Chapter 24
Computational Models in Science and Philosophy

Paul Thagard

Abstract Computer models provide formal techniques that are highly relevant to philosophical issues in epistemology, metaphysics, and ethics. Such models can help philosophers to address both descriptive issues about how people do think and normative issues about how people can think better. The use of computer models in ways similar to their scientific applications substantially extends philosophical methodology beyond the techniques of thought experiments and abstract reflection. For formal philosophy, computer models offer a much broader range of representational techniques than are found in traditional logic, probability, and set theory, taking into account the important roles of imagery, analogy, and emotion in human thinking. Computer models make possible investigation of the dynamics of inference, not just abstract formal relations.

Computer models are ubiquitous in the natural and social sciences, but are still rare in philosophy. This chapter will discuss the valuable contributions that such models make in the sciences and show how similar benefits can be gained in philosophy. Formal methods in philosophy have been limited to a relatively small set of tools such as predicate logic, set theory, and probability theory. But there are other branches of mathematics that are at least as relevant to central concerns in epistemology and metaphysics, including differential calculus, linear algebra, dynamic systems theory, and theory of computation. Computational models that draw on these kinds of mathematics can be highly valuable for understanding the structure and growth of knowledge and for grasping the nature of mind and reality.

P. Thagard (✉)
University of Waterloo, Waterloo, ON, Canada
e-mail: pthagard@uwaterloo.ca

© Springer International Publishing AG, part of Springer Nature 2018

S. O. Hansson, V. F. Hendricks (eds.), *Introduction to Formal Philosophy*, Springer Undergraduate Texts in Philosophy, https://doi.org/10.1007/978-3-319-77434-3_24

24.1 Computer Models in Scientific Applications

The development of digital computers and programs in the 1940s transformed many areas of science, starting with physics and later extending to biology, economics, cognitive psychology, and other fields. Physicists began to use computers to model the behavior of sub-atomic particles in nuclear fission and fusion ([13], ch. 8). To build bombs, physicists needed to understand how neutrons fission and scatter, but detailed experiments were not feasible and mathematical theory generated unsolvable equations. Hence John von Neumann and others employed the new tool of vacuum tube computers to recreate physical processes by modeling a sequence of random scatterings using what came to be called *Monte Carlo* methods. The differential equations in physical theory that assumed continuous quantities could be approximated by difference equations expressed in computer instructions. The new method replaced crude estimates of criticality by simulations that enable physicists to determine how detonations occur. Even the very primitive early computers could carry out calculations that would have taken humans hundreds of years. Now, some computers can perform quadrillions of operations per second, providing enormous capacity for simulating very complex systems.

Computer models are now widely used in fields of physics ranging from fluid dynamics to quantum mechanics [56]. Computational biology began in the 1960s and is now applied to many systems from cells to evolutionary development [23]. With the development of huge data bases in genomics and related fields, computers are used for bioinformatic purposes such as determining the function of model genes [18]. Computational chemistry is used to calculate the properties and changes of molecules and solids, with applications to the design of new drugs and materials [8]. Economists have long used computers to implement mathematical models of financial phenomena and are now turning to more realistic systems that model the interactions of somewhat intelligent agents (e.g. [4]). I will shortly give a more detailed account of the nature of computational models in science based on my own experience in developing models in cognitive psychology and neuroscience.

From the perspective of some traditional philosophical approaches, the use of computer models may seem puzzling. Consider the classical hypothetic-deductive method according to which theories consist of axiomatized hypotheses from which observations can be deduced. Why not just use mathematics to state the hypotheses and formal logic to deduce their consequences? There are many reasons why the logic-based version of hypothetico-deductivism is impractical.

First, scientific theories are rarely formalized so rigorously that deductions of the sort found in systems such as predicate logic can be made. Second, predicate logic is undecidable in the sense that there is no effective procedure for determining whether a formula is a consequence of a set of axioms. Third, more practically, theorists in physics and other fields have long known that calculating the consequences of their assumptions is mathematically very difficult. For example, it was already known in the eighteenth century that determining the motions of three bodies was difficult for Newtonian mechanics. Fourth, in the 1960s when computer models were newly used in meteorology, Edward Lorenz discovered that atmospheric systems are chaotic in that small differences in initial conditions can have very large effects in long-range predicted behavior. For these reasons, the logic-based view of hypothetico-deductive systems used to generate predictions and explanations in science does not capture well the actual practice of science. Computer models provide a powerful alternative to human deductions, generating valuable extensions to scientific methods [21].

I now give a more detailed description of how computer models work in science, drawing on my own experience building them for applications in psychology and neuroscience [47]. The methods I will describe are very common in the cognitive sciences, and are similar in many ways to how computer models operate in the natural and social sciences. I will note the relevant differences shortly. All computer models in science require ways of describing both conditions and changes. In physics, the conditions are usually represented by the values of variables, and the changes are represented by differential equations that describe how the values transform over time.

The first prominent computer model in psychology was the rule-based account of problem solving developed in the 1950s by Newell and Simon [26, 27], and this methodology expanded rapidly through the 1970s when cognitive science emerged as a recognized interdisciplinary field. I began building computer models in the 1980s in order to get a better understanding of analogical and other kinds of inference relevant to the discovery and acceptance of scientific theories [19, 20, 36]. The aim of computer modeling in psychology is to develop and test theories about how the mind works.

Since cognitive psychology supplanted behaviorism in the 1950s and 1960s, a psychological theory is an account of the structures and processes that enable minds to carry out such functions as perception, problem solving, learning, and language use. Candidate structures include propositions, concepts, images, Bayesian graphs, and neural networks [42]. Whereas many philosophers take propositions and concepts to be abstract entities, in cognitive science such structures are assumed naturalistically to be physical entities operating in brains and/or computers. Computer models of mind are different from computer models in physics and biology because of the fertile hypothesis developed in the 1950s that thinking is at least analogous to computation and perhaps more strongly is even a kind of computation. In contrast, computational models in physics and biology do not usually assume that entities such as atoms and non-neural cells are actually performing computations themselves.

Following the analogy between thinking and computing, mental structures can be modeled in computer programs via data structures, which are ways in which programming languages store and organize information for efficient use. Programming languages include a variety of data structures such as numbers, variables, strings, lists, and arrays. A high-level programming language such as LISP or Prolog contains extended ways of representing more complex information including propositions and concepts. Then a psychological theory about what kinds of representations the mind uses can be translated into a computer model with analogous kinds of data structures. A computer program is sometimes described just as a set of step-by-step instructions, but the instructions need to have data structures on which to operate, just as an inference needs propositions as well as rules of inference. Hence it is more accurate to describe computer programs and models as combinations of data structures and algorithms, which are effective methods expressed as finite steps of instructions.

It is surprisingly difficult to define more precisely what an algorithm is (see the Wikipedia article "Algorithm Characterizations"). For scientific purposes, algorithms are specified in order to capture changes taking place in the natural system being modeled. In the cognitive sciences, the algorithms specified serve to model the processes proposed in the psychological theory. For example, in rule-based psychological theories such as those of Newell and Simon [27] and Anderson [1], the algorithms specify how applying a rule to propositions can lead to inference to new propositions. This process is similar to use of modus ponens in formal logic, but much more complicated because many non-logical considerations such as past usefulness influence the algorithms that select what rules to fire. The data structures and algorithms of the computer program that implement the computational model correspond to the representations and processes that the psychological theory hypothesizes.

In computer modeling, it is important to distinguish between theories, models, and programs. Theories are general accounts of things, relations, and interactions that produce change. Computer models use data structures to characterize the things and relations, and use algorithms to capture the changes that result from the interactions. Computer programs are packages of code written in a specific language that implement the model and thereby provide a way of testing the theory. It is sometimes said that programs are theories, but programs contain myriad details particular to the programming language used. More accurately, programs implement models that approximate the claims made by theories. Cognitive scientists do not always move from theories to models to programs, because thinking about how to write a program in a familiar language can be a very useful way of developing ideas that can be used to specify models and programs. Computer modeling is a method for generating hypotheses as well as for testing them.

Psychological theories are not easy to test directly against experimental results, because their deductive implications are often unclear. When a theory, however, is specified in a model and implemented in a program, it becomes much easier to determine the implications of the theory. Unless the theory, model, and program are ridiculously simple, building a program and getting it to perform in psychologically realistic ways are highly non-trivial tasks. As the field of artificial intelligence has repeatedly found since its origin in the mid-1950s, computational implementation of functions such as perception and inference reveals unexpected difficulties. Some algorithms are computationally intractable in that the resources required increase exponentially with the size of the problem to be solved. For example, using truth tables to check for consistency in propositional logic is fine for very small numbers of sentences, but since n sentences require 2^n rows this method is not practical even on large computers for the millions of beliefs held by people and computer data bases. Hence implementing a theory in a running computer program provides preliminary evidence that the representations and processes postulated by the theory are physically realizable.

Given realizability, a computer program provides a way of testing a theory by examining whether the running program behaves in ways similar to how people behave in psychological experiments. There should be at least a qualitative fit between what the program does and what people do: the program performs roughly the same tasks in roughly the same ways. Ideally, there will also be a quantitative fit between program and human behavior, with statistics describing what the program does matching closely statistics generated in human experiments. Of course, even quantitative fit between program and experiments does not demonstrate that the original psychological theory is true, but it does provide some support. As in scientific theorizing in general, evaluation requires a full assessment of how well a theory compares to alternative theories in its ability to explain the full range of available evidence.

Computer modeling in the rule-based Newell and Simon tradition is still an important part of cognitive science, but it has been supplemented by approaches that more directly model the brain. In the 1980s neural networks models became prominent, also known by the terms *connectionist* (because they represent information by the connections between neurons) and *parallel distributed processing* [31]. These models are very different from rule-based and logic-based models in their data structures and algorithms. Instead of viewing problem solving and other cognitive tasks as a series of inferences applied to linguistic structures, neural network models adopt simpler data structures - artificial neurons and the links between them - and parallel algorithms that describe how activation (neural firing) spreads through populations of neurons. Current models in computational neuroscience are much more biologically realistic than connectionist models in employing much larger numbers of spiking neurons organized into populations that correspond to actual brain regions [7, 10–12, 28].

Although neural network models approach the mind very differently from the views of psychological operations found in folk psychology, formal logic, and rule-based systems, their use still fits with the general methodology I already described for computer modeling. Programs still consist of data structures and algorithms, although the structures are strange from the commonsense ones suggested by introspection and examination of written texts. Speech and writing are serial processes in which words, sentences, and inferences are generated one at a time using large structures such as concepts and propositions. From the perspective of computational neuroscience, concepts and propositions are patterns of activation in populations of thousands or millions of neurons (defenses and illustrations include [46, 47, 54]). For describing such patterns and exploring their operations, computer modeling is indispensable.

24.2 Philosophical Applications

Decades ago, Aaron Sloman [33] wrote that it was only a matter of time that any philosophers unfamiliar with computational modeling would be deemed incompetent! Currently, however, computer models are still rare in philosophy, although they have been used to study such topics as logic, causal reasoning, social evolution, ethical development, scientific reasoning and coherence. Specific examples will be provided below.

The key question is how computer models can be relevant to philosophical problems concerning the nature of knowledge, reality, and morality. On some views of philosophy, there would be no relevance. If the main goal of philosophy were to generate transcendent, a priori truths, then computer models would have little to contribute. Or if the main goal of philosophy were to analyze people's everyday concepts, then attention to language would obviate computer models. I think, however, that there are no significant a priori truths, and that philosophy should be like science in aiming to improve concepts rather than to analyze existing ones [45, 46, 48, 50]. Philosophy does not reduce to science, because its concerns have a degree of generality and normativity not found in any scientific field. But a naturalistic approach as pursued by Aristotle, Bacon, Locke, Hume, Mill, Peirce, Quine and many other philosophers, sees scientific results as highly relevant to philosophical issues, and hence opens the possibility that computational models might provide a useful philosophical methodology.

First consider epistemology. If one abandons as hopeless the traditional empiricist and rationalist goals to find an indubitable foundation for knowledge, then epistemology can reorient toward the much more interesting and accomplishable task of understanding the structure and development of knowledge. This task is very similar to the goal of cognitive psychology to understand how the mind/brain processes information about the world. Quine [30] also saw an alliance between epistemology and psychology, but was hampered by the theoretical and experimental limitations of the behaviorist psychology of his day. Current psychology has the intellectual resources to help address many key philosophical concerns about the nature of knowledge and inference. Here are some illustrations.

The main alternative to foundationalist epistemology is coherentism, according to which interlocking beliefs can be justified if they form a coherent set. Most philosophical discussions of coherence have only vaguely suggested how it can be objectively assessed. However, coherence can be made much more precisely calculable by considering it as a kind of constraint satisfaction problem of the sort naturally approached using neural network algorithms [37, 38, 40]. Moreover, coherence from this perspective can be formalized to an extent that enables proof that the problem of coherence is NP-hard, i.e. in a class of problems for which a guaranteed solution is unlikely to be found [55]. However, computer experiments show that connectionist and other algorithms can be used to model very large examples of scientific reasoning. Such modeling does not "prove" that coherentism is the best approach to epistemology, but it provides evidence that it can adequately characterize important aspects of belief evaluation.

The main alternative to coherentism in non-foundationalist epistemology is Bayesianism, which uses the tools of probability theory to analyze the structure and growth of knowledge. Merely assuming that probability theory provides answers to epistemological problems does not take one very far, but highly sophisticated computational tools for modeling Bayesian inference have been developed by philosophers, psychologists, and computer scientists (e. g. [14–16, 29, 34]). These computational tools have made possible the testing of Bayesian models as both accounts of actual human inference and as means of making accurate probabilistic inferences.

One advantage of formalizing philosophical ideas about inference in computational models is that it makes possible head-to-head comparison of their relative merits. For example, Thagard [41] compared coherence and Bayesian accounts of legal inference and argued that coherence is superior both descriptively and normatively. Epistemology, obviously, is concerned not just with the descriptive task concerning how people do think but also with the normative task of determining how people can think better. Normative concerns are not alien to science, as there are branches of applied science such as engineering and educational psychology that are as much concerned with improving the world as describing it. Computer models can contribute to normative deliberation by providing a means to explore the consequences of different ways of understanding the nature of knowledge. They are thus much more useful than thought experiments, in which philosophers' own intuitive reactions to stories they have made up are mysteriously used as evidence for the philosophers' preconceptions. As in science, computer models provide a link between theory and data, where the data can be actual cases of human knowledge development of the sort that occur in laboratory experiments and the history of science.

Computer models have other kinds of epistemological applications. For example, there is an old debate in the philosophy of science about whether there could be a "logic of discovery" [17]. This debate has been enriched by the development of various computer models of aspects of scientific discovery including generation of concepts, hypotheses, and descriptions of mechanisms (e.g. [3, 24, 39, 54]). Peirce's still-influential idea of abduction as a kind of inference involving both the generation and acceptance of explanatory hypotheses has been computationally explored using many techniques ranging from formal logic to neural networks (e.g. [22, 52]). Analogical inference can also be productively investigated using computational models [20, 35]. More traditional philosophical approaches involving formal logic can also be enhanced by computational modeling. In sum, computer modeling is as valuable a tool for epistemology as it is for cognitive psychology and other areas of science.

One might naturally suspect, however, that computer models are irrelevant to metaphysical questions about the fundamental nature of reality. As for epistemology, however, the potential arises within a naturalistic view of metaphysics that views it as continuous with science. For example, metaphysical questions about the nature of space and time might be informed by physical theories that are tested via computational models, although I do not know of any specific examples. But such models are clearly relevant to another central metaphysical question, the relation of mind and body.

Idealism, materialism, and dualism are the classic positions in the philosophy of mind. I think that evidence is rapidly mounting for a materialist resolution of the mind-body problem ([46, 48]; see also [2, 5, 6, 25]). Rather than pursuing inconclusive and prejudicial thought experiments, philosophers can examine evidence both for and against the hypothesis that mind events are brain events. This hypothesis is no different from many identity hypotheses that have come to be supported by large amounts of scientific evidence: water is H_2O, air is a mixture of gases, combustion is oxygenation, lightning is electricity, heat is motion of molecules, and so on. Support for mind-brain identity requires consideration of how well brain processes can explain the full range of psychological functions such as perception, inference, language, emotion, and conscience.

As my earlier discussion of computational neuroscience indicated, computer models are an important part of developing and testing neurocognitive theories. Philosophers can of course wait and watch for models most relevant to metaphysical concerns to be developed by scientists, but can accelerate progress by possessing the skills to build models themselves. For example, I had been investigating emotional thinking as a brain process [43], and was aware that conscious experience is a key part of emotion that according to some philosophers requires a non-materialist explanation. Hence I decided to develop a model of emotional consciousness, parts of which have been implemented computationally [49]. This model integrates two theories of emotion (cognitive appraisal and physiological perception) that have been taken as competitors by both philosophers and psychologists. Without computational tools that facilitate thinking of the brain as a parallel processor interconnecting both cognitive and bodily information, it would have been difficult to construct this model. By providing an evidence-based neural explanation of one important kind of consciousness, the model is highly relevant to the philosophical question of the relation between mind and body. Later work draws on new ideas from computational neuroscience to develop an improved theory of emotion [53].

I predict that further progress in computational neuroscience, along with rapidly growing evidence from brain scans and other experimental techniques, will provide further evidence for materialist metaphysics. Of course, those who favor dualism or idealism may see these developments as grounds for just ignoring scientific evidence and the computational models that connect them with theory. Ignorance is bliss.

Besides epistemology and metaphysics, the third major area of philosophy is ethics. Most computer modeling relevant to ethics has been performed by theorists interested in questions concerning the evolution of ethical strategies as modeled by game theory [9, 32]. I prefer a less abstract, more naturalistic approach to ethics that attempts to reach moral conclusions by developing coherent judgments about fundamental human needs [46, 48, 51]. From this perspective, moral intuitions are not a priori judgments achieving transcendent truths, but rather are the result of brain processes for emotional coherence. It follows that the model of emotional consciousness already described is highly relevant to understanding ethical judgments. The model provides a way of seeing how such judgments can be both cognitive and emotional, undercutting debates about emotivism that have exercised ethicists since the 1930s. Hence computer models can be highly relevant to ethical theory. Neuropsychological theories rooted in computational models can also be relevant to explaining puzzling ethical lapses such as conflicts of interest and self-deception [44].

In sum, computer models provide formal techniques that are highly relevant to philosophical issues in epistemology, metaphysics, and ethics. Such models can help philosophers to address both descriptive issues about how people do think and normative issues about how people can think better. The use of computer models substantially extends philosophical methodology beyond the timeworn techniques of thought experiments and abstract reflection.

For formal philosophy, computer models offer a much broader range of representational techniques than are found in traditional logic, probability, and set theory, allowing expansion to take into account the important roles of imagery, analogy, and emotion in human thinking. Just as significant, computer models make possible investigation of the dynamics of inference, not just abstract formal relations. Far from being oxymoronic, computational philosophy offers powerful new tools for investigating knowledge, reality, and morality.

References[1]

1. Anderson, J. R. (2007). *How can the mind occur in the physical universe?* Oxford: Oxford University Press.
2. *Bechtel, W. (2008). *Mental mechanisms: Philosophical perspectives on cognitive neuroscience*. New York: Routledge.
3. Bridewell, W., Langley, P., Todorovski, L., & Dzeroski, S. (2008). Inductive process modeling. *Machine Learning, 71*, 1–32.
4. Chen, S., Jain, L., & Tai, C. (2010). *Computational economics: A perspective from computational intelligence*. Hershe: Idea Group.
5. Churchland, P. M. (2007). *Neurophilosophy at work*. Cambridge: Cambridge University Press.
6. Churchland, P. S. (2002). *Brain-wise: Studies in neurophilosophy*. Cambridge, MA: MIT Press.
7. Churchland, P. S., & Sejnowski, T. (1992). *The computational brain*. Cambridge, MA: MIT Press.
8. Cramer, C. J. (2002). *Essentials of computational chemistry*. New York: Wiley.
9. Danielson, P. (1992). *Artificial morality: Virtuous robots for virtual games*. New York: Routledge.
10. Dayan, P., & Abbott, L. F. (2001). *Theoretical neuroscience: Computational and mathematical modeling of neural systems*. Cambridge, MA: MIT Press.
11. *Eliasmith, C. (2013). *How to build a brain: A neural architecture for biological cognition*. Oxford: Oxford University Press.

[1] Asterisks (*) indicate recommended readings.

12. Eliasmith, C., & Anderson, C. H. (2003). *Neural engineering: Computation, representation and dynamics in neurobiological systems*. Cambridge, MA: MIT Press.
13. Galison, P. (1997). *Image & logic: A material culture of microphysics*. Chicago: University of Chicago Press.
14. Glymour, C. (2001). *The mind's arrows: Bayes nets and graphical causal models in psychology*. Cambridge, MA: MIT Press.
15. Glymour, C. & Danks, D. (2008). Reasons as causes in Bayesian epistemology. *Journal of Philosophy, ADD*.
16. Griffiths, T. L., Kemp, C., & Tenenbaum, J. B. (2008). Bayesian models of cognition. In R. Sun (Ed.), *The Cambridge handbook of computational psychology* (pp. 59–100). Cambridge: Cambridge University Press.
17. Hanson, N. R. (1958). *Patterns of discovery*. Cambridge: Cambridge University Press.
18. Haubold, B., & Wiehe, T. (2006). *Introduction to computational biology: An evolutionary approach*. Basel: Birkhäuser.
19. Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. R. (1986). *Induction: Processes of inference, learning, and discovery*. Cambridge, MA: MIT Press/Bradford Books.
20. Holyoak, K. J., & Thagard, P. (1995). *Mental leaps: Analogy in creative thought*. Cambridge, MA: MIT Press/Bradford Books.
21. Humphreys, P. (2004). *Extending ourselves: Computational science, empiricism, and scientific method*. Oxford: Oxford University Press.
22. Josephson, J. R., & Josephson, S. G. (Eds.). (1994). *Abductive inference: Computation, philosophy, technology*. Cambridge: Cambridge University Press.
23. Kitano, H. (2002). Computational systems biology. *Nature, 420*, 206–210.
24. Langley, P., Simon, H., Bradshaw, G., & Zytkow, J. (1987). *Scientific discovery*. Cambridge, MA: MIT Press/Bradford Books.
25. McCauley, R. N., & Bechtel, W. (2001). Explanatory pluralism and the heuristic identity theory. *Theory & Psychology, 11*, 736–760.
26. Newell, A., Shaw, J. C., & Simon, H. (1958). Elements of a theory of human problem solving. *Psychological Review, 65*, 151–166.
27. Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs: Prentice-Hall.
28. O'Reilly, R. C., & Munakata, Y. (2000). *Computational explorations in cognitive neuroscience*. Cambridge, MA: MIT Press.
29. Pearl, J. (2000). *Causality: Models, reasoning, and inference*. Cambridge: Cambridge University Press.
30. Quine, W. V. O. (1969). *Ontological relativity and other essays*. New York: Columbia University Press.
31. *Rumelhart, D. E., & McClelland, J. L. (Eds.). (1986). *Parallel distributed processing: Explorations in the microstructure of cognition*. Cambridge MA: MIT Press/Bradford Books.
32. Skyrms, B. (1996). *The dynamics of rational deliberation*. Cambridge, MA: Harvard University Press.
33. *Sloman, A. (1978). *The computer revolution in philosophy*. Atlantic Highlands: Humanities Press.
34. Spirtes, P., Glymour, C., & Scheines, R. (1993). *Causation, prediction, and search*. New York: Springer-Verlag.
35. Steinhart, E. (2001). *The logic of metaphor: Analogous parts of possible worlds*. Dordrecht: Kluwer.
36. Thagard, P. (1988). *Computational philosophy of science*. Cambridge, MA: MIT Press.
37. Thagard, P. (1989). Explanatory coherence. *Behavioral and Brain Sciences, 12*, 435–467.
38. Thagard, P. (1992). *Conceptual revolutions*. Princeton: Princeton University Press.
39. Thagard, P. (1998). Computation and the philosophy of science. In T. W. Bynum & J. H. Moor (Eds.), *The digital phoenix: How computers are changing philosophy* (pp. 48–61). Oxford: Blackwell.
40. Thagard, P. (2000). *Coherence in thought and action*. Cambridge, MA: MIT Press.

41. Thagard, P. (2004). Causal inference in legal decision making: Explanatory coherence vs. Bayesian networks. *Applied Artificial Intelligence, 18*, 231–249.
42. Thagard, P. (2005). *Mind: Introduction to cognitive science* (2nd ed.). Cambridge, MA: MIT Press.
43. Thagard, P. (2006). *Hot thought: Mechanisms and applications of emotional cognition*. Cambridge, MA: MIT Press.
44. Thagard, P. (2007). The moral psychology of conflicts of interest: Insights from affective neuroscience. *Journal of Applied Philosophy, 24*, 367–380.
45. Thagard, P. (2009). Why cognitive science needs philosophy and vice versa. *Topics in Cognitive Science, 1*, 237–254.
46. Thagard, P. (2010). *The brain and the meaning of life*. Princeton: Princeton University Press.
47. *Thagard, P. (2012). *The cognitive science of science: Explanation, discovery, and conceptual change*. Cambridge, MA: MIT Press.
48. Thagard, P. (forthcoming). *Natural philosophy: From social brains to knowledge, reality, morality, and beauty*. Oxford: Oxford University Press.
49. Thagard, P., & Aubie, B. (2008). Emotional consciousness: A neural model of how cognitive appraisal and somatic perception interact to produce qualitative experience. *Consciousness and Cognition, 17*, 811–834.
50. Thagard, P., & Beam, C. (2004). Epistemological metaphors and the nature of philosophy. *Metaphilosophy, 35*, 504–516.
51. Thagard, P., & Finn, T. (2011). Conscience: What is moral intuition? In C. Bagnoli (Ed.), *Morality and the emotions* (pp. 150–159). Oxford: Oxford University Press.
52. Thagard, P., & Litt, A. (2008). Models of scientific explanation. In R. Sun (Ed.), *The Cambridge handbook of computational psychology* (pp. 549–564). Cambridge: Cambridge University Press.
53. Thagard, P., & Schröder, T. (2014). Emotions as semantic pointers: Constructive neural mechanisms. In L. F. Barrett & J. A. Russell (Eds.), *The psychological construction of emotions* (pp. 144–167). New York: Guilford.
54. Thagard, P., & Stewart, T. C. (2011). The Aha! Experience: Creativity through emergent binding in neural networks. *Cognitive Science, 35*, 1–33.
55. Thagard, P., & Verbeurgt, K. (1998). Coherence as constraint satisfaction. *Cognitive Science, 22*, 1–24.
56. Thijssen, J. M. (2007). *Computational physics*. Cambridge: Cambridge University Press.