Reprinted from *Advanced Planning Technology: Technological Achievements of the ARPA/Rome Laboratory Planning Initiative* (Austin Tate, editor), Menlo Park, California: AAAI Press, 1996, pp. 130-135.

# Toward Rational Planning and Replanning
## Rational Reason Maintenance, Reasoning Economies, and Qualitative Preferences

**Jon Doyle**
Massachusetts Institute of Technology
Laboratory for Computer Science
545 Technology Square
Cambridge, Massachusetts 02139
http://www.medg.lcs.mit.edu/doyle
doyle@mit.edu

**Abstract**

Efficiency dictates that plans for large-scale distributed activities be revised incrementally, with parts of plans being revised only if the expected utility of identifying and revising the subplans improves on the expected utility of using the original plan. The problems of identifying and reconsidering the subplans affected by changed circumstances or goals are closely related to the problems of revising beliefs as new or changed information is gained. But traditional techniques of reason maintenance—the standard method for belief revision—choose revisions arbitrarily and enforce global notions of consistency and groundedness which may mean reconsidering all beliefs or plan elements at each step. To address these problems, we developed (1) revision methods aimed at revising only those beliefs and plans worth revising, and tolerating incoherence and ungroundedness when these are judged less detrimental than a costly revision effort, (2) an artificial market economy in planning and revision tasks for arriving at overall judgments of worth, and (3) a representation for qualitative preferences that permits capture of common forms of dominance information.

## Planning and replanning

We view the activities of intelligent agents as stemming from interleaved or simultaneous planning, replanning, execution, and observation subactivities. In this model of the plan construction process, the agents continually evaluate and revise their plans in light of what happens in the world.

Planning is necessary for the organization of large-scale activities because decisions about actions to be taken in the future have direct impact on what should be done in the shorter term. But even if well-constructed, the value of a plan decays as changing circumstances, resources, information, or objectives render the original course of action inappropriate. When changes occur before or during execution of the plan, it may be necessary to construct a new plan by starting from scratch or by revising a previous plan.

To replan effectively in demanding situations, replanning must be *incremental*, so that it modifies only the portions of the plan actually affected by the changes. Given the information accrued during plan execution, which remaining parts of the original plan should be salvaged and in what ways should other parts be changed? Incremental replanning first involves *localizing* the potential changes or conflicts by identifying the subset of the extant beliefs and plans in which they occur. It then involves *choosing* which of the identified beliefs and plans to keep and which to change. For greatest efficiency, the choices of what portion of the plan to revise and how to revise it should be based on coherent expectations about and preferences among the consequences of different alternatives so as to be *rational* in the sense of decision theory (Savage 1972).

Our work toward mechanizing rational planning and replanning has focussed on four main issues:

- Identifying formal notions of rationality appropriate to agents of limited mental resources and methods for evaluating specific reasoning architectures with respect to these notions,
- Developing methods for incremental revision of plans based on a new approach to reason maintenance,
- Exploring a market-based approach to allocation of planning resources that simultaneously balances mental resources against each other and against physical resources, and
- Developing logics of preferences that make acquisition and specification of planning preferences convenient.

This paper focusses on the latter three issues; for our approach to the first, see (Doyle 1988; 1992).

## Reason maintenance for replanning

Replanning in an incremental and local manner requires that the planning procedures routinely identify the assumptions made during planning and connect plan elements with these assumptions, so that replanning may seek to change only those portions of a plan dependent upon assumptions brought into question by new information. Consequently, the problem of revising plans to account for changed conditions has much in common with the problem of revising beliefs in light of new information. In both cases, one must determine which existing beliefs or plans conflict with the new information, on what these existing beliefs or plans depend, and what gaps in plans or beliefs appear as the revisions or updates are made. That is, one must localize the potential changes or conflicts by identifying the subset of the extant beliefs and plans in which they occur. Similarly, both belief revision and plan revision involve choosing which of the identified beliefs and plans to keep and which to change.

The standard approach to belief revision, backtracking, and default reasoning is to use a reason maintenance system (RMS) to connect original information with derived conclusions and assumptions. Reason maintenance may be used in a similar way to revise plans as well as beliefs by indicating the dependence of plans on beliefs and on other plans, thus indicating the relevant portions for revision and the conflicts between prior plans and new circumstances. This possibility was, in fact, one of the original motivations for reason maintenance systems (see (de Kleer *et al.* 1977; Doyle 1979)). However, to employ reason maintenance techniques as integral parts of the replanning process requires reassessing and rethinking most of the architectures for reason maintenance developed previously, for these architectures do not make rational choices, they do not distribute effort, and they do not fit cleanly into existing methods and architectures for planning.

Essentially all the choices made by traditional RMSs are irrational since they are made without reference to any preferential information about what choices are better than others. The most obvious decisions concern backtracking: whether observed conflicts warrant resolution and if so, which assumption (maximal or otherwise) to retract in order to resolve them. Approaches to each of these decisions play prominent roles in the design of different reason maintenance systems. But if we are to achieve the efficiency required for revising large plans, reason maintenance must be redesigned to make these choices rationally whenever possible. Accordingly, we developed formal foundations for the theory of rational belief revision (Doyle 1988; 1991). But to really achieve efficiency, the RMS must do more than choose rationally among assumptions in backtracking. It must in addition be much more incremental than the original architectures, which were based on making unbounded (potentially global) optimizing computations that in some cases may reconsider the status of *every* item in the plan and knowledge base, even though very few of these statuses may change as the result of the revision. Put another way, the original systems maintain global coherence (propositions are believed if and only if there is a valid argument for them) and global groundedness (all believed propositions have a well-founded argument from premises). While these unbounded computations are manageable in relatively small knowledge bases, they are infeasible for use in systems manipulating very large plans. Instead of global computations, the RMS must control how much effort is spent on revision and trade off coherence and groundedness for time or other resources. Specifically, it must be able to decide whether the benefits of updating some arguments or consequences justify the costs of updating them.

Traditional RMSs also centralize information and computation in ways that frustrate the goal of rational revision. The original architectures make reason maintenance the base-level stratum upon which all other reasoning procedures are erected. To enable reason maintenance, one must encode every bit of information that might change in reasons and tell these reasons to the RMS (cf. (Rich 1985; Vilain 1985)). This can present an excessive burden, as manifest by the observation that the RMSs supplied in expert system shells all too often go unused. If one must apply it to every step of reasoning, at every level down to the smallest inference, reason maintenance becomes a demanding duty rather than a flexible service to use or ignore as appropriate. To integrate existing application tools and systems that do not use reason maintenance into AI systems that do, the RMS must be able to use other databases and processes to effect its revisions. In particular, the RMS must be able to treat external databases as the authorities about certain beliefs, and it must be able to operate even though other processes may be changing these databases independently of the RMS. This makes the RMS just one of a set of distributed databases rather than a systemic authority.

## Rethinking reason maintenance

We retain the idea that a reason maintenance system keeps track of what information has been computed from what, reconstructs the information "derivable" from given information, and revises the database states of the overall system by using records of inferences or computations to trace the consequences of initial changes. But given the preceding observations, we take the purpose of the RMS to be to maintain a description of the overall system's state of belief that is as good as possible given the reasoner's purposes and resources. This description may be approximate, partial, or imperfect, and it may be improved by performing further computation as the resources supplied to the RMS increase. As with the original architectures, the RMS still provides explanations, a way of answering hypothetical questions, and a way of maintaining coherence, groundedness, and consistency (given enough resources and information), but its primary purpose is to enable the reuse of past computations in whole or in part without having to repeat the possibly lengthy searches that went into constructing their results. That is, we view reasons as information about past computations or conditions which may be used to reconstruct results in changed circumstances, either exactly or in modified form (as in derivational analogy (Carbonell 1986) or case-based reasoning). Treating reasons as aids to recomputation is in marked contrast with the traditional use of reasons as rigid requirements that belief states must satisfy instead of as information which may be used or ignored as suits the reasoner's purposes. Naturally, in this setting the RMS is not expected to determine completely and accurately what the system believes. Instead, it only offers a theory of the composition of the overall system state, but not necessarily a complete or correct one.

Given this purpose, the basic operation of the RMS is to record reasons and other information, and, when so instructed, to revise beliefs in accordance with the expectations and preferences supplied by the reasoner. Put another way, the default operation of the RMS is to ignore the information it records until it is told to revise beliefs, and then to revise them only as far as can be justified by purposes of the reasoner. In the absence of more specific instructions, the default revision is trivial, simply adding the new reasons and their immediate conclusions to the belief set. Thus without explicit instructions, the RMS does not propagate changes, does not ensure beliefs are grounded, and does not automatically backtrack to remove inconsistencies. We do not require that all inference be explicitly controlled. Some amount of automatic inference is acceptable if it represents strictly bounded amounts of processing.

To make the RMS amenable to rational control of the effort expended on revisions, we divide the knowledge base into parts, called *locales*, each of which may be revised or preserved separately. Each locale contains its own set of beliefs and plans (as well as other information) corresponding to different elements and purposes of the overall plan or to different dimensions of structure (hierarchical abstraction, overlapping views, spatial separation, temporal separation, flow of material and information, etc.). We define revision instructions relative to the locales of the knowledge base. These instructions may indicate that changes should propagate within the locale containing the belief, or to its neighbors, or globally; or that all beliefs in the locale should be grounded with respect to the locale, with respect to its neighbors, or globally; or that backtracking should be confined to the locale, or should look further afield for assumptions to change.

## Rethinking reasons

Reasons ordinarily supply only partial information in that the reasoner need not register all inferences with the RMS. In the extreme case, the external reasoners may command the RMS to simply believe some proposition, independent of reasons. This corresponds to the "revision" operation in philosophical treatments of belief revision (Gärdenfors 1988). Because of this partiality, the RMS will sometimes be unable to track all the consequences of all beliefs. Although knowledge is usually preferable to ignorance, this incompleteness of the beliefs of the RMS need not be detrimental since the underlying knowledge and inferences of the reasoner are incomplete anyway. Moreover, these consequences may not influence the reasoner's actions, in which case all effort expended in recording them would be wasted. The only discipline required of the reasoner is that any inferences that will not be performed by some other agency and that cannot be determined after the fact during backtracking should be described to the RMS.

Correspondingly, reasons may be incorrect in the RMS. That is, the reasoner may use a reason to describe the result of a computation, but may leave out some underlying assumptions. The result is a reason that is valid when those unstated assumptions hold, but which may be invalid otherwise. Incorrect reasons can be very troublesome in the traditional architectures, since they would be enforced as requirements on the state of belief, but they need not cause special problems in the new conception. Since the RMS may obey or ignore reasons depending on its instructions and experience, all reasons are implicitly defeasible. Thus incorrect reasons pose no problems not already present in explicitly defeasible nonmonotonic reasons.

Just as reasons may be incomplete, so may be the theories of mental states constructed from them, since if reasons are ignored, their consequences will not be believed. More generally, the RMS makes it possible to vary how many conclusions are drawn from reasons. For example, the system will ordinarily use reasons to construct a single global set of beliefs, as in the original conception. But for some specific sets of reasons, say those corresponding to a circumscribed problem, the RMS may determine all consistent sets of beliefs as in the ATMS (de Kleer 1986). Alternatively, only some consistent interpretations may be constructed, such as those maximal in some order (as in preferential nonmonotonic logics (Shoham 1988), though the standard handling of nonmonotonic reasons in the RMS yields Pareto-optimal interpretations with respect to a natural preferential view of the content of reasons (Doyle 1983; 1994b; Doyle & Wellman 1991)). In general, the aim is to use the recorded reasons to draw as many conclusions as the reasoner needs.

One consequence of the incompleteness and incorrectness of reasons is that beliefs of the system may be inconsistent in routine operation. The overall set of beliefs may exhibit inconsistencies by including conflicting beliefs from different locales. Ordinarily the specialized beliefs corresponding to specific problems or subjects will be represented in locales that are internally consistent, but the RMS need not be forced to keep all these locales consistent with each other. But inconsistency can arise even within a locale if too little inference is specified.

Another consequence is that the beliefs of the system may not be fully grounded. In the first place, the set of beliefs may be so large as to make global groundedness too costly. More fundamentally, large sets of beliefs always contain interderivable sets of propositions—alternative definitions provide the most common example—and which of these sets to choose as axioms can depend on the specific reasoning task being addressed. For example, the standard definition of nonplanar graphs is best for some purposes (e.g., teaching the concept), but Kuratowski's characterization is best for other purposes (e.g., recognition algorithms). Thus lack of global groundedness need not be cause for alarm. Ordinarily, however, specialized locales corresponding to specific problems will be kept grounded in the axioms formulating these problems. The system of beliefs can thus be thought of as "islands" of groundedness floating in a sea of ungrounded beliefs.

Since reasons merely record some of the inferential history of the reasoner, they do not by themselves determine whether consequences are updated or supports are checked. Instead, to make these decisions the RMS uses annotations supplied by the reasoner which give instructions, expectations, and preferences about alternative courses of action. These include specification of the conditions under which the RMS should pursue consequences and check support. For example, local propagation may be expressed as processing changes within the locale containing the changed belief, but not externally. Alternatively, changes might be communicated to neighboring locales (with or without local propagation). Other regimes are possible too, including the extreme of propagating the change globally. Similarly, the annotations may indicate to persist in believing the proposition without reevaluating the supporting reason, to check that the reason is not invalidated by beliefs within the containing locale, or to check validity with respect to external beliefs. We have developed in (Doyle 1983) and (Doyle 1994b) a formalization of this more general framework of reason maintenance that permits local variability of consequential import, groundedness, and other properties of RMS states.

It is this limited scope, variety, and fine grain of RMS operations, that makes RMS choices (which reasons to use in reconstructing results, whether to propagate changes, whether to ground a conclusion, and whether to backtrack) amenable to rational control. For decisions about updating consequences and checking support, it is important that the individual operations be well-characterized computationally. Domain knowledge of probabilities and preferences should also be reflected in the revision policies. Because such information is not always available, the architecture provides default choices for each of these classes of decisions. Each domain may override these with other defaults that are more appropriate in its specific area. These default choices are then used whenever there is no evidence that a decision requires special treatment.

See (Doyle 1994a) for a more detailed description of the structure and implementation of a prototype rational and distributed RMS.

## Market-guided reason maintenance

In order to provide rational guidance to distributed reason maintenance activities about how to allocate effort, we constructed a market-guided RMS, called MRMS, by combining the new RMS with an extension of the WALRAS computational economy developed by Wellman (Wellman 1993). Designed to make use of the main ideas of theoretical economics about markets, WALRAS provides a general mechanism for implementing markets in arbitrary sets of goods, traded by arbitrary consuming and producing agents. MRMS represents each significant revision task by means of a market good and a consumer of that good, and each revision method by a producer of a revision good. Effort is then allocated among these tasks on the basis of relative prices of these goods.

The "glue" used in this combination is a system called RECON, the Reasoning ECONomy (Doyle 1994c; 1994a). The reasoning economy extends WALRAS and determines rational allocations of the full range of resources, computational and otherwise, by determining prices or trading ratios among resources. RECON builds on WALRAS by augmenting WALRAS's generic notions of goods, consumers, or producers with notions specific to reasoning tasks, and by introducing computation goods to allocate to different tasks. Its main additions to WALRAS are a good representing computation, the notions of computation consumers and producers, a taxonomy of action types, and an execution mechanism that takes actions in an order determined by bids for computation. The initial approach to allocating effort was based on an auction of computation opportunities, while a later approach developed by Nathaniel Bogan (1994) conducts auctions to rent computational resources to the highest bidders.

## Representing planning and reasoning preferences

Preferences constitute one of the basic elements of information about economic agents, and a key problem in both our investigation of market-guided reason maintenance and in rational planning in general is finding a good representation of preference information. Decision-theoretic treatments of preferences represent the objectives of a decision maker by an ordering over the possible outcomes of available plans. We view this ordering relation as an ideal, but cannot hope to completely and directly encode it in a planning system, as the domain of outcomes is combinatorially large or infinite, and the relevant preference criteria vary across problem instances. Therefore, in designing preference languages, we seek constructs for describing general *patterns* of preference that hold over *classes* of outcomes and situations. Toward this end, we have developed in collaboration with Michael Wellman a qualitative logic of preference *ceteris paribus* or preference "other things equal" (Wellman & Doyle 1991; Doyle, Shoham, & Wellman 1991; Doyle & Wellman 1994). This logic affords flexible specification of objectives, underpinned by a decision-theoretic semantics.

Our qualitative logic of preference *ceteris paribus* provides a uniform language in which one can express both ordinary decision-theoretic preferences as well as the standard notion of goal, which we interpret to mean conditions preferred to their opposites other things equal. We are continuing to develop the theoretical structure and inferential capabilities of this logic and some close variants, but the basic language already provides a useful tool for encoding qualitative preferential information. Of course, a rich language for encoding preference information would include quantitative representations as well, and we are working toward a preference language that spans the spectrum from completely qualitative representations like our language of comparative preference to ordinary numeric utility functions, including intermediate representations of multiattribute utility functions such as subutility composition trees (Wellman & Doyle 1992), the standard forms of multiattribute utility functions, and their application to expressing different types of planning goals (Haddawy & Hanks 1990; 1992).

## Conclusion

Reason maintenance offers important abilities for use in planning and replanning, but to prove useful for large-scale activities, the techniques must be capable of incremental application that does not incur the costs of global reconsideration. We extended traditional reason maintenance techniques to make use of instructions, expectations, preferences, and market mechanisms in deciding how to establish and revise beliefs and plan elements. In our conception, the *rational distributed reason maintenance service* maintains only as much coherence and grounded support as is called for by the planner's purposes. In essence, the fundamental operations of finding supporting arguments and pursuing consequences become flexible rather than routine, with different sorts of reasons indicating different sorts of processing during revisions in addition to their more overt indications of likelihood and preference information.

### Acknowledgments

Much of this work, and especially the work on representing preferences, was conducted in collaboration with Michael Wellman, who coauthored some of the papers on which this report draws, and who originated the WALRAS system. The research described here was supported by ARPA and Rome Laboratory under contract F30602-91-C-0018. Preparation of this paper was supported by ARPA and the Naval Command Control Ocean Surveillance Center under contract number N66001-95-D-6019.

## References

Bogan, N. R. 1994. Economic allocation of computation time with computation markets. Master's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139.

Carbonell, J. G. 1986. Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In Michalski, R. S.; Carbonell, J. G.; and Mitchell, T. M., eds., *Machine Learning: An Artificial Intelligence Approach*, volume 2. Morgan Kaufmann. 371–392.

de Kleer, J.; Doyle, J.; Steele Jr., G. L.; and Sussman, G. J. 1977. AMORD: Explicit control of reasoning. In *Proceedings of the ACM Symposium on Artificial Intelligence and Programming Languages*, 116–125.

de Kleer, J. 1986. An assumption-based TMS. *Artificial Intelligence* 28:127–162.

Doyle, J., and Wellman, M. P. 1990. Rational distributed reason maintenance for planning and replanning of large-scale activities. In Sycara, K., ed., *Proceedings of the DARPA Workshop on Planning and Scheduling*, 28–36. San Mateo, CA: Morgan Kaufmann.

Doyle, J., and Wellman, M. P. 1991. Impediments to universal preference-based default theories. *Artificial Intelligence* 49(1-3):97–128.

Doyle, J., and Wellman, M. P. 1994. Representing preferences as *ceteris paribus* comparatives. In Hanks, S.; Russell, S.; and Wellman, M. P., eds., *Proceedings of the AAAI Spring Symposium on Decision-Theoretic Planning.*

Doyle, J.; Shoham, Y.; and Wellman, M. P. 1991. A logic of relative desire (preliminary report). In Ras, Z. W., and Zemankova, M., eds., *Methodologies for Intelligent Systems, 6*, volume 542 of *Lecture Notes in Artificial Intelligence*, 16–31. Berlin: Springer-Verlag.

Doyle, J. 1979. A truth maintenance system. *Artificial Intelligence* 12(2):231–272.

Doyle, J. 1983. Some theories of reasoned assumptions: An essay in rational psychology. Technical Report 83-125, Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA.

Doyle, J. 1988. Artificial intelligence and rational self-government. Technical Report CS-88-124, Carnegie-Mellon University Computer Science Department.

Doyle, J. 1991. Rational belief revision (preliminary report). In Fikes, R. E., and Sandewall, E., eds., *Proceedings of the Second Conference on Principles of Knowledge Representation and Reasoning*, 163–174. San Mateo, CA: Morgan Kaufmann.

Doyle, J. 1992. Rationality and its roles in reasoning. *Computational Intelligence* 8(2):376–409.

Doyle, J. 1994a. Final report on rational distributed reason maintenance for planning and replanning of large-scale activities. Submitted to Rome Laboratory, available via http://www.medg.lcs.mit.edu/doyle.

Doyle, J. 1994b. Reasoned assumptions and rational psychology. *Fundamenta Informaticae* 20(1-3):35–73.

Doyle, J. 1994c. A reasoning economy for planning and replanning. In Burstein, M. H., ed., *Proceedings of the 1994 ARPA/Rome Laboratory Knowledge-Based Planning and Scheduling Initiative Workshop*, 35–43. San Francisco, CA: Morgan Kaufmann.

Gärdenfors, P. 1988. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. Cambridge, MA: MIT Press.

Haddawy, P., and Hanks, S. 1990. Issues in decision-theoretic planning: Symbolic goals and numeric utilities. In *Proceedings of the DARPA Workshop on Innovative Approaches to Planning, Scheduling, and Control*, 48–58.

Haddawy, P., and Hanks, S. 1992. Representations for decision-theoretic planning: Utility functions for deadline goals. In Nebel, B.; Rich, C.; and Swartout, W., eds., *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*, 71–82. San Mateo, CA: Morgan Kaufmann.

Rich, C. 1985. The layered architecture of a system for reasoning about programs. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*.

Savage, L. J. 1972. *The Foundations of Statistics*. New York: Dover Publications, second edition.

Shoham, Y. 1988. *Reasoning about Change: Time and Causation from the Standpoint of Artificial Intelligence*. Cambridge, MA: MIT Press.

Vilain, M. B. 1985. The restricted language architecture of a hybrid representation system. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, 547–551.

Wellman, M. P., and Doyle, J. 1991. Preferential semantics for goals. In *Proceedings of the National Conference on Artificial Intelligence*, 698–703.

Wellman, M. P., and Doyle, J. 1992. Modular utility representation for decision-theoretic planning. In *Proceedings of the First International Conference on AI Planning Systems*.

Wellman, M. P. 1993. A market-oriented programming environment and its application to distributed multicommodity flow problems. *Journal of Artificial Intelligence Research* 1:1–23.