Computational Simulation as Indirect and Bounded Verification for Inductive Inferences (first draft)

by Joana Stella Kompa, Bangkok

Abstract

The following informal script explores the epistemological value of computational simulation in regards to inductive and deductive inferences, their advantages and limitations. The guiding question would be if any rational methods exists that lead us to the correct explanation of observations with certainty. The subsequent question from here is if then rational prediction is in fact possible. The given direction of conclusive flows from the specific to the general for inductive inferences and from the general to the specific for deductive inferences plays a critical role in the presented argumentation.

Keywords: Computational Simulation, Induction, Deduction, Truth-Range, Granularity (of Reality)

Basic Properties of Computational Simulation and Epistemic Consequences

Computational Simulation displays two intrinsic properties:

Firstly, computational simulation is entirely based and scripted on the deductive logic of the employed programming language. Computational models are therefore logical in their inherent processing and rational by architecture. This is a stark contrast to a world which seems ruled by fluctuations, anomalies and stochastic processes.

Secondly, computer models are low-resolution, coarse-grained simulations of real-world, fine-grained processes. The complexity of real-world networks can only be simulated within certain boundary conditions. These boundary conditions are set inherently by the resolution and detail of translated real-world instances into the form of virtual objects and externally by the availability of sufficient computing power.

This leads to the following critical arguments concerning computational simulation:

From property one: How can a fully deductive system possibly add to the truth-value of real-world observations? From property two: On which grounds can we even postulate truth value when we have to translate between computational simulation and our experience back and forth?

Finally, we could invite the devil’s advocate claiming that our world is actually inherently unpredictable and therefore any computational simulation contemplating prediction is simply a waste of time. Furthermore a radical skeptic could claim that even if prediction was possible it would run into a number of circular arguments by itself. We shall address these radical arguments later.
We can counter the concerns regarding the validity of justification for CS on the following two critical accounts:

1. **The Granularity Argument** *(Dimensional Reduction)*

The explanatory power of a fine-grained process $p_f$ and a coarse-grained process $p_c$ is identical if they derive independently at the same outcome while $p_c$ is element of $p_f$ and/or $p_c$ is the (extracted) **Eigen-Process** of $p_f$. An Eigen-Value is, simply put, an extracted subspace from the measurement of many samples. More than just being the average, Eigen-values are created from the values present in all instances and therefore make for highly realistic priors. They are vectors of a square matrix that remain parallel to their original vectors (instances). Deductive containment means that inductive real-world observations are now nested within a relating deductive computational framework. Thus the truth-value of our rational model is expressed in *the congruence* between computational simulation and actual predicted observations, so neither in the computational model nor in the physical evidence at hand by itself.

2. **The Cross-Referencing Argument of Truth**

Inductive inferences are circular (Hume) and, albeit rational, can therefore not be justified whereby deductive inferences *(modus ponens, modus tollens)* only transform the information set out in the premises as a 'fail safe’ mechanism.

A cross-referenced structure of truth means that it is not possible to create rational models without employing external real-world evidence and it’s relating prior conditions *(materia dependency)*. It is also not possible to evaluate evidence without the perspective of a mental model to view it from *(rational dependency)*. Since truth is created in the dialogue between external and internal arguments it cannot possibly have a single point of convergence. To speak with Hume it is the cross-referencing of ‘matters of facts’ and the ‘consultation of ideas’ that empowers us: without evidence our theories can neither be verified nor falsified against world-data and without competing, critical theories we have no way of comparing our premises and assumptions to the observations at hand.

Induction by itself will lead to an over-generalization error and deduction by itself will lead to an over-specification error. Over-generalization occurs when the global capture of measured probabilistic instances is too small to infer a reliable and representative statistical model, e.g., at the beginning of a learning process. Without a referenced model simple induction would remain blind to its own performance, so it eventually has to answer to the question ‘When and under which circumstances can a number of samples and sequence of observations satisfactorily form a hypothesis?’

Specification errors by contrast occur when deductive inferences suggest a too narrow range of what we can know or could expect. By reducing too many potential true beliefs we may miss out on capturing the fine-grained and multi-facetted structure of reality.

Minimizing both types of errors can be achieved by cross-referencing inductive and deductive results of an investigation to find the ‘sweet spot’ where both errors are minimized against each other in terms of providing the greatest explanatory power for a hypothesis in both directions.
The Avoidance of Logical Fallacies in Computational Simulation

We should not naively assume that computational simulation is immune to logical fallacies such as e.g., Affirming the Consequent. CS is only valid if the model is created genuinely bottom-up, this means that the properties of virtual objects and their priors create an authentic sequential output, a prerequisite which can be verified by a program’s structure. It is well possible to make up pretty much any outcome in CS and one could make the mistake to attribute explanation of instances ad posteriori, top-down, just because the model looks neat and suggestive.

Rational Confidence: About what we can know with Certainty

Since a computational simulation is by definition an abstracted, coarse-grained model of our world it is restricted to its authorized boundary conditions. This also means that its explanatory power is limited to a specific level of detail: we are e.g., able to simulate the conditions within a supernova explosion for fractions of a second, the weather for a couple of days and the collision between galaxies for millions of years. The reliability of computational simulation depends much on the capture of the underlying complexity of instances to be computed, divided by the timeframe that complexity is given to evolve plus the available computational power to calculate virtual processes.

Very successful are simulations of properties such as Finite Element Analysis for the automotive industry (crash-testing) as well as material stress-test simulations for the aerospace and aeronautical industries as they can predict future states with the highest precision. Despite the obvious technological applications that suggest a merely pragmatic approach we need to recognize that methodologically we can establish what I would call a ‘truth-range’: we can deductively infer the probability and prediction from observed and experienced phenomena and map them next to one another. ‘Range’ means in the context of computational simulation a temporal-spatial boundary; meaning that our claim of truth is only valid within the congruence with the physical world.

Certainty is expressed by the mutual verification of identical outcomes by inductive observation and deductive description and prediction.

The Devil’s Advocate & the Ghost in The Machine

A deeper argument plays on the fact that, if you could predict your actions, you could deliberately violate your predictions which means the predictions were wrong after all. (...)

At a personal level, even if the world is as deterministic as a computer program, you still can’t predict what you’re going to do. This is because your prediction method would involve a mental simulation of you that produces its results slower than you. You can’t think faster than you think. You can’t stand on your own shoulders.

I very much liked Rudi Rucker’s radical and intuitive statement as it runs contrary to the popular assumption of artificial consciousness. What his examples show nicely is that we would run into circular contradictions if we could in fact simulate our own consciousness.

However, both the Granularity Argument as well as the Cross-Referencing of Truth Argument prevents such a possibility: since computational simulation is a reconstruction of reality with an inherently limited truth-range the creation of 1:1 grained models are not a given. Based on what we have argued at this stage is that artificial thought would only be possible as a very rough simulated model, a shadow of its original. It would furthermore lose its synchronicity with the real world based on its boundary definitions.

The bigger question, which will not be answered in this paper, is if Artificial Intelligence (as in conscious awareness) is indeed possible. Most of my programmer-friends consider the term an oxymoron. I may quote a comment by David Gelernter, professor of computer science at Yale University, that we would not expect our laptop to perform photosynthesis in the same manner that we are not expecting computers to become conscious - they are simply not made out of the ‘right stuff’ (David Gelernter/ DREAM-LOGIC, THE INTERNET AND ARTIFICIAL THOUGHT). If consciousness is an intrinsic quality of a living organ, our brain, then we would not be able to simulate consciousness. David Chalmer’s ‘hard problem of consciousness’ may turn out to be a hard fact. I shall close with this little thought-provoking statement.

Summary

From what has been elaborated Computational Simulation as an indirect and bounded verification for inductive inferences needs to understood as a) ‘indirect’ - since the virtual world is only a coarse-grained simulation and translation of the physical world b) ‘bounded’ - since CS is set within the limits of boundary conditions leading to a partial but never complete congruence with the physical world c) ‘deductive verification’ - since the deductive computer-model predicts the coherency of inductive real-world observations. The advantage of deriving at more realistic priors has been pointed out, allowing for a more rigorous testing of additional assumptions as set out in the Duhem-Quine problem.

Popper’s criterion of falsification only discredits a false hypothesis or theory; it does not create additional knowledge. Computational simulation however does create new knowledge, in particular knowledge of future states while we have to keep their limited truth-range in mind.

Referring back to Hume, the daring claim is to suggest that induction can not only be validated but can also, at least indirectly, be vindicated within a cross-referenced analogue-digital model since we can give a full account how a mixed methodology achieves its goal. To correct one’s own assumptions and to continuously adjust one’s model of our world starts with heuristic guesswork, necessary errors and we may end up with a well-calibrated working model that allows predictions with certainty, e.g., in a Bayesian network. In the meantime, instead of waiting for the magic formula of knowledge: what is wrong with learning by mistakes?
Graphic: The relation of inductive and deductive inferences

Induction (from the specific to the general) works only towards a certain point towards the future based on current evidence at hand. Deduction (from the general to the specific) needs to relate to facts and can only reach to a certain point successfully explaining the past. Both methods will necessarily run into prediction errors when extrapolated.