Synthese (2018) 195:491–510
https://doi.org/10.1007/s11229-014-0532-0

Rational analysis, intractability, and the prospects of ‘as if’-explanations

Iris van Rooij · Cory D. Wright ·
Johan Kwisthout · Todd Wareham

Received: 17 December 2013 / Accepted: 24 July 2014 / Published online: 10 October 2014

Abstract The plausibility of so-called ‘rational explanations’ in cognitive science is often contested on the grounds of computational intractability. Some have argued that intractability is a pseudoproblem, however, because cognizers do not actually perform the rational calculations posited by rational models; rather, they only behave *as if* they do. Whether or not the problem of intractability is dissolved by this gambit critically depends, inter alia, on the semantics of the ‘as if’ connective. First, this paper examines the five most sensible explications in the literature, and concludes that none of them actually circumvents the problem. Hence, rational ‘as if’ explanations must obey the minimal computational constraint of tractability. Second, this paper describes how rational explanations could satisfy the tractability constraint. Our approach suggests a computationally unproblematic interpretation of ‘as if’ that is compatible with the original conception of rational analysis.

Keywords Psychological explanation · Rational analysis · Computational-level theory · Intractability · NP-hard · Approximation

I. van Rooij (✉) · J. Kwisthout
Donders Institute for Brain, Cognition, and Behaviour, Radboud Universiteit Nijmegen,
Nijmegen, The Netherlands
e-mail: i.vanrooij@donders.ru.nl

J. Kwisthout
e-mail: j.kwisthout@donders.ru.nl

C. D. Wright
Department of Philosophy, California State University Long Beach, Long Beach, CA, USA
e-mail: cory.wright@zoho.com

T. Wareham
Department of Computer Science, Memorial University of Newfoundland, St. John’s, NL, Canada
e-mail: harold@mun.ca

1 Introduction

1.1 Rational analysis

A central aim of the cognitive sciences is to explain certain human capacities, such as those for visual recognition, language learning, reasoning, or planning. A widely-used framework for casting such explanations was formulated by Marr (1982), who proposed that explanations of such capacities toggle between three different levels. At the first level—what Marr called the 'computational-level theory'—one addresses 'what'-questions: e.g., what is the function that the capacity ϕ realizes? An answer to such questions usually comes in the form of a well-defined input-output mapping $f : I \rightarrow O$.[1] Having postulated f, one can move to a second level—what Marr called the 'algorithmic-level theory'—and address functional 'how'-questions: e.g., how is f is being computed? An answer to such questions usually comes in the form of an algorithm A, i.e., a finite procedure that outputs $f(i)$ in a finite number of computational steps for any input $i \in I$. At the third and last level of explanation—what Marr called the 'implementational-level theory'—one addresses physical 'how'-questions: e.g., how is the algorithm A physically realized (e.g., by physiological processes in the human brain)?

Marr is widely known for promoting a top-down approach to explaining cognition. In particular, he believed that one may be in a better position to understand the 'how' of cognition, both physically and functionally, by first understanding the 'what' of cognition:

> an algorithm is likely to be understood more readily by understanding the nature of the problem being solved than by examining the mechanism (and the hardware) in which it is embodied. (1982, p. 27)

Marr's top-down explanatory approach is á propos for many contexts—not only in forward-engineering design projects of AI, but also for reverse engineering efforts in the cognitive sciences (Dennett 1994; Wimsatt 2007). It's also not without its challenges. For instance, even though fixing one's computational-level theory can indeed help constrain the space of possible 'how'-explanations of cognition, it can prove difficult to determine the function f that correctly characterizes some cognitive capacity ϕ. After all, for any limited set of observed input-output behaviors attributable to ϕ, there exist multiple functions $f_1, \ldots, f_n$ that are each consistent with the observed behaviors; and yet, only one generalizes the observations correctly. In other words, which function f best characterizes ϕ may be difficult if not impossible to determine on the basis of empirical input-output observations alone.[2]

[1] Note that Marr saw 'what'-questions as intimately tied to 'why'-questions: e.g., why is f the appropriate function for ϕ to realize? An answer to such questions would presuppose a specification of the conditions for appropriateness. We'll later return to this point when addressing rational explanation.

[2] Such cases of underdetermination of theory by evidence themselves do not determine any stronger antirealist conclusion. An alternative lesson to be drawn is just that cognitive scientists taking top-down explanatory approaches may benefit from theoretical constraints on computational-level theories (van Rooij 2008).

To attenuate the computational-level underdetermination problem, Anderson (1990, 1991a, b, c) proposed what became a highly influential framework for constraining the logical space of candidate computational-level theories. In particular, Anderson proposed that cognitive scientists can exploit a so-called *principle of rationality*: 'The cognitive system optimizes the adaptation of the behavior of the organism' (1991c, p. 3). This principle allows cognitive scientists to assume that cognitive capacities are characterized by functions that are optimized relative to agents' needs and their environments. As a general methodology for developing computational-level theories, Anderson proposed a six-step procedure, called *rational analysis*:

1. *Goal*: Specify precisely the goals of the cognitive system (goals G).
2. *Environment*: Develop a formal model of the environment to which the system is adapted (environment E).
3. *Computational limitations*: Make minimal assumptions about computational limitations (limitations L).
4. *Optimization*: Derive the optimal behavior function f given 1–3 above.
5. *Data*: Examine the empirical evidence to see whether the predictions of the behavior function are confirmed.
6. *Iteration*: Repeat, iteratively refine the theory.

The type of functions yielded by step 4 of this procedure are sometimes referred to as 'rational' or 'rational-level' explanations.[3]

1.2 The problem of intractability

The program of rational analysis has garnered significant enthusiasm and acceptance in cognitive science over the last two decades. In part, this is because it fecundly yields functions that adequately describe and predict human behavior in a wide diversity of cognitive tasks, spanning psychological domains such as word recognition (Norris 2006), cognitive control (Gray et al. 2006), language learning (Ellis 2006), categorization (Anderson and Matessa 1990), concept learning (Goodman et al. 2008), and memory retrieval and reasoning (Chater and Oaksford 1999). Unfortunately, there is a major wrinkle in the program: many of the functions postulated in rational analyses are, de facto, computationally intractable.

[3] A common presumption is that such fs not only characterize the 'what' of a capacity ϕ, i.e., f itself, but also provide the grounds for understanding why f is what characterizes ϕ. Of course, to presume that f is what ϕ does because f is optimal for the goals G of ϕ in environment E is not to claim that f itself explains why f is the function that characterizes ϕ. Rather, the 'why' explanation is a narrative that f is as it is because of the principle of rationality and the assumptions that ϕ has goals G and the environment of adaptation is E. Moreover, to genuinely be a 'why' explanation, rational analysts should not only show that f is optimal for G and E, but also that f is as it is *because* it is optimal for G and E (Danks 2008). Because this difference between f as a computational-level description of ϕ and the rational narrative surrounding and motivating f is seldomly made explicit, cognitive scientists are sometimes led to the mistaken idea that fs derived via rational analysis are rational explanations in the sense of explaining the 'why' of f. In our view, the distinction should be made explicit and the 'why' explanation should be differentiated from the computational-level f itself (cf. Danks 2013). After all, even if the rational narrative were falsified, f could still correctly characterize ϕ; the correctness of that characterization is independent of the truth-value of the rational story about how f came to be.

Formally proving a function f to be NP-hard is sufficient to demonstrate computational intractability.[4] A function's being NP-hard implies that there cannot exist any algorithm that can compute f in a realistic time (formally, polynomial-time) for all inputs $i \in I$.[5] In other words, all algorithms computing such an f require time that grows exponentially in the size of i, which makes for astronomically-long periods of computation for all but the smallest inputs. Consequently, discovering that f is NP-hard implies that f is no longer a plausible characterization of a capacity ϕ computing its outputs in fractions of seconds, or minutes at most. And since the vast majority of everyday human psychological capacities do just this, discovering that f is NP-hard effectively precludes f from being part of their psychological explanation.

Advocates of the program of rational analysis acknowledge that functions derived through the aforementioned procedure typically are computationally intractable. For instance, Nick Chater and Mike Oaksford write:

> Indeed, formal rational theories in which the optimization calculations are made, including probability theory, decision theory, and logic are typically computationally intractable for complex problems. Intractability results imply that no computer algorithm could perform the relevant calculations given the severe time and memory limitations of a 'fast and frugal' cognitive system. [...] Thus it might appear that there is an immediate contradiction between the limitations of the cognitive system and the intractability of rational explanations. (Chater and Oaksford 2000, pp. 109–110)

The computational intractability of f does seem to flatly contradict step 3 in the Anderson's procedure—i.e., the requirement of making minimal assumptions about computational limitations—which Anderson himself has called 'the true Achilles heel of the rationalist enterprise' (1991b, p. 473).

Faced with this problem of intractability, how should rational analysts proceed with the task of delivering good psychological explanations? One very natural response would be to opt for computational-level theory revision—more specifically, an iterative refinement of f so that it satisfies the minimal computational limitation of tractability (van Rooij 2008; van Rooij et al. 2012). Chater and Oaksford and colleagues, however, pursue a very different response. They simply amend Anderson's six-step procedure so that no response is needed, noting that 'sometimes 'minimal' assumptions will be no assumptions at all' (2003, p. 69). Effectively, this makes step 3 discretionary—even immaterial. Their rationale for this amendment is as follows:

> There is no contradiction, however, because the optimal behavior function is an explanatory tool, not part of an agent's cognitive equipment. Using an analogy from Marr (1982), the theory of aerodynamics is a crucial component of explaining why birds can fly. But clearly birds know nothing about aerodynamics, and

[4] NP-hard functions owe their name to being as hard as any function in the class NP, where 'NP' abbreviates '*N*ondeterministic *P*olynomial time'.

[5] This claim assumes P $\neq$ NP, a widely-endorsed conjecture in theoretical computer science (Garey and Johnson 1979; Fortnow 2009).

> the computational intractability of aerodynamic calculations does not in any way prevent birds from flying. Similarly, people do not need to calculate their optimal behavior functions in order to behave adaptively. They simply have to use successful algorithms; they do not have to be able to make the calculations that would show that these algorithms are successful. (Chater and Oaksford 2000, p. 110; see also Chater et al. 2003, p. 70)

Here and elsewhere, Chater and Oaksford exploit Marr's aerodynamical analogy in order to deflate the idea that some revisionary response is needed to the problem of intractability. At root, their idea seems to be that the rational explanations postulated are logically independent of the rational calculations postulated in the exercise of a given capacity. That is, when rational analysts posit an optimal behavior function f that is later discovered to be NP-hard, we're not to interpret the rational analyst as being seriously committed to the claim that cognitive agents actually rationally calculate f. Rather, such an f is at most just part of the theorist's 'explanatory toolkit'.

Understanding how this response works is far from trivial. In the remainder of this section, we try to fully and charitably represent their rationale by canvasing several further passages that we think jointly lay bare the core of their position.

1.3 'As if' calculation

Chater and Oaksford and colleagues repeatedly appear to argue that, ultima facie, computational tractability is not necessarily a minimal computational constraint, and thus not a regulative norm governing the construction of rational explanations of human psychological capacities. Their advancement of this position pivots on the threefold distinction between rational explanation, description, and calculation. By 'rational explanation' or 'description', they mean explanations or descriptions of the input-output behavior that conform approximately with the results that would be obtained by some rational calculation, where 'rational calculation' is meant to refer to the mental execution of certain probabilistic, logical, or decision-theoretic operations (see, e.g., Chater et al. 2003, pp. 66–67).

Given this distinction, Chater and Oaksford subsequently contend that the rational calculations posited to explain how agents execute their capacities are not logically implied by rational explanations that posit them:

> [w]e suggest that the view that rational explanation requires that people themselves carry out the relevant rational calculations is a fundamental mischaracterization of how rational principles are used to explain thought and behavior in behavioral ecology, economics, and psychology. (Chater et al. 2003, p. 66)

They go on to contend that this suggestion then dissolves the problem of intractability:

> if we adopt the view that we have been advocating, that rational explanation should be understood in terms of 'rational description' rather than 'rational calculation,' then these concerns about computational complexity disappear. (Chater et al. 2003, p. 70)

To bolster their case, Chater and Oaksford and colleagues extend this argument to other disciplines, from ecology to economics, where rational explanations have a longer and stronger tradition than in cognitive science:

> [Rational description] does not assume (though it does not rule out) that the thought processes underlying behavior involves any rational calculation. An analogy may be useful: the wings of a bird may approximate the results of a rational calculation of optimal aerodynamic design. Moreover, this observation helps explain why the wing has the structure that it does; but there is, of course, no presumption that the bird conducts any calculations in designing its wing. Behavioral ecologists extend this pattern of biological explanation from anatomy and physiology to behavior [...and] expressly disavow a rational calculation interpretation of their theories as being patently at variance with the cognitive limitations of the animals they study. Contemporary economics also aims to explain behavior by rational description, rather than rational calculation. (Chater et al. 2003, p. 67)
>
> Economists do not assume that people make complex game-theoretic or macroeconomic calculations; zoologists do not assume that animals calculate how to forage optimally; and, in psychology, rational analyses of, for example, memory, do not assume that the cognitive system calculates the optimal forgetting function with respect to the costs of retrieval and storage. Such behavior may be built in by evolution or be acquired via a long process of learning, but it need not require on-line computation of the optimal solution. (Chater and Oaksford 2000, p. 110)
>
> There is, moreover, a recognition in economics that applying rational theories, such as probability theory, expected utility theory, and game theory will only provide an approximation model of people's behavior. Economists allow that [...] faced with complexity, individuals resort to rules of thumb, to 'back of the envelope' calculations, to satisficing behavior. [...] Economists thus recognize that behavior only approximates to rationality; but economic theory typically idealizes away from such limitations. [...] For our purposes, the important point is that most economists interpret their theories as about rational description, but not rational calculation (at best, people act 'as if' they made rational calculations; but they do not actually perform such calculations); and they agree furthermore that actual behavior is only an approximation to rational standards. (Chater et al. 2003, p. 67)

Together, these passages provide a canvassing of the deflationary response that Chater and Oaksford believe is available to the rational analyst. Rather than supposing that the target explanandum is explained by the exact computation of the intractable function modeling it, they instead just construe the target explanandum *as if* the intractable function modeling it were computed exactly.

It seems to us that in these passages one finds an admixture of—not necessarily mutually consistent—perspectives on rational analysis, which bottom out in the notions of 'as if' explanation and calculation. These notions are then deployed to screen off the charge of intractability. But it's far from obvious that such a response could be successful. At the very least, whether it is successful depends, inter alia, on the semantics of the 'as if' connective. First, Chater and Oaksford's relocation of the optimal behavior function f derived in rational analysis from the agent's 'cognitive equipment' to the analyst's 'explanatory toolkit' may implicate some form of noncomputationalism (i.e., a rejection or even denial of computational realism about the processes by which f is realized). This reading is inconsistent, however, with Chater and Oaksford's other commitments; for instance, the claim that people 'have to use successful algorithms' suggests just such a commitment to computationalism after all. Second, the claim that birds can fly despite knowing nothing of aerodynamics seems to suggest that the computational processes underlying rational behavior functions f are believed to be of a special type, viz., algorithms which are implicit or subsymbolic, and so need not look anything like the calculations postulated in probability theory, logic, or decision theory. While this second reading is consistent with Chater and Oaksford's claim that 'people do not need to calculate their optimal behavior functions', this latter claim itself suggests an additional third reading, viz., that the main computational work is performed offline (e.g., by evolution; see Chater et al. 2003, p. 70). A fourth and fifth interpretation of 'as if' rational calculation is suggested by their use of terms such as 'rules of thumb', 'back of the envelope', 'satisficing behavior', and 'approximation', viz., a reading of "as if' rational calculation' as meaning heuristics or approximation algorithms, respectively.

In the next §2 we consider these five possible meanings of 'as if', and for each we analyze how rational explanation fares with respect to intractability. We will show that none of these possible meanings of 'as if' succeeds in dissolving the intractability problem. In the subsequent §3, we will show how rational explanations can meet the tractability constraint. Our approach suggests a computationally unproblematic interpretation of 'as if' that is compatible with the original conception of rational analysis.

2 Intractability and five meanings of 'as if' calculation

2.1 Noncomputationalism

Throughout their works, including the quotes above, Chater, Oaksford, and other rational analysts make clear that they view cognitive behavior as ultimately the result of efficient and effective algorithms; and this suggests that they take f to be computed somehow (even if just approximately, see §2.5). Consequently, Chater and Oaksford probably do not wish to appeal to noncomputationalism as a way of fencing off the charge of intractability.

Be that as it may, we can imagine that their bird analogy is quite appealing for some readers—including some rational analysts—and, to them, it may still suggest that intractable functions f may be realized tractably by physical means other than computation. Were this intuition correct, a noncomputationalist interpretation of rational explanations could still circumvent the problem of computational intractability. As it turns out, however, there is no good reason to believe that this intuition could be correct; for there are zero reasons to think, given current knowledge of physical reality, that birds' capacity for flight defies intractability.

The scientific literature is replete with claims of physical systems presumably defying the computational limits of classic Turing machines. For instance, it has been claimed by some researchers that soap bubbles are able to efficiently solve the NP-hard Steiner Tree Problem, and that proteins are able to solve NP-hard folding problems. Such claims invariably have been shown to be theoretically and empirically unsubstantiated (Aaronson 2005; Ngo et al. 1994). We suspect that claims such as these may arise from a faulty understanding of NP-hardness, or possibly an all-too-narrow interpretation of the term ‘computation’.

For a function f to be intractable in the sense of being NP-hard just is for there to exist *no* algorithm that can compute that function in a reasonable (polynomial) time. Here, ‘algorithm’ means *any* finite procedure under *any* formalism of computation, include those based in natural physical systems. Even though NP-hardness was originally defined in terms of the Turing-machine formalism, it has been proven to generalize directly to equivalent formalisms, including recurrent neural networks and cellular automata. Even under the most powerful formalisms to this day, quantum computation and analog neural networks, which are known to encompass more functions than the Turing formalism, intractable (NP-hard) functions cannot be computed in realistic (polynomial) time (Aaronson 2008; Šíma and Orponen 2003; van Rooij 2008). In other words, unless one were to ascribe more powers to cognitive agents than there are known to be available for Turing or quantum computers with respect to the input-output functions that they can tractably realize, cognitive agents would be unable to tractably realize intractable (NP-hard) functions.

As a last resort, rational analysts may wish to ascribe ‘superpowers’ to cognitive agents. However, the bird analogy notwithstanding, it would be a mystery how such powers could be physically realized (Aaronson 2005; Bournez and Campagnolo 2008; Cotogno 2003; Davis 2004; Nayebi 2014; Piccinini 2011; Tsotsos 1990; van Rooij 2008), whereas no such mystery exists for the other formalisms, in principle. Since it is clear that rational analysts like Oaksford and Chater do not think that cognitive agents have superpowers anyway (see quotes above and see §2.4 and 2.5), let’s move on to the other possible meanings of ‘as if’ calculation.

2.2 Implicit or subsymbolic computation

We’ve shown that rational analysts likely don’t intend the meaning of “as if’ calculation’ to be interpreted noncomputationally, and that such an interpretation isn’t a feasible dissolution of the problem of intractability. In this subsection, we turn to a second set of possible meanings. Rational analysts may intend their construct, “as if’ calculation’, to be understood in terms of computations that are merely implicit in the cognitive agent in which they occur, or that occur unconsciously, or subsymbolically. This interpretation is suggested by Chater and Oaksford’s claims that birds *know nothing* about the very aerodynamical theories that seem to be so crucial for explaining their capacity for flight (2000, p. 110; see also Chater et al. 2003, p. 70), which Gigerenzer paraphrases as ‘[a]s-if optimization models are silent about the actual process, although it is sometimes suggested that the measurements and calculations might happen *unconsciously*’ (2004, p. 64, emphasis ours). Chater and Oaksford also make the same point this way:

> […] people do not need to calculate their optimal behavior functions in order to behave adaptively. They simply have to use successful algorithms; *they do not have to be able to make the calculations that would show that these algorithms are successful*. (Chater and Oaksford 2000, p. 110)

The point being made in these and other such passages seems to be that, even though the postulated f in rational analysis is somehow computed by the cognitive agent (otherwise see §2.1), it's not assumed to be computed by the agent using algorithms that build on explicit knowledge, such as that which would be used by a cognitive scientist for deriving or calculating f. That is, even though scientists need to know logic and probability theory to explain rational behavior (just as scientists need to know aerodynamics to be able to explain bird flight), cognitive agents do not need to know logic or probability theory to behave rationally (just as birds do not need to know aerodynamics to fly). Hence, the computational process underlying f may better be construed as implicit and/or subsymbolic algorithms that act 'as if' they perform rational calculations in the mere sense that they are weakly (i.e., input-output) equivalent to such calculations, without themselves being like those calculations.

This interpretation of "as if' rational calculation' is in line with the recent rise of proposals for so-called 'rational process models' (Shi et al. 2010; Sanborn et al. 2010; Griffiths et al. 2012), that aim to provide algorithmic-level explanations of functions f derived in rational analyses. Rational process models postulate computational processes that are considered cognitively plausible *a priori*, such as spreading activation in a neural network or probabilistic sampling; and the computational processes so postulated are very much unlike the explicit calculations that a logician or probability theorist would generate when computing f with, say, paper and pencil, or on a desktop computer, or using a hand-held calculator. Moreover, these computational processes are in no way aware of their own 'rationality'; for they would not 'know' that f is what they compute, much less 'know' why f is what they compute. The assertion that such computational processes are good algorithmic-level explanations for a rational function f is motivated by reference to theorems showing that, given unlimited computational resources, these algorithms converge exactly on the input-output function f for which they aim to give 'how'-explanations (i.e., weak equivalence) (e.g. Sanborn et al. 2010). Consequently, rational process models appear to be in line with an interpretation of the meaning of 'as if' rational calculation in terms of implicit, unconscious, or subsymbolic computations. This appearance then raises the question: can the type of implicit or subsymbolic algorithms postulated by rational process models tractably compute an intractable function f? The answer must be 'no'. Again, if a function f is intractable (i.e., NP-hard), then there exists no algorithm that can tractably (polynomial-time) compute f.

A tempting response is to acknowledge both the inexistence of any such algorithm, alongside the possibility that someone still could in the future discover such algorithm. Rational analysts sometimes give the impression of this response by mentioning intractability in the context of current scientific knowledge:

> Current calculi for reasoning, including standard and non-standard logics, probability theory, decision theory and game theory, are computationally intractable

> [... T]here is presently little conception either of how such probabilistic models can capture the 'global' quality of everyday reasoning, or how these probabilistic calculations can be carried out in real-time to support fluent and rapid inference, drawing on large amounts of general knowledge, in a brain consisting of notoriously slow and noisy neural components. (Chater and Oaksford 2001, p. 211)

The phrase 'there exists no algorithm' should not be relativized to the current state of our scientific knowledge, however. The correct reading is the stronger, modal statement that there cannot exist, in principle, an algorithm that can tractably compute f, if only for the reason that any claim otherwise would contradict the premise that establishes the intractability of f.[6] While the intractability of f defies all and any tractable algorithms, regardless of the nature of these algorithms, rational process modelers in cognitive science often observe that the algorithms they implement to compute f may run quite fast and perform quite well in practice. We realize that the assertion that no rational process model can ever explain how an intractable function f can be tractably computed may therefore seem counterintuitive from their perspective. Nonetheless, such circumstances can be understood as a mismatch between the generality of the intractable computational-level function f and the domain-specific tractability and/or quality of the algorithms so simulated. If the computational-level function f is intractable, then any algorithm for computing it can run fast and perform well only for a proper subset of input domains, viz., those domains for which the computation is tractable (see also Kwisthout et al. 2011; van Rooij et al. 2012; and §3).

2.3 Offline computation

Interpreting Chater and Oaksford's concept of 'as if' calculation as implicit or subsymbolic computation assumes that rational functions $f : I \rightarrow O$ are computed online, i.e., that the computational process has to run its course from input to output every time it is presented with a new input $i \in I$. Given that most cognitive capacities ϕ which are the target explananda of rational analysis occur on a scale of milliseconds or seconds, intractable computations requiring weeks, months, or centuries for their completion are, de facto, implausible accounts of those capacities. However, were the computations to be performed offline instead of online, e.g., at a scale of development or evolution, then perhaps the problem of intractability wouldn't be an explanatory obstacle anymore.

This consideration suggests a third possible way of understanding rational analysts concept of 'as if' calculation—i.e., as offline computation. Exegetically, the suggestion is supported by various statements by rational analysts, such as the claim that '[...] a successful probabilistic rational analysis of a cognitive task does not necessarily require that the cognitive system be carrying out probabilistic calculations—any more

[6] This point cannot be overemphasized. The classification of a function f as intractable does not come lightly. For even were there to exist a proof that some such tractable algorithm exists, then even if we could know nothing else about it, we would be led to the classification of f as tractable.

than the bird is carrying out aerodynamic calculations in growing a wing perfectly adapted for flight' (Oaksford and Chater 2009, p. 111). Or again:

> Economists do not assume that people make complex game-theoretic or macroeconomic calculations; zoologists do not assume that animals calculate how to forage optimally; and, in psychology, rational analyses of, for example, memory, do not assume that the cognitive system calculates the optimal forgetting function with respect to the costs of retrieval and storage. Such behavior may be built in by evolution or be acquired via a long process of learning—but it need not require online computation of the optimal solution. (Chater and Oaksford 2000, p. 110)

Unfortunately, such passages raise more questions than they answer. In particular, we're left with no clearer understanding of what it is that rational analysts take to be computed offline. For instance, is f 'computed offline' in the sense that agents needn't determine online whether f is the optimal behavior function, given their goals G and environment E, because development or learning has already determined this for them? Or, is it that evolution or development has pre-computed the mapping for every input $i \in I$, such that, when confronted with any $i \in I$, the output $f(i)$ can simply be retrieved rather than being computed online? The former possibility is consistent with Chater and Oaksford and colleagues' (2003, p. 67) claim that their program helps explain why birds wings have the structure that they do without there being any presumption that birds conduct any calculations in *designing* their wings. (Here, wing-design by evolution is presumably analogous with the design of ϕ by evolution, where ϕ is characterized by f in a rational analysis.) The latter possibility, however, is consistent with Chater and Oaksford (2000, p. 110; 2008, p. 36) claim that behavior may be built in by evolution or acquired by a lengthy learning process, but needn't require online computation of the optimal solution. (Here, 'offline computation of the optimal solution' may mean that the optimal output $f(i)$ for any given i needn't be computed online because it has already been computed offline.) Either way, it's apparent that some ambiguity remains with respect to the particular meaning of 'offline computation'. In what follows, we'll attempt to disambiguate the meanings of 'offline computation' by exploring some of their implications.

Plausibly, a cognitive agent needn't continuously determine its optimal behavior function f, and needn't compute its own capacity from scratch before being able to exercise that capacity. Also plausible is the idea that, instead, such capacities are shaped through evolution and development. Yet, neither consideration circumvents the problem of intractability. After all, intractability is a property of the input-output function f itself, not of the processes by which such functions may be derived—be they evolution, development, or rational analysis. Of course, derivation of an optimal function f, relative to any arbitrary set of goals G and E, may itself also be intractable; possibly, the derivation is even uncomputable (i.e., there may be no (tractable) algorithm for performing step 5 in Anderson's 6-step procedure for rational analysis). But that locus of intractability is not to be confused with the complexity of computing $f : I \rightarrow O$ itself, for any $i \in I$. The computation of $f : I \rightarrow O$, for any $i \in I$, is intractable for many rational functions derived in rational analysis, and it is this intractability problem that is the topic of this paper. An appeal to offline computation in the first sense fails to dissolve this problem.

So what about the idea that f is computed offline in the second sense, i.e., that evolution or development has pre-computed the corresponding $f(i)$ for every input i, such that both retrieval suffices and no online computation of $f(i)$ is necessary? This idea may indeed circumvent the problem of intractability of f. However, such offline computation is highly implausible for most cognitive capacities, which are believed to have the property of productivity (i.e., an in principle unbounded competence; Fodor and Pylyshyn 1988; Chater and Oaksford 1990). For instance, humans can categorize an in principle unbounded number of sets of objects, can understand an in principle unbounded set of sentences, reason from in principle unbounded sets of possible subsets of premises, etc. It's very unlikely that evolution or development would have pre-computed offline the outputs for all such unbounded sets of possible inputs—especially if the inputs were never before encountered during evolution or development, but completely novel.

Chater and Oaksford (2000, p. 110) have acknowledged that offline computation would be of limited relevance for cognition: '[i]n some contexts, however, some online computations may be required', they write, '[s]pecifically, if behavior is highly flexible with respect to environmental variation'. They mention visual perception as an example:

> [...] leading theories of perceptual organization assume that the cognitive system seeks to optimize online either the *simplicity* or *likelihood* [...] of the organization of the stimulus array. These calculations are recognized to be computationally intractable (2000, pp. 110–111).[7]

Arguably, all but the most trivial forms of cognitive behaviors will require some form of online computation. Given that the program of rational analysis is particularly concerned to explain non-trivial forms of cognitive behavior, the attempt to make sense of this crucial construct, 'as if' calculation, in terms of offline computation gives no solace from the problem of intractability.

2.4 Heuristic computation

All interpretations of 'as if' calculation considered thus far have taken rational analysts to mean both that the function f describes exactly what is realized by cognitive agents, but that agents realize f by other means than rational calculation, e.g., by noncomputational means (§2.1), by implicit or subsymbolic computation (§2.2), or by offline computation (§2.3). We've shown that none of these constructs, plausibly interpreted, will shield rational explanations from the problem of intractability. However, perhaps rational analysts mean that 'as if' calculation should be understood as the *inexact* realization of f rather than as the *exact* realization of that function. That is,

[7] Note that, here, Chater and Oaksford do use the term 'calculation' to refer to the computational process involved in determining the output of some ϕ, in this case perceptual organization. Perhaps 'calculation' is intended in a broad sense, meaning 'computation' and not being synonymous with 'rational calculation'.

‘as if’ calculation could be interpreted as the idea that cognitive agents use so-called ‘heuristics’, where a heuristic H is an algorithm known not to compute f exactly; instead, it will output something different from $f(i)$ for at least some $i \in I$[8]. Accordingly, some relationship—weaker than equality—is believed to obtain between f and the particular function $f_H : I_H \to O_H$, with $I_H \subseteq I$ and $O_H \subseteq O$, computed exactly by H.[9]

As Gigerenzer notes, though: ‘[h]euristics are distinct from ‘as if’-optimization models’ (2004, p. 64); and indeed, the heuristics-interpretation isn’t obviously superior. On the other hand, rational analysts themselves have explicitly suggested it:

> [I]ntractability results are not necessarily taken to rule out the possibility of practical computation. No algorithm [...] may be tractable, and yet there may be more or less reliable heuristics which often solve the problem, or at least provide something close enough to the solution to be useful. These heuristics need not necessarily be computationally intractable. Computational tractability may be bought at the price of the reliability of the procedures. (Oaksford and Chater 1998, pp. 83–84)

Interpreting ‘as if’ calculation as heuristic computation is also consistent with rational analysts’ other claims that cognitive agents may use ‘rules of thumb’, ‘back of the envelope’ calculations, and display ‘satisficing behavior’ (Chater et al. 2003, p. 67). Moreover, moving to heuristic explanation puts rational analysts in seemingly good company: when faced with the intractability of ones computational-level theory f, appealing to heuristics as algorithmic-level explanations is one of the most widely-adopted strategies by cognitive scientists.

Unfortunately, this strategy runs into serious conceptual problems (van Rooij 2008; van Rooij et al. 2012). Proposing both that a cognitive capacity ϕ is adequately described at the computational level by an intractable function f, and that ϕ’s processes are adequately described at the algorithmic level by some heuristic H, introduces a fundamental inconsistency between computational- and algorithmic-level explanation. To see why, consider all inputs $i \in I$ for which $f(i) \neq f_H(i)$, of which there must be infinitely many.[10] For each such input i, H is an inadequate algorithmic-level explanation. This is because, by definition, H outputs $f_H(i) \neq f(i)$, and so doesn’t explain how ϕ produces output $f(i)$ given i as input. So, either intractable f at the computational level adequately describes the capacity ϕ but heuristic H at the algorithmic level misdescribes how ϕ produces output $f(i)$ given i as input, or else H describes how ϕ produces output $f(i)$ given i as input but the computational-level model needed to describe ϕ is instead f_H. As the degree of this inconsistency remains unbounded by heuristic explanation (otherwise see §2.5), maintaining explanatory consistency between postulating that intractable f is an (approximate) description

[8] Otherwise, H would be an exact algorithm, in which case the earlier problems noted in §2.1–2.3 would reoccur.

[9] For instance, it may be believed that $f(i)$ and $f_H(i)$ are the same for many inputs $i \in I = I_H$, or it may be conjectured that the difference between $f(i)$ and $f_H(i)$ is small for many inputs $i \in I$.

[10] Otherwise, f would not be intractable (Schöning 1990; van Rooij et al. 2012).

of ϕ and that ϕ operates by tractable heuristics becomes impossible (van Rooij and Wright 2006).

The conceptual problems introduced by this inconsistency also manifest when cognitive scientists attempt to empirically (dis)confirm the computational-level theory f and algorithmic-level theory H. After all, what should they predict as the observable outcome of presenting subjects directly with some input $i \in I$ (as produced by the capacity ϕ of interest)? According to the computational-level explanation, they should predict the output will be $f(i)$; but, according to the algorithmic-level explanation, the predicted output should be $f_H(i)$. And since $f_H(i) \neq f(i)$, the two predictions are inconsistent competitors: the computational-level explanation is confirmed by the data only if the algorithmic-level theory is disconfirmed, and vice versa.

While the inconsistency between levels of explanation that's introduced by a heuristics interpretation of 'as if' is potentially resolvable, doing so would render that interpretation inapplicable because H would then be an *exact* algorithm for the new computational-level theory f_H. In other words, any such resolution would require that cognitive scientists recognize, inter alia, that the commitment to H as an algorithmic-level explanation implies that computational-level function f_H—and not f—is the function that adequately describes ϕ. Contra Chater and Oaksford, computational tractability is bought by an appeal to computational-level theory revision, not an appeal to 'as if' in the sense of *inexact* (i.e., heuristic) computation of intractable functions (van Rooij 2008; van Rooij et al. 2012).[11]

As we shall detail in §3, this revisionary response may yield an interpretation of 'as if' compatible with rational analysis that is both unproblematic and, more importantly, different from any of the interpretations suggested by rational analysts to date (including the one considered in the next subsection).

2.5 Approximate computation

A fifth interpretation of 'as if' calculation is latent in rational analysts' claims about approximation. For example, Chater et al.'s (2003, p. 67) claim that 'actual behavior is only an approximation to rational standards' suggests that rational analysts both deny that cognitive agents compute the rational function f using an exact algorithm (which always yields the precise output $f(i)$ for each input i), and assert that agents instead calculate using approximation algorithms (that, for each input i, may output something that is inexactly similar to $f(i)$ or comes close to being or computing $f(i)$).

[11] We thank an anonymous reviewer for raising this possibility: even if it is f_H rather than f that accurately characterizes the capacity of interest ϕ, in practice f_H may be unknown; so couldn't rational analysts contend that the appeal to 'as if' merely serves as a sort of 'promissory note'? That is, until we determine what f_H is, f can serve as a (instrumentalist) working hypothesis that allows research to productively continue. This may be the case. We don't contest that intractable fs can instrumentally lead to important results on occasion. For instance, postulating intractable fs raises scientifically fruitful questions about the conditions under which those functions may be tractable, and answering those questions may lead to (realist) hypotheses about f_H. What we do contest is the claim that the intractability of f is rendered permanently irrelevant by an appeal to 'as if'. Whatever instrumentalist commitments are invoked, it's still the case that, in order for the computational-level theory to be computationally plausible and explanatory, at some point in time—sooner or later—f_H needs to be determined.

It might be thought that the approximation interpretations of the concept of ‘as if’ reduces to the heuristics interpretation already discussed in §2.4. So much the worse for rational analysts. We think that a more charitable thought holds them as prima facie distinct until demonstrated otherwise. Subsequently, to enforce this distinction (in the context of ‘as if’ calculation), let ‘approximation’ imply that there is some form of *bound* on the extent to which the output generated by the approximation algorithm can deviate from $f(i)$. Without such a bound, the interpretation of ‘as if’ calculation as approximation does indeed collapse back into that for heuristics, and the problems already discussed in §2.4 re-occur.

Assuming that talk of approximation is meant to imply bounds on how far the output can deviate from the rational output $f(i)$, then it’s legitimate to ask whether an intractable rational function f be tractably approximated within a reasonable bound.

As previous research on computational intractability in cognitive science evinces, the answer depends both on how one defines the dimension within which the approximation takes place, as well as what one considers a reasonable bound. As van Rooij and Wareham (2012) and Kwisthout and van Rooij (2013) observe, for the types of optimisation functions yielded by rational analysis at least three different approximation dimensions can be defined. For instance, when such f' approximates f, the deviation between f and f' may be bounded by the *structure* of the output (i.e., $f'(i)$ must structurally resemble $f(i)$); by the *value* that is optimized by the output (i.e., the value associated with $f'(i)$ must be close to the value of $f(i)$), or by the *likelihood* of deviation (i.e., $f'(i)$ must have a high probability of being equal to $f(i)$).

It is often assumed that computing approximate outputs for optimization functions is always tractable—even or especially when computing those optimization functions is intractable (see, e.g., Chater et al. 2006; Sanborn et al. 2010). However, this assumption is provably wrong relative to the previously described forms of approximation; in particular, it is for the many probabilistic models that are commonly used in rational analysis (Kwisthout et al. 2011). For example, computing probability distributions exactly is NP-hard (Cooper 1990) and it’s no less hard to approximate these distributions (Dagum and Luby 1993). Likewise, inferring explanations that have maximum posterior probability given the evidence is NP-hard (Shimony 1994) and it’s no less hard to find an explanation that approximates the maximum posterior probability or whose inner structure either resembles the optimal solution or is just guaranteed to have a non-zero probability (Abdelbar and Hedetniemi 1998; Kwisthout and van Rooij 2013; Kwisthout 2011). Moreover, the idea that approximation is tractable also fails on a much broader scale. For example, many NP-hard functions are not tractably value-approximable unless $\mathrm{P} = \mathrm{NP}$ (Arora 1998), efficient expectation-approximation of *any* NP-hard function is impossible[12] (Kwisthout and van Rooij 2013), and the efficient value- *or* structure-approximation of an NP-hard function is ruled out if that function

[12] This claim assumes $\mathrm{NP} \not\subseteq \mathrm{BPP}$, a widely-endorsed conjecture in theoretical computer science (see Johnson 1990, p. 120 and Zachos 1986, p. 396), where ‘BPP’ abbreviates *B*ounded-error *P*robabilistic *P*olynomial time.

is self-paddable (van Rooij and Wareham 2012)—a property apparently holding for many functions including those associated with cognition.[13]

To be clear, pointing out that computing approximate outputs for functions is not always tractable—even or especially when computing optimal functions is intractable—is not to say that tractable approximation of intractable functions is always impossible. There may be functions that are tractably approximable relative to one of the three forms of approximation previously mentioned, but if we have learned anything from computational tractability analysis to date then it is that such functions are the exception rather than the rule. Consequently, if the approximation interpretation of ‘as-if’ calculation is to be intended for a particular target function f, the rational analyst or other researchers making such claims bear the justificatory burden of demonstrating that f is both truly intractable and tractably approximable.

3 How explanation can be rational, ‘as if’, *and* tractable

Readers may be wondering whether the program of rational analysis is fundamentally flawed, given both that it often yields intractable functions f, and that the construct centrally used to work around the problem of intractability—‘as if’ calculation—doesn’t actually work. We don’t think so—quite the contrary. Rational analysis can be a very useful and productive approach to conjecturing computational-level theories. To be clear, our criticism isn’t a criticism of rational analysis per se, as proposed by Anderson (1990), but rather with the way in which the approach has been adopted by many cognitive scientists without sufficient consideration of step 3 in Anderson’s procedure.

In §2, we argued that appeals to ‘as if’ rational calculation fail to circumvent the problem of intractability—certainly for at least some intractable fs generated through rational analysis, and possibly for all. Yet, this doesn’t mean that the problem of intractability cannot be circumvented in other ways. One straightforward way of doing so involves testing fs generated in computational-level theorizing for intractability, and, where necessary, revising them so as to meet the tractability constraint. In fact, as has been shown by van Rooij (2008), an intractable function f may often even be transformed into a tractable function f' with a minimal amount of theory revision. For instance, it often suffices just to determine that the input domain on which ϕ operates in normal (ecologically relevant) situations has restricted parameter ranges.

To explain this idea intuitively, let us reconsider the bird analogy of Marr (1982) and Chater and Oaksford (2000). Flight may be intractable for birds if the conditions under which they were expected to fly would be outside the normal ranges in which they can effectively fly (e.g., in a storm with exceptional speeds of wind that move in arbitrarily complex ways). Under such exceptional conditions birds would fail to fly successfully; their mechanism for flight cannot effectively deal with such extreme circumstances. Nevertheless, under normal conditions—in which air and body parameters remain

[13] A function f is self-paddable if and only if a set of instances $i_1, i_2, \dots, i_m$ of f can be embedded in a single instance i_E of f such that $f(i_1), f(i_2), \dots, f(i_m)$ can be derived from $f(i_E)$. For more details, see Definition 6 in van Rooij and Wareham (2012).

within normal ranges—birds can effectively fly. Their ability to effectively fly under normal circumstances, despite the intractability of flight under arbitrary circumstances, is thus not only to be understood in terms of the internal mechanisms that support flight, but also in terms of the parameters that define 'normal' circumstances.

By analogy, a computational-level theory f may be intractable for an unconstrained input domain, but a cognitive capacity ϕ may not need to deal with arbitrary inputs under 'normal' conditions. That is, the situation in which ϕ is effectively exercised by cognitive agents in their normal lives may be better modeled by a restricted input domain $I' \subset I$; and the function f restricted to that input domain, i.e., $f' : I' \rightarrow O$, can be tractable, even if f itself, given its unrestricted input domain $I \supset I'$, is not. This insight can be used to revise intractable functions f into more restricted functions f' that are tractable.

Note that this perspective also yields a notion of 'as if' that is unproblematic. That is, having discovered a pair consisting of intractable function f and tractable restricted-domain function f', the rational analysts may contend that it is 'as if' cognitive agents compute f in the sense that they compute f' and in the normal range of inputs the functions f' and f are indistinguishable. Importantly though, this indistinguishability is mere appearance; for if one were to present the cognitive agents with inputs outside the normal range then the behavior of the cognitive agents would no longer be guaranteed to look anything like f. Indeed, such behavior would necessarily differ from f on infinitely many inputs outside the normal range (otherwise f would not be intractable; see also footnote 10).[14]

These considerations suggest that rational analysis may be fruitfully extended, by extending the Optimization step in rational analysis with a 4-step subprocedure, called *tractability analysis*:

1. *Intractability:* Check whether f derived in the Step 4 of rational analysis is intractable. If 'yes', proceed with Steps 2–4 of tractability analysis below. If otherwise, continue with Step 5 of rational analysis.
2. *Parameters:* Define a restricted-domain function f' with plausible bounds on input parameters that define 'normal' conditions.
3. *Analysis:* Analyze whether f' is tractable. If so, continue Step 5 of rational analysis with f'.
4. *Iteration:* If no tractable parameter ranges can be found then revise f by returning to Steps 1–3 of rational analysis.

[14] The formal tools for putting this type of revisionary approach into practice have been extensively described by van Rooij (2008) (see also Blokpoel et al. 2013; van Rooij and Wareham 2008), and builds on the the mathematical theory of parameterized complexity (Downey and Fellows 1999). Using proof techniques from this mathematical theory, it can be shown that some intractable (NP-hard) functions $f : I \rightarrow O$ can be computed in fixed-parameter (fp-) tractable time $O(g(K)|i|^c)$, i.e., where g can be any function of the parameters $k_1, k_2, \ldots, k_m$ in set $K = \{k_1, k_2, \ldots, k_m\}$, $|i|$ denotes the input size, and c is a constant. Note that in such event, the intractable f can be computed efficiently (in polynomial-time), even for large inputs, provided the assumption that f operates only on inputs in which the parameters in K are restricted to relatively small values (each $k << |i|$). If rational analysts were to have theoretical and/or empirical reasons for this assumption, then revising f to f'—where f' is f restricted to inputs with small values for parameter $k_1, k_2, \ldots, k_m$—would yield a tractable function f' that will be rational according to the rational analysis that yielded it as f.

Note that the functions derived by rational tractability analyses, whether f or f', would be both rational *and* tractable. Consequently, the computational-level theories so derived will also satisfy both the rationality principle and the tractability constraint. Not only does this combined rational-and-tractability ensure that rational explanations meet the minimal constraint of tractability, but the extended procedure also poses stronger theoretical constraints on what are viable computational-level theories. Combined rational tractability analyses, then, are very much in the spirit of rational analysis as Anderson envisioned it: they're a principled way of coping with the underdetermination of computational-level theory by empirical data.

4 Conclusion

Rational analysis has proven to be an effective and productive way of generating computational-level characterizations of cognitive capacities; more precisely, the functions derived describe and predict cognitive behaviors well under a wide variety of conditions and do so for a wide variety of cognitive capacities. That such functions often face the problem of intractability has been typically disregarded as a pseudoproblem, since rational explanations are agnostic about the *how* of cognition, and merely aim to use the *why* of cognition to derive the *what* of cognition. When the problem has been regarded, rational analysts have pursued a dissolution using the construct of 'as if' calculation.

As we have argued in this paper, the idea that cognitive capacities may be realized by 'as if' calculations provides no shelter from the charge of intractability for rational explanations. Indeed, there is no possible way in which to conceive the *how* of cognition, now or in the future, such that intractable functions can plausibly be accurate descriptions of everyday cognitive capacities operating on the time scale of minutes, seconds, or milliseconds. In the end, such explanations also must satisfy the minimal constraint of computational tractability. This, however, should not be seen as a loss for rational analysis. On the contrary, as we have articulated in §3, rational analyses can be combined with tractability analyses in ways which ensures that computational-level explanations are both rational and tractable, thus paving the way for an even better constrained methodology for conjecturing computational-level theories.

References

Aaronson, S. (2005). NP-complete problems and physical reality. *SIGACT News*, *36*, 30–52.

Aaronson, S. (2008). The limits of quantum computers. *Scientific American*, *298*, 62–69.

Abdelbar, A. M., & Hedetniemi, S. M. (1998). Approximating MAPs for belief networks is NP-hard and other theorems. *Artificial Intelligence*, *102*, 21–38.

Anderson, J. R., & Matessa, M. (1990). A rational analysis of categorization. In B. Porter & R. Mooney (Eds.), *Proceedings of the 7th international workshop on machine learning* (pp. 76–84). San Francisco: Morgan Kaufmann.

Anderson, J. R. (1990). *The adaptive character of thought*. Hillsdale: Lawrence Erlbaum Associates Inc.

Anderson, J. R. (1991a). The adaptive nature of human categorization. *Psychological Review*, *98*, 409–429.

Anderson, J. R. (1991b). Is human cognition adaptive? *Behavioral and Brain Sciences*, *14*, 471–517.

Anderson, J. R. (1991c). The place of cognitive architectures in a rational analysis. In K. Van Lehn (Ed.), *Architectures for intelligence* (pp. 1–24). Hillsdale: Erlbaum.

Arora, S. (1998). The approximability of NP-hard problems. *Proceedings of the 30th annual symposium on the theory of computing* (pp. 337–348). New York: ACM Press.

Blokpoel, M., Kwisthout, J., van der Weide, T. P., Wareham, T., & van Rooij, I. (2013). A computational-level explanation of the speed of goal inference. *Journal of Mathematical Psychology*, *57*, 117–133.

Bournez, O., & Campagnolo, M. L. (2008). A survey of continuous-time computation. In S. B. Cooper, B. Löwe, & A. Sorbi (Eds.), *New computational paradigms: Changing conceptions of what is computable* (pp. 383–423). Berlin: Springer.

Chater, N., & Oaksford, M. (1990). Autonomy, implementation, and cognitive architecture: A reply to Fodor and Pylyshyn. *Cognition*, *34*, 93–107.

Chater, N., & Oaksford, M. (1999). Ten years of the rational analysis of cognition. *Trends in Cognitive Sciences*, *3*, 57–65.

Chater, N., & Oaksford, M. (2000). The rational analysis of mind and behavior. *Synthese*, *122*, 93–131.

Chater, N., & Oaksford, M. (2001). Human rationality and the psychology of reasoning: Where do we go from here? *British Journal of Psychology*, *92*, 193–216.

Chater, N., & Oaksford, M. (2008). *The probabilistic mind: Prospects for Bayesian cognitive science*. Oxford: Oxford University Press.

Chater, N., Oaksford, M., Nakisa, R., & Redington, M. (2003). Fast, frugal, and rational: How rational norms explain behavior. *Organizational Behavior and Human Decision Processes*, *90*, 63–86.

Chater, N., Tenenbaum, J. B., & Yuille, A. (2006). Probabilistic models of cognition. *Trends in Cognitive Science*, *10*, 287–293.

Cooper, G. F. (1990). The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, *42*, 393–405.

Cotogno, P. (2003). Hypercomputation and the physical Church-Turing Thesis. *British Journal of Philosophy of Science*, *54*, 181–223.

Dagum, P., & Luby, M. (1993). Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial Intelligence*, *60*, 141–153.

Danks, D. (2008). Rational analyses, instrumentalism, and implementations. In N. Chater & M. Oaksford (Eds.), *The probabilistic mind: Prospects for rational models of cognition* (pp. 59–75). Oxford: Oxford University Press.

Danks, D. (2013). Moving from levels and reduction to dimensions anand constraints. In M. Knauff, M. Pauen, N. Sebanz, & I. Wachsmuth (Eds.), *Proceedings of the 35th annual conference of the cognitive science society* (pp. 2124–2129). Oxford: Oxford University Press.

Davis, M. (2004). The myth of hypercomputation. In C. Tuescher (Ed.), *Alan Turing: Life and legacy of a great thinker* (pp. 195–211). Berlin: Springer.

Dennett, D. C. (1994). Cognitive science as reverse engineering: several meanings of 'top down' and 'bottom up'. In D. Prawitz, B. Skyrms, & D. Westerstahl (Eds.), *Logic, methodology, and philosophy of science IX* (pp. 679–689). Amsterdam: Elsevier Science.

Downey, R. G., & Fellows, M. R. (1999). *Parameterized complexity*. New York: Springer.

Ellis, N. C. (2006). Language acquisition as rational contingency learning. *Applied Linguistics*, *27*, 1–24.

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, *28*, 3–71.

Fortnow, L. (2009). The status of the P versus NP problem. *Communications of the ACM*, *52*, 78–86.

Garey, M., & Johnson, D. (1979). *Computers and intractability. A guide to the theory of NP-completeness*. San Francisco: W. H. Freeman & Co.

Gigerenzer, G. (2004). Fast and frugal heuristics: The tools of bounded rationality. In D. Koehler & N. Harvey (Eds.), *Blackwell handbook of judgment and decision making* (pp. 62–88). Malden: Blackwell.

Goodman, N. D., Tenenbaum, J. B., Feldman, J., & Griffiths, T. L. (2008). A rational analysis of rule-based concept learning. *Cognitive Science*, *32*, 108–154.

Gray, W. D., Sims, C. R., Fu, W., & Schoelles, M. J. (2006). The soft constraints hypothesis: A rational analysis approach to resource allocation for interactive behavior. *Psychological Review*, *113*, 461–482.

Griffiths, T. L., Vul, E., & Sanborn, A. N. (2012). Bridging levels of analysis for probabilistic models of cognition. *Current Directions in Psychological Science*, *21*, 263–268.

Johnson, D. (1990). A catalog of complexity classes. In J. van Leeuwen (Ed.), *Handbook of theoretical computer science; volume A: Algorithms and complexity* (pp. 67–161). Cambridge: MIT Press.

Kwisthout, J. (2011). Most probable explanations in Bayesian networks: Complexity and tractability. *International Journal of Approximate Reasoning*, *52*, 1452–1469.

Kwisthout, J., & van Rooij, I. (2013). Bridging the gap between theory and practice of approximate Bayesian inference. *Cognitive Systems Research*, *24*, 2–8.

Kwisthout, J., Wareham, T., & van Rooij, I. (2011). Bayesian intractability is not an ailment that approximation can cure. *Cognitive Science*, *35*, 779–784.

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing visual information*. San Francisco: W. H. Freeman & Co.

Nayebi, A. (2014). Practical intractability: A critique of the hypercomputation movement. *Minds and Machines, 24*, 275–305.

Ngo, J. T., Marks, J., & Karplus, M. (1994). Computational complexity, protein structure prediction, and the Levinthal paradox. In K. Merz Jr & S. Le Grand (Eds.), *The protein folding problem and tertiary structure prediction* (pp. 433–506). Boston: Birkhauser.

Norris, D. (2006). The Bayesian reader: Explaining word recognition as an optimal Bayesian decision process. *Psychological Review*, *113*, 327–357.

Oaksford, M., & Chater, N. (1998). *Rationality in an uncertain world: Essays on the cognitive science of human reasoning*. Sussex: Psychology Press.

Oaksford, M., & Chater, N. (2009). Précis of Bayesian rationality: The probabilistic approach to human reasoning. *Behavioral and Brain Sciences, 32*, 69–120.

Piccinini, G. (2011). The physical Church-Turing thesis: Modest or bold? *British Journal of Philosophy of Science*, *62*, 733–769.

Sanborn, A. N., Griffiths, T. L., & Navarro, D. J. (2010). Rational approximations to rational models: Alternative algorithms for category learning. *Psychological Review*, *117*, 1144–1167.

Schöning, U. (1990). Complexity cores and hard problem instances'. In T. Asano, T. Ibaraki, H. Imai, & T. Nishizeki (Eds.), *Proceedings of the international symposium on algorithms (SIGAL'90)* (pp. 232–240). Berlin: Springer.

Shi, L., Griffiths, T. L., Feldman, N. H., & Sanborn, A. N. (2010). Exemplar models as a mechanism for performing Bayesian inference. *Psychonomic Bulletin & Review*, *17*, 443–464.

Shimony, S. E. (1994). Finding MAPs for belief networks is NP-hard. *Artificial Intelligence*, *68*, 399–410.

Šíma, J., & Orponen, P. (2003). General-purpose computation with neural networks: A survey of complexity-theoretic results. *Neural Computation*, *15*, 2727–2778.

Tsotsos, J. K. (1990). Analyzing vision at the complexity level. *Behavioral and Brain Sciences*, *13*, 423–469.

van Rooij, I. (2008). The tractable cognition thesis. *Cognitive Science*, *32*, 939–984.

van Rooij, I., & Wareham, T. (2008). Parameterized complexity in cognitive modeling: Foundations, applications and opportunities. *Computer Journal*, *51*, 385–404.

van Rooij, I., & Wareham, T. (2012). Intractability and approximation of optimization theories of cognition. *Journal of Mathematical Psychology*, *56*, 232–247.

van Rooij, I., & Wright, C. D. (2006). The incoherence of heuristically explaining coherence. In R. Sun & N. Miyake (Eds.), *Proceedings of 28th annual conference of the cognitive science society* (p. 2622). Mahwah: Lawrence Erlbaum Associates.

van Rooij, I., Wright, C. D., & Wareham, T. (2012). Intractability and the use of heuristics in psychological explanations. *Synthese*, *187*, 471–487.

Wimsatt, W. C. (2007). *Re-engineering philosophy for limited beings: Piecewise approximations to reality*. Cambridge: Harvard University Press.

Zachos, S. (1986). Probabilistic quantifiers, adversaries, & complexity classes: An overview. In A. L. Selman (Ed.), *Structure in complexity theory* (pp. 383–400). Berlin: Springer.