How to Do Philosophy Informationally

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

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

Abstract. In this paper we introduce three methods to approach philosophical problems informationally: Minimalism, the Method of Abstraction and Constructionism. Minimalism considers the specifications of the starting problems and systems that are tractable for a philosophical analysis. The Method of Abstraction describes the process of making explicit the level of abstraction at which a system is observed and investigated. Constructionism provides a series of principles that the investigation of the problem must fulfil once it has been fully characterised by the previous two methods. For each method, we also provide an application: the problem of visual perception, functionalism, and the Turing Test, respectively.

1 Introduction

The Philosophy of Information is a new area of research at the intersection of philosophy, computer science and ICT (information and communication technology) [6] and [8]. It concerns (a) the critical investigation of the conceptual nature and basic principles of information, including its dynamics (especially computation), utilization (especially ethical issues) 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). 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 theatre. 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" [12].
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 and flying, eating or 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, although a system has to be allowed to terminate or not, even if one does not observe the output). Pancomputationalists (e.g. [3]) have the hard task of providing a credible answers to the following two questions: (1) 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 (2) 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 (to put it in Popperian terms), 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 separately

2   Minimalism

Philosophical questions pose multi-faceted problems. According to Descartes, a problem space can be decomposed by a divide-and-conquer approach [19]. The outcome is a set of more approachable sub-problems, interconnected in a sort of Quinean web of dependencies [18]. 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 to study it. *Minimalism* outlines three criteria to orientate this choice: *controllability*, *implementability* and *predictability*. Each deserves a brief comment.

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 of 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 should be predictable, at least in principle. 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.

Three properties further characterise Minimalism as we advocate it.

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 related to, but not to be confused with, Ockham's razor. The two methods are of course compatible. However, whilst 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 [13].

A practical example of Minimalism applied to the philosophy of perception may be helpful. Suppose our investigation concerns the nature of visual perception. We start from

1. *The identification of the question.* One begins by asking "what is visual perception?". This question poses a wide problem space, hitherto approached with different methods [15].

2. *The Cartesian decomposition of the problem is followed by a Quinean construction of the problem space.* One proceeds by identifying some well-known sub-problems of this problem space, such as the nature of internal representations, the role of mind in perception, and the interpretation of 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 [10], 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. It can be referred to as "Gibson problem".

4. *The selection of the system to be used to study the starting problem.* The system selected 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 [2], satisfies 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, according to which errors are incorrect computations made over internal representations, to an action-based interpretation, according to which errors are unsuccessful actions performed by agents in their environment. If the sensorimotor features of the mobot enable it to move randomly in its environment then perception is successful, otherwise its perception is erroneous. The mobot that collides 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.

We have just seen how Minimalism orientates the choice of problems and systems that are tractable. The answer to the initial question "what is visual perception?" is reached via the solution of localized problems. The minimalist choice of the problem — and possibly of the system — is reiterated to ensure the controllability of the whole inferential process of finding an answer to the initial philosophical question.

The definition of Minimalism is based on two main assumptions. The first concerns the existence of a problem space. Minimalism does not give an account of the decomposition process of the problem space to which it applies. The second assumption is that a dynamic system, whether conceptual or physically implemented, is useful in finding the answer to the investigated question. Minimalism does not explain why and how this approach works. In the next sections, two other methods are presented to ground both these assumptions: the *Method of Abstraction* and *Constructionism.*

The Method of Abstraction is used to describe the observables that compose the problem space and how they are related. Minimalism is always relative to a given Level of Abstraction as the structure of the problem space depends on the LoA assumed by the investigator.

Constructionism is the precise answer to the need to clarify how to choose the system and how to use it in order to investigate the set of minimalist problems that will lead to the required answer.

3   The Method of Abstraction

The process of making explicit the Level of Abstraction (LoA) at which a system is considered is called *Method of Abstraction* [9]. This method pertains to the analysis of discrete systems, by which we mean those systems whose evolution is described by explicit transition rules. It applies both to conceptual and to physical systems. Its pivotal element is the concept of LoA.

The terminology and the study of the method are rooted in a branch of theoretical computer science known as Formal Methods. Intuitively, Formal Methods are a collection of mathematical techniques used to prove that the concrete code implementation fits the abstract specifications of a computer system [23]. More precisely, Formal Methods are a variety of mathematical modelling techniques used to specify and model the behaviour of a computer system and to verify, mathematically, that the system design and implementation satisfy functional requirements.

The metaphor of interface in a computer system is helpful to illustrate what a LoA is[1]. As is well known, most users seldom think about the fact that they employ a variety of interfaces between themselves and the real electromagnetic and 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 together to form broader structures of abstraction, from hierarchy of abstractions to nets of abstraction. One of the possible relations between LoAs is that of simulation.

Traditionally, a simulation is considered a dynamical representation of a system. This means that, if one wishes to produce a simulation, one must extract a model, by selecting some variables from the investigated system; and then construct an update function, which lets the variables in the simulator change as if they were the variables observed in the system [20]. In a nutshell, a simulation is considered the observation of a model that evolves over time. Such definition, though correct, is still imprecise because, in order to understand what a simulation is, one also needs to clarify explicitly and precisely what a model is. This clarification is currently one of the most controversial issues in the philosophy of science, and it is far from clear how one may best deal with it. However,

---

[1] For a more technical and in-depth introduction the reader should refer to [9].

using the Method of Abstraction it becomes possible to characterize the notion of simulation in a different way and hence bypass this difficulty.

A simulation relation is now the relation between the observables of a simulator system and a simulated one [21]. 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. By way of explanation, let us consider a simple example. Let us apply the Method of Abstraction and the simulation relation to a new definition of *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 "stuff" 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 [17]. This is the *multi-realizability thesis*.

Some philosophers try to rule out multi-realizability from the functionalist approach [1]. For example, they argue that multi-realizability could lead to a weakening of a neuroscientific approach in the explanation of human behaviour. For why should one be concerned with the 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.

Multirealizability cannot be detached 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 being $f(x) = y$. However, this simply means that the variables in the equation could be replaced or interpreted by an endless set 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. 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. The argument is then that:

1. multi-realizability and functionalism are coupled concepts, and
2. a simulation relation is equivalent to a realization relation.

   But then it follows that

3. a common functional organization does not exist at a LoA higher than 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 it 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 device. The functionalist explanation is configured as a specification of simulations between the LoAs at which the realizations are organised by the epistemic agent.

4   Constructionism

Providing the guidelines for choosing a problem and supplying a method for observing and analysing it are two of the fundamental steps in the informational treatment of philosophical problems. In order to be complete and sound, the general methodology must also give an account of how the problem must be investigated once it has been fully delineated. We refer to this method as *Constructionism*.

Constructionism is at the core of the epistemological theory proposed by PI. As for *Minimalism* and the *Method of Abstraction*, Constructionism finds its roots in both the philosophical tradition and computational theories and practices.

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, well known in Computer Science and Artificial Intelligence, lays in the wake of the so-called *maker's knowledge tradition*, according to which:

1. one can only know what one makes, and therefore
2. one cannot know the genuine nature of reality in itself.

Like Vico and Hobbes [4], philosophers who emphasise (2) 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 emphasize (1) 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 [16] and it is related to Kantism. 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. And when Kant stresses the importance of understanding the conditions of possibility of our knowledge, he is working within the maker's knowledge tradition. One can investigate scientifically the phenomena one experiences only insofar as one is epistemically responsible for them.

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 working hypotheses.
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 the 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" [14] (for constructionist approaches in Cybernetics and proto-Cybernetics see [5]).

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: (a) one can change its internal structures and rules; (b) the system can be implemented by adding or removing new parts in order to test a variety of hypotheses; and since one knows the rules of the system, (c) one can know its behaviour. Suppose that the mobot one has built behaves like the ant one observed. The Principle of Confirmation prevents her from generalizing the working hypotheses, as if they were the real cause (or internal structure) of the simulated. It is obviously possible to provide an endless number of simulations with different internal structures whilst still obtaining the same behaviour. 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 supposed to be mere copies or portraits of some otherwise mysterious reality in itself. From the constructionist point of view, on the contrary, knowledge is a modelling process, which shapes and edits reality to make it intelligible. It therefore rejects more "mimetic" theories such as Plato's, Descartes' or Locke's, in favour of a more Kantian approach. 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. This more modest commitment makes errors less likely. As in the case of Minimalism, the constructionist Principle of Economy differs from Ockham's razor too. While the latter is a post-production revision tool, for it provides a criterion for choosing among theories already produced, the former is a pre-design planning norm, for it requires to be fully aware of the initial assumptions before one undertakes the investigation process and it binds the construction of any explanatory model to the conceptual resources available.

The Turing Test (TT) is an enlightening example of how the methodology outlined in this paper and, more specifically, the constructionist method, work, since it respects the minimalist criterion, uses the LoAs and is constructionist.

As we saw earlier, Minimalism concerns the choice of the problem, not of the explanation. 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", because it involves vague concepts such as 'machine' and 'thinking' [22]. 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 one consider that a machine is thinking *at this Level of Abstraction*?". The rules of the game define the conditions of observability [9]. If the machine passes the test at that LoA, one can clearly state that the machine is thinking, at 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, that is, by refusing to provide a universal definition of intelligence, 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, that means at that LoA (Principle of Constructability).

3. The system is controllable. One knows how it works and how it can be modified, so it can be implemented to test other features such as creativity, learning, and ethical behaviour (Principle of Controllability).
4. If a machine passes the test implies only that the machine can be considered intelligent at that LoA (Principle of Confirmation).
5. Finally, in tackling the problem of artificial intelligence, Turing refuses to avail himself of those approaches that require a large amount of conceptual resources. This is why, for instance, he refuses to deal with any psychological assumption about intelligence (Principle of Economy)[2].

A strongly constructionist approach grounds not only the design of TT but also what Turing conjectured as a potentially successful strategy to obtain a machine that would pass the test. In the final section of his 1950 paper, entitled "Learning Machines", Turing suggests, as a working hypothesis, that a child-machine could learn and gain its own knowledge through educational processes. Then he builds a system, the child-programme, in order to test that hypothesis. This system is controllable, for example through punishment and reward processes. Any hints or results in the process concern only the system and not how a human child learns. The choice itself of a child-machine, instead of a "grown up", respects not only the minimalist criterion but also the Principle of Economy.

5   Conclusion

In this paper, we have introduced three methods and shown how they can be imported, with some adaptations, 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 [9], in epistemology [7], and in the philosophy of information [8]. 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[3].

---

[2] The TT is anti-psychological in the classical Fregean sense.

[3] An earlier version of this paper was presented at NA-CAP@CMU 2004, the annual Computing and Philosophy meeting organised at Carnegie Mellon (4-6 August 2004). We wish to thank the participants in that meeting, especially Seth Casana, Charles Ess, Susan Stuart and John Taylor, as well as the participants in the WSPI 2005, and the two anonymous referees for their feedback. As usual, Jeff Sanders' input was fundamental in shaping our ides. This paper is a revised version of [11]. 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.

References

1. W. Bechtel and J. Mundale. Multiple realizability revisited: Linking cognitive and neural states. *Philosophy of Science*, 66:175–207, 1999.

2. R. A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47(1-3):139–159, 1991.

3. D. J. Chalmers. *The Conscious Mind: in Search of a Fundamental Theory*. Oxford University Press, Oxford, 1996.

4. A. Child. Making and knowing in hobbes, vico, and dewey. *University of California Publications in Philosophy*, 16(13):271–310, 1953.

5. R. Cordeschi. *The Discovery of the Artificial: Behavior, Mind, and Machines Before and Beyond Cybernetics*. Kluwer Academic Publishers, Dordrecht, 2002.

6. L. Floridi. What is the philosophy of information? *Metaphilosophy*, 33(1-2):123–145, 2002.

7. L. Floridi. On the logical unsolvability of the gettier problem. *Synthese*, 142(1):61–79, 2004.

8. L. Floridi. Open problems in the philosophy of information. *Metaphilosophy*, 35(4):554–582, 2004b.

9. L. Floridi and J. W. Sanders. The method of abstraction. In M. Negrotti, editor, *Yearbook of the Artificial. Nature, Culture and Technology. Models in contemporary sciences*, volume 2, pages 177–220. P. Lang, Bern, 2004.

10. J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, 1979.

11. G. M. Greco, G. Paronitti, M. Turilli, and L. Floridi. The philosophy of information - a methodological point of view. In K. D. Althoff, A. Dengel, R. Bergmann, M. Nick, and Th. Roth-Berghofer, editors, *WM2005: Professional Knowledge Management. Experiences and Visions*, pages 563–570. DFKI GmbH, Kaiserslautern, 2005.

12. P. Grim. Computational modeling as a philosophical methodology. In L. Floridi, editor, *The Blackwell Guide to the Philosophy of Computing and Information*, pages 337–349. Blackwell, Oxford, New York, 2003.

13. M. McCord Adams. *William Ockham*. University of Notre Dame Press, Notre Dame, 1987.

14. A. Newell and H. A. Simon. Computer science as empirical enquiry: Symbols and search. *Communications of the ACM*, 19(3):113–126, 1976.

15. A. Noé and E. Thompson, editors. *Vision and Mind: Selected Readings in the Philosophy of Perception*. The MIT Press, Cambridge, 2002.

16. A. Pérez-Ramos. *Francis Bacon's Idea of Science and the Maker's Knowledge Tradition*. Oxford University Press, New York, 1988.

17. H. Putnam. Psychological predicates. In W. H. Captain and D. D. Merrill, editors, *Art, Mind and Religion*. Pittsburgh University Press, Pittsburgh, 1967.

18. W. V. O. Quine. Two dogmas of empiricism. *The Philosophical Review*, 60(1):20–43, 1951.

19. A. Raftopoulos. Cartesian analysis and synthesis. *Studies in History and Philosophy of Science, Part A*, 34(2):265–308, 2003.

20. S. Rasmussen and C. L. Barrett. Elements of a theory of simulation. In F. Moràn, J. J. Moreno, P. Merelo, and P. Chacòn, editors, *ECAL 95. Advances in Artificial Life: Proceedings of the Third European Conference on Artificial Life*, pages 515–529. Springer-Verlag, Berlin, 1995.

21. W. P. de Roever and K. Engelhardt. *Data Refinement: Model-oriented Proof Methods and their Comparison*. Cambridge tracts in theoretical computer science. Cambridge University Press, Cambridge, 1998.

22. A. M. Turing. Computing machinery and intelligence. *Mind*, 49(236):433–460, 1950.

23. B. P. Zeigler. *Theory of Modelling and Simulation*. Wiley, New York, 1976.