The Philosophy of Information
A Methodological Point of View

Gian Maria Greco, Gianluca Paronitti, Matteo Turilli, and Luciano Floridi

Information Ethics Group, Oxford University Computer Laboratory,
Oxford, United Kingdom

1 Introduction

The Philosophy of Information is a new area of research at the intersection of philosophy and computer science [4]. It concerns (a) the critical investigation of the conceptual nature and basic principles of information, including its dynamics (especially computation), utilization (especially computer ethics) and sciences; and (b) the elaboration and application of computational and information-theoretic methodologies to philosophical problems. Past work by members of our group has concentrated on (a), and in this paper we explore (b). In a nutshell, we ask what computer science can do for philosophy, rather than what the latter can do for the former.

Applications of computational methods to philosophical issues may be approached in three main ways:

1. *Conceptual experiments in silico*, or the externalization of the mental theater. As Patrick Grim has remarked "since the eighties, philosophers too have begun to apply computational modeling to questions in logic, epistemology, philosophy of science, philosophy of mind, philosophy of language, philosophy of biology, ethics, and social and political philosophy. [...] A number of authors portray computer experimentation in general as a technological extension of an ancient tradition of thought experiment" [10].
2. *Pancomputationalism*, or the fallacy of a powerful metaphor. According to this view, computational and informational concepts are so powerful that, given the right Level of Abstraction (see section 3), anything could be presented as a computational system, from a building to a volcano, from a forest to a dinner, from a brain to a company, and any process could be simulated computationally heating, flying and knitting. Even non-computable functions would be representable, although by abstracting them to such a high level that they would no longer count as a system (one would have to abstract output and even termination and the existence of output, but a system has to be allowed to terminate or not, even if one does not observe the output). But then pancomputationalists (e.g. [2]) have the hard task of providing a credible answers to the following two questions: how can one avoid blurring all differences among systems, thus transforming pancomputationalism into a night in which all cows are black, to paraphrase Hegel? And what would

it mean for the system under investigation not to be an informational system (or a computational system, if computation = information processing)? Pancomputationalism does not seem vulnerable to a refutation, in the form of a possible counterexample in a world nomically identical to the one to which pancomputationalism is applied.

3. *Regulae ad directionem ingenii*, or the Cartesian-Kantian approach. Are there specific methods in computer science that can help us to approach philosophical problems computationally?

In the following sections we answer this last question by introducing three main methods: *Minimalism*, the *Method of Abstraction* and *Constructionism*. Each one is discussed in a separate section.

2 Minimalism

Philosophical questions pose multi-faceted problems. According to Descartes, a problem space can be decomposed by a divide-and-conquer approach. The outcome is a set of more approachable sub-problems, interconnected in a sort of Quinean web of dependencies. When dealing with a philosophical question, the starting problem often presupposes other open problems and the strength of the answer depends on the strength of the corresponding assumptions. A minimalist starting problem relies as little as possible on other open problems, thereby strengthening the final answer to the philosophical question.

Philosophers may improve the tractability of a problem space by choosing discrete systems with which it can be studied. Minimalism outlines three criteria to orientate this choice: *controllability*, *implementability* and *predictability*. A system is controllable when its structure can be modified purposefully. Given this flexibility, the system can be used as a case study to test different solutions for the problem space. The second minimalist criterion recommends that systems be implementable physically or by simulation. The system becomes a white box the opposite of a black box (see section 4). Metaphorically, the maker of the system is a Platonic "demiurge", fully cognisant of the components of the system and its state transition rules. The system can therefore be used as a laboratory to test specific constraints on the problem space. The third criterion follows from the previous two: the chosen system must be such that its behaviour is predictable. The demiurge can predict the behaviour of the system in that she can infer the correct consequences from her explanations of the system. The system outcomes become then the benchmarks of the tested solutions.

The following three elements characterise Minimalism. First, Minimalism is relational. Problems and systems are never absolutely minimalist, but always connected with the problem space posed by the philosophical question. Second, Minimalism provides a way to choose critically the starting problem for the analysis of a problem space, thus guaranteeing the strength of the next step in the forward process of answering the philosophical question. According to a minimalist approach, the tractability of a philosophical problem is a function of the

three criteria outlined above. They allow the use of dynamic systems to test possible solutions and to derive properties of the problem space. Finally, Minimalism is a matter of inferential relations between a problem and its space, but it is not a way to privilege simple or elementary problems. Minimalist problems may be difficult or complex.

Minimalism is an economic method that may be confused with Ockham's razor. The two methods are compatible, but while Ockham's Razor avoids inconsistencies and ambiguities by eliminating redundant explicative or ontological elements in a theory, Minimalism provides a set of criteria for choosing problems and systems relative to a given specific question. Moreover, Ockham's principle of parsimony is absolute and is applied to any theoretical element, while Minimalism's main maxims of strength and tractability are always relative to a given problem space.

A practical example of Minimalism applied to the philosophy of perception may be helpful:

1. *The identification of the question.* We begin by asking e.g. "what is visual perception?". This question poses a wide problem space, hitherto approached with different methods.
2. *The Cartesian decomposition of the problem followed by a Quinean construction of the problem space.* Some well-known sub-problems of this problem space are the nature of internal representations, the role of mind in perception, vision as computation.
3. *The identification of the starting problem.* The standard representational interpretation of perception is rich in assumptions about open problems. Perception is based, for example, on the presumed existence of internal representations. The sensorimotor approaches to visual perception are less demanding. Perception is chained to action while information is externalised. James Gibson [9], one of the main advocates of the sensorimotor hypothesis, cannot explain the nature of perceptual errors. This problem does not rely on other open problems and therefore can be assumed as a minimalist starting problem. We shall label it the "Gibson problem".
4. *The selection of the system to be used to study the starting problem.* This system has to be consistent with the requirements of Gibson's sensorimotor theory and with the criteria for Minimalism. The subsumption architecture, proposed by Rodney Brooks [1], fits these requirements. The architecture of Brooks' robots is reactive, parallel and decentralised. Perception and action are directly connected without any explicit internal representation or centralised inferential engines. Moreover, subsumption architectural behaviour is fully specified by the topological structure of its layers composed of single behavioural units. Its demiurge has full control and predictability power over the system she has built. The Gibson problem can therefore be studied by means of Brooks' mobots.
5. *The solution of the problem.* In the sensorimotor approaches to vision, seeing is something done by agents in their environments. The definition of perceptual errors must be shifted from a representational interpretation errors are

wrong computations made over internal representations to an action-based interpretation errors are unsuccessful actions made by agents in their environment. If the mobot's sensorimotor features enable it to move randomly in its environment then perception is successful, otherwise its perception is erroneous. The mobot that bumps against a window lacks either the right features or the sensorimotor capabilities relative to a given specific environment and its task of moving around randomly.

3 The Method of Abstraction

The process of making explicit the Level of Abstraction at which a system is considered is called Method of Abstraction [8]. This epistemological method applies both to conceptual and phenomenological systems and it is based on the key concept of Level of Abstraction (LoA).

The metaphor of interface in a computer system is helpful to illustrate what a LoA is. We all know that users seldom think about the fact that they use a variety of interfaces between themselves and the real electro-Boolean processes that carry out the required operations. An interface may be described as an intra-system, which transforms the outputs of system A into the inputs of system B and vice versa, producing a change in data types. LoAs are comparable to interfaces because:

1. they are a network of observables;
2. the observables are related by behaviours that moderate the LoA and can be expressed in terms of transition rules;
3. they are conceptually positioned between data and the agents' information spaces;
4. they are the place where (diverse) independent systems meet, act on or communicate with each other.

LoAs can be connected to form broader structures of abstraction, from hierarchy of abstractions to nets of abstraction. One of the possible relations between LoAs is simulation. A simulation relation [13] is the relation between the observables of a simulator system and a simulated one. This relation must occur between pairs of observables in order to guarantee a satisfactory degree of congruence not only for the current state of the two systems but also for their evolution. In the simulation relation, the epistemic agent is coupling the state evolution of two systems by observing these two systems at different LoAs. This means that an epistemic agent tries to construct an equivalence relation between the two systems, seeking to understand at what LoA those systems could be considered congruent.

As example, let us now apply the Method of Abstraction and the simulation relation in order to re-define functionalism. Functionalism argues that a physical or abstract entity is identified by its causal or operational role. From this viewpoint, a system is not evaluated by its structures and their interactions, but

rather by the functions it shows. If the "matter" constituting a system is irrelevant for its identification, then the same functional organization can be realized by different systems and substrates, which are usually called realizations [12]. This is the *multi-realizability thesis*.

Some philosophers try to rule out multi-realizability from the functionalist approach. They argue that multi-realizability could lead to a weakening of a neuroscientific approach in the explanation of human behaviour. Why bother with actual neural structures if one can execute an algorithm to instantiate the same behaviours shown by these neural structures? It is argued that a computational approach is therefore more suitable for processing those algorithms.

Unfortunately, multi-realizability cannot be disconnected from functionalism since, without it, functionalism becomes inexplicable. This is clear if we consider the mathematical concept of function. A function is usually expressed by an operation on one or more variables. The well-known scheme is $f(x) = y$, but this simply means that the variables in the equation could be realized by an infinite class of numbers or by points over the Cartesian plane or by means of a Turing machine or by set theory. Without all these instantiations, it would be impossible to explain the function $f(x) = y$. We shall therefore conclude that functionalism entails multi-realizability.

Now, in the classic account of functionalism we deal with relata (the *functional organization* and the *realizations*) and relations (the *realization relation* between the functional organization and the realizations, and the *simulation relation* between the various realizations).

Our goal is to show that realization and simulation are equivalent. One can say that an epistemic agent can observe any functional organization at a specific LoA and the realization of that functional organization at another LoA. Then the realization relation between the two LoAs is characterized by: (a) the codification of the inputs of the functional organization LoA into the inputs of the various realizations LoAs, and (b) the de-codification of the outputs of the latter into the outputs of the former. Basically, simulation relation and realization relation are equivalent because they are relations which describe the same processes.

Given that multi-realizability and functionalism are coupled concepts and that a simulation relation is equivalent to a realization relation, it follows that a common functional organization does not exist at a higher LoA than that of its realizations. The functional organization is the net of abstraction constructed by the epistemic agents with the simulation relation between the various realizations conceived at different LoAs. This means that a functional organization is the relational structure produced by various realizations and by the simulation relation that connects them. For example, a carpenter who is making a piece of furniture by following a blueprint is not handling a functional organization (the blueprint) and a realization (the piece of furniture), but two realizations at different LoAs, which are related in a simulation relation specified by his work.

This new interpretation of functionalism leads us to reconsider functionalistic explanations within the philosophy of AI and the philosophy of mind by introducing simulation relation as a new player. The functionalistic explanation

is configured as a specification of simulations between the LoAs at which the realizations are disposed by the epistemic agent.

4 Constructionism

A black box is a system whose internal structure, rules and composition remain undisclosed. A white box is a system about which one knows everything, because one has constructed it. This perspective lays in the wake of the so-called *maker's knowledge tradition*, according to which: (a) one can only know what one makes and therefore (b) one cannot know the genuine nature of reality in itself. Philosophers who stress (b) argue that, since any attempt to know the intrinsic nature of the world will inevitably fail, it is better to concentrate on those sciences whose subject is created by us, such as politics and social sciences. Philosophers who stress (a) argue that it is possible to improve our knowledge of reality through the improvement of our knowledge of the techniques by which reality is investigated. This tradition finds its champion in Francis Bacon's philosophy of technology. Following Bacon, technology becomes the main subject of philosophical enquiry, because it is both a human product and the means through which the world is investigated. Constructionism explicitly refers to the maker's knowledge tradition. Its method consists of the following five principles:

1. The *Principle of Knowledge*: only what is constructible can be known. Anything that can not be constructed could be subject, at most, to a working hypothesis.
2. The *Principle of Constructability*: working hypotheses are investigated by (theoretical or practical) simulations based on them.
3. The *Principle of Controllability*: simulations must be controllable.
4. The *Principle of Confirmation*: any confirmation or refutation of the hypothesis concerns the simulation, not the simulated.
5. The *Principle of Economy*: the fewer conceptual resources used, the better. In any case, the resources used must be fewer than the results accomplished.

Constructionism suggests that, given a theory, one implements and tests it in a system. Because one constructs the system, one can also control it. As Newell and Simon remarked "neither machines nor programs are black boxes; they are artefacts that have been designed, both hardware and software, and we can open them up and look inside" [11] (for constructionist approaches in Cybernetics and proto-Cybernetics see [3]). Consider for example behaviour-based robotics. One may observe an ant and offer a hypothesis about its internal structures in order to explain its behaviours. Then one may build a system to test that hypothesis. The resulting system is controllable in that it is *modifiable*, *compositional* and *predictable*. This means that, as far as the constructed system is concerned, one can change its internal structures and rules; the system can be implemented by adding or removing new parts; and since one knows the rules of the system, one can know its behaviour. Suppose that the mobot we have constructed behaves like an ant. The Principle of Confirmation prevents us from

generalizing the working hypotheses, as if the simulation were the real cause (or internal structure) of the simulated. From this, the *sub-Principle of Context-dependency* is derived: isomorphism between the simulated and simulation is only local, not global. The mobot accounts for the behaviour of the ant only under the constraints specified by the simulation. If the constraints change, so does the evaluation of the hypotheses.

Constructionism is in plain contrast to any mimetic approach in epistemology. The latter assumes that reality is approached through some reproductive or representational mechanism: ideas, mental images, corresponding pictures, concepts and so forth are copies or portraits of some otherwise mysterious reality in itself. Constructionism, on the contrary, considers knowledge a modelling process, which shapes and edit reality to make it intelligible. It therefore rejects mimetic theories such as Plato's, Descartes' or Locke's. The Principle of Economy refers to the "careful management of resources". On the one hand, in defining knowledge processes, mimetic theories use a large amount of resources. Assuming that there is a reality and that it works in some particular way means making a heavy ontological commitment. On the other hand, Constructionism does not state anything about reality in itself. A more modest commitment makes errors less likely.

The Turing Test (TT) is an enlightening example of how the methodology outlined in this paper and, more specifically, the constructionist method work, for it respects the minimalist criterion, uses the LoAs and is constructionist. Turing refuses even to try to provide an answer to the question "can a machine think?". He considers it a problem "too meaningless to deserve discussion" [14], because it involves vague concepts such as machine' and thinking'. Turing suggests replacing it with the imitation game, which is exactly more manageable and less demanding from the minimalist point of view. By so doing, he specifies a LoA and asks a new question, which may be summed up thus: "can we consider that a computer is thinking, *at this Level of Abstraction*?". The rules of the game define the conditions of observability [8]. If we observe the behaviour under those conditions, we can accept an operational definition of a thinking machine for that LoA. By changing the rules of the game, one changes the LoA and consequently the answer. Note how TT respects the constructionist principles:

1. By satisfying Minimalism, Turing also respects the Principle of Knowledge.
2. Turing makes a hypothesis based on the common assumption that conversation skills require intelligence, and then he devises a system to evaluate whether a machine is intelligent comparatively.
3. The system is controllable. We know how it works and how it can be modified.
4. Whether a machine passes the test implies only that the machine can, or cannot, be considered intelligent at that LoA.
5. Finally, in tackling the problem of artificial intelligence, Turing refuses to consider those ways requiring a large amount of conceptual resources. This is, for instance, why he refuses to deal with any psychological assumption about intelligence.

5 Conclusion

In this paper, we have introduced three methods and shown how they can be imported from computer science into philosophy, in order to model and analyse conceptual problems. We have outlined their main features and advantages. The methods clarify implicit assumptions, facilitate comparisons, enhance rigour and promote the resolution of possible conceptual confusions. Some applications of the methods discussed in this paper have already been successfully provided in computer ethics [7], in epistemology [5], and in the philosophy of information [6]. Of course, the adoption of the methods raises important further questions. We mention only three of them that seem to us particularly pressing: (a) What is the logic of problem spaces? (b) What are the logical relations between LoAs? (c) How can Constructionism avoid solipsism? We have not attempted to answer these questions, which we hope to address in future work*.

References

1. Brooks, R.: Intelligence without Representation. Artificial Intelligence **47** (1991) 139-159.
2. Chalmers, D. J.: The Conscious Mind: In Search of a Fundamental Theory. Oxford University Press, Oxford (1996)
3. Cordeschi, R.: The Discovery of the Artificial. Behavior, Mind and Machines Before and Beyond Cybernetics. Kluwer, Dordrecht (2002)
4. Floridi, L.: What Is the Philosophy of Information? Metaphilosophy **1-2** (2002) 123-45
5. Floridi, L.: On the Logical Unsolvability of the Gettier Problem. Synthese **142.1** (2004a) 61-80
6. Floridi, L.: Open Problems in the Philosophy of Information. Metaphilosophy **4** (2004b) 554-82
7. Floridi, L., Sanders, J. W.: On the Morality of Artificial Agents. Minds and Machines **3** (2004a) 349-79
8. Floridi, L., Sanders, J. W.: The Method of Abstraction. In Lang, P. (ed.): Yearbook of the Artificial. Issue II: Models in contemporary sciences (2004b) 177-220
9. Gibson, J. J.: The ecological approach to visual perception. Houghton Mifflin, Boston (1986).
10. Grim, P.: Computational Modeling as a Philosophical Methodology. In Floridi, L. (ed.): The Blackwell Guide to the Philosophy of Computing and Information. Blackwell, Oxford (2004)
11. Newell, A., Simon, H. A.: Computer Science as Empirical Enquiry: Symbols and Search. Communications of the ACM **3** (1976) 113-126
12. Putnam, H.: Psychological Predicates. In Captain, W. H., Merrill, D. D.: Art, Mind and Religion. Pittsburgh University Press, Pittsburgh (1967)
13. Roever, W.-P. de and Engelhardt, K.: Data Refinement: Model-Oriented Proof Methods and their Comparison. Cambridge University Press, New York (1998)
14. Turing, A. M.: Computing Machinery and Intelligence. Mind **49** (1950) 433-460

* For Italian legal requirements, Gianluca Paronitti must be considered the author of section 3, Matteo Turilli of section 2, Luciano Floridi of sections 1 and 5, Gian Maria Greco of section 4 and the first author of the whole paper. We wish to thank Jeff Sanders for all his fundamental input when discussing previous versions of this paper.