RICHARD SAMUELS

The Complexity of Cognition

Tractability Arguments for Massive Modularity

A core commitment of contemporary nativism is that human beings possess innate, domain-specific mental structure, not merely for low-level perceptual processes but also for various "higher" cognitive tasks—paradigmatically, involving reasoning and decision-making—that would traditionally be viewed as parts of *central cognition*. One would be hard pressed to find any nativist who did not subscribe to this general thesis; and yet the precise nature of the specialized endowment on which central cognition depends remains a point of considerable controversy.

According to one venerable proposal that continues to exert a profound influence on psychological theorizing, the specialized structures on which central cognition depends primarily take the form of *representational* items, such as beliefs and bodies of mentally represented information somewhat akin to theories (Carey, 1985; Fodor, 2000; Gopnik & Meltzoff, 1998). This kind of nativism figures prominently in the rationalist tradition that traces from Plato, through Descartes, to Chomsky's work on language; and for this reason I refer to it as *psychological rationalism* (or just "rationalism" for short).

In recent years, however, an alternative and more radical nativist proposal has attained a certain prominence—not to mention notoriety. The view in question is sometimes called *massive modularity* (MM) and maintains that, in addition to whatever innate representational structure we may possess, central processes also rely on a multitude of innate, special-purpose information processing mechanisms or "modules" (Cosmides & Tooby, 1994; Fodor, 2000; Samuels, 1998; Sperber, 1994, 2001). So, for example, it has been suggested that we possess modules for folk

I would like to thank Peter Carruthers, Keith Hossack, Stephen Laurence, David Papineau, Gabe Segal, and Mark Textor for commenting on earlier drafts of this chapter. I would also like to thank Joanna Bryson for helpful discussion of the material in section 7.

biology, naive physics, theory of mind, and arithmetic. Thus construed, massive modularity differs from its more traditional, rationalist counterpart in being primarily a nativism about cognitive *mechanisms* as opposed to cognitive contents (Fodor, 2000; Samuels, 1998).

The commitments of MM and psychological rationalism overlap to a considerable degree. Both acknowledge that central cognition depends on substantial amounts of innate, domain-specific structure. Moreover, contemporary advocates of both positions almost invariably adopt some version of the *peripheral modularity hypothesis*, on which both perceptual (or input) processes and motor (or output) processes are subserved by an array of innate modules (Fodor, 1983). In view of this, it is seldom easy to discriminate between the two views on experimental grounds alone. Even so, advocates of MM maintain that their conception of cognition is independently plausible in the light of various general, theoretical considerations, of which perhaps the most prominent and widely invoked is what we might call the *tractability argument* for massive modularity. According to this argument, central cognition must be subserved by modular mechanisms because the alternatives—including psychological rationalism—are *computationally intractable*.

The central aim of this chapter is to assess the scope and limits of the tractability argument. In doing so, I argue for two claims. First, I argue that when explored with appropriate care and attention, it becomes clear that the argument provides no good reason to prefer massive modularity to the more traditional rationalist alternative. Second, while I deny that tractability considerations support massive modularity per se, I do not claim that they show nothing whatsoever. In particular, I argue that a careful analysis of tractability considerations suggest a range of characteristics that any plausible version of psychological rationalism is likely to possess.

Before arguing for these claims, however, there are a number of preliminary issues that need to be addressed. In section 1, I outline and clarify the general form of the tractability argument; and in section 2 I explain how massive modularity is supposed to resolve intractability worries. The remainder of the chapter—sections 3 to 7—is largely concerned with highlighting the deficiencies of the main extant arguments for claiming that nonmodular mechanisms are intractable. In section 8, I conclude by sketching some of the general characteristics that a plausible rationalist alternative to massive modularity—one capable of subserving tractable cognitive processes—is likely to possess.

1 Tractability Arguments for Massive Modularity

Although versions of the tractability argument vary considerably in detail, they all share the following pair of commitments. First, they assume that the *classical computational theory of mind* (CTM) is true:[1]

1. Though sometimes only tacitly and sometimes only for the sake of argument.

CTM: Human cognitive processes are classical computational ones—roughly, algorithmically specifiable processes defined over syntactically structured mental representations.

As has been commonly observed, however, the truth of CTM requires more than *mere* computability, since there are many algorithms that demand more time and resources—memory, information, and computational power—than actual human beings possess. Rather, what it requires is that mental processes are in some suitable sense *tractably* computable: roughly speaking, that they are specifiable in terms of algorithms that do not require more time or resources than humans can reasonably be expected to possess.[2] It is on this point that advocates of the tractability argument seek to undermine alternatives to MM. That is, they endorse the following *intractability thesis* (IT):

IT: Nonmodular cognitive mechanisms—in particular mechanisms for reasoning—are computationally intractable.

As will soon become apparent, the arguments for IT vary considerably. Nonetheless, the *source* of intractability is almost invariably assumed to be what many have called the "frame problem,"[3] though it is perhaps more accurately (and less contentiously) referred to as the *problem of relevance*. Nomenclature aside, the problem is this: How can a device determine in a computationally tractable manner which operations, options, or items of information are relevant to the cognitive task at hand? A satisfactory computational theory of mind must address this problem. Yet, according to IT, non-MM theories are unable to do so because relevance poses an insurmountable problem for nonmodular reasoning mechanisms. So, it would seem to follow that:

MM: The mind—including those parts responsible for reasoning—is composed of modular mechanisms.

And this is, of course, precisely what the massive modularity hypothesis requires.

2. According to one characterization of tractability familiar from computer science, an algorithm for solving some problem is *tractable* if, in the worst case, it is *polynomial* in the size of the input; that is, the resources required to compute a solution to every input can be expressed as a polynomial (or better) function of input size—e.g., n^2 or n^{300}. In contrast, an algorithm is intractable if, in the worst case, it is *superpolynomial*, in the sense that resource requirements increase *exponentially* (or worse) as a function of input size and can thus only be expressed as superpolynomial functions, such as 2^n or 100^n. But for current purposes this characterization of (in)tractability is doubly unsuitable. First, it is very widely assumed on inductive grounds by those who model cognitive processes that pretty much any interesting computational problem is superpolynomial in the *worst* case. Thus, the current criterion for intractability does little more than characterize those problems that are *not* of interest to a computational account of cognition. Second, it is entirely possible for a superpolynomial algorithm to very frequently—indeed *normally*—be significantly less expensive than the worst case. In which case, it's hard to see why intractability, in this sense, poses a problem for CTM. After all, it may just be that performance limitations prevent the algorithm being used in the worst case.

3. Dennett (1987); Fodor (1983, 2000); Sperber (1994); Tooby and Cosmides (1992).

In the following discussion I assume for the sake of argument that CTM is true and focus on the intractability thesis. What I aim to show is that a commitment to IT is built on shaky foundations, since the main arguments for it are deeply unsatisfactory. But first I need to explain how MM is supposed to secure tractability where the alternatives allegedly fail.

2 How Does Massive Modularity Help Resolve Tractability Problems?

The answer to the above question can be divided into two parts. First, according to MM—and in contrast to an earlier, well-known thesis defended by Fodor (1983) and others—modularity is not restricted to the periphery of the mind: to those *input systems* responsible for perception and *output systems* responsible for the production of action. According to MM, *central systems* for reasoning and decision-making can be divided into modules as well (Jackendoff, 1992). Thus MM maintains that our minds are modular in precisely those places where relevance is traditionally assumed to pose the greatest threat to tractable computation.

Second, according to the proposal, modules themselves possess certain core characteristics that engender feasible computation: in particular, *domain specificity* and *informational encapsulation*. The rough idea is that by virtue of possessing either or both of these, modular mechanisms can avoid the sorts of tractability problems that (allegedly) plague nonmodular devices. In the remainder of this section I explain this suggestion in more detail. But first a terminological point: The term "module" is notoriously ambiguous;[4] and it is often unclear how theorists intend it to be understood. But since we are concerned primarily with how modularity helps address tractability problems, we can safely restrict our attention to those characteristics of modules that are supposed to resolve such problems: namely, domain specificity and encapsulation. In what follows, then, I adopt a minimal definition of modules as computational mechanisms that possess one or both of these characteristics.

2.1 *Domain Specificity and Feasible Computation*

What is domain specificity and how is it supposed to engender feasible computation? To a first approximation, a mechanism is domain specific if it operates only in a highly restricted cognitive domain.[5] Standard candidates include mechanisms for face recognition, language, and arithmetic. There are, however, at least two broad views about cognitive domains that give rise to different conceptions of domain specificity. According to the first, the domain of a cognitive mechanism is the class of

4. See Segal (1996) and Samuels (2000) for discussions of the various uses of "module" in cognitive science.

5. It should go without saying—though I'll say it anyway—that the notion of domain specificity *admits of degree* and that researchers who use the notion are interested in whether we possess mechanisms that are domain specific to some *interesting* extent. The same points also apply to the notion of informational encapsulation.

representations that it can take as input: its *input domain*. On this conception of domains, a cognitive mechanism is domain specific to the extent that it can only take as input a highly restricted range of representations. According to the second conception of cognitive domains, the domain of a mechanism is the *task* (or function) that it performs: its *task domain*. On this conception of domains, a mechanism is domain specific if it is dedicated to performing a highly restricted task.

Why suppose that domain specificity in either of these senses engenders feasible computation? The claim cannot be that domain specificity is *sufficient* for tractability, since many of the paradigms of intractable computation—such as algorithms for solving the traveling salesman problem—are very domain specific indeed.[6] Nevertheless, if a mechanism is sufficiently domain specific, then it becomes possible to utilize a potent design strategy for reducing computational load, namely, to build into the mechanism substantial amounts of information about the domain in which it operates. This might be done in a variety of ways. It might be only implicit in the organization of the mechanism, or it might be explicitly represented; it might take the form of rules or procedures or bodies of propositional knowledge and so on. But however this information gets encoded, the key point is that a domain-specific mechanism can be *informationally rich* and, as a result, capable of rapidly and efficiently deploying those strategies and options most relevant to the domain in which it operates. Such mechanisms thereby avoid the need for computationally expensive search and assessment procedures that might plague a more general-purpose device. For this reason, domain specificity has seemed to many a plausible candidate for reducing the threat of combinatorial explosion without compromising the reliability of cognitive mechanisms (Sperber, 1994; Tooby & Cosmides, 1992).

2.2 *Informational Encapsulation and Feasible Computation*

I turn now to the notion of informational encapsulation. According to the standard definition, an encapsulated cognitive mechanism or faculty is one that "has access, in the course of its computations, to less than all of the information at the disposal of the organism whose cognitive faculty it is" (Fodor, 1987, p. 25). Paradigmatic examples—such as mechanisms for length perception or phonological processing—cannot draw upon the full range of the organism's beliefs, goals, and intentions. In contrast, a highly *unencapsulated* mechanism—paradigmatically for reasoning—would be one that has access to (virtually) all of our beliefs, goals, and intentions (Fodor, 1983; Stanovich, 1999).

A number of further comments are in order. First, although it is not uncommon to confound informational encapsulation and domain specificity (in particular with regard to the specificity of *input* domains), they are distinct properties. Both concern

6. In brief, the traveling salesman problem is to find the shortest path that a salesman can take between a network of cities. This is a highly specialized task and, moreover, the inputs to the process—the names of cities and representations of inter-city distances—are highly restricted as well. Yet it is notoriously hard to solve in a computationally tractable manner. This suggests that domain specificity is not plausibly viewed as sufficient for tractability.

the *access* that a mechanism has to representations. Yet the kind of access is quite different. Input-specificity concerns the class of representations that a mechanism can take as input: that "trigger" it or "turn it on." In contrast, the informational encapsulation of a mechanism concerns the class of representations that it can use as a resource once it has been so activated. Paradigmatically, encapsulation concerns the information encoded in memory that the mechanism is able to consult in the course of providing solutions to the particular inputs that it receives.

Second, encapsulation proper is not just any sort of restriction on access. Rather, it is supposed to be *architecturally* imposed. Minimally, this implies the following. First, encapsulation is a relatively enduring characteristic of the device. Second, it is not a mere product of *performance* factors, such as fatigue, lack of time or lapses in attention. Finally, and most important for my purposes, the encapsulation of a device is supposed to be *cognitively impenetrable* (Pylyshyn, 1984). To a first approximation: it is not a property of the mechanism that can be changed as a result of alterations in the beliefs, goals, and other representational states of the organism. Or roughly equivalently: it is not a property of the mechanism that can be changed by *psychological processes* alone.

Third, although there are various ways encapsulation might be architecturally imposed, the standard suggestion is that encapsulated mechanisms have access to only the information contained within a restricted, *proprietary database*. One important implication is that such mechanisms are unable to deploy information located elsewhere in the system even when that information is relevant to the task at hand. Suppose, for example, that mechanisms for face recognition only have access to a database of previously encountered faces. Such a device would be unable to utilize other sorts of information—for example, geographic or autobiographical information—even though it might sometimes be highly relevant to the task of recognizing faces

Finally, it is worth noting an ambiguity in the standard definition of encapsulation between a synchronic and a diachronic reading:

A mechanism M is *synchronically encapsulated* if, at any time, there is at least some (kind of) information possessed by the organism that is inaccessible to M.

A mechanism M is *diachronically encapsulated* if there is some (kind of) information that is inaccessible to M, not merely at some particular time but over a long period—paradigmatically the entire history of the mechanism.

I assume for two reasons that it is the *diachronic* notion that should concern us here. First, the paradigmatic examples of encapsulated modules are clearly diachronically encapsulated. So, for example, the perceptual mechanisms implicated in the production of persistent illusions—such as the Muller-Lyer or phi phenomenon—are not merely synchronically encapsulated with respect to our beliefs about the illusory phenomena.[7] (It is not as if, for example, two years hence

7. In the case of the Muller-Lyer illusion, the mechanisms responsible for visual length perception do not have access the belief that, contrary to appearances, the lines are of identical length.

they might access the relevant beliefs and the illusions dissipate.) Rather, the claim is that such beliefs are never accessible to the mechanism. Second, the synchronic notion is *too liberal* and classifies as encapsulated mechanisms that would not normally be counted as such. So, for example, it will count as encapsulated (1) any deterministic computational device that does not engage in exhaustive memory search,[8] and (2) any reasoning mechanism whose access to information is mediated via a limited working memory.[9] But not all such systems would ordinarily be construed as encapsulated.

How, then, is encapsulation supposed to facilitate feasible computation? As with domain specificity, encapsulation is not *sufficient* for feasibility; and again the traveling salesman illustrates the point. Algorithms designed to solve this task typically have access to only the information contained in the input to the process; yet they are computationally very expensive indeed. Even so, there are two plausible explanations of how encapsulation might engender tractability: a superficial and a deeper one.

According to the superficial explanation, encapsulation reduces computational load in two ways. First, because the device only has access to a highly restricted database or memory, the costs incurred by memory search are considerably reduced. (There just isn't that much stuff over which the search can be performed.) Second, by reducing the range of accessible items of information, there is a concomitant reduction in the number of relations *between* items—paradigmatically, relations of confirmation and relevance—that can be computed.

Yet one might reasonably wonder what all the fuss is about. After all, computer scientists have generated a *huge* array of methods—literally hundreds of different search and approximation techniques—for reducing computational overheads (Russell & Norvig, 2003). What makes encapsulation of *particular* interest? Here's where the deeper explanation comes into play. Most of the methods that have been developed for reducing computational load require that the implementing mechanisms treat the assessment of relevance as a *computational* problem. Roughly: they need to implement computational procedures that select from the available information some subset that is estimated to be relevant. In contrast, encapsulation is supposed to obviate the need for such computational solutions. According to this view, an encapsulated device (at least paradigmatically) only has access to a very small amount of information. As a consequence, it can perform (near) exhaustive search on whatever information it can access, and thereby avoid the need to assess

8. Consider, for example, a domain-general reasoning device with sole access to a general encyclopedic memory system that contains all the information possessed by the organism of which it is a part. Such a reasoning mechanism would ordinarily be construed as a paradigm of nonmodularity. But if it were deterministic and also deployed procedures (e.g., heuristics) for delimiting which portion of the database to access, then it would, on the synchronic reading, count as encapsulated.

9. Consider a mechanism that can access any part of encyclopedic memory but does so via a working memory of the Miller "magic number seven" variety. Since at any specific time it would only have access to seven items of information (give or take a bit), it would, on the synchronic reading, be highly encapsulated.

relevance. There is a sense, then, in which highly encapsulated devices avoid the relevance problem altogether (Fodor, 2000).

Assume that the above is correct—that domain specificity and informational encapsulation help engender feasible computation—then it should be clear how MM is supposed to address the threat that intractability poses for CTM. What it does is ensure that reasoning mechanisms are architecturally constrained with respect to what options and items of information they can consider. Yet it is one thing to claim that modularity is an important way to engender tractability and quite another to claim that it is the *only* plausible way. The former is compatible with a broad range of architectural hypotheses—including a psychological rationalism that posits radically nonmodular reasoning mechanisms—while the latter demands that computationalists adopt an extreme form of MM. In the following sections, I consider arguments that purport to establish this stronger claim—the intractability thesis—and show that they are unsatisfactory.

3 Informational Impoverishment

Perhaps the most prominent argument for IT is one made popular by the evolutionary psychologists Leda Cosmides and John Tooby (Cosmides & Tooby, 1994). The argument proceeds from the assumption that a nonmodular, hence domain-general, mechanism "lacks any content, either in the form of domain-specific knowledge or domain-specific procedures that can guide it towards the solution of problems" (Cosmides & Tooby, 1994, p. 94). As a consequence, it "must evaluate all the alternatives it can define" (p. 94). But as Cosmides and Tooby observe, such a strategy is subject to serious intractability problems, since even routine cognitive tasks are such that the space of alternative options tends to increase *exponentially*. Nonmodular mechanisms would thus seem to be computationally intractable: at best intolerably slow and at worst incapable of solving the vast majority of problems they confront.

Though frequently presented as an objection to non-MM accounts of cognitive architecture, this argument is really only a criticism of theories that characterize cognitive mechanisms as suffering from a particularly extreme form of informational impoverishment. Any appearance to the contrary derives from the stipulation that domain-general mechanisms possess no specialized knowledge. But this conflates claims about the need for *informationally rich* cognitive mechanisms—a claim that I do not wish to deny—with claims about the need for modularity; and although modularity is one way to build specialized knowledge into a system, it is not the only way. Another is for nonmodular devices to have access to bodies of specialized knowledge. Indeed, it is commonly assumed by nonmodular—especially rationalist—accounts of central possessing that such devices have access to *huge* amounts of information. This is pretty obvious from even the most cursory survey of the relevant literatures. Fodor (1983), for example, maintains explicitly that nonmodular central systems have access to huge amounts of information; as do Gopnik, Newell, and many others who adopt a nonmodular conception of central systems (Gopnik & Meltzoff, 1997; Newell, 1990). The argument currently under discussion thus succeeds only in refuting a straw man.

4 Optimality

Another argument for IT turns on the claim that nonmodular reasoning mechanisms implement *optimization* processes. In this context, "optimization" refers to reasoning that broadly conforms to standards of *ideal rationality*, such as those characterized by Bayesian accounts of probabilistic inference or standard approaches to decision theory. There are a range of results that show such reasoning processes are computationally very expensive indeed (Osherson, 1995);[10] and for this reason they are commonly termed *unbounded* or even *demonic* conceptions of reasoning (Gigerenzer, 2001; Simon, 1972). So if advocates of nonmodular reasoning mechanisms are committed to optimization, then the view they endorse would be subject to serious intractability worries as well.

It is not at all clear to me that anyone *explicitly* endorses the above argument, though it is strongly suggested by some recent discussions of nonmodular reasoning architectures (Dietrich & Fields, 1996; Gigerenzer, 2001; Gigerenzer et al., 1999). The argument is not, however, a good one. Though optimal reasoning is (at least in the general case) intractable,[11] nonmodularists are in no way committed to such a view of human reasoning. What *is* true is that for a mechanism to optimize it needs to be *unencapsulated*, hence nonmodular; and this is because, as ordinarily construed, optimization demands the updating of all one's beliefs in the light of new information. But the converse is not true: an unencapsulated mechanism need not be an optimizer. On the contrary, since the inception of artificial intelligence (AI) it has been commonplace to combine a nonmodular conception of reasoning with the explicit denial of optimization. Consider, for example, Newell and Simon's seminal work on the general problem solver (GPS). As the name suggests, GPS was designed to apply across a very wide range of content domains without architectural constraint on what representations is could use. It is thus not plausibly viewed as modular. But, to use Simon's famous expression, it was designed to *satisfice*—to arrive at solutions that were *good enough*—not to optimize. Much the same could be said for many of the nonmodular, accounts of reasoning to be found in classical AI and cognitive science, including Laird and Newell's SOAR architecture (Newell, 1990). These are among the paradigm nonmodular approaches to cognition, yet they are in no way committed to optimization.

10. To use one well-known example, on standard Bayesian accounts, the equations for assessing the impact of new evidence on our current beliefs are such that if one's system of beliefs has n elements, then computing the new probability of a single belief, B, will require 2^n additions (Harman, 1986). Such methods thus involve an exponential growth in number of computations as a function of belief system size. To give some idea of just how expensive this is, on the hyperconservative assumption that we possess 100 beliefs, calculating the probability assignment of a belief B on the basis of new information will require the performance of more than 10^{30} additions, which is considerably more than the number of microseconds that have elapsed since the Big Bang!

11. Though there is lots of good research that aims to discover tractable methods for applying ideal standards of rationality to interesting—but restricted—domains. See, for example, the literature on Bayesian networks (Pearle, 1988).

5 Exhaustive Search

Still, even if optimization *as such* is not a problem for nonmodular accounts of reasoning, it might still be that there are properties of optimal reasoning to which the nonmodularist *is* committed and that these properties are sufficient to generate intractability problems. Exhaustive search is perhaps the most plausible candidate for this role. The rough idea is that nonmodular reasoning mechanisms must perform *exhaustive* searches over our beliefs. But, given even a conservative estimate of the size of any individual's belief system, such a search would be unfeasible in practice. In which case, it would seem that nonmodular reasoning mechanisms are computationally intractable.

Again, it's not at all clear to me that anyone really endorses this argument, though some have found it hard not to view advocates of nonmodular central systems as somehow committed to exhaustive search (Carruthers, 2004; Glymour, 1985). Yet this view is incorrect. What the nonmodularist does accept is that unencapsulated reasoning mechanisms have *access* to huge amounts of information—paradigmatically, all the agent's background beliefs. But the relevant notion of access is a *modal* one. It concerns what information—given architectural constraints—a mechanism *can* mobilize in solving a problem. In particular, it implies that any background belief can be used, not that the mechanism *in fact* mobilizes the entire set of background beliefs—that is, that it engage in exhaustive search. And this is just as well, since it would be absurd to hold a nonmodular view of reasoning if it implied exhaustive search (Fodor, 1985).

Of course, the fact that the nonmodularist does not *endorse* the claim that central systems engage in exhaustive search is perfectly consistent with there being an argument that shows such processes would need to occur if a nonmodular account of reasoning were true. In the next section, I consider a recent argument from Fodor (2000) that has been widely interpreted by advocates of MM as supporting this conclusion.

6 The Locality Argument

Fodor's argument is a complex one, but the core idea can be framed in terms of a tension between two claims. The first is that classical computational processes are *local* in roughly the following sense: what computations apply to a particular representation is determined solely by its *constituent structure*—that is, by how the representation is constructed from its parts (Fodor, 2000, p. 30). To take a very simple example, whether the addition function can be applied to a given representation is solely determined by whether or not it has the appropriate syntactic structure—for example, whether it contains a permissible set of symbols related by "+."

The second claim is that much of our reasoning is *global*, in that it is sensitive to *context-dependent* properties of the *entire* belief system. In arguing for this, Fodor focuses primarily on abductive reasoning (or inference to the best explanation).[12]

12. Though he thinks that the same considerations apply to decision-making or planning as well.

Such inferences routinely occur in science and, roughly speaking, consist in coming to endorse a particular belief or hypothesis on the grounds that it constitutes the best available explanation of the data. One familiar feature of such inferences is that the relative quality of hypotheses are not assessed merely in terms of their ability to fit the data but also in terms of their simplicity and conservativism.[13] According to Fodor, however, these properties are not intrinsic to a belief or hypothesis but are global characteristics that a belief or hypothesis possesses by virtue of its *relationship* to a constantly changing system of background beliefs. The problem, then, is this: If classical computational operations are local, how could global reasoning processes, such as abduction, be computationally tractable?

Notice that if the above is correct, then a classical abductive process could not operate merely by looking at the hypotheses to be evaluated. This is because, by assumption, what classical computations apply to a representation is determined solely by its constituent structure, whereas the simplicity and conservativism of a hypothesis, H, depend not only on its constituent structure but its relations to our system of background beliefs, K. In which case, a classical implementation of abduction would need to look at both H *and* whatever parts of K determine the simplicity and conservativism of H. The question is: How *much* of K needs to be consulted in order for a classical system to perform reliable abduction? According to Fodor, the answer is that lots—indeed, very often, the totality—of the background will need to be accessed, since this is the "only *guaranteed* way" of classically computing a global property. But this threatens to render reliable abduction computationally intractable. As Fodor puts its:

> Reliable abduction may require, in the limit, that the whole background of epistemic commitments be somehow brought to bear on planning and belief fixation. But feasible abduction requires in practice that not more than a small subset of even the relevant background beliefs are actually consulted. (2000, p. 37)

In short: if classicism is true, abduction cannot be reliable. But since abduction presumably is reliable, classicism is false.

If sound, the above argument would appear to show that classicism itself is untenable. So, why would anyone think it supports MM? The suggestion appears to be that MM provides the advocate of CTM with a way out: a way of avoiding the tractability problems associated with the globality of abduction without jettisoning CTM (Sperber, chapter 4 here; Carruthers, chapter 5 here). Fodor himself put the point as well as anyone:

> Modules are informationally encapsulated by definition. And, likewise by definition, the more encapsulated the informational resources to which a computational mechanism has access, the less the character of its operations is sensitive to global properties of belief systems. Thus to the extent that the information accessible to a device is architecturally constrained to a proprietary database, it won't have a frame

13. Very roughly: (1) one hypothesis is *simpler* (or more parsimonious) than another if it posits fewer entities/causes/parameters, and (2) one hypothesis is more *conservative* than another if it requires less revision to our belief system.

> problem and it won't have a relevance problem (assuming that these are different); not, at least, if the database is small enough to permit approximations to exhaustive searches. (2000, p. 64)

The modularity of central systems is thus supposed to render reasoning processes sufficiently *local* to permit tractable computation.

There are a number of serious problems with the above line of argument. One that I will *not* address here concerns the extent to which MM provides a satisfactory way of shielding CTM from the tractability worries associated with globality.[14] What I will argue, however, is that although simplicity and conservativism are plausibly context dependent, Fodor provides us with no reason whatsoever to think that they are *global* in any sense that threatens nonmodular versions of CTM.

First, when assessing the claim that abduction is global, it is important to keep firmly in mind the general distinction between normative and descriptive-psychological claims about reasoning: claims about how we *ought* to reason and claims about how we *actually* reason. This distinction applies to the specific case of assessing the simplicity and conservativism of hypotheses. On the normative reading, assessments of simplicity and conservativism ought to be global: that is, normatively correct assessments ought to take into consideration one's total background epistemic commitments. But of course it is not enough for Fodor's purposes that such assessments *ought* to be global. Rather, it needs to be the case that the assessments humans make are, *in fact*, global; and to my knowledge, there is no reason whatsoever to suppose that this is true.

A comparison with the notion of consistency may help to make the point clearer. Consistency is frequently construed as a normative standard against which to assess one's beliefs (Dennett, 1987). Roughly: all else being equal, one's beliefs ought to be consistent with each other. When construed in this manner, however, it is natural to think that consistency should be a global property in the sense that any belief ought to be consistent with the entirety of one's background beliefs. But there is absolutely no reason to suppose—and indeed some reason to deny—that human beings conform to this norm (Cherniak, 1986). Moreover, this is so in spite of the fact that consistency really does play a role in our inferential practices. What I am suggesting is that much the same may be true of simplicity and conservativism. When construed in a normative manner, it is natural[15] to think of them as global properties, but when construed as properties of the beliefs that figure in actual human inference, there is no reason to suppose that they accord with this normative characterization.

Second, even if we suppose that the simplicity and conservativism are global properties of actual beliefs, the locality argument still does not go through, since it turns on the implausible assumption that we are guaranteed to make successful assessments of simplicity and conservativism. Specifically, in arguing for the conclusion that abduction is computationally unfeasible, Fodor relies on the claim that "the only guaranteed way of Classically computing a syntactic-but-global property" is to take

14. Though see Samuels (forthcoming) for an extended discussion of this issue.
15. Though by no means mandatory.

"whole theories as computational domains" (2000, p. 36). But guarantees are *beside the point*. Why suppose that we always successfully compute the global properties on which abduction depends? Presumably we do not. And one very plausible suggestion is that we fail to do so when the cognitive demands required are just too great. In particular, for all that is known, we may well fail under precisely those circumstances that the classical view would predict—namely, when too much of a belief system needs to be consulted in order to compute the simplicity or conservativism of a given belief.

7 The Robot Argument

The final argument for IT that I will discuss consists in an induction from recent trends in AI and robotics (Carruthers, 2004; Goodie et al., 1999). The starting point for this argument is that if one wants to assess the computational feasibility of classical, non-MM architectures, then the repeated efforts of computer scientists to produce feasible intelligent systems—paradigmatically, robots—constitute an important source of evidence. According to the robot argument, however, research in the past decade or so has increasingly converged on one form or other of massive modularity. To mention just two examples, *behavior-based* approaches have had an enormous influence on robotics (Brooks, 1999) while so-called multiagent systems has been among the most rapidly developing areas of AI in recent years (Ferber). Moreover, so the argument continues, this convergence is largely a consequence of the problems that researchers encounter in trying to develop practically feasible real-time systems. Roughly: nonmodular systems have in practice turned out to be unfeasible, whereas modular ones have been far less prone to such problems. It would seem, then, that the pattern of successes and failures in AI and robotics provide us with good—albeit nondemonstrative—grounds for accepting MM (Carruthers, 2004; Gigerenzer, 2001, p. 43).

The general form of the argument is a perfectly respectable one. Indeed, if CTM is true, then careful and accurate analysis of contemporary AI and robotics might have much to tell us about the architecture of human cognition. My concern, however, is that the analysis on which the robot argument depends is neither careful nor accurate. What *is* true is that research—especially in robotics—has converged on the need for a kind of module that Rodney Brooks calls *reactive behaviors*. Such modules are a commonplace feature of contemporary robots and are designed to generate rapid, real-time responses—such as avoidance behavior—to prespecified sets of environmental conditions (Brooks, 1999; Bryson, 2000). Moreover, the popularity of these kinds of modules is, in large measure, a response to the dramatic failure of a less modular approach to robotics—the *sense-model-plan-act* paradigm—which assumed that virtually all robot behavior should be mediated by the activity of a general-purpose planning system (Bonasso et al., 1998; Brooks, 1999).[16]

But this alone does not constitute an argument for *massive* modularity. What the Robot Argument needs to show is that there has been convergence on the idea that

16. The most famous product of the SMPA paradigm was Shakey, the Stanford Research Institute robot (Nilsson, 1984).

central systems are modular; and no such convergence of opinion exists within the AI community. Even in robotics where tractable, real-time performance is of a premium, the dominant kind of computational architectures—so-called three-layered or hybrid systems—incorporate a *deliberative* layer of nonmodular mechanisms for planning and world-modeling quite similar to those that figured in the discredited sense-model-plan-act paradigm (Bonasso et al., 1998; Gat, 1998).[17] In contrast to earlier proposals, however, the hybrid approach incorporates two additional design principles. First, the system has a *reactive layer* that contains a multitude of Brookian modules that enable it to respond rapidly to environmental contingencies. Second, in large measure because of this, the reasoning mechanisms within the deliberative layer of the system can be "decoupled" from real-time activities—such as obstacle avoidance—and instead deployed to generate solutions to complex, informationally intensive, decision-making tasks. The result of combining these various features is a kind of system that is both more flexible than those composed solely of reactive behaviors and more capable of real-time performance than those that assign a larger role to reasoning mechanisms (Russell & Norvig, 2003).

8 Conclusion

The main burden of this chapter has been to argue that we currently possess no good reason to accept IT, hence no reason to endorse MM on the grounds of tractability. Thus formulated, the project is a largely negative one. But my discussion of the arguments for IT also yield a series of positive suggestions about the general properties that the kind of computational architecture proposed by psychological rationalists is likely to possess. None of these suggestions are, I think, particularly surprising; and many of them are utterly commonplace in those regions of cognitive science most concerned with the computational implementation of cognitive processes. In view of the confusions that surround debate over MM, however, it is perhaps worth concluding this chapter by assembling these claims.

1. *Informational richness* (sec. 3). In view of the sorts of problems that Cosmides and Tooby pose for informationally impoverished cognitive mechanisms, it seems highly likely that nonmodular reasoning systems will almost invariably possess specialized bodies of knowledge about the domains in which they operate. Indeed, on a rationalist construal of such systems, they are likely to possess lots of innate, domain-specific information.
2. *Suboptimality* (sec. 4). There are overwhelming reasons to think that "optimal" reasoning processes of the kind associated with ideal theories of rationality are computationally intractable. In view of this, the reasoning processes subserved by nonmodular central systems will be suboptimal or bounded.

17. Another prominent example of a nonmodular reasoning system in AI is the procedural reasoning system (PRS; Georgeff & Lansky, 1987; d'Inverno et al., 1997).

3. *Limited search* (sec. 5). Nonmodular central systems will also not engage in exhaustive search of the information available to them, since, given a reasonable estimate of the size of a human belief system, it would pose serious tractability problems.
4. *Limited sensitivity to the global properties of cognition* (sec. 6). Fodor is right to claim that a computational, reasoning mechanism would be intractable if it were both highly reliable and sensitive to global properties of the belief system. But as I argued in section 6, this does not imply there are no nonmodular, computational mechanisms for reasoning. All that follows is that our reasoning is not all that sensitive to global properties after all; and this is, I maintain, an entirely sensible position for the advocate of nonmodular reasoning mechanisms to adopt.
5. *Autonomy from real-time control of action.*(sec 7). If we are to take seriously the last two decades of research in robotics, it would seem that incorporating nonmodular reasoning mechanisms into a cognitive system while avoiding practical tractability problems requires that the operations of such devices are *decoupled* from fine-grained, real-time behavioral operations. Instead, nonmodular reasoning mechanisms are likely to operate at a more coarse-grained temporal scale in order to make crucial decisions, construct relatively long-term plans, and provide rich representations of the world that can aid in the pursuit of the agents epistemic and practical goals.
6. *The need for reactive behaviors* (sec. 7). Since human beings do succeed in responding in real time to environmental conditions, claim 5 implies that nonmodular reasoning mechanisms need to be located within an architecture that contains other mechanisms that are responsible for the production of fine-grained, real-time responses. This claim is not at all contentious among nativists, since, as mentioned earlier, they almost invariably assume that humans possess a variety of input systems and output systems that play this role. Nonetheless, I would suggest that the past few decades of research in robotics makes it plausible to posit an additional kind of mechanism that aids in the production of real-time behavior: modular "reactive behaviors" that produce rapid behavioral responses to stereotypic environmental conditions.

Where do these comments leave us? What I think they provide is a rough sketch of some characteristics that a cognitive architecture of the kind advocated by psychological rationalists would be likely to possess. Is there any reason to suppose that this rationalist view is *preferable* to a thoroughgoing MM that denies the existence of nonmodular reasoning mechanisms? Clearly, I have provided no argument for such a conclusion in the foregoing discussion. For what its worth, however, I suspect that a non-MM account of cognition is likely to do far better at explaining the peculiar flexibility of human behavior and cognition. But an explanation of why this is so will have to wait for another day.