Running head: BETWEEN COMPUTATIONAL AND ALGORITHMIC

Rational use of cognitive resources:

Levels of analysis between the computational and the algorithmic

Thomas L. Griffiths and Falk Lieder

Department of Psychology

University of California, Berkeley

Noah D. Goodman

Department of Psychology

Stanford University

Word count: 4102

Address for correspondence:
Tom Griffiths
University of California, Berkeley
Department of Psychology
3210 Tolman Hall # 1650
Berkeley CA 94720-1650
E-mail: tom_griffiths@berkeley.edu
Phone: (510) 642 7134
Fax: (510) 642 5293

Abstract

Marr's levels of analysis – computational, algorithmic, and implementation – have served cognitive science well over the last 30 years. But the recent increase in the popularity of the computational level raises a new challenge: how do we begin to relate models at different levels of analysis? We propose that it is possible to define levels of analysis that lie between the computational and the algorithmic, providing a way to build a bridge between computational- and algorithmic-level models. The key idea is to push the notion of rationality, often used in defining computational-level models, deeper towards the algorithmic level. We offer a simple recipe for reverse-engineering the mind's cognitive strategies by deriving optimal algorithms for a series of increasingly more realistic abstract computational architectures which we call "resource-rational analysis".

Rational use of cognitive resources:

Levels of analysis between the computational and the algorithmic

Marr and Poggio (1977) proposed a key methodology of cognitive science: proceed by forming *levels of analysis*, that can, and should, be studied independently. The three levels of analysis identified by Marr (1982) have stood the test of time, still being the canonical scheme for organizing formal analyses of information processing systems over thirty years after they were first introduced. Marr's three levels correspond to an abstract characterization of the computational problem being solved (the "computational" level), the algorithm executing that solution (the "algorithmic" level), and the hardware implementing that algorithm (the "implementation" level).

While the concrete contribution of Marr was three specific levels; the deeper contribution made by Marr and Poggio was the idea that it is valid, fruitful, and even necessary to analyze cognition by forming abstraction barriers (which result in levels of analysis). Subsequent work has preserved this idea, even as it has offered refinements and alternatives to Marr's scheme (e.g., Pylyshyn, 1984; Newell, 1982; Anderson, 1990). These other schemes generally preserve the distinctions between levels identified by Marr, but make a finer distinction between algorithms and the cognitive architecture that executes them, essentially dividing the algorithmic level into two parts. In this paper we explore another application of the methodology introduced by Marr and Poggio, refining Marr's levels by considering the possibility of levels of analysis that lie between the computational and the algorithmic.

The longevity of Marr's scheme suggests that we should have a high bar for considering modifications to it. We believe that there are two important reasons to begin to explore what lies between the computational and algorithmic levels. First, the recent rise in the popularity of rational models of cognition and Bayesian analysis (e.g., Anderson, 1990; Chater & Oaksford, 1999; Tenenbaum, Kemp, Griffiths, & Goodman, 2011) has resulted in theories of a wide range of different aspects of human cognition that are framed at the computational level. This form of

theorizing is relatively new to cognitive psychology, which has traditionally tended to focus on the cognitive strategies – i.e. algorithms – by which problems are solved. As a consequence, we sometimes have explanations of a given phenomenon expressed at different levels of analysis, and little idea as to how those explanations should be evaluated or even if they are incompatible. Marr made it clear that he expected theories at different levels of analysis to inform one another, but left open the question of how this might play out in practice. Building a bridge between the computational and algorithmic levels is a step towards being able to answer questions about whether theories at these different levels are compatible, and a tool for generating theories at one level from theories at another. For example, it might allow us to be clear when the assumptions of a Bayesian model (at the computational level) are inconsistent with those of a connectionist model (at the algorithmic level), addressing an ongoing debate between proponents of these two approaches (Griffiths, Chater, Kemp, Perfors, & Tenenbaum, 2010; McClelland et al., 2010). Even more importantly, it might give us a generic recipe for constructing connectionist and other psychological process models that approximate the ideal solution expressed in a Bayesian model, producing testable claims about cognitive processes.

A second reason to consider levels of analysis between the computational and algorithmic is to clarify and organize research that has begun to explore this space. A recognition that human cognition necessarily involves making approximations or otherwise limiting rational solutions has been a common theme in psychological theory since Simon (1955) introduced the notion of "bounded rationality". Anderson's (1990) influential framework for "rational analysis" explicitly includes a step where cognitive constraints are considered when defining a rational model. But what constraints should we assume, and when should we assume them? We propose that the consideration of these constraints should be done somewhat independently of the definition of computational-level models, but that the principle of rationality provides the key to exploring the new levels of analysis that result.

Distinguishing between computational problems and the resources available for solving

them provides a way to build a bridge between the computational and algorithmic levels. While there is presumably only one true cognitive architecture, there can be many different abstractions – capturing different sets of constraints – that provide complementary insights into human cognition. We will argue that a series of levels should be considered, starting with weak assumptions about how the mind computes and then gradually strengthening those assumptions to converge towards solutions that resemble the outcome of traditional algorithmic-level analyses of human cognition. In analogy to *abstract machines* in theoretical computer science, we will refer to the assumed abstractions from the mind's cognitive resources as *abstract computational architectures* and define them by a set of basic operations and their costs. Abstract computational architectures are models of mental computation in the sense in which the Turing machine is a model of how computers process information.

The first stop on our brief tour between levels is the computational level itself, revisiting the role that rationality has come to play in defining this level. We then consider the strategy of bridging levels by constructing "rational process models" that push the notion of rationality towards the algorithmic level by postulating cognitive mechanisms that resemble the approximation algorithms that statisticians and computer scientists use to solve the problem identified by the computational level theory. This is a heuristic for scientific discovery, but it is merely a heuristic. This brings us to a stronger application of rationality to cognitive modeling – "resource-rational analysis", which derives the strategy that makes optimal use of finite computational resources. Finally, we use these notions to lay the groundwork for defining a set of levels that lie between the computational and the algorithmic.

Computation and rationality

Marr's notion of the computational level has been criticized for being poorly named and difficult to apply (Arbib, 1987; Anderson, 1990). The focus of this level is not on computation *per se*, in the formal sense defined by Turing (1937) or the informal sense of carrying out calculations,

but rather on the structure of the abstract problem being solved and the nature of its ideal solution. The problem and the solution can be defined purely mathematically, with no requirement of conforming to any notion of computation. The central question is what *function* an aspect of cognition serves, leading recent proponents of computational-level analysis to call this approach a "function-first" strategy (Griffiths et al., 2010).

Anderson (1990) made the important observation that an implicit adaptationist principle underlies the idea of computational-level analysis: we should expect ideal solutions to the problems that human beings face to provide insight into human cognition only to the extent that human minds solve those problems well. Making this adaptationist principle explicit led Anderson to suggest a refined strategy for finding computational-level explanations, founded on the principle of rationality, which he states as the idea that "the cognitive system operates at all times to optimize the adaptation of the behavior of the organism" (Anderson, 1990, p. 28). He then laid out a six-step program for applying this system (Anderson, 1990, p. 29):

1. Precisely specify what are the goals of the cognitive system.
2. Develop a formal model of the enviornment to which the system is adapted.
3. Make the minimal assumptions about computational limitations.
4. Derive the optimal behavioral function given items 1 through 3.
5. Examine the empirical literature to see if the predictions of the behavioral function are confirmed.
6. If the predictions are off, iterate.

There are several interesting features of this program – not least that there is no stopping criterion, which would concern critics of rational models (Jones & Love, 2011; Bowers & Davis, 2012) – but our focus here will be on Step 3.

What role should constraints play in computational-level theories? If the goal is really to characterize the problem being solved and its ideal solution, computational limitations have little role to play. Rather, they seem like a device for bringing that ideal solution into closer alignment

with human cognition. If human beings have only finite computational resources – processing power, memory, and attention – to be used in solving a problem, then the best they can do is to approximate the ideal solution using those limited resources. But taking these resources into account is already a move towards the algorithmic level, considering the kinds of cognitive processes that might be available to execute a computational-level solution.

Rather than blurring these lines and building constraints into computational-level theories, we suggest a different approach: define the computational-level theory without considering limitations on its execution, and then explore the consequences of those limitations as a further analysis that brings us closer to an algorithmic-level theory (see Figure 1). Various proposals about limitations – or alternatively abstract computational architectures – provide us with levels of analysis between the computational and the algorithmic, and the principle of rationality provides us with a methodology for developing models at those intermediate levels. In the remainder of the paper, we discuss two recent lines of research that illustrate this approach.

Rational process models

The main role that computational limitations played in the theories presented by Anderson (1990) was in justifying approximations to challenging probabilistic computations. For example, in Anderson's rational model of categorization (later appearing in Anderson, 1991), objects are assigned to clusters by considering each object in turn and maintaining only a single clustering, rather than by evaluating or averaging over all possible clusterings. This is certainly more plausible as a cognitive model, but building in this constraint misses two opportunities: to compare human behavior to the ideal predictions from an unconstrained computational-level account, and to explore the consequences of adopting different approximation schemes.

The computational challenges posed by rationality are not just a problem for human minds – they are also faced by computer scientists and statisticians who work with complex probabilistic models. These scientists have developed a variety of strategies for approximating the resulting

computations, and those strategies are a source of hypotheses about the cognitive processes by which the mind approximates the optimal solutions identified by computational-level theories. The result is what has been dubbed a "rational process model" (Shi, Griffiths, Feldman, & Sanborn, 2010; Sanborn, Griffiths, & Navarro, 2010).

Anderson's rational model of categorization could be interpreted as a rational process model, using an approximation strategy based on a greedy maximization algorithm. But there are other approximation algorithms that are directly applicable to the problem it was used to solve. Sanborn, Griffiths, and Navarro (2006; 2010) showed how two algorithms commonly used to perform probabilistic inference in related models in statistics – Markov chain Monte Carlo and particle filters –could be used to approximate the average over all clusterings in Anderson's model. Both are Monte Carlo algorithms, producing an approximation that is based on a sample of clusterings. With a large sample, the algorithms give a close approximation to the ideal rational model, making it possible to compare this model against human behavior. With a small sample, they deviate from this ideal in systematic ways, producing biases (such as order effects) that are easy to compare against human performance. Such comparisons yield clues about the computational constraints that might be relevant to explaining human behavior.

Rational process models are valuable not just as a source of hypotheses about how people might go about approximating challenging probabilistic computations, but as a way to reinterpret existing proposals about psychological processes. For example, Shi and colleagues (Shi, Feldman, & Griffiths, 2008; Shi et al., 2010) showed that exemplar models – a popular class of cognitive process models based on memorizing examples and recalling those examples most similar to a current stimulus – can be interpreted as approximating Bayesian inference using a Monte Carlo algorithm known as importance sampling. This connection establishes a direct link between levels of analysis, and a simple way to define a process model that implements a Bayesian model (even in complex settings such as intuitive physics, e.g., Sanborn, Mansinghka, & Griffiths, 2013). It is also a source of interesting connections between Bayesian inference and connectionist architectures

(Shi & Griffiths, 2009).

Considering different algorithms for approximating the optimal solution can inspire abstract computational architectures that are useful for defining models that lie between the computational and algorithmic levels. For instance, drawing one sample from the posterior probability distribution per unit time is the elementary operation of one such abstract computational architecture, namely the bounded sampling agent. That is, we can consider a class of models that approximate rational solutions under the assumption that samples from relevant probability distributions are available. Digging deeper, we can consider specific schemes for generating those samples – Markov chain Monte Carlo, particle filters, or importance sampling – as providing a set of abstract computational architectures that each have their own constraints in terms of time, memory, and the conditions under which they succeed and fail. By exploring the properties of these abstract computational architectures, we can work towards having a well-understood, standardized set of methods for moving computational-level models in the direction of proposals about the algorithmic level.

Process models based on approximation algorithms (such as Monte Carlo methods) are rational in that they are generated from a computational-level solution to a problem of cognition. However, this is a weak form of rationality – rationality by derivation, or rationality in asymptote (similar to the notion of "calculative rationality" presented in Russell, 1997). Another way to push the principle of rationality below the computational level is to consider algorithms that are optimal solutions when a specific abstract architecture, such as sampling, is used. For example, rather than merely assuming a Monte Carlo approximation, we might ask what the *best* way to use a set of random samples might be. In the next section we lay out a methodology based on this principle.

Resource-rational analysis

We propose a concrete methodology—which we call *resource-rational* analysis—for establishing and studying levels of analysis between the computational and the algorithmic level. Like Anderson's schema for developing rational models of cognition, a resource-rational can be

developed in four simple steps:

1. *Function*: Formalize the problem that the cognitive mechanism solves and characterize the optimal solution (computational level of analysis).
2. *Model of mental computation*: Posit a family of algorithms that approximate the optimal solution and their computational costs. This can be done by defining an abstract computational architecture by a set of elementary operations (e.g. to draw one sample from the posterior distribution) and their costs (e.g. based on execution time).
3. *Optimal resource allocation*: Find the algorithm in this class that optimally trades off approximation accuracy against time and other resources (e.g. by maximizing expected utility per unit time or the value of computation; see below).
4. *Evaluate and refine*: Compare the model's predictions to human behavior. Revise the functional characterization (Step 1), or the model of mental computation (Step 2) and the resulting algorithm (Step 3) accordingly. Alternatively, one might proceed to the next level below by modeling how the basic operations might be approximated or considering additional resource constraints.

By explicitly positing a class of possible algorithms and a cost to the resources used by these algorithms, we can invoke an optimality principle to derive the algorithm that will be interpreted as a rational process model. Resource-rationality is thus a principle for analyzing an information processing system at an intermediate level defined by an idealized model of mental computation. This method enables us to reverse-engineer not only the problem that a system solves (computational level of analysis) but also its computational resources. The process might converge to a cognitive architecture similar to *ACT-R* (Anderson et al., 2004), *EPIC* (Meyer & Kieras, 1997), or *SOAR* (Laird, 1987), or to something entirely different.

Importantly, the transition from Step 1 to Step 2 need not be ad-hoc for a particular task or cognitive ability. First, there are very general "algorithm recipes" that take a (somewhat arbitrary) computational-level model and generate a (large) family of approximate solution algorithms. Many

such "compilation" methods exist in the computer science literature that we can use to generate new levels of analysis for particular cognitive abilities. For instance, particle filtering (mentioned above) is a general approach that leads to specific algorithms carrying in the number of particles, the resampling criteria, and so on (Abbott & Griffiths, 2011). This results in infinitely many algorithms that can have qualitatively different properties (e.g., one particle vs. millions of particles). Steps 2 and 3 allow us to find reasonable points within this algorithmic profusion, which we may then compare to human behavior. To the degree that evolution, development, and learning have adapted the system to make optimal use of its finite computational resources, resource-rational analysis can be used to derive the system's algorithm from assumptions about its computational resources. Second, the cost of computation can be formally derived from the opportunity cost of time (Vul, Goodman, Griffiths, & Tenenbaum, 2014).

To formalize Steps 2 and 3, a sequence $\mathbf{c}$ of elementary operations ($\mathcal{C}$) is resource-rational, if it maximizes the expected utility of the result minus the cost of computation (Lieder, Griffiths, & Goodman, 2013). A general version of this definition can be formulated in terms of the value of computation (VOC) (Lieder, Goodman, & Huys, 2013):

$$\mathbf{c} = \arg\max_{\mathbf{c}\in\mathcal{C}^n} \mathrm{VOC}(\mathbf{c})$$
$$\mathrm{VOC}(\mathbf{c}) = \mathbb{E}_{P(B|c)}\left[\max_a \mathbb{E}_{P(Q,S|B)}\left[Q(s,a)\right]\right] - \mathrm{cost}(\mathbf{c}),$$

where $Q(s,a)$ is the unknown expected cumulative reward of taking action a in the current state s, and B is the agent's belief about Q and s. Each computation $c \in \mathcal{C}$ costs time and energy ($\mathrm{cost}(c)$) but may improve the system's decision by changing its belief B. The formalization of resource rationality in terms of the value of computation was inspired by earlier research in artificial intelligence (Russell, 1997; Horvitz, 1988; Russell & Wefald, 1991a; Hay, Russell, Tolpin, & Shimony, 2012).

Resource-rational analysis has been applied to decision-making (Vul, Goodman, Griffiths, & Tenenbaum, 2009; Lieder, Hsu, & Griffiths, in press), prediction (Lieder, Griffiths, & Goodman, 2013) and planning (Lieder, Goodman, & Huys, 2013), yielding predictions that are surprisingly different from the optimal solution to the problem defined at the computational level. Vul et al. (2014) analyzed an abstract computational architecture whose elementary operation was to sample from the posterior distribution on whether or not an action will yield a reward. The analysis found that as the cost of sampling increases, the resource-rational algorithm changes from expected utility maximization (i.e., infinitely many samples) to probability matching (i.e. using a single sample). For a wide range of realistic computational costs the optimal number of samples was surprisingly close to one, indicating that–in contrast to expected-utility theory–resource-rationality is consistent with probability matching (Herrnstein & Loveland, 1975).

The algorithms posited in Step 2 of our recipe may involve computationally difficult steps that we assume to be solved ideally. To borrow a notion from theoretical computer science, we might assume that the abstract computational architecture has access to an "oracle" that perfectly solves a complex problem. These steps are potential loci for the next resource-rational approximation, as indicated in Step 4. For instance the availability of perfect posterior samples was assumed in the analysis performed by Vul and colleagues, but generating even a single perfect sample from the posterior distribution can be a very hard problem. This suggests forming a further level of analysis by replacing the basic operation of drawing a perfect sample by performing one step of an approximate sampling algorithm, such as the Metropolis-Hastings algorithm (propose and evaluate a potential adjustment to the current sample). A resource-rational analysis of this more complex abstract computational architecture predicts that the system should, in many situations, tolerate biased samples in exchange for spending less time computing (Lieder, Griffiths, & Goodman, 2013; Lieder, Goodman, & Griffiths, 2013). A model based on this analysis predicts a phenomenon that has been taken as evidence for human irrationality – anchoring and adjustment, in which people "anchor" on a first guess and "adjust" it insufficiently (Tversky & Kahneman,

1974). Taking into account the cost of computation reveals that this is exactly what an abstract computational architecture whose elementary operations are the propose and the accept/reject step of the Metropolis-Hastings algorithm should do.

These examples show how resource-rational analysis can reconcile rational models with the idea that human cognition is dominated by simple heuristics (Tversky & Kahneman, 1974; Gigerenzer & Todd, 1999). Resource-rational analysis of abstract computational architectures provides a rigorous way to derive the heuristics that represent a good compromise between accuracy and effort (see also Lieder et al., in press). Resource-rational analysis can also be used to understand how bounded agents can possibly solve extremely challenging problems, such as planning in a complex, partially unknown environment, and to explain how different styles of information processing arise from differences in beliefs, utility functions and cognitive resources (e.g., Lieder, Goodman, & Huys, 2013).

Resource-rational analysis opens up several avenues for future research. First, future studies may analyze and reverse-engineer increasingly more realistic models of mental computation, pushing rational analysis increasingly further down towards the algorithmic level. More realistic models of mental computation may provide elementary operations for searching through long-term memory, allocating attention, storing information in working-memory, comparison, modification and combination of information. More realistic abstract computational architectures may also comprise multiple, hierarchically structured sub-systems. Second, instead of assuming that the mind makes optimal use of its computational resources, future research may investigate whether and how the mind learns to compute resource-rationally. Metacognition is an extremely important aspect of intelligent behavior, and our framework makes it possible to formalize exactly what metacognition should do.

Relation to previous work

The idea of combining rationality and cognitive limitations to derive cognitive mechanisms is not new. For instance, the activation mechanism of the declarative memory modules of the ACT-R architecture is based on a rational analysis of memory (Anderson et al., 2004). However, in contrast to resource-rational analysis, rational analysis itself did not provide a recipe for deriving algorithms and did not clearly distinguish between different levels of analysis.

The theoretical frameworks closest to resource rationality are *boundedly rational analysis* (Icard, in press) and *computational rationality* (Lewis, Howes, & Singh, in press), which builds on cognitively bounded rational analysis (Howes, Lewis, & Vera, 2009). Resource rationality, boundedly rational analysis, and computational rationality are all mathematical frameworks that span levels of analysis and were inspired by Russell's (1997) notion of *bounded optimality*. Consequently, all three approaches construe agents as acting optimally subject to cognitively-motivated constraints. Resource rationality and boundedly rational analysis differ from computational rationality in how they characterize the constraints and the problem being solved. Concretely, they characterize the constraints by the computational costs imposed by an abstract computational architecture, and the problem being solved is construed as optimizing a function that combines accuracy and computational cost. In the limit where these costs become negligible, they recover the pure computational-level account. Resource-rationality also leverages the insights that approximation algorithms from computer science, machine learning, and statistics give into resource-efficient computation to derive hypotheses about the mind's abstract computational architecture. By contrast, computational rationality characterizes constraints by concrete psychological assumptions about an underlying cognitive architecture, and the problem being solved is construed as finding the best strategy supported by that architecture. In this framework, whether or not computational cost is taken into account depends on their impact on behavior. Another difference is that resource rationality formalizes optimal resource-allocation as the solution to a *rational meta-reasoning* problem (Russell & Wefald, 1991b), providing further links

to a literature on meta-reasoning in artificial intelligence research.

Halpern and Pass (2011) have developed a concept similar to resource-rationality, which they call *algorithmic rationality*. However, they apply it primarily in the context of game theory. By contrast, our emphasis is on leveraging computational-level theories and approximation algorithms to develop psychological process models of individual cognition.

So far we have only considered the cost of time, but resource-rational analysis can also incorporate metabolic costs (Gershman & Wilson, 2010) and mental opportunity costs (cf. Kurzban, Duckworth, Kable, & Myers, 2013). Although the rational process models presented above were inspired by sampling algorithms, resource-rational analysis can also leverage variational inference (Gershman & Wilson, 2010; Sanborn & Silva, 2013) and other approximation algorithms.

Conclusion

The principle of rationality has tended to be associated with the idea of developing models of cognition at the computational level, but we think it can be applied more broadly. Once we take into account that having to act in real-time limits how much computation an agent can perform, rationality becomes a fundamental tool for building a bridge from the computational to the algorithmic level. Whether it is asymptotic rationality, as assumed in rational process models, or the rational use of finite resources, as assumed in resource-rational models, we see this principle as playing a major role in the refinement of the notion of levels of analysis as the needs of cognitive science change.

While our focus here has been on algorithms, it is worth noting that a similar kind of argument can be made about representations (cf. Tenenbaum et al., 2011). As has been pointed out in the past (Anderson, 1978), algorithms and representations go hand in hand. This is an important point in the context of recent debates about Bayesian models of cognition (e.g., Griffiths et al., 2010; McClelland et al., 2010), in which the fundamental disagreement is not about whether

human minds and brains use statistics, but what kinds of representations those statistics are computed over. Developing a similarly rigorous treatment of the implications of different representational assumptions at the computational level for models at the algorithmic level would be an important step towards resolving these debates.

References

Abbott, J. T., & Griffiths, T. L. (2011). Exploring the influence of particle filter parameters on order effects in causal learning. In *Proceedings of the 33rd Annual Conference of the Cognitive Science Society.* Austin, TX: Cognitive Science Society.

Anderson, J. R. (1978). Arguments concerning representations for mental imagery. *Psychological Review*, *85*(4), 249.

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

Anderson, J. R. (1991). The adaptive nature of human categorization. *Psychological Review*, *98*(3), 409–429.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036–1060.

Arbib, M. A. (1987). Levels of modeling of mechanisms of visually guided behavior. *Behavioral and Brain Sciences*, *10*(03), 407–436.

Bowers, J. S., & Davis, C. J. (2012). Bayesian just-so stories in psychology and neuroscience. *Psychological Bulletin*, *138*(3), 389–414.

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

Gershman, S., & Wilson, R. (2010). The neural costs of optimal control. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. Zemel, & A. Culotta (Eds.), *Advances in Neural Information Processing Systems 23* (pp. 712–720). Red Hook, NY: Curran Associates, Inc.

Gigerenzer, G., & Todd, P. (1999). *Simple heuristics that make us smart*. New York: Ocford University Press.

Griffiths, T. L., Chater, N., Kemp, C., Perfors, A., & Tenenbaum, J. B. (2010). Probabilistic models of cognition: exploring representations and inductive biases. *Trends in Cognitive Sciences*, *14*(8), 357–364.

Halpern, J. Y., & Pass, R. (2011). Algorithmic rationality: Adding cost of computation to game theory. *ACM SIGecom Exchanges*, *10*(2), 9–15.

Hay, N., Russell, S., Tolpin, D., & Shimony, S. (2012). Selecting computations: Theory and applications. In N. de Freitas & K. Murphy (Eds.), *Uncertainty in Artificial Intelligence: Proceedings of the Twenty-Eighth Conference.* Corvallis, OR: AUAI Press.

Herrnstein, R. J., & Loveland, D. H. (1975). Maximizing and matching on concurrent ratio schedules1. *Journal of the Experimental Analysis of Behavior*, *24*(1), 107–116.

Horvitz, E. J. (1988). Reasoning under varying and uncertain resource constraints. In *Proceedings of the Seventh National Conference on Artificial Intelligence* (pp. 111–116). Menlo Park, CA: The AAAI Press.

Howes, A., Lewis, R. L., & Vera, A. (2009). Rational adaptation under task and processing constraints: Implications for testing theories of cognition and action. *Psychological Review*, *116*(4), 717–751.

Icard, T. (in press). Toward boundedly rational analysis. In P. Bello, M. Guarini, M. McShane, & B. Scassellati (Eds.), *Proceedings of the 36th Annual Conference of the Cognitive Science Society.* Austin, TX: Cognitive Science Society.

Jones, M., & Love, B. C. (2011). Bayesian fundamentalism or enlightenment? on the explanatory status and theoretical contributions of Bayesian models of cognition. *Behavioral and Brain Sciences*, *34*(4), 169–188.

Kurzban, R., Duckworth, A., Kable, J. W., & Myers, J. (2013). An opportunity cost model of subjective effort and task performance. *Behavioral and Brain Sciences*, *36*, 661–679.

Laird, J. (1987). SOAR: An architecture for general intelligence. *Artificial Intelligence*, *33*(1), 1–64.

Lewis, R. L., Howes, A., & Singh, S. (in press). Computational rationality: linking mechanism and behavior through bounded utility maximization. *Topics in Cognitive Science*.

Lieder, F., Goodman, N. D., & Griffiths, T. L. (2013). *Reverse-engineering resource-efficient*

algorithms [Paper presented at NIPS-2013 Workshop Resource-Efficient ML, Lake Tahoe, USA].

Lieder, F., Goodman, N. D., & Huys, Q. J. M. (2013). Controllability and resource-rational planning. *Cosyne Abstracts 2013*.

Lieder, F., Griffiths, T. L., & Goodman, N. D. (2013). Burn-in, bias, and the rationality of anchoring. In P. Bartlett, F. C. N. Pereira, L. Bottou, C. J. C. Burges, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems 26.* Red Hook, NY: Curran Associates, Inc.

Lieder, F., Hsu, M., & Griffiths, T. L. (in press). The high availability of extreme events serves resource-rational decision-making. In *Proceedings of the 36th Annual Conference of the Cognitive Science Society.* Austin, TX: Cognitive Science Society.

Marr, D. (1982). *Vision*. San Francisco, CA: W. H. Freeman.

Marr, D., & Poggio, T. (1977). From understanding computation to understanding neural circuitry. *Neurosciences Research Program Bulletin*(15), 470-488.

McClelland, J. L., Botvinick, M. M., Noelle, D. C., Plaut, D. C., Rogers, T. T., Seidenberg, M. S., & Smith, L. B. (2010). Letting structure emerge: connectionist and dynamical systems approaches to cognition. *Trends in Cognitive Sciences*, *14*(8), 348–356.

Meyer, D. E., & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance: Part i. basic mechanisms. *Psychological Review*, *104*(1), 3–65.

Newell, A. (1982). The knowledge level. *Artificial intelligence*, *18*(1), 87–127.

Pylyshyn, Z. W. (1984). *Computation and cognition*. Cambridge, MA: Bradford Books, MIT Press.

Russell, S. J. (1997). Rationality and intelligence. *Artificial Intelligence*, *94*(1-2), 57–77.

Russell, S. J., & Wefald, E. (1991a). *Do the right thing: studies in limited rationality*. Cambridge, MA: MIT press.

Russell, S. J., & Wefald, E. (1991b). Principles of metareasoning. *Artificial Intelligence*, *49*(1-3),

361–395.

Sanborn, A. N., Griffiths, T. L., & Navarro, D. J. (2006). A more rational model of categorization. In *Proceedings of the 28th Annual Conference of the Cognitive Science Society.* Mahwah, NJ: Erlbaum.

Sanborn, A. N., Griffiths, T. L., & Navarro, D. J. (2010). Rational approximations to rational models: Alternative algorithms for category learning. *Psychological Review*, *117*(4), 1144–1167.

Sanborn, A. N., Mansinghka, V. K., & Griffiths, T. L. (2013). Reconciling intuitive physics and Newtonian mechanics for colliding objects. *Psychological review*, *120*(2), 411.

Sanborn, A. N., & Silva, R. (2013). Constraining bridges between levels of analysis: A computational justification for locally Bayesian learning. *Journal of Mathematical Psychology*, *57*(3), 94–106.

Shi, L., Feldman, N., & Griffiths, T. L. (2008). Performing Bayesian inference with exemplar models. In *Proceedings of the Thirtieth Annual Conference of the Cognitive Science Society.* Austin, TX: Cognitive Science Society.

Shi, L., & Griffiths, T. (2009). Neural implementation of hierarchical Bayesian inference by importance sampling. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, & A. Culotta (Eds.), *Advances in Neural Information Processing Systems 22* (pp. 1669–1677). Red Hook, NY: Curran Associates, Inc.

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*(4), 443–464.

Simon, H. A. (1955). A behavioral model of rational choice. *The Quarterly Journal of Economics*, *69*(1), 99–118.

Tenenbaum, J. B., Kemp, C., Griffiths, T. L., & Goodman, N. D. (2011). How to grow a mind: Statistics, structure, and abstraction. *Science*, *331*(6022), 1279–1285.

Turing, A. M. (1937). On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, *2 (42)*, 345–363.

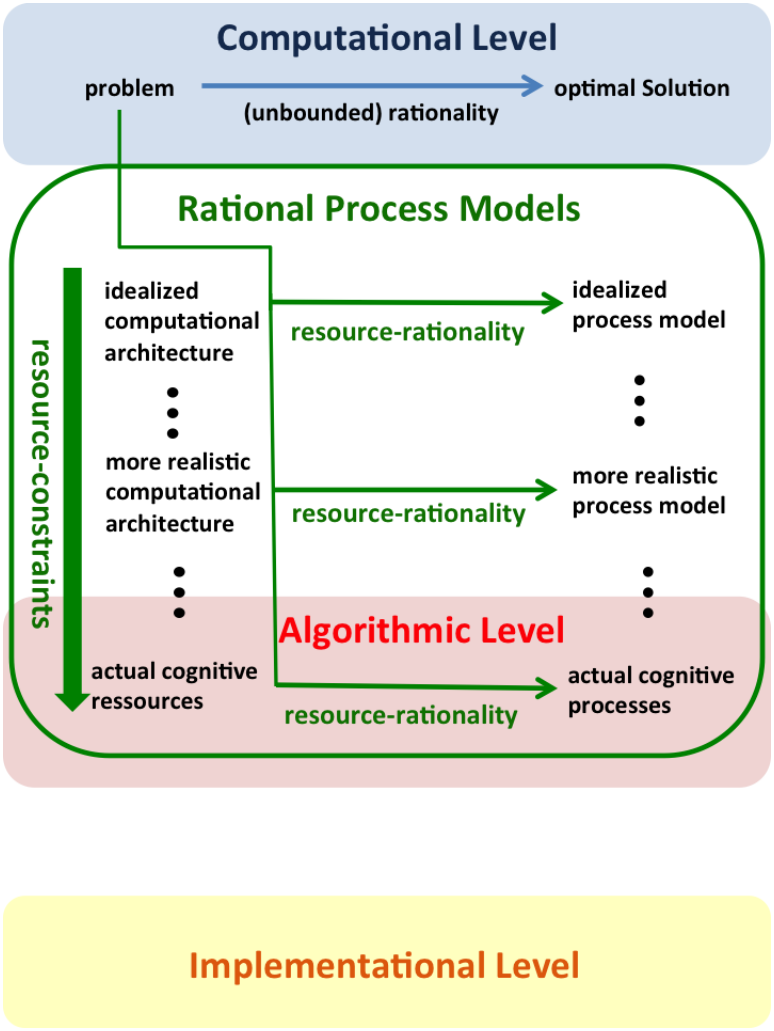
Tversky, A., & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science*, *185*(4157), 1124–1131.

Vul, E., Goodman, N., Griffiths, T. L., & Tenenbaum, J. B. (2014). One and done? Optimal decisions from very few samples. *Cognitive science*, *38*(4), 599–637.

Vul, E., Goodman, N. D., Griffiths, T. L., & Tenenbaum, J. B. (2009). One and done? Optimal decisions from very few samples. In N. Taatgen & H. van Rijn (Eds.), *Proceedings of the 31st Annual Conference of the Cognitive Science Society*. Austin, TX: Cognitive Science Society.

Figure Captions

Figure 1. The algorithmic level of analysis takes into account the cognitive resources available for solving a problem, but resource constraints are not normally part of analyses at the computational level. This gap can be bridged by considering a series of increasingly more realistic models of mental computation (abstract computational architectures), and resource-rational analysis can be used to derive the algorithm that makes optimal use of computational resources assumed by each of these models. The resulting algorithm can then be interpreted as a rational process model. As the model of mental computation becomes more realistic the resulting resource-rational models become psychological process models.


Computational Level
problem
optimal Solution
(unbounded) rationality
Rational Process Models
idealized computational architecture
resource-rationality
idealized process model
more realistic computational architecture
resource-rationality
more realistic process model
resource-constraints
Algorithmic Level
actual cognitive ressources
resource-rationality
actual cognitive processes
Implementational Level