CAN COMPUTERS FEEL?

- Mark Bishop, Goldsmiths College, UK.

In Science and Science Fiction the hope is periodically reignited that a computer system will one day be conscious in virtue of its execution of an appropriate program; indeed the EPSRC recently awarded an 'Adventure Fund' grant of £493,000 to a team of 'Roboteers' and Psychologists at Essex and Bristol, led by Owen Holland, with a goal of instantiating machine consciousness through appropriate computational 'internal modelling'. In contrast, below I outline a brief reductio style argument based on [1] that either suggests such optimism is misplaced or that Panpsychism, (the belief that 'the physical universe is composed of elements each of which is conscious'), is true.

In his 1950 paper, 'Computing Machinery and Intelligence', Turing defined Discrete State Machines, DSMs, as "machines that move in sudden jumps or clicks from one quite definite state to another", and explained that modern digital computers fall within the class of them. An example DSM from Turing is one that cycles through three computational states (Q_1, Q_2 & Q_3) at discrete clock clicks. Such a device, which cycles through a linear series of state transitions 'like clockwork', may be implemented by a simple wheel-machine that revolves through 120^0 intervals.

By labelling the three discrete positions of the wheel (A, B, C) we can map computational states of the DSM (Q_1, Q_2, Q_3) to the physical positions of the wheel, (A, B, C), such that, for example, (A => Q_1; B => Q_2; C => Q_3). Clearly this mapping is observer relative: position A could map to Q_2 or Q_3 and, with other states appropriately assigned, the machine's function would be unchanged. In general, we can generate the behaviour of any K-state (input-less) DSM, (f(Q) => Q'), by a K-state wheel-machine (e.g. a digital counter), and a function that maps each 'counter' state C_n to each computational state Q_n as required.

In addition, Turing's machine may be stopped by the application of a brake and whenever it enters a specific computational state a lamp will come on. Input to the machine is thus the state of the brake, (I = {ON | OFF}), and its output, (Z), the state of the lamp. Hence the operation of a DSM with input is described by a series of 'contingent branching state transitions', which map from current state to next state, f(Q, I) => Q' and define output, (in Moore form), f(Q') => Z.

However, (over a finite time interval), defining the input to the device entails that such 'contingent behaviour' reverts to 'clockwork', (f(Q) => Q'). E.g. If Turing's DSM starts in Q_1 and the brake is OFF for two clicks, its behaviour, (execution trace), is described by the sequence of state transitions, (Q_1; Q_2; Q_3). Hence, over a finite time window, if the input to a DSM is defined, we can map from each counter state C_n to each computational state Q_n, as required. In [1] I similarly demonstrate, pace Putnam, how to map any computational state sequence with defined input onto the [non-repeating] internal states generated by any Open Physical System, OPS, (e.g. a rock).

Now, returning to a putative conscious robot; at the heart of such a beast there is a computational system – typically a microprocessor; memory and peripherals. Such a system is a DSM. Thus, with input to the robot defined over a finite time interval, we can map its execution trace onto the state evolution of any digital counter or, ibid, any OPS. Hence, if the state evolution of a DSM instantiates phenomenal experience, then so must the state evolution of any OPS and we are inexorably led to a ‘Panpsychist’ worldview where phenomenal consciousnesses is found everywhere.

In [1] I discuss several objections to the above reductio with perhaps the most potent coming from David Chalmers who argues that, ‘as the above only implements one execution trace of the DSM it is not sensitive to counterfactuals; and it is only the possibility of appropriate counterfactual behaviour that guarantees phenomenal experience’.

But consider what happens if a putatively conscious robot, R_1, with full counterfactual sensitivity, is step-by-step transformed into new robot R_2, such that its resulting behaviour is determined solely by a linear series of state transitions; substituting each conditional branching state transition in the evolution of R_1, with a linear state transition defined by current state and the defined input. It is clear that, over a finite time interval and with identical input, the phenomenal experience of R_1 and R_2 must be the same. Otherwise we have a robot, R_n, whose phenomenal experience is contingent upon the deletion of state sequences that it does transit.

Counterfactuals cannot count and, being wary of Panpsychism, I conclude Computers cannot feel.

1. J.M. Bishop, *Counterfactuals Cannot Count: a rejoinder to David Chalmers*, **Consciousness & Cognition 11** (4), pp: 642-652, 2002.