylinder (invented by Bell) serving as a patent reminder that not all of these technologies were electric. Many were not, and many were not adopted for the same uses to which we put them today. The telephone, for instance, was an early form of broadcasting prior to radio (which is why a radio was originally referred to as a "wireless"), and of the ten uses Edison enumerates for the phonograph, recording music is listed fourth, with distance education suggested ninth and then last, "connection with the telephone, so as to make that instrument an auxiliary in the transmission of permanent and invaluable records, instead of being the recipient of momentary and fleeting communication" (Edison 1878).

Edison's vision of connecting phonographs to the telephone system to aid in the preservation and transmission of information anticipates Turing's observations in the following essay that "it would be quite possible to arrange to control a distant computer by means of a telephone line," costing "a few hundred pounds at most" (???). This would entail modulating wave forms over the telephone to allow for digital transmission, which was easy enough to do. However, the important thing is that for both men, hints of a networked world are present because of the invention and quick adoption of the telephone. In fact, by 1910 the telephone had become so popular as to warrant its own history. Herbert Casson's *The History of the Telephone* paints a vivid picture of the societal transformation in progress:

> What we might call the telephonization of city life, for lack of a simpler word, has remarkably altered our manner of living from what it was in the days of Abraham Lincoln. It has enabled us to be more social and cooperative. It has literally abolished the isolation of separate families, and has made us members of one great family. It has become so truly an organ of the social body that by telephone we now enter into contracts, give evidence, try lawsuits, make speeches, propose marriage, confer degrees, appeal to voters, and do almost everything else that is a matter of speech. (199)

Indeed, long before the emergence of computing machinery, "human computers," as Turing liked to call people, were quickly interconnecting, and it is difficult to imagine the invention of the mechanical computer (and a host of other technologies) without this affordance of the telephone. But there are deeper technological connections between the telephone and computing machinery which I will discuss momentarily. Before doing so, it is worth sketching a bit more of the history that belongs to the "multimedia revolution" by reminding the reader of a few more dates.

In 1891, Edison made pictures move with his invention of the motion picture camera. Radio and teletype would enter the scene in 1906, to be followed in 1914 by Edison's "telescribe," a precursor to the answering machine that has now recently been replaced with "voice mail." Pictures would take to the airwaves with the public appearance of the television in 1926 and the National Broadcasting System in 1928, the same year that magnetic tape would become available. The speed of information transmission and its reach would continue with cable television in 1948, the cassette tape in 1958, the touch tone phone in 1963, color television in 1966 and the VCR in 1969.

Turing belongs to this historical trajectory, which I have intentionally recounted without mention of computing machinery to make a few points. The first is that the information revolution was well underway prior to Turing, but without one affordance that will greatly alter its landscape. Edison, Bell and several others could store information and move it around. They could not, however, capture it temporarily in a memory store, process it, and then produce meaningful output. Turing's paper of 1937 will introduce a theory of mechanical computation sufficient to put automated information processors into the mix of inter-networked human computers, but it will not provide the schematic for a practical piece of hardware. That work comes later, not solely by Turing, but partly so, as we see in this 1947 lecture to the London Mathematical Society. Second, not all of the technologies that belong to this information revolution were electric, as I previously noted, for instance, the phonograph, early cameras, motion picture cameras, etc. Electricity is important for computing machinery, but for practical reasons only, at least for what matters to the mathematician. Turing writes at the beginning of this lecture that "the property of being digital should be of greater interest than that of being electronic. That [computing machinery] is electronic is certainly important because these machines owe their high speed to this, and without the speed it is doubtful if financial support for their construction would be forthcoming. But this is virtually all that there is to be said on the subject" (???).

It is perhaps worth noting two things in regard to this quote. Computing machinery was not the first of the digital information mechanisms to enter this revolution. At the very beginning, Samuel Morse's famous code (c. 1830) was digital; the telegraph worked by sending pulses of current across an electrical line. Signals were encoded digitally, but not in binary, since the code used five "tokens" not two, a short electric pulse, a long one, and less obviously the empty space between them that used a short moment of silence to separate "dots" and "dashes," a longer one to separate letters and a longer one still for words. More importantly, the quote from Turing makes reference to the necessity of building economically feasible equipment. This concern is not trivial; not at all. In fact, as the essay indicates, cost concerns are going to drive the architecture of the machine and the need for "subsidiary [instruction] tables," that are stored in memory as early analogs to our functions or procedures. Getting the most computational power from minimal hardware resources, as I mentioned earlier, is at the heart of this essay, but for economic and not mathematical reasons. "Code" then becomes a strategy for creating temporary circuitry inside of the machine that can be reconfigured later for a different processing task. This highlights the notion that computer code is not essential for information processing tasks; separate hardware (i.e., circuitry) could be built for each task, but this would be both costly and inefficient. This makes it clear that one can, in fact, trade off software for hardware as needed. Indeed, Turing's goal in this lecture is to describe a machine that has minimal circuitry, but that can be configured with code for any processing task that is computable. Here, he does this with a 200KB memory store that is "probably comparable with the memory capacity of a minnow" (???), quite a remarkable accomplishment even by today's standards.

Code, then, is useful to the extent that it can reconfigure circuitry, and hence is not a necessary condition for information processing tasks. However, such a machine could not get by without easily accessible, writeable and erasable memory. This fact leads us to one of the central practical insights of this lecture, the development of an accessible and efficient memory store. The infinite tape suggested in the idealized Turing machine of 1937 will not do because of the time it would take to jump around the tape. In explaining why, Turing ties himself again to the history of information technology by referring back to the affordances of the book over papyrus scrolls, noting nonetheless that even an automated memory structure based on the book would be highly inefficient, though the memory store he will advocate here is based on an analogy with it. The book will be to the papyrus scroll as memory circuitry will be to the infinite tape of the idealized machine. Here, the memory circuitry will consist of 200 separate mercury acoustic delay lines (analogous to pages in a book) that can each contain 1024 bits of information. I will not go into a complete description of how the system works here, since the details are spelled out in the lecture. I will, however, make a few final observations, that tie Turing to the history of information technology.

To start, many of the inventions of this era that industrialized the flow of information did so by modulating wave forms of various sorts. I have already mentioned Bell's experiments with light in this regard, but it was his early experiments with electricity that made the telephone (prior to cell phones) practical. Others, of course, were involved in a range of experiments (Tesla, Marconi, etc.) that modulated radio waves to support both radio and television. In this lecture, Turing considers the use of magnetic wires and cathode ray tubes before settling on the use of acoustic delay lines. Though analog, a close inspection of the schematics of Bell, Marconi or Tesla shows sufficient similarity to Turing's designs to plot him on the same trajectory. So, of course, the idea of modulating wave forms does not belong to Turing. In fact, some might even be surprised to learn that his early work was this explicitly dedicated to such.

What then is important about this lecture? It would seem to be its patently practical nature and its mechanical commitments that challenge our picture of a mathematical Turing by presenting another Turing, the mechanist, who gets into the nuts and bolts of computing machinery. In doing so, as I have tried to make clear, he does not show a clean break that starts a new era, but strong attachments to the preceding one. This is not to suggest that Turing did not issue in an unprecedented era in the processing and storage of information and the speed of its dissemination. He did, along with several others. In this regard, I wish to close with one final comment about technological visionaries in general.

Turing foresaw the possibility of connecting the computer to the telephone in much the same way that Edison did. Bell was already on the track of wireless telephony, so much so that in 1897, William Ayrton could make a prediction about the future that is still so painted in the language and technology of his past that today it reads like a 1930's science fiction film:

> There is no doubt that the day will come, maybe when you and I are forgotten, when copper wires, gutta-percha coverings, and iron sheathings will be relegated to the Museum of Antiquities. Then, when a person wants to telegraph to a friend, he knows not where, he will call in an electro-magnetic voice, which will be heard loud by him who has the electro-magnetic ear, but will be silent to everyone else. He will call, 'Where are you?' and the reply will come, 'I am at the bottom of the coal-mine' or 'Crossing the Andes,' or 'In the middle of the Pacific'. (Fahie, 1900, vii)

No one back then, it seems, could imagine what would happen when the computer revolution, inaugurated by Turing, and the telephone revolution, inaugurated by Bell, would come crashing together at the end of the 20th century to connect everyone to computers (both human and mechanical) by way of hand-held computer/telephones and other devices that are both affordable and have more computational power than Turing himself imagined practical.

We stand on the shoulders of giants, information visionaries of the past two hundred years. Yet, if we could somehow transport them into today's information environment, I cannot help but think that even with the grand visions that they had, they would nonetheless be in utter surprise over where we have arrived. Shocked and baffled, amazed, they would perhaps feel both proud and humble. This makes it all the more shameful that Turing himself was never permitted to experience the appreciation of a world that owes him an incredible debt, not solely because of his brilliant mind, but also because he was not afraid to get his hands dirty. We should never disparage those who are willing to get dirty in the process of making the world a better place.

References

Beavers, A. 2012. In the beginning was the word and then four revolutions in the history of information. In Demir, H. (Ed.). *Luciano Floridi's Philosophy of Technology: Critical Reflections* (forthcoming). Springer.

Casson, H. 1910. *The history of the telephone*. Chicago: A. C. McClurg & Co.

Edison, T. 1878 (June). *The North American Review.* Retrieved from the U. S. Library of Congress, http://memory.loc.gov/ammem/edhtml/edcyldr.html, August 1st, 2010.

Fahie, J. 1900. *A history of wireless telegraphy, 1828-1899, including some bare-wire proposals for subaqueous telegraphs*. London: William Blackwood and Sons.

Floridi, L. 2008. Artificial intelligence's new frontier: Artificial companions and the fourth revolution. *Metaphilosophy* 39.4/5: 652-654.

Marr D. 1982. *Vision. A computational investigation into the human representation and processing of visual information*. New York, NY: W. H. Freeman.

Turing, A. 1937. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society* 2.1, 230-265.

Turing, A. 1950. Computing machinery and intelligence. *Mind* 59.236, 433-460.