# Metareasoning

Stuart J. Russell
Computer Science Division
University of California, Berkeley

November 21, 1997

Metareasoning is "reasoning about reasoning"—in its broadest sense, any computational process that is concerned with the operation of some other computational process within the same entity (see also METACOGNITION). The term relies on a conceptual distinction between *object-level* deliberation about *external* entities—for example, considering the merits of various opening moves one might make in a game of chess—and *metalevel* deliberation about internal entities (computations, beliefs, and so on)—for example, deciding that it is not worth spending much time deliberating about which opening move to make. Genesereth and Nilsson (1987, Ch. ??) provide formal definitions along these lines. Smith (1986) makes a further distinction between *introspection* about purely internal entities and *reflection* relating internal and external entities. In this view, a proposition such as, "If I open the window I will know if the birds are singing" is reflective, since it relates a physical action to a future state of knowledge (see also MODAL LOGIC).

The capacity for metareasoning serves several purposes in an intelligent agent. First, it allows the agent to *control* its object-level deliberations—to decide which ones to undertake and when to stop deliberating and act. This is essential given the pervasive problem of COMPUTATIONAL COMPLEXITY in decision making and the consequent need for BOUNDED RATIONALITY. In GAME-PLAYING SYSTEMS, for example, the *alpha-beta* algorithm makes a simple metalevel decision to eschew certain lines of deliberation about future moves, taking advantage of a metalevel theorem to the effect that these lines cannot affect the ultimate object-level decision. Second, it allows the agent to *generate* computational and physical behaviors, such as planning to obtain information, that require introspective or reflective reasoning. Third, it allows the agent to recover from errors or impasses in its object-level deliberations.

Most early work on metareasoning focused on designing an INTELLIGENT AGENT ARCHITECTURE (see also COGNITIVE ARCHITECTURE) that could support introspection and reflection. The idea of metareasoning to control deduction seems to have been proposed first by Hayes (1973), although the first implementation was in the TEIRESIAS system (Davis, 1980) which used metarules to control deliberation within a rule-based EXPERT SYSTEM. The MRS system (Genesereth & Smith, 1981) used LOGIC PROGRAMMING for both object and metalevel inference and provided a very flexible interface between the two. Because MRS allowed reasoning about which procedure to use for each object-level inference, and about which representation to use for each object-level fact, it enabled many different representations and reasoning methods to operate together seamlessly. By far the most ambitious metalevel architecture is SOAR (Laird, Newell, & Rosenbloom, 1987), which is based on PROBLEM SOLVING as its fundamental mode of computation. Whenever SOAR does not have an unambiguous rule telling it which problem-solving step to take next, it invokes *universal subgoaling* to set up a metalevel problem space that will resolve the issue. As might be imagined from these examples, designers of such systems must take care to avoid an *infinite regress* of metameta. . .reasoning.

Does metareasoning differ from "ordinary" reasoning? In all metalevel architectures, the metalevel is given direct access to object-level data structures. Thus, metareasoning (at least in computers) can assume a completely and perfectly observable object-level state—seldom the case with ordinary reasoning about the external world. Furthermore, it is possible to represent fully and exactly the nature of the available object-level computations. Thus, it is possible for the metalevel to simulate completely the object-level computations under consideration (as is done in SOAR). However, as a way of selecting among object-level computations this would seem counterproductive, since simulating a computation (and hence knowing its outcome) is just a very slow way of doing the computation itself—knowledge of the outcome of a computation *is* the outcome! For this reason, SOAR always compiles the results of subgoaling into a new rule, thereby avoiding deliberation in similar cases in future. Compilation of metareasoning into more efficient forms is perhaps the principal way in which an agent's computational performance can improve over time.

The research outlined in the preceding paragraphs established the basic mechanics of metareasoning. In most cases, however, the metareasoning consisted of the application of simple "IF–THEN" rules encoding the system designer's computational expertise; no standard of rationality for metareasoning was provided. The concept of *rational metareasoning* (Horvitz, 1989; Russell & Wefald, 1989) had its roots in early work by I. J. Good (1971) on "Type II rationality" and in *information value theory* (Howard, 1966). The latter theory places a value on acquiring a piece of information based on the expected improvement in decision quality that results from its acquisition. A computation can be viewed as the process of making explicit some information that was previously implicit, and therefore value can be placed on computations in the same way. That is, a computation can be viewed as an action whose benefit is that it may result in better external decisions, and whose cost is the delay it incurs. Thus, given a model of the effects of computations and information about object-level utility, the metalevel can infer the value of computations. It can decide which computations to do and when computation should give way to action.

The simplest applications of rational metareasoning arise in the context of *anytime algo-*

*rithms* (Horvitz, 1987; Dean & Boddy, 1988), that is, algorithms that can be interrupted at any time and whose output quality improves continuously with time. Each such algorithm has an associated *performance profile* describing its output quality as a function of time. The availability of the profile makes the metalevel decision problem—of which algorithm to run and when to terminate—fairly trivial. The use of anytime algorithms has resulted in a widely applicable methodology for building complex, real-time decision-making systems (Zilberstein & Russell, 1996), and anytime algorithms have been devised for a wide variety of computational tasks.

A more fine-grained approach to metareasoning can be obtained by evaluating individual computation steps within an algorithm. Consider the decision-making situation shown in Figure 1a. An agent has two possible actions, $A$ and $B$. Based on a quick assessment, the outcome of $A$ appears to be worth 10 with a standard deviation of 1, whereas the outcome of $B$ seems to be worth 8 with a standard deviation of 4. The agent can choose A immediately, or it can refine its estimates by looking further into the future. For example (Figure 1b), it can consider the actions $B_1$ and $B_2$, with the outcomes shown. At this point, action $B$ (followed by $B_1$) seems to lead to a state with value 12; thus, the lookahead computation has changed the agent's decision, with an apparent benefit of 2. Obviously, this is a *post hoc* analysis, but, as shown by Russell and Wefald (1991), an expected value of computation can be computed efficiently—*prior* to performing the lookahead. In Figure 1a, this value is 0.3 for lookahead from $A$ and 0.82 for lookahead from $B$ [[check]]. If the initial estimated outcome of $A$ were 12, however, these values would drop to 0.002 and 0.06 respectively—hence, as one would expect, the value of computation depends strongly on whether a clear choice of action has already emerged. If, however, the initial estimates for $A$ and $B$ were both 10, with standard deviations of 0.1, then the value of computation becomes 0.03—computation is worthless when it doesn't matter which action one eventually chooses.

Rational metareasoning can be applied to control deliberations in a wide variety of object-

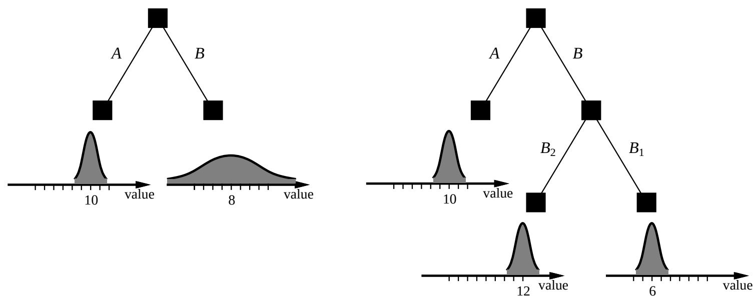


Figure 1: The consequences of computation: lookahead reveals that B may in fact be better than A.

level algorithms including HEURISTIC SEARCH and GAME-PLAYING (Russell & Wefald, 1991), LOGICAL REASONING SYSTEMS (Smith, 1989), and MACHINE LEARNING (Rivest & Sloan, 1988). An important insight to emerge from this work is that a metareasoning capability can, in principle, be *domain-independent* (Russell & Wefald, 1991; Ginsberg & Geddis, 1991) since the necessary domain-specific information (such as the utility function) can be extracted from the object level. One can therefore view successful computational behavior as emerging not from carefully crafted, domain-specific algorithms but from the interaction of a general capacity for rational metareasoning with object-level domain knowledge. More efficient, domain-specific computational behaviors might then result from processes of compilation and metalevel REINFORCEMENT LEARNING.

# References

Davis, R. (1980). Meta-rules: reasoning about control. *Artificial Intelligence*, *15*(3), 179–222.

Dean, T., & Boddy, M. (1988). An analysis of time-dependent planning. In *Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI-88)*, pp. 49–54 St. Paul, Minnesota. Morgan Kaufmann.

Genesereth, M. R., & Smith, D. (1981). Meta-level architecture. Memo HPP-81-6, Computer Science Department, Stanford University, Stanford, California.

Genesereth, M. R., & Nilsson, N. J. (1987). *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann, San Mateo, California.

Ginsberg, M. L., & Geddis, D. F. (1991). Is there any need for domain-dependent control information?. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91)*, Vol. 1, pp. 452–457 Anaheim, California. AAAI Press.

Good, I. J. (1971). Twenty-seven principles of rationality.. In Godambe, V. P., & Sprott, D. A. (Eds.), *Foundations of Statistical Inference*, pp. 108–141. Holt, Rinehart, Winston, Toronto.

Hayes, P. J. (1973). Computation and deduction. In *Proceedings of the Second Symposium on Mathematical Foundations of Computer Science* Czechoslovakia. Czechoslovakian Academy of Science.

Horvitz, E. J. (1987). Problem-solving design: Reasoning about computational value, tradeoffs, and resources. In *Proceedings of the Second Annual NASA Research Forum*, pp. 26–43 Moffett Field, California. NASA Ames Research Center.

Horvitz, E. J. (1989). Reasoning about beliefs and actions under computational resource constraints. In Kanal, L. N., Levitt, T. S., & Lemmer, J. F. (Eds.), *Uncertainty in Artificial Intelligence 3*, pp. 301–324. Elsevier/North-Holland, Amsterdam, London, New York.

Howard, R. A. (1966). Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, *SSC-2*, 22–26.

Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). SOAR: an architecture for general intelligence. *Artificial Intelligence*, *33*(1), 1–64.

Rivest, R. L., & Sloan, R. (1988). [a new model for inductive inference. In Vardi, M. (Ed.), *Proceedings of the Second Conference onTheoretical aspects of reasoning about knowledge*, pp. 13–27 San Mateo, California. Morgan Kaufmann.

Russell, S. J., & Wefald, E. H. (1989). On optimal game-tree search using rational meta-reasoning. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89)*, pp. 334–340 Detroit, Michigan. Morgan Kaufmann.

Russell, S. J., & Wefald, E. H. (1991). *Do the Right Thing: Studies in Limited Rationality.* MIT Press, Cambridge, Massachusetts.

Smith, B. C. (1986). Varieties of self-reference. In Halpern, J. (Ed.), *Theoretical aspects of reasoning about knowledge* San Mateo, California. Morgan Kaufmann.

Smith, D. E. (1989). Controlling backward inference. *Artificial Intelligence*, *39*(2), 145–208.

Zilberstein, S., & Russell, S. (1996). Optimal composition of real-time systems. *Artificial Intelligence*, *83*.