Life's code script

Turing machines and cells have much in common, argues **Sydney Brenner**.

Biological research is in crisis, and in Alan Turing's work there is much to guide us. Technology gives us the tools to analyse organisms at all scales, but we are drowning in a sea of data and thirsting for some theoretical framework with which to understand it. Although many believe that 'more is better', history tells us that 'least is best'. We need theory and a firm grasp on the nature of the objects we study to predict the rest.

Three of Turing's papers are relevant to biology. In 1952, 'The chemical basis of morphogenesis'[1] explored the hypothesis that patterns are generated in plants and animals by "chemical substances called morphogens, reacting together and diffusing through a tissue". Using differential equations, Turing set out how instabilities in a homogeneous medium could produce wave patterns that might account for processes such as the segregation of tissue types in the developing embryo.

Yet biological support for Turing's idea has been marginal. The pre-ordered patterns found in *Drosophila* development do not fit the instability theory, which, until recently, could describe only chemical systems. Skin patterning has, however, been shown to follow a broader interpretation of Turing's terms[2], where cell-to-cell signalling pathways, rather than individual molecules, are considered. The ion channels postulated by Alan Lloyd Hodgkin and Andrew Huxley[3], also in 1952, were discovered more immediately by molecular biology.

Turing published another biology-related paper, in 1950. 'Computing machinery and intelligence'[4] introduced the Turing test as an imitation game in which an outside interrogator tries to distinguish between a computing machine and a human foil through their responses to questions. But the Turing test does not say whether machines that match humans have intelligence, nor does it simulate the brain. For that, we need a theory for how the brain works.

The most interesting connection with biology, in my view, is in Turing's most important paper: 'On computable numbers with an application to the *Entscheidungsproblem*'[5], published in 1936, when Turing was just 24.

Computable numbers are defined as those whose decimals are calculable by finite means. Turing introduced what became known as the Turing machine to formalize the computation. The abstract machine is provided with a tape, which it scans one square at a time, and it can write, erase or omit symbols. The scanner may alter its mechanical state, and it can 'remember' previously read symbols. Essentially, the system is a set of instructions written on the tape, which describes the machine. Turing also defined a universal Turing machine, which can carry out any computation for which an instruction set can be written — this is the origin of the digital computer.

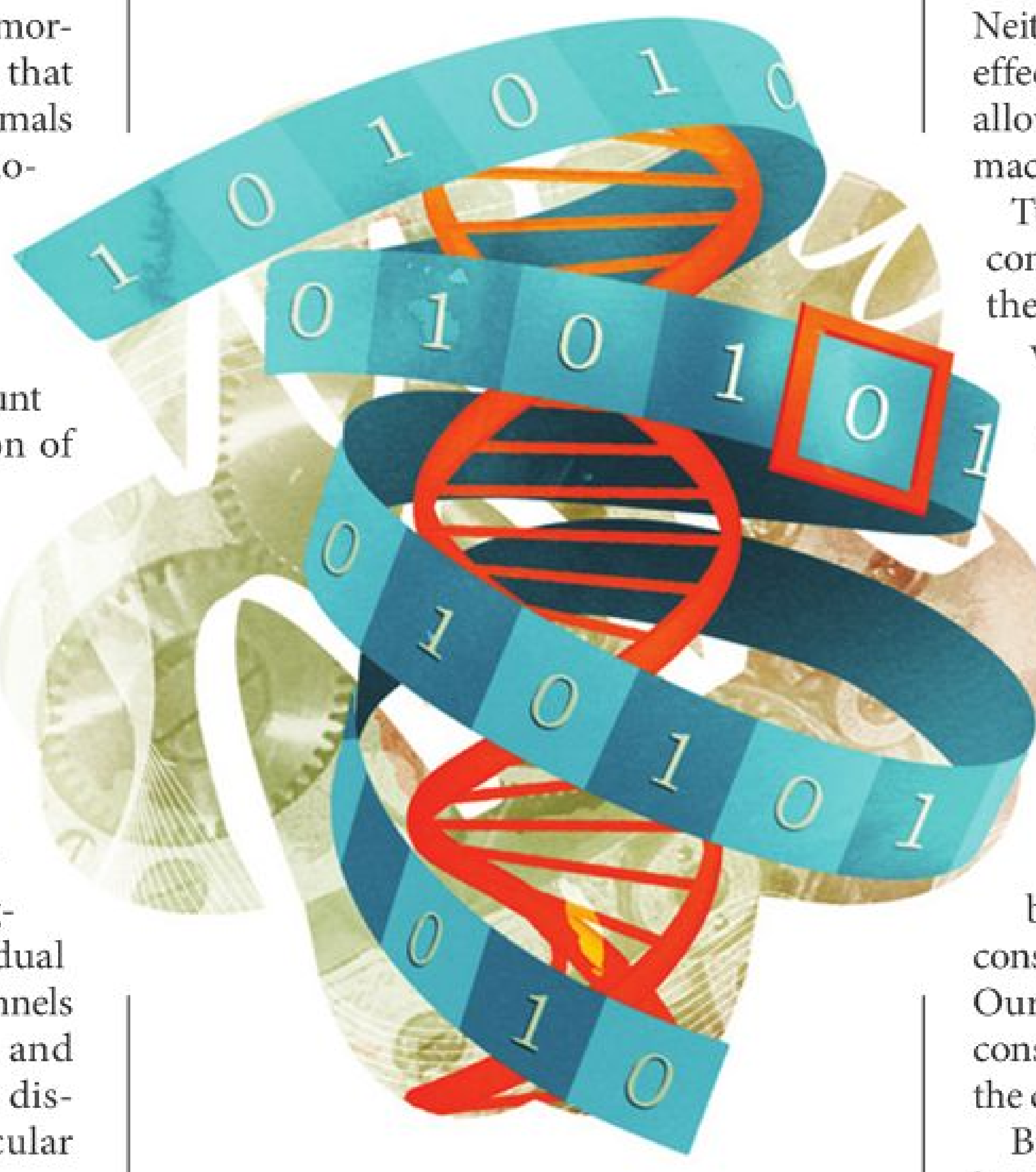

ILLUSTRATION BY ANDY POTTS

Turing's ideas were carried further in the 1940s by mathematician and engineer John von Neumann, who conceived of a 'constructor' machine capable of assembling another according to a description. A universal constructor with its own description would build a machine like itself. To complete the task, the universal constructor needs to copy its description and insert the copy into the offspring machine. Von Neumann noted that if the copying machine made errors, these 'mutations' would provide inheritable changes in the progeny.

Arguably the best examples of Turing's and von Neumann's machines are to be found in biology. Nowhere else are there such complicated systems, in which every organism contains an internal description of itself. The concept of the gene as a symbolic representation of the organism — a code script — is a fundamental feature of the living world and must form the kernel of biological theory.

Turing died in 1954, one year after the discovery of the double-helical structure of DNA by James Watson and Francis Crick, but before biology's subsequent revolution. Neither he nor von Neumann had any direct effect on molecular biology, but their work allows us to discipline our thoughts about machines, both natural and artificial.

Turing invented the stored-program computer, and von Neumann showed that the description is separate from the universal constructor. This is not trivial. Physicist Erwin Schrödinger confused the program and the constructor in his 1944 book *What is Life?*, in which he saw chromosomes as "architect's plan and builder's craft in one". This is wrong. The code script contains only a description of the executive function, not the function itself.

Thus, Hodgkin and Huxley's equations represent properties of the nerve impulse as an electrical circuit, but the required channels and pumps are constructed from specifications in the genes. Our problems reside in understanding the constructor part of the machinery, and here the cell is the right level of abstraction[6].

Biologists ask only three questions of a living organism: how does it work? How is it built? And how did it get that way? They are problems embodied in the classical fields of physiology, embryology and evolution. And at the core of everything are the tapes containing the descriptions to build these special Turing machines. ■

Sydney Brenner *is a senior fellow at the Janelia Farm Research Campus, Howard Hughes Medical Institute, Ashburn, Virginia, 20147, USA.*

1. Turing, A. M. *Phil. Trans. R. Soc. Lond. B* **237**, 37–72 (1952).
2. Kondo, S. & Miura, T. *Science* **329**, 1616–1620 (2010).
3. Hodgkin, A. L. & Huxley, A. F. *J. Physiol.* **117**, 500–544 (1952).
4. Turing, A. M. *Mind* **49**, 433–460 (1950).
5. Turing, A. M. *Proc. Lond. Math. Soc.* **s2–42**, 230–265 (1936–37).
6. Brenner, S. *Phil. Trans. R. Soc. B* **365**, 207–212 (2010).