Chapter 7
Induction by Rule Inference

Induction is the inference of general rules "causing" and "generating" (in an algorithmic interpretation) physical behaviours from these very behaviours (without any extra assumptions) alone. Thus induction is "bottom up:" given the phenomena and how observers perceive them operationally, these observers somehow obtain the causes and rules which supposedly underlie these phenomena. Thereby we shall restrict ourselves to algorithmic methods of induction; We shall not consider others, such as intuition, guesses or oracles, or means other than intrinsic.

Again it can be shown that for any deterministic system strong enough to support Peano arithmetic or universal computation, the induction problem for general algorithms (laws) or behaviours (phenomenology) is provable unsolvable. Induction is thereby reduced to the unsolvability of the rule inference problem [8, 14, 64, 246, 336]. This is the task to identify a rule or law reproducing the behaviour of a deterministic system by observing its input-output performance by purely algorithmic means (not by oracles or intuition).

Informally, the algorithmic idea of the proof is to take any sufficiently powerful rule or method of induction and, by using it, to define some functional behaviour which is not identified by it. This amounts to a sort of diagonalization; that is, the construction of an algorithm which (passively) fakes the guesser by simulating some particular function until the guesser pretends to be able to guess the function correctly. In a second, diagonalization step, the faking algorithm then switches to a different functional behaviour to invalidate the guesser's guess.

One can also relate this result to the recursive unsolvability of the halting problem, or in turn interpret it quantitatively in terms of the *busy beaver* function: there is no recursive bound on the time the guesser has to wait to make sure that the guess is correct. More generally, one could relate induction also to the problem of *functional equivalence,* which is provable undecidable [435, Sect. 2.1, pp. 33,34]: do two or more algorithms compute the same function? Two algorithms φ and ψ are equivalent if and only if they share a common domain (and image), and for any argument x of the domain, they are conditionally equal $\varphi(x) = \psi(x)$; that is, both sides are meaningful at the same time and, if meaningful, they assume the same value.

K. Svozil, *Physical (A)Causality*, Fundamental Theories of Physics 192,
https://doi.org/10.1007/978-3-319-70815-7_7