Formal Rationality and Limited Agents

Jonathan King Tash
Group in Logic and the Methodology of Science
University of California
Berkeley, CA 94720
tash@math.berkeley.edu

Abstract

Many efforts have been made to use normative theories of rational decision-making, such as Bayesian decision theory, to construct and model agents exhibiting intelligent behavior. In order to accommodate agents possessing only limited computational resources to apply to their decision making, however, a significant change is required in how the role of formal rationality is to be viewed. This paper argues that rationality is best seen as a property of the relationship between the agent and a designer. Such a perspective has several consequences for the design and modelling of agents, bearing on assessment of rationality, induction, reactivity, and metalevel control. It also illuminates several concerns put forth by critics of the work of the artificial intelligence community.

Introduction

There is a long and varied history of attempts to construct formal systems which will serve as a model for rational decision making. A powerful current paradigm, building on work of Savage (1972) and Jeffrey (1983), among others, has attempted to establish Bayesian decision theory as a framework for judging the rationality of action choice. Much recent work in artificial intelligence has the goal of implementing an agent capable of rational behavior on a computer (Doyle, 1990). Such work has made questions about the adequacy of formal modelling techniques more pointed. Some critics (for example, Winograd and Flores (1987) and Dreyfus (1992)) have expressed concern that starting with a specification of a formal problem, which can be attacked by an approach such as decision theory, bypasses much of the important work of an intelligent agent. However, this paper will consider issues which arise even with acceptance of the adequacy of formal statements to capture the relevant aspects of problems requiring intelligent solution, and analyze the coherence of using a formal concept of rationality in this arena. There are many interesting problems, such as chess strategy or route-finding given a formal map representation, where an analysis of the requirements of rational thinking about the problem remains interesting and deep even though availability of formal representation is not an issue. This approach will lead to an informative new perspective on some of the complaints voiced by AI's critics.

Rational Agents

Let us consider agents which are said to behave rationally if they choose their actions so as to satisfy their goals as best they can. We can begin with the formal framework for discussing rational action choice provided by Bayesian decision theory. Following this framework, we can assume that the agent has available to it at each time a set of possible actions, a utility function specifying the value of the various possible projected histories of the world to the agent, and probabilistic knowledge representing the various possible states of the world given particular action choices and the agent's assessment of their likelihood. One can define a plan for the agent (as in (Tash 1993)) as a sequence (actually, a tree) of actions, where the action choice made at a given time can be conditionalized on all the information available to the agent at that time. Decision theory then recommends acting according to the plan of highest expected value, calculated by averaging the values of the possible resulting world histories using their probabilities, which are determined by composing the state probabilities given the action choices.

Given some representation of the required probabilistic and utility information, this is a complete specification of what actions the agent should perform (up to choices having no discernible impact on the agent's welfare) in order to reap the greatest expected reward. However, there are many domains where, although the required information is available, the computational task of determining the recommended plan is infeasible, especially when it may recommend taking an action immediately. For example, when playing speed chess against an able program, even when you know its specifications, and therefore have available the necessary information to maximize your chance of winning, such a maximization is not a viable option due to the enormity of the required calculations. Such considerations require a reassessment of the suitability of decision theory as a definition of rationality for agents having only limited computational resources available.

Limited Rationality

These concerns have been central to the field of artificial intelligence since its inception. Simon (1955) addresses the problem by introducing the concept of *bounded rationality*, achieved by demanding of an intelligent agent only "satisficing" rather than a full decision-theoretic optimization. This amounts to restricting possible utility functions to ones taking on a few discrete values (e.g. "adequate" and "inadequate"), obviating the need for probabilities by using minimax techniques (examining the worst case of the possible consequences), and looking at

only a subset of the plan space. This reduces the computational problem to simply finding a plan in the considered set whose actions lead in all cases to adequate consequences. Such a search makes fewer computational demands, but choosing adequacy levels and plan subsets which reasonably approximate the agent's task without requiring excessive computation is still a hard problem, and it is not clear that these reductions in decision complexity are either adequate or required by the computational constraints of a given agent. Simon does discuss choosing levels and subsets so as to make the computational burden of a reasonable level of complexity, but these choices are a decision problem not obviously simpler than the one started with. (Such control problems are discussed in the context of metalevel architectures below).

A more recent conception of how to respond to these issues is that of *limited rationality*, advanced by Russell and Wefald (1991). The problem is reformulated as one of designing an agent, subject to certain architectural constraints, which does as well as it can in the problem domain. This lifts the onus of doing complete computations of all the decision-theoretically relevant consequences of an agent's knowledge from its shoulders, because the goal is now to choose that agent specification within the given resource constraints which performs optimally. Such a formulation allows the designer of such agents to consider tradeoffs between complexity and decision quality leading to best use of the limited computational resources available to a situated agent.

Such a conception requires that a distinction be made between the roles of *agent* and of a *designer* external to the agent in determining the rationality of the agent's behavior. The agent can be viewed as a particular implementation of a particular algorithm, for which the concept of choice is not necessarily a natural one. It is the designer's choice of agent from a set of possibilities which is subject to the strictures of rationality. Such a viewpoint has far-reaching consequences for the role of formal systems of rationality in determining appropriate agent behavior.

One immediately apparent problem with considering such a proposal as a solution to the problem of limited rational agents is that the entire burden of decision-theoretic computation has not been removed, but merely shifted to the designer. The work of choosing an agent which will behave optimally in a given environment is not necessarily any easier than determining an optimal plan. Choosing a program for a given platform which will play the best possible game of speed chess can be as intractable as finding the speed chess strategy which would be ideal in the absence of resource constraints. If the designer has limited computational resources, how is it to rationally use them to decide upon a particular agent?

Metalevel Control

Consideration of a research programme which attempts to incorporate many of the characteristics we have assigned to the role of designer into the agent itself may be illuminating at this point. The idea is to construct an agent from two parts: a metalevel sub-agent and a base-level sub-agent. The metalevel sub-agent chooses the computational tasks to be performed by the base-level sub-agent in much the same way that the agent as a whole chooses its actions in the external world. Such an architecture is said to perform metalevel control of its reasoning, or metareasoning (Hacking, 1967; Russell & Wefald, 1991). From the perspective of the designer, it is an open question whether a metareasoning architecture provides the most rational use of the agent's limited computational resources, but it does at least provide a straightforward way of incorporating flexibility to perform the kind of decision quality vs. computational complexity tradeoffs probably necessary for efficient control.

Clearly, for such an architecture to provide useful control of the complexity of the total agent's computations, the metalevel cannot simply try the base level computations in order to determine which are the most useful for the base level to perform, as then the agent as a whole has still performed all of computations we were attempting to control. Therefore, in deciding among computations, the metalevel cannot use all of the deductively available knowledge of the agent (such as the results of deterministic computational procedures), as required by the rationality strictures imposed by formal systems such as decision theory. The metalevel must do something less computationally demanding. If it is to apply rational methods, as would be required if it is to absorb some of the responsibility of the designer for producing a rational agent, then these methods must be applied to a problem simpler than that facing the agent as a whole. Some details deductively available to the agent must be ignored, or abstracted away, by the metalevel.

Let us consider an example provided in an early discussion of metalevel computational control by Hacking (1967). He examines the situation of an agent required to gamble on the relative magnitudes of products of pairs of five-digit binary numbers. The agent has some probabilistic guesses as to the likelihood of different relative magnitudes, and some measure of the expected work of actually doing the multiplications. The metalevel must decide whether the cost of doing a particular multiplication outweighs the potential loss of incorrectly guessing the relative magnitudes. This description of the metalevel task embodies several assumptions about what elements in the agent's deductively available knowledge base are actually considered available to the metalevel. For example, the metalevel can do the required computations for assessing the work of multiplication, the potential loss due to guessing, and comparing them, but does not avail itself of the results of the multiplication in making its decision. The metalevel also does not compare the value of its own computations with their cost.

Hacking does recognize that in a case such as this, metalevel costs may be deemed necessary of control themselves, but implies that the computations required at each level of the generated metalevel hierarchy diminish on the way up to a point where they can be safely ignored at some level. However, in the example given, the metalevel algorithm seems more complicated (in specification, if not in required computational work) than either base-level algorithm (doing the multiplications or just guessing based on prior odds estimates), and only one of a large range of

possibilities constructible by abstracting away different parts of the agent's total knowledge. The meta-metalevel decision of an appropriate metalevel algorithm is even more difficult than that of the metalevel, unless it is simplified by appropriate abstractions which must be chosen by an even more complicated higher metalevel or by a designer.

Thus, the difficulty facing metalevel approaches regarding how to appropriately abstract the base-level task reintroduces the need for a designer external to the agent to settle questions about the rationality of its operation. Such approaches may be of use in designing agents exhibiting a certain flexibility of behavior, but are ultimately of no help in addressing issues of rationality for a limited agent as a whole. The presence of an external designer's perspective is essential for judgments of rationality.

The Roles of Agent and Designer

The claim that rational control procedures cannot be embodied within an agent without fundamental architectural decisions being provided from outside amounts to a claim that a formal system of rationality such as the one discussed here cannot be fully automated. The work that a formal system like decision theory can do is intrinsically of a different nature than that demanded of a situated, limited agent. A designer can apply such a system to the construction and judgment of an agent, but neither the designer nor the agent can apply a notion of rationality to itself. Formal rationality is definable only in the interaction of these two distinct perspectives.

In a metalevel architecture, a higher level sub-agent can be considered to be deciding rationally among procedures for use by the base-level agent, using something like decision theory, but with respect to undeliberated background assumptions abstracting the base-level procedures to a point of tractable rational comparison. In a similar way, whether a designer's resources are being put to rational use can only be determined by treating the designer as an agent and examining its undeliberated background abstraction assumptions, taking a perspective external to it in order to judge its performance in comparison with those of its alternatives. Determinations of rationality by the designer presume these background assumptions, and are only addressable to agents external to itself. Attempts to determine its own rationality effectively endow it with metalevel structure, subject to all the considerations and limitations thereof.

These two roles of agent and designer embody differing perspectives on a variety of issues important to concerns over the automatability and formal definability of intelligent behavior. Consider, for example, how they treat time and reactivity. An agent is an algorithm running in the world which acts as it does at some particular rate. Whether it uses its time rationally and efficiently is determined not by whether or not its algorithm has embedded within it some formal rules for rational behavior, but rather whether from the designer's perspective its behavior is better than that of alternative implementable algorithms. In contrast, the designer, in judging the rational use of time by the agent, is using a formalized system of rationality, and the time involved in such application is not an issue reflected in the criteria of judgment. Considerations of the efficiency of the designer only find their way in through the assumptions made in formalizing the problem of agent judgment, which are in turn only judgable from the viewpoint of one outside the designer, treating the designer as an agent. Rationality can only address itself to temporal control issues as these are nontemporally represented in agent description, not as they occur in the application of the system of rationality itself.

A similar clarification is afforded the issue of induction by this distinction between perspectives. Again, the agent simply exhibits some form of inductive behavior, and judgments as to its rationality are made with respect to a formal system possessed by the designer. If the designer is using Bayesian decision theory, methods for correct updating of beliefs in response to new information about the world are implicit in the prior. So the designer can judge the agent's use of induction using the deductive tools provided by his formal theory. A judgment of the appropriateness of the designer's prior can only be made with respect to an external designer's perspective, which contains its own background assumptions embodied in a prior.

Bayesian decision theory recommends believing in response to new information that which one previously believed would hold given the truth of this new information. It therefore only provides a logic for induction when all possible consequences of all possible observations have already been taken into account, included implicitly in the prior, and presumes the availability of this information in recommending a decision. In particular, its recommendations for which new information is worth seeking out are predicated on full knowledge of the deductive consequences of possessing that information. It cannot be used to judge the value of information provided by the deductions it needs to perform in judging value. Its use in controlling deductive effort is restricted to efforts not intrinsic to its own application, those of a separate agent whose computations can be modelled probabilistically, not those involved in working out the consequences of that model.

On Critiques of Formalized Intelligence

This interpretation of rationality as a relation between the roles of agent and designer provides an alternative perspective as well on some of the issues of concern to those offering "Heideggerian" critiques of artificial intelligence. A distinction commonly relied on in their arguments is that between a system embedded in the world, responding directly without a formal representation and theory to guide its activity, and one involved in symbolic representation and modelling of the world, using a formal, rational methodology for determining its actions. The former is described as *being-in-the-world* by Dreyfus (1992) and as the condition of *thrownness* by Winograd and Flores (1987). The distinction is held to cut between people and the kinds of systems developed by researchers in artificial intelligence. Various difficulties are encountered in trying to make a system of the latter sort with limited resources which can behave successfully in the world. One, discussed by Dreyfus, is the infinite regress and expansion encountered in trying to account for the contexts in which decision

making is to take place. (Notice that this is essentially the same difficulty found above in trying to determine appropriate abstractions for rational decision making by using a metalevel architecture.) Difficulties of this sort, involving use of a predetermined symbolic representation and manipulation scheme, are purported to prevent such an agent from exhibiting the flexibility of behavior required for dealing with all the vagaries of situations presented by the world. Another sort of difficulty, discussed by Winograd and Flores, is the different social role played by a computer in expressing the intentions and commitments of its designers, compared to that of the designers themselves, held responsible for their own actions. These difficulties are taken to show the impossibility of a system which uses formal methods to achieve intelligent behavior.

When formal rationality is seen as a property of the relation between a designer and an agent rather than a property of an agent itself, the above distinction takes on a different form. The embedded, situated nature of being-in-the-world, requiring response without recourse to a full rational justification of one's actions, is actually an apt description of the situation of any resource limited agent, those implemented by a designer on a computer as well as those engaged in such designing. It is only as interpreted from an external perspective that an agent can be said to be a rational, formal system, and the "limitations" such a system possesses are limitations of the interpretation rather than of the agent. Inadequacies in the behavioral capabilities of such an agent are *with respect to* the analysis afforded by a formal system of rationality, not *due to* its embodiment of such a system.

In fact, the roles of agent and designer are interpretational roles applied to systems in order to make sense of the relation between the strictures of rationality and agent behavior. Where the distinction is to be drawn is open to wide variance, depending on the agent whose behavior is to be the subject of rational analysis or control (reminiscent of the condition of psycho-physical parallelism on the division in physics between observer and observed, as described by von Neumann (1955)). We have seen above how in the context of metalevel architectures it is often useful to apply the role descriptions to two parts of the same computational agent. In contrast, the tools of rationalistic analysis are also of use in examining and modelling the behavior of other people, as is common in microeconomics. The distinction between formal and "thrown" systems does not cut neatly between the human and the computer.

This same flexibility in role assignment has implications for the ascription of commitment and intention to a system. The designer has the responsibility of judging the rationality of an agent, and hence its adequacy for a given task. Therefore, in the common situation formed by assigning the designer role to a person and the agent role to a machine, social commitment is indeed most naturally ascribed to the person, the machine being a means for fulfilling the person's social contracts. However, if the designer role is assigned to a metalevel within an agent, it can reasonably be spoken of as committed to the rational behavior of its controlled subagent. Only by revoking its status and considering it an agent whose rationality is to be tested from outside do we transfer its commitment to its designer.

Conclusion

We have seen that the rationality of an agent with only limited computational resources can only be determined from a perspective external to it, and that this conception has a variety of consequences for efforts to formally implement rational agents. On the one hand, the requirement of an external perspective for rationality judgments places limits on claims to normativeness on the part of particular architectures; except in rare cases where rational choice among agent designs presents a trivial computational burden, abstraction assumptions necessarily made in applying formal analysis to the agent are again challengeable from the perspective of one examining the rationality of limited resource use by the analyst. On the other hand, the resulting role distinction seems to capture many of the concerns of critics of the application of formal methods to intelligent agent design without forcing the same negative conclusions that they reach. The resulting characterization of the relationship between formal, normative methodologies and the design and modelling of intelligent agents will hopefully be of service in clarifying the nature of the task undertaken by the modellers and designers.

Acknowledgements

I would like to thank Stuart Russell and David Schultz for valuable comments on an earlier draft. This work was supported by NASA JPL through the Graduate Student Researchers Program.

References

Doyle, Jon (1990). Rationality and its roles in reasoning. In *Proceedings of the Eighth National Conference on Artificial Intelligence* (pp. 1093-1100). Boston: AAAI Press.

Dreyfus, H. (1992). *What Computers Still Can't Do.* Cambridge, MA: MIT Press.

Hacking, I. (1967). Slightly more realistic personal probability. *Philosophy of Science*, 34, 311-325.

Jeffrey, R. (1983). *The Logic of Decision*, 2nd Ed. Chicago, IL: University of Chicago Press.

Russell, S. & Wefald, E. (1991). *Do The Right Thing.* Cambridge, MA: MIT Press.

Savage, L. (1972). *The Foundations of Statistics.* New York, NY: Dover.

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

Tash, J. (1993). A framework for planning under uncertainty. In *Spring Symposium on Foundations of Automatic Planning*, AAAI Tech. Report.

von Neumann, J. (1955). *Mathematical Foundations of Quantum Mechanics.* Princeton, NJ: Princeton University Press.

Winograd, T. & Flores, F. (1987). *Understanding Computers and Cognition.* Reading, MA: Addison Wesley.